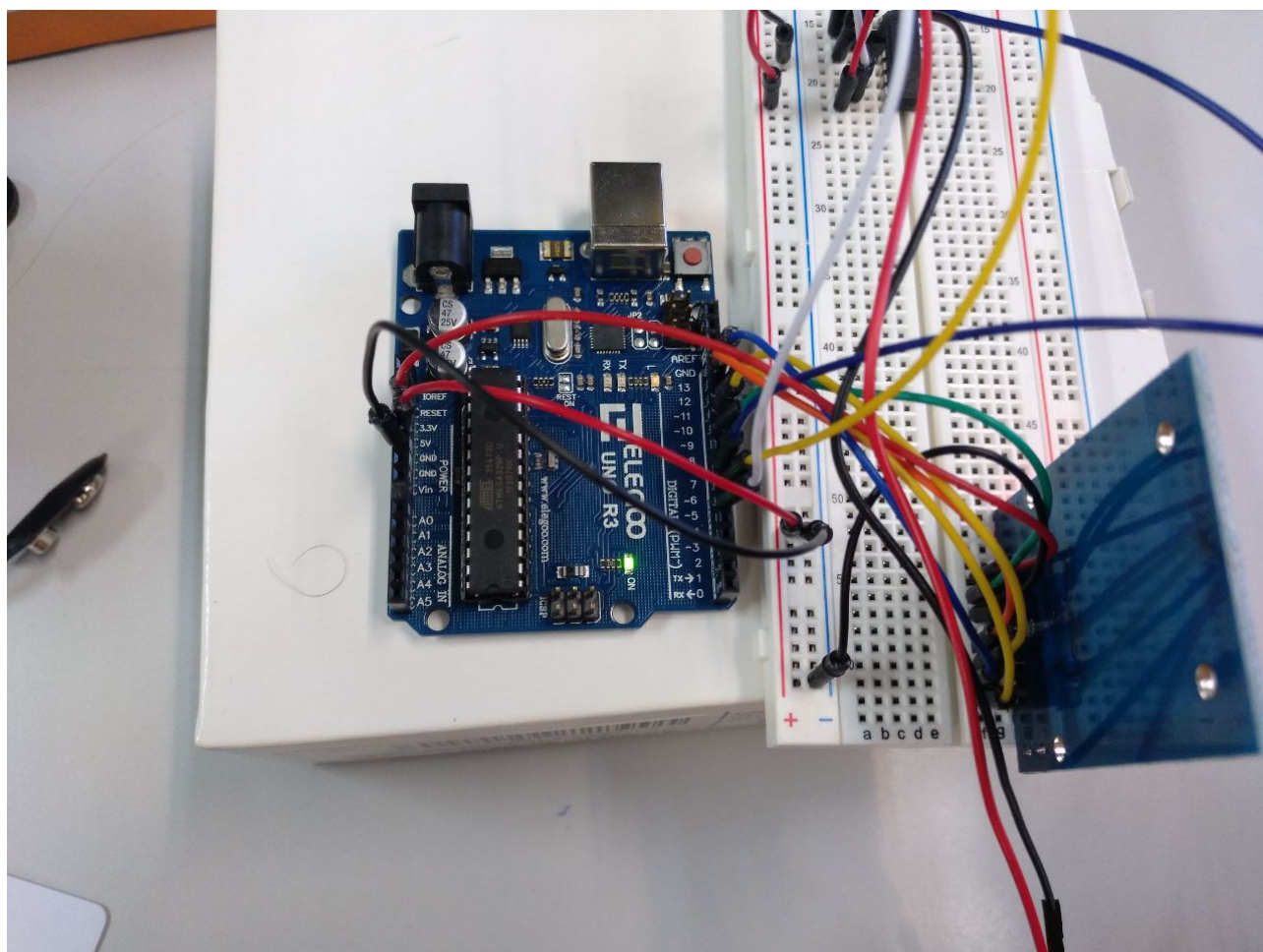
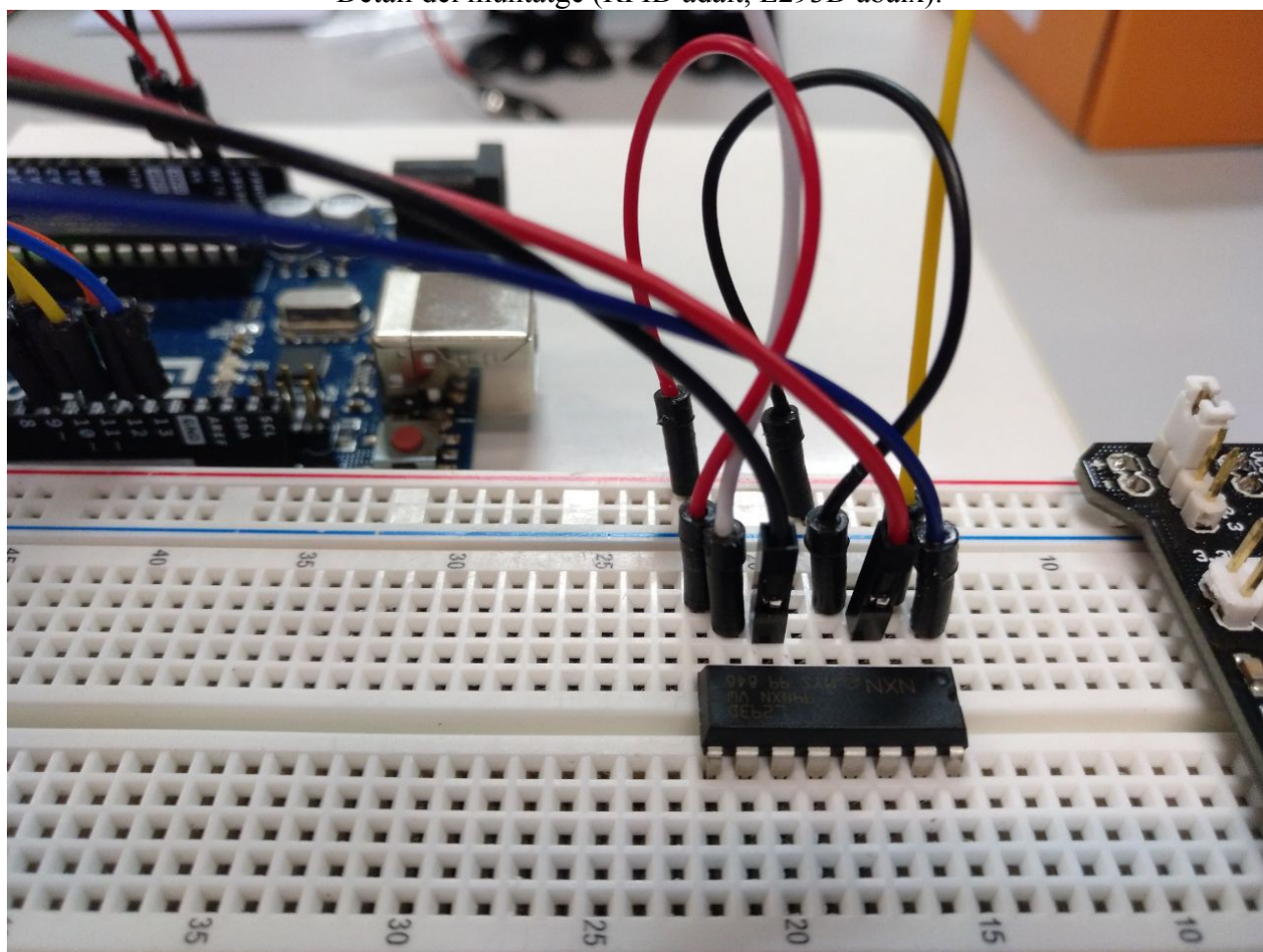


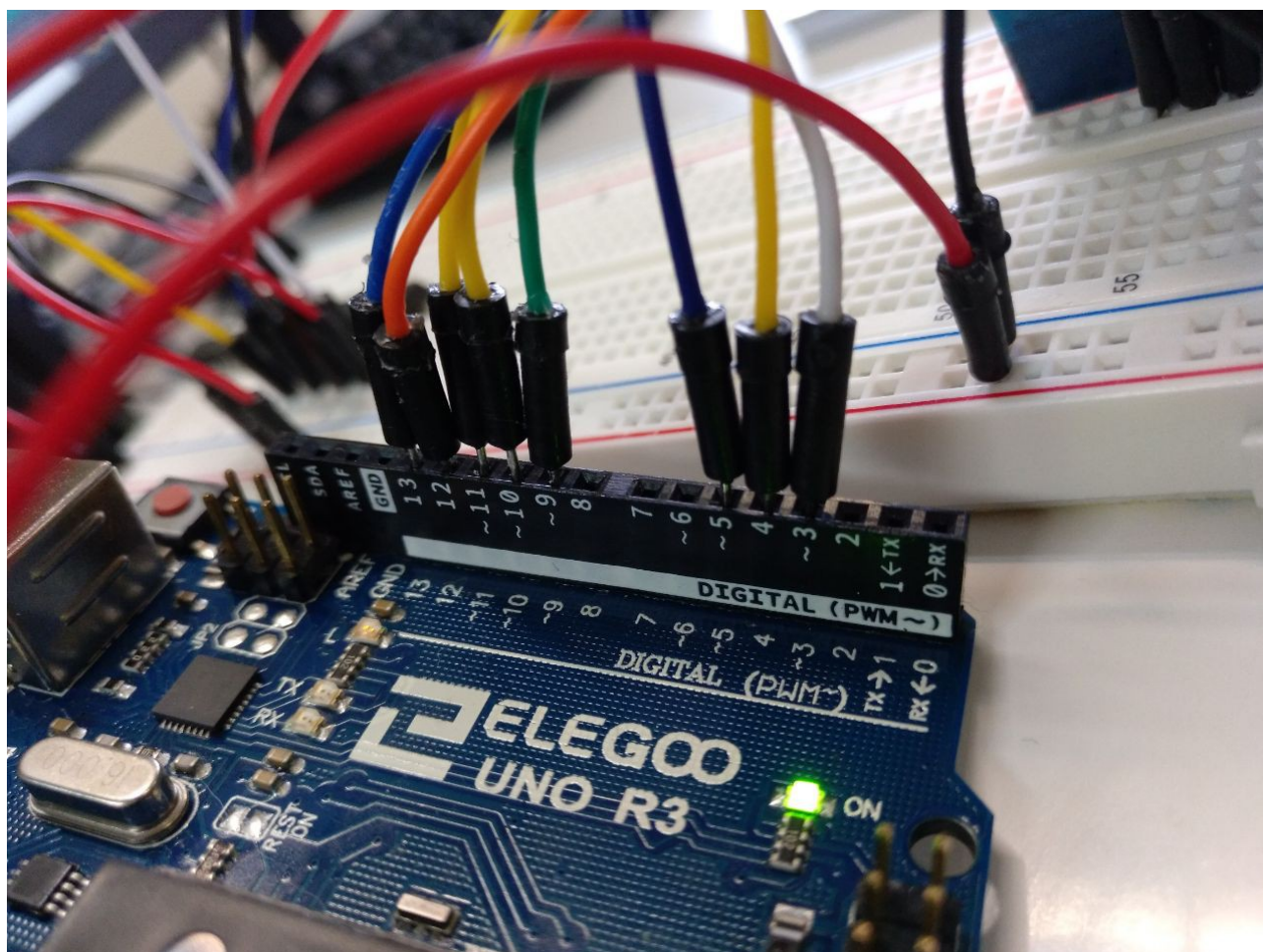
Visió general del muntatge.



Detall del muntatge (RFID adalt, L293D abaix).



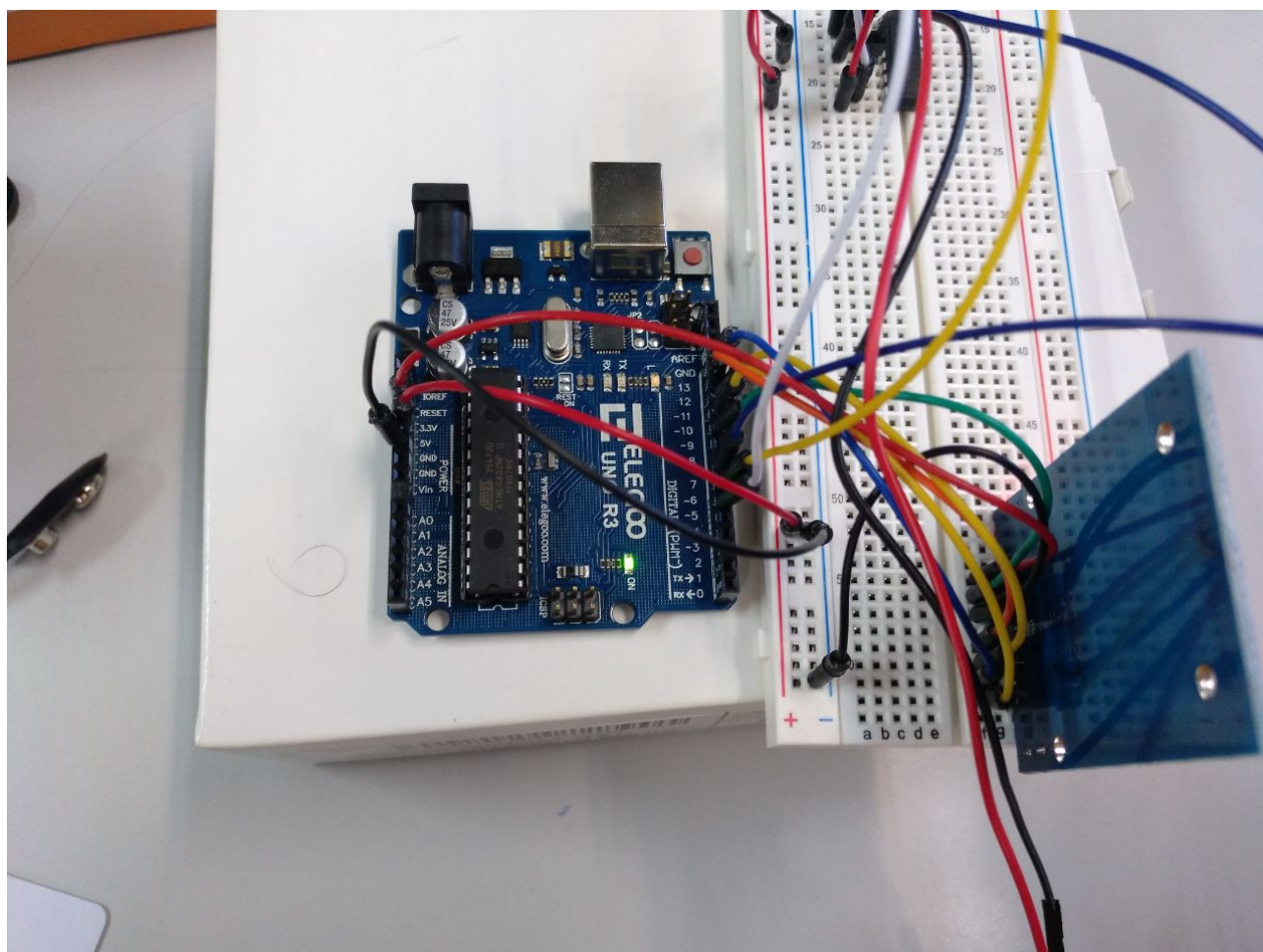




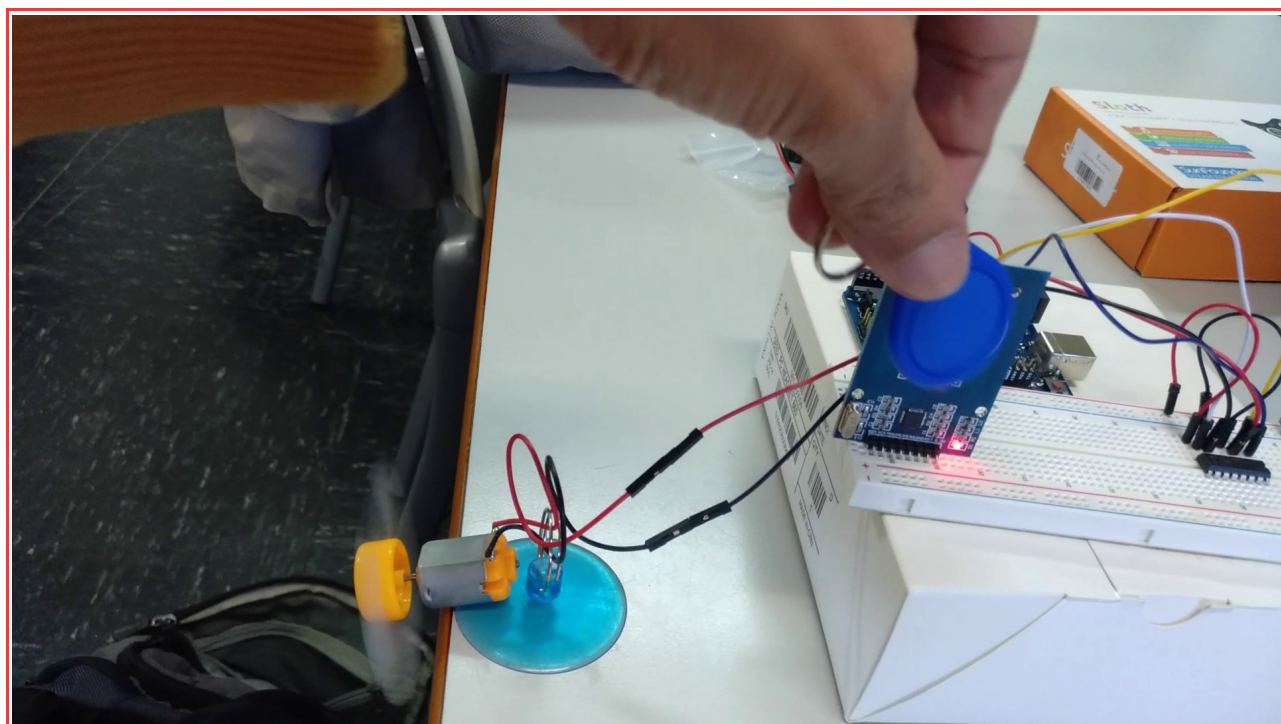
Detall del muntatge (connexió pins adalt, motor abaix)







Detall del muntatge i vídeo funcionament



### Codi usat:

```
#include <SPI.h>
#include <MFRC522.h>
#define RST_PIN 9 //Pin 9 para el reset del RC522
#define SS_PIN 10 //Pin 10 para el SS (SDA) del RC522
MFRC522 mfrc522(SS_PIN, RST_PIN); ///Creem l'objecte per al RC522

//Defineix els pins del motoret
#define ENABLE 7
#define DIRA 6
#define DIRB 5

int estat=0; //estat del motor
int p = 100; //pausa entre lectures per donar temps a treure el
tag
byte ActualUID[4]; //almacenará el código del Tag leído
byte Usuari[4]= {0xE9, 0x03, 0xAD, 0xA3} ; //código de l'usuari

void setup() {
    //Activa els pins del motoret
    pinMode(ENABLE,OUTPUT);
    pinMode(DIRA,OUTPUT);
    pinMode(DIRB,OUTPUT);
    //Inicia el port sèrie per mostrar missatges
    Serial.begin(9600);
    //Iniciem el necessari per l'RFID
    SPI.begin(); //Iniciem el Bus SPI
    mfrc522.PCD_Init(); // Iniciem el MFRC522
    //Mostrem missatge inicial
    Serial.println("Apropi targeta:");
}

void loop() {
    if(llegirTag() and compareArray(mfrc522.uid.uidByte,Usuari)){
        canviafan();
    }
}

//funció per llegir els tags
boolean llegirTag(){
    if ( mfrc522.PICC_IsNewCardPresent()){ // Si detecta una nova
targeta
        if ( mfrc522.PICC_ReadCardSerial()){ // Si llegeix la tarja
            // Mostrem l'UID al port sèrie
            Serial.print(F("Card UID:"));
            for (byte i = 0; i < mfrc522.uid.size; i++) {
                Serial.print(mfrc522.uid.uidByte[i] <0x10 ? " 0" : " ");
                Serial.print(mfrc522.uid.uidByte[i],HEX);
                ActualUID[i]=mfrc522.uid.uidByte[i];
            }
            Serial.print(" ");
        }
    }
}
```

```

        // Terminamos la lectura de la tarjeta tarjeta actual
        mfrc522.PICC_HaltA();
        Serial.println();
        return true;
    }
}
return false;
}

//Funció per comparar dos vectors
boolean compareArray(byte array1[],byte array2[]){
    if(array1[0] != array2[0])return(false);
    if(array1[1] != array2[1])return(false);
    if(array1[2] != array2[2])return(false);
    if(array1[3] != array2[3])return(false);
    return(true);
}

//Funció per activar/desactivar el ventilador
void canviafan(){
    if(estat==0){
        Serial.println("Encén :)");
        digitalWrite(ENABLE,HIGH); // encén
        digitalWrite(DIRA,HIGH); //quina direcció
        digitalWrite(DIRB,LOW);
        estat=1;
        Serial.println("Apropi targeta:");
    } else {
        Serial.println("Apaga :(");
        digitalWrite(ENABLE,LOW); // apaga
        estat=0;
        Serial.println("Apropi targeta:");
    }
}
}

```