SIC4310

228-Byte ISO14443A RFID/NFC Tag IC with UART Interface

REV 1.9

Features Summary

Highlight Features

- Write/read through NFC smartphone/RFID NFC/RFID reader device
- Direct data transfer from RFID to UART or vice versa
- Operating from either RFID power or external DC
- 3.3-V On-chip regulator for power harvesting mode
- Up to 10 mA source to power external circuit (Depending on harvested power from RF)
- Compatible with NFC Forum Tag Type 2
- +/-2% 1.8432MHz on-chip factory-trimmed oscillator

Interface and Peripheral

- RF interface based on ISO14443A - 106 kbps
- UART interface speed from 9600 to 115200 bps
- UART interface with hand-shaking option
- 8 programmable GPIOs
- Activity indicator pins
  - RF detect
  - RF Busy
  - Reserve Power Ready

Memory

- 228 bytes EEPROM accessible from RF and UART
- 196 bytes user memory
- EEPROM organization enabling NDEF format
- EEPROM erase/write cycle up to 100,000 times
- EEPROM memory retention up to 10 years at 70°C
- 2 x 64-byte deep FIFO for UART data transfer – TX/RX

Operating Conditions

- Operating temperature from -40 to 85°C

Package

- QFN3x3 - 16-Pin Package with Heat sink

Applications

- Firmware upgrade via NFC
- NFC bridge for embedded systems
- Metering/vending machines
- Smart interactive poster
- Smart home appliances
- Customized/proprietary system RFID

Revision History

Revision	Date	Description	Change/Updated/Comment
1.0	03 October 2013	1st Release	1st Official Release
1.1	25 October 2013	Correction/ Addition	Correct RxUR command format in section 8.2.2, table 8-10, figure 8-18 and figure 8-19.
1.2	28 November 2013	Correction/ Addition	**Throughout Document:** • Redefine register 0x01.1 name from **PW_RDY** to **RSPW_RDY** to make it more pertinent. (**RSPW_RDY** stand for "Reserve Power Ready"). • Redefine signal from *PWR_RDY#* to *RSPW_RDY#.* • Redefine flag named **"PWR_LOW"** to be **"RSPW_LOW"**. **Ordering Information:** Update ordering part number to be "P002HS4310MQFN3-02" (Rev C) **Section 3.2:** Update electrical parameter based on silicon. - **Table** 3-3**:** Add RF limiter level if **RFLM_LV** = 1**.** - **Table 6-2:** Add a functional register 0x0E.0 name **2B_FLAG** for response compatible with some NFC chips in market. Its default value is "1". **Section 6.2.2:** Change content of register 0x01.1 : – from "RF input power…" to "Reserve power from RF …" – from "Power supplying level can be set …" to "Reserve power qualifying level can be …" **Section 6.2.13:** • Correct term **PWR_LEV** to be **PW_LV**, indicating reserve power qualifying level. • Correct power level in register **PW_LV[1: 0]** • Explain more detail of register **PWCHK_EN** **Section 6.2.14:** Add a functional register 0x0E.0 name **2B_FLAG** **Section 7:** • Add effect of **2_Flag** in figure throughout this section. • Add information about mode of operation of SIC4310. **Section 7.1.2:** • Correct content of register **SCAP_RDY** : – "The maximum voltage on pin **SCAP** can be selected either …" – "The voltage on pin **SCAP** can be monitored if it is over 4.5 …" • Add method in using power from super capacitor "To use power…" **Section 7.2:** Rename term "surplus" to be "reserve" to make it more pertinent. **Table 7-2**: Add register **RFLM_LV**. **Section 8.2.1:** Add information about **"UL_FF_OVF"** in this command. **Section 8.2.2:** Correct content related to behaviour of uplink transmission **From** "uplink FIFO stops receive data from UART end device and only 64-bytes remaining data in uplink FIFO is transmitted …" **To** "uplink FIFO stops receive data from UART end device and stops transmission in uplink frame" **Section 8.3.1:** Add information "If the reading address is out of range … "

			Figure 8-20: Correct behaviour of uplink transmission in case of **UL_FF_OVF** **Section 0:** Add addition information about register **2B_FLAG** Table 8-18 and **Table 8-19:** Correct term **"DL_FIFO_OVF", "UL_FIFO_EMP", "UL_FIFO_OVF"** to **"DL_FF_OVF", "UL_FF_EMP", "UL_FF_OVF"** respectively and correct term **"PWR_LOW"** to **"RSPW_LOW"** **Table 8-19:** Correct Description "Insufficient input RF power" to "Insufficient power to source load"
1.3	03 December 2013	Correction/ Addition	**Correct** response packet (B_NAK and B_ACK) throughout section 8.2 and 8.3 for default value of **2B_FLAG** of '1'. **Update** information in table through out
1.4	27 January 2014	Correction/ Addition	**Update** information to new form (version 2.0)
1.5	13 August 2014	Correction/ Addition	**Add** caution about pin **UMAS** usage in Table 7-1. *** In case of power harvesting, it is recommend to not tide pin **UMAS** to pin **XVDD** as a logic '1' during initialization to prevent system stuck in reset state. **Correct** response time in Figure 8-1 to Figure 8-27 from (N*128+52)/fc **to** (N*128-204)/fc**.**
1.6	19 October 2017	Correction/ Addition	**Update Ordering information** Update product code from P002HS4310MQFN3-02 to P002HS4310MQFN3-03 Remove "NDEF format" description from description
1.7	18 January 2022	Correction/ Addition	**Update Ordering information** Update product code from P002HS4310MQFN3-03 to P10CVQK4P20UT1001E3 Update description form
1.8	13 December 2022	Template	Update document template
1.9	21 December 2023	Correction/ Addition	**Update Ordering information** Update product code from P002HS4310MQFN3-03 to P10CVQK4P20UT1001T6 Update wording and page layout

Ordering Information

Part No.	Description	Package
P10CVQK4P20UT1001T6	SIC4310-01, NFC-Forum Type-2 Tag IC with 228-bytes, UART Interface and 8 GPIOs QFN 0.85 mm, TnR, IC	QFN3x3-16 Pin

Content

List of Figures

List of Tables

0. Notation

0.1 Styles and Fonts for key words

This part defines styles and fonts used for the key words throughout this document. The key words are names of signal, register, pin, state of operation and command. The styles, fonts, and their indications are shown in Table 0-1.

Table 0-1: Styles and Fonts for key words

Symbol	Indication
Signal	Signal name
Register	Register name or Bit name
pin RX	Pin name
"State of Operation"	State of operation
Command	Command name for RF interface and UART interface
"Flag"	Flag name in B_ACK or B_NAK response

- To refer to a register address and a value in a register, a hexadecimal number proceeding with letter "0x" is used, for example 0x0A.
- To refer to a bit located in a register address, a symbol "." following by a number reflecting the bit location starting from 0 to 7 is used. For example, 0x0A.0 refers to bit 0, least significant bit, in the register 0x0A.
- To refer to a set of consecutive bits located in a register address, a format ".[msb:lsb]" is used after a register value. For example, a value of 0x0A.[3:0] refers to bit 3, 2, 1, and 0 in the register 0x0A.
- To refer to a binary value in some registers, the letter "b" is placed at the end of the binary number, for example "1010b".
- To refer to logic level, the number in single quote '1' and '0' are used to refer to binary logic level.

0.2 Abbreviation

Table 0-2: Abbreviation

Abbreviation	Term
fc	Carrier frequency
SOF	Start of Frame
EOF	End of Frame
FIFO	First-In, First-Out Memory
CRC	Cyclic redundancy check
EEPROM	Electrically Erasable Programmable Read-Only Memory
UID	Unique Identifier
B_ACK	Byte Acknowledge
B_NAK	Byte Negative Acknowledge
ACK	Acknowledge
NAK	Negative Acknowledge
GPIO	General Purpose Input/Output
LDO	Low-Drop-Out Regulator
AFE	Analog-Front-End
QFN	Quad-flat no-leads package
UL_FIFO	Uplink First In, First Out Memory
DL_FIFO	Downlink First In, First Out Memory
UID	Unique ID
OTP	One-time program
OSC	Oscillator
FDT	Frame Delay time
UART	Universal Asynchronous Receiver/Transmitter

1. Functional Overview

The SIC4310 is a dual-port, 228-byte, NFC-tag IC with an UART interface. The EEPROM memory can be accessed via either NFC/RFID reader devices or UART and is organized to be compliant with NFC Forum Tag Type 2. Apart from the memory device, the SIC4310 is intended to be a protocol converter that avails direct transparent data transfer from NFC device to UART or vice versa as well. The UART interface facilitates data communication with various kinds of devices for multipurpose applications such as MCUs or sensor modules.

1.1 Block diagram

Figure 1-1 depicts a conceptual block diagram of the SIC4310. The SIC4310 mainly consists of seven parts as listed below.

- RF Analog Front End (RF-AFE)
- On-chip 3.3V LDO Regulator
- Digital Controller
- EEPROM
- GPIOs
- On-chip Oscillator
- FIFO

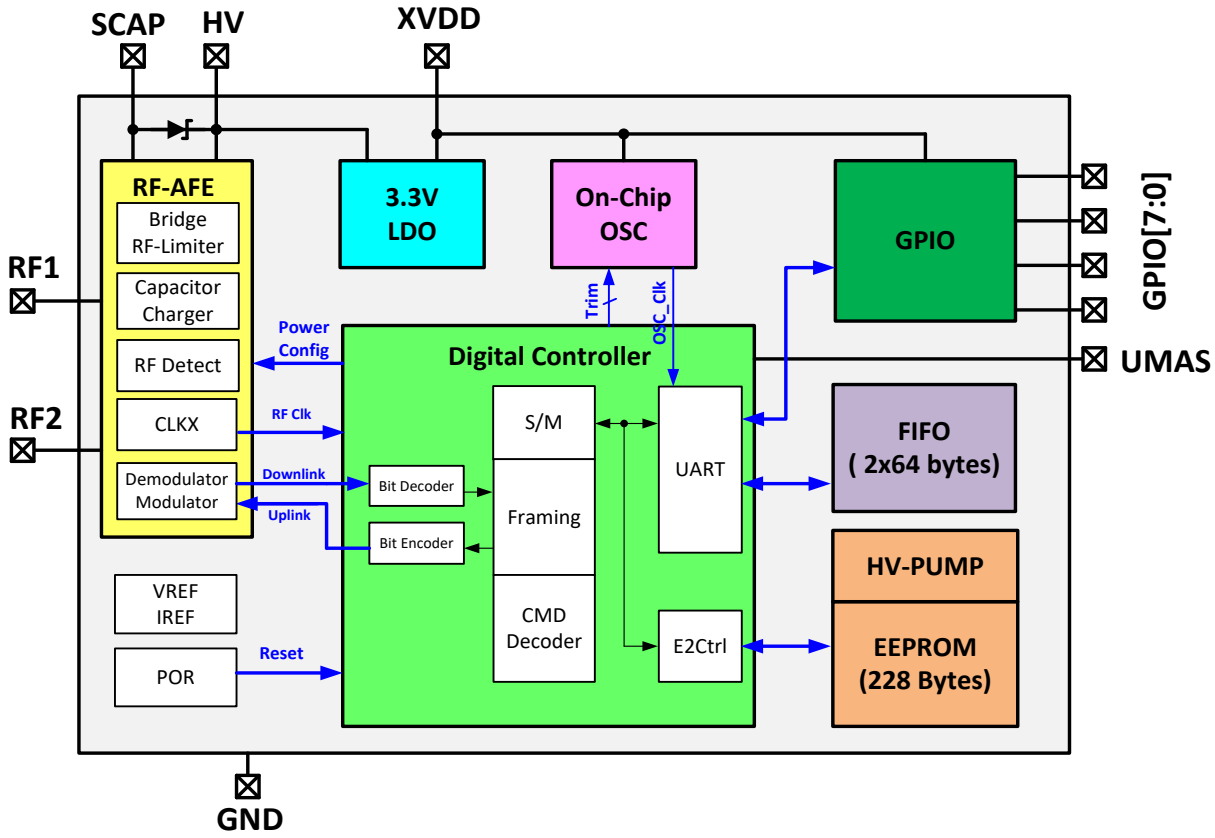


Figure 1-1: Functional block diagram

1.1.1 RF Analog-Front-End (RF-AFE)

The RF Analog-Front-End (RF-AFE), where RF1 and RF2 terminals connect to an external coil, harvests RF power to supply the internal circuit when the power source is RF. Also, the RF-AFE provides facilities for RF communication such as Modulator/Demodulator for uplink and downlink data communication, a clock extractor for the system clock and data synchronization, RF field detector for detecting the presence of the RF field. The RF-AFE is designed to collect excess energy from the RF field to store in the capacitor, optionally connected to the pin **SCAP**.

1.1.2 On-chip 3.3V LDO Regulator

Provided that the input power is high enough, the on-chip 3.3V Low-Drop-Out (LDO) regulator provides a stable power supply voltage for external devices connected to pin XVDD, on-chip oscillator and GPIO (General Purpose Input/Output) in the power harvesting mode. The LDO regulator can be enabled or disabled via a control register. Note that if the SIC4310 is operated by power from external source, the On-chip 3.3V LDO Regulator should be turned off to prevent loading effect to RF.

1.1.3 Digital controller

The digital controller manipulates data transactions between the external interfaces, (RF and UART), and internal memory. Digital controller handles operations as follows.

- Decoding incoming RF downlink commands and encoding RF uplink data
- Receiving and transmitting UART packets.
- Reading and programing data from/to EEPROM.
- Setting and resetting values to GPIOs.

Moreover, digital controller contains control registers to define behaviors of all functional parts such as UART, RF-AFE, LDO, GPIOs, etc.

1.1.4 EEPROM

EEPROM consists of EEPROM memory blocks and high-volt generator. The EEPROM memory is used to store UID, user data, and memory lock control to serve NFC applications. The EEPROM also contains a portion of register-reloading values for predefining the control register after a power-on-reset. The on-chip high-volt charge pump generates high voltage to program and erase the EEPROM.

1.1.5 GPIOs (General Purpose Inputs/Outputs)

The GPIOs (General Purpose Input/Output) are I/O control units for configuring the pins function. Each pin can be configured to be general I/O or special functional like UART interface or RF signal indicators. Also, pin direction can be set via the control registers.

1.1.6 On-chip oscillator

The on-chip oscillator generates a stable 1.8432-MHz clock source trimmed from the factory. The oscillator is designed to be insensitive to the power supply and temperature for reliability in UART communication. Also, the clock from the oscillator is used to operate the chip when the RF field is absent.

1.1.7 FIFO

Two 64-byte-depth FIFOs provides for the data transfer between the NFC/RFID and the UART. One (DL FIFO) is for downlink and another (UL FIFO) is for uplink communication. In the downlink, decoded data from the RF is buffered in the DL FIFO before sending out to the UART TX. In the uplink, the external host shall store data in the buffer prior to the NFC device requesting data. When the SIC4310 performs as a memory device within an embedded system, the FIFO also serves as a buffer for data-accessing command from the UART-connected device.

1.2 Typical operating system

The SIC4310 can be configured in various arrangements as illustrated in Figure 1-2 to Figure 1-5. A loop antenna is directly connected from pin **RF1** and **RF2** for NFC communication, whereas the UART pins can connect to

- End host device such as microcontroller, end slave device such as sensor module,
- Bridge device such as RS232 converter,
- Basic indication such as LED in GPIO mode.

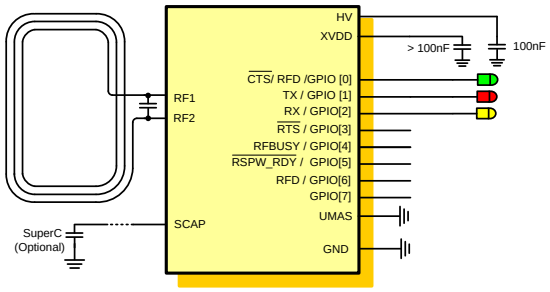


Figure 1-2: Basic configuration with LED indicator (Power harvesting)

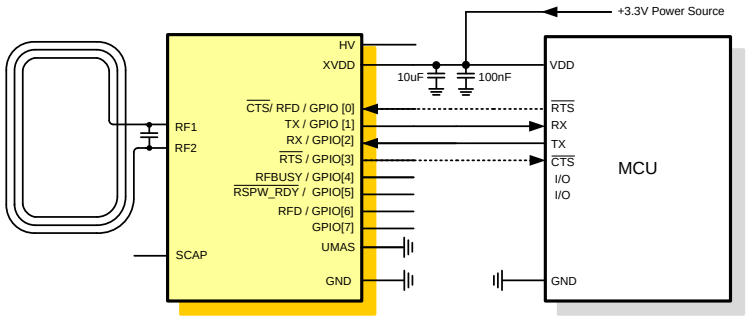


Figure 1-3: Basic UART connection to MCU (handshake is optional)

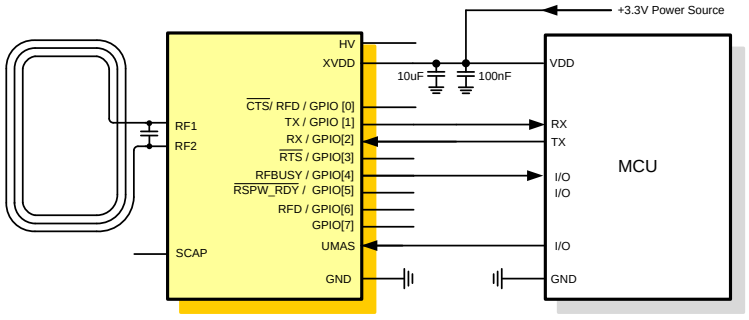


Figure 1-4: UART connection to MCU with EEPROM accessibility (UMAS connection)

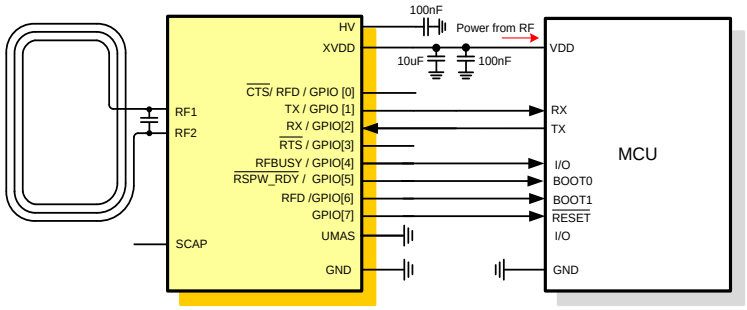


Figure 1-5: RF-powered configuration for firmware upgrade (Power harvesting)

2. Pin configuration

2.1 Pin configuration

2.1.1 QFN 3x3 – 16 Pin

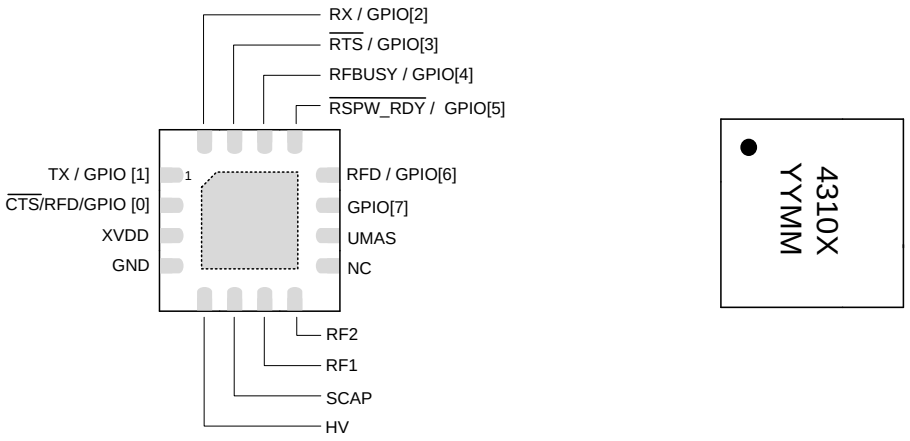


Figure 2-1: QFN 3x3-16 Pin arrangement and Marking (Top View)

Table 2-1: QFN 3x3 - 16 Pin description

Pin	Symbol	Type	Description
1	**TX /GPIO[1]**	I/O	UART-TX, GPIO port[1]
2	**CTS#/RFD/GPIO[0]**	I/O	Clear_to_Sent (Active low) , RF detect, GPIO port[0]
3	**XVDD**	Power	VDD supply or Regulated VDD supply output from RF power harvesting
4	**GND**	Power	Ground
5	**HV**	Power	Unregulated power supply
6	**SCAP**	Power	Super capacitor connection
7	**RF1**	Power	RF-Coil Connection Pin1
8	**RF2**	Power	RF-Coil Connection Pin2
9	**NC**	-	Not connected
10	**UMAS**	I	UART as UMAS Side to access internal EEPROM
11	**GPIO[7]**	I/O	GPIO port[7]
12	**RFD/GPIO[6]**	I/O	RF detect, GPIO port[6]
13	**RSPW_RDY#/GPIO[5]**	I/O	Reserve Power Ready (Active low), GPIO port[5]
14	**RFBUSY/GPIO[4]**	I/O	RF Busy or GPIO port[4]
15	**RTS#/ GPIO[3]**	I/O	Request_to_Sent (Active low), RF detect, GPIO port[3]
16	**RX /GPIO[2]**	I/O	UART-RX, GPIO port[2]

Note the “#” indicates active low signal.

Note the “X” in marking represents a silicon revision.

Note the “YYMM” in marking represents a production lot.

3. Specifications

3.1 Absolute maximum rating

Stresses above those listed under absolute maximum ratings may cause permanent damage to the device. Exposure to the absolute maximum rating conditions for an extended period of time may affect the device reliability.

Only one absolute maximum rating can be applied at a time.

Table 3-1: Absolute maximum rating

Parameter	Rating
Supply voltage (XVDD)	-0.3 V to 3.6 V
Input voltage	-0.3 V to XVDD + 0.3 V
Output voltage	-0.3 V to XVDD + 0.3 V
RF input current	30mA
Operating temperature range	-40 C to +85 C
Storage temperature range	-65 C to +150 C
Junction temperature	125 C
Thermal impedance (θJA)(1) – QFN 3x3 – 16 pins	TBD

Note θ_{JA} is determined by 2s2p 76.2x114.3-mm PCB following JEDEC51-5, -7

3.2 Electrical characteristic

Table 3-2: Operating condition

Parameter	Description	Min	Typ	Max	Unit	Conditions
XVDD	Supply voltage	2.7	3.3	3.6	V	3.3-V version
ESD	Electrostatic discharge tolerance	1.5			kV	HBM model

Table 3-3: RF front end characteristic

Parameter	Description	Min	Typ	Max	Unit	Conditions
f-op	RF operating frequency		13.56		MHz	
Vcoil-pp	POR threshold		6		Vpk-pk	
	EEPROM programming		6		Vpk-pk	
VRFlimit	RF Limiter Level @ 10mA input		10.5		Vpk-pk	**RFLM_LV** = 0
			12.5		Vpk-pk	**RFLM_LV** = 1
Vmod	Modulation Level @ 1mA input		2		Vpk-pk	
	Modulation Level @ 10mA input		5		Vpk-pk	
Cr	On-chip resonance capacitor		30.3		pF	
Vscap_max	Maximum voltage on pin **SCAP**		4.8		V	**RFLM_LV** = 0
			6.0		V	**RFLM_LV** = 1
TRF_off	Minimum period for RF field-off to ensure reset	1			mS	

Table 3-4: Power consumption

Parameter	Description	Min	Typ	Max	Unit	Conditions
Idd-RF1,RF2	Power supply current into RF1, RF2		100		uA pk	Standalone RFID mode, UART disabled, EEPROM read mode, Vrf1,2= 6 Vpk-pk
			120		uA pk	Standalone RFID mode, UART disabled, EEPROM programing mode, Vrf1,2= 6 Vpk-pk
Ixvdd	Idle current into XVDD		80		uA	Power source from pin **XVDD**, No RF field, Register **OSCEN** = 0, LDO is tuned off, Pin **UMAS** = 0
			176		uA	Power source from pin **XVDD**, No RF field, Register **OSCEN** = 0, LDO is turned off, Pin **UMAS** = 1

Table 3-5: Pin characteristics

Parameter	Description	Min	Typ	Max	Unit	Conditions
C_I/O	GPIO pin capacitance		10		pF	
VINL	Digital logic input Low voltage			0.8	V	XVDD = 3.3 V
VINH	Digital logic input High voltage	2.4			V	XVDD = 3.3 V
VOL	Digital logic output low voltage		0.15	0.20	V	XVDD = 3.3 V, IL = 1 mA
			0.50	0.60	V	XVDD = 3.3 V, IL = 4 mA
VOH	Digital logic output high voltage	3.10	3.15		V	XVDD = 3.3 V, IL = 1 mA
		2.65	2.75		V	XVDD = 3.3 V, IL = 4 mA
Tr	Rise time		4	6	nS	XVDD = 3.3 V , CL = 10pF
Tf	Fall time		4	6	nS	XVDD = 3.3 V , CL = 10pF
Iinlogic1	Logic 1 input current			1	uA	VINH = XVDD
Iinlogic0	Logic 0 input current			1	uA	VINL = 0
Ioutlogic1	Logic 1 output source current			6	mA	XVDD = 3.3 V
Ioutlogic0	Logic 0 output sink current			6	mA	XVDD = 3.3 V
Rpullup	Pull up resistance at GPIO		115		kohm	PU[n] = 1

Table 3-6: Operation timing

Parameter	Description	Min	Typ	Max	Unit	Conditions
Tpowerup	Startup time from power up		2	5	mS	After burst RF field until chip ready to receive command.
TPwrRDY	Power ready delay time after burst		1.5		mS	After burst RF field until PowerRDY = 1 Input power from RF is > level set by PW_LEV [1:0]
TRFRDY	RF ready delay time after burst		1.5		mS	After burst RF field until PowerRDY = 1 Input power from RF is > level set by PW_LEV [1:0]

Table 3-7: EEPROM

Parameter	Description	Min	Typ	Max	Unit	Conditions
TEEprog	EEPROM programming time		3.85		mS	Programming 1 block
XVddMinProg	Minimum XVDD voltage for programming voltage	2.4	3		V	
RFMINProg	Minimum RF voltage for programming voltage		6		Vpk-pk	

Table 3-8: LDO regulator

Parameter	Description	Min	Typ	Max	Unit	Conditions
VREGIN	Regulator input voltage	4.0	5	7	V	
VXVDDOUT	Regulator output voltage	3.2	3.3	3.4	V	Iload = 0 mA
IREGOUT	Output regulator current			10	mA	
ΔVoutLoadReg	Load regulation (ΔVout)		2.9		mV	Iout = 1 mA, Vin = 5V
			6.6		mV	Iout = 5 mA, Vin = 5V
IREGBias	Regulator bias current		20		uA	5 V < VREGIN < 7 V
XVDDcap	XVDD decoupling capacitor	100			nF	Regulator stable
VXVDDdrop	XVDD drop level		2.4		V	**LDO_D_LV** = 0
			2.7		V	**LDO_D_LV** = 1
VUARTop	Operating voltage for UART operating	2.1			V	

Table 3-9: On-chip oscillator

Parameter	Description	Min	Typ	Max	Unit	Conditions
Fosc	Nominal oscillator frequency		1.8432		MHz	XVDD = 3.3V
ΔFosc	Frequency deviation	-2		2	%	XVDD = 3.3V Temp = -40C to 85C
Iosc	Current consumption when oscillator is on		15		uA	
Tosc_settle	Oscillator settling time		100	120	uS	
Fsen_XVDD	Power supply sensitivity to frequency		0.77		%/V	Temp = 25 C

3.3 Peripheral specification

Table 3-10: Peripheral specification

Block	Properties	Min	Typ	Max	Unit
UART	UART_Baud Rate	457	115,200		kbps
FIFO	Downlink FIFO size		64		Bytes
	Uplink FIFO size		64		Bytes
EEPROM	Total size		228		Bytes
	Write endurance	100,000			Times
	Retention	10			Years

4. Communication

The SIC4310 is a dual-interface transponder IC which can be accessed by an NFC/RFID device and UART interface provides facilities to be connected to various UART devices. In addition, the SIC4310 enables transparent data transmission between NFC/RFID reader device and UART end device. The RF interface and UART interface behaviour is delineated in this section.

4.1 RF interface

The RF interface of SIC4310 is based on the standard for contactless smart cards ISO 14443A-2. PCD and PICC according to ISO standard are referred respectively as NFC/RFID device and SIC4310/tag/transponder/chip throughout this document.

The SIC4310 activates itself by energizing RF field generated by its companion NFC/RF device. When the transponder is powered up and internal supply voltage is higher than the POR threshold, the chip initiates itself and waits silently for an operational command and then starts transmitting in uplink as a response.

4.1.1 Downlink

In downlink, the RF device starts sending a command to the transponder by interrupting the field. The downlink communication takes place using 100% ASK modulation with the Miller coding. The transmission bitrate is 106 kbps (fc/128). Figure 4-1 depicts an example of downlink telegram.

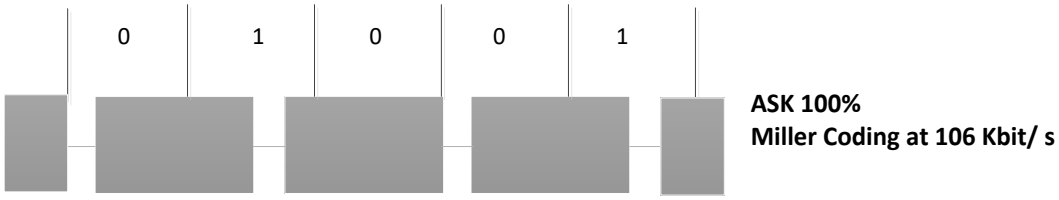


Figure 4-1: Example of downlink telegram

4.1.1.1 Downlink bit pattern

Downlink bit pattern is based on the ISO 14443 type-A protocol as defined in Table 4-1 and Table 4-2.

Table 4-1: Sequences for the downlink bit-pattern

Sequence X	After a time of 64/fc a “pause” shall occur
Sequence Y	For the full bit duration (128/fc) no modulation shall occur
Sequence Z	At the beginning of the bit duration a “pause” shall occur

Table 4-2: Information to code with the downlink sequences

Logic '1'	**Sequence X**
Logic '0'	**Sequence Y** with the following two exceptions: If there are two or more contiguous '0's, sequence Z shall be used from the second '0' on If the first bit after a "start of frame" is '0', sequence Z shall be used to represent this and any '0's which follow directly thereafter
Start of communication	**Sequence Z**
End of communication	Logic '0' followed by **Sequence Y**
No information	At least two **Sequence Y**

4.1.2 Uplink

After the SIC4310 executes a command from NFC device and the SIC4310 starts transmission in uplink as a response, the transponder communicates with NFC/RFID device by load modulation through inductive coupling field. Uplink bit pattern is defined based on ISO14443 type A. The uplink bit definition is described in Table 4-3 and Table 4-4. The uplink data is encoded in the Manchester format with subcarrier frequency of 847 KHz (fc/16). One-bit duration is 8 periods of the subcarrier, equivalent to bitrate of 106 kbps (fc/128).

Figure 4-2 depicts an example of data encoding in uplink telegram.

Table 4-3: Sequences for the uplink bit pattern

Sequence D	The carrier shall be modulated with the subcarrier for the first half (50%) of the bit duration
Sequence E	The carrier shall be modulated with the subcarrier for the second half (50%) of the bit duration
Sequence F	The carrier is not modulated with the subcarrier for one bit duration

Table 4-4: Uplink data coding

Logical '1'	**Sequence D**
Logical '0'	**Sequence E**
Start of communication	**Sequence D**
End of communication	**Sequence F**
No information	No subcarrier

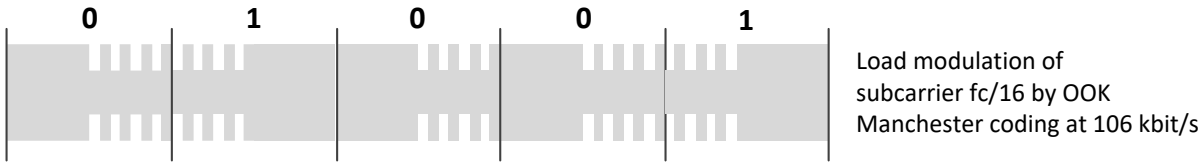


Figure 4-2: Example of uplink telegram

4.1.3 Frame pattern

The frame pattern for RF communication is based on the ISO14443 type-A protocol. There are three types of frame pattern illustrated in Figure 4-3. The frame types are as follows: short frame, standard frame, and bit-oriented anti-collision frames. The purposes of each frame type are summarized in

Table 4-5. This frame format is applied for both downlink and uplink. Each frame begins with a start bit and ends with an end bit. Transmission starts with the LSB of the lowest byte of transmission data. Each byte is transmitted with an odd parity. Note that, for transferring data from RF to UART, the maximum frame size for UART data transmission is 64 bytes excluding the CRC. Hence, the maximum bit length (Start of frame + Payload + End of frame) of downlink frame is 597 bits (1+66x9+2) and 596 bits (1+66x9+1) in uplink. For more information, please refer to ISO14443-3.

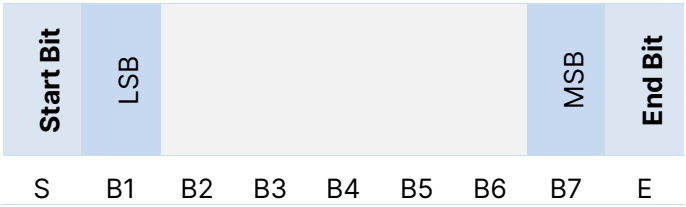
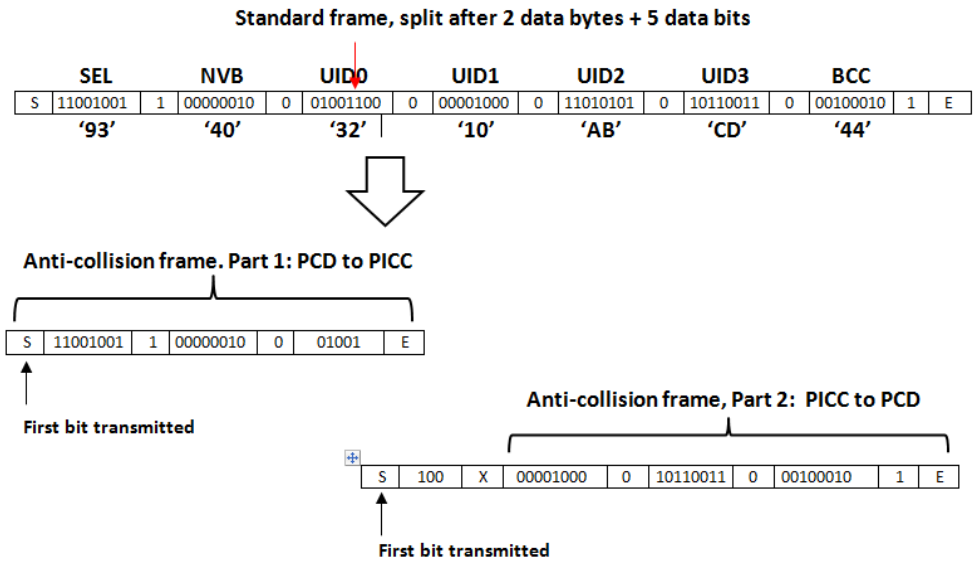
Short frame	 Short frame contains 7 data bits and the LSB is transmitted first.
Standard frame	 The frame contains n* (8 data bits+ odd parity bit) bits where n≥1.
Bit-oriented anti-collision frame	 In bit-oriented anti-collision frame, standard frames are split into two parts for downlink and uplink. Depending on the splitting position within a byte two cases arise FULL BYTE and SPLIT BYTE

Figure 4-3: Frame format for RF communication

Table 4-5: Information to code with the uplink sequences

Frame type	Purpose	Command example
Short frame	Initiating	ISO14443A : ***REQA, WUPA***
Standard frame	Transmitting regular command and data exchange between the transponder and NFC device.	ISO14443A : ***SEL, HLTA, READE2, WRITEE2, Compatible WriteE2*** RFID_UART : ***TxRU, RxUR, TRxRU, Clear_Flag*** RF-CONFIG : ***ReadReg, WriteReg***
Bit-oriented anti-collision frame	Transmitting and receiving data during anti-collision loops.	ISO14443A : ***ANTI_COLLISION***

4.1.4 Timing

The commands and response timing of SIC4310 are according to the standard of frame delay time of the ISO 14443A. Based on the ISO14443A, there is frame guard time between downlink and uplink and vice versa. Downlink frame delay time is the guard time between end of the last pause transmitted by the NFC/RFID device and the first modulation edge of the start bit transmitted by the transponder. Depicted in Figure 4-4, the downlink frame delay time is (n*128+84)/fc or (n*128+20)/fc depending on end bit value ('0' or '1' respectively). The n value must be more than 9. The transponder response starts in a defined time slot. On the other hand, uplink frame delay is the guard time between the last modulation transmitted by the transponder and the first pause transmitted by the NFC/RFID, which is approximately at least 1172/fc or 87 µs. The uplink frame delay is shown in Figure 4-5.

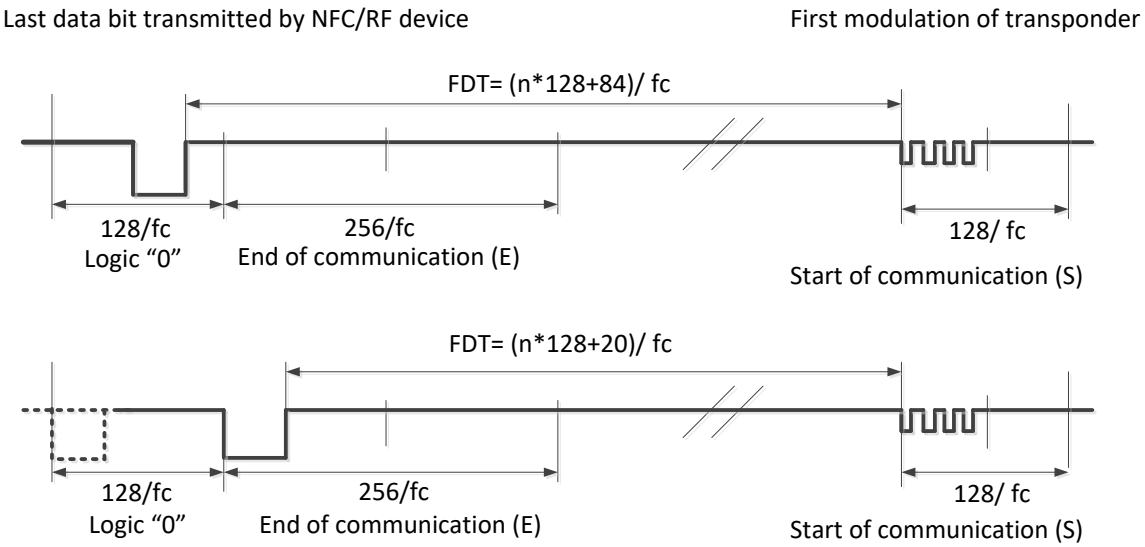


Figure 4-4: Downlink frame delay time

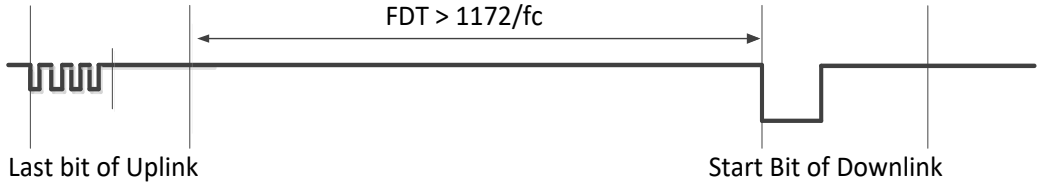


Figure 4-5: Uplink frame delay time

4.1.5 State of operation

When the SIC4310 gets an operational command from NFC/RFID device, the digital controller processes incoming commands and operates based on a current state. Figure 4-6 depicts the transponder's state diagram based on the ISO 14443-3 type A.

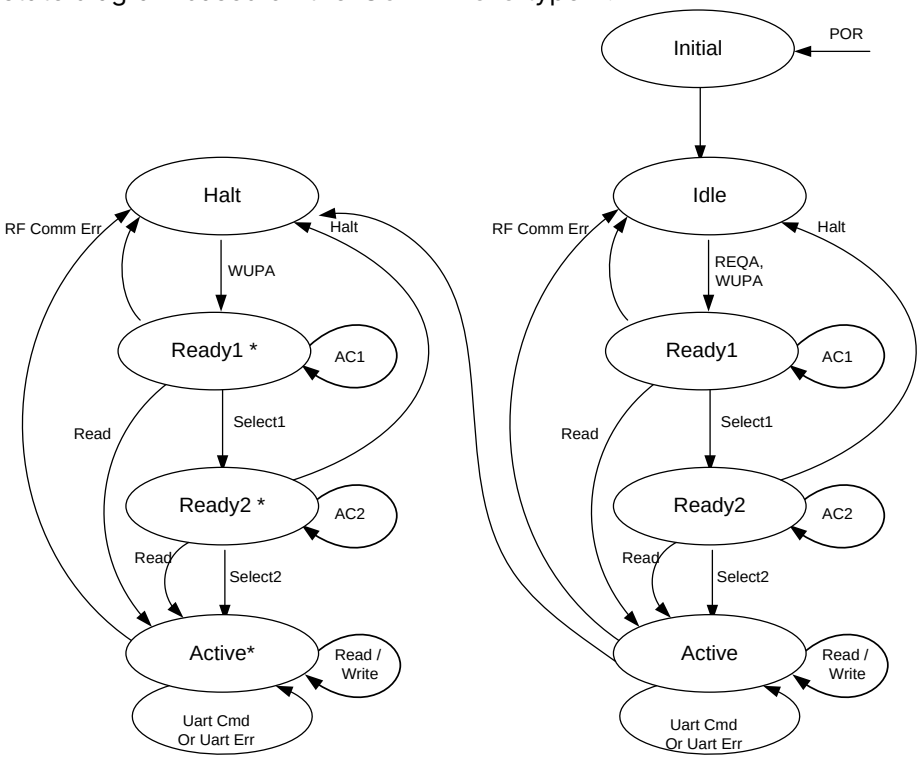


Figure 4-6: State of operation (pin **UMAS** = 0)

Initial state: After POR, state of the SIC4310 enters the ***"Initial"*** state to initialize itself. In this state, the digital controller loads a pre-programed reloading value from EEPROM to initialize a register before entering ***"Idle".*** Entering this state can also occur when the pin **UMAS** is changed from '0' to '1' or '1' to '0', or the RF field is absent. Note that the RF field is treated to be absence if the field disappears more than 500 us.

Halt & Idle state: After initialization, the SIC4310's state switches to ***"Idle"*** waiting for the command ***WUPA*** or ***REQA***. These commands change the state to ***"Ready1".*** Any other command obtained in this state is considered as an error and the state persists in the same state. There is an equivalent state ***"Halt",*** entering from the command ***Halt.*** Only the command ***WUPA*** can make the SIC4310 leave this state to ***"Ready1*".***

Ready1 & Ready1*: In the state ***"Ready1",*** anti-collision level-1 method is applied. The digital controller expects a matched ***Selection1*** or ***ANTI-COLLISION1***. For the ***ANTI-COLLISION1***, the rest of UID is responded. For the ***Select1***, when a cascaded level1 UID is matched, the digital controller responds SAK and transits to ***"Ready2".*** Except ***ReadE2,*** any other command obtained in this state is considered as an error and the digital controller returns to ***"Idle"*** or ***"Halt".***

Ready2 & Ready2*: In the state ***"Ready2",*** anti-collision level-2 method is applied. The digital controller decoder expects a matched ***Selection2*** or ***ANTI-COLLISION2.*** For the ***ANTI-COLLISION2***, the rest of UID is responded. For the ***Select2,*** when a cascaded level-2 UID is matched, the digital controller responds SAK and transits to the state ***"Active".*** Except ***ReadE2,*** any other command obtained in this state is considered as an error and digital controller returns to ***"Idle"*** or ***"Halt".***

Active & Active*: In the active state, the SIC4310 can perform RFID-memory access, UART interface and register accessing commands. The RFID-memory access commands are ***ReadE2,***

WrireE2, Compatible WriteE2. The UART interface commands are ***TxRU, RxUR, TRxRU*** and ***Clear_Flag.*** The register accessing commands are ***ReadReg, WriteReg.***

The Digital state can exit ***"Active"*** and switches to the state ***"Halt"*** by the command ***Halt.*** If RF communication error occurs during transmission in this state, the digital controller returns to ***"Idle"*** or ***"Halt".*** In case of RF-access memory command error or framing error, transponder replies a 4-bit NAK. In case of UART interface command error, transponder replies an 8-bit B_NAK and remains in ***"Active"*** state. Note that, RF communication is enabled only when the pin **UMAS** is set to '0'. When pin **UMAS** is set to '1', the digital state is switched to the UART memory mode. EEPROM can only be accessed from an external UART device such as MCU. Note that there is no response to an NFC/RFID reader device, when pin **UMAS** is set in this state to '1'.

4.2 UART Interface

The SIC4310 can directly pass data from RF to UART, or vice versa, without wasting time in programming content into EEPROM and then transferring. RF-to-UART communication takes place when the state of digital controller is in the ***"Active"/"Active*"*** states. Valid UART-TX data can be transmitted to an end device right after a valid downlink command packet is received. UART-connected device can store data into the SIC4310's uplink FIFO when the digital controller enters the ***"Ready"/"Ready*"*** or ***"Active"/"Active*"***states.

UART byte packet format is depicted in Figure 4-7.

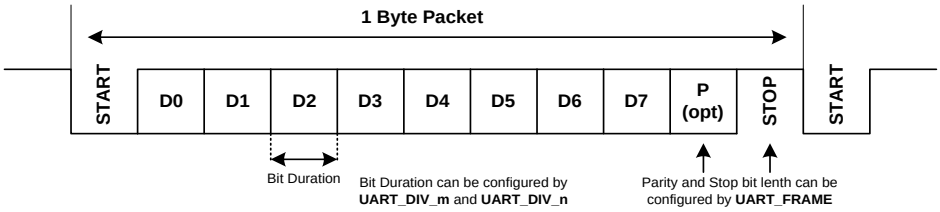


Figure 4-7: UART byte packet

For direct data transfer from NFC/RFID device to UART, only payload data in RFID packet is transmitted and received through UART. The command for RFID-UART interface is described in Section 8.2.

Internal EEPROM can also be accessed from UART-connected devices by rising the pin **UMAS** to '1', although the RF field is absent. To access EEPROM, UART-connected devices send a command to the pin **RX** and receive response data from the pin **TX.** The command format detail is shown in Section 8.4. When the pin **UMAS** is set to '1', the on-chip oscillator is automatically turned on. The EEPROM content can always be accessed from UART, regardless to the value of **OSC_EN.**

It is important to note that an NFC/RFID device cannot communicate with SIC4310 when a logic on the pin **UMAS** is set to '1'. In other word, there is no response to any downlink commands on the RF side in this state. After **UMAS** goes back to the logic '0', the RF state restarts from ***"Initial".***

Toggling the pin **UMAS** during command operation either from RF or EEPROM, accessing from the UART side, can make current operation corrupted. Note that the current operation is not suddenly broken, and the digital controller waits until the current operation is finished before switches to a receive command from the RF side or UART side. In accessing EEPROM from UART, a UART-connected device shall monitor the activity from *RF_Detect* (**GPIO[0], GPIO[6]**) and *RFBusy* (**GPIO[4]**) before raising the pin **UMAS** to '1'.

5. EEPROM Organization

The SIC4310 contains a 228-byte non-volatile EEPROM memory, conforming to the NFC Tag Type 2 arrangement as shown in Figure 5-1. The memory is incorporated with 56 pages of 4 bytes each. The EEPROM content can be accessed from either the RF side or UART side. In Figure 5-1, the section of memory is the NFC static memory area, and the rest areas are NFC dynamic memory areas. For control bits, **UID, Static Lock Byte, OTP** are stored in the NFC static memory area while **Dynamic Lock Byte** and **Register Reload Value** are stored in the NFC dynamic memory area. Usable user memory for application is 192 bytes (Page 3 to Page 51).

Page (Dec)	Page (Hex)	Byte 0	Byte 1	Byte 2	Byte 3	Memory Type	Description	Note
0	00	UID0	UID1	UID2	BCC0	R/O	UID / Lock	64-byte NFC Static Memory
1	01	UID3	UID4	UID5	UID6	R/O		
2	02	BCC1	Internal	Lock Byte0	Lock Byte1	R/O, R/W		
3	03	OTP	OTP	OTP	OTP	R/W (OTP)	48-byte User Data	
4	04					R/W		
…	…					R/W		
15	0F					R/W		
16	10					R/W	144-byte User Data	164-byte NFC Dynamic Memory
…	…					R/W		
51	33					R/W		
52	34	Lock Byte2	Lock Byte3	Lock Byte4	RFU	R/W	Lock Byte	
53	35	RL REG 0	RL REG 1	RL REG 2	RL REG 3	R/W	16-byte Reload Register	
54	36	RL REG 4	RL REG 5	RL REG 6	RL REG 7	R/W		
55	37	RL REG 8	RL REG 9	RL REG 10	RL REG 11	R/W		
56	38	RL REG 12	RL REG 13	RL REG 14	RL REG 15	R/W		

Figure 5-1: SIC4310 EEPROM memory map

5.1 UID

UID is a factory pre-programmed, write-protected identification number that is composed of a 7-byte serial number along with its two check bytes. **UID** is stored in byte 0 of page 0 to byte 0 of page 3 of the EEPROM as depicted in Figure 5-1. When the SIC4310 receives ***the ANTI-COLLSION*** command, it responds the NFC/RFID reader device with **UID. BCC** is kept in the EEPROM during manufacturing to ensure that uplinked **UID** is stored in EEPROM correctly.

5.2 Static lock byte

Byte 2 and 3 of page 2 (0x02) of EEPROM memory contain static lock bytes named **Lock Byte0** and **Lock Byte1.** Each bit can be called a lock bit, controls programmability for its addressed page or corresponding group of lock bits itself. When a certain lock bit in **Lock Byte0** or **Lock Byte1** is set to '1', the addressed page cannot be changed. Bits of these lock bytes are one-time program (OTP). Therefore, once it is programmed to '1', such a bit is unable to clear back to '0'.

Three LSB bits of **Lock Byte0** function as lock of lock-bits of static user memory. When an individual bit in these three LSB bits is set, the corresponding page lock bit values cannot be altered everlastingly, and the addressed pages remain locked or unlocked state based on the last individual lock bit value. Note that new lock bit configuration is loaded and effective after (re)entering the ***"Idle"*** or ***"Halt"*** state.

Byte in Page 2 (Page 0x02)		**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Byte 2	Lock Byte 0	Lock Page 7	Lock Page 6	Lock Page 5	Lock Page 4	Lock Page OTP	Lock of Lock Bit 15-10	Lock of Lock Bit 9-4	Lock of OTP Lock Bit
Byte 3	Lock Byte 1	Lock Page 15	Lock Page 14	Lock Page 13	Lock Page 12	Lock Page 11	Lock Page 10	Lock Page 9	Lock Page 8

Figure 5-2: Lock configuration in static memory

5.3 OTP

Page 3 of EEPROM memory is the OTP page with four OTP bytes. All bits of these OTP bytes are set to '0' from manufacturing and can be programmed to '1' bitwise by the ***WriteE2*** and ***Compatible WriteE2*** commands. Once any OTP bit is programmed to '1', it cannot be reprogrammed such a bit back to '0' by any write command. OTP programming behaviour is shown in Figure 5-3.

Absolute byte Address	**Byte 12**	**Byte 13**	**Byte 14**	**Byte 15**
Byte in Page 3 (Page 0x03)	**Byte 0**	**Byte 1**	**Byte 2**	**Byte3**
Default value	0000 0000	0000 0000	0000 0000	0000 0000
Program with	1111 1111	0000 1100	0000 0101	0000 0000
Result in page 3	1111 1111	0000 1100	0000 0101	0000 0000
Program with	0000 0000	1111 1100	0000 0000	0000 0111
Result in page 3	1111 1111	1111 1100	0000 0101	0000 0111

Figure 5-3: OTP behavior in Page 3

5.4 User memory

Page 4 to page 51 of the EEPROM is the user memory. Initially, all blocks of user memory are programmed to '0' during manufacturing. They can be written by ***WriteE2*** or ***Compatible WriteE2*** commands and read by ***ReadE2*** command. Address of EEPROM is designed to support NFC data in the TLV format such as Lock Control, Memory control or NDEF message. NFC data such URL can be stored in the memory. Figure 5-4 shows an example of URL written in the user memory under the TLV format. For more information, please refer to the "NFC Forum Tag Type2 specification" standard.

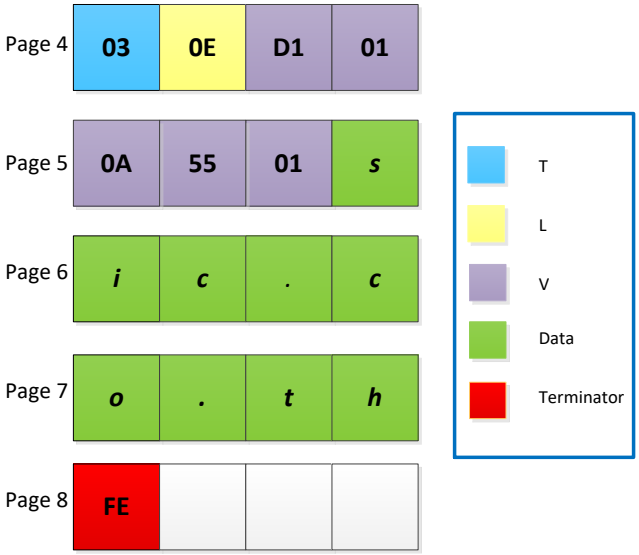


Figure 5-4: URL written in the user memory under the TLV format

5.5 Dynamic lock byte

Page 52 of the EEPROM contains **Dynamic lock** bytes. Function of each bit of **Dynamic lock** bytes is to set an associated read/write memory area to be read-only. Lock configuration of dynamic memory is shown in Figure 5-5.

Byte in Page 52 (Page 0x34)		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	Lock Byte 2	Lock Page 36-39	Lock Page 32-35	Lock Page 28-31	Lock of Lock Byte2 Bit 5-7	Lock Page 24-27	Lock Page 20-23	Lock Page 16-19	Lock of Lock Byte2 Bit 1-3
Byte 1	Lock Byte 3	RFU (OTP)	RFU (OTP)	RFU (OTP)	RFU (OTP)	Lock Page 48-51	Lock Page 44-47	Lock Page 40-43	Lock of Lock Byte3 Bit 1-3
Byte 2	Lock Byte 4	RFU (OTP)	RFU (OTP)	RFU (OTP)	Lock Reload	RFU (OTP)	RFU (OTP)	RFU (OTP)	Lock of Lock Byte3 Bit 4
Byte 3	Lock Byte 5	RFU (OTP)	RFU (OTP)	RFU (OTP)	RFU (OTP)	RFU (OTP)	RFU (OTP)	RFU (OTP)	RFU (OTP)

Figure 5-5: Lock configuration of dynamic memory

5.6 Register Reload value

Page 52 to 56 of EEPROM contains reloading values for the register page. Reloading process to the register page performs during the ***"Initial"*** state. 16-byte data in EEPROM starting from Byte0 of Page53 to Byte3 of Page56 is transferred to Address 0 to Address 15 of the register page.

6. Register

6.1 Register overview

The SIC4310 consists of two types of 16-byte addressable register pages namely Read only and Read/Write. The register pages initialize automatically from EEPROM after POR. Behaviours of register pages are described in the following Table 6-1. The overview of the register map is shown in Table 6-2. The register names are listed in the far-right column.

Table 6-1: Type of register

Type	Description
Read only	The read only register is used to display the status of the internal state machine. Writing these registers will not affect their values.
Read/ Write	The read-write register is used to configure and control behaviors of the NFC/RFID reader device IC. These registers can be written and read by the external controller.

Table 6-2: SIC4310 register map

Group	Addr	Type	Bit								Register Name
			7	6	5	4	3	2	1	0	
UART Status & Control	0x00	Status			CTS	RTS	DL_FF_EMT	DL_FF_OVF	UL_FF_EMT	UL_FF_OVF	UART_Status
	0x01	Status				SCAP_RDY	UART_RDY	XVDD_RDY	RSPW_RDY	LDO_ON	Power_Status
	0x02	RFU	RFU								RFU
	0x03	Config					TRxRU_Time				TRxRU Response Time
	0x04	Config					Stop_Len	Parity			UART_Frame
	0x05	Config			UART_DIV_m						UART_Divisor_m
	0x06	Config			UART_DIV_n						UART_Divisor_n
	0x07	Config					OSC_Tuning				OSC_Tuning
IO and Peripheral	0x08	Config	DIR7	DIR6	DIR5	DIR4	DIR3	DIR2	DIR1	DIR0	GPIO_DIR
	0x09	Config	MODE7	MODE6	MODE5	MODE4	MODE3	MODE2	MODE1	MODE 0	GPIO_Mode
	0x0A	Config	OUT7	OUT6	OUT5	OUT4	OUT3	OUT2	OUT1	OUT0	GPIO_Out
	0x0B	Status	IN7	IN6	IN5	IN4	IN3	IN2	IN1	IN0	GPIO_In
	0x0C	Config	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0	GPIO_PU
	0x0D	Config			PW_LV[1:0]			PWC_EN	LDO_EN	OSC_EN	Peripheral_ Config
	0x0E	Config						LDO_LV	RFLM_LV	2B_FLAG	Peripheral_ Adjustment
	0x0F	Config	RFU								RFU

Table 6-3: Factory preprogram register value

Group	Addr	Type	Bit								Register Name
			7	6	5	4	3	2	1	0	
UART Status	0x00	Status									UART_Status
	0x01	RFU									Power_Status
	0x02	RFU									RFU
	0x03	Config					0010b				TRxRU Response Time
	0x04	Config					0	000b			UART_Byte_Configuration
	0x05	Config			000001b						UART_Divisor_m
	0x06	Config			001000b						UART_Divisor_n
	0x07	Config					1000b				OSC_Tuning
IO	0x08	Config	0	0	0	0	0	0	1	1	GPIO_DIR
	0x09	Config	0	0	0	0	0	1	1	1	GPIO_Mode
	0x0A	Config	0	0	0	0	0	0	0	0	GPIO_Out
	0x0B	Status									GPIO_In
	0x0C	Config	0	0	0	0	0	0	0	0	GPIO_PullUp
	0x0D	Config			00b			1	1	1	Peripheral_Config
	0x0E	Config						0	0	1	Peripheral_Adjustment
	0x0F	Config									RFU

6.2 Register detail

6.2.1 Register 0x00: UART_Status

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-	-	**CTS**	**RTS**	**DL_FF_EMT**	**DL_FF_OVF**	**UL_FF_EMT**	**UL_FF_OVF**

Parameter	Type	Function and Description
CTS	R/O - Status	**CTS** pin status indicator **CTS** become '1', when, external UART device is ready to receive data. If CTS function is disable (by **GPIO_Mode** and **GPIO_DIR**), **CTS** = '1'.
RTS	R/O - Status	**RTS** pin status indicator **RTS** become '1', when UL FIFO is ready to receive data. If RTS function is disable (by **GPIO_Mode** and **GPIO_DIR**), **RTS** is always '1'.
DL_FF_EMT	R/O - Status	Downlink FIFO Empty Indicator
DL_FF_OVF	R/O - Status	Downlink FIFO Overflow Indicator flag (Can be clear by command ***Clear_Flag***)
UL_FF_EMT	R/O - Status	Uplink FIFO Empty Indicator
UL_FF_OVF	R/O - Status	Uplink FIFO Overflow Indicator flag (Can be clear by command ***Clear_Flag***)

Signal at pin **GPIO[0]** and **GPIO[3]**, functioning as **UART-CTS#** and **UART-RTS#** in handshaking mode, are an inversion of value in the flag **CTS** and **RTS** in the register page. When downlink or uplink overflow status is set, such FIFO cannot receive any further data. NFC/RFID device must send the command ***Clear_Flag*** to reset the FIFO status before performing data transmission through the FIFO.

6.2.2 Register 0x01: Power_Status

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-	-	-	**SCAP_RDY**	**UART_RDY**	**XVDD_RDY**	**RSPW_RDY**	**LDO_ON**

Parameter	Type	Function and Description
SCAP_RDY	R/O - Status	**SCAP_RDY** that equals to '1' means that voltage on pin **SCAP** is higher than 4.5V.
UART_RDY	R/O - Status	**UART_RDY** that equals to '1' means that on-chip oscillator is stable and ready to operate the UART communication.
XVDD_RDY	R/O - Status	**XVDD_RDY** that equals to '1' means that voltage on pin **XVDD** is higher than **XVDD** drop level. Register **LDO_D_LV** (0x0E.2) defines a threshold level for **XVDD** drop level.
RSPW_RDY	R/O - Status	When **RSPW_RDY** is set to '1', reserve power from RF is higher than a defined supplying level. Reserved power qualifying level can be set from the register **RFLM_LV** (0x0D.[5:4]).
LDO_ON	R/O - Status	When **LDO_ON** is set to '1', on-chip LDO regulator is successfully turned on.

6.2.3 Register 0x03: TRxRU_Response_Time

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-	-	-	-	**TRxRU_Time**			

Parameter	Type	Function and Description
TRxRU_ Time	R/W - Config	**TRxRU_Time** defines a response timeout for the ***TRxRU*** command. Transceive Response Time = 2 **TRxRU_Time** x (1 ms). Maximum value of **TRxRU_Time** is 12. Hence, the maximum response time is 4 s.

6.2.4 Register 0x04: UART_Frame

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-	-	-	-	**Stop_Len**	**Parity**		

Parameter	Type	Description	
Stop_Len	R/W - Config	**Stop_Len** configures a stop bit of UART bytes.	
		0	1 bit
		1	2 bit
Parity	R/W - Config	**Parity** configures the parity of UART bytes.	
		0XX	None
		100	'0' (Space)
		101	'1' (Mark)
		110	Even
		111	Odd

6.2.5 Register 0x05: UART_Divisor_m

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-	-	UART_DIV_m					

Parameter	Type	Description
UART_DIV_m	R/W - Config	**UART_Divisor_m** defines a divisor m for the UART clock to set a speed of the UART communication. **UART_Divisor_m** must be used together with **UART_Divisor_n** (Reg0x06). When m is set to 0, the divisor is 64.

6.2.6 Register 0x06: UART_Divisor_n

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-	-	UART_DIV_n					

Parameter	Type	Description
UART_DIV_n	R/W - Config	**UART_Divisor_n** defines a divisor n for the UART clock to set a speed of the UART communication. **UART_Divisor_n** must be used together with **UART_Divisor_m** (Reg0x05). The minimum value of n is 2. If n < 2 is set, SIC4310 cannot perform the communication.

6.2.7 Register 0x07: OSC_Tuning

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-	-	-	-	OSC_Tuning			

Parameter	Type	Description
OSC_Tuning	R/W	**OSC_Tuning** is used to fine tune a frequency of the on-chip oscillator. 0000 : frequency is set to the highest adjustable value. 1111 : frequency is set to the lowest adjustable value. Note that on-chip frequency is tuned to 1.8432MHz during manufacturing. Generally, users don't need to tune the frequency. **OSC_Tuning** optionally provides a way to tune the frequency.

6.2.8 Register 0x08: GPIO_DIR

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DIR 7	DIR 6	DIR 5	DIR 4	DIR 3	DIR 2	DIR 1	DIR 0

Parameter	Type	Description	
DIR [7:0]	R/W Config	Defines the direction of GPIOs	
		0	Input
		1	Output

6.2.9 Register 0x09: GPIO_MODE

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
MODE 7	MODE 6	MODE 5	MODE 4	MODE 3	MODE 2	MODE 1	MODE 0

Parameter	Type	Description	
MODE [7:0]	R/W Config	Defines the function of GPIOs	
		0	General Purpose I/O
		1	Special Function

6.2.10 Register 0x0A: GPIO_OUT

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Out 7	Out 6	Out 5	Out 4	Out 3	Out 2	Out 1	Out 0

Parameter	Type	Description	
OUT [7:0]	R/W Config	Defines the output values when function as general output	
		0	Logic Low
		1	Logic High

6.2.11 Register 0x0B: GPIO_IN

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IN 7	IN 6	IN 5	IN 4	IN 3	IN 2	IN 1	IN 0

Parameter	Type	Description
IN [7:0]	R/O Status	Register IN[n] report logic value present at IO[n] Pins. If direction is output, return OUT[n] value.

6.2.12 Register 0x0C: GPIO_PullUp

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PU 7	PU 6	PU 5	PU 4	PU 3	PU 2	PU 1	PU 0

Parameter	Type	Description	
PU [7:0]	R/W Config	PU [n] controls a pull up resistor when IO is set to GPIO and input	
		0	Disable Pull Up resistor
		1	Enable Pull Up resistor when set to Input

6.2.13 Register 0x0D: Peripheral_Config

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RFU		PW_LV[1:0]		RFU	PWCHK_EN	LDO_EN	OSC_EN

Parameter	Type	Description	
PW_LV[1:0]	R/W – Config	**PW_LV** defines a threshold of reserved power level received from the coil to turn on LDO for driving an external load on XVDD.	
		00	500 µA
		01	1.25 mA
		10	2.50 mA
		11	5.00 mA
PWCHK_EN	R/W – Config	**PWCHK_EN** enables a qualifying process before turning on the on-chip LDO.	
		0	Disabled (on-chip LDO is turned on immediately when **LDO_EN** is set to '1' without qualifying process and **RSPW_RDY** is always be '1')
		1	Enabled
LDO_EN	R/W – Config	**LDO_EN** enables the on-chip LDO Regulator	
		0	Disables LDO
		1	Enables LDO
OSC_EN	R/W – Config	**OSC_EN** enables on-chip oscillator for UART communication.	
		0	Disables on-chip oscillator
		1	Enables on-chip oscillator

6.2.14 Register 0x0E: Peripheral_Adjustment

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RFU					LDO_D_LV	RFLM_LV	2B_FLAG

Parameter	Type	Description	
LDO_D_LV	R/W - Config	Defines a threshold voltage drop level at the pin **XVDD** that set **XVDD_RDY** flag.	
		0	2.4 volt
		1	2.7 volt
RFLM_LV	R/W - Config	Defines RF limiter level to limit a charging voltage on the pin **SCAP** and pin **HV**	
		0	5.2 volt (maximum of SCAP voltage = 4.8 volt)
		1	6.5 volt (maximum of SCAP voltage = 6.0 volt)
2B_FLAG	R/W – Config	Defines a number of response flags in an RF response packet to make the response compatible with some NFC chips/phone in market to guarantee minimum two bytes in a response frame.	
		0	Response Flag in a response package is set to one byte.
		1	Response Flag in a response package is set to two bytes for some downlink commands to guarantee that uplink response frame contains at least 2 bytes to make operation compatible with some NFC phones in market that don't accept one byte response.

7. Architecture and Peripheral Interface

Highlighted in colored blocks in Figure 7-1, the configurable peripheral from registers is RF-AFE, LDO, GPIO, UART and On-chip Oscillator. This section describes the relation between function of the sub blocks and related registers.

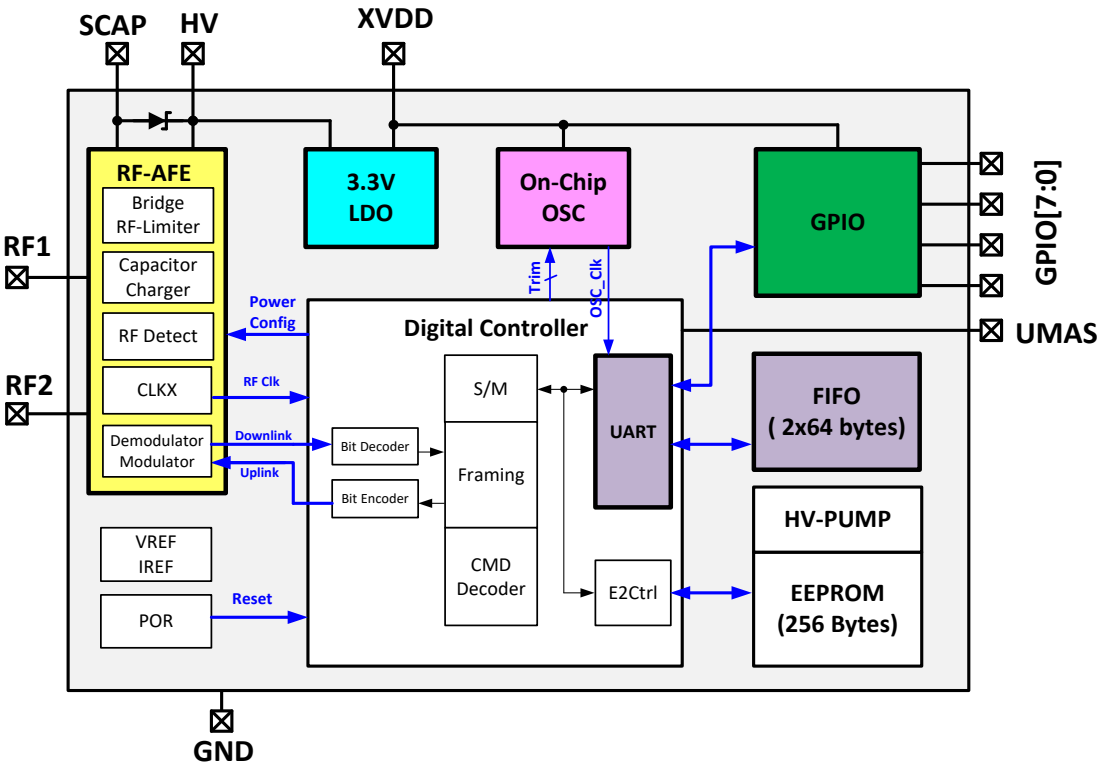


Figure 7-1: Configurable peripheral components

Mode of operation of SIC4310 can be classified in term of power source for operations and communications.

In terms of power source, modes of operation can be

- Power harvesting, where the device relies on power from RF and,
- Battery operated where power source is fed into the chip via the pin **XVDD**.

In terms of communication, modes of operation can be

- NFC mode, when logic on pin **UMAS** = '0'
- UART-connected EEPROM memory when logic on pin **UMAS** = '1'.

To operate a desired mode correctly, it is required to properly set key registers and logic on pin. Table 7-1 shows modes of operation versus key registers which affect to accessibility from both RF end and UART end.

Table 7-1: Communication versus Pin **UMAS** setting, register setting and power source

PIN UMAS	External XVDD Usage on pin XVDD	Register LDO_EN	Register OSC_EN	RF → EEPROM (RdE2 , WrE2)	RF → GPIO	RF→UART (TxRU, RxUR, RxRU)	UART→EEPROM *(UART_Write_E2, UART_Read_E2)*
0	No	0	X	Can			
		1	0	Can	Can		
		1	1	Can	Can	Can	
	Yes*	X	0	Can	Can		
		X	1	Can	Can	Can	
1	No***	1**	X				Can**
	Yes*	X	X				Can**

X mean do not care

* External XVDD Supply > 3.0 V.

** EEPROM reload values for initialization must be configured to enable UART and LDO.

*** In case of power harvesting, it is recommended to not tide pin **UMAS** to pin **XVDD** as a logic '1' during initialization to prevent system stuck in reset state.

Note that all case, RF Power must be enough to drive a load.

7.1 RF Analog-Front-End

7.1.1 RF-field detector (RFD)

RF-field detector (RFD) detects presenting of the RF field and resets the RF state to ***"Idle".*** If pin **GPIO[0]** or **GPIO[6]** is configured to a special mode, the signal can be used to inform an external UART-connected device that the RF field is present. The RF field detector can operate as soon as the system is powered on regardless of digital controller's state.

7.1.2 Super capacitor charger

The RF-AFE is designed to collect excess energy from the RF field to store in a super-capacitor connected to the pin **SCAP**. If the energy stored in this capacitor is high enough (> 4.5 Volt), storage energy can be used to power the SIC4310 itself when the RF power is absent for a certain time or to stabilize the power for a load connected on the pin **XVDD** including an internal oscillator. The maximum voltage on the pin **SCAP** can be selected either to be 4.8 volt or 6.0 volt depending on a rated voltage of an attached capacitor. The voltage on the pin **SCAP** can be monitored if it is over 4.5 volt by reading the register **SCAP_RDY** (0x01.4). the pin **SCAP** can be left floating without any effect. To use power from the charged super capacitor in power harvesting configuration through the pin **XVDD**, UART connected devices e.g., MCU relying on power from **XVDD** must set the pin **UMAS** to '1' before the RF field is off and preset values for **LDO_EN** in EEPROM must be set to '1'. Alternatively, user can use power from the super capacitor directly. Functions of registers associated with the pin **SCAP** are described in Table 7-2. Note that the register **SCAP_RDY** is not available in SIC4310 rev.A, for which marking is 4310A.

Table 7-2: Registers associated with Super Capacitor Charger

Register	Address	Indication	Type	Default Value
SCAP_RDY	0x01.4	**SCAP_RDY** that equals to 1 means that voltage on the pin **SCAP** is higher than 4.5V	Read only	'0'
RFLM_LV	0x0E.1	**RFLM_LV** can set a voltage on the RF limiter and the maximum voltage on the pin **SCAP**. **RFLM_LV** = 0 means that the maximum of SCAP voltage = 4.8 volt. **RFLM_LV** = 0 means that the maximum of SCAP voltage = 6.0 volt.	Read-Write	'0'

7.2 Low-Drop-Out (LDO) regulator

The Low-Drop-Out (LDO) regulator is generally used in applications that require power from RF to operate an external load and UART communication such as offline parameters upgrading to MCU. The LDO supplies a stable 3.3-V power to an internal on-chip oscillator, external load on the pin **XVDD** and the GPIO. Provided that input power is high enough, the LDO has a maximum driving capability of 10mA. The LDO is factory trimmed with ±2% accuracy. Typically, the regulator requires 0.1µF and 10µF decoupling capacitor in parallel on the pin **XVDD** but at least 0.1µF decoupling capacitor is required to ensure the stability of the LDO. For loading higher than 1mA, it is recommended to connect another 0.1µF decoupling capacitor on the pin **HV.** External power source can supply power back through the pin **XVDD** without loading effect due to the LDO output circuit whether it is turned on or off.

Simplified circuit diagram of power flow from RF is illustrated in Figure 7-2. Input power from RF is designed to supply LDO for an external load on the pin **XVDD** as the first priority. The surplus power from supplying load on LDO is stored into a super capacitor, if any is, as reserved power. If the super capacitor is fully charged or not connected, surplus current sinks into a shunt regulator. Current flowing into a super capacitor or shunt regulator is monitored and qualified to turn on the LDO.

The LDO can operate when the **LDO_EN** (0x0D.1) is set to '1' and the reserve power before turning on is high enough. After setting **LDO_EN** to '1', the LDO may not immediately turn on until reserve power is detected that it is higher than a predefined level. If the reserve power is higher than the predefined level in a current monitoring unit, the power status **RSPW_RDY** (0x01.1) shows '1'. Note that the status of **RSPW_RDY** can be displayed at the pin **GPIO[5]** in the inversion logic *RSPW_RDY#* when it is set in a special function. If the **LDO_ON** (0x01.0) shows '1', the LDO is successfully turned on. This power qualifying mechanism is to ensure the stability in supplying an external load on the pin **XVDD**. Options for checking level are 500µA, 1.25mA, 2.5mA, or 5.0 mA which can be set by the register **PW_LEV** [1:0] (00b = 500 µA, 01b = 1.25mA, 10b = 2.5mA, 11b = 5mA). Although this power qualifying mechanism can be disabled by setting **PWCHK_EN** (0x0D.2) to '0', it is recommended to turn on to guarantee the power supplying stability.

The load power consumption on the pin **XVDD** should be less than the reserve power from the RF during power qualifying. After the LDO is turned on, the reserve power, is continuously monitored. If the reserve power is dropped below 20% of a pre-set checking level, the warning flag **"RSPW_LOW"** is set to '1' in the byte **B_NAK** in a response frame and status register **RSPW_RDY** and *RSPW_RDY#* toggle to '0' and '1" respectively.

The flag **"RSPW_LOW"** is still held at '1', although the reserve power becomes higher than 20% of a checking level. Afterwards, every UART response is equipped with a **B_NAK** until the command ***Clear_Flag*** with right flag has been received. The flag will be successfully cleared by this command if reserve power is higher than 20% of the checking level. When the **"RSPW_LOW"** is set, the LDO is not turned off and still supply power to a load until it hits its maximum current-sourcing capability. When LDO is overloaded, voltage on **XVDD** drops from 3.3V. Output regulated voltage level on the pin **XVDD** can be monitored via the register **XVDD_RDY** and flag **"XVDD_DROP".** Users should carefully ensure the maximum amount of load which should be less than the sourcing capability. Note that voltage on **XVDD** may drop until flag **"XVDD_DROP"** or even **"UART_FAIL"** is set during receiving commands and large load on **XVDD.**

In some occurrences, load on **XVDD** may draw a current from input until there is no reserve power to charge a super capacitor. Hence, the flag **"RSPW_LOW"** and status **RSPW_RDY** can be suddenly set because the reserve power becomes low after turning on the LDO.

The LDO is designed to protect communications between the SIC4310 and NFC/RFID devices from overloading, although the pin **XVDD** is shorted to ground. If LDO is not turned on and no power forced from external to the pin XVDD, **XVDD_RDY** and **UART_RDY** are always set to '0'. Also, the associated flag **"XVDD_DROP"** and **"UART_FAIL"** flag is assert. The LDO can only be switched off by setting **LDO_EN** to '0'. Generally, it is recommended to turn on the regulator by commands and default loading from registers should be '0'. Functions of associate registers are delineated in Table 7-3. Pins related to LDO are described in Table 7-4.

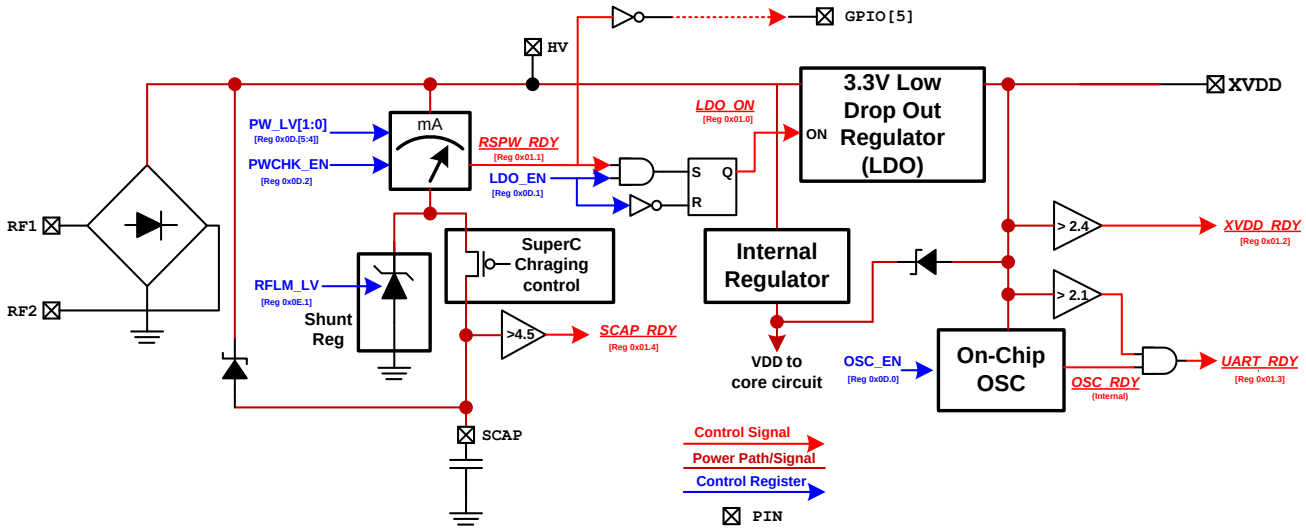


Figure 7-2: Simplified circuit diagram of power flow from RF to **XVDD** in SIC4310.

Table 7-3: Registers associated with the LDO

Register	Address	Function	Type	Factory preset value
LDO_ON	0x01.0	Indicator shows that LDO is already turned on.	Read only	'0'
RSPW_RDY	0x01.1	Indicator shows that the reserve power is high enough to turn LDO on.	Read only	'0'
XVDD_RDY	0x01.2	Indicator shows voltage on the pin **XVDD** is higher than a defined level. Power can be either from RF power harvesting or external source at the pin **XVDD**.	Read only	'0'
UART_RDY	0x01.3	Indicator shows that the UART module is ready to operate.	Read only	'0'
LDO_EN	0x0D.1	Configuration for turning on LDO.	Read/Write	'0'
PWCHK_EN	0x0D.2	Configuration to enable the power qualifying process.	Read/Write	'1'
PW_LV[1:0]	0x0D.[5:4]	Configuration for defining a threshold of reserved power level received from RF to turn on LDO.	Read/Write	10b
LDO_D_LV	0x0E.2	Adjustment for a threshold voltage drop level at the pin **XVDD** for **XVDD_RDY** status.	Read/Write	'0'

Table 7-4: Pins related to LDO

Pin Name	Pin #	Function
HV	5	**Unregulated power supply pin** - For large load supplying, 0.1μF decoupling capacitor is required at this pin.
XVDD	3	**VDD supply pin** - Output for power port for RF power harvesting mode. - Input for power port for external-power-source operated device. - At least 0.1μF decoupling capacitor is required at this pin. - For large load supplying, 0.1μF and 10μF decoupling capacitor are required at this pin.
GPIO[5] (RSPW_RDY#)	13	**Reserve Power Ready indicator pin** - Indicator showing reserve power is high enough to turn LDO on (active low).

7.3 GPIO (General Purpose Input and Output)

The GPIO in the SIC4310 is intended for the UART communication, chip's status indicator and general-purpose input/output operations based on **GPIO_MODE.** Register **GPIO_DIR** sets the direction of GPIO pins to be either input or output. NFC/RFID reader devices can write to the register **GPIO_OUT** to set the output value to '0' or '1' in case of output and can read the status from the register **GPIO_IN** to get logic value present on each pin in case of input, although the GPIO is set to function as output. When GPIO is input, each GPIO can be individually pulled-up by **GPIO_PU**. The internal pull-up register is approximately 120k. The sourcing capability is 4 mA per GPIO pin and operating voltage is from 2.7V to 3.3V. Functionalities and direction for GPIOs are summarized in Table 7-5.

Table 7-5: Pin functionalities

PIN #	PIN Name	DIR[x] = 0 (Input)		DIR[x] = 1 (Output)	
		Mode[x] = 0	Mode[x] = 1	Mode[x] = 0	Mode[x] = 1
2	**GPIO[0]**	**Input [0]**	*UART-CTS#*	**Output [0]**	*RF_Detect*
1	**GPIO[1]**	**Input [1]**		**Output [1]**	*UART-TX*
16	**GPIO[2]**	**Input [2]**	*UART-RX*	**Output [2]**	
15	**GPIO[3]**	**Input [3]**		**Output [3]**	*UART-RTS#*
14	**GPIO[4]**	**Input [4]**		**Output [4]**	*RFBusy*
13	**GPIO[5]**	**Input [5]**		**Output [5]**	*RSPW_RDY#*
12	**GPIO[6]**	**Input [6]**		**Output [6]**	*RF_Detect*
11	**GPIO[7]**	**Input [7]**		**Output [7]**	

Table 7-6: Registers associated with the LDO

Register	Address	Functions	Type	Factory preset Value
GPIO_DIR	0x08	Direction of GPIO	Read/Write	0x03
GPIO_MODE	0x09	Special Function of GPIO	Read/Write	0x07
GPIO_OUT	0x0A	Output value of GPIO	Read/Write	0x00
GPIO_IN	0x0B	Input value of GPIO	Read only	0x00
GPIO_PU	0x0C	Pull up enable of GPIO	Read/Write	0x00

When **GPIO[0]** is set to function as *UART-CTS#,* if **GPIO[0]** receives logic '0', SIC4310 sends data out through *UART-TX* to an end device when data is available in the FIFO. When **GPIO[0]** is not set to function as *UART-CTS#,* register **CTS** is always set to '1'. This means SIC4310 always sends data out through *UART-TX* to an end device when data is available in the FIFO. The **GPIO[0]** and/or **GPIO[6]** can be set to display *RF_Detect* indicating a presence of the RF field. When *RF_Detect* displays logic '1', UART-connected device shall aware in rising the pin **UMAS** to '1', else NFC/RF device will be disconnected from the SIC4310.

When **GPIO[3]** is set to function as *UART-RTS#,* signal *UART-RTS#* is active low when uplink FIFO (UART-receive FIFO) is ready to receive data.

The **GPIO[4]** can be set to display signal *RFBusy* to inform UART-connected device to know an internal status of the SIC4310. The purpose of *RFBusy* is to prevent an external UART device to interrupt chip's process being operated e.g., EEPROM programming. If the *RFBusy* is active high, UART-connected device shall not raise the pin **UMAS** to '1'. The *RFBusy* is active high when the SIC4310 is operating between beginning of receiving downlink and end of transmitting uplink. Behaviour of *RFBusy* is depicted in Figure 7-3.

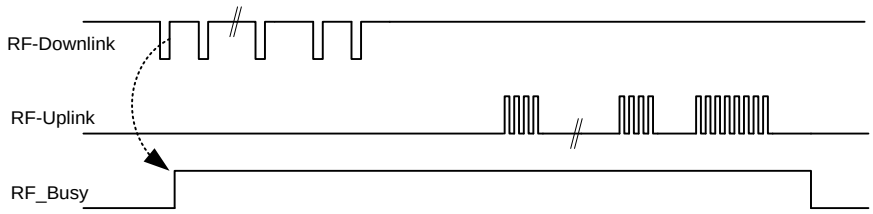


Figure 7-3: Activity of RFBusy versus downlink and uplink

The **GPIO[5]** can be set to display *RSPW_RDY#,* indicating a status of input power, in a special function mode.

7.4 UART

The UART module can operate when the register **OSC_EN** is set to '1' and voltage on the pin **XVDD** is higher than 2.1V. Also, the UART module is definitely set to operate when the pin **UMAS** is raised to '1'. When the **UART_RDY** status becomes active high, this indicates that the UART module is ready to operate. The pin **GPIO[1]** and **GPIO[2]** must be set to function as UART ports namely UART-TX and UART-RX respectively. Also, the SIC4310 can communicate to UART end device by handshaking function. This function is automatically enabled when the direction and mode of associate **GPIO[0]** and **GPIO[3]**are matched to the function as described in Table 7-5. Handshaking may require for high-speed communication applications.

When the SIC4310 gets any RF-UART commands from its companion NFC/RFID device, RF command decoder checks integrity of UART data in the command packet and stores the correct data into the downlink FIFO for UART transmission. The UART data is transmitted towards the UART end device via the pin **GPIO[1]** functioning as UART-TX. When pin **GPIO[2]**functioning as UART-RX receives any incoming data packet, the UART receiver stores them into the uplink FIFO and waits until it gets read command ***RxUR*** from NFC device.

Transmitting and receiving packet format can be set by UART-related register configuration, which **are UART_Frame, UART_Devisor_m, UART_Devisor_n. UART_Frame** defines a length of stop bit and parity appended at the end of each byte. Then, when there is a parity error during reception, such receiving byte is neglected. **UART_Devisor_m** and **UART_Devisor_n** defines UART bit rate.

The speed of UART bit rate can be configured by **UART_DIV_m (m)** and **UART_DIV_n (n).** UART bit rate is defined as follows.

UART_Bit_Rate = (1843200) / (m*n)

The value of **n** should be set higher than **m**. The higher value of **n** results in a higher accuracy in receiving UART packets. The appropriate range of bitrate is from 115200 bps to 457bps. The maximum stable data rate is 115200 bps (m =1, n = 16) while the minimum data rate is 457 bps (m=0, n=3F). Note that the divisor is 64, when **m** is set to 0. Minimum value of n is 2. If **n** < 2 is set, the SIC4310 cannot perform communication. In real practise, the UART speed can be configured from 9.6 kbps to 115.2 kbps. Table 7-7 shows an example of UART data rate. Functions of registers associated with UART module are listed in the Table 7-8. Note that the factory pre-program bit rate is 115200 bps.

Table 7-7: Example of UART data rate

Bit Rate (bps)	m	n	Data Rate (bps)	Error (%)
115200	1	16	115200	0.00
57600	2	16	57600	0.00
38400	1	48	38400	0.00
19200	2	48	19200	0.00
9600	4	48	9600	0.00

Table 7-8: Registers associated with UART module

Register	Address	Functions	Type	Factory preset Value
CTS	0x00.5	Indicator shows that UART device is ready to receive data. If pin **GPIO[0]**is set to not function as *UART-CTS#,* CTS becomes '1'	Read Only	-
RTS	0x00.4	Indicator shows that UART device is ready to receive data. If pin **GPIO[3]**is set to not function as *UART-RTS#,* RTS becomes '1'	Read Only	-
DL_FF_EMT	0x00.3	Indicator shows Downlink FIFO is empty.	Read Only	-
DL_FF_OVF	0x00.2	Indicator shows Downlink FIFO has been overflowed.	Read Only	-
UL_FF_EMT	0x00.1	Indicator shows Uplink FIFO is empty.	Read Only	-
UL_FF_OVF	0x00.0	Indicator shows Uplink FIFO has been overflowed.	Read Only	-
UART_RDY	0x01.3	Indicator shows UART module is ready to operate.	Read/Write	-
TRxRU_Time	0x03	Configuration defines the response timeout for ***TRxRU*** command.	Read/Write	0100b
Stop_Len	0x04.3	UART Stop length bit control register.	Read/Write	'1'
Parity	0x04.[2:0]	UART Parity bit control register.	Read/Write	000b
UART_DIV_m	0x05	Devisor m for UART Clock	Read/Write	000001b
UART_DIV_n	0x06	Devisor n for UART Clock	Read/Write	010000b

7.5 On-chip oscillator

The oscillator is to provide a stable clock source for UART communication and to be a base frequency for UART controller which runs at the frequency of 1.8432 MHz. The temperature sensitivity of the oscillator is compensated, and it is factory-trimmed within ±2% over temperature. The oscillator can operate when register **OSC_EN** is set to '1'. If the register **OSC_EN** is set and the final frequency settles, the **UART_RDY** shows '1' which indicates the oscillator is turned on stably and UART module is ready to operate.

The oscillator can be fine trimmed by user through register **OSC_Tunning.** When **OSC_Tuning** is set to "1111b", oscillator frequency is brought to minimum tuneable frequency. On the other hand, oscillator frequency is set to maximum tuneable frequency when **OSC_Tuning** is set to "0000b". Typically, it is not necessary to set this register because the oscillator is well-trimmed from the factory. Tuning resolution is about 0.2-0.5% per step. Functions of associate registers are delineated in Table 7-9.

Table 7-9: Registers associated with oscillator module

Register	Address	Functions	Type	Factory preset Value
UART_RDY	0x01.3	Indicator showing UART is ready to operate.	Read only	-
OSC_Tuning	0x07	Configuration for fine tuning frequency of oscillator for the UART communication.	Read/Write	0x08
OSC_EN	0x0D.0	When UART function is not required, OSC_EN can be set to '0' to save power.	Read/Write	'0'

8. Commands

The SIC4310 supports four sets of operational commands which are basic RFID commands, RFID-UART commands, RF-configuration commands and UART commands.

8.1 Basic RFID commands

The basic RFID commands make the SIC4310 communicate with NFC/RFID reader devices in both downlink and uplink. This group of commands' formats is based on the PICC states of the ISO 14443-3 standard. The Basic RFID commands are utilized in identifying UID and accessing EEPROM memory as a normal RFID.

8.1.1 REQA

REQA command changes the SIC4310 being in the ***"Idle"*** state into the ***"Ready1"*** state and make the transponder participate in further anti-collision and selection procedures. In response of ***REQA,*** the transponder sends 2 bytes **ATQA** back to the NFC/RFID reader device. Although the formula for response time (128*n-204) looks different from that stated in the ISO 14443 standard, it is as same as that of the ISO 14443 standard because it counts from end of downlink frame.

Table 8-1: ***REQA*** command format

CMD	**REQA**	
Format	0x26 (7 bits)	
Response	Successful operation	**ATQA** (2 bytes)
	Error	No response
Operation	Change state from the ***"Idle"*** state into the ***"Ready1"*** state.	

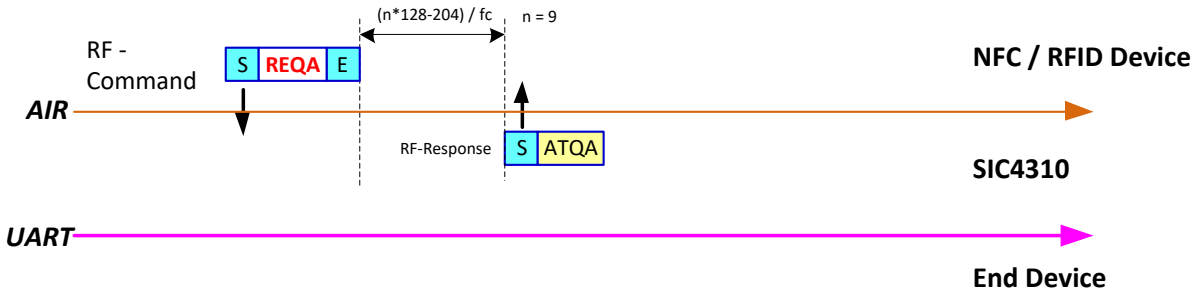


Figure 8-1: ***REQA*** command frame with a response

8.1.2 WUPA

The purpose of ***WUPA*** command is as same as ***REQA.*** The only difference is ***WUPA*** can be used in both ***"Idle"*** and ***"Halt"*** state.

Table 8-2: ***WUPA*** command format

CMD	**WUPA**	
Format	0x52 (7 bits)	
Response	Successful operation	**ATQA** (2 bytes)
	Error	No response
Operation	Change state from the ***"Idle"*** or ***"Halt"*** state into the ***"Ready1"*** state.	

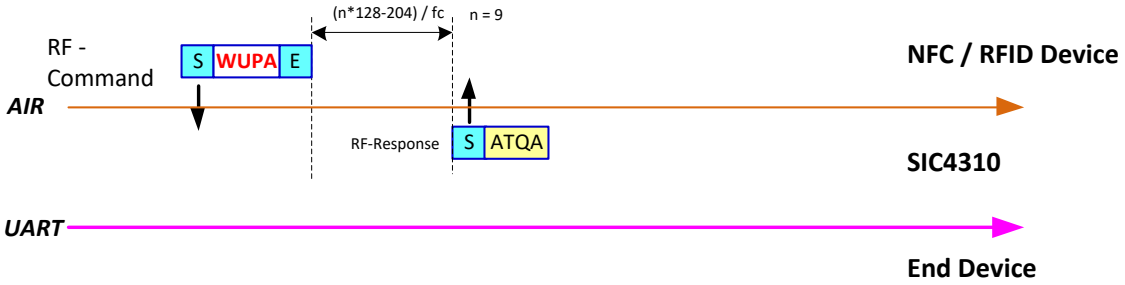


Figure 8-2: ***WUPA*** command frame with a response

8.1.3 ANTI-COLLISION

The ***ANTI-COLLISION*** command is used in the anti-collision procedure with bit oriented anti-collision frames. The purpose of ***ANTI-COLLISION*** command is to identify the target transponder and retrieve UID. The ***ANTI-COLLISION*** command can be used in both cascade level 1, which states are ***"Ready1"***, ***"Ready1*"*** and cascade level 2 which states are ***"Ready2"*** and ***"Ready2*"*** state. The ***ANTI-COLLISION*** consists of **SEL** code representing current cascaded level, number of valid bit **(NVB)** and data. In the cascade level 1, the **SEL** code is 0x93 while the SEL code is 0x95 for the cascaded level 2. Transaction of ***ANTI-COLLISION*** command and its response in both cascade level 1 and cascade level 2 are depicted in Figure 8-3 and Figure 8-4. For the cascade level 1, the SIC4310 response CT (cascade tag) code and first 3-byte of UID. The **CT** code is 0x88.

Table 8-3: ***ANTI-COLLISION*** command format

CMD	***ANTI-COLLISION***	
Format	**SEL + NVB + Data** Cascade level1 : **0x93 + NVB + Data** Cascade level2 : **0x95 + NVB + Data**	
Response	Successful Operation	UID
	Error	No response
Operation	Response remaining part of UID and it BCC	

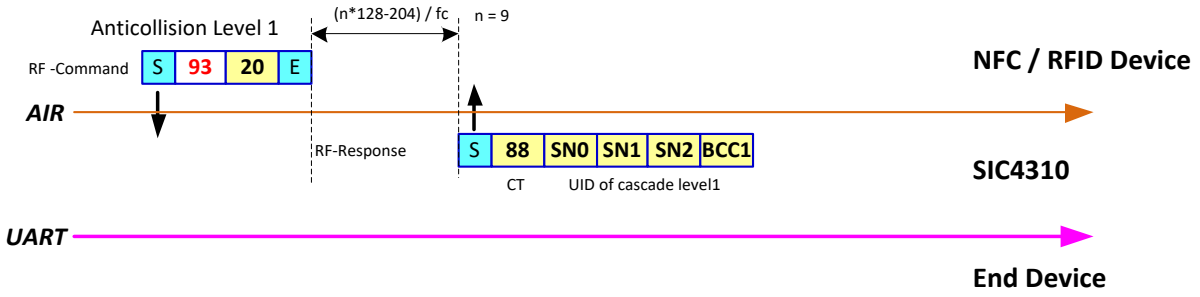


Figure 8-3: ***ANTI-COLLISION*** in the cascade level 1 with a response

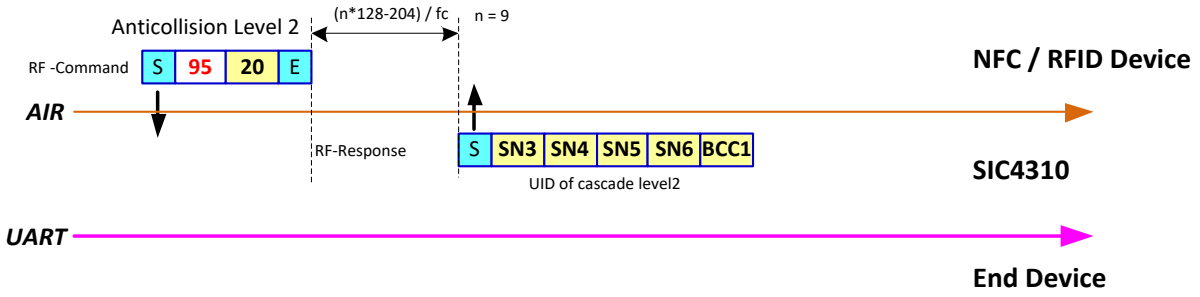


Figure 8-4: ***ANTI-COLLISION*** in the cascade level 2 with a response

8.1.4 SELECT

The ***SELECT*** command format is based on the same structure as the ***ANTI-COLLISION*** command with 2-byte CRC appended at the end. The SIC4310 responds to NFC/RFID reader device with a **SAK** (select acknowledgement) code of 0x04 in ***"Ready1", "Ready1*"*** state, indicating UID is not complete and **SAK** code of 0x00 in ***"Ready2", "Ready2*",*** indicating UID is complete and state transits to ***"Active"*** state or ***"Active*".*** Figure 8-5 and Figure 8-6 show the ***SELECT*** command for the cascade level1 and cascade level2, respectively.

Table 8-4: ***SELECT*** command format

CMD	***SELECT***	
Format	SEL + NVB + Data Cascade level1 : **0x93 + 0x70 + UID (4 bytes) + BCC + CRC** Cascade level2 : **0x95 + 0x70 + UID (4 bytes) + BCC + CRC**	
Response	Successful operation	**SAK + CRC** **SAK** = 0x04 for cascade level 1 **SAK** = 0x00 for cascade level 2
	Error	No response
Operation	Change state from ***"Ready1"*** or ***"Ready1*"*** to ***"Ready2"*** or ***"Ready2*"*** , or change state from ***"Ready2"*** or ***"Ready2*"*** to ***"Active"*** or ***"Active*".*** Respond SAK (select acknowledgement).	

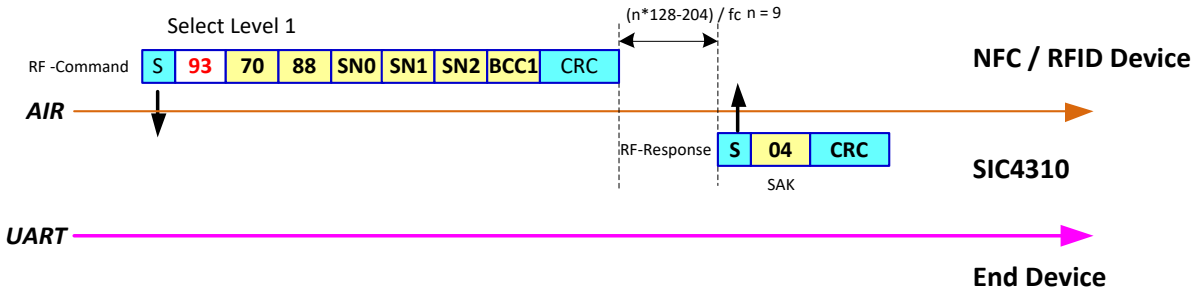


Figure 8-5: ***SELECT*** level1 command frame with a response

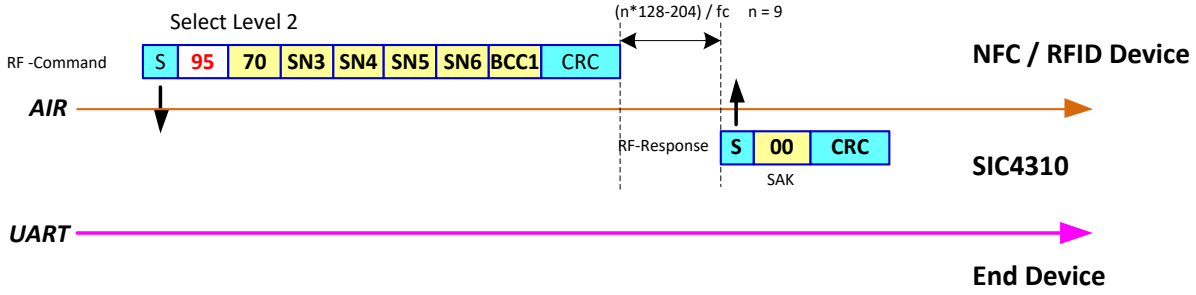


Figure 8-6: ***SELECT*** level2 command frame with a response

8.1.5 HLTA

The purpose of ***HLTA*** command is to move transponder that is already processed into a waiting state. The SIC4310 receiving ***HLTA*** in ***"Active"*** or ***"Active*"*** state changes its state to ***"Halt".*** By using this command, the NFC/RFID reader device can identify the transponders, which are already read and those have not yet been read. The SIC4310 that receives ***HLTA*** in ***"Ready1"*** and ***"Ready2"*** transits to ***"Idle".*** Receiving ***HLTA*** in other state changes the state to the ***"Halt"*** state. There is no response sent back to NFC/RFID reader device for this command. Note that any change of lock bit in EEPROM is reloaded when state jumps back to ***"Idle"*** or ***"Halt"*** and lock bit effects after that.

Table 8-5: ***HLTA*** command format

CMD	***HLTA***
Format	**0x50 + 0x00 + CRC**
Response	None
Operation	Change state from ***"Active"*** or ***"Active*"*** to ***"Halt"*** state Reload lock bit in EEPROM to make it effect take place

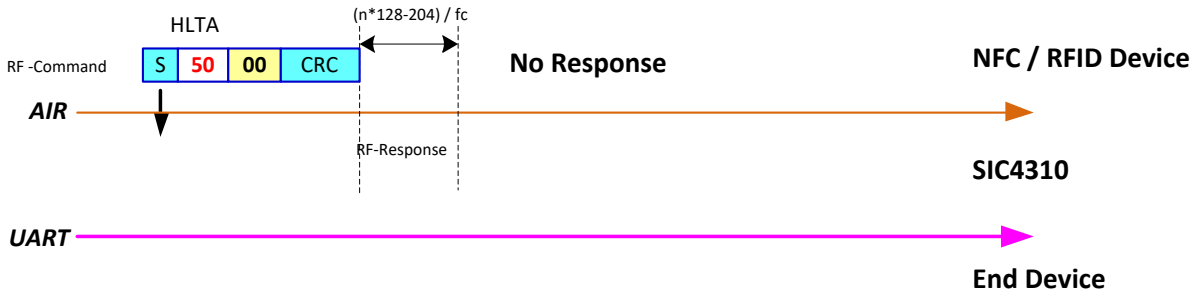


Figure 8-7: ***HALT*** command frame

8.1.6 ReadE2

The purpose of ***ReadE2*** command is to read the EEPROM content. The ***ReadE2*** command contains a page address with a valid CRC. If the transponder gets a valid address in command, it responds the NFC/RFID reader device by sending 16 bytes (4 pages) stating from the addressed page and if the address is not valid it sends a 4-bit NAK.

Table 8-6: ***ReadE2*** command format

CMD	***ReadE2***	
Format	**0x30 + Block + CRC** (2 bytes)	
Response	Successful operation	**BlockData** (16 bytes) + **CRC** (2 bytes)
	Error	**NAK** (4 bits)
Operation	Read data from EEPROM at a specific address	

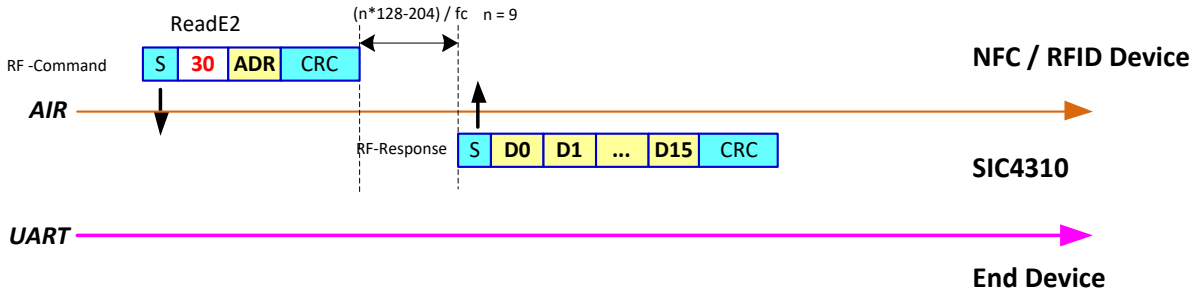


Figure 8-8: ***ReadE2*** command frame with response

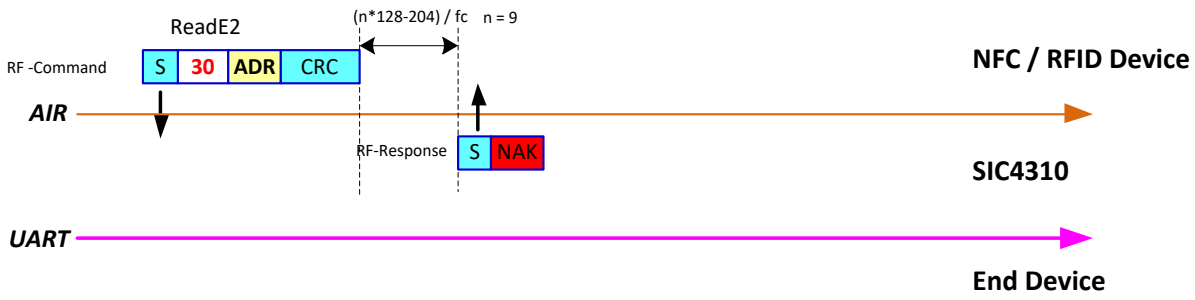


Figure 8-9: ***ReadE2*** command frame with a negative acknowledgement in response

8.1.7 WriteE2

The purpose of ***WriteE2*** command is to write the EEPROM, program lock bits, set bits in the OTP byte and preset an initial register value. The SIC4310 is receiving the ***WriteE2*** command with a valid address in the ***"Active"*** or ***"Active*"*** state programs received 4-bytes data to the addressed page and sends an **ACK** to the NFC/RFID reader device. If the address is not valid or the addressed page is already locked, the SIC4310 responds with a **NAK**.

Table 8-7: ***WriteE2*** command format

CMD	***WriteE2***	
Format	**0xA2 + ADR + D0 + D1 + D2 + D3 + CRC** (2 bytes)	
Response	Successful operation	**ACK** (4 bits)
	Error	**NAK** (4 bits)
Operation	Check permission at target address and write data to EEPROM	

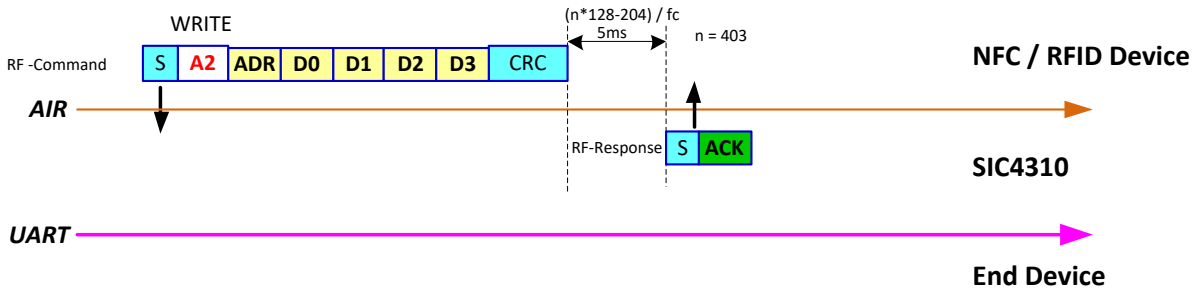


Figure 8-10: ***WriteE2*** command frame with an **ACK** response indicating successful operation

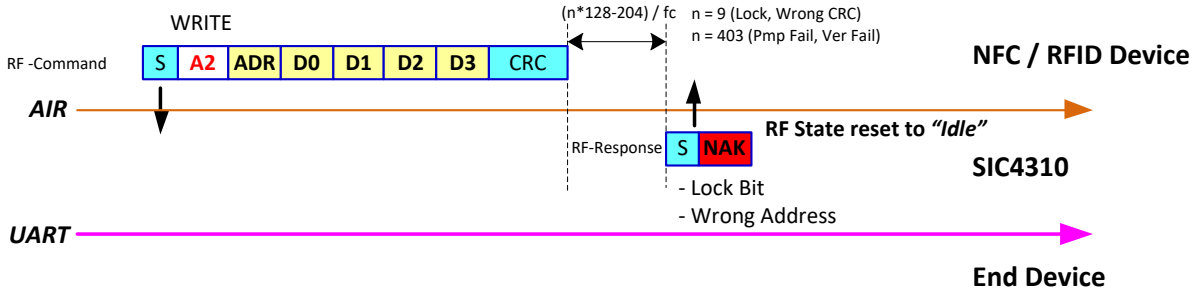


Figure 8-11: ***WriteE2*** command frame with a **NAK** response indicating unsuccessful operation

8.1.8 Compatible WriteE2

The purpose of ***Compatible WriteE2*** command is to make programming process backward compatible with the MIFARE classic system. The command contains a page address with a CRC. If the SIC4310 gets a valid address, it responses the NFC/RFID reader device with an **ACK**, else a **NAK**. The NFC/RFID reader device again sends 16-byte data but only the first 4 bytes are written into the memory. It is recommended to set the remaining bytes to '0'. Process of executing the ***Compatible WriteE2*** command is depicted in the Figure 8-12 and Figure 8-13.

Table 8-8: ***Compatible WriteE2*** command format

CMD	***Compatible Write E2***	
Format1	**0xA0 + ADR + CRC** (2 bytes)	
Response1	Successful operation	**ACK** (4 bits)
	Error	**NAK** (4 bits)
Format2	Block Data (16 bytes) + CRC (2 bytes)	
Response2	Successful operation	**ACK** (4 bits)
	Error	**NAK** (4 bits)
Operation	Check permission at target address and write data to EEPROM (only first 4 bytes are written)	

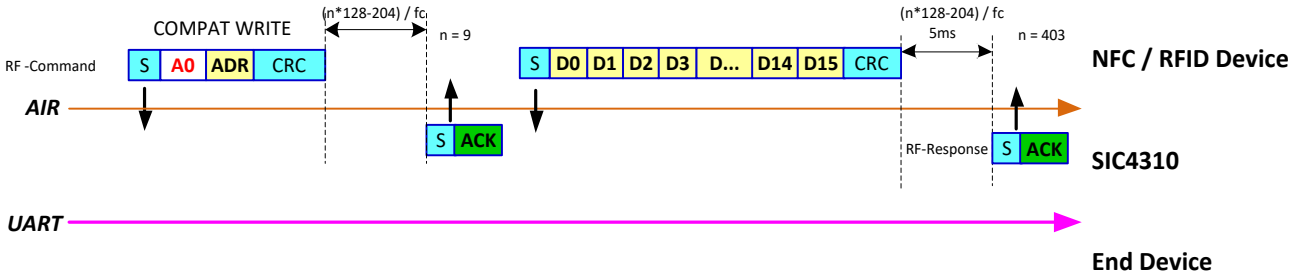


Figure 8-12: Two-step operation of ***Compatible Write E2*** with an ACK response

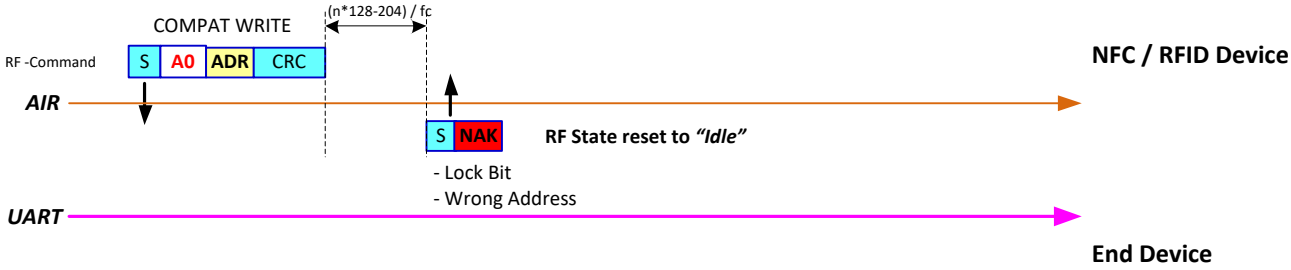


Figure 8-13: One-step operation of ***Compatible Write E2*** with a NAK response

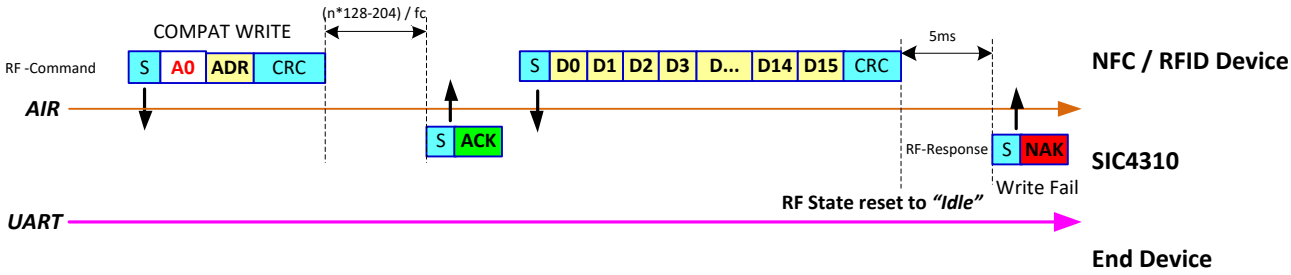


Figure 8-14: Two-step operation of ***Compatible Write E2*** with a NAK response

8.2 RF-UART commands

RF-UART commands are employed to control a register to transmit direct transparent data from NFC/RFID device to UART or vice versa.

8.2.1 TxRU

The purpose of ***TxRU*** command is to convey data in a payload from NFC/RFID reader device to the UART. The data in the payload must be less than 64 bytes. If the received frame has no error, the SIC4310 sends a **B_ACK** back to the NFC/RFID reader device and transmits data in downlink FIFO to UART. If a framing error occurs during the RF downlink, transponder transmits a 4-bit **NAK** and change state to ***"Idle"*** or ***"Halt".*** If downlink FIFO overflows, The SIC4310 sends an 8-bit **B_NAK** and chip's state remains in ***"Active".*** Possible response of ***TxRU*** command is depicted in Figure 8-15, Figure 8-16 and Figure 8-17. Note that the flag **"UL_FF_OVF"** is not reported in this command, although it is set.

Table 8-9: ***TxRU*** command format

CMD	***TxRU***		
Format	**0xB1 + Data_Payload + CRC** (Data_PayLoad < 64 Bytes)		
Response	**2B_FLAG = 1**	Successful operation	**B_ACK + B_ACK + CRC**
		SIC4310 Error	**B_NAK + B_NAK + CRC** When - Downlink FIFO is overflown, or - Any power indicators are flagged.
		CRC error, RF error, Framing error	**NAK** (4 bits)
	2B_FLAG = 0	Successful operation	**B_ACK + CRC**
		SIC4310 Error	**B_NAK + CRC** When - Downlink FIFO is overflown, or - Any power indicators are flagged.
		CRC error, RF error, Framing error	**NAK** (4 bits)
Operation	Write Data to Downlink FIFO. If received frame has no error, start sending received payload data to UART.		

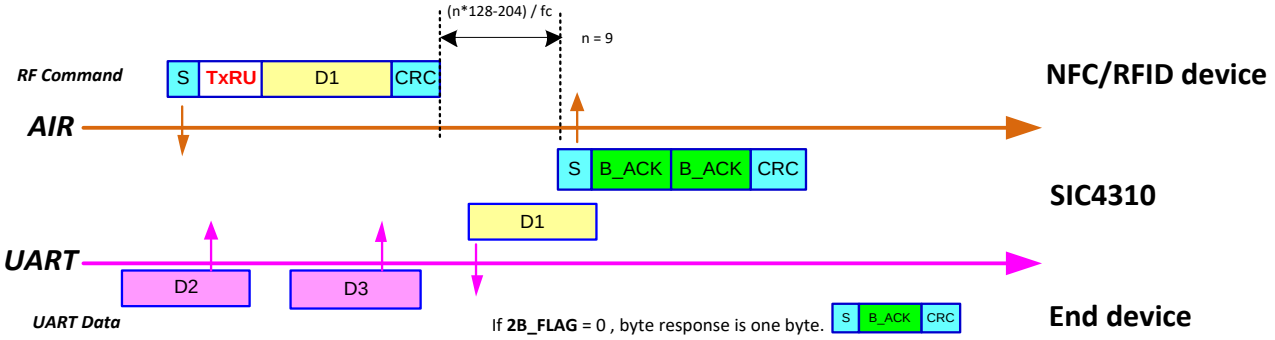


Figure 8-15: successful ***TxRU*** command frame.

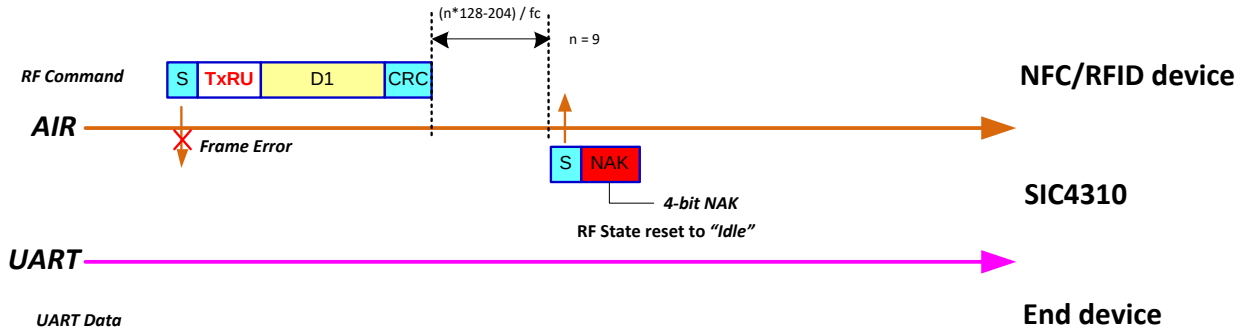


Figure 8-16: Framing error during transmitting the ***TxRU*** command frame with a 4-bit NAK response

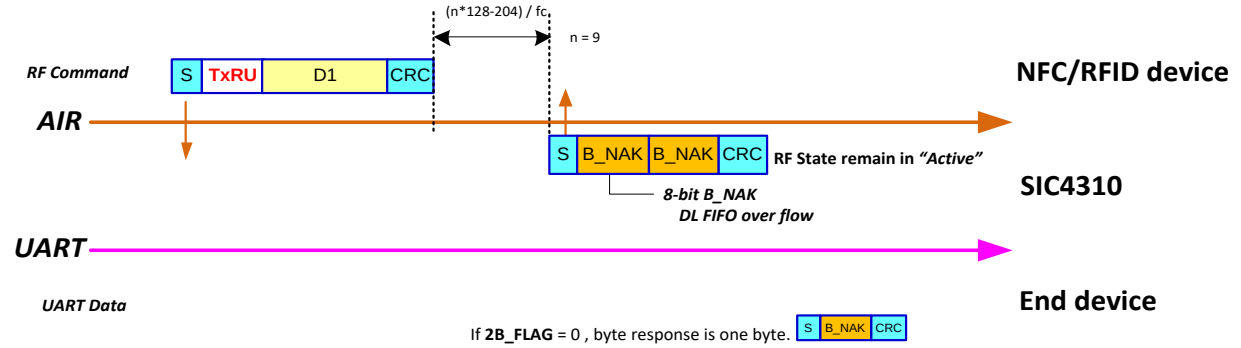


Figure 8-17: ***B_NAK*** response when Downlink FIFO overflows

8.2.2 RxUR

The purpose of command ***RxUR*** is to retrieve remaining data in the uplink FIFO. After receiving the ***RxUR*** command, the SIC4310 responds with **B_NAK** with remaining data in the uplink FIFO as a payload. If the uplink FIFO is empty, an 8-bit **B_NAK** is sent back in a response frame and state remains at ***"Active".*** If the uplink FIFO has been written until overflown or is overflown during the uplink process, uplink FIFO stops receiving data from UART end device and stops transmission in the uplink frame. State remains at ***"Active".*** For this reason, after executing the ***RxUR*** *command,* NFC device/RFID reader needs to check the uplink overflow flag **UL_FF_OVF.** If the overflow is set, **Clear_Flag** command must be sent to clear the overflow flag. Example of ***RxUR*** process is shown in Figure 8-18, Figure 8-19 and Figure 8-20.

Table 8-10: ***RxUR*** command format

CMD	***RxUR***		
Format	**0xB2 + 0x00 + CRC**		
Response	**2B_FLAG = 1**	Successful operation	**B_ACK + Data_Payload + CRC**
		Operation Error	**B_NAK + B_NAK + CRC** When - Uplink FIFO is overflowed or empty, or - Any power indicators are flagged.
		CRC error, RF error, Framing error	**NAK** (4 bits)
	2B_FLAG = 0	Successful operation	**B_ACK + Data_Payload + CRC**
		Operation Error	**B_NAK + CRC** When - Uplink FIFO is overflowed or empty, or - Any power indicators are flagged
		CRC error, RF error, Framing error	**NAK** (4 bits)
Operation	Retrieve data from uplink FIFO and uplink to NFC device/RFID reader.		

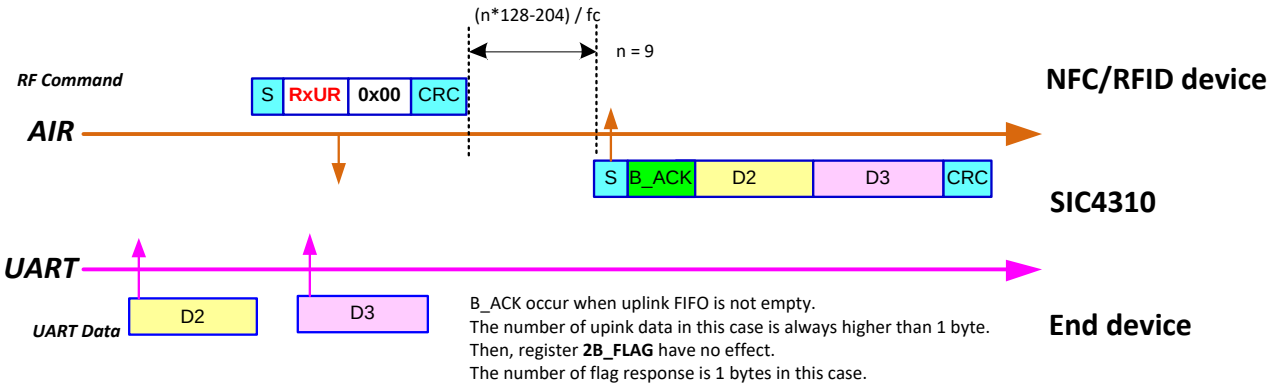


Figure 8-18: ***RxUR*** command frame and a B_ACK response from SIC4310 with a payload

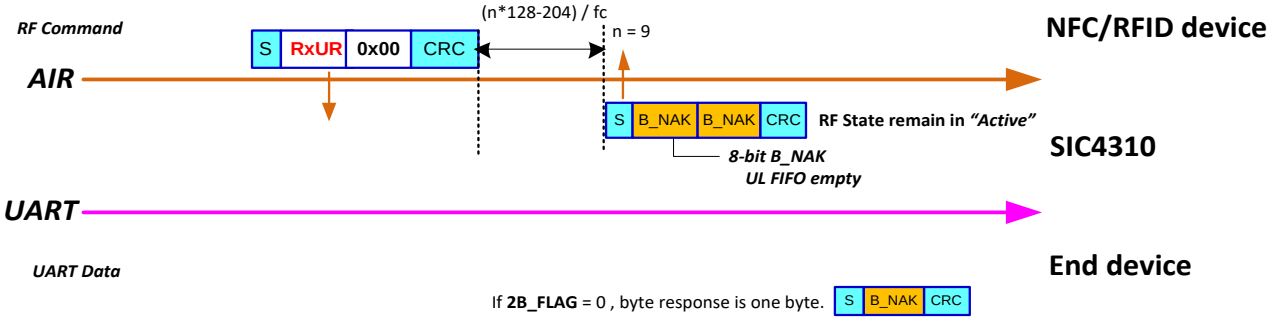


Figure 8-19: ***RxUR*** command and a B_NAK response due to the empty uplink FIFO

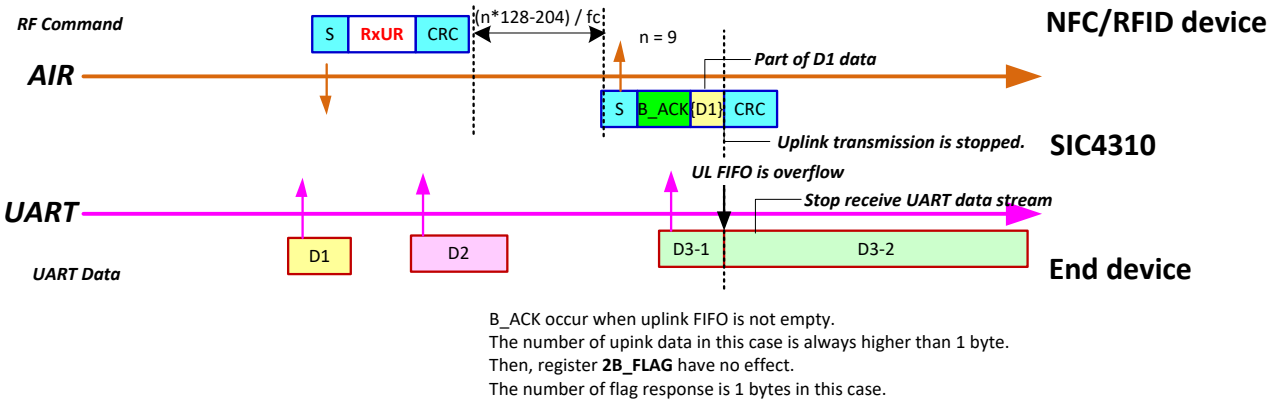


Figure 8-20: ***RxUR*** command and response of ***RxUR*** during uplink-FIFO overflow

8.2.3 TRxRU

The purpose of command ***TRxRU*** is to transmit data from NFC device/RFID readers and retrieve data back from UART end device in one command. Depending on systems or applications, wait time for response packets from UART end device, called TRxRU_Resp_Time, must be appropriately set via the register **TRxRU_Time** to prevent system stuck in case of no data in uplink FIFO**.** After receiving a packet from UART end device, SIC4310 responds back to NFC device/RFID reader with **B_ACK** and received UART data packet**.** If the UL FIFO is empty, the SIC4310 sends a B_NAK to the NFC device/RFID reader. Operation of **TRxRU** is depicted in Figure 8-21 and Figure 8-22.

Table 8-11: ***TRxRU*** command format

CMD	***TRxRU***		
Format	**0xB3 + Data_Payload + CRC** Data_PayLoad < 64 Bytes		
Response	**2B_FLAG = 1**	Successful operation	**B_ACK + Data_Payload + CRC**
		Operation error	**B_NAK + B_NAK + CRC** when - Downlink FIFO is overflown, or - Uplink FIFO is still empty within a defined timeout or - Uplink FIFO is already overflown, or - Any power indicators are flagged
		CRC error, RF error, Framing error	**NAK** (4 bits)
	2B_FLAG = 0	Successful operation	**B_ACK + Data_Payload + CRC**
		Operation error	**B_NAK + CRC** when - Downlink FIFO is overflow or - Uplink FIFO is still empty within a defined timeout or - Uplink FIFO is already overflown, or - Any power indicators are flagged
		CRC error, RF error, Framing error	**NAK** (4 bits)
Operation	Transmit data from the NFC device/RFID reader to downlink FIFO and retrieve data from the uplink FIFO.		

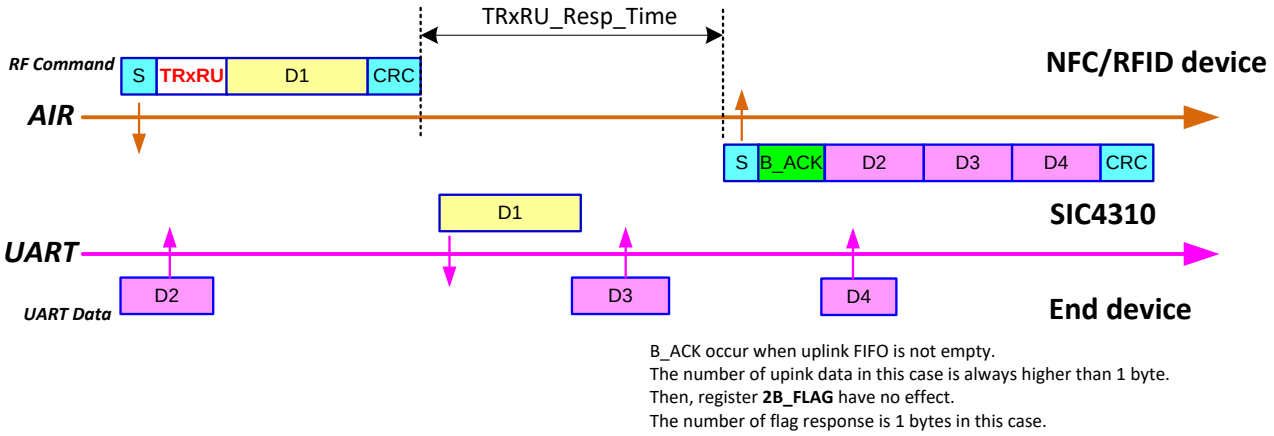


Figure 8-21: ***TRxRU*** command frame with a **B_ACK** response

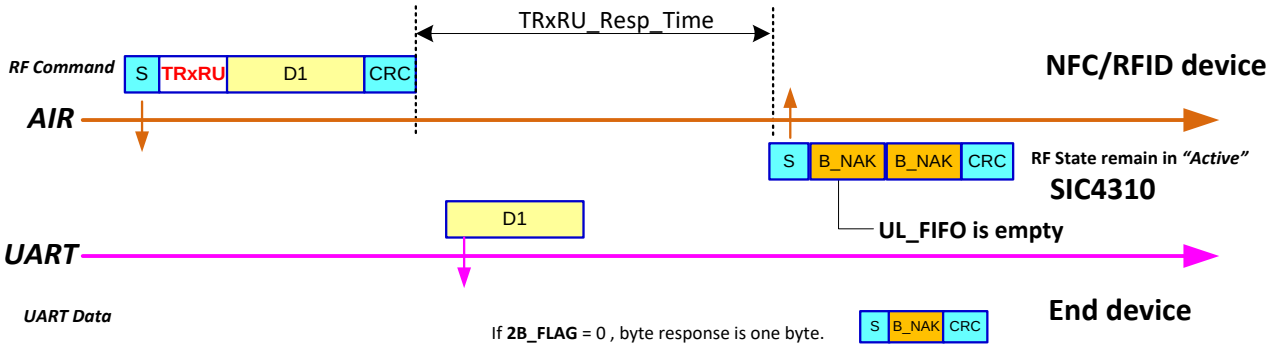


Figure 8-22: ***TRxRU*** command frame with a B_NAK response when the downlink FIFO is empty.

8.2.4 Clear_Flag

NFC/RF devices send the ***Clear_Flag*** command to clear the error flags presenting in the last **B_NAK** to make the communication process going on. The error flags are **DL_FF_OVF**, **UL_FF_OVF**, **RSPW_LOW**, **XVDD_DROP** and **UART_FAIL.** The command contains "Clear_Byte", which can be set to clear each error flag individually. The detail of "Clear_Byte" is shown in Table 8-12.

If there is no protocol error such as CRC error, RF error or framing error, the response from this command is always **B_ACK.** Although response is **B_ACK,** it does not ensure that the associated flag in **"Clear_Byte"** is cleared. NFC/RF device needs to read the register 0x00 to check the current status of the flags again or check **ACK or B_NAK** in the next response frame. It is possible that the failed situation is still present. For example, input power is still weak. Then, **RSPW_LOW** is still flagged, although the command ***Clear_Flag*** is sent.

Table 8-12: ***Clear_Flag*** command format

CMD	***Clear_Flag***		
Format	**0xB4 + Clear_Byte + CRC**		
Response	**2B_FLAG = 1**	Successful operation	**B_ACK + B_ACK + CRC**
		Operation error	-
		CRC error, RF error, Framing error	**NAK** (4 bits)
	2B_FLAG = 0	Successful operation	**B_ACK + CRC**
		Operation Error	-
		CRC error, RF error, Framing error	**NAK** (4 bits)
Clear_Byte	Clear_Byte.bit0 : Clear **"DL_FF_OVF"** flag Clear_Byte.bit1 : RFU Clear_Byte.bit2 : Clear **"UL_FF_OVF"** flag Clear_Byte.bit3 : Clear **"RSPW_LOW"** flag Clear_Byte.bit4 : Clear **"XVDD_DROP"** flag Clear_Byte.bit5 : Clear **"UART_FAIL"** flag *Example : Clear_Byte = 0x05 :* **DL_FF_OVF** *and* **UL_FF_OVF**		

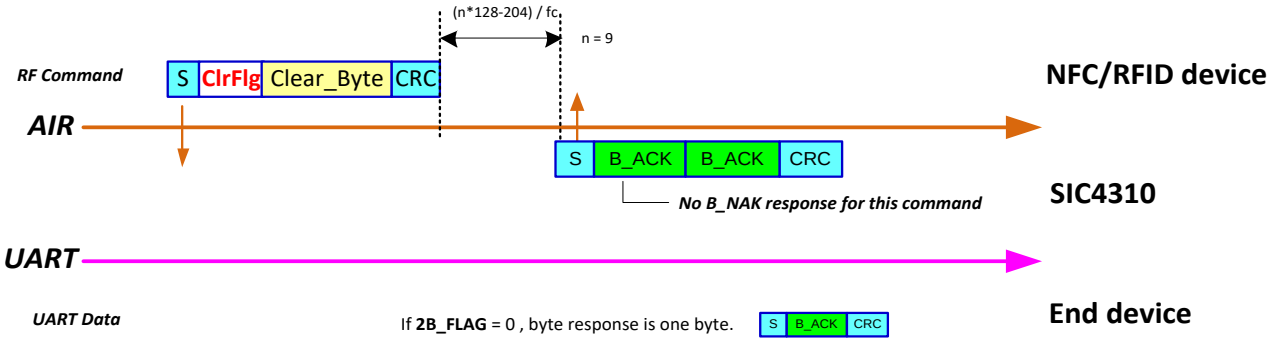


Figure 8-23: ***Clear_Flag*** command frame

8.3 RF-Reg commands

8.3.1 ReadReg

The purpose of ***ReadReg*** command is to read register value. The response frame consists of **B_ACK** or **B_NAK** with the current accessed register value. If the reading address is out of range, data in response package is 0x00. If the last clearing is not complete or power is insufficient until the indicator flags again, the response contains with a **B_NAK**.

Table 8-13: ***ReadReg*** command format

CMD	***ReadReg***		
Format	**0xB5 + Reg_Addr + CRC**		
Response	**2B_FLAG = 1** or **2B_FLAG = 0**	Successful operation	**B_ACK + Data + CRC**
		Operation error	**B_NAK + Data + CRC** when any power indicators are flagged
		CRC error, RF error, Framing error	**NAK** (4 bits)
Operation	Read Data from register. If the address is out of range, response frame consists of B_ACK with data 0x00		

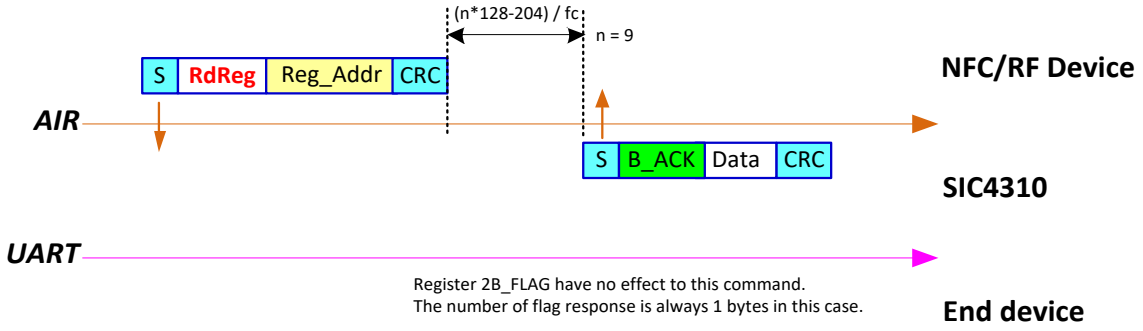


Figure 8-24: ***ReadReg*** command frame with a positive acknowledge response

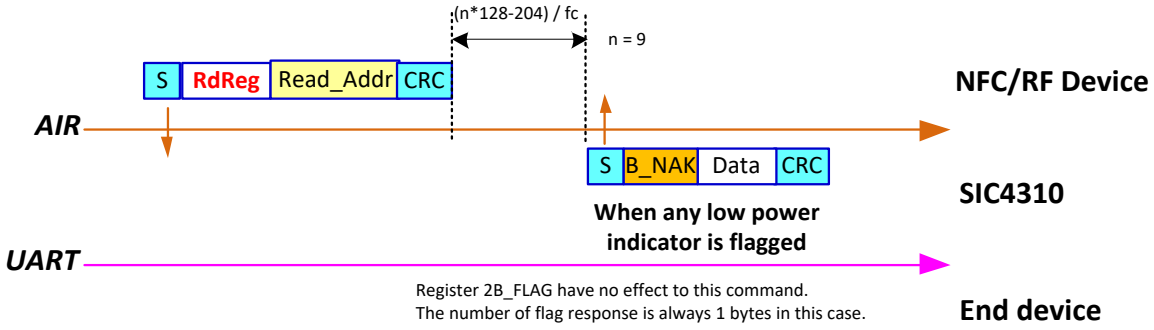


Figure 8-25: ***ReadReg*** command frame with a negative acknowledge response

8.3.2 WriteReg

The purpose of WriteReg command is to write a register value. The response frame can be B_ACK or B_NAK depending on the input power level. If the address is out of range or address is read-only, response frame is still B_ACK. If the last clearing is not complete or power is insufficient until the indicator flags again, the response contains with a B_NAK.

Table 8-14: ***WriteReg*** command format

CMD	***WriteReg***		
Format	**0xB6 + Reg_Addr + Data + CRC**		
Response	**2B_FLAG = 1**	Successful operation	**B_ACK + B_ACK + CRC**
		Operation error	**B_NAK + B_NAK + CRC** when any power indicators are flagged
		CRC error, RF error, Framing error	**NAK** (4 bits)
	2B_FLAG = 0	Successful operation	**B_ACK + CRC**
		Operation error	**B_NAK + CRC** when any power indicators are flagged
		CRC error, RF error, Framing error	**NAK** (4 bits)
Operation	Write data to register. If the address is out of range or address is read-only, response frame is still B_ACK.		

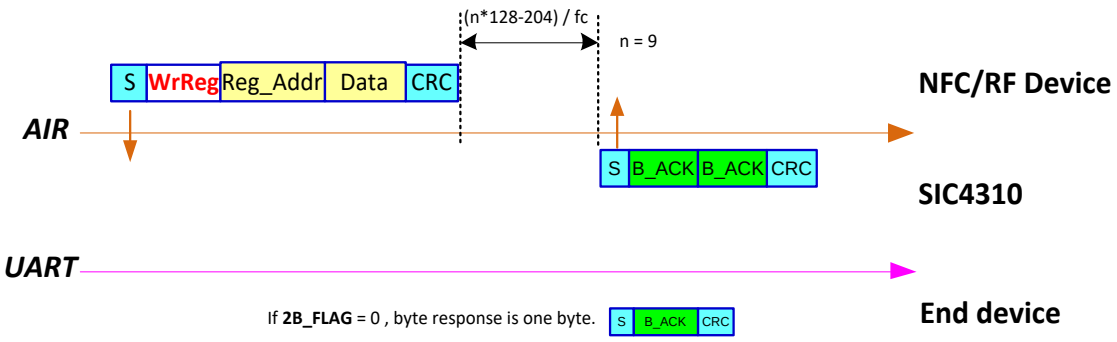


Figure 8-26: ***WriteReg*** command frame with an 8-bit ACK response

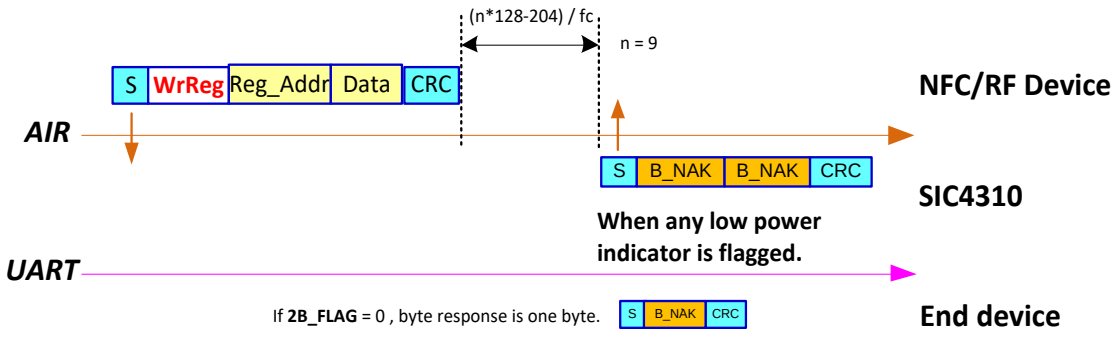


Figure 8-27: ***WriteReg*** command frame with an 8-bit NAK response

8.4 UART commands

This group of commands is used to access the EEPROM from the UART side. This command can be activated when the pin **UMAS** is raised to '1'. The external component should send a command to the pin **GPIO[2]** functioning as *UART-RX* and receive a response from the pin **GPIO[1]** functioning as *UART-TX.*

To prevent UART RX system stuck from remaining data in the received FIFO (Uplink FIFO), UART-connected devices should control time interval between each byte in command packet to be within 1 second. Otherwise, the received command will be cancelled and UART RX system waits for next coming header (0xAA).

8.4.1 UART_Write_E2

The purpose of command ***UART_Write_E2*** is to write data into the EEPROM from an external UART-connected device. The command contains the data to be written with an XOR byte appended in the end byte. There is no response from this command. UART-connected device can verify by re-reading the written address.

Table 8-15: ***UART_Write_E2*** command format

CMD	***UART_Write_E2***
Format	0xAA + 0xA2 + ADR + Data[0] + Data[1] + Data[2] + Data[3] + XOR
Response	No response
Remark	The first byte 0xAA is treated to be a header. XOR is calculated from A2 to Data[3]

8.4.2 UART_Read_E2

The purpose of command ***UART_Read_E2***is to read the EEPROM content. The command contains an address with an XOR byte appended in the end byte. The response is 4 bytes of addressed page.

Table 8-16: ***UART_Read_E2*** command format

CMD	***UART_Read_E2***
Format	0xAA + 0x30 + ADR + XOR
Response	Data[0] + Data[1] + Data[2] + Data[3]
Remark	The first byte 0xAA is treated to be a header. XOR is calculated from 0x30 to ADR Response is only plain data without the frame format.

8.5 Response Acknowledge

During communication, the SIC4310 uses two kinds of acknowledge to respond to NFC/ RFID devices. The first type of acknowledge is a 4-bit type compliant to the NFC Tag Type 2 standard and the second type of acknowledge is an 8-bit type used in the RF-to-UART communication and the register page access. If the SIC4310 answers with a 4-bit NAK, the RF state goes back to ***"Idle"*** or ***"Halt".*** If the response is an 8-bit NAK, the RF state still remains in ***"Active".*** The 8-bit acknowledge contains status flags, resulting from event capturing in the UART communication and power related level.

The number of bytes of 8-bit flag can be set to send either 1 byte or 2 bytes with a repetitive value by setting the register **2B_FLAG.** The purpose of **2B_FLAG** is to set a response flag in a response package to two bytes for some downlink commands to guarantee that the uplink response frame contains at least 2 bytes to make the operation compatible with some NFC phone in market that don't accept one-byte responses. The detail of 4-bit and 8-bit flag responses are summarized in Table 8-17 and Table 8-18 respectively.

Table 8-17: 4-bits ACK/NAK

Response Flag	Code	Description
ACK	1010b	Positive acknowledge indicate operation is successful.
NAK	0000b	Negative acknowledge indicate accessing address is out of range or accessed block is locked.
	0001b	Negative acknowledge indicate parity or CRC is error, or data in write command is less than 4 bytes.

Table 8-18: 8-bits ACK/NAK

Response Flag	Code	Description		
B_ACK	0x1A	Positive acknowledge indicate operation is successful.		
B_NAK	0xYY	**Bit**	**Error flag**	**Command can cause error**
		Bit 0	**DL_FF_OVF**	TxRU, TRxRU
		Bit 1	**UL_FF_EMP**	RxUR, TRxUR
		Bit 2	**UL_FF_OVF**	RxUR, TRxUR
		Bit 3	**RSPW_LOW**	TxRU, RxUR, TRxRU, RdReg, WrReg
		Bit 4	**XVDD_DROP**	TxRU, RxUR, TRxRU, RdReg, WrReg
		Bit 5	**UART_FAIL**	TxRU, RxUR, TRxRU, RdReg, WrReg
		Bit 6	-	-
		Bit 7	Always set to logic '1'	

Table 8-19 shows meaning of error flag in B_NAK and its trigger event. When any flags except **"UL_FF_ EMP"** are set, command ***Clear_Flag*** can reset these bits to '0'. Note that each error can individually be flagged depending on a situation. The purpose of the flags is to record events related to power and FIFO status to show the reliability of the RF data transmission.

Table 8-19: Meaning of error flag in B_NAK

Error flag	Type	Description	Trigger Event
DL_FF_OVF	Flag	Downlink FIFO Overflow	Downlink FIFO overflow occurs.
UL_FF_EMP	Status	Uplink FIFO Empty	-
UL_FF_OVF	Flag	Uplink FIFO Overflow	Uplink FIFO overflow occurs.
RSPW_LOW	Flag	Insufficient power to source load	**RSPW_RDY** status becomes '0'
XVDD_DROP	Flag	Voltage on **XVDD** drop	**XVDD_RDY** status become '0'
UART_FAIL	Flag	UART is fail	**UART_RDY** status become '0'

When a device is used in the power harvesting mode, SIC4310 monitors the power system and displays through three power status bits in the register page as shown in the left column of Table 8-20. If any status bits go to a failed state, the inverted value of such fail status is stored in the **B_NAK** flag as shown in the right column of Table 8-20.

Table 8-20: Power Status and Power Flag

Power Status	Power Error Flag in "BNAK"
RSPW_RDY (Reg : 0x01.1)	**"RSPW_LOW"** (BNAK: Bit 3)
XVDD_RDY (Reg : 0x01.2)	**"XVDD_DROP"** (BNAK: Bit 4)
UART_RDY (Reg : 0x01.3)	**"UART_FAIL"** (BNAK: Bit 5)

When the **"RSPW_LOW"** or **"XVDD_DROP"** flag is set, the SIC4310 responds **B_NAK** with data resulted from operation as if it is the frame contains **B_ACK** from normal operation. The purpose of **"RSPW_LOW"** is only to indicate that the received power from RF is insufficient and the purpose of **"XVDD_DROP"** is to inform that LDO cannot supply a load on the pin **XVDD** and voltage on the pin **XVDD** is drop below 2.7 volt. The **"XVDD_DROP"** can also indicate a fault event on the pin **XVDD**.

When an NFC/RF device receives this flag, the design shall be optimized to use power within a limit of pre-qualified available power, or the associated software application shall inform the user to place the device closer to the SIC4310's antenna to receive more power. The NFC/RF device can check if the power status from registers **RSPW_RDY** and **XVDD_RDY** is back to a normal state. If a status bit becomes active high, then the NFC/RF device can successfully clear the associated flag by the command ***Clear_Flag.***

Similar to the **"RSPW_LOW"** or **"XVDD_DROP"** flag, the operation of RF-to-UART command is also not affected by the **RSPW_RDY** and **XVDD_RDY** status. Transaction in the event that the RF input power drops and voltage on the pin **XVDD** drops are shown Figure 8-28 and Figure 8-29.

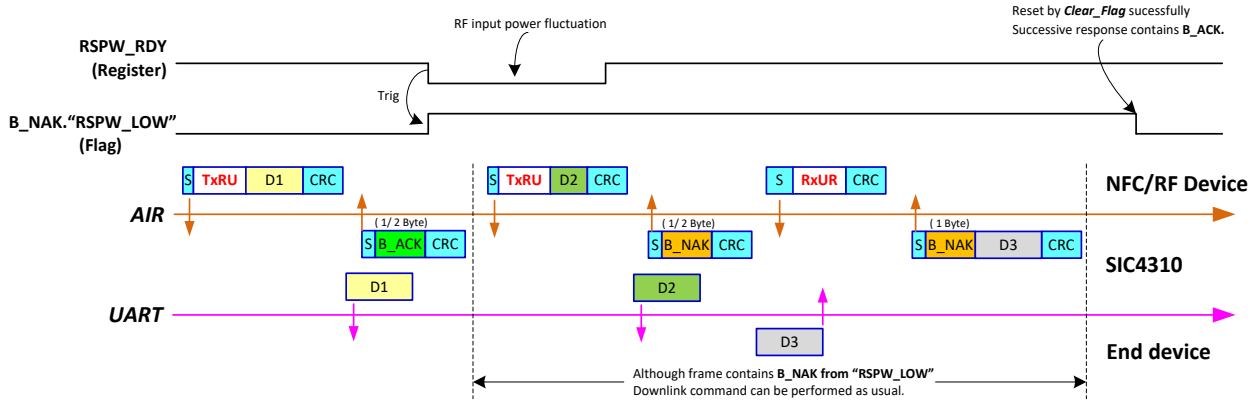


Figure 8-28: Example of RF transaction when the "**RSPW_LOW**" flag is set

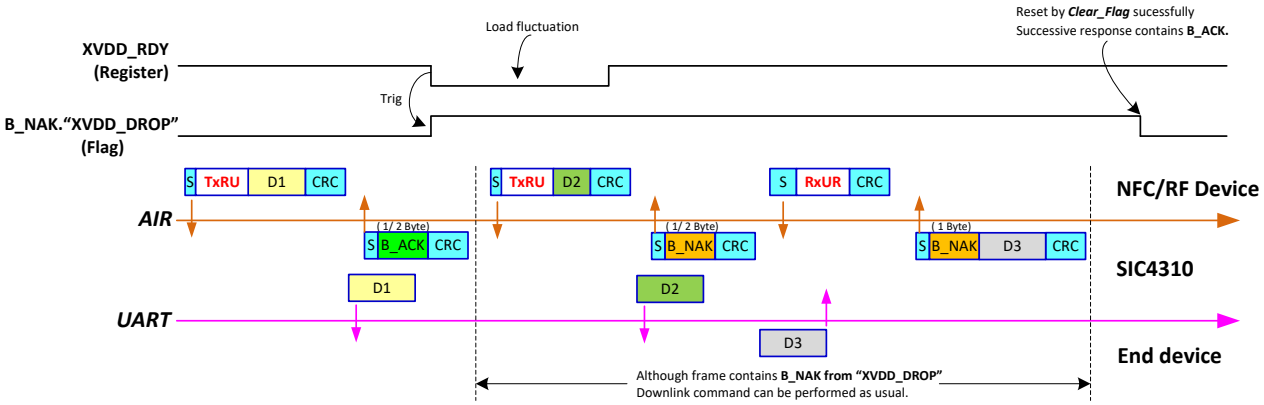


Figure 8-29: Example of RF transaction when the "**XVDD_DROP**" flag is set

The **"UART_FAIL"** indicates that the oscillator becomes unstable for a while during the UART communication. Then, the UART transaction may be unreliable and incorrect. If the **UART_RDY** status is still '0', the SIC4310 cannot operate any RF-to-UART or RF-Reg commands. In this case, the response contains only **B_NAK** without data. If the **UART_RDY** status is back to '1', the SIC4310 can operate RF-to-UART downlink commands and responses contain **B_NAK** with data**. B_NAK** is still in all successive responses until **UART_FAIL** is successfully cleared. Figure 8-30 depicts the transaction when the **UART_FAIL** flag is set. Table 8-21 shows corrective actions for each error flag during both design phase and real usage situation.

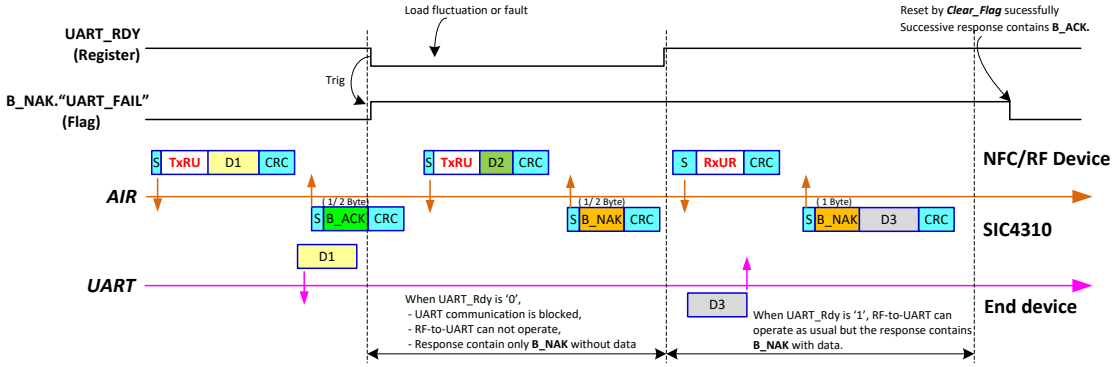
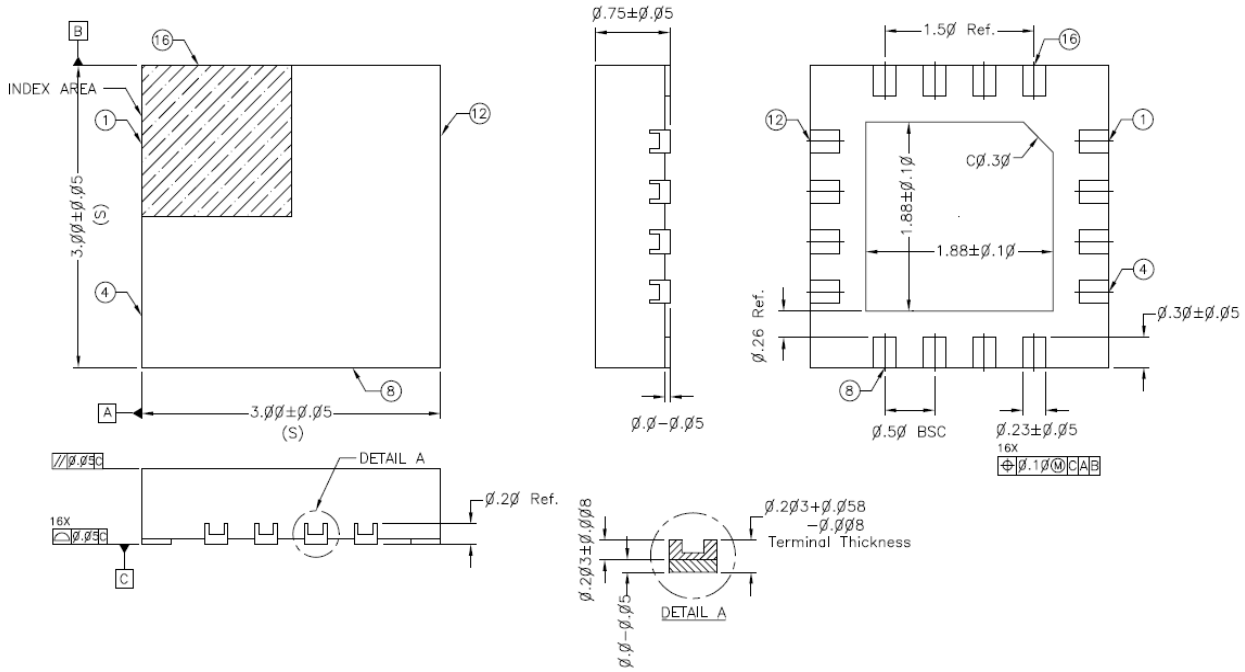


Figure 8-30: Example of RF transaction when "**UART_FAIL**" flag is set

Table 8-21: B_NAK and corrective action

Flag	Corrective Action
DL_FF_OVF	**During design** – Prolong time between each downlink frame to match throughput of UART. – Increase baud rate of UART. **In usage situation** – Send ***Clear_Flag*** – Reduce amount of downlink data – Resend the packet again.
UL_FF_EMP	**In usage situation** – Inform UART end device to respond. – No need to send ***Clear_Flag***
UL_FF_OVF	**During design** – Use handshaking in the implementation. – Decrease baud rate of UART. **In usage situation** – Send ***Clear_Flag*** – Retrieve previous data packets from the UART end device.
RSPW_LOW	**During design** – Optimize a size of antenna and power of NFC/RFID device **In usage situation** – Inform users through the application to move the NFC/RFID device closer to SIC4310's antenna, if possible. – Send ***Clear_Flag***
XVDD_DROP	**During design** – Limit load on the pin **XVDD**. – Check fault on the pin **XVDD**. **In usage situation** – Inform uses through the application to move the NFC/RFID device closer to SIC4310's antenna, if possible. – Check status of **LDO_ON, LDO_ON** must be '1' in case of power harvesting. – Send ***Clear_Flag***
UART_FAIL	**During design** – Check fault on the pin **XVDD**. **In usage situation** – Inform users through the application to move the NFC/RFID device closer to SIC4310's antenna, if possible. – Check status of **UART_RDY**; it must be '1' – Check status of **OSC_EN, OSC_EN** must be '1' – Check status of **LDO_ON, LDO_ON** must be '1' in case of power harvesting. – Send ***Clear_Flag*** to make further transaction operate

9. Packaging and Dimension



NOTE : CONTROL DIMENSION IN MM.

Figure 9-1: QFN3x3-16pin package dimension

10. Disclaimer

- The information described herein is subject to change without notice.
- Although the IC contains a static electricity protection circuit, static electricity or voltage that exceeds the limit of the protection circuit should not be applied.
- Silicon Craft Technology assumes no responsibility for the way in which this IC is used in products created using this IC or for the specifications of that product, nor does Silicon Craft Technology assume any responsibility for any infringement of patents or copyrights by products that include this IC either in Thailand or in other countries.
- Silicon Craft Technology is not responsible for any problems caused by circuits or diagrams described herein whose related industrial properties, patents, or other rights belong to third parties. The application circuit examples explain typical applications of the products, and do not guarantee the success of any specific mass-production design.
- Use of the information described herein for other purposes and/or reproduction or copying without the express permission of Silicon Craft Technology is strictly prohibited.
- The products described herein cannot be used as part of any device or equipment affecting the human body, such as exercise equipment, medical equipment, security systems, gas equipment, or any apparatus installed in airplanes and other vehicles, without prior written permission of Silicon Craft Technology.
- Although Silicon Craft Technology exerts the greatest possible effort to ensure high quality and reliability, the failure or malfunction of semiconductor products may occur. The user of these products should therefore give thorough consideration to safety design, including redundancy, fire-prevention measures, and malfunction prevention, to prevent any accidents, fires, or community damage that may ensue.