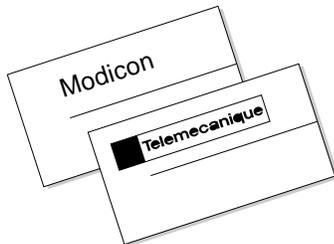
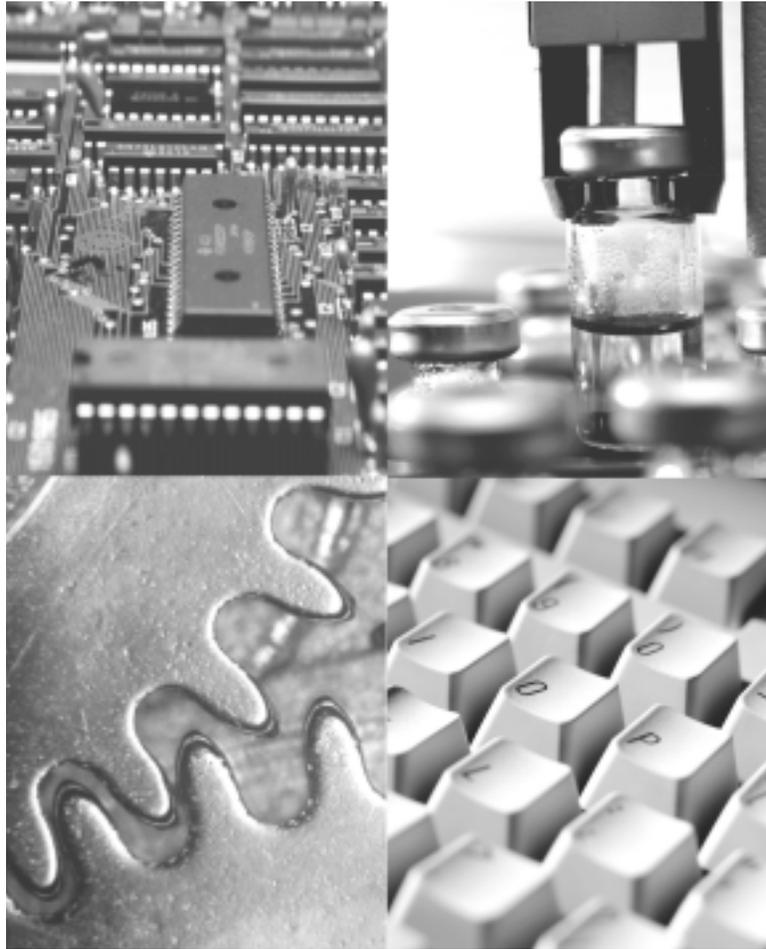


Modicon

ModLink User Guide

890 USE 129 00 Version 2.0



GROUPE SCHNEIDER

■ Modicon ■ Square D ■ Telemecanique

Data, Illustrations, Alterations

Data and illustrations are not binding. We reserve the right to alter products in line with our policy of continuous product development. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us by e-mail at techcomm@modicon.com.

Training

Schneider Automation Inc. offers suitable further training on the system.

Hotline

See addresses for Technical Support Centers at the end of this publication.

Trademarks

All terms used in this publication to denote Schneider Automation Inc. products are trademarks of Schneider Automation Inc.

All other terms used in this publication to denote products may be registered trademarks and/or trademarks of the corresponding corporations. Microsoft and MS-DOS are registered trademarks of Microsoft Corporation. Windows is a brand name of Microsoft Corporation in the USA and other countries. IBM is a registered trademark of International Business Machines Corporation. Intel is a registered trademark of Intel Corporation.

Copyright

All rights are reserved. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including copying, processing or by online file transfer, without permission in writing from Schneider Automation Inc. You are not authorized to translate this document into any other language.

© 1997 Schneider Automation Inc. All rights reserved.

Modicon

ModLink User Guide

890 USE 129 00 Version 2.0

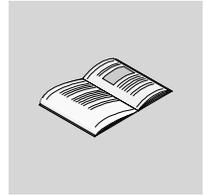
February 1999



GROUPE SCHNEIDER

■ Modicon ■ Square D ■ Telemecanique

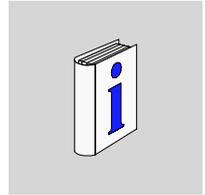
Contents



	About This Book	1
Chapter 1	Overview of ModLink	3
1.1	Parts Complement	3
1.2	System Requirements	4
1.2.1	Platform	4
1.2.2	Software	4
1.3	What ModLink Does	5
1.4	Context-Sensitive Help	6
Chapter 2	Installing ModLink	7
Chapter 3	Using ModLink	9
3.1	Data Definition	9
3.2	Main Menu Functional Selections	10
3.3	ModLink Text Files	11
3.4	Modsoft Files	16
3.5	Direct Addressing	17
3.6	Setting the Link	19
3.7	Inserting a Link	21
Chapter 4	Customizing ModLink	25
4.1	Modicon.INI File	25
4.2	[ModLink Settings]	26
4.3	[AutoLoad]	27
4.4	[Ports]	28
4.5	[Modbus Spy]	30
4.6	[DDE Spy]	31
Chapter 5	Troubleshooting	33
5.1	Mapping Error	33
5.2	Symbol Name Mismatch	34

5.3	Invalid File	34
5.4	Modbus Plus Fault	35
5.5	No ModLink Execution	35
5.6	Link Interruption	35
5.7	Topic Specification	36
5.8	Data Log Assist	36
5.9	Starting ModLink from a Client Application	36
	Index	37

About This Book



Revision History

This version of the ModLink User Guide includes corrections and more detailed instructions.

Document Scope

The focus of this document is to explain how to install and use the ModLink application. Concepts or details of operation of DDE, Windows, Modsoft, or Modbus communications are not discussed in this document.

Validity Note

This manual is valid for ModLink 2.1.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techcomm@modicon.com.

Overview of ModLink



1.1 Parts Complement

There are three distinct ModLink products, which contain the following components:

ModLink Software (Full Function)	352SMD49300
ModLink Diskette	043506377
ModLink User Manual	GM-MLNK-001
License Agreement	GI-0001-003
ModLink Lite (Modbus support only)	352SMD49310
ModLink Lite Diskette	043506378
ModLink User Manual	GM-MLNK-001
License Agreement	GI-0001-003
ModLink Trial (Only 16 references of each type)	352SMD49320
ModLink Trial Diskette	043506379
ModLink User Manual	GM-MLNK-001

1.2 System Requirements

1.2.1 Platform

Your PC must be able to support Windows 3.1 at a minimum and function with the complete range of display adapters. You must have approximately seven megabytes of disk space available.

The PC can run OS/2, for which ModLink will run in a Windows compatibility box.

ModLink 2.1 has been qualified under Windows 95.

The communications protocol may be Modbus, Modbus Plus or TCP/IP.

To run Modbus Plus, you must have an SA85 card. To run TCP/IP, you must have an Ethernet card and a TCP/IP protocol stack.

ModLink also supports the CyberLogic Windows 95 Modbus Plus driver.

1.2.2 Software

Run the ModLink Application in Windows 3.1 or greater.

The target application for data may include any DDE compliant applications such as:

- Excel 4.0 or greater
- Microsoft Word for Windows
- Visual Basic

Your data source program should be running in a PLC.

1.3 What ModLink Does

ModLink is a Modicon Dynamic Data Exchange (DDE) Client/Server application.

In its simplest form, ModLink is the conduit through which data created by separately executing applications can be remotely accessed or altered. This form of communications, in Microsoft Windows terminology, is called remote linking.

ModLink uses the Windows based *Copy* and *Paste (Special)* technique to create the links. The nature of the Windows operation creates a command structure for directing the exchange between the *source* (Server) application and the *target* application (Client). The dynamic nature of the exchange is such that once the link is defined, changes in the specified source data are immediately reflected in the other application because the connection is maintained.

The link definition, in essence, becomes a linking formula (begins with =) that defines:

1. Which application is the *server* followed by a | character.
2. An alias to define the PLC (*topic*) followed by a ! character.
3. Direct reference or symbol (*item*) of where actual data is.

For example: **Server|Topic!Item** could look like

Modlink|PLC!Symbol or

Excel|[filename]sheetx!cell

Figure 1 is a very basic conceptual configuration in which ModLink may be employed. You will find the steps necessary to get data **from** a controller, into an Excel spreadsheet or to route Excel based data **to** the controller, in chapter 3.

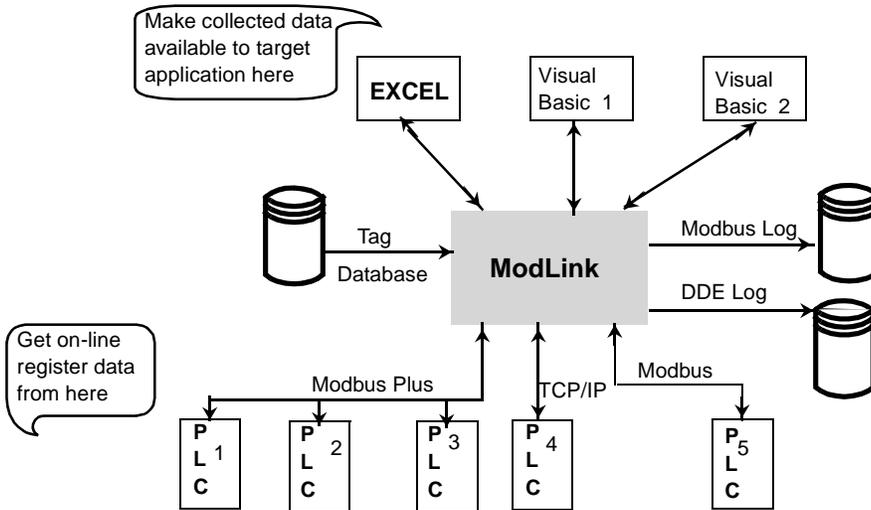


Figure 1 Simplified ModLink Concept

1.4 Context-Sensitive Help

ModLink operates in a standard Windows dialog box. Each operator window you happen to be working with can provide instant documentation, regarding the function, by clicking on the dialog **Help** button or pressing the **F1** key while the cursor is in a variable dialog field.

Installing ModLink

2

The two ModLink diskettes contain all of the executable Dynamic Link Lists (DLL's) and examples that you will need.

To install ModLink, execute the program called **Setup** on the installation floppy. The installation is a Windows-based program. You can execute it from the Windows Program Manager or from DOS.

If you execute the Setup program from DOS, Windows will automatically be started. Total time for the installation procedure should be less than five minutes.



Procedure: Installing ModLink

Run **A:setup** from the Windows File menu or from DOS.

Setup will ask you for the destination disk and directory where ModLink should be installed.

After Setup has run you should have a new Windows group called Modicon Products. It should contain the **ModLink** icon and the icons for the supplemental Modicon applications, **MODSCOPE** and **MODTREND**. To start ModLink (or either support application), double click on the icon.

File Allocation

After the installation is complete, you should have the following files (and their destination directories):

FILE	DIRECTORY	COMMENT
ModLink.EXE	\modlink	ModLink Executable
ModLink.HLP	\modlink	ModLink Help File
Qdebug.DLL	\windows\system	Debug
MLres.DLL	\modlink	ModLink Resource DLL
Read.me	\modlink	Application notes

FILE	DIRECTORY	COMMENT
Modicon.INI	\windows	Initialization file
BC30RTL.DLL	\windows\system	Borland Runtime
GLCV31BW.DLL	\windows\system	Commonview Library
GLFS31BW.DLL	\windows\system	Commonview Library
MBPLUS.DLL	\windows\system	Modicon Modbus Plus
C4DLL.DLL	\windows\system	Codebase 5.0 Library
CTL3DV2.DLL	\windows\system	3D Control
GSW.EXE	\windows\system	Auxiliary File For Sample
THREED.VBX	\windows\system	Auxiliary File For Sample
GRAPH.VBX	\windows\system	Auxiliary File For Sample
ANIBUTON.VBX	\windows\system	Auxiliary File For Sample
GAUGE.VBX	\windows\system	Auxiliary File For Sample
GSWDDL.DLL	\windows\system	Auxiliary File For Sample
VBRUN300.DLL	\windows\system	Auxiliary File For Sample
GLCR31BW.DLL	\windows\system	
ONLN984X.DLL	\modlink	
NCC_BASX.DLL	\modlink	
NCC_R.DLL	\modlink	
ONL984_E.DLL	\modlink	
GLCVXXBW.DLL	\modlink	
TIMEML.MLT	\modlink\examples	Example Text Database
ModLink5.XLS	\modlink\examples\excel	Example Excel 5.0 Sheet
DDE.MAK	\modlink\examples\vb	Visual Basic Example
DDEDESN.MAK	\modlink\examples\vb	Visual Basic Example
EXECUTE.FRM	\modlink\examples\vb	Visual Basic Example
DDEDESIG.FRM	\modlink\examples\vb	Visual Basic Example
MAIN.FRM	\modlink\examples\vb	Visual Basic Example
DDE.EXE	\modlink\examples\vb	Visual Basic Example
DDEDESN.EXE	\modlink\examples\vb	Visual Basic Example
DDE.BAS	\modlink\examples\vb	Visual Basic Example
MODSCOPE.EXE	\modlink\examples\apps	Sample Application
MODTREND.EXE	\modlink\examples\apps	Sample Application

Using ModLink

3

3.1 Data Definition

ModLink can function as a dynamic data exchange (DDE) client or server without having any database access loaded. Without such data, however, it can not access any controllers or references symbolically.

ModLink provides a mapping from a symbolic PLC or reference name by using one of its data definition files. ModLink supports two types of data definition files in the Windows environment:

1. User defined ModLink Text File (.MLT)
2. User designated Modsoft Program Environment file (Program_name.ENV)



Note: Although you can use a text editor to create the data definition file, the ModLink Main Menu Edit dialog provides all the data definition editing you will need.

3.2 Main Menu Functional Selections

Modicon ModLink provides user selectable functions from a Main Menu dialog.

Modicon ModLink						
File	_Edit	Search	Options	Log	Window	Help
New	C opy	F ind		DDE	C ascade	Index
Open	Paste Link	S earch Again		M odbus	T ile	A bout
Close	Insert Link				A rrange Icons	
Save	A dd				C lose All	
Save As	D elete					
Exit	Poll Time					
	Symbol					
	PLC					
	Serial Port					
	Links					
	Show Inactive Links					

Figure 2 Summary of Main Menu Functional Layout

Pulldown functions not available due to prerequisite conditioning are shaded.

3.3 ModLink Text Files

You can define a database definition called `ModLink Text Files` or `ModLink Database File` using your choice of path and *filename* with an extension of `.MLT`.



Note: Filename should be limited to eight characters plus the extension. If the filename is completed and not saved, the incomplete structure can cause errors. These errors occur when ModLink is started if the file Autoload Option is set where neither the file nor Data base can be opened. You can delete the faulty filename using the *Options* menu.



Figure 3 New File Dialog With Path Shown As An Example

Inside a ModLink text file definition there are three sections, set off by identifiers similar to a Windows INI file. The three sections are:

filename.mlt

- 1) **[PLC]**
; field definition here
- 2) **[TAGS]**
; field definition here
- 3) **[Links]**
; field definition here



Note: Beginning with release 1.2 of ModLink, field definitions start with a digit followed by an = sign for [PLC] and [TAGS] sections.

Comment lines begin with a semicolon ; and continue to the end of the line.

[PLC]

The **[PLC]** section defines PLC symbolic names, and their mappings to physical devices. The format of **Add** or **Edit**PLC data in the PLC section is:

Figure 4 Example of Dialog To Add PLC Definition



Note: If the selected port is TCP/IP, the Address field will read: TCP/IP Node Name or IP Address.

Name Port Address

The Name field is the symbolic name that you assign to this specific PLC. It must be 32 or fewer characters. It may then contain any combination of letters, digits and underscores. Case is not significant. Spaces are not allowed.

The port field determines the PC port to which this PLC is attached. This field contains a four or five character designation for the port.

If this PLC is to communicate using the Modbus protocol over a serial port, then this field should contain one of:

COM[1- 8] (This is COM1, COM2, COM3 ... COM8).

If this PLC is to be accessed using Modbus Plus communications, the field should contain one of:

MBP[0-3] (this is MBP0, MBP1, MBP2, MBP3).

If this PLC is to be accessed via TCP/IP, the field should contain:

TCP/IP

This field is not case sensitive. The low level parameters for these ports are defined in the MODICON.INI file. For serial ports, these parameters include the Baud Rate, Parity, Stop Bits, Data Bits and Time Out Parameters.

For Modbus Plus controllers, the MODICON.INI file will determine the Software Interrupt of the Modbus Plus card. If you haven't defined these parameters for a serial port, ModLink will scan the WIN.INI file for these parameters.

The Port field will read TCP/IP Node Name or IP Address if the port is TCPIP. A symbol name, resolved by the DNS, or the IP address must be entered here.

The Address field specifies the communications address of the PLC. For a Modbus controller, this is a decimal number in the range of [1-247]. For a Modbus Plus controller, this should contain as much information as is necessary to access the proper PLC. The valid address range for Modbus Plus is 1...64. The address could range from a single numeric address to a fully qualified Modbus Plus path (i.e. 12 or 12.0.0.0.0 to access the controller at address 12).

Example [PLC] Entries

Given: Time is a Modbus Controller connected to COM1, at address 1

Then: TIME COM1 1

Given: Mixer is a Modbus Plus PLC on adapter 0, at address 12

Then: MIXER MBP0 12.0.0.0

Given: Shaker is a Modbus Plus PLC, adapter 1, address 14, via a bridge

Then: SHAKER MBP1 1.14.0.0.0

Given: Packet is a PLC on a TCPIP Internet address

Then: PACKET TCPIP 205.167.7.22

[TAGS]

Entries in the **[TAGS]** field require more detail. These entries have the following format and can be created using **Edit|Add|Symbol** :

Name Plc Type Address Display Format NumberOfReferences

The screenshot shows a dialog box titled "Create A New Symbol". It has the following fields and values:

- Name:** ERR_CODE
- PLC:** plc1
- Type:** Output Register
- Format:** Integer
- Bit:** 5
- Address:** 2
- Number Of References:** 7

Buttons at the bottom: OK, Cancel, Help.

Figure 5 Example of Dialog to Create a Symbol

The **Name** field is the symbolic name that you would like to assign to this reference. It must be 32 or fewer characters. It may then contain any combination of letters, digits or underscores. Case is not significant. Spaces are not allowed.

The **Plc** field is the name of the PLC with which this reference is to be associated. It should match one of the names in the [PLC] database specifier. Case is not significant here either.

The **Type** field determines what kind of PLC reference is to be accessed. Type is a single digit field with one of the following values:

- 0 (zero) Coil
- 1 (one) Discrete
- 3 (three) Input Register
- 4 (four) Output Register
- 6 (six) Extended Memory (where supported)

The **Address** field determines which one of the discretives or registers will be accessed. The reference is a decimal number in the range of [1-65535]. Leading zeros are not significant. For example, if you wanted to access register 400002, you would select `Output Register` in the type field and enter 2 in the address field. Attempting to access a register or coil beyond the configured range of the controller will result in a Modbus error, and the DDE access will not be resolved properly.

The **Bit** field will appear when you have selected `Input Register` or `Output Register` in the type field. Enter the bit number to be accessed, from 1 to 16, or 0 if the entire register is desired.

The **File** field will appear when you have specified `Extended Memory Register` in the type field. Enter the number of the extended memory file to be accessed, from 0 to 9.

The **Format** field specifies how ModLink will package the data values to be sent to the requesting application. This is a 1 or 2 character field. Case is not significant. This field may be omitted, in which case ModLink will default to integer. ModLink supports the following fields:

i	Integer	1 Register or Discrete
u	Unsigned Integer	1 Register or Discrete
ld	Long Decimal	2 Registers
li	Long Integer Modulo 10000	2 Registers
lu	Long Unsigned Integer	2 Registers
f	Floating Point	2 Registers (single precision)
a	ASCII	1 Register (2 characters max)



Note: Review the data type requirements for your particular application before passing it in a ModLink session.

The **Number of References** Field defines how many of the specified types to access. If the native type accesses two registers then the total number of registers accesses will be two times the specified number of references.

For instance, if the number of references field is 4 and the type is `i` then four registers will be accessed. If the number of references field is 4 and the type is `f`, then 8 registers will be accessed as 4 floating point values.

If the reference address specified a bit field, then the number of references becomes the number of bits to access in the bit field. The number of references should not exceed 17; if it does, ModLink truncates.

3.4 Modsoft Files

ModLink also has the ability to access controller internal files based on Modsoft (revision 1.12 or later) data files. This is done in a two step approach.

To specify the Modsoft file, you instruct ModLink to open the associated .ENV file. ModLink then parses the .ENV file to determine the controller parameters and mode of communication. A PLC alias is assigned that is the base name of the .ENV file. For example using the ModLink initial dialog window File|**Open** you can direct ModLink to open:

c:\modsoft\Programs\mixer\mixera.env

ModLink will assign the PLC an alias of **mixera**.

ModLink extracts the name and path of the *Reference Symbol file* from the .ENV file. The reference symbol file is opened and the symbol data is read and posted to the PLC Name/Symbol Name dialog window. The default display format is signed integer.



Note: When ModLink performs a SAVE or SAVE AS... operation, it is only able to write files in ModLink Text (.MLT) format.

3.5 Direct Addressing

If you do not enter the DDE specification via a COPY/PASTE (Special) operation you can take advantage of direct addressing to specify either a *Topic PLC* or an *Item Symbolic Reference*. If you use direct addressing to specify a Modbus Plus controller, you must have a corresponding entry for the port in the MODICON.INI file. Otherwise, ModLink will not be able to properly resolve the controller's parameters. Some applications may require you to specify the direct addresses in single quotes.

The format for directly addressing a TOPIC PLC is: `' :Port,Address '`

The leading colon is significant. It tells ModLink that this should not be resolved as a symbol. When accessing a Modbus PLC, the Port parameter should be one of :COM[1-8]. If there is no entry in the MODICON.INI file for the port parameters, ModLink will try to get them from the WIN.INI file. When accessing MB+, the port should be defined as one of :MBP0, 1, 2 or 3. The address should be a valid Modbus or Modbus Plus address. If you specify a port using direct addressing, then the ITEM specification for the reference must also be a direct address.

The format for directly addressing an ITEM reference in an input or output register is:

`` :Reference,NumberOfReferences DisplayFormat '`

The format for directly addressing an ITEM reference in an extended memory register is:

`' :6,File,Offset,DisplayFormat,NumberOfReferences '`

The leading colon may be omitted. The Reference field is a 984 type reference where the leading digit implies the type. For example 01 is Coil 1, 40002 is output register 2, etc. The Display Format is the one or two character description that matches the previously described display formats from the database files. It may be omitted in which case it defaults to integer. The NumberOfReferences field is the number of references of the specified display type to be accessed. This can be omitted in which case it defaults to 1.

Example **Direct Addressing with Microsoft Excel**

While using Microsoft Excel you may type the following specifications into a cell. As long as ModLink is running, the data will appear in the cell associated with the specification after you respond with the OK in the ModLink DDE Window, which appears when you terminate the cell specification with a carriage return.

```
=ModLink|':com1,1'!'40001'
```

This instructs ModLink to access the controller attached to COM1, at PLC address 1 and return the contents of register 40001.

Another example from Excel:

```
=ModLink|Mixer!'40002,1 f'
```

This tells ModLink to access the contents of register 40002 as 1 floating point value. ModLink will search its open databases for a PLC named "Mixer" and use the parameters associated with that controller for communications.

An extended memory example from Excel follows:

```
=ModLink|Mixer!'6,1,0,i,5'
```

This tells ModLink to access the contents of extended memory register 61000 (file1, offset 0) as 5 integer values.



Note: Remember to end the formula with `Shift+Control+Enter` for multiple references.

3.6 Setting the Link

In general, working with Excel as either a Client (session initiator) or a Server (session responder) is accomplished by properly filling in the dialog or cell formula. If Excel is to be the Client (and thus ModLink is the Server), start with ModLink:



Working with Excel as the Client

1. Run ModLink, opening up the necessary PLC data base definition
2. Select a symbol from the PLC data base that you want to monitor in Excel.
3. Select the Edit|**C**opy command
4. Go to the Program Manager and start Excel.
5. Position your cursor on the cell where you want the value of the selected (copied) symbol to appear. (If you're copying five registers, select five cells.)
6. Select the Edit|**P**aste **S**pecial command and choose the **Paste Link** (radio button) in the secondary dialog box. (This is Edit|Paste Link in Version 4.0). The value of the symbol will be displayed, in the cell, as it changes.

Using the example provided, you can create DDE pokes to write data to registers through ModLink. Alternatively, you can use Excel as the Server and ModLink as the Client:



Procedure: Using Excel as the Server

1. From Excel, select the cell where you will type new values.
2. Select Edit|**C**opy
3. Start ModLink
4. Display the data definition window (File), **plc**, and **symbol** associated with the register or coil you want written to.
5. Select Edit|**P**aste **L**ink

6. Observe the incoming DDE syntax on the top of the Paste Link window, Application, Topic PLC, and Item Reference. For example, you will see something in the format:

```
EXCEL|[Book1]sheet1!R1C1
```

You then click on the OK button.

7. Now return to Excel. Re-select the cell and change the value (followed by a carriage return). The respective PLC register or coil is changed.



Note: When Using Excel version 5, make sure you have checked the selection 'Update Remote References' from the Tools|Options|Calculation Dialog Box.



Tip: Don't forget, if you want to retain this formula for reuse, you must save the program.

Linking arrays

If you wish to link to references that are arrays, select a group of cells first, then perform the Edit|Paste Special. Alternately, you can select the group of cells and type in the DDE Syntax directly, ensuring that you conclude with Shift+Ctrl+Enter, not just ENTER. An example of the DDE Syntax from Excel is:

```
=ModLink|Time!'40001,i 4' [Shift+Control+Enter]
```

This statement requests 4 registers starting at 40001 in signed integer format from the PLC named `Time`.

You can also create DDE Links while in Design Mode either by filling in the Link**Server**, Link**Topic**, and Link**Item** properties appropriately, or by selecting a symbol in ModLink and performing a Edit|**Copy** and then selecting the object in **Visual Basic** that is the destination and performing a Edit|**Paste** (Special) Link. This method is demonstrated with the DDEDESN example included on the distribution disk.

This operation works similarly if a Visual Basic form is the DDE Server and ModLink is the Client. In this case, you first select the form, set its LinkMode parameter to 1, and perform an Edit|**Copy**, then go to the appropriate symbol in ModLink and perform an Edit|**Paste** Link.

3.7 Inserting a Link

When you are actively engaged in a data linked session you may create another link, including Symbol identification by selecting the **Insert Link** function from the Edit pulldown. This feature minimizes the steps that are otherwise necessary to create a link definition. If your DDE application is to be a server and PLC the client, the field entries may look like the example below:

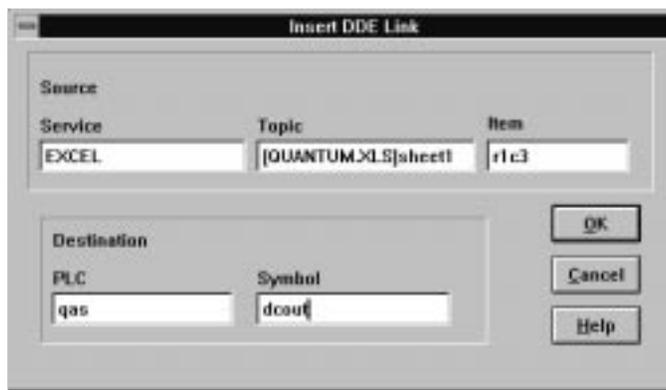


Figure 6 Insert Link Example Dialog

In this example, when you click on the OK button an “Insert Link” message appears in the message area of the dialog box. You can then return to Excel and enter data in the row/column defined (follow with a ↵).



Note: This of course assumes the symbol was created/edited otherwise Excel would not accept the data.

If you happen to be monitoring a PLC variable in the current Excel sheet that is modified by the insert link just created you will see the effect in the cell. If not actively monitoring you may have to display the logic to verify the correct link has been executed.



Note: If you do not include the .xls extension when you define the Excel topic the specified program is not found and a new Excel spreadsheet may be opened.



Note: If ModLink can not resolve the specified Server|Topic it will try to launch the Server application.

Running the Examples

The Microsoft Excel example included only works with Microsoft Excel Version 5.0. Version 5.0 has changed its macro language to Visual Basic Application, which is a derivative of Microsoft Visual Basic. The Excel application, ModLink5.XLS, found in \Modlink\examples\Excel, provides a simple Register Access Panel, allowing you to view and change any symbol or reference in a controller.



Procedure: Using the Microsoft Excel Example File

1. Choose the ModLink file definition for the controller whose activity you want to monitor.
2. From Excel, select the ModLink5.xls file and select the link.
3. Enter the required data in the RAP panel form.

Example

Using the Modsoft Time Example File

For example, if you use a controller program like Modsoft Time example and the ModLink example definition `timeml.mlt` you can define the parameters required to see the seconds register updates as illustrated below:



Note: If the ModLink link is dropped and then re-established you will have to restart Excel.

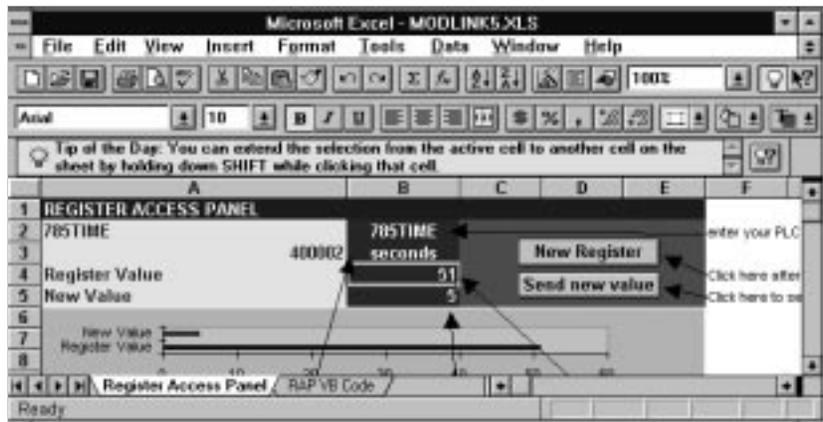


Figure 7 Excel ModLink5.xls Example

Example Using the Microsoft Visual Basic Example

The Microsoft Visual Basic example included with this product - named DDE.exe - is based on the DDE sample included with Visual Basic 3.0. All source files are included so that you may alter the application as you wish. Modicon has added ModLink to the list of DDE sources (servers) to simplify accessing Modbus information from a Visual Basic form.

To run this example:

Add DDE.EXE to your windows filesystem as a New Program Object (Item) and complete the windows program item properties (Edit PIF).

Put the Icon in the group you want then double click on it to execute. The DDE Experimenter dialog is displayed.

Companion Applications

Also included with ModLink are two applications named ModScope and ModTrend. Operating instructions for these two applications are found under their Help menu.

- **ModScope** provides a contact (0 and 1x references) histogram of up to 9 references.
- **ModTrend** provides a trend chart of up to 3 registers.

Customizing ModLink

4

4.1 Modicon.INI File

The Modicon.INI file is a typical Windows .INI file that contains application-specific information, including the areas discussed in this chapter.

There may be sections and entries in the Modicon.INI file that are not specific to ModLink, but have some significance to other Modicon applications.

Because all of the information stored in the Modicon.INI file can be altered by various editing operations within the application (except Modbus Plus interrupt settings), there should be little need to edit this file directly. However, some customers may want to do an edit, so this chapter provides a list of the fields and their meanings.

Data in the Modicon.INI file is not case sensitive.

4.2 [ModLink Settings]

1. Poll Interval Initial default value = 125

Meaning: This is the number of milliseconds that ModLink will wait between DDE operations. Valid poll intervals are from 1 - 32,767 inclusive.

2. MainMode Initial default value = 2

Meaning: This is the initial appearance mode of the main window. Acceptable values are 0 (zero) Maximize the window; 1 (one) Minimize the window; 2 (two) Restore the window to its last state.

3. MainWindow Initial default (random) value = 156 72 624 378 485

Meaning: This field will determine the actual appearance of the Main window if the MainMode parameter is a 2. This is a 5 digit field with the following values:

Mode	X	Y	SizeX	SizeY
------	---	---	-------	-------

The Mode is similar to the main mode. A zero indicates that the main window should be maximized. A one indicates that the window should be minimize. A two means that the window should be placed and sized according to the X, Y, SizeX and SizeY parameters.

4. ChildMode Initial default value = 2

Meaning: This is the mode that specifies the initial appearance of the database windows. It applies to any newly opened databases as well as any databases that are loaded automatically at startup time.

5. ChildfN1 Initial default value = 2 20 1 513 328

Meaning: The [N] is replaced by a digit which matches one of the Load[N] parameters in the AutoLoad section. This field specifies the initial appearance of any automatically loaded database files if the Child Mode parameter is a 2. It has the same parameters as the Main Window entry.

4.3 [AutoLoad]

1. Load[N] Initial default value =
C:\ModLink\EXAMPLES\TIME.MLT

Meaning: The [N] is replaced by a digit. This is a list of database files that should be loaded automatically when ModLink is invoked. The parameter to this field should be a valid DOS file name with a path sufficiently qualified to allow ModLink to access the proper file. For example:

Load0=c:\modsoft\files\mixer\mixera.mlt

2. SaveOpen Initial default value = 1

Meaning: This field determines whether ModLink should add any databases that are open when ModLink terminates to the autoloading list. Valid values are:

0 (zero) - Don't add to the autoloading list

1 (one) - Add open databases to the autoloading list.

3. Confirm Restoring Of Links Initial default value = 1

Meaning: The confirmation dialog allows the user to decide whether a database's client links are to be restored. If this check box is set, a confirmation dialog is presented to the user before the client links of an auto-loaded database are restored. If this box is not set, the dialog will not be presented and client links will be automatically restored when a database is auto-loaded.

4. SuppressCommErrors Initial default value = 1

Meaning: When this line is added to the Modicon.INI file, the Comm Error dialog box is not displayed. The Comm Error, however, will be shown in the Modbus log file if the log file is enabled.



Note: The following parameters can be edited at the ModLink application level. The values are provided here for consistency and completeness.

4.4 [Ports]

1. COM[1-8]

Meaning: These fields determine the mode in which ModLink will open the specified serial port. These entries have the format of:

Port=Port: ,Baud,Parity,DataBits,StopBits Mode

Where:

Port is one of COM1, COM2, COM3...COM8

Baud is the baud rate

Parity is a single letter with the following definitions:

e- even

o- odd

n- none

Databits is the number of data bits. It should be either a 7 or an 8.

StopBits is the number of stop bits. It should be a 1 or a 2.

Mode is the Modbus communications mode. If the number of data bits is 8, this field should be "RTU". If the number of data bits is a 7, this field should be "ASCII".

Example Serial Port Parameters

COM1=COM1:9600,e,8,1 RTU

This defines the parameters for COM1 to be 9600 baud, even parity, 8 data bits, 1 stop bit, RTU communications.

Initial default is Com1,2,3 and 4 @ 9600,parity,8,1 RTU 1000

Com 1 and 2 with even parity and

Com 3 and 4 with No parity

2. MBP[0-3] Default = 5c

Meaning: This is the definition for the Modbus Plus SA85 cards. The entry should be one of MBP0, MBP1, MBP2, MBP3. The format for this field is:

PORT=Interrupt

Where:

PORT is one of MBP0, MBP1, MBP2, MBP3

Interrupt is a 2 digit hexadecimal number matching the software interrupt for the SA85 card. For example:

MBP0=5c

defines MBP0 to be at interrupt 5C.

3. TCPIP=yes

This entry is necessary and must be the first entry under the **[PORTS]** heading if a PLC is to be accessed via TCP/IP.

4.5 [Modbus Spy]

1. Logging Default = Off

Meaning: This field determines if data logging of Modbus messages is to be on or off. Valid values for this field are "ON" or "OFF"

2. Log To Default = Screen

Meaning: If Modbus logging is on, this field determines where the output goes. It may go to a window, a log file, or both. Valid values are "FILE", "SCREEN" or "BOTH"

3. Log Data Default = Interpreted

Meaning: This field determines how the log data is to be presented. The data may be presented as numeric strings of Modbus Commands, interpreted commands or both. Valid values for this field are "INTERPRETED" or "RAW".

Interpreted will show data values. Raw will not show data values.

4. Log File Default = MBUS.log

Meaning: If logging is on, and the output is to a file, then this field specifies the name of the file to contain the log data. Note that this file can get pretty large as each Modbus command gets written here.

4.6 [DDE Spy]

1. Logging Default = OFF

Meaning: This field determines if data logging of DDE messages is to be on or off. Valid values for this field are "ON" or "OFF"

2. Log To Default = Both

Meaning: If DDE logging is on, this field determines where the output goes. It may go to a window, a log file, or both. Valid values are "FILE", "SCREEN" or "BOTH"

3. Log File Default = DDE.log

Meaning: If logging is on, and the output is to a file, then this field specifies the name of the file to contain the log data.



Note: This file can get large because each DDE command is written here.

5.1 Mapping Error

The most common cause for failure is an improper or nonexistent mapping to a controller. A symptom of this would be a “#N/A” in an Excel cell when attempting to specify a link to ModLink. There could be several reasons for this:

- The entry in the PLC section of the database definition file doesn't have an address that matches the actual PLC.
- The parameters in MODICON.INI are either not correct or not present. Make sure that the communications parameters in the MODICON.INI file, or your WIN.INI file, match those for your mode of communications. Remember Modbus Plus will not be able to operate without an entry in the MODICON.INI file.
- The entry in the application's DDE specification does not match a known controller. If the DDE specification was entered via a Copy/Paste operation, this should not be a problem. If it was typed in directly, make sure that the TOPIC field matches a PLC in an open datadefinition file. Case is not significant, but all letters, underscores and digits must match. Also, make sure that a database is in fact opened. If the client application launched ModLink, the database must be specified via an entry in the AutoLoad section of the MODICON.INI file.
- When specifying a direct address, forgetting the leading colon will cause ModLink to fail to recognize the port.

5.2 Symbol Name Mismatch

If you placed the symbol in the client application via a Copy/Paste sequence this should not be a problem. If, however, you directly typed the DDE specification you must make sure that the symbol matches.

Case is not significant, but it must be spelled *exactly* the same, all underscores present, etc.

Also, the data definition must be loaded into ModLink.

If your client application launches ModLink, you should have that data definition as an entry in the AutoLoad section of the MODICON.INI file.

5.3 Invalid File

Attempting to load an invalid file as a data definition may still result in General Protection faults or assertion failures. Make sure that the file you specify to be opened is a valid ModLink Text file (.MLT), or a Modsoft Revision 2.0/higher environment file (.ENV).

The DDE application should either be in your path or its path included in the Topic Specification i.e. Excel|c:\excel\[thisfile.xls]sheetx!r1c1

5.4 Modbus Plus Fault

It is possible that ModLink was not able to open any data paths. If ModLink has previously aborted with a `General Protection Fault` or an `assertion failure` while performing Modbus Plus communications, the paths are not closed properly. This would leave these paths unavailable for future access. The only way to remedy this fault is to exit Windows and reset the SA85 card.

Another possible reason for Modbus Plus failure would be that your specification in the MODICON.INI file does not match the parameters for your Modbus Plus card in your CONFIG.SYS file. Make sure that the interrupts match and that there is in fact a PLC located at the specified address.

5.5 No ModLink Execution

The client application fails to launch ModLink. Make sure that ModLink.EXE is available via the DOS PATH.

5.6 Link Interruption

When launching ModLink from Excel with a severed link, Excel reports an error, then continues. This is because ModLink rejects the first DDE format the Excel specifies. Excel will continue polling until ModLink eventually accepts the TEXT format request.

5.7 Topic Specification

The typed in topic is not matched by ModLink. If, for example, you forgot the leading colon while directly specifying a PLC in an Excel cell, you would get a “#REF!” in the cell. This is because ModLink did not match the topic as a known PLC. Just re-type the specification with the colon.

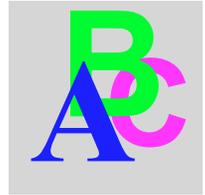
5.8 Data Log Assist

Using the data logging capabilities to trouble shoot. The facilities for logging Modbus and DDE messages, can be an aid in trouble shooting your application. Things to look for in the Modbus log are invalid PLC addresses, access to registers beyond the configured range and unrecognized commands. Things to look for in the DDE log are unknown topics and items. Keep in mind that logging to an output file can produce enough to fill a disk over some period of time.

5.9 Starting ModLink from a Client Application

When using a client application to launch ModLink, there may, in some cases, be a problem when the client application attempts to activate its DDE links. Certain client applications (e.g., Excel 7.0) do not give ModLink enough time to open its auto-load databases before trying to access them and declaring the links as bad. To work around this problem, start ModLink before starting any client applications. Alternatively, with some clients (e.g., Excel 7.0) you can manually "restart" the links after ModLink has successfully opened the appropriate databases. (In Excel this is done by clicking the **Update Now** button in the Links dialog, which is accessed via the Edit | Links menu items.)

Index



Symbols

.ENV, 16
.MLT, 11, 16

A

application path, 34
Autoload, 11
autoload parameters, 27

C

client/server, 5, 19
communications parameters, 12, 28
communications protocol, 4

D

DDE Experimenter, 23
display data code, 15

E

Error #N/A, 33
Error #REF, 36
error on Modbus Plus, 35
Excel example, 22
Extended Memory, 15, 18

L

link definition, 5
log file Characteristics, 36
log file characteristics, 30

M

Modbus, 28
Modbus Plus, 29, 35
Modicon.INI, 13
ModLink initial settings, 26
Modsoft, 16

N

number of references, 17

P

part number, 3
Paste Special, 19
PLC address, 13
PLC alias, 12

R

register type ID, 14

S

symbol tags, 14

T

TCP/IP, 12, 29

V

Visual Basic example, 23

W

Windows, 4

Moticon, Square D and Telemecanique are PLC brand names from Schneider. These products are sold in the US by Square D, in Canada, Latin America, Europe, Africa, Asia/Pacific and Middle East by Schneider, in Germany by AEG Schneider Automation, in China and Persian Gulf by Schneider Automation, in South Africa by ASA Systems Automation, in Austria by Orlithe.

Schneider Automation, Inc.
One High Street
North Andover, MA 01945
Tel: (1) 508-734-0800
Fax: (1) 508-975-9400

Schneider Automation GmbH
Steinheil Strasse 117
D-65300 Seligenstadt
Tel: (49) 6182 81-2584
Fax: (49) 6182 81-2860

Schneider Automation S.A.
2/6, Route des Lucioles-BP 147
F-06903 Sophia-Antipolis Cedex
Tel: (33) 92 96 20 00
Fax: (33) 93 65 57 15