## RFID with Arduino integration (MFRC522)

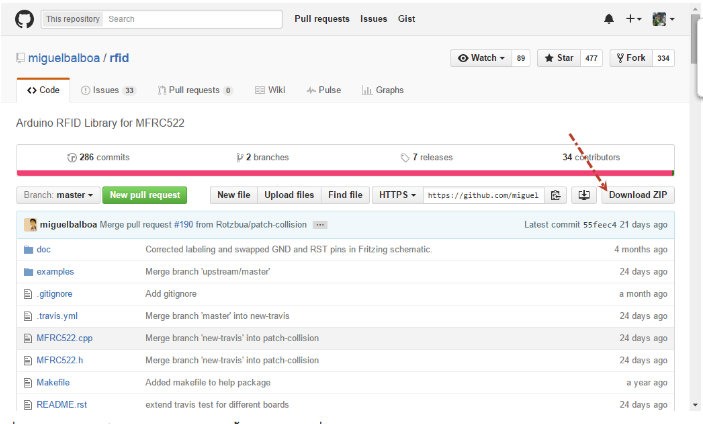
RFID can be very useful. In embedded systems, it is widely used in various applications such as  [clocking systems.](http://www.elec-za.com/clocking-by-linkit-smart-7688-and-aruino/" \t "_blank) Or make the door unlock system etc.

The RFID reader module, MFRC522, can be easily used with the Arduino board. People have written the library to use it already. Which can be downloaded at [https://github.com/miguelbalboa/rfid](https://github.com/miguelbalboa/rfid" \t "_blank) When entering, click the **Download ZIP** button on the right.

[https://github.com/miguelbalboa/rfid](https://github.com/miguelbalboa/rfid" \t "_blank)

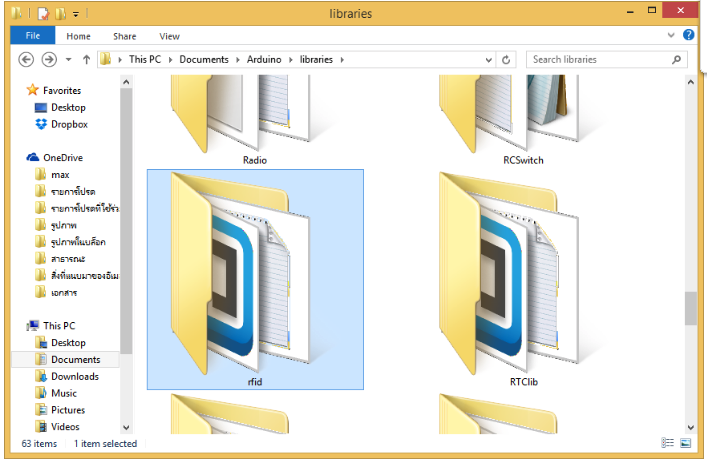
“In the latest release of Arduino (1.8.5 at the time of this writing), it’s easy to include libraries distributed as ZIP files. Simply choose Sketch ⇒   Include Library ⇒   Add .ZIP Library and select the ZIP file containing the library.

sketch>include libraries>add zip library



When the download is complete To extract all files to Arduino

Documents\Arduino\libraries



After the library is installed, restart Arduino and choose Sketch ⇒   Include Library to check that your library is in the list, as shown in Figure 12-5.

sketch>include library> Manage library

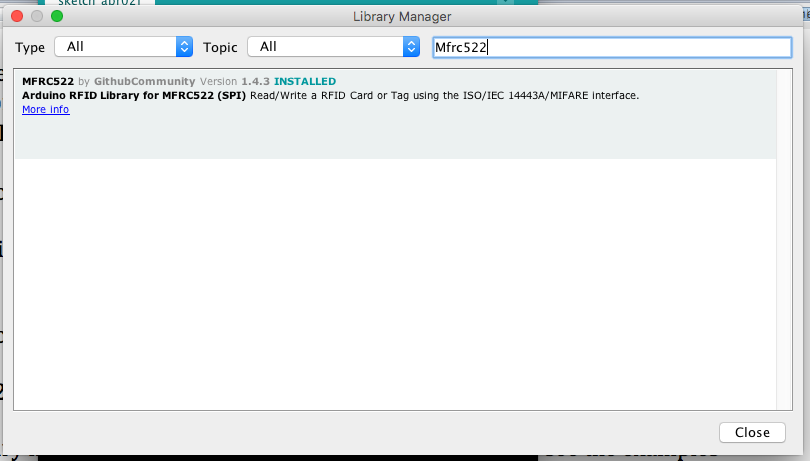


FIGURE 12-5: The Arduino menu shows the library in the Import Library drop-down list.

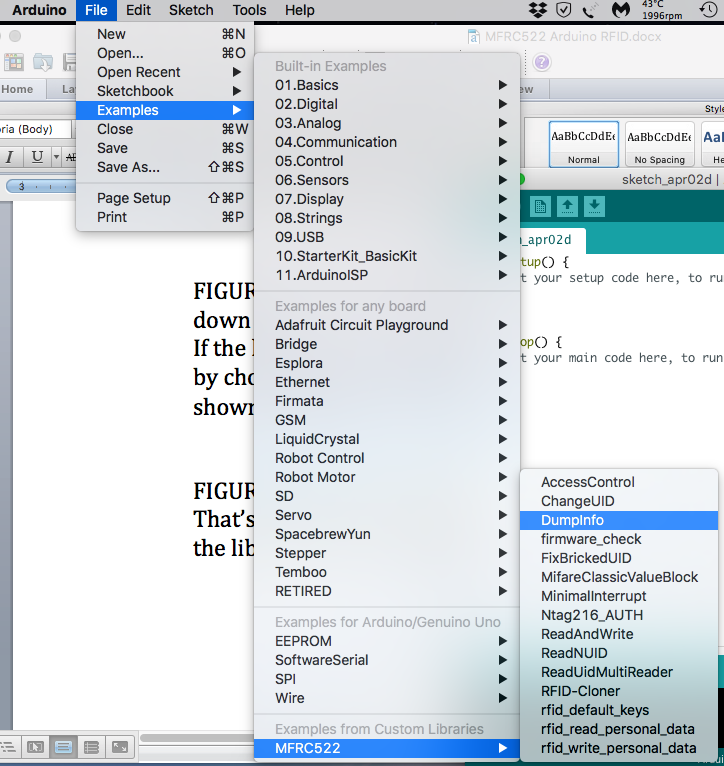
If the library has an Examples folder, you should also be able to see the examples by choosing File ⇒   Examples and then choosing the name of the library, as shown in Figure 12-6.

FIGURE 12-6: If there are examples with the library, you'll see them in the menu.

That’s all there is to installing a library. Removing a library is as simple: just take the library folder out of the Arduino Sketch folder.”

Open the Arduino program, then click on the **File** menu **> Examples.** Scroll to find **MFRC522.**

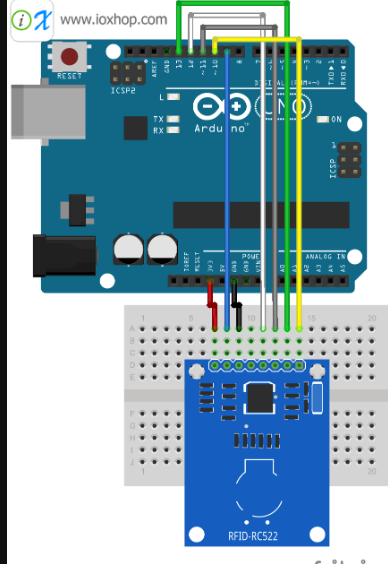
**File > Examples** > **MFRC522**

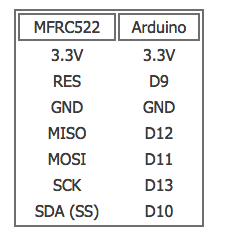


Come back to use the circuit first When the bag is removed, the MFRC-522 module will not be sold to the solder. Must solder the fishbone to make it easier to use before (If you want to plug in the ProBoard, it is recommended to use straight legs. If you want to attach to the box or come to the extension line, recommend using bent legs)

The circuit can be connected as shown below.

\*\*\* Be careful about connecting the voltage to 3.3V. If connected to 5V, the module may be damaged. \*\*\*





Return to the Arduino program to copy the code below. And upload the code to the Arduino board (save the file first too)

read cards

/\*

\* --------------------------------------------------------------------------------------------------------------------

\* Example sketch/program showing how to read data from a PICC to serial.

\* --------------------------------------------------------------------------------------------------------------------

\* This is a MFRC522 library example; for further details and other examples see: https://github.com/miguelbalboa/rfid

\*

\* Example sketch/program showing how to read data from a PICC (that is: a RFID Tag or Card) using a MFRC522 based RFID

\* Reader on the Arduino SPI interface.

\*

\* When the Arduino and the MFRC522 module are connected (see the pin layout below), load this sketch into Arduino IDE

\* then verify/compile and upload it. To see the output: use Tools, Serial Monitor of the IDE (hit Ctrl+Shft+M). When

\* you present a PICC (that is: a RFID Tag or Card) at reading distance of the MFRC522 Reader/PCD, the serial output

\* will show the ID/UID, type and any data blocks it can read. Note: you may see "Timeout in communication" messages

\* when removing the PICC from reading distance too early.

\*

\* If your reader supports it, this sketch/program will read all the PICCs presented (that is: multiple tag reading).

\* So if you stack two or more PICCs on top of each other and present them to the reader, it will first output all

\* details of the first and then the next PICC. Note that this may take some time as all data blocks are dumped, so

\* keep the PICCs at reading distance until complete.

\*

\* @license Released into the public domain.

\*

\* Typical pin layout used:

\* -----------------------------------------------------------------------------------------

\* MFRC522 Arduino Arduino Arduino Arduino Arduino

\* Reader/PCD Uno/101 Mega Nano v3 Leonardo/Micro Pro Micro

\* Signal Pin Pin Pin Pin Pin Pin

\* -----------------------------------------------------------------------------------------

\* RST/Reset RST 9 5 D9 RESET/ICSP-5 RST

\* SPI SS SDA(SS) 10 53 D10 10 10

\* SPI MOSI MOSI 11 / ICSP-4 51 D11 ICSP-4 16

\* SPI MISO MISO 12 / ICSP-1 50 D12 ICSP-1 14

\* SPI SCK SCK 13 / ICSP-3 52 D13 ICSP-3 15

\*/

#include <SPI.h>

#include <MFRC522.h>

#define RST\_PIN 9 // Configurable, see typical pin layout above

#define SS\_PIN 10 // Configurable, see typical pin layout above

MFRC522 mfrc522(SS\_PIN, RST\_PIN); // Create MFRC522 instance

void setup() {

Serial.begin(9600); // Initialize serial communications with the PC

while (!Serial); // Do nothing if no serial port is opened (added for Arduinos based on ATMEGA32U4)

SPI.begin(); // Init SPI bus

mfrc522.PCD\_Init(); // Init MFRC522

mfrc522.PCD\_DumpVersionToSerial(); // Show details of PCD - MFRC522 Card Reader details

Serial.println(F("Scan PICC to see UID, SAK, type, and data blocks..."));

}

void loop() {

// Look for new cards

if ( ! mfrc522.PICC\_IsNewCardPresent()) {

return;

}

// Select one of the cards

if ( ! mfrc522.PICC\_ReadCardSerial()) {

return;

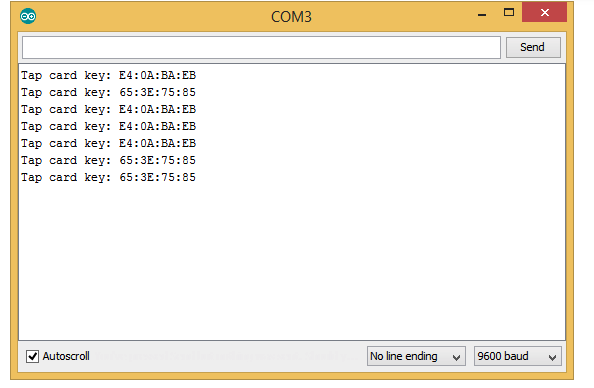
}

// Dump debug info about the card; PICC\_HaltA() is automatically called

mfrc522.PICC\_DumpToSerial(&(mfrc522.uid));

}

Once the upload is complete, open the Serial Moniter and set Baud rate to 9600. Try to use both tags to touch the different UID values.



This UID (or key) will be stored in the strID variable, which will be used after line 41. The code example below is to apply the RFID tag to control the LED light.

//read card led on

#include "SPI.h"

#include "MFRC522.h"

#define SS\_PIN 10

#define RST\_PIN 9

#define LED\_PIN A0

MFRC522 rfid(SS\_PIN, RST\_PIN);

MFRC522::MIFARE\_Key key;

void setup() {

Serial.begin(9600);

SPI.begin();

rfid.PCD\_Init();

pinMode(LED\_PIN, OUTPUT);

}

void loop() {

if (!rfid.PICC\_IsNewCardPresent() || !rfid.PICC\_ReadCardSerial())

return;

// Serial.print(F("PICC type: "));

MFRC522::PICC\_Type piccType = rfid.PICC\_GetType(rfid.uid.sak);

// Serial.println(rfid.PICC\_GetTypeName(piccType));

// Check is the PICC of Classic MIFARE type

if (piccType != MFRC522::PICC\_TYPE\_MIFARE\_MINI &&

piccType != MFRC522::PICC\_TYPE\_MIFARE\_1K &&

piccType != MFRC522::PICC\_TYPE\_MIFARE\_4K) {

Serial.println(F("Your tag is not of type MIFARE Classic."));

return;

}

String strID = "";

for (byte i = 0; i < 4; i++) {

strID +=

(rfid.uid.uidByte[i] < 0x10 ? "0" : "") +

String(rfid.uid.uidByte[i], HEX) +

(i!=3 ? ":" : "");

}

strID.toUpperCase();

Serial.print("Tap card key: ");

Serial.println(strID);

// LED strID -one card to turn on other to tutn off

if (strID.indexOf("E4:0A:BA:EB") >= 0) {

digitalWrite(LED\_PIN, HIGH);

} else {

digitalWrite(LED\_PIN, LOW);

}

rfid.PICC\_HaltA();

rfid.PCD\_StopCrypto1();

}

1. File -Eamples –MFRC522- Dump
2. Validate then Upload , then place card to get their uid’s

my Card uid 79 54 A5 55

my phob uid 59 9A AB 47

1. Use upload led code
2. Change uid to one of the cards, then validate and upload
3. One card will set on LED other will switch off the LED, + to A0 , - to GND.

