



LinkIt 7697 for Arduino

環境設定 >

開發指南 ▼

GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

Software Serial

[Resources](#) / [LinkIt 7697 for Arduino](#) / ... / [MCS 函式庫 API 使用手冊](#) / [MCSDataChannel 相關類別](#)

MCSDataChannel 相關類別

藉由繼承自 `MCSDataChannel` 類別的資料通道實體，可用來與 MCS 伺服器上對應的資料通道進行傳送與接收資料點的操作。

例如下例的開/關控制器通道：

該通道可被表示為一個 `MCSControllerOnOff` 類別的實體：

```
MCSControllerOnOff channelOnOff("channel_id_1");
device.addChannel(channelOnOff);           // device is an instance of MCSDevice or MCSLiteDevice.
```





LinkIt 7697 for Arduino

環境設定



開發指南



GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

Software Serial

```
device.connect();
bool controllerValue = channel.value();
```

雖然隨著種類不同而有不一樣的資料格式，但每種資料通道都提供了 **set()** 與 **value()** 這兩個函式用來存取資料數值。下面章節將列出在 **Arduino sketch** 中所有可用的資料通道類別。

資料通道有兩個主要的子類別：**控制器通道**類別與**顯示器通道**類別。例如 **MCSControllerOnOff** 和 **MCSControllerInteger** 的**控制器通道**用來表示 MCS 上的控制器通道；而 **MCSDisplayFloat** 和 **MCSDisplayString** **顯示器通道**則對應到 MCS 上的顯示器通道。

建構函式

▼ **MCSDataChannel()**

產生一個可加進 **MCSDDevice** 或 **MCSLiteDevice** 裝置實體的資料通道。

Syntax

MCSControllerOnOff *dataChannel*(data_channel_ID)

MCSDisplayOnOff *dataChannel*(data_channel_ID)

MCSControllerCategory *dataChannel*(data_channel_ID)

MCSDisplayCategory *dataChannel*(data_channel_ID)

MCSControllerInteger *dataChannel*(data_channel_ID)

MCSDisplayInteger *dataChannel*(data_channel_ID)

MCSControllerFloat *dataChannel*(data_channel_ID)

MCSDisplayFloat *dataChannel*(data_channel_ID)

MCSControllerHex *dataChannel*(data_channel_ID)

MCSDisplayHex *dataChannel*(data_channel_ID)

MCSControllerString *dataChannel*(data_channel_ID)

MCSDisplayString *dataChannel*(data_channel_ID)

MCSControllerGPS *dataChannel*(data_channel_ID)





LinkIt 7697 for Arduino

環境設定 >

開發指南 v

GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

MCSDisplayGPS *dataChannel*(data_channel_ID)MCSControllerGPIO *dataChannel*(data_channel_ID)MCSDisplayGPIO *dataChannel*(data_channel_ID)MCSControllerPWM *dataChannel*(data_channel_ID)MCSDisplayPWM *dataChannel*(data_channel_ID)MCSControllerAnalog *dataChannel*(data_channel_ID)

Parameters

data_channel_ID : 在 MCS 伺服器上建立的資料通道 channel id。

Returns

dataChannel 即為繼承自 **MCSDataChannel** 類別的子類別實體。

MCSDataChannel 函式

所有的資料通道種類皆支援下列函式：

▼ **updated()**

檢查資料通道是否有接收到新的資訊點。在進行檢查之前，開發者需要先呼叫 *device.process()* 函式處理來自 MCS 伺服器的指令。若該資料通道有多次更新，則只有最後 (即最新的) 一筆資料會被保留下來。

Syntax

dataChannel.updated()

Parameters

無

Returns

若此資料通道有新的資料點更新，則回傳布林值 **true**；反之則回傳 **false**。



LinkIt 7697 for Arduino

環境設定 >

開發指南 ▾

GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

Software Serial

▼ `valid()`

檢查該資料通道欲接收 / 傳送的值是否為有效值。

Syntax

`dataChannel.valid()`

Parameters

無

Returns

若此資料通道的值為有效值，則回傳布林值 `true`；反之則回傳 `false`。

▼ `value()`

取得此資料通道的數值 (MCSControllerGPS 與 MCSDisplayGPS 類別除外)。

Syntax

`dataChannel.value()`

Parameters

無

Returns

該資料通道的值。其資料類型依各資料通道而定，例如整數、浮點數、字串等。

顯示器通道函式

▼ `set()`

設定此顯示器通道的數值。

Syntax

`dataChannel.set(value)`



LinkIt 7697 for Arduino

環境設定 >

開發指南 v

GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

Software Serial

MCSDisplayPWM.set(value, period)

MCSDisplayGPS.set(latitude, longitude, altitude)

Parameters

dataChannel：繼承自 MCSDataChannel 類別的實體，如 MCSDisplayOnOff 和 MCSDisplayCategory...

MCSDisplayPWM：MCSDisplayPWM 類別的實體。

MCSDisplayGPS：MCSDisplayGPS 類別的實體。

value：欲設定至該資料通道的值。其資料類型依各資料通道而定，例如整數、浮點數、字串等。

period：指定給 PWM 資料通道的整數週期值，其值為 PWM 頻率的倒數。

latitude、**longitude**、**altitude**：GPS 資料通道的座標值，其資料類型為浮點數。

Returns

若該資料通道的數值被成功更新至 MCS 伺服器，則回傳布林值 **true**；反之則回傳 **false**。

控制器通道函式

› **setServerValue()**
設定此控制器通道的數值。

Syntax

controllerChannel.setServerValue(value)

MCSControllerGPS.setServerValue(latitude, longitude, altitude)

Parameters

controllerChannel：繼承自 MCSDataChannel 類別的實體，如 MCSControllerOnOff 和 MCSControllerCategory...

MCSControllerGPS：MCSControllerGPS 類別的實體。

value：欲設定至該資料通道的值。其資料類型依各資料通道而定，例如整數、浮點數、字串等。





LinkIt 7697 for Arduino

環境設定



開發指南



GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

Software Serial

通道類別	資料類型
MCSControllerOnOff	bool
MCSControllerFloat	float
MCSControllerInteger	int
MCSControllerAnalog	int
MCSControllerGPIO	int
MCSControllerHex	long
MCSControllerCategory	String
MCSControllerString	String
MCSControllerPWM	MCSPWMValue
MCSControllerGPS	MCSGPSValue

latitude、longitude、altitude：GPS 資料通道的座標值，其資料類型為浮點數。

Returns

若該資料通道的數值被成功更新至 MCS 伺服器，則回傳布林值 **true**；反之則回傳 **false**。

MCSControllerPWM 控制器通道函式

針對 **PWM 控制器** 資料通道，提供了：

▼ **dutyCycle()**

取得此 PWM 控制器通道的 duty cycle 值。

Syntax

MCSControllerPWM.dutyCycle()



LinkIt 7697 for Arduino

環境設定 >

開發指南 v

GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

Software Serial

Parameters

MCSControllerPWM : MCSControllerPWM 類別的實體。

Returns

該 PWM 控制器通道的 duty cycle 設定值 (在 MCS 控制介面顯示為 value) · 其資料類型為整數。

▼ `period()`

取得此 PWM 控制器通道的週期值。

Syntax

MCSControllerPWM.period()

Parameters

MCSControllerPWM : MCSControllerPWM 類別的實體。

Returns

該 PWM 控制器通道的週期設定值 · 其資料類型為整數。

GPS 通道函式

針對 **GPS 控制器**與 **GPS 顯示器**資料通道，提供了以下函式供查詢經緯度及高度資訊。

▼ `latitude()`

取得此 GPS 控制器通道的緯度值。

Syntax

MCSControllerGPS.latitude()

Parameters

MCSControllerGPS : MCSControllerGPS 類別的實體。

Returns



LinkIt 7697 for Arduino

環境設定 >

開發指南 v

GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

Software Serial

該 GPS 控制器通道的緯度值，其資料類型為浮點數。

▼ `longitude()`

取得此 GPS 控制器通道的經度值。

Syntax

`MCSControllerGPS.longitude()`

Parameters

MCSControllerGPS：MCSControllerGPS 類別的實體。

Returns

該 GPS 控制器通道的經度值，其資料類型為浮點數。

▼ `altitude()`

取得此 GPS 控制器通道的高度值。

Syntax

`MCSControllerGPS.altitude()`

Parameters

MCSControllerGPS：MCSControllerGPS 類別的實體。

Returns

該 GPS 控制器通道的高度值，其資料類型為浮點數。

Gamepad 控制器通道函式

Gamepad 為用來表示按鍵資訊的**控制器通道**，包含兩種按鍵狀態：

- 哪顆鍵 (**BTN_UP**、**BTN_DOWN**、**BTN_LEFT**、**BTN_RIGHT**、**BTN_A**、**BTN_B**) 被操作，以及
- 該按鍵是被按下 (**BTN_PRESSED**) 或放開 (**BTN_RELEASED**)。





LinkIt 7697 for Arduino

環境設定



開發指南



GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

Software Serial

可表示的按鍵 **enumeration** 如下：

```
enum MCSGamePadButton{
    BTN_UP = 1,
    BTN_DOWN,
    BTN_LEFT,
    BTN_RIGHT,
    BTN_A,
    BTN_B,
    BTN_INVALID
};
```

表示按壓狀態的 **enumeration** 為：

```
enum MCSGamePadButtonEvent{
    BTN_PRESSED = 1,
    BTN_RELEASED = 0,
    BTN_NO_EVENT = -1
};
```

因此可透過 *button* 與 *event* 欄位對從 **value()** 函式所取得的資訊進行解讀：

```
if(BTN_UP == gamepadChannel.value().button &&
    BTN_PRESSED == gamepadChannel.value().event)
{
    // Do something when the "UP" key is pressed...
}
```

透過下列函式，亦可取得按鍵資訊與按壓狀態：

› **button()**

取得目前正在作用的按鍵。

Syntax

MCSControllerGamePad.button()

Parameters



LinkIt 7697 for Arduino

環境設定 >

開發指南 v

GPIO

UART

ADC

EINT (外部中斷 / External Interrupt)

I2C

SPI

EEPROM

Timer

Flash (索引式儲存空間)

RTC (Real-Time Clock)

MCSControllerGamePad : MCSControllerGamePad 類別的實體。

Returns

正在作用按鍵的 **MCSGamePadButton** enumeration 值。目前支援的按鍵種類如下：

```
enum MCSGamePadButton{  
    BTN_UP = 1,  
    BTN_DOWN,  
    BTN_LEFT,  
    BTN_RIGHT,  
    BTN_A,  
    BTN_B,  
    BTN_INVALID  
};
```

> event()

取得目前作用按鍵的狀態 (被按下或是放開)。

Syntax

MCSControllerGamePad.button()

Parameters

MCSControllerGamePad : MCSControllerGamePad 類別的實體。

Returns

目前作用按鍵的狀態 (資料類型為 **MCSGamePadButtonEvent**)。提供的資訊種類有：

```
enum MCSGamePadButtonEvent{  
    BTN_PRESSED = 1,  
    BTN_RELEASED = 0,  
    BTN_NO_EVENT = -1  
};
```

[< MCSLiteDevice](#)[使用 BLE 低功耗藍牙 >](#)Powered by [Atlassian Confluence](#) and the [Scroll Content Management Add-ons](#).

LinkIt 7697 for Arduino

[環境設定 >](#)[開發指南 v](#)[GPIO](#)[UART](#)[ADC](#)[EINT \(外部中斷 / External Interrupt\)](#)[I2C](#)[SPI](#)[EEPROM](#)[Timer](#)[Flash \(索引式儲存空間\)](#)[RTC \(Real-Time Clock\)](#)[Software Serial](#)