自傳

2024/7月從國立東華大學資訊工程學系畢業，大一時第一個程式語言就是c 語言，大二學習第二個程式語言是c++物件導向，使用c/c++寫資料結構加深了我對指標和陣列的使用方式，畢業後想繼續寫c/c++往韌體工程師深造

## 專長

sql/restful api/jwt

c/c++寫資料結構(double link list/stack/queue/二元樹...)

自己買stm32開發版透過youtube課程[keys king](https://www.youtube.com/@keysking4403)自學MCU韌體，主要是使用HAL庫進行操作，包括學習GPIO的五大模式(推挽輸出/開漏輸出/浮空輸入/上拉輸入/下拉輸入)，中斷和中斷優先級和callback回調函數的使用，按鈕的使用，串口接收發送消息，I2C控制AHT20量測溫溼度，模塊化為每個外設添加.h和.c檔案然後由main.c調用.c的函數，計時器的使用(設置晶振和預分頻器和重裝載器

## 專案成就

STM32 時鐘日曆系統 with Rotary Encoder (C語言)

開發平台：STM32、HAL 驅動庫

開發語言：C

顯示模組：OLED（使用自定義字型與圖形介面）

使用者介面：旋鈕編碼器（支援方向旋轉與按壓）

專案期間：2025/05 – 2025/05

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

專案簡介：

使用 STM32 做一個時鐘日曆系統，透過旋鈕進行時間設定，結合 OLED 螢幕顯示年/月/日、時/分/秒與星期。

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

旋鈕編碼器驅動（kk\_knob）模組化封裝：

利用 TIM1 的 Encoder 模式偵測旋鈕旋轉方向與按鍵偵測（含消抖機制）。

自定義callback有旋鈕順時針callback旋鈕逆時針callback按下旋鈕callback，觸發callback就透過串口發送告知旋鈕是做了順時針還是逆時針還是按下按鈕

時間顯示與設定邏輯（task\_main）：

使用者可透過按下旋鈕個別設定「年/月/日/時/分/秒」。

設定完成後即寫入 RTC 模組，回復為即時顯示模式。

設計閃爍光標動畫標示目前正在設定的時間項目

模組化架構：

kk\_knob.h/.c：獨立封裝旋鈕邏輯，易於重用。

task\_main.h/.c：主任務控制流程、畫面更新與使用者互動。

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

設計與實作 Rotary Encoder 驅動模組。

撰寫 OLED 顯示與畫面更新邏輯（含字串排版、閃爍光標）。

撰寫主任務流程，包括模式切換邏輯、RTC 時間同步與使用者操作流程。

使用 UART 傳輸除錯資訊以驗證旋鈕操作是否正確呼叫對應回調函數。

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

熟悉 STM32 HAL TIM 編碼器模式與 GPIO 中斷處理。

學會如何設計 callback-based 模組接口。