

**WARNING - EXPERIMENTERS ONLY!**

Host mode on TNC-2 and derivative TNCs at this time remains experimental. Obvious features received a quick "going over" and rather cursory debug, but discovery of the remaining inevitable bugs and critical programming oversights awaits the adventurous reader/developer. Please direct discoveries and criticisms:

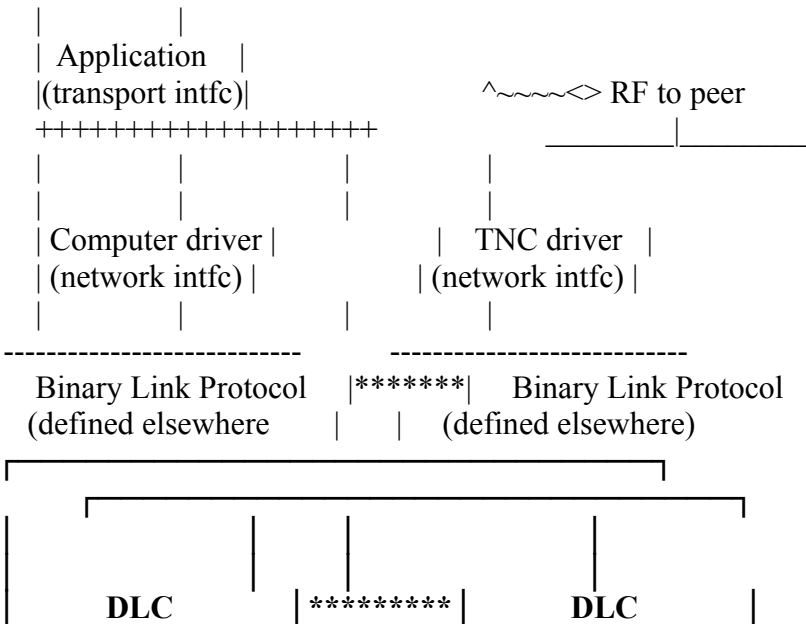
via slow boat : to Howard Goldstein  
5201-23rd Ave. N.  
St. Petersburg, FL 33710 USA  
via compuserv : to 75006,702  
via ham packet: to N2WX @ W4DPH.FL

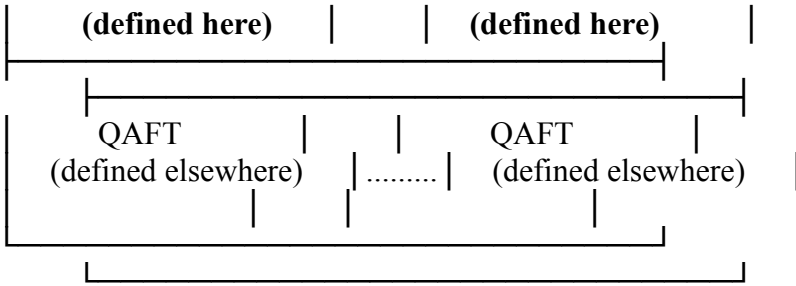
Introduction

This document describes the data-link control protocol (DLC) used by the TNC-2 terminal node controller on its RS-232 line.

Where it fits in

DLC provides a bidirectional error-protected two-way queue, for BLP (the binary interface protocol). DLC uses the underlying QAFT physical layer driver as its transmission medium.





Key:

..... One connection

\*\*\*\*\* One or more logical connections

### DLC encapsulation

The DLC interface communicates with its peer interface by transceiving DLC packets using QAFT services along the underlying QAFT link.

### DLC packet structure

>| DESTINATIONbyte | SOURCEbyte | CTRLbyte | [variable per CTRL] |<

byte :: 8 bit unsigned integer

### DESTINATION

**SOURCE** : Physical address of the sending (SOURCE) and sinking (DESTINATION) DLC devices for this DLC packet. <Optional in TNC-2>

**CTRL** : type of packet

The high nibble of the control byte contains the DLC packet type. Values in the low nibble are sequence numbers.

hex	---		
10	RESET	- Link restart	(request)
20	RESET_ACK	- Link restarted	(response)
4x	DATA	- Sequenced data (x := mod 16 sequence # of data)	(request)
5x	DACK	- Sequenced data acknowledgement (x := mod 16 acknowledges up through seq # x-1)	(response)

## Defining DLC

DLC is a balanced protocol. Since there are no master-slave relationships, any one DLC device can communicate with any other electrically connected DLC device. (for ex., one computer-one TNC, or two computers and no TNCs, or two TNCs back-to-back). Note however that the current TNC-2 implementation does not specify source/destination addressing, therefore only one DLC link may take place across the physical medium.

DLC is described and was implemented as an event-driven finite-state machine.

## DLC Objects

Each DLC link uses these variables and objects:

DLSTATE: One of  $n$  states. (see section on DLC states, below)

DQUEUE: FIFO queue of packets as yet unacknowledged by the peer DLC device.

## DLC States

Four states and seven events are defined. See the appendix for the state tables.

### States

#### DLIDLE

The DLC link is now inactive and awaits either (1) receipt of a link reset (*RESET*) packet; or (2) local command to initiate an outgoing call from the upper-level BLP driver.

If the DLC machine receives *RESET*, the DLC driver shall respond with a reset-acknowledged (*RESET\_ACK*) packet and reset all of its upper level BLP links.

#### DLRESET

The DLC machine enters this state when a local event occurs (ex: upper-level BLP attempts to place a call, or a datagram is received). When the BTIMER-expired event occurs, the machine in this state sends a *RESET* packet to the peer.

The DLC machine leaves this state and is ready for data transfer when a reset-acknowledged (*RESET\_ACK*) or another *RESET* indication is received.

DLC 1.a Mar 91

## DLDATA

Local and peer DLCs are ready to transact. The state changes to DWAIT when new outgoing sequenced data packets are enqueued. DDATA state is reentered once all outstanding packets are acknowledged.

## DLDWAIT

The DLC machine has data outstanding for the remote DLC, and is awaiting a response. If a BTIMER-expired event occurs, the data packet(s) is (are) retransmitted. If a DACK packet acknowledging all outstanding packets is received, the state changes to DDATA.

## DLC packet types

### RESET - Call setup

format:

|DESTINATION|SOURCE|10|  
^^^^^^^^^^^^^^^^

(optional)

If the DLC link is idle, one side will transmit a RESET packet to fix both sides of the link to a known (DLDATA) state.

When a device receives a RESET packet it shall reply with a RESET\_ACK packet and reset all of the BLP circuits its associated with.

### RESET\_ACK - DLC resetted

format:

|DESTINATION|SOURCE|LCN|20|  
^^^^^^^^^^^^^^^^

(optional)

Receipt of the RESET\_ACK packet indicates that the DLC link is successfully opened and BLP may begin using the link.

### DLDATA - Sequenced data command packet

format:

|DESTINATION|SOURCE|4x|[data0...datan]|  
^^^^^^^^^^^^^^^^

DLC 1.a Mar 91  
(optional)

x :: sequence number, mod 16

DACK - Sequenced data acknowledgement

format:

|DESTINATION|SOURCE|5x|  
^^^^^^^^^^^^^^^^

(optional)

x :: sequence number, mod 16

Acknowledges packets through sequence number 'x'

DLC State table

queued	rx RESET Local start	rx RESET_ACK	rx DATA	rx DACK, all acknowledged	timer xpd	new data
DLIDLE	Reset BLP,clr queues state>DLDATA	send RESET_ACK,				state>DLRESET
DLRESET	"		state>DLDATA		send RESET pkt	
DLDATA	" state>DLDWAIT			send DACK		
DLDWAIT DLDATA	"			send DACK	state>DLDATA	send all