About This Remade Document

Title: Econet Data Link and Physical Layer Specification (v3.0)
              (also known as the "Rick Rand document")
Last remade: 30-Mar-2025
Repository: https://github.com/acheton1984/AcornDocsRemade

This is not quite the beautifully "remastered" document as seen elsewhere, but it is intended to be very similar to the original in an easy-to-read and searchable PDF format.

Reconstruction Notes

Source scan (no longer live): http://www.andrewgordon.org.uk/rick_rand_doc.pdf

* Mention of this was first spotted on a Stardot thread - https://www.stardot.org.uk/forums/viewtopic.php?p=309196#p309196
* Recreated from a scan of the original document.
* Font sizes, margins, line spacing, and character spacing are only approximate. Some words may appear on different lines, but the overall layout and pagination are preserved.
* Page numbers have been added.
* The index has been recreated as closely as LibreOffice, the original's header numbering, and I could manage. It now includes page numbers.
* Fig 5.1 and Fig 7.0.1 are now included in-line, instead of being tucked in near the end.
* Spelling errors have been corrected where spotted.
* What I believe to be the station circuit diagram at the end of the original has been titled as such.
* Some items are missing: the original ToC mentions Terminator/Clock circuit diagrams and References which should be at the end. Also section 9.0 mentions "Fig 9.1".
* A title and remade document history page have been added, along with a link to this repository.

Disclaimer

---- original document follows ----

THE ECONET

A local area network

Data Link Layer
and
Physical Layer
Specifications

Version 3.0
August 1982

Acorn Computers
Cambridge, England.

All correspondence should be addressed to:-

Customer Service Department
Acorn Computers Limited
Fulbourn Road
Cherry Hinton
Cambridge.
CB1 4JN

Abstract.

This document has been compiled as a design reference document. It describes the subnet communication layers of Econet which includes electrical characteristics and communication protocols. The intention is to provide adequate information concerning the behaviour of the data link layer while leaving individual implementations to exploit whatever technologies are appropriate. In addition, the specific implementation used by Acorn Computers is described in detail as an example of one such technology. To this end a large proportion of the report is dedicated to describing the implementation of the MC6854 Advanced Data Link Controller, which is at the heart of the data link layer used by Acorn Computers.

Index.

NB: "Appendix 1.3 Terminator/Clock circuit diagram" and "(13.0) References" are mentioned in, but missing from, the source material.

(1.0) Introduction.

Econet has been developed to provide an extremely low cost method of connecting a number of microcomputers on a local data link. It is thought that this system will be most suited for use in schools, laboratories and small commercial operations for network lengths up to 500 metres and data rates up to 250 Kbaud. Communications are via a single bit broadcast channel with all stations on the channel having equal priority. With bus orientated networks such as this, the system is both reliable and resilient to station failures. This means that stations on the network can be added or removed without having any effect on any other station. The communication channel consists of two pairs of differentially driven twisted pair cables. All stations on the network are synchronized by a clock which is conveyed over one of the differentially driven wires and data is transmitted over the other. Each station will normally consist of a host computer and an interface to the communication channel which is both cheap and simple.

Econet is a layered system which can be split up into various levels which are separate from each other. In this way, as long as the rules concerning each layer are obeyed, users may implement their own ideas and technologies within the structure of the network.

Errors are detected in hardware at both the bit and packet levels. This gives the fastest possible detection of inter-packet collisions and detection of data corruption during the transmission of a packet.

The data link layer of Econet is characterised by the low level primitives which were developed by Acorn Computers. The protocols required to implement the primitives are described throughout this text and for those who wish to communicate with existing stations on the Econet, the section on transmit/receive will be essential. This section describes the four way handshake protocol which is the main method of passing messages between stations.

The flexibility of Econet is increased greatly by two expansion units which allow variable topologies and increase the number of stations which can be contacted from any one station. The first is an extender unit which enables the length of a network to be increased and the topology to be varied from the simple straight line to an H, Y or X configuration. The second extension facility which is already built into the existing data link layer, is the facility for communicating with other networks (such as the Cambridge Ring or Ethernet), via a device called a gateway.

(2.0) Econet layering.

Econet has been designed as a three layered network, each layer being system independent of the others. The three layers are the Physical, Data link and Client layers. The Physical and Data link layers are described fully in this document and the Client layer is described in the Econet User manual.

(2.1) The Physical layer.

The Physical layer describes the electrical characteristics and the bit stream requirements of the communication channel which includes byte formatting (ie zero bit insertions), timing requirements and voltage waveforms etc.

Two types of error are detected at this layer, the first is the detection of a collision which is checked bit by bit while a packet is being transmitted. The second is the cyclic redundancy check character which is examined at the end of a packet and indicates whether a packet is correct or not.

(2.2) The Data link layer.

The Data link layer forms the interface between the client and physical layers. The communication protocols are defined in this layer which allow messages to be passed to and from the client and physical layers without error. This means that errors which may occur at the physical layer during the transmission of a message, are detected and reported to the client layer in a predictable manner.

(2.3) The Client layer.

This layer may be split up into several other layers in various implementations but for the purposes of this document it is considered as a single layer which initiates transmissions and handles errors conditions due to bad packet transfers etc.

(2.4) Comparison to the ISO model.

The International Standards Organisation have described a seven layered model for network communications in an attempt to set an international standard. Econet is compared with this model in the following diagram.

Fig 2.4.1 Approximate correspondence between Econet and the ISO model.

<table>
<tr><th>Layer</th><th>ISO model</th><th>Econet model</th></tr>
<tr><td>7</td><td>Application</td><td rowspan="5">Client</td></tr>
<tr><td>6</td><td>Presentation</td></tr>
<tr><td>5</td><td>Session</td></tr>
<tr><td>4</td><td>Transport</td></tr>
<tr><td>3</td><td>Network</td></tr>
<tr><td>2</td><td>Data link</td><td>Data link</td></tr>
<tr><td>1</td><td>Physical</td><td>Physical</td></tr>
</table>

See Ref 1. for further details of the ISO model.

(3.0) Functions of the ADLC.

The MC6854 Advanced Data Link Controller is an intelligent peripheral device used to transmit and receive data packets over a synchronous communications channel between two or more computers using a Bit Orientated Protocol. The ADLC performs the conversion of parallel to serial data and the construction of the packets which are sent over the network. Its functions include the automatic generation of opening and closing flags, Cyclic Redundancy Check calculations/checking and zero insertion/deletions. The opening and closing flags are used to indicate the boundaries of a data packet and are eight bit words consisting of 01111110. These flags will always be detected as flags and not as data because the ADLC inserts a zero after a series of five ones in all the data and CRC bytes. On receipt of an opening flag the ADLC will delete a zero if it follows five ones, in this way the packets can be framed without confusion with data and the host need not be concerned with framing problems. The CRC is positioned immediately before the closing flag and consists of a two byte word generated by a sixteenth order polynomial. The complete packet format can be seen in the next section.

(3.1) Programming the ADLC.

The ADLC is a complex device with four control registers, two status registers and quad transmit and receive first in first out data buffers. The functions which are required for the implementation used by Acorn Computers are described in detail below. (See Ref 2. for further details of the ADLC).

(3.1.1) Control register 1.

Bit 0. Switch which controls access to CR3 or CR4.
Bit 1. When set enables RX interrupts.
Bit 2. When set enables TX interrupts.
Bit 3. Not used.
Bit 4. Not used.
Bit 5. Not used.
Bit 6. Resets all RX status and FIFO registers.
Bit 7. Resets all TX status and FIFO registers.

(3.1.2) Control register 2.

Bit 0. Status bits prioritized.
Bit 1. This bit should be set.
When set this causes the ADLC to work in two byte mode. This means that Receive Data Available and Transmit Data Register Available are asserted when there are two data bytes ready and two transmit registers available.
Bit 2. Determines whether flag time fill or mark idle is used between packet transfers.
Bit 3. Selects frame complete or TX data register ready.
This bit should be cleared, selecting TDRA.
Bit 4. Transmit last byte.
This is initially set to zero and is used to indicate that CRC and closing flag should be transmitted.
Bit 5. When set RX status bits are cleared.
Bit 6. When set TX status bits are cleared.
Bit 7. Request to send.
This bit controls the RTS output which is used to enable the output transmit driver.

(3.1.3) Control register 3.

Bit 0. This bit is cleared.
Indicates that no logical control byte is included in a packet.

Bit 1.	This bit is cleared. Indicates that the control field is 8 bits only.
Bit 2.	No auto address extend. This means that an 8 bit address is assumed by the ADLC. Although Econet uses 15 bit addresses this facility is not used as the MS bit of the second byte of the address field is used to indicate special operations. (See section on packet format for more details).
Bit 3.	Idle is set to be all ones.
Bit 4.	Not used.
Bit 5.	Not used.
Bit 6.	Not used.
Bit 7.	Not used.

(3.1.4) Control register 4.

Bit 0.	This determines whether the closing flag of one packet is used as the opening flag of the next. This is not important in the Acorn implementation as transmission is always followed by a reception.
Bits 1&2.	Set TX word length to 8 bits.
Bits 3&4.	Set RX word length to 8 bits.
Bit 5.	Not used.
Bit 6.	Not used.
Bit 7.	Not used.

(3.1.5) Status register 1.

Bit 0.	Receive data available. This bit is used to confirm that two bytes are available in the receive FIFO.
Bit 1.	Not used.
Bit 2.	Not used.
Bit 3.	Not used.

Bit 4. Clear to send.
This bit reflects the condition of the CTS input. When enabled, a low to high transition of CTS causes an interrupt.

Bit 5. Not used.

Bit 6. TX data register available.
This should be read before a byte is sent. When CTS is high TDRA is deasserted which indicates that transmission should end.

Bit 7. IRQ.
Interrupts are enabled at various times which indicate transmit, receive and error conditions.

(3.1.6) Status register 2.

Bit 0. Address present.
When set, this indicates that the first byte of a packet is available to be read.

Bit 1. Frame valid.
This indicates that a packet has finished and the CRC was found to be correct.

Bit 2. Receive idle.
When set, this causes an IRQ after 15 ones have been received.

Bit 3. Not used.

Bit 4. Not used.

Bit 5. Data carrier detect.
This reflects the DCD input, which indicates the state of the clock signal. If enabled, an interrupt will be generated when the clock fails.

Bit 6. Not used.

Bit 7. Receive data available.
This is used to confirm that after a receive interrupt that data is actually ready. Other causes of interrupt indicate an error condition.

(3.2) Cyclic Redundancy Check.

The CRC is a 16 bit error checking sequence which is the result of a mathematical computation performed on the binary values of all the bits (excluding inserted zeros and flags), in a packet. It is used to validate the contents of a packet and if found to be incorrect it causes the frame valid flag to be deasserted. The CRC field is the ones complement of the remainder of a calculation performed using the polynomial $X^{16}+X^{12}+X^{5}+1$ which results in a constant of HEX F0B8.

(4.0) Packet Format.

There are three types of packet which are transmitted over the Econet, all of which are involved in the four way handshake as described in the transmit/receive section. The number before each field represents the byte position in the packet and each field has the following meanings.

(4.0.1) Opening/Closing flag:

A single byte with a 01111110 bit pattern.

(4.0.2) Destination Address:

This is the local network station ID of where the packet is going to, (1 byte).

(4.0.3) Destination Network/Special operation:

If the most significant bit of this byte is clear then the rest of the byte represents the network address of where the packet is going to. If the top bit is set then this will indicate a special operation which has not yet been defined. The intention is that individual implementations will be able to use these codes totally independent of the Acorn Computer stations.

(4.0.4) Source Address:

This is the local network station ID indicating where the packet came from.

(4.0.5) Source network/Special operation:

This is similar to the Destination network/Special operations field but indicates where the packet came from or which special operation etc.

(4.0.6) Control byte:

This byte can be used to indicate things about a packet such as associated operation or number.

(4.0.7) Port byte:

The port byte is used to associate a meaning to a packet.

(4.0.8) CRC field:

This is a two byte frame check sequence.

(See section 3.2).

(4.0.9) Data field:

This is a variable length field in the range 1 to N, where 1 is the minimum and N is the maximum number of bytes in the field.

(4.1) Calculating the upper bound on the data field.

The upper bound on the data field is determined by the number of bytes that can be transmitted before a timeout condition is generated at another station. A timeout occurs after a station has found the line busy for 10 seconds which means that no transmission should last longer than that time. When determining the upper bound, it is important to remember that the whole of the four way handshake must be included and that zero insertions will also increase the number of bits sent. With this in mind N can be estimated to be equal to the clock frequency of the network. ie. A clock frequency of 100 KHz would allow an upper bound of 100 Kbytes of data per four way handshake.

(4.2) Scout packet.

Format of the scout packet used in the four way handshake is as follows.

(1) Opening flag:
(2) Destination Address field:
(3) Destination Network/Special operation field:
(4) Source Address:
(5) Source Network/Special operation:
(6) Control byte:
(7) Port byte:
(8-9) CRC 16 bits:
(10) Closing flag.

(4.3) Data packet.

The second type is the data packet which contains all the data to be transmitted. Its structure is as follows.

(1) Opening flag
(2) Destination Address:
(3) Destination Network/Special operation:
(4) Source Address:
(5) Source Network/Special operation:
(6-n+5) Data field: (Must be in the range 0<n<N bytes)
(n+7-n+8) CRC 16 bits:
(n+9) Closing flag.

(4.4) Acknowledge packet.

The third type is the acknowledge packet which is simply a copy of the to and from addresses. ie.

(1) Opening flag:
(2) Destination Address:
(3) Destination Network/Special operation:
(4) Source Address:
(5) Source Network/Special operation:
(6-7) CRC 16 bits:
(8) Closing flag.

The data and scout acknowledge packets are identical.

(5.0) Transmit and Receive.

When attempting to transmit a packet, the source station should check the data line to see if it is currently undriven. An undriven line is indicated by a string of 15 or more consecutive ones. If the ADLC is used then this condition can be detected by reading the "Idle" flag and if it is set then transmission can begin. It is important that no transmission is made before the 15 ones have been detected as this may cause an error during a four way handshake already taking place. The reason for this restriction is that the low level software assumes a minimum period between packets of 15 clock cycles and therefore makes no attempt to jam the line during the four way handshake if it believes it has enough time to reclaim the bus between packets.

When a transmit procedure is called, the two stations involved in the transaction are required by the Econet software to go through a four way handshake procedure which is described below.

(i) The handshake is initiated by the client layer and begins by sending a scout packet which alerts the destination station of an imminent data transfer.

The destination station should have already been in a receive state and would then compare the scout against an internal data structure to try to find a match. While the destination station is searching its data structures, it jams the data line by sending a continual stream of flags to prevent any other stations from claiming the line. This condition is called Flag Fill and is required at this point because the search may take more than 15 cycles to complete.

(ii) If a match is found then the receiving station deasserts Flag Fill then transmits a positive acknowledge packet which contains the address of the source and destination stations.

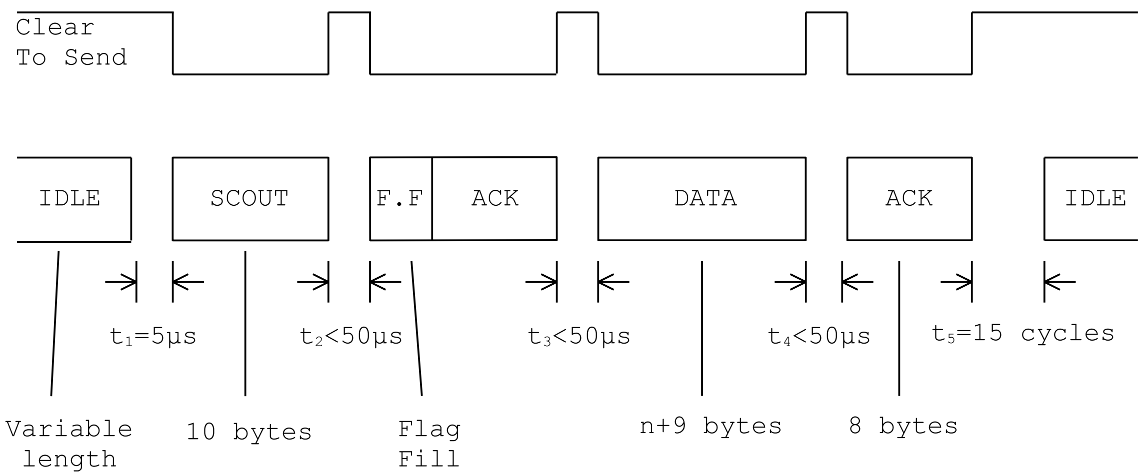
(iii) On receipt of the scout acknowledge the source station proceeds to send the required data buffer.

(iv) Finally the receiving station transmits the message acknowledge signal if the transfer was successful.

A schematic diagram of the four way handshake can be seen in Fig. 5.1.

NOTE: If the final acknowledge fails to reach the source station, it will assume that the data packet did not reach the destination station and will therefore retransmit the whole message. This can lead to stations accepting duplicate data if appropriate action is not taken. There are a number of ways this problem can be overcome and it is the responsibility of the client layer to arbitrate this condition. One solution is to tag the message with an identification number which can be put into the control byte of the scout packet. When the destination station receives two messages in succession it knows it can ignore the second if it is identical to the first with the exception of the tag number. It must still acknowledge the scout and the data packets sent by mistake to prevent further retries.

Fig 5.1 Schematic diagram of four way handshake.



(6.0) Immediate operations.

The immediate operations are different to transmit and receive in that the source station does not need the cooperation of the destination station.

(6.1) Peek.

This operation takes a block of memory from a remote station. The source station sends a scout with the start and end address of the block to be transferred between the port and CRC fields. The destination station replies with a data packet.

(6.2) Poke.

This operation puts a block of memory into a remote station. The four way handshake protocol is used in this operation.

(6.3) Remote subroutine call.

A remote station can be made to jump to a subroutine by the local station. Arguments can be passed to the remote station in the data block of the four way handshake. The absolute address of the jump is sent immediately after the port byte in the scout packet. Once the handshake has completed, an event is generated in the remote machine which causes the jump.

Note: Care should be taken when using this operation at the remote station as it can be made to jump to any position in memory.

(6.4) Event procedure.

This is similar to the remote subroutine call, except the event is handled by the remote station. In this way the remote station can ensure that the procedure call is a sensible one.

(6.5) High level procedure.

This is the same as the event procedure except that the event is dealt with by the Econet software.

(6.6) Stop.

This operation forces the remote station to stop processing. This is achieved using a simple two way handshake which consists of a scout and an acknowledge package.

(6.7) Start.

This causes remote station to restart. two way handshake is used.

(6.8) Machine type peek.

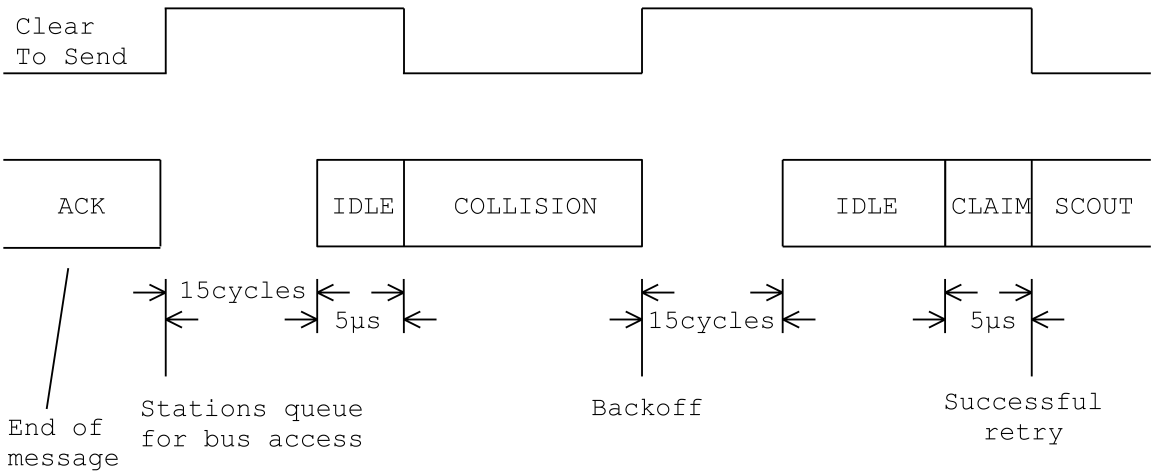
This is a two way handshake consisting of a scout and a data packet. The call enables a local station to find out the type of a remote station. (ie File server, Printer server etc).

(7.0) Collision detection.

One characteristic of broadcast networks like the Econet, is that collisions occur when two or more stations choose to transmit on the network at the same time. This is largely avoided by stations testing the line before trying to transmit in an attempt to find out if the line is currently undriven. A contention can exist if a second station attempts to transmit between the time the first station checks and claims the line. This period of contention lasts for only 5 microseconds and therefore is a very rare event. If a collision does occur in this period, the minimum time that the collision can be guaranteed to be flagged, is at the 40th bit of the scout packet which is the least significant bit of the source station address. The bus is claimed by setting the Request_To_Send bit of CR2 which enables the transmit line driver.

When the network data bus is undriven, the line is left in a biased state by the terminating components (see section 9.0 on termination), indicating a logic level one to the receive comparator and a collision condition to the collision detection circuitry. A collision is detected when the differential voltage between the lines is less than 1 Volt. This causes the collision detection circuitry to deassert Data_Valid and Clear_To_Send of the ADLC which generates an interrupt request which indicates that transmission of data should end. Once the transmitting stations detect that the bus is no longer clear to send, the transmissions then back off. At this point the collision arbitration software is entered which should ensure that the line is cleared and that subsequent attempts by each station to retransmit are successful. A diagrammatic view of the contention period in Fig. 7.0.1.

Fig 7.0.1 Contention period.



(7.1) Collision Arbitration.

When a collision has occurred, the error is reported by the data link layer so that the attempted transmission may be rescheduled such that it will not cause any further collisions. This is achieved by delaying an integer multiple of the window time of 5 microseconds (which is the time required to reset and claim the data bus), before the error is flagged to the client layer. The multiple factor is determined by the station identification code which is different for each station on the network. In this way all stations involved in a collision attempt to retransmit at different times so avoiding further collisions between them.

The backoff software is arranged in such a manner as to give priority to stations with the highest identification numbers.

(8.0) Clock Generation.

Econet is a synchronous system so the network requires a single system clock which is generated by an MC14411 clock generator driven by a 1.8432MHz crystal. There are 14 different frequencies available from this configuration ranging from 614KHz to 4.8KHz. The clock frequency for a given network will be dependent on the network topology. This is documented more fully by the installation manual but the table given below gives an indication of the length speed tradeoffs. There are two limiting factors which determine the maximum speed of the system clock. The first factor is that the Econet software is at present designed to run up to 250 Kbaud and the second is that no two stations should be more than half a clock cycle out of phase.

The figures quoted in Table 8.1, are the maximum clock frequencies for a given clock to furthest station length, assuming the propagation velocity of twisted pair cables to be 0.57 c which is 171 Mm/s and that MC68B54 ADLCs are used.

Accurate figures can be obtained from the following formula:

$$\text{Max clock frequency} = V/(4.L + 2.D.V)$$

Where:

L = length between the clock and the furthest station.

V = Propagation Velocity of the signal

D = Station propagation delay = 420 ns for Acorn devices.

Table 8.1 Clock frequency versus length tradeoffs.

Length metre	Max freq KHz
100	250
150	230
200	180
250	150
300	125
350	110
400	95
450	85
500	80

(9.0) Network termination.

It is recommended that all Econets be terminated in the manner described in this and the installation manual, for the following reasons.

Correct termination is required to minimise the signal reflections which occur at the ends of the lines. The amount of potential reflections is dependent on the network topology and speed and appears as overshoot and ringing on the signal (see Fig. 9.1 for appropriate waveforms)[1]. In general long fast networks with several branches are more prone to termination problems than others. Termination also provides the appropriate +0.6 volt bias required to hold the lines at logic one when undriven (this is the only reason that a terminator requires a power supply). It is recommended that twisted pair cables be used as they reduce the likelihood of external interference and crosstalk. Such cables have a characteristic impedance in the order of 120 ohms, which means that the terminating impedance should also be 120 ohms. A detailed description of twisted pair cables is given in the section on cables.

SEE APPENDIX 1. FOR CIRCUIT DIAGRAM.

(10.0) Line drivers.

Both the data and clock lines are driven by the SN75159N which is a tristate differential line driver. The differential drive voltages are typically 0.25 and 3 volts, driving into 50ohms. (see Ref. 3 for full specification).

The diagram (Fig. 9.1), shows the typical settling times of a 400m network running at 100KHz.

1 Fig 9.1 is missing from the source material.

(11.0) Cables.

The Econet cable uses two twisted pair wires with overall screen. The twisted pairs are important for a number of reasons.

(i) External Noise sources generate voltages which can be induced into cables in a similar way to a radio aerial picking up a signal. It is difficult to prevent this from happening especially with very long cables but its effects can be minimised by using differential detectors. This is because each of the wires in a pair are effected by an equal amount, hence the difference between them is not altered.

(ii) A balanced differential drive gives minimal radiation as average voltage on both lines is constant.

(11.1) End-to-end resistance.

It is important that the cable has a low end to end resistance. This is because a signal will become attenuated as it travels down a line but it must never become so attenuated that the line is taken to be undriven. The maximum end to end resistance is therefore specified as 15 ohms leaving 5 ohms for connection resistance. The total maximum end to end resistance is therefore 20 ohms.

For networks with more than 10 stations, it is recommended that station taps (ie the physical connection to the network channel), are made with solder connections rather than in line "T" plug connectors. This will increase connection reliability and will keep the network resistance low.

(11.2) Network earthing.

The cable must have the ground wire connected to earth at one point on the network only. Current loops may exist via earthed stations but the effect is minimised by the differential arrangement.

(11.3) Cable specification.

Max end to end resistance for 500 m	15 Ω.
Minimum propagation velocity	0.5 c.
Characteristic impedance	120 Ω ± 20 Ω.
Cable type	Twisted pair with overall screen.

Suitable cable types.

BICC CS 7227 0.9mm 2 pairs with screen.
(Recommended for networks up to 500m).

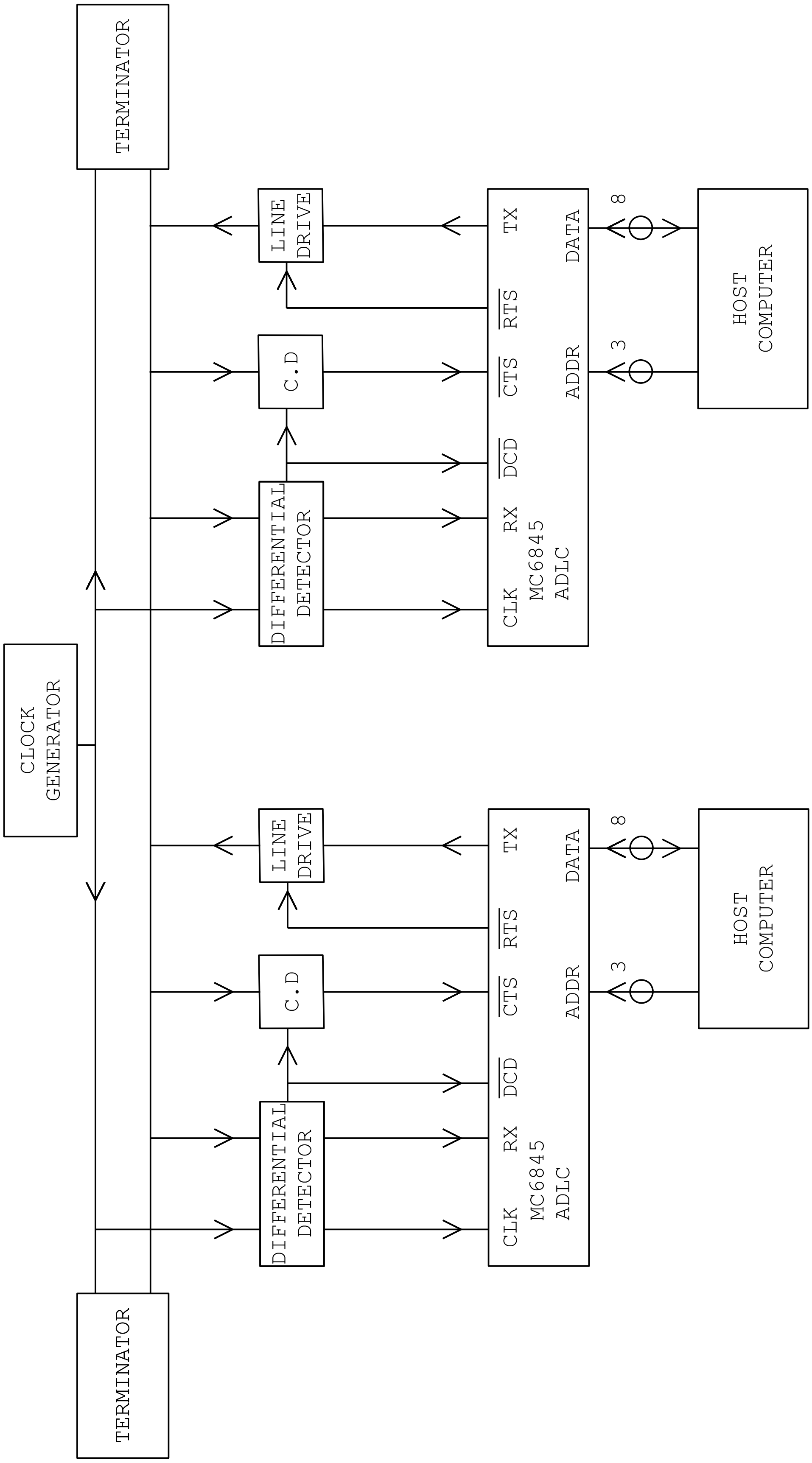
RS 367-921 0.643mm 2 pairs with screen.
(Recommended for networks up to 300m). Available in 100m reels.

(11.4) Wiring conventions.

The following cable wiring is recommended for connecting the 5 pin 180 Din sockets.

Pin No	RS	BICC
1	Black	White
4	Red	Orange
2	Shield	Shield
5	Black	Blue
3	White	White

Appendix 1.1 Econet System Diagram



C.D = Collision Detection Circuitry.

Appendix 1.2 Station Circuit Diagram