

	智慧照明系統標準 第二部：廣域網路介面規範	總號	
		類號	

Intelligent Lighting System Standard, Part II: Wide Area Network

目錄

節次	頁次
1. 前言	3
1.1. 簡介	3
1.2. 文件範圍	3
1.3. 版權聲明	3
2. 引用標準	4
3. 用語釋義	5
3.1. 範圍	5
3.2. 文件格式	5
3.3. 名詞定義	5
4. 廣域網路介面標準概要	7
5. 網路驅動介面規範	8
5.1. 共通規範	8
5.2. 照明監控伺服器端	8
5.3. 照明監控閘道器端	8
6. 通訊協定規範	10
6.1. 適用範圍	10
6.2. 封包傳遞方式	10
6.3. 封包格式	10
6.4. 連線流程與管理	12
7. 封包內容解析	16
7.1. 命令代號索引	16
7.2. 系統監控命令	17
7.3. 裝置監控命令	26
8. 功能模組	37
8.1. 功能模組代號索引	37
8.2. 系統預設功能模組	38
8.3. 電力計	45
8.4. 感測器	53
8.5. 控制器	63

公 布 日 期	修 訂 公 布 日 期
年 月 日	年 月 日

印行年月 年 月

附錄 A	JSON 文件格式概述.....	75
附錄 B	cron 格式概述	78
附錄 C	通用代號定義	80
C.1.	通用回傳值.....	80
C.2.	功能模組代號表.....	81
C.3.	時區名稱對照表.....	82
附錄 D	JSON 授權文件.....	96

1. 前言

1.1. 簡介

本系列標準規定智慧照明系統必要之管理介面、控制與通訊協議，以及智慧型照明設備之功能規格，以適用於道路照明、停車場照明、商業大樓照明與一般家庭照明用途。

智慧照明系統涵蓋設備與設備之間的近端通訊、以及個別照明場域與中控中心之間的廣域網路通訊。且涵蓋之種類包含照明設備、感測設備、自動控制設備、以及資通訊系統。現行之自動控制業界標準，皆僅專注於設備之間的通訊，而未曾將由中控中心、閘道器、直到設備端的通訊以一貫化、系統化的方式進行設計。故特制定本標準，期能統合不同設備、乃至於不同元件之生產製造者，以「標準相容」的精神，降低系統整合複雜度。

本系列標準所涵蓋之範圍較廣，故分成多部文件，個別文件涵蓋系統不同層級之設備。實作者應依據其產品之位階，參照其對應之文件。

1.2. 文件範圍

本文件規範之範圍包含智慧照明系統中，照明監控伺服器與照明監控閘道器之間的網路連線通訊協定。此項規範之目的為確保不同廠商、不同系統間之互通性，包含連線方式、封包格式、設備定址原則等。照明監控伺服器及照明監控閘道器，皆須依循本文件所規範之通訊協定內容進行實作。

1.3. 版權聲明

本文件內所有內容，包含文字、圖片、表格，除特別標記引用來源以外，皆由草擬單位自行整理製作。所引用之第三方技術，亦皆於文件內包含其授權聲明。

—以下空白—

2. 引用標準

下列標準所引用部分視為本規範內容之一部分。對於有標註日期者，僅引用其所標示之版次，對於未標註日期者，則以最新版次（包含所有增、修訂部分）為主。

- 智慧照明系統標準—第一部：系統功能
- 智慧照明系統標準—第三部：場域網路介面規範
- JSON (JavaScript Object Notation), part of ECMA Standard 262, JavaScript Programming Language, 3rd Edition.
- Institute of Electrical and Electronics Engineers (IEEE): MAC-48, EUI-48, and EUI-64, part of standard 802-2001, Feb. 2002, ISBN 0-7381-2941-0.
- RFC-793: Transmission Control Protocol, 1981.
- RFC-1321: The MD5 Message-Digest Algorithm, April 1992.
- RFC-5342: IANA Considerations and IETF Protocol Usage for IEEE 802 Parameters, Sep. 2008.
- crontab: schedule periodic background work – Commands & Utilities Reference, The Single UNIX® Specification, Issue 7, The Open Group.
- UNIX epoch timestamp, part of POSIX.1, IEEE Std 1003.1-2001, 2001.
- IANA Time Zone Database, v.2011n, Nov. 2011.
- IANA US-ASCII, 1986.

3. 用語釋義

3.1. 範圍

本標準適用「智慧照明系統標準—第一部：系統功能」第 3 節所列之名詞定義、規範約束程度定義，以及其他常用詞句定義。

3.2. 文件格式

本文件編輯格式依循 CNS 3689「國家標準之編修規則及格式」之規範，包含小節段落、圖片與表格編號方式等。其餘自行宣告之規則如下：

- 標示十六進位數據時，為與十進位數據區別，本文件一律採用 C/C++、POSIX shell 通用之標示法，於數值前置「0x」。如「0x5A」。
- 用於表示 ASCII 字符、程式碼、變數、常數之英數字符代號，將以粗體等距英文字型標記。
- 用於規範約束程度之用語，亦即「須」、「應」、「可」、「宜」與「不可」，一律使用粗體字型標記。上述單字用於無關「約束」之詞彙時（例如「回應」、「可能」），則使用一般內文字型，無特殊標記。

3.3. 名詞定義

本文件內額外適用下列之特定名詞定義：

- **封包 (Packet)**：網路通訊中一連串特定數據資料的集合。本標準並未規範單一數據拆分/合併之細節。
- **標頭 (Header)**：指一個封包內較早傳輸的部分，通常標頭內會包含定址 (Address) 資訊、封包種類、控制旗標等。
- **承載資料 (Payload)**：指一個封包內緊接著標頭之後傳輸的部分，包含控制、查詢等主要資訊。
- **純文字數據封包 (ASCII-format)**：指封包內的内容 (包括標頭與承載資訊) 係以人類可閱讀之 ASCII 字符標示；亦即只使用英文大小寫字母、數字及分號、逗號、句號、加減乘除符號、換行 (CR/LF) 等可列印字符 (printable symbol) 構成。
- **純文字十進位數據表示法**：指利用 ASCII 字符「0」至「9」表示任意十進位數值。例如十進位數值 50 則標記為字符「5」與字符「0」；前者 ASCII 碼為 0x35，後者 ASCII 碼為 0x30。本文件於此類數值表示，必定會標示其進位計算方式。
- **純文字十六進位數據表示法**：指利用 ASCII 字符「0」至「9」與「A」至「F」表示任意十六進位數值。例如十六進位數值 0x5A，則標記為字符「5」與字符「A」；前者 ASCII 碼為 0x35，後者 ASCII 碼為 0x41。本文件於此類數值表示，必定會標示其進位計算方式。
- **雜湊 (Hash)**：一種由給定數據中建立固定數據格式「特徵值」的演算法。此依特徵值可當做原始數據的「簽名」或「指紋」，用於快速辨識兩比數據之間是否有差異。
- **命令 (Request)**：通訊協定傳輸流程的一個步驟，泛指由照明監控系統伺服器發送封包至照明監控閘道器的動作，或是該動作內傳輸之封包內容。用於後者時適用「**命令封包**」此一名詞。
- **確認 (Confirm)**：通訊協定傳輸流程的一個步驟，泛指照明監控閘道器接收到上述「命令」之後，將執行結果回報給照明監控系統伺服器之動作，或是該動作內傳輸之封包內容。用於後者時適用「**確認封包**」此一名詞。

- **通知 (Indication)**：通訊協定傳輸流程的一個步驟，泛指由照明監控閘道器在未接收到任何命令的情況下，主動發送封包至照明監控系統伺服器的動作，或是該動作內傳輸之封包內容。用於後者時適用「**通知封包**」此一名詞。
- **回覆 (Response)**：通訊協定傳輸流程的一個步驟，泛指照明監控閘道器接收到上述「通知」之後，將執行結果回報給照明監控伺服器之動作，或是該動作內傳輸之封包內容。通常用於後者時適用「**回覆封包**」此一名詞。
- **事件 (Event)**：泛指發生上述「通知」行為的原因，以及後續的動作。

—以下空白—

4. 廣域網路介面標準概要

智慧照明廣域網路介面規範如圖 1 所示，其中包含兩項規範。網路驅動介面之相關規範將在第 5 節說明，通訊協定規範將在第 6 節說明。通訊協定之內容細節與參數定義，將在第 7 節（封包內容解析）及第 8 節（功能模組）說明。

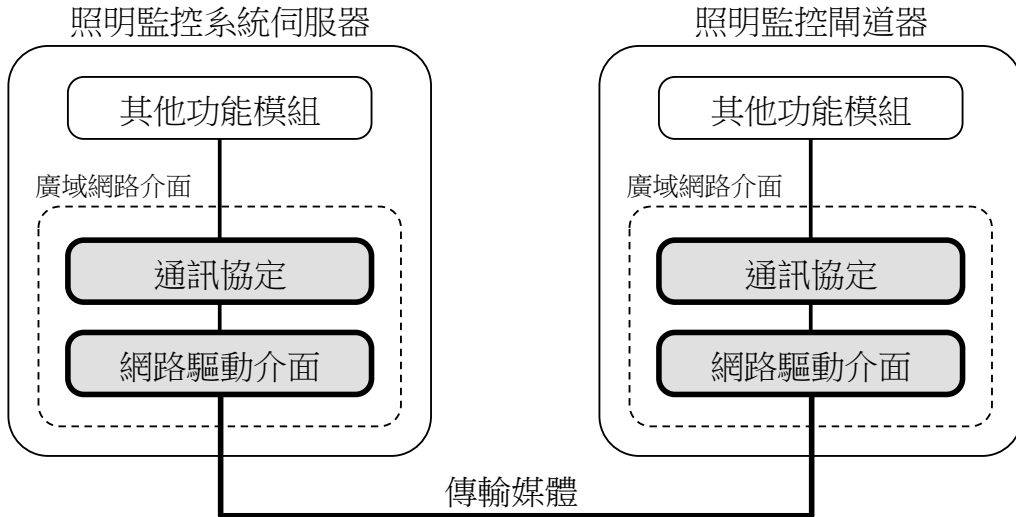


圖 1. 智慧照明廣域網路介面規範

—以下空白—

5. 網路驅動介面規範

5.1. 共通規範

5.1.1. 定義

本節敘述照明監控伺服器端（以下簡稱「監控伺服器」）與照明監控閘道器（以下簡稱「閘道器」）皆須遵循之共通規範。網路驅動介面負責與傳輸網路界接，實際執行資料點對點交換之功能。網路驅動介面實作須具備本節所述之功能特性。

5.1.2. 支援 IP 網路

網路驅動介面須能支援 IP 網路。至少須能支援 IPv4 網路通訊，可選擇性提供 IPv6 網路。

5.1.3. 支援 TCP 傳輸

網路驅動介面須採用 TCP Socket 通訊，並於開始交換資料前先建立網路連線。監控伺服器為 TCP 伺服器端（Server），而閘道器則是客戶端（Client）。連線目標位址與通訊埠均由實作指定，詳細規範參照 5.2 與 5.3。

5.1.4. 安全相關規範

本標準文件並未規範連線安全性相關要求。實作時可自行採用加密連線、虛擬私人網路（Virtual Private Network, VPN）等技術，確保連線的安全性。

若伺服器端之連線具有防火牆（Firewall）或類似設定，亦由實作自行設定相關之允諾規則。

5.2. 照明監控伺服器端

監控伺服器端須提供 TCP 伺服器（Server Socket），接受由閘道器端主動建立之連線。連線兩端皆具備主動傳送資訊至另一方之能力。

TCP 伺服器所關連之 IP 通訊介面宜採用固定實體 IP，若無法提供實體 IP，則應至少提供一固定之外部 IP，以及通訊埠對應。若監控伺服器接受來自網際網路之連線，並且採用網域名稱對應 IP，則監控伺服器主機之網際網路連線須使用靜態對應，不可使用動態網域名稱（Dynamic Domain Name）對應。

單一照明監控伺服器須能允許至少一個以上閘道器連線。最大連線數量由實作依據應用情境、硬體資源等因素自行決定。當連線到達最大設定值時，網路驅動介面仍須維持 TCP 聆聽埠開啟，並在客戶端嘗試新連線後，隨即關閉該連線。

監控伺服器於連線期間須定期發送連線監控封包（PING.REQ，詳細操作說明參見第 7.2.3 節），藉以檢測連線是否失效。監控伺服器網路驅動介面同時須具備主動關閉連線之功能，並由通訊協定層主動要求關閉連線，藉以排除失效或逾時之連線。

5.3. 照明監控閘道器端

閘道器端須提供 TCP 客戶連線（Client Socket），由閘道器主動建立至監控伺服器之連線。連線兩端皆具備主動傳送資訊至另一方之能力。

閘道器端建立連線之時機，以及每次連線期間的長度，皆可由實作自行定義，亦即閘道器可在連線建立後，發送一定量資料，之後由閘道器端主動斷線¹。但在連線建立時，伺服器亦可透過連線回覆封包（CONN.RSP，參見第 7.2.1 節）要求閘道

¹ 此類應用通常常見於感測器之「定期回報」。

器延長連線時間，以便於伺服器發送積存之設定命令。閘道器接收到此項要求後，則**須**依照指示，於指定之延長時間內維持正常連線。

若閘道器端偵測發現非預期之異常連線中斷，則**須**重新嘗試連線，直到恢復連線為止。嘗試重新建立連線之間隔由實作依據網路通訊狀況自行決定；若連線型態為持續連線類型，則嘗試重新建立連線之間隔**宜**採用 30 秒至 600 秒之間隔時間²。同時閘道器於實作時**宜**提供一變更此項設定之功能與方法，以便於調適不同狀況。

閘道器預設連線之目標與通訊埠號碼（Port）由實作自行決定，但**須**提供變更此項設定之功能。連線目標之設定方式**須**能接受 IPv4 網址及網域名稱，**可**選擇性提供 IPv6 連線設定。若閘道器並未與網際網路連線，無法使用公共領域名稱伺服器，則實作**須**自行調整相關設定，以提供網域名稱解析功能；例如藉由手動設定 host table 檔案以提供對應解析。

—以下空白—

² 實作**宜**依照連線狀況動態調整，以降低系統與網路處理負擔。例如斷線後 30 秒重連，若連線失敗則 60 秒後重連，再次連線失敗則 120 秒後重連，依此類推。最大連線重試間距**宜**採用 10 分鐘以下。

6. 通訊協定規範

6.1. 適用範圍

依據第 5 節規範內容，監控伺服器須能同時允許多個閘道器建立網路連線。由於個別連線之間彼此獨立互不干涉，故本節內容將以單一連線為主要描述對象，實作對象建立多重連線時，個別連線均須依循本節規範之內容。

6.2. 封包傳遞方式

當監控伺服器與閘道器之間網路連線建立完成之後（參照 5.2、5.3），監控伺服器與閘道器方能進行封包傳輸。由於 TCP 網路之封包傳輸可由任一方主動發送，為明確表示封包的差異，本標準文件將依循「智慧照明系統標準—第一部：系統功能」第 4.3 小節描述，將監控伺服器定義為通訊主控端，閘道器定義為通訊受控端。

依據此定義，由監控伺服器主動發送給閘道器，包含控制、查詢命令資訊之封包，於本文件中將稱呼為「命令封包」，其餘「確認」、「通知」、「回覆」封包之定義亦依循此一原則，在此不贅述。

通訊接收端對於封包之解析與執行順序，參照 6.3、6.4.4。

6.3. 封包格式

6.3.1. 共通格式

封包之內容格式依循 ECMA Standard 262, JavaScript Programming Language, 3rd Edition 規範，採用 JSON (JavaScript Object Notation) 文件標記格式構成，以便於機器轉寫與解析。所有內容皆由 ASCII 編碼之可列印字元構成。

本標準所採用之通訊封包為一完整 JSON 文件 (JSON Object)，內含數個固定主要參數 (member)；主要參數之數值 (value) 包含純數值或另一個 JSON 文件。為清楚辨識，本文件後續將用「第一階」來形容通訊封包本身的 JSON 文件，並用「第二階」來形容主要參數數值內包含的 JSON 文件。

本標準更規範監控伺服器或閘道器發送之封包，於第一階 JSON 文件完結後（即字符「}」後）須加上連續兩個 DOS 換行字元組 (CRLF，即 0x0D、0x0A，重複兩次，亦即 4 字元長度)，以作為封包與封包之間的時間隔 (boundary)。

JSON 之文件格式說明參照附錄 A。

6.3.2. 封包傳輸限制

監控伺服器發送至閘道器之命令、回覆封包，資料長度限制為 1000 bytes 以內。閘道器於實作時若讀取到單一 JSON 文件超過 1000 bytes 長度尚未完結，則可捨棄整個封包。

本標準規範之命令、回覆封包皆在此一限制以內，實作者自行定義之命令格式亦須依循此一限制。

閘道器發送至監控伺服器之通知、確認封包，資料長度不限制。

6.3.3. 命令、通知封包格式

命令、通知封包之第一階 JSON 文件內，須包含四個主要參數（亦即「第一階」參數）：

表 1. 命令、通知封包第一階參數定義

參數名稱	資料型態
"cmd"	字串
"ack"	整數
"addr"	字串
"payload"	各種形態（依命令而定）

「cmd」欄位內容為不固定長度字串，代表該封包預計執行之動作（Command）。個別命令及其對應之動作將於後續章節內說明。

「ack」欄位內容為整數，數值範圍為 0 至 99,999,999 之整數，代表封包發送端發送封包之累進序號，作為非同步執行時，執行結果回報之辨識代碼。

「addr」欄位為包含 16 個字元之固定長度字串，代表個別裝置之定址，於智慧照明系統內，所有裝置（含具有硬體設備之實體裝置及純軟體之虛擬裝置，包含閘道器在內）皆具有一個裝置代號，並且該代號在系統內是獨一無二，不同裝置不可使用同一個裝置代號。有關裝置代號之細節規範，參照 6.3.5。

「payload」欄位內容為一單純數值、陣列，或是一 JSON 文件，代表該封包之承載（Payload）。不同命令對應不同格式之封包承載，細部格式定義將於個別命令章節內說明。

上述四個主要參數於實際封包內之排列順序，不須依照上表所標記之次序。但本文件為求容易理解，後續範例將沿用此一排序。以下為完整 JSON 命令封包之範例。

```
{
  "cmd": "GSET.REQ",
  "ack": 147534,
  "addr": "A000030000000045",
  "payload": {
    "#EP": 2,
    "LEVEL": 50
  }
}
```

圖 2. 完整 JSON 命令封包範例

上述範例封包為「調光至一半強度」，相關說明參照 7.3.3。

6.3.4. 確認、回覆封包格式

確認、回覆封包之第一階 JSON 文件內，須包含三個主要參數：

表 2. 確認、回覆封包第一階主要參數定義

參數名稱	資料型態
"cmd"	字串
"ack"	整數
"result"	各種形態（依命令而定）

「**cmd**」欄位內容為不固定長度字串，代表該封包預計執行之動作（Command）。個別命令及其對應之動作將於後續章節內說明。

「**ack**」欄位內容為整數，數值範圍為 0 至 99,999,999 之整數，須填入與其對應命令、通知封包中「**ack**」欄位相同之數值。

「**result**」欄位內容為一整數、陣列，或是一完整 JSON 文件，代表執行對應命令/通知封包之執行結果。不同命令對應之執行結果，將於後續章節說明。

上述三個參數於實際封包內之排列順序，不須依照上表所標記之次序，但本文件為求容易理解，後續範例將沿用此一排序。

6.3.5. 定址

裝置代號為 16 字元字串，以十六進位數值可讀取形式表示。格式為十六進位數值可讀取形式，包含字元「0」至「9」，以及「A」至「F」（限大寫）英文字母。亦即該字串代表一 64 bits 無號長整數（unsigned long integer），位元組排序為高位元數值在前（較早送出）。例如：數值「0xA000030000000045」即須標示為「**A000030000000045**」字串。

此一裝置代號在一系統內須為獨一無二，不同裝置不可重複使用同一個辨識碼。裝置代號宜使用該設備之 IEEE MAC-48 硬體辨識碼（亦即俗稱之「網路卡卡號」），或 EUI-64 擴充辨識碼。實際實作之方式本標準不予規範，由實作單位自行指定。

6.4. 連線流程與管理

6.4.1. 連線建立

依據第 5 節規範，監控伺服器與閘道器之間的連線須由閘道器端主動建立。當雙方連線建立完成後，閘道器須傳送一閘道器連線通知封包（參照 7.2.1）至監控伺服器。監控伺服器須依據實作自定之認證方式決定該閘道器是否可建立連線，並比對裝置列表之版本號碼。之後監控伺服器利用上述比對結果，建立一閘道器連線回覆封包，並傳送至閘道器。

上述之連線建立流程須在連線建立後 60 秒內執行完成，若逾時仍未完成，則監控伺服器須關閉該連線。

單一閘道器同時只能建立一個連線。監控伺服器須維持一閘道器狀態連線狀態檢索表；若新建立之連線，經上述通訊協定確認完成後，發現該閘道器已有一啟動之連線存在，則監控伺服器須將舊有的連線關閉。

6.4.2. 裝置登記

當閘道器與監控伺服器依循 6.4.1 規範流程，完成連線建立，且閘道器端接收到閘道器連線回覆封包之後，閘道器隨即須檢視回覆封包之結果。若認證失敗，後續動

作可由實作自行定義：閘道器可重新建立連線、重新發送閘道器連線通知封包、或是通知管理人員。

確認認證結果成功後，閘道器隨即須檢視裝置列表版本是否符合。若不符合，則閘道器須將該閘道器下所管轄之裝置，透過裝置登記通知封包（參照 7.2.1）回報至監控伺服器端。閘道器可在單一封包內包含所有裝置資訊，但若管轄之裝置較多，則閘道器亦可將所有裝置分多次依序發送。裝置登記之排序方式本標準不予規範，由實作自行決定。

當所有裝置資訊皆傳送至伺服器後，閘道器須重新發送閘道器連線通知封包，並依循 6.4.1 規範流程執行。其中伺服器端計算連線逾時時間，由接收到最後一筆裝置登記通知封包之後開始起算。

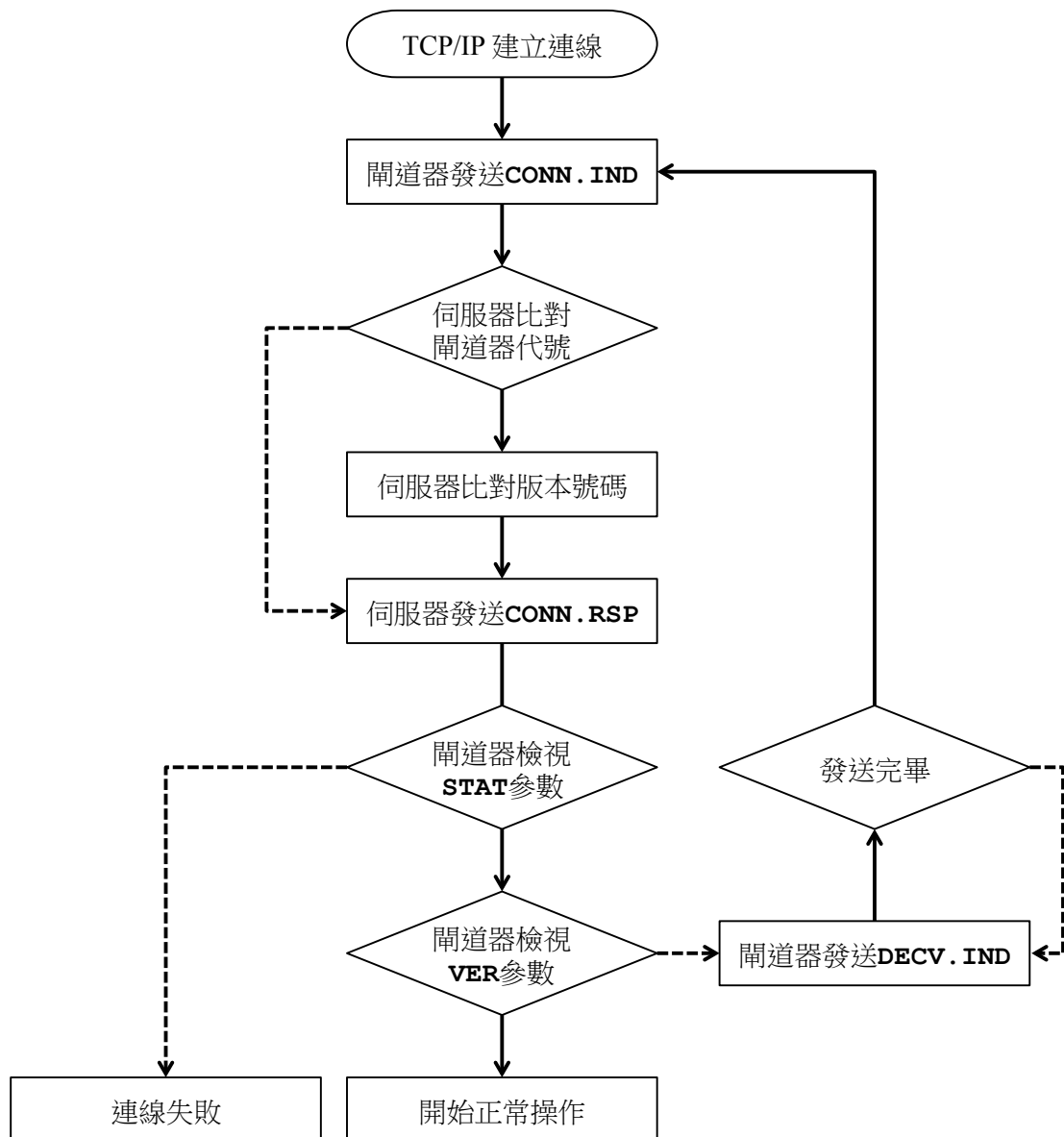


圖 3. 連線建立與裝置登記流程圖，其中虛線代表負邏輯，實線為正邏輯

6.4.3. 路由選徑

監控伺服器端接收到裝置登記通知封包後，應留存一份裝置對應管轄閘道器之列表，以便於在多重連線時作為路由選徑之用。任一受控裝置不可同時與一台以上閘道器建立連線；若實作期望一受控裝置建立多個預備連線，則須選定一台閘道器作為該受控裝置之主要管轄閘道器。

當監控伺服器須要針對單一裝置進行存取時，須利用上述之對應表，尋找該裝置對應之管轄閘道器，並利用該閘道器之連線發送命令封包。

6.4.4. 封包之接收與回應

6.4.4.1. 一般規範

當通訊接收方收到一命令、通知封包，且辨識該封包格式無誤之後，無論該封包所關連之動作是否能執行，皆須發送相對應之確認、回覆封包至原發送方。

監控伺服器與閘道器端對於封包之發送與接收，須能處理非同步、非循序式之執行順序。亦即通訊接收方不須依照接收命令封包的順序回應，通訊發送方亦可在收到前一命令封包回應之前，即發送次一命令封包。

6.4.4.2. 同步執行與非同步執行

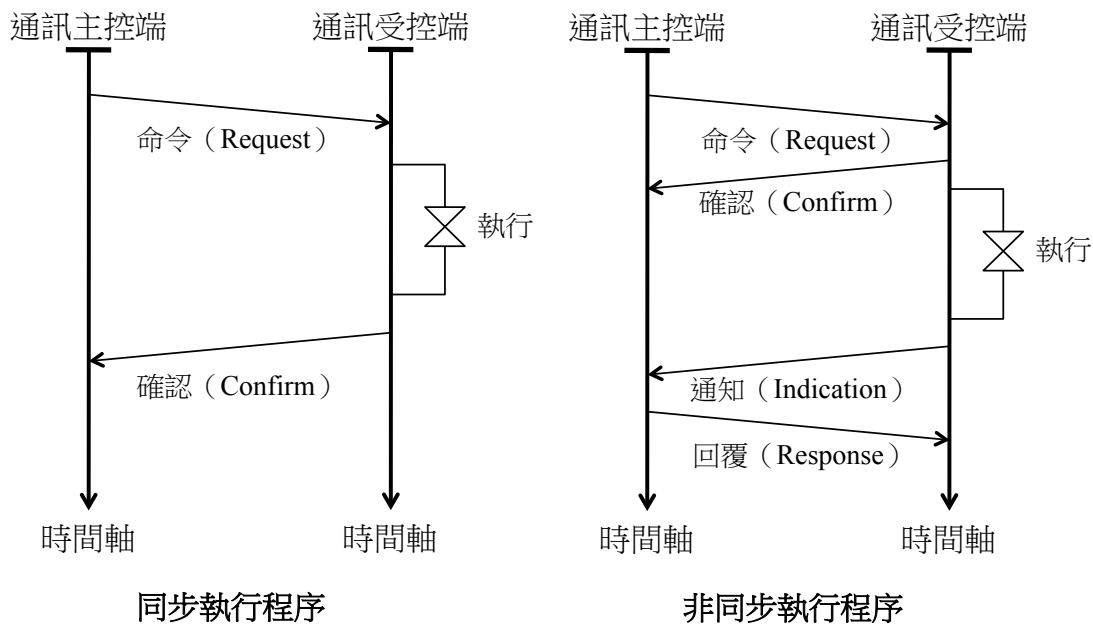


圖 4. 同步執行與非同步執行

「同步執行」為通訊受控端（即閘道器）在接收並辨識命令封包之後，隨即依據該封包之指示執行相關動作，並將執行結果以確認封包格式回傳至通訊主控端（即監控伺服器）。

「非同步執行」為通訊受控端在接收並辨識命令封包之後，隨即發送一確認封包，告知通訊主控端該命令是否能執行。若該命令正確且能執行，則通訊受控端將在該命令執行完畢（或執行失敗）之後，將執行結果以通知封包格式回傳至通訊主控端。

本標準所涵蓋之通訊封包，包含上述兩種執程序。前者（同步執行）通常用於能即時處理之命令，例如取得設定值；後者（非同步執行）則通常應用於須要耗費一

段較長時間處理之命令，或是回應內容較為龐大之命令。

有關個別指令之執程序，將在後續章節分別描述。

6.4.4.3. 監控伺服器端封包發送、接收與回應規範

監控伺服器端在發送命令封包時，**須**在封包「**ack**」參數填入一序號（參照 6.3）。此序號**應**為累計之流水號，或其他具有獨特之識別性之代號。

監控伺服器端於發送命令封包之後，**應**暫存該封包之發送記錄，並於接收到具有相同命令序號之確認封包時，標示該命令封包已執行完畢，並記錄執行結果。若一命令封包超過 60 秒尚未收到相對之確認封包，則標記該命令為逾時。本標準未規範逾期封包之後續動作，實作**可**自行定義重傳、通知等功能。

監控伺服器端於接收通知命令封包後，無論該封包是否能被執行，皆**須**回傳一回覆封包。該回覆封包之「**ack**」欄位**須**填入與通知封包相同之序號。回覆封包其他內容及回覆封包發送時機，**須**依照個別指令之規範，進行實作。

6.4.4.4. 閘道器端封包發送、接收與回應規範

閘道器端之實作通常僅具備較小的記憶體空間與較低之運算能力，故本標準並未規範閘道器端對於接收封包之保存與逾時處理。但閘道器端在發送通知封包時，仍**須**在封包「**ack**」參數中填入一序號。此一序號**應**為累計之流水號，或其他於短期間內具有獨特之識別性之代號。

閘道器端於接收通知命令封包後，無論該封包是否能執行，皆**須**回傳一確認封包。該確認封包之「**ack**」欄位**須**填入與命令封包相同之序號。確認封包其他內容，以及確認封包發送時機，**須**依照個別指令規範，進行實作。

6.4.5. 連線管理

監控伺服器與閘道器**須**定期監控連線，以確保連線能正常運作。除作業系統本身提供之 TCP 連線管理事件以外，監控伺服器端亦**須**定時發送一連線偵測命令封包（參照 7.2.3），閘道器端接收到該命令時，**須**立刻回應連線偵測確認封包。

監控伺服器若在前述之逾時時間內未收到確認封包，則**須**立即重新發送一次連線偵測命令。若連續逾時達一定次數以上，則認定該連線已經失效，監控伺服器**須**主動關閉該連線。逾時次數**宜**以三次為限，但可由施作單位一致同意變更。

監控伺服器進行偵測的週期及逾時次數，**可**依據實際網路通訊狀況而定，本標準不予規範，但兩次偵測之間的時間**應**為 3 分鐘以上，1 小時以下，連續逾時判定次數**應**為三次以上。監控伺服器於實作時**宜**取得相關施作單位之一致同意，或提供變更此項設定之功能與方法，以便於調適不同狀況。

—以下空白—

7. 封包內容解析

7.1. 命令代號索引

本節描述照明監控伺服器（以下簡稱「監控伺服器」）與照明監控閘道器（以下簡稱「閘道器」）之間傳輸封包之內容定義。

依據執行方式，封包分為「系統監控」以及「裝置監控」兩大類別。系統監控封包主要用於連線之建立與維護，裝置監控封包則用於受控裝置之狀態回報與控制，如燈具、感測器等。下表為本標準所涵蓋之封包種類與命令代號。

表 3. 命令代號索引列表

功能分類	命令代號	定義	對應章節
系統監控	CONN . IND	閘道器連線通知	7.2.1
	DEVC . IND	裝置登記通知	7.2.2
	PING . REQ	連線偵測命令	7.2.3
	SYNC . IND	系統時間校正通知	7.2.4
裝置監控	GGET . REQ	取得裝置屬性命令	7.3.2
	GSET . REQ	設定裝置屬性命令	7.3.3
	GUPD . IND	裝置屬性更新通知	7.3.4
	GERR . IND	屬性存取結果通知	7.3.1

上表所列以外之命令代號皆為保留，實作可自行定義私有命令代號，但命令代號須迴避使用上表所列之命令代號。本節其餘部分將依命令代號字母排序，逐項介紹個別命令之細部定義。

7.2. 系統監控命令

本節將依命令代號字母排序，介紹系統功能指令之細節定義。

—接續次頁—

7.2.1. CONN.IND 閘道器連線通知

7.2.1.1. 發送時機

閘道器與監控伺服器之間連線建立之後，閘道器須隨即發送閘道器連線通知封包至監控伺服器。

閘道器於發送完畢裝置列表通知封包之後，亦須發送閘道器連線通知封包至監控伺服器。

7.2.1.2. 通知封包主要參數

- 「**addr**」欄位須填入閘道器之裝置代號（16 字元）。
- 「**cmd**」欄位須填入字串「**CONN.IND**」。
- 「**ack**」欄位應填入閘道器端計數之發送封包序號（流水號）。

7.2.1.3. 通知封包承載內容

閘道器連線通知封包之「**payload**」欄位為 JSON 文件，其下包含以下參數：

表 4. 閘道器連線通知封包承載第二階參數

參數名稱	資料型別	說明
" VER "	布林	版本號碼是否相同
" ZONE "	字串	閘道器設備所在位置之時區

「**VER**」欄位資料型態為字串，代表裝置列表版本號碼。詳見第 7.2.1.7 節說明。

「**ZONE**」欄位資料型態為字串，代表閘道器使用之時區（含日光節約時間）設定，參照附錄 C 時區名稱對照表。台灣地區為「**Asia/Taipei**」。

7.2.1.4. 接收端執行動作

本命令為同步執行。監控伺服器端接收到該命令時須立即執行，並在獲得結果之後立即發送回覆封包。

監控伺服器端接收到閘道器連線通知封包時，須依據封包標頭之「**addr**」欄位，比對該閘道器之身分認證。

若身分認證失敗，監控伺服器須隨即發送含有相對應錯誤代號之回覆封包至伺服器，並關閉該連線。

若身分認證成功，監控伺服器須檢視目前已存在之連線，是否有關連至相同閘道器。若有關連至相同閘道器之既存連線存在，則監控伺服器須關閉該既存連線，並以新建立之連線作為該閘道器之預設通訊管道。

身分認證成功之後，監控伺服器須比對裝置列表版本號碼與目前伺服器端之最新記錄是否吻合，並將查證結果填入閘道器連線回覆封包至閘道器端，告知已完成連線程序。

7.2.1.5. 回覆封包主要參數

- 「**cmd**」欄位須填入字串「**CONN.RSP**」。
- 「**ack**」欄位須填入與通知封包「**ack**」欄位相同之數值。

7.2.1.6. 回覆封包承載內容

閘道器連線回覆封包之「**result**」欄位為 JSON 文件，其下包含以下參數：

表 5. 閘道器連線回覆封包承載第二階參數

參數名稱	資料型別	說明
" VER "	布林	版本號碼是否相同
" HOLD "	整數	維持連線時間長度
" STAT "	整數	執行狀況

「**VER**」欄位資料型態為布林值，代表裝置列表版本號碼與伺服器端之最新記錄是否相同。若相同則填入「**true**」，若不相同則填入「**false**」。

「**HOLD**」欄位資料型態為整數，代表伺服器端要求閘道器維持連線狀態的最短時間長度，單位為分鐘。若伺服器不須要求閘道器維持連線、或該應用情境無此必要，則填入數值 0。

「**STAT**」欄位資料型態為整數，須依照實際執行狀況，由下表中選擇一對應數值。

表 6. 閘道器連線回覆封包執行結果定義

數值	執行狀況	說明
100	OK	認證成功
101	封包格式錯誤	整體 JSON 文件解析錯誤
102	主要參數格式錯誤	addr 、 cmd 、 seq 對應數值格式錯誤
103	封包承載格式錯誤	payload 欄位並非有效 JSON 文件，或其下找不到應有的第二階參數
105	數值超出範圍	payload 欄位第二階參數對應數值錯誤
201	服務已達上限	監控伺服器無法接受新的連線
204	閘道器未登記	addr 欄位指定之目標裝置並未在系統內預先登錄

7.2.1.7. 版本編號字串

版本編號字串之內容為裝置列表之雜湊（Hash），計算方式為 MD5（RFC 1321），數值表示方式為 32 字元長度之純文字十六進位數值表示法，英文字母大小寫視為相同。雜湊計算之標的內容為所有裝置之裝置代號串接，如下圖所示：

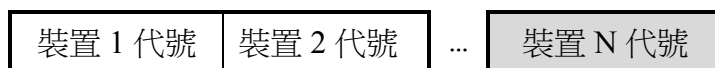


圖 5 版本編號計算雜湊之標的內容

其中裝置代號皆為純文字十六進位數值表示之「字串」，所有十六進位英文字符皆使用大寫英文字母（**A、B、C、D、E、F**）表示。計算時由裝置 1 代號開始計算，後接續裝置 2 代號、裝置 3 代號，依此類推。裝置與裝置之間不加任何空白字符或連接符號。

7.2.1.8. 後續處理

閘道器接收到閘道器連線回覆封包後，若回覆封包之執行結果不成功，或閘道器與監控伺服器連線中斷，後續動作本標準不予規範，由實作自行定義。實作可執行重新連線動作，或發送通知給相關管理人員。

若回覆封包之「**VER**」欄位內容為「**false**」，則閘道器須準備發送裝置登記通知封包（參照 6.4.2），並於裝置登記通知封包發送完畢後，重新發送本連線通知封包。

若「**VER**」欄位內容為「**true**」，則閘道器隨即可開始執行正常操作。

若閘道器之應用情境會主動中斷與伺服器之間的連線，且「**HOLD**」欄位內容數值不為 0，則閘道器須維持連線至少達「**HOLD**」欄位內容數值以上之時間長度，單位為分鐘。

7.2.1.9. 範例

一典型閘道器連線通知封包範例如下：

```
{
  "cmd": "CONN.IND",
  "ack": 1251,
  "addr": "F026B85D006100A0",
  "payload": {
    "VER": "9e107d9d372bb6826bd81d3542a419d6",
    "ZONE": "Asia/Taipei"
  }
}↵↵
```

圖 6. 閘道器連線通知封包範例

閘道器連線回覆封包範例如下：

```
{
  "cmd": "CONN.RSP",
  "ack": 1251,
  "result": {"VER": true, "HOLD": 0, "STAT": 100}
}↵↵
```

圖 7. 閘道器連線回覆封包範例

7.2.2. DEVC.IND 裝置登記通知

7.2.2.1. 發送時機

閘道器接收到閘道器連線回覆封包，經檢查發現裝置列表版本比對結果為不相符，便須發送裝置登記通知封包至監控伺服器（參照 6.4.2）。

7.2.2.2. 通知封包主要參數

- 「**addr**」欄位須填入閘道器之裝置代號（16 字元）。
- 「**cmd**」欄位須填入字串「**DEVC.IND**」。
- 「**ack**」欄位應填入閘道器端計數之發送封包序號（流水號）。

7.2.2.3. 通知封包承載內容

裝置登記通知封包「**payload**」欄位為一陣列，內容定義如下：

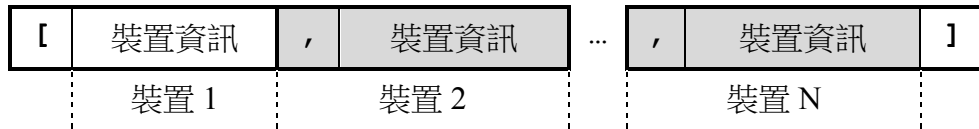


圖 8. 裝置登記通知封包承載

其中「裝置資訊」欄位須填入該閘道器所管轄之裝置資訊。該欄位之資料型態為 JSON 文件，內含兩個參數，定義如下：

表 7. 裝置資訊次欄位（第三階）參數

參數名稱	資料型別	說明
"ID"	字串	裝置代號
"CL"	陣列	裝置功能模組代號列表

「ID」欄位資料型態為字串，須填入待回報裝置之裝置代號（16 字元）。

「CL」欄位資料型態為陣列，其中個別元素資料型態為整數，須填入該裝置所有功能模組代號（Cluster ID）。陣列內之資料為有序（Ordered）；其中第一個元素³即代表「1 號服務終端對應之功能」，第二個元素即代表 2 號服務終端對應功能，依此類推。若一服務終端之功能模組已停用，則填入固定值「255」。

一裝置登記通知封包內須包含至少一個裝置資訊。單一裝置登記通知封包內，包含之裝置資訊數量無限制，以封包大小以及傳輸媒體特性。

7.2.2.4. 接收端執行動作

監控伺服器接收到裝置登記通知時，須發送一裝置登記回覆封包至閘道器。

7.2.2.5. 回覆封包內容

- 「**cmd**」欄位須填入字串「**DEVC.RSP**」。
- 「**ack**」欄位須填入與通知封包「**ack**」欄位相同之數值。

³ 以 C/C++/Java 語言表示，第一個元素即為 array[0]。

- 「**result**」欄位為資料型態為「整數」，須依據執行狀況，由下表選擇一對應數值。

表 8. 裝置登記回覆封包執行結果定義

數值	執行狀況	說明
100	OK	登記成功
101	封包格式錯誤	整體 JSON 文件解析錯誤
102	主要參數格式錯誤	addr 、 cmd 、 seq 對應數值格式錯誤
103	封包承載格式錯誤	payload 欄位並非有效陣列，或其下之裝置資訊次欄位解析錯誤
104	資訊不完整	部分裝置資訊之 CL 陣列是空值
201	服務已達上限	監控伺服器無法接受新的裝置登記
204	閘道器未登記	該閘道器 並未在系統內預先登錄
301	未定義該項服務	部分裝置資訊內含有未定義的功能模組代號

7.2.2.6. 後續處理

監控伺服器接收到裝置登記通知封包後，應依據封包內容，建立並登記相關裝置清單，並建立路由選徑等相關記錄（參照 6.4.3）。

閘道器端若無法在單一封包內將所有管轄之裝置全部納入，則須在接收到發送裝置登記回覆封包之後，繼續發送裝置登記通知封包，直到所有管轄之裝置都完成登記為止。但若回覆封包執行結果為「201」或「204」，則停止後續發送。

裝置登記通知封包發送完畢後，須重新發送閘道器連線通知封包（**CONN.IND**）。

7.2.2.7. 範例

```
{
  "cmd": "DECV.IND",
  "ack": 1252,
  "addr": "F026B85D006100A0",
  "payload": [
    {"ID": "A000030000000045", "CL": [101, 201]},
    {"ID": "B000010000000031", "CL": [103, 101, 101, 101]},
    {"ID": "D000030000000044", "CL": [151]},
    {"ID": "E000090000000158", "CL": [201, 202]}
  ]
}↵↵
```

圖 9. 裝置登記通知封包範例

上圖為一典型裝置登記通知封包範例，其中包含四個裝置：

- 「**A000030000000045**」：附帶繼電器開關之單相電力計，該裝置有兩個服務終端，1 號服務終端對應的功能是單相電力計（功能模組代號「**101**」），2 號終端對應的是繼電器（功能模組代號「**201**」）。
- 「**B000010000000031**」：三相多迴路電力計，該裝置總共配置了四個迴路，第一個迴路（1 號服務終端）是三相四線總迴路配置（功能模組代號「**103**」），其餘迴路（2 至 4 號服務終端）皆是三相四線式配線中的單一迴路。
- 「**D000030000000044**」：感測器（光度計），該裝置只有一個服務終端（功能模組代號「**151**」）。
- 「**E000090000000158**」：可調光照明燈具，該裝置有兩個服務終端，1 號服務終端是繼電器（功能模組代號「**201**」），2 號服務終端是單頻道調光器（功能模組代號「**202**」）。

個別功能模組代號之定義，參照第 8 節說明。

— 接續次頁 —

7.2.3. PING.REQ 連線偵測命令

7.2.3.1. 發送時機

當閘道器與監控伺服器建立連線時，監控伺服器須定期發送連線偵測命令封包至閘道器，以確認雙方連線正常（參照 6.4.5）。

7.2.3.2. 命令封包內容

- 「**addr**」欄位須填入閘道器之裝置代號（16 字元）。
- 「**cmd**」欄位須填入字串「**PING.REQ**」。
- 「**ack**」欄位應填入監控伺服器端計數之發送封包序號（流水號）。
- 「**payload**」欄位為空值（填入字串「**null**」）。

7.2.3.3. 接收端執行動作

本命令為同步執行，閘道器接收到連線偵測命令封包後，須隨即發送一連線偵測確認封包至監控伺服器。

7.2.3.4. 確認封包內容

- 「**cmd**」欄位須填入字串「**PING.CFM**」。
- 「**ack**」欄位須填入與命令封包「**ack**」欄位相同之數值。
- 「**result**」欄位為「整數」，數值為固定值「**100**」（表示「OK」）⁽⁴⁾。

7.2.3.5. 後續處理

若監控伺服器在逾時時間內未收到連線偵測確認封包，則依循 6.4.5 規範執行。

7.2.3.6. 範例

一典型連線偵測命令封包範例如下：

```
{
  "cmd": "PING.REQ",
  "ack": 11504,
  "addr": "F026B85D006100A0",
  "payload": null
}↵↵
```

圖 10. 連線偵測命令封包範例

— 接續次頁 —

⁴ PING.REQ 命令僅作為確認通訊連線正常，故不須回報錯誤。

7.2.4. SYNC.IND 時間校正通知

7.2.4.1. 發送時機

當閘道器須要校正系統時間時，可採用標準 NTP 方式校時。若閘道器作業系統未實作 NTP 客戶端功能，或是該處網路環境不允許使用 NTP 協定，則閘道器須利用本通知封包進行校時。

使用本機制進行校時，閘道器首先發送一時間校正通知封包至伺服器端，伺服器端則將該系統之 UTC 系統時間填入時間校正回覆封包，回傳至閘道器，以便於閘道器設定系統時間。

本機制並未考慮閘道器端與監控伺服器端網路延遲，亦未規範延遲校正等相關機制。故採用本方式校時，通訊雙方之系統時間可能存在差異。若在實作環境中網路延遲過大，以致系統時間差異會影響到排程、定時回報等功能，則實作者宜採用其他方式進行系統校時。

7.2.4.2. 通知封包內容

- 「**addr**」欄位須填入閘道器之裝置代號（16 字元）。
- 「**cmd**」欄位須填入字串「**SYNC.IND**」。
- 「**ack**」欄位應填入閘道器端計數之發送封包序號（流水號）。
- 「**payload**」欄位為空值（填入字串「**null**」）。

7.2.4.3. 接收端執行動作

監控伺服器端接收到時間校正通知封包時，隨即擷取監控伺服器之 UTC 系統時間，填入時間校正回覆封包，並回傳至閘道器。

7.2.4.4. 回覆封包內容

- 「**cmd**」欄位須填入字串「**SYNC.RSP**」。
- 「**ack**」欄位須填入與通知封包「**ack**」欄位相同之數值。
- 「**result**」欄位資料型態為整數，須填入監控伺服器建立該回覆封包時的系統時間，以 Unix epoch timestamp 格式表示，亦即 UTC 時間 1970 年 1 月 1 日 00:00:00 起算經過的秒數。

7.2.4.5. 後續執行動作

閘道器接收到時間校正回覆封包後，即須設定閘道器端系統時間。時區轉換相關動作須由閘道器端自行處理。

7.2.4.6. 範例

```
{
  "cmd": "SYNC.RSP",
  "ack": 20856,
  "result": 1325428465
}↵↵
```

圖 11. 時間校正回覆封包範例

其中「**1325428465**」此一數值即表示 2012-01-01 13:34:25 UTC，亦即台灣地區時間 2012-01-01 21:34:25。

7.3. 裝置監控命令

依據「智慧照明系統標準—第一部：系統功能」第 5 節內容規範，對於受控裝置的監控，將以存取「功能模組」的「屬性」方式進行。因此監控命令亦能簡化為四項：讀取、寫入、通報及錯誤通知。

本節將依命令代號字母排序，介紹裝置監控命令之細節定義。個別功能模組的屬性，將在第 8 節介紹。

—接續次頁—

7.3.1. GERR.IND 屬性存取結果通知

7.3.1.1. 發送時機

當閘道器執行「**GGET.REQ**」命令、讀取遠端受控裝置屬性時，若該讀取命令發生錯誤，則須發送屬性存取結果通知封包至監控伺服器，告知錯誤原因。

當閘道器執行「**GSET.REQ**」命令、設定遠端受控裝置屬性時，當非同步執行完成時，無論成功或失敗，皆須發送屬性存取結果通知封包至監控伺服器，告知該命令執行結果。

若「**GGET.REQ**」、「**GSET.REQ**」命令之目標為閘道器本身時，不須發送屬性存取結果通知封包。

注意「**GERR.IND**」為回報非同步執行結果的「通知」封包，由閘道器端主動發出。回報標的之命令封包（**GGET.REQ**、**GSET.REQ**）序號係由「**GERR.IND**」封包承載（**payload**）中特定欄位標記，並非由封包主要參數「**ack**」欄位記錄。

7.3.1.2. 通知封包主要參數

- 「**addr**」欄位須填入閘道器之裝置代號（16 字元）。
- 「**cmd**」欄位須填入字串「**GERR.IND**」。
- 「**ack**」欄位應填入監控伺服器端計數之發送封包序號（流水號）。

7.3.1.3. 通知封包承載內容

設定裝置屬性命令封包「**payload**」欄位之資料型態為 JSON 文件，內含兩個固定參數，定義如下：

表 9. 屬性存取結果通知封包承載第二階參數

參數名稱	資料型別	說明
"IND"	整數	發生錯誤的命令封包序號
"ERR"	整數	執行結果

「**IND**」欄位資料型態為整數，須填入與其相關命令封包（**GGET.REQ**、**GSET.REQ**）主要參數「**ack**」欄位相同之數值。

「**ERR**」欄位資料型態為整數，須依執行狀況，由下表選擇一數值填入。

表 10. 屬性存取非同步執行結果代號

數值	執行狀況	說明
100	執行成功	GSET.REQ 設定命令已成功發送至受控裝置端，且命令能被執行
301	未定義該項服務	#EP 欄位格式不正確，或包含不正確的數值（例如負值）
302	未定義該項屬性	封包範疇內包含不支援之屬性

數值	執行狀況	說明
303	屬性數值格式錯誤	GSET.REQ 欲設定之屬性格式錯誤
304	裝置不支援該項服務	裝置於指定終端代號無對應之服務，或是指定屬性無法設定（Read-only）
403	場域網路無法傳輸	閘道器無法與指定裝置通訊

7.3.1.4. 接收端執行動作

監控伺服器接收到屬性存取結果通知時，須隨即發送一屬性存取結果回覆封包至閘道器。

7.3.1.5. 回覆封包內容

- 「cmd」欄位須填入字串「**GERR.RSP**」。
- 「ack」欄位須填入與命令封包「ack」欄位相同之數值。
- 「result」欄位為「整數」，數值為固定值「100」（表示「OK」）。

7.3.1.6. 後續處理

後續動作本標準未定義，實作可依據應用情境自行定義。例如「**GSET.REQ**」命令，可在確認執行成功以後，變更監控伺服器端資料庫內的設備狀態，或是在接到錯誤訊息時將錯誤通報管理人員。

7.3.2. GGET.REQ 取得裝置屬性命令

7.3.2.1. 發送時機

取得裝置屬性命令封包由監控伺服器主動發送至閘道器，以便從指定受控裝置取得特定屬性，閘道器接收此命令封包後，便須由指定裝置擷取屬性回傳。

7.3.2.2. 命令封包主要參數

- 「**addr**」欄位須填入指定裝置之裝置代號（16 字元）。
- 「**cmd**」欄位須填入字串「**GGET.REQ**」。
- 「**ack**」欄位應填入監控伺服器端計數之發送封包序號（流水號）。

7.3.2.3. 命令封包承載內容

取得裝置屬性命令封包「**payload**」欄位之資料型態為 JSON 文件，內含兩個參數，定義如下：

表 11. 屬性讀取次欄位（第二階）內含參數

參數名稱	資料型別	說明
"#EP"	整數	服務終端代號
"ATT"	陣列	屬性列表

「#EP」欄位資料型態為整數，須填入指定功能模組對應之服務終端代號。

「ATT」欄位資料型態為陣列，其中個別元素資料型態為字串，須填入欲取得之屬性名稱。

7.3.2.4. 接收端執行動作

本命令為非同步執行。閘道器接收到取得裝置屬性命令時，須隨即發送一取得裝置屬性確認封包至監控伺服器。

7.3.2.5. 回覆封包內容

- 「**cmd**」欄位須填入字串「**GGET.CFM**」。
- 「**ack**」欄位須填入與命令封包「**ack**」欄位相同之數值。
- 「**result**」欄位為資料型態為「整數」，須依據執行狀況，由下表選擇一對應數值。

表 12. 取得裝置屬性回覆封包執行結果定義

數值	執行狀況	說明
100	OK	閘道器已開始執行
101	封包格式錯誤	整體 JSON 文件解析錯誤
102	主要參數格式錯誤	addr 、 cmd 、 seq 對應數值格式錯誤
103	封包承載格式錯誤	payload 欄位解析錯誤，或其下之次欄位解析錯誤

數值	執行狀況	說明
201	服務已達上限	閘道器暫時無法處理該請求
401	指定裝置並未登記	該裝置 不屬於閘道器所管轄
402	指定裝置並未上線	該裝置 並未建立場域網路連線

7.3.2.6. 後續處理

裝置屬性取得為非同步執行。閘道器端在接收到裝置屬性取得命令後，隨即開始擷取相關屬性資料。有關屬性擷取之相關實作細節，本標準不予規範，由實作自行定義。實作可於接收到裝置屬性取得命令封包後，隨即發送場域網路通訊命令至受控裝置，即時取得資訊，或使用閘道器端快取（Cache）暫存之資料。但本標準規範：擷取得之屬性資料，皆須以「**GUPD.IND**」裝置屬性更新通知封包回傳至監控伺服器（參照 7.3.4）。

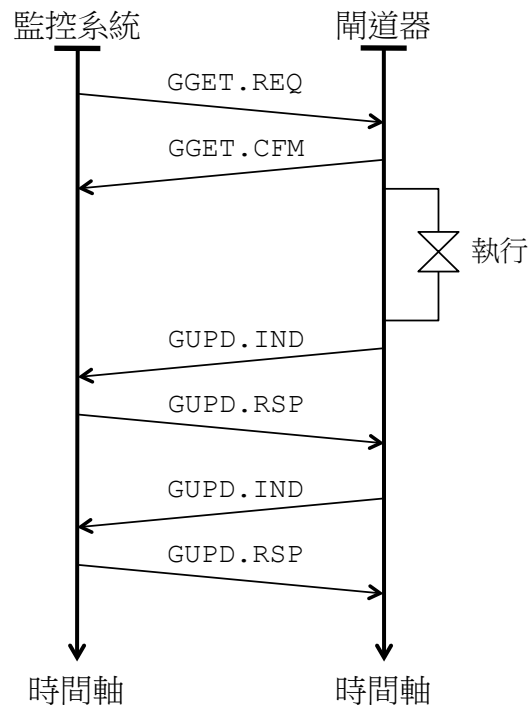


圖 12. 取得裝置屬性非同步執行示意圖：執行成功

「**GUPD.IND**」封包之組成方式本標準不予規範，同時「**GUPD.IND**」封包之內容亦不限定僅包含「**GGET.REQ**」要求擷取之屬性。實作可依實際狀況，將個別屬性拆分、排序、組合，並以不限定數量之封包分批回傳。同時閘道器由接收到「**GGET.REQ**」封包至傳輸「**GUPD.IND**」封包的執行時間，以及相關逾時處理，本標準亦不予規範，皆可由實作自行定義。

若在上述擷取相關屬性的過程中發生錯誤，例如目標裝置無法連線，或是指定存取的功能模組不存在，則閘道器端須發送一「**GERR.IND**」屬性存取結果通知封包至監控伺服器，告知錯誤原因（參照第 7.3.1）。

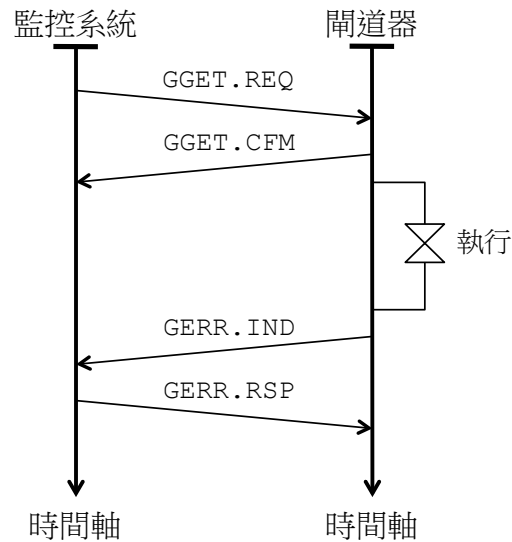


圖 13. 取得裝置屬性非同步執行示意圖：執行失敗

7.3.2.7. 實作相關議題

「**GGET.REQ**」並不適合用於具有時效性、緊急性的訊息擷取。若有此方面之應用情境需求，實作宜使用「閾值警示」類型之服務。亦即以受控裝置端以主動回報方式回傳，而非由監控伺服器擷取資料後再行計算比對判斷。

7.3.2.8. 範例

一典型取得裝置屬性命令封包範例如下：

```
{
  "cmd": "GGET.REQ",
  "ack": 14235,
  "addr": "B000010000000031",
  "payload": {
    "#EP": 1, "ATT": ["METER.R", "METER.S", "METER.T"]
  }
}↵↵
```

圖 14. 裝置屬性命令封包範例

以上範例為讀取多迴路電力計之量測值：三相四線總迴路個相迴路之基本量測資料（1 號服務終端）。個別功能模組屬性之定義參照第 8 節。

7.3.3. GSET.REQ 設定裝置屬性命令

7.3.3.1. 發送時機

設定屬性命令由監控伺服器主動發送至閘道器，以便控制特定裝置時，或是變更閘道器設定，閘道器接收設定裝置屬性命令後，便須執行相關控制程序。

7.3.3.2. 命令封包主要參數

- 「**addr**」欄位須填入指定裝置（含閘道器）之裝置代號（16 字元）。
- 「**cmd**」欄位須填入字串「**GSET.REQ**」。
- 「**ack**」欄位應填入監控伺服器端計數之發送封包序號（流水號）。

7.3.3.3. 命令封包承載內容

設定裝置屬性命令封包「**payload**」欄位之資料型態為 JSON 文件，內含一固定參數，以及不定數量之功能屬性參數。其中固定參數定義如下：

表 13. 設定裝置命令封包承載第二階固定參數

參數名稱	資料型別	說明
"#EP"	整數	服務終端代號

「#EP」欄位資料型態為整數，須填入指定功能模組對應之服務終端代號。

其餘功能屬性參數，則須填入該功能模組所具有之屬性名稱與期望變化數值。其資料型態、數值範圍、表現行為等，皆依該屬性而定。一設定裝置屬性命令封包內須包含至少一組功能屬性參數。

單一封包內包含之功能屬性參數數量無上限，但命令整體封包容量不可超過本標準規範之上限（參照 6.3）。

7.3.3.4. 接收端執行動作

本命令為非同步執行。閘道器接收到設定裝置屬性命令時，須隨即發送一設定裝置屬性確認封包至監控伺服器。

7.3.3.5. 回覆封包內容

- 「**cmd**」欄位須填入字串「**GSET.CFM**」。
- 「**ack**」欄位須填入與命令封包「**ack**」欄位相同之數值。
- 「**result**」欄位為資料型態為「整數」，須依據執行狀況選擇一對應數值：

表 14. 設定裝置屬性回覆封包執行結果定義

數值	執行狀況	說明
100	OK	閘道器已開始執行
101	封包格式錯誤	整體 JSON 文件解析錯誤
102	主要參數格式錯誤	addr 、 cmd 、 seq 對應數值格式錯誤
103	封包承載格式錯誤	payload 欄位解析錯誤，或其下之次欄位解析錯誤

數值	執行狀況	說明
201	服務已達上限	閘道器暫時無法處理該請求
401	指定裝置並未登記	該裝置 不屬於閘道器所管轄
402	指定裝置並未上線	該裝置 並未建立場域網路連線

7.3.3.6. 後續處理

閘道器端在接收到設定裝置屬性命令後，若該命令之目標為遠端受控裝置，則閘道器隨即開始透過場域網路進行遠端控制。

當閘道器之遠端控制執行完畢時，無論結果為成功或失敗，閘道器皆須發送一屬性存取結果通知封包（參照 7.3.1）至監控伺服器端，告知監控伺服器執行結果，如下圖所示。

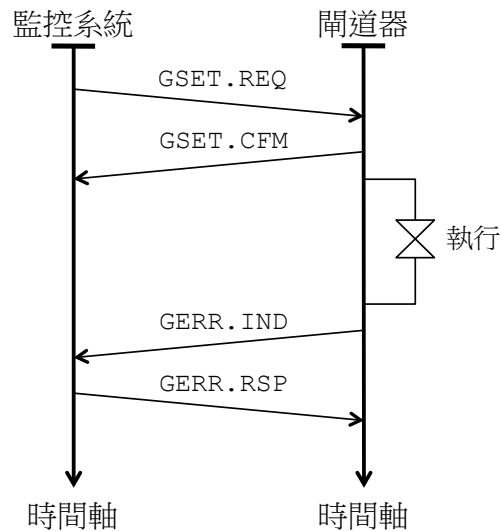


圖 15. 設定遠端受控裝置屬性之非同步執行示意圖

有關遠端控制執行，以及取得執行結果之相關實作細節，超出本文件範疇，實作應參考「智慧照明系統標準—第三部：場域網路介面規範」，以及「智慧照明系統標準—第四部：場域網路設計準則」以進行實作。

監控伺服器端須持續追蹤「**GSET.REQ**」命令之執行結果。若超過逾時時間尚未收到「**GERR.IND**」屬性存取結果通知封包，則該次屬性設定操作視為失敗。逾時時間之長度依照個別閘道器實作而定，監控伺服器可參考閘道器之「**TIMEOUT**」屬性（參照 8.2.2）作為判斷之依據。

7.3.3.7. 錯誤處理原則

監控伺服器利用「**GSET.REQ**」設定屬性時，須採用新值覆蓋舊值方式處理。

若該封包中包含部分不正確之資訊，實作可自行決定要捨棄該「**GSET.REQ**」封包所更新之所有項目（rollback transaction），或是僅捨棄錯誤的部分；但包含複合型態資料之屬性（例如陣列）不可只更新部分元素，須確定該屬性之複合資訊皆完全正確，才覆蓋原有資料。

7.3.3.8. 範例

一典型設定裝置屬性命令封包範例如下：

```
{  
  "cmd": "GSET.REQ",  
  "ack": 14235,  
  "addr": "E000090000000158",  
  "payload": {  
    "#EP": 2,  
    "LEVEL": 50  
  }  
}↵↵
```

圖 16. 設定裝置屬性命令封包範例

以上範例為設定一可調光燈具之光度（2 號服務終端）至 50%。個別功能模組屬性之定義參照第 8 節。

— 接續次頁 —

7.3.4. GUPD.IND 裝置屬性更新通知

7.3.4.1. 發送時機

裝置屬性更新通知由閘道器端主動發送至監控伺服器，用於回報受控裝置之感測數值更新、事件通知，或作為其他命令之執行結果回應。

7.3.4.2. 通知封包主要參數

- 「**addr**」欄位須填入之裝置代號（16 字元）。
- 「**cmd**」欄位須填入字串「**GSET.REQ**」。
- 「**ack**」欄位應填入閘道器端計數之發送封包序號（流水號）。

7.3.4.3. 通知封包承載內容

裝置屬性更新通知封包「**payload**」欄位之資料型態為 JSON 文件，內含兩個固定參數，以及不定數量之功能屬性參數。其中固定參數定義如下：

表 15. 裝置屬性更新通知封包承載第二階固定參數

參數名稱	資料型別	說明
"#EP"	整數	服務終端代號
"#DATE"	字串	通知封包產生時間

「**#EP**」欄位資料型態為整數，須填入指定功能模組對應之服務終端代號。

「**#DATE**」欄位須填入閘道器產生該通知封包之當地時間（local time），資料型態為字串，格式定義如下：

"	年	-	月	-	日	時	:	分	:	秒	"
---	---	---	---	---	---	---	---	---	---	---	---

圖 17. 日期字串格式定義

- 年：西元年，四字元長度。
- 月：數字 01 至 12，前置補零（固定兩字元長度）。
- 日：數字 01 至 31，前置補零（固定兩字元長度）。
- 時：數字 00 至 23，前置補零（固定兩字元長度）。
- 分：數字 00 至 59，前置補零（固定兩字元長度）。
- 秒：數字 00 至 59，前置補零（固定兩字元長度）。

其餘功能屬性參數，則須填入該功能模組所具有之屬性名稱與對應數值，資料型態依該屬性而定。一設定裝置屬性命令封包內須包含至少一組功能屬性參數，單一封包內包含之功能屬性參數數量無上限。

7.3.4.4. 接收端執行動作

監控伺服器接收到裝置屬性更新通知封包時，須隨即發送一裝置屬性更新回覆封包至監控伺服器。

7.3.4.5. 回覆封包內容

- 「**cmd**」欄位須填入字串「**GUPD.CFM**」。

- 「**ack**」欄位須填入與通知封包「**ack**」欄位相同之數值。
- 「**result**」欄位為資料型態為「整數」，須依據執行狀況，由下表選擇一對應數值。

表 16. 裝置屬性更新回覆封包執行結果定義

數值	執行狀況	說明
100	OK	接收成功
101	封包格式錯誤	整體 JSON 文件解析錯誤
102	主要參數格式錯誤	addr 、 cmd 、 seq 對應數值格式錯誤
103	封包承載格式錯誤	payload 欄位解析錯誤，或其下之次欄位解析錯誤

若發生錯誤，例如裝置未登記（錯誤代碼 **401**），或是回報之屬性與先前登記之功能模組不匹配（錯誤代碼 **300** 系列），後續處理方式由實作依據實際情況，自行定義。例如監控伺服器可要求閘道器重新連線。

7.3.4.6. 後續處理

監控伺服器接收到裝置屬性更新通知之後，後續處理方式為實作自行決定，本標準不予規範。監控伺服器宜將更新之屬性資料儲存於資料庫中，以備後續查詢，或將特定屬性視為「事件」，例如超過閾值警示，並觸發執行後續程序。

7.3.4.7. 範例

一典型裝置屬性更新通知封包範例如下：

```

{
  "cmd": "GUPD.IND",
  "ack": 14235,
  "addr": "E000090000000158",
  "payload": {
    "#EP": 2,
    "#DATE": "2012-01-01 13:56:42",
    "LEVEL": 50
  }
}

```

圖 18. 裝置屬性更新通知範例

以上範例為 7.3.3.8 中「**GSET.REQ**」範例之執行結果：可調光燈具回應最終之調光值。此一調光動作可能是源自於遠端監控，或是由現場人員直接操作燈具之實體開關。

8. 功能模組

本節描述構成受控裝置之通用功能模組，以及個別模組之屬性定義。

8.1. 功能模組代號索引

依據不同用途，功能模組分為「系統預設」、「電力計」、「感測器」、「控制器」等數個分類。實作須依照裝置之特性，選取符合該裝置功能描述之項目加以組合。

表 17. 功能模組代號索引表

功能分類	功能模組代號	名稱	對應章節
系統預設	0	泛用服務對應表	8.2.1
	11	泛用閘道器設定	8.2.2
	12	泛用閘道器排程	8.2.3
電力計	101	單相電力計	8.3.1
	102	單相雙迴路電力計	8.3.2
	103	三相電力計	8.3.3
感測器	151	泛用感測計	8.4.1
	152	泛用觸發警報器	8.4.2
	153	泛用閾值警報器	8.4.3
	154	LED 燈具狀態回報	8.4.4
控制器	201	二元開關	8.5.1
	202	定時器	8.5.2
	203	單頻道調光器	8.5.3
	204	三頻道 RGB 調光器	8.5.4

於上述制定之功能模組以外，實作可自行定義私有之功能模組，但其代號須迴避保留之代號。有關保留代號說明，參照附錄 C。

本節其餘部分將依功能模組代號字母排序，逐項介紹個別功能模組之細部定義。

— 接續次頁 —

8.2. 系統預設功能模組

本節將依功能模組代號字母排序，介紹系統預設功能模組之細節定義。

—接續次頁—

8.2.1. 0 泛用服務對應表

8.2.1.1. 用途

泛用服務對應表包含裝置之基本資訊：裝置代號、裝置資訊，以及裝置內服務終端與功能模組對應。

泛用服務對應表功能模組為所有裝置（包含閘道器）皆須實作之功能模組，且此一功能模組對應之服務終端為須固定值，即為 0 號服務終端。亦即於整體系統中，所有裝置之 0 號服務終端皆代表此一泛用服務對應表功能模組。

本模組不須加入裝置登記通知之「**CL**」陣列屬性中，亦不計入該裝置所包含之功能模組數量統計。

8.2.1.2. 屬性

表 18. 泛用服務對應表之屬性

#	屬性名稱	資料型態	支援方法	定義
1	EVT	整數	UPD	泛用事件回報
2	MODEL	字串	GET	裝置廠商、型號資訊
3	TYPE	字串	GET	傳輸媒體形式
4	STAT	整數	GET/SET	裝置重新啟動
5	CNT	整數	GET	裝置包含之功能模組數
6	CL	陣列	GET	個服務終端對應之功能模組

「**MODEL**」屬性為實作自行定義之設備版本資訊，資料型態為字串，最大長度為 20 字元，內容由實作自行定義。字串之內容須依循 JSON 標準規範。

「**TYPE**」屬性為該裝置所具備之傳輸媒體形式，資料型態為字串，最大長度為 4 字元，內容參照「智慧照明系統標準—第三部：場域網路介面規範」附錄 A。監控閘道器之 **TYPE** 屬性須填入固定值「**TCPC**」（TCP/IP client）。

「**CNT**」屬性之數值須預先填入該裝置所包含之功能模組數，但「泛用服務對應表」本身不計入計算。一裝置包含之功能模組數量由實作自行決定，但 **CNT** 屬性數值須與 **CL** 陣列內元素個數相同。

「**CL**」屬性之資料型態為陣列，陣列元素資料型態皆為整數，代表個別服務終端之功能模組代號，但「泛用服務對應表」本身不列入。若一服務終端功能模組已停用，則填入固定值「**255**」。

「**STAT**」屬性表示該裝置之「啟動」狀態。監控伺服器端可藉由寫入數值「**false**」來要求裝置重置（Reset）。

- 若目標裝置為閘道器，則閘道器須關閉與監控伺服器之連線，並嘗試重新建立連線。本項功能可用於變更閘道器連線目標。詳見 8.2.2 說明。
- 若目標裝置為其他受控裝置，則裝置應重新啟動。裝置宜採用硬體完全重置，但若裝置無法自行控制硬體完整重新啟動，則應嘗試將資料重置（Reset）、重新建立場域網路連線等。亦即執行「軟體重置」。

- 閘道器應於下達重新啟動命令後，於一定間隔時間內，主動檢視該裝置是否成功重置，並以「EVT」屬性回報監控伺服器。間隔時間可由實作依據實際狀況而定，但不可超過 10 分鐘。若裝置之硬體完全重置時間超過 10 分鐘（例如裝有大電容的電力設備），則該受控裝置應改用上述之軟體重置。
- 若裝置無法執行該項動作、或是受控裝置之場域網路不支援狀態檢視、資料回傳⁵，則閘道器應回應監控伺服器執行錯誤，錯誤代碼「405」。

「EVT」屬性為裝置之泛用事件回報。本屬性主要用於閘道器之場域網路管理，可用於通報裝置失聯、故障等狀況。相關事件可由裝置主動回報，或由閘道器檢測。資料型態為整數，數值定義為事件代號如下表所示：

表 19. 泛用事件回報事件代號

事件代號	定義
0	裝置執行正常。
404	裝置場域網路連線失效：無回應
405	裝置場域網路連線正常，但執行狀態不明
406	裝置硬體設備故障

— 接續次頁 —

⁵ 部分場域網路通訊技術，例如 DMX512-A，不支援資料回傳。此類網路之實作中，受控裝置不適合採用需回報資料之功能模組，且泛用服務對應表亦應以靜態資料表方式儲存於閘道器端。故對於該屬性之變更，實際上並無作用。詳見第四部說明。

8.2.2. 11 泛用閘道器設定

8.2.2.1. 用途

泛用閘道器設定包含閘道器之連線設定、時區資訊等。

泛用閘道器設定功能模組為所有閘道器皆須實作之功能模組。且此一功能模組須對應之服務終端為固定值，即為 1 號服務終端。亦即於整體系統中，所有閘道器之 1 號服務終端皆代表此一泛用閘道器設定模組。

本模組須加入泛用服務對應表之 **CL** 陣列中。

8.2.2.2. 屬性

表 20. 泛用閘道器設定之屬性

#	屬性名稱	資料型態	支援方法	定義
1	TMZONE	字串	GET	閘道器設備安裝地之時區
2	TIMEOUT	整數	GET	閘道器非同步執行逾時時間
3	HOST	字串	GET/SET	連線之目標網路位址
4	PORT	整數	GET/SET	連線之目標通訊埠號

「**TMZONE**」屬性代表該閘道器設備所在時區（含日光節約時間設定），須由附錄 C 時區名稱對照表中選擇一數值填入；台灣地區為「**Asia/Taipei**」。此一欄位之數值應由閘道器實作於佈建時設定。有關時區轉換，以及日光節約時間是否啟用，本標準不予規範，若監控伺服器實作須要進行時區轉換時，可利用本欄之數值作為基準。

「**TIMEOUT**」屬性代表該閘道器設備執行遠端控制時，最長的逾時時間。本數值為固定值，應由閘道器實作於佈建時設定。

「**HOST**」屬性代表監控伺服器端接受連線之網路位址，資料型態為字串，格式為 IPv4 網路位址或網域名稱（參照 5.3）。

「**PORT**」屬性與「**S.PORT**」屬性，分別代表監控伺服器端接受連線之網路通訊埠號碼，資料型態為整數。

上述四個連線屬性設定完成後，監控伺服器應將設定閘道器之泛用服務對應表之 **STAT** 屬性為 **false**，要求閘道器重新連線。

— 接續次頁 —

8.2.3. 12 泛用閘道器排程

8.2.3.1. 用途

泛用閘道器排程包含閘道器設定排程執行、情境控制等資訊。本標準之通訊協定支援以「情境」(Scene)方式執行批次控制(batch)，個別情境皆代表一預先設定之批次控制程序。上述之「批次控制程序」為閘道器端實作，由閘道器針對其下管轄之受控裝置，執行一系列預先規劃好之主動控制，以避免監控伺服器端重複傳送大量之控制命令。

由於監控伺服器可主動要求閘道器執行特定情境，為清楚辨識，個別情境皆給予一代號，稱為情境代號(Scenario ID)。但不同閘道器所定義之批次執行程序皆不相同，故情境代號與其代表之批次執行程序，不同閘道器可能有不同定義。批次執行程序之組成方式，與實際應用相關，超出本標準範疇，但本標準仍規範五種通用之預設情境：

表 21. 五種通用預設情境

情境代號	批次執行程序
101	關閉所有受控裝置之二元開關 (Switch)
102	開啟所有受控裝置之二元開關 (Switch)
103	讀取所有泛用感測器裝置之感測資料並回報
104	讀取所有電力計裝置、所有迴路之基本資料並回報
105	讀取所有電力計裝置、所有迴路之完整資料並回報

閘道器須依照實際應用，實作與其應用相關之批次執行程序，並須提供即時、定時、週期三種方式觸發批次控制程序，同時情境之觸發、排程須能由監控伺服器以遠端方式設定。

預設情境以外之控制，則由實作依據應用及實際佈建情況，自行定義情境代號，以及對應的批次控制內容。

泛用閘道器排程功能模組為所有閘道器皆須實作之功能模組，且此一功能模組須對應之服務終端為固定值，即為 2 號服務終端。亦即於整體系統中，所有閘道器之 2 號服務終端皆代表此一泛用閘道器排程模組。

本模組須加入泛用服務對應表「CL」參數之陣列中。

8.2.3.2. 泛用閘道器排程屬性

表 22. 泛用閘道器排程之屬性

#	屬性名稱	資料型態	支援方法	定義
1	CAP	整數	GET	閘道器支援之最大排程數
2	PLAY	整數	SET/GET/UPD	目前閘道器執行之情境
3	S01	字串	SET/GET	排程項目 1

#	屬性名稱	資料型態	支援方法	定義
4	S02	字串	SET/GET	排程項目 2
...
101	S99	字串	SET/GET	排程項目 99

「CAP」屬性代表該閘道器實作允許同時設定之最大排程數量。此數值應在實作時預先設定。

「PLAY」屬性代表閘道器目前執行情境之情境代號，資料格式為整數。監控伺服器可藉由設定此項屬性，要求閘道器立即執行指定情境；閘道器端在執行定時排程時，亦須同步更新此屬性數值。當該屬性更新時，無論是由監控伺服器主動更新或是排程觸發，皆須發送 GUPD.IND 封包通知監控伺服器。

「S01」至「S99」屬性代表閘道器允許設定的定時/週期排程。資料型態為字串，字串內容包含六個欄位，以空白字元（ASCII 代碼 0x20）分隔，格式如下：

"	分	時	日	月	週	情境代號	"
---	---	---	---	---	---	------	---

圖 19. 排程項目字串定義

前五個欄位（分、時、日、月、週）為採用 cron 格式之時間排程設定。cron 格式詳細說明參照附錄 B。

第六個欄位「情境代號」資料型態為整數，代表排程執行情境代號。

上述 cron 格式排程設定，須以閘道器時區設定之當地時間為主。監控伺服器須依據泛用閘道器設定之「TMZONE」屬性，處理時區轉換等相關事項。

監控伺服器利用「GSET.REQ」設定排程項目屬性時，若排程項目設定之字串格式不正確，則視為無效設定，該項目將不執行。監控伺服器亦可利用「GSET.REQ」將排程屬性設定為空字串，以「關閉」該排程項目。

8.2.3.3. 實作相關議題

以下所列事項，皆由實作依據實際設備具備之計算資源自行決定，本標準僅規範實作須滿足之最低限度。

- 閘道器允許同時設定之排程項目數量，由實作自行決定，但實作至少須提供 1 組排程項目，最多不超過 99 組。泛用服務對應表超出的排程項目屬性不須實作；例如若閘道器支援 4 組排程，則須實作排程項目屬性「S01」至「S04」即可，「S05」至「S99」不須實作。
- 閘道器之實作宜採用「週期檢查」之方式實作排程之觸發：於檢查時若發現上次未逾時，且本次逾時之項目，即為本次須執行之排程。採用這種方式將避免在執行中發生系統時鐘設定變更，而導致系統鬧鈴事件（Alarm）觸發失效。本標準並未規範排程時間觸發的精確度，週期檢查之間隔時間可依實際應用而定，宜使用「一分鐘」作為排程精確度之基本單位。

8.2.3.4. 範例

一典型設定閘道器排程之命令封包範例如下：

```
{
  "cmd": "GSET.REQ",
  "ack": 17813,
  "addr": "F026B85D006100A0",
  "payload": {
    "#EP": 2,
    "S01": "0 7 * * * 101",
    "S02": "30 18 * * * 102",
    "S02": "*/15 * * * * 103"
  }
}↵↵
```

圖 20. 設定閘道器排程命令封包範例

以上範例為設定三個排程，第一個排程是「每天早上七點關閉所有開關（預設情境 **101**）」，第二個排程是「每天下午六點開啟所有開關（預設情境 **102**）」，第三個排程是「每 15 分鐘回報所有感測器量測資料（預設情境 **103**）」。

8.3. 電力計

電力計本節將依功能模組代號字母排序，介紹電力計功能模組之細節定義。

—接續次頁—

8.3.1. 101 單相電力計

8.3.1.1. 用途

單相電力計為量測單一交流/直流迴路上電力耗用狀況之感測裝置。

本功能模組僅回報單一迴路之資訊，包括一般家庭常見的單相三線式單一迴路、一般商業場所常見的三相四線式單一迴路，以及使用單相三線式 220V 配置的單一設備（如一般家用冷氣機）。

若要偵測一般家庭單相三線式總電源，兩條火線皆須要回報數據時，實作宜使用功能模組「102 單相雙迴路電力計」。

8.3.1.2. 屬性

表 23. 單相電力計之屬性

#	屬性名稱	資料型態	支援方法	定義
1	METER	陣列	GET/UPD	基本用電資訊
2	FULL	陣列	GET/UPD	完整用電資訊
3	SAMPL	整數	SET/GET	電表量測取樣時間
4	CLR	布林	SET	重置累計資料

「**SAMP**」資料型態為整數，代表單相電力計量測之取樣時間，單位為秒。

「**CLR**」表示清除累計用電資訊，亦即「**kWh**」、「**KVAh**」等資料。監控伺服器將藉由設定本屬性為「**true**」來要求電力計清除累計用電資訊。

「**METER**」屬性資料型態為有序陣列，代表單相電力計量測所得之基本用電資訊。本屬性可由監控伺服器端以「**GGET.REQ**」取得裝置屬性方式擷取，或由閘道器主動擷取並回報。陣列內容定義如下：

表 24. **METER** 屬性（基本用電資料）陣列內容

元素	資料型態	定義	單位
1 ⁽⁶⁾	實數	電壓	V
2	實數	電流	A
3	實數	功率因數 (0~1)	—
4	實數	實功率	W
5	實數	總電能 (瓦時)	kWh

「**FULL**」資料型態為有序陣列，代表單相電力計量測之完整用電資訊。本屬性可由監控伺服器端以「**GGET.REQ**」取得裝置屬性方式擷取，或由閘道器主動擷取並回報。陣列內容定義如下：

⁶ 以 C/C++/Java 語言表示，元素 1 即為 array[0]。

表 25. **FULL** 屬性（完整用電資料）陣列內容

元素	資料型態	定義	單位
1	實數	電壓	V
2	實數	電流	A
3	實數	功率因數（0~1）	—
4	實數	實功率	W
5	實數	總電能（度數）	kWh
6	實數	短期總電能	kWh
7	實數	視在功率	VA
8	實數	虛功率	VAR
9	實數	視在電能	kVAh
10	實數	無功電能	kVARh
11	實數	頻率	Hz
12	實數	用電時間	小時

8.3.1.3. 實作相關議題

所有包含單相電力計功能模組的裝置皆須支援上述三個屬性，若設備本身未提供部分用電資訊，則於「**METER**」或「**FULL**」回報時，陣列內對應（無法提供資訊）之項目須填入「0」。

「**SMPL**」與「**CLR**」兩項屬性之寫入操作為選擇性實作，若設備端不支援該項操作，則對該寫入操作可直接忽視，不須回報錯誤訊息。

8.3.1.4. 範例

一典型單相電力計回報之封包範例如下：

```

{
  "cmd": "GUPD.IND",
  "ack": 17531,
  "addr": "B000010000000031",
  "payload": {
    "#EP": 2,
    "#DATE": "2012-01-01 13:52:26,Asia/Taipei",
    "METER": [115.2, 1.3, 144.5, 0.95, 104.1]
  }
}

```

圖 21. 單相電力計回報封包範例

以上範例回報基本用電資訊為：115.2V，1.3A，144.5W，功率因數 95%，累計用電 104.1 度 (kWh)。

—接續次頁—

8.3.2. 102 單相雙迴路電力計

8.3.2.1. 用途

單相雙迴路電力計之功能為量測單相三線式電源配置之總電源（亦即兩條火線），回報數據包括總用電資訊，以及兩條主迴路個別之用電資訊。

8.3.2.2. 屬性

表 26. 單相雙迴路電力計之屬性

#	屬性名稱	資料型態	支援方法	定義
1	METER	陣列	GET/UPD	基本用電資訊（總迴路）
2	FULL	陣列	GET/UPD	完整用電資訊（總迴路）
3	METER.A	陣列	GET/UPD	基本用電資訊（火線 A）
4	FULL.A	陣列	GET/UPD	完整用電資訊（火線 A）
5	METER.B	陣列	GET/UPD	基本用電資訊（火線 B）
6	FULL.B	陣列	GET/UPD	完整用電資訊（火線 B）
7	SMPL	整數	SET/GET	電表量測取樣時間（單位秒）
8	CLR	布林	SET	重置累計資料

「**METER**」屬性與「**FULL**」屬性代表兩條火線加總之用電資訊，資料型態及內容格式定義與單相電力計相同（參照 8.3.1.2）。

「**METER.A**」與「**METER.B**」屬性，分別代表第一條火線與第二條火線之基本用電資訊，資料型態及內容格式定義與「**METER**」屬性完全相同。

「**FULL.A**」與「**FULL.B**」屬性，分別代表第一條火線與第二條火線之完整用電資訊，資料型態及內容格式與「**FULL**」屬性完全相同。

「**SAMP**」資料型態為整數，代表電力計量測之取樣時間，單位為秒。

「**CLR**」表示清除累計用電資訊，亦即 kWh、kVAh 等資料。監控伺服器將藉由設定本屬性為「**true**」來要求電力計清除累計用電資訊。

8.3.2.3. 實作相關議題

所有包含單相雙迴路電力計功能模組的裝置皆須支援上述八個屬性，若本身未提供部分用電資訊，則於「**METER**」、「**FULL**」回報時，陣列內對應（無法提供資訊）之項目須填入「0」。

「**SMPL**」與「**CLR**」兩項屬性之寫入操作為選擇性實作，若設備端不支援該項操作，則對該寫入操作可直接忽視，不須回報錯誤訊息。

8.3.2.4. 範例

一典型單相雙迴路電力計回報之封包範例如下：

```
{
  "cmd": "GUPD.IND",
```

```
"ack":1436,  
"addr":"A000030000000055",  
"payload":{  
  "#EP":1,  
  "#DATE":"2012-01-01 14:07:42,Asia/Taipei",  
  "METER.A":[115.24,1.32,144.5,0.95,104.1]  
  "METER.B":[115.24,1.14,126.1,0.97,90.8]  
  "METER":[115.24,2.46,270.6,0.96,194.9]  
}  
}↵↵
```

圖 22. 單相雙迴路電力計回報封包範例

8.3.3. 103 三相電力計

8.3.3.1. 用途

單相雙迴路電力計之功能為量測三相四線式電源配置之總電源（亦即三條火線），回報數據包括總用電資訊，以及個別相位迴路之用電資訊。

8.3.3.2. 屬性

表 27. 三相電力計之屬性

#	屬性名稱	資料型態	支援方法	定義
1	METER	陣列	GET/UPD	基本用電資訊（總迴路）
2	FULL	陣列	GET/UPD	完整用電資訊（總迴路）
3	METER.R	陣列	GET/UPD	基本用電資訊（R 相）
4	FULL.R	陣列	GET/UPD	完整用電資訊（R 相）
5	METER.S	陣列	GET/UPD	基本用電資訊（S 相）
6	FULL.S	陣列	GET/UPD	完整用電資訊（S 相）
7	METER.T	陣列	GET/UPD	基本用電資訊（T 相）
8	FULL.T	陣列	GET/UPD	完整用電資訊（T 相）
9	SMPL	整數	SET/GET	電表量測取樣時間
10	CLR	布林	SET	重置累計資料

「**METER**」屬性與「**FULL**」屬性代表三個迴路加總之用電資訊，資料型態及內容格式定義與單相電力計相同（參照 8.3.1.2）。

「**METER.R**」、「**METER.S**」與「**METER.T**」屬性，代表個別相位迴路之基本用電資訊，資料型態及內容格式定義與「**METER**」屬性完全相同。

「**FULL.R**」、「**FULL.B**」與「**FULL.B**」屬性，分別代表個別相位迴路之完整用電資訊，資料型態及內容格式與「**FULL**」屬性完全相同。

「**SAMP**」資料型態為整數，代表電力計量測之取樣時間，單位為秒。

「**CLR**」表示清除累計用電資訊，亦即 kWh、kVAh 等資料。監控伺服器將藉由設定本屬性為「**true**」來要求電力計清除累計用電資訊。

8.3.3.3. 實作相關議題

所有包含三相電力計功能模組之裝置皆須支援上述十個屬性，若設備本身未提供部分用電資訊，則於「**METER**」、「**FULL**」回報時，陣列內對應（無法提供資訊）之項目須填入固定值「0」。

「**SMPL**」與「**CLR**」兩項屬性之寫入操作為選擇性實作，若設備端不支援該項操作，則對該寫入操作可直接忽視，不須回報錯誤訊息。

8.3.3.4. 範例

一典型三相電力計回報之封包範例如下：

```
{
  "cmd": "GUPD.IND",
  "ack": 7216,
  "addr": "B000010000000031",
  "payload": {
    "#EP": 1,
    "#DATE": "2012-01-01 15:17:23,Asia/Taipei",
    "METER.R": [115.24,1.32,144.5,0.95,104.1]
    "METER.S": [115.24,1.14,126.1,0.97,90.8]
    "METER.T": [115.24,1.44,154.3,0.94,111]
    "METER": [115.24,3.9,424.9,0.95,305.9]
  }
}□□
```

圖 23. 三相電力計回報封包範例

8.4. 感測器

本節將依功能模組代號字母排序，介紹感測器功能模組之細節定義。

—接續次頁—

8.4.1. 151 泛用感測器

8.4.1.1. 用途

泛用感測器功能模組之功能為配合傳感器（Transducer）設備，擷取物理量讀值，轉化為數據資料並回傳。本功能模組之設計為主動回報（參照 7.3.4），回報封包內屬性種類與數量由實作依據該受控裝置之功能性質而定。

8.4.1.2. 屬性

表 28. 泛用感測器之屬性

#	屬性名稱	資料型態	支援方法	定義	單位
1	SAMP	整數	SET	感測器取樣週期	毫秒
2	FREQ	整數	SET	裝置量測週期	秒
3	READ	整數	GET	資料回報	—
4	TEMP	陣列	UPD	溫度 (-45 ~ +150)	°C
5	HUMD	陣列	UPD	濕度 (0 ~ 100)	%
6	LUX	陣列	UPD	照度 (0 ~ 20000)	lux (=lm/m ²)
7	IRAD	陣列	UPD	日照強度 (0 ~ 6000)	W/m ²
8	INSO	陣列	UPD	累積日射量 (0 ~ 600)	MJ/m ²
9	CO2	陣列	UPD	二氧化碳濃度 (0 ~ 2000)	ppm
10	CO	陣列	UPD	一氧化碳濃度 (0 ~ 2000)	ppm

「**SAMP**」屬性表示感測模組擷取物理量的取樣週期。範圍為 0 至 65535，單位為毫秒（ms）。本項屬性設定功能為選擇性實作，若受控裝置不支援取樣週期設定，對本屬性進行設定之操作可忽略，不須回報錯誤。

「**FREQ**」屬性表示泛用感測器裝置更新參數表內量測讀值的更新週期。範圍為 0 至 65535，單位為秒。本項屬性設定功能為選擇性實作，若受控裝置不支援更新週期設定，對本屬性進行設定之操作可忽略，不須回報錯誤。

「**READ**」屬性資料型態為整數，數值表示該裝置具備的感測資料數量。當受控裝置端接收到 **READ** 屬性讀取命令時，即須回報所有感測讀值。亦即監控伺服器將藉由讀取 **READ** 屬性，強制裝置回報感測資料。

其餘屬性之資料型態皆為有序陣列（Ordered），陣列元素之資料型態皆為實數，個

別元素皆代表一感測點。陣列大小（元素數量）與內容元素順序之定義，由實作自行決定，例如第一個元素表示外部環境氣溫、第二個元素表示設備外殼溫度、第三個元素表示電源供應器溫度等。

若量測數值超出定義範圍，則表示受控裝置之傳感器量測物理量超過容許範圍，或是該傳感器已停用。

8.4.1.3. 範例

一典型泛用感測器回報之封包範例如下：

```
{
  "cmd": "GUPD.IND",
  "ack": 7216,
  "addr": "D000030000000044",
  "payload": {
    "#EP": 1,
    "#DATE": "2012-01-01 19:27:03,Asia/Taipei",
    "LUX": [125.1],
    "TEMP": [18.5,20.1,36.4]
  }
}↵↵
```

圖 24. 泛用感測器回報封包範例

— 接續次頁 —

8.4.2. 152 泛用觸發警報器

8.4.2.1. 用途

泛用觸發警報器適用於僅具備單純事件觸發之警報器設備。具有傳感器，且允許設定閾值進行比對之警報器設備，宜使用「泛用閾值警報器」。

8.4.2.2. 屬性

表 29. 泛用觸發警報器之屬性

#	屬性名稱	資料型態	支援方法	定義
1	TYPE	整數	GET/UPD	警報器種類
2	STAT	布林	GET	警報器狀態
3	COUNT	整數	GET/UPD	警報器觸發累計次數
4	CONFIG	陣列	SET	觸發警報設定

「**TYPE**」屬性資料型態為整數，此數值為常數，須由實作預設填入，定義如下：

表 30. 泛用觸發警報事件

事件代號	定義
1	人工操作（緊急按鈕等）
2	箱體開啟
3	雷擊警示
4	傳感器失效
5	物體移動（紅外線、微波）
6	迴路切斷（入侵）
7	設備斷電警示
8	煙霧、火災警示
9	一氧化碳警示
10	水位警示
11	設備移動/墜落警示
255	其他狀況

「**STAT**」參數表示在讀取的時間點，警報器是否為觸發狀態。「**false**」表示未觸發，「**true**」表示觸發。

「**COUNT**」屬性資料型態為整數，表示該警報器功能模組觸發累計次數。

「**CONFIG**」屬性資料型態為有序陣列，作為監控伺服器遠端設定。陣列內容定義如下：

表 31. **CONFIG** 屬性（觸發警報）陣列內容

元素	資料型態	定義
1 ⁽⁷⁾	布林	警示啟動/關閉
2	整數	警示間隔
3	整數	警示持續時間

第一個元素資料型態為布林，表示是否啟動警報器觸發功能。「**false**」表示關閉，「**true**」表示啟動。受控裝置須完整實作本屬性之功能。

第二個元素資料型態為整數，表示兩次警示觸發之間的最短間隔時間，範圍為 1 至 255，單位為秒。本屬性為選擇性實作，若受控裝置不支援觸發間隔時間設定，對本屬性進行設定之操作可忽略，不須回報錯誤。

第三個元素資料型態為整數，表示警報器於觸發時，維持警報狀態的持續時間。範圍為 0（只觸發一次）至 65535，單位為秒。本屬性為選擇性實作，若受控裝置不支援警示持續時間設定，對本屬性進行設定之操作可忽略，不須回報錯誤。

上述「持續時間」設定，主要是用於受控裝置之附屬「蜂鳴器、閃燈」等實體警示裝置。若實作受控裝置支援主動回報功能，亦可作為主動回報之頻率設定。

— 接續次頁 —

⁷ 以 C/C++/Java 語言表示，元素 1 即為 array[0]。

8.4.3. 153 泛用閥值警報器

8.4.3.1. 用途

泛用閥值警報器允許設定單一閥值範圍，適用於單純監測環境監控之場合，例如設備工作溫度、室內二氧化碳濃度等。

8.4.3.2. 屬性

表 32. 泛用閥值警報器之屬性

#	屬性名稱	資料型態	支援方法	定義
1	TYPE	整數	GET/UPD	警報器種類
2	STAT	布林	GET	警報器狀態
3	ALARM	實數	GET/UPD	目前物理量量測值
4	THRES	陣列	SET	閥值範圍
5	CONFIG	陣列	SET	閥值警報設定

「**TYPE**」屬性資料型態為字串，代表該警報器的物理量種類，內容須從下表中選擇：

表 33. 泛用閥值警報器物理量定義

TYPE	定義	閥值範圍	單位/精確度
"TEMP"	溫度	-45.0 ~ +150.0	0.1 °C
"HUMD"	濕度	0.0 ~ 100.0	0.1 %
"LUX"	照度 (Illuminance)	0 ~ 20000	1 Lux
"IRAD"	日照強度 (Irradiance)	0 ~ 6000	1 W/m ²
"CO2"	二氧化碳濃度	0 ~ 2000	1 ppm
"CO"	一氧化碳濃度	0 ~ 2000	1 ppm

「**STAT**」屬性表示在讀取的時間點，警報器是否為觸發狀態。「**false**」表示未觸發，「**true**」表示觸發。

「**ALARM**」屬性資料型態為實數，表示受控裝置目前物理量量測值。數值單位與定義依據感測器種類不同而異，參照表 33。若受控裝置不支援物理量回傳，則該屬性數值須填入固定值「0」。

「**CONFIG**」屬性資料型態為有序陣列，作為監控伺服器遠端設定。陣列內容定義如下：

表 34. CONFIG 屬性（閾值警報設定）陣列內容

元素	資料型態	定義
1 ⁽⁸⁾	布林	警示啟動/關閉
2	整數	警示間隔
3	整數	警示持續時間

第一個元素資料型態為布林，表示是否啟動警報器觸發功能。「**false**」表示關閉，「**true**」表示啟動。受控裝置須完整實作本屬性之功能。

第二個元素資料型態為整數，表示兩次警示觸發之間的最短間隔時間，範圍為 1 至 255，單位為秒。本屬性為選擇性實作，若受控裝置不支援觸發間隔時間設定，對本屬性進行設定之操作可忽略，不須回報錯誤。

第三個元素資料型態為整數，表示警報器於觸發時，維持警報狀態的持續時間。範圍為 0（只觸發一次）至 65535，單位為秒。本屬性為選擇性實作，若受控裝置不支援本項設定，對本屬性進行設定之操作可忽略，不須回報錯誤。

「**THRES**」屬性資料型態為陣列，內容包含兩個元素，代表物理量之閾值設定。第一個元素為下限閾值，感測器讀取數值低於該數值時，則觸發警報。第二個元素為上限閾值，感測器讀取數值超過該數值時，則觸發警報。兩個元素的資料型態皆為實數，數值單位與定義依據感測器種類不同而異，參照表 33。

若閾值下限/上限兩個數值為獨立設定，觸發判定也是獨立判定。閾值下限之數值可設定為比上限數值高，代表「進入上下限範圍時觸發警報」。

若閾值上限/下限設定之數值超過允許範圍，則視為關閉上限/下限檢查。

8.4.3.3. 範例

```

{
  "cmd": "GSET.REQ",
  "ack": 251,
  "addr": "D000030000000074",
  "payload": {
    "#EP": 1,
    "CONFIG": [ true, 10, 60 ], "THRES": [-50, 45]
  }
}

```

圖 25. 泛用閾值警報器設定封包範例

上圖為一典型閾值警報器設定封包。該範例為一設備工作溫度警示器；若量測值持續超過閾值範圍，每 10 秒發出一警報，最久持續 60 秒。閾值設定上限 45 度，下限不檢測。

⁸ 以 C/C++/Java 語言表示，元素 1 即為 array[0]。

泛用閥值警報器之通報封包範例如下：

```
{
  "cmd": "GUPD.IND",
  "ack": 1216,
  "addr": "D000030000000074",
  "payload": {
    "#EP": 1,
    "#DATE": "2012-01-01 09:17:03,Asia/Taipei",
    "TYPE": "TEMP", "ALARM": 48.5
  }
}↵↵
```

圖 26. 泛用閥值警報器通報封包範例

該範例為一設備工作溫度警示器，原訂閥值為上限 45 度，發生警報事件時讀值為 48.5 度。

8.4.4. 154 LED 燈具狀態回報

8.4.4.1. 用途

LED 燈具狀態回報功能模組適用於具有多個獨立燈條的 LED 照明設備，可用於回報設備目前的執行狀態。

8.4.4.2. 屬性

表 35. LED 燈具狀態回報之屬性

#	屬性名稱	資料型態	支援方法	定義
1	ACCUM	實數	GET	照明設備累計使用時間
2	UNITS	整數	GET	照明設備內含燈條數量
3	HEALTH	整數	GET/UPD	目前設備健康度
4	THRES	整數	GET/SET	故障警示閾值
5	CONFIG	陣列	SET	警報設定

「**ACCUM**」屬性代表 LED 燈具累計點亮時間，單位為小時。本屬性為選擇性實作，若裝置不支援計時，則回傳固定值「0」。

「**UNITS**」屬性代表燈具包含的燈條數量（或 LED 單元數量），此數值為常數，由實作於裝置製造時指定。

「**HEALTH**」屬性代表燈具目前可正常運作的元件之比例，數值單位為百分比。100 表示完全正常（100%），0 表示完全損壞（0%）。比例計算方式由裝置自行定義。

「**THRES**」屬性代表燈具判定為「故障」的健康度閾值。若閾值低於此一數值，則受控裝置應主動發出警示。單位為百分比，比例計算方式由裝置定義。

「**CONFIG**」屬性資料型態為有序陣列，作為監控伺服器遠端設定。陣列內容定義如下：

表 36. **CONFIG** 屬性（LED 燈具狀態回報）陣列內容

元素	資料型態	定義
1 ⁽⁹⁾	布林	警示啟動/關閉
2	整數	警示間隔
3	整數	警示持續時間

第一個元素資料型態為布林，表示是否啟動警報器觸發功能。「**false**」表示關閉，「**true**」表示啟動。受控裝置須完整實作本屬性之功能。

第二個元素資料型態為整數，表示兩次警示觸發之間的最短間隔時間，範圍為 1 至 255，單位為秒。本屬性為選擇性實作，若受控裝置不支援觸發間隔時間設定，對

⁹ 以 C/C++/Java 語言表示，元素 1 即為 array[0]。

本屬性進行設定之操作可忽略，不須回報錯誤。

第三個元素資料型態為整數，表示警報器於觸發時，維持警報狀態的持續時間。範圍為 0（只觸發一次）至 65535，單位為秒。本屬性為選擇性實作，若受控裝置不支援本項設定，對本屬性進行設定之操作可忽略，不須回報錯誤。

8.4.4.3. 範例

```
{
  "cmd": "GUPD.IND",
  "ack": 1521,
  "addr": "D000030000000096",
  "payload": {
    "#EP": 1,
    "#DATE": "2012-01-01 19:48:31,Asia/Taipei",
    "CUMULATE": 3021.5,
    "UNITS": 4,
    "HEALTH": 75
  }
}
}↵↵
```

圖 27. LED 燈具狀態回報封包範例

上圖為一典型 LED 燈具狀態回報封包。上圖範例中，該裝置累計點亮時間為 3021.5 小時，內含四個燈條，於讀取時有四分之一燈條故障。

8.5. 控制器

本節將依功能模組代號字母排序，介紹控制器功能模組之細節定義。

—接續次頁—

8.5.1. 201 二元開關

8.5.1.1. 用途

二元開關為具備 ON/OFF 兩個型態的開關控制元件，可為純遠端控制，或是混用實體手動控制開關。

8.5.1.2. 屬性

表 37. 二元開關之屬性

#	屬性名稱	資料型態	支援方法	定義
1	TYPE	整數	GET	開關種類
2	SWITCH	布林	GET /UPD	開關狀態
3	SET	字串	SET	控制開關

「**SWITCH**」屬性資料型態為布林，「**true**」代表開關開啟，「**false**」代表開關關閉。

「**SET**」屬性資料型態為字串，字串內容對應控制方式如下表：

表 38. 二元開關控制方法

" SET "	定義
" ON "	開啟
" OFF "	關閉
" TOGGLE "	切換（原本關閉則開啟，原本開啟則關閉）

「**TYPE**」屬性表示該開關設備之型態，本欄位數值為常數，須由受控裝置端實作預設填入。數值定義如下表：

表 39. 二元開關種類

" TYPE "	定義
1	撥動式（Flip）開關
2	切換式（Toggle）開關
3	邏輯開關（由遠端設定之程式旗標）

若該裝置具備手動控制之實體開關，或是與裝置內其他功能模組連動控制（例如定時器），於開關狀態變更時，受控裝置須同步更新裝置端「**SWITCH**」屬性數值。

若受控裝置因上述原因（手動控制、連動控制）而改變開關狀態時，可將此一事件（Event）回傳至監控伺服器。此一功能為實作依據應用情境選擇性實作，若受控

裝置本身使用之場域網路傳輸媒體不具備狀態回報之功能，或是其應用情境不允許（例如會由人工進行頻繁切換），則主動回報之功能亦可選擇不實作。

—接續次頁—

8.5.2. 202 定時器

8.5.2.1. 用途

定時器是具備定時、定期控制的二元開關。

定時器功能模組本身僅提供時間設定，不包含開關狀態控制與回報。若實作必要由遠端控制開關狀態，或是主動回報開關事件，實作宜在受控裝置中加入二元開關功能模組（參見 8.5.1），並設定與定時器模組連動控制。二元開關功能模組即能提供遠端監控功能。

8.5.2.2. 屬性

表 40. 定時器之屬性

#	屬性名稱	資料型態	支援方法	定義
1	CAP	整數	GET	最大允許設定數量
2	SYNC	布林	SET	時間校正
3	TIMER	陣列	GET/SET	定時設定

「**CAP**」屬性代表裝置最大允許設定之開啟/關閉排程之總數，此數值為常數，由裝置實作定義。

「**SYNC**」屬性對該裝置進行校時。當閘道器接收到對此一參數寫入時，無論寫入數值為何（**true** 或 **false**），皆需對目標受控裝置進行校時。具體校時方法係由閘道器使用本地端時間，對受控裝置定時器功能模組的「**CLOCK**」參數執行寫入，詳細資訊參照「智慧照明系統標準—第三部：場域網路介面規範」第 8.2.9 小節。採用此項設計，而不開放由伺服器直接設定數值的原因，係為避免閘道器以佇列方式同時處理多項請求時，造成額外的延遲。亦即閘道器應在送出更新命令之前，再填入正確數值。

「**TIMER**」屬性代表裝置設定啟動或關閉的時間，屬性資料型態為陣列，內容元素資料型態為字串，格式定義如下：

"	小時	:	分鐘	:	開關	"
---	----	---	----	---	----	---

圖 28. 定時設定格式定義

其中「小時」與「分鐘」定義與上述時間字串格式相同，「開關」次欄位內容須填入字串「**ON**」或「**OFF**」，代表開關的開啟或關閉。

8.5.2.3. 範例

定時器設定範例如下：

```
{
  "cmd": "GSET.REQ",
  "ack": 1376,
```

```
"addr": "D000030000000048",  
"payload": {  
  "#EP": 1,  
  "SYNC": true,  
  "TIMER": [  
    "09:00:ON", "13:30:ON", "19:30:ON",  
    "12:30:OFF", "18:30:OFF", "21:00:OFF"  
  ]  
}  
}↵↵
```

圖 29. 定時器設定範例

以上範例設定定時器在每天早上九點開啟、十二點半關閉；下午一點半開啟、下午六點半關閉；晚上七點半開啟、晚上九點半關閉。

— 接續次頁 —

8.5.3. 203 單頻道調光器

8.5.3.1. 用途

單頻道調光器接受遠端控制、調整強度輸出，範圍為 0 至 100%。

8.5.3.2. 屬性

表 41. 單頻道調光器之屬性

#	屬性名稱	資料型態	支援方法	定義
1	TYPE	整數	GET	設備種類
2	LEVEL	整數	GET/SET/UPD	調光強度

「**LEVEL**」屬性資料型態為整數，範圍為 0 至 100，0 代表最弱，100 代表最強。

「**TYPE**」屬性表示該設備之種類，本欄位數值為常數，須由受控裝置端實作預設填入。數值定義如下表：

表 42. 單頻道「調光器」設備種類

"TYPE"	定義
1	可調光燈具
11	邏輯參數（由遠端設定之程式變數）
12	電壓輸出
13	電流輸出
14	頻率
15	時間控制
41	音響設備（主音量）
42	音響設備（高頻音量）
43	音響設備（低頻音量）
51	馬達（轉速）
61	轉軸（水平角度）
62	轉軸（垂直角度）
63	轉軸（斜向軸角度）
71	液壓（水平縱向長度）
72	液壓（垂直長度）
73	液壓（水平橫向長度）
84	液壓（斜向軸長度）

"TYPE"	定義
81	通道（拉門、卷門）
82	通道（窗扇、卷簾）
83	通道（銀幕、布幕）
91	閥門（水閘）
92	閥門（蒸汽）
103	多元開關：三種狀態
104	多元開關：四種狀態
105	多元開關：五種狀態
106~	多元開關：六~九種狀態
141~255	實作自行定義

若該裝置具備手動控制之實體開關，或是與裝置內其他功能模組連動控制（例如定時器），於實際調光強度數值變更時，受控裝置須同步「**LEVEL**」屬性之數值。

若受控裝置因上述原因（手動控制、連動控制）而改變調光強度時，可將此一事件（Event）回傳至監控伺服器。此一功能為實作依據應用情境選擇性實作，若受控裝置本身使用之場域網路傳輸媒體不具備狀態回報之功能，或是其應用情境不允許（例如會由人工進行頻繁操作），則主動回報之功能亦可選擇不實作。

— 接續次頁 —

8.5.4. 204 雙頻道調光器

8.5.4.1. 用途

雙頻道調光器具有兩個控制器，獨立控制強度輸出 0 至 100%。

8.5.4.2. 屬性

表 43. 雙頻道 RGB 調光器之屬性

#	屬性名稱	資料型態	支援方法	定義
1	TYPE	整數	GET	設備種類
2	LEVEL	陣列	GET/SET/UPD	調光強度

「**LEVEL**」屬性資料型態為有序陣列，內容元素資料格式皆為整數，範圍為 0 至 100。個別元素定義如下表，其中個別元素之數值以 0 代表最弱，100 代表最強。

表 44. **LEVEL** 屬性陣列內容

元素	資料型態	定義
1 ⁽¹⁰⁾	整數	頻道 1 強度
2	整數	頻道 2 強度

「**TYPE**」屬性表示該設備之種類，本欄位數值為常數，須由受控裝置端實作預設填入。數值定義如下表：

表 45. 雙頻道「調光器」設備種類

TYPE	定義
2	雙頻道可調光燈具
4	可調色溫燈具
21	邏輯參數（遠端設定之程式變數）
22	雙頻道電壓輸出
23	雙頻道電流輸出
24	雙頻道頻率控制
25	雙頻道時間控制
44	音響設備（高頻、低頻）
50	馬達（轉速）
60	轉軸（角度）

¹⁰ 以 C/C++/Java 語言表示，元素 1 即為 array[0]。

TYPE	定義
70	液壓（長度）
80	通道
90	閥門
113	多元開關：三種狀態
114	多元開關：四種狀態
115	多元開關：五種狀態
116~	多元開關：六~九種狀態
131~255	實作自行定義

若該裝置具備手動控制之實體開關，或是與裝置內其他功能模組連動控制（例如定時器），於實際調光強度值變更時，受控裝置須同步更新「**LEVEL**」屬性數值。

若受控裝置因上述原因（手動控制、連動控制）而改變調光強度時，可將此一事件（Event）回傳至監控伺服器。此一功能為實作依據應用情境選擇性實作，若受控裝置本身使用之場域網路傳輸媒體不具備狀態回報之功能，或是其應用情境不允許（例如會由現場進行頻繁操作），則主動回報之功能亦可選擇不實作。

— 接續次頁 —

8.5.5. 205 三頻道調光器

8.5.5.1. 用途

三頻道調光器具有三個控制器，獨立控制強度輸出 0 至 100%。

8.5.5.2. 屬性

表 46. 三頻道 RGB 調光器之屬性

#	屬性名稱	資料型態	支援方法	定義
1	TYPE	整數	GET	設備種類
2	LEVEL	陣列	GET/SET/UPD	調光強度

「**LEVEL**」屬性資料型態為有序陣列，內容元素資料格式皆為整數，範圍為 0 至 100。個別元素定義如下表，其中個別元素之數值以 0 代表最弱，100 代表最強。

表 47. **LEVEL** 屬性陣列內容

元素	資料型態	定義
1 ⁽¹¹⁾	整數	頻道 1 強度
2	整數	頻道 2 強度
3	整數	頻道 3 強度

「**TYPE**」屬性表示該設備之種類，本欄位數值為常數，須由受控裝置端實作預設填入。數值定義如下表：

表 48. 三頻道「調光器」設備種類

TYPE	定義
3	可調光 RGB 燈具（依序為：紅、藍、綠）
4	可調色溫燈具
6	探照燈（依序為：水平角度、仰角、光強度）
7	旋轉燈（依序為：水平轉速、仰角轉速、光強度）
31	邏輯參數（遠端設定之程式變數）
32	三頻道電壓輸出
33	三頻道電流輸出
34	三頻道頻率控制
35	三頻道時間控制

¹¹ 以 C/C++/Java 語言表示，元素 1 即為 array[0]。

TYPE	定義
40	音響設備
50	馬達（轉速）
60	轉軸（角度）
70	液壓（長度）
80	通道
90	閥門
123	多元開關：三種狀態
124	多元開關：四種狀態
125	多元開關：五種狀態
126~	多元開關：六~九種狀態
131~255	實作自行定義

若該裝置具備手動控制之實體開關，或是與裝置內其他功能模組連動控制（例如定時器），於實際調光強度值變更時，受控裝置須同步更新「**LEVEL**」屬性數值。

若受控裝置因上述原因（手動控制、連動控制）而改變調光強度時，可將此一事件（Event）回傳至監控伺服器。此一功能為實作依據應用情境選擇性實作，若受控裝置本身使用之場域網路傳輸媒體不具備狀態回報之功能，或是其應用情境不允許（例如會由現場進行頻繁操作），則主動回報之功能亦可選擇不實作。

—以下空白—

本頁保持空白

附錄A JSON 文件格式概述

完整 JSON 格式之規範參照網站 (<http://www.json.org/>) 之文件說明。本節將簡單介紹本標準所使用之 JSON 基本格式。

依據 JSON 規範，封包內容為 ASCII 可列印碼 (Printable) 所構成，包含標點符號、換行、空白 (Space) 與縮排 (Tab) 字元。所有內部記載之字串、整數、小數等表示皆以一般人可閱讀之書面表示方式編排。

舉例：字串「-12.3」於電腦內部儲存與網路傳輸時，係以純二進位數值之 0x2D、0x31、0x32、0x2E、0x33 標記上述五個字符 (Symbol)，但於系統解析時則須轉換為「-12.3」實數 (浮點數) 數值。本文件內所有「封包內容」之相關描述，皆依循相同表示方式，後續不再贅述。

本標準內所使用封包，內含一完整 JSON 文件 (JSON Object)，亦即以左右大括號 (字符「{ }」) 標記範圍之參數列表。如下圖所示：



圖 A.1. JSON 文件 (Object) 結構圖

其中「參數列表」包含數個「參數名稱/數值組合」(Key/Value pair)，各參數組合之間以字符「,」(逗號) 分隔。一 JSON 文件內包含之參數組合數量未限定，且參數組合於解析時不分排列順序。

參數組合之前後可加入任意空白字元，含空白 (Space)、縮排 (Tab) 與換行，但不可連續使用兩個換行字元⁽¹²⁾。

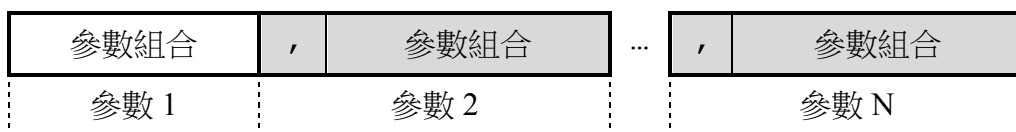


圖 A.2. 參數列表定義

一參數組合包含一「參數名稱」與一「數值」，中間以字符「:」分隔，其中參數名稱為一字串。

於本標準內，所有參數之參數名稱皆須依循規範。若實作於本規範範圍之外，自行定義額外支援參數，則參數名稱須使用數字、英文字母 (大小寫不同)、底線符號 (字符「_」)、錢號 (字符「\$」)、連結號 (字符「~」) 等字符構成，不可使用其他字符，參數名稱內也不可有空白字符 (space、tab、換行)。

¹² 連續兩個換行字元於本標準中標記為文件結束，參照 6.3。

"	參數名稱 (Key)	"	:	數值 (Value)
---	------------	---	---	------------

圖 A.3. 參數組合格式 (Key/Value Pair)

「數值」欄位可填入單純數值型態或複合數值型態。其中單純數值型態包含字串、整數、帶小數的實數、布林二元值 (boolean) 以及空值 (null)：

- 整數、實數：以十進位表示，無後置補零，無千位數標記，小數點記號為字符「.」，負數標記前置符號為字符「-」。
- 字串：可列印字元，含與空白字符 (space)，前後以雙引號 (字符「"」) 標記。空白字符 (space) 以外的脫溢字元 (escape character) 須以前置反斜線符號標記：
 - ◆ 雙引號：標記為「\"」。
 - ◆ 反斜線：標記為「\\」。
 - ◆ 斜線：標記為「\/」。
 - ◆ 換行 (LF)：標記為「\n」。
 - ◆ 換行 (CR)：標記為「\r」。
 - ◆ 縮排 (Tab)：標記為「\t」。
 - ◆ Unicode 字元：標記為「\u****」，其中****為十六進位數值。
- 布林值：不帶引號的「true」或「false」字串
- 空值：不帶引號的「null」字串

表 A.1. 單純數值表示方式範例

參數種類	範例
整數	108025
	-3
實數	110.25
	-2.6
字串	"type:\t3-phase meter"
	"2012/01/01 09:35:20"
	"02603A519F04"
空字串	""
布林	true
	false
空值	null

上述之「數值」欄位亦可填入複合型態資料，包括陣列 (Array)，或是另一個完整 JSON 文件 (JSON Object，亦即另一以字符「{」包圍之參數列表)。其中「陣列」之表示方式定義如下：

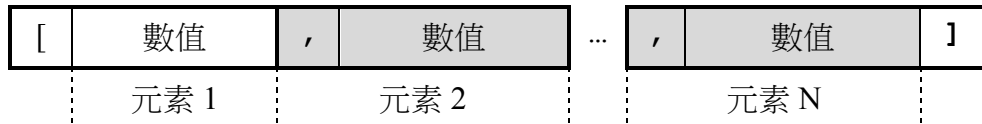


圖 A.4. 複合型態「陣列」之定義

陣列內之「元素」(Element) 應填入任一種合法數值，亦即上述之純數值，另一個完整 JSON 文件，或是另外一個陣列。下圖範例中陣列「**load**」內包含四個元素，第一個元素是整數 2，第二個元素是字串「**basic**」，第三與第四個元素則是 JSON 物件。

```
"temperature": [17.0, 36, 45.8],  
"load": [  
  2,  
  "basic",  
  {"volt": 110.24, "curr": 3.1, "watt": 312.5},  
  {  
    "volt" : 110.24,  
    "curr" : 2.48,  
    "watt" : 262.56,  
  }  
]
```

圖 A.5. 複合型態「陣列」之範例

—以下空白—

附錄B cron 格式概述

cron 是 UNIX 與類 UNIX 系統中常見的一種服務，用於設置週期執行命令。一個標準的 cron 服務，會在執行時讀取一設定檔，並依據該設定檔內容設定排程。由於 cron 服務應用相當廣泛，幾乎所有的 UNIX/類 UNIX 發行皆內含 cron 服務，因此 cron 設定檔格式（以下簡稱「cron 格式」）也隨之成為一種約定俗成的標準。

本標準文件同樣採用 cron 格式，作為指派閘道器之情境控制（參見 8.2.3）。本標準文件依循之規範為「crontab: schedule periodic background work – Commands & Utilities Reference, The Single UNIX® Specification, Issue 7 from The Open Group」，完整文件請參照 Open Group 網頁說明。本節將簡單介紹本標準內使用之 cron 格式。

本標準內使用之 cron 格式字包含六個欄位，以空白字元字符（ASCII 代碼 0x20）分隔。字串格式定義如下：

"	分	時	日	月	週	情境代號	"
---	---	---	---	---	---	------	---

圖 B.1. cron 格式字串

cron 格式字串前五個時間排程欄位定義如下：

- 分：代表每小時第幾分觸發。數值範圍為 0 至 59，不須前置補零。
- 時：代表每天第幾小時觸發。數值範圍為 0 至 23，不須前置補零。
- 日：代表每月第幾日觸發。數值範圍為 1 至 31，不須前置補零。
- 月：代表每年第幾月觸發。數值範圍為 1 至 12，不須前置補零。
- 週：代表每週第幾天觸發。數值範圍為 0 至 6，0 代表週日。

cron 服務運作時，若任一時間與上述五個排程欄位比對皆吻合時，則該排程將觸發執行。上述五個排程欄位內容可由控制字元進行擴充，定義如下：

表 B.1. cron 格式數值擴充字元

標記	比對吻合條件
x	時間等於數值 x
x,y,z	時間等於數值 x 、或數值 y 、或數值 z
x-y	時間等於 x 至 y 此一範圍內之任何數值
*	該欄位不須比對，一律視為吻合
x/y	時間等於 x 加上 y 的任意倍數的數值
x-z/y	時間等於 x 加上 y 的任意倍數的數值，但不超過 z
*/y	時間等於第一次執行時間加上 y 的任意倍數的數值

表 B.2. cron 格式範例

排程	定義
0 12 1 2 *	二月一日中午十二點整
0 12 * * *	每天中午十二點整
0 12 * * 3	每週三中午十二點整
0 12 * 2 *	二月份每週三中午十二點整
0 12 1 * *	每月一號中午十二點整
30 19,20,21 * * *	每天晚上七點半、八點半、九點半
0 18-22 * * 0,6	每週六日晚上六點到十點，每個整點
0 0/1 * * *	每隔一小時
0/10 7,8 * * *	每天早上七點到八點，每隔十分鐘
15-45/5 18 1 2 *	一月二日晚上六點十五分到四十五分，每隔五分鐘

—以下空白—

附錄C 通用代號定義

C.1. 通用回傳值

表 C.1. 通用回傳值一覽表

分類	代號	定義
共通	100	OK
	101	封包格式錯誤
	102	主要參數格式錯誤
	103	封包承載格式錯誤
	104	資訊不完整
	105	數值超出範圍
系統錯誤	201	服務已達上限
	204	閘道器未登記
	202	交易 (Transaction) 失敗
	203	資料重複登錄
裝置監控	301	未定義該項服務
	302	未定義該項參數
	303	參數數值格式錯誤
	304	裝置不支援該項服務
網路傳輸	401	指定裝置並未登記
	402	指定裝置並未上線
	403	場域網路連線失效：通訊模組故障
事件回報	404	裝置場域網路連線失效：無回應
	405	裝置場域網路連線正常，但執行狀態不明
	406	裝置硬體設備故障
系統內部錯誤	500~599	保留
服務錯誤	600~999	保留

C.2. 功能模組代號表

表 C.2. 功能模組代號一覽表

功能模組代號	名稱	說明
0	泛用服務對應表	所有受控裝置（含閘道器）皆須實作之預設功能模組，固定對應 0 號服務終端。
1~10	保留	不可使用
11	泛用閘道器設定	閘道器預設之功能模組，固定對應 1 號服務終端。用於連線等設定。
12	泛用閘道器排程	閘道器預設之功能模組，固定對應 2 號服務終端。用於設定定時排程之情境控制動作。
13~100	保留	不可使用
101	單相電力計	一般單相三線式單一迴路，或三相四線式單一迴路。
102	單相雙迴路電力計	一般單相三線式總迴路
103	三相電力計	三相四線式總迴路
104~129	保留	不可使用
151	泛用感測器	適用於純物理量之傳感器
152	泛用警報器	單純觸發之警報器
153	泛用閾值警報器	可設定閾值之警報器
154	LED 燈具狀態回報	多燈條 LED 燈具故障回報
154~179	保留	不可使用
201	二元開關	泛用型 ON/OFF 二元開關
202	定時器	泛用型 ON/OFF 定時開關
203	單頻道調光器	可設定強度之調光器
204	雙頻道 RGB 調光器	可設定個別頻道強度之調光器
205	三頻道 RGB 調光器	可設定個別頻道強度之調光器
206~229	保留	不可使用
240~254	保留	不可使用
255	已停用	該服務終端之功能模組已停用

C.3. 時區名稱對照表

本對照表依循本份文件撰寫時最新之 IANA Time Zone Database, v.2011n (2011/11/02) 製成。

表 C.3. 時區名稱對照表

國家/區域	時區名稱
GMT	GMT+0
GMT	GMT+1
GMT	GMT+2
GMT	GMT+3
GMT	GMT+4
GMT	GMT+5
GMT	GMT+6
GMT	GMT+7
GMT	GMT+8
GMT	GMT+9
GMT	GMT+10
GMT	GMT+11
GMT	GMT+12
GMT	GMT-0
GMT	GMT-1
GMT	GMT-2
GMT	GMT-3
GMT	GMT-4
GMT	GMT-5
GMT	GMT-6
GMT	GMT-7
GMT	GMT-8
GMT	GMT-9
GMT	GMT-10
GMT	GMT-11
GMT	GMT-12
GMT	GMT-13
GMT	GMT-14

國家/區域	時區名稱
Aaland Islands	Europe/Mariehamn
Afghanistan	Asia/Kabul
Albania	Europe/Tirane
Algeria	Africa/Algiers
Andorra	Europe/Andorra
Angola	Africa/Luanda
Anguilla	America/Anguilla
Antarctica	Antarctica/Casey
Antarctica	Antarctica/Davis
Antarctica	Antarctica/DumontDUrville
Antarctica	Antarctica/Macquarie
Antarctica	Antarctica/Mawson
Antarctica	Antarctica/McMurdo
Antarctica	Antarctica/Palmer
Antarctica	Antarctica/Rothera
Antarctica	Antarctica/South_Pole
Antarctica	Antarctica/Syowa
Antarctica	Antarctica/Vostok
Antigua & Barbuda	America/Antigua
Argentina	America/Argentina/Buenos_Aires
Argentina	America/Argentina/Catamarca
Argentina	America/Argentina/Cordoba
Argentina	America/Argentina/Jujuy
Argentina	America/Argentina/La_Rioja
Argentina	America/Argentina/Mendoza
Argentina	America/Argentina/Rio_Gallegos
Argentina	America/Argentina/Salta
Argentina	America/Argentina/San_Juan
Argentina	America/Argentina/San_Luis
Argentina	America/Argentina/Tucuman
Argentina	America/Argentina/Ushuaia
Armenia	Asia/Yerevan
Aruba	America/Aruba
Australia	Australia/Adelaide

國家/區域	時區名稱
Australia	Australia/Brisbane
Australia	Australia/Broken_Hill
Australia	Australia/Currie
Australia	Australia/Darwin
Australia	Australia/Eucla
Australia	Australia/Hobart
Australia	Australia/Lindeman
Australia	Australia/Lord_Howe
Australia	Australia/Melbourne
Australia	Australia/Perth
Australia	Australia/Sydney
Austria	Europe/Vienna
Azerbaijan	Asia/Baku
Bahamas	America/Nassau
Bahrain	Asia/Bahrain
Bangladesh	Asia/Dhaka
Barbados	America/Barbados
Belarus	Europe/Minsk
Belgium	Europe/Brussels
Belize	America/Belize
Benin	Africa/Porto-Novo
Bermuda	Atlantic/Bermuda
Bhutan	Asia/Thimphu
Bolivia	America/La_Paz
Bosnia & Herzegovina	Europe/Sarajevo
Botswana	Africa/Gaborone
Brazil	America/Araguaina
Brazil	America/Bahia
Brazil	America/Belem
Brazil	America/Boa_Vista
Brazil	America/Campo_Grande
Brazil	America/Cuiaba
Brazil	America/Eirunepe
Brazil	America/Fortaleza

國家/區域	時區名稱
Brazil	America/Maceio
Brazil	America/Manaus
Brazil	America/Noronha
Brazil	America/Porto_Velho
Brazil	America/Recife
Brazil	America/Rio_Branco
Brazil	America/Santarem
Brazil	America/Sao_Paulo
Britain (UK)	Europe/London
British Indian Ocean Territory	Indian/Chagos
Brunei	Asia/Brunei
Bulgaria	Europe/Sofia
Burkina Faso	Africa/Ouagadougou
Burundi	Africa/Bujumbura
Cambodia	Asia/Phnom_Penh
Cameroon	Africa/Douala
Canada	America/Atikokan
Canada	America/Blanc-Sablon
Canada	America/Cambridge_Bay
Canada	America/Dawson
Canada	America/Dawson_Creek
Canada	America/Edmonton
Canada	America/Glace_Bay
Canada	America/Goose_Bay
Canada	America/Halifax
Canada	America/Inuvik
Canada	America/Iqaluit
Canada	America/Moncton
Canada	America/Montreal
Canada	America/Nipigon
Canada	America/Pangnirtung
Canada	America/Rainy_River
Canada	America/Rankin_Inlet

國家/區域	時區名稱
Canada	America/Regina
Canada	America/Resolute
Canada	America/St_Johns
Canada	America/Swift_Current
Canada	America/Thunder_Bay
Canada	America/Toronto
Canada	America/Vancouver
Canada	America/Whitehorse
Canada	America/Winnipeg
Canada	America/Yellowknife
Cape Verde	Atlantic/Cape_Verde
Cayman Islands	America/Cayman
Central African Rep.	Africa/Bangui
Chad	Africa/Ndjamena
Chile	America/Santiago
Chile	Pacific/Easter
China	Asia/Chongqing
China	Asia/Harbin
China	Asia/Kashgar
China	Asia/Shanghai
China	Asia/Urumqi
Christmas Island	Indian/Christmas
Cocos (Keeling) Islands	Indian/Cocos
Colombia	America/Bogota
Comoros	Indian/Comoro
Congo (Dem. Rep.)	Africa/Kinshasa
Congo (Dem. Rep.)	Africa/Lubumbashi
Congo (Rep.)	Africa/Brazzaville
Cook Islands	Pacific/Rarotonga
Costa Rica	America/Costa_Rica
Cote d'Ivoire	Africa/Abidjan
Croatia	Europe/Zagreb
Cuba	America/Havana
Cyprus	Asia/Nicosia

國家/區域	時區名稱
Czech Republic	Europe/Prague
Denmark	Europe/Copenhagen
Djibouti	Africa/Djibouti
Dominica	America/Dominica
Dominican Republic	America/Santo_Domingo
East Timor	Asia/Dili
Ecuador	America/Guayaquil
Ecuador	Pacific/Galapagos
Egypt	Africa/Cairo
El Salvador	America/El_Salvador
Equatorial Guinea	Africa/Malabo
Eritrea	Africa/Asmara
Estonia	Europe/Tallinn
Ethiopia	Africa/Addis_Ababa
Falkland Islands	Atlantic/Stanley
Faroe Islands	Atlantic/Faroe
Fiji	Pacific/Fiji
Finland	Europe/Helsinki
France	Europe/Paris
French Guiana	America/Cayenne
French Polynesia	Pacific/Gambier
French Polynesia	Pacific/Marquesas
French Polynesia	Pacific/Tahiti
French Southern & Antarctic Lands	Indian/Kerguelen
Gabon	Africa/Libreville
Gambia	Africa/Banjul
Georgia	Asia/Tbilisi
Germany	Europe/Berlin
Ghana	Africa/Ackra
Gibraltar	Europe/Gibraltar
Greece	Europe/Athens
Greenland	America/Danmarkshavn
Greenland	America/Godthab

國家/區域	時區名稱
Greenland	America/Scoresbysund
Greenland	America/Thule
Grenada	America/Grenada
Guadeloupe	America/Guadeloupe
Guam	Pacific/Guam
Guatemala	America/Guatemala
Guernsey	Europe/Guernsey
Guinea	Africa/Conakry
Guinea-Bissau	Africa/Bissau
Guyana	America/Guyana
Haiti	America/Port-au-Prince
Honduras	America/Tegucigalpa
Hong Kong	Asia/Hong_Kong
Hungary	Europe/Budapest
Iceland	Atlantic/Reykjavik
India	Asia/Kolkata
Indonesia	Asia/Jakarta
Indonesia	Asia/Jayapura
Indonesia	Asia/Makassar
Indonesia	Asia/Pontianak
Iran	Asia/Tehran
Iraq	Asia/Baghdad
Ireland	Europe/Dublin
Isle of Man	Europe/Isle_of_Man
Israel	Asia/Jerusalem
Italy	Europe/Rome
Jamaica	America/Jamaica
Japan	Asia/Tokyo
Jersey	Europe/Jersey
Jordan	Asia/Amman
Kazakhstan	Asia/Almaty
Kazakhstan	Asia/Aqtau
Kazakhstan	Asia/Aqtobe
Kazakhstan	Asia/Oral

國家/區域	時區名稱
Kazakhstan	Asia/Qyzylorda
Kenya	Africa/Nairobi
Kiribati	Pacific/Enderbury
Kiribati	Pacific/Kiritimati
Kiribati	Pacific/Tarawa
Korea (North)	Asia/Pyongyang
Korea (South)	Asia/Seoul
Kuwait	Asia/Kuwait
Kyrgyzstan	Asia/Bishkek
Laos	Asia/Vientiane
Latvia	Europe/Riga
Lebanon	Asia/Beirut
Lesotho	Africa/Maseru
Liberia	Africa/Monrovia
Libya	Africa/Tripoli
Liechtenstein	Europe/Vaduz
Lithuania	Europe/Vilnius
Luxembourg	Europe/Luxembourg
Macau	Asia/Macau
Macedonia	Europe/Skopje
Madagascar	Indian/Antananarivo
Malawi	Africa/Blantyre
Malaysia	Asia/Kuala_Lumpur
Malaysia	Asia/Kuching
Maldives	Indian/Maldives
Mali	Africa/Bamako
Malta	Europe/Malta
Marshall Islands	Pacific/Kwajalein
Marshall Islands	Pacific/Majuro
Martinique	America/Martinique
Mauritania	Africa/Nouakchott
Mauritius	Indian/Mauritius
Mayotte	Indian/Mayotte
Mexico	America/Bahia_Banderas

國家/區域	時區名稱
Mexico	America/Cancun
Mexico	America/Chihuahua
Mexico	America/Hermosillo
Mexico	America/Matamoros
Mexico	America/Mazatlan
Mexico	America/Merida
Mexico	America/Mexico_City
Mexico	America/Monterrey
Mexico	America/Ojinaga
Mexico	America/Santa_Isabel
Mexico	America/Tijuana
Micronesia	Pacific/Chuuk
Micronesia	Pacific/Kosrae
Micronesia	Pacific/Pohnpei
Moldova	Europe/Chisinau
Monaco	Europe/Monaco
Mongolia	Asia/Choibalsan
Mongolia	Asia/Hovd
Mongolia	Asia/Ulaanbaatar
Montenegro	Europe/Podgorica
Montserrat	America/Montserrat
Morocco	Africa/Casablanca
Mozambique	Africa/Maputo
Myanmar (Burma)	Asia/Rangoon
Namibia	Africa/Windhoek
Nauru	Pacific/Nauru
Nepal	Asia/Kathmandu
Netherlands	Europe/Amsterdam
Netherlands Antilles	America/Curacao
New Caledonia	Pacific/Noumea
New Zealand	Pacific/Auckland
New Zealand	Pacific/Chatham
Nicaragua	America/Managua
Niger	Africa/Niamey

國家/區域	時區名稱
Nigeria	Africa/Lagos
Niue	Pacific/Niue
Norfolk Island	Pacific/Norfolk
Northern Mariana Islands	Pacific/Saipan
Norway	Europe/Oslo
Oman	Asia/Muscat
Pakistan	Asia/Karachi
Palau	Pacific/Palau
Palestine	Asia/Gaza
Panama	America/Panama
Papua New Guinea	Pacific/Port_Moresby
Paraguay	America/Asuncion
Peru	America/Lima
Philippines	Asia/Manila
Pitcairn	Pacific/Pitcairn
Poland	Europe/Warsaw
Portugal	Atlantic/Azores
Portugal	Atlantic/Madeira
Portugal	Europe/Lisbon
Puerto Rico	America/Puerto_Rico
Qatar	Asia/Qatar
Reunion	Indian/Reunion
Romania	Europe/Bucharest
Russia	Asia/Anadyr
Russia	Asia/Irkutsk
Russia	Asia/Kamchatka
Russia	Asia/Krasnoyarsk
Russia	Asia/Magadan
Russia	Asia/Novokuznetsk
Russia	Asia/Novosibirsk
Russia	Asia/Omsk
Russia	Asia/Sakhalin
Russia	Asia/Vladivostok
Russia	Asia/Yakutsk

國家/區域	時區名稱
Russia	Asia/Yekaterinburg
Russia	Europe/Kaliningrad
Russia	Europe/Moscow
Russia	Europe/Samara
Russia	Europe/Volgograd
Rwanda	Africa/Kigali
Samoa (American)	Pacific/Pago_Pago
Samoa (western)	Pacific/Apia
San Marino	Europe/San_Marino
Sao Tome & Principe	Africa/Sao_Tome
Saudi Arabia	Asia/Riyadh
Senegal	Africa/Dakar
Serbia	Europe/Belgrade
Seychelles	Indian/Mahe
Sierra Leone	Africa/Freetown
Singapore	Asia/Singapore
Slovakia	Europe/Bratislava
Slovenia	Europe/Ljubljana
Solomon Islands	Pacific/Guadalcanal
Somalia	Africa/Mogadishu
South Africa	Africa/Johannesburg
South Georgia & the South Sandwich Islands	Atlantic/South_Georgia
Spain	Africa/Ceuta
Spain	Atlantic/Canary
Spain	Europe/Madrid
Sri Lanka	Asia/Colombo
St Barthelemy	America/St_Barthelemy
St Helena	Atlantic/St_Helena
St Kitts & Nevis	America/St_Kitts
St Lucia	America/St_Lucia
St Martin (French part)	America/Marigot
St Pierre & Miquelon	America/Miquelon
St Vincent	America/St_Vincent

國家/區域	時區名稱
Sudan	Africa/Khartoum
Suriname	America/Paramaribo
Svalbard & Jan Mayen	Arctic/Longyearbyen
Swaziland	Africa/Mbabane
Sweden	Europe/Stockholm
Switzerland	Europe/Zurich
Syria	Asia/Damascus
Taiwan	Asia/Taipei
Tajikistan	Asia/Dushanbe
Tanzania	Africa/Dar_es_Salaam
Thailand	Asia/Bangkok
Togo	Africa/Lome
Tokelau	Pacific/Fakaofu
Tonga	Pacific/Tongatapu
Trinidad & Tobago	America/Port_of_Spain
Tunisia	Africa/Tunis
Turkey	Europe/Istanbul
Turkmenistan	Asia/Ashgabat
Turks & Caicos Is	America/Grand_Turk
Tuvalu	Pacific/Funafuti
US minor outlying islands	Pacific/Johnston
US minor outlying islands	Pacific/Midway
US minor outlying islands	Pacific/Wake
Uganda	Africa/Kampala
Ukraine	Europe/Kiev
Ukraine	Europe/Simferopol
Ukraine	Europe/Uzhgorod
Ukraine	Europe/Zaporozhye
United Arab Emirates	Asia/Dubai
United States	America/Adak
United States	America/Anchorage
United States	America/Boise
United States	America/Chicago
United States	America/Denver

國家/區域	時區名稱
United States	America/Detroit
United States	America/Indiana/Indianapolis
United States	America/Indiana/Knox
United States	America/Indiana/Marengo
United States	America/Indiana/Petersburg
United States	America/Indiana/Tell_City
United States	America/Indiana/Vevay
United States	America/Indiana/Vincennes
United States	America/Indiana/Winamac
United States	America/Juneau
United States	America/Kentucky/Louisville
United States	America/Kentucky/Monticello
United States	America/Los_Angeles
United States	America/Menominee
United States	America/New_York
United States	America/Nome
United States	America/North_Dakota/Center
United States	America/North_Dakota/New_Salem
United States	America/Phoenix
United States	America/Shiprock
United States	America/Yakutat
United States	Pacific/Honolulu
Uruguay	America/Montevideo
Uzbekistan	Asia/Samarkand
Uzbekistan	Asia/Tashkent
Vanuatu	Pacific/Efate
Vatican City	Europe/Vatican
Venezuela	America/Caracas
Vietnam	Asia/Ho_Chi_Minh
Virgin Islands (UK)	America/Tortola
Virgin Islands (US)	America/St_Thomas
Wallis & Futuna	Pacific/Wallis
Western Sahara	Africa/El_Aaiun
Yemen	Asia/Aden

國家/區域	時區名稱
Zambia	Africa/Lusaka
Zimbabwe	Africa/Harare

—以下空白—

附錄D JSON 授權文件

以下 JSON 授權文件內容原始出處為 JSON 組織官方網站 (<http://www.json.org>)。依據授權內容規範，以下內容須視為本標準文件的一部分。

The JSON License

Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.