

ite technical n  
AD-A214 091  
bn

2

# Aircraft Reply and Interference Environment Simulator (ARIES) Hardware Principles of Operation: Volume II (Appendixes)

Edward Mancus

October 1989

DOT/FAA/CT-TN88/4-II

Document is on file at the Technical Center Library, Atlantic City International Airport, N.J. 08405

DISTRIBUTION STATEMENT A



U.S. Department of Transportation  
Federal Aviation Administration

Technical Center  
Atlantic City International Airport, N.J. 08405

DTIC  
OCT 09 1989

89 11 01 035

#### **NOTICE**

This document is disseminated under the sponsorship of the U.S. Department of Transportation in the interest of information exchange. The United States Government assumes no liability for the contents or use thereof.

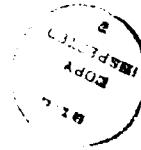
The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the objective of this report.

1. Report No. DOT/FAA/CT-TN88/4-II		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle AIRCRAFT REPLY AND INTERFERENCE ENVIRONMENT SIMULATOR (ARIES) HARDWARE PRINCIPLES OF OPERATION: VOLUME II (APPENDIXES)				5. Report Date October 1989	
				6. Performing Organization Code	
7. Author(s) Edward Mancus				8. Performing Organization Report No. DOT/FAA/CT-TN88/4-II	
9. Performing Organization Name and Address Federal Aviation Administration Technical Center Atlantic City International Airport, New Jersey 08405				10. Work Unit No. (TRAIS)	
				11. Contract or Grant No.	
12. Sponsoring Agency Name and Address U.S. Department of Transportation Federal Aviation Administration Technical Center Atlantic City International Airport, New Jersey 08405				13. Type of Report and Period Covered  Technical Note	
				14. Sponsoring Agency Code	
15. Supplementary Notes					
16. Abstract <p>The Aircraft Reply and Interference Environment Simulator (ARIES) makes possible the performance assessment of the Mode Select (Mode S) sensor under its specific maximum aircraft load. To do this, ARIES operates upon disk files for traffic model and interference to generate simulated aircraft replies and fruit, feeding them to the sensor at radio frequency.</p> <p>Support documentation for ARIES consists of:</p> <p>ARIES Hardware Maintenance Manual: Volume I (DOT/FAA/CT-TN88/3)  Appendixes of the Hardware Maintenance Manual: Volume II  ARIES Hardware Principles of Operation: Volume I (DOT/FAA/CT-TN88/4)  Appendixes of the Hardware Principles of Operation: Volume II  ARIES Software Principles of Operation (DOT/FAA/CT-TN87/16)  ARIES Software User's Manual (DOT/FAA/CT-TN88/15)</p> <p>The Appendixes to the Hardware Principles of Operation provides (1) the acronyms and abbreviations used within the document, (2) detailed information covering the development and implementation of controller microcode, and (3) Uplink Receiver digital alignment. (KR)</p>					
17. Key Words ARIES Mode S Microcode			18. Distribution Statement Document is on file at the Technical Center Library, Atlantic City International Airport, New Jersey 08405		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 131	22. Price

TABLE OF CONTENTS

	Page
EXECUTIVE SUMMARY	v
APPENDIXES	
A - List of Acronyms and Abbreviations	A-1
B - Controller Microcode	
B.1 INTRODUCTION	B-1
B.2 BASIC AMD DEVELOPMENT SYSTEM PROCEDURES	B-5
B.2.1 AMDASM Assembly Program	B-5
B.2.2 AMPROM Support Program	B-7
B.2.3 AMD-to-ARIES Data Transfer Procedure	B-12
B.3 MICROCODE PROGRAMS	B-13
B.3.1 MRG Microcode Program	B-13
B.3.2 FRG Microcode Program	B-37
B.3.3 RRG Microcode Program	B-78
C - Uplink Receiver PAM/DPSK Alignment	

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	<input type="checkbox"/>
<i>form 50</i>	
By	
Distribution/	
Availability Codes	
Dist	
<b>A-1</b>	



## EXECUTIVE SUMMARY

The Aircraft Reply and Interference Environment Simulator (ARIES) is a stand-alone target simulation system designed and fabricated at the Federal Aviation Administration (FAA) Technical Center to analyze the performance characteristics of the Mode Select (Mode S) Beacon System sensor operating within capacity and high density beacon environments. A total of four ARIES systems have been fabricated to support the development of the Mode S sensor during production phases. In addition, the ARIES will be used to support acceptance testing to be conducted by the FAA.

Volume II of the Hardware Principles of Operations Manual contains the list of acronyms and abbreviations found in Volume I, the digital alignment procedure for the Uplink Receiver, and a detailed description of the microcode implemented in the three controllers in the ARIES. The development of the microcode for the controllers is described in sufficient detail for a user familiar with the microcode to make modifications to the code, if it should become necessary.

APPENDIX A

LIST OF ACRONYMS AND ABBREVIATIONS

AC	Altitude Code Field
ACP	Azimuth Change Pulse
ADC	ADCCP Data Converter
A/DC	Analog-to-Digital Converter
ADCCP	Advanced Data Communication Control Procedure
ADRS	Controller Address Code
AIP	ATCRBS Interrogation Processing
ALU	Arithmetic Logic Unit
AMD	Advanced Micro-Drive
ARIES	Aircraft Reply and Interference Environment Simulator
ARP	Azimuth Reference Pulse
ARPG	ATCRBS Random Process Generator
ASR-9	Air Surveillance Radar (model 9)
ATCRBS	Air Traffic Control Radar Beacon System
AU	Azimuth Unit
AZGEN	Azimuth Generator/Decoder
BSELCH	Buffered Selector Channel
CA	Capability Field
CD-2	Common Digitizer (model 2)
dB	decibel
DF	Downlink Format
DPSK	Differential Phase Shift Keying
DR	Downlink Request
FAA	Federal Aviation Administration
FAT	Fruit ARIES Target
FCS	Frame Check Sequence
FIFO	First-In First-Out Memory
FRC	Fruit Reply Controller
FRG	Fruit Reply Generator
FS	Flight Status Field
GPI	General Purpose Interface
IC	Integrated Circuit
ID	Identity Code Field
IF	Intermediate Frequency
I/O	Input/Output
ISR	Interrupt Service Routine
LO	Local Oscillator
L/R	Left/Right
LSB	Least Significant Bit
LSI	Large-Scale Integration
Mode S	Mode Select
MAT	Modeled ARIES Target
MB	Comm-B Message Field
MSB	Most Significant Bit
MC	Comm-C Message Field
MD	Comm-D Message Field
MHz	Megahertz
MIP	Mode S Interrogation Processing
MIT	Missing Interrogation Timer
MRC	Modeled Reply Controller
MRG	Modeled Reply Generator
MRPG	Mode S Random Process Generator
M/S	Mainbeam/Sidelobe
MSI	Medium Scale Integration

MSR	Machine Status Register
MUX	Multiplexer Bus
OP	Controller Operation Code
PAM	Pulse Amplitude Modulation
PI	Parity Identifier
PPM	Pulse Position Modulation
RAM	Random Access Memory
RAR	Read Address Register
RGEN	Reply Generator
RCVR	Uplink Receiver
RF	Radio Frequency
RNG	Random Number Generator
RPG	Random Process Generator
ROM	Read Only Memory
RRC	Radar Report Controller
RRG	Radar Report Generator
RTQC	Real Time Quality Control
SCC	Subchannel Controller
S/H	Sample and Hold
S/L	Short/Long
SPDT	Single Pole Double Throw
SPI	Special Position Indicator
SPR	Synchronous Phase Reversal
SR	Status Request
STU	Self Test Unit
TOAR	Time of Arrival
TOY	Time-of-Year
TTL	Transistor-to-Transistor Logic
UF	Uplink Format
UIT	Universal Interval Timer
μSR	Micro Status Register
VPQ	Video Pulse Quantizer
WAR	Write Address Register
WCS	Writable Control Store



APPENDIX B  
CONTROLLER MICROCODE

## LIST OF ILLUSTRATIONS

Figure		Page
B.1-1	Example of Microinstruction Fields and Functions	B-3
B.1-2	Microdevelopment Equipment Block Diagram	B-4
B.2-1	Interrelationship Between AMD Software Programs	B-6
B.2.1-1	Definition File (4 Sheets)	B-8
B.3.1-1	MRG Microcode Flow Diagrams (9 Sheets)	B-14
B.3.1-2	MRG Microcode Program Listing (14 Sheets)	B-23
B.3.2-1	FRG Microcode Flow Diagrams (15 Sheets)	B-38
B.3.2-2	FRG Microcode Program Listing (24 Sheets)	B-54
B.3.3-1	RRG Microcode Flow Diagrams (14 Sheets)	B-79
B.3.3-2	RRG Microcode Program Listing (23 Sheets)	B-94

## LIST OF TABLES

Table		Page
B.1-1	Microinstruction Set	B-2
B.3.1-1	MRG Mode Selection Field	B-13
B.3.2-1	FRG Mode Selection Field	B-53
B.3.3-1	RRG Mode Selection Field	B-93
B.3.3-2	RRG Message Types	B-93

## B.1 INTRODUCTION.

Three bit-slice microprocessors are used in the ARIES system. They serve as controllers in three hardware generators. Two of the generators, the Modeled Reply Generator (MRG) and the Fruit Reply Generator (FRG), output beacon replies to the Mode S sensor at the radio frequency (RF) level. The third generator, the Radar Report Generator (RRG), transmits digital radar report messages to the Mode S conforming to the Advanced Data Communication Control Protocol (ADCCP).

Each controller is loaded with its own microcode program. Each program consists of a normal operation routine, and all of the self-check and diagnostic routines needed to support hardware validation. These programs will be covered later in section B.3. (For further information on the controller hardware, refer to section 6.4 in Volume I of the ARIES Hardware Principles of Operation.)

The microcode instructions, making up the microcode program, are single executable words, each resident in a programmable storage memory. This memory is made up of seven high speed 8-bit x 512-word programmable read only memory (PROMs).

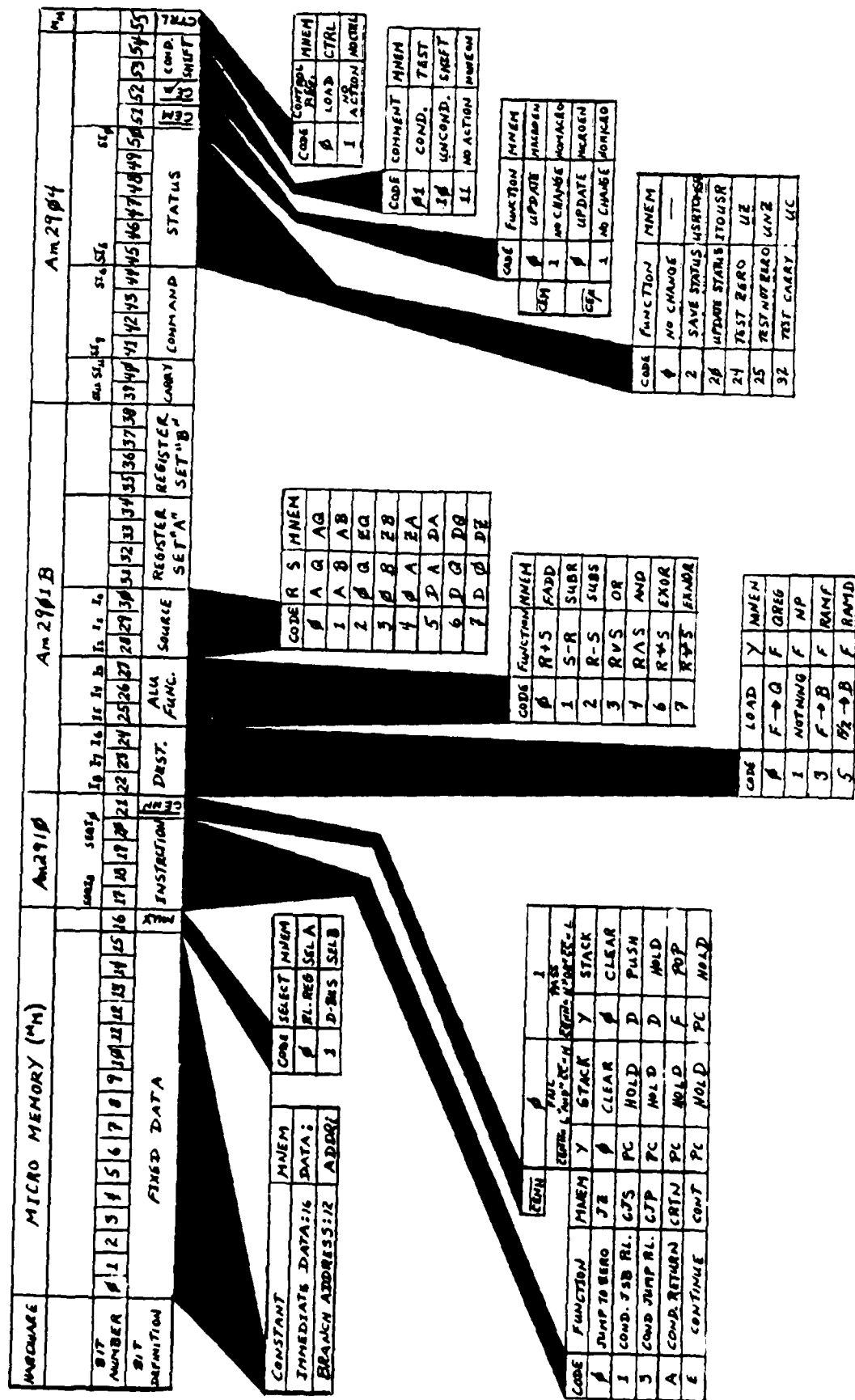
The three ARIES controllers use the same microcode instruction set. This instruction set is listed in table B.1-1, where the mnemonics for each instruction, along with a brief description, is given. Each microcode instruction is a 56-bit word. The fields and functions within each word are presented in figure B.1-1. (Note the functions shown in figure B.1-1 are only the ones implemented in the ARIES microcode instruction set.) There are seven classes of instructions: (1) program control, (2) data movement, (3) arithmetic, (4) logical, (5) register manipulation, (6) shift, and (7) special functions. The symbols used are as follows:

<u>Symbols</u>	<u>Meaning</u>
Rn	R0 - R15: as specified in the A address field A0, A1, A2, A3 (A0 = MSB)
Rm	R0 - R15: as specified in the B address field B0, B1, B2, B4, (B0 = MSB)
DATAi	Data field: as specified by bits 0 through 15. (Bit 0 = MSB)
ADDRi	Branch address field: as specified by bits 4 through 15. (Bit 4 = MSB)

The microcode programs were developed on the Advanced Micro-Device (AMD) Microprogram/Bit-Slice Development System. A functional block diagram of the microcode development equipment is shown in figure B.1-2. A line printer and a video display terminal are used as the standard input/output (I/O) peripherals. An additional output line is shown in figure B.1-2, which is used to transfer the microcode data files (one's and zero's pattern) to the Aircraft Reply and Interference Environment Simulator (ARIES) Development/Maintenance System for disk file storage. These files contain the data pattern to be programmed (burned) into the controllers microcode memory (PROM).

TABLE B.1-1. MICROINSTRUCTION SET

Instruction Class	Mnemonic	Description
Program Control	JMP ADDRi	Unconditional Jump to Address
	BNZ ADDRi	Branch to Address if CC(Z)=0
	BRZ ADDRi	Branch to Address if CC(Z)=1
	BRC ADDRi	Branch to Address if CC(C)=1
	JSR ADDRi	Unconditional Jump to Subroutine
	RTN	Return from Subroutine
	JPZ	Jump to Address 0
Data Movement	IN Rn	Input Data from D-Bus --> Rn
	OUT Rn	Rn --> Output Data to Y-Bus
	INQ	Input Data from D-Bus --> Q
	LCR Rn	Rn --> Control Register
	MOV Rn,Rm	Rn --> Rm
Arithmetic	ADD Rn,Rm	Rn + Rm --> Rm
	ADDI DATAi,Rn,Rm	DATAi + Rn --> Rm
	CMPI DATAi,Rn,Rm	DATAi - Rn --> Rm
Logical	AND Rn,Rm	Rn "AND" Rm --> Rm
	ANDI DATAi,Rn,Rm	DATAi "AND" Rn --> Rm
	ANDIQ DATAi	DATAi "AND" Q --> Q
	COM Rn	Rn --> /Rn
Register Manipulation	CLR Rn	0 --> Rn
	DEC Rn	Rn - 1 --> Rn
	DECQ	Q - 1 --> Q
	LIM DATAi,Rn	DATAi --> Rn
Shift	SRA Rn	Rn/2 --> Rn
Special Functions	NOP	No Operation
	RSTCR	Reset Control Register
	LIMCR DATAi	DATAi --> Control Register



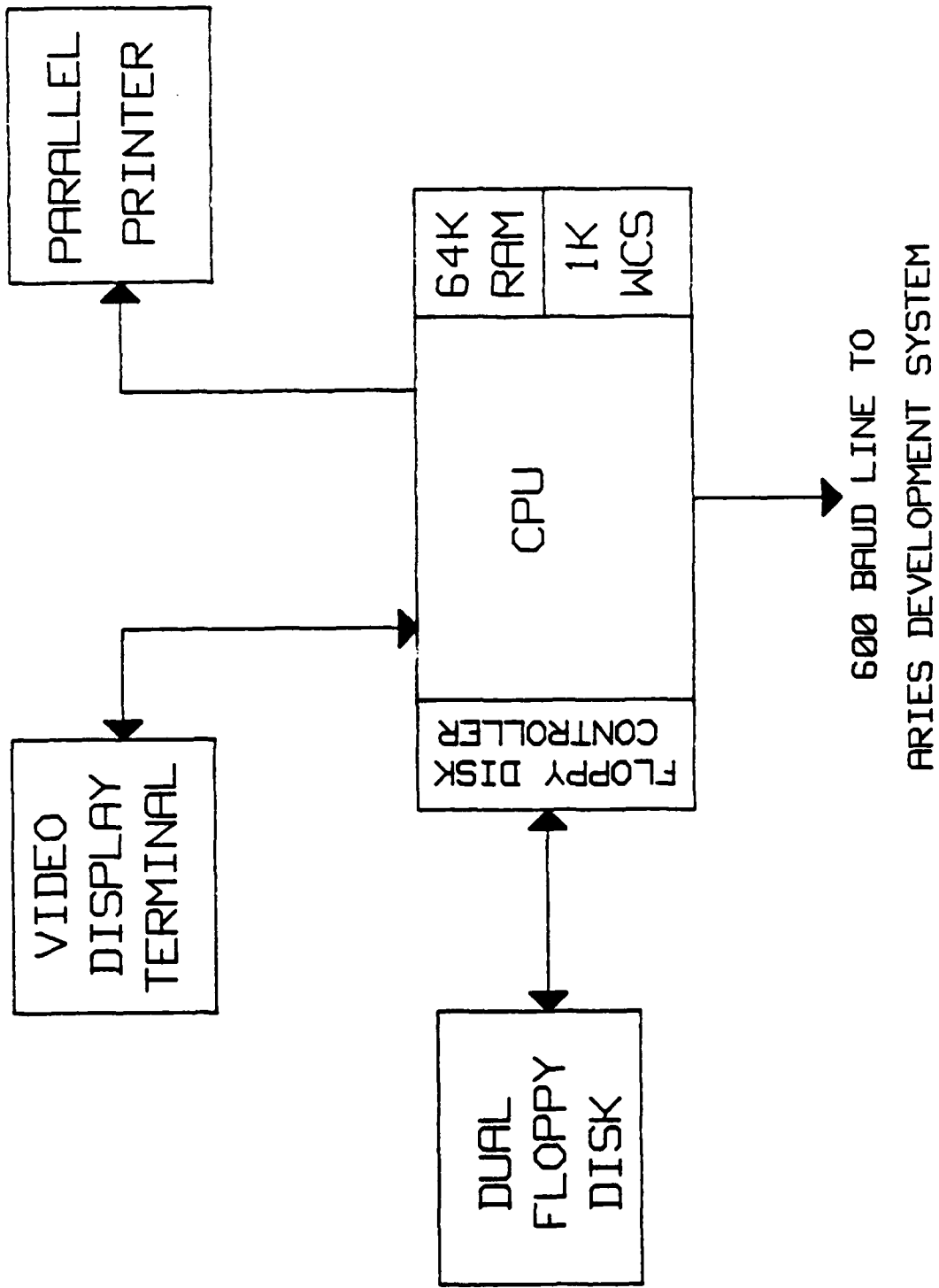


FIGURE B.1-2. MICRODEVELOPMENT EQUIPMENT BLOCK DIAGRAM

## B.2 BASIC AMD DEVELOPMENT SYSTEM PROCEDURES.

### System Boot-Up Procedure:

- a. Apply power to the Video Display Terminal and the Line Printer.
- b. Apply power to the AmSYS29/10B Computer Chassis and the Dual Floppy Disk Drive Chassis.
- c. Insert the system diskette "AMDOS System" in drive A (the main disk drive).
- d. Depress the reset toggle switch on the front of the AmSYS29/10B Computer Chassis to initiate the bootstrap operation. Note the AMDOS will identify itself with a short message and a prompt on the display as follows:

```
AMDOS 29 64K, Version 3.00 - 06/16/81
A>
```

At this stage, the AMDOS operating system (OS) is ready to receive commands. (For further information on the OS commands refer to the AMD AmSYS29/10B Microprogram Development System User's Manual, No. 059920027-003.)

- e. To obtain hard copies, select the parallel printer by entering the following:

```
A>STAT LST: - UL1: <return>
```

- f. To print out a file enter the following:

```
A><CTRL>P (Activate display printing)
A>TYPE filename.ext
A><CTRL>P (Discontinue display printing)
```

The following procedures are described with the assumption that the AMD Development System has been properly brought up and running correctly. These procedures summarize the overall process in developing the microcode for the three ARIES controllers. Figure B.2-1 shows the interrelationship between the programs covered below.

### B.2.1 AMDASM Assembly Program.

The AMDASM assembly program operates on the Advanced Micro Devices' AmSYS29 under the AMDOS29 Operating System. AMDASM operates in two phases: (1) the definition phase (PHASE1) and (2) the assembly phase (PHASE2). The definition phase establishes word length and definition of formats and constants (the definition file). The assembly phase is the assembly procedure (the assembly file) performed on a user's source program using the formats and constants from the definition file. This phase reads a symbolic program (the user's program) and handles most common assembler features, such as labeling and setting the cross-reference tables. The definition phase is executed first to set up the table which associates the user's format names and constant names with their corresponding bit patterns. (Refer to AmSYS29/10 Microprogram Support Software User's Manual, No. 059910515-003 for further information on the AMDASM assembly program.)

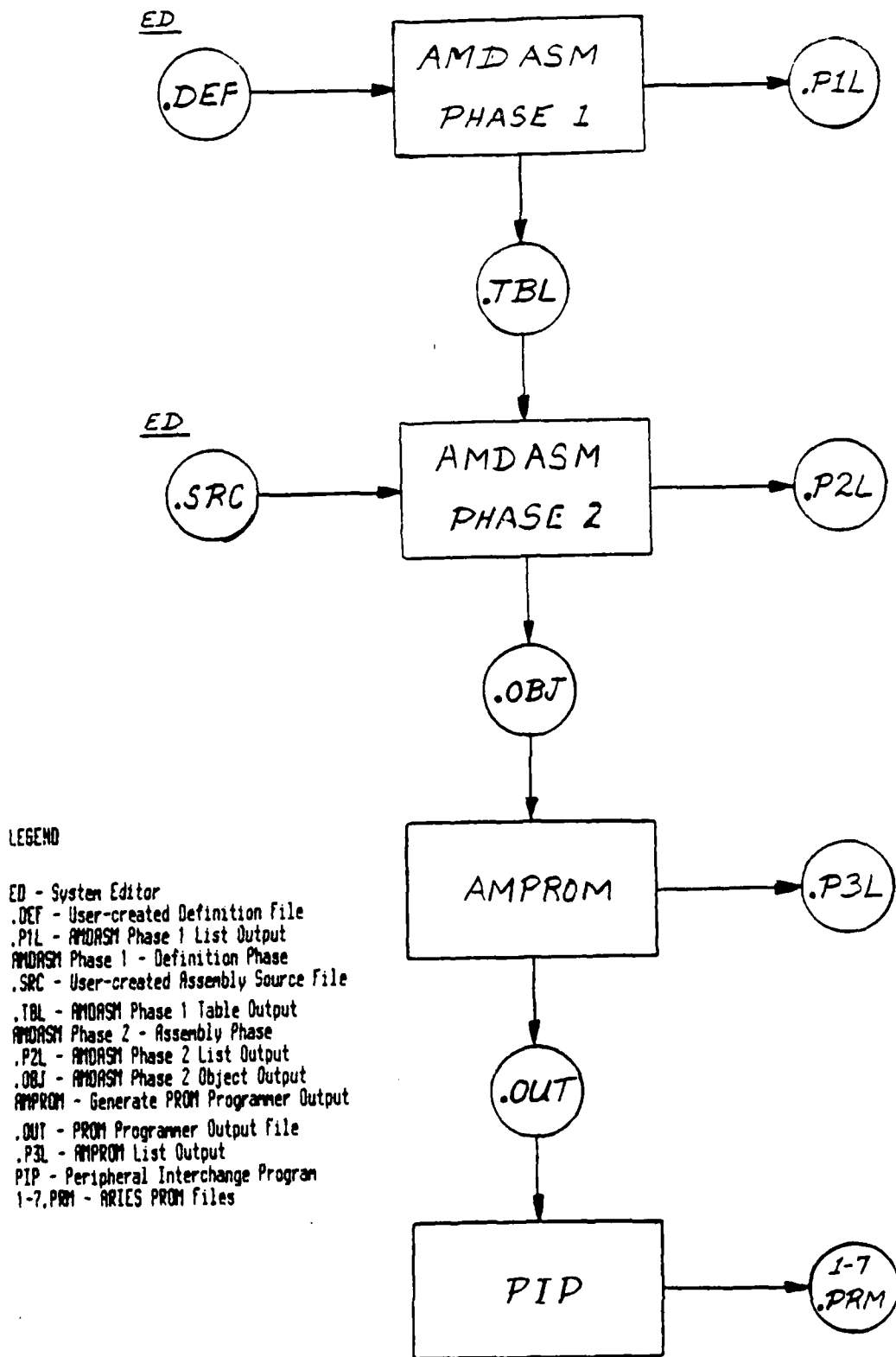


FIGURE B.2-1. INTERRELATIONSHIP BETWEEN AMD SOFTWARE PROGRAMS



The user enters AMDASM P1-filename P2-filename then depresses the <return> key to run the AMDASM assembler. The "filename" is the name of two existing files: (1) the PHASE1 input file (filename.DEF) and (2) the name of the PHASE2 input file (filename.SRC). If no errors are detected, the following sequence should be observed on the display after the program is finished.

A>AMDASM P1-filename P2-filename <return>

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

TOTAL PHASE 1 ERRORS - 0

TOTAL PHASE 2 ERRORS - 0

A>

The definition phase allows the user to define the microword length and the constants and formats used to write the source programs. An example of a definition file is shown in figure B.2.1-1. This file defines the micro-instructions for all three microcode programs discussed in section B.3. Note in the definition file, the microinstruction length is defined as 56 bits (WORD 56). Each user defined symbol is given a specific bit pattern associated with it. Also, a format name is used to define all or part of each micro-instruction. Once the definition file has been executed by the AMDASM assembly program, its output (filename.TBL) is retained for use during the assembly phase (PHASE2).

The definition file, as well as the source program, are created by the user using the AMD Text Editor. The Text Editor is accessed by entering ED filename.ext <return>, where "filename" is the name assigned by the user, and ".ext" is the proper extension defining the file type (".DEF" for a definition file or ".SRC" for a source file).

The Text Editor will prompt the user with an "\*" when it is ready to receive a command. The user then enters #A <return> to move the entire source file to memory. The user is now ready to create or modify the source file. (For further details on the Line Editor commands refer to the AMD Text Editor User's Manual, No. 050510294-001.)

### B.2.2 AMPROM Support Program.

The AMPROM support program is used to output a binary output object file in the form that corresponds with the PROMs selected. AMPROM allows the user to specify the depth (number of instructions) and the width (number of bits of the microword) for each PROM. (Refer to AmSYS29/10 Microprogram Support Software User's Manual, No. 059910515-003 for further information on the AMPROM support program.) The following procedure should be executed on the object code generated by the AMDASM support program:

a. At the AMD Development System, enter AMPROM O-filename and depress the <return> key. This support program will take the object file "filename.OBJ" produced by the AMDASM assembler and convert it to an output file "filename.OUT" for subsequent PROM programming.

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0  
 ARIES MICRO DEFINITION FILE (PHASE 1)

```

;
; FILE: EFM.DEF
;
; ED MANCUS                               MARCH 27, 1985
;
WORD 56
;
; INPUT DATA PATH SELECT (D-BUS)
;
SELA:    EQU B#0       ;SELECT P.L. REG
SELB:    EQU B#1       ;SELECT D-BUS
;
; AM2910 MICROPROGRAM CONTROLLER
;
; NEXT MICROINSTRUCTION ADDRESS SELECT
;
JZ:      EQU H#0       ;JUMP ZERO
CJS:     EQU H#1       ;COND JSB PL
JMAP:    EQU H#2       ;JUMP MAP
CJP:     EQU H#3       ;COND JUMP PL
PUSH:    EQU H#4       ;PUSH/COND LOAD CNTR
JSRP:    EQU H#5       ;COND JSB R/PL
CJV:     EQU H#6       ;COND JUMP VECTOR
JRP:     EQU H#7       ;COND JUMP R/PL
RFCT:    EQU H#8       ;REPEAT LOOP
RPCT:    EQU H#9       ;REPEAT PL
CRTN:    EQU H#A       ;COND RTN
CJPP:    EQU H#B       ;COND JUMP PL & POP
LOCT:    EQU H#C       ;LD CNTR & CONTINUE
LOOP:    EQU H#D       ;TEST END LOOP
CONT:    EQU H#E       ;CONTINUE
TWB:     EQU H#F       ;THREE-WAY BRANCH
;
; CONDITION CODE ENABLE (/CCEN)
;
COND:    EQU B#0       ;CONDITIONAL TEST
UNCOND:  EQU B#1       ;UNCONDITIONAL PASS
;
; AM2901 4-BIT MICROPROCESSOR SLICE
;
; DESTINATION CONTROL
;
QREG:    EQU Q#0
NP:      EQU Q#1
RAMA:    EQU Q#2
RAMF:    EQU Q#3
RAMQD:   EQU Q#4
RAMD:    EQU Q#5
RAMQU:   EQU Q#6
RAMU:    EQU Q#7
;
; ALU FUNCTION SELECT
;
FADD:    EQU Q#0
SUBR:    EQU Q#1

```

FIGURE B.2.1-1. DEFINITION FILE (Sheet 1 of 4)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0  
 ARIES MICRO DEFINITION FILE (PHASE 1)

```

SUBS: EQU Q#2
FOR: EQU Q#3
FAND: EQU Q#4
NOTRS: EQU Q#5
EXOR: EQU Q#6
EXNOR: EQU Q#7
;
; ALU SOURCE OPERANDS (R S)
;
AQ: EQU Q#0
AB: EQU Q#1
ZQ: EQU Q#2
IB: EQU Q#3
ZA: EQU Q#4
DA: EQU Q#5
DQ: EQU Q#6
DZ: EQU Q#7
;
; REGISTER DEFINITIONS
;
R0: EQU H#0
R1: EQU H#1
R2: EQU H#2
R3: EQU H#3
R4: EQU H#4
R5: EQU H#5
R6: EQU H#6
R7: EQU H#7
R8: EQU H#8
R9: EQU H#9
R10: EQU H#A
R11: EQU H#B
R12: EQU H#C
R13: EQU H#D
R14: EQU H#E
R15: EQU H#F
;
; ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
; ::::: AM2904 STATUS AND SHIFT CONTROL UNIT :::::
; ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
;
; CARRY-IN BIT
;
CIZERO: EQU B#00
CIONE: EQU B#01
;
; SHIFTLINKAGE MUX INSTRUCTION CODE (I9 - I6)
;
NOOP: EQU H#0
SHFTRA: EQU H#0
;
; TEST CONTROL FIELD (I5 - I0)
;
UZ: EQU Q#25 ;MICRO ZERO
UNZ: EQU Q#24 ;MICRO NOT-ZERO
UOVR: EQU Q#27 ;MICRO OVERFLOW
UNOVR: EQU Q#26 ;MICRO NOT-OVERFLOW

```

FIGURE B.2.1-1. DEFINITION FILE (Sheet 2 of 4)

AM00S/29 AMOASM MICRO ASSEMBLER, V2.0  
 ARIES MICRO DEFINITION FILE (PHASE 1)

```

UC:      EQU Q#33           ;MICRO CARRY
UNC:     EQU Q#32           ;MICRO NOT-CARRY
UN:      EQU Q#37           ;MICRO NEGATIVE
UNN:     EQU Q#36           ;MICRO NOT-NEGATIVE
;
; STATUS REGISTER CONTROL FIELD (IS - IO)
;
USRTOMSR: EQU Q#02
MSRTOUSR: EQU Q#02
ITOUSR:   EQU Q#20
;
; MICRO STATUS CONTROL BIT
;
NOMICRO: EQU B#1           ;PRESERVE MICRO STATUS
MICROEN: EQU B#0           ;LATCH NEW MICRO STATUS
;
; MACRO STATUS CONTROL BIT
;
NOMACRO: EQU B#1           ;PRESERVE MACRO STATUS
MACROEN: EQU B#0           ;LATCH NEW MACRO STATUS
;
; TEST AND SHIFT CONTROL BITS
;
NONEON:  EQU B#11          ;NO TEST OR SHIFT
SHIFT:   EQU B#10          ;ENABLE SHIFT OPERATION
TEST:    EQU B#01          ;ENABLE TEST OPERATION
;
;-----;
; COMMAND REG. CONTROL BIT
;
CTRLN:  EQU B#0           ;ENABLE COMMAND REGISTER
NOCTRL: EQU B#1           ;DISABLE COMMAND REGISTER
;
;-----;
SUB1:   SUB 16H*0000,SELA,CONT,COND
SUB2:   SUB NP,FADD,AQ,R0,R0
SUB3:   SUB CIZERO,NOOP,6Q*00,NOMACRO,NOMICRO,NONEON
SUB4:   SUB CIZERO,NOOP,ITOUSR,NOMACRO,MICROEN,NONEON,NOCTRL
SUB5:   SUB 16V:H*,SELA,CJP,COND
SUB6:   SUB 16VX,SELA,CONT,COND
;
;-----;
; MICRO INSTRUCTION LIST
;-----;
NOP:    DEF SUB1,SUB2,SUB3,NOCTRL
LIMCR:  DEF SUB6,NP,FOR,DZ,R0,R0,SUB3,CTRLN
LCR:    DEF SUB1,NP,FOR,ZA,4VX,R0,SUB3,CTRLN
RSTCR:  DEF 16H*00FF,SELA,CONT,COND,NP,FOR,DZ,R0,R0,
/SUB3,CTRLN
IN:     DEF 16H*0000,SELB,CONT,COND,
/RAMF,FADD,DZ,R0,4VX,SUB4
OUT:    DEF SUB1,NP,FOR,ZA,4VX,R0,SUB3,NOCTRL
INQ:    DEF 16H*0000,SELB,CONT,COND,QREG,FADD,DZ,
/R0,R0,SUB4
DECQ:   DEF SUB1,QREG,SUBR,ZQ,R0,R0,SUB4

```

FIGURE B.2.1-1. DEFINITION FILE (Sheet 3 of 4)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0  
ARIES MICRO DEFINITION FILE (PHASE 1)

```
ANDIQ:  DEF SUB6,QREG,FAND,DQ,R0,R0,SUB4
AND:    DEF SUB1,RAMF,FAND,AB,4UX,4UX,SUB4
ADD:    DEF SUB1,RAMF,FADD,AB,4UX,4UX,SUB4
MOV:    DEF SUB1,RAMF,EXOR,ZA,4UX,4UX,SUB4
CLR:    DEF SUB1,RAMF,FAND,ZB,R0,4UX,SUB4
COM:    DEF SUB1,RAMF,EXNOR,ZB,R0,4UX,SUB4
DEC:    DEF SUB1,RAMF,SUBR,ZB,R0,4UX,SUB4
SRA:    DEF SUB1,RAMD,FOR,ZB,R0,4UX,SUB4
LIM:    DEF SUB6,RAMF,EXOR,DZ,R0,4UX,SUB4
ADDI:   DEF SUB6,RAMF,FADD,DA,4UX,4UX,SUB4
JSR:    DEF 16V%:H#,SELA,CJS,UNCOND,SUB2,CIZERO,
/NOOP,USRTOMSR,MACROEN,NOMICRO,NONEON,NOCTRL
RTN:    DEF 16H#0000,SELA,CRTN,UNCOND,SUB2,CIZERO,
/NOOP,MSRTOUSR,NOMACRO,MICROEN,NONEON,NOCTRL
JMP:    DEF 16V%:H#,SELA,CJP,UNCOND,SUB2,SUB3,NOCTRL
BNZ:    DEF SUB5,SUB2,CIZERO,NOOP,UNZ,NOMACRO,
/MICROEN,TEST,NOCTRL
BRZ:    DEF SUB5,SUB2,CIZERO,NOOP,UZ,NOMACRO,
/MICROEN,TEST,NOCTRL
BRC:    DEF SUB5,SUB2,CIZERO,NOOP,UC,NOMACRO,
/MICROEN,TEST,NOCTRL
ANDI:   DEF SUB6,RAMF,FAND,DA,4UX,4UX,SUB4
JPZ:    DEF 16V%:H#0000,SELA,JZ,COND,SUB2,SUB3,NOCTRL
;
END
```

TOTAL PHASE 1 ERRORS = 0

FIGURE B.2.1-1. DEFINITION FILE (Sheet 4 of 4)

b. The program will then prompt the user for information related to the PROMs which will hold the microcode. This sequence is shown below.

```
DON'T CARES? 0 <return>
ENTER PROM WIDTHS: 8 <return>
ENTER PROM DEPTHS: 512 <return>
*** ERROR 6: (WARNING) DEPTH EXCEEDS MAXIMUM PC
```

Note the last statement is notifying the user that the program count (number of instructions) is not equal to the number of PROM locations specified.

c. The user response to the next prompt as follows:

```
WHICH PROMS DO YOU WISH TO PRINT? 1-7 <return>
```

d. The action shown below should be observed on the display. The program is completed when the system prompt returns on the screen.

```
PROM 1 STARTING PC - 0000 512 WORDS
PROM 2 STARTING PC - 0000 512 WORDS
PROM 3 STARTING PC - 0000 512 WORDS
PROM 4 STARTING PC - 0000 512 WORDS
PROM 5 STARTING PC - 0000 512 WORDS
PROM 6 STARTING PC - 0000 512 WORDS
PROM 7 STARTING PC - 0000 512 WORDS
```

A>

### B.2.3 AMD-to-ARIES Data Transfer Procedure.

a. At an available Concurrent 3230 MTM Terminal, under any account number, enter; READAMD and depress the <return> key. This command loads and executes a special program which sets up seven files to store the microcode data. These files will be formatted to hold 8 X 512 bit arrays which match the size of the PROMs selected to hold the microcode routines.

b. The program will prompt the user for a filename for the anticipated data entry. The prompt will appear on the screen as ENTER FILENAME. At this prompt, the user will enter a maximum of six characters for the label followed by depressing the <return> key. No suffix (extension) is entered. At this point, the READAMD program is ready to accept data from the AMD Development System.

c. At the AMD Development System, insert the floppy disk containing the PROM files into disk drive B.

d. From the AMD terminal enter; PIP PUN:-filename.OUT and depress the <return> key. The "filename" is the actual data file desired.

e. Data transfer will require about 5 minutes to complete. As each PROM data file is filled, the READAMD program will attach a number, 1 through 7, after the filename plus the extension ".PRM." After all of the data has been received, the prompt "DATA TRANSFER COMPLETE" will appear on the MTM display. If any other message appears, the data transfer was not successful. Steps (a) through (d) should be repeated.

## B.3 MICROCODE PROGRAMS.

### B.3.1 MRG Microcode Program.

This microcode program, resident in the Modelled Reply Controller (MRC) read only memory (ROM), provides five modes of operation for the MRC: (1) normal operation, (2) MRG Interface Loopback Validation, (3) MRC Validation, (4) Modelled ARIES Target (MAT) Validation, and (5) MAT RF Validation. These routines are illustrated by the flow charts presented in figure B.3.1-1. There are four ways to initialize the MRC microcode program: (1) system clear (CLR0) generated when the ARIES system is first powered up, (2) Central Processor Unit (CPU) controlled reset command to the MRG, (3) depressing the manual reset button on the MRC digital board, and (4) via the program itself whenever an error or fault is detected. Upon initialization, the MRC resets all the MAT reply generators.

The CPU selects the mode of operation by setting the appropriate value in the operation mode field of the MRG command. (Refer to the ARIES Hardware Maintenance Manual, Volume II, Appendix D for additional information on the MRG command format.) Table B.3.1-1 lists the select mode options.

TABLE B.3.1-1. MRG MODE SELECTION FIELD

Mode (Oct)	Routine Selection
0	Normal Operation
1	MAT RF Validation
2	MAT Validation
3	MRC Validation
4 - 7	MRG Interface Loopback Validation

A printout of the MRG Microcode program is presented in figure B.3.1-2. This is the AMDASM Phase II output listing, consisting of 15 pages. All of the MRC commands are presented on page 1. Note the commands to a fourth MAT. The reason for this is that all of the digital reply generator boards were designed to be interchangeable. Each board contains two independent reply generators. Therefore, the digital logic for a fourth MAT exists, but is not used. However, since the reply generator boards are interchangeable, all of the digital logic on the reply generator boards must be verified.

When the operation mode selected by the CPU is normal, the MRC proceeds to sequence replies to the MATs in MAT-1, MAT-2, and MAT-3 order. (See pages 3 and 4, titled "MAIN PROGRAM," and "LOAD MATs.") Two conditions must be met before a reply is transferred: (1) a reply must be available in the MRG Buffer Interface, and (2) the MAT must be ready to accept a reply. The number words transferred in a reply depends upon the type: 4 words for Air Traffic Control Radar Beacon System (ATCRBS) and 10 words for Mode S. After each reply is sent, the reply counter in the MRG Buffer Interface is decremented.

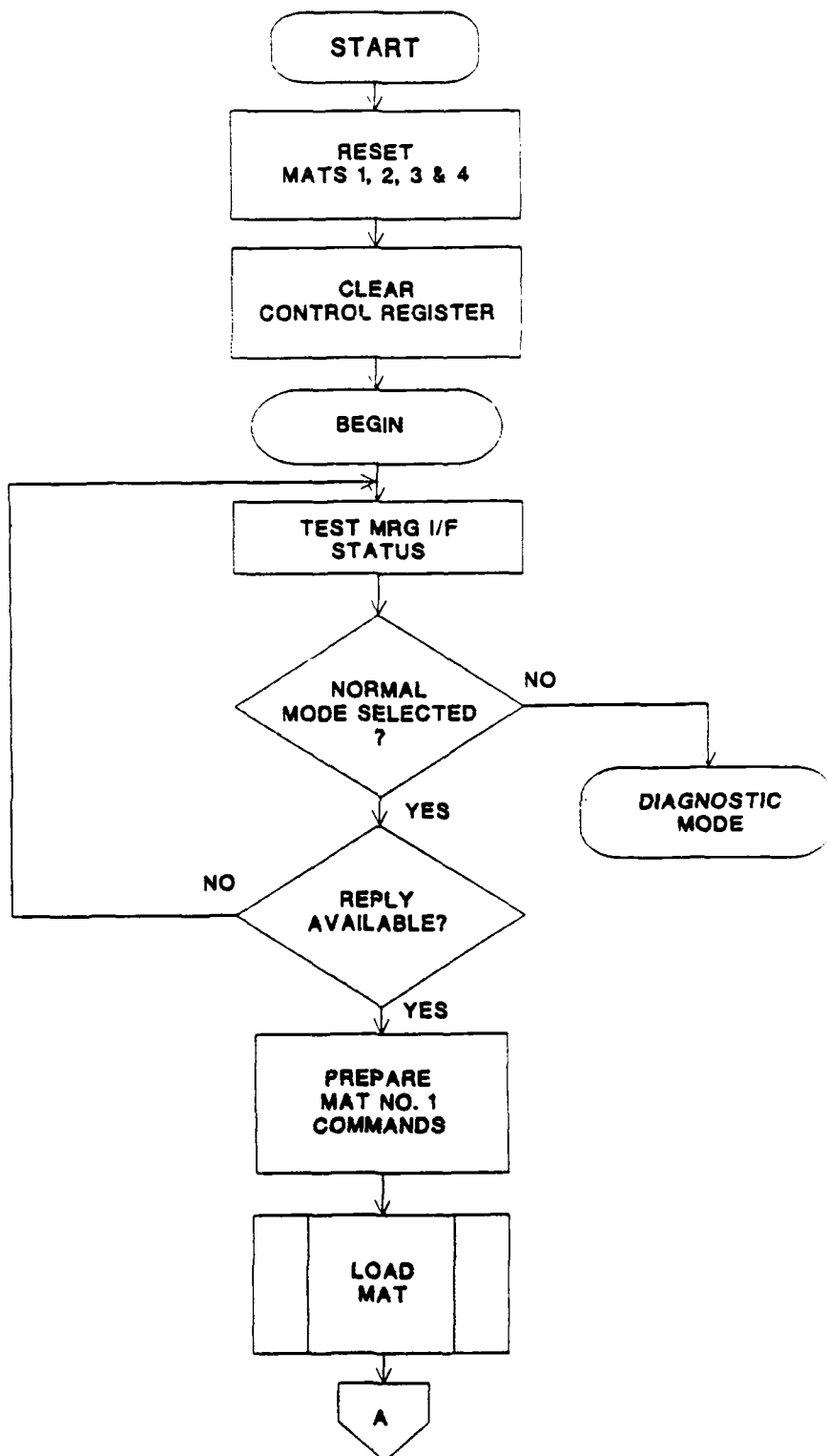


FIGURE B.3.i-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 1 of 9)



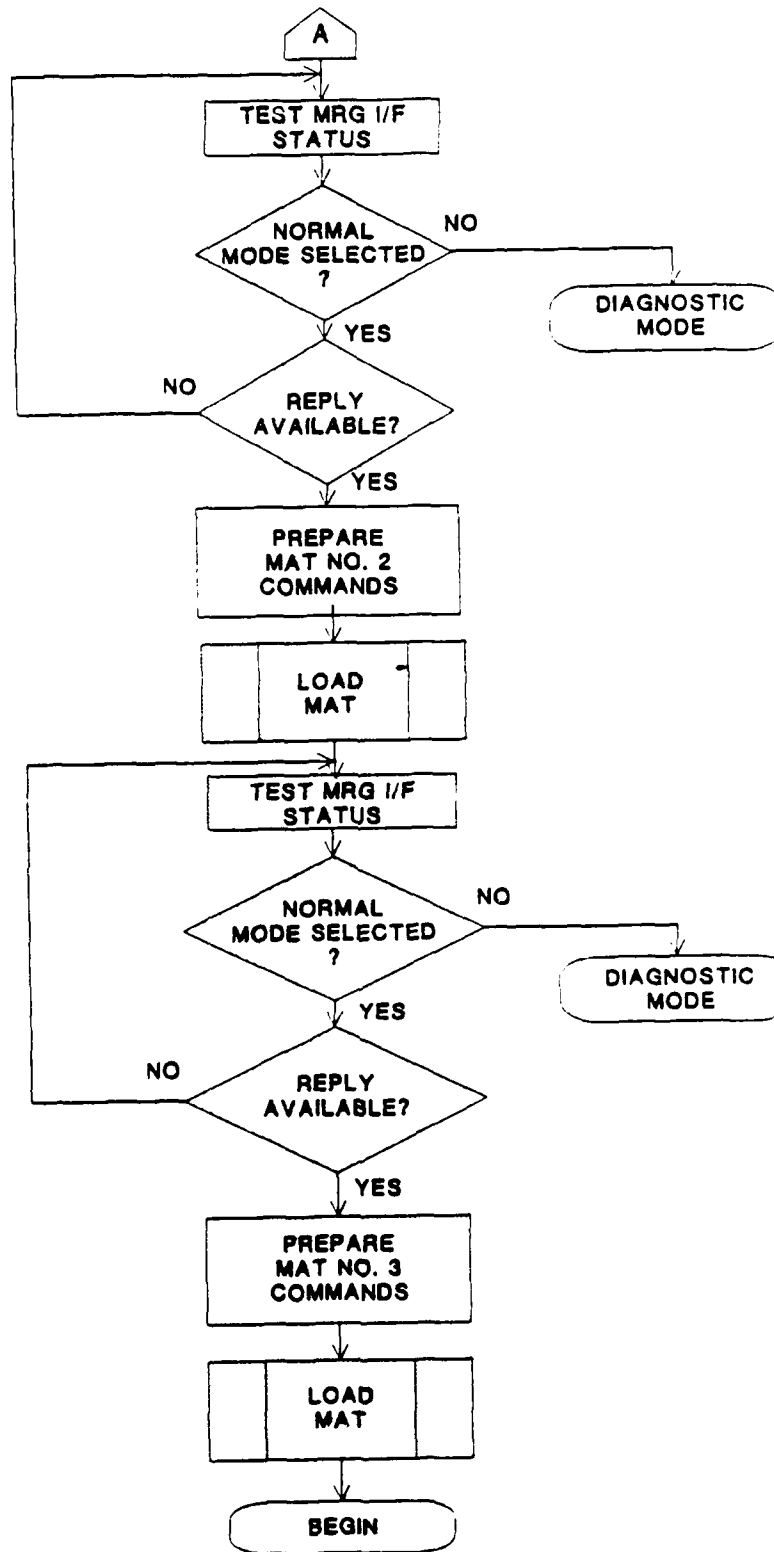


FIGURE B.3.1-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 2 of 9)

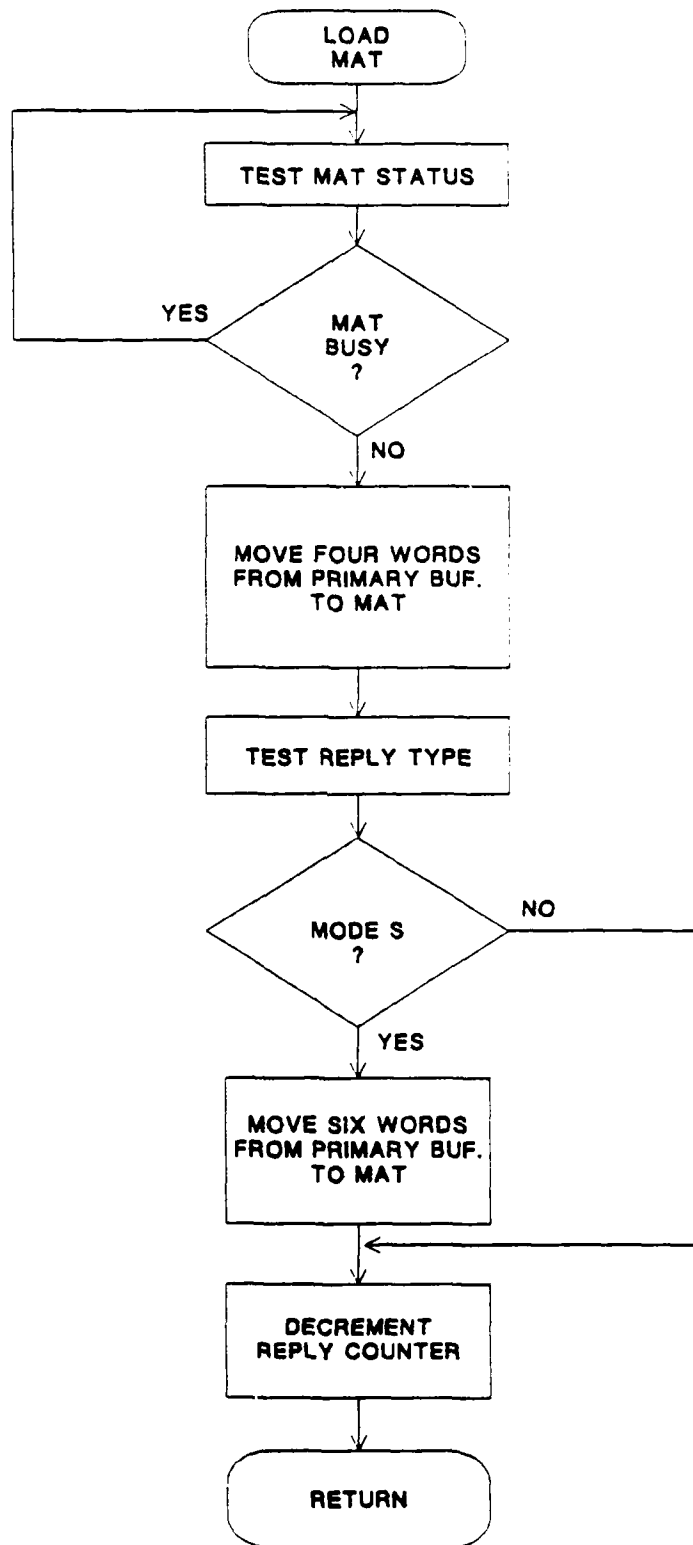
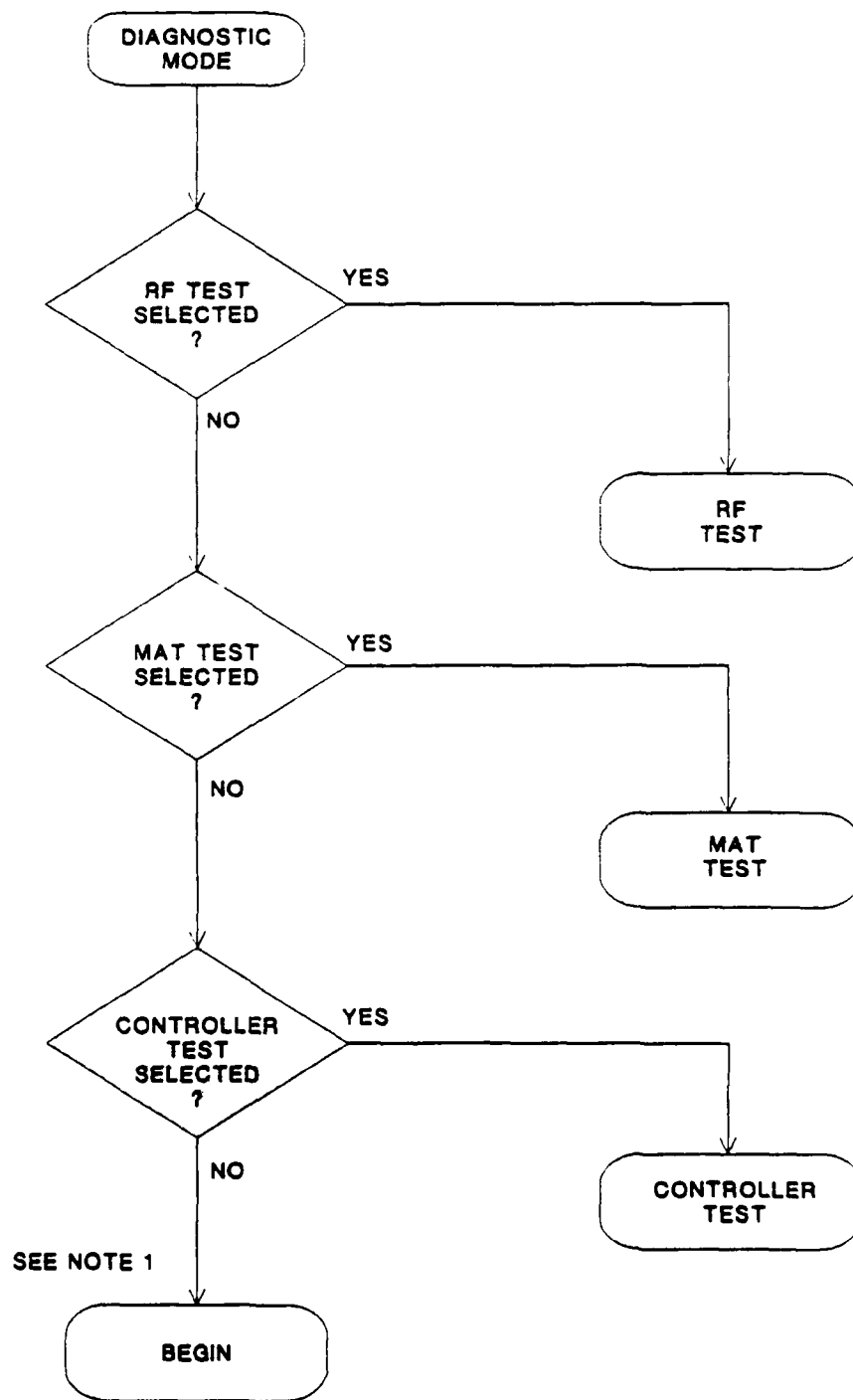


FIGURE B.3.1-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 3 of 9)



**NOTE 1: IF THE MICROCODE REACHES THIS POINT, THE MRG INTERFACE LOOPBACK TESTS ARE SELECTED.**

**FIGURE B.3.1-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 4 of 9)**

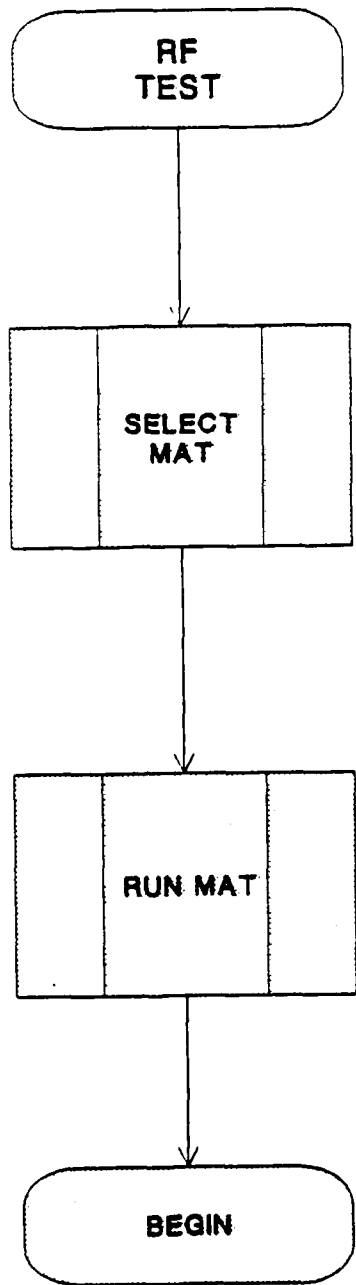


FIGURE B.3.1-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 5 of 9)

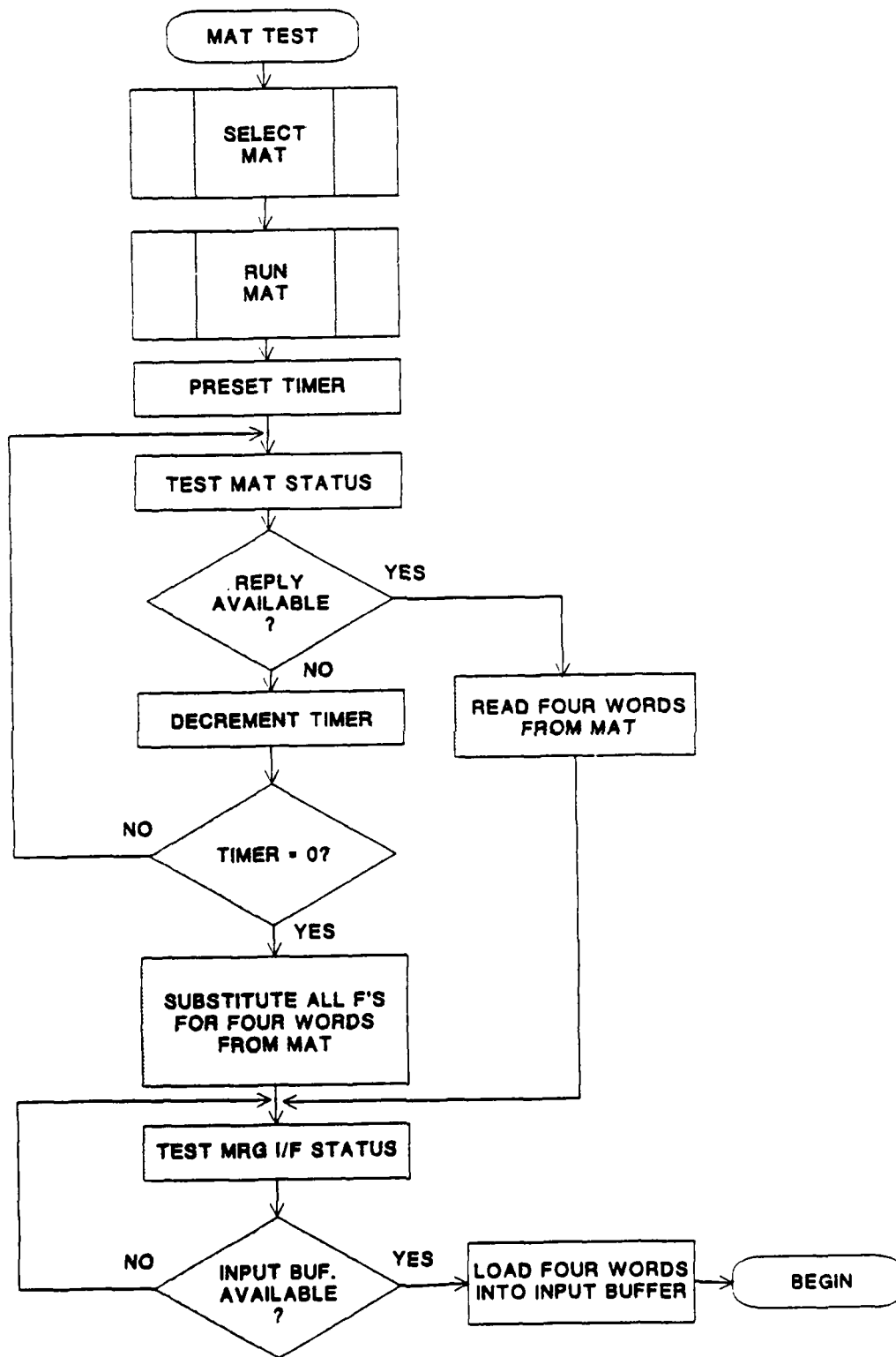


FIGURE B.3.1-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 6 of 9)

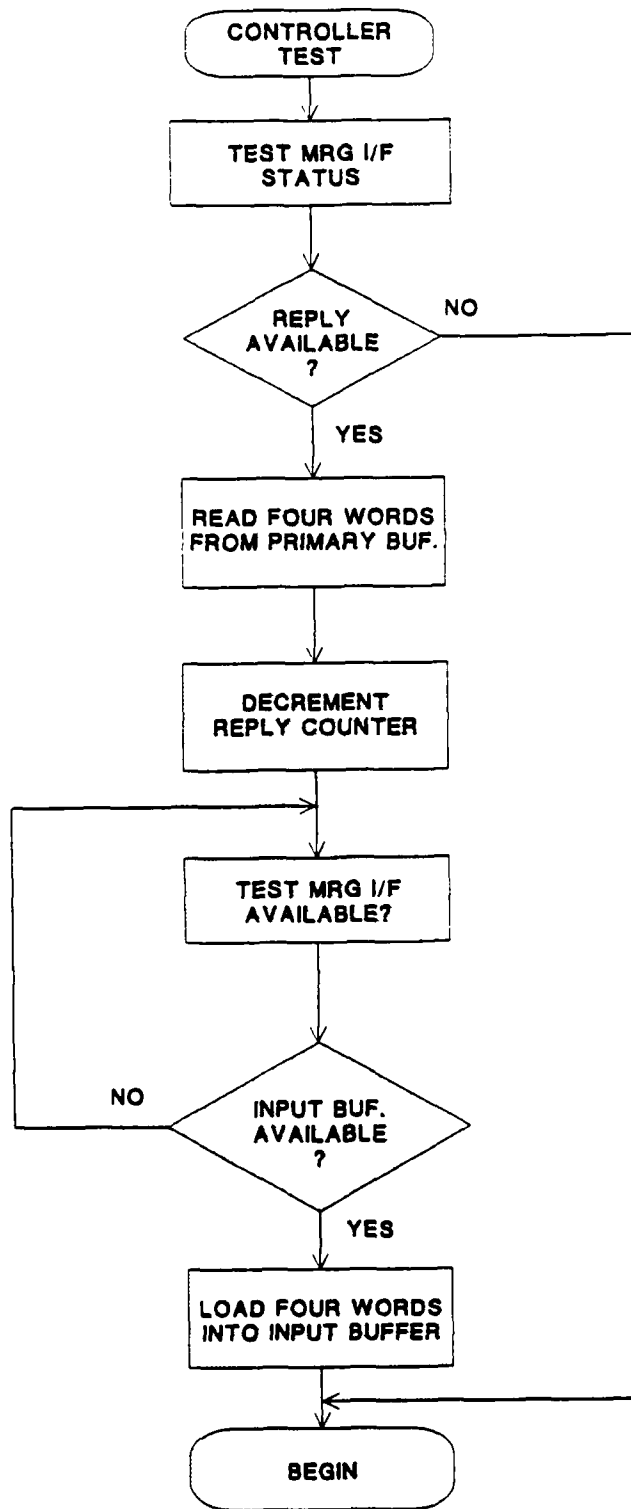


FIGURE B.3.1-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 7 of 9)

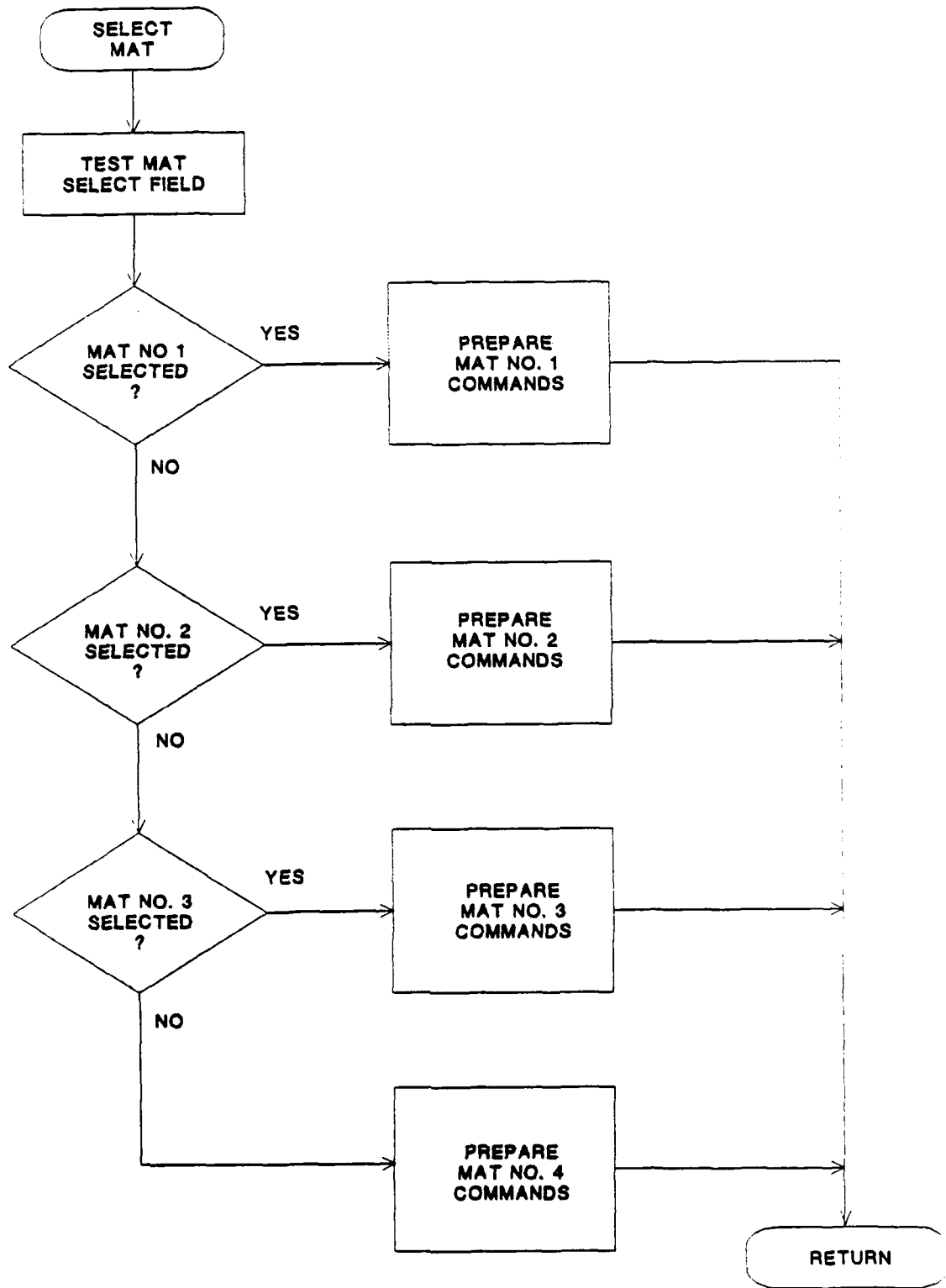


FIGURE B.3.1-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 8 of 9)

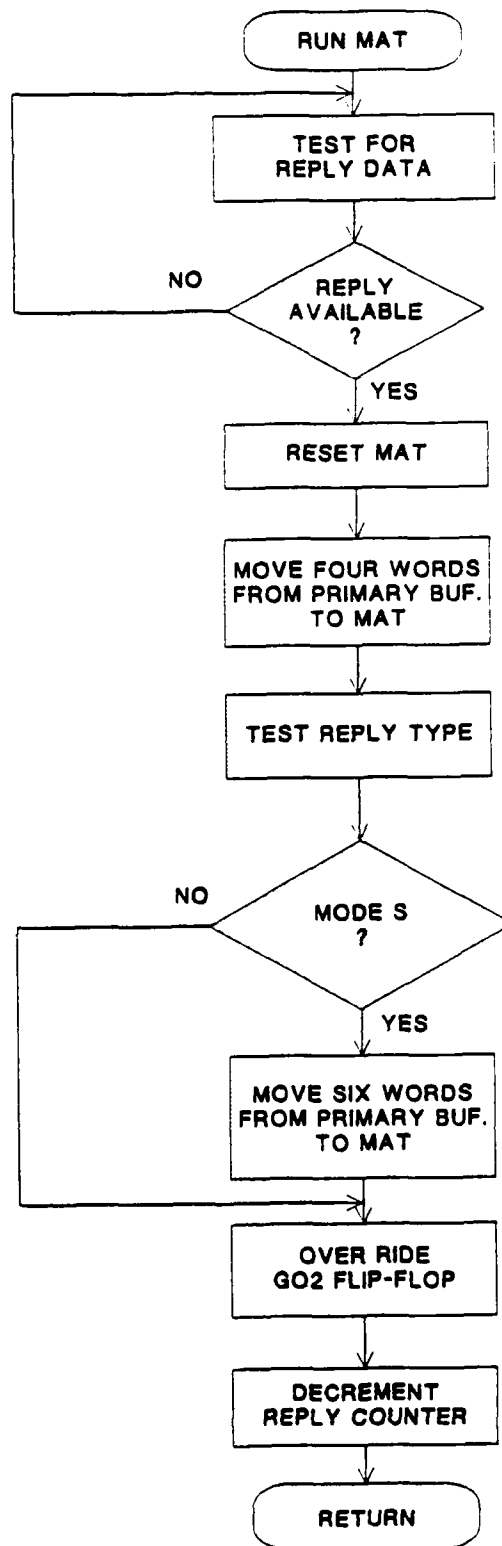


FIGURE B.3.1-1. MRG MICROCODE FLOW DIAGRAMS (Sheet 9 of 9)



```

: MICRO PROCESSOR COMMANDS
:
:           :          :UNIT: OP :
:           :          : ID :CODE:  DESCRIPTION
:-----
: MAT#1    :MAT1STAT: 0 : 0 : READ MAT#1 STATUS
:           :MAT1RST: 0 : 1 : RESET MAT#1
:           :MAT1LD4: 0 : 2 : LOAD 4 WORDS INTO MAT#1
:           :MAT1LD6: 0 : 3 : LOAD 6 WORDS INTO MAT#1
:           :MAT1RD4: 0 : 4 : READ 4 WORDS FROM MAT#1
:           :MAT1CW:  0 : 5 : MAT#1 CW ON
:           :MAT1NCW: 0 : 6 : MAT#1 CW OFF
:           :MAT1DIAG: 0 : 7 : MAT#1 DIAGNOSTIC MODE
:
: MAT#2    :MAT2STAT: 1 : 0 : READ MAT#2 STATUS
:           :MAT2RST: 1 : 1 : RESET MAT#2
:           :MAT2LD4: 1 : 2 : LOAD 4 WORDS INTO MAT#2
:           :MAT2LD6: 1 : 3 : LOAD 6 WORDS INTO MAT#2
:           :MAT2RD4: 1 : 4 : READ 4 WORDS FROM MAT#2
:           :MAT2CW:  1 : 5 : MAT#2 CW ON
:           :MAT2NCW: 1 : 6 : MAT#2 CW OFF
:           :MAT2DIAG: 1 : 7 : MAT#2 DIAGNOSTIC MODE
:
: MAT#3    :MAT3STAT: 2 : 0 : READ MAT#3 STATUS
:           :MAT3RST: 2 : 1 : RESET MAT#3
:           :MAT3LD4: 2 : 2 : LOAD 4 WORDS INTO MAT#3
:           :MAT3LD6: 2 : 3 : LOAD 6 WORDS INTO MAT#3
:           :MAT3RD4: 2 : 4 : READ 4 WORDS FROM MAT#3
:           :MAT3CW:  2 : 5 : MAT#3 CW ON
:           :MAT3NCW: 2 : 6 : MAT#3 CW OFF
:           :MAT3DIAG: 2 : 7 : MAT#3 DIAGNOSTIC MODE
:
: MAT#4    :MAT4STAT: 3 : 0 : READ MAT#4 STATUS
:           :MAT4RST: 3 : 1 : RESET MAT#4
:           :MAT4LD4: 3 : 2 : LOAD 4 WORDS INTO MAT#4
:           :MAT4LD6: 3 : 3 : LOAD 6 WORDS INTO MAT#4
:           :MAT4RD4: 3 : 4 : READ 4 WORDS FROM MAT#4
:           :MAT4CW:  3 : 5 : MAT#4 CW ON
:           :MAT4NCW: 3 : 6 : MAT#4 CW OFF
:           :MAT4DIAG: 3 : 7 : MAT#4 DIAGNOSTIC MODE
:
: MRG INTERFACE
:IFSTATUS: 6 : 1 : READ MRG I/F STATUS
:READ4 : 6 : 2 : READ 4 WORDS FROM MRG I/F
:READ6 : 6 : 3 : READ 6 WORDS FROM MRG I/F
:LOAD4 : 6 : 4 : LOAD 4 WORDS INTO MRG I/F
:DECRC : 6 : 5 : DECREMENT REPLY COUNTER
:MRGATN : 6 : 7 : ISSUE MRG INTERRUPT
:
:
0000 MAT1STAT: EQU H#0000
0010 MAT2STAT: EQU H#0010
0020 MAT3STAT: EQU H#0020
0030 MAT4STAT: EQU H#0030
0001 MAT1RST: EQU H#0001
0011 MAT2RST: EQU H#0011

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 1 of 14)

```

0021 MAT3RST: EQU H#0021
0031 MAT4RST: EQU H#0031
0002 MAT1LD4: EQU H#0002
0012 MAT2LD4: EQU H#0012
0022 MAT3LD4: EQU H#0022
0032 MAT4LD4: EQU H#0032
0003 MAT1LD6: EQU H#0003
0013 MAT2LD6: EQU H#0013
0023 MAT3LD6: EQU H#0023
0033 MAT4LD6: EQU H#0033
0004 MAT1RD4: EQU H#0004
0014 MAT2RD4: EQU H#0014
0024 MAT3RD4: EQU H#0024
0034 MAT4RD4: EQU H#0034
0005 MAT1CW: EQU H#0005
0015 MAT2CW: EQU H#0015
0025 MAT3CW: EQU H#0025
0035 MAT4CW: EQU H#0035
0006 MAT1NCW: EQU H#0006
0016 MAT2NCW: EQU H#0016
0026 MAT3NCW: EQU H#0026
0036 MAT4NCW: EQU H#0036
0007 MAT1DIAG: EQU H#0007
0017 MAT2DIAG: EQU H#0017
0027 MAT3DIAG: EQU H#0027
0037 MAT4DIAG: EQU H#0037
0061 IFSTATUS: EQU H#0061
0062 READ4: EQU H#0062
0063 READ6: EQU H#0063
0064 LOAD4: EQU H#0064
0065 DECRC: EQU H#0065
0067 MRGATN: EQU H#0067

```

.....

```

;
; REGISTER ALLOCATIONS
;

```

```

; R0 - R5 = RESERVED FOR DATA TRANSFER
; R6 = GENERAL PURPOSE REGISTER
; R7 - R14 = RESERVED FOR MAT COMMANDS
; R15 = GENERAL PURPOSE REGISTER
;

```

```

;
; MR6 I/F STATUS & COMMAND MASKS
;

```

```

0007 NORMASK: EQU H#0007
0003 GOMASK: EQU H#0003
8000 PEMASK: EQU H#8000
4000 RPYAVAIL: EQU H#4000
1000 CWMSK: EQU H#1000

```

```

;
; OTHER THINGS
;

```

```

DAC0 TIMER: EQU H#DAC0 ;70 MILLISECOND TIMEOUT
000E TYPE: EQU H#000E ;REPLY PARAMETER TYPE MASK
0002 RPYDATA: EQU H#0002 ;MAT DATA AVAIL MASK
0001 BUSY: EQU H#0001 ;MAT BUSY MASK

```

.....

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 2 of 14)

```

: : : : : INITIALIZATION : : : : :
: : : : :
0000 ORG 0
0000 INIT: LIMCR MAT1RST ;INITIALIZE ALL MODELLED ARIES
0001 LIMCR MAT2RST ;TARGET (MAT) GENERATORS.
0002 LIMCR MAT3RST
0003 LIMCR MAT4RST
0004 RSTCR ;INITIALIZE CONTROL REGISTER.
: : : : :
: : : : : MAIN PROGRAM : : : : :
: : : : :
0005 BEGIN: LIMCR IFSTATUS ;LOAD THE "I/F CMD & STATUS" WORD
0006 NOP ;INTO THE Q-REG. TEST THE OPER-
0007 LOOPA: INQ ;ATION FIELD TO DETERMINE WHETHER
0008 ANDIQ NORMASK ;TO RUN THE NORMAL OR THE
0009 BNZ DIAG ;DIAGNOSTIC MODE.
;
000A INQ ;DETERMINE IF A REPLY IS AVAILABLE
000B ANDIQ RPYAVAIL ;IN THE PRIMARY BUFFER. IF NOT,
000C BRZ LOOPA ;CHECK OPERATING MODE AGAIN.
;
000D MAT1: LIM MAT1LD4,R12 ;SET UP MAT1 COMMANDS.
000E LIM MAT1LD6,R13
000F LIM MAT1STAT,R14
0010 JSR LOADMAT
;
0011 LIMCR IFSTATUS ;LOAD THE "I/F CMD & STATUS" WORD
0012 NOP ;INTO THE Q-REG. TEST THE OPER-
0013 LOOPB: INQ ;ATION FIELD TO DETERMINE WHETHER
0014 ANDIQ NORMASK ;TO RUN IN THE NORMAL OR THE
0015 BNZ DIAG ;DIAGNOSTIC MODE.
;
0016 INQ ;DETERMINE IF A REPLY IS AVAILABLE
0017 ANDIQ RPYAVAIL ;IN THE PRIMARY BUFFER. IF NOT,
0018 BRZ LOOPB ;CHECK OPERATING MODE AGAIN.
;
0019 MAT2: LIM MAT2LD4,R12 ;SET UP MAT2 COMMANDS.
001A LIM MAT2LD6,R13
001B LIM MAT2STAT,R14
001C JSR LOADMAT
;
001D LIMCR IFSTATUS ;LOAD THE "I/F CMD & STATUS" WORD
001E NOP ;INTO THE Q-REG. TEST THE OPER-
001F LOOPC: INQ ;ATION FIELD TO DETERMINE WHETHER
0020 ANDIQ NORMASK ;TO RUN IN THE NORMAL OR THE
0021 BNZ DIAG ;DIAGNOSTIC MODE.
;
0022 INQ ;DETERMINE IF A REPLY IS AVAILABLE
0023 ANDIQ RPYAVAIL ;IN THE PRIMARY BUFFER. IF NOT,
0024 BRZ LOOPC ;CHECK OPERATING MODE AGAIN.
;
0025 MAT3: LIM MAT3LD4,R12 ;SET UP MAT3 COMMANDS.
0026 LIM MAT3LD6,R13
0027 LIM MAT3STAT,R14
0028 JSR LOADMAT
0029 JMP BEGIN
: : : : :

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 3 of 14)

```

; SUBROUTINE "LOAD MAT'S"
; THIS ROUTINE FIRST CHECKS TO SEE IF THE MAT IS ABLE TO
; RECEIVE A REPLY PARAMETER. WHEN THE MAT BECOMES AVAILABLE
; A REPLY PARAMETER IS READ FROM THE MRG PRIMARY BUFFER
; AND LOADED INTO THAT MAT. AFTER DATA TRANSFER IS COMPLETE
; THE PRIMARY BUFFER REPLY COUNTER IS DECREMENTED.
;
002A LOADMAT: LCR R14 ;TEST BUSY STATUS OF THE MAT.
002B NOP
002C NOP
002D LOOPD: INQ ;LOOP UNTIL THE MAT IS NOT BUSY.
002E ANDIQ BUSY
002F BNZ LOOPD
;
0030 LIMCR READ4 ;READ FOUR WORDS FROM THE MRG
0031 RSTCR ;PRIMARY BUFFER.
0032 NOP
0033 IN R0
0034 IN R1
0035 IN R2
0036 IN R3
;
0037 LCR R12 ;LOAD THE MAT WITH FOUR WORDS.
0038 OUT R0
0039 OUT R1
003A OUT R2
003B OUT R3
003C RSTCR
;
003D ANDI TYPE,R1,R1 ;CHECK WHETHER THE REPLY PARAMETER
003E BNZ MREPLY ;IS MODE S TYPE. IF SO, CONTINUE
; ;LOADING REPLY PARAMETER. IF NOT,
003F LIMCR DECR4 ;ATCRBS TRANSFER IS COMPLETE.
0040 RSTCR ;DECREMENT REPLY COUNTER AND RETURN.
0041 RTN
;
0042 MREPLY: LIMCR READ6 ;READ THE NEXT SIX WORDS FROM THE
0043 RSTCR ;MRG PRIMARY BUFFER.
0044 NOP
0045 IN R0
0046 IN R1
0047 IN R2
0048 IN R3
0049 IN R4
004A IN R5
;
004B LCR R13 ;LOAD THE MAT WITH WORDS 5 THRU 10.
004C OUT R0
004D OUT R1
004E OUT R2
004F OUT R3
0050 OUT R4
0051 OUT R5
;
0052 LIMCR DECR4 ;DECREMENT REPLY COUNTER AND RETURN.
0053 RSTCR

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 4 of 14)



```

007D      LIMCR LOAD4      ;WHEN THE PE-RAM BECOMES AVAILABLE
007E      RSTCR           ;TRANSFER DIAGNOSTIC DATA TO IT.
007F      OUT R0
0080      OUT R1
0081      OUT R2
0082      OUT R3
0083      JMP BEGIN
;
; :::::::::::::::::::: MICROPROCESSOR LOOPBACK ROUTINE ::::::::::::::::::::
;
0084 UPRCDIAG: LIMCR IFSTATUS ;CHECK FOR A FOUR WORD BLOCK IN THE
0085          NOP           ;PRIMARY BUFFER.
0086          INQ
0087          ANDIQ RPYAVAIL
0088          BRZ BEGIN
;
0089          LIMCR READ4   ;READ THE FOUR WORD BLOCK FROM
008A          RSTCR       ;THE PRIMARY BUFFER.
008B          NOP
008C          IN R0
008D          IN R1
008E          IN R2
008F          IN R3
;
0090          LIMCR DECRC   ;DECREMENT PRIMARY BUFFER REPLY COUNTER.
0091          RSTCR
;
0092          LIMCR IFSTATUS ;LOAD THE "I/F STATUS & COMMAND" WORD
0093          NOP           ;INTO THE Q-REG. DETERMINE WHETHER
0094 LOOPK:   INQ           ;THE PE-RAM IS AVAILABLE FOR RECEIVING
0095          ANDIQ PEMASK  ;DATA. CONTINUE CHECKING UNTIL IT
0096          BRZ LOOPK    ;BECOMES AVAILABLE.
;
0097          LIMCR LOAD4   ;LOAD THE FOUR WORD BLOCK INTO
0098          RSTCR       ;THE PE-RAM.
0099          OUT R0
009A          OUT R1
009B          OUT R2
009C          OUT R3
009D          JMP BEGIN
;
; :::::::::::::::::::: SUBROUTINE "SELECT MAT" ::::::::::::::::::::
; THIS ROUTINE IS USED BY THE "MRGLOOP" TEST AND THE "REPLY
; GENERATOR LOOPBACK" TEST. IT IDENTIFIES WHICH MAT WAS
; SELECTED AND SETS UP ALL THE APPROPRIATE COMMANDS FOR THAT MAT.
; ::::::::::::::::::::
009E SELMAT: LIMCR IFSTATUS
009F          NOP
00A0          IN R6           ; R6 = :X:X:X:0:0:0:0:0:0:0:0:G:0:0:0:
00A1          SRA R6         ; R6 = :0:X:X:X:0:0:0:0:0:0:0:0:G:0:0:0:
00A2          SRA R6         ; R6 = :0:0:X:X:X:0:0:0:0:0:0:0:0:G:0:0:0:
00A3          SRA R6         ; R6 = :0:0:0:X:X:X:0:0:0:0:0:0:0:0:G:0:0:0:
00A4          ANDI 60MASK,R6 ; R6 = :0:0:0:0:0:0:0:0:0:0:0:0:0:0:G:0:0:
00A5          BRZ SELMAT1
00A6          DEC R6
00A7          BRZ SELMAT2
00A8          DEC R6

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 6 of 14)

```

00A9          BRZ SELMAT3
;
00AA SELMAT4: LIM MAT4RST,R9 ;SET UP TO EXECUTE MAT4 COMMANDS.
00AB          LIM MAT4DIAG,R10
00AC          LIM MAT4LD4,R11
00AD          LIM MAT4LD6,R12
00AE          LIM MAT4RD4,R13
00AF          LIM MAT4STAT,R14
00B0          LIM MAT4CW,R8
00B1          LIM MAT4NCW,R7
00B2          RTN
;
00B3 SELMAT3: LIM MAT3RST,R9 ;SET UP TO EXECUTE MAT3 COMMANDS.
00B4          LIM MAT3DIAG,R10
00B5          LIM MAT3LD4,R11
00B6          LIM MAT3LD6,R12
00B7          LIM MAT3RD4,R13
00B8          LIM MAT3STAT,R14
00B9          LIM MAT3CW,R8
00BA          LIM MAT3NCW,R7
00BB          RTN
;
00BC SELMAT2: LIM MAT2RST,R9 ;SET UP TO EXECUTE MAT2 COMMANDS.
00BD          LIM MAT2DIAG,R10
00BE          LIM MAT2LD4,R11
00BF          LIM MAT2LD6,R12
00C0          LIM MAT2RD4,R13
00C1          LIM MAT2STAT,R14
00C2          LIM MAT2CW,R8
00C3          LIM MAT2NCW,R7
00C4          RTN
;
00C5 SELMAT1: LIM MAT1RST,R9 ;SET UP TO EXECUTE MAT1 COMMANDS.
00C6          LIM MAT1DIAG,R10
00C7          LIM MAT1LD4,R11
00C8          LIM MAT1LD6,R12
00C9          LIM MAT1RD4,R13
00CA          LIM MAT1STAT,R14
00CB          LIM MAT1CW,R8
00CC          LIM MAT1NCW,R7
00CD          RTN

;
; SUBROUTINE "LOAD MAT"
; THIS ROUTINE FIRST TESTS TO DETERMINE WHETHER A REPLY PARAMETER
; IS AVAILABLE IN THE MRG PRIMARY BUFFER. WHEN ONE BECOMES
; AVAILABLE, IT IS READ FROM THE BUFFER AND LOADED INTO THE
; SELECTED MODELLED ARIES TARGET (MAT) GENERATOR.
;
00CE DIAGMAT: LIMCR IFSTATUS ;CHECK FOR A REPLY PARAMETER BLOCK
00CF          NOP ;IN THE MRG PRIMARY BUFFER.
00D0 LOOPE:   INQ
00D1          ANDIQ RPYAVAIL
00D2          BRZ LOOPE ;LOOP UNTIL A REPLY BLOCK BECOMES
; AVAILABLE.
00D3          LCR R9 ;RESET SELECTED MODELLED ARIES
00D4          RSTCR ;TARGET GENERATOR.
;

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 7 of 14)

```

0005      LIMCR IFSTATUS      ;LOAD THE "I/F CMD-STAT" WORD
0006      NOP                  ;INTO THE Q-REG.  DETERMINE WHETHER
0007      INQ                  ;THE REPLY GEN. OUTPUT IS TO BE A
0008      ANDIQ CWMSK          ;CODE TRAIN OR CW SIGNAL.
0009      BRZ CWOFF
000A      LCR R8                ;SET CW OUTPUT ON SELECTED
000B      RSTCR                ;REPLY GENERATOR.
000C      JMP JUMP1
000D CWOFF:  LCR R7                ;SET CODE TRAIN OUTPUT ON SELECTED
000E      RSTCR                ;REPLY GENERATOR.
000F JUMP1:  LIMCR READ4        ;READ FOUR WORDS FROM THE MRG
00E0      RSTCR                ;PRIMARY BUFFER.
00E1      NOP                  ;NOP REQUIRED FOR PROPER H/W TIMING.
00E2      IN R0
00E3      IN R1
00E4      IN R2
00E5      IN R3
;
00E6      LCR R11              ;LOAD THE MAT WITH WORDS 1-4.
00E7      OUT R0
00E8      OUT R1
00E9      OUT R2
00EA      OUT R3
00EB      RSTCR
;
00EC      ANDI TYPE,R1,R1      ;CHECK WHETHER THE REPLY PARAMETER
00ED      BNZ DMREPLY          ;IS MODE S TYPE.  IF SO, CONTINUE
;                                ;LOADING.  OTHERWISE RETURN.
00EE      LCR R10              ;SET /MATDIAG BIT TO OVERRIDE G02 F/F.
00EF      LIMCR DECRC          ;DECREMENT MRG I/F REPLY COUNTER.
00F0      RSTCR
00F1      RTN
;
00F2 DMREPLY: LIMCR READ6        ;READ THE NEXT SIX WORDS FROM THE MRG
00F3      RSTCR                ;PRIMARY BUFFER.
00F4      NOP
00F5      IN R0
00F6      IN R1
00F7      IN R2
00F8      IN R3
00F9      IN R4
00FA      IN R5
;
00FB      LCR R12              ;LOAD THE MAT WITH WORDS 5 THRU 10.
00FC      OUT R0
00FD      OUT R1
00FE      OUT R2
00FF      OUT R3
0100      OUT R4
0101      OUT R5
0102      LCR R10              ;SET /MATDIAG BIT TO OVERRIDE G02 F/F.
0103      LIMCR DECRC          ;DECREMENT MRG I/F REPLY COUNTER.
0104      RSTCR
0105      RTN
;
0106 END

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 8 of 14)



```

0000 0000000000000001 0111000010111110 0000000000000000 00011110
0001 0000000000010001 0111000010111110 0000000000000000 00011110
0002 0000000000100001 0111000010111110 0000000000000000 00011110
0003 0000000000110001 0111000010111110 0000000000000000 00011110
0004 0000000011111111 0111000010111110 0000000000000000 00011110
0005 0000000001100001 0111000010111110 0000000000000000 00011110
0006 0000000000000000 0111000010000000 0000000000000000 00011111
0007 0000000000000000 1111000000001110 0000000000000010 00010111
0008 0000000000000111 0111000010011000 0000000000000010 00010111
0009 0000000001010101 0001100010000000 0000000000000010 10010011
000A 0000000000000000 1111000000001110 0000000000000010 00010111
000B 0100000000000000 0111000010011000 0000000000000010 00010111
000C 0000000000000111 0001100010000000 0000000000000010 10110011
000D 0000000000000010 0111000111101110 0001100000000010 00010111
000E 0000000000000011 0111000111101110 0001100000000010 00010111
000F 0000000000000000 0111000111101110 0001110000000010 00010111
0010 0000000000101010 0000110010000000 0000000000000000 01001111
0011 0000000001100001 0111000010111110 0000000000000000 00011110
0012 0000000000000000 0111000010000000 0000000000000000 00011111
0013 0000000000000000 1111000000001110 0000000000000010 00010111
0014 0000000000000111 0111000010011000 0000000000000010 00010111
0015 0000000001010101 0001100010000000 0000000000000010 10010011
0016 0000000000000000 1111000000001110 0000000000000010 00010111
0017 0100000000000000 0111000010011000 0000000000000010 00010111
0018 0000000000100111 0001100010000000 0000000000000010 10110011
0019 0000000000100101 0111000111101110 0001100000000010 00010111
001A 0000000000100111 0111000111101110 0001101000000010 00010111
001B 0000000000100000 0111000111101110 0001110000000010 00010111
001C 0000000001010101 0000110010000000 0000000000000000 01001111
001D 0000000001100001 0111000010111110 0000000000000000 00011110
001E 0000000000000000 0111000010000000 0000000000000000 00011111
001F 0000000000000000 1111000000001110 0000000000000010 00010111
0020 0000000000000111 0111000010011000 0000000000000010 00010111
0021 0000000001010101 0001100010000000 0000000000000010 10010011
0022 0000000000000000 1111000000001110 0000000000000010 00010111
0023 0100000000000000 0111000010011000 0000000000000010 00011111
0024 0000000000011111 0001100010000000 0000000000000010 10110011
0025 0000000001000101 0111000111101110 0001100000000010 00010111
0026 0000000001000111 0111000111101110 0001101000000010 00010111
0027 0000000001000000 0111000111101110 0001110000000010 00010111
0028 0000000001010101 0000110010000000 0000000000000000 01001111
0029 0000000000000101 0001110010000000 0000000000000000 00011111
002A 0000000000000000 0111000010111001 1100000000000000 00011110
002B 0000000000000000 0111000010000000 0000000000000000 00011111
002C 0000000000000000 0111000010000000 0000000000000000 00011111
002D 0000000000000000 1111000000001110 0000000000000010 00010111
002E 0000000000000001 0111000010011000 0000000000000010 00010111
002F 0000000001011011 0001100010000000 0000000000000010 10010011
0030 0000000001100010 0111000010111110 0000000000000000 00011110
0031 0000000011111111 0111000010111110 0000000000000000 00011110
0032 0000000000000000 0111000010000000 0000000000000000 00011111
0033 0000000000000000 1111000110001110 0000000000000010 00010111
0034 0000000000000000 1111000110001110 0000001000000010 00010111
0035 0000000000000000 1111000110001110 0000010000000010 00010111
0036 0000000000000000 1111000110001110 0000011000000010 00010111
0037 0000000000000000 0111000010111001 1000000000000000 00011110
0038 0000000000000000 0111000010111000 0000000000000000 00011111

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 9 of 14)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

```

0039 0000000000000000 0111000010111000 0010000000000000 00011111
003A 0000000000000000 0111000010111000 0100000000000000 00011111
003B 0000000000000000 0111000010111000 0110000000000000 00011111
003C 0000000011111111 0111000010111110 0000000000000000 00011110
003D 0000000000001110 011100011001010 0010001000000010 00010111
003E 000000000100010 0001100010000000 0000000000000010 10010011
003F 0000000001100101 0111000010111110 0000000000000000 00011110
0040 0000000011111111 0111000010111110 0000000000000000 00011110
0041 0000000000000000 0101010010000000 0000000000000000 01010111
0042 0000000001100011 0111000010111110 0000000000000000 00011110
0043 0000000011111111 0111000010111110 0000000000000000 00011110
0044 0000000000000000 0111000010000000 0000000000000000 00011111
0045 0000000000000000 1111000110001110 0000000000000010 00010111
0046 0000000000000000 1111000110001110 0000001000000010 00010111
0047 0000000000000000 1111000110001110 0000010000000010 00010111
0048 0000000000000000 1111000110001110 0000011000000010 00010111
0049 0000000000000000 1111000110001110 0000100000000010 00010111
004A 0000000000000000 1111000110001110 0000101000000010 00010111
004B 0000000000000000 0111000010111001 1010000000000000 00011110
004C 0000000000000000 0111000010111000 0000000000000000 00011111
004D 0000000000000000 0111000010111000 0010000000000000 00011111
004E 0000000000000000 0111000010111000 0100000000000000 00011111
004F 0000000000000000 0111000010111000 0110000000000000 00011111
0050 0000000000000000 0111000010111000 1000000000000000 00011111
0051 0000000000000000 0111000010111000 1010000000000000 00011111
0052 0000000001100101 0111000010111110 0000000000000000 00011110
0053 0000000011111111 0111000010111110 0000000000000000 00011110
0054 0000000000000000 0101010010000000 0000000000000000 01010111
0055 0000000000000000 0111000000010100 0000000000000010 00010111
0056 0000000001011100 0001100010000000 0000000000000010 10110011
0057 0000000000000000 0111000000010100 0000000000000010 00010111
0058 0000000001011111 0001100010000000 0000000000000010 10110011
0059 0000000000000000 0111000000010100 0000000000000010 00010111
005A 0000000010000100 0001100010000000 0000000000000010 10110011
005B 0000000000000101 0001110010000000 0000000000000000 00011111
005C 0000000010011110 0000110010000000 0000000000000000 01001111
005D 0000000011001110 0000110010000000 0000000000000000 01001111
005E 0000000000000101 0001110010000000 0000000000000000 00011111
005F 0000000010011110 0000110010000000 0000000000000000 01001111
0060 0000000011001110 0000110010000000 0000000000000000 01001111
0061 1101101010000000 0111000111101110 0001111000000010 00010111
0062 0000000000000000 0111000010111001 1100000000000000 00011110
0063 0000000000000000 0111000010000000 0000000000000000 00011111
0064 0000000000000000 0111000010000000 0000000000000000 00011111
0065 0000000000000000 1111000000011110 0000000000000010 00010111
0066 0000000000000010 0111000001001100 0000000000000010 00010111
0067 0000000001101111 0001100010000000 0000000000000010 10010011
0068 0000000000000000 0111000110010110 0001111000000010 00010111
0069 0000000001100101 0001100010000000 0000000000000010 10010011
006A 1010101010101010 0111000111101110 0000000000000010 00010111
006B 1010101010101010 0111000111101110 0000001000000010 00010111
006C 1010101010101010 0111000111101110 0000010000000010 00010111
006D 1010101010101010 0111000111101110 0000011000000010 00010111
006E 0000000001111000 0001110010000000 0000000000000000 00011111
006F 0000000000000000 0111000010111001 1010000000000000 00011110
0070 0000000000000000 0111000010000000 0000000000000000 00011111
0071 0000000000000000 0111000010000000 0000000000000000 00011111

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 10 of 14)

```

0072 0000000000000000 0111000010000000 0000000000000000 00011111
0073 0000000000000000 1111000110001110 0000000000000010 00010111
0074 0000000000000000 1111000110001110 0000001000000010 00010111
0075 0000000000000000 1111000110001110 0000010000000010 00010111
0076 0000000000000000 1111000110001110 0000011000000010 00010111
0077 0000000011111111 0111000010111110 0000000000000000 00011110
0078 0000000001100001 0111000010111110 0000000000000000 00011110
0079 0000000000000000 0111000010000000 0000000000000000 00011111
007A 0000000000000000 1111000000001110 0000000000000010 00010111
007B 1000000000000000 0111000010011000 0000000000000010 00010111
007C 0000000001111010 0001100010000000 0000000000000010 10110011
007D 0000000001100100 0111000010111110 0000000000000000 00011110
007E 0000000011111111 0111000010111110 0000000000000000 00011110
007F 0000000000000000 0111000010111000 0000000000000000 00011111
0080 0000000000000000 0111000010111000 0010000000000000 00011111
0081 0000000000000000 0111000010111000 0100000000000000 00011111
0082 0000000000000000 0111000010111000 0110000000000000 00011111
0083 0000000000000101 0001110010000000 0000000000000000 00011111
0084 0000000001100001 0111000010111110 0000000000000000 00011110
0085 0000000000000000 0111000010000000 0000000000000000 00011111
0086 0000000000000000 1111000000001110 0000000000000010 00010111
0087 0100000000000000 0111000010011000 0000000000000010 00010111
0088 0000000000000101 0001100010000000 0000000000000010 10110011
0089 0000000001100010 0111000010111110 0000000000000000 00011110
008A 0000000011111111 0111000010111110 0000000000000000 00011110
008B 0000000000000000 0111000010000000 0000000000000000 00011111
008C 0000000000000000 1111000110001110 0000000000000010 00010111
008D 0000000000000000 1111000110001110 0000001000000010 00010111
008E 0000000000000000 1111000110001110 0000010000000010 00010111
008F 0000000000000000 1111000110001110 0000011000000010 00010111
0090 0000000001100101 0111000010111110 0000000000000000 00011110
0091 0000000011111111 0111000010111110 0000000000000000 00011110
0092 0000000001100001 0111000010111110 0000000000000000 00011110
0093 0000000000000000 0111000010000000 0000000000000000 00011111
0094 0000000000000000 1111000000001110 0000000000000010 00010111
0095 1000000000000000 0111000010011000 0000000000000010 00010111
0096 0000000010010100 0001100010000000 0000000000000010 10110011
0097 0000000001100100 0111000010111110 0000000000000000 00011110
0098 0000000011111111 0111000010111110 0000000000000000 00011110
0099 0000000000000000 0111000010111000 0000000000000000 00011111
009A 0000000000000000 0111000010111000 0010000000000000 00011111
009B 0000000000000000 0111000010111000 0100000000000000 00011111
009C 0000000000000000 0111000010111000 0110000000000000 00011111
009D 0000000000000101 0001110010000000 0000000000000000 00011111
009E 0000000001100001 0111000010111110 0000000000000000 00011110
009F 0000000000000000 0111000010000000 0000000000000000 00011111
00A0 0000000000000000 1111000110001110 0000110000000010 00010111
00A1 0000000000000000 0111001010110110 0000110000000010 00010111
00A2 0000000000000000 0111001010110110 0000110000000010 00010111
00A3 0000000000000000 0111001010110110 0000110000000010 00010111
00A4 0000000000000011 0111000111001010 1100110000000010 00010111
00A5 0000000011000101 0001100010000000 0000000000000010 10110011
00A6 0000000000000000 0111000110010110 0000110000000010 00010111
00A7 0000000010111100 0001100010000000 0000000000000010 10110011
00A8 0000000000000000 0111000110010110 0000110000000010 00010111
00A9 0000000010110011 0001100010000000 0000000000000010 10110011
00AA 000000000110001 011100011101110 0001001000000010 00010111

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 11 of 14)

```

00AB 0000000000110111 0111000111101110 0001010000000010 00010111
00AC 0000000000110010 0111000111101110 0001011000000010 00010111
00AD 0000000000110011 0111000111101110 0001100000000010 00010111
00AE 0000000000110100 0111000111101110 0001101000000010 00010111
00AF 0000000000110000 0111000111101110 0001110000000010 00010111
00B0 0000000000110101 0111000111101110 0001000000000010 00010111
00B1 0000000000110110 0111000111101110 0000111000000010 00010111
00B2 0000000000000000 0101010010000000 0000000000000000 01010111
00B3 0000000000100001 0111000111101110 0001001000000010 00010111
00B4 0000000000100111 0111000111101110 0001010000000010 00010111
00B5 0000000000100010 0111000111101110 0001011000000010 00010111
00B6 0000000000100011 0111000111101110 0001100000000010 00010111
00B7 0000000000100100 0111000111101110 0001101000000010 00010111
00B8 0000000000100000 0111000111101110 0001110000000010 00010111
00B9 0000000000100101 0111000111101110 0001000000000010 00010111
00BA 0000000000100110 0111000111101110 0000111000000010 00010111
00BB 0000000000000000 0101010010000000 0000000000000000 01010111
00BC 000000000010001 0111000111101110 0001001000000010 00010111
00BD 000000000010111 0111000111101110 0001010000000010 00010111
00BE 000000000010010 0111000111101110 0001011000000010 00010111
00BF 000000000010011 0111000111101110 0001100000000010 00010111
00C0 000000000010100 0111000111101110 0001101000000010 00010111
00C1 000000000010000 0111000111101110 0001110000000010 00010111
00C2 000000000010101 0111000111101110 0001000000000010 00010111
00C3 000000000010110 0111000111101110 0000111000000010 00010111
00C4 0000000000000000 0101010010000000 0000000000000000 01010111
00C5 0000000000000001 0111000111101110 0001001000000010 00010111
00C6 0000000000000111 0111000111101110 0001010000000010 00010111
00C7 0000000000000010 0111000111101110 0001011000000010 00010111
00C8 0000000000000011 0111000111101110 0001100000000010 00010111
00C9 0000000000000100 0111000111101110 0001101000000010 00010111
00CA 0000000000000000 0111000111101110 0001110000000010 00010111
00CB 0000000000000101 0111000111101110 0001000000000010 00010111
00CC 0000000000000110 0111000111101110 0000111000000010 00010111
00CD 0000000000000000 0101010010000000 0000000000000000 01010111
00CE 0000000001100001 0111000010111110 0000000000000000 00011110
00CF 0000000000000000 0111000010000000 0000000000000000 00011111
00D0 0000000000000000 1111000000001110 0000000000000010 00010111
00D1 0100000000000000 0111000001001100 0000000000000010 00010111
00D2 0000000011010000 0001100010000000 0000000000000010 10110011
00D3 0000000000000000 0111000010111001 0010000000000000 00011110
00D4 0000000011111111 0111000010111110 0000000000000000 00011110
00D5 000000001100001 0111000010111110 0000000000000000 00011110
00D6 0000000000000000 0111000010000000 0000000000000000 00011111
00D7 0000000000000000 1111000000001110 0000000000000010 00010111
00D8 0001000000000000 011100001001100 0000000000000010 00010111
00D9 0000000011011101 0001100010000000 0000000000000010 10110011
00DA 0000000000000000 0111000010111001 0000000000000000 00011110
00DB 0000000011111111 0111000010111110 0000000000000000 00011110
00DC 0000000011011111 0001110010000000 0000000000000000 00011111
00DD 0000000000000000 0111000010111000 1110000000000000 00011110
00DE 0000000011111111 0111000010111110 0000000000000000 00011110
00DF 0000000001100010 0111000010111110 0000000000000000 00011110
00E0 0000000011111111 0111000010111110 0000000000000000 00011110
00E1 0000000000000000 0111000010000000 0000000000000000 00011111
00E2 0000000000000000 1111000110001110 0000000000000010 00010111
00E3 0000000000000000 1111000110001110 0000001000000010 00010111

```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 12 of 14)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

```
00E4 0000000000000000 1111000110001110 0000010000000010 00010111
00E5 0000000000000000 1111000110001110 0000011000000010 00010111
00E6 0000000000000000 0111000010111001 0110000000000000 00011110
00E7 0000000000000000 0111000010111000 0000000000000000 00011111
00E8 0000000000000000 0111000010111000 0010000000000000 00011111
00E9 0000000000000000 0111000010111000 0100000000000000 00011111
00EA 0000000000000000 0111000010111000 0110000000000000 00011111
00EB 0000000011111111 0111000010111110 0000000000000000 00011110
00EC 0000000000001110 0111000111001010 0010001000000010 00010111
00ED 0000000011110010 0001100010000000 0000000000000010 10010011
00EE 0000000000000000 0111000010111001 0100000000000000 00011110
00EF 0000000001100101 0111000010111110 0000000000000000 00011110
00F0 0000000011111111 0111000010111110 0000000000000000 00011110
00F1 0000000000000000 0101010010000000 0000000000000000 01010111
00F2 0000000001100011 0111000010111110 0000000000000000 00011110
00F3 0000000011111111 0111000010111110 0000000000000000 00011110
00F4 0000000000000000 0111000010000000 0000000000000000 00011111
00F5 0000000000000000 1111000110001110 0000000000000010 00010111
00F6 0000000000000000 1111000110001110 0000001000000010 00010111
00F7 0000000000000000 1111000110001110 0000010000000010 00010111
00F8 0000000000000000 1111000110001110 0000011000000010 00010111
00F9 0000000000000000 1111000110001110 0000100000000010 00010111
00FA 0000000000000000 1111000110001110 0000101000000010 00010111
00FB 0000000000000000 0111000010111001 1000000000000000 00011110
00FC 0000000000000000 0111000010111000 0000000000000000 00011111
00FD 0000000000000000 0111000010111000 0010000000000000 00011111
00FE 0000000000000000 0111000010111000 0100000000000000 00011111
00FF 0000000000000000 0111000010111000 0110000000000000 00011111
0100 0000000000000000 0111000010111000 1000000000000000 00011111
0101 0000000000000000 0111000010111000 1010000000000000 00011111
0102 0000000000000000 0111000010111001 0100000000000000 00011110
0103 0000000001100101 0111000010111110 0000000000000000 00011110
0104 0000000011111111 0111000010111110 0000000000000000 00011110
0105 0000000000000000 0101010010000000 0000000000000000 01010111
```

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 13 of 14)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

SYMBOLS

AB	0001	AQ	0000	BEGIN	0005	BUSY	0001
CIONE	0001	CIZERO	0000	CJP	0003	CJPP	000B
CJS	0001	CJV	0006	COND	0000	CONT	000E
CRTN	000A	CTRLN	0000	CWMSK	1000	CWQFF	000D
DA	0005	DECRC	0065	DIAG	0055	DIAGMAT	00CE
DMREPLY	00F2	DQ	0006	DZ	0007	EXNOR	0007
EXOR	0006	FADD	0000	FAND	0004	FOR	0003
GETDATA	006F	GOMASK	0003	IFSTATUS	0061	INIT	0000
ITOUSR	0010	JMAP	0002	JRP	0007	JSRP	0005
JUMP1	000F	JZ	0000	LDCT	000C	LOAD4	0064
LOADDATA	0078	LOADMAT	002A	LOOP	000D	LOOPA	0007
LOOPB	0013	LOOPC	001F	LOOPD	002D	LOOPE	0000
LOOPH	0065	LOOPI	007A	LOOPK	0094	MACROEN	0000
MAT1	0000	MATICW	0005	MAT1DIAG	0007	MAT1LD4	0002
MAT1LD6	0003	MAT1NCW	0006	MAT1RD4	0004	MAT1RST	0001
MAT1STAT	0000	MAT2	0019	MAT2CW	0015	MAT2DIAG	0017
MAT2LD4	0012	MAT2LD6	0013	MAT2NCW	0016	MAT2RD4	0014
MAT2RST	0011	MAT2STAT	0010	MAT3	0025	MAT3CW	0025
MAT3DIAG	0027	MAT3LD4	0022	MAT3LD6	0023	MAT3NCW	0026
MAT3RD4	0024	MAT3RST	0021	MAT3STAT	0020	MAT4CW	0035
MAT4DIAG	0037	MAT4LD4	0032	MAT4LD6	0033	MAT4NCW	0036
MAT4RD4	0034	MAT4RST	0031	MAT4STAT	0030	MICROEN	0000
MREPLY	0042	MRGATN	0067	MSRTOUSR	0002	NOCTRL	0001
NOMACRO	0001	NOMICRO	0001	NONEON	0003	NOOP	0000
NORMASK	0007	NOTRS	0005	NP	0001	PEMASK	8000
PUSH	0004	QREG	0000	R0	0000	R1	0001
R10	000A	R11	000B	R12	000C	R13	000D
R14	000E	R15	000F	R2	0002	R3	0003
R4	0004	R5	0005	R6	0006	R7	0007
R8	0008	R9	0009	RAMA	0002	RAMD	0005
RAMF	0003	RAMQD	0004	RAMQU	0006	RAMU	0007
READ4	0062	READ6	0063	RFCT	0008	RFRQDIAG	005C
RPCT	0009	RPLYDIAG	005F	RPYAVAIL	4000	RPYDATA	0002
SELA	0000	SELB	0001	SELMAT	009E	SELMAT1	00C5
SELMAT2	00BC	SELMAT3	00B3	SELMAT4	00AA	SFTRA	0000
SHIFT	0002	SUBR	0001	SUBS	0002	TEST	0001
TIMER	DAC0	TWB	000F	TYPE	000E	UC	001B
UN	001F	UNC	001A	UNCOND	0001	UNN	001E
UNOVR	0016	UNZ	0014	UOVR	0017	UPRCDIAG	0084
USRTOMSR	0002	UZ	0015	ZA	0004	ZB	0003
ZQ	0002						

TOTAL PHASE 2 ERRORS = 0

FIGURE B.3.1-2. MRG MICROCODE PROGRAM LISTING (Sheet 14 of 14)

If the Interface Loopback Validation is selected, the data paths and memories of the MRG Buffer Interface can be verified. However, the MRC does not participate in this test. (See page 5, labeled "DIAG:.") It constantly monitor the operation mode field.

If the MRC Validation is selected, the MRC proceeds to fetch 4-word blocks (unformatted) from the MRG Buffer Interface and return them to the CPU. (See page 6, titled "MICROPROCESSOR LOOPBACK ROUTINE.") In this mode, the data paths between the MRC and the Buffer Interface can be verified.

If the MAT Validation is selected, the MRC proceeds to sequence known ATCRBS replies, provided by the CPU, to a selected MAT. (See page 5, titled "REPLY GENERATOR LOOPBACK ROUTINE.") As the MAT transmits the reply, the reply data is reassembled and returned to the MRC, which in turn returns the data to the CPU. In this mode, the data paths between the MRC and the MATs are verified. Also, many of the hardware functions of the MATs are verified. (Functions unique to the Mode S replies are not checked by this routine, but are checked by the MAT RF Validation.) Again, two conditions must be met before a reply is transferred: (1) a reply must be available in the MRG Buffer Interface, and (2) the selected MAT must be ready to accept a reply. MAT selection is under CPU control. (See pages 6 and 7, titled "SELECT MAT.") If the MAT fails to return the reply block after 70 millisecond (ms), the MRC will substitute an all "A" data pattern back to the CPU. In this way, it can be determined that the MAT did not respond.

If the MAT RF Validation mode is selected, the MRC proceeds to sequence replies to a selected MAT for transmission. (See page 5, titled "MRGLOOP DIAGNOSTIC ROUTINE.") The reply transfer conditions to a MAT follows the same conditions mentioned in the previous paragraph. This test mode can operate in conjunction with the Self Test Unit (STU) Receiver, to collect the replies transmitted (ATCRBS or Mode S) for verification. In this mode, the MRG analog circuitry can be verified. Also, the MAT hardware functions, unique to the Mode S replies, can be verified.

### B.3.2 FRG Microcode Program.

This microcode program, resident in the Fruit Reply Controller (FRC) ROM, provides seven modes of operation for the FRC: (1) normal operation, (2) FRG Interface Loopback Validation, (3) FRC Validation, (4) ATCRBS Random Process Generator (RPG) Validation, (5) Mode S RPG Validation, (6) Fruit ARIES Target (FAT) Validation, and (7) FAT RF Validation. These routines are illustrated by the flow charts presented in figure B.3.2-1. As in the MRC, there are four ways to initialize the FRC microcode program. Upon initialization, the FRC resets all four FAT reply generators and resets both RPGs with initial fruit parameters. (See figure B.3.2-2, page No. 3, titled "INITIALIZATION.")

As with the MRG, the CPU selects the mode of operation by setting the appropriate value in the operation mode field of the FRG command. (Refer to the ARIES Hardware Maintenance Manual, Volume II, Appendix D for detailed information on the FRG command format.) Table B.3.2-1 lists the select mode options.

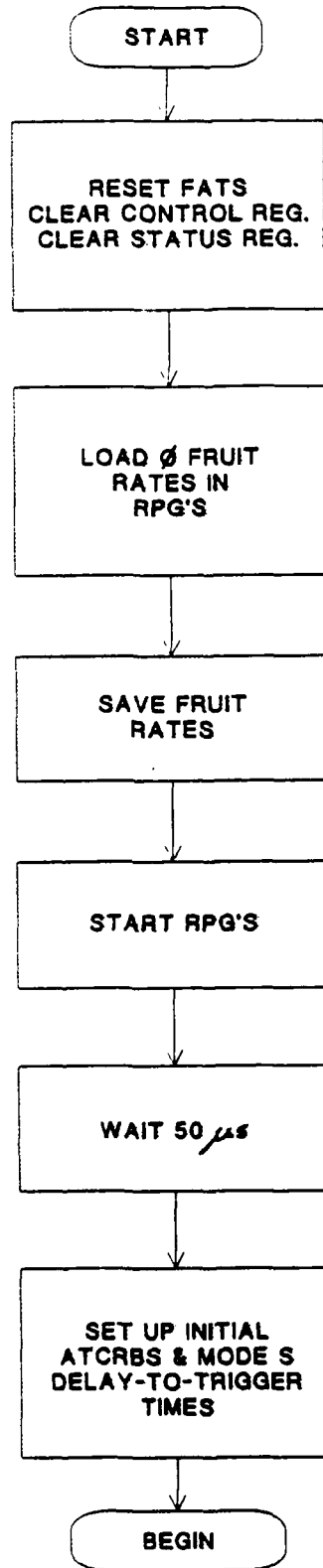


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 1 of 15)



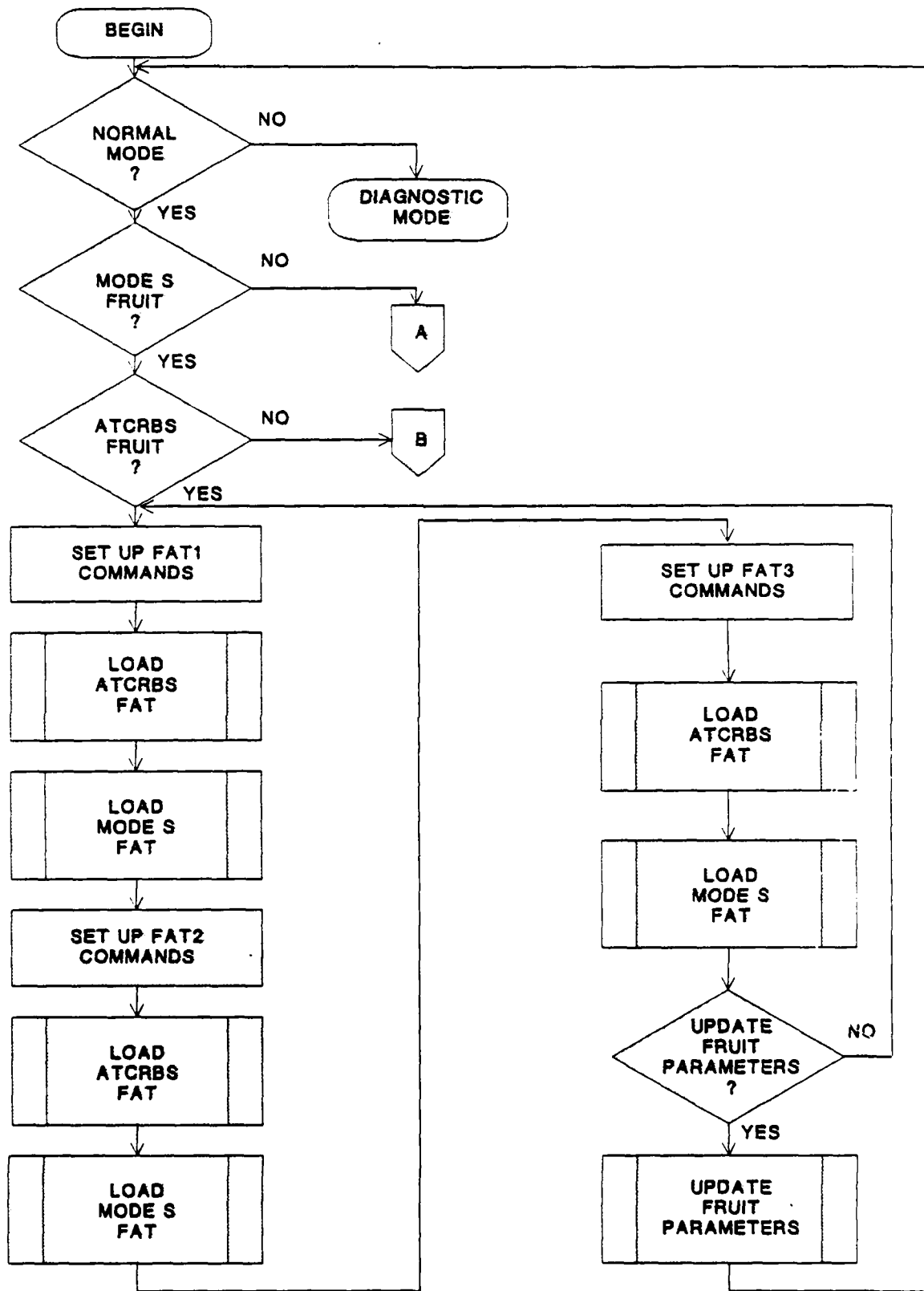


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 2 of 15)

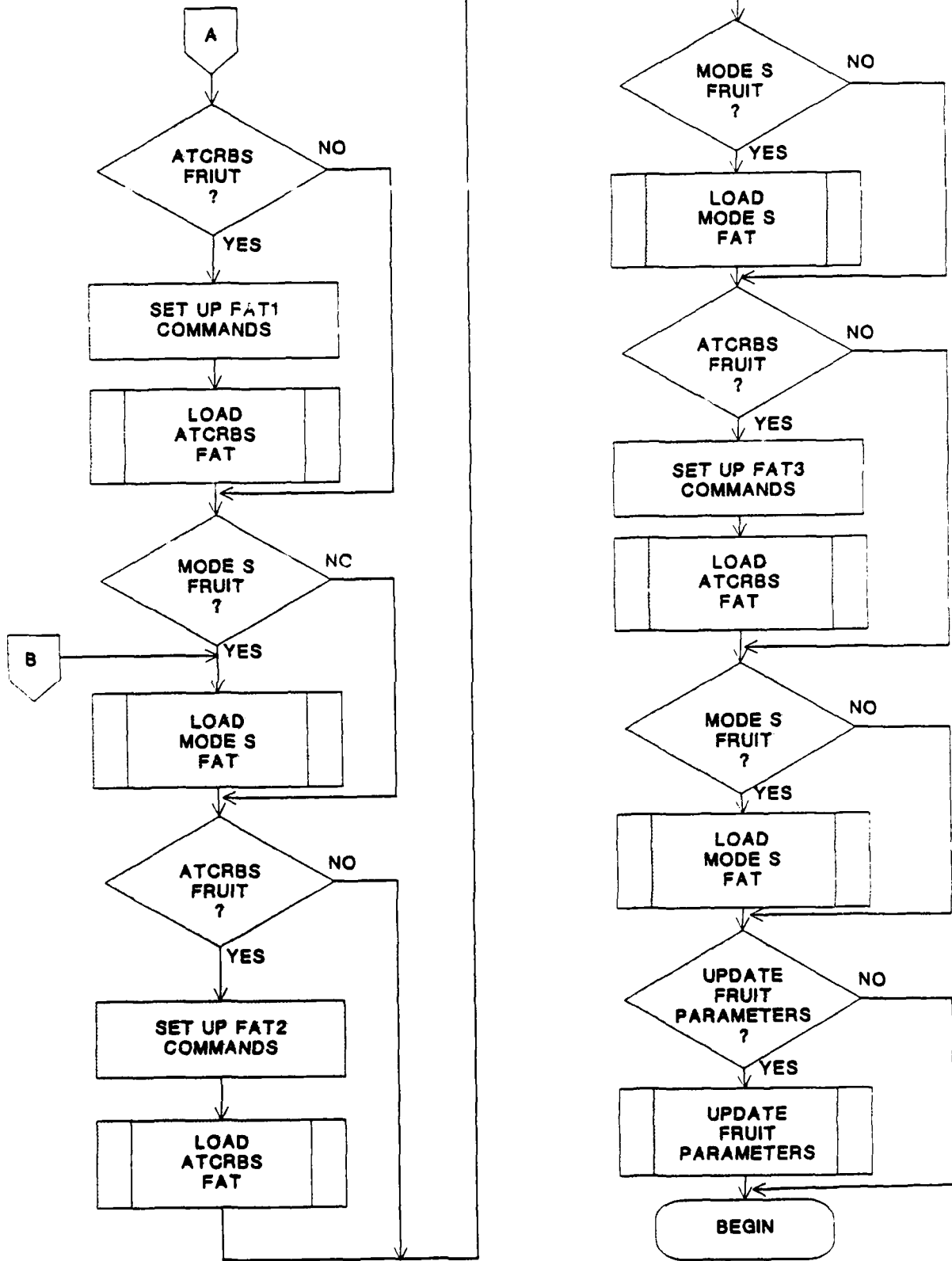


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 3 of 15)

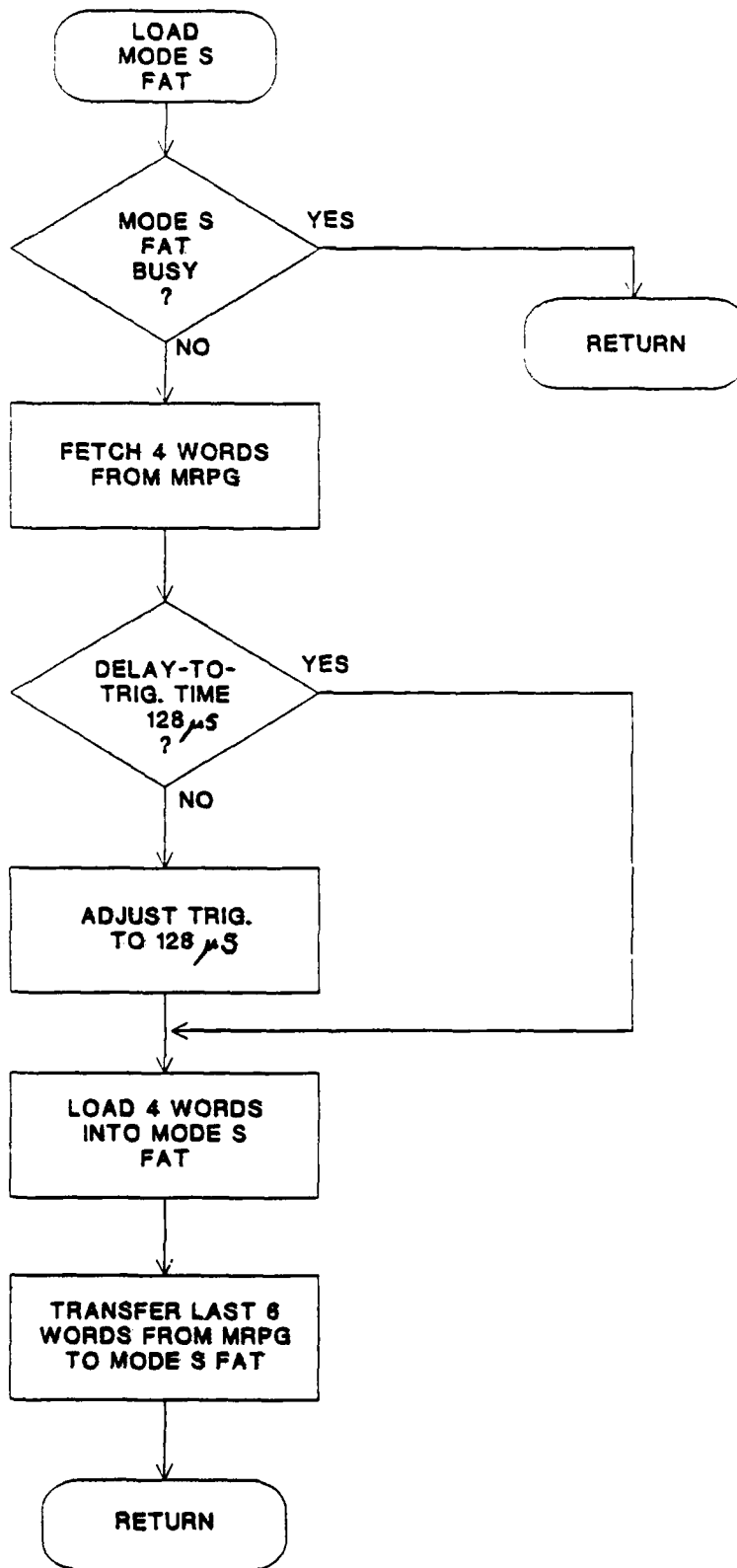


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 4 of 15)

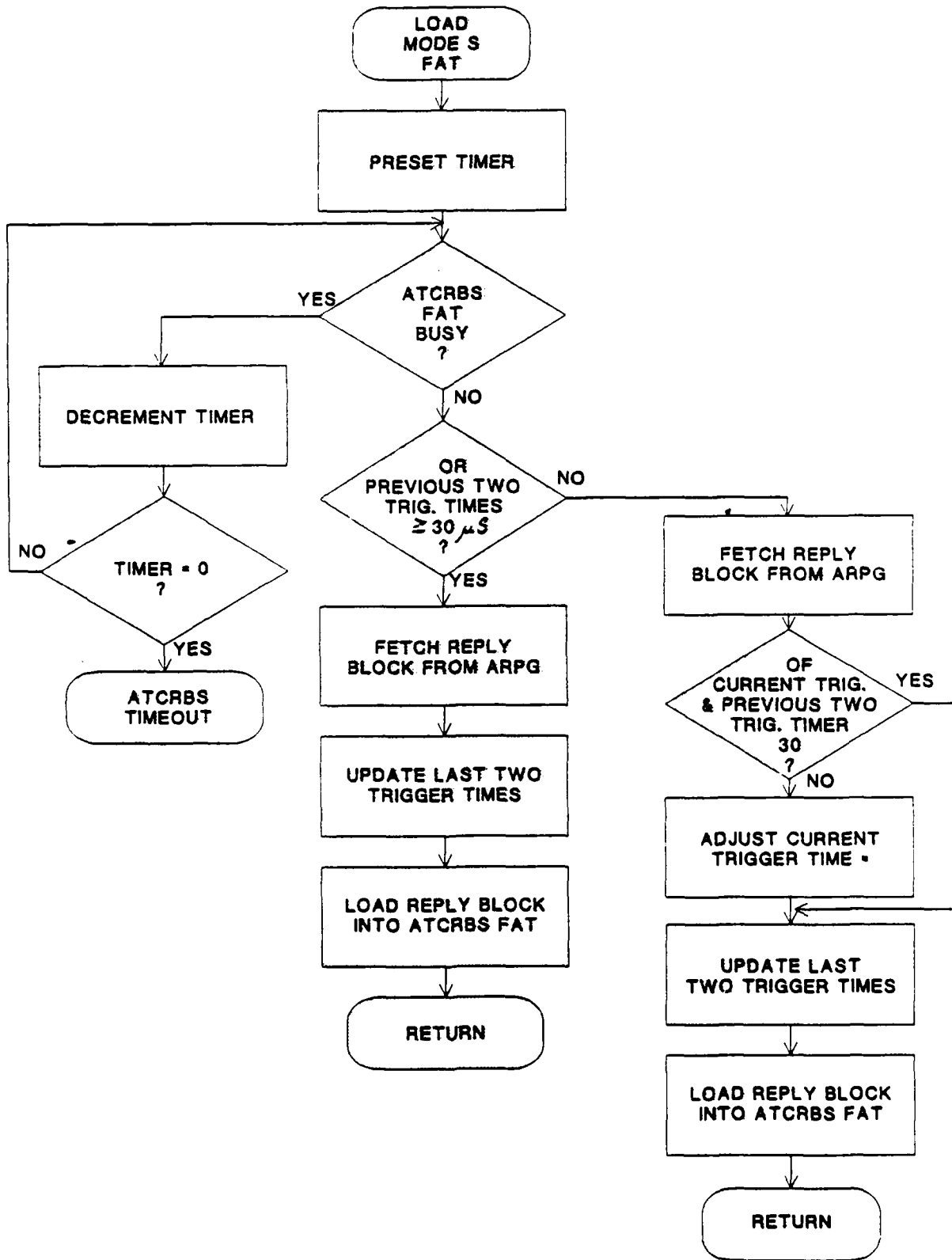
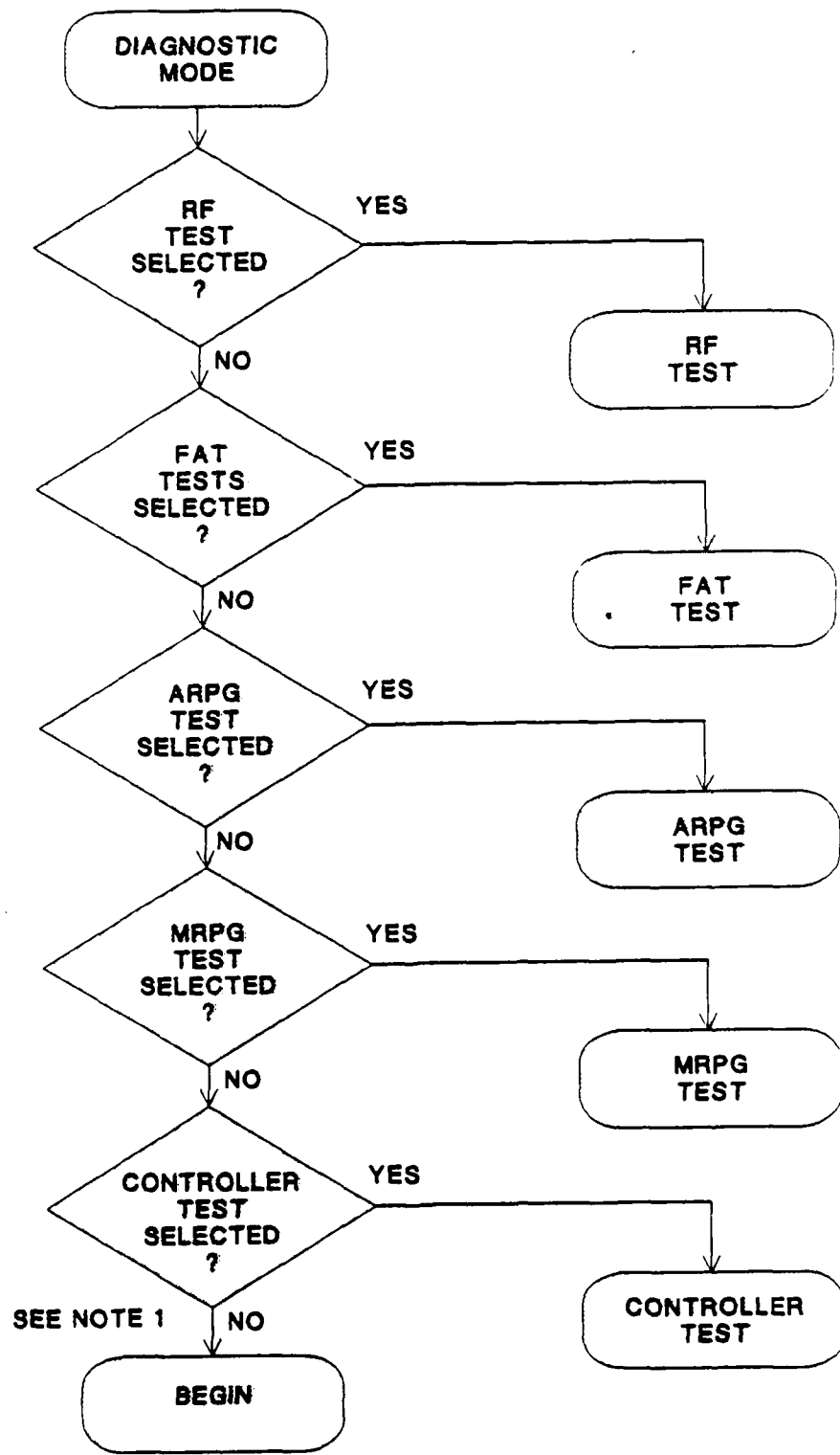


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 5 of 15)



NOTE 1: IF THE MICROCODE REACHES THIS POINT, THE FRG INTERFACE LOOPBACK TESTS ARE SELECTED.

FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 6 of 15)

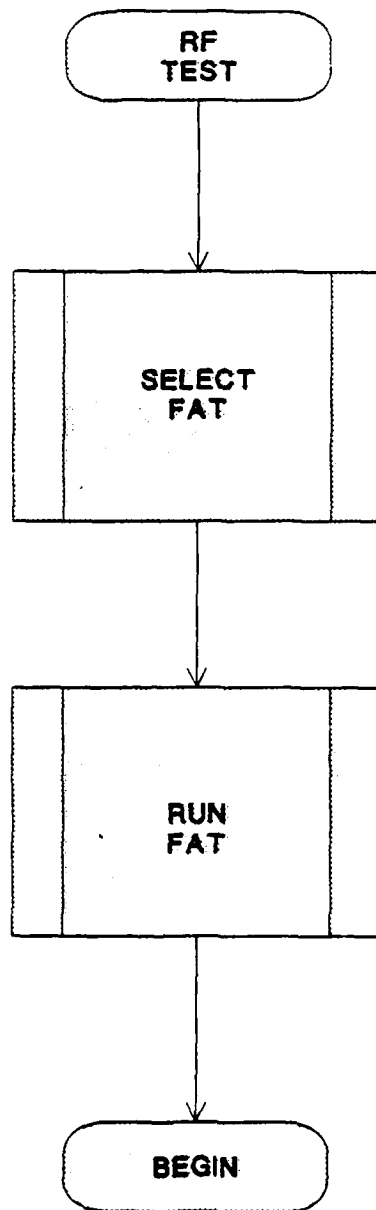


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 7 of 15)

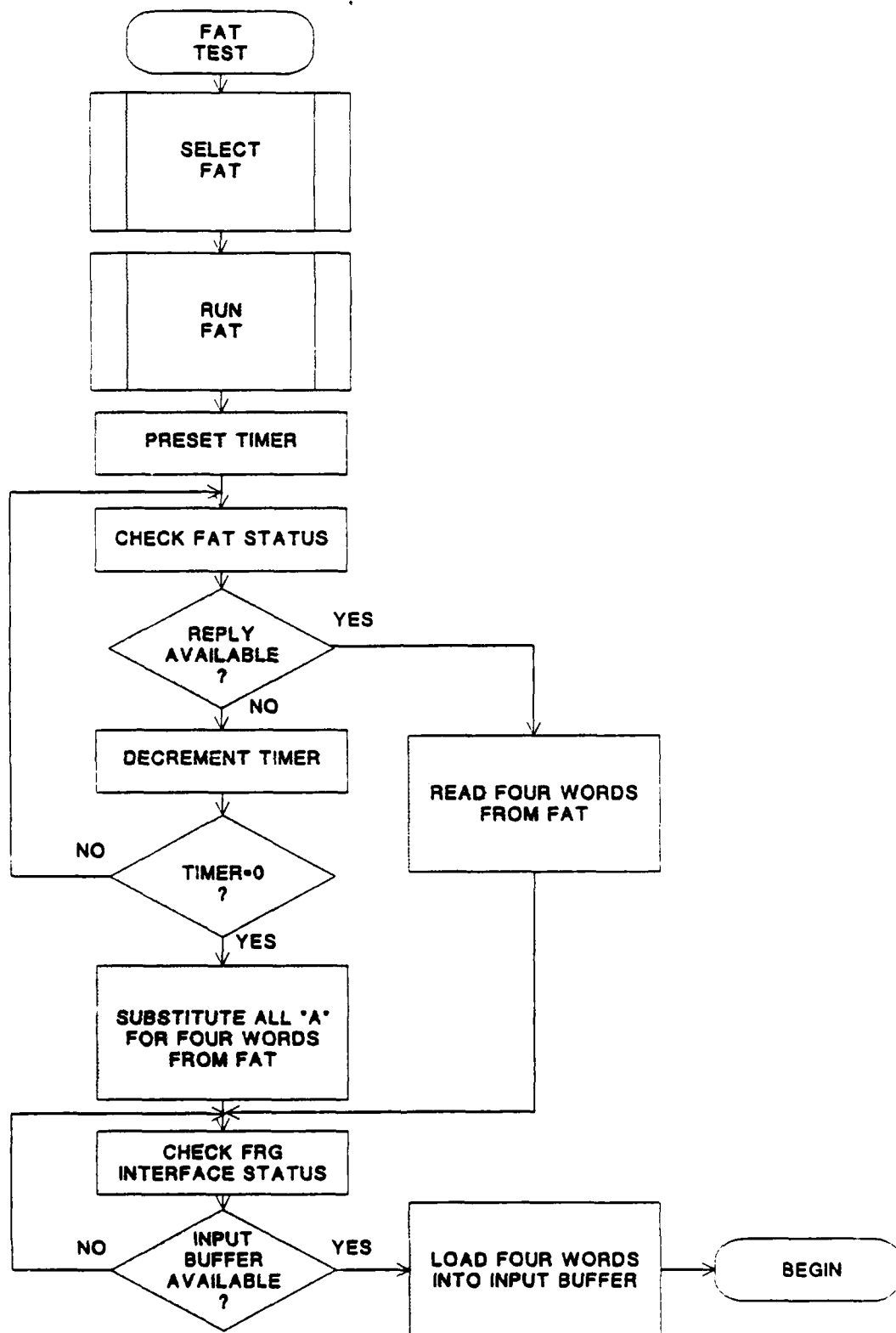


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 8 of 15)

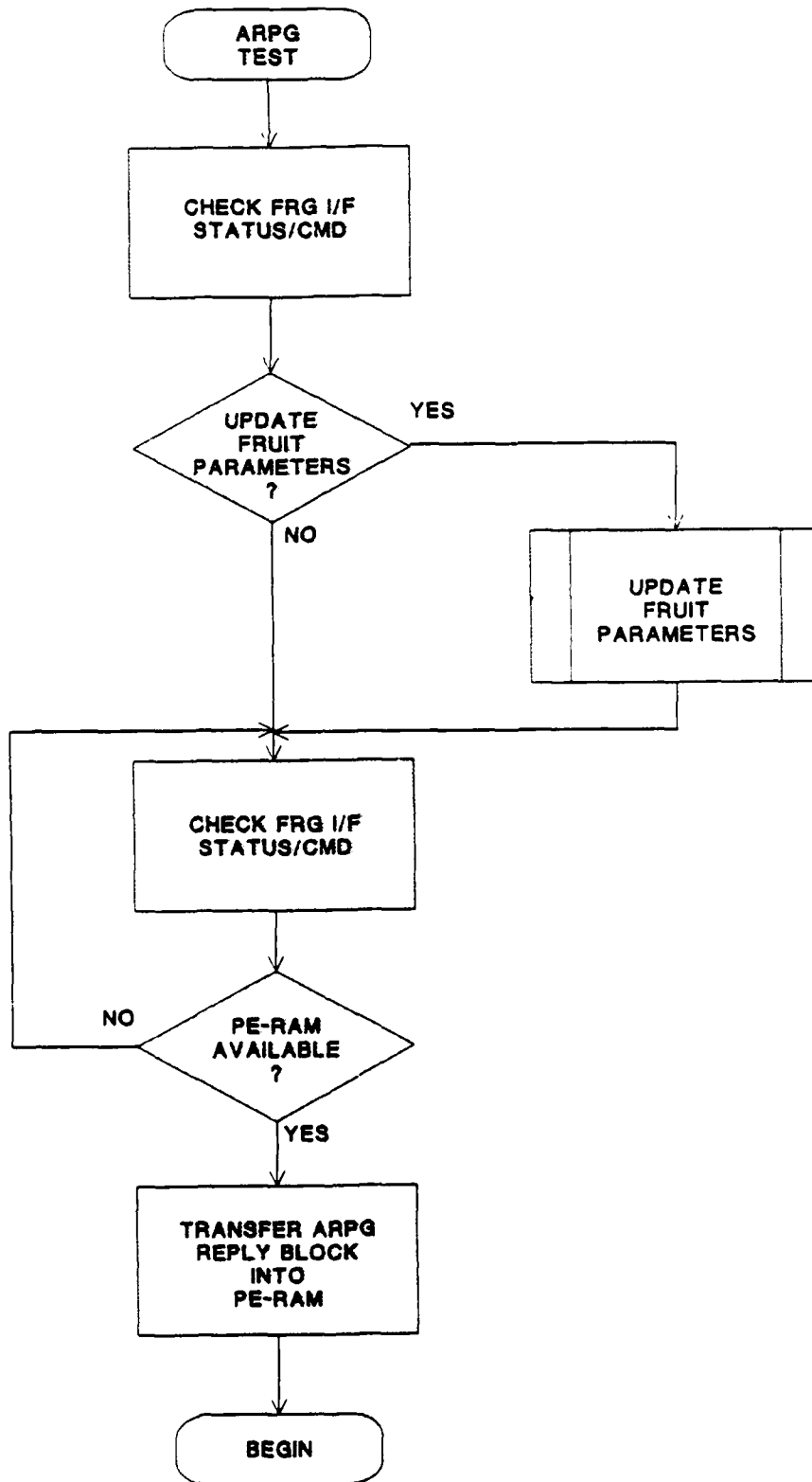


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 9 of 15)



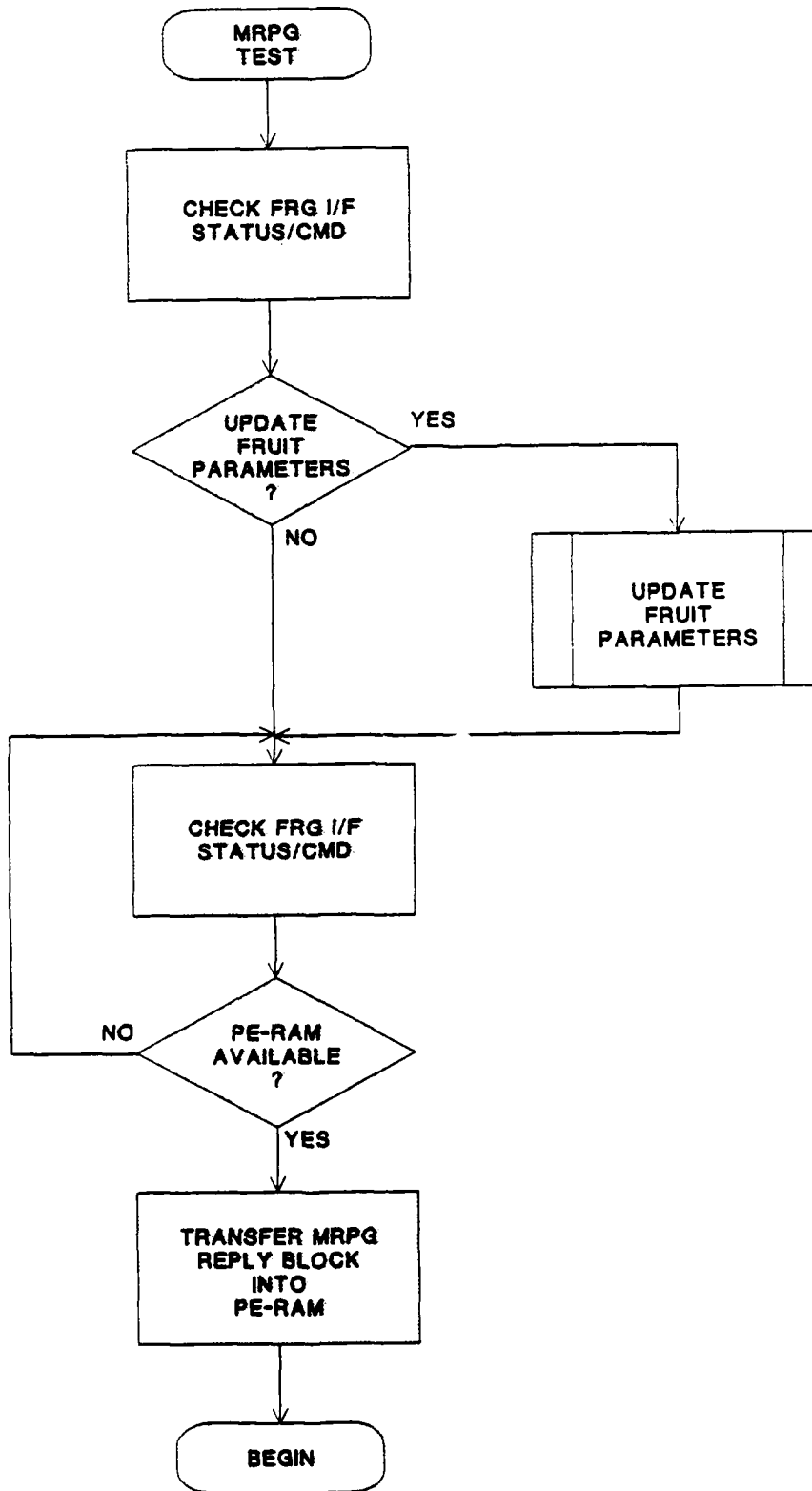


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 10 of 15)

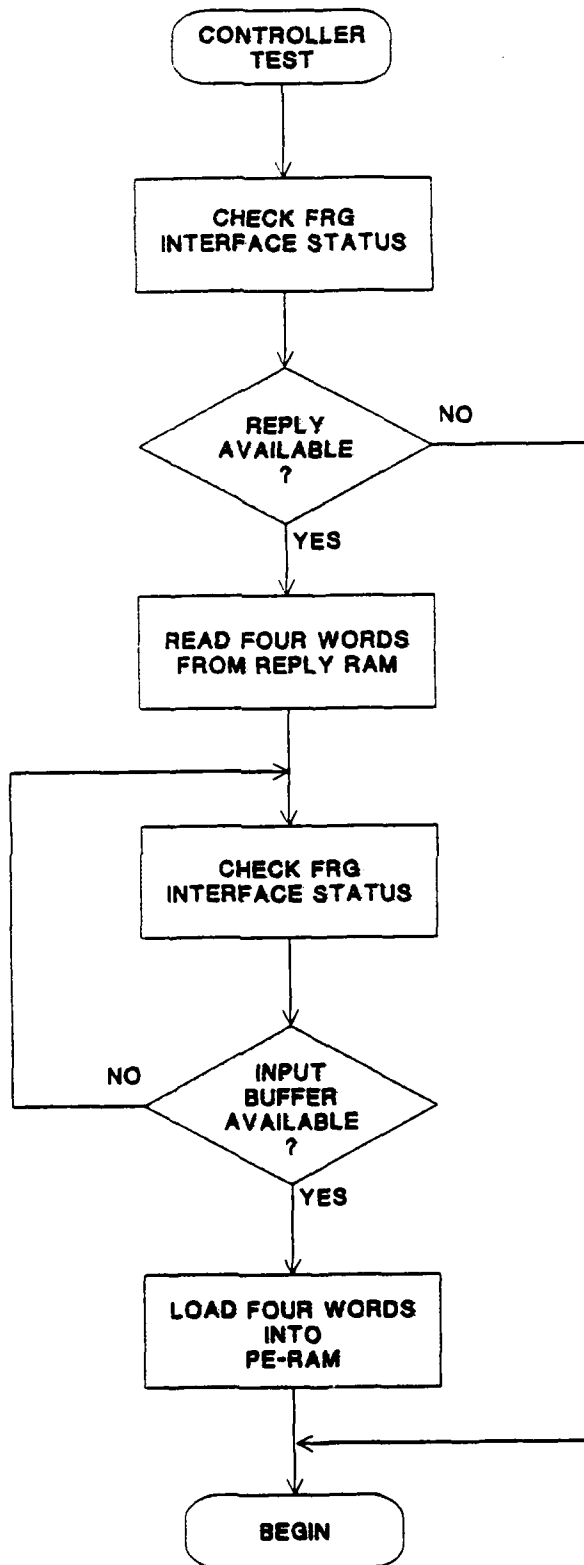


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 11 of 15)

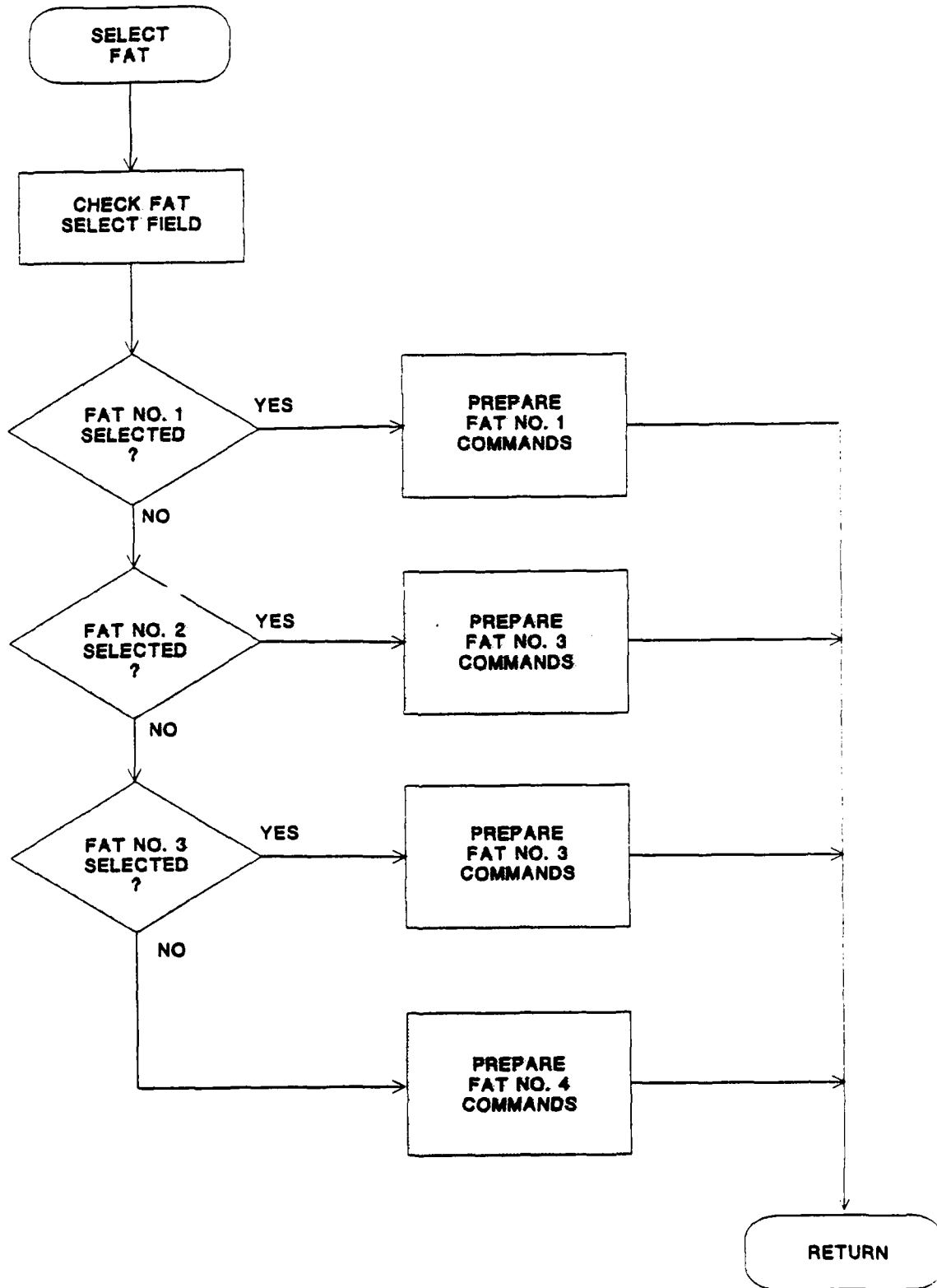


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 12 of 15)

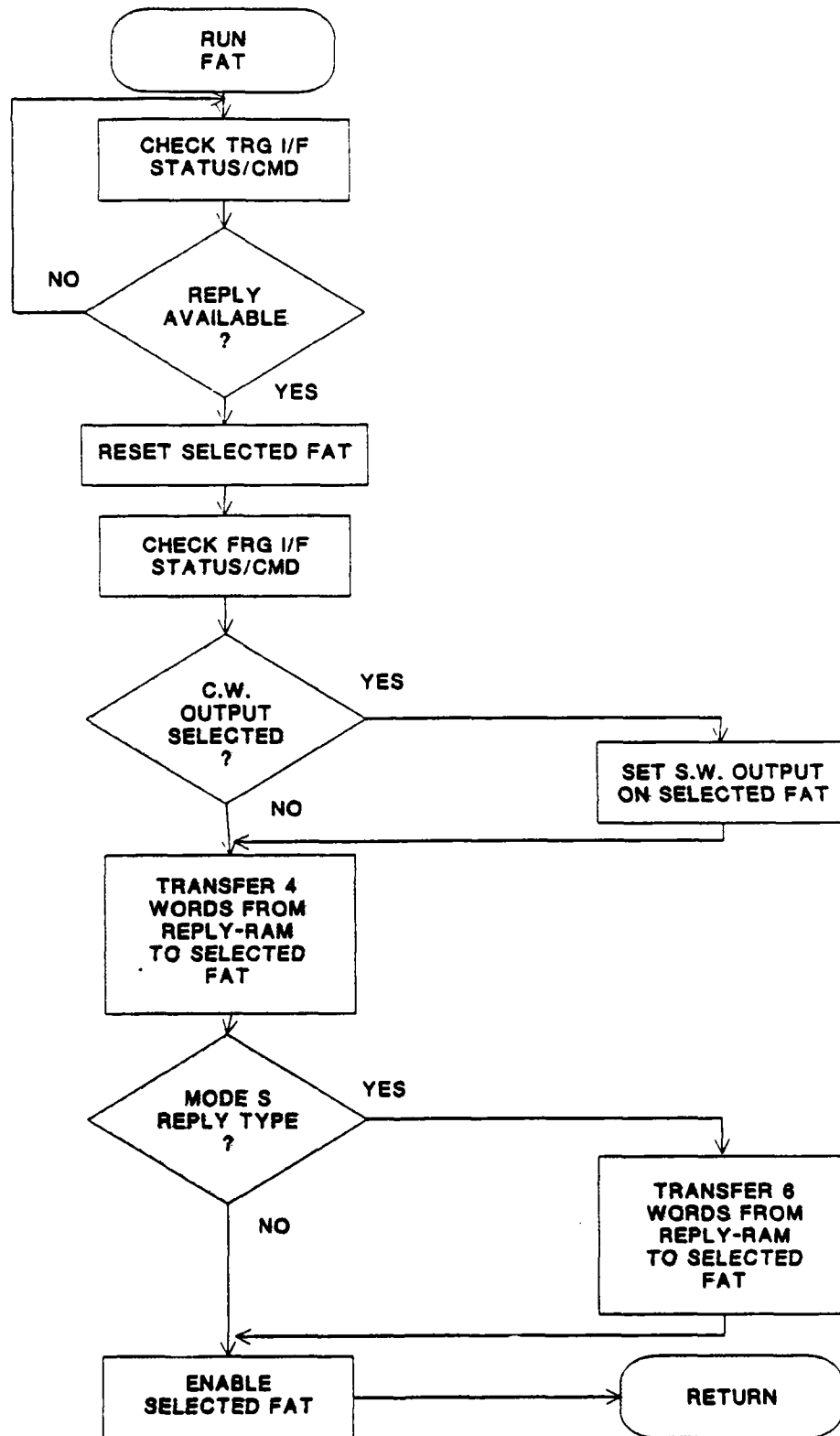


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 13 of 15)

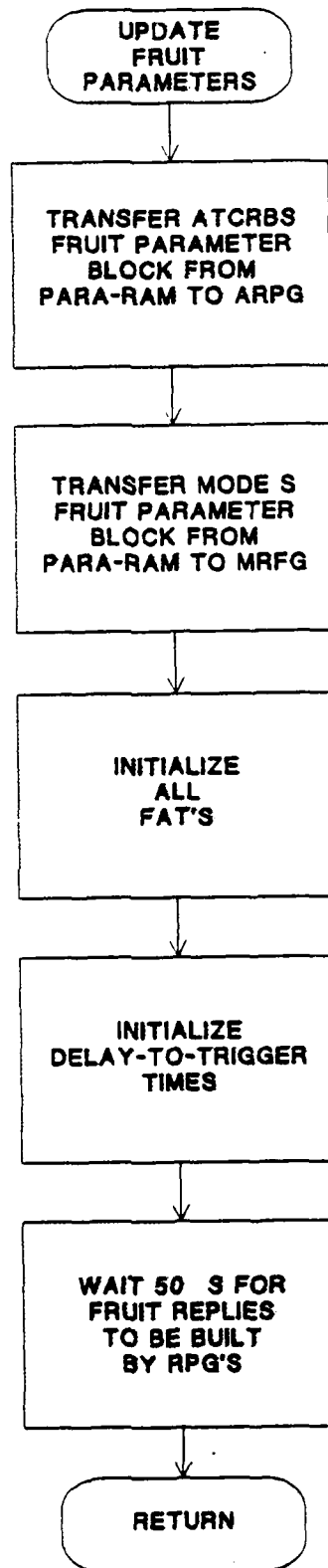


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 14 of 15)

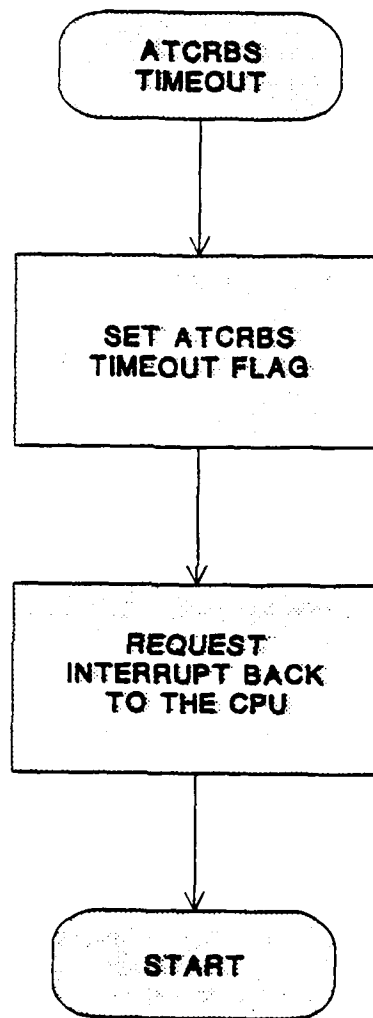


FIGURE B.3.2-1. FRG MICROCODE FLOW DIAGRAMS (Sheet 15 of 15)

TABLE B.3.2-1. FRG MODE SELECTION FIELD

Mode (Oct)	Routine Selection
0	Normal Operation
1	FAT RF Validation
2	FAT Validation
3	Mode S RPG Validation
4	ATCRBS RPG Validation
5	FRC Validation
6 - 7	FRG Interface Loopback Validation

A printout of the FRG Microcode program is presented in figure B.3.2-2. This is the AMDASM Phase II output listing, consisting of 25 pages. All of the FRG commands are defined on the first two pages.

When the mode selected by the CPU is normal, the FRC will then determine whether ATCRBS and/or Mode S fruit is to be generated. (See page 4, titled "NORMAL MODE.") The controller then begins to sequence replies from the appropriate RPG to the appropriate FATs. For ATCRBS fruit, the FRC sequences replies from the ATCRBS RPG (ARPG) to three of the FATs in a round robin fashion. For Mode S fruit, the FRC sequences replies from the Mode S RPG (MRPG) to a fourth FAT. Also, the FRC will periodically update the RPGs with new fruit parameters when they are provided by the CPU. (See page 13, titled "UPDATE FRUIT PARAMETERS.")

Before each reply is sent to a FAT, the FRC will examine the delay-to-trigger time. In the case of ATCRBS replies, if the sum of the current ATCRBS and the two previous ATCRBS delay-to-trigger times is less than 30 microsecond ( s), the current time will be modified to meet the requirement. (See page 7, titled "LOAD ATCRBS FATs.") In the case of Mode S replies, if the current Mode S delay-to-trigger time is less than 128 s, the current time will be modified to meet the requirement. (See page 6, titled "LOAD MODE S FAT.") These adjustments are necessary since a FAT cannot start another reply until the previous reply is over. Furthermore, if any ATCRBS FAT remains busy for more than 5 ms, i.e., does not respond to a poll, an interrupt will be sent back to the CPU to signal the problem. (See page 14, titled "ATCRBS TIMEOUT OCCURRED.") Note that each ATCRBS FAT should reply within 4.096 ms since the delay-to-trigger time counter will wrap around at that point. Nonresponse by the Mode S FAT is handled by hardware.

If the Interface Loopback Validation is selected, the data paths and memories of the FRG Buffer Interface can be verified. However, the FRC does not participate in this test. (See page 8, labled "DIAG:.") It constantly monitors the operation mode field.

If the FRC Validation is selected, the FRC proceeds to fetch 4-word blocks (unformatted) from the FRG Buffer Interface reply random access memory (RAM) and return them to the CPU. (See page 11, titled "MICROPROCESSOR LOOPBACK.") In this mode, the data paths between the FRC and the Buffer Interface can be verified.

```

:
: MICRO PROCESSOR COMMANDS
:
:           :          :UNIT: OP :
:           :          : ID :CODE:   DESCRIPTION
:           :-----:-----:-----:-----:
: FAT#1    :FAT1STAT: 0 : 0 : READ FAT#1 STATUS
:           :FAT1RST: 0 : 1 : RESET FAT#1
:           :FAT1LD4: 0 : 2 : LOAD 4 WORDS INTO FAT#1
:           :FAT1LD6: 0 : 3 : LOAD 6 WORDS INTO FAT#1
:           :FAT1RD4: 0 : 4 : READ 4 WORDS FROM FAT#1
:           :FAT1CW  : 0 : 5 : FAT#1 CW ON
:           :FAT1DIAG: 0 : 7 : FAT#1 DIAGNOSTIC MODE
:           :          :   :   :
: FAT#2    :FAT2STAT: 1 : 0 : READ FAT#2 STATUS
:           :FAT2RST: 1 : 1 : RESET FAT#2
:           :FAT2LD4: 1 : 2 : LOAD 4 WORDS INTO FAT#2
:           :FAT2LD6: 1 : 3 : LOAD 6 WORDS INTO FAT#2
:           :FAT2RD4: 1 : 4 : READ 4 WORDS FROM FAT#2
:           :FAT2CW  : 1 : 5 : FAT#2 CW ON
:           :FAT2DIAG: 1 : 7 : FAT#2 DIAGNOSTIC MODE
:           :          :   :   :
: FAT#3    :FAT3STAT: 2 : 0 : READ FAT#3 STATUS
:           :FAT3RST: 2 : 1 : RESET FAT#3
:           :FAT3LD4: 2 : 2 : LOAD 4 WORDS INTO FAT#3
:           :FAT3LD6: 2 : 3 : LOAD 6 WORDS INTO FAT#3
:           :FAT3RD4: 2 : 4 : READ 4 WORDS FROM FAT#3
:           :FAT3CW  : 2 : 5 : FAT#3 CW ON
:           :FAT3DIAG: 2 : 7 : FAT#3 DIAGNOSTIC MODE
:           :          :   :   :
: FAT#4    :FAT4STAT: 3 : 0 : READ FAT#4 STATUS
:           :FAT4RST: 3 : 1 : RESET FAT#4
:           :FAT4LD4: 3 : 2 : LOAD 4 WORDS INTO FAT#4
:           :FAT4LD6: 3 : 3 : LOAD 6 WORDS INTO FAT#4
:           :FAT4RD4: 3 : 4 : READ 4 WORDS FROM FAT#4
:           :FAT4CW  : 3 : 5 : FAT#4 CW ON
:           :FAT4DIAG: 3 : 7 : FAT#4 DIAGNOSTIC MODE
:           :          :   :   :
: ATRCBS RP6 :RSTARPG: 4 : 1 : RESET ATRCBS RP6
:           :LDARPG  : 4 : 2 : LOAD 4 WORDS INTO ATRCBS RP6
:           :RDARPG  : 4 : 4 : READ 4 WORDS FROM ATRCBS RP6
:           :          :   :   :
: MODE S RP6 :RSTMRP6: 5 : 1 : RESET MODE S RP6
:           :LDMRP6  : 5 : 2 : LOAD 4 WORDS INTO MODE S RP6
:           :ROMRPGA: 5 : 4 : READ 4 WORDS FROM MODE S RP6
:           :ROMRPG6: 5 : 5 : READ 6 WORDS FROM MODE S RP6
:           :          :   :   :
: FRG INTERFACE :IFSTATUS: 7 : 1 : READ FRG I/F STATUS
:           : READ4  : 7 : 2 : READ 4 WORDS FROM FRG I/F
:           : READ6  : 7 : 3 : READ 6 WORDS FROM FRG I/F
:           : LOAD4  : 7 : 4 : LOAD 4 WORDS INTO FRG I/F
:           : LOAD6  : 7 : 5 : LOAD 6 WORDS INTO FRG I/F
:           :UPSTATUS: 7 : 6 : LOAD USTATUS IN FRG STATUS REG
:           :FRGATN  : 7 : 7 : ISSUE FRG INTERRUPT
:           :          :   :   :
:
:
:

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 1 of 24)



```

:
:
0000 FAT1STAT: EQU H#0000
0010 FAT2STAT: EQU H#0010
0020 FAT3STAT: EQU H#0020
0030 FAT4STAT: EQU H#0030
0001 FAT1RST: EQU H#0001
0011 FAT2RST: EQU H#0011
0021 FAT3RST: EQU H#0021
0031 FAT4RST: EQU H#0031
0002 FAT1LD4: EQU H#0002
0012 FAT2LD4: EQU H#0012
0022 FAT3LD4: EQU H#0022
0032 FAT4LD4: EQU H#0032
0003 FAT1LD6: EQU H#0003
0013 FAT2LD6: EQU H#0013
0023 FAT3LD6: EQU H#0023
0033 FAT4LD6: EQU H#0033
0004 FAT1RD4: EQU H#0004
0014 FAT2RD4: EQU H#0014
0024 FAT3RD4: EQU H#0024
0034 FAT4RD4: EQU H#0034
0005 FAT1CW: EQU H#0005
0015 FAT2CW: EQU H#0015
0025 FAT3CW: EQU H#0025
0035 FAT4CW: EQU H#0035
0007 FAT1DIAG: EQU H#0007
0017 FAT2DIAG: EQU H#0017
0027 FAT3DIAG: EQU H#0027
0037 FAT4DIAG: EQU H#0037
0041 RSTARPG: EQU H#0041
0042 LDARPG: EQU H#0042
0044 RDARPG: EQU H#0044
0051 RSTMRPG: EQU H#0051
0052 LDMRPG: EQU H#0052
0054 RDMRPGA: EQU H#0054
0055 RDMRPGB: EQU H#0055
0071 IFSTATUS: EQU H#0071
0072 READ4: EQU H#0072
0073 READ6: EQU H#0073
0074 LOAD4: EQU H#0074
0075 LOAD6: EQU H#0075
0076 UPSTATUS: EQU H#0076
0077 FRGATN: EQU H#0077

```

```

:
: REGISTER ALLOCATIONS
:
: R0 - R5 = RESERVED FOR DATA TRANSFER
: R6 = PREVIOUS MODE S REPLY TRIGGER TIME
: R7 = PREVIOUS ATCRBS REPLY TRIGGER TIME (T1)
: R8 = PREVIOUS ATCRBS REPLY TRIGGER TIME (T2)
: R9 - R14 = RESERVED FOR FAT COMMANDS
: R15 = GENERAL PURPOSE REGISTER
:
:

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 2 of 24)



```

0015          OUT R0          ;WORD#1 = FIXED CODE
0016          OUT R1          ;WORD#2 = M/S RATIO & % FIXED CODE
0017          OUT R2          ;WORD#3 = ATRCBS FRUIT RATE SCALING FACTOR
0018          OUT R3          ;WORD#4 = SPARE
;
;::: INITIALIZE MODE S RPG FRUIT PARAMETERS
;
0019          LIMCR LDMRPG    ;LOAD FRUIT PARAMETERS INTO RPG
001A          RSTCR          ;CLEAR COMMAND REGISTER
001B          OUT R4          ;WORD#1 = SPARE
001C          OUT R5          ;WORD#2 = M/S RATIO & % LONG/SHORT
001D          OUT R6          ;WORD#3 = MODE S FRUIT RATE SCALING FACTOR
001E          OUT R7          ;WORD#4 = SPARE
;
001F          LIMCR RSTARPG  ;RESET ATRCBS RPG
0020          LIMCR RSTM RPG  ;RESET MODE S RPG
0021          RSTCR          ;CLEAR COMMAND REGISTER
;
0022          LIM DELAY,R15   ;SET UP 50 US DELAY
0023 LOOPG:   DEC R15        ;DECREMENT DELAY VALUE
0024          BNZ LOOPG      ;LOOP UNTIL DELAY EQUALS ZERO
;
;::: INITIALIZE DELAY-TO-TRIGGER TIME
;
0025          LIM H#0020,R6
0026          LIM H#001E,R7
0027          LIM H#001E,R8
;
;::: MAIN PROGRAM
;
0028 BEGIN:   LIMCR IFSTATUS  ;LOAD THE "I/F CMD & STATUS" WORD
0029          NOP             ;INTO THE Q-REG. TEST THE OPER-
002A BEGINI:  INQ            ;ATION FIELD TO DETERMINE WHETHER
002B          ANDIQ NORMASK   ;TO RUN IN THE NORMAL OR THE
002C          BNZ DIAG        ;DIAGNOSTIC MODE
;
;::: NORMAL MODE
;
002D          ANDI NOFRUIT,R13,R13 ;CHECK FOR MODE S FRUIT ENVIRONMENT.
002E          BNZ STEP1       ;IF NONE BRANCH TO STEP1.
;
002F          ANDI NOFRUIT,R12,R12 ;CHECK FOR ATRCBS FRUIT ENVIRONMENT.
0030          BNZ STEP2.S     ;IF NONE BRANCH TO STEP2.S.
;                               ;OTHERWISE FULL FRUIT LOAD.
0031 FULLLOAD: LIM FAT1LD4,R11 ;LOAD AN ATRCBS FRUIT REPLY PARAMETER
0032          LIM FAT1STAT,R14 ;BLOCK INTO FAT1.
0033          JSR A.FAT123
;
0034          JSR M.FAT4       ;LOAD FAT4 IF IT IS NOT BUSY.
;
0035          LIM FAT2LD4,R11 ;LOAD AN ATRCBS FRUIT REPLY PARAMETER
0036          LIM FAT2STAT,R14 ;BLOCK INTO FAT2.
0037          JSR A.FAT123
;
0038          JSR M.FAT4       ;LOAD FAT4 IF IT IS NOT BUSY.
;

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 4 of 24)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

```

0039          LIM FAT3LD4,R11 ;LOAD AN ATRCBS FRUIT REPLY PARAMETER
003A          LIM FAT3STAT,R14 ;BLOCK INTO FAT3.
003B          JSR A.FAT123
;
003C          JSR M.FAT4      ;LOAD FAT4 IF IT IS NOT BUSY.
;
003D          INQ             ;CHECK WHETHER NEW FRUIT PARAMETERS
003E          ANDIQ PARMASK   ;ARE AVAILABLE. IF THEY ARE UPDATE
003F          BRZ FULLLOAD   ;THE RPG PARAMETERS. OTHERWISE
;                               ;CONTINUE LOADING FAT'S.
0040          JSR NEWPARM     ;UPDATE FRUIT PARAMETERS.
;
0041          JMP BEGIN
;
0042 STEP1:   ANDI NOFRUIT,R12,R12 ;CHECK FOR ATRCBS FRUIT ENVIRONMENT.
0043          BNZ STEP2      ;IF NONE, BRANCH TO STEP2.
;
0044          LIM FAT1LD4,R11 ;LOAD AN ATRCBS FRUIT REPLY PARAMETER
0045          LIM FAT1STAT,R14 ;BLOCK INTO FAT1.
0046          JSR A.FAT123
;
0047 STEP2:   ANDI NOFRUIT,R13,R13 ;CHECK FOR MODE S FRUIT ENVIRONMENT.
0048          BNZ STEP3      ;IF NONE, BRANCH TO STEP3.
;
0049 STEP2.S: JSR M.FAT4      ;LOAD FAT4 IF IT IS NOT BUSY.
;                               ;OTHERWISE CONTINUE.
004A STEP3:   ANDI NOFRUIT,R12,R12 ;CHECK FOR ATRCBS FRUIT ENVIRONMENT.
004B          BNZ STEP4      ;IF NONE, BRANCH TO STEP4.
;
004C          LIM FAT2LD4,R11 ;LOAD AN ATRCBS FRUIT REPLY PARAMETER
004D          LIM FAT2STAT,R14 ;BLOCK INTO FAT2.
004E          JSR A.FAT123
;
004F STEP4:   ANDI NOFRUIT,R13,R13 ;CHECK FOR MODE S FRUIT ENVIRONMENT.
0050          BNZ STEPS      ;IF NONE, BRANCH TO STEPS.
;
0051          JSR M.FAT4      ;LOAD FAT4 IF IT IS NOT BUSY.
;                               ;OTHERWISE CONTINUE.
0052 STEPS:   ANDI NOFRUIT,R12,R12 ;CHECK FOR ATRCBS FRUIT ENVIRONMENT.
0053          BNZ STEPS6     ;IF NONE, BRANCH TO STEPS6.
;
0054          LIM FAT3LD4,R11 ;LOAD AN ATRCBS FRUIT REPLY PARAMETER
0055          LIM FAT3STAT,R14 ;BLOCK INTO FAT3.
0056          JSR A.FAT123
;
0057 STEPS6:   ANDI NOFRUIT,R13,R13 ;CHECK FOR MODE S FRUIT ENVIRONMENT.
0058          BNZ TPARM      ;IF NONE, BRANCH BACK TO BEGIN.
;
0059          JSR M.FAT4      ;LOAD FAT4 IF IT IS NOT BUSY.
;                               ;OTHERWISE CONTINUE.
005A TPARM:   INQ             ;CHECK WHETHER NEW FRUIT PARAMETERS
005B          ANDIQ PARMASK   ;ARE AVAILABLE. IF THEY ARE UPDATE
005C          BRZ BEGIN1     ;THE RPG PARAMETERS. OTHERWISE
;                               ;CONTINUE LOADING FATS.
005D          JSR NEWPARM     ;-DATE FRUIT PARAMETERS.
;
005E          JMP BEGIN

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 5 of 24)

```

: SUBROUTINE "LOAD MODE S FAT"
: THIS ROUTINE FIRST TESTS TO DETERMINE WHETHER FAT4 IS BUSY.
: IF IT IS BUSY, THE SUBROUTINE TERMINATES. IF IT IS NOT BUSY,
: A MODE S REPLY PARAMETER BLOCK IS READ FROM THE MODE S RP6
: AND LOADED INTO THE MODE S FAT GENERATOR.
:
005F M.FAT4: LIMCR FAT4STAT ;TEST STATUS OF FAT4
0060 RSTCR ;IF FAT4 IS BUSY RETURN,
0061 LIMCR IFSTATUS ;OTHERWISE STUFF ONE MODE S
0062 INQ ;FRUIT REPLY PARAMETER
0063 ANDIQ BUSY ;BLOCK INTO FAT4.
0064 BRZ LOADFAT4 ;IF ZERO, FAT IS IDLE.
0065 RTN

;
0066 LOADFAT4: LIMCR ROMRPGA ;GET THE FIRST FOUR WORDS OF THE REPLY
0067 RSTCR ;PARAMETER BLOCK FROM THE MODE S
0068 NOP ;RANDOM PROCESS GENERATOR.
0069 NOP ;
006A NOP ;
006B NOP ;
006C IN R0 ;UPPER 16 BITS OF REPLY TIME
006D IN R1 ;LOWER 4 BITS OF REPLY TIME, FRUIT TYPE
006E IN R2 ;POWER, MONOPULSE, LR BIT
006F IN R3 ;FIRST WORD OF MODE S REPLY.
;R0(REPLY TIME) = R0(NEW DTT)
0070 ADDI MTD,R0,R6 ;-128US + R0(DDT) --> R6(DDT - 128US)
0071 BRC DUMP ;WHERE MTD = 128 US.
0072 COM R6 ;/R6(DDT - 128US) --. R6(128US - DDT)
0073 ADD R6,R0 ;ADD R6(128US - DDT) + R0(DDT) --> R0(128US)

;
0074 DUMP: LIMCR FAT4LD4 ;LOAD THE FIRST FOUR WORDS OF THE REPLY
0075 OUT R0 ;PARAMETER BLOCK INTO THE FAT.
0076 OUT R1
0077 OUT R2
0078 OUT R3

;
0079 LIMCR ROMRPG6 ;GET THE NEXT SIX WORDS OF THE REPLY
007A RSTCR ;PARAMETER BLOCK FROM THE MODE S
007B NOP ;RANDOM PROCESS GENERATOR.
007C NOP
007D NOP
007E NOP
007F IN R0
0080 IN R1
0081 IN R2
0082 IN R3
0083 IN R4
0084 IN R5
0085 LIMCR FAT4LD6 ;LOAD THE LAST SIX WORDS OF THE REPLY
0086 OUT R0 ;PARAMETER BLOCK INTO THE FAT.
0087 OUT R1
0088 OUT R2
0089 OUT R3
008A OUT R4
008B OUT R5
008C LIMCR IFSTATUS ;RESET CONTROL REGISTER.

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 6 of 24)

```

008D             RTN             ;RETURN FROM SUBROUTINE.
:
: SUBROUTINE "LOAD ATCRBS FAT'S"
: IT WILL CONTINUE TO CHECK UNTIL THE FAT BECOMES AVAILABLE.
: AT THAT TIME AN ATCRBS REPLY PARAMETER BLOCK IS READ FROM THE
: ATCRBS RPG AND LOADED INTO THE ATCRBS FAT GENERATOR.
:
008E A.FAT123:  LCR R14
008F             LIM TIMER1,R15 ;SET UP ATCRBS TIMEOUT LOOP.
0090             NOP
0091 LOOPB:     INQ             ;TEST STATUS OF FAT. IF TIMEOUT OCCURS
0092             ANDIQ BUSY     ;BEFORE FAT BECOMES AVAILABLE JUMP TO
0093             BRZ LOADAFAT   ;FAULTA ROUTINE. WHEN THE FAT BECOMES
0094             DEC R15        ;AVAILABLE LOAD ONE ATCRBS REPLY
0095             BNZ LOOPB      ;PARAMETER BLOCK INTO IT.
0096             JMP FAULTA    ;

;
0097 LOADAFAT:  LIMCR ROARPG   ;GET A REPLY PARAMETER
0098             RSTCR         ;BLOCK FROM THE ATCRBS
0099             ADD R7,R8     ;RANDOM PROCESS GENERATOR.
009A             ADDI ATD,R8,R8
009B             BRC ATC2
009C             NOP
009D             IN R0
009E             IN R1
009F             IN R2
00A0             IN R3

;
; ADD R7,R8          ;IF THE SUM (X) OF THE CURRENT TRIGGER
; ADDI ATD,R8,R8    ;TIME (TC) AND THE PREVIOUS TWO IS
; BRC ATC           ;LESS THAN 30 US, THEN ADD (30US -X)
00A1             ADD R0,R8    ;TO THE CURRENT TIME.
00A2             BRC ATC
00A3             COM R8
00A4             ADD R8,R0

;
00A5 ATC:       MOV R7,R8     ;PUT T1 INTO R8 (NEW T2)
00A6             MOV R0,R7    ;PUT TC INTO R7 (NEW T1)

;
00A7             LCR R11     ;TRANSFER REPLY PARAMETER
00A8             OUT R0      ;BLOCK TO FAT.
00A9             OUT R1
00AA             OUT R2
00AB             OUT R3
00AC             LIMCR IFSTATUS ;CLEAR COMMAND REGISTER.
00AD             RTN        ;RETURN FROM SUBROUTINE.

;
00AE ATC2:     NOP
00AF             IN R0
00B0             IN R1
00B1             IN R2
00B2             IN R3

;
00B3             MOV R7,R8   ;PUT T1 INTO R8 (NEW T2)
00B4             MOV R0,R7   ;PUT TC INTO R7 (NEW T1)

;
00B5             LCR R11     ;TRANSFER REPLY PARAMETER
    
```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 7 of 24)



```

00DF      NOP      ;COMMAND CLOCKED INTO CONTROL REGISTER.
00E0      NOP      ;CMD CLOCKED INTO INPUT LATCH OF GEN.
00E1      NOP      ;FIRST DIAGNOSTIC WORD TO D-BUS.
00E2      IN R0
00E3      IN R1
00E4      IN R2
00E5      IN R3
00E6      RSTCR    ;CLEAR COMMAND REGISTER.

1
00E7 LOADDATA: LIMCR IFSTATUS ;LOAD THE "I/F COMMAND & STATUS" WORD
00E8      NOP      ;INTO THE Q-REG. DETERMINE WHETHER
00E9 LOOPI:  INQ     ;THE PE-RAM IS AVAILABLE FOR RECEIVING
00EA      ANDIQ PEMASK ;DATA. CONTINUE CHECKING UNTIL IT
00EB      BRZ LOOPI  ;BECOMES AVAILABLE.

1
00EC      LIMCR LOAD4 ;WHEN THE PE-RAM BECOMES AVAILABLE
00ED      RSTCR    ;TRANSFER DIAGNOSTIC DATA INTO IT.
00EE      OUT R0
00EF      OUT R1
00F0      OUT R2
00F1      OUT R3
00F2      JMP BEGIN

; ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
; :::::::::::::: ATCRBS RPG VALIDATION ROUTINE ::::::::::::::::::::::::::::
; ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
; THIS ROUTINE TAKES ONE ATCRBS FRUIT REPLY PARAMETER BLOCK FROM
; THE ATCRBS RPG AND LOADS IT INTO THE FRG I/F PE-RAM FOR ACCESS
; BY THE CPU.
; ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

00F3 ARPGDIAG: LIMCR IFSTATUS ;LOAD THE "I/F COMMAND & STATUS" WORD
00F4      NOP      ;INTO THE Q-REG. TEST FOR NEW FRUIT
00F5      INQ     ;ENVIRONMENT PARAMETERS. IF AVAILABLE
00F6      ANDIQ PARMASK ;JUMP TO SUBROUTINE "NEWPARM" TO
00F7      BRZ ARPG  ;UPDATE THE RANDOM PROCESS GENERATORS.
00F8      JSR NEWPARM ;SUBROUTINE TO LOAD RPG'S W/ NEW PARAMETERS.

1
00F9 ARPG:    LIMCR IFSTATUS ;CHECK THE "I/F COMMAND & STATUS" WORD
00FA      NOP      ;TO DETERMINE WHETHER THE PE-RAM IS
00FB LOOPC:  INQ     ;AVAILABLE FOR RECEIVING DATA.
00FC      ANDIQ PEMASK
00FD      BRZ LOOPC  ;LOOP UNTIL THE PE-RAM BECOMES AVAILABLE.

1
00FE      LIMCR RDARPG ;GET A FRUIT REPLY PARAMETER BLOCK FROM
00FF      RSTCR    ;THE ATCRBS RANDOM PROCESS GENERATOR.
0100      NOP
0101      NOP
0102      NOP
0103      NOP
0104      IN R0
0105      IN R1
0106      IN R2
0107      IN R3
0108      LIMCR LOAD4 ;TRANSFER ATCRBS FRUIT REPLY PARAMETER
0109      RSTCR    ;BLOCK INTO THE PE-RAM.
010A      OUT R0
010B      OUT R1
010C      OUT R2

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 9 of 24)





```

013D      OUT RS
013E      JMP BEGIN
;
; .....
; ..... MICROPROCESSOR LOOPBACK VALIDATION ROUTINE .....
; .....
; THIS ROUTINE READS A FOUR WORD DATA BLOCK FROM THE REPLY-RAM
; WHEN IT IS IN THE READ MODE AND TRANSFERS THE DATA TO THE
; PE-RAM WHEN IT IS IN THE WRITE MODE VIA MICROPROCESSOR LINK.
; .....
013F UPRCDIAG: LIMCR IFSTATUS ;CHECK THE I/F STATUS WORD TO DETERMINE
0140      NOP ;WHETHER THE REPLY-RAM IS IN THE
0141      INQ ;READ MODE. IF IT IS NOT, CHECK
0142      ANDIQ RPLYMASK ;WHETHER THE MICROPROCESSOR DIAGNOSTIC
0143      BRZ BEGIN ;TEST IS STILL SELECTED.
;
0144      LIMCR READ4 ;IF IT IS IN THE READ MODE,
0145      RSTCR ;READ FOUR WORDS FROM THE RAM.
0146      NOP
0147      IN R0
0148      IN R1
0149      IN R2
014A      IN R3
014B      LIMCR IFSTATUS ;LOOP UNTIL THE PE-RAM IS IN
014C      NOP ;THE WRITE MODE.
014D LOOPF: INQ
014E      ANDIQ PEMASK
014F      BRZ LOOPF
;
0150      LIMCR LOAD4 ;LOAD FOUR WORDS INTO THE PE-RAM.
0151      RSTCR
0152      OUT R0
0153      OUT R1
0154      OUT R2
0155      OUT R3
0156      JMP BEGIN
;
; .....
; ..... SUBROUTINE "SELECT FAT" .....
; .....
; THIS ROUTINE IS USED BY THE "FRGLOOP" TEST AND THE "REPLY
; GENERATOR LOOPBACK" TEST. IT IDENTIFIES WHICH FAT WAS
; SELECTED AND SETS UP ALL THE APPROPRIATE COMMANDS FOR THAT FAT.
; .....
0157 SELFAT: LIMCR IFSTATUS ;
0158      NOP ;
0159      IN R11 ; R11 = :X:X:X:0:0:0:0:0:0:0:0:0:0:G:0:0:0:
015A      SRA R11 ; R11 = :0:X:X:X:0:0:0:0:0:0:0:0:0:0:G:G:0:0:
015B      SRA R11 ; R11 = :0:0:X:X:X:0:0:0:0:0:0:0:0:0:0:G:G:0:
015C      SRA R11 ; R11 = :0:0:0:X:X:X:0:0:0:0:0:0:0:0:0:0:G:G:
015D      LIM GOMASK,R12 ; R12 = H$0003
015E      AND R12,R11 ; R11 = :0:0:0:0:0:0:0:0:0:0:0:0:0:0:G:G:
015F      BRZ SELFAT1
0160      DEC R11
0161      BRZ SELFAT2
0162      DEC R11
0163      BRZ SELFAT3
;
0164 SELFAT4: LIM FAT4RST,R9 ;SET UP TO EXECUTE FAT4 COMMANDS.

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 11 of 24)

```

0165          LIM FAT4DIAG,R10
0166          LIM FAT4LD4,R11
0167          LIM FAT4LD6,R12
0168          LIM FAT4RD4,R13
0169          LIM FAT4STAT,R14
016A          LIM H*0108,R15 ;254 MILLISECOND TIMEOUT VALUE.
016B          LIM FAT4CW,R8
016C          RTN

;
016D SELFAT3: LIM FAT3RST,R9 ;SET UP TO EXECUTE FAT3 COMMANDS.
016E          LIM FAT3DIAG,R10
016F          LIM FAT3LD4,R11
0170          LIM FAT3LD6,R12
0171          LIM FAT3RD4,R13
0172          LIM FAT3STAT,R14
0173          LIM H*0046,R15 ;70 MILLISECOND TIMEOUT VALUE.
0174          LIM FAT3CW,R8
0175          RTN

;
0176 SELFAT2: LIM FAT2RST,R9 ;SET UP TO EXECUTE FAT2 COMMANDS.
0177          LIM FAT2DIAG,R10
0178          LIM FAT2LD4,R11
0179          LIM FAT2LD6,R12
017A          LIM FAT2RD4,R13
017B          LIM FAT2STAT,R14
017C          LIM H*0046,R15 ;70 MILLISECOND TIMEOUT VALUE.
017D          LIM FAT2CW,R8
017E          RTN

;
017F SELFAT1: LIM FAT1RST,R9 ;SET UP TO EXECUTE FAT1 COMMANDS.
0180          LIM FAT1DIAG,R10
0181          LIM FAT1LD4,R11
0182          LIM FAT1LD6,R12
0183          LIM FAT1RD4,R13
0184          LIM FAT1STAT,R14
0185          LIM H*0046,R15 ;70 MILLISECOND TIMEOUT VALUE.
0186          LIM FAT1CW,R8
0187          RTN

;
; SUBROUTINE "LOAD FAT"
; THIS ROUTINE FIRST TESTS TO DETERMINE WHETHER A REPLY PARAMETER
; IS AVAILABLE IN THE FRG I/F REPLY RAM. WHEN ONE BECOMES
; AVAILABLE, IT IS READ FROM THE REPLY RAM AND LOADED INTO THE
; PREDETERMINED FRUIT ARIES TARGET (FAT) GENERATOR.
;
0188 DIAGFAT: LIMCR IFSTATUS ;CHECK FOR A REPLY PARAMETER BLOCK
0189          NOP ;IN THE INTERFACE REPLY RAM.
018A LOOPE:   INQ
018B          ANDIQ RPLYMASK
018C          BRZ LOOPE ;LOOP UNTIL A REPLY BLOCK BECOMES
; AVAILABLE.
018D          LCR R9 ;RESET SELECTED FRUIT ARIES
018E          RSTCR ;TARGET GENERATOR.

;
018F          LIMCR IFSTATUS ;LOAD THE I/F CMD-STAT WORD
0190          NOP ;INTO THE Q-REG. DETERMINE WHETHER
0191          INQ ;THE REPLY GENERATOR OUTPUT IS TO BE A

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 12 of 24)

```

0192      ANDIQ CWMSK      :CODE TRAIN OR A CW SIGNAL.
0193      BRZ LOADFAT
0194      LCR R8           ;IF YES, SET CW OUTPUT ON SELECTED
0195      RSTCR           ;REPLY GENERATOR.
0196 LOADFAT: LIMCR READ4  ;READ FOUR WORDS FROM THE FRG
0197      RSTCR           ;INTERFACE REPLY-RAM.
0198      NOP             ;NOP REQUIRED FOR PROPER H/W TIMING.
0199      IN R0
019A      IN R1
019B      IN R2
019C      IN R3

;
019D      LCR R11          ;LOAD THE FAT WITH WORDS 1-4.
019E      OUT R0
019F      OUT R1
01A0      OUT R2
01A1      OUT R3
01A2      RSTCR

;
01A3      ANDI TYPE,R1,R1 ;CHECK WHETHER THE REPLY PARAMETER
01A4      BNZ MREPLY      ;IS MODE S TYPE. IF SO CONTINUE
;                               ;LOADING. OTHERWISE RETURN.
01A5      LCR R10
01A6      RSTCR
01A7      RTN

;
01A8 MREPLY: LIMCR READ6  ;READ THE NEXT SIX WORDS FROM THE FRG
01A9      RSTCR          ;INTERFACE REPLY-RAM.
01AA      NOP
01AB      IN R0
01AC      IN R1
01AD      IN R2
01AE      IN R3
01AF      IN R4
01B0      IN R5

;
01B1      LCR R12          ;LOAD THE FAT WITH WORDS 5-10.
01B2      OUT R0
01B3      OUT R1
01B4      OUT R2
01B5      OUT R3
01B6      OUT R4
01B7      OUT R5
01B8      LCR R10         ;SET /FATDIAG BIT TO OVERRIDE G02 F/F.
01B9      RSTCR
01BA      RTN

;
;=====
;          SUBROUTINE "UPDATE FRUIT PARAMETERS"
;=====
; THIS ROUTINE TRANSFERS THE ATRCBS AND MODE S FRUIT ENVIRONMENT
; PARAMETERS FROM THE FRG I/F PARAMETER RAM INTO THE ATRCBS AND
; MODE S RANDOM PROCESS GENERATORS, RESPECTIVELY. IT THEN,
; RESETS THE RANDOM NUMBER GENERATORS IN BOTH RP6'S AND ALLOWS
; 50 MICROSECONDS TO PASS TO GIVE THE RP6'S TIME TO BUILD REPLY
; PARAMETERS BEFORE ATTEMPTING TO GET THEM.
;=====
01BB NEWPARM: LIMCR READ4 ;GET NEW ATRCBS FRUIT PARAMETERS
01BC      RSTCR

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 13 of 24)

```

0180      NOP
018E      IN R0      ;WORD#1 = FIXED CODE
018F      IN R1      ;WORD#2 = M/S RATIO & %FIXED CODE
01C0      IN R2      ;WORD#3 = ATRCBS FRUIT RATE SCALING FACTOR
01C1      IN R3      ;WORD#4 = SPARE
01C2      MOV R2,R12 ;SAVE ATRCBS FRUIT RATE SCALING FACTOR
01C3      LIMCR LDARPG ;UPDATE ATRCBS RPG PARAMETERS.
01C4      RSTCR      ;RESET COMMAND REGISTER.
01C5      OUT R0      ;WORD#1 = FIXED CODE
01C6      OUT R1      ;WORD#2 = M/S RATIO & %FIXED CODE
01C7      OUT R2      ;WORD#3 = ATRCBS FRUIT RATE SCALING FACTOR
01C8      OUT R3      ;WORD#4 = SPARE
01C9      LIMCR READ4 ;GET NEW MODE S FRUIT PARAMETERS.
01CA      RSTCR
01CB      NOP
01CC      IN R0      ;WORD#1 = SPARE
01CD      IN R1      ;WORD#2 = M/S RATIO & %LONG/SHORT
01CE      IN R2      ;WORD#3 = MODE S FRUIT RATE SCALING FACTOR
01CF      IN R3      ;WORD#4 = SPARE
01D0      MOV R2,R13 ;SAVE MODE S FRUIT RATE SCALING FACTOR.
01D1      LIMCR LDMRPG ;UPDATE MODE S RPG PARAMETERS.
01D2      RSTCR      ;RESET COMMAND REGISTER.
01D3      OUT R0      ;WORD#1 = SPARE
01D4      OUT R1      ;WORD#2 = M/S RATIO & %LONG/SHORT
01D5      OUT R2      ;WORD#3 = MODE S FRUIT RATE SCALING FACTOR
01D6      OUT R3      ;WORD#4 = SPARE

;
01D7      LIMCR FAT1RST ;INITIALIZE ALL FRUIT ARIES
01D8      LIMCR FAT2RST ;TARGET (FAT) GENERATORS
01D9      LIMCR FAT3RST
01DA      LIMCR FAT4RST
01DB      RSTCR
;
; ::::::::::::::: INITIALIZE DELAY-TO-TRIGGER TIME
;
01DC      LIM H#0020,R6
01DD      LIM H#001E,R7
01DE      LIM H#001E,R8
;
01DF      LIM DELAY,R15 ;SET UP 50 US DELAY
01E0 LOOPK: DEC R15      ;DECREMENT DELAY VALUE
01E1      BNZ LOOPK      ;LOOP UNTIL DELAY EQUALS ZERO
01E2      RTN            ;RETURN FROM SUBROUTINE.
;
; ::::::::::::::: FAULT "ATRCBS TIMEOUT OCCURRED" :::::::::::::::
;
01E3 FAULTA: LIM TIMEOUT,R15 ;SET ATRCBS TIMEOUT FLAG IN
01E4      LIMCR UPSTATUS ;MICROPROCESSOR STATUS REGISTER
01E5      RSTCR          ;WHEN FAT FAILS TO RESPOND WITHIN
01E6      OUT R15        ;5 MS THEN GENERATE AN INTERRUPT
01E7      LIMCR FRGATN   ;BACK TO THE CPU.
01E8      RSTCR
01E9      JPZ           ;REINITIALIZE FRUIT REPLY GENERATOR.
;
; :::::::::::::::
01EA END

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 14 of 24)

```

0000 0000000000000001 0111000010111110 0000000000000000 00011110
0001 0000000000010001 0111000010111110 0000000000000000 00011110
0002 0000000000010001 0111000010111110 0000000000000000 00011110
0003 00000000000110001 0111000010111110 0000000000000000 00011110
0004 0000000011111111 0111000010111110 0000000000000000 00011110
0005 0000000000000000 0111000111101110 0001111000000010 00010111
0006 0000000001101110 0111000010111110 0000000000000000 00011110
0007 0000000011111111 0111000010111110 0000000000000000 00011110
0008 0000000000000000 0111000010111001 1110000000000000 00011111
0009 10100000000100010 0111000111101110 0000000000000010 00010111
000A 0100000000001000 0111000111101110 0000001000000010 00010111
000B 0001000000000000 0111000111101110 0000010000000010 00010111
000C 0000000000000000 0111000111101110 0000010000000010 00010111
000D 0000000000000000 0111000111101110 0000100000000010 00010111
000E 0100000000001000 0111000111101110 0000101000000010 00010111
000F 0001000000000000 0111000111101110 0000110000000010 00010111
0010 0000000000000000 0111000111101110 0000111000000010 00010111
0011 0000000000000000 0111000111101000 0101100000000010 00010111
0012 0000000000000000 0111000111101000 1101101000000010 00010111
0013 0000000001000010 0111000010111110 0000000000000000 00011110
0014 0000000011111111 0111000010111110 0000000000000000 00011110
0015 0000000000000000 0111000010111000 0000000000000000 00011111
0016 0000000000000000 0111000010111000 0010000000000000 00011111
0017 0000000000000000 0111000010111000 0100000000000000 00011111
0018 0000000000000000 0111000010111000 0110000000000000 00011111
0019 0000000001010010 0111000010111110 0000000000000000 00011110
001A 0000000011111111 0111000010111110 0000000000000000 00011110
001B 0000000000000000 0111000010111000 1000000000000000 00011111
001C 0000000000000000 0111000010111000 1010000000000000 00011111
001D 0000000000000000 0111000010111000 1100000000000000 00011111
001E 0000000000000000 0111000010111000 1110000000000000 00011111
001F 0000000001000001 0111000010111110 0000000000000000 00011110
0020 0000000001010001 0111000010111110 0000000000000000 00011110
0021 0000000011111111 0111000010111110 0000000000000000 00011110
0022 0000000001100100 0111000111101110 0001111000000010 00010111
0023 0000000000000000 0111000110010110 0001111000000010 00010111
0024 0000000000010001 0001100010000000 0000000000000010 10010011
0025 0000000000100000 0111000111101110 0000110000000010 00010111
0026 0000000000011110 0111000111101110 0000111000000010 00010111
0027 0000000000001110 0111000111101110 0001000000000010 00010111
0028 0000000001110001 0111000010111110 0000000000000000 00011110
0029 0000000000000000 0111000010000000 0000000000000000 00011111
002A 0000000000000000 1111000000001110 0000000000000010 00010111
002B 0000000000000111 0111000010011000 0000000000000010 00010111
002C 0000000010111100 0001100010000000 0000000000000010 10010011
002D 0001000000000000 0111000111001011 1011101000000010 00010111
002E 0000000001000010 0001100010000000 0000000000000010 10010011
002F 0001000000000000 0111000111001011 1001100000000010 00010111
0030 0000000001001001 0001100010000000 0000000000000010 10010011
0031 0000000000000010 0111000111101110 0001011000000010 00010111
0032 0000000000000000 0111000111101110 0001110000000010 00010111
0033 0000000010001110 0000110010000000 0000000000000000 01001111
0034 0000000001011111 0000110010000000 0000000000000000 01001111
0035 0000000000010010 0111000111101110 0001011000000010 00010111
0036 0000000000010000 0111000111101110 0001110000000010 00010111
0037 0000000010001110 0000110010000000 0000000000000000 01001111
0038 0000000001011111 0000110010000000 0000000000000000 01001111

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 15 of 24)

```

0039 000000000100010 011100011110110 000101100000010 00010111
003A 000000000100000 011100011110110 000111000000010 00010111
003B 000000001000110 0000110010000000 000000000000000 01001111
003C 000000000101111 0000110010000000 000000000000000 01001111
003D 000000000000000 111100000001110 000000000000010 00010111
003E 100000000000000 0111000001001100 000000000000010 00010111
003F 0000000000110001 0001100010000000 000000000000010 10110011
0040 000000011011101 0000110010000000 000000000000000 01001111
0041 0000000000101000 0001110010000000 000000000000000 00011111
0042 0001000000000000 0111000111001011 100110000000010 00010111
0043 0000000001000111 0001100010000000 000000000000010 10010011
0044 0000000000000010 0111000111101110 000101100000010 00010111
0045 0000000000000000 0111000111101110 000111000000010 00010111
0046 0000000010001110 0000110010000000 000000000000000 01001111
0047 0001000000000000 0111000111001011 101110100000010 00010111
0048 0000000001001010 0001100010000000 000000000000010 10010011
0049 0000000001011111 0000110010000000 000000000000000 01001111
004A 0001000000000000 0111000111001011 100110000000010 00010111
004B 0000000001001111 0001100010000000 000000000000010 10010011
004C 0000000000010010 0111000111101110 000101100000010 00010111
004D 0000000000010000 0111000111101110 000111000000010 00010111
004E 0000000010001110 0000110010000000 000000000000000 01001111
004F 0001000000000000 0111000111001011 101110100000010 00010111
0050 0000000001010010 0001100010000000 000000000000010 10010011
0051 0000000001011111 0000110010000000 000000000000000 01001111
0052 0001000000000000 0111000111001011 100110000000010 00010111
0053 0000000001010111 0001100010000000 000000000000010 10010011
0054 0000000000100010 0111000111101110 000101100000010 00010111
0055 0000000000100000 0111000111101110 000111000000010 00010111
0056 0000000010001110 0000110010000000 000000000000000 01001111
0057 0001000000000000 0111000111001011 101110100000010 00010111
0058 0000000001011010 0001100010000000 000000000000010 10010011
0059 0000000001011111 0000110010000000 000000000000000 01001111
005A 0000000000000000 111100000001110 000000000000010 00010111
005B 1000000000000000 0111000001001100 000000000000010 00010111
005C 0000000000101010 0001100010000000 000000000000010 10110011
005D 0000000110111011 0000110010000000 000000000000000 01001111
005E 0000000000101000 0001110010000000 000000000000000 00011111
005F 0000000000110000 0111000001011110 000000000000000 00011110
0060 0000000011111111 0111000001011110 000000000000000 00011110
0061 0000000001110001 0111000001011110 000000000000000 00011110
0062 0000000000000000 111100000001110 000000000000010 00010111
0063 0000000000000001 0111000001001100 000000000000010 00010111
0064 0000000001100110 0001100010000000 000000000000010 10110011
0065 0000000000000000 0101010010000000 000000000000000 01010111
0066 0000000001010100 0111000001011110 000000000000000 00011110
0067 0000000011111111 0111000001011110 000000000000000 00011110
0068 0000000000000000 0111000001000000 000000000000000 00011111
0069 0000000000000000 0111000001000000 000000000000000 00011111
006A 0000000000000000 0111000001000000 000000000000000 00011111
006B 0000000000000000 0111000001000000 000000000000000 00011111
006C 0000000000000000 1111000110001110 000000000000010 00010111
006D 0000000000000000 1111000110001110 000000100000010 00010111
006E 0000000000000000 1111000110001110 000001000000010 00010111
006F 0000000000000000 1111000110001110 000001100000010 00010111
0070 111111111011111 0111000110001010 000011000000010 00010111
0071 000000000110100 0001100010000000 000000000000011 01110011

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 16 of 24)

```

0072 0000000000000000 011100011110110 000011000000010 00010111
0073 0000000000000000 011100011000010 110000000000010 00010111
0074 0000000000110010 011100001011110 000000000000000 00011110
0075 0000000000000000 011100001011100 000000000000000 00011111
0076 0000000000000000 011100001011100 001000000000000 00011111
0077 0000000000000000 011100001011100 010000000000000 00011111
0078 0000000000000000 011100001011100 011000000000000 00011111
0079 0000000001010101 011100001011110 000000000000000 00011110
007A 0000000011111111 011100001011110 000000000000000 00011110
007B 0000000000000000 011100001000000 000000000000000 00011111
007C 0000000000000000 011100001000000 000000000000000 00011111
007D 0000000000000000 011100001000000 000000000000000 00011111
007E 0000000000000000 011100001000000 000000000000000 00011111
007F 0000000000000000 111100011000110 000000000000010 00010111
0080 0000000000000000 111100011000110 000000100000010 00010111
0081 0000000000000000 111100011000110 000001000000010 00010111
0082 0000000000000000 111100011000110 000001100000010 00010111
0083 0000000000000000 111100011000110 000010000000010 00010111
0084 0000000000000000 111100011000110 000010100000010 00010111
0085 0000000000110011 011100001011110 000000000000000 00011110
0086 0000000000000000 011100001011100 000000000000000 00011111
0087 0000000000000000 011100001011100 001000000000000 00011111
0088 0000000000000000 011100001011100 010000000000000 00011111
0089 0000000000000000 011100001011100 011000000000000 00011111
008A 0000000000000000 011100001011100 100000000000000 00011111
008B 0000000000000000 011100001011100 101000000000000 00011111
008C 0000000001110001 011100001011110 000000000000000 00011110
008D 0000000000000000 010101001000000 000000000000000 01010111
008E 0000000000000000 011100001011100 110000000000000 00011110
008F 0000111110011110 011100011110110 000111100000010 00010111
0090 0000000000000000 011100001000000 000000000000000 00011111
0091 0000000000000000 111100000000110 000000000000010 00010111
0092 0000000000000001 011100001001100 000000000000010 00010111
0093 0000000010010111 000110001000000 000000000000010 10110011
0094 0000000000000000 0111000110010110 000111100000010 00010111
0095 0000000010010001 000110001000000 000000000000010 10010011
0096 0000000111100011 000111001000000 000000000000000 00011111
0097 0000000001000100 011100001011110 000000000000000 00011110
0098 0000000011111111 011100001011110 000000000000000 00011110
0099 0000000000000000 011100011000010 111100000000010 00010111
009A 1111111111000001 0111000110001011 000100000000010 00010111
009B 0000000010101110 000110001000000 000000000000011 01110011
009C 0000000000000000 011100001000000 000000000000000 00011111
009D 0000000000000000 111100011000110 000000000000010 00010111
009E 0000000000000000 111100011000110 000000100000010 00010111
009F 0000000000000000 111100011000110 000001000000010 00010111
00A0 0000000000000000 111100011000110 000001100000010 00010111
00A1 0000000000000000 011100011000010 000100000000010 00010111
00A2 0000000010100101 000110001000000 000000000000011 01110011
00A3 0000000000000000 011100011110110 000100000000010 00010111
00A4 0000000000000000 011100011000011 000000000000010 00010111
00A5 0000000000000000 011100011110100 111100000000010 00010111
00A6 0000000000000000 011100011110100 000011100000010 00010111
00A7 0000000000000000 011100001011100 011000000000000 00011110
00A8 0000000000000000 011100001011100 000000000000000 00011111
00A9 0000000000000000 011100001011100 001000000000000 00011111
00AA 0000000000000000 011100001011100 010000000000000 00011111

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 17 of 24)



```

00AB 0000000000000000 011100001011000 0110000000000000 00011111
00AC 000000000110001 011100001011110 0000000000000000 00011110
00AD 0000000000000000 0101010010000000 0000000000000000 01010111
00AE 0000000000000000 0111000010000000 0000000000000000 00011111
00AF 0000000000000000 1111000110001110 0000000000000010 00010111
00B0 0000000000000000 1111000110001110 0000001000000010 00010111
00B1 0000000000000000 1111000110001110 0000010000000010 00010111
00B2 0000000000000000 1111000110001110 0000011000000010 00010111
00B3 0000000000000000 0111000111101000 1111000000000010 00010111
00B4 0000000000000000 0111000111101000 0000111000000010 00010111
00B5 0000000000000000 0111000010111001 0110000000000000 00011110
00B6 0000000000000000 0111000010111000 0000000000000000 00011111
00B7 0000000000000000 0111000010111000 0010000000000000 00011111
00B8 0000000000000000 0111000010111000 0100000000000000 00011111
00B9 0000000000000000 0111000010111000 0110000000000000 00011111
00BA 0000000001110001 0111000010111110 0000000000000000 00011110
00BB 0000000000000000 0101010010000000 0000000000000000 01010111
00BC 0000000000000000 0111000000101000 0000000000000010 00010111
00BD 0000000011000111 0001100010000000 0000000000000010 10110011
00BE 0000000000000000 0111000000101000 0000000000000001 00010111
00BF 0000000011001010 0001100010000000 0000000000000010 10110011
00C0 0000000000000000 0111000000101000 0000000000000010 00010111
00C1 0000000011110011 0001100010000000 0000000000000010 10110011
00C2 0000000000000000 0111000000101000 0000000000000010 00010111
00C3 0000000100001111 0001100010000000 0000000000000010 10110011
00C4 0000000000000000 0111000000101000 0000000000000010 00010111
00C5 0000000100111111 0001100010000000 0000000000000010 10110011
00C6 0000000001010000 0001110010000000 0000000000000000 00011111
00C7 0000000101010111 0001100010000000 0000000000000000 01001111
00C8 0000000110001000 0000110010000000 0000000000000000 01001111
00C9 0000000001010000 0001110010000000 0000000000000000 00011111
00CA 0000000101010111 0001100010000000 0000000000000000 01001111
00CB 0000000110001000 0000110010000000 0000000000000000 01001111
00CC 0000000000000000 0111000010111001 1100000000000000 00011110
00CD 0000000000000000 0111000010000000 0000000000000000 00011111
00CE 0000000000000000 0111000010000000 0000000000000000 00011111
00CF 0000000000000000 1111000000001110 0000000000000010 00010111
00D0 0000000000000010 0111000001001100 0000000000000010 00010111
00D1 0000000011011110 0001100010000000 0000000000000010 10010011
00D2 0000001111100111 0111000111101110 0000000000000010 00010111
00D3 0000000000000000 0111000010000000 0000000000000000 00011111
00D4 0000000000000000 0111000010000000 0000000000000000 00011111
00D5 0000000000000000 0111000110010110 0000000000000010 00010111
00D6 0000000011010011 0001100010000000 0000000000000010 10010011
00D7 0000000000000000 0111000110010110 0001111000000010 00010111
00D8 0000000011001111 0001100010000000 0000000000000010 10010011
00D9 1010101010101010 0111000111101110 0000000000000010 00010111
00DA 1010101010101010 0111000111101110 0000001000000010 00010111
00DB 1010101010101010 0111000111101110 0000010000000010 00010111
00DC 1010101010101010 0111000111101110 0000011000000010 00010111
00DD 0000000011100111 0001100010000000 0000000000000000 00011111
00DE 0000000000000000 0111000010111001 1010000000000000 00011110
00DF 0000000000000000 0111000010000000 0000000000000000 00011111
00E0 0000000000000000 0111000010000000 0000000000000000 00011111
00E1 0000000000000000 0111000010000000 0000000000000000 00011111
00E2 0000000000000000 1111000110001110 0000000000000010 00010111
00E3 0000000000000000 1111000110001110 0000001000000010 00010111

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 18 of 24)

```

00E4 0000000000000000 1111000110001110 0000010000000010 00010111
00E5 0000000000000000 1111000110001110 0000011000000010 00010111
00E6 0000000011111111 0111000010111110 0000000000000000 00011110
00E7 0000000001110001 0111000010111110 0000000000000000 00011110
00E8 0000000000000000 0111000010000000 0000000000000000 00011111
00E9 0000000000000000 1111000000001110 0000000000000010 00010111
00EA 0010000000000000 0111000001001100 0000000000000010 00010111
00EB 0000000011101001 0001100010000000 0000000000000010 10110011
00EC 000000001110100 0111000010111110 0000000000000000 00011110
00ED 0000000011111111 0111000010111110 0000000000000000 00011110
00EE 0000000000000000 0111000010111000 0000000000000000 00011111
00EF 0000000000000000 0111000010111000 0010000000000000 00011111
00F0 0000000000000000 0111000010111000 0100000000000000 00011111
00F1 0000000000000000 0111000010111000 0110000000000000 00011111
00F2 000000000101000 0001110010000000 0000000000000000 00011111
00F3 000000001110001 0111000010111110 0000000000000000 00011110
00F4 0000000000000000 0111000010000000 0000000000000000 00011111
00F5 0000000000000000 1111000000001110 0000000000000010 00010111
00F6 1000000000000000 011100001001100 0000000000000010 00010111
00F7 0000000011111001 0001100010000000 0000000000000010 10110011
00F8 0000000110111011 0000110010000000 0000000000000000 01001111
00F9 000000001110001 0111000010111110 0000000000000000 00011110
00FA 0000000000000000 0111000010000000 0000000000000000 00011111
00FB 0000000000000000 1111000000001110 0000000000000010 00010111
00FC 0010000000000000 011100001001100 0000000000000010 00010111
00FD 0000000011111011 0001100010000000 0000000000000010 10110011
00FE 000000001000100 0111000010111110 0000000000000000 00011110
00FF 0000000011111111 0111000010111110 0000000000000000 00011110
0100 0000000000000000 0111000010000000 0000000000000000 00011111
0101 0000000000000000 0111000010000000 0000000000000000 00011111
0102 0000000000000000 0111000010000000 0000000000000000 00011111
0103 0000000000000000 0111000010000000 0000000000000000 00011111
0104 0000000000000000 1111000110001110 0000000000000010 00010111
0105 0000000000000000 1111000110001110 0000001000000010 00010111
0106 0000000000000000 1111000110001110 0000010000000010 00010111
0107 0000000000000000 1111000110001110 0000011000000010 00010111
0108 000000001110100 0111000010111110 0000000000000000 00011110
0109 0000000011111111 0111000010111110 0000000000000000 00011110
010A 0000000000000000 0111000010111000 0000000000000000 00011111
010B 0000000000000000 0111000010111000 0010000000000000 00011111
010C 0000000000000000 0111000010111000 0100000000000000 00011111
010D 0000000000000000 0111000010111000 0110000000000000 00011111
010E 000000000101000 0001110010000000 0000000000000000 00011111
010F 0000000001110001 0111000010111110 0000000000000000 00011110
0110 0000000000000000 0111000010000000 0000000000000000 00011111
0111 0000000000000000 1111000000001110 0000000000000010 00010111
0112 1000000000000000 011100001001100 0000000000000010 00010111
0113 0000000100010101 0001100010000000 0000000000000010 10110011
0114 0000000110111011 0000110010000000 0000000000000000 01001111
0115 000000001110001 0111000010111110 0000000000000000 00011110
0116 0000000000000000 0111000010000000 0000000000000000 00011111
0117 0000000000000000 1111000000001110 0000000000000010 00010111
0118 0010000000000000 011100001001100 0000000000000010 00010111
0119 0000000100010111 0001100010000000 0000000000000010 10110011
011A 0000000001010100 0111000010111110 0000000000000000 00011110
011B 0000000011111111 0111000010111110 0000000000000000 00011110
011C 0000000000000000 0111000010000000 0000000000000000 00011111

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 19 of 24)

```

011D 0000000000000000 0111000010000000 0000000000000000 00011111
011E 0000000000000000 0111000010000000 0000000000000000 00011111
011F 0000000000000000 0111000010000000 0000000000000000 00011111
0120 0000000000000000 1111000110001110 0000000000000010 00010111
0121 0000000000000000 1111000110001110 0000001000000010 00010111
0122 0000000000000000 1111000110001110 0000010000000010 00010111
0123 0000000000000000 1111000110001110 0000011000000010 00010111
0124 0000000001110100 0111000010111110 0000000000000000 00011110
0125 0000000011111111 0111000010111110 0000000000000000 00011110
0126 0000000000000000 0111000010111000 0000000000000000 00011111
0127 0000000000000000 0111000010111000 0010000000000000 00011111
0128 0000000000000000 0111000010111000 0100000000000000 00011111
0129 0000000000000000 0111000010111000 0110000000000000 00011111
012A 0000000001010101 0111000010111110 0000000000000000 00011110
012B 0000000011111111 0111000010111110 0000000000000000 00011110
012C 0000000000000000 0111000010000000 0000000000000000 00011111
012D 0000000000000000 0111000010000000 0000000000000000 00011111
012E 0000000000000000 0111000010000000 0000000000000000 00011111
012F 0000000000000000 0111000010000000 0000000000000000 00011111
0130 0000000000000000 1111000110001110 0000000000000010 00010111
0131 0000000000000000 1111000110001110 0000001000000010 00010111
0132 0000000000000000 1111000110001110 0000010000000010 00010111
0133 0000000000000000 1111000110001110 0000011000000010 00010111
0134 0000000000000000 1111000110001110 0000100000000010 00010111
0135 0000000000000000 1111000110001110 0000101000000010 00010111
0136 0000000001110101 0111000010111110 0000000000000000 00011110
0137 0000000011111111 0111000010111110 0000000000000000 00011110
0138 0000000000000000 0111000010111000 0000000000000000 00011111
0139 0000000000000000 0111000010111000 0010000000000000 00011111
013A 0000000000000000 0111000010111000 0100000000000000 00011111
013B 0000000000000000 0111000010111000 0110000000000000 00011111
013C 0000000000000000 0111000010111000 1000000000000000 00011111
013D 0000000000000000 0111000010111000 1010000000000000 00011111
013E 000000000101000 0001110010000000 0000000000000000 00011111
013F 0000000001110001 0111000010111110 0000000000000000 00011110
0140 0000000000000000 0111000010000000 0000000000000000 00011111
0141 0000000000000000 1111000110001110 0000000000000010 00010111
0142 0100000000000000 011100001001100 0000000000000010 00010111
0143 0000000000101000 0001100010000000 0000000000000010 10110011
0144 0000000001110010 0111000010111110 0000000000000000 00011110
0145 0000000011111111 0111000010111110 0000000000000000 00011110
0146 0000000000000000 0111000010000000 0000000000000000 00011111
0147 0000000000000000 1111000110001110 0000000000000010 00010111
0148 0000000000000000 1111000110001110 0000001000000010 00010111
0149 0000000000000000 1111000110001110 0000010000000010 00010111
014A 0000000000000000 1111000110001110 0000011000000010 00010111
014B 0000000001110001 0111000010111110 0000000000000000 00011110
014C 0000000000000000 0111000010000000 0000000000000000 00011111
014D 0000000000000000 1111000000001110 0000000000000010 00010111
014E 0010000000000000 011100001001100 0000000000000010 00010111
014F 0000000101001101 0001100010000000 0000000000000010 10110011
0150 0000000001110100 0111000010111110 0000000000000000 00011110
0151 0000000011111111 0111000010111110 0000000000000000 00011110
0152 0000000000000000 0111000010111000 0000000000000000 00011111
0153 0000000000000000 0111000010111000 0010000000000000 00011111
0154 0000000000000000 0111000010111000 0100000000000000 00011111
0155 0000000000000000 0111000010111000 0110000000000000 00011111

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 20 of 24)

```

0156 000000000101000 0001110010000000 0000000000000000 00011111
0157 0000000001110001 0111000010111110 0000000000000000 00011110
0158 0000000000000000 0111000010000000 0000000000000000 00011111
0159 0000000000000000 1111000110001110 0001011000000010 00010111
015A 0000000000000000 0111001010110110 0001011000000010 00010111
015B 0000000000000000 0111001010110110 0001011000000010 00010111
015C 0000000000000000 0111001010110110 0001011000000010 00010111
015D 0000000000000011 0111000111101110 0001000000000010 00010111
015E 0000000000000000 0111000111000011 1001011000000010 00010111
015F 0000000101111111 0001100010000000 0000000000000010 10110011
0160 0000000000000000 0111000110010110 0001011000000010 00010111
0161 0000000101110110 0001100010000000 0000000000000010 10110011
0162 0000000000000000 0111000110010110 0001011000000010 00010111
0163 0000000101110110 0001100010000000 0000000000000010 10110011
0164 0000000000110001 0111000111101110 0001001000000010 00010111
0165 0000000000110111 0111000111101110 0001010000000010 00010111
0166 0000000000110010 0111000111101110 0001011000000010 00010111
0167 0000000000110011 0111000111101110 0001100000000010 00010111
0168 0000000000110100 0111000111101110 0001101000000010 00010111
0169 0000000000110000 0111000111101110 0001110000000010 00010111
016A 0000000100001000 0111000111101110 0001111000000010 00010111
016B 0000000000110101 0111000111101110 0001000000000010 00010111
016C 0000000000000000 0101010010000000 0000000000000000 01010111
016D 0000000000100001 0111000111101110 0001001000000010 00010111
016E 0000000000100111 0111000111101110 0001010000000010 00010111
016F 0000000000100010 0111000111101110 0001011000000010 00010111
0170 0000000000100011 0111000111101110 0001100000000010 00010111
0171 0000000000100100 0111000111101110 0001101000000010 00010111
0172 0000000000100000 0111000111101110 0001110000000010 00010111
0173 0000000000100010 0111000111101110 0001111000000010 00010111
0174 0000000000100101 0111000111101110 0001000000000010 00010111
0175 0000000000000000 0101010010000000 0000000000000000 01010111
0176 0000000000010001 0111000111101110 0001001000000010 00010111
0177 0000000000010111 0111000111101110 0001010000000010 00010111
0178 0000000000010010 0111000111101110 0001011000000010 00010111
0179 0000000000010011 0111000111101110 0001100000000010 00010111
017A 0000000000010100 0111000111101110 0001101000000010 00010111
017B 0000000000010000 0111000111101110 0001110000000010 00010111
017C 00000000001000110 0111000111101110 0001111000000010 00010111
017D 0000000000010101 0111000111101110 0001000000000010 00010111
017E 0000000000000000 0101010010000000 0000000000000000 01010111
017F 0000000000000001 0111000111101110 0001001000000010 00010111
0180 0000000000000111 0111000111101110 0001010000000010 00010111
0181 0000000000000010 0111000111101110 0001011000000010 00010111
0182 0000000000000011 0111000111101110 0001100000000010 00010111
0183 0000000000000100 0111000111101110 0001101000000010 00010111
0184 0000000000000000 0111000111101110 0001110000000010 00010111
0185 00000000001000110 0111000111101110 0001111000000010 00010111
0186 0000000000000101 0111000111101110 0001000000000010 00010111
0187 0000000000000000 0101010010000000 0000000000000000 01010111
0188 0000000001110001 0111000010111110 0000000000000000 00011110
0189 0000000000000000 0111000010000000 0000000000000000 00011111
018A 0000000000000000 1111000000011110 0000000000000010 00010111
018B 0100000000000000 0111000001001100 0000000000000010 00010111
018C 0000000110001010 0001100010000000 0000000000000010 10110011
018D 0000000000000000 0111000010111001 0010000000000000 00011110
018E 0000000011111111 0111000010111110 0000000000000000 00011110

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 21 of 24)

```

018F 000000001110001 011100001011110 000000000000000 00011110
0190 000000000000000 011100001000000 000000000000000 00011111
0191 000000000000000 111100000001110 000000000000010 00010111
0192 000100000000000 011100001001100 000000000000010 00010111
0193 0000000110010110 000110001000000 000000000000010 10110011
0194 000000000000000 0111000010111001 000000000000000 00011110
0195 000000001111111 011100001011110 000000000000000 00011110
0196 000000001110010 011100001011110 000000000000000 00011110
0197 000000001111111 011100001011110 000000000000000 00011110
0198 000000000000000 011100001000000 000000000000000 00011111
0199 000000000000000 111100011000110 000000000000010 00010111
019A 000000000000000 111100011000110 0000001000000010 00010111
019B 000000000000000 111100011000110 0000010000000010 00010111
019C 000000000000000 111100011000110 0000011000000010 00010111
019D 000000000000000 0111000010111001 011000000000000 00011110
019E 000000000000000 0111000010111000 000000000000000 00011111
019F 000000000000000 0111000010111000 001000000000000 00011111
01A0 000000000000000 0111000010111000 010000000000000 00011111
01A1 000000000000000 0111000010111000 011000000000000 00011111
01A2 000000001111111 011100001011110 000000000000000 00011110
01A3 0000000000001110 0111000111001010 0010001000000010 00010111
01A4 0000000110101000 000110001000000 0000000000000010 10010011
01A5 000000000000000 0111000010111001 010000000000000 00011110
01A6 000000001111111 011100001011110 000000000000000 00011110
01A7 000000000000000 010101001000000 000000000000000 01010111
01A8 0000000001110011 011100001011110 000000000000000 00011110
01A9 000000001111111 011100001011110 000000000000000 00011110
01AA 000000000000000 011100001000000 000000000000000 00011111
01AB 000000000000000 1111000110001110 000000000000010 00010111
01AC 000000000000000 111100011000110 0000001000000010 00010111
01AD 000000000000000 111100011000110 0000010000000010 00010111
01AE 000000000000000 111100011000110 0000011000000010 00010111
01AF 000000000000000 111100011000110 0000100000000010 00010111
01B0 000000000000000 111100011000110 0000101000000010 00010111
01B1 000000000000000 0111000010111001 100000000000000 00011110
01B2 000000000000000 0111000010111000 000000000000000 00011111
01B3 000000000000000 0111000010111000 001000000000000 00011111
01B4 000000000000000 0111000010111000 010000000000000 00011111
01B5 000000000000000 0111000010111000 011000000000000 00011111
01B6 000000000000000 0111000010111000 100000000000000 00011111
01B7 000000000000000 0111000010111000 101000000000000 00011111
01B8 000000000000000 0111000010111001 010000000000000 00011110
01B9 000000001111111 011100001011110 000000000000000 00011110
01BA 000000000000000 010101001000000 000000000000000 01010111
01BB 0000000001110010 011100001011110 000000000000000 00011110
01BC 000000001111111 011100001011110 000000000000000 00011110
01BD 000000000000000 011100001000000 000000000000000 00011111
01BE 000000000000000 1111000110001110 0000000000000010 00010111
01BF 000000000000000 111100011000110 0000001000000010 00010111
01C0 000000000000000 111100011000110 0000010000000010 00010111
01C1 000000000000000 111100011000110 0000011000000010 00010111
01C2 000000000000000 0111000111101000 0101100000000010 00010111
01C3 000000001000010 011100001011110 000000000000000 00011110
01C4 000000001111111 011100001011110 000000000000000 00011110
01C5 000000000000000 0111000010111000 000000000000000 00011111
01C6 000000000000000 0111000010111000 001000000000000 00011111
01C7 000000000000000 0111000010111000 010000000000000 00011111

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 22 of 24)

```

01C8 0000000000000000 0111000010111000 0110000000000000 00011111
01C9 0000000001110010 0111000010111110 0000000000000000 00011110
01CA 0000000011111111 0111000010111110 0000000000000000 00011110
01CB 0000000000000000 0111000010000000 0000000000000000 00011111
01CC 0000000000000000 1111000110001110 0000000000000010 00010111
01CD 0000000000000000 1111000110001110 0000001000000010 00010111
01CE 0000000000000000 1111000110001110 0000010000000010 00010111
01CF 0000000000000000 1111000110001110 0000011000000010 00010111
01D0 0000000000000000 0111000111101000 0101101000000010 00010111
01D1 0000000001010010 0111000010111110 0000000000000000 00011110
01D2 0000000001111111 0111000010111110 0000000000000000 00011110
01D3 0000000000000000 0111000010111000 0000000000000000 00011111
01D4 0000000000000000 0111000010111000 0010000000000000 00011111
01D5 0000000000000000 0111000010111000 0100000000000000 00011111
01D6 0000000000000000 0111000010111000 0110000000000000 00011111
01D7 0000000000000001 0111000010111110 0000000000000000 00011110
01D8 00000000000010001 0111000010111110 0000000000000000 00011110
01D9 00000000000100001 0111000010111110 0000000000000000 00011110
01DA 00000000000110001 0111000010111110 0000000000000000 00011110
01DB 0000000001111111 0111000010111110 0000000000000000 00011110
01DC 0000000000100000 0111000111101110 0000110000000010 00010111
01DD 00000000000011110 0111000111101110 0000111000000010 00010111
01DE 00000000000011110 0111000111101110 0001000000000010 00010111
01DF 0000000001100100 0111000111101110 0001111000000010 00010111
01E0 0000000000000000 0111000110010110 0001111000000010 00010111
01E1 0000000011110000 0001100010000000 0000000000000010 10010011
01E2 0000000000000000 0101010010000000 0000000000000000 01010111
01E3 0000000001000000 0111000111101110 0001111000000010 00010111
01E4 0000000001110110 0111000010111110 0000000000000000 00011110
01E5 0000000011111111 0111000010111110 0000000000000000 00011110
01E6 0000000000000000 0111000010111001 1110000000000000 00011111
01E7 0000000001110111 0111000010111110 0000000000000000 00011110
01E8 0000000001111111 0111000010111110 0000000000000000 00011110
01E9 0000000000000000 0000000010000000 0000000000000000 00011111

```

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 23 of 24)

SYMBOLS

A.FAT123	008E	AB	0001	AQ	0000	ARPG	00F9
ARPGDIAG	00F3	ATC	00A5	ATC2	00AE	ATD	FFE1
BEGIN	0028	BEGIN1	002A	BUSY	0001	CIONE	0001
CIZERO	0000	CJP	0003	CJPP	0008	CJS	0001
CJU	0006	COND	0000	CONT	000E	CRTN	000A
CTRLN	0000	CUMSK	1000	DA	0005	DELAY	0054
DIAG	008C	DIAGFAT	0188	DQ	0006	DUMP	0074
DZ	0007	EXNOR	0007	EXOR	0006	FADD	0000
FAND	0004	FATICW	0005	FAT1DIAG	0007	FAT1LD4	0002
FAT1LD6	0003	FAT1RD4	0004	FAT1RST	0001	FAT1STAT	0000
FAT2CW	0015	FAT2DIAG	0017	FAT2LD4	0012	FAT2LD6	0013
FAT2RD4	0014	FAT2RST	0011	FAT2STAT	0010	FAT3CW	0025
FAT3DIAG	0027	FAT3LD4	0022	FAT3LD6	0023	FAT3RD4	0024
FAT3RST	0021	FAT3STAT	0020	FAT4CW	0035	FAT4DIAG	0037
FAT4LD4	0032	FAT4LD6	0033	FAT4RD4	0034	FAT4RST	0031
FAT4STAT	0030	FAULTA	01E3	FOR	0003	FRGATN	0077
FULLLOAD	0031	GETDATA	00DE	GOMASK	0003	IFSTATUS	0071
INIT	0000	ITOUSR	0010	JMAP	0002	JRP	0007
JSRP	0005	JZ	0000	LDARPG	0042	LDCT	000C
LDMRPG	0052	LOAD4	0074	LOAD6	0075	LOADAFAT	0097
LOADDATA	00E7	LOADFAT	0196	LOADFAT4	0056	LOOP	0000
LOOPB	0091	LOOPC	00FB	LOOPD	0117	LOOPE	018A
LOOPF	0140	LOOPG	0023	LOOPH	00CF	LOOPI	00E9
LOOPJ	0003	LOOPK	01E0	M.FAT4	005F	MACROEN	0000
MICROEN	0000	MREPLY	01A8	MRPG	0115	MRPGDIAG	010F
MSRTOUSR	0002	MTD	FFDF	NEWPARM	018B	NOCTRL	0001
NOFRUIT	1000	NOMACRO	0001	NOMICRO	0001	NONEON	0003
NOOP	0000	NORMASK	0007	NOTRS	0005	NP	0001
PARMMASK	8000	PEMASK	2000	PUSH	0004	QREG	0000
R0	0000	R1	0001	R10	000A	R11	000B
R12	000C	R13	0000	R14	000E	R15	000F
R2	0002	R3	0003	R4	0004	R5	0005
R6	0006	R7	0007	R8	0008	R9	0009
RAMA	0002	RAMD	0005	RAMF	0003	RAMQD	0004
RAMQU	0006	RAMU	0007	RDARPG	0044	RDMRPGA	0054
RDMRPG8	0055	READ4	0072	READ6	0073	RFCT	0008
RFRQDIAG	00C7	RPCT	0009	RPLYDIAG	00CA	RPLYMASK	4000
RPYDATA	0002	RSTARPG	0041	RSTMTPG	0051	SELA	0000
SELB	0001	SELFAT	0157	SELFAT1	017F	SELFAT2	0176
SELFAT3	0160	SELFAT4	0164	SHFTRA	0000	SHIFT	0002
STEP1	0042	STEP2	0047	STEP2.S	0049	STEP3	004A
STEP4	004F	STEPS	0052	STEP6	0057	SUBR	0001
SUBS	0002	TEST	0001	TIMEOUT	0040	TIMER1	0F9E
TPARM	005A	TWB	000F	TYPE	000E	UC	001B
UN	001F	UNC	001A	UNCOND	0001	UNN	001E
UNQUR	0016	UNZ	0014	UQUR	0017	UPRCDIAG	013F
UPSTATUS	0076	USRTOMSR	0002	UZ	0015	ZA	0004
ZB	0003	ZQ	0002				

TOTAL PHASE 2 ERRORS = 0

FIGURE B.3.2-2. FRG MICROCODE PROGRAM LISTING (Sheet 24 of 24)

If the MRPG Validation is selected, the FRC will send replies generated by the MRPG to the CPU, thus providing a means of verifying the data generated by the MRPG against the software simulated data. (See page 10, titled "MODE S RPG VALIDATION.")

If the ARPG Validation is selected, the FRC will send replies generated by the ARPG to the CPU, thus providing a means of verifying the data generated by the ARPG against the software simulated data. (See page No. 9, titled "ATCRBS RPG VALIDATION.")

If the FAT Validation is selected, the FRC proceeds to sequence known ATCRBS replies provided by the CPU to a selected FAT. (See page No. 8, titled "REPLY GENERATOR LOOPBACK ROUTINE.") As the FAT transmits the reply, the reply data is reassembled and returned to the FRC, which in turn returns the data to the CPU. In this mode, the data paths between the FRC and the FATs are verified. Also, many of the hardware functions of the FATs are verified. (Functions unique to the Mode S replies are not checked by this routine, but are checked by the FAT RF Validation.) FAT selection is under CPU control. (See pages 11 and 12, titled "SELECT FAT.") If the ATCRBS FAT fails to return the reply block after 70 ms (264 ms for the Mode S FAT), the FRC will substitute an all "A" data pattern back to the CPU. In this way, it can be determined that the FAT did not respond.

If the FAT RF Validation mode is selected, the FRC proceeds to sequence replies to a selected FAT for transmission. (See page 8, titled "FRGLOOP DIAGNOSTIC.") This test mode can operate in conjunction with the STU Receiver, to collect the replies transmitted (ATCRBS or Mode S) for verification. In this mode, the FRG analog circuitry can be verified. Also, the FAT hardware functions, unique to the Mode S replies, can be verified.

### B.3.3 RRG Microcode Program.

This microcode program, resident in the Radar Report Controller (RRC) ROM, provides five modes of operation for the RRC: (1) normal operation, (2) RRG Interface Loopback Validation, (3) RRC Validation, (4) Advanced Data Communication Control Protocol (ADCCP) Data Converter (ADC) Validation, and (5) ADC Diagnostic. These routines are illustrated in flow charts presented in figure B.3.3-1. There are four ways to initialize the RRC microcode program: (1) system clear (CLR0) generated when the ARIES system is first powered up, (2) CPU controlled reset command to the RRC, (3) depressing the manual reset button on the RRC digital board, and (4) via the program itself whenever an error or fault is detected. Upon initialization, the RRC resets the ADC.

The CPU selects the mode of operation by setting the appropriate value in the operation mode field of the RRG command. (Refer to the ARIES Hardware Maintenance Manual, Volume II, Appendix D for additional information on the RRG command format.) Table B.3.3-1 lists the select mode options.



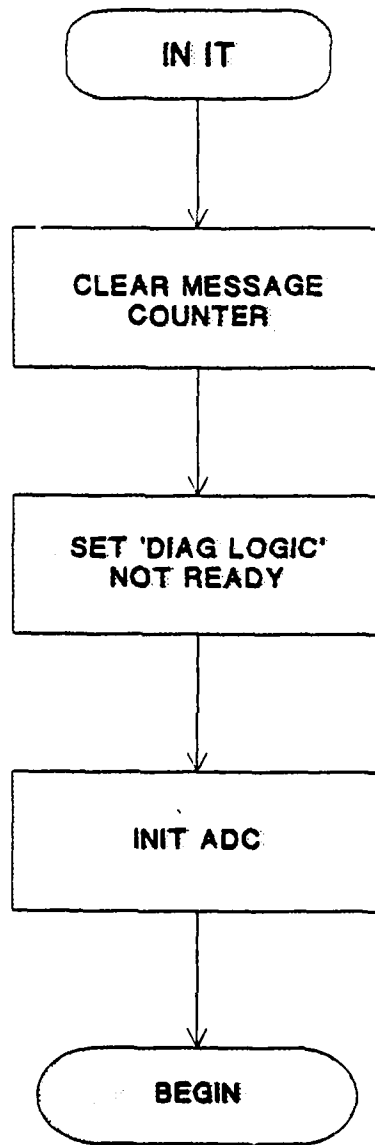


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 1 of 14)

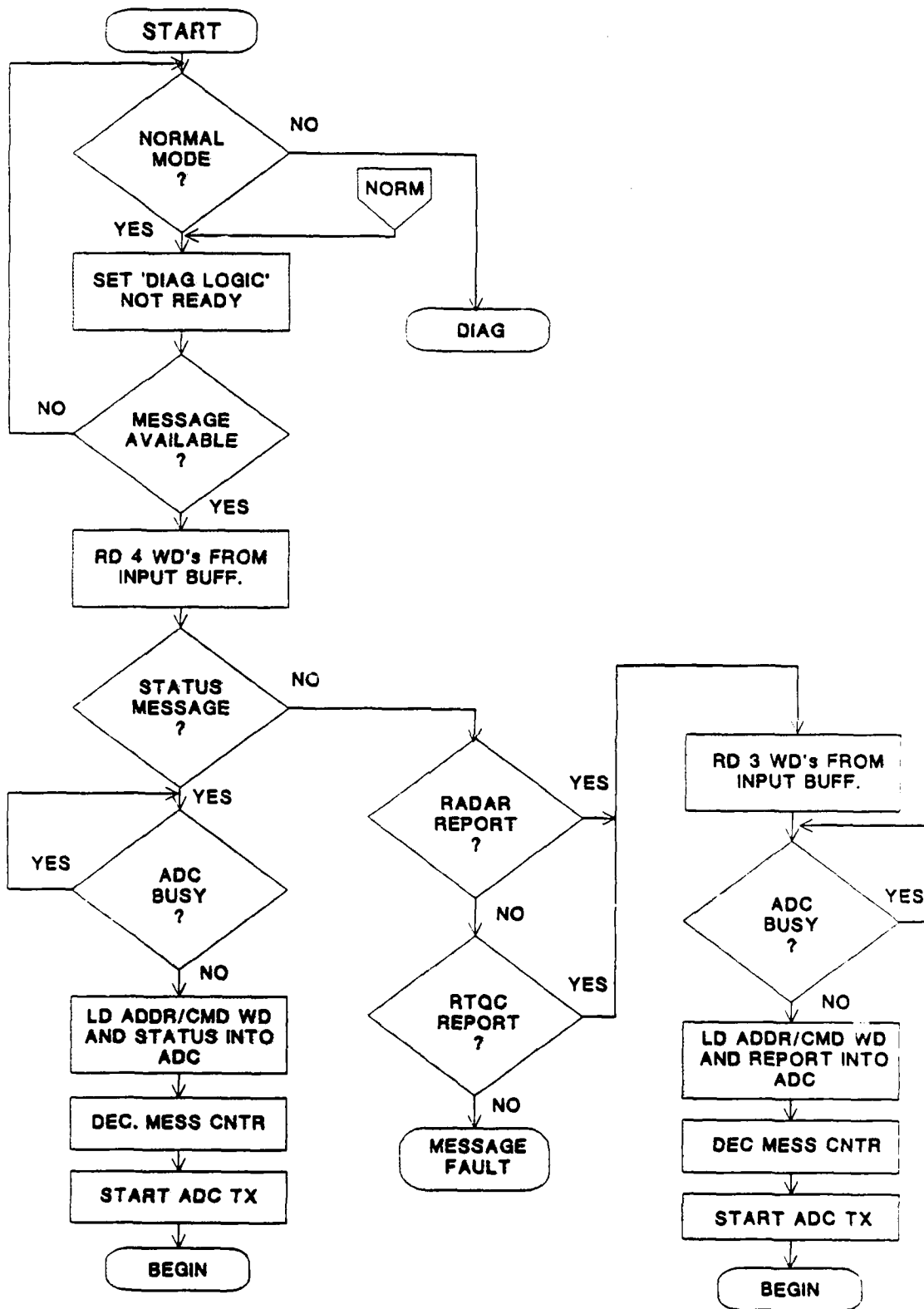


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 2 of 14)

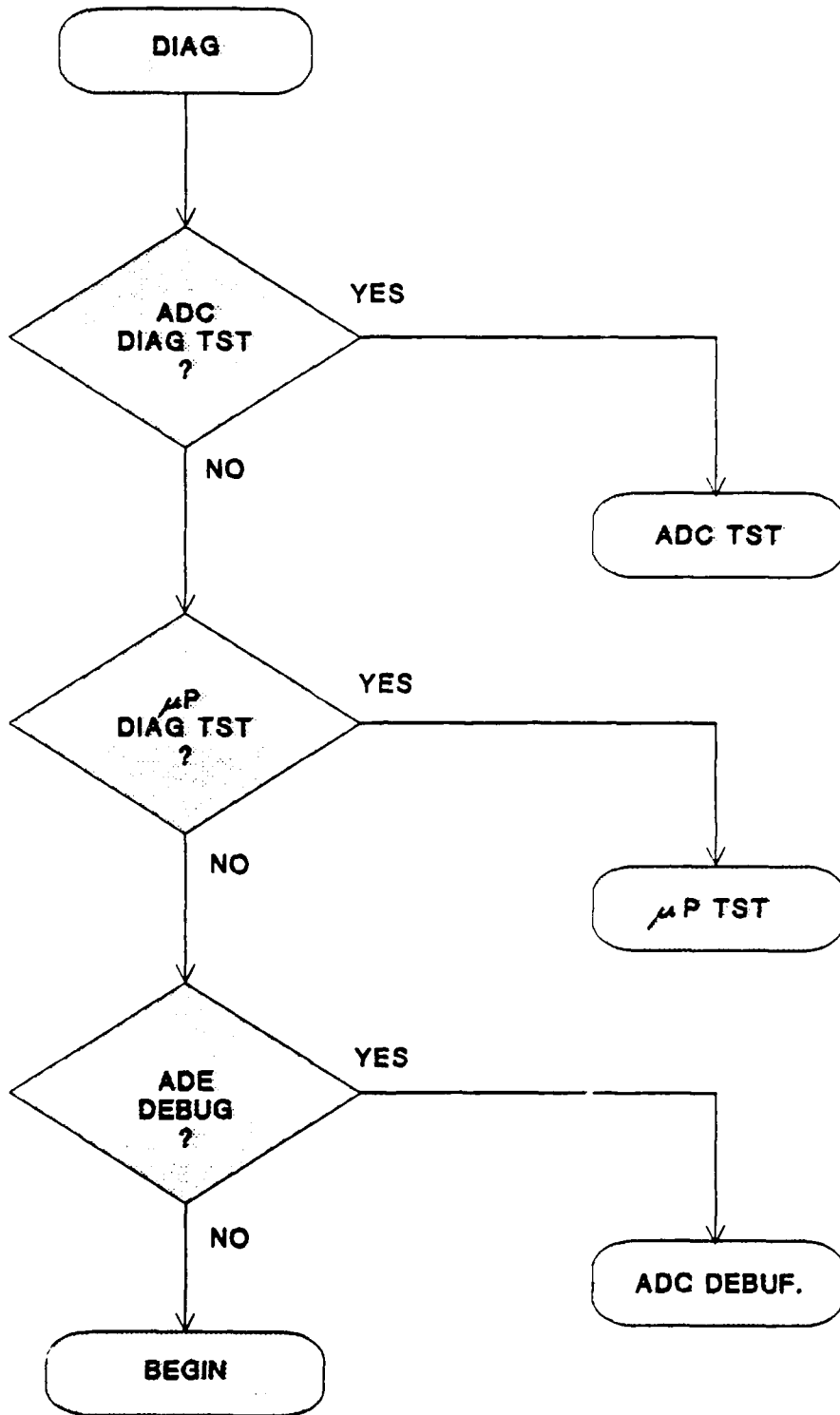


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 3 of 14)

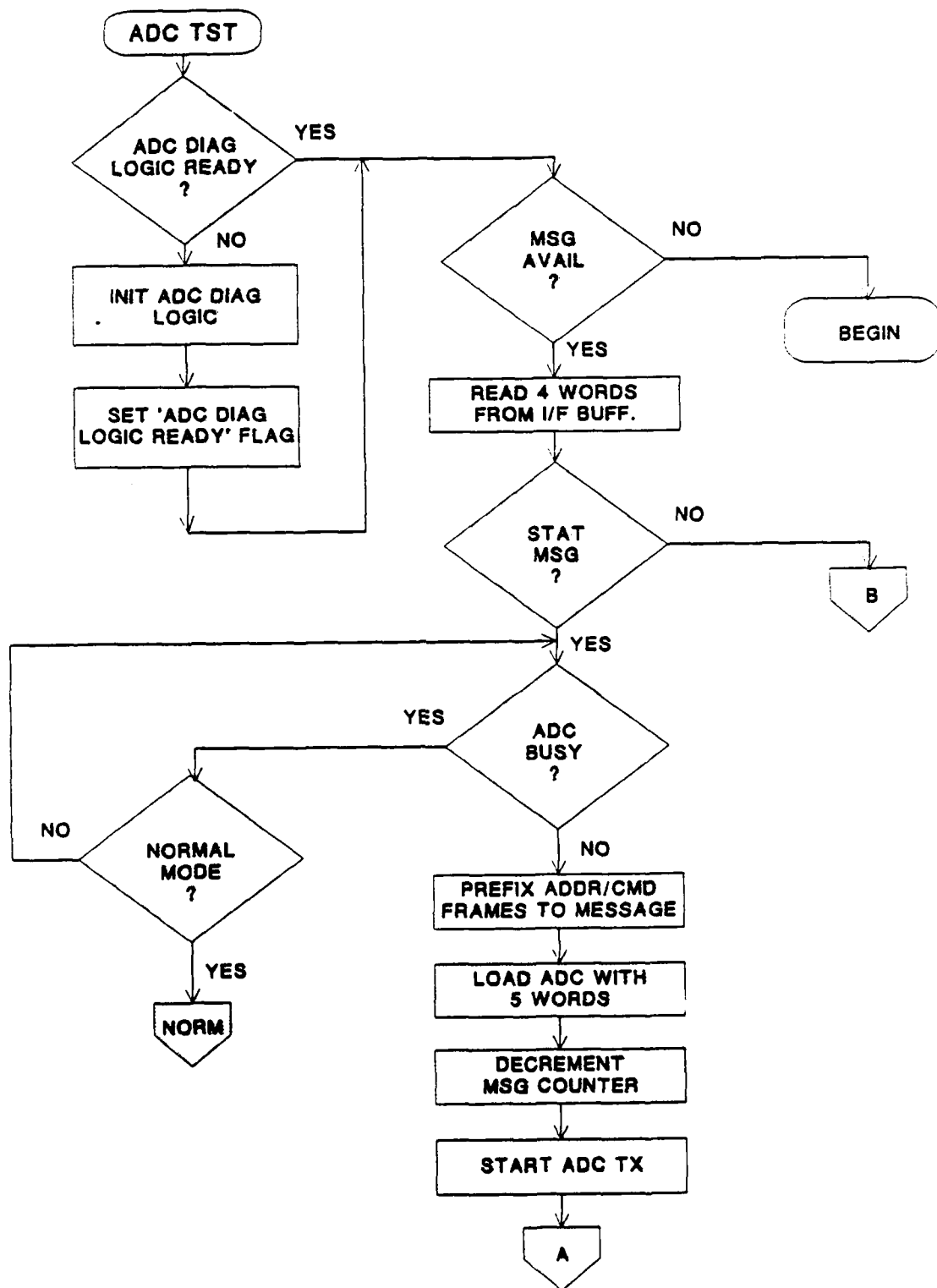


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 4 of 14)

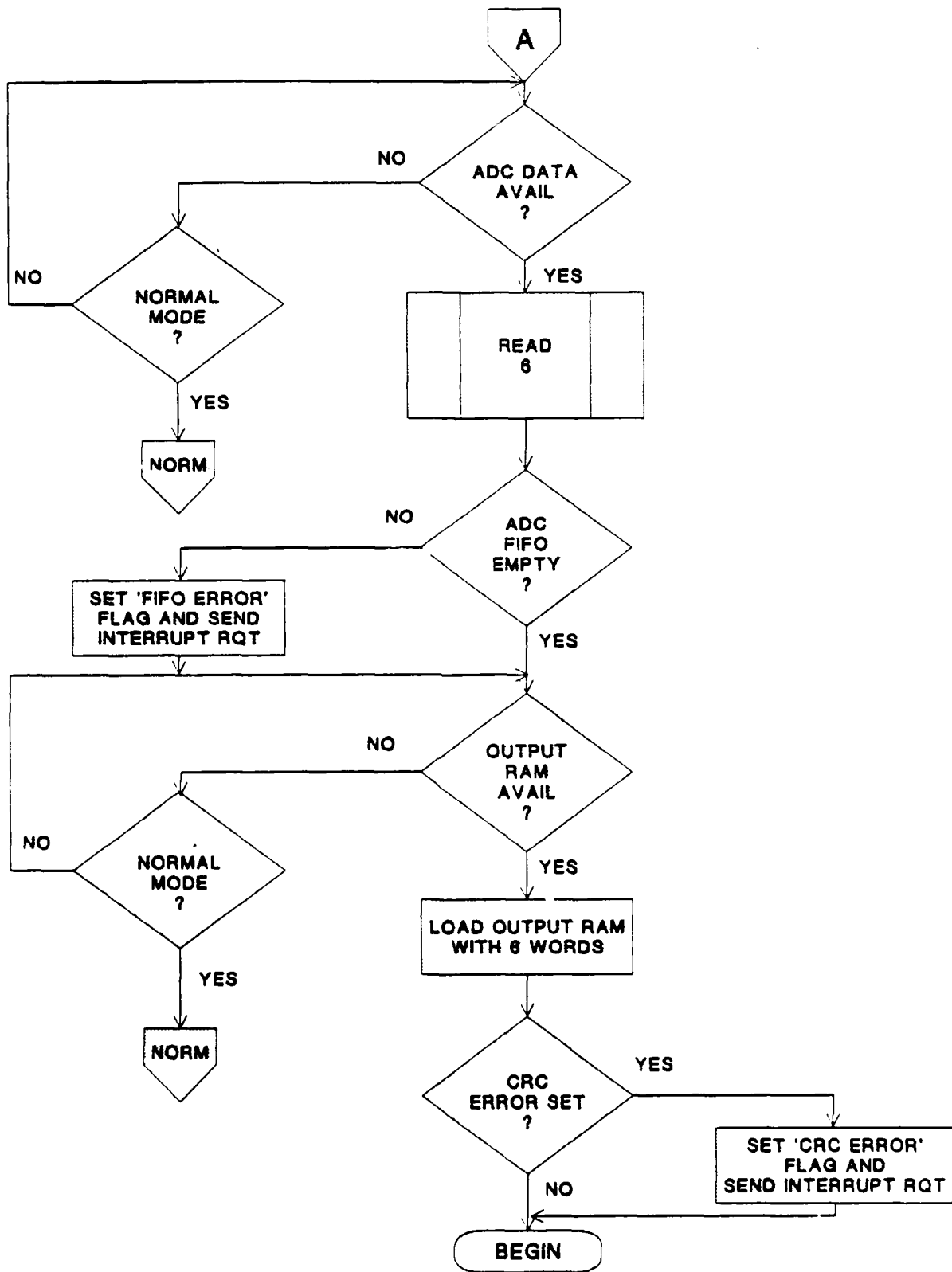


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 5 of 14)

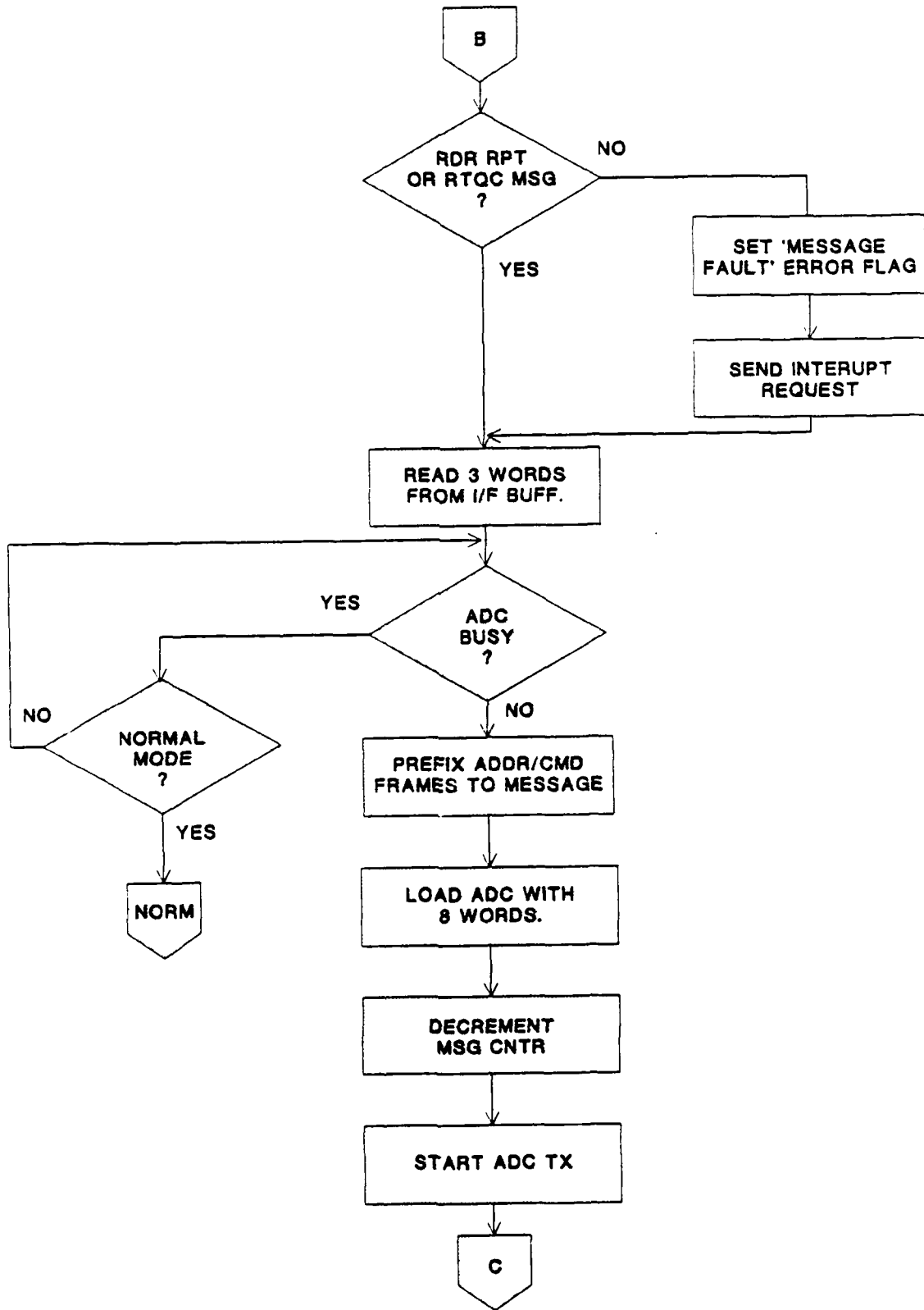


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 6 of 14)

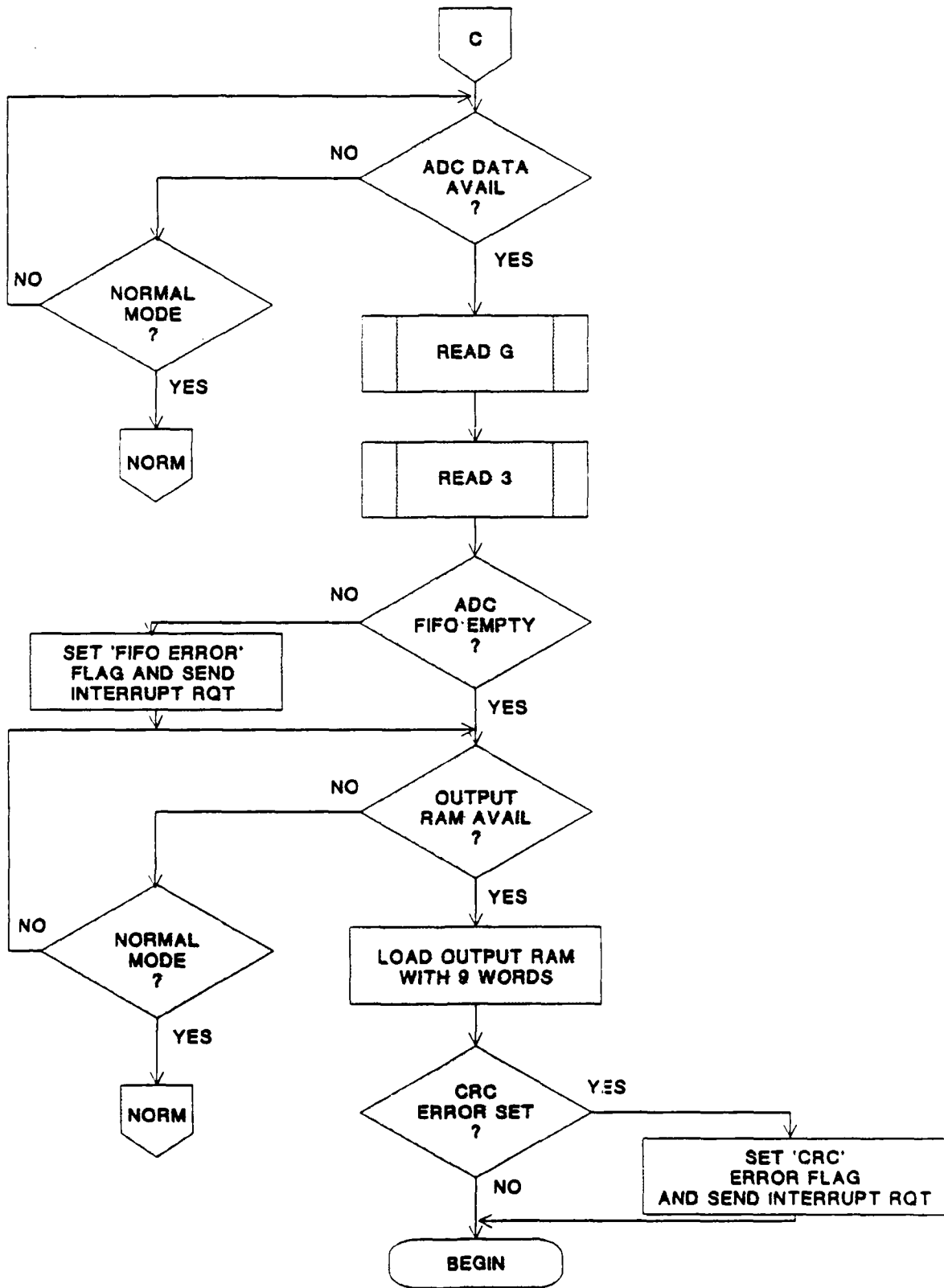


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 7 of 14)

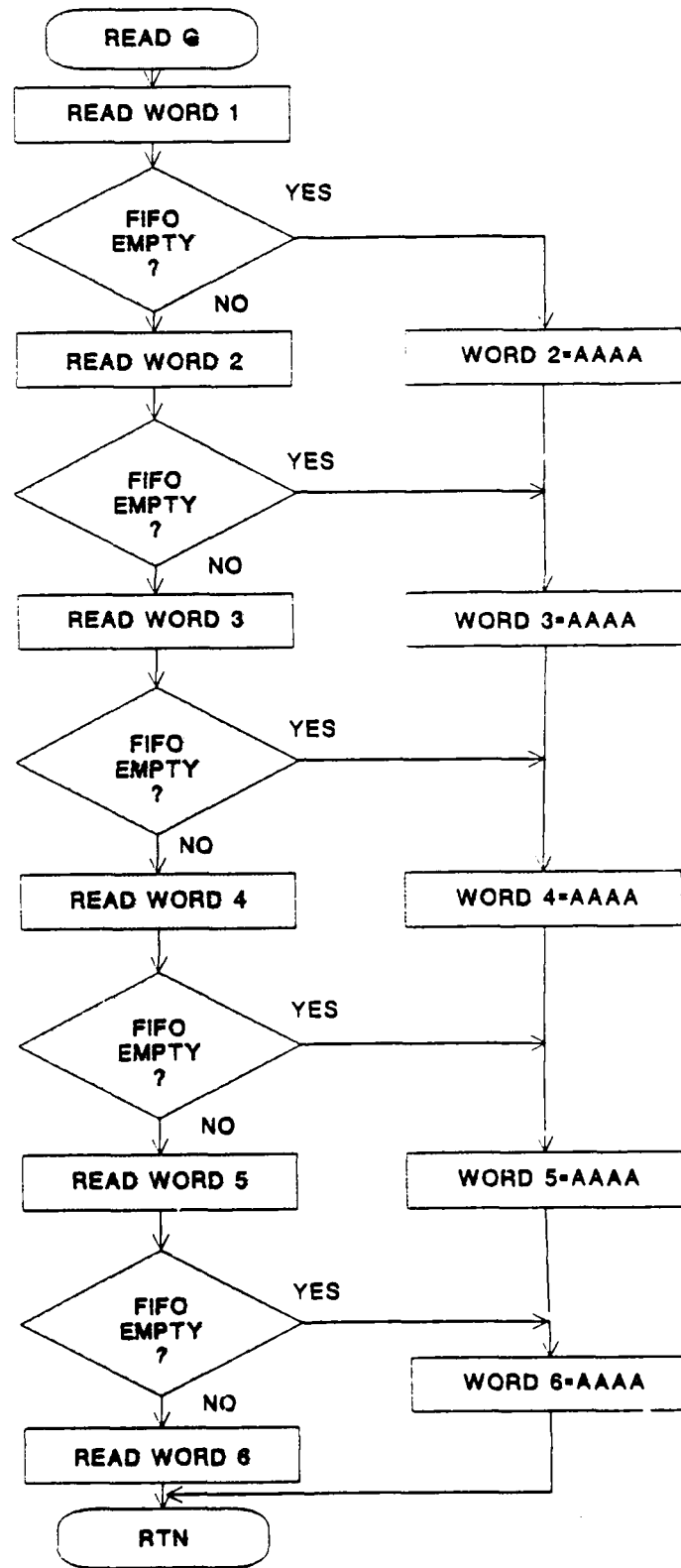


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 8 of 14)



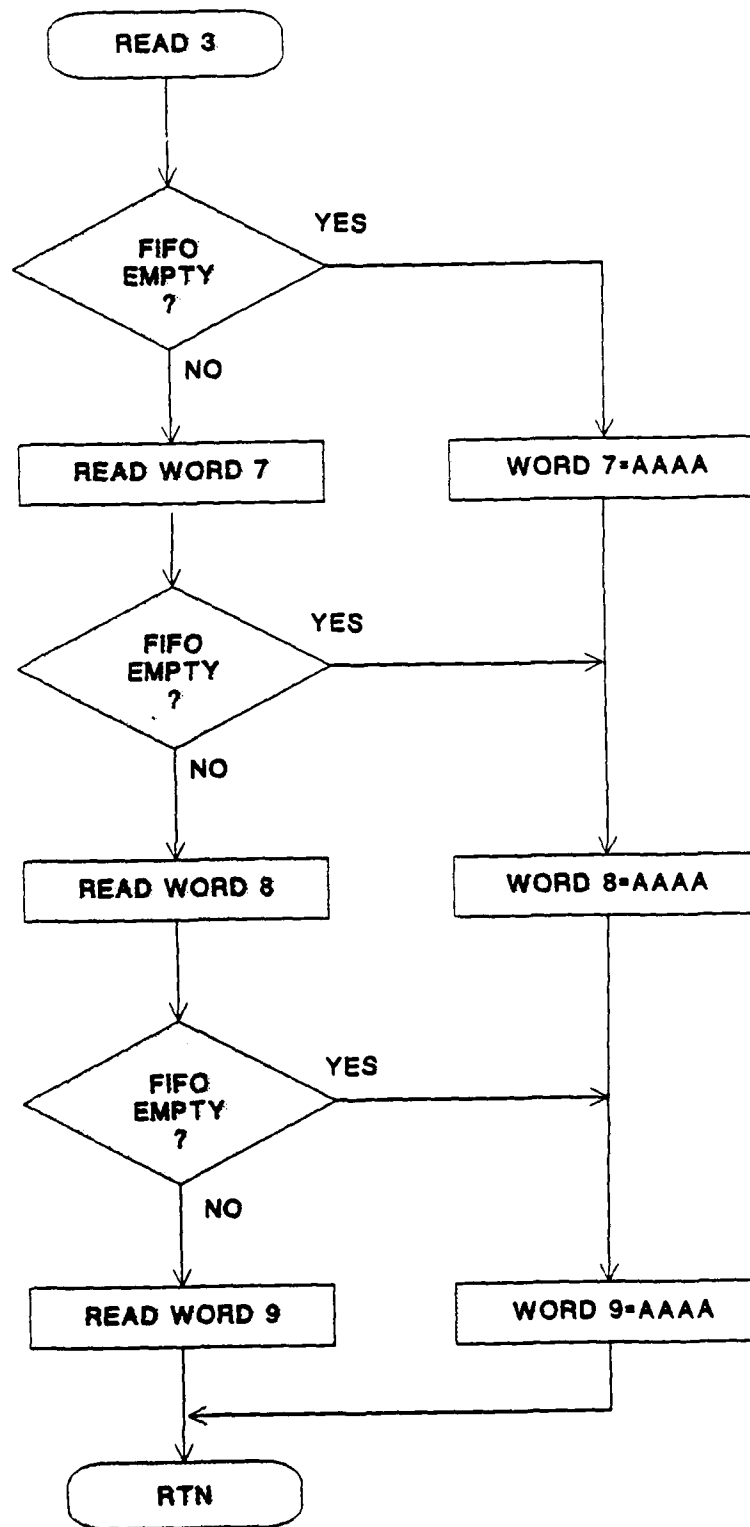


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 9 of 14)

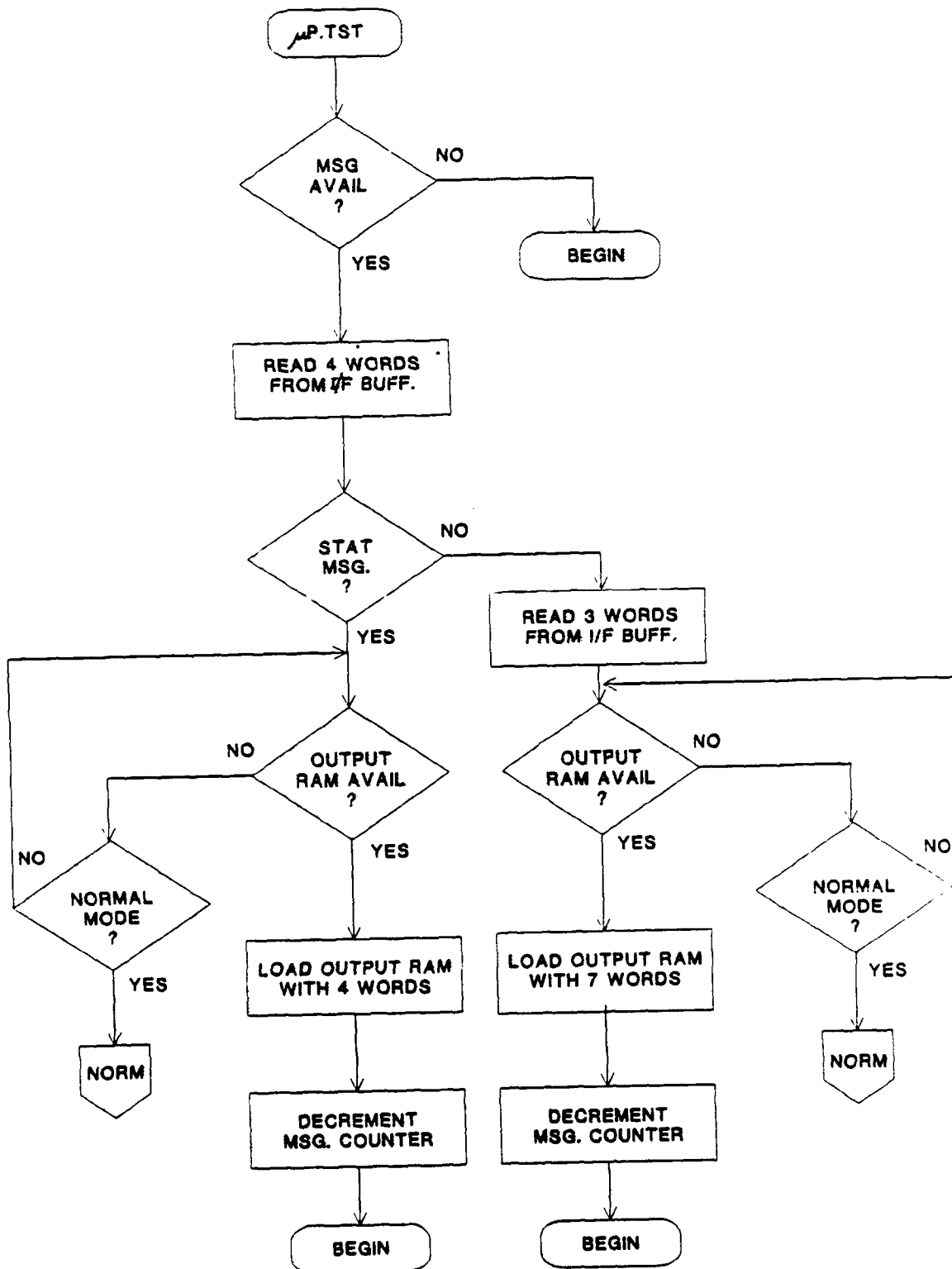


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 10 of 14)

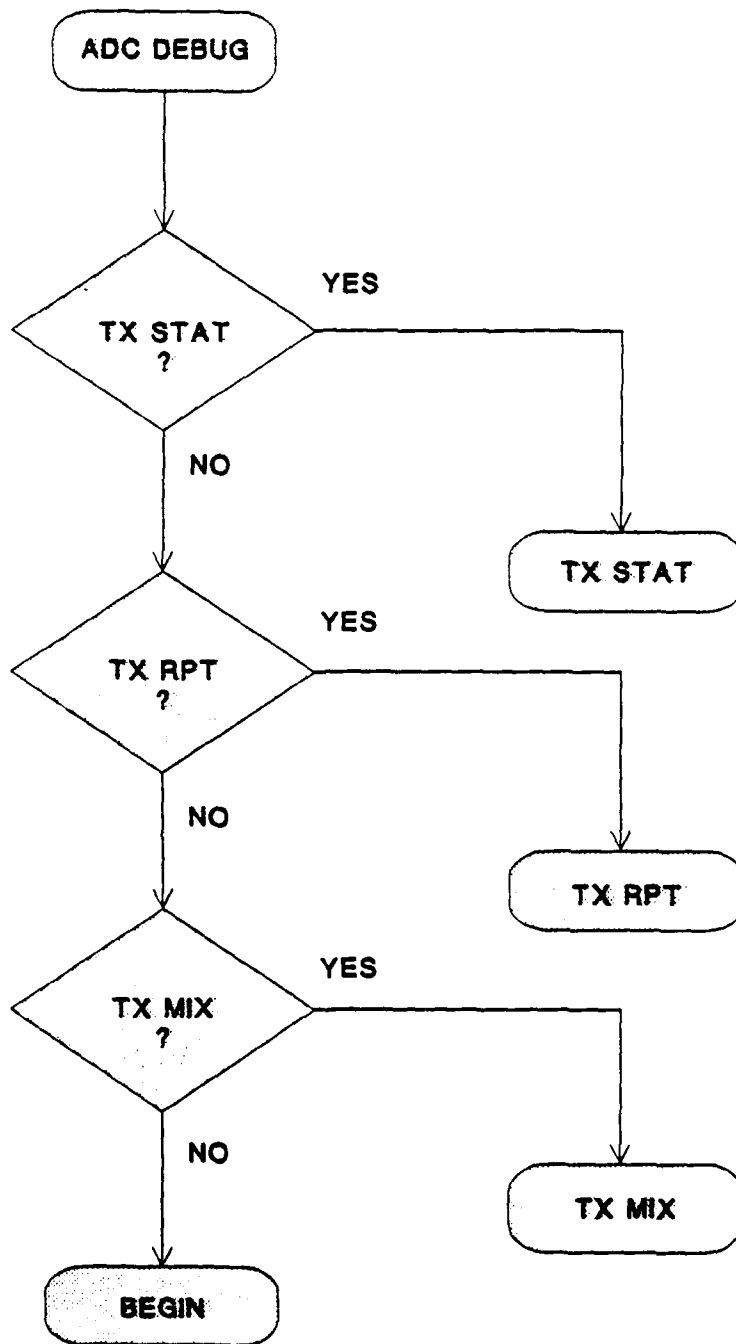


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 11 of 14)

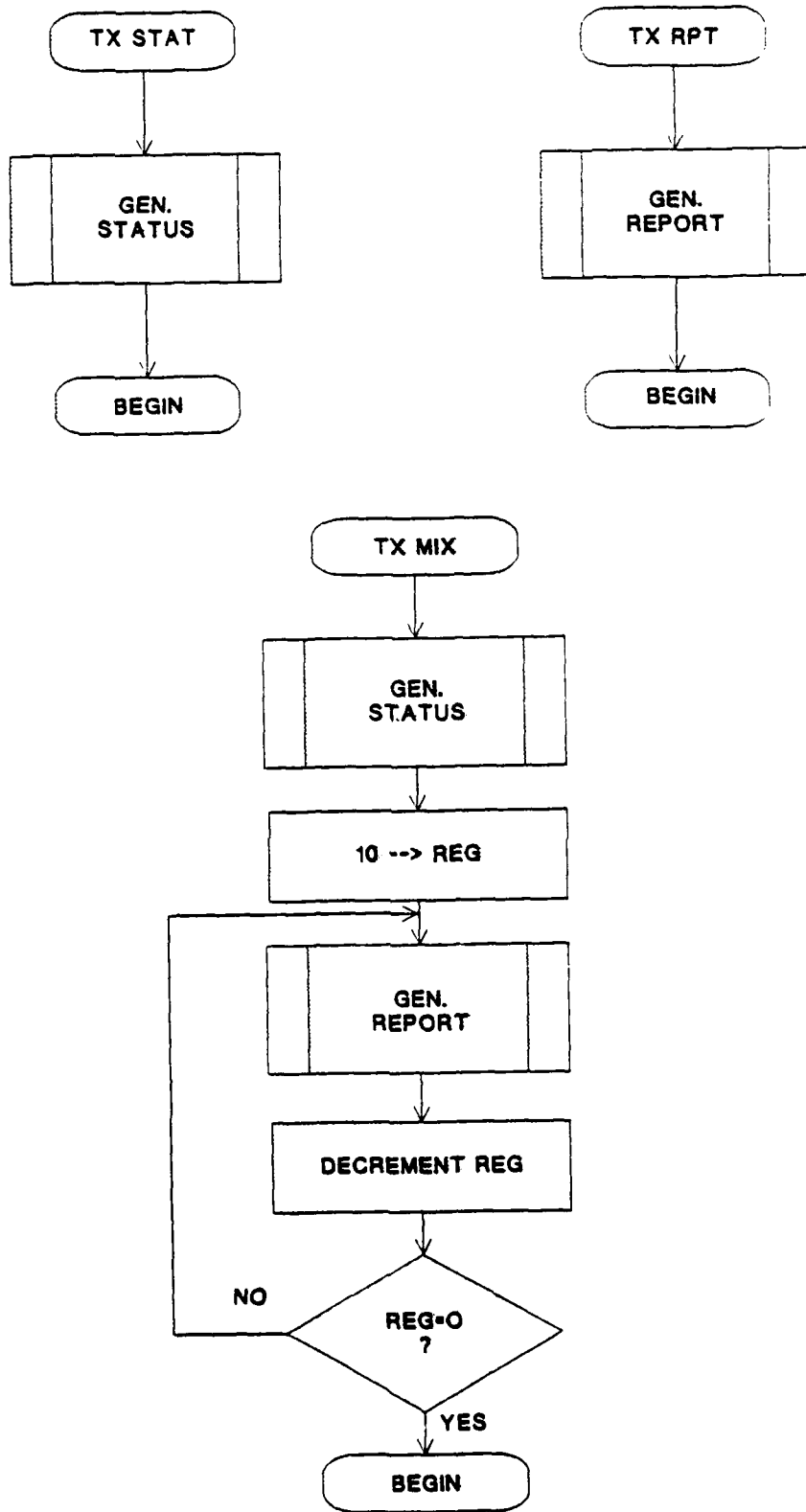


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 12 of 14)

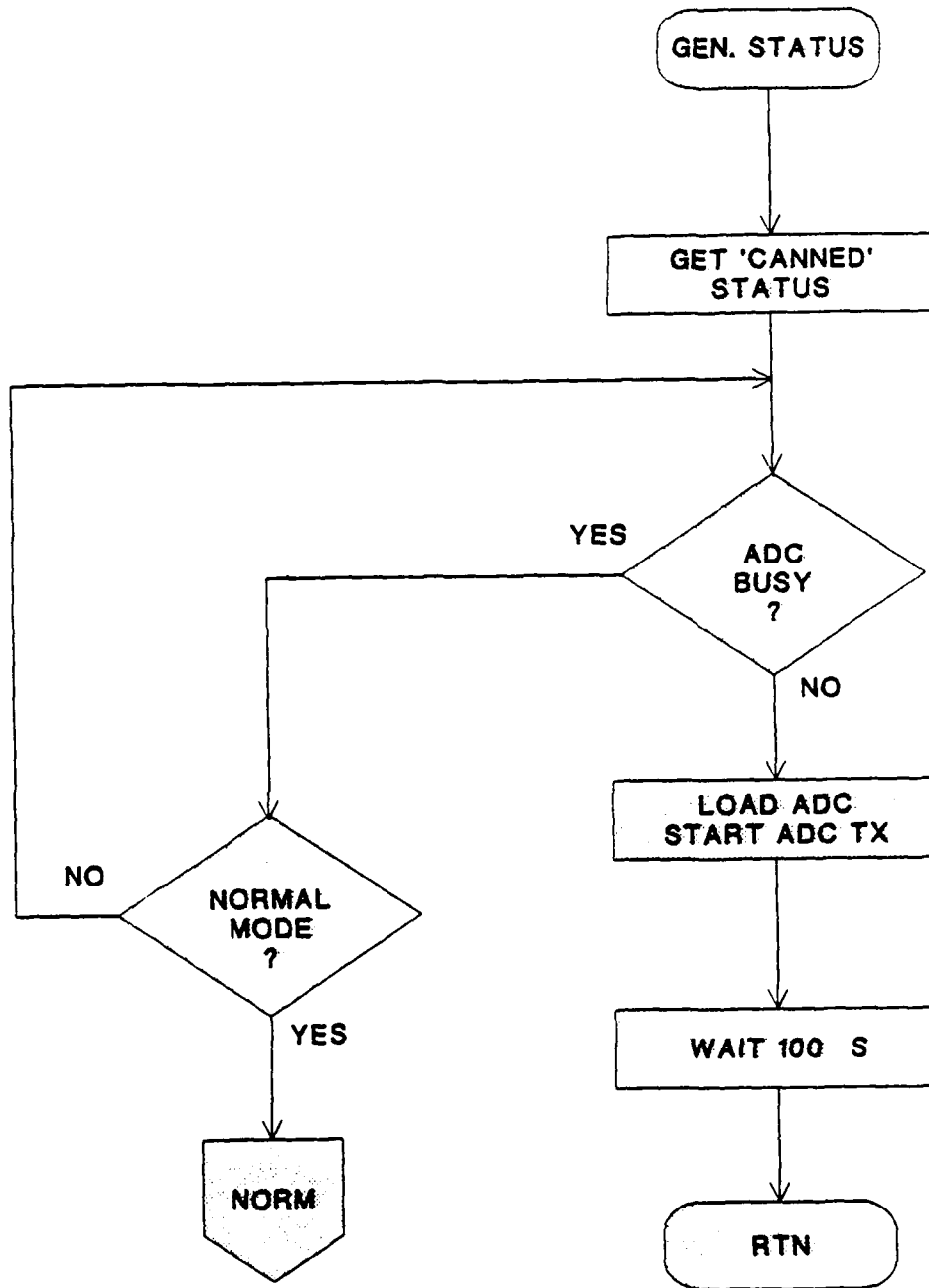


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 13 of 14)

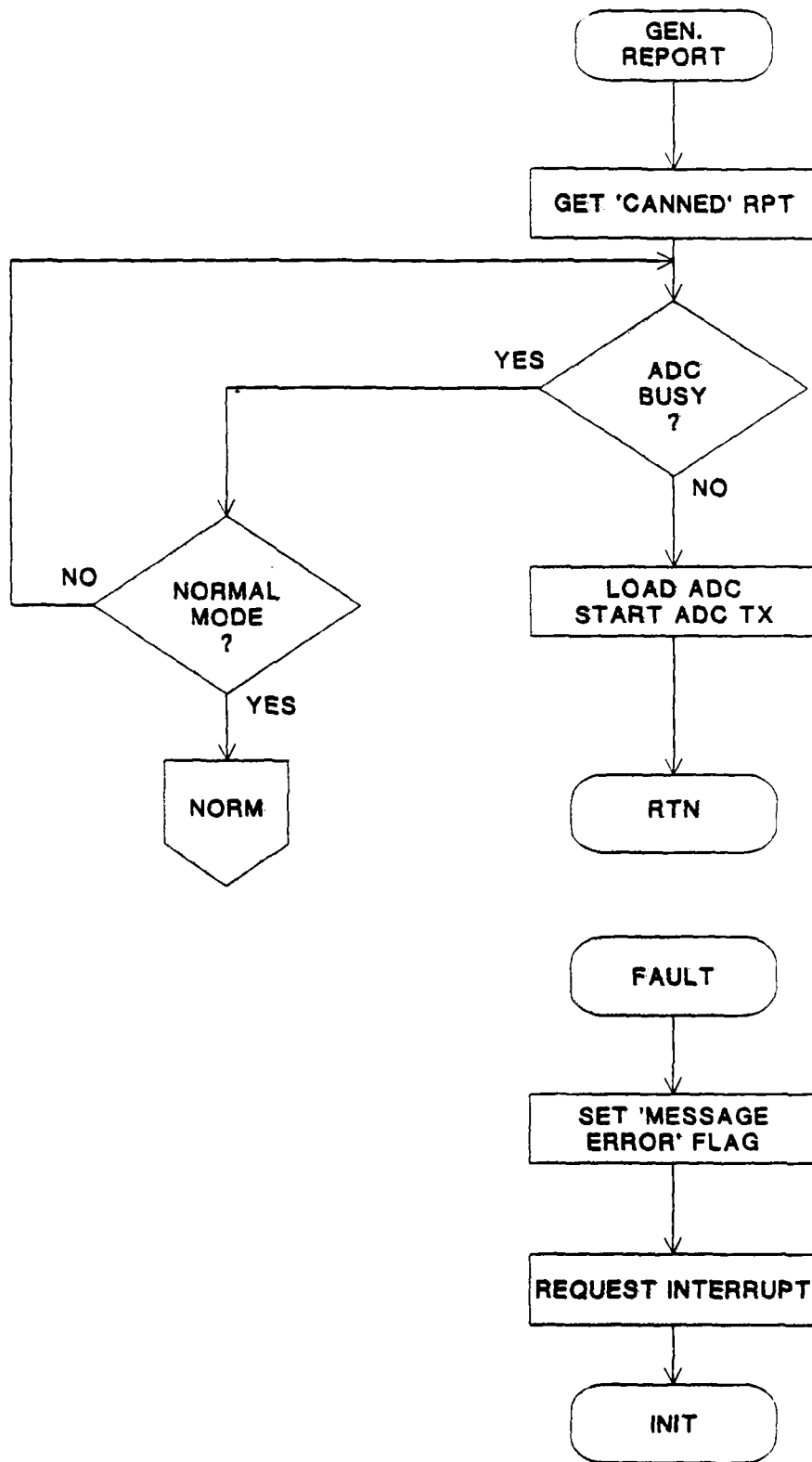


FIGURE B.3.3-1. RRG MICROCODE FLOW DIAGRAMS (Sheet 14 of 14)

TABLE B.3.3-1. RRG MODE SELECTION FIELD

Mode (Oct)	Routine Selection
0	Normal Operation
1	ADC Validation
2	RRC Validation
3	ADC Diagnostic
4-7	RRG Interface Loopback Validation

A printout of the RRG Microcode program is presented in figure B.3.3-2. This is the AMDASM Phase II output listing, consisting of 24 pages. All of the RRC commands are presented on page 1.

When the mode selected by the CPU is normal, the RRC proceeds to sequence radar and status messages to the ADC. Two conditions must be met before a message is transferred: (1) a message must be available in the RRG Buffer Interface, and (2) the ADC must be ready to accept the message. The number of words transferred in the message depends upon the type. Table B.3.3-2 lists the message types handled by the RRC. After each messages is sent, the message counter in the RRG Buffer Interface is decremented.

TABLE B.3.3-2. RRG MESSAGE TYPES

Primary Radar	Message Type	I.D. (Hex)	Word Length
ASR-9	Status	00C0	4
ASR-9	Radar (Search)	01B0	7
ASR-9	Search RTQC	0920	7
CD-2	Status	00C2	4
CD-2	Radar (Search)	01B2	7
CD-2	Radar RTQC	0922	7

If the Interface Loopback Validation is selected, the data paths and memories of the RRG Buffer Interface can be verified. However, like the MRG and the FRG, the RRG does not participate in this test (see page 4, labeled "DIAG:.") It constantly monitors the operation mode field.

If the RRC Validation is selected, the RRC proceeds to fetch 4-word blocks (unformatted) from the RRG Buffer Interface and return them to the CPU. (See page 8, titled "MICROPROCESSOR LOOPBACK ROUTINE.") In this mode, the data paths between the RRC and the RRG Buffer Interface can be verified.

```

:-----:
:
:  RADAR REPORT GENERATOR MICROPROCESSOR COMMANDS
:
:          :          :UNIT: OP :
:          :          : ID :CODE:  DESCRIPTION
:-----:
:  ADCCP DATA :ADCIREAD: 0 : 4 : READBACK ADC DATA
:  CONVERTER   :ADCISEND: 0 : 5 : SEND MESSAGE
:              :ADCIDLE: 0 : 6 : LOAD ADC IDLE REGISTER
:              :ADCIDATA: 0 : 7 : LOAD ADC DATA BUFFER
:              :      : : :
:  RRG        :MSGFAULT: 6 : 0 : SET "MESSAGE FAULT" ERROR FLAG
:  INTERFACE  :RRGSTAT: 6 : 1 : READ RRG STATUS
:              : READ3 : 5 : 2 : READ 3 WORDS FROM RRG I/F
:              : READ4 : 6 : 3 : READ 4 WORDS FROM RRG I/F
:              :FIFOFULL: 6 : 4 : SET "FIFO FULL" DIAG. ERROR FLAG
:              :LOADRAM: 6 : 5 : LOAD DATA INTO I/F RAM
:              :DECMC : 6 : 6 : DECREMENT MESSAGE COUNTER
:              :RRGATN : 6 : 7 : ISSUE RRG INTERRUPT
:              :FCSFAULT: 6 : 8 : SET "FCSFAULT" ERROR FLAG
:              :      : : :
:-----:
0004 ADCIREAD: EQU H*0004
0005 ADCISEND: EQU H*0005
0006 ADCIDLE: EQU H*0006
0007 ADCIDATA: EQU H*0007
0060 MSGFAULT: EQU H*0060
0061 RRGSTAT: EQU H*0061
0062 READ3: EQU H*0062
0063 READ4: EQU H*0063
0064 FIFOFULL: EQU H*0064
0065 LOADRAM: EQU H*0065
0066 DECMC: EQU H*0066
0067 RRGATN: EQU H*0067
0068 FCSFAULT: EQU H*0068
:-----:
:
:  REGISTER ALLOCATIONS
:
:  R0 = RESERVED FOR ADDRESS/CONTROL FIELDS
:  R1 - R9 = RESERVED FOR DATA TRANSFER
:  R10 & R11 = RESERVED FOR IDLE FRAME AND ABORT FRAME
:  R12 - R13 = GENERAL PURPOSE REGISTERS
:  R14 = ADC DIAG LOGIC READY 0 = NO / 1 = YES
:  R15 = GENERAL PURPOSE REGISTER
:
:-----:
:
:  RRG I/F STATUS & COMMAND MASK
:-----:
0007 NORMASK: EQU H*0007
0003 GOMASK: EQU H*0003
8000 PEMASK: EQU H*8000
4000 MSGMASK: EQU H*4000
:-----:
:
:  OTHER THINGS
:-----:

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 1 of 23)



AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

```

2000 ADCBUSY: EQU H*2000 ;ADCCP DATA CONVERTER BUSY?
1000 DIAGADC: EQU H*1000 ;ADCCP DIAGNOSTIC DATA AVAILABLE?
0800 CRCERROR: EQU H*0800 ;ADCCP DIAGNOSTIC CRC ERROR DETECT?
007E FLAG: EQU H*007E ;IDLE FRAME
00FF ABORT: EQU H*00FF ;ABORT FRAME
FF40 ADDR CMD: EQU H*FF40 ;ADDRESS AND CONTROL FIELDS
F0C0 ASR9STID: EQU H*F0C0 ;ASR9 STATUS MESSAGE ID
F920 ARTQCID: EQU H*F920 ;ASR9 SEARCH RTQC MESSAGE ID
F180 ASR9RID: EQU H*F180 ;ASR9 SEARCH MESSAGE ID
F0C2 CD2STID: EQU H*F0C2 ;CD2 STATUS MESSAGE ID
F922 CRTQCID: EQU H*F922 ;CD2 SEARCH RTQC MESSAGE ID
F182 CD2RID: EQU H*F182 ;CD2 SEARCH MESSAGE ID
0FFF HDRMSK: EQU H*0FFF ;ARIES HEADER MASK
0080 DELAY1: EQU H*0080 ;64 MICROSECOND DELAY
0100 DELAY2: EQU H*0100 ;128 MICROSECOND DELAY
AAAA FILLER: EQU H*AAAA ;FILLER FOR MISSING ADC DATA
01A8 TIMER1: EQU H*01A8 ;100 MICROSECOND DELAY
0000 ZERO: EQU H*0000 ;ADC DIAG LOGIC NOT READY
0001 ONE: EQU H*0001 ;ADC DIAG LOGIC READY
000A COUNT: EQU H*000A ;TEN
;
; CANNED "STATUS" & "SEARCH TARGET" REPORTS
;
00C0 STATWD1: EQU H*00C0 ;HEADER
0AAA STATWD2: EQU H*00AA ;STATUS FLAGS
0CCC STATWD3: EQU H*0CCC
0F0F STATWD4: EQU H*0F0F
;
0180 RPTWD1: EQU H*0180 ;HEADER
0800 RPTWD2: EQU H*0800 ;RANGE 32 NAUTICAL MILES
; AZIMUTH
;
0B38 RPTWD4: EQU H*0B38 ;USED FOR WORDS 4 THROUGH 7
;
; INITIALIZATION
;
0000 ORG 0
0000 INIT1: LIM FLAG,R10 ;STORE IDLE FLAG IN REG "10"
0001 LIM ABORT,R11 ;STORE ABORT FLAG IN REG "11"
0002 LIM ADDR CMD,R0 ;STORE ADDRESS/CONTROL FIELDS
; IN REG "0"
0003 LIM ZERO,R14 ;ADC DIAG LOGIC NOT READY
;
0004 INIT2: LIMCR RRGSTAT ;INITIALIZE THE MESSAGE COUNTER
0005 NOP ;TO ZERO
0006 INQ
0007 ANDIQ MSGMASK
0008 BRZ INIT3
0009 LIMCR DECMC
000A RSTCR
000B JMP INIT2
;
000C INIT3: LIMCR ADCIDLE ;LOAD ABORT FIELD INTO ADC TO
000D RSTCR ;RESET THE DEVICE
000E OUT R11
;
000F LIM DELAY1,R15 ;SET UP FOR A 64US PAUSE
0010 LOOPT1: DEC R15 ;LOOP UNTIL R15 REACHES ZERO

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 2 of 23)

```

0011          BNZ LOOPT1

0012          LIMCR ADCIDLE          ;LOAD IDLE FIELD INTO ADC TO
0013          RSTCR                  ;SYNCHRONIZE THE DEVICE
0014          OUT R10

0015          LIM DELAY2,R15         ;SET UP FOR A 128US PAUSE
0016 LOOPT2:  DEC R15                ;LOOP UNTIL R15 REACHES ZERO
0017          BNZ LOOPT2

;::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
;:::::::::::::::::::::::::: MAIN PROGRAM ::::::::::::::::::::::::::::
;::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

0018 BEGIN:  LIMCR RRGSTAT          ;LOAD RRG STATUS INTO THE
0019          NOP                    ;IO-REGISTER. TEST THE
001A LOOPA:  INQ                    ;OPERATION FIELD TO DETERMINE
001B          ANDIQ NORMASK         ;WHETHER TO RUN IN THE NORMAL
001C          BNZ DIAG              ;MODE OR THE DIAGNOSTIC MODE.

001D          LIM ZERO,R14          ;ADC DIAG LOGIC NOT READY
;                                     IF IN THE NORMAL MODE, TEST THE
001E NORM:  INQ                    ;MESSAGE COUNTER TO DETERMINE IF
001F          ANDIQ MSGMASK         ;ANY MESSAGES ARE IN THE INPUT BUFFER.
0020          BRZ LOOPA             ;IF EMPTY, CHECK OPERATING STATUS
;                                     AGAIN.

0021          LIMCR READ4           ;IF A MESSAGE IS AVAILABLE, READ
0022          RSTCR                 ;FOUR WORDS FROM THE INPUT
0023          NOP                    ;BUFFER.
0024          IN R1                  ;
0025          IN R2                  ;
0026          IN R3                  ;
0027          IN R4                  ;

0028          CMPI CD2STID,R1,R15   ;TEST FOR CD2 STATUS MESSAGE.
0029          BRZ SHORT              ;IF TRUE, GO TO SHORT TRANSFER.
002A          CMPI ASR9STID,R1,R15  ;TEST FOR ASR9 STATUS MESSAGE.
002B          BRZ SHORT              ;IF TRUE, GO TO SHORT TRANSFER.

002C          CMPI CD2RID,R1,R15    ;TEST FOR CD2 SEARCH MESSAGE.
002D          BRZ SHORT              ;IF TRUE, GO TO SHORT TRANSFER.
002E          CMPI ASR9RID,R1,R15   ;TEST FOR ASR9 SEARCH MESSAGE.
002F          BRZ LONG              ;IF TRUE, GO TO LONG TRANSFER.
0030          CMPI CRTQCID,R1,R15   ;TEST FOR CD2 RTQC MESSAGE.
0031          BRZ SHORT              ;IF TRUE, GO TO SHORT TRANSFER.
0032          CMPI ARTQCID,R1,R15   ;TEST FOR ASR9 RTQC MESSAGE.
0033          BNZ FAULT              ;IF FALSE, GO TO FAULT ROUTINE.

0034 LONG:  ANDI MORMSK,R1,R1       ;MASK OFF ARIES SYNC FIELD.

0035          LIMCR READ3           ;READ THREE ADDITIONAL WORDS FROM
0036          RSTCR                 ;THE INPUT BUFFER.
0037          NOP
0038          IN R5
0039          IN R6
003A          IN R7

003B          LIMCR RRGSTAT         ;LOAD RRG STATUS INTO THE

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 3 of 23)

```

003C      NOP                      ;Q-REGISTER. TEST STATUS TO
003D LOOPC: INQ                    ;DETERMINE WHETHER THE ADC IS
003E      ANDIQ ADCBUSY             ;BUSY. IF BUSY, CONTINUE
003F      BRZ LOOPC                ;LOOPING UNTIL AVAILABLE
;
0040      LIMCR ADCIOATA            ;LOAD MESSAGE INTO ADC
0041      OUT R0                    ;PREFIX ADDRESS AND CONTROL FIELDS
0042      OUT R1                    ;TO THE MESSAGE PRIOR TO LOADING
0043      OUT R2
0044      OUT R3
0045      OUT R4
0046      OUT R5
0047      OUT R6
0048      OUT R7
;
0049      LIMCR DECMC              ;DECREMENT MESSAGE COUNTER
004A      LIMCR ADCISEND           ;SEND "START MESSAGE TRANSMISSION"
004B      LIMCR RRGSTAT           ;TO THE ADC
;
004C      JMP LOOPA
;
004D SHORT: ANQI HORMSK,R1,R1      ;MASK OFF ARIES SYNC FIELD.
;
004E      LIMCR RRGSTAT           ;IF THE MESSAGE IS A STATUS REPORT
004F      NOP                      ;GO AND CHECK THE BUSY STATUS OF
0050 LOOPB: INQ                    ;THE ADCCP DATA CONVERTER. IF
0051      ANDIQ ADCBUSY           ;NOT BUSY GO AND LOAD THE MESSAGE
0052      BRZ LOOPB              ;INTO THE ADC. IF BUSY WAIT UNTIL
;                                ;IT BECOMES AVAILABLE.
0053      LIMCR ADCIOATA            ;LOAD MESSAGE INTO THE ADC
0054      OUT R0                    ;PREFIX ADDRESS AND COMMAND FIELDS
0055      OUT R1                    ;TO MESSAGE.
0056      OUT R2
0057      OUT R3
0058      OUT R4
;
0059      LIMCR DECMC              ;DECREMENT MESSAGE COUNTER
005A      LIMCR ADCISEND           ;SEND "START MESSAGE TRANSMISSION"
005B      LIMCR RRGSTAT           ;TO THE ADC.
;
005C      JMP LOOPA
;
;=====
;                   DIAGNOSTIC PACKAGE                   ;=====
;=====
005D DIAG:  DECO
005E      DECO                      ;IS THE ADC DIAG MODE SELECTED
005F      BRZ ADCDIAG             ;IF SO EXECUTE ADCDIAG ROUTINE
0060      DECO                      ;IS THE MICROPROCESSOR DIAG. SELECTED
0061      BRZ UPRCDIAG           ;IF SO, EXECUTE UPRCDIAG
0062      DECO                      ;IS THE ADC DEBUG ROUTINE SELECTED?
0063      BRZ ADCDEBUG           ;IF SO EXECUTE ADCDEBUG
0064      JMP BEGIN                ;OTHERWISE ASSUME I/F LOOPBACK DIAG
;
;          ADCCP DATA CONVERTER DIAGNOSTIC ROUTINE          ;
;
; THIS ROUTINE WHEN SELECTED PERFORMS AN ADC LOOPBACK TEST.

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 4 of 23)

```

;      MESSAGES ARE READ FROM THE MESSAGE INPUT BUFFER AND LOADED
;      INTO THE ADC. THE ADC IS ISSUED A START TRANSMISSION
;      TO OUTPUT THE MESSAGE IN THE PROPER ADCCP MESSAGE FORMAT.
;      THE MESSAGE IS LOOPEd BACK THROUGH THE AJC DIAGNOSTIC LOGIC
;      ALONG WITH THE FRAME CHECK SEQUENCE AND STORED IN THE
;      OUTPUT BUFFER TO BE READ BY THE CPU. ALSO THE FCS IS TESTED
;      FOR PROPER GENERATION. IF THE TEST FAILS THE FCS FLAG IS SET.
;
;=====
0065 ADDDIAG:  CMPI ZERO,R14,R13      ;ADC DIAG LOGIC READY?
0066           BNZ ADDDIAG1          ;IF SO GO AND TEST FOR AVAILABLE
;                                     ;MESSAGES. IF NO PREPARE ADC DIAG LOG
0067           LIMCR ADCIDLE         ;LOAD ABORT FIELD INTO ADC TO
0068           RSTCR                 ;RESET THE DEVICE.
0069           OUT R11
;
006A           LIM DELAY1,R15        ;SET UP FOR A 64 US PAUSE
006B LOOP3:   DEC R15                ;LOOP UNTIL R15 REACHES ZERO
006C           BNZ LOOP3
;
006D           LIMCR ADCIDLE         ;LOAD IDLE FIELD INTO ADC TO
006E           LIMCR RRGSTAT        ;SYNCHRONIZE THE DEVICE
006F           OUT R10
;
0070           LIM DELAY2,R15        ;SET UP FOR A 128 US PAUSE
0071 LOOP4:   DEC R15                ;LOOP UNTIL R15 REACHES ZERO
0072           BNZ LOOP4
;
0073           LIM ONE,R14           ;SET ADC DIAG LOGIC READY
;
0074 ADDDIAG1: INQ                   ;LOAD RRG STATUS INTO THE
0075           ANDIQ MSGMASK         ;Q-REGISTER. TEST MESSAGE COUNTER
0076           BRZ BEGIN             ;TO DETERMINE IF ANY MESSAGES ARE
;                                     ;AVAILABLE. IF A MESSAGE IS
0077           LIMCR READ4           ;AVAILABLE, READ FOUR WORDS
0078           RSTCR                 ;FROM THE INPUT BUFFER
0079           NOP
007A           IN R1
007B           IN R2
007C           IN R3
007D           IN R4
;
007E           CMPI ASR9STID,R1,R15 ;NEXT, DETERMINE WHETHER THE
007F           BNZ ARDRPT            ;MESSAGE IS A STATUS REPORT. IF
;                                     ;NOT GO AND TEST FOR SEARCH REPORTS
;
0080           LIMCR RRGSTAT         ;IF THE MESSAGE IS A STATUS REPORT,
0081           NOP                   ;TEST TO SEE IF THE ADC IS BUSY. IF
0082 LOOPD:   INQ                   ;THE ADC IS AVAILABLE PROCEED TO
0083           ANDIQ ADCBUSY         ;TRANSFER THE STATUS MESSAGE
0084           BNZ SENDSTAT          ;INTO THE ADC
;
0085           INQ                   ;IF NOT CHECK THE OPERATING STATUS OF
0086           ANDIQ NORMASK         ;RRG. IF STILL IN DIAGNOSTIC
0087           BNZ LOOPD             ;MODE CONTINUE LOOPING UNTIL THE
;                                     ;ADC BECOMES AVAILABLE
0088           JMP NORM              ;OTHER WISE JUMP TO THE NORMAL ROUTINE
;
0089 SENDSTAT: LIMCR ADCIDATA        ;OUTPUT MESSAGE TO THE ADCCP DATA

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 5 of 23)

```

008A      OUT R0      ;CONVERTER
008B      OUT R1
008C      OUT R2
008D      OUT R3
008E      OUT R4

008F      LIMCR DECMC ;DECREMENT MESSAGE COUNTER.
0090      LIMCR ADCISEND ;SEND "START MESSAGE TRANSMISSION"
                                TO THE ADC.
0091      LIMCR RRGSTAT ;LOAD THE RRG STATUS INTO THE
0092      NOP          ;Q-REGISTER. TEST TO SEE IF
0093 LOOPE: INQ        ;LOOPBACK DATA IS AVAILABLE. IF SO
0094      ANDIQ DIAGADC ;GO AND READ THE DATA
0095      BNZ READ6

0096      INQ          ;IF NO DATA IS AVAILABLE, CHECK THE
0097      ANDIQ NORMASK ;OPERATING STATUS OF THE RRG. IF
0098      BNZ LOOPE    ;STILL IN THE DIAGNOSTIC MODE CONTINUE
                                ;LOOPING UNTIL DATA BECOMES AVAILABLE
0099      JMP NORM     ;OTHERWISE JUMP TO NORMAL ROUTINE.

009A READ6: JSR ROADCS ;READ ADC LOOPBACK DATA.

009B      INQ          ;CHECK FOR CRC ERROR.
009C      ANDIQ CRCERROR
009D      BRZ LOOPR
009E      LIMCR FCSFAULT
009F      LIMCR RRGATN
00A0      LIMCR RRGSTAT
00A1      NOP

                                ;TEST TO SEE IF THE ADC OUTPUT FIFO
                                ;IS EMPTY. IF NOT, SET FIFO FULL
                                ;FLAG AND ISSUE AN INTERRUPT BACK TO
                                ;THE CPU
00A2 LOOPR: INQ
00A3      ANDIQ DIAGADC
00A4      BRZ LOOPF
00A5      LIMCR FIFOFULL
00A6      LIMCR RRGATN ;SET "FIFO FULL" FLAG
                                ;ISSUE INTERRUPT REQUEST

00A7      LIMCR RRGSTAT
00A8      NOP
00A9 LOOPF: INQ
00AA      ANDIQ PEMASK
00AB      BNZ DUMP6 ;TEST TO SEE IF THE OUTPUT RAM IS
                                ;IN THE WRITE MODE. IF THE OUTPUT
                                ;RAM IS IN THE WRITE MODE PROCEED
                                ;TO TRANSFER DATA INTO THE RAM.
                                ;IF NOT, CHECK THE OPERATING STATUS
                                ;OF THE RRG. IF STILL IN THE DIAGNOSTIC
                                ;MODE, CONTINUE LOOPING UNTIL THE
                                ;OUTPUT RAM IS IN THE WRITE MODE.
                                ;OTHERWISE JUMP TO THE NORMAL ROUTINE.
00AC      INQ
00AD      ANDIQ NORMASK
00AE      BNZ LOOPF

00AF      JMP NORM

00B0 DUMP6: LIMCR LOADRAM ;LOAD ALL DATA INTO THE OUTPUT RAM
00B1      OUT R1      ;ADDRESS AND CONTROL FRAMES
00B2      OUT R2      ;STATUS MESSAGE
00B3      OUT R3
00B4      OUT R4
00B5      OUT R5
00B6      OUT R6      ;FRAME CHECK SEQUENCE

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 6 of 23)

```

00B7          RSTCR
00B8          JMP BEGIN
00B9 ARDRPT:  CMPI ASR9RID,R1,R15 ; DETERMINE IF THE MESSAGE IS A SEARCH
00BA          BRZ ARDRPT1        ; TARGET MESSAGE OR SEARCH RTQC MESSAGE
00BB          CMPI ARTQCID,R1,R15 ; IF IT IS GO AND READ THREE ADDITIONAL
00BC          BNZ FAULT          ; WORDS FROM THE INPUT BUFFER. IF IT
                                ; IS UNKNOWN JUMP TO FAULT ROUTINE.

00BD ARDRPT1: LIMCR READ3        ; READ 3 ADDITIONAL WORDS FROM
00BE          RSTCR              ; THE INPUT BUFFER
00BF          NOP
00C0          IN R5
00C1          IN R6
00C2          IN R7

00C3          LIMCR RRGSTAT      ; LOAD THE RRG STATUS INTO THE
00C4          NOP                ; Q-REGISTER TEST TO SEE IF THE
00C5 LOOPG:   INQ                ; ADC IS BUSY. IF NOT BUSY TRANSFER
00C6          ANDIQ ADCBUSY      ; COMPLETE MESSAGE TO THE ADC
00C7          BNZ SENDRPT

00C8          INQ                ; IF BUSY CHECK THE OPERATING STATUS
00C9          ANDIQ NORMASK      ; OF THE RRG. IF STILL IN THE DIAG.
00CA          BNZ LOOPG         ; MODE CONTINUE LOOPING UNTIL THE ADC
                                ; BECOMES AVAILABLE.
00CB          JMP NORM          ; OTHERWISE JUMP TO THE NORMAL ROUTINE

00CC SENDRPT: LIMCR ADCIDATA     ; OUTPUT MESSAGE TO THE ADCCP DATA
00CD          OUT R0              ; CONVERTER
00CE          OUT R1
00CF          OUT R2
00D0          OUT R3
00D1          OUT R4
00D2          OUT R5
00D3          OUT R6
00D4          OUT R7

00D5          LIMCR DECMC        ; DECREMENT MESSAGE COUNTER
00D6          LIMCR ADCISEND     ; SEND "START MESSAGE TRANSMISSION"
                                ; TO ADC
00D7          LIMCR RRGSTAT      ; LOAD THE RRG STATUS INTO THE
00D8          NOP                ; Q-REGISTER. TEST TO SEE IF
00D9 LOOPH:   INQ                ; LOOPBACK DATA IS AVAILABLE. IF SO
00DA          ANDIQ DIAGADC      ; GO AND READ THE DATA
00DB          BNZ READ9

00DC          INQ                ; IF NO DATA IS AVAILABLE, CHECK THE
00DD          ANDIQ NORMASK      ; OPERATING STATUS OF THE RRG. IF
00DE          BNZ LOOPH         ; STILL IN THE DIAG MODE, CONTINUE
                                ; LOOPING UNTIL DATA BECOMES AVAILABLE
00DF          JMP NORM          ; OTHERWISE JUMP TO THE NORMAL ROUTINE

00E0 READ9:   JSR ROADCC        ; READ DATA FROM ADCCP DATA CONVERTER
00E1          JSR ROADCC1

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 7 of 23)

```

00E2      INQ                      ;CHECK FOR CRC ERROR.
00E3      ANDIQ CRCERROR
00E4      BRZ LOOPQ
00E5      LIMCR FCSFAULT
00E6      LIMCR RRGATN
00E7      LIMCR RRGSTAT
00E8      NOP
;
;
00E9 LOOPQ: INQ                      ;TEST TO SEE IF THE ADC OUTPUT FIFO
;                                     IS EMPTY. IF NOT, SET 'FIFO FULL'
;                                     FLAG AND ISSUE AN INTERRUPT BACK TO

00EA      ANDIQ DIAGADC             ;THE CPU
00EB      BRZ LOOPI
00EC      LIMCR FIFOFULL
00ED      LIMCR RRGATN             ;SET THE FIFO FULL DIAG ERROR FLAG
;                                     ;ISSUE INTERRUPT REQUEST
;
00EE      LIMCR RRGSTAT             ;TEST TO SEE IF THE OUTPUT RAM IS
00EF      NOP                      ;IN THE WRITE MODE. IF THE OUTPUT
00F0 LOOPI: INQ                      ;RAM IS IN THE WRITE MODE PROCEED
00F1      ANDIQ PEMASK              ;TO TRANSFER THE DATA INTO THE RAM
00F2      BNZ DUMPS
;
00F3      INQ                      ;IF NOT CHECK THE OPERATING STATUS
00F4      ANDIQ NORMASK             ;OF THE RRG. IF STILL IN DIAGNOSTIC
00F5      BNZ LOOPI                ;MODE, CONTINUE LOOPING UNTIL THE
;                                     OUTPUT RAM IS IN THE WRITE MODE
00F6      JMP NORM                  ;OTHERWISE JUMP TO THE NORMAL ROUTINE
;
00F7 DUMPS: LIMCR LOADRAM           ;LOAD ALL DATA INTO THE OUTPUT RAM
00F8      OUT R1                    ;ADDRESS AND CONTROL FRAMES
00F9      OUT R2                    ;SEARCH TARGET MESSAGE
00FA      OUT R3
00FB      OUT R4
00FC      OUT R5
00FD      OUT R6
00FE      OUT R7
00FF      OUT R8
0100      OUT R9                    ;FRAME CHECK SEQUENCE
0101      RSTCR
;
0102      JMP BEGIN
;
;          MICROPROCESSOR LOOPBACK ROUTINE          ;
;
0103 UPRCDIAG: INQ                 ;LOAD RRG STATUS INTO THE
0104      ANDIQ MSGMASK             ;IQ-REGISTER. TEST MESSAGE
0105      BRZ BEGIN                 ;COUNTER TO DETERMINE IF ANY
;                                     MESSAGES ARE IN THE INPUT BUFFER
0106      LIMCR READ4               ;IF A MESSAGES IS AVAILABLE READ
0107      RSTCR                     ;FOUR WORDS FROM THE INPUT BUFFER
0108      NOP
0109      IN R1
010A      IN R2
010B      IN R3
010C      IN R4
;
;                                     DETERMINE WHETHER THE MESSAGE IS

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 8 of 23)

0100	CMPI ASR9STID,R1,R15	IF A STATUS REPORT OR A RADAR REPORT.
010E	BNZ URDRPT	IF THE MESSAGE IS A RADAR REPORT
		READ IN THREE ADDITIONAL WORDS.
010F	LIMCR RRGSTAT	IF THE MESSAGE IS A STATUS REPORT
0110	NOP	TEST TO SEE IF THE OUTPUT RAM
0111 LOOPJ:	INQ	IS IN THE WRITE MODE. IF THE OUTPUT
0112	ANDIQ PEMASK	RAM IS IN THE WRITE MODE PROCEED TO
0113	BNZ DUMP4	TRANSFER STATUS MESSAGE INTO RAM
0114	INQ	IF NOT CHECK THE OPERATING STATUS OF
0115	ANDIQ NORMASK	THE RRG. IF STILL IN THE DIAGNOSTIC
0116	BNZ LOOPJ	MODE CONTINUE LOOPING UNTIL THE
		OUTPUT RAM IS IN THE WRITE MODE.
0117	JMP NORM	OTHERWISE JUMP TO THE NORMAL ROUTINE.
0118 DUMP4:	LIMCR LOADRAM	LOAD MESSAGE INTO OUTPUT RAM.
0119	OUT R1	
011A	OUT R2	
011B	OUT R3	
011C	OUT R4	
011D	LIMCR DECMC	DECREMENT MESSAGE COUNTER.
011E	RSTCR	
011F	JMP BEGIN	
0120 URDRPT:	LIMCR READ3	R11A OUT R2
011B	OUT R3	
011C	OUT R4	
011D	LIMCR DECMC	DECREMENT MESSAGE COUNTER.
011E	RSTCR	
011F	JMP BEGIN	
0120 URDRPT:	LIMCR READ3	RAM IS
0127	NOP	IN THE WRITE MODE. IF THE OUTPUT
0128 LOOPK:	INQ	RAM IS IN THE WRITE MODE PROCEED
0129	ANDIQ PEMASK	TO TRANSFER LONG MESSAGE INTO THE
012A	BNZ DUMP7	RAM.
012B	INQ	IF NOT CHECK THE OPERATING STATUS OF
012C	ANDIQ NORMASK	THE RRG. IF STILL IN THE DIAGNOSTIC
012D	BNZ LOOPK	MODE CONTINUE LOOPING UNTIL THE
		OUTPUT RAM IS IN THE WRITE MODE.
012E	JMP NORM	OTHERWISE JUMP TO THE NORMAL ROUTINE
012F DUMP7:	LIMCR LOADRAM	LOAD MESSAGE INTO OUTPUT RAM
0130	OUT R1	
0131	OUT R2	
0132	OUT R3	
0133	OUT R4	
0134	OUT R5	
0135	OUT R6	
0136	OUT R7	
0137	LIMCR DECMC	DECREMENT MESSAGE COUNTER
0138	RSTCR	

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 9 of 23)



```

0139          JMP BEGIN
;
; :::::::::::::::::::: ADC DEBUG ROUTINE ::::::::::::::::::::
;
013A ADCDEBUG: IN R12          ;TEST THE RRG STATUS TO DETERMINE
013B          SRA R12          ;THE OPERATING MODE OF THE DEBUG
013C          SRA R12          ;ROUTINE
013D          SRA R12
013E          ANDI GOMASK,R12,R12
013F          BRZ TXSTAT      ;TRANSMIT STATUS MESSAGE?
0140          DEC R12
0141          BRZ TXRPT      ;TRANSMIT SEARCH TARGET MESSAGE?
0142          DEC R12
0143          BRZ TXMIX      ;TRANSMIT BOTH MESSAGES?
0144          JMP BEGIN
;
0145 TXSTAT:   JSR GENSTAT
0146          JMP BEGIN
;
0147 TXRPT:   JSR GENRPT
0148          JMP BEGIN
;
0149 TXMIX:   JSR GENSTAT      ;THE TRANSMIT MIX GENERATES 10 SEARCH
014A          LIM COUNT,R12   ;TARGET MESSAGES FOR EVERY 1 STATUS
014B LOOPL:   JSR GENRPT      ;MESSAGE
014C          DEC R12
014D          BNZ LOOPL
014E          JMP BEGIN
;
; :::::::::::::::::::: SUBROUTINE "TRANSMIT STATUS MESSAGE" ::::::::::::::::::::
;
; THE PURPOSE FOR THIS ROUTINE IS TO TROUBLE SHOOT THE ADCCP
; DATA CONVERTER HARDWARE FOR THE FOLLOWING : RECEIVING A STATUS
; MESSAGE FROM THE MICROCONTROLLER AND TRANSMITTING THE MESSAGE
; IN THE CORRECT ADCCP FORMAT. A CANNED STATUS MESSAGE IS
; LOADED INTO THE ADC FOR TRANSMISSION ONCE EACH TIME THIS
; ROUTINE IS CALLED
;
; ::::::::::::::::::::
014F GENSTAT: LIM STATW01,R1
0150          LIM STATW02,R2
0151          LIM STATW03,R3
0152          LIM STATW04,R4
;
0153 LOOPM:   INQ
0154          ANDIQ ADCBUSY
0155          BNZ STATOUT
;
0156          INQ
0157          ANDIQ NORMASK
0158          BNZ LOOPM
;
0159          RTN
;
015A STATOUT: LIMCR ADC1DATA
015B          OUT R0
015C          OUT R1
015D          OUT R2

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 10 of 23)

```

015E          OUT R3
015F          OUT R4
0160          RSTCR
0161          LIMCR ADCISENO
0162          LIMCR RRGSTAT

0163          LIM TIMER',R12
0164 LOOPN:   DEC R12
0165          BNZ LOOPN

0166          RTN

:
: SUBROUTINE "TRANSMIT SEARCH MESSAGE"
:
: THE PURPOSE OF THIS ROUTINE IS TO TROUBLE SHOOT THE ADCCP
: DATA CONVERTER HARDWARE FOR THE FOLLOWING: RECEIVING A SEARCH
: MESSAGE FROM THE MICROCONTROLLER AND TRANSMITTING THE MESSAGE
: IN THE CORRECT ADCCP FORMAT. A CANNED SEARCH TARGET MESSAGE
: IS LOADED INTO THE ADC FOR TRANSMISSION EACH TIME THIS ROUTINE
: IS CALLED.
:
0167 GENRPT:  LIM RPTWD1,R1
0168          LIM RPTWD2,R2
0169          ADDI ONE,R3,R3
016A          ANDI H#0FFF,R3,R3
016B          LIM RPTWD4,R4

016C LOOP0:  INQ
016D          ANDIQ ADCBUSY
016E          BNZ RPTOUT

016F          INQ
0170          ANDIQ NORMASK
0171          BNZ LOOP0

0172          RTN

0173 RPTOUT:  LIMCR ADCIOATA
0174          OUT R0
0175          OUT R1
0176          OUT R2
0177          OUT R3
0178          OUT R4
0179          OUT R4
017A          OUT R4
017B          OUT R4
017C          RSTCR
017D          LIMCR ADCISENO
017E          LIMCR RRGSTAT

017F          RTN

:
: SUBROUTINE "READ ADC (PART #1)"
:
: THIS ROUTINE ATTEMPTS TO READ SIX WORDS FROM THE ADCCP
: DATA CONVERTER . EACH TIME A WORD IS READ FROM THE FIFO
: THE FIFO'S OUTPUT STATUS IS CHECKED. IF THE FIFO IS FOUND

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 11 of 23)

```

;   EMPTY BEFORE SUCCESSFULLY READING ALL SIX WORDS.
;   THE REMAINDER OF THE SIX REGISTERS ARE LOADED WITH A FILLER
;   (VALUE OF H#AAAA) BEFORE EXITING THE SUBROUTINE.
;   ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
0180 ROADCB: LIMCR ADCIREAD
0181          RSTCR
0182          LIMCR RRGSTAT
0183          IN R1
0184          INQ
0185          ANDIQ DIAGADC
0186          BRZ PASS1

;
0187          LIMCR ADCIREAD
0188          RSTCR
0189          LIMCR RRGSTAT
018A          IN R2
018B          INQ
018C          ANDIQ DIAGADC
018D          BRZ PASS2

;
018E          LIMCR ADCIREAD
018F          RSTCR
0190          LIMCR RRGSTAT
0191          IN R3
0192          INQ
0193          ANDIQ DIAGADC
0194          BRZ PASS3

;
0195          LIMCR ADCIREAD
0196          RSTCR
0197          LIMCR RRGSTAT
0198          IN R4
0199          INQ
019A          ANDIQ DIAGADC
019B          BRZ PASS4

;
019C          LIMCR ADCIREAD
019D          RSTCR
019E          LIMCR RRGSTAT
019F          IN R5
01A0          INQ
01A1          ANDIQ DIAGADC
01A2          BRZ PASS5

;
01A3          LIMCR ADCIREAD
01A4          RSTCR
01A5          LIMCR RRGSTAT
01A6          IN R6
01A7          RTN

;
01A8 PASS1: LIM FILLER,R2
01A9 PASS2: LIM FILLER,R3
01AA PASS3: LIM FILLER,R4
01AB PASS4: LIM FILLER,R5
01AC PASS5: LIM FILLER,R6
01AD          RTN
;   ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 12 of 23)

```

: SUBROUTINE "READ ADC (PART #2)"
: THIS ROUTINE IS A CONTINUATION OF THE SUBROUTINE "READ ADC
: (PART #1)" FOR LONG MESSAGES. THIS ROUTINE ATTEMPTS
: TO READ THREE ADDITIONAL WORDS FROM THE ADCCP DATA
: CONVERTER. EACH TIME A WORD IS READ FROM THE FIFO
: THE FIFO'S OUTPUT STATUS IS CHECKED. IF THE FIFO IS
: FOUND EMPTY BEFORE SUCCESSFULLY READING THE THREE WORDS,
: THE REMAINDER OF THE THREE REGISTERS ARE LOADED WITH
: A FILLER (VALUE OF H*AAAA) BEFORE EXITING THE SUBROUTINE.
:
01AE R0ADC3:  INQ
01AF          ANDIQ DIAGADC
01B0          BRZ PASS6
:
01B1          LIMCR ADC1READ
01B2          RSTCR
01B3          LIMCR RRGSTAT
01B4          IN R7
01B5          INQ
01B6          ANDIQ DIAGADC
01B7          BRZ PASS7
:
01B8          LIMCR ADC1READ
01B9          RSTCR
01BA          LIMCR RRGSTAT
01BB          IN R8
01BC          INQ
01BD          ANDIQ DIAGADC
01BE          BRZ PASS8
:
01BF          LIMCR ADC1READ
01C0          RSTCR
01C1          LIMCR RRGSTAT
01C2          IN R9
01C3          INQ
01C4          ANDIQ DIAGADC
01C5          BRZ PASS9
01C6          RTN
:
01C7 PASS6:  LIM FILLER,R7
01C8 PASS7:  LIM FILLER,R8
01C9 PASS8:  LIM FILLER,R9
01CA PASS9:  RTN
:
: FAULT "UNKNOWN MESSAGE"
:
01CB FAULT:  LIMCR MSGFAULT           ;SET "MESSAGE FAULT" FLAG. UNKNOWN
01CC          LIMCR RRGATN           ;ID FOUND.  ISSUE INTERRUPT BACK TO
:                                     THE CPU
01CD          JPZ                    ;REINITIALIZE RADAR REPORT GENERATOR
:
01CE END

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 13 of 23)

```

0000 0000000001111110 0111000111101110 000101000000010 00010111
0001 0000000011111111 0111000111101110 000101100000010 00010111
0002 1111111101000000 0111000111101110 000000000000010 00010111
0003 0000000000000000 0111000111101110 000111000000010 00010111
0004 0000000001100001 0111000101111110 000000000000000 00011110
0005 0000000000000000 0111000100000000 000000000000000 00011111
0006 0000000000000000 1111000000001110 000000000000010 00010111
0007 0100000000000000 0111000010011100 000000000000010 00010111
0008 0000000000001100 0001100010000000 000000000000010 10110011
0009 0000000001100110 0111000010111110 000000000000000 00011110
000A 0000000011111111 0111000010111110 000000000000000 00011110
000B 0000000000001100 0001110010000000 000000000000000 00011111
000C 0000000000000110 0111000010111110 000000000000000 00011110
000D 0000000011111111 0111000010111110 000000000000000 00011110
000E 0000000000000000 0111000010111001 011000000000000 00011111
000F 0000000010000000 0111000111101110 000111000000010 00010111
0010 0000000000000000 0111000110010110 000111100000010 00010111
0011 0000000000100000 0001100010000000 000000000000010 10010011
0012 0000000000000110 0111000010111110 000000000000000 00011110
0013 0000000011111111 0111000010111110 000000000000000 00011110
0014 0000000000000000 0111000010111001 010000000000000 00011111
0015 0000000010000000 0111000111101110 000111100000010 00010111
0016 0000000000000000 0111000110010110 000111100000010 00010111
0017 0000000000101110 0001100010000000 000000000000010 10010011
0018 0000000001100001 0111000010111110 000000000000000 00011110
0019 0000000000000000 0111000010000000 000000000000000 00011111
001A 0000000000000000 1111000000001110 000000000000010 00010111
001B 0000000000000111 0111000010011000 000000000000010 00010111
001C 0000000001011101 0001100010000000 000000000000010 10010011
001D 0000000000000000 0111000111101110 000111000000010 00010111
001E 0000000000000000 1111000000001110 000000000000010 00010111
001F 0100000000000000 0111000010011100 000000000000010 00010111
0020 0000000000011010 0001100010000000 000000000000010 10110011
0021 0000000001100011 0111000010111110 000000000000000 00011110
0022 0000000011111111 0111000010111110 000000000000000 00011110
0023 0000000000000000 0111000010000000 000000000000000 00011111
0024 0000000000000000 1111000110001110 0000001000000010 00010111
0025 0000000000000000 1111000110001110 000001000000010 00010111
0026 0000000000000000 1111000110001110 000001100000010 00010111
0027 0000000000000000 1111000110001110 000010000000010 00010111
0028 1111000011000010 0111000110101010 0011111010000010 00010111
0029 0000000001001101 0001100010000000 000000000000010 10110011
002A 1111000011000000 0111000110101010 0011111010000010 00010111
002B 0000000001001101 0001100010000000 000000000000010 10110011
002C 1111000110110010 0111000110101010 0011111010000010 00010111
002D 0000000001001101 0001100010000000 000000000000010 10110011
002E 1111000110110000 0111000110101010 0011111010000010 00010111
002F 0000000000110100 0001100010000000 000000000000010 10110011
0030 1111100100100010 0111000110101010 0011111010000010 00010111
0031 0000000001001101 0001100010000000 000000000000010 10110011
0032 1111100100100000 0111000110101010 0011111010000010 00010111
0033 0000000111001011 0001100010000000 000000000000010 10010011
0034 0000111111111111 0111000111001010 0010001000000010 00010111
0035 0000000001100010 0111000010111110 000000000000000 00011110
0036 0000000011111111 0111000010111110 000000000000000 00011110
0037 0000000000000000 0111000010000000 000000000000000 00011111
0038 0000000000000000 1111000110001110 0000101000000010 00010111

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 14 of 23)

```
0039 0000000000000000 1111000110001110 0000110000000010 00010111
003A 0000000000000000 1111000110001110 0000111000000010 00010111
003B 0000000001100001 0111000010111110 0000000000000000 00011110
003C 0000000000000000 0111000010000000 0000000000000000 00011111
003D 0000000000000000 1111000000001110 0000000000000010 00010111
003E 0010000000000000 0111000010011000 0000000000000010 00010111
003F 0000000000111101 0001100010000000 0000000000000010 10110011
0040 0000000000000111 0111000010111110 0000000000000000 00011110
0041 0000000000000000 0111000010111000 0000000000000000 00011111
0042 0000000000000000 0111000010111000 0010000000000000 00011111
0043 0000000000000000 0111000010111000 0100000000000000 00011111
0044 0000000000000000 0111000010111000 0110000000000000 00011111
0045 0000000000000000 0111000010111000 1000000000000000 00011111
0046 0000000000000000 0111000010111000 1010000000000000 00011111
0047 0000000000000000 0111000010111000 1100000000000000 00011111
0048 0000000000000000 0111000010111000 1110000000000000 00011111
0049 0000000001100110 0111000010111110 0000000000000000 00011110
004A 0000000000000101 0111000010111110 0000000000000000 00011110
004B 0000000001100001 0111000010111110 0000000000000000 00011110
004C 0000000000011010 0001110010000000 0000000000000000 00011111
004D 0000111111111111 0111000111001010 0010001000000010 00010111
004E 0000000001100001 0111000010111110 0000000000000000 00011110
004F 0000000000000000 0111000010000000 0000000000000000 00011111
0050 0000000000000000 1111000000001110 0000000000000010 00010111
0051 0010000000000000 0111000010011000 0000000000000010 00010111
0052 0000000001010000 0001100010000000 0000000000000010 10110011
0053 0000000000000111 0111000010111110 0000000000000000 00011110
0054 0000000000000000 0111000010111000 0000000000000000 00011111
0055 0000000000000000 0111000010111000 0010000000000000 00011111
0056 0000000000000000 0111000010111000 0100000000000000 00011111
0057 0000000000000000 0111000010111000 0110000000000000 00011111
0058 0000000000000000 0111000010111000 1000000000000000 00011111
0059 0000000001100110 0111000010111110 0000000000000000 00011110
005A 00000000000000101 0111000010111110 0000000000000000 00011110
005B 0000000001100001 0111000010111110 0000000000000000 00011110
005C 00000000000011010 0001110010000000 0000000000000000 00011110
005D 0000000000000000 0111000000101000 0000000000000010 00010111
005E 0000000000000000 0111000000101000 0000000000000010 00010111
005F 0000000001100101 0001100010000000 0000000000000010 10110011
0060 0000000000000000 0111000000101000 0000000000000010 00010111
0061 0000000100000011 0001100010000000 0000000000000010 10110011
0062 0000000000000000 0111000000101000 0000000000000010 00010111
0063 0000000100111010 0001100010000000 0000000000000010 10110011
0064 0000000000011000 0001110010000000 0000000000000000 00011111
0065 0000000000000000 0111000110101011 1101101010000010 00010111
0066 0000000001110100 0001100010000000 0000000000000010 10010011
0067 0000000000000110 0111000010111110 0000000000000000 00011110
0068 0000000001111111 0111000010111110 0000000000000000 00011110
0069 0000000000000000 0111000010111001 0110000000000000 00011111
006A 0000000001000000 0111000111101110 0001111000000010 00010111
006B 0000000000000000 0111000110010110 0001111000000010 00010111
006C 0000000001101011 0001100010000000 0000000000000010 10010011
006D 0000000000000110 0111000010111110 0000000000000000 00011110
006E 0000000001100001 0111000010111110 0000000000000000 00011110
006F 0000000000000000 0111000010111001 0100000000000000 00011111
0070 0000000100000000 0111000111010110 0001111000000010 00010111
0071 0000000000000000 0111000110010110 0001111000000010 00010111
```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 15 of 23)

```

0072 0000000001110001 00011000!0000000 0000000000000010 10010011
0073 0000000000000001 0111000111101110 0001110000000010 00010111
0074 0000000000000000 1111000000001110 0000000000000010 00010111
0075 0100000000000000 011100001001100 0000000000000010 00010111
0076 0000000000011000 0001100010000000 0000000000000010 10110011
0077 0000000001100011 0111000010111110 0000000000000000 00011110
0078 0000000011111111 0111000010111110 0000000000000000 00011110
0079 0000000000000000 0111000010000000 0000000000000000 00011111
007A 0000000000000000 11110001!0001110 0000001000000010 00010111
007B 0000000000000000 11110001!0001110 0000010000000010 00010111
007C 0000000000000000 11110001!0001110 0000011000000010 00010111
007D 0000000000000000 11110001!0001110 0000100000000010 00010111
007E 111100001!0000000 01110001!0101010 0011111010000010 00010111
007F 0000000010111001 0001100010000000 0000000000000000 10010011
0080 0000000001100001 0111000010111110 0000000000000000 00011110
0081 0000000000000000 01110000!0000000 0000000000000000 00011111
0082 0000000000000000 1111000000001110 0000000000000010 00010111
0083 0010000000000000 0111000001001100 0000000000000010 00010111
0084 0000000010001001 0001100010000000 0000000000000010 10010011
0085 0000000000000000 1111000000001110 0000000000000010 00010111
0086 0000000000000011 0111000001001100 0000000000000010 00010111
0087 0000000010000010 00011000!0000000 0000000000000010 10010011
0088 0000000000011110 0001110010000000 0000000000000000 00011111
0089 0000000000000011 0111000010111110 0000000000000000 00011110
008A 0000000000000000 0111000010111000 0000000000000000 00011111
008B 0000000000000000 01110000!0111000 0010000000000000 00011111
008C 0000000000000000 0111000010111000 0100000000000000 00011111
008D 0000000000000000 01110000!0111000 0100000000000000 00011111
008E 0000000000000000 01110000!0111000 1000000000000000 00011111
008F 0000000001100110 01110000!0111110 0000000000000000 00011110
0090 0000000000000101 01110000!0111110 0000000000000000 00011110
0091 0000000001100001 01110000!0111110 0000000000000000 00011110
0092 0000000000000000 01110000!0000000 0000000000000000 00011111
0093 0000000000000000 1111000000001110 0000000000000010 00010111
0094 0001000000000000 0111000001001100 0000000000000010 00010111
0095 0000000010011010 00011000!0000000 0000000000000010 10010011
0096 0000000000000000 1111000000001110 0000000000000010 00010111
0097 0000000000000011 0111000001001100 0000000000000010 00010111
0098 0000000010010011 0001100010000000 0000000000000010 10010011
0099 0000000000011110 0001110010000000 0000000000000000 00011111
009A 00000001!0000000 00001100!0000000 0000000000000000 01001111
009B 0000000000000000 1111000000001110 0000000000000010 00010111
009C 0000100000000000 0111000001001100 0000000000000010 00010111
009D 0000000010100010 0001100010000000 0000000000000010 10110011
009E 0000000001101000 01110000!0111110 0000000000000000 00011110
009F 0000000001100111 01110000!0111110 0000000000000000 00011110
00A0 0000000001100001 01110000!0111110 0000000000000000 00011110
00A1 0000000000000000 01110000!0000000 0000000000000000 00011111
00A2 0000000000000000 1111000000001110 0000000000000010 00010111
00A3 0001000000000000 0111000001001100 0000000000000010 00010111
00A4 0000000010101001 0001100010000000 0000000000000010 10110011
00A5 0000000001100100 01110000!0111110 0000000000000000 00011110
00A6 0000000001100111 01110000!0111110 0000000000000000 00011110
00A7 0000000001100001 01110000!0111110 0000000000000000 00011110
00A8 0000000000000000 01110000!0000000 0000000000000000 00011111
00A9 0000000000000000 1111000000001110 0000000000000010 00010111
00AA 1000000000000000 01110000!001100 0000000000000010 00010111

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 16 of 23)

```

00AB 0000000010110000 0001100010000000 0000000000000010 10010011
00AC 0000000000000000 1111000000001110 0000000000000010 00010111
00AD 0000000000000111 0111000010011100 0000000000000010 00010111
00AE 0000000010101001 0001100010000000 0000000000000010 10010011
00AF 0000000000111110 0001110010000000 0000000000000000 00011111
00B0 0000000001100101 0111000010111110 0000000000000000 00011110
00B1 0000000000000000 0111000010111000 0010000000000000 00011111
00B2 0000000000000000 0111000010111000 0100000000000000 00011111
00B3 0000000000000000 0111000010111000 0110000000000000 00011111
00B4 0000000000000000 0111000010111000 1000000000000000 00011111
00B5 0000000000000000 0111000010111000 1010000000000000 00011111
00B6 0000000000000000 0111000010111000 1100000000000000 00011111
00B7 0000000011111111 0111000010111110 0000000000000000 00011110
00B8 0000000000110000 0001110010000000 0000000000000000 00011111
00B9 1111000110110000 0111000110101010 0011111010000010 00010111
00BA 0000000010111101 0001100010000000 0000000000000010 10110011
00BB 1111100100100000 0111000110101010 0011111010000010 00010111
00BC 0000000111001011 0001100010000000 0000000000000010 10010011
00BD 0000000001100010 0111000010111110 0000000000000000 00011110
00BE 0000000011111111 0111000010111110 0000000000000000 00011110
00BF 0000000000000000 0111000010000000 0000000000000000 00011111
00C0 0000000000000000 1111000110001110 0000101000000010 00010111
00C1 0000000000000000 1111000110001110 0000110000000010 00010111
00C2 0000000000000000 1111000110001110 0000111000000010 00010111
00C3 0000000001100001 0111000010111110 0000000000000000 00011110
00C4 0000000000000000 0111000010000000 0000000000000000 00011111
00C5 0000000000000000 1111000000001110 0000000000000010 00010111
00C6 0010000000000000 0111000010011100 0000000000000010 00010111
00C7 0000000011001100 0001100010000000 0000000000000010 10010011
00C8 0000000000000000 1111000000001110 0000000000000010 00010111
00C9 0000000000000111 0111000010011100 0000000000000010 00010111
00CA 0000000011000101 0001100010000000 0000000000000010 10010011
00CB 0000000000111110 0001110010000000 0000000000000000 00011111
00CC 0000000000000111 0111000010111110 0000000000000000 00011110
00CD 0000000000000000 0111000010111000 0000000000000000 00011111
00CE 0000000000000000 0111000010111000 0010000000000000 00011111
00CF 0000000000000000 0111000010111000 0100000000000000 00011111
00D0 0000000000000000 0111000010111000 0110000000000000 00011111
00D1 0000000000000000 0111000010111000 1000000000000000 00011111
00D2 0000000000000000 0111000010111000 1010000000000000 00011111
00D3 0000000000000000 0111000010111000 1100000000000000 00011111
00D4 0000000000000000 0111000010111000 1110000000000000 00011111
00D5 0000000001100110 0111000010111110 0000000000000000 00011110
00D6 0000000000000101 0111000010111110 0000000000000000 00011110
00D7 0000000001100001 0111000010111110 0000000000000000 00011110
00D8 0000000000000000 0111000010000000 0000000000000000 00011111
00D9 0000000000000000 1111000000001110 0000000000000010 00010111
00DA 0001000000000000 0111000010011100 0000000000000010 00010111
00DB 0000000001110000 0001100010000000 0000000000000010 10010011
00DC 0000000000000000 1111000000001110 0000000000000010 00010111
00DD 0000000000000111 0111000010011100 0000000000000010 00010111
00DE 0000000011011001 0001100010000000 0000000000000010 10010011
00DF 0000000000011110 0001110010000000 0000000000000000 00011111
00E0 0000000011000000 0000110010000000 0000000000000000 01001111
00E1 0000000011010110 0000110010000000 0000000000000000 01001111
00E2 0000000000000000 1111000000001110 0000000000000010 00010111
00E3 0000100000000000 0111000010011100 0000000000000010 00010111

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 17 of 23)



```

00E4 0000000011101001 0001100010000000 000000000000010 10110011
00E5 0000000001101000 0111000010111110 0000000000000000 00011110
00E6 0000000001100111 0111000010111110 0000000000000000 00011110
00E7 0000000001100001 0111000010111110 0000000000000000 00011110
00E8 0000000000000000 0111000010000000 0000000000000000 00011111
00E9 0000000000000000 1111000000001110 0000000000000010 00010111
00EA 0001000000000000 0111000010011000 0000000000000010 00010111
00EB 0000000011110000 0001100010000000 0000000000000010 10110011
00EC 0000000001100100 0111000010111110 0000000000000000 00011110
00ED 0000000001100111 0111000010111110 0000000000000000 00011110
00EE 0000000001100001 0111000010111110 0000000000000000 00011110
00EF 0000000000000000 0111000010000000 0000000000000000 00011111
00F0 0000000000000000 1111000000001110 0000000000000010 00010111
00F1 1000000000000000 0111000010011000 0000000000000010 00010111
00F2 0000000011110111 0001100010000000 0000000000000010 10010011
00F3 0000000000000000 1111000000001110 0000000000000010 00010111
00F4 0000000000000011 0111000010011000 0000000000000010 00010111
00F5 0000000011110000 0001100010000000 0000000000000010 10010011
00F6 0000000000011110 0001110010000000 0000000000000000 00011111
00F7 0000000001100101 0111000010111110 0000000000000000 00011110
00F8 0000000000000000 0111000010111000 0010000000000000 00011111
00F9 0000000000000000 0111000010111000 0100000000000000 00011111
00FA 0000000000000000 0111000010111000 0110000000000000 00011111
00FB 0000000000000000 0111000010111000 1000000000000000 00011111
00FC 0000000000000000 0111000010111000 1010000000000000 00011111
00FD 0000000000000000 0111000010111000 1100000000000000 00011111
00FE 0000000000000000 0111000010111000 1110000000000000 00011111
00FF 0000000000000000 0111000010111001 0000000000000000 00011111
0100 0000000000000000 0111000010111001 0010000000000000 00011111
0101 0000000011111111 0111000010111110 0000000000000000 00011110
0102 0000000000011000 0001110010000000 0000000000000000 00011111
0103 0000000000000000 1111000000001110 0000000000000010 00011111
0104 0100000000000000 0111000010011000 0000000000000010 00010111
0105 0000000000011000 0001100010000000 0000000000000010 10110011
0106 0000000001100011 0111000010111110 0000000000000000 00011110
0107 0000000011111111 0111000010111110 0000000000000000 00011110
0108 0000000000000000 0111000010000000 0000000000000000 00011111
0109 0000000000000000 1111000110001110 0000001000000010 00010111
010A 0000000000000000 1111000110001110 0000010000000010 00010111
010B 0000000000000000 1111000110001110 0000011000000010 00010111
010C 0000000000000000 1111000110001110 0000100000000010 00010111
010D 1111000011000000 0111000110101010 0011111010000010 00010111
010E 0000000100100000 0001100010000000 0000000000000010 10010011
010F 0000000001100001 0111000010111110 0000000000000000 00011110
0110 0000000000000000 0111000010000000 0000000000000000 00011111
0111 0000000000000000 1111000000001110 0000000000000010 00010111
0112 1000000000000000 0111000010011000 0000000000000010 00010111
0113 0000000100011000 0001100010000000 0000000000000010 10010011
0114 0000000000000000 1111000000001110 0000000000000010 00010111
0115 0000000000000011 0111000010011000 0000000000000010 00010111
0116 0000000100010001 0001100010000000 0000000000000010 10010011
0117 0000000000001110 0001110010000000 0000000000000000 00011111
0118 0000000001100101 0111000010111110 0000000000000000 00011110
0119 0000000000000000 0111000010111000 0010000000000000 00011111
011A 0000000000000000 0111000010111000 0100000000000000 00011111
011B 0000000000000000 0111000010111000 0110000000000000 00011111
011C 0000000000000000 0111000010111000 1000000000000000 00011111

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 18 of 23)

```

011D 0000000001100110 0111000010111110 0000000000000000 00011110
011E 0000000011111111 0111000010111110 0000000000000000 00011110
011F 0000000000011000 00011100:0000000 0000000000000000 00011111
0120 0000000001100010 0111000010111110 0000000000000000 00011110
0121 0000000011111111 0111000010111110 0000000000000000 00011110
0122 0000000000000000 0111000010000000 0000000000000000 00011111
0123 0000000000000000 1111000110001110 000010100000010 00010111
0124 0000000000000000 1111000110001110 0000111000000010 00010111
0125 0000000000000000 1111000110001110 0000111000000010 00010111
0126 0000000001100001 0111000010111110 0000000000000000 00011110
0127 0000000000000000 0111000010000000 0000000000000000 00011111
0128 0000000000000000 1111000000001110 000000000000010 00010111
0129 1000000000000000 0111000010011100 000000000000010 00010111
012A 0000000100101111 0001100010000000 0000000000000010 10010011
012B 0000000000000000 1111000000001110 000000000000010 00010111
012C 0000000000000111 0111000010011100 000000000000010 00010111
012D 0000000100101000 0001100010000000 000000000000010 10010011
012E 000000000011110 0001110010000000 0000000000000000 00011111
012F 000000001100101 0111000010111110 0000000000000000 00011110
0130 0000000000000000 0111000010111000 0010000000000000 00011111
0131 0000000000000000 0111000010111000 0100000000000000 00011111
0132 0000000000000000 0111000010111000 0110000000000000 00011111
0133 0000000000000000 0111000010111000 1000000000000000 00011111
0134 0000000000000000 0111000010111000 1010000000000000 00011111
0135 0000000000000000 0111000010111000 1100000000000000 00011111
0136 0000000000000000 0111000010111000 1110000000000000 00011111
0137 000000001100110 0111000010111110 0000000000000000 00011110
0138 0000000011111111 0111000010111110 0000000000000000 00011110
0139 0000000000011000 0001110010000000 0000000000000000 00011111
013A 0000000000000000 1111000110001110 0001100000000010 00010111
013B 0000000000000000 0111001010110110 0001100000000010 00010111
013C 0000000000000000 0111001010110110 0001100000000010 00010111
013D 0000000000000000 0111001010110110 0001100000000010 00010111
013E 0000000000000011 0111000111001011 1001100000000010 00010111
013F 0000000101000101 0001100010000000 0000000000000010 10110011
0140 0000000000000000 0111000110010110 0001100000000010 00010111
0141 0000000101000111 0001100010000000 0000000000000010 10110011
0142 0000000000000000 0111000110010110 0001100000000010 00010111
0143 0000000101001001 0001100010000000 0000000000000010 10110011
0144 000000000011000 0001110010000000 0000000000000000 00011111
0145 0000000101001111 0000110010000000 0000000000000000 01001111
0146 000000000011000 0001110010000000 0000000000000000 00011111
0147 0000000101100111 0000110010000000 0000000000000000 01001111
0148 000000000011000 0001110010000000 0000000000000000 00011111
0149 0000000101001111 0000110010000000 0000000000000000 01001111
014A 0000000000001010 0111000111011110 0001100000000010 00010111
014B 0000000101100111 0000110010000000 0000000000000000 01001111
014C 0000000000000000 0111000110010110 0001100000000010 00010111
014D 0000000101001011 0001100010000000 0000000000000010 10010011
014E 000000000011000 0001110010000000 0000000000000000 00011111
014F 0000000011000000 0111000111011110 0000001000000010 00010111
0150 0000101010101010 0111000111011110 0000001000000010 00010111
0151 0000110011001100 0111000111011110 00000011000000010 00010111
0152 0000111100001111 0111000111011110 0000100000000010 00010111
0153 0000000000000000 1111000000001110 0000000000000010 00010111
0154 0010000000000000 0111000010011100 0000000000000010 00010111
0155 0000000101011010 0001100010000000 0000000000000010 10010011

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 19 of 23)

```

0156 0000000000000000 1111000000001110 0000000000000010 00010111
0157 0000000000000111 011100001001100 0000000000000010 00010111
0158 0000000101010011 0001100010000000 0000000000000010 10010011
0159 0000000000000000 0101010010000000 0000000000000000 01010111
015A 0000000000000011 0111000010111110 0000000000000000 00011110
015B 0000000000000000 0111000010111000 0000000000000000 00011111
015C 0000000000000000 0111000010111000 0010000000000000 00011111
015D 0000000000000000 0111000010111000 0100000000000000 00011111
015E 0000000000000000 0111000010111000 0110000000000000 00011111
015F 0000000000000000 0111000010111000 1000000000000000 00011111
0160 0000000011111111 0111000010111110 0000000000000000 00011110
0161 0000000000000101 0111000010111110 0000000000000000 00011110
0162 0000000001100001 0111000010111110 0000000000000000 00011110
0163 0000000110101000 0111000111101110 0001100000000010 00010111
0164 0000000000000000 0111000110010110 0001100000000010 00010111
0165 0000000101100100 0001100010000000 0000000000000010 10010011
0166 0000000000000000 0101010010000000 0000000000000000 01010111
0167 0000000110110000 0111000111101110 0000001000000010 00010111
0168 0000100000000000 0111000111101110 0000010000000010 00010111
0169 0000000000000000 0111000110001010 0110011000000010 00010111
016A 0000111111111111 011100011001010 0110011000000010 00010111
016B 0000101100111000 0111000111101110 0000100000000010 00010111
016C 0000000000000000 1111000000001110 0000000000000010 00010111
016D 0010000000000000 011100001001100 0000000000000010 00010111
016E 0000000101110011 0001100010000000 0000000000000010 10010011
016F 0000000000000000 1111000000001110 0000000000000010 00010111
0170 0000000000000111 011100001001100 0000000000000010 00010111
0171 0000000101101100 0001100010000000 0000000000000010 10010011
0172 0000000000000000 0101010010000000 0000000000000000 01010111
0173 0000000000000111 0111000010111110 0000000000000000 00011110
0174 0000000000000000 0111000010111000 0000000000000000 00011110
0175 0000000000000000 0111000010111000 0010000000000000 00011111
0176 0000000000000000 0111000010111000 0100000000000000 00011111
0177 0000000000000000 0111000010111000 0110000000000000 00011111
0178 0000000000000000 0111000010111000 1000000000000000 00011111
0179 0000000000000000 0111000010111000 1000000000000000 00011111
017A 0000000000000000 0111000010111000 1000000000000000 00011111
017B 0000000000000000 0111000010111000 1000000000000000 00011111
017C 0000000011111111 0111000010111110 0000000000000000 00011110
017D 0000000000000101 0111000010111110 0000000000000000 00011110
017E 0000000001100001 0111000010111110 0000000000000000 00011110
017F 0000000000000000 0101010010000000 0000000000000000 01010111
0180 0000000000000100 0111000010111110 0000000000000000 00011110
0181 0000000011111111 0111000010111110 0000000000000000 00011110
0182 0000000001100001 0111000010111110 0000000000000000 00011110
0183 0000000000000000 1111000110001110 0000001000000010 00010111
0184 0000000000000000 1111000000001110 0000000000000010 00010111
0185 0001000000000000 011100001001100 0000000000000010 00010111
0186 0000000110101000 0001100010000000 0000000000000010 10110011
0187 0000000000000100 0111000010111110 0000000000000000 00011110
0188 0000000011111111 0111000010111110 0000000000000000 00011110
0189 0000000001100001 0111000010111110 0000000000000000 00011110
018A 0000000000000000 1111000110001110 0000010000000010 00010111
018B 0000000000000000 1111000000001110 0000000000000010 00010111
018C 0001000000000000 011100001001100 0000000000000010 00010111
018D 0000000110101001 0001100010000000 0000000000000010 10110011
018E 0000000000000100 0111000010111110 0000000000000000 00011110

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 20 of 23)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

```

018F 0000000011111111 0111000010111110 0000000000000000 00011110
0190 000000001100001 0111000010111110 0000000000000000 00011110
0191 0000000000000000 1111000110001110 0000011000000010 00010111
0192 0000000000000000 1111000000001110 0000000000000010 00010111
0193 0001000000000000 011100001001100 0000000000000010 00010111
0194 0000000110101010 0001100010000000 0000000000000010 10110011
0195 0000000000000100 0111000010111110 0000000000000000 00011110
0196 0000000011111111 0111000010111110 0000000000000000 00011110
0197 000000001100001 0111000010111110 0000000000000000 00011110
0198 0000000000000000 1111000110001110 0000100000000010 00010111
0199 0000000000000000 1111000000001110 0000000000000010 00010111
019A 0001000000000000 011100001001100 0000000000000010 00010111
019B 0000000110101011 0001100010000000 0000000000000010 10110011
019C 0000000000000100 0111000010111110 0000000000000000 00011110
019D 0000000011111111 0111000010111110 0000000000000000 00011110
019E 0000000001100001 0111000010111110 0000000000000000 00011110
019F 0000000000000000 1111000110001110 0000101000000010 00010111
01A0 0000000000000000 1111000000001110 0000000000000010 00010111
01A1 0001000000000000 011100001001100 0000000000000010 00010111
01A2 0000000110101100 0001100010000000 0000000000000010 10110011
01A3 0000000000000100 0111000010111110 0000000000000000 00011110
01A4 0000000011111111 0111000010111110 0000000000000000 00011110
01A5 0000000001100001 0111000010111110 0000000000000000 00011110
01A6 0000000000000000 1111000110001110 0000110000000010 00010111
01A7 0000000000000000 0101010010000000 0000000000000000 01010111
01A8 1010101010101010 0111000111011110 0000010000000010 00010111
01A9 1010101010101010 0111000111011110 0000011000000010 00010111
01AA 1010101010101010 0111000111011110 0000100000000010 00010111
01AB 1010101010101010 0111000111011110 0000101000000010 00010111
01AC 1010101010101010 0111000111011110 0000110000000010 00010111
01AD 0000000000000000 0101010010000000 0000000000000000 01010111
01AE 0000000000000000 1111000000001110 0000000000000010 00010111
01AF 0001000000000000 011100001001100 0000000000000010 00010111
01B0 0000000111000111 0001100010000000 0000000000000010 10110011
01B1 0000000000000100 0111000010111110 0000000000000000 00011110
01B2 0000000011111111 0111000010111110 0000000000000000 00011110
01B3 0000000001100001 0111000010111110 0000000000000000 00011110
01B4 0000000000000000 1111000110001110 0000111000000010 00010111
01B5 0000000000000000 1111000000001110 0000000000000010 00010111
01B6 0001000000000000 011100001001100 0000000000000010 00010111
01B7 0000000111001000 0001100010000000 0000000000000010 10110011
01B8 0000000000000100 0111000010111110 0000000000000000 00011110
01B9 0000000011111111 0111000010111110 0000000000000000 00011110
01BA 0000000001100001 0111000010111110 0000000000000000 00011110
01BB 0000000000000000 1111000110001110 0001000000000010 00010111
01BC 0000000000000000 1111000000001110 0000000000000010 00010111
01BD 0001000000000000 011100001001100 0000000000000010 00010111
01BE 0000000111001001 0001100010000000 0000000000000010 10110011
01BF 0000000000000100 0111000010111110 0000000000000000 00011110
01C0 0000000011111111 0111000010111110 0000000000000000 00011110
01C1 0000000001100001 0111000010111110 0000000000000000 00011110
01C2 0000000000000000 1111000110001110 0001001000000010 00010111
01C3 0000000000000000 1111000000001110 0000000000000010 00010111
01C4 0001000000000000 011100001001100 0000000000000010 00010111
01C5 0000000111001010 0001100010000000 0000000000000010 10110011
01C6 0000000000000000 0101010010000000 0000000000000000 01010111
01C7 1010101010101010 0111000111011110 0000111000000010 00010111

```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 21 of 23)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

```
01C8 10101010101010 0111000111101110 0001000000000010 00010111
01C9 10101010101010 0111000111101110 0001001000000010 00010111
01CA 0000000000000000 0101010010000000 0000000000000000 01010111
01CB 0000000001100000 0111000010111110 0000000000000000 00011110
01CC 0000000001100111 0111000010111110 0000000000000000 00011110
01CD 0000000000000000 0000000010000000 0000000000000000 00011111
```

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 22 of 23)

AMDOS/29 AMDASM MICRO ASSEMBLER, V2.0

SYMBOLS

AB	0001	ABORT	00FF	ADCIDATA	0007	ADCIDLE	0005
ADCIREAD	0004	ADCISEND	0005	ADCBUSY	2000	ADCDEBUG	013A
ADCDIAG	0065	ADCDIAG1	0074	ADDRCMD	FF40	AQ	0000
ARORRPT	0089	ARORRPT1	008D	ARTQCID	F920	ASR9RID	F1B0
ASR9STID	F0C0	BEGIN	0018	CD2RID	F1B2	CD2STID	F0C2
CIONE	0001	CIZERO	0000	CJP	0003	CJPP	0008
CJS	0001	CJV	0006	COND	0000	CONT	000E
COUNT	000A	CRCERROR	0800	CRTN	000A	CRTQCID	F922
CTRLEN	0000	DA	0005	DECMC	0066	DELAY1	0080
DELAY2	0100	DIAG	005D	DIAGADC	1000	DQ	0006
DUMP4	0118	DUMPE	0080	DUMP7	012F	DUMPS	00F7
DZ	0007	EXNOR	0007	EXOR	0006	FADD	0000
FAND	0004	FAULT	01C8	FCSFAULT	0068	FIFOFULL	0064
FILLER	AAAA	FLAG	007E	FOR	0003	GENRPT	0167
GENSTAT	014F	GOMASK	0003	HDRMSK	00FF	INIT1	0000
INIT2	0004	INIT3	000C	ITOUSR	0010	JMAP	0002
JRP	0007	JSRP	0005	JZ	0000	LOCT	000C
LOADRAM	0065	LONG	0034	LOOP	000D	LOOPA	001A
LOOPB	0050	LOOPC	003D	LOOPD	0082	LOOPE	0093
LOOPF	00A9	LOOPG	00C5	LOOPH	0009	LOOPI	00F0
LOOPJ	0111	LOOPK	0128	LOOPL	0148	LOOPM	0153
LOOPN	0164	LOOPO	016C	LOOPQ	00E9	LOOPR	00A2
LOOPPT1	0010	LOOPPT2	0016	LOOPPT3	0068	LOOPPT4	0071
MACROEN	2000	MICROEN	0000	MSGFAULT	0060	MSGMASK	4000
MSRTOUSR	0002	NOCTRL	0001	NOMACRO	0001	NOMICRO	0001
NONEON	0003	NOOP	0000	NORM	001E	NORMASK	0007
NOTRS	0005	NP	0001	ONE	0001	PASS1	01A8
PASS2	01A9	PASS3	01AA	PASS4	01AB	PASS5	01AC
PASS6	01C7	PASS7	01C8	PASS8	01C3	PASS9	01CA
PEMASK	8000	PUSH	0004	QREG	0000	R0	0000
R1	0001	R10	000A	R11	0008	R12	000C
R13	000D	R14	000E	R15	000F	R2	0002
R3	0003	R4	0004	R5	0005	R6	0006
R7	0007	R8	0008	R9	0009	RAMA	0002
RAMD	0005	RAMF	0003	RAMQD	0004	RAMQU	0006
RAMU	0007	RDADC3	01AE	RDADC6	0180	READ3	0062
READ4	0063	READ6	009A	READ9	00E0	RFTCT	0008
RPCT	0009	RPTOUT	0173	RPTWD1	0180	RPTWD2	0800
RPTWD4	0838	RRGATN	0067	RRGSTAT	0061	SELA	0000
SEL8	0001	SENORPT	00CC	SENDSTAT	0089	SHFTRA	0000
SHIFT	0002	SHORT	0040	STATOUT	015A	STATWD1	00C0
STATWD2	0AAA	STATWD3	0CCC	STATWD4	0F0F	SUBR	0001
SUBS	0002	TEST	0001	TIMER1	01A8	TWB	000F
TXMIX	0149	TXRPT	0147	TXSTAT	0145	UC	001B
UN	001F	UNC	001A	UNCOMO	0001	UNN	001E
UNOVR	001E	UNZ	0014	UOVR	0017	UPRCDIAG	0103
URORRPT	0120	USRTOMSR	0002	UZ	0015	ZA	0004
Z8	0003	ZERO	0000	ZQ	0002		

TOTAL PHASE 2 ERRORS = 0

FIGURE B.3.3-2. RRG MICROCODE PROGRAM LISTING (Sheet 23 of 23)

If the ADC Validation is selected, the RRC proceeds to sequence known messages provided by the CPU to the ADC. (See pages 4 through 8, title "ADCCP DATA CONVERTER DIAGNOSTIC.") As the ADC outputs the message, the data is reassembled and returned to the RRC, which in turn returns the data to the CPU. In this mode, the data paths between the RRC and the ADC are verified. Also, the hardware functions of the ADC are verified. Again, two conditions must be met before a reply is transferred: (1) a message must be available in the RRG Buffer Interface, and (2) the ADC must be ready to accept the message. If the RRC fails to receive all of the data expected, it will substitute an all "A" data pattern in the remaining portion of the message before sending it back to the CPU. In this way, it can be determined that data was lost through the ADC.

The ADC Diagnostic is used as a debug tool for the ADC digital logic. (See pages 10 and 11, titled "ADC DEBUG ROUTINE.") It has four debug modes: (1) load canned status messages into the ADC for continuous message transmission, (2) load canned search target messages into the ADC for continuous message transmission, (3) load canned search target and status messages into the ADC, at a 10:1 ratio, respectively, for continuous message transmission, and (4) load no messages into the ADC for continuous idle frame transmission. The CPU selects the option by setting the appropriate value in the 2-bit option field of the RRG command. In the case of search target messages, the contents of word No. 3 is incremented by 1 each time the message is loaded into the ADC. In this way, zero bit insertion and ADCCP parity generation can be fully checked.

APPENDIX C

UPLINK RECEIVER PAM/DPSK ALIGNMENT



The Uplink Receiver detects and decodes Mode S interrogations from the Mode S sensor at the radio frequency (RF) level, and from the Self Test Unit (STU) at the RF and the digital level. The interrogations received at the digital level are the modulation control pulses generated by the STU. These control pulses are coupled directly to the Uplink Receiver from the STU Transmitter, bypassing the analog hardware in between. A fourth input comes from the Uplink Receiver itself.

Basically, an alignment is required to synchronize the pulse amplitude modulation (PAM) stream with the differential phase shift keying (DPSK) data stream. The PAM stream carries the P1, P2, and P6 pulses. The DPSK data stream carries the Mode S message as specified in the Mode S National Standard (Order No. 6365.1A). Within the DPSK data stream, a synchronization pulse (referred to as the "Phase Sync Reversal" pulse) is generated 1.25 microsecond ( $\mu$ s) into the P6 pulse. It is used to synchronize the receiver logic to the incoming data. Therefore, this timing must be maintained in the Uplink Receiver to insure the proper detection of a Mode S message.

The RF alignment of the sensor's PAM and DPSK data streams must be done first. This is necessary since the Mode S sensor alignment will effect the RF alignment of the STU's PAM and DPSK data streams. Alignment of the STU's digital paths could be performed at any time since it has no effect on the RF timing paths, and vice versa. The Uplink Receiver alignment procedures are given below.

#### Uplink Receiver RF Alignment for Mode S Sensor Interrogations:

- a. Remove Uplink Receiver board from ARIES digital chassis, Slot 11.
- b. Remove any wire feeding the PAM data from the delay module 27E to input pins 26S-1 and 26S-2 of the shift register, and to pin 27F-9 from any output pin from shift register 26S and 25S. (Refer to ARIES:Uplink Receiver logic schematic No. 3.)
- c. Jump pin 27E-1 to pin 26S-1, and jump pin 26S-13 to pin 27F-9.
- d. Place the board on an extender and insert it back in its original slot.
- e. Set up a dual channel oscilloscope to monitor the following two signals: (1) the Sync Phase Reversal window "SPRDET-G" on pin 18F-4, and (2) the DPSK lead edge detected pulse train "DPSK-LE" on pin 18F-5. (Refer to ARIES:Uplink Receiver logic schematic No. 5.) Trigger on the rising edge of the Sync Phase Reversal window.
- f. Load the complete maintenance software package into the ARIES and the STU. Access diagnostic No. 1 of the Uplink Receiver. Send a reset command (Hex 00E0) followed with a select RF interrogation/enable interrupt command (Hex 0040).
- g. Set up the Mode S sensor to output Mode S roll-call interrogations at a steady rate, then start the sensor.
- h. While monitoring the two channels of the oscilloscope, stabilize the DPSK-LE pulse train with respect to the SPRDET-G window to no more than a 2-clock wide jitter by moving the jumper on pin 27E-1 along the digital delay line 27E (Belfure 0447-0100-00). Then adjust the first generated DPSK-LE pulse in the center of the window SPRDET-G by moving the jumper 26S-13 along the shift register 26S and 25S.

i. Set up the oscilloscope to monitor the status bit ST101 on pin 18K-9 (ARIES:Uplink Receiver digital schematic No. 6). Trigger on the lead edge of this signal. If the two paths are properly aligned, the oscilloscope will not be triggered and the sensor RF input alignment is complete. If the oscilloscope is being triggered, the Sync Phase Reversal lead edge pulses are not falling completely within the detection window causing the Mode S detection sequence to abort. Repeat steps "e" through "i." (Note the error interrupt counter for the Uplink Receiver can be read via the diagnostic DIAG#1 to determine how often the Mode S detection sequence is being aborted.)

#### Uplink Receiver RF Alignment Procedure for STU Interrogations:

- a. Remove Uplink Receiver board from ARIES digital chassis, Slot 11.
- b. Remove any wire on pin 27T-1 going to shift register 27S, and on pin 27R-12 going to delay line 27T. (Refer to ARIES:Uplink Receiver logic schematic No. 3.)
- c. Jump pin 27S-1 to pin 27T-1, and jump pin 27T-8 to pin 27R-12.
- d. Place the board on an extender and insert it back into its original slot.
- e. Set up a dual channel oscilloscope to monitor the following two signals: (1) the Sync Phase Reversal window "SPRDET-G" on pin 18F-4, and (2) the DPSK lead edge detected pulse train "DPSK-LE" on pin 18F-5. (Refer to ARIES:Uplink Receiver logic schematic No. 5.) Trigger on the rising edge of the Sync Phase Reversal window.
- f. Load the complete maintenance software package into the ARIES and the STU. Access diagnostic No. 1 of the Uplink Receiver. Send a reset command (Hex 00E0) followed with a select RF interrogation/enable interrupt command (Hex 0040).
- g. Set up the STU Transmitter to output a Mode S short roll-call interrogation at a steady rate, using STU Transmitter Diagnostic No. 2. (Refer to ARIES Hardware Maintenance Manual, Volume I, section 7.10.2.2 for detailed information on the STU Transmitter diagnostic No. 2.)
- h. While monitoring the two channels of the oscilloscope, adjust the first generated DPSK-LE pulse in the center of the window SPRDET-G by moving the jumper 27S-1 along the shift register 27S. Then move the jumper on 27T-8 along the digital delay line 27T to stabilize the DPSK-LE pulse jitter to within two clock periods.
- i. Set up the oscilloscope to monitor the status bit ST101 on pin 18K-9 (ARIES:Uplink Receiver digital schematic No. 6). Trigger on the lead edge of this signal. If the two paths are properly aligned, the oscilloscope will not be triggered and the STU Transmitter RF input alignment is complete. If the oscilloscope is being triggered, the Sync Phase Reversal lead edge pulses are not falling completely within the detection window causing the Mode S detection sequence to abort. Repeat steps "e" through "i." (Note the error interrupt counter for the Uplink Receiver can be read via the diagnostic DIAG#1 to determine how often the Mode S detection sequence is being aborted.)

Uplink Receiver Digital Alignment Procedure for STU Interrogations:

- a. Remove Uplink Receiver board from ARIES digital chassis, Slot 11.
- b. Remove any wire on pin 26R-5 going to delay line 26E. (Refer to ARIES:Uplink Receiver logic schematic No. 3.)
- c. Jump pin 26R-5 to pin 26E-1.
- d. Place the board on an extender and insert it back in its original slot.
- e. Set up a dual channel oscilloscope to monitor the following two signals: (1) the Sync Phase Reversal window "SPRDET-G" on pin 18F-4, and (2) the DPSK lead edge detected pulse train "DPSK-LE" on pin 18F-5. (Refer to ARIES:Uplink Receiver logic schematic No. 5.) Trigger on the rising edge of the Sync Phase Reversal window.
- f. Load the complete maintenance software package into the ARIES and the STU. Access diagnostic No. 1 of the Uplink Receiver. Send a reset command (Hex 00E0) followed with a select digital input/enable interrupt command (Hex 0140).
- g. Set up the STU Transmitter to output a Mode S short roll-call interrogation at a steady rate, using STU Transmitter Diagnostic No. 2. (Refer to ARIES Hardware Maintenance Manual, Volume I, section 7.10.2.2 for detailed information on the STU Transmitter diagnostic No. 2.)
- h. While monitoring the two channels of the oscilloscope, adjust the first generated DPSK-LE pulse in the center of the window SPRDET-G and stabilized to within a 2-clock jitter by moving the jumper 26E-1 along the delay line 26E.
- i. Set up the oscilloscope to monitor the status bit ST101 on pin 18K-9 (ARIES:Uplink Receiver digital schematic No. 6). Trigger on the lead edge of this signal. If the two paths are properly aligned, the oscilloscope will not be triggered and the STU Transmitter digital input alignment is complete. If the oscilloscope is being triggered, the Sync Phase Reversal lead edge pulses are not falling completely within the detection window causing the Mode S detection sequence to abort. Repeat steps "e" through "i." Note the error interrupt counter for the Uplink Receiver can be read via the diagnostic DIAG#1 to determine how often the Mode S detection sequence is being aborted.