PPE & Modbus
(NT Version)

April, 99

Praxis Instruments, Inc Houston, TX 713.747.2820

PRAXIS Protocol Engine User's Guide

Document Release 1

May, 1998

2. Trademarks

Microsoft, Windows, and Excel are trademarks of Microsoft Corp. Intouch is a trademark of Wonderware Corp. IBM is a trademark of International Business Machines Corp.

PRAXIS

7545 South Freeway (Hwy. 288)

Houston, Texas 77021

Tel:(713) 747-2820

FAX:(713) 747-4502

EMAIL: ENG@NEOSOFT.COM

INTRODUCTION

The Purpose of This Manual

The idea behind the User's Guide is to provide the operator with a document that is easy to understand and follow, providing all of the information necessary to employ the PPEngine Server, from installation through several years of use.

The first section of the PPE User's Guide contains an overview of the system, including information about the Dynamic Data Exchange (DDE) architecture and PPEngine specifications.

The rest of the sections cover installation and configuration of the PPEngine, along with a diagnostic section, which should be referred to prior to contacting Praxis for technical support. This will better enable us to deal more efficiently with any problems, which might arise.

Several appendices follow the end of this manual. Appendix A is a list of reserved items which are already assigned to tasks by PPE, Appendix B provides all pertinent data for the chosen protocol(s) for your system, and Appendix C contains a glossary of terms that may be unfamiliar to the user.

Commands & Instructions

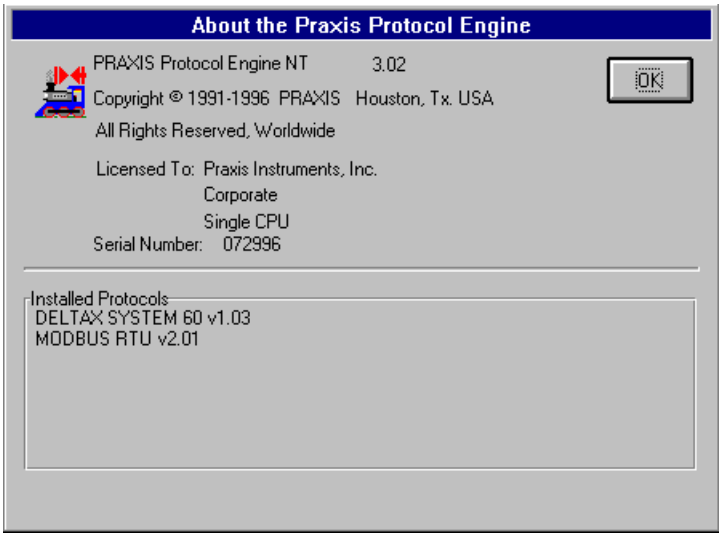
Because PPEngine is based in Microsoft Windows NT, most commands will be executed with the mouse. If the operator is new to Windows, it is advised that he should practice clicking, dragging, and double clicking with the mouse until comfortable with the techniques.

Pointing refers to the act of moving the mouse on a flat desk or other surface until the cursor is pointing to the desired object. Clicking means positioning the arrow cursor over an object, box, or button, and pressing on one of the mouse buttons one time. Highlighting refers to the technique of depressing the mouse button and while holding it down, dragging the mouse across an area until the desired word(s) or field is highlighted. Double clicking is lightly and quickly depressing the left button on the mouse twice. Double-clicks usually activate an application or program.

The quickest way to change a field is to highlight the contents of the box by dragging the mouse as described above or tabbing to the field. Immediately begin typing the new data while the word(s) is highlighted and the old data will be overwritten.

The keyboard may also be used in some instances. The <TAB> key is used to move around the fields within a dialog box. **NOTE** that pressing the <ENTER> key will take the operator back to the Main Menu, so do not use it until finished with a dialog box and ready to save any changes. Clicking on Cancel will return the user to the Main Menu with no values changed or added, as will pressing the Escape key. The space bar may be used to erase data in a field.

Refer to the Windows Tutorial for further information and guidance regarding use of a mouse in Windows applications.



To view this screen, double-click on the PPEngine icon to open the application, then click once on the File button from the Main PPEngine Menu. Select the About PPEngine option. The About screen provides the customer's name, serial number of the PPEngine, and which protocols

Table of Contents

1.0 AN OVERVIEW OF THE PRAXIS PROTOCOL ENGINE

WHAT IS THE PPE?

The Praxis Protocol Engine (PPE or PPEngine) is a Microsoft Windows-hosted server designed to provide multi-port, multi-protocol communications common to the industrial SCADA (Supervisory Control and Data Acquisition) market. The PPE accesses data from a variety of field devices, including radio, direct lines, microwave, and satellite, using different communication protocols through a single server. The data is presented to the user via Dynamic Data Exchange (DDE), a standard communication protocol developed by Microsoft for interapplication data sharing (Refer to Section 1.1 for further information regarding DDE). This means that it is no longer necessary to send someone out in the field to check on an RTU or retrieve needed data.

WHO USES THE PPE?

There are two levels of users for the Praxis Protocol Engine. Upon receipt of the PPEngine package, the communications technician or system engineer will install and configure the software for the system's unique protocol(s) and RTUs. Once PPE has been installed and configured, the operator will use it to monitor the RTU flow and gather requested information. Ideally, PPE is virtually invisible to the operator unless an error causes a visible signal, requiring action by the user, or until an equipment change requires the PPE to be reconfigured.

1.1 A WORD ABOUT DDE

Dynamic Data Exchange (DDE) is a communication protocol that works with PPEngine to allow an interchange of data called a "conversation". The conversation takes place between two programs, following the client/server model. An application that initiates a conversation and requests data is called a Client application. An application that provides data and accepts requests and commands is called a Server

application. The client/server model is a logical relationship that defines which program must be responsible for starting a conversation. Programs can be written to have both client and server functionality at the same time. Microsoft's Excel and Wonderware Software Development Corporation's Intouch are examples of application programs that can function simultaneously as clients and servers.

To transfer SCADA information via DDE, the client initiates a conversation with PPEngine by specifying three pieces of information: the *application name*, the *topic name,* and the *item name* within the topic.

The application name is the name of the DDE Server program from which data is being requested. For example, to request data from Microsoft Excel, the application name is "EXCEL"; to request data from Intouch, the application name is "VIEW"; to request data from the Praxis Protocol Engine, the application name is "PPENGINE".

A topic is an application-specific group of data elements. The topic name, when accessing data from Excel, is the name of the spreadsheet containing the data you

wish to view. The topic name when requesting data from Intouch is the predefined topic name "TAGNAME". Topic names for the PPE are user-defined and represent "logical" RTUs. They are logical because a defined topic may reference all data from a field device or just a subset of data. Multiple topics can also reference the same physical field device. The terms "topic" and "RTU" may be used interchangeably in the context of the PPE. They are limited to 32 characters.

The item name indicates a specific data element within a topic. Item names are application specific and follow some kind of naming convention. Item names for data from Excel are the cell names, e.g., "R1C1". Item names in Intouch "View" are the database tagnames defined in the database tagname dictionary. Item names in the PPEngine include some predefined names, e.g., Scan, Status, and a set of naming rules based on each supported protocol.

In the context of PPEngine, regardless of how protocol data is transmitted, it is converted into a format that is recognizable by a DDE application. Client applications may request data (read), advise data (inform of change), or poke data (write).

DDE may be easily extended across NetBIOS compatible networks and RS-232 serial links using third party products such as Wonderware's NetDDE, Windows for Workgroups NetDDE, and Imaginary Systems, Inc.'s RemoteDDE.

1.2 SPECIFICATIONS

System Requirements:

A standard 80486/586 based IBM-PC compatible personal computer with VGA or better monitor

66 MHz or better processor speed

System RAM (at least 4 MB, with 8 MB recommended)

Windows/NT 3.51 or later

Mouse

Dynamic Data Exchange (DDE) client program such as Intouch from Wonderware

Contents Of PPE Package:

Disk containing PPE program, including installation procedures

PPEngine User's Manual

Support:

Every effort has been made to supply useful and helpful information to assist the users in entering remote terminal unit data into a client system. Praxis assumes that the operator is familiar with Microsoft Windows/NT.

However, should problems arise which cannot be diagnosed and remedied using the trouble-shooting and diagnostics sections provided in this manual, Praxis offers technical support after the sale at (713) 747-2820.

2.0 INSTALLATION

Installation instructions are provided on the PPEngine diskette included in the system package. The README.TXT file contains all information needed for initial installation of the PPEngine. Use Notepad as an editor to read this instruction document.

3.0 HOW TO CONFIGURE A SYSTEM FOR THE PPE SERVER

The configuration of the Praxis Protocol Engine (PPEngine) is a two-step process. Before the PPEngine can perform as a Dynamic Data Exchange (DDE) communications server with a client application, certain data must be obtained and entered into the PPEngine. This is called configuration. Once configuration is complete on the PPEngine/server end, the client application, which is usually Wonderware's Intouch software, must also be configured before communications with remote devices can take place. No conversations may take place until configuration on both sides has been administered successfully.

3. THE CLIENT APPLICATION

PPEngine is the server in any exchange because it provides data, accepts requests and commands, and never initiates a conversation. A software application acting as client requests, advises and pokes data, and ALWAYS initiates the conversation. A request refers to a one-time request for specific data. Advising means that the RTU should let the client know when data changes and poking simply means to write data to the PPEngine. Contact with the PPEngine/server is always initiated using the *Application, Topic, Item* format used in all DDE architectures. Refer to your specific client application documentation for specific syntax for initiating a conversation.

The *application* in DDE conversations is the name of the program from which data is being requested. Using the Praxis Protocol Engine, the application name is always "PPEngine". The *topic* is an application-specific group of data elements. Under PPEngine, the topic name is the name of the remote terminal unit (RTU) to which the client wishes communication. The *item* name indicates a specific data element within the specific topic. Items are application-specific and follow a naming convention dictated by the application. In PPEngine, some item names are predefined. These are generic items commonly used by all protocols. Other item names, representing the unique identifier of a specific RTU point or register, are based on the supported protocol. Under the Modbus

protocol, for example, the item name is the register number. See Appendix A for a listing of the generic item names used by PPEngine and Appendix B for protocol-specific items.

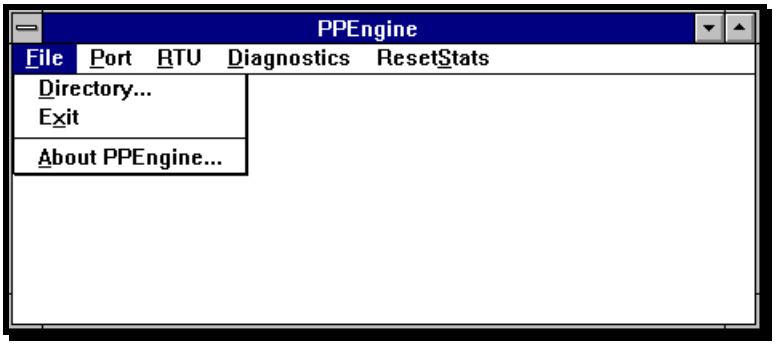
3.1 CONFIGURATION OF THE PRAXIS PROTOCOL ENGINE

3.1 ENGINE

The PPEngine is designed to be easily configured with clear and concise dialog boxes to lead the user through the process. Prior to beginning configuration, the operator should have the following information available, as it will be required during the configuration procedure:

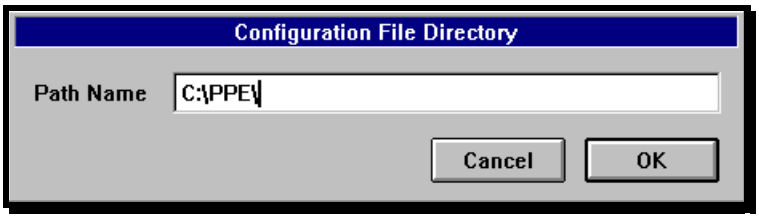
- The communications port number through which the PPE and the RTUs will be communicating
- The names and addresses of the RTUs. The name of the RTU usually has something to do with the location, while the address is usually the number of the RTU.
- The modem warmup and warmdown time. Sometimes this time is only arrived at after several trial and error experiments. Warmup and warmdown time will be more clearly explained in the following sections.
- Polling Interval. This entry represents the time, in milliseconds, that PPEngine waits before polling each RTU.
- Other protocol-specific data, as required.

Values for each port and each RTU to be monitored from that port will be requested. These values are entered through the appropriate dialog boxes as shown on the following pages. RTU dialog boxes for specific protocols are described in Appendix A of this manual. To begin the configuration process, open the Praxis Protocol Engine by positioning the mouse on the PPEngine icon and double clicking the left button on the mouse. When the PPEngine headlight is on, it is indicating that at least one conversation has been initiated from the client to the PPEngine. Once PPEngine has been opened, the Main Menu box will appear.



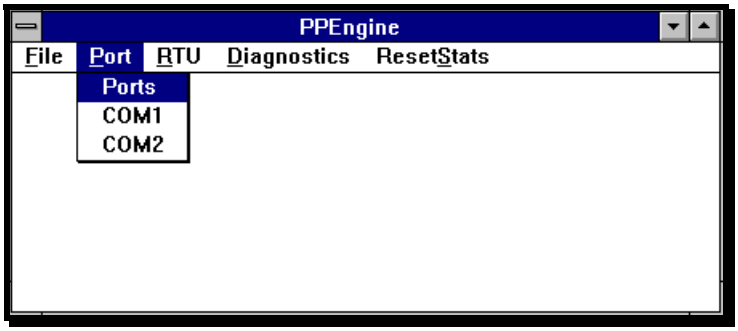
The first option on the Main Menu is File. Click once on the File option to view the pulldown menu. Under File are the Directory, Exit and About PPEngine choices. Choose Exit to exit the PPEngine system. The About screen was illustrated in the introduction.

Under Directory is the DOS file path name where PPEngine configuration files are stored. If this path is changed for any reason, the new path name will take effect the next time PPEngine is started.



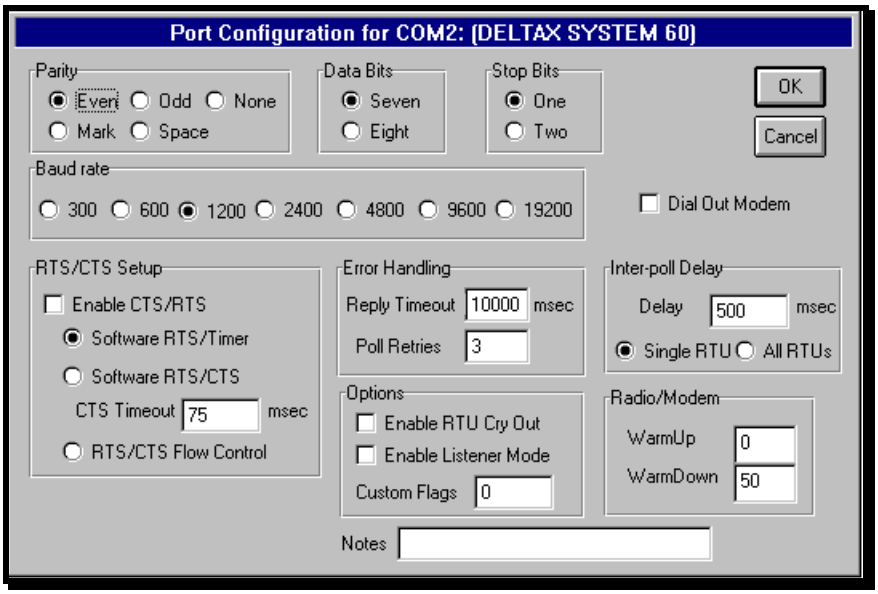
3.1.1 Port Configuration

To begin port configuration, click on the PORT option at the Main Menu.



All available comm ports will be listed. Click once on the port to which the RTU is connected. This will open the **Port Configuration** dialog box.

The **Port Configuration** dialog box provides defaults for most of the values.



However, these are not fixed. Following are the descriptions of each field, along with the Praxis default for comm ports:

- Parity - Set to odd
- Stop Bits - One
- Data Bits - Eight
- Baud Rate - 1200
- RTS/CTS Setup - This box allows the system to raise "Request To Send" and "Clear to Send" signals between the PPE and the communication device (RTU or radio). If the Enable RTS/CTS box is checked, handshaking will be enabled for communications.

  There are two ways to provide this "handshake". If Software RTS/Timer is chosen (the default), the request to send is raised for the duration of the warmup and warmdown times which will be entered in the Station Configuration dialog box. This is the recommended method of enabling RTS.

  Use of hardware RTS is not recommended for any Windows application because the Windows operating system does not handle this method very efficiently. It is not supported at all in the PPEngine. What happens is that a "request to send" signal is sent to the RTU via the connecting cable. It is then up to the modem at the RTU to return a "clear to send" signal, enabling the Master to then request the data it needs.

- Error Handling - Error handling will differ, depending on equipment. The Reply Timeout is the value that tells PPEngine how long to wait for an RTU to respond. In the default, 3000 msec represents three seconds. This time will vary greatly. If

using a satellite system, the reply timeout may have to be as much as 10,000 milliseconds.

- Poll Retries represent how many times PPEngine should retry a poll to an RTU if an invalid or no response is received. In the default, PPEngine will retry polling the RTU three times after no or an invalid response is received before continuing to the next RTU.

- Options - This box will only be useful if the system is operating through a front-end processor. Also, Praxis only supports these options in Modbus with Praxis RTUs.

    1. Enable RTU Cry Out is chosen if there are RTUs in the field which are seldom polled. If this box is checked for an RTU, it will 'cry out' when it has data to transmit. This lets the Master know that it should poll the RTU.

    2. Enable Listener Mode provides a means to use the PPEngine alongside an existing system if desired. When this box is checked, it is telling the FEP to only listen' to the polled messages coming from another Master with which it is sharing a line. PPE waits for the response and processes the data without it actually polling the RTU itself.

Both the Custom Flags and Notes fields are reserved for Praxis custom implementations and have no set definition at this time.

- Inter-Poll Delay - This figure represents the interval in milliseconds that PPEngine waits before polling an RTU.

  If setting is on "Single RTU only", the specified interval is only applied to multiple messages going to the same RTU. This gives the RTU time to reset itself after responding to messages. This setting is commonly used when an RTU requires multiple poll messages to gather all the data.

  If setting is on "All RTUs", PPEngine will delay the specified interval set before polling the next RTU.

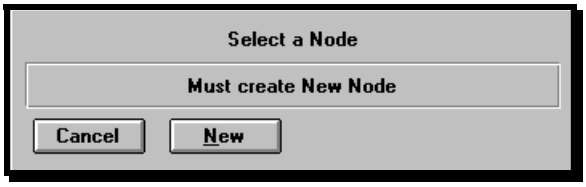
- MODEM/RADIO CONTROL - The times represented in these boxes tell the host PC how long to wait for modem warmup before a message is sent, and how long to wait for warmdown before repolling the RTU. This time in milliseconds will vary, depending on equipment. A satellite will have a much longer warmup time than a direct phone line, etc.

To revise existing values with a mouse, position the arrow on the box to be changed, click and drag the mouse across the old values until the box is highlighted. Type any changes and the new data will overwrite the old. Using the <TAB> key will also move the cursor within the dialog box. Once values have been entered in the appropriate field, do not press the <ENTER> key until ready to exit the dialog box. Use <TAB> should another field require entry or revision.

Clicking once on the OK button or pressing the <ENTER> key accepts the values just entered (along with existing defaults) and returns the operator to the Main Menu. Clicking on Cancel leaves all values unchanged and also returns the operator to the Main Menu.

3.1.2 RTU Configuration

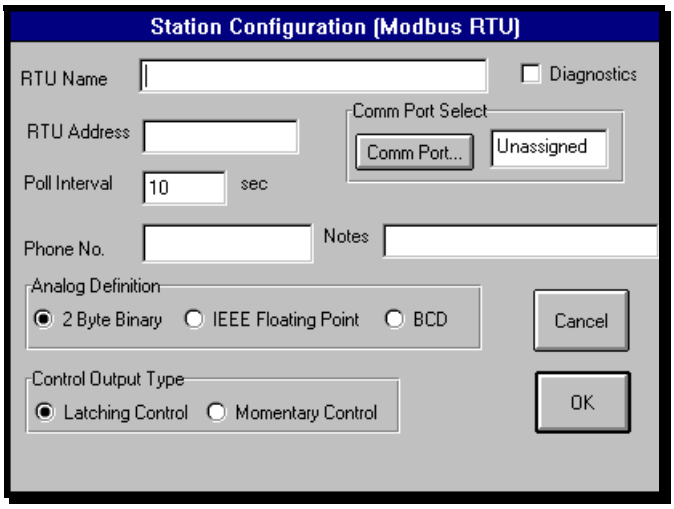
To configure an RTU, click once on the RTU function key from the Main Menu. The **Select an RTU** menu box opens.



If this is an initial setup, no RTU names will be displayed. Click once on New to open the **Select a Protocol** menu box.

The only protocols displayed in this menu will be the ones available for use in the system installed by the operator. The menu example shown here lists all protocols supported by Praxis Instruments. Click once on the selected protocol that applies. This opens the **Station Configuration** dialog box. The options in this box may vary,

depending upon which protocol is being used. The following screen and its field definitions correspond to the MODBUS protocol, which is most commonly used with PPEngine. Please refer to Appendix B for addition protocol descriptions for your unique system.



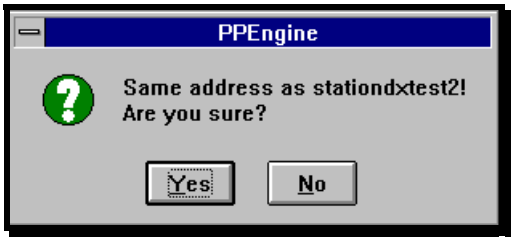
Use this dialog box to enter all new RTUs to the system. Type in the new RTU name (up to 32 characters in length), address, and other values as described in the following pages. Change the default settings, if necessary, by typing over the displayed information. Clicking on OK accepts values as entered and returns the operator to the Main Menu. The following information must be provided for each RTU:

- RTU NAME - Usually an alphanumeric name of one or more words pertaining to the physical location or other identifiable aspect of the RTU.

- RTU ADDRESS - The address is simply an arbitrary unique number referring to the RTU which can be used when requesting data.

- COMM PORT - Clicking on this box displays a list of communications ports available. The operator MUST select a port by positioning the cursor on the

appropriate choice and clicking once to highlight it, then clicking on the OK button. If an attempt is made to exit the Station Configuration dialog box without selecting a comm port, PPEngine displays the following message:



- POLL INTERVAL - This setting tells the PPEngine how often to poll an RTU. It only represents the minimum length of time PPEngine will wait before each polling. If it takes longer to poll all RTUs, PPEngine will simply poll as quickly as possible.

The remaining information required to configure an RTU is protocol-specific and must be referenced in Appendix B of this manual.

Once all information has been entered for the RTU, click once on the OK button to save the information. Should an RTU be entered with the same address as one previously submitted, PPEngine will display a verification message as a precaution against faulty data.

3.1.3 RTU Modification

Should it become necessary to modify an existing RTU, PPEngine is designed to make revision a simple task. From the Main Menu, click once on RTU. All entered RTUs will be displayed along with additional buttons for modification or deletion. The

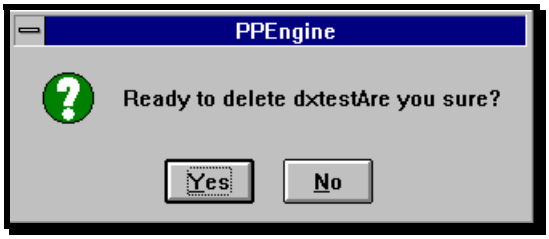
operator should highlight the RTU he wishes to modify and click once on the Modify button.

Clicking on Modify opens the Station Configuration dialog box. With the name of the RTU highlighted, begin typing the revised RTU name if necessary. Enter and/or modify other existing information until all changes are completed. Click on Cancel to return to the Main Menu or OK to accept modifications as entered.

3.1.4 RTU Deletion

Deleting an RTU is as simple as highlighting the RTU from the Select and RTU dialog box and clicking once on the Delete button. Before removing the RTU,

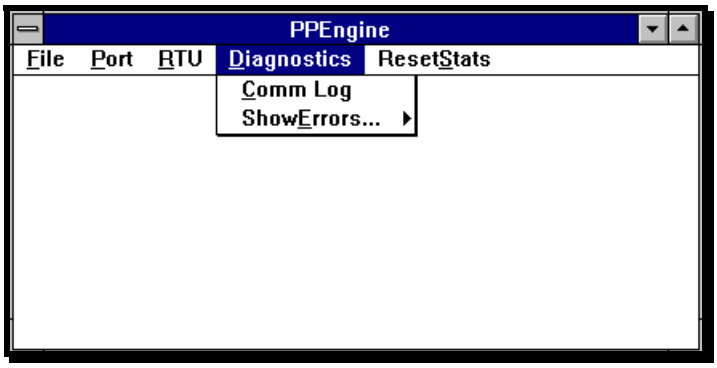
PPEngine will verify that the operator really intends to delete that device from the system. Click on Yes to continue deletion or No to return to the Main Menu with no deletion done.



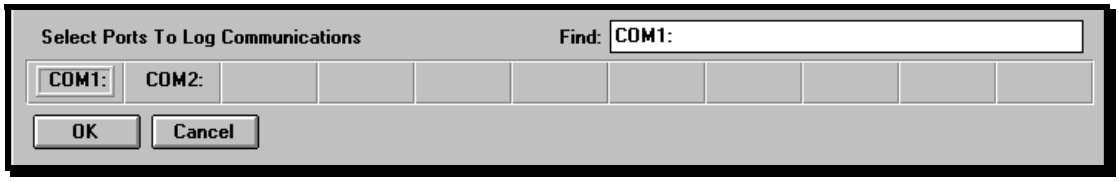
4.0 DIAGNOSTICS

Once installed and configured correctly, the PPEngine can be expected to run smoothly. However, if using WonderWare InTouch software with PPE, diagnostics are available to assist in commissioning the system, should the need arise. The Diagnostics option may also be used when a remote terminal unit is not communicating with the Master PC.

From the Main Menu, click once on Diagnostics to view the drop down menu. Two choices, Comm Log and ShowErrors are available for selection.

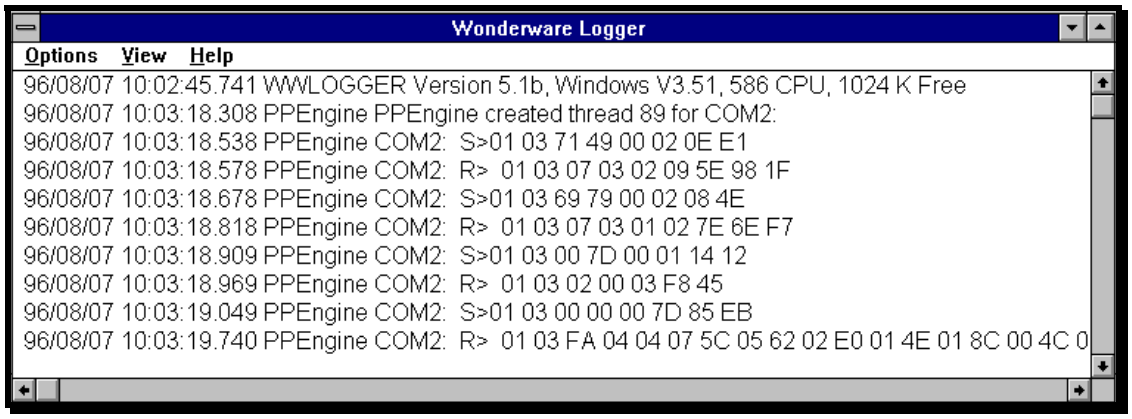


Clicking once on Comm Log displays a menu box with a choice of ports. By choosing one of the available port numbers, the operator is telling WonderWare that he wishes to monitor all transmitted and received data coming in through the RTU. The RTU data will be rerouted to the WonderWare Logger. Choose OK to proceed and open the logger window. Escape or Cancel will end the operation and return the operator to the Main Menu. Either restarting the PPE system or choosing the port again and pressing the OK button (it works as a toggle) will turn the Logger monitor off.

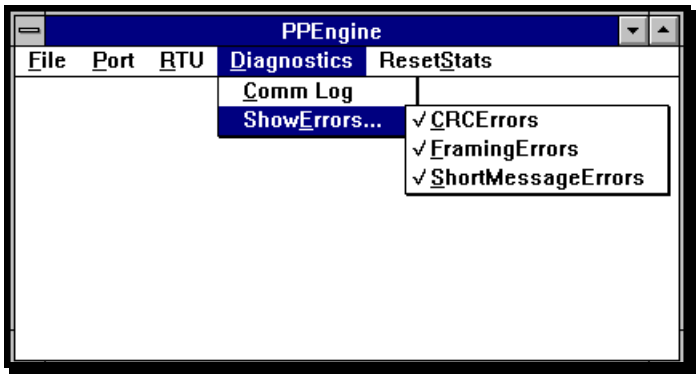


The WonderWare Logger screen will display

all send and receive messages for the selected port



The second choice under Diagnostics is ShowErrors. Clicking once on this option produces a new menu offering specific error choices to view.


PPEngine
File
Port
RTU
Diagnostics
ResetStats
Comm Log
ShowErrors...
√ CRCErrors
√ FramingErrors
√ ShortMessageErrors

Please note that these error names are somewhat generic. Refer to the appropriate protocol documentation for the correct error name for your system.

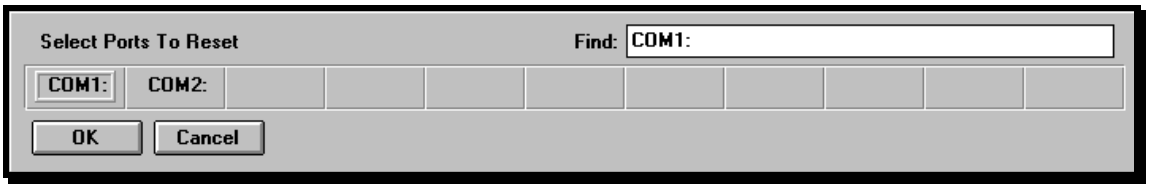
Click once on **CRCErrors** to see which RTUs are showing a cyclic redundancy check error (a mathematical method for verifying that data transmissions are error free).

Click once on **Framing Error** to see which RTUs are showing this error. A framing error occurs when data sent via synchronous communications deviates from the required sequence.

Click once on **ShortMessage Errors** to see which RTUs are giving incomplete messages.

4.1 RESETTING THE PORTS

After troubleshooting and resolving a problem, it is possible to reset all statistics for individual ports using the ResetStats option from the Main Menu.



After clicking on the ResetStats option, a menu will appear displaying available ports to reset. Choose a port, click on OK, and all statistics will be set back to zero for that port only.

ResetStats would usually be used after resolving a comm problem. All accumulative errors will be zeroed and the operator may start with a clean slate.

THE MODBUS PROTOCOL

3.2 Summary

This section to the Praxis Protocol Engine (PPEngine) User's Manual describes the Modbus protocol extension (only). Information about the PPEngine that is not specific to Modbus RTU may be found in the main documentation. This section is NOT intended to be a stand-alone document. Familiarity with the PPEngine documentation is a prerequisite for understanding this section.

3.3 Terms

The term "RTU" is used as a synonym for the Modbus field device which could be any type of device capable of communicating using the Modbus RTU protocol (PLC, flow computer, etc.). The RTU is also referred to as the DDE "topic".

3.3.1 *Protocol Specifics*

The PPEngine supports the Modbus RTU standard that transmits a synchronous binary data in an 8-bit, no-parity format. Note that the slower Modbus ASCII format is now supported.

The register naming convention uses different ranges of register addresses to imply different function codes for polling and output commands. These conventions are based on the Modicon PLC types 484 and 584/984. The following table summarizes the register ranges.

MODBUS TYPE	584/984 Range	**Data /Type**
Coil	1-9999	Read/Write Discrete
Status Input	10001-19999	Read Only Discrete
Analog Input Register	30001-39999	Read only Real/Integer
Analog Holding Register	40001-49999	Read/Write Real/Integer

Modbus Protocol Function Codes

To support the registers listed above, read and write requests for DDE items (i.e., registers) are translated into protocol messages using the following function codes:

3.3.1.1 ode	Meaning	Action
01	Read Coil Status	Obtains current status (ON/OFF) of a group of logic coils.
02	Read Input Status	Obtains current status (ON/OFF) of a group of discrete inputs.
03	Read Holding Registers	Obtain current binary value in one or more holding registers.
04	Read Input Registers	Obtain current binary value in one or more input registers.
15	Force Multiple Coils *	Forces a series of consecutive logic coils to defined ON or OFF states.
16	Preset Multiple Registers *	Places specific binary values into a series of holding registers.

** Note: The Microsoft Windows DDE standard provides for setting a single item (only) with a "POKE" command. It does not support setting of multiple values simultaneously. This means that only a single value is sent in a Modbus protocol message. However, the "multiple" function codes are used to set a single value rather than function code 5 (Force Single Coil) and code 6 (Preset Single Register). This was done to be compatible with a greater number of Modbus devices -- some do not support function codes 5 and 6.*

PPEngine handles advises in the best manner possible to increase the speed of transmissions. Once it receives the first advise for a register, it builds a message (poll request) to send to the RTU. Using this as a base, subsequent advises cause additional registers to be added to the poll request until the limit has been reached (125 for 2-byte binary registers and 62 for IEEE floating registers). Once that limit has been exceeded, another message is built specifying the number of registers to read. This way, one message does not have to be sent for each advised register. Please note that DDE **requests** are handled individually, requiring one message for each requested register.

The PPEngine provides an efficient method of packing coils or status inputs in holding registers (40001-49999) in conjunction with the Modbus protocol. This is useful if there are a limited number of coils or status inputs in an RTU or PLC and you do not want to waste time polling them in addition to the holding registers. Under normal circumstances, a coil or status inputs is polled for its values. If an

RTU or PLC is programmed to pack coils or status inputs in analog holding registers, the following is helpful.
To extract coils or status inputs packed in a holding register, the item name is the register number plus bit position separated by a colon. The bit position ranges from 0-15, with 0 being the least significant bit.

Example: 40001:0

This item will get the least significant bit out of holding register 40000.

To allow writing to coils which are returned packed in a holding register, PPE uses a ":W" to indicate that the coil is write only and will not be polled.

Example: 1:W

This item indicates that PPEngine will only write to this coil and will not poll it. It is understood that you are getting this coil as a packed register and has an item defined somewhere else.

The PPEngine Modbus protocol supports reading and writing of single precision IEEE (32-bit) floating point data with word/byte data order specifications from Modbus Binary type RTUs and PLCs (Gould Modbus 16 Bit Binary Integer Registers). The floating point number is packed into two adjacent 16-bit integer "analog" or "holding" registers. To indicate to the PPEngine that a given DDE item should be treated as a floating point, the item name will be a register number followed by colon and "f". For example, a DDE request for "40002:f" will indicate to the PPEngine to actually poll the Modbus device for registers 40001 and 40002, combine the two 16-bit integers into a 32-bit single precision floating point number, and return the floating point number to the DDE client program. Conversely, a DDE poke to "40002:f" with a floating point value will cause the PPEngine to split the floating point number into two 16-bit integers, and issue a single (combined) write command to the Modbus device for registers 40001 and 40002.

The word/byte data order option is included to support various models of RTUs and PLCs. This feature is implemented by modifying the DDE item name ":f" suffix with an additional word/byte data order digit. The format for reading and writing single precision IEEE (32-bit) floating point data is as follows:

Itemname:f1 Most significant word (sign, exponent, and most significant bits of the fraction) of the floating point number is in the first Modbus register and the byte order in the data words is unaltered. ***Note that this matches the data storage method of TI PLC. See the SIMATIC TI505 Programming Reference, Section 2.3, Data Representation of Real Numbers, page 2-5.***

Itemname:f2 Most significant word of floating point number is in the first Modbus register and the byte order in the data words is swapped.

Itemname:f3 Least significant word of floating point number is in the first Modbus register and the byte order in the data words is normal.

Itemname:f4 Least significant word of floating point number is in the first Modbus register and the byte order in the data words is swapped.

Note that if the word/byte order digit is not specified, it will be assumed to be 1.
Also, the PPEngine Modbus protocol supports reading and writing of 32-bit signed long integers with word/byte data order specifications from Modbus Binary type RTUs and PLCs (Gould Modbus 16 Bit Binary Integer Registers). A signed long integer is packed into two adjacent 16-bit integer "analog" or "holding" registers. To indicate to the PPEngine that a given DDE item should be treated as signed long integer, the item name will be a register number following by colon and "l". For example, a DDE request for "40002:l" will indicate to the PPEngine to actually poll the Modbus device for registers 40001 and 40002, combine the two 16-bit integers into a signed long integer, and return the signed long integer to the DDE client program. Conversely, a DDE poke to "40002:l" with a signed long integer value will cause the PPEngine to split the signed long integer value into two 16-bit integers, and issue a single (combined) write command to the Modbus device for registers 40001 and 40002.

The word/byte data order option is included to support various models of RTUs and PLCs. This feature is implemented by modifying the DDE item name ":l" suffix with an additional word/byte data order digit. The format for reading and writing 32-bit signed long integers is as follows:

Itemname:l1 Most significant word of signed long integer is in the first Modbus register and the byte order in the data words is unaltered.

Itemname:l2 Most significant word of signed long integer is in the first Modbus register and the byte order in the data words is swapped.

Itemname:l3 Least significant word of signed long integer is in the first Modbus register and the byte order in the data words is normal.

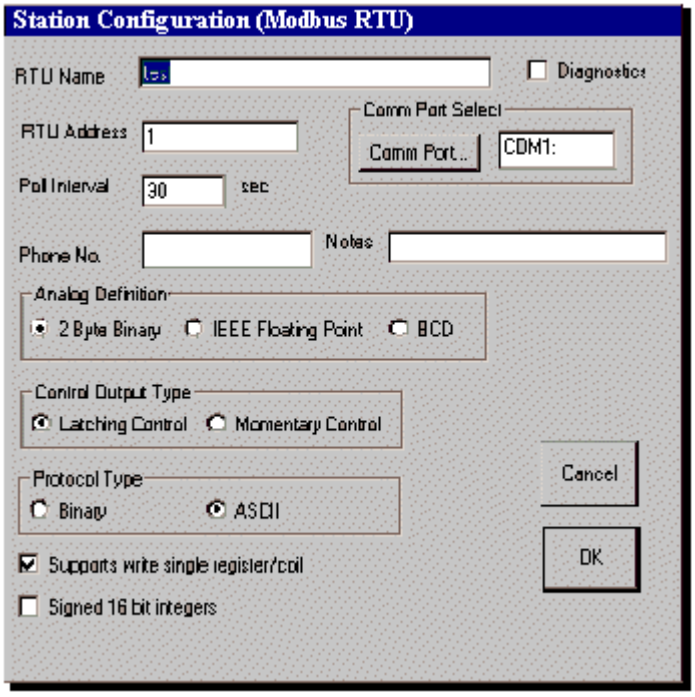
Itemname:l4: Least significant word of signed long integer is in the first Modbus register and the byte order in the data words is swapped.

Note that if the word/byte order digit is not specified, it will be assumed to be 1.

3.4 RTU Configuration

Before the PPEngine can be used to acquire data from Modbus RTUs, configuration information describing the communication channel and RTUs must be entered into the PPEngine.

You must define a configuration for each RTU to be polled. An arbitrary name must be assigned to each device that will uniquely identify it as a DDE topic. To define the topic, select the RTU option from the menu bar (refer to Section 3.1.2). The "Select an RTU" dialog box will appear. To create a new RTU, click on New, to modify an existing RTU, select Modify. The "Station Configuration (Modbus RTU)" dialog box will now appear.



The following describes each field in the dialog box:

RTU Name

This field is an arbitrary used-defined name that will be used as the DDE topic name by DDE client applications requesting data from this RTU. The field allows names of up to 32 characters.

4. *RTU Address*

This field is the MODBUS polling address and must be in the range 1-254. This field is in decimal numeric format.

5. Comm Port Select

Click on the Comm Port button to pop up a select box to choose which serial communication port this RTU will be polled through. The currently selected Comm port is displayed in the box to the right of the Comm Port button.

6. *Poll Interval*

The poll interval is the amount of time, in seconds, between successive polls of this MODBUS device. The normal, common sense rules apply about setting polling intervals: don't set the polling interval for ten RTUs on the same port to one second; it can't be done.

Notes

Up to 60 characters of optional notes may be entered. This is commonly used to provide additional descriptive information about the Modbus device's characteristics, location, etc.

Analog Definition

This selection box is for selecting the format of the analog input and holding register data for this RTU:

2 Byte Binary - This option specifies that values are returned in a two-byte binary format, with the high order byte first, followed by the low order byte.

IEEE Floating Point - This option specifies that values are returned in four-byte IEEE single precision floating point format.

BCD - This option specifies that values are returned in two-byte Binary Coded Decimal format.

Control Output Type

This specifies the behavior of control outputs within the MODBUS device.

Latching Control - This option specifies that the control outputs sent to the RTU will latch to the 0 or 1 value in the protocol message.

Momentary Control - This option specifies that a control output of 1 sent to the RTU will momentarily pulse to a value of 1, then reset to 0.

Protocol Type

This selection box is for selecting the protocol type for this RTU (ASCII or Binary).

Supports Single Register Write

This check box indicates if this RTU supports single coil/register writes (function codes 5 and if checked, they will be used for single coil/register writes. If not, the write multiple coil/register function codes (15 and 16) will be used for all writes.

7. Signed 16 bit integers

If checked, 16 bit registers will be returned as signed integers, else unsigned.

APPENDIX A

PPENGINE GENERIC ITEM NAMES

This appendix provides the generic *Item Names* that are used throughout the PPEngine system, regardless of protocol. Other items, which are considered protocol-specific may be found in Appendix B, Protocol Specific Data. The items in this appendix may only be used for the tasks listed. The first list contains status items. All are available under each RTU topic.

DDE ITEM NAME	DESCRIPTION
Protocolname	Protocolname shows the name and version of the protocol in use. This is a read only item.
Scan	This item controls scanning or polling of an RTU. When Scan is set to one, all polls with the RTU are fully enabled. This is the default condition. When Scan is set to zero, time-based polling of the RTU is disabled until Scan is reset to one. However, a DDE poke to an item or a Demand Scan will be processed normally and sent to the RTU. Scan is a read/write/discrete item type.
Online	This item controls communication to an RTU. When Online is set to zero, communications are disabled until Online is set to one. The default is one (enabled).
Status	The Status item reflects the state of communications with an RTU. When communications are successful, the status is zero. If an RTU fails to respond after the specified number of retry attempts, the status item is set to one. Status is a read only/discrete item, meaning that it is going to be either zero or one.
Priority Scan	This item controls the scanning priority of the RTU. When set to zero, scanning is done at normal priority by default. When set to one, scanning is done at the demand scan priority.
Lastpolltime	This item contains the date and read only message time of the last successful communication with an RTU. The item is formatted as YYY/MM/DD HH::MM:SS'. It is read only.

Lastpollattempt	This item contains the date and read only message time of the last successful communication attempted with an RTU. The item is formatted as YYY/MM/DD HH::MM:SS'. It is read only.
Demandscan	Demandscan is a command to poll an RTU now (for a one time poll). It is a read/write/discrete item type.
PPE_Heartbeat	Heartbeat contains the "heartbeat" of the specific RTU topic. It is incremented by one each minute. Normally, a client application will use this item to monitor the health of the topic and/or the PPEngine, or associated network/serial links to the client. If the heartbeat value stops increasing, some combination of the client, PPEngine, and/or remote link has failed.
RTUcryout	Cryout is used to notify the client application of an unsolicited "cry out" message sent by a specific RTU. Handling of RTUcryout is protocol specific; however, the format of the item is generic up to a point: MM/DD HH:MM:SS PROTOCOLNAME data1 data2... dataN MM/DD is the month and day, HH:MM:SS represent the time in hours, minutes, and seconds, and PROTOCOLNAME is the protocol of the RTU. Data1 through dataN are protocol-specific data items. For examples, refer to the protocol appendix found in this manual.

The items in the following list contain communications statistical data. When using the port name as the topic, only these item names may be employed. However, a similar restriction does not apply to the item names (i.e., they may be used with any topic name). Each of the items produces the requested information for the current day, current hour, previous day, or previous hour. The correct syntax for the item name is "*Itemname curhour", curday, prevhour,* or *prevday.* An example is shown for the first item name.

DDE ITEM NAME	**DESCRIPTION**
Totaltries curhour	The Totaltries item provides a count for the total number
Totaltries prevhour	of times a poll of an RTU was attempted.
Totaltires prevday	
Totaltries curday	
Totalgood ----	The Totalgood item provides a count for the total number of successful polls of an RTU.
Totalbad ----	The Totalbad item provides a count for the total number of errors of all types.
Thruput ----	The Thruput item shows the percentage of successful communications during the current hour, previous hour, current day, and previous day. The percentage is derived by dividing Totalgood by Totaltries and multiplying by 100.
Noreply ----	The Noreply item provides a count for the number of attempts made to communicate with an RTU that resulted in an error.
Shorterr ----	The Shorterr item provides a count for incomplete messages. It shows the number of attempts made to communicate with an RTU that resulted in an incomplete message received for the current hour, the previous hour, the current day, and previous day.
CRCerr ----	The CRCerr item provides a count of data packet error detections. This may be CRC16 or checksum, depending on protocol being used.
Frameerr ----	Frameerr gives the count that shows the number of attempts made to communicate with an RTU that resulted in framing errors.
Echoerr ----	The Echoerr item provides a count that shows the number of attempts made to communicate with an RTU with the wrong RTU responding.

The following list contains system items supported by the *System Topic*. The SYSTEM topic is a special reserved topic name supported by all "well-behaved" DDE server applications. The items contain global data related to PPEngine specifically.

SYSTEM ITEM NAME	DESCRIPTION
Sysitems	Sysitems contains a list of the items supported by the SYSTEM topic. It is a read only item.
Formats	The Formats item contains the data format available from the PPEngine. Specifically, it is "TEXT" only.
Topics	The Topics item contains a list of the current topics available in the PPEngine. This includes the SYSTEM topic, all user-defined RTU topics and the list of communication port names. It is a read only item.
PPE_Heartbeat	PPE-Heartbeat contains the "heartbeat" of the PPEngine. It is incremented by one each minute. Normally, a client application will use this item to monitor the health of the PPEngine or associated network/serial links to the client. If the heartbeat value stops increasing, some combination of the client, PPEngine, and/or remote link has failed.
RTUcryout	RTUcryout is used to notify the client application of an unsolicited "cry out" message sent by an RTU. Handling of RTUcryout is protocol specific; however, the format of the item is generic up to a point:

RTUNAME MM/DD HH:MM:SS PROTOCOLNAME data1 data2... dataN

The RTUNAME is the name entered when configuring the RTU in the PPEngine. MM/DD are the month and day, HH:MM:SS represent the time in hours, minutes, and seconds, and PROTOCOLNAME is the protocol of the RTU that issued the cryout message. Data1 through dataN are protocol-specific data items. For examples, refer to the protocol appendix found in this manual.

APPENDIX C

GLOSSARY OF TERMS

TERM	DEFINITION
Abort	Cancel operation in progress.
Application Name	The second component of a conversation initiated by a client to a server. The name of the server application (PPENGINE).
Application Software	A program designed to perform a specific job using data input by the operator or from another program, e.g., spreadsheets.
Baud Rate	A measurement for electronic signals that tracks the number of signal changes occurring between modems in one second.
Button	A three-dimensional "button" is used in some Windows applications to represent options in a dialog box. A button is also the part of the mouse used to click on menu selections.
Click (Double-Click)	Lightly pressing the button on a mouse to choose a menu itemor pressing a Windows "button". A double-click is two rapid clicks and is generally used to execute a command.
Client Application	The first component in a protocol which initiates conversations with a server. Usually residing on the Master PC.
Configuration	Setting all parameters and entering all data required to run a system, application, or program.
Conversation	An interchange of data between a client application and a server application.
CRC	Cyclic redundancy check
CTS	"Clear to Send" a message or request for data.
Data Bits	A group of bits that represent a single character of data to be transmitted.

DDE	Dynamic Data Exchange - Microsoft utility that provides the ability to transfer data among Windows application.
Dialog Box	Screen image that asks for data to be entered or choices to be made. Sometimes commands may be issued from a dialog box.

TERM	**DEFINITION**
DigiBoard	Hardware component used in Praxis' Front-End Processor (FEP).
Discrete	Discrete defines a number as either zero or one.
DOS	Disk Operating System
Drag	A mouse technique in which the left button on the mouse is depressed while physically moving or "dragging" the mouse over the word or sentence being selected.
Drop-Down Menu	A selection of choices that "drops down" when a topic in the menu bar is opened.
Framing Error	In synchronous communications, a frame is a package of information transmitted as a single unit. A framing error occurs when something within the frame or package is garbled.
Front-End Processor	The DigiBoard product coupled with Praxis' proprietary (FEP) operating system is known as the front-end processor. It facilitates the transmission of data to and from the PPEngine.
Item Name	The third part of a conversation between the client and server applications. It represents specific data related to the topic, such as a register or coil number.
Interleaving	A command to PPEngine via DDE to poll one or more RTUs every other time, i.e., an example of a polling sequence is 1-45; 2-45; 3-45; 6-45; 7-45; 8-45; 9-45.
InTouch	A WonderWare software program used to create IBM personal computer-based (or compatible equipment) man-machine interfaces. InTouch runs under Microsoft Windows Version 3.0.

	It establishes the communication links that enable commands to be executed via the keyboard, mouse or a touch screen.
Microsoft Excel	A spreadsheet program (application).
Modem	An abbreviation for modulator/demodulator. It is a device that changes computer (digital) signals so they can be transmitted via telephone (analog).
Mouse	Hand held pointing device.

TERM	**DEFINITION**
Network	A group of computers and associated devices connected by communications links.
Node	A junction point capable of communicating with other devices. In the PPEngine User's Manual, a node is synonymous with remote terminal unit (RTU).
Parity	A way to check the transmitted data to verify that it has been received accurately.
Port	A location to which devices are connected which enables data to be transmitted in and out.
PPE	Praxis Protocol Engine, the server application which provides data on request to the Master PC.
PLC	Programmable logic controller.
Preemptive (Nonpreemptive)	An operating system which allows programs to receive a regular and relatively predictable share of processor time. A Nonpreemptive operating system means that each program is responsible for performing its task and voluntarily giving up the system to let another program have processor time.
Protocol	Required way data is requested from Praxis Protocol Engine, i.e., data requests follow the order: *Application, Topic, Item.*
Real Number	A real number is also called a floating point number. It is a number with a fixed point such as a decimal. Both 2.1 and .99999 are real numbers.

RTS	"Request to Send" messages are sent to the modem from the Master PC to let it know that it wishes to make a data transmission of some sort.
RTU	Remote Terminal Unit
Reserved Items	Reserved items are RTU communications-related data, supported by PPEngine, that are not specific to a protocol. These items cannot be used as the conversation components of a DDE operation.

TERM	DEFINITION
Scan	In computer and television technology, scanning means to move an electron beam one line at a time so that it lights the phosphor that creates a displayed image.
Server Application	The component or program that provides information when requested by the client application, i.e., PPEngine is a server application.
Short Message Error	A message containing fewer bits than a program specifies.
Start Bit	A space or even binary number that alerts the receiver that a character is coming.
Station	The term station is synonymous with RTU.
Stop Bit	In a synchronous transmission, a stop bit signals the end of a character.
Topic Name	The topic is the second component of a conversation between the client and server applications. In PPEngine the topic represents the logical name of an RTU.
Transport Layers	Transport layers are the mode by which a signal travels for a data interchange. Radio, lease lines, and satellites are transport layers.
VGA	Video Graphics Array. VGA enables color graphics to display on the monitor screen.
Warmdown	The length of time required for the host PC to wait before repolling an RTU.
Warmup	The length of time required for the host PC to wait before a message can be sent to an RTU.
Windows	The Microsoft operating system that lets users view information graphically. *It also allows multiple applications to run simultaneously.