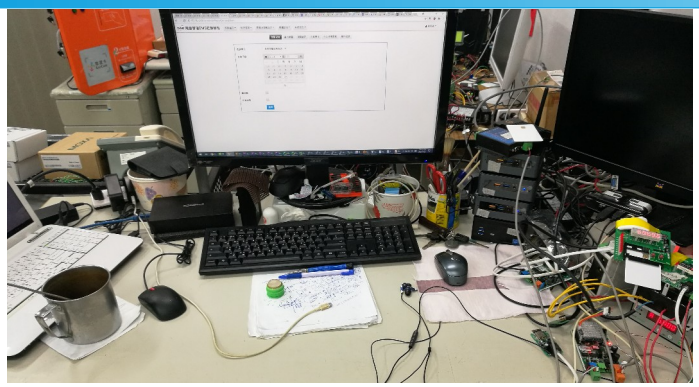
這次要分享的議題是「用愛發電；為愛鼓掌，從 EMS 班班有冷氣，了解電能管理」

2019 年分享，致敬「米家」，米家就在我家的主題，隔了一年後，今年藉著 EMS 班班有冷氣這個政府政策，來示範什麼是蹭熱度 XD

曬桌面



黃志賢　Hoyo.idv.tw　　台科電科技研發工程師

我是黃志賢網路代號 Hoyo ，經營 hoyo.idv.tw 網站從 2000 年底至今，

現職為台科電科技的研發工程師，台科電科技是台灣一間，專精於智慧能源管理系統產品，研發及製造的公司。

是 EMS 班班有冷氣專案，其中一間供貨廠商，負責學校需量電表、班級電表、卡機硬體設備，以及 EMS Server 中間資料交換及處理 OpenADR 需量反應卸載功能媒介

有關 EMS 此專案的細節後續會介紹

附圖是 Hoyo 上班座位的實景，和一般軟體工程師不同，負責工控、韌體的工程師，周遭推滿了各種硬體設備，不過這麼亂應該是個人因素吧 :)

聽完後

- 知道政府「 EMS 班班有冷氣」政策
- 知道地球人圍繞在用電需量的努力
- 知道台灣可以考慮買 TaiSEIA 家電 DIY 智能居家
- 知道地球上 Pi 這個選擇，不是只有樹莓 Pi

本次分享希望可以讓大家了解政府 EMS 班班有冷氣政策，(→)
也可以知道地球人發明了電這個東西後費了多少勁來維持，(→)
還有認識台灣制定的智能居家 TaiSEIA 101 是什麼，(→)
最後介紹的是 Pi ，現在時代進步了，不是只有樹莓派了，就像現在提起個人電腦誰還會想到 IBM ？

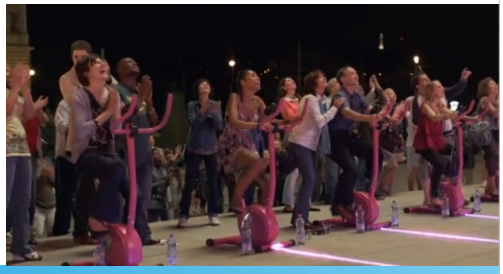
今年有停電經歷的舉個手，好～好～好～很踴躍，有很多妙麗 :)

今年經歷了 513 大停電、 517 限電、 727 椅子停電，讓大家再度重視了電這個議題，我只希望台電不要再出錯了，因為停電我就又要改簡報了

發生的當下大家才會注意台電的存在，隨著時間過去熱度散去，大家就忘了，我也忘記了，因為我是哈利 XD

為什麼會跳電、停電？不跳行不行

- 基於設備安全
- 發電量固定、需求卻不固定
- 儲存代價龐大



https://www.youtube.com/watch?v=FlKEKEfEjN4

為什麼會跳電？為什麼要停電？不能撐一下撐過去就好嗎？

電因為跑在電線上，電線是真實的東西，那自然就有所有的物理極限

電線超過極限就是起火，搜尋電線走火就有很多資料，零件超過極限就是壞掉，發電機組本身也是電器設備也是需要遵循這個原則，所以在超過負載前斷開負載是為了保護設備，

所以發電是不能像乳溝那樣擠的，硬擠是會擠出人命的 XD

發電量固定，台電的發電量要怎麼知道，可以到以下這個網址查詢，除了總發電之外還可以知道每種發電產出的量及比例，從資料看來台灣目前主要還是「燒東西」來發電

需求不固定，供電緊不緊張？使用率如何得知？

台電一樣有公開資訊，從圖可知，查詢的當下使用率只有百分之 84 ，還有盈餘

不過那也是在班班有冷氣執行前，執行後會多多少？ (→)

根據統計約有 10 萬教室，每間教室裝設 2 台冷氣，每台冷氣功率為 7 千瓦，當冷氣全開可達 14 萬千瓦，以這張截圖的發電量來計算，等於多了百分之 0.4 的用電

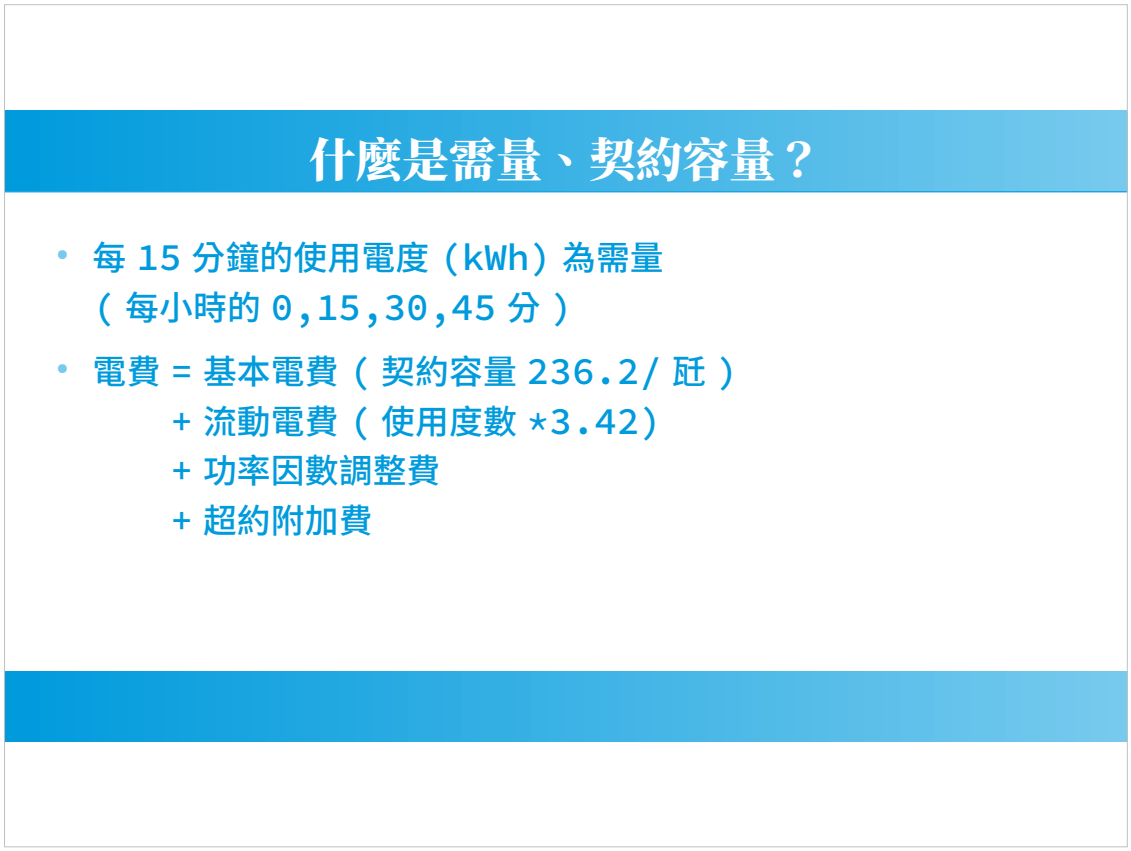
儲存代價

- 電腦用 UPS 不斷電 – 交流電充電
- 家庭儲能 – 太陽能充電
  - Tesla Powerwall 14kWh >30 萬
  - 台灣客制 3kW <10 萬
- 煮鹽、搬石頭、太陽塔、壓縮空氣 ...

https://www.youtube.com/watch?v=kOoOPuvqA_8

https://www.tesla.com/en_gb/powerwall/design
https://www.tesla.com/sites/default/files/pdfs/powerwall/Powerwall_2_AC_Datasheet_EN_NA.pdf

電不夠就跳電、停電，那把電先存起來不就好了？
把電存起來，在停電的時候切換要花多少錢？以電腦用途使用交流電充電正經八百的 UPS 不斷電來說，只看容量 1kW 的 UPS 從 2000 到上萬都有， (→)
家庭用的規模可以使用「家庭儲能」作為關鍵字搜尋，基本上儲能的來源都是太陽能，最知名的產品應該是特斯拉的 Powerwall ，這類產品因為牽扯到安裝，費用無法給出準確的價格，特斯拉最基本的組合就要 30 萬，沒有這麼多預算，也可以找專業的店家客制組合，不到 10 萬就有一個雛型，這些儲存方案，都是使用鋰電池或鉛酸電池來當作儲存的媒介，已經是現今最好的方式了 (→)
電池行不通就延伸了很多奇奇怪怪的方法，例如使用運水、搬石頭的靠位移的發電方式，能源充足的時候運到高處，需要的時候再落下發電。或是靠鏡子反射陽光將鹽加熱，需要發電的時候就拿來煮水，靠水蒸氣發電
總之你想的到的獨眼龍也想的到，為什麼現在沒有就代表不符合經濟效益，如果符合經濟效益大家都有了， Hoyo 也沒機會上台了 XD



前面知道了台電供電有「同時量」的限制，打個比喻來說就像是
你今天賣便當，公司每天固定訂大量、有人臨時買少量，每天要賣的數量至少要把固定訂的數量做出來，多的才賣散戶，這個固定的數量就是契約容量
契約容量是容量，既然是容量就有結算的時間，每 15 分鐘為區間做結算，這 15 分鐘的使用量稱為需量 (→)
契約容量電費計算是基本電費 + 流動電費也就是實際使用的電度 + 功率因數調整費 + 超約罰款
流動電費這個很好理解，用多少花多少使用者付費很合理，基本費用以及超約就比較難理解，這也是一般用電需量管理的重點，以剛剛的便當舉例就是當你訂 2 個便當的時候就要先付訂金， 1 個便當訂金 10 元， 2 個就要 20 元，實際有吃掉的話一個便當還要付 50 元，但是訂 2 個只吃一個，訂金也是不退還，萬一今天特別餓吃了 3 個，當然第 3 個也要付 50 元，不過因為多吃了所以原先的訂金 10 元要變成改收 20 元，這就是超約罰款，功率因數調整費和本次主題無關就先不講解

需量超約附加費

- 超過 10% 以內罰 2 倍， 10% 以上罰 3 倍
- 契約容量（100 kW）超約用電（130 kW）

基本電費 = 100
+ 10×2
+ 20×3
= 180kW

23,620 → **42,516** (18,896 = 80%)

https://www.lhu.edu.tw/e_paper/96/lunghwa_paper_96008/night.htm

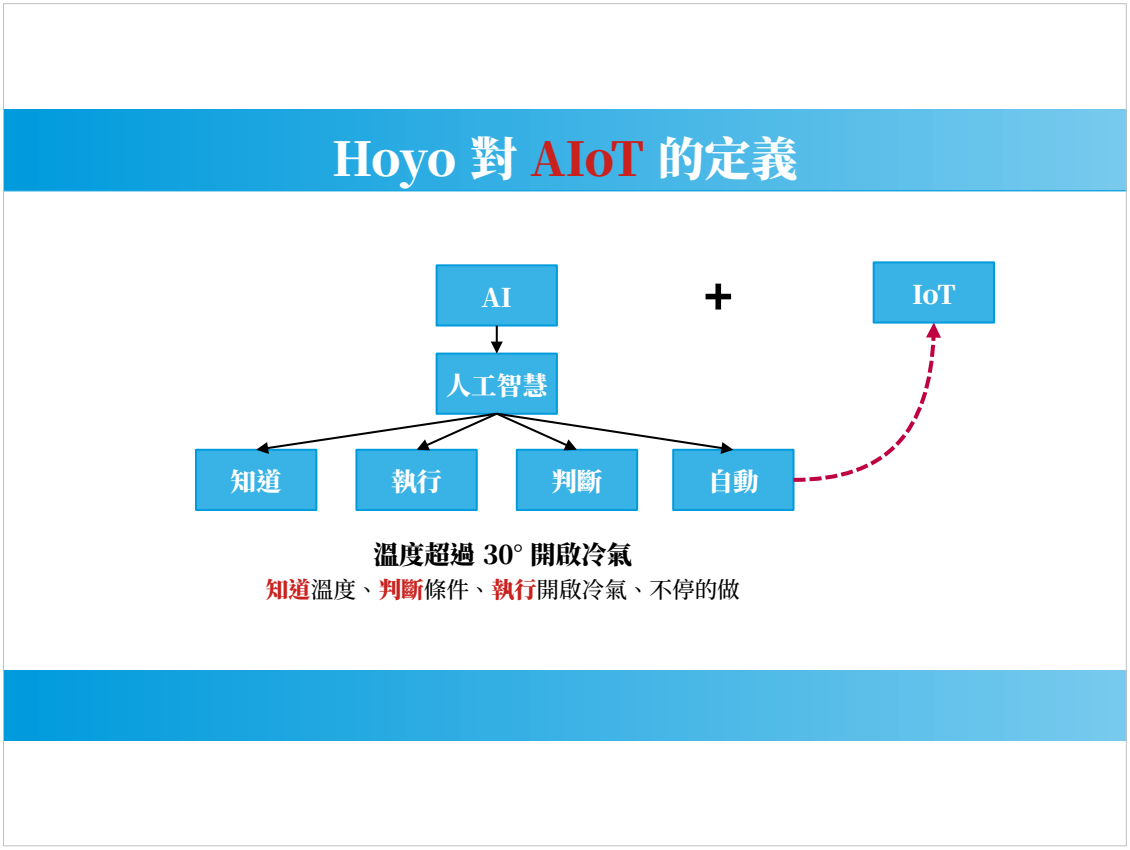
剛剛說到超約罰款，那實際多吃便當是怎麼處罰的呢？啊～不是，是超約是怎麼計算的呢？

當超過契約容量未超過百之分 10 是罰 2 倍，百分之 10 以上是罰 3 倍

舉例來說契約容量 100 千瓦，當有一個區間使用達到 130 千瓦

未超過百分之 10 的那 10 千瓦要罰 2 倍，超過的百分之 10 的那 20 千瓦要罰 3 倍 (→)

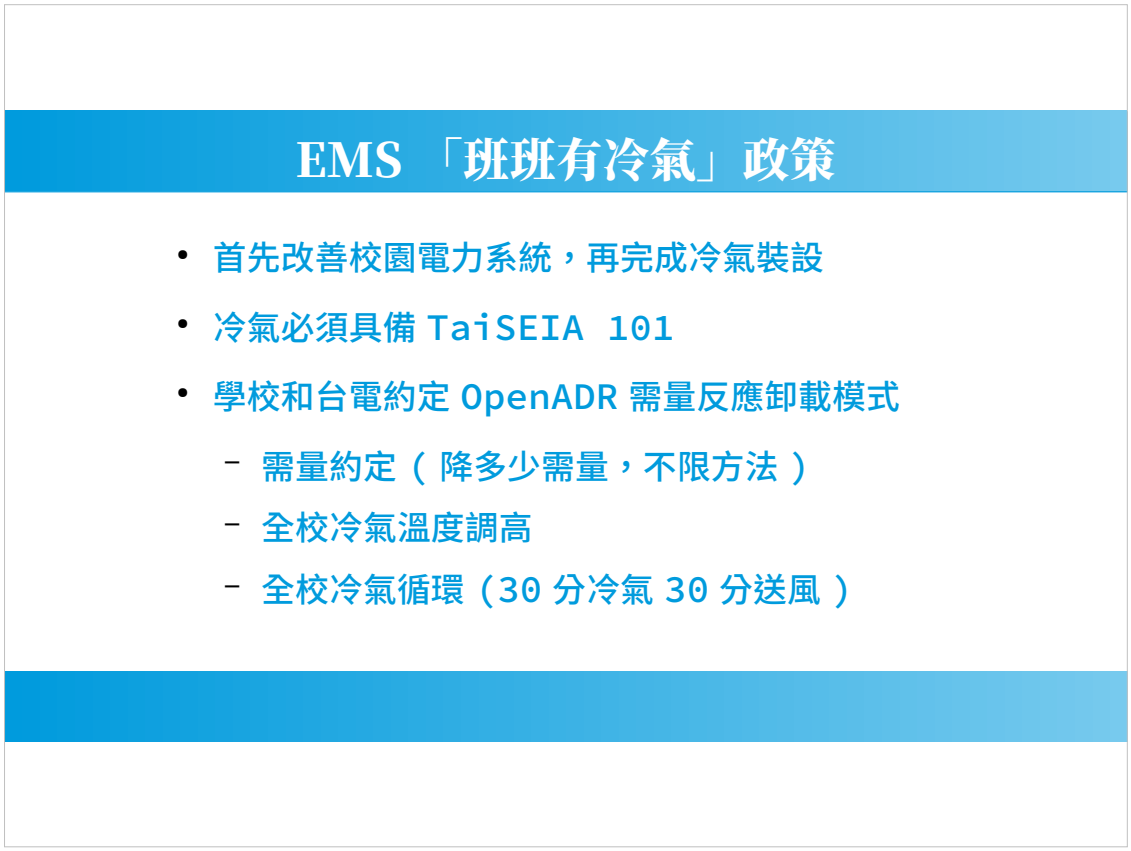
所以最後算出是 180 千瓦，也就是這個月的基本電費不是用 100kW 計算費用是用 180kW 也就是基本電費從兩萬三漲到四萬二多了一萬八



開始介紹 EMS 之前 Hoyo 想先說一下我對於 AIoT 的定義，我將 AIoT 簡單粗暴的拆成 AI 和 IoT ， IoT 是一切末端設備， AI 是人工智慧，對於人工智慧又拆分為知道、執行、判斷、自動

知道就是回傳數據以及狀態，溫度幾度、 I/O 是 0 還是 1 也就是讀取，執行是修改設定或是實際的硬體動作也就是寫入，判斷就是 IF ，假如什麼什麼就幹嘛幹嘛的那個 IF ，自動就是不斷的重複執行， Hoyo 認為自動是最困難的環節

例如，溫度超過 30 度要開啟冷氣，就包含了 3 種完整的能力，也就是知道溫度、判斷是否大於 30 度、動作開啟冷氣，如此不停的執行，我就認為是 AIoT
EMS 班班有冷氣就是一套完整的 AIoT 系統，接下來就來講講 EMS 內如何對應到 AIoT 的



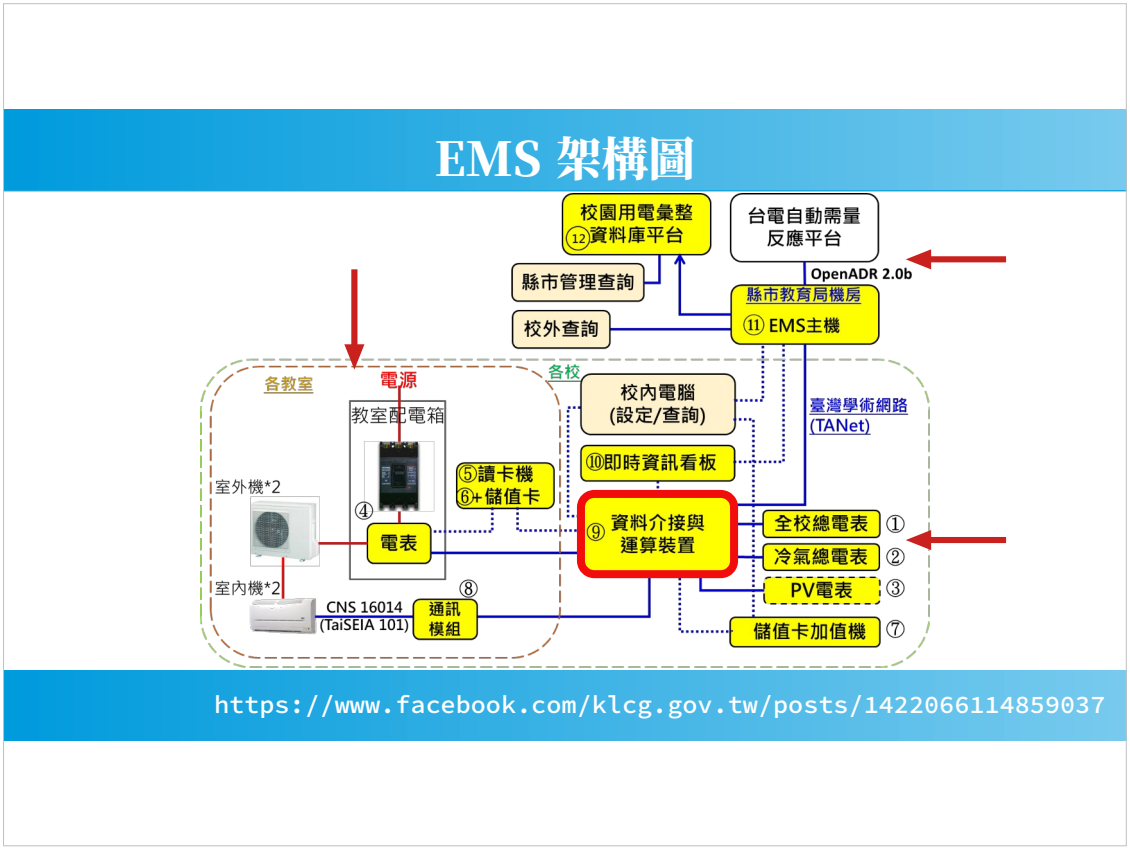
首先簡單認識一下 EMS 班班有冷氣政策有關的項目

因應冷氣增加耗電需求，先進行電力硬體改善工程，根據預計增加的冷氣使用電量，來增加或更替變電、供電設備

為了可以有效的控制冷氣，規範只能使用 TaiSEIA 101 規格的冷氣，什麼是 TaiSEIA 後面會說明

最後就是學校和台電簽訂 OpenADR 需量反應的卸載模式，總共有 3 種模式

需量約定是學校自行根據需量卸載需求降低用電，要用什麼方法並沒有規定，不過為了有效卸載一般都是選擇關閉電器，第 2 種是將全校冷氣溫度調高，第 3 種是冷氣和送風的模式循環切換



這張是架構圖，

簡單解說一下，每間教室架設了具備 CNS16014 也就是 TaiSEIA 101 功能的冷氣，使用冷氣卡插卡供電，計費單位是電度，電表須經過檢驗合格，

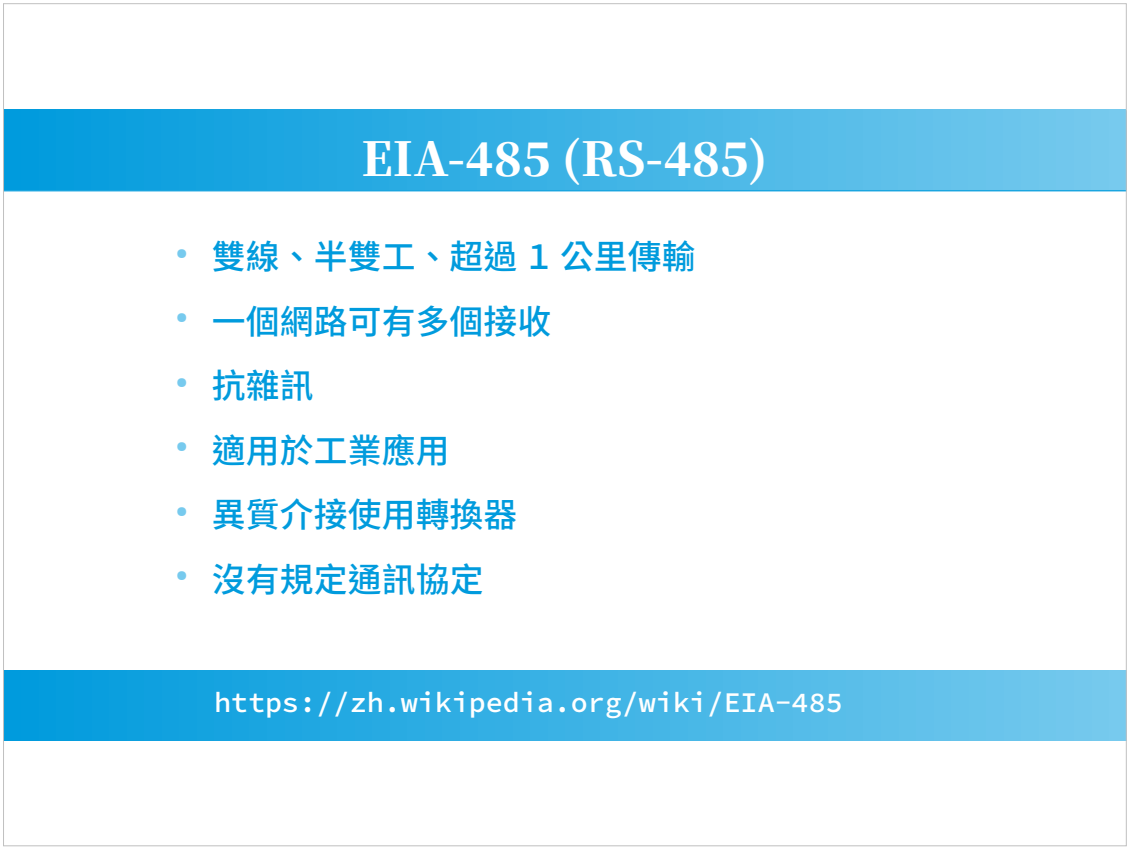
平時 IPC 不斷的收集電表資訊上傳給 EMS 主機，當需量反應事件發生時，由台電 OpenADR VTN 通知 EMS 主機的 VEN ，然後使用 MQTT 通知校內的資料介接與運算裝置，我們稱為 IPC ，再由 IPC 進行冷氣控制

此次議題只針對 OpenADR 自動需量反應、智慧電表資料讀取以及 TaiSEIA 101 進行討論

蛤～你問 9 號這格為什麼有紅邊這麼突兀？我這麼低調才不會告訴你因為這是我負責的！

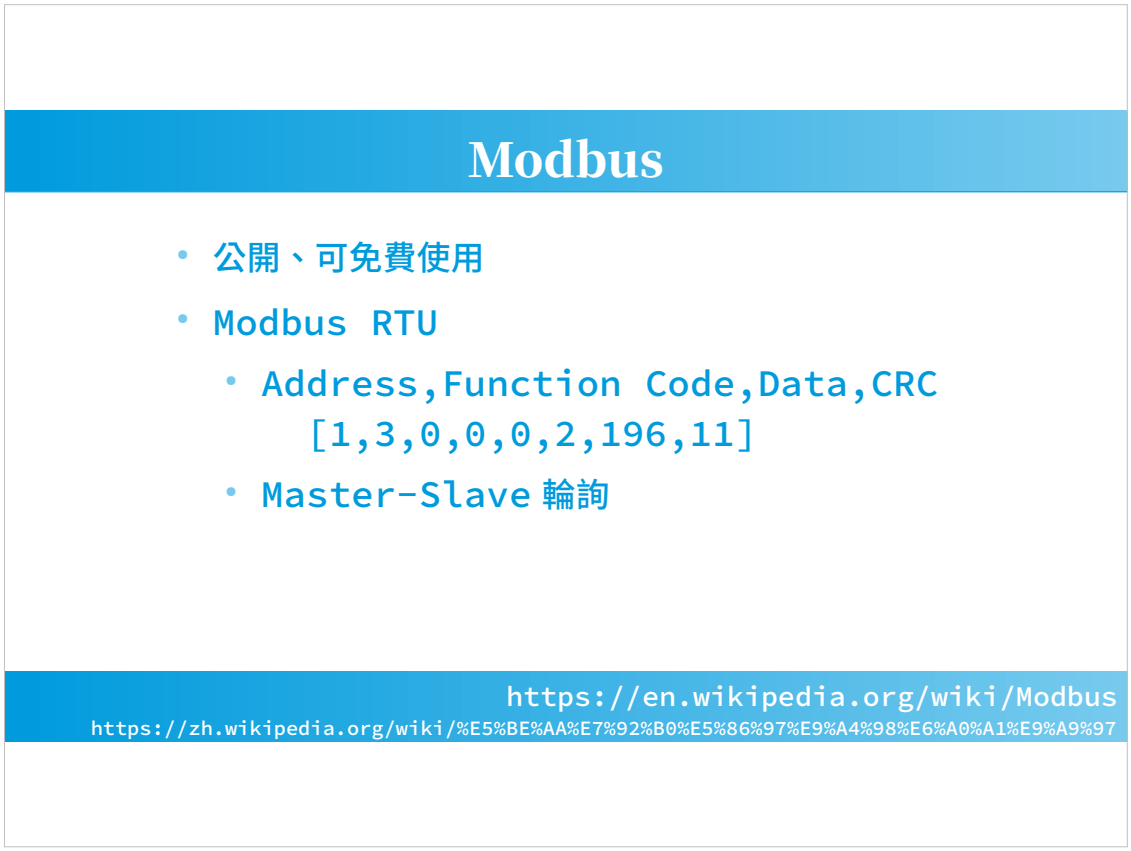
EMS 為了可以得知用電資訊，採用了智慧電表，智慧電表目前的定義是具備通訊界面，可以取得用電資訊，例如：電壓、電流、電度等資訊，一般也提供了適度的寫入設定方法，例如電度歸零，或是有繼電器功能，也可以控制繼電器，通訊界面一般都是 RS485 ，通訊協定則是 Modbus

--



剛剛說到智慧電表使用 RS485 以及 Modbus ，接下來就來說明為什麼要這樣選擇
EIA-485 也就是業界較常聽到的 RS485
因為雙線採用電壓差，可使用超過 1 公里的資料傳輸，還可以保證一定的穩定性，
一組線又可以連接多達上百個設備，又抗雜訊的特性，剛好適用於工業使用，也就是
「俗又大碗」的特性

因為太廣泛了，所以幾乎所有的界面都可以轉 RS485 ，用電腦有 USB to
RS485 ， MCU Arduino 有 UART to RS485 ，不過工業控制上用最多的是網路轉
RS485 ，也就是 Ethernet to RS485 ， WiFi 穩定性較差使用場景侷限大
RS485 是物理媒介，沒有規定通訊協定，就像網路線也不是一定要跑 HTTP ，可以跑
SSH,MQTT 等協定，不過通常都是搭配 Modbus 通訊協定，稍後會說明



講到工控以及 RS485 ，就一定會提到 Modbus

Modbus 作為通訊協定，能夠廣為流傳，自然就是公開並且沒有版權可免費使用，因為 RS485 是串接，因此使用 Modbus RTU 協議

Modbus RTU 的字元格式是 Byte ，資料格式由位址 Address, Function Code, Data, CRC 組成

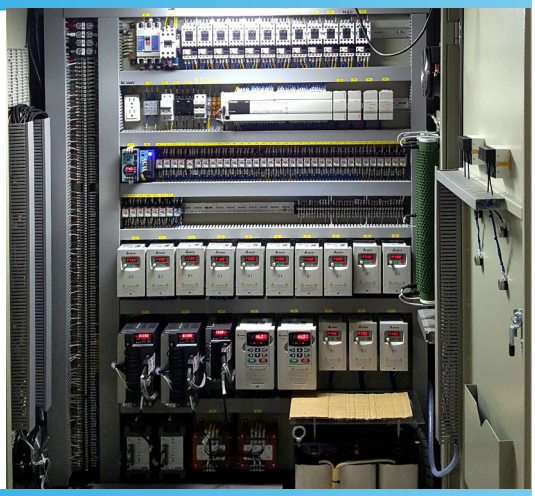
1 就是位址 Address ，同一串線位址不能重複，理論上最大值是 0-255 ，也就是一條 RS485 可以串接 256 個設備， 3 是 Function Code ， Data 也就是需要執行的內容，最後是兩個 Byte 的 CRC ， CRC 的格式是 CRC-16-IBM

執行是使用 Master-Slave 輪詢的方式

--

控制箱、配電盤 ...

- 環境複雜
- 緊湊
- 干擾多

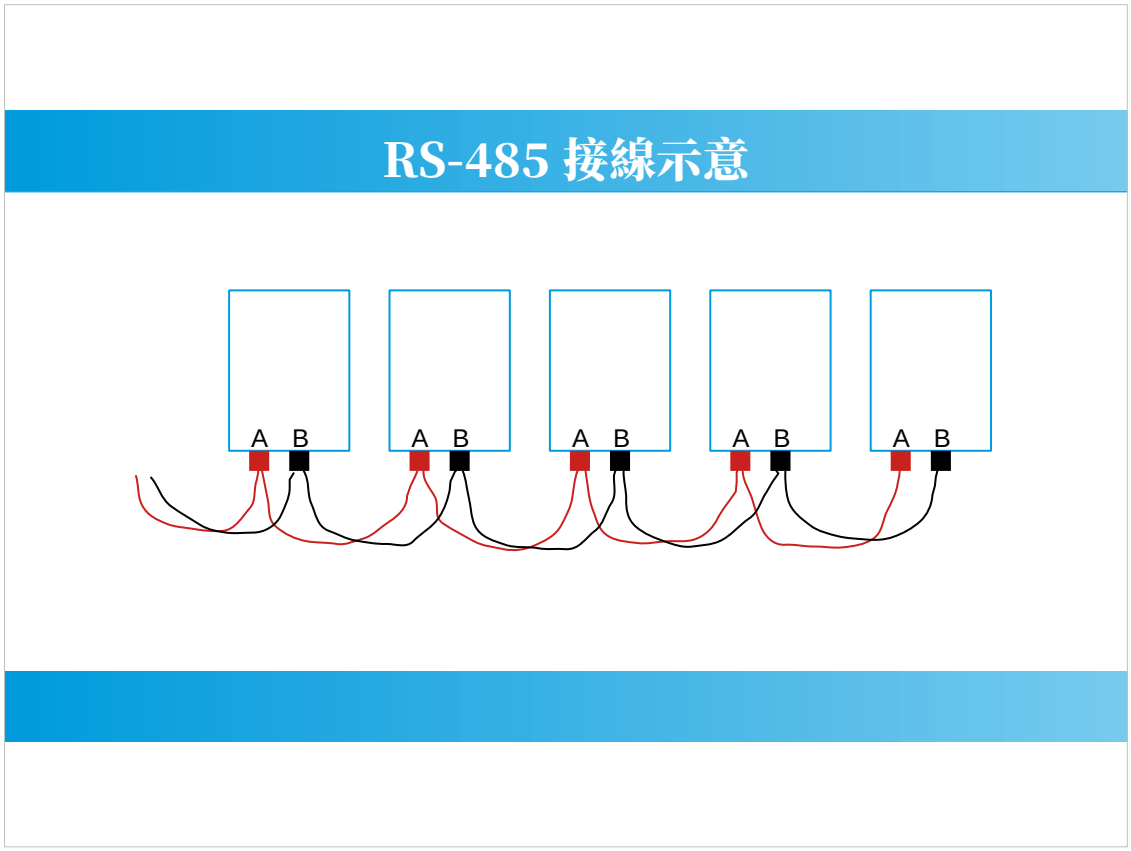


https://www.huicheng.com.tw/product-detail-2421114.html

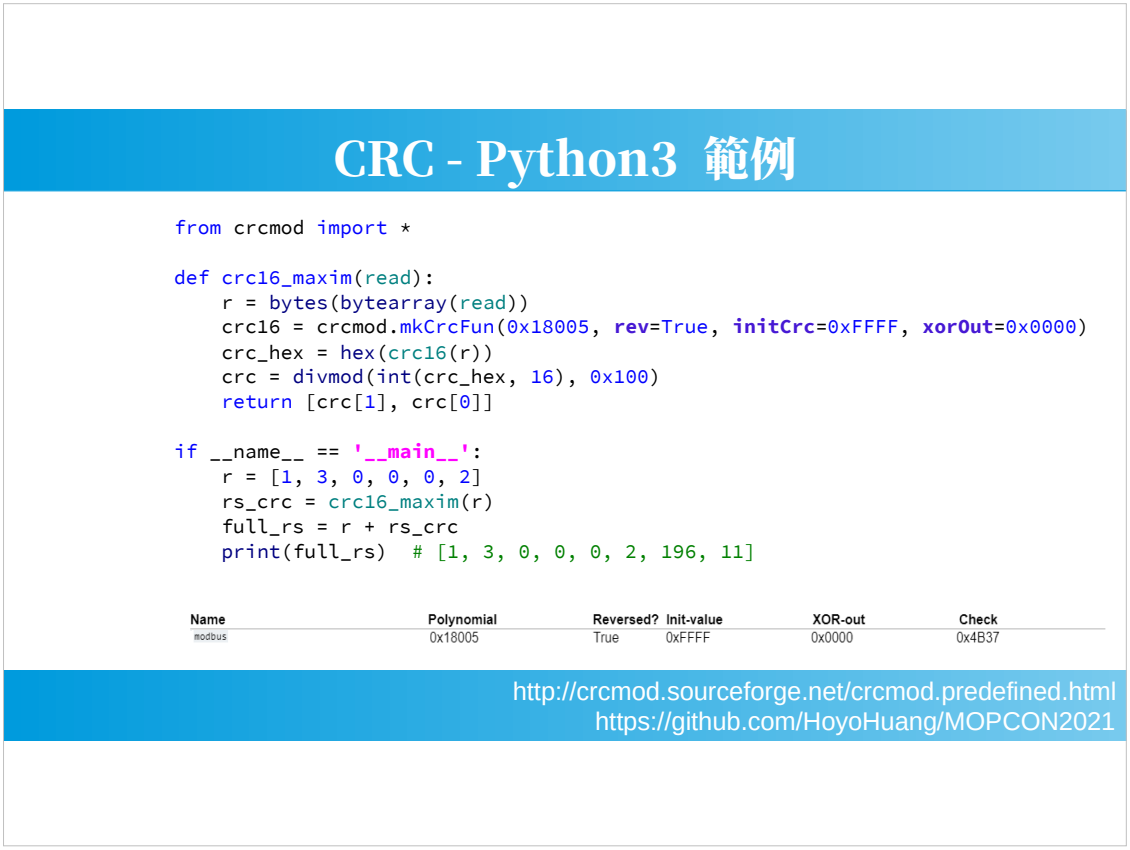
了解為什麼工業控制大力使用 RS485 以及 Modbus ，就要先了解實際的使用場景，實際的環境中，一大堆不同的設備都被裝在同一個空間內，為了有效利用空間，通常不會保留間隙，不同的設備作用不同，有的可能是高頻干擾，例如 Lora ，有的會發出高熱，所以設備本身需要抗干擾之外，通訊也需要穩定以及省空間、省事

以這張範例圖來說，中間那排設備如果都需要通訊，如果使用乙太網路，那就會看到一大把網路線，以及一台 switch hub ，使用 RS485 一台設備就只需要 2 條很細的通訊線

--



這裡要特別說一下 485 串接，兩個設備只要彼此連接，注意不要變成星形拓樸即可，
不過在一些短距離的使用環境，星形拓樸也不一定會有問題，所以在使用上還是相當方便的

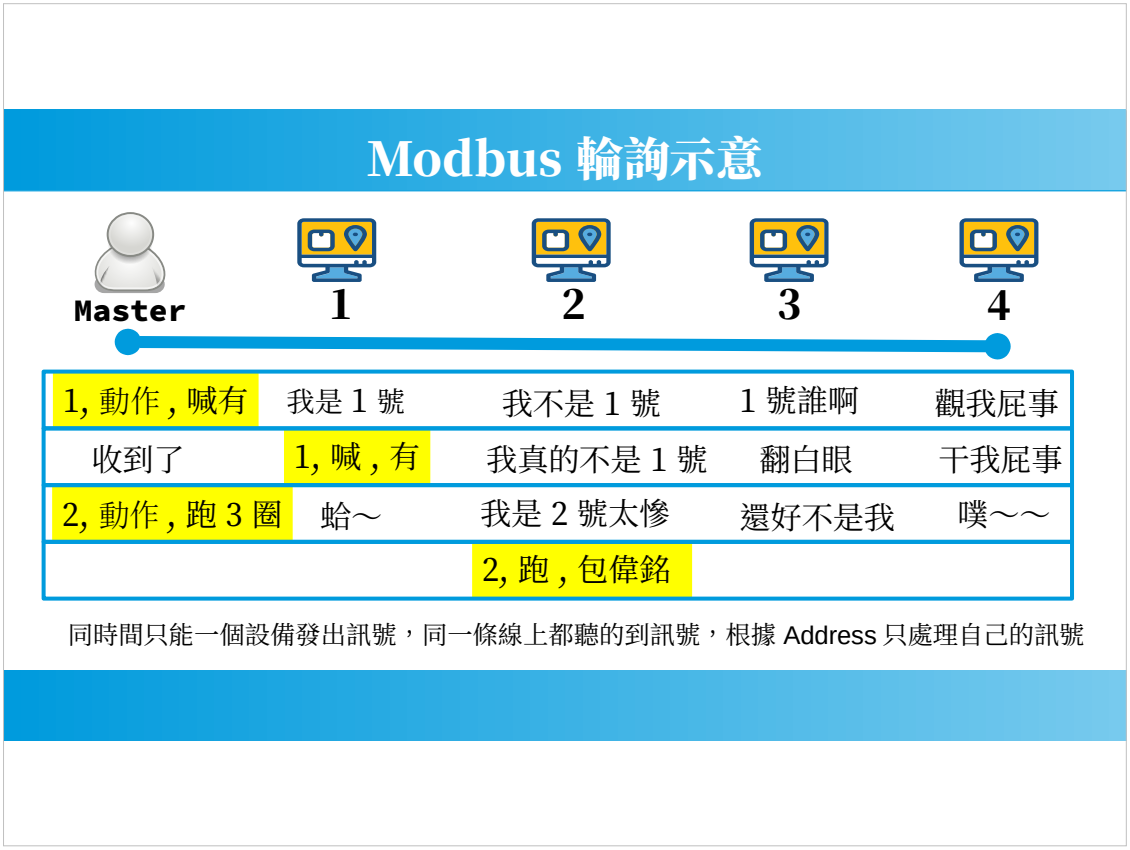


Modbus 第一個難關就是計算 CRC ，這裡以 Python3 舉例，使用 CRCMOD 這個 package 來講解

在 crcmod 的官網說明 predefined 預定義這篇網頁資料可以查到，多項式、初始值、 XOR 輸出設定以及是否需要反轉等設定值，找到 modbus 這行，得到多項式是 16 進制 0x18005 ，需要反轉，多項式是 16 進制 0xFFFF ， XOR 輸出是 16 進制的 0x0000

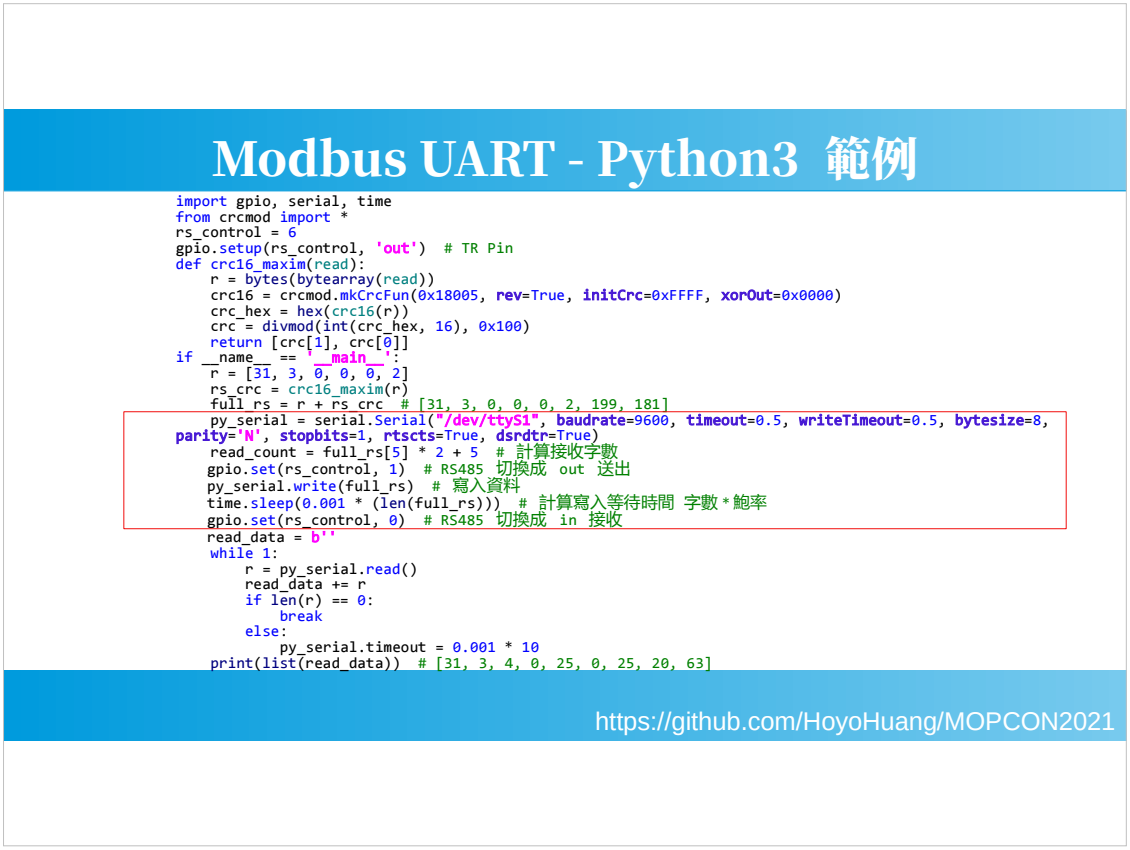
假設今天 RS485 設備 Address 設定為 1 ，要讀取前 2 個 word 的資料就可以使用 list 1,3,0,0,0,2 來執行，拼上 CRC 之後就可以丟給設備執行

使用其他程式實做如果也是使用 1,3,0,0,0,2 這個值，那 CRC 就是 196, 11 ，



RS485 加上 Modbus 的使用場景，是半雙工輪詢，整個流程大概就是像簡報中，由 Master 發出指令，因為是串連，所以全部的設備都會收到訊號，設備從 Address 判斷，是不是該自己處理，處理完成後在靜默時間之後，回傳結果， Master 在收到回應後，即可繼續下一輪

實際使用要注意，雖然表面看起來是 Master 主控，但是因為訊號線是串接，所以訊號線上所有的設備就會收到，實務上，有時指令間必須將間隔時間加長才能確保通訊正常



這段程式才是這個主題的乾貨， RS485 和一般 serial 的程式沒什麼差別，主要差在，因為半雙工的特性，因此需要控制 TR 腳來設定輸出還是輸入

問題絕大多數都是出在 TR 腳控制，切換太快，輸出未完成，切換太慢，輸入讀取缺失

看手冊學習會使用 flush 來判斷寫入輸出是否完成，不過通常控制的都很差，通訊成功率無法到 99% 甚至 100% ，有可能是原來的設計是給全雙工的通訊使用，而不是這種半雙工的應用場景

在等待時間有一個 0.001 乘上字數的公式，字數就看最後送出的 byte 數量，這個沒問題，有問題是 0.001 是怎麼來的？

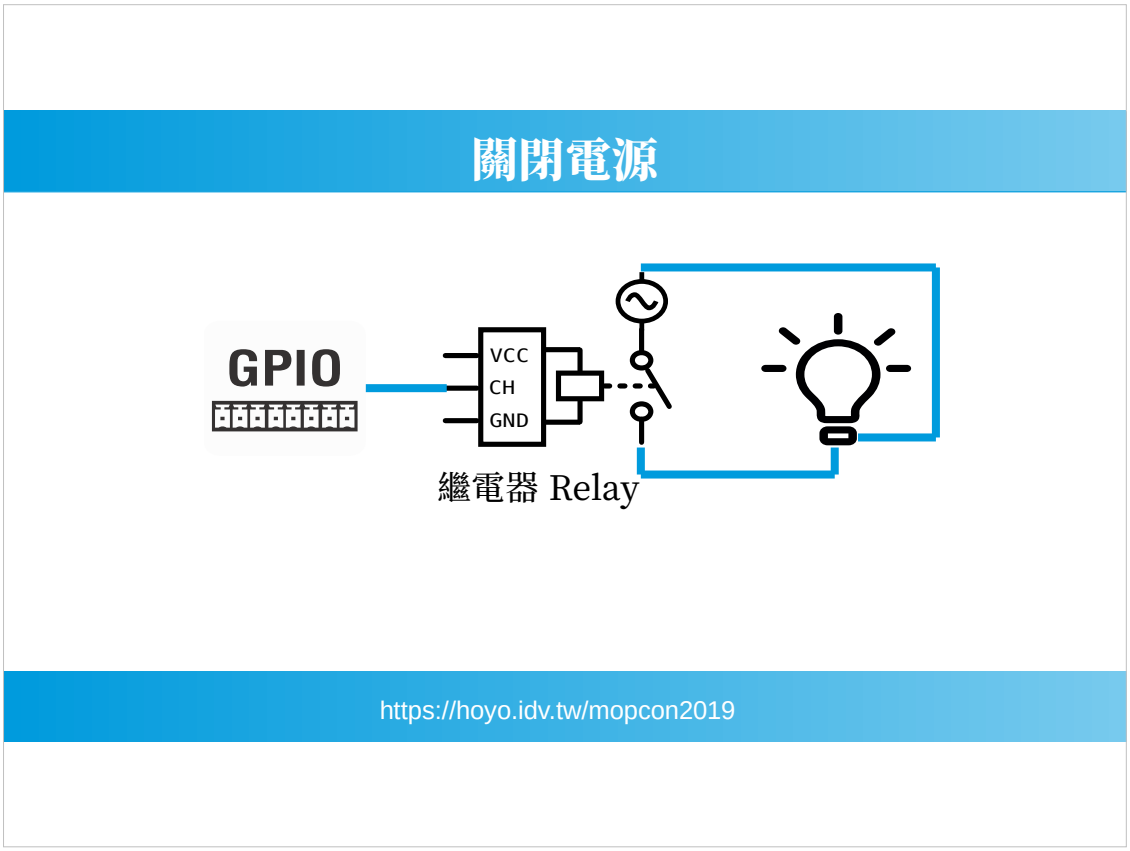
上面可以看到設定鮑率為 9600 N81 的通訊設定，因為是 N81 所以每個 byte 增加一個開始 bit 和一個結束 bit ，最後一個字就是 10 個 bit 鮑率 9600 用 1 秒除，得到 0.000104 ，乘上每次 10bit 傳輸，就等於 0.00104 ，在此只取 0.001 來當作基數，測試沒問題，所以實際上遇到問題時，可以適度還原小數點位數，或是直接加大來調整，但是不能調整太多，會有超時切換，接收不完整的問題，就像 flush 函數一樣的問題

如何省電（卸載負載）- AIoT ：執行

- 關閉電源
- 分段控制（ 降低風速、提高溫度 ）

https://www.moeaboe.gov.tw/ecw/populace/Law/Content.aspx?menu_id=4354

要達到省電的目的，最直接的方法就是關閉電源，隨手關燈救地球
其他方式就很多樣了，以冷氣舉例有改變模式，如冷氣變送風，
冷氣模式時也可以提高溫度來節約用電，降低風速也可以達到目的
政府之前一直提倡的，室內冷氣溫度平均值不得低於攝氏二十六度，就是一種節能的方式



關閉電源的方法是最相對容易的方法，只要讓電源供電經過繼電器之後，再控制繼電器即可達到供電或斷電的目的

繼電器要達到可以控制，最少也要提供一個控制點，這樣就可以使用 Arduino 或是 Pi 這類有 GPIO 的設備來控制，最好是可以提供狀態讀取，前年分享的，米家就是我家使用的是，從網路 JSON 轉換成 GPIO 控制繼電器，就是一種控制電源的方式

只能通電、關電除了太粗暴之外，很多電器或應用開啟電源後也不會動作，例如冷氣或電視，也可以說是主要用遙控器操作的電器，除了通電開啟電源外還需要另外的控制，

底層的控制無非就是數位 I/O 或是類比 I/O ，上層可能是 UART 或是更上層的網路訊號

例如 EMS 就是從 UART 的 TaiSEIA 通訊，轉換成冷氣的控制方法

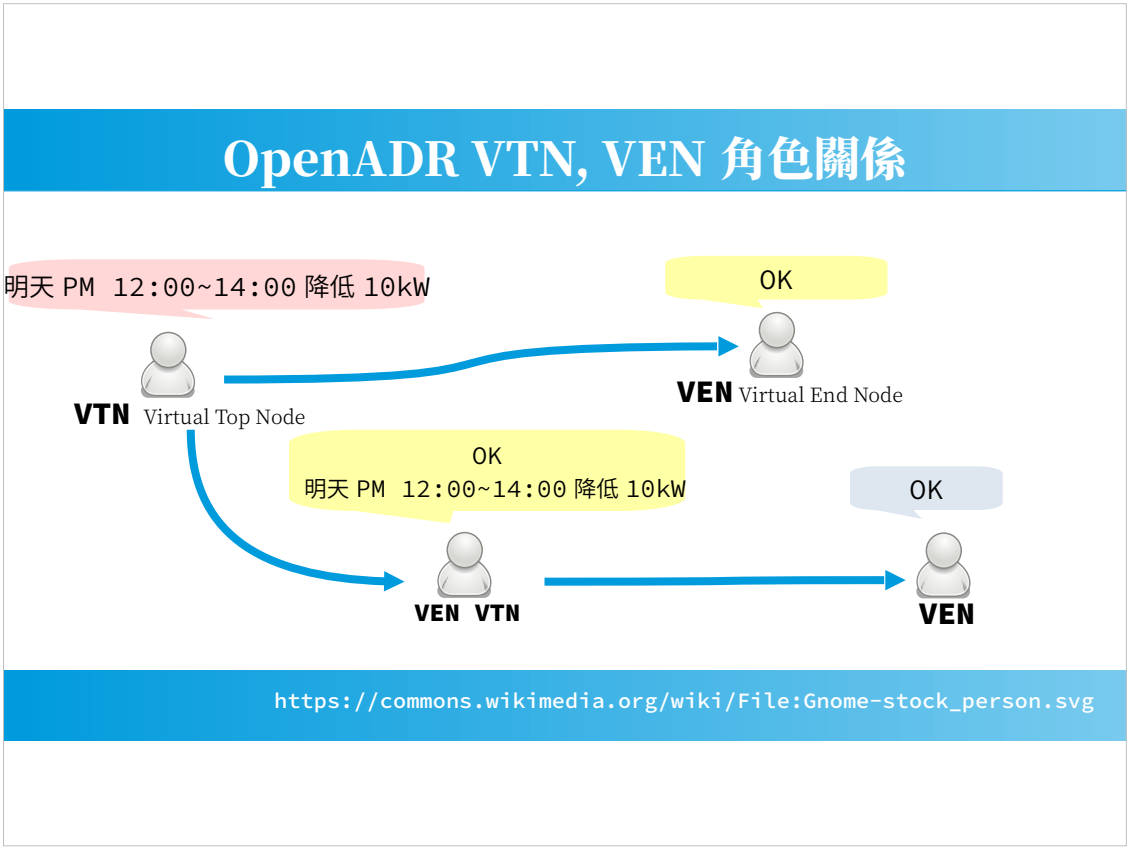
知道需量除了監控，甚至還需要受到控管，這時候就要談到 OpenADR ，OpenADR 是通訊規範後來為了推廣也組織了聯盟，作用是當供電電網需要降低供電需求時，向使用者端發出卸載需求，使用者可使用調高冷氣溫度或是關閉非必要用電，來達到降低用電負載

重點是 OpenADR 只是資訊交流，該如何提高冷氣溫度，或是關閉電源那就要靠自己了，可以自己手動按遙控器、關開關，當然最好的方式就是讓設備自動化，也就是 AIoT

在 EMS 上台電就是擔任了 OpenADR 的 VTN 角色，發送 DR 也就是需量反應給學校使用端也就是 VEN

OpenADR 對於一般使用者學習沒有必須性，不過看在高昂的入會費以及高昂的技術移轉費，有興趣或是想省錢試試看的可以嘗試 GitHub 上的 OpenLEADR 專案，使用 Python 實做

是少數或是唯一可以免費獲得並且可完整使用 VTN 及 VEN 角色的方式



OpenADR VTN VEN 的角色關係

OpenADR VTN 就是發命令的單位， VEN 是末端執行單位， VEN 必須主動到 VTN 回報或取得 OpenADR 事件， VEN 同時也可以是 VTN 角色，類似代理人，收到 ADR 事件再往下轉發到 VEN

什麼又是 TaiSEIA ？

TaiSEIA 是台灣智慧能源產業協會的簡稱

TaiSEIA 101 才是智慧家庭物聯網通訊標準， 2014 年就提出了，一直到 EMS 案子才熱門了起來，因為 EMS 強制規定使用 TaiSEIA 101 控制冷氣

現在 TaiSEIA 也是 CNS 國家標準

TaiSEIA 101 通訊是主從架構， dongle 是主，家電是從，現在就有 TaiSEIA 101 dongle 在賣，有興趣的自行 Google ，有業配需求的已經規劃了廣告位出租位

Dongle 界面是 UART ，鮑率是 9600

將控制變成一致性的標準之後就可以輕鬆實現智慧家庭，只要購買的家電支援 TaiSEIA 101 ，理論上…

TaiSEIA 智慧家庭產品插拔大會

- 黃腔是官方開的！
- 威利在哪裡？

https://www.facebook.com/Taiseia2014/posts/802898833663779

TaiSEIA 台灣智慧能源產業協會因應 EMS 的到來，協助我們這些提供通訊 dongle 的廠商舉辦了產品相容測試活動

官方根據活動的性質和實際的操作貼心的叫做「插拔大會」，不過看了現場全部都是男性工程師的情況，我覺得官方滿滿的惡意 :)

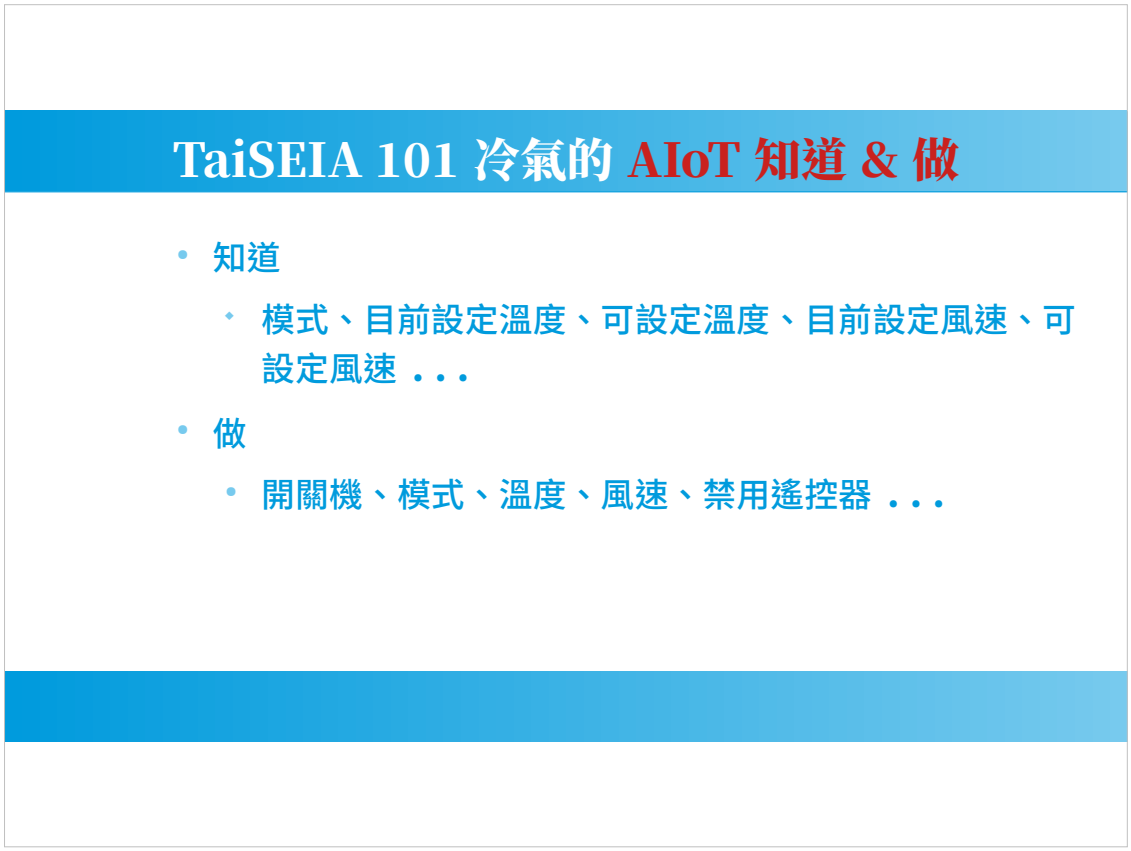
照片內有一位本公司的型男，看他撐著桌子不知道菊花有沒有開花

TaiSEIA 101 智慧家庭產品驗證標章

TaiSEIA 101 可以控制什麼

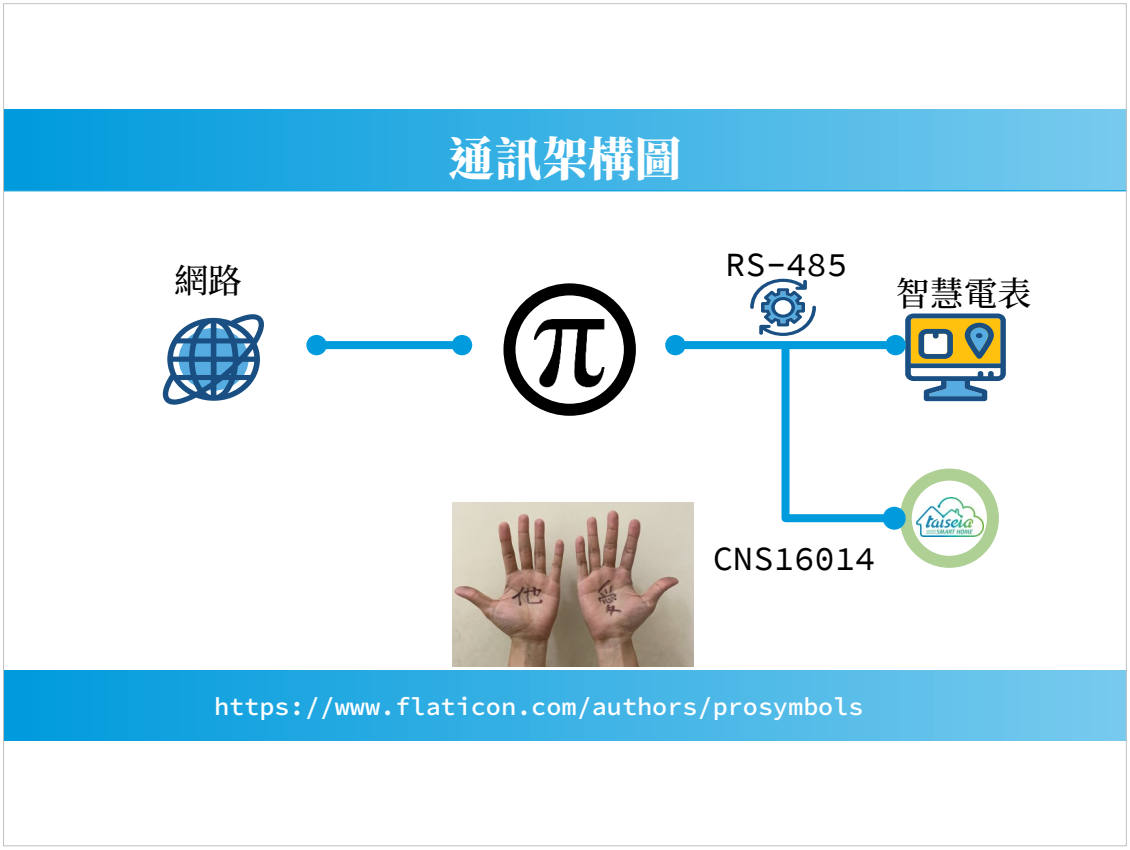
TaiSEIA 101 目前參與認證的類型不多，例如就沒有電視機，也沒有機上盒，Hoyo 就很需要機上盒因為遙控器品質差，買又貴，我已經不想繼續使用鋁箔紙修理遙控器了 :)

現在還是以冷氣機居多



所以我們知道 AIoT 的知道和做，就是當需要控制負載時，先知道冷氣目前的狀態，進而控制冷氣

控制器可以控制的功能，大部分也都可以獲取資訊或控制，有些還支援更詳密的資訊，如室內機、室外機消耗的電流，特別的是，部份機種還可以設定禁用遙控器



前面講到了一邊是 OpenADR 的網路資訊，另一邊則是 TaiSEIA 101 的 UART 訊號，以及溝通智慧電表的 RS485 ，
要把兩邊串起來就需要一個有網路又有 UART ，還要有可轉換成 RS485 的硬體界面，
除了以上的硬體需求，因為是 IoT 架構，可能是一個電表或一台冷氣就要一個介接的設備，所以成本要足夠合理才行，這裡我們選擇 Pi ，一點都不為難

要特別說明， EMS 案子的 IPC 不是使用 Pi 而是使用工業電腦
選 Pi 的原因是為了迎合主題為愛鼓掌啪啪啪的需求 XD

Pi 是 AIoT 負責判斷及自動執行的硬體，為什麼選 Pi ？如果你只知道樹莓派，那為了成本考量是不太可能選 Pi 的，實際上 Pi 的選擇眾多，因為 IoT 盛行，廠商出了很多 4cm*4cm 大小，含網路功能的 Pi ，疫情前 10 美金就有，現在漲很多，網拍買會貴

1GHz CPU,Ethernet,SD 卡 ,256MB 記憶體的組合足以應付大多數不需要圖形界面的使用需求

台幣 400 就具備 Ethernet 是 MCU 很難擁有的成本，非常有競爭力

Pi 的優勢

Pi 使用 ARM 架構，也就是軟體只要 for ARM 就可以使用，大部分的軟體都有，系統方面 Linux,FreeBSD,Android,Windows 根據不同的 CPU 以及廠商有不同的包裝，甚至以後可能還有 MAC 可以用，軟體方面常用的 Apache,PHP,Python 都有，資料庫沒有 MySQL ，改用 MariaDB 或 MongoDB 其他資料庫，影響不大

硬體規格方面就算是 10 美金的 Pi ，規格也是遠勝 MCU

RK3568 或是樹莓派 4 都已經有 8GB 記憶體，圖形界面、觸控螢幕、 WiFi 、 USB 啟動系統或是 SD 卡儲存界面，效能及規格甚至比一些文書電腦高，更別說驍龍 888 了

Pi 應用多是獨立的產品，不是放在身邊當作主機，因此產品化的考量和一般就有不同，學習 Linux 都知道可以把 partition mount 成唯讀，問題是在正式主機誰會把 / 或是其他系統 partition mount readonly ，可是在 Pi 上就有可能會是這是產品化的常態，因為大部分的 Pi 使用 SD 當作主要處存媒體，為了保護 SD 卡 mount readonly 就是一種選擇方案

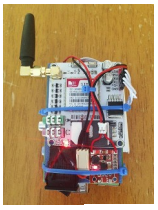
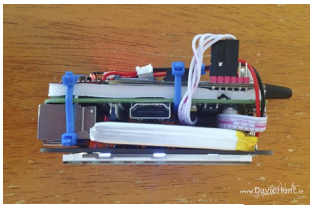
Pi Linux 和 x86 Linux 差異

- Watch Dog (/dev/watchdog)
- GPIO (/sys/class/gpio/*)
- SPI, I²C
- UART

https://www.163.com/dy/article/E4J570EC05148K5G.html

Pi for ARM 相較 x86 少了一些軟體，卻多了一些特別的硬體，
看門狗 Watch Dog 的作用是當啟用後，必須定時塞資料到 /dev/watchdog 這個檔案，這個動作叫做餵狗，不是餵狗糧，這裡沒有人放閃
如果在設定時間內，狗沒吃到狗糧，那看門狗就會自動重新開機，其實軟體也可以設計相同的機制，只是沒有硬體保險
接下來就是板子上那一堆密密麻麻的接腳，最常見以及討論的就是 GPIO ，也就是輸出、輸入的數位接腳，另外還有 SPI 和 I2C 接腳適合其他通訊設備的連接，
UART 腳位在沒有 HDM 或 LVDS 等其他螢幕規格情況之下，可以使用 TTL 轉 USB ，接上電腦後打開 putty 就可以顯示操作畫面，使用起來就和 ssh 一樣，差別只在有電源就有畫面，也不需要登入就像是真正的螢幕
光是有 GPIO 可以控制一些數位 IO 其實就很方便了

你有 iPhone 他有 PiPhone ；你有 GameBoy 他也有 PiBoy

http://www.davidhunt.ie/piphone-a-raspberry-pi-based-smartphone/

https://www.experimentalpi.com/Retro-Arcade_c_12.html
https://forum.odroid.com/viewtopic.php?f=187&t=41283

第一次接觸 Arduino ，面對 Arduino 的生態，最常做的就是智能插座、智能溫濕度計，比較有錢的會製作自走車， Hoyo 只做的出來智能插座這個等級
但是第一次面對 Pi 的生態，剛剛說過，有觸控螢幕、 Linux 完整作業系統、圖形界面、 SD 卡儲存界面、 WiFi 藍芽有些型號也有，後來還知道有 3G, 4G 模組，哇～～～ 這樣組起來就是一台手機了啊，我要變成賈伯斯第二，我這麼聰明、機智，上網搜尋一下，太陽底下沒有新鮮事， 2014 年已經有人做出來了，看到 Linux 上有家用遊戲主機模擬器，加上幾個按鈕接 GPIO ，這不就是掌上遊樂器，啊～～～這個也有人做了，總之， Pi 足夠強大的硬體規格加上擴充性，可想像的空間很大，這裡只是冰山一角， Hoyo 下一台網站主機就想要使用 RK3399 或是樹莓派 4 來當作主機，不使用桌上電腦了，省空間又省電

Pi 的劣勢

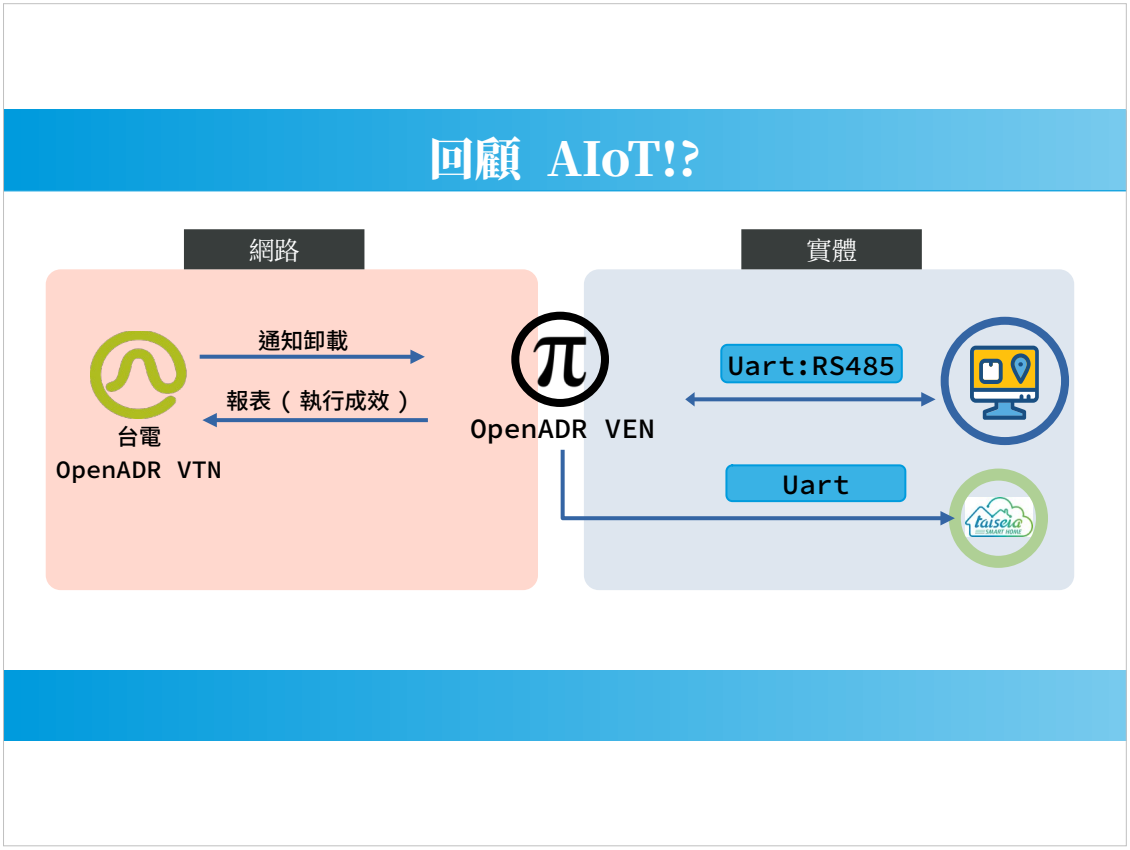
- Linux 資訊安全
- 直譯式程式（Python,PHP）商品化程式碼保護
- SD 卡故障（ 寫入或電源因素？ ）

Pi 雖然前面說了那麼多優點不過也不是完全沒缺點， Pi 的劣勢首先就是如果為了商用版權問題，而採用 Linux ，如果沒有特別調整系統，預設可能安裝了很多用不到的服務及帳號，這些服務及帳號就有可能會造成系統漏洞
另外就是因為效能允許，貪圖方便可能就用直譯式程式開發產品，這樣被好奇的小夥伴拿到系統的帳號、密碼就會外流，程式預先編譯保護程式碼就很重要，反則就有可能被當成 DDoS 肉雞大軍的一員，偏偏 Pi 的系統更新又無法像 MCU 的 OTA 那麼全面方便
大部分的 Pi 都有 SD 界面，提供系統主要或是唯一啟動方式， SD 卡在使用上有寫入次數限制，如果是零軌整個系統就無法運作，不過前面有說到可以將系統設定為唯讀，所以寫入就沒問題了，實際的問題是，實務經驗上有遇到因為電源不穩定或忽然斷線因素，造成檔案直接損壞問題，無法穩定重現，所以原因不明不知道問題在那

Pi 有眾多選擇，如果換了一種 Pi ，例如樹莓派換到 NanoPi ，就要重新選擇系統以及設定方法，那學習效率就比較不理想，有些 Pi 發行版提到有可以使用 DietPi 這類第三方的作業系統，說是適合進階者，不過在 Hoyo 的測試之下，反而是初學適合選擇 DietPi 或是 armbian ，因為就算換了 Pi ，還是相同邏輯的系統架構，以及設定方法。

主要的設定也都是使用對話框互動操作，不需要像原來的 Linux 管理系統一樣，輸入指令或編輯設定檔。

在更新方面，有些產品原廠可能維持很久不更新，可是 armbian 目前有支援的 ARM 都已經更新到 kernel 5.13
這兩者對應不同的設計理念， armbian 是盡量在提供完整的功能下，又不佔用過多的儲存空間及硬體資源，但是又不像預設的 Debian 那麼無聊，例如第一次開機就可以選擇 zsh 的 shell ，這樣就可以使用美美的指令界面， DietPi 是拿 armbian 來修改，主要精神是極簡，減輕系統的負擔，降低硬體的使用率，所以使用 DietPi 遇到問題時，有時也要去找是否是 armbian 的問題，例如最近的 DietPi7.6 在 NanoPi 沒有 UART ，才發現要到 armbian 的設定檔去啟用



最後回顧，大部分的工業控制或是 IoT 設備為了成本，只提供了基礎功能以及通訊界面，為了通訊我們需要一個相對精簡，但是功能齊全的設備，來溝通應用以及控制的橋樑

，我們選擇了 Pi ，除了有完整的硬體規格外，足夠快 CPU 運算以及足夠大的儲存空間，讓實務應用沒有侷限

在 EMS 班班有冷氣的應用中，也可以使用 Pi 來連接 OpenADR 以及智慧電錶和 TaiSEIA 的電器控制

此簡報使用 LibreOffice ，以及 7zip ，重新錄製第 3 次影片時遇到簡報檔案壞掉，有人比我慘嗎？還好 7zip 可以將文字檔案複製出來，更新到之前的舊版本備份，還原了文字內容，不然我就 GG 了

字型使用思源字型

錄製軟體 OBS ，影片剪輯使用剪映，本來想要使用智能辨識字幕功能打上字幕的，結果 Hoyo 這個口齒不清辨識出來一堆錯字，還是後面提供逐字稿就好吧

其他使用的網路資源，或是來源已經附在原簡報下方

我們用 OpenADR 加上 TaiSEIA
冷氣溫度調高了，需量沒有超約
但這一切值得嗎？

感謝聆聽，希望本次議題對大家有所收穫，終於講完了，趕快來囤積蠟燭比較實在，全部有14個梗，你找到或知道幾個？喜歡我的影片不要忘了一鍵三連，開啟小鈴鐺，我是Hoyo，我們下次見