SPERRY UNIVAC

SPECIFICATION APPROVAL SHEET

SHEET 1

PRODUCT FUNCTION SPECIFICATION

DATA PROCESSING SET AN/UYK-20(V) AND AN/UYK-20X(V)

SHEET INDEX

SHEET	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
REV	H	H	H	F	G	F	F	H	F	F	H	F	H	G	G	-	F	-	H	-	H	F	H

SHEET	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46
REV	F	F	F	F	F	H	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	H

SHEET	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69
REV	F	F	H	F	F	F	F	F	F	F	F	H	G	F	G	F	G	G	-	G	G	G	-

SHEET	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92
REV	G	F	H	H	H	G	F	F	G	G	G		G	F	F	F	F	F	F	F	F	F	F

THE AUTHORIZED ISSUE OF THIS SPECIFICATION CONSISTS OF THE SHEET REVISIONS ("—" IN REVISION SPACE DENOTES ORIGINAL ISSUE) SHOWN IN THE SHEET INDEX SECTION ON THE LATEST ISSUE OF THIS APPROVAL SHEET. THE DATE OF THE LATEST ISSUE IS SHOWN ON THE REVISION RECORD SHEET.

ORIGINAL		MO	DAY	YR
WRITTEN BY	D.A. Klobe	2	7	73
ORGANIZATION	Technical Specifications			
APPROVAL	R.R. Reeder	2	7	73
ORGANIZATION	TECHNICAL SPECIFICATIONS			
APPROVAL	M.J. DeBlauw	2	7	73
ORGANIZATION	Product Development			
APPROVAL	N/A			
ORGANIZATION				
APPROVAL	N/A			
ORGANIZATION				

SPERRY UNIVAC
UNIVAC PARK, P.O. BOX 3525
ST. PAUL, MINNESOTA 55165

CODE IDENTIFICATION NUMBER 90536

SPECIFICATION NUMBER	SPEC. REV.
SB-10160	H

UNIVAC

SPECIFICATION APPROVAL SHEET CONTINUATION

SHEET 2	REVISION H

SPECIFICATION SYMBOL
SB-10160

SHEET INDEX

SHEET	93	94	95	96	97	98	99	100	101	102	103	104	105	106
REV.	F	F	F	F	F	H	H	F	F	H	H	H	F	H
SHEET	107	108	109	110	111	112	113	114	115	116	117	118	119	120
REV.	F	F	G	G	F	F	H	F	F	F	F	F	F	F
SHEET	121	122	123	124	125	126	127	128	129	130	131	132	133	134
REV.	F	F	F	F	H	F	H	F	F	H	H	F	F	F
SHEET	135	136	137	138	139	140	141	142	143	144	145	146	147	148
REV.	H	F	F	F	F	F	F	F	H	H	H	F	F	F
SHEET	149	150	151	152	153	154	155	156	157	158	159	160	161	162
REV.	F	F	F	F	F	F	F	F	F	F	F	F	F	F
SHEET	163	164	165	166	167	168	169	170	171	172	173	174	175	176
REV.	F	H	F	F	F	F	G	F	F	G	G	G	F	G
SHEET	177	178	179	180	181	182	183	184	185	186	187	188	189	190
REV.	G	F	H	H	H	F	F	H	F	F	F	F	F	F
SHEET	191	192	193	194	195	196	197	198	199	200	201	202	203	204
REV.	F	H	H	F	F	F	F	F	F	H	F	F	F	F
SHEET	205	206	207	208	209	210	211	58.1	70.1	172.1	173.1	174.1	177.1	131.1
REV.	F	F	F	F	G	F	H	G	G	J	J	J	J	H
SHEET	135.1	200.1												
REV.	H	H												
SHEET														
REV.														
SHEET														
REV.														
SHEET														
REV.														
SHEET														
REV.														
SHEET														
REV.														

UDI-2902-1

SPERRY RAND

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

DATA PROCESSING SET
AN/UYK-20(V) and
AN/UYK-20X(V)

1. SCOPE

1.1 Scope. - This specification establishes the requirements for a microprogrammed 16-bit data processing set, hereinafter referred to as the DPS. The DPS provides a standard processor that meets the processing requirements of various shipboard, land based, and submarine applications.

2. APPLICABLE DOCUMENTS

2.1 Government documents. - The following documents of the exact issue shown form a part of this specification to the extent specified herein.

SPECIFICATIONS

Military

MIL-S-901C	Shock Test, H. I. (High Impact); Shipboard Machinery, Equipment and Systems Requirements for
MIL-M-7793D with Supplement 1 31 December 1969	Meter, Time Totalizing
MIL-Q-9858A 16 December 1963	Quality Program Requirements
MIL-P-15024D with Supplement 1 9 March 1972	Plates, Tags and Bands for Identification of Equipment
MIL-E-15090B Amendment 2 29 February 1956	Enamel, Equipment, Light Gray Formula 111
MIL-E-16400F Amendment 5 28 November 1969	Electronic Equipment, Naval Ship and Shore: General Specification
MIL-E-17555G Amendment 2 15 April 1970	Electronic and Electrical Equipment, Accessories, and Repair Parts: Packaging and Packing of
MIL-F-18870D 18 August 1967	Fire Control Equipment, Naval Ship and Shore, General Specification
MIL-P-19834B 7 November 1972	Plate, Identification, Metal Foil, Adhesive Backed

Naval Electronics Systems Command

ELEX-C-135 No Revision 27 November 1972	Computer, Digital Data, Combat System

STANDARDS

Federal

FED-STD-151b 24 Nov 1967	Metals, Test Methods

Military

MIL-STD-108E 04 August 1966	Definitions of the Basic Requirements for Enclosures for Electric and Electronic Equipment
MIL-STD-130D with change Notice 1 30 July 1971	Identification Marking of U. S. Military Property
MIL-STD-167B 11 August 1969	Mechanical Vibrations of Shipboard Equipment
MIL-STD-188C 24 Nov 1969	Military Communication System Technical Standards
MIL-STD-189 with Notice 2 14 March 1961	Racks, Electrical Equipment, 19-Inch and Associated Panels
MIL-STD-454C 15 Oct 1970	Standard General Requirements for Electronic Equipment
MIL-STD-461A 01 May 1970	Electromagnetic Interference Characteristics, Requirements for Equipment
MIL-STD-462 with Notice 3 9 February 1971	Electromagnetic Interference Characteristics, Measurement of
MIL-STD-740B Notice 1 22 June 1965	Airborne and Structureborne Noise Measurements and Acceptance Criteria of Shipboard Equipment

MIL-STD-1399A 20 December 1972	Interface Standard for Shipboard Systems
MIL-STD-1472A 15 May 1970	Human Engineering Design Criteria for Military Systems, Equipment and Facilities

HANDBOOKS

Military

MIL-HDBK-217A with Notice 2 30 December 1972	Reliability Stress and Failure Rate Data for Electronic Equipment

(Copies of specifications, standards, drawings, and publications required by suppliers in connection with specific procurement functions should be obtained from the procuring activity or as directed by the contracting officer).

PUBLICATIONS

Electronic Industries Association (EIA)

RS-232-C August 1969	Interface Between Data Terminal and Data Communication Equipment Employing Serial Binary Data Interchange.

(Copies of EIA publication may be obtained from EIA Engineering Dept. 2001 Eye St. NW, Washington D.C. 20006)

2.2 Non-Government documents. - The following documents form a part of this specification to the extent specified herein. Unless otherwise indicated the latest issue in effect shall apply.

SPECIFICATIONS

Sperry Univac

DS 4772	Interface Characteristics, Univac Defense Computer Equipment
SB-10441-100	Data Processing Set AN/UYK-20(V) Acceptance Procedures for
SB-10441-400	Data Processing Set AN/UYK-20(V) Test Data Record
SB-10592	Microprogrammed Controller, AN/UYK-20(V)
SB-10631	Serial Interface MIL-STD-188 and EIA STD RS-232
SB-12407	Serial Interface Characteristics Univac Defense Computer Equipment

DRAWINGS

905411	Connector, Plug, Electrical-- Female, 7 contact, MS 3106
7101943	Connector Assembly, Electrical - 120 pin
7126400	Data Processing Set AN/UYK-20(V) Outline and Installation Drawing
7901743	Connector, Plug Electrical -- Female, 5 Contact, MS 3126F

(Application for copies should be addressed to Univac, Division of Sperry Rand, P.O. Box 3525, St. Paul, MN 55165.)

3. REQUIREMENTS

3.1 Design and construction. -

3.1.1 General design. - Except as specified herein, the DPS shall meet all requirements of MIL-E-16400. The objectives specified in MIL-E-16400 for reliability, simplicity, ease of installation and maintenance shall be considered during all phases of the equipment design.

3.1.1.1 Modular construction. - The DPS shall employ modular construction in accordance with MIL-E-16400. The DPS shall use plug-in printed-wiring assemblies to the fullest extent practical.

3.1.1.2 Interchangeability. - The DPS shall meet the following interchangeability requirements:

a. The DPS shall meet the requirements of MIL-STD-454, Requirement 7, Interchangeability.

b. All replaceable assemblies shall be physically and electrically interchangeable with corresponding parts of all DPS's covered by this specification without drilling, filing, or the use of undue force. Replacement assemblies, when installed, shall not cause a departure from the performance specified herein.

c. The DPS shall be interchangeable with the same type of DPS when configured with the same options as defined in 6.1.

3.1.1.3 Standardization. - The DPS design shall use the greatest practical uniformity of engineering criteria, items, processes, equipments, and parts to ensure the minimum variety of such items, and to effect optimum interchangeability of equipment parts and components. The DPS design shall use equipment and components already supported in the Naval/Defense Supply Agency supply system to the maximum extent possible.

3.1.1.4 Installed options. - The DPS shall be fully wired to accommodate all optional features and requirements (see 6.1). Except that the input power option shall be specified prior to manufacture, all configuration options shall be installed or removed by the pluggable addition, removal or replacement of appropriate hardware modules. The removal or addition of all options shall be permitted within the constraints specified herein.

3.1.2 Mechanical design. -

3.1.2.1 Physical size and weight. - The size and weight of a maximum configuration DPS shall be as specified in table I. The DPS shall be capable of being passed through a circular hatch 25 inches in diameter and through an opening 20 inches wide and 28 inches high, with 10 inch radius rounded corners. The DPS mounting and clearance dimensions shall be as specified on Drawing 7126400.

Table I. Size and Weight

Parameter	
Height	20 inches max.
Width	19 inches max.
Depth	24 inches max. (not including shock pins)
Weight	220 pounds max.

3.1.2.2 Mechanical configuration. - The DPS mechanical configuration shall be as shown in figure 1.

3.1.2.3 Rack mounting. - The DPS shall be capable of being base mounted or rack mounted. Mounting in electrical equipment racks shall be in accordance with MIL-STD-189.

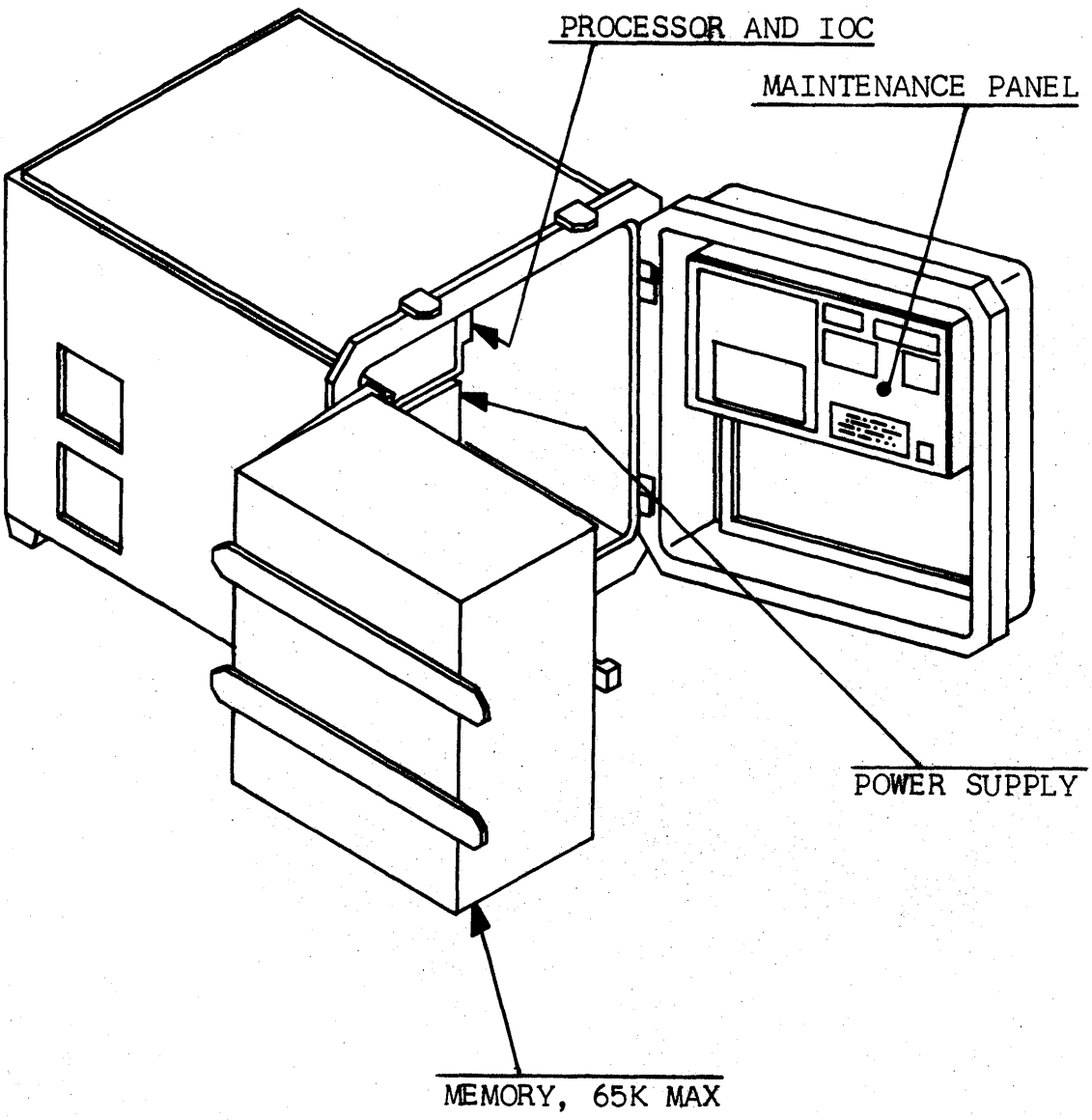


FIGURE 1 - DPS MECHANICAL CONFIGURATION

3.1.2.4 Interface connectors. - The DPS interface connector part numbers shall be as specified on Drawing 7126400. The DPS interfaces shall mate with the mating connectors specified on Drawing 7126400. The jack and pin assignments shall be as specified in Appendix 10.

3.1.2.5 Cooling. - The DPS shall be air cooled using ambient air within the operating temperature range specified herein. The DPS shall exhaust 3420 BTU's per hour maximum into the ambient environment. Cooling fans shall be integral to the cabinet. The DPS shall be designed such that water cooling can be implemented through redesign of the cabinet only.

3.1.2.6 Plug-in assembly keying. - All plug-in assemblies shall be mechanically keyed.

3.1.2.7 Human engineering. - The DPS shall incorporate human engineering in accordance with MIL-STD-1472.

3.1.2.8 Identification and marking. -

3.1.2.8.1 Nameplate. - The nameplate used on the DPS shall be in accordance with MIL-P-15024 except that it shall be bonded as specified in MIL-P-19834. Units with 400 Hz power shall have the AN/UYK-20(V) designation. Units with 60 Hz power shall have the AN/UYK-20X(V) designation. No company identification other than that allowed for the nameplate shall appear on the DPS.

3.1.2.8.2 Field change identification plate. - An identification plate shall be provided on the DPS for the sole purpose of noting field changes.

3.1.2.8.3 Markings. - Markings used on the DPS shall be in accordance with MIL-STD-130 and shall include serialization for the DPS replaceable assemblies, sub-assemblies, and printed wiring boards.

3.1.2.8.4 Reference designations. - Parts identification by reference designation shall be in accordance with MIL-E-16400.

3.1.2.9 Finish and color. - The exterior painted surfaces shall be in light gray which matches the color specified in MIL-E-15090.

3.1.3 Electrical design. -

3.1.3.1 Overload protection. - Overload protection shall be provided by the use of the circuit breakers in accordance with MIL-E-16400.

3.1.3.2 Time meter. - The DPS shall have a time meter that shall cumulatively record the time that ac power is being applied and the logic power switch is activated. The range of the meter shall be 0 to 9999 hours; the meter shall not be capable of being reset. The meter shall be in accordance with MIL-M-7793.

3.1.3.3 Grounding. - Circuit grounding shall be in accordance with MIL-F-18870. The DPS enclosure shall provide a point for grounding, by means of a suitable ground strap, to a single common ground point. All grounds within a section or subassembly shall be connected to a common ground on the enclosure.

3.1.3.4 Circuits. - The DPS circuits shall be designed such that grounding of the output elements or partial or complete power failure shall not cause damage to components. The sudden application of normal loads to output amplifiers or circuits shall not cause overload protection circuits to operate. Where printed wiring cards or modular construction are used, ground potential and voltages shall be assigned to pins of all plug-in assemblies in a uniform manner so that a given potential is applied to pins of the same number on all plug-in assemblies. All relay coils shall have diode suppression, or equivalent.

3.1.3.5 Test points. - Test points shall be accessible from the front of the DPS with the access doors open. MIL-E-16400 shall be used as a guide for selecting signals to be available at the test points. An adequate number of test points shall be located within the DPS to facilitate the performance of troubleshooting. Test points on plug-in assemblies shall be accessible without removing the assembly from its connector. Short circuits from a test point to ground shall cause no circuit damage.

3.1.3.6 Safety. - Safety requirements for the DPS shall be as specified in MIL-E-16400 and as specified herein.

3.1.3.7 Overtemperature protection. - The DPS shall have an overtemperature indicator that shall light and an audible alarm that shall sound when the cabinet internal air temperature is within 25°F of the maximum temperature at which the DPS can operate without component damage. The DPS shall stop operation when the cabinet internal air temperature exceeds the maximum safe operating temperature. The DPS shall have a switch to disable the audible alarm and a switch to override the overtemperature shutdown.

3.2 Primary power. - The DPS shall operate as specified herein when supplied with one of the sources (see 6.1) specified in table II. Each type of primary power shall require a corresponding type of power supply. Table II meets the requirements of Type I shipboard power of MIL-STD-1399 and conforms to the transient requirements of MIL-E-16400 except that steady state tolerance band shall be as specified in MIL-STD-1399.

Table II. Primary Power Options

Characteristics	Option 1	2	3	4	5	6
Voltage (volts)	115 line-to-line	115 line-to-line	208 line-to-line	208 line-to-line	115	115
Steady state voltage tolerance	± 5%	± 5%	± 5%	± 5%	± 7%	± 7%
Phases	3-Delta	3-Delta	3-Wye	3-Wye	1	1
Power (watts maximum for maximum configuration)	1000	1000	1000	1000	1000	1000
Phase rotation	A:B:C	A:B:C	A:B:C	A:B:C	Not applicable	Not applicable
Frequency	400 Hz	60 Hz	400 Hz	60 Hz	400 Hz	60 Hz
Steady state frequency tolerance	± 5%	± 5%	± 5%	± 5%	± 5%	± 5%
Maximum transient voltage(see note 1)	± 20%	± 20%	± 20%	± 20%	± 20%	± 20%
Maximum transient frequency (see note 2)	± 3%	± 3%	± 3%	± 3%	± 3%	± 3%

Note 1: Maximum transient voltage shall be in addition to the steady state tolerance and shall recover to the steady state tolerance within 2 seconds.

Note 2: Maximum transient frequency is 3% with 1% outside the steady state frequency tolerance limts.

3.3 General performance. - The DPS shall be a digital data processor which shall sequentially execute instructions from a program stored in memory. The DPS word length shall be 16-bits. The DPS shall use both single length (16-bit) and double length (32-bit) instructions. The DPS shall perform operations using literal (4-bit), byte (8-bit), single length (16-bit) and double length (32-bit) operands. The DPS shall have memory for program and data storage and input-output channels for data transfer operations with other equipments. The DPS shall use a microprogrammed control structure. The DPS shall have relative addressing and cascaded indirect addressing.

3.3.1 DPS arithmetic. - The DPS shall perform arithmetic operations using two's complement integer arithmetic. Arithmetic operations shall use signed numbers. The most significant bit of an operand shall be the sign bit: 0 shall mean a positive quantity, 1 shall mean a negative quantity.

3.4 Detail performance. -

3.4.1 Instruction repertoire. - The DPS instruction repertoire shall be as specified in table III. The instruction execution time shall be as specified in table III, with a tolerance of $\pm$ 10%, and shall be predicated upon a single-port memory with no indirect addressing. If the Direct Memory Access (DMA) option is implemented, the execution times for each instruction may increase 65 ns nominal. The execution times shall include all the time necessary for the instruction read, interpret, and operand read, as well as perform the instruction itself. The DPS shall perform the instruction repertoire in any sequence with any operand address or data word bit pattern without malfunction or erroneous result. When executing an instruction, the contents of R_a, R_m, and Y shall not be changed unless specified in the individual instruction. An attempt to execute any unassigned or unused instruction shall result in an instruction fault interrupt. The detailed individual instruction requirements shall be as specified in Appendix 30. The Condition Code usage shall be as specified in table IV. The symbols used in the instruction definitions shall be as specified in table V.

Table III. Instruction Repertoire

Operation Code	Format	Instruction	Execution Time (Microseconds)
00	RR	Diagnostic Return	0.7
	RI	Unassigned	-
	RK	Unassigned	-
	RX	Byte Load	2.25
01	RR	Load	.75
	RI-2	Load	1.5
	RK	Load	1.5
	RX	Load	2.25
02	RR	Unary-Arithmetic	1.0
	RI-2	Load Double	2.25
	RK	Unassigned	-
	RX	Load Double	3.0
03	RR	Unary-Control	0.9 → 8.0**
	RI	Unassigned	-
	RK	Unassigned	-
	RX	Load Multiple	1.5 +*N times.75

*N = Number of registers loaded.

**Execution times for 03RR, Unary-Control Instruction

m value	Execution time (microseconds)
0	8.0
1	0.9
2	0.9
3	0.9
4	1.2
5	1.65
6	0.9
7	0.9
10	1.2
11	1.2
12	1.35
13	0.9
14	1.5
15	1.5
16	0.9
17	0.9

Table III. Instruction Repertoire(continued)

Operation Code	Format	Instruction	Execution time (Microseconds)
*04	RR	Unary-Shift	3.3 → 8.5
	RI	Unassigned	-
	RK	Unassigned	-
	RX	Byte Load and Index by 1	2.25
05	RR	Set Bit	1.5
	RI-2	Load and Index by 1	1.5
	RK	Unassigned	-
	RX	Load and Index by 1	2.25
06	RR	Clear Bit	1.5
	RI-2	Load Double and Index by 2	2.55
	RK	Unassigned	-
	RX	Load Double and Index by 2	3.3

* Execution times for 04RR, Unary-Shift Instruction

m value	Execution time (microseconds)
0 (optional)	8.5
1	6.3
2	7.2
3	3.3
4-17	-

Table III. Instruction Repertoire (continued)

Operation Code	Format	Instruction	Execution Time (Microseconds)
07	RR	Compare Bit	1.8
	RI-2	Load PSW	3.0
	RK	Unassigned	-
	RX	Load PSW	3.75
10	RR	Logical Right Single Shift	1.0
	RI	Unassigned	-
	RK	Logical Right Single Shift	1.7
	RX	Byte Store	2.4
11	RR	Algebraic Right Single Shift	1.0
	RI-2	Store	1.7
	RK	Algebraic Right Single Shift	1.7
	RX	Store	2.4
12	RR	Logical Right Double Shift	2.6
	RI-2	Store Double	2.4
	RK	Logical Right Double Shift	3.2
	RX	Store Double	3.2
13	RR	Algebraic Right Double Shift	2.6
	RI	Unassigned	-
	RK	Algebraic Right Double Shift	3.2
	RX	Store Multiple	1.4 + *N x 1.1
14	RR	Algebraic Left Single Shift	1.0
	RI	Unassigned	-
	RK	Algebraic Left Single Shift	1.7
	RX	Byte Store and Index by 1	2.4

* N = number of registers stored

Table III. Instruction Repertoire (continued)

Operation Code	Format	Instruction	Execution Time (Microseconds)
15	RR	Circular Left Single Shift	1.0
	RI-2	Store and Index by 1	1.7
	RK	Circular Left Single Shift	1.7
	RX	Store and Index by 1	2.4
16	RR	Algebraic Left Double Shift	2.7
	RI-2	Store Double and Index by 2	2.6
	RK	Algebraic Left Double Shift	3.3
	RX	Store Double and Index by 2	3.3
17	RR	Circular Left Double Shift	2.4
	RI-2	Store Zeros	1.7
	RK	Circular Left Double Shift	3.0
	RX	Store Zeros	2.4
20	RR	Subtract	.75
	RI-2	Subtract	1.5
	RK	Subtract	1.5
	RX	Subtract	2.25
21	RR	Subtract Double	1.7
	RI-2	Subtract Double	2.25
	RK	Unassigned	-
	RX	Subtract Double	3.0
22	RR	Add	.75
	RI-2	Add	1.5
	RK	Add	1.5
	RX	Add	2.25

Table III. Instruction Repertoire (continued)

Operation Code	Format	Instruction	Execution Time (Microseconds)
23	RR	Add Double	1.5
	RI-2	Add Double	2.25
	RK	Unassigned	-
	RX	Add Double	3.0
24	RR	Compare	.90
	RI-2	Compare	1.5
	RK	Compare	1.7
	RX	Compare	2.25
25	RR	Compare Double	1.7
	RI-2	Compare Double	2.25
	RK	Unassigned	-
	RX	Compare Double	3.0
26	RR	Multiply	3.8
	RI-2	Multiply	4.0
	RK	Multiply	4.4
	RX	Multiply	4.6
27	RR	Divide	6.8
	RI-2	Divide	7.0
	RK	Divide	7.4
	RX	Divide	7.5
30	RR	AND	.75
	RI-2	AND	1.5
	RK	AND	1.5
	RX	AND	2.25
31	RR	OR	.75
	RI-2	OR	1.5
	RK	OR	1.5
	RX	OR	2.25
32	RR	Exclusive OR	.75
	RI-2	Exclusive OR	1.5
	RK	Exclusive OR	1.5
	RX	Exclusive OR	2.25

Table III. Instruction Repertoire (continued)

Operation Code	Format	Instruction	Execution Time (Microseconds)
33	RR	Masked Substitute	1.4
	RI-2	Masked Substitute	1.5
	RK	Masked Substitute	2.0
	RX	Masked Substitute	2.25
34	RR	Compare Masked	1.5
	RI-2	Compare Masked	1.7
	RK	Compare Masked	2.1
	RX	Compare Masked	2.4
35	RR	I/O Command	4.0* + I/O Inst
	RI-2	Biased Fetch	2.25
	RK	Remote Execute	1.5 + Inst
	RX	Biased Fetch	3.0
36	RR	Unassigned	
	RI	Unassigned	
	RK	Unassigned	
	RX	Unassigned	
37	RR	Trigonometric	11.8 (without Prescale)
		- Hyperbolic**	15.3 (with Prescale)
	RI	Unassigned	
	RK	Unassigned	
	RX	Unassigned	
40	RR	Conditional Jump	1.1
	RI-1	Local Jump	1.2
	RK	Conditional Jump	1.7
	RX	Conditional Jump	2.4
41	RR	Index Jump	1.4
	RI-1	Local Jump Indirect	2.0
	RK	Index Jump	2.1
	RX	Index Jump	2.25
42	RR	Jump and Link Register	1.2
	RI	Unassigned	-
	RK	Jump and Link Register	1.2
	RX	Jump and Link Register	2.25

* Includes the time to clear bits 14 and 15 of memory address 000140.
** Optional.

Table III. Instruction Repertoire (continued)

Operation Code	Format	Instruction	Execution Time (Microseconds)
43	RR	Unassigned	-
	RI-1	Local Jump and Link Memory	2.0
	RK	Jump and Link Memory	2.9
	RX	Jump and Link Memory	3.2
44	RR	Jump Register = 0	1.4
	RI-1	Local Jump Equal	1.2
	RK	Jump Register = 0	2.1
	RX	Jump Register = 0	2.25
45	RR	Jump Register ≠ 0	1.4
	RI-1	Local Jump Not Equal	1.2
	RK	Jump Register ≠ 0	2.1
	RX	Jump Register ≠ 0	2.25
46	RR	Jump Register Positive	1.4
	RI-1	Local Jump Greater Than or Equal	1.2
	RK	Jump Register Positive	2.1
	RX	Jump Register Positive	2.25
47	RR	Jump Register Negative	1.4
	RI-1	Local Jump Less Than	1.2
	RK	Jump Register Negative	2.1
	RX	Jump Register Negative	2.25
50	RR	Floating Point Subtract*	7.7 — 17.4**
	RI	Floating Point Subtract*	7.7 — 17.4**
	RK	Unassigned	-
	RX	Floating Point Subtract*	7.7 — 17.4**
51	RR	Floating Point Add*	7.7 — 17.4**
	RI	Floating Point Add*	7.7 — 17.4**
	RK	Unassigned	-
	RX	Floating Point Add*	7.7 — 17.4**
52	RR	Floating Point Multiply*	15.2 — 18.9**
	RI	Floating Point Multiply*	15.2 — 18.9**
	RK	Unassigned	-
	RX	Floating Point Multiply*	15.2 — 18.9**

* Optional
** Execution time of this instruction is variable depending on the operand.

Table III. Instruction Repertoire (continued)

Operation Code	Format	Instruction	Execution Time (Microseconds)
53	RR	Floating Point Divide*	7.7 — 17.7**
	RI	Floating Point Divide*	7.7 — 17.7**
	RK	Unassigned	-
	RX	Floating Point Divide*	7.7 — 17.7**
54	RR	Load Page Register	1.8
	RI-2	Load Page Register	2.6
	RK	Unassigned	-
	RX	Load Page Register Multiple	3.0 + .75 x n***
55	RR	Store Page Register	1.8
	RI-2	Store Page Register	2.6
	RK	Unassigned	-
	RX	Store Page Register Multiple	3.0 + 1.1 x n***
56	RR	Double Precision Multiply*	5.5 — 15.3**
	RI	Double Precision Multiply*	5.5 — 15.3**
	RK	Unassigned	-
	RX	Double Precision Multiply*	5.5 — 15.3**
57	RR	Double Precision Divide*	17.6 — 21.0**
	RI	Double Precision Divide*	17.6 — 21.0**
	RK	Unassigned	-
	RX	Double Precision Divide*	17.6 — 21.0**
60	RL-1	Logical Right Single Shift	1.3
	RL-2	Algebraic Right Single Shift	1.3
	RL-3	Logical Right Double Shift	2.8
	RL-4	Algebraic Right Double Shift	2.8
61	RL-1	Algebraic Left Single Shift	1.3
	RL-2	Circular Left Single Shift	1.3
	RL-3	Algebraic Left Double Shift	2.8
	RL-4	Circular Left Double Shift	2.8
62	RL-1	Subtract	.9
	RL-2	Subtract Double	1.8
	RL-3	Add	.9
	RL-4	Add Double	1.8

* Optional
** Execution time of this instruction is variable depending on the operand.
***n = Number of address registers.

Table III. Instruction Repertoire (continued)

Operation Code	Format	Instruction	Execution Time (Microseconds)
63	RL-1	Load	.9
	RL-2	Compare	1.2
	RL-3	Multiply	4.4
	RL-4	Divide	7.4
64	RR	Unassigned	
	RI	Unassigned	
	RK	Unassigned	
	RX	Byte Subtract	2.25
65	RR	Unassigned	
	RI	Unassigned	
	RK	Unassigned	
	RX	Byte Add	2.25
66	RR	Unassigned	
	RI	Unassigned	
	RK	Unassigned	
	RX	Byte Compare	2.25
67	RR	Reserved for user growth otherwise unassigned	
	RI	Reserved for user growth otherwise unassigned	
	RK	Reserved for user growth otherwise unassigned	
	RX	Byte Compare and Index by 1	2.25
70	RR	Channel Control (Command)	30.0 for m=0-7; 2.0 for m=10-17
	RR	Channel Control (Chaining)	2.25
	RI	Unassigned	
	RK	Unassigned	
	RX	Initiate Transfer (Chaining)	4.5

Table III. Instruction Repertoire (continued)

Operation Code	Format	Instruction	Execution Time (Microseconds)
71	RR	Unassigned	-
	RI	Unassigned	-
	RK	Initiate Chain (Command)	2.25
	RK	Load Control Memory (Chaining)	2.25
	RX	Load Control Memory (Command)	3.0
	RX	Load Control Memory (Chaining)	3.0
72	RR	Unassigned	-
	RI	Unassigned	-
	RK	Unassigned	-
	RX	Store Control Memory (Command)	3.0
	RX	Store Control Memory (Chaining)	3.0
73	RR	Halt/Interrupt (Chaining)	1.5
	RI	Unassigned	-
	RK	Unassigned	-
	RX	Set/Clear Flag (Chaining)	3.0
74	RR	Unassigned	-
	RI	Unassigned	-
	RK	Conditional Jump (Chaining)	2.25
	RX	Unassigned	-
75	RR	Search for Sync;Set Monitor/Set Suppress (Chaining)	1.5*
	RI	Unassigned	-
	RK	Unassigned	-
	RX	Unassigned	-
76	RR	Set/Clear Discretes (Command)	1.5
	RR	Set/Clear Discretes (Chaining)	1.5
	RI	Unassigned	-
	RK	Unassigned	-
	RX	Store Status (Command)	3.0
	RX	Store Status (Chaining)	3.0
77	RR	Unassigned	-
	RI	Unassigned	-
	RK	Unassigned	-
	RX	Unassigned	-

* The time is for Set Monitor and Set Suppress. The time for Search for Sync depends on the modulation rate of the interface.

Table IV. Condition Code Designator Functions

Combined Value		Arithmetic Operation	Compare Operation 24,25,34,63 RL-2, 66RX, 67RX
Bit 9	Bit 8		
0	0	Zero (positive)	$(R_a)=(R_m)$ or (Y)
0	1	Not zero and positive	$(R_a)>(R_m)$ or (Y)
1	0	Not used	Not used
1	1	Not zero and negative	$(R_a)<(R_m)$ or (Y)

Table V. Symbol Definitions

Symbol	Description
a	The a-designator from instruction words.
R_a	The register designated by a.
m	The m-designator from instruction words.
R_m	The register designated by m.
Y	The operand or memory address generated in the execution of an instruction.
y	The contents of the second word of an RK or RX instruction.
P	The Program Address register.
()	The contents of the location specified within the parenthesis.
d	The deviation value in a local jump instruction.
Truth Tables	Exclusive OR: 0 0 1; 1 1 0 — Logical OR: 0 0 1; 1 1 1 — Logical AND: 0 0 0; 1 0 1

3.4.2 Instruction words. - The DPS shall perform instructions using five instruction word formats, and operations using 4-bit literals, 8-bit bytes, 16-bit words, and 32-bit double length words as specified in the following subparagraphs.

3.4.2.1 Format RR. - The Format RR instructions shall use the format as specified in figure 2. The Format RR instructions shall perform operations using the general registers. Unless otherwise specified in an individual instruction, the a and m-designators shall be used as follows:

a. The a-designator shall select the R_a register.

b. The m-designator shall select the R_m register.

3.4.2.2 Format RI. - The Format RI instructions shall be either Type 1 or Type 2, and shall be as specified in the following subparagraphs.

3.4.2.2.1 Format RI Type 1. - The Format RI Type 1 instructions shall use the format as specified in figure 4. The memory address Y shall be generated by adding to (P) the 16-bit value specified in figure 5.

3.4.2.2.2 Format RI Type 2. - The Format RI Type 2 instructions shall use the format as specified in figure 2. The Format RI Type 2 instructions shall perform operations using the computer general registers and a memory reference. Unless otherwise specified in an individual instruction, the a and m-designators shall be used as follows:

a. The a-designators shall select the R_a register.

b. The m-designator shall select the R_m register whose contents shall be used as a memory address Y.

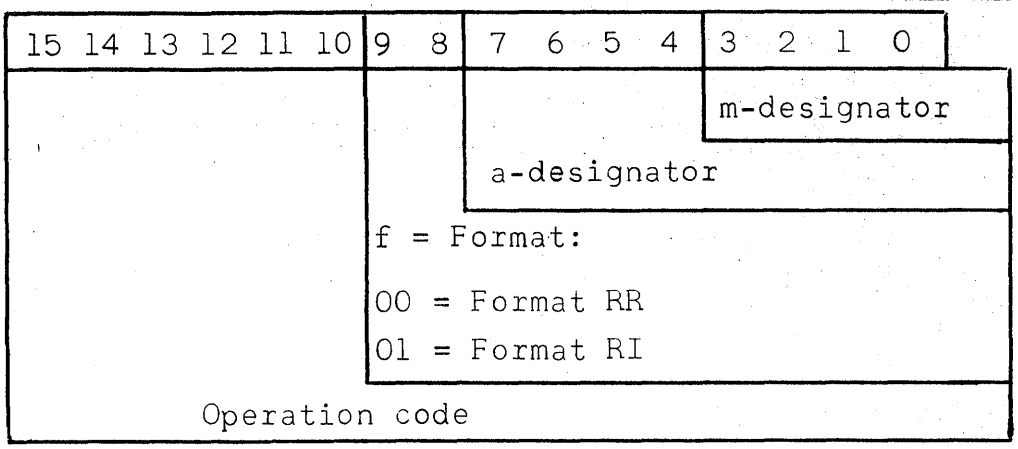


Figure 2. Instruction Word Format for Formats RR and RI Type 2.

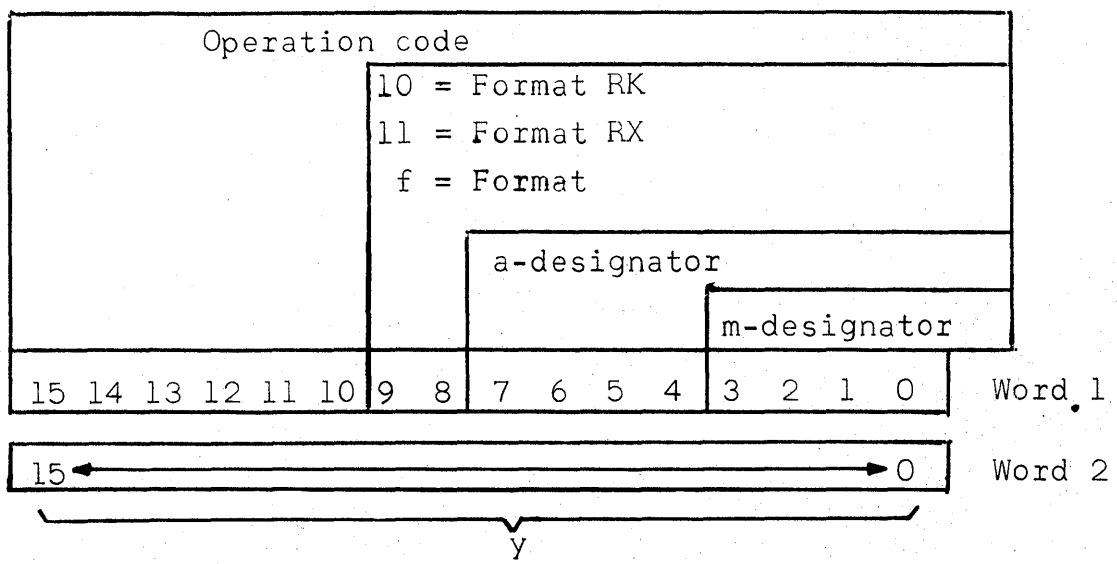


Figure 3. Instruction Word Format for Formats RK and RX

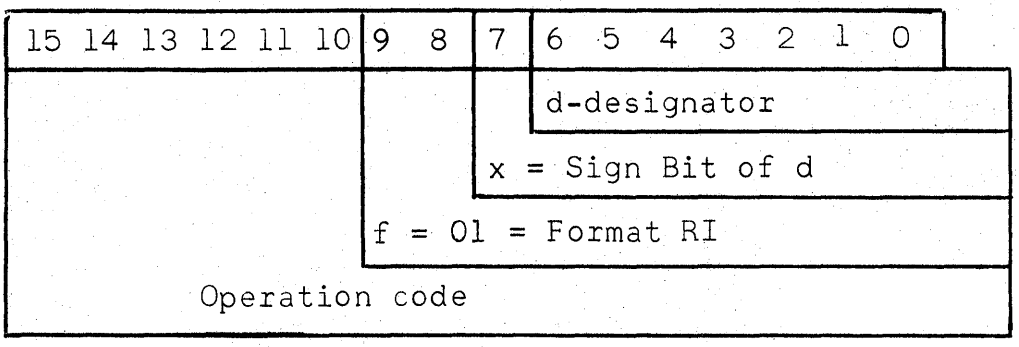


Figure 4. Instruction Word Format for Format RI Type 1.

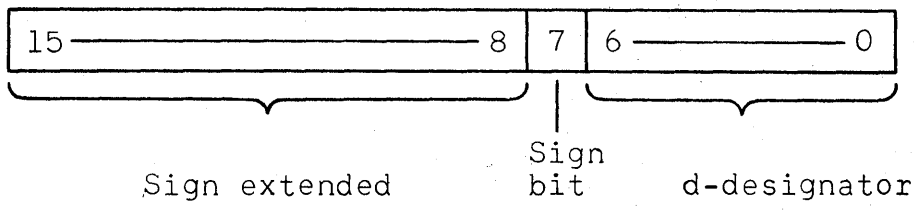


Figure 5. Address Modifier in Format RI Type I Instructions.

3.4.2.3 Format RK. - Each Format RK instruction shall consist of two words which shall be required to be stored in two consecutive memory addresses. The words shall use the format as specified in figure 3. The second word shall be a 16-bit quantity designated y. The Format RK instructions shall perform operations using the general registers and memory references. Unless otherwise specified in an individual instruction, the a, m, and y-designators shall be used as follows:

a. The a-designator shall select the R_a register.

b. The m and y-designators shall be used to form an operand or a memory address designated Y. Y shall be used as specified in the individual instruction. Y shall be formed as follows:

(1) When m equals zero, Y shall be equal to y.

(2) When m does not equal zero, Y shall be the sum of the contents of R_m and y. The m-designator shall select the R_m register.

3.4.2.4 Format RX. - The Format RX instructions shall consist of two words which shall be required to be stored in two consecutive memory addresses. The words shall use the format as specified in figure 3. The second word shall be a 16-bit quantity designated y. The Format RX instructions shall perform either double-word (32-bit), whole word (16-bit) or byte (8-bit) operations using the DPS general registers and memory references as specified in the following subparagraphs.

3.4.2.4.1 Format RX whole word operations. - The DPS shall have indirect addressing for instruction format RX using R_m 10, 12, 14, or 16. When the format RX instruction designates whole word operations, the a, m, and y-designators, unless otherwise specified in an individual instruction, shall be used as follows:

a. The a-designator shall select the R_a register.

b. The m and y designators shall be used to form the memory address Y:

(1) When m equals zero, Y shall be equal to y.

(2) When m is equal to 1 through 7_8, 11_8, 13_8, 15_8, or 17_8, Y shall be the sum of the (R_m) and y.

(3) When m is equal to 10_8, 12_8, 14_8, 16_8 and the indirect control bits of Status Register #2 corresponding to the particular R_m register are equal to 00_2 or 01_2, Y shall be the sum of (R_m) and y.

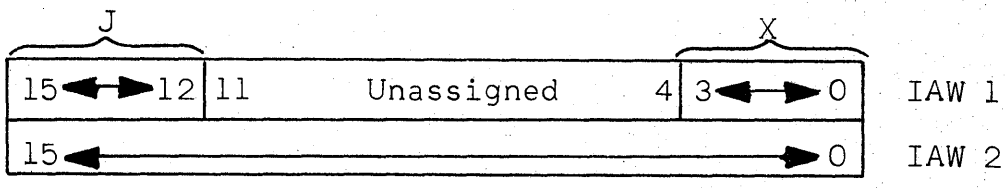
(4) When m equals 10_8, 12_8, 14_8, or 16_8, and the indirect control bits of Status Register #2 corresponding to the particular Rm register are equal to 10_2, the address of indirect word 1 shall be equal to y. The indirect address words shall be determined in accordance with figure 6.

(5) When m equals 10_8, 12_8, 14_8, or 16_8 and the 2-bit field of Status Register #2 corresponding to the particular R_m register are equal to 11_2, the address of indirect word 1 shall be equal to the sum of y + (R_m). The indirect address words shall be determined in accordance with figure 6.

3.4.2.4.1.1 Indirect addressing. - Addressing shall be under control of a pair of indirect address words (IAW) as specified in figure 6. For J equal to 0, 1, 2, or 3 the final address shall be as specified.

For J equal to 4, 5, 6, or 7; IAW 2, modified as specified by J, shall be the address of IAW 1 of the next IAW pair. This cascading shall continue until J equals 0, 1, 2, or 3 in a referenced IAW.

The pairs of IAW's shall be required to be located in sequential memory addresses, with IAW 1 at an even address.



J Value	Address determination
0	Final operand at address specified by IAW 2 and if byte mode, upper byte is used.
1	*Final operand at address specified by IAW 2 + (R_x) and if byte mode, the LSB of R_x determines the byte (see 3.4.2.4.2).
2	*Final operand at address specified by IAW 2 + (R_m) and if byte mode, the LSB of R_m determines the byte (3.4.2.4.2).
3	*Final operand at address specified by IAW 2 + (R_{m+1}). and if byte mode, the LSB of R_{m+1} determines the byte (see 3.4.2.4.2).
4	Cascaded IAW at address specified by IAW 2.
5	Cascaded IAW at address specified by IAW 2 + (R_x).
6	Cascaded IAW at address specified by IAW 2 + (R_m).
7	Cascaded IAW at address specified by IAW 2 + (R_{m+1}).
10-17	Unassigned.

Figure 6. Indirect Address Words

* To determine the operand address when in byte mode, the contents of R_x, R_m, or R_{m+1} shall be right shifted 1 bit position and zero filled in the left most position.

3.4.2.4.2 Format RX byte operations. - When the Format RX instruction designates byte operations, the a, m, and y-designators, unless otherwise specified in an individual instruction, shall be used as follows:

a. The a-designator shall select the R_a register.

b. When m equals 1-17$_8$, the m-designator shall be used to select a general register designated R_m. The operation shall be as follows:

(1) Unless otherwise specified under indirect addressing, the least significant bit (LSB) of the contents of R_m (bit 0) shall determine the byte position in the selected memory location:

(a) When the LSB is 0, the byte shall be the eight most significant bits (bits 8 through 15) in the memory location.

(b) When the LSB is 1, the byte shall be the eight least significant bits (bits 0 through 7) in the memory location.

(2) Unless otherwise specified under indirect addressing, Y shall be the sum of y and the contents of R_m right shifted one position and zero filled in the left most position. The original value of R_m shall remain in R_m.

c. When m equals zero, the contents of R_m shall not be added to y to form the memory address Y. Y shall be equal to y and the byte shall be the eight most significant bits (bits 8 through 15) in the memory location.

3.4.2.5 Format RL. - The format RL instructions shall use the format as specified in figure 7. Unless otherwise specified in the individual instructions, the a and m designator usage shall be:

a. The a-designator shall select the R_a register.

b. The m-designator shall be a 4-bit, unsigned literal.

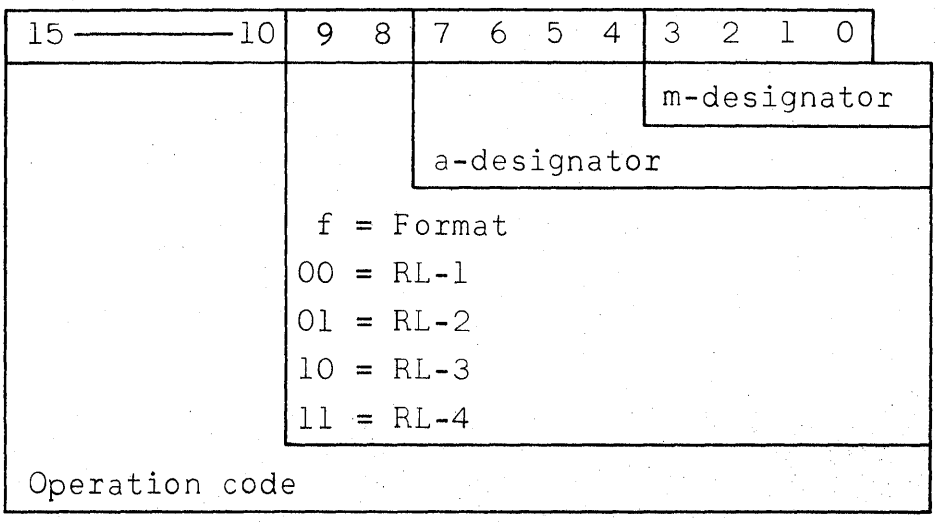


Figure 7. Instruction Word Format for Format RL.

3.4.2.6 Double length operations. - The DPS shall perform double length word operations when specified in an individual instruction. In these instructions, the contents of two adjacent registers or memory locations shall be used as one 32-bit word. The most significant 16-bits of the double length operand shall be designated R_a, R_m, or Y. The least significant 16 bits shall be designated R_{a+1}, R_{m+1} or Y + 1 respectively. R_a, R_m, or Y shall be required to be an even number. For double length (32-bit) words, the sign bit shall be bit 15 of the most significant 16 bits (R_a, R_m. or Y).

3.4.2.7 Shift instruction execution. - Shift instructions shall shift the contents of a register or registers to the right, left, or left circular. The number of shifts executed (places shifted) shall be the contents of bits 0 through 5 of the quantity specified in the individual instructions. The shift operations shall be as follows:

a. Right shift shall mean that the contents of the specified register shall be shifted right, toward the least significant bit position. Bits shifted out of the least significant bit position shall be lost. After each shift, the most significant bit position shall be filled with either a zero (zero extended to fill) or a sign bit (sign extended to fill) as specified in the individual instruction.

b. Left shift (algebraic) shall mean that the contents of the specified register (including sign bit) shall be shifted left, toward the most significant bit position. Bits shifted out of the most significant bit position shall be lost. After each shift, the least significant bit position shall be filled with a zero. When the value of the most significant bit changes during the shift operation, the overflow designator shall be set.

c. Left shift circular shall mean that the contents of the specified register shall be shifted left, and for each shift the bit shifted out of the most significant bit position shall be transferred to the least significant bit position.

3.4.3 Relative addressing. - All DPS main memory references (including I/O data references, allocated memory addresses, P and Y) shall be relative to a fixed base. Generation of absolute addresses shall be as specified in figure 8. The fixed base shall be contained in a page register having the format specified in figure 9. The computer shall provide 64 page registers. The page registers shall be under program control (instructions 54 and 55). The DPS shall set bit 15 of the page register when the contents of any of the 1024 addresses in the page (see 6.2) are modified. A master clear shall clear bit 15 of all page registers and set bits 5-0 of each page register equal to the page register address.

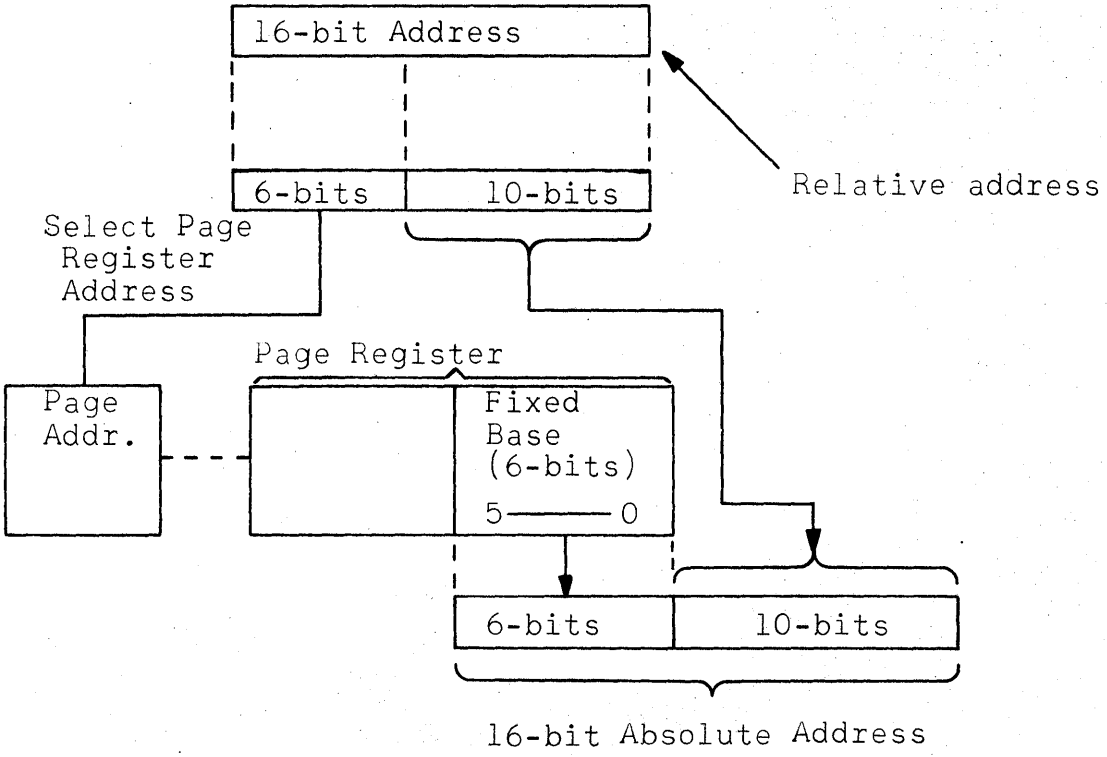


Figure 8. Relative Addressing

15	14 ◄——► 6	5 ◄——► 0
	Not addressable	Fixed base

0 = Page has not been modified
1 = Page has been modified

Figure 9. Page Register

3.4.4 Memories. -

3.4.4.1 Main memory. - The DPS main memory shall be random access, nonvolatile magnetic core storage with allocated addresses as specified in table VI. Memory addresses unassigned in table VI may be used as required in programs. The main memory shall be either a single-port (CP) memory or a two-port memory with a direct memory access (DMA). The Memory type, specified in the procurement document (see 6.1), may be modified at the installation site to provide DMA or DMA may be removed if it was installed in the factory. The two-port memory shall provide one CP port and one direct memory access (DMA) port (external) as specified in Appendix 20.

3.4.4.1.1 Word length. - The DPS main memory shall have a word length of 16-bits.

3.4.4.1.2 Cycle time. - The read-restore cycle time of the main memory shall be 0.75 ± 10% microseconds.

3.4.4.1.3 Capacity. - Main memory capacity shall be as follows:

a. Word length: 16 Data Bits

b. Size: Up to 65,536 (16-bit) addresses in 8,192 address increments. The main memory size shall be specified by the procurement document (see 6.1).

3.4.4.1.4 Addressing. - All words of main memory shall be contiguously addressable. The DPS addressing capability shall be 131,072 8-bit, 65,536 16-bit, and 32,768 32-bit words.

3.4.4.1.5 Protection. - Main memory shall be protected against any loss of contents due to a power failure.

3.4.4.1.6 Two-port priority. - Requests at the CP port shall have priority over requests at the DMA port.

3.4.4.2 NDRO memory (bootstrap). - The DPS shall have a non-destructive readout (NDRO) memory. The NDRO memory shall have 192 words at addresses 00_8-77_8 and 300_8-477_8. The addresses shall contain a sequence of instructions for program load operation (bootstrap). Access to the NDRO shall be controlled by the condition of the NDRO Mode bit of Status Register 1. The contents of the NDRO memory shall not be altered by write references. Except that the entire program shall be changeable by replacing the NDRO memory assembly, the NDRO memory contents shall be programmed at time of manufacture and shall not be changeable. The NDRO memory contents shall be required to be specified in the procurement document (see 6.1). Address 000 of NDRO memory is reserved for the AUTO START function, and contains a jump to the address specified by the contents of main memory address 177_8. Address 002 is reserved for bootstrap start.

3.4.4.3 Microprogrammed read-only memory (ROM). - The processor shall use a microprogrammed control structure with the microinstructions and control data stored in a microprogrammed ROM. The microprogrammed control structure shall provide the full set of macroinstructions listed in the instruction repertoire. Microprogrammed instructions shall be as specified in Specification SB-10592.

3.4.4.3.1 User-defined microprogram. - 512 words of microprogrammed ROM shall be available as an option. The optional ROM and microprogram shall be specified by the procurement document (see 6.1).

3.4.4.3.2 Diagnostic microprogram. - The DPS shall have up to 1024 words of 16-bit microprogrammed ROM to aid in diagnosing failures.

3.4.4.3.3 Math pack. - The DPS shall have an optional 1024 words of 16-bit microprogrammed ROM for math pack algorithms. These algorithms shall consist of the following functions:

a. Square root.

b. Trigonometric and hyperbolic (cordic).

c. Floating point add, subtract, multiply, and divide.

d. Double precision multiply and divide.

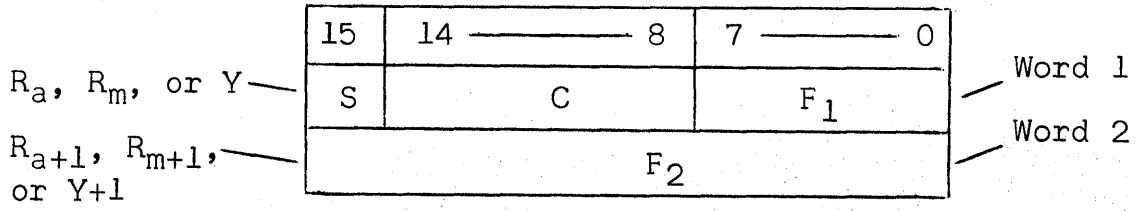
Operation of these functions as macroinstructions shall be as specified in Appendix 30. When using R_a, R_{a+2}, R_m, and Y, in any of the math pack operations, these parameters shall be required to be located in even numbered general registers and an even numbered memory address. The math pack option shall be required to be specified in the procurement document (see 6.1).

3.4.4.3.3.1 Square root operation. - The square root operation shall be implemented by the unary-shift instruction, operation code 04 RR, when the m value is equal to 0 and shall perform as specified in table 30-III.

3.4.4.3.3.2 Trigonometric and hyperbolic functions. - The trigonometric and hyperbolic functions shall provide a coordinate conversion (CORDIC) capability and shall operate as specified in Appendix 50. The trigonometric and hyperbolic functions shall be implemented by operation code 37 RR as specified in Appendix 30.

3.4.4.3.3.3 Floating point instructions. - The floating point instructions, subtract, add, multiply, and divide shall be implemented by operation codes 50, 51, 52, and 53 and shall perform as specified in Appendix 30.

All floating point operations shall be capable of being performed with round or with residue and shall use a double length format as follows:



Word 1 shall contain the following fields:

S = the sign of the fraction

0 = positive

1 = negative

C = the biased representation of the characteristic

F_1 = the two most significant digits of the fraction

Word 2 shall contain the four least significant digits of the fraction (F_2).

The fraction of the floating point number shall be represented in 4-bit hexadecimal digits and shall always be a magnitude number. The binary point shall be between the right most bit of C and the left most bit of F_1.

C shall be a seven bit binary quantity equal to the characteristic plus 64_{10}. The approximate decimal range covered by the magnitude (M) of a normalized floating point number shall be

$$5.4 \times 10^{-79} \leq M \leq 7.2 \times 10^{+75}$$

The exact hexadecimal range covered by the magnitude (M) of a normalized floating point number shall be

$$.100000_{16} \times 16^{-64} \leq M \leq .FFFFFF_{16} \times 16^{+63}$$

3.4.4.3.3.3.1 Normalization. - A floating point number shall be normalized when the most significant hexadecimal digit is non-zero. The results of all floating point operations (excluding the residue) shall be normalized if the inputs were normalized.

3.4.4.3.3.3.2 Floating point zero. - A floating point zero shall be represented by a zero sign, zero characteristic, and zero fraction.

3.4.4.3.3.3.3 Floating point round. - Floating point rounding shall be performed on floating point add, subtract, multiply and divide operations contingent upon bit 6 of Status Register 1. If bit 6 is zero, rounding shall occur. When rounding is not specified (bit 6 of Status Register 1 is 1), the DPS shall form a floating point number of the unused digits in R_{a+2} and R_{a+3}, as specified in the individual instruction. The rounding action shall add 1 to the result (R_a,R_{a+1}) if the unused digits are equal to or greater than half the least-significant digit. No action shall occur if less than half.

3.4.4.3.3.3.4 Floating point residue. - The floating point residue shall be defined as follows:

a. As a result of the floating point add, subtract, and multiply operations; the residue shall represent the least significant bits of the result that cannot be contained within the 24 bit (6 hexadecimal digits) fraction result. The residue shall have the same sign as the result .

b. As a result of a floating point divide operation, the residue shall represent the remainder and shall have the same sign as the dividend.

3.4.4.3.3.3.5 Characteristic underflow and overflow. - A floating point operation shall cause a characteristic underflow condition whenever a number smaller than $.100000_{16} \times 16^{-64}$ is generated as part of a result. A floating point operation shall cause a characteristic overflow condition whenever a number larger than $.FFFFFF_{16} \times 16^{+63}$ is generated as part of the result. No underflow condition shall occur in forming a residue. If a residue number smaller than $.100000_{16} \times 16^{-64}$ is generated, the residue value shall be all zeros. When a floating point operation causes a characteristic underflow or overflow, the DPS shall:

a. Inhibit storing the erroneous result in the general registers.

b. Modify the Condition Code of Status Register 1.

c. Test bit 7 and bit 2 of Status Register 1. If bit 7 is a 1, no interrupt shall occur; if bit 7 is a 0 and bit 2 is a 1, the floating point interrupt shall be set and processed; if bit 7 is a 0 and bit 2 is a 0, the interrupt is lost.

The original contents of the general registers and memory addresses Y, Y+1 shall remain unchanged.

3.4.4.3.3.4 Double precision multiply instruction. - The double multiply instruction shall be implemented by operation code 56 and shall perform as specified in Appendix 30.

3.4.4.3.3.5 Double precision divide instruction. - The double divide instruction shall be implemented by operation code 57 and shall perform as specified in Appendix 30.

Table VI. Main Memory Address Allocation

Relative Address	Function
000000	Unassigned
000107	
000110	CP class III interrupt address for Store P
000111	CP class III interrupt address for Store Status # 1
000112	CP class III interrupt address for Store Status # 2
000113	CP class III interrupt address for Store RTC Lower
000114	CP class III interrupt address for Load P
000115	CP class III interrupt address for Load Status # 1
000116	CP class III interrupt address for Load Status # 2
000117	CP class III interrupt address for Store RTC Upper
000120	CP class II interrupt address for Store P
000121	CP class II interrupt address for Store Status # 1
000122	CP class II interrupt address for Store Status # 2
000123	CP class II interrupt address for Store RTC Lower
000124	CP class II interrupt address for Load P
000125	CP class II interrupt address for Load Status # 1
000126	CP class II interrupt address for Load Status # 2
000127	CP class II interrupt address for Store RTC Upper
000130	CP class I interrupt address for Store P
000131	CP class I interrupt address for Store Status # 1
000132	CP class I interrupt address for Store Status # 2

Table VI. (Continued)

Relative Address	Function
000133	CP class I interrupt address for Store RTC Lower
000134	CP class I interrupt address for Load P
000135	CP class I interrupt address for Load Status # 1
000136	CP class I interrupt address for Load Status # 2
000137	CP class I interrupt address for Store RTC Upper
000140	IO Command Cell Location 1
000141	IO Command Cell Location 2
000142 thru 000176	Unassigned
000177	Absolute address used with Auto Start Function
000200	Channel 0 EI Interrupt Storage
000201	Channel 1 EI Interrupt Storage
000202	Channel 2 EI Interrupt Storage
000203	Channel 3 EI Interrupt Storage
000204	Channel 4 EI Interrupt Storage
000205	Channel 5 EI Interrupt Storage
000206	Channel 6 EI Interrupt Storage
000207	Channel 7 EI Interrupt Storage
000210	Channel 10 EI Interrupt Storage
000211	Channel 11 EI Interrupt Storage
000212	Channel 12 EI Interrupt Storage
000213	Channel 13 EI Interrupt Storage
000214	Channel 14 EI Interrupt Storage
000215	Channel 15 EI Interrupt Storage
000216	Channel 16 EI Interrupt Storage
000217	Channel 17 EI Interrupt Storage
000220 thru 177777	Unassigned

3.4.5 I/O channels. -

3.4.5.1 I/O channel characteristics. - The DPS shall provide asynchronous parallel I/O channels, expandable in groups of four channels (0-3, 4-7, etc.), to a possible total of 16 parallel I/O channels. Except that the NTDS serial channels shall be as specified in 3.4.5.4.3, the DPS shall provide serial channels, expandable in groups of two channels (0-1, 2-3, etc.), to a possible total of 16 channels. A mixture of both types shall be possible, within the constraints specified herein, to a total of 16 channels. Channels in any incremental group, both serial and parallel, shall be of only one type as specified in the following paragraphs and in the procurement document (see 6.1).

3.4.5.2 I/O control memory. - Each Input/Output channel shall have 16 words of program addressable control memory assigned as specified in figure 10.

3.4.5.2.1 BCW definitition. - The contents of the fields of the Buffer Control Word (BCW) control memory locations shall define the parallel and serial data transfers on the corresponding channel, as specified in the following subparagraphs.

3.4.5.2.1.1 Transfer mode (TM). - These bits shall specify I/O channel word length for external function, input data, and data transfers, as specified in figure 11.

Location Within Channel	Register	
0	Buffer Word Control	Input BCW
1	Buffer Address Pointer	
2	Input Chain Address Pointer	
3	Unassigned	
4	Buffer Word Control	Output BCW
5	Buffer Address Pointer	
6	Output Chain Address Pointer	
7	Unassigned	
10	Monitor Register	
11	Suppress Register	
12	Serial Mode Information	
13-17	Unassigned	

Figure 10. I/O Control Memory Locations

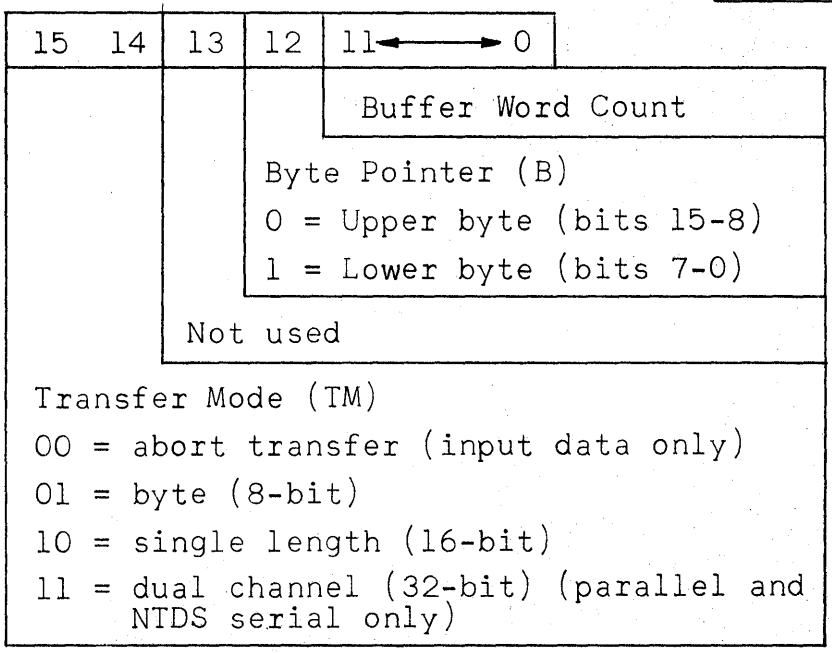


Figure 11. Buffer Word Control Format

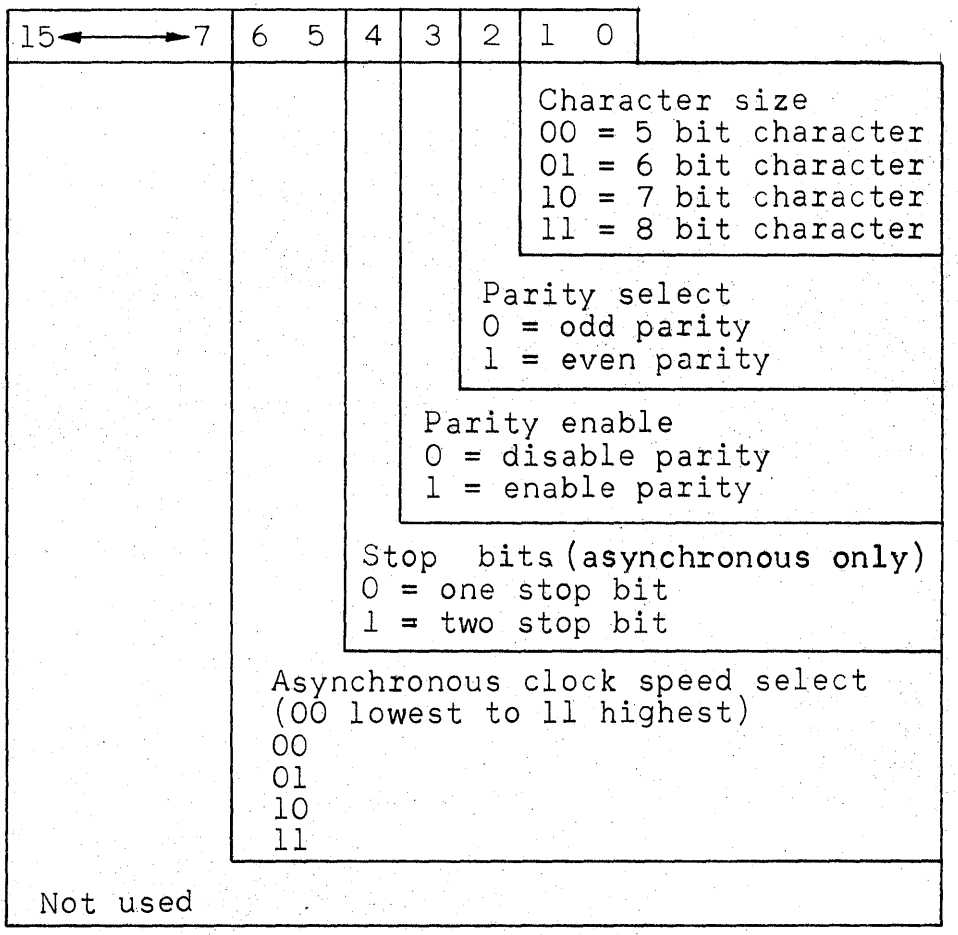


Figure 12. Serial Mode Information

3.4.5.2.1.2 Byte pointer (B). - When performing 8-bit (byte) transfers, this bit shall specify which byte of the memory location specified by the BCW shall be used for the next transfer, as specified in figure 11. This bit shall change state after each data transfer, but shall only be interpreted for byte transfers. All bytes transferred on the I/O channels shall be in data bits 7-0 of the channel interface.

3.4.5.2.1.3 Buffer word count. - These bits shall specify the number of bytes, single length words, or double length words to be transferred during the selected input data, output data, or output external function operations. The contents of the buffer word count shall be decremented by one for each transfer. An initial buffer word count of zero (0000) shall specify a maximum number of transfers (4096). An initial buffer word count not equal to zero shall specify the number of transfers (001 = 1 transfer). The buffer shall terminate when the buffer word count has been decremented to zero.

3.4.5.2.1.4 Buffer address pointer. - This word shall specify the memory address for the next input data, output data, or output external function transfer. For byte operation, the contents shall be incremented by one whenever the byte pointer bit changes from 1 to 0. For single length operation, the contents shall be incremented by one for each transfer. For double length operation the contents shall be incremented by 2.

3.4.5.2.2 Serial mode information. - The contents of the fields of the serial mode information control memory locations shall define the serial data transfers on the corresponding channel as specified in figure 12 for MIL-STD-188 or RS-232, and as specified in figure 13 for Variable Character Length Synchronous (VACALES) mode.

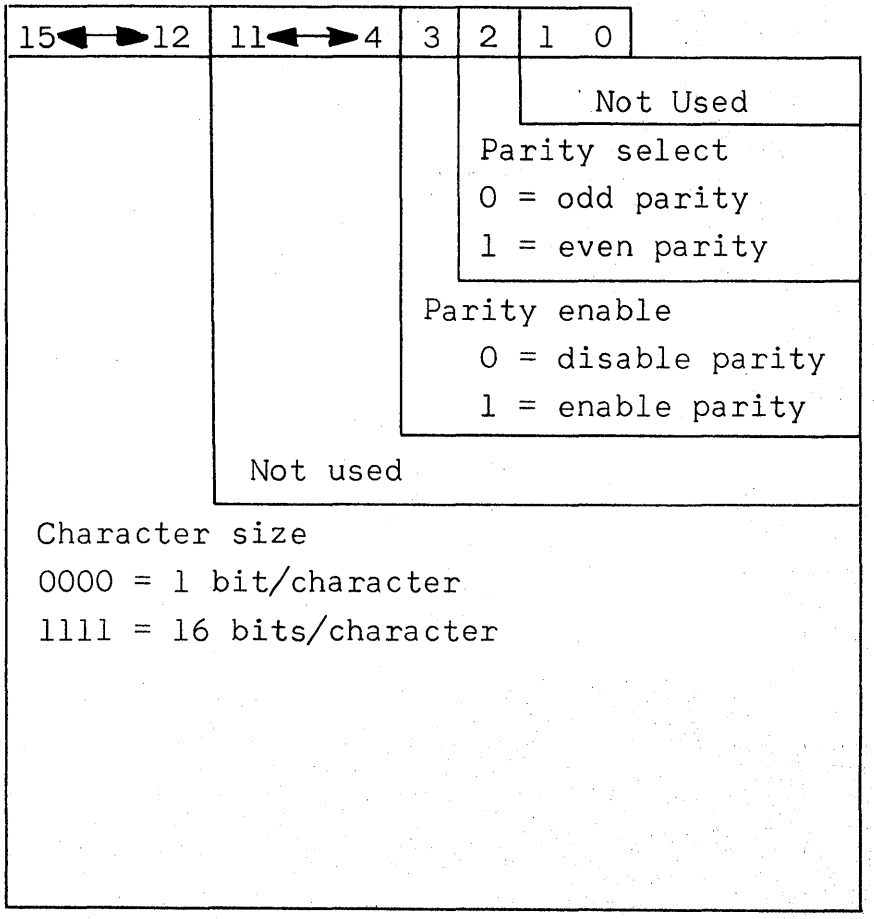


Figure 13. VACALES Mode Format

3.4.5.3 Parallel I/O channels. - The parallel I/O channels shall be a pluggable option, and shall operate in duplex (full duplex) mode which shall allow transmission in both directions simultaneously. The parallel I/O channels shall operate in single (16-bit) channel or dual (32-bit) channel modes. Both single and dual channels shall be capable of operation in the intercomputer or normal modes. Dual channels shall be capable of operation in the Externally Specified Addressing (ESA) mode. Parallel I/O channel transfer operations shall be in accordance with the sequence specified in Specification DS 4772. Input data, output data, and external function transfers shall be specified by the currently active BCW for an I/O channel. External interrupt inputs shall be stored at assigned memory addresses. Each parallel I/O channel shall have the interface lines as specified in figure 14. The interface shall be the -3V, -15V, or +3.5V type in accordance with Specification DS 4772 and as specified by the procurement document (see 6.1). The parallel data transfer rate shall be as specified in table VII. The parallel channels shall be under control of the I/O control memory words.

Table VII. Parallel I/O Channel Transfer Rates

Number of Active Channels - * Single ** Dual	Input or Output Transfer Rates (16-bit wds/sec or 8-bit wds/sec when in byte mode)		Input or Output Transfer Rates (32-bit wds/sec)	
	Normal Channel and IC Channel -3V, +3.5V (DS 4772) Interface	Normal Channel, IC Channel, and PIC -15V (DS 4772) Interface	Dual Channel, ESA Channel, and 32-bit IC Channel -3V, +3.5V (DS 4772) Interface	Dual Channel, ESA Channel, and 32-bit IC Channel -15V (DS 4772) Interface
* 1-4 (1 group)	190,000	41,600		
* 5-8 (2 groups)	444,000 Input 333,000 Output	83,300		
* 9-12 (3 groups)	570,000	125,000		
*13-16 (4 groups)	889,000 Input 666,000 Output	167,000		
** 1-4 (2 groups)			167,000	40,500
** 5-8 (4 groups)			380,000	81,000

NOTE: The specified transfer rates are maximum with ideal conditions for the specified number of active input channels or output channels. With both input channels and output channels active, the ideal maximum would be 1.75 x transfer rate per number of groups active for the specified number of channels up to a maximum limit of 1,000,000 16-bit words/second transferred. The 32-bit rates are slower due to an additional 750 ns needed for each word to transfer the additional 16-bits to and from memory.

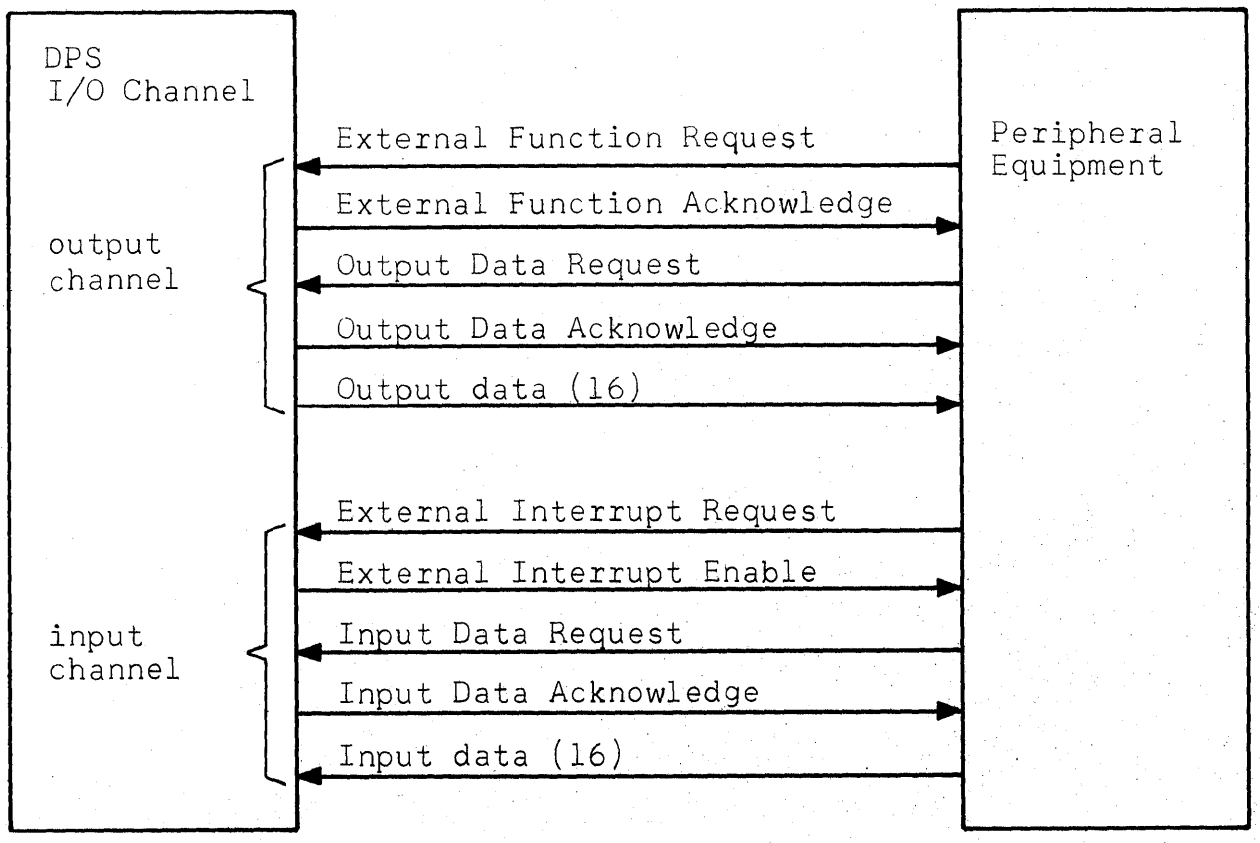


Figure 14. Parallel I/O Channel Interface for Normal Mode

3.4.5.3.1 Dual channel operation. - The DPS shall perform 32-bit transfers using two I/O channels simultaneously (n, n+4) when the following conditions are satisfied:

a. A jumper plug, wired in accordance with table 10-VIII, Appendix 10, shall be required to be plugged into both the input and output connectors on channel n+4.

b. The dual channel mode selector card shall be required to be wired to enable the dual channel mode.

c. The two channels which shall operate as a dual channel shall be corresponding channels in consecutive 4-channel groups.* Channel n shall be required to be in channel group 0 or 2. The daul channel interface shall be the channel n jack and shall include all 32 data lines and all control signals, channel n+4 control shall be disabled. Both channel groups shall be required to have the same interface voltage. For each input data, output data or external function transfer, data transfers at the dual channels shall use 2 adjacent memory locations as follows: channel n shall use address Y+1 and channel n+4 shall use address Y, where Y is the address as specified in the BCW. Y shall be required to be an even number and shall contain the 16 most significant bits of the double length word; Y+1 shall contain the 16 least significant bits.

d. The transfer mode, TM, designator in the BCW shall specify 32-bit transfers. Channels to have dual channel operation shall be required to be specified on the procurement document (see 6.1).

* Group 0 contains channels 0, 1, 2, 3
Group 1 contains channels 4, 5, 6, 7
Group 2 contains channels 8, 9, 10, 11
Group 3 contains channels 12, 13, 14, 15

3.4.5.3.1.1 ESA mode operation. - Dual channels shall be capable of operating in the ESA mode to send or receive data on a word-by-word basis. The peripheral device shall specify an address for each output requested and input word sent.

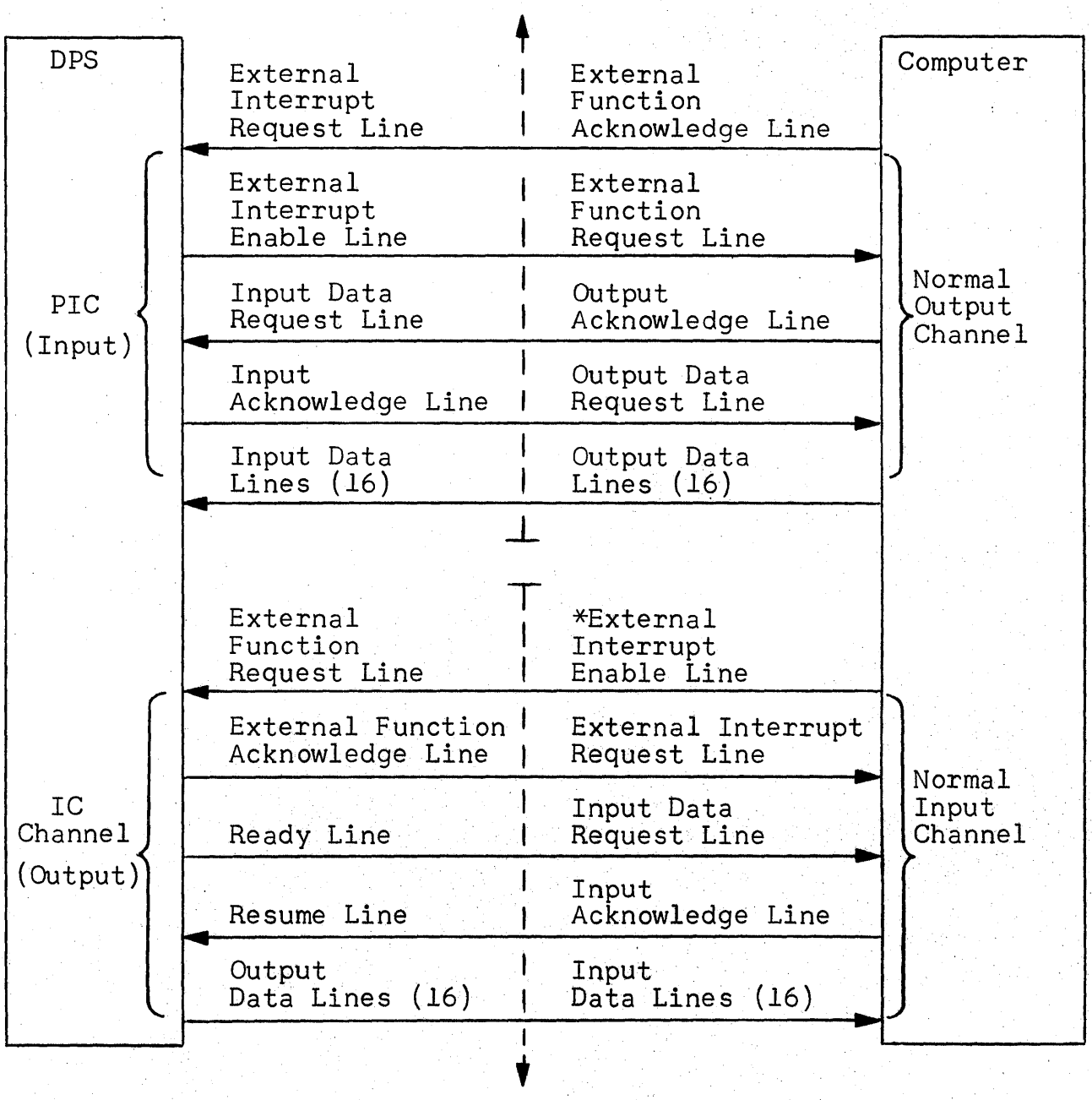
For output, the peripheral device shall place the address on the lower 16 bits of the input channel and signal via the Output Data Request line of the output channel. The DPS shall place the 32 bits of data from the requested memory address on the output channel and send an Output Acknowledge.

For input, the peripheral device shall place the data information on the high-order 16 data lines and the address on the low-order 16 data lines and signal via the Input Data Request line, respectively, of the input channel. The DPS shall store the 32 data lines (data and address) at the requested memory address and send an Input Acknowledge. The transfers shall be deactivated by the DPS executing a 70 RR instruction with an m designator of 0 to 10 by normal buffer termination. Channels to have ESA operation shall be required to have the mode selector card wired to select the ESA mode. ESA channels shall be required to be specified in the procurement document (see 6.1).

For External Interrupt or External Function transfers, normal dual channel operation shall occur.

3.4.5.3.2 Intercomputer mode. - Parallel channels shall be capable of being modified to operate in the intercomputer mode as specified in Specification DS 4772 and figure 15. Unless otherwise specified in the procurement document (see 6.1), the DPS I/O channels shall be provided with the normal buffer mode of operation.

3.4.5.3.3 Peripheral input channel. - The peripheral input channel (PIC) option shall enable a DPS parallel input channel to accept data from a normal computer output channel. The PIC shall accept data in two modes; input data mode, and external interrupt mode. In the input data mode, the PIC shall accept output data from an output device that meets the requirements of a normal computer output channel as specified in Specification DS 4772. The DPS shall have four PIC's maximum. The PIC shall be the lowest number channel of a four channel group (channel 0, 4, 8 or 12). The PIC option and channel numbers shall be required to be specified in the procurement document (see 6.1). The PIC shall interface with an output device as specified in figure 15. The PIC shall not change any other functions of the I/O channels specified herein. The PIC shall provide NTDS Slow (-15V) type interface in accordance with Specification DS 4772. The output channel corresponding to the PIC shall operate in the intercomputer or normal modes as specified by Specification DS 4772.



* Not all computers have an External Interrupt Enable Line.

Figure 15. Peripheral Input Channel Interface and IC Channel Interface

3.4.5.3.3.1 Input data mode. - When the output device has data available, the transfer of data to the PIC shall be as follows:

a. The input buffer of the PIC shall be required to be activated by the program. When the input buffer is active, the PIC shall set the Input Acknowledge line.

b. The output device shall be required to place a data word on the Input Data lines to the PIC.

c. The output device shall be required to set the Output Acknowledge line to the PIC.

d. The PIC shall detect the setting of the Input Data Request line.

e. The PIC shall accept the data word on the Input Data lines and clear the Input Acknowledge line.

f. In accordance with internal priorities, the DPS shall store the data word at the memory address as specified in the BCW and, if the buffer is still active, set the Input Acknowledge line.

The PIC and the output device shall repeat steps a through f until the transfer of the data words specified by the BCW is complete.

3.4.5.3.3.2 External Interrupt Mode. - When the output device has an external interrupt word available, the transfer of the external interrupt word to the PIC shall be as follows:

a. When the DPS is ready to accept an external interrupt word, the PIC shall set the External Interrupt Enable line as required by the program.

b. The output device shall be required to place an external interrupt word on the Input Data lines to the PIC.

c. The output device shall be required to set the External Interrupt Request line to the PIC by setting the External Function Acknowledge line.

d. The PIC shall detect the setting of the External Interrupt Request.

e. The PIC shall accept the external interrupt word on the Input Data lines, and clear the External Interrupt Enable line.

f. In accordance with internal priorities, the DPS shall store the external interrupt word at the assigned external interrupt storage address as specified herein. Additional external interrupts shall be accepted on the PIC by repeating steps a through f.

3.4.5.4 Serial I/O channels. - The DPS shall be capable of operating with four types of serial I/O channels which are required to be selected in the procurement document (see 6.1).

a. MIL-STD-188
b. RS-232
c. NTDS (32 bit)
d. VACALES

The serial channels shall be under control of words stored in control memory as specified herein.

3.4.5.4.1 MIL-STD-188 I/O. - The MIL-STD-188 serial data channels shall be a pluggable option, in groups of two full duplex I/O channels. Either synchronous or asynchronous serial channels shall be allowed. The channel interface shall be in accordance with MIL-STD-188, with the physical and electrical interface as specified in Specification SB-10631. All interface lines of one channel shall be in one interface connector. Each channel shall be capable of internal loopback testing under program control.

3.4.5.4.1.1 Synchronous modulation rates. - The MIL-STD-188 synchronous channels shall have the capabilities of modulation rates up to 9600 bits/sec.

3.4.5.4.1.2 Asynchronous modulation rates. - The MIL-STD-188 asynchronous channels shall have capabilities of 6 possible modulation rates (75, 150, 300,600,1200, and 2400 bits/sec), only four of which shall be available per 2-channel group and shall be required to be specified in the procurement document (see 6.1). Any one of these four rates shall then be selectable under program control for each channel.

3.4.5.4.1.3 Synchronous character size. - The MIL-STD-188 synchronous channels shall have 5, 6, 7, or 8 level characters selectable under program control. The parity bit shall be program selectable, and if used, shall be included in the character interval.

3.4.5.4.1.4 Asynchronous character size. - The MIL-STD-188 asynchronous channels shall have 5, 6, 7, or 8 level characters selectable under program control. The parity bit shall be program selectable, and if used, shall be included in the character interval. Each character shall have added to its interval a start bit, and under program control, one or two stop bits.

3.4.5.4.1.5 MIL-STD-188 interrupt. - The MIL-STD-188 serial interrupt word shall be as specified in figure 16. Bits 8, 9, and 10 shall be the only bits interpreted. Any of the events specified shall set the Class III external interrupt and the DPS shall store the interrupt word at the assigned address.

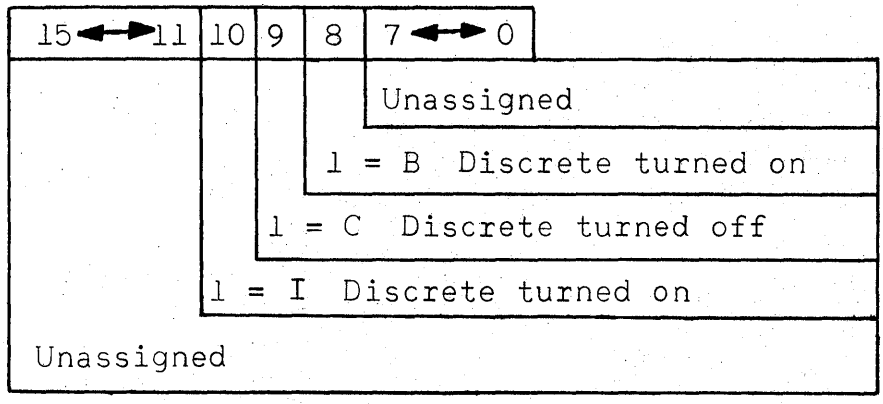


Figure 16. MIL-STD-188 Interrupt Word Format

3.4.5.4.2 RS-232 serial I/O. - The RS-232 serial data channels shall be a pluggable option, in groups of two full duplex I/O channels. Either synchronous or asynchronous serial channels shall be allowed. The channel interface shall be in accordance with EIA RS-232 with the physical and electrical interface as specified in Specification SB-10631. All interface lines of one channel shall be in one interface connector. Each channel shall be capable of internal loopback testing under program control.

3.4.5.4.2.1 Synchronous modulation rates. - The synchronous channels shall have the capabilities of modulation rates up to 9600 bits/sec.

3.4.5.4.2.2 Asynchronous modulation rates. - The RS-232 asynchronous channels shall have capabilities of 6 possible modulation rates (75, 150, 300, 600, 1200, and 2400 bits/sec), only four of which shall be available per 2-channel group and shall be required to be specified in the procurement document (see 6.1). Any one of these four rates shall then be selectable under program control for each channel.

3.4.5.4.2.3 Synchronous character size. - The RS-232 synchronous channels shall have 5, 6, 7, or 8 level characters selectable under program control. The parity bit shall be program selectable, and if used, shall be included in the character interval.

3.4.5.4.2.4 Asynchronous character size. - The RS-232 asynchronous channels shall have 5, 6, 7, or 8 level characters selectable under program control. The parity bit shall be program selectable, and if used, shall be included in the character interval. Each character shall have added to its interval a start bit, and under program control, one or two stop bits.

3.4.5.4.2.5 RS 232 interrupt. - The RS 232 serial interrupt word shall be as specified in figure **17.** Bits 8 and 9 shall be the only bits interpreted. Any of the events specified shall set the Class III external interrupt and the computer shall store the interrupt word at the assigned address.

RS-232 serial interface

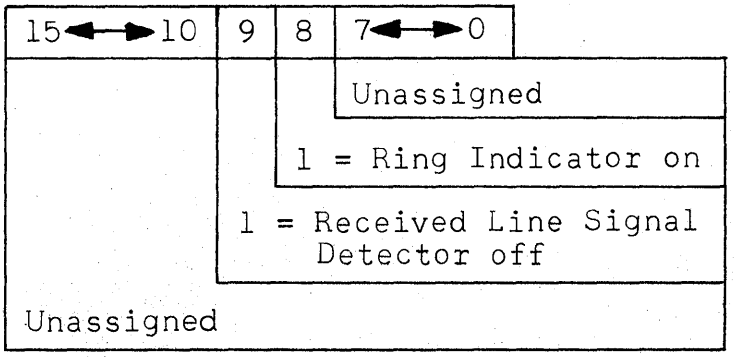


Figure **17.** RS 232 Interrupt Word Format

3.4.5.4.3 NTDS serial I/O - The NTDS serial data channels shall be a pluggable option providing full duplex I/O transfers. Each group shall consist of an output channel and an input channel (see figure **18**). The output channel shall use one cable connected at the even-numbered output channel location. The input channel shall use one cable connected at the even numbered input channel location. The odd-numbered I/O channel connector locations shall not be used. Except that the connectors shall be as specified in Appendix 10, the channel shall be in accordance with Specification SB-12407. Both the output channel and the input channel shall transfer 32 bits of data along with sync bit, identifier bit, and control frames. Each 32 bit word shall contain 32, 16, or 8 bits of significant data, as specified by the TM field of the BCW.

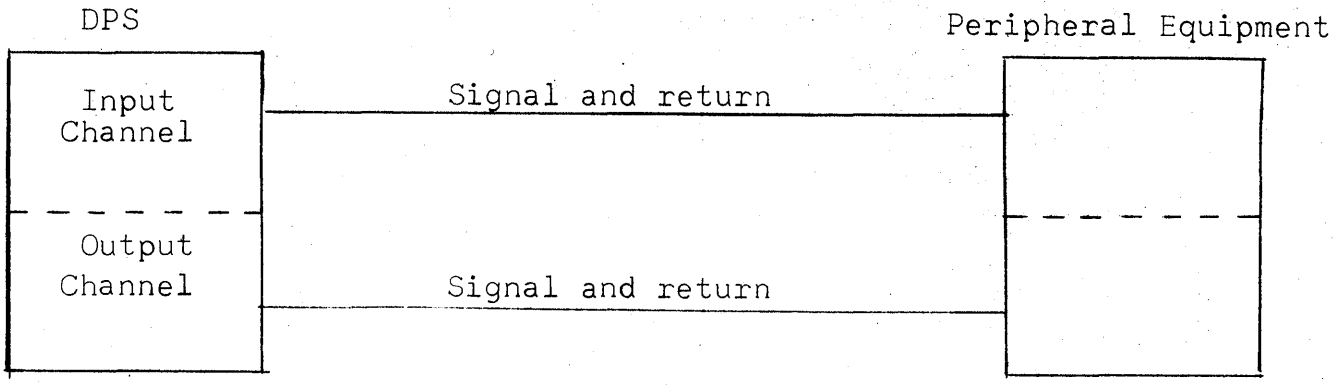


Figure 18. NTDS Serial I/O Channels

3.4.5.4.3.1 NTDS serial output transfers. - The DPS shall continually interrogate the peripheral equipment on each NTDS serial output channel by sending output enable control frames (OECF) with output data enable and external function enable bits set to the peripheral equipment. The peripheral equipment shall respond with an output request control frame (ORCF). If the DPS has an active output buffer corresponding to its OECF, the DPS in accordance with internal priorities, shall send a 32-bit output or external function word to the peripheral equipment. If the DPS did not have an active output buffer corresponding to its OECF, another OECF from the DPS and another ORCF from the peripheral equipment shall occur. All output control frames and output data shall be transferred on the output channel cable. Each 32-bit output word transferred shall be accompanied by an OECF and an ORCF.

3.4.5.4.3.2 NTDS serial input transfers. - The peripheral equipment shall be required to interrogate the DPS on each NTDS serial input channel by sending input request control frames (IRCF) with input data request and external interrupt request bits set to the DPS. The DPS shall respond with an input enable control frame (IECF). If the DPS is ready to receive a 32-bit input or interrupt word, the peripheral equipment shall send the word if ready. If the peripheral equipment did not have data corresponding with the IECF from the DPS, another IRCF from the peripheral equipment and another IECF from the DPS shall occur. All input control frames and input data shall be transferred on the input channel cable. Each 32-bit input word transferred shall be accompanied by an IRCF and an IECF.

3.4.5.4.3.3 NTDS serial I/O timing. - All NTDS serial I/O timing shall be in accordance with Specification SB-12407. In the case of output, if ideal data transfers occur, the DPS shall transfer each 32-bit word, along with the required OECF, every 8 usec nominal for an effective data transfer rate of 125,000 words/second (32-bit words).

3.4.5.4.4 VACALES serial I/O. - The VACALES serial data channels shall be a pluggable option in groups of four full duplex I/O channels. Except as specified herein, the channel interface shall meet the physical, functional, and electrical requirements specified for a MIL-STD-188 synchronous interface as specified in Specification SB-10631.

3.4.5.4.4.1 Modulation rates. - The VACALES channels shall have the capabilities of modulation rates up to 10,000 bits/second.

3.4.5.4.4.2 Character size. - The VACALES channels shall have character lengths of from one bit per character through 16 bits per character selectable under program control. The parity bit shall be program selectable, and if used, shall be included in the character size.

3.4.5.4.4.3 Interrupt word. - The VACALES channels interrupt word shall be as specified in figure 19. Bits 8 through 12 shall be the only bits interpreted. Any of the events specified shall set the Class III interrupt and the DPS shall store the interrupt word at the assigned address.

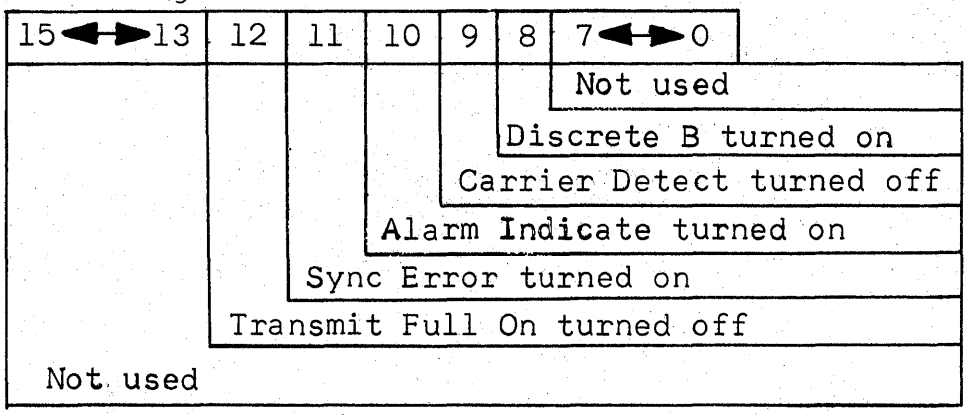


Figure 19. VACALES Interrupt Word Format

3.4.5.5 I/O operating program control. - I/O operations shall be established using the I/O instructions of the DPS instruction repertoire. The DPS shall execute the I/O instructions using the operations specified in the following subparagraphs.

3.4.5.5.1 Command operations. - The DPS shall be capable of executing I/O instructions designated command as follows:

a. The DPS shall execute a 35 RR instruction as part of the normal program sequence.

b. The action of the 35 RR instruction shall be to direct the DPS to execute a command I/O instruction stored at memory addresses 000140 and 000141.

c. The DPS shall clear bits 14 and 15 of the contents of relative address 000140 after executing the contents of that address before resuming the normal program sequence.

3.4.5.5.2 Chaining operations. - The DPS shall enable chaining for an input or output channel when the DPS executes an Initiate Chain (71RK) command instruction. Chaining means that the chain address pointer, within I/O control memory, shall address a program which shall control the operations of the channel. The DPS shall be capable of executing I/O instructions designated chaining as follows:

a. The DPS shall obtain the instruction from the memory location specified by the contents of the chain address pointer CM location and shall increment by 1 (single length instruction) or 2 (double length instruction) the chain address pointer for each chaining instruction executed.

b. The DPS shall execute the chaining instruction. A chaining instruction shall perform operations only for the channel corresponding to the chain address pointer referencing the instruction.

c. The DPS shall repeat (a) and (b) for each chaining instruction until chaining terminates. Chaining shall terminate under program control (instructions 73 RR a=0, 70 RR a=0 or 10).

3.4.5.5.3 Channel priority. - The DPS shall do either a CP operation or an IOC related operation at any one time (CP and IOC share the memory interface and the microprogrammed controller). CP related operations shall be performed on a lower priority than IOC related operations. I/O channel related operations shall be performed according to channel priority first and function priority second. Channel priority shall be in order from channel 15 highest to channel 0 lowest. Channel function priority for parallel channels and NTDS serial channels shall be highest to lowest as follows:

a. External interrupt input

b. Output buffer (OD or EF)

c. Input buffer (ID or EF)

Channel function priority for RS-232 channels and MIL-STD-188 channels shall be highest to lowest as follows:

a. Input data

b. Output data

c. External interrupt data

3.4.6 Addressable registers. - The registers as specified in the following subparagraphs shall be addressable for the specified operations.

3.4.6.1 General registers. - The DPS shall have one general register set with an option (see 6.1) for a second set. Each general register set shall consist of sixteen 16-bit registers designated R_0 through R_{17} (octal). Each R register shall be capable of being used as an accumulator for arithmetic, shift and logical functions, an index register for address and operand modification, and as temporary storage. The R registers shall be addressable by designators of the instruction words. Each instruction that requires a reference to an R register shall define the use of the register for that instruction. Only one set of R registers shall be active at any one time as selected by bit 14 of Status Register 1.

3.4.6.2 Program address register. - The DPS shall have a program address register designated P. The contents of P shall specify the address of the next instruction. The DPS shall increment by one the contents of P for each single length instruction and shall increment by two the contents of P for each double length instruction. Instructions which cause program jumps shall enter P with the address of the instruction to which program control is transferred. When the operations specified by the current instruction are completed, the contents of P shall then be used to obtain the next instruction.

3.4.6.3 Page registers. - The DPS shall have 64 page registers containing the base address for memory locations. All memory references shall be relative to the fixed base contained in a page register. The page registers shall be capable of modification under program control.

3.4.6.4 Breakpoint register. - The DPS shall have a breakpoint register, which shall contain an address which shall be used as a relative address. If the breakpoint read stop on the maintenance panel is enabled, the DPS shall stop after reading data from the address specified by the breakpoint register. If the breakpoint write stop on the maintenance panel is enabled, the DPS shall stop after writing data into the address specified by the breakpoint register. The DPS shall stop after completing the instruction being executed when breakpoint occurred. Loading the breakpoint register shall be under manual control through the maintenance panel interface. Except that breakpoint shall sense only the even-addressed memory reference for these two-word operations: double-length operands, or ESA data; the breakpoint register shall sense all memory references including input data, output data, and indirect addresses.

3.4.6.5 Status registers. - The DPS shall have status registers designated Status Register 1 and Status Register 2. The DPS shall use Status Register 1 as specified in figure 20 and Status Register 2 as specified in figure 21.

3.4.6.6 Real time clock (RTC) register. - The DPS shall have a 32-bit RTC register. The contents of the RTC register shall increment at the frequency of the clock source. The DPS shall generate the RTC Overflow interrupt when the lower 16 bits of the RTC register change from all ones to all zeros. Loading the RTC register and enabling and disabling the incrementing sequence and enabling and disabling the RTC interrupt shall be under program control using instruction 03RR. When enabled, the RTC shall increment only when the DPS is executing instructions in the run mode.

3.4.6.7 Monitor Clock register. - The DPS shall have a 16-bit Monitor Clock register. When enabled, the contents of the Monitor Clock register shall decrement at the frequency of the clock source. The DPS shall generate the Monitor Clock interrupt and disable the Monitor Clock when the contents of the Monitor Clock equal zero. Loading the Monitor Clock register and enabling and disabling the decrementing sequence shall be under program control using instruction 03RR. When enabled, the Monitor Clock register shall decrement only when the DPS is executing instructions in the run mode.

3.4.7 Clock source. - The DPS shall have an internal clock source and shall accept signals from an external clock source to provide increment timing for the RTC register and decrement timing for the Monitor Clock register. Whether the internal or the external source shall furnish the timing shall be under control of a maintenance panel switch. A RTC disable switch shall be provided on the maintenance panel, which shall override the RTC programmed enable.

3.4.7.1 Internal clock source. - The DPS shall have an internal clock source with a frequency of 1000 Hz with an accuracy of not less than $\pm$ 1 count per 10 seconds.

3.4.7.2 External clock source. - The DPS shall accept signals from an external clock source with a frequency of from 0 to 50 kHz. The clock receiver shall have the characteristics specified in Specification DS 4772 for -3V interface receivers.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

0 = DMA locked out

1 = DMA enabled

Interrupt lockout designators:

bit 1 = Class III
bit 2 = Class II
bit 3 = Class I

A bit cleared shall mean the interrupts in that class are locked out.

Not used

0 = Round the floating point results.

1 = Provide floating point residue.

0 = Set floating point interrupt on overflow or underflow.

1 = Do not interrupt on floating point overflow or underflow.

Condition code designator; the condition of these bits shall specify the result of arithmetic and compare instructions as specified in table IV and as specified in Appendix 40.

Overflow designator; this bit shall set when an arithmetic or shift operation produces a result which requires more bits than provided in the register and as specified in Appendix 40.

Carry designator; this designator shall set when an arithmetic operation generates a carry out of the most significant bit in the register and as specified in Appendix 40.

NDRO Mode: The condition of this bit shall select access to NDRO memory.

0 = Select NDRO memory

1 = Select main memory

Not used

0 = General Register set 0 (standard set)

1 = General Register set 1 (optional set)

Not used

Figure 20. Status Register # 1

15 14	13 12	11 10	9 8	7 6 5 4 3 2 1 0
				Illegal IOC Instruction Interrupt* or Memory Resume Interrupt**
			Indirect control bits for R_m = 10 00 Normal addressing 01 Normal addressing 10 Indirect addressing 11 Indirect addressing with indexing	
		Indirect control bits for R_m = 12 (interpret same as R_m = 10)		
	Indirect control bits for R_m = 14 (interpret same as R_m = 10)			
Indirect control bits for R_m = 16 (interpret same as R_m = 10)				

*Illegal IOC Instruction Interrupt

7 6 5 4	3	2	1 0
Command instruction			
0 0 0 0	0	0	0 1
Chain instruction			
			1 0
			0 = input chain 1 = output chain
		0	
Channel number (octal)			

**Memory Resume Interrupt

7	6 5 4	3 2 1 0
		0 0 1 0
	0 0 0	Module 1
	0 0 1	Module 2
	0 1 0	Module 3
	0 1 1	Module 4
	1 0 0	Module 5
	1 0 1	Module 6
	1 1 0	Module 7
	1 1 1	Module 8
1		

8K Modules of Memory

Figure 21. Status Register #2

3.4.8 Interrupts. - The DPS shall have an interrupt structure which shall permit the suspension of the normal program sequence to perform special functions related to the type of interrupt present. Interrupt processing shall be dependent on a priority network. Interrupts shall be assigned priority in order of importance. The action of an interrupt shall be to store pertinent data to allow resuming normal program operation from point of interruption and to transfer control to an instruction located at an address corresponding to the interrupt.

The interrupt priority and the events causing interrupts shall be as specified in table VIII. The interrupts shall be grouped in three classes with decreasing priority as follows: Class I, Class II, Class III.

If two or more interrupts occur simultaneously, the DPS shall process the interrupt with higher priority.

3.4.8.1 Interrupt processing. - When an interrupt has priority, processing of that interrupt shall be as follows:

a. Terminate the current program sequence. Lockout all interrupts during the following operations: b, c, and d.

b. Store the contents of P, STATUS registers #1 and #2, and the RTC register at assigned memory addresses.

c. Load STATUS registers #1 and #2 with the contents of the assigned memory addresses.

d. Load the P register with the interrupt entrance address.

e. Enable honoring interrupts not locked out.

f. Execute the instruction at the interrupt entrance address.

Table VIII. Interrupt Priority

Class	Priority Within Class	Interrupt	Interrupt Code (binary)
Class I	1	Power Fault. No lockout.	000
	2	Memory resume; generated when the memory fails to acknowledge a request within 12 usec. The interrupt is lost if Class I lockout is set.	001
	3	Unassigned	010
	4	Unassigned	011
	5	Unassigned	100
	6	Unassigned	101
	7	Unassigned	110
	8	Unassigned	111
Class II	* 1	CP Instruction Fault: generated when the CP attempts to execute an instruction with an unused operation code or 7X instruction of any kind. No lockout.	000
	* 2	IOC Instruction Fault: generated when the IOC attempts to execute an instruction with an unassigned operation code or any instruction other than 7X. If the instruction is an I/O chain instruction, terminate the chain. No lockout.	001
	3	Floating Point (optional): generated contingent upon bit 7 of Status Register 1, whenever a floating point overflow or underflow occurs. No operation if Class II lockout is set.	010
	4	Executive Return: generated when the DPS executes the 03RR m=0 instruction. No operation if Class II lockout is set.	011

*Sets Program Fault indicators on maintenance and control panels.

Table VIII. Interrupt Priority (Continued)

Class	Priority Within Class	Interrupt	Interrupt Code (binary)
Class II	5	RTC Overflow: generated when the contents of bits 0 through 15 of the RTC register increments from all ones to all zeros. One level of queuing.	100
	6	Monitor Clock: generated when the contents of the Monitor Clock register equals zero. One level of queuing.	101
Class III	*** 1	Intercomputer time out: generated when the sending DPS fails to receive a resume from a receiving computer within the allotted time after sending an output or external function word.	11
	, * 2	External Device Interrupt: generated when the DPS has stored an External Interrupt word.	00
	*** 3	Output Chain: I/O Monitor Interrupt: generated when the DPS executes the 73 RR a = 1 instruction from an output chain.	10
	*** 4	Input Chain: I/O Monitor Interrupt: generated when the DPS executes the 73 RR a = 1 instruction from an input chain.	01

**If RS-232, MIL-STD-188, or VACALES Serial interface, this shall indicate a discrete interrupt.

*** The IC timeout interrupt has one level of queuing per channel group. All other Class III Interrupts have one level of queuing per channel.

3.4.8.2 Power fault operation. - Except that the power fault interrupt shall be locked out upon power up, after bootstrap load, or after a previous power fault interrupt until load PSW is executed; the power fault interrupt shall occur when a voltage out of tolerance condition occurs. When a voltage out of tolerance condition occurs, the DPS shall set the power fault interrupt. At this time there shall be a minimum of 250 us to execute a power fault interrupt evaluation program before a master clear occurs. The power fault interrupt evaluation program should be a routine that stores in memory, volatile data from registers. The last instruction of the routine should be the Jump on Voltage out of Tolerance instruction with a jump address back to itself. If the voltage out of tolerance is only a low voltage marginal condition and a master clear does not occur, the DPS shall continue executing this instruction until the voltage out of tolerance condition is corrected.

3.4.8.3 Intercomputer (IC) time-out interrupt. - When transferring either external function words or output data words in the IC mode, the transmitting DPS shall hold the word in its output register until the receiving DPS has acknowledged receipt of the word. If the transmitting DPS does not receive this acknowledge within a specified time after the word is placed in its output register, an IC time out interrupt shall be generated. The IC timeout interrupt shall be 256 msec to 512 msec when the RTC is controlled by internal timing. If the RTC is controlled by external timing, the IC timeout interrupt shall be equal to the time necessary for bit 8 of the RTC to change state twice.

3.4.8.4 Interrupt entrance address generation. - The CP shall generate the interrupt entrance address dependent on the interrupt class as follows:

a. Class I and II: The interrupt entrance address shall be the sum of the contents of the Load P address (from assigned memory) and the index as specified in figure 22.

b. Class III: The interrupt entrance address shall be the sum of the contents of the Load P address (from assigned memory) and the index as specified in figure 23.

3.4.9 Automatic start. - The DPS shall automatically start at bootstrap address 00000 when DPS power is applied and the AUTO START/START switch on the control panel is in the AUTO START position or after power is restored following a Power Fault Interrupt and master clear when the AUTO START/START switch is in the AUTO START position.

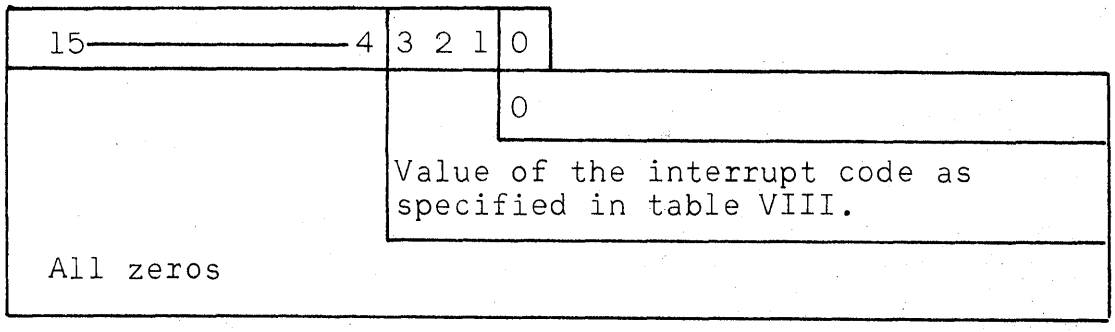


Figure 22. Class I and II Interrupt Entrance Address Index

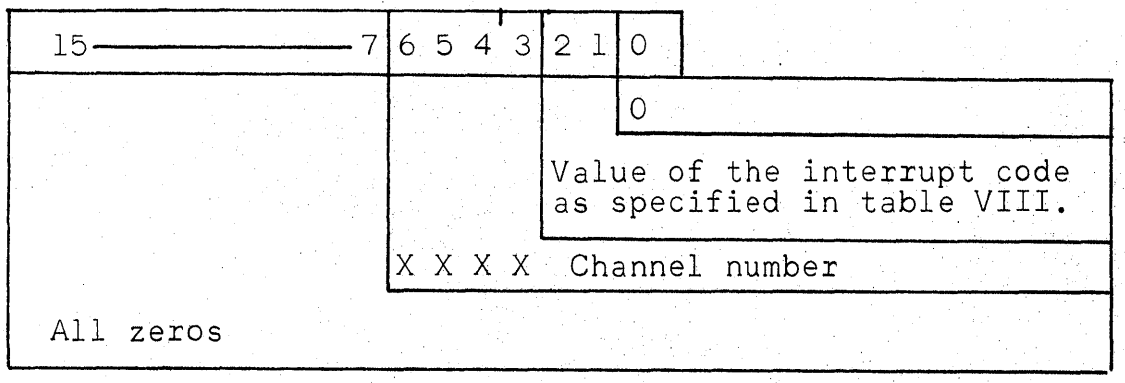


Figure 23. Class III Interrupt Entrance Address Index

3.4.10 Control and maintenance functions. - A control panel and a maintenance panel shall be provided for manual operation and maintenance functions. The maintenance panel shall be mounted on the inside of the DPS access door and the control panel on the outside of the DPS. The controls and indicators for the control panel and maintenance panel shall be as specified herein.

3.4.10.1 Control panel controls and indicators. - the DPS control panel shall provide the switches and controls specified in table IX.

3.4.10.2 Maintenance panel controls and indicators. - The DPS maintenance panel shall provide the switches and controls specified in table X.

3.4.10.3 Display select codes. - The display select codes shall be as specified in table XI. In addition, to select a page register, an output channel, or contents of an I/O control memory address for display; perform the following steps:

a. Select NORMAL DSPL.

b. Select INST REG.

c. For a page register: enter the address into bits 5-0 of REGISTER/DATA
For output data: Enter the channel number into A.
For an I/O CM address: Enter the channel number into A and the word number into M.

d. Set the DISPLAY NUMBER to the corresponding code (page, output data, or control memory).

3.4.11 Master clear. - The DPS shall have a master clear capability. This capability shall:

a. Clear the P register.

b. Clear Status Register 1.

c. Clear Status Register 2.

d. Disable incrementing of the real time clock and decrementing of the monitor clock registers.

e. Set bits 0 through 5 of all page registers equal to each respective page register address.

f. Clear channels as specified for the 70 RR instruction with an m-designator = 0.

g. Clear PROGRAM AND POWER FAULT indicators on Control and Maintenance Panels.

h. Set outgoing control lines 5 and 6 for MIL-STD-188 serial channels. Clear all other discrete lines for MIL-STD-188 and VACALES mode serial channels.

i. Clear all discrete lines for RS-232 serial channels.

j. Select NORMAL DSPL indicator-switch on maintenance panel.

k. Set the MIL-STD-188 and RS-232 transmitters to the idle state (logic 1).

Table IX. Control Panel Controls and Indicators

Identification	Type	Function
CIRCUIT BREAKER ON/OFF	Two-position toggle switch	ON position enables primary power to the DPS. OFF position disables primary power to the DPS.
BLOWER POWER ON/OFF	Two-position toggle switch	ON position enables power to the DPS cooling fans and enables LOGIC POWER ON/OFF switch function. OFF position disables power to the DPS cooling fans and disables LOGIC POWER ON/OFF switch function.
BLOWER POWER	Indicator	When lit, indicates blower power is applied and enabled.
LOGIC POWER ON/OFF	Two-position toggle switch	ON position enables power to the DPS logic. OFF position disables power to the DPS logic.
LOGIC POWER	Indicator	When lit, indicates logic power is applied and enabled.
POWER FAULT	Indicator	When lit, indicates a Power Fault Interrupt has occurred.
POWER FAULT CLR	Two-position return-to-neutral toggle switch	When momentarily operated to the CLR position, clears the POWER FAULT indicators on both the control panel and the maintenance panel.
PROGRAM FAULT	Indicator	When lit, indicates the DPS has attempted to execute an illegal CP or IOC instruction.

Table IX. Control Panel Controls and Indicators (cont.)

Identification	Type	Function
PROGRAM FAULT CLR	Two-position return-to-neutral toggle switch	When momentarily operated to the CLR position, clears the PROGRAM FAULT indicator on the control panel and the PROG FAULT indicator on the maintenance panel.
PROG RUN	Indicator	When lit, indicates DPS is executing instructions in Run Mode.
OVER TEMP	Indicator	When lit, indicates DPS internal cabinet air temperature is within 25° F. of the maximum temperature at which the DPS can operate without component damage.
ALARM	Audible	With the ALARM ENABLE/DISABLE/TEST switch in the ENABLE position, when sounding, indicates DPS internal cabinet air temperature is within 25° F. of the maximum temperature at which the DPS can operate without component damage.
ALARM ENABLE/DISABLE/TEST	Three-position toggle switch	ENABLE position enables the audible alarm function. DISABLE position disables the audible alarm function. TEST position causes the audible alarm to sound and the OVER TEMP indicator to light.

Table IX. Control Panel Controls and Indicators (cont.)

Identification	Type	Function
BATTLE SHORT ON/OFF	Two-position toggle switch.	ON position disables DPS over-temperature shutdown function. OFF position enables DPS over-temperature shutdown function.
BATTLE SHORT	Indicator	When lit, indicates BATTLE SHORT switch is in the ON position.
BOOTSTRAP 1-2	Two-position toggle switch.	Position 1 enables the execution of the first of two possible bootstrap programs in NDRO memory. Position 2 enables the execution of the second of two possible bootstrap programs in NDRO memory.
LOAD/STOP	Three-position return-to-neutral toggle switch.	When momentarily operated to the LOAD position, causes the DPS to execute a master clear, then begin executing instructions from NDRO address 0002. When momentarily operated to the STOP position, causes the DPS to stop executing instructions if the DPS was executing instructions.

Table X. Maintenance Panel Indicators and Switches

Identification	Type	Function
PROG RUN	Indicator-switch	Indicator function - When lit, indicates the DPS is executing instructions in the Run mode. Switch function - Selects run condition in microstep mode.
POWER FAULT	Indicator-switch	Indicator function - When lit, indicates a Power Fault Interrupt has occurred. Switch function - When operated, clears the POWER FAULT indicators on both the control and maintenance panels.
PROG FAULT	Indicator-switch	Indicator function - When lit, indicates the DPS has attempted to execute an illegal CP or IOC instruction. Switch function - When operated, clears the PROGRAM FAULT indicator on the control panel and the PROG FAULT indicator on the maintenance panel.
PROGRAM STOP	Indicator	When lit, indicates a program stop condition has been satisfied.
PROGRAM STOP 1/OFF	Two-position toggle switch	In 1 position, causes a program stop when the DPS executes a jump instruction with an a value = 12.
PROG STOP 2/OFF	Two-position toggle switch	In 2 position, causes a program stop when the DPS executes a jump instruction with an a value = 13.

Table X. Maintenance Panel Indicators and Switches (cont.)

Identification	Type	Function
Time meter	4 digit, 0000 to 9999	Indicates time in hours that DPS logic power has been applied and enabled.
DIAGNOSTIC JUMP	Two-position toggle switch	In the JUMP position, causes the microprogram to jump from the execution of the microjump instruction.
DIAGNOSTIC DISPLAY	Two-position toggle switch	In the DISPLAY position while the DPS is in the Microstep mode. a. With MICRO ADRS set, REGISTER/DATA displays the address of the next microinstruction to be executed. b. With MICRO INSTR set, REGISTER/DATA displays the microinstruction currently being executed. c. With NORMAL DSPL set, REGISTER/DATA displays the data on the source bus associated with the microinstruction currently being executed.
PROCESSOR DISABLES RT CLK DISABLE/INT/EXT	Three-position toggle switch	DISABLE position inhibits incrementing of the Realtime Clock Register and decrementing of the Monitor Clock register. INT position causes the Real-Time-Clock Register and the Monitor Clock register to use the internal clock source for timing. EXT position causes the Real-time Clock Register and the Monitor Clock register to use the external clock source for timing.

Table X. Maintenance Panel Indicators and Switches (cont.)

Identification	Type	Function
PROCESSOR DISABLES ADV P	Two-position toggle switch	Up position inhibits incrementing of the P-Register
PROCESSOR DISABLES INTERCMPTR TIME OUT	Two-position toggle switch	Up position inhibits the occurrence of a Class III Intercomputer Timeout Interrupt.
MODE MICRO STEP	Indicator-switch	Indicator function - When lit, indicates DPS is in Microstep mode. Switch function - When operated, places DPS in Microstep mode.
MODE OP STEP	Indicator-switch	Indicator function - When lit, indicates DPS is in Op Step Mode or in Microstep Mode to execute a single instruction. Switch function - When operated, clears Run mode and enables the Op Step mode.
MODE RUN	Indicator-switch	Indicator function - When lit, indicates DPS is in Run mode or in Microstep Mode to execute successive instructions. Switch function - When operated, clears Op Step mode and enable the Run mode.
DISPLAY SELECT CLR	Pushbutton switch	When operated clears DISPLAY NUMBER 0 through 3 and clears the Microstep mode.

Table X. Maintenance Panel Indicators and Switches (cont.)

Identification	Type	Function
ALTER MODE SET/CLEAR	Two-position toggle switch	In the SET position, a. causes each REGISTER/DATA indicator-switch to set when operated. b. causes all REGISTER/DATA indicator-switches to clear when the REGISTER/DATA SET/CLR switch is operated. In the CLEAR position, a. causes each REGISTER/DATA indicator-switch to clear when operated. b. causes all REGISTER/DATA indicator-switches to set when the REGISTER/DATA SET/CLR switch is operated.
REGISTER/DATA SET/CLR	Pusbutton switch	When operated, sets or clears (dependent on ALTER MODE SET/CLR position) REGISTER/DATA indicator-switches 0 through 15
REGISTER/DATA 0 through 15	Indicator-switches	Indicator function - Display contents of selected register Switch function - Modify contents of selected register.

Table X. Maintenance Panel Indicators and Switches (cont.)

Identification	Type	Function
DISPLAY SELECT MICRO ADRS MICRO INSTR NORMAL DSPL	Indicator-switches	Indicator function - When lit as specified in table XI indicate REGISTER/DATA is displaying the corresponding register.
INSTR REG GENL DSPL GENL REG DISPLAY NUMBER 0-3		Switch function. - When operated as specified in table XI causes REGISTER DATA to display the corresponding register contents. Operation of any one of MICRO ADRS, MICRO INSTR, or NORMAL DSPL causes the other two to clear. Operation of any one of INSTR REG, GENL DSPL, or GENL REG causes the other two to clear.

Table X. Maintenance Panel Indicators and Switches (continued)

Identification	Type	Function
AUTO START/START	Three-position toggle switch	When set to the AUTO START position, causes the DPS to execute the instruction at NDRO memory address 000000 when power is applied or restored. When momentarily operated to the START position, causes the DPS to execute instructions in the selected mode.
STOP	Two-position return-to-neutral toggle switch	When operated to the STOP position while the DPS is executing instructions in the Run mode, causes the DPS to stop executing instructions.
MA CLR	Pushbutton switch	When operated while the DPS is not in Run condition, causes the DPS to execute a Master Clear function. When operated while the DPS is in the Run mode, clears the FAULT indicators on the control and maintenance panels.
BREAK PT READ/OFF	Two-position toggle	READ position causes the DPS to stop executing instructions after reading data from the memory address specified by the contents of the breakpoint register.
BREAK PT WRITE/OFF	Two-position toggle	WRITE position causes the DPS to stop executing instructions after writing data in the memory address specified by the contents of the breakpoint register.

Table XI. Display Select Codes

MICRO ADRS MICRO INSTR NORMAL DSPL	INSTR REG GENL DSPL GENL REG	DISPLAY NUMBER 3 2 1 0	Register Selected	
1 0 0	X X X	X X X X	uP Register **	
0 1 0	X X X	X X X X	uI Register **	
0 0 1	1 0 0	X X X X	Instruction Register	
0 0 1	0 1 0	0 0 0 0	P Register	
0 0 1	0 1 0	1 0 0 0	Memory Address (P)	
0 0 1	0 1 0	0 0 0 1	Status Register 1	
0 0 1	0 1 0	0 0 1 0	Status Register 2	
0 0 1	0 1 0	0 0 1 1	RTC Register - Lower 16 Bits	
0 0 1	0 1 0	0 1 0 0	RTC Register - Upper 16 Bits	
0 0 1	0 1 0	0 1 0 1	Breakpoint Register	
0 0 1	0 1 0	1 0 1 0	Monitor Clock Register	
0 0 1	0 1 0	0 1 1 0	I/O Control Memory	
0 0 1	0 1 0	0 1 1 1	Page Register	
0 0 1	0 1 0	1 0 0 1	Output Data	
0 0 1	0 1 0	1 1 1 1	Load uP	
0 0 1	0 0 1	0 0 0 0	General Register R_0	*
0 0 1	0 0 1	0 0 0 1	General Register R_1	
0 0 1	0 0 1	0 0 1 0	General Register R_2	
0 0 1	0 0 1	0 0 1 1	General Register R_3	
0 0 1	0 0 1	0 1 0 0	General Register R_4	
0 0 1	0 0 1	0 1 0 1	General Register R_5	
0 0 1	0 0 1	0 1 1 0	General Register R_6	
0 0 1	0 0 1	0 1 1 1	General Register R_7	
0 0 1	0 0 1	1 0 0 0	General Register R_{10}	
0 0 1	0 0 1	1 0 0 1	General Register R_{11}	
0 0 1	0 0 1	1 0 1 0	General Register R_{12}	
0 0 1	0 0 1	1 0 1 1	General Register R_{13}	
0 0 1	0 0 1	1 1 0 0	General Register R_{14}	
0 0 1	0 0 1	1 1 0 1	General Register R_{15}	
0 0 1	0 0 1	1 1 1 0	General Register R_{16}	
0 0 1	0 0 1	1 1 1 1	General Register R_{17}	

*Clearing Status Register 1 bit 14 shall display general register set 1. Setting Status Register 1 bit 14 shall display general register set 2.

**DPS must be in the Microstep Mode.

3.5 Environmental and service conditions. -

3.5.1 Temperature. -

3.5.1.1 Operating temperature. - The DPS shall operate as specified herein without electrical or mechanical damage in any ambient temperature between the operating limits specified in MIL-E-16400 for Class 4 equipment (0°C to +50°C).

3.5.1.2 Nonoperating temperature. - The DPS shall be capable of extended storage within the nonoperating temperature range specified in MIL-E-16400 for Class 4 equipment (-62°C to +75°C). The DPS shall not incur physical damage from extended storage within the nonoperating temperature range and shall operate as specified herein when restored to the operating temperature range.

3.5.2 External radiation. - The DPS shall withstand externally radiated, narrow band field intensities as specified in MIL-STD-461 without sustaining a fault or malfunction.

3.5.3 Electromagnetic interference. - The DPS shall meet the electromagnetic interference requirements specified by MIL-STD-461 for Class IC equipment.

3.5.4 Noise. - The DPS shall conform to the airborne and structure borne noise requirements for Grade A, Type 3 equipment as specified in MIL-STD-740.

3.5.5 Shock. - Except for the circuit breaker, the DPS shall meet the Type A requirements of MIL-S-901 for Grade A, hull mounted, Class I, lightweight equipment. The DPS shall be capable of operating as specified herein during and after the specified shock conditions, except that momentary malfunction is acceptable during operation under high impact shock, provided that no manual intervention is required for continued operation, no loss of data occurs, and no loss of I/O communication occurs.

3.5.6 Vibration. - The DPS shall be capable of operating as specified herein during and after the application of environmental vibration (Type I) in accordance with MIL-STD-167.

3.5.7 Enclosure. - The DPS shall meet the enclosure requirements of MIL-E-16400 and shall adequately protect personnel from electrical and physical injury when the equipment is being operated. The DPS degree of enclosure shall be drip proof from 0 to 45 degrees in accordance with MIL-STD-108.

3.5.8 Salt spray. - The DPS shall meet the salt spray requirements of FED-STD-151, Method 811. The test duration shall be 200 hours for the frame and enclosure and 48 hours for the parts.

3.5.9 Humidity. - The DPS shall perform as specified herein at relative humidities ranging up to 95 percent for both continuous and intermittent periods.

3.5.10 Inclination. - The DPS shall maintain operation as specified herein during and after inclination tests as specified in MIL-E-16400.

3.5.11 Magnetic characteristics. - Insofar as practicable, no material shall be employed in the DPS which has a relative magnetic permeability greater than 2.0, as specified in MIL-E-16400.

3.5.12 Warmup time. - The time required for the DPS to warm up prior to operation shall not exceed 0.5 minutes at 20°C, and 2.0 minutes at 0°C.

3.6 Reliability. - The reliability of the maximum configuration DPS shall be 2000 hours minimum mean-time-between-failures (MTBF) when calculated using the methods of MIL-HDBK-217 and Univac generic failure rates.

3.7 Maintainability. -

3.7.1 Corrective maintenance. - The DPS shall be such that ninety-five percent of all organizational corrective maintenance actions shall be performed in less than 120 minutes with the aid of a diagnostic program, [M_{max} (95 percentile) = 120 minutes] and the mean corrective maintenance M_{ct} shall be less than or equal to 15 minutes when repair is accomplished by replacement of line replaceable items (LRI) and chassis mounted components. The specified corrective maintenance times shall not apply to combat damage repair, intermittent fault location, or multiple faults. As here applied the organizational corrective maintenance action includes the following tasks as defined by MIL-HDBK-472:

a. Localization
b. Isolation
c. Disassembly
d. Interchange
e. Reassembly
f. Checkout

3.7.2 LRI - corrective maintenance. - The DPS shall be designed so that LRI's are throwaway, except memory and power supply modules which cost in excess of $500.00. All assemblies shall be capable of repair in an emergency situation with the exception of solid state micro-circuits.

3.7.3 LRI - electrical alignment if applicable. - The DPS LRI's shall be designed so that no electrical alignment of LRI's will be required at the organizational level. The mean time to align (MTTA) the LRI's (including printed wiring assemblies that require alignment at the depot after repair), shall be less than or equal to 1 hour.

3.7.4 Qualitative maintainability requirements. - The DPS shall be constructed to permit accessibility and replacement of all LRI's, power supply modules, memory modules, and operator/maintenance panel parts and components. Access panels and doors which must be opened for maintenance or inspection shall have a maximum time to open of 30 seconds.

3.8 Workmanship. - Workmanship shall be in accordance with MIL-STD-454, Requirement 9, Workmanship.

4. QUALITY ASSURANCE PROVISIONS

4.1 Responsibility for inspection. - The contractor shall be responsible for the performance of all inspection requirements as specified herein. The contractor shall use any facilities suitable for the performance of the inspection requirements specified herein, unless disapproved by the Government. The Government reserves the right to perform any of the inspections set forth in the specification where such inspections are deemed necessary to assure supplies and services conform to prescribed requirements.

4.1.1 Quality program. - The contractor shall provide and maintain a quality program acceptable to the Government for supplies and services covered by this specification. The quality program shall be in accordance with MIL-Q-9858.

4.2 Classification of inspections. - Production DPS's shall undergo the following inspections.

Category	Inspection Type
A	Production inspection
B	Production control inspection
C	Reliability tests
D	Maintainability tests
E	Environmental tests

4.2.1 Category A, production inspection. - The production inspection shall consist of:

a. Operating tests.

b. Controls and control circuits test.

c. Warm-up/burn-in tests.

d. Software transferability tests.

The production acceptance procedures shall be as specified in Specification SB-10441-100 and the test data record shall be in accordance with Specification SB-10441-400.

4.2.1.1 Operating test. - The operating test shall include the running of three types of programs: A memory check program, an instruction check program, and an I/O program. All memory locations, all options of all instructions, and all terminated I/O channels shall be thoroughly checked, using worst-case formats.

4.2.1.2 Controls and control circuits test. - The DPS controls and control circuits shall be tested to verify compliance with the requirements specified herein.

4.2.1.3 Warm-up/burn-in tests. - The computer shall be subjected to temperature testing to eliminate the infant mortality failures, thus increasing the equipment MTBF thereafter. The tests shall consist of a one hour power off period at 0°C and after a maximum of 2 minutes warm-up under the power-on condition, the computer will be placed in the operating mode and perform an operating test for 48 hours under the conditions of temperature cycling between 0° and +50°C. The cycle shall be as follows and shall be repeated four times.

a. Four hours at 0°C.

b. Two hours transition from 0° to +50°C.

c. Four hours at +50°C.

d. Two hours transition from +50°C to 0°C.

4.2.1.4 Software transferability test. - The software transferability test shall demonstrate that any program written for functional demonstration models or preproduction models of the DPS will operate correctly, with no changes whatsoever, on production DPS's. This test shall include as a minimum, utilization of all DPS instructions, DPS input-output, DPS interrupts, and all DPS memory modules.

4.2.2 Category B, production control inspection. - The production control inspections shall consist of the following tests:

a. Weights and dimensions.

b. Supply line voltage and frequency.

c. Heat test.

d. Enclosure test.

e. Power.

f. EMI, interference, emission and susceptibility.

4.2.2.1 Weights and dimensions. - The weights and overall dimensions for the DPS shall be verified.

4.2.2.2 Supply line voltage and frequency. - The DPS shall be functionally operated over the supply steady state voltage and frequency ranges. The DPS shall be operated 15 minutes minimum at each of the four worst case conditions of voltage and frequency.

4.2.2.2.1 Transient voltage test. - With the DPS operating with the supply voltage at the upper steady state voltage tolerance limit, a transient voltage of +20 percent of nominal, recovering to the steady-state band within 2 seconds, shall be superimposed. With the DPS operating with the supply voltage at the lower steady state voltage tolerance limit, a transient voltage of -20 percent of nominal, recovering to the steady-state band within 2 seconds, shall be superimposed. Momentary impairment of operation shall be permissible. No part failure shall occur and the DPS shall recover automatically.

4.2.2.2.2 Transient frequency test. - With the DPS operating with the supply 3 percent above nominal frequency, a transient frequency of +3 percent of nominal, recovering to the steady-state band within 2 seconds, shall be superimposed. With the DPS operating with the supply 3 percent below nominal frequency, a transient frequency of -3 percent of nominal, recovering to the steady-state within 2 seconds, shall be superimposed.

4.2.2.2.3 Power interruption test. - The power interruption test shall be performed as specified in MIL-E-16400.

4.2.2.3 Heat test. - The DPS shall be operated with the supply voltage and frequency at nominal and with its cooling system in normal operation at an ambient temperature of +50° C for 8 hours.

4.2.2.4 Enclosure test. - The enclosure test shall be as specified in MIL-STD-108 for dripproof (45°) equipment using the actual drip test.

4.2.2.5 Power test. - The DPS shall have the supply line power required and the power factor measured.

4.2.2.6 EMI interference, emission, and susceptibility. - The DPS EMI interference emission, and susceptibility tests shall be as specified in MIL-STD-462 for the following tests:

a. CE03

b. RE02

c. RS03

4.2.3 Category C, Reliability tests. - The DPS shall be tested to verify the reliability prediction. The 2000 hour specified MTBF shall be demonstrated in accordance with test level C of MIL-STD-781 modified as follows:

a.	Test time	4600 operating hours
b.	Failures allowed	3
c.	Producer's risk (L)	0.2
d.	Consumer's risk (S)	0.3
e.	Discrimination ratio	2.07
f.	Temperature cycling	Continuous between an ambient of 0°C for two hours and 50°C for six hours with a maximum transition rate of 5°C per minute.
g.	On-off cycle	On for 3 3/4 hours, off for 1/4 hour.
h.	Vibration	None.
i.	Operation	An operating test consisting of a memory test, I/O test, add instruction test shall be continuously cycled while power is on.

4.2.4 Category D, maintainability test. - The DPS corrective maintenance requirements shall be demonstrated by replacement of LRI's and chassis mounted electronic, electrical, electromechanical, and mechanical parts or components. Test method 2 of MIL-STD-471 shall be used to demonstrate the corrective maintenance indices. A beta of 5 percent shall be used in demonstrating the Mct and Mpt. The 95th percentile point shall be used in demonstrating M max. The supplier shall provide a list of 100 faults for the Mct and M max demonstration. The procuring activity shall use this list to select a sample of 50 faults to be used in this test. A diagnostic program shall be used to detect and identify failures.

4.2.5 Category E, environmental tests. - The environmental tests shall consist of:

a. Temperature tests

b. Humidity tests

c. Shock tests

d. Vibration

e. Inclination

f. Noise test

g. Salt spray test

4.2.5.1 Temperature test. - The temperature test shall be the nonoperating and Class 4 operating condition tests specified in MIL-E-16400.

4.2.5.2 Humidity test. - Except that the maximum temperature shall be +50°C, the humidity test shall be as specified in MIL-E-16400.

4.2.5.3 Shock test. - The shock test shall be as specified in MIL-S-901 for Grade A, Type A, Class I equipment. The DPS shall be operated before, during, and after the test.

4.2.5.4 Vibration. - The vibration test shall be the Type I vibration test specified in MIL-STD-167. The DPS shall be operated before, during, and after the test.

4.2.5.5 Inclination. - The inclination test shall be as specified in MIL-E-16400. The DPS shall be operated before, during, and after the test.

4.2.5.6 Noise test. - The noise test shall be in accordance with MIL-STD-740 for Grade A, Type 3 equipment.

4.2.5.7 Salt spray test. - The salt spray test shall be in accordance with FED-STD-151, method 811.1. Test duration shall be 200 hours for the frame and enclosure, and 48 hours for the parts.

4.3 Frequency of testing. - Every DPS, including those to undergo Category B, C, D, or E testing, shall undergo the category A production inspection. DPS's to undergo Category B, C, D, or E testing shall be as specified in the contract (see 6.1).

Sheet 103 is intentionally blank.

Sheet 104 is intentionally blank.

5. PREPARATION FOR DELIVERY

5.1 Cleaning, preservation, packaging, packing and marking. - The equipment, accessories, technical publications, and repair parts shall be cleaned, preserved, packaged, packed, and marked in accordance with MIL-E-17555 as specified in the contract or order.

6. NOTES

6.1 Ordering data. - The procurement document should specify the following:

a. Which primary power source is required (see 3.2).

b. Whether the main memory is to be single-port or two-port with DMA (see 3.4.4.1).

c. Size of main memory required (see 3.4.4.1.3).

*d. NDRO contents, or peripheral equipment, bootstrap position, and I/O channel for initial loading (see 3.4.4.3).

**e. Program or program specification for the user-defined ROM (see 3.4.4.3.1).

f. Number of serial and parallel I/O channels and channel assignments (see 3.4.5.1).

g. If parallel channels, specify the interface for each group (see 3.4.5.3).

h. If dual channel operation is required specify the dual channel channels (see 3.4.5.3.1).

i. If intercomputer channels are required specify the channel number(s) (see 3.4.5.3.2).

j. If ESA operation is required specify the ESA channels (see 3.4.5.3.1.1).

k. If serial channels are required specify the type or types (see 3.4.5.4).

*If nonstandard, separate authorization is required to develop and implement.
**Requires separate authorization to develop and implement.

l. For asynchronous serial channels, specify which 4 modulation rates are to be available (see 3.4.5.3.1.2 and 3.4.5.3.2.2).

m. If the second general register set is required (see 3.4.6.1).

n. If the math pack option is required (see 3.4.4.3.3).

o. If peripheral input channels are required, specify the channel number(s) (see 3.4.5.3.3).

p. Number of DPS's to undergo Category B, C, D, or E testing, if any (see 4.3).

6.2 Definitions. -

6.2.1 Bits. - Bits referred to by numbers are 2 to that power. For example, bit 8 = 2^8.

6.2.2 Set. - Set is a binary 1 condition, and for an indicator or an indicator-switch means that the indicator illuminates or the indicator switch is on.

6.2.3 Clear. - Clear is a binary 0 condition, and for an indicator or an indicator-switch means that the indicator extinguishes or the indicator switch is off.

6.2.4 Page. - A page consists of 1024 consecutive memory locations such that each page ending address is $1777_8 + n(2000_8)$ where n = 0, 1, 2, 3, ..., 77_8.

6.3 Recommended cables. -

6.3.1 MIL-STD-188 or VACALES cable. - Cable in accordance with Drawing 7956256-03 (MIL-C-915 2U-19) is recommended for communication up to 100 feet.

6.3.2 RS-232 cable. - Cable in accordance with Drawing 7957138-00 is recommended for communication up to 50 feet.

6.3.3 -15V (DS 4772) interface cables. - Cables in accordance with Drawings 7956256-00 (MIL-C-915 2U-45), 7956687 (MIL-C-915 2AU-40), and 7956785 (MIL-C-915 2U-60) are recommended for communication up to 300 feet.

6.3.4 +3.5V (DS 4772) interface cables. - Cables in accordance with Drawings 7956256-00 (MIL-C-915 2U-45) and 7956785 (MIL-C-915 2U-60) are recommended for communication up to 100 feet. Cable in accordance with Drawing 7956687 (MIL-C-915 2AU-40) is recommended for communication up to 250 feet.

6.3.5 -3V (DS 4772) interface cables. - Cables in accordance with Drawings 7956256-00 (MIL-C-915 2U-45) and 7956785 (MIL-C-915 2U-60) are recommended for communication up to 200 feet. Cable in accordance with Drawing 7956687 (MIL-C-915 2AU-40) is recommended for communication up to 300 feet.

6.3.6 NTDS serial cable. - Cable in accordance with Drawing 7956906-00 (RG-12, See SB-12407) is recommended for communication up to 1000 feet.

APPENDIX

APPENDIX 10

Interface Arrangement

10.1 Scope. - This appendix establishes the physical interface arrangements for the parallel, MIL-STD-188, RS-232, VACALES and NTDS serial interfaces used in the DPS.

10.2 Jack assignments. - The jack assignments shall be as specified in table 10-I. The part numbers of mating connectors shall be as specified below.

Parallel channels	Univac	Federal Stock No.
Output	7101943-03	
Input	7101943-02	
NTDS serial channels		
Output	7101943-01	*
Input	7101943-00	
MIL-STD-188 and VACALES mode serial channels	7101943-05	
RS-232 serial channels	7101943-06	
DMA channel	7101943-04	
External RTC	7901743-00	
Input Power, 60 Hz.	905411-06	
Input Power, 400 Hz.	905411-04	
Output Dual Channel Jumper Plug	7126375-01	
Input Dual Channel Jumper Plug	7126375-00	
Connector Cap	7084897-00	

*To be determined.

Table 10-I. Jack Assignments

Jack Ref. Desig.	Channel No. or Type	Interface Type			
		Parallel	NTDS Serial	MIL-STD-188 Serial and VACALES	RS-232 Serial
J1	4	output	output	-	-
2	4	input	input	I/O	I/O
3	10	output	output	-	-
4	10	input	input	I/O	I/O
5	14	output	output	-	-
6	14	input	input	I/O	I/O
7	5	output	-	-	-
8	5	input	-	I/O	I/O
9	11	output	-	-	-
10	11	input	-	I/O	I/O
11	15	output	-	-	-
12	15	input	-	I/O	I/O
13	6	output	output	-	-
14	6	input	input	I/O	I/O
15	12	output	output	-	-
16	12	input	input	I/O	I/O
17	16	output	output	-	-
18	16	input	input	I/O	I/O
19	7	output	-	-	-
20	7	input	-	I/O	I/O
21	13	output	-	-	-
22	13	input	-	I/O	I/O
23	17	output	-	-	-
24	17	input	-	I/O	I/O
25	0	input	input	I/O	I/O
26	1	input	-	I/O	I/O
27	2	input	input	I/O	I/O
28	3	input	-	I/O	I/O
29	0	output	output	-	-
30	1	output	-	-	-
31	2	output	output	-	-
32	3	output	-	-	-
33	DMA				
J34	External RTC				
J35	Input Power				

10.3 Connector pin assignments. - All input, output, discretes, and control signals for one channel shall be available at one interface connector. The connector pin assignments shall be as specified in tables 10-II, 10-III, 10-IV, 10-V, 10-VI, 10-VII, 10-VIII, and 10-IX.

10.4 Keying. - The interface connector shall have one fixed key arrangement to permit ready interchange of cables within any system which uses this method.

10.5 Cable. - Cable, as used herein, does not include connectors. Recommended cables are described in 6.3.

Table 10-II. Connector Pin Assignment
Parallel I/O Channels

Function		Pin Assignment	
Input	Output	Signal	Return
IDR	Out Ack	B-5	A-5
In Ack	ODR	B-6	A-6
EIR	EFA	B-7	A-7
EIE	EFR	B-8	A-8
	Data Bit 00	D-1	C-1
	01	D-2	C-2
	02	D-3	C-3
	03	D-4	C-4
	04	D-5	C-5
	05	D-6	C-6
	06	D-7	C-7
	07	D-8	C-8
	08	D-9	C-9
	09	D-10	C-10
	10	D-11	C-11
	11	D-12	C-12
	12	G-1	H-1
	13	G-2	H-2
	14	G-3	H-3
	15	G-4	H-4
Used in Dual-Channel Modes on Even Numbered Channel	16	G-5	H-5
	17	G-6	H-6
	18	G-7	H-7
	19	G-8	H-8
	20	G-9	H-9
	21	G-10	H-10
	22	G-11	H-11
	23	G-12	H-12
	24	J-1	K-1
	25	J-2	K-2
	26	J-3	K-3
	27	J-4	K-4
	28	J-5	K-5
	29	J-6	K-6
	30	J-7	K-7
	Data Bit 31	J-8	K-8
	Shield	-	B-1
	Spare	B-2	A-2
	Spare	B-3	A-3
	Spare	B-4	A-4

Table 10-III Connector Pin Assignment, MIL-STD-188, VACALES, and RS-232 Channels

Pin Number		MIL-STD-188	RS-232	VACALES
* Group A	** Group B			
D8	G4	A	Loop Test	Loop Back
D4	D12	B	Ring Indicator	B
C4	C12	C	Received Line Signal Detect	Carrier Detect
C8	H4	D	Data Terminal Ready	D
D5	G1	E	Clear to Send	Sync Error
D7	G3	F	New Sync.	F
C7	H3	G	Request to Send	G
D6	G2	H	-	Transmitter Prep
D3	D11	I	-	Alarm Indicate
C6	H2	J	-	J
C3	C11	K	Data Set Ready	Receive on Full
D2	D10	L	-	Transmit on Full
B5	B5	Transmit Clock	Transmitter Signal Element Timing	Transmit Clock
A5	A5	Transmit Data	Transmitted Data	Transmit Data
A7	A7	Receive Clock	Receiver Signal Element Timing	Receive Clock
B7	B7	Receive Data	Received Data	Receive Data
A6	A6	Signal Ground	Signal Ground	Signal Ground
B1	B1	Cable Shield	Cable Shield	Cable Shield

* Group A = Channels 0,1; 4,5; 10,11; and 14,15.

** Group B = Channels 2,3: 6,7; 12,13; and 16,17.

Table 10-IV. Connector Pin Assignment, NTDS Serial Channels

Pin Number	
Signal	Return
B08	A08

Table 10-V. Connector Pin Assignment, Single Phase Input Power

Function	Pin Number
Vac	J35-A
Common	J35-B
Ground	J35-G

Table 10-VI. Connector Pin Assignment, Three Phase Input Power

Function	Pin Number
Vac	J35-A
Vac	J35-B
Vac	J35-C
Common *	J35-D
Ground	J35-G

* Pin is not used for three phase delta.

Table 10-VII. Connector Pin Assignment
DMA Channel

Function		Pin Assignment	
		Signal	Return
Data Bit	00	K-2	K-1
	01	J-2	J-1
	02	H-2	H-1
	03	G-2	G-1
	04	F-2	F-1
	05	E-2	E-1
	06	D-2	D-1
	07	C-2	C-1
	08	K-5	K-4
	09	J-5	J-4
	10	H-5	H-4
	11	G-5	G-4
	12	F-5	F-4
	13	E-5	E-4
	14	D-5	D-4
	15	C-5	C-4
Write Upper		A-5	A-4
Write Lower		A-2	A-1
Read Initiate		K-8	K-7
Write Initiate		J-8	J-7
Full Cycle		H-8	H-7
Data Available		G-8	G-7
Address Bit	00	F-8	F-7
	01	E-8	E-7
	02	D-8	D-7
	03	C-8	C-7
	04	B-8	B-7
	05	A-8	A-7
	06	K-11	K-10
	07	J-11	J-10
	08	H-11	H-10
	09	G-11	G-10
	10	F-11	F-10
	11	E-11	E-10
	12	D-11	D-10
	13	C-11	C-10
	14	B-11	B-10
	15	A-11	A-10
Not Used		B-2	B-1
Not Used		B-5	B-4

Table 10-VIII. Dual Channel Plug Pin Assignments

Origin	Destination	Function
D-1	F-1	Data Bit 00
C-1	E-1	Return
D-2	F-2	Data Bit 01
C-2	E-2	Return
D-3	F-3	Data Bit 02
C-3	E-3	Return
D-4	F-4	Data Bit 03
C-4	E-4	Return
D-5	F-5	Data Bit 04
C-5	E-5	Return
D-6	F-6	Data Bit 05
C-6	E-6	Return
D-7	F-7	Data Bit 06
C-7	E-7	Return
D-8	F-8	Data Bit 07
C-8	E-8	Return
D-9	F-9	Data Bit 08
C-9	E-9	Return
D-10	F-10	Data Bit 09
C-10	E-10	Return
D-11	F-11	Data Bit 10
C-11	E-11	Return
D-12	F-12	Data Bit 11
C-12	E-12	Return
G-1	B-9	Data Bit 12
H-1	A-9	Return
G-2	B-10	Data Bit 13
H-2	A-10	Return
G-3	B-11	Data Bit 14
H-3	A-11	Return
G-4	B-12	Data Bit 15
H-5	A-12	Return

Table 10-IX. External Realtime Clock Pin Assignments

Pin Number	Function
A	Spare
B	Spare
C	Clock Signal Return
D	Clock Signal
E	Spare
F	Spare

APPENDIX 20

DMA Memory Interface

20.1 Scope. - This section describes the interface lines, voltage levels, and cycle timing of the DMA memory interface.

20.2 DMA memory interface. - The DMA memory port shall have the interface lines specified in figure 20-1.

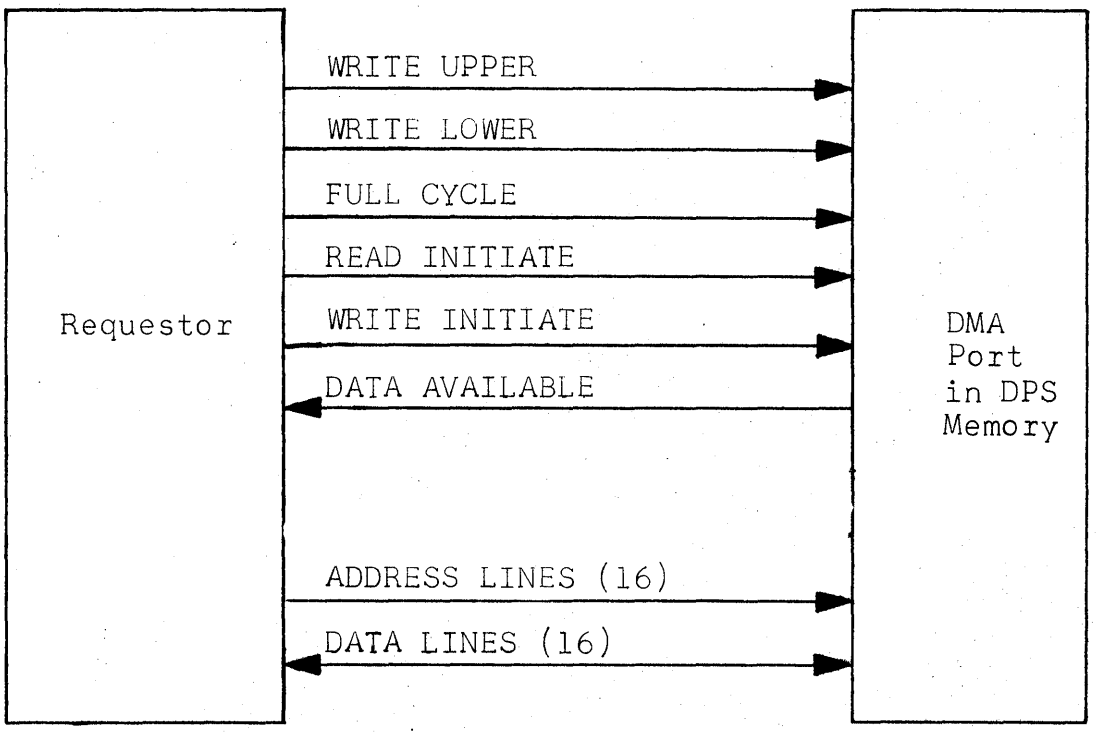


Figure 20-1. DMA Port Interface Signals

20.2.1 Read-write control signal lines. - The read and write operations shall be selected using the WRITE UPPER and and WRITE LOWER lines which shall be required to be set and cleared by the requestor.

20.2.1.1 Write upper byte. - The WRITE UPPER line shall be set to specify that bits 8-15 be written.

20.2.1.2 Write lower byte. - The WRITE LOWER line shall be set to specify that bits 0-7 be written.

20.2.1.3 Full word write. - The WRITE UPPER line and WRITE LOWER line shall both be set to specify that a full word be written.

20.2.1.4 Read operation. - The WRITE UPPER line and WRITE LOWER line shall be cleared to specify that a read operation be performed. Read operations shall always be full word reads.

20.2.2 FULL CYCLE line. - The FULL CYCLE line shall be required to be set and cleared by the requestor. The FULL CYCLE line shall be set to select the read-write cycle. The FULL CYCLE line shall be cleared to select the read-modify-write cycle.

20.2.3 READ INITIATE line. - The READ INITIATE line shall be used to specify that the selected full cycle operation be performed or the first half (read portion) of the read-modify-write operation be performed. The READ INITIATE line shall be required to be set and cleared by the requestor.

20.2.4 WRITE INITIATE line. - The WRITE INITIATE line shall be used for the write portion of the read-modify-write operation. The WRITE INITIATE line shall be required to be set and cleared by the requestor.

20.2.5 DATA AVAILABLE line. - The DATA AVAILABLE line shall specify that data from memory is available on the DATA lines and shall be required to be set and cleared by the DMA memory. When pulsed, the DATA AVAILABLE line shall also serve as a memory acknowledge. The DATA AVAILABLE line shall be used in all modes.

20.2.6 ADDRESS lines. - The 16 ADDRESS lines shall specify the address within the memory bank and shall be required to be set and cleared by the requestor.

20.2.7 DATA lines. - The 16 DATA lines shall be bidirectional lines, each of which shall be used to transmit one data bit.

20.3 DMA memory cycles. - The DMA memory shall be available with read-write and read-modify-write cycles.

20.3.1 Read-write cycle. - The read-write cycle shall be used for normal read or write operations.

20.3.2 Read-modify-write cycle. - The read-modify-write cycle shall consist of a short cycle read operation followed by a short cycle write operation. When the DMA port has accepted the read operation, the requestor shall be required to complete the short cycle write operation before the DMA memory shall accept a request from the CP port.

20.4 DMA interface. - The DMA interface shall be characterized by nominal values of 0 volts differential and +2.5 volts differential on a balanced, terminated line to represent binary one (set) and binary zero (clear) respectively and by a switching threshold between +1.0 and +2.0 volts differentially.

20.4.1 DMA input amplifiers. - Each input amplifier shall have the following characteristics:

a. The output of the circuit shall switch from binary zero to binary one whenever the input differential signal changes in the negative direction through the range of +2.0 volts to +1.0 volts.

b. The output of the circuit shall switch from binary one to binary zero whenever the input differential signal changes in the positive direction through the range of +1.0 volts to +2.0 volts.

c. The output of the circuit shall not switch as a result of any input transient-pulse signal that has an amplitude between +3.0 volts and -3.0 volts if its duration and amplitude are common to both sides of the line (common mode).

d. The output of the circuit shall be binary zero whenever its input is open circuited.

e. The output of the circuit shall be binary zero whenever the steady state input signal is more positive than +2.0 volts.

f. The output of the circuit shall be binary one whenever the steady state input signal is more negative than +1.0 volts.

g. The input circuit shall present a terminal impedance to the line equivalent to a resistance of 3k ohms minimum.

h. The input resistances of the signal input and return input terminals shall be matched to within ± 8 percent.

20.4.2 DMA output drivers. - Each interface line driver circuit shall have the following characteristics when driving a balanced twisted pair with any characteristic impedance between 100 ohms and 150 ohms. Figure 20-2 specifies the balanced line biasing scheme.

a. The steady state output voltage representing a binary one shall be between 0.0 and +0.40 volt differential. The output driver shall sink a current of 48 milliamperes at the +0.40 volt level.

b. The steady state driver output representing a binary zero shall be an impedance across the lines of 100 k ohms minimum.

c. The driver propagation delay and output voltage fall time (90% to 10%) shall be less than 40 nanoseconds.

d. The driver propagation delay and output voltage rise time (10% to 90%) shall be less than 40 nanoseconds.

e. The circuit shall be capable of driving the line terminations and input amplifier while switching, as well as in the steady state condition.

f. Whenever power to a control line driver circuit is removed, the driver shall present not less than 100 k ohms impedance to the line.

g. DMA output drivers shall withstand transient common mode noise between +3.0 and -3.0 volts amplitude and requestor drivers shall withstand transient common mode noise between +1.0 and -1.0 volts amplitude without causing degradation of the data state on the line.

20.5 Cycle timing. - The timing for the DMA interface cycles shall be as specified in figures 20-3 and 20-4.

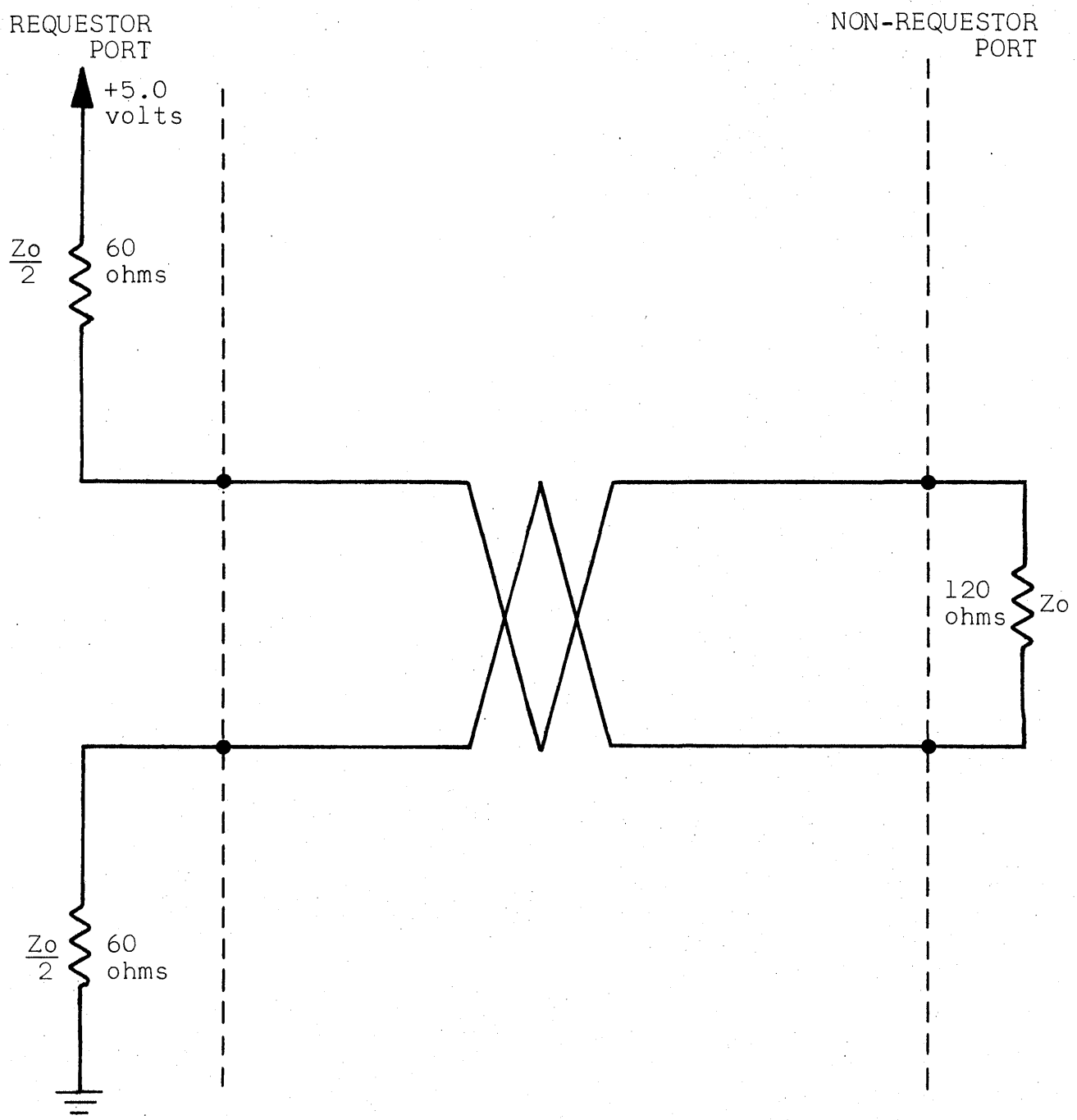


Figure 20-2. Differential Balanced Line Biasing

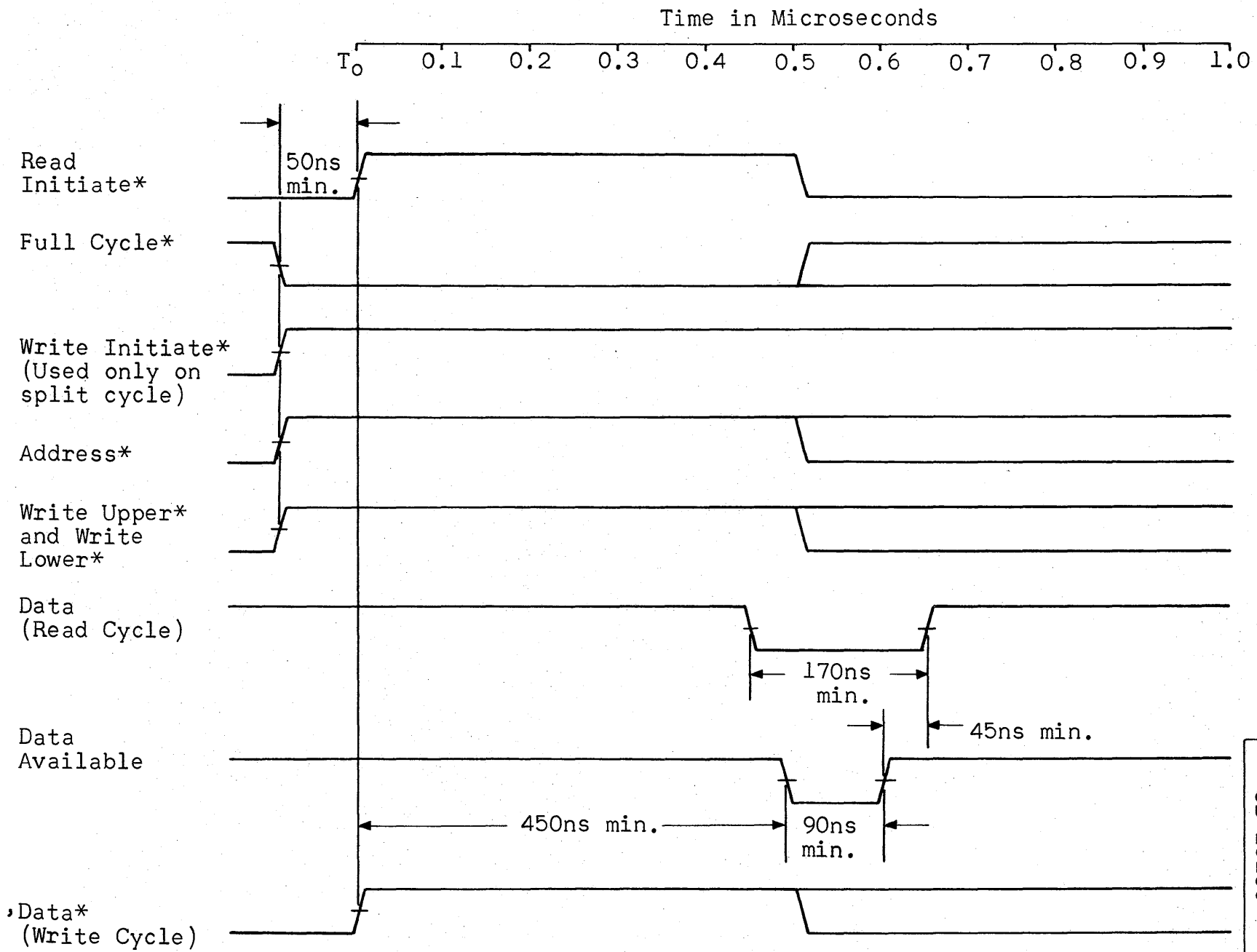


Figure 20-3. DMA Interface Read-Write Cycle

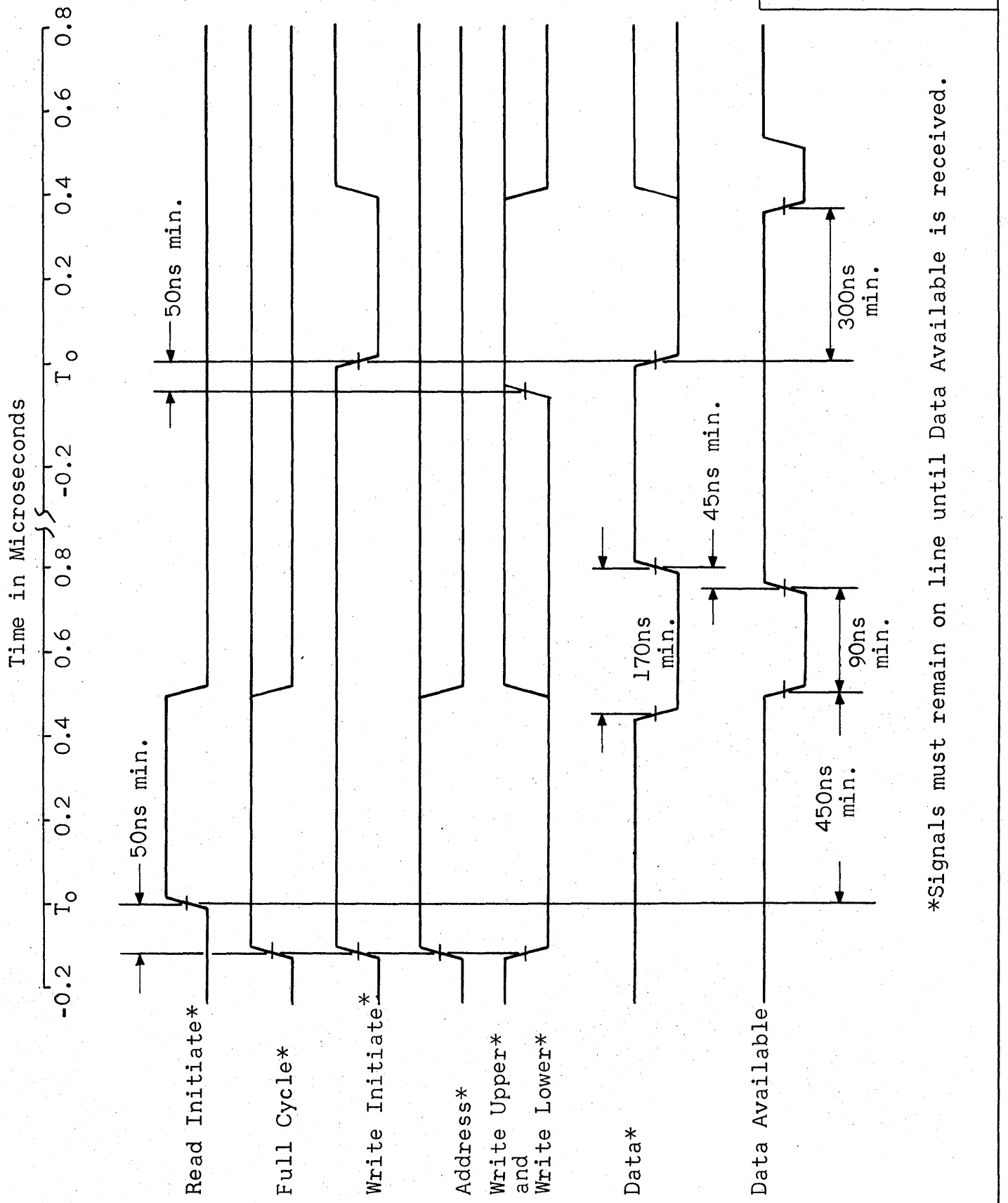


Figure 20-4. DMA Interface Read-Modify-Write Cycle

APPENDIX 30

Instruction Repertoire

30.1 Scope. - This appendix shall establish the register operations performed for each instruction.

30.2 Operation code 00. - The operation code 00 instructions shall be as follows:

a. Format RR - This instruction shall transfer the contents of General Register 17_8 to the uP register, when the DIAGNOSTIC JUMP switch on the Maintenance Panel is Up. Otherwise, this instruction is unassigned.

b. Format RI - Unassigned

c. Format RK - Unassigned

d. Format RX, Byte Load - This instruction shall load the selected byte from memory address Y in bits 0 through 7 of R_a, clear bits 8 through 15 and set the Condition Code.

30.3 Operation code 01. - The operation code 01 instructions shall be as follows:

a. Format RR, Load - This instruction shall load (R_m) in R_a and set the Condition Code.

b. Format RI Type 2, Load - This instruction shall load the contents of memory address Y in R_a and set the Condition Code.

c. Format RK, Load - This instruction shall load the operand Y in R_a and set the Condition Code.

d. Format RX, Load - This instruction shall load the contents of memory address Y in R_a and set the Condition Code.

30.4 Operation code 02. - The operation code 02 instructions shall be as follows:

a. Format RR, Unary-Arithmetic - This instruction shall perform the operation specified by the m-value as specified in table 30-I, and then set the Condition Code in accordance with the resultant quantity in the register specified by the a-designator.

b. Format RI Type 2, Load Double - This instruction shall load the contents of memory addresses Y and Y + 1 in R_a and R_{a+1} respectively and set the Condition Code.

Table 30-I. Unary-Arithmetic Instruction m Values

m Value	Operation	Description
0	Make Positive	If (R_a) is negative, perform the twos complement of (R_a) and store the result in R_a. When the maximum negative number (1000000000000000) is complemented, set the overflow designator. If (R_a) is positive, do not change (R_a).
1	Make Negative	If (R_a) is positive and not zero perform the twos complement of (R_a) and store the result in R_a. If (R_a) is negative or zero, do not change (R_a).
2	Round R_a	Add bit 15 of R_{a+1} to (R_a) and store the result in R_a, R_a must be even.
3	Unassigned	
4	Twos Complement, Single	Perform the twos complement of (R_a) and store the result in R_a. When the maximum negative number is complemented, set the overflow designator.
5	Twos Complement, Double	Perform the twos complement of double length (R_a, R_{a+1}), and store the result in R_a and R_{a+1}. When the maximum negative number is complemented, set the overflow designator.
6	Ones Complement, Single	Perform the bit-by-bit logical complement of (R_a) and store the result in R_a.
7	Unassigned	

Table 30-I. Unary-Arithmetic Instruction m Values (continued)

m Value	Operation	Description
10	Increment R_a by 1	Increment (R_a) by 1 and store the result in R_a.
11	Decrement R_a by 1	Decrement (R_a) by 1 and store the result in R_a.
12	Increment R_a by 2	Increment (R_a) by 2 and store the result in R_a.
13	Decrement R_a by 2	Decrement (R_a) by 2 and store the result in R_a.
14	Unassigned	
15	Unassigned	
16	Unassigned	
17	Unassigned	

c. Format RK - Unassigned

d. Format RX, Load Double - This instruction shall load the contents of memory addresses Y and Y + 1 in R_a and R_{a+1} respectively and set the Condition Code.

30.5 Operation code 03. - The operation code 03 instructions shall be as follows:

a. Format RR, Unary-Control - This instruction shall perform the operation specified by the m-value as specified in table 30-II and set the Condition Code for m = 0-3, and 15.

b. Format RI - Unassigned

c. Format RK - Unassigned

d. Format RX, Load Multiple - This instruction shall load the contents of sequential memory addresses beginning at Y, in sequential registers beginning at R_a and ending at R_m. If a is greater than m, the registers loaded shall be R_a, R_{a+1}, ..., R17, R_0 ... R_m. In this instruction Y shall be equal to y (no indexing or indirect addressing shall be performed).

30.6 Operation code 04. - The operation code 04 instructions shall be as follows:

a. Format RR, Unary-Shift - This instruction shall perform the operation specified by the m-value as specified in table 30-III, and set the Condition Code.

b. Format RI - Unassigned

c. Format RK - Unassigned

d. Format RX, Byte Load and Index by 1 - This instruction shall perform the following sequence of operations:

(1) Formulate memory address Y.

(2) Increment (R_m) by one.

(3) Load the selected byte from memory address Y into bits 0 through 7 of R_a, clear bits 8 through 15 and set the Condition Code.

Table 30-II. Unary-Control Instruction m Values

m Value	Operation	Description
0	Executive return	When Class II interrupt lockout is not set, generate an Executive Return Interrupt and store (P) + 1 in R_a. When Class II interrupt lockout is set, no operation.
1	Store STATUS #1 Register	Store the contents of the STATUS #1 Register in R_a.
2	Store STATUS #2 Register	Store the contents of the STATUS #2 Register in R_a.
3	Store RTC Lower	Store the lower 16 bits of the Real Time Clock Register into R_a.
4	Load P	Load (R_a) in P.
5	Load STATUS #1	Load (R_a) in the STATUS #1 Register.
6	Load STATUS #2 Register	Load (R_a) in the STATUS #2 Register.
7	Load RTC Lower	Load (R_a) in the lower 16 bits of the Real Time Clock Register.
10	Enable RTC	Enable the Real Time Clock Register to increment by one for each cycle of the clock source. Generate the RTC Overflow Interrupt when the contents of the lower half (bits 0 through 15) of the Real Time Clock Register changes from all ones to all zeros.
11	Disable RTC	Disable the Real Time Clock Register from incrementing. The clock source continues to operate.

Table 30-II. Unary-Control Instruction m Values (Continued)

m Value	Operation	Description
12	Load and Enable Monitor Clock	Load (R_a) in the Monitor Clock Register and enable the Monitor Clock Register to decrement by one for each cycle of the clock source. Generate the Monitor Clock interrupt when the contents of the Monitor Clock Register equals zero.
13	Disable Monitor Clock	Disable Monitor Clock and Monitor Clock Interrupt
14	Load RTC Double	Load (R_a, R_{a+1}) in the Real Time Clock Register and enable the Real Time Clock Register to increment by one for each cycle of the clock source. Enable-Disable RTC Interrupt is not affected.
15	Store RTC Double	Store the contents of the Real Time Clock into R_a and R_{a+1}.
16	Enable RTC Interrupt	Enable generation of the RTC Overflow Interrupt when the contents of the lower half (bits 0 through 15) of the Real Time Clock Register changes from all ones to all zeros.
17	Disable RTC Interrupt	Disable generation of the RTC Overflow Interrupt.

Table 30-III. Unary-Shift Instruction m Values

m Value	Operation	Description
0	Square Root (optional)	Determine the square root of (R_a, R_{a+1}). Store the result (root) in R_{a+1} and the residue in R_a. The residue shall equal the number minus the root squared. The residue in R_a shall be treated as a 16-bit magnitude quantity. The largest number that the square root shall be determined for is 7777777777_8. The results of any larger number or a negative number shall set the overflow designator, bit 10 in Status Register 1.
1	Reverse Register	Change (R_a) to the reverse order (see the following example). 1 1 . . . . . 0 1 Initial 1 0 . . . . . 1 1 Final
2	Count Ones	Count the number of set bits in (R_a), and store the count in R_{a+1}.
3	Scale Factor	Shift the double length (R_a, R_{a+1}) left algebraically, with zeros extended to fill, until bit 15 of R_a does not equal bit 14 of R_a and store the shift count in R_{a+2}. The shift limit shall be 31 (decimal) shifts.
4 - 17		Unassigned

30.7 Operation code 05. - The operation code 05 instructions shall be as follows:

a. Format RR, Set Bit - This instruction shall set the bit in R_a corresponding to the value of m and set the Condition Code. (Example: $m=15_8$, set bit 2^{13} in R_a)

b. Format RI Type 2, Load and Index by 1 - This instruction shall perform the following, in sequence:

(1) Formulate memory address Y.

(2) Increment (R_m) by one.

(3) Load the contents of memory address Y into R_a and set the Condition Code.

c. Format RK - Unassigned.

d. Format RX, Load and Index by 1 - This instruction shall perform the following, in sequence:

(1) Formulate memory address Y.

(2) Increment (R_m) by one.

(3) Load the contents of memory address Y into R_a and set the Condition Code.

30.8 Operation code 06. - The operation code 06 instructions shall be as follows:

a. Format RR, Clear Bit - This instruction shall clear the bit in R_a corresponding to the value of m and set the Condition Code.

b. Format RI Type 2, Load Double and Index by 2 - This instruction shall perform the following, in sequence:

(1) Formulate the memory address Y.

(2) Increment (R_m) by one.

(3) Load the contents of memory address Y + 1 into R_{a+1}.

(4) Increment (R_m) by one.

(5) Load the contents of memory address Y into R_a.

(6) Set the Condition Code.

c. Format RK - Unassigned.

d. Format RX, Load Double and Index by 2 - This instruction shall perform the following, in sequence:

(1) Formulate the memory address Y.

(2) Increment (R_m) by one.

(3) Load the contents of memory address Y + 1 into R_{a+1}.

(4) Increment (R_m) by one.

(5) Load the contents of memory address Y into R_a.

(6) Set the Condition Code.

30.9 Operation code 07. - The operation code 07 instructions shall be as follows:

a. Format RR, Compare Bit - This instruction shall compare the bit in R_a corresponding to the value m with zero, and set the Condition Code.

b. Format RI Type 2, Load PSW (Program Status Word) - This instruction shall load the contents of memory addresses Y, Y + 1, and Y + 2 in the Program Address Register, Status Register 1 and Status Register 2, respectively.

c. Format RK - Unassigned

d. Format RX, Load PSW (Program Status Word) - This instruction shall load the contents of memory addresses Y, Y+1, and Y + 2 in the Program Address Register, and Status Register 1, and Status Register 2, respectively.

30.10 Operation code 10. - The operation code 10 instructions shall be as follows:

a. Format RR, Logical Right Single Shift - This instruction shall shift (R_a) right n places with zeros extended to fill and set the Condition Code. n shall be the value in bits 0-5 of R_m.

b. Format RI - Unassigned

c. Format RK, Logical Right Single Shift - This instruction shall shift (R_a) right n places with zeros extended to fill and set the Condition Code. n shall be the value in bits 0-5 of operand Y.

d. Format RX, Byte Store - This instruction shall store in the selected byte of memory address Y bits 0-7 of (R_a).

30.11 Operation code 11. - The operation code 11 instructions shall be as follows:

a. Format RR, Algebraic Right Single Shift - This instruction shall shift (R_a) right n places with sign extended to fill and set the Condition Code. n shall be the value in bits 0-5 of R_m.

b. Format RI Type 2, Store - This instruction shall store (R_a) at memory address Y.

c. Format RK, Algebraic Right Single Shift - This instruction shall shift (R_a) right n places with sign extended to fill and set the Condition Code. n shall be the value in bits 0-5 of operand Y.

d. Format RX, Store - This instruction shall store (R_a) at memory address Y.

30.12 Operation code 12. - The operation code 12 instructions shall be as follows:

a. Format RR, Logical Right Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) right n places with zeros extended to fill and set the Condition Code. n shall be the value in bits 0-5 of R_m.

b. Format RI Type 2, Store Double - This instruction shall store (R_a) and (R_{a+1}) at memory addresses Y and Y + 1 respectively.

c. Format RK, Logical Right Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) right n places with zeros extended to fill and set the Condition Code. n shall be the value in bits 0-5 of operand Y.

d. Format RX, Store Double - This instruction shall store (R_a) and (R_{a+1}) at memory addresses Y and Y + 1 respectively.

30.13 Operation code 13. - The operation code 13 instructions shall be as follows:

a. Format RR, Algebraic Right Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) right n places with the sign extended to fill and set the Condition Code. n shall be the value in bits 0-5 of R_m.

b. Format RI - Unassigned.

c. Format RK, Algebraic Right Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) right n places with R_a sign extended to fill and set the Condition Code. n shall be the value in bits 0-5 of operand Y.

d. Format RX, Store Multiple - This instruction shall store in sequential memory addresses beginning at address Y, the contents of sequential registers beginning at R_a and ending at R_m. If a is greater than m the registers stored shall be R_a, R_{a+1}, ..., R_{17}, R_0, ..., R_m. In this instruction Y shall be equal to y (no indexing or indirect addressing shall be performed).

30.14 Operation code 14. - The operation code 14 instructions shall be as follows:

a. Format RR, Algebraic Left Single Shift - This instruction shall shift (R_a) left n places with zeros extended to fill and set the Condition Code. n shall be the value in bits 0-5 of R_m.

b. Format RI - Unassigned.

c. Format RK, Algebraic Left Single Shift - This instruction shall shift (R_a) left n places with zeros extended to fill and set the Condition Code. n shall be the value in bits 0-5 of operand Y.

d. Format RX, Byte Store and Index by 1 - This instruction shall store bits 0-7 of (R_a) in the selected byte at memory address Y; and then increment (R_m) by 1.

30.15 Operation code 15. - The operation code 15 instructions shall be as follows:

a. Format RR, Circular Left Single Shift - This instruction shall shift (R_a) left circular n places and set the Condition Code. n shall be the value of bits 0-5 of R_m.

b. Format RI Type 2, Store and Index by 1 - This instruction shall store (R_a) at memory address Y; and then increment (R_m) by 1.

c. Format RK, Circular Left Single Shift - This instruction shall shift (R_a) left circular n places and set the Condition Code. n shall be the value of bits 0-5 operand Y.

d. Format RX, Store and Index by 1 - This instruction shall store (R_a) at memory address Y; and then increment (R_m) by 1.

30.16 Operation code 16. - The operation code 16 instructions shall be as follows:

a. Format RR, Algebraic Left Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) left n places with zeros extended to fill and set the Condition Code. n shall be the value in bits 0-5 of R_m.

b. Format RI Type 2, Store Double and Index by 2 - This instruction shall perform the following, in sequence:

(1) Formulate the memory address Y.

(2) Store (R_{a+1}) at memory address Y + 1.

(3) Increment (R_m) by one.

(4) Store (R_a) at memory address Y.

(5) Increment (R_m) by one.

c. Format RK, Algebraic Left Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) left n places with zeros extended to fill and set the Condition Code. n shall be the value in bits 0-5 of operand Y.

d. Format RX, Store Double and Index by 2 - This instruction shall perform the following, in sequence:

(1) Formulate the memory address Y.

(2) Store (R_{a+1}) at memory address Y + 1.

(3) Increment (R_m) by one.

(4) Store (R_a) at memory address Y.

(5) Increment (R_m) by one.

30.17 Operation code 17. - The operation code 17 instructions shall be as follows:

a. Format RR, Circular Left Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) left circular n places, with bit 15 of R_a transferred to bit 0 of R_{a+1} in each shift and set the Condition Code. n shall be the value in bits 0-5 of R_m.

b. Format RI Type 2, Store Zeros - This instruction shall store all zeros at memory address Y.

c. Format RK, Circular Left Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) left circular n places with bit 15 of R_a transferred to bit 0 of R_{a+1} in each shift and set the Condition Code. n shall be the value in bits 0-5 of operand Y.

d. Format RX, Store Zeros - This instruction shall store all zeros at memory address Y.

30.18 Operation code 20. - The operation code 20 instructions shall be as follows:

a. Format RR, Subtract - This instruction shall subtract (R_m) from (R_a) and store the result in R_a; and then set the Condition Code.

b. Format RI Type 2, Subtract - This instruction shall subtract the contents of memory address Y from (R_a) and store the result in R_a; and then set the Condition Code.

c. Format RK, Subtract - This instruction shall subtract operand Y from (R_a) and store the result in R_a; and then set the Condition Code.

d. Format RX, Subtract - This instruction shall subtract the contents of memory address Y from (R_a) and store the result in R_a; and then set the Condition Code.

30.19 Operation code 21. - The operation code 21 instructions shall be as follows:

a. Format RR, Subtract Double - This instruction shall subtract the double length (R_m, R_{m+1}) from the double length (R_a, R_{a+1}) and store the result in R_a and R_{a+1}; and then set the Condition Code.

b. Format RI Type 2, Subtract Double - This instruction shall subtract the double length contents of memory addresses Y, Y + 1 from the double length (R_a, R_{a+1}) and store the result in R_a and R_{a+1}; and then set the Condition Code.

c. Format RK - Unassigned.

d. Format RX, Subtract Double - This instruction shall subtract the double length contents of memory addresses Y, Y + 1 from the double length (R_a,R_{a+1}) and store the result in R_a and R_{a+1}; and then set the Condition Code.

30.20 Operation code 22. - The operation code 22 instructions shall be as follows:

a. Format RR, Add - This instruction shall add (R_m) to (R_a) and store the result in R_a; and then set the Condition Code.

b. Format RI Type 2, Add - This instruction shall add the contents of memory address Y to (R_a) and store the result in R_a; and then set the Condition Code.

c. Format RK, Add - This instruction shall add operand Y to (R_a) and store the result in R_a; and then set the Condition Code.

d. Format RX, Add - This instruction shall add the contents of memory address Y to (R_a) and store the result in R_a; and then set the Condition Code.

30.21 Operation code 23. - The operation code 23 instructions shall be as follows:

a. Format RR, Add Double - This instruction shall add the double length (R_m, R_{m+1}) to the double length (R_a, R_{a+1}) and store the result in R_a and R_{a+1}; and then set the Condition Code.

b. Format RI Type 2, Add Double - This instruction shall add the double length contents of memory addresses Y, Y + 1 to the double length (R_a,R_{a+1}) and store the result in R_a and R_{a+1}; and then set the Condition Code.

c. Format RK - Unassigned

d. Format RX, Add Double - This instruction shall add the double length contents of memory addresses Y, Y + 1 to the double length (R_a, R_{a+1}) and store the result in R_a and R_{a+1}; and then set the Condition Code.

30.22 Operation code 24. - The operation code 24 instructions shall be as follows:

a. Format RR, Compare - This instruction shall arithmetically compare (R_a) to (R_m), and set the Condition Code.

b. Format RI Type 2, Compare - This instruction shall arithmetically compare (R_a) to the contents of memory address Y, and set the Condition Code.

c. Format RK, Compare - This instruction shall arithmetically compare (R_a) to operand Y, and set the Condition Code.

d. Format RX, Compare - This instruction shall arithmetically compare (R_a) to the contents of memory address Y, and set the Condition Code.

30.23 Operation code 25. - The operation code 25 instructions shall be as follows:

a. Format RR, Compare Double - This instruction shall arithmetically compare the double length (R_a, R_{a+1}) to the double length (R_m, R_{m+1}) and set the Condition Code.

b. Format RI Type 2, Compare Double - This instruction shall arithmetically compare the double length (R_a, R_{a+1}) to the double length contents of memory addresses Y, Y + 1 and set the Condition Code.

c. Format RK - Unassigned

d. Format RX, Compare Double - This instruction shall arithmetically compare the double length (R_a, R_{a+1}) to the double length contents of memory address Y, Y + 1 and set the Condition Code.

30.24 Operation code 26. - The operation code 26 instructions shall be as follows:

a. Format RR, Multiply - This instruction shall multiply (R_m) by (R_{a+1}) and store the double length result in R_a and R_{a+1}; and then set the Condition Code.

b. Format RI Type 2, Multiply - This instruction shall multiply the contents of memory address Y by (R_{a+1}) and store the double length result in R_a and R_{a+1}; and then set the Condition Code.

c. Format RK, Multiply - This instruction shall multiply operand Y by (R_{a+1}) and store the double length result in R_a and R_{a+1}; and then set the Condition Code.

d. Format RX, Multiply - This instruction shall multiply the contents of memory address Y by (R_{a+1}) and store the double length result in R_a and R_{a+1}; and then set the Condition Code.

30.25 Operation code 27. - The operation code 27 instructions shall be as follows: For all divides, the remainder shall have the same sign as the dividend and the absolute value of the remainder shall be less than the absolute value of the divisor.

a. Format RR, Divide - This instruction shall divide the double length (R_a, R_{a+1}) by (R_m), store the quotient in R_{a+1} and the remainder in R_a; and then set the Condition Code.

b. Format RI Type 2, Divide - This instruction shall divide the double length (R_a, R_{a+1}) by the contents of memory address Y, store the quotient in R_{a+1} and the remainder in R_a; and then set the Condition Code.

c. Format RK, Divide - This instruction shall divide the double length (R_a, R_{a+1}) by operand Y, store the quotient in R_{a+1} and the remainder in R_a; and then set the Condition Code.

d. Format RX, Divide - This instruction shall divide the double length (R_a, R_{a+1}) by the contents of memory address Y, store the quotient in R_{a+1} and the remainder in R_a; and then set the Condition Code.

30.26 Operation code 30. - The operation code 30 instructions shall be as follows:

a. Format RR, AND - This instruction shall perform the bit by bit logical AND of (R_a) and (R_m), and store the result in R_a, then set the Condition Code.

b. Format RI Type 2, AND - This instruction shall perform the bit by bit logical AND of (R_a) and the contents of memory address Y, and store the result in R_a, then set the Condition Code.

c. Format RK, AND - This instruction shall perform the bit by bit logical AND of (R_a) and operand Y, and store the result in R_a, then set the Condition Code.

d. Format RX, AND - This instruction shall perform the bit by bit logical AND of (R_a) and the contents of memory address Y, and store the result in R_a, then set the Condition Code.

30.27 Operation code 31. - The operation code 31 instructions shall be as follows:

a. Format RR, OR - This instruction shall perform the bit by bit logical OR of (R_a) and (R_m), and store the result in R_a, then set the Condition Code.

b. Format RI Type 2, OR - This instruction shall perform the bit by bit logical OR of (R_a) and the contents of memory address Y, and store the results in R_a, then set the Condition Code.

c. Format RK, OR - This instruction shall perform the bit by bit logical OR of (R_a) and operand Y, and store the result in R_a, then set the Condition Code.

d. Format RX, OR - This instruction shall perform the bit by bit logical OR of (R_a) and the contents of memory address Y, and store the result in R_a, then set the Condition Code.

30.28 Operation code 32. - The operation code 32 instructions shall be as follows:

a. Format RR, Exclusive OR - This instruction shall perform the bit by bit exclusive OR of (R_a) and (R_m), and store the result in R_a, then set the Condition Code.

b. Format RI Type 2, Exclusive OR - This instruction shall perform the bit by bit exclusive OR of (R_a) and the contents of memory address Y, and store the result in R_a, then set the Condition Code.

c. Format RK, Exclusive OR - This instruction shall perform the bit by bit exclusive OR of (R_a) and operand Y, and store the result in R_a, then set the Condition Code.

d. Format RX, Exclusive OR - This instruction shall perform the bit by bit exclusive OR of (R_a) and the contents of memory address Y, and store the result in R_a, then set the Condition Code.

30.29 Operation code 33. - The operation code 33 instructions shall be as follows:

a. Format RR, Masked Substitute - This instruction shall perform as follows: For each bit set in (R_{a+1}) the value of the corresponding bit in (R_m) shall be transferrred to the corresponding bit in R_a. For each bit not set in (R_{a+1}) the corresponding bit in R_a is unaltered. Set the Condition Code.

b. Format RI Type 2, Masked Substitute - This instruction shall perform as follows: For each bit set in (R_{a+1}) the value of the corresponding bit in the contents of memory address Y shall be transferred to the corresponding bit in R_a. For each bit not set in (R_{a+1}) the corresponding bit in R_a is unaltered. Set the Condition Code.

c. Format RK, Masked Substitute - This instruction shall perform as follows:

For each bit set in (R_{a+1}) the value of the corresponding bit in operand Y shall be transferred to the corresponding bit in R_a. For each bit not set in (R_{a+1}) the corresponding bit in R_a is unaltered. Set the Condition Code.

d. Format RX, Masked Substitute - This instruction shall perform as follows:

For each bit set in (R_{a+1}) the value of the corresponding bit in the contents of memory address Y shall be transferred to the corresponding bit in R_a. For each bit not set in (R_{a+1}) the corresponding bit in R_a is unaltered. Set the Condition Code.

30.30 Operation code 34. - The operation code 34 instructions shall be as follows:

a. Format RR, Compare Masked - This instruction shall compare the result of the logical AND of (R_a) and (R_{a+1}) to the result of the logical AND of (R_m) and (R_{a+1}) and set the Condition Code. The Condition Code shall be interpreted contingent upon bit 15 of (R_{a+1}) as follows:

(1) When a zero, the Condition Code shall be interpreted as specified in table IV.

(2) When a one, a Condition Code of 00 shall indicate equal, all other Condition Code values shall indicate not-equal.

b. Format RI Type 2, Compare Masked - This instruction shall compare the result of the logical AND of (R_a) and (R_{a+1}) to the result of the logical AND of contents of memory address Y and (R_{a+1}) and set the Condition Code. The Condition Code shall be interpreted contingent upon bit 15 of (R_{a+1}) as follows:

(1) When a zero, the Condition Code shall be interpreted as specified in table IV.

(2) When a one, a Condition Code of 00 shall indicate equal, all other Condition Code values shall indicate not-equal.

c. Format RK, Compare Masked - This instruction shall compare the result of the logical AND of (R_a) and (R_{a+1}) to the result of the logical AND of operand Y and (R_{a+1}) and set the Condition Code. The Condition Code shall be interpreted contingent upon bit 15 of (R_{a+1}) as follows:

(1) When a zero, the Condition Code shall be interpreted as specified in table IV.

(2) When a one, a Condition Code of 00 shall indicate equal, all other Condition Code values shall indicate not-equal.

d. Format RX, Compare Masked - This instruction shall compare the result of the logical AND of (R_a) and (R_{a+1}) to the result of the logical AND of contents of memory address Y and (R_{a+1}) and set the Condition Code. The Condition Code shall be interpreted contingent upon bit 15 of (R_{a+1}) as follows:

(1) When a zero, the Condition Code shall be interpreted as specified in table IV.

(2) When a one, a Condition Code of 00 shall indicate equal, all other Condition Code values shall indicate not-equal.

30.31 Operation code 35. - The operation code 35 instructions shall be as follows:

a. Format RR, I/O Command - This instruction shall cause the DPS to execute the I/O command instruction from main memory address 000140 and clear bits 14 and 15 at this address.

b. Format RI Type 2, Biased Fetch - This instruction shall:

(1) Examine the contents of memory address Y.

(2) Set the Condition Code.

(3) Set the most significant two bits of the contents of memory address Y leaving the remaining bits unchanged.

c. Format RK, Execute Remote - This instruction shall cause the next instruction executed to be that specified by the contents of memory address Y. After executing the instruction the DPS shall continue with the next sequential instruction (after the Execute Remote instruction).

d. Format RX, Biased Fetch - This instruction shall:

(1) Examine the contents of memory address Y.

(2) Set the Condition Code.

(3) Set the most significant two bits of the contents of memory address Y leaving the remaining bits unchanged.

30.32 Operation code 36. - Unassigned.

30.33 Operation code 37. - The operation code 37 instruction shall be as follows:

a. Format RR, Trigonometric and Hyperbolic - This instruction shall perform trigonometric rotate and vector, and hyperbolic rotate and vector functions as specified in Appendix 50. The input parameters (X, Y, and angle) and the output results shall be contained in R_a, R_{a+1}, and R_{a+2}.

b. Format RI - Unassigned.

c. Format RK - Unassigned.

d. Format RX - Unassigned.

30.34 Operation code 40. - The operation code 40 instructions shall be as follows:

a. Format RR, Condition Jump - This instruction shall test for the condition specified by the a-value as specified in table 30-V and perform one of the following:

(1) If the condition specified in table 30-IV is met, jump to the instruction located at the address specified by (R_m). If the a-value specifies a Stop (unconditional) or a Stop Key set operation, the DPS shall halt the RTC, then stop. When the DPS is restarted, load P with (R_m) and execute the instruction at P.

(2) If the condition specified in table 30-IV is not met, execute the next instruction, unless otherwise specified.

b. Format RI Type 1, Local Jump - This instruction shall jump to the instruction located at memory address Y.

Table 30-IV. Jump Instruction a Value Conditions

a Value	Jump Condition
0	Zero if arithmetic operation, or equal if compare operation. (Condition Code bit 8 equals zero.)
1	Not zero if arithmetic operation, or not equal if compare operation. (Condition Code bit 8 equals one.)
2	Positive if arithmetic operation, or greater than or equal to if compare operation. (Condition Code bit 9 equals zero.)
3	Negative if arithmetic operation, or less than if compare operation. (Condition Code bit 9 equals one.)
4	Overflow
5	Carry
6	Power out of tolerance
7	Bootstrap 2 selected
10	Unconditional Jump
11	Unconditional Stop
12	Stop if Stop Key 1 selected; if not, execute jump.
13	Stop if Stop Key 2 selected; if not, execute jump.
14	Unassigned } but shall perform an unconditional jump
15	Unassigned } but shall perform an unconditional jump
16	Unassigned } but shall perform an unconditional jump
17	Unassigned } but shall perform an unconditional jump

c. Format RK, Conditional Jump - This instruction shall test for the condition specified by the a-value as specified in table 30-IV and perform one of the following:

(1) If the condition specified in table 30-IV is met, jump to the instruction located at the address specified by operand Y. If the a-value specifies a Stop (unconditional) or a Stop Key set operation, the DPS shall halt the RTC, then stop. When the DPS is restarted, load P with operand Y and execute the instruction at P.

(2) If the condition specified in table 30-IV is not met, execute the next instruction, unless otherwise specified.

d. Format RX, Conditional Jump - This instruction shall test for the condition specified by the a-value as specified in table 30-IV and perform one of the following:

(1) If the condition specified in table 30-IV is met, jump to the instruction located at the address specified by the contents of memory address Y. If the a-value specifies a Stop (unconditional) or a Stop Key set operation, the DPS shall halt the RTC, then stop. When the DPS is restarted, load P with the contents of memory address Y and execute the instruction at P.

(2) If the condition specified in table 30-IV is not met, execute the next instruction, unless otherwise specified.

30.35 Operation code 41. - The operation code 41 instructions shall be as follows:

a. Format RR, Index Jump - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) does not equal zero, decrement (R_a) by 1 and jump to the instruction located at the address specified by (R_m).

(2) If (R_a) equals zero, execute the next instruction.

b. Format RI Type 1, Local Jump Indirect - This instruction shall jump unconditionally to the address specified by the contents of memory address Y.

c. Format RK, Index Jump - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) does not equal zero, decrement (R_a) by 1 and jump to the instruction located at the address specified by operand Y.

(2) If (R_a) equals zero, execute the next instruction.

d. Format RX, Index Jump - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) does not equal zero, decrement (R_a) by 1 and jump to the instruction located at the address specified by the contents of memory address Y.

(2) If (R_a) equals zero, execute the next instruction.

30.36 Operation code 42. - The operation code 42 instructions shall be as follows:

a. Format RR, Jump and Link Register - This instruction shall store (P)+1 in R_a, and jump to the instruction located at the address specified by (R_m).

b. Format RI - Unassigned.

c. Format RK, Jump and Link Register - This instruction shall store (P)+2 in R_a, and jump to the instruction located at the address specified by operand Y.

d. Format RX, Jump and Link Register - This instruction shall store (P)+2 in R_a, and jump to the instruction located at the address specified by the contents of address Y.

30.37 Operation code 43. - The operation code 43 instructions shall be as follows:

a. Format RR - Unassigned.

b. Format RI Type 1, Local Jump and Link Memory - This instruction shall store (P)+1 at memory address Y, and jump to the instruction located at memory address Y + 1.

c. Format RK, Jump and Link Memory - This instruction shall store (P)+2 at memory address Y, and jump to the instruction located at memory address Y + 1.

d. Format RX, Jump and Link Memory - This instruction shall store (P)+2 at the address specified by the contents of address Y, and jump to the instruction located at 1 plus the address specified by the contents of address Y.

30.38 Operation code 44. - The operation code 44 instructions shall be as follows:

a. Format RR, Jump Register = 0 - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) equals zero, jump to the instruction located at the address specified by (R_m).

(2) If (R_a) does not equal zero, execute the next instruction.

b. Format RI Type 1, Local Jump Equal - This instruction shall test the Condition Code in the Status Register and perform one of the following:

(1) If bit 8 of the Condition Code is equal to zero, jump to the instruction located at memory address Y.

(2) If bit 8 of the Condition Code is equal to one, execute the next instruction.

c. Format RK, Jump Register = 0 - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) equals zero, jump to the instruction located at the address specified by operand Y.

(2) If the (R_a) does not equal zero, execute the next instruction.

d. Format RX, Jump Register = 0 - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) equals zero, jump to the instruction located at address specified by the contents of memory address Y.

(2) If (R_a) does not equal zero, execute the next instruction.

30.39 Operation code 45. - The operation code 45 instructions shall be as follows:

a. Format RR, Jump Register ≠ 0 - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) does not equal zero, jump to the instruction located at the address specified by (R_m).

(2) If (R_a) equals zero, execute the next instruction.

b. Format RI Type 1, Local Jump Not Equal - This instruction shall test the Condition Code and perform one of the following:

(1) If bit 8 of the Condition Code is equal to one, jump to the instruction located at memory address Y.

(2) If bit 8 of the Condition Code is equal to zero, execute the next instruction.

c. Format RK, Jump Register ≠ 0 - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) does not equal zero, jump to the instruction located at the address specified by operand Y.

(2) If (R_a) equals zero, execute the next instruction.

d. Format RX, Jump Register $\neq$ 0 - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) does not equal zero, jump to the instruction located at the address specified by the contents of memory address Y.

(2) If (R_a) equals zero, execute the next instruction.

30.40 Operation code 46. - The operation code 46 instructions shall be as follows:

a. Format RR, Jump Register Positive - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) is equal to or greater than zero, jump to the instruction located at the address specified by (R_m).

(2) If (R_a) is less than zero, execute the next instruction.

b. Format RI Type 1, Local Jump Greater Than or Equal - This instruction shall test the Condition Code and perform one of the following:

(1) If bit 9 of the Condition Code is equal to zero, jump to the instruction located at memory address Y.

(2) If bit 9 of the Condition Code is equal to one, execute the next instruction.

c. Format RK, Jump Register Positive - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) is equal to or greater than zero, jump to the instruction located at the address specified by operand Y.

(2) If (R_a) is less than zero, execute the next instruction.

d. Format RX, Jump Register Positive - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) is equal to or greater than zero, jump to the instruction located at address specified by the contents of memory address Y.

(2) If (R_a) is less than zero, execute the next instruction.

30.41 Operation code 47. - The operation code 47 instructions shall be as follows:

a. Format RR, Jump Register Negative - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) is less than zero, jump to the instruction located at the address specified by (R_m).

(2) If (R_a) is equal to or greater than zero, execute the next instruction.

b. Format RI Type 1, Local Jump Less Than - This instruction shall test the Condition Code and perform one of the following:

(1) If bit 9 of the Condition Code is equal to one, jump to the instruction located at memory address Y.

(2) If bit 9 of the Condition Code is equal to zero, execute the next instruction.

c. Format RK, Jump Register Negative - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) is less than zero, jump to the instruction located at the address specified by operand Y.

(2) If (R_a) is equal to or greater than zero, execute the next instruction.

d. Format RX, Jump Register Negative - This instruction shall test (R_a) and perform one of the following:

(1) If (R_a) is less than zero, jump to the instruction located at address specified by the contents of memory address Y.

(2) If (R_a) is equal to or greater than zero, execute the next instruction.

30.42 Operation code 50. - The operation code 50 instructions shall be as follows:

a. Format RR, Floating Point Subtract- This instruction shall subtract the floating point number (R_m, R_{m+1}) from the floating point number (R_a, R_{a+1}), store the normalized floating point difference in R_a and R_{a+1}, and then set the Condition Code. If rounding is not specified, R_{a+2} and R_{a+3} shall contain the residue in floating point format.

b. Format RI, Type 2, Floating Point Subtract- This instruction shall subtract the floating point number at memory address Y and Y+1 from the floating point number (R_a, R_{a+1}), store the normalized floating point difference in R_a and R_{a+1}, and then set the Condition Code. If rounding is not specified, R_{a+2} and R_{a+3} shall contain the residue in floating point format.

c. Format RK- Unassigned.

d. Format RX, Floating Point Subtract- Same as RI, Type 2.

30.43 Operation code 51. - The operation code 51 instructions shall be as follows:

a. Format RR, Floating Point Add- This instruction shall add the floating point number (R_m, R_{m+1}) to the floating point number (R_a, R_{a+1}), store the normalized floating point sum in R_a and R_{a+1}, and then set the Condition Code. If rounding is not specified, R_{a+2} and R_{a+3} shall contain the residue in floating point format.

b. Format RI, Type 2, Floating Point Add- This instruction shall add the floating point number at memory addresses Y and Y+1 to the floating point number (R_a, R_{a+1}), store the normalized floating point sum in R_a and R_{a+1}, and then set the Condition Code. If rounding is not specified, R_{a+2} and R_{a+3} shall contain the residue in floating point format.

c. Format RK- Unassigned.

d. Format RX, Floating Point Add- Same as RI, Type 2.

30.44 Operation Code 52. - The operation code 52 instructions shall be as follows:

a. Format RR, Floating Point Multiply- This instruction shall multiply the floating point number (R_m, R_{m+1}) by the floating point number (R_a, R_{a+1}), store the normalized floating point product in R_a and R_{a+1}, and then set the Condition Code. (R_a, R_{a+1}) shall be a floating point number representing the most significant digits of the product. If rounding is not specified, R_{a+2} and R_{a+3} shall be a floating point number representing the least significant portion of the product.

b. Format RI, Type 2, Floating Point Multiply- This instruction shall multiply the floating point number at memory addresses Y and Y+1 by the floating point number (R_a, R_{a+1}), store the normalized floating point product in R_a and R_{a+1}, and then set the Condition Code. (R_a, R_{a+1}) shall be a floating point number representing the most significant digits of the product. If rounding is not specified, R_{a+2} and R_{a+3} shall be a floating point number representing the least significant portion of the product.

c. Format RK- Unassigned

d. Format RX, Floating Point Multiply-Same as RI, Type 2.

30.45 Operation Code 53. - The operation code 53 instructions shall be as follows:

a. Format RR, Floating Point Divide- This instruction shall divide the floating point number (R_a, R_{a+1}) by the floating point number (R_m, R_{m+1}), store the normalized floating point quotient in R_a and R_{a+1} and then set the Condition Code. If rounding is not specified R_{a+2} and R_{a+3} shall be the remainder in floating point format.

b. Format RI, Type 2, Floating Point Divide- This instruction shall divide the floating point number (R_a, R_{a+1}) by the floating point number at memory addresses Y and Y+1, store the normalized floating point quotient in R_a and R_{a+1}, and then set the Condition Code. If rounding is not specified, R_{a+2} and R_{a+3} shall be the remainder in floating point format.

c. Format RK- Unassigned

d. Format RX, Floating Point Divide-Same as RI, Type 2.

30.46 Operation code 54. - The operation code 54 instructions shall be as follows:

a. Format RR, Load Page Register- This instruction shall load the page register specified by (R_a) with (R_m). Only bits 0 through 5 of R_a shall be interpreted.*

b. Format RI Type 2, Load Page Register- This instruction shall load the page register specified by (R_a) with the contents of the memory address specified by (R_m). Only bits 0 through 5 of R_a shall be interpreted.*

c. Format RK - Unassigned.

d. Format RX, Load Page Register Multiple- This instruction shall load the contents of sequential memory addresses beginning at Y, into sequential page registers beginning at the address defined by bits 0 through 5 of R_a, and continuing until the number of registers loaded equals one plus the count defined by bits 8 through 13 of R_a. A count of zero shall cause one page register to be loaded.*

*Bits 6 through 14 of all page registers are all zeros and cannot be altered.

30.47 Operation code 55. - The operation code 55 instructions shall be as follows:

a. Format RR, Store Page Register - This instruction shall store the page register specified by (R_a) in R_m. Only bits 0 through 5 of R_a shall be interpreted.

b. Format RI Type 2, Store Page Register - This instruction shall store the page register specified by (R_a) in the memory address specified by (R_m). Only bits 0 through 5 of R_a shall be interpreted.

c. Format RK - Unassigned.

d. Format RX, Store Page Register Multiple - This instruction shall store sequential page register contents beginning at the address defined by bits 0 through 5 of R_a into sequential memory addresses beginning at Y and continuing until the number of registers stored equals one plus the count defined by bits 8 through 13 of R_a. A count of zero shall cause the contents of one page register to be stored.

30.48 Operation code 56. - The operation code 56 instruction shall be as follows:

a. Format RR, Double Precision Multiply - This instruction shall multiply the double length number (R_m, R_{m+1}) by the double length number (R_a, R_{a+1}) and store the result* in R_a, R_{a+1}, R_{a+2}, R_{a+3}; and set the Condition Code.

b. Format RI, Type 2, Double Precision Multiply - This instruction shall multiply the double length contents of memory address Y and Y+1 by the double length number (R_a, R_{a+1}) and store the result* in R_a, R_{a+1}, R_{a+2}, R_{a+3}; and set the Condition Code.

c. Format RK - Unassigned.

d. Format RX, Double Precision Multiply - This instruction shall multiply the double length contents of memory address Y and Y+1 by the double length number (R_a, R_{a+1}) and store the result* in R_a, R_{a+1}, R_{a+2}, R_{a+3}; and set the Condition Code.

*The sign bit for the 64-bit quantity shall be bit 15 of the most significant 16 bits (R_a).

30.49 Operation code 57. - The operation code 57 instructions shall be as follows:

a. Format RR, Double Precision Divide - This instruction shall divide the number* (R_a, R_{a+1}, R_{a+2}, R_{a+3}) by the double length number (R_m, R_{m+1}), store the double length quotient in R_{a+2} and R_{a+3}, and the double length remainder in R_a and R_{a+1}; and set the Condition Code.

b. Format RI, Type 2, Double Precision Divide - This instruction shall divide the number* (R_a, R_{a+1}, R_{a+2}, R_{a+3}) by the double length contents of memory address Y and Y+1, store the double length quotient in R_{a+2} and R_{a+3}, and the double length remainder in R_a and R_{a+1}; and set the Condition Code.

c. Format RK - Unassigned.

d. Format RX, Double Precision Divide - This instruction shall divide the number* (R_a, R_{a+1}, R_{a+2}, R_{a+3}) by the double length contents of memory address Y and Y+1, store the double length quotient in R_{a+2} and R_{a+3} and the double length remainder in R_a and R_{a+1}; and set the Condition Code.

*The sign bit for the 64-bit quantity shall be bit 15 of the most significant 16 bits (R_a).

30.50 Operation Code 60. - The operation code 60 instructions shall be as follows:

a. Format RL-1, Logical Right Single Shift- This instruction shall shift (R_a) right n places with zeroes extended to fill and set the Condition Code. n shall be the value in bits 0-3 of the instruction m-designator.

b. Format RL-2, Algebraic Right Single Shift - This instruction shall shift (R_a) right n places with sign extended to fill and set the Condition Code. n shall be the value in bits 0-3 of the instruction m-designator.

c. Format RL-3, Logical Right Double Shift - This instruction shall shift the double length (R_a,R_{a+1}) right n places with zeroes extended to fill and set the Condition Code. n shall be the value in bits 0-3 of the instruction m-designator.

d. Format RL-4, Algebraic Right Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) right n places with sign extended to fill and set the Condition Code. n shall be the value in bits 0-3 of the instruction m-designator.

30.51 Operation Code 61. - The operation code 61 instructions shall be as follows:

a. Format RL-1, Algebraic Left Single Shift - This instruction shall shift (R_a) left n places with zeroes extended to fill and set the Condition Code. n shall be the value in bits 0-3 of the instruction m-designator.

b. Format RL-2, Circular Left Single Shift - This instruction shall shift (R_a) left circular n places and set the Condition Code. n shall be the value in bits 0-3 of the instruction m-designator.

c. Format RL-3, Algebraic Left Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) left n places with zeroes extended to fill and set the Condition Code. n shall be the value of bits 0-3 of the instruction m-designator.

d. Format RL-4, Circular Left Double Shift - This instruction shall shift the double length (R_a, R_{a+1}) left circular n places and set the Condition Code. n shall be the value in bits 0-3 of the instruction m-designator.

30.52 Operation Code 62. - The operation code 62 instructions shall be as follows:

a. Format RL-1, Subtract - This instruction shall subtract the 4-bit literal contained in the m-designator of the instruction from (R_a) and store the result in R_a; and set the Condition Code.

b. Format RL-2, Subtract Double - This instruction shall subtract the 4-bit literal contained in the m-designator of the instruction from the double length (R_a,R_{a+1}) and store the result in R_a and R_{a+1}; and set the Condition Code.

c. Format RL-3, Add - This instruction shall add the 4-bit literal contained in the m-designator of the instruction to (R_a) and store the result in R_a; and then set the Condition Code.

d. Format RL-4, Add Double - This instruction shall add the 4-bit literal contained in the m-designator of the instruction to the double length (R_a, R_{a+1}) and store the result in R_a and R_{a+1}; and set the Condition Code.

30.53 Operation code 63. - The operation code 63 instructions shall be as follows:

a. Format RL-1, Load - This instruction shall load the 4-bit literal contained in the m-designator of the instruction into R_a and set the Condition Code.

b. Format RL-2, Compare - This instruction shall arithmetically compare (R_a) to the 4-bit literal contained in the m-designator of the instruction and set the Condition Code.

c. Format RL-3, Multiply - This instruction shall multiply the 4-bit literal contained in the m-designator of the instruction by (R_{a+1}) and store the double length result in R_a, R_{a+1}; and then set the Condition Code.

d. Format RL-4, Divide - This instruction shall divide the double length (R_a, R_{a+1}) by the 4-bit literal contained in the m-designator of the instruction, store the quotient in R_{a+1} and the remainder in R_a; and then set the Condition Code.

30.54 Operation code 64. - The operation code 64 instructions shall be as follows:

a. Format RR - Unassigned

b. Format RI - Unassigned

c. Format RK - Unassigned

d. Format RX - Byte Subtract - This instruction shall subtract the selected byte of the contents of memory address Y from (R_a) and store the result in R_a, and set the Condition Code.

30.55 Operation code 65. - The operation code 65 instructions shall be as follows:

a. Format RR - Unassigned

b. Format RI - Unassigned

c. Format RK - Unassigned

d. Format RX, Byte Add - This instruction shall add the selected byte from the contents of memory address Y to (R_a), store the sum in R_a, and set the Condition Code.

30.56 Operation code 66. - The operation code 66 instructions shall be as follows:

a. Format RR - Unassigned.

b. Format RI - Unassigned.

c. Format RK - Unassigned.

d. Format RX, Byte Compare - This instruction shall arithmetically compare (R_a) to the selected byte of memory address Y, and set the Condition Code.

30.57 Operation code 67. - The operation code 67 instructions shall be as follows:

a. RR, RI, and RK - Reserved for user-defined macroinstructions, otherwise unassigned. When 512 words of microprogrammed ROM for a user-defined microprogram have been specified in the procurement document (see 6.1), operation code 67 formats RR, RI, and RK shall provide access to the ROM. For these instructions, instruction word bits 15 through 10 shall be 67_8, instruction word bits 7 through 0 shall be free form, and instruction word bits 9 and 8 shall be 00 for the RR instruction, 01 for the RI instruction, and 10 for the RK instruction. If more than one word is associated with any of these instructions, the additional words shall be free form.

b. Format RX, Byte Compare and Index by 1 - This instruction shall arithmetically compare (R_a) to the selected byte of memory address Y, set the Condition Code; then increment (R_m) by 1.

30.58 Operation code 70. - The operation code 70 instructions shall be as follows:

a. Format RR, Channel Control (Command) - This instruction shall perform the operation specified by the m-designator as specified in table 30-V on all I/O channels when m = 0 through 7 or the channel specified by the a-designator when m = 10 through 17.

b. Format RR, Channel Control (Chaining) - This instruction shall perform the operation specified by the m-designator as specified in table 30-V. The a-designator shall not be used and the channel referenced shall be associated with the active chain.

c. Format RI - Unassigned.

d. Format RK - Unassigned.

e. Format RX, Initiate Transfer (Chaining) - This instruction shall load the control memory BCW and BAP locations with the contents of memory addresses Y and Y+1 respectively, and enable input or output transfers on the channel corresponding to the chain executing the instruction. Address Y shall be required to be even. Chaining on the channel (input or output) shall be disabled until transfer termination. Transfer termination shall result when the buffer word count decrements to zero. Chaining shall be re-enabled after transfer termination. The a-designator shall be applicable for all I/O transfers and shall be interpreted and specified in table 30-VI.

Table 30-V. Channel Control Instruction m-Designator

m Value	Operation
0	Clear all channels. Deactivate all data buffers, disable all external interrupt data, disable class III interrupts and clear monitor and suppress flags.
1	Unassigned.
2	Unassigned.
3	Unassigned.
4	Enable all channels to accept external interrupt requests. Each channel shall set the external interrupt enable (EIE). Receipt of an external interrupt request shall then store the external interrupt word (one word) at the assigned memory address, and clear the EIE on the channel.
5	Do not accept external interrupt data from any channel. Clear the EIE on all channels.
6	For all channels, generate the class III priority 2 interrupt if any external interrupt was accepted. Enable generation of class III priority 2, 3, and 4 interrupts.
7	For all channels, disable the generation of class III priority 2, 3, and 4 interrupts to the processor.
10	Master clear the associated channel. Clear the EIE, disable class III interrupts, deactivate the data buffers, clear the monitor flag, and clear the suppress flag.
11	Unassigned.
12	Unassigned.
13	Unassigned.
14	Enable the associated channel to accept external interrupt requests. The channel shall set the external interrupt enable line. Receipt of an external interrupt request shall then store the external interrupt word (one word) at the assigned memory address, and clear the EIE.
15	Do not accept external interrupt data from the associated channel. Clear the EIE.
16	For the associated channel, enable the generation of class III priority 2, 3, and 4 interrupts to the processor. Generate a class III priority 2 interrupt if any external interrupt was accepted.
17	For the associated channel, disable the generation at class III priority 2, 3, and 4 interrupts to the processor.

Table 30-VI Initiate Transfer Instruction a-Designator

a-Value	Type of Transfer
0	Input data
1	Output data
2	External function
3	External function with force

30.59 Operation code 71. - The operation code 71 instructions shall be as follows:

a. Format RR - Unassigned.

b. Format RI - Unassigned.

c. Format RK, Initiate Chain (Command) - This instruction shall initiate chaining for the channel specified by the a-designator. The m-designator shall specify the chain; m = 2, input chain; m = 6, output chain. The instruction shall load the corresponding Chain Pointer with the operand Y for use as the starting address for the selected chain.

d. Format RK Load Control Memory, (Chaining) - This instruction shall load the control memory location specified by the m-designator as specified in table 30-VII with Y. The a-designator is not used, and the channel referenced shall be associated with the active chain.

e. Format RX Load Control Memory (Command) - This instruction shall load the control memory location specified by the m-designator as specified in table 30-VII with the contents of the memory address specified by Y. The a-designator shall specify the channel.

f. Format RX Load Control Memory (Chaining) - This instruction shall load the control memory location specified by the m-designator as specified in table 30-VII with the contents of address Y. The a-designator is not used, and the channel referenced shall be associated with the active chain.

Table 30-VII. Load, Store, Control Memory Instructions m-Designator

m-Value	Location
0	TM, O, B and Buffer Word Count (IN)
1	Buffer Address Pointer (IN)
2	Chain Address Pointer (IN)
3	Unassigned
4	TM, O, B and Buffer Word Count (OUT)
5	Buffer Address Pointer (OUT)
6	Chain Address Pointer (OUT)
7	Unassigned
10	Monitor Register (MIL-STD-188, RS-232 Serial, and VACALES)
11	Suppress Register (MIL-STD-188, RS-232 Serial, and VACALES)
12	Serial Mode Information (MIL-STD-188, RS-232 Serial, and VACALES)
13-17	Unassigned

30.60 Operation code 72. - The operation code 72 instructions shall be as follows:

a. Format RR - Unassigned.

b. Format RI - Unassigned.

c. Format RK - Unassigned.

d. Format RX Store Control Memory (Command) - This instruction shall store the contents of the control memory location specified by the m-designator as specified in table 30-VII at memory address Y. The a-designator shall specify the channel.

e. Format RX Store Control Memory (Chaining) - This instruction shall store the contents of the control memory location specified by the m-designator as specified in table 30-VII at memory address Y. The a-designator is not used, and the channel referenced shall be associated with the active chain.

30.61 Operation code 73. - The operation code 73 instructions shall be as follows:

a. Format RR, Halt/Interrupt (Chaining) - This instruction shall perform one of the following as specified by the a-designator; the m-designator is not used:

(1) If a = 0, halt the chaining action.

(2) If a = 1, generate the Chain interrupt.

b. Format RI - Unassigned.

c. Format RK - Unassigned.

d. Format RX, Set/Clear Flag (Chaining) - This instruction shall set or clear the most significant two bits (flag) of the memory location specified by Y as specified by the a-designator; the m-designator is not used.

(1) If a = 1, set flag.

(2) If a = 0, clear flag.

30.62 Operation code 74. - The operation code 74 instructions shall be as follows:

a. Format RR - Unassigned.

b. Format RI - Unassigned.

c. Format RK, Conditional Jump (Chaining) - This instruction shall cause a jump to the address specified by the operand Y if the condition specified by table 30-VIII is met. If none of the conditions are met, the DPS shall execute the next instruction.

d. Format RX - Unassigned.

Table 30-VIII. Conditional Jump Instruction a-designator

a-designator	Jump Operation
0	Unconditional Jump
1	Jump if Suppress Flag Not Set
2	Jump if Monitor Flag Set

30.63 Operation code 75. - The operation code 75 instructions shall be as follows:

a. Format RR, Set Suppress, Set Monitor, and Search for Sync (Chaining) - The function performed when each bit of the m-designator is set shall be as specified herein. m-designator values specified in table 30-IX, used with the interface specified, shall perform the functions specified. Other m-designator values shall be unassigned.

(1) Bit 0, Set Synchronous Serial Active - When bit 0 is set in the m-designator, the synchronous serial channel shall be set active, and chaining shall be enabled for the synchronous serial interface. When bit 0 is cleared in the m-designator, synchronization shall be disabled.

(2) Bit 1, Set Suppress - When bit 1 is set in the m-designator and the interface is either the synchronous or asynchronous serial interface, input data matching the character loaded into the suppress register shall not be transferred to memory. The suppress flag shall be set. The suppress flag shall then be cleared upon input of the next serial character. When bit 1 is cleared in the m-designator, Set Suppress shall be disabled.

(3) Bit 2, Set Monitor - When bit 2 is set in the m-designator and the interface is either the MIL-STD-188 or RS-232 synchronous or asynchronous serial interface, the last n bits of input data shall be compared to the monitor register. When a match occurs, the monitor flag shall be set and that character shall be input as data. The buffer shall be terminated and the chain shall be enabled. The monitor flag shall then be cleared upon input of the next serial character. When bit 1 is cleared in the m-designator, Set Suppress shall be disabled. In the VACALES mode interface, Set Monitor is enabled when bit 2 is set and bit 3 is cleared.

(4) Bit 3, Search for Sync - When bit 3 is set in the m-designator and the MIL-STD-188 or RS-232 synchronous serial interface is set active (bit 0 is set), at each bit time the input channel shall compare the value of the last n bits of input data to the contents of the suppress register (when n is equal to the character length of the channel). When a match occurs, the next character of n bits shall be compared to the suppress register. If the next character compare results in a match, the suppress designator shall be set and the next instruction of the chain shall be enabled. If the next character compare does not result in a match, the suppress designator shall not be set and the next instruction of the chain shall be enabled. When bit 3 is cleared in the m-designator, Search for Sync shall be disabled.

(5) Search for Sync, Bit-by-Bit - When bit 3 is set and bit 2 is cleared in the m-designator and the interface is the VACALES mode interface, at each bit time the input channel shall compare the value of the last n bits of input data to the contents of the suppress register (where n is equal to the character length of the channel). When a match occurs the interface shall start assembling the next input data bits into a n bit character and shall enable the next chain instruction. When bit 3 is cleared in the m-designator, Search for Sync shall be disabled.

(6) Search for Sync, Character - When bits 2 and 3 are set in the m-designator and the interface is the VACALES mode interface, the channel shall assemble input data bits into n bit characters and compare each character to the contents of the suppress register (where n is equal to the character length of the channel). If the compare results in a match, the suppress flag shall be set and the next chain instruction shall be enabled. If the compare does not result in a match, the suppress flag shall not be set and the next chain instruction shall be enabled.

b. Format RI - Unassigned.

c. Format RK - Unassigned.

d. Format RX - Unassigned.

Table 30-IX. Set Suppress, Set Monitor, and Search for Sync m-designator

* m 3210	Interface	Function
0000	Sync	Disable search for sync, disable set monitor, disable set suppress, and enable next chain instruction.
0000	Async	Disable set monitor, disable set suppress, and enable next chain instruction.
0001	Sync	Hold sync active, enable next chain instruction.
0010	Async	Enable set suppress, enable next chain instruction.
0011	Sync	Enable set suppress, enable next chain instruction.
0100	Async	Enable set monitor, enable next chain instruction.
0101	Sync	Enable set monitor, enable next chain instruction.
0110	Async	Enable set monitor, enable set suppress, enable next chain instruction.
0111	Sync	Enable set monitor, enable set suppress, enable next chain instruction.
1001	Sync MIL-STD-188 RS-232	Enable search for sync, disable chaining until function is complete.
1001	VACALES	Enable search for synch bit-by-bit, disable chaining until function is complete.
1011	Sync MIL-STD-188 RS-232	Enable search for sync, enable set suppress, disable chaining until function is complete.
1011	VACALES	Enable search for sync bit-by-bit, enable set suppress, disable chaining until function is complete.

Table 30-IX. Set Suppress, Set Monitor, and Search, for Sync m-designator (continued)

* m (3210)	Interface	Function
1101	VACALES	Enable search for sync character compare, set suppress flag if character compares, enable next chain instruction after each character.
1111	VACALES	Enable search for sync character compare, set suppress flag if compares, enable set suppress, enable next chain instruction after each character.

* m-designator values of 1000_2, 1010_2, and 1110_2 have unassigned functions and are not recommended for use.

30.64 Operation code 76. - The operation code 76 instruction shall be as follows:

a. Format RR, Set - Clear Discretes (Command) - This instruction shall set or clear the discretes associated with the MIL-STD-188,VACALES, or RS-232 serial interface as specified by the m-designator. The a-designator shall specify the channel. Discrete set-clear functions in the MIL-STD-188 or VACALES interface are specified in table 30-X, with the corresponding interface control line. Discrete set-clear functions in the RS-232 interface are specified in table 30-XI.

Table 30-X MIL-STD-188 and VACALES Discrete Set-Clear Functions

m-designator	Function	Discrete	MIL-STD-188 Line Designator	VACALES Line Designator
1111	Set	Outbound Control Line 1	A (ON)	Loop Back (Modem)
1110	Clear	Outbound Control Line 1	A (OFF)	Loop Back (Modem)
1101	Set	Outbound Control Line 2	D (ON)	D1
1100	Clear	Outbound Control Line 2	D (OFF)	D1
1011	Set	Outbound Control Line 3	F (ON)	F1
1010	Clear	Outbound Control Line 3	F (OFF)	F1
1001	Set	Outbound Control Line 4	G (ON)	G1
1000	Clear	Outbound Control Line 4	G (OFF)	G1
0111	Set	Outbound Control Line 5	H (ON)	Transmitter Prep
0110	Clear	Outbound Control Line 5	H (OFF)	Transmitter Prep
0101	Set	Outbound Control Line 6	J (ON)	J1
0100	Clear	Outbound Control Line 6	J (OFF)	J1
0011	Set	Not used	Not used	Not used
0010	Clear	Not used	Not used	Not used
0001	Clear	Internal Loop Test	N/A	N/A
0000	Set	Internal Loop Test	N/A	N/A

Table 30-XI. RS-232 Discrete Set-Clear Functions

m-designator	Function	Discrete
1111	Set (ON)	Loop Test
1110	Clear(OFF)	Loop Test
1101	Set (ON)	Data Terminal Ready
1100	Clear(OFF)	Data Terminal Ready
1011	Set (ON)	New Sync
1010	Clear(OFF)	New Sync
1001	Set (ON)	Request to Send
1000	Clear(OFF)	Request to Send
0111	Set	Enable Ring Indicator*
0110	Clear	Enable Ring Indicator*
0101		Spare
0100		Spare
0011		Spare
0010		Spare
0001	Clear	Internal Loop Test*
0000	Set	Internal Loop Test*

*Internal functions which have no corresponding control line.

b. Format RR, Set - Clear Discretes (Chaining) - This instruction shall set or clear the discretes associated with the MIL-STD-188, VACALES, or RS-232 serial interface as specified by the m-designator. The a-designator is not used. Discrete set-clear functions are specified in table 30-X and table 30-XI.

c. Format RI - Unassigned.

d. Format RK - Unassigned.

e. Format RX, Store Status (Command) - This instruction shall store the channel status at the memory location specified by operand Y. The data shall be interpreted as specified in table 30-XII for the MIL-STD-188 serial interface and as specified in table 30-XIII for the RS-232 serial interface. The a-designator shall specify the channel.

f. Format RX, Store Status (Chaining) - This instruction shall store the channel status at the memory location specified by operand Y. The data shall be interpreted as shown in table 30-XII for the MIL-STD-188 serial interface, as specified in table 30-XII for the **RS-232 serial** interface, and as specified in table XIV for the VACALES mode interface.

30.65 Operation code 77. - Unassigned.

Table 30-XII. Store Status Bit Interpretation, MIL-STD-188 Serial Interface.

Bit	Function	Description
0	Break	The serial I/O did not detect a stop bit. Used in asynchronous mode only.
1	Overrun	The serial I/O did not transfer a data word to memory before another I/O word was received.
2	Parity Error	The serial I/O detected a parity error on an input data word.
3	E Active	Control Line E was set active by an external device.

Table 30-XIII. Store Status Bit Interpretation, RS-232 Serial Interface.

Bit	Function	Description
0	Break	The serial I/O did not detect a stop bit. Used in asynchronous mode only.
1	Overrun	The serial I/O did not transfer to memory before another I/O word was received.
2	Parity error	The serial I/O detected a parity error on an input data word.
3	Clear to Send	Clear to Send was set active by an external device.

Table 30-XIV. Store Status Bit Interpretation, VACALES Mode Serial Interface

Bit	Function	Description
1	Overrun	The serial I/O did not transfer to memory before another I/O word was received.
2	Parity Error	The serial I/O detected a parity error on an input data word.
3	Sync Error	The inbound discrete control line, Sync Error, was set by an external device.

APPENDIX 40

Condition Code, Carry, and Over-Underflow Designators

40.1 Scope. - This Appendix shall establish which of the conditions apply for instructions which set the Condition Code bits, the carry designator, and the overflow-underflow designator. The applicability shall be as specified in table 40-I.

In table 40-I:

NC shall mean: no change in the designator

0 shall mean: end result is 0.

X shall mean: contingent upon the designator function for that instruction

NA shall mean: not applicable

Table 40-I. Condition Designators

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
00 RR Diagnostic jump when DIAGNOSTIC JUMP switch is selected, otherwise unassigned	NC	NC	NC	NC	
RI Unassigned	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
RX Byte Load	0	0	0	X	(R_a)
01 RR Load	0	0	X	X	(R_a)
RI Load	0	0	X	X	(R_a)
RK Load	0	0	X	X	(R_a)
RX Load	0	0	X	X	(R_a)
02 RR Unary-Arithmetic					
m = 0 Make Positive	X	X	X	X	(R_a)
m = 1 Make Negative	X	0	X	X	(R_a)
m = 2 Round R_a	X	X	X	X	(R_a)
m = 3 Unassigned	NC	NC	NC	NC	
m = 4 Twos Complement Single	X	X	X	X	(R_a)
m = 5 Twos Complement Double	X	X	X	X	(R_a, R_{a+1})
m = 6 Ones Complement Single	0	0	X	X	(R_a)
m = 7 Unassigned	NC	NC	NC	NC	
m = 10 Increment (R_a) by 1	X	X	X	X	(R_a)
m = 11 Decrement (R_a) by 1	X	X	X	X	(R_a)
m = 12 Increment (R_a) by 2	X	X	X	X	(R_a)
m = 13 Decrement (R_a) by 2	X	X	X	X	(R_a)
m = 14 - 17 Unassigned	NC	NC	NC	NC	

Table 40-I. Condition Designators (Continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
RI-2 Load	0	0	X	X	(R_a, R_{a+1})
RK Unassigned	NC	NC	NC	NC	
RX Load Double	0	0	X	X	(R_a, R_{a+1})
03 RR Unary-Control					
m = 0-3	0	0	X	X	(R_a)
m = 4-14,16, & 17	NC	NC	NC	NC	
m = 15	0	0	X	X	(R_{a+1})
RI Unassigned	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
Rx Load Multiple	NC	NC	NC	NC	
04 RR Unary-Shift					
m = 0 Square Root	0	X	X	X	(R_a)
m = 1 Reverse Register	0	0	X	X	(R_a)
m = 2 Count Ones	NC	NC	NC	NC	
m = 3 Scale Factor	NC	NC	NC	NC	
m = 4 - 17	NC	NC	NC	NC	
RI Unassigned	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
RX Byte Load & Index by 1	0	0	0	X	(R_a)
05 RR Set Bit	0	0	X	X	(R_a)
RI Load & Index by 1	0	0	X	X	(R_a)
RK Unassigned	NC	NC	NC	NC	
RX Load and Index by 1	0	0	X	X	(R_a)

Table 40-I. Condition Designators (Continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
06 RR Clear Bit	0	0	X	X	(R_a)
RI Load Double & Index by 2	0	0	X	X	(R_a, R_a+1)
RK Unassigned	NC	NC	NC	NC	
RX Load Double & Index by 2	0	0	X	X	(R_a, R_a+1)
07 RR Compare Bit	0	0	X	X	(R_a)
* RI Load PSW	NA	NA	NA	NA	
RK Unassigned	NC	NC	NC	NC	
* RX Load PSW	NA	NA	NA	NA	
10 RR Logical Right Single Shift	0	0	X	X	(R_a)
RI Unassigned	NC	NC	NC	NC	
RK Logical Right Single Shift	0	0	X	X	(R_a)
RX Byte Store	NC	NC	NC	NC	
11 RR Algebraic Right Single Shift	0	0	X	X	(R_a)
RI Store	NC	NC	NC	NC	
RK Algebraic Right Single Shift	0	0	X	X	(R_a)
RX Store	NC	NC	NC	NC	
12 RR Logical Right Double Shift	0	0	X	X	(R_a, R_a+1)
RI Store Double	NC	NC	NC	NC	
RK Logical Right Double Shift	0	0	X	X	(R_a, R_a+1)
RX Store Double	NC	NC	NC	NC	

* The status register 1 contents shall be under program control.

Table 40-I. Condition Designators (Continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
13 RR Algebraic Right Double Shift	0	0	X	X	(R_a, R_a+1)
RI Unassigned	NC	NC	NC	NC	
RK Algebraic Right Double Shift	0	0	X	X	(R_a, R_a+1)
RX Store Multiple	NC	NC	NC	NC	
14 RR Algebraic Left Single Shift	0	X	X	X	(R_a)
RI Unassigned	NC	NC	NC	NC	
RK Algebraic Left Single Shift	0	X	X	X	(R_a)
RX Byte Store & Index by 1	NC	NC	NC	NC	
15 RR Circular Left Single Shift	0	0	X	X	(R_a)
RI Store & Index by 1	NC	NC	NC	NC	
RK Circular Left Single Shift	0	0	X	X	(R_a)
RX Store & Index by 1	NC	NC	NC	NC	
16 RR Algebraic Left Double Shift	0	X	X	X	(R_a, R_a+1)
RI Store Double & Index by 2	NC	NC	NC	NC	
RK Algebraic Left Double Shift	0	X	X	X	(R_a, R_a+1)
RX Store Double & Index by 2	NC	NC	NC	NC	

Table 40-I. Condition Designators (Continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
17 RR Circular Left Double Shift	0	0	X	X	(R_a, R_a+1)
RI Store Zeros	NC	NC	NC	NC	
RK Circular Left Double Shift	0	0	X	X	(R_a, R_a+1)
RX Store Zeros	NC	NC	NC	NC	
20 RR Subtract	X	X	X	X	(R_a)
RI Subtract	X	X	X	X	(R_a)
RK Subtract	X	X	X	X	(R_a)
RX Subtract	X	X	X	X	(R_a)
21 RR Subtract Double	X	X	X	X	(R_a, R_a+1)
RI Subtract Double	X	X	X	X	(R_a, R_a+1)
RK Unassigned	NC	NC	NC	NC	
RX Subtract Double	X	X	X	X	(R_a, R_a+1)
22 RR Add	X	X	X	X	(R_a)
RI Add	X	X	X	X	(R_a)
RK Add	X	X	X	X	(R_a)
RX Add	X	X	X	X	(R_a)
23 RR Add Double	X	X	X	X	(R_a, R_a+1)
RI Add Double	X	X	X	X	(R_a, R_a+1)
RK Unassigned	NC	NC	NC	NC	
RX Add Double	X	X	X	X	(R_a, R_a+1)

Table 40-I. Condition Designators (Continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
24 RR Compare	X	X	X	X	Result
RI Compare	X	X	X	X	Result
RK Compare	X	X	X	X	Result
RX Compare	X	X	X	X	Result
25 RR Compare Double	X	X	X	X	Result
RI Compare Double	X	X	X	X	Result
RK Unassigned	NC	NC	NC	NC	
RX Compare Double	X	X	X	X	Result
26 RR Multiply	0	0	X	X	(R_a, R_a+1)
RI Multiply	0	0	X	X	(R_a, R_a+1)
RK Multiply	0	0	X	X	(R_a, R_a+1)
RX Multiply	0	0	X	X	(R_a, R_a+1)
27 RR Divide	0	X	X	X	(R_a+1)
RI Divide	0	X	X	X	(R_a+1)
RK Divide	0	X	X	X	(R_a+1)
RX Divide	0	X	X	X	(R_a+1)
30 RR AND	0	0	X	X	(R_a)
RI AND	0	0	X	X	(R_a)
RK AND	0	0	X	X	(R_a)
RX AND	0	0	X	X	(R_a)

Table 40-I. Condition Designators (Continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
31 RR OR	0	0	X	X	(R_a)
RI OR	0	0	X	X	(R_a)
RK OR	0	0	X	X	(R_a)
RX OR	0	0	X	X	(R_a)
32 RR Exclusive OR	0	0	X	X	(R_a)
RI Exclusive OR	0	0	X	X	(R_a)
RK Exclusive OR	0	0	X	X	(R_a)
RX Exclusive OR	0	0	X	X	(R_a)
33 RR Masked Substitute	0	0	X	X	(R_a)
RI Masked Substitute	0	0	X	X	(R_a)
RK Masked Substitute	0	0	X	X	(R_a)
RX Masked Substitute	0	0	X	X	(R_a)
34 RR Compare Masked	0	0	X	X	Result
RI Compare Masked	0	0	X	X	Result
RK Compare Masked	0	0	X	X	Result
RX Compare Masked	0	0	X	X	Result
35 RR I/O Command	NC	NC	NC	NC	
RI Biased Fetch	0	0	X	X	(Y)
RK Execute Remote	NC	NC	NC	NC	
RX Biased Fetch	0	0	X	X	(Y)
36 RR Unassigned	NC	NC	NC	NC	
RI Unassigned	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
RX Unassigned	NC	NC	NC	NC	

Table 40-I. Condition Designators (Continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
37 Trigonometric - Hyperbolic	NC	NC	NC	NC	
40 - 47	NC	NC	NC	NC	
50 RR Floating Point Subtract	0	X	X	X	(R_a,R_{a+1})
RI-2 Floating Point Subtract	0	X	X	X	(R_a,R_{a+1})
RK Unassigned	NC	NC	NC	NC	
RX Floating Point Subtract	0	X	X	X	(R_a,R_{a+1})
51 RR Floating Point Add	0	X	X	X	(R_a,R_{a+1})
RI-2 Floating Point Add	0	X	X	X	(R_a,R_{a+1})
RK Unassigned	NC	NC	NC	NC	
RX Floating Point Add	0	X	X	X	(R_a,R_{a+1})
52 RR Floating Point Multiply	0	X	X	X	(R_a,R_{a+1})
RI-2 Floating Point Multiply	0	X	X	X	(R_a,R_{a+1})
RK Unassigned	NC	NC	NC	NC	
RX Floating Point Multiply	0	X	X	X	(R_a,R_{a+1})
53 RR Floating Point Divide	0	X	X	X	(R_a,R_{a+1})
RI-2 Floating Point Divide	0	X	X	X	(R_a,R_{a+1})
RK Unassigned	NC	NC	NC	NC	
RX Floating Point Divide	0	X	X	X	(R_a,R_{a+1})

Table 40-I. Condition Designators (Continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
54 RR Load Page Register	NC	NC	NC	NC	
RI Load Page Register	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
RX Load Page Register Multiple	NC	NC	NC	NC	
55 RR Store Page Register	NC	NC	NC	NC	
RI Store Page Register	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
RX Store Page Register Multiple	NC	NC	NC	NC	

Table 40-I. Condition Designators (continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE		REGISTER
			(9)	(8)	
56 RR Double Precision Multiply	0	0	X	X	(R_a,R_{a+1}, R_{a+2}, R_{a+3})
RI-2 Double Precision Multiply	0	0	X	X	(R_a,R_{a+1}, R_{a+2}, R_{a+3})
RK Unassigned	NC	NC	NC	NC	
RX Double Precision Multiply	0	0	X	X	(R_a,R_{a+1}, R_{a+2}, R_{a+3})
57 RR Double Precision Divide	0	X	X	X	(R_{a+2},R_{a+3})
RI-2 Double Precision Divide	0	X	X	X	(R_{a+2},R_{a+3})
RK Unassigned	NC	NC	NC	NC	
RX Double Precision Divide	0	X	X	X	(R_{a+2},R_{a+3})
60 RL-1 Logical Right Single-Shift	0	0	X	X	(R_a)
RL-2 Algebraic Right Single-Shift	0	0	X	X	(R_a)
RL-3 Logical Right Double-Shift	0	0	X	X	(R_a,R_{a+1})
RL-4 Algebraic Right Double Shift	0	0	X	X	(R_a,R_{a+1})

Table 40-I Condition Designators (continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
61 RL-1 Algebraic Left Single Shift	0	X	X	X	(R_a)
RL-2 Circular Left Single Shift	0	0	X	X	(R_a)
RL-3 Algebraic Left Double Shift	0	X	X	X	(R_a, R_{a+1})
RL-4 Circular Left Double Shift	0	0	X	X	(R_a, R_{a+1})
62 RL-1 Subtract	X	X	X	X	(R_a)
RL-2 Subtract Double	X	X	X	X	(R_a, R_{a+1})
RL-3 Add	X	X	X	X	(R_a)
RL-4 Add Double	X	X	X	X	(R_a, R_{a+1})
63 RL-1 Load	0	0	0	X	(R_a)
RL-2 Compare	X	X	X	X	Result
RL-3 Multiply	0	0	X	X	(R_a, R_{a+1})
RL-4 Divide	0	X	X	X	(R_{a+1})

Table 40-I Condition Designators (continued)

INSTRUCTION	CARRY DESIGNATOR	OVERFLOW DESIGNATOR	CONDITION CODE (9)	CONDITION CODE (8)	REGISTER
64 RR Unassigned	NC	NC	NC	NC	
RI Unassigned	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
RX Byte Subtract	X	X	X	X	(R_a)
65 RR Unassigned	NC	NC	NC	NC	
RI Unassigned	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
RX Byte Add	X	X	X	X	(R_a)
66 RR Unassigned	NC	NC	NC	NC	
RI Unassigned	NC	NC	NC	NC	
RK Unassigned	NC	NC	NC	NC	
RX Byte Compare	X	X	X	X	Result
67 RR Reserved	As user-programmed	As user-programmed	As user-programmed		
RI Reserved	As user-programmed	As user-programmed	As user-programmed		
RK Reserved	As user-programmed	As user-programmed	As user-programmed		
RX Byte Compare and Index by 1	X	X	X	X	Result
70 -77	NA	NA	NA	NA	

APPENDIX 50

Trigonometric and Hyperbolic Function

50.1 Scope. - This appendix defines the operation of the trigonometric and hyperbolic functions.

50.2 Instruction format. - The option shall provide for trigonometric vector and rotate and hyperbolic vector and rotate functions in the central processor (CP). The functions shall be implemented using the 37 RR instruction. The RR instruction format shall be as specified in this specification. The a and m-designators shall be interpreted as follows:

a. The a-designator shall select 3 general registers designated R_a, R_{a+1}, and R_{a+2}. Before execution of the instruction, these registers shall contain the input parameters. Register R_a shall be referred to herein as Y; the value in Y shall be the value of the y-coordinate. Register R_{a+1} shall be referred to herein as X; the value in X shall be the value of the x-coordinate. Register R_{a+2} shall be referred to herein as W; when required for the function, the value in W shall be the angle (θ for the trigonometric functions and v for the hyperbolic functions). After execution of the instruction, the registers shall contain the results as specified for each instruction. The radix point for registers R_a and R_{a+1} shall be assumed to be between bit 15 and bit 14. The radix point for W = constant in hyperbolic mode shall be assumed to be between bit 14 and bit 13.

b. The m-designator shall specify the trigonometric or hyperbolic function as specified in 50.3.

50.3 Data format. -

50.3.1 Trigonometric functions. -

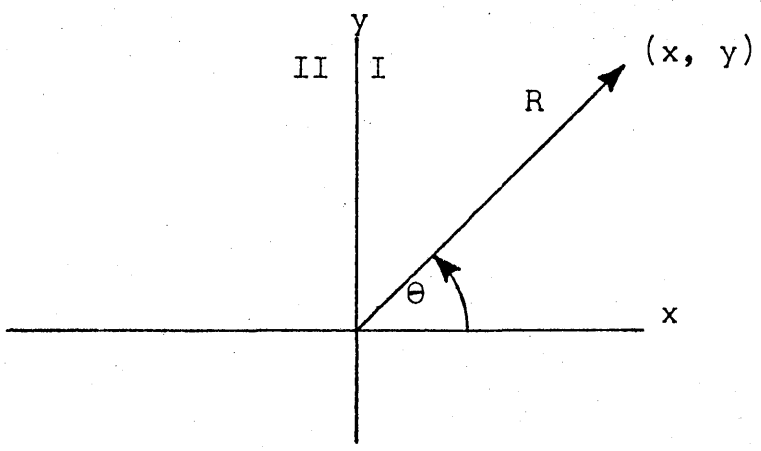
50.3.1.1 Coordinates x and y. - Coordinates x and y shall be specified by 16-bit values. Coordinates shall be required to be scaled to prevent erroneous results. The most significant bit (bit 15) shall specify the sign of the coordinate; 0 shall mean positive and 1 shall mean negative. Negative quantities shall be in 2's complement notation.

50.3.1.2 Angle θ. - The angle θ shall be represented in Binary Angular Measurement (BAMS) which shall be a 16-bit value. Bit 15 shall represent 180°. Bit 14 shall represent 90°. Each less-significant bit shall represent an angle ½ the value of the previous bit position. (The value of the LSB shall be $90^\circ/2^{14}$ or 0.00549° nominal.) The magnitude of the angle shall be the sum of the values corresponding to the bits set. An alternative method of interpreting BAMS is to consider bit 15 as a sign bit; 0 meaning positive and 1 being negative (see figures 50-1 and 50-2). Negative angles shall be in 2's complement representation and the magnitude shall be the sum of the bits cleared.

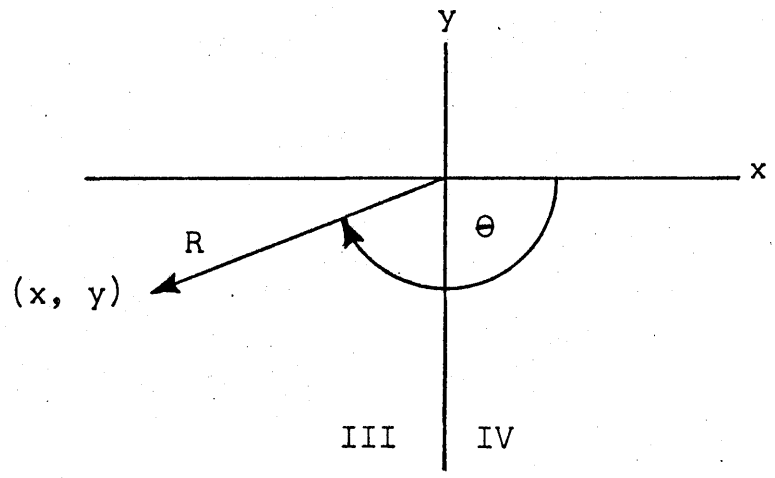
50.3.2 Hyperbolic functions. -

50.3.2.1 Coordinates x and y. - Coordinates x and y shall be specified by 16-bit values. Coordinates shall be required to be scaled to prevent erroneous results. The most significant bit (bit 15) shall specify the sign of the coordinate; 0 shall mean positive and 1 shall mean negative. Negative quantities shall be in 2's complement notation. The absolute value of y/x (y/x) for the hyperbolic vector function shall be required to be ≤ 1.

50.3.2.2 Hyperbolic angle v. - The hyperbolic angle v shall be specified by a 16 bit value. Bit 15 shall specify the sign of the angle; 0 shall mean positive and 1 shall mean negative (see figures 50-3 and 50-4). Negative quantities shall be in 2's complement notation. Values of v shall be required to be scaled such that e^v will not overflow a 15-bit register to avoid erroneous results.

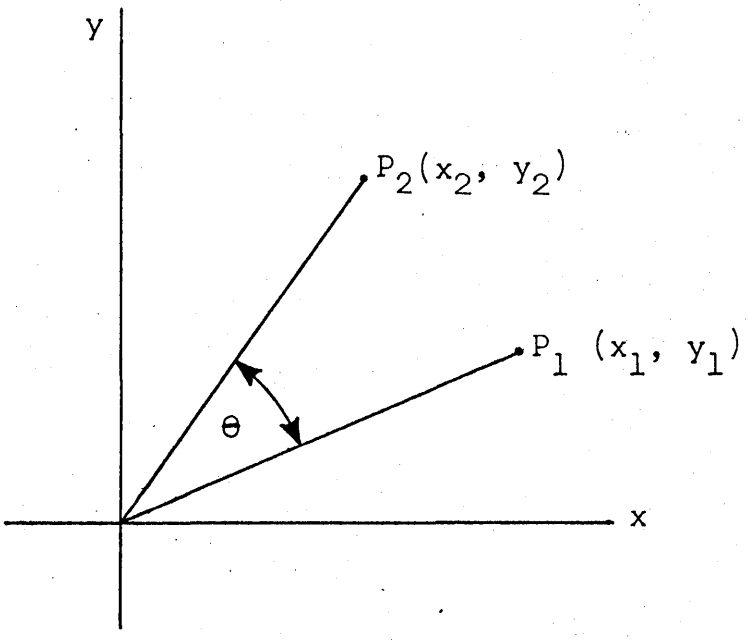


For x and y in either quadrant I or II, θ shall be positive (bit $2^{15} = 0$).
The angle 0^{o} (x positive, y=0) shall be positive (bit $2^{15} = 0$).
The maximum value for coordinates x and y is $0.40000_{8} = 0.50000_{10}$.



For x and y in either quadrant III or IV, θ shall be negative (bit $2^{15} = 1$).
The angle 180^{o} (x negative, y=0) shall be negative (bit $2^{15} = 1$).

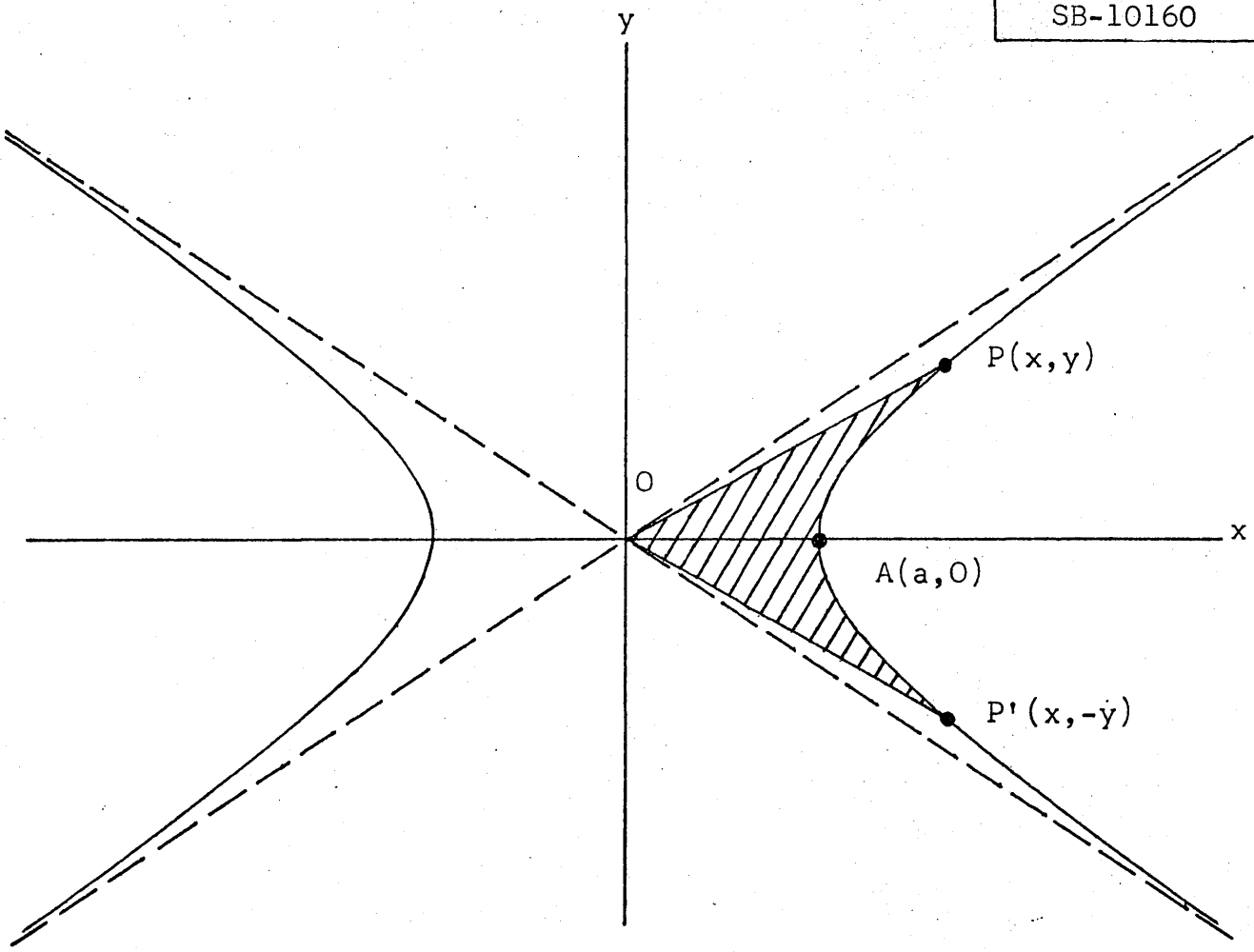
Figure 50-1. Trigonometric Vector Parameters



For a trigonometric rotate with θ positive and x and y initially equal to x_1 and y_1 (P_1), the result shall define a point P_2 counterclockwise rotation).

For a trigonometric rotate with θ negative and x and y initially equal to x_2 and y_2 (P_2), the result shall define a point P_1 clockwise rotation).

Figure 50-2. Trigonometric Rotate Parameters

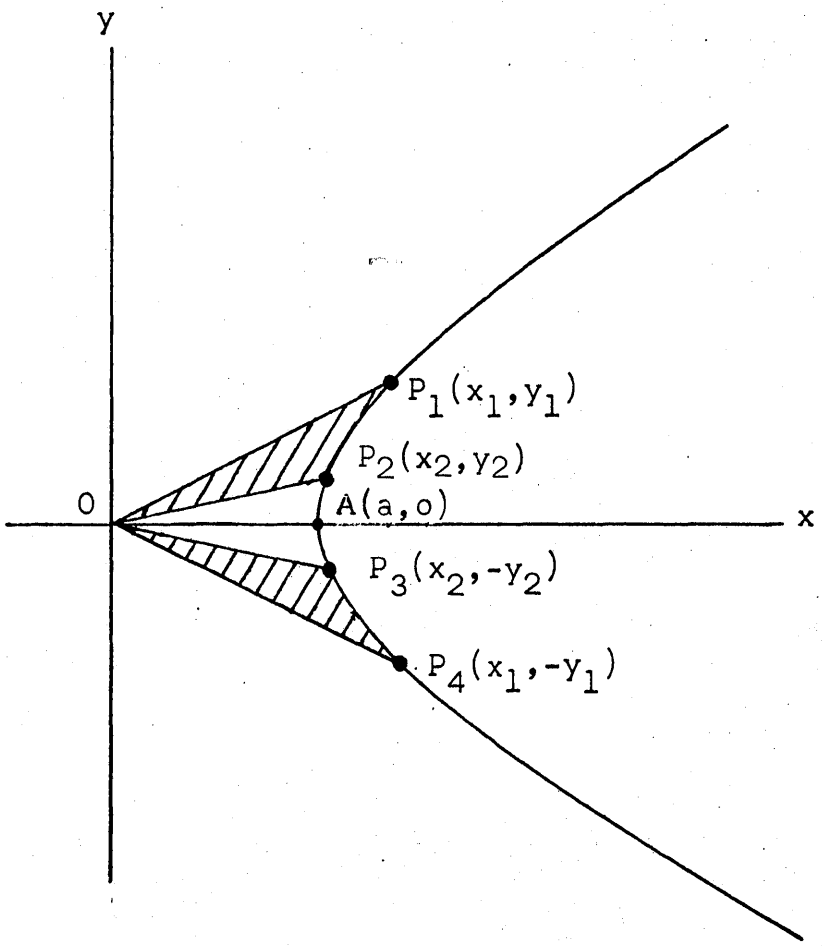


The hyperbolic angle v is equal to the shaded area (OPAP') divided by a^2. For a hyperbolic vector function, v shall be positive if $y>0$ and v shall be negative if $y\leq 0$. The following identity defines v in terms of the input coordinates x and y:

$$v = \tanh^{-1}(y/x) = \tfrac{1}{2} \ln \frac{x + y}{x - y}$$

The maximum value for positive hyperbolic coordinates x and y is $0.20000_8 = 0.25000_{10}$.

Figure 50-3 Hyperbolic Vector Parameters



For a hyperbolic rotate with v negative and x and y initially equal to x_1 and y_1 (P_1), the result shall define a point P_2.

For a hyperbolic rotate with v negative and x and y initially equal to x_2 and $-y_2$ (P_3), the result shall define a point P_4.

For a hyperbolic rotate with v positive and x and y initially equal to x_2 and y_2 (P_2), the result shall define a point P_1.

For a hyperbolic rotate with v positive and x and y initially equal to x_1 and $-y_1$ (P_4), the result shall define a point P_3.

The hyperbolic angle v is equal to the sum of the shaded areas ($OP_1P_2 + OP_3P_4$) divided by a^2.

Figure 50-4. Hyperbolic Rotate Parameters

50.4 Instruction operations. -

50.4.1 Trigonometric vector without prescale. - The 37 RR instruction with m=0 shall perform a trigonometric vector function without prescale. When the input parameters y, x, and O, are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-1):

a. R_a (Y) = 0

b. R_{a+1} (X) = $(\sqrt{x^2 + y^2})/K^* = R/K$

c. R_{a+2} (W) = $\tan^{-1}$ (y/x) = θ

*K = 0.46672_8

50.4.2 Trigonometric rotate without prescale. - The 37 RR instruction with m=1 shall perform a trigonometric rotate function without prescale. When the input parameters y, x, and θ, are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-2):

a. $R_a(Y) = (y \cos \theta + x \sin \theta)/K^*$

b. $R_{a+1}(X) = (x \cos \theta - y \sin \theta)/K$

c. $R_{a+2}(W) = 0$

When the input parameters 0, R, and θ, are contained in registers R_a, R_{a+1}, R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-2).

a. $R_a(Y) = (R \sin \theta)/K$

b. $R_{a+1}(X) = (R \cos \theta)/K$

c. $R_{a+2}(W) = 0$

50.4.3 Trigonometric vector with prescale. - The 37 RR instruction with m=2 shall perform a trigonometric vector function with prescale. When the input parameters y, x, and 0 are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-1):

a. $R_a(Y) = 0$

b. $R_{a+1}(X) = \sqrt{x^2 + y^2} = R$

c. $R_{a+2}(W) = \tan^{-1}(y/x) = \theta$

50.4.4 Trigonometric rotate with prescale. - The 37 RR instruction with m=3 shall perform a trigonometric rotate function with prescale. When the input parameters y, x, and θ, are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-2):

a. $R_a(Y) = y \cos \theta + x \sin \theta$

b. $R_{a+1}(X) = x \cos \theta - y \sin \theta$

c. $R_{a+2}(W) = 0$

When the input parameters 0, R, and θ, are contained in registers R_a, R_{a+1}, R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-2):

a. $R_a(Y) = R \sin \theta$

b. $R_{a+1}(X) = R \cos \theta$

c. $R_{a+2}(W) = 0$

*$K = 0.46672_8$

50.4.5 Hyperbolic vector without postscale. - The 37 RR instruction with m=4 shall perform a hyperbolic vector function without postscale. When the input parameters y, x, and 0, are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-3):

a. $R_a(Y) = 0$

b. $R_{a+1}(X) = (\sqrt{x^2 - y^2})/K_1$*

c. $R_{a+2}(W) = \tanh^{-1}(y/x) = v$

50.4.6 Hyperbolic rotate without postscale. - The 37 RR instruction with m=5 shall perform a hyperbolic rotate function without postscale. When the input parameters y, x, and v, are contained in registers R_a, R_{a+1}, R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-4):

a. For v negative, $R_a(Y) = (-x \sinh |v| + y \cosh v)/K_1$
$R_{a+1}(X) = (x \cosh v - y \sinh |v|)/K_1$

b. For v positive, $R_a(Y) = (x \sinh v + y \cosh v)/K_1$
$R_{a+1}(X) = (x \cosh v + y \sinh v)/K_1$

c. $R_{a+2}(W) = 0$

50.4.7 Hyperbolic vector with postscale. - The 37 RR instruction with m=6 shall perform a hyperbolic vector function with postscale. When the input parameters y, x, and 0, are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows (see figure 50-3):

a. $R_a(Y) = 0$

b. $R_{a+1}(X) = \sqrt{x^2 - y^2}$

c. $R_{a+2}(W) = \tanh^{-1}(y/x) = v$ (see 50.5.1)

50.4.8 Hyperbolic rotate with postscale. - The 37 RR instruction with m=7 shall perform a hyperbolic rotate function with postscale. When the input parameters y, z, and v, are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows(see figure 50-4):

a. For v negative, $R_a(Y) = -x \sinh |v| + y \cosh v$
$R_{a+1}(X) = x \cosh v - y \sinh |v|$ (see 50.5.2)

b. For v positive, $R_a(Y) = x \sinh v + y \cosh v$
$R_{a+1}(X) = x \cosh v + y \sinh v$

c. $R_{a+2}(W) = 0$

*$K_1 = 1.15217_8$

50.5 Other functions. -

50.5.1 Natural logarithm. - The 37 RR instruction with m=6 (hyperbolic vector) shall form the natural logarithm of a number. When the input parameters x-1, x+1, and 0 are contained in registers R_a, and R_{a+1}, and R_{a+2}, respectively, and conditions in a and b are met, the results of the instruction shall be as specified in (1), (2), and (3).

a. Substitute p for (y/x) or $(\frac{x+1}{x-1})$

b. Using the identity $\tanh^{-1}(p)=\frac{1}{2}\log_e \frac{1+p}{1-p}$ and substituting $(\frac{x+1}{x-1})$ for p shall give the following result:

(1) $R_a\ (y) = 0$

(2) $R_{a+1}\ (X) = 2x$

(3) $R_{a+2}\ (W) = \frac{1}{2}\log_e x = \tanh^{-1}(\frac{x+1}{x-1})$

50.5.2 Exponential (e^v). - The 37 RR instruction with m=7 (hyperbolic rotate) shall form the value e^v. When input parameters 1, 1, and v positive are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows:

a. $R_a\ (Y) = e^v = \sinh v+\cosh v$

b. $R_{a+1}\ (X) = e^v \cosh v+\sinh v$

c. $R_{a+2}\ (W) = 0$

50.5.3 Sin θ and cos θ. - The 37RR instruction with m=1 (trigonometric rotate) shall form the cos θ and sin θ. When input parameters 0, 0.46672 , and θ are contained in registers R_a, R_{a+1}, and R_{a+2}, respectively, the results of the instruction shall be as follows:

a. $R_a\ (Y) = \sin \theta$

b. $R_{a+1}\ (X) = \cos \theta$

c. $R_{a+2}\ (W) = 0$

APPENDIX 60

60.1 Scope. - This Appendix shall establish the requirements for NDRO's. These requirements must be adhered to when nonstandard NDRO programs are preapred.

60.2 Illegal NDRO instructions. - The following instructions are not executable in NDRO memory unless their operands reside in main memory.

a. All floating point instructions using RI and RX Formats.

b. All instructions using the following operation codes and Formats:

Op Code	Formats
02	RI, RX
06	RI, RX
12	RI, RX
16	RI, RX
21	RI, RX
23	RI, RX
56	RI, RX
57	RI, RX

INDEX