



## 1. GENERAL INFORMATION

Dear customer,

thank you very much for choosing our product.

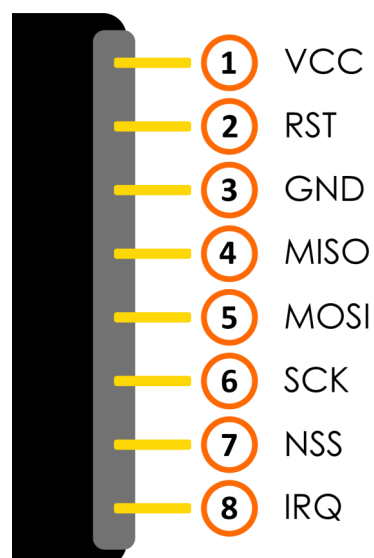
In the following, we will introduce you to what to observe while starting up and using this product.

Should you encounter any unexpected problems during use, please do not hesitate to contact us.

## 2. OVERVIEW

Frequency	13.56 MHz
Communication	SPI
RFID Protocol	Mifare
Supply voltage	DC 3,3V

## 3. PIN ASSIGNMENT



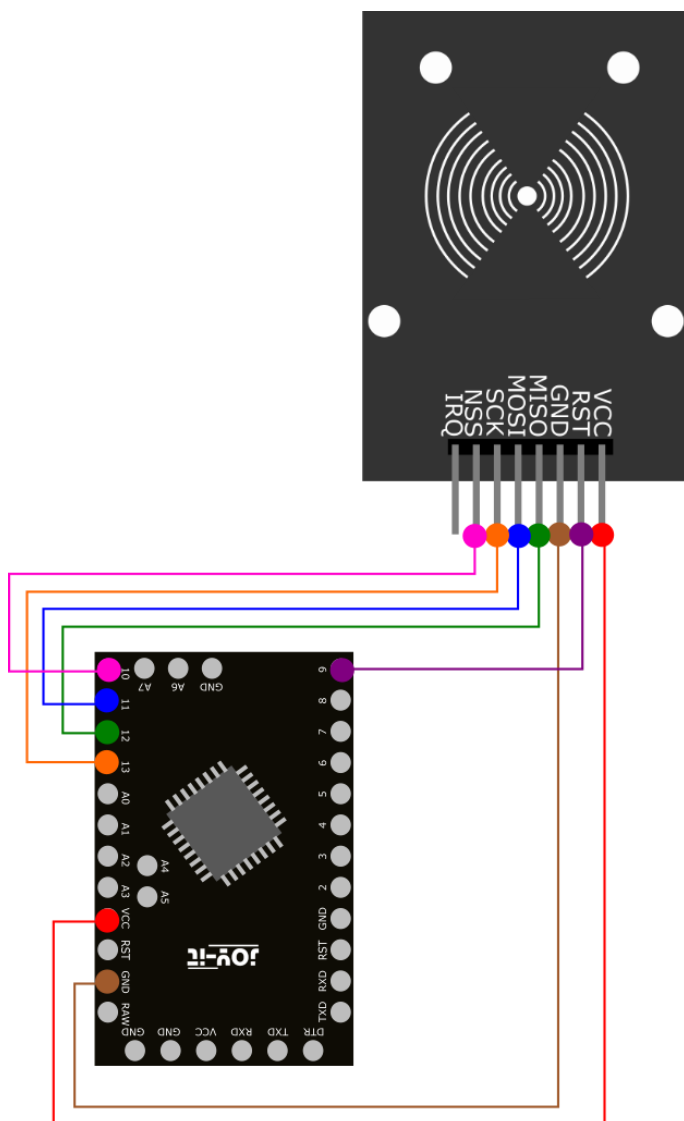
## 4. USE WITH AN ARDUINO

### 4.1 Connection of the module

Since the module works with a 3 V logic level and most Arduinos work with 5V, we use an Arduino Pro Mini 3.3 V in this example.

If you want to use an Arduino with a 5 V logic level, such as an Arduino Uno, you need to downgrade all data lines leading from the Arduino to the module from 5 V to 3.3 V with a voltage translator.

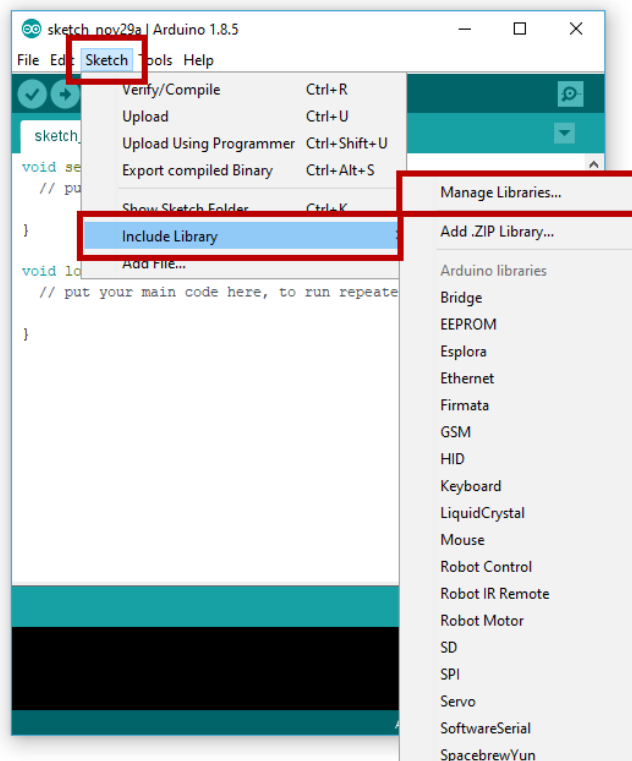
Connect the RFID module as shown in the following picture to the pins of your Arduino.



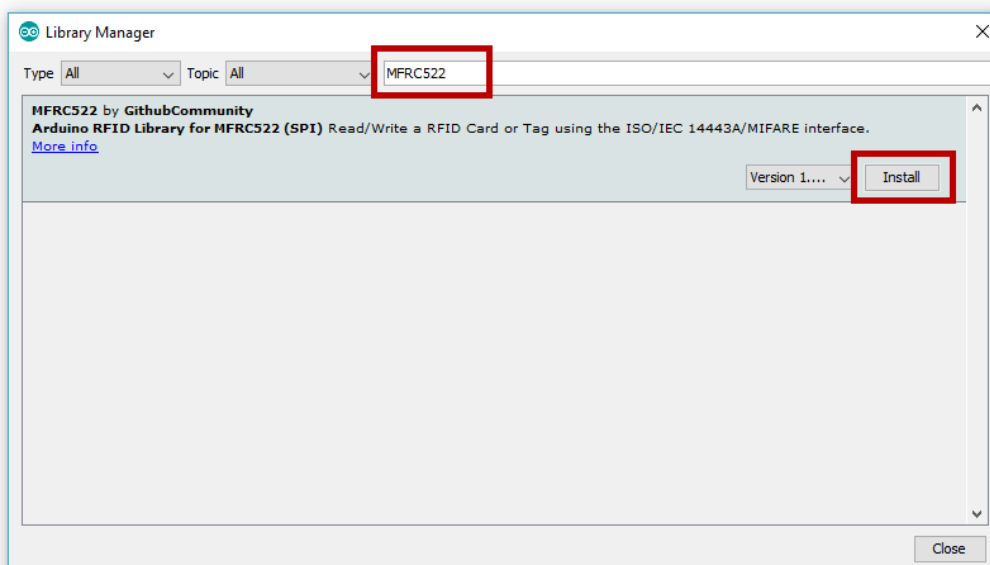
RFID module	Arduino
VCC	VCC
RST	Pin 9
GND	GND
MISO	Pin 12
MOSI	Pin 11
SCK	Pin 13
NSS	Pin 10
IRQ	-

## 4.2 Installation of the module

To use the module with your Arduino, you must first install the **MFRC522** library from the Arduino library manager.



In the window that opens, enter the term **"MFRC522"** in the search field and install the library.



### 4.3 Example application

After you have successfully installed the library, your module is ready for use.

You can use the following code example and transfer it to your Arduino, to commission and test the module:

```
#include <SPI.h>
#include <MFRC522.h>

#define RST_PIN      9
#define SS_PIN       10

MFRC522 mfrc522(SS_PIN, RST_PIN);

void setup() {
  //Initialization of the RFID module
  Serial.begin(9600);
  while (!Serial);
  SPI.begin();
  mfrc522.PCD_Init();
  mfrc522.PCD_DumpVersionToSerial(); //Output details of the reader
  Serial.println(F("Scan PICC to see UID, type, and data blocks..."));
}

void loop() {
  //Search for new cards
  if ( ! mfrc522.PICC_IsNewCardPresent() ) {
    return;
  }

  if ( ! mfrc522.PICC_ReadCardSerial() ) {
    return;
  }

  //Information retrieval of the RFID device
  mfrc522.PICC_DumpToSerial(&(mfrc522.uid));
}
```

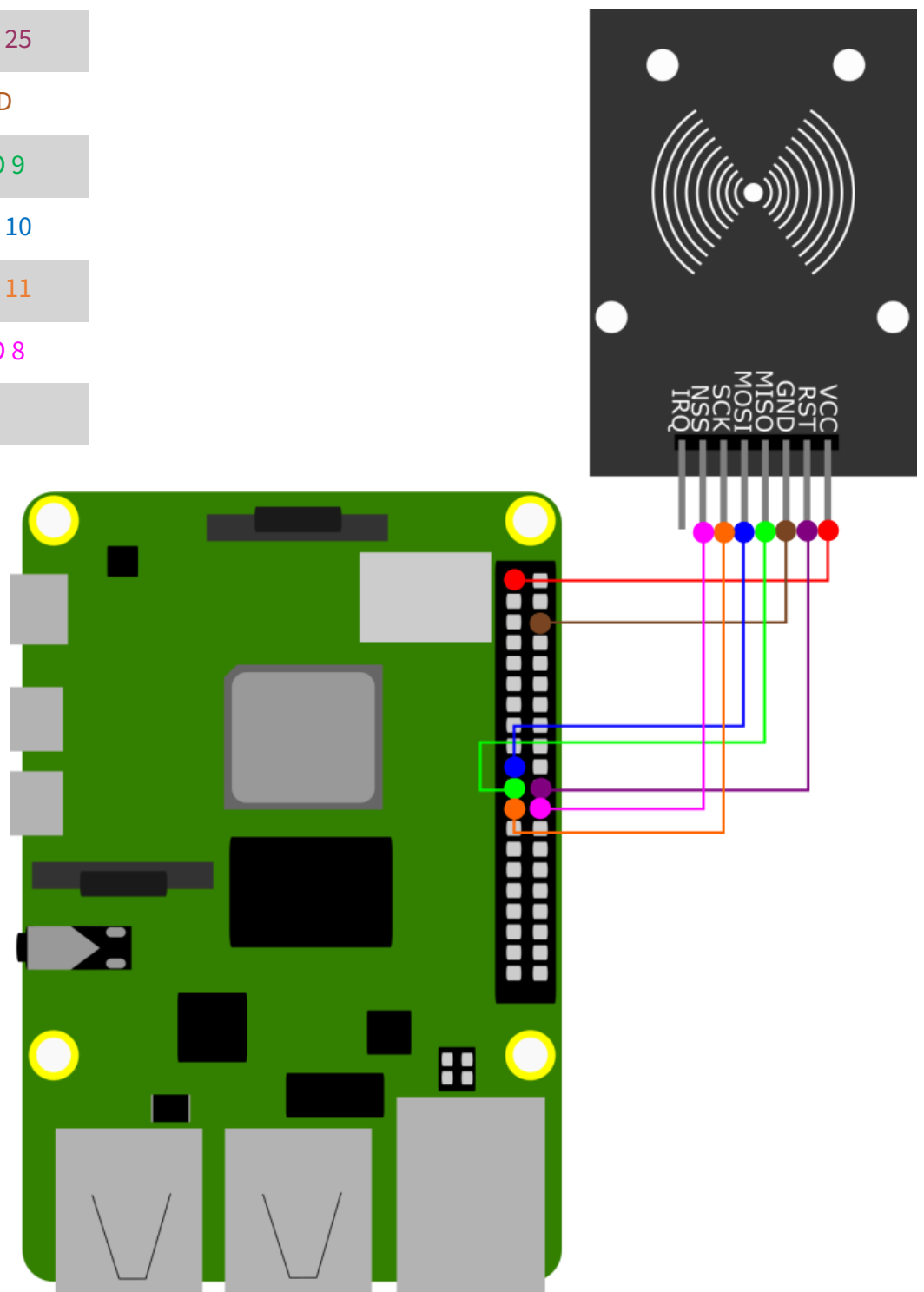
Under **File** -> **Examples** -> **MFRC522** you can find even more example files.

## 5. USE WITH A RASPBERRY PI

### 5.1 Connection of the module

Connect the RFID module to the pins of your Raspberry Pis as shown in the following picture .

RFID module	Raspberry Pi
VCC	3,3V
RST	GPIO 25
GND	GND
MISO	GPIO 9
MOSI	GPIO 10
SCK	GPIO 11
NSS	GPIO 8
IRQ	-



## 5.2 Installation of the module

Before you can use the module, you have to install additional libraries and enable the SPI interface. To do this, open a terminal window on your Raspberry Pi and enter the following commands:

```
sudo raspi-config
```

Use the arrow keys to select "**3 Interfacing Options**" and confirm with Enter. Select "**I3 SPI**" and confirm the activation with "**Yes**".

Now after preparations have been completed, the actual library can be installed:

```
sudo apt-get install -y python3-dev python3-pip git
```



With Bookworm OS, global installations can no longer be carried out. Therefore, a virtual environment must be created in which this installation takes place. All code which requires these installations must be in this virtual environment.

To create a virtual environment, execute the following commands:

```
mkdir your_project
```

```
cd your_project
```

```
python -m venv --system-site-packages env
```

```
source env/bin/activate
```

Now we can install the required libraries in this virtual environment:

```
pip3 install spidev
```

```
git clone https://github.com/joy-it/MFRC522-python
```

```
cd MFRC522-python
```

```
python3 setup.py install
```

```
cd ..
```

### 5.3 Example applications

To run our sample codes, run the file to write something on a clip or card with the following command:

```
python3 MFRC522-python/examples/write.py
```

You can start the application for reading with the following command:

```
python3 MFRC522-python/examples/read.py
```

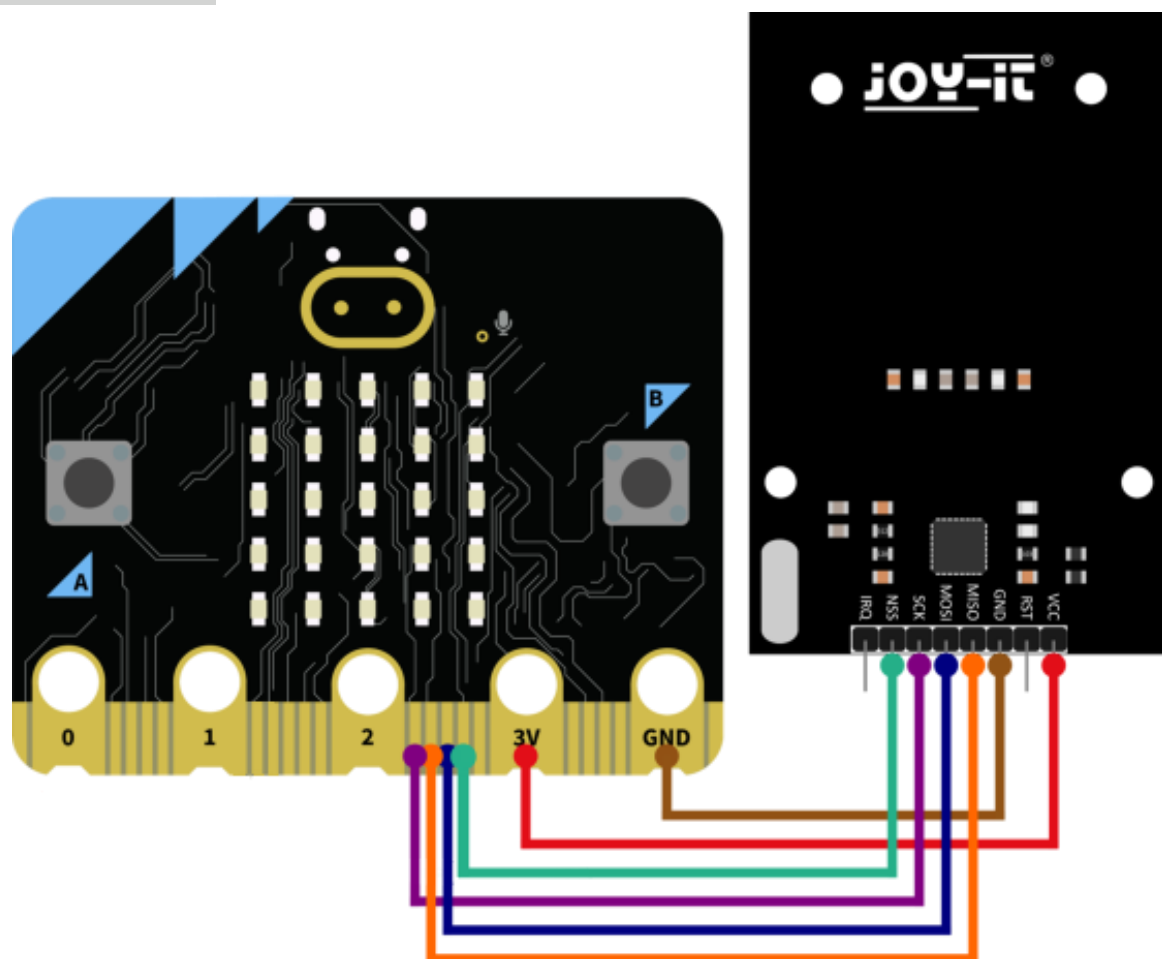


## 6. USE WITH A BBC MICRO:BIT

### 6.1 Connection of the module

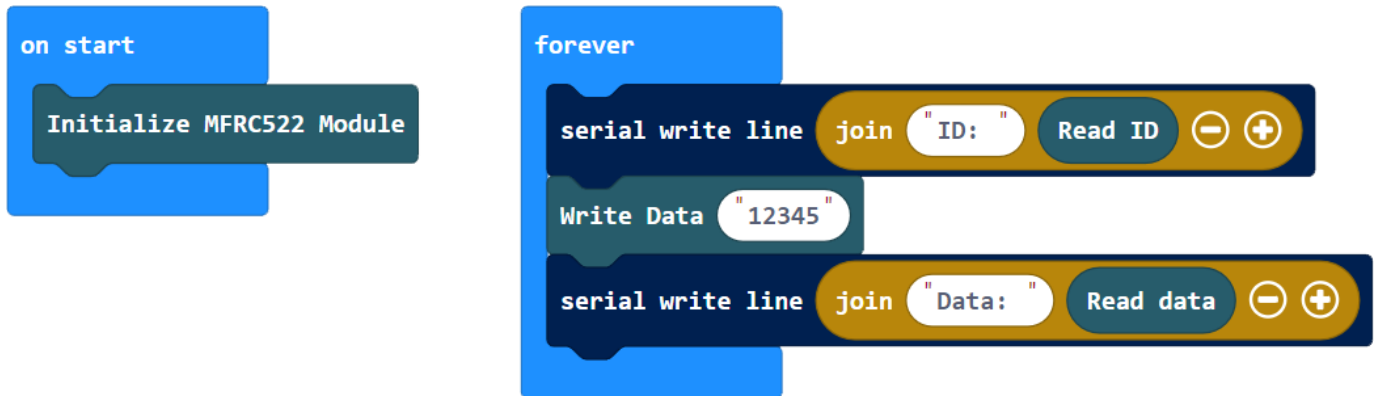
Connect the RFID module to your micro:bit as shown in the following picture.

RFID Modul	Micro:bit
VCC	3,3 V
RST	/
GND	GND
MISO	P14
MOSI	P15
SCK	P13
NSS	P16
IRQ	-



## 6.2 Example applications

To test the module on your micro:bit you first need a suitable library. To do this, click on the "Add extension" button in your MakeCode project and search here for the term "RFID". Install here the "rfid-mfrc522" library from Joy-IT. Now you can transfer the following blocks into your project:



```
on start
  Initialize MFRC522 Module

forever
  serial write line join "ID: " Read ID
  Write Data "12345"
  serial write line join "Data: " Read data
```

The image shows two code blocks from a MakeCode project. The first block is an 'on start' block containing an 'Initialize MFRC522 Module' block. The second block is a 'forever' loop containing three blocks: a 'serial write line' block with a 'join' block containing 'ID: ' and a 'Read ID' block; a 'Write Data' block with the value '12345'; and another 'serial write line' block with a 'join' block containing 'Data: ' and a 'Read data' block.

## 7. USE WITH A RASPBERRY PI PICO

### 7.1 Connection of the module

Connect the RFID module to your Raspberry Pi Pico as shown in the following picture.

RFID Modul	Raspberry Pi Pico
------------	-------------------

VCC	3,3 V
-----	-------

RST	GP0
-----	-----

GND	GND
-----	-----

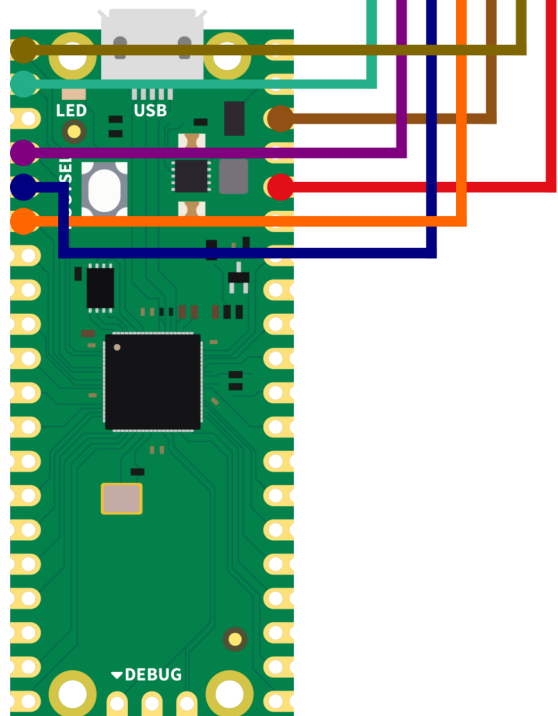
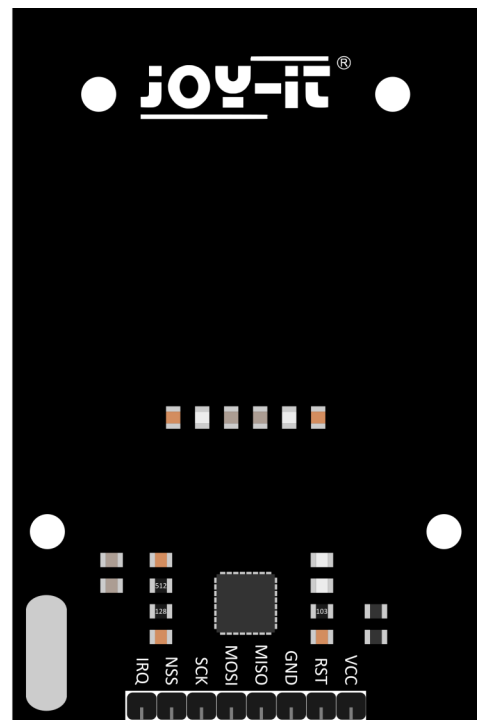
MISO	GP4
------	-----

MOSI	GP3
------	-----

SCK	GP2
-----	-----

NSS	GP1
-----	-----

IRQ	-
-----	---



### 7.2 Example applications

To use the RFID module, you need the [micropython-mfrc522 library](#). Download the provided zip file [here](#). If you already have a folder named "lib" on your Raspberry Pi Pico, simply add the included library to it. Otherwise, create such a folder and add the library inside. Afterward, you can execute the example code.

## 8. INFORMATION AND TAKE-BACK OBLIGATIONS

Our information and take-back obligations according to the Electrical and Electronic Equipment Act (ElektroG)



### Symbol on electrical and electronic equipment:

This crossed-out dustbin means that electrical and electronic appliances do **not** belong in the household waste. You must return the old appliances to a collection point.

Before handing over waste batteries and accumulators that are not enclosed by waste equipment must be separated from it.

### Return options:

As an end user, you can return your old device (which essentially fulfils the same function as the new device purchased from us) free of charge for disposal when you purchase a new device.

Small appliances with no external dimensions greater than 25 cm can be disposed of in normal household quantities independently of the purchase of a new appliance.

### Possibility of return at our company location during opening hours:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn, Germany

### Possibility of return in your area:

We will send you a parcel stamp with which you can return the device to us free of charge. Please contact us by e-mail at [Service@joy-it.net](mailto:Service@joy-it.net) or by telephone.

### Information on packaging:

If you do not have suitable packaging material or do not wish to use your own, please contact us and we will send you suitable packaging.

## 9. SUPPORT

If there are still any issues pending or problems arising after your purchase, we will support you by e-mail, telephone and with our ticket support system.

E-Mail: [service@joy-it.net](mailto:service@joy-it.net)

Ticket system: <http://support.joy-it.net>

Telephone: +49 (0)2845 9360-50 (Mon - Thur: 09:00 - 17:00 o'clock,  
Fri: 09:00 - 14:30 o'clock)

For further information please visit our website:

[www.joy-it.net](http://www.joy-it.net)