D

C

United Kingdom Atomic Energy Authority

HARWELL

Computer port interface for the Camac serial highway

R. Bayliff and J.W. Hall
Materials Physics Division
AERE Harwell, Oxfordshire
May 1979

C8

COMPUTER PORT INTERFACE FOR THE CAMAC SERIAL HIGHWAY

R. Bayliff and J. W. Hall

ABSTRACT

A simple interface unit has been built to connect a CAMAC serial crate controller to an asynchronous communication port on a computer or to a teletype. The unit has been used to connect neutron spectrometers over long cable lengths to a control computer and work at a preselected rate between 110 and 9600 baud; this is ideal for CAMAC applications associated with mechanical movements.

A program is described that illustrates the use of the interface with a SCC-1 module. Full listings of the PDP-8 version of this program are given.

CONTENTS

INTRODUCTION

1. The CAMAC serial highway (Ref. 1,2) is particularly useful when it is required to drive systems that are widely dispersed or where the crates are at some distance from the computer. Transfer speeds up to a few megabytes per second may be achieved using the "byte mode" of the serial highway and nearly 10^6 bytes per second may be sent in the single bit mode. To provide reliable operation at these rates a simple error detection method is implemented in which the parity of each byte is checked and at the end of the sequence of bytes comprising a message, the binary a longitudinal parity check is made.

2. In the more complex CAMAC systems a fast response to stimulii originating in the CAMAC hardware itself is often required. To achieve this it is necessary to keep the highway "idling" at full speed by continually transmitting dummy data along it. The potential high loading on the driving computer is avoided by using units designed to go between the computer port and the highway. These units produce idling data, only interacting with the computer when genuine information is to be passed. Additional hardware in the interface unit also performs the lateral and longitudinal parity checks. The resulting interface has a general application but is fairly complex and costly. A simpler interface adaptor is described in this paper that is suitable for CAMAC systems using lower data transfer rates. A program to drive the Serial Crate Controller SCC-L1 from a PDP-8 computer via the standard "KL8J Asynchronous Serial Line Interface" (Ref. 3) and this unit is also included.

THE ORIGINAL APPLICATION

3. Triple axis neutron spectrometers play an important part in materials physics research and an important machine of this type has been in use on the Harwell PLUTO reactor since 1966. This spectrometer was updated in 1973 (Ref. 4) and amongst the added features was a CAMAC system with a Harwell 7048 Crate Controller situated locally to the PDP-8 computer. This spectrometer was very reliable and easy to use and it was natural when two similar spectrometers, the MARX and DTA, were computerised in 1976 that the main features of the PLUTO spectrometer were incorporated in the new designs. The major parts of the computer programs for all three systems are common but for the later ones the computer and spectrometer mechanism are separated by 200 and 50 metres respectively.

4. Separation of computer and spectrometer was not new, other neutron spectrometers had been in use for some years using the Harwell 7035 and 7036 buffer units to drive 100 pair interconnecting cables. Obsolescence made the use of the 7048, 7035, 7036 combination unattractive especially since a CAMAC Serial Highway Controller was due to be available commercially by late 1975 (Ref. 5). The CAMAC serial highway had been defined after the design of the original spectrometer in 1973 and was well suited to the type of data transfer used in these machines.

5. The data transfer rates for the triple axis spectrometer are quite low: there are six shaft registers and four scalers that require to be set up or read for each measurement, a measurement rarely lasting less than 30 seconds. Read or write instructions over the serial highway require 12 bytes and a communication speed of 200 bytes/second extends measuring time by only about 3% in the worst case and generally by very much less than 1%.

6. Unfortunately, although the crate controller was available for the new spectrometers a suitable adaptor to connect it to a PDP-8 output port could not be delivered in time. An interface was necessary to convert signal levels and possibly relieve the computer of the burden of continuously driving and monitoring the highway. It was decided to construct a simple unit that will now be described.

CONSIDERATIONS FOR A SERIAL HIGHWAY ADAPTOR

7. A Serial Crate Controller to computer interface could have been constructed using a published design such as SHIFT (Ref. 6), but this was more complex than the application warranted. The design of a simple unit using LSI components such as a bit rate generator and UART was as easy as modifying circuits for use with SHIFT; additionally it would be easier to test, commission and maintain.

8.   The simplest standard PDP-8 "port" that will drive over long cables is the KL8 Asynchronous Line Interface. This was taken as a starting point in the design of the simple interface unit.

9.   The later version of the KL8, the KL8J provides a selected communication rate between 110 and 9600 baud while earlier versions go up to 1200 baud (KL8E) or 2400 baud (KL8F). The higher rates are intended for modem drive and EIA RS232 circuits are included but these only handle 25 metres of cable. The 20ma current loop however drives over much longer distances and can be made effective at all baud rates by removing the contact bounce suppression capacitor on the receiver input connections. Since neither the 20ma nor the RS232 circuits can drive the 'D port' of the controller directly, some additional circuits were needed. The CAMAC mechanical format rather than DEC cards was chosen for the interfacing circuits since this can be used with any future system even if the computer is changed. The interface can be plugged into the crate near the controller and the 20ma current loop used to drive over long lines to the computer. Power (+6V) for the unit is obtained from the crate supply via the back connector.

10.   A data transfer rate of 2400 baud was anticipated as the usual speed for the line, this being compatible with both the KL8J and the earlier KL8F, but to allow flexibility it was decided to provide a number of the more "standard" baud rates; the provision of 110 baud being particularly useful in permitting testing using a teletype ASR-33 rather than tying up a computer. The lower baud rates are also useful in permitting the interface to plug directly into a computer console port for testing or occasional use rather than requiring a dedicated channel. The use of a bit rate generator to give the selected baud rates kept the design simple.

11.   The use of interrupts using LAM signals within the CAMAC necessitates continual monitoring of the CAMAC highway. In the spectrometer system however requests for attention come from front panel signals such as the "RIN" from shaft positioning units (Ref. 7) and the "A" outputs on 7039 scalers. These signals can be combined to give a call for attention to the computer. This call can take the form of a signal on the line taking the CAMAC signals to the computer which with this approach would only be in use when the computer initiated a transfer in response to a call for attention or an internal (program) request. The main advantage of this independent call for attention is that the hardware or software is simplified since there is no need to keep the highway live by feeding dummy data along it. The call for attention is faster than the more conventional CAMAC demand in achieving interrupt response and is no more restrictive in the need to hold interrupts when the line is busy than any other CAMAC highway arrangement.

SCC INTERFACE - CIRCUIT DESCRIPTION

12.   Figure 1 is a photograph of the interface unit and a full circuit diagram is given as figure 2. Table 1 gives a component cross reference from figure 2 to the elements in the simplified diagram, figure 3, which will be used in this description of the operation. Appendix A gives manufacturer's data on the "UART", "Bit Rate Generator" and 12 volt power supply circuits.

13.   The control signals to and from the unit are bit serial on 20ma loops using a standard convention with 1 start, 8 data and 2 stop intervals. The start is signalled by a zero current (open circuit line) interval, the stop intervals correspond to full (20ma) current. The least significant data bit follows the start, no current flowing gives '0' data and 20ma corresponds to '1'. The serial input loop is electrically isolated from the unit by the opto-coupler circuit 'A' which include a bounce suppression circuit in case a teletype is used to provide control. The serial output 'B' similarly uses an opto-coupler to provide isolation although in this case a Darlington transistor circuit is used to boost the opto-coupler current to 20ma and to provide boosted open circuit voltage rating. The incoming serial data feed directly into the UART, 'E', which is clocked at 16 times the baud rate selected on an 8 position DIL switch from the Bit Rate Generator, block H.

14.   The incoming serial data is also used to operate the priority circuit comprising the elements K, L, M, N, P, Q, R. The priority circuit allows the data that has been latched into the attention register J to be loaded into the UART output buffer provided that:

   (a)   No data has come from the control/teletype for at least 16 baud intervals.

- 4 -

(b)   No data has been sent to the control computer/teletype for 5 baud intervals.

This interval is defined by the divide by 256 counter "K" which counts the clock output from "H" after being reset by a '0' pulse from the input circuit "A" or from the back edge of the TBRL pulse via "L" (as the data is accepted by the UART). This data will take 11 baud intervals to be sent leaving a wait of 5 intervals.

15.   If no data are transmitted or received for 16 baud intervals the output from "K" will set the bistable "N". This will prime the gate "Q". If or when one or more of the latches in the register "J" are set, the output from "P" causes an output from "Q" which causes "D" to select the data from the 8 latches in the attention register. The output from "Q" also triggers the monostable "G" which causes the data to be loaded into the transmitter buffer register of "E". The data is loaded (in this case "E's" transmitter register is already empty - see condition (b) above) by the training edge of the TBRL pulse. This edge also causes "L" to trigger giving an output from "R" which was already primed by the output from "Q". The "R" output will reset all the "J" latches and the bistable "N". Once "J" or "N" reset the "R" output ceases but with the circuit delays there is sufficient time to ensure proper resetting of "J" and "N". It can be seen that attention data can only be sent at 16 baud intervals rather than the 11 baud interval rate that CAMAC data is sent.

16.   If the computer sends data to the SCC interface this ensures that "N" is cleared and thus gate "Q" is shut ensuring that its output will cause "D" to select data from the CAMAC bus input via "C". Once the UART receiver register is full the data is presented to the CAMAC bus via the balanced line transmitters "F", the "ready" output from "E" is sent to the CAMAC "clock" line. Data from "F" goes to "D IN" on the "Serial Crate Controller" and the signals feeding into 100Ω have amplitudes in excess of 200mV. The "D OUT" data from the SCC are one byte behind the "D IN" and either echo the input or are overwritten by information from the controller when it is active (Refs. 2,5). Thus data signals (>200mV) come via the balanced line receiver, "C", 22 baud intervals after the computer signal started, the CAMAC bus clock pulse triggers "G" which causes the incoming data to be loaded into the transmitter buffer register via "D" ("Q" is still off). The CAMAC bus clock is controlled by the "ready" signals in this unit, giving a byte rate of 11 baud intervals.

SOFTWARE COMPATIBILITY

17.   A considerable amount of software exists for the spectrometers but nearly all CAMAC operations are carried out using a set of handling routines. These routines, which occupied about 1 page of memory on the original system, were rewritten for the serial highway interface. The new routines used exactly the same calling sequence as those they replaced and thus no changes were needed to the bulk of the software. The new CAMAC handler programs occupied about half a page more memory mainly due to the need to transmit data 6 bits at a time rather than 12 bits and the need to calculate longitudinal parity.

18.   The main CAMAC subroutine uses a directory which is a table of values giving the position of the module in its crate and the sub-address that will identify the required register or operation. Two parameters are required from the calling program:

(i)   Directory address in the AC.

(ii)   CAMAC function is specified in the location following the subroutine call.

Data is taken from or put into two locations "CAMACH" and "CAMACL".

19.   To simplify the addressing of sets of modules the CAMAC routine may be called, first with the first directory location in the AC and subsequently with the AC set to zero: this causes the directory pointer to increment to the subsequent entries.

THE "CAMAC" SUBROUTINE

20.   The operation of the routine is shown in the flow chart (figure 4). Upon entry the content of the AC is examined, if it is zero the directory pointer is incremented, otherwise it is deposited in the pointer.

21. The hardware attention register is not serviced by the interface electronics until all 12 bytes of a message have been dealt with (c.f. para. 16). Attention data coming in too fast will be lost during a message, however if its period is greater than 26 byte intervals it will always be accepted. Data occurring with a period between 13 and 26 byte intervals may be dealt with by synchronising the CAMAC routine: by waiting for an event before sending the message.

22. Only data from the attention register needs to be dealt with by the interrupt service program and so while the CAMAC routine is transmitting, the interface interrupt is disabled.

23. At this point in the program, bytes are put onto the serial highway; to ensure that the crate controller can clear itself of error conditions a space byte and two wait bytes are sent before the proper message bytes. The crate number, module address and sub-address are held in the directory entry. A few program locations are saved in the spectrometer system since only a single CAMAC crate is used, by making the crate byte a fixed value. The directory entry is accessed to obtain the sub-address which is sent with the M1, M2 bits as zero to give a command message identifier. As each byte is sent, the odd parity bit is calculated and inserted and the running "longitudinal" parity is computed by an exclusive-or subroutine. The third byte is constructed using the specified function code from call +1. The directory is accessed again and the module address is obtained to be sent in the fourth byte. If a dataless function or a read function is to be executed the command message is completed at this point by transmitting a byte containing longitudinal parity. For a write operation the 24 bits of data are transmitted in 4 bytes, 6 bits at a time prior to sending the longitudinal parity byte.

24. At this point the reply from the crate controller must be obtained; since there is a delay of a number of bytes that may be introduced into the line by the interface and controls the next received byte is not necessarily part of the reply message. In any case data cannot be transmitted by the SCC unless it receives a clocking signal: this is in the form of a "space" byte (code 277) during a message, or an "end" byte (code 340) once the reply data has been received. The returned character is checked by the program and when it corresponds to the byte sent as crate address the program starts to process the reply message. If no reply message is received before the 13th space byte is sent the program assumes that some error has occurred and restarts its transmission.

25. The byte following the crate byte in the reply contains a general error flag in the least significant bit. Q and X that are also available at this point are not needed in this application and are ignored by this version of the program. If the error flag is set, it has proved sufficient in practice to restart the transmission since this will overcome transitory failures while catastrophic failure requires user intervention and will be obvious to the user.

26. If the reply is to a read command then the next 4 bytes contain the 24 bits of data and are dealt with by the program. The third (seventh) byte of the reply message should be the end sum byte which contains the longitudinal parity of this message. It is convenient, however, for the program to recognise the end bit (7) as a '1' in case an error occurred and extra bytes were returned. The longitudinal parity of the reply message in the 6 least significant bits of the end-sum byte is checked against the parity calculated by the program as each reply byte was received. If the parity is different an error is assumed and transmission is restarted, otherwise the line is "cleared out" by transmitting bytes with a value zero and waiting until this is echoed back. This is necessary to avoid receiving echoed bytes in the interrupt service routine which would treat any bits set as coming from the attention register if the interrupt is re-enabled too soon.

27. The final operations of the CAMAC routine are to clear the hardware flag, re-enable the interrupt and then release the subroutine for use by other programs in the multi-program environment.

INTERRUPT SERVICE ROUTINE

28. The flow chart (figure 5) and the coding shown in the Appendix give an example of the interrupt service program suitable for use with the interface. It can be seen that this program is only concerned with the attention register, any data originating from the controller is expected to have bit 7 set (a further protection to the transmission of zeroes described above) and is ignored. The attention register has bit 8 allocated to a clock that normally operates at 1Hz or 10Hz and is used by the computer to make sure that any operation is completed

within a reasonable time - for mechanical movements, about 6 minutes. Bit 7 of the attention register is not used in this application while bit 6 is set by an overflow on a 7039 counting register producing an "A" output. Bits 1-5 come up when the individual mechanical positionings are completed. These bits are sensed by the interrupt service program and the appropriate software stimulii are made to other programs in the system to initiate further actions.

29. It will be noted that a synchronising register (CAMAD) is set to −1 by this program. This register may be used by the "CAMAC" subroutine to wait for clock interrupts before commencing to transmit command messages (see para. 21). In the application using these programs the transmission baud rate is 2400 baud corresponding to 218 bytes/second; thus for a clock speed of 10Hz there is a chance that pulses will be lost. The clock pulses are used only for time-out and the loss of a few pulses during the infrequent CAMAC messages is unimportant, program space can be saved by omitting the synchronising increment at the start of the 'CAMAC' subroutine. "CAMAD" has been set before rather than after detecting the clock bit for convenience in programming, this gives a further but very occasional error.

MAINTENANCE

30. The recovery procedures in the CAMAC programs normally mask transitory faults and it would be virtually impossible to determine the causes of these errors. Only when these faults occur very frequently is it practical to investigate and rectify them.

31. In many cases spectrometer programs are self-checking: for example a persistent error in the R or W data lines causes shafts to mis-position and produces measurements that are statistically inconsistent. Since the occurrence of faults is recognised when the results are inspected every few hours in these experiments there is no need for special on-line test programs.

32. Maintenance is carried out using a standard rig that will test all the units used with the spectrometers. A dataway indicator or lamp display is adequate for testing the operation of both SCC and interface and a simple program loop to flash lamps can be used to isolate faults on the interface unit.

33. The complexity of the SCC makes local repair uneconomic and a go/no go test is used - if the unit is faulty it is returned to a central repair depot.

CONCLUSION

34. The serial highway is a very convenient way to connect a CAMAC crate and a computer over any distance for all applications except those needing very high data transfer rates.

35. In a great many CAMAC applications, control of mechanical equipment is involved giving response times in the order of a second and if data collection is involved, this too will be at fairly low rates. In other words a great many CAMAC based systems have requirements that are similar to the triple axis spectrometer and in these cases the simplified interface described in this paper gives lower initial costs and reduces the maintenance effort needed.

36. The program described in this paper demonstrate how the CAMAC serial highway may be used with the simple interface. These programs include features such as the directory and calling sequences that have been found convenient and successful in many years of usage.

37. CAMAC serial crate controllers, connected through the interface described in this paper to PDP-8 computers have been in use on two spectrometer systems and a test rig for over two years. The spectrometers operate at 2400 baud and the test rig uses a link in series with the console teletype at 100 baud. Six interface units have been commissioned using the test rig and location of faulty wiring and ICs has been quite straightforward. There has been one IC failure in service.

38. A hardware design error presented some problems although a patch to the program eliminated any serious effects and the fault could be tolerated in the working systems. The error has since been corrected and the units work very reliably. It is now planned to replace CAMAC controllers on other spectrometers by these Serial Highway units.

REFERENCES

1. "CAMAC - A Modular Instrumentation System for Data Handling, Revised Description and Specification". CEC, Luxembourg, 1972. Report EUR 4100.

2. "Specification of the CAMAC Serial Highway and Serial Crate Controller Type L2". CEC Luxembourg, 1976. Report EUR 6100.

3. "Small Computer Handbook 1973". Digital Equipment Corp.

4. COOPER, M. J., HALL, J. W. and HUTCHINGS, M. W. "An Improved Computer Controlled Triple-Axis Neutron Spectrometer". J. Appl. Cryst. (1976) 9, pp.444-453.

5. GEC-Elliott Process Automation Ltd. Serial Crate Controller SCC 2401, leaflet A2952-5.

6. ABBOTT, D. L. "Shift - A Serial Highway Interface for Teletypes". CAMAC Bulletin No. 19, 1974, pp.19-20.

7. WALLACE-SIMS, G. R. "A CAMAC Shaft Positioning System" (1975), AERE Report R 7299.

TABLE 1

Component Cross Reference for Schematic Diagrams

Simplified Schematic Block Label	Circuit Diagram Component Label
A	IC16, IC19/8, R21, R22, R27, C7
B	IC17, IC19/2, TR2, R23, R18
C	IC28-32, R1-9
D	IC9, IC10
E	IC21
F	IC23-27
G	IC7, R19, C2
H	IC5, IC12/11, TR1, D3, R29-32
J	IC1-4, R11-17
K	IC14, IC15, IC18/8, IC18/6
L	IC20, C5
M	IC12/8, IC11/12
N	IC13/9
P	IC8/8
Q	IC12/6, IC13/12
R	IC11/2, IC12/3, IC6/6, IC6/3
Power Up Resets	R20, C3 and R28, C6

NOTE: Where one section of an IC is used it is designated by its output pin: thus IC12/6 is that gate in IC12 whose output is pin 6.

APPENDIX A

I.C. Component Data Sheets

CONTENTS

- Transmits, receives, formats data
- Full-duplex or half-duplex operation
- Operation from dc to 200 kHz
- Static logic
- Buffered parallel inputs and outputs
- Programmable word lengths — 5, 6, 7, 8 bits
- Programmable information rate
- Programmable parity generation/verification
- Programmable parity inhibit
- Automatic data formatting
- Automatic status generation
- 3-state push-pull buffers
- Low-threshold technology
- Standard power supplies: +5 V, −12 V
- Full TTL compatibility — no external components
- Dual-in-line package — ceramic or plastic

description

The TMS 6011 JC, NC is an MOS/LSI subsystem designed to ensure the data interface between a serial communicator link and data processing equipment such as a peripheral or computer. The device is referred to in the industry as an asynchronous data interface or as Universal Asynchronous Receiver/Transmitter (UART).

The receiver section of the TMS 6011 will accept serial data from the transmission line and convert it to parallel data. The serial word will have start, data, and stop bits. Parity may be generated and verified. The receiver section will validate the received data transmission by checking proper start, parity, and stop bits, and convert the data to parallel.

The transmitter section will accept parallel data, convert it to serial form and generate the start, parity, and stop bits.

Receiver and transmitter sections are separate and the device can operate in full duplex mode.

The TMS 6011 is a fully programmable circuit allowing maximum flexibility of operation, defined as follows:

- The device can operate either in full-duplex (simultaneous transmission and reception) or in half-duplex mode (alternate transmission and reception).
- The data word may be externally selected to be 5, 6, 7, or 8 bits long.
- Baud rate is externally selected by the clock frequency. Clock frequency can vary between 0 to 2½ kHz.
- Parity, which is generated in the transmit mode and verified in the receive mode, can be selected either odd or even. It is also possible to disable the parity bit by inhibiting the parity generation and verification.
- The stop bit can be selected as either a single- or a double-bit stop.
- Static logic is used to maximize flexibility of operation and to simplify the task of the user. The data holding registers are static and will hold a data word until it is replaced by another word.

— continued

operation timing diagram

TRANSMITTER TIMING†

† Transmitter initially assumed inactive at start of diagram, shown for 8 level code and parity and 2 stops.

NOTES: 1. Bit time is 16 clock cycles.
2. If transmitter is inactive the start pulse will appear on line within one clock cycle of time data strobe occurs (see detail below).
3. Because transmitter is double buffered, another data strobe can occur anywhere during transmission of character 1.

RECEIVER TIMING

NOTES: 1. This is the point at which the error condition is detected, if error occurs.
2. Data available is set only when the received data has been transferred to the buffer register. Data available going High also transfers PE, FE, or the status word holding register (see block diagram).
3. All information is good in buffer register until data available tries to set for next character.
4. Above shown for 8-level code, parity and 2-stop. For no parity, stop bits follow data.
5. For all-level code, the data in the buffer register must be right-justified, i.e., RD₁ (pin 12).

operation (continued)

receiver section

The data is received in serial form on the receive input RInput.

The data is presented in parallel form on the eight data outputs RR1 through RR8.

RInput is the data input terminal. The data from RInput enters the receiver register at a point determined by the character length, the parity, and the number of stop bits. RInput must be maintained High when no data is being received. The data is clocked through the RR clock. The clock rate is 16 times faster than the data rate.

Data is transferred from the receiver register to the receiver buffer register and appears on the 8 RR outputs. The MOS output buffers used for the eight RR terminals are 3-state push-pull output buffers which permit the wire-OR configuration through use of the RRDisable terminal. When a logic High is applied to RRDisable, the RR outputs are floating. If the word length is less than 8 bits, the most significant bits will be at a logic low. The output word is right justified. RR1 is the least significant bit and RR8 is the most significant bit.

A logic low applied to the DRReset terminal resets the DReady output to a logic Low.

Several flags are provided in the receiver section. There are three error flags (parity error, framing error and overrun error) and a data-ready flag. All status flags may be disabled through a logic High on the SFDisable terminal.

A logic High on the PError terminal indicates an error in parity.

A logic High on the FError terminal indicates a framing error that is an invalid or nonexistent stop bit in the received word.

A logic High on the OError terminal indicates an overrun. An overrun occurs when the previous word has not been read, i.e., when the DReady line has not been reset before the present data was transferred to the data-receive holding register.

A logic High on the DReady terminal indicates that a word has been received, stored in the receiver-buffer register and that the data is available on outputs RR1 through RR8. The DReady terminal can be reset through the DRReset terminal.

common control section

The common control section will direct both the receiver and the transmitter sections.

The initialization of the TMS 6011 is performed through the MReset terminal. The Master Reset is strobed to a logic High after power turn-on to reset all registers and to reset the serial output line to a logic High.

All status flags (parity error, framing error, overrun error, data ready, transmitter buffer register) are disabled when the SFDisable is at a logic High. When disabled, the status flags float (3-state buffers in high-impedance state).

The number of bits per word is controlled by the WLSelect 1 and WLSelect 2 lines. The word length may be 5, 6, 7, or 8 bits. The selection is as follows:

WORD LENGTH	WLS1	WLS2
5	Low	Low
6	High	Low
7	Low	High
8	High	High

operation (concluded)

common control section (continued)

The parity to be checked by the receiver and generated by the transmitter is determined by the PSelect line. A logic High on the PSelect line selects even parity and a logic Low selects odd parity.

The parity will not be checked or generated if a logic High is applied to PInhibit; in this case the stop bit or bits will immediately follow the data bit.

When a logic High is applied to PInhibit, the PError status flag is brought to a logic Low, indicating a no-parity error because parity is disregarded in this mode."

To select either one or two stop bits, the SBSelect terminal is used. A logic High on this terminal will result in two stop bits while a logic Low will produce only one.

To load the control bits (WLSelect 1, WLSelect 2, PSelect, PInhibit, SBSelect) a logic High is applied to the CRLoad terminal. This terminal may be strobed or hardwired to a logic High.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage V_{DD} range (See Note 1) . . . . −20 V to 0.3 V
Supply voltage V_{GG} range (See Note 1) . . . . −20 V to 0.3 V
Clock input voltage range (See Note 1) . . . . −20 V to 0.3 V
Data input voltage range (See Note 1) . . . . −20 V to 0.3 V
Operating free-air temperature range . . . . −25°C to 85°C
Storage temperature range . . . . −55°C to 150°C

NOTE 1: These voltage values are with respect to V_{SS} (substrate).

recommended operating conditions

PARAMETER	MIN	NOM	MAX	UNIT
Operating Voltage				
Substrate supply, V_{SS}	+4.75	5.0	5.25	V
Drain supply, V_{DD}		0		V
Gate supply, V_{GG}	−12.5	−12	−11.5	V
Logic Levels (See Note 2)				
Input High level, V_{IH}	V_{SS} −1.5		V_{SS} +0.3	V
Input Low level, V_{IL}	−12		0.8	V
Clock Voltage Levels (See Note 2)				
Clock High level, $V_{\phi H}$	V_{SS} −1.5		V_{SS} +0.3	V
Clock Low level, $V_{\phi L}$	−12		0.8	V
Clock Frequency (See Note 3)	dc		200	kHz

NOTES: 2. All data, clock, and command inputs have internal pull-up resistors to allow direct clock by any TTL logic circuit.
3. Clock frequency is 16 times baud rate.

– continued

MOTOROLA Semiconductors

MC14411L
MC14411P

Advance Information

McMOS
(LOW-POWER COMPLEMENTARY MOS)

BIT RATE GENERATOR

BIT RATE GENERATOR

The MC14411 bit rate generator is constructed with complementary MOS enhancement mode devices. It utilizes a frequency divider network to provide a wide range of output frequencies.

A crystal controlled oscillator is the clock source for the network. A two-bit address is provided to select one of four multiple output clock rates.

Applications include a selectable frequency source for equipment in the data communications market, such as teletypes, printers, CRT terminals, and microprocessor systems.

- Single 5.0 Vdc (± 5%) Power Supply
- Internal Oscillator Crystal Controlled for Stability (1.8432 MHz)
- Sixteen Different Output Clock Rates
- 50% Output Duty Cycle
- Programmable Time Bases for One of Four Multiple Output Rates
- Buffered Outputs Compatible with Low Power TTL
- Noise Immunity = 45% of V_{DD} Typical
- Diode Protection on All Inputs

L SUFFIX
CERAMIC PACKAGE
CASE 684

P SUFFIX
PLASTIC PACKAGE
CASE 709

PIN ASSIGNMENT

1	F1	V_{DD} 24
2	F3	RS_A 23
3	F5	RS_B 22
4	F7	$Xtal_{in}$ 21
5	F8	$Xtal_{out}$ 20
6	F10	F16 19
7	F9	F15 18
8	F11	F2 17
9	F14	F4 16
10	Reset	F6 15
11	Not Used	F12 14
12	V_{SS}	F13 13

V_{DD} = Pin 24
V_{SS} = Pin 12

MAXIMUM RATINGS (Voltages referenced to V_{SS}, Pin 12.)

Rating		Symbol	Value	Unit
DC Supply Voltage Range	MC14411L MC14411P	V_{DD}	5.25 to –0.5	Vdc
Input Voltage, All Inputs		V_{in}	V_{DD} + 0.5 to V_{SS} –0.5	Vdc
DC Current Drain per Pin		I	10	mAdc
Operating Temperature Range	MC14411L MC14411P	T_A	–40 to +85	°C
Storage Temperature Range		T_{stg}	–65 to +150	°C

BLOCK DIAGRAM

Rate Select$_A$ 23
Rate Select$_B$ 22
Crystal$_{in}$ 21
*Crystal$_{out}$ 20
Oscillator Circuit
÷2 Divider
X1 X8 X16 X64
Rate Select Logic
Dividers
Reset 10

o 1 F1
o 17 F2
o 2 F3
o 16 F4
o 3 F5
o 15 F6
o 4 F7
o 5 F8
o 7 F9
o 6 F10
o 8 F11
o 14 F12
o 13 F13
o 9 F14
o 18 F15
o 19 F16

*See Figure 2 for typical crystal oscillator circuit.

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{DD}$.
Unused inputs must always be tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{DD}).

McMOS is a trademark of Motorola Inc.
This is advance information and specifications are subject to change without notice.

APPLICATIONS INFORMATION

Typical applications of the Bit Rate Generator (BRG) include providing standard clock frequencies for data communications equipment, and external synchronization of a BRG output to a data source. The synchronization is accomplished by releasing the $\overline{Reset}$ input of the BRG during a data transition of the data source.

A typical data communication system is shown in Figure 3. In this example a standard frequency from the BRG is used for the clock input to the terminal transmitter and receiver (MC2257, MC2259). In a similar system the BRG, via Rate Select inputs, can provide up to 64 standard data communications frequencies for a multiple frequency system. Some examples of equipment frequency requirements are shown in Table 2.

FIGURE 3 – TYPICAL DATA COMMUNICATION TERMINAL BLOCK DIAGRAM
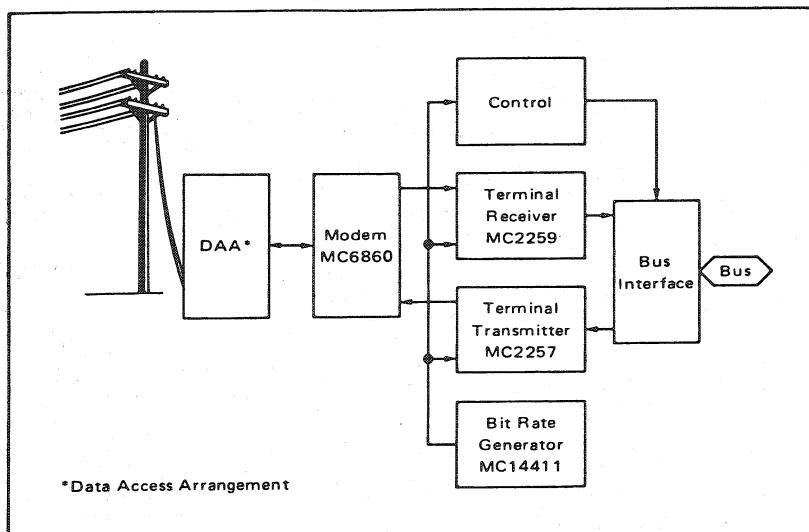
TABLE 2 – TYPICAL DATA COMMUNICATION EQUIPMENT BIT RATE FREQUENCIES

Frequency (Hz)	Use
75	Asynchronous Mode
110	Teletypes
134.49	Printers, typewriters
150	CRT Terminals
200	etc.
300	
600	Asynchronous Mode (high speed)
1200	Printers
	CRT Terminals
	(i.e., Credit Card Verification,
	Personal Bank Checks, etc.)
2400	Synchronous Mode
3600	Such as Communication from
4800	Computer to Computer or
7200	Computer to Peripheral
9600	

MOTOROLA *Semiconductor Products Inc.*

V-PAC*

COMPONENT SPECIFICATIONS

Reliability, Inc.

A POWER SOURCE FOR MOS CIRCUITS

- Compact
- Reliable
- Uses standard +5v source voltage
- Can be plugged into IC sockets, flow soldered
- Reduces manufacturing costs
- Available in single or dual output models
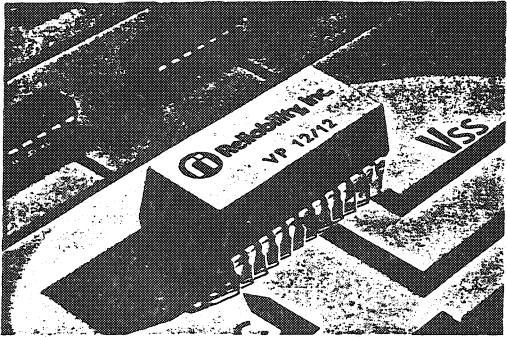
GENERAL DESCRIPTION:

The V-PAC component supplies the special voltages required for MOS devices from the standard + 5 volt supply used for bipolar ICs. Thus, when designers combine both bipolar and MOS devices in the same circuit or system, only one power supply is required. This reduces problems in package design, PC card design, shielding, and many other areas. It also materially reduces manufacturing costs, and increases reliability.

Packaging of the V-PAC devices is in a molded filled thermosetting resin. Size and pin spacing is completely compatible with modern card layout and assembly standards. Height of the device is such that it allows use on cards which must be plugged into racks designed for half-inch centers.

APPLICATIONS:

In any application where MOS devices, including static and dynamic shift registers, ROMs, and RAMs, must be combined with standard bipolar ICs, the V-PAC component may be used to advantage. Benefits derived include reduced noise pickup, critical to reliable high speed logic circuitry, through use of short power supply distribution lines, since the source is on the same card, and usually in close proximity to, the MOS device which it powers.

PIN CONNECTION CHART AND PACKAGE DIMENSIONS:

```
                                     ⑬ ◄─── Ground
                                     ⑪ ◄─ + ⎫
                                            ⎬ Output 1
                                     ⑩ ◄─ − ⎭
                                     ⑨ ◄───
                                     ⑧      NOTE: Strap connections
                                                  shown are
                                     ⑦            internal.
                                     ⑥
                                     ⑤

                                     ◄─── Pin Omitted

                                     ③ ◄─ + ⎫
                                            ⎬ Output 2
                                     ② ◄─ − ⎭
                                     ① ◄─── + 5 V input
```

1.25" nom.   1.10"   .60"
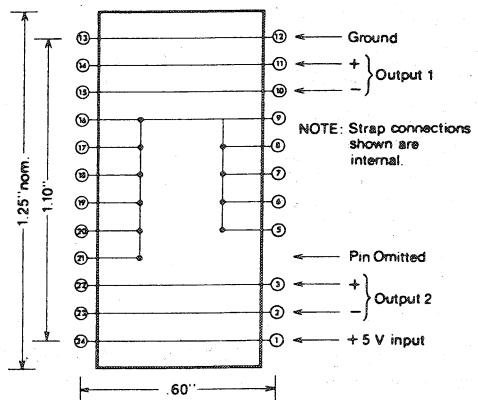
HEIGHT ABOVE MOUNTING PLANE   ³/₈ MAX.
LEAD LENGTH FROM MOUNTING PLANE   ¹/₈ MIN.

TYPICAL APPLICATION:

```
        ┌─────────┐
   5v   │ V-PAC   │  OUTPUT #1 ──► to MOS
        │         │
        │ UNIT    │  OUTPUT #2 ──► to MOS
        └─────────┘
```

3-72
*Trademark, Reliability, Inc.
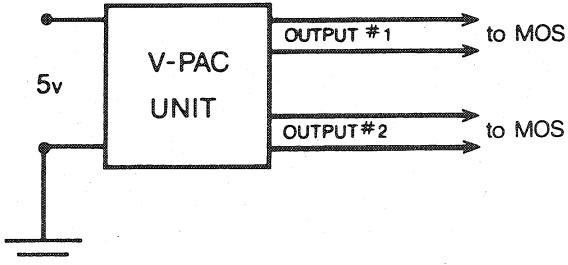
ABSOLUTE MAXIMUM RATINGS:

Input voltage: 5.5 v peak
Free air operating and storage temperature: 0 to 70°C
Total load current:

VP5	200 mA DC	VP10	100 mA
VP12	80 mA DC	VP14	70 mA
VP12/12	80 mA DC	VP15	65 mA
VP14/14	70 mA DC	VP15/15	65 mA

TYPICAL LOAD CURVES:
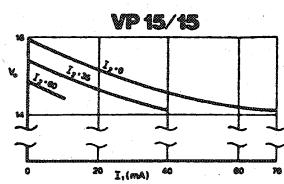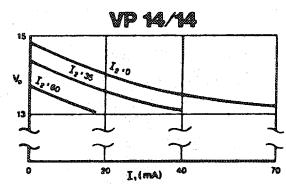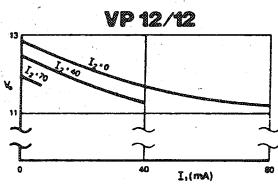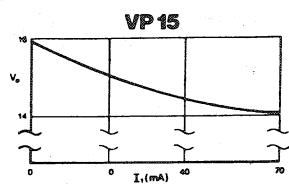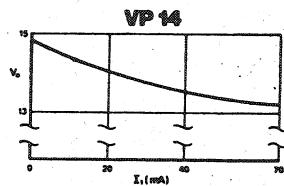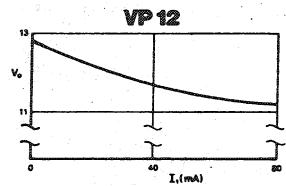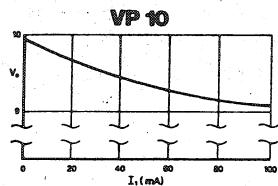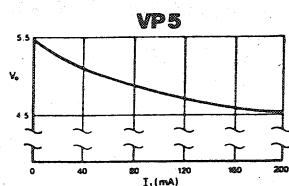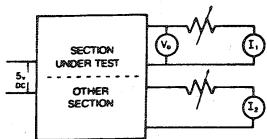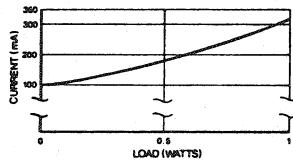
CHARACTERISTICS:

at 25° C, 5 v DC

	SINGLE OUTPUT SOURCES					DOUBLE OUTPUT SOURCES						Units	Conditions
	VP5	VP10	VP12	VP14	VP15	VP12/12		VP14/14		VP15/15			
Maximum Output Voltage	5.5	10.9	13	15	16	±13	±13	±15	±15	±16	±16	v DC	No load
Minimum Output Voltage	4.5	9.1	11	13	14	±11	±11	±13	±13	±14	±14	v DC	Total loads at maximum rating: Any combination
Maximum Ratio Output Change/ Input Change	1.2	2.5	3.2	4.0	4.2	3.2	3.2	4.0	4.0	4.2	4.2	v DC/v DC	Any load
Maximum D. C. Output Resistance	7.5	12	28	40	50	28	28	40	40	50	50	ohms	Input impedance = 0. Any load
Output Voltage —Temperature Coefficient	.004	.006	.006	.006	.007	.006	.006	.006	.006	.007	.007	v DC/°C	Any load
Maximum Input Current	400	400	400	400	400	400		400		400		DC mA	Total loads at maximum rating: Any combination
Maximum R.M.S. Noise	50	50	50	50	50	50		50		50		mv	Reducible by external capacitor, 0.1 uf, maximum

7430
NAND

75185

75183
Dual

7493
4 Bit Binery
Counter

7400
Quad 2 Input
NAND

7404
Hex Inverter

7420
Dual 4 Input
NAND

TIL111
OPTO Coupled

74157
QUAD 2-1
MULTIPLEXER

74122
Retriggerable Mono
with Clear

74473
Dual J-K
Master-Slave
Flip-Flops

AERE-R9115 - IC PIN CONNECTIONS
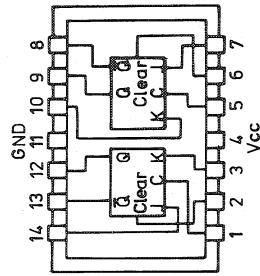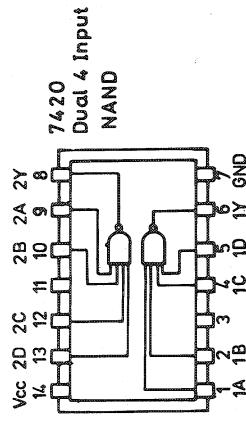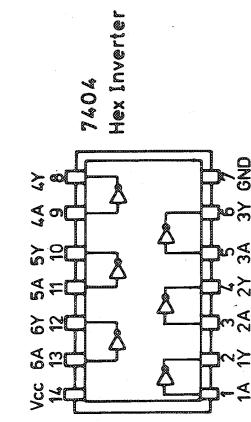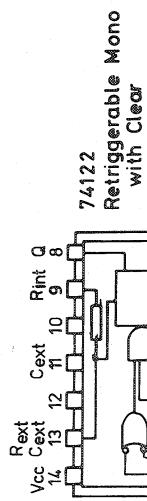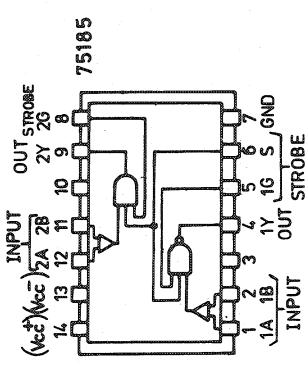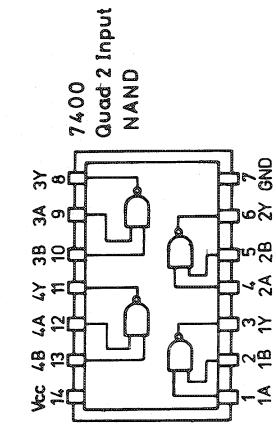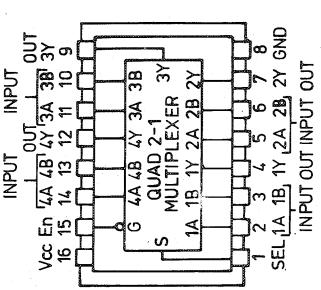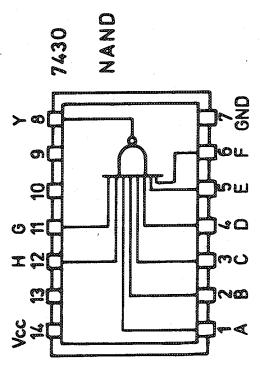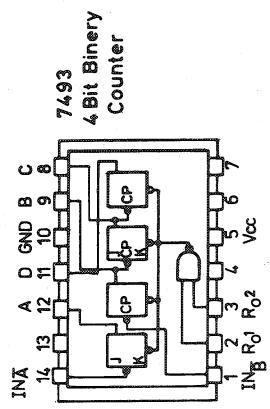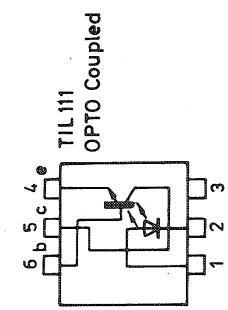
APPENDIX B

CAMAC Serial Highway Program Listing

```
KIE    = 6035
KIEA   = 6405

                        / CAMAC  ASYNCH  LINE  IOTS

TCFC   = 6432           / SEND
TLSC   = 6436           / SKIP IF FLAG IS SET
TSFC   = 6431           / RECEIVE DATA
KRBC   = 6426           / SKIP IF RECEIVE FLAG IS SET
KSFC   = 6421           / DISABLE(AC=0) OR ENABLE(AC=1) CAMAC INTERR
KIEC   = 6425

                        / CAMAC  COMMAND  CODES

CREAD  = 0
CWRITE = 20
CAWSR  = 21             / WRITE TO CAMAC CONTROLLER STATUS REGISTER
DPUL   = 20
DPUSP  = 22
DPUG   = 36

                        *66

66     0       CAMACH, 0
67     0       CAMACL, 0
70     0       CAMACA, 0
71     0       CAMACF, 0
72     0       CAMAD , 0

73     1600    PCAMAC, CAMAC
74     2000    PCINIT, CINIT
75     1474    PLPROC, LPROC
76     1474    PSFOK , SFOK

77     37      P37   , 37
100    40      CAM40 , 40
101    77      P77   , 77
102    200     P200  , 200
103    4000    P4000 , 4000
```

```
                        *1200

                        / THIS PAGE CONTAINS USER ROUTINE DEMONSTRATING THE WAY THAT THE
                        // CAMAC SUBROUTINES MAY BE USED
                        // DUMMY ROUTINES SIMULATE ELEMENTS OF AN OPERATING SYSTEM(SIGNPOSTS)
                        / /THIS EXAMPLE OPERATES WITHOUT USING INTERRUPTS

                        / CWX
                        // PROGRAM TO DRIVE A SHAFT VIA THE CAMAC SERIAL HIGHWAY CONTROLLER
1200  4474   CWX   JMS   I   PCINIT    / SET CAMAC ON LINE
1201  4226         JMS   KEYLK         / GET KEYBOARD COMMAND
1202  213          213
1203  5250         JMP   CSTOP         / CTRL/K - STOP
1204  307          307
1205  5254         JMP   CGO           / G - GO
1206  332          332
1207  5266         JMP   CZ            / Z - SET ANGLE TO ZERO
1210  253          253
1211  5272         JMP   CPLUS         / + - GO ON 1 BIT
1212  255          255
1213  5276         JMP   CMINUS        / - - BACK 1 BIT
1214  317          317
1215  5302         JMP   CON           / O - ON 100 BITS
1216  302          302
1217  5306         JMP   CBACK         / B - BACK 100 BITS/
1220  301          301
1221  5312         JMP   CANGL         / A - SET TO 130 DEGRREES
1222  322          322
1223  5260         JMP   CWRED         / R - DISPLAY DPU CONTENT
1224  0            0                   / END OF COMMAND CODES
1225  5201         JMP   CWX           / NOT FOUND - WAIT ON
```

```
                        / KEYLK
                        // GETS CHARACTER TYPED AT CONSOLE AND COMPARES WITH LIST FOLLOWING SR
                        / CALL - IF MATCH RETURN TO DISPATCH AT

1226    0       KEYLK , 0
1227    7300            CLA
1230    6031            KSF
1231    5230            JMP   .-1
1232    6036            KRB
1233    7000            NOP
1234    3247            DCA   KEYLKA

1235    1626    KEYLK1, TAD I  KEYLK      / LIST CHARACTER
1236    7450            SNA
1237    5626            JMP I  KEYLK      / END
1240    7041            CIA
1241    1247            TAD   KEYLKA
1242    2226            ISZ   KEYLK
1243    7650            SNA               / SAME ?
1244    5626            JMP I  KEYLK      // YES - GO TO DISPATCH
1245    2226            ISZ   KEYLK       / NO - GO TO NEXT
1246    5235            JMP   KEYLK1

1247    0       KEYLKA, 0
```

```
                CSTOP ,   TAD   AAC
1250    1367              JMS   I   ACAMAC
1251    4766              DPUSP            / STOP FUNCTION
1252      22              JMP   CWX
1253    5201

                CGO   ,   TAD   AAC
1254    1367              JMS   I   ACAMAC
1255    4766              DPUG
1256      36              JMP   CWX
1257    5201

                CWRED ,   /
1260    1367              TAD   AAC
1261    4766              JMS   I   ACAMAC
1262       0              CREAD            /READ DPU PUT RESULT IN DISPLAY REGISTER
1263    4766              JMS   I   ACAMAC
1264      20              CWRITE
1265    5201              JMP   CWX

                CZ    ,   CLA
1266    7200              DCA   CANGLH
1267    3344              DCA   CANGLL     / SET ANGLE TO ZERO
1270    3345              JMP   CLOD
1271    5316
```

```
                CPLUS ,  JMS   CADD
1272   4347              JMS   CADD
1273   0                 0
1274   1                 1
1275   5316              JMP   CLOD

                CMINUS,  JMS   CADD        /-1 TO CAMAC
1276   4347         /    JMS   CADD
1277   7777              7777
1300   7777              7777
1301   5316              JMP   CLOD

                CON   ,  JMS   CADD        /ON 1 DEGREE
1302   4347         /    JMS   CADD
1303   0                 0
1304   144               144
1305   5316              JMP   CLOD

                CBACK ,  JMS   CADD        /BACK 1 DEGREE
1306   4347         /    JMS   CADD
1307   7777              7777
1310   7634              7634
1311   5316              JMP   CLOD

                CANGL ,  TAD   CANGLA      /SET TO 130 DEGREES
1312   1342         /    TAD   CANGLA
1313   3344              DCA   CANGLH
1314   1343              TAD   CANGLB
1315   3345              DCA   CANGLL
```

```
               CLOD ,  TAD  CANGLL    / TRANSFER DATA TO CAMAC REGISTERS
1316  1345
1317  3067             DCA  CAMACL
1320  1344             TAD  CANGLH
1321  3066             DCA  CAMACH

1322  1066             TAD  CAMACH    / WAS THE VALUE NEGATIVE ?
1323  7700             SMA  CLA       / NO  - GO TO LOAD CAMAC THEN
1324  5336             JMP  CLODP     / YES - SET UP SIGN BIT WITH MOD DATA
1325  7100             CLL
1326  1067             TAD  CAMACL
1327  7041             CIA
1330  3067             DCA  CAMACL
1331  7024             CML  RAL       / OVERFLOW IF LS PART WAS 0 ORIGINALLY
1332  1066             TAD  CAMACH    / OTHERWISE CANCEL IAC IN NEXT INSTN
1333  7041             CIA
1334  1346             TAD  CLODA     / PUT IN SIGN BIT
1335  3066             DCA  CAMACH

1336  1367  CLODP ,  TAD  AAC
1337  4766             JMS  I  ACAMAC
1340  20              DPUL
1341  5201             JMP  CWX

1342  3     CANGLA ,  3
1343  1310  CANGLB ,  1310
1344  0     CANGLH ,  0
1345  0     CANGLL ,  0
1346  10    CLODA ,   10
```

```
1347    0        CADD,   0
1350    7300             CLA CLL
1351    1747             TAD  I  CADD
1352    2347             ISZ  CADD
1353    1344             TAD  CANGLH
1354    3344             DCA  CANGLH
1355    1747             TAD  I  CADD
1356    2347             ISZ  CADD
1357    7100             CLL
1360    1345             TAD  CANGLL
1361    3345             DCA  CANGLL
1362    7004             RAL
1363    1344             TAD  CANGLH
1364    3344             DCA  CANGLH
1365    5747             JMP  I  CADD

1366    1600    ACAMAC,  CAMAC
1367    2124    AAC   ,  DPU
```

```
                    * 1400

                              / INTERRUPT SERVICE PROGRAM

                    INTPT ,   ISZ   INTFLG        / RE ENTRY?
                              SKP                 / YES
                              JMP   INTS1         / NO
                              DCA   I  PROGNO     / SAVE AC
                              TAD   PROGNO        / FETCH POINTER
                              IAC                 / +1
                              DCA   POYNT
                              TAD   0
                              DCA   I  POYNT      / SAVE PC
                              ISZ   POYNT
                              RIB
                              DCA   I  POYNT      / SAVE INTERRUPT BUFFER
                              ISZ   POYNT
                              RAR
                              DCA   I  POYNT      / SAVE LINK
                    INTS1 ,   CLA   CLL   CMA     / AC=-1
                              KSFC                / SKIP IF RECEIVE FLAG SET
                              JMP   CLRIN         / GO TO CLEAR INTERRUPTS

1400    2274
1401    7410
1402    5217
1403    3674
1404    1274
1405    7001
1406    3274
1407    1000
1410    3674
1411    2274
1412    6234
1413    3674
1414    2274
1415    7010
1416    3674

1417    7340

1420    6421
1421    5264
```

```
/ CAMIN
/ SECTION OF INTERRUPT SERVICE PROGRAM DEALING WITH "CAMAC" INTERRUPTS

CAMIN,
1422  6426    KRBC                / RECEIVE DATA
1423  7104    CLL   RAL           // ROTATE TO LOSE SHAFT 1
1424  3271    DCA   CAMINB        // KEEP INTERRUPT STATUS
1425  1271    TAD   CAMINB        // CHECK IN CASE OF END
1426  0102    AND   P200          // END BIT
1427  7640    SZA   CLA           // SET?
1430  5674    JMP   I   PINTSN    // YES - IGNORE
1431  7240    CLA   CMA           // SET 'CAMAC' INTERRUPT FLAG
1432  3072    DCA   CAMAD
1433  1271    TAD   CAMINB        // GET INTERRUPT BITS
1434  7106    CLL   RTL           // PUT BIT 4 INTO THE LINK
1435  7006    RTL
1436  7430    SZL                 // WAS BIT 4 SET ?
1437  2676    ISZ   I   LCLOCK    // YES - UPDATE CLOCK
1440  5247    JMP   CAMIN1
1441  1677    TAD   I   LSTATU    / REMEMBER STATUS FOR ERROR PRINT OUT
1442  3274    DCA   PRINTE
1443  1677    TAD   I   LSTATU    // ABORT THIS OP BY SETTING PSEUDO COUNT DONE
1444  0270    AND   CAMINA        // REMOVE SHAFT MOVEMENT, START END COUNT BITS
1445  7440    SZA                 // ONLY IF A PROGRAM IS OPERATING ON HARDWARE
1446  5253    JMP   CAMIN2        / AND GO TO SET UP END OF COUNT BIT

CAMIN1,
```

```
1447   7004              RAL
1450   7700              SMA    CLA
1451   5256              JMP    CAMIN3            / WAS BIT 6 OF INTERRUPT REGISTER SET ?
1452   1677              TAD  I LSTATU            / NO
                 CAMIN2,                          / YES - END OF COUNT

1453    273              AND    P7377C           / REMOVE END OF COUNT BIT IF ALREADY SET
1454   1272              TAD    P400             / NOW ENSURE IT IS SET
1455   3677              DCA  I LSTATU
                 CAMIN3,

1456   1271              TAD    CAMINB           / NOW RESET SHAFT MOVEMENT BITS AS REQUIRED
1457    101              AND    P77
1460   7040              CMA                      / COMPLEMENT TO MASK OUT BITS
1461    677              AND  I LSTATU
1462   3677              DCA  I LSTATU
1463   5674              JMP  I PINTSN            / AND CARRY ON

1464   6432    CLRIN ,   TCFC                     / MAKE SURE CAMAC TRANSMIT FLAG IS CLEAR

                                  /              /ENSURE APPROPRIATE INTERRUPTS ARE DISABLED
1465   6035              KIE                      / CONSOLE
1466   6405              KIEA                     / REMOTE PRINTER
1467   5674              JMP  I PINTSN

1470   7000    CAMINA,   7000
1471      0    CAMINB,   0
1472    400    P400 ,    400
1473   7377    P7377C,   7377
```

```
                  / THE FOLLOWING NAMES WOULD APPLY TO REGISTERS AND ROUTINES ELSEWHERE
                  / IN YHE USER PROGRAMS THEY ARE ADDRESSED8 HERE TO A DUMMY REGISTER
                  / TO AVOID ASSEMBLY DIAGNOSTICS

                            DUMMY ,
                            LPROC ,
                            PINTSN,
                            SFOK  ,
                            INTFLG,
                            PRINTE,
                            PROGNO,
1474    0             POYNT , 0
1475    7402          HLT

1476    1474          LCLOCK, DUMMY
1477    1474          LSTATU, DUMMY
```

```
/ CAMAC
/ SUBROUTINE TO TRANSFER DATA TO AND FROM CAMAC UNITS
/ USING THE SERIAL CRATE CONTROLLER
/
/ ***N.B.****
/          THIS VERSION WILL ONLY HANDLE MODULES IN THE MASTER CRATE
/
/ ENTER WITH :-
/     AC = LOCATION OF CAMAC MODULE ADDRESS WITHIN DIRECTORY
/     AC = 0 - GO TO NEXT ITEM IN THE MODULE ADDRESS DIRECTORY
/ CALL + 1 CONTAINS 00FN(OCTAL), WHERE
/     FN = CAMAC FUNCTION
/ THE DIRECTORY LOCATION IS HELD -IN 'CAMACA'
/ THE LOCATIONS 'CAMACH' AND 'CAMACL' ARE USED TO HOLD THE 24 BIT DATA
/ TO BE TRANSMITTED OR RECEIVED
/
/ CALLING SEQUENCE FOR READING FROM A CAMAC MODULE :-
/          IOF
/          JMS SFOK        /BOOK CAMAC ROUTINES AND REGISTERS
/          CAMACF
/          TAD MODIC       /DIRECTORY ENTRY FOR THIS MODULE
/          JMS CAMAC
/          00FN            /CAMAC READ FUNCTION
/          TAD CAMACH      /GET MOST SIGNIFICANT CAMAC DATA BITS
/          DCA KEEPHI
/          TAD CAMACL      /NOW SAVE THE LEAST SIGNIFICANT PART
/          DCA KEEPLO
/          DCA CAMACF      /CLEAR "CAMAC IN USE" FLAG TO RELINQUISH
/          CONTINUE
/
/ CALLING SEQUENCE FOR WRITING TO A CAMAC MODULE
/          TAD DATAHI      /GET MOST SIGNIFICANT 12 BITS TO BE SENT
/          DCA CAMACH      /FOR USE BY THE 'CAMAC' ROUTINES
/          TAD DATALO      /GET THE REMAINING 12 BITS
/          DCA CAMACL
/          IOP

       * 1600
```

```
        JMS SFOK          /BOOK CAMAC ROUTINES AND REGISTERS
        CAMACF
        TAD MODIC         /DIRECTORY ENTRY FOR THIS MODULE
        JMS CAMAC
        0OFN              /CAMAC WRITE FUNCTION
        DCA CAMACF        /CLEAR "CAMAC IN USE" FLAG TO RELINQUISH
        CONTINUE
```

THE DATALESS FUNCTIONS ARE CALLED IN THE SAME WAY BUT THE CONTENT
OF 'CAMACH' AND 'CAMACL' ARE NOT SENT(F=24-31) OR SET UP(CHANGED) FOR
F=8-15

```
1600   0        CAMAC,  0
1601   2070      ISZ  CAMACA      / GOT TO NEXT DIRECTORY ENTRY
1602   7440      SZA              / UNLESS DIRECTORY POSITION WAS IN AC
1603   3070      DCA  CAMACA      / IT WAS - SET IT FOR USE
1604   6425      KIEC             / STOP INTERRUPTS FROM CAMAC

1605   1373     CAMACO, TAD  M14  / SET COUNT FOR RECEIVING CHARACTERS
1606   3374      DCA  CAMACD      / WHILE WAITING IN CASE SYNC IS LOST
1607   4305      JMS  CUNIT       / SEND UNIT NUMBER
1610   1312      TAD  CAMACC      / INITIALISE LONGITUDINAL PARITY WITH BYTE
                                  /JUST SENT

1611   3376      DCA  CAMACY      /
1612   1470      TAD I CAMACA     / GET MODULE SUBADDRESS
1613    372      AND  CA1700      / 4 BITS FOR SA
1614   1100      TAD  CAM40       / M1,M2=0 , SET RESERVED BIT FOR "F" BYTE
1615   1600      TAD I CAMAC      / GET FUNCTION
1616   4766      JMS I ACWRD      / SEND TWO BYTES - SA AND SF
1617   1470      TAD I CAMACA     / GET MODULE ADDRESS
1620    77       AND  P37         / 5 BITS FOR "N"
1621   1100      TAD  CAM40       / SET RESERVED BIT
1622   4315      JMS  CBYT        / SEND BYTE - SN
1623   1600      TAD I CAMAC      / GET FUNCTION
1624   7012      RTR              / GET BITS - F(16) IN LINK, F(8) IN AC0
1625   7012      RTR
1626   7050      CMA RAR          / COMPLEMENT F(16) BIT TO GIVE 'AND'
                                  /CONDITION IN THE NEXT INSTRUCTION
1627   7730      SPA SZL CLA      / BIT F(16) AND F(8)=0 ?
1630   5235      JMP  CAMAC1      / NO
1631   1066      TAD  CAMACH      / YES - GET MS 12 BITS OF DATA
1632   4766      JMS I ACWRD      / TRANSMIT TWO BYTES
1633   1067      TAD  CAMACL      / GET LS 12 BITS OF DATA
1634   4766      JMS I ACWRD      / TRANSMIT TWO BYTES
```

```
                CAMAC1,
1635   1376          TAD   CAMACY        / NOW GET THE COMPUTED LONGITUDINAL PARITY
1636   4315          JMS   CBYT          // PUT IN TRANSVERSE PARITY AND SEND IT
                CAMAC2,
                     /                   /NOW SEND SPACE BYTESTO GET RESPONSE
1637   3376          DCA   CAMACY        / INITIALISE RECEIVED LONGITUDINAL PARITY
1640   4347          JMS   CWBYT         // SEND A SPACE BYTE - GET REPLY
1641   7041          CIA                 // CHECK REPLY FOR CRATE BYTE
1642   1312          TAD   CAMACC        // SAME AS SENT CRATE BYTE
1643   7640          SZA                 // WAS IT CRATE ?
1644   5763          JMP   I ACAMWT      // NO - TEST FOR RECEIVE SYNC OUT
1645   4347          JMS   CWBYT         // SEND A SPACE BYTE - GET REPLY
1646   7010          RAR                 // CHECK ERROR BIT
1647   7630          SZL                 // ANY ERROR ?
1650   5205          JMP   CAMACO        // YES - TRY AGAIN
1651   1600          TAD   I CAMAC       // GET FUNCTION
1652   370           AND   CAM30         // IN CASE IT WAS A READ (F<10(OCTAL)
1653   7640          SZA                 // WAS IT A READ ?
1654   5261          JMP   CAMAC3        // NO
1655   4762          JMS   I ACAMRD      // YES - GET MS 12 BITS OF READ DATA
1656   3066          DCA   CAMACH
1657   4762          JMS   I ACAMRD      / GET LS PART
1660   3067          DCA   CAMACL
                CAMAC3,
1661   1310          TAD   CAMACB        // SEND AN END /WAIT BYTE
1662   4347          JMS   CWBYT         // AND GET REPLY
1663   371           AND   CAM100        // LONGITUDINAL PARITY AND 'END' RECEIVED ?
1664   7650          SNA                 // I.E. BIT 7 OF THE BYTE SET ?
1665   5261          JMP   CAMAC3        // NO - WAIT ON
1666   1376          TAD   CAMACY        // WAS THE CHECKSUM ZERO ?
1667   7640          SZA
1670   5205          JMP   CAMACO        // NO - TRY AGAIN
1671   2200          ISZ   CAMAC         // FOR RETURN
```

```
                SENDO ,  JMS  I  ACSER     / SEND A ZERO
1672    4765             KSFC              // ANY DATA RECEIVED YET ?
1673    6421             JMS  I  ACWAIT    // NO - WAIT VIA 'SIGNPOSTS'
1674    4767             KRBC              // YES - GET IT
1675    6426             SZA CLA           // WAIT FOR ZEROES
1676    7640             JMP  SENDO        // ZEROES NOT YET RECEIVED - LOOP BACK
1677    5272             TCFC              // CLEAR TRANSMIT FLAG
1700    6432             IAC
1701    7001             KIEC              // ALLOW CAMAC INTERRUPTS AGAIN
1702    6425             DCA  CAMACD
1703    3374             JMP  I  CAMAC
1704    5600

                / CUNIT
                / SEND OUT CRATE NUMBER

1705    0        CUNIT ,  0
1706    4764             JMS  I  ACPAR     / MAKE SURE SYNCH OF SERIAL HIGHWAY IS OK
1707    277     CAM277,  277               // SPACE BYTE
1710    340     CAMACB,  340               // WAIT BYTES WILL GET IT IN SYNCHRONISM
1711    340              340
1712    1       CAMACC,  001               / CRATE NUMBER
1713    0                0                 / END OF LIST
1714    5705             JMP  I  CUNIT
```

```
                / CBYT
                // ROUTINE TO SEND A BYTE WITH THE CORRECT (ODD) TRANSVERSE PARITY TO
                // THE SERIAL HIGHWAY AND UPDATE THE LONGITUDINAL PARITY REGISTER 'CAMACY'

                // ENTER WITH DATA IN BITS 6-11 OF THE AC, AC0-5 MAY HAVE
                // ANY VALUE AND IS IGNORED

1715   0     CBYT  ,  0
1716   101            AND   P77        / DATA HAS ONLY 6 BITS
1717   3375           DCA   CAMACX     // REMEMBER DATA
1720   1375           TAD   CAMACX
1721   7100           CLL              / MAKE SURE LINK IS ZERO TO START
                                       /CALCULATE TRANSVERSE PARITY
             CBYT1 ,
1722   7430           SZL              // IS THE CURRENT BIT A "1" ?
1723   1103           TAD   P4000      // YES - CARRY TO NEXT BIT
1724   7104           CLL   RAL        // LOSE OLD BIT - GET NEXT
1725   7440           SZA              // ANY "1" BITS LEFT ?
1726   5322           JMP   CBYT1      // YES - CARRY ON
1727   7420           SNL              // NO - WAS PARITY ODD ?
1730   1102           TAD   P200       // NO - THEN MAKE IT ODD
1731   1375           TAD   CAMACX     // NOW COMBINE IT WITH LS BITS
1732   4765           JMS   I  ACSER   // TRANSMIT THIS BYTE
1733   4335           JMS   CXOR       // NOW GENERATE LONGITUDINAL PARITY
1734   5715           JMP   I  CBYT
```

```
                  / CXOR
                  / SUBROUTINE TO PERFORM THE EXCLUSIVE OR FUNCTION ON THE
                  / DATA HELD IN THE REGISTERS :-
                  /         CAMACX
                  /         CAMACY
                  / AND LEAVE THE RESULT IN THE REGISTER :-
                  /         CAMACY
                  /
                  / THIS IS USED TO GENERATE THE LONGITUDINAL PARITY OF THE BYTE STREAM
                  / TO AND FROM THE CAMAC SERIAL HIGHWAY CONTROLLER
                  /
                  / THE XOR FUNCTION IS USED TO OBTAIN ADDITION WITHOUT CARRIES

1735   0    CXOR  , 0
1736   1375       TAD  CAMACX          / CURRENT VALUE
1737   376        AND  CAMACY          / GET LOCATIONS THAT WILL OVERFLOW
1740   7104       CLL  RAL             / IDENTIFY BIT POSITIONS RECEIVING CARRIES
1741   7041       CIA                  / SUBTRACT THECARRIES FROM THE SUM
1742   1375       TAD  CAMACX
1743   1376       TAD  CAMACY
1744   101        AND  P77             / LIMIT TO LS 6 BITS ONLY
1745   3376       DCA  CAMACY
1746   5735       JMP  I  CXOR
```

```
                        / CWBYT
                        // SUBROUTINE TO TRANSMIT CONTENT OF AC OR A SPACE BYTE (AC=0)
                        // AND GET THE RETURNED BYTE FROM THE SERIAL HIGHWAY
                        // RETURN WITH THE RECEIVED BYTE IN THE AC AND THE
                        / LONGITUDINAL PARITY(RECEIVED) IN 'CAMACY'

1747    0       CWBYT , 0
1750    7450            SNA                 / ANYTHING IN THE AC ?
1751    1307            TAD   CAM277        // NO - GET SPACE CODE
1752    4765            JMS   I  ACSER      // SEND IT
1753    6421            KSFC                // ANY DATA IN YET ?
1754    4767            JMS   I  ACWAIT     // NO - WAIT VIA "SIGNPOSTS"
1755    6426            KRBC                // YES - GET IT
1756    3375            DCA   CAMACX        // REMEMBER IT
1757    4335            JMS   CXOR          / CALCULATE LONGITUDINAL PARITY
1760    1375            TAD   CAMACX
1761    5747            JMP   I  CWBYT
```

```
1762    2015    ACAMRD,  CAMRD
1763    2031    ACAMWT,  CAMWT
1764    2037    ACPAR ,  CPAR
1765    2047    ACSER ,  CSER
1766    2063    ACWRD ,  CWRD
1767    2054    ACWAIT,  CWAIT

1770      30    CAM30 ,   30
1771     100    CAM100,  100
1772    1700    CA1700,  1700
1773    7764    M14   ,   -   14

1774       0    CAMACD,   0
1775       0    CAMACX,   0
1776       0    CAMACY,   0
```

```
                        * 2000

                                  / CINIT
                                  / ROUTINE TO INITIALISE THE SERIAL CAMAC HIGHWAY CONTROLLER

                                  / CALLS THE CAMAC SUBROUTINES TO SET UP THE STATUS REGISTER IN THE
                                  / CAMAC SERIAL HIGHWAY CONTROLLER - IN PARICULAR THIS SETS THE "DOP"
                                  / FLAG TO ZERO TO PUT THE UNIT ON LINE
                                  / THE SEQUENCE OF BYTES TRANSMITTED FOR THIS OPERATION IS :-

                                       340        /CRATE 1
                                       340        /M1, M2=0,  SA=0
                                       340        /F=17  (40(OCTAL)  BIT = 1)
                                       001        /N=30(DECIMAL)   40(OCTAL BIT =1)
                                       200        /DATA =0'S TO STATUS
                                       061        / DOP=0 (BIT 13)
                                       076
                                       200        /LONGITUDINAL PARITY
                                       200        /NOW SPACE BYTES
                                       200
                                       200
                                       016
                                       277
                                       277
                                       277        /TERMINATES
                                       277
                                       277
                                       340

2000     0        CINIT ,  0
2001   6002                 IOF
2002   4476                 JMS  I   PSFOK           / TO BOOK FLAG
2003    71        CAMACF
2004   7301                 CLA  CLL  IAC            / SET AC=1
2005   3067                 DCA  CAMACL              // SET BIT 1 FOR A "Z"
2006   3066                 DCA  CAMACH
2007   1214                 TAD  ACASRA              / GET MODULE ADDRESS POSITION OF STATUS REG
2010   4473                 JMS  I   PCAMAC
2011    21        CAWSR                              / F=17(10)
```

- 39 -

```
2012   3071        DCA    CAMACF
2013   5600        JMP    I   CINIT        / RELEASE CAMAC

2014   2123   ACASRA, CASRA
```

```
/ CAMRD
// SUBROUTINE TO TAKE 2 BYTES FROM THE SERIAL HIGHWAY
// AND ASSEMBLE 12 BITS OF DATA IN THE AC

2015   0        CAMRD , 0
2016   4630        JMS  I  ACWBYT    / SEND A SPACE BYTE
2017   101         AND     P77       / GET MS READ BITS
2020   7106        CLL     RTL
2021   7006        RTL
2022   7006        RTL
2023   3067        DCA     CAMACL    / REMEMBER
2024   4630        JMS  I  ACWBYT    // SEND ANOTHER SPACE BYTE - GET
2025   101         AND     P77       // LS BITS
2026   1067        TAD     CAMACL    / COMBINE WITH MS BITS
2027   5615        JMP  I  CAMRD

2030   1747     ACWBYT, CWBYT


/ CAMWT
// PROGRAM SECTION EXECUTED WHILE WAITING FOR THE REPLY MESSAGE
// FROM THE SERIAL CRATE CONTROLLER
// REPLY SHOULD BE WITHIN A SMALL NUMBER OF CHARACTERS
/ AND SHOULD START WITH THE CRATE NUMBER

2031   2072     CAMWT ,  ISZ  CAMAD     / EXCEEDED CHARACTER COUNT ?
2032   5635        JMP  I  ACAMCI       // NO - WAIT FOR ANOTHER
2033   4277        JMS     CAMINT       // YES
2034   5636        JMP  I  ACAMCO       / START MESSAGE AGAIN

2035   1637     ACAMCI, CAMAC2
2036   1605     ACAMCO, CAMACO
```

```
                    / CPAR

                    / CPAR
                    // SUBROUTINE TO TAKE THE LIST OF CODES FOLLOWING THE CALL POINT
                    // AND TRANSMIT THEM TO THE CAMAC DRIVER
                    // THE LIST IS TERMINATED BY A ZERO VALUE

2037    0     CPAR  , 0
2040    7300  CPAR1 , CLA CLL

2041    1637          TAD  I CPAR      / GET CODE
2042    2237          ISZ  CPAR        / FOR NEXT
2043    7450          SNA              // END?
2044    5637          JMP  I CPAR      // YES - RETURN
2045    4247          JMS  CSER        // NO TRANSMIT THE CODE
2046    5241          JMP  CPAR1       / GET NEXT

                    / CSER
                    // SEND THE DATA IN THE AC, WAIT UNTIL THE TRANSMISSION ISCOMPLETED
                    // CIRCULATING VIA 'SIGNPOSTS'

2047    0     CSER  , 0
2050    6436          TLSC             / SEND IT
2051    6431          TSFC             // GONE ?
2052    4254          JMS  CWAIT       // NO - WAIT
2053    5647          JMP  I CSER      / YES - RETURN
```

```
                        / CWAIT
                        / SUBROUTINE TO SIMULATE "JMP .-1"
                        / USED TO WAIT FOR A HARDWARE SKIP ON FLAG
                        / CIRCULATES VIA "SIGNPOSTS" AND RETURNS TO CALL - 1
                        / AC IS CLEARED BY THIS ROUTINE

2054   0       CWAIT ,  0
2055   7344             CLA   CLL  CMA  RAL    / -2 IN THE AC
2056   1254             TAD   CWAIT           / GENERATE RETURN POSITION
2057   3262             DCA   CWAITA
2060   6002             IOF                   / NOW CIRCULATE ONCE
2061   4475             JMS   I  PLPROC
2062   0       CWAITA,  0                     / RETURN TO CALL-1


                        / CWRD
                        / SUBROUTINE TO TRANSMIT THE CONTENTS OF THE AC
                        / SA TWO BYTES TO THE SERIAL HIGHWAY

2063   0       CWRD  ,  0
2064   3276             DCA   CAMACZ          / REMEMBER VALUE
2065   1276             TAD   CAMACZ
2066   7112             CLL   RTR             / GET MS 6 BITS
2067   7012             RTR
2070   7012             RTR
2071   4675             JMS   I  ACBYT        / SEND
2072   1276             TAD   CAMACZ
2073   4675             JMS   I  ACBYT        / GET LS 6 BITS
2074   5663             JMP   I  CWRD         / SEND

2075   1715    ACBYT ,  CBYT
2076   0       CAMACZ,  0
```

- 43 -

```
                    / CAMINT
                    // MAKE SURE THE SERIAL HIGHWAY CONTROLLER
                    // IS ON-LINE BY SETTING THE"DOF" FLAG TO ZERO
                    // ROUTINE ALSO ALLOWS CAMAC HARDWARE TO CLEAR FAULTS
                    // BY WAITING FOR DATA TO BE SENT BEFORE TRYING TO
                    / RE-ESTABLISH HANDSHAKE COMMUNICATIONS

2077     0    CAMINT,  0
2100  2320         ISZ   CAMINC
2101  4254         JMS   CWAIT          // WAIT SOME MORE TIME ?
2102  4722         JMS I ACUNIT         // YES - CIRCULATE
2103  4237         JMS   CPAR           // NO - CAMAC SHOULD HAVE CLEARED
2104   200         200                  // SEND INITIATING MESSAGE
2105    61         061
2106    76         076
2107   200         200
2110   200         200
2111   200         200
2112   200         200
2113    16         016
2114   277         277
2115   277         277
2116   277         277
2117   340         340
2120     0    CAMINC,  0
2121  5677         JMP I   CAMINT

2122  1705    ACUNIT, CUNIT            / TIME REGISTER - ALSO USED AS MESSAGE DELIM
```

```
/  CAMAC MODULE ADDRESS DIRECTORY
/  THE DIRECTORY ENTRIES IDENTIFY THE MODULE,
/  THE SUB ADDRESS FOR THE DESIRED OPERATION
/  THE CRATE NUMBER OF THE CRATE CONTAING THE MODULE (UP TO 4 CRATES)

/  THE DIRECTORY ENTRY HAS THE FORMAT :-
/          AKNN
/
/  WHERE
/          NN = MODULE ADDRESS
/          AK = 20(OCTAL)* C + SUB ADDRESS
/          C=0 FOR THE MASTER CRATE
/          C=1 FOR THE FIRST SLAVE CRATE
/          C=2 FOR THE SECOND SLAVE CRATE
/          C=3 FOR THE THIRD SLAVE CRATE
/
```

```
2123    36    CASRA  ,  36     / N FOR STATUS REGISTER
2124    22    DPU    ,  22     / DPU MODULE ADDRESS
2125    20    DDR    ,  20     / DISPLAY REGISTER ADDRESS
```

```
                        /  BACK
                        /  SIMPLE TEST PROGRAM TO FLASH LAMPS VIA A DRIVER OUTPUT REGISTER

              BACKG ,    JMS   CINIT         / INITIALISE STATUS REGISTER IN SCC
2126  4200              CLA   CLL   CMA
2127  7340
2130  3066              DCA   CAMACH         / SET UP DISPLAY DATA INITIALLY

              BACK  ,    DCA   CAMACL
2131  3067              TAD   TESTA
2132  1344              JMS   I   PCAMAC     / SET UP DIRECTORY AD
2133  4473              CWRITE
2134    20              CLA
2135  7200              TAD   CAMACH
2136  1066              DCA   TEMP
2137  3345              TAD   CAMACL
2140  1067              DCA   CAMACH
2141  3066              TAD   TEMP           / REVERSE DATA
2142  1345              JMP   BACK
2143  5331

2144  2125    TESTA ,   DDR
2145    0     TEMP  ,   0
                        £
```

***PASS 2 COMPLETE: 0 ERRORS.
                        0 OVERWRITES.

***OUTPUT OPTIONS SPECIFIED PAPERTAPE.

PAL3HCROSS-REFERENCE TABLE***

SYMBOL:	VAL:	REFERENCES:								
AAC	(=1367)	1250	1254	1260	1336					
ACAMAC	(=1366)	1251	1255	1261	1263	1337				
ACAMRD	(=1762)	1655	1657							
ACAMWT	(=1763)	1644								
ACAMCI	(=2035)	2032								
ACAMCO	(=2036)	2034								
ACASRA	(=2014)	2007								
ACBYT	(=2075)	2071	2073							
ACPAR	(=1764)	1706								
ACSER	(=1765)	1672	1732	1752						
ACUNIT	(=2122)	2102								
ACWAIT	(=1767)	1674	1754							
ACWBYT	(=2030)	2016	2024							
ACWRD	(=1766)	1616	1632	1634						
ASR	(=7415)	NOT REFERENCED.								
BACKG	(=2126)	NOT REFERENCED.								
BACK	(=2131)	2143								
CADD	(=1347)	1272	1276	1302	1306	1351	1352	1355	1356	1365
CAMACH	(= 66)	1321	1322	1332	1335	1631	1656	2006	2130	2136
CAMACL	(= 67)	1317	1326	1330	1633	1660	2005	2023	2026	2131
CAMACA	(= 70)	1601	1603	1612	1617					
CAMACF	(= 71)	2003	2012							
CAMAD	(= 72)	1432	2031							
CAMAC	(=1600)	73	1366	1615	1623	1651	1671	1704		
CAMAC0	(=1605)	1650	1670	2036						
CAMAC1	(=1635)	1630	1703							
CAMAC2	(=1637)	2035								
CAMAC3	(=1661)	1654	1665							
CAMACB	(=1710)	1661								
CAMACC	(=1712)	1610	1642							
CAMACD	(=1774)	1606	1703							
CAMACX	(=1775)	1717	1720	1731	1736	1742	1756	1760		
CAMACY	(=1776)	1611	1635	1637	1666	1737	1743	1745	2141	
CAMACZ	(=2076)	2064	2065	2072	2140					

Symbol	Address	References					
CAM	(=7621)	NOT REFERENCED.					
CAMIN	(=1422)	NOT REFERENCED.					
CAMIN1	(=1447)	1440					
CAMIN2	(=1453)	1446					
CAMIN3	(=1456)	1451					
CAMINA	(=1470)	1444					
CAMINB	(=1471)	1424	1425	1433	1456		
CAMINT	(=2077)	2033	2121				
CAMINC	(=2120)	2100					
CAMRD	(=2015)	1762	2027				
CAMWT	(=2031)	1763					
CAM100	(=1771)	1663					
CAM277	(=1707)	1751					
CAM30	(=1770)	1652					
CAM40	(= 100)	1614	1621				
CANGL	(=1312)	1221					
CANGLA	(=1342)	1312					
CANGLB	(=1343)	1314					
CANGLH	(=1344)	1267	1313	1320	1353	1354	1363
CANGLL	(=1345)	1270	1315	1316	1360	1361	
CASRA	(=2123)	2014					
CAWSR	(= 21)	2011					
CA1700	(=1772)	1613					
CBACK	(=1306)	1217					
CBYT	(=1715)	1622	1636	1734	2075		
CBYT1	(=1722)	1726					
CDF	(=6201)	NOT REFERENCED.					
CGO	(=1254)	1205					
CIF	(=6202)	NOT REFERENCED.					
CINIT	(=2000)	74	2013	2126			
CLOD	(=1316)	1271	1275	1301	1305	1311	
CLODP	(=1336)	1324					
CLODA	(=1346)	1334					
CLRIN	(=1464)	1421					
CMINUS	(=1276)	1213					
CON	(=1302)	1215					
CPAR	(=2037)	1764	2041	2042	2044	2103	
CPAR1	(=2041)	2046					

Symbol	Value	Ref 1	Ref 2	Ref 3	Ref 4	Ref 5
CPLUS	(=1272)	1211				
CREAD	(= 0)	1262				
CSER	(=2047)	1765	2045	2053		
CSTOP	(=1250)	1203				
CUNIT	(=1705)	1607	1714	2122		
CWAIT	(=2054)	1767	2052	2056	2101	
CWAITA	(=2062)	2057				
CWBYT	(=1747)	1640	1645	1662	1761	2030
CWRD	(=2063)	1766	2074			
CWRED	(=1260)	1223				
CWRITE	(= 20)	1264	2134			
CWX	(=1201)	1225	1253	1257	1265	1341
CXOR	(=1735)	1733	1746	1757		
CZ	(=1266)	1207				
DCEA	(=6611)	NOT REFERENCED.				
DCMA	(=6601)	NOT REFERENCED.				
DDR	(=2125)	2144				
DEAC	(=6616)	NOT REFERENCED.				
DEAL	(=6615)	NOT REFERENCED.				
DFSC	(=6622)	NOT REFERENCED.				
DFSE	(=6621)	NOT REFERENCED.				
DMAC	(=6626)	NOT REFERENCED.				
DMAR	(=660,3)	NOT REFERENCED.				
DMAW	(=6605)	NOT REFERENCED.				
DPU	(=2124)	1367				
DPUG	(= 36)	1256				
DPUL	(= 20)	1340				
DPUSP	(= 22)	1252				
DSAC	(=6612)	NOT REFERENCED.				
DTCA	(=6762)	NOT REFERENCED.				
DTLB	(=6774)	NOT REFERENCED.				
DTRA	(=6761)	NOT REFERENCED.				
DTRB	(=6772)	NOT REFERENCED.				
DTSF	(=6771)	NOT REFERENCED.				
DTXA	(=6764)	NOT REFERENCED.				
DUMMY	(=1474)	1476	1477			
DVI	(=7407)	NOT REFERENCED.				
FATN	(= 5)	NOT REFERENCED.				

```
FCDF    (=7001)    NOT REFERENCED.
FCOS    (=   4)    NOT REFERENCED.
FCRLF   (=  15)    NOT REFERENCED.
FEXP    (=   6)    NOT REFERENCED.
FEXT    (=   0)    NOT REFERENCED.
FHLT    (=7004)    NOT REFERENCED.
FIN     (=  10)    NOT REFERENCED.
FISZ    (=   0)    NOT REFERENCED.
FIXD    (=  12)    NOT REFERENCED.
FLOG    (=   7)    NOT REFERENCED.
FLOT    (=  14)    NOT REFERENCED.
FNEG    (=  13)    NOT REFERENCED.
FNOP    (=7010)    NOT REFERENCED.
FNOR    (=7000)    NOT REFERENCED.
FOUT    (=  11)    NOT REFERENCED.
FSIN    (=   3)    NOT REFERENCED.
FSKP    (=7020)    NOT REFERENCED.
FSMA    (=7110)    NOT REFERENCED.
FSNA    (=7040)    NOT REFERENCED.
FSPA    (=7100)    NOT REFERENCED.
FSQR    (=   2)    NOT REFERENCED.
FSQU    (=   1)    NOT REFERENCED.
FSW0    (=7002)    NOT REFERENCED.
FSW1    (=7003)    NOT REFERENCED.
FSZA    (=7050)    NOT REFERENCED.
HLT     (=7402)    1475
INTFLG  (=1474)    1400
INTPT   (=1400)    NOT REFERENCED.
INTS1   (=1417)    1402
IOF     (=6002)    2001   2060
ION     (=6001)    NOT REFERENCED.
KCC     (=6032)    NOT REFERENCED.
KEYLK   (=1226)    1201   1235   1237   1242   1244   1245
KEYLK1  (=1235)    1246
KEYLKA  (=1247)    1234   1241
KIEA    (=6405)    1466
KIE     (=6035)    1465
KIEC    (=6425)    1604   1702
```

Symbol	Address					
KRB	(=6036)	1232	1675	1755		
KRBC	(=6426)	1422				
KRS	(=6034)	NOT REFERENCED.				
KSF	(=6031)	1230				
KSFC	(=6421)	1420	1673	1753		
LCLOCK	(=1476)	1437				
LPROC	(=1474)	75				
LSR	(=7417)	NOT REFERENCED.				
LSTATU	(=1477)	1441	1443	1452	1455	1461 1462
MMCC	(=6762)	NOT REFERENCED.				
MMCF	(=6772)	NOT REFERENCED.				
MMLC	(=6764)	NOT REFERENCED.				
MMLF	(=6754)	NOT REFERENCED.				
MMLM	(=6752)	NOT REFERENCED.				
MMLS	(=6751)	NOT REFERENCED.				
MMMF	(=6756)	NOT REFERENCED.				
MMML	(=6766)	NOT REFERENCED.				
MMMM	(=6757)	NOT REFERENCED.				
MMRS	(=6774)	NOT REFERENCED.				
MMSC	(=6771)	NOT REFERENCED.				
MMSF	(=6761)	NOT REFERENCED.				
MQA	(=7501)	NOT REFERENCED.				
MQL	(=7421)	NOT REFERENCED.				
MUY	(=7405)	NOT REFERENCED.				
M14	(=1773)	1605				
NMI	(=7411)	NOT REFERENCED.				
PCAMAC	(= 73)	2010	2133			
PCF	(=6022)	NOT REFERENCED.				
PCINIT	(= 74)	1200				
PINTSN	(=1474)	1430	1463	1467		
PLPROC	(= 75)	2061				
PLS	(=6026)	NOT REFERENCED.				
POYNT	(=1474)	1406	1410	1411	1413	1414 1416
PPC	(=6024)	NOT REFERENCED.				
PRINTE	(=1474)	1442				
PROGNO	(=1474)	1403	1404			
PSF	(=6021)	NOT REFERENCED.				
PSFOK	(= 76)	2002				

| Symbol	Value	References				
P200	(= 102)	1426	1730			
P37	(= 77)	1620				
P4000	(= 103)	1723				
P400	(=1472)	1454				
P7377C	(=1473)	1453				
P77	(= 101)	1457	1716	1744	2017	2025
RDF	(=6214)	NOT REFERENCED.				
RFC	(=6014)	NOT REFERENCED.				
RIB	(=6234)	1412				
RIF	(=6224)	NOT REFERENCED.				
RMF	(=6244)	NOT REFERENCED.				
RRB	(=6012)	NOT REFERENCED.				
RSF	(=6011)	NOT REFERENCED.				
SCA	(=7441)	NOT REFERENCED.				
SCL	(=7403)	NOT REFERENCED.				
SEND0	(=1672)	1677				
SFOK	(=1474)	76				
SHL	(=7413)	NOT REFERENCED.				
TCF	(=6042)	NOT REFERENCED.				
TCFC	(=6432)	1464	1700			
TEMP	(=2145)	2137	2142			
TESTA	(=2144)	2132				
TLS	(=6046)	NOT REFERENCED.				
TLSC	(=6436)	2050				
TPC	(=6044)	NOT REFERENCED.				
TSF	(=6041)	NOT REFERENCED.				
TSFC	(=6431)	2051				
XXX	(= 0)	NOT REFERENCED.				
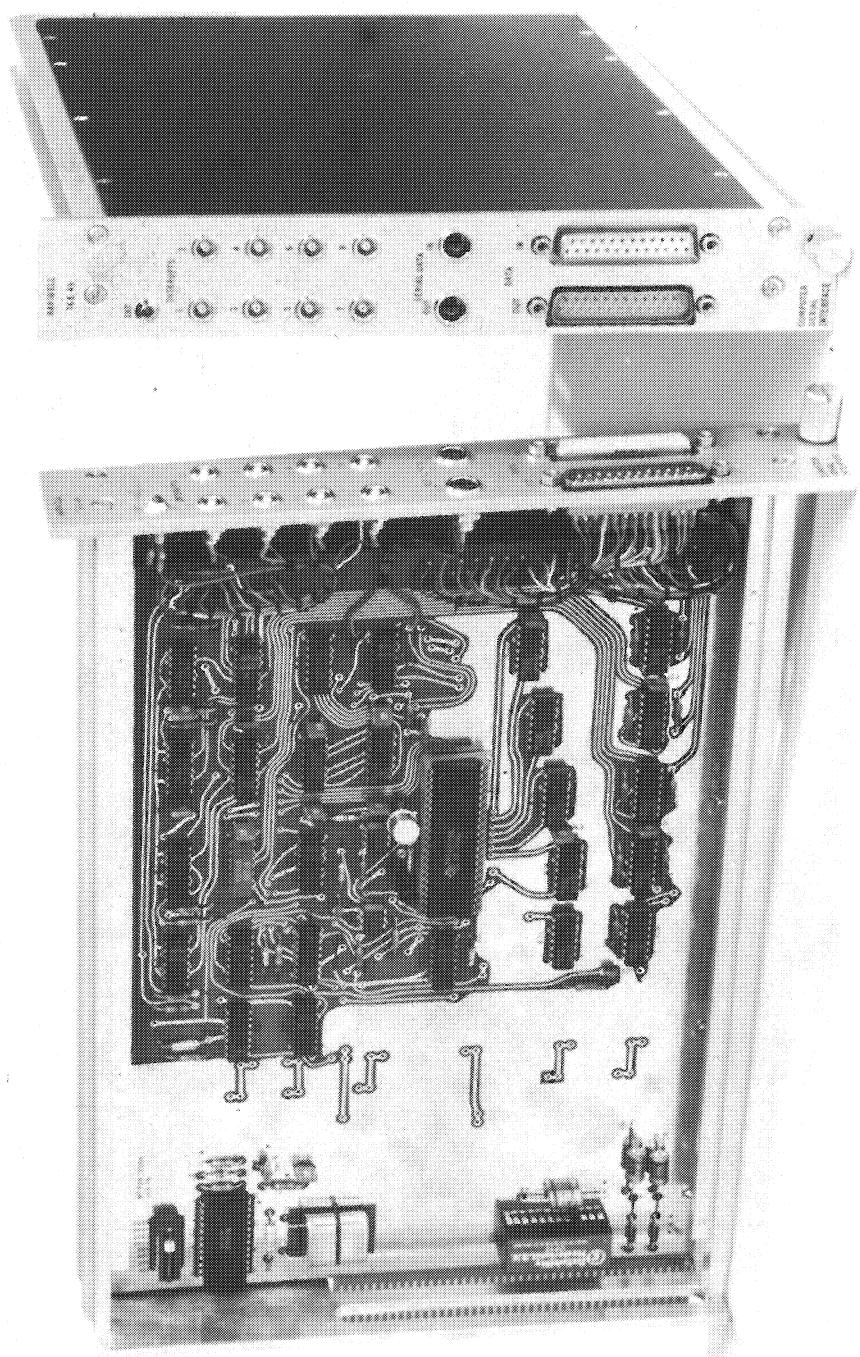
AERE - R 9115   Fig. 1
The SCC interface unit

AERE - R 9115   Fig. 2
Circuit diagram - SCC interface unit
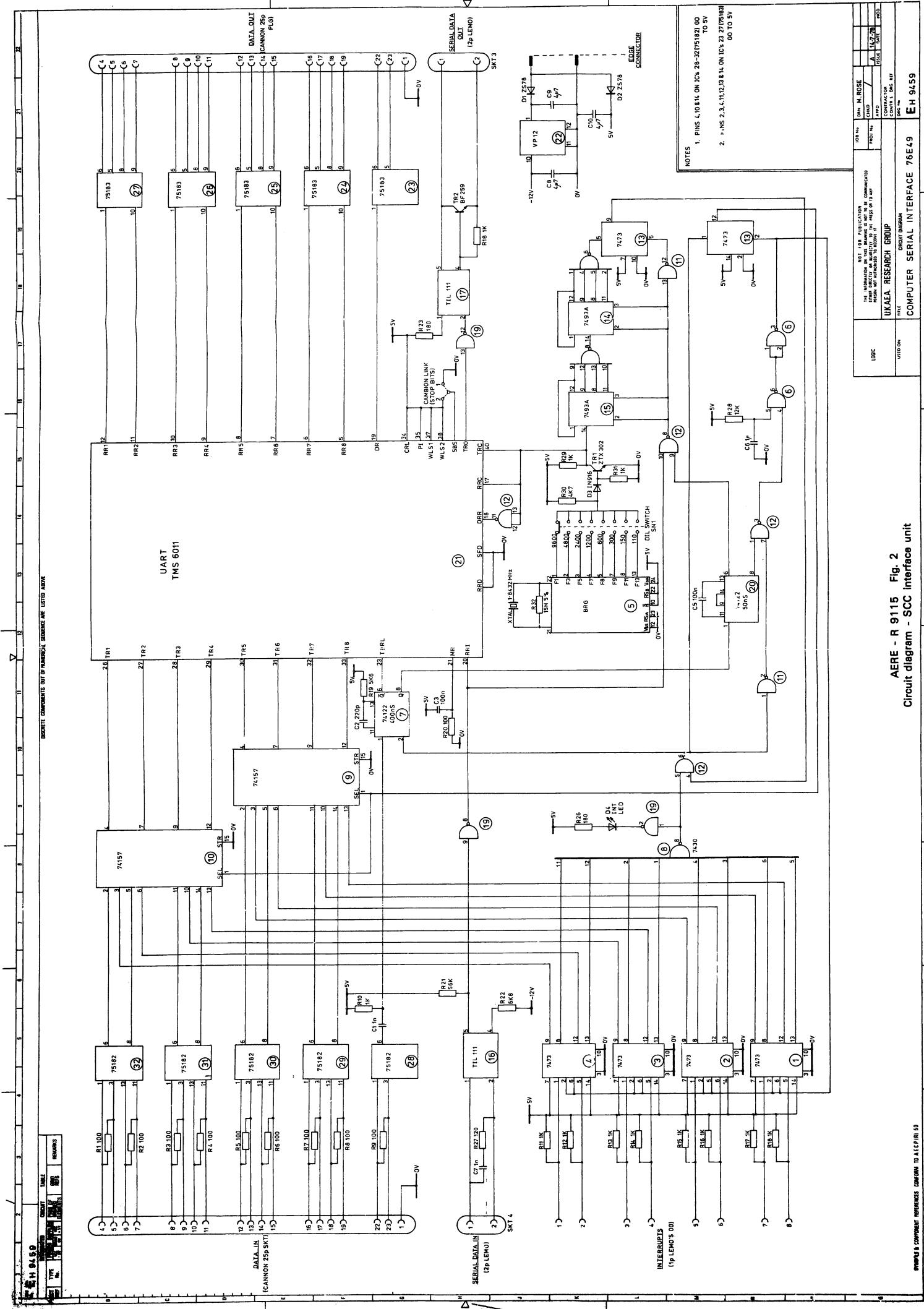
AERE-R9115 FIG.3 SIMPLIFIED SCHEMATIC – SCC. INTERFACE

'CAMAC'

Set up Directory
Pointer and synch.
to Clock Pulses

Disable CAMAC
interrupts

CAMAC Ø

Send 'Wait' bytes
to get hardware
'byte sync'

Fetch call parameters
to calculate and send
4 command bytes

Byte Content
1 SC1 – 5
2 M1, M2, SA1 – 8
3 SF1 – 16
4 SN1 – 16

Read Function ? — No → Send 4 bytes with W1 – W24

Yes

CAMAC 1

Send Computed
Longitudinal
Parity Byte

CAMAC 2

Send a 'Space' byte
Get returned byte

Same Value as (1st) Crate address byte ? — No →

YES

Send a 'Space' byte
Get returned byte

12th 'Space' byte sent ? — No →

YES

Error (L Sig. bit Set) ? — Yes →

No

Read Function ? — Yes → Send 4 'Space' bytes and assemble returned bytes to give R1 – R24

No

CAMAC 3

Send 'End' bytes
until returned byte
has bit 7 set

6 L.Sig bits same as computed value of long Parity ? — No: Error →

SEND Ø Yes

Send zero bytes until
returned byte is zero

Clear flag Hardware
Enable CAMAC Interrupts
Release CAMAC Routine

Exit

AERE-R.9115 FIG.4 CAMAC SUBROUTINE-FLOW CHART
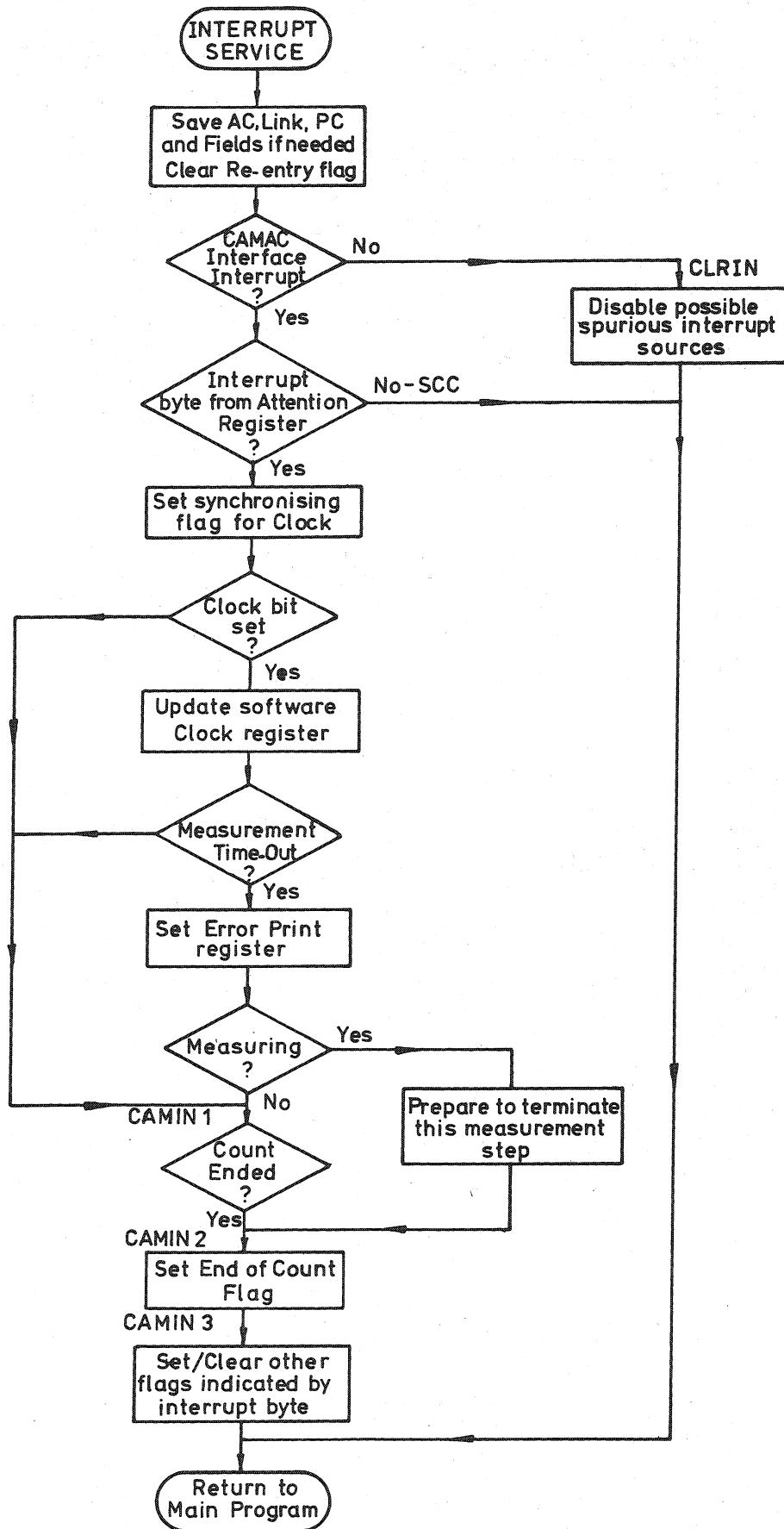
AERE-R.9115 FIG.5 INTERRUPT SERVICE PROGRAM-FLOW CHART
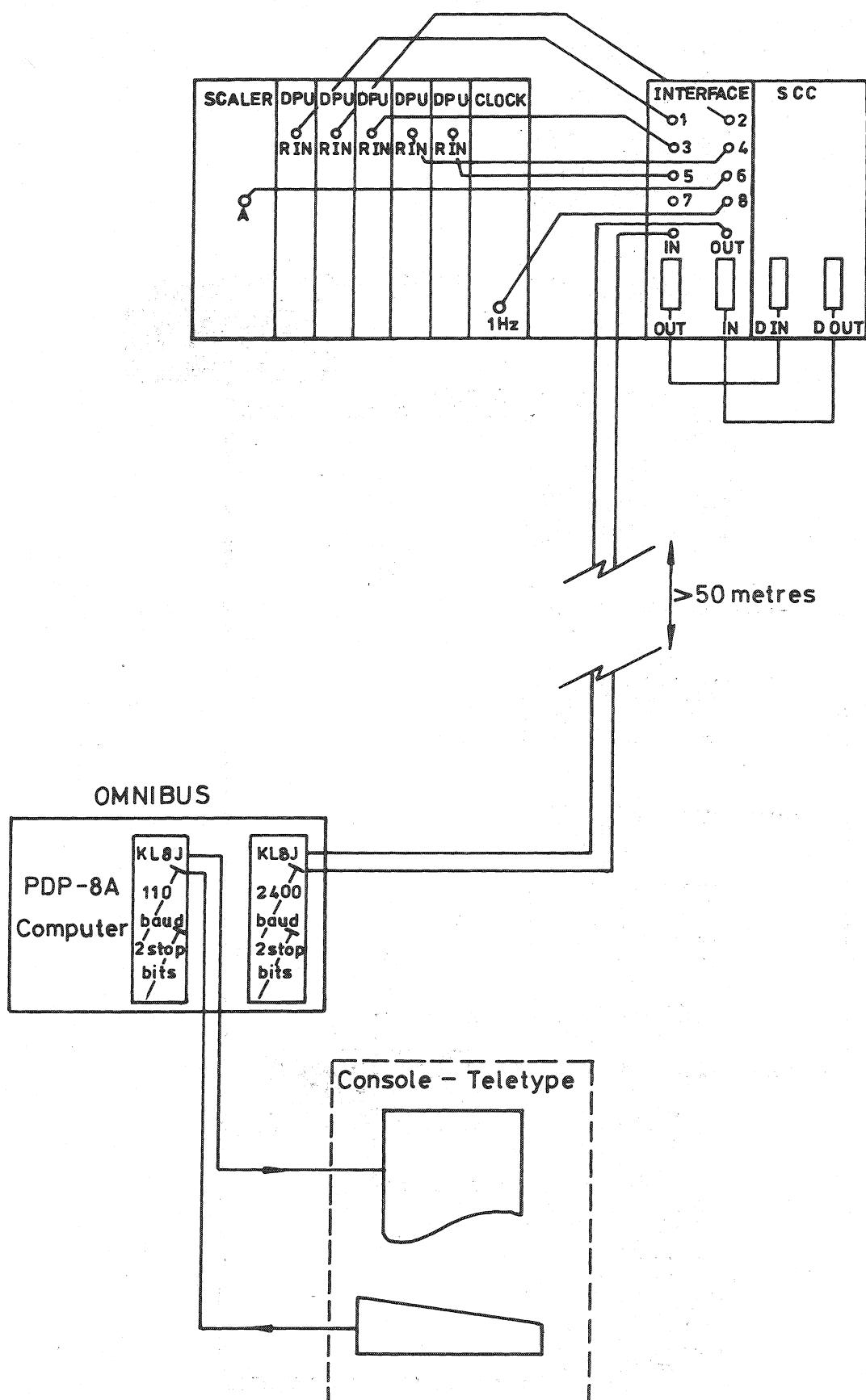
AERE-R9115 FIG.6 COMPUTER TO CAMAC INTERCONNECTION FOR DIDO 3 AXIS SPECTROMETER