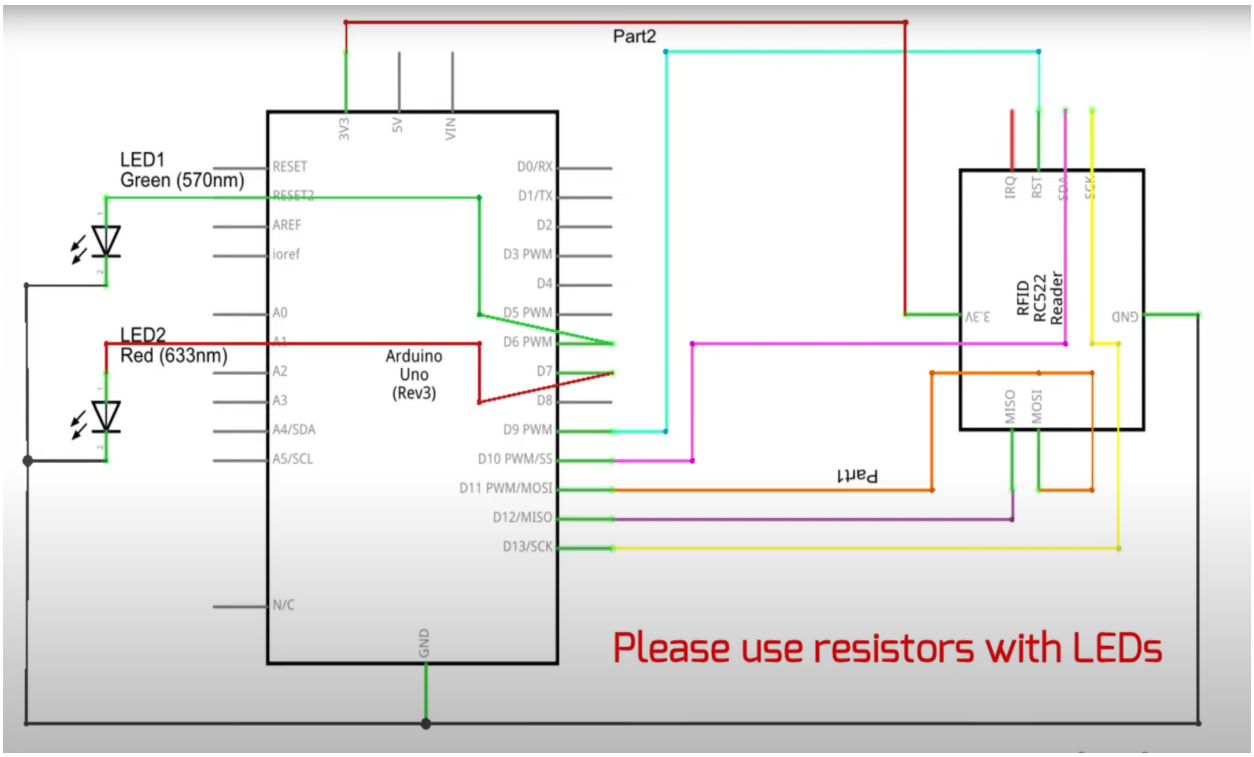
Introduction to Next-Generation Wireless Network - HW2

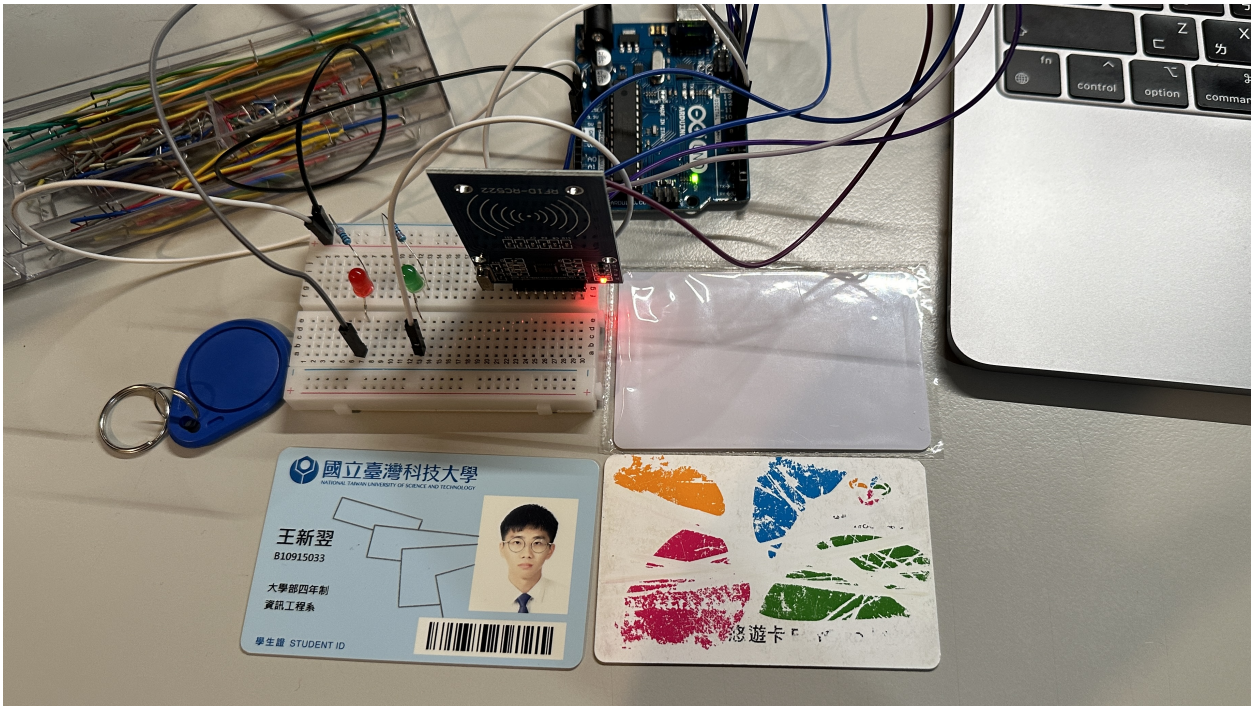
B10915033 王新翌

專案概述

- 透過Arduino RFID模組RC522讀取ID卡，透過Arduino程式管理卡片權限，並透過紅綠LED和Serial Monitor顯示權限和資料。

硬體接線與系統結構圖





- 系統結構圖參照Youtube教學https://www.youtube.com/watch?v=MA3hWp2efZ8
- RC522參照教學將線路接上Arduino
- 紅綠LED正極由D7和D6接出，再接330歐姆電阻接地

- 基本上就按圖施工而已，沒有特別變化

程式碼解釋

重要變數宣告、創建

- 定義卡片數量上限、腳位
- 創建authorizedCards結構，包含卡片id和編號(count)

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

#define SS_PIN 10
#define RST_PIN 9
#define MAX_CARDS 10 // Maximum number of cards: 10

MFRC522 rfid(SS_PIN, RST_PIN); // Instantiate rfid

const int greenLEDPin = 6;
const int redLEDPin = 7;

// Structure to store authorized IDs
struct {
  String ids[MAX_CARDS];
  int count = 0;
} authorizedCards;

bool commandMode = false;
String initialID = "";
```

setup

- 初始設定rfid

- LED腳位設定為OUTPUT，初始為LOW

```
1
2  void setup() {
3    Serial.begin(9600);
4    SPI.begin();
5    rfid.PCD_Init();
6    pinMode(greenLEDPin, OUTPUT);
7    pinMode(redLEDPin, OUTPUT);
8
9    digitalWrite(greenLEDPin, LOW);
10   digitalWrite(redLEDPin, LOW);
11 }
```

readRFIDCard

- 讀取rfid的function
- 將rfid資料轉為HEX後全部轉為大寫並回傳

```
// Read RFID card ID
String readRFIDCard() {
  String rfidID = "";
  for (byte i = 0; i < rfid.uid.size; i++) {
    rfidID += String(rfid.uid.uidByte[i], HEX); // Read RFID ID
  }
  rfidID.toUpperCase();
  return rfidID;
}
```

isAuthorized

- 確認此id是否被授權
- 掃描傳入的id是否存在於目前的id陣列

```
// Check if ID is authorized
bool isAuthorized(String id) {
  for (int i = 0; i < authorizedCards.count; i++) {
    if (authorizedCards.ids[i] == id) {
      return true;
    }
  }
  return false;
}
```

addCard

- 若超過卡片數量上限，則報錯並return
- 若id已存在，回報ID已存在並return
- 否則將卡片加入目前的陣列中，並顯示新增成功資訊及該卡id

```
// Add authorized ID
bool addCard(String id) {
  if (authorizedCards.count >= MAX_CARDS) {
    Serial.println("Error: Authorization list is full");
    return;
  }
  for (int i = 0; i < authorizedCards.count; i++) {
    if (authorizedCards.ids[i] == id) {
      Serial.println("Id already exists: " + id);
      return;
    }
  }
  authorizedCards.ids[authorizedCards.count++] = id;
  Serial.println("ID added successfully: " + id);
  return true;
}
```

removeCard

- 確認要刪除的卡片id是否在範圍內，若超出範圍則報錯
- 將該卡往後的所有卡片往前移一位，刪除該卡資料
- 顯示刪除成功資訊和該卡id

```
// Remove authorized ID
bool removeCard(int index) {
  String id = "";
  if (index < 0 || index >= authorizedCards.count) {
    Serial.println("Error: Invalid index number");
    return false;
  }
  id = authorizedCards.ids[index - 1];
  for (int i = index; i < authorizedCards.count - 1; i++) {
    authorizedCards.ids[i] = authorizedCards.ids[i + 1];
  }
  authorizedCards.count--;
  Serial.println("ID removed successfully: " + id);
  return true;
}

```

listCards

- 列出所有已儲存的卡片id和當前卡片總數

```
// Display list of authorized IDs
void listCards() {
  Serial.println("Authorized ID List:");
  Serial.print("Total count: ");
  Serial.println(authorizedCards.count);
  for (int i = 0; i < authorizedCards.count; i++) {
    Serial.print(i + 1);
    Serial.print(": ");
    Serial.println(authorizedCards.ids[i]);
  }
  Serial.println("--------------------");
}
```

processCommand

- 將輸入的指令透過trim進行修剪
- ADD
  - 讀到卡後呼叫readRFIDCard讀卡
  - 透過addCard儲存資料
  - 將commandMode設為false
  - delay(1000)
    - 避免新的卡片新增完後馬上再觸發一次isAuthorized
- LIST
  - 呼叫listCards顯示當前卡片資訊
  - 將commandMode設為false
- REMOVE

- index為第七位（英文字母佔六位），轉為數字後-1
- 呼叫removeCard並傳入index
- 將commandMode設為false

```
void processCommand(String command) {
  command.trim();

  if (command.startsWith("ADD")) {
    Serial.println("Please scan the card to add...");
    while(!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial()) {
      delay(100);
    }
    String newID = readRFIDCard();
    addCard(newID);
    commandMode = false;
    delay(1000);
  }
  else if (command == "LIST") {
    listCards();
    commandMode = false;
  }
  else if (command.startsWith("REMOVE")) {
    int index = command.substring(7).toInt() - 1;
    removeCard(index);
    commandMode = false;
  }
}

```

loop

- 讀取卡片
  - 首次讀取創建initialID
    - Serial Monitor顯示初始ID授權成功
    - 顯示初始卡片id，並加入清單中
    - 綠燈亮起1秒

    - 若該讀取卡片已被授權
        - 綠燈亮起1秒
        - 將commandMode設為true
        - Serial Monitor提示可輸入指令
    - 若該卡未被授權
        - 紅燈亮起1秒
        - Serial Monitor顯示該卡為授權
- 處理指令
    - 讀取指令直到換行
    - 呼叫processCommand並傳入指令

```
void loop() {
  if (rfid.PICC_IsNewCardPresent() && rfid.PICC_ReadCardSerial()) {
    String cardID = readRFIDCard();
    Serial.println("ID Read: " + cardID);
    if (initialID == "") {
      initialID = cardID;
      Serial.println("Initial authorized ID set successfully");
      addCard(initialID);
      digitalWrite(greenLEDPin, HIGH);
      delay(1000);
      digitalWrite(greenLEDPin, LOW);
    }
    // Check authorization
    else if (isAuthorized(cardID)) {
      digitalWrite(greenLEDPin, HIGH);
      delay(1000);
      digitalWrite(greenLEDPin, LOW);
      commandMode = true;
      Serial.println("Entering command mode");
      Serial.println("Available commands: ADD, LIST, REMOVE [number]");
    }
    else {
      digitalWrite(redLEDPin, HIGH);
      delay(1000);
      digitalWrite(redLEDPin, LOW);
      Serial.println("Unauthorized ID");
    }
    rfid.PICC_HaltA();
  }

  // Process serial commands
  if (commandMode && Serial.available()) {
    String command = Serial.readStringUntil('\n');
    processCommand(command);
  }
}

```

實作過程與挑戰

- 很幸運的是，網路上已有RC522的接線教學，線路很簡單

- 每次在下ADD指令新增卡片時，都會因為檢測過快，導致卡片被新增後馬上又跳出該卡的操作指令提示
    - 正確應該是1. 新增卡片 2. 重新刷卡再顯示操作指令
    - 解決方式為在processCommand下的ADD中的最後一部分增加delay(1000)，避免系統短時間內重複感應卡片