Version

3

MCS ELECTRONICS

Making Things Easy

EASY-TCP/IP Guide

MCS ELECTRONICS

EASY – TCP/IP Guide

www.mcselec.com

Table of Contents

Chapter

1

Introduction

With the Internet gaining in popularity and ADSL becoming available in many countries, we can do business from anywhere in the world. Tele banking, sending and receiving emails, shopping, it is all-possible from any location in the world that has access to the internet.

Although many people still use a dial-in connection, within a few years an Internet outlet will be as common as a mains outlet.

Already many people have a small LAN. (local area network) that consist of 2 or more computers.

> A LAN or WAN(Wide Area Network) is a number of computers that are connected to each other. The purpose is to share information or resources such as a printer, scanner or disk.

Much household equipment will be equipped with a RJ-45 connector in the future. Imagine an oven that can get new recipes from an internet database or a TV that can be controlled from your PC, or a VCR that can be programmed via a simple web interface? I know this sounds far-fetched but only the future will tell.

While PC's are cheap and can run big applications, they have the downside that they must be constantly powered-up and consume a lot of power.

Embedded systems do not have this disadvantage. They consume only little power. But until the W3100A chip was invented by Wiznet, it was hard to connect a micro to the internet.

MCS evaluated many chips that offer Ethernet connectivity but they all needed a lot of resources (either program code and/or memory) to perform a simple task.

The W3100A chip, from Wiznet (www.i2chip.com) has changed that.

The W3100A offers four socket connections and has internal memory for receiving and sending data. Since the W3100A is also offered in module form (IIM7000A) that can be used without the need for soldering SMD components, MCS Electronics decided to support this powerful chip.

Don't worry if you don't understand TCP/IP because Easy TCP/IP is developed with the novice in mind.

Chapter 2

Experience

This chapter explains which user experience is required.

In order to use Easy TCP/IP successfully, you must have a small network. The minimum requirement is that you must have one PC with a working network card. Basic knowledge of BASCOM-AVR is required.

As this manual focus on the Easy TCP/IP extension, it will not describe how BASCOM-AVR works. BASCOM-AVR is however very simple to use.

You need a commercial version of BASCOM in order to compile the sample programs. The free DEMO version is limited to 4KB and will only compile the small tcp/ip programs.

Chapter

3

TCP/IP Basics

This chapter explains the TCP/IP basics and is intended to get some more background.

Networks are invented to exchange data between multiple terminals and/or computers. A protocol is needed so computers can understand each other. You can see a protocol as a language.

The most popular network protocol is TCP/IP. TCP means Transmission Control Protocol.

There are multiple layers in the network protocol and the ones we use are the hardware layer(your network card and the Easy TCP/IP board) and the application layer(the software that uses the protocols).

Both TCP/IP and RS-232 are serial data protocols. In the PC, in place of the serial port, we use a LAN card. Externally we can connect it to a stand-alone Ethernet board such as Easy TCP/IP, TINI, etc.

In contrast to RS-232 TCP/IP is a much more advanced protocol. Why is such a complex protocol needed?

A protocol is just an agreement between two or more parties so they can communicate with each other. For RS-232, the baud rate, number of data bits, etc. must be known by both parties otherwise the sender and receiver can't communicate at all, or will exchange false data.

There is a lot to write about how TCP/IP works. Only the parts that are needed will be discussed.

With TCP/IP we use "sockets" to communicate between two applications. You can think of a socket like a channel in QB/VB. When you use : OPEN "file" for BINARY as #1 , you open a channel that is used to read or write data to the file named "file". The channel number is all that is needed to identify which file you are writing to or reading from.

With TCP/IP these channels are named sockets. You must open a socket before you can communicate with a client or server.

We define a server as a process that is waiting for clients to connect.

Most servers allow multiple clients to connect at the same time.

We define a client as an application that connects to a server.

When you read your email, you actually connect to a mail server. Your email program is called the client or email-client.

When there are multiple servers, how can we specify which server we want to connect to? We do this by specifying an IP address. IP addresses consist of 4 bytes separated by dots : 192**.**168**.**0**.**10 for example

Although each byte represents an 8 bit quantity, the numbers are expressed as decimal numbers. (Actually a LONG integer is used to store the address).

When you use your web browser you also specify the IP address http:// 64.5.37.122

This will connect you to mcselec.com. Since people are not good in remembering 32 bit numbers, a protocol was invented called DNS (domain name service).

This service will translate the domain name "mcselec.com" to its corresponding IP address.

Every IP address is unique. So IP # 64.5.37.122 will always bring you to the mcselec.com server.

The **http://** you enter in your browser tells the browser that you will use the HTTP protocol. The default port for HTTP servers is port 80.

So what is this port about?

You can see the port as a department inside an office. The office address is the same for all departments, but to address a department you must specify the department name.

Do not confuse Ports with the microprocessor ports. See the port as a sub address.

The port number is a word in the range from 0 – 65535. A lot of ports are used by well-known services (a list is provided at the end of this document).

By default a web server uses port 80, and a POP3 email server uses port 110.

Of course you can also create a web server that is listening to port 5000. In that case the client (browser) must specify that it wants to connect to port 5000.

For example : http:// 64.5.37.122**:5000**/index.htm

The additional **:5000** is the port number of the server you want to connect to.

To make it a bit more complete, not only the server is using a port, but the client also needs to use a port. We call this the local port.

With the **netstat** command at the command prompt you can view the active connections and ports.

Active connections

Proto	Local address	External address	Status
TCP	lptp:3803	66.102.11.99:http	ESTABLISHED
TCP	lptp:3804	66.102.11.99:http	ESTABLISHED

To conclude, we always need a socket, and we need to act as a client or as a server.

Once a connection is established we can send or receive data. What kind of data we transmit depends on the protocol. The web protocol for example is a simple text orientated protocol.

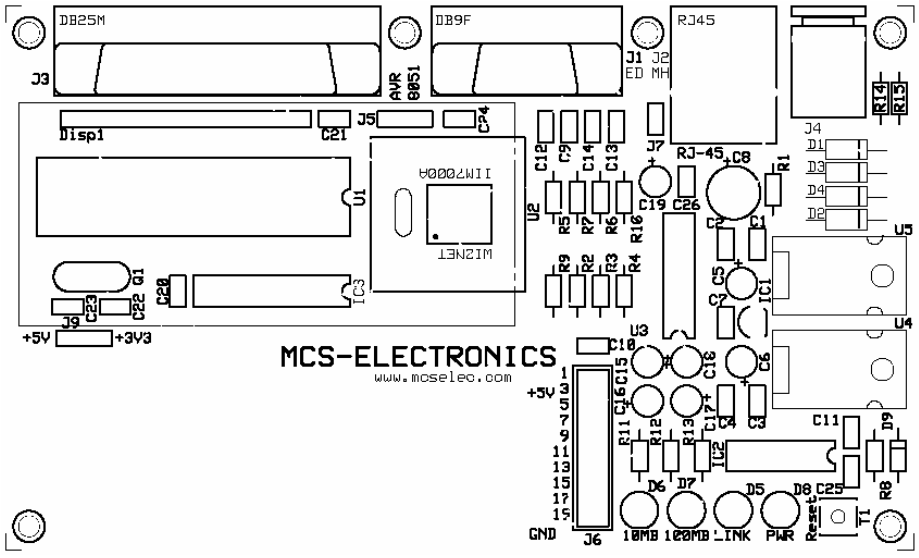
We send and receive plain ASCII text. But you could write a server that will send and receive binary data too.

Chapter

4

Getting Started

Now lets get started. Things will become clearer once you have a unit to try! Start by assembling the PCB. Mount and solder the power supply/passive components first:



You can use the order of the part list on the next page. In any case it is advised to use sockets for the IC's and the IIM7000A.

Component	Description
R1	1K resistor
R2,R3,R4,R9	51 resistor, 1%
R5,R6,R7	150 resistor
R8	10K resistor
R10	22 resistor. Optional for backlight of LCD
R11,R12,R13	1K resistor

R14,R15	390 ohm resistor
D1,D2,D3,D4	1N4001. Watch out for the polarity.
D9	3V3 zener. Watch out for the polarity.
C1, C2, C3, C4, C7, C9, C10, C11, C13, **C14**, C20, C21, C24, C25, C26	Capacitor 100nF. C14 is optional and only needed for older IIM7000 (not IIM7000A) module.
C22,C23	22 pF Ceramic capacitors
C12	100 pF ceramic capacitor.
IC sockets	14 pin, 16 pin, 20 pin and 40 pin
Crystal socket	This is optional when you like to exchange the crystal easily.
U5	7805
U4	LD2517V33 or LD33V or a 3.3 V regulator in TO-220 house.
Female double header for U2	Optional when you want to reuse the module on another board. Note that 2 mm is used. That is, the distance between to pins on the IIM7000, is 2 mm and not 2.54 mm.
Female double header 20x2 for J6	Optional.
Male double Header for J7, J8, J5	
Female header for LCD	Optional when you want to connect an LCD display
D5, D6,D7	Led , yellow, 3 mm. The round pad is the anode.(longest wire of the LED)
D8	Led, green, 3 mm. The round pad is the anode.(longest wire of the LED)
C15, C16, C17, C18 , C19	Capacitor, 1 uF/35V. Watch out for the polarity
C5, C6	10 uF/35V capacitor. Watch out for the polarity

C8	220 uF/35V
T1	Reset switch
J3	DB-25 male connector
J1	DB-9 female connector
J2	RJ-45 connector with integrated transformer.
J4	Power connector
IC1	ZSM560C, TO-92. This power monitor IC is optional as most microprocessors have a brown out detection you can activate.

Do not insert the IC's yet.

Once you have soldered all parts, check the PCB for small solder dots between tracks and remove them.

Connect a power supply. The power must be in the range from 6-15 Volt DC.

The polarity is not important since a diode bridge is used. The power LED should light now. Measure the voltage of U4 and U5. The measured voltage must be 5V on pin 3 of U5 and 3.3V on pin 2 of U4. Notice that the 3V3 regulator is different compared to the normal 78XX series of regulators regarding the pins.

If the voltage is not right, disconnect the power and check the board.

It is best to use a 3.3V regulator that has its TAB connected to GND. When you have doubts, use isolation and a plastic/nylon nut and bolt to fasten the regulator.

If the voltages are ok, disconnect the power supply and insert the CMOS (74HC00 and 74HC573) and the MAX-232 serial buffer chip. Also insert the IIM7000A module. Take care to insert it right. The crystal on the IIM7000 must match with the position on the PCB.

It depends on the PCB version you have if the silk of the PCB matches the IIM7000A. As of version MCSTCPIP5, the silk matches the IIM7000A. Previous versions, were designed for the IIM7000. Version 5 of the PCB has clearly printed IIM7000A on the top of the PCB.

Insert jumper **J5** for the AVR platform. Also place the microprocessor. Use the Mega162(L). Other possible micro's are Mega161, 90S8515, Mega8515.

Also place jumper **J9**. The position depends on the used micro.

Use +5V for the AT90S8515. For all other micro's use 3.3V position.

Jumper **J7** must not be inserted. Use J7 only when you use an old IIM7000 module.

For the serial communication you need a straight cable. TX of the Easy TCP/IP is connected to RX(pin 2 of PC) .

The LPT cable for the programmer must be wired straight through (1:1). A common printer extension cable is suitable.

Also connect a network cable (cross or straight depending on your configuration).

Start BASCOM-AVR and select the Sample Electronics programmer. Select the proper LPT address. The BIOS of your PC must be set to EPP mode.

Connect the power and load the tcpip.bas example. Compile and program the chip.

The tcpip.bas sample sets the IP address of the Easy TCP/IP. You now should be able to PING the board from your PC!

Use PING from the command line :

PING 192.168.0.8

The 10 MB led should flicker and you should get a response. (When you have 100 MB network, the 100MB LED will flicker)

The board is now functioning and ready for some more advanced programs.

Note that in this document we assume that the IP address of the Easy TCP/IP board is **192.168.0.8.** You need to change the programs if you want to assign another IP address.

Also note that we will use IP address 192.168.0.3 for the PC network Card. You will need to change the programs if your PC uses a different IP address.

If it does not work, you may want to check the next section, which explains how to setup your network card.

IMPORTANT

Since the W3100A is used in external address mode, you need to turn ON the Options, Compiler, Chip, External Memory access mode. And since a lot of parameters are passed via the softstack, you need to make the softstack at least 64.

As the chip is used in external address mode, and some chips have a JTAG interface that is active on the used ports, you need to disable the JTAG interface by changing the fuse bits.

Chapter

5

Network card setup

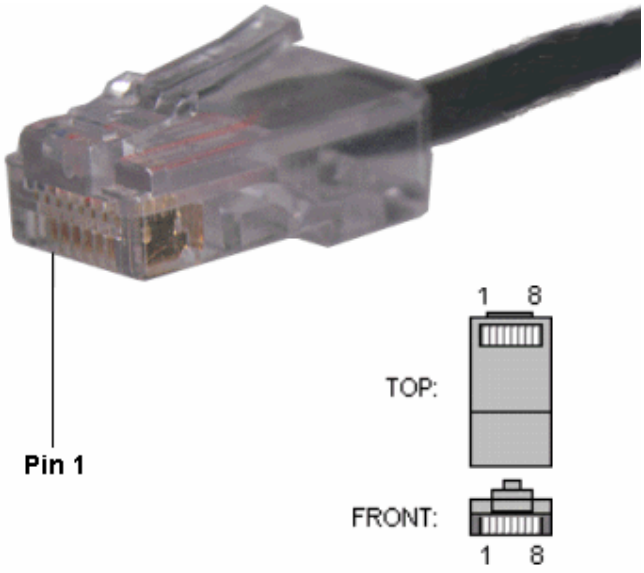
There are a lot of options to create a simple network. Typically you install a network card into your PC. This card can be connected to the Easy TCP/IP PCB with a crossover cable.

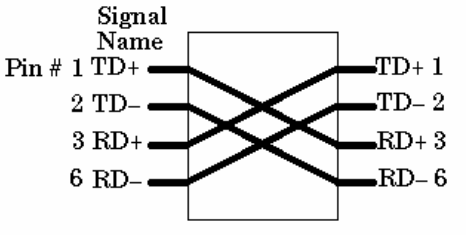
You can also buy a HUB or a switch and connect the PC LAN card to the HUB. The Easy TCP/IP must be connected to the HUB or router with a standard patch cable.

A HUB is a kind of RJ-45 splitter. It is used to connect the network cards of multiple PC's so the PC's form a LAN.

A switch is a more intelligent HUB. It routes traffic to make the network run faster.

When you do not have a HUB or switch and want to connect the Easy TCP/IP to your LAN card directly you need a crossover cable. You can buy one in a computer shop, or create one yourself:





The figure above shows the correct wiring for a 10Base-T crossover cable (assuming you're using RJ-45 connectors).

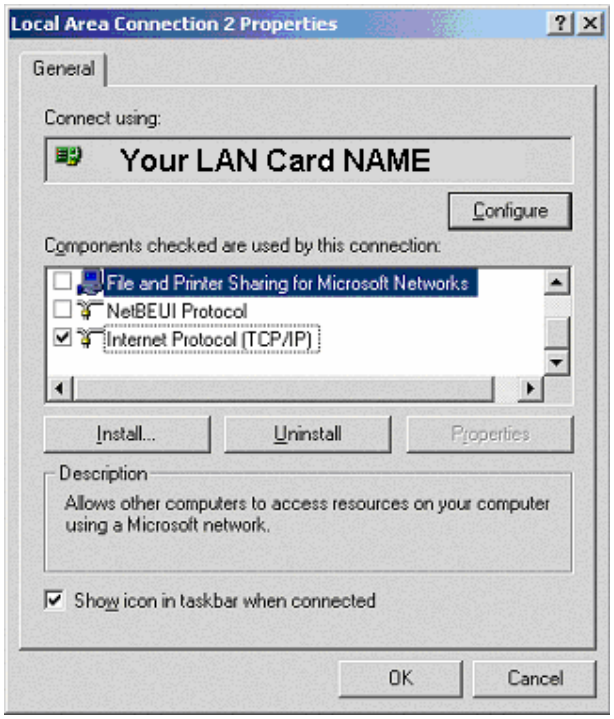
Only use a cross cable when you connect the Easy TCP/IP board directly to the network card of your PC!

Chapter

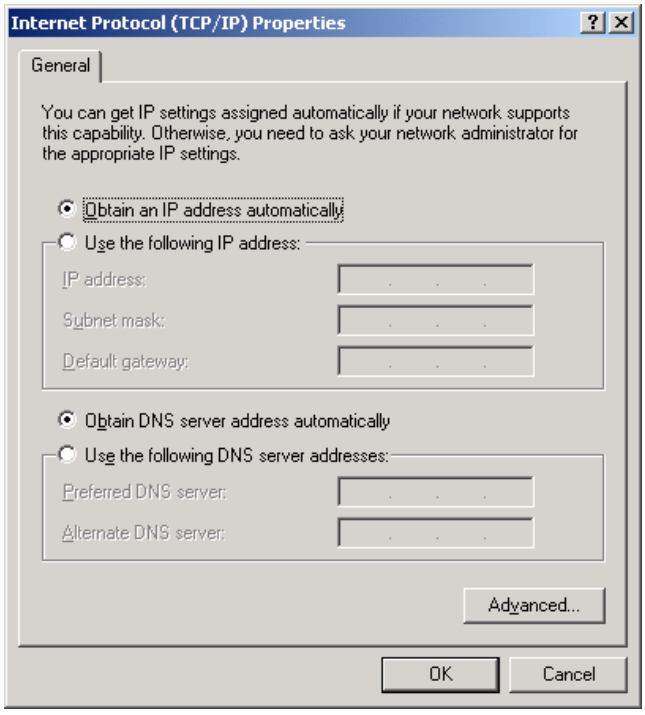
6

Setup of the LAN card

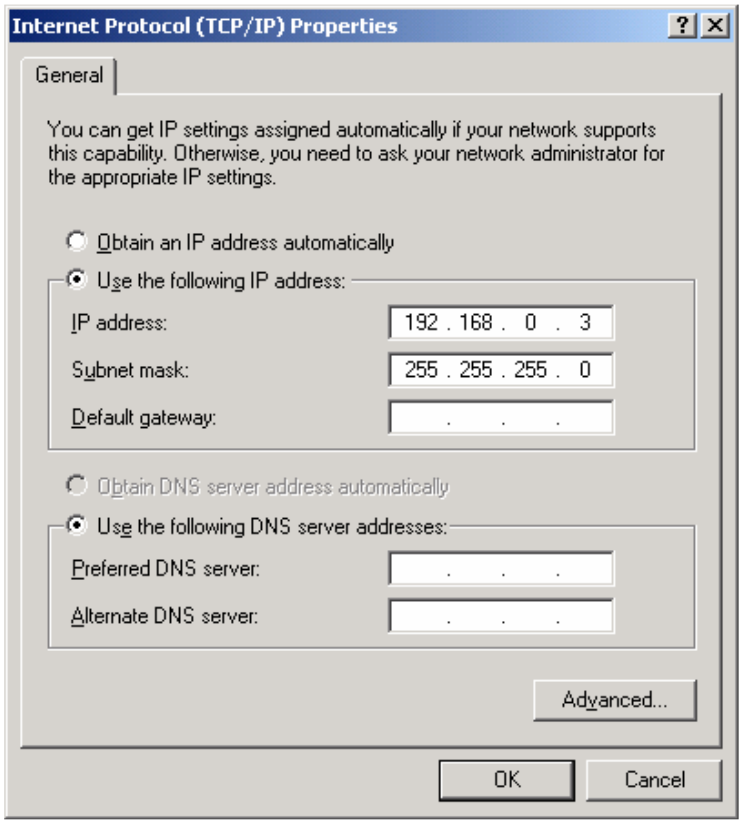
Select the properties of your LAN card. The following window will appear:



The network card name will be shown. Select Internet Protocol TCP/IP and click the Properties button. The following window will appear:



When you use a DHCP server to get the PC's IP address, you do not need to do anything. When you do not have a DHCP server, the screen will look like this :



The IP address might be blank. Fill in 192.168.0.3 for the IP address. And use subnet mask 255.255.255.0

All other settings can be left empty.

After you press ok, depending on the operating system, you might need to reboot.

After the optional reboot, create a DOS/Command box and enter :

IPCONFIG

This will displays the IP settings :

Microsoft Windows 2000 [Version 5.00.2195]

(C) Copyright 1985-2000 Microsoft Corp.

C:\>ipconfig

Windows 2000 IP Configuration

Ethernet adapter Local Area Connection 2:

Connection-specific DNS Suffix . :

IP Address. . . . . : **192.168.0.3**

Subnet Mask . . . . . : **255.255.255.0**

You can now try to PING your PC :

Enter from the DOS/command box : PING 192.168.0.3

C:\>**ping 192.168.0.3**

Pinging 192.168.0.3 with 32 bytes of data:

Reply from 192.168.0.3: bytes=32 time<10ms TTL=128

Reply from 192.168.0.3: bytes=32 time<10ms TTL=128

Reply from 192.168.0.3: bytes=32 time<10ms TTL=128

Reply from 192.168.0.3: bytes=32 time<10ms TTL=128

Ping statistics for 192.168.0.3:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms

The PING command will send data packets to the specified IP address. And the network card will reply by sending the data back. Ping is used to see if connections and servers are on line, and to measure the delay.

A DHCP server is a server that will assign IP addresses to clients. This is convenient because otherwise you will have to assign all IP addresses of the clients yourself manually.

When the PC boots, it will contact the DHCP server and will get an IP address assigned automatically. Depending on the settings of the DHCP server, the IP address can be the same (Static) or can change at every boot (Dynamic). When you use a fixed IP address, the IP address will be always the same. If you use the same IP address for multiple clients(PC's) a conflict will occur. That is also why a DHCP server is the preferred way to get an IP address. It will not give out duplicate IP addresses.

IP address 192.168.0.0 – 192.168.0.255 is a special address range. It is not used on the internet! Since it is not used on the net, this address range is often used within LAN's that are also connected to the internet. When you do not need to connect to the internet from your LAN(clients) you may use any IP address you like. But I advise that you use 192.168.0.xxx.

When address 192.168.0.3 was an assigned Internet address, you would contact a client/server on the internet instead of a PC/device on your LAN.

The next section will explain the special TCP/IP functions and statements that are available in BASCOM from version 1.11.7.2 and higher.

Chapter 7

TCP/IP Library

The BASCOM TCP/IP library allows you to use the W3100A internet chip from www.i2chip.com

The W3100A chip is placed on the IIM7000 module. You can also buy the W3100A chip and use it on your own design. Besides the IIM7000 module, there is also available the IIM7010 module which has an integrated RJ-45 connector as well. Consult the pdf files that are bundled with the Easy TCP/IP PCB.

The tcpip.lib contains assembler functions that are used by the following BASIC statements and functions :

CONFIG TCPIP , GETSOCKET , SOCKETCONNECT , SOCKETSTAT

TCPWRITE , TCPWRITESTR , TCPREAD

CLOSESOCKET , SOCKETLISTEN , GETDSTIP

GETDSTPORT , BASE64DEC

UDPWRITE , UDPWRITESTR , UDPREAD

The UDP functions work similar to the TCP/IP functions. UDP is a connection-less protocol. It means that you do not work in listen mode with UDP and that you don't connect to a server.

You just need to get a socket with GETSOCKET. After this you can send data or you can read data.

Normally you specify the IP address and PORT when you connect, but as you do not connect with UDP, you need to specify it when you send data. This is the main difference between the TCP/UDP- WRITE functions.

UDP is faster but also less reliable compared with TCP.

Since there are no servers and clients, you can only tell if you received data, by checking the received number of bytes .

When you send data, you will never know if it arrives, and if it arrives in the same order.

Sending 3 packets with UDP, does not mean that they arrive in the same order.

I advise to use TCP/IP when it is possible.

Important : Check the BASCOM Help file for the proper syntax.

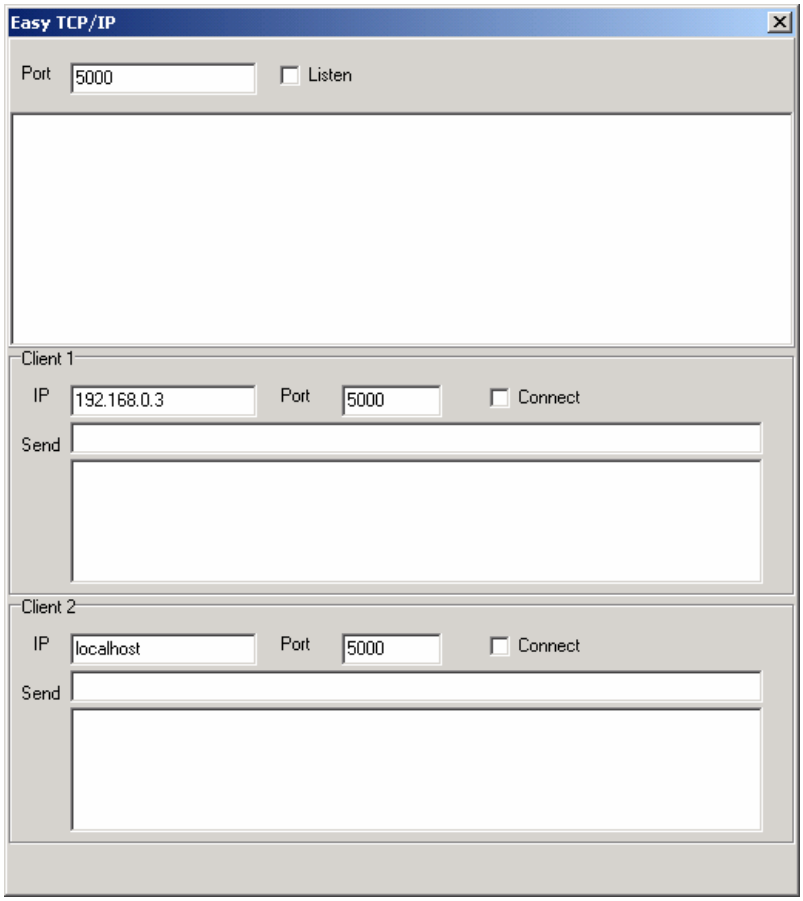
Chapter

8

Easy TCP/IP Windows Tool

The program easytcpip.exe can be used to test the various functions.

When you run easytcpip.exe the following screen will appear:

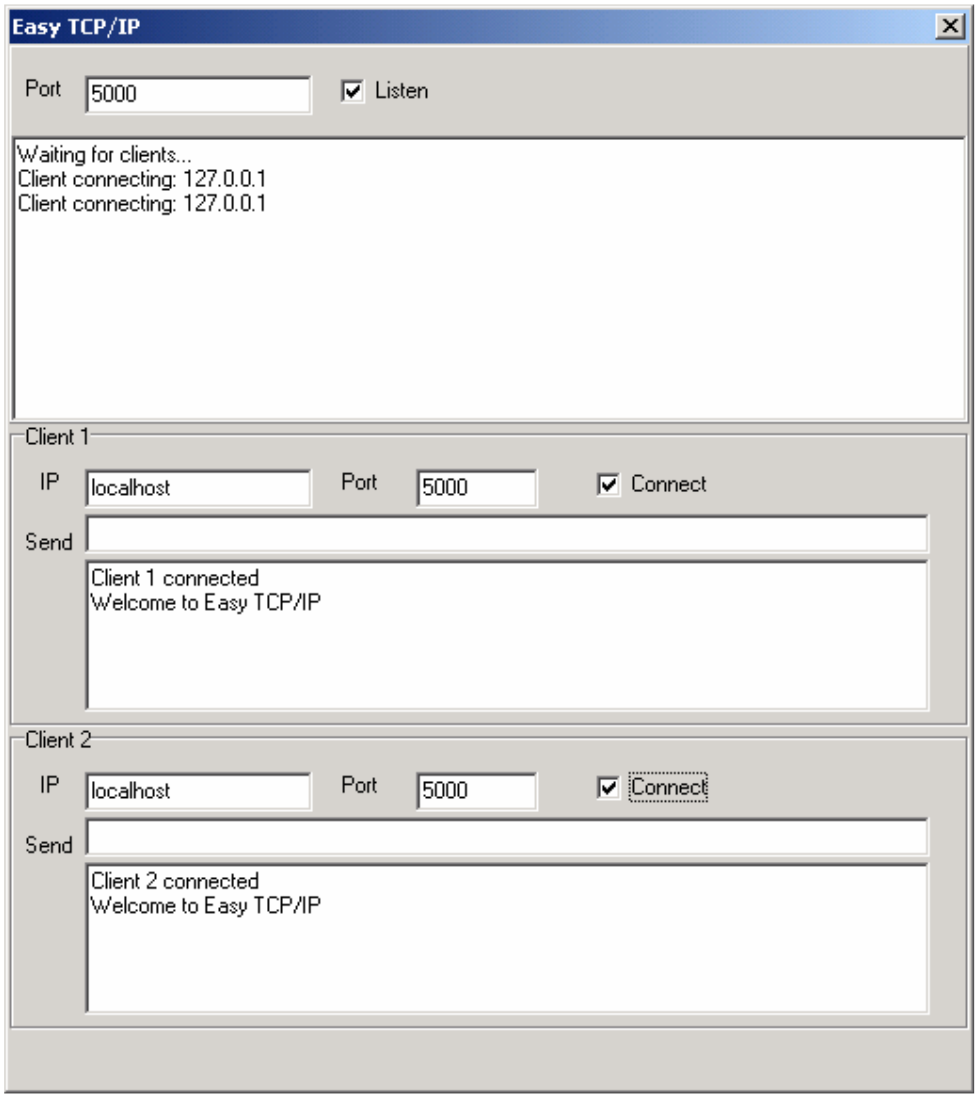


The program can act as a server and you can create two client connections. To test your network card perform the following test:

- Fill in 5000 for the server port number.
- Fill in 5000 for the client port numbers
- Click 'Listen'

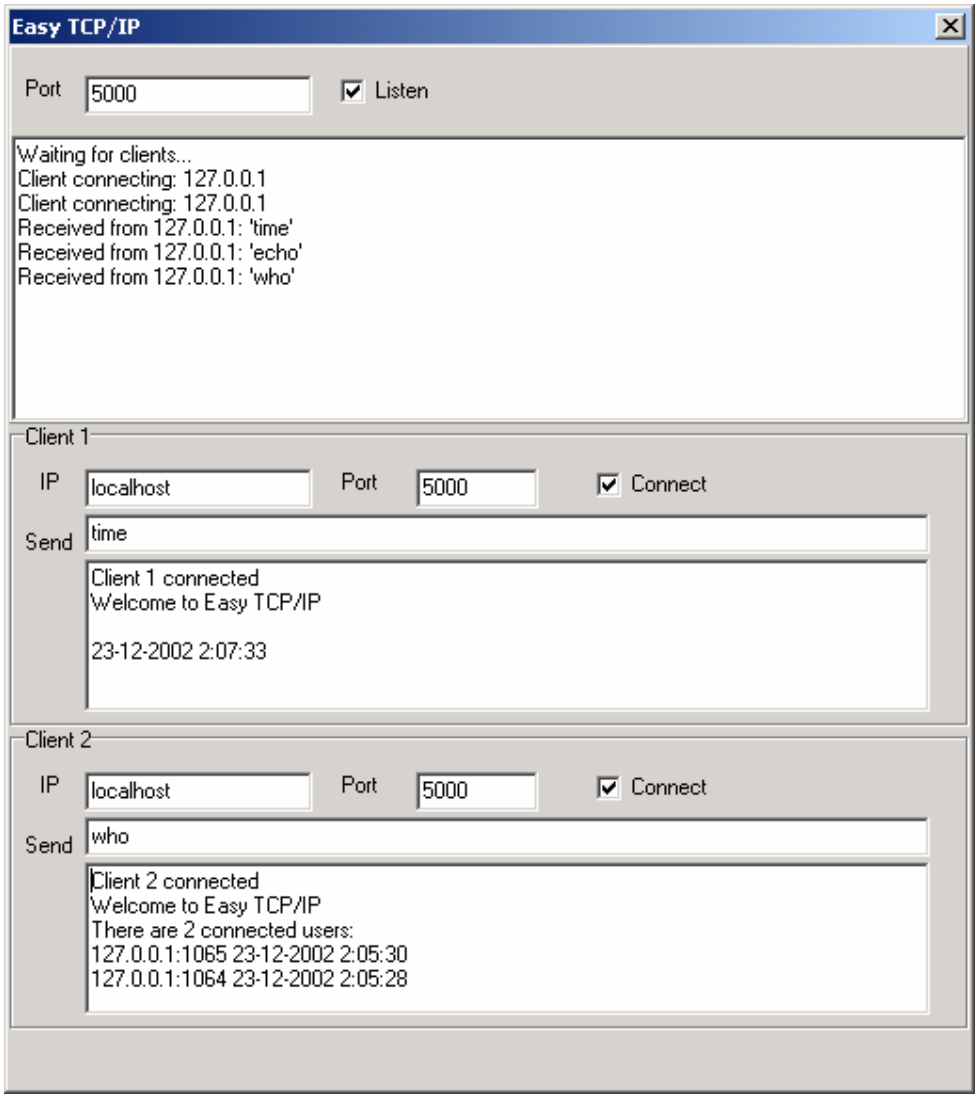
- Fill in localhost for the IP address under the Client1 section
- Fill in localhost for the IP address under the Client2 section
- Click 'Connect' of Client1
- Click 'Connect' of Client2

The screen will now look like this:



As soon as a client connects to the server, the server will send a message 'Welcome to Easy TCP/IP'

You can send a command to the server by typing TIME<ENTER> or WHO<ENTER>. The server will send the time or the connected clients:



By clicking 'Connect' again, the connection will be terminated.

You can also send EXIT to terminate the connection.

Localhost will use IP address 127.0.0.1 of the network card.

You could also specify the IP address of the network card : 192.168.0.3 (or the IP address you use).

When the above works, we are ready for some more tests.

Disconnect the clients, and leave the server in listen mode.

Open in BASCOM the clienttest.bas program.

Compile and program the chip. Open the terminal window to view the debug output.

The program will create 4 sockets and they all will connect to the server. When all is functioning, you must see the clients connecting.

In the terminal emulator you must see the welcome message from the server.

By pressing the <ESC> key, you wil be asked to enter a command.

Enter TIME or WHO and press <RETURN>. This command will be send to the server and you will notice that these commands are received. The server in turn, will send the known response that will be displayed in the terminal emulator.

When you are ready, press <ESC> again and enter EXIT. Now the client connections will be terminated. You will see this in the server program and in the terminal emulator.

After the connections are terminated, the Easy TCP/IP program need to be reset because there are no connections anymore.

Now we will test the server mode. Open the servertest.bas program, compile and program the chip. In the Easy TCP/IP program, fill in 192.168.0.8 for the IP address we want to connect to.

This is the address that we assigned to the W3100A.

Since the server is listening to port 5000, we fill in 5000 for the PORT too.

Now click 'Connect' . You will see a message send by the BASCOM server.

You can send some data back by entering data followed by a <RETURN>.

When you are ready, click 'Connect' again to disconnect the client from the server. By sending TIME, 12:00:00 will be returned by the server. By sending EXIT, the connection will be terminated.

When this works, create two connections and repeat the test.

I hope that by now you understand how easy it is.

There are some more examples provided:

pop3.bas that will read out the number of emails in your mailbox

smtp.bas that will send an email

webserver.bas that will display some simple web pages.

Dhcp2.bas. This is a DHCP client

The samples are intended as simple examples, more info you can find in the used RFC's. RFC's(Request For Comment) you can find on the internet.

The pop3 and smtp samples need a connection to the internet. For this option, the gateway address must be provided in the network card settings and on the CONFIG TCPIP line.

The webserver demo does not need an active Internet connection.

Chapter

9

Miscellaneous

Part number	Description	Misc
R2,R3,R4,R9	Resistor, 51	
R10	Resistor, 22	Optional for LCD
R8	Resistor, 10K	
R5,R6,R7	Resistor, 150	
R1, R11, R12, R13	Resistor 1K	
R14, R15	Resistor 390	
D1, D2,D3,D4	Diode, 1N4001	
D5, D6, D7	LED yellow, 3mm	
D8	LED green, 3mm	
D9	Zener diode 3V3, 400 mW	
U1	IC socket, 40 pin	
IC2	IC socket, 14 pin	
U3	IC socket, 16 pin	
IC3	IC socket, 20 pin	
IC1	ZSM560C, TO-92	Optional , Brown out circuit
U5	78S05	5V power regulator
U4	LM1117T-3.3 Or LD2517V33	3V3 power regulator
IC2	74HC00	
U1	90S8515 or M161	
IC3	74HC573	
U3	MAX232CPE	
C22, C23	Capacitor 22 pF	
C12	Capacitor 100 pF	
C1, C2, C3, C4, C7, C9, C10, C11, C13, C14, C20, C21, C24, C25, C26	Capacitor 100 nF	Decoupling capacitor
C15, C16, C17, C18 , C19	1uF/35V	
C5, C6	10 uF/35V	
C8	220 uF/35V	
J1	DB9, male connector, 90 degrees for PCB mounting	
J3	DB25, female connector, 90 degrees for PCB mounting	
	XTAL socket	
Q1	XTAL, 4 MHz	Use the freq. You need
J4	Power connector	
J2	RJ-45 magnetics	
U2	IIM7000A module	
T1	Reset switch, small	

Application ports

Application	Port/Protocol
tcpmux	1/tcp # TCP port service multiplexer
echo	7/tcp
echo	7/udp
discard	9/tcp sink null
discard	9/udp sink null
systat	11/tcp users
daytime	13/tcp
daytime	13/udp
netstat	15/tcp
qotd	17/tcp quote
msp	18/tcp # message send protocol
msp	18/udp # message send protocol
chargen	19/tcp ttytst source
chargen	19/udp ttytst source
ftp-data	20/tcp
ftp	21/tcp
fsp	21/udp fspd
ssh	22/tcp # SSH Remote Login Protocol
ssh	22/udp # SSH Remote Login Protocol
telnet	23/tcp # 24 - private
smtp	25/tcp
mail #	26 - unassigned
time	37/tcp time server
time	37/udp time server
rlp	39/udp resource # resource location
nameserver	42/tcp name # IEN 116
whois	43/tcp nicname
re-mail-ck	50/tcp # Remote Mail Checking Protocol
re-mail-ck	50/udp # Remote Mail Checking Protocol
domain	53/tcp nameserver # name-domain server
domain	53/udp nameserver
mtp	57/tcp # deprecated
bootps	67/tcp # BOOTP server
bootps	67/udp
bootpc	68/tcp # BOOTP client
bootpc	68/udp
tftp	69/udp
gopher	70/tcp # Internet Gopher
gopher	70/udp
rje	77/tcp netrjs
finger	79/tcp
www	80/tcp http # WorldWideWeb HTTP
www	80/udp # HyperText Transfer Protocol
link	87/tcp ttylink
kerberos	88/tcp kerberos5 krb5 # Kerberos v5
kerberos	88/udp kerberos5 krb5 # Kerberos v5
supdup	95/tcp
	# 100 - reserved
hostnames	101/tcp hostname # usually from sri-nic
iso-tsap	102/tcp tsap # part of ISODE.
csnet-ns	105/tcp cso-ns # also used by CSO name server
csnet-ns	105/udp cso-ns
#3com-tsmux	106/tcp poppassd
#3com-tsmux	106/udp poppassd
rtelnet	107/tcp # Remote Telnet
rtelnet	107/udp
pop-2	109/tcp postoffice # POP version 2
pop-2	109/udp
pop-3	110/tcp # POP version 3

pop-3	110/udp
sunrp	111/tcp portmapper # RPC 4.0 portmapper TCP
sunrpc	111/udp portmapper # RPC 4.0 portmapper UDP
auth	113/tcp authentication tap ident
sftp	115/tcp
uucp-path	117/tcp
nntp	119/tcp readnews untp # USENET News Transfer Protocol
ntp	123/tcp
ntp	123/udp # Network Time Protocol
netbios-ns	137/tcp # NETBIOS Name Service
netbios-ns	137/udp
netbios-dgm	138/tcp # NETBIOS Datagram Service
netbios-dgm	138/udp
netbios-ssn	139/tcp # NETBIOS session service
netbios-ssn	139/udp
imap2	143/tcp imap # Interim Mail Access Proto v2
imap2	143/udp imap
snmp	161/udp # Simple Net Mgmt Proto
snmp-trap	162/udp snmptrap # Traps for SNMP
cmip-man	163/tcp # ISO mgmt over IP (CMOT)
cmip-man	163/udp
cmip-agent	164/tcp
cmip-agent	164/udp
xdmcp	177/tcp # X Display Mgr. Control Proto
xdmcp	177/udp
nextstep	178/tcp NeXTStep NextStep # NeXTStep window
nextstep	178/udp NeXTStep NextStep # server
bgp	179/tcp # Border Gateway Proto.
bgp	179/udp
prospero	191/tcp # Cliff Neuman's Prospero
prospero	191/udp
irc	194/tcp # Internet Relay Chat
irc	194/udp
smux	199/tcp # SNMP Unix Multiplexer
smux	199/udp
at-rtmp	201/tcp # AppleTalk routing
at-rtmp	201/udp
at-nbp	202/tcp # AppleTalk name binding
at-nbp	202/udp
at-echo	204/tcp # AppleTalk echo
at-echo	204/udp
at-zis	206/tcp # AppleTalk zone information
at-zis	206/udp
qmtp	209/tcp # The Quick Mail Transfer Protocol
qmtp	209/udp # The Quick Mail Transfer Protocol
z3950	210/tcp wais # NISO Z39.50 database
z3950	210/udp wais
ipx	213/tcp # IPX
ipx	213/udp
imap3	220/tcp # Interactive Mail Access
imap3	220/udp # Protocol v3
rpc2portmap	369/tcp
rpc2portmap	369/udp # Coda portmapper
codaauth2	370/tcp
codaauth2	370/udp # Coda authentication server
ulistserv	372/tcp # UNIX Listserv
ulistserv	372/udp
https	443/tcp # MCom
https	443/udp # MCom
snpp	444/tcp # Simple Network Paging Protocol
snpp	444/udp # Simple Network Paging Protocol
saft	487/tcp # Simple Asynchronous File Transfer
saft	487/udp # Simple Asynchronous File Transfer
npmp-local	610/tcp dqs313_qmaster # npmp-local / DQS
npmp-local	610/udp dqs313_qmaster # npmp-local / DQS
npmp-gui	611/tcp dqs313_execd # npmp-gui / DQS

npmp-gui	611/udp dqs313_execd # npmp-gui / DQS
hmmp-ind	612/tcp dqs313_intercell # HMMP Indication / DQS
hmmp-ind	612/udp dqs313_intercell # HMMP Indication / DQS
# UNIX specific services	
exec	512/tcp
biff	512/udp comsat
login	513/tcp
who	513/udp whod
shell	514/tcp cmd # no passwords used
syslog	514/udp
printer	515/tcp spooler # line printer spooler
talk	517/udp
ntalk	518/udp
route	520/udp router routed # RIP
timed	525/udp timeserver
tempo	526/tcp newdate
courier	530/tcp rpc
conference	531/tcp chat
netnews	532/tcp readnews
netwall	533/udp # -for emergency broadcasts
uucp	540/tcp uucpd # uucp daemon
afpovertcp	548/tcp # AFP over TCP
afpovertcp	548/udp # AFP over TCP
remotefs	556/tcp rfs_server rfs # Brunhoff remote filesystem
klogin	543/tcp # Kerberized `rlogin' (v5)
kshell	544/tcp krcmd # Kerberized `rsh' (v5)
kerberos-adm	749/tcp # Kerberos `kadmin' (v5)
webster	765/tcp # Network dictionary
webster	765/udp
ingreslock	1524/tcp
ingreslock	1524/udp
prospero-np	1525/tcp # Prospero non-privileged
prospero-np	1525/udp
datametrics	1645/tcp old-radius # datametrics / old radius entry
datametrics	1645/udp old-radius # datametrics / old radius entry
sa-msg-port	1646/tcp old-radacct # sa-msg-port / old radacct entry
sa-msg-port	1646/udp old-radacct # sa-msg-port / old radacct entry
radius	1812/tcp # Radius
radius	1812/udp # Radius
radacct	1813/tcp # Radius Accounting
radacct	1813/udp # Radius Accounting
cvspserver	2401/tcp # CVS client/server operations
cvspserver	2401/udp # CVS client/server operations
venus	2430/tcp # codacon port
venus	2430/udp # Venus callback/wbc interface
venus-se	2431/tcp # tcp side effects
venus-se	2431/udp # udp sftp side effect
codasrv	2432/tcp # not used
codasrv	2432/udp # server port
codasrv-se	2433/tcp # tcp side effects
codasrv-se	2433/udp # udp sftp side effect
mysql	3306/tcp # MySQL
mysql	3306/udp # MySQL
rfe	5002/tcp # Radio Free Ethernet
rfe	5002/udp # Actually uses UDP only
cfengine	5308/tcp # CFengine
cfengine	5308/udp # CFengine
bbs	7000/tcp # BBS service
kerberos4	750/udp kerberos-iv kdc # Kerberos (server) udp

kerberos4	750/tcp kerberos-iv kdc # Kerberos (server) tcp
kerberos_master	751/udp # Kerberos authentication
kerberos_master	751/tcp # Kerberos authentication
passwd_server	752/udp # Kerberos passwd server
krb_prop	754/tcp # Kerberos slave propagation
krbupdate	760/tcp kreg # Kerberos registration
kpasswd	761/tcp kpwd # Kerberos "passwd"
kpop	1109/tcp # Pop with Kerberos
knetd	2053/tcp # Kerberos de-multiplexor
zephyr-srv	2102/udp # Zephyr server
zephyr-clt	2103/udp # Zephyr serv-hm connection
zephyr-hm	2104/udp # Zephyr hostmanager
eklogin	2105/tcp # Kerberos encrypted rlogin
# Unofficial but necessary (for NetBSD) services	
supfilesrv	871/tcp # SUP server
supfiledbg	1127/tcp # SUP debugging
# Datagram Delivery Protocol services	
rtmp	1/ddp # Routing Table Maintenance Protocol
nbp	2/ddp # Name Binding Protocol
echo	4/ddp # AppleTalk Echo Protocol
zip	6/ddp # Zone Information Protocol
# Services added for the Debian GNU/Linux distribution	
poppassd	106/tcp # Eudora
poppassd	106/udp # Eudora
mailq	174/tcp # Mailer transport queue for Zmailer
mailq	174/tcp # Mailer transport queue for Zmailer
ssmtp	465/tcp # SMTP over SSL
gdomap	538/tcp # GNUstep distributed objects
gdomap	538/udp # GNUstep distributed objects
snews	563/tcp # NNTP over SSL
ssl-ldap	636/tcp # LDAP over SSL
omirr	808/tcp omirrd # online mirror
omirr	808/udp omirrd # online mirror
rsync	873/tcp # rsync
rsync	873/udp # rsync
simap	993/tcp # IMAP over SSL
spop3	995/tcp # POP-3 over SSL
socks	1080/tcp # socks proxy server
socks	1080/udp # socks proxy server
rmtcfg	1236/tcp # Gracilis Packeten remote config server
xtel	1313/tcp # french minitel
support	1529/tcp # GNATS
cfinger	2003/tcp # GNU Finger
ninstall	2150/tcp # ninstall service
ninstall	2150/udp # ninstall service
afbackup	2988/tcp # Afbackup system
afbackup	2988/udp # Afbackup system
icp	3130/tcp # Internet Cache Protocol (Squid)
icp	3130/udp # Internet Cache Protocol (Squid)
postgres	5432/tcp # POSTGRES
postgres	5432/udp # POSTGRES
fax	4557/tcp # FAX transmission service (old)
hylafax	4559/tcp # HylaFAX client-server protocol (new)
noclog	5354/tcp # noclogd with TCP (nocol)
noclog	5354/udp # noclogd with UDP (nocol)
hostmon	5355/tcp # hostmon uses TCP (nocol)
hostmon	5355/udp # hostmon uses TCP (nocol)
pcANYWHERE32	5631/tcp
pcANYWHERE32	5631/tcp

ircd	6667/tcp # Internet Relay Chat
ircd	6667/udp # Internet Relay Chat
webcache	8080/tcp # WWW caching service
webcache	8080/udp # WWW caching service
tproxy	8081/tcp # Transparent Proxy
tproxy	8081/udp # Transparent Proxy
mandelspawn	9359/udp mandelbrot # network mandelbrot
amanda	10080/udp # amanda backup services
kamanda	10081/tcp # amanda backup services (Kerberos)
kamanda	10081/udp # amanda backup services (Kerberos)
amandaidx	10082/tcp # amanda backup services
amidxtape	10083/tcp # amanda backup services
isdnlog	20011/tcp # isdn logging system
isdnlog	20011/udp # isdn logging system
vboxd	20012/tcp # voice box system
vboxd	20012/udp # voice box system
binkp	24554/tcp # Binkley
binkp	24554/udp # Binkley
asp	27374/tcp # Address Search Protocol
asp	27374/udp # Address Search Protocol
tfido	60177/tcp # Ifmail
tfido	60177/udp # Ifmail
fido	60179/tcp # Ifmail
fido	60179/udp # Ifmail

How the board works

The wall transformer is plugged into J4. A diode bridge will create DC which is filtered with capacitor C8. The DC voltage is regulated with a 3.3Volt and 5V regulator.

The IIM7000A contains the W3100A and Ethernet PHY chip. The W3100A is used in bus mode. For this reason we use a micro processor that is capable of addressing external memory. In the case of the AVR we use the Mega162L. The microprocessor is serial programmed via the LPT port with the Sample Electronics programmer. The MAX232 chip is used for the serial interface.

The 74HC573 latches the micro's lower address lines (a0-a7) that are available on the data lines. The W3100A chip is mapped into the &h8000-&hFFFF memory range. Most of this space is used for the internal transmit and receive buffers. The remaining part is used by the W3100A control and status registers.

A15 is inverted to create the /CS(chip select) signal.

Jumper J5 can provide either an active low or active high reset, so that both an AVR and 8051 microprocessors can be used. (a 8051 library is not available yet)

The IIM 7000A module drives the link, 10MB and 100 MB leds.

The board also has a proto type area. Connector J6 could be used to insert your own PCB.

Connector J6 has the following pin out

	PIN	PIN	
+5V	1	2	PORTB.0
PORTB.1	3	4	PORTB.2
PORTB.3	5	6	PORTB.4
PORTB.5	7	8	PORTB.6
PORTB.7	9	10	GND
+3.3V	11	12	PORTD.0
PORTD.1	13	14	PORTD.2
PORTD.3	15	16	PORTD.4
PORTD.5	17	18	PORTD.6
PORTD.7	19	20	GND

Miscellaneous TCP/IP programs

There are a number of convenient programs available that are installed by Windows.

NETSTAT

This will list all active connections of your PC.

NBTSTAT

Displays protocol statistics and current TCP/IP connections using NBT(Netbios over TCP/IP)

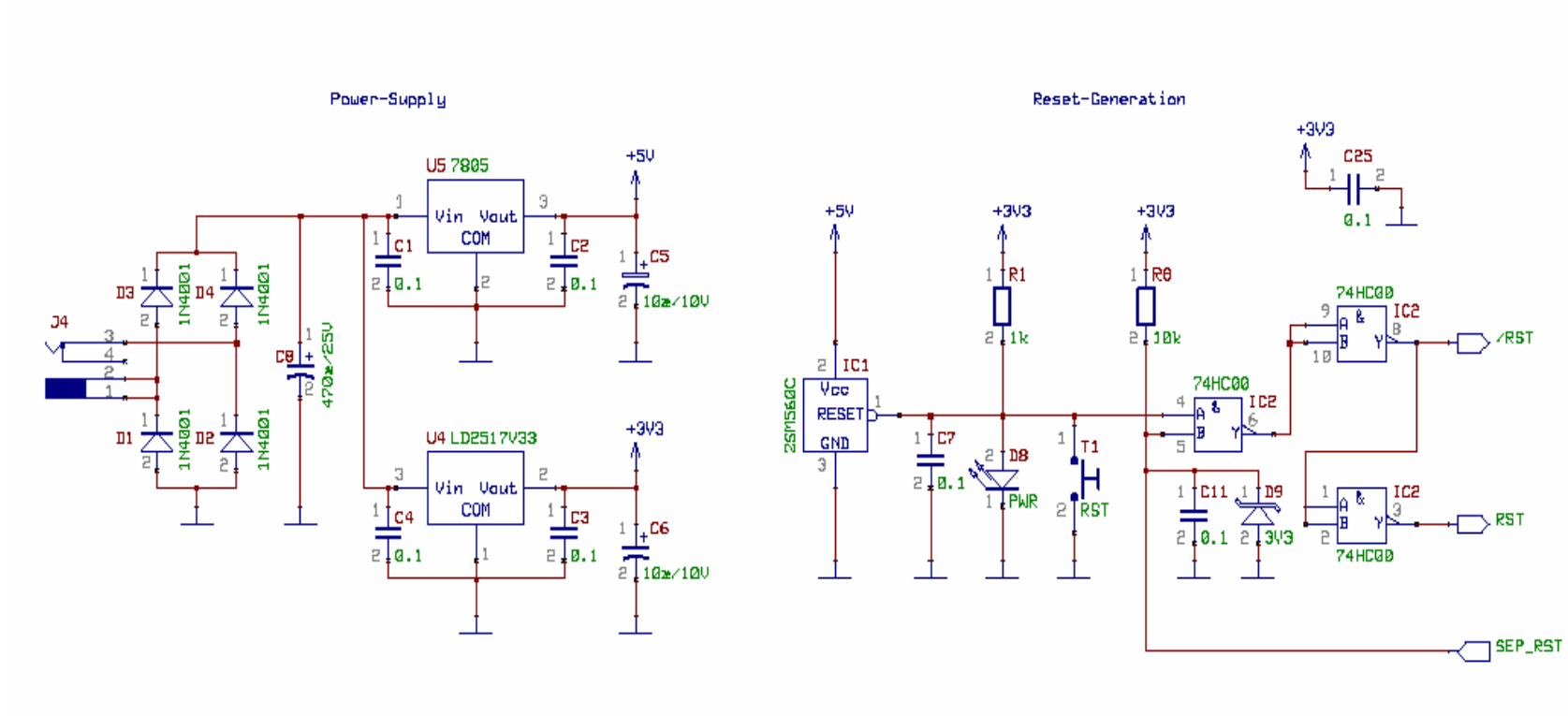
PING

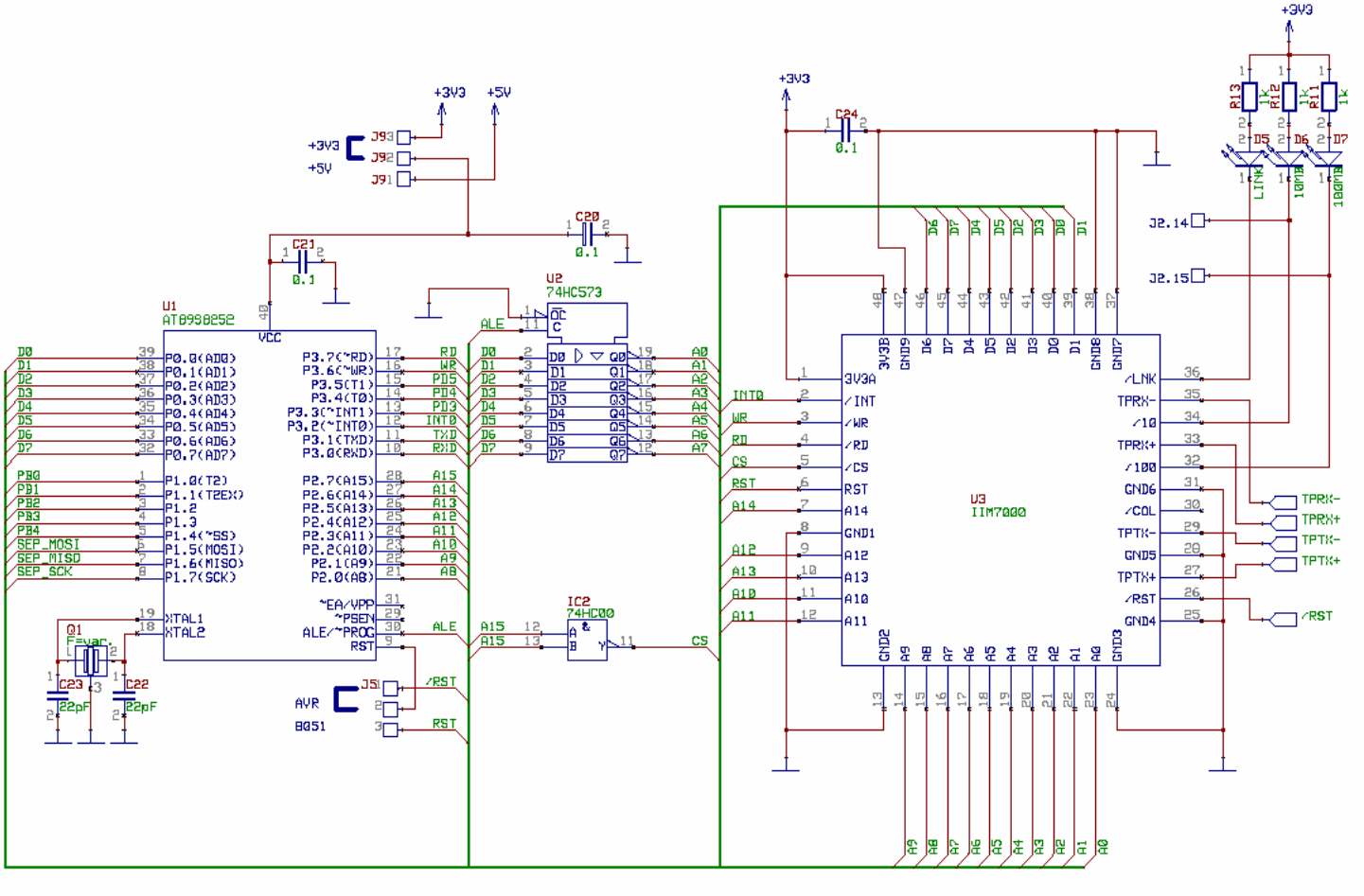
Will send data to a host or IP number.

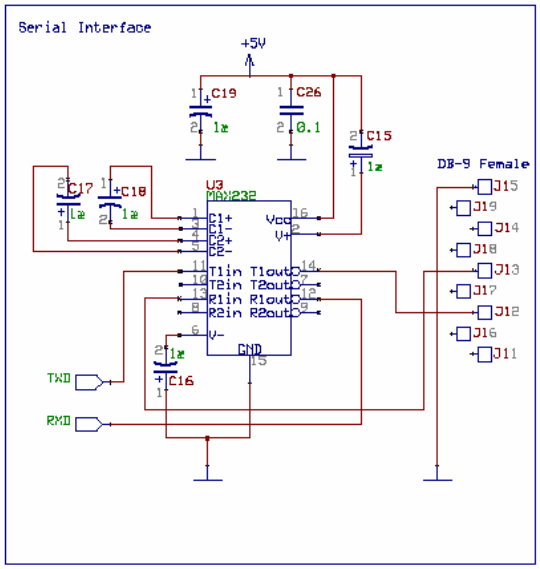
IPCONFIG

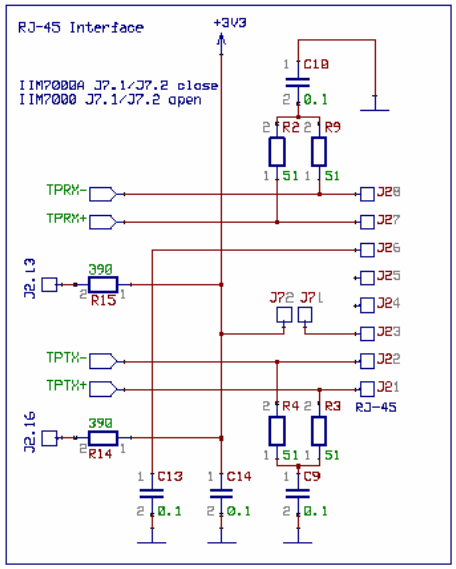
Displays Ip Configuration.

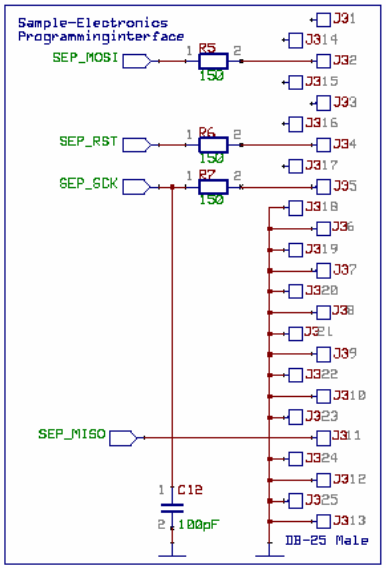
I advise you to take a look at these commands. Most of them need parameters that are explained at the command/DOS prompt.

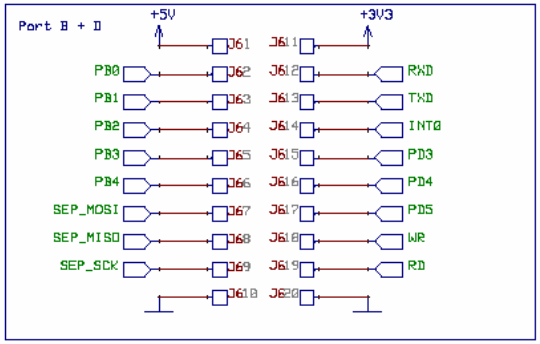

Power-Supply
U5 7805
Vin Vout
COM
+5V
C1
0.1
C2
0.1
C5
10u/10V
D3
D4
D1
D2
1N4001
J4
C8
470u/25V
U4 LD2517V33
+3V3
C4
0.1
C3
0.1
C6
10u/10V
Reset-Generation
+5V
+3V3
+3V3
+3V3
C25
0.1
R1
1k
R0
10k
74HC00
IC2
/RST
IC1
Vcc
RESET
GND
2SM568C
C7
0.1
D8
PWR
T1
RST
C11
0.1
D9
3V3
RST
SEP_RST

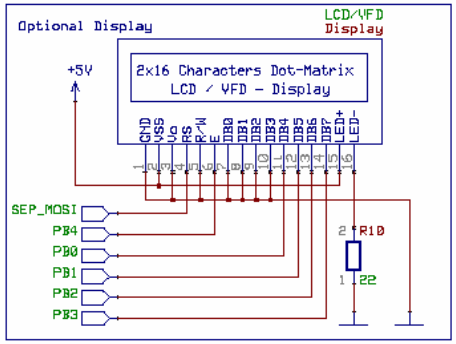

+3V3
+5V
J93
J92
J91
C20
0.1
C21
0.1
U1
AT89S8252
VCC
P0.0(AD0)
P0.1(AD1)
P0.2(AD2)
P0.3(AD3)
P0.4(AD4)
P0.5(AD5)
P0.6(AD6)
P0.7(AD7)
P1.0(T2)
P1.1(T2EX)
P1.2
P1.3
P1.4(~SS)
P1.5(MOSI)
P1.6(MISO)
P1.7(SCK)
P3.7(~RD)
P3.6(~WR)
P3.5(T1)
P3.4(T0)
P3.3(~INT1)
P3.2(~INT0)
P3.1(TXD)
P3.0(RXD)
P2.7(A15)
P2.6(A14)
P2.5(A13)
P2.4(A12)
P2.3(A11)
P2.2(A10)
P2.1(A9)
P2.0(A8)
~EA/VPP
~PSEN
ALE/~PROG
RST
XTAL1
XTAL2
PB0
PB1
PB2
PB3
PB4
SEP_MOSI
SEP_MISO
SEP_SCK
Q1
Fevac
C23
22pF
C22
22pF
J51
AVR
8051
/RST
RST
RD
WR
PB5
PB6
PB7
INT0
TXD
RXD
A15
A14
A13
A12
A11
A10
A9
A8
ALE
U2
74HC573
OE
C
D0
D1
D2
D3
D4
D5
D6
D7
Q0
Q1
Q2
Q3
Q4
Q5
Q6
Q7
A0
A1
A2
A3
A4
A5
A6
A7
IC2
74HC00
A
B
Y
CS
+3V3
C24
0.1
R13
1k
R12
1k
R11
1k
D5
D6
D7
LINK
10M
100M
J2.14
J2.15
U3
IIM7000
3V3A
/INT
/WR
/RD
/CS
RST
A14
GND1
A12
A13
A10
A11
GND2
A9
A8
A7
A6
A5
A4
A3
A2
A1
A0
GND3
GND4
/RST
GND5
TPTX+
TPTX-
GND6
/COL
/100
TPRX+
/10
TPRX-
/LNK
GND7
GND8
D1
D0
D3
D2
D5
D4
D7
D6
GND9
3V3B
INTR
TPRX-
TPRX+
TPTX-
TPTX+
/RST

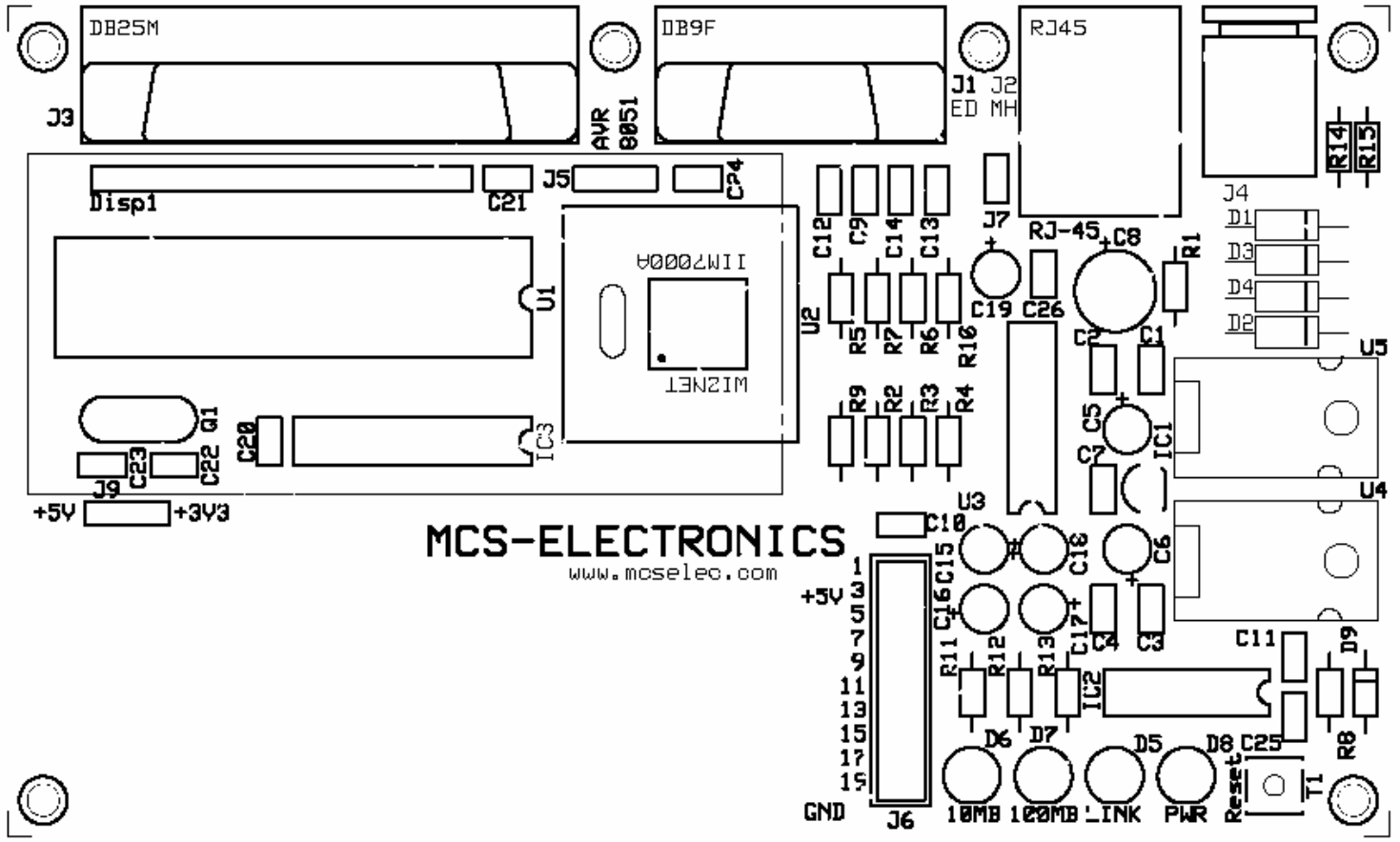

Serial Interface
+5V
C19
1u
C26
0.1
C15
1u
DB-9 Female
C17
1u
C18
1u
U3
MAX232
C1+
C1-
C2+
C2-
Vcc
V+
V-
T1in T1out
T2in T2out
R1in R1out
R2in R2out
GND
J15
J19
J14
J18
J13
J17
J12
J16
J11
C16
1u
TXD
RXD


RJ-45 Interface
+3V3
IIM7000A J7.1/J7.2 close
IIM7000 J7.1/J7.2 open
C10
0.1
R2
R9
51
51
TPRX-
TPRX+
TPTX-
TPTX+
J2.13
J2.16
390
R15
390
R14
J7.2
J7.1
R4
R3
51
51
RJ-45
C13
0.1
C14
0.1
C9
0.1
J28
J27
J26
J25
J24
J23
J22
J21


Sample-Electronics
Programminginterface
SEP_MOSI
SEP_RST
SEP_SCK
SEP_MISO
R5
150
R6
150
R7
150
C12
100pF
DB-25 Male


Port B + D
+5V
+3V3
PB0
PB1
PB2
PB3
PB4
SEP_MOSI
SEP_MISO
SEP_SCK
RXD
TXD
INT0
PD3
PD4
PD5
WR
RD


Optional Display
LCD/VFD
Display
2x16 Characters Dot-Matrix
LCD / VFD - Display
+5V
GND
VSS
Vo
RS
R/W
E
DB0
DB1
DB2
DB3
DB4
DB5
DB6
DB7
LED+
LED-
SEP_MOSI
PB4
PB0
PB1
PB2
PB3
R10
22


DB25M
DB9F
RJ45
J3
J1 J2
ED MH
AVR
8051
Disp1
C21
J5
C24
J4
D1
D3
D4
D2
RJ-45
C8
R1
R14
R15
U1
U2
C12
C9
C14
C13
J7
C19
C26
R5
R7
R6
R16
C2
C1
U5
R9
R2
R3
R4
C5
C7
IC1
U3
U4
C10
C15
C18
C6
C16
C17
C4
C3
C11
D9
R11
R12
R13
IC2
R8
D6
D7
D5
D8
C25
T1
Reset
J6
10MB
100MB
LINK
PWR
GND
+5V
+3V3
J9
C23
C22
Q1
C20
IC3
MCS-ELECTRONICS
www.mcselec.com

Index