AN/AYK-14(V)

NAVY STANDARD
AIRBORNE COMPUTER

TECHNICAL DESCRIPTION

STANDARD AIRBORNE COMPUTER
★ AN/AYK-14 ★

CONTROL DATA
CORPORATION

RESTRICTIVE NOTICE

This manual is intended to inform the reader regarding the general construction, operational characteristics, and capabilities of the AN/AYK-14 computer. It should not, however, be considered as an equipment specification, and Control Data in no way warrants the accuracy or completeness of the manual for procurement purposes. Products and services described herein are available for sale only to the federal government of the United States of America or its designees.

AN/AYK-14 (V)

**TECHNICAL
DESCRIPTION**

**CONTROL DATA
CORPORATION**

TABLE OF CONTENTS

TABLE OF CONTENTS (CONT.)

TABLE OF CONTENTS (CONT.)

LIST OF TABLES

LIST OF ILLUSTRATIONS

AN/AYK-14(V) TECHNICAL DESCRIPTION

SUMMARY

The AN/AYK-14(V) computer system is a family of microprogrammed computers designed to provide low-cost standard airborne computers applicable to a wide range of vehicles and missions. The AN/AYK-14(V) computers operate in MIL-E-5400 environments; however, the basic module design is also applicable to configurations for shipboard and land environments.

The AN/AYK-14(V) system architectural philosophy is based on the following key features:

1) The AN/AYK-14(V) architecture and instruction set is upward compatible with that of the AN/UYK-20, permitting the adaptation and use of existing AN/UYK-20 support software.

2) The hardware is functionally partitioned into pluggable modules. These modules are the standard building blocks used in configuring functionally large or small computers.

3) Intermodule communications are standardized via uniform internal bus structures to permit reconfiguration and new module addition without impact on the architecture.

These combined features permit configuration of specific AN/AYK-14(V) computers to efficiently meet the processing requirements of a wide variety of military systems. Currently the AN/AYK-14(V) computer system provides 22 module types which can be configured in various combinations in three different chassis types. Configurations range from a two-module dedicated processor to multiple processor computers with up to 524,288 words of memory and with additional chassis, up to 16 I/O channels of various types. Figure 1 illustrates the hardware, including the functional modules and typical airborne type enclosure.

BACKGROUND

The AN/AYK-14(V) is designated as the U.S. Navy Standard Airborne Computer and is intended for broad applications to a wide range of airborne missions. The design selected by NAVAIR for the AN/AYK-14(V) is a subset of the recently developed CDC 480 computer family.

Initial applications under the AN/AYK-14(V) program include the two central computers on the F-18 fighter, the computer for the LAMPS MKIII, and eight other airborne applications. The AN/AYK-14(V) program provides for qualification, documentation, training, logistics support, and extensive software transferred from the AN/UYK-20 program.

GENERAL CHARACTERISTICS

The AN/AYK-14(V) is a variable configuration, general-purpose, 16-bit computer featuring a performance range of up to 675 KOPS (thousands of operations per second). The computer features a high degree of functional and mechanical modularity and is designed for flexible growth and extensive hardware commonality over a wide range of applications. The AN/AYK-14(V) architecture discernable to the user is not changed by modular hardware configuration changes, permitting common firmware and support software systems for all users. These design concepts are the key to providing a low-cost, versatile Navy standard airborne computer system. Table 1 summarizes the AN/AYK-14(V) specifications and features.

The AN/AYK-14(V) consists of a family of pluggable modules, chassis, interconnecting buses, support equipment, software, firmware, documentation, and training necessary to provide the user with a completely supported computer system. Figure 2 depicts the system elements by subsystem and shows the functional modules applicable to each subsystem. Table 2 briefly defines the AN/AYK-14(V) element nomenclature with each of the elements described more fully in the Module Description section.

Figure 1. AN/AYK-14(V) Modules and Enclosure

TABLE 1. AN/AYK-14(V) COMPUTER SYSTEM SPECIFICATIONS AND FEATURES

GENERAL FEATURES

GP, 16-bit digital computer
Physically and functionally modular
Expandable by plug-ins and additional enclosures
Microprogrammed, emulates extended AN/UYK-20
LSI components
ATR enclosures
Variable configurations

CENTRAL PROCESSOR

Microprogrammed
2's complement arithmetic
Executive and user states
Two sets of 16-word by 16-bit general registers
Two status registers
Three-level interrupt system
Addressing to 524,288 words
Fixed and floating point arithmetic
4-, 8-, 16-, and 32-bit operands
16- and 32-bit instructions
Direct, indirect, and indexed addressing
Optional hardware floating point module
Loadable/readable 32-bit RTC clock, 1-MHz rate;
        16-bit monitor clock, 10-KHz rate
Built-in-test functions
Bootstrap PROM memory
Power failure shutdown/recovery
I/O controller capability:
        Chaining capability
        Control memory for each channel
        Up to 16 channels in various combinations
Interface to support equipment
Sample instruction times:
        Shift   1.5 usec
        Add, subtract   0.8
        Multiply   4.2
        Divide   8.4
        Basis: single GPM, core memory, overlapped
                    access, interleaved addresses

MEMORY CONTROL AND MEMORY

Core memory module (CMM), 32K words of 18
    bits
Semiconductor memory module (SMM), 32K
    words of 18 bits
Interchangeable core and semiconductor memory
    modules
CMM has 900-nanosecond cycle time and 350-
    nanosecond access time
SMM has 400-nanosecond cycle time and 200
    nanosecond access time
Interleaved or non-interleaved addressing
Read/write expandable memory (RXM), 4K x 18-
    bit RAM with optional 4K PROM
Parity bit per byte
Protect features:
    Write protect
    Read protect
    Execute protect
    Block protect in paging system
Memory controller with paging to 524,288 words

I/O PROCESSOR (OPTIONAL)

I/O controller capability
Instruction subset compatible with central pro-
    cessor
Microprogrammed
Usable in conjunction with central processor or as
        stand-alone processor
Real-time and system clocks
16-word by 16-bit general register set
Addressing to 65,536-words
Fixed point 16-bit arithmetic
Interface to support equipment

DISCRETE INTERFACE

Eight program selectable external device interrupts
32 bidirectional input or output discretes
16 differential input discretes
16 "switch closure" input discretes

3

SERIAL 1553A INTERFACE

MIL-STD-1553A redundant multiplexed bus
1-MHz bit rate
32 terminals per bus and 1 bus controller
Operation as bus controller or remote terminal

NTDS CHANNELS

MIL-STD-1397
Parallel channels:
    NTDS slow (41,667 words/sec)
    NTDS fast (125K words/sec)
    ANEW (125K words/sec)
    16-bit and 32-bit (dual channel)(125K words/
        sec) operation
    Computer-to-peripheral and computer-to-
        computer modes
    Externally specified addressing on dual channels
Serial channels:
    10-MHz bit rate
    16- or 32-bit (dual channel) message formats

RS-232-C SERIAL INTERFACE

Asynchronous 75 to 9600 baud
Synchronous to 9600 baud

PROTEUS INTERFACE

10-MHz bit rate
Serial transfer, 32-bit message format

PHYSICAL

Chassis	Height	Width	Depth*	Weight*
XN-1	7.62"	10.12"	19.56"	45-55**
XN-5	7.62"	10.12"	14.00"	42
XN-3	7.62"	10.12"	12.75"	35-45**

NOTES: *Does not include fan
           **Weight varies as functions of optional
              modules installed

Service conditions as specified in MIL-E-5400 for
    class 1, 1A, 1B, and 2X equipment

PRIMARY POWER

115-Vac, 400-cycle, three-phase, wye-connected
    as per MIL-STD-704B
400 to 600 watts for XN-1*
250 to 450 watts for XN-5*
150 to 350 watts for XN-3*

NOTE: *Power varies as function of optional
           modules installed.

AVAILABLE OPTIONAL BOLT-ON FAN COOLING

Fan	Length	Dia.	Weight (LB)	Power (at S.L.)	Altitude (FT)
IMC 5026	3.10"	2.75"	2.00	100	30,000

Processing Subsystem

The general processing module (GPM) contains all the microprogrammed control, arithmetic unit, registers, and bus interfaces. The processor support module (PSM) contains the supporting elements such as micromemory, real-time clocks, bootstrap memory, bus interfaces, and event (interrupt) logic required to complete the function of the GPM. Together they form a 16-bit central processing unit (CPU) of a general-purpose computer. The extended arithmetic unit (EAU) provides high-speed, 32-bit, floating-point hardware and operates under the control of the GPM.
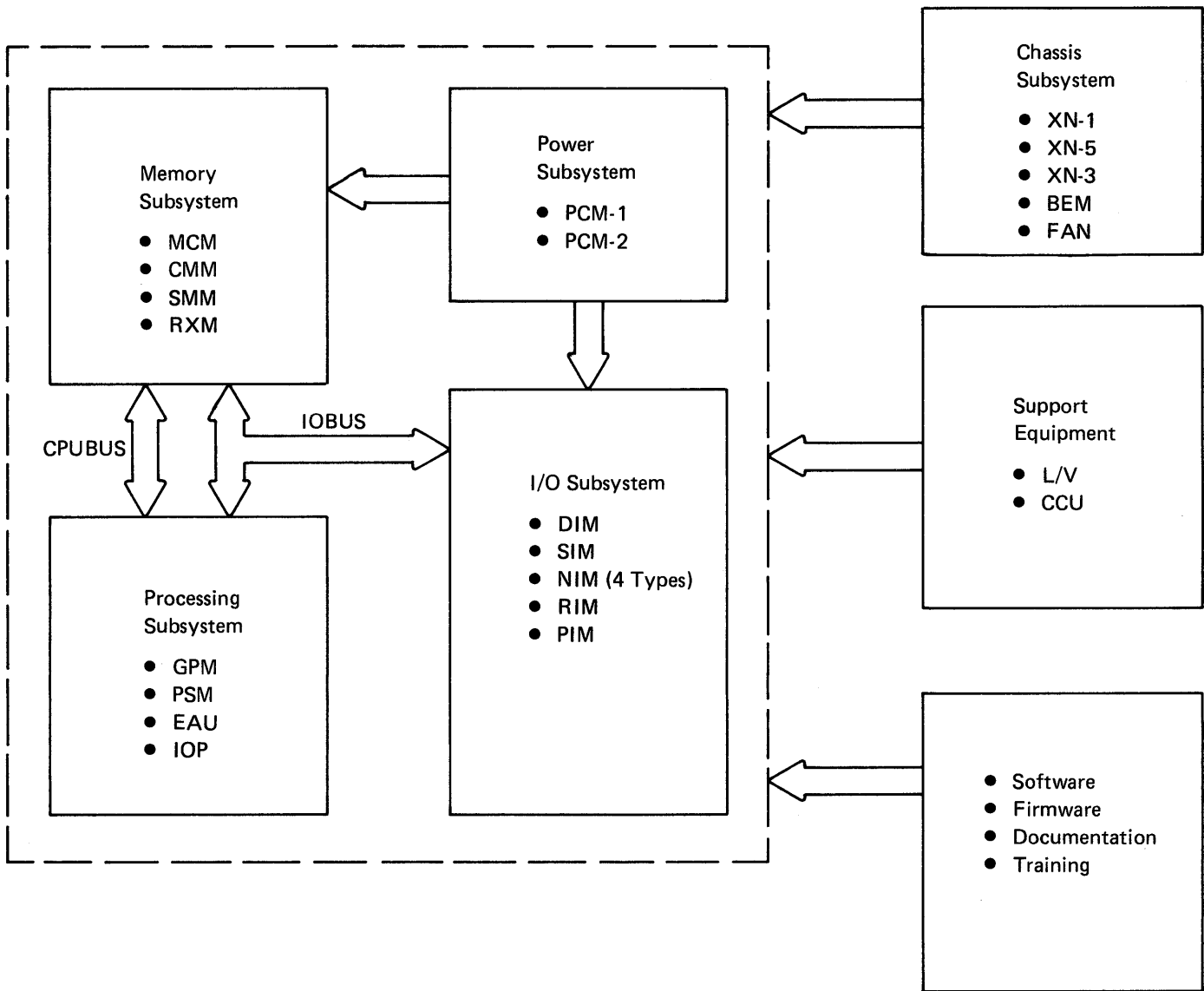
Figure 2. AN/AYK-14(V) System Elements

The incorporation of an IOP into an AN/AYK-14(V) system, operating in conjunction with the central processor, greatly enhances the processing throughput. The IOP combines the basic functions of the GPM and PSM on one module with a reduced instruction set and performance level. The IOP is microprogrammed to serve either as an IOC or as a single-module, 16-bit, general-purpose CPU without modification.

Intermodule Communication

The functional modules communicate via one or two identical internal buses: the CPUBUS and the IOBUS. These high-speed, 24-bit parallel buses are the principal data transfer paths between processing modules, memory, and the I/O channels. Additional control signals are transmitted via the EVENTBUS, which transfers interrupt and other event signals. Internal common module interfaces permit flexible module configuration and ensure that module modification or addition of new types will not result in existing module modification.

5

TABLE 2. HARDWARE IDENTIFICATION

Module	Typical Power (Watts)	Typical Weight (Lb)	Name	Function
GPM	46	2.1	General processor	16-bit processor with control, registers, and arithmetic unit
PSM	39	2.1	Processor support	Micromemory, real-time clock, interrupt system
EAU	44	2.1	Extended arithmetic unit	High-speed, floating point arithmetic
IOP	44	2.2	I/O processor	16-bit computer with control, registers, arithmetic unit, micro-memory, and interrupt system
MCM	35	1.9	Memory control	Memory controller with paging, protect, parity, and two ports
CMM	38	3.1	Core memory	32K words by 18-bit core memory
SMM	20	3.3	Semiconductor memory	32K words by 18-bit semiconductor memory
RXM	12	1.0	Read/write expandable memory	4K words by 18-bit semiconductor RAM memory with optional addition of 4K PROM
DIM	14	1.0	Discrete I/O	32-input discretes, 32-I/O programmable discretes, eight interrupts
SIM	17	1.1	Serial I/O	1553A serial multiplex channel, 1-MHz bit rate
NIM	11	1.1	NTDS fast I/O	NTDS fast interface 125K words/second
NIM	13	1.1	NTDS slow I/O	NTDS slow interface, 41,667 words/second
NIM	10	1.0	NTDS ANEW I/O	NTDS ANEW interface 125K words/second
NIM	15	1.1	NTDS serial I/O	NTDS serial interface, 10 Mb/s

TABLE 2. HARDWARE IDENTIFICATION (CONT.)

Module	Typical Power (Watts)	Typical Weight (Lb)	Name	Function
RIM	8	1.0	RS-232-C I/O	One serial RS-232-C channel, 9600 bauds
PIM	13	1.1	PROTEUS I/O	I/O serial channel pair, 10-MHz bit rate
BEM	24	1.0	Bus extender	Extends all AN/AYK-14(V) internal buses outside the enclosure
*PCM-1 PCM-2	Varies	9.5 11.4	Power converter	Regulated power supply. MIL-STD-704 power input, status outputs
Chassis				
XN-1	N/A	16.3		19.56" by 10.125" by 7.625" ATR enclosure and chassis
XN-5	N/A	11.9		14" by 10.125" by 7.625" ATR enclosure and chassis
XN-3	N/A	12.3		12.75" by 10.125" by 7.625" ATR enclosure and chassis for computer extension
Support Equipment				
L/V	80	43	Loader/verifier	Portable militarized tape loader and control panel
CCU	440	383	Computer control unit with tape unit and formatter	Laboratory operator console for firmware and software debugging and maintenance. Interfaces to commercial peripherals

NOTE: *PCM-1 provides 410 watts of output power, $n \approx 71\%$.
PCM-2 provides 540 watts of output power, $n \approx 71\%$.

7

Memory Subsystem

The memory subsystem includes interchangeable 32K-word core memory modules (CMM) and 32K-word semiconductor memory modules (SMM) with 18-bit word length. The CMM cycle time is 900 nanoseconds and the SMM cycle time is 400 nanoseconds. The memory control module (MCM) interfaces between the GPM and the memory modules (CMM or SMM). The MCM has both CPUBUS and IOBUS interfaces which permit the GPM to use one bus for instruction access and the other for operands to enhance effective access time. The MCM also provides two channels to memory modules, the OMEMBUS and EMEMBUS, which can increase effective access time through interleaved addressing between two memory banks.

The read/write expandable memory module (RXM) is a 4K word by 18-bit semiconductor memory with 400-nanosecond cycle time which operates directly with the IOP via the IOBUS. An optional addition of 4K read-only memory (PROM) is also available. The primary application of the RXM is to provide memory for use with the IOP as a controller or as a small dedicated 2-card computer.

I/O Subsystem

The AN/AYK-14(V) system organization provides for up to 16 I/O channels, each on individual functional modules which communicate with the computer system via the IOBUS interface. The standardization of internal interfaces permits any I/O channel module type to be interchanged in the chassis I/O slots by simple plug-in replacement. Available chassis provide from four to six I/O channels, including the DIM. Expansion to more I/O channels requires the additional XN-3 type enclosures. A number of I/O module types are currently available to match standard I/O channel characteristics. These are:

- MIL-STD-1553A avionics serial multiplex bus
- NTDS (fast, slow, ANEW, and serial) MIL-STD-1397
- RS-232-C
- PROTEUS

The I/O controller (IOC) functions can be executed by either the central processor (GPM and PSM) or the optional I/O processor (IOP).
Special I/O channel types may be added as required without modification to backpanel wiring, internal interfaces, or microcode. This is an important feature of the AN/AYK-14(V), since a principal problem area in military system applications involves accommodating special equipment and sensor interfaces.

User-equipment interrupts can be brought into the system either through the associated I/O channel or via the discrete interface module (DIM), which also has provision for 32 input and 32 input or output discrete signals.

Expansion of the computer beyond a single enclosure or implementation of direct memory access (DMA) I/O is effected through the use of the bus extender module (BEM), which provides a buffered extension of all internal computer buses to another enclosure.

Power Subsystem

Power for all modules in an enclosure is supplied by a power converter module (PCM) with appropriate regulated voltage and current capabilities. Present designs operate on MIL-STD-704 power, 115-Vac, 400-cycle, three-phase, wye-connected.

Chassis Subsystem

All modules plug into an ATR-type chassis equipped with slots to accommodate a combination of module types. Currently three standard chassis types designed for MIL-E-5400, Class II environments are available for 16-bit computers. Figure 3 shows the three standard chassis types along with the module configurations available for each. Connector location and basic dimensions are shown. It should be noted that the XN-3 is an extension unit to be used with the XN-1 chassis to provide additional memory, processing, and/or I/O capability. Multiple XN-3 chassis can be used to further expand the system.

Environment

The basic module of the AN/AYK-14(V) family is designed for use in MIL-E-5400 (airborne) when installed in a suitable enclosure. The total range of conditions includes temperatures of $-54^{\circ}C$ to $71^{\circ}C$, altitudes to 70,000 feet, and levels of shock, vibration, humidity, and EMI appropriate to these environments.

Qualification of the chassis types listed in Table 2, to MIL-E-5400, class 2 requirements will occur under the current AN/AYK-14 contract from the U.S. Navy.

All modules are designed for conduction cooling via a heat sink backing the printed circuit boards. The modules have ramp clamps along both short edges to provide solid mechanical and thermal contact to the slots in the chassis. Heat is transferred from the chassis heat sink via an air plenum, which may be supplied by a vehicle cooling air system or optional bolt-on fan. No cooling air is needed over module components. Figure 4 illustrates a single and a double module.

The rigid module structure, stiffened by the heat sink, withstands severe shock and vibration environments.

All modules are conformal coated for moisture resistance. The module design permits great flexibility in chassis cooling provisions to meet multiple application requirements.

Configuration Capability

The functional partitioning of the modules and the internal bus structures provide for flexible configuration of a wide range of AN/AYK-14(V) computers. The AN/AYK-14(V) system allows the building up the system by addition of modules to meet the problem computing bandwidth and capacity requirements. Some examples are given to show how these building blocks can be used to balance computer size, weight, power, and cost against performance.

Figure 5 shows the minimum AN/AYK-14(V) computer configuration, which consists of an IOP as the 16-bit processor and an RXM 4K by 18-bit random access memory (RAM) semiconductor memory (with optional 4K PROM). This is a "bare" module configuration and assumes that the modules are incorporated as components into the user's equipment. The user's equipment power supply would provide regulated 5-Vdc power for the modules and the user would also provide the I/O adaptation to the IOBUS interface. IOP/RXM combinations can also be used effectively as computing elements in distributed processing systems.

An expanded configuration (Figure 6) yields a complete 16-bit, general-purpose processor with high-speed floating-point hardware, hardware I/O controller, 128K words of 18-bit core memory, and up to 16 I/O channels of various types. This example illustrates the role of the identical CPUBUS and IOBUS in organizing the modules into a powerful computer. Since the GPM has two bus interfaces to the MCM, it is possible to overlap instruction and operand fetches from memory. In addition, it should be noted that the two memory channels, OMEMBUS and EMEMBUS, permit interleaving of memory addresses between memory banks for high, effective access speed.

AN/AYK-14(v) CONFIGURATOR

XN-1 COMPUTER
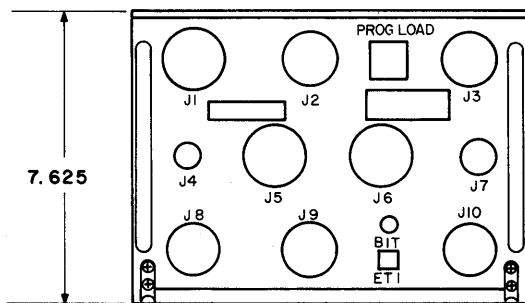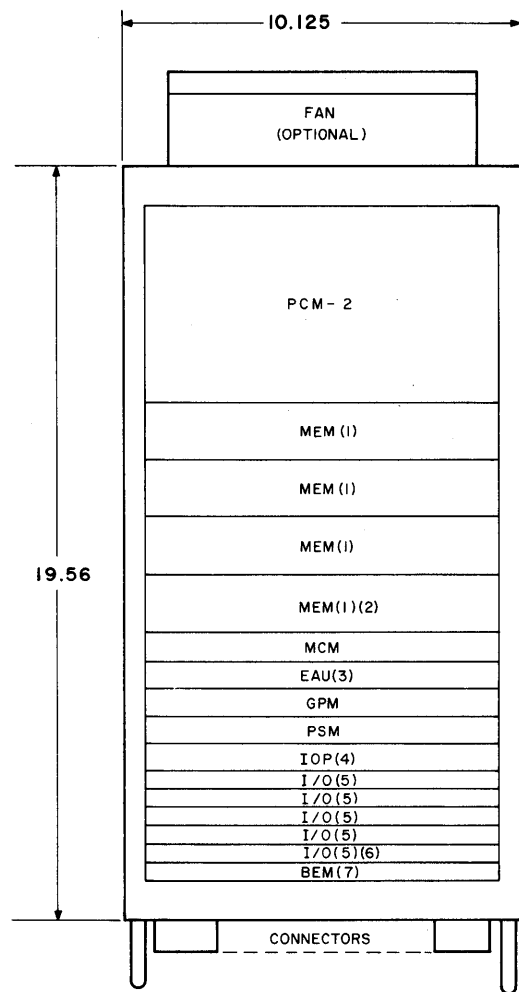
STANDARD

The XN-1 contains the following basic module set:

- Power Conversion Module (PCM-2)
- General Processor Module (GPM)
- Processor Support Module (PSM)
- Memory Control Module (MCM)

OPTIONS

1) Combinations of the following memory modules, up to four total:

- 32K x 18-word Core Memory Module (CMM-32)
- 32K x 18-word Semiconductor Memory Module (SMM)

2) 4K x 18-word Read/Write Memory (RXM) is used with the IOP (additional 4K PROM optional).

3) Extended Arithmetic Unit (EAU) for high speed floating point is optional.

4) Input/Output Processor (IOP) for independent I/O processing is optional.

5) Any combination of the following I/O modules, up to five total:

- 1553A Serial (SIM)
- NTDS Fast (NIM-B)
- NTDS Slow (NIM-A)
- NTDS ANEW (NIM-C)
- NTDS Serial (NIM-S)
- PROTEUS (PIM)
- RS-232 (RIM)
- Discrete I/O (DIM) with 32 discretes and 8 interrupts wired to the front panel.

6) Discrete I/O (DIM) with 64 discretes and 8 interrupts wired to the front panel.

7) Bus Extension Module is required if additional memory or I/O expansion is needed.

Connector	Type	Use
J1	MS27656T25-35	I/O
J2, J8, J9, J10, J3	MS27656T23-35	I/O
J5, J6	MS27656T25-35	BSXTND
J4	MS27468T11-35	Power
J7	MS27656T17-35	Maintenance

Figure 3. AN/AYK-14(V) Configurator – XN-1 Layout (Sheet 1 of 3)

10

AN/AYK-14(v) CONFIGURATOR

XN-5 COMPUTER

STANDARD

   The XN-5 contains the following basic module set:

- Power Conversion Module (PCM-1)
- General Processor Module (GPM)
- Processor Support Module (PSM)
- Memory Control Module (MCM)

OPTIONS

   1)   Combinations of the following memory modules, up to two total:
- 32K x 18-word Core Memory Module (CMM-32)
- 32K x 18-word Semiconductor Memory Module (SMM)

   2)   Input/Output Processor (IOP) for independent I/O Processing is optional.

   3)   Any of the following I/O modules

- 1553A Serial (SIM)
- PROTEUS (PIM)
- NTDS Serial (NIM-S)

   4)   Any of the following:
- SIM   ● NIM-S
- NIM-A   ● PIM
- NIM-B   ● RIM
- NIM-C   ● DIM

(Discrete I/O with 32 discretes and 8 interrupts wired to front panel)

   5)   Discrete I/O (DIM) with 64 discretes and 8 interrupts
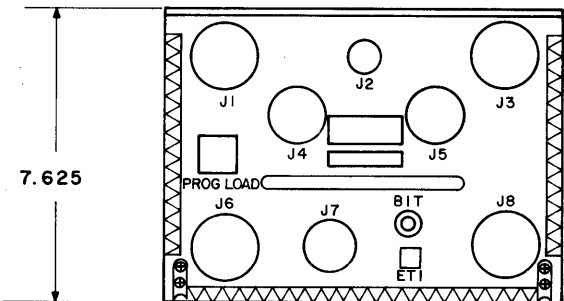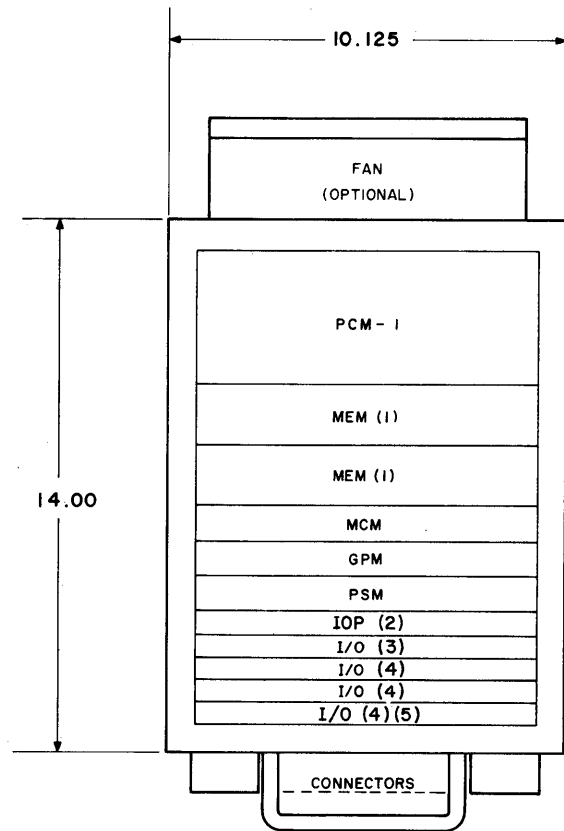
Connector	Type	Use
J1, J3	MS27656T25-35	I/O
J6, J8	MS27656T25-35	I/O
J2	MS27468T11-35	Power
J4, J5	MS27656T15-35	I/O
J7	MS27656T17-35	Maintenance

Figure 3. AN/AYK-14(V) Configurator — XN-2 Layout (Sheet 2 of 3)

11

AN/AYK-14(v) CONFIGURATOR
XN-3 COMPUTER
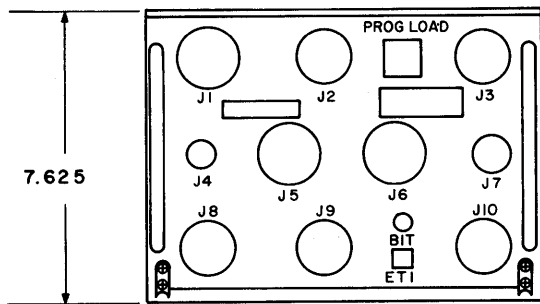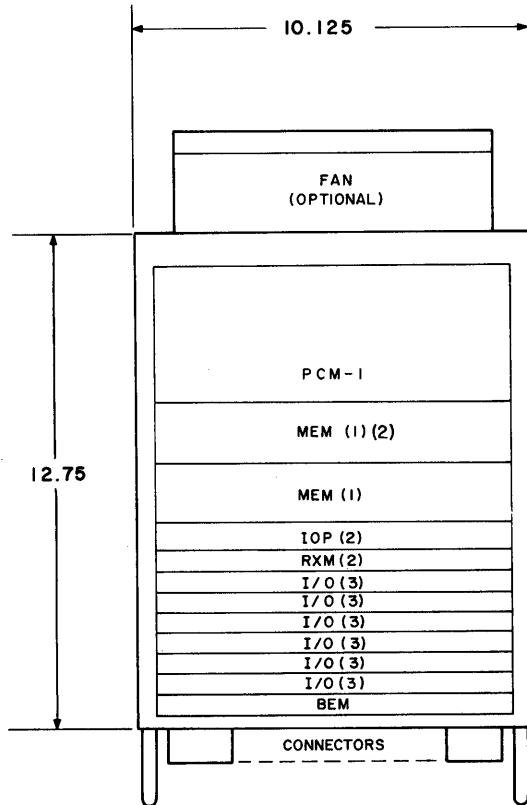
STANDARD

    The XN-3 contains a Power Conversion Module (PCM-1) and a Bus Extender Module (BEM) for interface to the XN-1 computer.

OPTIONS

    1) Memory expansion with the following modules, up to two total:

- 32K x 18-word Core Memory Module (CMM-32)
- 32K x 18-word Semiconductor Memory Module (SMM)

System memory expandable to 512K words.

    2) Input/Output Processor (IOP) is optional as a XN-3 stand-alone processing unit; a 4K x 18-word Read/Write Memory (RXM) and/or one additional memory is used with the IOP (additional 4K PROM optional).

    3) Any combination of the following I/O modules, up to six total:

- 1553A Serial (SIM)
- PROTEUS (PIM)
- RS-232 (RIM)
- NTDS Fast (NIM-B)
- NTDS Slow (NIM-A)
- NTDS ANEW (NIM-C)
- NTDS Serial (NIM-S)
- Discrete I/O (DIM) with 32 discretes and 8 interrupts wired to the front panel.

NOTE: System I/O expandable to 16 I/O channels.

Connector	Type	Use
J1, J3	MS27656T25-35	I/O
J5, J6	MS27656T25-35	BSXTND
J2, J8, J9, J10	MS27656T23-35	I/O
J4	MS27468T11-35	Power
J7	MS27656T17-35	Maintenance

Figure 3. AN/AYK-14(V) Configurator – MEU Layout (Sheet 3 of 3)
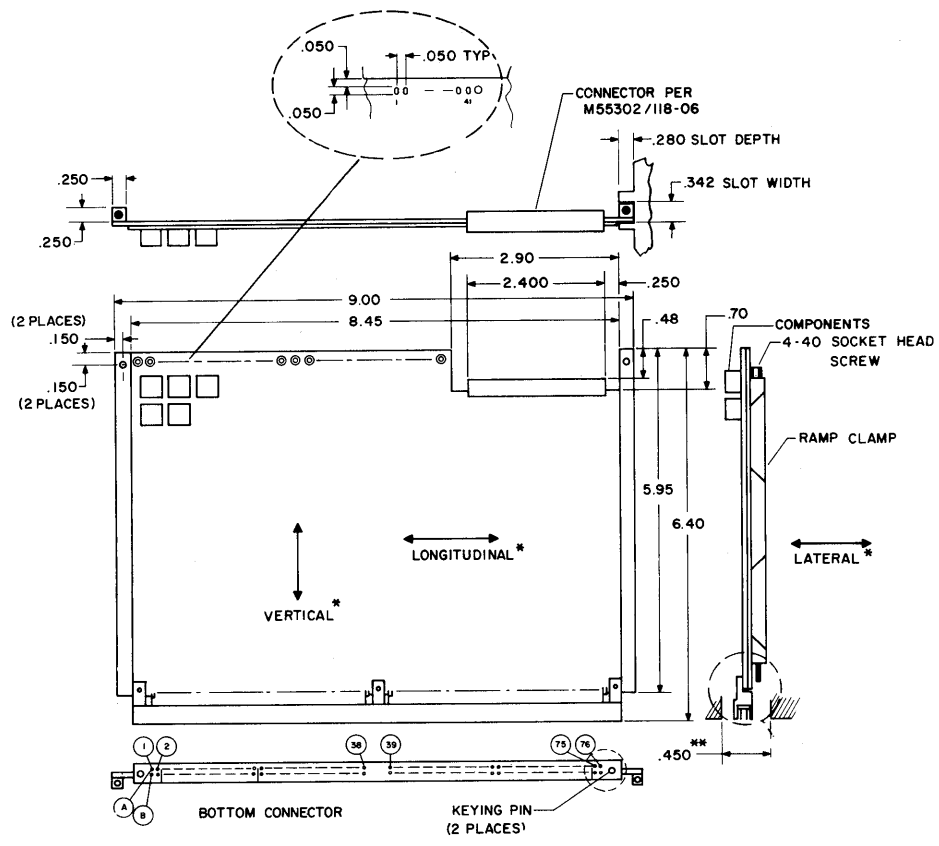
SINGLE
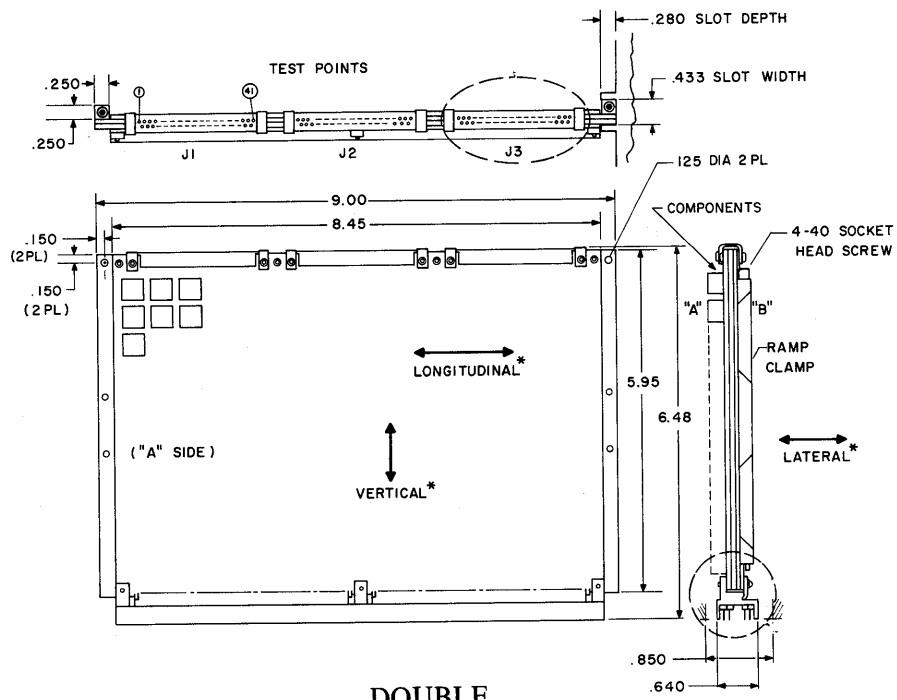
DOUBLE

Figure 4. Module Outlines

13
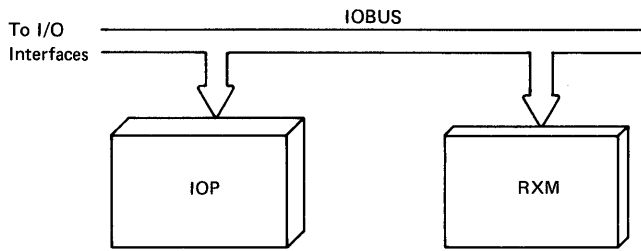
Figure 5. Minimum Configuration
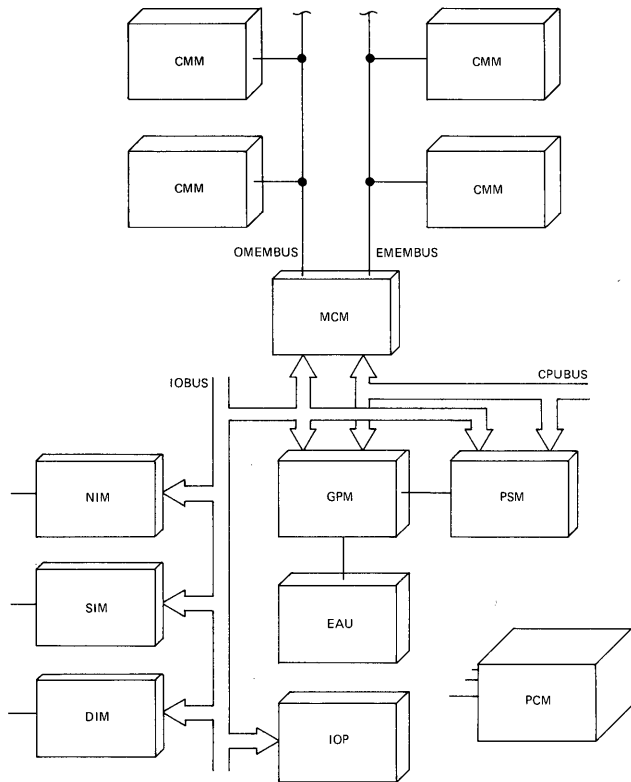
Figure 6. Expanded Configuration

PHYSICAL CHARACTERISTICS

The physical characteristics of an AN/AYK-14(V) computer depend on the module complement and chassis type required to meet the processing and environmental requirements.

All processor modules, except the PCM, are 6 by 9 inches. The GPM, PSM, BEM, MCM, and IOP modules are mountable on 0.85-inch centers

and weigh about 2 pounds each. All I/O modules are mountable on 0.45-inch centers and weigh about 1.1 pound each. Memory modules are mountable on 1.45-inch centers and weigh about 3.1 pounds each. Modules and chassis have a provision for keying to prevent improper module insertion into the chassis.

RELIABILITY

To ensure a high degree of reliability, a reliability plan was implemented during the design phase. This plan was prepared in accordance with MIL-STD-785A and paragraph 3.1.1.4 of RM-533D2.1.

This plan provides for certain proven reliability concepts to achieve AN/AYK-14(V) reliability requirements. These concepts are as follows:

- Reliability disciplines were integrated into the design process.
- All junction temperatures are less than 110°C.
- Management reviews enforced engineering disciplines.
- A rigorous parts quality program was implemented.
- Stringent subassembly screening programs were established.
- Test programs apply profiles for operation and environmental stress.
- A comprehensive failure reporting program was implemented.

Various reliability analyses were conducted to ensure reliability requirement compliance. These analyses are:

1) Failure mode and effects
2) Fault tree
3) Worst case
4) Sneak circuit
5) Thermal
6) Dynamic

14

Table 3 illustrates the results of these reliability efforts by showing predicted MTBF for two configurations in a MIL-E-5400 class 2 environment. MTBF for other configurations will depend upon the environment and module assortment used.

TABLE 3. RELIABILITY

Module Type	No. of Modules	
	Config. 1	Config. 2
GPM	1	1
PSM	1	1
MCM	1	1
CMM	3	1
PCM-1		1
PCM-2	1	
SIM	2	1
DIM	1	1
IOP		1
PROTEUS	1	
XN-1 Chassis	1	
XN-5 Chassis		1
MTBF (Hours)	1700	2200

MAINTAINABILITY

Upon a malfunction in the AN/AYK-14(V) while installed in an aircraft, the resident diagnostic program detects the fault and identifies its occurrence on the bit fault indicator on the front of the computer. After the malfunction, the computer is removed and replaced with a properly performing computer.

The faulty computer is sent to the shop level maintenance facility where the loader/verifier (L/V) is used to isolate the malfunctioning shop replaceable assembly (SRA) through a fault isolation diagnostic.

Upon module isolation by the isolation diagnostic the computer cover is removed and the failed module is removed by the release of the ramp clamp fasteners. The module is replaced with a spare and the program diagnostic is performed to verify proper operation. The computer is now ready for return to the organizational level.

The faulty SRA is forwarded to the repair facility at the depot level for isolation and repair of the faulty components through use of automatic test equipment (ATE). Test points are also available to allow isolation to lower levels in a functional sequence, so that repairs can be made efficiently. After repair and verification checks, the SRA is returned to the shop level maintenance facility stores for reissue as required.

Predictions and analysis concerning maintainability of the AN/AYK-14(V) are in accordance with AR-10A. The MTTR at the shop level is predicted to be less than 20 minutes. This time includes the following:

1) Installation of unit for test
2) Verification of fault
3) Isolation of faulty SRA
4) Removal/replacement of faulty SRA
5) Verification of proper operation
6) Removal of unit from test

PERFORMANCE

The processing throughput of a specific AN/AYK-14(V) computer depends on the configuration of processor and memory modules used. In addition, the measured throughput depends on the mix of instruction types involved in the problem and the rate of interrupt and I/O processing required.

Processing throughputs can be characterized by performance on an instruction mix typifying usual applications. For the AN/AYK-14(V) computer, the performance on a mix containing 80-percent fixed point addition and 20-percent fixed point multiply instructions is given in Table 4 for various memory types and configurations and as a function of instruction and operand type. These preliminary performance figures are given for a single 16-bit

CPU in Table 4 (a) and for a stand-alone IOP in Table 4 (b). Table 4 (c) gives the floating point arithmetic performance of the CPU when configured with the optional EAU. The performance is given in thousands of operations (instructions) per second (KOPS).

Additional performance data is provided in the Instruction Repertoire section, which lists individual instruction execution times for a specific AN/AYK-14(V) core memory configuration using overlap and interchanging features.

Techniques for performance enhancement include:

- Use of memory overlap by using CPUBUS and IOBUS for independent access for instructions and operands
- Use of interleaved addresses between two memory banks
- Use of SMMs
- Use of the EAU for high-speed floating point (4-microsecond add, 5-microsecond multiply)

TABLE 4. AN/AYK-14 PERFORMANCE ON INSTRUCTION MIX

(a) CPU (GPM and PSM)
   80% Fixed Point Add, 20% Multiply

Instruction Type	Data Length	KOPS* Interleaved Core/Semicond.	KOPS Non-Interleaved Core/Semicond.
16-bit, RR	16	650/675	630/675
16-bit, RI	16	520/575	405/560
16-bit, RR**	32	395	395
16-bit, RI **	32	295/300	250/295
32-bit, RX	16	390/430	300/380
32-bit, RX	32	265/290	210/280

(b) IOP
   80% Fixed Point Add, 20% Multiply

Instruction Type	Data Length	KOPS
16-bit, RR	16	430
16-bit, RI	16	290
32-bit, RX	16	250

(c) EAU (with GPM and PSM)
   80% Floating Point Add, 10% Floating Multiply, 10% Floating Divide

   Typical performance is 200 KOPS but exact performance is data dependent.

NOTE: *KOPS is defined as thousands of operations per second.
   **The 32-bit data instruction reflects a 32 x 32 multiply yielding a 64-bit product.

- Use of the IOP to share I/O and instruction processing functions
- Incorporation of frequently used algorithms into direct microcode

Overlap is standard in the AN/AYK-14(V) whenever a GPM is used. Interleaving is standard in all configurations using an MCM and an even number of identical memory modules. Interleaving can be inhibited, if desired, by the use of a jumper wire on a front panel connector.

The combined use of the CPU and IOP is a plug-in enhancement available in the XN-1 and XN-5 chassis and requires no microcode changes to implement.

Use of microcoded algorithms can frequently attain a five to 10 times speedup in algorithm processing over the conventional software subroutine approach. The GPM microcommand processing time is generally 180 nanoseconds (some microcommands, particularly those involving arithmetic logic units [ALU] shifting, require 210 nanoseconds) and the IOP microcommand time is 250 nanoseconds. These times include both microcommand execution and reading the next microcommand.

Control Data is prepared to assist users in configuring suitable AN/AYK-14(V) computers to match specific application requirements and to estimate expected performance.

MODULE DESCRIPTIONS

The modules described in this section represent the current AN/AYK-14(V) module designs. It is expected that new module types will be added to meet future applications requirements. New or special modules will be designed to interface with standard AN/AYK-14(V) internal buses to preserve system integrity.

General Processor Module (GPM)

The GPM is a 16-bit microprogrammable processor based on the AMD 2900 series microprocessor slice LSI devices. The architecture is augmented for high-speed performance with additional registers, internal data, and control transfer paths. The GPM features which contribute to its performance include:

- 48-bit microcommand control
- Microprogram address sequencing to 4K words
- 180-nanosecond microcommand cycle
- 256 by 16-bit word register file
- 256 by 16-bit word multiport CFILE
- Dual identical parallel bus interfaces (CPUBUS and IOBUS)
- Event interface
- Interface to micromemory on PSM
- Serial interface to support equipment
- Interface to EAU

The GPM operates from microcommands stored on the PSM module (up to 4K words of micro-memory).

Processor Support Module (PSM)

The PSM augments GPM functions to form a complete 16-bit computer in two modules. The partitioning of the functions between GPM and PSM was designed to allocate those functions to the PSM that might require modification as applications change. The PSM features include:

- Up to 4K by 48 bits of PROM micro-memory for the GPM.
- 1K by 16 bits of PROM bootstrap memory for computer system initiation via the 1553A I/O channel.
- Two parallel bus interfaces (CPUBUS and IOBUS).
- Event interface.
- Event monitor logic, which forms the basic hardware portion of the event (interrupt) processing.
- Four loadable/readable clocks for monitoring and timing functions (1-microsecond resolution).
- 32-bit high-speed multiply logic.
- BIT timer with 2.097-second increment, 4-bit count.

17

The AN/AYK(V) computer micromemory may contain commands for processing a variety of functions including:
1) AN/AYK-14(V) instruction set interpretation and maintenance of the computer status.
2) Built-in-test (BIT) functions.
3) Diagnostic and fault isolation functions.
4) Special macroinstruction or algorithm processing.

Extended Arithmetic Unit (EAU)

The EAU utilizes the AN/UYK-20 floating-point format which consists of an 8-bit exponent and a 24-bit mantissa. Typical execution times, including GPM control, are 5 microseconds for add and 7.7 microseconds for multiply. All AN/AYK-14(V) computers configured with a GPM and PSM execute all AYK-14(V) floating point arithmetic instructions. When configured without the EAU, the instructions are implemented via firmware. The incorporation of the EAU automatically increases floating point execution speed without firmware changes.

In addition the EAU also performs the following instructions

Square Root	Sine	Arcosine
Exponent	Cosine	Arcsine
Log base e	Tangent	Arctangent

Arguments and results are expressed in floating point values.

Input/Output Processor (IOP)

The IOP is a complete 16-bit processor combining the basic functions of the GPM and PSM on one module. The instruction set is a subset of the total AN/AYK-14(V) instruction set. To accomplish a one-module processor, the performance and features are reduced from the GPM/PSM capability. The IOP is intended for use in three general applications types:
1) As a small scale, stand-alone, general-purpose processor with emulation capabilities.
2) As an I/O controller (IOC) in conjunction with a GPM/PSM as instruction processor.
3) As a combination IOC and instruction processor in conjunction with a GPM/PSM.

Features of the IOP include:
- 48-bit microcommand control
- Up to 2K micromemory on the module
- 250-nanosecond microcommand cycle
- 256 by 16-bit word register file
- Single parallel bus interface (IOBUS)
- Event interface
- Serial interface to support equipment
- Real-time clock with 1-microsecond resolution
- BIT timer, 2.097-second increment, 3-bit count
- Event monitor logic
- Microcommand format identical to GPM

I/O Modules

All I/O channel modules are physically the same size and are interchangeable in any I/O module chassis slot. Each I/O channel module implements a single I/O channel of a designated type and has a common set of intermodule interfaces including the IOBUS and event interface. Each I/O module type contains the logic to implement the specific channel type characteristics and operate with a standardized IOBUS communication procedure. All I/O modules have provision for a module test operation in which test data is looped through the module and returned to the processor. The standard I/O channel module set can be augmented with special channel modules to meet system requirements. The special channels will use the same IOBUS and event interface as the standard I/O modules.

Discrete Interface Module (DIM)

The DIM is used to provide a convenient interface for communicating single-bit status, event, or control information between user devices and the computer.

The DIM provides the following interface capabilities:
- Eight external device interrupts. These can have program selectable priority and can be individually masked.

18

- 32 bidirectional input or output discretes. These use differential TTL interface signals. They are program selectable as inputs or outputs in groups of four.
- 16 differential input discretes. These use differential lines, ac terminated.
- 16 "switch closure" input discretes.
- The 32 bidirectional discretes have a loop test capability.

Serial Interface Module (SIM)

The SIM implements a serial multiplex data channel meeting the channel control and format characteristics of MIL-STD-1553A. This channel type is the standard intersystem communication facility on board modern military aircraft. The module interfaces to two 1553 buses for redundant operation.

The module can operate with any MIL-STD-1553A protocol and can function as either a bus controller or remote terminal unit. Information is transferred on a single shielded twisted pair line at a 1-MHz bit rate. Data is transferred in 20-microsecond frames, each divided into 17-bit times of 1 microsecond and one 3-microsecond sync interval. All messages are addressed and use three types of words, as shown in Figure 7:

- Command word — sent by bus controller to address appropriate terminal, specify message type, and set data word count for subsequent transfer.
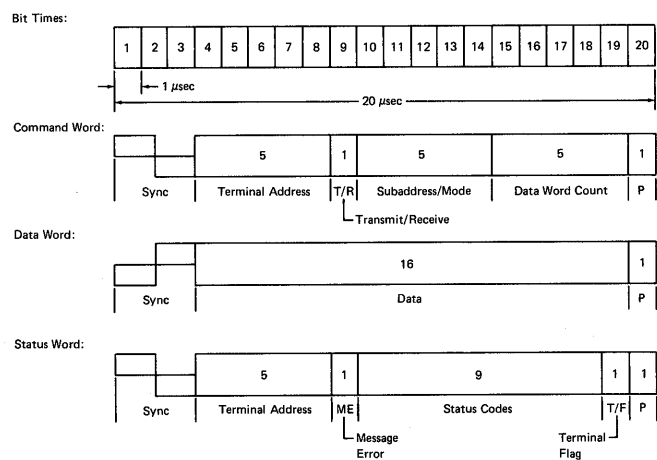
Figure 7. MIL-STD-1553A Word Formats

- Status word — set by a terminal in response to command word. Identifies terminal and reports status.
- Data word — containing 16 bits of message data, sync pattern, and a parity bit.

Up to 32 terminals and 1 bus controller can interface on a single bus. All transmissions and receptions are initiated and controlled by the bus controller using message formats as shown in Figure 8.

The SIM contains interfaces to two 1553A buses and has the capability of data transfer on one and monitoring the other at any time.
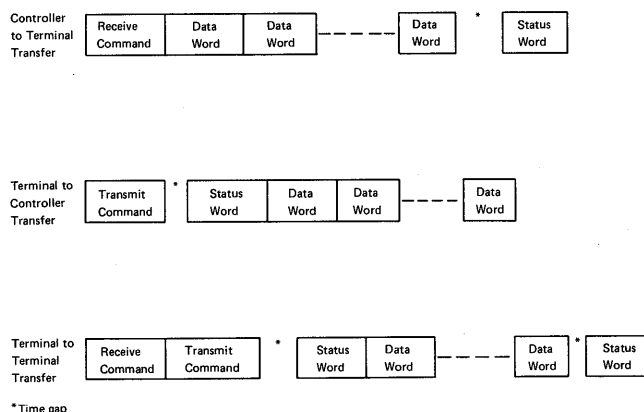
Figure 8. MIL-STD-1553A Message Formats

NTDS Interface Modules (NIM)

There are four types of NIMs, each capable of operation according to MIL-STD-1397:

1) NTDS slow — 16-bit parallel transfer up to 41,667 words per second. Binary voltage levels of 0 Vdc (logical 1) and −15 Vdc (logical 0).

2) NTDS fast — 16-bit parallel transfer of up to 125,000 words per second. Binary voltage of 0 Vdc (logical 1) and −3 Vdc (logical 0).

19

3) ANEW — 16-bit parallel transfer of up to 125,000 words per second. Binary voltage levels of 0 Vdc (logical 1) and 3.5 Vdc (logical 0).

4) Serial — serial data transfer of up to 10 megabits per second on one cable. Bipolar ±3.25V signals.

Channel interface lines for NTDS fast, slow, and ANEW are shown in Figure 9, and for serial in Figure 10. Two NIM parallel channels can be operated together to form a 32-bit wide parallel channel. Transfer operations on the serial channel involve the use of 3-bit control frames and 34-bit data frames (32-bit message data, function, or interrupt code, and 1-bit word ID, 1-bit sync), according to procedures defined in MIL-STD-1397. The modules support operation in computer-to-computer, computer-to-peripheral, externally specified addressing modes as described in MIL-STD-1397.

RS-232-C Serial Interface Module (RIM)

The RIM provides a full-duplex RS-232-C serial channel operable at selectable baud rates from 75 to 9600 baud for the asynchronous mode and in synchronous mode to 9600 baud. See Figure 11 for cable configuration.

The module can be converted to operate to MIL-STD-188C with some component changes, but without circuit board modifications.

PROTEUS Interface Module (PIM)

The PIM contains the logic to implement a PROTEUS digital channel pair capable of full-duplex data transmission at a nominal 10-MHz bit rate. The channel is designed to NADC Specification No. A30-15590.

Transmission on the PROTEUS channel is between a source and a sink, with initiation and control by the source. A source transmits 6-bit control words and 34-bit data words (32 message bits, one parity bit, and one word-type bit). The sink responds to each source word with an appropriate 6-bit control word to accomplish a positive handshaking procedure on a word-by-word basis.
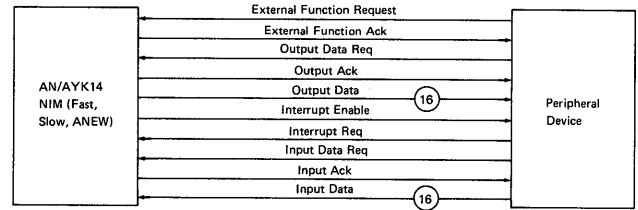
Figure 9. NTDS Slow, Fast, and ANEW Channel Interface

Figure 10. NTDS Serial Channel Interface and Message Format
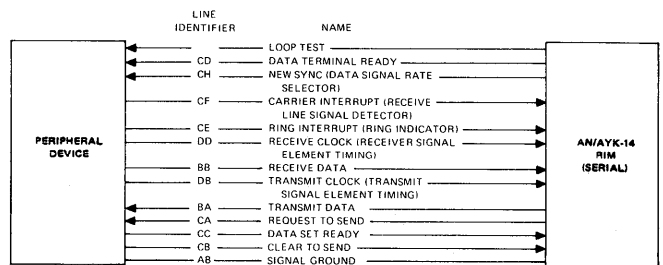
Figure 11. RS-232-C Serial Channel Interface

Parity is provided on both control and data words for error detection, and retransmission is used for error correction.

The channel pair uses a total of eight differential NRZ signals as depicted in Figure 12.
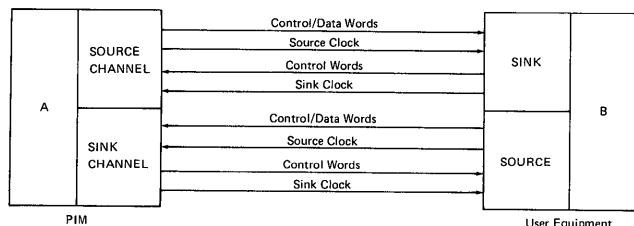
Figure 12. PROTEUS Channel Pair

Bus Extender Module (BEM)

The BEM provides an extension of the internal AN/AYK-14(V) buses and interfaces outside the enclosure to permit extension of memory, processor, and/or I/O subsystems to additional enclosures up to 15 feet (total cable length) from the computer. All voltage levels are TTL compatible and employ differential line drivers/receivers for all I/O lines. The electrical and logical design permits BEM-to-BEM communication. The BEM does not have a channel address as do other I/O modules, but instead appears transparent to bus operation. Any communication via the BEM results in a slight interface delay of approximately 75 nanoseconds in each direction, relative to direct module intercommunication. The BEM can be used to interface a DMA channel.

Memory Modules

Memory Control Module (MCM)

The MCM provides a two-port paged interface to two independent, interleaved memory channels, and thus allows simultaneous access by two users. The MCM contains the control, interface, and paging logic to operate core and SMMs with the AN/AYK-14(V) processor system. The MCM features include:

- Interfaces to CPUBUS and IOBUS
- Dual memory bus interfaces to memory modules OMEMBUS and EMEMBUS
- 16-bit address to 19-bit address paging system
- Phasing of memory modules between memory buses
- Parity bit logic, one parity bit per byte
- Block protect in paging system
  - Read protect
  - Write protect
  - Execute protect

Core Memory Module (CMM)

The CMM is available as a 32K by 18-bit word module. The CMM is a plug-in unit containing all of the specified core storage, associated drive and and sense electronics, timing and control logic, and interface circuitry. The form-factor and electrical interface of the 32K CMM is identical to the SMM which provides for complete interchangeability as shown in Figure 13.

The CMM features are:
- 900-nanosecond read/write cycle time
- 350-nanosecond access time
- Low power, average 38 watts for 32K words (based on half 1's, 50-percent standby), maximum 70 watts
- Byte operation
- Interface to 0MEMBUS or EMEMBUS
- Mountable on 1.45-inch centers
- Read/modify/write capability
- Data guard, indicates power supply out of tolerance (optional)

Semiconductor Memory Module (SMM)

The SMM provides 32K by 18-bit words in a module which is compatible to and interchangeable with the CMM.

21
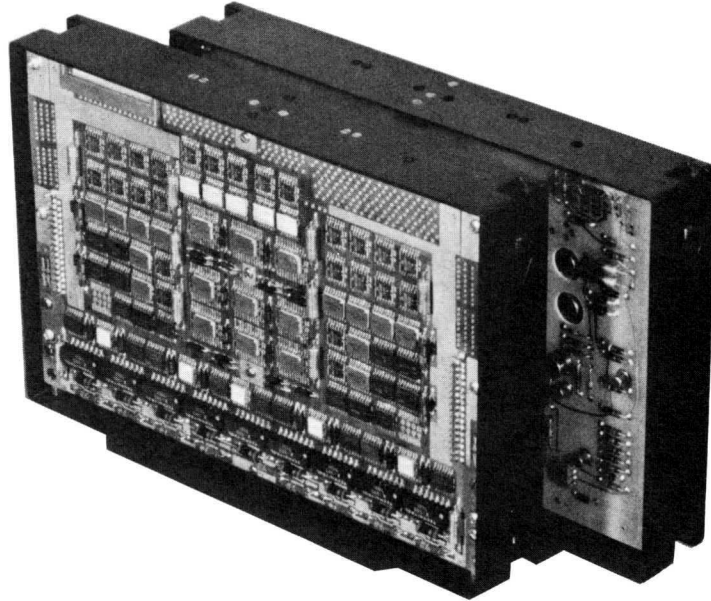
Figure 13. Interchangeable Core (Front) and Semiconductor (Rear) Memory Modules

The SMM features are:

- 200-nanosecond read access time
- Low power, 20 watts average for 32K words
- Interface to OMEMBUS or EMEMBUS
- Mountable on 1.45-inch centers

Read/Write Expandable Module (RXM)

The RXM contains 4K words by 18 bits of read/write static semiconductor memory and an optional additional 4K words of read-only memory (PROM). Features include:

- 400-nanosecond cycle time
- 275-nanosecond access time
- Interface to IOBUS or CPUBUS
- Mountable on 0.45-inch centers
- Parity logic, one bit per byte
- If the PROM option is desired, memory contents must be specified at time of order

NOTE

This memory is unpaged and does not interface with the MCM. It is intended to operate directly with a processor via the IOBUS interface. Multiple RXMs can be used in a system up to a total of 65,536 words; however, the present AN/AYK-14(V) chassis (XN-1 and XN-3) provides space for only one RXM each. The primary application of RXMs is to provide memory functions for small AN/AYK-14(V) configurations using the IOP as a stand-alone processor. An RXM can also be used as a private program memory for the IOP when used in configurations employing both the IOP and CPU in combination. In the latter case the CPU will not have access to the RXM. When installed in the XN-1 or XN-3 chassis, the RXM is assigned address ranges F000 to FFFF (Hex) for the RAM portion and E000 to EFFF (Hex) for the optional PROM portion.

Power Converter Module (PCM)

The PCM provides regulated dc power required to operate AN/AYK-14(V) modules from military aircraft power sources. Two sizes of PCMs are currently available to power various computer configurations. The PCMs are themselves modular and new capacities can be developed to meet other power source or computer configuration requirements.

The PCMs operate from 115-Vac, three-phase, 400-Hz, wye-connected input power. The design is compatible with MIL-STD-704B and MIL-STD-461A per requirements of AS-4197.

PCM-1 and PCM-2 supply thermal protection and power failure signals as well as sequencing for power-up and power-down operations. Approximately 280 microseconds are available for saving machine state and registers upon input power loss detection. PCM capacities are given in Table 5 below.

TABLE 5. PCM CAPACITIES

Module Type	Maximum Output				Size	Typical Efficiency
	+5V	+15V	-12V	-5V		
PCM-1	51A	3.3A	8.3A	1A	7" by 9" by 3.5"	71%
PCM-2	78A	3.3A	8.3A	1A	7" by 9" by 4.9"	71%

SUPPORT EQUIPMENT

Application and maintenance of the AN/AYK-14(V) computer family is aided by the availability of several pieces of support equipment.

Computer Control Unit (CCU)

The CCU is standard lab support equipment for use with AN/AYK-14(V) computers. It interfaces with the computer through the support equipment channel and provides operator console functions as well as peripheral equipment interfaces. It is intended for use as ground support equipment and is useful in application software debugging and maintenance operations. The CCU consists of the following elements mounted in and on an operator stand as shown in Figure 14 and the block diagram, Figure 15.

- CRT display and keyboard
- Magnetic tape unit and formatter
- CCU electronics package including an 8080A microprocessor and memory
- RS-232-C interfaces for optional printer and other peripherals
- Interface to the AN/AYK-14(V) computer via the computer support channel

The CCU and its peripheral equipment operate on 115V, 60 Hz, single-phase power. An external, three-phase, 115-Vac, 400-Hz supply is also required for operating the AN/AYK-14(V) computer.

The magnetic tape unit included with the CCU is a Pertec Model 7820-9-12.5 and its associated Pertec Model F829-9/7 Formatter Unit. This tape is IBM compatible, NRZI format, nine-track, and permits transfer of programs and data from a commercial computer to the AN/AYK-14(V).

The CRT and keyboard make up a Control Data Model 752-10 conversational display terminal and include a 12-inch CRT, 67-key keyboard, with the capacity to display 24 lines of up to 80 characters per line, and a self-contained character refresh memory.

A recommended useful option is an impact line printer for use in listing memory contents. This printer is a desk-top unit which uses a 7 by 7 dot matrix print head and prints at a 173-character-per-second rate. It should be noted that the CCU peripherals are controlled by the 8080A processor and do not require interface channels in the AN/AYK-14(V). This is important, since the I/O channels in the AN/AYK-14(V) can remain connected to the military system devices while the CCU is in use.
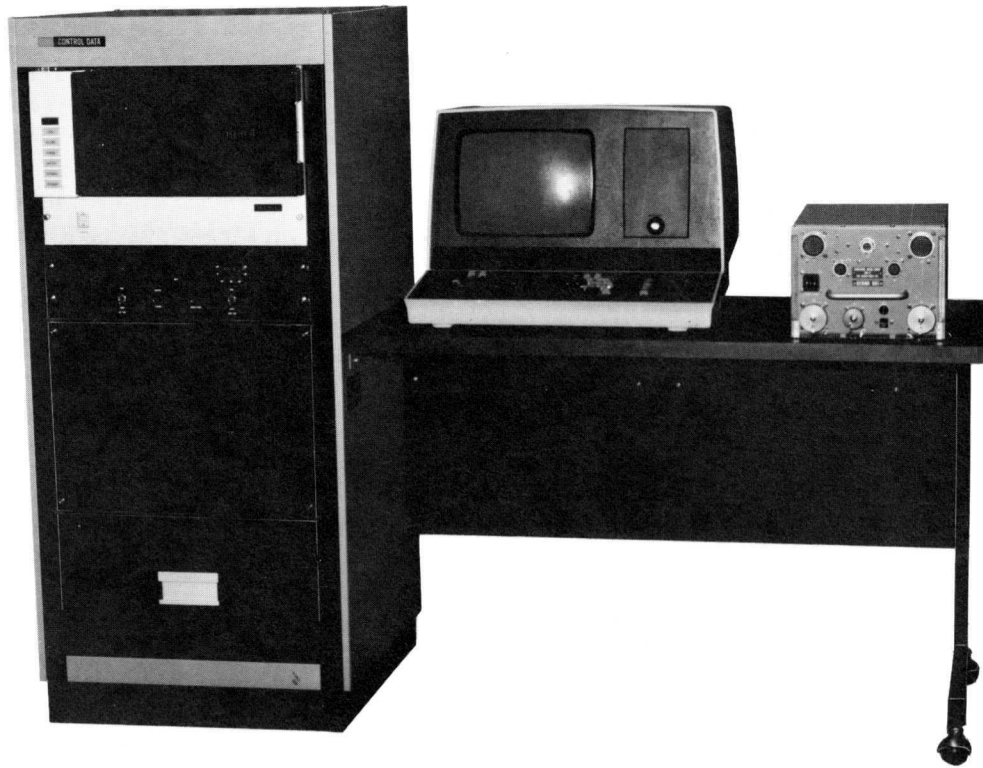
Figure 14. Computer Control Unit (Front View)

Since the CCU logic is based on an 8080A microprocessor, the CCU functions are readily modified or expanded to provide very flexible operation and adaptability to a wide range of computer support procedures.

The CCU contains all controls and displays necessary for computer hardware and software operation, monitoring, and servicing, thus ensuring simplified operation.

Switches are provided to perform the following:
- Reset the computer and maintenance/control panel circuits to initial operating conditions.
- Stop execution of the program at the completion of either the current instruction or pending I/O.
- Start execution at an address selected on the keyboard.
- Stop after completion of the previous instruction and before execution of an instruction located at a preselected address.
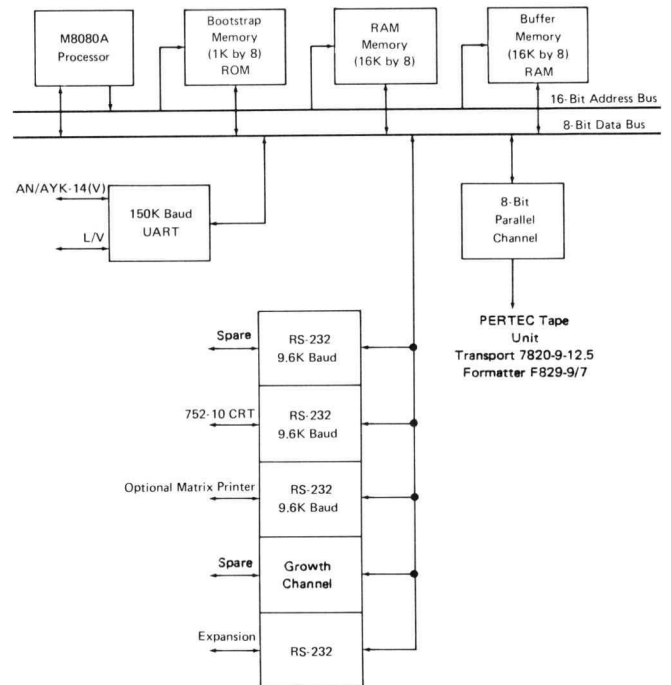
Figure 15. Computer Control Unit Block Diagram

24

- Read and write the contents of any memory location or programmable register.
- Verify and/or modify parity bits in memory.
- Release or modify memory protection. Execute the program one instruction at at time under manual control.
- Execute the program one microcommand cycle at a time under manual control.
- Inhibit all external interrupts.
- Separately inhibit RTC and timer interrupts from operating.
- Selectively dump memory contents to the L/V, a bulk storage device, or a line printer. Dumps begin at any selected memory address with no restriction on block size.
- Load and verify memory with selected blocks of data taken from the bulk storage media.
- Execute an instruction entered from the CCU without modification of the computer program counter or stored program.
- Stop at the completion of an instruction which references a preselected address.
- Load and verify computer memory from the L/V.
- Monitor and display computer diagnostic results.

Display formats can be programmed and stored into the CCU memory. Upon power up, the 8080 microprocessor will output the display format to the CRT/keyboard unit, where it is retained until power is shut down or until it is changed by the 8080 microprocessor. All of the digital information is displayed in either hexadecimal or octal notation at the operator's option. The contents of the computer registers or any memory location or block of memory locations, along with the corresponding memory address, can be selected for display on the CRT. Blocks of memory being displayed can be sequenced without preselecting a new memory location for each block.

The status of various indicators can be selected for display on the CRT. These indicators show the following switch status:

- Computer running or stopped
- I/O busy
- Machine fault
- Parity error
- Power ON/OFF
- Power fault
- Over temperature warning for computer
- Microcommand step mode

The CCU will also provide a display of the microcommand, the two previous addresses, the microcommand address, and the address of the next microcommand to be executed.

Loader/Verifier (L/V)

The L/V is a portable tape loader and computer control device designed to MIL-T-21200/2, Cat III Specifications for flight-line application. It consists of a control panel and militarized sealed cassette tape unit with a capacity for 14.4 by 10^6 bits of storage. The unit is packaged in a portable combination case and interfaces to the computer via the maintenance interface over up to 20 feet of cable. The L/V has the following capabilities:

- Load and verify operational program from cassette. The L/V has multiple file capabilities.
- Initiate AN/AYK-14(V) self test and display results on L/V control panel indicators.
- Load AN/AYK-14(V) computer diagnostics, provide operator interface and control.
- Display and change register and memory locations.

The L/V has indicators for the following functions:
- L/V POWER ON
- COMPUTER POWER ON
- ELAPSED TIME
- REGISTER DISPLAY, total of 19 alphanumeric digits
- AN/AYK-14 OVER TEMPERATURE

Switches and controls provided are:
- LAMP TEST
- MAIN POWER ON/OFF
- L/V RESTART
- Hexadecimal keyboard

The L/V operates from 115V, 400-Hz, three-phase power.

FUNCTIONAL ARCHITECTURE

The functional architecture of the AN/AYK-14(V), that is the architecture perceived by the programmer or other user, is implemented via the AN/AYK-14(V) microcode operating in a suitable configuration of AN/AYK-14(V) modules. The architecture is upward compatible with the AN/UYK-20 architecture. All instructions common to both AN/AYK-14(V) and AN/UYK-20 have identical formats, operation codes, and results.

Memory Architecture

The main memory consists of combinations of CMM and SMM modules, up to a maximum of 524,288 words, and an MCM to provide dual access port, paging, parity operation, and checking features.

Memory Interfaces

The memory interfaces to the CPU with the capability to overlap. Overlap is always used with the GPM which accesses instructions on the CPUBUS and operands on the IOBUS. The IOP interfaces to the memory system only via the IOBUS, and thus does not use overlap.

The AN/AYK-14(V) can interleave memory addresses between the memory modules interfacing on the OMEMBUS and with those on the EMEMBUS. This interleaving enhances effective access time in transferring sequentially addressable words. Whenever the configuration of memory modules is identical on both OMEMBUS and EMEMBUS, interleaving is automatically provided unless a jumper on a front panel connector is used to inhibit interleaving. Interleaving is not used when an odd number of memory modules are installed or the assortment of memory types is not symmetrical on both memory buses.

Overlap and interleaving are independent features.

Direct Memory Access (DMA) Capability

The BEM provides for extending memory interfaces external to the chassis. This module permits a DMA capability through a user-provided external DMA controller. DMA transfer does not require processing by either the CPU or IOP.

Memory Addressing

The memory addressing capability provides addressing to 524,288 words through the paging features incorporated in the MCM. The 16-bit relative address from the CPU or IOP is converted to a 19-bit address using the scheme depicted in Figure 16. The lower 10 bits of the relative address specify one of 1024 words within a page, while the upper six bits specify which of the 64 page registers will be referenced to determine the 9-bit page base address. Software instructions B0 through B7 (hexadecimal) provide the capability for loading and storing the page registers.

Memory Parity

The memory system incorporates a parity bit for each 8-bit byte. Parity is generated and checked by the MCM and an interrupt is generated upon parity error.

Memory Protection

Memory protection features on a 1024-word page basis are provided for data security and assurance of program integrity. Three types of protection are implemented via bits stored in the page register as shown in Figure 16:
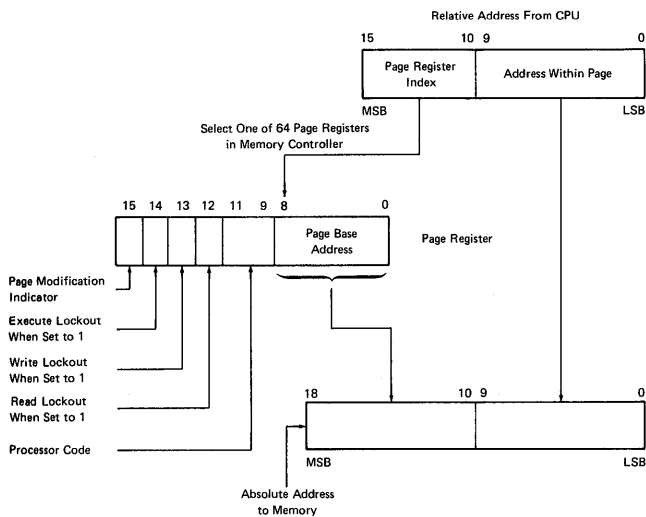
Figure 16. Memory Address Generation

- Execute Protection — generates an interrupt if instruction execution is attempted from protected page.
- Write Protection — generates an interrupt if a write operation is attempted in a protected page.
- Read Protection — generates an interrupt if a read operation is attempted in a protected page.

In addition to the three protection bits, a bit in each page register serves as a page modification register. This bit, when set, indicates that a write operation was made in the associated page. The Load Address Register instructions permit modification of protected areas.

Assigned Memory Addresses

In general programs, constants and data can be stored in any address. There are, however, some assigned locations (as shown in Table 6) which are associated with executive, interrupt, I/O functions, and NDRO mode.

Non-Destructive Read-Only (NDRO) Memory

The memory system includes two segments of words of NDRO memory containing the bootstrap program. This memory duplicates the address space of words 0 to $3F_{16}$ and $C0 - 3FF_{16}$ of main memory and is entered upon initiation of operations. A bit in status register number 1 controls the selection of NDRO or main memory. Bootstrap operations are provided via a 1553-A I/O channel.

CPU Architecture

The AN/AYK-14(V) operates in two modes:
- Executive — used for executive functions. In this mode all instructions, including privileged instructions, can be executed.
- Program — used for user program functions. In this mode any instructions except privileged instructions can be executed.

Modification of status registers and page address registers is restricted to executive mode.

This two-mode feature simplifies and increases the speed of the executive control and aids in integration of user program modules into the system software.

General Registers

The AN/AYK-14(V) has two sets of 16-word by 16-bit general registers, each set designated RO through RF (hexadecimal), and an instruction set tailored to their manipulation. The selection of the register set to be used is designated by status register 1, bit 14. These registers can be used as:
- Accumulators for arithmetic, shift, and logical operations
- Index registers for address and operand modification
- Temporary storage locations for addresses and operands

The large number of general registers and the register-register instructions, yield benefits in execution time and decreased storage requirements compared to architectures using an A/Q register organization or fewer registers.

TABLE 6. ASSIGNED MEMORY ADDRESSES

Function	CPU						IOP					
	I		II		III		I		II		III	
	Hex	Octal	Hex	Octal	Hex	Octal	Hex	Octal	Hex	Octal	Hex	Octal
Store P	58	130	50	120	48	110	68	150	70	160	48	110
Store SR1	59	131	51	121	49	111	69	151	71	161	49	111
Store SR2	5A	132	52	122	4A	112	6A	152	72	162	4A	112
Store RTC lower	5B	133	53	123	4B	113	6B	153	73	163	4B	113
P reload	5C	134	54	124	4C	114	6C	154	74	164	4C	114
SR1 reload	5D	135	55	125	4D	115	6D	155	75	165	4D	115
SR2 reload	5E	136	56	126	4E	116	6E	156	76	166	4E	116
Store RTC upper	5F	137	57	127	4F	117						

Function	CPU	IOP
I/O Command cell	$60\text{-}61_{16}, 140\text{-}141_8$	$62\text{-}63_{16}, 142\text{-}143_8$
Auto start entrance	$7F_{16}, 177_8$	
External interrupt word storage	$80\text{-}8F_{16}, 200\text{-}217_8$	$80\text{-}8F_{16}, 200\text{-}217_8$
Bootstrap ROM	$0\text{-}3F_{16}$ & $C0\text{-}3FF_{16}, 0\text{-}77_8$ & $300\text{-}1777_8$	

The general registers are referenced by the register designator fields (a,m) of the AN/AYK-14(V) instructions.

Program Address Register

The program address register, P, holds the address of the next instruction to be executed in a program sequence. Its contents are automatically advanced by one each time a single length (16-bit) instruction is executed and by two for a double word (32-bit) instruction. Jump instructions load the P-register with entry address of the program that receives control.

Real-Time Clock (RTC) and Monitor (MON) Clock Features

The AN/AYK-14(V) contains an RTC and a MON clock which are loadable and readable under software control, and provide interrupts when enabled. The RTC counts up a 32-bit register at a 1-MHz rate, allowing for timed intervals up to $(2^{32}-1)$ microseconds or approximately 1.19 hours.

The MON clock is a 16-bit counter which counts down at a 10-KHz rate to provide interrupts at intervals up to approximately 6.5 seconds. These features are useful for scheduling periodic processing activities, coordinating I/O operations, and timing real-time events. The high resolution of the AN/AYK-14(V) clock is particularly useful for signal processing applications and weapons control functions associated with high-speed vehicles.

Power Failure Protection Feature

The power failure protection feature provides for orderly shutdown and preparation for recovery if computer power falls below a safe threshold. The PCM monitors power and, upon detection of a power failure, provides a signal to generate a CPU interrupt. The CPU has about 280 microseconds to store desired registers and status.

Status Registers

The AN/AYK-14(V) contains two 16-bit status registers, status register number 1 (SR1), and status register number 2 (SR2), which provide an indication of the computer state, error conditions, and interrupt states. These registers are

accessible to all programs, but can only be modified by software in the executive mode. Upon program interruption, SR1 and SR2 are automatically stored in memory, where they can be recalled and reinstated upon completion of the interrupt processing routine to allow continuation of the original program with the status existing before interruption.

The functions of the fields in SR1, as shown in Figure 17, are:

- Bit 0 — permits processor to enable/disable DMA.

- Bits 1 to 3 — provide for lockout or enable of interrupts by class.

- Bit 6 — when bit 6=0, the residue in floating point arithmetic operations will not be saved.

- Bit 7 — when bit 7=0, the floating point overflow or underflow interrupt is enabled.

- Bits 9 and 8 — form a condition code and indicate the results of arithmetic and compare instructions as shown in Figure 17.

- Bit 10 — overflow designator which is set when an arithmetic or shift operation produces a result that overflows the register.

- Bit 11 — carry designator is set when an arithmetic operation generates a carry beyond the most significant bit of the register.

- Bit 12 — when bit 12=0, the processor uses NDRO memory.

- Bit 13 — Bit 13=0 designates CPU.

- Bit 14 — Bit 14=0 designates that general register set 0 is to be used (program state).

- Bit 15 — Bit 15=0 designates that the CPU is in executive mode.
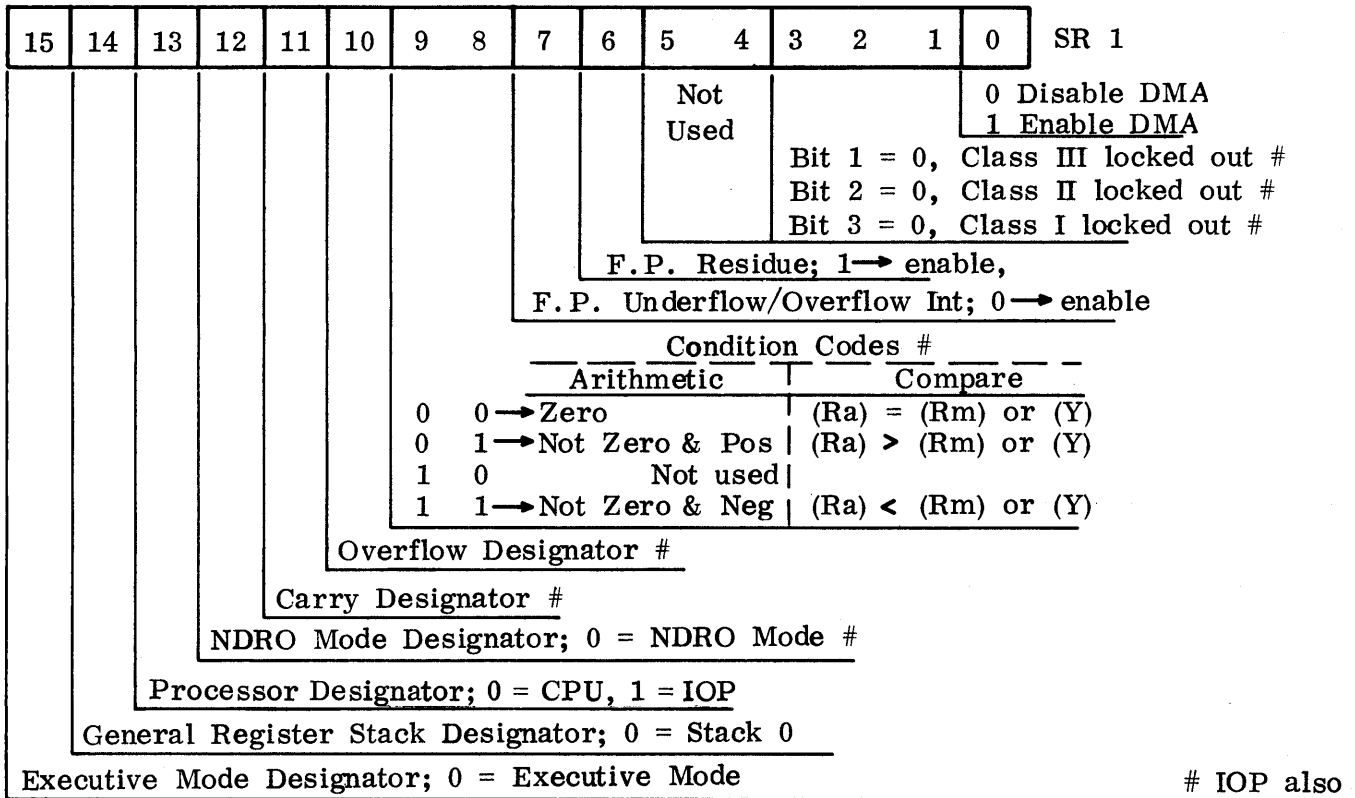
Figure 17. Status Register Number 1 Format

29

The functions of the fields in SR2, as shown in Figure 18, are:

- <u>Bits 0 to 7</u> — provide indication of I/O instruction faults and associated I/O channel number. Also provide memory interrupt data for memory timeout, memory parity error, and protect faults with indication of bank number and whether the bank is on OMEMBUS or EMEMBUS.

    Memory timeout occurs if the memory fails to respond to a memory operation within 2 to 5 usec.

- <u>Bits 8 to 15</u> — this field, in conjunction with the instruction m-field, controls indirect addressing.

Instructions

The AN/AYK-14(V) instruction set contains all the instructions comprising the standard AN/UYK-20 instruction set, plus instructions to enhance performance and flexibility of use. Instruction types in addition to the standard AN/UYK-20 set include types for:

- Floating point arithmetic operations
- Stack and queue processing
- Maintenance and diagnostic functions

In addition, some of the AN/AYK-14(V) instructions are designated as privileged and will result in an interrupt upon attempted execution when not in executive mode.
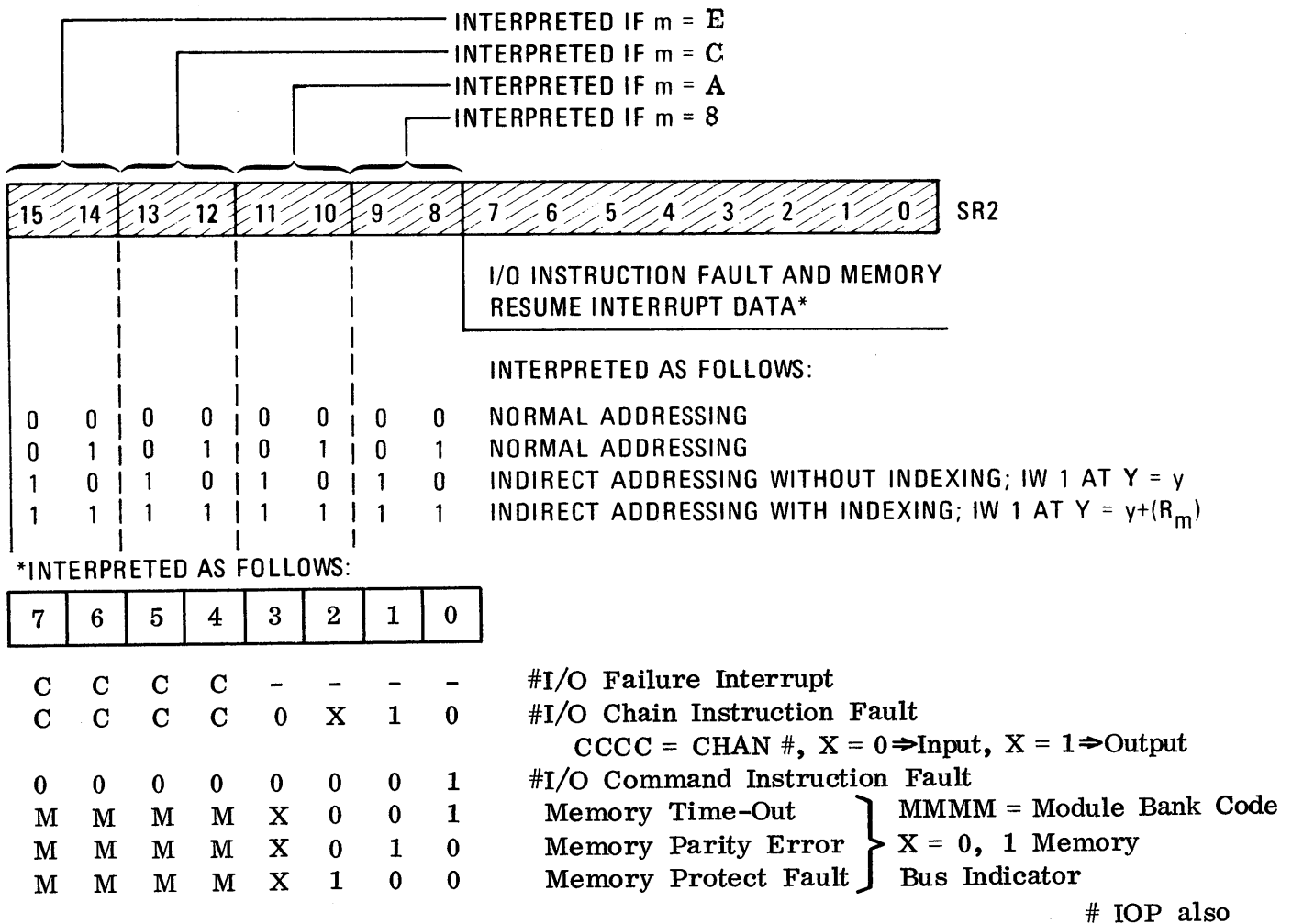
Figure 18. Status Register Number 2 Format

30

Table 7 lists the AN/AYK-14(V) instruction set given operation codes in both octal and hexadecimal notation, the coding format for assembly-level coding, and a brief description of the operation performed. Table 8 defines abbreviations used in Table 7.

The instruction operation codes prefixed with a "#" symbol in Table 7 are instructions which can be executed by the IOP.

The preliminary instruction execution times given are for the CPU in a CPU-only configuration and are based on the following conditions:
- Single GPM
- 900-nanosecond cycle time core memory
- Interleaved addresses
- Overlapped memory access

The instruction execution times for the IOP are 0 to 40 percent longer than for the CPU, depending on instruction types and module configurations.

Instruction Addressing

In general, instruction addresses are determined by the contents of the 16-bit P-register. The 16-bit relative address in the P-register is transformed to a 19-bit memory address by the hardware-controlled process discussed in the Memory Addressing paragraph. Single-word instructions result in incrementing (P) by one and double-word instructions result in incrementing (P) by two. In the use of jump instructions, the P-register is set to the address to which control is to be transferred.

An exception to the use of the P-register for instruction addressing occurs in I/O operations involving chaining. Chain address pointers stored in I/O control memory serve as instruction address counters associated with each I/O channel. Incrementing of the chain address pointer is analogous to the P-register operation. A Load Control Memory instruction can be used to change the address pointer to provide a jump capability.

Instruction Word Formats

The instruction word formats for the AN/AYK-14(V) include both single (16-bit) and double (32-bit) word types. The 16-bit instructions conserve memory and enhance processing speed while the double word provides memory addressing over the full range of addresses and for in-line storage of constants. Figure 19 defines the fields for the four types of instruction formats and Figure 20 defines the associated operand formation. The various instruction types provide for a variety of operand addressing processes including direct, indirect, and indexed types.

Instructions using double length (32-bit) operands use two sequential registers or memory locations as shown in Figure 19. Ra, Rm, or Y contains the most significant portion and sign bit; Ra+1, Rm+1, or Y+1 contains the least significant portion of the operand.

The instruction word formats of the AN/AYK-14(V) are identical to those of the AN/UYK-20:

- RR format instructions use general registers for operands instead of main memory. The a- and m-designators specify general registers Ra and Rm, respectively, that are used in the operation.

- RL format instructions perform operations involving one or two general registers. The a-designator selects Ra or Ra and Ra+1 depending on the particular instruction. The m-designator is a 4-bit unsigned literal, which can be used, for example, as a count or increment depending on the particular instruction.

- RI format, Type 1 instructions are local jump instructions which increase or decrease (P) by the D value in the instruction. The effective jump address Y+(P) + xD, where xD is the 2's complement deviation value.

- RI format, Type 2 instructions perform operations that involve general registers and a main memory reference. The a- and m-designators select general registers Ra and Rm, respectively. Rm in this case contains an address, Y, that is used for the main memory reference.

TABLE 7. AN/AYK-14(V) INSTRUCTION REPERTOIRE

Octal Format	Hexadecimal Format	Coding Format	Instruction	Operation	C	OV	CC	CPU** Time in μsec
#00 2 a m	02 a m	SPT a,y,m	Stack Put Top	(Y) → (Ra), (Ra) → Y		-NA-		8.5
#00 3 a m	03 a m	BL a,y,m	Byte load	(Y) byte → Ra	0	0	X	2.11
#01 0 a m	04 a m	LR a,m	Load (Register)	(Rm) → Ra	0	0	X	.80
#01 1 a m	05 a m	LI a,m	Load (Indirect)	(Y*) → Ra	0	0	X	1.49
#01 2 a m	06 a m	LK a,y,m	Load (Constant)	Y → Ra	0	0	X	1.62
#01 3 a m	07 a m	L a,y,m	Load	(Y) → Ra	0	0	X	1.85
#02 0 a 00	08 a 0	PR a	Make positive	If (Ra) < 0, (Ra)' → Ra	X	X	X	.90
#02 0 a 01	08 a 1	NR a	Make negative	If (Ra) > 0, (Ra)' → Ra	X	0	X	.88
02 0 a 02	08 a 2	RR a	Round	(Ra) + (Ra ⊕ 1):15 → Ra	X	X	X	.90
#02 0 a 03	08 a 3	IPI a	°Initiate Processor Int	Set processor interrupt a		-NA-		1.75
#02 0 a 04	08 a 4	TCR a	Two's Complement	(Ra)' → Ra	X	X	X	.80
02 0 a 05	08 a 5	TCDR a	Two's Complement Double	(Ra, Ra ⊕ 1)' → Ra, Ra ⊕ 1	X	X	X	.93
#02 0 a 06	08 a 6	OCR a	One's Complement	(Ra) bit-by-bit complement → Ra	0	0	X	.80
#02 0 a 07	08 a 7	SCI a	Support Channel Input	Support channel input → Ra	0	0	X	.93
#02 0 a 10	08 a 8	IROR a	Increase Ra by 1	(Ra) + 1 → Ra	X	X	X	.80
#02 0 a 11	08 a 9	DROR a	Decrease Ra by 1	(Ra) - 1 → Ra	X	X	X	.80
#02 0 a 12	08 a A	IRTR a	Increase Ra by 2	(Ra) + 2 → Ra	X	X	X	.80
#02 0 a 13	08 a B	DRTR a	Decrease Ra by 2	(Ra) - 2 → Ra	X	X	X	.80
#02 0 - 14	08 - C	IPLF	°IPL Failed	Set IPLF discrete		-NA-		1.50
#02 0 a 15	08 a D	DJ a	°Diagnostic Jump	R15 → uP		-NA-		VAR.
#02 0 - 16	08 - E	RBT	°Reset Bit Timer	0 → Bit Timer		-NA-		3.75
#02 0 - 17	08 - F	SBT	°Set BIT Indicator	Set BIT indicator		-NA-		4.26
02 1 a m	09 a m	LDI a,m	Load Double (Indirect)	(Y*, Y* ⊕ 1) → Ra, Ra ⊕ 1	0	0	X	2.02
02 3 a m	0B a m	LD a,y,m	Load Double	(Y, Y ⊕ 1) → Ra, Ra ⊕ 1	0	0	X	2.40
#03 0 a 00	0C a 0	ER a	Executive Return	Generate interrupt; P+1 → Ra	0	0	X	.9 – 1.1
#03 0 a 01	0C a 1	SSOR a	Store SR1	(SR1) → Ra	0	0	X	.96
#03 0 a 02	0C a 2	SSTR a	Store SR2	(SR2) → Ra	0	0	X	.80
#03 0 a 03	0C a 3	SCR a	Store Clock	(RTC register):15-0 → Ra	0	0	X	1.25
03 0 a 04	0C a 4	LPR a	Load P	(Ra) → P		-NA-		1.70
#03 0 a 05	0C a 5	LSOR a	°Load SR1	(Ra) → SR1		-NA-		6.02
#03 0 a 06	0C a 6	LSTR a	°Load SR2	(Ra) → SR2		-NA-		.90
#03 0 a 07	0C a 7	LCR a	°Load RTC lower	(Ra) → RTC register:15-0		-NA-		5.40
#03 0 - 10	0C - 8	ECR	°Enable Clock and Interrupt	Enable RTC register and interrupt		-NA-		3.51
#03 0 - 11	0C - 9	DCR	°Disable Clock	Disable RTC register		-NA-		1.5
03 0 a 12	0C a A	LEM a	°Load and Enable Monitor Clock	(Ra) → Monitor clock register; enable countdown		-NA-		5.28
03 0 - 13	0C - B	DM	°Disable Monitor Clock	Disable monitor clock register		-NA-		1.8
03 0 a 14	0C a C	LCRD a	°Load Double and Enable Clock	(Ra, Ra ⊕ 1) → RTC; enable count up		-NA-		6.0
#03 0 a 15	0C a D	SCRD a	Store Clock Double	(RTC register) → Ra, Ra ⊕ 1	0	0	X	2.29
#03 0 - 16	0C - E	ECIR	°Enable Clock Interrupt	Enable RTC overflow interrupt		-NA-		1.62
#03 0 - 17	0C - F	DCIR	°Disable Clock Interrupt	Disable RTC overflow interrupt		-NA-		1.32
#03 1 a m	0D a m	SCIO a,m	Support Channel I/O	(Ra) → support channel buffer m → I/O code, Set channel busy		-NA-		—
03 3 a m	0F a m	LM a,y,m	Load multiple	(Y...Y+m-a) → Ra...Rm		-NA-		2.15+1.2n
04 0 a 01	10 a 1	RVR a	Reverse Register	Reverse (Ra)	0	0	X	7.26
04 0 a 02	10 a 2	CNT a	Count Ones	Number of binary ones in Ra → Ra + 1		-NA-		7.0
04 0 a 03	10 a 3	SFR a	Scale Factor	Shift (Ra, Ra ⊕ 1) left until (Ra):15 ≠ (Ra):14; shift count → Ra ⊕ 1+1 ①		-NA-		3.48 – 8.82
04 0 a 04	10 a 4	SMC a	Store Monitor Clock	(Mon) → Ra		-NA-		1.75
#04 2 a m	12 a m	QPT a,y,m	Queue Put Top	(Y) → (Ra), (Ra) → Y; if (Y) was = 0 then (Ra) → Y ⊕ 1		-NA-		8.29
#04 3 a m	13 a m	BLX a,y,m	Byte Load and Index by 1	(Y) byte → Ra; (Rm)+1 → Rm	0	0	X	2.38
#05 0 a m	14 a m	SBR a,m	Set Bit	1 → (Ra):m	0	0	X	.80
#05 1 a m	15 a m	LXI a,m	Load and Index by 1 (Indirect)	(Y*) → Ra; (Rm)+1 → Rm	0	0	X	1.29
#05 2 a m	16 a m	QPB a,y,m	Queue Put Bottom	(Ra) → (Y ⊕ 1), (Ra) → Y ⊕ 1, 0 → (Ra)		-NA-		10.34
#05 3 a m	17 a m	LX a,y,m	Load and Index by 1	(Y) → Ra; (Rm)+1 → Rm	0	0	X	1.92
#06 0 a m	18 a m	ZBR a,m	Zero Bit	0 → (Ra):m	0	0	X	.80
06 1 a m	19 a m	LDXI a,m	Load Double Index by 2 (Indirect)	(Y*, Y* ⊕ 1) → Ra, Ra ⊕ 1; (Rm)+2 → Rm	0	0	X	2.25
#06 2 a m	1A a m	SGT a,y,m	Stack Get Top	(Y) → Ra, if (Y) ≠ 0 then ((Y)) → Y and P+3 → P, if (Y) = 0 then P+2 → P		-NA-		9.80
06 3 a m	1B a m	LDX a,y,m	Load Double, Index by 2	(Y, Y ⊕ 1) → Ra, Ra ⊕ 1; (Rm)+2 → Rm	0	0	X	2.42
#07 0 a m	1C a m	CBR a,m	Compare Bit	Compare bit m of Ra with zero	0	0	X	.80
#07 1 - m	1D - m	LPI m	°Load PSW (Indirect)	(Y*, Y*+1, Y*+2) → P, SR1, SR2		-NA-		7.92
#07 2 a m	1E a m	QGT a,y,m	Queue Get Top	(Y) → Ra; if (Y) = 0 then P+2 → P; if (Y) ≠ 0 then P+3 → P, ((Y)) → Y; if ((Y)) = 0 then Y → Y ⊕ 1		-NA-		9.80
#07 3 - m	1F - m	LP y,m	°Load PSW	(Y, Y+1, Y+2) → P, SR1, SR2		-NA-		7.78
#10 0 a m	20 a m	LRSR a,m	Logical Right Shift (Register)	Shift (Ra) right (Rm):5-0 places, zero fill	0	0	X	2.28+.21s
#10 2 a m	22 a m	LRS a,y,m	Logical Right Shift	Shift (Ra) right Y:5-0 places, zero fill	0	0	X	3.10+.21s
#10 3 a m	23 a m	BS a,y,m	Byte Store	(Ra):7-0 → Y byte		-NA-		2.32
#11 0 a m	24 a m	ARSR a,m	Algebraic Right Shift (Register)	Shift (Ra) right (Rm):5-0 places, sign fill	0	0	X	2.28+.21s
#11 1 a m	25 a m	SI a,m	Store (Indirect)	(Ra) → Y*		-NA-		1.41
#11 2 a m	26 a m	ARS a,y,m	Algebraic Right Shift	Shift (Ra) right Y:5-0 places, sign fill	0	0	X	3.06+.21s

\# IOP Instructions   ° Executive Mode Instructions   ① Count = 31 for all zeros or all ones

**With 900 ns core memory, interleaved, No IOP or EAU, Preliminary data

TABLE 7. AN/AYK-14(V) INSTRUCTION REPERTOIRE (CONT.)

Octal Format	Hexadecimal Format	Coding Format	Instruction	Operation	C	OV	CC	CPU** Time in μsec.
#11 3 a m	27 a m	S a,y,m	Store	(Ra) → Y		-NA-		1.91
12 0 a m	28 a m	LRDR a,m	Logical Right Double Shift (Register)	Shift (Ra, Ra ⊕ 1) right (Rm):5–0 places, zero fill	0	0	X	2.61
12 1 a m	29 a m	SDI a,m	Store Double (Indirect)	(Ra, Ra⊕1) →Y*, Y* ⊕ 1		-NA-		2.06
12 2 a m	2A a m	LRD a,y,m	Logical Right Double Shift	Shift (Ra, Ra ⊕ 1) right Y:5–0 places, zero fill	0	0	X	3.39+.21s
12 3 a m	2B a m	SD a,y,m	Store Double	(Ra, Ra ⊕ 1) → Y, Y ⊕ 1		-NA-		2.4
13 0 a m	2C a m	ARDR a,m	Algebraic Right Double Shift (Register)	Shift (Ra, Ra ⊕ 1) right (Rm):5–0 places, sign fill	0	0	X	2.61+.21s
13 2 a m	2E a m	ARD a,y,m	Algebraic Right Double Shift	Shift (Ra, Ra ⊕ 1) right Y:5–0 places, sign fill	0	0	X	3.42+.21s
13 3 a m	2F a m	SM a,y,m	Store Multiple	(Ra...Rm) → Y...Y+m–a		-NA-		2.50+1.1n
#14 0 a m	30 a m	ALSR a,m	Algebraic Left Shift (Register)	Shift (Ra) left (Rm):5–0 places, zero fill	0	X	X	2.64+.21s
#14 2 a m	32 a m	ALS a,y,m	Algebraic Left Shift	Shift (Ra) left Y:5–0 places, zero fill	0	X	X	3.25+.21s
#14 3 a m	33 a m	BSX a,y,m	Byte Store, Index by 1	(Ra):7–0 → Y byte; (Rm)+1 → Rm		-NA-		2.52
#15 0 a m	34 a m	CLSR a,m	Circular Left Shift (Register)	Shift (Ra) circularly left (Rm):5–0 places	0	0	X	2.46
#15 1 a m	35 a m	SXI a,m	Store, Index by 1 (Indirect)	(Ra) → Y*; (Rm)+1 → Rm		-NA-		1.15
#15 2 a m	36 a m	CLS a,y,m	Circular Left Shift	Shift (Ra) circularly left Y:5–0 places	0	0	X	3.26
#15 3 a m	37 a m	SX a,y,m	Store, Index by 1	(Ra) → Y; (Rm)+1 → Rm		-NA-		2.13
16 0 a m	38 a m	ALDR a,m	Algebraic Left Double Shift (Register)	Shift (Ra, Ra ⊕ 1) left (Rm):5–0 places, zero fill	0	X	X	3.10
16 1 a m	39 a m	SDXI a,m	Store Double, Index by 2 (Indirect)	(Ra, Ra ⊕ 1) → Y*, Y* ⊕ 1; (Rm)+2 → Rm		-NA-		2.12
16 2 a m	3A a m	ALD a,y,m	Algebraic Left Double Shift	Shift (Ra, Ra ⊕ 1) left Y:5–0 places, zero fill	0	X	X	3.86+.21s
16 3 a m	3B a m	SDX a,y,m	Store Double, Index by 2	(Ra, Ra ⊕ 1) → Y, Y ⊕ 1 ; (Rm)+2 → Rm		-NA-		2.63
17 0 a m	3C a m	CLDR a,m	Circular Left Double Shift (Register)	Shift (Ra, Ra ⊕ 1) circularly left (Rm):5–0 places	0	0	X	2.61+.21s
17 1 – m	3D – m	SZI m	Store Zeros (Indirect)	0 → Y*		-NA-		1.91
17 2 a m	3E a m	CLD a,y,m	Circular Left Double Shift	Shift (Ra, Ra⊕1) circularly left Y:5–0 places	0	0	X	3.39+.21s
17 3 – m	3F – m	SZ y,m	Store Zeros	0 → Y		-NA-		1.91
#20 0 a m	40 a m	SUR a,m	Subtract (Register)	(Ra) – (Rm) → Ra	X	X	X	.80
#20 1 a m	41 a m	SUI a,m	Subtract (Indirect)	(Ra) – (Y*) → Ra	X	X	X	1.30
#20 2 a m	42 a m	SUK a,y,m	Subtract (Constant)	(Ra) – Y → Ra	X	X	X	1.60
#20 3 a m	43 a m	SU a,y,m	Subtract	(Ra) – (Y) → Ra	X	X	X	1.83
21 0 a m	44 a m	SUDR a,m	Subtract Double (Register)	(Ra, Ra ⊕ 1) – (Rm, Rm ⊕ 1) → Ra, Ra ⊕ 1	X	X	X	1.11
21 1 a m	45 a m	SUDI a,m	Subtract Double (Indirect)	(Ra, Ra ⊕ 1) – (Y*, Y* ⊕ 1) → Ra, Ra ⊕ 1	X	X	X	2.15
21 3 a m	47 a m	SUD a,y,m	Subtract Double	(Ra, Ra ⊕ 1) – (Y, Y ⊕ 1) → Ra, Ra ⊕ 1	X	X	X	2.4
#22 0 a m	48 a m	AR a,m	Add (Register)	(Ra) + (Rm) → Ra	X	X	X	.80
#22 1 a m	49 a m	AI a,m	Add (Indirect)	(Ra) + (Y*) → Ra	X	X	X	1.30
#22 2 a m	4A a m	AK a,y,m	Add (Constant)	(Ra) + Y → Ra	X	X	X	1.60
#22 3 a m	4B a m	A a,y,m	Add	(Ra) + (Y) → Ra	X	X	X	1.83
23 0 a m	4C a m	ADR a,m	Add Double (Register)	(Ra, Ra ⊕ 1) + (Rm, Rm ⊕ 1) → Ra, Ra ⊕ 1	X	X	X	1.11
23 1 a m	4D a m	ADI a,m	Add Double (Indirect)	(Ra, Ra ⊕ 1) + (Y*, Y* ⊕ 1) → Ra, Ra ⊕ 1	X	X	X	2.15
23 3 a m	4F a m	AD a,y,m	Add Double	(Ra, Ra ⊕ 1) + (Y, Y ⊕ 1) → Ra, Ra ⊕ 1	X	X	X	2.40
#24 0 a m	50 a m	CR a,m	Compare (Register)	(Ra) – (Rm)	X	X	X	.80
#24 1 a m	51 a m	CI a,m	Compare (Indirect)	(Ra) – (Y*)	X	X	X	1.30
#24 2 a m	52 a m	CK a,y,m	Compare (Constant)	(Ra) – Y	X	X	X	1.60
#24 3 a m	53 a m	C a,y,m	Compare	(Ra) – (Y)	X	X	X	1.89
25 0 a m	54 a m	CDR a,m	Compare Double (Register)	(Ra, Ra ⊕ 1) – (Rm, Rm ⊕ 1)	X	X	X	1.11
25 1 a m	55 a m	CDI a,m	Compare Double (Indirect)	(Ra, Ra ⊕ 1) – (Y*, Y* ⊕ 1)	X	X	X	2.15
25 3 a m	57 a m	CD a,y,m	Compare Double	(Ra, Ra ⊕ 1) – (Y, Y ⊕ 1)	X	X	X	2.40
#26 0 a m	58 a m	MR a,m	Multiply (Register)	(Ra⊕1) · (Rm) → Ra, Ra ⊕ 1	0	0	X	4.14
#26 1 a m	59 a m	MI a,m	Multiply (Indirect)	(Ra⊕1) · (Y*) → Ra, Ra ⊕ 1	0	0	X	4.50
#26 2 a m	5A a m	MK a,y,m	Multiply (Constant)	(Ra⊕1) · Y → Ra, Ra ⊕ 1	0	0	X	4.94
#26 3 a m	5B a m	M a,y,m	Multiply	(Ra⊕1) · (Y) → Ra, Ra ⊕ 1	0	0	X	4.98
#27 0 a m	5C a m	DR a,m	Divide (Register)	(Ra, Ra ⊕ 1)/(Rm) → Ra ⊕ 1; remainder → Ra	0	X	X	8.40
#27 1 a m	5D a m	DI a,m	Divide (Indirect)	(Ra, Ra ⊕ 1)/(Y*) → Ra ⊕ 1; remainder → Ra	0	X	X	9.03
#27 2 a m	5E a m	DK a,y,m	Divide (Constant)	(Ra, Ra ⊕ 1)/Y → Ra ⊕ 1; remainder → Ra	0	X	X	9.50
#27 3 a m	5F a m	D a,y,m	Divide	(Ra, Ra ⊕ 1)/(Y) → Ra ⊕ 1; remainder → Ra	0	X	X	10.66
#30 0 a m	60 a m	ANDR a,m	AND (Register)	(Ra) ⊙ (Rm) → Ra	0	0	X	.80
#30 1 a m	61 a m	ANDI a,m	AND (Indirect)	(Ra) ⊙ (Y*) → Ra	0	0	X	1.30
#30 2 a m	62 a m	ANDK a,y,m	AND (Constant)	(Ra) ⊙ Y → Ra	0	0	X	1.60
#30 3 a m	63 a m	AND a,y,m	AND	(Ra) ⊙ (Y) → Ra	0	0	X	1.83
#31 0 a m	64 a m	ORR a,m	OR (Register)	(Ra) ⊕ (Rm) → Ra	0	0	X	.80
#31 1 a m	65 a m	ORI a,m	OR (Indirect)	(Ra) ⊕ (Y*) → Ra	0	0	X	1.30

IOP Instructions

33

TABLE 7. AN/AYK-14(V) INSTRUCTION REPERTOIRE (CONT.)

Octal Format	Hexadecimal Format	Coding Format	Instruction	Operation	C	OV	CC	CPU** Time in μsec.
#31 2 a m	66 a m	ORK a,y,m	OR (Constant)	(Ra) ⊕ Y → Ra	0	0	X	1.60
#31 3 a m	67 a m	OR a,y,m	OR	(Ra) ⊕ (Y) → Ra	0	0	X	1.83
32 0 a m	68 a m	XORR a,m	Exclusive OR (Register)	(Ra) ⊕ (Rm) → Ra	0	0	X	.80
32 1 a m	69 a m	XORI a,m	Exclusive OR (Indirect)	(Ra) ⊕ (Y*) → Ra	0	0	X	1.30
32 2 a m	6A a m	XORK a,y,m	Exclusive OR (Constant)	(Ra) ⊕ Y → Ra	0	0	X	1.60
32 3 a m	6B a m	XOR a,y,m	Exclusive OR	(Ra) ⊕ (Y) → Ra	0	0	X	1.83
33 0 a m	6C a m	MSR a,m	Masked Substitute (Register)	If (Ra ⊕ 1):n = 1; (Rm):n → Ra:n	0	0	X	1.08
33 1 a m	6D a m	MSI a,m	Masked Substitute (Indirect)	If (Ra ⊕ 1):n = 1; (Y*):n → Ra:n	0	0	X	1.73
33 2 a m	6E a m	MSK a,y,m	Masked Substitute (Constant)	If (Ra ⊕ 1):n = 1; Y:n → Ra:n	0	0	X	1.88
33 3 a m	6F a m	MS a,y,m	Masked Substitute	If (Ra ⊕ 1):n = 1; (Y):n → Ra:n	0	0	X	2.28
#34 0 a m	70 a m	CMR a,m	Compare Masked (Register)	[(Ra) ⊙ (Ra ⊕ 1)]: [(Rm) ⊙ (Ra ⊕ 1)]	0	0	X	1.08
#34 1 a m	71 a m	CMI a,m	Compare Masked (Indirect)	[(Ra) ⊙ (Ra ⊕ 1)]: [(Y*) ⊙ (Ra ⊕ 1)]	0	0	X	1.73
#34 2 a m	72 a m	CMK a,y,m	Compare Masked (Constant)	[(Ra) ⊙ (Ra ⊕ 1)]: [Y ⊙ (Ra ⊕ 1)]	0	0	X	1.88
#34 3 a m	73 a m	CM a,y,m	Compare Masked	[(Ra) ⊙ (Ra ⊕ 1)]: [(Y) ⊙ (Ra ⊕ 1)]	0	0	X	2.28
#35 0 - -	74 - -	IOCR	°Input/Output Command	Execute (C Cell); 0 → C Cell:15-14		-NA-		—
35 1 - m	75 - m	BFI m	Biased Fetch (Indirect)	(Y*):15 → CC; 1 → (Y*):15,14	0	0	X	5.60
35 2 - m	76 - m	REX y,m	Remote Execute	Execute (Y); P + 2 → P unless jump		-NA-		VAR.
35 3 - m	77 - m	BF y,m	Biased Fetch	(Y):15 → CC; 1 → (Y):15,14	0	0	X	6.10
#40 0 00 m	80 0 m	JER m	Jump Equal	If CC indicates = or 0; (Rm) → P		-NA-		.9, 1.75
#40 0 01 m	80 1 m	JNER m	Jump Not Equal	If CC indicates ≠ or not 0; (Rm) → P		-NA-		.9, 1.75
#40 0 02 m	80 2 m	JGER m	Jump Greater or Equal	If CC indicates ≥ or +; (Rm) → P		-NA-		.9, 1.75
#40 0 03 m	80 3 m	JLSR m	Jump Less	If CC indicates < or -; (Rm) → P		-NA-		.9, 1.75
#40 0 04 m	80 4 m	JOR m	Jump Overflow	If overflow set; (Rm) → P		-NA-		.9, 1.75
#40 0 05 m	80 5 m	JCR m	Jump Carry	If carry set; (Rm) → P		-NA-		.9, 1.75
#40 0 06 m	80 6 m	JPTR m	Jump Power out of Tolerance	If power out of tolerance: (Rm) → P		-NA-		.9
40 0 07 m	80 7 m	JBR m	Jump Bootstrap 2 Selected	If bootstrap 2 selected: (Rm) → P		-NA-		.9, 1.75
#40 0 10 m	80 8 m	JR m	Jump	(Rm) → P		-NA-		1.75
#40 0 11 m	80 9 m	JSR m	°Jump after Stop	Stop; (Rm) → P		-NA-		2.28
40 0 12 m	80 A m	JKSR 1,m	°Jump. If Key Set - Stop, then Jump (Register)	If key 1 set, stop; (Rm) → P		-NA-		2.28
40 0 13 m	80 B m	JKSR 2,m	°Jump. If Key Set - Stop, then Jump (Register)	If key 2 set, stop; (Rm) → P		-NA-		2.28
#40 0 14 m	80 C m	JSCR m	Jump Support Channel Busy - RR	If support channel busy; (Rm) → P		-NA-		.9 – 1.11
#40 1 d	81 d	LJ xD	Local Jump	P + xD → P		-NA-		1.65
#40 2 00 m	82 0 m	JE y,m	Jump Equal	If CC indicates = or 0; Y → P		-NA-		1.75, 1.95
#40 2 01 m	82 1 m	JNE y,m	Jump Not Equal	If CC indicates ≠ or not 0; Y → P		-NA-		1.75, 1.95
#40 2 02 m	82 2 m	JGE y,m	Jump Greater than or Equal	If CC indicates ≥ or +; Y → P		-NA-		1.75, 1.95
#40 2 03 m	82 3 m	JLS y,m	Jump Less	If CC indicates < or -; Y → P		-NA-		1.75, 1.95
#40 2 04 m	82 4 m	JO y,m	Jump on Overflow	If overflow set; Y → P		-NA-		1.75, 1.95
#40 2 05 m	82 5 m	JC y,m	Jump on Carry	If carry set, Y → P		-NA-		1.75, 1.95
40 2 06 m	82 6 m	JPT y,m	Jump if Power out of Tolerance	If power out of tolerance; Y → P		-NA-		1.75
40 2 07 m	82 7 m	JB y,m	Jump if Bootstrap 2 Selected	If bootstrap 2 selected; Y → P		-NA-		1.75, 1.95
#40 2 10 m	82 8 m	J y,m	Jump	Y → P		-NA-		1.75
#40 2 11 m	82 9 m	JS y,m	°Jump after Stop	Stop; Y → P		-NA-		1.7
40 2 12 m	82 A m	JKS 1,y,m	°Jump. If Key Set - Stop, then Jump	If key 1 set, stop; Y → P		-NA-		2.30
40 2 13 m	82 B m	JKS 2,y,m	°Jump. If Key Set - Stop, then Jump	If key 2 set, stop; Y → P		-NA-		2.30
40 2 14 m	82 C m	JSC y,m	Jump Support Channel Busy - RK	If support channel busy; Y → P		-NA-		1.75, 1.95
#40 3 00 m	83 0 m	JE *y,m	Jump Equal	If CC indicates = or 0; (Y) → P		-NA-		1.75, 2.54
#40 3 01 m	83 1 m	JNE *y,m	Jump Not Equal	If CC indicates ≠ or not 0; (Y) → P		-NA-		1.75, 2.54
#40 3 02 m	83 2 m	JGE *y,m	Jump Greater or Equal	If CC indicates ≥ or +; (Y) → P		-NA-		1.75, 2.54
#40 3 03 m	83 3 m	JLS *y,m	Jump Less	If CC indicates < or -; (Y) → P		-NA-		1.75, 2.54
#40 3 04 m	83 4 m	JO *y,m	Jump on Overflow	If overflow set; (Y) → P		-NA-		1.75, 2.54
#40 3 05 m	83 5 m	JC *y,m	Jump on Carry	If carry set; (Y) → P		-NA-		1.75, 2.54
40 3 06 m	83 6 m	JPT *y,m	Jump if Power out of Tolerance	If power out of tolerance; (Y) → P		-NA-		1.75
40 3 07 m	83 7 m	JB *y,m	Jump if Bootstrap 2 Selected	If bootstrap 2 selected; (Y) → P		-NA-		1.75, 2.54
#40 3 10 m	83 8 m	J *y,m	Jump	(Y) → P		-NA-		2.48
#40 3 11 m	83 9 m	JS *y,m	°Jump after Stop	Stop; (Y) → P		-NA-		1.9
40 3 12 m	83 A m	JKS 1,*y,m	°Jump. If Key Set - Stop, then Jump	If key 1 set, stop; (Y) → P		-NA-		3.24
40 3 13 m	83 B m	JKS 2,*y,m	°Jump. If Key Set - Stop, then Jump	If key 2 set, stop; (Y) → P		-NA-		3.24
#40 3 14 m	83 C m	JSC *y,m	Jump Support Channel Busy - RX	If support channel busy; (Y) → P		-NA-		1.75, 2.54
#41 0 a m	84 a m	XJR a,m	Index Jump Register	If (Ra) ≠ 0; (Ra) - 1 → Ra, (Rm) → P		-NA-		.9, 1.75
#41 1 d	85 d	LJI xD	Local Jump (Indirect)	[P + xD] → P		-NA-		2.46
#41 2 a m	86 a m	XJ a,y,m	Index Jump	If (Ra) ≠ 0; (Ra) - 1 → Ra; Y → P		-NA-		1.75, 1.95
#41 3 a m	87 a m	XJ a,*y,m	Index Jump	If (Ra) ≠ 0; (Ra) - 1 → Ra; (Y) → P		-NA-		1.75, 2.54
#42 0 a m	88 a m	JLRR a,m	Jump, Link Register (Register)	(P) + 1 → Ra; (Rm) → P		-NA-		1.86
#42 2 a m	8A a m	JLR a,y,m	Jump, Link Register	(P) + 2 → Ra; Y → P		-NA-		1.86
#42 3 a m	8B a m	JLR a,*y,m	Jump, Link Register	(P) + 2 → Ra; (Y) → P		-NA-		2.69
#43 1 d	8D d	LJLM xD	Local Jump, Link Memory	(P) + 1 → P + xD; P + xD + 1 → P		-NA-		1.92
#43 2 - m	8E - m	JLM y,m	Jump, Link Memory	(P) + 2 → Y; Y + 1 → P		-NA-		2.1
#43 3 - m	8F - m	JLM *y,m	Jump, Link Memory	(P) + 2 → (Y): (Y) + 1 → P		-NA-		3.28

IOP Instructions ° Executive Mode Instructions

TABLE 7. AN/AYK-14(V) INSTRUCTION REPERTOIRE (CONT.)

Octal Format	Hexadecimal Format	Coding Format	Instruction	Operation	C	OV	CC	CPU** Time in μsec.
#44 0 a m	90 a m	JZR a,m	Jump Zero (Register)	If (Ra) = 0; (Rm) → P	-NA-			.9, 1.75
#44 1 d	91 d	LJE xD	Local Jump Equal	If CC indicates = or 0; (P) + xD → P	-NA-			.9, 1.75
#44 2 a m	92 a m	JZ a,y,m	Jump Zero	If (Ra) = 0; Y → P	-NA-			1.75, 1.95
#44 3 a m	93 a m	JZ a,*y,m	Jump Zero	If (Ra) = 0; (Y) → P	-NA-			1.75, 2.54
#45 0 a m	94 a m	JNZR a,m	Jump Not Zero (Register)	If (Ra) ≠ 0; (Rm) → P	-NA-			.9, 1.75
#45 1 d	95 d	LJNE xD	Local Jump Not Equal	If CC indicates ≠ or not 0; (P) + xD → P	-NA-			.9, 1.75
#45 2 a m	96 a m	JNZ a,y,m	Jump Not Zero	If (Ra) ≠ 0; Y → P	-NA-			1.75, 1.95
#45 3 a m	97 a m	JNZ a,*y,m	Jump Not Zero	If (Ra) ≠ 0; (Y) → P	-NA-			1.75, 2.54
#46 0 a m	98 a m	JPR a,m	Jump Positive (Register)	If (Ra) ≥ 0; (Rm) → P	-NA-			.9, 1.75
#46 1 d	99 d	LJGE xD	Local Jump Greater or Equal	If CC indicates ≥ or +; (P) + xD → P	-NA-			.9, 1.75
#46 2 a m	9A a m	JP a,y,m	Jump Positive	If (Ra) ≥ 0; Y → P	-NA-			1.75, 1.95
#46 3 a m	9B a m	JP a,*y,m	Jump Positive	If (Ra) ≥ 0; (Y) → P	-NA-			1.75, 2.54
#47 0 a m	9C a m	JNR a,m	Jump Negative (Register)	If (Ra) < 0; (Rm) → P	-NA-			.9, 1.75
#47 1 d	9D d	LJLS xD	Local Jump Less	If CC indicates < or -; (P) + xD → P	-NA-			.9, 1.75
#47 2 a m	9E a m	JN a,y,m	Jump Negative	If (Ra) < 0; Y → P	-NA-			1.75, 1.95
#47 3 a m	9F a m	JN a,*y,m	Jump Negative	If (Ra) < 0; (Y) → P	-NA-			1.75, 2.54
50 0 a m	A0 a m	FSUR a,m	Floating Point Subtract (Register)	(Ra, Ra ⊕ 1) - (Rm, Rm ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
50 1 a m	A1 a m	FSUI a,m	Floating Point Subtract (Indirect)	(Ra, Ra ⊕ 1) - (Y*, Y* ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
50 3 a m	A3 a m	FSU a,y,m	Floating Point Subtract	(Ra, Ra ⊕ 1) - (Y, Y ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
51 0 a m	A4 a m	FAR a,m	Floating Point Add (Register)	(Ra, Ra ⊕ 1) + (Rm, Rm ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
51 1 a m	A5 a m	FAI a,m	Floating Point Add (Indirect)	(Ra, Ra ⊕ 1) + (Y*, Y* ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
51 3 a m	A7 a m	FA a,y,m	Floating Point Add	(Ra, Ra ⊕ 1) + (Y, Y ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
52 0 a m	A8 a m	FMR a,m	Floating Point Multiply (Register)	(Ra, Ra ⊕ 1) · (Rm, Rm ⊕ 1) → Ra, Ra+1, Ra+2, Ra+3; Res. → Ra+2, Ra+3		X	X	
52 1 a m	A9 a m	FMI a,m	Floating Point Multiply (Indirect)	(Ra, Ra ⊕ 1) · (Y*, Y* ⊕ 1) → Ra, Ra+1, Ra+2, Ra+3; Res. → Ra+2, Ra+3		X	X	
52 3 a m	AB a m	FM a,y,m	Floating Point Multiply	(Ra, Ra ⊕ 1) · (Y, Y ⊕ 1) → Ra, Ra+1, Ra+2, Ra+3; Res. → Ra+2, Ra+3		X	X	
53 0 a m	AC a m	FDR a,m	Floating Point Divide (Register)	(Ra, Ra ⊕ 1)/(Rm, Rm ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
53 1 a m	AD a m	FDI a,m	Floating Point Divide (Indirect)	(Ra, Ra ⊕ 1)/(Y*, Y* ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
53 3 a m	AF a m	FD a,y,m	Floating Point Divide	(Ra, Ra ⊕ 1)/(Y, Y ⊕ 1) → Ra, Ra+1; Res. → Ra+2, Ra+3		X	X	
54 0 a m	B0 a m	°Load Address Register (Register) LARR a,m	°Load Address Register (Register)	(Rm) → ARr	-NA-			3.16
54 1 a m	B1 a m	LARI a,m	°Load Address Register (Indirect)	(Y*) → ARr	-NA-			3.57
54 3 a m	B3 a m	LARM a,y,m	°Load Address Register Multiple	(Y,....Y + u) → ARr....ARr+u	-NA-			7.01+2.0n
55 0 a m	B4 a m	SARR a,m	Store Address Register (Register)	(ARr) → Rm	-NA-			1.82
55 1 a m	B5 a m	SARI a,m	Store Address Register (Indirect)	(ARr) → Y*	-NA-			2.96
55 3 a m	B7 a m	SARM a,y,m	Store Address Register Multiple	(ARr...ARr+u) → Y,...Y+u	-NA-			5.03+1.8n
56 0 a m	B8 a m	MDR a,m	Multiply Double (Register)	(Ra, Ra ⊕ 1) · (Rm, Rm ⊕ 1) → Ra, Ra+1, Ra+2, Ra+3	0	0	X	8.06
56 1 a m	B9 a m	MDI a,m	Multiply Double (Indirect)	(Ra, Ra ⊕ 1) · (Y*, Y* ⊕ 1) → Ra, Ra+1, Ra+2, Ra+3	0	0	X	8.56
56 3 a m	BB a m	MD a,y,m	Multiply Double	(Ra, Ra ⊕ 1) · (Y, Y ⊕ 1) → Ra, Ra+1, Ra+2, Ra+3	0	0	X	8.31
57 0 a m	BC a m	DDR a,m	Divide Double (Register)	(Ra, Ra+1, Ra+2, Ra+3)/(Rm, Rm ⊕ 1) → Ra+2, Ra+3; Rem. → Ra, Ra+1	0	X	X	44.76
57 1 a m	BD a m	DDI a,m	Divide Double (Indirect)	(Ra, Ra+1, Ra+2, Ra+3)/(Y*, Y* ⊕ 1) → Ra+2, Ra+3; Rem. → Ra, Ra+1	0	X	X	44.96
57 3 a m	BF a m	DD a,y,m	Divide Double	(Ra, Ra+1, Ra+2, Ra+3)/(Y, Y ⊕ 1) → Ra+2, Ra+3; Rem. → Ra, Ra+1	0	X	X	45.60
#60 0 a m	C0 a m	LLRS a,m	Literal Logical Right Shift	Shift (Ra) right m places, zero fill	0	0	X	1.32+.21s
#60 1 a m	C1 a m	LARS a,m	Literal Algebraic Right Shift	Shift (Ra) right m places, sign fill	0	0	X	1.32+.21
60 2 a m	C2 a m	LLRD a,m	Literal Logical Right Double Shift	Shift (Ra, Ra ⊕ 1) right m places, zero fill	0	0	X	1.47+.21s
60 3 a m	C3 a m	LARD a,m	Literal Algebraic Right Double Shift	Shift (Ra, Ra ⊕ 1) right m places, sign fill	0	0	X	1.47+.21s
#61 0 a m	C4 a m	LALS a,m	Literal Algebraic Left Shift	Shift (Ra) left m places, zero fill	0	X	X	1.68+.21s
#61 1 a m	C5 a m	LCLS a,m	Literal Circular Left Shift	Shift (Ra) left circular m places	0	0	X	1.32+.21s
61 2 a m	C6 a m	LALD a,m	Literal Algebraic Left Double Shift	Shift (Ra, Ra ⊕ 1) left m places, zero fill	0	X	X	2.12+.21s
61 3 a m	C7 a m	LCLD a,m	Literal Circular Left Double Shift	Shift (Ra, Ra ⊕ 1) left circular m places	0	0	X	1.47+.21s
#62 0 a m	C8 a m	LSU a,m	Literal Subtract	(Ra) - m → Ra	X	X	X	.80
62 1 a m	C9 a m	LSUD a,m	Literal Subtract Double	(Ra, Ra ⊕ 1) - m → Ra, Ra ⊕ 1	X	X	X	1.11
#62 2 a m	CA a m	LA a,m	Literal Add	(Ra) + m → Ra	X	X	X	.80
62 3 a m	CB a m	LAD a,m	Literal Add Double	(Ra, Ra ⊕ 1) + m → Ra, Ra ⊕ 1	X	X	X	1.11
#63 0 a m	CC a m	LL a,m	Literal Load	m → Ra	0	0	X	.80
#63 1 a m	CD a m	LC a,m	Literal Compare	(Ra) - m	X	X	X	.80

# IOP Instructions   ° Executive Mode Instructions

TABLE 7. AN/AYK-14(V) INSTRUCTION REPERTOIRE (CONT.)

Octal Format	Hexadecimal Format	Coding Format	Instruction	Operation	C	OV	CC	CPU** Time in μsec.
63 2 a m	CE a m	LMUL a,m	Literal Multiply	(Ra ⊕ 1) · m →Ra, Ra ⊕ 1	0	0	X	4.15
63 3 a m	CF a m	LDIV a,m	Literal Divide	(Ra, Ra ⊕ 1)/m →Ra ⊕ 1; remainder →Ra	0	X	X	8.53
64 3 a m	D3 a m	BSU a,y,m	Byte Subtract	(Ra) - (Y) byte →Ra	X	X	X	2.07
65 3 a m	D7 a m	BA a,y,m	Byte Add	(Ra) + (Y) byte →Ra	X	X	X	2.07
66 3 a m	DB a m	BC a,y,m	Byte Compare	(Ra) - (Y) byte	X	X	X	2.07
67 3 a m	DF a m	BCX a,y,m	Byte Compare and Index by 1	(Ra) - (Y) byte; (Rm) + 1 →Rm	X	X	X	2.25

COMMAND/CHAIN INSTRUCTION

Octal Format	Hexadecimal Format	Coding Format	Instruction	Operation	C	OV	CC	CPU** Time in μsec.
#70 0 00 00	E0 0 0	ACR	Channel Control	Master clear all channels				6.7/7.8
#70 0 00 04	E0 0 4	ACR 4 CCR 0,4	Channel Control	Enable external interrupts, all channels				6.7/7.8
#70 0 00 05	E0 0 5	ACR 5 CCR 0,5	Channel Control	Disable external interrupts, all channels				6.7/7.8
#70 0 00 06	E0 0 6	ACR 6 CCR 0,6	Channel Control	Enable Class III Interrupts, Priorities 2, 3, 4				6.7/7.8
#70 0 00 07	E0 0 7	ACR 7 CCR 0,7	Channel Control	Disable Class III Interrupts, Priorities 2, 3, 4				6.7/7.8
#70 0 a 10	E0 a 8	CCR a,8	Channel Control	Master clear channel a				6.7/7.8
#70 0 a 14	E0 a C	CCR a,12	Channel Control	Enable channel a external interrupts				6.7/7.8
#70 0 a 15	E0 a D	CCR a,13	Channel Control	Disable channel a external interrupts				6.7/7.8
#70 0 a 16	E0 a E	CCR a,14	Channel Control	Enable channel a, Class III, priorities 2, 3, 4				6.7/7.8
#70 0 a 17	E0 a F	CCR a,15	Channel Control	Disable channel a, Class III, priorities 2, 3, 4				6.7/7.8

COMMAND INSTRUCTION

Octal Format	Hexadecimal Format	Coding Format	Instruction	Operation	C	OV	CC	CPU** Time in μsec.
#71 2 a 02	E6 a 2	ICK a,y	Initiate Input Chain	Y → Channel a Chain Pointer; initiate input chain				5.7
#71 2 a 06	E6 a 6	OCK a,y	Initiate Output Chain	Y → Channel a Chain Pointer; initiate output chain				5.7
#71 3 a m	E7 a m	WIM a,y,m	Write Control Memory	(Y) → Channel a CMm				6.6
#72 3 a m	EB a m	RIM a,y,m	Read Control Memory	Channel a (CMm) →Y				6.6
#76 0 a m	F8 a m	SICR a,m	Set and Clear Discretes	Set or clear channel a discrete function				5.3
#76 3 a m	FB a m	SST a,y,m	Store Status	Channel a Status bits per m →Y				5.3
#77 0 - m	FC - m	SIOP m,y	Start IOP ③	m:0 →IOP SR1:12, Y →IOP P if m = 0 or 1				VAR.
#77 2 a m	FE a m	XIM a,y,m	Exchange Control Memory ②	Channel a (CMm) →Y; (Y ⊕ 1) →Channel a CMm				7.5

CHAIN INSTRUCTION

Octal Format	Hexadecimal Format	Coding Format	Instruction	Operation	C	OV	CC	CPU** Time in μsec.
#70 2 a m	E2 a m	IM a,y,m	Initiate Message	Y → CMm; Initiate message activity				6.7
#70 3 a 00	E3 a 0	IO a,y	IO Function a	(Y, Y ⊕ 1) → BCW, BAP; initiate transfer				7.6
#71 3 00 m	E6 0 m	LCMK m,y	Load Control Memory	Y → CMm				6.7
#71 3 00 m	E7 0 m	LCM m,y	Load Control Memory	(Y) → CMm				7.6
#72 3 00 m	EB 0 m	SCM m,y	Store Control Memory	(CMm) → Y				7.6
#73 0 00 00	EC 0 0	HCR	Halt Chain	Halt chaining, a even				5.4
#73 0 01 00	EC 1 0	IPR	Interrupt Processor	Generate chain interrupt, a odd				5.4
#73 3 00 00	EF 0 0	ZF y	Zero Flag	0 →Y15,14, a even				7.6
#73 3 01 00	EF 1 0	SF y	Set Flag	1 → Y15,14, a odd				7.6
#74 2 00 00	F2 0 0	SJMC 0,y	Serial Jump on Met Condition	Unconditional Y → CAP; clear flag				5.4
#74 2 01 00	F2 1 0	SJMC 1,y	Serial Jump on Met Condition	If suppress flag not set, Y → CAP; clear flag				5.4
#74 2 02 00	F2 2 0	SJMC 2,y	Serial Jump on Met Condition	If monitor flag set, Y→CAP; clear flag				8.7
#75 0 00 m	F4 0 m	SFSC m	Search For Sync	Perform function(s) assigned to m-bits				
#76 0 00 m	F8 0 m	CSIR m	Serial Interface Control	Set or clear discrete function				5.4
#76 3 - m	FB - m	CSST y,m	Store Status	Status bits per m → Y				7.6
#77 1 - m	FD - m	BJ m,y	Bit Jump	(Y) → CAP if (CM3):m = 1				6.7, 7.6
#77 2 - m	FE - m	XCM m,y	Exchange Control Memory	(CMm) →Y; (Y ⊕ 1) → CMm				8.5

IOP Instructions   ② m = 2 or 6   ③ no operation unless IOP

TABLE 8. LEGEND FOR INSTRUCTION REPERTOIRE

B	Byte pointer, 0 → Upper, 1 → Lower	BAP	Buffer Address Pointer
C	Carry	CM	Control Memory Word
CC	Condition Code	CAP	Chain Address Pointer
OV	Overflow	RTC	Real-Time Clock
IW	Indirect Word	( )	Contents of register or address
J	Designator Field in IW	r	$(R_a)_{5-0}$
x	General Register Designator in IW1	u	$(R_a)_{13-8}$
y	Contents of Second Instruction Word or IW2	.	Multiply
Y	Effective Operand Address or Constant	⊕	Or
Y*	Effective Operand Address in Rm	⊕	Exclusive or
TM	I/O Transfer Mode	⊙	And
	00 → Abort Input Transfer	SR1	Status Register No. 1
	01 → 8-bit Byte Transfer	SR2	Status Register No. 2
	10 → 16-bit Word Transfer	:	Bit number follows
	11 → 32-bit Dual Word Transfer	/	2's Complement
BWC	Buffer Word Count	C Cell	I/O Command Cell
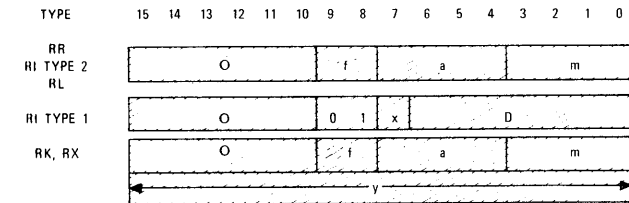
DEFINITION OF FIELDS

( )   Operation (Function) Code
f   Format Designator
a   General Register or Subfunction Designator
m   General Register or Subfunction Designator
    4-bit Unsigned Literal Constant in RL Format
xD   Signed Deviation Value (Two's Complement)
y   Address or Arithmetic Constant

Figure 19. Instruction Word Format

- RK format instructions are double-word instructions stored in sequential memory locations. The first word contains the operation code and designator fields. The second word is a value, Y, that may be used as a constant operand or address or as a modified constant or address. The a-designator selects general register Ra. When m=0, the operand or address Y equals y, no Rm is selected. m≠0 selects a general register Rm; the operand Y=y+(Rm) (i.e., y is indexed by the contents of Rm). Also operand Y is used as an address in RK format jump instructions and in remote execute instructions.

- RX format instructions are also double-word instructions. The first word contains the a- and m-designators and the next word contains the y-value. RX format instructions perform 8-bit byte, whole-word (16-bit), and double-word (32-bit) operations with general registers and main memory. The a-designator selects Ra for all three types of operands.

RX instructions provide very flexible operand addressing including direct, indirect, cascaded indirect, and indexed types. The m-designator selects the addressing as follows:

m=0 — direct addressing without indexing
m-1-7,9,B,D,or F (hex) — direct addressing with indexing, with Rm used as the modifier.
m = 8,A,C,E (hex) — address mode depends on bits set in $SR2_{8-15}$ as shown in Figure 8. If indirect addressing is indicated a pair of indirect words (IW1 and IW2) will be used to define further operand address processing according to the scheme depicted in Figure 21. The address of IW1 is a Y=y if not indexed and Y-y+(Rm) if indexed. This process provides a means for cascading indirect addresses as desired.

37

FORMAT	OPERAND FORMATION
RR	Operand = (R_m)
RI-1	Local Jump Address Y = (P) + xD
RI-2	Operand at Y* = (R_m)
RK	Operand Y = y + (R_m) if m ≠ 0
	Operand Y = y if m = 0
#RX Word	Operand at Y = y if m = 0
	Operand at Y = y + (R_m) if m ≠ 0, 8, A, C, E
	Operand at indirect address if m = 8, A, C, E
#RX Byte	Operand at Y upper if m = 0
	Operand at Y = $(R_m)/2$ + y if m ≠ 0, 8, A, C, E
	B = $(R_m)_0$
	Operand at indirect address if m = 8, A, C, E
RL	Operand = m (an absolute literal)

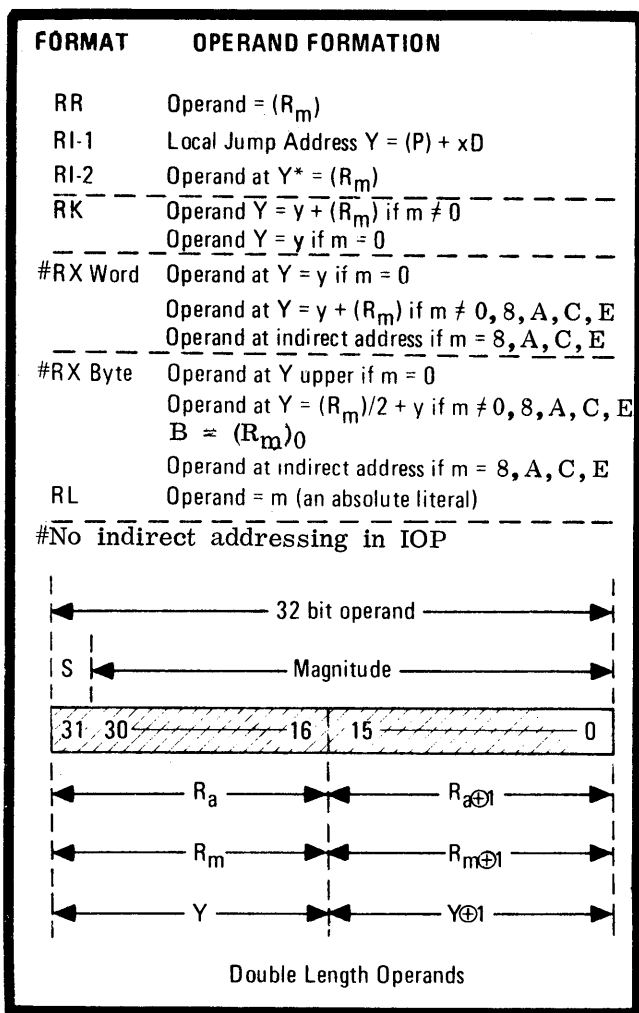#No indirect addressing in IOP

Double Length Operands

Figure 20. Operand Formation

Double length operands are assigned even numbered addresses or registers, with the most significant portion in the even numbered location and the least significant portion in the next location. Memory addressing for the first portion of a double length operation is formed like that of the single word operands.

Byte operand addressing uses a byte identifier (B) to select the upper or lower 8-bit portion of the 16-bit word. B=0 designates the most significant half of the word in address Y as the operand, B=1 designates the lower half.

The least significant bit of the indexing register is used as the byte identifier and the value in the remaining bits is used as the index to generate the effective address as illustrated in the following examples:

    m=0, address Y=y and B=0
    m=1-7,9,B,D, or F (hex)

$$Y = y + \frac{(Rm)}{(2)} \text{ and } B = LSB \text{ of } (Rm)$$

when indirect addressing is used and

    if j=0, y=(IW2) and B=0

    if j=1, $Y = (IW2) + \frac{(Rx)}{2}$ and B=LSB of (Rx)

    if j=2, $Y = (IW2) + \frac{(Rm)}{2}$ and B=LSB of (Rm)

    if j=3, $Y = (IW2) + \frac{(Rm+1)}{2}$ and

        B=LSB of (Rm+1)

Floating Point Operands

Floating point addition, subtraction, multiplication, and division may be performed with a normalized result with or without a residue. The format is defined by Figure 22.

Word 1 of the operand contains the algebraic sign of the fractional mantissa, a biased characteristic in the range $0 \leq C \leq 7F$ (hex) and the two most significant hexadecimal digits of the fractional mantissa. A normalized floating point number has a non-zero hexadecimal digit in the most significant four bits of the mantissa. When a residue is requested by the program, the computer stores the floating point number in Ra and Ra+1 and the residue (lower order bits) in Ra+2 and Ra+3 in floating-point data format (i.e., the same format as shown in Figure 22).

A change of 1 in the characteristic represents a factor of 16 change in the value of the number and represents a 4-bit position shift of the fractional mantissa. The magnitude range of floating point numbers is approximately $5.4 \times 10^{-79} \leq M \leq 7.2 \times 10^{75}$. A zero quantity is represented by a positive sign (0), a zero characteristic and a zero mantissa.
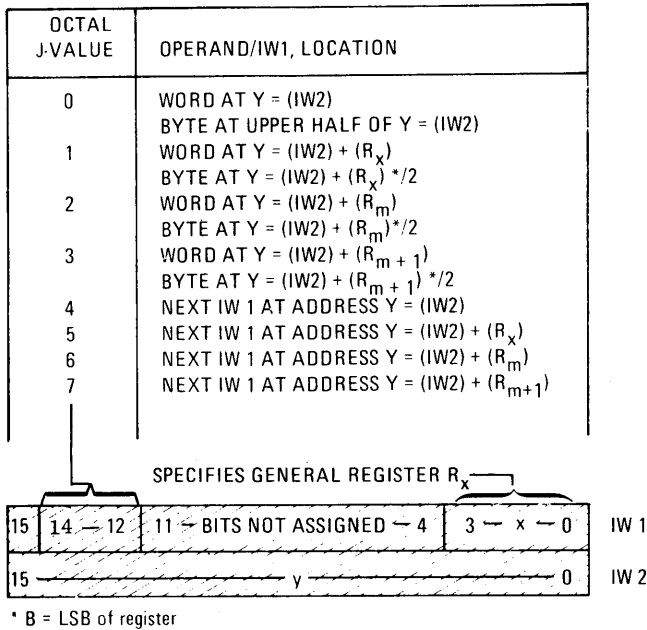
38

OCTAL J-VALUE	OPERAND/IW1, LOCATION
0	WORD AT Y = (IW2)
	BYTE AT UPPER HALF OF Y = (IW2)
1	WORD AT Y = (IW2) + (R_x)
	BYTE AT Y = (IW2) + (R_x) */2
2	WORD AT Y = (IW2) + (R_m)
	BYTE AT Y = (IW2) + (R_m)*/2
3	WORD AT Y = (IW2) + (R_{m+1})
	BYTE AT Y = (IW2) + (R_{m+1}) */2
4	NEXT IW 1 AT ADDRESS Y = (IW2)
5	NEXT IW 1 AT ADDRESS Y = (IW2) + (R_x)
6	NEXT IW 1 AT ADDRESS Y = (IW2) + (R_m)
7	NEXT IW 1 AT ADDRESS Y = (IW2) + (R_{m+1})

SPECIFIES GENERAL REGISTER R_x

| 15 | 14 — 12 | 11 — BITS NOT ASSIGNED — 4 | 3 — x — 0 | IW 1 |
| 15 | v | 0 | IW 2 |

* B = LSB of register

Figure 21. Indirect Addressing

Interrupts

The AN/AYK-14(V) CPU and IOP can be interrupted in their program execution either by events within the computer or by interrupt requests from peripheral I/O devices. There are three major priority levels or classes of interrupts as defined in Table 9.

Interrupts within each class are assigned a priority within the class. The interrupt code is generated by hardware logic in the CPU (or IOP) to vector

	32 BIT OPERAND	
	(R_a), (R_m) or (Y)	(R_{a+1}), (R_{m+1}) or (Y+1)
	WORD 1	WORD 2
15	14 —— 8	7 —— 0	15 —————— 0
S	CHAR.	FRACTIONAL MANTISSA

PRESUMED RADIX POINT

a,m and address Y are even numbers

Figure 22. Floating Point Format

the program, upon interrupt, to the memory location containing the associated interrupt processing routine. Highest priority is assigned to lowest numbered classes and numbers within the class. As each interrupt is recognized, all classes of equal or lower priority can be locked out by the reloaded status register until released by the interrupt processing routine. This allows a higher priority class interrupt to temporarily suspend processing of lower priority interrupts. The higher priority interrupt will be processed to completion and then the suspended interrupt processing will be resumed.

RTC and MON clock registers of the CPU can be loaded, read, enabled, or disabled under program control. When enabled, the RTC counts at a 1-MHz rate. When the register lower order 16 bits overflow, the CPU (or IOP) generates an RTC overflow interrupt and control is transferred to the appropriate processing routine. The RTC continues to count until disabled by the Disable RTC instruction. The RTC and MON clock do not count when the computer is stopped. Upon restart the count resumes if it was running prior to stop. The MON clock function is enabled by the load and Enable MON Clock instruction (0CaA) which also loads the register with starting count. When the MON clock counts down to zero, the MON clock interrupt is generated, the MON clock disabled, and control transferred to the MON clock interrupt routine. The MON clock can be disabled by programming a Disable MON Clock instruction (0C-B).

A memory time-out interrupt is generated when the memory fails to acknowledge a request within 2 to 5 microseconds. If Class I interrupts are disabled, the interrupt is lost.

Instruction fault interrupts are generated when the computer attempts to execute an instruction that is not assigned to the repertoire. The "not assigned" group and those with HEX code E0 through FF, when addressed by the P-register, generate a CP instruction fault. The "not assigned" group and those with HEX code other than E0 through FF, when addressed by an I/O chain address pointer or CP command, generate an I/O instruction fault interrupt.

39
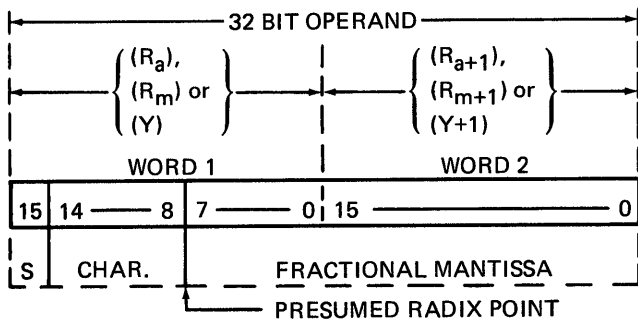
TABLE 9. INTERRUPT PRIORITY

Class	Priority Within Class	Interrupt	Binary Interrupt Code Generated
Class I	1	Power Fault#	00000
	2	Memory Time-Out#	00010
	3	Memory Parity#	00100
	4	Hardware Fault Warning#	00110
	5	I/O Failure#	01010
	6	Thermal Overload#	01100
	7	Hardware Fault#	01110
Class II	1	CP Instruction Fault#	00000
	2	I/O Instruction Fault#	00010
	3	Floating Point Overflow/Underflow	00100
	4	Executive Return#	00110
	5	Executive Mode Fault	10000
	6	Protect Fault	11000
	7	RTC Overflow#	01000
	8	Monitor Clock	01010
	9	System Reset#	10110
	10	Processor Interrupt 0#	01100
	11	Processor Interrupt 1#	01110
Class III	1	I/O Channel Abnormal	110
	2	External#	000
	3	Output Chain#	100
	4	Input Chain#	010

IOP Interrupts

External devices that have the capability of interrupting the computer (such as those communicating via NTDS channels) can cause an interrupt ie, Class III priority and "external" interrupt. The peripheral device attempts to interrupt the computer by sending a coded message via the I/O channel. The coded message is stored in an assigned memory location (see Table 6) for that channel. If the program has also enabled the external interrupt monitor a Class III, priority 2 interrupt suspends the CPU program and transfers control to the associated interrupt processing routine. The program can disable or enable either or both of these functions. A disabled monitor prevents interrupt generation but allows storage of the external interrupt data.

Interrupt Processing

When an interrupt is honored, the CPU (or IOP) hardware enters the following interrupt processing sequence:

1) Terminates the current program sequence and locks out all interrupts.

2) Stores P, SR1, SR2, and RTC in assigned memory locations (Table 6).

3) Reloads SR1 and SR2 from assigned memory locations (Table 6). P is reloaded with the contents of the interrupt entrance address index appropriate to the interrupt class (Figure 23) plus the contents of the assigned memory address (Table 6). Interrupt lockouts and their release are controlled by program via the Load Status Register instruction or the Load PSW instructions.

4) Enables interrupts which are not locked out by the new contents of SR1.

5) Executes the instruction at address P and continues the program from that point.

15 ——————— 5	4	3	2	1	0
Zeros					

Interrupt Code Per Table 9

Class I & II Interrupts

15 ——————— 7	6	5	4	3	2	1	0
Zeros							

Channel Number   Interrupt Code Per Table 9 3 Low Order Bits
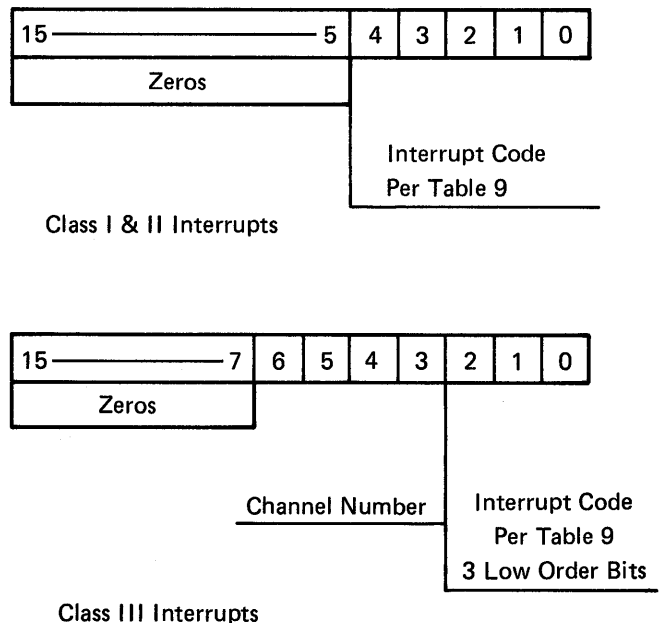
Class III Interrupts

Figure 23. Interrupt Entrance Address Index

I/O Channel Structure

The I/O channel structure of the AN/AYK-14 computer provides for communications between the computer and peripheral equipment and/or other AN/AYK-14 computers. Each of up to 16 I/O channels in a given computer configuration may be assigned a unique 4-bit channel logical number which the software uses to address the given channel. A second unique 4-bit channel priority number is defined for each I/O channel in a given configuration by the physical I/O channel module location for use in resolving possible conflicts among the several I/O channels. Available I/O channel types include the following:

- Parallel NTDS Slow (-15V)
- Parallel NTDS Fast (-3V)
- Parallel NTDS ANEW (+3.5V)
- Serial NTDS (10 MHz)
- Serial 1553A (1 MHz)
- Serial PROTEUS (10 MHz)
- Serial RS-232-C Asynchronous (Selectable Baud)
- Discrete

Any assortment of these I/O channel types may be intermixed in a given AN/AYK-14(V) computer configuration, subject to chassis connector limitations. The channel structure is sufficiently general so that new channel types can be defined without modifying the architecture or the rest of the computer hardware modules.

I/O Channel Programs and Chaining

I/O channel operation is initiated by execution of a processor instruction defined as INPUT/OUTPUT COMMAND (Op Code 74 hexadecimal, mnemonic IOCR). This processor instruction, when encountered in the software sequence during program execution, causes the I/O channel instruction located at address 0060 hexadecimal in main memory to be executed as depicted in Figure 24. I/O channel instructions (Op Codes E0 through FF, hexadecimal) are otherwise illegal when encountered in an AN/AYK-14 processor software program execution, unless encountered in this "out-of-sequence" manner via the IOCR processor instruction.
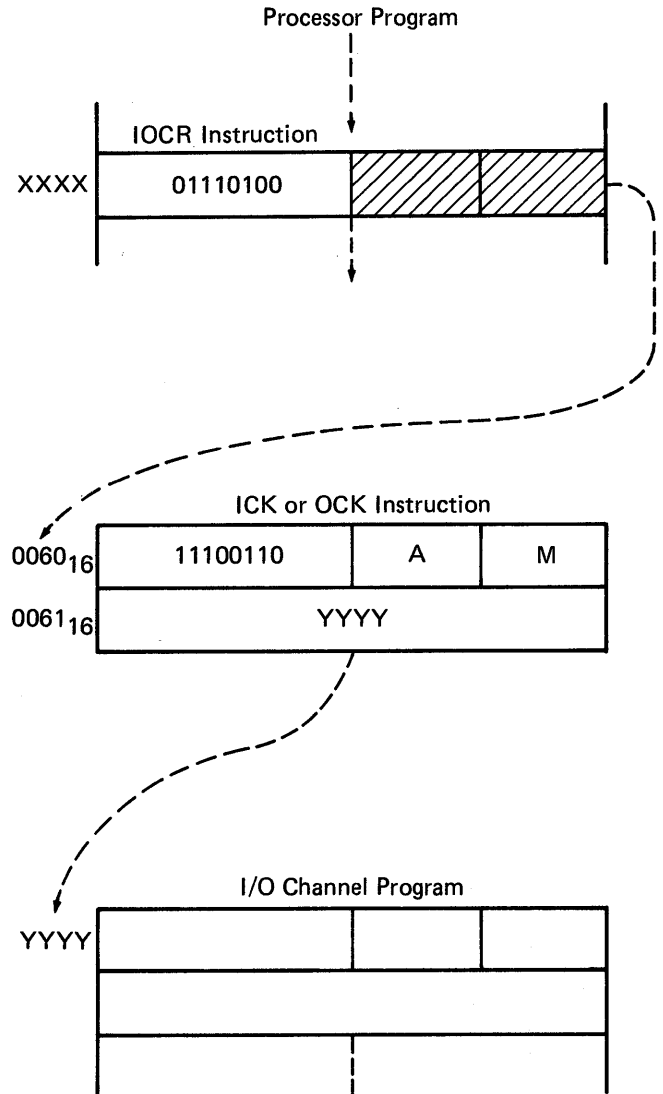
Figure 24. I/O Channel Program Initiation

After initiation by the processor, further I/O channel operations on the affected channel are controlled by I/O channel instructions in an I/O channel program in main memory. Once initiated by the processor, execution of the I/O channel program on the active I/O channel or channels requires no further intervention by the processor software but operates independently of it. I/O channel instructions are sequenced by the contents of the chain address pointer in the associated I/O channel control memory. It should be noted that several I/O channels may be simultaneously active in execution of I/O channel programs.

41

When an I/O channel is active (has an I/O channel program in execution), two forms of I/O channel activity may be in progress:

- I/O information transfer activity
- I/O chaining activity

Under certain conditions, various I/O channel types may have both forms of I/O channel activity in progress at the same time.

I/O chaining activity exists on an I/O channel when sequential I/O channel instructions are being executed on that channel. These I/O channel instructions normally provide for transferring parameters between the main memory and the I/O channel control memory, and for initiation of transfer of blocks or buffers of data and/or control words on the channel interface lines. When an I/O channel instruction that initiates a buffer of words on the interface lines is encountered, I/O chaining activity is halted and I/O information transfer activity begins. Upon completion of I/O information transfer activity (buffer termination), I/O chaining activity is re-initiated. I/O channel instructions also exist to provide branching or jumping within the I/O channel program and halting of I/O channel operations. Certain I/O channel types are able to execute two I/O channel programs at the same time, one which performs I/O information input transfers and the other which performs I/O information output transfers.

I/O information input transfers are subject to any existing main memory page write protection defined by the CPU processor program.

I/O Control Memory

Each I/O channel has associated with it a 12-word by 16-bit control memory which contains the parameters which direct and control the operation of the associated I/O channel. Chain addresses, data buffer addresses, data buffer word counts, data word length definitions, channel mode definitions, and parity bit polarity control are examples of the type of information contained in the control

memory parameters. The format and definition of each location in a given I/O channel control memory are dependent on the given I/O channel type. Figure 25 shows the control memory definition for a SIM channel, which is typical.

I/O CONTROL MEMORY LOCATION

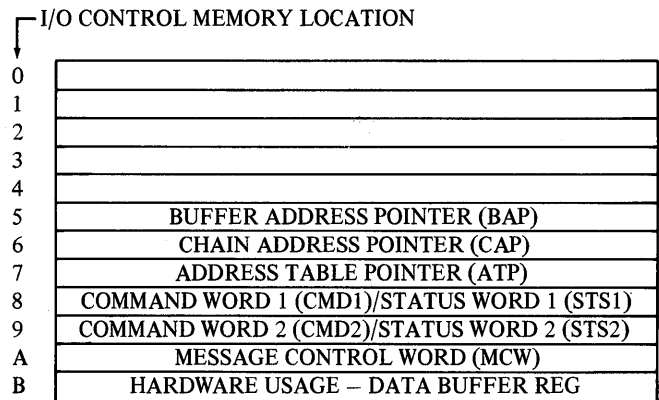|	
0	
1	
2	
3	
4	
5	BUFFER ADDRESS POINTER (BAP)
6	CHAIN ADDRESS POINTER (CAP)
7	ADDRESS TABLE POINTER (ATP)
8	COMMAND WORD 1 (CMD1)/STATUS WORD 1 (STS1)
9	COMMAND WORD 2 (CMD2)/STATUS WORD 2 (STS2)
A	MESSAGE CONTROL WORD (MCW)
B	HARDWARE USAGE – DATA BUFFER REG

Figure 25. SIM Control Memory Definition

Control memory location codes C, D, E, and F hexadecimal are illegal and should not be used. Location B hexadecimal is reserved and used as a one-word output buffer register by all I/O channel types.

The buffer address pointer (BAP) contains the memory address for the input or output of the next data word and is incremented by one for each data word transferred. The chain address pointer (CAP) contains the memory address of the next I/O channel instruction. The CAP is updated during I/O chaining activity in a manner analogous to the processor P-register. The message control word (MCW) defines the mode and control of the SIM channel and the command/status words are used in processing channel protocol and maintaining channel status.

For other channel types, different definitions of control memory words are assigned to accommodate channel characteristics, but the overall concept is similar. For example, channels such as NTDS with simultaneous input and output capability use an input BAP and CAP and output BAP and CAP. Brief descriptions of operation of specific channel types are given in the following paragraphs.

The SIM I/O channel provides the following modes of operation:

- Bus controller (BC) — controls and initiates all data transfer on the 1553A bus.
- Remote terminal (RT) — Responds to BC requests for transmission and reception.
- Bus monitor (BM) — monitors bus for activity and stores every word when received on the bus.
- OFF — no channel activity.
- Self Test (ST) — is a subset of "OFF" in which an internal wraparound test is performed. No data is transmitted or received on the bus.

The 1553A word formats and message formats are given in Figures 7 and 8 respectively. When in the BC or RT mode the channel can communicate via:

- BC to RT data transfer
- RT to BC data transfer
- RT to RT data transfer
- BC to RT mode command transfer

The SIM channel operation meets the requirements of MIL-STD-1553A. The channel structure provides for up to 32 users per bus. Messages on the bus start with one or two command words from the BC, followed by a status word from the RT and up to 32 data words as shown in Figure 8.

Figure 25 defines the SIM control memory assignments. The BAP and CAP are used for all channel types to point to data word address in memory and the address of the next I/O channel instruction, respectively.

The address table pointer (ATP) is used in the RT mode to point to an address in memory of a list of addresses corresponding to one of 31 input and 31 output possible message types. Control memory locations 8 and 9 contains the message COMMAND WORD/STATUS WORD 1 and 2 respectively for use in controlling message sequences and reporting message status.

The MCW defines the mode and control of the SIM (Figure 26). Note that the MCW permits selection of either the A or B bus interfacing with the SIM. Automatic switchover can be enabled while in the RT mode to permit switchover to the redundant bus upon detection of activity on the non-used bus.

Figure 26. SIM I/O Channel Message Control Word
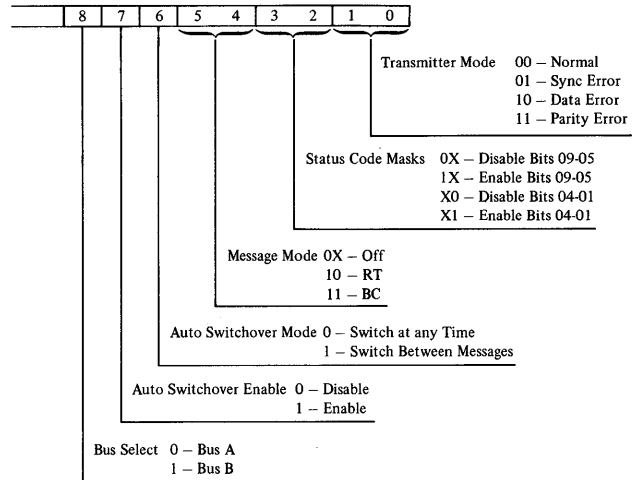
SIM I/O channel status words 0 and 1 are defined in Figures 27 and 28 and contain fields designating current channel status. These I/O channel status words control channel program operation in a manner analogous to status words for the main CPU program. The SIM channel can have ERI (abnormal) interrupts, EII interrupts, and OCI interrupts.
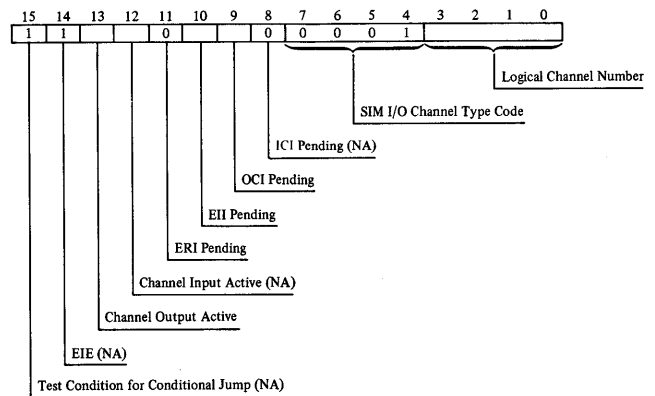

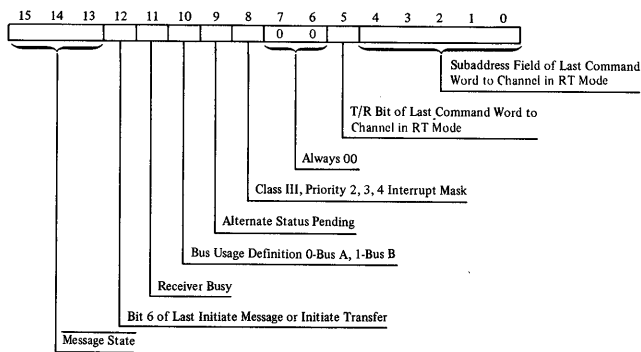Figure 27. SIM I/O Channel Status Word 0

Figure 28. SIM I/O Channel Status Word 1

Discrete Interface Channel Operation

The discrete I/O channel module (DIM) provides the following I/O interfaces:

- Eight external interrupts with individual mask bits and program selectable priority. These appear to the software as class III, EII interrupts.
- 32 Bi-directional discretes (DIO) program selectable in groups of four as input or output signals.
- 16 input discretes (DID).
- 16 "switch closure" input discretes (DIS).

The DID and DIS inputs are accessible only through a jumper cable from the top of the DIM, and are available only when the DIM is inserted in special SRA locations in certain AN/AYK-14(V) chassis configurations.

While the DIM communicates with external equipment via single lines, the software interface to the DIM is via 16-bit parallel words. If a given output bit is desired to be set, an "image word" in main memory would be set and the entire word output to the DIM. For inputs, an entire 16-bit word is read into memory from the DIM and an individual bit then tested. This feature permits use of DIM discretes in a parallel channel mode.

Input and output transfers between DIM and main memory are initiated via an Initiate Message (E2 hex) or Initiate Transfer (E3 hex) instruction executed in an I/O channel program. Either input or output (but not both) chaining can be active at any given time.

Figure 29 defines the control memory for the DIM I/O channel type. Locations 0, 1, and 2 are associated with input activity; locations 4, 5, and 6 are associated with output activity; locations 8 and 9 define the priorities and mask bits for the eight interrupt signals, and location A (hex) defines the enable/disable state of the DIO signal drivers.
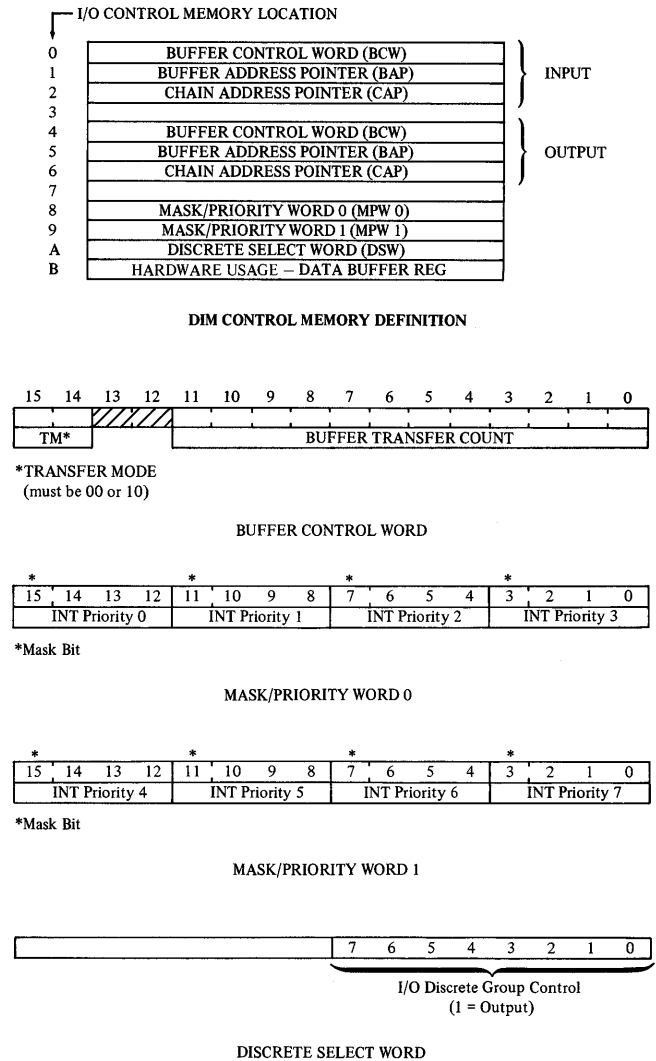

Figure 29. DIM Control Memory Definition and Formats

Figure 29 also defines the formats of those control memory words with definitions unique to DIM. Locations 0 and 4 are termed "buffer control words" and define the word count for input and output data transfer respectively. The maximum

44

useful count for input is 4 (64 input discretes) and for outputs is 2 (32 output discretes). The mask/priority words (MPW 0 and MPW 1) contain eight 4-bit groups with each group corresponding to an interrupt priority level. Within each group the most significant bit is a mask bit and the other three are a binary code for INT:0 through INT:7 to be selected at that priority level. A logic 1 in a given bit position in the discrete select word enables the transmitters for the associated group of 4 DIO signals and selects that group as outputs rather than inputs.

DIM channel status word formats are defined in Figure 30.

The DIM channel is capable of generating EII, OCI, and ICI type interrupts. No ERI type interrupts are generated by the DIM.
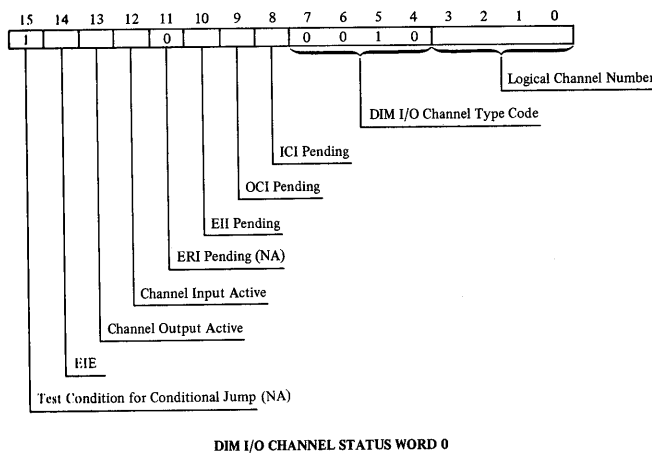

DIM I/O CHANNEL STATUS WORD 0

DIM I/O CHANNEL STATUS WORD 1

Figure 30. DIM I/O Channel Status Words

RS-232-C Channel Operation

An EIA-STD-RS-232-C serial channel communicates over a serial interface which transfers data and control information in both directions using the input and output cable configuration in Figure 11. Full-duplex operation at rates to 9600 baud is possible. The control lines are turned "on" and "off" by I/O command and chaining instructions to communicate with peripheral equipment. The peripheral equipment can, in turn, set control lines to transfer interrupt, responses to controls, and status information to the computer. While the RS-232-C channel module is capable of either synchronous or asynchronous operation, the chassis configuration (connector wiring) permits only a single type of operation on a given channel.

The RS-232-C control memory is defined by Figure 31. Separate input and output control words provide for simultaneously active input and output chain programs. In the monitor mode, the monitor word (location 8) is compared to each character as received. Upon a match, the character is stored in memory and the monitor flag is set. The suppress word (location 9) is used to compare to received characters to prevent storage of these suppressed character in memory. Figure 31 shows the format of the serial mode information and refers to Table 10 for channel speed selection.
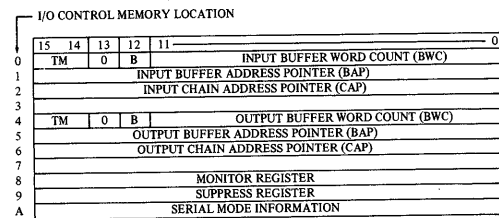
RS-232-C CONTROL MEMORY DEFINITION
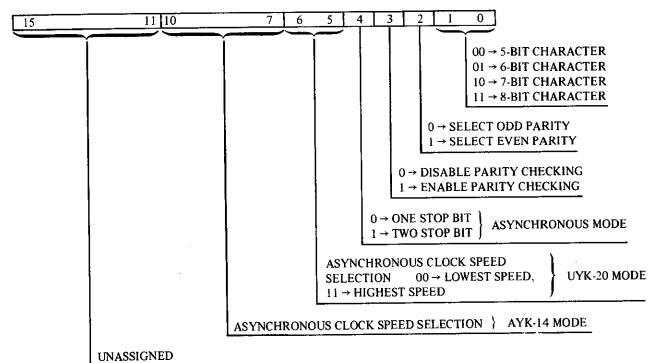
RS-232-C CONTROL MEMORY USAGE

Figure 31. RS-232-C Control Memory Usage

45

The AN/AYK-14(V) provides program selection of any available speed shown in the table.

In the AN/UYK-20 mode any four of 10 possible baud rates may be selected via jumpers in the I/O cable connector. Bits 5 and 6 provide for program selection from among the four selectable speeds.

TABLE 10. RS-232-C CHANNEL SPEED SELECTION FOR ASYNCHRONOUS OPERATION

AYK-14 PROGRAMMABLE BAUD RATE					
SERIAL MODE BITS					
10	9	8	7	BAUD RATE	UYK-20 AVAILABLE BAUD RATES
0	0	0	0	–	
0	0	0	1	–	75
0	0	1	0	50	110
0	0	1	1	75	150
0	1	0	0	134.5	300
0	1	0	1	200	600
0	1	1	0	600	1200
0	1	1	1	2400	1800
1	0	0	0	9600	2400
1	0	0	1	4800	4800
1	0	1	0	1800	9600
1	0	1	1	1200	
1	1	0	0	2400	
1	1	0	1	300	
1	1	1	0	150	
1	1	1	1	110	

The RS-232-C channel programs use the two channel status words defined in Figure 32. The channel is capable of EII, OCI, and ICI interrupts. Also shown is the external interrupt word format. When the interrupt is generated, this word is stored in main memory at the assigned external interrupt word for that channel.

NTDS Parallel Channel Operation

The three NTDS parallel channel types, Slow (Type A), Fast (Type B), and ANEW (Type C), have identical operating characteristics from the software viewpoint and differ only in interface electrical signal characteristics and data rates. The NIM-type's A, B, and C (parallel) I/O channels provide the following modes of operation:

- Computer to peripheral – 16-bit
- Computer to computer – 16-bit

- Test mode
- Computer to peripheral – 32-bit
- Computer to computer – 32-bit
- Externally specified addressing – 32-bit

RS-232-C I/O CHANNEL STATUS WORD 0 FORMAT

RS-232-C I/O CHANNEL STATUS WORD 1 FORMAT

RS-232-C I/O CHANNEL INTERRUPT WORD FORMAT
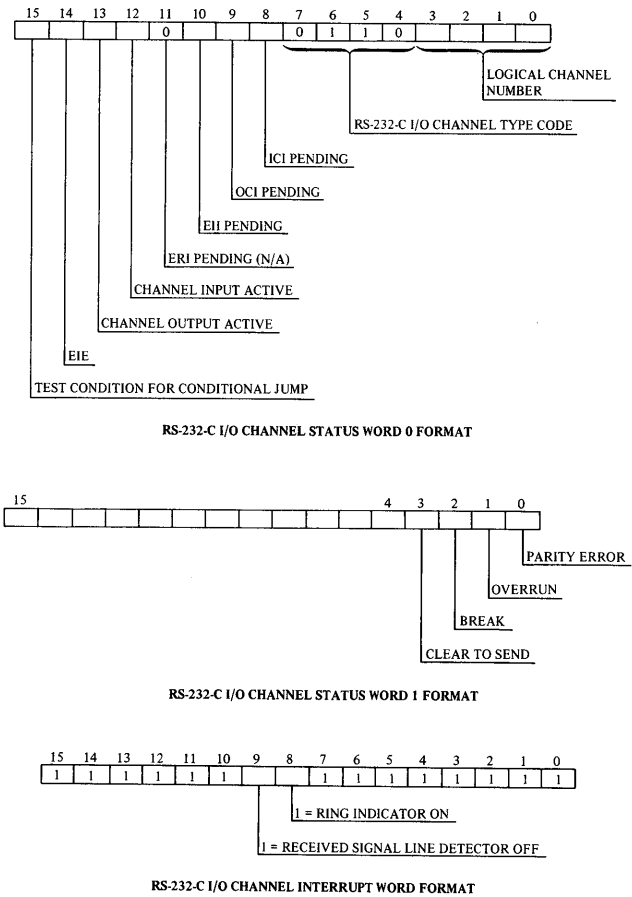
Figure 32. RS-232-C I/O Channel Status and Interrupt Word Formats

The channels are capable of full-duplex operation. The parallel channels, which are normally 16-bit channels, may be combined in pairs to form a single 32-bit dual channel, with the following restrictions: both channels must be the same interface type and the two channels must be in the same physical channel pair. That is, the modules must be installed in the chassis in slots corresponding to a channel pair (examples I/O slots 0 and 1 or 2 and 3).

The I/O channel capacity of the AN/AYK-14(V) is reduced by one channel for each 32-bit NTDS

46

channel implemented. All AN/AYK-14(V) NTDS channels meet the requirements of MIL-STD-1397.

For input operations, when a device is ready to transmit data or an interrupt code, it places the information on the input data lines (Figure 9) and raises the input data request or the interrupt request. The channel stores the word in memory and acknowledges using the input acknowledge line.

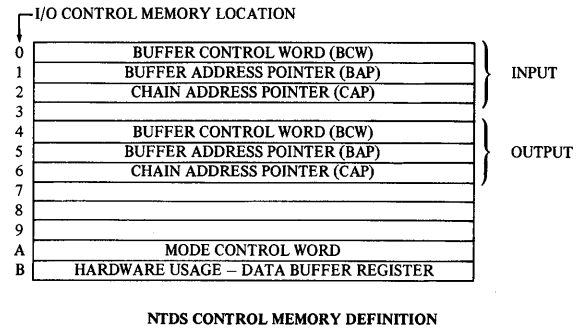For output operations, when an external device is ready to accept a command, it raises the external function request line to the channel. The channel responds by placing a command code on the output data lines and sets the external function acknowledge line. The channel can also "force" command words to external devices using the INITIATE Transfer command (Op Code E3, a=3). The channel places the command code on the data lines and sets the external function acknowledge. The external device reads the code and performs as commanded. When the device is ready to receive data, it raises the output request and the channel responds by placing a data word on the output lines and sets the output acknowledge line. External functions can be performed from either input or output chains.

Any parallel channel may be used for computer-to-computer communication with any computer with a compatible interface. The channel generates a timeout interrupt if the receiving computer does not respond in 300 to 400 milliseconds.

The control memory for NTDS channel operation is defined in Figure 33. The buffer control words format for input (word 0) and output (word 4) operation is also defined. The mode control word is defined in Figure 33. Mode status is maintained in the I/O mode status word and hardware status in the I/O channel hardware status word (Figure 34).
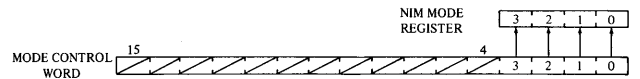
NTDS Serial Channel Operation

The NTDS serial channel provides asynchronous full-duplex communication at up to 10 megabits/second using the signals and message formats depicted in Figure 10.

NTDS CONTROL MEMORY DEFINITION

NTDS BUFFER CONTROL WORD

NTDS PARALLEL MODE CONTROL WORD

NTDS SERIAL MODE CONTROL WORD

Figure 33. NTDS Control Memory Usage

The control memory for NTDS serial channel operation has the same word assignments as for NTDS parallel channels (Figure 33) and the same buffer control word (Figure 33). When 32-bit word operation is used, a NIM serial module must be located in an even I/O module slot in the chassis and the related odd-number slot left empty (example: module in slot 0 and slot 1 empty). The I/O channel capacity of the AN/AYK-14(V) is reduced by one for each NTDS serial channel used in the 32-bit mode. Figure 34 defines the NTDS serial mode control word. Mode status is maintained in the NTDS serial channel mode status word and hardware status in the NTDS serial I/O channel hardware status word, Figure 35.

NTDS PARALLEL I/O MODE STATUS WORD

PRESENT OPERATING MODE
MODE SELECTION IDENT.
HARDWARE OPTION MODE
DUAL CHANNEL HALF WORD IDENT.

NTDS PARALLEL CHANNEL HARDWARE STATUS WORD

LOGICAL CHANNEL NUMBER
NTDS PARALLEL I/O CHANNEL TYPE
ICI PENDING
OCI PENDING
EII PENDING
ERI PENDING
CHANNEL INPUT ACTIVE
CHANNEL OUTPUT ACTIVE
EIE
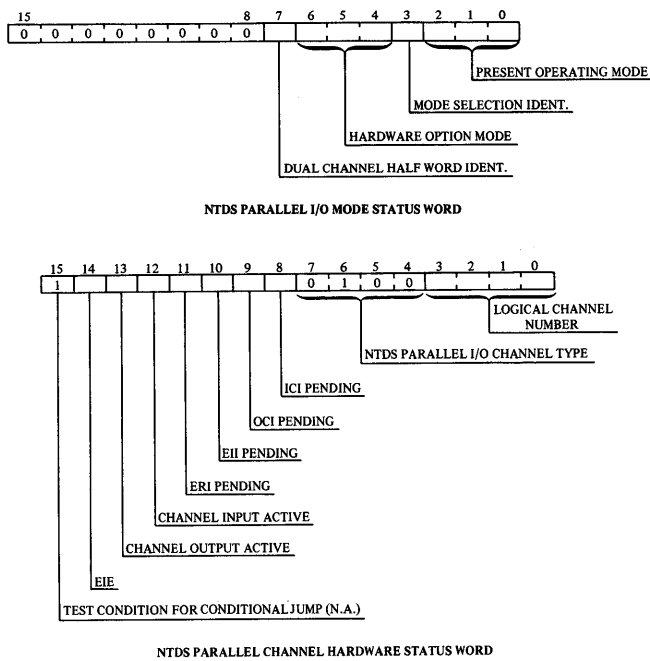TEST CONDITION FOR CONDITIONAL JUMP (N.A.)

Figure 34. NTDS Parallel I/O Status Words

The NIM-SERIAL I/O channel provides the following modes of operation:

- Computer to peripheral — 16-bit
- Computer to computer — 16-bit
- Test mode
- Computer to peripheral — 32-bit
- Computer to computer — 32-bit

The NIM I/O channel type mode of operation is defined by the four lower ordered bits of I/O channel control memory address A hexadecimal.

NTDS serial channels transfer output data and external functions over a single coaxial line and transfer input data and external interrupt codes over another coaxial line. Words are identified by a single bit that follows the synchronizing bit. Transmissions are initiated by the transfer of appropriate control frames between the computer and the peripheral device. An interchange of compatible control frames is required for each word that will be transmitted over the interface with the exception of a "forced" external function transfer. In this case the computer transmits the external function word even though the output request control frame does not specify the external function request.

NTDS SERIAL I/O CHANNEL MODE STATUS WORD

HARDWIRED OPTION
0 = 16 BIT MODE
1 = 32 BIT MODE
CURRENT OPERATING MODE
0X = 16 BIT MODE
1X = 32 BIT MODE
XI = TEST MODE

NTDS SERIAL I/O CHANNEL HARDWARE STATUS WORD

LOGICAL CHANNEL NUMBER
NTDS SERIAL I/O CHANNEL TYPE
ICI PENDING
OCI PENDING
EII PENDING
ERI PENDING
CHANNEL INPUT ACTIVE
CHANNEL OUTPUT ACTIVE
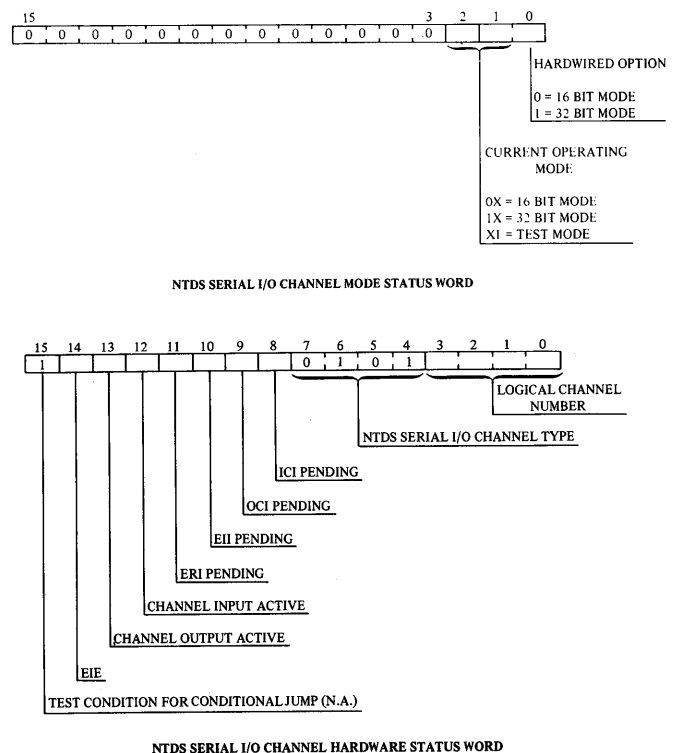EIE
TEST CONDITION FOR CONDITIONAL JUMP (N.A.)

Figure 35. NTDS Serial I/O Status Words

PROTEUS Digital Channel Operation

The PROTEUS serial channel module provides two PROTEUS digital channels (PDC) for full-duplex operation at a peak data transfer rate of 10 megabits/second according to NADC No. A30-15590. The channel provides failure indications and an error correction capability.

The PDC which transmits information is defined as the source and the receiving PDC is defined as the sink, as shown in Figure 12. The source transmits control words, data, and a sampling clock to an external sink. The control words and data are transmitted on the same pair of differential lines and the clock is transmitted on a separate pair of differential lines. The external sink responds on its own two pairs of differential lines: one transmits control words; and the other transmits a sampling clock signal. The internal sink receives control words and data from an external source and transmits control word responses back to the source. The control words and data are received on one pair of differential lines and the sampling clock on another pair. The sink responds with control words on one pair of lines and a sampling clock on a separate pair.

Messages include 4-bit control frames and 32-bit data words as shown in Figure 36. Each has an odd parity bit and a length bit to distinguish control from data frames.

A sequence is a series of transmissions and receptions which follow a predetermined channel protocol. There are sequences for control, data transfer, error indication, and error correction as shown in Table 11. Synchronizing control frames (Trans Code 5) are used in word sequences to delay word transfer while maintaining proper transmission rates.

The PIM control memory is defined in Figure 37. Since two 16-bit words are transferred between the PDM and memory for each 32-bit PROTEUS transmission, the word count must be set to an even number. Figure 37 also shows the formats of the sink mode control word and source mode control word.

The PIM has three channel status words as defined in Figure 38. The PIM is capable of ERI, EII, OCI, and ICI interrupts. Both input and output chains can be simultaneously active.

TABLE 11.
PROTEUS CHANNEL SEQUENCES

Hex Trans Code	Source to Sink Trans	Sink to Source Trans
0	Control interrupt word active	Request control interrupt word
1	Normal interrupt word active	Request normal interrupt word
2	Control word active	Request control word (RCW)
3	Data word active	Request data word (RDW)
4	Repeat, error detected	Repeat, error detected
5	Wait	RSB
6	STOP	ENDS
7	Sequence error	SEQE
8-D	Undefined	
E	System reset and load	SRL acknowledge
F	Channel reset	CRS acknowledge

I/O Processor (IOP) Operation

AN/AYK-14(V) computers can be configured as CPU only, IOP only, and CPU and IOP configurations. The processing functions of the CPU and IOP depend on the configuration as shown in Table 12.

The IOP operates in either processor or I/O controller (IOC) modes. When used as a stand-alone

49

processor, the IOP operates in processor mode. When used in conjunction with a CPU, the IOP can be operated in either processor or IOC mode as selected by the CPU program.

Figure 36. PROTEUS Message Formats

Figure 37. PROTEUS Control Memory Usage

Figure 38. PROTEUS Channel Status Words

TABLE 12. CPU/IOP FUNCTIONS

	CPU Only	IOP Only	CPU and IOP	
Processor instruction execution	Complete set	IOP Subset	CPU, complete set IOP, IOP Subset if IOP operating in processor mode	
I/O instruction execution	All I/O instructions	All I/O instructions	IOP executes IO Inst.	
Interrupt processing	All classes	All classes	<u>IOP in IOC mode</u> CPU, classes I and II directly; III indirectly IOP, class III, pass to CPU	<u>IOP in Processor mode</u> CPU, Classes I and II IOP, Class III, Classes I and II for IOP-related interrupts

Communications between the CPU and IOP are maintained using IOCR, START IOP, and Initiate External Interrupt instructions.

In a configuration with both CPU and IOP, the IOP initially enters the IOC mode. In this mode, the IOP shall perform all operations necessary for executing I/O command and chaining instructions. As an I/O controller, the IOP shall execute all I/O instructions and chain programs encountered by the CPU. I/O command instructions are executed by the CPU via an IOCR instruction and the CPU I/O command cells located at 60_{16} and 61_{16}. The IOCR command instructs the IOP to execute the instruction in the command cell. If this instruction initiates an I/O chain program, the IOP continues to execute the remaining instructions in the program.

If the instruction located in the command cell is a START IOP instruction, the IOP shall enter the processor mode. In this mode, the IOP shall execute a subset of the CPU instruction set. The IOP can initiate chains in this mode using the IOCR command and the IOP command cells 62_{16} and 63_{16}. Two software interrupts, the external interrupts 0 and 1, are provided for communication when both CPU and IOP are executing as processors. Execution of STOP instructions by the IOP shall cause it to return to the I/O controller mode.

IOP Interrupt Processing

When operating as an I/O controller, the IOP shall pass all class III interrupts to the CPU. Class I and II interrupts are handled directly by the CPU.

When operating as a processor, the IOP shall handle interrupts indicating hardware faults in its associated modules or software interrupts within the program it executes. The CPU shall not handle Class III interrupts when the IOP is in processor mode.

Operator Interface and Control

The AN/AYK-14(V) is designed to be used on applications requiring unattended operation. These operations do not require a computer operators panel, instead, the computer is operated indirectly from interfaces provided on the mission system control panel(s).

51

Operator control for maintenance and program development and debugging is provided by the loader/verifier (L/V) and computer control unit (CCU) as described in the Support Equipment section. These pieces of equipment interface to either the CPU or IOP via the computer support interface, which consists of a high speed serial channel and is accessible by front panel connector. The L/V is designed for flight line maintenance and program loading. The CCU provides full computer console display and control functions.

The computer support interface is in addition to the 16 AN/AYK-14(V) IO channels and is not addressable as an IO channel.

SOFTWARE AND FIRMWARE

The basic approach to software support for the AN/AYK-14(V) is to preserve existing operational and support software from the AN/UYK-20 program. The AN/AYK-14(V) computer executes the AN/AYK-14 instruction set, which is a compatible extension of the AN/UYK-20 instruction set. The U.S. Navy supports both AN/AYK-14 and AN/UYK-20 support software. The AN/AYK-14 software includes the CMS-2 compiler, FORTRAN IV, assembler, simulator, library, and system tape generator.

The general approach to AN/AYK-14 software development is to use a commercial host computer such as the CDC 6000 series to prepare software and then transfer to the AN/AYK-14 via magnetic tape and the CCU. The self-hosted AN/AYK-14 support software and a real-time executive program are also being developed.

BIT Firmware

The AN/AYK-14(V) BIT firmware is stored in the AN/AYK-14(V) PROM micromemory and detects and identifies failing SRAs in the AN/AYK-14(V) computer system. The BIT is executed on power-up, master clear, or conditionally via the software diagnostic jump command. The execution time of the BIT is 2.2 milliseconds. If an error is identified by the BIT, the BIT indicator is set, the hardware failure interrupt enabled, and a status/isolation code is made available for the computer support equipment.

The BIT verifies portions of the GPM, MCM, and the GPM interface to other system modules. The following routines are contained in the BIT:

- Microsequence test — checks out computer system microaddressing
- Register test — checks the ALU registers as well as the K, U, and I registers
- ALU test — checks for operation of arithmetic and logical functions
- File and C-file test — checks for storing ability
- MCM page file test — checks for storing from the CPUBUS and IOBUS
- Memory interface test — checks transfer of data to and from memory on both the CPUBUS and the IOBUS
- Firmware event test — sets and clears firmware events to determine if coding and acknowledgement is correct
- Bus Verification and PSM RTC test — checks the CPUBUS and IOBUS as well as PSM real-time clocks 0, 1, and 2

The BIT establishes a high confidence level in the computer's ability to properly perform the SRA fault isolation diagnostic.

In-Flight Performance Monitoring (IFPM)

The IFPM modules ensure that the AN/AYK-14 computer system is operational and can perform its tactical mission. The modules are written in MACRO 20/14 assembly language; however, their calling sequence follows the CMS-2 convention.

The IFPM modules are intended to be imbedded in user code at points appropriate to provide continuous operational monitor. The instructions chosen provide use from either the CPU or IOP. Table 13 lists the tests, their size, and execution time.

The hardware built-in test equipment (BITE) interfaces and supports the IFPM to provide failure detection predicted to be 94% of the faults that can occur in the systems resources used. Table 14 lists the BITE functions.

TABLE 13. IFPM MODULES

Name	Symbol	Test Responsibility	Size (Words)	Time (MS)
Quick Look	BCMQL	CPU, Memory Interface	300	2.0
Memory	BMEM	MCM and Memory	200	30/page
CPU	BCPU1	CPU Functions	500	3.1
	BCPU2		850	11.0
IOP	BIOP	IOP Functions	600	3.5
IO	BSIM	Channel Tests	300	6.5
	BDIM		300	4.1
	BNIM		250	2.3
	BPIM		310	4.5
	BRIM		260	119.1

TABLE 14. FUNCTIONS CHECKED BY BIT EQUIPMENT

- MEMORY PARITY
- MEMORY PROTECT
- MEMORY CHANNEL TIMEOUT
- POWER MONITORING
- OVERTEMPERATURE MONITORING
- BUS TIMEOUTS
- I/O CHANNEL PARITY
- I/O CHANNEL TIMEOUT
- SIM MANCHESTER CODE FORMAT VERIFICATION
- BIT TIMER
- BIT INDICATOR

SRA Fault Isolation Diagnostic

The SRA diagnostics are software programs to detect and isolate hardware failures in the AN/AYK-14 computer system. The diagnostics are modular and configurable to test all possible AN/AYK-14 computer configurations and thoroughly test the three major functions of the system: memory, processor, and input/output. These diagnostics isolate a fault to one SRA 86 percent of the time and to two SRAs 93 percent

of the time. The diagnostic is loaded via the L/V or CCU, which provides a means of displaying test results.

CCU Program

The CCU program controls the 8080 processor in the CCU and provides the operator console control functions described in the CCU Support Equipment section. The 8080 processor within the CCU is also programmed to transfer data from magnetic tape to the RAM buffer memory and then to the AN/AYK-14 memory. The CCU also transfers data from the AN/AYK-14 to magnetic tape via the RAM buffer memory. The program resides in the CCU memory.

L/V Program

The L/V program controls data transfers from the magnetic tape to the AN/AYK-14. Information is also transferred to and from the switch and display indicators on the control panel.

CCU and L/V Diagnostics

The CCU and L/V BITS are programmed tests designed to provide fault detection and isolation capabilities. These programs are initiated manually (power-up an operator initiated) and will test CCU and L/V hardware functions.

Microcode Cross-Assembler

The microcode cross-assembler is a FORTRAN coded assembler capable of accepting microcode instruction from source cards and tapes, and of producing program listings and absolute object code onto magnetic tape and disk. This absolute object code is executable by the AN/AYK-14. The assembler is capable of assembling a microcode program of at least 4K microcode words. It can be hosted on any computer hosting ANSI Standard FORTRAN (Version 3.9, 1966) with sufficient memory and two tape units.

Microcode Simulator

The microcode simulator provides an independent host capability to enable a user to test the microcode program assembled for the AN/AYK-14(V). The simulator fully simulates the microcode instruction repertoire, and interrupts, accepts, and executes assembled AN/AYK-14(V) microcode. The simulator requires less than 60K bytes of memory and is coded in FORTRAN. It is operable on any computer hosting ANSI Standard FORTRAN (Version 3.9, 1966).

MTASS

The Machine Transferable AN/UYK-20, AN/AYK-14 Support Software system (MTASS) is a highly integrated software development and maintenance tool created for U.S. Navy standard computer systems. It provides an AN/UYK-20 or AN/AYK-14 programming environment on computers other than the target computer and is composed of the following components:

- CMS-2M Cross Compiler
- MACRO 20/14 Cross Assembler
- SYSGEN 20/14 Loader/System Generator
- SIM20 AN/UYK-20 Simulator
- SIM14 AN/AYK-14 Simulator
- FORTRAN Cross Compiler*

MTASS provides the capability of maintaining source and object elements, compiling and assembling source elements, and checking out AN/UYK-20 or AN/AYK-14 programs on large program generation (host) computers. After the programs have been designed, coded, debugged and integrated on the host computer, they can be transferred to the target (AN/UYK-20 or AN/AYK-14) computer and integrated with the target application system.

*Approval for use must be obtained from NAVMAT 09Y (TADSO).

MTASS is supported by large-scale computer installations and is available for any number of simultaneous users to utilize the system in batch, remote batch, or timeshare mode.

MTASS allows programming organizations to react immediately to requirements for software development and maintenance projects prior to or in connection with the availability of the AN/UYK-20 or AN/AYK-14 computer system.

MTASS Host Computers

MTASS requires a host computer with a 32-bit word (minimum), an operating system with a FORTRAN compiler, random and sequential access I/O, and a file management system.

The existing host computer includes the following:

- Control Data 6600 and Cyber Series
- IBM 360/370 Series
- Univac 1100 Series
- AN/UYK-7
- Honeywell 600/6000 Series
- Digital DEC 10 Series
- Digital DEC 20 Series
- Digital VAX 11/780

Standard Real-Time Executive (SDEX/M)

SDEX/M is the nucleus of the AN/AYK-14(V) real-time system operating in the AN/AYK-14(V) computer. It is designed to operate in modes for both AN/AYK-14(V) and AN/UYK-20 Computers. A complete system is formed and optimized by the addition of site-specific system functions and user modules. Functions of SDEX/M are: initialization, scheduling, interrupt management, I/O management, and error management.

APPLICATIONS

The AN/AYK-14(V) is applicable to a wide variety of military systems. The basic module design is applicable to MIL-E-5400 Class II, MIL-E-4158,

and MIL-E-16400 environments. The emphasis is on military applications and this is reflected in the I/O modules designed to meet standard military interfaces such as NTDS and MIL-STD-1553A without modification of either computer or peripheral device. Currently planned applications are configured from the existing modules and chassis (as shown in Table 15) to provide a high degree of hardware commonality and complete software compatibility.

The AN/AYK-14(V) can be used efficiently in applications ranging from a dedicated processor incorporated as "bare" modules in the user's system, to large-scale multiprocessor configurations with extensive memory and I/O capacity. Because of the functional modularity, the size, weight, and power of the processor configuration will be proportional to the processing capacity. The microprogrammable feature permits extension of the instruction set or the incorporation of frequently used algorithms directly into firmware for high-speed processing.

Some of the general application areas for which the AN/AYK-14(V) can be effectively employed are:

- Avionics central processor — overall management of aircraft systems, operator interfaces, and data processing. The AN/AYK-14(V) modularity permits an integral MIL-STD-1553A bus controller for compatible interfacing subsystems.

- Communications processing — applicable to a broad range of tasks from processing for message terminal to store-and-forward switch applications. Bit and byte data manipulation, memory protection features, and capacity for large memory systems provide benefits in these applications.

- Sensor data processing — process radar or sonar data, control system operation, and perform required coordinate conversion. The use of the IOP module and firmware algorithms can enhance processing throughout.

- Weapons control/fire control — high-speed processing in a compact unit

efficiently matches requirements of modern weapon systems involving multiple simultaneous engagements and sophisticated weapons management and control procedures. Availability of high-speed, floating-point arithmetic simplifies program development and updating to accommodate new weapons and tactics.

- Control systems — high reliability of the AN/AYK-14(V) processor used with PROM micromemory and program memory provides a high-performance computer in a few modules for dedicated critical digital flight control applications.

- Navigation — single and double precision fixed-point arithmetic, floating-point arithmetic, and flexible real-time clock system are valuable features in navigation system applications.

- Command and control — flexible interrupt system, multiple registers, memory protection features, large memory addressing capability, and multiple processor configuration capability foster application of the AN/AYK-14(V) to large-scale command and control systems.

Figure 39 depicts how various configurations of AN/AYK-14(V) computer modules can be employed in an avionics application to provide a high degree of hardware commonality. In this example, the MIL-STD-1553A multiplex bus, implemented by the AN/AYK-14(V) SIMs provides the integrating interface between subsystems. The 1553A remote terminals are implemented with SIM, IOP, PCM, and special I/O modules suitable to the connecting devices. Even though the module configuration may vary, as shown for the navigation and central computer configurations, a common instruction set and software system can be employed to further reduce cost to the user.

TABLE 15. MODULE REQUIREMENTS OF SELECTED COMPUTER APPLICATIONS

User System Requirements	PROGRAMS									
	LAMPS	EP-3E		SSIS	AIDS	LAMPS ESM	MADAIR	E2C	F-18	FIREBRAND
	XN-1A	XN-1D	XN-3B	XN-1E	XN-1F	XN-1J	XN-1L	XN-1M	XN-5A	XN-5B
PCM-1			1						1	1
PCM-2	1	1		1	1	1	1	1		
Core Memory (K)	96	96	64	64	128	32	128	64	64	32
MCM	1	1		1	1	1	1	1	1	1
EAU		1		1	1		1	1		
GPM	1	1		1	1	1	1	1	1	1
PSM	1	1		1	1	1	1	1	1	1
IOP		1					1	1	1	1
IOP Jumper Plug	1		1	1	1	1				
SIM (1553A)	2	1		1		2		1	3	
NIM-B (Fast)			1							
NIM-S (Serial)			1							
NIM-C (ANEW)		2	1	1			4	2		
RIM (RS-232)		1	2	2						
PIM (PROTEUS)	1		1							
RXM		1						1		
RXM Adapter		1						2		
DIM (Discrete)	1	1		1	1	1	1	1	1	
BEM (BUSXTND)		1 ·	1	1	1	1				
XN-1 Chassis	1	1		1	1	1	1	1		
XN-3 Chassis			1							
XN-5 Chassis									1	1
Fan	1	1	1	1	1	1	1			1
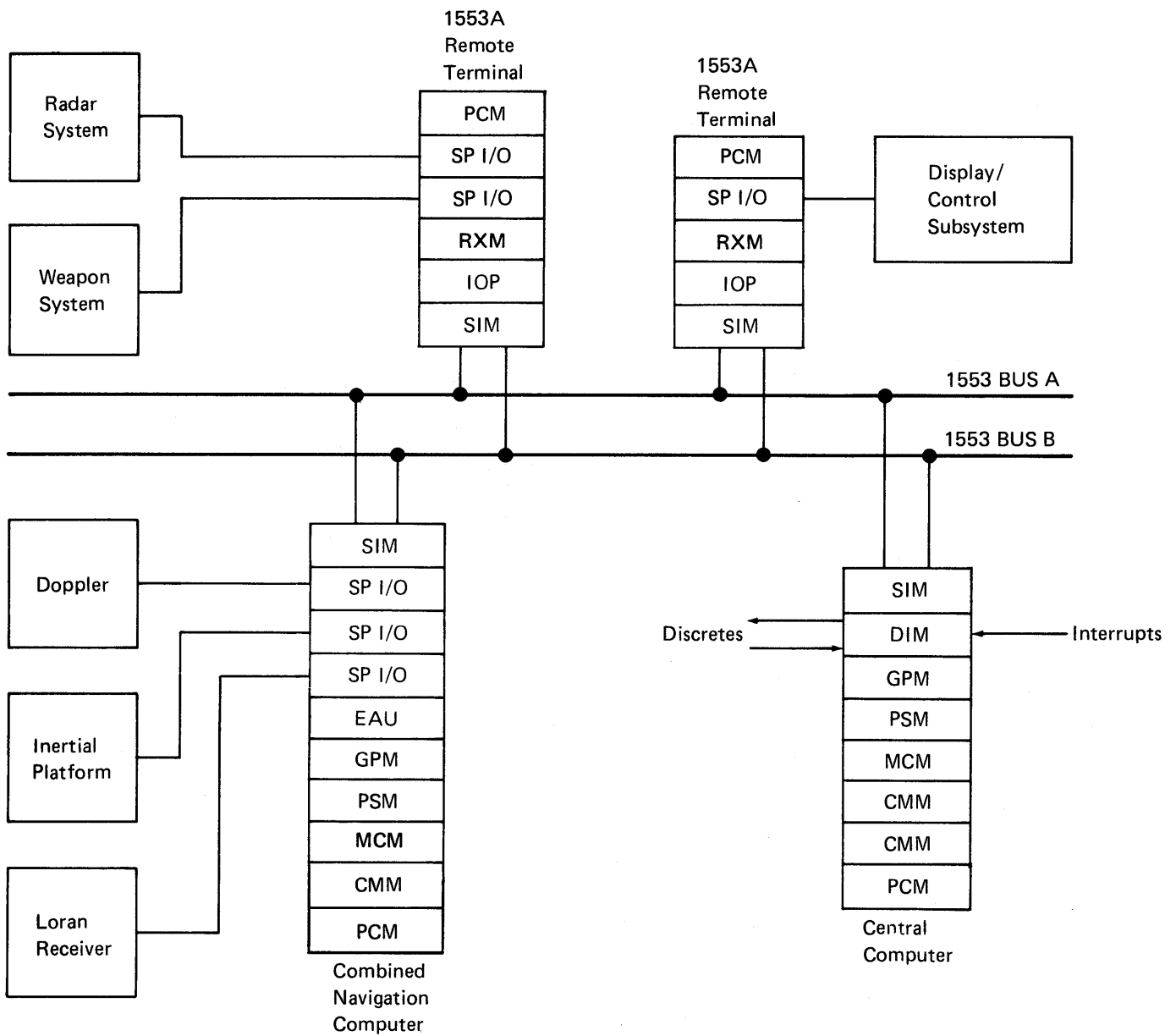
Figure 39. Commonality of Avionics Processing Based on AN/AYK-14(V) Computer Family

STANDARD AIRBORNE COMPUTER

★AN/AYK-14★

CONTROL DATA CORPORATION

Aerospace Division
P.O. Box 609
Minneapolis, Minnesota 55440

G13693

0180