SAND80-1952
Unlimited Release

UC-37

ISOLATED DATA ACQUISITION SYSTEM

Gerald C. Stoker, Clifford L. Sharp

Prepared by Sandia Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550 for the United States Department of Energy under Contract DE-AC04-76DP00789.

Printed June 1981

Sandia National Laboratories

SF 2900-Q(3-80)

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Printed in the United States of America

Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

Price: Printed Copy $4.50; Microfiche $3.00

SAND80-1952 UC-37
Unlimited Release
Printed June 1981

ISOLATED DATA ACQUISITION SYSTEM

Gerald C. Stoker
Clifford L. Sharp
Nondestructive Testing Technology Division 1552
Sandia National Laboratories
Albuquerque, New Mexico 87185

ABSTRACT

This report describes a microprocessor controlled data acquisition module which is battery powered and communicates with a master computer via a fiber optic link. This allows the module to be used in hazardous areas with complete electrical isolation. The module allows digital inputs, digital outputs, and analog to digital conversion with analog input multiplexing. A complete description is given along with diagrams, schematics, and program listings.

TABLE OF CONTENTS

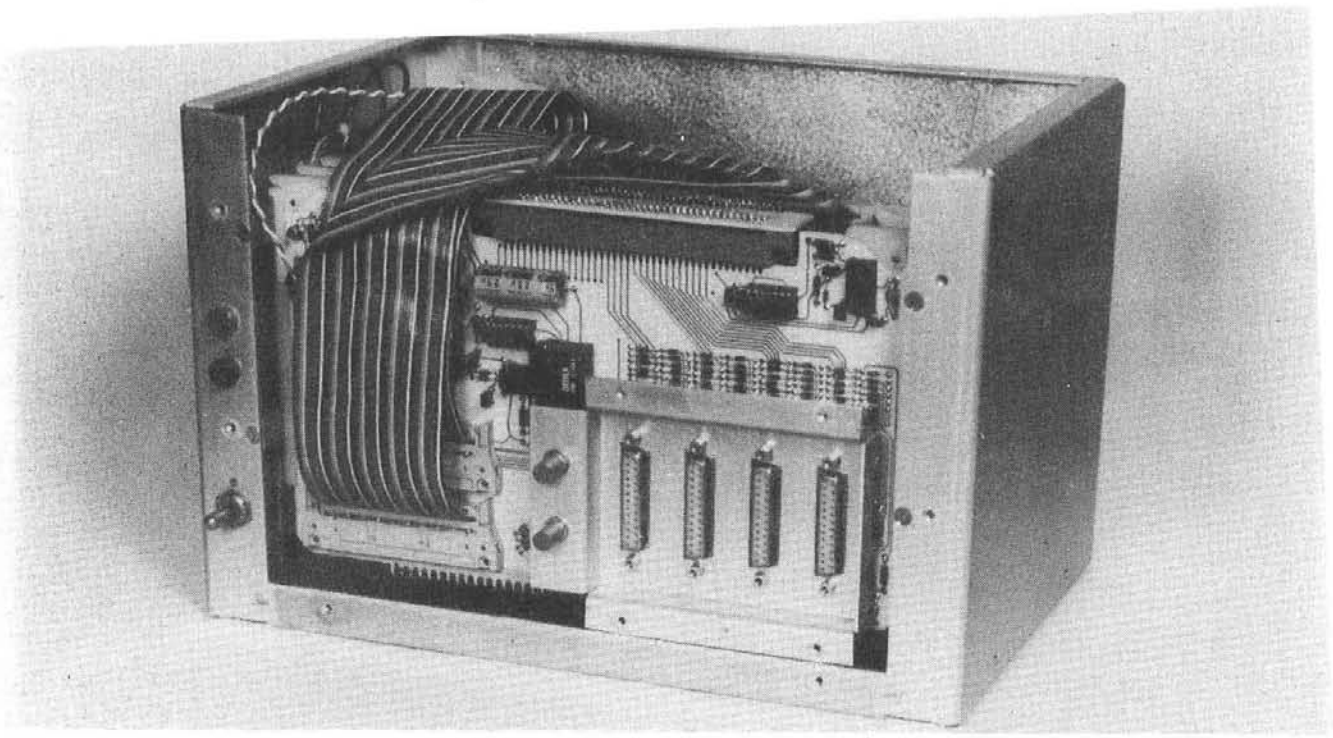
The Problem

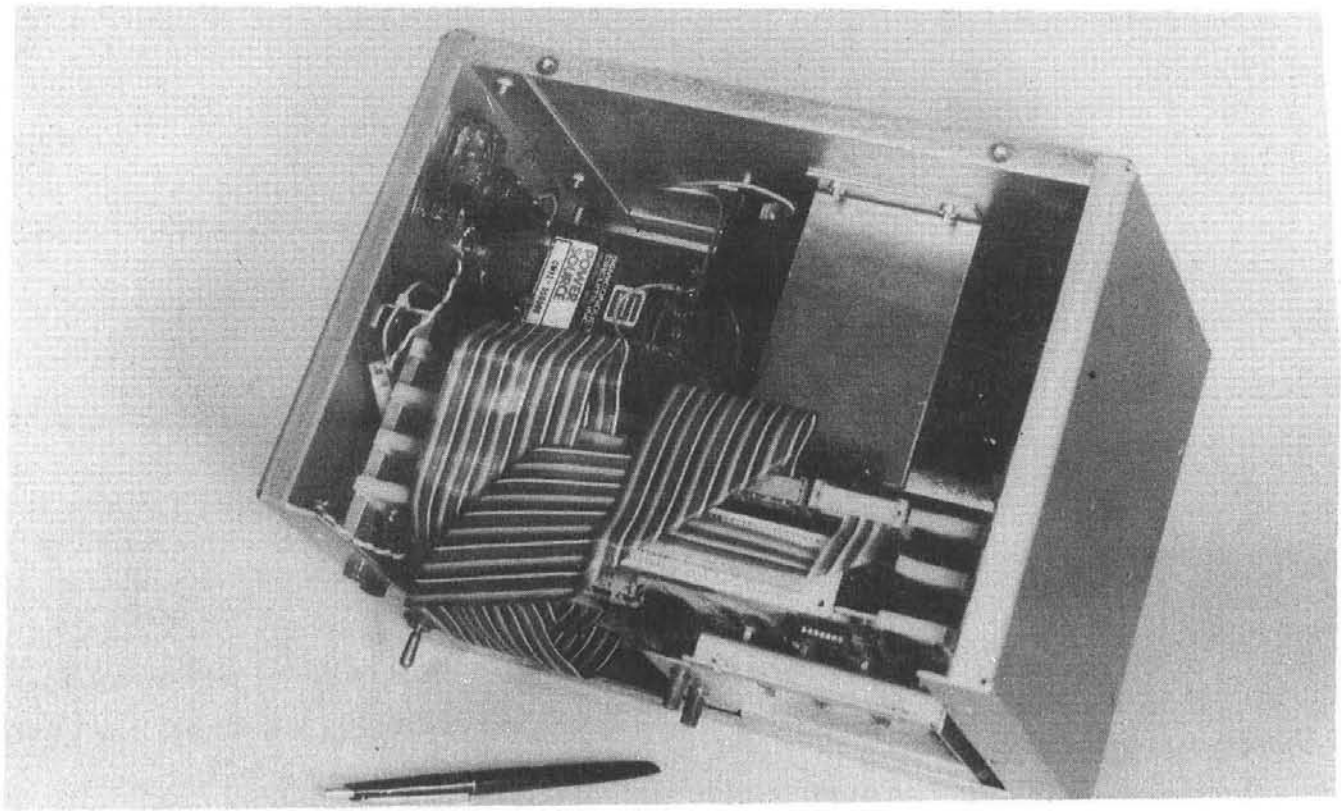
A facility is being built which generates artificial lightning for testing purposes. The facility (called "LIX") will include several Marx high voltage generators and other types of equipment which require monitoring and control for proper operation. Voltages from voltage dividers and current viewing resistors will be measured, various switches and safety interlocks will be monitored, and control signals for the Marx hardware will be generated.

All of these things could be done with a standard computer-controlled data acquisition system using conventional analog to digital converters and digital input/output hardware. This would mean, however, that all points to be monitored would be hard wired to the computer system. Because of the potentially lethal voltages present, this method is considered unsafe both for the operating personnel and the computer hardware. Fiber optic isolation seemed ideal. However, it could become both expensive and unwieldy if an individual fiber optic cable is used for each required input.

The Solution

A small microprocessor controlled data acquisition module was built that provides the proper isolation. This module functions as an interface between the high voltage experiments and a minicomputer system which provides data analysis and display. The module and minicomputer communicate through a full duplex fiber optic link using two fiber optic cables.





ISOLATED DATA ACQUISITION MODULE

All input multiplexing is done in the module. The minicomputer transmits ASCII commands to the module, and the module transmits binary data back to the minicomputer. This transmission is serial asynchronous, at 9600 baud.

The module contains: a sealed lead-acid battery to provide all power; several DC-DC converters to provide the needed voltages; special hardware to allow power-down on command; a microcomputer to control all operations; an analog to digital converter and multiplexer to allow voltage measurements; digital input ports; digital output ports; an internal timer which provides relative timing information; and protection circuitry for all inputs and outputs.

Performance

As configured any of fifteen differential analog inputs can be measured over a range of plus and minus 10 volts. A sixteenth channel is dedicated to monitoring the battery or to initiate power-down. Thus, sixteen inputs and fifteen outputs are available for TTL level signals. These TTL lines are available as four eight-bit bytes which may be read or written to on command. The internal timer is a sixteen bit binary counter which is incremented every 10 milliseconds (ms). This gives an overall range of a little more than 10 minutes. On command, this timer may be read, as two bytes, or cleared to zero.

Although the A-D conversion is 12 bits wide, a fast mode option exists that allows only the 8 most significant bits to

be transmitted to the minicomputer. This results in increased throughput. Because the A-D converter is relatively fast, overall data rates are restricted mainly by the serial transmission rate. It takes about 1 ms to transmit a command or other argument to the module, about 100 microseconds to interpret the command and get the data, and about 1 ms to transmit each byte of data back to the minicomputer. Non A-D operations are only slightly faster.

Software

Command Structure

Basically the program operates as a simple interpreter. Commands are sent to the module and certain responses are returned to the minicomputer. The nature of these responses is determined by the specific commands sent and the arguments (if any) following them. Each command is represented by an ASCII letter. The contiguous letters M through Z are used in order to simplify command interpretation. To understand how the module uses these commands, it is necessary to consider two RAM tables whose functions form the nucleus for the module's programs.

Each of the module's twenty inputs is assigned an ASCII identifier according to the following table:

Input	ASCII Identifier
A-D Channel 0	0
A-D Channel 1	1
A-D Channel 2	2
A-D Channel 3	3

Input	ASCII Identifier
A-D Channel 4	4
A-D Channel 5	5
A-D Channel 6	6
A-D Channel 7	7
A-D Channel 8	8
A-D Channel 9	9
A-D Channel 10	A
A-D Channel 11	B
A-D Channel 12	C
A-D Channel 13	D
A-D Channel 14	E
A-D Channel 15	F
Input Port A	G
Input Port C	H
Timer (LSB)	I
Timer (MSB)	J

A RAM table called "QLIST" exists where these identifiers can be stacked. QLIST can have from 0 to 22 identifiers in it. Identifiers may be stacked in any sequence, and the same identifier may be stacked more than once. This stacking operation will be done by commands from the minicomputer.

QLIST ultimately determines which inputs are to be evaluated. For example, if the minicomputer needed to measure A-D channel 0, A-D channel 1, input port C, and the high and low bytes of the timer, QLIST would look like this:

QLIST 0

1

H

J

I

Another RAM table called "QDATA" exists where data will be stored. On command the program will interpret the identifiers contained in QLIST one at a time and in order. As each identifier is interpreted the corresponding input is evaluated. The results of that evaluation are stored in QDATA. On command QDATA will be transmitted to the minicomputer over the fiber optic link. QDATA can be from 0 to 44 bytes long.

Each A-D channel identifier would normally require two bytes in QDATA after conversion. The conversion yields a twelve bit twos compliment number which is right-hand justified (with sign extension) into a sixteen bit result. This is the "normal" or slow mode of operation and is the mode selected by the initialization portion of the program. Another mode called FAST is also available which left-hand justifies the result and stores only the eight most significant bits of the answer in QDATA. By using only one byte the throughput is nearly doubled. The non-AD identifiers require only one byte each in QDATA.

Each of the available commands is described below. The first six of these modify or otherwise use the tables QDATA and QLIST.

M The "M" command loads QLIST. An ASCII M is transmitted to the module followed by the desired identifiers.

These are loaded into QLIST until the list is full or until an illegal identifier is received.

W The "W" command also loads QLIST but not from received identifiers. This time QLIST is loaded from a standard list of identifiers stored in EPROM. This list is comprised of the consecutive identifiers 0 through J.

R The "R" command will cause the program to interpret QLIST once filling QDATA, and then transmit QDATA.

N The "N" command causes a continual interpretation of QLIST and updating of QDATA. This continues until any character (command or not) is received by the module The "N" command does not cause any data to be transmitted.

S The "S" command simply causes the transmission of QDATA. No new data is taken with this command.

T The "T" command causes the transmission of QLIST not QDATA. All of the identifiers in QLIST are transmitted back to the minicomputer as ASCII characters.

O The "O" command resets both the high and low bytes of the clock to zero.

P The "P" command writes to output port "B" and the command is followed by an 8 bit binary value which will be written to output port "B".

Q The "Q" command writes to output port "D" and the command is followed by an 8 bit binary value which will be written to output port "D".

U The "U" command causes the module's status bytes to be transmitted. There are four status bytes: The first

byte contains either an ASCII "S" or "F" for slow or fast mode. The second byte contains the last command received (never "U"). The third byte contains the number of bytes in QDATA. The fourth byte contains the number of identifiers in QLIST.

V The "V" command will toggle the module's mode either from fast to slow or slow to fast.

X The "X" command causes a hardware power-down to occur. This is accomplished by clearing bit 7 on output port B.

The letters "Y" and "Z" provide two more contiguous alphabetic commands that could be used but they are undefined at this time.

Considerable flexibility exists with this command structure. For example, in its simplest use a single "W" command will set up QLIST so that all available inputs will be read and the data sent to the minicomputer with each occurrence of subsequent "R" commands. In a more complex situation only desired inputs might be loaded into QLIST using the "M" command. Then the "N" command could be sent initiating a continuous sampling loop. At some future point in time an "S" command could be sent to stop the sampling and transmit the latest data. Or, any non-command character could have been sent to stop the sampling, and the "S" command used even later in time to transmit the data back to the minicomputer.

Program Flow

At power-on, the reset vector points to the initialization routine. There the stack is set up, the programmable hardware is initialized, and the mode is set to slow. At this point the

program goes into an "idle" loop waiting for a command to be received. If a received character is a valid command, a jump address is computed and the proper command routine is entered. It is ignored if it is not valid. If the command requires another argument, it will be interpreted from inside the command routine. After the command is finished the program returns to the idel loop. This is shown graphically in Figure 1.

The heart of the program is a routine called "FILQUE". This is the routine that interprets QLIST and fills QDATA. A flow chart for this routine is given in Figure 2. An assembly listing for the entire program is given in Appendix A.

Since the entire program is only 500 bytes long, it fits easily into one 2708 type EPROM. Although a full 1024 bytes of RAM are available, only 81 bytes are used.

Hardware

Three bus compatible circuit cards, two DC to DC converters and a latching relay make up the data acquisition module. The circuit cards used are the Motorola micromodule (M68MM01A), Datel analog to digital peripheral system card (ST-6800), and a special hardware card.

The M68MM01A is a complete microcomputer on a card. It is made up of a MC6800 microprocessor unit, 1 K of RAM, sockets for up to 4 K of EPROMS, two programmable peripheral interface adapters (FIAS) for parallel data transfers, and one asynchronous communications interface adapter (ACIA) for RS232C serial data transfers. The card also incorporates the necessary two phase

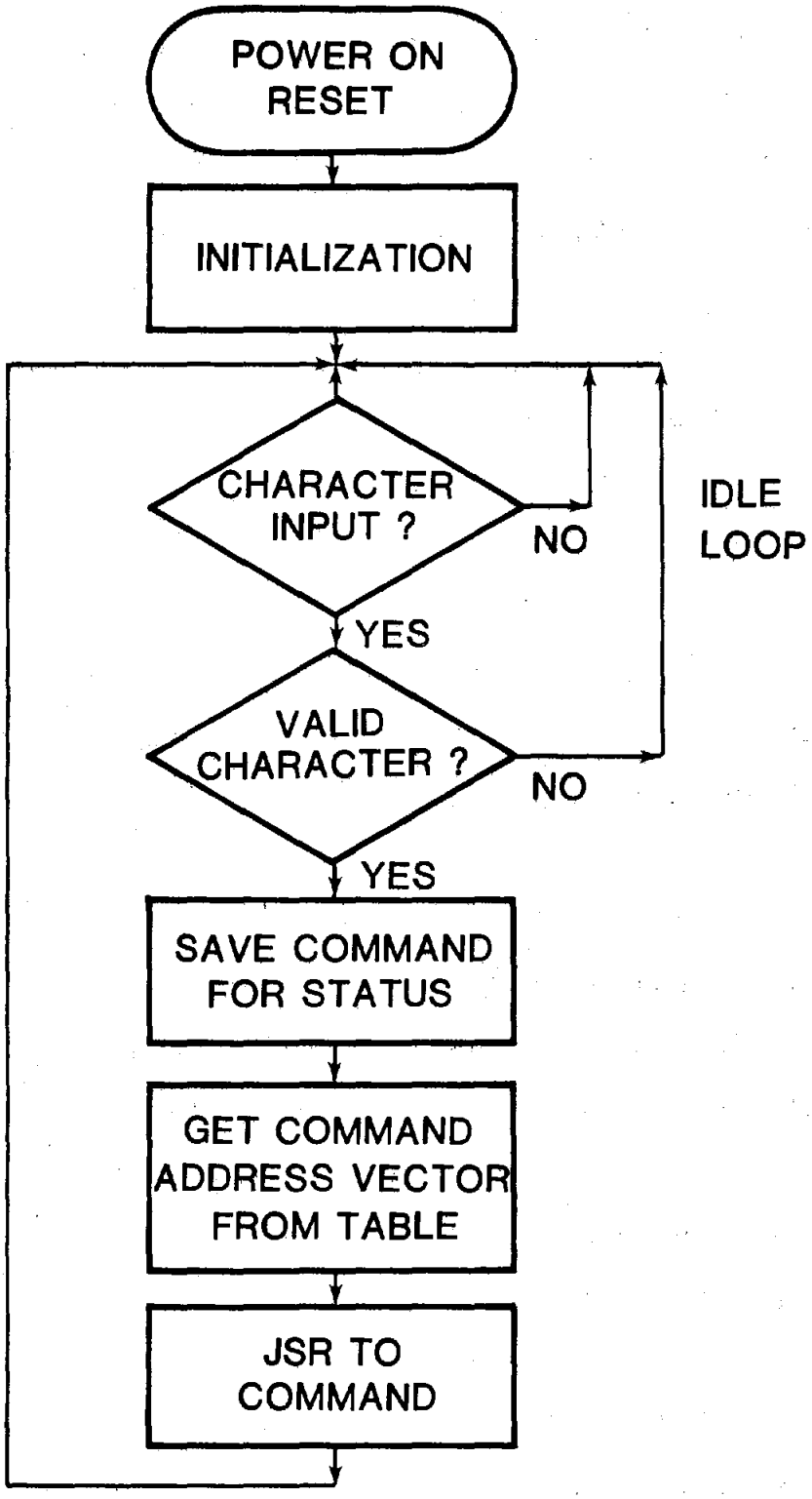

POWER ON
RESET
INITIALIZATION
CHARACTER
INPUT ?
NO
IDLE
LOOP
YES
VALID
CHARACTER ?
NO
YES
SAVE COMMAND
FOR STATUS
GET COMMAND
ADDRESS VECTOR
FROM TABLE
JSR TO
COMMAND

FIGURE 1

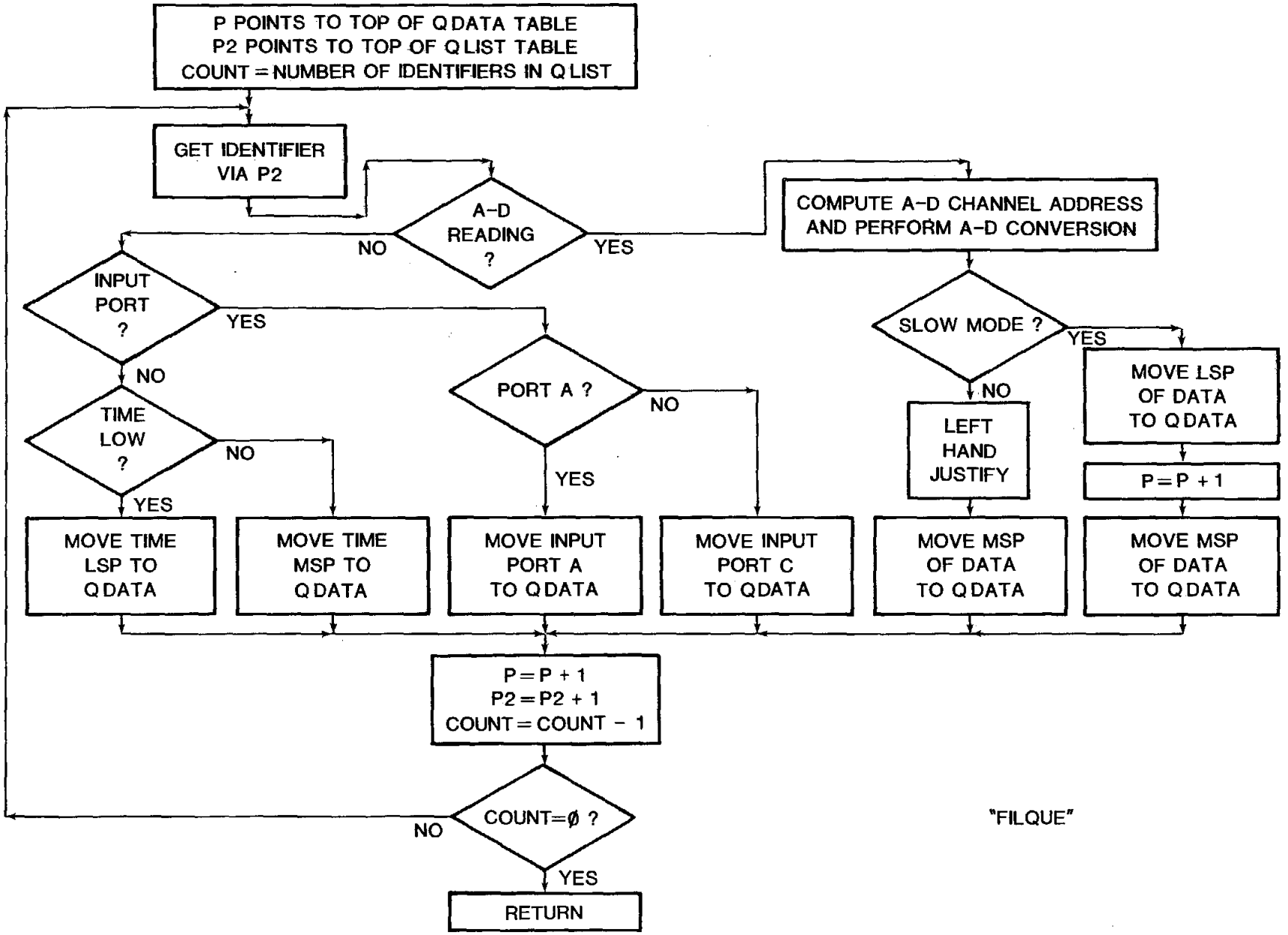

P POINTS TO TOP OF QDATA TABLE
P2 POINTS TO TOP OF QLIST TABLE
COUNT = NUMBER OF IDENTIFIERS IN QLIST
GET IDENTIFIER VIA P2
A-D READING ?
NO
YES
COMPUTE A-D CHANNEL ADDRESS AND PERFORM A-D CONVERSION
INPUT PORT ?
YES
NO
SLOW MODE ?
YES
NO
MOVE LSP OF DATA TO QDATA
P = P + 1
LEFT HAND JUSTIFY
PORT A ?
NO
YES
TIME LOW ?
NO
YES
MOVE TIME LSP TO QDATA
MOVE TIME MSP TO QDATA
MOVE INPUT PORT A TO QDATA
MOVE INPUT PORT C TO QDATA
MOVE MSP OF DATA TO QDATA
MOVE MSP OF DATA TO QDATA
P = P + 1
P2 = P2 + 1
COUNT = COUNT - 1
COUNT=Ø ?
NO
YES
RETURN
"FILQUE"

FIGURE 2

clock generator and the reset timer for "power on" initialization. The ACIA was jumpered for a 9600 baud rate while software sets the PIAS for sixteen input and sixteen output lines.

The ST-6800 is an analog to digital converter card which is bus compatible with the M68MM01A micromodule. The card can be strapped for a variety of input voltages and output codes. For this application it was strapped for 16 differential inputs, a range of plus and minus 10 volts, and a 2's complement output.

The special card contains a third DC to DC converter, fiber optic circuitry, power control circuit, input and output protection circuitry, and an internal timer.

The DC to DC converter on this card has plus 5 volt and minus 9 volt outputs and provides power to the fiber optic receiver and power control circuit. With the module's master power switch "ON," these circuits are powered and remain so regardless of whether the module is powered up or powered down.

Schematics for the special card are in Appendix B. Also included is a master interconnection diagram. These three sheets are designated as CK-T70185 and contain the following:

Sheet 1 of 3	I/O connectors and protection circuitry
Sheet 2 or 3	DC-DC converter #3, power control logic, detector preamp, and internal clock
Sheet 3 of 3	Main DC-DC converters and master interconnection

Also available, but not included, is a layout drawing for the special card which is designated T70185. The descriptions which follow will reference these drawings.

The power control circuit, shown on sheet 2 of 3, monitors the output of the fiber optic receiver for a power up command. The minicomputer must set the fiber optic cable to its marking state (light on) before the power control circuit will recognize the power up command. This command consists of any ASCII character sent down the fiber optic cable to the module. Receiving this character causes the power control circuit to toggle the latching relay which applies power to the rest of the module. This relay connects the twelve volt battery to a DC to DC converter with an output of five volts. This 5 volts is connected to the module bus and also to another DC to DC converter. The output of this second converter is plus and minus twelve volts which is also applied to the bus. The power applied to the M68MM01A causes it to enter the initialization routine.

The power control circuit is also used to enter the power down state. This can be done either of two ways. One way is for the microprocessor to give the circuit a high to low transition on one dedicated PIA output line. Another way is for the fiber optic cable to be turned off at the minicomputer. After about ten milliseconds this light off condition causes the module to enter the power down state.

The plus 5 volt and minus 9 volt DC to DC converter on the special card is made using a universal switching regulator subsystem from Fairchild (UA78S40).

The switching regulator is set up as an inverting regulator to produce -9 volts from 12 volts. An internal operational amplifier on the UA78S40 is used as a linear regulator to produce +5 volts.

Also contained on the special card is a RS232C to TTL receiver (1489) and a TTL to RS232C driver (1488). These are used to interface between the ACIA port on the M68MM01A card and the fiber optic emitter driver and detector preamplifier circuitry. Two sets of Augat emitter drivers and fiber optic cables along with two high gain low bandwidth detector preamplifiers are used for the optical link with the minicomputer.

When the M68MM01A talks to the minicomputer, its output via the ACIA is connected to a 1489. The output of the 1489 is connected to an emitter driver to send the signal over the fiber optic cable. At the minicomputer the output of the fiber is detected, preamplified, and then applied to a 1488 to convert back to RS232C levels for the minicomputer interface.

When the minicomputer wants to talk to the module, its output from the RS232C interface goes to a 1489, which applies its output to an emitter driver module to send the signal over the optical cable. The output of the cable is detected by the detector preamplifier, and applied to two Schmitt gates for buffering. The output of the Schmitt gates goes to the power control circuitry and a 1488 to convert to RS232C levels for the ACIA.

The preamplifier circuit shown on sheet 2 of 3 is used at both ends of the fiber optic link. With only dark current

flowing in the detector, the output is approximately 0 volts. When light is on the detector, the current through it increases causing the output of the OP-AMP to go to approximately 3.3 volts. The two diodes and the 1 K ohm pull up resistor are used to allow the final output to reach about 4.5 volts. This level is desired for a logic "high" input to the Schmitt gate.

The protection circuitry for the digital lines shown on sheet 1 of 3 consists of a 5.1 volt zener (1N751A) across each of the digital input and output lines to help protect them from high voltage transients.

The protection for the analog input lines also shown on sheet 1 consists of 10 K ohm quarter watt resistors in series with the analog inputs and two diodes (1N914) strapping the output of the resistors to the positive and negative 12 volt supplies. The protection circuitry will not permit the analog inputs, to the ST-6800 card, to go outside the voltage limits set by the ±12 volt supplies.

The interval clock consists of two counters (4518) used as dividers. The dividers are used to divide the one megahertz phase 2 clock of the M68MM01A down to one hundred hertz. This provides an interrupt through the CA1 line of a PIA causing the microprocessor to update its time count every 10 milliseconds.

APPENDIX A

Assembly Listing

```
00001                    NAM    LIX
00002                    *
00003                    *
00004                    *
00005                    *
00006                    *
00007                    * FILE LIX0
00008                    *
00009                    * THIS PROGRAM CONTROLS THE CONVERSION, FORMATTING,
00010                    * AND TRANSMISSION OF DATA GATHERED IN EXPERIMENTS
00011                    * AT THE LIX FACILITY.  OPERATING AS AN INTERPRETER,
00012                    * THE PROGRAM RECEIVES CONTROL COMMANDS VIA A FIBER
00013                    * OPTIC LINK, THEN TRANSMITTS FORMATTED DATA TO THE
00014                    * MASTER COMPUTER VIA ANOTHER FIBER OPTIC LINK.
00015                    *
00016                    *
00017                    * FOLLOWING IS A LIST OF COMMAND DESCRIPTIONS
00018                    * AND THE ASCII CHARACTERS THAT INVOKE THEM.
00019                    *
00020                    * CHAR.          DESCRIPTION
00021                    *
00022                    *  M  LOAD IDENTIFIERS INTO THE QUEUE LIST.
00023                    *
00024                    *  N  CONTINUOUSLY INTERPRET THE QUEUE LIST
00025                    *     AND UPDATE THE QUEUE DATA ARRAY. THIS
00026                    *     CONTINUES UNTIL ANOTHER CHARACTER IS
00027                    *     RECEIVED.
00028                    *
00029                    *  O  RESET BOTH THE HIGH AND LOW BYTES
00030                    *     OF THE CLOCK TO ZERO.
00031                    *
00032                    *  P  STORE ARGUMENT IN OUTPUT PORT "B".
00033                    *
00034                    *  Q  STORE ARGUMENT IN OUTPUT PORT "D".
00035                    *
00036                    *  R  INTERPRET THE QUEUE LIST ONCE, UPDATE
00037                    *     THE QUEUE DATA, AND THEN TRANSMIT THE
00038                    *     QUEUE DATA.
00039                    *
00040                    *  S  TRANSMIT THE QUEUE DATA.
00041                    *
00042                    *  T  TRANSMIT THE QUEUE LIST.
00043                    *
00044                    *  U  TRANSMIT SYSTEM STATUS.
00045                    *
00046                    *  V  TOGGLE MODE, EITHER HIGH TO LOW OR LOW
00047                    *     TO HIGH.
00048                    *
00049                    *  W  LOAD THE QUEUE LIST FROM A STANDARD LIST
00050                    *     WHICH CONTAINS 0 THRU J INCLUSIVE.
00051                    *
00052                    *  X  SHUTDOWN POWER. UNIT NOW REQUIRES A
00053                    *     HARDWARE RESTART.
00054                    *
```

```
00056                   *   QUEUE IDENTIFIERS
00057                   *
00058                   *   CHAR.     DESCRIPTION
00059                   *
00060                   *     0       A-D CHANNEL 0
00061                   *
00062                   *     1       A-D CHANNEL 1
00063                   *
00064                   *     2       A-D CHANNEL 2
00065                   *
00066                   *     3       A-D CHANNEL 3
00067                   *
00068                   *     4       A-D CHANNEL 4
00069                   *
00070                   *     5       A-D CHANNEL 5
00071                   *
00072                   *     6       A-D CHANNEL 6
00073                   *
00074                   *     7       A-D CHANNEL 7
00075                   *
00076                   *     8       A-D CHANNEL 8
00077                   *
00078                   *     9       A-D CHANNEL 9
00079                   *
00080                   *     A       A-D CHANNEL 10
00081                   *
00082                   *     B       A-D CHANNEL 11
00083                   *
00084                   *     C       A-D CHANNEL 12
00085                   *
00086                   *     D       A-D CHANNEL 13
00087                   *
00088                   *     E       A-D CHANNEL 14
00089                   *
00090                   *     F       A-D CHANNEL 15
00091                   *
00092                   *     G       INPUT PORT A
00093                   *
00094                   *     H       INPUT PORT C
00095                   *
00096                   *     I       TIMER (LSB)
00097                   *
00098                   *     J       TIMER (MSB)
00099                   *
00100                   *
00101                   *
```

```
00103A 0000                        ORG     $0000
00104A 0000  002C  A QDATA   RMB   44      QUE DATA TABLE
00105A 002C  0016  A QLIST   RMB   22      QUE IDENTIFIER TABLE
00106        0016  A QFULL   EQU   22      MAXIMUM # OF IDENTIFIERS
00107A 0042  0001  A STATUS  RMB   01      STATUS
00108A 0043  0001  A LASCOM  RMB   01      LAST COMMAND
00109A 0044  0001  A QDNUM   RMB   01      NUMBER OF DATA
00110A 0045  0001  A QCOUNT  RMB   01      CURRENT # OF IDENTIFIERS
00111A 0046  0002  A TIMEHI  RMB   02      MSP OF TIMER
00112A 0048  0002  A TIMELO  RMB   02      LSP OF TIMER
00113A 004A  0002  A QDX     RMB   02      QUE DATA POINTER
00114A 004C  0002  A QLX     RMB   02      QUE LIST POINTER
00115A 004E  0002  A P0      RMB   02      SCRATCH POINTER
00116A 0050  0001  A P1      RMB   01      SCRATCH POINTER
00117A 0051  0001  A MASK    RMB   01      BIT MASK
00118                *
00119                *
00120        8400  A PORTA   EQU   $8400
00121        8401  A PORTAC  EQU   $8401
00122        8402  A PORTB   EQU   $8402
00123        8403  A PORTBC  EQU   $8403
00124        8404  A PORTC   EQU   $8404
00125        8405  A PORTCC  EQU   $8405
00126        8406  A PORTD   EQU   $8406
00127        8407  A PORTDC  EQU   $8407
00128        8408  A ACIAC   EQU   $8408
00129        8409  A ACIA    EQU   $8409
00130        E000  A ADBASE  EQU   $E000
```

```
00132                                  *
00133                                  *
00134                                  *
00135                                  *
00136                                  *
00137                                  * FILE LIX1
00138                                  *
00139A FC00                            ORG     $FC00
00140A FC00     FCA2   A COMTAB FDB     LQLIST  COMMAND M
00141A FC02     FCD1   A        FDB     CONFIL  N
00142A FC04     FCDC   A        FDB     CLRCLK  O
00143A FC06     FCE3   A        FDB     SPORTB  P
00144A FC08     FCEE   A        FDB     SPORTD  Q
00145A FC0A     FCF8   A        FDB     ONEQUE  R
00146A FC0C     FD04   A        FDB     QUEOUT  S
00147A FC0E     FD0D   A        FDB     OUTLST  T
00148A FC10     FD16   A        FDB     STAOUT  U
00149A FC12     FD1F   A        FDB     TOGMOD  V
00150A FC14     FD2F   A        FDB     STANDQ  W
00151A FC16     FD63   A        FDB     PWRDWN  X
00152A FC18     FD67   A        FDB     UNDEF   Y UNDEFINED
00153A FC1A     FD67   A        FDB     UNDEF   Z UNDEFINED
00154A FC1C 0F          INIT    SEI             SET INTERRUPT MASK
00155A FC1D CE 03FF    A        LDX     #$03FF  TOP OF RAM
00156A FC20 35                  TXS             INIT. STACK
00157                                  *
00158                                  * THE FOLLOWING CHANGES WERE MADE
00159                                  * IN SEPTEMBER OF 1980.  THESE CHANGES
00160                                  * ALLOW THE PIA DATA REGISTERS TO BE
00161                                  * PRE-LOADED WITH ALL ONES.  THIS MEANS
00162                                  * THAT WHEN THE PORTS ARE ESTABLISHED
00163                                  * AS OUTPUTS, THE CORRESPONDING DATA
00164                                  * LINES WILL "COME UP" TO A HIGH LEVEL.
00165                                  *
00166                                  * THIS IS PARTICULARLY ESSENTIAL FOR
00167                                  * BIT 7 ON PORT B AS THIS BIT CONTROLS
00168                                  * POWERDOWN.
00169                                  *
00170A FC21 86 2C       A               LDAA    #$2C    TO GET TO DATA REG.
00171A FC23 B7 8403     A               STAA    PORTBC  PORT B CONTROL
00172A FC26 B7 8407     A               STAA    PORTDC  PORT D CONTROL
00173A FC29 86 FF       A               LDAA    #$FF
00174A FC2B B7 8402     A               STAA    PORTB   SET ALL BITS "ON"
00175A FC2E B7 8406     A               STAA    PORTD   SET ALL BITS "ON"
00176A FC31 4F                          CLRA            DDR ACCESS
00177A FC32 B7 8401     A               STAA    PORTAC  PORT A CONTROL
00178A FC35 B7 8403     A               STAA    PORTBC  PORT B CONTROL
00179A FC38 B7 8405     A               STAA    PORTCC  PORT C CONTROL
00180A FC3B B7 8407     A               STAA    PORTDC  PORT D CONTROL
00181A FC3E B7 8400     A               STAA    PORTA   SET FOR INPUT
00182A FC41 B7 8404     A               STAA    PORTC   SET FOR INPUT
00183A FC44 86 FF       A               LDAA    #$FF
00184A FC46 B7 8402     A               STAA    PORTB   SET FOR OUTPUT
00185A FC49 B7 8406     A               STAA    PORTD   SET FOR OUTPUT
00186A FC4C 86 2D       A               LDAA    #$2D    IRQ ON, STROBES ON R/W
00187A FC4E B7 8401     A               STAA    PORTAC
00188A FC51 B7 8403     A               STAA    PORTBC
00189A FC54 B7 8405     A               STAA    PORTCC
```

```
00190A FC57 B7 8407   A        STAA   PORTDC
00191A FC5A 86 03     A        LDAA   #$03      MASTER RESET
00192A FC5C B7 8408   A        STAA   ACIAC
00193A FC5F 86 19     A        LDAA   #$19      8 BITS, EVEN P, ONE STOP, CLOCK/16
00194A FC61 B7 8408   A        STAA   ACIAC
00195A FC64 86 53     A        LDAA   #'S       SLOW MODE
00196A FC66 97 42     A        STAA   STATUS    MODE
00197A FC68 4F                 CLRA
00198A FC69 97 43     A        STAA   LASCOM
00199A FC6B 97 44     A        STAA   QDNUM
00200A FC6D 97 45     A        STAA   QCOUNT
00201A FC6F 0E                 CLI              ENABLE INTERRUPTS
```

```
00203                          *
00204                          *
00205                          *
00206                          *
00207                          *
00208                          * FILE LIX2
00209                          *
00210A FC70 B6 8408   A IDLE   LDAA   ACIAC     GET STATUS
00211A FC73 85 01     A        BITA   #$01      CHECK FOR READY
00212A FC75 27 F9 FC70         BEQ    IDLE      BRANCH IF NOT READY
00213A FC77 B6 8409   A        LDAA   ACIA      GET CHARACTER
00214A FC7A 81 5A     A        CMPA   #'Z       GREATER THAN "Z"
00215A FC7C 22 F2 FC70         BHI    IDLE
00216A FC7E 81 4D     A        CMPA   #'M       LESS THAN "M"
00217A FC80 25 EE FC70         BCS    IDLE
00218A FC82 81 55     A        CMPA   #'U       "STAOUT" COMMAND??
00219A FC84 27 02 FC88         BEQ    ID2       IF SO, BRANCH
00220A FC86 97 43     A        STAA   LASCOM    PREVIOUS COMMAND
00221A FC88 80 4D     A ID2    SUBA   #'M       SUBTRACT OFFSET
00222A FC8A 48                 ASLA             X2
00223A FC8B CE FC00   A        LDX    #COMTAB   START OF COMMAND TABLE
00224A FC8E DF 4E     A        STX    P0        SCRATCH POINTER
00225A FC90 9B 4F     A        ADDA   P0+1      LSB OF POINTER
00226A FC92 97 4F     A        STAA   P0+1
00227A FC94 86 00     A        LDAA   #$00
00228A FC96 99 4E     A        ADCA   P0        MSB OF POINTER
00229A FC98 97 4E     A        STAA   P0
00230A FC9A DE 4E     A        LDX    P0        ADDRESS OF COMMAND VECTOR
00231A FC9C EE 00     A        LDX    0,X       GET VECTOR
00232A FC9E AD 00     A        JSR    0,X       EXECUTE COMMAND
00233A FCA0 20 CE FC70         BRA    IDLE      LOOK FOR NEXT COMMAND
```

```
00235                          *
00236                          *
00237                          *
00238                          *
00239                          *
00240                          * FILE LIX3
00241                          * * * * * * * * * * * * * * * * *
00242                          *
00243                          * LQLIST
00244                          *
00245A FCA2 CE 0000   A LQLIST LDX    #$0000
00246A FCA5 7F 0045   A        CLR    QCOUNT
00247A FCA8 B6 8408   A LQ2    LDAA   ACIAC     KEY IN??
00248A FCAB 85 01     A        BITA   #$01
00249A FCAD 27 F9 FCA8         BEQ    LQ2       WAIT FOR CHARACTER
00250A FCAF B6 8409   A LQ4    LDAA   ACIA      GET CHARACTER
00251A FCB2 81 4C     A        CMPA   #'L       GREATER THAN "L"??
00252A FCB4 22 1A FCD0         BHI    LQ8
00253A FCB6 81 30     A        CMPA   #'0       LESS THAN "0"??
00254A FCB8 25 16 FCD0         BCS    LQ8
00255A FCBA 81 39     A        CMPA   #'9       LESS THAN "9"??
00256A FCBC 23 06 FCC4         BLS    LQ6
00257A FCBE 81 41     A        CMPA   #'A       GREATER THAN "A"??
00258A FCC0 24 02 FCC4         BCC    LQ6
00259A FCC2 20 0C FCD0         BRA    LQ8       INVALID, BRANCH
00260A FCC4 A7 2C     A LQ6    STAA   QLIST,X   STACK IDENTIFIER
00261A FCC6 08                 INX
00262A FCC7 7C 0045   A        INC    QCOUNT
00263A FCCA 96 45     A        LDAA   QCOUNT
00264A FCCC 81 16     A        CMPA   #QFULL    LIST FULL??
00265A FCCE 26 D8 FCA8         BNE    LQ2       NOT FULL, BRANCH
00266A FCD0 39           LQ8    RTS              DONE, RETURN
00267                          *
00268                          * CONFIL
00269                          *
00270A FCD1 BD FD8B   A CONFIL JSR    FILQUE    FILL QUE DATA TABLE
00271A FCD4 B6 8408   A        LDAA   ACIAC     CHARACTER IN??
00272A FCD7 85 01     A        BITA   #$01
00273A FCD9 27 F6 FCD1         BEQ    CONFIL    NO CHARACTER, DO AGAIN
00274A FCDB 39                 RTS              DONE, RETURN
00275                          *
00276                          * CLRCLK
00277                          *
00278A FCDC 7F 0048   A CLRCLK CLR    TIMELO
00279A FCDF 7F 0046   A        CLR    TIMEHI
00280A FCE2 39                 RTS              DONE,RETURN
00281                          *
00282                          * SPORTB
00283                          *
00284A FCE3 CE 8402   A SPORTB LDX    #PORTB
00285A FCE6 86 80     A        LDAA   #$80      BIT 7 "ON"
00286A FCE8 97 51     A        STAA   MASK
00287A FCEA BD FD7C   A        JSR    SETPOR
00288A FCED 39                 RTS              DONE, RETURN
00289                          *
00290                          * SPORTD
00291                          *
00292A FCEE CE 8406   A SPORTD LDX    #PORTD
```

```
00293A FCF1 7F 0051  A          CLR     MASK       ALLOW ALL BITS
00294A FCF4 BD FD7C  A          JSR     SETPOR
00295A FCF7 39                  RTS                DONE,RTEURN
00296                    *
00297                    * ONEQUE
00298                    *
00299A FCF8 BD FD8B  A ONEQUE   JSR     FILQUE     FILL QUE DATA TABLE
00300A FCFB CE 0000  A          LDX     #QDATA     SET POINTER
00301A FCFE D6 44    A          LDAB    QDNUM      SET COUNTER
00302A FD00 BD FD68  A          JSR     OUTPUT     OUTPUT QUE DATA
00303A FD03 39                  RTS                DONE, RETURN
00304                    *
00305                    * QUEOUT
00306                    *
00307A FD04 CE 0000  A QUEOUT   LDX     #QDATA     SET POINTER
00308A FD07 D6 44    A          LDAB    QDNUM      SET COUNTER
00309A FD09 BD FD68  A          JSR     OUTPUT     OUTPUT QUE DATA
00310A FD0C 39                  RTS                DONE, RETURN
00311                    *
00312                    * OUTLST
00313                    *
00314A FD0D CE 002C  A OUTLST   LDX     #QLIST     SET POINTER
00315A FD10 D6 45    A          LDAB    QCOUNT     SET COUNTER
00316A FD12 BD FD68  A          JSR     OUTPUT     OUTPUT QUE LIST
00317A FD15 39                  RTS                DONE, RETURN
00318                    *
00319                    * STAOUT
00320                    *
00321A FD16 CE 0042  A STAOUT   LDX     #STATUS    SET POINTER
00322A FD19 C6 04    A          LDAB    #$04       SET COUNTER
00323A FD1B BD FD68  A          JSR     OUTPUT     OUTPUT STATUS
00324A FD1E 39                  RTS                DONE, RETURN
00325                    *
00326                    * TOGMOD
00327                    *
00328A FD1F 96 42    A TOGMOD   LDAA    STATUS     GET MODE
00329A FD21 81 53    A          CMPA    #'S        SLOW MODE??
00330A FD23 27 05 FD2A          BEQ     TO2        IF SO, BRANCH
00331A FD25 86 53    A          LDAA    #'S
00332A FD27 97 42    A          STAA    STATUS     SET SLOW MODE
00333A FD29 39                  RTS                DONE, RETURN
00334A FD2A 86 46    A TO2      LDAA    #'F
00335A FD2C 97 42    A          STAA    STATUS     SET FAST MODE
00336A FD2E 39                  RTS                DONE, RETURN
00337                    *
00338                    * STANDQ
00339                    *
00340A FD2F C6 14    A STANDQ   LDAB    #20        # OF ELEMENTS IN STANDARD LIST
00341A FD31 D7 45    A          STAB    QCOUNT
00342A FD33 CE 002C  A          LDX     #QLIST
00343A FD36 DF 4E    A          STX     P0
00344A FD38 CE FD4F  A          LDX     #STDLST
00345A FD3B DF 50    A          STX     P1
00346A FD3D DE 50    A ST2      LDX     P1
00347A FD3F A6 00    A          LDAA    0,X        FROM STANDARD LIST
00348A FD41 08                  INX
00349A FD42 DF 50    A          STX     P1
00350A FD44 DE 4E    A          LDX     P0
```

```
00351A FD46 A7 00     A          STAA    0,X        TO QUE LIST
00352A FD48 08                   INX
00353A FD49 DF 4E     A          STX     P0
00354A FD4B 5A                   DECB               DECREMENT COUNT
00355A FD4C 26 EF FD3D           BNE     ST2        IF NOT DONE, BRANCH
00356A FD4E 39                   RTS                DONE, RETURN
00357A FD4F    30     A STDLST FCC     /0123456789ABCDEFGHIJ/
00358                    *
00359                    * PWRDWN
00360                    *
00361A FD63 7F 8402   A PWRDWN CLR     PORTB
00362                    * THE FOLLOWING INSTRUCTION AND ANY THAT
00363                    * LOGICALLY FOLLOW MAY NOT BE EXECUTED
00364                    * DUE TO THE POWER DOWN COMMAND
00365A FD66 39                   RTS
00366                    *
00367                    * UNDEF
00368                    *
00369A FD67 39           UNDEF   RTS
```

```
00371                          *
00372                          *
00373                          *
00374                          *
00375                          *
00376                          * FILE LIX4
00377                          * * * * * * * * * * * * * * * * *
00378                          *
00379                          * OUTPUT
00380                          *
00381A FD6B 5D          OUTPUT  TSTB            COUNT
00382A FD69 27 10 FD7B         BEQ     OUT4    IF EMPTY, BRANCH
00383A FD6B B6 8408  A OUT2    LDAA    ACIAC   TRANSMITTER READY??
00384A FD6E 85 02     A        BITA    #$02
00385A FD70 27 F9 FD6B         BEQ     OUT2    BRANCH IF NOT READY
00386A FD72 A6 00     A        LDAA    0,X     GET OPERAND
00387A FD74 B7 8409  A         STAA    ACIA    OUTPUT
00388A FD77 08                 INX
00389A FD78 5A                 DECB
00390A FD79 26 F0 FD6B         BNE     OUT2    BRANCH IF NOT DONE
00391A FD7B 39          OUT4   RTS             DONE,RETURN
00392                          *
00393                          * SETPOR
00394                          *
00395A FD7C B6 8408  A SETPOR  LDAA    ACIAC   CHAR. IN?
00396A FD7F 85 01     A        BITA    #$01
00397A FD81 27 F9 FD7C         BEQ     SETPOR  WAIT FOR CHARACTER
00398A FD83 B6 8409  A         LDAA    ACIA    GET CHARACTER
00399A FD86 9A 51     A        ORAA    MASK
00400A FD88 A7 00     A        STAA    0,X     SET OUTPUT PORT
00401A FD8A 39          SP8    RTS             DONE, RETURN
00402                          *
00403                          * FILQUE
00404                          *
00405A FD8B 7F 0044  A FILQUE  CLR     QDNUM   # OF DATA
00406A FD8E D6 45     A        LDAB    QCOUNT  SET COUNTER
00407A FD90 CE 0000  A         LDX     #QDATA  GET DATA POINTER
00408A FD93 DF 4A     A        STX     QDX     STORE
00409A FD95 CE 002C  A         LDX     #QLIST  GET LIST POINTER
00410A FD98 DF 4C     A        STX     QLX     STORE
00411A FD9A A6 00     A FQ2    LDAA    0,X     GET ARG. FROM LIST
00412A FD9C 81 46     A        CMPA    #'F
00413A FD9E 23 1E FDBE         BLS     FQA     IF A-D CHANNEL, BRANCH
00414A FDA0 81 48     A        CMPA    #'H
00415A FDA2 23 0C FDB0         BLS     FQ6     IF INPUT PORT, BRANCH
00416A FDA4 81 49     A        CMPA    #'I
00417A FDA6 27 04 FDAC         BEQ     FQ4
00418A FDA8 96 46     A        LDAA    TIMEHI  "J"
00419A FDAA 20 56 FE02         BRA     FQE
00420A FDAC 96 48     A FQ4    LDAA    TIMELO  "I"
00421A FDAE 20 52 FE02         BRA     FQE
00422A FDB0 81 47     A FQ6    CMPA    #'G
00423A FDB2 27 05 FDB9         BEQ     FQ8
00424A FDB4 B6 8404  A         LDAA    PORTC   "H"
00425A FDB7 20 49 FE02         BRA     FQE
00426A FDB9 B6 8400  A FQ8     LDAA    PORTA   "G"
00427A FDBC 20 44 FE02         BRA     FQE
00428A FDBE 81 39     A FQA    CMPA    #'9
```

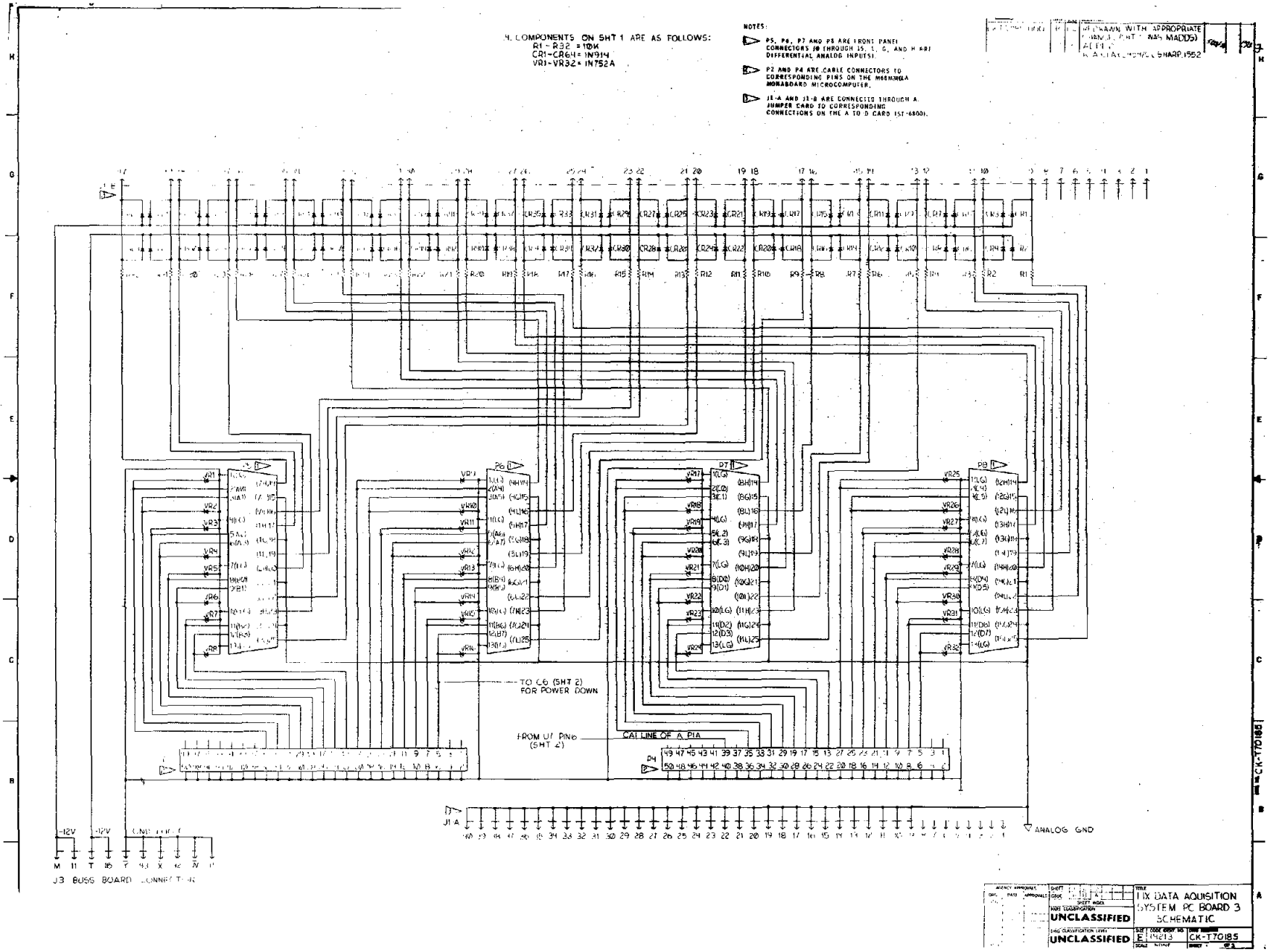
```
00429A FDC0 23 02 FDC4          BLS     FQC         IF DIGIT, BRANCH
00430A FDC2 80 07    A          SUBA    #$07
00431A FDC4 80 30    A FQC      SUBA    #$30
00432A FDC6 48                  ASLA                TIMES TWO
00433A FDC7 CE E000  A          LDX     #ADBASE
00434A FDCA DF 4E    A          STX     P0
00435A FDCC 9B 4F    A          ADDA    P0+1        LSB
00436A FDCE 97 4F    A          STAA    P0+1
00437A FDD0 86 00    A          LDAA    #$00
00438A FDD2 99 4E    A          ADCA    P0          MSB
00439A FDD4 97 4E    A          STAA    P0
00440A FDD6 DE 4E    A          LDX     P0
00441A FDD8 A7 00    A          STAA    0,X         STROBE A-D CONVERTER
00442A FDDA 01                  NOP
00443A FDDB 96 42    A          LDAA    STATUS
00444A FDDD 81 53    A          CMPA    #'S
00445A FDDF 27 10 FDF1          BEQ     FQD         IF SLOW MODE, BRANCH
00446A FDE1 A6 00    A          LDAA    0,X         GET MSB DATA
00447A FDE3 37                  PSHB                SAVE COUNT
00448A FDE4 E6 01    A          LDAB    1,X         GET LSB DATA
00449A FDE6 58                  ASLB
00450A FDE7 49                  ROLA                LEFT HAND JUSTIFY
00451A FDE8 58                  ASLB
00452A FDE9 49                  ROLA
00453A FDEA 58                  ASLB
00454A FDEB 49                  ROLA
00455A FDEC 58                  ASLB
00456A FDED 49                  ROLA
00457A FDEE 33                  PULB                RESTORE COUNT
00458A FDEF 20 11 FE02          BRA     FQE
00459A FDF1 A6 00    A FQD      LDAA    0,X         GET MSB DATA
00460A FDF3 37                  PSHB                SAVE COUNT
00461A FDF4 E6 01    A          LDAB    1,X         GET LSB DATA
00462A FDF6 DE 4A    A          LDX     QDX
00463A FDF8 A7 00    A          STAA    0,X
00464A FDFA 08                  INX
00465A FDFB DF 4A    A          STX     QDX
00466A FDFD 7C 0044  A          INC     QDNUM
00467A FE00 17                  TBA                 SETUP LSB
00468A FE01 33                  PULB                RESTORE COUNT
00469A FE02 DE 4A    A FQE      LDX     QDX
00470A FE04 A7 00    A          STAA    0,X
00471A FE06 08                  INX
00472A FE07 DF 4A    A          STX     QDX
00473A FE09 7C 0044  A          INC     QDNUM       NUMBER OF DATA
00474A FE0C DE 4C    A          LDX     QLX
00475A FE0E 08                  INX
00476A FE0F DF 4C    A          STX     QLX
00477A FE11 5A                  DECB                DECREMENT COUNT
00478A FE12 26 86 FD9A          BNE     FQ2         IF NOT DONE, DO AGAIN
00479A FE14 39                  RTS
00480                    *
00481                    * CLOCK
00482                    *
00483A FE15 B6 8404  A CLOCK    LDAA    PORTC       CLEAR INTERRUPT
00484A FE18 7C 0048  A          INC     TIMEL0      LSB
00485A FE1B 26 03 FE20          BNE     CL2
00486A FE1D 7C 0046  A          INC     TIMEHI      MSB
```

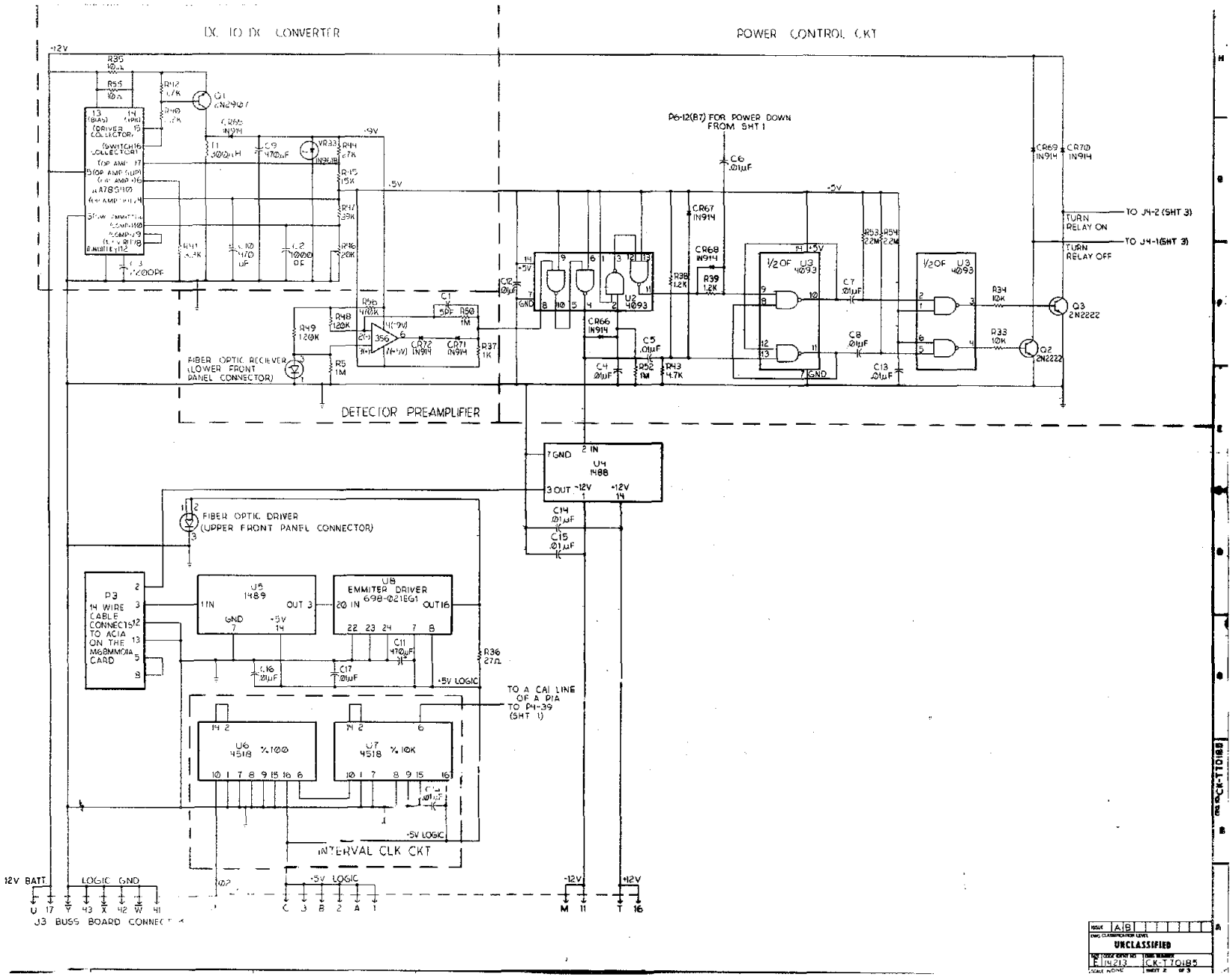
```
00487A FE20 3B          CL2     RTI             RETURN FROM INTERRUPT
00488                   *
00489                   * SWI
00490                   *
00491A FE21 3B          SWI     RTI
00492                   *
00493                   * NMI
00494                   *
00495A FE22 3B          NMI     RTI
00496A FFF8                     ORG     $FFF8   RESET AND INTERRUPT VECTORS
00497A FFF8     FE15  A         FDB     CLOCK   IRQ
00498A FFFA     FE21  A         FDB     SWI     SWI
00499A FFFC     FE22  A         FDB     NMI     NMI
00500A FFFE     FC1C  A         FDB     INIT    RESET
00501           FC1C  A         END     INIT
TOTAL ERRORS 00000--00000


   8409 ACIA    00129*00213 00250 00387 00390
   8408 ACIAC   00128*00192 00194 00210 00247 00271 00383 00395
   E000 ADBASE  00130*00433
   FE20 CL2     00485 00487*
   FE15 CLOCK   00483*00497
   FCDC CLRCLK  00142 00278*
   FC00 COMTAB  00140*00223
   FCD1 CONFIL  00141 00270*00273
   FD8B FILQUE  00270 00299 00405*
   FD9A FQ2     00411*00478
   FDAC FQ4     00417 00420*
   FDB0 FQ6     00415 00422*
   FDB9 FQ8     00423 00426*
   FDBE FQA     00413 00428*
   FDC4 FQC     00429 00431*
   FDF1 FQD     00445 00459*
   FE02 FQE     00419 00421 00425 00427 00458 00469*
   FC88 ID2     00219 00221*
   FC70 IDLE    00210*00212 00215 00217 00233
   FC1C INIT    00154*00500 00501
   0043 LASCOM  00108*00198 00220
   FCA8 LQ2     00247*00249 00265
   FCAF LQ4     00250*
   FCC4 LQ6     00256 00258 00260*
   FCD0 LQ8     00252 00254 00259 00266*
   FCA2 LQLIST  00140 00245*
   0051 MASK    00117*00286 00293 00399
   FE22 NMI     00495*00499
   FCF8 ONEQUE  00145 00299*
   FD6B OUT2    00383*00385 00390
   FD7B OUT4    00382 00391*
   FD0D OUTLST  00147 00314*
   FD68 OUTPUT  00302 00309 00316 00323 00381*
   004E P0      00115*00224 00225 00226 00228 00229 00230 00343 00350 00353
                00434 00435 00436 00438 00439 00440
   0050 P1      00116*00345 00346 00349
   8400 PORTA   00120*00181 00426
   8401 PORTAC  00121*00177 00187
   8402 PORTB   00122*00174 00184 00284 00361
```

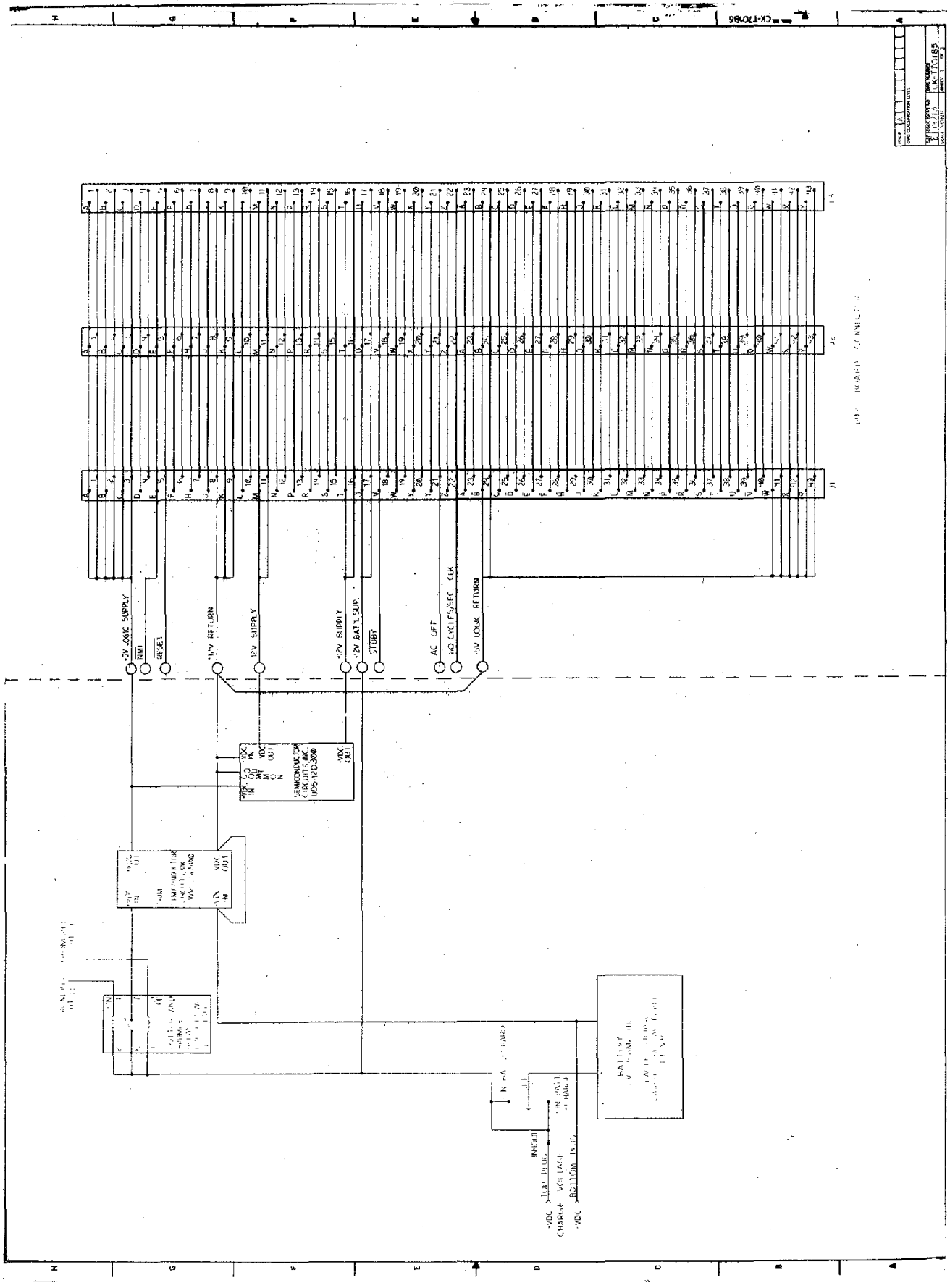
```
8403 PORTBC  00123*00171  00178  00180
8404 PORTC   00124*00182  00424  00483
8405 PORTCC  00125*00179  00189
8406 PORTD   00126*00175  00185  00292
8407 PORTDC  00127*00172  00180  00190
FD63 PWRDWN  00151  00361*
0045 QCOUNT  00110*00200  00246  00262  00263  00315  00341  00406
0000 QDATA   00104*00300  00307  00407
0044 QDNUM   00109*00199  00301  00308  00405  00466  00473
004A QDX     00113*00408  00462  00465  00469  00472
0016 QFULL   00106*00264
002C QLIST   00105*00260  00314  00342  00409
004C QLX     00114*00410  00474  00476
FD04 QUEOUT  00146  00307*
FD7C SETPOR  00287  00294  00395*00397
FD8A SP8     00401*
FCE3 SPORTB  00143  00284*
FCEE SPORTD  00144  00292*
FD3D ST2     00346*00355
FD2F STANDQ  00150  00340*
FD16 STAOUT  00148  00321*
0042 STATUS  00107*00196  00321  00328  00332  00335  00443
FD4F STDLST  00344  00357*
FE21 SWI     00491*00498
0046 TIMEHI  00111*00279  00418  00486
0048 TIMELO  00112*00278  00420  00484
FD2A TO2     00330  00334*
FD1F TOGMOD  00149  00328*
FD67 UNDEF   00152  00153  00369*
```

APPENDIX B

Schematics


4. COMPONENTS ON SHT 1 ARE AS FOLLOWS:
R1 - R32 = 10K
CR1-CR64 = 1N914
VR1-VR32 = 1N752A
NOTES:
P5, P6, P7 AND P8 ARE FRONT PANEL CONNECTORS J0 THROUGH J5, L, G, AND H ARE DIFFERENTIAL ANALOG INPUTS).
P2 AND P4 ARE CABLE CONNECTORS TO CORRESPONDING PINS ON THE M68MMDLA MONABOARD MICROCOMPUTER.
J1-A AND J1-B ARE CONNECTED THROUGH A JUMPER CARD TO CORRESPONDING CONNECTIONS ON THE A TO D CARD 1ST-6800).
P6
P7
P8
TO C6 (SHT 2) FOR POWER DOWN
FROM U1 PIN6 (SHT 2)
CAT LINE OF A PIA
P4
J1 A
ANALOG GND
-12V
+12V
J3 BUSS BOARD CONNECTOR
TX DATA AQUISITION SYSTEM PC BOARD 3 SCHEMATIC
UNCLASSIFIED
UNCLASSIFIED
E
CK-T70185


DC TO DC CONVERTER
POWER CONTROL CKT
+12V
R39 10Ω
R55 10Ω
R12 1.7K
R60 1.7K
Q1 2N2907
CR65 IN914
L1 30Ω@1KH
C9 470μF
VR33 IN5118
R14 27K
R15 15K
+5V
-9V
R47 39K
R46 20K
C1 5PF
R50 1M
R48 120K
R49 120K
R5 1M
CR72 IN914
CR71 IN914
R37 1K
FIBER OPTIC RECEIVER (LOWER FRONT PANEL CONNECTOR)
DETECTOR PREAMPLIFIER
P6-12(B7) FOR POWER DOWN FROM SHT 1
C6 .01μF
CR67 IN914
CR68 IN914
R38 12K
R39 12K
CR66 IN914
C5 .01μF
R52 1M
R43 4.7K
C4 .01μF
C12 .01μF
U2 4093
1/2 OF U3 4093
C7 .01μF
C8 .01μF
C13 .01μF
R53 2.2M
R54 2.2M
R34 10K
R33 10K
Q3 2N2222
Q2 2N2222
CR69 IN914
CR70 IN914
TO J4-2 (SHT 3)
TURN RELAY ON
TO J4-1(SHT 3)
TURN RELAY OFF
U4 1488
C14 .01μF
C15 .01μF
FIBER OPTIC DRIVER (UPPER FRONT PANEL CONNECTOR)
P3 14 WIRE CABLE CONNECTS TO ACIA ON THE M68MMOIA CARD
U5 1489
U8 EMMITER DRIVER 698-021BG1
C11 470μF
C16 .01μF
C17 .01μF
R36 27Ω
+5V LOGIC
TO A CA1 LINE OF A PIA TO P4-39 (SHT 1)
U6 4518 ×100
U7 4518 ×10K
C18 .01μF
INTERVAL CLK CKT
12V BATT
LOGIC GND
J3 BUSS BOARD CONNECTOR
-12V
+12V
UNCLASSIFIED
E 14213
CK-T70185


+5V LOGIC SUPPLY
+12V SUPPLY
-12V BATT. SUP.
AC OFF
+5V LOGIC RETURN

DISTRIBUTION:

400 C. Winter
1200 L. D. Smith
1220 J. M. Wiesen
1400 L. J. Heilman
1470 R. S. Wilson
1500 W. A. Gardner
1520 T. J. Hoban
1530 W. E. Caldes
1540 R. L. Brin
1550 F. W. Neilson
1551 D. W. Ballard
1552 O. J. Burchett
1552 G. C. Stoker (15)
1552 C. L. Sharp
1553 R. L. Parker
1554 J. C. Bushnell
1554 W. P. Brigham
1554 R. J. Goode
1554 J. G. Kostas
1556 S. A. Ingham
4300 R. L. Peurifoy Jr.
4310 C. C. Burks
4320 G. J. Hildebrandt
4330 E. E. Ives
4340 H. W. Schmitt
4360 J. A. Hood
5520 T. B. Lane
5600 D. B. Shuster
5800 R. S. Claassen
8214 M. A. Pound
8310 D. M. Schuster
8440 D. E. Gregson
8444 A. R. Willis (2)
3141 L. J. Erickson (5)
3151 W. L. Garner (3)
For: DOE/TIC
(Unlimited Release)
DOE/TIC (25)
(C. Dalin, 3154-3)