
RPAC User Manual

Using a C Program to Access Win-GRAF Variables

Chapter 1 Using a C Program to Access Win-GRAF Variables

The Chapter describes how to create variables in the Win-GRAF project for the C program to read/write data. The C demo program is developed by using the GCC tool as well as Win-GRAF project are provided.

C demo program - Quicker_Demo.tgz

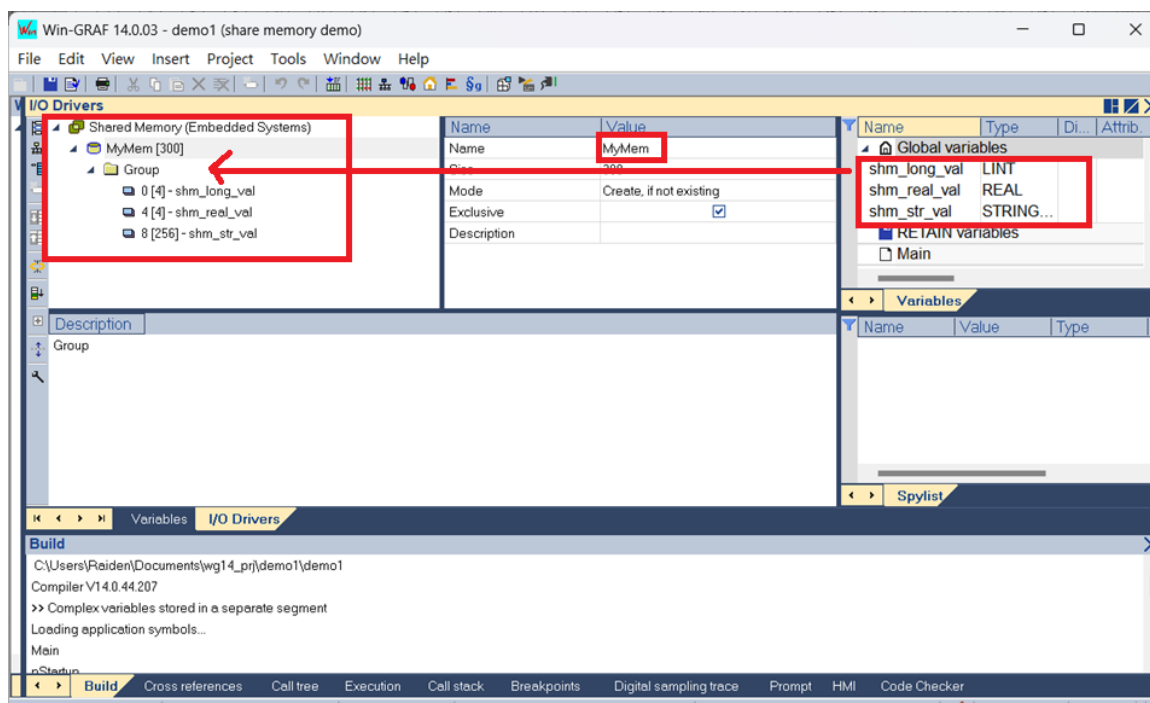
Used to read/write Win-GRAF Char, Short, Long, ULONG, Int64, Float, Double, or String variable.

Win-GRAF demo programs – demo1.zip

Download [the demo program](#) and run the **File** menu commands **Add Existing Project** and then **From Zip...** in the Workbench to open the project.

1.1 Publishing the Win-GRAF Variable for C Program

In the Win-GRAF, variables and ID must be set in the "I/O Drivers" window so that the C program can access data with the same address.



1.2 Downloading and Compiling the C Program

Please follow the "SSH Connection Instructions" to connect and transfer the Quicker_Demo.tgz file to RPAC.

Follow the steps below to compile the C program:

1. Extract the **Quicker_Demo.tgz** archive file.

Using the '`tar -zxvf ./Quicker_Demo.tgz`' command to extract files to the Quicker_Demo folder.

```
ubuntu@LP-2648M:~$ tar -zxvf ./Quicker_Demo.tgz
./Quicker_Demo/
./Quicker_Demo/v1.10/
./Quicker_Demo/v1.10/Makefile
./Quicker_Demo/v1.10/lib/
./Quicker_Demo/v1.10/lib/Quicker.a
./Quicker_Demo/v1.10/inc/
./Quicker_Demo/v1.10/inc/Quicker.h
./Quicker_Demo/v1.10/Quicker_demo.c
ubuntu@LP-2648M:~$ _
```

2. Run the **make** command to build the demo program.

- 1) Type the '**cd Quicker_Demo/vXX.XX**' command to enter the directory.
- 2) Type the '**make**' command to build the program in the current folder.

```
ubuntu@LP-2648M:~$ cd Quicker_Demo/v1.10/
ubuntu@LP-2648M:~/Quicker_Demo/v1.10$ make
gcc -I./ -I./inc -o Quicker_demo Quicker_demo.c ./lib/Quicker.a
-lm ./lib/Quicker.a -lm -lpthread -lrt
ubuntu@LP-2648M:~/Quicker_Demo/v1.10$ ls
Makefile Quicker_demo Quicker_demo.c inc lib
ubuntu@LP-2648M:~/Quicker_Demo/v1.10$ _
```

3. Run the demo program to access Win-GRAF variables.

```
ubuntu@LP-2648M:~/Quicker_Demo/v1.10$ ./Quicker_demo
=====
Usage : ./Quicker_demo MemoryNAME MemorySize Set/Get Char/Short/Long/ULong/I
nt64/Float/Double/Str Offset(0~65535) (Value)
Ex : ./Quicker_demo MyMem 1000 Set Char 0 0
Ex : ./Quicker_demo MyMem 1000 Get Char 0
Ex : ./Quicker_demo MyMem 1000 Set Str 0 hello
Ex : ./Quicker_demo MyMem 1000 Get Str 0
=====
ubuntu@LP-2648M:~/Quicker_Demo/v1.10$ ./Quicker_demo MyMem 300 Set Long 0 1
Set_Long: ret=0, addr=0, value=1
ubuntu@LP-2648M:~/Quicker_Demo/v1.10$ ./Quicker_demo MyMem 300 Get Long 0
Get_Long: ret=0, addr=0, value=1
ubuntu@LP-2648M:~/Quicker_Demo/v1.10$ _
```

1.3 Descriptions of Functions in the "Quicker.a"

The Section describes functions in the "Quicker.a" static library.

"Quicker.a" provides many functions for users to read/write Win-GRAF variables data as follows:

1. Read/Write Boolean
2. Read/Write the 8-bit, 16-bit, 32-bit, or 64-bit Integer
3. Read/Write the 32-bit or 64-bit REAL

1.3.1 R/W a Boolean Variable

■ Set_BOOL

Description:

Used to set the value of a Win-GRAF Boolean variable with the specific offset no.

Syntax:

```
unsigned char UserShare_Set_Coil(unsigned short iUserAddress, unsigned char iStatus)
```

Parameter:

iUserAddress: The offset no of the Win-GRAF variable (0 to 65535)

iStatus: Set a Boolean value. (1: True; 0: False)

Example:

'Set a Boolean variable value to True with offset no.0

UserShare_Set_Coil(0, 1)

■ Get_BOOL

Description:

Used to get the value of a Win-GRAF Boolean variable with the specific offset no.

Syntax:

```
unsigned char UserShare_Get_Coil(unsigned short iUserAddress, unsigned char *iStatus)
```

Parameter:

iUserAddress: The offset no of the Win-GRAF variable (0 to 65535)

iStatus: Get a Boolean value. (1: True; 0: False)

Example:

'Get a Boolean variable value with offset no "0"

unsigned char value;

UserShare_Get_Coil(0, &value);

1.3.2 R/W an Integer Variable

■ **Set_Char** ■ **Set_Short** ■ **Set_Long** ■ **Set_Int64**
■ **Set_UChar** ■ **Set_UShort** ■ **Set_ULong** ■ **Set_UInt64**

Description:

Used to set the value of a Win-GRAF 8-bit, 16-bit, 32-bit, or 64-bit Integer variable with the specific offset no.

Syntax:

```
unsigned char UserShare_Set_Char (unsigned short iUserAddress, char iStatus)
unsigned char UserShare_Set_Short (unsigned short iUserAddress, short iStatus)
unsigned char UserShare_Set_Long (unsigned short iUserAddress, long iStatus)
unsigned char UserShare_Set_Int64 (unsigned short iUserAddress, long long iStatus)
unsigned char UserShare_Set_ULong (unsigned short iUserAddress, unsigned long iStatus)
unsigned char UserShare_Set_UChar (unsigned short iUserAddress, unsigned char iStatus)
unsigned char UserShare_Set_UShort (unsigned short iUserAddress, unsigned short iStatus)
unsigned char UserShare_Set_UInt64 (unsigned short iUserAddress, unsigned long long iStatus)
```

Parameter:

iUserAddress: The offset no of the Win-GRAF variable (0 to 65535)

iStatus: Set an 8-bit, 16-bit, 32-bit, or 64-bit Integer value

Example:

‘Set the value of a 32-bit integer variable with offset no "1" to "1234567"’

UserShare_Set_Long(1, 1234567)

‘Set the value of a 32-bit integer variable with offset no "2" to "-1234"’

UserShare_Set_Long(2, -1234)

‘Set the value of a 64-bit integer variable with offset no "3" to "123456789012345"’

UserShare_Set_Int64(3, 123456789012345)

‘Set the value of an 8-bit integer variable with offset no "4" to "125"’

UserShare_Set_Char(4, 125)

 **Get_Char**
  **Get_Short**
  **Get_Long**
  **Get_Int64**
 **Get_UChar**
  **Get_UShort**
  **Get_ULong**
  **Get_UInt64**

Description:

Used to get the value of a Win-GRAF 8-bit, 16-bit, 32-bit, or 64-bit Integer variable with the specific offset no.

Syntax:

```

unsigned char UserShare_Get_Char (unsigned short iUserAddress, char *iStatus)
unsigned char UserShare_Get_Short (unsigned short iUserAddress, short *iStatus)
unsigned char UserShare_Get_Long (unsigned short iUserAddress, long *iStatus)
unsigned char UserShare_Get_Int64 (unsigned short iUserAddress, long long *iStatus)
unsigned char UserShare_Get_UChar (unsigned short iUserAddress, unsigned char *iStatus)
unsigned char UserShare_Get_UShort (unsigned short iUserAddress, unsigned short *iStatus)
unsigned char UserShare_Get_ULong (unsigned short iUserAddress, unsigned long *iStatus)
unsigned char UserShare_Get_UInt64 (unsigned short iUserAddress, unsigned long long *iStatus)
    
```

Parameter:

iUserAddress: The offset no of the Win-GRAF variable (0 to 65535)
 iStatus: Get an 8-bit, 16-bit, 32-bit, or 64-bit Integer value

Example:

```

long long Dlong_val;
long long_val;
short short_val;
char sbyte_val;
    
```

'Get the value of a 64-bit integer variable with offset no "7"

UserShare_Get_Int64(7, &Dlong_val);

'Get the value of a 32-bit integer variable with offset no "8"

UserShare_Get_Long(8, &long_val);

'Get the value of a 16-bit integer variable with offset no "9"

UserShare_Get_Short(9, &short_val);

'Get the value of an 8-bit integer variable with offset no "10"

UserShare_Get_Char(10, &sbyte_val);

1.3.3 R/W a REAL Variable

■ Set_Float ■ Set_Double

Description:

Used to set the value of a Win-GRAF 32-bit or 64-bit REAL variable with the specific offset no.

Syntax:

```
unsigned char UserShare_Set_Float(unsigned short iUserAddress, float iStatus)
unsigned char UserShare_Set_Double(unsigned short iUserAddress, double iStatus)
```

Parameter:

iUserAddress: The offset no of the Win-GRAF variable (0 to 65535)

iStatus: Set a floating-point number

Example:

‘Set the value of a 64-bit REAL variable with offset no "7" to “11234.234567”

UserShare_Set_Double(7, 11234.234567)

‘Set the value of a 32-bit REAL variable with offset no "8" to “123.12”

UserShare_Set_Float(8, 123.12);

■ Get_Float ■ Get_Double

Description:

Used to get the value of a Win-GRAF 32-bit or 64-bit REAL variable with the specific offset no.

Syntax:

```
unsigned char UserShare_Get_Float(unsigned short iUserAddress, float *iStatus)
unsigned char UserShare_Get_Double(unsigned short iUserAddress, double *iStatus)
```

Parameter:

iUserAddress: The offset no of the Win-GRAF variable (0 to 65535)

iStatus: Get a floating-point number

Example:

float float_val;

double double_val;

‘Get the value of a 64-bit REAL variable with offset no "7"

UserShare_Get_Double(7, &double_val);

‘Get the value of a 32-bit REAL variable with offset no "8"

UserShare_Get_Float(8, &float_val);

1.3.4 R/W a String Variable

■ Set_STRING

Description:

Used to set Win-GRAF String data with the specific offset no.

Syntax:

```
int UserSetReg_Str(unsigned short iUserAddress, char iStatus, unsigned short len)
```

Parameter:

iUserAddress: The offset no of the Win-GRAF variable (0 to 65535)

iStatus: Get/Set Win-GRAF String pointers

len: Set the length of a Win-GRAF String

Example:

```
char str_value[256];  
//Set String data with offset no "7"  
sprintf(str_value,"hello world \n");  
UserSetReg_Str(7, str_value, strlen(str_value));
```

■ Get_STRING

Description:

Used to get Win-GRAF String data with the specific offset no.

Syntax:

```
int UserGetReg_Str(unsigned short iUserAddress, char *iStatus, unsigned short len);
```

Parameter:

iUserAddress: The offset no of the Win-GRAF variable (0 to 65535)

iStatus: Get/Set Win-GRAF String pointers

len: Get the length of a Win-GRAF String

Example:

```
char str_value[256];  
// Get String data with offset no "7"  
UserGetReg_Str(7, str_value, 256);
```