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

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

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

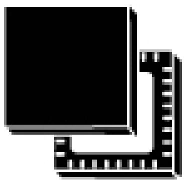
Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

CR95HF

13.56-MHz multi-protocol contactless transceiver IC with SPI and UART serial access

Datasheet - production data



VFQFPN32 5x5 mm

Features

- CR95HF belongs to the ST25 family which includes all ST's NFC/RFID tag and reader products
- Operating modes supported:
  - Reader/Writer
- Hardware features
  - Dedicated internal frame controller
  - Highly integrated Analog Front End (AFE) for RF communications
  - Transmission and reception modes
  - Optimized power management
  - Tag Detection mode
- RF communication @13.56 MHz
  - ISO/IEC 14443 Type A and B
  - ISO/IEC 15693
  - ISO/IEC 18092
  - MIFARE® Classic compatible [(a)] [(b)]
- Communication interfaces with a Host Controller
  - Serial peripheral interface (SPI) Slave interface
  - Universal asynchronous receiver/transmitter (UART)
  - Up to 528-byte command/reception buffer (FIFO)
- 32-lead, 5x5 mm, very thin fine pitch quad flat (VFQFPN) ECOPACK®2 package

Applications

Typical protocols supported:

- ISO/IEC 14443-3 Type A and B tags
- ISO/IEC 15693 tags
- ISO/IEC 18000-3M1 tags
- NFC Forum tags: Types 1, 2, 3 and 4
- ST short-range interface (SRI) tags
- ST long-range interface (LRI) tags
- ST Dual Interface EEPROM

---

a. MIFARE and MIFARE Classic are registered trademarks of NXP B.V. and are used under license.

b. Parity Framing mode is compatible with MIFARE® Classic requirements. However, access to Authenticated state must be supported by an external secure host which embeds the MIFARE® Classic library.

Contents

1 Description . . . 7

1.1 Block diagram . . . 7

1.2 List of terms . . . 8

2 Pin and signal descriptions . . . 9

3 Power management and operating modes . . . 11

3.1 Operating modes . . . 11

3.2 Startup sequence . . . 12

4 Communication protocols . . . 14

4.1 Universal asynchronous receiver/transmitter (UART) . . . 14

4.2 Serial peripheral interface (SPI) . . . 15

4.2.1 Polling mode . . . 15

4.2.2 Interrupt mode . . . 17

4.3 Error codes . . . 17

4.4 Support of long frames . . . 18

5 Commands . . . 19

5.1 Command format . . . 19

5.2 List of commands . . . 19

5.3 IDN command (0x01) description . . . 20

5.4 Protocol Select command (0x02) description . . . 20

5.5 Send Receive (SendRecv) command (0x04) description . . . 25

5.6 Idle command (0x07) description . . . 30

5.6.1 Idle command parameters . . . 32

5.6.2 Using LFO frequency setting to reduce power consumption . . . 33

5.6.3 Optimizing wake-up conditions . . . 34

5.6.4 Using various techniques to return to Ready state . . . 34

5.6.5 Tag detection calibration procedure . . . 36

5.7 Read Register (RdReg) command (0x08) description . . . 37

5.8 Write Register (WrReg) command (0x09) description . . . 38

5.8.1 Improving RF performance . . . 38

5.8.2 Improving frame reception for ISO/IEC 14443 Type A tags 40
5.8.3 Improving RF reception for ISO/IEC 18092 tags 40
5.9 BaudRate command (0x0A) description 41
5.10 Echo command (0x55) description 41

6 Electrical characteristics 42
6.1 Absolute maximum ratings 42
6.2 DC characteristics 43
6.3 Power consumption characteristics 43
6.4 SPI characteristics 45
6.5 RF characteristics 47
6.6 Oscillator characteristics 48

7 Package information 49
7.1 VFQFPN32 package information 49

8 Part numbering 52

Appendix A Additional Idle command description 53

Appendix B Example of tag detection calibration process 54

Appendix C Example of tag detection command using results of tag detection calibration 57

Appendix D Examples of CR95HF command code to activate NFC Forum and ISO/IEC 15693 tags 58
D.1 ISO/IEC 14443 Type A 58
D.1.1 NFC Forum Tag Type 1 (Topaz) 58
D.1.2 NFC Forum Tag Type 2 60
D.1.3 NFC Forum Tag Type 2 or 4: Using split frames to resolve collisions . . 61
D.1.4 NFC Forum Tag Type 2 64
D.1.5 NFC Forum Tag Type 4A 67
D.2 ISO/IEC 14443 Type B 70
D.2.1 NFC Forum Tag Type 4B 70
D.3 ISO/IEC 18092 72
D.3.1 NFC Forum Tag Type 3 72

D.4 ISO/IEC 15693 73

D.4.1 ISO/IEC 15693 tag 73

Revision history 75

List of tables

Table 1. List of terms . . . 8
Table 2. CR95HF pin descriptions . . . 9
Table 3. CR95HF operating modes and states . . . 11
Table 4. Select serial communication interface selection table . . . 13
Table 5. Interpretation of flags . . . 16
Table 6. Possible error codes and their meaning . . . 17
Table 7. Format of ResultCode . . . 18
Table 8. Examples of ResultCode: Len pairs . . . 18
Table 9. List of CR95HF commands . . . 19
Table 10. IDN command description . . . 20
Table 11. ProtocolSelect command description . . . 20
Table 12. List of `<Parameters>` values for the ProtocolSelect command for different protocols . . . 21
Table 13. SendRecv command description . . . 25
Table 14. List of `<Data>` Send values for the SendRecv command for different protocols . . . 26
Table 15. List of `<Data>` Response values for the SendRecv command for different protocols . . . 28
Table 16. Structure of Parity byte . . . 30
Table 17. Idle command description . . . 31
Table 18. Idle command structure . . . 32
Table 19. Summary of Idle command parameters . . . 32
Table 20. RdReg command description . . . 37
Table 21. WrReg command description (Modulation Index and Receiver Gain) . . . 38
Table 22. Possible Modulation Index values . . . 39
Table 23. Possible Receiver Gain values . . . 39
Table 24. ARC_B default code for available Reader protocols . . . 40
Table 25. WrReg command description (Timer Window) . . . 40
Table 26. BaudRate command description . . . 41
Table 27. Echo command description . . . 41
Table 28. Absolute maximum ratings . . . 42
Table 29. DC characteristics . . . 43
Table 30. Power consumption characteristics (VPS_Main from 2.7 to 3.3 V) . . . 43
Table 31. Power consumption characteristics (VPS_TX from 2.7 to 3.3 V) . . . 43
Table 32. Power consumption characteristics (VPS_TX from 4.5 to 5.5 V) . . . 44
Table 33. SPI interface characteristics . . . 45
Table 34. Reader characteristics . . . 47
Table 35. HFO 27.12 MHz oscillator characteristics . . . 48
Table 36. VFQFPN32 - 32-pin, 5x5 mm, 0.5 mm pitch very thin profile fine pitch quad flat package mechanical data . . . 50
Table 37. Ordering information scheme . . . 52
Table 38. Wake-up source register . . . 53
Table 39. Wake-up event register . . . 53
Table 40. Document revision history . . . 75

List of figures

Figure 1. CR95HF application overview 7
Figure 2. CR95HF block diagram 7
Figure 3. CR95HF pinout description. 9
Figure 4. CR95HF initialization and operating state change 12
Figure 5. Power-up sequence 12
Figure 6. UART communication. 14
Figure 7. Echo command and response example 14
Figure 8. Sending command to CR95HF 15
Figure 9. Polling the CR95HF until it is ready 15
Figure 10. Reading data from CR95HF 16
Figure 11. Reset the CR95HF 16
Figure 12. Long frame format 18
Figure 13. Data transfer (in both command and response) when Parity Framing mode is enabled . . 28
Figure 14. SPI timing diagram (Slave mode and CPOL = 0, CPHA = 0) 45
Figure 15. SPI timing diagram (Slave mode and CPOL = 1, CPHA = 1) 46
Figure 16. Typical application with a 27.12 MHz crystal 48
Figure 17. VFQFPN32 - 32-pin, 5x5 mm, 0.5 mm pitch very thin profile fine pitch quad flat package outline 49
Figure 18. VFQFPN32 - 32-pin, 5x5 mm, 0.5 mm pitch very thin profile fine pitch quad flat package recommended footprint 51

1 Description

The CR95HF is an integrated transceiver IC for contactless applications.

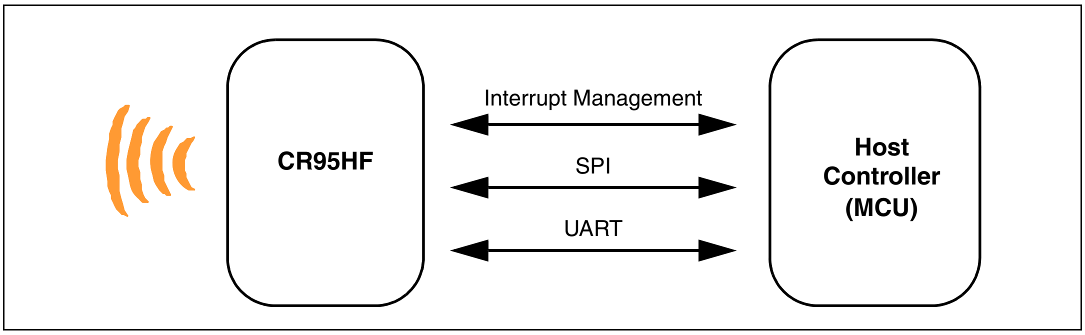
The CR95HF manages frame coding and decoding in Reader mode for standard applications such as near field communication (NFC), proximity and vicinity standards.

The CR95HF embeds an Analog Front End to provide the 13.56 MHz Air Interface.

The CR95HF supports ISO/IEC 14443 Type A and B, ISO/IEC 15693 (single or double subcarrier) and ISO/IEC 18092 communication protocols.

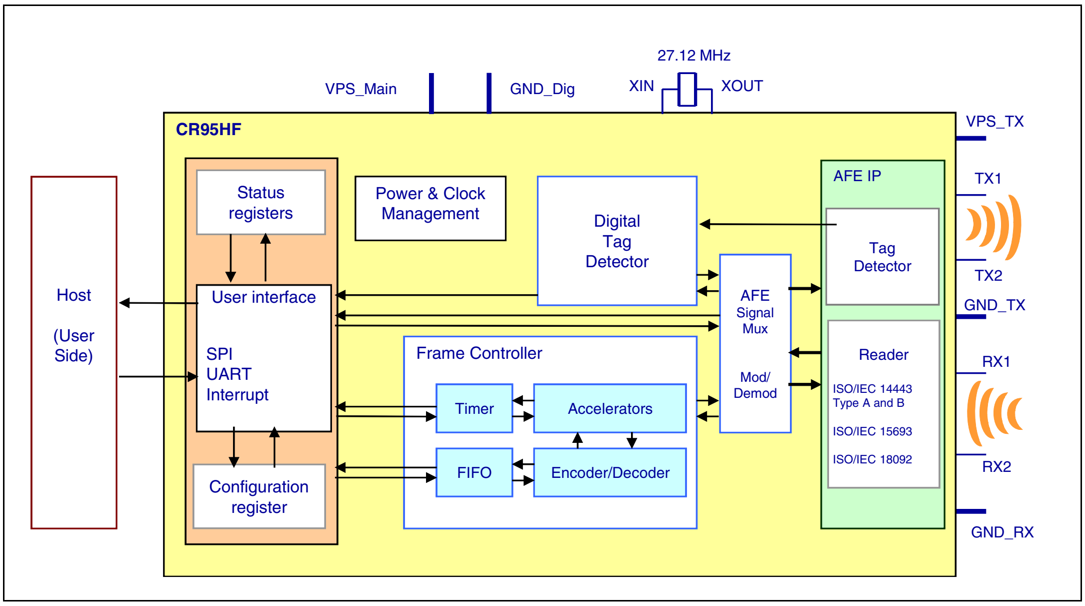
The CR95HF also supports the detection, reading and writing of NFC Forum Type 1, 2, 3 and 4 tags.

Figure 1. CR95HF application overview



1.1 Block diagram

Figure 2. CR95HF block diagram



1.2 List of terms

Table 1. List of terms

Term	Meaning
DAC	Digital analog converter
GND	Ground
HFO	High frequency oscillator
LFO	Low frequency oscillator
MCU	Microcontroller unit
NFC	Near Field Communication
RFID	Radio Frequency Identification
RFU	Reserved for future use
SPI	Serial peripheral interface
t_L	Low frequency period
t_{REF}	Reference time
UART	Universal asynchronous receiver-transmitter
WFE	Wait For Event

2 Pin and signal descriptions

Figure 3. CR95HF pinout description

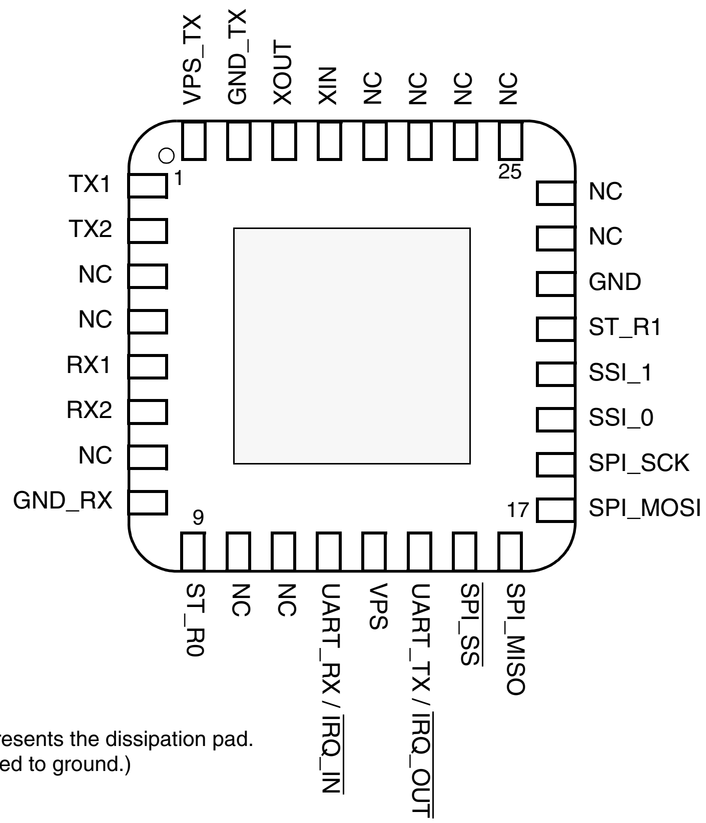


Table 2. CR95HF pin descriptions

Pin	Pin name	Type(1)	Main function	Alternate function
1	TX1	O	Driver output 1	
2	TX2	O	Driver output 2	
3	NC		Not connected	
4	NC		Not connected	
5	RX1	I	Receiver input 1	
6	RX2	I	Receiver input 2	
7	NC		Not connected	
8	GND_RX	P	Ground (analog)	
9	ST_R0	O	ST Reserved(2)	
10	NC		Not connected	
11	NC		Not connected	

Table 2. CR95HF pin descriptions (continued)

Pin	Pin name	Type[1]	Main function	Alternate function
12	UART_RX / IRQ_IN	I [3]	UART receive pin [4]	Interrupt input
13	VPS	P	Main power supply	
14	UART_TX / IRQ_OUT	O [5]	UART transmit pin	Interrupt output
15	SPI_SS	I [6]	SPI Slave Select (active low)	
16	SPI_MISO	O [6]	SPI Data, Slave Output	
17	SPI_MOSI	I [6]	SPI Data, Slave Input [6]	
18	SPI_SCK	I [7]	SPI serial clock	
19	SSI_0	I [6]	Select serial communication interface	
20	SSI_1	I [6]	Select serial communication interface	
21	ST_R1	I [8]	ST Reserved	
22	GND	P	Ground (digital)	
23	NC		Not connected	
24	NC		Not connected	
25	NC		Not connected	
26	NC		Not connected	
27	NC		Not connected	
28	NC		Not connected	
29	XIN		Crystal oscillator input	
30	XOUT		Crystal oscillator output	
31	GND_TX	P	Ground (RF drivers)	
32	VPS_TX	P	Power supply (RF drivers)	

1. I: Input, O: Output, and P: Power
2. Must add a capacitor to ground (~1 nF).
3. Pad internally connected to a Very Weak Pull-up to VPS.
4. We recommend connecting this pin to the V_{PS} pin using a 3.3 kOhm pull-up resistor.
5. Pad internally connected to a Weak Pull-up to VPS.
6. Must not be left floating.
7. Pad internally connected to a Weak Pull-down to GND.
8. Pad input in High Impedance. Must be connected to VPS.

3 Power management and operating modes

3.1 Operating modes

The CR95HF has 2 operating modes: Wait for Event (WFE) and Active. In Active mode, the CR95HF communicates actively with a tag or an external host (an MCU, for example). WFE mode includes four low consumption states: Power-up, Hibernate, Sleep and Tag Detector.

The CR95HF can switch from one mode to another.

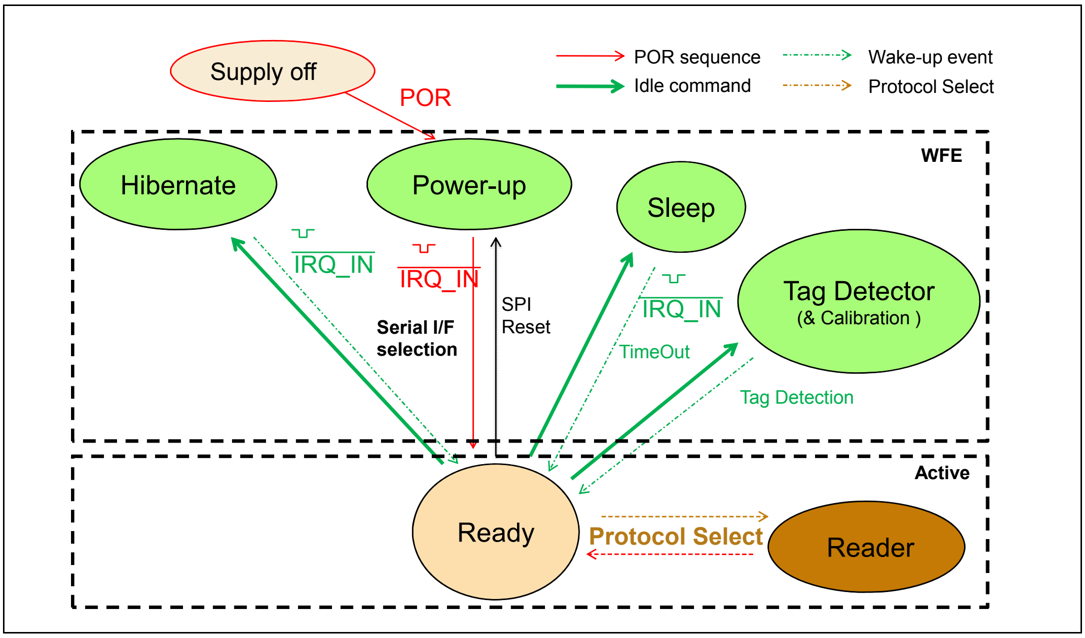
Table 3. CR95HF operating modes and states

Mode	State	Description
Wait For Event (WFE)	Power-up	This mode is accessible directly after POR. Low level on IRQ_IN pin (longer than 10 µs) is the only wakeup source. LFO (low-frequency oscillator) is running in this state.
	Hibernate	Lowest power consumption state. The CR95HF has to be woken-up in order to communicate. Low level on IRQ_IN pin (longer than 10 µs) is the only wakeup source.
	Sleep	Low power consumption state. Wakeup source is configurable: – Timer – IRQ_IN pin – SPI_SS pin LFO (low-frequency oscillator) is running in this state.
	Tag Detector	Low power consumption state with tag detection. Wakeup source is configurable: – Timer – IRQ_IN pin – SPI_SS pin – Tag detector LFO (low-frequency oscillator) is running in this state.
Active	Ready	In this mode, the RF is OFF and the CR95HF waits for a command (ProtocolSelect, ...) from the external host via the selected serial interface (UART or SPI).
	Reader	The CR95HF can communicate with a tag using the selected protocol or with an external host using the selected serial interface (UART or SPI).

Hibernate, Tag Detector, and Sleep states can only be activated by a command from the external host. As soon as any of these three states are activated, the CR95HF can no longer communicate with the external host. It can only be woken up.

The behavior of the CR95HF in 'Tag Detector' state is defined by the Idle command.

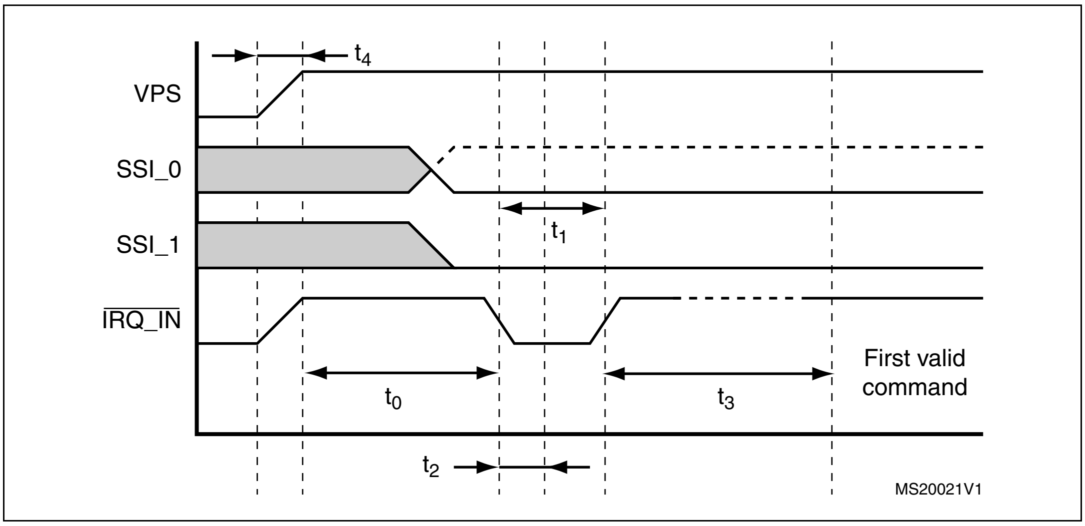
Figure 4. CR95HF initialization and operating state change



3.2 Startup sequence

After the power supply is established at power-on, the CR95HF waits for a low pulse on the pin IRQ_IN (t_1) before automatically selecting the external interface (SPI or UART) and entering Ready state after a delay (t_3).

Figure 5. Power-up sequence



1. Note for pin SSI0: - - - SPI selected, ——— UART selected
2. Pin IRQ_IN low level < 0.2 VPS_Main.

Note: When CR95HF leaves WFE mode (from Power-up, Hibernate, Tag Detector, or Sleep) following an IRQ_IN/RX low level pulse, this pulse is NOT interpreted as the UART start bit character.

Figure 5 shows the power-up sequence for a CR95HF device; where,

- t_0 is the initial wake-up delay 100 µs (minimum)
- t_1 is the minimum interrupt width 10 µs (minimum)
- t_2 is the delay for the serial interface selection 250 ns (typical)
- t_3 is the HFO setup time ($t_{SU(HFO)}$) 10 ms (maximum)
- t_4 is the V_{PS} ramp-up time from 0V to V_{PS} 10 ms (max. by design validation)

Note: VPS must be 0V before executing the start-up sequence.

The serial interface is selected after the following falling edge of pin $\overline{IRQ\_IN}$ when leaving from POR or Hibernate state.

Table 4 lists the signal configuration used to select the serial communication interface.

Table 4. Select serial communication interface selection table

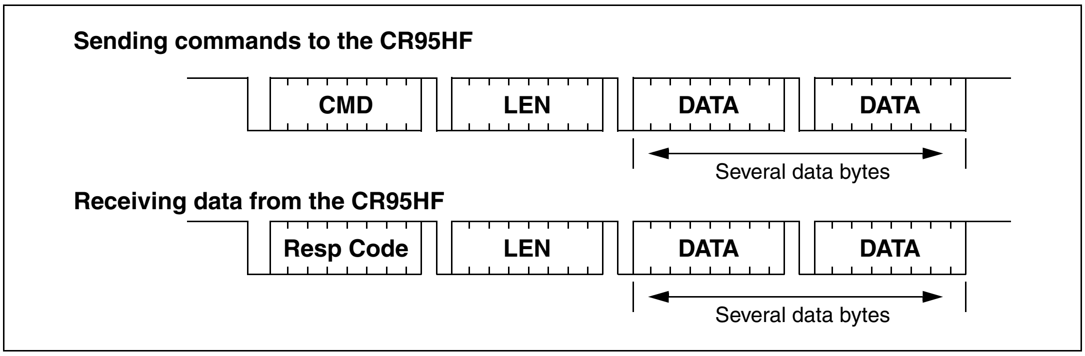
Pin	Serial interface	
SSI_0	UART: 0	SPI: 1
SSI_1	UART: 0	SPI: 0

4 Communication protocols

4.1 Universal asynchronous receiver/transmitter (UART)

The host sends commands to the CR95HF and waits for replies. Polling for readiness is not necessary. The default baud rate is 57600 baud. The maximum allowed baud rate is 2 Mbps.

Figure 6. UART communication



When sending commands, no data must be sent if the LEN field is zero.

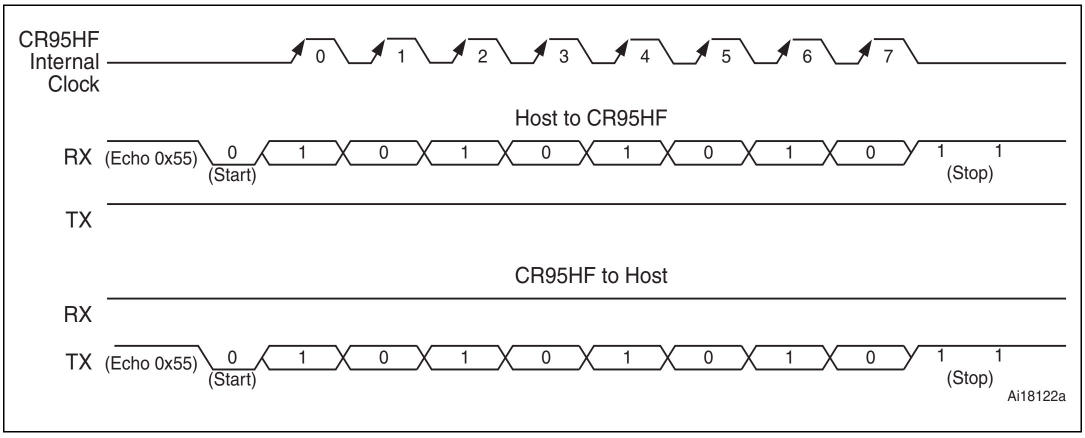
When receiving data from the CR95HF, no data will be received if the LEN field is zero.

The formats of send and receive packets are identical.

If an ECHO command is sent, only one byte (0x55) is sent by the host.

Figure 7 shows an example of an ECHO command.

Figure 7. ECHO command and response example



Caution: UART communication is LSB first. Stop bit duration is two Elementary Time Units (ETUs).

Note: 1 When CR95HF leaves WFE mode (from Power-up, Hibernate, Sleep Detector or Tag Detector) following an IRQ_IN/RX low level pulse, this pulse is NOT interpreted as the UART start bit character.

2 If the user loses UART synchronization, it can be recovered by sending an ECHO command until a valid ECHO reply is received. Otherwise, after a maximum of 528 ECHO commands, *CR95HF will reply with an error code meaning its input buffer is full. The user can now restart a UART exchange.*

4.2 Serial peripheral interface (SPI)

4.2.1 Polling mode

In order to send commands and receive replies, the application software has to perform 3 steps.

1. Send the command to the CR95HF.
2. Poll the CR95HF until it is ready to transmit the response.
3. Read the response.

The application software should never read data from the CR95HF without being sure that the CR95HF is ready to send the response.

The maximum allowed SPI communication speed is f_{SCK}.

A Control byte is used to specify a communication type and direction:

- 0x00: Send command to the CR95HF
- 0x03: Poll the CR95HF
- 0x02: Read data from the CR95HF
- 0x01: Reset the CR95HF

The $\overline{\text{SPI\_SS}}$ line is used to select a device on the common SPI bus. The $\overline{\text{SPI\_SS}}$ pin is active low.

When the $\overline{\text{SPI\_SS}}$ line is inactive, all data sent by the Master device is ignored and the MISO line remains in High Impedance state.

Figure 8. Sending command to CR95HF

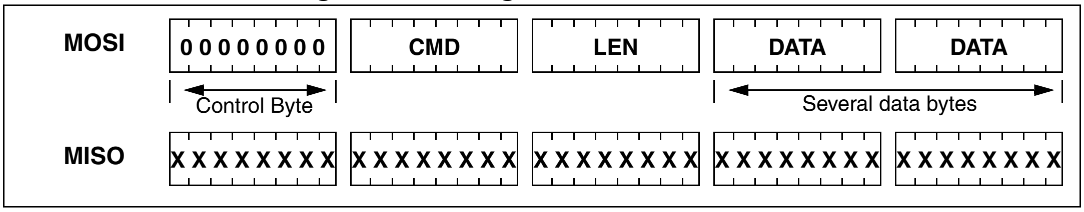


Figure 9. Polling the CR95HF until it is ready

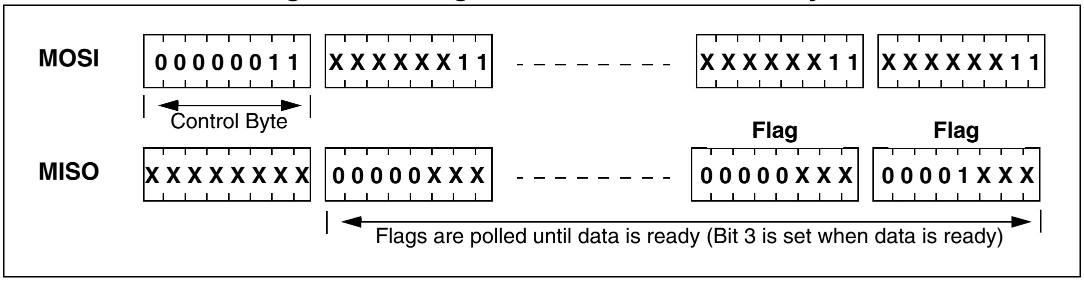
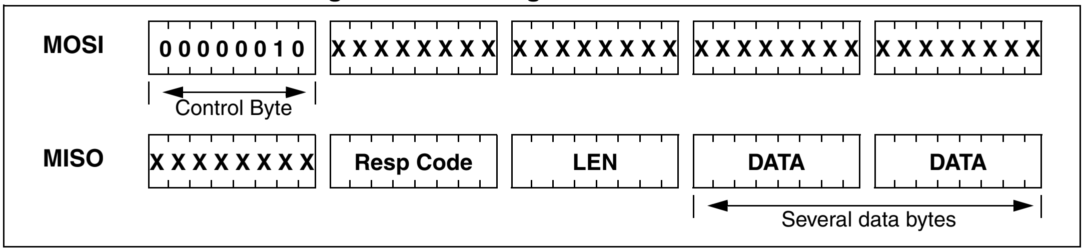


Table 5. Interpretation of flags

Bit	Meaning (Application point of view)
[7:4]	Not significant
3	Data can be read from the CR95HF when set.
2	Data can be sent to the CR95HF when set.
[1:0]	Not significant

Figure 10. Reading data from CR95HF



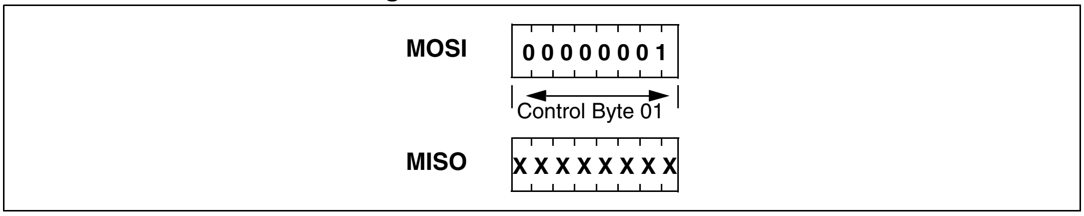
Data must be sampled at the rising edge of the SCK signal.

'Sending', 'Polling' and 'Reading' commands must be separated by a high level of the SPI_SS line. For example, when the application needs to wait for data from the CR95HF, it asserts the SPI_SS line low and issues a 'Polling' command. Keeping the SPI_SS line low, the Host can read the Flags Waiting bit which indicates that the CR95HF can be read. Then, the application has to assert the SPI_SS line high to finish the polling command. The Host asserts the SPI_SS line low and issues a 'Reading' command to read data. When all data is read, the application asserts the SPI_SS line high.

The application is not obliged to keep reading Flags using the Polling command until the CR95HF is ready in one command. It can issue as many 'Polling' commands as necessary. For example, the application asserts SPI_SS low, issues 'Polling' commands and reads Flags. If the CR95HF is not ready, the application can assert SPI_SS high and continue its algorithm (measuring temperature, communication with something else). Then, the application can assert SPI_SS low again and again issue 'Polling' commands, and so on, as many times as necessary, until the CR95HF is ready.

Note that at the beginning of communication, the application does not need to check flags to start transmission. The CR95HF is assumed to be ready to receive a command from the application.

Figure 11. Reset the CR95HF



To reset the CR95HF using the SPI, the application sends the SPI Reset command (Control Byte 01, see *Figure 11*) which starts the internal controller reset process and puts the CR95HF into Power-up state. The CR95HF will wake up when pin IRQ_IN goes low. The CR95HF reset process only starts when the SPI_SS pin returns to high level.

Caution: **SPI communication is MSB first.**

4.2.2 Interrupt mode

When the CR95HF is configure to use the SPI serial interface, pin IRQ_OUT is used to give additional information to user. When the CR95HF is ready to send back a reply, it sends an Interrupt Request by setting a low level on pin IRQ_OUT, which remains low until the host reads the data.

The application can use the Interrupt mode to skip the polling stage.

Caution: SPI communication is MSB first.

4.3 Error codes

Table 6. Possible error codes and their meaning

Code	Name	Meaning
0X63	EEmdSOFerror23	SOF error in high part (duration 2 to 3 etu) in ISO/IEC 14443B
0x65	EEmdSOFerror10	SOF error in low part (duration 10 to 11 etu) in ISO/IEC 14443B
0x66	EEmdEgt error	Extennded Guard Time error in ISO/IEC 14443B
0x67	ETr1 Too Big Too long	TR1 send by the card, reception stopped in ISO/IEC 14443BT
0x68	ETr1Too small Too small	TR1 send by the card in ISO/IEC 14443B
0x71	EinternalError	Wrong frame format decodes
0x80	EFrameRecvOK	Frame correctly received (additionally see CRC/Parity information)
0x85	EUserStop	Stopped by user (used only in Card mode)
0x86	ECommError	Hardware communication error
0x87	EFrameWaitTOut	Frame wait time out (no valid reception)
0x88	EInvalidSof	Invalid SOF
0x89	EBufOverflow	Too many bytes received and data still arriving
0x8A	EFramingError	if start bit = 1 or stop bit = 0
0x8B	EEgtError	EGT time out
0x8C	EInvalidLen	Valid for ISO/IEC 18092, if Length <3
0x8D	ECrcError	CRC error, Valid only for ISO/IEC 18092
0x8E	ERecvLost	When reception is lost without EOF received (or subcarrier was lost)
0x8F	ENoField	When Listen command detects the absence of external field
0x90	EUnintByte	Residual bits in last byte. Useful for ACK/NAK reception of ISO/IEC 14443 Type A.

4.4 Support of long frames

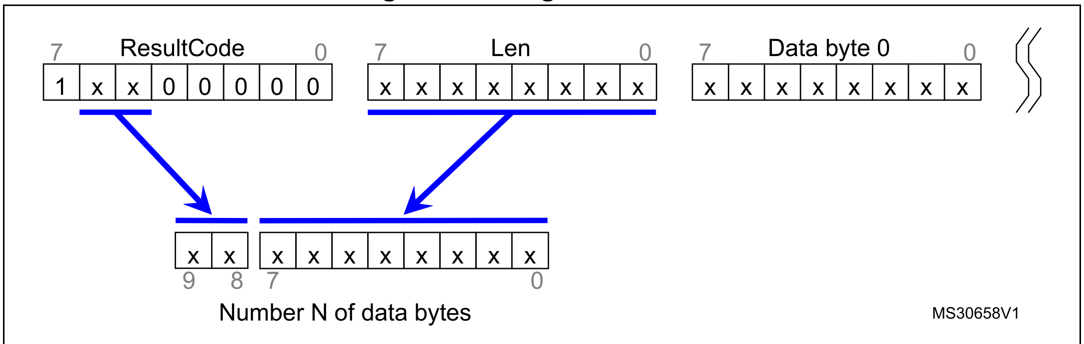
In Reader mode it is possible to receive up to 528 bytes of frame data from VICC and Type-B cards and up to 256 bytes of frame data from Type-A cards. In this case, the device sends a reply to the external MCU in the following format:

```
<ResultCode> + <Len> + <N bytes of data>
```

Table 7. Format of ResultCode

Bit	Meaning	
7	Always 1	
6	Bit 9 of Length	See examples and explanation below
5	Bit 8 of Length	
4	If set, there are residual bits in the last byte. Applicable only for Type-A protocol.	
3:0	Always 0	

Figure 12. Long frame format



The number of databytes is 10-bit long.

Table 8. Examples of ResultCode: Len pairs

ResultCode	Len	Length of data
0x80	0x00	0
0x80	0x01	1
0x80	0xFF	255
0xA0	0x00	256
0xA0	0x01	257
0xA0	0xFF	511
0xC0	0x00	512
0xC0	0x01	513

5 Commands

5.1 Command format

- The frame from the Host to the CR95HF has the following format:
  `<CMD><Len><Data>`
- The frame from the CR95HF to Host has the following format:
  `<RespCode><Len><Data>`

These two formats are available either in both UART and SPI modes.

Fields <Cmd>, <RespCode> and <Len> are always 1 byte long. <Data> can be from 0 to 253 bytes.

Note: The ECHO command is an exception as it has only one byte (0x55).

The following symbols correspond to:

>>> Frame sent by the Host to CR95HF

<<< Frame sent by the CR95HF to the Host

5.2 List of commands

Table 9 summarizes the available commands.

Table 9. List of CR95HF commands

Code	Command	Description
0x01	IDN	Requests short information about the CR95HF and its revision.
0x02	PROTOCOLSELECT	Selects the RF communication protocol and specifies certain protocol-related parameters.
0x04	SendRecv	Sends data using the previously selected protocol and receives the tag response.
0x07	IDLE	Switches the CR95HF into a low consumption Wait for Event (WFE) mode (Power-up, Hibernate, Sleep or Tag Detection), specifies the authorized wake-up sources and waits for an event to exit to Ready state.
0x08	RDREG	Reads Wake-up event register or the Analog Register Configuration (ARC_B) register.
0x09	WRREG	Writes Analog Register Configuration (ARC_B)) register or writes index of ARC_B register address. Writes the Timer Window (TimerW) value dedicated to ISO/IEC 14443 Type A tags. Writes the AutoDetect Filter enable register dedicated to ISO/IEC 18092 tags.
0x0A	BaudRate	Sets the UART baud rate.
0x55	Echo	CR95HF returns an ECHO response (0x55).
Other codes		ST Reserved

5.3 IDN command (0x01) description

The IDN command (`0x01`) gives brief information about the CR95HF and its revision.

Table 10. IDN command description

Direction	Data	Comments	Example
Host to CR95HF	0x01	Command code	>>>0x0100
	0x00	Length of data	
CR95HF to Host	0x00	Result code	<<<0x000F4E4643204653324A41535434002ACE In this example, <<<0x4E4643204653324A4153543400 : 'NFC FS2JAST4', #4 (Last Character of NFC FS2JAST4 means ROM code revision 4.) 0x2ACE: CRC of ROM (real CRC may differ from this example)
	<Len>	Length of data	
	<Device ID>	Data in ASCII format (13 bytes)	
	<ROM CRC>	CRC calculated for ROM content (2 bytes)	

It takes approximately 6 ms to calculate the CRC for the entire ROM. The application must allow sufficient time for waiting for a response for this command.

5.4 Protocol Select command (0x02) description

This command selects the RF communication protocol and prepares the CR95HF for communication with a contactless tag.

Table 11. ProtocolSelect command description

Direction	Data	Comments	Example
Host to CR95HF	0x02	Command code	See *Table 12: List of <Parameters> values for the ProtocolSelect command for different protocols on page 21* for a detailed example.
	<Len>	Length of data	
	<Protocol>	Protocol codes: 00: Field OFF 01: ISO/IEC 15693 02: ISO/IEC 14443-A 03: ISO/IEC 14443-B 04: ISO/IEC 18092 /NFC Forum Tag Type 3	
	<Parameters>	Each protocol has a different set of parameters. See *Table 12*.	
CR95HF to Host	0x00	Result code	<<<0x0000 Protocol is successfully selected
	0x00	Length of data	
CR95HF to Host	0x82	Error code	<<<0x8200 Invalid command length
	0x00	Length of data	

Table 11. ProtocolSelect command description (continued)

Direction	Data	Comments	Example
CR95HF to Host	0x83	Error code	<<<0x8300 Invalid protocol
	0x00	Length of data	

Note that there is no ‘Field ON’ command. When the application selects an RF communication protocol, the field automatically switches ON .

When the application selects a protocol, the CR95HF performs all necessary settings: it will choose the appropriate reception and transmission chains, switch ON or OFF the RF field and connect the antenna accordingly.

Different protocols have different sets of parameters. Values for the `<Parameters>` field are listed in *Table 12*.

Table 12. List of <Parameters> values for the ProtocolSelect command for different protocols

Protocol	Code	Parameters			Examples of commands
		Byte	Bit	Function	
Field OFF	0x00	0	7:0	RFU	>>>0x02020000
ISO/IEC 15693	0x01	0	7:6	RFU	H 100 S: >>>0x02 02 01 01 H 100 D: >>>0x02 02 01 03 H 10 S: >>>0x02 02 01 05 H 10 D: >>>0x02 02 01 07 L 100 S: >>>0x02 02 01 21 L 100 D: >>>0x02 02 01 23 L 10 S: >>>0x02 02 01 25 L 10 D: >>>0x02 02 01 27 In these examples, the CRC is automatically appended.
			5:4	00: 26 Kbps (H) 01: 52 Kbps 10: 6 Kbps (L) 11: RFU	
			3	0: Respect 312-µs delay 1: Wait for SOF (1)	
			2	0: 100% modulation (100) 1: 10% modulation (10)	
			1	0: Single subcarrier (S) 1: Dual subcarrier (D)	
			0	Append CRC if set to ‘1’. (1)	

Table 12. List of <Parameters> values for the ProtocolSelect command for different protocols (continued)

Protocol	Code	Parameters			Examples of commands
		Byte	Bit	Function	
ISO/IEC 14443 Type A NFC Forum Tag Type 1 (Topaz) NFC Forum Tag Type 2 NFC Forum Tag Type 4A	0x02	0	7:6	Transmission data rate 00: 106 Kbps 01: 212 Kbps (2) 10: 424 Kbps 11: RFU	>>>0x02020200: ISO/IEC 14443 Type A tag, 106 Kbps transmission and reception rates, Time interval 86/90 Note that REQA, WUPA, Select20 and Select70 commands use a fixed interval of 86/90 µs between a request and its reply. Other commands use a variable interval with fixed granularity. Refer to the ISO/IEC 14443 standard for more details.
			5:4	Reception data rate 00: 106 Kbps 01: 212 Kbps (2) 10: 424 Kbps 11: RFU	
			3	RFU	
			2:0	RFU	
		1	7:0	PP	These 5 bytes are optional. The default PP:MM:DD value is 0 (corresponds to FDT 86/90µs) . For other values, FDT = (2^PP)*(MM+1)*(DD+128) *32/13.56 µs
		2	7:0	MM	
		3	7:0	DD (optional to PP:MM)	
		4	7:0	ST Reserved (Optional)	
		5	7:0	ST Reserved (Optional)	

Table 12. List of `<Parameters>` values for the ProtocolSelect command for different protocols (continued)

Protocol	Code	Parameters			Examples of commands
		Byte	Bit	Function	
ISO/IEC 14443 Type B NFC Forum Tag Type 4B	0x03	0	7:6	Transmission data rate 00: 106 Kbps 01: 212 Kbps 10: 424 Kbps 11: 848 Kbps	>>>0x02020301: ISO/IEC 14443 Type B tag with CRC appended
			5:4	Reception data rate 00: 106 Kbps 01: 212 Kbps 10: 424 Kbps 11: 848 Kbps	
			3:1	RFU	
			0	Append CRC if set to '1'. (1)	
		1	7:0	PP	These 9 bytes are optional. Default value of PP:MM:DD is 0 and corresponds to FWT ~302µs. FWT = (2^PP)*(MM+1)*(DD+128)* 32/13.56 µs
		2	7:0	MM	
		3	7:0	DD (optional to PP:MM)	
		5:4	7:0	TTTT (Optional)	TR0 = TTTT/FC (LSB first), default 1023 = 0x3FF
		6	7:0	YY (Optional)	PCD Min TR1 (Min_TR1 = 8 * XX / f_S), default = 0
		7	7:0	ZZ (Optional)	PCD Max TR1 (Max_TR1 = 8 * ZZ / f_S), default = 26 = 0x1A
		8	7:0	ST Reserved (Optional)	
		9	7:0	ST Reserved (Optional)	

Table 12. List of `<Parameters>` values for the ProtocolSelect command for different protocols (continued)

Protocol	Code	Parameters: Byte	Parameters: Bit	Parameters: Function	Examples of commands
ISO/IEC 18092 NFC Forum Tag Type 3	0x04	0	7:6	Transmission data rate 00: RFU 01: 212 Kbps 10: 424 Kbps 11: RFU	>>>0x02020451: ISO/IEC18092 tag, 212 Kbps transmission and reception rates with CRC appended. Parameter 'Slot counter' is not mandatory. If it is not present, it is assumed that SlotCounter = 0x00 (1 slot) For device detection commands, byte 1 bit 4 must be set to '0'. In this case, the FWT is 2.4 ms for the 1st slot and 1.2 ms more for each following slot, if slot counter is specified. If slot counter = 0x10, the CR95HF does not respect reply timings, but polls incoming data and searches a valid response during ~8.4 ms.
			5:4	Reception data rate 00: RFU 01: 212 Kbps 10: 424 Kbps 11: RFU	
			3:1	RFU	
			0	Append CRC if set to '1'.[1]	
		1	7:5	RFU	
			4	0: FWT = 2.4 ms 1: FWT is specified by PP:MM bits	
			3:0	Slot counter 0: 1 slot 1: 2 slots ... F: 16 slots	
		2	7:0	PP	These 3 bytes are optional. Default value PP:MM:DD: is 0 and corresponds to RWT ~302µs. RWT = (2^PP)*(MM+1)*(DD+128)*32/13.56µs
		3	7:0	MM	
		4	7:0	DD (optional to PP:MM)	

1. It is recommended to set this bit to '1'.
2. Not characterized.

5.5 Send Receive (SendRecv) command (0x04) description

This command sends data to a contactless tag and receives its reply.

Before sending this command, the Host must first send the PROTOCOLSELECT command to select an RF communication protocol.

If the tag response was received and decoded correctly, the `<Data>` field can contain additional information which is protocol-specific. This is explained in *Table 14*.

Table 13. SENDRECV command description

Direction	Data	Comments	Example
Host to CR95HF	0x04	Command code	See *Table 14* and *Table 18* for detailed examples.
	<Len>	Length of data	
	<Data>	Data to be sent	
CR95HF to Host	0x80	Result code	<<<0x800F5077FE01B30000000000 71718EBA00 The tag response is decoded. This is an example of an ISO/IEC 14443 ATQB response (Answer to Request Type B)
	<Len>	Length of data	
	<Data>	Data received. Interpretation depends on protocol	
CR95HF to Host	0x90	Result code	<<<0x90040x240000 (exception for 4-bit frames where 'x' represents ACK or NAK value) 90: Result code for "non-integer number of bytes are received" 04: total length of data 0A or 00: Data 24: "2" means no CRC, "4" means 4 significant bits in Data byte. 00 00: No collision in response Example ACK <<< 0x90040A240000 Example NAK <<< 0x900400240000
	<Len>	Length of data	
	ACK or NAK	ISO 14443-A ACK or NAK detection	
	xx yy zz	3-byte response flag analysis	xx: Error type and number of significant bits in first data byte yy: First byte collision zz: First bit collision (1)
CR95HF to Host	X0 + <Len> + Data (See *Support of long frames on page 18*)		
CR95HF to Host	0x86	Error code	<<<0x8600 Communication error
	0x00	Length of data	
CR95HF to Host	0x87	Error code	<<<0x8700 Frame wait time out or no tag
	0x00	Length of data	