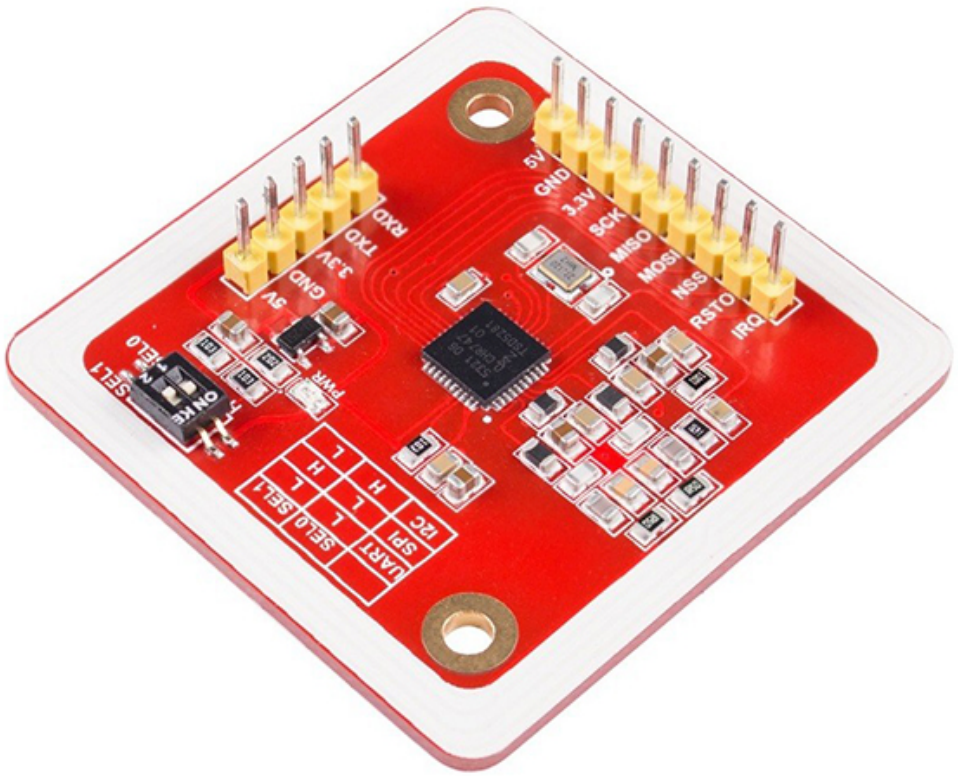
PN532 NFC RFID Module

From Wiki

Contents

Introduction



SunFounder PN532 NFC RFID Module is a highly integrated transmission module for Near Field Communication at 13.56MHz. With the mode switch on board, you can change easily between I2C, SPI, and UART modes. The integrated level shifter provides 3.3V or 5V working voltage for your choice. In addition, it supports RFID reading and writing, and NFC function with Android phone, which makes it quite convenient for wireless connection. This module is equipped with two 3mm mounting holes, of which the small dimension makes it easy for using in your project!
Note: When your wiring is correct (no short circuit), the module may be a little heated, which is

just normal for use.

Schematic Diagram

The diagram is shown as below:
PN532 Schematic.pdf (http://wiki.sunfounder.cc/images/a/af/PN532_Schematic.pdf)

Features

- Supports I2C, SPI and HSU (High Speed UART), easy to switch between these modes.
- Works in the NFC mode and RFID reader/writer mode
- The integrated 3.3V voltage regulator provides 3.3V or 5V working voltage for your choice
- Maximum distance for communication: 3 cm
- Equipped with two 3mm mounting holes, easy to use in your project with its small dimension

Test for Arduino

Preparations

- Arduino Uno/Mega 2560 board (or compatible SunFounder Uno/Mega 2560)
- Dupont Jumper Wires
- Blank NFC card supporting the ISO14443A
- PC with Arduino software installed
- PN532 NFC Module

Note: Pay attention to avoid short circuits between the module's power and the ground, such as the VCC and the GND

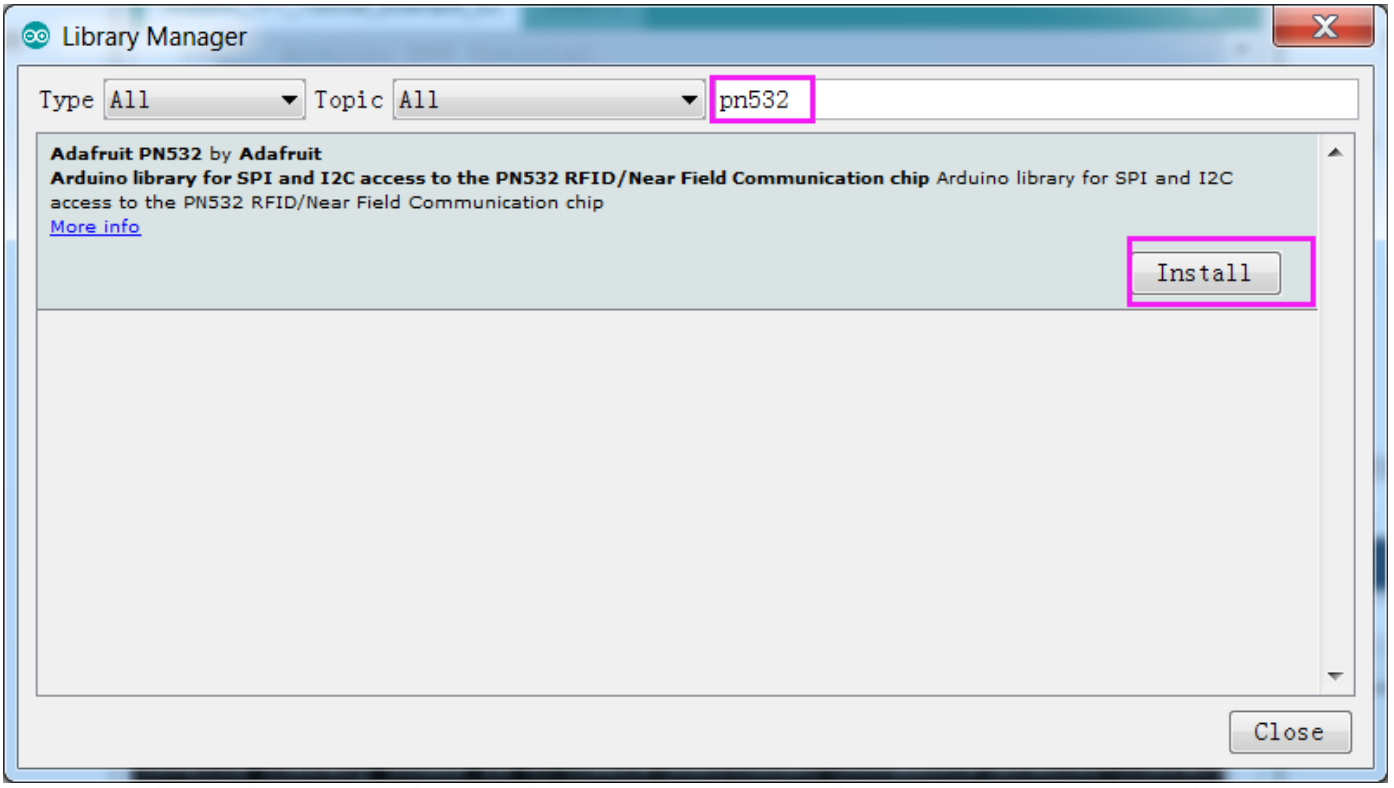
Procedures

The test is implemented on the Uno board in the Arduino IDE V1.6.12.

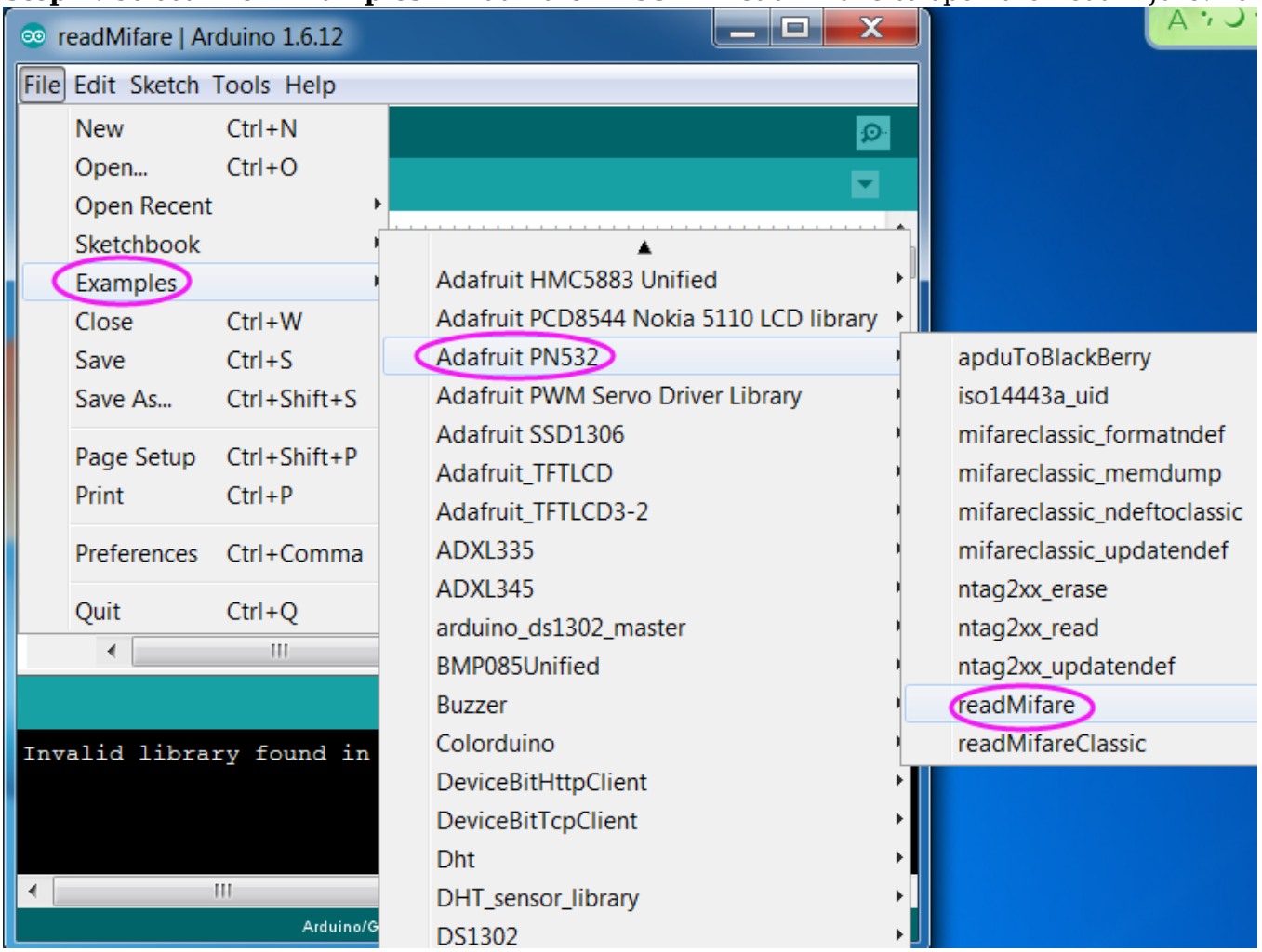
Install the library

Adafruit has a complete library for this, so we will just use its code to test the module directly.

Step 1: In IDE, select **Sketch->Including Library->Manage Libraries**, type in **PN532** to search. Then we can see **Adafruit PN532** searched out, and click **INSTALL** at the right side to start the installation.



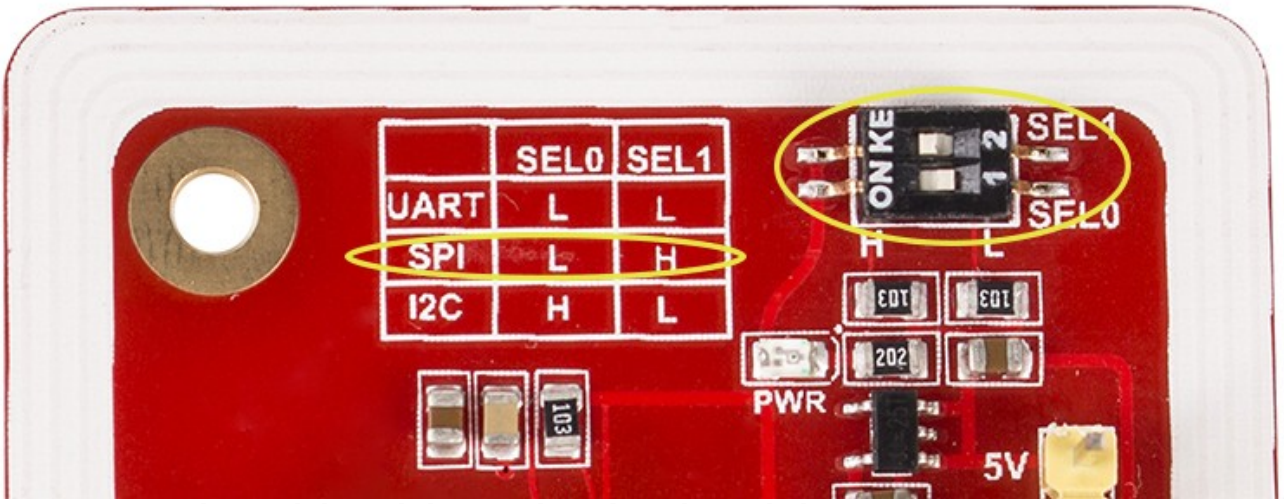
Step 2: Select **File->Examples ->Adafruit PN532->ReadMifare** to open the *ReadMifare.ino*



Test under SPI Mode

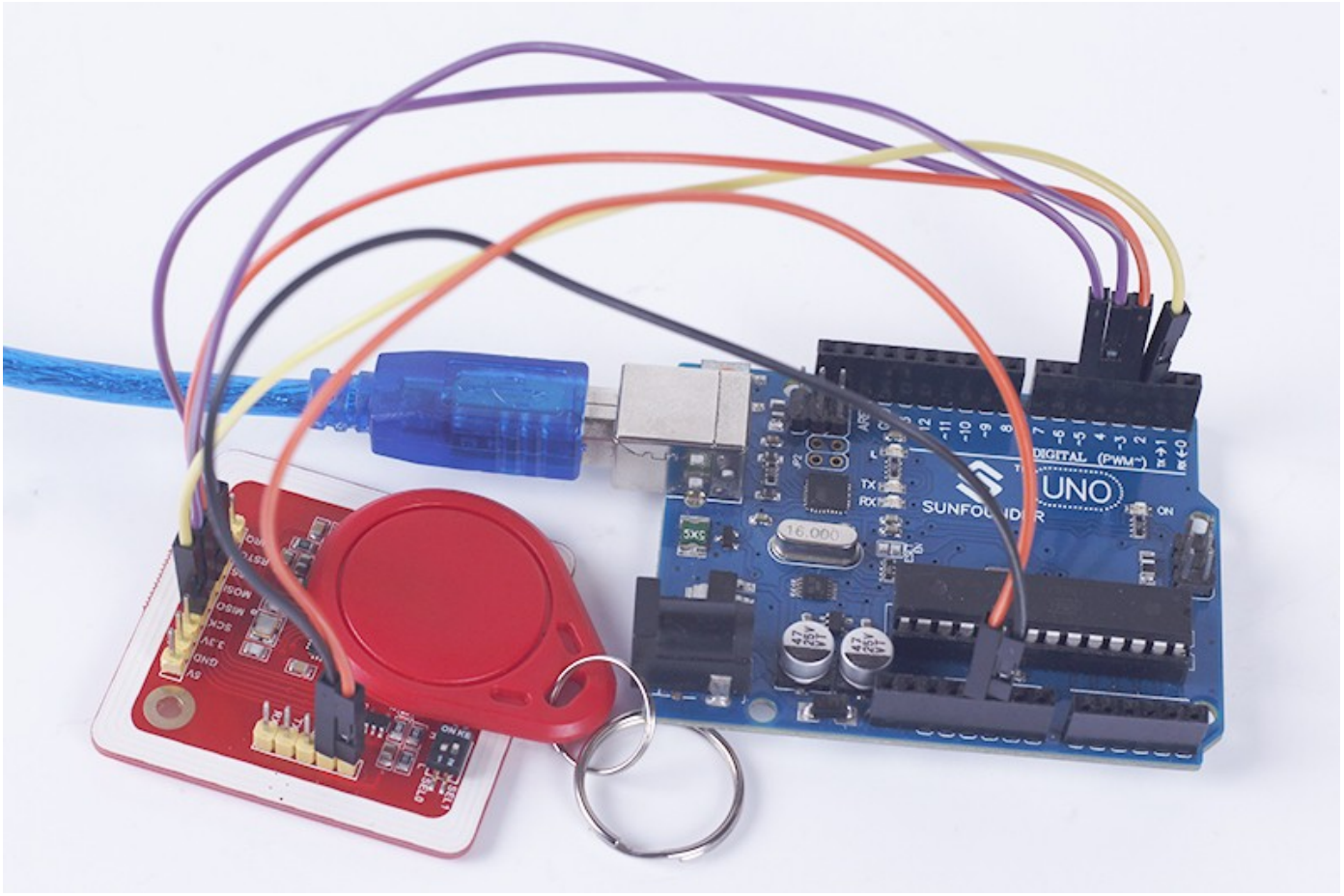
Step 1: Slide the switch to SPI mode:

SEL0	SEL1
L	H



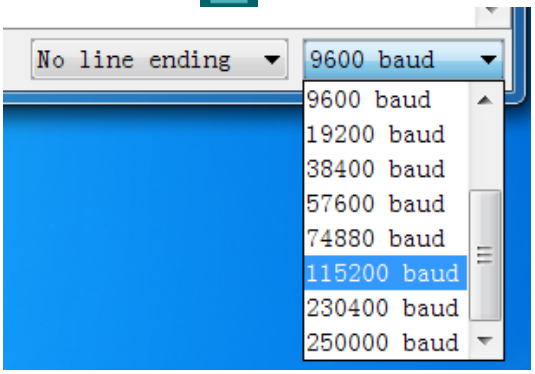
Step 2: Connect the module and the Uno board as shown below:

PN532 Module	SunFounder Uno
GND	GND
5V	5V
SCK	2
MOSI	3
NSS	4
MISO	5

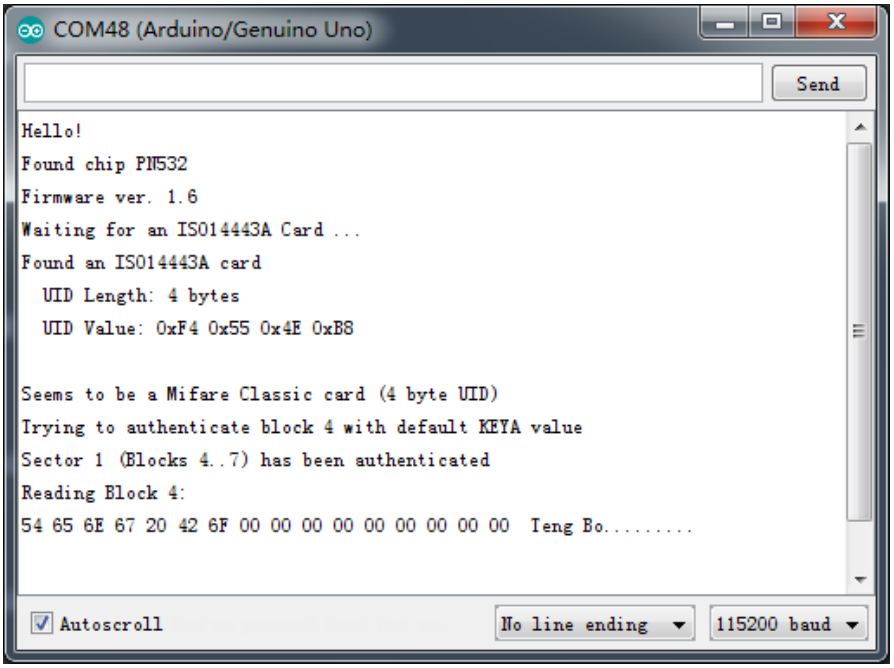


Step 3: Select the corresponding board and port, and upload the code to the Uno.

Step 4: Click  at the upper right corner to open the serial monitor, and select 115200 baud.



Step 5: Thus, you can see the basic information of the card here



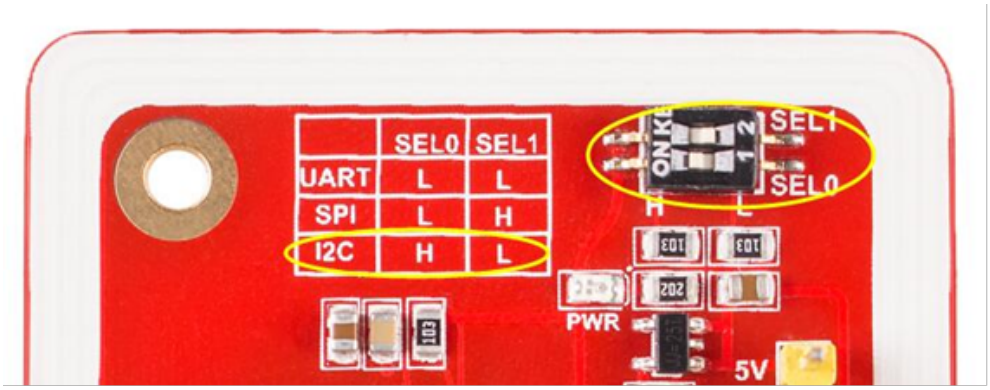
Test under I2C Mode

Step 1: Since the SPI mode is set as default, we need to make some changes to the code before testing under I2C mode. Use "//" to comment out the line56, and activate Line 65 by removing the double slashes (comment mark) as shown below:

```
55 // Use this line for a breakout with a software SPI connection (recommended):
56 //Adafruit_PN532 nfc(PN532_SCK, PN532_MISO, PN532_MOSI, PN532_SS);
57
58 // Use this line for a breakout with a hardware SPI connection.  Note that
59 // the PN532 SCK, MOSI, and MISO pins need to be connected to the Arduino's
60 // hardware SPI SCK, MOSI, and MISO pins.  On an Arduino Uno these are
61 // SCK = 13, MOSI = 11, MISO = 12.  The SS line can be any digital IO pin.
62 //Adafruit_PN532 nfc(PN532_SS);
63
64 // Or use this line for a breakout or shield with an I2C connection:
65 Adafruit_PN532 nfc(PN532_IRQ, PN532_RESET);
66
```

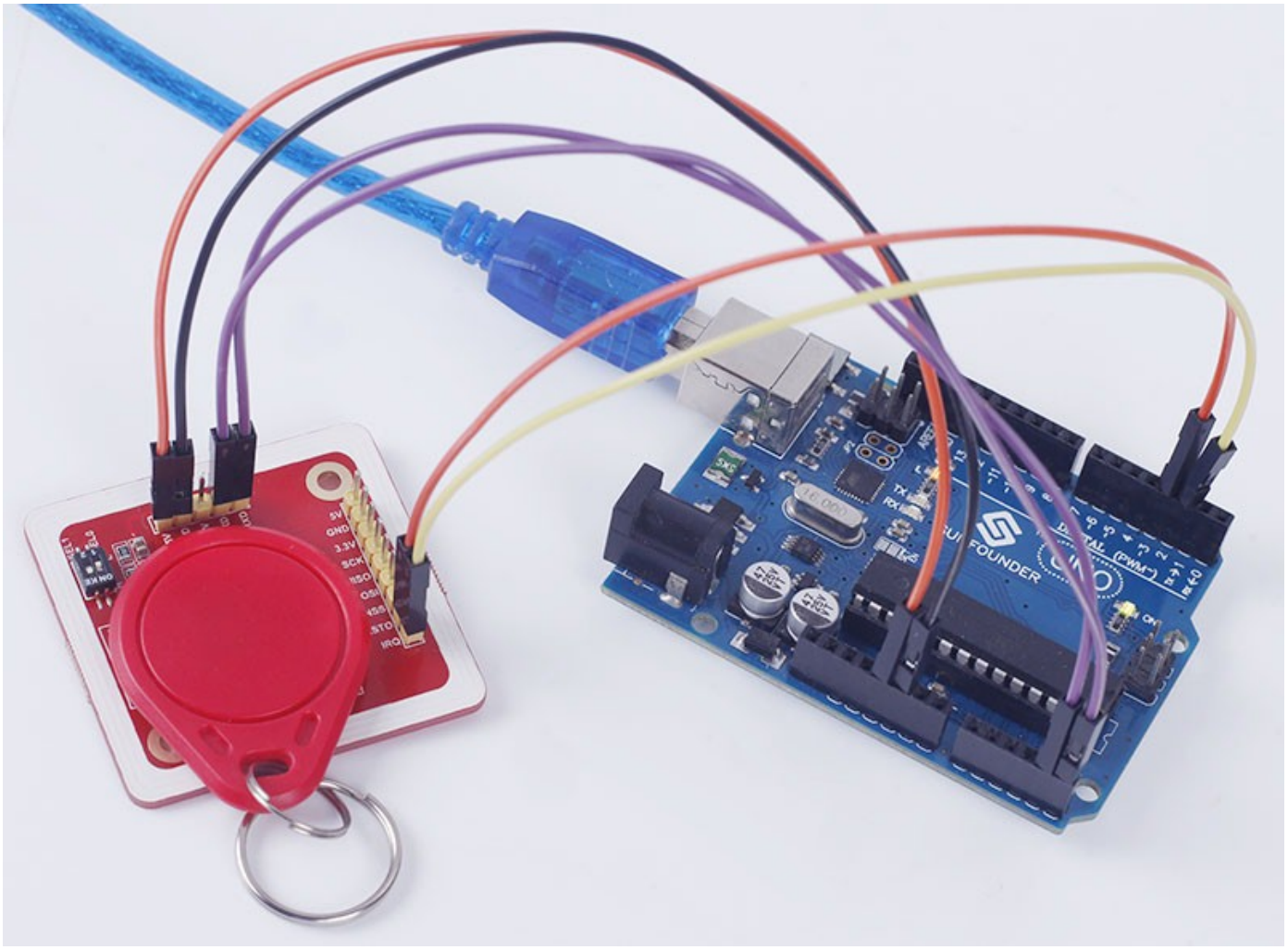
Step 2: Slide the switch to I2C mode on the module:

SEL0	SEL1
H	L

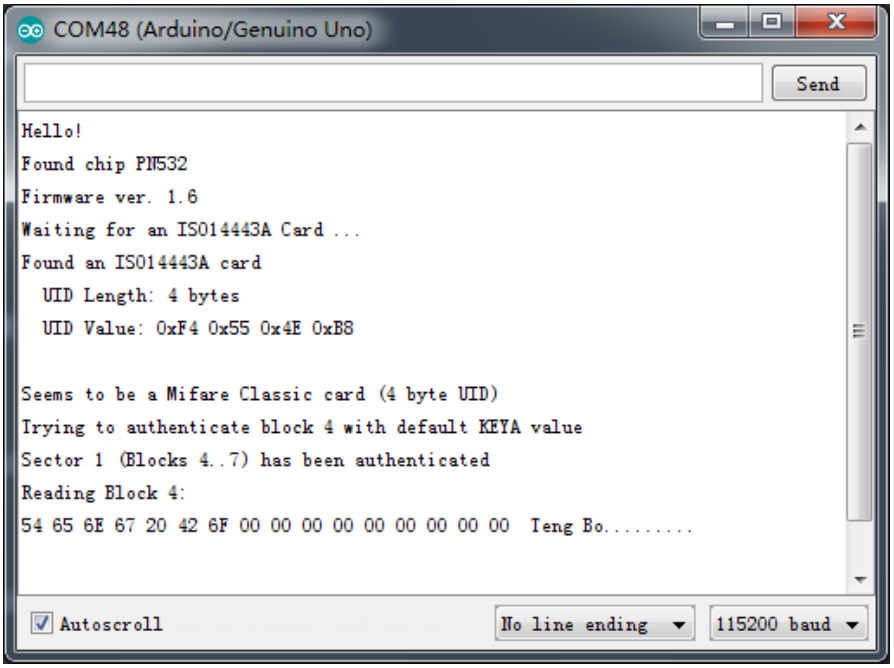


Step 3: Connect the PN532 and the Uno/Mega2560 as shown below:

PN532 Module	Uno/Mega2560
GND	GND
5V	5V
SDA	A4/ pin20 Mega2560
SCL	A5 /pin21 Mega2560
RSTO	3
IRQ	2



Step 4: Upload the program to the board, and open the Serial Monitor. Change the baud rate to 115200, then put the blank NFC card on the PN532 NFC Module, and then you can see the data of the card as shown below: (the UID Value varies from different UID settings.)



Test for Raspberry

Premise

Note: If you have configured libnfc before, please delete the config file.

```
sudo rm –rf /etc/nfc
```

This module is equipped with on-board antenna, so there is no external antenna coil. It is compatible with SPI, IIC interfaces to communicate. With the support of NFC library, Raspberry Pi can connect products with the function of NFC, thus it is easy to use.

I2C Communication Instructions for Raspberry Pi

1. Open I2C of the Raspberry Pi :

```
sudo raspi-config
```

Select **9 Advanced Options** -> **I2C** -> **yes**.

2. Install some dependent packages

```
sudo apt-get update
sudo apt-get install libusb-dev libpcsclite-dev i2c-tools
```

3. Download and unzip the source code package of libnfc

```
cd ~
wget http://dl.bintray.com/nfc-tools/sources/libnfc-1.7.1.tar.bz2
tar -xf libnfc-1.7.1.tar.bz2
```

4. Compile and install

```
cd libnfc-1.7.1
./configure --prefix=/usr --sysconfdir=/etc
make
sudo make install
```

5. Write the configuration file for NFC communication

```
cd /etc
sudo mkdir nfc
sudo nano /etc/nfc/libnfc.conf
```

Check the following details of the file *etc/nfc/libnfc.conf*:

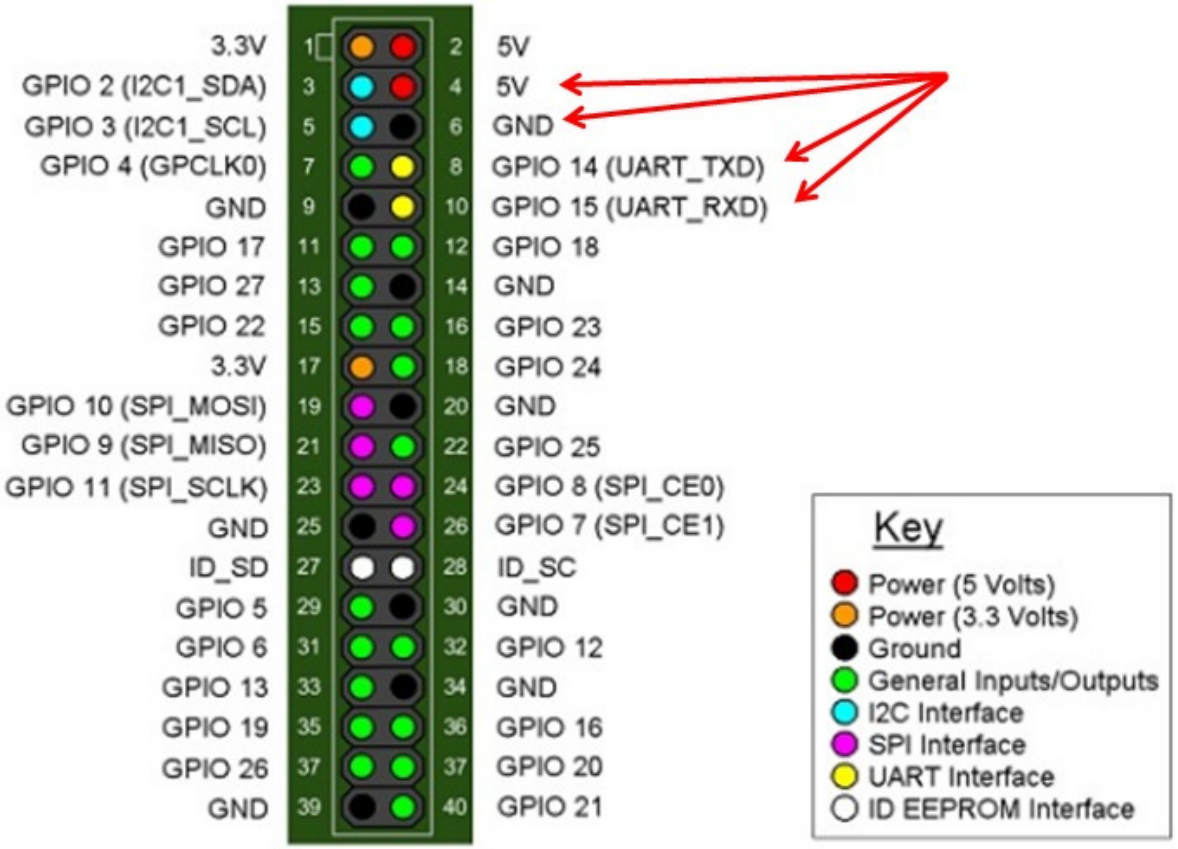
```
# Allow device auto-detection (default: true)
# Note: if this auto-detection is disabled, user has to set manually a device
# configuration using file or environment variable
allow_autoscan = true

# Allow intrusive auto-detection (default: false)
# Warning: intrusive auto-detection can seriously disturb other devices
# This option is not recommended, user should prefer to add manually his device.
allow_intrusive_scan = false

# Set log level (default: error)
# Valid log levels are (in order of verbosity): 0 (none), 1 (error), 2 (info), 3 (debug)
# Note: if you compiled with --enable-debug option, the default log level is "debug"
log_level = 1

# Manually set default device (no default)
# To set a default device, you must set both name and connstring for your device
# Note: if autoscan is enabled, default device will be the first device available in device list.
#device.name = "_PN532_SPI"
#device.connstring = "pn532_spi:/dev/spidev0.0:500000"
device.name = "_PN532_I2c"
device.connstring = "pn532_i2c:/dev/i2c-1"
```

6. Wiring

Toggle the switch to the **I2C mode**

SEL0	SEL1
H	L

Pin diagram of Raspberry pi



Connect the devices:

PN532	Raspberry
5V	5V
GND	GND
SDA	SDA0
SCL	SCL0

7. Run *i2cdetect -y 1* to check whether the I2C device is recognized.
If yes, it means both the module and the wiring work well.
Then type in *nfc-list* to check the NFC module:

```
pi@raspberrypi: ~
pi@raspberrypi:~ $ i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- 24 -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
pi@raspberrypi:~ $ nfc-list
nfc-list uses libnfc 1.7.1
NFC device: pn532_i2c:/dev/i2c-1 opened
pi@raspberrypi:~ $
```

Run *nfc-poll* to scan the RFID tag and you can read information on the card:

```
pi@raspberrypi:~ $ nfc-list
nfc-list uses libnfc 1.7.1
NFC device: pn532_i2c:/dev/i2c-1 opened
pi@raspberrypi:~ $ nfc-poll
nfc-poll uses libnfc 1.7.1
NFC reader: pn532_i2c:/dev/i2c-1 opened
NFC device will poll during 30000 ms (20 pollings of 300 ms for 5 modulations)
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 00  04
       UID (NFCID1): f4  55  4e  b8
      SAK (SEL_RES): 08
nfc_initiator_target_is_present: Target Released
Waiting for card removing...done.
pi@raspberrypi:~ $
```

SPI Communication Instructions for Raspberry Pi

1. Open SPI of the Raspberry Pi:

```
sudo raspi-config
```

Select **9 Advanced Options** -> **SPI** -> **yes**.

2. Install some dependent packages

```
sudo apt-get update
sudo apt-get install libusb-dev libpcsclite-dev i2c-tools
```

3. Download and unzip the source code package of libnfc

```
cd ~
wget http://dl.bintray.com/nfc-tools/sources/libnfc-1.7.1.tar.bz2
tar -xf libnfc-1.7.1.tar.bz2
```

4. Compile and install

```
cd libnfc-1.7.1
./configure --prefix=/usr --sysconfdir=/etc
make
sudo make install
```

5. Write the configuration file for NFC communication

```
cd /etc
sudo mkdir nfc
sudo nano /etc/nfc/libnfc.conf
```

Check the following details of the file *etc/nfc/libnfc.conf*:

```
# Allow device auto-detection (default: true)
# Note: if this auto-detection is disabled, user has to set manually a device
# configuration using file or environment variable
allow_autoscan = true

# Allow intrusive auto-detection (default: false)
# Warning: intrusive auto-detection can seriously disturb other devices
# This option is not recommended, user should prefer to add manually his device.
allow_intrusive_scan = false

# Set log level (default: error)
# Valid log levels are (in order of verbosity): 0 (none), 1 (error), 2 (info), 3 (debug)
# Note: if you compiled with --enable-debug option, the default log level is "debug"
log_level = 1

# Manually set default device (no default)
# To set a default device, you must set both name and connstring for your device
# Note: if autoscan is enabled, default device will be the first device available in device list.
device.name = "_PN532_SPI"
device.connstring = "pn532_spi:/dev/spidev0.0:500000"
#device.name = "_PN532_I2c"
#device.connstring = "pn532_i2c:/dev/i2c-1"
```

6. Wiring

Toggle the switch to the **SPI mode**

SEL0	SEL1
L	H

Connect the devices:

PN532	Raspberry
5V	5V
SCK	SCKL
MISO	MISO
MOSI	MOSI
NSS	CE0

7. Run *ls /dev/spidev0.** to check whether the SPI is opened or not.

If yes, it means both the module and the wiring work well.
Then type in *nfc-list* to check the NFC module:
/dev/spidev0.0 /dev/spidev0.1
If two devices are detected, it means the SPI is already opened.
Then type in *nfc-list* to check the NFC module:

```
pi@raspberrypi:~ $ nfc-list
nfc-list uses libnfc 1.7.1
NFC device: pn532_spi:/dev/spidev0.0 opened
pi@raspberrypi:~ $
```

Run *nfc-poll* to scan the RFID tag and you can read information on the card:

```
pi@raspberrypi:~ $ nfc-poll
nfc-poll uses libnfc 1.7.1
NFC reader: pn532_spi:/dev/spidev0.0 opened
NFC device will poll during 30000 ms (20 pollings of 300 ms for 5 modulations)
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 00  04
       UID (NFCID1): f4  55  4e  b8
      SAK (SEL_RES): 08
nfc_initiator_target_is_present: Target Released
Waiting for card removing...done.
pi@raspberrypi:~ $
```

WARNING:
After test, we know that currently for using the Raspberry Pi 3 model B in the SPI way, there will be an error prompt of TFI Mismatch, while this does not happen to the model B+ and 2 model B.

```
pi@raspberrypi:~ $ nfc-list
nfc-list uses libnfc 1.7.1
error   libnfc.driver.pn532_spi TFI Mismatch
pn53x_check_communication: Input / Output Error
error   libnfc.driver.pn532_spi TFI Mismatch
nfc-list: ERROR: Unable to open NFC device: pn532_spi:/dev/spidev0.0:500000
```

raspberry Pi 3

Retrieved from "http://wiki.sunfounder.cc/index.php?title=PN532_NFC_RFID_Module&oldid=3020"

- This page was last modified on 20 March 2017, at 06:52.