The ROSE X.25 Packet Network MS-DOS Device Driver

Thomas A. Moulton, W2VY
J. Gordon Beattie, Jr., N2DSY

The Radio Amateur Telecommunications Society

INTRODUCTION

There has always existed a great barrier to the casual programmer (a.k.a. hacker) who wants to write a program that communicates with another system. The process of buffering and sending and receiving characters is tedious and wards off all but the very committed programmers.

The ROSE X.25 Packet Network MS-DOS Device Driver is going to redefine the rules as far as how hard it is to establish and maintain a connection through a packet network.

EXISTING METHODS

Presently there are a few methods a programmer can use to access the packet network. They are:

- INT 14 to a TNC

- INT 14 to a Virtual TNC

- INT 14 to a KISS TNC

- INT 14 to a TNC in Host Mode

Each of these have their own advantages and disadvantages, but they have one painful thing in common, INT 14, which is a simple character by character transmission and reception method.

There are also Terminate-and-Stay-Resident (TSR) programs that expand the functionality of INT 14, but there are also problems of incompatibility with other terminal programs.

The biggest problem is that there is no official method to access the COM ports from most programming languages. They do however provide methods for accessing "Files". What is needed is a scheme that lets the network appear to th.e user and programmer as a File.

MS-DOS DEVICES

It is a very straight forward process to set up a MS-DOS Device Driver to control a COM Port. There are two types of MS-DOS Devices, Block Devices and Character Devices. Both can be accessed as a File and read/written with standard high level language statemen ts.

A Block Device has a fixed record. length and must be formatted like a disk drive. This will not suit our needs.

A Character Device is a serial stream of information, can be named with up to an 8 character name, and MS-DOS does not place any constraints on the format of the information.

DEVICE ROSE:

To keep the interface as simple as possible all of the connection information should be described in the file name.

In a ROSE Network the user needs to specify the **Callsign** and Network Address of the destination station. For example the Command **"C W1AW V W2VY-3,203666"** will establish a connection to a station with the **callsign W1AW** at the Switch that provides coverage to Network Address 201666. To establish the same connection using the ROSE Device Driver, the user would open a file with the name **"ROSE:W1AW@203666"**. The user then may read and write the file to interact with the network. [Note: The terms "Read" and "Write" are not being used in the context of an editor reading in a file, but to indicate an interactive reading and writing of the connection to the remote user.]

In MS-DOS a program can have a large number of files open at a given time. The ROSE Device Driver also allows for this, in fact since it is a true Device Driver, it will automatically allow open files from different programs running under a multitasking environment.

EXISTING SOFTWARE

In order to be compatible with the existing software that has already been written for packet radio, the driver will also support the **MBBIOS/COMBIOS** interface via a simple Virtual TNC much like what was developed in the **G8BPQ** Switch. This interface will be provided to allow a simple method for users to use the Driver with existing software until the developers can modify their code to the new, simpler interface. This virtual TNC will also support KISS in later releases.

IMPLEMENTATION

The Driver will communicate to the outside world through a COM port to a ROSE X.25 Packet Switch asynchronous port or matrix of Switches using the Asynchronous Framing **Technique** (AFT).

Other configurations that will be supported include COM port to KISS TNC, as well as interfacing to the HDLC cards currently in use.

The code is being developed in C and is still underway. There are a lot of hooks in MS-DOS that needed to be trapped. The File Open, Close, Read, and Write interface is complete as well as the Timer Tick and Multitasking Interrupt. As of the writing of this paper, the COM port interrupt handler and the glue between the ROSE code and the Driver code need to be written.

CONCLUSION

The ROSE X.25 Packet **Switch**-based networks have been growing rapidly over the last two years due to their simplicity and rich functionality. With this new addition to the Network, application programmers with great ideas will be able to make valuable contributions to the **state-of**-the-art and enhance the services and pleasures of Amateur Radio packet network operation.