

ETH044 - 4 SSR (230VAC) and 4 Digital IO

Technical Documentation



WARNING

Mains Voltages can be Lethal.

If you are not confident with using these voltages, please use a qualified electrician to wire this up for you.

Overview

The ETH044 provides four solid state relay outputs at 230VAC with a current rating of up to 5Amps per channel and up to 10Amps total per module, 4 digital IO channels are also provided for flexible additional functionality. The ETH044 module is powered directly from a 230VAC mains supply at either 50hz or 60hz, the live, earth and neutral outputs for each channel are brought out to screw terminals, while the live is the only output that is switched the neutral and earth are provided for easy wiring. Each channel uses phase angle controlled switching for dimmable lighting.

A feature is also implemented (input mapping) that allows the digital inputs to remotely control relays or digital outputs on other ETHxxx(x) series modules such as the ETH008 etc. This offers the opportunity to construct a system where an input can control an output anywhere on the earth provided both locations are connected to the network/internet.

Operating Temperature

-30C to +70C

LED Indication

The ETH044 provides a red LED mounted immediately next to each SSR and the intensity will increase with power output, each digital IO has a red LED to reflect the channel status, there is also two LED's mounted in the Ethernet connector which will flash with Ethernet traffic. Finally there is a green power LED just above the processor.

So what is a SSR (Solid State Relay)?

A Solid State Relay is a purely electronic switch, in our application it is switching a live feed out of the PCB. There is no mechanical element to the operation meaning it can provide benefits such as:

Operate at higher frequencies - there is no time for the relay arm to mechanically switch

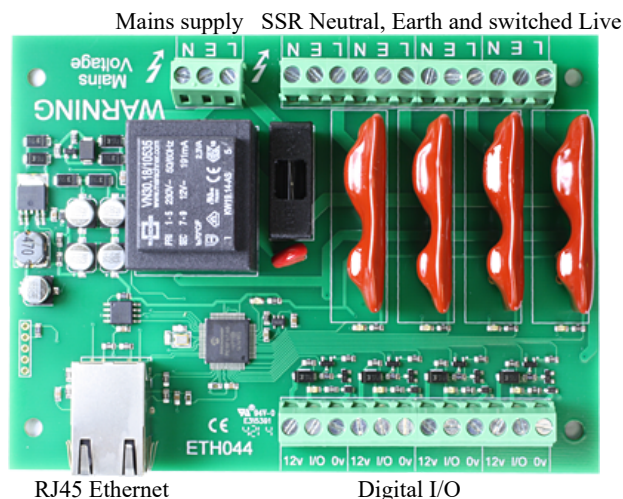
Longer life than a mechanical relay - as the Solid State Relays is purely electronic there is potentially far longer life

Ability to handle high current switch on devices - Loads that draw high current at switch on like a halogen bulb will cause an arc between mechanical relay contacts, this will occur as they become in close proximity just before contact. Solid State Relays are unaffected by these types of loads.

SSR switching capabilities

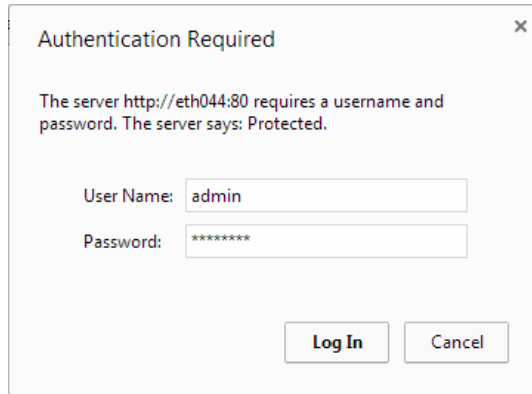
A [datasheet](#) for the SSR provides a full description of maximum switching capabilities.

Connections



First Test

Having plugged in your 12vdc power supply and Ethernet connection, start up your web browser and type **http://eth044** into the address bar, please note this only works in windows. You will be prompted for a password as shown below:



The default login is:
Username: admin
Password: password

The ability to change these settings is shown in the configuration section

You should now see the following web page:

ETH044 Test Application

Below you will see the current status of the ETH044 board, Solid state relay 1 and input/output1 are on the right hand side. Click the bullets to toggle the relays or turn change the output states on the board.

The status is updated in real time. You will see this page automatically update when changing states with commands over TCP/IP.

Firmware V1 Module Id 29
Hardware V1

[Technical documentation](#)

[Configuration](#)

[Mapped inputs](#)

SSR 4 - 0% +	SSR 3 - 0% +	SSR 2 - 0% +	SSR 1 - 0% +
Inputs ● ● ● ●		Outputs (click to toggle) ● ● ● ●	

Copyright © 2014 Devantech Ltd.

This web page will allow you to vary the SSR power outputs by clicking the - and + buttons, there is also the ability to make digital outputs active (drive low) and an input status. The web page also contains a link to this technical documentation page.

Configuration

By clicking the configuration link it's possible to configure the ETH044 IP address and subnet mask together with the ability to set a password for entry to control screens. Gateway address and DNS address is configurable and is used with mapped inputs which are described in section below. The configuration page also offers the option to set a password that will be required to change any of the relay states or digital outputs using TCP IP commands, this is explained in the TCP/IP password section.

All settings are saved to memory so be careful to remember the username and password! Default password settings are shown in the picture below.

Board Configuration

This section allows the configuration of the board's network settings.

CAUTION: Incorrect settings may cause the board to lose network connectivity.

MAC Address	00:1E:C0:94:11:4C
Host Name	ETH044
<input checked="" type="checkbox"/> HTTP authentication	
Username:	admin
Password:	password
Port	17494
<input checked="" type="checkbox"/> Enable DHCP	
IP Address	192.168.0.79
Subnet Mask	255.255.255.0
Gateway address	192.168.0.1
DNS address	192.168.0.1
<input type="checkbox"/> TCP/IP Password	password
<input type="checkbox"/> Latched outputs	
<input type="button" value="Save Config"/>	

Mapped inputs

Digital inputs are able to be configured to remotely control outputs on an ETH002, ETH008,ETH484 or ETH8020, this offers simple linking and versatile usage. An input in one country can control a output in another, or across a small network.

There are eight independent inputs that can be mapped to eight different relays (on the same or different boards).

The "Address of target board" field accepts an IP address or hostname (which will be converted to an IP address by the DNS server supplied in the board configuration above). If the module is on the local network then you can use the assigned IP address, if the target is over the internet then you need to supply the gateway in the configuration (internet source IP like your router) and the "Address of target board" is the IP address of the targets internet connection (to point at the router). Accessing the target via a router is dealt with in the section "**Access from the Internet**".

Input mapping configuration

You can map an input to switch a relay over on another board using the configuration below.

<input type="checkbox"/> Enable Input1	
Address of target board	192.168.0.1
Port	17494
Output number	1
TCP Password: (if required)	password
<input type="checkbox"/> Enable Input2	
Address of target board	192.168.0.1
Port	17494
Output number	2
TCP Password: (if required)	password
<input type="checkbox"/> Enable Input3	
Address of target board	192.168.0.1
Port	17494
Output number	3
TCP Password: (if required)	password
<input type="checkbox"/> Enable Input4	
Address of target board	192.168.0.1
Port	17494
Output number	4
TCP Password: (if required)	password
<input type="button" value="Save Config"/>	

Mapping inputs to custom devices

Following customer requests for obtaining input states without the need for polling the ETH044, this can be achieved with the existing input mapping function.

If you would like the inputs to be mapped to a custom device then we have a simple command structure to achieve this, the ETH044 will send the commands in blue, your device will respond with commands in yellow.

A TCP packet with 0x79 (password entry) in the first byte, then the following bytes will be the password supplied above

To acknowledge a password match, respond with 1, else send 2

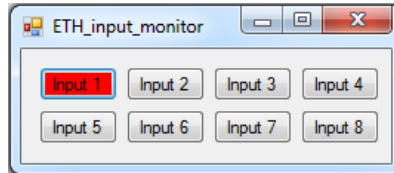
Digital active (0x20) or Digital inactive (0x21) followed by the output number

Reply with a 0 for success, else send 1

Note that the complete sequence must be followed, even if the password fails.

Input monitoring example and source code

We have an example of mapping the inputs to a PC, it operates on the default port of 17494, the default password of "password" (although both are easily changed in the source code) and requires the ETH044 input mapping to be pointed at the host PC IP address. As it's having to constantly listen it's a multi thread program and monitors up to 8 inputs.



The input monitor program is available as Visual C# express ready built installation files [here](#), or as Visual C# express project with source files [here](#). Visual studio express is provided free from Microsoft: [Visual Express](#)

Factory Reset

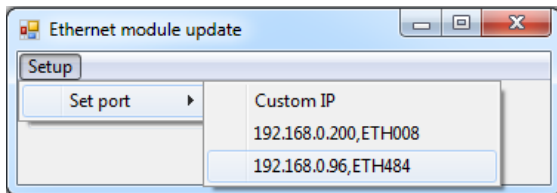
Should it be necessary to reset the ETH044 to its shipped condition then the end two contacts of the row of 5 holes nearest the Ethernet connection must be shorted together at board power up. The green LED should then flash as the settings are reset, please wait until the LED finishes flashing and do not remove power during this period.

Firmware Updates

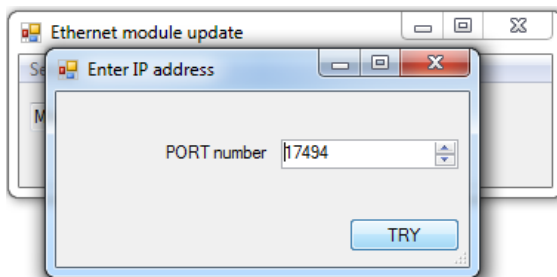
The firmware is fully updateable by re-flashing the board using our custom windows program. The updates currently available are: [V2 - 03/04/17](#) - Mapped inputs socket closure correction.

Using the [Ethernet module update](#):

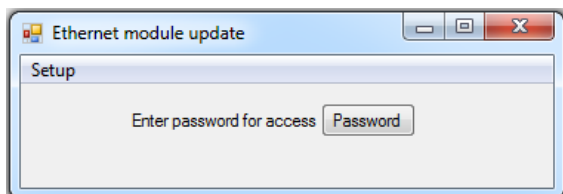
First you need to connect to the board, the program will scan the local network and provide any boards in a list that it can see.



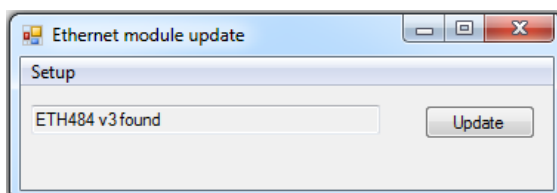
Next you will need to specify what port the board is located on, this will be 17494 unless you have changed it in the settings.

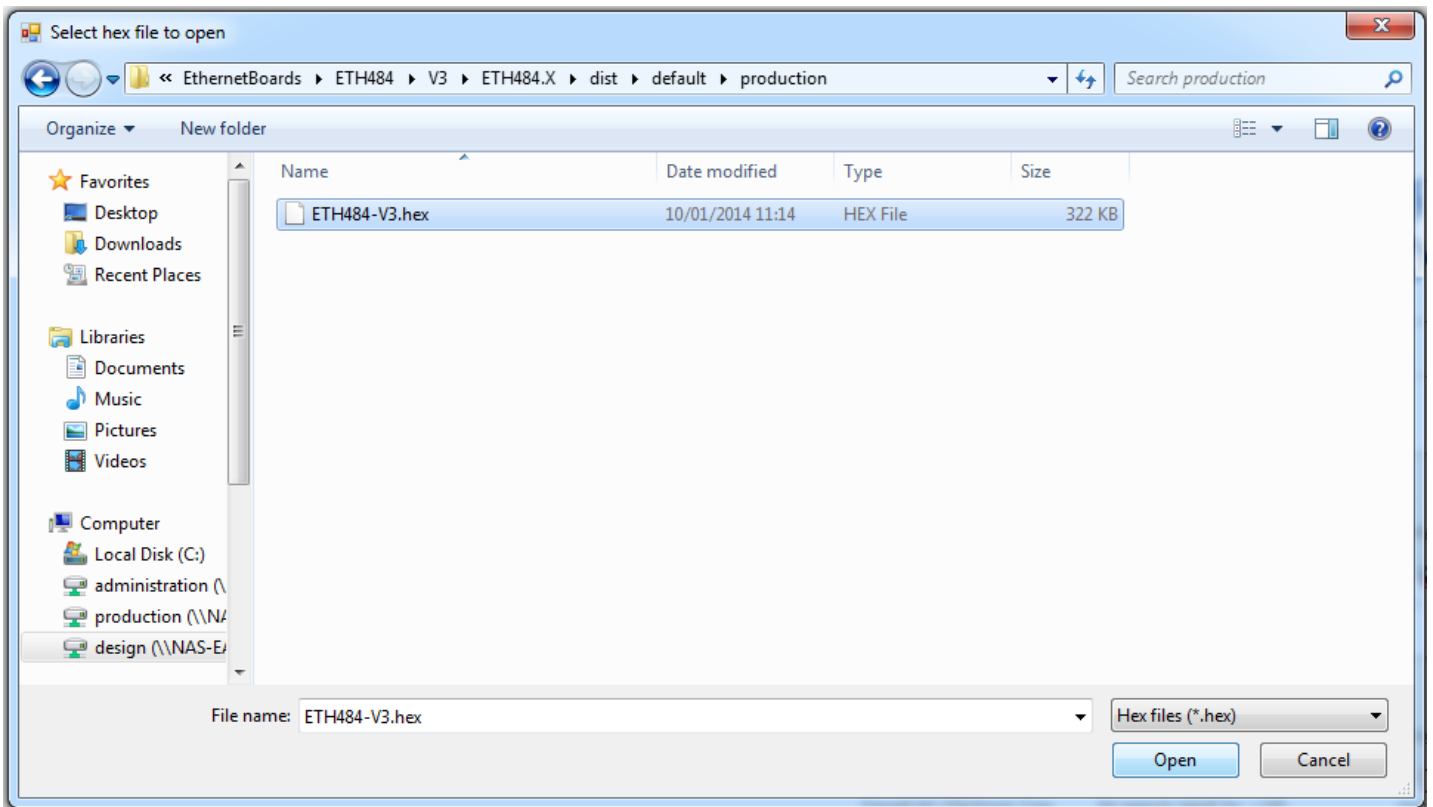


If there is a TCP/IP password the program will require it to be entered by clicking the password button.

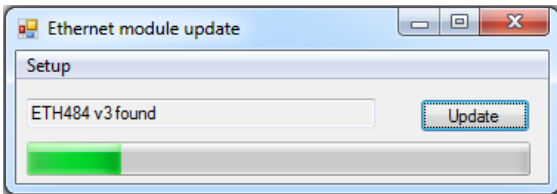


Following a successful connection you will now be able to click the update button to select a file





Once the file has been opened the programming sequence will now begin, it's fairly short and following it's completion the board will reset (automatically) and the LED will flash quickly while the programming sequence completes. Do not turn the power off in this sequence.



ETH044 Command Set

The command set designed to provide consistent expansion and new features, they are sent over TCP/IP on port 17494 (0x4456). This is the default port, it can be changed in the configuration settings.

Five connections are allowed at any one time, these are independently protected but all using the same password as defined in the board configuration.

Command		Action
dec	hex	
16	10	Get Module Info - returns 3 bytes. Module Id (29 for ETH044), Hardware version, Firmware version.
32	20	Digital Active - follow with 1-4 to set SSR power output to 100%, or 9-12 for digital I/O then a time for pulsed output from 1-255 (100ms resolution) or 0 for permanent Board will return 0 for success, 1 for failure
33	21	Digital Inactive - follow with 1-4 to set SSR power output to 0%, or 9-12 for digital I/O then a time for pulsed output from 1-255 (100ms resolution) or 0 for permanent Board will return 0 for success, 1 for failure
35	23	Digital Set Outputs - the first byte will set all selected SSR power levels to 100%, All 100% = 255 (xxxx1111) All off = 0, 2nd byte sets digital outputs (lower 4 bits) Board will return 0 for success, 1 for failure
36	24	Digital Get Outputs - returns 2 bytes, the first corresponds with any SSR channel power output and the 2nd corresponds with active digital outputs, both are lower 4 bits in bytes.
37	25	Digital Get Inputs - returns 2 bytes, the 1st is always 0 as the SSR are not inputs, the 2nd bytes bits correspond with the digital io, a high bit meaning input is active (driven low) in lower 4 bits
48	30	Set Analogue Voltage - follow with 1-4 for SSR channel, a 0 byte then 0-100 for the % power and finally a time for pulsed output from 1-255 (100ms resolution) or 0 for permanent Board will return 0 for success, 1 for failure
58	3A	ASCII text commands - allows a text string to switch outputs, see section below
119	77	Get Serial Number - Returns the unique 6 byte MAC address of the module.
120	78	Get Volts - returns relay supply voltage as byte, 125 being 12.5V DC
121	79	Password Entry - see TCP/IP password

122	7A	Get Unlock Time - see section below
123	7B	Log Out - immediately re-enables TCP/IP password protection

Active and Inactive I/O - What do we mean by that?

Our Ethernet modules could potentially have many types of outputs. The ETH008 only has one type - Relays. The ETH044 has both Relay outputs and NPN Open Collector Transistor outputs. Activating a relay means turning the relay on. Likewise activating an output means turning the transistor on. This will cause it to sink current to 0v ground. If you had an LED connected from the output to 12v (via a resistor of course) it would light up. Other modules (not this one) could have PNP Open Collector Transistor outputs. These types will source current from the supply when active.

So here's the point: Active does not mean a high voltage comes out. It means that the output has been activated. That could result in the output sinking or sourcing current, depending on its type. The ETH044 outputs will sink current (up to 100mA) when active.

The same principle applies to the ETH044's inputs. These are designed to allow you to directly connect a VFC (Volt Free Contact). This could be from other relay contacts, thermostat contacts, alarm contacts etc. When the contacts are closed the input will read as active. In fact anything that pulls the input pin down to 0v will read as active. Do not think of the I/O in terms of a high or low voltage output. Think of it in terms of Active (or on, something is actively driving the I/O), or inactive (or off, nothing is driving the I/O).

It's a subtle point but one you need to be clear on.

Digital Active/Inactive Commands

These are 3 byte commands:

The first byte is the command, 32 (active means on) or 33 (inactive).

Second byte is the output number, 1-4 for the SSR's, or 9 - 12 for digital outputs.

Third byte is the on time. Set this to zero for un-timed operation, or 1-255 for a pulse in 100mS intervals (100mS to 25.5 seconds).

For example:

0x20 - SSR power to 100% command

0x03 - SSR channel 3

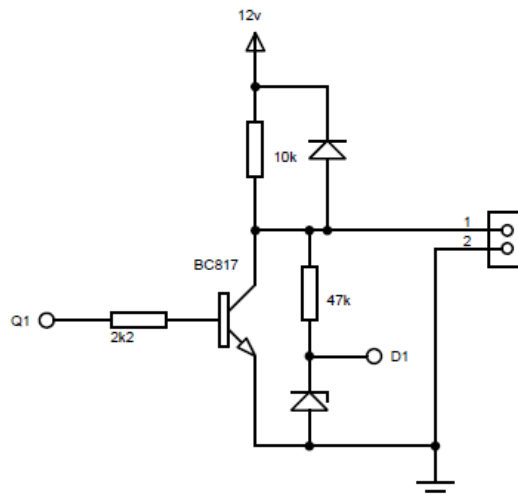
0x32 (50) - 5 seconds (50 * 100ms)

Board will return 0 for success, 1 for failure.

Note - All bytes in a command must be sent in one TCP/IP packet . Digital outputs pull the output pin down to 0v when active and pull up to 12v via a 10k resistor when inactive.

Digital inputs

Digital Inputs can be read with a command of 0x25. The two bytes returned will be encoded with each bit corresponding to whether the input is active or inactive. A high bit means the pin is either being pulled low (active) by the output or has an external device pulling low. To use the pins as inputs, the corresponding output must be made inactive. The first byte returned will always be zero. The 4 inputs are in the lower 4 bits of second byte.



Representative Digital I/O Schematic.

TCP/IP Password

If this option is enabled in the http configuration page then a password will be required to be entered before relay states can be changed. In the following example the password was set to "apple":

0x79 - 1st byte in frame sent to ETH044 to indicate password entry

'a' (0x61) - 2nd byte in frame (ASCII hex equivalent in brackets, full table is available at <http://www.asciitable.com/>)

'p' (0x70) - 3rd byte in frame

'p' (0x70) - 4th byte in frame

'l' (0x6C) - 5th byte in frame

'e' (0x65) - 6th byte in frame

These 6 bytes are then transmitted in the same transaction to the ETH044 and if the password is correct then 1 will be transmitted back, a failure will send 2.

The board will now accept changes from the device that entered the password. If communication becomes idle for more than 30 seconds then the password protection is re-enabled. There is also a log-out command of 0x7B to enable the protection immediately.

Get Unlock Time

Returns TCP/IP password protection status:

0 - password protection is enabled and password entry is required before changes can be made

1 to 30 - seconds until TCP/IP password protection is re-enabled. All authorised commands set the timer back to 30 seconds (including this one).

255 - TCP/IP password is not enabled.

ASCII text commands DOA and DOI

Following customer request we have added a feature that allows the outputs to be switched using an ASCII string, devices like a Mobotix camera can now switch relays with simple strings.

The string for activating output1 for 5 seconds is formatted using comma seperated variables with the following syntax:

```
":DOA,1,50,password"
```

To break this down ":" (hex 3A) at the start of the string indicates that there is an ASCII message to follow, "DOA" is digital output active, "1" is the output number, then "50" for 5 seconds (50x100ms) and finally the TCP password (if applicable).

If I wanted to make output 2 inactive for 3 seconds I would use:

```
":DOI,2,30,password"
```

To break this down ":" (hex 3A) at the start of the string indicates that there is an ASCII message to follow, "DOI" is digital output inactive, "2" is the output number, then "30" for 3 seconds (30x100ms) and finally the TCP password (if applicable).

Assuming no password is used the previous command would simply be:

```
":DOI,2,30 "
```

HTML commands DOAx and DOIx

Another customer requested feature, allowing the digital outputs to be switched by the http get function such as used in some voice over ip phones (VOIP). You can use the http get function to write to the io.cgi file with the following syntax:

```
192.168.0.200/io.cgi?DOA2=10
```

This would use the default address (192.168.0.200) and make output 2 active for 1 second.

Another example would be to set output 1 inactive for 10 seconds:

```
192.168.0.200/io.cgi?DOI1=10
```

You can test these functions by typing them directly into the address bar of most internet browsers. Also be aware that you may need to disable http authentication in the http configuration if your control device does not support it.

IP Addresses & DHCP Servers

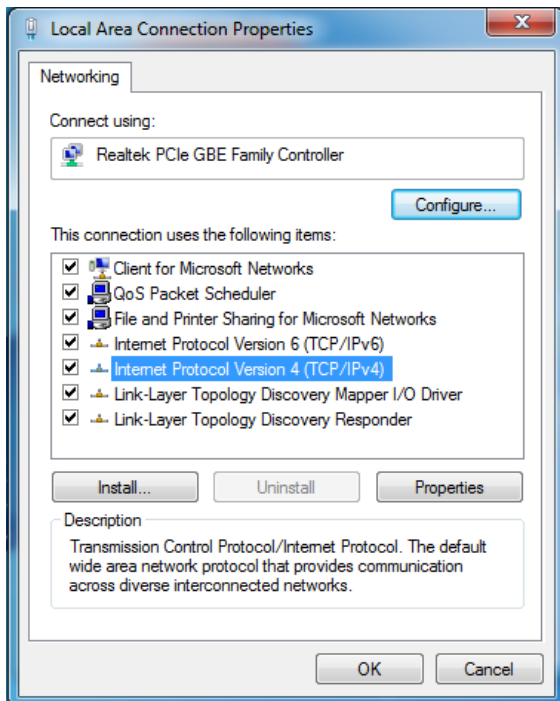
The easiest way to use the ETH044 is to connect it to a network with a DHCP server. In this case the ETH044 will have its IP address assigned automatically by the DHCP server.

If there is no DHCP server on the network, then a fixed IP address of 192.168.0.200 is used. To control the ETH044 using this fixed IP address your computer MUST be on the same subnet.

The next step is to set your computers IP address to 192.168.0.x where x is in the range of 1 to 255 but not 200 (the ETH044 is there!) or any other used IP addresses on the network.

The subnet mask dictates what IP addresses the PC can communicate with, we set this to 255.255.255.0 so the PC can talk to any module with an IP address of 192.168.0.x

The default gateway is likely to be the IP address that the internet connection is located at and the DNS server can be either your router address or google provide a DNS service at 8.8.8.8..



This image is for a PC running Windows 7.

To get to the TCP/IP properties screen, go to:

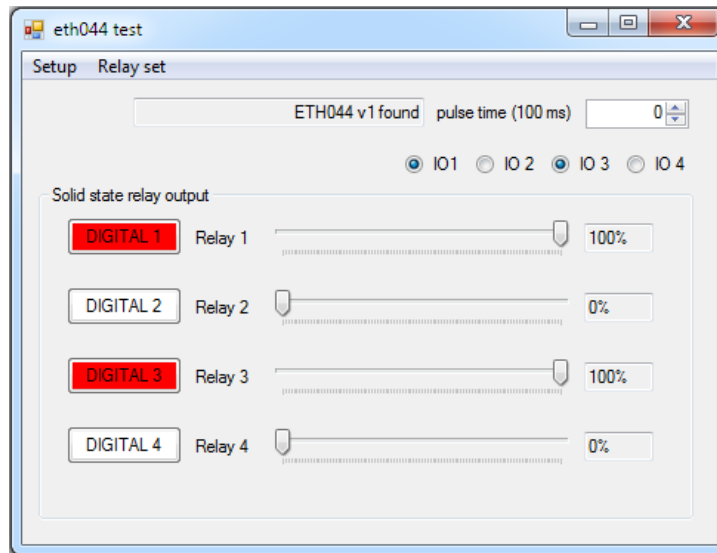
Control Panel->Network and Sharing Center->Local Area Connection->Properties.

Select Internet Protocol Version 4(TCP/IPv4) from the scroll box and click Properties.

That gets you to the dialog box shown left.

Test program and example source code

To get the ETH044 up and running in the minimum amount of time we have put together an example program to demonstrate the functionality of the module. We provide the full source code for this program. You may examine this code to see how it works or use it as a starting point for your own application.



Visual studio express C# examples

The test program is available as Visual C# express ready built installation files [here](#), or as Visual C# express project with source files [here](#).

Visual studio express is provided free from Microsoft: [Visual Express](#)

Access from the Internet

The ETH044 can be controlled over the internet almost as easily as on your local network. Your network will most likely be connected to the internet with a broadband router. This will provide NAT (Network Address Translation) and Firewall services. To access the ETH044 from the internet you will need to open up port 17494 (0x4456) to allow incoming TCP connections. Be careful not to open up any other ports. There are a wide variety of routers and we cannot give details for all of them. If in doubt ask your system administrator for assistance. The following shows how to open up a port on a Netgear DG834 router. The example is for a board on the network at 192.168.0.96 and port 17494, although these values are easily altered.

Add Services

Service Definition

Name:

Type:

Start Port:

Finish Port:

After logging on to your routers setup page, the first thing to do is create a new service. Click on the "Services" menu then "Add Custom Service". Enter a name for the service, select TCP and enter the ETH044's port address for both the start and finish ports. Click "Apply".

Inbound Services

Service:

Action:

Send to LAN Server: . . .

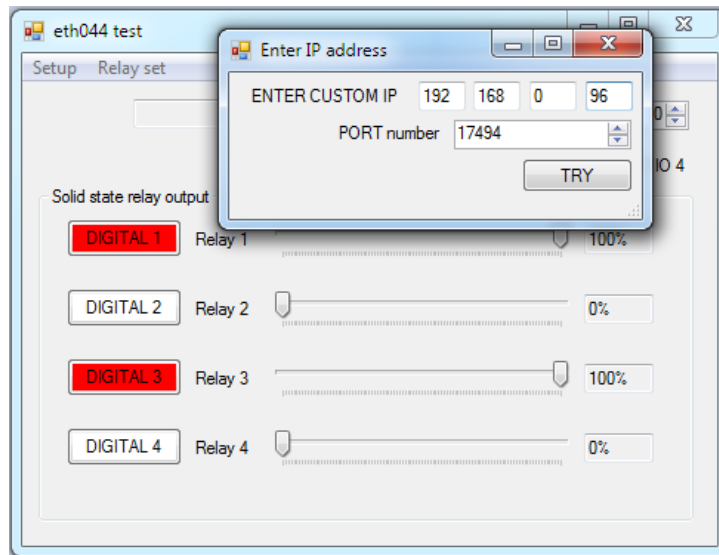
WAN Users:

start: . . .

finish: . . .

Log:

Now go to the "Firewall Rules" menu and click "Add" in the Inbound services section. Select the ETH044 service and ALLOW always. The "Send to LAN Server" IP address is the ETH044's IP address, 192.168.0.96 in the example above but check what it is on your network. Click "Apply" and that's it. The ETH044 is now accessible over the internet. Before you close the routers setup pages, go to the "Router Status" menu and make a note of its ADSL port IP address. This is the routers internet facing IP address.



To test this you will need a computer that has its own internet connection and is NOT connected to the same network as the ETH044. Download and run the test program above and select Custom IP. In the pop-up box enter your routers internet facing IP address. Click on "Try IP" and it will connect you to the ETH044 just as if it were on your own network.

Android & iPhone Apps.

We have a free app **IO network** available for Android and iPhone to remotely control your relays, download from Google Play or iTunes. Search for "Devantech" and you will find the app.

Board dimensions

