



CIT Engineering nv

Kleinhoefstraat 6 - B-2440 Geel - Belgium
Tel. +32 (14) 57 96 50 - Fax +32 (14) 57 96 51

E-mail : info@citeng.com
Website : www.citeng.com

Bank : CERA 733-3093437-40
BTW : BE 459 001 426



Microsoft Certified
Solution Provider

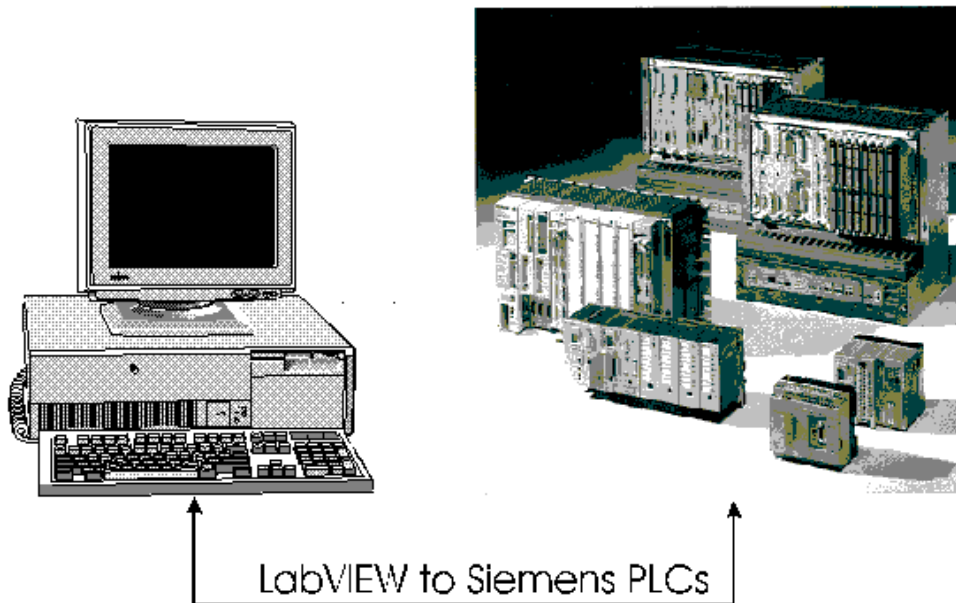
CIT is a member of
the National Instruments Alliance Program

SinecVIEW

RK512 (3964R) protocol for LabVIEW 5.1

User Manual

Windows 3.1/95/NT Mac and PowerMac version



User Manual

SinecVIEW

RK512 (3964R) protocol for LabVIEW 5.1

For :

- AT-PC running DOS/Windows 3.11/95/NT
- Macintosh
- PowerMac

Note:

We are constantly attempting to improve the quality of our manuals and products.

Should you have any suggestions or recommendations for improvement, we would appreciate it if you would send your recommendation to CIT

Copyright by CIT
All rights reserved.

Revision 5.1
November 96



Limited Warranty & Copyright

Important warning

Because of the variety of uses of this product, and the impossibility of our anticipating every use, those responsible for using and applying this product must satisfy themselves that it is appropriate and acceptable for their application. That responsibility includes using this product in accordance with all engineering, programming and safety practices and particularly in accordance with the use of Siemens PLCs as recommended by the Siemens Company.

License

You have a nonexclusive license to use the SinecVIEW software. You may use this product on one computer at a time. For backup purposes only, you may copy SinecVIEW software to other disks which you must retain in your possession and control. You may not copy SinecVIEW software to any other computer by any means for any other purpose.

Limited warranty

CIT warrants both this manual and the SinecVIEW program disks to be free from defects for one year from the date of purchase. If during this warranty period, you find that this manual or the SinecVIEW disks contain physical defects, return the defective item along with proof of purchase and description of the defect. You will receive a replacement at no charge.

CIT makes no warranty or representation, express or implied, with respect to this software, its performance or its fitness for a particular purpose. As a result, the software is sold "as is" and you, The purchaser, assume the entire risk to quality and performance. In no event shall CIT be liable for direct or indirect, special incidental or consequential damages resulting from any defect in the software or its documentation. Such damages include but are not limited to loss or profit, opportunity, program or data.

This warranty excludes all other warranties, oral or written, express or implied.



Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of CIT.

Intended Audience

This manual assumes you are familiar with Macintosh computers or with Windows 3.1/95/NT and the basic Macintosh or Windows operations, such as pointing and clicking, launching an application and moving files. Because SinecVIEW works only with LabVIEW 5.1 software and Siemens PLCs, familiarity with LabVIEW 5.1 and Siemens PLCs is assumed. If you have just purchased LabVIEW 5.1 or PLCs, you should familiarize yourself with those products before you attempt to install or use SinecVIEW.

Product names mentioned in this manual are trademarks or registered trademarks of their manufacturers.



Table of Contents

1. Introduction	5
2. The Siemens RK512 (3964R) Protocol	6
3. Installing the RK512 (3964R) Library	7
3.1. <u>SOFTWARE INSTALLATION</u>	7
3.2. <u>AT-PC WINDOWS 3.1 VERSION</u>	7
3.3. <u>MACINTOSH AND POWERMAC VERSION</u>	8
3.4. <u>AT-PC WINDOWS 95/NT VERSION</u>	8
4. LabVIEW's VIs Description	9
4.1. <u>THE PLCREAD VI</u>	9
4.2. <u>THE PLCSEND VI</u>	13
4.3. <u>THE RW3964R32 VI (ONLY FOR WIN 95/NT)</u>	17
4.4. <u>THE UTILITY VI</u>	19
4.5. <u>SPECIFIC ERROR FOR THE RK512 (3964R) PROTOCOL</u>	20
5. Hardware Connection	21
5.1. <u>PHYSICAL LINK BETWEEN AT-PC AND THE RK512 (3964R) COMMUNICATION CONTROLLER</u>	22
5.2. <u>PHYSICAL LINK BETWEEN A MACINTOSH AND THE RK512 (3964R) COMMUNICATION CONTROLLER</u>	23
6. RK512 (3964R) Example VI	25



1. Introduction

This is a brief manual describing the use of the **RK512 (3964R)** library as well as for an AT-PC running Windows 3.1/95/NT as for a Macintosh.

RK512 (3964R) is a member of **SinecVIEW**, a set of three drivers for LabVIEW 5.1 to communicate with Siemens PLCs of the S5 series.

- Sinec-L1 (AT-PC Windows 3.1/95/NT /Mac and PowerMac)
- Sinec-L2-FDL (AT-PC Windows 3.1 only)
- 3964R (AT-PC Windows 3.1/95/NT /Mac and PowerMac)
- Sinec-L2-DP (AT-PC Windows 3.1/NT)

With SinecVIEW there are four different ways to connect the Siemens PLCs to LabVIEW 3.1.1.

- The first way is by the **Sinec L1** protocol.

The Sinec-L1 VI can communicate on a L1 point-to-point basis for single PLC applications or on a L1 network basis (up to 30 PLCs). The PC is always the Master. The PLC can be connected via a Siemens BT 777 bus terminal or a programming cable (only point-to-point) to a serial port of the PC. No special plug-in board is required for the PC.

- A second way (AT-PC/Windows 3.1 only!) is by the **Sinec L2-FDL** protocol.

The Simatic S5 apparatus can be connected to the PC via Siemens hardware in a network of PLCs and PCs (up to 127 stations). You need a special L2 communication plug-in board in your PC (CP 5412 (A1) or CP 5412 (A2)).

- The third way to access data on the Siemens PLC is by the serial **RK512 (3964R)** protocol.

With a communication processor (for S5: CP524 or CP525 and CPU 928B with RS323C interface module; for S7-400: CP 441-2) that is configured with the RK512 (3964R) protocol it is possible to connect the module directly to one of the serial ports of the PC. No special plug-in board is required for the PC.

- A fourth way (AT-PC/Windows 3.1/NT) is by the **Sinec L2-DP** protocol.

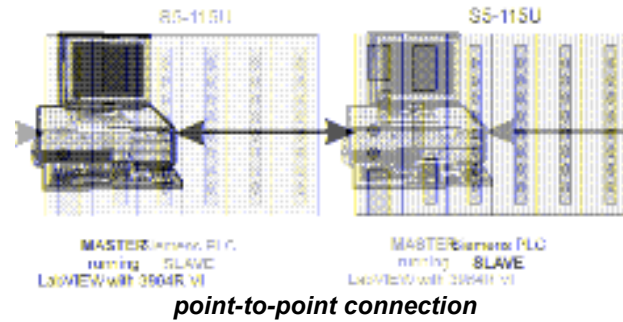
The Simatic S5 and -S7 apparatus can be connected to the PC via Siemens hardware in a network of PLCs and PCs (up to 125 stations). You need a special L2 communication plug-in board in your PC (CP 5412 (A2)).

These drivers can be ordered separate.



2. The Siemens RK512 (3964R) Protocol

The RK512 (3964R) protocol allows Siemens PLCs to communicate with each other or with a PC on a point-to-point basis, i.e. two communication units on a line.



The RK512 (3964R) protocol uses the method of asynchronous transmission via an RS-232 interface. In asynchronous transmission, the transmitter transmits the characters one at a time, with their respective start and stop bits. The receiver knows that each character will be followed by a character which has to be decoded. The stop bit completing the message resets the receiver. About 90-95% of serial data transmission is asynchronous. The transmitter and the receiver must operate with the same baud rate.

With the RK512 (3964R) protocol you can send/receive data blocks with a length up to 2048 data **words**. RK512 (3964R) is a protocol that frames the data into small packets of information that are transmitted one by one. These packets consist of a header, 128 data bytes and a footer.

Whenever a frame is received and the redundancy check fails, the receiver instructs the transmitter to send the data (the whole data block) all over again. Therefore it is better to practice to divide the data into blocks of max. 64 data words in LabVIEW before sending it to the PLC.

3.3. Macintosh and PowerMac version



Files included on the disk.

- PLCSendPPC.VI 3964R send data VI for PowerMac
- PLCReadPPC.VI 3964R read data VI for PowerMac
- PLCSend68K.VI 3964R send data VI for Macintosh
(68xxx)
- PLCRead68K.VI 3964R read data VI for Macintosh
(68xxx)
- Manage PLC data.VI Utility VI
- Example.VI An example VI for 3964R
- Errorh.vi An error handler vi

Install the VI library by simply copying the files from your floppy drive to a location on your local drive.

3.4. AT-PC Windows 95/NT version



Files included on the disk.

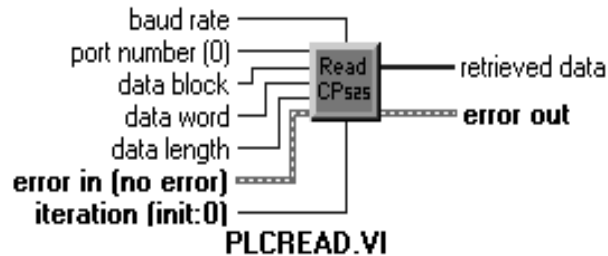
- READ32.VI The 3964R read data VI
- WRITE32.VI The 3964R send data VI
- RW3964R32.VI The 3964R send and read VI
- MANADATA.VI Utility VI
- Errorh.vi An error handler vi
- Example.vi An example VI for 3964R

- 3964r32.DLL DLL for Windows 95/NT

Install the VIs by simply copying them from your floppy drive to a location on your local drive and copy the *.DLL files to your LabVIEW or Windows directory.

4. LabVIEW's VIs Description

4.1. The PLCRead VI



Input wiring

- **baud rate** : baud rate is the rate of transmission (default 9600 Baud)
Not changeable for windows 95/NT.

- **port number (0)** : The serial port you are communicating with.



PC

When you use the serial port VIs under Windows 3.1, the port number parameter can have the following values:

0: COM1
1: COM2
2: COM3
3: COM4
4: COM5
etc.

If you add extra ports to your computer with a plug-in board, be sure that the configuration is correct (base address and IRQ).

MAC

0 =  modem port
1 =  printer port

On the Macintosh, port 0 is the modem, using the drivers .ain and .aout. Port 1 is the printer, using the drivers .bin and .bout. To get more ports on a Macintosh, you must install other boards, with the accompanying drivers.

Remark : *If you use a plug-in board on the Macintosh to get more serial ports, do not forget to add the appropriate values for the 'input driver names' and 'output driver names' into the LabVIEW Global serpOpen.vi*



- **data block** : specifies the data block (DB) in the PLC where you want to retrieve data from.
- **data word** : specifies the offset in the specified data block (DB) to start the read function.

Because the representation of **data word** is one byte, the maximum range is 255. So if you want data from your PLC beyond DW255 , e.g. DW300- DW350, you can't start at a DW past 255.

You can solve this problem by moving the data in the PLC to an other address or by requesting more data than needed (start at DW255 and adjust **data length** to your needs).

- **data length** : number of words to retrieve starting with the specified data word (DW) in the specified data block (DB).
- **error in (no error)** : error in is a cluster that describes the error status before this VI executes. If error in indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its error out cluster. If no error has occurred, then this VI executes normally and sets its own error status in error out. Use the error handler VIs to look up the error code and to display the corresponding error message. Using error in and error out clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.
 - **status** : status is TRUE if an error occurred before this VI was called, or FALSE if not. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code.
 - **code** : code is the number identifying an error or warning. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be

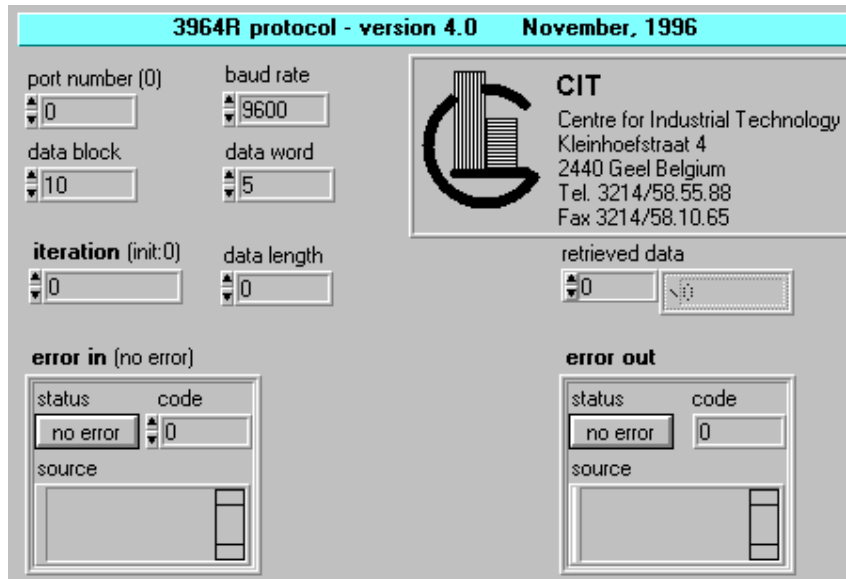
zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.

- **source** : source is a string that indicates the origin of the error, if any. Usually source is the name of the VI in which the error occurred.
- **iteration (init:0)** : iteration controls when the PLCRead VI performs initialization. If iteration is 0, the VI initialize the serial port with the hardware specifications for RK512 (3964R), then performs a transceive cycle. If iteration is greater than zero, the RK512 (3964R) VI assumes that the serial port is already configured and just performs a transceive cycle. You usually wire this input to a loop iteration terminal. Not for windows 95/NT

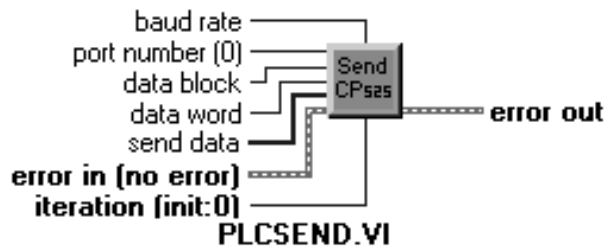
Output wiring

- **retrieved data** : retrieved data from the PLC
- **error out** : error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, error out is the same as error in. Otherwise, error out shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using error in and error out clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.
 - **status** : status is TRUE if an error occurred before this VI was called, or FALSE if not. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code.
 - **code** : code is the number identifying an error or warning. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.
 - **source** : source is a string that indicates the origin of the error, if any. Usually source is the name of the VI in which the error occurred.

The PLCRead front panel



4.2. The PLCSend VI



Input wiring

- **baud rate** : baud rate is the rate of transmission (default 9600 Baud)
Not changeable for windows 95/NT.
- **port number (0)** : The serial port you are communicating with.



PC

When you use the serial port VIs under Windows 3.1, the port number parameter can have the following values:

0: COM1
1: COM2
2: COM3
3: COM4
4: COM5
etc.

If you add extra ports to your computer with a plug-in board, be sure that the configuration is correct (base address and IRQ).

MAC

0 =  modem port
1 =  printer port

On the Macintosh, port 0 is the modem, using the drivers .ain and .aout. Port 1 is the printer, using the drivers .bin and .bout. To get more ports on a Macintosh, you must install other boards, with the accompanying drivers.

Remark : *If you use a plug-in board on the Macintosh to get more serial ports, do not forget to add the appropriate values for the 'input driver names' and 'output driver names' into the LabVIEW Global serpOpen.vi*



- **data block** : specifies the data block (DB) in the PLC where you want to retrieve data from.
- **data word** : specifies the offset in the specified data block (DB) to start the read function.

Because the representation of **data word** is one byte, the maximum range is 255. So if you want to send data to your PLC beyond DW255 , e.g. DW300- DW350, you can't start at a DW past 255.

You can solve this problem by moving the data in the PLC to the appropriate address or by sending more data than needed (start at DW255 and adjust **data length** to your needs).

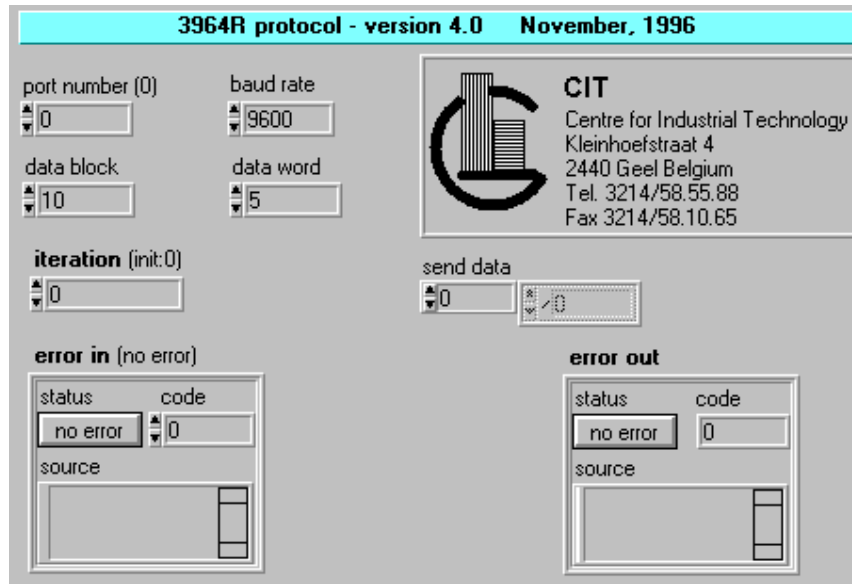
- **data length** : number of words to retrieve starting with the specified data word (DW) in the specified data block (DB).
- **send data** : Data to send to the PLC.
- **error in (no error)** : error in is a cluster that describes the error status before this VI executes. If error in indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its error out cluster. If no error has occurred, then this VI executes normally and sets its own error status in error out. Use the error handler VIs to look up the error code and to display the corresponding error message. Using error in and error out clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.
 - **status** : status is TRUE if an error occurred before this VI was called, or FALSE if not. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code.

- **code** : code is the number identifying an error or warning. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.
- **source** : source is a string that indicates the origin of the error, if any. Usually source is the name of the VI in which the error occurred.
- **iteration (init:0)** : iteration controls when the PLCRead VI performs initialization. If iteration is 0, the VI initialize the serial port with the hardware specifications for RK512 (3964R), then performs a transceive cycle. If iteration is greater than zero, the RK512 (3964R) VI assumes that the serial port is already configured and just performs a transceive cycle. You usually wire this input to a loop iteration terminal. Not for windows 95/NT.

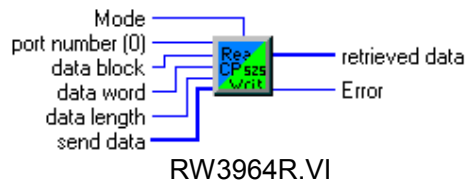
Output wiring

- **error out** : error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, error out is the same as error in. Otherwise, error out shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using error in and error out clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.
 - **status** : status is TRUE if an error occurred before this VI was called, or FALSE if not. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code.
 - **code** : code is the number identifying an error or warning. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.
- source** : source is a string that indicates the origin of the error, if any. Usually source is the name of the VI in which the error occurred.

The PLCSend front panel



4.3. The RW3964R32 VI (only for win 95/NT)



Input wiring

- **mode** : specifies the mode: read data from PLC or write data to PLC.
- **port number (0)** : The serial port you are communicating with.

When you use the serial port VIs under Windows 95/NT, the port number parameter can have the following values:

0: COM1
1: COM2
2: COM3
3: COM4
4: COM5
etc.

If you add extra ports to your computer with a plug-in board, be sure that the configuration is correct (base address and IRQ).

- **data block** : specifies the data block (DB) in the PLC where you want to retrieve data from.
- **data word** : specifies the offset in the specified data block (DB) to start the read function.

Because the representation of **data word** is one byte, the maximum range is 255. So if you want to send data to your PLC beyond DW255 , e.g. DW300-DW350, you can't start at a DW past 255.

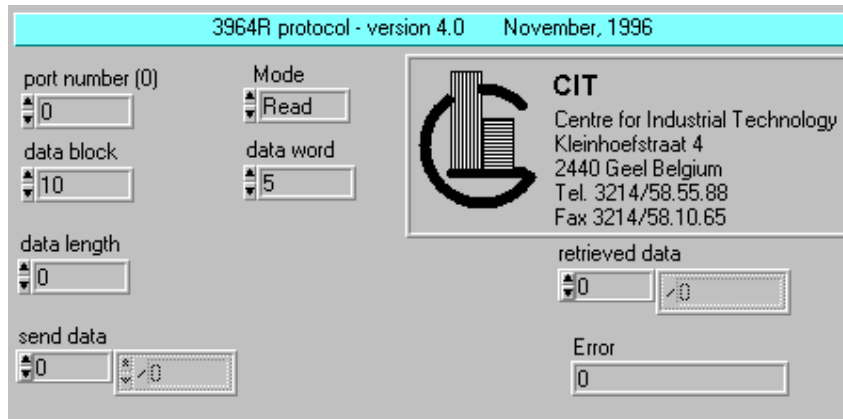
You can solve this problem by moving the data in the PLC to the appropriate address or by sending more data than needed (start at DW255 and adjust **data length** to your needs).

- **data length** : number of words to retrieve starting with the specified data word (DW) in the specified data block (DB).
- **send data** : Data to send to the PLC.

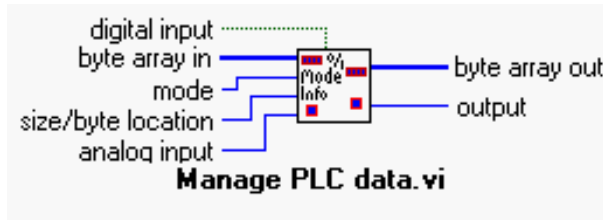
Output wiring

- **retrieved data** : retrieved data from the PLC

- **error** : error out describes the error status after this VI executes. . Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.



4.4. The utility VI



A VI used to write and read data (words/bytes/bits) to and from a data array. This VI is very useful to manage your data to and from the PLC.

There are 7 different operation modes.

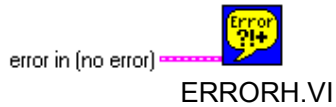
Input wiring

- **digital input** : Boolean input when mode=Write bit.
- **byte array in** : The input byte array.
- **mode** : The mode selection :
 0. Initialize
 1. Read word
 2. Read byte
 3. Read bit
 4. Write word
 5. Write byte
 6. Write bit
- **size/byte location** : Dimension size of the array (*mode=initialize*). Index of the element in the array (*all other modes*).
- **analog input** : Data input when mode=Write word/Write byte.Bit index(of byte) when mode=Read bit/Write bit.

Output wiring

- **byte array out** : Output byte array.
- **output** : Data output when mode = Read word/Read byte/Read bit.

4.5. Specific error for the RK512 (3964R) protocol



This error handler is used primarily to inform the user if an input error exists, to describe the error, and to identify where it occurred. The information for this is derived from the inputs error in, error code, and error source, and from an internal error description table. The table describes all errors that can be created by LabVIEW or its associated I/O operations. The handler has provisions to take alternative actions, such as to cancel or set an error status, and to test for and describe user-defined errors.

From PC ->PLC

- 0 : No error
- 7101 : PLC does not respond
- 7102 : PLC did not returned the correct response
- 7103 : Got NAK from PLC with building communication link
- 7104 : Got NAK from PLC after sending data to the data
- 7105 : PLC did not returned the correct response after sending the data
- 7108 : PLC did not respond with the DLE after sending the data to the PLC

From PLC ->PC

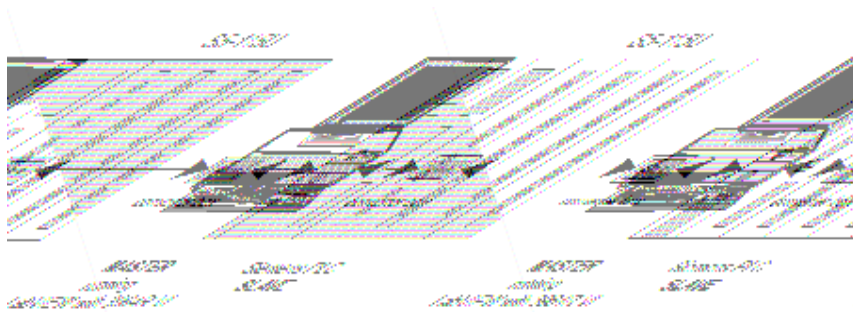
- 7109 : PLC does not respond with STX.
- 7106 : Got NAK from PLC instead of STX.
- 7107 : PLC did not start the reaction response correctly.
- 7110 : Did not received the correct header from PLC.
- 7111 : Did not received the correct footer from PLC.
- 7112 : BCC is not correct.
- 7115 : Did not received the correct data from the PLC.

PLC header errors

Negative errors are PLC header errors and no protocol errors.

5. Hardware Connection

The RK512 (3964R) communication controllers use a RS-232 serial interface, this is the most common interface for data communications. RS-232 recommends that the cable should be not longer than 15 meters. At greater distances, up to several kilometers, or in noisy environments short-haul modems can be used to transform the RS-232 signal into a signal that is less vulnerable to interference.



using modems for longer distances

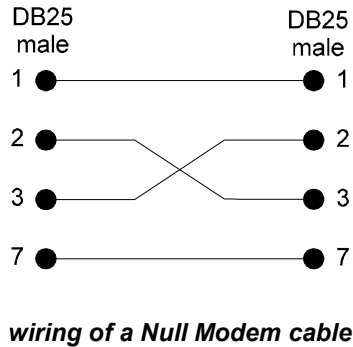
The cabling from your computer to the PLC communication interface depends on the PC's serial interface.

Of course, SinecVIEW will not work properly unless the Siemens PLC is first installed correctly. To minimize problems, try to verify your Siemens PLC configuration before attempting to install SinecVIEW.

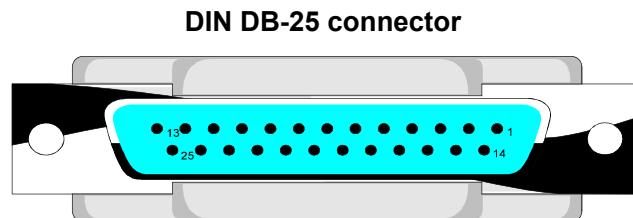
Important warning : *When installing or removing interface cables, always power down the PLC, the interface module and the computer. Otherwise, you may damage the computer, the PLC or the interface module.*

5.1. Physical link between AT-PC and the RK512 (3964R) communication controller

An AT-PC has standard serial ports with an RS-232 interface. So you can easily connect your computer to the PLC using a so called Null Modem cable. The picture below shows the wiring of a Null Modem cable.



The picture below shows the pin assignment of a DB-25 connector as most commonly used RS-232 interface connector.

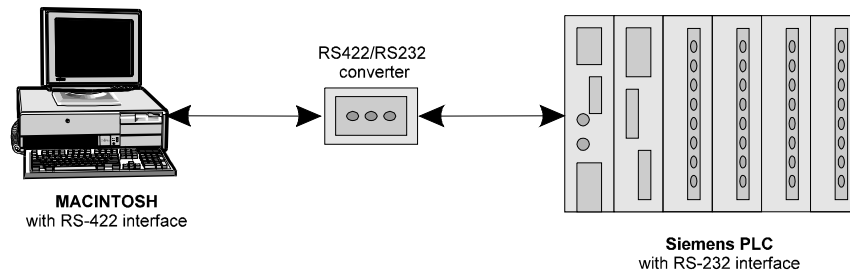


1 : Shield	4 : RTS	7 : Signal ground
2 : TxD	5 : CTS	8 : DCD
3 : RxD	6 : DSR	20 : DTR

The pin assignment of a DB-25 connector

5.2. Physical link between a Macintosh and the RK512 (3964R) communication controller

A Macintosh PC normally has two serial ports with an RS-422 interface on board. So you need to transform the RS-422 signal to a RS-232 signal with a interface converter before you can connect your computer to the PLC.

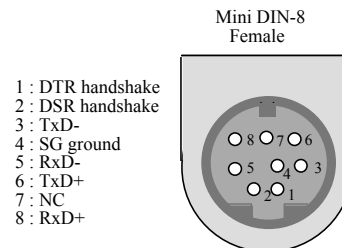


Remarks :

- You need a 4-wire RS-422 converter.
- Other names for RS-422 to RS-232 converter : RS232/RS422-RS485 converter or V24/V11 converter.
- *RS422 is electrical compatible with RS485.*
- *We use the Phoenix Contact PSM-V24/V11 converter.*



The picture below shows the pin assignment of the mini DIN-8 connector (serial port of a Macintosh). You do not have to connect the DTR and DSR signals to the converter device.



Use a Null Modem cable (see AT-PC) to connect the RS422/RS232 converter to the Siemens PLC.

6. RK512 (3964R) Example VI

The RK512 (3964R) pack comes with a well-documented example VI. The example VI is located in the RK512 (3964R) library.

This example VI simply writes an array of extended precision numbers to the location marked by **data block** (DB) and **data word** (DW) in your PLC. Then the VI reads data from the same location and places them in an array of unsigned 16 bit numbers. This VI continuously keeps running until you stopped communication with the stop button.

You can use this example to experiment with the RK512 (3964R) VIs. Study this example VI carefully, and try to make it work with your PLC-PC configuration before you do anything else.

RK512(3964R) Transceive Example
RK512(3964R) protocol - version 5.1 August, 1999

port number (0): 0
data block: 10
data word: 10

stop communication: **STOP**

send array: 0
Array of 10 elements, each <0>

send error out: status: no error, code: 0, source: [empty]

read array: 0
Array of 10 elements, each <0>

read error out: status: no error, code: 0, source: [empty]

CIT Engineering NV
Centre for Industrial Technology
Kleinhoefstraat 6
2440 Geel Belgium
Tel. 3214/57.96.50
Fax 3214/57.96.51

Note :
This example VI simply writes an array of extended precision numbers to a certain location in your PLC then reads them out again at that same location and places them in an array of unsigned 16 bit numbers.

Data send to PLC
Try changing the representation of this control and check out the dataword that you received back from the PLC.

Data received from PLC
You should examine the number of DATA WORDS returned as a function of the data type that you have transmitted.

Remark : Do not forget to declare the used DBs in your PLC.