WINFORD ENGINEERING, LLC
4561 Garfield Road • Auburn, MI 48611
Phone: 1-877-634-2673
FAX: 1-989-671-2941
www.winford.com

ETH32 UDP Configuration Protocol Reference

Updated for firmware Version 3.000

Introduction

This document describes the protocol used to configure the network settings (IP address, ...) of Winford Engineering's ETH32 I/O Device. Please note that most users will not need to read this document since Winford Engineering provides ETH32 configuration utilities for Windows and Linux platforms. You will need to consult this document if you are planning to write a configuration utility for an unsupported platform.

Configuration Process

The discovery and configuration of the ETH32 device is performed using UDP broadcast packets. The main reason for this approach is that it allows any ETH32 device located on the same physical network segment (connected by hub or switch) to be discovered and reconfigured, regardless of whether the ETH32's currently-configured IP address is on the same subnet as your PC's IP address. In other words, since broadcast packets are used, all ethernet devices on the local network segment will receive the packets, although devices other than the ETH32 will simply ignore the packets. The configuration process generally proceeds as follows:

- The PC broadcasts a packet ("Device Query" below) to discover the presence of any ETH32 devices on the local network segment.
  - It is recommended to broadcast this packet at least twice, separated by a 1-2 second pause. This helps to ensure all devices are detected in case one of the packets becomes corrupted.
- Each ETH32 which receives the query packet will respond ("Query Response" below) to let the PC know it exists.
- The PC receives the response packets and creates a list of devices that are presented to the user.
  - The configuration utility must remove duplicate responses. Since it sent out at least two query packets, in most cases it will receive two responses from each device.
- The user chooses which device he wants to configure and enters the new configuration settings.
- The PC sends a broadcast packet ("Set Configuration" below) to configure the device with the new settings
  - The packet includes the MAC address and serial number of the device which should be configured. Only the desired device acts on the packet. All others simply ignore it.
- The ETH32 device responds to confirm that the new configuration was received
  - If the configuration switch on the ETH32 device was set to disable configuration, the device will respond to indicate that the new configuration information was rejected.

Please note that if a router separates your PC and the ETH32 device, the router will not forward broadcast packets and the ETH32 device will not be detected.

UDP Port

The ETH32 devices listen on UDP Port **7151** for the configuration packets.

Broadcast Packets

The programming details of sending broadcast packets will vary depending on the platform and programming language being used. Typically, you should enable the Broadcast flag (SO_BROADCAST or equivalent) and set the destination address to 255.255.255.255.

On some systems, particularly those with more than one network interface, you may need to set the destination address to the broadcast address of the particular network interface connected with the ETH32 (on the same segment). For example, if your ethernet card is assigned an IP address of 192.168.1.1 with a subnet mask of 255.255.255.0, the appropriate broadcast address would be 192.168.1.255. Please note that in this case the appropriate broadcast address is only necessary in order to get the packets broadcast out onto the right network segment and does not limit which ETH32 devices will be found. All ETH32 devices on that segment will be discovered, regardless of their currently-configured IP address.

Response Packets

All replies from the ETH32 to the PC will have the following characteristics:

- They will be sent to the source MAC address of the incoming request packet. In other words, replies are sent directly to the PC, not broadcast.
- The destination IP address will be the source IP address from the incoming request packet (the IP address of the PC)
- The source IP address will be the currently-configured or active DHCP-provided IP of the ETH32.
- The destination UDP port will be the source UDP port from the incoming request packet
- The source UDP port will be 7151

Packet Descriptions

The contents of each packet type is described below. All information is sent in binary format. Multi-byte information is sent in "network order" (big-endian), which means the most-significant byte (MSB) is first and the least-significant byte is last.

Device Query

This packet is broadcast by the PC to discover which ETH32 devices are present on the local network segment. A specific identification number is included as part of the packet to let the ETH32 know this is a real query (as opposed to traffic from other devices/applications that happen to use the same UDP port). If this number is not present, the ETH32 will ignore the request.

Index	*Length*	*Value*	*Description*
0	1	0x01	Command ID
1	4	0x44EE4411	Identifies the packet as an ETH32 device query

Query Response

This packet is sent back to the PC by any ETH32 which receives a Device Query packet. It provides identification information, current configuration, and other information about the ETH32 device.

Index	*Length*	*Value*	*Description*
0	1	0x02	Command ID
1	1	0x69	Product ID: ETH32 = 0x69 (105)
2	6		MAC address of ETH32 device
8	2		Serial number (batch number portion)
10	2		Serial number (unit number portion)
12	4		IP address
16	4		Gateway IP address
20	1		Number of 0xFF bytes in subnet mask *
21	1		Mask of last non-0xFF byte of subnet mask *
22	1		Firmware version (major portion)
23	1		Firmware version (minor portion)
24	1		Configure switch on device is enabled (1) or disabled (0)

*** Note about network mask**: This protocol uses a somewhat unusual form of representing network subnet masks, designed mainly for efficiency of processing within the ETH32.

- The first byte represents the number of bytes of the subnet mask that are 0xFF (255).
  - For example, for the subnet mask of 255.255.255.0, this would be 3.
- The second byte represents the actual mask value of the last non-0xFF byte.
  - For 255.255.255.0, this would be 0.
- In the rare case that a netmask of 255.255.255.255 needs to be represented, either 4,0 or 3,255 are valid representations.
- Examples:
  - 255.255.255.0: 3, 0
  - 255.255.0.0: 2, 0
  - 255.255.128.0: 2, 128
  - 255.255.255.192: 3, 192

Set Configuration

This packet is sent to the ETH32 device by a PC in order to set new network configuration settings.

Index	*Length*	*Value*	*Description*
0	1	0x03	Command ID
1	6		MAC address of ETH32 to configure
7	2		Serial batch of ETH32 to configure
9	2		Serial unit of ETH32 to configure
11	4		New IP address
15	4		Gateway IP address
19	1		Number of 0xFF bytes in subnet mask *
20	1		Mask of last non-0xFF byte of subnet mask *
21	1		DHCP (firmware v3.000 and on - previously padding): 0: Disabled (use stored values indicated above) 1: DHCP Enabled
22	2		Internet checksum of this command block, from Command ID through DHCP **

* See note about network mask representation in Query Response section.

** The checksum must be valid or the ETH32 device will ignore the Set Configuration packet. The checksum is calculated using the "Internet Checksum" method described by RFC 1071. In the C language, one implementation example based on RFC 1071 is as follows:

```
unsigned short cksum(unsigned char *buf, int count)
{
      /* Compute an Internet checksum of the given buffer.
       *   buf: Points to start of buffer
       * count: Number of bytes on which to compute checksum
       */
      long sum = 0;

      while( count > 1 )
      {
            /*  This is the inner loop */

            sum += *(unsigned short *)buf;
            buf+=2;
            count -= 2;
      }

      /*  Add left-over byte, if any */
      if( count > 0 )
            sum += *(unsigned char *)buf;

      /*  Fold 32-bit sum to 16 bits */
      while (sum>>16)
            sum = (sum & 0xffff) + (sum >> 16);

      return(~sum);
}
```

Confirmation

This packet is sent from the ETH32 device to the PC when a valid Set Configuration packet has been received. This packet is only sent if the Set Configuration packet listed the correct MAC address and serial number of the ETH32 device and the checksum was valid. This packet indicates whether the new configuration was accepted or rejected due to the configuration switch on the device.

Index	*Length*	*Value*	*Description*
0	1	0x04	Command ID
1	1		Status: 0: Rejected 1: Accepted and saved

IP Configuration Query

This packet is implemented on firmware versions 3.000 and on. This packet is broadcast by the PC to discover which ETH32 devices are present on the local network segment and request each device to send its currently-configured IP settings. This packet should be used to discover what IP address an ETH32 has been assigned by the DHCP server, if DHCP is enabled on the device. However, it can also be used to discover devices that are configured with static IPs. A specific identification number is included as part of the packet to let the ETH32 know this is a real query (as opposed to traffic from other devices/applications that happen to use the same UDP port). If this number is not present, the ETH32 will ignore the request. This command includes the ability to filter which ETH32 devices will respond to the query. If one or both bits are set in the Filter flag, only those ETH32 devices that match the information provided will reply to the query. If the Filter flag is zero, all ETH32 devices will respond.

Index	*Length*	*Value*	*Description*
0	1	0x05	Command ID
1	4	0x44EE4411	Identifies the packet as an ETH32 device query
5	1		Filter flag: Bit 0: MAC address must match Bit 1: Serial Number batch/unit must match
6	6		MAC address of ETH32 device
12	1	0x69	Product ID: ETH32 = 0x69 (105)
13	2		Serial number (batch number portion)
15	2		Serial number (unit number portion)

IP Configuration Response

This packet is implemented on firmware versions 3.000 and on. This packet is sent back to the PC by any ETH32 device that receives a IP Configuration Query that matches the information indicated by the Filter flag of that request. It provides the current status of whether DHCP is enabled on the device as well as the currently active IP configuration settings. If DHCP is enabled, this packet returns the DHCP-assigned IP configuration settings. If DHCP is enabled, but the device has not been able to obtain an IP address lease, the returned settings will be all zeros. **If DHCP is disabled, this packet will return the static IP configuration settings currently stored and being used by the ETH32 device (the same IP settings returned by the Query Response).**

Index	***Length***	***Value***	***Description***
0	1	0x06	Command ID
1	1		DHCP Status: 0: Disabled 1: Enabled
2	6		MAC address of ETH32 device
8	1	0x69	Product ID: ETH32 = 0x69 (105)
9	2		Serial number (batch number portion)
11	2		Serial number (unit number portion)
13	4		IP address
17	4		Gateway IP address
21	1		Number of 0xFF bytes in subnet mask
22	1		Mask of last non-0xFF byte of subnet mask