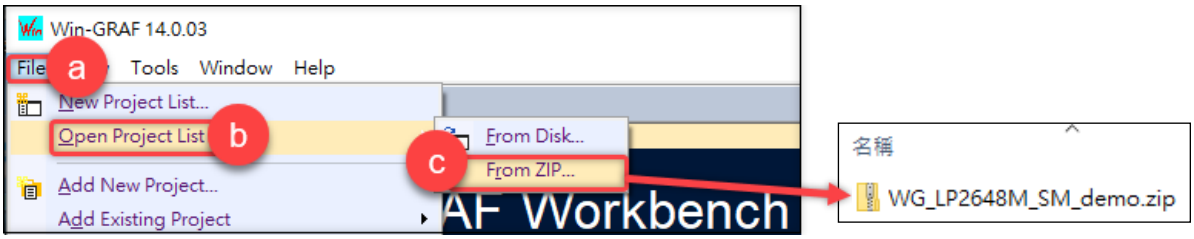
How to Make Win-GRAF Variables Accessible to C Programs?

Download FAQ-020 Demo

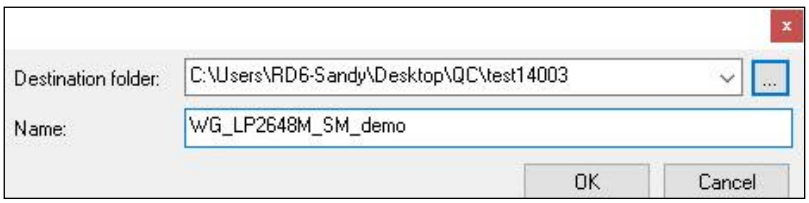
This section uses the **Win-GRAF Workbench project "WG_LP2648M_SM_demo"** as an example to explain how to use the **Shared Memory** function under *Fieldbus* for variable binding.
The example also includes a basic ST (Structured Text) program corresponding to this function, which will be explained in detail below.

Importing the WG_LP2648M_SM_demo Project:

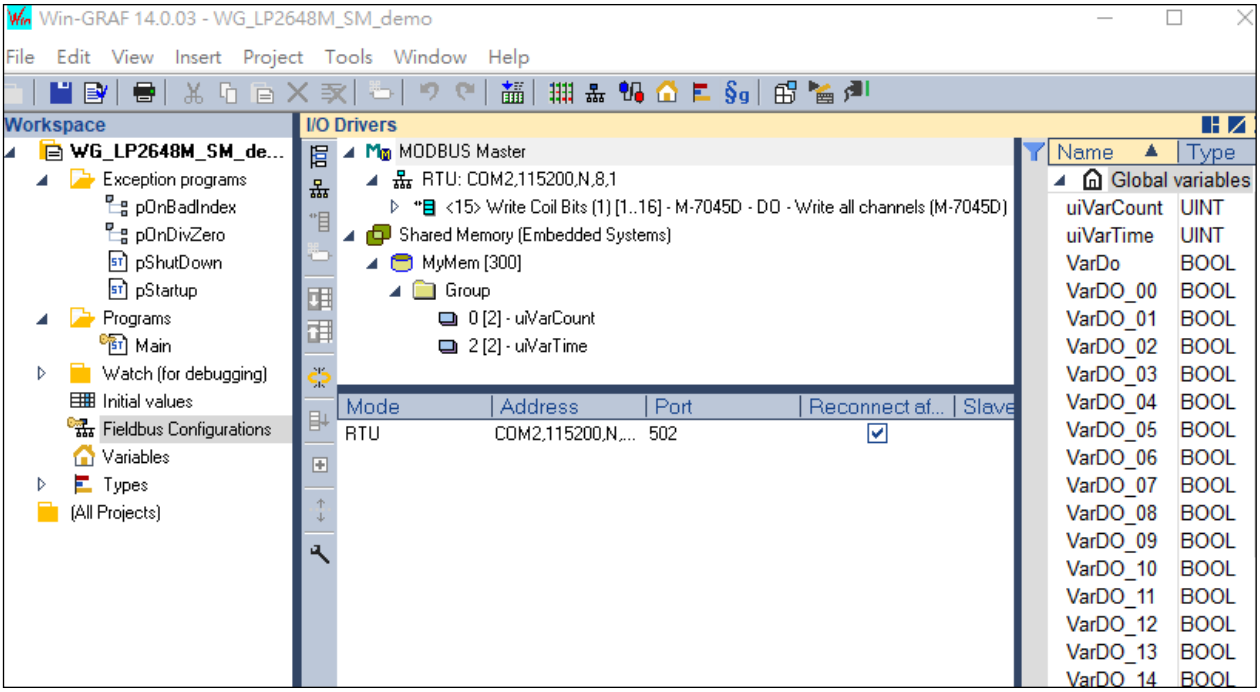
1. Open **Win-GRAF Workbench**, and click: (a) File →(b) Open Project List →(c) From Zip...
2. Select the compressed file: WG_LP2648M_SM_demo.zip



3. Set the project destination folder and file name, then click **OK** to create and import the project.



After these steps, the example project will open as shown below:



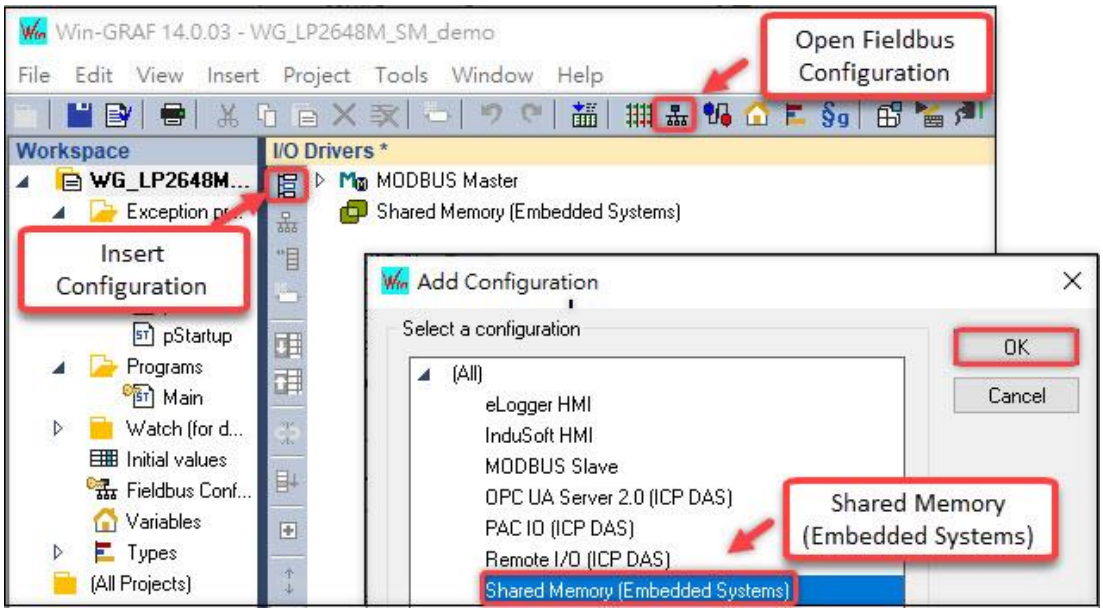
LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

WG_LP2648M_SM_demo Overview:

1. Fieldbus – Shared Memory

To enable the Shared Memory function:

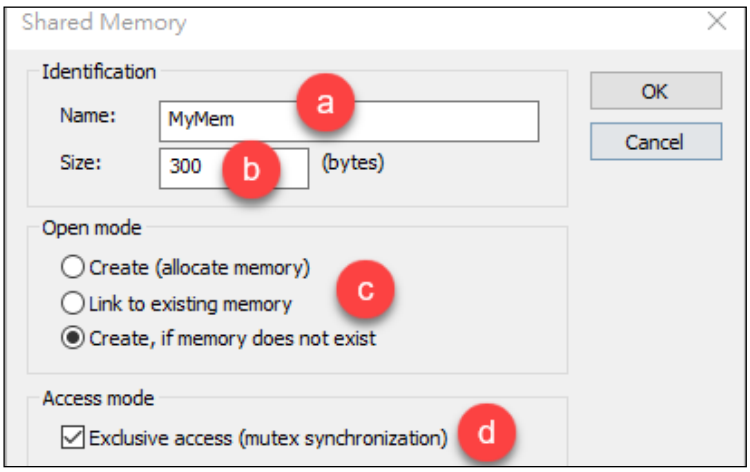
Open Fieldbus Configuration → Insert Configuration → (All) → Shared Memory (Embedded Systems) → OK



✧ Insert Master/Port

Configure Shared Memory:

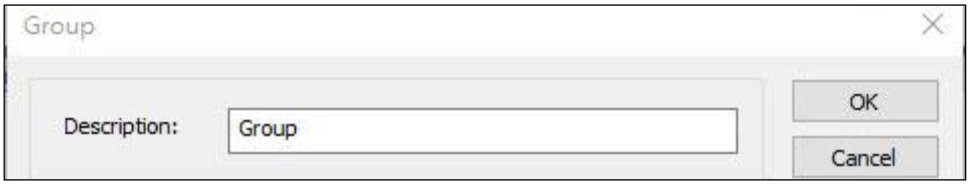
a. **Name**: Shared memory name, e.g., *MyMem*
b. **Size**: Define based on the total size of all variable data types (BOOL, DINT, etc.); it is recommended to exceed the total slightly. Example: *300*
c. **Open mode**: Specify how the shared memory is opened; the default value is recommended.
d. **Access mode**: *Exclusive Access (Mutex Synchronization)* is enabled by default, ensuring that only one program reads or writes data at a time to avoid data conflicts from concurrent access



Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

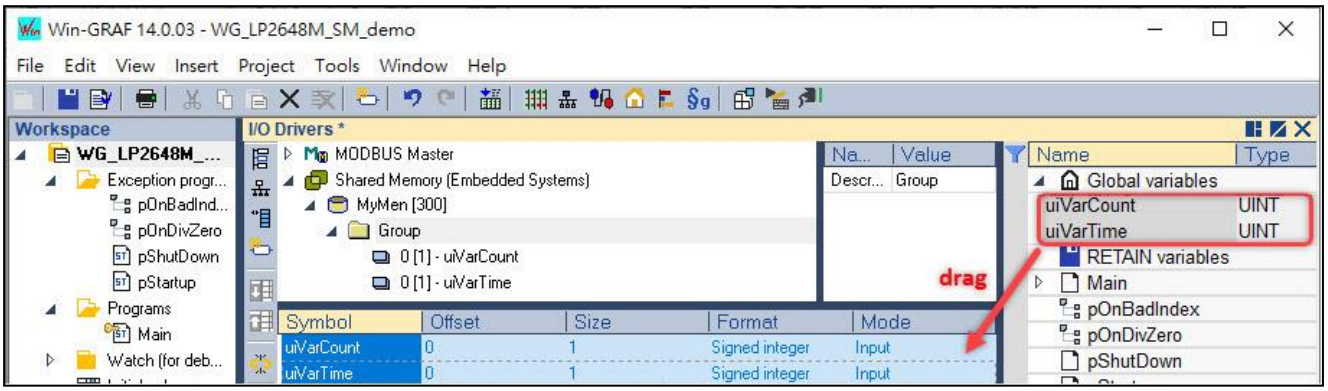
✧ Insert Slave Data Blocks

This example uses the default “Group” category; no modification is needed.

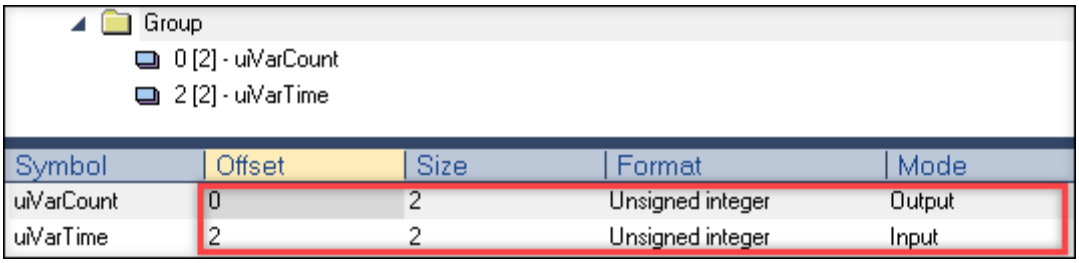


✧ Insert Variable

Drag variables **uiVarCount** and **uiVarTime** into the *Lab* area to complete the variable binding.
Both variables are of type **UINT**.



Double-click the variable (e.g., *uiVarCount* or *uiVarTime*) in the **Offset**, **Size**, **Format**, or **Mode** field to modify the parameter settings.
The updated values will appear immediately, as shown in the figure below.



Shared Memory Variable Address Configuration

> Variable Address:

- The address refers to the starting position (Offset) of a variable within the Shared Memory.

> Addressing Rules:

- The first variable starts at offset 0.
- The starting address of the next variable equals the previous variable’s starting address plus its memory size (in bytes).

Example:

Variable Name	Start Address (Offset)	Data Type	Bytes
uiVarCount	0	UINT	2
uiVarTime	2 (= 0 + 2)	UINT	2

❖ UINT is a 16-bit (2-byte) data type.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

2. Modbus Master RTU

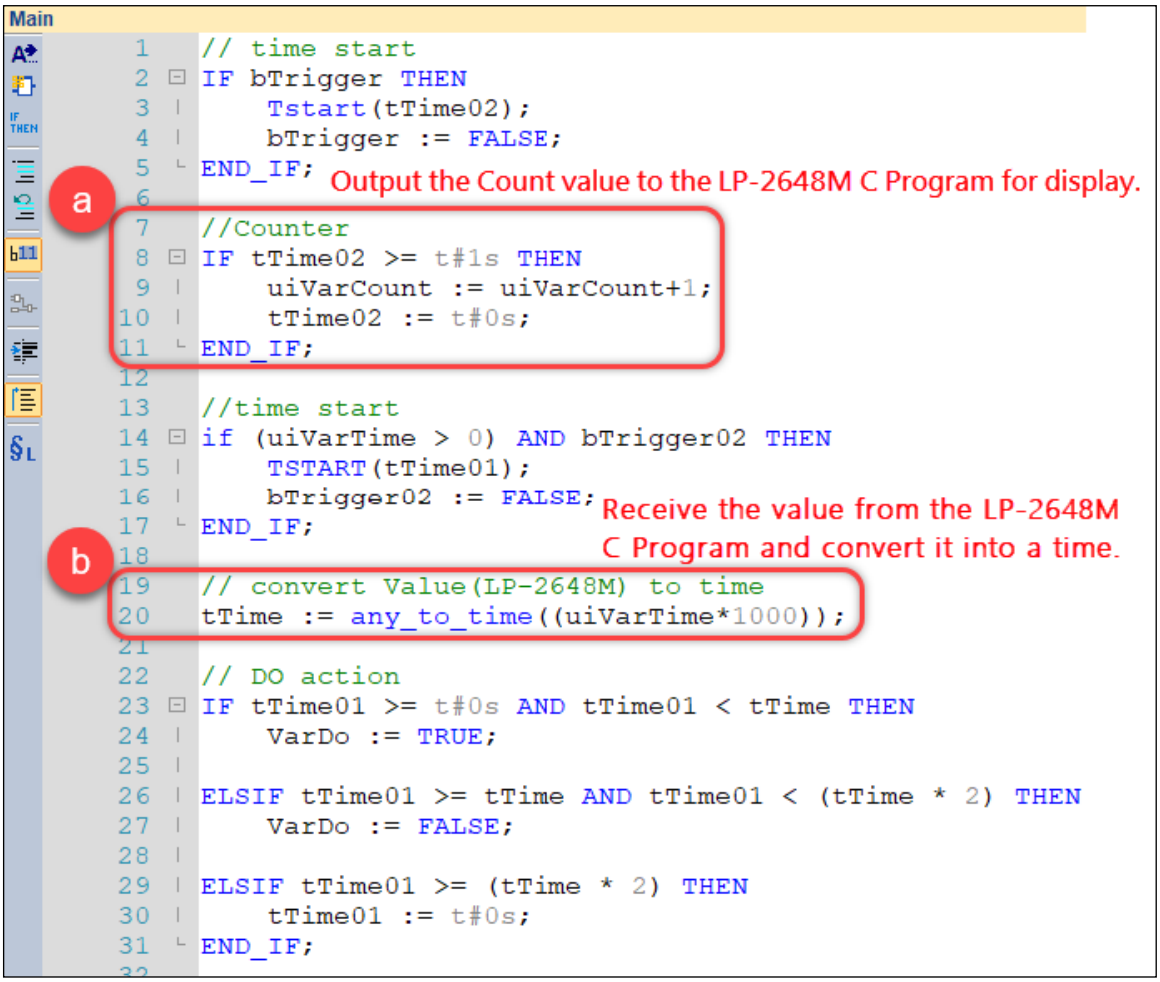
In this example, the ST program controls the DO output time of the **M-7045D** module through Modbus RTU.

Before use, make sure the **M-7045D module** has been properly configured. (Refer to Chapter 5 of the Win-GRAF User Manual)

3. ST Program

The ST program in the demo mainly performs the following functions:

a. Transmit the accumulated **Counter value (uiVarCount)** to the LP-2648M C program for display.
b. Receives a value (**uiVarTime**) from the C program and converts it into the DO output time for the **M-7045D** module.




```
1  // time start
2  IF bTrigger THEN
3      Tstart(tTime02);
4      bTrigger := FALSE;
5  END_IF;
6
7  //Counter
8  IF tTime02 >= t#1s THEN
9      uiVarCount := uiVarCount+1;
10     tTime02 := t#0s;
11 END_IF;
12
13 //time start
14 if (uiVarTime > 0) AND bTrigger02 THEN
15     TSTART(tTime01);
16     bTrigger02 := FALSE;
17 END_IF;
18
19 // convert Value(LP-2648M) to time
20 tTime := any_to_time((uiVarTime*1000));
21
22 // DO action
23 IF tTime01 >= t#0s AND tTime01 < tTime THEN
24     VarDo := TRUE;
25
26 ELSIF tTime01 >= tTime AND tTime01 < (tTime * 2) THEN
27     VarDo := FALSE;
28
29 ELSIF tTime01 >= (tTime * 2) THEN
30     tTime01 := t#0s;
31 END_IF;
```

WG_LP2648M_SM_demo project is applicable to all Win-GRAF PACs.

After downloading this project to different PAC models, you can use various external programs to access the project's variable data. Please follow the program language and the corresponding Win-GRAF FAQ for each PAC model, as shown in the table below.

PAC Model	External Programming Language	Win-GRAF FAQ Reference
LP-2648M	C	FAQ-021

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000