

**RFID MODULE**

**Mifare Reader / Writer**

**SL025B**  
**User Manual**



**Version 3.0**  
**Jan 2018**  
**StrongLink**

# CONTENT

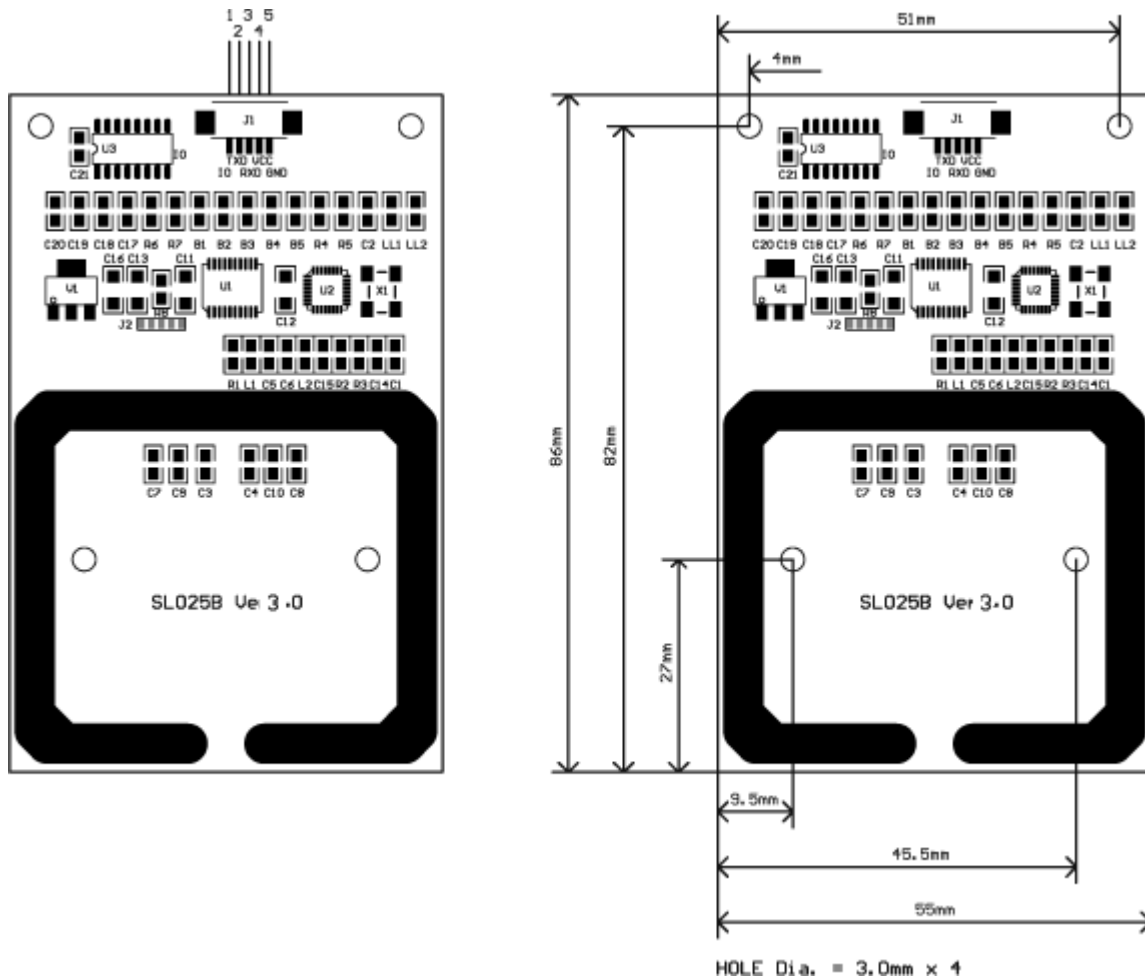
<b>1. MAIN FEATURES .....</b>	<b>3</b>
<b>2. PINNING INFORMATION .....</b>	<b>4</b>
<b>3. BAUD RATE SETTING .....</b>	<b>5</b>
<b>4. COMMUNICATION PROTOCOL.....</b>	<b>5</b>
4-1. COMMUNICATION SETTING .....	5
4-2. COMMUNICATION FORMAT .....	5
4-3. COMMAND OVERVIEW .....	6
4-4. COMMAND LIST .....	7
4-4-1. Select Mifare card .....	7
4-4-2. Login to a sector .....	7
4-4-3. Download Key into SL025.....	7
4-4-4. Login sector via stored key .....	8
4-4-5. Read a data block .....	8
4-4-6. Write a data block .....	8
4-4-7. Read a value block .....	8
4-4-8. Initialize a value block .....	9
4-4-9. Write master key (key A).....	9
4-4-10. Increment value .....	9
4-4-11. Decrement value.....	10
4-4-12. Copy value.....	10
4-4-13. Read a data page (UltraLight) .....	10
4-4-14. Write a data Page (UltraLight).....	11
4-4-15. Manage Red Led .....	11
4-4-16. Get firmware version.....	11

## 1. MAIN FEATURES



- Tags supported: Mifare 1k, Mifare 4k, Mifare UltraLight and NFC NTAG203
- Auto-detecting tag, Built-in antenna
- RS232 interface, baud rate 9,600 ~ 115,200 bps
- 4.4 ~ 12.0VDC power supply
- Work current less than 80mA
- Operating distance: Up to 70mm, depending on tag
- Storage temperature: -40 °C ~ +85 °C
- Operating temperature: -25 °C ~ +70 °C
- Dimension: 86 × 55 mm
- Two LEDs, green led is auto light when tag in detection range, red led is controlled by host
- The OUT pin at low level indicates tag in detective range, and high level indicating tag out
- Support IAP firmware update

## 2. PINNING INFORMATION



Connector: Würth Elektronik [653 105 131 822](https://www.wuerth-elektronik.com/en/connector/653105131822)

PIN	SYMBOL	TYPE	DESCRIPTION
1	TagSta	Output	Tag detect signal, RS232 level Logic 0 indicating tag in detection range Logic 1 indicating tag out
2	TXD	Output	Serial output port
3	RXD	Input	Serial input port
4	VCC	PWR	Power Supply
5	GND	PWR	Ground

### 3. BAUD RATE SETTING

Two 820 ohm resistances R6 & R7 are used for setting baud rate as follows sheet

	R6	R7	Baud rate bps
Assembled	no	no	9,600
	yes	no	19,200
	no	yes	57,600
	yes	yes	115,200 ( default )

## 4. COMMUNICATION PROTOCOL

### 4-1. Communication Setting

The communication protocol is byte oriented. Both sending and receiving bytes are in hexadecimal format. The communication parameters are as follows

Baud rate: 9,600 ~ 115,200 bps

Data: 8 bits

Stop: 1 bit

Parity: None

Flow control: None

### 4-2. Communication Format

#### Host to SL025:

Preamble	Len	Command	Data	Checksum
----------	-----	---------	------	----------

Preamble: 1 byte equal to 0xBA

Len: 1 byte indicating the number of bytes from Command to Checksum

Command: 1 byte Command code, see Table 3

Data: Variable length depends on the command type

Checksum: 1 byte XOR of all the bytes from Preamble to Data

#### SL025 to Host:

Preamble	Len	Command	Status	Data	Checksum
----------	-----	---------	--------	------	----------

Preamble: 1 byte equal to 0xBD

Len: 1 byte indicating the number of bytes from Command to Checksum

Command: 1 byte Command code, see Table 3

Status: 1 byte Command status, see Table 4

Data: Variable length depends on the command type.

Checksum: 1 byte XOR of all the bytes from Preamble to Data

### 4-3. Command Overview

Table 3

Command	Description
0x01	Select Mifare card
0x02	Login to a sector
0x03	Read a data block
0x04	Write a data block
0x05	Read a value block
0x06	Initialize a value block
0x07	Write master key (key A)
0x08	Increment value
0x09	Decrement value
0x0A	Copy value
0x10	Read a data page (UltraLight & NTAG203)
0x11	Write a data page (UltraLight & NTAG203)
0x12	Download Key
0x13	Login sector via stored Key
0x40	Manage Red Led
0xF0	Get firmware version

### Status Overview

Table 4

Status	Description
0x00	Operation succeed
0x01	No tag
0x02	Login succeed
0x03	Login fail
0x04	Read fail
0x05	Write fail
0x06	Unable to read after write
0x08	Address overflow
0x09	Download Key fail
0x0D	Not authenticate
0x0E	Not a value block
0xF0	Checksum error
0xF1	Command code error

## 4-4. Command List

### 4-4-1. Select Mifare card

0xBA	Len	0x01	Checksum
------	-----	------	----------

#### Response:

0xBD	Len	0x01	Status	UID	Type	Checksum
------	-----	------	--------	-----	------	----------

Status: 0x00: Operation succeed  
 0x01: No tag  
 0xF0: Checksum error

UID: The uniquely serial number of Mifare card,

Type: 0x01: Mifare 1k, 4 byte UID  
 0x02: Mifare 1k, 7 byte UID <sup>[1]</sup>  
 0x03: Mifare UltraLight or NATG203<sup>[2]</sup>, 7 byte UID  
 0x04: Mifare 4k, 4 byte UID  
 0x05: Mifare 4k, 7 byte UID <sup>[1]</sup>  
 0x06: Mifare DesFire, 7 byte UID  
 0x0A: Other

### 4-4-2. Login to a sector

0xBA	Len	0x02	Sector	Type	Key	Checksum
------	-----	------	--------	------	-----	----------

Sector: Sector need to login, 0x00 – 0x27  
 Type: Key type (0xAA: authenticate with KeyA, 0xBB: authenticate with KeyB)  
 Key: Authenticate key, 6 bytes

#### Response:

0xBD	Len	0x02	Status	Checksum
------	-----	------	--------	----------

Status: 0x02: Login succeed  
 0x01: No tag  
 0x03: Login fail  
 0x08: Address overflow  
 0xF0: Checksum error

### 4-4-3. Download Key into SL025

0xBA	Len	0x12	Sector	Type	Key	Checksum
------	-----	------	--------	------	-----	----------

Sector: 0x00 – 0x27  
 Type: Key type (0xAA: KeyA, 0xBB: KeyB)  
 Key: 6 bytes, stored into SL025

#### Response:

0xBD	Len	0x12	Status	Checksum
------	-----	------	--------	----------

Status: 0x00: Operation succeed  
 0x08: Address overflow  
 0x09: Download fail  
 0xF0: Checksum error

**4-4-4. Login sector via stored key**

0xBA	Len	0x13	Sector	Type	Checksum
------	-----	------	--------	------	----------

Sector: Sector need to login, 0x00 – 0x27

Type: Key type (0xAA: KeyA, 0xBB: KeyB)

**Response:**

0xBD	Len	0x13	Status	Checksum
------	-----	------	--------	----------

Status: 0x02: Login succeed

0x03: Login fail

0x08: Address overflow

0xF0: Checksum error

**4-4-5. Read a data block**

0xBA	Len	0x03	Block	Checksum
------	-----	------	-------	----------

Block: The absolute address of block to be read, 1 byte

**Response:**

0xBD	Len	0x03	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation succeed

0x01: No tag

0x04: Read fail

0x0D: Not authenticate

0xF0: Checksum error

Data: Block data returned if operation succeeds, 16 bytes.

**4-4-6. Write a data block**

0xBA	Len	0x04	Block	Data	Checksum
------	-----	------	-------	------	----------

Block: The absolute address of block to be written, 1 byte.

Data: The data to write, 16 bytes.

**Response:**

0xBD	Len	0x04	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation succeed

0x01: No tag

0x05: Write fail

0x06: Unable to read after write

0x0D: Not authenticate

0xF0: Checksum error

Data: Block data written if operation succeeds, 16 bytes.

**4-4-7. Read a value block**

0xBA	Len	0x05	Block	Checksum
------	-----	------	-------	----------

Block: The absolute address of block to be read, 1 byte.

**Response:**

0xBD	Len	0x05	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation succeed



0x01: No tag  
 0x04: Read fail  
 0x0D: Not authenticate  
 0x0E: Not a value block  
 0xF0: Checksum error

Value: Value returned if the operation succeeds, 4 bytes.

#### 4-4-8. Initialize a value block

0xBA	Len	0x06	Block	Value	Checksum
------	-----	------	-------	-------	----------

Block: The absolute address of block to be initialized, 1 byte.

Value: The value to be written, 4 bytes.

#### Response:

0xBD	Len	0x06	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation succeed  
 0x01: No tag  
 0x05: Write fail  
 0x06: Unable to read after write  
 0x0D: Not authenticate  
 0xF0: Checksum error

Value: Value written if the operation succeeds, 4 bytes.

#### 4-4-9. Write master key (key A)

0xBA	Len	0x07	Sector	Key	Checksum
------	-----	------	--------	-----	----------

Sector: The sector number to be written, 0x00 – 0x27

Key: Authentication key, 6 bytes

#### Response:

0xBD	Len	0x07	Status	Key	Checksum
------	-----	------	--------	-----	----------

Status: 0x00: Operation succeed  
 0x01: No tag  
 0x05: Write fail  
 0x08: Address overflow  
 0x0D: Not authenticate  
 0xF0: Checksum error

Key: Authentication key written if the operation succeeds, 6 bytes.

Attention: Be sure KeyB is readable, otherwise KeyB will be change to 000000000000 after this command.

#### 4-4-10. Increment value

0xBA	Len	0x08	Block	Value	Checksum
------	-----	------	-------	-------	----------

Block: The absolute address of block to be increased, 1 byte.

Value: The value to be increased by, 4 bytes.

#### Response:

0xBD	Len	0x08	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation succeed  
 0x01: No tag  
 0x05: Write fail  
 0x06: Unable to read after write  
 0x0D: Not authenticate  
 0x0E: Not a value block  
 0xF0: Checksum error

Value: The value after increment if the operation succeeds, 4 bytes

#### 4-4-11. Decrement value

0xBA	Len	0x09	Block	Value	Checksum
------	-----	------	-------	-------	----------

Block: The absolute address of block to be decreased, 1 byte  
 Value: The value to be decreased by, 4 bytes

#### Response:

0xBD	Len	0x09	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation succeed  
 0x01: No tag  
 0x05: Write fail  
 0x06: Unable to read after write  
 0x0D: Not authenticate  
 0x0E: Not a value block  
 0xF0: Checksum error

Value: The value after decrement if the operation succeeds, 4 bytes

#### 4-4-12. Copy value

0xBA	Len	0x0A	Source	Destination	Checksum
------	-----	------	--------	-------------	----------

Source: The source block copy from, 1 byte  
 Destination: The destination copy to, 1 byte  
 Attention: The source and destination must in the same sector

#### Response:

0xBD	Len	0x0A	Status	Value	Checksum
------	-----	------	--------	-------	----------

Status: 0x00: Operation succeed  
 0x01: No tag  
 0x05: Write fail  
 0x06: Unable to read after write  
 0x0D: Not authenticate  
 0x0E: Not a value block (Source)  
 0xF0: Checksum error

Value: The value after copy if the operation succeeds, 4 bytes

#### 4-4-13. Read a data page (UltraLight & NTAG203)

0xBA	Len	0x10	Page	Checksum
------	-----	------	------	----------

Page: The page number to be read, 0x00 – 0x0F

**Response:**

0xBD	Len	0x10	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation succeed  
 0x01: No tag  
 0x04: Read fail  
 0x08: Address overflow  
 0xF0: Checksum error

Data: Block data returned if operation succeeds, 4 bytes.

**4-4-14. Write a data Page (UltraLight & NTAG203)**

0xBA	Len	0x11	Page	Data	Checksum
------	-----	------	------	------	----------

Page: The page number to be written, 0x00 – 0x0F

Data: The data to write, 4 bytes.

**Response:**

0xBD	Len	0x11	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation succeed  
 0x01: No tag  
 0x05: Write fail  
 0x06: Unable to read after write  
 0x08: Address overflow  
 0xF0: Checksum error

Data: Page data written if operation succeeds, 4 bytes.

**4-4-15. Manage Red Led**

0xBA	Len	0x40	Code	Checksum
------	-----	------	------	----------

Code: 0 command red led turn off , other red led turn on, 1 byte

**Return:**

0xBD	Len	0x40	Status	Checksum
------	-----	------	--------	----------

Status: 0x00: Operation succeed  
 0xF0: Checksum error

**4-4-16. Get firmware version**

0xBA	Len	0xF0	Checksum
------	-----	------	----------

**Response:** <sup>[3]</sup>

0xBD	Len	0xF0	Status	Data	Checksum
------	-----	------	--------	------	----------

Status: 0x00: Operation success  
 0xF0: Checksum error

Data: firmware version.

**Remark**

<sup>[1]</sup> In order to support 7 byte UID Mifare class, the firmware of SL025 has been updated to Ver1.2 in Mar 2011.

And older firmware version (such as Ver1.0, 1.1) only supports 4 byte UID.

Please refer to NXP [Customer Letter UID](#) for detailed information of 4 byte & 7 byte UID of Mifare products.

<sup>[2]</sup> To support NATG203, the firmware of SL025 has been updated to Ver1.6 in May 2012. The older firmware version only supports reading/writing data page address less than 16.

<sup>[3]</sup> One sample of SL025 response

	Preamble	Len	Command	Status	Data (Firmware version)	Checksum
HEX	BD	15	F0	00	53 4C 30 32 35 2D 33 2E 30 2D 32 30 31 36 31 31 31 34	69
ASCII					“SL025-3.0-20161114 ”	