

微算機原理與應用 — 期末專題

主題：電子鎖系統

組員

F74116291 歐冠亭

F74114037 江曉明

F74116275 陳柏淮

F74111178 沈亞稼

壹、系統功能與原理說明

專題動機：

結合感應式 ID 卡與門禁系統，不僅能有效提升用戶的便利性，減少攜帶傳統鑰匙的需求，還可以透過記錄進出資訊，達到人員管理與追蹤的目的，進一步增強安全性與效率。

系統功能與使用說明：

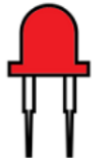
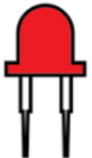
使用 RFID-RDM6300 的電子鎖系統，此系統總有三個狀態，分別為註冊狀態、註銷狀態、檢查狀態。

不同狀態會用 LED 燈泡表示，且狀態間可以使用可變電阻與按鈕進行切換。

以下將針對不同狀態做細緻說明

1. 註冊狀態

LED 燈跑呈現如下

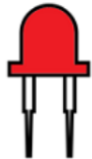
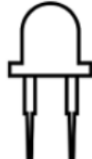
LED1	LED2
 LED on	 LED on

在此狀態下感應卡片，會將卡片資訊儲存於系統

資料庫內。

2. 註銷狀態

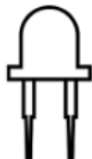

LED 燈跑呈現如下

LED1	LED2
 LED on	 LED off

在此狀態下感應卡片，會將卡片資訊從系統資料庫中移除。

3. 檢查狀態

LED 燈跑呈現如下

LED1	LED2
 LED off	 LED on

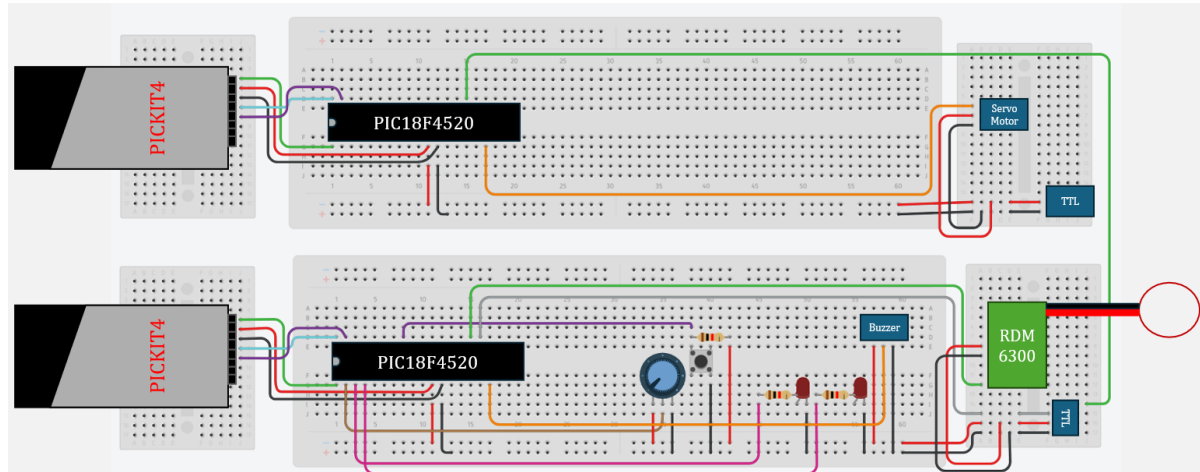
在此狀態下感應卡片，會檢查卡片資料是否存在於系統資料庫內；若存在，則轉動伺服馬達（模擬開鎖動作）。

以上狀態在感應卡片後，均有數種可能結果，且將結果訊息透過 UART 傳遞給電腦，詳情請至 [GitHub](#)

Repository 查閱。

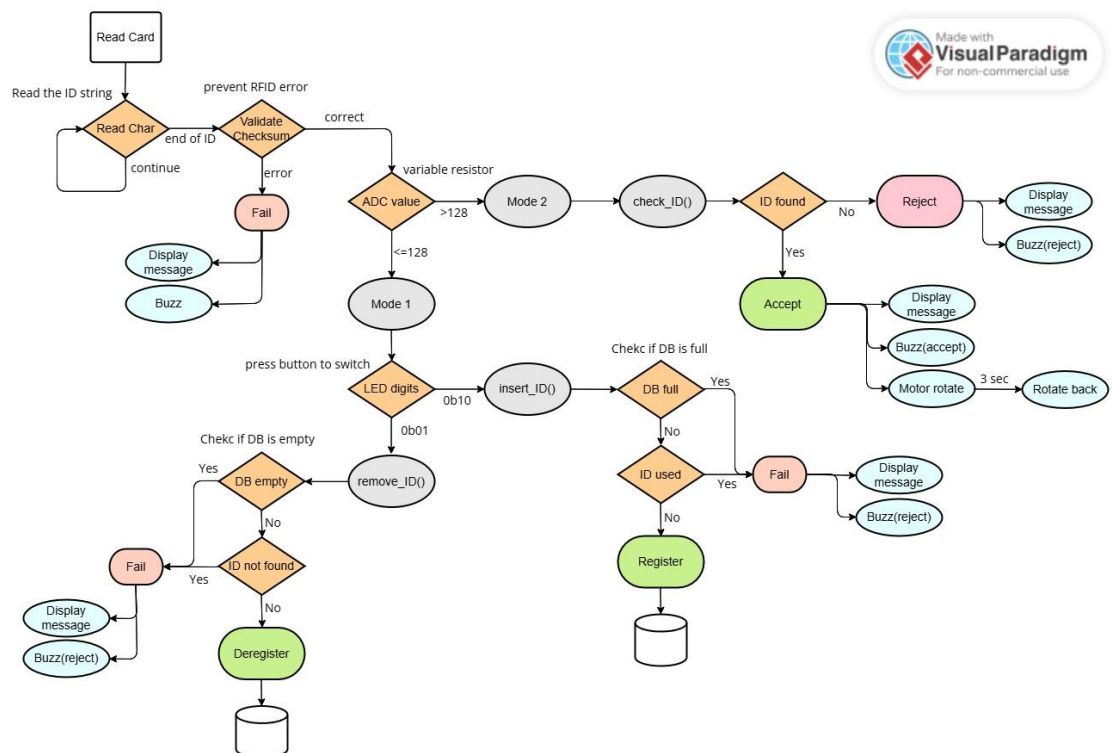
貳、系統完整架構圖、流程圖、電路圖、設計

電路接線示意圖



註：若接線圖不夠清楚，可以參考 GitHub Repository 中的 Components 區塊。

程式流程圖



參、系統開發工具、材料及技術

使用材料

- PIC18F4520 x 2

由於蜂鳴器與伺服馬達均須要使用到 PWM 模式，
故使用到 2 個 PIC18F4520 晶片。

- RFID-RDM6300 讀卡器模組與 125kHz 感應卡

使用到 UART 與 PIC18F4520 做溝通，其中的
RDM6300 只能感應 125kHz 感應卡且 Baud Rate
為 9600。

- 按鈕

使用 INT0 Interrupt，用以在〔註冊〕跟〔註
銷〕模式切換。

- LED x 2

用於表示目前狀態。

- 蜂鳴器

使用到 PWM 模式，透過調整 PR2 和 CCPR1L 控制
音頻。

- 可變電阻

使用 Busy Waiting 偵測可變電阻的變化，用以

在〔註冊〕跟〔檢查〕模式切換。

- **TTL 線**

使用 UART 將 PIC18F4520 的訊息傳遞至電腦，並顯示給使用者看。

- **SG90 伺服馬達**

使用到 PWM 模式，用於模擬開鎖動作。

肆、 周邊接口或 Library 及 API 使用說明

由於並沒有用到比較特別的 Library 與 API，故此處用來說明 RDM6300、PIC18F4520 與電腦傳遞訊息的細節。

當 RDM6300 感應卡片時，會傳送一串 ASCII 字元，從 0x02 代表開始，0x03 代表結束。在 0x02~0x03 之間，會傳遞 12 個十六進位數字的 ASCII 字元，其後 2 個十六進位數字的 ASCII 字元是用於做錯誤檢測(使用 Checksum)，故實際用於表示該感應卡的資訊為前 10 個 ASCII 字元。

舉例來說：假設輸入為 170025882C96，則前十位數 170025882C 的部分為感應卡資訊，而 96 則為核

對和（Checksum）的部分；下圖為詳細的步驟說明



程式碼撰寫方面，第一個晶片會不斷檢查由讀卡器傳入的 ASCII 碼，若不為 0x03，則將字元放入輸入陣列中；若為 0x03，則代表該感應卡讀取完畢，接著即可做錯誤檢查，並進一步判斷。

在第一個晶片執行完後，將訊息結果傳遞給電腦，電腦顯示訊息結果，並判斷訊息是否為檢查成功；若是，則再傳遞訊號給第二個晶片，以旋轉伺服馬達。

伍、系統使用環境及對象

- 使用環境：適用於辦公室入口、宿舍門禁管理、房間門鎖、個人儲物櫃等需要提升安全性與便利性的場所。
- 使用對象：特定場所的授權人員，例如：持有有效學生證的學生、經授權的職員或住戶。

陸、遇到的困難及如何解決

1. 材料購買錯誤

最初選購讀卡器時，並沒有認真看元件規格，便買了較常見的讀卡機（RFID-RC522），後來發現不太適合用於 PIC18F4520 晶片上。故最終改採用 RFID-RDM6300 讀卡器。

2. 蜂鳴器與伺服馬達使用衝突

由於蜂鳴器與伺服馬達均需要使用到 PWM 模式，但只有 TMR2 能夠用於 PWM 模式。因此，起初將伺服馬達排除在外，但後來想到能夠使用兩個 PIC18F4520，一個的 TMR2 用於蜂鳴器，另一個用於伺服馬達，這才能同時使用兩個元件。

3. PIC18F4520 之間的通訊

我們總共測試了三種方法，如下圖所示

方法一

裝置	晶片(1)	晶片(2)	電腦(TTL線)
對接腳位	RC6/TX	RC7/RX	
		RC6/TX	白線
說明	此方法使用UART通訊，晶片(2)作為中繼站的角色，將訊號在原封不動地傳遞給電腦；與此同時，檢查傳來的訊號是否為檢查成功。		
問題	即便晶片(1)跟晶片(2)的Baud Rate相同，但仍舊無法將訊息完整保留，有時訊息傳入晶片(2)後，會變成亂碼，故不使用此法。		

方法二

裝置	晶片(1)	晶片(2)	電腦(TTL線)
對接腳位	RB1/INT1	RB0/INT0	
	RC6/TX		白線
說明	此方法在晶片(1)就判斷是否要旋轉馬達，若要旋轉馬達，則令RB1短暫變為1，之後再變回0。當晶片(2)判斷到該PIN為1時，便旋轉馬達。		
問題	即便晶片(1)有確實改變腳位輸出值，但在晶片(2)卻仍舊檢測不到（相關腳位數值未變），用了很久仍然無法解決，故不使用此法。		

方法三

裝置	晶片(1)	晶片(2)	電腦(TTL線)
對接腳位	RC6/TX		白線
		RC7/RX	綠線
說明	此方法為晶片(1)先傳入電腦顯示，在電腦中判斷是否需要旋轉馬達，若需要，則傳遞訊息給晶片(2)。		
問題	如同方法一，傳入晶片(2)仍舊是亂碼，或者不顯示。		
解決方法	由於晶片(2)動作很單純，只要轉動馬達即可，故根本不必在意是否為亂碼，只要確認有正常的東西(非0)傳進來即可。		

每種方法均有各自的問題，儘管不算完美解決，但方法三勉強可以用來達成我們的功能，故最終被我們採用。

柒、 實際組員之分工項目

● 歐冠亭：

1. 大綱提案撰寫(100%)
2. 系統文件撰寫(30%)
3. 按鈕、可變電阻、LED、UART 程式碼撰寫

● 江曉明：

1. 系統文件撰寫(40%)
2. 晶片(一)主程式、讀卡器系統程式碼撰寫
3. 除蟲、疑難雜症解決

- 陳柏淮：

1. 材料購買
2. 系統文件撰寫(15%)
3. 晶片(二)程式碼、蜂鳴器程式碼撰寫

- 沈亞稼：

1. 系統文件撰寫(15%)
2. 晶片(二)程式碼、伺服馬達程式碼撰寫

捌、 其他

GitHub Repository:

<https://github.com/MingMinNa/RFID-Electronic-Lock>