

THE NETWORK USER INTERFACE

David W. Borden, K8MMO
Director, AMRAD
P.O. Drawer 6148
McLean, VA 22106-6148

Abstract

This paper discusses the user interface, the way the human operator interacts with the amateur digital network using their Packet Assembler/Disassembler (PAD) or older Terminal Node Controller (TNC). In this day of increasing features and networking, the packet radio user is expected to remember more and more commands to communicate on the amateur radio packet network. The user desires more and more features and yet wants to keep it simple. The past two years has seen more and more TNCs come on the market, but no standardization of user commands. Now that software is starting to appear to transform these TNCs into Packet Assembler/Disassemblers (PADs), the time is here to make this right, to standardize the user interface using X.3 and X.28 protocols.

Introduction

In the beginning, there was Doug Lockhart. Several Washington area amateur radio operators purchased his TNC and began packeteering (actually framing). As is described in his excellent paper "X.3 AND X.28 PROTOCOLS FOR TERMINAL NODE CONTROLLERS", presented at the Fourth Computer Networking Conference, this TNC software allowed two commands. Control-X caused a connection to be made and Control-Y caused a disconnect. This had a number of drawbacks, the most noteworthy of which was that you had no control over the device for dynamic situations. When you wanted to work HF and should have changed your maximum frame size, or re-try count, or FRACK, you did not even know that such things were possible. You could do three things, monitor the channel seeing everything, communicate in the unconnected mode with no acknowledgements that your packets were received, or communicate in connected mode with one other amateur. Those were the simple days. But, packet radio matured, more control was required.

Doug's Vancouver code was then subjected (the only correct word) to many changes regarding the user interface. Hank Magnuski and his San Francisco gang did most of it until we in AMRAD learned the trick. We hated Guru assigned numbers for addresses (limited to 253) so we implemented the Terry Fox addressing scheme (actually amateur call signs). Hank made a digipeater using a STD Bus computer. This required changes to allow the direct frame not to interfere with the repeated frame. Transparent mode was added to allow computers connected to our Vancouvers. AX.25 got coded, requiring more user changes. The real breakthrough on Vancouver was the AMRAD Daughter board, giving us more PROM space. Code could grow even more without giving up an old feature. More commands were added. Today you have to consult WD5DBC, Howard Cunningham, the keeper of the Vancouver board AX.25 code, to tell you all the commands possible. The basic San Francisco code remains, however. To get to command mode, you type a Control-P and then follow it by some character, like C for connect or D for disconnect. Here you get a hint of the problem, however. As more features were added, more commands were required to control them. In a very short time you can forget how to get into DEBUG mode, or turn MONITOR off, or string digipeater call signs. Typically what you did then was give your Vancouver away to your club for digipeater use and get a TAPR TNC. Surprise! You have a whole new set of commands to learn.

Menu Driven TNCs/PADs

Since I own a Macintosh and friends of mine own Commodores, I realize that some user interfaces do not require strange commands to deal with the amateur packet radio network. But these computers are really acting as super-smart terminals and provide the desired TNC with the commands required, transparent to the user who points with their mouse or whatever. The intent of this paper is to ignore those menu driven units as the majority of packeteers have TAPR or TAPR clone units and use terminals or computers with less sophisticated user interfaces to communicate. Whatever changes are made to the TNCs/PADs, the menu driven software will be changed to keep up and to keep the whole thing transparent to the user.

Features Requiring Support

Basically the user wants complete control over everything while keeping it simple (just like computer users everywhere, they want control of the world with two commands). I have identified the following actions requiring command support. There are probably others, but these are supported by my TAPR2 PAD running experimental NET1.0, AX.25/AX.75 Level 3 code. Note, I only define new network commands, other older commands are in the TAPR documentation:

- a. Getting to the command mode and back

Control-C
CONVERS

- b. Monitoring the channel or not with features

MALL ON
MALL OFF
MFILTER n1[,n2[n3[,n4]]]
MONITOR ON
MONITOR OFF
MHEARD
MHCLEAR
MSTAMP ON
MSTAMP OFF

- c. Level 2 Connection with another station or packet switch (essentially "framing") with features

CMSG ON
CMSG OFF
CONMODE CONVERS
CONMODE TRANS
CONNECT call #VIA call2[,call3...,call19]]
CONOK ON
CONOK OFF
CONPERM ON
CONPERM OFF
CONSTAMP ON
CONSTAMP OFF
CPACTIME ON
CPACTIME OFF
CTEXT text

- d. Level 2 Multiple Connection (essentially multiple framing)

LCSTREAM ON
LCSTREAM OFF
STREAMSW n
STREAMCA on
STREAMCA off
STREAMDB on
STREAMDB off
USERS n

e. Level 2 Disconnection

DISCONNECT

f. Level 3 call placement (essentially "packeting")

CALL [callsign] [@address] - Try to place a Level 3 call

g. Level 3 call servicing

CRESET - Reset Level 3 flow control on my existing call

INT - Cause an interrupt received indication to be displayed at the far end of my call

h. Level 3 call clearing

CLR - Tear down my Level 3 call (like hanging up my phone so I can redial)

i. Enabling/Disabling Digipeating

DIGIPEAT ON
DIGIPEAT OFF

j. Enabling/Disabling Packet Switching (TAPR acts as packet switch)

SWITCH ON - Cause my TAPR device to act as a packet switch

SWITCH OFF - Return my TAPR device to normal user status from packet switch status

SWITCHID nnnnnnn - The ID number of my switch is nnnnnnn

k. Enabling/Disabling Level 3 Network Operations

NETWORK ON - Place my TAPR device in PAD status, vice TNC status

NETWORK OFF - Return my TAPR device from packet PAD status to the old frame TNC status

l. Setting/changing my Level 2 address (callsign)

MYCALL call

m. Setting/changing my Level 3 network address (using X.121NA)

MYNADDR nnnnnnnnn - My Level 3 network address is nnnnnnnnn

n. Determining my status as regards connection, or setting of any parameter

CSTATUS
DISPLAY

o. Setting/changing of PAD transmission parameters

DWAIT n
FRACK n
PACLEN n
RETRY n
TRIES n
TXDELAY n

p. Enabling/Disabling Level 2 Trace and Level 3 Trace for debugging

NETBUG ON (Level 3)
NETBUG OFF (Level 3)
NETTRACE ON (Level 3 - Trace packets)
NETTRACE OFF (Level 3)
TRACE ON (Level 2 - Trace frames)
TRACE OFF (Level 2)

q. Miscellaneous nice to have features

8BITCONV ON	8BITCONV OFF	AUTOLF ON
AUTOLF OFF	AWLEN n	AX25L2V2 ON
AX25L2V2 OFF	AXDELAY n	AXHANG n
BEACON EVERY n	BEACON AFTER n	BKONDEL ON
BKONDEL OFF	BTEXT text	BUDLIST ON
BUDLIST OFF	CALIBRA	CALSET
CANLINE n	CANPAC n	CHECK n
CLKADJ n	CMDTIME n	COMMAND n
CR ON	CR OFF	
DAYTIME yymmddhhmm		
DAYUSA ON	DAYUSA OFF	DELETE ON
DELETE OFF	ECHO ON	ECHO OFF
ESCAPE ON	ESCAPE OFF	FLOW ON
FLOW OFF	FULLDUP ON	FULLDUP OFF
HEADERLN ON	HEADERLN OFF	HID ON
HID OFF	ID	LCALLS
LCOK ON	LCOK OFF	LFADD ON
LFADD OFF	MAXFRAME n	MCOM ON
MCOM OFF	MRPT ON	MRPT OFF
MYALIAS	NEWMODE ON	NEWMODE OFF
NUCR ON	NUCR OFF	NULF ON
NULF OFF	NULLS n	
PACTIME EVERY n		
PARITY n	PASS n	PASSALL ON
PASSALL OFF	REDISPLA n	RESET
RESPTIME n	SCREENLN n	SENDPAC n
START n	STOP n	TRANS
TRFLOW ON	TRFLOW OFF	TXFLOW ON
TXFLOW OFF		TXFLOW OFF
UNPROTO cal11 [via cal12[,cal13...,cal19]]		
XFLOW ON	XFLOW OFF	XMITOK ON
XMITOK OFF	XOFF n	XON n

These available features are a 'big step forward from my Vancouver, but still lack standardization. The TAPR way of addressing these features is probably the widest know in the packet community, but is not standard. Other TNCs do not use these TAPR commands unless they use the same code (essentially TAPIR "clones").

Time To Implement

Reference to Lockhart's paper on X.28 and X.3 (he points out it is premature to discuss X.29, a Level 5 issue) reveals details about those two protocols and is excellent reading. The intent of this paper is not to attempt to improve on Doug's paper or even bring it up to date (For example: Interrupt and Reset commands are being used now>. But rather, this paper is an appeal to implement these protocols on TAPR boards and clones. The time for implementation is now, since Level 3 has arrived. Users have to learn new commands to give their PADs anyway, so why not go to the standard now?

Switch Supplied Network Access PAD

There is a simple way out of the need for standardization. The packet switch can be made super-smart, like the Macintosh or Commodore software, and act as a big gateway to the network for all current users. The switch looks at what gibberish you are typing to it and does the right thing (tries to place your call, etc.). Howard Goldstein has done this for his TAPR packet switch (thats a normal TAPR board with experimental software, allowing it to act as a packet switch - **not the NNC**). A current user having a normal TAPR1 or TAPR2 board and current software (no new ROMS required) can tie into the Level 3 network. The packet switch performs two functions then, acts as a PAD (accepting TAPR Level 2 frames and pushing Level 3 packets out to the network, receiving Level 3 packets back and giving them back to the TAPR users as Level 2 frames). This solution will be required in the near term, but ultimately, users should have PAD software in their devices instead of TNC software.

One Last Appeal for Standarization

I know there will be plenty of room for software in the modern packet switch. But the reason X.28, X.29 and X.3 were written originally was so that a user could travel anywhere in the world, walk up to a PAD, and use it. I know from experience that it takes a while to learn a new set of commands to do the same old thing you have been doing all along on your previous packet hardware. Why not relearn our packet commands just one more time? As pointed out by Doug in his paper, we can add commands that are just not covered in the protocols but are demanded by amateur radio's strange requirements. But when we want to show link status on all streams, let us use STAT, not CSTATUS. When we want to reset our Level 3 call, the command to do it is RESET, not CRESET (another command can be found for resetting the device,

warm and cold). Actually a study of the current TAPR commands reveal that most of the changing required to go to the X. protocols are centered in the parameter setting area.

Conclusion

Now is the acceptable time to become converted to the X. protocols for our amateur packet radio user interface. To deal with the enhanced network, we will all have to learn new commands anyway. Smart packet switches will insulate us from this for a while, but the ultimate answer is to do it in the standard fashion. We of AMRAD will try to do our part in software we provide with our PCPAD board, but that would not have the impact that changing the basic TAPR PAD software would have.