

Macro Editor™ Control Guide



January 2007 Edition
Part Number 01072007-01

Worldwide Technical Support and Product Information

www.manualends.com

ManualEnds.com
ManualEnds.com



For further support information, see the [Technical Support Resources](#) appendix.
To comment on the documentation, send e-mail to

support@manualends.com

© Copyright 2001-2007 ManualEnds Technology. All rights reserved

1 Try Out Macro Editor™

Using Controls

This User Guide contains the following exercises that lead you through the following Macro Editor activities:

- *Trigger*
- *Communication*
- *Remote Execution (Networking)*
- *Hook*
- *Script*






Tip The Macro Editor online help contains information on all the Macro Editor objects, features, and services, as well as a developer tour that includes a brief tutorial.

You use Control to get action input and to connect Target output. Each control has its own set of properties, methods, and events. This chapter introduces you to the standard controls in Macro Editor.

Introduction to Controls

The Control Module box contains the tools you use to set controls on your process.

General tool

Icon	Control name	Command name	Description
	Wait 3 Seconds	Wait3	Pause 3 seconds
	Loop	Control-Loop	Start process again
	Pass	-	Do nothing just pass.
	Wait ? Second	Wait?	Enables you to input a number (time) to pause the process.
	Loop ? Times	Loop?	Enables you to input a loop time to run the process.

Note *Wait ? Second*



Once you click on *Wait ? Second* button, the popup input window will show at Figure 1.

Figure 1 The popup *Wait ? Second* input window



Input a value for wait period and then click on **Yes**.

Note *Loop ? Times*



Once you click on *Loop ? Times* button, the popup input window will show at Figure 2.

Figure 2 The popup *Loop ? Times* input window



Input a value for loop count and then click on **Yes**.

Tip *Loop ? Times*

Remember that once you create a Loop command, your process will go back to beginning. Any command (after the Loop command) will not be proceed.

Trigger tool

Icon	Control name	Command name	Description
	Frequency	Frequency	Enables your task to take place by seconds, minute, hours, weeks.
	Time	Time	Enables your task to take place in particular date.
	Windows Running Time	WindowsRunningTime	Enables your task to take place in Windows OS running time.
	System Idle	SystemIdle	Trigger when the computer is unoccupied.
	Hard Disk Space	HardDiskSpace	Trigger when the Disk space is limited.
	Memory Status	MemoryUsing	Trigger when the system Memory is limited.
	CPU Status	CPUStatus	Trigger when the CPU go to un-limited loop..
	Modem Status	ModemStatus	Trigger when the Modem data transformation is stopped.
	Running Program	RunningProgram	Trigger when running program existed..
	Not Running Program	NotRunningProgram	Trigger when running program Not existed..
	Interface I/O Input	InterfaceIO	Trigger when Interface I/O Input value.
	Com Port I/O Input	ComIO	Trigger when Com Port I/O Input value.
	IF Any Error	IFAnyError	Trigger when there is any error at process..
	IF Error At ?	IFErrorAt?	Trigger when there is any error at particular process.
	Message Title	MessageTitle-?	Trigger when there is a message box appeared (full title text)
	Set Part of Message Title	SetAnyMessageText-?	Trigger when there is any selected message title text appeared
	Watch Folder	FolderWatch	Trigger when there is any change in Folder / File.
	Registry Monitor	RegistryMonitor	Trigger when Registry Table changed.

Remote Execution Tool



NetWork
Monitor

NetWorkMonitor

Displays and allows a user to monitor Network status.



NetWork
RemoteWakeUp

NetWorkRemoteWakeUp

Allows waking up a remote Host computer via Network.

Hook tool



Control name
Keyboard Hook

Command name
KeyboardHook

Description
Hook Keyboard. Trigger when key text match.



Program Hook

ProgramHook

Hook two different program windows and Trigger when key text match.

Script tool



Control name
Script Program

Command name
ProgramScript

Description
Execute Script data file with Input / Output capability.

Trigger

If you have access to an event or time that causes a task to run on your work, try out Macro Editor's easy-to-use Trigger. There are several types of Triggers in Macro Editor:

Message

- Message Title (Any Window Box, Full title text)
- Part of Message Title (Any Window Box , Any part of message title text)

Date / Time

- Frequency
- Time
- Windows Running Time

System

- System Idle
- Hard Disk Space
- Memory Status
- CPU Status
- Modem Status
- Registry Monitor

Program / Folder

- Running Program
- Not Running Program
- Watch Folder

Hardware

- Interface I/O Input
- Com Port I/O Input

Logic

- IF Any Error
- IF Error At?

Create a Frequency Control



A Macro Editor Frequency control is a schedule action to set your process using Time value. A Frequency trigger allows you to configure a task to run at a certain time.. Use this function to create a monitor routine work, such as Ping any Server or Client to check the system is responding.

Create a Time Control



A Macro Editor Time control is a schedule trigger allows you to configure a task to run at a certain date and time.

Create a Windows Running Time Control



A Windows Running Time control is a Operation time monitor trigger allows you to configure a task to run at a certain Operation time.

Limit: Set Time Maximum: < 24 hours

Create a System Idle Control



A Macro Editor System Idle control is a Mouse & Keyboard monitor action. When doing nothing on the system, the computer is referred to as being in an Idle state. By using the Idle trigger to check the system and then trigger any action.

Create a Hard Disk Space Control



A Macro Editor Hard Disk Space control is a Hard Disk monitor action. By using the Hard Disk Space trigger to check the Hard Disk space and then trigger any action..

Create a Memory Status Control



A Macro Editor Memory Status control is a system Memory monitor action. By using the Memory Status trigger to check the system memory using status and then trigger any action.

Create a CPU Status Control



A Macro Editor CPU Status control is a system CPU monitor action. By using the CPU Status trigger to check the system CPU using status (such as, CPU enter a un-limit loop) and then trigger any action.

Create a Modem Status Control



A Macro Editor Modem Status control is a Modem Transfer Data monitor action. When Modem transfer data is unoccupied, the limit of waiting time will start to count down. By using the Modem Status trigger to check the data transfer status under Modem and then trigger any action.

Create a Running Program Control



A Macro Editor Running Program control is a running program monitor action. If the program starts to run, the trigger will be detected and then launch tasks.

Create a Not Running Program Control



A Macro Editor **Not** Running Program control is a **Not** running program monitor action. If the program is not existed, the trigger will detect and then launch tasks.

Create a Interface I/O Input Control



A Macro Editor Interface I/O Input control is a Hardware Interface Signal Detect monitor action. You could build your Interface card then pass the signal to trigger and launch tasks for feedback operation.

Create a Com Port I/O Input Control



A Macro Editor Com Port I/O Input control is a Com Port Signal Detect monitor action. You could connect 2 systems and monitor the signal to trigger and launch tasks.

Create a IF Any Error Control



A Macro Editor IF Any Error / IF Error At? control are logic actions to check your automation processes shown in Figure 3. This could be very useful if any error occurred under process.

Figure 3 IF Any Error and IF Error At? Application window



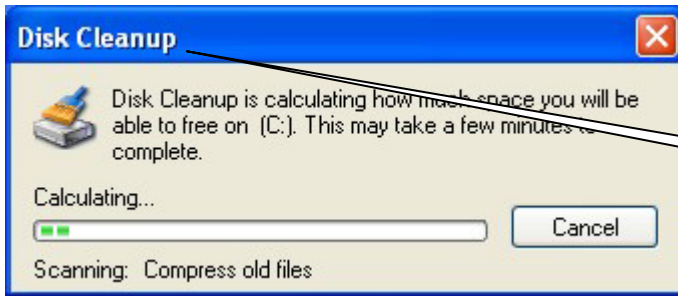
Create a Message Title Control



A Macro Editor Message Title control is a Windows Message Title monitor action shown in Figure 4. You could trigger any Windows Message Title (Full Title Text) and then launch tasks.

Message Title

Figure 4 Message Title window



Disk Cleanup

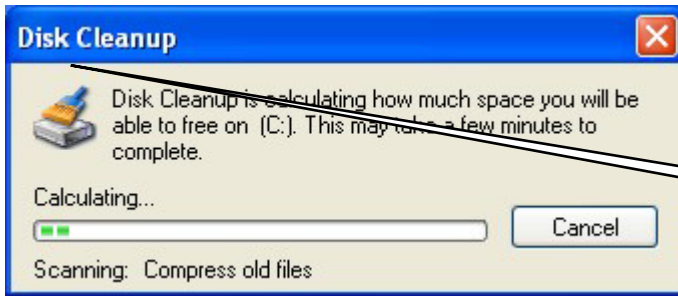
Create a Set Part of Message Title Control



A Macro Editor Set Part of Message Title control is a Windows Title monitor action shown in Figure 5. You could trigger any selected Windows title text and then launch tasks.

Set Part of Message Title

Figure 5 The Set Part of Message Title window



Disk

Create a Watch Folder Control



A Macro Editor Watch Folder control is a Folder monitor action and allows further enhancing your productivity by watching for certain Folder and then launching tasks.

Create a Registry Monitor Control



A Macro Editor Register Monitor control is a Windows Registry monitor action. Enable to watch for System Setting and then trigger to launch tasks.

Create a Net Work Monitor Control



A Macro Editor Net Work Monitor control is a monitor action to check the data transformation status of Network between two machines.

Create a Net Work Remote WakeUp Control



A Macro Editor Net Work Remote WakeUp control is a call action to wake up the remote host computer of Network using Advanced Configuration and Power Interface (ACPI) technology..

Before using this function, please check the following:

Computer BIOS

Machine's BIOS must comply with the Advanced Power Management (APM) specification v1.2 or the Advanced Configuration and Power Interface (ACPI) specification v1.0 or later. Machines that comply with APM v1.2 must also support resume timers. The BIOS must set Wake On LAN [Enable]. Wake on LAN is an ACPI function allowing a powered down (sleeping) computer to be powered ON from a remote station.

Ethernet Adapter

The machine's Ethernet Adapter must have "Wake On LAN (WOL)" function on board with WOL Cable connected to Motherboard.

Computer Networking

The machine must be connected to a network and must use the TCP/IP protocol.

The purpose of the second machine is to wake up the test machine. It functions as a "Wake Server". This server only needs to satisfy the networking requirement.

Find the IP address and Ethernet Adapter Physical Address (Local Ethernet Adapter Physical Address).

Hook

If you have access to two data of programs on your system simultaneity, you can communicate between your program data and try out Macro Editor's easy-to-use Hook technology.
This exercises show how a program data can be distributed across any other programs in your system..

Create a Keyboard Hook Control



A Macro Editor Keyboard Hook control is a keyboard hook action to monitor key press of keyboard.

Create a Program Hook Control



A Macro Editor Program Hook control is a program hook action to do a data communication between two different programs.

Script

BASIC Scripting is available in Macro Editor. This powerful scripting engine lets you integrate BASIC language support into your automated procedures. With Script Program, you can now have Macro Editor make decisions while a task is running. Scripting gives you full control of Window Message Box, Mathematical Calculation, Hardware Interface I/O,...etc.

Create Script Program Control



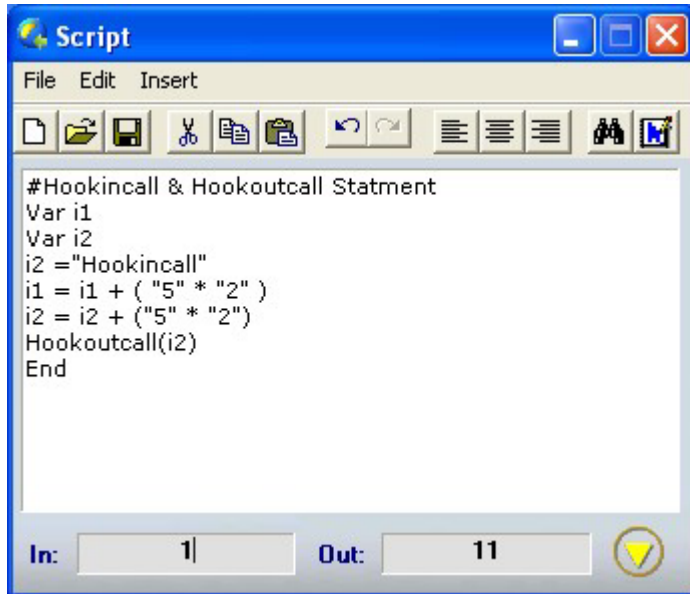
A Macro Editor Script Program control is a Script Macro file action. It is very useful to execute your calculation process through Script Program. Such as, input a value (from window number A), do calculation, then output a new value to another program (to window number B).

Note:

You could design and test your script process at Script Program Language Editor Window as shown in Figure 6

Script Program Language Editor

Figure 6 Script Program Language Editor Window



Script Program Language Function

Basic Operators

1: Comments are like in BASIC only instead of

' you have a #

2: All constant strings/values must be surrounded by quotes, (especially numbers).

3: To pass data from a function into a variable, use a tilde

~ instead of an =

4: When calling functions always use ()'s

5: When doing comparisons it's safest to put ()'s each side like: if ("10") > (i)

- There are four **Operation** operators in Basic:

Operator	Operation
^	Exponentiation
* /	Multiplication and division
\	Integer division (truncates)
+ -	Addition and subtraction

- There are six **Comparison** operators in Basic:

Operator	Comparison
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
=	Equal to
<>	Not equal to

Basic Functions

Function	Description
= =	String equal to
^	Use and make an arithmetic figure promote the time square of becoming the index number
&	String Concentration

Function	Description	Sample
Abs ("number")	Return an Absolute value of a number	Demo8.scr
Asc ("string")	Return an ASC value of a string	Demo7.scr
Atn ("number")	Return a numeric arctangent value	Demo8.scr
Bold ()	Enables the use of boldfaced text in the input window.	Demo4.scr
BorderStyle ("number")	Enables the use of border style in the window. Number = 0 ~ 5.	Demo16.scr

Cls ()	Clearance for the generated graphics and character in window.	Demo4.scr
Chr ("number")	Character corresponding to a given ASCII or ANSI code	Demo7.scr
Cos ("number")	Cosine of an angle	Demo8.scr
End	End statement	Demo2.scr
Else	If...Else...End statement	Demo2.scr
Exp ("number")	Returns e (the base of natural logarithms) raised to a power	Demo9.scr
Fix ("number")	Return to numeric integer part	Demo10.scr
Getcaption ()	Return window Caption string	Demo17.scr
GetCurrentx ()	Return current x position of text	Demo17.scr
GetCurrenty ()	Return current y position of text	Demo17.scr
GetWidth ()	Return the width value of window size	Demo4.scr
Getheight ()	Return the high value of window size	Demo4.scr
Goto #direction:	Goto statement	Demo2.scr
Hide ()	Hide window	
Hex ("number")	Return Hex value.	Demo11.scr
Variable = Hookincall	Pass a value from outside into a variable	Demo1.scr
Hookoutcall (variable)	Pass a value to outside	Demo1.scr
Variable ~ InputBox (expression, title, pre-input")	Displays a prompt in a dialog box, waits for the user to input text or click a button, and returns the contents of the text box.	Demo3.scr
Italic ()	sets the font style of the Font object to italic	Demo4.scr
InStr ("start number", "string variable", "target string")	Returns the position of the first occurrence of one string within another.	Demo7.scr
If (variable) = (expression)	If...Else...End statement	Demo2.scr
In ("address")	Read Hardware Interface Input value	Demo15.scr
Int ("number")	The value to be represented by the Integer	Demo10.scr
LCase ("string")	Returns a string that has been converted to lowercase	Demo7.scr
Line ("start x", "start y", "end x", "end y")	Places a one-pixel-wide line that stretches along the diagonal of the rectangle.	Demo17.scr
Len ("string")	Number of characters in a text string	Demo7.scr
Log ("number")	Returns the natural logarithm of a number	Demo12.scr
Mid ("string", "start", "length")	Selected portion of a text string	Demo7.scr
MsgBox (expression, show icon, title)	Displays a message in a dialog box, waits for the user to click a button, and returns a value indicating which button the	Demo3.scr

	user clicked.	
Oct ("number")	Returns a string representing the octal value of a number	Demo11.scr
Out ("address", "value")	Write Hardware Interface Output value at Address	Demo15.scr
Print (variable)	Print variable in window	Demo4.scr
Picture ("path/file name")	Load picture to the window	Demo16.scr
Randomize ()	generating Random number	Demo13.scr
Rnd ("number")	Random number	Demo13.scr
Show ("title string")	Show window	Demo16.scr
Strike ()	Set text in strike-through type	Demo4.scr
Shell ("path & file name")	Runs an executable program	Demo6.scr
Sgn ("number")	Returns an integer indicating the sign of a number	Demo14.scr
Sin ("number")	Sine of an angle	Demo8.scr
Sqr ("number")	Square root of a number	Demo12.scr
Setwidth (variable)	Set window width	
Setcurrentx (variable)	Set the text x position for Print.	Demo4.scr
Setcurrenty (variable)	Set the text y position for Print.	Demo4.scr
Setheight (variable)	Set window high	
Setforecolor (variable)	Set Foreground color	
Setbackcolor ("number", "number", "number")	Set the window background color. (0 ~ 255, 0 ~ 255, 0 ~ 255)	Demo16.scr
Setcaption (variable)	Set window Caption	
SetFont (variable)	Set window font	
Textwidth (variable)	Return width value of string.	Demo4.scr
Textheight (variable)	Return high value of string.	Demo4.scr
Tan ("number")	Returns the tangent of an angle.	Demo8.scr
UCase ("string")	Returns a string that has been converted to uppercase	Demo7.scr
Underline ()	Enables the use of underline text in the input window.	Demo4.scr
Var variable, variable,...	Define variable	
While...End	Loop statement	Demo5.scr

Defining Your Own Constants

- You can also define your own constants for use in Macro Editor. The format for defining a constant named **PI** with a value **3.14159** is:

```
Var PI
PI = 3.14159
```