

Classification	Win-GRAF Chinese FAQ-021						
Author	Sandy Lin	Version	14.0.03	Date	2025, 10	Page	1 / 7

如何下載自定義 C 程式到 LP-2648M 並進行編譯？

[下載 FAQ-021 範例](#)

本節使用在 LP-2648M 上開發的 C 程式《LP2648M_SM_demo》及 SmarTTY 工具，進行程式的上傳與編譯執行，並說明如何透過共用記憶體讀寫資料，與 Win-GRAF 專案中的變數進行互動。

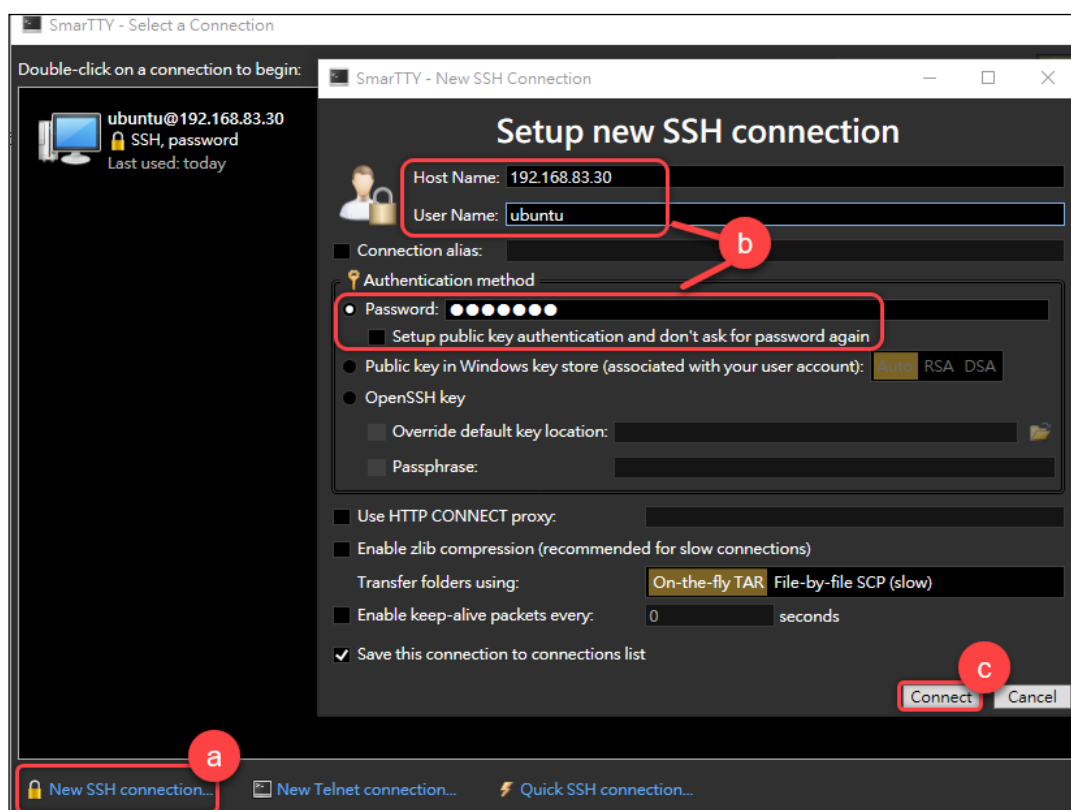
SmarTTY 是一款 SSH 客戶端連線工具。接下來的操作將在 Windows 環境中進行，並透過 SmarTTY 連線至 LP-2648M，進行程式的上傳與編譯。

請先下載並安裝 SmarTTY：<https://sysprogs.com/SmarTTY/download/>

LP2648M_SM_demo 設定與說明

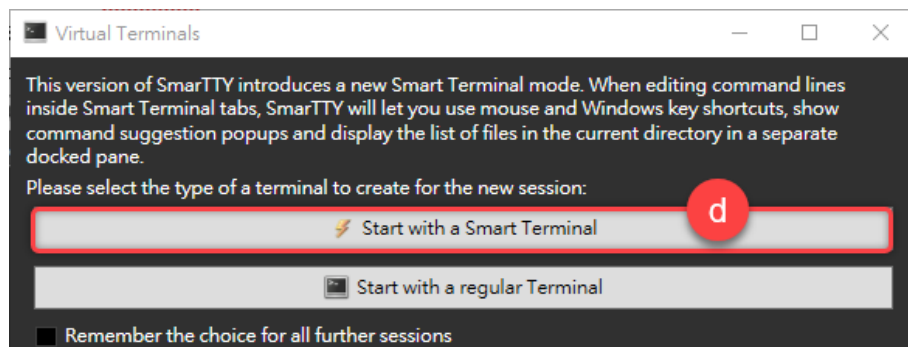
1. 透過 SmarTTY 連結 LP-2648M

- 點選 "New SSH Connection" 項目
- 輸入以下資訊：
Host Name: 192.168.83.30
User Name: ubuntu
Password: tempwd
取消勾選 "Setup public key"
- 點擊 "Connect" 按鈕

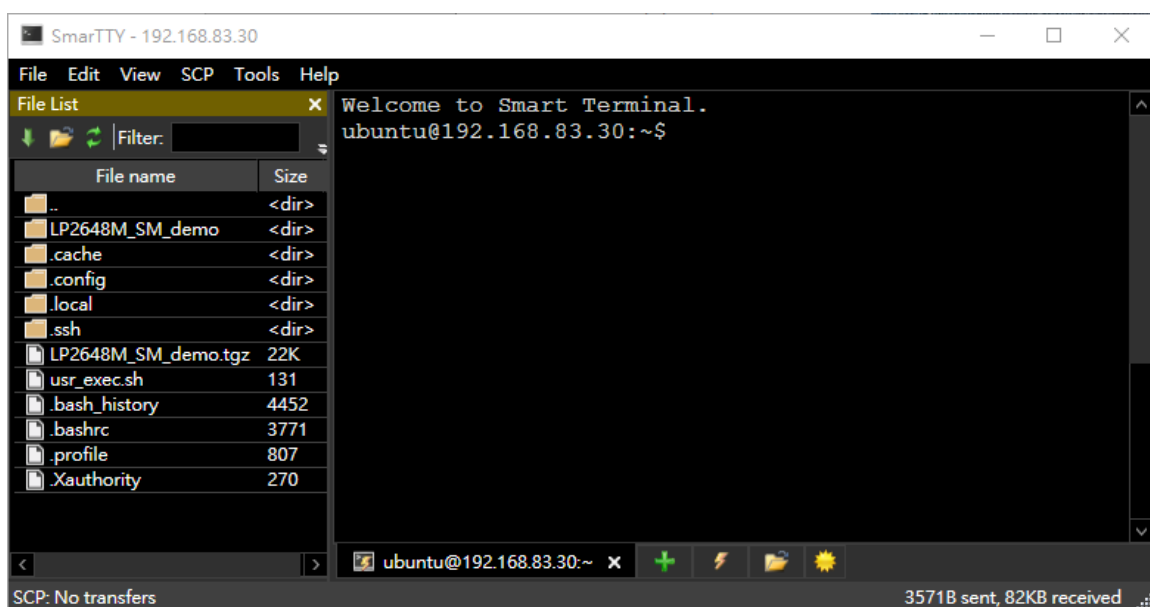


Classification	Win-GRAF Chinese FAQ-021						
Author	Sandy Lin	Version	14.0.03	Date	2025, 10	Page	2 / 7

d. 點擊 “Start with a Smart Terminal” 按鈕

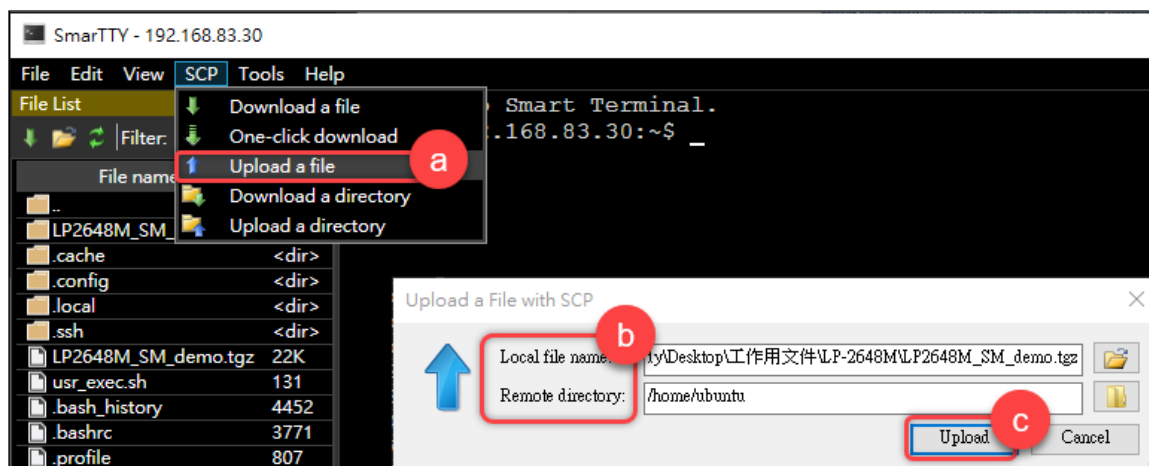


連線成功會出現以下畫面。



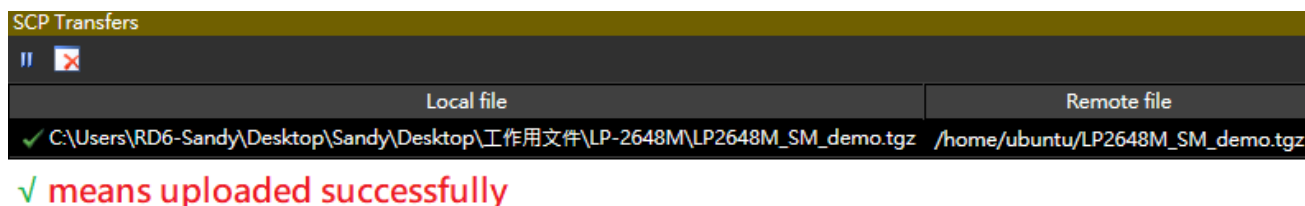
2. 上傳程式

- 點選 SCP -> Upload a file
- 選擇檔案以及檔案目的地
- 點擊 “Upload” 按鈕



Classification	Win-GRAF Chinese FAQ-021						
Author	Sandy Lin	Version	14.0.03	Date	2025, 10	Page	3 / 7

若檔案上傳成功後，會出現下圖。



3. 解壓縮 LP2648M_SM_demo.tgz

使用指令 `tar -zxvf ./LP2648M_SM_demo.tgz` 進行解壓縮。

```
Welcome to Smart Terminal.
ubuntu@192.168.83.30:~$ tar -zxvf ./LP2648M_SM_demo.tgz
./LP2648M_SM_demo/
./LP2648M_SM_demo/LP2648M_SM_demo.c
./LP2648M_SM_demo/Makefile
./LP2648M_SM_demo/lib/
./LP2648M_SM_demo/lib/Quicker.a
./LP2648M_SM_demo/inc/
./LP2648M_SM_demo/inc/Quicker.h
./LP2648M_SM_demo/LP2648M_SM_demo
ubuntu@192.168.83.30:~$
```

4. 進行編譯

a. 使用指令 `cd LP2648M_SM_demo` 移至 LP2648M_SM_demo 資料夾

b. 使用指令 `make` 編譯建置程式

(若輸入指令 `make` 出現 **Nothing to be done for 'all'**，表示這個程式的編譯與建置工作已經完成過，而且檔案沒有任何變動，所以不需要重新編譯，直接進行以下步驟。)

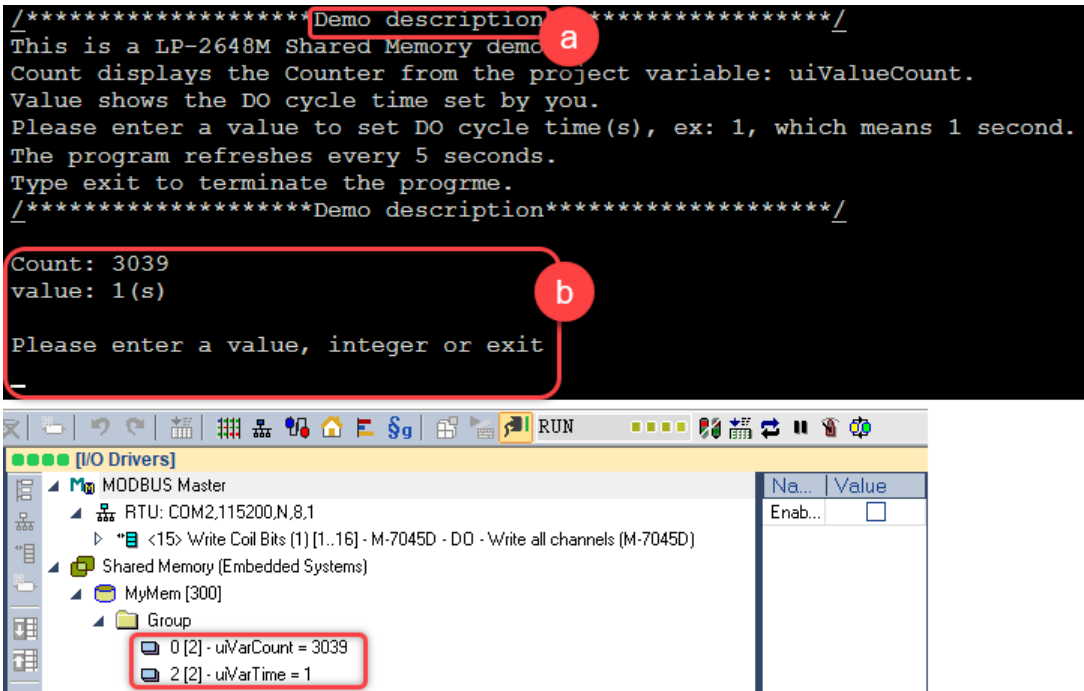
```
ubuntu@192.168.83.30:~$ cd LP2648M_SM_demo
ubuntu@192.168.83.30:~/LP2648M_SM_demo$ make
gcc -I./ -I./inc -o LP2648M_SM_demo LP2648M_SM_demo.c ./lib/Quicker.a -lm ./lib/Quicker.a -lm -pthread
ubuntu@192.168.83.30:~/LP2648M_SM_demo$ ls
LP2648M_SM_demo LP2648M_SM_demo.c Makefile inc lib
ubuntu@192.168.83.30:~/LP2648M_SM_demo$
```

5. 執行程式

※注意: 請先將 Win-GRAF 上的 WG_LP2648M_SM_demo 下載至 LP-2648M，再執行以下操作。

使用指令 `./LP2648M_SM_demo` 執行程式

Classification	Win-GRAF Chinese FAQ-021						
Author	Sandy Lin	Version	14.0.03	Date	2025, 10	Page	4 / 7



a. Demo 使用說明

b. Count: 顯示 Win-GRAF demo 變數 uiVarCount 的數值

(此範例程式會讓畫面上的數值，每 5 秒自動更新一次。)

value: 顯示輸出給 uiVarTime 的數值

Please enter a value, integer or exit

(這裡輸入要輸出給 uiVarTime 的數值，輸出 **exit** 則是結束程式)

6. 程式說明

LP2648M_SM_demo 使用 Quicker.a static library 來達成 Shared Memory 的應用。

```

10 volatile unsigned short iCount = 0;
11 volatile unsigned short Value = 0;
12 char CMD_output[128] = "";
13 volatile int exitFlag = 0;
14
15 char mem_name[128] = "MyMem";
16 int mem_size = 300;
17 int ret count, ret CMD;
64 a UserShare_Init(mem_name, mem_size);

50 b ret_CMD = UserShare_Set_UShort((unsigned short)2, Value);

75 c unsigned short Count = iCount;
76 c ret_count = UserShare_Get_UShort((unsigned short)0, &Count);

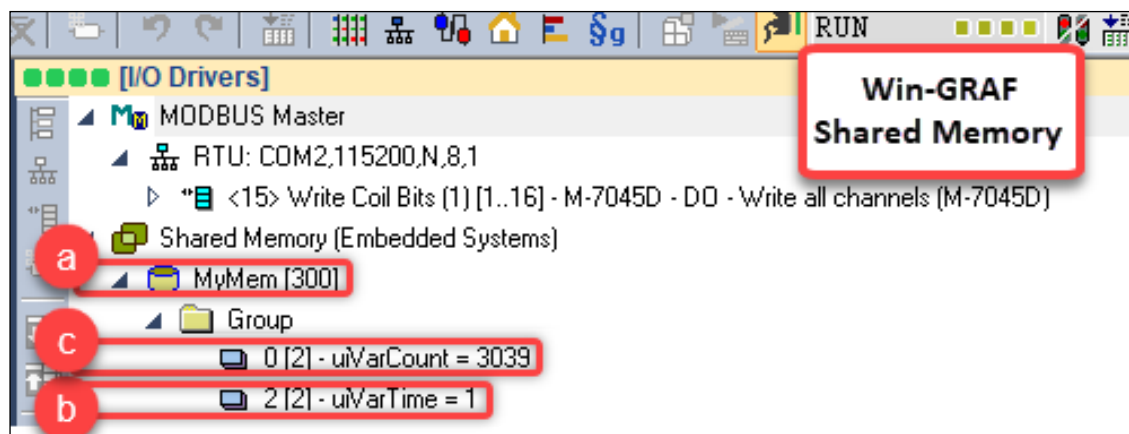
94 d UserShare_Close();

```

**Quicker.a
static library**

Classification	Win-GRAF Chinese FAQ-021						
Author	Sandy Lin	Version	14.0.03	Date	2025, 10	Page	5 / 7

Quicker.a static library 對應於 Win-GRAF 的 Shared Memory 功能，如下圖所示：



以下是 demo 中使用的 API:

a. **void UserShare_Init(char *mmap_name, int mmap_size)**

依 Win-GRAF Shared Memory 名稱: MyMem, 大小:300 bytes 連結共用記憶體。

b. **unsigned char UserShare_Set_UShort(unsigned short iUserAddress, unsigned short iStatus)**

寫入 unsigned short 數值到 Win-GRAF demo 綁定變數 uiVarTime。該變數位於“ MyMem”共用記憶體 offset 2 的位置，所以呼叫的方式為 **UserShare_Set_UShort((unsigned short) 2, value)**。

c. **unsigned char UserShare_Get_UShort(unsigned short iUserAddress, unsigned short *iStatus)**

讀取 unsigned short 數值從 Win-GRAF demo 綁定變數 uiVarCount。該變數位於“ MyMem”共用記憶體 offset 0 的位置，所以呼叫的方式為 **UserShare_Get_UShort ((unsigned short)0,&Count)**。

d. **void UserShare_Close(void)**

結束共用記憶體的使用。

Classification	Win-GRAF Chinese FAQ-021						
Author	Sandy Lin	Version	14.0.03	Date	2025, 10	Page	6 / 7

以下是 Quicker.a API 的說明

“Quicker.a” 為使用者提供多種操作 Win-GRAF 變數資料的功能，包括：

- ✧ 連結/結束 Shared Memory 共用記憶體
- ✧ 讀/寫 Boolean
- ✧ 讀/寫 8 bits、16 bits、32 bits 或 64 bits Integer
- ✧ 讀/寫 32 bits 或 64 bits REAL
- ✧ 讀/寫 至多 256 Characters

● 連結/結束 Shared Memory 共用記憶體

a. void UserShare_Init(char *mmap_name, int mmap_size);

mmap_name: Win-GRAF Shared Memory 設定的共用記憶體名稱。

Mmap_size: Win-GRAF Shared Memory 設定的共用記憶體大小。

b. void UserShare_Close(void);

在程式結束前，呼叫該 function

● Shared Memory API 讀寫對應表

API 讀寫類型	Win-GRAF 對應型態	資料型態 大小	API 名稱
寫(Set)	BOOL	8 bits	unsigned char UserShare_Set_Coil (unsigned short iUserAddress, unsigned char iStatus);
	INT	16 bits	unsigned char UserShare_Set_Short (unsigned short iUserAddress, short iStatus);
	UINT	16 bits	unsigned char UserShare_Set_UShort (unsigned short iUserAddress, unsigned short iStatus);
	LINT	64 bits	unsigned char UserShare_Set_Int64 (unsigned short iUserAddress, long long iStatus);
	ULINT	64 bits	unsigned char UserShare_Set_UInt64 (unsigned short iUserAddress, unsigned long long iStatus);
	DINT	32 bits	unsigned char UserShare_Set_Long (unsigned short iUserAddress, long iStatus);
	UDINT	32 bits	unsigned char UserShare_Set_ULong (unsigned short iUserAddress, unsigned long iStatus);
	SINT	8 bits	unsigned char UserShare_Set_Char (unsigned short iUserAddress, char iStatus);
	USINT	8 bits	unsigned char UserShare_Set_UChar (unsigned short iUserAddress, unsigned char iStatus);
	REAL	32 bits	unsigned char UserShare_Set_Float (unsigned short iUserAddress, float iStatus);

Classification	Win-GRAF Chinese FAQ-021						
Author	Sandy Lin	Version	14.0.03	Date	2025, 10	Page	7 / 7

API 讀寫類型	Win-GRAF 對應型態	資料型態 大小	API 名稱
寫(Set)	LREAL	64 bits	unsigned char UserShare_Set_Double (unsigned short iUserAddress, double iStatus);
	STRING	256 bits 或可自訂	int UserSetReg_Str (unsigned short iUserAddress, char iStatus, unsigned short len);
讀(Get)	BOOL	8 bits	unsigned char UserShare_Get_Coil (unsigned short iUserAddress, unsigned char *iStatus);
	INT	16 bits	unsigned char UserShare_Get_Short (unsigned short iUserAddress, short *iStatus);
	UINT	16 bits	unsigned char UserShare_Get_UShort (unsigned short iUserAddress, unsigned short *iStatus);
	LINT	64 bits	unsigned char UserShare_Get_Int64 (unsigned short iUserAddress, long long *iStatus);
	ULINT	64 bits	unsigned char UserShare_Get_ULInt64 (unsigned short iUserAddress, unsigned long long *iStatus);
	DINT	32 bits	unsigned char UserShare_Get_Long (unsigned short iUserAddress, long *iStatus);
	UDINT	32 bits	unsigned char UserShare_Get_ULong (unsigned short iUserAddress, unsigned long *iStatus);
	SINT	8 bits	unsigned char UserShare_Get_Char (unsigned short iUserAddress, char *iStatus);
	USINT	8 bits	unsigned char UserShare_Get_UChar (unsigned short iUserAddress, unsigned char *iStatus);
	REAL	32 bits	unsigned char UserShare_Get_Float (unsigned short iUserAddress, float *iStatus);
	LREAL	64 bits	unsigned char UserShare_Get_Double (unsigned short iUserAddress, double *iStatus);
	STRING	256 bits 或可自訂	int UserGetReg_Str (unsigned short iUserAddress, char *iStatus, unsigned short len);

❖ 在 API 中，iUserAddress 表示 Win-GRAF 的 Offset 位址。