

MAXDDE

User Manual for Version 1.1

**New Micros Inc.
1601 Chalk Hill Road
Dallas, Texas**

Written by

**Matthew Mercaldo
41 South Park Street
Oconomowoc, Wisconsin**

Liscense Agreement

Limits of Liability

Whereas Matthew Mercaldo and New Micros Inc. have to best of their ability tested MAXDDE and this document, Matthew Mercaldo and New Micros Inc. make no warranty of any kind, expressed or implied, with regards to the contents of this manual on MAXDDE and its use.

Trademarks

EXCEL, Visual Basic, and Windows are all trademarks of Microsoft Corporation.

Quattro Pro is a trademark of Borland International.

Table of Contents

Table of Contents.....	3
Quick Start	5
Requirements.....	5
Quick Installation.....	5
MAXDDE An Overview.....	6
The DDE of MAXDDE.....	6
MAXDDE on startup	7
Node Initialization Files.....	7
Item Initialization Files.....	7
Item Words.....	8
Items as Defined in MAXDDE.INI	9
Debugging information.....	10
Editor	10
The COM Port.....	10
BAUD Rate	10
Network Reconfiguration Time.....	10
MAXDDE's Modes of Operation	11
Polled Mode	11
Node Initialization.....	11
Item Initialization.....	12
Polling	12
Reconfiguration.....	12
Terminal Mode	13
Open File.....	13
Send File	13
Copy Buffer	14
Clear Buffer	15
DDE Link Status Messages	15
MAXDDE File Menu Items.....	16
File->Open File	16
File->Send File.....	17
File->Restart Polling	17
File->Stop Polling	18
File->Exit	18
MAXDDE Edit Menu Items	19
Edit->Copy Links.....	19
Edit->Copy Buffer.....	20
Edit->Clear Buffer.....	20
MAXDDE Help Menu Items	21
Help->About.....	21
Help->Help On Help	21
Help->Help.....	21
Trouble Shooting	23

Is the Network and wiring Correct?	23
Is the MAXDDE.INI file Properly Configured?.....	23
Is the PC Hardware Properly Configured?.....	23
MAXDDE and Sample Applications	24
Appendix A. MAXDDE.INI	25
Appendix B. MAXDDE and EXCEL	27
Appendix C. MAXDDE and Quattro	28
Appendix D. MAXDDE and Visual Basic.....	29
Appendix E. MAXDDE and EASYA.FTH	30
Appendix F. EASYA.FTH	31

Quick Start

Requirements

This section assumes that you have a working knowledge of DDE applications, the EASY-A protocol and have a network up and running with the appropriate support words and Item words tested and running.

You will need to setup the MAXDDE.INI file as described in section named **MAXDDE.INI**. It is important that the INI file is setup correctly or the EASY-A network definition will be incorrect resulting in network reconfigurations.

Quick Installation

Insert the disk into the appropriate 3 1/2 floppy drive (A: or B:). From the Windows File Manager's File menu item choose Run. At the Command Line prompt type **A:setup** if A: is the source drive OR **B:setup** if B: is the source drive. Respond to the questions as appropriate to your system or preference.

Once MAXDDE is installed, you will need to set the COM port properly. The default setting is COM2 as set in MAXDDE.INI. If you need to change this then edit MAXDDE.INI; the [COMPort] key item. Values allowed are 1,2,3 or 4. You can also set the baud rate by using the [BAUDRate] key item. Values allowed are 4800, 9600, 14400, or 19200.

If you run into problems, use the **Trouble Shooting** section of this manual to help resolve the problems OR read on.

MAXDDE An Overview

MAXDDE provides a way for DDE capable Windows applications to interact with an EASY-A network of Max-Forth nodes. EASY-A is a simple network protocol that can be programmed into the EEPROM of a Max-Forth capable 68HC11. Along with EASY-A, the node can be programmed with other application specific capabilities.

The DDE of MAXDDE

The Dynamic Data Exchange (**DDE**) is a mechanism that the Windows operating system uses to allow Windows applications to communicate with each other. Data to be passed between applications is identified by a three tiered name. The **Application name** is the first tier, the **Topic name** is the second tier, and the **Item name** is the third tier. All DDE capable applications let users link to server applications by this addressing scheme. MAXDDE is a DDE Server; it is the server for information that can be gotten from the configured EASY-A / Max-Forth network. Applications such as Microsoft EXCEL and Quattro Pro can be configured as DDE clients. Microsoft Visual Basic programs can be written quickly to provide a Man Machine interface to the application.

The information passed between applications will be of a specific type. MAXDDE will work with the TEXT type of information.

Some DDE clients make linking easy while others make it a little harder. In EXCEL a simple formula in a cell is all that is required to connect to MAXDDE. This formula follows the syntax:

```
=Application name|Topic name!Item name
```

Quattro Pro requires the use of its macro language to connect properly. Other applications just ask for the **Application|Topic|Item** triad and are straightforward to use through a system of questions asked from dialog boxes.

When configuring a client application to talk with MAXDDE, you can be assured that if the application talks DDE, somewhere it will let you set the Application name, the Topic name and the Item name. It is just a matter of finding out where that is in the context of the client Application.

MAXDDE on startup

Upon powerup or reconfiguration, MAXDDE will send each node a *Node initialization file* as well as *Item initialization files*. These files are downloaded and either reset the node or program the node's available internal RAM with *Item words*. These Max-Forth words respond to MAXDDE's subsequent requests and commands.

Node Initialization Files

These are files that setup the nodes to receive the Item Words. They typically include the word **COLD** to reset the Max-Forth internal dictionary pointer. The initialization files must follow the following rules.

1. Always begin the file with a carriage return.
2. Always end the file with a carriage return.
3. The file name must always end in **.FTH**.
4. The file name must always be the same as the Topic name that it represents. Nodes are represented by the Topic name. There must exist a file with the node name. For example, if you wish to use node A on the EASY-A network, then there must exist the file **NODEA.FTH** in the directory local to MAXDDE.

Item Initialization Files

Item initialization files are simple Max-Forth source files. By following a few simple rules the files will work properly. Of course the words to be sent to the system must be tested first, but assuming working definitions, the following rules must be followed.

1. Always begin the file with a carriage return.
2. Always end the file with a carriage return.
3. Always leave the BASE in a form that will work properly with the client application's expectations. This means leaving the base in DECIMAL if the client application expects to see only decimal numbers.
4. The file name must always end in **.FTH**.
5. The file name must always be named the same as the Item that it represents. If the Item that this file represents is **AD_1**, then the file must be named **AD_1.FTH**. The file should also contain a word definition called **AD_1**, but is not necessary if the kernel already supports that name.

Item Words

Item Words are Max-Forth words that respond to DDE items. They are divided into two types of words: **[itemsReceived]**, that is items that MAXDDE expects a numeric response from, and **[itemsSent]**, or items that MAXDDE expects to Poke or Set to a new value.

All **[itemsReceived]** words must end with a display value response.

```
: V_1 ( -- ) B000 C@ . ;
```

Word **V_1** will read the contents of address **B000** and display the value. Because this is an item to be received by MAXDDE, MAXDDE expects to receive the following text string:

```
V_1 nnn OK<cr><lf>
```

where **nnn** is a number or some text. (Remember that Max-Forth echoes the characters that it receives while interpreting.) The response between the word (**V_1** in our example) and the **OK** can be up to 32 characters in length. The response value does not have to be a number as shown in the above **V_1** example.

All **[itemsSent]** must expect a value and simply return.

```
: V_2 ( n -- ) B000 C! ;
```

When **V_2** is run it will set a number into address **B000**. Because this is an item to be sent by MAXDDE, MAXDDE transmits the following string:

```
nnn V_2<cr>
```

where **nnn** is a number, and receives back

```
nnn V_2 OK<cr><lf>
```

MAXDDE uses the correct echoing back of commands and the presence of **OK** at the end of the received message to validate the packet.

Items as Defined in MAXDDE.INI

Each item is further defined in MAXDDE.INI. Each item must have a keyword section in the file. Let us assume the following in MAXDDE.INI:

```
[itemsRecieved]
```

```
item1=AD_1
```

```
[AD_1]
```

```
scan=500
```

```
node=A,B
```

An item called **AD_1** is defined as an item that receives data from the EASY-A network. The naming of the keys, **item1** in our example above, is arbitrary. It is necessary though that a key be used followed by an **=** sign for proper operation. (**item1=AD_1**)

AD_1 has its own keyword section called **[AD_1]**. Each item will have two entries: **scan** and **node**. The **scan** entry tells in milliseconds when the item is to be polled. In our example every **AD_1** on the EASY-A network will be polled every 500 milliseconds. *The scan resolution is 100 milliseconds.* You must be careful when scheduling these items. Assuming 1 millisecond per character at 9600 baud. You can run four, four letter definitions in the given 100 millisecond time frame, before overruns begin to occur.

It is important to keep the words as short as possible for timing considerations. Since MAXDDE currently operates at 9600 baud, each character takes about 1 millisecond to transmit. Each character is echoed, taking another millisecond. So really each character takes 2 milliseconds. With a 4 letter definition, we can assume about 8 milliseconds to transmit and receive. For each word, an OK a Carriage Return and a linefeed will come across the wire, so we are up to 12 milliseconds per word. For each word we will assume that it returns a three digit number. Each number is preceded by a space and followed by a space. This totals up to 17 milliseconds per word. Assuming a worst case of 5 milliseconds for Max-Forth to think about what it has to do, and MAXDDE to send and receive the packet, we can assume 22 milliseconds per word. That is about 88 milliseconds for 4 words. Windows also takes some short time to respond to events, so this can throw some more time into the formula.

All in all running on a 486SX PC, Over extended periods of time, Four, four letter definitions run properly.

The **node** entry tells MAXDDE which nodes this item belongs to. In our example NODEA and NODEB will be maintained with **AD_1**. The node list is a comma delineated list of nodes in the set of upper case alphabetical characters [A..Z].

Debugging information

The two debugging statements tell MAXDDE to display file downloads and or polling messages. If both statements are set to OFF, then MAXDDE will startup minimized. If either or both are ON, MAXDDE will come up windowed.

Editor

The editor section of MAXDDE.INI specifies the editor that will be used by the **File->Open File** menu item. It is defaulted to the notepad program.

The COM Port

The COM port is setup by the [COMPort] item found in MAXDDE.INI. If no value is found in the item, the COM Port defaults to COM2. Ports are set by the numbers 1, 2, 3 or 4.

BAUD Rate

The baud rate is set in the [BAUDRate] item. Set Speed to one of 4800, 9600, 14400, or 19200. Baud rate defaults to 9600 if an illegal value is found in MAXDDE.INI.

Network Reconfiguration Time

Network reconfiguration time is the time between network health evaluations. This value in seconds tells when the network will be evaluate for lost EASY-A nodes on the network. The default value is 10 seconds for the key [ReconfigurationTime]

MAXDDE's Modes of Operation

MAXDDE supports two modes of operation. These are the *Polled mode* and the *Terminal mode* of operation. When MAXDDE first starts running it will attempt to connect to EASY-A based on the MAXDDE.INI file's configuration.

The mode of operation that handles the connecting, initializing and polling is the Polled mode of operation. When you select the **File->Stop Polling** menu item, MAXDDE switches to its Terminal mode of operation. From this mode of operation, you can interactively type to any node on the network, as well as download source files to any node for testing.

Polled Mode

The Polling mode is made up of three main states.

- ☐ Node Initializations
- ☐ Item Initializations
- ☐ Polling

When MAXDDE first enters the Polled mode of operation, it attempts to connect to the configured COM port. If MAXDDE cannot connect to the COM port then you will see a dialog box describing the situation. MAXDDE will continue to attempt to connect to the port.

Debug Note:

If the COM port is incorrectly configured then you must set the correct value in MAXDDE.INI and restart MAXDDE.

Node Initialization

MAXDDE will attempt to download the first item's item file in its items list. If upon inspection of its database, MAXDDE sees that the Node supporting that item is not initialized, it will attempt to send the node's Node initialization file. This file will be named NODEx.FTH where x is a letter between A and Z or between 0 and 9. If it cannot find that file MAXDDE will let you know that the file cannot be found.

Debug Note:

*Even if the contents of the file is a few Carriage returns, there **MUST** be a node initialization file for the node.*

Item Initialization

Once the node file has been set and the database updated, the item files are downloaded to their respective, initialized, nodes. These files are named as described above in the section named **Item Initialization Files**.

Debug Note:

*Even if the contents of the file are a few Carriage returns, there **MUST** be an item initialization file for the corresponding item.*

Polling

Once the initialization files are sent, MAXDDE starts to poll items based on the MAXDDE.INI file's configuration. Poll scheduling happens every 100 milliseconds. Every 100 milliseconds, MAXDDE first checks to see that a scheduling overrun has not happened. This overrun is seen as an item taking more than 1 second to complete its task response. If an overrun is detected then MAXDDE sets the active node associated with the stuck item's status to NOT BOOTED, then sets all items associated with that node to NOT BOOTED, then leaves the scheduling mechanism.

If there are no overruns, All items are evaluated for polling and scheduled, then the polling chain is started: All items scheduled for polling are polled. This includes any items that are going to send commands or values (from DDE POKE commands) to a node.

Reconfiguration

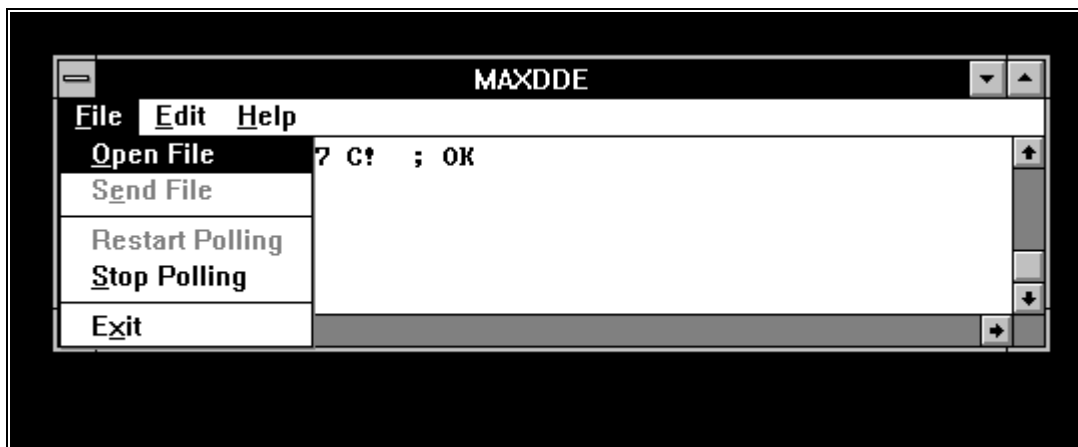
Periodically based on the value set in MAXDDE.INI for the [ReconfigureTime] entry, the database is evaluated for nodes that are not Initialized. These are nodes that come up as NOT BOOTED. If any are found, the timer system that monitors and controls scheduling is temporarily shut down. The COM port is reset, and an attempt is made to reboot the downed nodes.

While MAXDDE is polling, you cannot download a file with the **File->Send File** menu item. While MAXDDE is polling typing of characters is ignored.

Terminal Mode

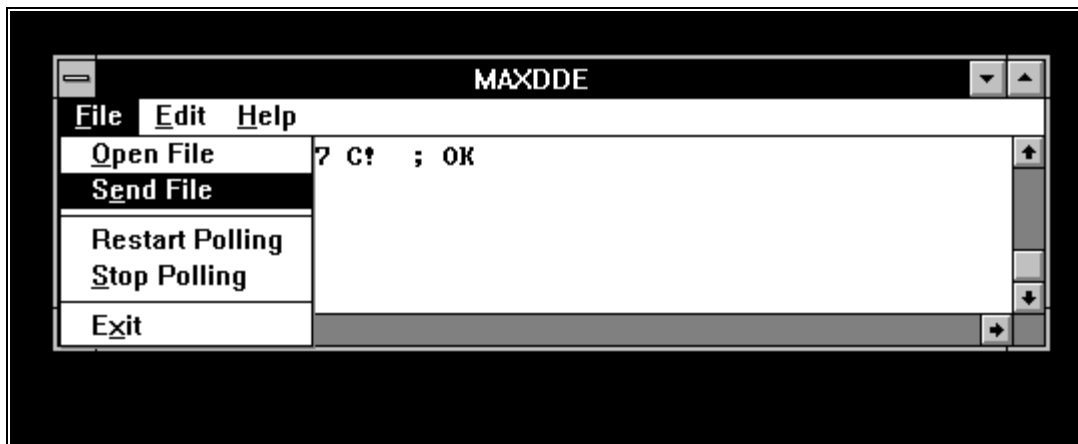
When you select the **File->Stop Polling** menu item MAXDDE switches to the terminal mode of operation. MAXDDE is now a power debugging terminal. From this mode of operation you can select a node following the EASY-A protocol, test any node for its presence on the network, send, and test Max-Forth words and source files (.FTH files).

Open File



The **Open File** menu item summons an editor. This option is setup in the MAXDDE.INI file under the **[editor]** keyword. It is defaulted to the Windows NOTEPAD program.

Send File

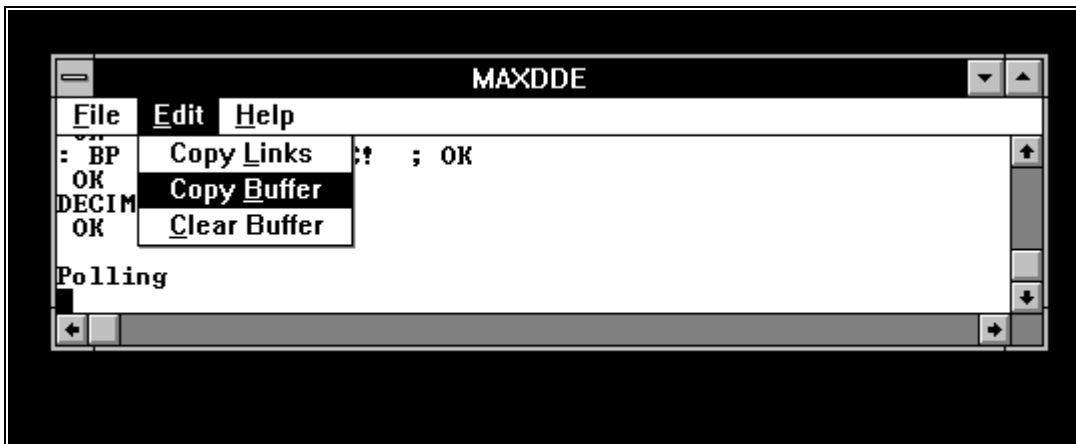


The **Send File** menu option lets you select a .FTH file and send it to a node on the EASY-A network. The download is done following the appropriate text transfer protocol. (Wait for echoed character, send the next line upon reception of a linefeed.)

Debug Note:

It is extremely important to remember that at the time the **File->Stop Polling** menu item was selected, polling was actively taking place. This means that when polling stopped, the last node being polled has the terminal's attention. You must select the address of the node you wish to be talking to following the EASY-A protocol (**CTRL+A** followed by the node's address character.)

Copy Buffer



The **Copy Buffer** menu item lets you copy the contents of the screen buffer to the clipboard for future pastes into an editor or other application.

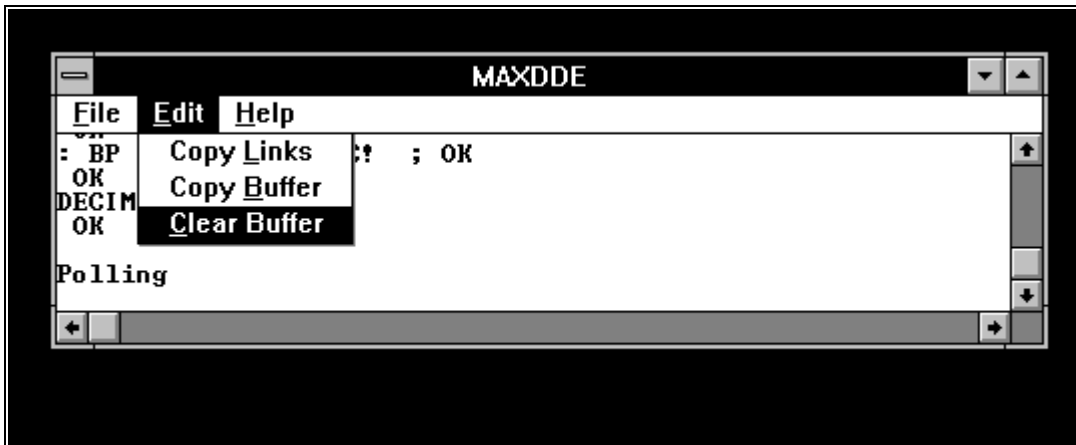
Application Note:

You can quickly copy the buffer to a file following this sequence of commands:

- ☐ Select Edit->Copy Buffer
- ☐ Select File->Open File
- ☐ Select any text editable file to get into the editor
- ☐ From the editor select File->New to get a new file opened.
- ☐ From the editor select the Edit->Paste menu item to Paste the contents of the clipboard into the file

- ☐ From the editor select the **File->SaveAs** menu item to save the buffer to a file.
- ☐ Exit the editor to return to MAXDDE.

Clear Buffer



The **Clear Buffer** menu item simply erases the contents of the Buffer.

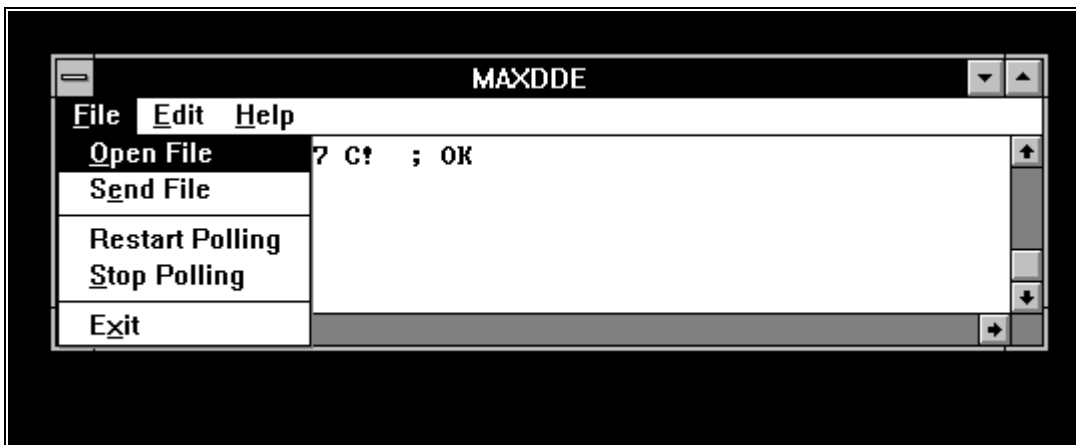
DDE Link Status Messages

If DDE links are active for items, the following messages can be sent as status messages. The DDE client application should be aware of these messages and trap for them when necessary.

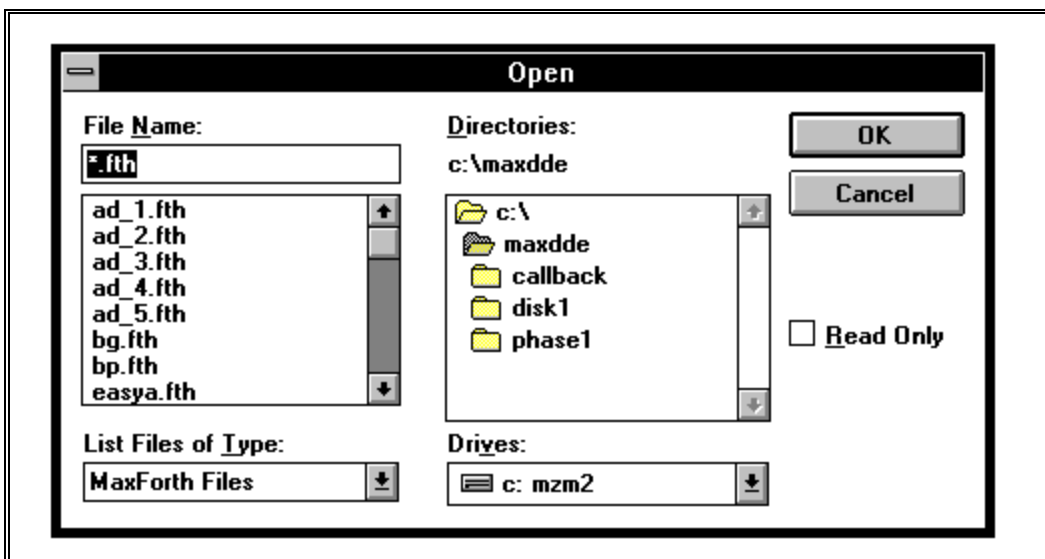
Stopped	The user has selected the File->Stop Polling menu item.
BOOTING	The item is in the node/item initialization process.
LostItem	The item was lost while polling.

MAXDDE File Menu Items

File->Open File

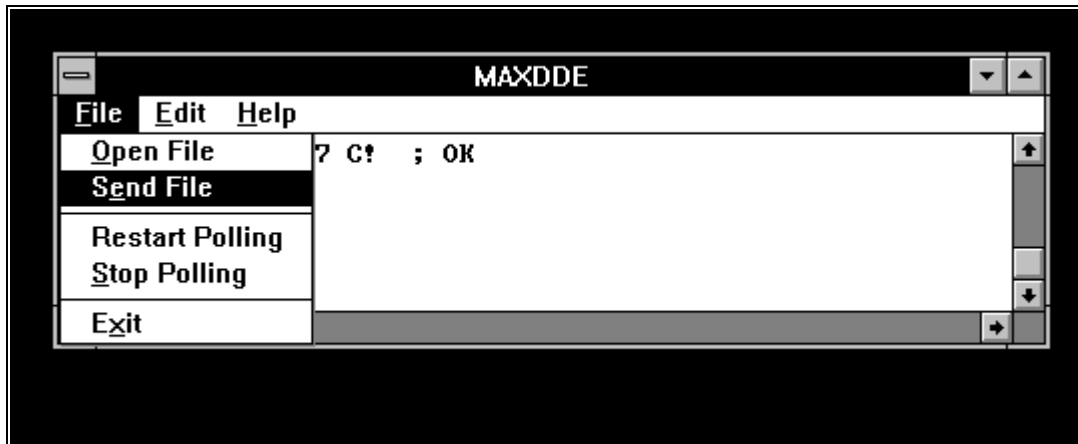


This command summons the editor. You will be given the opportunity to choose a .FTH file first. Choice is manifest by the Common File select dialog box.



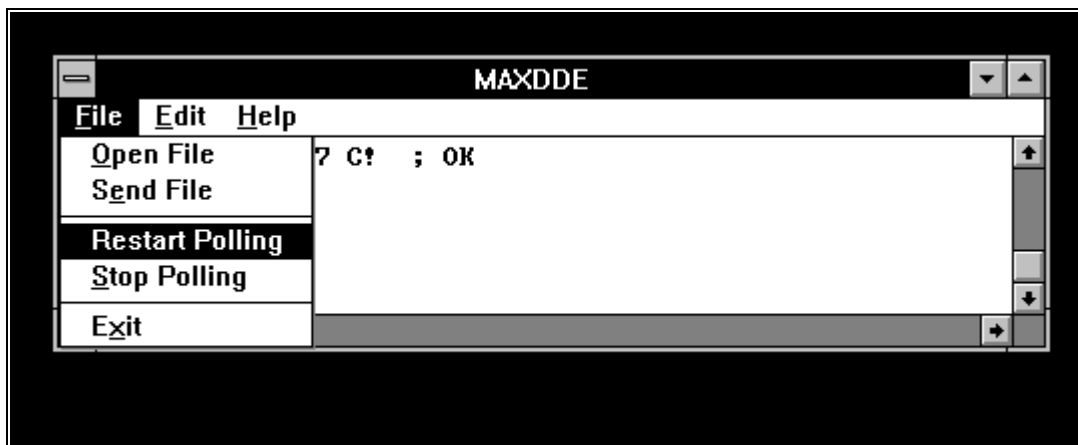
The editor can be set from the MAXDDE.INI file found in the Windows root directory. The [editor] section is defaulted to the Notepad application.

File->Send File



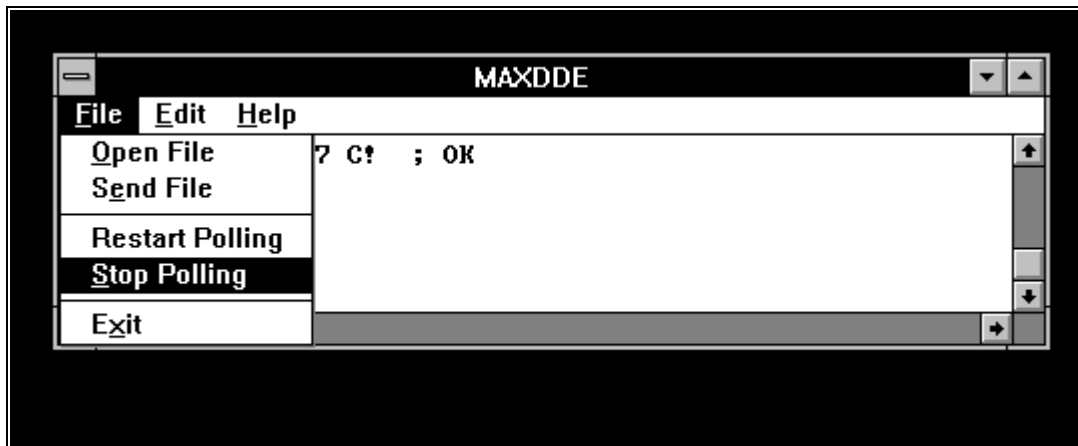
File->Send File lets you select a Max-Forth source file and download it to the EASY-A node in focus. Remember to set an EASY-A node into focus before using this option. File downloading is stopped by the **File->Stop Polling** menu item.

File->Restart Polling



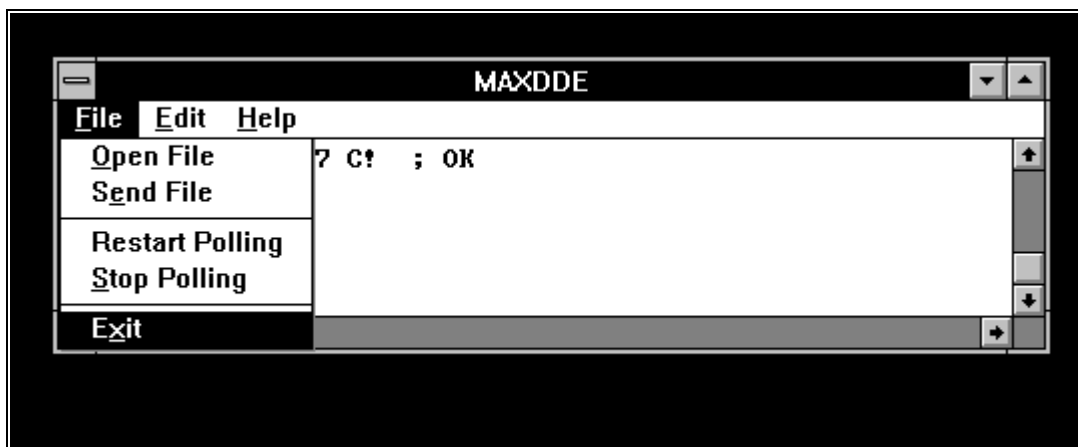
The **File->Restart Polling** menu item is only active while MAXDDE is in terminal mode. When selected it switches MAXDDE back to Polled mode. All initialization files are downloaded and polling commences.

File->Stop Polling



The **File->Stop Polling** menu item stops polling and switches MAXDDE to its terminal mode of operation.

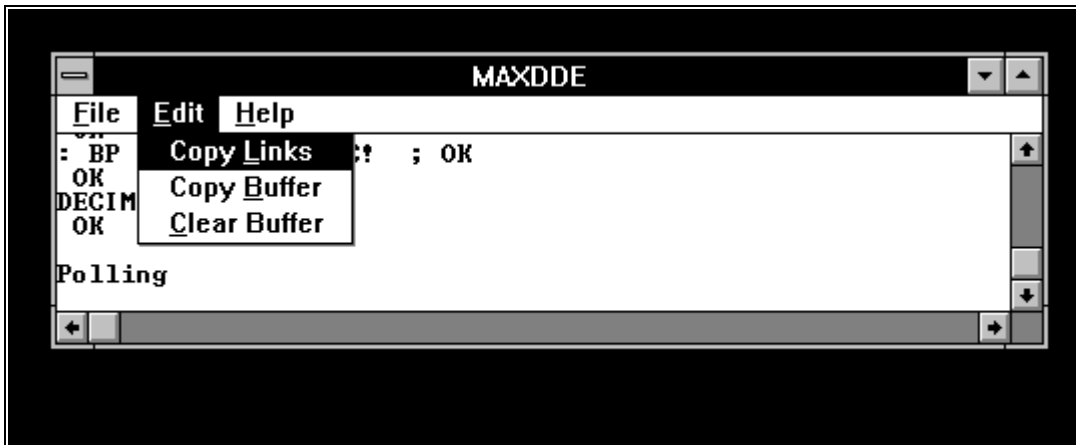
File->Exit



The **File->Exit** menu item lets you exit MAXDDE.

MAXDDE Edit Menu Items

Edit->Copy Links

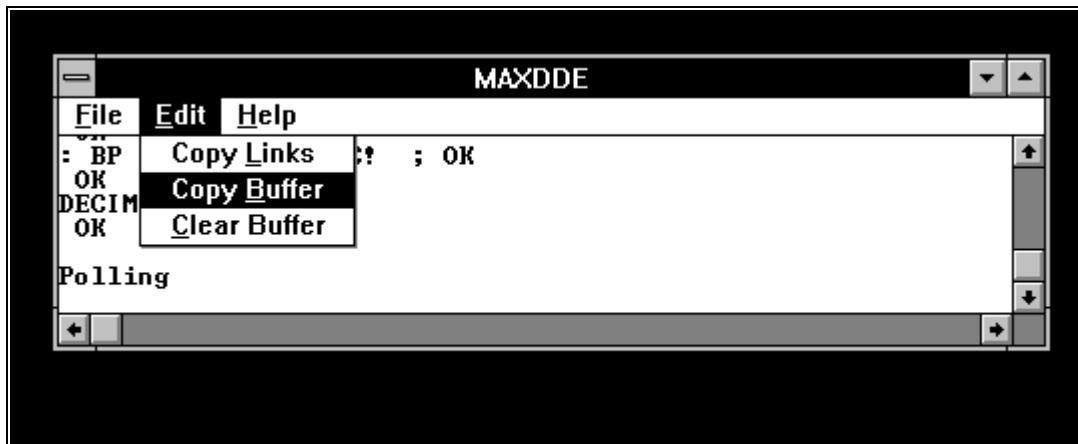


The **Edit->Copy Links** copies all of the links to a Microsoft standard link syntax. It follows the **Application|Topic!Item** format that works with Microsoft products.

Application Note

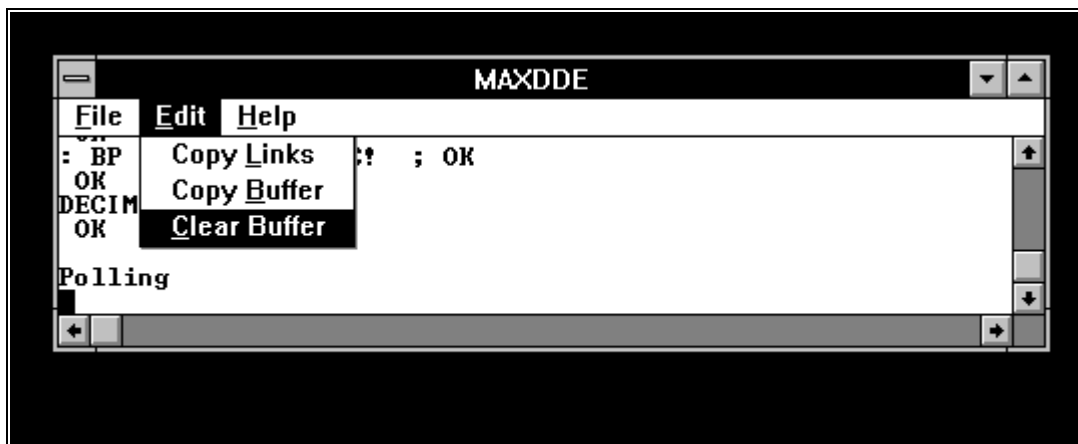
You can use this format with EXCEL by simply Pasting the clipboard into EXCEL and prefixing each cell with an equal sign (=) signifying that the cell contains a formula. Other client applications could parse this string to get the Application, Topic and Item for their own unique syntax.

Edit->Copy Buffer



Edit->Copy Buffer copies the contents of the 512 line buffer to the clipboard.

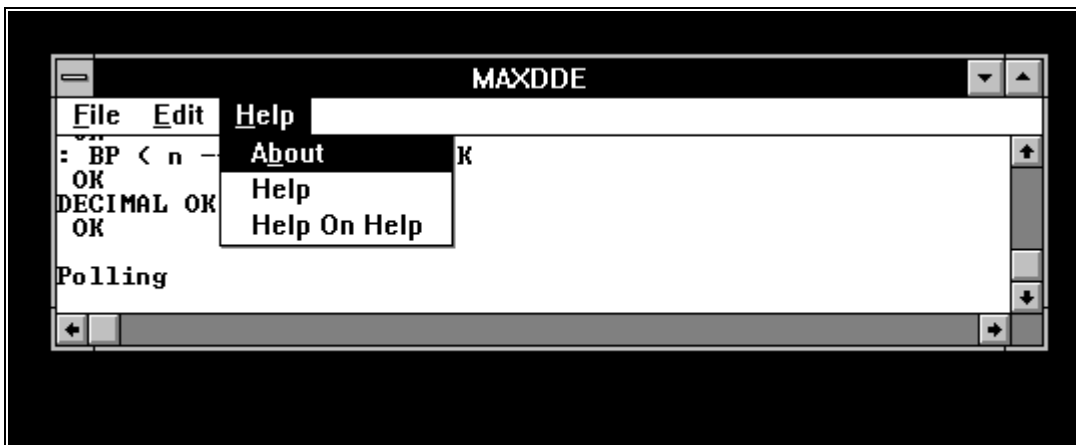
Edit->Clear Buffer



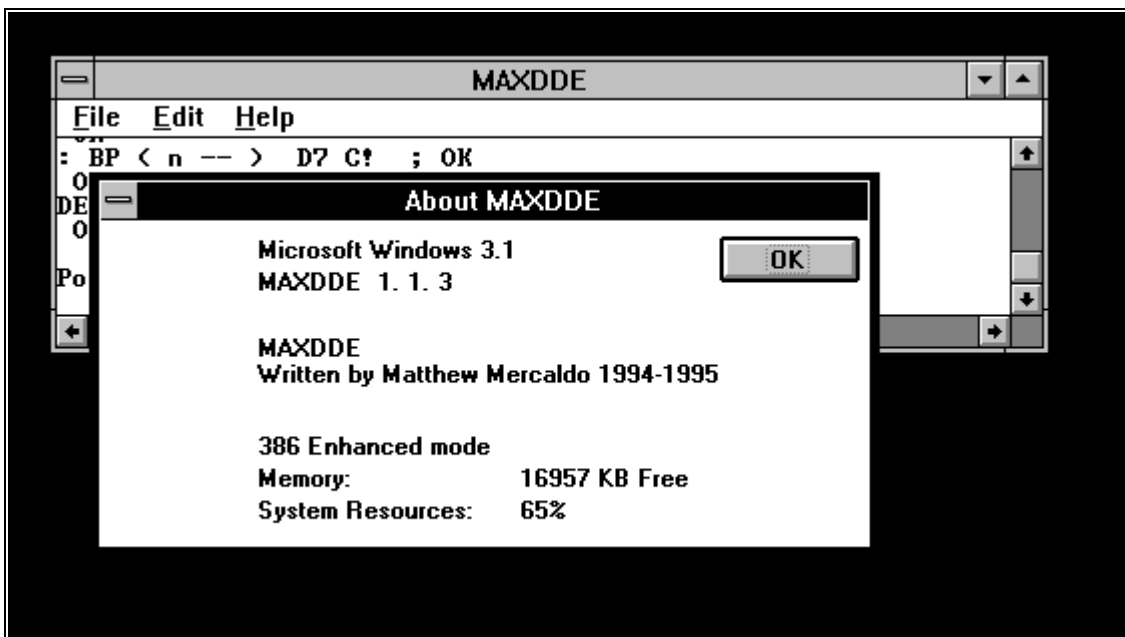
Edit->Clear Buffer clears the 512 line buffer and resets the scroll bars to the bottom of the buffer.

MAXDDE Help Menu Items

Help->About



The About menu item summons the About dialog box.



The about dialog box tells you some information about your computer's loading at the time you are running MAXDDE.

Help->Help On Help

Help On Help is a Windows basics help file. You can use this to learn more about the Windows operating system.

Help->Help

Unfortunately MAXDDE 1.1.3 does not have a help system fully installed.
(Engineers are working around the clock to create the help files.)

Trouble Shooting

You are here because it didn't work. We must first determine what went wrong.

Is the Network and wiring Correct?

The first thing to check is if the EASY-A network of nodes is running properly. This can be done by using the Terminal capabilities of MAXDDE. With the proper COM port setup, Select the **Stop** item in MAXDDE's **File** menu. This turns MAXDDE into a terminal program.

Try to access all of the nodes on your EASY-A network. To access a node on the EASY-A first type **CTRL+A** followed by the node address (a character in the set of uppercase letters [A..Z]). If you do not get a response back then we know that something is wrong on either the RS-422 converter or the network nodes themselves. If something is wrong with your EASY-A network, it is beyond the scope of this help guide due to the applications specific nature of your network (Are you sure that the nodes have power?). You can test the converter and or its wiring by simply looping it back. Looping back simply means connecting the transmit lines to the receive lines. (TX+ to RX+ and TX- to RX-.) If you do not see the characters you have typed, echoed back then something is wrong with your wiring and you need to re-examine it for errors.

Is the MAXDDE.INI file Properly Configured?

If the network appears to be OK then *check to see that the MAXDDE.INI file found in the Windows directory is configured properly.* All items should have an associated .FTH file in the MAXDDE directory. There should also be a node initialization file in the MAXDDE directory. MAXDDE will try to send these files to the appropriate nodes on the network. See the manual on this topic under the **MAXDDE Overview** section.

Is the PC Hardware Properly Configured?

If you PC locks up upon entering MAXDDE you could have a conflict between a motherboard based serial port and an installed card serial port. It is possible to configure the local serial port as inactive with active interrupts. If you are using the card based serial port then the local port must be completely disabled.

MAXDDE and Sample Applications

All of the sample programs require at least one EASY-A node with EASY-A.FTH resident in EEPROM at address A.

EXCEL can connect quickly and easily to MAXDDE. The sample code MAXDDE.XLS is a simple EXCEL monitoring program. EXCEL can be used as a MAN-MACHINE interface through its macro language. See MAXDDE.XLS for a simple EXCEL monitoring example.

Quattro didn't really work the way the standard says it should. The way to access DDE in Quattro is through the macro language. See the sample code called MAXDDE.WB1.

Visual Basic is the "Volkswagen" of PC languages! To do DDE, the trick is to create a text box on a form, then connect that textbox with DDE. This is all done without a line of code. Each object (such as a textbox) has a property list; text box properties include ddelink attributes that are simply filled in at creation time OR can be modified at run time. Visual basic lets the developer quickly turn rapid prototypes into standalone applications in a simple one step operation. I think with the correct group of "gadgets" (histogram, speedometer and other like objects) one could develop screenware comparable to Wonderware! See the Application note called **Visual Basic Application Note 1**.

If you have Visual Basic 3.0 loaded you can load **PROJECT1.MAK**.

To restate - Microsoft recommends keeping to a Text format to exchange information over DDE. Microsoft products keep to this - Other products (Quattro) will but through coercion.

Appendix A. MAXDDE.INI

What MAXDDE knows about the content of the EASY-A network that it is supposed to be connected to is found in the MAXDDE.INI file that is found the Windows directory. This file is loaded at startup time.

This is an example of a MAXDDE.INI file.

```
[itemsReceived]      These are items that MAXDDE will get
item1=AD_1           There will exist a word called AD_1
item2=AD_2
item3=AD_3
item4=AD_4

                        item1 - item5 are tags and can be anything followed by
                        '='

[itemsSent]           These are items that MAXDDE will set
item5=BP              There will exist a word called BP

[AD_1]
scan=100              AD_1 is scheduled to run every 100 ms
node=A,B              AD_1 is to be found on NODEA and NODEB
                        Node values allowed are A to Z and 0 to 9.

[AD_2]
scan=500
node=A,B

[AD_3]
scan=500
node=A,B

[AD_4]
scan=500
node=A,B

[BP]
scan=500              BP will run every 500 ms
node=A,B              BP will be found on NODEA and NODEB

[debug]
SHOWFILES=OFF         Debug lets you see what is happening
                        When the Node initialization files and Item files are
                        being downloaded they will be seen on the screen

SHOWPOLL=OFF          When items are being polled, they will be displayed to
                        the screen.

[editor]
editor=NOTEPAD         This tells the Edit->Edit Max-Forth File menu option
                        what editor to use.

[COMPort]
Port=2                The COM Port is configured to COM2. Values allowed are
                        1, 2, 3, or 4.

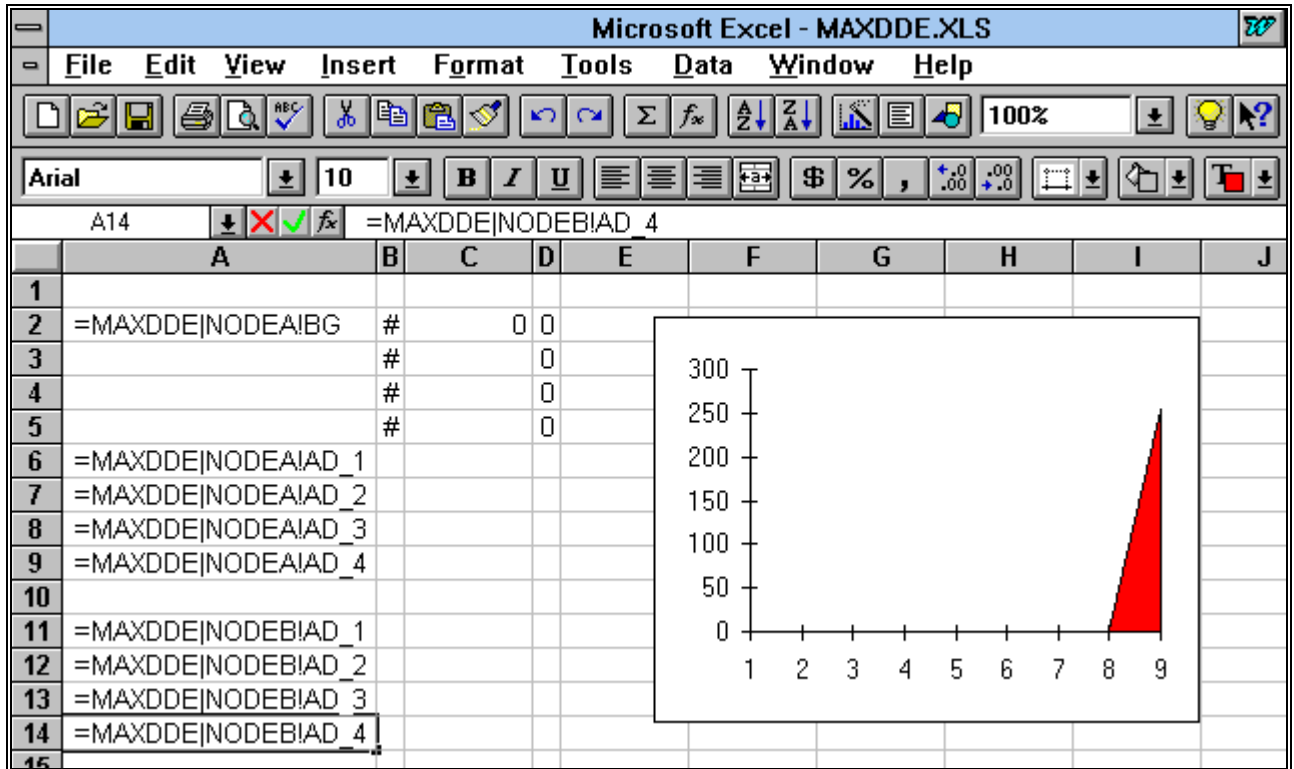
[BAUDRate]
Speed=9600             The BAUDRate is configured here. Allowed values are
                        4800, 9600, 14400, and 19200.

[ReconfigureTime]
```

seconds=10

This is the time when the network sentinel will check if the network is operating properly. If it is not it will force a reconfigure. Allowed values are from 1 to 60.

Appendix B. MAXDDE and EXCEL



The Excel spreadsheet option is useful for monitoring. An Excel macro language programmer can add data acquisition capability, mathematical analysis capability, DDE Poke capability, as well as file I/O capability.

Please note that the picture above displays the formulas in the cells. The cells would have numbers displayed or a message stating the condition of the link. This formulas are displayed in the cells for clarity of the formulas. This example can be opened from EXCEL as MAXDDE.XLS found in the MAXDDE directory.

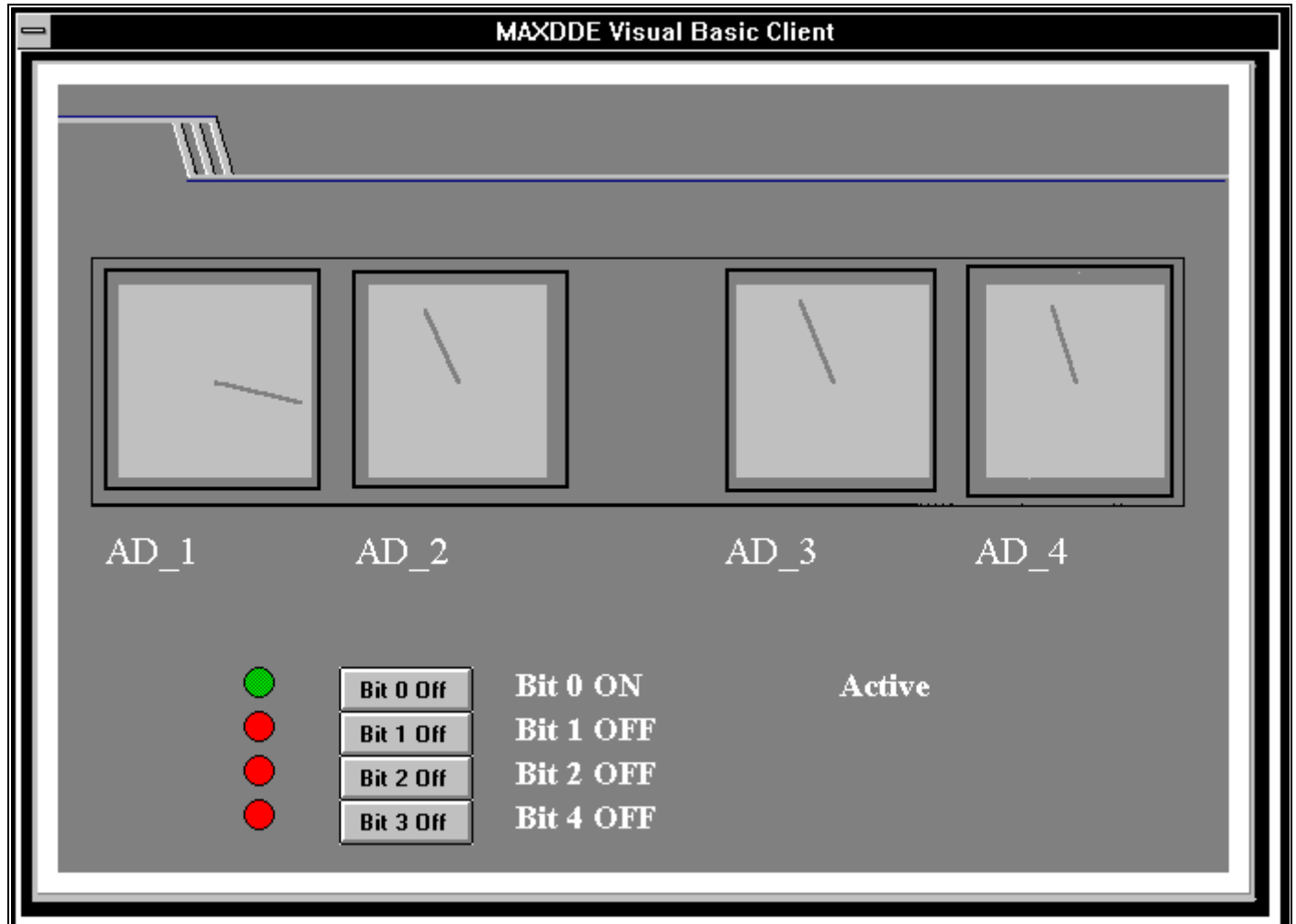
Appendix C. MAXDDE and Quattro

Quattro Pro for Windows - MAXDDE.WB1					
File Edit Block Data Tools Graph Property Window Help					
A:B8					
	A	B	C	D	E
1	2	{INITIATE "MAXDDE","NODEA",CHANNEL}			
2	56	{REQUEST CHANNEL,"AD_1",VALUE1}			
3	42	{REQUEST CHANNEL,"AD_2",VALUE2}			
4	44	{REQUEST CHANNEL,"AD_3",VALUE3}			
5	44	{REQUEST CHANNEL,"AD_4",VALUE4}			
6		{TERMINATE CHANNEL}			
7		{RETURN}			
8					
9					
10					
11					
12					
13					
14					
15					

Quattro communicates to MAXDDE by using its macro commands on a workbook. As seen above, cell B1 initiates or opens a link with MAXDDE and writes the channel number to cell A1 which has been named CHANNEL. It then does a DDE request of the data. This implements a cold link or polled link. When the data has been retrieved the channel is terminated in cell B6. A Quattro macro programmer can write programs that poll for data as well as poke data. This can be incorporated into a data acquisition program that does application specific work.

MAXDDE.WB1 can be found in the MAXDDE directory.

Appendix D. MAXDDE and Visual Basic



The Visual Basic program developed lets the user set bits via the **Bit 1 Off** through **Bit 3 Off** keys. When pressed they say **Bit 1 On** through **Bit 3 On**. The text to the right of the buttons tells the status of the input bits and the "LEDs" the left of the buttons turn RED when an input is OFF and GREEN when an input is ON. The gauges are Visual basic controls. There is a bit map backdrop for the whole form that would have the markings for the gauges given units of measure etc.

This program can be found as file **PROJECT1.MAK**. You will need the Profession Developer's Version of Visual Basic 3.0.

Appendix E. MAXDDE and EASYA.FTH

EASYA.FTH is a Max-Forth file that implements EASY-A as well as a small data acquisition package. Upon autostart, a 16.38 ms interrupt is established that monitors 4 AD channels (PE.0 - PE.3) debounces 4 bits (PE.4 - PE.7) and will write 4 bits (PA.4 - PA.7) - with even a small amount of room left over!

AD_1.FTH through AD_4.FTH read locations in the internal ram - at the end of the TIB that the interrupt sources with the AD values. The bits are debounced and are also put in a location at the end of the TIB. This is done by the file BG.FTH. BP.FTH puts bits found in the high order nibble of the passed value into PORTA.

Please note that because interrupts are running, one can not change the EASYA node address online. A disable interrupt word needs to be sent to the node and executed (Unless one is in Max-Forth) before changing the address.

```
CODE-SUB DI
0F C,      ( SEI to disable interrupts )
39 C,      ( RTS to exit the routine   )
END-CODE
```

Please look over EASYA.FTH, AD_1.FTH, AD_2.FTH, AD_3.FTH, AD_4.FTH, BP.FTH as well as BG.FTH as a model for writing MAXDDE aware words. All of these files can be found in the MAXDDE directory.

Appendix F. EASYA.FTH

Caution!

This program should only be loaded into a MAXFORTH V3.5E F68HC11 component. Loading this program into other NewMicros based products will result in improper operation.

This program can be loaded into the chip using the **Edit->Download Max-Forth File** menu option. Local to the MAXDDE directory you will find EASYA.FTH. After downloading the file set the new address as appropriate. To set the address to a new address *first Disable Interrupts (See Appendix E. above)*, then set the address to B7BE.

Remembering that A is 41 hex and Z is 5A hex type the following for address A.

41 B7BE EEC!

The following is a listing of EASYA.FTH

```
COLD
HEX
( -----
CODE-SUB A-C
F1 C, B7BE ,      ( CMPB B7BE
39 C,              ( RTS
END-CODE
EEWORD
( -----
CODE-SUB N-I
( GET CHARACTER
( LABEL E1
BD C, F8CB ,      ( JSR OLD-KEY-SUB
C1 C, 01 C,       ( CMPB # ^A
26 C, 22 C,       ( BNE E2
( IF == ^A
( POP OFF THE STACK
18 C, 08 C,       ( INY
18 C, 08 C,       ( INY
( TURN OFF TRANSMIT
F6 C, B000 ,      ( LDAB PORTB
C4 C, F7 C,       ( ANDB #F7
F7 C, B000 ,      ( STAB PORTB

( GET ADDRESS
BD C, F8CB ,      ( JSR OLD-KEY-SUB
( TAKE CHAR OFF STACK
18 C, 08 C,       ( INY
18 C, 08 C,       ( INY
BD C, ' A-C @ ,   ( CHECK ADDRESS
( IF NOT MY ADDRESS
( GO BACK TO GET CHAR
26 C, -1F C,      ( BNE E1:
( IF THIS IS MY ADDRESS
( TURN ON TRANSMITTER
F6 C, B000 ,      ( LDAB PORTA
CA C, 08 C,       ( ORAB #08
F7 C, B000 ,      ( STAB PORTA
( GO BACK TO GET CHAR
20 C, -29 C,      ( BRA E1:
( CHARACTER IS NOT ^A
( CHECK MODE
( LABEL E2:
B6 C, B000 ,      ( LDAA PORTA
84 C, 08 C,       ( ANDA # 08
26 C, 06 C,       ( BNE E3:
( IF MODE NO GO
( TAKE CHAR OFF STACK
18 C, 08 C,       ( INY
```

```

18 C, 08 C,      ( INY
( GO BACK TO GET CHAR
20 C, -36 C,     ( BRA E1:

( IF MODE GO THEN GET CHAR
( LABEL E3:

4F C,           ( CLRA
( STRIP 8TH BIT
C4 C, 7F C,     ( ANDB # 7F
18 C, ED C, 00 C, ( STD 0 ,Y
39 C,           ( RTS
END-CODE
EEWORD
( -----
CODE-SUB RTII
( ACK THE INTERRUPT
86 C, 40 C,     ( LDAA # 40
B7 C, B025 ,    ( STAA B025 -
                ( ACK INTERRUPT

( START AD CONVERSION
86 C, 10 C,     ( LDAA # 10
B7 C, B030 ,    ( STAA B030 ADCTL
( GET BITS FROM HIGH OF PORTE
B6 C, B00A ,    ( LDAA B00A PORTE
44 C,           ( LSRA
44 C,           ( LSRA
44 C,           ( LSRA
44 C,           ( LSRA
D6 C, D5 C,     ( LDAB D5 - INDEX
CE C, 00CD ,    ( LDX # CD - BITQ
3C C,           ( PSHX
3A C,           ( ABX
( STORE SAMPLE INTO THE Q
A7 C, 00 C,     ( STAA 0 ,X
( NORMALIZE INDEX
5C C,           ( INCB
C4 C, 03 C,     ( ANDB # 03
D7 C, D5 C,     ( STAB D5 - INDEX
( SEE IF WE HAVE A DEBOUNCE

```

```

38 C,           ( PULX
A6 C, 00 C,     ( LDAA 0 ,X
A1 C, 01 C,     ( CMPA 1 ,X

26 C, 0A C,     ( BNE L40 +10d
A1 C, 02 C,     ( CMPA 2 ,X
26 C, 06 C,     ( BNE L40 +06d
A1 C, 03 C,     ( CMPA 3 ,X
26 C, 02 C,     ( BNE L40 +02d
97 C, D6 C,     ( STAA ^ D6
( LABEL L40
( WRITE BITS FOUND IN D7
B6 C, B000 ,    ( LDAA ^ PORTA
84 C, 08 C,     ( ANDA # 8
9A C, D7 C,     ( ORAA ^ D7
B7 C, B000 ,    ( STAA ^ B000
( DO A/D CONVERSION COMPLETE
B6 C, B030 ,    ( LDAA B030
84 C, 80 C,     ( ANDA # 80
27 C, -7 C,     ( BEQ L40
CE C, 00D1 ,    ( LDX # D1
B6 C, B031 ,    ( LDAA B031
A7 C, 00 C,     ( STAA 0 ,X
B6 C, B032 ,    ( LDAA B032
A7 C, 01 C,     ( STAA 1 ,X
B6 C, B033 ,    ( LDAA B033
A7 C, 02 C,     ( STAA 2 ,X
B6 C, B034 ,    ( LDAA B034
A7 C, 03 C,     ( STAA 3 ,X
3B C,           ( RTI

END-CODE
EEWORD
( -----
CODE EI
0E C,          ( CLI
39 C,          ( RTS

END-CODE
EEWORD

```



```

( -----
: ENA

[ ' N-I @ ] LITERAL
4 @ 10 + !

40 B025 C! ( CLEAR INTERRUPT

00 B024 C! ( REMOVE INTERRUPT ENABLE

B026 C@ 8A OR B026 C! ( PA3 AS OUTPUT
                      ( PA7 AS OUTPUT
                      ( RTII TO 16.38

40 B025 C! ( CLEAR INTERRUPT

40 B024 C! ( SET interrupt MASK

00 D5 C! ( set index to 0

00 D6 C! ( set rcv bits to 0

00 D7 C! ( set tx bits to 0

EI ( ENGAGE
;

EEWORD

( -----

8A B026 C!

8 B000 C!

7E B7E6 EEC!

' RTII B7E7 2 EEMOVE

1FC AUTOSTART ENA

A44A 1FC !

1FC 4 DUMP

1FC B600 4 EEMOVE

41 B7BE EEC! ( address to 'A'

HERE U.

```