

Host Interface Spec for the ARDOP Protocol

Rick Muething, KN6KB John Wiseman, G8BPQ
Revised: 0.2 Oct 15, 2017

Please send questions, comments or suggestions to
<mailto:rmuething@cfl.rr.com>

1.0 Scope and Contents

This document is the Host interface specification for those wishing to use the ARDOP protocol. It is meant to cover (except where noted) the ARDOP Windows Virtual TNC and Firmware implementations of the TNC such as those using SBC (Teensie CPU) implementations of the TNC. It describes how the host program (Client or Host Application) interfaces to the TNC driver and the commands implemented. The ARDOP protocol specification document may also be useful in conjunction with this host mode interface specification. The interface may be via serial (RS232, RS422, Bluetooth etc. or TCPIP protocols.

Appendix A ARDOP Serial Interface Protocol..

These basic commands share some commonality with similar host mode commands for Packet or Pactor.

Appendix B SCS CRC Hostmode Description (See the PTC-IIIusb Manual Chapter 10 for details of CRC Hostmode)

Appendix C (Native ARDOP Commands)

Appendix D: ARDOP TNC Initiated Asynchronous Responses

Appendix E: TCPIP interface

2.0 ARDOP Protocol Name and Purpose

The protocol name is ARDOP (Amateur Radio Digital Open Protocol). This document details how a host program (e.g. a radio client or server) can interface to an ARDOP_Win modem or firmware/DSP implementation (TNC) that implements the ARDOP protocol. This document should provide sufficient detail to allow a programmer to write a compatible driver to either the ARDOP Virtual TNC or a firmware implementation of the ARDOP TNC.

3.0 Host to TNC Connection

The host to TNC connection is designed to work using one of three basic protocols:

- 1) TCPIP connection either on the same computer or over a WiFi or Ethernet connection. The TCPIP interface uses two ports one for Host to TNC commands and data, and one (next sequential port) for TNC to Host data.
- 2) Serial bit stream interface. Examples include RS-232, RS-422, USB using a virtual serial port. .

- 3) Bluetooth wireless serial interface. Bluetooth 2.0 to 4.0 (need further details here???)

4.0 WA8DED Host mode Host to TNC Protocol.

The host to TNC protocol will use an expanded WA8DED host mode. (see Appendix A)
This popular Host to TNC interface is used in many packet and in most Pactor modems in some form. The Preferred form is the Extended CRC Host mode interface as also supported by the SCS Pactor modems. This provide the highest security and recovery mechanism for serial data.

Intialization

The host program would normally use some initialization file (e.g. "ARDOP TNC.ini") that resides with the host program. That file should contain the minimum information to allow the host program to start the ARDOP TNC automatically. This includes:

TCP IP Control Port number used for commands (the data port is the next sequential port) and the Address for those ports (e.g. 127.0.0.1 or "localhost" if on the same computer)

Or

Serial COM Port and baud rate if using Serial interface

Or

Bluetooth interface details for pairing if the Bluetooth interface to the TNC is used.

If the ARDOP TNC is a virtual TNC (software + sound card implementation) it can be launched from a command line with the above necessary parameters which are sufficient to initialize the Host <> Virtual TNC link.

e.g. for TCPIP connection from Host to Virtual TNC:

```
Shell("ARDOP_Win.exe TCPIP 8515 127.0.0.1")
```

Would start the ARDOP_Win virtual TNC using a TCPIP port listening on ports 8515 and 8516 of address 127.0.0.1

e.g. for Serial (RS232 or USB) connection from Host to Virtual TNC:

```
Shell("ARDOP_Win.exe SERIAL COM5 38400")
```

Would start the ARDOP_Win virtual TNC using a serial port listening on COM5 baud rate of 38400.

e.g. for BlueTooth connection from Host to Virtual TNC: *(preliminary)*

```
Shell("ARDOP_Win.exe BLUETOOTH ARDOPWINTNC")
```

Would start the ARDOP_Win virtual TNC using a BlueToothl port pairing to "ARDOPWINTNC".

All remaining property values and setting may be supplied using the commands list or in some cases may use a separate ini file for ARDOP virtual TNC

4.0 ARDOP TNC User Interface

The host program normally would include a parameter setup menu with appropriate parameters for the ARDOP TNC (Virtual TNC such as the ARDOP_Win TNC or firmware implementation). There is also a limited set of Pactor commands implemented that will allow basic operation of the TNC using drivers intended for the Pactor PTC and Dragon modems by SCS.

Appendix A ARDOP Serial Interface Protocol.

(From John Wiseman)

Overview

The Host to TNC protocol used over Serial and Bluetooth links operates in two modes, Text or Hostmode. Text mode is primarily used for initialization and testing, but can be used for simple iterative QSO's using a standard ASCII terminal program. There is no error checking on the text interface. Hostmode is used for automatic operation, and provides error detection and recovery over the serial/Bluetooth line. ARDOP Hostmode is based on the SCS CRC Hostmode (as used in the PTC and Dragon Controllers) which is itself an extension of WA8DED Hostmode.

Operating Modes.

Either Protocol Mode can be used on one of two Operating Modes. ARDOP Native or PTC Emulation. The former supports the full ARDOP functionality, the latter enables ARDOP to be used with programs that support an SCS Pactor controllet, but don't have native ARDOP support.

ARDOP Native Mode.

ARDOP Native Mode is designed to be functionally similar to the TCP mode, to simplify host program and TNC support. Two DED Channels are assigned to the functions of the two TCP ports, Channel 32 for Control and Channel 33 for Data. A third channel (34) is used for debug information, as a standalone TNC doesn't usually have local storage for writing a debug log. The Protocol control channels (255 for polling, 254 for status and 253 for rig control) are used as defined by SCS. See Appendix C for a list of ARDOP Native mode commands.

PTC Emulation Mode.

This supports basic ARQ mode operation with software designed for Pactor. FEC operation isn't supported, but the Pactor Levels 1-4 are mapped to the ARDOP Bandwidths of 200 to 2000. Pactor Host Mode commands are converted to equivalent ARDOP commands, and ARDOP link status values mapped to equivalent Pactor values.

Protocol Overview

The protocol is polled master/slave, with a single bit sequence toggle to detect lost or duplicated frames. The host program is the master and the TNC the slave. The polling frequency isn't defined, but a maximum interval of 100mS is recommended to minimise latency.

The link is considered Idle when the master has received an in-sequence response to its previous transmission. The master can transmit at any time the link is idle. If it has data to send, it sends it, otherwise it sends a General Poll message. The slave will respond with a list (possibly empty) of channels with data available. The master then polls each channel in the list.

If the master doesn't receive a valid response from the slave in a reasonable time, it will repeat its last transmission. If it doesn't get a response after a reasonable number of retries it will report a link failure to the higher level software, which will abort any transmission in progress, then try to reestablish the link.

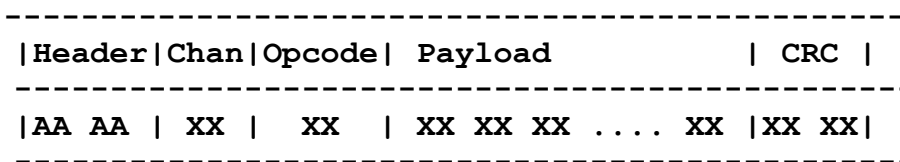
If the master receives an out of sequence response following a timeout and retry it will assume a delayed response from the slave, discard the repeat, and continue to wait for a valid frame. If it receives an out of sequence response at any other time it will assume a protocol error and reset and restart the link.

If the slave receives an out of sequence message it will assume that its previous response was lost, discard the message and resend the previous response unchanged.

Packet Formats

All packets have a two byte header of 0xAAAA and a two byte CRC-16 checksum on end. The top bit of the Opcode field is a sequence toggle.

The basic packet format is:



Payload can have two formats, either a Null terminated ASCII string or a Byte Count of 0 to 255 followed by 1 to 256 bytes of data.

There are two opcodes for Host to TNC packets, and eight for TNC to Host, though not all are used in ARDOP Native Mode.

From Host to TNC

Opcode 0 - Data

Opcode 1 - Command

From TNC to Host

- Opcode 0 - Response Success (no data follows)
- Opcode 1 - Response Success, followed by null terminated message
- Opcode 2 - Response Failure, followed by null terminated message
- Opcode 7 - Data, preceded by (length-1)

Channel 32 is used for Commands, Channel 33 for Data and Channel 34 for Debug information. Typical messages are shown below (control fields in Hex). Note that the Command format is used for hostmode protocol level commands. ARDOP commands (such as "ARQCALL") are sent as data on the Command channel. The Debug channel is for information to be written to a debug log.

General Poll

```
-----  
|Header|Chan|Opcode|Payload| CRC |  
-----  
|AA AA | FF | 01 | 00 47 |XX XX|  
-----
```

Response is a null terminated list of channels with available data. Value is Channel plus 1

```
-----  
|AA AA | FF | 01 | 21 00 |XX XX| Channel 32  
has data (0x21 = 32 + 1)  
-----
```

Poll to TNC Command Channel

```
-----  
|Header|Chan|Opcode|Payload| CRC |  
-----  
|AA AA | 20 | 01 | 00 47 |XX XX|  
-----
```

Response is an async ARDOP response message

```
-----  
|Header|Chan|Opcode| Payload | CRC |  
-----  
|AA AA | 20 | 07 | <0F>c:NEWSTATE ISS <0D>|XX XX|  
-----
```

ARDOP Command to TNC

|Header|Chan|Opcode| Payload | CRC |

|AA AA | 20 | 00 |C:MYCALL G8BPQ <00> |XX XX|

response

|AA AA | 20 | 01 |c:MYCALL now G8BPQ <00> |XX XX|

Data to be transmitted

|Header|Chan|Opcode| Payload | CRC |

|AA AA | 21 | 00 |<12>D:(Len)Message to Send|XX XX|

response

|AA AA | 21 | 00 | |XX XX|

Received Data from TNC

Data to be transmitted

|Header|Chan|Opcode| Payload | CRC |

|AA AA | 21 | 07 |<13>d:(Len)ARQMsg Received|XX XX|

This is no response to inbound frames, apart from the implied ACK of the next host frame having an inverted tog

Appendix B: SCS CRC Hostmode

From John Wiseman

See the PTC-IIIusb Manual Chapter 10 for details of CRC Hostmode .

WA8DED Packet Format

There are two data formats, null terminated or counted.

Byte 0	Channel Number
byte 1	Type Code (see table below)
byte 3-end	Null Terminated string (max length 255)
or	
byte 3	Info Length - 1
byte 4-end	Info (length 1 to 256)

SCS CRC Hostmode adds two byte of 170 to the front and a crc-16 checksum to the end. It also defines the top bit (bit 7) of the Type Code as a sequence toggle and next bit (bit 6) as a sequence reset flag. Data transparency is ensured by adding a null byte after any occurrence of 170 in the message or crc.

Command Codes

Host to Tnc		
CHANNEL	CODE	DESCRIPTION
-----	----	-----
n	0	Information (preceeded by length-1)
n	1	Command (preceeded by length-1)

Tnc to Host		
CHANNEL	CODE	DESCRIPTION
-----	----	-----
n	0	Success (nothing follows)
n	1	Success (message follows, null terminated)
n	2	Failure (message follows, null terminated)
n	3	Link Status (null terminated)
n	4	Monitor Header (null terminated)
n	5	Monitor Header (null terminated)
n	6	Monitor Information (preceeded by length-1)
n	7	Connect Information (preceeded by length-1)

Appendix C (Native ARDOP Commands)

These Native (to ARDOP) TNC commands can be entered via the WA8DED host mode by

Commands in **BLUE** are for a virtual (software) implementation meant to operate with local sound card hardware.

The following is the current list of commands with Explanation. What is between the “< >” are the command options separated by “|” or the response given in the reply to the command.

ABORT Immediately aborts an ARQ Connection (dirty disconnect) or a FEC Send session.

ARQBW <200MAX|500MAX|1000MAX|2000MAX|200FORCED|500FORCED|1000FORCED|2000FORCED>

Set/gets the bandwidth for ARQ mode. This sets the maximum negotiated bandwidth or sets the forced bandwidth to a specific value. Attempting to change bandwidth while a connection is in process will generate a FAULT. If no parameter is given will return the current bandwidth setting. This bandwidth setting applies to all call signs used (MYCALL plus optional call signs MYAUX)

ARQCALL<Target Callsign Repeat Count> Target Call sign must be a legitimate call sign syntax or “CQ” Repeat count must be 2 to 15 e.g. ARQCALL W1AW 5

ARQTIMEOUT<30-600> Set/get the ARQ Timeout in seconds. If no data has flowed in the channel in ARQTimeout seconds the link is declared dead. A DISC command is sent and a reset to the DISC state is initiated. If either end of the ARQ session hits it's ARQTIMEOUT without data flow the link will automatically be terminated.

AUTOBREAK <False|True>

Disables/enables automatic link turnover (BREAK) by IRS when IRS has outbound data pending and ISS reaches IDLE state. Default = True.

BREAK

Initiates a BREAK (link turnover request to the ISS) if in IRS state otherwise it has no effect. Forces ISS to clear its outbound queue and acknowledge with ACK. Normally not required if AUTOBREAK is enabled.

BUFFER

Gets the current outbound data buffer size in bytes. BUFFER is also sent asynchronously whenever there is a change in the outbound buffer size. The reported size includes any data that is currently in process (being sent but not yet acknowledged received by the Information Receiving Station, IRS)

BUSYBLOCK <False|True>

Disable/Enable Busy channel blocking. Busyblock will block a connection request unless there have been Tquiet ms of non-busy status preceding the connect request. (See appendix B) Default = False

BUSYDET <0-10>

Returns or sets the current Busy detector threshold value (default = 5). The default value should be sufficient for most installations. BUSYDET affects the sensitivity of the busy detector (low values = higher sensitivity but increased false triggering). A value of 0 disables the Busy detector (only should be disabled in emergency situations or very high local noise environments)

CAPTURE <device name>

Sets desired sound card capture device. If no device name will reply with the current assigned capture device.

CAPTUREDEVICES

Returns a comma delimited list of all currently installed capture devices.

CLOSE

Provides an orderly shutdown of all connections, release of all sound card resources and closes the Virtual TNC Program or hardware.

CMDTRACE<True|False> Get/Set Command Trace flag to log all commands to from the TNC to the ARDOP_Win TNC debug log.

CODEC <False|True>

Start the Codec with True, Stop with False. No parameter will return the Codec state.

CWID <False|True>

Disable/Enable the CWID option. CWID is optionally sent at the end of each ID frame.

DATATOSEND <0>

If sent with the parameter 0 (zero) it will clear the TNC's data to send Queue. Cleared data can be recovered using the RESTORBUFFER command. If sent without a parameter will return the current number of data to send bytes queued.

DEBUGLOG <False|True>

Enable/disable the debug log.

DISCONNECT

Initiates a normal disconnect cycle for an ARQ connection. If not connected command is ignored.

DISPLAY<Frequency in KHz>

Sets the Dial frequency display of the Waterfall or Spectrum display. If sent without parameters will return the current Dial frequency display. If > 100000 Display will read in MHz.

DRIVELEVEL <0-100>

Set Drive level. Default = 100 (max)

ENABLEPINGACK<True, False> Enable or Disable PINGACK reply to a PING targeted to MYCALL or one of the MYAUX Call signs. Default = True No parameter will return the ENABLEPINGACK state. (PINGACK only sent when in the DISC protocol state). Applies to both ARQ and FEC modes.

FECID<False|True>

Disable/Enable ID (with optional grid square) at start of FEC transmissions

FECMODE<8FSK.200.25|4FSK.200.50S|4FSK.200.50,4PSK.200.100S|4PSK.200.100|8PSK.200.100|16QAM.200.100|16FSK.500.25S|16FSK.500.25|4FSK.500.100S|4FSK.500.100|4PSK.500.100|8PSK.500.100|16QAM.500.100|4FSK.1000.100|4PSK.1000.100|8PSK.1000.100|4PSK.1000.167|8PSK.1000.167|4FSK.2000.600S|4FSK.2000.600|4FSK.2000.100|4PSK.2000.100|8PSK.2000.100|4PSK.2000.167|8PSK.2000.167|>

Sets the modulation mode and bandwidth for FEC (broadcast/multicast) transmission. Details on the specific frame types can be found in the ARDOP frame type spreadsheet. In general the first component of the frame ID is the modulation type e.g. 4FSK, 8PSK etc. The second is the bandwidth in Hz (@-26 dB). The third is the baud rate. Some modes also allow a shortened frame designated by a trailing "S". The above are in order of robustness for most channel types. Note baud rates over 300 are not currently permitted in the US below 29 MHz.

FECREPEATS <0-5> Sets the number of times a frame is repeated in FEC (multicast) mode. Higher number of repeats increases good copy probability under marginal conditions but reduces net throughput.

FECSEND <True, False>

Start/Stop FEC broadcast/multicast mode for specific FECMODE. FECSEND <False> will abort a FEC broadcast.

GRIDSQUARE <4, 6 or 8 character grid square>

Sets or retrieves the 4, 6, or 8 character Maidenhead grid square (used in ID Frames) an improper grid square syntax will return a FAULT.

INITIALIZE Clears any pending queued values in the TNC interface. Should be sent upon initial connection and before any other parameters are sent.

LEADER<120-2500>

Get/Set the leader length in ms. (Default is 240 ms). Rounded to the nearest 20 ms. Note for ARQ connections leader length is automatically adjusted based on the leader reported received by the remote station.

LISTEN <False|True>

Enables/disables server's response to an ARQ connect request to MYCALL or any of MYAUX call signs. Default = True. Also enabled the decoding of a PING frame to MYCALL or any of the MYAUX call signs in either ARQ or FEC modes. May be used to block connect requests during scanning or periods when server is offline or in the process of changing frequency.

MONITOR <False|True>

Enables/disables monitoring of FEC or ARQ Data Frames, ID frames, or Connect request in disconnected ARQ state. Default=True

MYAUX <aux call sign1, aux call sign2, ... aux call sign10>

Sets up to 10 auxiliary call signs that will answer ARQ connect requests. Call signs must be valid radio call signs and separated by commas. If sent with an illegal call sign (e.g. "MYAUX x" it will clear the MYAUX list and return a FAULT on the first syntax Error. If sent without a parameter will return a comma delimitate string of current MYAUX call signs. Legitimate call signs include from 3 to 7 ASCII characters (A-Z, 0-9) followed by an optional "-" and an SSID of -0 to -15 or -A to -Z. An SSID of -0 is treated as no SSID.

MYCALL <call sign>

Sets current call sign. If not a valid call generates a FAULT. Legitimate call signs include from 3 to 7 ASCII characters (A-Z, 0-9) followed by an optional "-" and an SSID of -0 to -15 or -A to -Z. An SSID of -0 is treated as no SSID.

PING<Target Callsign Repeat Count> Target Call sign must be a legitimate call sign syntax " Repeat count must be 2 to 15 e.g. PING W1AW 5. If the target callsign is not connected and decodes a PING and has ENABLEPINGACK = True and LISTEN = True it will reply with a PINGACK which includes the received PING S:N and decode quality. A properly decoded PINGACK will terminate the Ping. An attempt to PING if not in the DISC state will cause a Fault response from the TNC.

PLAYBACK <device name>

Sets desired sound card playback device. If no device name will reply with the current assigned playback device.

PLAYBACKDEVICES

Returns a comma delimited list of all currently installed playback devices.

PROTOCOLMODE<ARQ|FEC> Sets/Gets the protocol mode. If ARQ and LISTEN above is TRUE will answer Connect requests to MYCALL or any call signs in MYAUX. If FEC will decode but not respond to any connect request.

PURGEBUFFER Clears any data in the outbound buffer. Should precipitate a "BUFFER 0" asynchronous response. Data purged goes into a TNC temporary buffer that can be restored to the outbound buffer (outbound Queue) with a RESTOREBUFFER command.

The following RADIO commands support optional radio control that may be used in some applications (e.g. Virtual TNC is running on a remote computer located with the radio). All radio commands begin with "RADIO".

RADIOANT<0|1|2> Selects the radio antenna 1 or 2 for those radios that support antenna switching. If the parameter is 0 will not change the antenna setting even if the radio supports it. If sent without a parameter will return 0, 1 or 2. If RADIOCONTROL Is false or RADIOMODEL has not been set will return FAULT

RADIOCTRL<True|False> Enables/disables the radio control capability of the ARDOP_Win TNC. If sent without a parameter will return the current value of RADIOCONTROL enable.

RADIOCTRLBAUD<1200-115200) . (Note: baud rates >4800 recommended for PTT control to minimize T>R Latency)

RADIOCTRLDTR<True|False> Enable/disable DTR Line on Control port.

RADIOCTRLPORT<COMn> Set/get the radio control com port to use for radio control.

RADIOCTRLRTS<True|False> Enable/disable RTS Line on Control port.

RADIOFILTERBW<0|200|500|1000|2000> This sets (for selected radios) the filter bandwidth for the radio. A value of 0 will disable filter control. The actual bandwidth set in the radio is a function of the desired ARDOP bandwidth (200, 500, 1000, or 2000 Hz) + allowance for up to +/- 200 Hz tuning range centered on 1500 Hz. If a radio will not support the requested filter bandwidth then the next largest bandwidth will be used. A requested Filter Bandwidth. 0 (default) disables filter control.

RADIOFREQ<Frequency in Hz> If Radio Control is enabled in the ARDOP_Win TNC sets the Dial frequency of the radio and the display of the Waterfall or Spectrum display. If sent without parameters will return the current Dial frequency of the radio.

RADIOICOMADD<00-FF> Sets/reads the current Icom Address for radio control (Icom radios only). Values must be hex 00 through FF

RADIOISC<True|False> Enable/Disable Radio's internal sound card (some radios)

RADIOMENU<True|False> Enable/Disable the Radio menu item on the TNC Form.

RADIOMODE<USB,USBD, FM> Sets the radio modulation mode to USB, USB Digital (some radios) or FM (some radios). If sent without a parameter will return the current value of RADIOMODE.

RADIOMODEL<Radio Model> If radio control is enabled accepts the radio model. If sent without a parameter returns the current radio model. If radio not supported returns Fault.

RADIOMODELS<> Returns a comma delimited list of supported radio models.

RADIOPTT<CATPTT|VOX/SIGNALINK|COMn> Selects CATPTT,VOX(SignalLink or RigBlaster Advantage) mode or COM port for PTT Control

RADIOPTTDTR<True|False> Enables/disables PTT keying using DTR signal on RADIOPTT Com port

RADIOPTTRTS<True|False> Enables/disables PTT keying using RTS signal on RADIOPTT Com port

RADIOTUNER<True|False> Enable/Disable internal radio tuner (some radios).

End of Radio Commands.

RESTOREBUFFER<> Restores data from the most recent PURGEBUFFER command. Should precipitate an asynchronous "BUFFER nnnn" response.

SENDID

This will send an ID frame and if CWID above is enabled followed by a FSK CW ID. The protocol must be in the DISC state or a <Fault: Not from state...> will be replied.

SETUPMENU<True|False> Enabled/Disable the Setup Menu on the ARDOP_Win TNC main form.

SLOWCPU <True|False> Reduces DSP loading at some expense in sensitivity when True (Default = False)

SQUELCH <1-10>

Returns or sets the current squelch value (default = 5). The default value should be sufficient for most installations. Squelch affects the sensitivity of the leader detector (low values = higher sensitivity but increased false triggering). Typical range is 3-7.

STATE

Gets the current ARDOP protocol state <OFFLINE|DISC|ISS|IRS|IRStoISS|IDLE|FECSend|FECCv> Every State change is also reported asynchronously with the NEWSTATE reply below.

TRAILER<0-200> Get/Set the trailer length in ms. (Default is 0 ms). Rounded to the nearest 20 ms. Normally not required except for some SDR Radios.

TUNERANGE<0|50|100|150|200> Get/set the DSP tuning range in Hz. 0 should only be used in FM or AM radio connections.

TWOTONETEST

Send 5 second two-tone burst at the normal leader amplitude. May be used in adjusting drive level to the radio. If sent while in any state except DISC will result in a fault “not from state”

VERSION

Returns the name and version of the ARDOP TNC program or hardware implementation.

Appendix D: ARDOP TNC Initiated Asynchronous Responses

Asynchronous Responses

The ARDOP TNC codec will respond on the command port with possible asynchronous responses. All asynchronous responses begin with “c: and terminate in <Cr> + 2 Byte CRC” (No CRC used for TCPIP interfaces) There is no response expected or processed from the Host.

This is the list of the current responses:

BUFFER <data out queued>

Value is in integer bytes. BUFFER may also be polled using the BUFFER command with no parameters. BUFFER value includes any data frame currently being sent but not yet ACKed by the remote IRS.

BUSY FALSE

Clear channel detected

BUSY TRUE

Busy channel detected

CANCELPENDING Indicates to the host that the prior PENDING Connect Request or PING was not to MYCALL or one of the MYAUX call signs) This allows the Host to resume scanning.

CONNECTED <remote Call Bandwidth in Hz>

e.g. “CONNECTED W1ABC 500” An ARQ connection has been established. <remote Call> contains the connected call sign with a negotiated bandwidth. This follows the TARGET <target call sign> asynchronous response if the requested bandwidths are compatible and the connection negotiation is complete.

DISCONNECTED

An existing ARQ link has been disconnected

FAULT <description>

A program fault or error condition.

FREQUENCY<Frequency in Hz> If TNC Radio control is enabled the FREQUENCY command is sent to the Host upon a change in frequency of the radio. This can be caused by either a new programmed frequency, a Dial turn on radios with frequency read back or a mouse click on the Waterfall or Spectrum displays (in USB or USBD modes causing a retune). Does not return the new frequency when the command RADIOFREQ sent to the modem. The frequency reported is the DIAL frequency of the radio.

NEWSTATE reports any protocol state change.

Reply options: <OFFLINE|DISC|ISS|QUIET|IRS|IRStoISS|FECSend|FECrcv>

PENDING

Indicates to the host application a Connect Request or PING frame type has been detected (may not necessarily be to MYCALL or one of the MYAUX call signs). This provides an early warning to the host that a connection may be in process so it can hold any scanning activity.

PING<SenderCallsign> TargetCallsign S:NdB DecodeQuality> (space delimited) If the TNC receives a PING and is in the DISC state it reports the decoded Sender's call sign>TargetCallsign, S:N (in dB relative to 3 KHz noise bandwidth) and the decoded constellation quality (30-100). E.g." PING KN6KB>W1AW 10 95"

PINGACK< S:NdB DecodeQuality> If the TNC receives a PINGACK and is in the DISC state it reports the S:N (in dB relative to 3 KHz noise bandwidth) that the PINGACK send received and the decoded constellation quality (30-100). E.g. "PINGACK 10 80" A S:N value of 21 indicates the true S:N value > 20 dB. The PINGACK asynchronous reply to host is sent ONLY if a prior PING to this station was received within the last 5 seconds.

PINGREPLY If the TNC receives a PING to MYCALL or any of the MYAUX call signs is in the DISC state and has LISTEN=True and ENABLEPINGACK = True it will send a PINGACK frame and send confirmation to the Host with a PINGREPLY asynchronous reply.

PTT <True|False>

Indicates to the host application to key the PTT on (PTT True) or off (PTT False) To operate correctly the transmitter PTT should be activated within 50 ms of receipt of this response. Excessive delay in PTT application or removal may cause a failure in ARQ modes if measured latency exceeds 250 ms. Not sent if local TNC radio control is used.

REJECTEDBW <Remote Call sign> Used to signal the host that a connect request to or from Remote Call sign was rejected due to bandwidth incompatibility

REJECTEDBUSY <Remote Call sign> Used to signal the host that a connect request to/from Remote Call sign was rejected due to channel busy detection.

STATUS <status text > Used to signal information text to user. e.g. "STATUS CONNECT WITH W1AW FAILED". The syntax of STATUS commands is not rigidly controlled so if used should be for user display purposes only.

TARGET <target call>

Identifies the target call sign of the connect request. The target call will be either MYC or one of the MYAUX call signs. This command precedes the CONNECTED <remote Call> asynchronous response.

TUNE <Tuning offset in integer Hz> Sent when the waterfall or spectrum is clicked and can be used by the host to adjust radio settings. This is bounded to approx. +/- 1200 Hz. If radio control is enabled will also tune the radio to the new frequency.

APPENDIX E TCPIP INTERFACE

The TCPIP INTERFACE provides an alternate to the WA8DED serial host mode interface. It consists of two TCPIP channels on sequential ports. The first port is for the exchange of Commands the next sequential port is for the exchange of data.

TCPIP commands and data are not CRC protected as the TCPIP protocol itself has built in CRC and auto repeat if necessary. All commands are plain (readable) text using 7 bit ASCII character encoding. Commands sent from the host begin with "C:" and terminate with a <Cr> followed by a 2 byte CRC check for non TCPIP interfaces. No command name or parameter may include the strings "C:", "c:", or the <Cr> character. Commands that do not include parameters will cause an "echo back" of the command and current parameters as described below. Commands or replies from the TNC always begin with "c:" (lower case c) and end with a "<Cr>+2 byte CRC" (no CRC used for TCPIP connections). Commands and data are no longer acknowledged by a "RDY" reponse as in prior versions. All commands and data will respond with an echo of the command followed by the current value as below:

Host CMD Sent:	TNC Response:
Command sent to query current value: C:ARQBW<Cr>	c:ARQBW 2000MAX <Cr>
Command sent to change current value: C:ARQBW 1000MAX<Cr>	c:ARQBW NOW 1000MAX<Cr>
Response to command that neither accepts or returns a value: C:ABORT<Cr>	c:ABORT<Cr>

All commands and data are buffered and processed on a first in first out basis. Host programs should normally sequence commands based on the responses above. If a command is incorrect or received corrupted it will prompt a FAULT response allowing the host program to repeat the command or take alternate action. Command execution and reply is usually < 100 ms but a few commands may take a few seconds to execute.

When a command from the host is received with improper syntax or parameter values the Virtual TNC will reply with "c:FAULT <fault description + Echo back of command><CR>+2 byte CRC" A CRC fault (not used with a TCPIP connection) should prompt the Host to repeat the command.

If debug logging is enabled and Enable Command Trace is enabled each command will be logged as received by the Virtual TNC and each response will be logged to the ARDIO_Win TNC's debug log as sent. This is helpful in initial debugging the host application interface. Debug logging carries an accurate (1 ms resolution) time tag. Since Command tracing greatly increases the size of the debug log it should only be enabled when necessary as in the integration with a new Host program.

If the Host decodes a CRC Failure on any command or data it must reply with a CRCFAULT command.

7.0 Data transfer:

Data is transferred on the next sequential TCP address/port, or the same Serial COM port or Bluetooth pairing as commands using binary byte transfers with a specific header defined below.

The format for data going from the Host to the TNC is:

”D:” +2 byte count (Hex 0001 – FFFF most significant byte first), binary data,+2 Byte CRC”. (For TCPIP interfaces the 2 Byte CRC is not used since CRC and repeats are integral to the TCPIP protocol.) The “D:” identifies this as a data frame from the Host. Unlike earlier versions no RDY response is issued upon a successful decode. All data frames are buffered and processed in a first in first out basis. If the TNC received the data incorrectly it replies a CRC failure “c:CRCFAIL<Cr> + 2 byte CRC). (No CRC on TCPIP interface commands) The TNC will buffer this data in its outbound queue until transmitted. A failed CRC should prompt the Host to resend the frame up to 3 times before declaring an interface fault.

The format for data going from the TNC to the Host is:

”d:” + 2 byte count (Hex 0001 – FFFF most significant byte first), + “ARQ|FEC|ERR|IDF”, + binary data, +2 Byte CRC”. (For TCPIP interfaces the 2 Byte CRC is not used since CRC and repeats are integral to the TCPIP protocol.) The “d:” identifies this as a data frame from the TNC. ”ARQ” indicates this is data received from a connected (ARQ) session and should be error free. ”FEC” indicates this is data received from an unconnected session (a monitored FEC or monitored ARQ frame) and should be error free. ERR indicates this is Data received from an unconnected session which failed CRC check of the received frame . It can be masked or ignored by the Host if desired. ”IDF” indicates this is ID frame information. If the Host received the data incorrectly a CRC failure “C:CRCFAIL<Cr> + 2 byte CRC). The TNC will then respond with the next data frame or a repeat of the failed data frame. For TCPIP interfaces the 2 Byte CRC is not used since CRC and repeats are integral to the TCPIP protocol.

