How DHCP Works

For a detailed description of DHCP, we suggest that you download RFC 1541 from any of the Internet draft repository sites. A good place to start is www.rfc-archive.org

When a DHCP device attaches itself to the network for the first time, it broadcasts a DHCPDISCOVER packet. A DHCP servers on the local segment will see the broadcast and return a DHCPOFFER packet that contains an IP address and other information. The servers may or may not conduct some sort of preliminary testing prior to offering the address, such as generating an ARP or an ICMP echo to see if the address is already in use by another node somewhere.

The client may receive multiple DHCPOFFER packets from any number of servers, so it must choose between them, and broadcast a DHCPREQUEST packet that identifies the explicit server and lease offer that it likes the best.

Assuming that the offer is still valid, the chosen server would return a DHCPACK that tells the client the lease is finalized. If the offer is no longer valid for some reason-perhaps due to a time-out or another client allocating the lease-then the selected server must respond with a DHCPNAK message. This would cause the client to send another DHCPDISCOVER packet, starting the process over again.

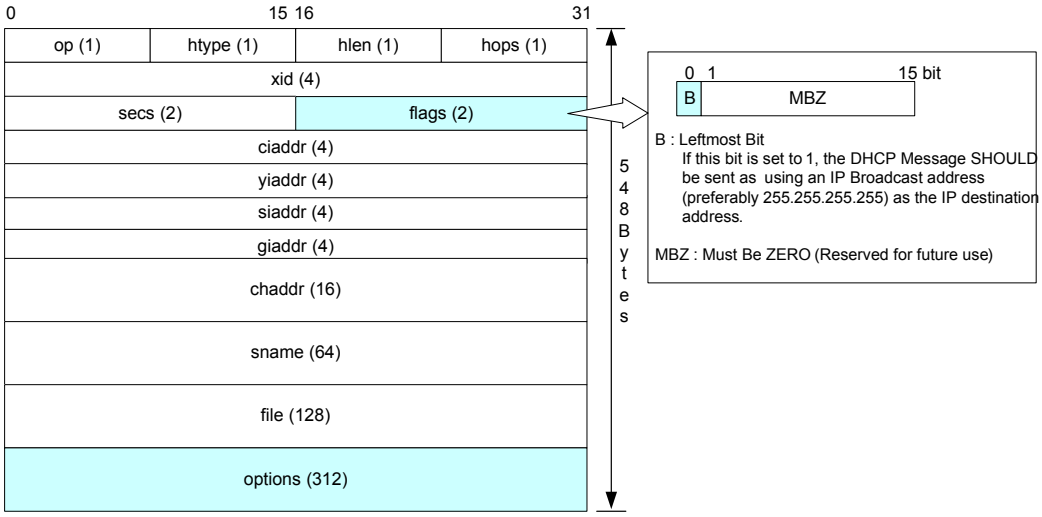
Once the client receives a DHCPACK, then all ownership and maintenance of the lease is the responsibility of the client. For example, a client may refuse an offer that is detailed in the DHCPACK message, and it is the client's responsibility to do so. Clients are supposed to test the addresses that have been offered to them by conducting ARP broadcasts. So if another node responds to the ARP, the client would assume that the offered address is in use. At this point, the client would reject the offer by sending a DHCPDECLINE message to the offering server, and would also send another DHCPDISCOVER packet, thereby starting the process yet again.

Once the client has the lease, it must be renewed prior to the lease expiration through another DHCPREQUEST message. If the server doesn't hear from the client by the end of the lease, it marks the lease as non-renewed, and makes it available for other clients to use.

Implementation of DHCP client

DHCP clients and servers communicate through the exchange of message described in the protocol specification. All DHCP messages include a fixed-format section and a variable-format option section. The variable-format section holds options, which carry additional configuration parameters. The contents of the fixed-format section and the format of the options section vary according to the type of DHCP message.

Here is the DHCP message format. The numbers in parentheses indicate the size of each field in octets



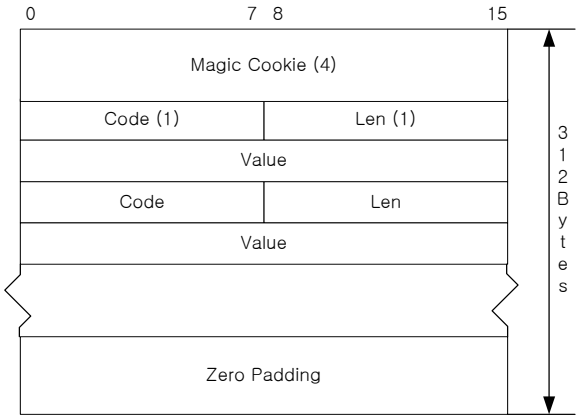
op	Message op code / message type. 1 = BOOTREQUEST, 2 = BOOTREPLY
htype	Hardware address type (e.g., '1' = 10Mb Ethernet)
hlen	Hardware address length (e.g. '6' for 10Mb Ethernet)
hops	Client sets to zero, optionally used by relay agents when booting via a relay agent.
xid	Transaction ID. A random number chosen by the client, used by the client and server to associate the request message with it's and response.
secs	Seconds passed since client began the request process
ciaddr	Client IP address. Filled in by client if it knows it's IP address (from previouse requests or from manual configurations). and can respond to ARP requests.
yiaddr	'your' (client) IP address. Server's response to client
siaddr	Server IP address. Address of sending server or of the next server to use in the next

	bootstrap process step.
giaddr	Relay agent IP address, used in booting via a relay agent.
chaddr	Client hardware address.
sname	Optional server host name. Null terminated string.
file	Boot file name. Null terminated string; "generic" name or null in request, fully qualified directory-path name in reply.
options	Optional (BOOTP semantics) parameters field. In real DHCP messages at least one option (message type) must always be present, so this field is never empty

The DHCP message are defined in the dhcp.h as RIP_MSG data type.

```
typedef struct _RIP_MSG
{
        u_char  op;             // DHCP_BOOTREQEUST or DHCP_BOOTREPLY
        u_char  htype;          // DHCP_HTYPE10MB
        u_char  hlen;           // DHCP_HLENETHERNET
        u_char  hops;           // DHCP_HOPS
        u_long  xid;            // DHCP_XID
        u_int   secs;           // DHCP_SECS
        u_int   flags;          // DHCP_FLAGSBROADCAST
        u_char  ciaddr[4];
        u_char  yiaddr[4];
        u_char  siaddr[4];
        u_char  giaddr[4];
        u_char  chaddr[16];
        u_char  sname[64];
        u_char  file[128];
        u_char  OPT[312];
}RIP_MSG;
```

Here are some important DHCP options:



Message Type (a DHCP control).

Specifies the type of the DHCP message in order to be more specific than the originally BOOTP field 'op'. Has to appear in every DHCP message. Different message types are used at different stages of the client/server interaction.

Renewal Time Value (a DHCP control).

Specifies the time interval from address assignment until the client attempts to contact the server that originally issued the client's network address before the lease expire.

Parameter Request List (a DHCP control).

A list of valid DHCP option codes. Used by a DHCP client to request values for specified configuration parameters.

Subnet Mask (a Configuration parameter)

Specifies the client's subnet mask.

DNS Option (a Configuration parameter)

Specifies a list of DNS name servers available to the client.

Below enumeration type defined in the “dhcp.h” file and other options skipped.

Code	Enumeration Type	Description
0	padOption	used to cause subsequent fields to align on word boundaries
1	subnetMask	specifies the client's subnet mask
3	routersOnSubnet	a list of IP addresses for routers on the client's subnet
6	dns	specifies a list of DNS servers available to the client
12	hostName	specifies the name of the client
50	dhcpRequestedIPaddr	request that a particular IP address be assigned by the server
51	dhcpIPaddrLeaseTime	a lease time for the IP address
53	dhcpMessageType	used to convey the type of the DHCP message
54	dhcpServerIdentifier	the IP address of the selected server
55	dhcpParamRequest	request values for specified configuration parameters
61	dhcpClientIdentifier	specify client unique identifier
255	endOption	marks the end of valid information

About the sample DHCP client source

1. Testing environment

   -. DHCP server : Windows2000 server

   -. DHCP client : EVB-B1 (Atmega128 16Mhz)

2. Little endian & Big endian

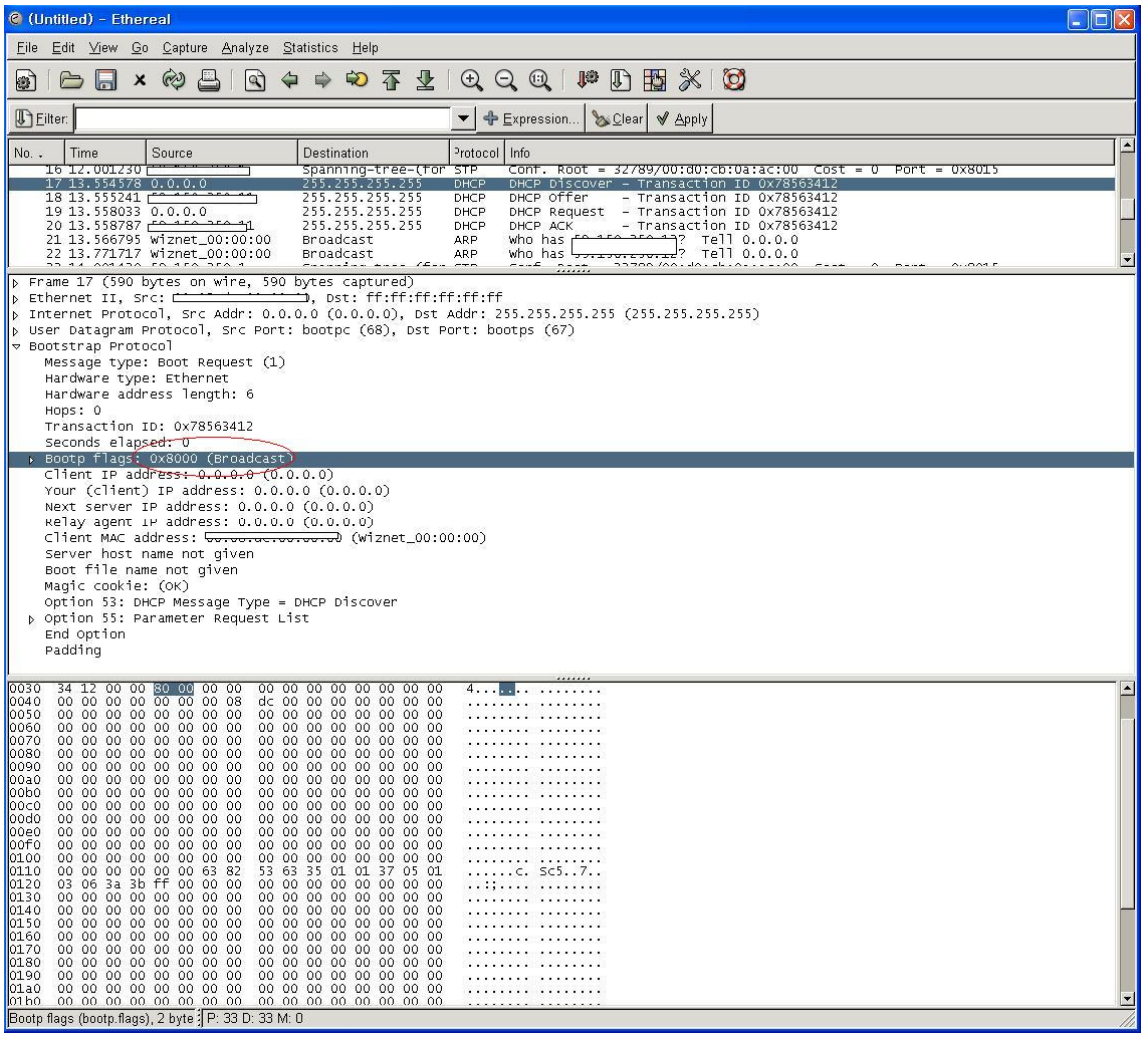
Our sample source program run on the little-endian CPU so if you use big-endian CPU change the byte ordering.

In the “dhcp.h”

#define DHCP_FLAGSBROADCAST 0x0080 (little-endian)

#define DHCP_FLAGSBROADCAST 0x8000 (big-endian)

When sending the DHCP discover and DHCP request message, the Bootp flags must be set as broadcasting as below.



In the “socket.h”

if you use big-endian CPU, just remove below define statement.

#define LITTLE_ENDIAN

3. Explanation of sample source code

-. MAC address: We use our test MAC address, so when you port our source code to your system, you should use your own MAC address

-. getIP_DHCPS() function: In the main.c file, this function is the entry point of the DHCP operation. If it acquire the IP address from the DHCP server successfully, it return 1.
If it fail, just set the IP address as 192.168.0.5.

-. check_DHCP_state() function: After getting the IP address from the DHCP server, DHCP client should renewal this IP address by sending DHCP REQUEST message in every half of the lease time. This function compare the lease time with internal timer of CPU and if it exceed the lease time, sending DHCP REQUEST message.