電通二甲微處理器實驗 實驗結報

|  |  |  |  |
| --- | --- | --- | --- |
| 實驗名稱 | Lab 11 – MCS Cloud | | |
| 組別 |  | 組員 | 吳東燁 |

1. 實驗目的

到MCS註冊帳號

建立網頁應用程式（web application）

透過MCS REST API進行溝通

LinkIt 家族開發板、手機、瀏覽器

1. 實驗步驟

建立Prototype

依照所需功能，新增Data channel，記下Data channel Id

建立Test device，記下DeviceId與DeviceKey

MCS 控制 LED 閃爍

CHECKPOINT1  
修改 MCS\_GET code, 加入 MCS Remote

CHECKPOINT2  
將 A0 電阻值上傳

1. 程式碼

|  |
| --- |
| #include <LWiFi.h>  #include <WiFiClient.h>  #include "MCS.h"  // Assign AP ssid / password here  #define \_SSID "HTC Portable Hotspot B047"  #define \_KEY "0975621919"  // Assign device id / key of your test device  MCSDevice mcs("D0LzAX2G", "1Q05WC6c96QSoCT0");  // Assign channel id  // The test device should have 2 channel  // the first channel should be "Controller" - "On/Off"  // the secord channel should be "Display" - "On/Off"  MCSControllerOnOff led("123");  MCSDisplayOnOff remote("456");  MCSDisplayInteger Temp("Temp");  #define LED\_PIN 7  void setup() {a  // setup Serial output at 9600  Serial.begin(9600);  // setup LED/Button pin  pinMode(LED\_PIN, OUTPUT);  pinMode(A0, INPUT);  // setup Wifi connection  while(WL\_CONNECTED != WiFi.status())  {  Serial.print("WiFi.begin(");  Serial.print(\_SSID);  Serial.print(",");  Serial.print(\_KEY);  Serial.println(")...");  WiFi.begin(\_SSID, \_KEY);  }  Serial.println("WiFi connected !!");  // setup MCS connection  mcs.addChannel(led);  mcs.addChannel(remote);  mcs.addChannel(Temp);  while(!mcs.connected())  {  Serial.println("MCS.connect()...");  mcs.connect();  }  Serial.println("MCS connected !!");  // read LED value from MCS server  while(!led.valid())  {  Serial.println("read LED value from MCS...");  led.value();  }  Serial.print("done, LED value = ");  Serial.println(led.value());  digitalWrite(LED\_PIN, led.value() ? HIGH : LOW);  }  void loop() {  // call process() to allow background processing, add timeout to avoid high cpu usage  Serial.print("process(");  Serial.print(millis());  Serial.println(")");  mcs.process(100);    // updated flag will be cleared in process(), user must check it after process() call.  if(led.updated())  {  Serial.print("LED updated, new value = ");  Serial.println(led.value());  digitalWrite(LED\_PIN, led.value() ? HIGH : LOW);  if(!remote.set(led.value()))  {  Serial.print("Failed to update remote");  Serial.println(remote.value());  }  }  if(!Temp.set(analogRead(A0)))  {  Serial.print("Failed to update remote");  Serial.println(Temp.value());  }    // check if need to re-connect  while(!mcs.connected())  {  Serial.println("re-connect to MCS...");  mcs.connect();  if(mcs.connected())  Serial.println("MCS connected !!");  }  } |

1. 實驗結果及分析

有亮有反應

1. 心得討論

這次實驗只要了解程式意思以及如何連接MCS Cloud 大致上就沒難度了

1. 修正電路圖

