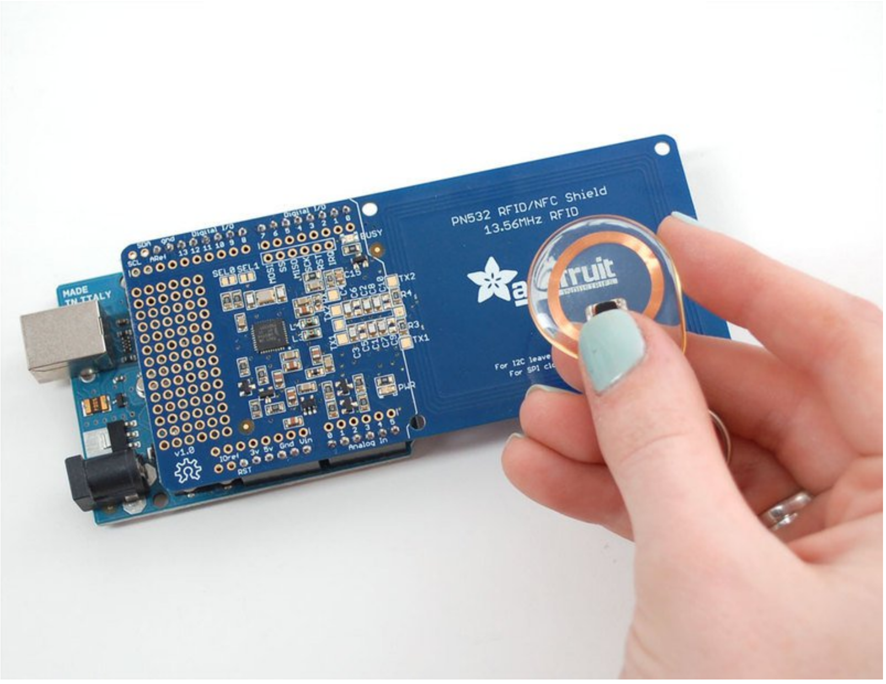
0

- Search Adafruit
- SHOP
- BLOG
- LEARN
- FORUMS
- VIDEOS
- SIGN IN
- CLOSE MENU

0 Items
Sign In

- SHOP
- BLOG
- LEARN
- FORUMS
- VIDEOS



Adafruit PN532 RFID/NFC Breakout and Shield

Radio Frequency ID and Near Field Communication using the PN532

Contributors
lady ada
Feedback? Corrections?
SENSORS / RFID / NFC

MiFare Cards & Tags

by lady ada
MiFare is one of the four 13.56MHz card 'protocols' (FeliCa is another well known one) All of the cards and tags sold at the Adafruit shop use the inexpensive and popular MiFare Classic chipset

MiFare Classic Cards

MIFARE Classic cards come in 1K and 4K varieties. While several varieties of chips exist, the two main chipsets used are described in the following publicly accessible documents:

- MF1S503x Mifare Classic 1K data sheet
- MF1S70yyX MIFARE Classic 4K data sheet

Mifare Classic cards typically have a **4-byte NUID** that uniquely (within the numeric limits of the value) identifies the card. It's possible to have a 7 byte IDs as well, but the 4 byte models are far more common for Mifare Classic.

EEPROM Memory

Mifare Classic cards have either 1K or 4K of EEPROM memory. Each memory block can be configured with different access conditions, with two seperate authentication keys present in each block.

Mifare Classic cards are divided into section called **sectors** and **blocks**. Each "sector" has individual access rights, and contains a fixed number of "blocks" that are controlled by these access rights. Each block contains 16 bytes, and sectors contains either 4 blocks (1K/4K cards) for a total of 64 bytes per sector, or 16 blocks (4K cards only) for a total of 256 bytes per sector. The card types are organised as follows:

- **1K Cards** - 16 sectors of 4 blocks each (sectors 0..15)
- **4K Cards** - 32 sectors of 4 blocks each (sectors 0..31) and 8 sectors of 16 blocks each (sectors 32..39)

4 Block Sectors

1K and 4K cards both use 16 sectors of 4 blocks each, with the bottom 1K of memory on the 4K cards being organised identically to the 1K models for compatability reasons. These individual 4 block sectors (containing 64 byts each) have basic security features are can each be configured with seperate read/write access and two different 6-byte authentication keys (the keys can be different for each sector). Due to these security features (which are stored in the last block, called the **Sector Trailer**), only the bottom 3 blocks of each sector are actually available for data storage, meaning you have 48 bytes per 64 byte sector available for your own use.

Each 4 block sector is organised as follows, with four rows of 16 bytes each for a total of 64-bytes per sector. The first two sectors of any card are shown:
Copy Code

```
1. Sector  Block   Bytes                                                           Description
2.   ------  -----   -----                                                           -----------
3.                   0    1    2    3    4    5    6    7    8    9    10   11   12   13   14   15
```

Copy Code

```
1.   1       3       [-------KEY A-------]   [Access Bits]   [-------KEY B-------]   Sector Trailer
2.           2       [                            Data                         ]   Data
3.           1       [                            Data                         ]   Data
4.           0       [                            Data                         ]   Data
```

Copy Code

```
1.   0       3       [-------KEY A-------]   [Access Bits]   [-------KEY B-------]   Sector Trailer
2.           2       [                            Data                         ]   Data
3.           1       [                            Data                         ]   Data
4.           0       [                     Manufacturer Data                   ]   Manufacturer Block
```

Sector Trailer (Block 3)
The sector trailer block contains the two secret keys (Key A and Key B), as well as the access conditions for the four blocks. It has the following structure:
Copy Code

```
1.        Sector Trailer Bytes
2.        --------------------------------------------------------------
3.        0   1   2   3   4   5   6   7   8   9   10  11  12  13  14  15
4.        [       Key A       ]   [Access Bits]   [       Key B       ]
```

For more information in using Keys to access the clock contents, see Accessing Data Blocks further below.

Data Blocks (Blocks 0..2)
Data blocks are 16 bytes wide and, depending on the permissions set in the access bits, can be read from and written to. You are free to use the 16 data bytes in any way you wish. You can easily store text input, store four 32-bit integer values, a 16 character uri, etc.

Data Blocks as "Value Blocks"
An alternative to storing random data in the 16 byte-wide blocks is to configure them as "Value Blocks". Value blocks allow performing electronic purse functions (valid commands are: read, write, increment, decrement, restore, transfer).

Each Value block contains a single signed 32-bit value, and this value is stored 3 times for data integrity and security reasons. It is stored twice non-inverted, and once inverted. The last 4 bytes are used for a 1-byte address, which is stored 4 times (twice non-inverted, and twice inverted).

Data blocks configured as "Value Blocks" have the following structure:
Copy Code

```
1.        Value Block Bytes
2.        --------------------------------------------------------------
3.        0   1   2   3   4   5   6   7   8   9   10  11  12  13  14  15
4.        [   Value   ]   [   ~Value  ]   [   Value   ]   [A  ~A  A   ~A]
```

Manufacturer Block (Sector 0, Block 0)
Sector 0 is special since it contains the Manufacturer Block. This block contains the manufacturer data, and is read-only. It should be avoided unless you know what you are doing.

16 Block Sectors

16 block sectors are identical to 4 block sectors, but with more data blocks. The same structure described in the 4 block sectors above applies.
Copy Code

```
1.   Sector  Block   Bytes                                                           Description
2.   ------  -----   -----                                                           ----------
3.                   0    1    2    3    4    5    6    7    8    9    10   11   12   13   14   15
```

Copy Code

```
1.   32      15      [-------KEY A-------]   [Access Bits]   [-------KEY B-------]   Sector Trailer 32
2.           14      [                            Data                         ]   Data
3.           13      [                            Data                         ]   Data
4.           ...
5.           2       [                            Data                         ]   Data
6.           1       [                            Data                         ]   Data
7.           0       [                            Data                         ]   Data
```

Accessing EEPROM Memory

To access the EEPROM on the cards, you need to perform the following steps:

1. You must retrieve the 4-byte NUID of the card (this can sometimes be 7-bytes long as well, though rarely for Mifare Classic cards). This is required for the subsequent authentication process.
2. You must authenticate the sector you wish to access according to the access rules defined in the Sector Trailer block for that sector, by passing in the appropriate 6 byte Authentication Key (ex. 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF for new cards).
3. Once authenication has succeeded, and depending on the sector permissions, you can then read/write/increment/decrement the contents of the specific block. Note that you need to re-authenticate for each sector that you access, since each sector can have it's own distinct access keys and rights!

Note on Authentication

Before you can do access the sector's memory, you first need to "authenticate" according to the security settings stored in the Sector Trailer. By default, any new card

will generally be configured to allow full access to every block in the sector using Key A and a value of 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF. Some other common keys that you may wish to try if this doesn't work are:
Copy Code

```
0XFF 0XFF 0XFF 0XFF 0XFF 0XFF
0XD3 0XF7 0XD3 0XF7 0XD3 0XF7
0XA0 0XA1 0XA2 0XA3 0XA4 0XA5
0XB0 0XB1 0XB2 0XB3 0XB4 0XB5
0X4D 0X3A 0X99 0XC3 0X51 0XDD
0X1A 0X98 0X2C 0X7E 0X45 0X9A
0XAA 0XBB 0XCC 0XDD 0XEE 0XFF
0X00 0X00 0X00 0X00 0X00 0X00
0XAB 0XCD 0XEF 0X12 0X34 0X56
```

Example of a New Mifare Classic 1K Card

The follow memory dump illustrates the structure of a 1K Mifare Classic Card, where the data and Sector Trailer blocks can be clearly seen:
Copy Code

```
[-------------------------Start of Memory Dump------------------------]
-----------------------Sector 0------------------------
Block 0  8E 02 6F 66 85 08 04 00 62 63 64 65 66 67 68 69  ?.of?...bcdefghi
Block 1  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 2  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 3  00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 1------------------------
Block 4  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 5  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 6  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 7  00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 2------------------------
Block 8  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 9  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 11 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 3------------------------
Block 12 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 13 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 14 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 15 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 4------------------------
Block 16 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 17 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 18 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 19 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 5------------------------
Block 20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 21 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 22 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 23 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 6------------------------
Block 24 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 25 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 26 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 27 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 7------------------------
Block 28 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 29 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 30 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 31 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 8------------------------
Block 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 35 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 9------------------------
Block 36 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 37 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 38 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 39 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 10------------------------
Block 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 41 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 42 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 43 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 11------------------------
Block 44 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 45 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 46 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 47 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 12------------------------
Block 48 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 49 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 50 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 51 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 13------------------------
Block 52 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 53 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 54 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 55 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 14------------------------
Block 56 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 57 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 58 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 59 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
-----------------------Sector 15------------------------
Block 60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 61 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 62 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
Block 63 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF  ......ÿ.?iÿÿÿÿÿÿ
[-------------------------End of Memory Dump--------------------------]
```

MiFare Ultralight Cards

MiFare Ultralight cards typically contain 512 bits (64 bytes) of memory, including 4 bytes (32-bits) of OTP (One Time Programmable) memory where the individual bits can be written but not erased.

MF0ICU1 MiFare Ultralight Functional Specification

MiFare Ultralight cards have a **7-byte UID** that uniquely identifies the card.

EEPROM Memory

MiFare Ultralight cards have 512 bits (64 bytes) of EEPROM memory, including 4 byte (32 bits) of OTP memory. Unlike Mifare Classic cards, there is no authentication on a per block level, although the blocks can be set to "read-only" mode using Lock Bytes (described below).

EEPROM memory is organised into 16 pages of four bytes eachs, in the following order:
Copy Code

```
Page   Description
----   ------------
0      Serial Number (4 bytes)
1      Serial Number (4 bytes)
2      Byte 0:    Serial Number
       Byte 1:    Internal Memory
       Byte 2..3: lock bytes
3      One-time programmable memory (4 bytes)
4..15  User memory (4 bytes)
```

Here are the pages and blocks arranged in table format:
Copy Code

```
Page   Block 0    Block 1    Block 2   Block 3
-----  --------------------------------------
0      [            Serial Number            ]
1      [            Serial Number            ]
2      [Serial] - [Intern] - [   Lock Bytes  ]
3      [    One Time Programmable Memory     ]
4      [              User Data              ]
5      [              User Data              ]
6      [              User Data              ]
7      [              User Data              ]
8      [              User Data              ]
9      [              User Data              ]
10     [              User Data              ]
11     [              User Data              ]
12     [              User Data              ]
13     [              User Data              ]
14     [              User Data              ]
15     [              User Data              ]
```

Lock Bytes (Page 2)

Bytes 2 and 3 of page 2 are referred to as "Lock Bytes". Each page from 0x03 and higher can individually locked by setting the corresponding locking bit to "1" to prevent further write access, effectively making the memory read only.

For more information on the lock byte mechanism, refer to section 8.5.2 of the datasheet (referenced above).

OTP Bytes (Page 3)

Page 3 is the OTP memory, and by default all bits on this page are set to 0. These bits can be bitwise modified using the MiFare WRITE command, and individual bits can be set to 1, but can not be changed back to 0.

Data Pages (Page 4-15)

Pages 4 to 15 are can be freely read from and written to, provided there is no conflict with the Lock Bytes described above.

After production, the bytes have the following default values:
Copy Code

```
Page    Byte Values
----    ----------------------
        0     1     2     3
4       0xFF  0xFF  0xFF  0xFF
5..15   0x00  0x00  0x00  0x00
```

Accessing Data Blocks

In order to access the cards, you must following two steps:

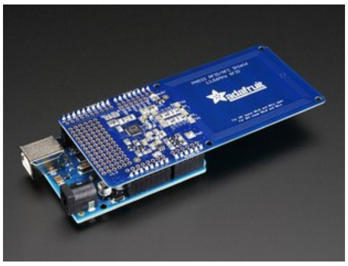
1. 'Connect' to a Mifare Ultralight card and retrieve the 7 byte UID of the card.
2. Memory can be read and written directly once a passive mode connection has been made. No authentication is required for Mifare Ultralight cards.

Read/Write Lengths

For compatability reasons, "Read" requests to a Mifare Ultralight card will retrieve 16 bytes (4 pages) at a time (which corresponds to block size of a Mifare Classic card). For example, if you specify that you want to read page 3, in reality pages 3, 4, 5 and 6 will be read and returned, and you can simply discard the last 12 bytes if they aren't needed. If you select a higher page, the 16 byte read will wrap over to page 0. For example, reading page 14 will actually return page 14, 15, 0 and 1.

"Write" requests occur in pages (4 bytes), so there is no problem with overwriting data on subsequent pages.
ABOUT NFC ABOUT THE NDEF FORMAT
Last updated on 2015-05-04 at 04.27.56 PM Published on 2012-12-30 at 03.44.38 PM

Breadboarding wire bundle
$6.00 ADD TO CART



Adafruit PN532 NFC/RFID Controller Shield for Arduino + Extras
$39.95 ADD TO CART
ADD ALL TO CART

RELATED GUIDES

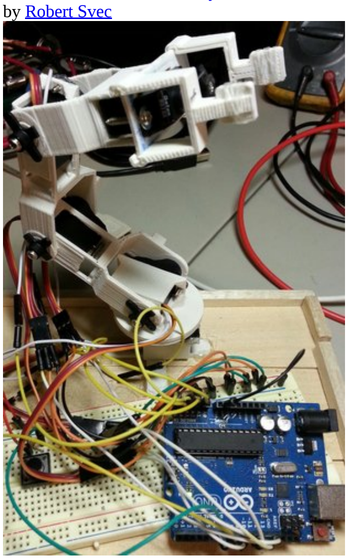
WiFi Controlled LED Christmahanukwanzaa Tree

Control NeoPixels through the web to light up a tree for any holiday occasion!
by Tony DiCola

Control the colors of a NeoPixel strip over a WiFi network using either an Arduino Yun or CC3000 & regular Arduino.

Trainable Robotic Arm

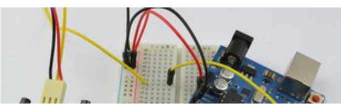
Teach this arm to move with your own hands
by Robert Svec

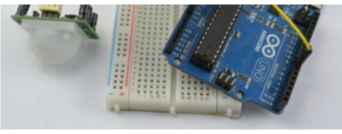


Inspired by the Baxter robot and made possible with the Adafruit Analog Feedback Servos, this robotic arm can be trained to move around by simply manipulating it with your hands. After the Arduino records the motions you taught it, the arm can replay the motion with the press of a button.

Arduino Lesson 17. Email Sending Movement Detector

Use an Arduino to send an email, whenever movement is detected with a PIR sensor.
by Simon Monk





In this lesson you will learn how to use a PIR movement detector with an Arduino and to have the Arduino communicate with a Python program running on your computer to send an email whenever movement is detected by the sensor.

Arduino Lesson 11. LCD Displays - Part 1

Learn Arduino, Lesson 11. LCD Displays - Part 1
by Simon Monk

This is Lesson 11 in the Learn Arduino Adafruit series. In this lesson, you will learn how to wire up and use an alphanumeric LCD display.
×

OUT OF STOCK NOTIFICATION

YOUR NAME
YOUR EMAIL
NOTIFY ME

- CONTACT
- SUPPORT
- DISTRIBUTORS
- EDUCATORS
- JOBS
- FAQ
- SHIPPING & RETURNS
- TERMS OF SERVICE
- PRIVACY & LEGAL
- ABOUT US

ENGINEERED IN NYC Adafruit ®
"Art is I; science is we" - Claude Bernard