



M519 Data Sheet and Integration Guide

© SpringCard, all rights reserved

Category
Datasheets
Family
SpringCore / M519

Reference
PFT22217
Diffusion
Public
Date
17/09/2025

Version
BA
State
Approved
Page count
56

REVISION HISTORY

Ver.	Date	Author	Verifier	Approb.	Details
00	07/11/2022	JDA			Draft preview
AA	16/08/2023	JDA	CFE	JDA	First public release (but still with a lot of <i>TBD</i>)
AB	18/09/2023	JDA	CFE		Added reference schematics in chapters 7, 8 and 9
AC	29/11/2023	JDA			Temperature range edited
AD	01/02/2024	JDA			Format of tape and reel edited, order code updated
AE	11/04/2024	JDA			Added the links to the online documentation of the Smart Reader protocols, fixed a reference error in § 8.4, overall improvements in chapter 8
AF	18/04/2024	JDA			Format of tape and reel edited, MOQ updated, following a change in provider specification
AG	27/11/2024	JDA			Changed max value for VIN_RF in § 5.2.1, recommended diode is now B260A (instead of B230A)
AH	14/01/2025	JDA			Completed current measurements in § 5.2.2
BA	17/06/2025	JDA			New template applied, new format OpenOffice → Word Corrected max height of module in § 3.1 and 4.1 (V-cut tolerance) Dropped support of TDA8026 smart card interface, since NXP has discontinued the IC Completed the various TDA8035 implementations in § 9.2

TABLE OF CONTENT

1	Introduction.....	6
1.1	Overview	6
1.2	Related documents	6
1.2.1	Documents available as PDF	6
1.2.2	Online materials	6
1.3	Order codes	6
2	About the SpringSeed M519	7
2.1	General description	7
2.2	Features and benefits.....	7
2.2.1	Ease of integration into any machine or assembly.....	7
2.2.2	Ease of integration for any application and use case	8
2.2.3	Open and interoperable	8
2.3	Typical applications.....	8
2.4	Integration and development.....	9
3	Technical data	10
3.1	General.....	10
3.2	NFC/RFID HF (contactless) interface.....	10
3.3	Card emulation and peer-to-peer operations.....	11
3.4	Smart card (contact) Interface.....	11
3.5	Host interfaces.....	11
3.5.1	USB.....	11
3.5.2	Serial	12
4	Mechanical specifications and pinouts.....	13
4.1	Dimensions	13
4.2	Pinout	14
4.2.1	West-bound header	15
4.2.2	East-bound header	16
4.3	Recommended PCB layout.....	17
4.4	Tape and reel packaging.....	18
5	Electrical specifications	19
5.1	Limiting values.....	19
5.2	Power.....	20
5.2.1	Power supply voltage.....	20
5.2.2	Current consumption	20
5.3	VOUT pin.....	21
5.4	SC_VCC pin.....	21
5.5	/RESET, /SUSPEND, /SC_IRQ pins.....	21
5.6	/FLASH, MODE0, MODE1 pins.....	21
5.7	RX, PER_RX pins	22
5.8	TX, PER_TX, /PER_EN, /WAKEUP, SC_RST, SC_CLK, I2C_SCL pins.....	22
5.9	SC_IO, /PER_CS, PER_CLK, I2C_SDA pins.....	22
5.9.1	As input	22
5.9.2	As output	22
5.10	GPIO1, GPIO2, GPIO3, GPIO4, GPIO5 pins	23

5.10.1	As input.....	23
5.10.2	As output.....	23
6	Application information	24
6.1	Operating modes, profiles and protocols.....	24
6.1.1	PC/SC Coupler mode	24
6.1.2	Smart Reader mode	25
6.1.3	Direct protocol and fail-safe mode.....	26
6.1.4	SpringProx Legacy.....	26
6.2	Firmware upgrade	27
6.2.1	Live upgrade	27
6.2.2	Bootloader upgrade	28
6.3	Reset configuration	29
6.4	Serial shell	31
6.5	Configuring the M519	32
6.5.1	SpringCard configuration software	32
6.5.2	Configuration through Master Cards	33
6.5.3	Configuration using shell commands	33
6.5.4	Configuration by the host application	33
7	Integration guide – Host interface and power supply	34
7.1	USB.....	34
7.1.1	Overview.....	34
7.1.2	Reference schematics	35
7.1.3	Precautions.....	35
7.1.4	USB IDs, profile and descriptors	37
7.2	Serial.....	37
7.2.1	Overview.....	37
7.2.2	Reference schematics	38
7.2.3	Validation of the implementation.....	38
8	Integration guide – RF antenna	39
8.1	Static matching circuit.....	40
8.1.1	Principle.....	40
8.1.2	Reference schematics	40
8.2	“Auto-tune” matching circuit.....	41
8.2.1	Principle.....	41
8.2.2	Reference schematics	41
8.3	Reference PCB layouts and BOMs.....	42
8.3.1	69x45 balanced antenna.....	43
8.3.2	25x25 balanced antenna.....	44
8.3.3	80x80 balanced antenna.....	45
8.4	Adapting the M519 to its actual antenna	47
8.4.1	Setting the antenna configuration by software.....	47
8.4.2	Setting the antenna configuration manually	47
8.5	Supporting other antennas.....	48
9	Integration guide – Contact smart card.....	49

- 9.1 Single slot, direct access 50
 - 9.1.1 Overview 50
 - 9.1.2 Guidelines 50
 - 9.1.3 Reference schematics 50
 - 9.1.4 Firmware implementation / PC/SC readers 51
 - 9.1.5 Card tracking 51
- 9.2 ID-000 or ID-1 slot(s) using TDA8035 52
 - 9.2.1 Guidelines 52
 - 9.2.2 Reference schematics 53
 - 9.2.3 Firmware implementation / PC/SC readers 54
 - 9.2.4 Card tracking 54

1 Introduction

1.1 Overview

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

The M519 is a OEM NFC/RFID HF module. It must be associated with an external antenna.

1.2 Related documents

1.2.1 Documents available as PDF

Reference	Title / Description
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

Order code	Product	MOQ
SC23218	M519	1 reel: 300 pieces Cut reel: 50 pieces

2 About the SpringSeed M519

2.1 General description

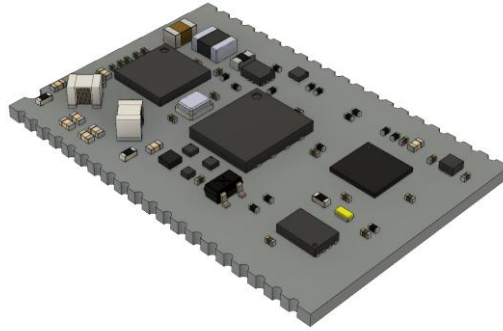


Figure 1: 3D view of the M519

The SpringCard SpringSeed M519 is a versatile OEM NFC/RFID HF module.

The M519 is designed to be used in any industrial equipment or consumer device, and has to be connected to an external antenna through a “mother board” PCB.

The M519 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 M519 features a contact smart card interface, that is able to drive either one SAM card (ID-000) either directly, or to drive up to 5 ID-1 and ID-000 slots through NXP TDA8035 ICs.

The M519 also features a protected storage for secret and private cryptographic keys, and is able to run secure transactions protected by AES or ECC schemes with contactless cards and NFC smartphones.

The M519 is either a Serial device or a USB device; it operates either in Coupler mode (reader/writer, APDU level) or in Smart Reader mode, freeing the host from all technical aspects of the transaction with the cards.

The M519 may either be surface-mounted on its “mother board” PCB using the edge-plated pads, or be soldered on 0.50" (1.27mm) headers using Ø0.6 half-holes on the outline.

2.2 Features and benefits

2.2.1 Ease of integration into any machine or assembly

- Smallest design on the market (26.67 x 17.78 x 2.6 mm or 1.05 x 0.7 x 0x10 "),
- Components mounted on the top side only, bottom pads and castellated edges ready for either SMD or THT mounting,
- NXP PN5190 NFC/RFID HF frontend allowing best in-class performance/power ratio,

- Self-antenna tuning capability,
- Single-source power supply, power saving modes, low power card detection features on less than 5µA,
- ISO 7816 interface supporting up to 4 external SIM/SAM (ID-000) slots and 1 ID-1 smart card slot.

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 module even from low-end MCUs,
- 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 tested¹ 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.

¹ Tests carried out in-house do not prejudice 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 Integration and development

The M519-SRK is a complete Starter Kit for customers who want to evaluate the M519 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. See [PMD23175] for details.

The SpringSeed M519 OEM module is also the core component of the SpringCard M519-SUV, a ready-to-use NFC/RFID HF OEM product (module + antenna with USB and Serial connectors). See [PMI23209] for details.

3 Technical data

3.1 General

Dimensions	27.07 x 17.78 x 2.6 mm
Weight	Approx. 7g
Power supply	3.3V nominal for Serial operation 5V (powered by the bus) for USB operation
Power requirement	Nominal: 200mA Max: 400mA Power saving: < 1mA
Temperature range	Operation: -40°C to +85°C Storage: -40°C to +85°C
Environment	Humidity 0 – 90% (non-condensing)

3.2 NFC/RFID HF (contactless) interface

Carrier frequency	13.56MHz
Antenna requirement	External antenna, 20Ω-matched, balanced
Operating distance	Up to 150mm depending on antenna, card and environment
RF power	Max 2W (33dBm)
Supported standards	ISO/IEC 14443 A & B (PCD) / NFC-A and NFC-B ISO/IEC 15693 (VCD) and ISO/IEC 18000-3M1 / NFC-V ISO/IEC 18000-3M3 (RFID HF) / EPC HF JIS X 6319-4 / NFC-F ISO/IEC 21481 (NFCIP-2)
Bitrates	26, 53, 106, 212, 424, 848 kbit/s depending on the protocol
Supported card technologies (partial list) ^A	NFC Forum types 1, 2, 3, 4 and 5 Mifare Classic with CRYPTO1 All cards in NXP Mifare families: Mifare Plus, Mifare UltraLight, Desfire, All cards in NXP NTAG and ICODE families Innovatron (Calypso cards) STMicroElectronics SR & LR, ST25 Infineon SLE44, SLE66, SRF55 Texas Instrument Tag-it Sony FeliCa Lite Apple ECP, Apple VAS (PassKit NFC), Google Smart Tap and other NFC applications on smartphones Inside Secure PicoPass / HID iClass (ID only)
Advanced features	Low power card detection Automatic waveform control

Note A

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 ISO/IEC 18092 (NFCIP-1) active and passive, initiator and target
Bitrates	106, 212, 424 kbit/s depending on the protocol

3.4 Smart card (contact) Interface

Card power class	Direct on M519: 3.3V (class B) Through TDA8035: 1.8V, 3.3V or 5V (classes A, B, C)
Card clock	4 MHz
Supported standards	ISO/IEC 7816-3, 7816-4
Bitrates	106, 212, 424 kbit/s depending on the protocol

3.5 Host interfaces

3.5.1 USB

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

3.5.2 Serial

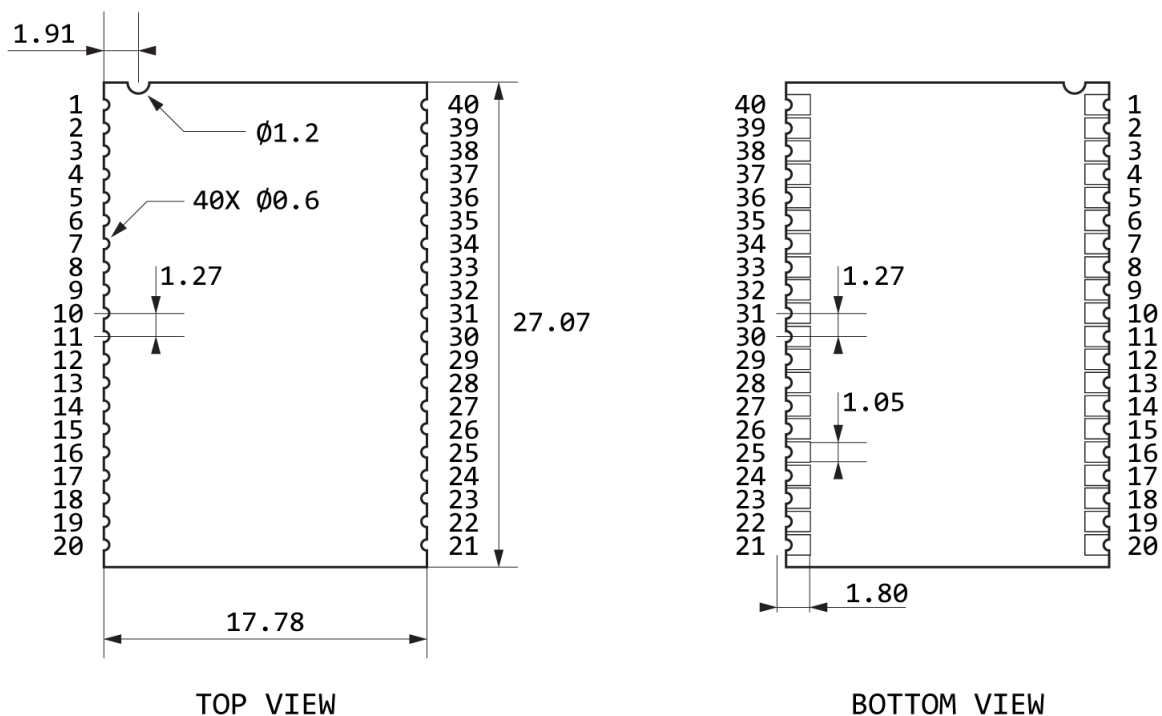
Physical interface	RX/TX @0/3.3V (compliant with TTL & CMOS peers)
Bitrate	38400bps (default), up to 500kbps after handshaking
Format	8 data bits, 1 stop bit, no parity, no flow control
Protocols	CCID over Serial SpringProx Legacy SpringCore Direct RDR MK1 \$SCRDR

4 Mechanical specifications and pinouts

4.1 Dimensions

The M519 is a single-sided PCB with dual castellated/through-hole pins around the edges. It is designed to be usable as a surface mount module as well as being in Dual Inline Package (DIP) type format, with the 40 main pins on a 1.27mm (0.05") pitch grid with 0.6mm holes.

A 1.2mm half-hole on the left-hand side of the north-bound edge acts as a poka yoke (polarizer) to let the pick and place machine know where pin #1 is.



ALL DIMENSIONS IN MILLIMETERS
PRINT VERSION NOT TO SCALE
(APPROX 2:1)
OUTLINE TOLERANCE $\pm 0.25\text{MM}$ - TOLERANCE ON DRILLING $\pm 0.05\text{MM}$
LOCATION OF PADS ACCORDING TO IPC-A-600 CLASS 2

Figure 2: Mechanical specifications

4.2 Pinout

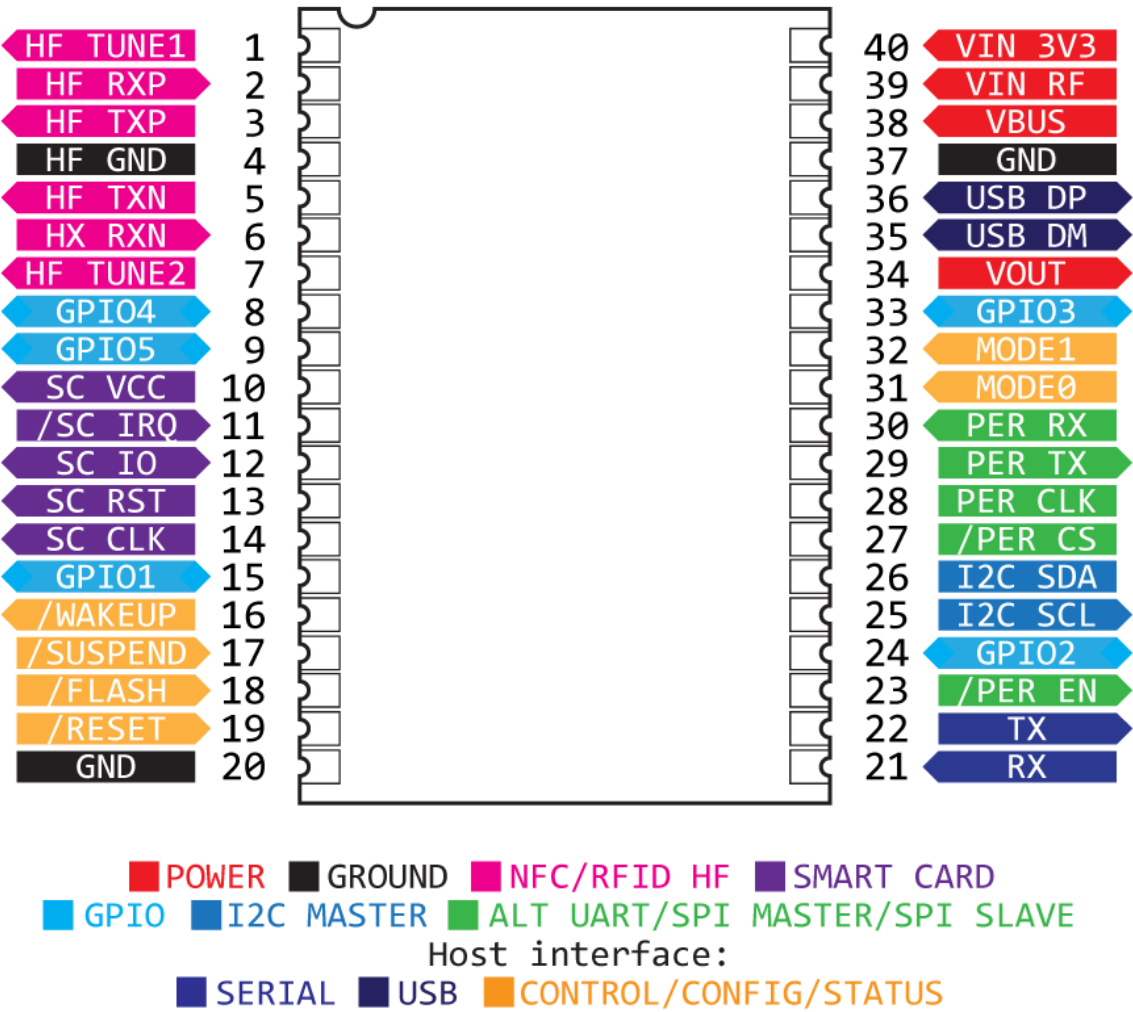


Figure 3: Pinout

4.2.1 West-bound header

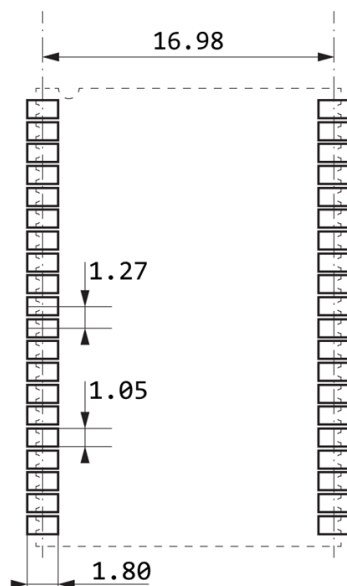
Pin	Symbol	Type	Description
#1	HF_TUNE1	Analog	Antenna self-tuning capacitor 1
#2	HF_RXP	Analog	Antenna receiver (plus side)
#3	HF_TXP	Analog	Antenna driver (plus side)
#4	HF_GND	Analog	Antenna ground
#5	HF_TXP	Analog	Antenna driver (minus side)
#6	HF_RXN	Analog	Antenna receiver (minus side)
#7	HF_TUNE2	Analog	Antenna self-tuning capacitor 2
#8	GPIO4	In/Out	
#9	GPIO5	In/Out	
#10	SC_VCC	Out	Card VCC signal
#11	/SC_IRQ	In	Card status / TDA8026 interruption
#12	SC_IO	In/Out	Card I/O signal
#13	SC_RST	Out	Card RST signal
#14	SC_CLK	Out	Card CLK signal (4MHz) / TDA8026 master clock (16MHz)
#15	GPIO1	In/Out	
#16	/WAKEUP	Out	Tell the host that the module is active
#17	/SUSPEND	In	Tell the module to enter standby mode
#18	/FLASH	In	Force DFU (bootloader) mode upon reset
#19	/RESET	In	Reset the module
#20	GND	Ground	

4.2.2 East-bound header

Pin	Symbol	Type	Description
#21	RX	In	Main UART, host to module
#22	TX	Out	Main UART, module to host
#23	/PER_EN	Out	Activate the peripherals
#24	GPIO2	In/Out	
#25	I2C_SCL	Out	I ² C SCL (master)
#26	I2C_SDA	In/Out	I ² C SDA (master)
#27	/PER_CS	In/Out	SPI chip select
#28	PER_CLK	In/Out	SPI clock
#29	PER_TX	Out	TX to peripherals
#30	PER_RX	In	RX from peripherals
#31	MODE0	In	Configure operating mode
#32	MODE1	In	Configure operating mode
#33	GPIO3	Out	
#34	VOUT	Out	3.3V output, max 100mA, to power the peripherals
#35	USB_DM	USB	USB D- signal
#36	USB_DP	USB	USB D+ signal
#37	GND	Ground	
#38	VBUS	USB	Digital power supply + USB presence, 5V
#39	VIN_RF	Power	Power supply for RF, either 3.3V or 5V
#40	VIN_3V3	Power	Digital power supply 3.3V

4.3 Recommended PCB layout

The following footprint is recommended for systems which will be reflow-soldering the M519 as a module.

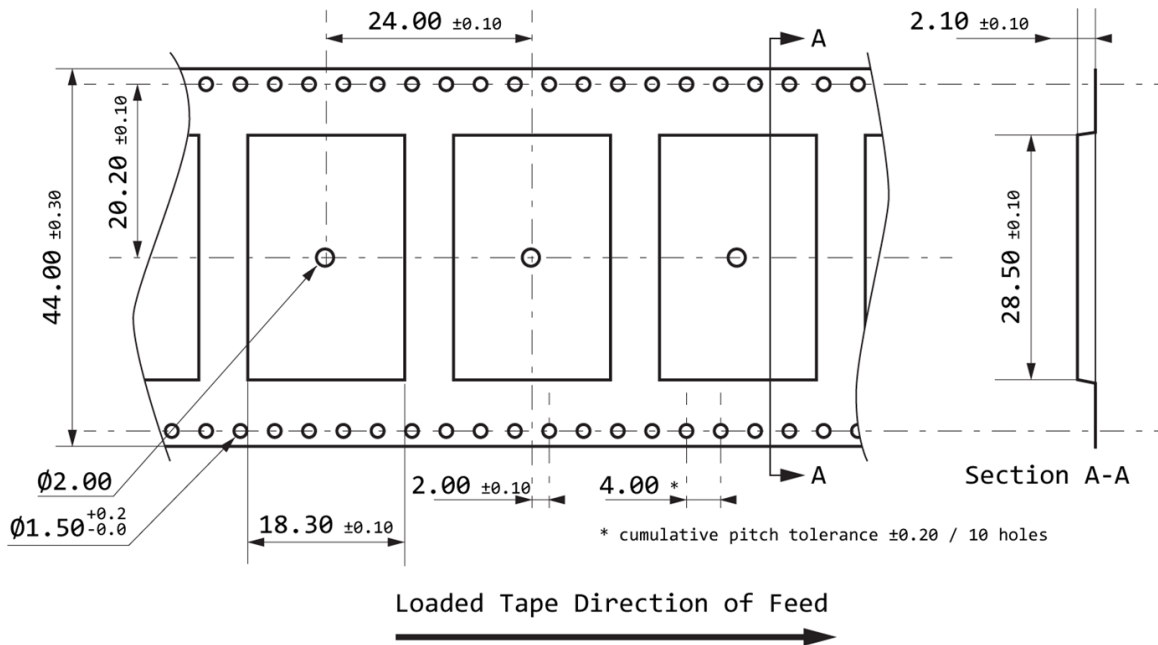


RECOMMENDED PCB LAYOUT

Figure 4: Recommended PCB layout

4.4 Tape and reel packaging

The M519 is delivered in Tape and Reel.



ALL DIMENSIONS IN MILLIMETERS
PRINT VERSION NOT TO SCALE
(APPROX 1:1)
MATERIAL: CONDUCTIVE POLYSTYRENE, BLACK
THICKNESS: 0.30 ± 0.05
CAMBER NOT TO EXCEED 1MM IN 250MM LENGTH
ONE 22 INCHES REEL = 40 METERS

Figure 5: Tape specifications

5 Electrical specifications

5.1 Limiting values

Symbol	Parameter	Conditions	Min	Max	Unit
VIN_3V3 _{ABS}	Supply voltage on pin VIN_3V3		0	3.6	V
VBUS _{ABS}	Supply voltage on pin VBUS		0	5.5	V
VIN_RF _{ABS}	Supply voltage on pin VIN_RF		0	5.5	V
V _{ANA}	Voltage on any analog pin		0	VIN_3V3	
V _{DIG}	Voltage on any digital pin		0	VIN_3V3 + 0.2	V
V _{ESD}	Electrostatic discharge voltage		-500	500	°C
T _{JUNCTION}	Junction temperature		—	+120	°C
T _{STORAGE}	Storage temperature		-40	+85	
VIN_3V3 _{ABS}	Supply voltage on pin VIN_3V3	No voltage applied	0	3.6	

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	Digital supply voltage, USB	4.5	5.0	5.2	V
VIN_RF _{VBUS}	RF supply voltage, USB	3.0	VBUS – 0.6	4.6	V

Warning

Pin #40 VIN_3V3 shall be left unconnected for USB operation.

5.2.1.2 Serial operation

Symbol	Parameter	Min	Typical	Max	Unit
VIN_3V3	Digital supply voltage, Serial	3.0	3.3	3.6	V
VIN_RF _{3V3}	RF supply voltage, Serial	3.0	VIN_3V3	4.6	V

Warning

Pin #38 VBUS 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,UNLOADED}	Digital power supply, USB, no power drain on VOUT, no smart card	—	25	35	mA
I _{VBUS,LOADED}	Digital power supply, USB, 100mA power drain on VOUT, smart card VCC active	—	230	240	mA
I _{VIN_RF,USB}	RF power supply, USB	—	110	300	mA

Warning

The USB 2.0 specification does not allow a device to drain more than 500mA. Keep this limit in consideration when designing the device: limit the RF power (by configuration, see point A, or by choosing a small antenna), accept only low power smart cards (class C / 1.8V, less than 50mA, and not class B / 3.3V nor class A / 5V), or configure the M519 so that the contactless slot is disabled when a contact card is inserted.

5.2.2.2 Serial operation

Symbol	Parameter	Min	Typical	Max	Unit
I _{3V3,UNLOADED}	Digital power supply, Serial, no power drain on VOUT, no smart card	—	35	50	mA
I _{3V3,LOADED}	Digital power supply, Serial, 100mA power drain on VOUT, smart card VCC active	—	240	260	mA
I _{VIN,RF,3V3}	RF power supply, Serial	—	115	310	mA

5.3 VOUT pin

Symbol	Parameter	Min	Typical	Max	Unit
VOUT	Peripheral(s) power supply voltage	3.0	3.3	3.6	V
I _{VOUT}	Peripheral(s) power supply current	—	—	100	mA

5.4 SC_VCC pin

Symbol	Parameter	Min	Typical	Max	Unit
SC_VCC	Smart card power supply voltage	3.0	3.3	3.6	V
I _{SC_VCC}	Smart card power supply current	—	—	100	mA

5.5 /RESET, /SUSPEND, /SC_IRQ pins

These digital input pins have permanent internal pull-up resistors.

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

5.6 /FLASH, MODE0, MODE1 pins

These digital input pins have internal pull-up resistors that are enabled at boot time, but disabled once the firmware is running.

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	0.1	165	μA
I _{IH}	High level input leakage current	-1	±0.01	1	μA

5.7 RX, PER_RX pins

These digital input pins have no internal pull-up resistors.

Symbol	Parameter	Min	Typical	Max	Unit
V _{IL}	Low level input voltage	—	—	1.0	V
V _{IH}	High level input voltage	2.4	—	VCC	V
I _L	Input leakage current	-1	±0.01	1	µA

5.8 TX, PER_TX, /PER_EN, /WAKEUP, SC_RST, SC_CLK, I2C_SCL pins

These are digital output pins.

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

5.9 SC_IO, /PER_CS, PER_CLK, I2C_SDA pins

These pins are digital input and output; when used as input, they have no internal pull-up resistors.

5.9.1 As input

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 _L	Input leakage current	-1	±0.01	1	µA

5.9.2 As output

Symbol	Parameter	Min	Typical	Max	Unit
V _{OL}	Low level output voltage	I _{OL} =1mA	0	—	0.6
V _{OH}	High level output voltage	I _{OH} =1mA	2.6	3.0	3.3
I _O	Output current		—	—	±2
SR	Up or down transition time		—	—	0.04

5.10 GPIO1, GPIO2, GPIO3, GPIO4, GPIO5 pins

These pins are digital input and output; when used as input, they do have internal pull-up resistors.

5.10.1 As input

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	—	± 0.01	± 1	μA

5.10.2 As output

Symbol	Parameter	Min	Typical	Max	Unit
V_{OL}	Low level output voltage	$I_{OL}=1\text{mA}$	0	—	0.6
V_{OH}	High level output voltage	$I_{OH}=1\text{mA}$	2.6	3.0	3.3
I_O	Output current		—	—	± 2
SR	Up or down transition time		—	—	0.04

6 Application information

6.1 Operating modes, profiles and protocols

The M519 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 choice of the operating mode and of the host interface is done upon reset (see § 28).

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 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 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 `_H02` into configuration register `_H02C0`, or by driving `MODE0=HIGH`, `MODE1=LOW` on reset.

6.1.1.1 USB

When the PC/SC Coupler mode is selected and the host interface is USB, the M519 is a compound device that exposes both the CCID profile and the SpringCore Direct profile (*more on the Direct profile in § 6.1.3*).

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 uses the CCID over Serial protocol.

See Application Note [PNA23174] for reference.

6.1.2 Smart Reader mode

In this mode, the M519 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 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 `03` into configuration register `02C0`, or by driving `MODE0=LOW`, `MODE1=HIGH` on reset.

6.1.2.1 USB

When the Smart Reader mode is selected and the host interface is USB, the M519 is a compound device that exposes both the HID keyboard profile and the SpringCore Direct profile (*more on the Direct profile in § 6.1.3*).

Thanks to the standard HID keyboard profile, the M519 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 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 sends the data collected from NFC/RFID tags over its serial line, using the protocol selected in configuration register `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 Direct protocol and fail-safe mode

SpringCore Direct is a proprietary protocol for new SpringCard devices that allows to flash, configure and use a device with more flexibility and faster than other protocols.

Most SpringCard software, like the SpringCore Tools and SpringCard Companion, rely on the SpringCore Direct protocol to access the devices in a seamless way, whatever their host interface and operating modes are. This protocol also allows an application to have full control over the reader, for advanced Smart Reader operation.

In the M519, the Direct protocol is associated to a fail-safe mode: the device bypass all configuration settings that may disable an interface or prevent communication, and disables most power sinks, to help debugging or fine-tuning the system.

The SpringCore Direct protocol and the fail-safe mode are selected by writing H00 into configuration register H02C0, or by driving MODE0=LOW, MODE1=LOW on reset.

6.1.4 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 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 `01` into configuration register `02C0`.

6.1.4.1 USB

When the SpringProx Legacy is selected and the host interface is USB, the M519 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 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.

6.1.4.2 Serial

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

6.2 Firmware upgrade

The M519 features a bootloader that allows to change its firmware. There are two ways to perform the upgrade: in-the-field, over a live system (Live upgrade) or after having explicitly activated the bootloader.

6.2.1 Live upgrade

6.2.1.1 USB

When the device is connected to the host through its USB interface, upgrading the firmware of the M519 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 is operating “normally” (this new firmware is written in a temporary storage area),
- When the M519 resets, its bootloader copies the new firmware from the storage area to the ROM of the microcontroller,
- The M519 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>

6.2.1.2 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. Consider implementing the Bootloader upgrade instead.

6.2.2 Bootloader upgrade

There are 2 different ways to start the device in Bootloader mode,

1. Send the software RESET command to the device, with `hFA hDA` 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

6.2.2.1 USB

The Bootloader uses the same Direct protocol mode as in § 6.2.1.1, but with a command-set limited only to firmware upgrade-related commands. The same tools may be used.

6.2.2.2 Serial

The Bootloader uses the same Direct protocol mode as in § 6.2.1.1, with a command-set limited only to firmware upgrade-related commands, but over the Serial interface.

6.3 Reset configuration

When the M519 is powered-up or its /RESET pin is risen (Low to High transition), the module resets and probes VIN_3V3, VBUS, /FLASH, MODE0 and MODE1 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	3.3V	Serial

Do not connect anything to VBUS 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.

3. when the main firmware is activated, which operating mode shall be selected,

MODE0	MODE1	Operating mode
HIGH	HIGH	As set in non-volatile memory (configuration register $\mu 02C0$)
HIGH	LOW	PC/SC Coupler
LOW	HIGH	RFID Scanner (USB) or Smart Reader (Serial)
LOW	LOW	SpringCore Direct, fail-safe mode

You may leave the MODE0 and MODE1 pins unconnected if your application does not need to change the mode.

The table below summarizes all the supported reset configurations and the associated USB profiles or Serial protocol:

VBUS	VIN_3V3 /FLASH	MODE0	MODE1	Interface	Operating mode	Profile/Protocol	
5V	—	LOW	—	—	USB	Bootloader	Direct (DFU)
5V	—	HIGH	HIGH	HIGH	USB	Set by configuration register H02C0	
5V	—	HIGH	HIGH	LOW	USB	PC/SC Coupler	CCID+Direct
5V	—	HIGH	LOW	HIGH	USB	RFID Scanner	HID keyboard+Direct
5V	—	HIGH	LOW	LOW	USB	Direct, fail-safe	Direct
N/C	3.3V	LOW	—	—	Serial	Bootloader	Direct (DFU)
N/C	3.3V	HIGH	HIGH	HIGH	Serial	Set by configuration register H02C0	
N/C	3.3V	HIGH	HIGH	LOW	Serial	PC/SC Coupler	CCID+Console
N/C	3.3V	HIGH	LOW	HIGH	Serial	Smart Reader	\$\$SCRDR+Console
N/C	3.3V	HIGH	LOW	LOW	Serial	Direct, fail-safe	Direct+Console

6.4 Serial shell

The M519 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, 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.

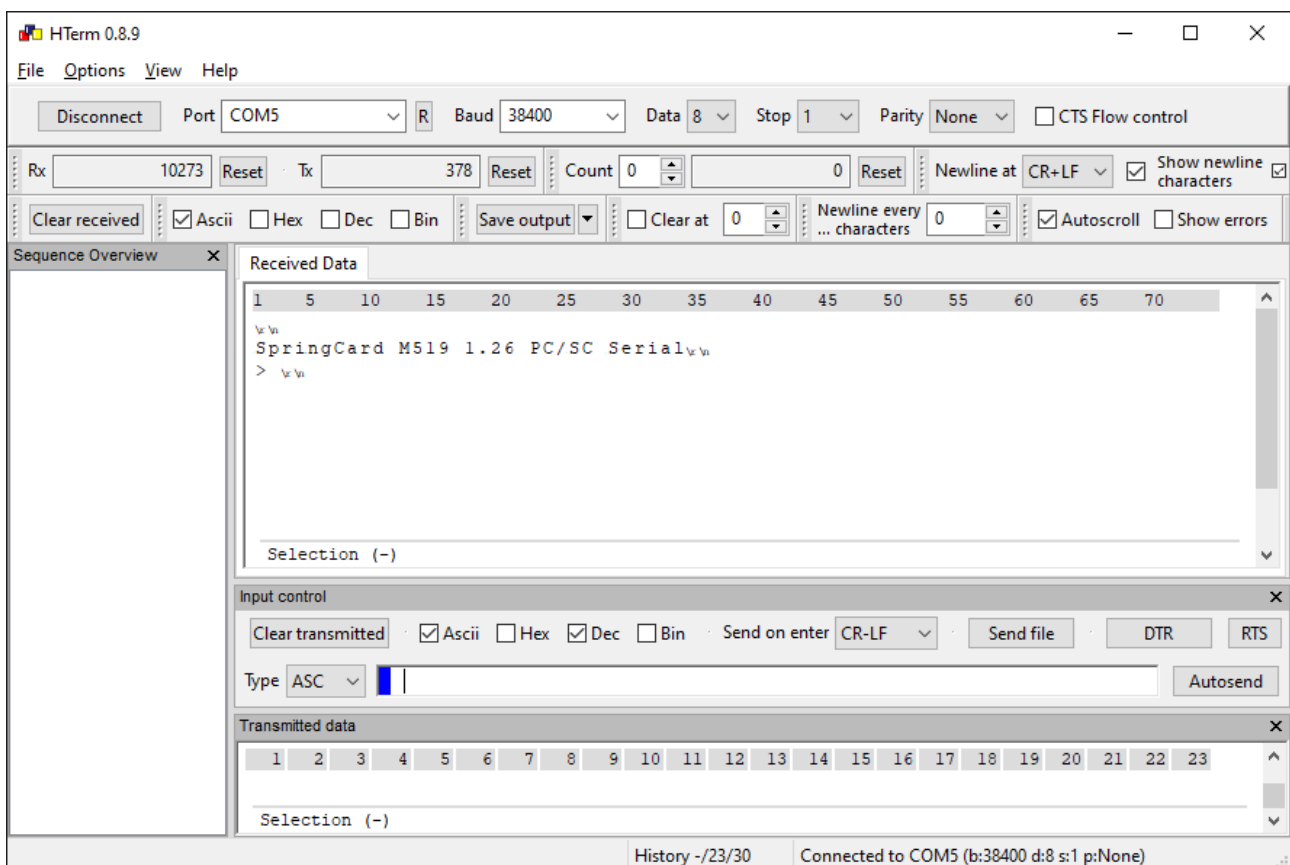


Figure 6: Terminal application 'HTerm' showing the M519's prompt

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. The firmware is generic and its prompt is “M519” in all situations.

6.5 Configuring the M519

The M519 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 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, 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.

6.5.1 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>

6.5.2 Configuration through Master Cards

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

6.5.3 Configuration using shell commands

In the Serial shell (see § 31),

- Send command `cfg<CR><LF>` to read the current configuration,
- Send command `cfgXX=YYYY<CR><LF>` to write value `ₕYYYY` in configuration register `H02XX`,
- Send command `cfgXX=<CR><LF>` to erase configuration register `ₕ02XX` (the default value will apply).

6.5.4 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.

For reference, please read:

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

7 Integration guide – Host interface and power supply

The M519 supports either USB or Serial as host interface.

The USB interface is enabled by powering the module at 5V DC, using the VBUS line coming from the USB host or the USB hub.

If VBUS is not active and the module is powered at 3.3V DC over its VIN_3V3 line, 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 reset or behave incorrectly. Leave the Serial interface unconnected when you want to use the device in USB.

7.1 USB

7.1.1 Overview

The M519 is a USB 2.0 full-speed (12Mbps) device, compliant with USB 3.

USB operation is enabled by connecting the VBUS pin of the module to the 5V supply provided by the bus, and the USB_DP and USB_DM pins to the D+ and D- signals of the bus, respectively. No control signal is required.

The RF supply voltage (VIN_RF pin) is taken from VBUS as well, but, according to the USB specification, VBUS voltage range is 4.75V to 5.25V where VIN_RF must stay under 4.6V (§ Erreur ! Signet non défini.). Therefore, we add a Schottky diode with a forward voltage drop of 0.7V (part number Diodes Incorporated B260A, or equivalent) in the trace.

The reference schematics next page is based on a USB-C connector, but it is trivial to adapt it to a USB mini-B or micro-B connector.

Warning

Designing a PCB that incorporates a USB 2.0 peripheral requires careful consideration of various factors to ensure reliable operation. Please read § 35 carefully and respect all the constraints and recommendations it contains.

7.1.2 Reference schematics

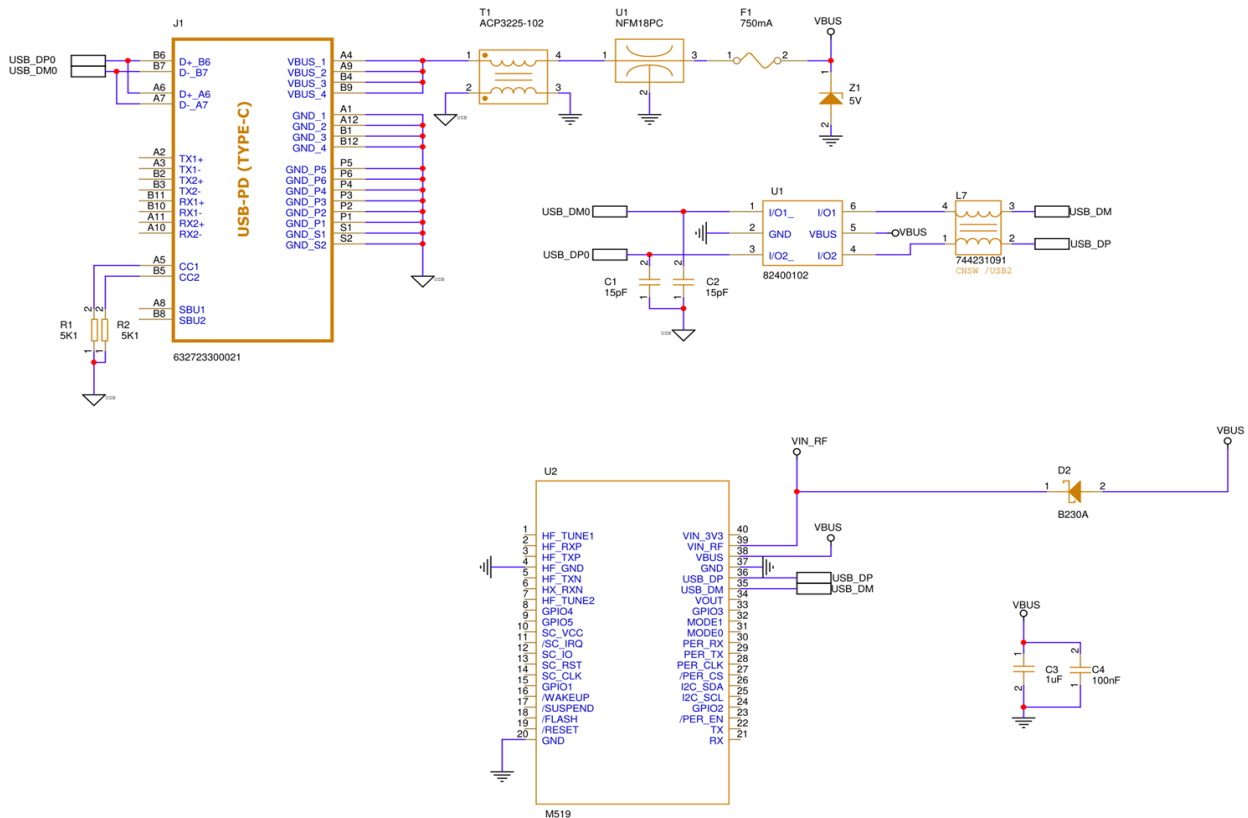


Figure 6: Reference schematics, M519 with USB-C as host interface

7.1.3 Precautions

Designing a PCB to create incorporates a USB peripheral requires careful consideration of various factors to ensure reliable operation.

Differential pair routing:

Maintain the USB differential pairs (D+ and D-) length matched. Mismatches can lead to signal integrity issues. Keep the differential traces close together to maintain the differential impedance. Avoid sharp bends or vias on the differential pairs. If vias are necessary, ensure both lines have a similar number and layout.

Trace length:

Keep USB trace lengths as short as possible. The total length of the USB differential pair traces from the USB connector to the module should be kept under 50mm. Minimize any stubs on the USB data lines. Stubs can introduce reflections and degrade signal integrity.

Separation from noisy traces:

Keep USB traces away from high-speed or noisy traces, such as clock lines, to avoid cross-talk or interference. Separate USB traces from switching power supplies or other potentially noisy components.

Decoupling and filtering:

Place the ESD suppressor (U1, part number [Wurth Elektronik 82400102](#)), the common-mode chokes (L7, part number [Wurth Elektronik 744231091](#) and T1, part number [TDK ACP3225](#)) and the EMI filter (U1, part number [Murata NFM18PC](#)) near the USB connector to suppress noise and protect against electrostatic discharges.

Place decoupling capacitors C3 and C4 close to the power pins of the module.

Ground plane:

A solid, continuous ground plane beneath the USB traces can help improve signal integrity. Ensure a good connection between the ground of the USB connector and the ground plane.

Connector quality:

Use high-quality USB connectors that are durable and have good shielding. Ensure the mechanical stability of the USB connector. It should be securely mounted to the PCB, with extra mechanical support if the application expects frequent plug/unplug cycles.

Shielding:

Do not connect the shield of the USB connector to the ground directly but by the mean of the common mode filter (T1, part number [TDK ACP3225](#)).

Cable, hub or host:

The quality of the USB cable used, as well as its length, can influence performance. Always fully unroll the USB cable. A cable that remains coiled introduces parasitic inductance and exposes the USB interface to the magnetic field generated by the antenna.

The stability of the voltage provided by either the hub or the host on VBUS can have a dramatic impact on the RF performances and even on the stability of the USB communication. Make sure that all requirements of the USB specification are met at the M519's pins, whatever the run-level or the power supply of the host is.

Testing, validation and qualification:

Once the PCB is fabricated, it's crucial to test USB functionality under various conditions (e.g., different cable lengths, with various hosts) to validate the design.

EMC product qualification (necessary for CE, FCC marking and other) should be conducted with the final product (including the cable that would be used in the field). Exchanges must be going on both on the USB interface and on the RF interface simultaneously during the measures.

SpringCard has a strong experience in designing USB NFC/RFID HF devices. Don't hesitate to contact us should you need the assistance of an expert in designing or qualifying your own product.

7.1.4 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 `H02C0`.

Operating mode	Vendor ID	Product ID	Profile
SpringCore Direct ^A	H1C34	H6210	WinUSB
SpringProx Legacy ^B	H1C34	H6211	Compound, CDC-ACM + WinUSB
PC/SC Coupler ^C	H1C34	H6212	Compound, CCID + WinUSB
RFID Scanner ^D	H1C34	H6213	Compound, HID keyboard + WinUSB
Smart Reader ^E	H1C34	H6214	Compound, CDC-ACM + WinUSB
PC/SC Coupler, no Direct interface ^F	H1C34	H621A	CCID

A: selected by `MODE0,MODE1=0,0` or `MODE0,MODE1=1,1` and `H02C0=H00` and `H02C1.bit3=B0`

B: selected by `MODE0,MODE1=1,1` and `H02C0=H01` and `H02C1.bit3=B0`

C: selected by `MODE0,MODE1=1,0` or `MODE0,MODE1=1,1` and `H02C0=H02` and `H02C1.bit3=B0`

D: selected by `MODE0,MODE1=0,1` or `MODE0,MODE1=1,1` and `H02C0=H03` and `H02C1.bit3=B0`

E: selected by `MODE0,MODE1=1,1` and `H02C0=H04` and `H02C1.bit3=B0`

F: selected by `MODE0,MODE1=1,0` or `MODE0,MODE1=1,1` and `H02C0=H02` and `H02C1.bit3=B1`

7.2 Serial

7.2.1 Overview

The serial interface of the M519 is enabled by leaving VBUS unconnected, and by powering the module by the mean of a unique 3.3V supply over both `VIN_3V3` and `VIN_RF`.

Asynchronous serial communication takes place over the UART of the module, on pins RX (# 21, host to module) and TX (# 22, module to host). The `/SUSPEND` control signal and the `/WAKEUP` signal for handshaking may be used, but could definitively be left unconnected in most situations.

Warning

The RX and TX pins operate at 0/3.3V. Do not connect them directly to an RS-232 cable. Take appropriate precautions when using the M519 with a 0/5V host.

7.2.2 Reference schematics

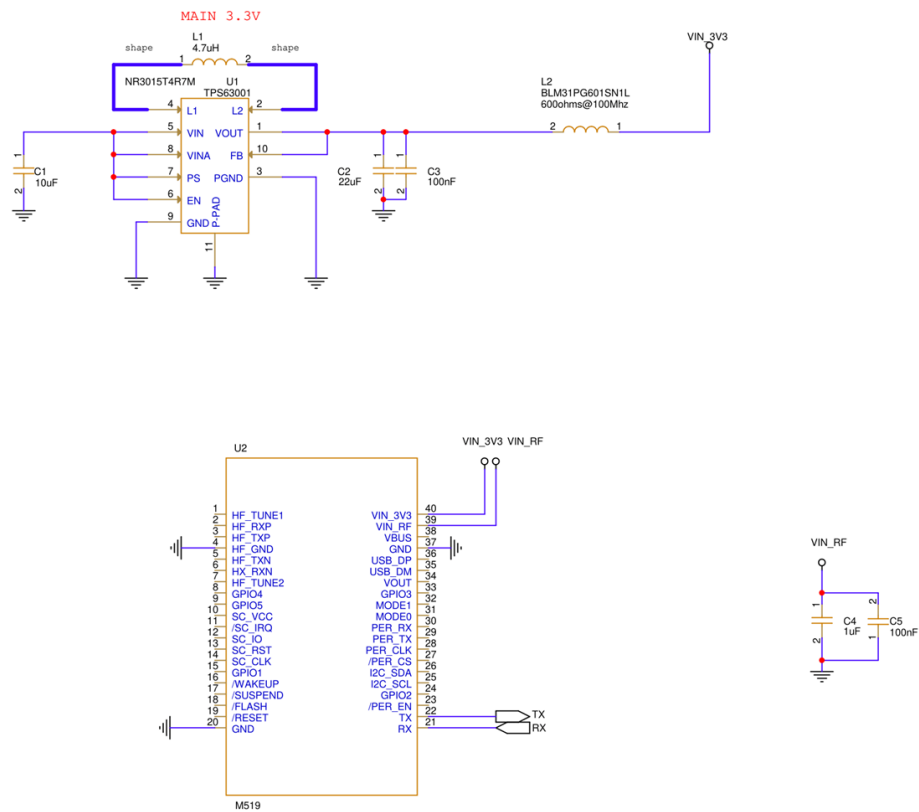


Figure 7: Reference schematics, M519 with a serial host interface

7.2.3 Validation of the implementation

See § 6.4.

8 Integration guide – RF antenna

“Long range” or *far field* radio technologies like Bluetooth, WiFi or RFID UHF, where only the electric field (E) is involved and where both peers are powered, work with miniature antennas; such antennas may even be embedded in small form-factor modules.

On the other hand, NFC/RFID HF is a *near field* radio technology where 1. only the magnetic field (B) is involved; 2. the card/tag does not have its own energy source, but uses power harvesting; 3. the card/tag does not transmit its own wave, but only alters (“load-modulate”) the wave coming from the reader/coupler. All these intrinsic characteristics lead to a (rather) high field strength and to a (relatively) large antenna.

Therefore, the M519 must be associated to an external antenna in order to provide the NFC/RFID HF feature.

SpringCard offers many ready-to-use NFC/RFID HF OEM devices with integrated antenna; for instance, the M519-SUV is made of a M519 mounted on a 69x45mm antenna that provides both an USB interface and a configurable Serial interface.

A *near field* antenna is basically an inductive coil, that can’t be miniaturized too much, because the typical operating distance (i.e. the distance where the RF field is strong enough to provide remote power to the card/tag) is in the same order as the size of the antennas.

The antenna must also be associated to a matching circuit; both must be designed carefully to 1. reach the expected operating range; 2. in the target environment; 3. with the expected cards or tags; and 4. while remaining compliant with EMC (electromagnetic compatibility) regulations, such as ETSI 300-330 for Europe and FCC part 15 for USA.

It is impossible to explain all the design consideration in a short Integration Guide. This chapter will provides the reference schematics of a typical antenna, and the PCB layout and BOM of three antennas that are likely to cover most use cases.

Readers who are interested in going further shall consider NXP document AN12549 “PN5190 antenna design guide” as the starting point. Also, SpringCard has a strong experience in designing NFC/RFID HF antennas. Don’t hesitate to contact us should you need the assistance of an expert in designing your own antenna.

8.1 Static matching circuit

8.1.1 Principle

The M519 needs to be connected to its antenna through an impedance matching circuit (T-filter). The values of the parallel capacitors should be chosen based on the antenna's inductance to make the resonance frequency as close as possible to 13.56MHz. The series capacitors then ensure a real impedance (purely resistive) at this frequency.

It is crucial to adjust the values of the tuning circuit to the exact characteristics of the antenna, which must be measured using a VNA (Vector Network Analyzer) for this purpose. Simulation alone is not sufficient because a PCB introduces resistances and parasitic capacitors that cannot be predicted with sufficient accuracy. Developing such a system, therefore, requires a certain level of expertise and a good understanding of Nyquist diagrams.

8.1.2 Reference schematics

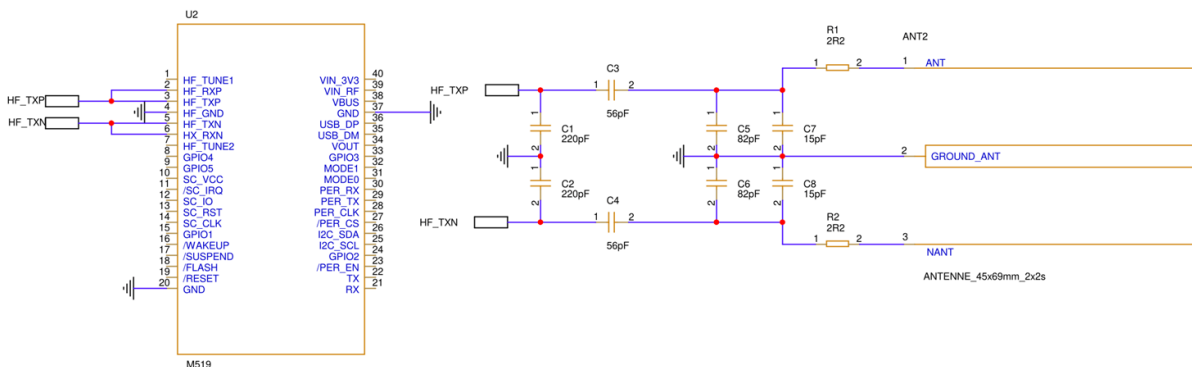


Figure 8: Reference schematics, matching circuit with fixed capacitors

8.2 “Auto-tune” matching circuit

8.2.1 Principle

Some electromagnetic environments, especially the presence of metal near the antenna, can have a significant impact on tuning. There are, in fact, two cumulative effects: a modification of the effective inductance of the antenna and an additional parasitic parallel capacitance between the signal and ground.

The M519 can control a variable tuning circuit that can, to some extent, compensate for the mismatch introduced by the environment. The HF_TUNE_1 signal is used to adjust the series capacitance, and the HF_TUNE_2 signal adjusts the parallel capacitance. When the auto-tune algorithm is activated, the M519 continuously adjusts the values of these two capacitors.

However, it's essential to remember that the role of this system is to optimize the characteristics of a well-tuned antenna in an environment that degrades its performance. It won't be able to deliver good performance with a poorly tuned antenna.

To be completed. The corresponding feature is not yet implemented in the current version of the firmware.

8.2.2 Reference schematics

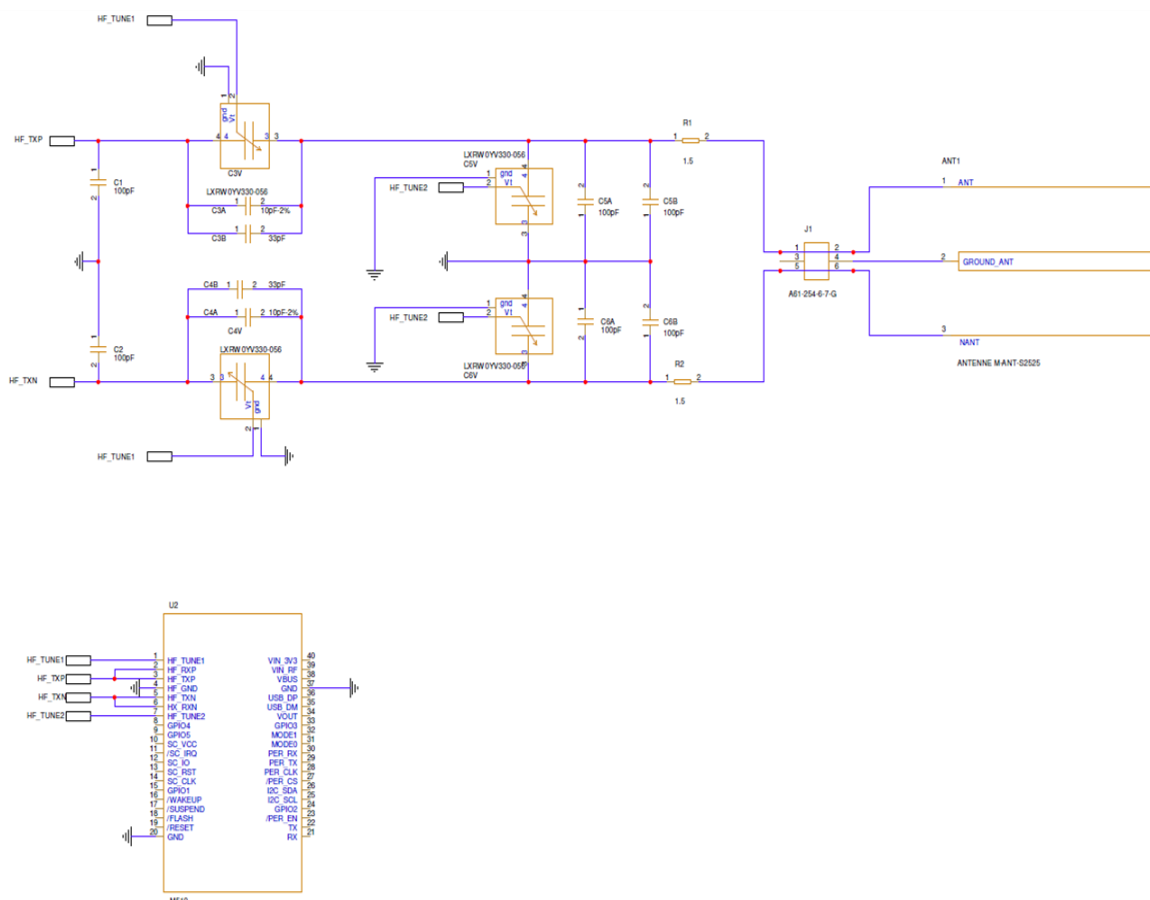


Figure 9: Reference schematics, matching circuit with voltage-driven adjustable capacitors

8.3 Reference PCB layouts and BOMs

SpringCard reference antennas are based on the static matching circuit. The Gerber are reproduced in the following pages for information (images show the layers seen from above). Don't hesitate to contact SpringCard Support to get the actual Gerber files if you want to copy/paste the layouts in your own product.

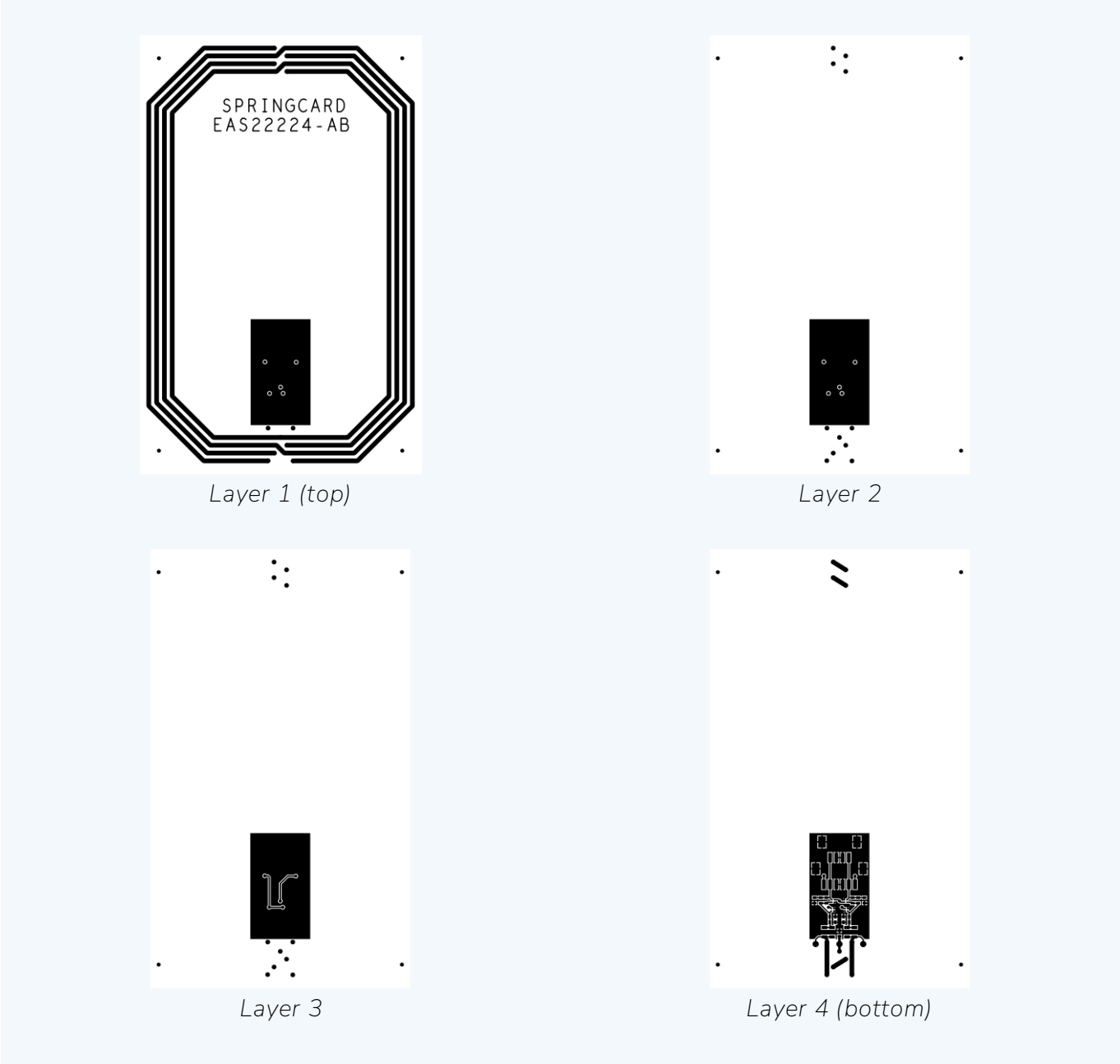
PCB specifications for the antennas are: Class 4, FR4-based, 4 layers with a total thickness of 1.6mm, copper thickness of 35µm.

The suggested BOMs (bill of materials) provide the nominal values of the components. If you are designing a product that utilizes the "auto-tune" function of the M519, adjust the values of both fixed and variable capacitors so that the nominal capacitance falls in the middle of the range of variation.

8.3.1 69x45 balanced antenna

This antenna is the foundation of the M519-SUV product, and available as ANT-S6945 in the M519-SRK.

8.3.1.1 Layout



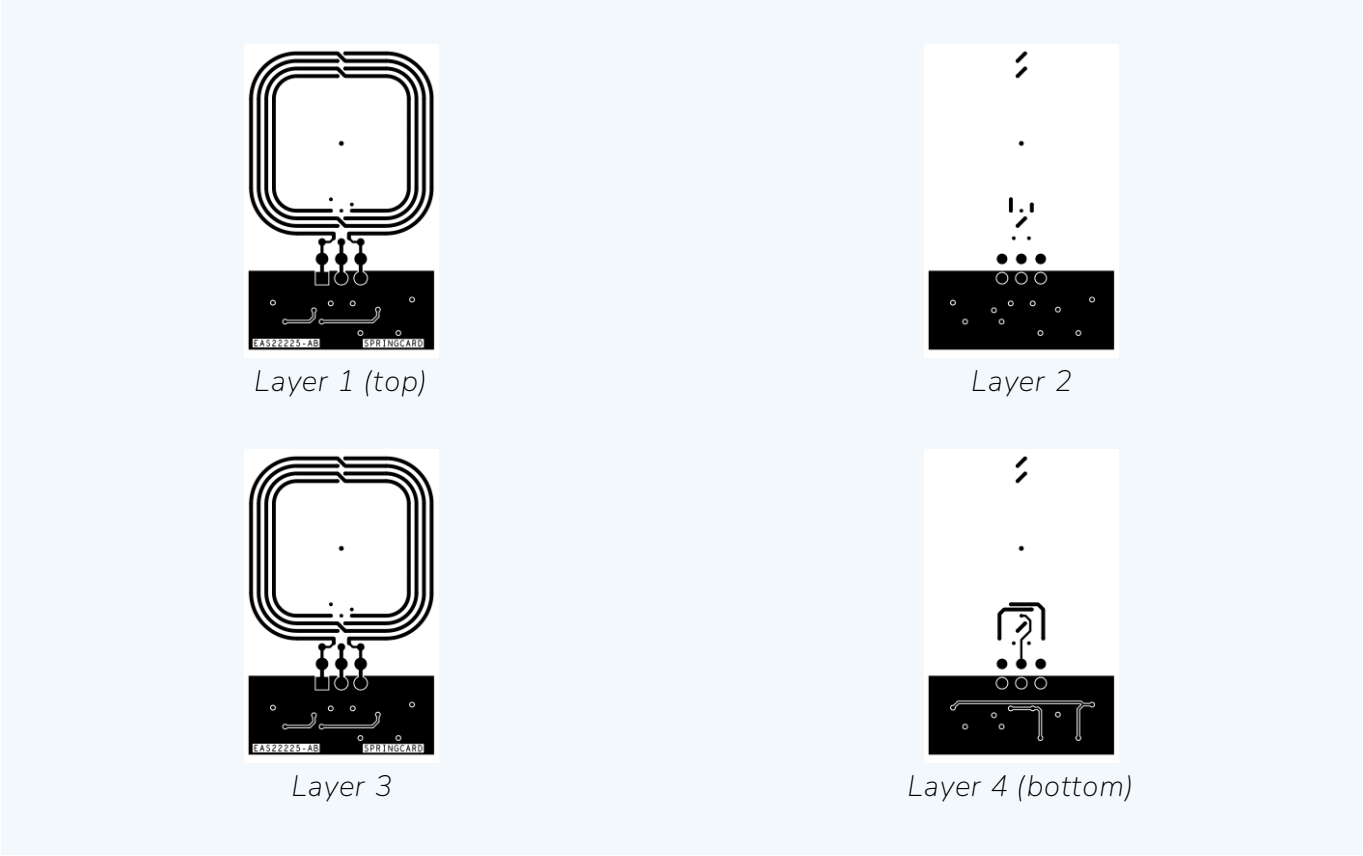
8.3.1.2 BOM

Part	Value	Tolerance	Remark
R1, R2	1.5Ω	±2%	
C1, C2	100pF	±2%	
C3, C4	33pF	±2%	
C5, C6	100pF	±2%	

8.3.2 25x25 balanced antenna

This antenna is available as ANT-S2525 in the M519-SRK.

8.3.2.1 Layout



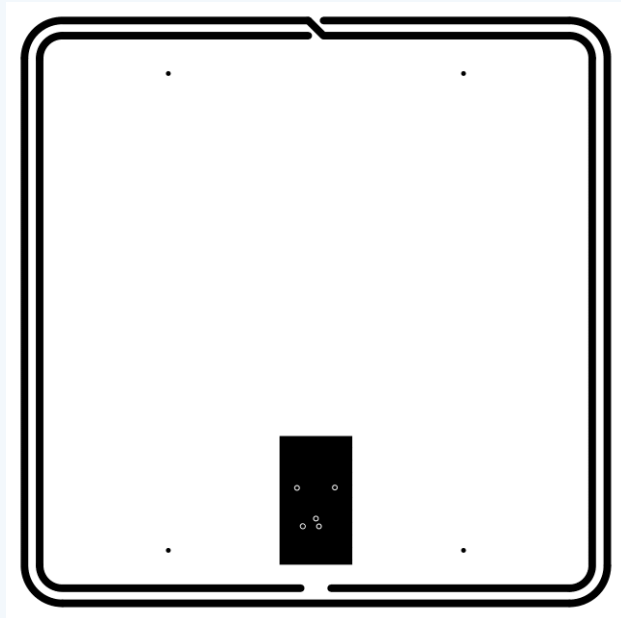
8.3.2.2 BOM

Part	Value	Tolerance	Remark
R1, R2	2.2Ω	±2%	
C1, C2	100pF	±2%	
C3, C4	27pF	±2%	
C5, C6	89pF	±2%	Use 56pF + 33pF

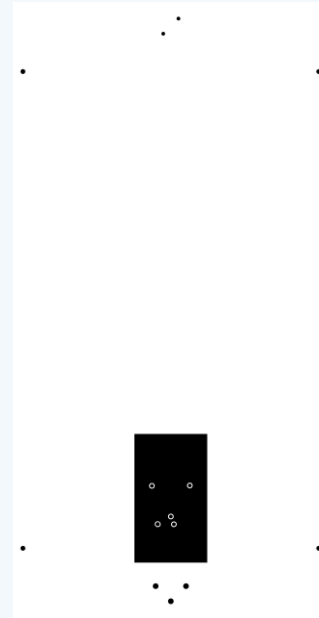
8.3.3 80x80 balanced antenna

This antenna is available as ANT-S8080 in the M519-SRK.

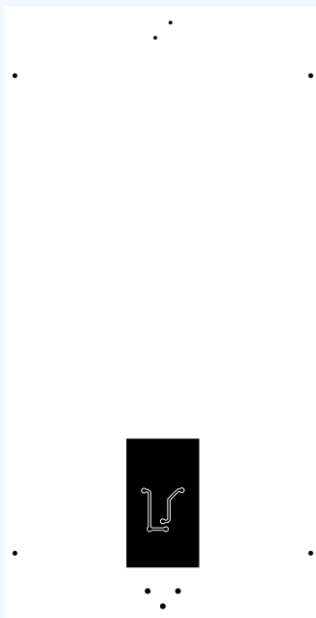
8.3.3.1 Layout



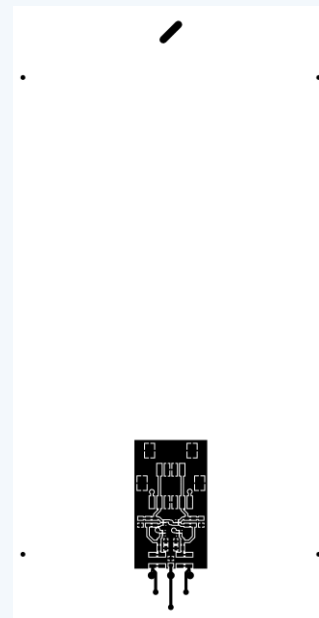
Layer 1 (top)



Layer 2



Layer 3



Layer 4 (bottom)

8.3.3.2 BOM

Part	Value	Tolerance	Remark
R1, R2	1.5Ω	±2%	
C1, C2	100pF	±2%	
C3, C4	56pF	±2%	
C5, C6	230pF	±2%	Use 220pF + 10pF

8.4 Adapting the M519 to its actual antenna

Different antenna have a different impedance and different RF characteristics. The PN5190 NFC/RFID HF frontend IC in the M519 must be configured, in order to adapt its behaviour to the very antenna that it is driving.

The settings for the three compliant antennas developed by SpringCard are stored in the firmware of the M519:

- 69x45mm balanced antenna (foundation of the M519-SUV product, and available as ANT-S6945 in the M519-SRK),
- 25x25mm balanced antenna (ANT-S2525 in the M519-SRK),
- 80x80mm balanced antenna (ANT-S8080 in the M519-SRK).

When the M519 is delivered, the settings for the 69x45mm balanced antenna are selected.

Therefore, the integrator must explicitly activate different settings if a different antenna is used in the final design.

Warning

Using the M519 with an antenna that is not coherent with the settings of the PN5190:

- breaks compliance with ISO/IEC 14443 and 15693 and give an inconsistent operating range
- is likely to break CE/FCC conformity as well
- causes overheating of the module, of the antenna and/or of the matching circuit.

8.4.1 Setting the antenna configuration by software

In factory, the provisioning software or test application shall use the following command to select the antenna once the M519 is mounted on its target hardware:

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

8.4.2 Setting the antenna configuration manually

Until the provisioning software or test application is created, it is possible to change the settings by sending commands in the console or using SpringCoreTool.

Refer to chapter 7.2 in [PMD23175] for comprehensive instructions on this matter.

8.5 Supporting other antennas

Antennas that are far from the 3 reference antennas detailed above can be used only after a custom configuration is created. Creating a custom configuration has to be done using NXP Cockpit application.

Please refer to chapter 5 in NXP document AN12549 “PN5190 antenna design guide” for details.

9 Integration guide – Contact smart card

The M519 features a smart card interface on its pin 10 to 14. The firmware is developed to be compliant with ISO/IEC 7816-3 at protocol level (T=0 and T=1), and with EMV Contact specification for the L1 digital part².

To be fully compliant with these standards (ISO/IEC 7816-3 electrical part, EMV Contact for the L1 analog part), a suitable physical smart card interface must be added to the design.

The M519 supports NXP TDA8035 to implement the smart card hardware. The TDA8035 offer full compliance with the standards and cover all the power classes for the smart cards: class A (5V), class B (3.3V) and class C (1.8V). It implements the ESD protections that are required for any system where the user may insert or remove the smart card. See § 9.2 for reference on using the M519 with one or more TDA8035, to implement up to five smart card slots.

For systems where the smart card is affixed to the PCB (typically when the “smart card” is a secure element in the form of an SMD IC and not actually a card), ESD protections and the support of many power levels are not mandatory. In this situation, it is possible to connect the “smart card” directly to the pins of the M519. See § 9.1 for reference regarding this use case.

Remark

Preliminary releases of the M519 also supported the NXP TDA8026, a multiple-slot smart card interface IC, but this IC has been deprecated by NXP, its support has been removed from the final product.

²The M519 being an OEM device, actual compliance with the standard and specifications is not asserted. Compliance tests and complete certifications could be performed on a “final”, assembled product only.

9.1 Single slot, direct access

9.1.1 Overview

It is possible to connect a smart card directly to the M519, without any physical smart card interface IC, provided that the smart card operates at 3.3V and no ESD protection is required.

Use this design to add a SMD secure element to your product with no extra costs.

9.1.2 Guidelines

The direct access smart card interface is implemented as follow:

- Connect /SC_IRQ to Ground,
- Connect SC_IO to the IO signal on the smart card,
- Connect SC_CLK to the CLK signal on the smart card (the M519 provides a 4MHz clock),
- Connect SC_RST to the RST signal on the smart card,
- Use SC_VCC to control the smart card power (voltage is up when SC_VCC is high). You may use VOUT as power source.

9.1.3 Reference schematics

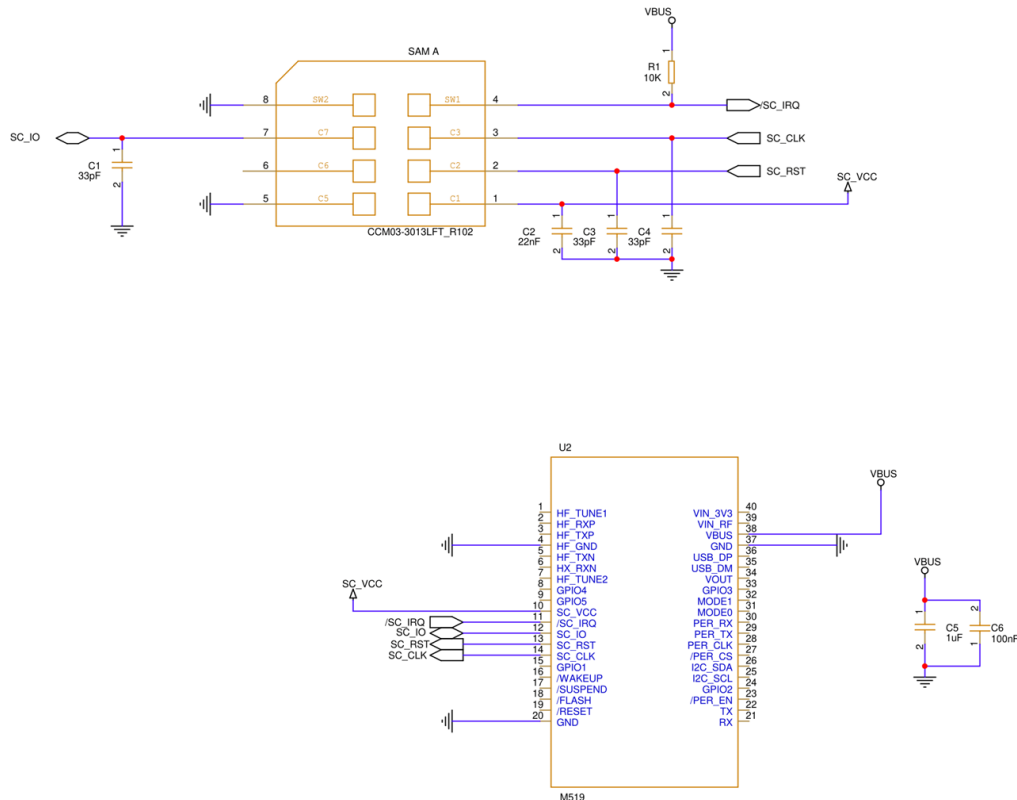


Figure 10: Reference schematics, single SMD SIM/SAM slot, direct access

9.1.4 Firmware implementation / PC/SC readers

Internally, the single slot is slot #1 (#0 is not used). Use this page to configure it:

https://docs.springcard.com/books/SpringCore/Non_volatile_memory/Configuration/Smartcards/First_ID_000_slot

(Of course, the power class settings is ignored in this design, since the hardware is limited to 3.3V only)

On the PC/SC interface, the slot is named “M519 SAM” (without a letter, unless the letter is enforced by configuration).

9.1.5 Card tracking

This design does not provide card tracking (the M519 has no way to be notified if the card is inserted or removed). Make sure that the card is already present when the M519 starts, and remains present all the time.

9.2 ID-000 or ID-1 slot(s) using TDA8035

NXP TDA8035 is a low-power, single-slot analogue smart card interface. It can be used as a smart card multiplexer, in the sense that many TDA8035 can share the same signal lines, provided that they are selected one by one.

For details, please consult the data sheet at NXP's:

<https://www.nxp.com/products/security-and-authentication/contact-readers/high-integrated-and-low-power-smart-card-interface:TDA8035HN>

Driving one TDA8035 require 4 lines. To limit the number of pins involved on the M519 and to extend the concept of smart card multiplexer, I²C I/O expanders are used. Supported ICs are:

- NXP PCA9536 (4 bits, address = $\text{H}41$)

<https://www.nxp.com/part/PCA9536DP>

- NXP PCA6408 (8 bits, address = $\text{H}20$ or $\text{H}21$)

<https://www.nxp.com/part/PCA6408APW>

- NXP PCA9535 (16 bits, address = $\text{H}25$)

<https://www.nxp.com/part/PCA9535D#/>

By combining one to five TDA8035 with the appropriate number of I²C I/O expanders, the M519 is able to drive one to five ID-000 or ID-1 slots.

9.2.1 Guidelines

This guideline is based on the PCA9536, that is suitable to drive one TDA8035. Adjust according when using the other interfaces with more than one TDA8035.

This interface is implemented as follow:

- Connect /SC_IRQ (M519 pin #11) to OFFn on the TDA8035,
- Connect SC_IO (M519 pin #12) to IOUC on the TDA8035,
- Connect SC_CLK (M519 pin #14) to X1 on the TDA8035 (the M519 provides a 4MHz clock), leave X2 unconnected, set CDIV1 and CDIV2 to select a 1:1 clock divider,
- Connect SC_RST (M519 pin #13) to RST on the TDA8035,
- Leave SC_VCC (M519 pin #10) unconnected (the M519 will drive the VCC signal of the card through I²C only; respect exactly the reference schematics in § 9.2.2 to connect the PCA9536 and the TDA8035 correctly),
- Use VOUT (M519 pin #34) to power the PCA9536 and the digital part of the TDA8035; its analog part shall be powered by another source.

9.2.2 Reference schematics

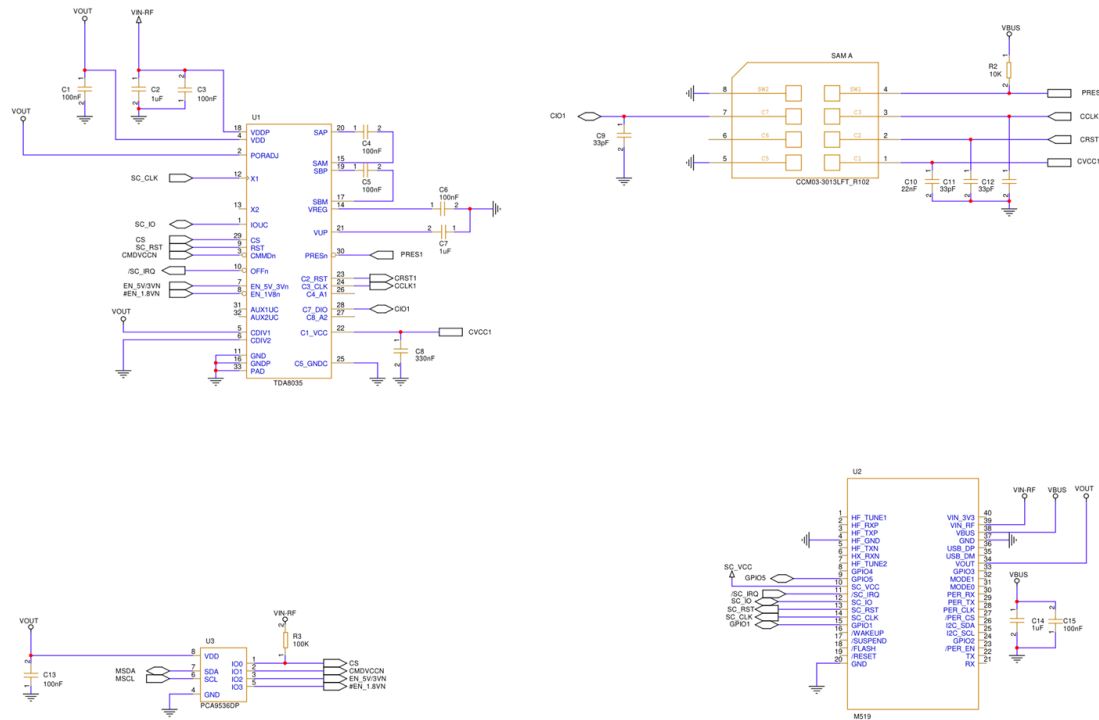


Figure 10: Reference schematics, ID-000 or ID-1 slot with TDA8035 PCA9536

9.2.3 Firmware implementation / PC/SC readers

The M519 supports the following combination of I/O expanders.

Combination	I2C address	Internal slot numbers	PC/SC reader names
PCA9536	$\text{H}41$	1	"SAM" ^A
PCA6408 (1 st address)	$\text{H}20$	1 2	"SAM A" "SAM B"
PCA9536 PCA6408 (1 st address)	$\text{H}41$ $\text{H}20$	0 1 2	"Contact" ^A "SAM A" "SAM B"
PCA6408 (1 st address) PCA6408 (2 nd address)	$\text{H}20$ $\text{H}21$	1 2 3 4	"SAM A" "SAM B" "SAM C" "SAM D"
PCA9536 PCA6408 (1 st address) PCA6408 (2 nd address)	$\text{H}41$ $\text{H}20$ $\text{H}21$	0 1 2 3 4	"Contact" ^A "SAM A" "SAM B" "SAM C" "SAM D"
PCA9535	$\text{H}25$	1 2 3 4	"SAM A" "SAM B" "SAM C" "SAM D"
PCA9536 PCA9535	$\text{H}41$ $\text{H}25$	0 1 2 3 4	"Contact" ^A "SAM A" "SAM B" "SAM C" "SAM D"

A: this is not a typo, the PCA9536+TDA8035 slot takes a different name, depending on the setup: "SAM" if alone, or "Contact" (as it is supposed to be an ID-1 slot) when the setup already contains SAM (ID-000) slots. This behaviour could be overridden by configuration.

Warning

No other combination than the ones listed in the previous tabs are supported.

9.2.4 Card tracking

Thanks to the OFFn signal of the TDA8035 connected to /SC_IRQ, the M519 is notified of card insertion or removal. This let the M519 report the actual card status to the application.

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

SpringCard and the SpringCard logo are registered trademarks of SpringCard SAS. All other brand or product names are the property of their respective owners.

This document is the exclusive property of SpringCard SAS. No part of this document may be reproduced or transmitted in any form or by any means without the written consent of SpringCard. SpringCard reserves the right to change this document at any time without notice.

All rights reserved.