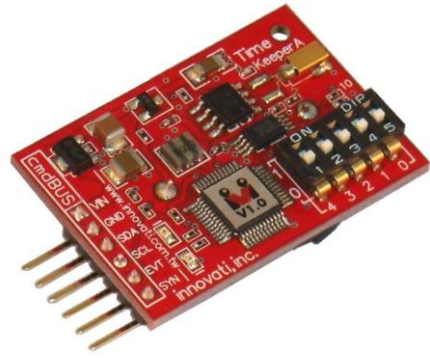


# Time Keeper A

## 計時模組

版本: V2.0



**產品介紹:** 利基 Time Keeper A 模組可以提供多樣的計時功能，內建有星期對應，輸入日期就可以知道當天是星期幾。另外提供第二時間可供設定，以及五個額外的倒數計時器，滿足使用者同時處理多個，需要同時判斷的倒數計時需求。

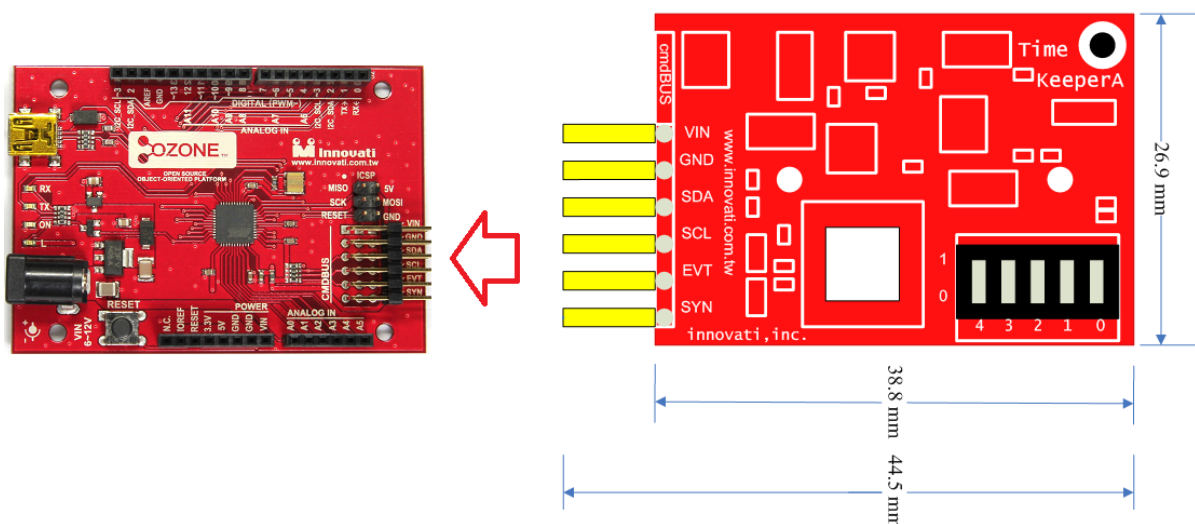
### 應用方向:

- 可加上 LCD 模組即時顯示時間，就是簡單的電子時鐘。
- 與動態的其他種類模組結合，可以於所需要的時間啟動模組動作。
- 可以加上開關做出各種家電定時開關或是預設啟動。

### 產品特色:

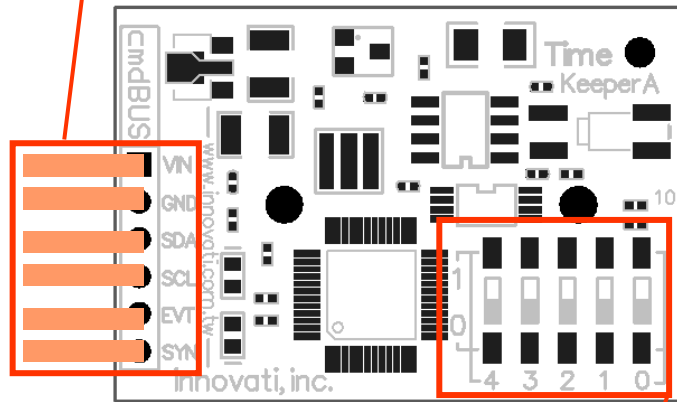
- 西元 2000~2099 年的年，月，日，星期，時，分，秒自動計時。
- 次要時間設定，可提供時，分，秒的輸入設定。
- 5 個倒數計時器，可以同時設定最大以日為單位，最小為秒的倒數值。
- 可透過 I2C 方式，下達指令。

**連接方式:** 直接將 ID 開關撥至欲設定的編號，再將 cmdBUS 連接至 Ozone 上對應的腳位，就可透過 Ozone 執行操作。



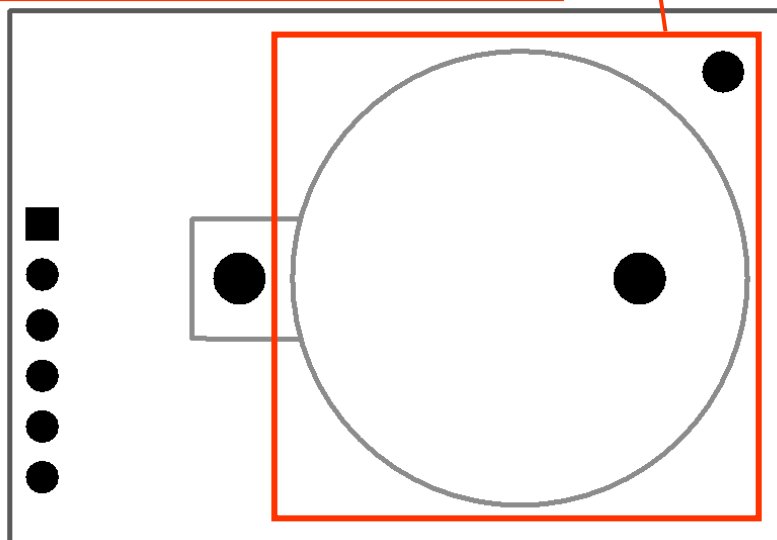
## 產品規格:

cmdBUS接腳，將此處腳位與Ozone對應腳位相接，即可透過Ozone操控Timer模組(連接時請注意腳位對應，將Vin對接Ozone上的Vin腳位，若是腳位錯誤可能造成模組損毀)。



模組編號設定開關，由右至左以二進制設定Timer模組的模組編號，編號可以讓Ozone操控時，判斷想要控制的模組。

電池安裝位置，請將電池安裝於此處，再開始Timer模組的操作。



使用電流量: 約 9 mA

## 操作注意事項:

請使用 CR2032 電池，更換不同電池請確定電池的操作溫度限制。

模組操作溫度	0 °C ~ 70 °C (此為模組溫度，不含電池)
模組儲存溫度	-50 °C ~ 125 °C

指令格式	指令功能
時間設定相關指令	
<b>SetTimeAndDate(uint8_t Year, uint8_t Month, uint8_t Day, uint8_t Hour, uint8_t Minute, uint8_t Second)</b>	依據使用者輸入的 <b>Year</b> (年), <b>Month</b> (月), <b>Day</b> (日), <b>Hour</b> (時), <b>Minute</b> (分), <b>Second</b> (秒), 設定模組現在的時間 ( <b>Year</b> 請輸入 0~99 之間的整數, 代表西元 2000~2099 年, 其他日期輸入請參考單獨設定的範圍值)
時間取得相關指令	
<b>GetTimeAndDate(uint8_t &amp;Year, uint8_t &amp;Month, uint8_t &amp;Day, uint8_t &amp;WeekDay, uint8_t &amp;Hour, uint8_t &amp;Minute, uint8_t &amp;Second)</b>	取得系統現在時間, 年存放於 <b>Year</b> , 月存放於 <b>Month</b> , 日存放於 <b>Day</b> , 星期存放於 <b>WeekDay</b> , 小時存放於 <b>Hour</b> , 分存放於 <b>Minute</b> , 秒存放於 <b>Second</b> ( <b>WeekDay</b> 規則為, 若是星期日以 0 代表, 星期一到六則依序是 1~6, 其他回傳值範圍請參考單獨取回的範圍值)
倒數計時器相關指令	
<b>CountDownTimerOn(uint8_t TimerID)</b>	啟動編號為 <b>TimerID</b> 的倒數計時器。
<b>CountDownTimerOff(uint8_t TimerID)</b>	關閉編號為 <b>TimerID</b> 的倒數計時器。
<b>GetCountDownTimer(uint8_t &amp;TimerID, uint8_t &amp;Day, uint8_t &amp;Hour, uint8_t &amp;Minute, uint8_t &amp;Second)</b>	取得 <b>TimerID</b> 所指定的倒數計時器計時資訊, 請輸入 0~7 之間的整數, <b>Day</b> 會回傳剩餘天數, 為 0~255 之間的整數, <b>Hour</b> 會回傳剩餘小時, 為 0~23 之間的整數, <b>Minute</b> 會回傳剩餘分鐘, 為 0~59 之間的整數, <b>Second</b> 會回傳剩餘秒數, 為 0~59 之間的整數
<b>SetCountDownTimer(uint8_t TimerID, uint8_t Day, uint8_t Hour, uint8_t Minute, uint8_t Second)</b>	設定 <b>TimerID</b> 所指定的倒數計時器計時資訊, <b>Day</b> 請輸入 0~255 之間的整數, <b>Hour</b> 請輸入 0~23 之間的整數, <b>Minute</b> 請輸入 0~59 之間的整數, <b>Second</b> 請輸入 0~59 之間的整數 ( <b>TimerID</b> 請輸入 0~7 之間的整數)

## 範例程式:

```
#include <ozone.h>

TimeKeeperA MyTime(0);           // 設定模組編號為 0

uint8_t CurYear , CurMonth, CurDay , CurWeek , CurHour , CurMinute , CurSecond;

void setup()
{
    Serial.begin(115200);
}













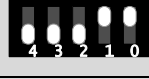



















void loop()
{
    // 主程式

    MyTime.GetTimeAndDate(CurYear, CurMonth, CurDay, CurWeek, CurHour, CurMinute, CurSecond);
    // 取得現在時間

    Serial.print(CurYear);
    Serial.print("/");
    Serial.print(CurMonth);
    Serial.print("/");
    Serial.print(CurDay );
    Serial.print(" ");
    Serial.print(CurHour);
    Serial.print(":");
    Serial.print(CurMinute);
    Serial.print(":");
    Serial.println(CurSecond);
    delay(1000);
}
```

# 附錄

## 1. 模組編號開關對應編號表:

	0		8		16		24
	1		9		17		25
	2		10		18		26
	3		11		19		27
	4		12		20		28
	5		13		21		29
	6		14		22		30
	7		15		23		31