

BMW Lab

Design Document

燒機室監控系統

底端程式

作者：黃智威

文件修改紀錄 (Document Revision History)

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人 | 日期 | 版本 | 備註 |
| 黃智威 | 10/19 | 1.0 |  |
|  |  |  |  |
|  |  |  |  |

目錄

[一、系統規格 4](#_Toc464821931)

[1. 系統架構圖 4](#_Toc464821932)

[2. 規格說明 4](#_Toc464821933)

[2.0燒機室資料蒐集 4](#_Toc464821934)

[2.1 gayway數據蒐集 5](#_Toc464821935)

[2.2連線狀態檢測 5](#_Toc464821936)

[2.3運轉操作 5](#_Toc464821937)

[2.4設定溫度 5](#_Toc464821938)

[2.5遠端config 5](#_Toc464821939)

[二、程式解說 6](#_Toc464821940)

[1.程式流程 6](#_Toc464821941)

[2.function 介紹 7](#_Toc464821942)

[2.1 副程式：CheckandReconnect 7](#_Toc464821943)

[2.2 副程式：datacollectionTask 8](#_Toc464821944)

[2.3 副程式：SearchHardWare 9](#_Toc464821945)

[2.4 副程式：AddSensor 9](#_Toc464821946)

[2.5 副程式：RestartProgramService 11](#_Toc464821947)

[2.6 副程式：SetTemperatureService 11](#_Toc464821948)

[2.7 副程式：SwitchService 11](#_Toc464821949)

# 一、系統規格

## 1. 系統架構圖

圖一、燒機室監控系統架構圖

燒機室底端系統架構圖如圖一所示，本系統分成讀取部分和上傳部分，其規格分述如下：

## 2. 規格說明

### 2.0燒機室資料蒐集

* 此部分是透過Modbus取得燒機室內部資料，溫度會分成設定溫度和實際溫度，運轉狀態分成啟動中止停止三種狀態。

### 2.1 gayway數據蒐集

* 利用指令將gateway內部CPU、Memory、HDD數值取出後，將它做計算得出每個的使用率。

### 2.2連線狀態檢測

* 利用Thingwox的已有的Function code檢查和遠端server是否斷線，並進行處理。

### 2.3運轉操作

* 控制機器啟動、中止和停止

### 2.4設定溫度

* 輸入溫度和上升斜率，設定目標溫度

### 2.5遠端config

* 輸入COM,ID,Address,BaudRate,ControlMode後測試該address是否有在線上。

# 二、程式解說

## 1.程式流程

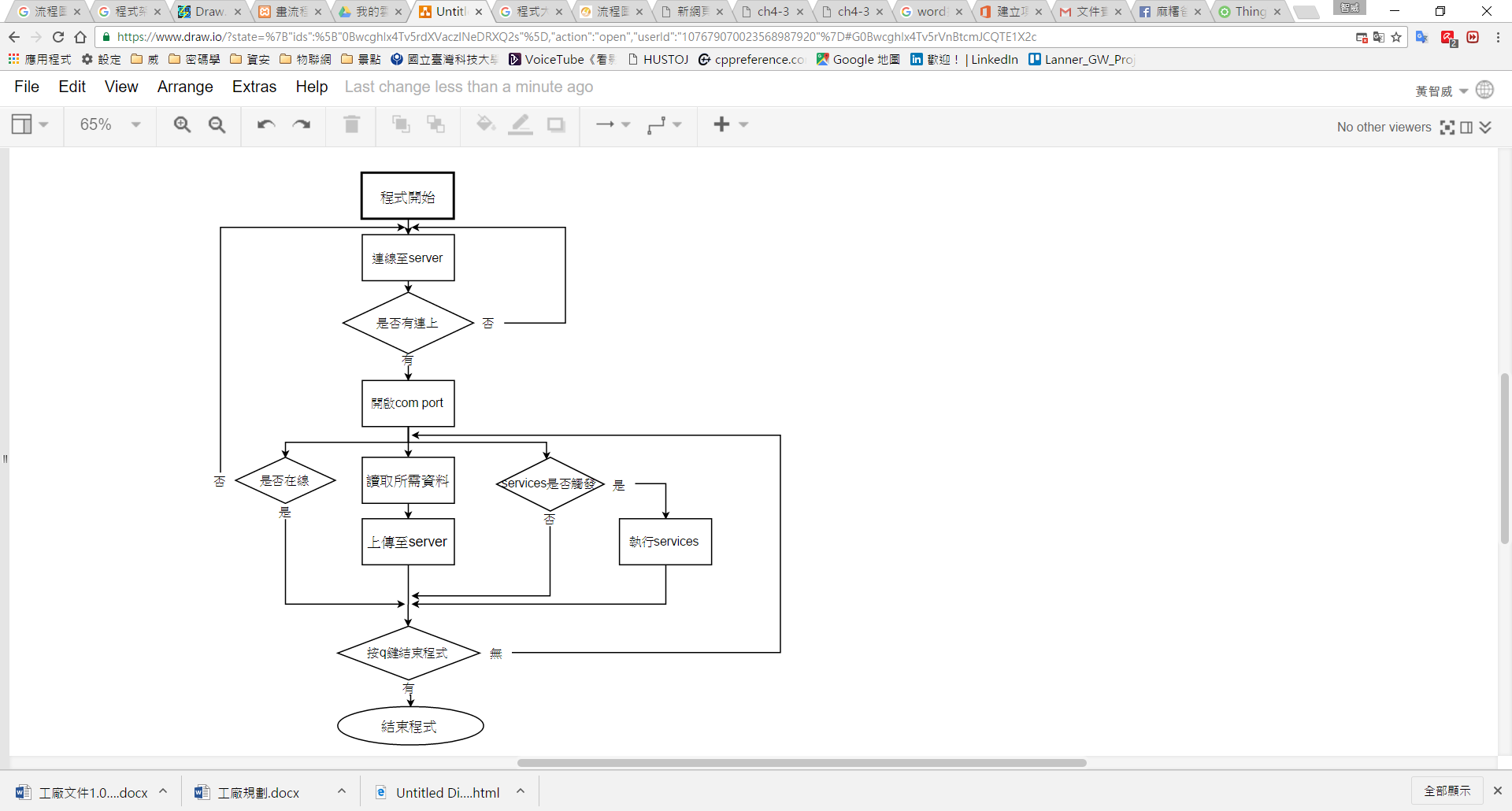


圖2 程式流程圖

## 2.function 介紹

|  |  |  |
| --- | --- | --- |
| 類型 | 名稱 | 簡介 |
| 收值 | CheckandReconnect | 利用thingwox內建API讓gateway重新對server連線，用於斷線時。 |
| datacollectionTask | 對modbus下指令將回傳值作轉換成為我們需要的數據在上傳至server。 |
| SearchHardWare | 利用linux指令得出各單位使用量。 |
| services | AddSensor | 檢查或新增新的sensor。 |
| multiServiceHandler | 多重Serivce用在無罰回傳的service。 |
| RestartProgramService | 重開程式。 |
| SetTemperatureService | 設定溫度。 |
| ShutdownTask | 關機。 |
| SwitchService | 設定機器啟動、終止和停止。 |
| 連線 | AuthEventHandler | Thingwox提供的function用來連線 |
| BindEvenHandler | Thingwox提供的function用來連線 |
| 校驗 | FCS\_cacluate | 每一串modbus指令皆要有校正碼，此為計算方式。 |
| 其他 | setupSerialPort | 開啟com port，並能作基礎設定 |

### 2.1 副程式：CheckandReconnect

* Gateway有時候會因為一些原因像是網路不穩而造成斷線的問題，因此利用這個function反覆地去檢查連線，就能確保連線狀態。

|  |  |  |
| --- | --- | --- |
| 重要function | twApi\_Connect() | 重新向server連線 |
| twApi\_isConnected() | 確認是否和server有連線 |
| 重要參數 | ret | 存放連線function出來的值 |

### 2.2 副程式：datacollectionTask

* 此程式主要是向modbus取資料，再將他整理好上傳至server
* For迴圈從1到3是因為總共有三台人機介面因此做三次迴圈之後就可取完所有值
* 由於需要的數值包含在三個function code裡所以迴圈裡必須傳入三次指令三次

|  |  |  |  |
| --- | --- | --- | --- |
| 重要參數 | FCS=0; | int | 檢查FCS是否正確 |
| Tempture | double | 實際溫度 |
| SetTempture | double | 設定溫度 |
| ColdBalance | double | 冷卻器功率 |
| HotBalance | double | 加熱器功率 |
| slop | int | 斜率 |
| Status | Char[3] | 狀態 |
| request | unsigned char[9] | 指令 |

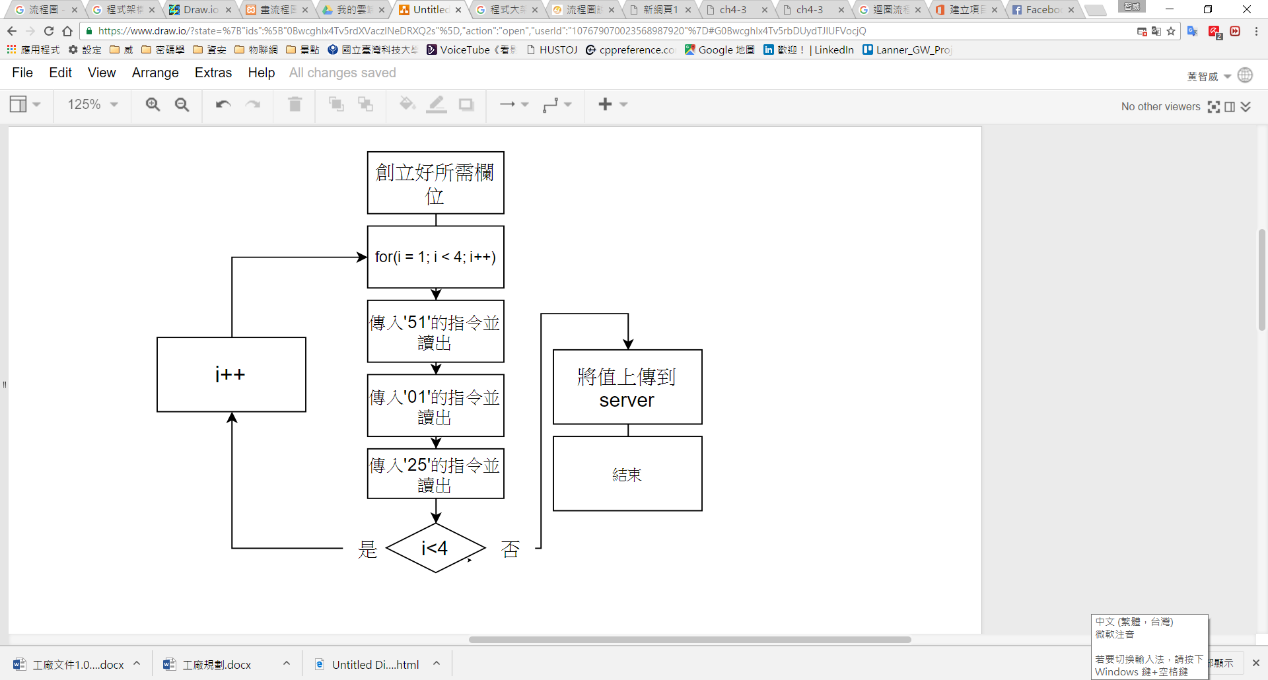


圖3 datacollectionTask流程圖

### 2.3 副程式：SearchHardWare

* 算CPU使用量先取兩個採樣點，然後計算其差值，也就是將下面兩個total互減即可。

total\_1 = usr\_1 + nic\_1 + sys\_1 +idle\_1 + iowait\_1+irq\_1+softirq\_1+steal\_1;

total\_2 = usr\_2 + nic\_2 + sys\_2 +idle\_2 +iowait\_2+irq\_2+softirq\_2+steal\_2;

* 算Memory時可先從linux指令中得到memtotal和memfree，再來將兩數相減再除total即可。

hardware.Memory = (float)((memtotal - memfree)) / (float)memtotal;

|  |  |  |  |
| --- | --- | --- | --- |
| 重要參數 | usr\_1 | unsigned long long | 用戶態的CPU時間 |
| nic\_1 | unsigned long long | 進程所佔用的CPU時間 |
| sys\_1 | unsigned long long | 核心時間 |
| idle\_1 | unsigned long long | 其它等待時間 |
| iowait\_1 | unsigned long long | 硬盤IO等待時間 |
| irq\_1 | unsigned long long | 硬中斷時間 |
| softirq\_1 | unsigned long long | 軟中斷時間 |
| steal\_1 | unsigned long long | 內存 |
| total\_1 | unsigned long long | 全部時間 |
| memtotal | unsigned long long | 全部memory |
| memfree | unsigned long long | 可用memory |
| fp | FILE \* | 將資料佔存在此file檔裡 |

### 2.4 副程式：AddSensor

* 在server端會將COM,ID,Address,BaudRate,ControlMode傳下來，此程式是將資料轉成指令，對modbus做單次傳輸，如果正確值就回傳正確。

|  |  |  |  |
| --- | --- | --- | --- |
| 重要參數 | ret | int | modbus傳入和傳出的數量 |
| value | int | 存放config成功回傳的數值 |
| MachineNum | int | 切換modbus模式 |
| Com | double | Port編號 |
| ID | double | Modbus ID |
| Addr | double | Modbus Address |
| Brate | double | Modbus baudrate |
| request | unsigned char[25] | 指令 |
| response | unsigned char[256] | 回傳值 |

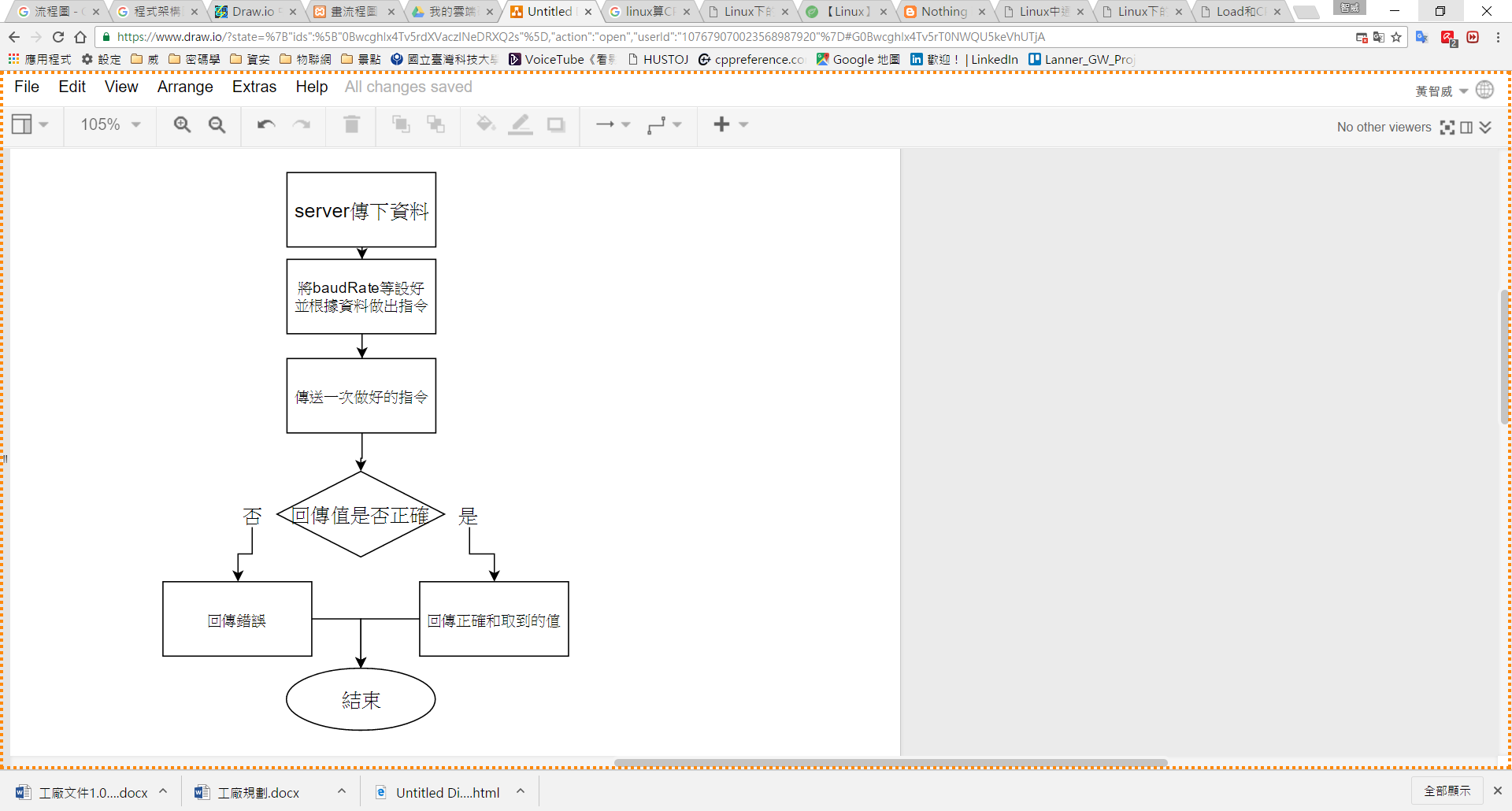


圖4 AddSensor流程圖

### 2.5 副程式：RestartProgramService

* 當電腦不穩的時候，可進行重開程式，將程式保持在最穩當的時候。
* 剛開始必須先fork出一個子程序，再利用execlp呼叫自己，再把母程序關掉，這樣即可重開。

if (fork() == 0) { //創出子程序

:

execlp(buf, buf, (char \*)0); //呼叫自己

:}

exit(EXIT\_SUCCESS); //關閉原本母程序

### 2.6 副程式：SetTemperatureService

* 利用server傳下來Address、上升斜率和溫度，組成一個指令，利用modbus傳送給燒機室，如果不是正確回傳值，會有重傳機制進行重傳。

### 2.7 副程式：SwitchService

* 利用server傳下來的address和狀態組成一個指令，再用modbus傳送給燒機室，而狀態分三種，啟動、中止和停止，每個不用同的數字代表。