

3S Gauge

User Application Note

目錄

目錄	2
1. 簡介	3
2. 硬體 HW	4
2.1 EVM	4
2.2 通訊轉接板	5
3. 硬體連接說明	6
4. 軟體 GUI	7
4.1 連線	7
4.2 電芯訊息讀取	8
5. 校準說明與流程	10
5.1 連接	10
5.2 軟體 GUI 校準流程	11
5.3 溝通協定(I2C)	16
5.3 流程圖	18
6. MCU 參數空間	21
6.1 連接	21
6.2 溝通協定(I2C)	21
6.3 流程圖	21

1. 簡介

3 串電量計解決方案，包含了 EVM 硬體、圖形化使用者介面(GUI)、訊息溝通所需要之通訊連接板，使用者藉由 GUI 軟體之豐富功能，透過通訊連接板，進而獲取連接在硬體板之電芯運行狀態，透過適當的資料處理與分析，最佳的參數配置與實驗，調適出最好的系統效能，本文將說明解決方案之整體線路連接，與 GUI 之基本操作功能。

硬體 EVM 與軟體 GUI 搭配，可提供的產品開發包括：電動工具類，如電鑽/電板手/軍刀鋸等、無線吸塵器、照明燈、打蠟機、測距儀、汽車救援行動電源等 3 串鋰電池保護板。

2.硬體HW

2.1 EVM

TH251A EVM 如下圖所示，主要介面接口有：

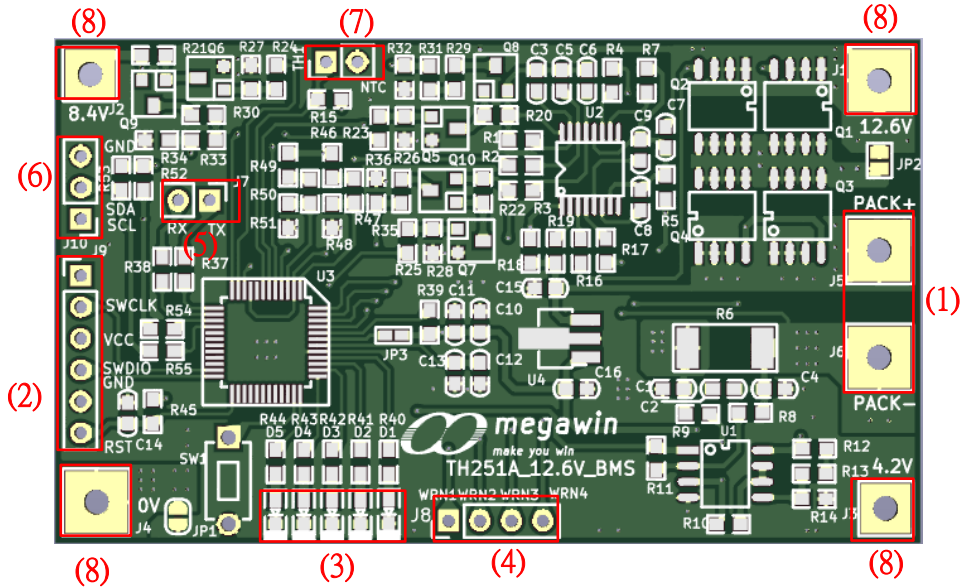


圖 2-1 MSE03GM1AD48 for 3S Gauge EVM 介紹

- (1) 充放電介面接口：充放電同孔接口，接電源對電芯做充電的動作，接負載做電力輸出的行為。
- (2) 序列除錯介面：系統開發使用，例如：更新韌體。
- (3) 電量與健康狀態顯示燈號：提供使用中電芯的剩餘容量燈號顯示與電芯間之電壓差距過大之健康狀態異常燈號提醒。
- (4) 警告狀態介面接口：提供電量(充飽、放空)與使用行為(充電、放電)及狀態(過壓、欠壓、過溫、過流)等 IO 輸出，方便使用者利用此接口，就可得知電芯的運行行為，進而做出其他應用。
- (5) 串口介面接口：系統開發使用，例如：代碼除錯。
- (6) I2C 介面接口：連接通訊板，讓上位機得知電芯訊息(可使用自己系統裝置，透過此介面接口，得知電芯訊息)。
- (7) NTC 介面接口：連接負溫度係數熱敏電阻，量測溫度使用。
- (8) 電芯介面接口：連接電芯模組，接法如下圖所示：

注意:電池芯安裝完成後，保護功能是作動的，因此於 Pack+與 Pack-量測不到電壓，取一條電線碰觸 Pack+ 與電池組正端，以解除保護。

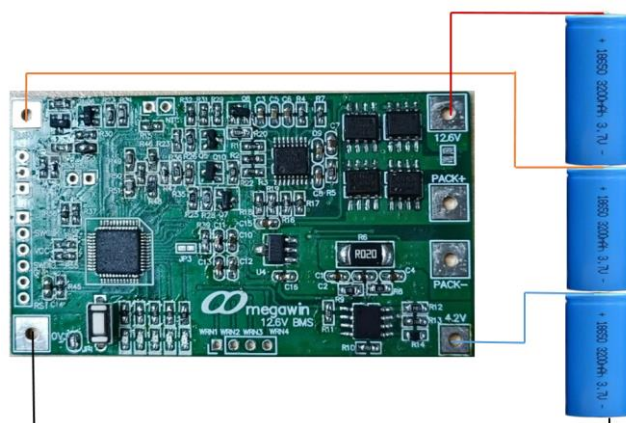


圖 2-2 EVM 連接電芯示意圖

TH251A EVM 尺寸：41 x 70 (mm)，單顆鋰電池尺寸 18 x 65 (mm)



圖 2-3 EVM 實際連接電芯圖

2.2 通訊轉接板

如下圖所示，負責連接電腦 USB 接口與硬體 EVM 之 I2C 接口，從上到下之信號線依序是 GND、SCL、SDA、VCC，目的是將 GUI 之命令，轉換成硬體 EVM 之 I2C 接收的命令，以讀取或運行狀態之電芯訊息，或是寫入更改參數設定等等。



圖 2-4 通訊轉接板

3. 硬體連接說明

(1) TH251A EVM：I2C 接口

(2) TH254A 通訊轉接板：I2C 接口轉 USB 接口

EVM 與 GUI 溝通的媒介，轉譯 GUI 命令給 EVM

注：也可自行透過 I2C(自有裝置)與 EVM 溝通，無需透過 GUI 及通訊轉接板

(3) megawin 3S Gauge 軟體 GUI：USB 接口

圖形化介面顯示讀取資料、立即更新參數確認結果



圖 3-1 硬體連接示意圖

4.軟體GUI

4.1 連線

當整體硬體線路連接成功後，開啟安裝的上位機軟體，將出現如下圖所示畫面。



圖 4-1 開啟上位機軟體

點選搜尋後，並在其左邊選擇通訊板所在的埠口(可從電腦之硬體管理員查知)，接著點選連線鍵，如無錯誤提示訊息出現，畫面多處禁止操作鍵(反白)將被恢復成操控模式，並顯示連線成功訊息，如下圖所示。



圖 4-2 上位機軟體連線成功

4.2 電芯訊息讀取

點選監看鍵(下圖標示 1)，GUI 將從設定顯示頁面，變成監看電芯訊息頁面(下圖 4-4)。



圖 4-3 上位機設定顯示頁面

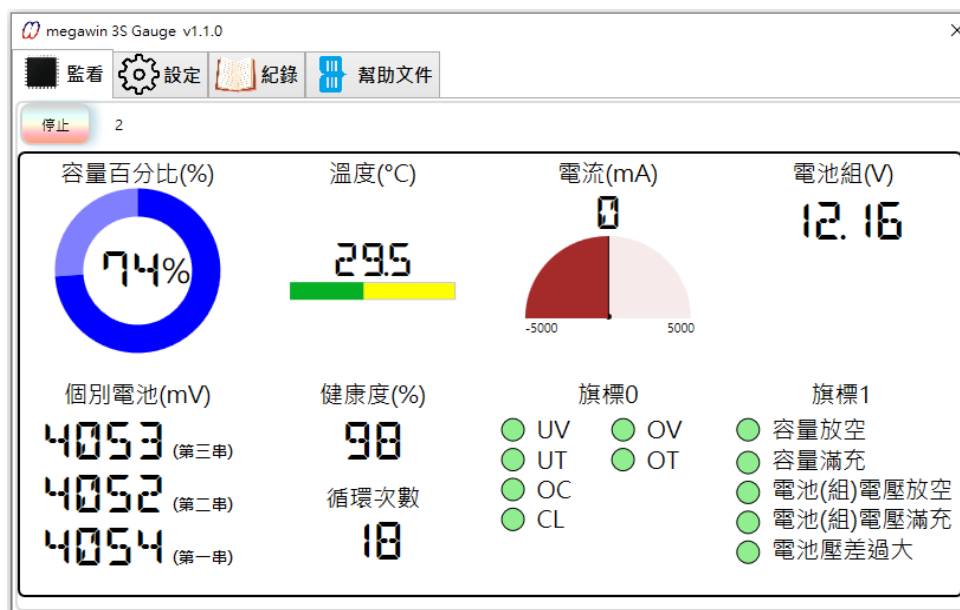


圖 4-4 監看電芯訊息顯示頁面

點選掃描鍵後，此鍵將變停止鍵(下圖標示 1)，此時每 1.5 秒更新一次顯示資料，並且顯示掃描次數(下圖標示 2)，從顯示頁面可即時得知電芯運行的全部狀態。

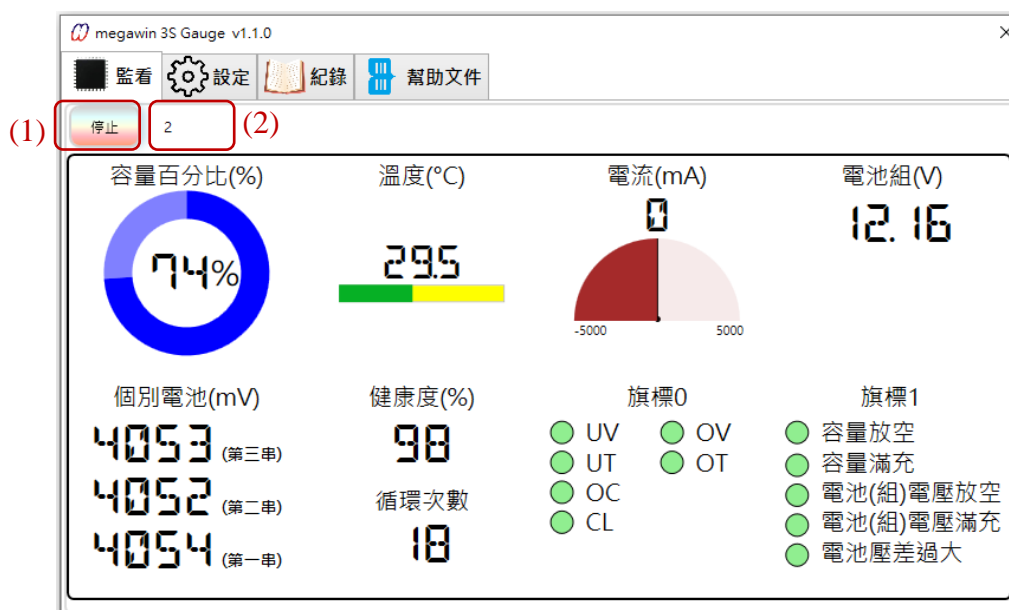


圖 4-5 執行掃描頁面

5. 校準說明與流程

校準 EVM 之電壓、電流、溫度以提升系統量測的精度。

5.1 連接

整體之校準連接示意圖如下圖所示，透過電腦(主控台)讀取 EVM 上之待校準通道(電壓、電流、溫度)，並與連接在待校準通道上之精密儀器電表值做一比較，如果誤差過大時，透過適當的放大增益或偏移量調適，使電腦(主控台)讀取 EVM 上之待校準通道與精密電表之量測，其誤差落在合理範圍內，完成整體通道校準程序。

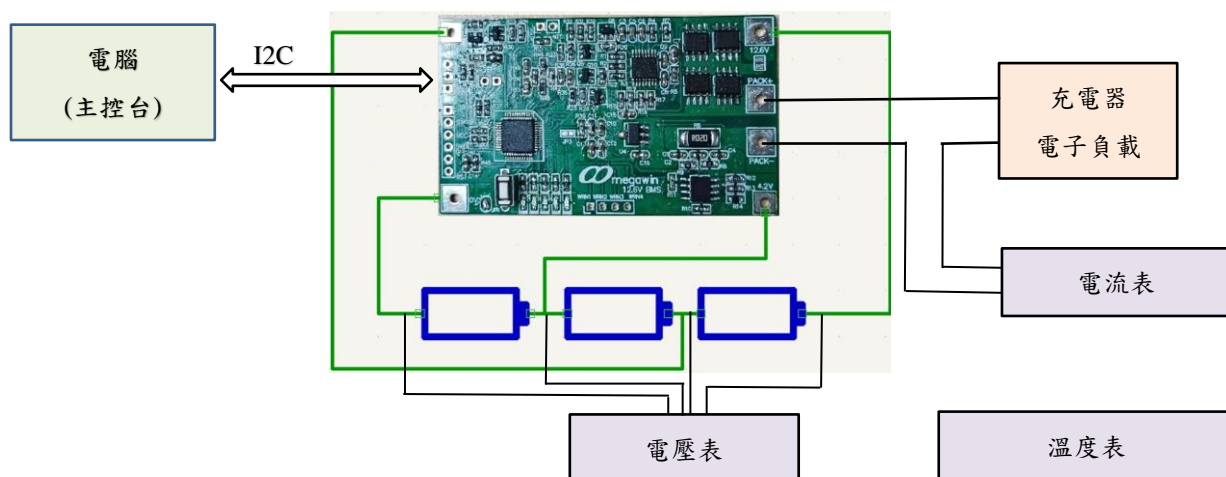


圖 5-1 校準連接示意圖

5.2 軟體GUI校準流程

提供電壓、電流、溫度校準與訊息顯示的功用，校準操作步驟如下：

先按掃描(下圖標示 1)按鈕，啟動讀取 MCU 電壓、溫度程序，此時畫面將如圖 2-12 所示。

註：電流必須在做完偏移校正才會顯示。



圖 5-2 校準頁面

操作的掃描按鈕已經變成停止鈕(下圖標示 1)，並且顯示讀取狀態(下圖標示 2)，MCU 讀回的串接電芯的各別電壓與溫度數值(下圖標示 3)。



圖 5-3 校準頁面-掃描

使用儀表量測各個串接電芯的各別電壓與溫度，將儀表量測數值填於對應欄位(下圖標示 1)，並將所需校準的通道的檢查盒選項打勾(下圖標示 2)，此時包含串接電芯的各別電壓與串接電芯的總電壓需校準，而溫度不需校準，此時需校準的通道的文字將反白，並且 MCU 的數值將不再改變(下圖標示 3)，確認 MCU 讀值與量測數值無誤後，按校準(下圖標示 4)按鈕開始啟動校準，校準完畢後，前面檢查盒選項將回復取消勾選，系統再一次回復掃描狀態，此時 MCU 的讀值將會與量測的數值接近，表示校準成功，無需再校準一次，此時可停止掃描，完成校準程序。



圖 5-4 校準頁面-校準

如果要啟動電流校準，需先做電流偏移量校準，此時不能有充電或放電的行為，避免在有電流狀態，產生錯誤的電流偏移量校準，按下偏移量調整後(下圖標示 1)，電流欄位將從遮蔽狀態回復到正常顯示圖 2-15，允許電流增益校準。



圖 5-5 校準頁面-電流偏移量

如同電壓校準程序，此時可使用充電或放電，讓 MCU 的電流讀值與儀表量測之實際電流值(填於目標欄位(下圖標示 1)完成電流校準程序。

此時整體校準參數儲存於 MCU 參數暫存空間，如果此時 MCU 重置校準參數將全部遺失，因此需按更新按鈕(下圖標示 2)，儲存於 MCU 參數空間保證不遺失。

注：按下更新按鈕前，需先確認無對系統做充放電的動作



圖 5-6 校準頁面-電流校準

5.3 溝通協定(I2C)

EVB 主要通過 I2C 接口與電腦(主控台)溝通，做為從機裝置，其主要可接受的從機地址有兩個，而其主要的封包格式如下：

(1) 格式 1：

從機地址為 0x55 (7 Bit 模式)，命令/地址為 1Byte 長度，主要應用為 register(RAM)區塊的讀寫。

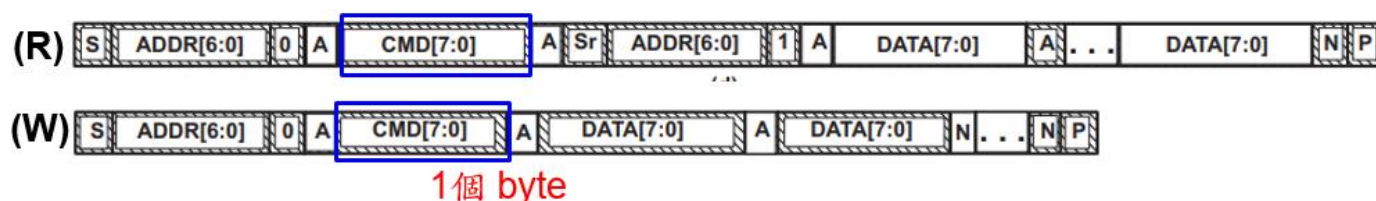


圖 5-7 I2C 格式 1 封包圖

支援讀的命令如下表所示：

表 1 格式 1 讀取命令表

命令/ 地址	名稱	(讀取)資料長度	描述
0x00	Cell_1 voltage	2 bytes	第一串電壓
0x02	Cell_2 voltage	2 bytes	第二串電壓
0x04	Cell_3 voltage	2 bytes	第三串電壓
0x06	Pack voltage	2 bytes	電芯模組總電壓
0x08	NTC temperature	2 bytes	溫度
0x0A	Average current	2 bytes	電流
0x0C	Current offset	2 bytes	電流校正偏移
0x0E	State of charge	2 bytes	剩餘容量階數
0x10	State of health	2 bytes	健康狀態階數
0x12	Cycle	2 bytes	放電循環次數
0x14	Status flag	2 bytes	旗標
0x16	Warning flag	2 bytes	旗標
0x18	Tick counter	2 bytes	上電總時間(單位:2 小時)
0x1C	Force power-up enable	2 bytes	強迫不進入省電模式

支援寫的命令如下表所示：

表 2 格式 1 寫入命令表

命令	名稱	(寫入)資料長度	描述
0x1E	Force power-up disable	2 bytes	恢復正常省電模式
0xE0	Adc control	2 bytes	強迫開啟/關閉 ADC 掃描
0xE1	Current 0A offset calibration	2 bytes	啟動電流偏移校正
0xE2	Flash page erase	2bytes	清除內部 flash 資料
0xF1	Update cell_1 gain to memory	2bytes	更新第一串增益到記憶體
0xF2	Update cell_2 gain to memory	2bytes	更新第二串增益到記憶體
0xF3	Update cell_3 gain to memory	2bytes	更新第三串增益到記憶體
0xF4	Update pack gain to memory	2bytes	更新串接模組增益到記憶體
0xF5	Update ntc offset to memory	2bytes	更新溫度偏移到記憶體
0xF6	Update current gain to memory	2bytes	更新電流增益到記憶體
0xF7	Update current offset to memory	2bytes	更新電流偏移到記憶體
0xF8	Update calibration memory data to flash	2bytes	校正記憶體資料寫入 MCU
0xFE	Software reset	2bytes	重置

(2) 格式 2:

從機地址為 0x14(7 Bit 模式)，命令為 2 Bytes 長度，主要應用為 MCU 設定資料的讀寫。



圖 5-8 I2C 格式 2 封包圖

5.3 流程圖

電腦(主控台)必須操控完成整體的校正程序，其流程步驟說明如下：

(1) 系統喚醒與啟動 ADC 掃描

喚醒系統：必須使用格式 1，下達 0x1C 讀取命令，如果讀回資料為 0x55、0xAA，代表命令已被接收，系統將被強迫喚醒，處於正常模式，等待新命令。

啟動 ADC 掃描：必須使用格式 1，下達 0xE0 寫入命令，寫入資料為 0x00、0x01，啟動內部 ADC 自動掃描動作(強迫)。

關閉 ADC 掃描：必須使用格式 1，下達 0xE0 寫入命令，寫入資料為 0x00、0x00，關閉內部 ADC 自動掃描動作(強迫)。

(2) 讀取現今校準參數

當啟動內部 ADC 自動掃描時，系統將會將現今校準參數置於 register(RAM)空間內，其命令如下表所示：

表 3 校正參數表

命令/地址	名稱	(讀取)資料長度	描述
0x22	Cell1_gain	2 bytes	第一串電壓增益
0x24	Cell2_gain	2 bytes	第二串電壓增益
0x26	Cell3_gain	2 bytes	第三串電壓增益
0x28	Pack_gain	2 bytes	電芯模組總電壓增益
0x2A	Ntc_offset	2 bytes	溫度偏移
0x2C	Curr_gain	2 bytes	電流增益

此時，可使用格式 1，下達 0x22 讀取命令，連續讀取 12 Bytes 資料，將全部校準參數讀出。

(3) 讀取待校準通道

依據需求，從表 1 選擇所用通道，例如：讀取待校準之串接第三串電壓，則需使用格式 1，下達 0x04 讀取命令，連續讀回 2 Bytes，即為所求。

由於系統每一秒更新全部數據，因此建議連續讀取 n 秒，n 筆數據平均以提高精度。並且為了節省校正時間，可同時做不同通道的校正(連續讀出不同的待校準通道)，而不用一次只校正一個通道。

(4) 計算新校準參數

增益校準的計算方式：

假設讀取現今校準參數之增益為 old_gain

校準通道之讀值為 now_value，電表量測之值為 target_value

新的增益(new_gain)計算為：

$$\text{new_gain} = (\text{old_gain} * \text{target_value}) / \text{now_value}$$

偏移計算方式：

假設讀取現今校準參數之偏移為 old_offset

待校準通道之讀值為 now_value，電表量測之值為 target_value

新的偏移(new_offset)計算為：

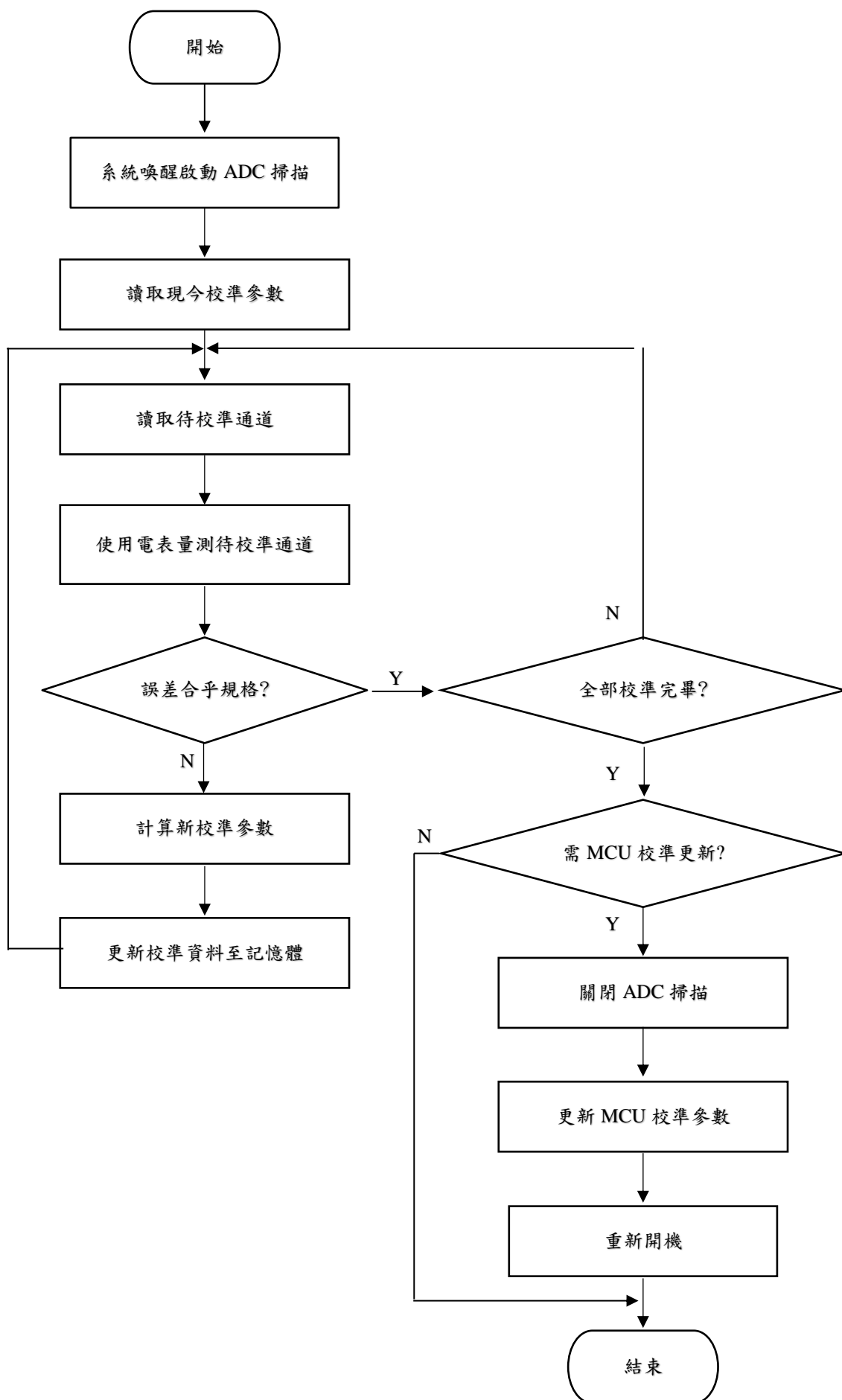
$$\text{new_offset} = \text{target_value} - \text{now_value} + \text{old_offset};$$

(5) 更新校準資料至 MCU 參數空間

當計算出新校準參數，必須通知系統對校正參數做出變更，以便確認新的校正參數是否正確，例如：更新溫度偏移到 MCU 參數空間，則必須使用格式 1，下達 0xF5 (參閱表 2) 寫入命令，寫入資料為之前所計算出的新校準參數。

(6) 更新 MCU 參數空間的校準參數

當全部待校準通道都已校正完畢後，誤差都落在可接受範圍內，此時可要求系統將記憶體內的校準參數寫入至 MCU 參數空間內，以免系統斷電後造成校準資料的流失，因此必須使用格式 1，下達 0xF8(表 2) 寫入命令，寫入資料為 0xA5、0xA5，此時系統收到命令後，將會更新 MCU 參數空間內的校正資料。



6. MCU 參數空間

介紹如何更新 MCU 參數空間上之電芯與系統參數，並合乎系統使用。

6.1 連接

整體之連接示意圖，如下圖所示。透過電腦(主控台)讀取 EVM 上之訊息，確認 I2C 通訊正常後，開始更新全部 MCU 參數空間上之資料，更新完畢後，請再一次讀回確認寫入的資料是正確的，如無錯誤，請重新開機，完成此次正確更新。

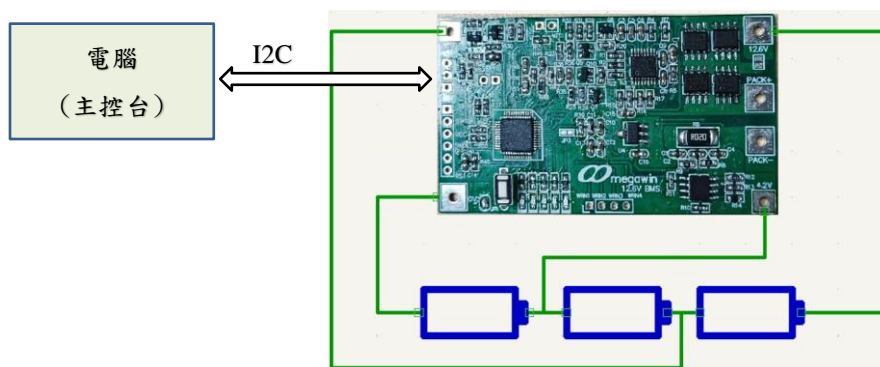


圖 6-1 連接示意圖

6.2 溝通協定(I2C)

與 5-3 章節一致。

6.3 流程圖

電腦(主控台)必須操控完成整體的程序，其流程步驟說明如下：

(1) 系統喚醒與關閉 ADC 掃描

喚醒系統：必須使用格式 1，下達 0x1C 讀取命令，如果讀回資料為 0x55、0xAA，代表命令已被接收，系統將被強迫喚醒，處於正常模式，等待新命令。

關閉 ADC 掃描：必須使用格式 2，下達 0xE0 寫入命令，寫入資料為 0x00、0x00，關閉內部 ADC 自動掃描動作(強迫)。

(2) 抹除 MCU 參數空間頁面

有 2 個頁面需處理，分別為 0x00 與 0x01，0x00 主要為放置參數的地方，0x01 為放置校準的地方，如果此次更新參數，包含了正確校準資料，則必須抹除 0x01 頁面資料，以免系統重新跑起來時，使用到原本第一頁的校準資料，同理，如果此次更新參數，不包含校準資料，則不必抹除 0x01 頁面資料。

必須使用格式 2，下達 0xE2 寫入命令，寫入資料為 0x00 或 0x01，抹除相應的頁面。

(3) 更新全部參數

依據需求，從表 1 選擇所用通道，例如：讀取待校準之串接第三串電壓，則需使用格式 1，下達 0x04 讀取命令，連續讀回 2 Bytes 即為所求。

由於系統每一秒更新全部數據，因此建議連續讀取 n 秒，n 筆數據平均以提高精度。並且為了節省校正時間，可同時做不同通道的校正(連續讀出不同的待校準通道)，而不用一次只校正一個通道。

