Centrenet – A High Performance Local Area Network

R.N. IBBETT, D.A. EDWARDS, T.P. HOPKINS, C.K. CADOGAN AND D.A. TRAIN*
Department of Computer Science, University of Manchester, Manchester M13 9PL

Centrenet is a high performance local area network designed to satisfy the requirements of both closely knit multi-computer systems and communities of users spread across large campus areas. It uses high speed parallel switching nodes arranged in a tree-structured hierarchy with connections between nodes being implemented in optical fibre. Within each node is a Network Intelligence Module which assists in the setting up of virtual calls across the network and in maintaining network integrity. A pilot system has been implemented and further work is in progress to extend both the network and its capabilities.

1. INTRODUCTION

Local area networks have generally been developed in response to a need for resource sharing among a group of computer users concentrated in a small geographical area, typically within a single building. At the University of Manchester there are many groups of users concentrated in a number of buildings scattered about a large campus. Many have computing facilities of their own but virtually all require access at some time to the central facilities provided by UMRCC (the University of Manchester Regional Computer Centre). UMRCC, furthermore, is housed in the same building as the Department of Computer Science, which has traditionally been involved in the design and implementation of the hardware and software of large computer systems, and which is currently involved in the implementation of a large multi-computer system, MU6. Centrenet has therefore been designed as a system capable of satisfying the requirements, not only of a closely knit multi-computer system such as MU6, but also of a scattered community of users who wish to transfer files between their own machines and the central site, to gain terminal access to a variety of systems, and to share a variety of hardware and software resources.

This paper describes the philosophy of Centrenet and a hardware implementation, parts of which are operational. Other papers describing aspects of software associated with the network are in preparation.

2. DESIGN INFLUENCES

The design of Centrenet[1] has been influenced mainly by the need to satisfy the requirements outlined above. These requirements imply a need for high performance, the provision of service over a large campus area and, in order to enable terminal users to gain direct access to the network, the provision of 'intelligence' within the network. Two previously built systems may also be seen as having influenced the Centrenet design however. These are the MU5 Exchange[2] and the ARPA Network[3]. Centrenet is similar to the ARPA communication subnet in that terminals and host processors can be attached to it at switching nodes (IMPs in the ARPA Network), and packets are passed from source to destination along links which interconnnect the switching nodes. Centrenet is topologically different from the ARPA Network, however, and the switching nodes are not processors but high speed logic devices. Each node in Centrenet does actually contain a processor (or Network Intelligence Module (NIM) which participates in the setting up of virtual circuits for both terminals and hosts, in maintaining network integrity and in providing user services) but this processor does not stand in the path of normal terminal-host or host-host traffic passing through the network.

The influence of the MU5 Exchange can be seen in the design of the switching nodes. The Exchange was designed to provide a completely general and flexible interconnection scheme allowing for efficient implementation of a message based operating system distributed across the various processors and stores which made up the MU5 system (fig 1). Logically the Exchange was a multiple width OR gate operated as a packet switching system at the star point of the interconnection of up to 16 units. Each unit attached to the Exchange provided a set of parallel inputs to the OR gate, and each was connected, via its own buffer register, to the output of this OR gate.

This configuration involved only a very short common path for transfers between the various units, allowing a much higher data rate than would be possible with a distributed highway or bus system. Thus transfers through the Exchange occured at a rate of one every 100 ns, and each could involve a 64-bit data word together with address and control bits. For example, a processor wishing to read a word from a random access store attached to the Exchange (ie other than its own local store) sent the appropriate store unit number (4 bits), the required store address (24 bits) and control information to the Exchange. In each 100 ns time slot the Exchange examined all current requests for service, selected the one of highest priority, and in the next time slot routed that request through the OR gate to the appropriate output buffer. The read request in the example quoted here would then proceed to the store, with the unit number transmogrified to that of the requesting processor. Once the store had executed a read cycle it sent a further request back to the Exchange in order to return the 64-bit data word to the processor from which the read request originated.

In Centrenet each switching node (or Starpoint) is also a 16-port parallel switch which routes packets from source to destination according to a 4-bit address. In practice the technology of the switch is different from that used in the Exchange and the four address bits are taken

*Department of Computer Science & Electrical Engineering, University of Vermont, Burlington, Vermont, USA

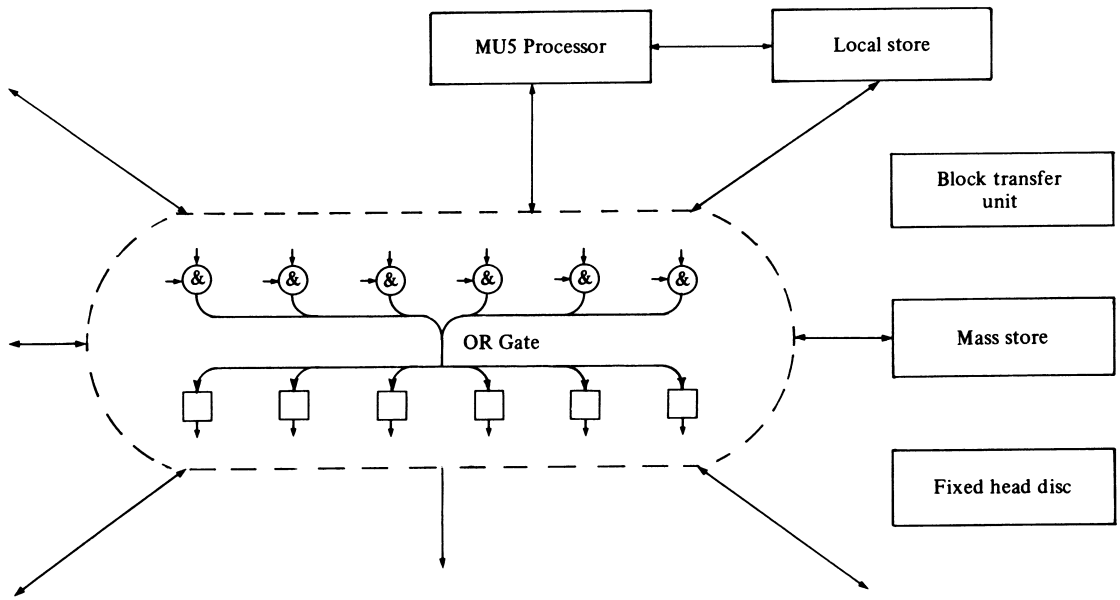

Fig 1. The MU5 Exchange

from within the packet destination address field (fig 2). Since there are 16 bits altogether in this address field the nodes naturally form a 4-level tree-structured hierarchy (fig 3). At a node at the lowest level in this hierarchy packets are routed from port to port according to the value of the 4 least significant destination address bits (bits 0 – 3) provided the 12 most significant address bits correspond to the 'address' of the node. If these bits are different the packet is routed via an UPLINK to the next node up in the hierarchy where bits 4 – 7 of the destination address field are used to route the packet, provided again that bits more significant in the destination address field correspond to the address of that node. Thus there is only one route between any two attached devices and packets only travel as far up and down the hierarchy of nodes as is necessary to reach their destination. The reliability afforded by the possibility, in the ARPA Network, of alternative routing has been sacrified in Centrenet for the high-speed switching of packets from port to port at a node.

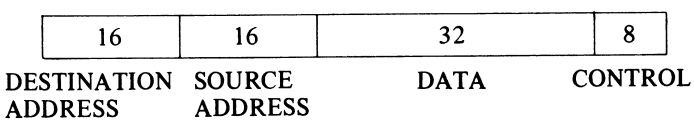

Fig 2. Centrenet Packet Format

The choice of 16 bits as the size of the address field in Centrenet was made as a compromise between minimising the packet size (in order to faciliate parallel switching) and providing for direct addressing of the numbers of computers and terminals which might reasonably be expected to exist on a large university or industrial campus, with the further constraint that the number chosen be a power of 2. The choice of 16 ports per Starpoint was made in the interest of modularity and uniformity of implementation throughout the network together with a number of engineering considerations. Thus the type of switching mechanism used in the Starpoint (Section 4) allows convenient partitioning of the system into one printed circuit board per Starpoint port with other boards acting as interfaces to processors, groups of terminals or interconnecting links. In such an implementation 16 ports can be conveniently accommodated in a single stardard rack.

3. PORTS AND SUPERPORTS

Terminals and processors are attached to the network via ports at the switching nodes. Whereas a terminal only requires use of a single input and output channel,

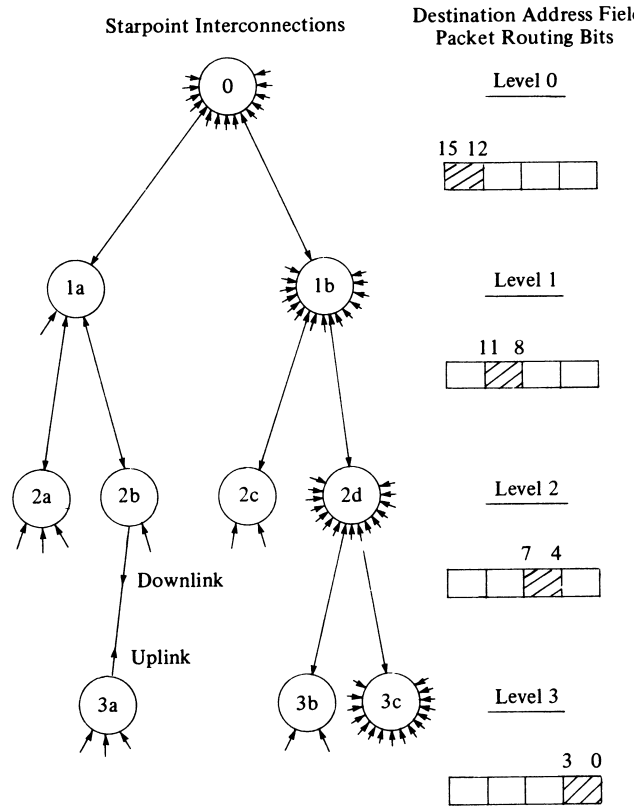

Fig 3. Hierarchical Structure of Centrenet

however, a processor normally requires a multiplicity of channels in order to support multiple on-line users connected via the network and multiple computer-computer information transfers. Connection of a processor to Centrenet is therefore normally made via a Superport, a device which gives access to a multiplicity of uniquely addressed network ports via a single physical connection. The Superport currently implemented is a 16 receive/transmit channel device suitable for attachment to a DEC PDP-11 Unibus. Because of the hierarchical nature of Centrenet, each port at the lowest level of the hierarchy corresponds to a single network address, while at the next level above each port corresponds to 16 network addresses. Thus terminals would normally be connected at the lowest level and Superports at the level above.

The nature of the traffic generated by terminals and computers is different, of course, and Centrenet offers two corresponding modes of operation. In Byte Mode a single character (to or from a terminal) is carried in the 32-bit data field, while in Block Mode all 32 bits are used. Indeed the use of a 32 bit data field represents a compromise between the 64-bit data field used in the MU5 Exchange (for performance reasons) and the need in Centrenet to handle single characters (in which case a 64-bit data field would be too wasteful). The Superport must be able to handle both these modes of operation, and so each channel may be set appropriately by means of mode bits in a status register.

Each channel acts as a direct memory access (DMA) device, controlled by its own set of registers (as shown in fig 4). In a Block Mode transfer a transmitting Superport channel accesses a sequence of words from the store of its host processor and transmits them through the network to a receiving Superport channel at a remote host. At the remote host the receiving Superport channel copies the sequence of words received from the network into the remote host's store. This mechanism is, in effect, an extension of the mechanism used in the Block Transfer Unit (BTU) attached to the MU5 Exchange to carry out paging and message transfers between the Local Store of MU5 and the Mass Store, an intermediate level of backing (core) store. The BTU carried out such transfers by first reading a word from one store (via the Exchange) and then writing it into the other. As each word was transferred the length count (which also formed the least significant part of each address by concatenation) was decremented, and when this count reached zero an interrupt was generated to indicate that the page transfer was complete.

In a Centrenet PDP-11 Superport each channel contains information relating to its local host allowing it to transmit to or receive from another similar Superport channel elsewhere in the network. A channel which is acting as a transmitter accesses from store the number of 16-bit words indicated by the Transmit Buffer Length register in sequence starting from the address held in the Transmit Buffer Start Address register and

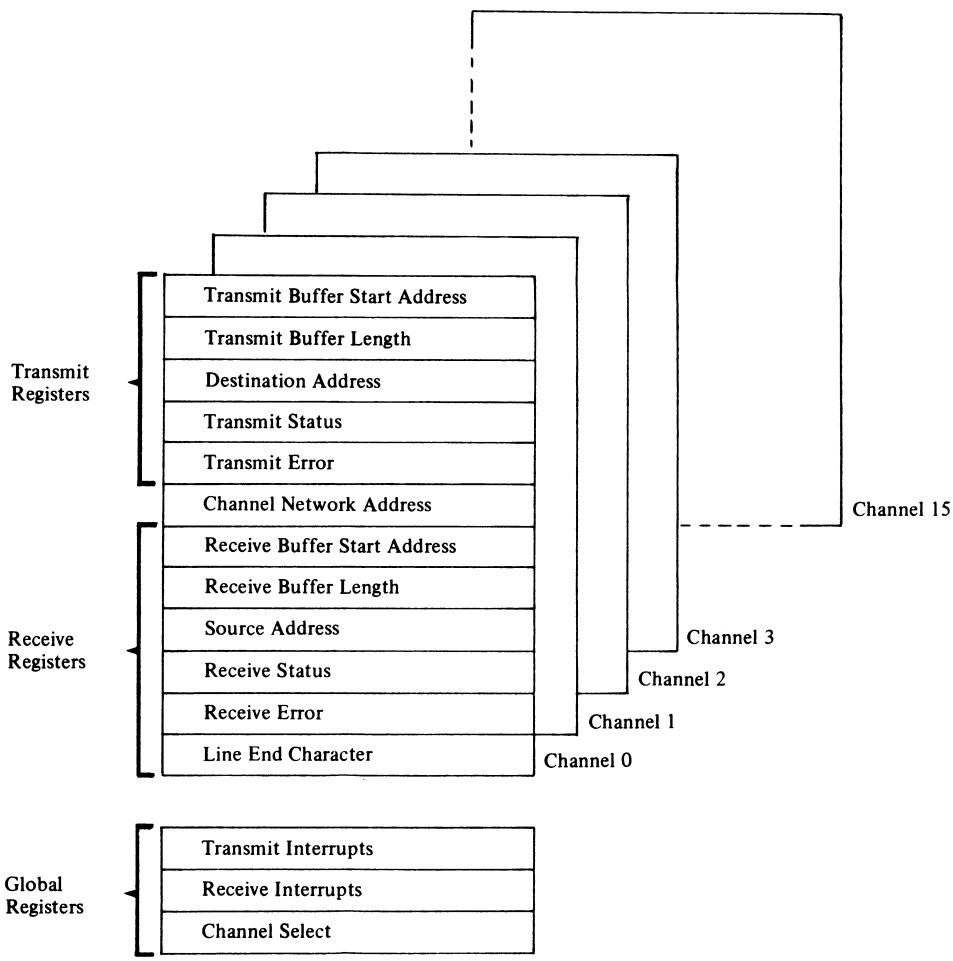

Fig 4. Superport Registers

launches them into the network (in pairs) in packets containing a destination address taken from the Destination Address register, and a source address taken from the Channel Network Address register. A channel which is acting as a receiver copies data words received from the network into store locations in its local host in sequence starting from the address held in the Receive Buffer Start Address register. When the last packet in a block is received (identified by a bit in the control field) the number of words received should match the number held in the Receive Buffer Length register. If it does not a Buffer Overflow or Buffer Underflow bit is set (appropriately) in the Receive Error register; overflow is always an error condition whereas underflow is not necessarily an error and the state of an Underflow Error Enable bit in the Receive Status register determines whether or not an interrupt will be generated. A transmitting channel may also (optionally) append a checksum to the end of a block. A receiving channel generates a similar checksum and compares it with the received checksum; if they are different and checking is enabled in the receiver an error is flagged in the Receive Error Register.

The Superport has two modes of operation corresponding to Byte Mode in the network; these are Character Mode and Line Mode. In each case the channel again acts as a DMA device transferring characters received from the network directly into store (and vice versa), but whereas in Character Mode an interrupt is generated in the processor for each character transferred, in Line Mode an interrupt is only generated on receipt of Carriage Return (or some other character defined in the Line End Character register) on input, or when the buffer length reaches zero on either input or output. For operating systems (such as MUSS[4]) which handle terminals on a line-by-line basis, this latter mode allows for more efficient use of processing power within the host.

The Superport is currently implemented as a microprogrammed LSTTL system made up of an arithmetic/logic unit, a 1K x 16-bit fast store containing the programmable registers, and interfaces to both the Unibus and Centrenet Port Card. This arrangement was chosen because it appeared to offer a good compromise between the performance of a dedicated hardwired design and the flexibility of a microprocessor implementation.

4. STARPOINT DESIGN

The most important consideration in the design of the Starpoint was performance. Accessing 32-bit words from the main store of a typical medium performance processor via a DMA mechanism takes of the order of 1 – 2 μs per word, and assuming that 8 processors can be sending and 8 receiving on a single 16-port Starpoint, the total required switching capacity is of the order of 128 – 256 Mbps. Data rates of this order are well beyond the capability of existing commercial local area networks, but are not uncommon in closely coupled multi-processor systems, and a number of alternative techniques were considered for implementation of the Starpoint. Among the more attractive were parallel rings, cross-point switches, bus systems and the MU5 Exchange.

Serial rings have been used as the basis of a number of local area network systems and in principle it would be possible to shrink a ring into a single cabinet with all stations in close proximity and with radial connections to attached devices. Placing the stations in such close proximity would then allow parallel rather than serial interconnections and the ring would become a 'barrel', ie a ring with thickness.[5] Such a system has been described elsewhere as a 'rotating bus'.[6] A barrel switch consists of a series of registers (the staves of the barrel) connected into a circle and operating with an 'empty slot' protocol. This system is, in effect, a circular version of a high-performance computer pipeline, and as in such a pipeline, the throughput rate is determined by the time taken for one stage to operate, although any one packet in the barrel is in general subjected to a number of stages of delay. With a relatively modest clock period of 100 ns, a 16-stave barrel switch handling Centrenet packets would have a throughput capability of 5 Gpbs of user data. Furthermore the system could be partitioned into one stave per printed circuit board allowing ease of construction. The disadvantages of the barrel switch are its reliability (all staves must be operational for the system as a whole to work), the fact that a full set of input and output connections is required for each stave, and the need either to populate the switch fully when not all staves are required, or else to re-wire the backplane when the number of staves in use changes.

A Starpoint based on either a cross-point switch or the MU5 Exchange would also be capable of providing the necessary throughput, but a cross-point switch would require a large amount of hardware and would be difficult to expand or contract, while in the case of a system similar to the Exchange, expansion or contraction would be virtually impossible once the system had been built, and partitioning of the system into modular form for construction would also be very difficult. Bus systems, by contrast, are easily partitioned into modular form and operate equally well when either fully or only partially populated. Furthermore a bus system can be constructed from relatively inexpensive LSTTL technology and operated with a clock period of under 200 ns to give a throughput capability of at least 160 Mbps of user data on Centrenet packets. Such a system was therefore chosen for implemention of the Centrenet Starpoint.

The switching mechanism in a Starpoint consists of up to 16 Port Cards (one of which acts as the NIM Interface) and an Uplink Card, all connected to a backplane interconnection bus (fig 5). Each card can load into its Input Buffer register a 72-bit parallel packet taken from the Packet Bus and can load a packet on to the Packet Bus from its Output Buffer register. The bus is controlled by clock and polling signals generated on the NIM Interface Port Card. When a card is polled it may load a packet onto the Packet Bus by enabling its Output Buffer. All other cards on the bus copy the new packet into their Input Buffer (unless they are still processing a packet received from the bus in a previous cycle) and then examine the Destination Address held in the Buffer. Bits in the Destination Address field are compared with bits of the address held in the Port Address register (initialised by the the NIM on power-up) to determine whether or not the received packet is intended for the card concerned. The number of bits taking part in the address comparison depends on the position of the

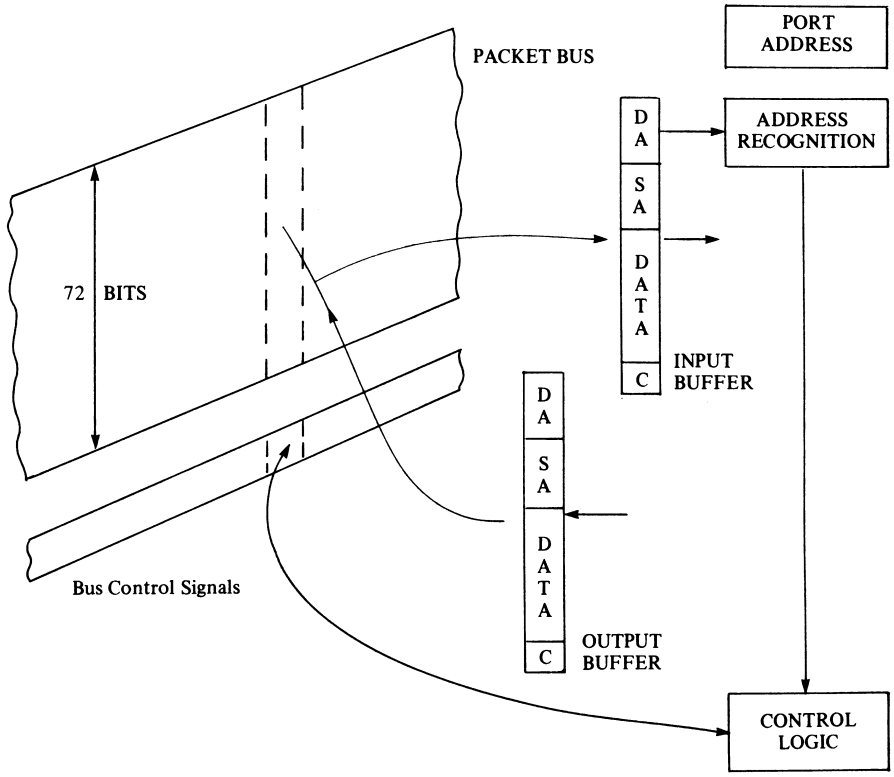

Fig 5. Starpoint Organisation

Starpoint in the hierarchy. At the highest level only the four most significant bits are compared on a Port Card, whereas at the lowest level all 16 are compared. On an Uplink Card the address comparison is slightly different; the least significant four of the destination bits being examined by the Port Cards on a given Starpoint are ignored by the Uplink Card, which tests for inequality between the remaining high order bits in order to determine whether it should accept the packet in order to send it to another Starpoint higher up in the hierarchy.

Port and Uplink Cards which determine that the packet is not addressed to them take no further action and remain free to receive another packet in the next cycle. A card which determines that a packet is addressed to it returns an acknowledge signal to the Packet Bus after a fixed interval of time from first receipt of the packet, at which point the card which sent the packet expects to receive an acknowledgement and hence notes that its packet has been delivered. If the card to which the packet was addressed was already busy when the packet was sent (and therefore did not copy the packet into its Input Buffer), no acknowledge is received at the appropriate time and the sending card re-sends the same packet when it is next polled. In order to prevent the network from becoming blocked in a fault situation, the number of such re-tries is strictly limited and the sending card eventually times out and activates a fault report signal on the bus when it is next polled. This notifies the NIM of the fault condition. At the same time the device attached to the card, and from which the failing packet was originally received, is also notified of the fault.

In the current prototype Starpoint the polling algorithm is simply a round-robin scheme based on a 17-state counter. Each card receives a unique polling signal in turn, but the NIM can selectively disable any one or more of these polling signals (except its own) via a mask register on the NIM port card. This facility is essential during network initialisation, for example, when the NIM sets up each card in turn (via a common 'NIM Override' control signal) with its proper source address, and in some cases a default destination address (usually that of the NIM itself). It is intended that a more sophisticated polling mechanism be introduced, however, involving the use of a larger counter which will address a store of, say, 256 locations, loaded by the NIM and with each location containing a port card number. Now locations in the store will be read sequentially, but the level of indirection introduced will allow the port card polling sequence to be varied. This will allow investigations to be made of adaptive polling techniques, in which some cards might be polled more frequently than others.

The interface between a port card and an attached device is made up of a 72-bit parallel bi-directional packet bus connection, together with additional control and handshaking signals. The direction of information flow across this bus is determined by logic in the port card which can copy a packet from the device into the Packet Bus Output Buffer in response to a 'Transmit Request' signal or can send a packet to the device from the Packet Bus Input Buffer, accompanied by an 'Incoming Packet' signal. Within each device there are again two registers, one acting as an input buffer receiving packets from the port card and one as an output buffer sending packets to the port card. The use of a common (bi-directional) interconnection bus reduces the number of physical connections required so that the Packet Bus, device interconnection bus and the various control signals can

all be accommodated on three standard Eurocard connectors (each with 64 signal and 32 earth connections).

5. LOCAL AND REMOTE LINKS

A Port Card may be connected to any one of a number of devices, including a Superport, a terminal multiplexer or a serial link card (allowing access to another Starpoint). Uplink Cards are always connected to serial link cards giving access to Starpoints higher up in the hierarchy, and clearly there must be an equal number of Port Cards in the system acting as Downlinks to Starpoints lower in the hierarchy. These high performance links may operate over distances of up to several kilometers and are referred to as Remote Links in Centrenet. There is also a requirement for lower performance links operating over shorter distances, as between a Starpoint and a PDP-11 Superport, for example, where the distance of the Superport from the Starpoint, which may be serving a number of processors, is greater than that which can be accommodated by a long Unibus cable. In this case a Local Link is interposed between the Superport and the Port Card to which it is nominally attached. Both types of link have been implemented in optical fibre and co-axial cable.

A Remote Link joins two high data rate switches and in a tree structured hierarchical network, the funnelling effect which results from sending data from many sources near the bottom of the hierarchy up to the top of the hierarchy would seem to imply the need for increased performance at the higher levels. However, it is anticipated that in a typical Centrenet configuration much of the traffic will be localised in such a way that it will only travel via one or two Starpoints and although there is no reason in principle why higher performance Starpoints should not be used at higher levels, there are good arguments for using identical technology throughout. Switching capacity at the lower levels may therefore be under utilised. In the current implementation each Port and Uplink Card requires around 2.5 μsec to service a request (and is therefore in principle able to respond to every 17th 200 ns Starpoint bus cycle without difficulty) and the Remote Link is therefore required to transmit a packet serially in a comparable time if it is not to become a bottleneck in the system. This implies the need for a high bit rate over the link (at least 28.8 Mbps to transmit 72 bits in 2.5 μsec) and as a consequence, over the potentially long distances envisaged, the packet time becomes shorter than the time of flight. In these circumstances one-at-a-time packet transmission with end-to-end acknowledgement across the link leads to poor link utilisation and a packet windowing mechanism has therefore been implemented. Thus the transmit section of a Remote Link (fig 6) can send out up to eight packets in succession before receiving an acknowledge from the first, and because the Starpoint at the far end cannot guarantee to accept these packets from the Remote Link as they arrive, the receive section of a Remote Link must provide buffering for these eight packets. Buffering is also advantageous in the transmit section since it allows fluctuations in the flow of packets from the Starpoint into the link to be smoothed out and, by buffering unacknowledged packets, a re-try mechanism can be implemented in the event of a faulty packet being detected at the far end. This re-try mechanism implies a further requirement on the Remote Link, ie error detection.

Within a Starpoint packets are handled entirely by digital logic which is sufficiently reliable for error detection to be unnecessary. The clock recovery and data detection circuitry within the links is essentially analogue, however, (and therefore subject to higher error rates), and where co-axial links are used packet corruption by external influences along the length of the links is also possible. Error checking on the links therefore becomes essential and this requirement, coupled with the use of a packet windowing mechanism, led to the adoption of a link protocol closely related to HDLC. Thus each Centrenet packet transmitted along a serial link is contained within a framing envelope, as shown in figure 6, and all information within the region delimited by the flags is bit-stuffed in order to ensure uniqueness of the flag. The total packet length is thus 112 bits (or more, with bit-stuffing) and at 40 Mbps (the bit rate used on the Remote Link) the time duration of a packet is between 2.8 and 3.2 μsec, closely matching the Port Card packet servicing time. The use of an eight packet window protocol with packets of this duration allows gap-free transmission over distances of up to 2.3 Km.

The logic circuitry within a Remote Link is implemented largely in LSTTL technology, but with some of the control logic implemented in STTL, and a double-banked shift register serialiser is used to obtain the required speed. Within the optical fibres efficient use of bandwidth is required since the links are implemented using standard LEDs and PIN diodes operating close to their maximum frequency. Furthermore, sufficient timing information is required for clock extraction and ideally the transmitted signal should contain a minimum of DC component since the receiver is AC coupled. An NRZI-S code was chosen in which a 0 is represented by a transition and a 1 by the absence of a transition. This code gives efficient use of bandwidth while the bit-stuffing inherent in the HDLC-like protocol (in which an extra 0 is inserted following five successive 1s) gives sufficient timing information and minimises the DC content. More complex codes such as MFM and a group coding scheme were examined in detail, but these are much[7] more complex to implement and are difficult to resynchronise in the event of data loss. The use of NRZI-S coding requires the link protocol to differ from HDLC in respect of the pattern transmitted in idle periods between packets. HDLC allows the transmission of either all 1s or flags (01111110) between packets. The former results in a complete absence of transitions in NRZI-S code, and thus cannot be used, while the latter represent the worst case allowable pattern with respect to DC component. An idle pattern of all 0s, which gives regular transitions and has a mean DC component level of zero, is therefore used instead.

Local Links also operate via optical fibre or co-axial cable but have been designed for lower cost and complexity, and therefore have lower performance. No buffering is included so that packets must be acknowledged individually, but a CRC is still appended to allow detection of corrupt packets. Such packets are discarded and the end-to-end data link level protocol (section 7) is therefore required to be able to recover from this

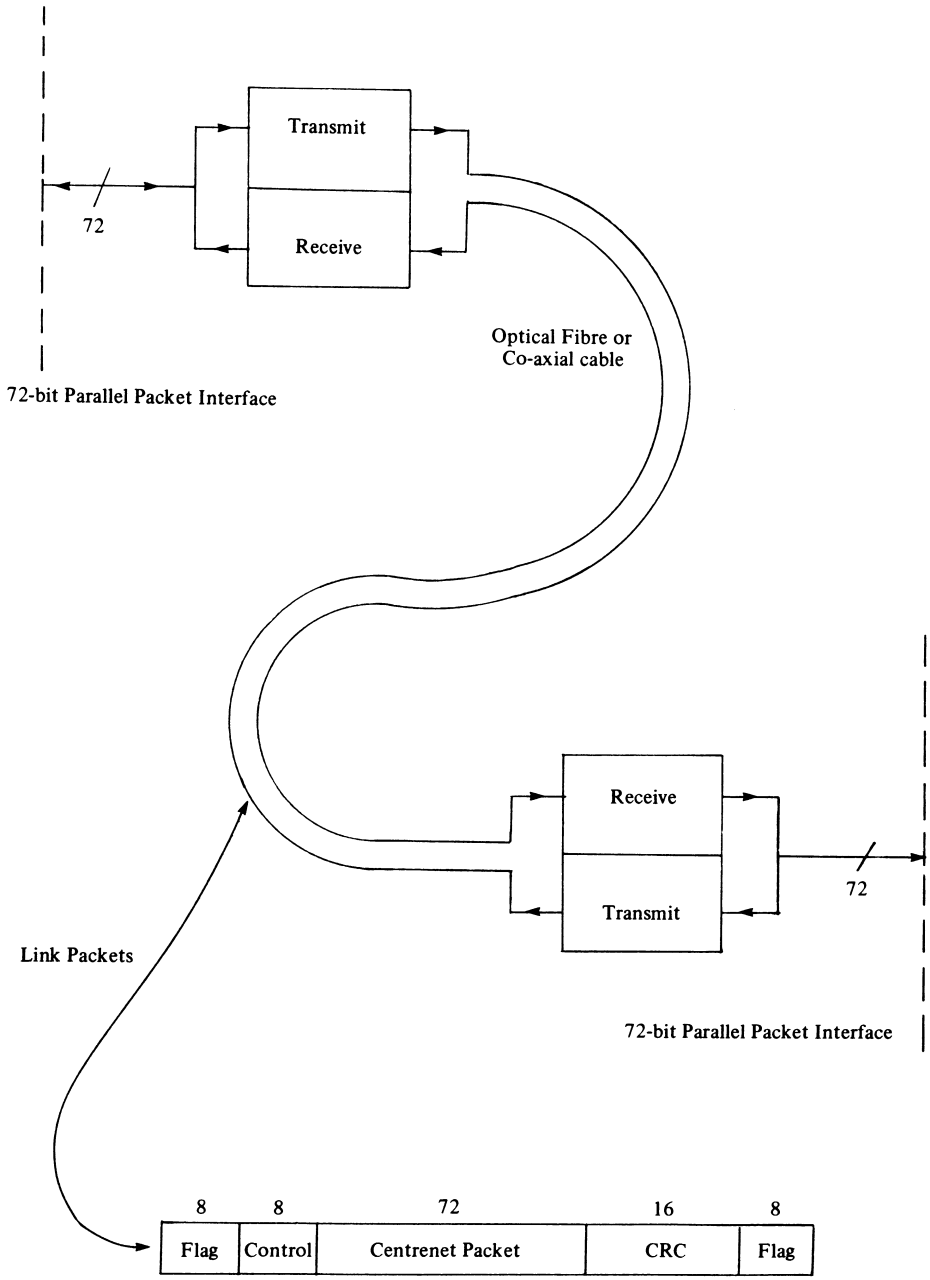

Fig 6. Remote Link Organisation

situation. Originally the same coding and bit stuffing scheme was used on the Local Link as on the Remote Link for reasons of compatibility. While this simplifies the coding logic, extra complexity is needed in the packet assembly logic. However, the use of flags and bit stuffing is not strictly necessary if the packet length is fixed. For this reason a simpler version of the Local Link has been implemented which avoids the need for bit stuffing. A self-clocking code is now required, however, and a phase encoding scheme has been adopted. The Local Links run at 15 Mbps, so that the extra bandwidth requirements of this code are not a problem.

6. THE NETWORK INTELLIGENCE MODULE

The Network Intelligence Module (NIM) contained within a Centrenet Starpoint serves a number of different purposes. At a low level the functions it performs are initialisation and error recovery. The NIM resets the Starpoint hardware at power up and initialises each port card with its own network address. It may also intialise device interface hardware attached to a port card. Any errors detected by Starpoint ports, links or device interfaces are reported to the local NIM, which may itself communicate such information to other NIMs. At a higher level the NIM is also able to interact with users of 'dumb' devices such as terminals, in order to determine user requirements. The NIMs also assist with the setting up and disconnection of virtual circuits across the network, and various other high-level functions such as name serving, the tables for which are distributed throughout the NIMs.[8] In order to perform these various tasks the NIM is implemented as a single board computer, connected to the Starpoint via the special NIM Interface port card which allows the NIM not only

to transmit and receive packets, but also to act as a controller for the Starpoint Bus and other port cards in the Starpoint.

The NIM requires a variety of memory types. During software development it is convenient to load the program under test into read/write memory so that the code may be readily modified. Basic operational software must be permanently loaded, however, and this is therefore more suitably contained in read-only memory. There is nevertheless a continuing requirement for read/write memory in an operational environment, not only for use as working space, but also because it is anticipated that higher level software for the NIMs will be subject to regular development and updating. This software can itself be transmitted to the NIMs via the network from the NIM at level 0 in the Centrenet hierarchy. Each NIM is also required to maintain a record of the network addresses to be loaded into the ports on its Starpoint during initialisation, together with other information specific to the devices attached to the ports. This information must be retained when power is removed, but must be readily updateable when changes are made to the hardware configuration. Low power read/write memory with a battery back up power supply is used for this purpose.

As currently implemented the NIM is a purpose built Z80 computer system. Alternative implementations can be substituted without difficulty, however, since the interface between the NIM and its NIM Interface port card is quite straightforward. It consists of a 16-bit bi-directional data bus used for loading and reading the various registers in the NIM Interface (including the various fields within the receive and transmit packet registers), and a set of 16 control signals. These are handled in the current implementation by a pair of parallel input/output (PIO) integrated circuits (fig 7).

The NIM also has provision for two terminal connections via asynchronous serial lines. This allows a terminal to be attached directly to the NIM to assist in program development during commissioning and in updating information during normal operation. The second line allows connection, during commissioning, to a separate disc-based 'host' computer in which NIM software can be generated and from which this software can be down-line loaded.

Memory in the NIM is split up into sections and a simple memory mapping scheme is used to extend the available address space. The processor address space is divided into 8 Kbyte blocks, and each block except the least significant has an associated 4-bit memory mapping register. When the processor addresses a memory location, the three most significant bits of the address (unless they are all zero) select one of the mapping registers and the value in this register selects one of sixteen 8 Kbyte blocks. The mapping registers themselves appear in the processor's I/O address space and are intialised through code held in the least significant memory block. This block is implemented in EPROM and contains not only the hardware intialisation code but also a simple monitor to control the down-line loading process. Eight blocks are implemented using 64K x 1 dynamic RAM circuits, one contains battery backed-up CMOS RAM, while the remainder are partially populated with EPROM.

7. NETWORK PROTOCOLS

The protocols used in Centrenet form a hierarchy in which there are two distinct bands separated by a transport service interface (fig 8). Above this interface are the high-level, network independent protocols while below it are the low-level, network dependent protocols. This strict division allows the development of higher level

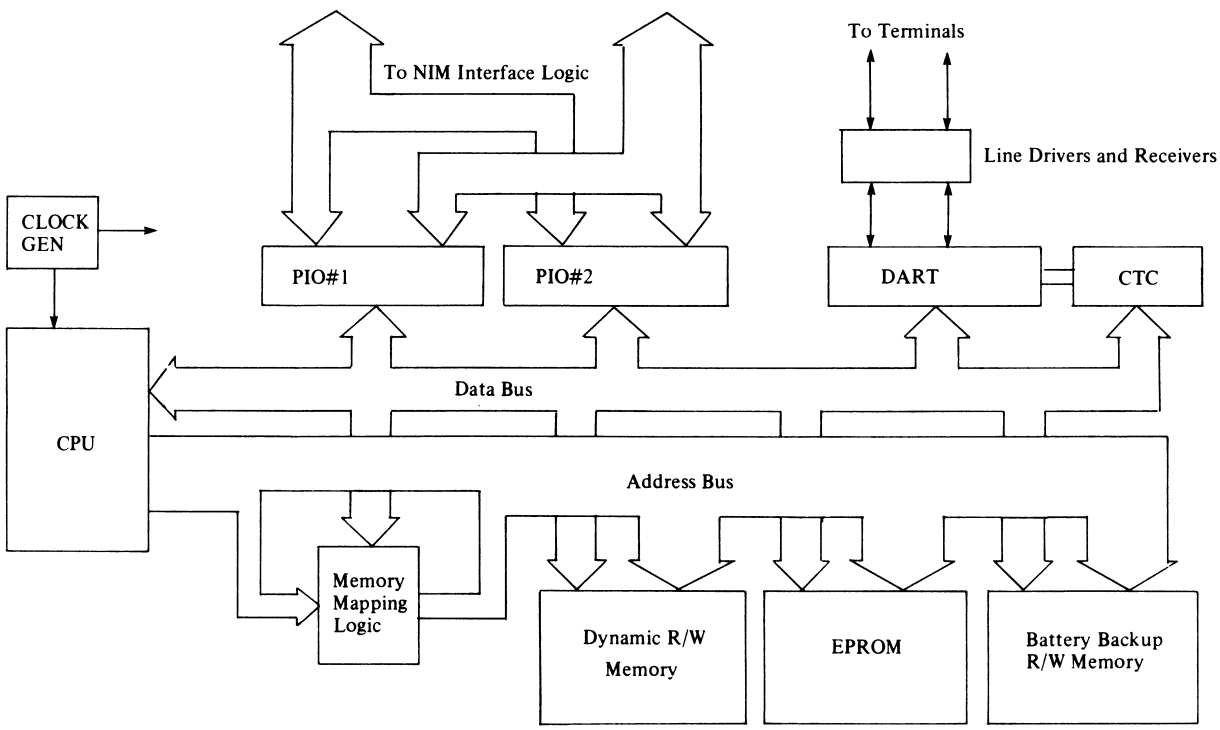

Fig 7. NIM Hardware

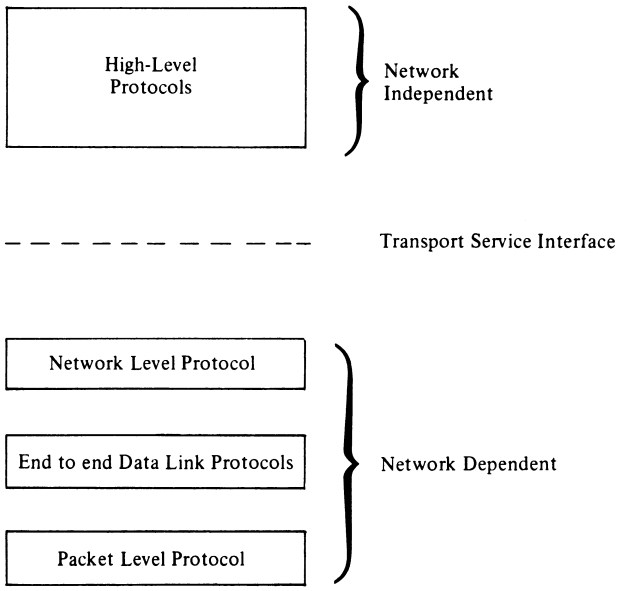

Fig 8. Centrenet Protocol Hierarchy

protocols (equivalent to layers 5 – 7 in the ISO Open Systems Interconnections (OSI) model[9] to be based on a single well defined interface and thus to be isolated from changes in low-level software and hardware. Furthermore, because this interface is based on the 'Yellow Book' transport service standard,[10] high level software used in Centrenet will be compatible with that used in other networks in the UK academic community.

Below the transport service interface the protocol layers do not fit easily into the OSI model. For performance reasons the fundamental data object in the network is the 72-bit packet and each packet contains (in its Destination Address Field) all the information needed to route it through the network from its point of entry to its final destination. Thus the network layer protocol in Centrenet is not concerned with the routing of packets within the network (as it is in the OSI model), although it is concerned with directing messages from specific processes within source host processors to specific processes in destination host processors. Packet routing is strictly a function of the lowest (packet) level protocol in Centrenet as indeed are all operations which control the movement of packets within network hardware. This protocol provides primitive operations to send and receive packets and in many cases these operations are purely hardware functions (as in the case of the Starpoint or link interfaces used by Superports, for example). In the case of microprocessor based systems such as the NIM, on the other hand, operations on 72-bit objects involve a combination of hardware and software functions.

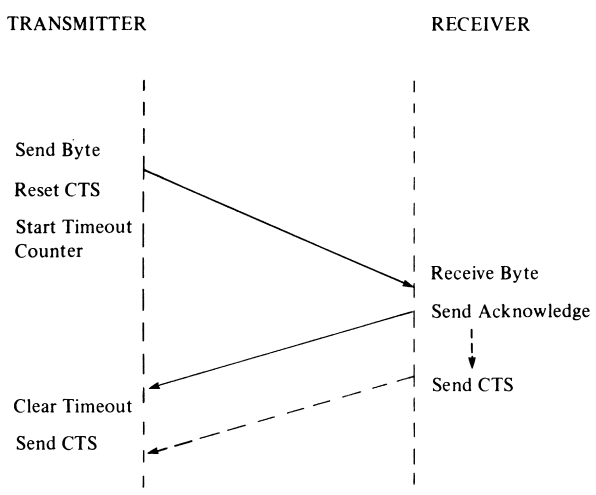

Fig 9. The Byte Protocol

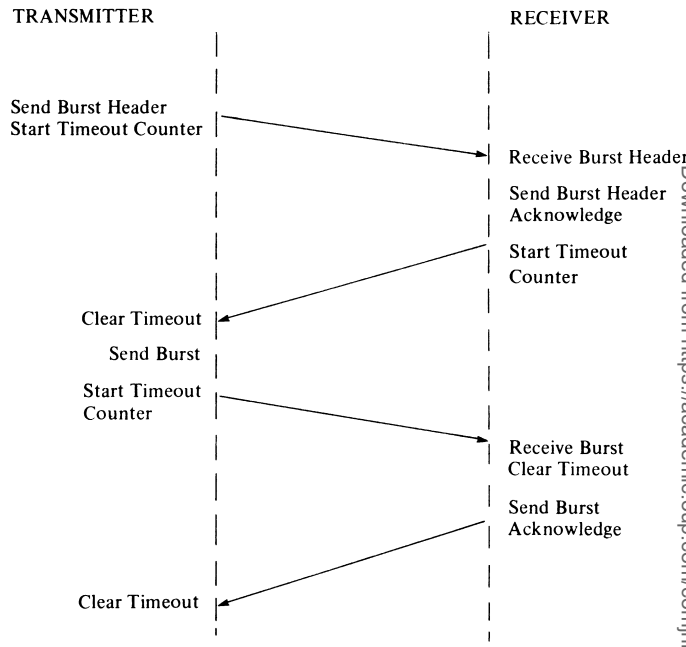

Fig 10. The Burst Protocol

Immediately above the packet level protocol in Centrenet two separate end-to-end link level protocols are defined. The 'Byte Protocol' (fig 9) allows individual 8-bit objects to be transferred across the network with an end-to-end acknowledgement mechanism, while the 'Burst Protocol' (fig 10) allows one or more bytes (up to 64K) to be transferred across the network apparently as a single entitiy and without individual end-to-end acknowledgements. It can be seen that six different types of object are handled by these protocols:

(i) Burst Header
(ii) Burst Header Acknowledge
(iii) Burst
(iv) Burst Acknowledge
(v) Byte
(vi) Byte Acknowledge

and bits within the packet control field (fig 11) distinguish between the different types of packet used in their implemention.

The Byte Protocol uses the network Byte Mode of operation to transfer packets across the network and also invokes the use of the CTS (Clear to Send) and ECHO bits in the packet control field to provide end-to-end flow control and network echoing. Although the arrival of an acknowledge packet at the source device indicates that the packet has been delivered to its destination, it does

ACK	packet is an Acknowledge Packet being returned from a remote terminal or computer.
ECHO	set by an originating device as a request for a network echo; causes content of an Acknowledge Packet to be returned to originating device.
CTS	Clear to Send; used for character handshaking in the Byte Protocol.
BURST	distinguishes between Byte Protocol and Burst Protocol Packets.
HEADER	identifies Header block packets in the Burst Protocol.
BLOCK END	identifies the last (or only) packet in a Block Mode transfer.
CRC	indicates that the packet contains a 16 bit block checksum.
NIM	packet is from a Network Intelligence Module and is to be interpreted as a network hardware control command.

Fig 11. Packet Control Field Bit Interpretation

not, of itself, permit a further packet to be sent. For this purpose a 'Clear-to-Send' indication must also be received. This may be asserted in the acknowledge packet, but may be sent separately if the destination is unable to receive a second packet straightaway. This has the disadvantage of increasing network traffic, but allows terminals and computer ports operating at different bit rates to be matched in terms of character rate by the network. What the arrival of an acknowledge packet does do is to halt and reset the timeout counter in the transmitter port, thus indicating to the transmitter that the character has been correctly delivered. In the event of a failure a timeout signal is returned to the transmitter.

Echoing is a normal feature of terminal operation (from keyboard to display) but in a network environment a number of different requirements can be identified. Some devices, for example, may operate permanently in half duplex mode, with local echoing, while some remote hosts may be responsible for character echoing, allowing for example, the suppression of password echoing. In other circumstances the terminal may expect remote echoing, but the host may not provide it, and in these circumstances echoing by the network of characters which arrive at the input port of a remote host gives the user some confidence in the correct operation of the network. By appropriate use of the ECHO bit in the packet control field, the Centrenet Byte Protocol allows all these alternatives to be implemented.

The Burst Protocol uses the network Block Mode of operation for each of the transfers involved and also relies on timeout counters as a part of its error recovery procedure. Figure 12 shows the structure of each of the blocks transferred. Each block is labelled with a 16-bit Logical Channel Identifier (LCI) as part of the mechanism used by the network level protocol to direct messages between source and destination processes. These identifiers are allocated from separate pools at source and destination so that the two processes involved refer to the logical channel between them in different ways. This technique localises the allocation of LCIs to the domain of the local processors, but requires that each end of the logical channel maintain a record of both LCIs. When a message is to be sent from one host to another the local identifier is converted, immediately prior to actual transmission, to the identifer of the destination. When a message is received, the LCI is inspected and the message passed on to the appropriate process. A zero LCI indicates that a connection is being set up and that an identifier has not yet been allocated. This use of LCIs is similar to the use of Port Numbers in the Cambridge Ring CR82 Protocol[11] specification. The use of LCIs in Centrenet may also be extended in the future to the Byte Protocol to allow a device with a single network address to maintain multiple logical connections concurrently.

The Burst Header and Burst Header Acknowledge blocks also include a 16-bit length field. In the Burst Header this indicates the length of block which the source wishes to send; in the Burst Header Acknowledge it indicates the length of block which the destination is willing to accept. If the latter is smaller than the former, then the transfer must be split across a number of bursts of the smaller size. The data block also includes a checksum in the last packet as a guard against packet corruption or loss in the network. Superports generate (and check) this checksum automatically, but it could be formed by software in some configurations. Individual packets are error checked by hardware in network links and are discarded if faulty.

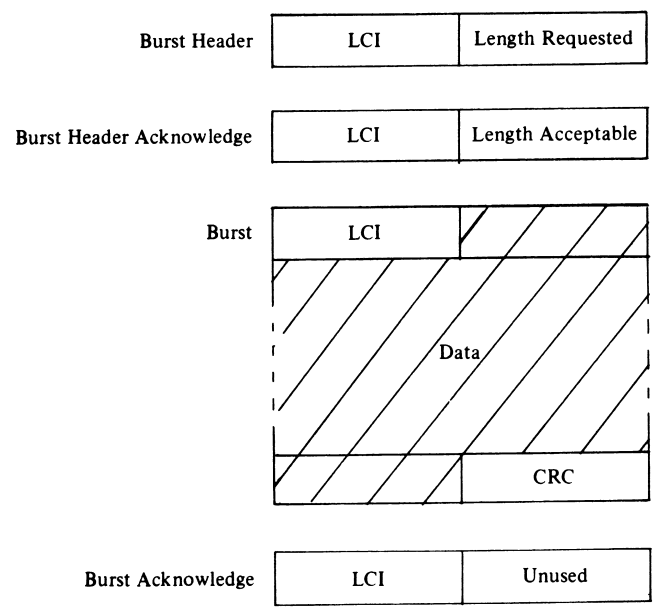

Fig 12. Burst Protocol Block Structure.

8. CONCLUSION

The Centrenet local area network system, in which high-speed parallel switching nodes are interconnected with computers, peripherals and other nodes via optical fibre links has a number of advantages over other local area networks. Principal among these is the fact that the total bandwidth available in the links and Starpoints is distributed about the network in such a way that only a subset of users are obliged to share any part of it. This is not the case in other forms of LAN such as common medium systems (eg Ethernet[12]) and ring systems (eg Cambridge Ring[13], the IBM token ring[14], and ICL's Macrolan[15]) where all users contend for all of the available bandwidth. These systems are also synchronous, in the sense that all parts of the network must operate at the same bit rate, and a performance upgrade would involve a complete system replacement. Centrenet, on the other hand is essentially asynchronous and is therefore incrementally upgradable; as the design and available technology improve those part of the network which become bottlenecks in a particular configuration can be replaced with higher bandwidth components. Common medium systems are also unsuited to implementation in optical fibre, even though an experimental system, Fibrenet[16] has been built, and in ring systems the use of a multiplicity of phase-locked loop circuits in a closed system appears to lead to stability problems. The point-to-point connection arrangement used in Centrenet avoids these problems and is ideally suited to implementation in optical fibre. Similar conclusions have been reached by Sikora and Franke[17], who describe a centralised bus architecture for local area networks in which point-to-point links interconnect devices and switching nodes in a manner similar to Centrenet. Here, however, the switching nodes operate serially and use a short contention bus to determine which of the attached modules may transmit its packet across the data bus. In Centrenet the use of a parallel switch provides sufficient bandwidth for a contention access scheme to be unnecessary.

Hardware for a pilot Centrenet system has been implemented and tested on a daily basis over a period of several weeks. This system consists of a Starpoint with NIM and Port Cards, two Local Links and two Superports interconnecting two PDP-11 computers. In the tests one of the PDP-11s continually generated and transferred to the other PDP-11 a sequence of 64K different data patterns in blocks of 512 packets. The second PDP-11 received each block and sent it back to the first PDP-11 which then checked the data in each received packet before proceeding to the next block. Each cycle of this activity took 85 ms, with each of the two block transfer involved occupying 36 ms. This corresponds to 70 μs per packet. Because the Local Link is a relatively simple device without packet buffering, it cannot accept a second packet until the first has been acknowledged. This involves the serialisation and de-serialisation of both the packet and its acknowledgement; at 15 Mbps the serialisation time of the 112-bit packet transmitted through the Local Link is 7.5 μs, giving a total end-to-end delay (including transmission delays) on the longer of the two links used in the pilot system (600 m) of 44 μs. The actions of the two links are overlapped by the packet buffering capability of the Starpoint, and the Superport overlaps some of its activities with the actions of the Local Link. There is, however, a 26 μs turnaround time within the Superport, both on transmission and receipt of packets, which cannot be overlapped in the present design, and this figure, when added to the 44 μs Local Link delay, gives rise to the observed 70 μs per packet transmission time.

During the tests errors occurred at a rate of the order of one per 8-hour period, giving an error rate of better than 1 in 2×10^{10}. Most of these errors were attributable to the effects of contact wear in the prototype hardware and a much better error rate should be obtained as this hardware is replaced. This will occur in the near future and expansion of the network to include a number of other hardware components, such as a microprocessor based terminal multiplexer and an IEEE 488 interface unit, is in progress. Software to allow use of the pilot system by the MUSS operating system is currently being implemented. Mechanisms to allow connection of Centrenet to other networks and to allow speech and image transmission across the network are being investigated.

Acknowledgements

The Centrenet pilot project was funded by the Department of Computer Science at the University of Manchester and the authors would like to thank Professor D. B. G. Edwards both for making funds available and for valuable suggestions concerning its implementation. Some support has also been received from GEC and current work is being funded by the SERC as part of its Distributed Computing Systems programme.

A number of staff and research students have contributed to the project and the authors would particularly like to acknowledge the contributions of Dr. C.J. Theaker, Dr. I.R. Wilson, Mr. S.C. Holden and Miss D.J. Bondi.

REFERENCES

1. T.P. Hopkins, The design of a local area computer network, Ph.D. Thesis, University of Manchester (1984).
2. D. Morris and R.N. Ibbett, The MU5 Computer System, The Macmillan Press (1979).
3. F.E. Heart, R.E. Kahn, S.M. Ornstein, W.R. Crowther and D.C. Walden, The Interface Message Processor for the ARPA computer network, *Proc AFIPS Spring Joint Computer Conference*, **36**, 551 – 567 (1970).
4. G.R. Frank and C.J. Theaker, The design of the MUSS operating system, *Software Practice and Experience*, **9**, 599 – 620 (1979).
5. T.P. Hopkins, An investigation of hardware requirements for the implementation of communications within a multi-computer system, M.Sc. Thesis, University of Manchester (1980).
6. N. Marovac, The rotating bus as a basis for interprocess communication in distributed systems, *The Computer Journal*, **25**, 22 – 31 (1982).
7. D.A. Train, An optical fibre communications system for a campus-wide local area network, Ph.D. Thesis, University of Manchester (1982).
8. T.C. Vaughn, Protocols and services for a high-speed

network, M.Sc. Dissertation, University of Manchester (1983).
9. ISO, ISO/TC97/SC16 Data processing – open systems interconnection – basic reference model, *Computer Networks*, **5**, 81 – 118 (1981).
10. British Telecom PSS User Forum (Group 3), A network independent transport service, SG3/CP(80)2 (1980).
11. J. Larmouth (ed.), Cambridge Ring 82 Protocol Specifications, Joint Network Team (1982).
12. J.F. Shock, D.D. Redell, Y.K. Dalal and R.C. Crane, Evolution of the Ethernet local computer network, *IEEE Computer*, **15**, 10 – 26 (1982).
13. M.V. Wilkes and D.J. Wheeler, The Cambridge commmunication ring, *Proc. Local Area Networks Symposium*, Boston Mass., 47 – 60 (1979).
14. R.C. Dixon, N.C. Strole and J.D. Markov, A token-ring network for local data communication, *IBM Systems Journal*, **22**, 47 – 61 (1983).
15. R.W. Stevens, Macrolan: a high-performance network, *ICL Technical Journal*, **3**, 289 – 296 (1983).
16. E.G. Rawson and R.M. Metcalfe, Fibrenet: multinode optical fibres for local computer networks, *IEE Transactions on Communication*, **26**, 983 – 990 (1978).
17. J.J. Sikora and D.C. Franke, A LAN based on a centralized-bus architecture, *Proceedings of Localnet* '83, New York, 147 – 157 (1983).