# M519-SUV Data Sheet and Integration Guide



Reference
PFT23234

Version
AF

Category
Datasheets

Diffusion
Public

State
Approved

Family
SpringCore / M519

Date
25/03/2025

Page count
46

## Revision History

| Ver. | Date | Author | Verifier | Approb. | Details |
|---|---|---|---|---|---|
| AA | 12/09/2023 | CFE | | | Preview |
| AB | 24/10/2023 | JDA | CFE | JDA | First public release (but still with a lot of TBD) |
| AC | 29/11/2023 | JDA | | | Temperature range increased |
| AD | 13/03/2024 | JDA | | | Added pictures (section 0)<br>Added sections 0, 6.7 and 6.8 |
| AE | 20/02/2025 | JDA | | | New template, minor corrections, new layout for section 7 in coherence with other devices in the family |
| AF | 20/03/2025 | JDA | | | Corrected pin-numbering of J1 for consistency (no change in actual pinout)<br>§ 3.1.1:<br>- Temperature range edited to reflect actual performance that could be achieved in the field<br>- MTBF added |
| | | | | | |

TABLE OF CONTENT

# 1 Introduction

## 1.1 Overview

This document describes the functionalities and electrical specifications of the SpringCard SpringSeed M519-SUV running the SpringCore firmware version 1.30 or higher.

The M519-SUV is a OEM NFC/RFID HF module with an integrated antenna.

## 1.2 Related documents

### 1.2.1 Documents available as PDF

| Reference | Title / Description |
|---|---|
| PFT22217 | M519 Datasheet and Hardware Integration Guide |
| PMD23175 | M519-SRK Getting Started Guide |
| PNA23174 | Using the M519 in PC/SC Coupler mode over a Serial interface |
| PNA23207 | Using the M519 in PC/SC Coupler mode over a USB interface |
| PNA23208 | Using the M519 in Smart Reader or RFID Scanner mode |
| PNA23189 | Using the M519 in SpringProx Legacy mode |

### 1.2.2 Online materials

Documentation of the SpringCore firmware.

https://docs.springcard.com/books/SpringCore/Welcome

SpringCard Tech Zone, the blog of the R&D Team

https://tech.springcard.com/

## 1.3 Order codes

The M519 itself is highly configurable, and supports many operating modes. With its remote antenna, the M519-SUV offers a wide choice or hardware variants. A variety of order codes are available, enabling customers to tailor the M519-SUV to streamline their integration process[A]. However, it must be remembered that changing the configuration of the device in the field or on the production line is straightforward, ensuring flexibility in deployment scenarios.

| Order code | Product | USB[B] | Serial[B] | Mode | Option | MOQ |
|---|---|---|---|---|---|---|
| SC23219 | M519-SUV | Yes | RS-TTL | PC/SC | | 10 units |
| SC23220 | M519-SUV | Yes | RS-232 | PC/SC | | 30 units |
| SC23222 | M519-SUV | Yes | RS-TTL | RFID Scanner | QWERTY | 30 units |
| SC23221 | M519-SUV | Yes | RS-TTL | RFID Scanner | AZERTY | 10 units |
| SC23223 | M519-SUV | Yes | RS-TTL | Smart Reader | Protocol MK1 | 10 units |
| SC23224 | M519-SUV | Yes | RS-232 | Smart Reader | Protocol MK1 | 30 units |
| SC23225 | M519-SUV | Yes | RS-485 | Smart Reader | Protocol MK1 | 30 units |
| SC23226 | M519-SUV | Yes | RS-TTL | SpringProx Legacy | | 10 units |
| SC23227 | M519-SUV | Yes | RS-232 | SpringProx Legacy | | 30 units |

Note A

Any custom configuration may be created upon request, with a MOQ of 120 units. Contact SpringCard Sales team for more information.

Note B

The M519-SUV is provided without any USB nor Serial cords. See § 0 for the accessories.

### Warning

The in-the-field reconfigurability of the M519-SUV allows for post-deployment modifications of the configuration. Consequently, a device's actual setup is likely to differ from its delivery setup. This may be a source of confusion for after-sales services, when doing a retrofit or when ordering new batches.

It is therefore recommended that customers (customers, integrators, etc.) carefully document any modifications made to the original configuration, keep a record of updates, and share this information with their own quality control, production and technical support teams.

When ordering new batches, it is also advisable to specify the desired configuration, to ensure consistency with the devices already deployed.

# 2 About the SpringSeed M519-SUV

## 2.1 General description

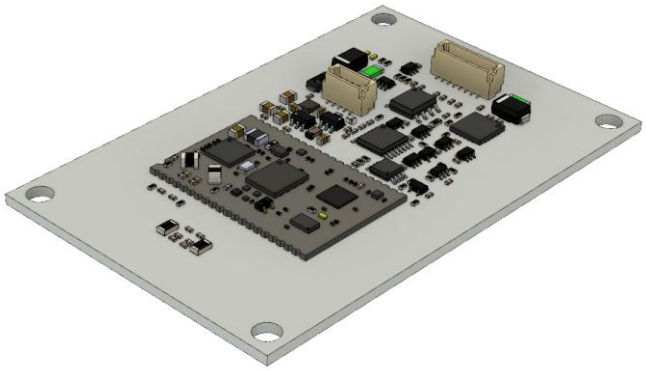


*Figure 1: 3D view of the M519-SUV*

The SpringCard SpringSeed M519-SUV is a compact OEM NFC/RFID HF versatile device that integrates the SpringSeed M519 core module and a coil antenna on a single board.

The M519-SUV is able to communicate with virtually any contactless smart card, RFID label, NFC tag or NFC smartphone compliant with one of the standard technologies in the 13.56MHz range. The basic 69x45mm antenna is optimised for ID-1 cards (class I, II and III as per ISO/IEC 14443-1) and smartphones, while maintaining compatibility with smaller form factors tags.

The M519-SUV inherits all the operating modes from the SpringSeed M519 core module (PC/SC Coupler, Smart Reader, RFID Scanner, etc), with a choice between USB and Serial interfaces. The Serial interface encompasses in a sithree electrical layers: RS-TTL, RS-232 and RS-485.

The M519-SUV is designed to be used in industrial equipment, professional terminals or consumer devices.

The overall dimensions (69x45mm), the location of the mounting holes and the connectivity (Serial on JST-8, USB on JST-5) make it a drop-in replacement for earlier SpringCard products from the K531, K632, K663 and H663 families.

## 2.2 Features and benefits

### 2.2.1 Ease of integration into any machine or assembly

- Small design (69.0x45.0mm, 6.2mm max. thickness),
- NXP PN5190 NFC/RFID HF frontend allowing best in-class performance/power ratio,
- Symmetrical (balanced) antenna delivering optimal performance even in harsh environments,

- Single-source power supply, power saving modes, low power card detection on less than 1mA on the RF power line.

### 2.2.2 Ease of integration for any application and use case

- Serial interface supporting a various choice of protocols, making it easy to operate the device even from low-end MCUs,
- Single hardware supporting both a USB interface and the three variants of the Serial interface (RS-TTL, RS-232, RS-485),
- USB interface supported by Linux (even low-end embedded SOCs), Windows and macOS,
- In-the-field configuration and firmware upgrade (flash) without interruption,
- Comprehensive starter kit and a wide range of SDK and samples available for free on GitHub (list of our repositories: https://github.com/springcard),
- Compliant with earlier SpringCard SDKs and software libraries (PC/SC, SpringProx, etc).

### 2.2.3 Open and interoperable

- Standard-compliant USB CCID (PC/SC) and USB HID keyboard wedge (RFID Scanner) profiles,
- Digital layer of the Contactless stack tested[1] against the following test suites :
  - EMV CL L1,
  - NFC Forum CR13,
  - CEN/TS 16794 aka ISO/IEC TS 24192, RCTIF,
- Support of Apple Pay ECP1 and ECP2 for Passkit / Apple VAS applications, support of Google Smart Tap,
- Support of NXP Mifare and NTAG families, the largest portfolio of contactless cards.
- Support of ST25 families, and more.

## 2.3 Typical applications

- Public transport, public bike systems, car sharing,
- Car-park gates or cashiers,
- Kiosk, vending machines,
- Mobile or affixed terminals for loyalty, events, gaming…
- Access control, secure identification,
- and more.

## 2.4 Integration and development

This document is the starting point for the M519-SUV integration process. Additional useful information can be found in the SpringCard document base.

[1] Tests carried out in-house do not prejudge the compatibility. As the M519-SUV is an OEM module, it is in any case the integrator's responsibility to verify the compliance and obtain Analog and Digital approvals for the end product.

### 2.4.1 Hardware

The integration of a loop antenna generating an alternating magnetic field and communicating in the near field with passive targets requires control of the electromagnetic environment.

Document [PMI9C2P] provides the recommendations that must be followed.

### 2.4.2 Software

The M519-SRK is a complete Starter Kit for customers who want to evaluate the SpringSeed M519 core module and then build their own solution around it. It makes it easy for integration and development engineers to get to grips with the module.

Software developers who want to integrate the M519-SUV in their solution will find all relevant information in the Getting Started Guide that comes with the M519-SRK ([PMD23175]) and with the related Application Notes.

## 2.5 Pictures

*Figure 2: M519-SUV, top view*


*Figure 3: M519-SUV, bottom view (actual label depends on part number)*

## 2.6 Related accessories

### 2.6.1 Serial interface and cord

| Order code | Description | Remark | MOQ |
|---|---|---|---|
| SC24051 | USB to Serial JST-8 interface, RS-232/RS-TTL/RS-485 level selectable by switches | SC15145 cord included<br>USB Mini-B cord not included<br>Based on FTDI USB to Serial bridge, driver available on www.ftdichip.com | 1 units |
| SC15145 | JST-8 to JST-8 cord, 30cm (red wires) | | 15 units |
| SC15046 | JST-8 to free wires, 30cm (red wires) | | 15 units |

#### Warning 1

On delivery, the default electrical level of the SC24051 interface is RS-TTL. Its switches must be set to the right position before using it with a M519-SUV configured with a RS-232 or RS-485 electrical level on the Serial interface.

#### Warning 2

The first generations of USB/Serial interfaces (part number SC19227) suffer from a limitation in the current peaks they can supply; therefore, they are not able to power the M519-SUV correctly. Only the "MK2" generation of USB/Serial interfaces (part number SC24051) can be used with the M519-SUV.

### 2.6.2 USB cord

| Order code | Description | Remark | MOQ |
|---|---|---|---|
| SC15252 | USB-A to JST-5 cord, 180cm (black) | | 10 units |

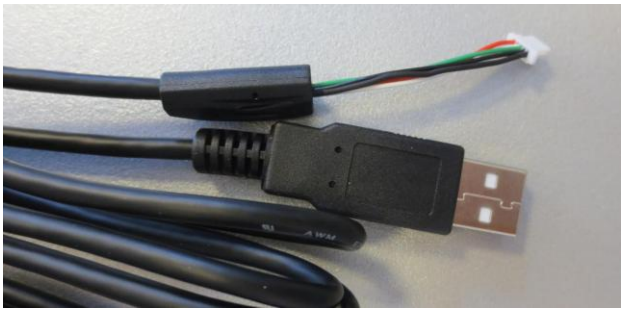


*Figure 4: SC15252 USB-A to JST-5 cord*

# 3 Technical data

## 3.1 General

### 3.1.1 Main board

| | |
|---|---|
| Dimensions | 69.0 x 45.0 x 6.2 mm |
| Weight | Approx. 12g |
| Power supply | 3.3V or 5V nominal for Serial operation<br>5V (powered by the bus) for USB operation |
| Power requirement | Nominal: 200mA<br>Max: 400mA<br>Power saving: < 1mA |
| Temperature range | Operation: –25°C to +75°C<br>Storage: –40°C to +85°C |
| Environment | Humidity 0 – 90% (non-condensing) |
| MTBF | 1 800 000 hours |
| Compliance | CE RED, FCC, REACH, ROHS |

## 3.2 NFC/RFID HF (contactless) interface

| | |
|---|---|
| Antenna | Integrated balanced coil, 2x2 loops |
| Gain | -57dBi |
| Carrier frequency | 13.56MHz |
| RF power | Max 2W (33dBm) |
| Typical field level | 1.5A/m at 40mm |
| Typical operating distance[A] | Up to 75mm |
| Supported standards | ISO/IEC 14443 A & B (PCD) / NFC-A and NFC-B<br>ISO/IEC 15693 (VCD) and ISO/IEC 18000-3M1 / NFC-V<br>ISO/IEC 18000-3M3 (RFID HF) / EPC HF JIS X 6319-4 / NFC-F<br>ISO/IEC 21481 (NFCIP-2) |
| Bitrates | 26, 53, 106, 212, 424, 848 kbit/s depending on the protocol |
| Supported card technologies (partial list)[B] | NFC Forum types 1, 2, 3, 4 and 5<br>Mifare Classic with CRYPTO1<br>All cards in NXP Mifare families: Mifare Plus, Mifare UltraLight, Desfire,<br>All cards in NXP NTAG and ICODE families<br>Innovatron (Calypso cards)<br>STMicroElectronics SR & LR, ST25<br>Infineon SLE44, SLE66, SRF55<br>Texas Instrument Tag-it<br>Sony FeliCa Lite<br>Apple ECP, Apple VAS (PassKit NFC), Google Smart Tap and other NFC applications on smartphones<br>Inside Secure PicoPass / HID iClass (ID only) |
| Advanced features | Low power card detection<br>Automatic waveform control |

Note A

The actual operating distance depends mainly on the characteristics of the target (card, tag or NFC object) and the constraints of the electromagnetic environment (noise, detuning, eddy currents, etc).

Note B

MIFARE®, MIFARE Classic®, MIFARE Plus®, MIFARE UltraLight® and MIFARE DESFire® are registered trademarks of NXP B.V. . All other trademarks belong to their respective owners.

## 3.3 Card emulation and peer-to-peer operations

| | |
|---|---|
| Supported standards | ISO/IEC 14443 A (PICC) / emulation of NFC Forum Type 4A Tag<br>ISO/IEC 18092 (NFCIP-1) active and passive, initiator and target |
| Bitrates | 106, 212, 424 kbit/s depending on the protocol |

## 3.4 Host interfaces

### 3.4.1 USB

| | |
|---|---|
| Standard | USB 2.0 device, compatible with USB 3 |
| Bitrate | Full speed (12Mbps) |
| Profiles | CCID (PC/SC)<br>HID keyboard<br>CDC-ACM (serial emulation)<br>SpringCore Direct |

### 3.4.2 Serial

The Serial interface of the M519-SUV supports three different electrical configurations (see § 6.4).

#### Warning

The default, out-of-factory configuration (that is defined by the order code and namely written on the label or the packaging) may have be overwritten by a user-defined configuration. Observe the LEDs (§ 4.3 and § 6.4) to know the active configuration

##### *3.4.2.1 Common data*

| | |
|---|---|
| Bitrate | 38400bps (default), up to 500kbps after handshaking |
| Format | 8 data bits, 1 stop bit, no parity, no flow control |
| Protocols | CCID over Serial<br>SpringProx Legacy<br>SpringCore Direct<br>RDR MK1<br>$SCRDR |

##### *3.4.2.2 RS-TTL configuration*

| | |
|---|---|
| With VCC=3.3V | RX/TX @0/3.3V |
| With VCC=5V | RX/TX @0/5V |

#### *3.4.2.3 RS-232 configuration*

| | |
|---|---|
| Standard | EIA-232 (RX/TX @-12/+12V) |

#### *3.4.2.4 RS-485 configuration*

| | |
|---|---|
| Standard | EIA-485 (differential) |

# 4 Mechanical specifications and pinouts (main board)

## 4.1 Dimensions

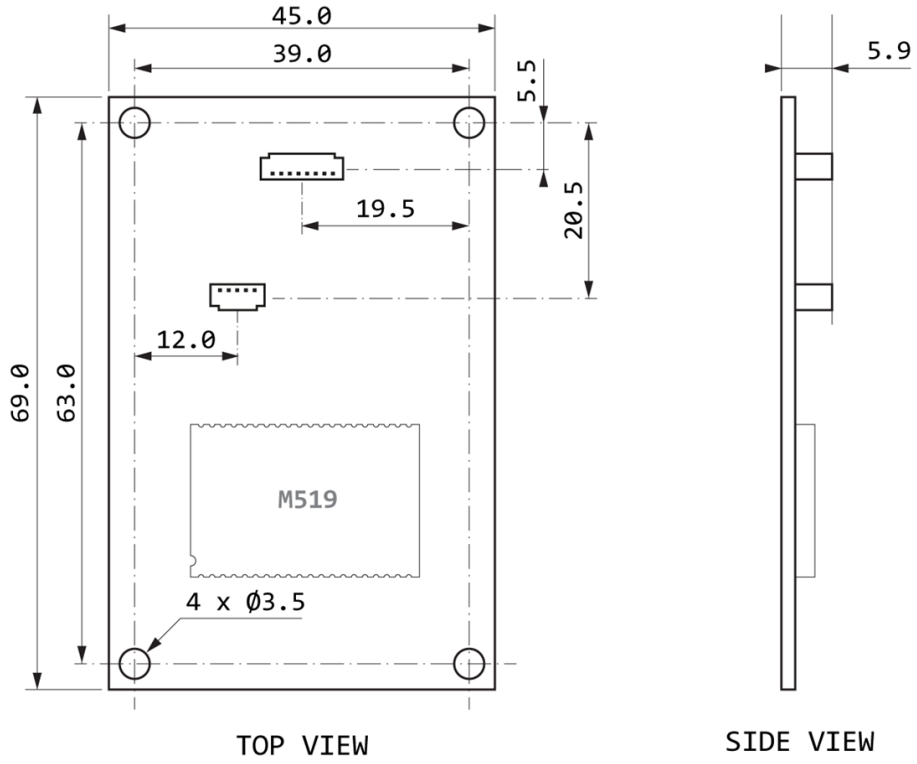



*Figure 5: Mechanical specifications, M519-SUV main board*

## 4.2 Pinout

### 4.2.1 JST-5 USB connector (J2)

J5 is a JST BM05B-SRSS-TB(LF)(SN) 5-position connector, dedicated to connecting the M519-SUV with its host, using USB as primary interface.

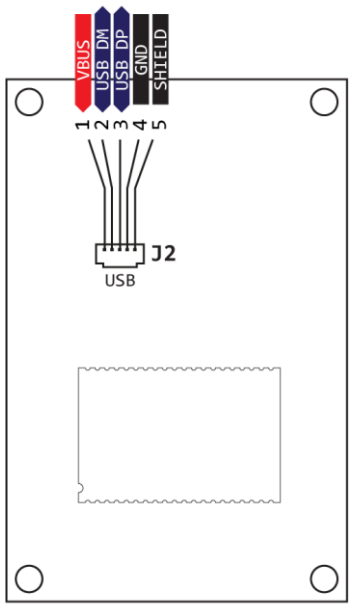



*Figure 6: J2 location and pinout*

| Pin | Symbol | Type | Description |
|---|---|---|---|
| #1 | VBUS | Power | Power supply + USB presence, 5V |
| #2 | USB_DM | USB | USB D- signal |
| #3 | USB_DP | USB | USB D+ signal |
| #4 | GND | Ground | Ground signal inside the USB cord |
| #5 | SHIELD | Ground | Shield of the USB cord |

### 4.2.2 JST-8 Serial connector (J1)

J1 is a JST BM08B-SRSS-TB(LF)(SN) 8-position connector, dedicated to connecting the M519-SUV with its host, using Serial (RS-232) as primary interface. Power is supplied to the device through the VCC (3.3V or 5V) power input.

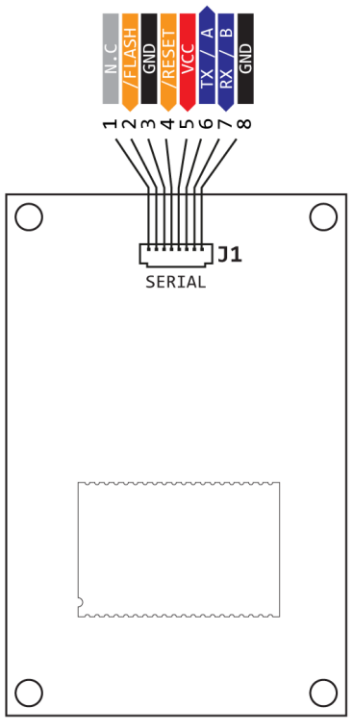
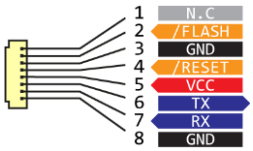


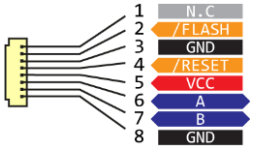

*Figure 7: J1 location and pinout*

#### *4.2.2.1 RS-TTL and RS-232*




| Pin | Symbol | Type | Description |
|---|---|---|---|
| #1 | NC | — | Leave unconnected |
| #2 | /FLASH | In | M519's force bootloader signal |
| #3 | GND | Ground | |
| #4 | /RESET | In | M519's reset signal |
| #5 | VCC | Power | External power supply, 5V or 3.3V |
| #6 | TX | Out | UART, M519 to host |
| #7 | RX | In | UART, host to M519 |
| #8 | GND | Ground | |

#### 4.2.2.2 RS-485




| Pin | Symbol | Type | Description |
|---|---|---|---|
| #1 | NC | — | Leave unconnected |
| #2 | /FLASH | In | M519's force bootloader signal |
| #8 | NC | — | Leave unconnected |
| #4 | /RESET | In | M519's reset signal |
| #5 | VCC | Power | External power supply, 5V or 3.3V |
| #6 | BUS_B | Out | UART, bus B signal |
| #7 | BUS_A | In | UART, bus A signal |
| #8 | GND | Ground | |

## 4.3 LEDs

The M519-SUV features 3 LEDs:

- LED 0 is on the M519 module itself; see [PFT22217] for details.
- LED 1 and LED 2 show the configuration of the Serial interface (see § 6.4 for details).

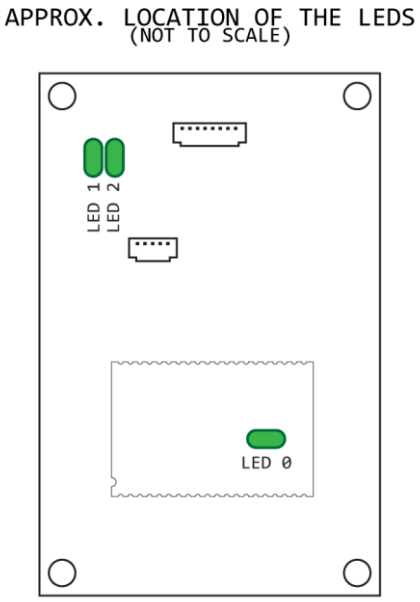



*Figure 8: Location of the LEDs*

# 5 Electrical specifications (main board)

## 5.1 Limiting values

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $VCC_{ABS}$ | Supply voltage on pin VCC | | 0 | 5.5 | V |
| $VBUS_{ABS}$ | Supply voltage on pin VBUS | | 0 | 5.5 | V |
| $V_{DIG}$ | Voltage on any digital pin | Powered by VCC | 0 | VCC + 0.2 | V |
| | | Powered by VBUS | 0 | VBUS + 0.2 | |
| $V_{ESD}$ | Electrostatic discharge voltage | | -500 | 500 | V |
| $T_{JUNCTION}$ | Junction temperature | | — | +120 | °C |
| $T_{STORAGE}$ | Storage temperature | No voltage applied | -40 | +85 | °C |

### Warning

Stresses beyond those listed under 'Limiting values' may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these conditions is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

## 5.2 Power

### 5.2.1 Power supply voltage

#### *5.2.1.1 USB operation*

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| VBUS | Power supply, USB | 4.5 | 5.0 | 5.2 | V |

### Warning

The Serial interface (J1) shall be left unconnected for USB operation.

#### *5.2.1.2 Serial operation*

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| VCC | Power supply, Serial | 3.0 | 3.3 or 5.0 | 5.2 | V |

### Warning

The USB interface (J2) shall be left unconnected for Serial operation.

### 5.2.2 Current consumption

REMARK

Max RF power may be restricted by configuration, to reduce the total power requirement, at the price of a shorter operating range.

#### *5.2.2.1 USB operation*

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $I_{VBUS,IDLE}$ | VBUS active but device not enumerated | — | 30 | 40 | mA |
| $I_{VBUS,RF\ OFF}$ | Device enumerated, RF not active | — | 30 | 40 | mA |
| $I_{VBUS,RF\ ON}$ | Device enumerated, RF active | — | 140 | 340 | mA |

#### *5.2.2.2 Serial operation, VCC=3.3V*

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $I_{VCC,RF\ OFF}$ | RF not active | — | 35 | 50 | mA |
| $I_{VCC,RF\ ON}$ | RF active | — | 300 | 430 | mA |
| $I_{VCC,LPCD}$ | Waiting for a card, low power mode | — | 0.2 | 1 | mA |

#### *5.2.2.3 Serial operation, VCC=5V*

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $I_{VCC,RF\ OFF}$ | RF not active | — | 30 | 40 | mA |
| $I_{VCC,RF\ ON}$ | RF active | — | 140 | 340 | mA |
| $I_{VCC,LPCD}$ | Waiting for a card, low power mode | — | 0.2 | 1 | mA |

## 5.3 /RESET and /FLASH pins

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{IL}$ | Low level input voltage | — | — | 1.0 | V |
| $V_{IH}$ | High level input voltage | 2.4 | — | — | V |
| $I_{IL}$ | Low level input leakage current | 55 | 80 | 165 | µA |
| $I_{IH}$ | High level input leakage current | -1 | ±0.01 | 1 | µA |

## 5.4 Serial interface, RS-TTL mode

### 5.4.1 RX pin

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{IL}$ | Low level input voltage | — | — | 0.8 | V |
| $V_{IH}$ | High level input voltage | 2.0 | — | VCC | V |
| $I_L$ | Input leakage current | — | — | ±5 | µA |

### 5.4.2 TX pin

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{OL}$ | Low level output voltage (at $I_{OL}$=1mA) | — | — | 0.65 | V |
| $V_{OH}$ | High level output voltage (at $I_{OH}$=1mA) | 2.4 | — | VCC | V |
| $I_{OS}$ | Short circuit output current | — | ±24 | ±32 | mA |
| SR | Up or down transition time | — | — | 0.4 | µs |

## 5.5 Serial interface, RS-232 mode

### 5.5.1 RX pin

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{IL}$ | Low level input voltage | — | -5 | 0.6 | V |
| $V_{IH}$ | High level input voltage | 2.4 | 5 | — | V |
| $R_I$ | Input resistance | 3 | 5 | 7 | kΩ |

### 5.5.2 TX pin

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{OL}$ | Low level output voltage (at $I_{OL}$=1mA) | — | -5.4 | -5 | V |
| $V_{OH}$ | High level output voltage (at $I_{OH}$=1mA) | 5 | 5.4 | — | V |
| $I_{OS}$ | Short circuit output current | — | ±35 | ±60 | mA |
| SR | Up or down transition time | — | — | 0.4 | µs |
| $R_O$ | Output resistance | 300 | 10M | — | Ω |

## 5.6 Serial interface, RS-485 mode

### 5.6.1 RX pin

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{IL}$ | Low level input voltage | — | -5 | 0.6 | V |
| $V_{IH}$ | High level input voltage | 2.4 | 5 | — | V |
| $R_I$ | Input resistance | 3 | 5 | 7 | kΩ |

### 5.6.2 TX pin

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| $V_{ID}$ | Differential input voltage, receiver | -12 | — | 12 | V |
| $I_I$ | Bus input current, receiver | -100 | — | 130 | µA |
| $V_{IL}$ | Low level differential voltage, receiver | — | -1.5 | -0.2 | V |
| $V_{IH}$ | High level differential voltage, receiver | -0.1 | 0 | — | V |
| $V_{OD}$ | Differential output voltage, transmitter | — | -1.5 | — | V |
| $I_{OS}$ | Short circuit output current, transmitter | — | — | ±265 | mA |

# 6 Application information

## 6.1 Operating modes, profiles and protocols

The M519-SUV is a versatile device, based on the SpringCard SpringCore firmware, that supports many operating modes. Every operating mode is associated to one or more USB profile and Serial communication protocol.

The paragraphs below summarizes the options and explains the basics; for a complete reference, please read the documentation of the SpringCore firmware:

https://docs.springcard.com/books/SpringCore/Introduction/Operating_Modes

### 6.1.1 PC/SC Coupler mode

In this mode, the M519-SUV is a complete smart card coupler: it can perform any transaction with a smart card, under full control of an application running in the host computer.

The smart card may be either contact or contactless (NFC/RFID) depending on only on its hardware interface, but operated the same way seen from the software interface.

As a PC/SC device, the M519-SUV is typically associated to a PC/SC driver and is supported by the computer's PC/SC stack. See Application Note [PNA23207] for reference.

The PC/SC Coupler mode is selected by writing $_H$02 into configuration register $_H$02C0.

#### *6.1.1.1 USB*

When the PC/SC Coupler mode is selected and the host interface is USB, the M519-SUV is a compound device that exposes both the CCID profile and the SpringCore Direct profile.

The CCID profile of the M519 is fully supported by the open-source CCID driver available on Linux together with the PC/SC-Lite stack, and by the CCID driver and PC/SC stack provided by Apple for macOS.

For Microsoft Windows, SpringCard driver SD16055 shall be used. This driver is available through Windows Update. It could be pre-installed using this setup package:

https://www.springcard.com/en/download/find/file/sd16055

REMARK

Microsoft also provides a CCID driver that works fine with the M519, but unfortunately this driver does not fully support other SpringCard devices, and does not support all the features of the SpringCore firmware. Using SpringCard driver instead simplifies the development and the technical support.

#### 6.1.1.2 Serial

When the PC/SC Coupler mode is selected and the host interface is Serial, the M519-SUV uses the CCID over Serial protocol.

See Application Note [PNA23174] for reference.

### 6.1.2 Smart Reader mode

In this mode, the M519-SUV is an autonomous or NFC/RFID reader.

It automatically grabs a token from a contactless cards, NFC tags or RFID labels, before transmitting it to the host. This simplifies the development of the host application, because the M519-SUV runs the transaction with the card, tag or label in a standalone-reader approach.

This also makes it possible to use a fast anticollision/inventory scheme, overriding the bottleneck introduced by a slow computer-based transaction or a too complex driver stack.

The Smart Reader mode is selected by writing $_H$03 into configuration register $_H$02C0.

#### 6.1.2.1 USB

When the Smart Reader mode is selected and the host interface is USB, the M519-SUV is a compound device that exposes both the HID keyboard profile and the SpringCore Direct profile.

Thanks to the standard HID keyboard profile, the M519-SUV is automatically recognized as a keyboard by all the major operating systems that support USB (Windows, macOS, Linux, Android, iOS…), and the data collected from NFC/RFID tags are automatically received as key-strokes in the active application. SpringCard calls this feature “RFID Scanner”.

For correct operation, the user must configure the M519-SUV with the same keyboard layout as the host system.

#### 6.1.2.2 Serial

When the Smart Reader mode is selected and the host interface is Serial, the M519-SUV sends the data collected from NFC/RFID tags over its serial line, using the protocol selected in configuration register $_H$02A0.

- $SCRDR protocol (default): data come as ASCII strings, starting with constant value “$SCRDR”.

Format specification:

```
$SCRDR;<Interface>;<Protocol>;<Template>;<RSSI>;<TagId>;<Move>;<TagData>;<TagDetails>*<Checksum><CR><LF>
```

Example:

```
$SCRDR;13.56;NFC-A;1;;047DAE02C84080;;00047DAE02C84080;*47<CR><LF>
```

- MK1 protocol: this is a legacy protocol, taken from earlier SpringCard devices, that provides only the TagId. The $SCRDR protocol shall be preferred.

Example:

```
+047DAE02C84080<CR><LF>
```

- JSON protocol: data is encapsulated in a JSON object.

Example:

```
{
      "Interface":"13.56",
      "Protocol":"NFC-A",
      "Template":1,
      "TagId":"047DAE02C84080",
      "TagData":"00047DAE02C84080",
      "Uptime":"699:16.850"
}
```

- TLV protocol: data object uses ASN.1 binary encoding rule.

Example:

```
B0 0E C1 04 00 03 01 01 C2 07 04 7D AE 02 C8 40 80
```

### 6.1.3 SpringProx Legacy

SpringProx Legacy is the proprietary protocol and mode of operation introduced in the 2000's with the first generation of SpringCard devices. PC/SC Coupler mode and the CCID protocol shall be preferred when designing a new system.

It must be understood by the developer/implementer that the M519-SUV is not a drop-in replacement for earlier devices based on the K531, K632 or K663 generations. The high-level behaviour is the same, but low-level control of the RF interface has to be rewritten. Anyway, in most situations, the migration process is straightforward provided that you have access to the source code of the Legacy application.

The SpringProx Legacy mode and protocol are selected by writing $_H$01 into configuration register $_H$02C0.

#### *6.1.3.1 USB*

When the SpringProx Legacy is selected and the host interface is USB, the M519-SUV uses the USB CDC ACM profile (communication device class, abstract control model). It is automatically recognized as a (virtual) communication port by all the major desktop operating systems (Windows, macOS, Linux…). The M519-SUV accepts SpringProx Binary and ASCII messages (not OSI3964) and it is possible to use the `springprox.dll` over this (virtual) communication port without a change.

#### *6.1.3.2 Serial*

When the SpringProx Legacy is selected and the host interface is Serial, the M519-SUV accepts SpringProx Binary and ASCII messages (not OSI3964) and it is possible to use the `springprox.dll` without a change.

## 6.2 Firmware upgrade

### 6.2.1 USB

When the device is connected to the host through its USB interface, upgrading the firmware of the M519-SUV could be done in-the-field, over a live system (live upgrade). The typical firmware upgrade procedure is:

- A new firmware is transmitted using the Direct protocol while the M519-SUV is operating "normally" (this new firmware is written in a temporary storage area),
- When the M519-SUV resets, its bootloader copies the new firmware from the storage area to the ROM of the microcontroller,
- The M519-SUV resets again, running its new firmware.

To perform the upgrade over USB, use either:

- SpringCoreFlash, a command line tool (member of the SpringCore Tools suite) that is easy to deploy and run unattended

  https://www.springcard.com/en/download/find/file/sq20029

- SpringCard Companion, a web application + local service, for a more user-friendly experience.

  https://companion.springcard.com

### 6.2.2 Serial

Live upgrade is possible over the Serial interface only when the device is configured for the SpringCore Direct protocol, fail-safe mode—which normally never happens in the field.

There are 3 different ways to upgrade a device that is connected through a Serial line,

1. Send the software RESET command to the device, with **$_H$FA $_H$DA** as parameters to have the device restart in bootloader mode, and write the new firmware using the bootloader over the Serial interface,

*Documentation of the RESET command:*

https://docs.springcard.com/books/SpringCore/Host_Protocols/Direct_Protocol/CONTROL_class/Actions/RESET

*Documentation of the bootloader protocol:*

https://docs.springcard.com/books/SpringCore/Host_Protocols/Direct_Protocol/DFU_class/index

2. Assert the /FLASH input line to LOW level and perform a hardware RESET of the device to have it restart in bootloader mode, and write the new firmware using the bootloader over the Serial interface,

*Documentation of the bootloader protocol:*

https://docs.springcard.com/books/SpringCore/Host_Protocols/Direct_Protocol/DFU_class/index

3. Connect a USB cable temporarily, and perform the firmware upgrade through USB.

Solution 3 is definitively the easiest to implement, but it requires a human intervention over the device.

## 6.3 Reset configuration

When the M519-SUV is powered-up or its /RESET pin is risen (Low to High transition), the module resets and probes VCC, VBUS and /FLASH to read its reset configuration.

The reset configuration defines:

1. whether the host interface is the serial line or the USB bus,

| VBUS | VCC | Host interface |
|---|---|---|
| 5V | — | USB |
| unconnected | 5V | Serial |

Do not connect anything to either the J2 connector unless you want to use USB.

2. whether the host interface is the serial line or the USB bus,

| /FLASH | Firmware activation |
|---|---|
| LOW | M519 runs its bootloader |
| HIGH | M519 runs its firmware |

Leave the /FLASH pin unconnected if your application does not need to activate the bootloader.

## 6.4 Electrical level of the Serial interface

The hardware configuration (electrical levels) of the Serial interface is driven by register $_H$029F.

| Register $_H$029F | Serial interface | LED 1 | LED 2 | Remark |
|---|---|---|---|---|
| $_H$00 | RS-TTL | ON | ON | |
| $_H$01 | RS-232 | ON | OFF | |
| $_H$02 | RS-485 | OFF | ON | |
| *other* | RS-TTL | OFF | OFF | *Do not use* |

On startup, the two LEDs on the antenna show which configuration has been activated during. After ten seconds, both LEDs go OFF to reduce the overall power consumption.

## 6.5 Serial shell

The M519-SUV features a “human console” shell, that is available over its Serial interface

- whatever the operating mode,
- whether the host interface is Serial or USB.

The shell is namely used to explore or fine-tune the M519-SUV, or debug the system.

The communication parameters are:

- Baudrate: 38400bps,
- Format: 8 data bits, 1 stop bit, no parity, no flow control.

Note that if the shell is always enabled with these parameters when the device resets, it is generally disabled once the host has activated another protocol or changed the communication parameters. Always reset the device before trying to use the shell.

Send *<CR><LF>* to get the prompt of the device, that exposes its version and current operating mode.

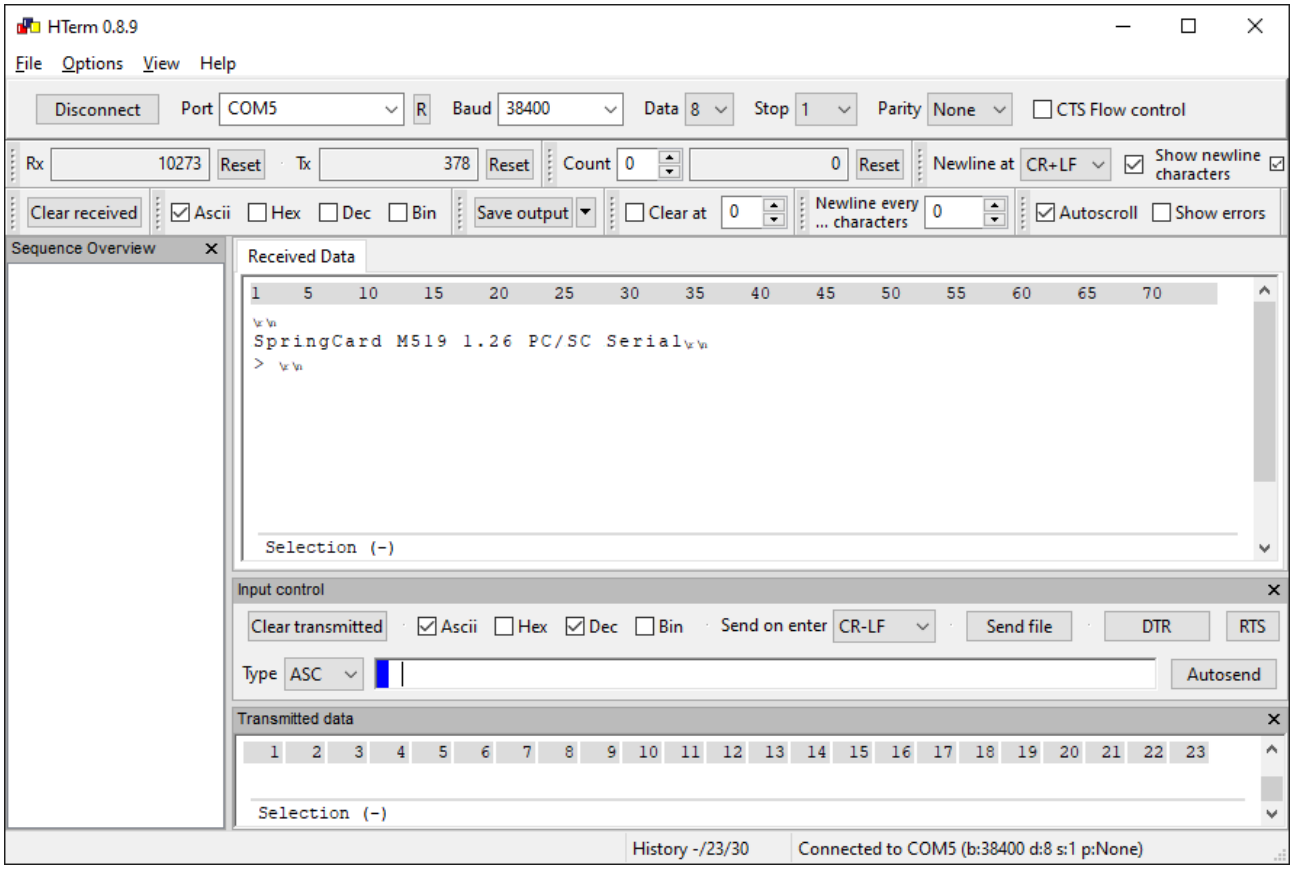



*Figure 6: Terminal application 'HTerm' showing the M519's prompt*

In the snapshot above,

- “v1.26” is the version number of the firmware. Newer firmwares will show another version number,

- "PC/SC" is the operating mode. Other possible values are "Direct", "SmartReader", "RFIDScan" and "Legacy",
- "Serial" is the primary host interface. Other possible value is "USB".

Enter `help<CR><LF>` to read the list of available commands.

REMARK

The prompt does not reflect the fact that the device is a M519-SUV. The firmware is generic and its prompt is "M519" in all situations.

## 6.6 Configuring the M519-SUV

The M519-SUV is highly configurable. You may download a configuration file from SpringCard Companion, or use a configuration file provided by SpringCard or a 3rd party. Configuration files use either the JSON or the CFG format.

The new configuration may be injected into the M519-SUV by the mean of a software provided by SpringCard, by the host application itself, or using a secure configuration card *(Master Card)*. When exploring/evaluating the M519-SUV, it is also possible to edit the configuration with the shell.

### Warning

Writing an invalid configuration may disable the Serial interface and/or the USB interface, and prevent any communication with the device. Pay a particular attention to register H029F since it drives the electrical level of the Serial interface.

### 6.6.1 SpringCard configuration software

When the host uses the USB interface, in-the-field, Live configuration by SpringCard-provided software is always possible, because the Direct protocol is always available (at least as a secondary profile in the compound device).

To write a new configuration over USB, use either:

- SpringCoreConfig, a command line tool (member of the SpringCore Tools suite) that is easy to deploy and run unattended

  https://www.springcard.com/en/download/find/file/sq20029

- SpringCard Companion, a web application + local service, for a more user-friendly experience.

  https://companion.springcard.com

### 6.6.2 Configuration through Master Cards

*This feature is not yet implemented in the current version of the firmware.*

### 6.6.3 Configuration using shell commands

In the Serial shell (see § 30),

- Send command `cfg<CR><LF>` to read the current configuration,
- Send command `cfgXX=YYYY<CR><LF>` to write value $_H$YYYY in configuration register H02XX,
- Send command `cfgXX=<CR><LF>` to erase configuration register $_H$02XX (the default value will apply).

### 6.6.4 Configuration by the host application

The CONTROL class of the Direct protocol, that is also available through the SCardControl function of the PC/SC Coupler mode, let the host application configure the M519-SUV.

For reference, please read:

https://docs.springcard.com/books/SpringCore/Host_Protocols/Direct_Protocol/CONTROL_class/index

## 6.7 Identifying the configuration of an unknown M519-SUV

The great versatility of the M519-SUV is a key feature that allows a system integrator to use the same device in the widest range of solutions. SpringCard offers a wide range of part numbers so that any manufacturer can order and provision the M519-SUV in the exact configuration required by their product line, but the device can also be easily reconfigured in the field.

However, this nice feature comes with its side-effect: it makes it difficult for an R&D engineer or a maintainer working on a device that has been deployed in-the-field or used by another developer to guess how the device has been configured, and therefore how it should be connected and operated. Of course the label printed under the M519-SUV tells the out-of-factory configuration, but the current configuration of the device may be different.

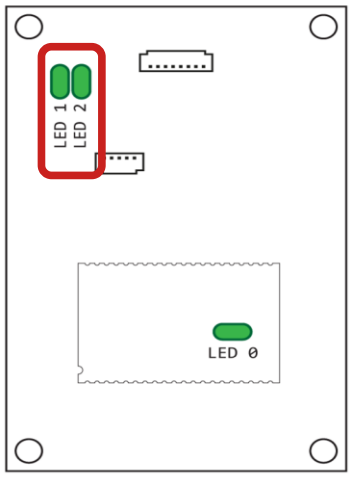
In this section we'll learn to identify the main settings.

### 6.7.1 Identifying the electrical level of the Serial interface

If your intention is to use the M519-SUV through its USB interface only, you may skip this section and proceed with § 35.

The electrical level of the Serial interface of the M519-SUV depends on its configuration. The out-of-factory configuration (RS-TTL, RS-232 or RS-485) is specified by the order code and printed on the label, but the actual configuration may be different if the device has been re-configured in-the-field (register $_{H}$029F).

Observe the LED sequence when the device is starting up to know its current configuration.



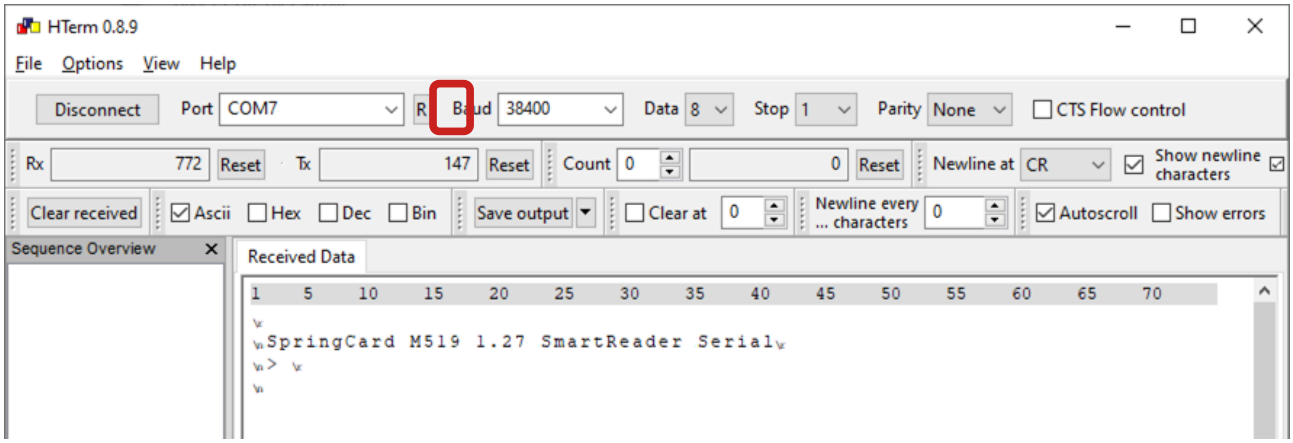

On startup, the two LEDs on the antenna show which configuration has been activated during. After ten seconds, both LEDs go OFF to reduce the overall power consumption.

| LED 1 | LED 2 | Serial interface | Remark |
|---|---|---|---|
| ON | ON | RS-TTL | |
| ON | OFF | RS-232 | |
| OFF | ON | RS-485 | |
| OFF | OFF | RS-TTL | Pre-delivery mode, not observable in-the-field |

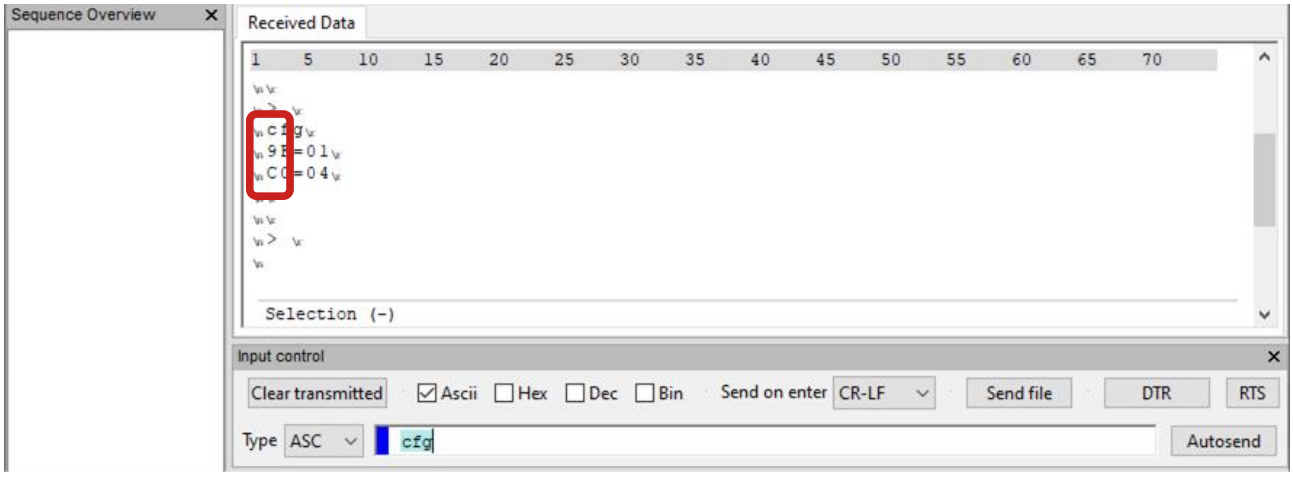
### 6.7.2 Identifying the operating mode, Serial interface

Connect the M519-SUV to the host using the RS-232 interface. Open a shell onto the device.

Send *<CR><LF>* to get the prompt of the device, that exposes its version and current operating mode.




It is also possible to read-back the configuration by entering the *cfg* command:

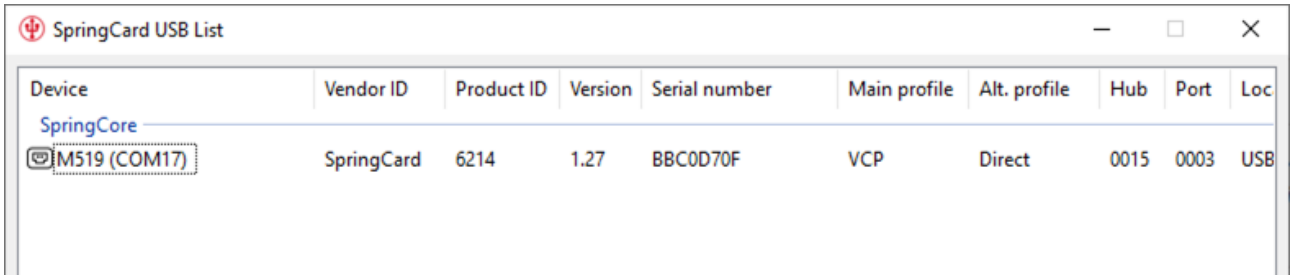


### 6.7.3 Identifying the operating mode, USB interface

The easiest way of identifying the operating mode of a SpringCard USB device is to download SpringCard UsbList software (for Windows). Look for “USB List” in this download page:

https://www.springcard.com/en/download/find/file/usblist

Launch the software and connect the M519-SUV to the host through its USB interface (see §Erreur ! Signet non défini.). Look for the (newly added *M519* entry in the list of devices, and read its USB Product ID (3rd column).




The mode can be deduced from the Product ID (see §38):

| Product ID | Operating mode |
|---|---|
| H6212 | PC/SC Coupler |
| H6213 | RFID Scanner |
| H6214 | Smart Reader |
| H6211 | SpringProx Legacy |

## 6.8 Out-of-factory configurations

The table below list the settings associated to every order code.

| | | | | Key configuration registers | | |
|---|---|---|---|---|---|---|
| Order code | Serial interface | Mode | Options | $_H$02C0 | $_H$029F | $_H$02A6 |
| SC23219 | RS-TTL | PC/SC | | $_H$02 | $_H$00 | — |
| SC23220 | RS-232 | PC/SC | | $_H$02 | $_H$01 | — |
| SC23222 | RS-TTL | RFID Scanner | QWERTY | $_H$03 | $_H$00 | $_H$00 |
| SC23221 | RS-TTL | RFID Scanner | AZERTY | $_H$03 | $_H$00 | $_H$01 |
| SC23223 | RS-TTL | Smart Reader | Protocol MK1 | $_H$04 | $_H$00 | — |
| SC23224 | RS-232 | Smart Reader | Protocol MK1 | $_H$04 | $_H$01 | — |
| SC23225 | RS-485 | Smart Reader | Protocol MK1 | $_H$04 | $_H$02 | — |
| SC23226 | RS-TTL | SpringProx Legacy | | $_H$01 | $_H$00 | — |
| SC23227 | RS-232 | SpringProx Legacy | | $_H$01 | $_H$01 | — |

# 7 Integration guide

The M519-SUV supports either USB or Serial as host interface.

The USB interface is enabled by connecting the device to a USB host through the J2 connector.

If VBUS is not active when the module starts, then the Serial interface is enabled.

Warning

For debugging and manufacturing reasons, the Serial interface remains active when the USB interface is enabled, but using both interfaces at the same time is absolutely not supported and is likely to make the M519-SUV reset or behave incorrectly. Leave J1 unconnected when J2 is connected.

## 7.1 USB

### 7.1.1 Overview

The M519-SUV is a USB 2.0 full-speed (12Mbps) device, compliant with USB 3. Connect the device to the host through J2 (JST-5) to activate its USB interface.

### 7.1.2 Precautions

- Use only the JST-5 USB cord provided by SpringCard, or a specific cord explicitly developed and/or validated by SpringCard for your target system,
- Fully extend the USB cord to avoid inductive coupling,
- Place the USB cord perpendicular to the antenna, ensuring that no segment of the cable runs parallel to any side of the antenna,
- Connect the product directly to the computer or exclusively through a USB-certified hub with external power supply; never use a hub that is only powered by the bus; avoid low-quality hubs that have unstable supply voltage or poor noise immunity.

### 7.1.3 USB IDs, profile and descriptors

The USB profile —and therefore the Product ID and the descriptors announced by the device— depends on the operating mode as selected by configuration register $_H$02C0.

| Operating mode | Vendor ID | Product ID | Profile |
|---|---|---|---|
| SpringCore Direct[A] | $_H$1C34 | $_H$6210 | WinUSB |
| SpringProx Legacy[B] | $_H$1C34 | $_H$6211 | Compound, CDC-ACM + WinUSB |
| PC/SC Coupler[C] | $_H$1C34 | $_H$6212 | Compound, CCID + WinUSB |
| RFID Scanner[D] | $_H$1C34 | $_H$6213 | Compound, HID keyboard + WinUSB |
| Smart Reader[E] | $_H$1C34 | $_H$6214 | Compound, CDC-ACM + WinUSB |
| PC/SC Coupler, no Direct interface[F] | $_H$1C34 | $_H$621A | CCID |

A: selected by $_H$02C0=$_H$00 and $_H$02C1.bit3=$_B$0
B: selected by $_H$02C0=$_H$01 and $_H$02C1.bit3=$_B$0
C: selected by $_H$02C0=$_H$02 and $_H$02C1.bit3=$_B$0
D: selected by $_H$02C0=$_H$03 and $_H$02C1.bit3=$_B$0
E: selected by $_H$02C0=$_H$04 and $_H$02C1.bit3=$_B$0
F: selected by $_H$02C0=$_H$02 and $_H$02C1.bit3=$_B$1

## 7.2 Serial

### 7.2.1 Overview

The serial interface of the M519-SUV is enabled by leaving J2 unconnected, and by powering the module by the mean of a unique 5V or 3.3V supply over VCC on J1.

The electrical interface (RS-TTL, RS-232 or RS-485) is selected by register $_H$029F (see § 6.4).

#### Warning

Check the module configuration before connecting it; observe LED1 and LED2 to determine the selected electrical interface. Connecting the M519-SUV to a communication port that uses different electrical levels may damage the M519-SUV and/or the host.

### 7.2.2 Precautions

- Use short cables and genuine JST male connectors,
- Fully extend the Serial cord to avoid inductive coupling,
- Place the Serial cord perpendicular to the antenna, ensuring that no segment of the cable runs parallel to any side of the antenna.

## 7.3 Electromagnetic environment

### 7.3.1 Overview

Communication between the M519-SUV and the contactless target (proximity/vicinity card, RFID label, NFC tag, NFC object…) uses an alternating magnetic field at 13.56MHz and rely on the principle of a transformer (inductive coupling). At the exception of smartphones and battery-powered NFC objects, the M519-SUV also provides remote power to the target.

Such a system is largely impacted by its electromagnetic environment:

- RF waves radiated by surrounding electronics parts, cables or PCB traces alter the signal over noise ratio. This is particularly the case with some displays that radiate their 27.12MHz clock frequency (2x13.56MHz) or with badly shielded USB devices that radiate side-bands around 12MHz.
- RF waves cannot cross conductive materials (PCB ground plane, metallic shield or shell…). More than that, conductive materials in the nearby will host eddy currents (Foucault's currents) so that the RF field will be wasted in heating the material instead of providing power to the contactless objects.

### 7.3.2 Precautions

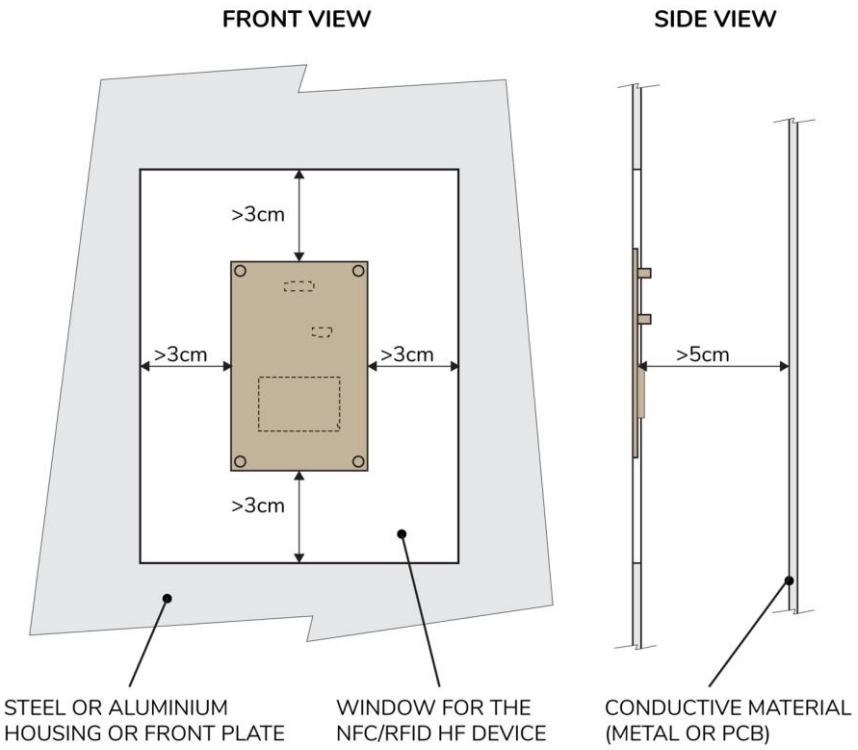
To ensure optimal functionality of the M519-SUV, adherence to the following installation guidelines is crucial.

SpringCard has a strong experience in designing NFC/RFID HF devices. Feel free to contact us should you need the assistance of an expert in validating your design.

#### *7.3.2.1 Distance to conductive materials*

Avoid any conductive materials in close proximity to the antenna's front. When embedding the M519-SUV' antenna within a metallic body, such as kiosks or gates, create an opening leaving at least 30mm between the antenna and the metal at the sides, and at least 50mm at the back.

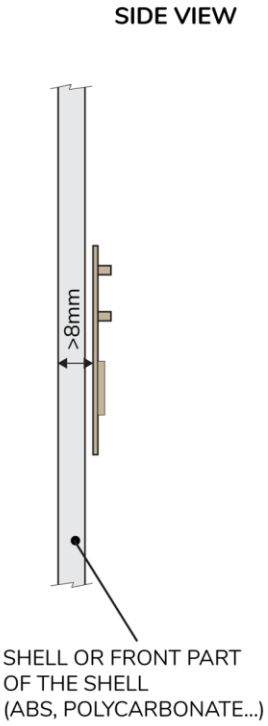
Conductive elements, including PCBs featuring ground planes or parallel traces, as well as cables, must also not be situated near the antenna. Maintain a minimum clearance of 30mm around the periphery of the antenna and ensure a minimum clearance of 50mm at the back of the antenna.



#### *7.3.2.2 Housing*

The permeability factor (μ) of materials placed in front of the antenna should be considered. For instance, the permeability of glass or glass-filled plastics have a permeability differing from that of vacuum ($\mu_0$) and air.

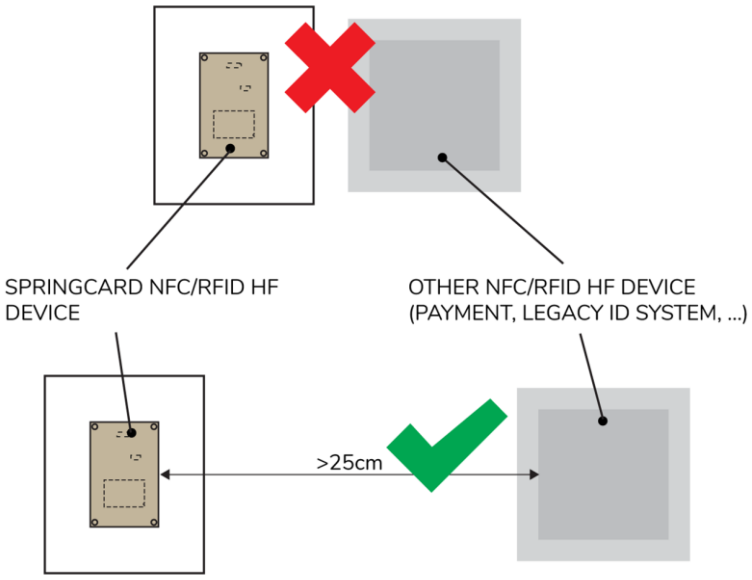
Design the housing or the product shell so a clearance of about 8mm is maintained between the card and the antenna. This enhance the interaction with poorly-tuned cards or with NFC smartphones that expose a significant static load (loading effect) to the base-station.



#### *7.3.2.3 Electromagnetic environment*

The vicinity should be free of other devices that emit radio frequencies.

A minimum distance of 250mm is advised from any adjacent contactless readers or couplers operating at 13.56MHz. Implement appropriate measures as to mitigate radiated noise within the 12 to 16MHz frequency band.

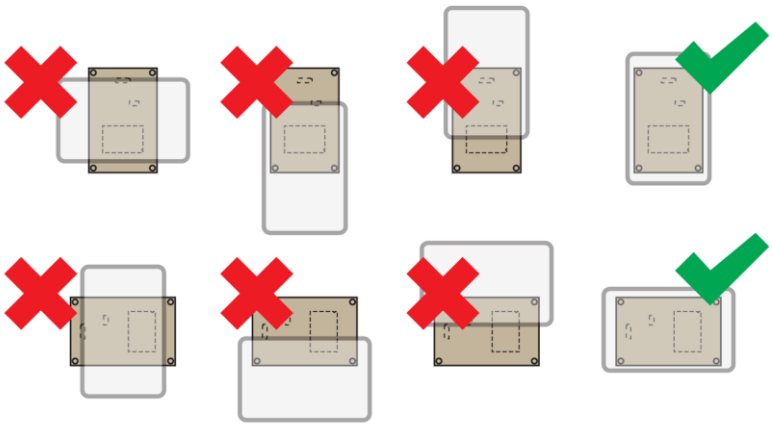



### 7.3.3 Improving the user experience

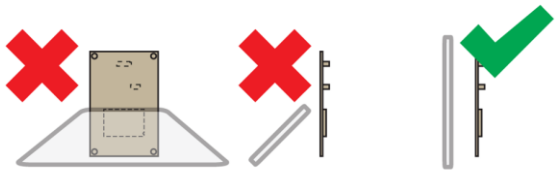
It is advisable to pay particular attention to the design of the system to ensure that the end-user places the card correctly in front of the antenna.

This is even more important when using an antenna that has a preferred direction because it is rectangular (like the default 69x45 antenna of the M519-SUV), rather than square or round.

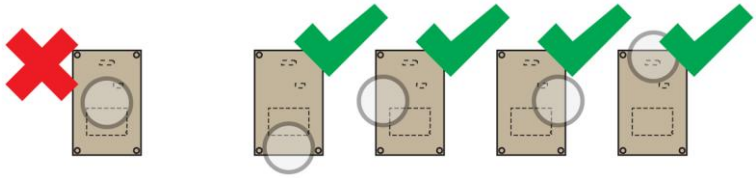
- Optimal performance is achieved when the card is aligned parallel to the antenna, with the longest sides corresponding. Take the user gesture in account when designing the product, and/or position a pictogram over the antenna to indicate the preferred card orientation.



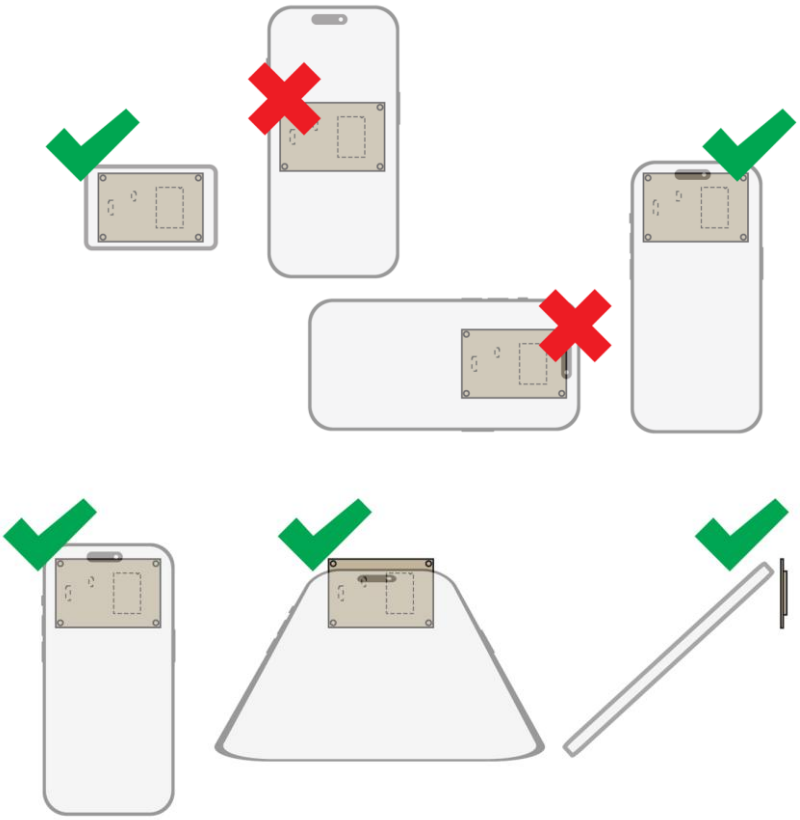
- Ensure the design intuitively guides the user to position the card parallel to the antenna, avoiding any angled placement for consistent communication.



- Smaller NFC tags or RFID labels usually achieve better performance when positioned near the coil's edge as opposed to the antenna's central axis.



- Most NFC smartphones have a preferred axis that is different from that of standard cards, and many are now engineered to communicate more effectively when held at an angle to the antenna.



- Ideally, the end-user shall be made aware of all these particularities via strategically positioned labels, detailed documentation, or targeted educational content to ensure the best possible user experience.

## 7.4 Testing, validation and qualification

EMC product qualification is imperative for compliance with CE, FCC markings, and other regulatory standards, and must be carried out on the final product. Although the M519-SUV is qualified as an independent device, the performance of the NFC/RFID HF and USB interfaces is contingent upon the integration's quality.

Furthermore, certifications or approvals that are specific to an application field or the targeted use-case (including those from NFC Forum, EMVCo, CEN/TS 16794, ISO/IEC TS 24192, RCTIF etc), must be obtained over the assembled product. This typically encompasses the full software solution and the device's actual configuration in-situ.

SpringCard has a strong experience in designing NFC/RFID HF devices. Feel free to contact us should you need the assistance of an expert in designing or qualifying your own product.

SpringCard SAS
Société par Actions Simplifiée au capital de 227 000 €
RCS Évry B 429 665 482
Parc Gutenberg, 2 voie La Cardon
91 120 Palaiseau — FRANCE

www.springcard.com