

Arduino e RC522: Scrivere dati su un tag RFID

Arduino è interfacciato con un dispositivo RFID RC522 che legge un tag RFID (una carta magnetica o un tag portachiavi). Se in base al codice letto riconosce l'oggetto, accende un diodo led verde, viceversa accende un diodo led rosso.

- al posto della carta magnetica e del tag portachiavi si può utilizzare un qualsiasi oggetto con tag RFID
- l'accensione del diodo verde simula l'azione che viene intrapresa: l'azionamento di un relé per aprire una porta o per accendere un motore, il tracciamento dell'oggetto, l'aggiornamento del magazzino, il monitoraggio dell'avanzamento della produzione, ecc.

Le tecnologie RFID

Le tecnologie **RFID** sono contrassegnate da un codice univoco, e quindi da un'identità digitale, e rappresentano una delle leve principali per lo sviluppo dell'Industria 4.0 e dell'Internet delle cose (IoT). RFID è l'acronimo di "Radio Frequency Identification" ovvero "identificazione a radiofrequenza" e si riferisce a una tecnologia in base alla quale i dati digitali codificati in un **tag RFID** vengono acquisiti da un dispositivo di lettura (**Reader**) tramite onde radio. L'RFID è simile al codice a barre ma presenta il grande vantaggio che i dati dei tag RFID possono essere letti in prossimità anche senza "contatto visivo": al contrario i codici a barre devono essere allineati con uno scanner ottico.

Un sistema RFID è costituito da due componenti principali:

1. **un transponder a radiofrequenza (Tag)** inserito nell'oggetto che si vuole identificare: il tag contiene un microchip, che memorizza ed elabora le informazioni, e un'antenna che consente la ricezione/trasmmissione di dati a corto raggio senza contatto fisico. Il tag è generalmente un dispositivo passivo, ovvero non contiene una batteria
2. **un ricetrasmittitore (Reader)** controllato da un microprocessore ed usato per leggere le informazioni contenute nel tag



- Il tag viene posizionato in prossimità del Reader (non deve necessariamente trovarsi "a vista" del Reader)
- Il Reader genera un campo elettromagnetico che fa muovere gli elettroni attraverso l'antenna del tag e alimenta il chip
- Il chip tramite un segnale radio trasmette al Reader le informazioni memorizzate al suo interno
- il Reader rileva e interpreta il segnale radio ricevuto e, attraverso un'interfaccia periferica seriale, invia i dati a un computer o ad un microcontrollore, nel nostro caso ad Arduino, che li gestisce per realizzare gli scopi dell'applicazione (ad esempio per azionare l'apertura di una porta o di un tornello)

Il Reader RFID RC522

Il modulo RFID RC522 è uno dei Reader RFID più diffusi ed economici e viene fornito di solito con una o più schede RFID e uno più tag portachiavi con 1 KB di memoria. Inoltre è in grado di scrivere un tag e di memorizzarvi un testo.

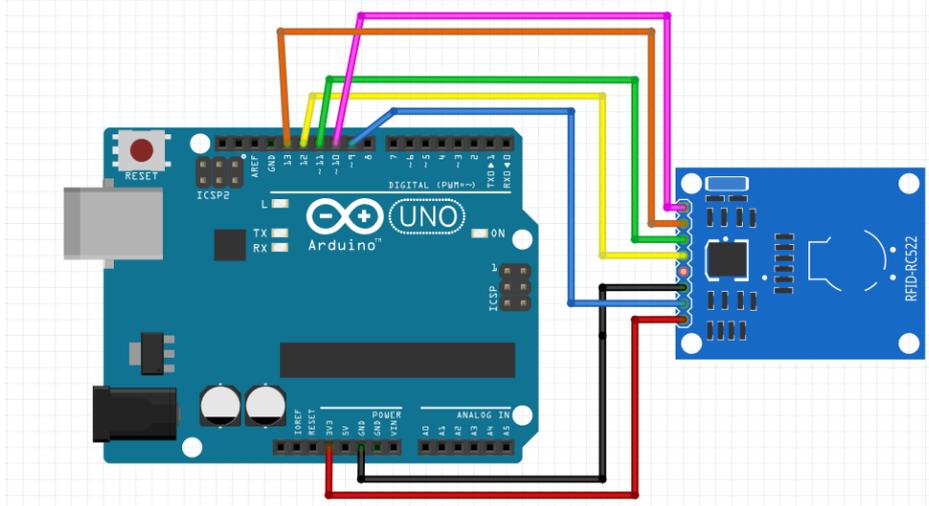
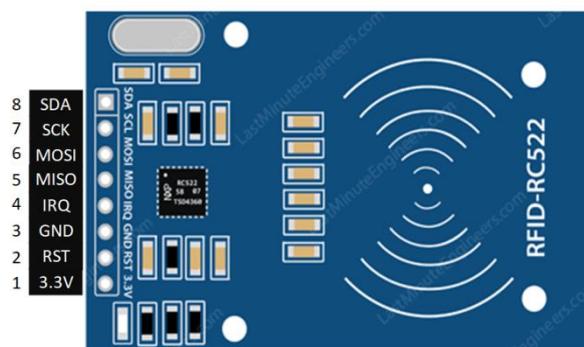


L'RC522 è progettato per creare un campo elettromagnetico a 13,56 MHz con il quale comunica con i tag RFID e supporta la comunicazione sui protocolli SPI, I2C e UART. Nel nostro caso, comunica con Arduino tramite SPI (interfaccia periferica seriale) a 4 pin e una velocità dati massima di 10 Mb/s.

La tensione di alimentazione operativa del modulo è compresa tra 2,5 a 3,3 V ma gli ingressi logici tollerano anche 5V per cui può essere interfacciato con Arduino senza utilizzare alcun convertitore di livello logico. La distanza di lettura di un tag è di 5 cm.

Piedinatura RFID RC522 e collegamenti con Arduino Uno

RC522	Arduino Uno	Colore
3.3V	3.3V	Red
RST	D9	Blue
GND	GND	Black
IRQ	Non connesso	
MISO/ SCL / TX	D12	Yellow
MOSI	D11	Green
SCK	D13	Orange
SDA/SS/RX	D10	Pink

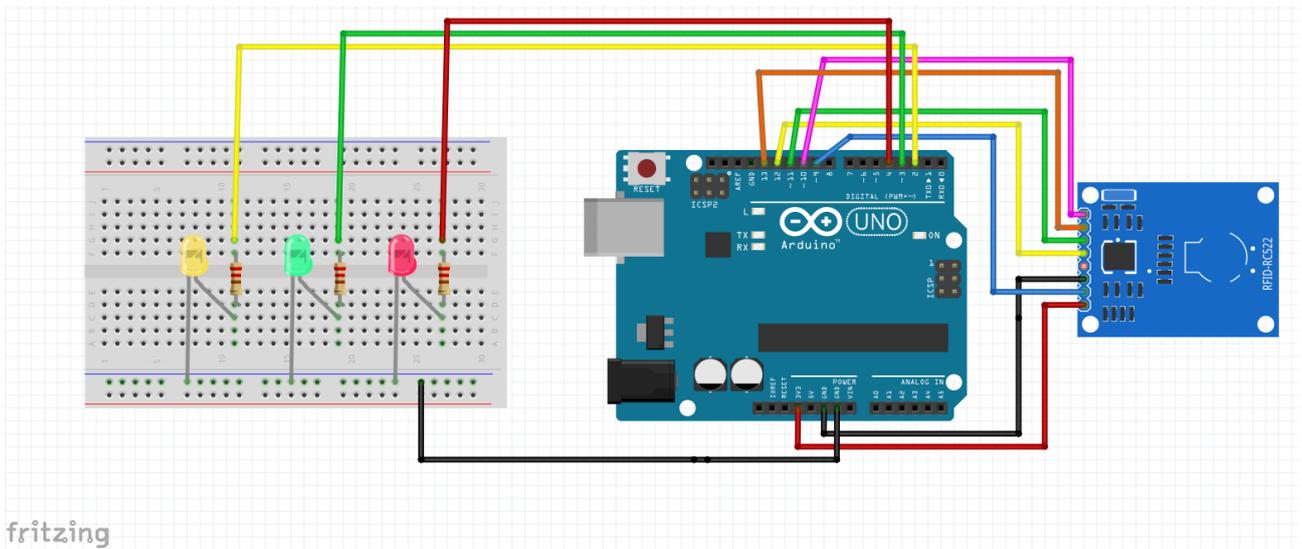


Evitiamo per semplicità di spiegare il significato e la funzione dei quattro pin MISO, MOSI, SCK e SDA utilizzati per la comunicazione SPI tra RC522 ed Arduino e prestiamo attenzione al tipo di scheda Arduino utilizzata. Se si utilizza una scheda diversa da Arduino Uno e da Arduino Nano, i pin MISO, MOSI, SCK e SDA potrebbero essere diversi da D12, D11, D13 e D10: in tali casi è opportuno consultare la documentazione ufficiale di Arduino.

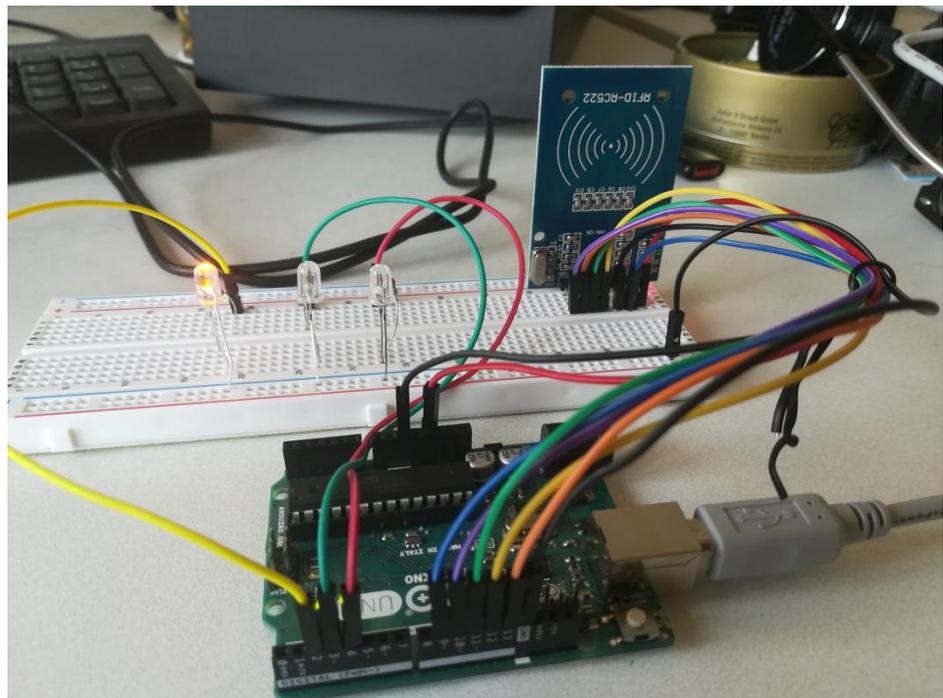
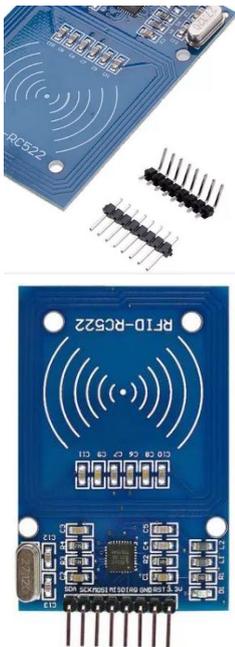
Collegamenti su breadboard

Una volta individuati i collegamenti tra Arduino e RC522, inseriamo sulla breadboard tre diodi led, ciascuno con una resistenza di limitazione di 220 Ohm:

1. il diodo led **giallo**, connesso al pin **D2**, indica che il Reader è pronto. E' sempre acceso tranne quando il Reader legge il tag e per 3 secondi si accende il diodo led verde (tag riconosciuto) o il diodo led rosso (tag non riconosciuto)
2. il diodo led **verde**, connesso al pin **D3**, si accende quando il Reader legge un tag e lo riconosce
3. il diodo led **rosso**, connesso al pin **D4**, si accende quando il Reader legge un tag e non lo riconosce



Nella realizzazione del circuito di prova ho provveduto a saldare sul dispositivo RC522 una striscia di pin (generalmente viene fornita dal Fornitore insieme con il dispositivo) e ho realizzato i collegamenti tra Arduino e RC522 utilizzando la breadboard



Componenti utilizzati

- Arduino Uno collegato ad un PC Windows X
- Breadboard con cavetteria
- 1 lettore RFID RC522
- 2 tag portachiavi e 1 scheda RFID
- 1 diodo led giallo
- 1 diodo led verde
- 1 diodo led rosso
- 3 Resistenze da 220 Ohm

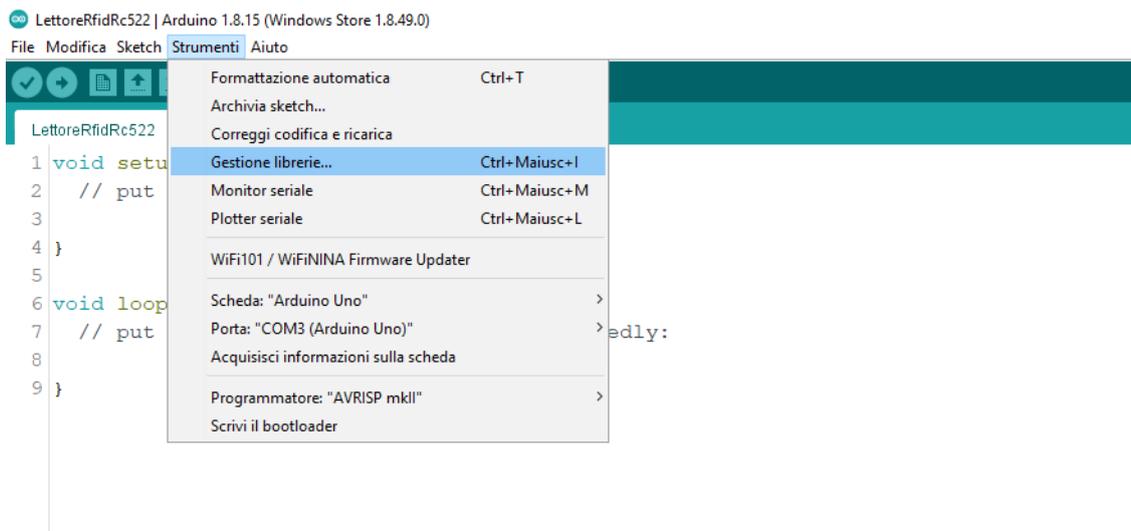
Per quel che riguarda lettore e tag, in genere per pochi euro viene fornito un kit con 3 lettori, 6 strisce di pin, 3 tag portachiavi e 3 schede RFID



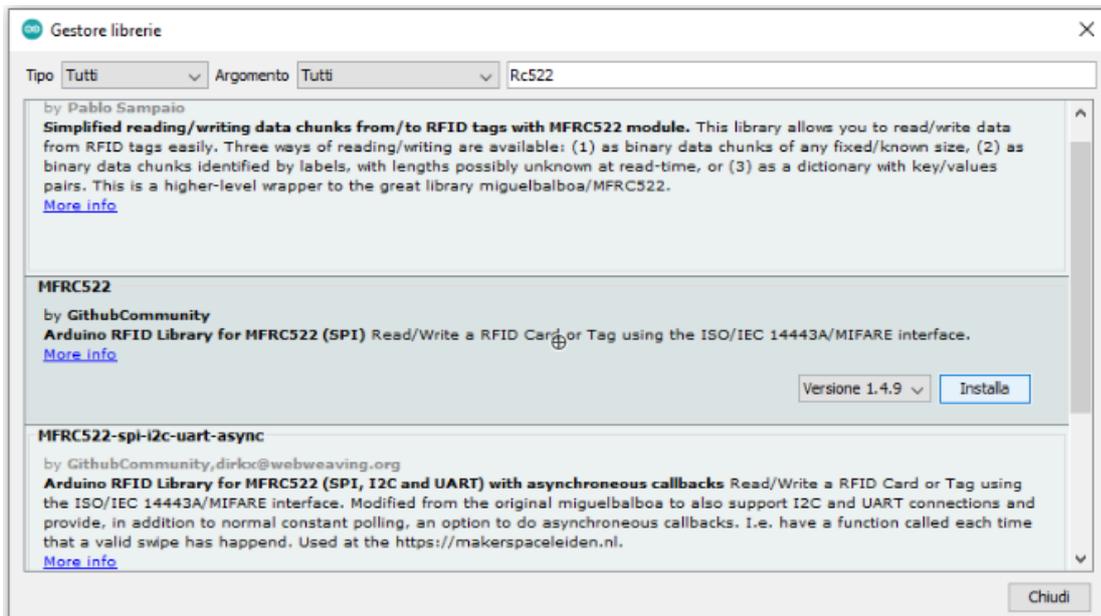
La Libreria MFRC522

Prima di iniziare la programmazione di Arduino, bisogna installare, e quindi includere nello sketch, la libreria necessaria per poter utilizzare il lettore di tag RFID RC522.

- Con l'IDE Arduino in esecuzione, scegliere *Strumenti-Gestione librerie...*



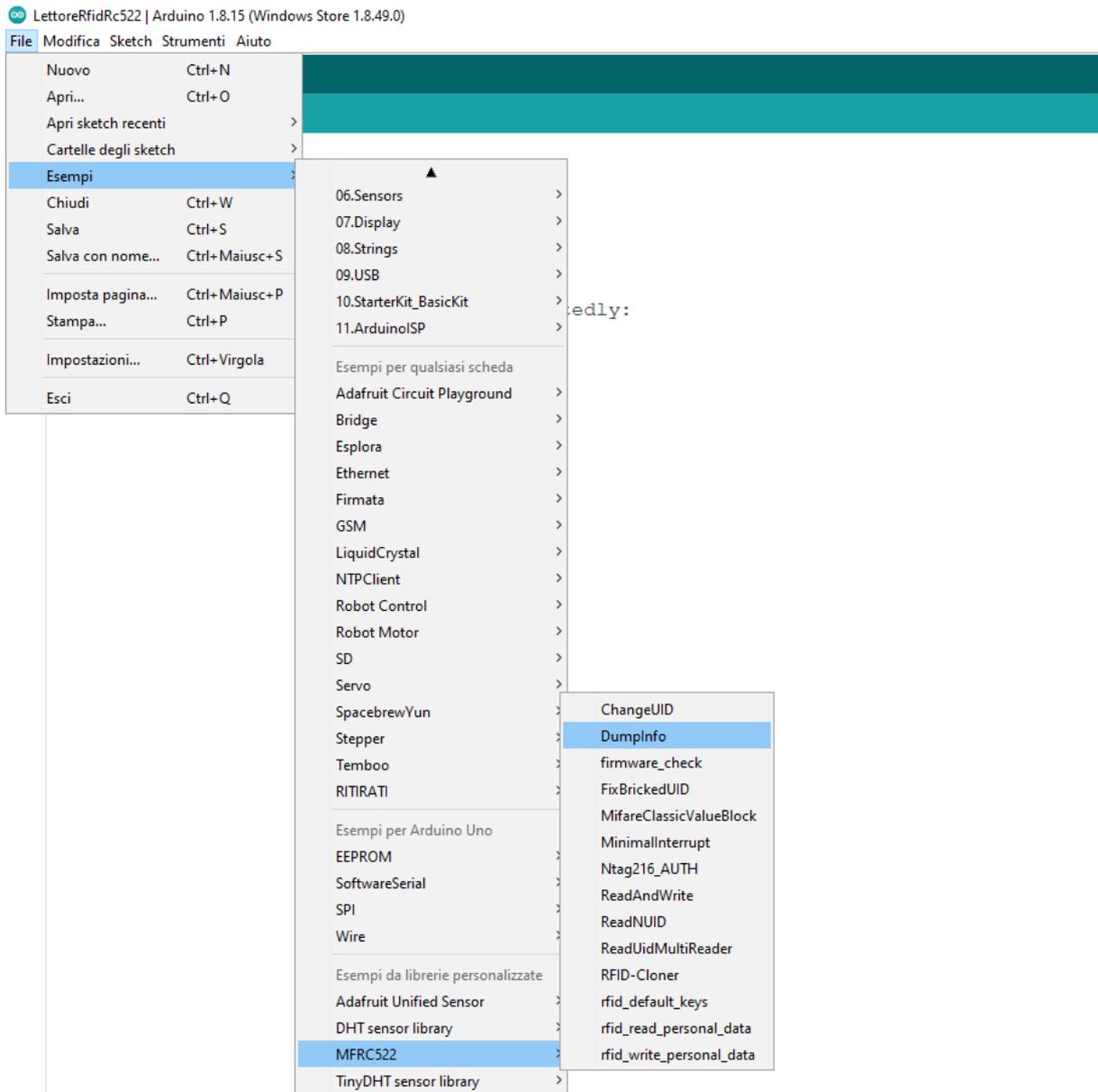
- Quando si apre la finestra del Gestore librerie, nel campo di ricerca inserire "RC522"
- Tra le diverse librerie proposte trovare, selezionare e installare "MFRC522" (by GithubCommunity)



La lettura dei codici identificativi dei tag a nostra disposizione

Una volta installata la libreria MFRC522, utilizziamo uno sketch della libreria che ci permette di leggere il contenuto dei tag a nostra disposizione, in particolare ci permette di conoscere il codice identificativo del tag.

Dall'opzione **File->Esempi** selezioniamo **MFRC522->DumpInfo**



Questo sketch, come suggerisce il suo nome, legge il tag e ci fornisce i dati in esso memorizzati, in particolare il codice identificativo che dobbiamo necessariamente conoscere e utilizzare nel nostro progetto per poter "autorizzare" o meno il tag

Avvicinando il tag al modulo (facendo attenzione a non spostarlo fin quando l'operazione di lettura non è completata), otteniamo sul monitor seriale tutti i dati in esso memorizzati. Facciamo questa operazione per uno dei due tag portachiavi utilizzati e per una scheda RFID, che nel nostro progetto saranno riconosciuti dal lettore RFID (accensione del led verde). L'altro tag portachiavi servirà ad illustrare il caso in cui il lettore non riconosce il tag e non autorizza alcuna azione (accensione del led rosso)

Codice dello sketch DumpInfo

Attenzione: ho modificato lo sketch DumpInfo sostituendo la riga `Serial.begin(9600)` con la riga `Serial.begin(115200)`

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

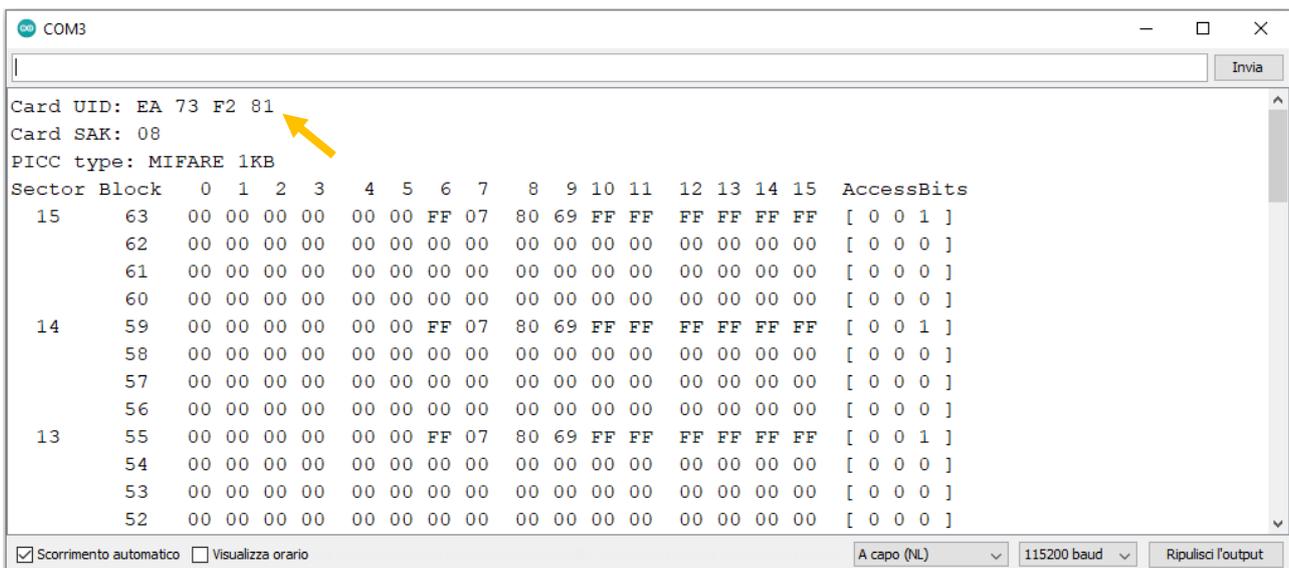
#define RST_PIN 9 // Configurable, see typical pin layout above
#define SS_PIN 10 // Configurable, see typical pin layout above

MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance

void setup() {
  Serial.begin(115200); // Initialize serial communications with the PC Valore di default 9600
  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
  delay(4); // Optional delay. Some board do need more time after init to be ready, see Readme
  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() {
  // Reset the loop if no new card present on the sensor/reader. This saves the entire process when idle.
  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));
}
```

Avviamo lo sketch DumpInfo, apriamo il monitor seriale, impostiamo la velocità a 115200 baud e avviciniamo il portachiavi RFID. Otteniamo il dump dei dati da cui ricaviamo il Card UID, ovvero il codice identificativo esadecimale, **EA73F281** (8 cifre esadecimali, ovvero 4 byte)



```
COM3
Card UID: EA 73 F2 81
Card SAK: 08
PICC type: MIFARE 1KB
Sector Block 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 AccessBits
15 63 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
62 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
61 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
14 59 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
58 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
57 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
56 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
13 55 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
54 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
53 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
52 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
```

Avviciniamo la scheda RFID. Otteniamo il dump dei dati da cui ricaviamo il Card UID, ovvero il codice identificativo esadecimale, **E9930EE3** (8 cifre esadecimali, ovvero 4 byte)

```

COM3
Card UID: E9 93 0E E3
Card SAK: 08
PICC type: MIFARE 1KB
Sector Block 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 AccessBits
15 63 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
62 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
61 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
14 59 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
58 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
57 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
56 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
13 55 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF FF [ 0 0 1 ]
54 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
53 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
52 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [ 0 0 0 ]
    
```

Evitiamo di descrivere in questo progetto il significato e le modalità di interpretazione di tutti i dati che compaiono nel monitor seriale: lo farò, eventualmente, in un progetto successivo in cui illustrerò come si possono scrivere dati personalizzati in un tag.

Mi preme invece di consigliare la buona pratica di consultare sempre gli esempi delle librerie utilizzate con Arduino: si tratta di ottimi programmi che spiegano le principali funzionalità delle librerie e spesso rappresentano un'eccellente base di partenza per le applicazioni che vogliamo realizzare.

Lo sketch di Arduino "LettoreRfidRc522.ino"

```

#include <SPI.h>
#include <MFRC522.h>
#define Led_Giallo 2
#define Led_Verde 3
#define Led_Rosso 4
#define RST_PIN 9 // pin di Arduino connesso con pin RST di RC522
#define SS_PIN 10 // pin di Arduino connesso con pin SS/SDA/RX di RC522

MFRC522 mfrc522(SS_PIN, RST_PIN); // Crea un'istanza della classe MFRC522 che chiamiamo mfrc522

/* Card UID dei due tag utilizzati per la prova e identificati utilizzando lo sketch DumpInfo*/
String tagPortachiavi = "EA73F281";
String tagSchedaRfid = "E9930EE3";
String tagLetto="";
bool lettura=true;

void setup() {
  Serial.begin(115200);
  SPI.begin(); // inizializza SPI bus
  mfrc522.PCD_Init(); // inizializza la libreria e il lettore
  pinMode(Led_Giallo,OUTPUT); //led giallo
  pinMode(Led_Verde,OUTPUT); //led verde
  pinMode(Led_Rosso,OUTPUT); // led rosso
  digitalWrite(Led_Giallo, HIGH); // led giallo acceso indica che il Reader è in attesa di leggere un tag
}
    
```

```

void loop() {

while (getID()) //Rimane in attesa di letture da parte del Reader
{

if (tagLetto == tagPortachiavi || tagLetto == tagSchedaRfid)
{
if (tagLetto == tagPortachiavi)Serial.println("Tag portachiavi riconosciuto");
if (tagLetto == tagSchedaRfid)Serial.println("Scheda RFID riconosciuta");
Serial.print("Card UID letto: ");
Serial.println(tagLetto);
Serial.println();
// inibisce la lettura e spegne il Led_Giallo e accende il Led_Verde per 3 secondi
lettura=false;
digitalWrite(Led_Giallo, LOW);
digitalWrite(Led_Verde, HIGH);
delay(3000);
digitalWrite(Led_Verde, LOW);
lettura=true; //ri-abilita la lettura e ri-accende il Led_Giallo
digitalWrite(Led_Giallo, HIGH);
}
else
{
Serial.println("----- Tag non riconosciuto");
Serial.print("----- Card UID letto: ");
Serial.println(tagLetto);
Serial.println();
// inibisce la lettura e spegne il Led_Giallo e accende il Led_Rosso per 3 secondi e
lettura=false;
digitalWrite(Led_Giallo, LOW);
digitalWrite(Led_Rosso, HIGH);
delay(3000);
digitalWrite(Led_Rosso, LOW);
lettura=true; //ri-abilita la lettura e ri-accende il Led_Giallo
digitalWrite(Led_Giallo, HIGH);
}
}
}

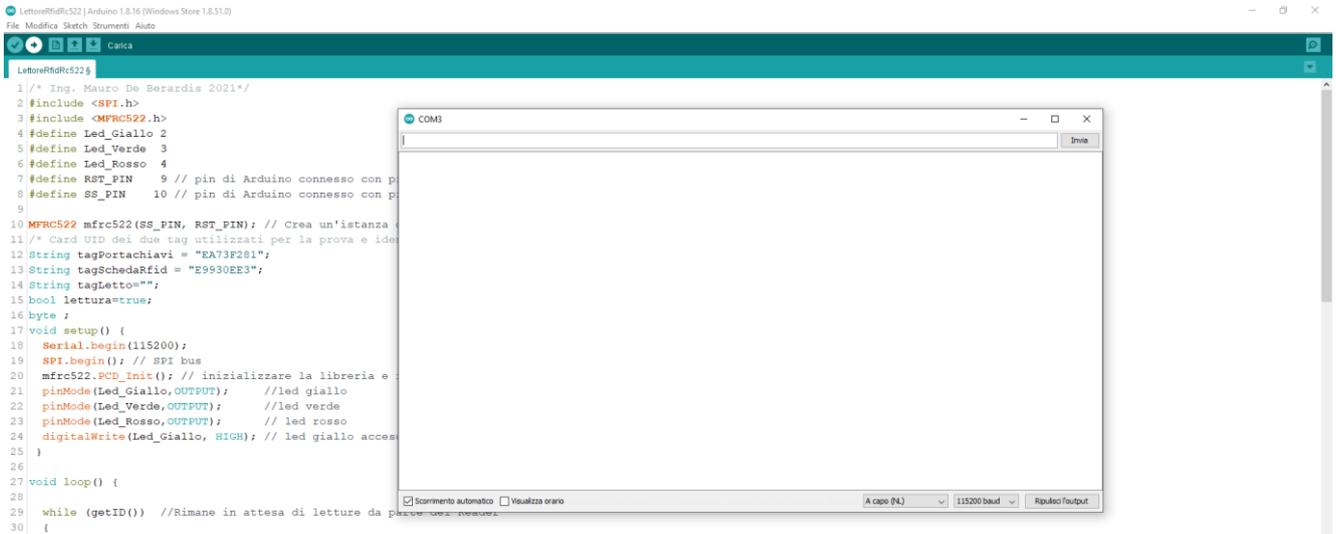
//Lettura
boolean getID()
{
if(!lettura) return false; // il Reader non può fare letture nei 3 secondi di accensione del led verde o del led rosso
if (! mfr522.PICC_IsNewCardPresent()) return false; // verifica se viene rilevato un tag RFID
if ( ! mfr522.PICC_ReadCardSerial()) return false; // verifica se i dati possono essere letti dal tag RFID
// legge i dati del tag
tagLetto = "";
for ( byte i = 0; i < 4; i++) // il Card UID contiene 4 byte ovvero 8 numeri esadecimali(ad esempio E9 93 0E E3)
{
int n=mfr522.uid.uidByte[i]; // ottiene il numero intero n corrispondente a ciascun byte del Card UID
String s=String(n, HEX); // converte n in una stringa s di numeri esadecimali
if(s.length()<2)s="0"+s; // se la stringa s contiene un solo carattere aggiunge lo 0 iniziale
tagLetto.concat(s); // concatena la stringa s al tagLetto;
}
tagLetto.toUpperCase(); // converte la stringa tagLetto in maiuscole
mfr522.PICC_HaltA(); // Termina la lettura
return true;
}
}

```

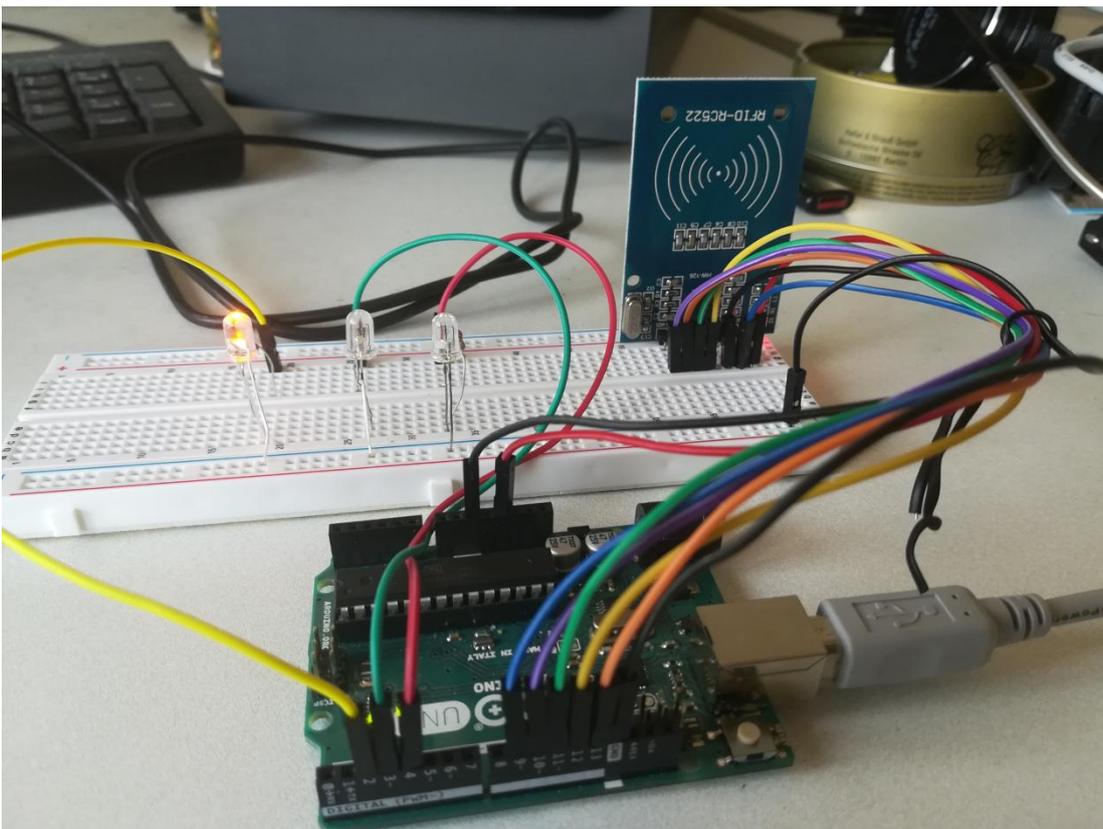
[Scarica LettoreRfidRc522.ino](#)

Prova del progetto

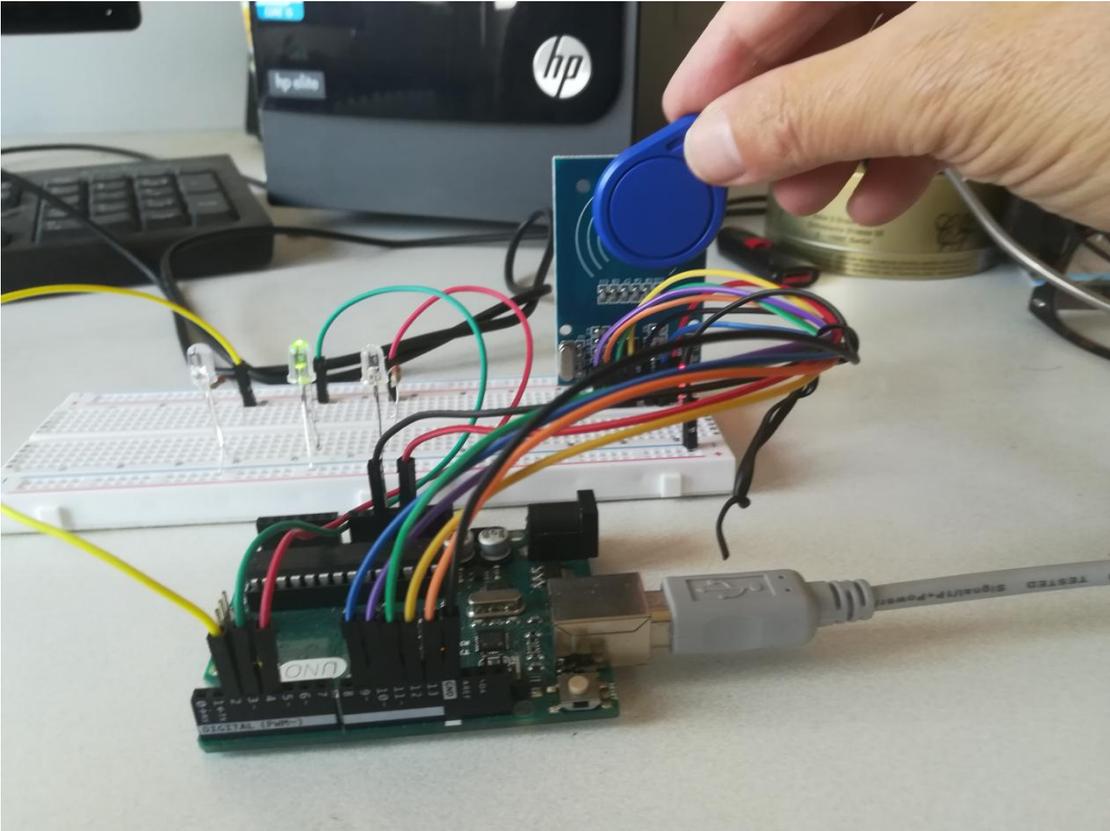
Sull'IDE Arduino apro il monitor seriale, con velocità impostata a 115200 baud



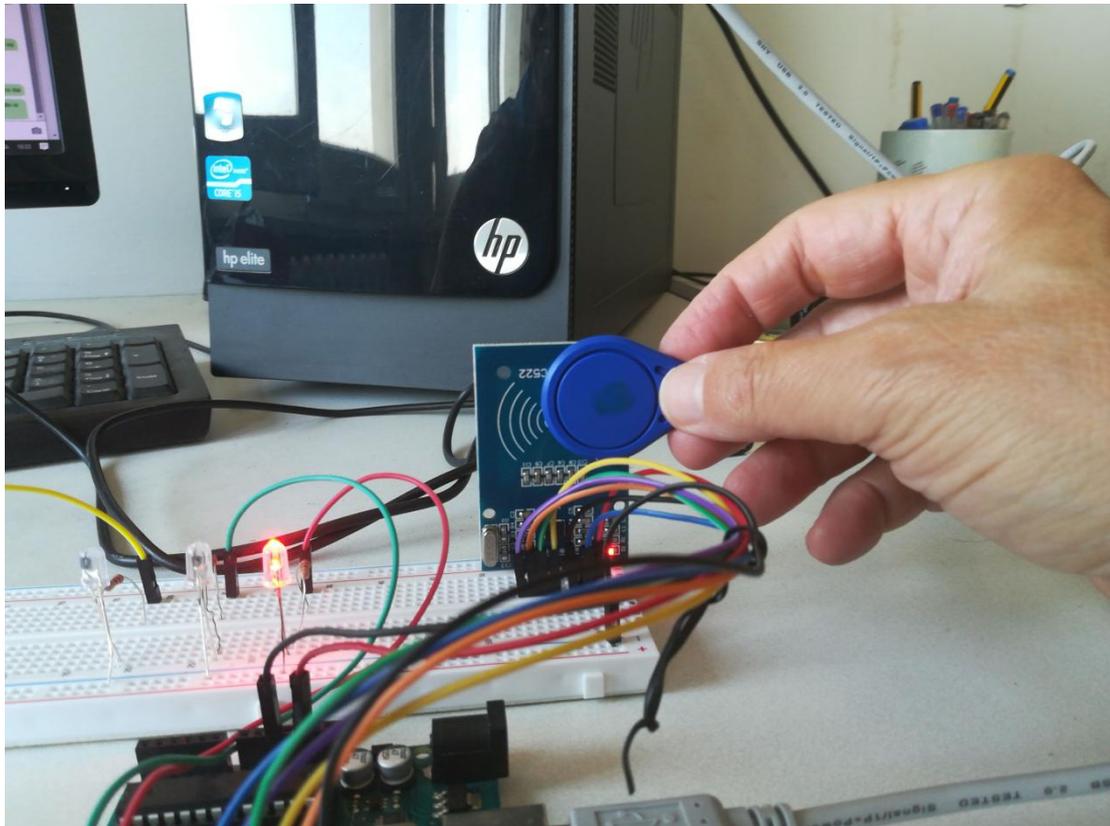
il diodo led **giallo** indica che il Reader RFID RC522 è in attesa di rilevare un tag



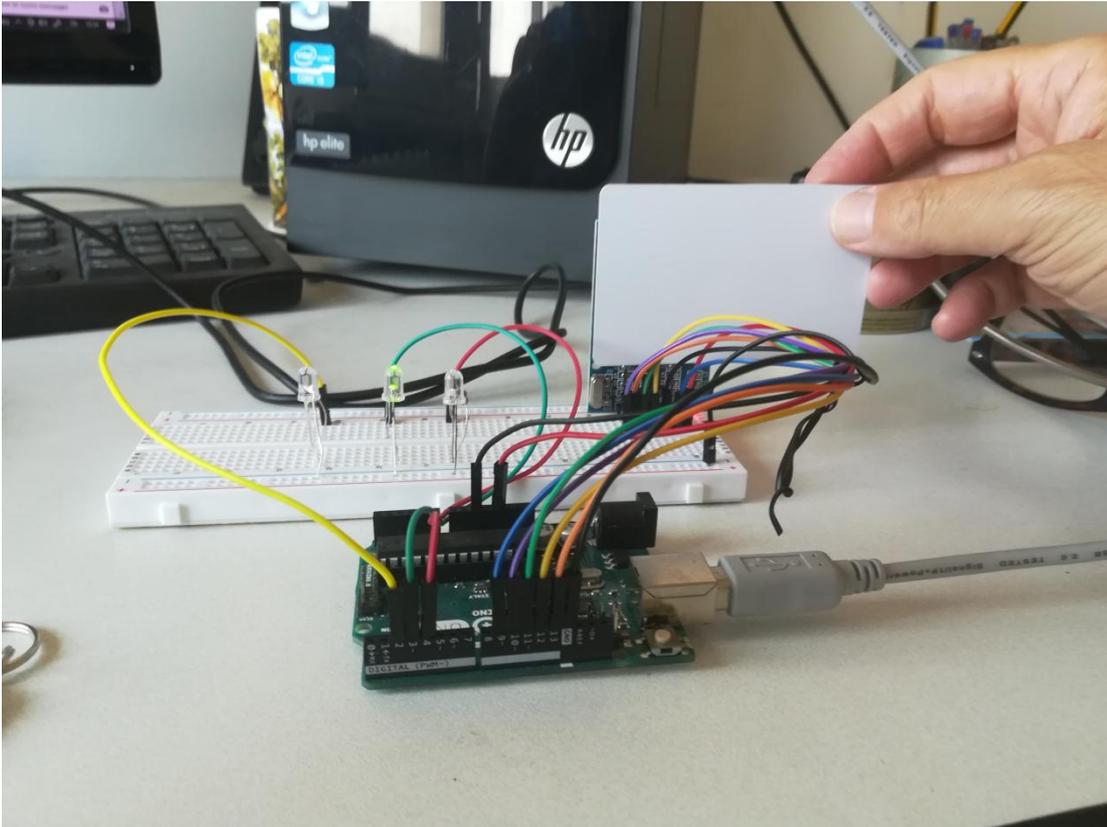
Avviciniamo al Reader il tag portachiavi precedentemente identificato con DumpInfo: il tag viene riconosciuto, il diodo led giallo si spegne e il diodo led **verde** si accende per 3 secondi



Avviciniamo al Reader l'altro tag portachiavi non identificato con DumpInfo: il tag non viene riconosciuto, il diodo led giallo si spegne e il diodo led **rosso** si accende per 3 secondi



Avviciniamo al Reader la scheda RFID precedentemente identificata con DumpInfo: il tag viene riconosciuto, il diodo led giallo si spegne e il diodo led verde si accende per 3 secondi



I risultati delle tre letture sono riportate sul monitor seriale

```
COM3
Tag portachiavi riconosciuto
Card UID letto: EA73F281

----- Tag non riconosciuto
----- Card UID letto: EA72FD80

Scheda RFID riconosciuta
Card UID letto: E9930EE3
```

Scorrimento automatico Visualizza orario A capo (NL) 115200 baud Ripulisci l'output