NETWORK ARCHITECTURE AND PROTOCOLS FOR A WIDESPREAD AMATEUR DIGITAL COMMUNICATION NETWORK

Douglas Lockhart, VE7APU
29 Shamokin Drive
Toronto, ON M3A 3H7 Canada

In the last couple of years the Vancouver Amateur Digital Communications Group (VADCG) programmable communication controller has been used in many areas of the U.S. and Canada. As one of those who worked on the development of the board and its software, I am very pleased to see that it has gained fairly widespread acceptance in the Amateur Radio fraternity. It was not so clear, a couple of years ago, whether or not it would be accepted because it involved the use of techniques unused in Amateur Radio at the time. My impression of Amateur Radio then was that it had a great deal of inertia or resistance to change. But at the same time, like a massive body, once it gets moving it has a large momentum. Now I believe that Amateur Radio is moving into digital communications, and nothing is going to stop it. We need only to guide it to the best system that we can. And what is the best digital communications system for Amateur Radio? I don't think that anyone knows. The design of a commercial digital communication network costs hundreds of millions of dollars. That's right! - just for the design, not for implementation. Yet, even after all this expenditure, most commercial systems have their problems and detractors. So, in spite of the small amount of money that Amateur Radio will be spending on network design, we may still be able to come up with a system equal to, or surpassing, commercial designs. With this in mind I will outline the general philosophy of the system that we are working on in the VADCG.

Firstly, we wanted a low-cost interface to the network for an end user. We felt that a user should not need to have a computer just to access the network. For this reason, we designed, produced and programmed the VADCG programmable controller. Of course, there were many other good reasons for going this way, but I am mainly trying to show the function of the controller in the network.

The network that we designed the board for was not intended to be homogeneous but a network in which nodes would have different functions. Some of the node functions identified were:

1. A 'Terminal' or 'End user' node. Typically, someone with only a teletypewriter or video terminal, although a user accessing the network through a microcomputer would also qualify in this category (i.e. an intelligent terminal).

2. A 'Gateway' node. A node which allows users on the network to access another digital communications system. Examples:

A gateway to the telephone system using an auto-answer/auto-dial Bell 103-type modem.

A gateway to a digital communications channel on a satellite.

A gateway to the Local vhf RTTY channel.

A gateway to another amateur digital communications network.

Note that if a node is used to interconnect two networks which have the same protocols, it should not be called a 'gateway' because, in this case, the two networks are actually only parts of one larger network.

3. A 'Repeater' node. Used to extend the coverage of the network.

4. A 'Logging' node. To record activity on the network to satisfy regulations as well as for performance analysis.

5. A 'Host' node. This is the computer system attached to the network and is usually the system that the end user wants to use. It contains the programs and files that the user wants to use, such as editors, games, compilers, assemblers, file-transfer programs, files of swap-and-shop information, mailing list, etc.

6. A 'Station' node. Coordinates the operation of the other types of nodes in the network. Provides network services and communication between the network and the end user, repeater, logging, gateway and host nodes. At present, the concept calls for all messages to pass through the station node, but this is not an absolute requirement in order for the station node to do *its* job. The station node provides the higher levels of network protocol that the simple end user cannot provide for himself because of the limitations in storage capacity and complexity in the end-user interface.

7. A 'Message-switching' node. This is a node which has sufficient storage capacity to be able to store messages and data for an extended period of time. Such

a node would be something like a computerized bulletin board system (CBBS). Information which could not be directly transmitted to its destination immediately could be left here to be sent onward when the destination node was available. The communication network is a packet-switching network and so has little storage capacity for messages. Messages are sent through the network only when both the source and destination nodes are available. The message switching node could have messages to be received by any user on request as well as messages intended only for a specific user.

As you can see, the station node has a much more complex task than any of the other node types. Furthermore, the station node becomes almost indispensable in a system designed to use it. Because of heavy reliance on this node, it should be backed up by another station node in the area or by a repeater node allowing communication to another area which also has a station node. The hardware for this station node should be fairly reliable because it involves no moving parts. The station node being used by the VADCG, for example, is a three-card S-100 bus system. One card is a standard CPU card, another is a 64-k dynamic memory card -- both of these are standard S-100 cards which are readily available from many suppliers. The third card is a special I/O card which the VADCG has developed for handling the special needs of the station node. The card has four channels of HDLC communication using the Intel 8273 chip and six interval timers. The interval timers are used to handle line timeouts and to simulate a time-of-day clock. The timers and the HDLC channels are all interrupt driven using 16 channels of vectored interrupts provided by two AMD9519 chips. Also using the interrupt structure is the power failure circuitry, the transmitter fault-detection circuitry and the circuitry to detect software failures or loops. A failure in any of these areas will cause the CPU to enter a program contained in up to 8 k of EPROM storage on the same card. This program allows up-line reloading of the station node software or down-line dumping of the station node software for analysis of software errors. Each channel has a choice of baud rates and can operate with either synchronous or asynchronous modems. A number of extra control lines for input or output are provided to control external devices.

Some of the functions and services provided by the station node are:

1. Establishment and termination of virtual connections between nodes in the network.

2. Communication with the end user in plain language. For example, the station node will provide an explanation of why a virtual connection could not be made. It will interpret and act on network commands submitted through a terminal keyboard. It will provide a list of the status of other users signed onto the local domain or provide a list of users in another domain, for example.

3. Drive a logging node to record the connection/disconnection of the users of the network giving times and dates as well as usage statistics.

4. Drive a repeater node so that the repeater will do intelligent repeating of frames. Not all frames received by the repeater should be repeated.

5. Provide the higher levels of protocol required for an extended network for the minimum end user system.

6. Make routing decisions and keep, dynamically, information on delay times. The routing system as planned will use a distributed delta-routing system allowing multiple paths for communication between station nodes -- something like the ARPANET routing scheme. Routing decisions will be based on minimal delay time. Changes in delay times detected by a station node will be passed to adjacent station nodes. New station nodes in the network will be integrated dynamically and will be deleted when communication is lost.

7. Communicate with non-end-user nodes in the network using concise coded or formatted network commands suitable for computer generation and interpretation.

The above is not a complete list of functions provided by the station node. Others will probably be incorporated as the system develops, but this list should give the idea of what the function of the station node is.

It should be noted that the above six types of nodes are not the only types possible but only the ones which we have identified as being the most important at the present time. Most functions can be identified as belonging to one of these six types, even though there may be occasions where there is an overlapping of functions. See Fig. 1 which should help to clarify the relationship of the nodes. Each station node has a 'domain' associated with it. The domain is the set of nodes that the station is providing services for. The domain is typically a geographical area such as a city, but different station nodes may operate on different frequencies in the same geographical area. The lines between the nodes on Fig. 1 represent logical communication links at the data-link level of communication. Not all possible communication links are allowed. Direct communication is allowed only between a station node and another node type or between two station nodes. However, a repeater node may be used as an intermediate node in this communication. Any messages sent between non-station nodes have to be routed through the station node in each domain. To some, this may appear as a harsh restriction on the communication possible, for, after all, there may be nodes in the domain that can communicate directly because of their proximity. To answer this let's look at the advantages of going through the station node and the reasons for communication with the station node.

1. Standardization of the radio link. Each node's equipment has to be set up to interface only with one point. This means that adjustments to the modem, power output of the transmitter, frequency adjustments of the transmitter, orientation of the antenna and various other requirements for establishing a communication link have to be set up for only one link. This avoids having a large amount of coordination with various other nodes. Once communication is established with the station node, no other concern for communication with the rest of the network is required.

2. Low power and directional antennas can be used for the link. No rotator is required, even if using a directional antenna.

3. Nodes which are out of broadcast range anyway would have to go through the station node.

4. Nodes which were using a different band would have to go through the station node to communicate.

5. Nodes which were using different speeds would have to go through the station node.

6. Nodes which required protocol translation would have to go through the station node. More on this later.

7. Nodes communicating outside of the local station node's domain would likely have to go through the station node.

8. All nodes using network services would have to communicate with the station node.

9. Establishment and termination of connections between nodes would have to be arranged through connection services in the station node.

The above considerations do not totally rule out the channel-sharing advantages of being able to have nodes communicating directly on the same channel as that of the station node when they are using the same speed and not requiring protocol translation and are within communication range. The protocol would have to be more complicated to allow these two types of communication to be carried out on the same channel and yet allow coordination by the station node. All I can say is that the present software does not coordinate communications on the channel which do not pass through the station node. However, the software does ignore all addresses which have not been assigned by the station node so that other digital communication can share the channel. It would probably be more appropriate that these nodes use another channel for their communication as they appear to have little need of the network.

You are probably wondering how the VADCG programmable communications controller fits into this network architecture because most users of the board are using software in the board which communicates directly from one end user to another end user. Well, in fact, this software which is commonly in use was written after the original software for the station node architecture had been written and was already in use. The terminal-to-terminal software is actually a modification of the original software to get it to work in a station-less environment. In fact, the hardware was limited to 4k of EPROM and 4k of RAM because the higher levels of protocol were going to be provided by the station node or by the host node. In spite of the general usage of the board for direct communication, it is still the intent of the VADCG to develop a network based on the station node concept. More circuits have been developed, and software has been written for use in this type of network recently.

As Fig. 1 shows, this architecture is distributed at the station node level but not at the lower levels. There are multiple communication paths between station nodes but only single paths between the station node and other nodes in a domain.

The VADCG board can be used in the terminal node, the repeater node, the logging node, the gateway node and the host node. However, it probably is not suitable for use in the station node due to limited memory and the fact that it is a single channel. With suitable programming, it could possibly be used as a type of front-end processor for the station node. The VADCG is developing separate hardware for the station node.

PROTOCOL LAYERS

THE PHYSICAL LAYER - This is the lowest level. It details the characteristics of the physical communications interface between the system components. We are adhering closely to RS-232-C standards in the use of connectors, pin assignments and voltage levels. But, in addition to the RS-232-C serial interface, we are providing a TTL-level parallel interface and a 20-mA current loop interface in order to accommodate the widest possible choice of end-user equipment.

THE DATA LINK LAYER - This layer manages the error-free transmission of frames over communication links between nodes in the system. Most communication networks are using a system very close to the HDLC standard as is the system that we are using. This protocol is the same as that being used in the VADCG programmable controller for direct communication now. Unlike IBM's SNA, which supports only an unbalanced version of HDLC, we are using a balanced version in which neither node at each end of the link operates in slave mode. Both nodes share packet transmission and recovery responsibility. When this layer receives a frame in error according to the frame check sequence contained within each frame, it requests the retransmission of that frame and all following frames. The reception of each frame is acknowledged, and if no acknowldgement is received, some transmission fault is assumed to have occurred, and corrective action is taken. This is usually an additional request for acknowledgement. If additional requests for acknowledgment fail, then the link is assumed to have failed, and other corrective action is taken. The protocol requires positive acknowledgment only after every 7 packets. The establishment of the link uses an initial connection protocol (ICP) in which information is exchanged between the connecting node and the station node. The connecting node passes a description of itself to the station node which keeps it in a table for the duration of the connection. The station node passes an assigned data link address to the connecting node which is used by both the connecting node and the station node for the duration of the connection. The protocol is half-duplex, multipoint and uses a carrier-sense technique (CSMA) to resolve contention on the radio channel and improve throughput. The contention protocol used by the station node is slightly different than that of the other nodes in order to give the station node an advantage when contending for use of the channel. This is done because all traffic in the domain must pass through the station node. The station node is working for all the other nodes.

THE NETWORK LAYER - This layer provides services which transport data through the network to its destination node. Messages that are transferred between domains in the network require a full network address and network flow-control functions. This information is added to the beginning of the packet as another block of information creating what I call a type 2 packet. The packets coming from a simple end user do not have this additional information and are in a type 1 format. These services are provided in the station node but may be provided by a multi-user host node. The decision to support type 1 or type 2 packets by a host node is indicated at the time of initial connection. When type 2 packets are selected, no translation of packets is done by the station and the management of the destination source address fields as well as management of the sequence number is left to the host node.

See Figs. 2 and 3 for the layout of the packets.

The following is an explanation of Fig. 4:

After receiving a packet passed to it from the next lower level (the data-link level), the packet is translated into a type 2 packet using tables kept in the station node. The packet may already be type 2 in which case this translation is not necessary. The packet is then analyzed to see what its destination is. If the packet is not for this domain, then it is routed back to data-link control. The router uses routing tables kept by higher layers to decide what link the data should be forwarded on. If the packet is for this domain, then it is either for network services or for another node in this domain. If it is for another node in this domain it is translated to type 1 if necessary and passed to the data-link layer. If it is for Network Services, then it is checked to see that it originated from an end-user terminal. If it did, it means that the data has been typed in using English words and must be parsed and analyzed by Terminal Input Services before being passed to Network Services for action.

As a result of the commands received by Network Services, Network Services may have control messages of its own to send to various points in the network. These control messages use codes suitable for interpretation by a computer. If they are to be sent to another domain, then they are sent via the router to Data-linkControl output. If they are for this domain and have to be interpreted by an end user (Terminal), they are passed to Terminal Output services which translates the codes to suitable English language sentences. The packet format is translated to type 1 if necessary before being passed directly to Data-link Control output. This technique has a couple of advantages. First, because a knowledge of the details of the characteristics of the terminal is kept in the domain's station node, Terminal output Services has all the information available to do fancy formatting of the message to the terminal. It knows the line length, whether the terminal supports lower case, highlighting, go to xy, erase screen, etc. This is not known at the remote Network Service point. Secondly, the computer format is more compact than the form put out by Terminal Output Services and so is more efficient at utilizing the longer communication channels.

Note that for every command that can be entered in through a keyboard by an end user, there is a corresponding coded command suitable for generation by a computer. Likewise, for every plain-language response to a command, there is a coded (or formatted) response for a computer program. This means, for example, that if there is a file-transfer program running in a host computer the file-transfer program can establish a virtual connection with another node using the network commands, transfer data across the connection and terminate the connection without human intervention. Host nodes are capable of establishing multiple virtual connections at the same time.

DEVICE SUPPORT

As mentioned earlier, the station node receives and holds information on the configuration of each connected node. This information is passed to the station at initial connection. In the case of a terminal node, this information contains details of the device characteristics and addresses in the node. When a connection is established between an application program and a device, the application program can request the device characteristic information from the station node. On the basis of this information, the application program can decide how to communicate with this device or even if it is capable of communicating with it. For example, suppose a user tried to use a full-screen editor program but had only a hard-copy ASR-33 terminal. The application program can send an error message to the user and disconnect. On the other hand, suppose the full-screen editor program found that it was communicating with a video display, then it would need to know how many lines and columns were in the display, whether lower case was supported, whether highlighting was supported, etc. The full-screen editor would then be able to communicate with the video display efficiently. This exchange of information binds the device and the application program it successful. There will be commands available to the end user to dynamically change the device characteristic information after connecting to the station node.

SUMMARY

I was hoping to be able to go into more detail on the routing, device support and packet formats in this paper but realize that each of these ought to be the subject of a separate paper.

The author feel that the station node concept of network development offers the most function for the least cost to the minimal end user. The specialization of function in the system prevents the waste incurred by duplicating the same code in every node. As new functions and services become available, they are instantly available to all users of the network. The routing decisions are made at the station node level, and the network is distributed at this level. This appears to be a reasonable tradeoff because the routing code is fairly complex and maintains a large amount of network information. Furthermore, there does not appear to be a simple distributed-routing system in the literature that is workable for the low-cost end user node. The many advantages that the station node offers appear to strongly outweigh the disadvantage of having to rely on it. In any case, we will have to rely on something if we are going to get our messages relayed across the continent reliably, and I am sure that Amateur Radio is going to have its own digital communications network operating across the continent before very long.

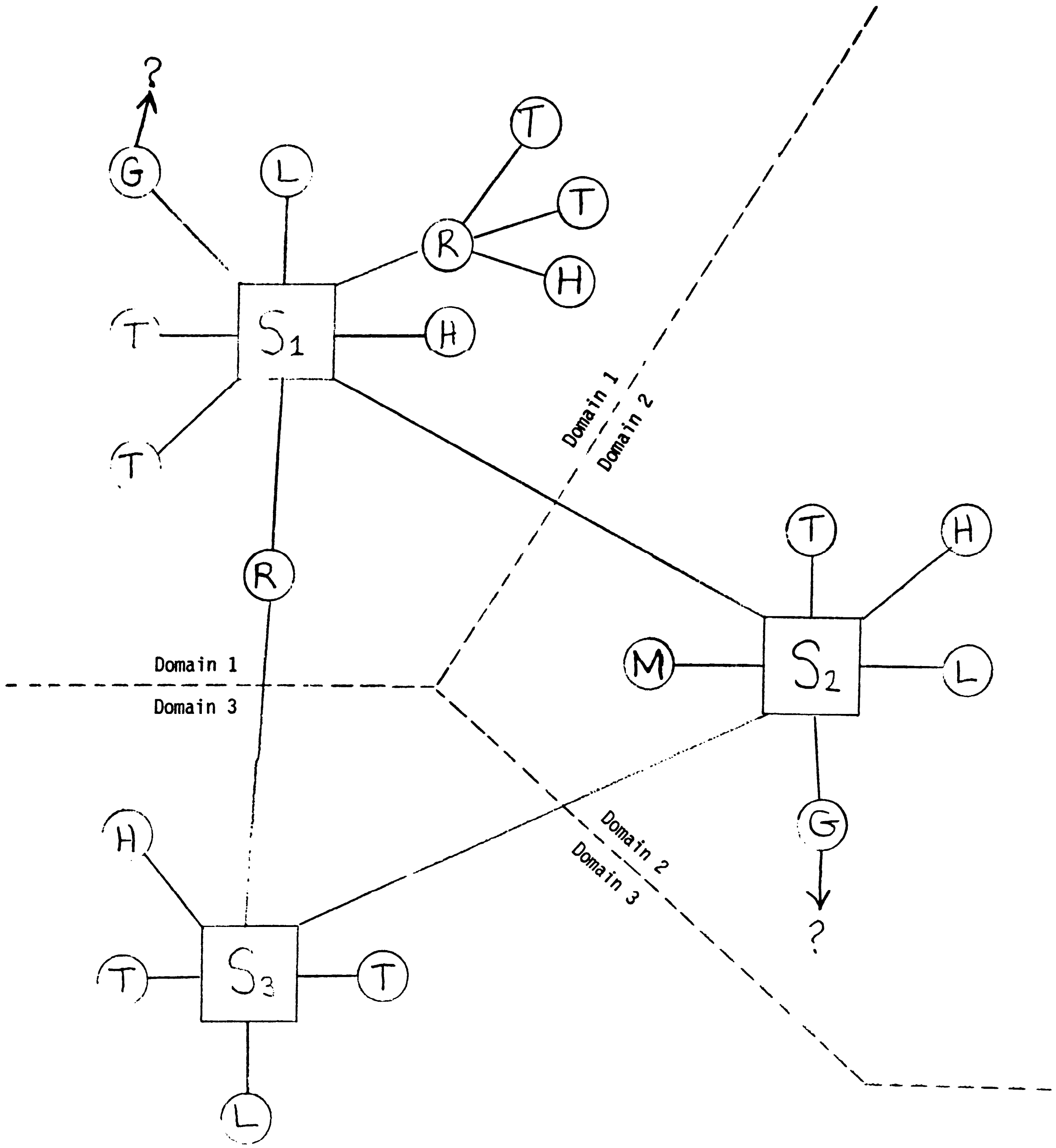
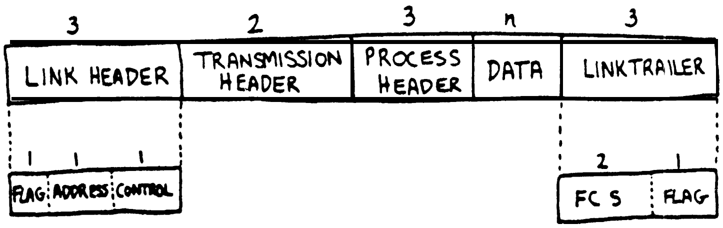


Figure 1. Relationship between nodes in a sample network.

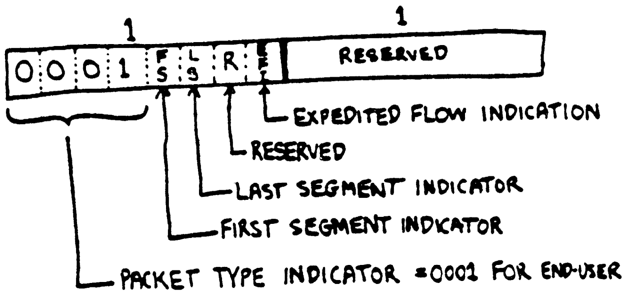
H = Host, S = Station, T = Terminal, G = Gateway, L = Logging, R = Repeater
M = Message Switcher ? = Unknown network or service

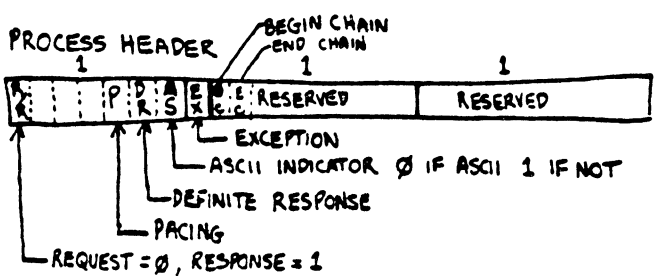
Fig. 2.
PACKET LAYOUT (END-USER TERMINAL NODE)



TRANSMISSION HEADER

Fig. 3.





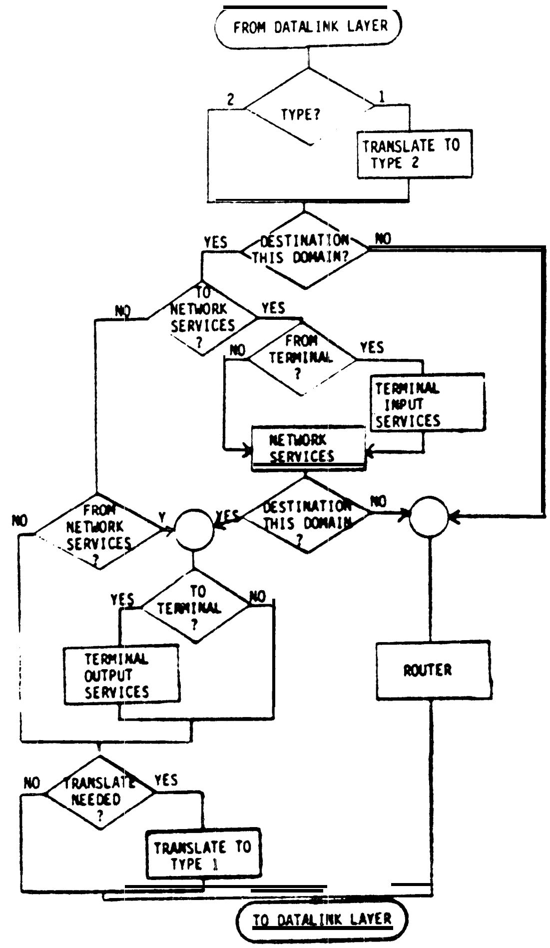


Figure 4. GENERAL DATA FLOW IN STATION NODE