Nom : ....

OBJECTIF :
SUPPORT D'IDENTIFICATION, LECTEURS, UNITÉS DE TRAITEMENT, GESTION DE COMMUNICATION

L'utilisation MF RC522 met en oeuvre la modulation et la démodulation à 13.56MHz. Elles sont complètement intégrées avec toutes sortes de méthodes de communication sans contact et de protocoles. La partie numérique gère les cadres ISO14443A et la détection d'erreur, l'algorithme de cryptage Quick CRYPTO1, et le terme de vérification série MIFARE.

Le module MFRC522 prend en charge le protocole de communications sans contact à haute vitesse MIFARE, avec des taux de transfert de données bidirectionnels jusqu'à 424 kbit/s. Le module MF522 utilise un composant d'origine Philips MFRC522 lecteur de carte de circuit. Il est facile à utiliser, faible coût, adapté au développement de l'équipement. Il prend en compte la nécessité de conception de terminal RF / utilisateurs de production. Ce module peut être directement chargé dans la multitude de forme de lecteur. Le module utilise une tension de 3.3V. Avec seulement quelques lignes simples à travers l'interface SPI, tout utilisateur connecte directement le CPU au module de communication, et peut de plus garantir un travail stable et fiable, distance du lecteur.

MATERIEL

Current:13-26mA / DC 3.3V
Idle Current:10-13mA / DC 3.3V
Sleep current: <80uA
Peak current: <30mA
Operating Frequency: 13.56MHz
Environmental Operating temperature: -20 to 80 degrees Celsius
Environment Storage temperature: -40 to 85 degrees Celsius
Relative Humidity: 5% -95%
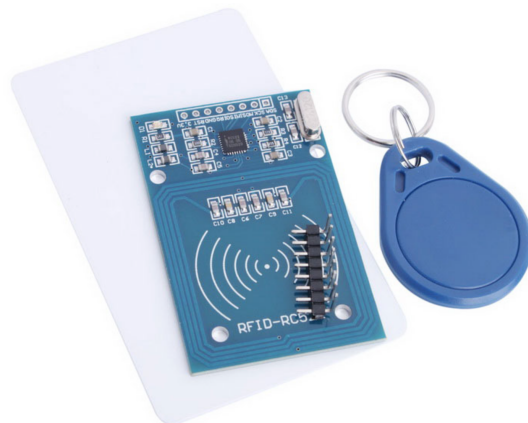
tag RFID et son lecteurs

LIRE LE TAG

1 Est-ce que le dispositif comporte une source d'énergie comme une pile ?

....

2 Quelle est alors l'origine de l'énergie faisant marcher les composants sur le tag ?

....

3 A quel composant d'électronique de puissance vous fait penser cette gestion de l'énergie ?

....

4 Une fois que votre montage marchera, tester la distance maximale à laquelle la communication fonctionne ?

....

5 Installer la librairie du RC522 (voir figure 1) et la tester.

6 Branchement
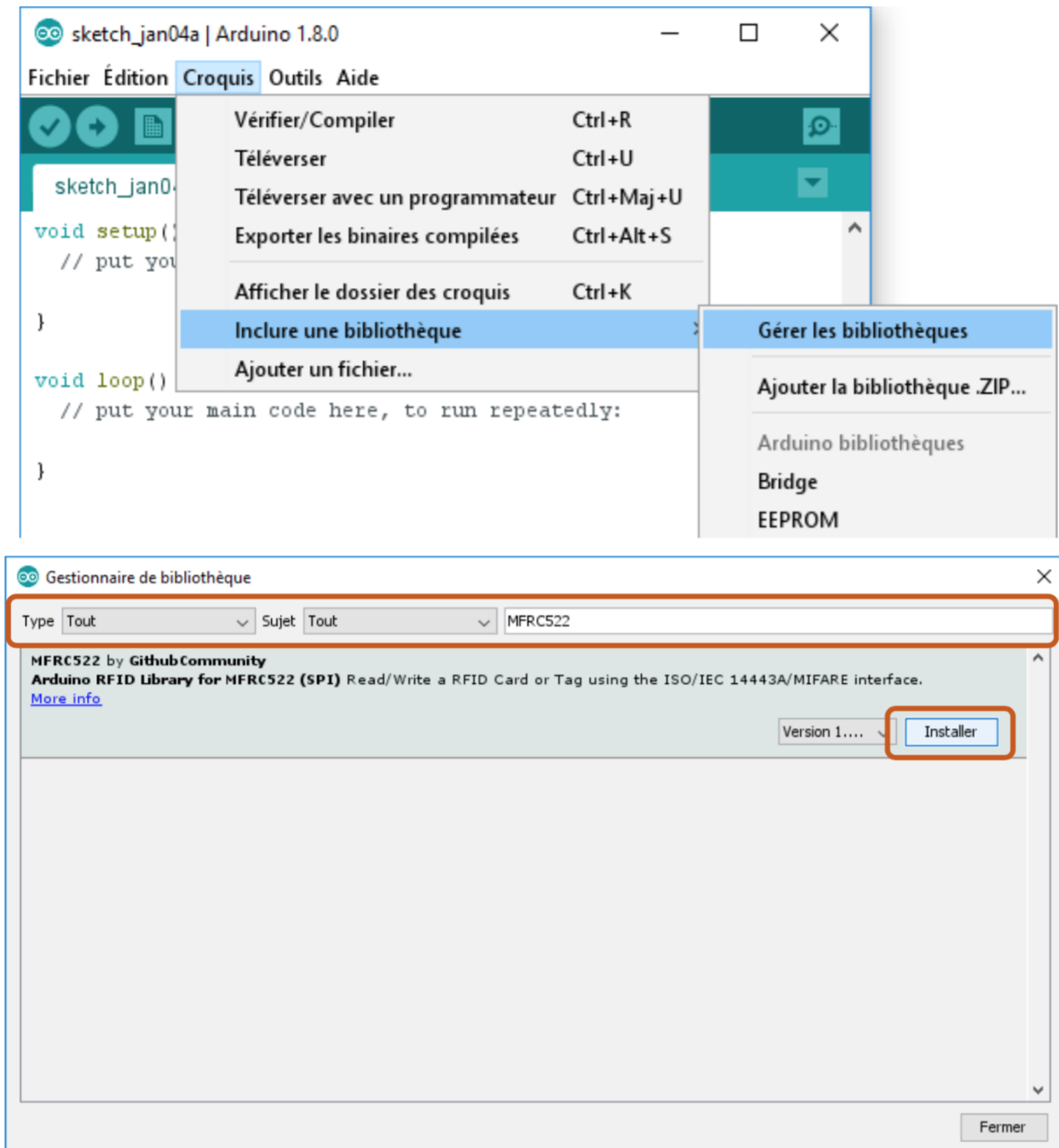
Figure 1: installation de la librairie SPI pour la carte RFID
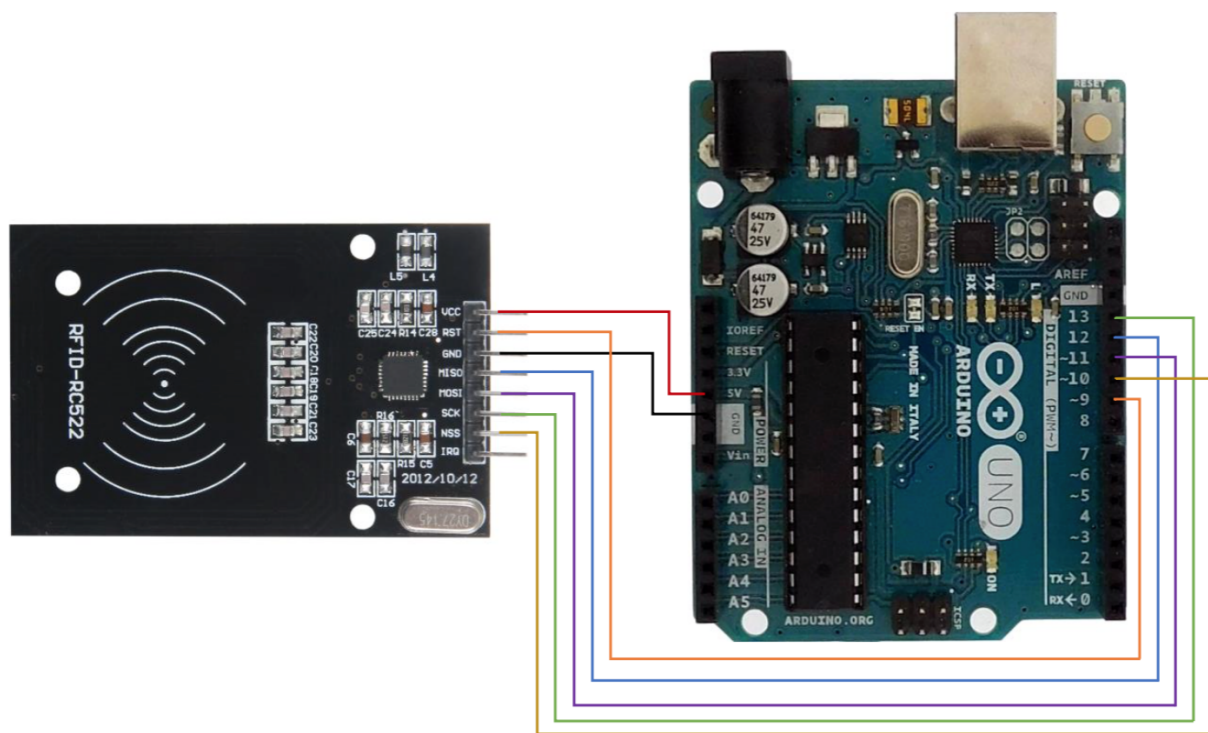
RFID RC522	Arduino
VCC	+5V
RST	9
GND	GND
MISO	12
MOSI	11
SCK	13
NSS	10
IRC	/

Figure 2: Connection du lecteur RFID à la carte arduino

7 Quel est le bus qui relie le arduino et le lecteur de tag RFID ?

....

8 Decrire le role de chaque fil ?

....

....

....

9 Implémentez le code suivant : [1]

```
1  /*
2   * MFRC522 - Library to use ARDUINO RFID MODULE KIT 13.56 MHZ WITH TAGS SPI W AND R BY COOQROBOT.
3   * The library file MFRC522.h has a wealth of useful info. Please read it.
4   * The functions are documented in MFRC522.cpp.
5   *
```

[1] http://makecourse.weebly.com/week10segment1.html

```
6     * Based on code Dr.Leong    ( WWW.B2CQSHOP.COM )
7     * Created by Miguel Balboa (circuitito.com), Jan, 2012.
8     * Rewritten by Soren Thing Andersen (access.thing.dk), fall of 2013
9     * (Translation to English, refactored, comments, anti collision, cascade levels.)
10    * Released into the public domain.
11    *
12    * Sample program showing how to read data from a PICC using a MFRC522 reader on the Arduino SPI interface.
13    *----------------------------------------------------------------------- empty_skull
14    * Aggiunti pin per arduino Mega
15    * add pin configuration for arduino mega
16    * http://mac86project.altervista.org/
17    *----------------------------------------------------------------------- Nicola Coppola
18    * Pin layout should be as follows:
19    * Signal        Pin           Pin           Pin
20    *               Arduino Uno   Arduino Mega   MFRC522 board
21    * ------------------------------------------------------
22    * Reset         9             5             RST
23    * SPI SS        10            53            SDA
24    * SPI MOSI      11            52            MOSI
25    * SPI MISO      12            51            MISO
26    * SPI SCK       13            50            SCK
27    *
28    * The reader can be found on eBay for around 5 dollars. Search for "mf-rc522" on ebay.com.
29    */
30
31    #include <SPI.h>
32    #include <MFRC522.h>
33
34    #define SS_PIN 10
35    #define RST_PIN 9
36    MFRC522 mfrc522(SS_PIN, RST_PIN);         // Create MFRC522 instance.
37
38    void setup() {
39      Serial.begin(9600);    // Initialize serial communications with the PC
40      SPI.begin();                   // Init SPI bus
41      mfrc522.PCD_Init();    // Init MFRC522 card
42      Serial.println("Scan_PICC_to_see_UID_and_type...");
43    }
44
45    void loop() {
46      // Look for new cards
47      if ( ! mfrc522.PICC_IsNewCardPresent()) {
48        return;//go to start of loop if there is no card present
49      }
50      // Select one of the cards
51      if ( ! mfrc522.PICC_ReadCardSerial()) {
52        //if ReadCardSerial returns 1,
53        // the "uid" struct (see MFRC522.h lines 238-45)) contains the ID of the read card.
54        return;
55      }
56      // Dump debug info about the card. PICC_HaltA() is automatically called.
57      mfrc522.PICC_DumpToSerial(&(mfrc522.uid));
58    }
```

10 Lecture de la carte

Constatation professeur :

LIRE LE TAG

11 Ecrire le tag avec des données de votre choix

```
1
2     /*****************************************
3             PURPOSE:          Learn to use the MF522-AN RFID card reader
4             Created by        Rudy Schlaf for www.makecourse.com
5             DATE:             2/2014
6     *****************************************/
7
8     /*
9      * This sketch uses the MFRC522 Library to use ARDUINO RFID MODULE KIT 13.56 MHZ WITH TAGS SPI W
10     * AND R BY COOQROBOT.
11     * The library file MFRC522.h has a wealth of useful info. Please read it.
12     * The functions are documented in MFRC522.cpp.
13     *
14     * Based on code Dr.Leong    ( WWW.B2CQSHOP.COM )
15     * Created by Miguel Balboa (circuitito.com), Jan, 2012.
16     * Rewritten by Soren Thing Andersen (access.thing.dk), fall of 2013 (Translation to English,
17     * refactored, comments, anti collision, cascade levels.)
18     *
19     * This library has been released into the public domain.
20     */
21
22
23    #include <SPI.h>//include the SPI bus library
24    #include <MFRC522.h>//include the RFID reader library
25
26    #define SS_PIN 10   //slave select pin
27    #define RST_PIN 5   //reset pin
28    MFRC522 mfrc522(SS_PIN, RST_PIN);         // instatiate a MFRC522 reader object.
29    MFRC522::MIFARE_Key key;//create a MIFARE_Key struct named 'key', which will hold the card information
30
31
```

```
32  void setup() {
33          Serial.begin(9600);              // Initialize serial communications with the PC
34          SPI.begin();                     // Init SPI bus
35          mfrc522.PCD_Init();              // Init MFRC522 card (in case you wonder what PCD means: proximity
36                                           //  coupling device)
37          Serial.println("Scan_a_MIFARE_Classic_card");
38
39          // Prepare the security key for the read and write functions — all six key bytes are set to
40          //   0xFF at chip delivery from the factory.
41          // Since the cards in the kit are new and the keys were never defined, they are 0xFF
42          // if we had a card that was programmed by someone else, we would need to know the key to
43          //   be able to access it. This key would then need to be stored in 'key' instead.
44
45          for (byte i = 0; i < 6; i++) {
46                  key.keyByte[i] = 0xFF;//keyByte is defined in the "MIFARE_Key" 'struct' definition in the
47                                        //  .h file of the library
48          }
49
50  }
51
52  int block=2; //this is the block number we will write into and then read. Do not write
53                  //  into 'sector trailer' block, since this can make the block unusable.
54
55  //an array with 16 bytes to be written into one of the 64 card blocks is defined
56  byte blockcontent[16] = {"makecourse_____"};
57  //byte blockcontent[16] = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};//all zeros.
58  //This can be used to delete a block.
59  //This array is used for reading out a block. The MIFARE_Read method
60  //requires a buffer that is at least 18 bytes to hold the 16 bytes of a block.
61  byte readbackblock[18];
62
63  void loop()
64  {
65
66    /********************************establishing contact with a tag/card********************************************************/
67
68    // Look for new cards (in case you wonder what PICC means: proximity integrated circuit card)
69    if ( ! mfrc522.PICC_IsNewCardPresent()) {//if PICC_IsNewCardPresent returns 1, a new card has been found and we continue
70      return;//if it did not find a new card is returns a '0' and we return to the start of the loop
71    }
72
73    // Select one of the cards
74    //if PICC_ReadCardSerial returns 1, the "uid" struct (see MFRC522.h lines 238−45) contains the ID of the read card.
75    if ( ! mfrc522.PICC_ReadCardSerial()) {
76      //if it returns a '0' something went wrong and we return to the start of the loop
77      return;
78    }
79
80    // Among other things, the PICC_ReadCardSerial() method reads the UID and the SAK (Select acknowledge) into the mfrc522.uid struct,
81    //   which is also instantiated during this process.
82    // The UID is needed during the authentication process
83    //The Uid struct:
84    //typedef struct {
85    //byte              size;                    // Number of bytes in the UID. 4, 7 or 10.
86    //byte              uidByte[10];             //the user ID in 10 bytes.
87    //byte              sak;                     // The SAK (Select acknowledge) byte returned from the PICC after successful selection.
88    //} Uid;
89
90    Serial.println("card_selected");
91
92    /********************************writing and reading a block on the card********************************************************/
93
94    writeBlock(block, blockcontent);//the blockcontent array is written into the card block
95    //mfrc522.PICC_DumpToSerial(&(mfrc522.uid));
96
97    //The 'PICC_DumpToSerial' method 'dumps' the entire MIFARE data block into the serial monitor.
98    \\Very useful while programming a sketch with the RFID reader...
99    //Notes:
100   //(1) MIFARE cards conceal key A in all trailer blocks, and shows 0x00 instead of 0xFF.
101   //    This is a secutiry feature. Key B appears to be public by default.
102   //(2) The card needs to be on the reader for the entire duration of the dump.
103   //    If it is removed prematurely, the dump interrupts and an error message will appear.
104   //(3) The dump takes longer than the time alloted for interaction per pairing between reader and card,
105   //    i.e. the readBlock function below will produce a timeout if the dump is used.
106
107   //mfrc522.PICC_DumpToSerial(&(mfrc522.uid));//uncomment this if you want to see the entire 1k memory with the block written into it.
108
109   readBlock(block, readbackblock);//read the block back
110   Serial.print("read_block:_");
111   for (int j=0 ; j<16 ; j++)//print the block contents
112   {
113     Serial.write (readbackblock[j]);//Serial.write() transmits the ASCII numbers as human readable characters to serial monitor
114   }
115   Serial.println("");
116  }
```

Constatation professeur :