95HF series
NFC / RFID Reader IC

June 2018
The 95HF product is an integrated reader IC for contactless applications with several key features:

- The 95HF series manages frame coding and decoding in Reader / Writer mode (CR95HF / ST95HF) and in Card Emulation mode (only ST95HF) for standard applications such as near field communication (NFC), proximity and vicinity standards (RFID).

- Multiprotocol support @13.56MHz
  - ISO/IEC 14443 Type A and B, ISO/IEC 15693, ISO/IEC 18092, MIFARE® Classic compatible

- Communication interfaces with a Host Controller
  - Serial Peripheral Interface (SPI) Slave Interface 2MHz
  - Universal Asynchronous Receiver/Transmitter (UART) (only for CR95HF)
  - Up to 528-Byte RAM buffer for Reader / Writer & 256-Byte RAM buffer for Card Emulation

- Fast data transfer speed
  - Up to 424 Kb/s (ISO14443-A / ISO18092), up to 848 Kb/s (ISO14443-B), up to 52.6 Kb/s (ISO15693)
Main 95HF Market Segments

**Smart Industry**
- Maintenance, Factory Automation

**Smart Home**
- Home Gateway, Gaming

**Smart City**
- Lighting, Access lock
Key Use Cases

Access control / data reading

- Activate / Deactivate access
- Data programming

Commissioning for Wireless industrial network

- ID Activation
- Parameter settings

Device programming in production

- In-the-box programming
- Simple and flexible

Servicing & Maintenance

- Download records history with contactless
- Update parameters
**Use cases**
- RFID / NFC reader
- Product identification or authentication
- Product configuration (parameter settings)
- Access Control, Digital Door Locks
- Medical, industrial & production reader equipment

**Key Features**
- All NFC modes supported (ISO14443, ISO15693, FeliCa)
- Fast data transfer (up to 848kb/s)
- Low power modes

**Key Benefits**
- Simple implementation / limited BOM
- Easy-to-use evaluation / development kits
- Reference designs, application notes
- Cost effective solution

---

### CR95HF

<table>
<thead>
<tr>
<th>Reader Writer</th>
<th>ISO14443</th>
<th>RAM BUFFER</th>
<th>SPI</th>
<th>UART</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ISO15693</td>
<td>2.7/5.5V</td>
<td>2Mb/s</td>
<td>2.7/5.5V</td>
</tr>
<tr>
<td></td>
<td>FeliCa</td>
<td>RAM BUFFER</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NFC</td>
<td>848kb/s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.23W</td>
<td>528-Byte</td>
<td>2Mb/s</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Digital output (IRQ)**

QFN32
ST95HF NFC / RFID Transceiver

**Use cases**
- Dynamic data exchange between a NFC phone and a NFC device
  - User settings update, information log download, firmware upgrade…
  - Dynamic wireless pairing with hand-over

**Key Features**
- Reader-Writer (R/W) and Card Emulation (CE)
- All NFC modes supported (ISO14443, ISO15693, FeliCa)
- Fast data transfer (up to 848kb/s)

**Key Benefits**
- Simple implementation / limited BOM
- Easy-to-use evaluation / development kits
- Reference design notes, application notes
- Cost effective solution
## ST25R HF Readers

<table>
<thead>
<tr>
<th></th>
<th>CR95HF</th>
<th>ST95HF</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
<td>Entry-Level Reader</td>
<td>Entry-Level Reader</td>
</tr>
<tr>
<td><strong>Reader/Writer mode</strong></td>
<td>ISO14443A/B</td>
<td>ISO14443A/B</td>
</tr>
<tr>
<td></td>
<td>ISO15693</td>
<td>ISO15693</td>
</tr>
<tr>
<td></td>
<td>Felica</td>
<td>Felica</td>
</tr>
<tr>
<td><strong>Card emulation mode</strong></td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>P2P mode</strong></td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>RF speed</strong></td>
<td>848kbps</td>
<td>848kbps</td>
</tr>
<tr>
<td><strong>Market certification</strong></td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>Advanced features</strong></td>
<td>Ind wake-up</td>
<td>Ind wake-up</td>
</tr>
<tr>
<td><strong>Interface</strong></td>
<td>SPI 2Mbps</td>
<td>SPI 2Mbps</td>
</tr>
<tr>
<td></td>
<td>UART 2MHz</td>
<td></td>
</tr>
<tr>
<td><strong>Power supply</strong></td>
<td>2.7V - 5.5V</td>
<td>2.7V - 5.5V</td>
</tr>
<tr>
<td><strong>Output power</strong></td>
<td>0.23W</td>
<td>0.23W</td>
</tr>
<tr>
<td><strong>Temperature range</strong></td>
<td>-25°C to +85°C</td>
<td>-25°C to +85°C</td>
</tr>
<tr>
<td><strong>Package</strong></td>
<td>32-pin QFN (5x5mm)</td>
<td>32-pin QFN (5x5mm)</td>
</tr>
</tbody>
</table>
CR95HF: Operating Modes

- CR95HF has 2 modes operating modes:
  - Wait for Event (WFE):
    - This mode includes four low consumption states:
      - Power-up
      - Hibernate
      - Sleep
      - Tag detector
  - Active mode:
    - Ready: RF is OFF and the CR95HF waits for a command (ProtocolSelect, ...) from external Host
    - Reader: CR95HF communicates actively with a tag or an external host (an MCU, for example)
- CR95HF can switch from one mode to another
ST95HF: Operating Modes

- ST95HF has 2 modes operating modes:
  - **Wait for Event (WFE):**
    - This mode includes four low consumption states:
      - Power-up
      - Hibernate
      - Sleep / Field Detector
      - Tag detector
  - **Active mode:**
    - Ready: RF is OFF and the ST95HF waits for a command (ProtocolSelect, …) from external Host
    - Reader: ST95HF communicates actively with a tag or an external host (an MCU, for example)
    - Card Emulation: The ST95HF can communicate as a Card or Tag with an external reader. The Card or Tag application is located in the Host and communicates with the ST95HF via the SPI interface.

- ST95HF can switch from one mode to another

ST95HF initialization and operating state change
CR95HF / ST95HF: Startup Sequence

- After the power supply is established at power-on, the 95HF waits for a low pulse on the pin IRQ_IN ($t_1$) before automatically selecting the external interface (SPI) or (UART in case of CR95HF) and entering Ready state after a delay ($t_3$)

![Diagram showing the startup sequence with time labels $t_0$, $t_1$, $t_2$, $t_3$ and associated descriptions]

- $t_0$: initial wake-up delay - 100µs (min)
- $t_1$: minimum interrupt width - 10µs (min)
- $t_2$: delay for the serial interface selection – 250ns (typ)
- $t_3$: High Frequency Oscillator setup time – 10ms (max)
- $t_4$: $V_{PS}$ ramp-up time from 0V to $V_{PS}$ - 10ms (max)
CR95HF / ST95HF: SPI Interface

• Serial Peripheral Interface (SPI)
  • Polling mode
    • In order to send commands and receive replies, the application SW has to perform 3 steps:
      • Send the command to the 95HF
      • Poll the 95HF until is ready to transmit the response
      • Read the response
    • The application SW should never read data from the 95HF without being sure that the 95HF is ready to send the response. The maximum allowed SPI communication speed is \( f_{\text{SCK}} \) (SPI clock frequency)
    • A control byte is used to specify a communication type and direction:
      • 0x00: Send command to the 95HF
      • 0x03: Poll the 95HF
      • 0x02: Read data from the 95HF
      • 0x01: Reset the 95HF
  • Interrupt mode
    • When the 95HF is configure to use the SPI serial interface, pin IRQ_OUT is used to give additional information to user. When the 95HF 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.
CR95HF: UART Interface

- **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 baud rate is 2Mbps.

  - 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.
CR95HF / ST95HF: Commands

• **Command format**
  • Frame sent by the Host to the 95HF: `<CMD><Len><Data>`
  • Frame sent by the 95HF to the Host: `<RespCode><Len><Data>`
  ➢ These 2 formats are available either in both UART and SPI modes.

• **Command list**
  • **IDN**: provides 95HF short information and revision
  • **ProtocolSelect**: selects and configure the communication protocol
  • **SendRecv**: sends RF commands and receives tag response
  • **Listen (ST95HF used)**: Listens for data using previously selected protocol (used in CE mode)
  • **IDLE**: sets the 95HF in a low power consumption mode “Wait for Event mode” (Power-up, Hibernate, Sleep or tag detection) and specifies the wake-up source
  • **RdREG**: allows to read the Wake-up register and the Analog configuration register
  • **WrREG**: allows to write the Analog configuration register
  • **BaudRate**: sets the UART baud rate
  • **Echo**: simple serial interface echo command
• **VFQFPN32 Package** – 5.0 x 5.0mm
95HF Support Eco-system

Easy-to-use and customer-oriented

- STM32Nucleo hardware ecosystem
- Discovery kit STM32 based
- Antenna e-design tool
- Schematic, BOM, Gerber
- STM32Cube software ecosystem
- e2e community
- PC software tool ST25 SDK
- Documentation

STM32Nucleo
Discovery kit
Antenna e-design tool
Schematic, BOM, Gerber
STM32Cube
e2e community
PC software tool
Documentation
CR95HF Evaluation Boards

**CR95HF demo board**
- CR95HF NFC multi-protocol reader IC
- 47x34 mm 2 turns double layer antenna etched on PCB and associated tuning circuit
- STM32F1 microcontroller
- USB & JTAG connectors

**CR95HF Nucleo shield board**
- CR95HF NFC multi-protocol reader IC
- 47x34mm 4 turns antenna etched on PCB
- SPI (Slave interface) or UART
- Up to 528-byte command/reception buffer
- Optimized power management
- Powered through Arduino™ UNO R3 connector

**ST95HF discovery kit**
- ST95HF NFC transceiver IC
- 47x34mm 4 turns antenna
- STM32F1 micro-controller
- LCD color display + Joystick + LEDs
- USB & JTAG connectors
## ST25R Part Numbers

<table>
<thead>
<tr>
<th>Package</th>
<th>Features</th>
<th>Part Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry-Level HF Readers</td>
<td>QFN32</td>
<td>Reader / Writer (R/W) R/W &amp; Card emulation (CE)</td>
</tr>
</tbody>
</table>
Thank You!

Solutions for NFC / RFID Tags and Readers