ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *MTS / DSP Technology 6001* at our website: *Click HERE*

DSP TECHNOLOGY INC.

TECHNICAL REFERENCE MANUAL

MODEL 6001/6002 CAMAC CRATE CONTROLLER
MODEL PC001 APPLE II INTERFACE CARD
MODEL PC004 IBM PC INTERFACE CARD
MODEL PC006 IBM PS/S INTERFACE CARD
MODEL 9100 IBM PC SOFTWARE DRIVERS

PROPRIETARY STATEMENT

"Notice is hereby given that all data contained in, revealed by, or shown on this document, whether a drawing or other writing, is proprietary and belongs to DSP TECHNOLOGY INC., Fremont, California. It is furnished and received in confidence solely for information purposes or procurement of the product to which it relates. Neither the document nor any information/data contained in it shall be used for any other purpose or duplicated or disclosed in whole or in part to anyone else without the prior written authorization of DSP TECHNOLOGY INC. in each instance. If such written authorization be given by DSP, any disclosure or revelation by the recipient to anyone else shall be accompanied by a written statement incorporating the language and contents of this legend"

TABLE OF CONTENTS

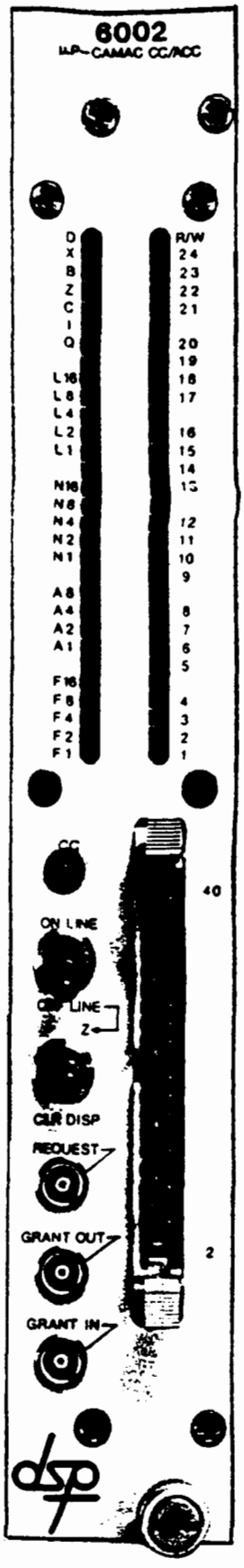
	PAGE
Data Sheet - Models 6001/6002 and PCOO4	i
Section 1 - Operational Description	1-1
A) General	1-1
B) Front Panel Control & Signals	1-1
Section 2 - Interfacing	2-1
A) Hardware	2-1
B) Software	2-3
Section 3 - Auxiliary/Master Operation	3-1
Section 4 - Apple II Interfacing	4-1
A) Hardware	4-1
B) Software	4-1
Section 5 - IBM Personal Computer Interfacing	5-1
A) Hardware	5-1
B) Software - Model 9100	5-3
Section 6 - PS/2 Hardware	6-1
Section 7 - Functional Description	7-1
A) 6001-1	7-1
B) 6002-2	7-2
C) PC001 APPLE II Interface Card	7-3
D) PC004 IBM DMA Interface Card	7-3
Appendix A - Software Trouble Report Form	A-1
Appendix B - History of PSP9100 Software Releases	B-1
CAMAC Appendix	
Schematics	
Assembly Drawings	
Parts Lists	

November, 1990

MODELS 6001/6002 and PC004
IBM PC/XT/AT CRATE CONTROLLER & DATAWAY DISPLAY

FEATURES

- High Speed DMA Data Transfer (to 900 KBytes Per Second)
- Includes CAMAC Support Software for Most Popular Programming Languages
- TurnKey Menu Driven Data Acquisition System Software Available for Set-up, Control, Data Transfer, Display and Analysis
- Built-in Dataway Display Simplifies System Maintenance and Program Development
- Compatible with Microcomputer Type Buses.



ACTUAL SIZE

DESCRIPTION

Either a Model 6001 or 6002 CAMAC Crate Controller together with a Model PC004 plug-in card for an IBM PC/XT/AT compatible personal computer (including 80386 based machines) provides a high speed DMA interface between the computer and the instrument modules installed in the CAMAC crate. Support software for most popular programming languages is included. Higher level menu driven data acquisition system support software is available separately.

The Model 6001 and 6002 are combined CAMAC Crate Controllers and Dataway Display Modules. This saves one CAMAC slot normally necessary for the dataway display function, a very useful tool in debugging data acquisition software. The status of the last CAMAC cycle executed is displayed on the front panel LEDs.

The module accepts data and commands from the computer, executes the CAMAC commands and transfers data back to the computer. Direct memory access (DMA) data transfers of up to 900 kbytes per second are supported.

The Model 6002, in addition to being able to function as a standard crate controller, can also be used as an Auxiliary Crate Controller (ACC). For example, two Model 6002s can be installed in a single crate at the same time to allow two different IBM compatible computers access to the instrumentation modules. Similarly, a Model 6002 can be installed along with another model Auxiliary Crate Controller to simultaneously interface a CAMAC crate to both an IBM compatible and to another type host computer such as a DEC VAX or an IEEE-488 (GPIB) controller.

The Model PC004 is an IBM PC plug-in card that interfaces to either a 6001 or 6002. Support software (Model 9100) for programmed I/O and DMA transfers are included with the PC004. A single PC004 provides connections to up to four 6001/6002s (i.e. up to four crates). An interconnect cable is included with the 6001/6002. Jumpers can be installed to permit LAM interrupt servicing.

The programming languages supported are:

BASIC -	Microsoft/IBM, Microsoft Quick BASIC
FORTRAN -	IBM, Microsoft, Lahey, Ryan-McFarlane
Pascal -	Microsoft, Borland
C-	Microsoft, Borland, Lattice
ASYST	

Assembly language source code is also provided.

SPECIFICATIONS - MODELS 6001/6002 and PC004

FRONT PANEL

LED Indicators: Display the result of the last CAMAC command executed. LAM lights, which are priority encoded, are on whenever a module generates a LAM.

Clear Display: Push button which clears the display.

Clear Dataway: Generates a CAMAC clear.

I/O: 40 pin connector which interfaces the module to the host computer.

Cable: Replacement cable P/N 8100-003

6002 ONLY:

CC LED: When lit, indicates that 6002 is in main Crate Controller mode and must be installed in two rightmost slots.

ON LINE/OFFLINE SWITCH: Allows 6002 to be set into "Offline" mode in which it will not issue CAMAC cycles.

REQUEST Connector: ACC Request line. LEMO connector.

GRANT OUT Connector: ACC Grant Out Line. LEMO Connector

GRANT IN Connector: ACC Grant In Wire. LEMO Connector.

NOTE: For proper Auxiliary Crate Controller operation, the highest-priority crate controller should have its Request connected to its Grant In. A lower-priority crate controller should have its Grant In connected to the Grant Out of the crate controller with the next highest priority. Operation of the 6002 without other crate controllers requires that it be configured as a Main Crate Controller and that its Request be connected to its Grant In.

AUXILIARY CONTROL BUS: The 6002 has a CAMAC standard auxiliary control bus connector on the rear panel. IEEE Std. #675 is the specification for this bus.

POWER REQUIREMENTS

+6V, 1.3A.

PACKAGING

#2 width CAMAC module
221 mm H, 34 mm W, 292 mm D* (8.7" x 1.4" x 11.5")
*Depth front to rear panel. Rear connector 13 mm (0.5").
In conformance with the CAMAC standard for RF shielded instrumentation modules (IEEE standard 583, European Esone Report # EUR4100e).

TEMPERATURE RANGE

0° to 40° C (32° F to 104° F)ambient to operate within specifications (when installed in crate with enough air flow to hold maximum air exit temperature to 55° C (131° F).

COMPUTER COMMANDS

LAMs: The CAMAC LAMs (Look at Me) are priority encoded with station 1 as highest priority and 23 as lowest.

A flag bit is generated in the 6001/6002 whenever a LAM signal is present within the crate.

Q, X: The Q and X bits reflect the status of the CAMAC X and Q lines during the last CAMAC cycle.

Z, C: A CAMAC Initialize or Clear command may be generated by loading the appropriate bit of register A6 and initiating a CAMAC cycle. These bits are reset after the CAMAC cycle is completed.

I: Sets the CAMAC inhibit line.

Data read or written is passed over a byte-wide, bidirectional bus controlled by four address lines and six control lines. The four address lines which select internal registers for reading/writing data are:

Address	Address Lines BA8	BA4	BA2	BA1	Function
A0	L	L	L	L	Stores CAMAC WH BYTE (W17-W24) Data
A1	L	L	L	H	Stores CAMAC WM BYTE (W9-W16) Data
A2	L	L	H	L	Stores CAMAC WL BYTE (W1-W8) Data
A3	L	L	H	H	Stores CAMAC A Subaddress Data (0-15)
A4	L	H	L	L	Stores CAMAC F Function Data (0-31)
A5	L	H	L	H	Stores CAMAC N Station Data (0-23)
A6	L	H	H	L	Stores CAMAC Z,C,I, and Aux Inhibit Hold Control Data
A7	L	H	H	H	Starts a CAMAC cycle
A8	H	L	L	L	Reads CAMAC Q,X, and encoded LAM
A9	H	L	L	H	Reads RH BYTE (R17-R24) Data
A10	H	L	H	L	Reads RM BYTE (R9-R16) Data
A11	H	L	H	H	Reads RL BYTE (R1-R8) Data
A12	H	H	L	L	Reads Status Register (LSB-CAMAC Cycle Complete, LSB+1-ACL State, LSB+2-High
A15	H	H	H	H	DMA control and status register (PC004) for on line

The six control lines are:

BDS: Master enable which must be low to address the unit.

BR/W: Controls data flow on the bidirectional bus-high to read, low when addressing registers 0-6 and high for registers 8-11.

BQ3: Timing signal used when writing data into the unit.

CC: Active low TTL pulse initiates CAMAC Cycle.

L: Active low TTL when LAM is present.

Q: Q status of last CAMAC cycle.

The following are trademarks or registered trademarks of their respective companies: IBM PC/XT/AT & IBM Professional Fortran of International Business Machines Corporation. DEC VAX of Digital Equipment Corporation. Microsoft of Microsoft Corporation, Turbo C and Turbo Pascal of Borland International, Inc. ASYST of Asyst Software Technologies, Inc.

SECTION 1 - OPERATIONAL DESCRIPTION

A) GENERAL

DSPT Models 6001/6002 are general-purpose CAMAC crate controllers designed to interface to microprocessors. The two units differ in that the 6001 can only operate as a master crate controller while the 6002 can function either as a master or as an auxiliary crate controller. The units act as external registers to hold CAMAC commands (N,F,A) and transfer data to/from the CPU. The modules have a minimum of interface control signals and are easily interfaced to any 8- or 16-bit microprocessor. Some specialized interface cards and software are available from DSPT. These are described later in this manual (see Sections 4 and 5).

B) FRONT PANEL SIGNALS AND CONTROLS

DATAWAY DISPLAY - Monitors dataway commands and data. The display LED's latch and hold the status of the last active CAMAC cycle (except for LAM lines, which may change at any time).

LED INDICATORS:

D	Lights for approximately 1/2 sec. whenever a CAMAC cycle is initiated.
X	CAMAC X signal
B	CAMAC BUSY signal
Z	CAMAC INITIALIZE signal
C	CAMAC CLEAR signal
I	CAMAC INHIBIT signal
Q	CAMAC Q signal
L16-L1	Binary-encoded value of the station generating the LAM signal; priority-encoded; station one has highest priority.
N16-N1	Binary-encoded value of last addressed CAMAC (station)
A8-A1	Binary-encoded value of CAMAC subaddress A (0-15)
F16-F1	Binary-encoded value of CAMAC command F (0-31)

R/W24-1 Data read/written during last CAMAC cycle. Read data (R24-R1) is displayed if the last cycle was F0-F7 and write data (W24-W1) is displayed for F16-F31 commands.

CC - (6002) Indicates the internal pull-up resistor packs are in and that the unit is not an auxiliary crate controller (i.e., unit is functioning as master crate controller).

CONTROLS:

ON/OFFLINE Two-position switch, enables/disables unit to request CAMAC cycles on the CAMAC dataway. Does not affect GRANTIN/GRANT OUT signals.

Z/CLR DISP Two-position momentary switch. If the unit is 'OFFLINE', momentarily pressing the 'Z' switch will generate a CAMAC INITIALIZE (Z) cycle.Pressing the 'CLR DISP' switch clears the dataway display LED's.

REQUEST LEMO-type connector,open collector, active (6002)low. This signal must be connected to 'GRANT IN' if the 6002 is the crate controller with highest priority.

GRANT OUT LEMO-type connector, open collector, active (6002)low. This signal is connected to the 'GRANT IN' of the next lower priority controller in the crate.

GRANT IN (6002) LEMO-type connector, open collector. Receives the 'GRANT OUT' from the previous auxiliary/master crate controller.

Note: 'GRANT IN' & 'GRANT OUT' set the priority for accessing the CAMAC dataway. The crate controller (CC) which is to have the highest priority has the 'REQUEST' signal connected to'GRANT IN'. The 'GRANT OUT' of this CC is then connected to the 'GRANT IN' of the CC which is to have second priority. The 'GRANT OUT' of this second CC would be fed to the 'GRANT IN' of any additional CC, etc. The master CC, occupying the rightmost slot, would typically have the highest priority.

INTERFACE I/O CONNECTOR:

40-pin connector, accepts standard flat cable header, TTL signals for data I/O and control. See Table 1 in Section 2 for pin assignments.

SECTION 2 - INTERFACING

A) HARDWARE

The 6001/6002 can be interfaced to any 8- or 16-bit microprocessor I/O bus. All commands are passed via the four-bit address bus (BA1,BA2, BA4,BA8) and all data via the 8-bit bi-directional data bus (DB7-DB0). See Table 1 for interface signal descriptions and Diagram 1 for timing. The 6001/6002 appears simply as twelve external 8-bit registers to the CPU. These registers are read or written using the data bus.

Registers (See Table 2):

0 Write-only register for CAMAC write data W24-W17

1 Write-only register for CAMAC write data W16-W9

2 Write-only register for CAMAC write data W8-W1

3 Write-only register for CAMAC A subaddress A8-A1

4 Write-only register for CAMAC F command F16-F1

5 Write-only register for CAMAC N station address N16-N1 (1-23 stations)

6 Write-only register for CAMAC Z, C, I bits and for the control of auxiliary crate controller modes (6002 only).

Note 1: Z or C clear the N, F, A registers during the CAMAC cycle. After completion of the cycle the Z or C bits are also cleared.

Note 2: Aux CC control bits are:

AX1=1 (bus inhibit) Once the 6002 initiates a CAMAC cycle after being granted the bus, it will not give it up to allow other crate controllers to access the CAMAC dataway (unless an ACL has occurred). Used mainly for DMA transfers. (Keeps auxiliary bus "REQUEST INHIBIT' on.)

AX2=1 Resets the bus inhibit initiated by setting AX1=1/ (Resets 'REQUEST INHIBIT' signal on auxiliary signal bus.)

AX3=1 Resets the ACL detection register to zero (state of ACL

register read by addressing register number 12).

AX4=1 Resets the 6002 (ACL detection register and bus 'REQUEST INHIBIT' are turned off).

7 Initiates the 1.1 microsecond CAMAC cycle. For CAMAC write cycles, the contents of register 0, 1, and 2 are gated onto the dataway with the N, A registers. For CAMAC read cycles, data is latched into registers 9, 10, and 11 for subsequent readout, if desired.

8 Read-only register to read LAM status, Q, and X. The Q and X bits reflect the status of the last CAMAC write cycle performed. LAM lines are priority-encoded, i.e., station 1 has the highest priority and station 23 the lowest. For example, if station 16 generates a LAM, the CPU will read LAM=16. However if station 1 generates a LAM, it will take priority and the computer will read LAM=1. The L line goes low anytime a LAM is generated and can be used as a test bit. A buffered L line is also available for interrupts.

9 Read-only register for CAMAC read data R24-R17

10 Read-only register for CAMAC read data R16-R9

11 Read-only register for CAMAC read data R8-R1

12 Read-only register for crate controller status
Note: crate controller status bits are:

AC=1 if the last CAMAC cycle requested was completed.
AL=1 if an ACL signal has occurred on the auxiliary bus.
AO=1 if the 6001/6002 is 'on line'.

Interface Timing (See Diagram 1)

Write Operations: Data is set-up to the registers when BR/W is low. (REGISTERS 0-7)The data is clocked in on the positive edge of BQ3. The CAMAC cycle starts approximately 200 ns. after the positive edge of BQ3.

Read Operations: Data is ready on DB7-DB0 within 100 ns. of receipt (REGISTERS 8-12)of addresses and BR/W going high.

B) SOFTWARE

These examples show how to control the 6001/6002 for CAMAC operation.

Write Operation:

1) Write W24-W17 byte into	Register 0
2) Write W16-W9 byte into	Register 1
3) Write W8-W1 byte into	Register 2
4) Write A subaddress into	Register 3
5) Write F command into	Register 4
6) Write N station into	Register 5
7) Initiate a CAMAC cycle by addressing	Register 7
8) Check Q & X responses by reading	Register 8

Read Operation:

1) Write A subaddress into	Register 3
2) Write F command into	Register 4
3) Write N station number into	Register 5
4) Initiate a CAMAC cycle by addressing	Register 7
5) Check Q & X responses by reading	Register 8
6) Read R24-R17 byte from	Register 9
7) Read R16-R9 byte from	Register 10
8) Read R8-R1 byte from	Register 11

TABLE 1
I/O CONNECTOR PINOUT - J4

PIN #	SIGNAL	DESCRIPTION
35	CC	Initiate CAMAC cycle without using BA8-BA1
33	L	LAM(look-at-me) active at some station
31	Q	Q status from last CAMAC cycle
29	BQ3	Timing strobe to write into 6001/6002 or start CAMAC cycle
27	BA8	Address register, MSB
25	BA4	Address register
23	BA2	Address register
21	BA1	Address register, LSB
19	BDS	DEVICE SELECT
17	BR/W	READ/WRITE SELECT
15	DB7	Data bit, MSB
13	DB6	Data bit
11	DB5	Data bit
9	DB4	Data bit
7	DB3	Data bit
5	DB2	Data bit
3	DB1	Data bit
1	DB0	Data bit, LSB

TABLE 2
6001/6002 REGISTER DESCRIPTION

REG:	ADDRESS BUS:				DATA BUS:								DESCRIPTION:
	BA8	BA4	BA2	BA1	BD7	BD6	BD5	BD4	BD3	BD2	BD1	BD0	
0	L	L	L	L	W24	W23	W22	W21	W20	W19	W18	W17	WRITE byte
1	L	L	L	H	W16	W15	W14	W13	W12	W11	W10	W9	" "
2	L	L	H	L	W8	W7	W6	W5	W4	W3	W2	W1	" "
3	L	L	H	H			A8	A4	A2	A1			A Address
4	L	H	L	L				F16	F8	F4	F2	F1	F Command
5	L	H	L	H				N16	N8	N4	N2	N1	N Station
6	L	H	H	L		AX4	AX3	AX2	AX1	I	C	Z	Z,C,I bits
7	L	H	H	H	----------ANY VALUE----------								CAMAC cycle
8	H	L	L	L	L	L16	L8	L4	L2	L1	X	Q	LAM,X,Q
9	H	L	L	H	R24	R23	R22	R21	R20	R19	R18	R17	READ byte
10	H	L	H	L	R16	R15	R14	R13	R12	R11	R10	R9	" "
11	H	L	H	H	R8	R7	R6	R5	R4	R3	R2	R1	" "
12	H	H	L	L					AO	AL	AC		Status

SECTION 3

6002 AUXILIARY/MASTER CRATE CONTROLLER OPERATION

The 6002 is delivered as a master crate controller. In this mode the user need only to connect the 'REQUEST' signal to the 'GRANT IN'.

To convert the 6002 to an auxiliary crate controller, open the unit and:

1) On the right-side board (6002-2) remove the 5 socketed resistor packs at the rear of the unit (RP1, RP2, RP3, RP4, RP5).

2) On the left-side board (6001-1) remove the four socketed resistor packs (RP1, RP2, RP3, RP4).

3) Close the unit, insert in the crate and connect a 40-pin flat cable to the rear connector. Connect the other end of the cable to the main crate controller.

4) Connect the 'GRANT OUT' of the main crate controller to the 'GRANT IN' of the 6002.

SECTION 4
APPLE II INTERFACING

A) HARDWARE

The Model PC001 is supplied by DSPT as an interface card for the APPLE II. It is inserted into APPLE I/O slot 2 (other slots can be used, but the DSPT-supplied software assumes the I/O card to be in this slot).

B) SOFTWARE

Three programs are supplied on the APPLE diskette for CAMAC I/O. The first program is a machine language routine called "CAMAC" which controls the 6001/6002 and consists of five callable routines:

1) CO - Outputs N%,F%,A%,D%; initiates a CAMAC cycle and returns Q% and X% (see below for variable explanations).
2) CI - Outputs N%,F%,A%,D%; initiates a CAMAC cycle and returns D%, X%, and Q%.
3) CL - Checks for the priority encoded LAM and returns the station number of the module generating the LAM in L%, L%=0 for no LAM.
4) CZ - generates a CAMAC Z (INITIALIZE cycle)
5) CC - generates a CAMAC C (CLEAR cycle)

This program is written in machine language and is loaded by typing "BLOAD CAMAC".

The second program is an example of a BASIC routine calling the "CAMAC" I/O subroutines. Line numbers 300 and above may be deleted or modified to develop the user's program. However, Lines 20-100 must not be changed. The order of the variables in Line 20 is used by "CAMAC" to determine their relative memory locations. This program is loaded by typing "LOAD CAMBAS".

The third program is a test program for the 6001 or 6002 and can be used to check dataway operation and LED status. The program is loaded by typing "LOAD TEST6001". This program contains some loops for scope probing of the unit which are exited only by hitting RESET.

Note: Both "CAMBAS" and "TEST6001" need "CAMAC" loaded before execution.

N% = CAMAC station number 1-23
F% = CAMAC command 0-31
A% = CAMAC subaddress 0-15
D% = READ/WRITE data, 16-bit maximum
X% = CAMAC X response (0 or 1)
Q% = CAMAC Q response (0 or 1)
L% = Station number of module generating LAM (1-23)

SECTION 5
IBM PERSONAL COMPUTER INTERFACING

A) HARDWARE

The interface card, the Model PC004, fits into any available full-length expansion slot of the IBM PC or XT System Unit or into any "8-bit" expansion slot (single-connector) of the AT System Unit. The card can control up to four 6001/6002's.

Two modes for data transfer to and from CAMAC are supported in the PC004. The two modes are:

1) "Word Count"-
   The number of data words to be transferred to/from a single crate, station number (N), function code (F), and subaddress value (A) is predetermined and a sufficient number of CAMAC cycles are issued to complete the transfer. No check of X or Q is made during this transfer.

2) "Q-STOP"-
   The number of data words transferred to/from a single crate, station number (N), function code (F), and subaddress value (A) is determined by the Q response of the addressed module. Data continues to be transferred as long as a Q=1 response is generated. As soon as a Q=0 response is detected, the DMA transfer is terminated. A maximum number of data transfers is specified prior to initiation of the DMA procedure.

In both modes, the amount of data transferred for each CAMAC cycle can be determined as: 1)one byte (R/W1-8); 2)two bytes (R/W1-16); or 3)three bytes (R/W1-24). This feature reduces the number of I/O bus transfers and memory locations required for those modules which use fewer than the full 24 bits available on the CAMAC dataway.

Two DMA modes are supported by the PC004:

a) Block Mode
b) Single Cycle Transfer mode.

Block mode is a very high speed DMA transfer done in bursts of 16 byte transfers. Single cycle is a slower DMA transfer, approximately one-half the speed of Block mode. Block mode will only run on machines whose memory access time is compatible with the full speed of the DMA controller chip. Machines with slower memory will generate parity errors during DMA block mode transfers. DSPT provides software routines for both types of transfer.

For Block mode or single cycle transfer the PC004 should have jumpers E12 to E13 and E11 to E9 installed.

Addressing:

The card uses 16 consecutive addresses on the IBM PC/XT I/O bus. The four LSB addresses on the IBM I/O connector are used to address the I/O card and, consequently, the 6001/6002 registers. An on-board set of switches select the I/O address range (I/O address lines A9-A4) as follows:

```
I/O ADDRESS LINE   [A9! A8! A7! A6! A5! A4]     Note: Switch "ON" corresponds
SWITCH NUMBER      [SW1!SW2!SW3!SW4!SW5!SW6]          responds to "0".
```

DSPT software assumes an address range of hex 240-24F. Consequently, the switch positions should be set as follows (unless the software has been modified):

```
SWITCH NUMBER      [SW1! SW2! SW3! SW4! SW5! SW6]
                   [OFF! ON!  ON!  OFF! ON!  ON ]
```

Register addresses 0-12 directly address the registers of the 6001/6002 crate controller (see Section 2A). In addition, the card contains an on-board eight-bit register at the sixteenth register location (e.g., I/O address hex 24F). The register serves as both a command (write) and a status (read) register. The command register is programmed as follows:

D0-D1	Program the number of bytes per CAMAC cycle to be transferred in DMA mode; = 0, transfer 1 byte per cycle (R8-R1 or W8-W1) = 1, transfer 2 bytes per cycle (R16-R1 or W16-W1) = 2, transfer 3 bytes per cycle (R24-R1 or W24-W1)
D2	= 0, set DMA to Block Transfer mode (i.e., transfer the requested number of bytes = 1, set DMA mode to Q-STOP Transfer mode (i.e., transfer until CAMAC Q=0 or until the number of bytes programmed, typically 64K)
D3	= 0, set DMA to READ mode (from CAMAC to memory) = 1, set DMA to WRITE mode (from memory to CAMAC)
D4-D5	Program which 6001/6002 is active; = 0, then 6001/6002 connected to J1 is active = 1, then 6001/6002 connected to J2 is active = 2, then 6001/6002 connected to J3 is active = 3, then 6001/6002 connected to J4 is active
D6	= 1 initiates a DMA transfer (hardware reset at end of DMA transfer).
NOTE:	DMA transfers require that the IBM DMA chip be programmed before start of DMA.

The status register provides information on the status of the DMA operation. If D0=0, then the DMA is still in progress. If D0=1, then the DMA operation is complete.

B) SOFTWARE

Software provided with the PC004 provides for convenient access to both the programmed I/O and DMA facilities of the PC004/6001 combination from a variety of interpretive and compiled high-level programming languages for the PC. Supported languages include:

Microsoft BASIC, both interpretive and compiled
Microsoft PASCAL
Turbo PASCAL
Microsoft FORTRAN
IBM Professional FORTRAN (Ryan-MacFarland FORTRAN)
ASYST

As delivered, the package, Model 9100, supports up to four crates, ie one PC004 at the default address of 240 Hex. Complete macro source code is provided so that, among other things, you can add more crates or change the I/O address.

There are two object code files on the distribution disk for each supported language. The standard object files, those with the extensions .OBJ and .BIN, support the single cycle mode of DMA transfer. This mode works reliably on all IBM PCs and compatibles.

The other set of object files, with extensions .OBK and .BBK support the high speed block mode of DMA transfer. To change to block mode, rename or delete the .OBJ or .BIN file for the language you are using and rename the corresponding .OBK or .BBK file to .OBK or .BBK as appropriate. Note, that this mode may not work reliably on some computers, particularly those of early vintage.

B1 Available Procedures

In the descriptions that follow, the following parameter definitions are used:

Variable	Description	Range
C	Crate number	1 to 4
N	Station number	1 to 23
F	Function code	0 to 31
A	Subaddress	0 to 15
D	Data	Depends on procedure
X	X status	0 or 1
Q	Q status	0 or 1

All variables are 16-bit integer scalers except Data (D) which may alternatively be an array of 16-bit integers or a 32-bit integer if the language supports that feature.

In the interest of maximizing performance, no range checking is done on any parameters. Out of range parameters may result in unpredictable behavior of the system.

The procedures are described in "generic" form, that is, independent of a particular language implementation. Details on specific languages is covered in section B2.

1. CAMO (N,F,A,D,Q,X)

Writes 16 bits of data from D to the dataway. F must be greater than 7. This procedure is also used for dataless dataway cycles.

Inputs: N,F,A,D
Returns: Q,X

2. CAMO24 (N,F,A,D,Q,X)

Writes 24 bits of data from D to the dataway. F must be greater than 7. D must be at least 24 bits long but may be longer, for example a two word array of integers or a long (32 bit) integer. Any extra bits are ignored.

Inputs: N,F,A,D
Returns: Q,X

3. CAMI (N,F,A,D,Q,X)

Reads 16 bits of data from the dataway to D. F must be less than 8.

Inputs: N,F,A
Returns: D,Q,X

4. CAMI24 (N,F,A,D,Q,X)

Reads 24 bits of data from the dataway to D. F must be less than 8. D must be at least 24 bits long but may be longer. Any additional bits are ignored and should probably be set to 0 before calling this procedure.

Inputs: N,F,A
Returns: D,Q,X

5. CRATE (C) or CRATE_SET (C)

Sets the current crate number to C. All subsequent CAMAC operations will be to this crate until the next call to this procedure or a call to DMASET described below. Note that this procedure has two equivalent names.

Inputs: C
Returns: None

6. CAML (L)

Returns the station number of the highest priority pending LAM in L. If no LAMS are pending, L = 0. Station 1 has the highest priority, station 23 the lowest. L is a 16 bit integer.

Inputs: None
Returns: L

7. CAMCL (I)

Provides access to the Command Register in the 6001/6002 permitting execution of Z and C cycles, setting and clearing of dataway Inhibit, etc. I is a 16-bit integer of which only the lower seven bits are used.

Bit	Description
1	1 = Execute dataway Z
2	1 = Execute dataway C
3	1 = Set dataway I
	0 = Clear dataway I
4	1 = Set Auxiliary Bus REQUEST INHIBIT line
5	1 = Clear Auxiliary Bus REQUEST INHIBIT line
6	1 = Clear ACL Detection Register
7	1 = Clear both REQUEST INHIBIT and ACL Detection Register.

CAMCL should be called with I = 64 (bit 7 = 1) at the beginning of a program to each crate to insure proper initialization of the 6001/6002.

Inputs: I
Returns: None

8. DMASET (C,NOB,QBL,NTR)

Sets up parameters for subsequent DMA transfers using DMAI or DMAO.

NOB- Number of bytes transferred per dataway cycle, 1 to 3. Note that the type of the array used for data transfer should match NOB; that is, NOB = 1 implies an array of bytes and NOB = 2 implies an array of 16-bit integers. 3-byte transfers are a special case in that the data will probably have to be reformatted into 32-bit integers for subsequent processing.

NTR- Maximum number of dataway transfers requested. Note that NTR*NOB must be less than 65536.

QBL- Mode control. 0 = block mode; ie, do NTR dataway cycles. 1 = Q mode; ie, transfer until Q = 0 or NTR cycles have been done.

The above parameters are 16-bit integers. NTR is treated as an unsigned integer with a maximum value of 65535.

DMASET must be called once before starting a DMA transfer. Subsequent transfers may be initiated without calling DMASET again provided that no parameters change. Note that DMASET sets the currently active crate.

Inputs: C,NOB,QBL,NTR
Returns: None

9. DMAO (N,F,A,D,E)

Transfers data from memory to the dataway according to the parameters set by DMA SET. F must be greater than 7. D is an array of at least NTR*NOB bytes. E is an error return code as follows:

0 No error
1 6002 could not get access to dataway.
2 ACL (Auxiliary Controller Lockout) was asserted sometime during the transfer. Operation should be repeated.
3 Hardware timeout in PC004. This indicates a hardware failure.
4 Addressed CAMAC module returned no Q or X to the first dataway cycle.

Error codes 1 and 2 only apply to the 6002 crate controller with Auxiliary Controller Bus support.

Inputs: N,F,A,D
Returns: E

10. DMAI (N,F,A,D,E)

Transfers data from the dataway to memory according to the parameters set by DMA SET. F must be less than 8. D must be an array of at least NTR*NOB bytes. Error return codes are identical to those for DMAO.

Inputs: N,F,A
Returns: D,E

11. CAMCYC (NTR)

Returns in NTR the number of valid dataway cycles executed during the most recent Q mode DMAI or DMAO operation. That is, the number of cycles that returned Q = 1. NTR is an unsigned integer with a maximum value of 65535. NTR is undefined for Block mode transfers and for operations which result in errors.

Inputs: None
Returns: NTR

B2. Language Specifics

This section presents information specific to each of the languages supported by CAM6001. The core software that specifies the processor operations for each procedure is common to all languages. What differs is the nature of the CALL to the procedure ("near" or "far"), the method of parameter passing (on the stack and in what order, by table, by value, by near pointer, by far pointer, etc.), and whether the call references are determined by a special method or by a link process. Each of the supported languages requires a specific choice of the possibilities (some support more than one choice). The table below indicates the choices made for each of the object file versions. If your language is not one of those supported, it is probable that one of the listed files may be used. Refer to the assembly language interface documentation for your language to determine the answer. Note, however, that DSPT will provide support only for the languages listed in the following sections.

Table of Language Interface Specifics

Filename	CALL form	Entry Point Deter-mination	Parameter Passing	Notes
CAMBASIC. BIN	near	special load	stack near pointer 1st parameter pushed 1st	
CAMBASIC. OBJ	near	link	stack near pointer 1st parameter pushed 1st	
CAMACMS. OBJ	far	link	stack far pointer 1st parameter pushed 1st	
CAMACPRO .OBJ	far	link	far pointer in register pair (ES:BX) points to table of far pointers	
CAMTURBO. BIN	near	special load	stack far pointer 1st parameter pushed 1st	special real number conversion routine
CAMTURBO. OBJ	near	link	stack far pointer 1st parameter pushed 1st	special real number conversion routine
CAMASYST. OBJ	far	special load	stack far pointer 1st parameter pushed 1st	segment/offset reversed from normal order

1. Interpretive BASIC

The above procedures are implemented in the form of a loadable "device driver" in the file CAMBASIC.BIN. This file must be loaded by your BASIC program and entry points for each of the procedures declared. The file CAMACINC.BAS contains a BASIC subroutine for accomplishing this. This file should be MERGEd with your program and the statement "GOSUB 10000" should be added near the beginning of the program.

All parameters are integer, that is the variable names must be suffixed with "%". All parameters must be defined before they are used in a CALL statement, even those like Q, X or E which are strictly return parameters. A statement of the form "Q% = 0" is sufficient.

Examples:

```
10    DIM AI%(1024)
20    GOSUB 10000:  REM  initialize CAMAC
30    C% = 1:  CALL CRATE (C%)
40    N% = 64:  CALL CAMCL (N%)
220   REM  wait for data to be ready
230   F% = 8: N% = 10: A% = 0: D% = 0
231   Q% = 0: X% = 0:
240   CALL CAMO (N%,F%,A%,D%,Q%,X%)
250   IF Q% = 0 GOTO 240
260   REM  set up DMA channel
270   NT% = 1024: QM% = 1: NB% = 2
280   CALL DMASET (C%,NB%,QM%,NT%)
290   F% = 2
295   REM  use Q% for error return
300   CALL DMAI (N%,F%,A%,AI%(1),Q%)
310   REM  now clear the LAM
320   F% = 10
330   CALL CAMO (N%,F%,A%,D%,Q%,X%)
340   REM  see how many transfers occurred
350   CALL CAMCYC (NT%)

10000 REM********** Load CAMAC procedures *******
10002 DEF SEG = 0:   REM look at low memory
10004 V = PEEK(&H510) + (256 * PEEK(&H511))
10006 SEGID       = V + &H2000:       REM go up 2 segments
10007 DEF SEG = SEGID
10008 BLOAD "CAMBASIC.BIN",0
10010 V$ = HEX$(V)
10011 PRINT "BASICA loaded in segment ";V$
10012 V$ = HEX$(SEGID)
10013 PRINT "CAMBASIC loaded in segment ";V$
10014 REM Declare entry points for CAMAC procedures
10016 CAMO = 0: CAMI = 6: CAML = 12: CAMCL = 18
10018 DMAO = 24: DMAI = 30: DMASET = 36: CRATE = 42
10020 CAMO24 = 48: CAMI24 = 54: CAMCYC = 60
10022 RETURN
```

2. Compiled BASIC

The use of the procedures in compiled BASIC is the same as for interpretive BASIC except for initialization requirements. Compiled BASIC does not require loading the driver file nor declaring procedure entry points. Instead, the file CAMBASIC.OBJ is linked to your compiled BASIC program as in:

```
BASCOM MYPROG;
LINK MYPROG + CAMBASIC;
```

Consequently, a BASIC program to be compiled should not include the initialization subroutine in CAMACINC.BAS nor the statement "GOSUB 10000".

QuickBASIC operates in a similar fashion, substituting "BC" for "BASCOM" in the compile command. To use CAMAC within the QuickBASIC environment, add CAMBASIC.OBJ to your QuickBASIC Library (.QLB) using the method described in the QuickBASIC User's Manual.

Note to QuickBASIC Version 4 users. The use of "dynamic arrays" as parameters in calls to these procedures requires special handling. All of the procedures must be declared (via the DECLARE statement) to have far pointer parameters (via the SEG declaration) and the object file CAMACMS.OBJ (rather than CAMBASIC.OBJ) must be used in the linking process. A sample declaration is as follows:

```
DECLARE SUB DMAI(SEG N, SEG F, SEG A, SEG DATA, SET STATUS)
```

Examples: See previous section

3. Microsoft PASCAL

The file CAMACMS.OBJ contains the procedures in a form that can be linked with your Microsoft PASCAL program as follows:

```
PAS1 MYPROG;
PAS2
LINK MYPROG + CAMACMS;
```

All parameters are passed by reference in segment and offset (ie VARS) form. Those parameters such as F, N and A which are strictly input may be declared as CONSTS so that a constant value may be passed. This eliminates the need for assigning a constant to a variable before invoking the procedure.

The file CAMMSPAS.INC includes external declarations for all of the procedures. This file should be included near the beginning of your PASCAL program with a statement of the form:

```
{$INCLUDE: 'CAMMSPAS.INC'}
```

Examples:

```
CONST     qmode = 1;
          data_source = 10;  {station number}

VAR       Q,X,error,transfer_count : INTEGER;
          Dat_array : ARRAY [1..1024] OF INTEGER;

{$INCLUDE: 'CAMMSPAS.INC'}

BEGIN
     crate_set (1);  {note use of constant}
     camcl (64);  {initialize crate}

     REPEAT   {wait for LAM}
          camo (data_source,8,0,0,Q,X)
     UNTIL Q = 1;

     dmaset (1,2,qmode,1024);
     dmai (data_source,2,0,ADS dat_array,error);

     IF error = 0 THEN camcyc (transfer_count)
     ELSE
     BEGIN
          { Handle error }
     END;
camo (data_source,10,0,0,Q,X);  {clear LAM}

END.
```

4. Turbo PASCAL

Both Versions 3 and 4 of Turbo Pascal are currently supported. Because of the difference in the way external assembly language subroutines are handled in the two versions, separate instructions are given for each version

4.1 Version 3

The file CAMTURBO.BIN implements the procedures as a loadable driver for Turbo PASCAL in much the same form as that for interpretive BASIC and the same concepts apply, namely the need for your program to load the file and declare entry points for the procedures. All parameters are passed by reference (VAR). The file CAMTURBO.V3 incorporates the necessary statements to accomplish this. This file should be included near the beginning of your program with a statement of the form:
{$I CAMTURBO.V3}

4.2 Version 4

The file CAMTURBO.OBJ implements the procedures as a linkable object file for Turbo PASCAL in much the same form as is done for Microsoft Pascal. All parameters are passed by reference (VAR). The file CAMTURBO.V4 incorporates the necessary statements to accomplish this. This file should be included near the beginning of your program with a statement of the form:
{$I CAMTURBO.V4}

CAMTURBO includes an additional procedure, Real_cnvt, for converting from IEEE short real format to IEEE long real used by TURBO-87, the version of Turbo which supports the 8087. The principal purpose of this procedure is reading the DSPT Model 2032 DVM which returns its data in short real form. Real_cnvt is declared as follows:

```
TYPE long_int = ARRAY [0..1] OF INTEGER;

     PROCEDURE Real_cnvt (VAR IEEE_in : long_int;
                          VAR Real_out : REAL);
```

The input parameter is declared integer to be compatible with the data parameter passed to CAMI and DMAI.

There are two requirements for the proper use of Real_cnvt:

1. You must be using Turbo-87.
2. The 2032 must be initialized for IEEE format readout

Since Turbo does not implement a CONST mechanism for procedure parameters like Microsoft PASCAL, constant values must be assigned to variables before invoking a procedure. Turbo also does not implement the unsigned integer (WORD) data type. Consequently, in calls to DMASET and CAMCYC, Turbo will interpret NTR values greater than 32767 as negative numbers.

Examples:

```
{
    The only difference from Microsoft PASCAL is the need to assign
    constant values to variables before calling procedures.
}

VAR  Q,X,error,transfer_count : INTEGER;
     crate,data_source,D,F,A,ntr : INTEGER;
     qmode,nbytes : INTEGER;
     Dat_array : ARRAY [1..1024] OF INTEGER;

{$I CAMTURBO.XX}

BEGIN
     crate := 1;  crate_set (crate);
     F := 64;  camcl (F);  {initialize crate}
     data_source := 10;  A := 0;

     F := 8;
     REPEAT   {wait for LAM}
          camo (data_source,F,A,D,Q,X)
     UNTIL Q = 1;

     qmode := 1;  ntr := 1024;  nbytes := 2;  F := 2;
     dmaset (crate,nbytes,qmode,ntr);

   {  Note the way in which the data array is passed}
     dmai (data_source,F,A,dat_array[1],error);

     IF error = 0 THEN camcyc (transfer_count)
     ELSE
     BEGIN
          {  Handle error  }
     END;

     F := 10;
     camo (data_source,F,A,D,Q,X);  {clear LAM}

END.
```

5. FORTRAN

Syntactically, procedure calls from the two versions of FORTRAN are identical. All parameters are integers. Constant values may be passed for input parameters.

The file CAMACMS.OBJ contains code for linking to Microsoft FORTRAN with commands of the form:

```
FOR1 MYPROG;
PAS2
LINK MYPROG+CAMACMS;
```

Note to users of Microsoft Fortran Version 4.0: You must use either the Medium or Large models because CAMACMS.OBJ uses FAR calls and passes parameters by FAR reference.

The file CAMACPRO.OBJ contains code for linking to IBM Pro FORTRAN with commands of the form:

```
PROFORT MYPROG;
LINK MYPROG + CAMACPRO,,,PROFORT
```

Note that you must specify the PROFORT library in the link command.

Examples:

```
      INTEGER*2  Q,X,D,error,xfers,datsrc
      INTEGER*2  datary (1024)

      CALL crate (1)
      CALL camcl (64)
      datsrc = 10   !station number

C  wait for LAM
100   CALL camo (datsrc,8,0,D,Q,X)
      IF (Q .EQ. 0) GO TO 100

C  Read data
      CALL dmaset (1,2,1,1024)
      CALL dmai (datsrc,2,0,datary,error)

      IF (error .NE. 0) GO TO 200
      CALL camcyc (xfers)
      GO TO 300
```

```
C  Handle error
200

C  Clear LAM
300    CALL camo (datsrc,10,0,Q,X)
```

6. C

Microsoft C (Version 5 or later) and Borland's Turbo C (Version 1.5 or later) are supported by the object file CAMACMS.OBJ. The "memory model" used must be either the "medium" or the "large" model.

The functions must be declared as Pascal types, in order to accommodate the parameter passing order assumed in CAMACMS.OBJ. The file CAMAC.H contains the appropriate declarations for both language versions.

7. ASYST[1] Version 2

The procedures listed in Section B.1 are implemented in the form of a loadable "device driver" in the file CAMASYST.EXE. This file must be installed by your ASYST program and the entry points associated with each of the procedures determined. The individual procedures are then accessed with the ASYST word 'CALL['.

For example,

```
CALL[ CAM.SEG, CAMI, N, F, A, Data, Q, X ]
```

Where:

CAM.SEG:	the segment portion of the procedure's entry point address
CAMI:	the offset portion of the procedure's entry point address. For clarity, the ASYST variable holding this offset is given the same name as the corresponding procedure name.
N, F,...:	the list of variables as specified in Section B.1 for the specific procedure.

The file CAMASYST.ASY provides ASYST words which implement the installation, entry point name definitions and determinations, and other initialization tasks associated with using the procedures. In addition CAMASYST.ASY defines a set of ASYST words, based on calls to the standard routines, which simplify access to the crate operations. To install the base procedures and the extended ASYST CAMAC words, enter "LOAD CAMASYST.ASY" from the ASYST prompt. (Note: to avoid having to load these words each time ASYST is invoked, a copy of

[1] ASYST is a trademark of Asyst Software Technologies, Rochester, NY

ASYST may then be made using the ASYST word, SAVE. The resulting copy of ASYST will contain these words as a permanent part of its dictionary.) Prior to invoking any of the CAMAC words, your code should invoke the word CAM.INIT, or an equivalent set of operations.

The loading of the file CAMASYST.ASY results in the definition of the following words, in addition to the CAMAC I/O procedure names:

Variables:

CAM.SEG	Integer containing the segment part of the CAMAC I/O procedures address.
CAM.OFF	Offset entry point for the procedure to determine the CAMAC I/O procedure entry points. The entry point names are those of Section B.1. Note that the procedure name "CRATE", not "CRATE_SET", is supported in CAMASYST.ASY.
CAM.Q	Integer containing the status of the Q signal following use of the CAMAC word (see below).
CAM.X	Integer containing the status of the X signal following use of the CAMAC word.
CAM.N	Integer used to determine the station address within the current crate during invocation of CAMAC word.
CAM.A	Integer used to determine the CAMAC subaddress used during invocation of CAMAC word.
CAM.CR	Integer containing the current crate number set by SET.CR ATE (see below). Used by DMA.SET (see below).
CAM.TMP	Integer used for temporary storage of CAMAC data.
CAM.TMP24	Double precision integer used for temporary storage of 24-bit CAMAC data.
CAM.IMAGE	Array used to hold CAMASYST.EXE executable code.

Commands:

(Note that the calling format of each command is shown using the ASYST notation convention showing Stack contents before and after invocation of the word.)

CAM.INIT

cam.init [--]
Performs loading and initialization of the procedures and procedure entry points. Must be invoked prior to invoking any other CAMAC I/O procedure.

SET.CRATE

set.crate [crate# --]
Sets the current crate to the value on the stack and updates CAM.CR.

SET.MODULE

set.module [station,subaddress --]
Sets the current station (CAM.N) and subaddress (CAM.A) to the values on the stack.

CAM.Z

cam.z [--]
Does a CAMAC Initialize cycle (Z) on the current crate.

CAM.C

cam.c [--]
Does a CAMAC Clear cycle (C) on the current crate.

?CAMQ

?camq [-- T/F]
Returns True/False on symbol stack if most recent CAMAC word invocation resulted in a Q = 1/0.

?CAMX

?camx [-- T/F]
Returns True/False on symbol stack if most recent CAMAC word invocation resulted in a X = 1/0.

CAMAC

camac [F -- read.data] F<8
camac [write.data, F --] 15<F<24
camac [F --] Otherwise
Issues CAMAC cycle to the current Crate (CAM.CR) and to the Current Module (CAM.N, CAM.A) with the function code F.

GET.LAM

get.lam [-- lam#]
Returns the station number of the highest priority LAM in the current crate. If no LAM is asserted, value returned is zero.

DMA.SET

dma.set [n.trans,q.mode,n.bytes --]
Sets conditions for subsequent Block (dma) operations:
Crate: CAM.CR;
Number of CAMAC cycles: n.trans;
Word count or Q-Stop mode: q.mode;
Number of bytes per cycle: n.bytes;

CAM.BLOCK

cam.block [data.array, F --]
Executes a block (dma) transfer using the function code F

to/from the current module in the current crate under the conditions of the most recent DMA.SET invocation. For F<8, the transfer is a read operation. For 15<F<24, the transfer is a write operation. For other values of F, an error message is generated.

Examples using words defined in CAMASYST.ASY:

```
INTEGER SCALAR yes

\ Initialize the CAMAC routines
cam.init

\ Determine if Crate 1, Station 20 contains a Model 4012
:check.module \ name the colon definition.
1 set.crate  \ set crate to 1
20 0 set.module \ set station to 20 and subaddress to 0
            \ F(3)A(0) reads the Model 4012 module ID code
3 camac 4012 =
IF 1 yes := ELSE 0 yes := THEN

\ Type message giving answer
yes 1 =
IF " Model 4012 is in Crate 1, Station 20 "
  ELSE " Model 4012 is not at Crate 1, Station 20"
  THEN "TYPE

\ If 4012 installed determine whether it is in local/remote mode
yes 1 =
IF
  8 17 camac \ F(17)A(0) with data = 8 requests read of
          \  local/remote mode
  0 camac    \ F(0)A(0) reads 4012 output register
  DUP 1 =
  IF " and is in Remote mode."   \ Remote Mode message
     DROP
    ELSE  0 =
    IF " and is in Local mode."  \ Local Mode message
      ELSE " and is in error."   \ error message
      THEN
    THEN "TYPE
  THEN
;               \terminate the definition
Check.module     \ call the definition
```

B3. Examples

The Model 9100 diskette includes several programs which serve as examples of CAMAC procedure usage and provide a quick verification of correct operation.

1. CAMTEST

This is a simple interactive program which asks you to enter a dataway Station Number (N), Subaddress (A) and Function code (F) and then executes the requested dataway cycle. If F is a write operation you are prompted to enter write data. If it is a read operation the returned data are printed on the screen.

Crate operations are executed by referencing Station Number 30 where the subaddress selects the function and the function code serves as a parameter as follows:

Subaddress	Operation
0	CRATE (F)
1	CAMCL (F)
2	CAML (data), data is printed

The operation may be repeated as often as desired. The program terminates when you enter Station Number 0.

Three versions of CAMTEST are supplied, one for each of the major languages supported; BASIC, FORTRAN and Pascal.

2. Other Test Programs

CAM6002.BAS - Program for testing the 6001/6002.

CAM5100.BAS - Tests the DMA operation of the 6001 using a DSPT model 5100 dual port memory.

CAM5100C.BAS - Same as CAM5100 but suitable for compiling.

CAM5100.PAS - PASCAL version of CAM5100.

B4. Working With the Source Code

The information in this section is not required in order to use the CAMAC access procedures. If you need to modify the source code and you understand 8086 assembly language this section provides some useful information. Please note that DSPT Technology cannot provide support for specific modifications of the code.

All of the various versions of the CAMAC access procedures are generated from one assembly language source file, CAM6001.ASM. This file has been written for the Microsoft macro assembler, MASM.

CAM6001.ASM contains conditional assembly directives which direct the assembler to assemble code for a specific version. Five symbols are declared for this purpose:

BASIC	For interpretive and compiled BASIC
MSFORT	For Microsoft FORTRAN and Pascal
TURBO	For Turbo Pascal
PROFORT	For IBM Pro FORTRAN
ASYST	For ASYST

To assemble CAM6001 for a specific version, first edit CAM6001.ASM to set the symbol corresponding to the desired version to a non-zero value (such as 1) and all other symbols to zero. Then invoke the assembler with a command of the form

```
MASM CAM6001,CAMACMS,CAMACMS;
```

Note that you should explicitly name the object file output of MASM to conform to the existing naming conventions. The listing file is, of course, optional.

MASM will generate an error message; "Open conditionals: 1" which can be ignored even though the assembler considers it a "Severe error".

The loadable driver versions for interpretive BASIC, Turbo Pascal Version 3, and ASYST require additional steps beyond assembly to generate code in the correct format. The batch file MAKEDRVR.BAT includes the DOS commands necessary to generate either of these versions. The specific form generated is determined by the parameter included with the invocation of MAKEDRVR as follows:

```
MAKEDRVR BASIC      ;generates the BASIC version
MAKEDRVR TURBO      ;generates the Turbo Pascal version
MAKEDRVR ASYST      ;generates the ASYST version
```

Before invoking MAKEDRVR, CAM6001.ASM must be edited to set the conditional assembly symbols appropriately.

B5. Using Interrupts

The CAMAC LAM, "Look-at-me", is the mechanism by which a CAMAC module indicates to the system that it requires service. Fundamentally, there are two ways of dealing with LAMs. The application program can "poll" the LAM by repeatedly executing F(8) to the module until it sees a Q = 1 response. This is usually sufficient for most applications.

Where higher performance or shorter response times are required, the LAM can "interrupt" the computer causing it to immediately execute a service routine for the module and ;then return to what it was doing. Real-time, multitasking operating systems provide extensive facilities for handling such interrupts in an efficient, reliable manner.

PC DOS is not a real-time system and has only rudimentary features for interrupt handling. Consequently, CAM6001 does not support LAMs as interrupts. Successful interrupt programming requires detailed knowledge of the processor and the operating system and should not be undertaken by inexperienced programmers. Unless you are thoroughly familiar with the 8088 (80286) interrupt structure and DOS, you should restrict LAM handling to polling.

The PC004 has the capability to selectively route the LAM signal from each of the four crates to one of the PC interrupt levels IRQ2 to IRQ7 by means of wire jumpers. Alternatively, all four LAMs may be routed to a single interrupt level. In the lower right hand corner of the PC004, just above the PC bus connector, is a jumper field with pads labelled L1 to L4, LT and 12 to 17. L1 to L4 are the LAM signals from crates 1 to 4, LT is the OR of all four LAM signals and 12 to 17 are the PC bus interrupt lines. To connect any LAM to any interrupt level, install a jumper between the corresponding pads.

* Note: If the LT jumper is used, "SW1" switch 8-9 must be closed.

For more information see schematic drawing PC004-01-SD in the back of this manual. From a software standpoint, interrupt handling in DOS is highly language dependent. Microsoft PASCAL for example has an INTERRUPT attribute which permits a procedure to be invoked via an interrupt.

SECTION 6
PS/2 HARDWARE

INSTALLING THE PC-006

In accordance with Microchannel standards, the PC-006 makes use of the Programmable Option Select (POS) feature to configure various operational parameters. The options, and their values, are described in an "Adapter Description File" found on the diskette that came with the PC-006. Before installing the board in your PS/2, you must install its Adapter Description File on an un-
protected working copy of the Reference Diskette that came with your PS/2. Proceed as follows:

1. Boot an unprotected copy of you Reference Diskette.

2. Select "5. Copy an Option Diskette" and follow the instructions.

3. Turn off the computer.

4. Remove the cover and install the PC-006 in an empty adapter slot.

5. Attach the flat cable(s) to the PC-006 with the red stripe(s) up. Route the cable(s) neatly out the back of the chassis.

6. Replace the cover and boot the Reference Diskette modified in step 2.

7. Following the screen with the IBM logo the system informs you that the configuration has changed and asks if you wish to autoconfigure. Reply "yes".

8. The PC-006 is now "installed". Follow the instructions to reboot DOS.

Note that the default values for the PC-006 options are correct for the vast majority of applications. They should be changed only if there is a clear reason to do so and you thoroughly understand what **it is you are changing.**

PC-006 DIAGNOSTIC
PRE-BOOT TROUBLESHOOTING

The PS/2 may fail to boot with a new PC-006 because either:

1. The module ID code is wrong and/or
2. The card is interfering with the Microchannel.

Case 1 results in an error code 165 and usually means the ID PROM, chip 1F, is bad. Try replacing it.

Case 2 may give one of several error codes, including 165, depending on the nature of the interference. This is more difficult but can be approached methodically. The following strategy is recommended:

1. Scope the four INTn lines at 2D-4, 2D-6, 2D-8 and 2D-10. They should all be high. If any of them is low, resolve the problem and try again.

2. Scope /DREQ at 2F-2. It should be high. If not, resolve the problem and try again

3. Replace the data bus transceiver at 4E.

4. Replace the Microchannel controller at 2F.

5. If the problem still persists, examine the board carefully for shorts or opens to the Microchannel connector.

ADAPTER DESCRIPTION FILE
@ID OF MICROCHANNEL (PC-006) 5510.ADF

```
AdapterID 5510H
AdapterName "PC-006: Model 6001/6002 CAMAC Crate Controller Adapter"
NumBytes 4

NamedItem
      Prompt "I/O Address Range"
      choice "240H - 24FH (Standard)" POS[3] = 00XXX100B IO 240H - 24FH
      choice "250H - 25FH" POS[3] = 00XXX101B IO 250H - 25FH
      choice "260H - 26FH" POS[3] = 00XXX110B IO 260H - 26FH
      choice "270H - 27FH" POS[3] = 00XXX111B IO 270H - 27FH
      choice "280H - 28FH" POS[3] = 01XXX000B IO 280H - 28FH
      choice "290H - 29FH" POS[3] = 01XXX001B IO 290H - 29FH
      choice "2A0H - 2AFH" POS[3] = 01XXX010B IO 2A0H - 2AFH
      choice "2B0H - 2BFH" POS[3] = 01XXX011B IO 2B0H - 2BFH
      choice "2C0H - 2CFH" POS[3] = 01XXX100B IO 2C0H - 2CFH
      choice "2D0H - 2DFH" POS[3] = 01XXX101B IO 2D0H - 2DFH

Help
"The PC-006 occupies a block of 16 I/O addresses.  The standard assignment is
240H to 24FH and represents crates 1 to 4.  If you have more than four crates in
your system, you must select an alternate address range for the additional PC-
006s.  250H to 25FH is suggested for the second adapter, 260H to 26FH for the
third adapter, etc.  You must also change the address range if POST reports any
conflicts.  Use F5=Previous and F6=Next keys to change this assignment in the
'Change configuration' window."

NamedItem
      Prompt "LAM Interrupts"
      choice "off" POS[2] = X0XXXXXXB
      choice "on"  POS[2] = X1XXXXXXB

Help
"Interrupts are normally 'off' meaning that LAMs must be detected by 'polling'.
If you wish to allow LAMs to interrupt the processor, use the F5=Previous or
F6=Next keys to select 'on'."

NamedItem
      Prompt "Crate 1 Interrupt Level"
      choice "INT 9"    POS[4] = XXXXXX00B      INT 9
      choice "INT 10"   POS[4] = XXXXXX01B      INT 10
      choice "INT 11"   POS[4] = XXXXXX10B      INT 11
      choice "INT 15"   POS[4] = XXXXXX11B      INT 15

Help
"Each crate may be independently assigned to one of four interrupt levels.
This allows all crates to be assigned to the same level, each crate assigned to
an independent level, or any combination in between.  Interrupt level is only
significant if LAM Interrupts have been set to 'on'. Use the F5=Previous and
F6=Next keys to select the desired level."

NamedItem
      Prompt "Crate 2 Interrupt Level"
      choice "INT 9"    POS[4] = XXXX00XXB      INT 9
      choice "INT 10"   POS[4] = XXXX01XXB      INT 10
      choice "INT 11"   POS[4] = XXXX10XXB      INT 11
      choice "INT 15"   POS[4] = XXXX11XXB      INT 15
```

```
Help - Default is 1st Item in List

Each crate may be independently assigned to one of four interrupt levels. This
allows all crates to be assigned to the same level, each crate assigned to an
independent level, or any combination in between.  Interrupt level is only
significant if LAM Interrupts have been set to 'on'.  Use the F5=previous and
F6=Next keys to select the desired level."

NamedItem
      Prompt "Crate 3 Interrupt Level"
      choice "INT 9"    POS[4] = XX00XXXXB      INT 9
      choice "INT 10"   POS[4] = XX01XXXXB      INT 10
      choice "INT 11"   POS[4] = XX10XXXXB      INT 11
      choice "INT 15"   POS[4] = XX11XXXXB      INT 15

Help
"Each crate may be independently assigned to one of four interrupt levels. This
allows all crates to be assigned to the same level, each crate assigned to an
independent level, or any combination in between.  Interrupt level is only
significant if LAM Interrupts have been set to 'on'.  Use the F5= Previous and
F6=Next keys to select the desired level."

NamedItem
      Prompt "Crate 4 Interrupt Level"
      choice "INT 9"    POS[4] = 00XXXXXXB      INT 9
      choice "INT 10"   POS[4] = 01XXXXXXB      INT 10
      choice "INT 11"   POS[4] = 10XXXXXXB      INT 11
      choice "INT 15"   POS[4] = 11XXXXXXB      INT 15

Help
"Each crate may be independently assigned to one of four interrupt levels. This
allows all crates to be assigned to the same level, each crate assigned to an
independent level, or any combination in between.  Interrupt level is only
significant if LAM Interrupts have been set to 'on'.  Use the F5=previous and
F6=Next keys to select the desired level."

NamedItem
      Prompt "DMA Arbitration Level"
      choice "Level 1"  POS[5] = XXXX0001B      ARB 1
      choice "Level 2"  POS[5] = XXXX0010B      ARB 2
      choice "Level 3"  POS[5] = XXXX0011B      ARB 3
      choice "Level 4"  POS[5] = XXXX0100B      ARB 4
      choice "Level 5"  POS[5] = XXXX0101B      ARB 5
      choice "Level 6"  POS[5] = XXXX0110B      ARB 6
      choice "Level 7"  POS[5] = XXXX0111B      ARB 7
      choice "Level 0"  POS[5] = XXXX0000B      ARB 0

Help
"This option sets the arbitration level for DMA operations.  It should only be
changed if there is an unavoidable conflict with another DMA device.  Use
F5=Previous and F6=Next keys to change the level."

NamedItem
      Prompt "DMA Burst Mode"
      choice "on"       POS[2] = 1XXXXXXXB
      choice "off"      POS[2] = 0XXXXXXXB

Help
"Burst mode is normally turned 'on' for improved transfer rates.  There is a
remote possibility that some memory boards may not work correctly in burst mode.
Therefore, if you experience DMA problems, try turning burst mod 'off'."
```

```
NameItem
      Prompt "Fairness"
      choice "on"       POS[5] = XXX1XXXXB
      choice "off"      POS[5] = XXX0XXXXB

Help
"'Fairness' assures that all devices competing (arbitrating) for the bus get a
fair share of bus bandwidth.  In certain critical applications it may be
necessary to let CAMAC 'hog' the bus by turning fairness 'off'."
```

SECTION 7

FUNCTIONAL DESCRIPTION

A discussion of the functional operation of each of the boards covered by this manual follows. Refer to the attached schematics for clarification or additional information.

A) 6001-1

The Models 6001 and 6002 use identical -1 boards so there is no distinction made between them in the following discussion.

1) Sheet 1 -

I.C.'s 1C,1B, and 1A latch data from the CPU for CAMAC write operations. I.C.'s 2C,2B, and 2A drive the write data onto the dataway. I.C.'s 3C, 3B, and 3A latch the write data during the CAMAC cycle and drive the proper LED's on the dataway display. These latches are enabled only if an F16 or greater command was executed.

2) Sheet 2 -

I.C.'s 5C, 5B, and 5A latch CAMAC read data for CPU readout. I.C.'s 4C, 4B, and 4A latch the data for LED display. These latches are enabled only if an F0-F7 command was executed.

3) Sheet 3 -

I.C.'s 2D, 1D, and 3D latch CAMAC signals and drive the display LED'S.

B) 6002-2

The Model 6002 differs from the 6001 in that the former supports the auxiliary controller protocol (IEEE Std 675-1982). Otherwise, the operations of the -2 boards are identical. Operation of the 6001 may be inferred from the following discussion by assuming that "GRANT-IN" is always true and ignoring references to auxiliary mode operation and to the auxiliary control bus signals.

1) Sheet 1 -

I.C.'s 6G and 7G address the data registers to read or write data. 6G strobes data into the write registers and 7G enables register outputs for reading. CAMAC cycles are initiated by 6G(9) or the signal CC(J4,pin35) which set 6F(5) high. This generates a request for a CAMAC cycle, 3G(8), which for a master crate controller is tied to 'GRANT-IN', 4G(5). The 1GRANT-IN'signal and requested CAMAC cycle go through a decision arbitration circuit, 1F. If both are true, 1F(2&7), then a CAMAC cycle is initiated. If 'GRANT-IN' is true, but there is no requested CAMAC cycle, a 'GRANT-OUT' is passed to the next auxiliary. Once a CAMAC cycle is initiated the 'REQUEST INHIBIT' line on the auxiliary bus is turned on. If the D3 bit in the status register is set, then the 'REQUEST INHIBIT' will remain on indefinitely; otherwise,it will be reset when the CAMAC cycle is completed.

The CAMAC cycle on the dataway starts when 8D(5) goes high for 1.1 microsec. The 6002 inserts an extra 100 nsec. at the beginning of the dataway cycle to allow for extra propagation delays in decoding the N lines over the auxiliary bus. The 1.1 microsec. cycle is timed by a crystal oscillator and shift register to give precise timing for the S1 and S2 pulses. The sequence is terminated when 6C(2) goes high.

2) Sheet 2 -

The sheet contains the data latches for the N, F, A, I, Z, and C data. These lines are gated onto the dataway through LS642 drivers. The encoded N lines, however, are gated onto the auxiliary bus for decoding by the master crate controller. If the 6002 is setup as a master (all resistor packs installed) the 74156 decoders are enabled onto the dataway.

3) Sheet 3 -

This sheet contains the LAM encoding circuits. For master crate controllers the LAM's are detected from the dataway and both transmitted over the auxiliary bus and priority encoded. For auxiliary units the LAM's are received over the auxiliary bus and encoded for readout.

C) PC001 - APPLE II Interface Card

I.C. ID acts as a buffer for the address and control lines from the APPLE bus to the 6001/6002. I.C. 2D is a bi-directional receiver passing 8 bits from computer bus to the selected 6001/6002 register for write operations and vice-versa for read operations.

D) PC004 - IBM PC Interface Card with DMA

The PC004 has two logic sections, one for programmed I/O and one for hardware DMA. For both modes, I.C.'s 17A,18A,18B,17B,15B,12B,10B,7B, AND 5B are used to transfer data and commands to the 6001/6002 crate controllers.

1) Programmed I/O

I.C.'s 14A and 13A decode the upper six (of ten) bits for the PC004 I/O address. The lower four bits are used to address the 6001/6002 registers and the PC004 register (16 registers, maximum). The four tri-stated gates at 12A are enabled except during DMA transfers. There are two registers on the PC004, one is write-only and the other is read-only; both have address 15 (hex F).

The write-only register controls:

D6	=	1	starts a DMA transfer
D4,D5	=	0	crate 0 (connector J1) is active
		1	crate 1 (connector J2) is active
		2	crate 2 (connector J3) is active
		3	crate 3 (connector J4) is active
D3	=	0	then do DMA read (from CAMAC to computer)
		1	then do DMA write (from computer to CAMAC)
D2	=	0	then do DMA Block transfer mode
		1	then do DMA Q-Stop transfer mode
D0,D1	=	0	then transfer one byte per CAMAC cycle
		1	then transfer two bytes per CAMAC cycle
		2	then transfer three bytes per CAMAC cycle

The read-only register provides the following information:

D0	=	0	then DMA not completed
		1	then DMA completed

2) DMA Transfers

Once the PC004 register is loaded for the requested DMA transfer, the cycle is initiated by setting D6 of register 15 high. Pin 9 at I.C. 9A then goes high, loading zero into counter 10A, forcing 11A(6) high, and requesting a DMA transfer at 1A(6) (DRQ). The DMA controller chip responds with DACK which disables the gates of 12A and enables PROM 4A to sequence.

PROM 4A initiates CAMAC cycles, reads or writes the correct 6001/6002 registers and controls the length of the I/O channel ready signal. The cycle time of the DMA transfer must be extended for some parts of the cycle since the 6002 takes 1.1 microsec. to complete a CAMAC cycle.

Counter 5A controls the number of bytes to transfer. Counter 10A counts 16 read/write strobes and then releases the bus for a refresh cycle. Immediately after refresh the DMA is reinitiated.

DMA is complete after T/C goes high from the DMA controller ship (Block mode) or Q goes low from the 6001/6002 (Q-Stop mode).

Appendix A
SOFTWARE/DOCUMENTATION PROBLEM REPORT FORM

Name_____ Date_____
Company_____
Address_____ Phone #_____
City_____ State____ ZIP_____

INSTRUCTIONS: Use this form to report suspected software bugs, documentation errors or omissions, or suggested enhancements. Attach any available listings or screen dumps that identify or support the suspected problem. Forward the material to DSPT.

PROBLEM CATEGORY
Software Problem _____ Documentation Problem _____ Enhancement _____
(document name _____)

HARDWARE ENVIRONMENT:
PSP/System 1 _____
PSP/System 2 _____
PSP/System 3 _____
Alternative System_____
Manufact._____
Model_____
Installed RAM _____kbytes
Options:
(manufact.,model,options)
Add-in Cards_____
&Peripherals_____
_____
_____
_____
_____

SOFTWARE ENVIRONMENT:
Program Name_____
Module Name_____
Procedure
/Subroutine Name_____
Operating System:
Supplier_____Vers_____
Other resident software:
Supplier_____Vers_____
Supplier_____Vers_____
Supplier_____Vers_____
Supplier_____Vers_____

Description of Problem:_____

Problem Report Vers 1.01 4/15/86
DSPT Technology Inc., 48500 Kato Rd, Fremont, CA 94538

Appendix B
History of PSP9100 Software Releases

Release 7.0 07/01/88

Added support for ASYST. Revised manual.

Release 6.0 02/02/88

Turbo Pascal Version 4 treats external assembly language subroutines differently from version 3. There are now two include files for Turbo to accommodate this difference; CAMTURBO.V3 for version 3 and CAMTURBO.V4 for version 4. The file CAMTURBO.OBJ must be in the current directory. It is not necessary to generate the loadable driver CAMTURBO.BIN.

Note to users of Microsoft Fortran Version 4.0: You must use either the Medium or Large models because CAMACMS.OBJ uses FAR calls and passes parameters by FAR reference.

Release 5.4 11/11/87

Alternate versions of object files provided to support both block mode and single cycle DMA transfers. See 6001 manual section 5B for details.

Release 5.3 10/2/87

Symbols have been defined for the DMA chip registers to make it easier to change channels and to use the Block mode. See the source code for details.

RELEASE 5 NOTES 7/28/86

This is a major upgrade of the Model 9100 CAMAC access procedures for the Model 6001/6002 Crate Controller and PC-004 interface board to the IBM PC and compatibles. If you are a new user of this package, the following notes are not of interest. All the information you need is in the 6001/6002 Users Manual.

If you have received this disk as an upgrade, you should also receive new documentation which replaces Section 5B, "Software", of the 6001/6002 Users Manual. The principal difference for existing users is that several of the files have changed names as follows:

CAM6001.ASM	Source code. Formerly CAMV4.ASM
CAMBASIC.BIN	Loadable driver for BASIC. Formerly CAMIO.BIN
CAMBASIC.OBJ	Linkable code for compiled BASIC. Formerly CAMIO.OBJ
CAMACMS.OBJ	Linkable code for Microsoft PASCAL and FORTRAN. Formerly CAMV4.OBJ

New files have been added, mainly for the support of additional programming languages.

CAMTURBO.BIN	Loadable driver for Turbo Pascal.
CAMTURBO.INC	External procedure declarations for Turbo Pascal.
CAMACPRO.OBJ	Linkable code for IBM Pro FORTRAN.
CAMACINC.BAS	External procedure declarations for interpretive BASIC.
CAMTEST.PAS	
CAMTEST.FOR	
CAMTEST.BAS	Simple test program for exercising the procedures.

September 10,1981

APPENDIX - CAMAC

STATION NUMBER N

Each normal station is addressed by a signal on an individual Station Number line N which comes from a separate contact at the control station. The stations are numbered in decimal from the left-hand end as viewed from the front, beginning with Station 1 (addressed by N1).

SUBADDRESS A8, A4, A2, A1

Different sections of a module are addressed by signals on the four A bus lines. These signals are decoded in the module to select one of up to sixteen subaddresses, numbered in decimal from A(0) to A(15).

The subaddress may be used to select, for example, a register within the module, or a feature that is to control the Response signal Q, or a section of the module that is to be operated on by functions such as Enable, Disable, and Execute.

The subaddress codes are designated A(0), A(1), A(2), A(3), etc., to distinguish them from the individual Subaddress lines A1, A2, A4, and A8. For example, the Subaddress signals A1 = 1, A2 = 1, A4 = 0, and A8 = 0 represent the code A(3).

FUNCTION F16, F8, F4, F2, F1

The function to be performed at the specified subaddress in the selected module or modules is defined by the signals on the five F bus lines. These signals are decoded in the module to select one of up to thirty-two functions, numbered in decimal from F(0) to F(31). The definitions of the thirty-two function codes are summarised in Table 1.

In some systems the controller partially decodes the Function signals in order to determine whether a data transfer is required to a module (writing) or from a module (reading).

The function codes are subdivided into three groups, involving read operations, write operations, and operations with no transfer of data. The standard function codes have defined actions in modules and controllers. There are also reserved codes, for any future additions to the standard codes, and nonstandard codes whose use is not defined in detail.

The function codes are designated F(0), F(1), F(2), F(3), etc., to distinguish them from the individual function lines F1, F2, etc. For example, the Function signals F1 = 1, F2 = 0, F4 = 0, F8 = 1, and F16 = 1 represent the code F(25).

STROBE SIGNALS S1 AND S2.

During each command operation the controller generates two Strobe signals S1 and S2 in sequence on separate bus lines. In response to these timing signals plug-in units initiate various actions appropriate to the command that is present on the Dataway.

Both Strobe signals must be generated during each command operation.

Plug-in units must not take irreversible action based on the command or data signals until the time of S1. Actions concerned with the acceptance of data and status information from the R, W, Q, and X lines must be initiated at the time of S1 (with the possible exception given below). Other actions may also be timed by S1, but must not change the state of signals on the R and W lines.

Any actions that can change the state of Dataway Read or Write signals must be initiated by the second strobe S2. For example, S2 must be used if it is required to clear a register whose output is connected to the Dataway.

DATA

All information carried by the Read and Write lines is conveniently described as data, although it may be information concerned with status or control features in modules. Information that is transferred to or from a control register in a module is thus regarded as data.

Up to 24 b may be transferred in parallel between the controller and the selected module. Independent lines are provided for the read and write directions of transfer.

WRITE LINES W1-W24

The controller generates data signals on the W bus lines during each write operation. The W signals must reach a steady state before S1 and must be maintained until the end of the operation, unless modified by S2. Strobe S1 must be used by the modules to strobe the data unless there are very strong technical reasons for choosing S2.

READ LINES R1-R24

Data signals are set up on the R bus lines by the module during a read operation. The R signals must reach a steady state before S1 and must be maintained for the full duration of the Dataway operation, unless the state of the data source is changed by S2. The controller must initiate action concerned with the acceptance of data from the R lines at the time of the Strobe S1 and must not take irreversible action before this.

STATUS INFORMATION

Status information is conveyed by signals on the Look-at-Me L, Busy B, Response Q, and Command Accepted X lines.

LOOK-AT-ME L

The Look-at-Me lines, like the N lines, are individual connections from each normal station to separate contacts at the control station (L1 from station 1, etc.).

Any module may generate a signal on its individual line L, to indicate that it requires attention. Modules that occupy more than one station may indicate different demands by signals on the appropriate L lines.

The L signal generated by a module may represent demands for attention originating from more than one Look-at-Me source (LAM source) in the module.

Individual bits of the LAM status register are set by the corresponding LAM sources, and are cleared by appropriate commands and by the initialize signal. The internal Look-at-Me signal (L signal), derived from the OR combination of the LAM requests, is tested by the command Test LAM, with subaddress A(k) distinguishing this from tests of individual LAM requests. Finally, the output of this signal as the Dataway L signal is inhibited while the module is addressed by N, possibly in conjunction with certain F and A codes or groups of codes.

LOOK-AT-ME: Clear, Disable, and Test.

Provision must be made for resetting each bit of the LAM status register individually, either by a Clear Look-at-Me operation F(10), or by a Selective Clear Group 2 operation F(23). All LAM status bits must be reset collectively by the Initialize signal.

If the LAM request calls for some specific action (for example, reading the contents of a data register), then the corresponding bit of the LAM status register should also be cleared when the appropriate Dataway operation is performed.

A module that has generated L = 1 must not clear the LAM status register until it receives an appropriate command or the Initialize signal.

Each module that generates L should have a means of enabling and disabling the LAM requests. This may be done by loading and clearing a mask register, or by Enable and Disable commands.

All LAM requests that can be disabled by commands must also be disabled by the Initialize signal Z.

A module that generates L must have a means of testing the L signal by a Test Look-at-Me operation (Function code F(8), with a subaddress distinguishing this from tests of individual LAM requests). If there are several LAM sources, the corresponding LAM requests must also be

capable of being examined either by Test Look-at-Me operations associated with appropriate subaddresses or by reading a LAM request pattern in a read operation.

BUSY B

The Busy signal is used to interlock various aspects of a system that can compete for the use of the Dataway. The signal B = 1 indicates to all units that a Dataway operation is in progress.

The Busy signal B = 1 must be generated during each Dataway command operation (when N signals are also generated).

RESPONSE Q

During every command operation the addressed module may generate a signal on the Q bus line to indicate the status of any selected feature of the module.

The controller must initiate action concerned with the acceptance of the status information from the Q line at the time of Strobe S1 and must not take irreversible action before this.

In read and write operations addressed modules must establish the signal Q = 0 or Q = 1 before Strobe S1 and must maintain it until at least Strobe S2.

In Test Look-at-Me operations addressed modules may initiate the Q signal at any time during a Dataway operation if the status of the appropriate LAM request changes. If Q = 1 has been initiated, it will remain static until the end of the operation since this command is not allowed to reset LAM status.

In all operations other than read, write, and Test Look-at-Me the Q signal is permitted to change at any time. There is a risk that status information will be missed if Q = 1 is initiated between Strobes S1 and S2 during an operation in which the module resets the status condition at the time of Strobe S2.

In any operation the Q signal conveys only one bit of information, which should be clearly defined for each subaddress and function code used by the module.

COMMAND ACCEPTED X

Whenever a module is addressed during a command operation it must generate X = 1 on the Command Accepted bus line if it recognizes the command as one that it is equipped to perform, either within the module or in association with external equipment. The signal on the X line must reach a steady state before S1 and must be maintained until S2. Controllers that accept the X signal must do so at the time of S1.

The signal X = 0 should indicate a serious malfunction, for example, a module that is not present, not powered, lacks external connections, or is not equipped to perform the required action. The action taken by the controller in response to X = 0 may, for example, be to call for intervention by the operator or operating system.

COMMON CONTROLS Z, C, I.

During unaddressed Dataway operations either Initialize Z or Clear C is generated by the controller and received by each unit connected to the appropriate bus line.

The common control signal Inhibit I is not associated with Dataway operations. It may be generated at any time and is received by each unit connected to the I bus line.

INITIALIZE Z

The Initialize signal is intended to be used during system startup.

Initialize must have absolute priority over other signals. In response to Z = 1 all data and control registers must be set to a defined initial state, all LAM status registers must be reset, and, if possible, all LAM requests must be disabled.

INHIBIT I

The signal I = 1 must inhibit any feature to which it is connected in a module.

The designer is free to choose which activities (for example, data taking) within a module are inhibited by this signal. The signal may be gated or routed within the module, for example, as a result of a previous command or by patch wiring.

When any unit generated the Initialize signal Z = 1 it must also generate I = 1.

CLEAR C

The signal C = 1 must clear all registers and bistables to which it is connected.

Units that generate C must also initiate a sequence including Busy and Strobe S2.

The designer is free to choose which registers and bistables are cleared in response to C·S2. The signal may be gated or routed within the module, for example as a result of a previous command or by patch wiring.

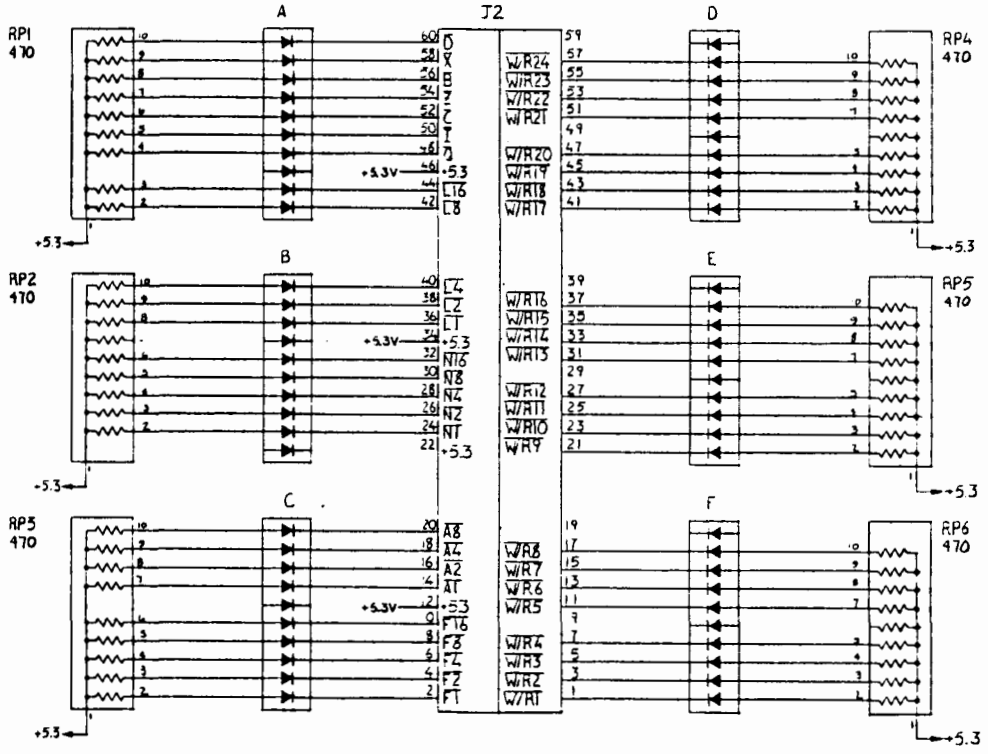
CONTACT ALLOCATION AT A NORMAL STATION
(Viewed from front of Crate)

Bus line	Free Bus line	P1	B	Busy	Bus line
Bus line	Free Bus line	P2	F16	Function	Bus line
Individual patch contact		P3	F8	Function	Bus line
Individual patch contact		P4	F4	Function	Bus line
Individual patch contact		P5	F2	Function	Bus line
Bus line	Command Accepted	X	F1	Function	Bus line
Bus line	Inhibit	1	A8	Subaddress	Bus line
Bus line	Clear	C	A4	Subaddress	Bus line
Individual line	Station Number	N	A2	Subaddress	Bus line
Individual line	Look-at-Me	L	A1	Subaddress	Bus line
Bus line	Strobe 1	S1	Z	Initialize	Bus line
Bus line	Strobe 2	S2	Q	Response	Bus line
Twenty-four Write Bus lines W1 = least significant bit W24 = most significant bit		W24	W23		
		W22	W21		
		W20	W19		
		W18	W17		
		W16	W15		
		W14	W13		
		W12	W11		
		W10	W9		
		W8	W7		
		W6	W5		
		W4	W3		
		W2	W1		
Twenty-four Read Bus lines R1 = least significant bit R24 = most significant bit		R24	R23		
		R22	R21		
		R20	R19		
		R18	R17		
		R16	R15		
		R14	R13		
		R12	R11		
		R10	R9		
		R8	R7		
		R6	R5		
		R4	R3		
		R2	R1		
Power Bus lines	-12 V dc	-12	-24	-24 V dc	
	+200 V dc	-200	-6	-6 V dc	
	117 V ac Live	ACL	ACN	117 V ac Neutral	
	Reserved	Y1	E	Clean Earth	
	+12 V dc	+12	+24	+24 V dc	
	Reserved	Y2	+6	+6 V dc	
	0 V (Power Return)	0	0	0 V (Power Return)	

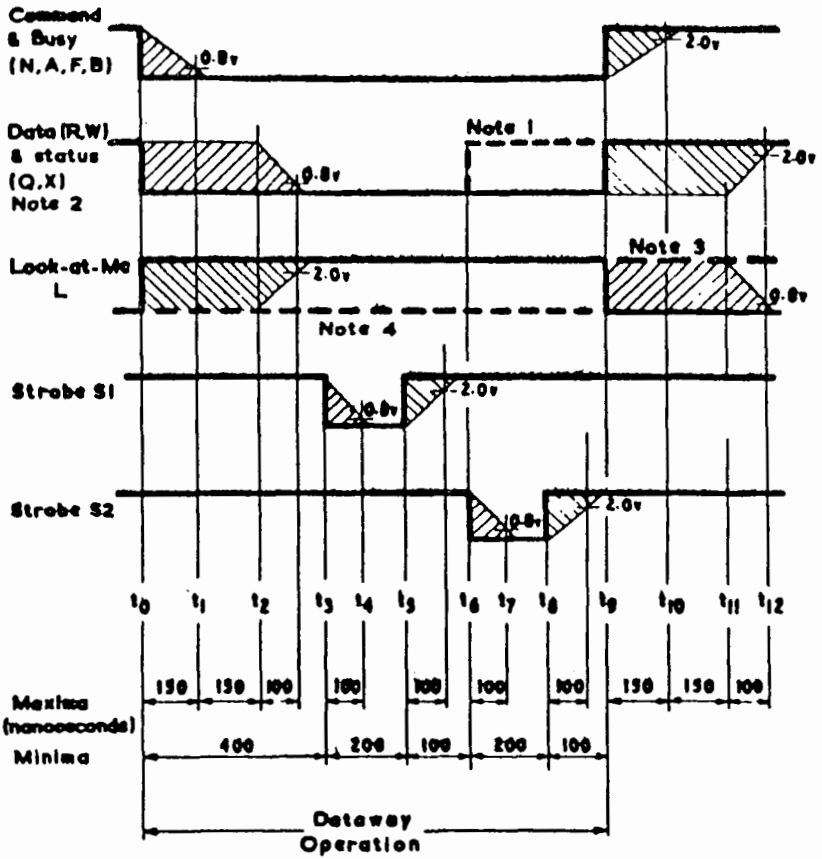
CONTACT ALLOCATION AT THE CONTROL STATION
(Viewed from front of Crate)

Individual patch contact		P1	B	Busy	Bus line
Individual patch contact		P2	F16	Function	Bus line
Individual patch contact		P3	F8	Function	Bus line
Individual patch contact		P4	F4	Function	Bus line
Individual patch contact		P5	F2	Function	Bus line
Bus line	Command Accepted	X	F1	Function	Bus line
Bus line	Inhibit	I	A8	Subaddress	Bus line
Bus line	Clear	C	A4	Subaddress	Bus line
Individual patch contact		P6	A2	Subaddress	Bus line
Individual patch contact		P7	A1	Subaddress	Bus line
Bus line	Strobe 1	S1	Z	Initialize	Bus line
Bus line	Strobe 2	S2	Q	Response	Bus line
Twenty-four individual Look-at-Me lines L1 from Station 1, etc.		L24	N24	Twenty-four individual Station Number lines N1 to Station 1, etc.	
		L23	N23		
		L22	N22		
		L21	N21		
		L20	N20		
		L19	N19		
		L18	N18		
		L17	N17		
		L16	N16		
		L15	N15		
		L14	N14		
		L13	N13		
		L12	N12		
		L11	N11		
		L10	N10		
		L9	N9		
		L8	N8		
		L7	N7		
		L6	N6		
		L5	N5		
		L4	N4		
		L3	N3		
		L2	N2		
		L1	N1		
Power Bus lines	-12 V dc	-12	-24	-24 V dc	
	+200 V dc	+200	-6	-6 V dc	
	117 V ac Live	ACL	ACN	117 V ac Neutral	
	Reserved	Y1	E	Clean Earth	
	+12 V dc	+12	+24	+24 V dc	
	Reserved	Y2	+6	+6 V dc	
	0 V (Power Return)	0	0	0 V (Power Return)	

Code F( )	Function	Use of R & W Lines	Function Signals F16	F8	F4	F2	F1	Code F( )
0	Read Group 1 register	Functions using the R lines	0	0	0	0	0	0
1	Read Group 2 register		0	0	0	0	1	1
2	Read & Clear Group 1 register		0	0	0	1	0	2
3	Read Complement of Group 1 register		0	0	0	1	1	3
4	Nonstandard		0	0	1	0	0	4
5	Reserved		0	0	1	0	1	5
6	Nonstandard		0	0	1	1	0	6
7	Reserved		0	0	1	1	1	7
8	Test Look-at-Me	Functions using the R or	0	1	0	0	0	8
9	Clear Group 1 register	W lines	0	1	0	0	1	9
10	Clear Look-at-Me		0	1	0	1	0	10
11	Clear Group 2 register		0	1	0	1	1	11
12	Nonstandard		0	1	1	0	0	12
13	Reserved		0	1	1	0	1	13
14	Nonstandard		0	1	1	1	0	14
15	Reserved		0	1	1	1	1	15
16	Overwrite Group 1 register	Functions using the W lines	1	0	0	0	0	16
17	Overwrite Group 2 register		1	0	0	0	1	17
18	Selective Set Group 1 register		1	0	0	1	0	18
19	Selective Set Group 2 register		1	0	0	1	1	19
20	Nonstandard		1	0	1	0	0	20
21	Selective Clear Group 1 register		1	0	1	0	1	21
22	Nonstandard		1	0	1	1	0	22
23	Selective Clear Group 2 register		1	0	1	1	1	23
24	Disable	Functions not using the R	1	1	0	0	0	24
25	Execute	or W lines	1	1	0	0	1	25
26	Enable		1	1	0	1	0	26
27	Test Status		1	1	0	1	1	27
28	Nonstandard		1	1	1	0	0	28
29	Reserved		1	1	1	0	1	29
30	Nonstandard		1	1	1	1	0	30
31	Reserved		1	1	1	1	1	31



		DSP TECH	
		LED BOARD	
		SCHEMATIC DIAGRAM	
		D	-0001-03-SD B



NOTES:

(1) Data and status may change in response to *S2*

(2) During some operations *Q* may change at any time

(3) LAM status may be reset during operation

(4) *L* signal may be maintained during operation

Timing of a Dataway Command Operation

DSP [redacted]nology Inc.

S I N G L E L E V E L B I L L O F M A T E R I A L R E P O R T

Assembly: PC006 (DMA 1F FOR PS2 MICROCHNL(UNRELEASE))
Revision: A
As of: 11/10/90

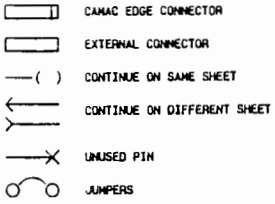
Date: 11/13/90
Page: 1

	Component	Description	Qty Per	As of	Until	Reference
1.	0100-0102	RES: RC07 1K OHM 5% 1/4W	2.00	03/01/90	12/31/99	R1,R2
2.	0100-0472	RES 5% 1/4W CC RESISTOR R=4.7K	4.00	07/27/90	12/31/99	R3-R6
3.	0521-0104	CMC C=0.1UF MONO CENTRALAB CZ20A104	31.00	03/01/90	12/31/99	C1-C31
4.	1750-0006	ICS SN7406	1.00	03/01/90	12/31/99	2D
5.	1760-0000	ICS SN74LS00	1.00	03/01/90	12/31/99	1E
6.	1760-0004	ICS SN74LS04	2.00	03/01/90	12/31/99	1D,4D
7.	1760-0008	ICS SN74LS08	1.00	03/01/90	12/31/99	2G
8.	1760-0020	ICS SN74LS20	1.00	03/01/90	12/31/99	3H
9.	1760-0027	ICS SN74LS27	2.00	03/01/90	12/31/99	3G,4B
10.	1760-0032	ICS SN74LS32	2.00	03/01/90	12/31/99	3B,2H
11.	1760-0074	ICS SN74LS74	2.00	03/01/90	12/31/99	1C,1H
12.	1760-0125	ICS SN74LS125	2.00	03/01/90	12/31/99	2B,4H
13.	1760-0138	ICS SN74LS138	1.00	03/01/90	12/31/99	1B
14.	1760-0151	ICS SN74LS151	1.00	03/10/90	12/31/99	3D
15.	1760-0174	ICS SN74LS174	1.00	03/01/90	12/31/99	2C
16.	1760-0175	ICS SN74LS175	1.00	03/01/90	12/31/99	1G
17.	1760-0193	ICS SN74LS193	1.00	03/01/90	12/31/99	3C
18.	1760-0373	ICS SN74LS373	2.00	03/01/90	12/31/99	3E,4G
19.	1760-0645	ICS SN74LS645	5.00	03/01/90	12/31/99	1A,3A,4E,1J,3J
20.	1851-0017	PRM PROM PROG/F 1850-0001 PC004	1.00	07/01/90	12/31/99	4C
21.	1851-0103	PROM: M/F 1850-0001 PC006 1F	1.00	07/27/90	12/31/99	1F
22.	1857-0155	PAL: M/F 1856-0011, 5005 2S	0.00	10/10/90	12/31/99	2E
23.	1857-0156	PAL: M/F 1856-0011 PC006 2E	1.00	10/10/90	12/31/99	
24.	1951-0612	ICS: INTERFACE CHIPS 82C612 PLC	1.00	03/01/90	12/31/99	2F
25.	5000-0001	CONN: IC CKT, PLCC 68PIN 821574-2	1.00	07/01/90	12/31/99	2F
26.	5000-0016	SKT CAMBION 703-1316-01-04-10	1.00	07/01/90	12/31/99	1F
27.	5000-0020	SKT CAMBION 703-1320-01-04-10	1.00	07/01/90	12/31/99	2E
28.	5502-0040	CONN: 40 PIN RA L&E T&B 500-4007E	2.00	03/01/90	12/31/99	J1,J2
29.	7010-0032	PCB: PC006 MICROCHANNEL	1.00	03/01/90	12/31/99	PCB
30.	7660-0002	HDW: REAR PNL MODIFIED PS2	1.00	07/01/90	12/31/99	
31.	7660-0003	HDW: FRONT EJECTOR PS2	1.00	07/01/90	12/31/99	
32.	7660-0004	HDW: REAR EJECTOR PS2	1.00	07/01/90	12/31/99	
33.	PSP9100-PS2	CAM6001 SOFTWARE, FOR PC006	1.00	07/12/90	12/31/99	

	Component	Description	Qty Per	As of	Until	Reference

Assembly: PC006 (DMA 1F FOR PS2 MICROCHNL(UNRELEASE))

SYMBOLS.



NOTES: (UNLESS OTHERWISE SPECIFIED)

1. ALL CAPACITORS ARE IN uf.
2. ALL RESISTORS ARE 1/4W 5% AND IN OHMS.

REV	DESCRIPTION	DATE
	REVISIONS	

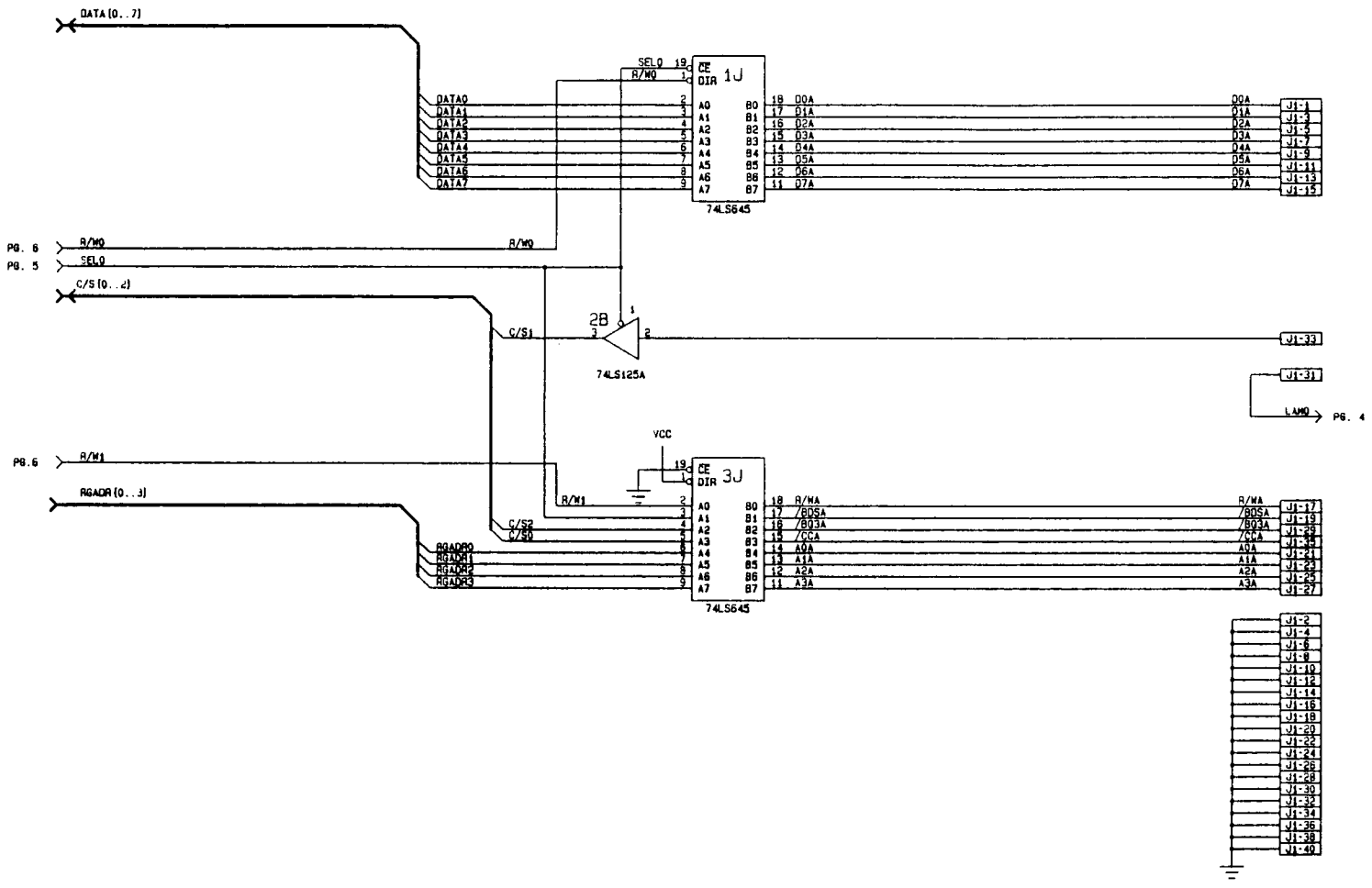
SIGNATURES	DATE
DRAWN N. PERALTA	3/90
CHECKED	
APPROVED	
ISSUED	

D S P TECHNOLOGY INC

SCHEMATIC, MICROCHANNEL

SIZE	FSCM NO.	DRAWING NO.	REVISION
B		PC006	A

SCALE NONE SHEET 1 OF 6



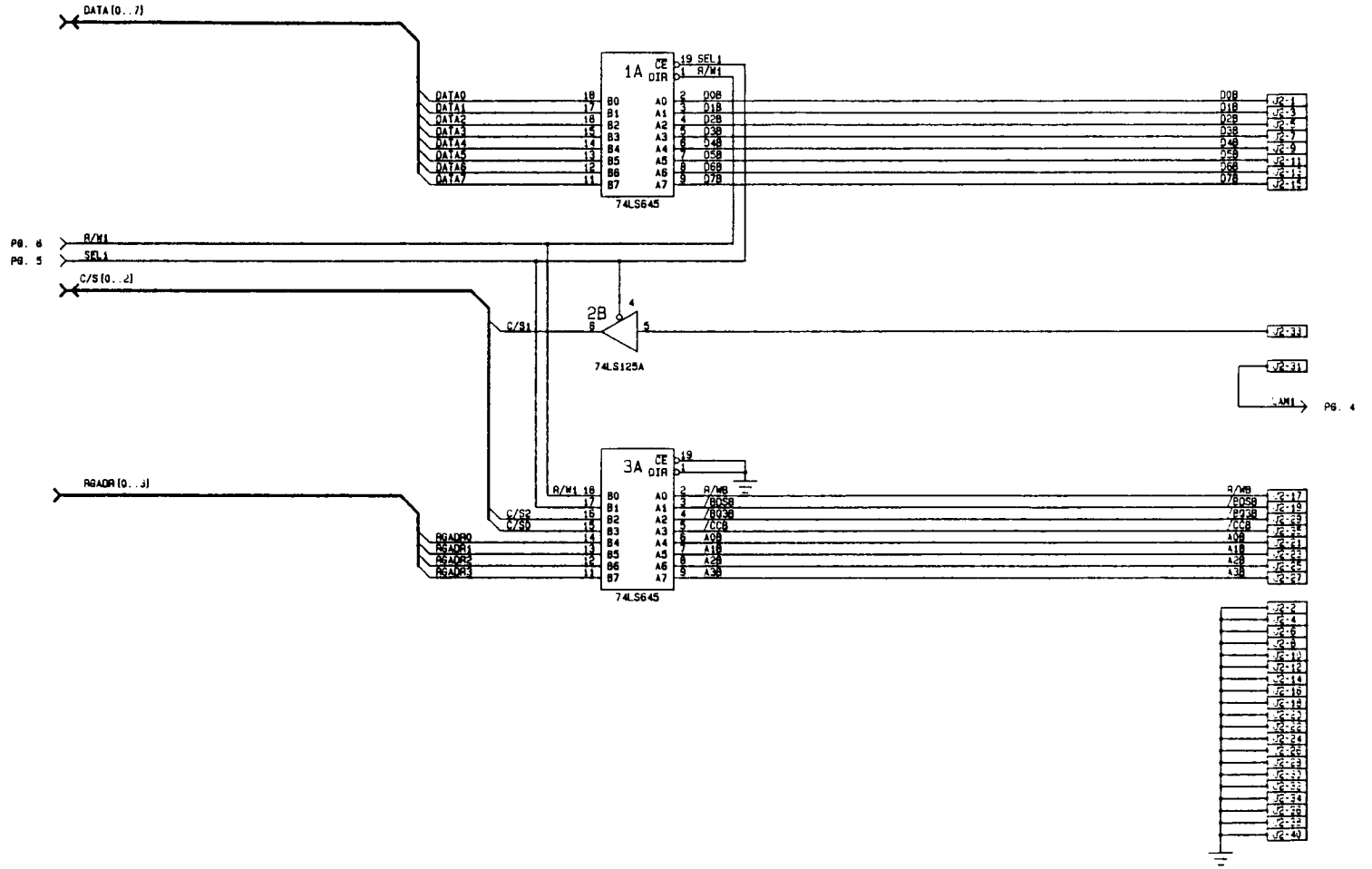
SIGNATURES	DATE
DRAWN N.PERALTA	3/90
CHECKED	
APPROVED	
ISSUED	

D S P TECHNOLOGY INC

SCHEMATIC, MICROCHANNEL
CHANNEL A CRATE INTERFACE

SIZE	DRAWING NO.	REVISION
B	PC006	A

SCALE NONE P-CAD SHEET 2 OF 6



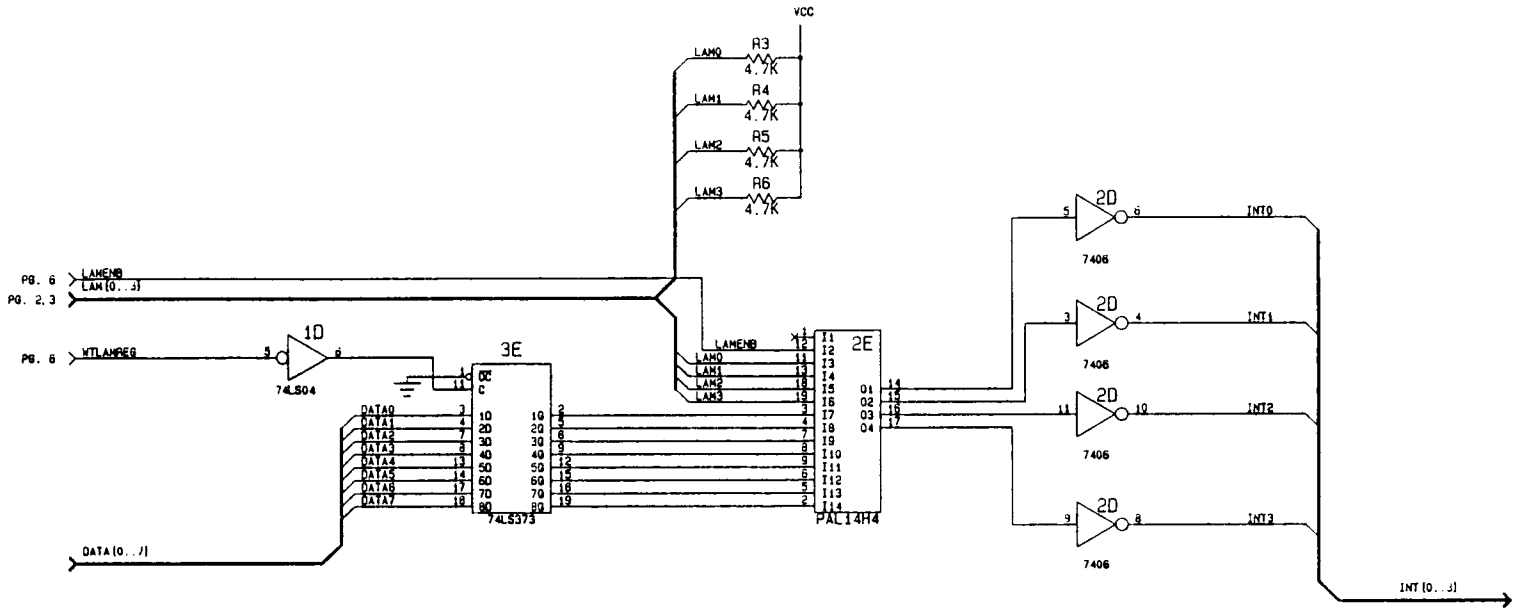
SIGNATURES	DATE
DRAWN N.PERALTA	3/90
CHECKED	
APPROVED	
ISSUED	

D S P TECHNOLOGY INC

SCHEMATIC, MICROCHANNEL
CHANNEL B CRATE INTERFACE

SIZE	DWG NO.	REVISION
B	PC006	A

SCALE NONE P-CAD SHEET 3 OF 6



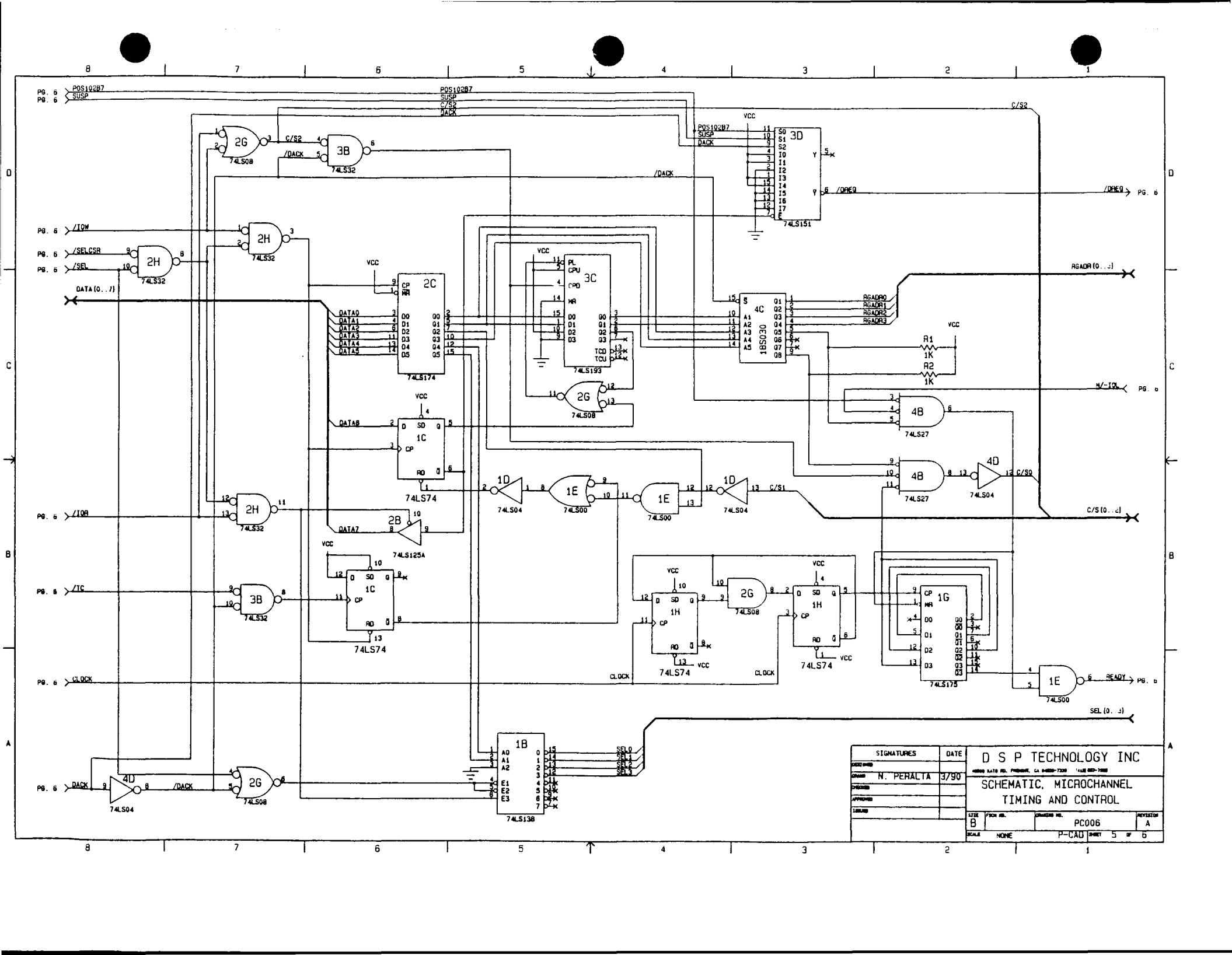
SIGNATURES	DATE
DESIGNED	
DRAWN N. PERALTA	3/90
CHECKED	
APPROVED	
ISSUED	

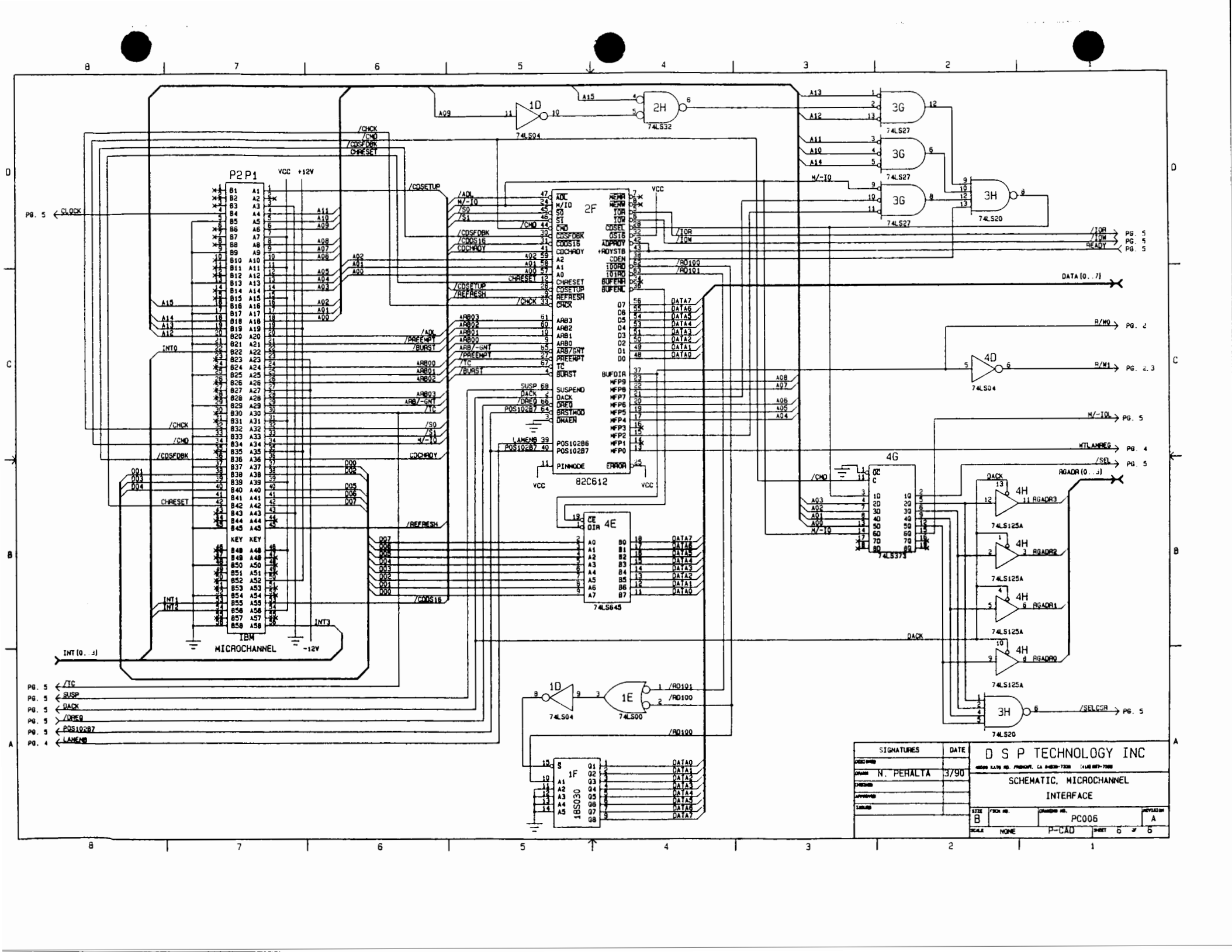
D S P TECHNOLOGY INC

SCHEMATIC. MICRCHANNEL
LAM INTERRUPTS

SIZE	FSCM NO.	DRAWING NO.	REVISION
B		PC006	A

SCALE NONE P-CAD SHEET 4 OF 6


POS102B7
SUSP
C/S2
/DACK
/DREQ
/IOW
/SELCSR
/SEL
DATA(0..7)
/IOR
/TC
CLOCK
DACK
DATA0
DATA1
DATA2
DATA3
DATA4
DATA5
DATA6
DATA7
RGADR(0..3)
RGADR0
RGADR1
RGADR2
RGADR3
R1
1K
R2
1K
M/-IO
C/S1
C/S0
C/S(0..2)
READY
SEL(0..3)
SEL0
SEL1
SEL2
SEL3
VCC
74LS08
74LS32
74LS74
74LS174
74LS193
74LS04
74LS00
74LS27
74LS125A
74LS138
74LS151
74LS175
74LS30
74LS125
PG. 6
PG. 5
SIGNATURES
DATE
N. PERALTA
3/90
D S P TECHNOLOGY INC
SCHEMATIC, MICROCHANNEL
TIMING AND CONTROL
PC006
A
NONE
P-CAD
5
6


P2 P1
IBM MICROCHANNEL
82C612
74LS645
74LS373
74LS125A
74LS04
74LS00
74LS32
74LS27
74LS20
18S030
VCC
+12V
-12V
/CHCK
/CMD
/CDSFDBK
CHRESET
/CDSETUP
/REFRESH
/CDS16
/ADL
M/-IO
/S0
/S1
/CMD
CDCHRDY
/BURST
/PREEMPT
/TC
SUSP
/DACK
/DREQ
POS102B7
LAMEMB
PINMODE
/DMAEN
/SUSPEND
/BRSTWOD
/ERROR
/IOR
/IOW
READY
DATA(0..7)
R/W0
R/W1
M/-IOL
MTLAMBEG
/SEL
AGADR(0..3)
DACK
/SELCSR
/AD100
/AD101
INT(0..3)
CLOCK
SIGNATURES
DATE
N. PEHALTA
3/90
D S P TECHNOLOGY INC
SCHEMATIC, MICROCHANNEL INTERFACE
PC006
B
A
NONE
P-CAD

SINGLE LEVEL

Assembly: A-6001-01 (ASY 6001/6002 DATAWAY BOARD ASY)
Revision: A
As of: 11/10/90

	Component	Description	Qty Per
1.	0216-0511	RPK DIP 15R=510 DALE MDP1601-511J	4.00
2.	0520-0104	CMC 0.1UF CNTLB:CZ20A104M2 KEM:C322	10.00
3.	0600-0685	CAP TANT 6.8UF 35V .250 LEAD SP	2.00
4.	1030-5624	DSR: DIODE RECT 3A GE IN5624/BYM56	1.00
5.	120440	HDW: SCREW 4-40 X 3/16 BHMS ZINC	4.00
6.	1760-0374	ICS SN74LS374	9.00
7.	1760-0534	ICS SN74LS534	6.00
8.	1760-1642	ICS SN74LS642-1	3.00
9.	3100-0402	FUS FUSE 4AMP LITTLEFUSE 251004	1.00
10.	5000-0016	SKT CAMBION 703-1316-01-04-10	4.00
11.	5000-0020	SKT CAMBION 703-1320-01-04-10	2.00
12.	5250-3703	JCK FUS CAMBION 450-3703-01-06-00	2.00
13.	6850-0002	RAIL: GUIDE, HNGD, UPPER, MDL	1.00
14.	6850-0003	RAIL: GUIDE, HNGD, LOWER, MDL	1.00
15.	7000-0009	PCB DWG# 6001-01-FD	1.00
16.	7410-0003	HWD: SWAGE TYPE STANDOFF	6.00
17.	8100-0004	CBL CABLE FLT ASM DWG# 8100-0004 40	1.00
18.	8100-0005	CBL CABLE FLT ASM DWG# 8100-0005 60	1.00

Component Description Qty Per

Assembly: A-6001-01 (ASY 6001/6002 DATAWAY BOARD ASY)

DSP[illegible]hnology Inc.

SINGLE LEVEL B[illegible]L OF MATERIAL REPORT

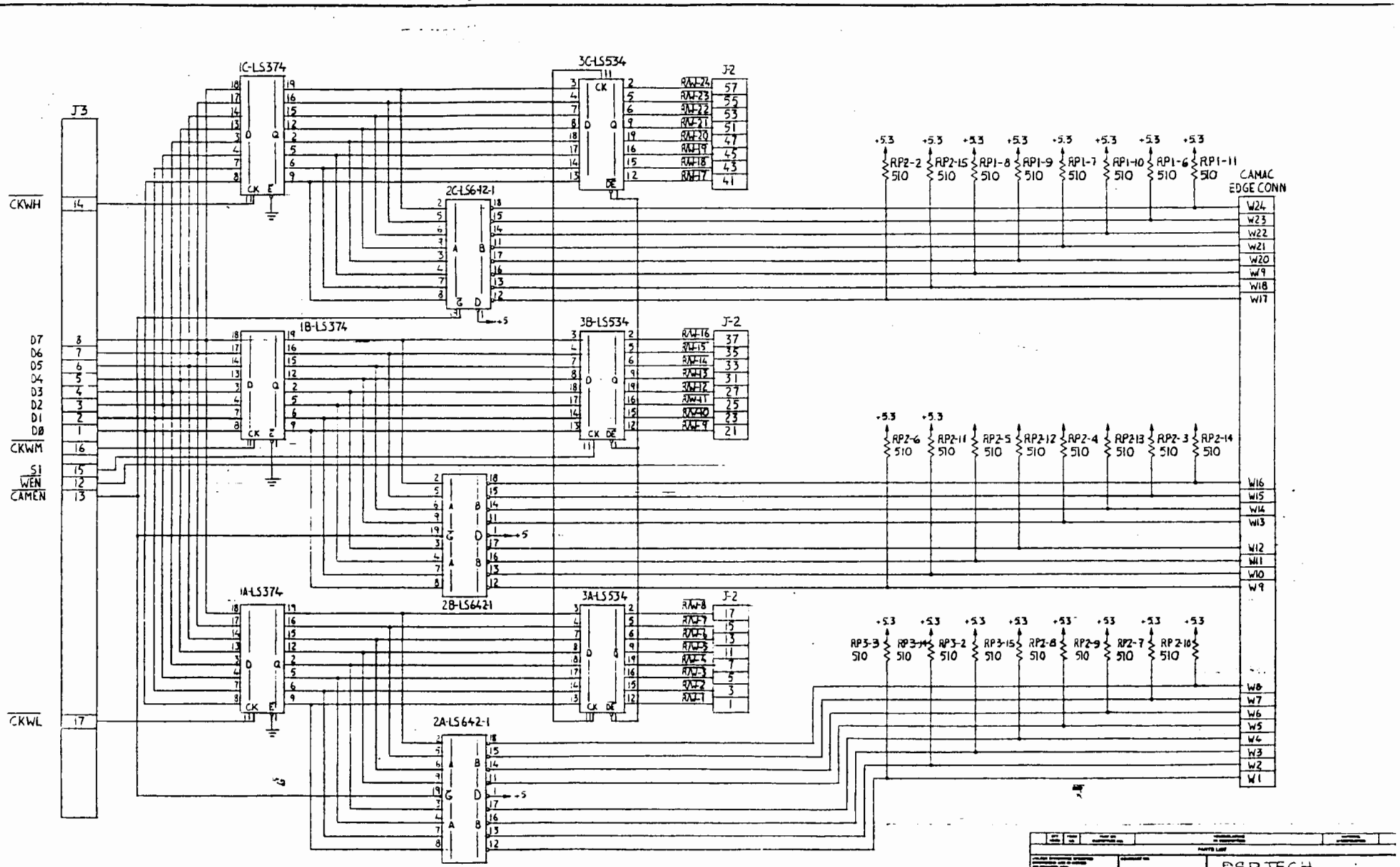
Assembly: A-6001-04 (ASY 6001:FRONT ASY)
Revision:
As of: 11/10/90

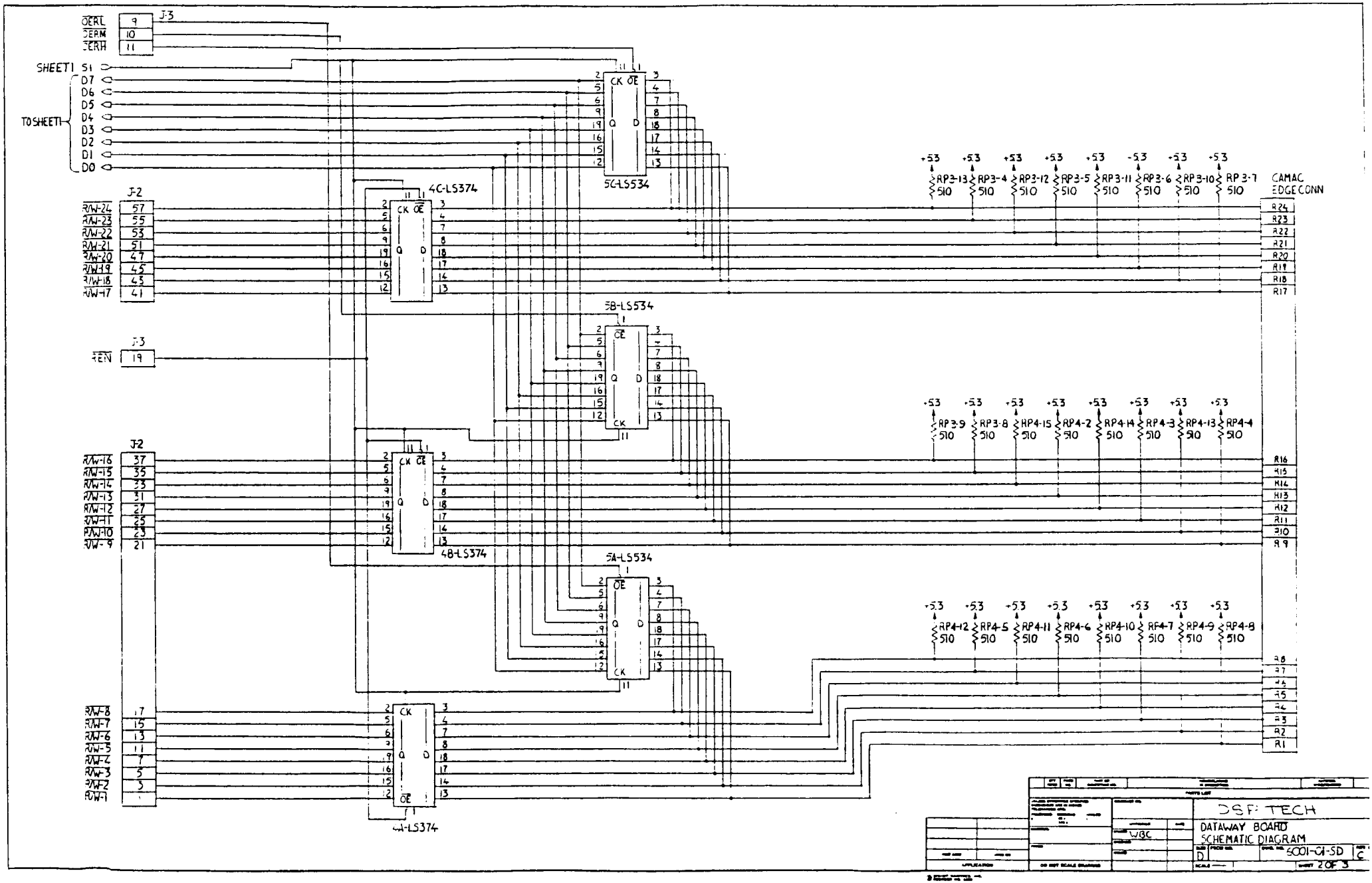
Date: 11/13/90
Page: 1

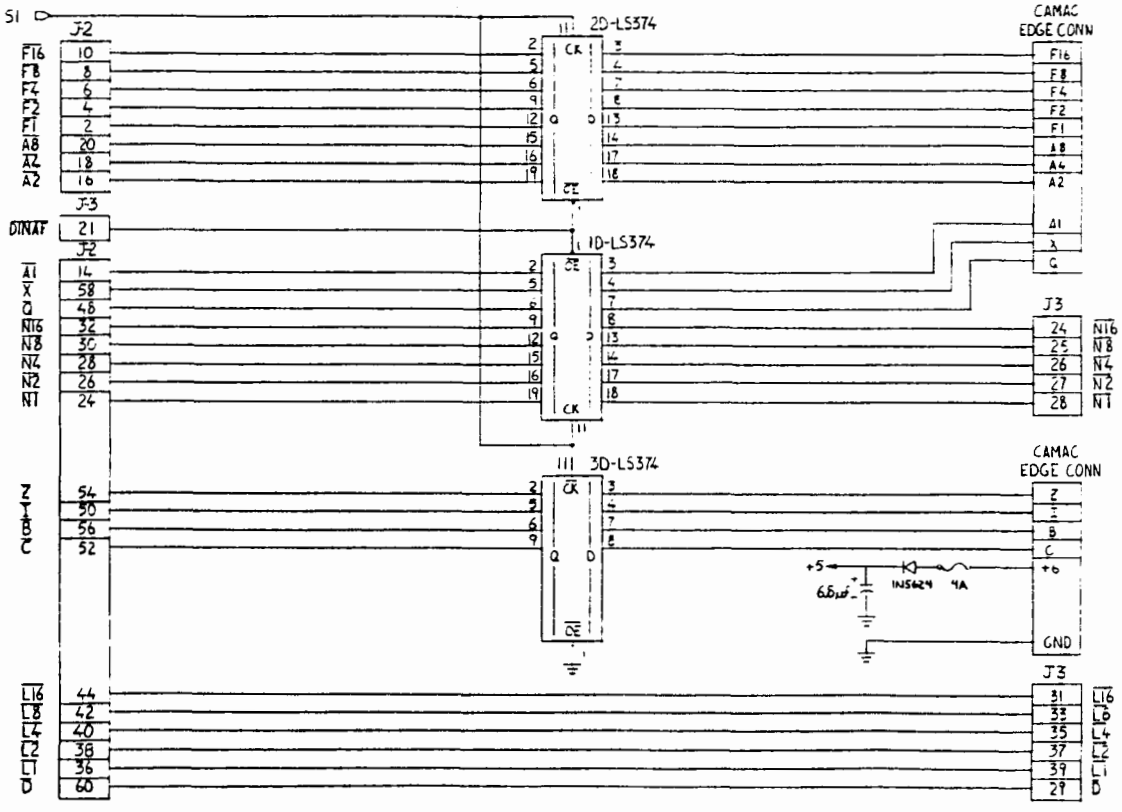
	Component	Description	Qty Per	As of	Until	Reference
1.	120425	HDW: SCREW 4-40 X 5/16 PHIL FLAT SS	8.00	05/01/89	12/31/99	
2.	120440	HDW: SCREW 4-40 X 3/16 BHMS ZINC	4.00	05/01/89	12/31/99	
3.	123043	HDW: #4 EXTERNAL LOCK WASH ZINC	4.00	05/01/89	12/31/99	
4.	123046	HDW: WASHER #4 FLAT NYLON	4.00	05/01/89	12/31/99	
5.	124044	HDW SPACER 4.40 X 7/16 HEX 1/4 THRE	4.00	05/01/89	12/31/99	
6.	6020-0001	SW:SPDT (ON)-OFF-(ON) MOM C&K 7105	1.00	12/09/85	12/31/99	
7.	6020-0002	CAP FOR 6020-0001 ALCO C-10-1-25	1.00	12/09/85	12/31/99	
8.	6020-0003	NUT DRESS ALCO N-40 CTLR-HMMR WNU-	1.00	12/09/85	12/31/99	
9.	6851-0010	PNL: FR, 6001	1.00	12/09/85	12/31/99	
10.	6852-0001	SCR:FT PNL JCK DEUTCH D7500M4B4	1.00	12/09/85	12/31/99	
11.	A-6001-03	ASY 6001:FRONT PANEL PCB ASY	1.00	12/09/85	12/31/99	

Component	Description	Qty Per	As of	Until	Reference

Assembly: A-6001-04 (ASY 6001:FRONT ASY)


1C-LS374
3C-LS534
J-2
J3
2C-LS642-1
CKWH
D7
D6
D5
D4
D3
D2
D1
D0
CKWM
S1
WEN
CAMEN
1B-LS374
3B-LS534
J-2
2B-LS642-1
1A-LS374
3A-LS534
J-2
CKWL
2A-LS642-1
+5.3
RP2-2 510
RP2-15 510
RP1-8 510
RP1-9 510
RP1-7 510
RP1-10 510
RP1-6 510
RP1-11 510
CAMAC EDGE CONN
W24
W23
W22
W21
W20
W19
W18
W17
RP2-6 510
RP2-11 510
RP2-5 510
RP2-12 510
RP2-4 510
RP2-13 510
RP2-3 510
RP2-14 510
W16
W15
W14
W13
W12
W11
W10
W9
RP3-3 510
RP3-14 510
RP3-2 510
RP3-15 510
RP2-8 510
RP2-9 510
RP2-7 510
RP2-10 510
W8
W7
W6
W5
W4
W3
W2
W1
DSP TECH
DATAWAY BOARD
SCHEMATIC DIAGRAM
5001-01-SD
SHEET 1 OF 3


OERL
OERM
OERH
J-3
SHEET1
S1
D7
D6
D5
D4
D3
D2
D1
D0
TOSHEET1
4C-LS374
5C-LS534
5B-LS534
4B-LS374
5A-LS534
4A-LS374
J-2
J-3
REN
R/W-24
R/W-23
R/W-22
R/W-21
R/W-20
R/W-19
R/W-18
R/W-17
R/W-16
R/W-15
R/W-14
R/W-13
R/W-12
R/W-11
R/W-10
R/W-9
R/W-8
R/W-7
R/W-6
R/W-5
R/W-4
R/W-3
R/W-2
R/W-1
+5.3
RP3-13 510
RP3-4 510
RP3-12 510
RP3-5 510
RP3-11 510
RP3-6 510
RP3-10 510
RP3-7 510
RP3-9 510
RP3-8 510
RP4-15 510
RP4-2 510
RP4-14 510
RP4-3 510
RP4-13 510
RP4-4 510
RP4-12 510
RP4-5 510
RP4-11 510
RP4-6 510
RP4-10 510
RP4-7 510
RP4-9 510
RP4-8 510
CAMAC EDGE CONN
R24
R23
R22
R21
R20
R19
R18
R17
R16
R15
R14
R13
R12
R11
R10
R9
R8
R7
R6
R5
R4
R3
R2
R1
DSP TECH
DATAWAY BOARD
SCHEMATIC DIAGRAM
WBC
5001-01-SD
SHEET 2 OF 3



J-2 CONNECTOR PINOUT

SIGNAL			SIGNAL
R/W — 59	o 60	— D	
24 — 57	o 58	— X	
23 — 55	o 56	— B	
22 — 53	o 54	— Z	
21 — 51	o 52	— C	
— 49	o 50	— Y	
20 — 47	o 48	— Q	
19 — 45	o 46	— +5	
18 — 43	o 44	— L16	
17 — 41	o 42	— L8	
— 39	o 40	— L4	
16 — 37	o 38	— L2	
15 — 35	o 36	— L1	
14 — 33	o 34	— +5	
13 — 31	o 32	— N16	
— 29	o 30	— N8	
12 — 27	o 28	— N4	
11 — 25	o 26	— N2	
10 — 23	o 24	— N1	
9 — 21	o 22	— +5	
— 19	o 20	— A8	
8 — 17	o 18	— A2	
7 — 15	o 16	— A1	
6 — 13	o 14	— A1	
5 — 11	o 12	— +5	
— 9	o 10	— F16	
4 — 7	o 8	— F8	
3 — 5	o 6	— F4	
2 — 3	o 4	— F2	
1 — 1	o 2	— F1	

DSP TECH
DATAWAY BOARD
SCHEMATIC DIAGRAM
6001-01-SD
SHEET 3 OF 3

Assembly: A-6002-02 (ASY 6001/6002 CONTROLLER BOARD ASY)
Revision:
As of: 11/10/90

	Component	Description	Qty Per
1.	0100-0103	RES: RC07 10K OHM 5% 1/4W	7.00
2.	0100-0104	RES: RC07 100K OHM 5% 1/4W	2.00
3.	0100-0152	RES 5% 1/4W CC RESISTOR R=1.5K	2.00
4.	0100-0221	RES 5% 1/4W CC RESISTOR R=220	1.00
5.	0100-0391	RES 5% 1/4W CC RESISTOR R=390	1.00
6.	0100-0511	RES 5% 1/4W CC RESISTOR R=510	1.00
7.	0150-2002	RMF RN55D 1% R=20.0K	1.00
8.	0216-0511	RPK DIP 15R=510 DALE MDP1601-511J	5.00
9.	0500-0101	CSM C=100PF CDE # CD5FC101J03	2.00
10.	0520-0104	CMC 0.1UF CNTLB:CZ20A104M2 KEM:C322	29.00
11.	0520-0105	CMC C=1.0UF MONO 50WVDC KEM# C320	1.00
12.	0600-0685	CAP TANT 6.8UF 35V .250 LEAD SP	3.00
13.	1000-4448	DGS DIODE SIL. SWITCH 1N4448	1.00
14.	1030-5624	DSR: DIODE RECT 3A GE IN5624/BYM56	1.00
15.	120440	HDW: SCREW 4-40 X 3/16 BHMS ZINC	4.00
16.	1750-0038	ICS SN7438	1.00
17.	1750-0156	ICS SN74156	3.00
18.	1760-0004	ICS SN74LS04	3.00
19.	1760-0008	ICS SN74LS08	4.00
20.	1760-0014	ICS SN74LS14	1.00
21.	1760-0032	ICS SN74LS32	3.00
22.	1760-0074	ICS SN74LS74	4.00
23.	1760-0112	ICS SN74LS112	5.00
24.	1760-0125	ICS SN74LS125	1.00
25.	1760-0138	ICS SN74LS138	3.00
26.	1760-0174	ICS SN74LS174	4.00
27.	1760-0175	ICS SN74LS175	2.00
28.	1760-0221	ICS SN74LS221	2.00
29.	1760-0348	ICS SN74LS348	4.00
30.	1760-0640	ICS SN74LS640	1.00
31.	1760-0642	ICS SN74LS642	1.00
32.	1760-0645	ICS SN74LS645	5.00
33.	1760-1642	ICS SN74LS642-1	2.00
34.	2250-0206	OSC XTAL CLOCK 20.0MHZ 0.01% K1100	1.00
35.	3100-0402	FUS FUSE 4AMP LITTLEFUSE 251004	1.00
36.	5000-0016	SKT CAMBION 703-1316-01-04-10	6.00
37.	5000-0020	SKT CAMBION 703-1320-01-04-10	1.00
38.	5250-3703	JCK FUS CAMBION 450-3703-01-06-00	2.00
39.	5500-0002	CON MFG-PN 3432-1202 SOLDER TAIL	1.00
40.	5500-0005	HDR 40 PIN APTRONICS 929667-01-20	1.00
41.	5500-0017	HDR 40 PIN 3M 3595-6002	1.00
42.	6850-0001	RAIL: GUIDE, MOL, 5 HOLE, C-BORE	2.00
43.	7000-0028	PCB DWG# 6002-02-FD	1.00
44.	7410-0003	HWD: SWAGE TYPE STANDOFF	6.00

Component | Description | Qty Per

Assembly: A-6002-02 (ASY 6001/6002 CONTROLLER BOARD ASY)

DSP[redacted]hnology Inc.
B[redacted]L OF MATERIAL REPORT

Date: 11/13/90
Page: 1

As of	Until	Reference
08/07/86	12/31/99	R2-R4 R11-R14
01/01/80	12/31/99	R7 R8
01/01/80	12/31/99	R6 R5
01/01/80	12/31/99	R1
01/01/80	12/31/99	R10
01/01/80	12/31/99	R15
07/08/87	12/31/99	RX1
08/07/86	12/31/99	RP1-RP5
07/08/87	12/31/99	C2,C3
12/09/85	12/31/99	
01/01/80	12/31/99	C4
07/08/87	12/31/99	C1,C5,CX1
12/09/85	12/31/99	DX1
01/01/80	12/31/99	D1
12/09/85	12/31/99	
01/01/80	12/31/99	3G
08/07/86	12/31/99	6B-8B
01/01/80	12/31/99	3E 7C 6E
01/01/80	12/31/99	7D 5F 3F 1E
01/01/80	12/31/99	4G
01/01/80	12/31/99	4F 5E 2F
01/01/80	12/31/99	4B 6D 5D 6F
01/01/80	12/31/99	2E 7E 7F 8C 8D
01/01/80	12/31/99	8E
01/01/80	12/31/99	5C 6G 7G
01/01/80	12/31/99	3D 4D 2D 6C
01/01/80	12/31/99	4E 1F
01/01/80	12/31/99	1G 2G
01/01/80	12/31/99	1C 3B 1B 2B
01/01/80	12/31/99	2C
01/01/80	12/31/99	3C
01/01/80	12/31/99	1D 8A 6A 5A 8G
01/01/80	12/31/99	4C 5B
01/01/80	12/31/99	8F
12/09/85	12/31/99	F1
05/01/89	12/31/99	8B RP1-RP5
01/01/80	12/31/99	8A
08/07/86	12/31/99	F1
10/23/85	12/31/99	J4
12/09/85	12/31/99	J5
01/01/80	12/31/99	J3
12/09/85	12/31/99	
01/01/80	12/31/99	
12/09/85	12/31/99	

As of	Until	Reference

SINGLE LEVEL

Assembly: A-6002-03 (ASY 6002:FRONT PANEL/MECH ASY)
Revision:
As of: 11/10/90

	Component	Description	Qty Per
1.	1040-0002	DLE LED RED HP HLMP-1301	1.00
2.	120425	HDW: SCREW 4-40 X 5/16 PHIL FLAT SS	8.00
3.	120440	HDW: SCREW 4-40 X 3/16 BHMS ZINC	4.00
4.	123043	HDW: #4 EXTERNAL LOCK WASH ZINC	4.00
5.	123046	HDW: WASHER #4 FLAT NYLON	4.00
6.	124044	HDW SPACER 4.40 X 7/16 HEX 1/4 THRE	4.00
7.	5510-0001	CON ERA 00.250 CTL & WASHER	3.00
8.	5510-0006	CON GND LUG LEMO #GCA.00.255.LT	3.00
9.	5510-0007	NUT FOR LEMO 5510-0001 RA00.240	3.00
10.	6020-0001	SW:SPDT (ON)-OFF-(ON) MOM C&K 7105	1.00
11.	6020-0002	CAP FOR 6020-0001 ALCO C-10-1-25	2.00
12.	6020-0003	NUT DRESS ALCO N-40 CTLR-HMMR WNU-	2.00
13.	6020-0005	SWH ON-OF TOGGLE C&K7101SYZQ	1.00
14.	6851-0022	PNL: FR, 6002	1.00
15.	6852-0001	SCR:FT PNL JCK DEUTCH D7500M4B4	1.00
16.	7250-0001	CLP LED MNT LITRONIX LD32-004-9016	1.00
17.	A-6001-03	ASY 6001:FRONT PANEL PCB ASY	1.00

	Component	Description	Qty Per

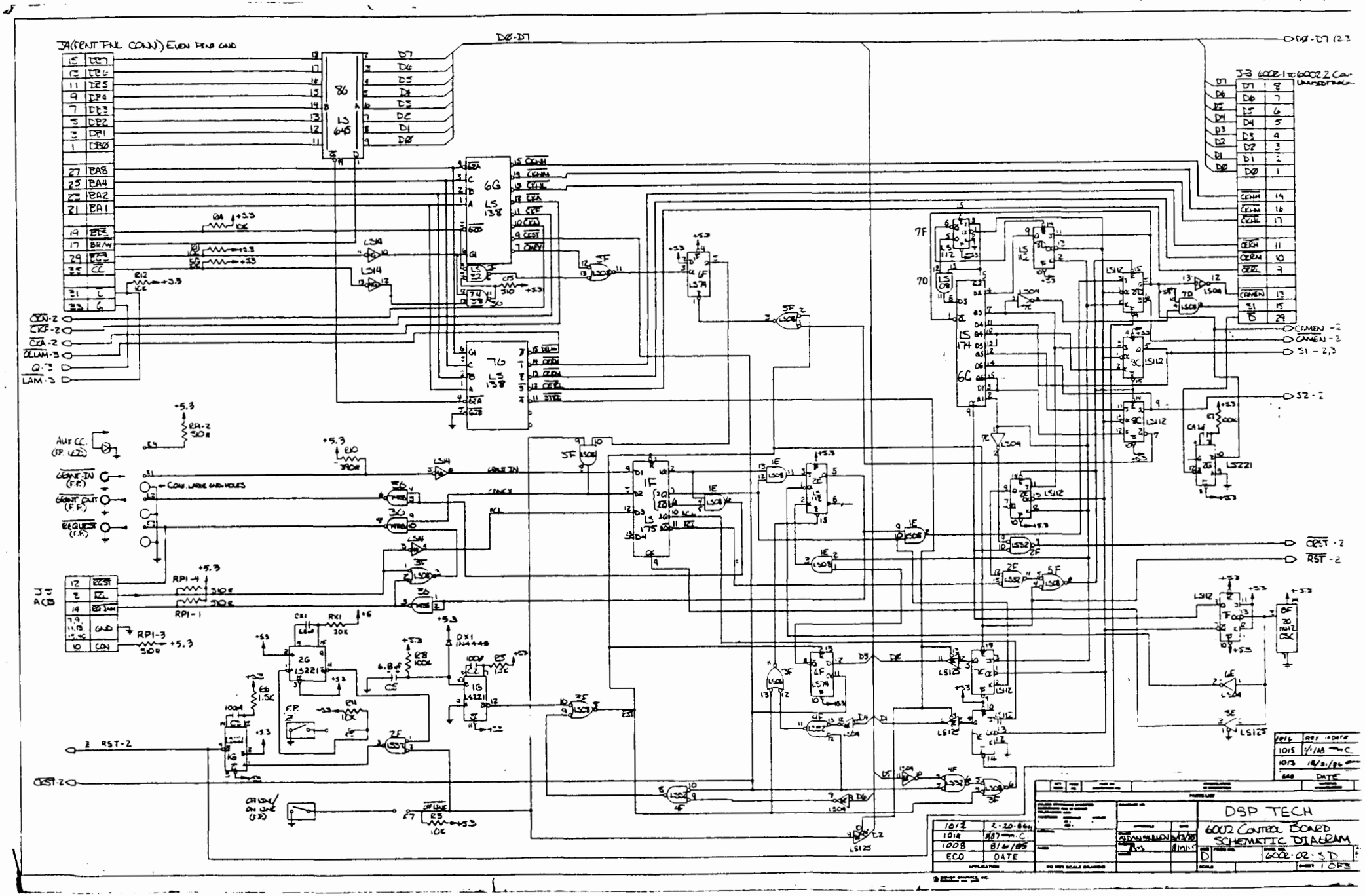
Assembly: A-6002-03 (ASY 6002:FRONT PANEL/MECH ASY)

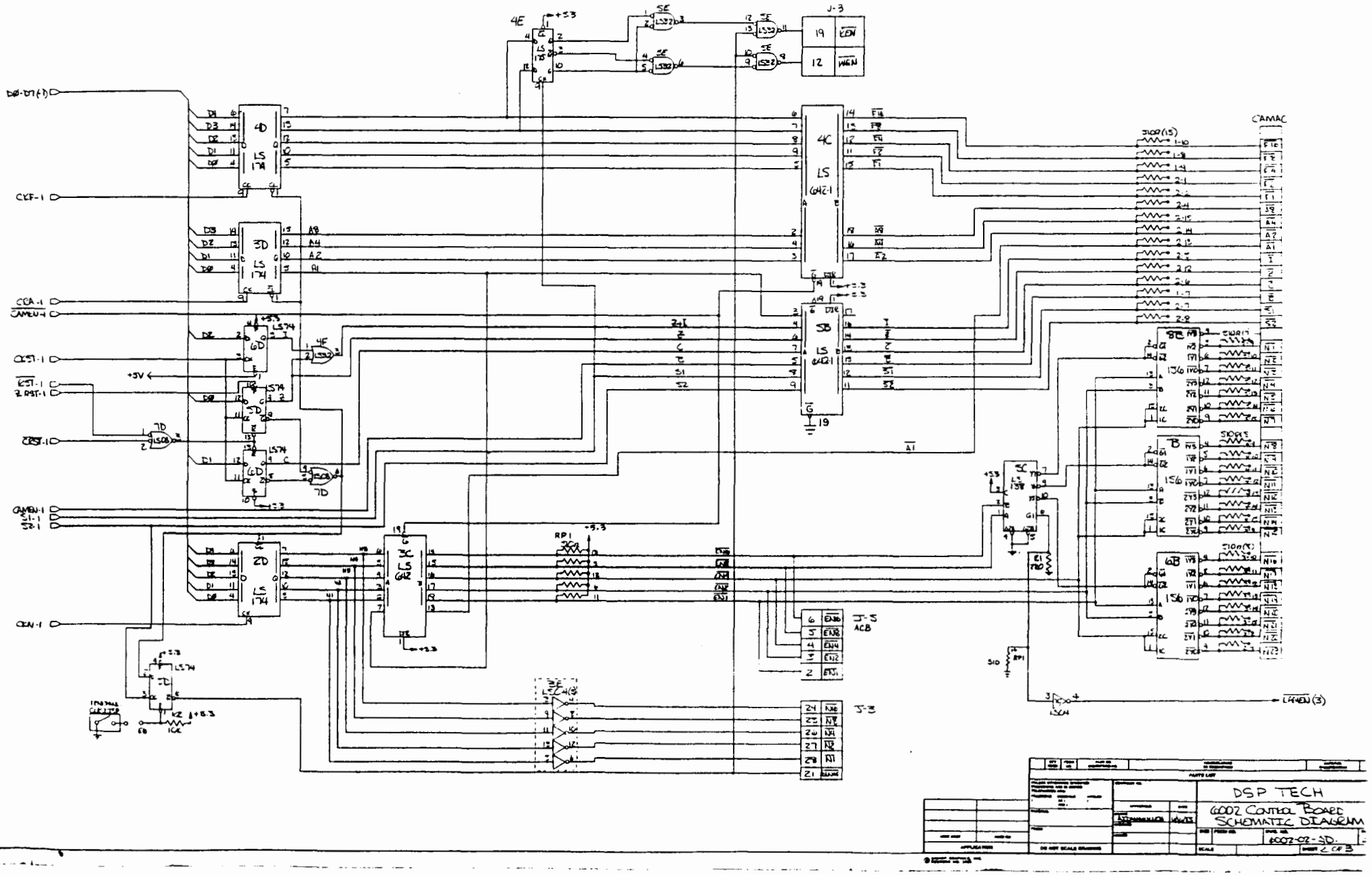
DSP[redacted]hnology Inc.
B[redacted]L OF MATERIAL REPORT

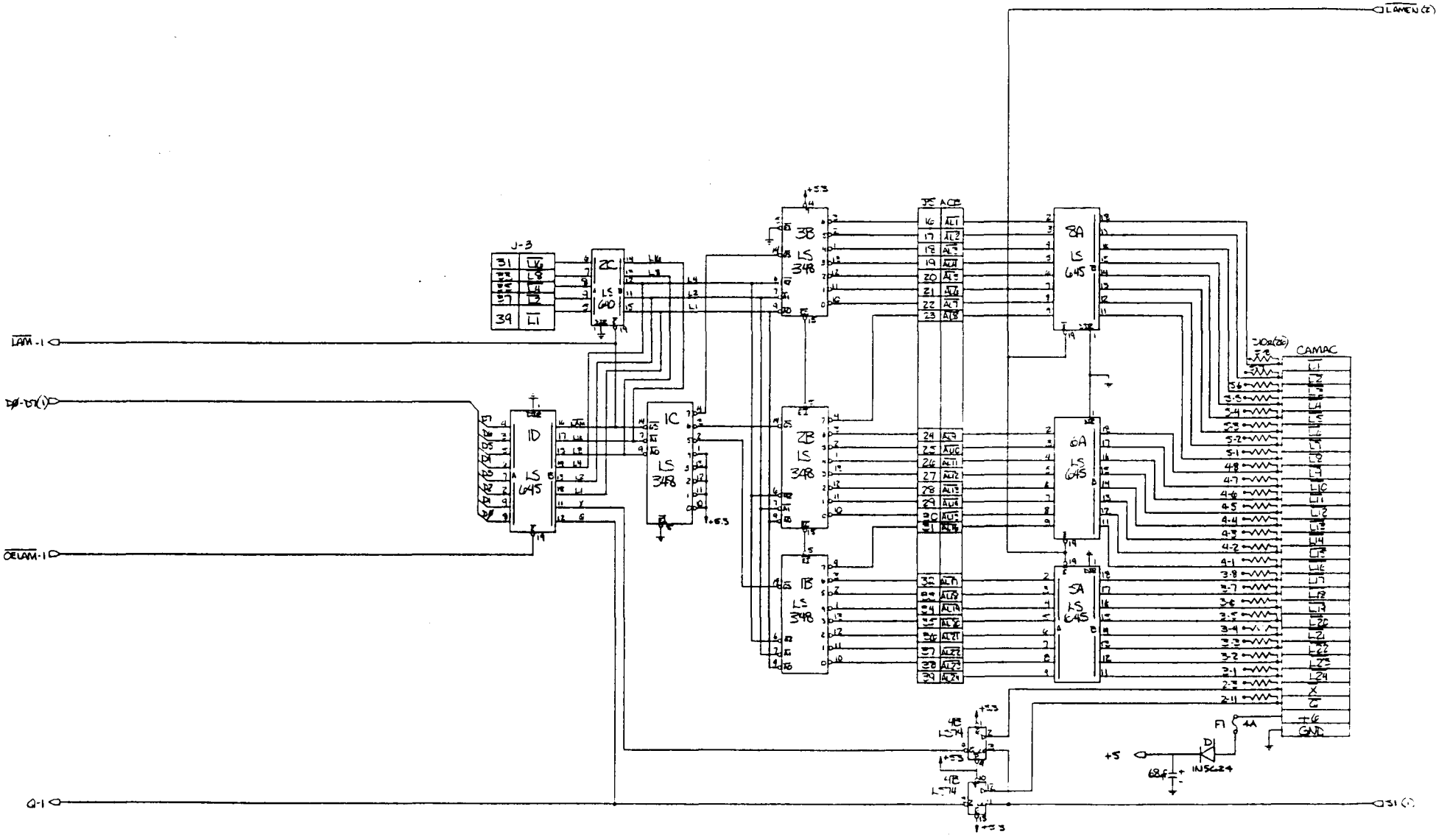
Date: 11/13/90
Page: 1

As of	Until	Reference
12/09/85	12/31/99	AUX CC
05/01/89	12/31/99	
05/01/89	12/31/99	
05/01/89	12/31/99	
05/01/89	12/31/99	
05/01/89	12/31/99	
12/09/85	12/31/99	
12/09/85	12/31/99	
12/09/85	12/31/99	
12/09/85	12/31/99	SW2
12/09/85	12/31/99	
12/09/85	12/31/99	
12/09/85	12/31/99	SW1
12/09/85	12/31/99	
12/09/85	12/31/99	
12/09/85	12/31/99	
12/09/85	12/31/99	

As of	Until	Reference


J4 (FRONT PNL CONN) EVEN PINS GND
D0-D7
J3 6002-1 TO 6002-2 CONN
DSP TECH
6002 CONTROL BOARD
SCHEMATIC DIAGRAM
6002-02-3D
ACB
RST-2
CRST-2
CMEN-2
S1-2,3
S2-2
GRANT-IN (F.P.)
GRANT OUT (F.P.)
REQUEST (F.P.)
AUX CC (F.P. LED)
RPI-4
RPI-1
RPI-3
F.P.
DX1 1N4448
ECO
DATE


DSP TECH
6002 CONTROL BOARD
SCHEMATIC DIAGRAM
6002-02-SD.
CAMAC


LAMEN(X)
J-3
LAM-1
D0-D7(1)
OELAM-1
Q-1
Q-31(1)
CAMAC
+5
68µF
1N5624
F1
4A
GND
NOTE: —— DESIGNATES CONNECTION TO +5V HEADER
DSP TECH
6C02 CONTROL BOARD
SCHEMATIC DIAGRAM
6C02-02-3D

DSP[redacted]hnology Inc.

SINGLE LEVEL B[redacted]L OF MATERIAL REPORT

Assembly: A-6001-03 (ASY 6001:FRONT PANEL PCB ASY)
Revision:
As of: 11/10/90

Date: 11/13/90
Page: 1

	Component	Description	Qty Per	As of	Until	Reference
1.	0201-0471	RPK RES PK BNS 4310R-101-471	6.00	01/01/80	12/31/99	
2.	2052-0001	LED (10) 20 PIN DIP GI MV57164	6.00	01/01/80	12/31/99	
3.	5050-0001	SKT 20 STRIP SAMTEC SS120G-2	6.00	01/01/80	12/31/99	
4.	5500-0011	HDR FLT CLB 72PIN 929665-01-36	1.00	01/01/80	12/31/99	
5.	7010-0002	PCB DWG# 6001-03-FD	1.00	01/01/80	12/31/99	

Component	Description	Qty Per	As of	Until	Reference

Assembly: A-6001-03 (ASY 6001:FRONT PANEL PCB ASY)

Assembly: PC004 (DMA IF FOR IBM XT/AT INCL. SFTWR)
Revision: D
As of: 11/10/90

	Component	Description	Qty Per
1.	0100-0102	RES: RC07 1K OHM 5% 1/4W	2.00
2.	0100-0202	RES 5% 1/4W CC RESISTOR R=2K	1.00
3.	0201-0103	RPK RES PK BNS 4310R-101-103	2.00
4.	0520-0104	CMC 0.1UF CNTLB:CZ20A104M2 KEM:C322	15.00
5.	0600-0685	CAP TANT 6.8UF 35V .250 LEAD SP	4.00
6.	1760-0004	ICS SN74LS04	3.00
7.	1760-0008	ICS SN74LS08	1.00
8.	1760-0020	ICS SN74LS20	2.00
9.	1760-0032	ICS SN74LS32	2.00
10.	1760-0074	ICS SN74LS74	1.00
11.	1760-0085	ICS SN74LS85	2.00
12.	1760-0125	ICS SN74LS125	3.00
13.	1760-0138	ICS SN74LS138	1.00
14.	1760-0175	ICS SN74LS175	1.00
15.	1760-0193	ICS SN74LS193	2.00
16.	1760-0244	ICS SN74LS244	4.00
17.	1760-0374	ICS SN74LS374	1.00
18.	1760-0645	ICS SN74LS645	5.00
19.	1851-0017	PRM PROM PROG/F 1850-0001 PC004	1.00
20.	5000-0016	SKT CAMBION 703-1316-01-04-10	1.00
21.	5500-0002	CON MFG-PN 3432-1202 SOLDER TAIL	4.00
22.	6000-0001	STH 16 PIN DIP SWT GRYHL 76RS808	1.00
23.	7010-0011	PCB PC004 DWG PC004-01-FD	1.00
24.	7705-0002	HDW PCB GUIDE IBM UNITRACK SPC3000	1.00
25.	PSP9100	CAM6001 SOFTWARE	1.00

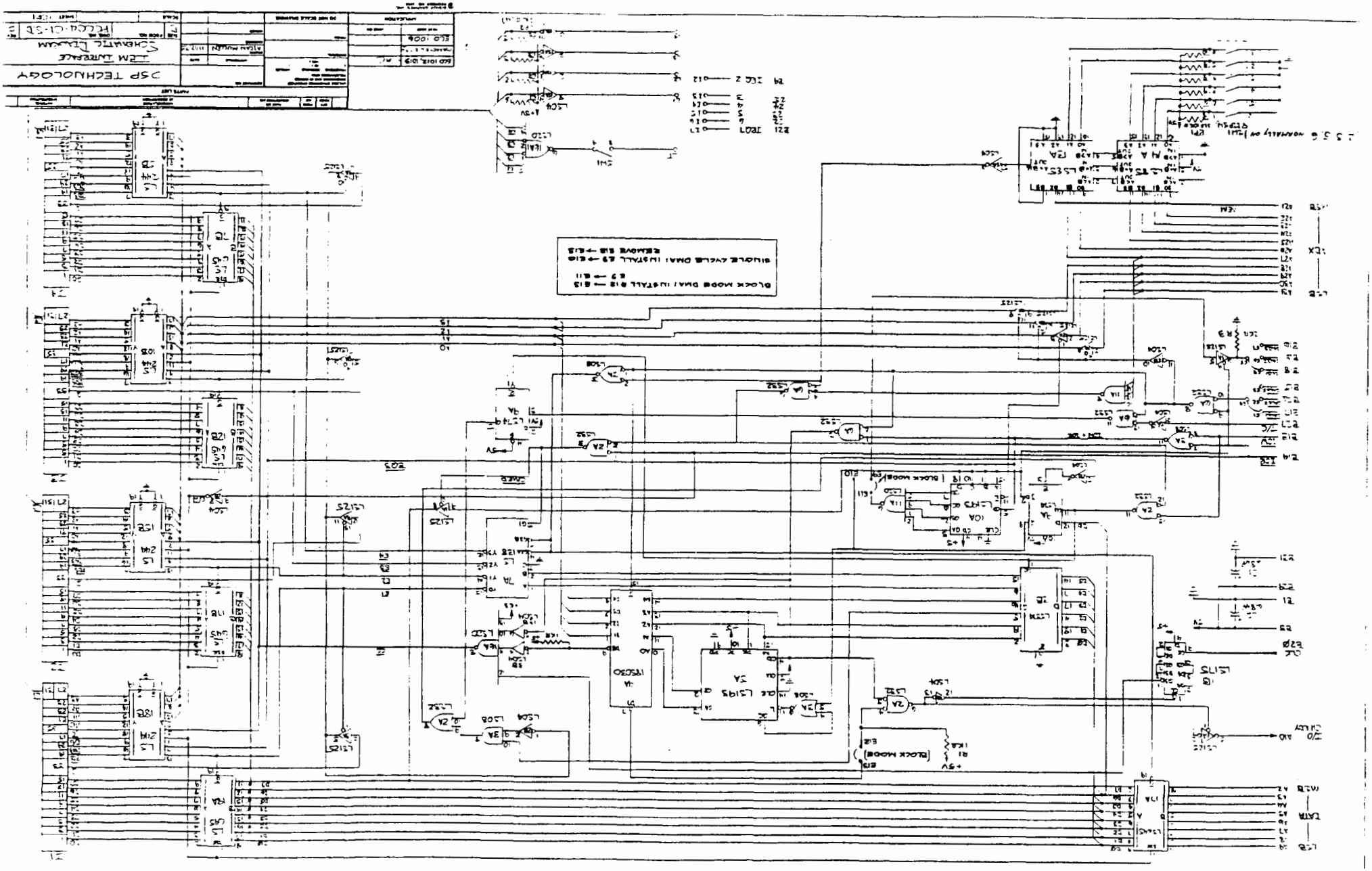
	Component	Description	Qty Per
	Assembly: PC004	(DMA IF FOR IBM XT/AT INCL. SFTWR)	

DSP[redacted]hnology Inc.
B[redacted]L OF MATERIAL REPORT

Date: 11/13/90
Page: 1

As of	Until	Reference
01/01/80	12/31/99	R1 R2
09/16/85	12/31/99	R3
01/01/80	12/31/99	RP1 RP2
01/01/80	12/31/99	
09/16/85	12/31/99	C1 C2 C3 C4
01/01/80	12/31/99	15A 2B 13B
01/01/80	12/31/99	3A
01/01/80	12/31/99	11A 16A
01/01/80	12/31/99	2A 6A
01/01/80	12/31/99	9A
01/01/80	12/31/99	13A 14A
01/01/80	12/31/99	1A 8A 12A
01/01/80	12/31/99	7A
01/01/80	12/31/99	1B
01/01/80	12/31/99	5A 10A
01/01/80	12/31/99	5B 10B 15B 18B
01/01/80	12/31/99	3B
01/01/80	12/31/99	17A 18A 7B 12B 17B
01/10/86	12/31/99	4A
01/01/80	12/31/99	4A
01/01/80	12/31/99	J1 J2 J3 J4
01/01/80	12/31/99	SW1
01/01/80	12/31/99	
01/01/80	12/31/99	
09/07/89	12/31/99	

As of	Until	Reference



dsp technology inc.

PRODUCT WARRANTY

(Excludes Software)

DSP TECHNOLOGY ("DSP") warrants its products to be free from defects in material and workmanship and to meet DSP's performance specifications. The warranty period is one year from the date of shipment to the buyer. The warranty is limited by the paragraphs below.

RETURN TO FACTORY

If a buyer discovers a defect in a DSP product covered by this agreement, buyer's exclusive remedy is to ship the product back to DSP's Fremont factory, where DSP will, at its option, either repair or replace the product. This remedy applies if DSP receives the returned product on or before the tenth day after the expiration of the warranty period and the buyer notifies DSP of the defect before returning the product.

COST TO BUYER OF REPAIRS OR REPLACEMENT

Buyer must prepay freight charges to DSP. DSP will pay standard return freight to buyer. Buyer will be charged for premium freight if that method of transport is requested by buyer. There is no other charge for repair or replacement during the warranty period.

TRANSFERRABLE WARRANTIES

In addition to the foregoing warranty, DSP also provides the buyer the transferrable warranty, if any, provided to DSP by the manufacturers of other products such as terminals and disk systems supplied by SEC as part of a total system.

LIMITATION OF WARRANTY AND LIABILITY

The foregoing constitutes DSP's entire warranty, expressed, implied, and/or statutory (except as to title), and to any other party for any breach of such warranty and for damages, whether direct, special, incidental, or consequential. Other than as expressly provided in this document, no warranties expressed or implied, including any warranty of merchantability or fitness for a particular purpose are made. No employee, representative, or agent or seller has any authority expressed or implied to alter or supplement the terms of this warranty.

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP