

From rec.radio.amateur.digital.misc 831
 Path: ucbox!agate!doc.ic.ac.uk!uknet!pipex!uunet!europa.eng.gtefsd.com!library.ucla.edu!network.ucsd.edu!news-mail-gateway
 From: postmaster@asarin.ORG.AR
 Newsgroups: rec.radio.amateur.digital.misc
 Subject: Understanding AX.25 Packet monitored frames
 Message-ID: <gate.6cD0Dc1w165w@asarin.org.ar>
 Date: 8 Dec 93 16:25:28 GMT
 Organization: ucsd usenet gateway
 Lines: 167
 NNTP-Posting-Host: ucsd.edu
 Originator: daemon@ucsd.edu

UNDERSTANDING MONITORED PACKET FRAMES

Don Rotolo N2IRZ

Have you ever monitored the packet channel and wondered just what all those letters and numbers at the beginning of each packet "frame" mean ? Well, wonder no more ! This article will explain the basics of packet control.

All packet HDLC (High-level Data Link Control) frames are similar, consisting of 6 unique "fields":

| Flag | Address | Control | PID & Data | FCS | Flag |

FLAG : The Flag fields are put at the beginning and end of each frame to indicate the boundaries of the frame. A unique bit sequence is used (01111110) so other parts of the frame won't be mistaken for a Flag.

ADDRESS: This field normally specifies the destination of the frame. This is the actual call signs of the source, destination, and any digipeaters.

CONTROL: This field identifies the frame type. See FRAME TYPES below.

PID: The first byte of the Data field is the Protocol ID, specifying the type of protocol in use.

DATA: This is the actual data to be transferred. Many frames do not have a data field.

FCS: Frame Check Sequence, this is how errors are detected in a frame. Your TNC creates a number based on the rest of the fields, using a special mathematical formula. The other TNC does the same, and if the numbers match, the frame has no errors and is accepted.

We will examine a typical frame:

```
KD6TH-4>N2DSY-2*>N2IRZ [I;3,1]:0042z, 838 msgs, #40407 last @KD6TH-4 Mailbox>
<----Address-----> <-----> <-----Data----->
                          Control
```

Note that you do not see the Flag, PID and FCS fields.

FRAME TYPES:

C Connect frame. Also known as a 'SABM' frame (Set Asynchronous Balanced Mode).
 D Disconnect frame.
 DM Disconnect Mode frame.
 I Information frame.
 UA Unnumbered Acknowledgement frame.
 UI Unnumbered Information frame.
 RR Receive Ready command and response.
 RJ ReJect frame.
 RNR Receive Not Ready command and response.
 FRMR FRaMe Reject response.
 DM Disconnected Mode response.

When you type into your TNC a connect command, the TNC generates a frame like:

```
N2IRZ*>KD6TH-4 [C,P]
```

This means that N2IRZ is trying to connect to KD6TH-4. The * shows who is transmitting. The > shows the direction the frame is going (to KD6TH-4). The "C" indicates that it is a Connect frame, and the "P" means that the "Poll" bit is set. When the Poll bit is set, it means the originating station is

Polling (or "searching") for the destination station (sort of asking "are you there ??"). If the destination station was to respond, it would send an "F", or Final, bit in response ("yep, I'm here"). If the radio path is "perfect", the Poll/Final bits will rarely be used again.

KD6TH-4 is able to establish a connection to N2IRZ, so it sends:

```
KD6TH-4*>N2IRZ [UA,F]
```

This means that KD6TH-4 has acknowledged receiving N2IRZ's Connect request, and confirms the connection. Your TNC generates the familiar "**** Connected to ..." message for you. KD6TH-4 now sends a greeting (his "Connect Text"):

```
KD6TH-4*>N2IRZ [I,P;0,0]:Hi Don, Welcome to KD6TH-4 BBS !
```

The "I" tells us that this is an information frame. The information (Hi Don...) can be any text. P is for Polling - KD6TH-4 is just making sure N2IRZ is still there. What is interesting here are the numbers in the "[I,P;0,0]:" part: The first number is the next frame number that KD6TH-4 expects from N2IRZ, and the second number is the number of the frame that is being sent. The order of the numbers switches when N2IRZ is sending. Upon receiving this frame without errors, N2IRZ sends:

```
N2IRZ*>KD6TH-4 [RR,F;1]
```

Telling KD6TH-4 that N2IRZ got frame 0 OK and is ready to receive frame 1. KD6TH-4 sends 2 more info frames to N2IRZ:

```
KD6TH-4*>N2IRZ [I;0,1]:*** You have Unread Mail !!!
KD6TH-4*>N2IRZ [I;0,2]:0045z, 745 msgs, #40498 last @KD6TH-4 Mailbox>
```

N2IRZ didn't get the First packet without errors, but did get the Second, so it sends:

```
N2IRZ*>KD6TH-4 [RJ;1]
```

Which tells KD6TH-4 that it is rejecting the received packet: N2IRZ still wants packet number 1. If both packets were received OK, N2IRZ would simply send [RR;3] and the process would continue, but if N2IRZ hadn't received EITHER packet, no response would have been sent. Then KD6TH-4 would have started polling:

```
KD6TH-4*>N2IRZ [RR,P;0]
```

N2IRZ would reply

```
N2IRZ*>KD6TH-4 [RR,F;1]
```

Eventually, N2IRZ is finished reading the mail, so he signs off:

```
N2IRZ*>KD6TH-4 [I;5,2]:Bye
```

This is N2IRZ's frame 2, and he expects frame 5 back. KD6TH-4 responds

```
KD6TH-4*>N2IRZ [RR;3]
KD6TH-4*>N2IRZ [D,P]
```

First the "Bye" packet is acknowledged, and then KD6TH-4 initiates a disconnect. N2IRZ responds

```
N2IRZ*>KD6TH-4 [UA,F]
```

and they are disconnected. KD6TH-4 now sends out its "MAIL_FOR:" beacon:

```
KD6TH-4*>BBS [UI]: Mail_for: N2DSY WA2SQQ KA2USU W5GZT KB2BAV WA2SPO
```

This last frame is an Unnumbered Information frame. The place it is addressed to (in this case, "BBS") is specified by the UNproto command, which usually has the default of "CQ".

Now you have a good idea of what C, UA, I, RR, RJ, D and UI frames are like. A DM frame is sent when the other station is busy, or CONOK is Off, and your TNC generates a message that "KD6TH-4 is Busy", while KD6TH-4's TNC generates a "**** Connect Request: N2IRZ" message.

The other frame types are somewhat rare: RNR is sent when one station polls another ("are you there ?") but the other station isn't ready to process another packet yet. FRMR is only sent when all hell breaks loose: the Frame Type (I, UA, RR, etc) is either undefined or improper protocol (Wrong type of

reply).

If you would like to learn more about the AX.25 Protocol, or Packet Radio in general, I suggest the following:

Terry Fox, "AX.25 Amateur Packet Radio, Link-Layer Protocol Version 2.0", (Available from the ARRL), October 1984.

Jim Grubbs, "Get ***CONNECTED to Packet Radio", QSKY Publishing, Springfield, Illinois. 1986. (Easy to read, great bibliography)

Max Adams, "Basic Amateur Radio Packet", CQ Magazine, November 1985, pp.13-20.

This article is part of an informal series of technical articles intended to promote experimentation, good operating habits, safety and technical knowledge. This article may be copied freely, with proper acknowledgement. Feedback is welcomed.

```

* * * * *
* 73 de Don Rotolo * Knowledge is the only instrument *
* N2IRZ@kd6th.nj.usa * of production that is not subject *
* CIS : 73227,2644 * to diminishing returns -J.M. Clark *
* * * * *

```