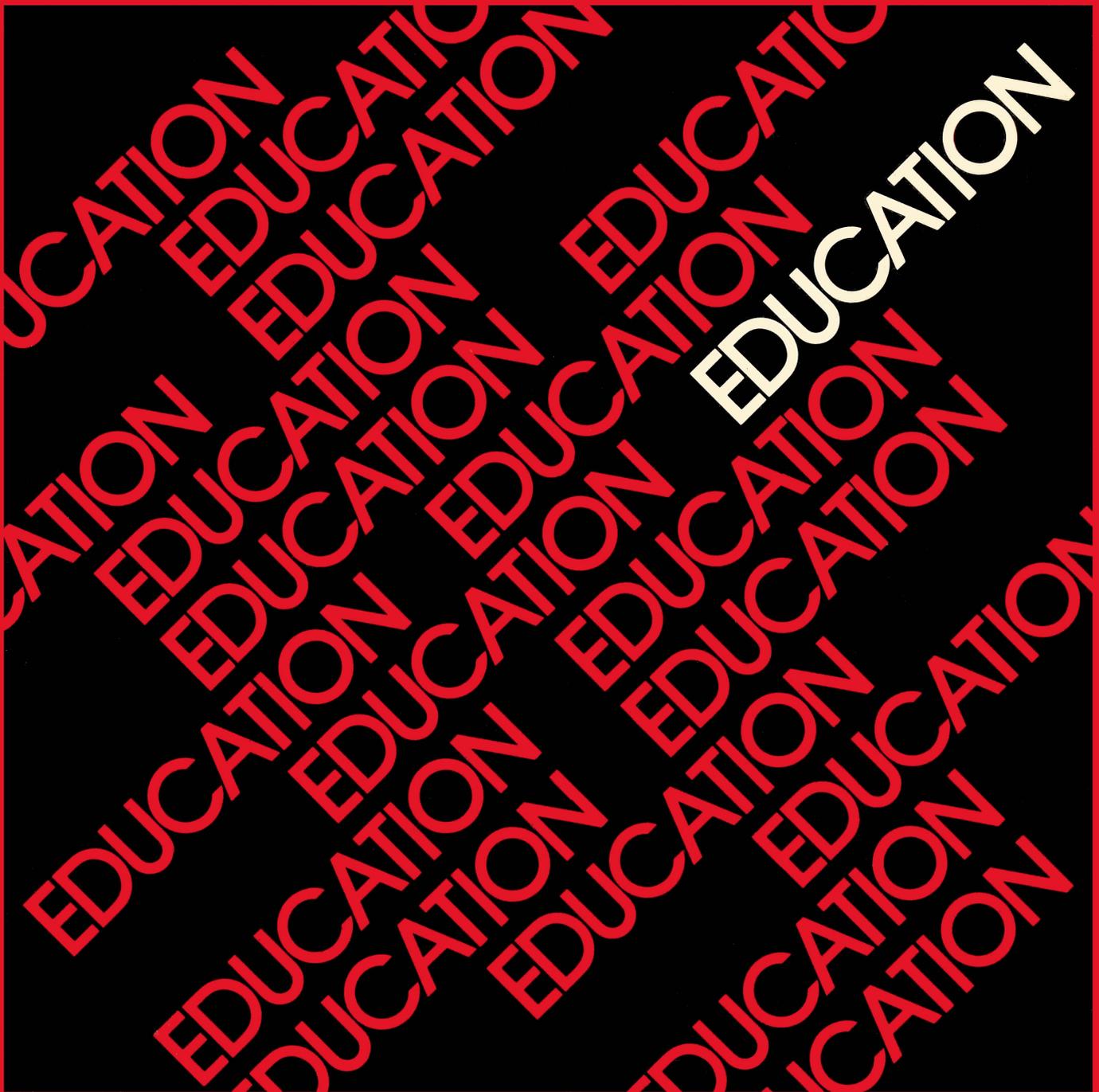


digital

11 MAINFRAME

PDP 11/40





11 MAINFRAME

PDP 11/40



KD11-A-000.000

LEARNING ACTIVITY UNIT

KD11-A INTRODUCTION AND OVERVIEW

DECEMBER 18, 1973



RATIONALE

Welcome to the wonderful world of the PDP-11/40. You have now completed your study of the PDP-11/05 and you should have a solid foundation of PDP-11 principles which you will use to develop your understanding of the 11/40. Don't feel that you are going on to something completely different and that you can therefore forget about the 05. The 40 is really just an extension of the 05 in that many of the concepts are similar and many of the basic operations are done in almost an identical fashion. Following are some of the features of the 40 that you will find to be similar to what you already know about the 05.

1. The operation of functions internal to the C.P. are controlled by a Read Only Memory (ROM).
2. The Bus Control is handled in a similar fashion, which must be so since all of the controlling signals on the PDP-11 Unibus are identical regardless of the C.P. being used.
3. The C.P. contains a set of 16 registers which are used by the programmer (R0 - R7) and for handling some of the internal operations (R10 - R17).
4. The instruction set is identical to that used on the 05 with the exception of a few additional instructions.
5. Console operations are identical with the addition of a few more indicators. So you can see that you already have a lot of information which is relevant to the 11/40.

During this unit you will be having what essentially amounts to a discussion period among you, your fellow students and your instructor. The instructor will attempt to point out all of the similarities and differences between the 11/05 and the 11/40. He will also discuss the specifications and applications of 11/40's. He will summarize the



the objectives of the 11/40 course and outline for you what you will be doing for the next four days or so. Please feel free to debate points with your instructor and among yourselves under the guidance of the instructor.

In some ways this is the most important lesson on the 40 since it will help you to tie together facts which you have learned about the 05 with information you are finding out about the 40. In addition this lesson should help you to make the transition from the 05 course to the 40 course and start you on your way to becoming PDP-11/40 giants.



PREREQUISITES

Successful completion of the PDP-11/05 portion of the course.

OBJECTIVES

To demonstrate knowledge of some of the major characteristics of the 11/40 by correctly answering five out of the six questions given in the post test.

LEARNING ACTIVITIES

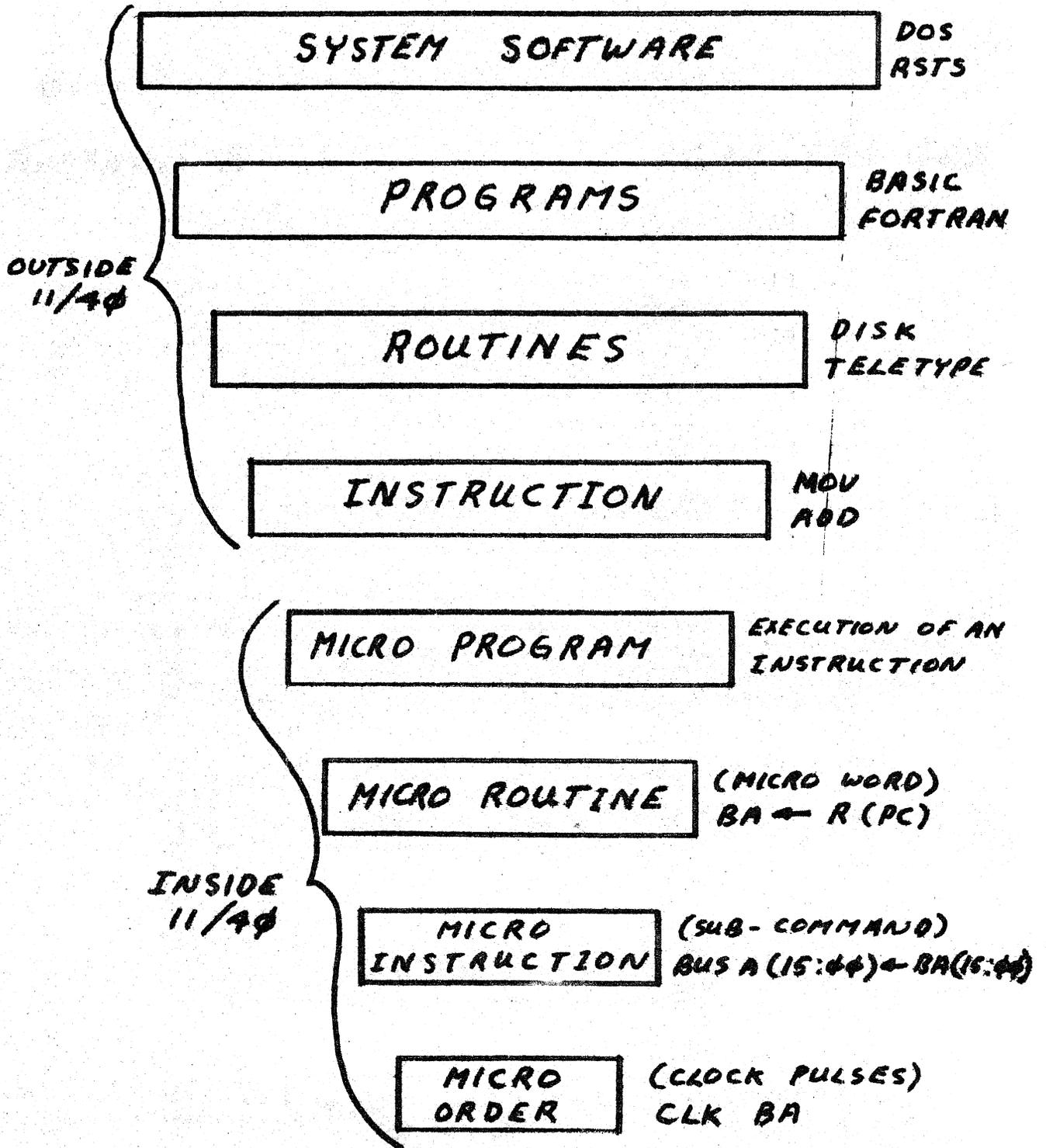
- I. Attend the 90 minute discussion period. Please ask any questions you have concerning 11/40 specifications, applications, systems concepts, configurations, etc. Be sure to take note of references made to figures, drawings, tables, etc. during the discussion.
  
- II. Attempt the post test. You have 20 minutes.

APPENDIX A

The following list is a point outline of the 11/40 course. Each point is a two hour LAU.

1. KD11A Introduction and Overview (this LAU).
2. Block Diagram: multiplexors and data paths.
3. Block Diagram: control section.
4. Block Diagram: clocking, Unibus control, branching.
5. Flow Diagram: introduction, overview, instruction format.
6. Flow Diagram: Jamupp to console, console.
7. Flow Diagram: single operand instructions.
8. Flow Diagram: double operand instructions, overlap.
9. Flow Diagram: traps/service.
10. Lab Project: maintenance module usage
11. Logic Diagrams: conventions, special circuits, layout
12. Logic Diagrams: data paths, word, IR decode.
13. Logic Diagrams: timing, Jamupp.
14. Logic Diagrams: status, bus control
15. Lab Project: scoping clock, Unibus.
16. Power System: block diagram, distribution.

SYSTEM HIERARCHY



APPENDIX C

## THE MARK INSTRUCTION

The MARK is an instruction that facilitates stack cleanup procedures when returning from a subroutine. Stack cleanup simply means resetting Stack Pointer R6 to the value it contained just prior to jumping to the subroutine. It is unique in that the instruction itself is executed off the stack.

## PROGRAMMING OPERATION

The routine shown below illustrates the operation of the MARK and may be used by the technician to check for proper operation.

Given R5 = 1234, R6 = 2000

5000/MOV R5, -(6)	; (1776) 1234, old R5
2/MOV #20, -(6)	; (1774) 20, P1
4/#20	
6/MOV #30, -(6)	; (1772) 30, P2
10/#30	
12/MOV #40, -(6)	; (1770) 40, P3
14/#40	
16/MOV #MARK3, -(6)	; (1766) 006403, MARK3 instr.
20/#MARK3	
22/MOV R6, R5	{ (5) 1766, content of R5 = adrs of MARK 3 instr ;dst → R[TEMP], (TEMP) 6000 ;R[SP] - 2 → R[SP], (6) 1764 ;R[SF] → @R[SP], (1764) 5030 ;R[PC] → R[SF], (7) 5030 ;R[TEMP] → R[PC], (7) 6000
24/JSR PC, 6000	
26/index	
5030/HALT	
6000/HALT	{ R[DF] → R[PC], (7) 1766: adrs of MARK3 on stack ;@R[SP] → R[DF], (5) 5030: return PC ;R[SP] + 2 → R[SP], (6) 1766
6002/RTS5	

Load address to 5000 and depress START. The instruction in 5000 saves the contents of R5 (325) on the stack in location 1776. The next three MOV instructions in 5002, 5006, and 5012 push three parameters onto the stack. Then the instruction in location 5016 pushes the MARK instruction (6403) onto the stack into location 1766. At this point, R6 contains 1766, which is transferred into R5 when the MOV instruction in location 5022 is executed.

Finally, the JSR is executed, which saves the return PC on the stack in location 1764 and causes program control to transfer to location 6000 where the machine halts.

After the machine halts, the registers and stack appear as shown below:

R 7 = 6002 ;address of next instr. in subtn.  
R 5 = 1766 ;address of the MARK instr.  
R 6 = 1764 ;stack

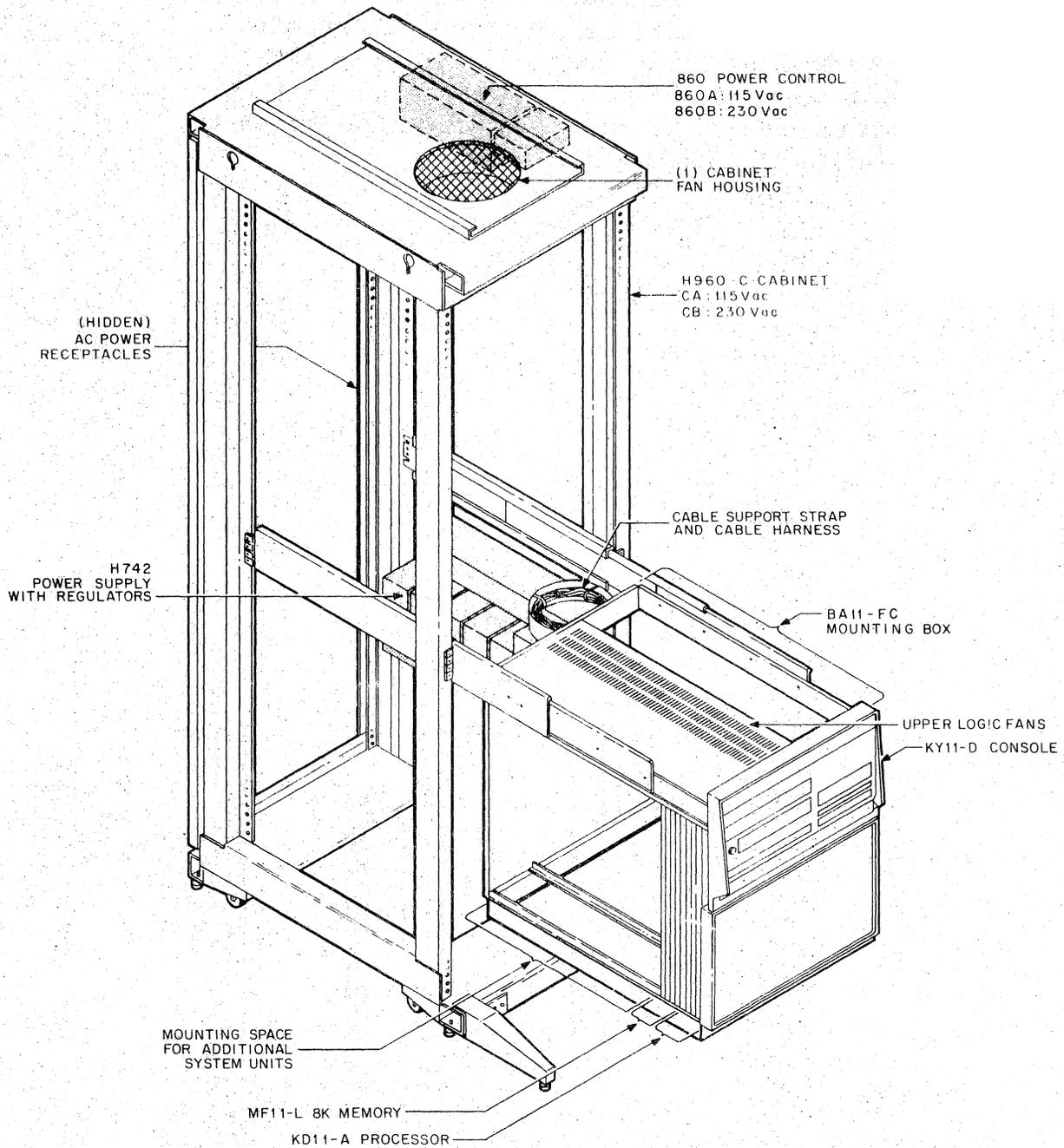
1764	return PC = 5030	result of instruction at 5024
1766	MARK3 006403	result of instruction at 5016
1770	P3 = 40	result of instruction at 5012
1772	P2 = 30	result of instruction at 5006
1774	P1 = 20	result of instruction at 5002
original R6= 2000		

Normally at this point the subroutine would use the SP to retrieve the parameters for its execution.

Now press CONT. This will cause execution of the RTS instruction in 6002. The RTS will transfer the contents of R5(1766) into the PC and pop 5030 off the stack into

R5. Since the PC contains 1766, the next instruction executed will be the MARK instruction. After fetching the MARK, the PC gets updated to 1770, then the SP gets the contents of the PC (1770) plus two times the number of arguments (6) which is 1776. The PC gets the contents of R5 (5030) therefore pointing to the next instruction after the JSR. Then 1234 is popped off the stack and put into R5 which restores the C.P. to the correct condition to continue on in the main program.

APPENDIX D-1



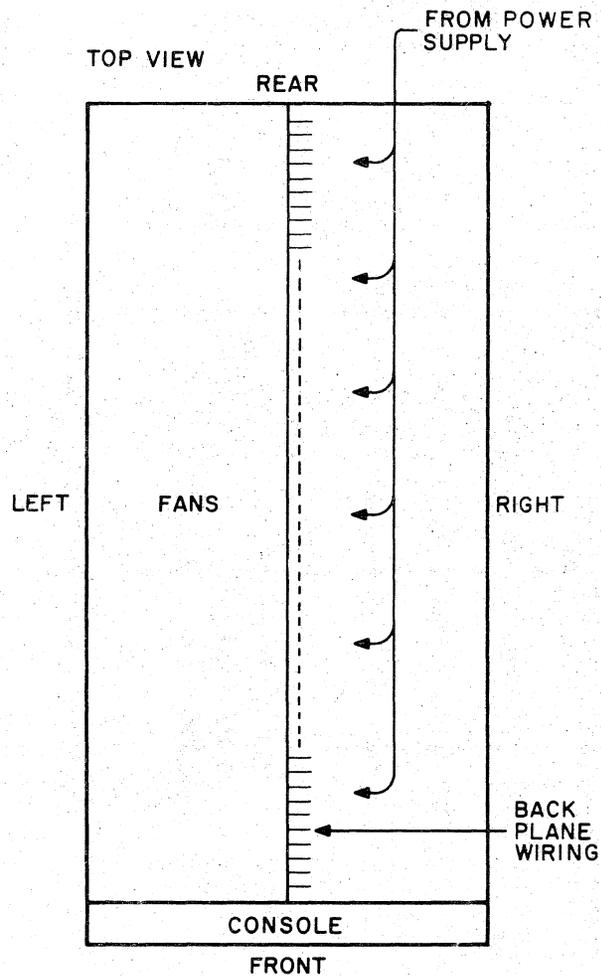
APPENDIX D-2

KD11-A MODULES

55-09701	KY11-D Console, hex, no module pins, two back side connectors to connect to K1 & K5 modules via cables	
M981	Unibus cable connection with terminator, double height	
M7231	K1, Data Path, hex, one back side connector to console	
M7232	K2, U Word, quad. three back side connectors to M7238 (EIS)	
M7233	K3, IR Decode, hex	
M7234	K4, Timing, hex, clock adjustment	
M7235	K5, Status, hex, one back side connector to console	
*M7236	KT11-D, Memory Management, hex	
*M7237	KJ11-A, Stack Limit Register, single height	
*M7238	KE11-E, EIS option, hex, three back side connectors to M7232	
*M7239	KE11-F, FIS option, quad	
*M787	KW11-L Line Clock option, single height	
*W130, W131	KM11, Maintenance module set	1) Processor 2) Options
Small Peripheral	1) KL11: M105 single M7820 single M780 double	2) DL11 M7800 quad

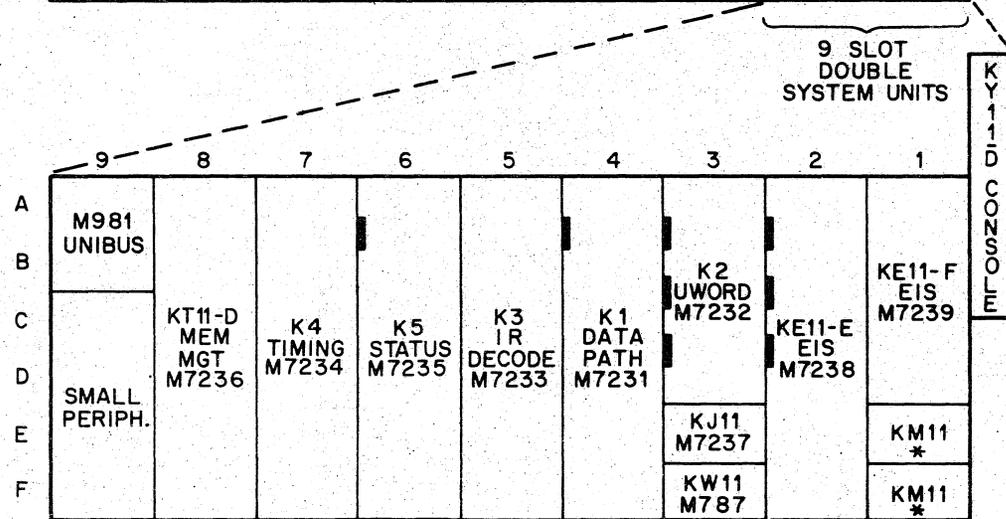
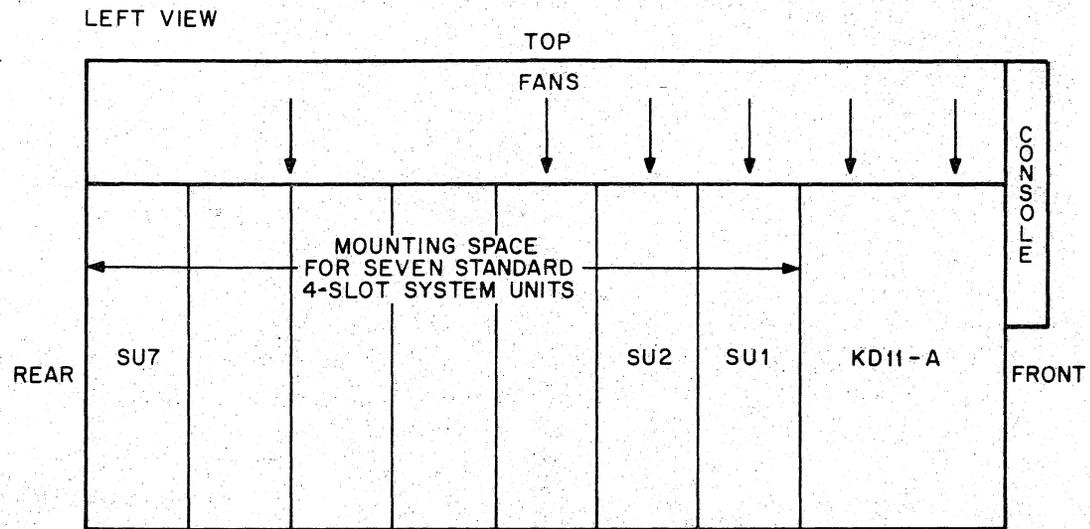
---

\*Pre-wired processor options-not included as part of the basic processor.



CONNECTORS ON BACK EDGE OF MODULE

\* MAINT. MODULE SLOTS  
 EI EIS/FIS/KT11-D  
 FI BASIC PROCESSOR



KD11-A MODULE LAYOUT (AS VIEWED FROM LEFT SIDE)

APPENDIX E

## LIST OF SOME 11/40 APPLICATIONS

- Communications: Message Switching, Data Concentration, Remote Terminals, System "Front Ends"
- Engineering: Calculation, Design and Drafting, Simulation, Graphics
- Computation: Scientific Calculation, In-house Time-sharing, Timesharing Utilities, Hybrid Systems (A/D)
- Business: Inventory Control, General Accounting, Payroll, Reservations, Text Editing, Data Storage and Retrieval, Order Entry, Source Data Collection, Key-to-Tape
- Education: Computer-aided Instruction, Programming Courses, Computer Science Courses, Problem Solving, Engineering Courses, Administrative Processing
- Industry: Data Acquisition and Control, Data Monitoring and Logging, Machine Control, Numerical Control, Testing, Process Control, Typesetting, Traffic Control, Electric Utility Monitoring and Control
- Research: Oceanographic Studies, Gas Chromatography/Mass Spectroscopy, Signal Processing

APPENDIX F

LIST OF 11/40 INTERNAL OPTIONS

KE11-E, Extended Instruction Set

KE11-F, Floating Instruction Set

KT11-D, Memory Management

KJ11-A, Stack Limit Register

KW11-L, Line Frequency Clock

APPENDIX G

## LIST OF THE BASIC 11/40 MAINDECS

CPU

T1 Branch  
 T2 Conditional Branch  
 T3 Unary  
 T4 Binary  
 T5 Rotate/Shift  
 T6 CMP Equality  
 T7 CMP Non-equality  
 T8 Move  
 T9 Bit Set Clear Test  
 T10 ADD  
 T11 Subtract  
 T12 Jump  
 T13 JSR, RTS, RTI  
 T14 40 Traps Test  
 T15 11 Family Instruction Exerciser  
 T17 System Exerciser  
 Power Fail  
 General Test Program

Memory

Basic Address Test Up  
 Basic Address Test Down  
 No Dual Address Test  
 Basic Test Patterns  
 Moving Ones and Zeros  
 One's Susceptibility  
 Worst Case Noise  
 Core Heating  
 Random Data  
 8K Special  
 Up-Down Address Test  
 Memory I/O Exerciser  
 0-124K Exerciser



KD11A - 010.000

LEARNING ACTIVITY UNIT

KD11A PROCESSOR

BLOCK DIAGRAM

DATA PATHS

1/9/73



RATIONALE

Previously, you have been introduced to the 11/40 and have studied it from a physical point of view. You have seen the mechanics of the mounting box, the cabinet, module locations, etc. You have also studied the 11/05 processor and have seen how a C.P. can be depicted in a block diagram format.

Now you are going to be introduced to the KD11A Block Diagram. You will observe how the C.P. can be broken into three basic sections of which, in this unit, we are primarily interested in the Data Path section. You will learn the name and function of each of the major components in this section and how data is manipulated for various operations.

The skills you learn in this unit will prove invaluable in future LAU's, since you will be doing similar analysis in studying the other portions of the block diagram. More importantly, you will find the block diagram to be indispensable as a trouble shooting aid. For example, once you have properly analyzed the symptoms of a problem, you will be able to narrow its location down to one of the major components within that section if you have a good understanding of the block diagram.



PREREQUISITES

Satisfactory performance to this unit.

OBJECTIVES

Using the list of Data Manipulation Functions in Appendix B and any other reference material, write the sets of functions required to implement a given instruction.

LEARNING ACTIVITIES

## I. LECTURE

Attend the twenty minute lecture. Pay close attention, and jot down notes on the block diagram itself. Hold questions until the end of the lecture.

## II. PROJECTS AND PROBLEMS

## A.

1. Do the problems and answer all the questions. You will find the correct answers in Appendix A.
2. Refer to the KD11A Processor Maintenance Manual, Table 3-3 on p. 3-23 to p. 3-34.
3. All page references made in this LAU are to the above mentioned manual unless specifically stated otherwise.
4. As you go through the following projects, make pertinent notes on your block diagram. These should prove to be invaluable trouble shooting aids in the lab and in the field.

## B. Locate the Arithmetic Logic Unit (ALU) approximately in the center of the page.

1. Observe that there are five control lines going into the ALU. These will cause the ALU to execute one of thirty-two functions which are shown in Table 3-2 on p. 3-17. Study this table and analyze the various functions.
2. Make special note of the function specified by the \*. What two instructions would make use of this operation? Mark this on your block diagram since this is the only place in the machine where this operation occurs.

- C. Locate the D Reg and the D Mux just below the ALU.
1. Specification of which input is to be gated through the D Mux is done by the \_\_\_\_\_ and \_\_\_\_\_ signals.
  2. Analyze the function of the box labeled D (C) .
  3. Refer to Table 3-1 on p. 3-14. Which of the following combinations of signals would allow an instruction to be read in from core, gated through the D Mux and presented to the I Reg?
    - (a) SDM1 (0) \* SDM0 (0)
    - (b) SDM1 (0) \* SDM0 (1)
    - (c) SDM1 (1) \* SDM0 (0)
    - (d) SDM1 (1) \* SDM0 (1)

Check your answer. If you were correct, go on to section D. If not, read the following description.

We are fetching an instruction. We have done the bus cycle and the data (which is the instruction) is on the Unibus data lines. We now want to take this data and clock it into the IReg. The data comes from BUS D (15:00) through receivers and into the D MUX via the left-most input on our diagrams. We then must select this input by setting up the gating signals so that SDM0 is a 1 and SDM1 is a 0. As you can see from Table 3-1 this is the combination which selects the Unibus data. The data (which you will remember is our instruction) will be on the D MUX Bus and all that is left is for us to generate the CLK IR signal to have the instruction clocked into the I Reg.

If you still do not understand this sequence, ask your instructor about it.

D. Locate the B Reg, B Constants and B Mux which feed the BIN of the ALU.

1. Analyze the B Mux function in Table 3-1. To swap bytes the gating signals must have what values?

B Mux Low (07:00)

(a) SBML0 = \_\_\_\_\_

(b) SBML1 = \_\_\_\_\_

B Mux High (15:08)

(c) SBMH0 = \_\_\_\_\_

(d) SBMH1 = \_\_\_\_\_

2. What would be the value of these signals to accomplish sign extension of the value in the B Reg?

(a) SBML0 = \_\_\_\_\_

(b) SBML1 = \_\_\_\_\_

(c) SBMH0 = \_\_\_\_\_

(d) SBMH1 = \_\_\_\_\_

OR

(e) SBML0 = \_\_\_\_\_

(f) SBML1 = \_\_\_\_\_

(g) SBMH0 = \_\_\_\_\_

(h) SBMH1 = \_\_\_\_\_

Indicate on your block diagram where byte manipulation takes place.

3. The CIN MUX feeds the input designated CIN with one of the following values:

(a) \_\_\_\_\_

(b) \_\_\_\_\_

(c) \_\_\_\_\_

Discuss some of the functions for which these inputs could be used.

4. The AIN input is fed from a 16 bit inverter labeled BUFFER which is fed from a wired -OR indicated by the dashed OR symbol. At this point, data comes from:

(a) \_\_\_\_\_

(b) \_\_\_\_\_

(c) \_\_\_\_\_

5. You will find the COUT MUX below and to the left of the ALU. Observe that this multiplexor has four inputs which are:

(a) \_\_\_\_\_

(d) \_\_\_\_\_

(b) \_\_\_\_\_

(c) \_\_\_\_\_

Analyze and discuss among yourselves the various functions of each of these inputs.

- E. Locate the General Purpose Registers (GPR's), sometimes called Scratch Pad Registers; the Register Address Inputs; and the list of register names in the upper right hand corner of the block.
1. A brief description of the function of each of these registers is given in the first paragraph of section 3.3.5 on p. 3-15. Read it.
  2. What enabling signal will be brought up to allow us to get the address contained in R [2] for the following instruction?

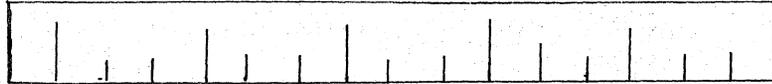
INC (2)

(a) \_\_\_\_\_

Which of the four groups of address lines will be used to select the register?

(b) \_\_\_\_\_

- F. Locate the Processor Status Word (PSW) between the B Mux and the GPR's.
1. Note how many bits are illustrated here. What are they?



2. Which bits can be affected implicitly?  
(a) \_\_\_\_\_

Discuss the reasons for this.

- G. Locate the BA Mux, the BA Reg, the Internal Data, and Address Decode and the Address Display which are found directly below the GPR's.

1. Note the BA Mux control in Table 3-1 on p. 3-14 and read section 3.3.2 on p. 3-15.

- H. At this point you should be familiar with the Data Path section of the block diagram. You should understand the function and operation of every block within this section.
- I. Refer to the list of Data Manipulation Functions in Appendix B. Write the set of functions required to implement the operations listed below. If more than one operation is required, group the functions into separate sets and complete each set with its required clocking signal.
  - 1. Move the contents of the PC into the BA.
  - 2. Increment the contents of R0, as in INC %0.
  - 3. Add the contents of R3 to the contents of R2, as in ADD %3, %2.
  - 4. Pop a word off the Processor Stack into the PC.

### III. ADDITIONAL EXERCISES

If you feel that you need more practice before you attempt the these, your instructor will be glad to provide you with more examples.

If you have been having trouble with the previous exercises, do not hesitate to ask your instructor for assistance.

### IV. POST TEST

Attempt the post test. You have twenty minutes. Do the best you can.

APPENDIX A

B 3 a 0  
 b 1  
 c PS (C)

4 a PSW ANDed with BUS RD FM PS  
 b GPR'S  
 c BUS RD (15:00) (Internal Options)

5 a PS (c)  
 b ALU 15  
 c COUT 07  
 d COUT 15

C 1 SDM0, SDM1

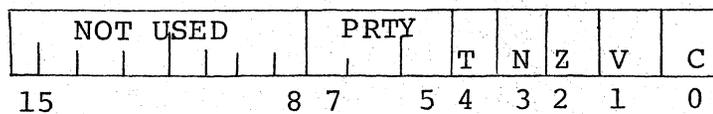
3 b SDM1 (0) \* SDM0 (1)

D 1 a SBML0 = 0  
 b SBML1 = 1  
 c SBMH0 = 0  
 d SBMH1 = 1

2 a SBML0 = 0  
 b SBML1 = 0  
 c SBMH0 = 1  
 d SBMH1 = 0  
 e SBML0 = 1  
 f SBML1 = 0  
 g SBMH0 = 1  
 h SBMH1 = 0

E 2 a SRD  
 b IR (2:0)

F 1



2 a N, Z, V, C, T, (7:5)

APPENDIX A

1 1 Operation BA ← PC

RIF (3:0) = 07  
RA (3:0) ← RIF (3:0)  
BA (15:00) ← R(15:00)  
CLK BA

2 Operation D ← R[0] + 1

RA(3:0) ← IR (2:0)  
AIN (15:00) ← R(15:00)  
CIN ← 1  
ALUout = A plus CIN  
CLK D

Alternate Method

RA (3:0) ← IR (2:0)  
AIN(15:00) ← R(15:00)  
BCON (15:00) ← 1  
BIN (15:00) ← BCON(15:00)  
CIN ← 0  
ALUout = A plus B plus CIN  
CLK D

Operation R [0] ← D

RA (3:0) ← IR (2:0)  
DMUX (15:00) ← D (15:00)  
R(15:08) ← DMUX(15:08)  
R(07:00) ← DMUX(07:00)  
WRH  
WRL

APPENDIX A

I 3. Operation  $B \leftarrow R[3]$

RA (3:0)  $\leftarrow$  IR(8:6)  
DMUX(15:00)  $\leftarrow$  R(15:00)  
BREG(15:00)  $\leftarrow$  DMUX(15:00)  
CLK B

Operation  $D \leftarrow B + R[2]$

RA(3:0)  $\leftarrow$  IR(2:0)  
AIN(15:00)  $\leftarrow$  R(15:00)  
BIN(15:08)  $\leftarrow$  BREG(15:08)  
BIN(07:00)  $\leftarrow$  BREG(07:00)  
CIN  $\leftarrow$  0  
ALUout = A plus B plus CIN  
CLK D

Operation  $R[2] \leftarrow D$

RA(3:0)  $\leftarrow$  IR (2:0)  
DMUX(15:00)  $\leftarrow$  D(15:00)  
R(15:08)  $\leftarrow$  DMUX(15:08)  
R(07:00)  $\leftarrow$  DMUX (07:00)  
WRH  
WRL

APPENDIX A

I. 4. Operation  $BA \leftarrow R[SP]$   
 $D \leftarrow R[SP] + 2$

RIF(3:0) = 06  
RA(3:0)  $\leftarrow$  RIF(3:0)  
BA(15:00)  $\leftarrow$  R(15:00)  
AIN(15:00)  $\leftarrow$  R(15:00)  
BCON(15:00)  $\leftarrow$  2  
BIN(15:00)  $\leftarrow$  BCON(15:00)  
CIN  $\leftarrow$  0  
ALUout = A plus B plus CIN  
CLK BA  
CLK D

Operation  $R[SP] \leftarrow D$

RIF(3:0) = 06  
RA(3:0)  $\leftarrow$  RIF(3:0)  
DMUX(15:00)  $\leftarrow$  D(15:00)  
R(15:08)  $\leftarrow$  DMUX(15:08)  
R(07:00)  $\leftarrow$  DMUX(07:00)  
WRH  
WRL

(DATI)

Operation  $R[PC] \leftarrow$  BUS Data

RIF(3:0) = 07  
RA(3:0)  $\leftarrow$  RIF(3:0)  
DMUX(15:00)  $\leftarrow$  BUS D(15:00)  
R(15:08)  $\leftarrow$  DMUX(15:08)  
R(07:00)  $\leftarrow$  DMUX(07:00)  
WRH  
WRL



KD11A DATA MANIPULATION FUNCTIONS

GENERAL PURPOSE  
REGISTERS

RA(3:0) ← RIF(3:0)	RIF(3:0) = 00
RA(3:0) ← IR(8:6)	01
RA(3:0) ← IR(2:0)	⋮
RA(3:0) ← BA(3:0)	⋮
	16
R(15:08) ← DMUX(15:08)	RIF(3:0) = 17
R(07:00) ← DMUX(07:00)	

ARITHMETIC LOGIC UNIT

AIN(15:00) ← R(15:00)	BCON(15:00) ← 1
AIN(07:00) ← PS(07:00)	← 2
	← 4
BREG(15:00) ← DMUX(15:00)	← 10
BIN(15:00) ← BCON(15:00)	BCON(15:00) ← 177570
BIN(15:08) ← BREG(15:08)	
BIN(07:00) ← BREG(07:00)	ALUout = $\frac{A}{A}$ plus CIN
BIN(15:08) ← BREG(07:00)	A plus CIN
BIN(07:00) ← BREG(15:08)	A plus B plus CIN
BIN(15:08) ← BREG(07)	A Minus B
	ALUout = B plus CIN

CIN ← 1    CIN ← 0    COUT ← PS(c)    COUT ← ALU15

D(15:00) ← ALUout(15:00)	PS(7:0) ← DMUX(7:0)
DMUX(15:00) ← D(15:00)	IR(15:00) ← DMUX(15:00)
DMUX(15:00) ← R(15:00)	
DMUX(15:00) ← BUS D(15:00)	BA(15:00) ← R(15:00)
DMUX(15:00) ← COUT, D(15:01)	BA(15:00) ← ALUout(15:00)

CLOCKING SIGNALS

WRH	CLKB	CLK BA	CLK IR
WRL	CLKD	CLK PS	



KD11A 020 000

LEARNING ACTIVITY UNIT

KD11A BLOCK DIAGRAM

CONTROL SECTION

1/15/74



RATIONALE

By now you have had a good overview of the KD11A Block Diagram. You have learned how data flows through and how it is manipulated within the CP. You have also learned how the 11/05 Control Section operates and how it affects the manipulation of data in the 05.

Now, in studying the 11/40 Control Section, you will first approach it from the point of view of how it is similar to the 11/05. After that, you will analyze the different concepts which the 11/40 utilizes.

A complete understanding of the 11/40 Control Section is imperative if you are to troubleshoot the machine properly. If you do not realize what is actually happening in this section, it will be almost impossible to comprehend the flow diagrams or to predict what will happen when you are using the maintenance board.



PREREQUISITES

Satisfactory completion of all units to date.

OBJECTIVES

- A. Given a list of Control Section Facilities, you must be able to match these with five generalized functional descriptions.
  
- B. Given a list of microinstructions, show what will be contained in certain registers and what will be on certain lines in the Control Section during the execution of each microinstruction.

You will have thirty minutes to do this and will be allowed one error in A and one in B.

## LEARNING ACTIVITIES

## I. LECTURE

Attend the forty-five minute lecture. Make notes on your block diagram. Hold your questions until the end of the lecture.

## II. PROJECTS/PROBLEMS (Answers in Appendix B)

A. Given the list of Control Section Facilities in Appendix A, choose one which is best described by each of the eight following functional statements.

1. It takes signals from the FLAG Control I REG and IR Decode to provide signals for the BUT MUX.

Ans. \_\_\_\_\_

2. It provides the controlling signals for the ALU

Ans. \_\_\_\_\_

3. It provides the controlling signals for the D MUX.

Ans. \_\_\_\_\_

4. Its output can be used to stop the processor at a particular micro instruction

Ans. \_\_\_\_\_

5. Its contents specify the address of the microword currently in the UREG.

Ans. \_\_\_\_\_

6. It contains the machine instruction currently being executed by the 11/40.

Ans. \_\_\_\_\_

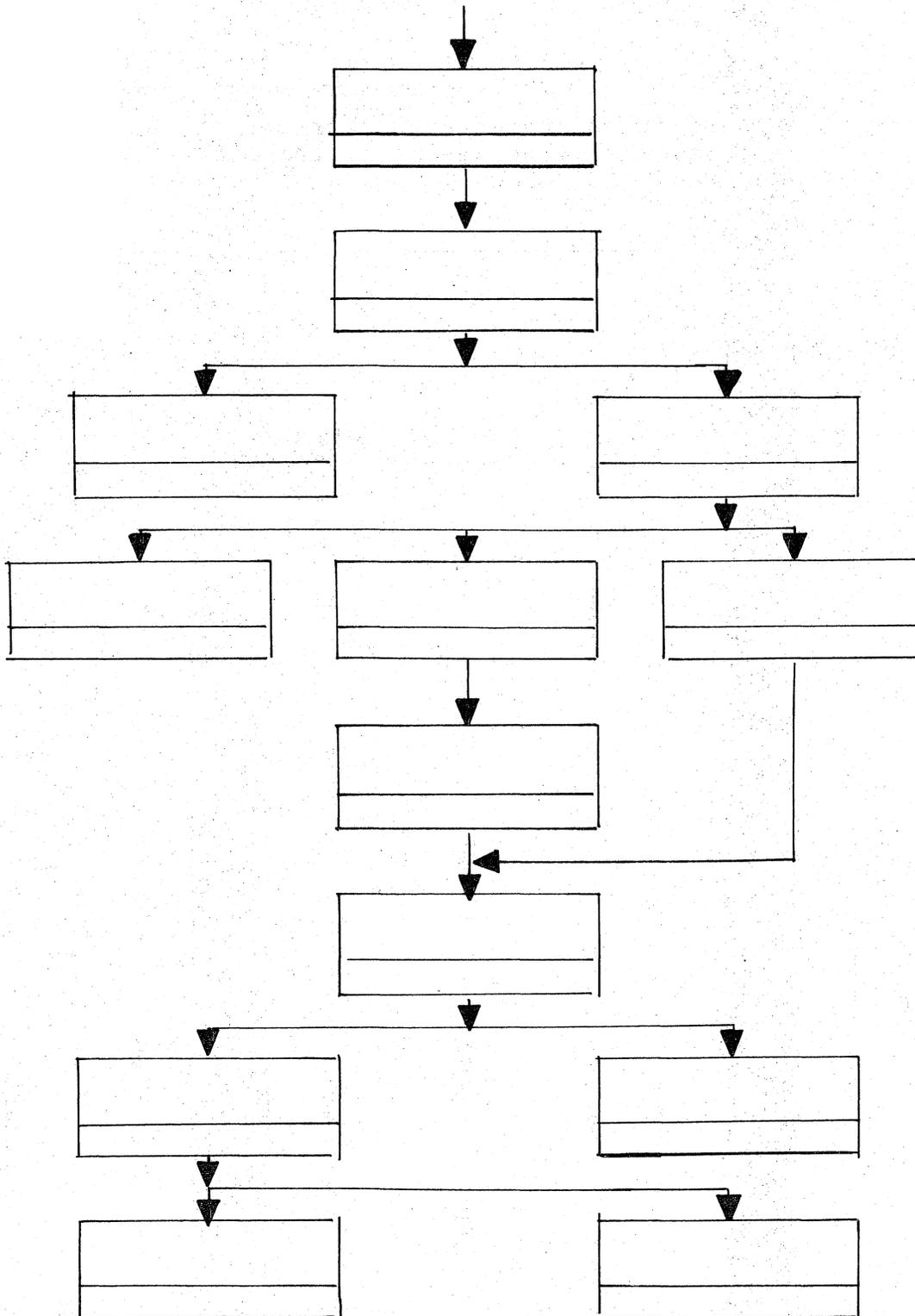
7. Its output forces the micro program to a particular address without regard to the microinstruction currently being executed.

Ans. \_\_\_\_\_

8. These lines indicate whether or not, and if so, what alterations of the base address will occur.

Ans. \_\_\_\_\_

- B.  
1. Which of the microinstructions drawn below would contain a BUT?



C. Take the POST TEST. Do the best you can.

III. ADDITIONAL READING

To enhance your understanding of this unit, it is recommended that you read and study pages 3-18 through 3-20 in the KD11A Processor Maintenance Manual.

WHAT'S NEXT

In the next unit you will complete your study of the Block Diagram. You will also have a chance to review the entire block.

APPENDIX A

## LIST OF CONTROL SECTION FACILITIES

1. I REG
2. IR Decode
3. FLAG Control
4. UBC Branch Control
5. ALU Control MUX
6. ALUM, ALUS (3:0)
7. BUT MUX
8. JAMUPP
9. UPP
10. Main Control ROM
11. PUPP
12. SR: BUPP Compare
13. UPF (7:0)
14. U REG
15. U (15:09)
16. UBF (4:0)
17. BUT Decoder

APPENDIX B

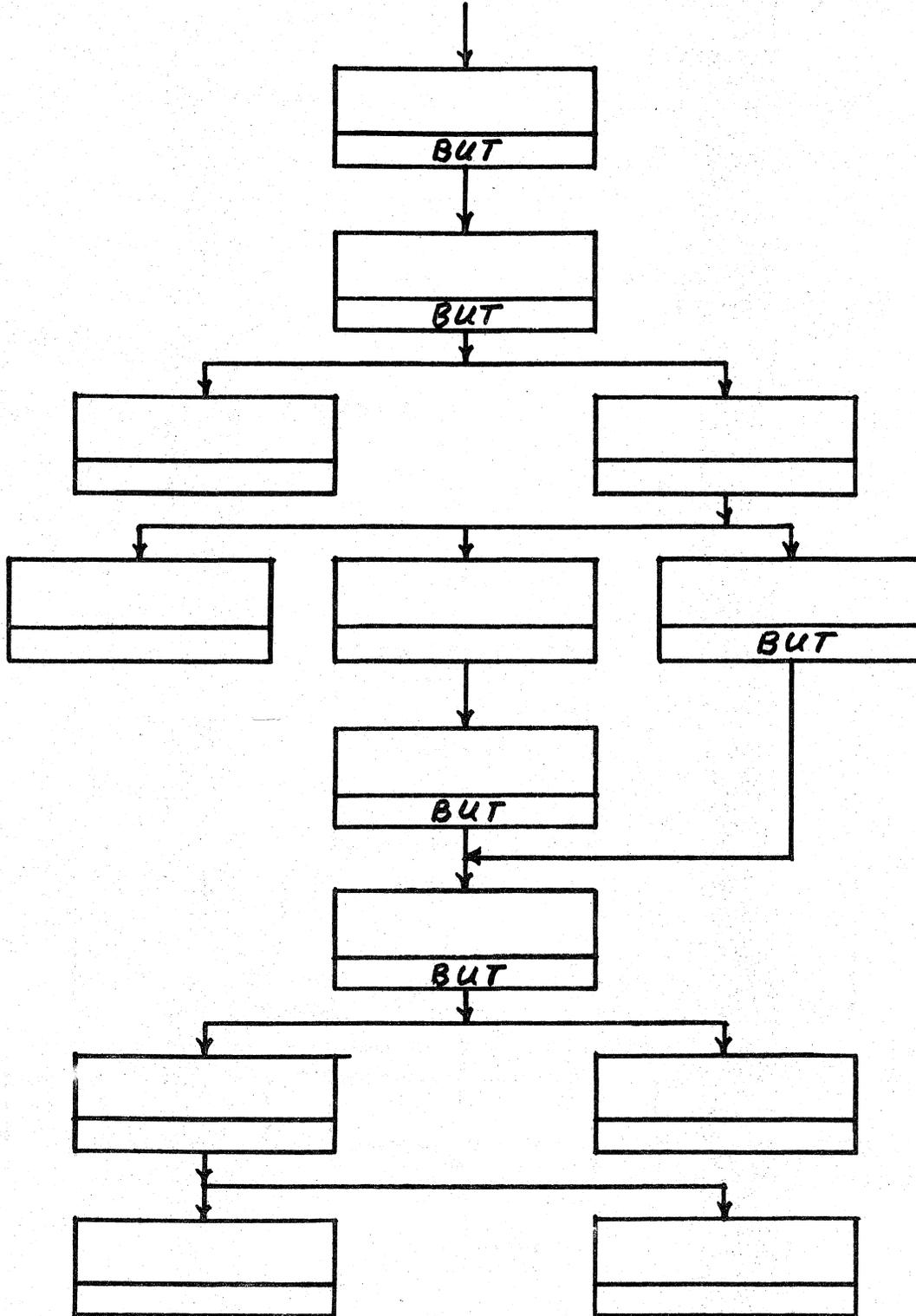
ANSWERS TO PROBLEMS

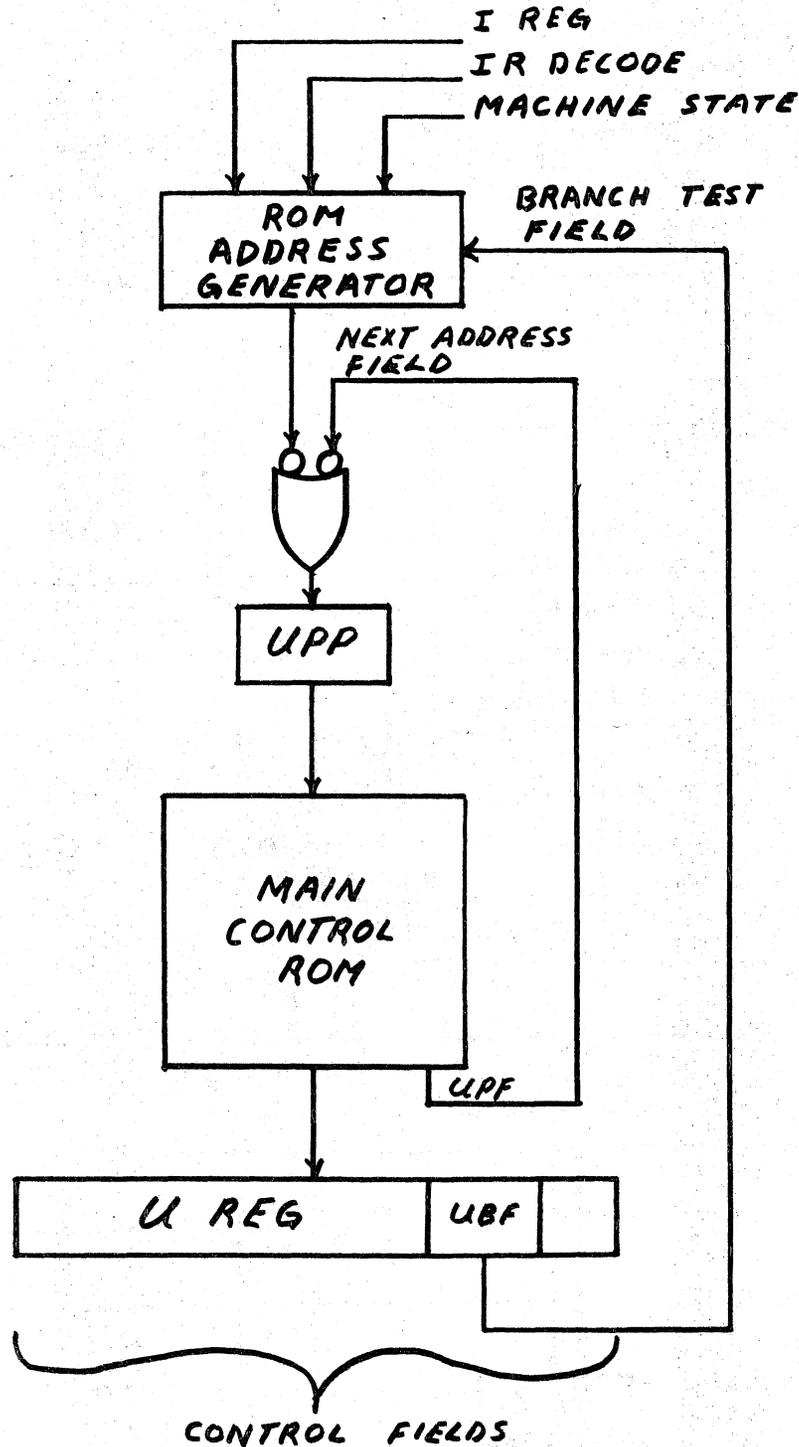
1. UBC Branch Control
2. ALU Control MUX
3. U REG
4. SR: BUFP Compare
5. PUPP
6. I REG
7. JAMUPP
8. UBF (4:0)

APPENDIX B

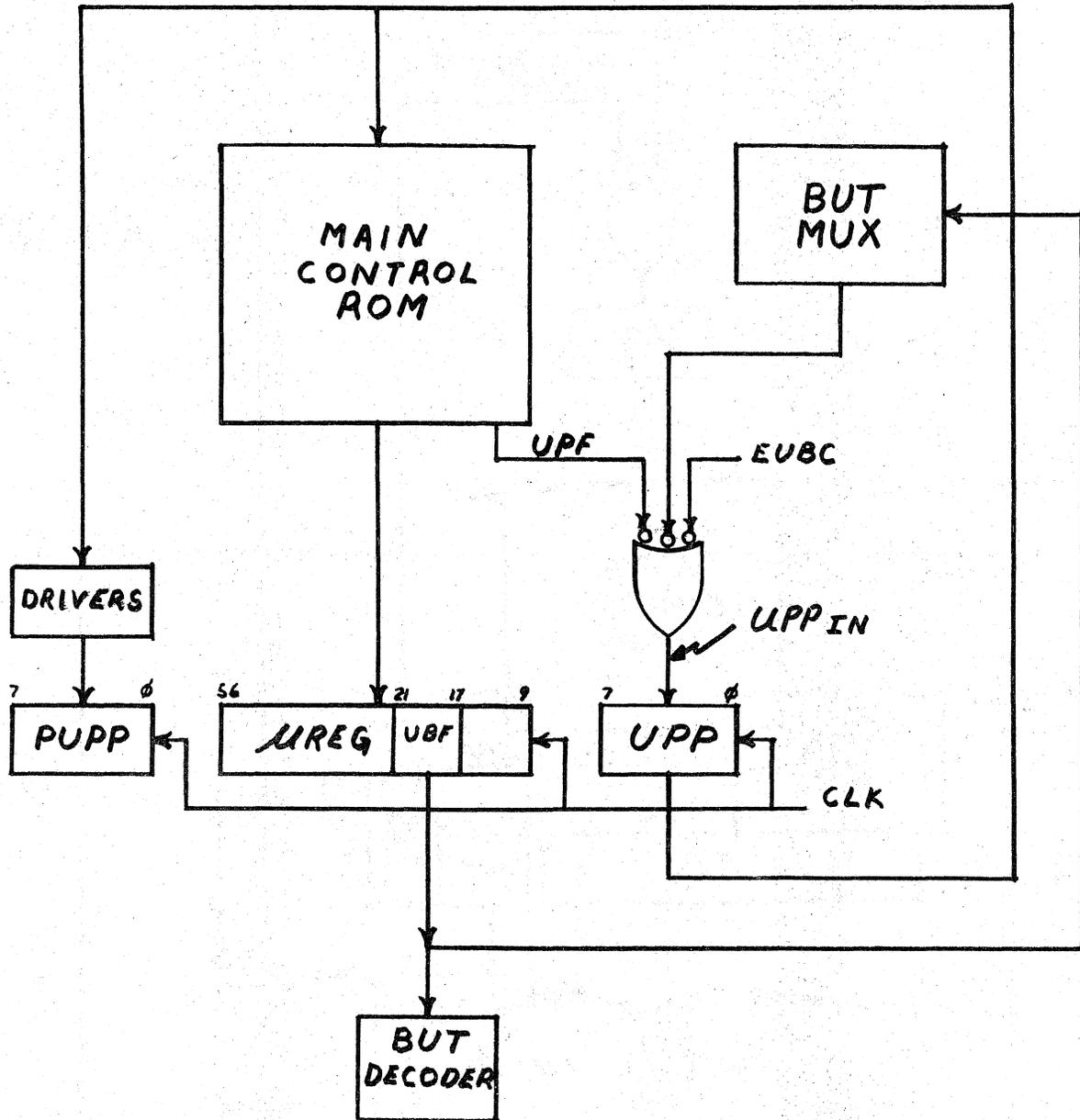
ANSWERS TO PROBLEMS

B.

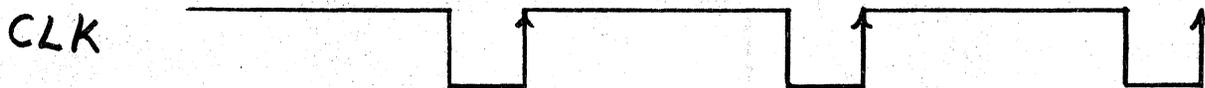
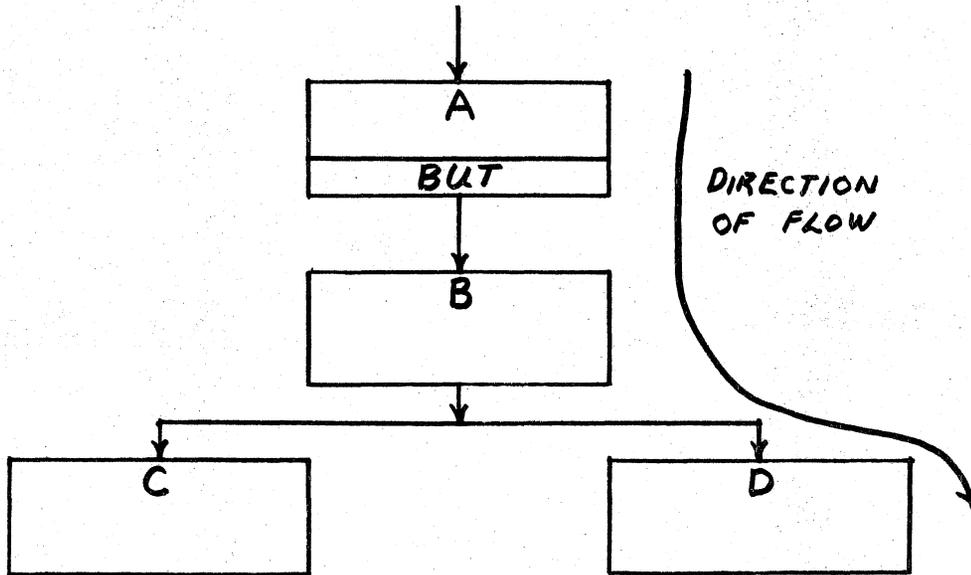




CONTROL SECTION  
SIMPLIFIED BLOCK DIAGRAM



SIMPLIFIED MICROBRANCH CONTROL



PUPP	A	B	D
U PP	B	D	
U PP IN	D		
U REG	A	B	D

EXAMPLE OF MICROBRANCHING



KD11A - 030.000

LEARNING ACTIVITY UNIT

KD11A BLOCK DIAGRAM  
CLOCK, UNIBUS, CONSOLE

JANUARY 17, 1974



RATIONALE

By now you are pretty familiar with the 11/40 Block Diagram. You have seen how data is manipulated within the C.P. and also how the Control Section causes this manipulation to take place.

In this unit, you will cover the remaining parts of the Block Diagram. You will learn primarily how the 11/40 Processor clock operates and how these clocking signals are utilized within the Control and Data Path sections. You will also learn about such miscellaneous portions of the block as the Unibus Interface, Console Interface, etc. This unit will also serve as a review of the entire KD11A Block Diagram.

Use this unit to make sure that you really understand the Block Diagram. You will find it immensely useful as you study the flows and the logic as well as when you are troubleshooting.



PREREQUISITES

Successful completion of all units to date.

OBJECTIVES

Given a list of blocks or signals, you must be able to indicate which one would be faulty to generate the symptoms described. You must successfully complete three of the four questions in the allotted thirty minutes.

LEARNING ACTIVITIES

## I. LECTURE

Attend the thirty minute lecture. Make notes on your block diagram. Do not ask your questions until the end of the lecture.

## II. PROJECTS AND PROBLEMS

A. Refer to Fig. 3-11 in your KD11A Processor Maintenance Manual. Write down which signal would be faulty to cause the following symptoms. Check your answers with those in Appendix B. Be sure to ask your instructor to clarify any point you do not fully understand.

1. The machine is hung doing nothing and you observe no clocking signals. However, you are able to do all operations successfully using the maintenance board.

Answer \_\_\_\_\_

2. Any microinstruction which specifies a CL2 results in the generation of both a P2 and a P3 pulse. Microinstructions specifying a CL1 or CL3 operate properly.

Answer \_\_\_\_\_

B. Refer to the list of Data Manipulation Functions in the Appendix of LAU# KD11A - 010.000 and to your KD11A Block Diagram. Write the set of facilities including the clock length which would execute the following operations most efficiently. The answers are in Appendix B.

1.  $PC \leftarrow PC + 2$
2.  $R5 \leftarrow R3 + R5$
3. Push the contents of the PC onto the Processor Stack.

4. Depressing the LOAD ADRS key
- C. Using the KD11A Block Diagram, find the signal which would be faulty to cause the following symptoms.
1. Whenever you do a LOAD ADRS operation, you observe on the ADDRESS DISPLAY that the upper Byte is always changed to contain garbage.

Answer \_\_\_\_\_

2. Whenever the D Mux is enabled such that you see the contents of the D Reg in the Data Display, you see garbage. The same garbage is seen every time the D Mux is enabled in this way.

Answer \_\_\_\_\_

3. You are trying to execute a program. After pressing and releasing the Start switch, the machine Halts. The same thing happens when you subsequently press CONT. You observe that the PC is incrementing properly and that R13 contains the instruction it should have executed.

Answer \_\_\_\_\_

- D. Take the Post Test. Do the best you can.

### III. ADDITIONAL READING

To enhance your understanding of the Clock section, read and study pages 3-21 to 3-22 of the KD11A Processor Maintenance Manual. You should, at this time, be able to read and understand all of Chapter 3 from page 3-12.

#### WHAT'S NEXT?

You should now be well prepared to continue your study of the Flow Diagrams. If you have the Block well under your belt, you should have no problem with the flows.

APPENDIX AREGISTER CLOCKING PULSES

<u>REGISTER</u>	<u>CLOCK</u>
I REG	P1, P3
B REG	P1, P3
D REG	P2
GPR	P1, P3
BA REG	P1, P2

APPENDIX BANSWERS TO PROBLEMS

A.

1. RECLK
2. CLKL0

B.

1.  $PC \leftarrow PC + 2$

$RIF(3:0) = 07$   
 $RA(3:0) \leftarrow RIF(3:0)$   
 $BCON(15:00) \leftarrow 2$   
 $BIN(15:00) \leftarrow BCON(15:00)$   
 $AIN(15:00) \leftarrow R(15:00)$   
 $CIN \leftarrow 0$   
 $ALUout = A PLUS B PLUS C$   
 $D(15:00) \leftarrow ALUout(15:00)$   
 CLK D  
 $DMUX(15:00) \leftarrow D(15:00)$   
 $R(15:08) \leftarrow D MUX(15:08)$   
 $R(07:00) \leftarrow D MUX(07:00)$   
 WRH  
 WRL  
 CLKL3

2.  $R5 \leftarrow R3 + R5$

- a.  $B \leftarrow R3$

$RA(3:0) \leftarrow IR(2:0)$   
 $D MUX(15:00) \leftarrow R(15:00)$   
 $B(15:00) \leftarrow D MUX(15:00)$   
 CLK B  
 CLKL1

- b.  $R5 \leftarrow B + R5$

$RA(3:0) \leftarrow IR(2:0)$   
 $AIN(15:00) \leftarrow R(15:00)$   
 $BIN(15:08) \leftarrow B(15:08)$   
 $BIN(07:00) \leftarrow B(07:00)$

APPENDIX B

## ANSWERS TO PROBLEMS (CONT'D)

B.

2.

b.  $CIN \leftarrow 0$   
 $ALUout = A PLUS B PLUS CIN$   
 $D (15:00) \leftarrow ALUout (15:00)$   
 CLK D  
 $D MUX (15:00) \leftarrow D (15:00)$   
 $R (15:08) \leftarrow D MUX (15:08)$   
 $R (07:00) \leftarrow D MUX (07:00)$   
 WRH  
 WRL  
 CLKL3

3. (SP)↓ ← PC

a. BA, SP ← SP - 2

$RIF (3:0) = (06)$   
 $RA (3:0) \leftarrow RIF (3:0)$   
 $AIN (15:00) \leftarrow R (15:00)$   
 $BCON (15:00) \leftarrow 2$   
 $BIN (15:00) \leftarrow BCON (15:00)$   
 $ALUout = A MINUS B$   
 $BA (15:00) \leftarrow ALUout (15:00)$   
 CLK BA  
 $D (15:00) \leftarrow ALUout (15:00)$   
 CLK D  
 $D MUX (15:00) \leftarrow D (15:00)$   
 $R (15:08) \leftarrow D MUX (15:08)$   
 $R (07:00) \leftarrow D MUX (07:00)$   
 WRH  
 WRL  
 CLKL3

b. BUSD ← PC

$RIF (3:0) = 07$   
 $RA (3:0) \leftarrow RIF (3:0)$   
 $AIN (15:00) \leftarrow R (15:00)$   
 $CIN \leftarrow 0$   
 $ALUout = A PLUS CIN$   
 $D (15:00) \leftarrow ALUout (15:00)$

APPENDIX B

## ANSWERS TO PROBLEMS (CONT'D)

B.

3.

b. CLK D  
CLKL2

(DATO)

4. LOAD ADRS

a. R(17) ← SR

RIF (3:0) = 17

RA (3:0) ← RIF (3:0)

BUS D (15:00) ← SR (15:00)

D MUX (15:00) ← BUS D (15:00)

R (15:08) ← D MUX (15:00)

R (07:00) ← D MUX (07:00)

WRH

WRL

CLKL1

b. BA ← R(17)

RIF (3:0) = 17

RA (3:0) ← RIF (3:0)

BA (15:00) ← R (15:00)

CLKBA

CLKL1

C. 1. WRH

2. CLKD

3. CLKIR



KD11A 040.000

LEARNING ACTIVITY UNIT

FLOW DIAGRAMS I

INTRODUCTION OVERVIEW / INST FORMAT



RATIONALE

So far you have learned about the Block Diagram and can now identify those areas of the block known as Data Paths, Control Section, Clock, Unibus Timing and Control. You can visualize, for instance, how some of the elementary operations are accomplished via the block like updating the PC, IR gets Unibus Data, etc. Now it is time to go one step farther and move on to the flows.

Probably everyone's definition of flow diagrams include something about a step by step explanation of an operation. Flows are designed to make some operation easily understandable. It usually consists of a set of symbols which must be identified and learned. The flows that you learned in the 11/05 were more like a listing and therefore, the ideas were there but not the symbology. This is not true, however, with the 11/40. Once we learn the "language of the flows" an attempt will be made to understand what the flows tell us concerning the 11/40's operation. Our primary objective in this unit then will be to learn how to "read" the flows.

The whole concept of the ROM control is captured in these flows. They identify and explain each micro instruction and show each possible sequence. Acquiring this reading the flows ability will provide you with the proper foundation to continue with the subsequent units concerning the flows.



PREREQUISITES

Satisfactory completion to date of all previous units.

OBJECTIVES

Given four flow machine state blocks with various amounts of information you must identify and define which micro instruction field is involved to select or make happen the questioned operation. To successfully complete this unit, you must correctly answer three of four of these questions. You will be allowed to use all available references and your notes but a twenty (20) minute time limit will be in effect.

PRETEST

To be filled in when fully programmed environment is in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend a 20 minute lecture where the instructor will give you an overview of the microprogram flows. He will point out to you the various places you will find information about the flows, and talk in general terms about the ideas and concepts behind them. The emphasis will be on how to use the flows in a maintenance and learning environment. Time will be allowed at the end of this lecture to answer questions prior to going on to your activities.

## II. PROJECTS/QUESTIONS

## A. Micro instruction format

As you already know, the 11/40 is a "ROM" machine. This means that all controlling of the machine operations are accomplished by micro instructions. These micro instructions are 56 bits in length and there are 256 of them.

1. Identify, define, describe each of the 56 bits of a micro instruction. A complete detailed description appears in the appendix. A ready reference appears in the:  
KD11A Processor Maintenance Manual DEC11 - HKDAA-A-D, Pg. 2-7 Fig, 2-4, Pg. 2-8, 2-9 Table 2-1
2. Note how the bits are grouped into the various fields, that is; the CLKL 1, CLKL0 and CLKOFF grouped together to become a field called CLK. Note all fields by referring to KD11A Processor Maintenance Manual DEC11-HKDAA-A-D, pg 2-6 Fig. 2-3

3. Refer to the 6th page in the PDP11/40 System Engineering Drawings and look over the tables found there. Each of the major field's uses are given by these tables. For instance when the BUS code is equal to 5, what type of bus transaction will be initialized; answer DATO.
4. Now familiarize yourself with the ROM (micro instruction) listing in the PDP11/40 System Engineering Drawings found on about the 22nd page just after the K2 print set.
5. Refer to your block diagram, now and identify some of these fields as to where they feed and what they control
  - a. SRI and RIF to select some general purpose register
  - b. SBC to generate a constant
  - c. UPF to affect the UPP register
  - d. UBF to control the BUT multiplexer
6. The SBC code will be equal to \_\_\_\_\_ when generating a constant of two (2).
7. When attempting to address R[PC] using SRI and the RIF bits, SRI is on a 1 and RIF equals \_\_\_\_\_.
8. When DAD says use the micro instruction to set up the ALU and we want the ALU to add, the ALU field equals \_\_\_\_\_.
9. The BUT decoder looks at the \_\_\_\_\_ field to translate the BUT number.
10. A BUT 12 really asks this question is \_\_\_\_\_?

B. Micro program Flow

1. Only three basic symbols are used in the PDP11/40 Flow diagrams, they are given here and on pg 4-1 of the KD11A Processor Maintenance Manual DEC11-HKDAA-A-D.

Entry Point



KD11A 040.000



Machine State



Exit Point

2. The most involved symbol of course is the machine state (one micro instruction) Refer now to the appendix and familiarize yourself with every aspect of this flow symbol.
3. Each machine state box represents one micro instruction, one elementary operation, one small step necessary for the overall accomplishment of a machine operation.
4. A machine operation could be;
  - a. To load an address
  - b. To examine a register
  - c. To execute a machine instruction
  - d. Etc.
5. Read pages 4-1 thru 4-4 in the KD11-A Processor Maintenance Manual DEC 11-HKDAA-A-D with special emphasis on the flow diagram example figure 4-2 on page 4-2
6. Start (1) indicates to you that start is a (condition/flip flop)
7. The signal - HALT SW is read as  

---
8. Referring to fig 4-2, page 4-2 in KD11A Processor Maintenance Manual DEC11-HKDAA-A-D
  - a. What flow sheet do you go to if HALT switch is sensed
  - b. What address is Fetch C
9. Check your answers to A and B in the appendix. Get help if you do not understand why you missed any of the one's you did.

III. SELF EVALUATION

A. Given the following Micro instruction flow example, answer the questions

SRC 09 147

GET INDEX DATA; MODIFY	D
P2: $D \leftarrow R[PC]$ plus 2	
P3: $R[PC] \leftarrow D$ ; CLKOFF	

1. What is the clock code?
  
2. If  $D=500$  and  $R[PC] =500$   
What is the Data Display?
  
3. The SBM field will equal \_\_\_\_\_
  
4. What method will be used to address  $R[PC]$ .
  
5. The PC will be written with the updated value and the next micro instruction clocked into the Register at the trailing edge of the \_\_\_\_\_ clock specified.

IV. ADDITIONAL READING

To assist in you understanding, the flow diagrams reading chapter two (2) of the KD11A Processor Maintenance Manual DEC11 - HKDAA A-D will be of some value. You should consider this for a more detailed understanding.

WHAT'S NEXT

After completing this unit, you will be prepared to continue on to the remaining units concerned with the 11/40 flow diagrams armed with the necessary knowledge knowing how to "read the FLoWs".

FLOW DIAGRAM DESCRIPTION

(A) FET 04

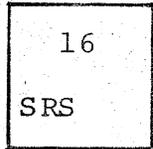
(B) 004

(C) MODIFY R[PC] ; OVERLAP	(D) R [PC]
(E) P2: D, BA ← R[PC] PLUS 2 :  DATA IF OVERLAP FETCH	
(F) BUT (INSTR 1)	(G) BUT 37

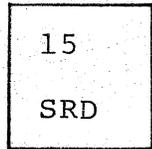
(H) 100

- A. Name of Micro instruction (Mnemonic)
- B. The ROM address of the  $\mu$ instruction (9 bits - (PUPP))
- C. What the  $\mu$ instruction will do. I.E. modify the program counter and start a BUS Cycle if overlap.
- D. The Console Data Display. This is read R of PC. Therefore, this Data Display is the contents of R[PC].
- E. What happens and when. This particular one means, at the trailing of the P2 pulse specified D Reg. and BA Reg., get the results of R[PC] plus 2, and start a DATA BUS Cycle if you meet the conditions for an overlap fetch. This is read as, at P2 time D and BA gets R[PC] plus 2 and start a DATA IN if overlap.
- F. Branch Micro test name. This one, but (instr 1), means this is first time you look at the machine instruction to make micro program branch decision.
- G. But Number. Give you the octal  $\mu$ BF code, let you refer to this BUT by number.
- H. This is the base address that the micro test will put in jeopardy. It is possible that any or all of BITS (5:0) will be modified due to conditions being sampled (tested).

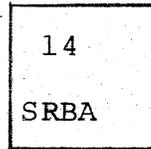
MICRO INSTRUCTION FORMAT PDP11/40



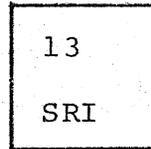
Allows IR source field to address the GPR's



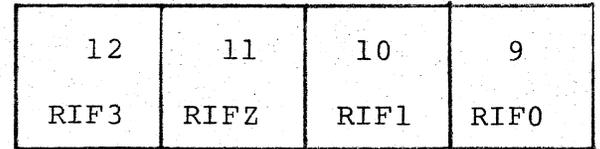
Allow IR destination field to address the GPR's.



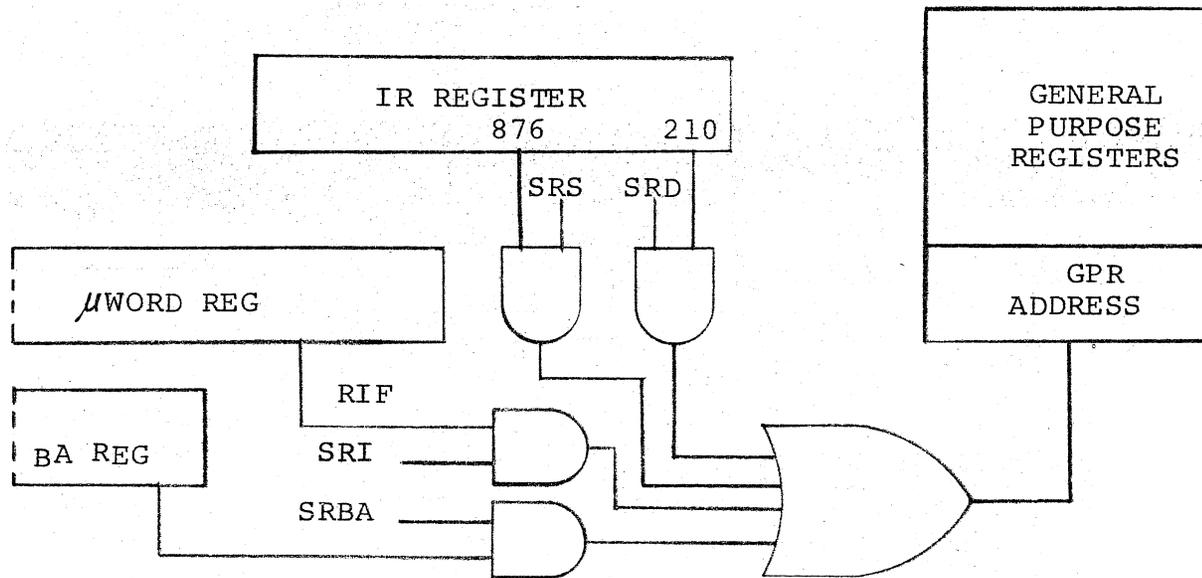
Allows bits (3:0) of the BA Reg. to address the GPR's.



Allows RIF to be used (select the GPR using the RIF field)



When SPI is active these bits (register immediate field) are used to provide the address for the general purpose registers.



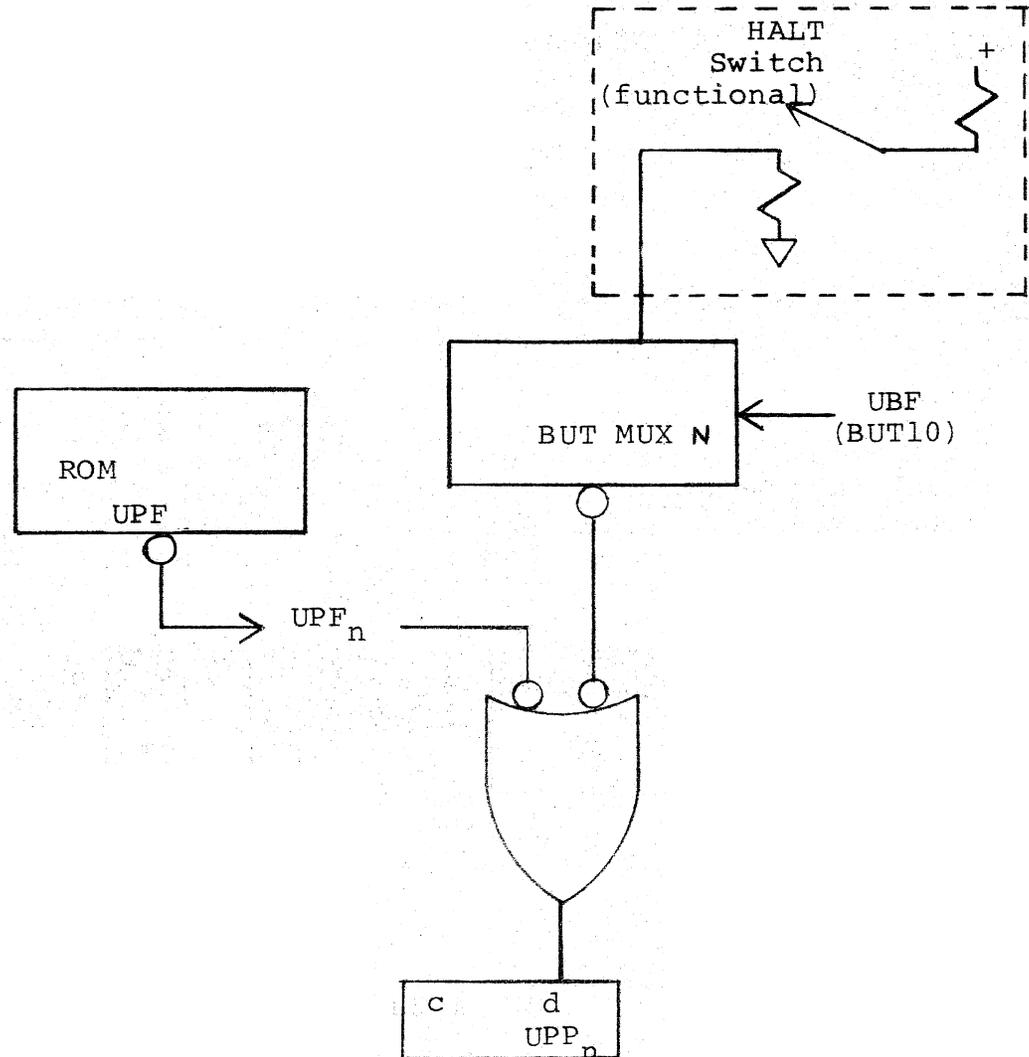
Functional representation of GPR addressing.

MICRO INSTRUCTION FORMAT PDP11/40

21	20	19	18	17
UBF4	UBF3	UBF2	UBF1	UBF0

Micro branch field - better known as BUT. It's this field that will select one of many inputs to be routed through the but mux, such that, that condition could modify the UPF bit depending on the fact if it's true or not.

BUT's asks question for instance, a but 10 called but (halt) says but is the halt switch on?  
 BUT 17, BUT IR03, BUT is IR bit 3 on a 1?



MICRO INSTRUCTION FORMAT PDP11/40

28	27	26	25
SBMH1	SBMH0	SBML1	SBML0

Select the B MUX field. Bits 28 & 27 select the input to the high byte of the B MUX. Bits 26 & 25 select the input to the LO byte of the B MUX.

24	23
SDM1	SDM0

Select the D MUX field. These bits select which of the 4 inputs to be routed through the D MUX.

22
SBAM

Select the BA MUX. This bit routes either the RD BUS or the ALU output to the BA Register.

See table 3-1 on page 3-14 in the K11A Processor Maintenance Manual DEC11-HKDAA-A-D for a complete listing of multiplexer selections.

MICRO INSTRUCTION FORMAT PDP11/40

ALU

37	36	35	34	33
SALUM	SALU3	SALUZ	SALU1	SALU0

These bits select which mode the ALU should function in and what it should do. I.E. Add, etc. see table 3-2, page 3-17 in the KD11-A Processor Maintenance Manual for complete list of ALU selections.

SBC

32	31	30	29
SBC3	SBC2	SBC1	SBC0

Select Binary Constants field  
This bit generates necessary constants needed by the 11/40. See SBC table about 6th page in PDP11/40 System Engineering Drawings for a complete list.

MICRO INSTRUCTION FORMAT PDP11/40

DAD

44	43	42	41
DAD3	DAD2	DAD1	DAD0

Discrete alteration of data. Does some special jobs in the logic for us. For instance, the micro program would go through the same sequence if the machine instruction was an Add or Sub. However, at that one micro instruction that the ALU actually operates on the two operands the DAD code allows the ALU to be told what to do (Add or Sub) by the OP code in the IR Register.

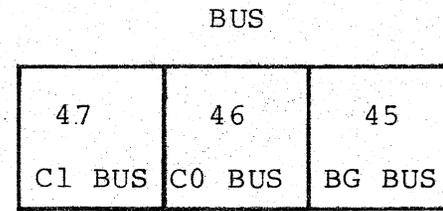
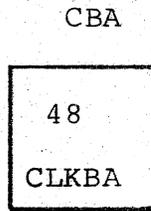
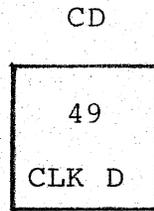
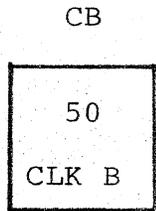
SPS

40	39	38
SPS2	SPS1	SPS0

Select the processor status code. These bits control the gating and clocking of the processor status flip flops.

A complete table of "DAD" and "SPS" is found about the 6th page of the PDP11/40 System Engineering Drawings.

MICRO INSTRUCTION FORMAT PDP11/40

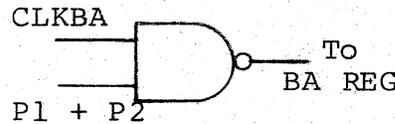
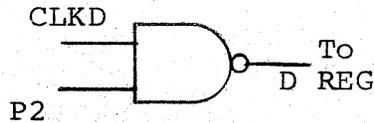
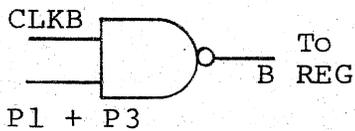


Allows clocking DMUX (15:00) into the B Register

Allows clocking the ALU output into the D Register

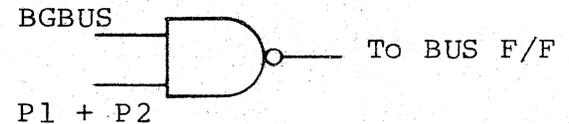
Allows clocking either the ALU output or a general register output into the BA Register

BUS Control bits. Bits 47 and 46 specify type of data transaction



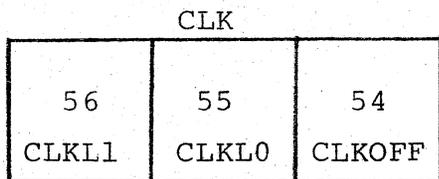
C1BUS	C0BUS	
0	0	DATI
0	1	DATIP
1	0	DATO
1	1	DATOB

Bit says to BEGIN a BUS Cycle

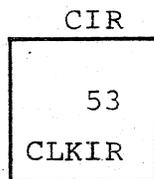


This is a three bit code given as one octal digit in the ROM listing. This code has a supplementary use - explained with tables.

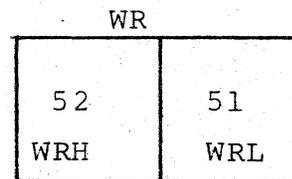
MICRO INSTRUCTION FORMAT PDP11/40



Clock Control bits  
Bit 56 and 55 select  
one of three clock  
lengths.



Allows clocking the  
instruction register  
with Unibus Data -  
the machine instruction.



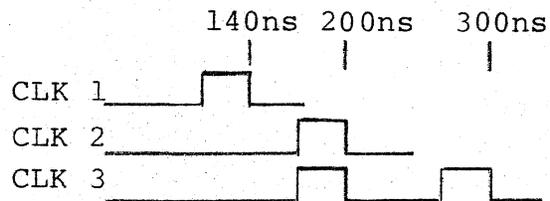
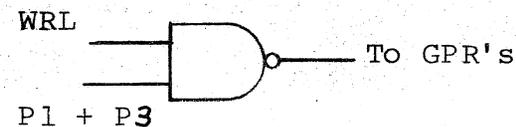
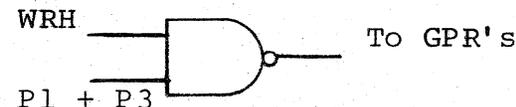
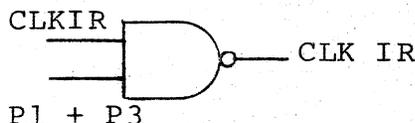
Allows the writing of  
DMUX information into  
a selected general  
purpose register. Notice  
a control signal for  
upper byte and lower  
byte.

APPENDIX

CLKL1	CLKL0	
0	0	CLK 1
0	1	CLK 1
1	0	CLK 2
1	1	CLK 2

Bit 54 allows micro instruction  
to turn off processor clock

This is a three bit code given as  
one octal digit in the ROM listing.



Shown so clocking  
happens on trailing  
edge.



APPENDIX

Answers to A and B

A

6. 02
7. 07
8. 11
9. UBF
10. Is D Reg equal to zero

B

7. Not HALT
- 8 a 10
- b 016

Answers to Self Evaluation

A

1. 7
2. 500
3. 17
4. SRI(1), RIF=17
5. P3



REVISION

Your instructor is interested in improving this unit based on you answers to the following questions. Please answer them objectively and return this sheet to your instructor.

1. Relevance of this unit to your job as a Field Service Engineer.
  - a. Very relevant
  - b. Average relevance
  - c. Not relevant
  - d. Worthless
  
2. Did you have sufficient time?
  - a. Just right
  - b. Too much
  - c. Need a little more
  - d. Need a lot more
  
3. Degree of difficulty experienced
  - a. Average
  - b. Too difficult
  - c. A little tough
  - d. A snap
  
4. Quality of the LAU with reference to previous ones
  - a. Very good
  - b. Good
  - c. Fair
  - d. Poor
  
5. Did you feel you were given adequate directions either from the instructor of the LAU itself.
  - a. Adequate
  - b. Need a little more
  - c. Need a lot more
  - d. I was confused
  
6. Additional comments on the unit:

---

---

---

---



LEARNING ACTIVITY UNIT

Flow Diagrams II

JAMUPP to Console and Console Flows

1/21/74



RATIONALE

In the previous unit you started the flow diagram learning process by learning how to read the microprogram flow diagrams. You also realized that many of the ideas and concepts you mastered in the 11/05 Flows could be applied with good results here in the 11/40. In our quest to master these flows this was the first of several steps.

Perhaps the one underlying concept in all ROM machines and at least in the 11/40 is that one small step at a time is accomplished. These small steps (micro instructions) when taken in the aggregate perform some major operation or job. For instance, execute a machine instruction, accomplish a console operation, etc. In this unit you will undertake the job of learning how the console operations in the 11/40 are accomplished using micro instructions. You shall see how, if we take one small step at a time, you can do the entire job that is required.

All major concepts are found in these console flows. All of the ideas, branching, testing, bus transactions, etc. are here. If you can really understand how these flows work you will obtain an appreciation for how much of the processor must be operational to do these console operations. The remaining flow diagrams will become much easier to master once a thorough knowledge of the console flows is obtained.



PREREQUISITES

Satisfactory completion to date of all previous units.

OBJECTIVES

There will be five (5) questions concerning microinstructions and their operations and how they relate to the console operations. To demonstrate successful completion of this unit, you must answer at least four (4) of these correctly. Your notes and any reference material are at your complete disposal, **however** there will be a twenty minute time limit in effect.

PRE-TEST

To be filled in when full programmed environment is in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend the short (20 minute) lecture where the instructor will give a summary of those events that get you to the Console Flows. Also during this lecture it will be pointed out to you that you should keep in mind what you already know about the console operations. You have been Depositing, Examining, Load Addressing, etc. for quite awhile; do not forget what the results were in each of these operations. The instructor will give you an overview of the two Flow Diagram pages concerned with the console operations (Sheets 11 and 12 of the Flow Diagrams, PDP 11/40 System Engineering Drawings). Hold your questions until the lecture is over and at that time they will be answered and discussed.

## II. PROJECTS/QUESTIONS

## A. Flow Diagram (Console Loop Sheet 11)

The purpose of this Flow is to provide the processor a place from which to start (after a power up or initialize) and/or a place to which he goes when a Halt instruction is executed or the Halt Switch is depressed. It is here the processor must be if you ever want to do a Load Address, Exam, Deposit, Continue, or Start.

Let us analyze and then summarize each of these micro instructions to verify that Page 11 does do its job.

1. The entry at the top right brings us into this page and we start things off by executing CON 05, a NO-OP. You then go to CON 13 where some interesting things begin to happen.
  - (a) BUT SWITCH - is there a control switch depressed?
  - (b) CONSL ← 1 - Console F/F gets set.

This BUT is referred to as a Working BUT because it serves two purposes, Branch Testing and setting a F/F. Assuming no switch is activated at this point we now go on.

2. CON 06 then CON 04, CON 06 etc. is known as the Console Loop and here is where we will stay (continually executing these two micro instructions) until such time that a control switch is depressed (activated) then we will go to CON 07.
3. Try to think of the BUT 06 in address 026 as asking this question. BUT where should I go when I'm done executing CON 06 - to CON 07 or back to CON 04? If switch then to CON 07, if not switch then to CON 04.
4. Upon leaving the Console Loop (026, 046, 026, etc.) the next three micro instruction have to do with contact bounce. There are two situations which you must consider; doing this dynamically or with the maintenance module. First, let us do it dynamically:
  - (a) CON 07 - Put some number into the D register as a function of SBC = 14. Determine this number by referring to your SBC table.
  - (b) CON 08 - Move that number to R[TEMPC] and ask this question; is D = 0? The answer - no. What this means is after I am done with CON 09 go back to CON 08.
  - (c) CON 09 - Increment this count by adding 1 to the value in R[TEMPC] then restore R[TEMPC] with this new value.
  - (d) Go back to CON 08 and continue to do this loop until D register does = 0, then go to CON 10. The time involved here is approximately 40 ms.

Now let us analyze these same micro instructions when using the maintenance module. (The maintenance module allows you to execute one micro instruction and stop, one micro instruction and stop, etc.).

- (e) CON 07 - The number put into the D register is zero because we're using the maintenance module.

- (f) CON 08 - When I ask the question this time, D is = to 0 and when I am finished with CON 09, which is next, I go to CON 10 immediately.
- (g) CON 09 - no meaningful operation this time.

Read the BUT Definitions on this sheet also, to substantiate your beliefs.

- 5. CON 10 provides for saying; BUT which of the various switches that you could have activated did you activate? In other words, the BUT 30 allows the activated switch to implement modifying the base address 030, such that we will go to the appropriate flow as we leave this page.
- 6. CON 11 - Puts the last Console Address in the D register as we leave this page.

B. Flow Diagram (Console Switches Sheet 12)

The purpose of this sheet is to set forth the action taken whenever a particular control switch has been activated. First we will summarize some of the operations then you will be given the opportunity to go through some on your own then some questions will be put to you to reinforce those things learned.

For purposes of discussion, let us assume that we are sitting in the Console Loop (026, 046) and the job we are required to do is (a) Load Address 500, and (b) Deposit a MOV #10, %R0, (c) Examine what you Deposited.

- 1. Set the number 500 into the switch register on the 11/40 Console. Activate the Load Address switch by depressing it and the micro program will react as previously discussed and as you come off sheet 11 you go to LAD00, 037 Sheet 12 and this is where we will pick up the action.
  - (a) LAD00 - Get the information from the switch register. First, create the switch register address (SBC = 10) and put it into the Bus Address Register. Then do a Data In Bus transaction to get the (500) information. Shut off the clock waiting for the Slave Sync to inform you (by starting the clock)

that you can accept this data.

- (b) LAD01 - Accept the 500 coming in from the switch register on the Unibus Data lines and put it into the General Purpose Register, R[ADRSC]. Notice the Exam and Deposit F/F both get cleared at this time.
- (c) LAD02 - Put the Loaded Address Information LAD03 - (500) into the Bus Address Register as you go back to Console Loop.
- (d) Now that we are back in the Console Loop let us decide where the pertinent information is located:
  - (1) The Console Data Display\_\_\_\_\_.
  - (2) The Console Address Display\_\_\_\_\_.
  - (3) The contents of R[ADRSC]\_\_\_\_\_.
  - (4) The contents of R[PC]\_\_\_\_\_.

Refer to the appendix to verify your answers.

2. Now we are ready to deposit. First, set the MOV #10 %R0 (octal)\_\_\_\_\_ into the switch register. Second, activate the deposit switch. We come off Sheet 11 in the same manner as before and proceed to DEP00, 034 this time due to the deposit switch. Let us pick up the action there.

- (a) DEP00 and 01 - Makes sure that the last loaded address gets put into BA register and in DEP 01 asks the question - is this a Register Deposit? In other words, are we depositing into a GPR? The answer now is no. So when we leave DEP02 will go to DEP03 and not bypass it.
- (b) DEP02 and 03 - Conditional plus 1 micro-instructions As you know, after the first deposit you must be prepared to increment. Two words are used because the incrementation if a General Register is by 1. The condition is whether the Deposit F/F is set or not and as you recall we cleared it in the Load Address Operation, therefore, it is still clear. You read this micro instruction as D and BA get the contents of R[ADRSC] plus something, that

something is as a function of  $SBC = 7$  and depends on the state of the Deposit F/F. If set add 1, if clear add 0. It becomes obvious then that if the Deposit F/F is set when I come through here incrementation will take place and when clear no incrementation.

- (c) DEP04 - Get the information I want to Deposit. Do a Data In from the switch register (that is where it is at).
- (d) DEP05 - Accept this data and put it into the B register.
- (e) DEP06 - Put the address I want to deliver the data to into the BA register.
- (f) DEP07 - Put the Data I want to deliver into the D register. The question is asked here again - is this a register I'm depositing - (no Data out would be required) the answer is still no, however. Notice the setting of the Deposit F/F here.
- (g) DEP08 - no-op needed because of the branch decision.
- (h) DEP09 - Do the Data Out Transaction to deliver the data to the proper address. Shut off the processor clock, wait for SSYN to inform you that memory has received the data then start up and go back to the Console Loop.
- (i) Again, let us analyze the pertinent data now that we are back in the Console Loop.
  - (1) The Console Data Display\_\_\_\_\_.
  - (2) The Console Address Display\_\_\_\_\_.
  - (3) The contents of R [ADRSC]\_\_\_\_\_.
  - (4) The contents of R [PC]\_\_\_\_\_.

3. Now you deposit the proper data at the next location and list the procedure you followed, the micro instructions executed, and answer the following questions.
- (a) After executing DEP 02 the Data Display is \_\_\_\_\_ and the Address Display is \_\_\_\_\_.
  - (b) When PUPP is 066 what is the Data Display \_\_\_\_\_; the Address Display \_\_\_\_\_.
  - (c) If this deposit had been a General Purpose Register, what micro instruction would have put the data into the GPR?
  - (d) How would the GPR have been selected in the previous question?
4. Let us discuss the Examine micro flow. After you finished depositing the MOV #10, %R0 you would have Loaded Address back to 500 (same manner as previously discussed) and now activate the Examine Switch. The Examine is pretty much the opposite of Deposit with the exception of a couple of micro instructions. I will discuss these and you check out the rest of the flow until you are satisfied you know how the Examine works.
- (a) EXM 05 - If this were a General Purpose register you were Examining then the low order 4 bits of the BA register would be used to provide the address. All the GPR's 7777XX.
  - (b) EXM 06 - If it is not a register examine start your Data In Bus transaction at this time and shut off the CPU clock. Also start your NO DAT timer and put

the SW address into D register. Now, if you get a time out (15  $\mu$ sec and no SSYN response, typically, trying to examine non-existent memory) then the JAMUPP logic will move you directly from here to the Console Loop. You will display in the Data lights the switch register address number 177570. This will be your time out indication flag. Of course, if you do not get a time out, you complete the Examine flow properly and go back to the Console Loop.

- (c) Some questions about the Examine Micro Routine.
- (1) The BUT 04 in EXM 01 modifies 056 to \_\_\_\_\_ if not examining a general register.
  - (2) In which micro instruction does the Examine F/F get set?
  - (3) What route does the Data take to accomplish EXM 06?

Refer to the appendix to verify the correct answers.

5. After examining the contents of location 502 and Loading Address to 500 again let us take off the Halt Switch and depress Start. Analyze the sequence that would be taken and answer the following questions:

- (a) What address do you go to when leaving Sheet 11 \_\_\_\_\_?
- (b) What turns off the console light?
- (c) As you leave Sheet 12 what is the contents of:
  - (1) R [PC]
  - (2) D register

- (d) As you leave Sheet 12 what sheet and address do you go to?

### III. SELF EVALUATION

Answer the following questions as an overall evaluation of the knowledge obtained in this unit. When you are finished check to see if you agree with those given in the appendix.

- A. In CON 07 what number will be put into the D register if not using the maintenance module?
- B. If, after the bounce count delay, there is no switch depressed what micro instruction is executed after CON 11?
- C. The Unibus Data that is being accepted in LAD 01 originated where?
- D. After completion of a deposit of 10 into General Purpose Register R0 give the Address Display and the Data Display.
- E. An Examine of a memory location requires how many and what type of bus transactions?

### IV. ADDITIONAL READING

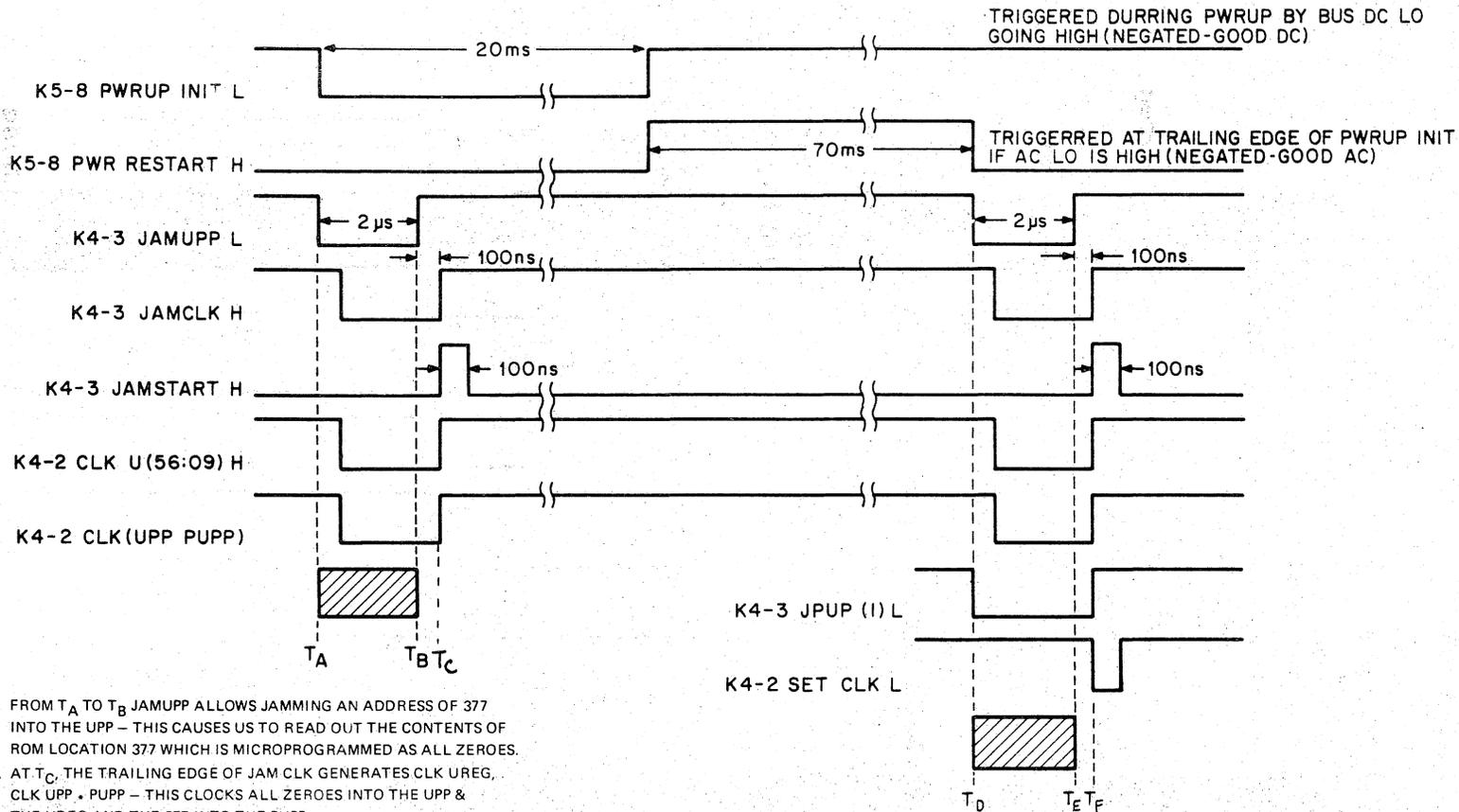
Operation Symbols, paragraph 4.2.5, Page 4-11 of the KD11-A Processor Maintenance Manual, DEC11-HKDAA-A-D will provide some supplementary information which you may find useful at this time.

#### WHAT'S NEXT

After completing this unit you should have a good idea of what a flow operation looks like. In the subsequent units you will be analyzing how the micro program accomplishes the execution of machine instructions.

#### SUPPLEMENTARY INFORMATION

If you feel the need for more practice concerning these console operations ask the instructor for additional assignments.



1. FROM  $T_A$  TO  $T_B$  JAMUPP ALLOWS JAMMING AN ADDRESS OF 377 INTO THE UPP – THIS CAUSES US TO READ OUT THE CONTENTS OF ROM LOCATION 377 WHICH IS MICROPROGRAMMED AS ALL ZEROES.
2. AT  $T_C$ , THE TRAILING EDGE OF JAM CLK GENERATES CLK UREG, CLK UPP + PUPP – THIS CLOCKS ALL ZEROES INTO THE UPP & THE UREG AND THE 377 INTO THE PUPP.
3. THE JAMSTART PULSE IS INHIBITED FROM STARTING THE CLOCK BY PWRUP INIT STILL BEING ACTIVE.
4. SINCE THE UREG CONTAINS ALL ZEROES WE HAVE FORCED THE MACHINE TO A KNOWN STATE – ALL CONTROL SIGNALS INACTIVE (LOCATION 000 IN THE ROM IS BEING READ OUT, BUT THE SUBSEQUENT JAM SEQUENCE WILL ALTER THE UPP PRIOR TO CLOCKING THE UREG AGAIN)
5. AT  $T_D$  THE TRAILING EDGE OF PWR RESTART SETS THE JPUP FLOP.
6. JPUP TRIGGERS THE JAMUPP ONE-SHOT AND FROM  $T_D$  TO  $T_E$  WE JAM A NEW ADDRESS INTO THE UPP. [IF HALT; UPP = 30 (CON05)]  
[IF – HALT; UPP = 337 (TRP17)]

7. ASSUME THE HALT SWITCH WAS ON; AT  $T_F$  THE TRAILING EDGE OF JAM CLK GENERATES CLKU, CLK(UPP.PUPP) – THIS CLOCKS THE CONTENTS OF LOC 30 INTO THE UREG, 315 INTO THE UPP, AND 30 INTO THE PUPP – WE NOW HAVE THE MICROPROGRAM IN THE CONSOLE ROUTINES.
8. THIS TIME, THE JAMSTART PULSE GENERATES THE SET CLK SIGNAL TO TURN ON THE PROCESSOR CLOCK – THE MICROPROGRAM SHOULD NOW SEQUENCE TO CON06 AND START LOOPING, WAITING FOR A CONSOLE CONTROL SWITCH.

11-1664

KD11-A JAMUPP Timing Sequence

APPENDIX

## I. ANSWERS TO PROJECTS/QUESTIONS

B. 1. (d)

- (1) zero
- (2) 000500
- (3) 000500
- (4) unknown

2. (i)

- (1) 012700
- (2) 000500
- (3) 000500
- (4) unknown

3. (a) 000501, 000501

(b) 000010, 177570

(c) DEP 10, 073

(d) SRBA (1), BA 03:00

4. (c) 1 = 57

2 = EXMØ4

3 = -REG.

5. (a) 032

(b) BUT 10

(c)

- (1) 000500
- (2) 000500

(d) 1, 016

## II. ANSWERS TO SELF EVALUATION

A. 000020

B. CON 05

C. AT SWITCH REGISTER

D. ADRS (777700), DATA (000010).

E. 1, DATA IN.



11/40 KD11A 060

LEARNING ACTIVITY UNIT

FLOW DIAGRAMS III

MICROPROGRAM WORKSHEETS AND SINGLE OPERAND INSTRUCTIONS

1/21/74



RATIONALE

In your first unit, you learned how to read the flow diagrams. Then you used that ability to go through the Console operations and interpret those flows. Now that you have accomplished that, you should have a firm understanding of the way the 11/40 does things. Also, what the flows look like to accomplish these operations. Now, it is time to go on.

In this unit, you will discover how the 11/40 executes machine instructions. Specifically single operand (simple) instructions. To assist you with this execution, you will be introduced to the microprogramming worksheets. Your understanding and use of these worksheets will be a valuable tool in the learning and the understanding of the 11/40.

Once you understand how the 11/40 accomplishes simple machine instructions, it becomes much easier to progress to the more difficult ones. Knowing how to implement the microprogram worksheet and then using that as a major learning tool will greatly enhance your knowledge and understanding of the 11/40.



PREREQUISITES

Satisfactory completion to date of all previous units.

OBJECTIVES

Given a single operand machine instruction, you must fill out a microprogram worksheet. To demonstrate successful completion of this unit, you must correctly identify at least 80% of the required items. You are allowed to use any and all reference materials at your disposal, however, a twenty minute (20) time limit will be effect. You are required to complete within this time frame.

PRE-TEST

To be filled in when fully programmed environment in progress.

## LEARNING ACTIVITIES

## I. LECTURE

Attend a short lecture where the instructor will introduce you to the microprogram worksheet. He will identify the format and point out to you some of the ways it can be beneficial to you as it applies to the 11/40. He will also stress that first you must decide how a particular machine instruction works, then take it through the 11/40 Block Diagram, and finally to the flows to verify and understand how the 11/40 accomplishes it. Please hold your questions until the discussion period at the end of the lecture.

## II. PROJECTS/QUESTIONS

A. The instructor will pass out microprogram worksheets to everyone and before filling in any information, will summarize the micro flow for the HALT instruction. Then together, the microprogram worksheet will be done.

1. Let us assume that we had deposited all zeros into memory location 500 (a HALT instruction). Then, after having depressed the START switch, we finally arrive at FETCH C, Flow Diagram Sheet C, PDP11/40 System Engineering Drawings.

2. At this point, I think it might be a benefit to first decide what sequence we will be taking, then come back and attempt an explanation. The sequence is:

a. FET 02

b. FET 03

c. FET 04

d. FET 05

e. CON 00

f. CON 12

g. CON 02

h. CON 03

i. CONSOLE LOOP

Now list this sequence on your microprogram worksheet under the column - name.

3. FET 02

In this micro instruction, we are fetching the machine instruction. Notice we put the address from where we want the FETCH to occur in the BA Register and start a Data in BUS TRANSACTION

4. FET 03

Accept the incoming instruction and put it three places IR, B, R[IR].

5. FET 04 & 05

In terms of the machine instruction, these two micro instructions look at it to see where to go after FET 05 (by doing the BUT 37). The additional accomplishment by FET 04 & 05 is updating the PC.

6. CON 00

Base address 100 gets modified to 122 because when the BUT 37 allowed the IR decode conditions to affect the base address, Bit 1 and Bit 4 were true. Refer to the appendix, and the instructor will put a visual via the overhead projector of the "Micro Branch Timing Simplified Overview for HALT" to assist in your understanding of the branching implementation.

7. CON 12

Move the Processor Status Word to the D Register. Set the Console F/F (light comes on) and clears the examine & Deposit F/F's. Do a BUT 24 - what brought me here a HALT Instruction or a HALT Switch - this time a HALT instruction therefore, when done with CON02 go to CON03 before dropping into the Console Loop. Allows the fact of HALT switch to modify Bit 1 of base address 024 if necessary.

## 8. CON 02

The main purpose of this micro instruction is to insure that the Processor has got control of the Unibus. The way it is accomplished is by turning off the CPU clock (IDLE (1)) then, setting the F/F await BUS Busy (AWBY (1)) and when the Processors BUS Busy F/F becomes Set - restart the clock and continue on. Now, if the Processor has the BUS; this will only be a slight hesitation but if not, he will stop here and wait until he does have it before continuing on.

## 9. CON 03

Take the contents of R0 and put it into the D Register, so that it will be displayed in the Console Loop.

10. CON 04 >  
CON 06 >

Console Loop. Stay here displaying R0 looping between CON 04 and CON 06 until a Control Switch is sensed and you start some future action.

Now that you have finished this first project together, it is time for you to do one on your own.

- B. Take a blank micro program worksheet and fill in the information as you just did. The instruction and pertinent information is given below. Start at FETCH C, Flow Diagram 8. Verify your answers with those given in the appendix.

GIVEN:	500/INC (R1)	R1=1000
	502/HALT	1000/001234

GOOD LUCK

- C. Take a few minutes now and investigate how a decrement would have worked as compared to the increment you just completed. Assume the same information given.

### III. SELF EVALUATION

Complete a micro program **worksheet** on the following instruction. When you finished, check your answers with those given in the appendix. Check with the instructor as necessary to clear up any misunderstanding. Start this worksheet at FET02, Flow Diagram 1, and go all the way through the sequence and back to FET 03.

GIVEN; 500/BR SELF  
502/HALT

### IV. ADDITIONAL READING

There is no specific reading associated with this unit, however, chapter 4 in the KD11A Processor Manintenance Manual DEC11 HKDAA-A-D provides information on the PDP11/40 Flow Diagrams.

### WHAT'S NEXT

After completing the unit, you should have a pretty good feel for the flows. In the next unit, you will continue with the flows tackling more difficult machine instructions. (double operand - type 1)

### SUPPLEMENTARY INFORMATION

More practice worksheet assignments are available from the instructor if you feel the need.

EXAMPLE:						
		500/INC (R1)			R1 = 1000	
		502/HALT			1000/001234	
NAME	PUPP	UPP	UPF	DATA DISPLAY	ADDRESS DISPLAY	COMMENTS
FET02	016	001	004	500	--	Fetch the instruction
FET03	001	004	005	005211	500	Accept the instruction
FET04	004	005	100	500	500	Update the Program Counter
FET05	005	161	266	502	502	
DST00	161	266	267	1000	502	Get 1000 to the BA and do a DATIP
DST14	266	267	220	005211	1000	But to see if ODD Byte if not, where to off this page
DST15	267	220	211	001234	1000	Accept Operand
SSL06	220	211	367	502	1000	Increment the (dest) let the machine instr. tell the ALU what to do
SSL10	211	367	375	001235	1000	Altering the PS BUT 16 where to next
DOP12	367	375	016	001235	1000	
DOP20	375	--	--	001235	1000	

EXAMPLE:						
				500/000777		
				502/HALT		
NAME	PUPP	UPP	UPF	DATA DISPLAY	ADDRESS DISPLAY	COMMENTS
FET02	016	001	004	500		
FET03	001	004	005	000777	500	
FET04	004	005	100	500	500	
FET05	005	111	340	502	502	
BRA00	111	340	341	502	502	Take the sign extended IO Byte of B Register and add it to the PC
BRA01	340	341	016	501	502	But where do I go after BRA02
BRA02	341	016	001	501	502	Same as BRA00
FET02	016	001	004	500	502	
FET03	001	004	005	000777	500	



11/40 - KD11A - 070

LEARNING ACTIVITY UNIT

FLOW DIAGRAMS IV

DOUBLE OPERAND INSTRUCTIONS AND  
OVERLAP

1/21/74



RATIONALE

In the previous units concerning the 11/40 Flow Diagrams, you learned how to interpret and understand the Flows, and how they apply to simple operations. Hopefully, you have realized what the flows can do to help you understand the PDP 11/40 operation. You have been introduced to the microprogramming worksheet and realize how it can help you master the flows. It is time to go even deeper into the 11/40 Flow Diagrams and try more difficult operations.

This unit mainly deals with the flows handling Double Operand instructions and the concepts concerning Overlap. You will complete several worksheets using a couple of examples to assist you in understanding the unit; that is, the way the 11/40 does these 'more difficult' operations.

Once you have mastered these Flows, the rest of the 11/40 becomes very easy. This is the next to the last unit in the Flow sequence and incorporates most of the ideas learned in the first three. The Flows can answer almost any question concerning the 11/40 operation - to know the Flows is to know the machine.



PREREQUISITES

Satisfactory completion to date of all previous units.

OBJECTIVES

Given a double operand machine instruction you must correctly complete a microprogram worksheet. To demonstrate successful completion of this unit you must correctly identify at least 80% of the required items. You can use any of your notes and/or reference material, however, a twenty (20) minute time interval will be in effect. You are required to complete within this time.

PRETEST

To be filled in when full programmed environment is in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend a short (10 - 20) minute lecture where the instructor will define the conditions for Overlap. He will present the ideas, concepts and advantages of the Overlap situation. He will again stress that a knowledge of the machine instructions operation and its movement through the block is necessary before attempting a microprogram flow sequence. Please hold your questions until the end of this lecture when a question and discussion period will be held.

## II. PROJECTS/QUESTIONS

- A. The instructor will pass out a microprogram worksheet and everyone will work this first one together. The first example will incorporate the Double Operand instruction and the Overlap feature together.

GIVEN:

500/MOV %R0, %R1  
502/HALT

R0 = 000050

R1 = 000100

This instruction will move the number 50 which is in R0 to R1. Now let us fill out a microprogram worksheet to see how this is accomplished.

- B. Now turn to Page 4-12, Paragraph 4.3 in the KD11A Processor Maintenance Manual and read the example found there. Go through Flow Diagram Example I, Table 4-2, pages 4-13 through 4-16, thoroughly, step by step and answer the following questions when you are finished.
1. What UPF does the BUT(INSTR 1) work on?
  2. What Bits does the BUT 37 modify?
  3. What General Purpose Register does the Source Operand get put into?

4. In what microinstruction is the ADD really accomplished?
- C. Now go through Flow Diagram Example 2, Table 4.3, Pages 4-17, 4-18 in the KD11A Processor Maintenance Manual using the same procedure as above. Answer the questions below when you complete this project.
1. Why is R(SF) incremented by 2 in SRC 01?
  2. What ALU input does the Source Operand go into?
  3. What is the 'C' Bit after this instruction is executed?

### III. SELF EVALUATION

Complete a microprogram worksheet for the following instruction. Check your results with those in the Appendix.

GIVEN:

500/CMP (R1), (R2)

1000/000050

R1 = 1000

R2 = 2000

2000/000040

777776/000017

This instruction compares the contents of memory location 1000 to the contents of memory location 2000 and affects the condition codes in some manner. Remember the destination operand is not changed - in other words, no results are delivered to the destination address.

When you complete this successfully, you should be ready for the Post Test which will be similar type project.

IV. ADDITIONAL READING

The remaining pages of Chapter 4 in the KD11A Processor Maintenance Manual contain a listing of all the 11/40 microinstructions; you may find this useful.

Any information about the instruction operation you find necessary to review can be found in the 11/40 Processor Handbook, Chapter 4. Feel free to look at this any time.

WHAT'S NEXT

After completing this unit, most of the major Flow Diagram operations are complete. In terms of the instruction operation, you are finished, therefore, the next unit will deal with the Traps and Service Flow Diagrams and that will complete the section dealing with the Flows.

SUPPLEMENTARY INFORMATION

More practice instructions are available from your instructor should you want or need more practice.

APPENDIX

ANSWERS TO II:

- B. 1. 100
- 2. BIT 5, BIT  $\emptyset$
- 3. R11
- 4. DOP 03, 225
  
- C. 1. R(SF) equals 7, PC
- 2. B input
- 3. Cleared, there was a carry.



11/40 KD11A 080

LEARNING ACTIVITY UNIT

FLOW DIAGRAMS V

TRAPS/SERVICE

1/21/74



RATIONALE

This is the last of the units concerning the PDP11/40 Microprogram Flow Diagrams. To date, you have learned how to read, interpret, use and understand the Flows. You are prepared to take any unknown or unfamiliar instruction and determine how the 11/40 accomplishes the execution of it by using your knowledge of the microprogram Flow Diagrams.

One of the most important aspects of any of these Processors: the 11/05, 11/40 and the 11/45 is how they interface to the Unibus, specifically how they handle priority transactions (servicing) and the procedures taken when error conditions exist (trapping). This unit will be concerned mainly with these two groups of microprogram Flow Diagrams.

Once you understand how the 11/40 handles these bus transactions, you can relate it to your knowledge of the 11/05. You will also have a better understanding of the Unibus and this unit will also help you later on with the peripherals. Although this is the last unit of the Flows, it may very likely be the most important one of the sequence.



PREREQUISITES

Satisfactory completion to date of all previous units.

OBJECTIVES

Given four multiple choice questions concerning the Service and the Trap Microprogram Flow Diagrams, you must choose an answer from four possibilities. To demonstrate successful completion, you must correctly answer three of four questions.

You will be able to use all available references but you must complete the test in 20 minutes.

PRETEST

To be filled in when fully programmed environment is in progress.

## LEARNING ACTIVITIES

## I. LECTURE

Attend a 40 minute lecture where the instructor will give a brief review of the priority transactions on the Unibus, namely BR's and NPR's. A summary of the BR and NPR timing will also be given. An overview of the Trap Micro routines will be given with particular emphasis on error handling, and a brief look at the Service Micro routines with the emphasis on priority transactions. Hold your questions until the lecture is over when there will be a short discussion period.

## II. PROJECTS/QUESTIONS

- A. The entry to the Service Flows, sheet 10, PDP 11/40 System Engineering Drawings, when doing a WAIT instruction is Service A, which you can readily determine. Verify that the WAIT Loop, then, consists of the following group of micro instructions. (assume nothing of a higher priority is pending)

SER00	114
SER05	015
SER06	012
SER07	020
SER08	021
SER02	017
SER05	015
etc	

Notice the micro instructions across the top of this sheet. All have in them a BUT Request. Check note 2 on this sheet for a complete definition.

You will stay in this WAIT Loop until such time that you get a BR - at that time you will drop through to SER09. Refer to "Additional Readings" if you feel more review on priority transactions is needed.

- B. Once a BR is received, arbitrated and a BG is issued, you go to SER09 shut off clock and wait for the interrupt to arrive informing you that the vector information is on the "D" lines.

Trace the steps you now take - pushing onto the stack your current PC and PSW and obtaining a new PC and PSW having been given this vector address. The sequence should correspond with the following.

SER10	002
SER11	023
TRP08	007
TRP09	115
TRP10	326
TRP11	327
TRP12	113
TRP13	330
TRP14	331
TRP15	077
TRP16	140
TRP20	332
TRP21	333
SER03	123
SER05	015
FET00	013

- C. Verify that, under the condition of Red Zone overflow and Double Bus Error, the Trap sequence would be altered as such.

TRP00	336
TRP01	317
TRP02	215
TRP09	115
etc	

### III. SELF EVALUATION

Answer the following questions and check your results with the answers found in the appendix.

- A. After arriving at the Service Flows and doing a BUT 26, you find two requests pending.

1. BERR

2. BR

Which would be serviced first?

- B. What is the next micro instruction executed after SER05 if you have a yellow zone overflow.
- C. What is displayed in the Console Data lights when you have executed a WAIT instruction and you are in the WAIT Loop?
- D. What location is the new Processor Status obtained from?
- E. If this BR and Interrupt Sequence had been set off due to a Teletype Key being depressed what would the Console Data lights read when PUPP equals 332?

#### IV ADDITIONAL READING

To reinforce your understanding of the priority transactions on the Unibus, you can read paragraphs 5.6 thru 5.10, pages 5-17 thru 5-34 in the PDP11 Peripherals Handbook and/or paragraph 3.2.2 and corresponding figures, pages 3-2 thru 3-12 in the KD11A Processor Maintenance Manual. General Unibus theory and operation explanations are found in chapter 5, pages 5-1 thru 5-17 of the PDP11 Peripherals Handbook.

#### WHAT'S NEXT

After completing this unit, you are finished with the microprogram Flow Diagrams. You now have the ability (given some machine instruction) to take it through the Flows intelligently describing the events that take place along the way. After a Lab period to reinforce the flows and the Maintenance Module usage, you will be ready to go into the Engineering Drawings.

#### SUPPLEMENTARY INFORMATION

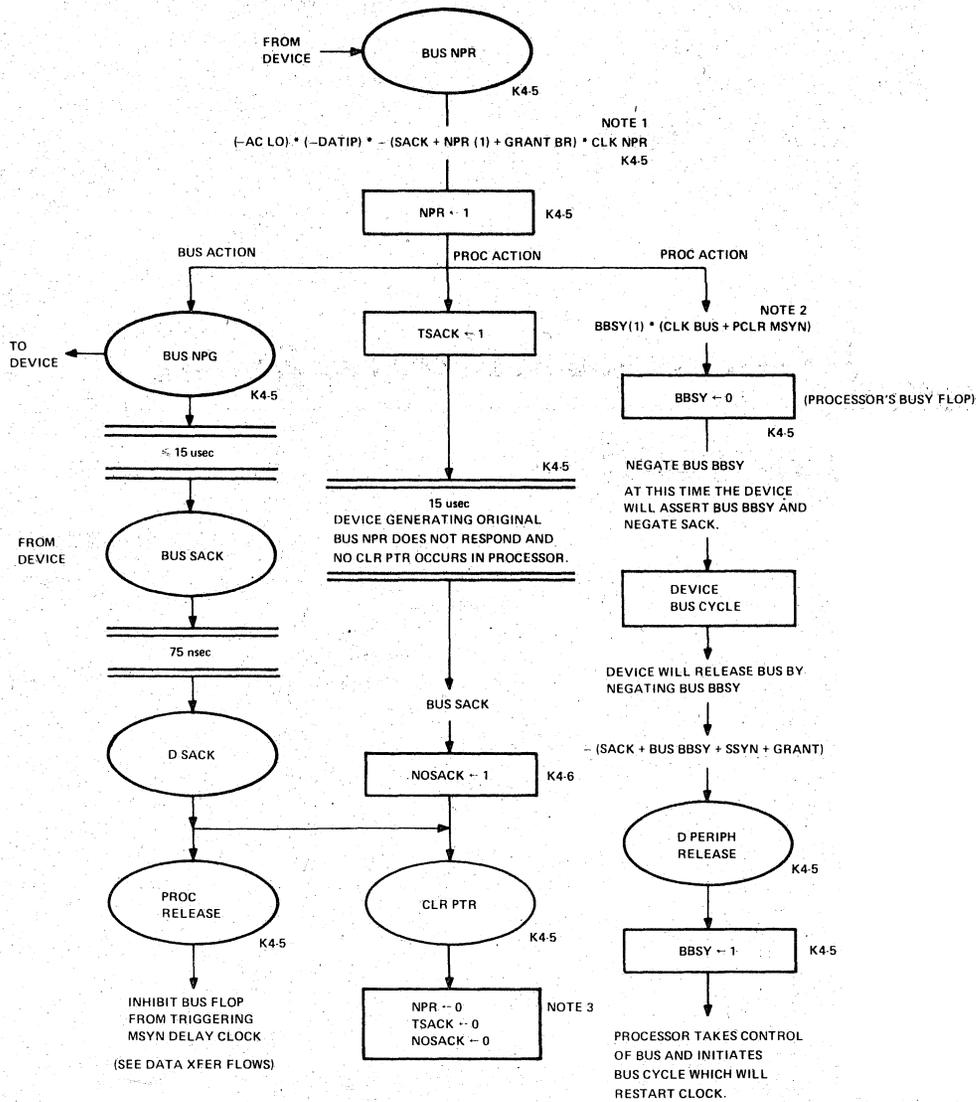
If you feel the need for more practice with the flows or more information concerning the operation of PDP11/40 consult with your instructor for additional assignments.

APPENDIX

Answers to self evaluation

- A. BERR
- B. TRP03
- C. The WAIT instruction inclusively ORed with the updated PC
- D. The vector address plus 2
- E. 060

APPENDIX

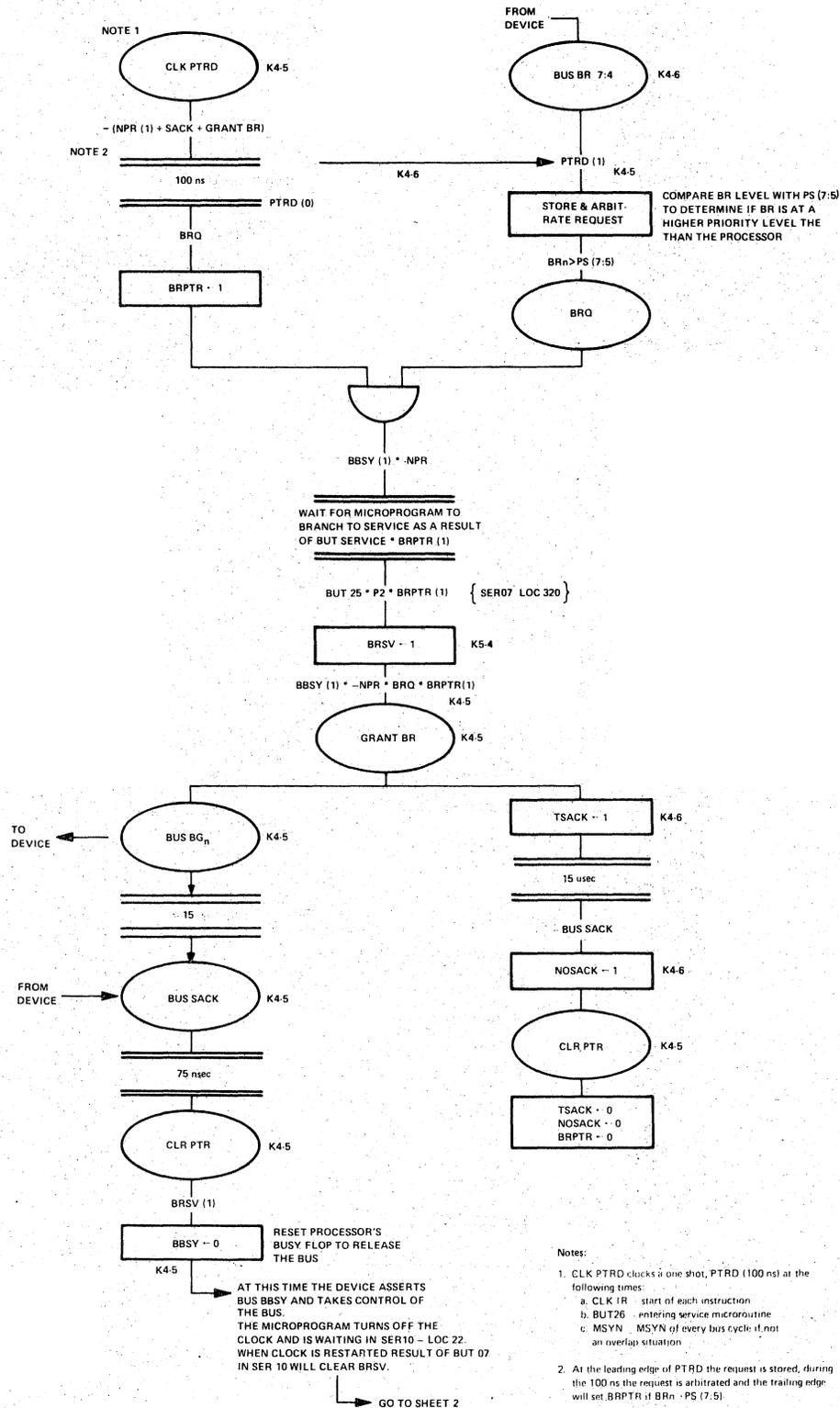


Notes:

1. NPRs are clocked frequently to determine if NPR Service needed. NPG occurs.
  - a. BGBUS (1) -- microword specifies a data transfer
  - b. ENPR CLK -- from EIS/FIS option
  - c. CLK IR -- microprogram in FETCH
  - d. BUT26\*P3 -- microprogram just entered SERVICE
  - e. --AWBY\*SET CLK -- clock restarting after bus cycle
  - f. P MSYN -- A device has asserted BUS MSYN.
2. Microprogram either just starting or finishing a data XFER Bus Cycle or in Service when processor gives up the bus to the NPR device.
  - a. CLK BUS -- just starting processor Data XFER -- clock will be turned off in this U word or succeeding 1 word.
3. SACK timeouts will not cause a trap but will generate CLR PTR which cleans up the NPR Priority XFER control flip-flops, and will restart Processor Clock.
- b. PCLR MSYN just finishing a Data XFER Bus Cycle -- clock is on but will go off when microprogram reaches next Data XFER microroutine.
- c. SERVICE -- give up the bus to do NPRs on BUT26 and on AWBY.

11-1680

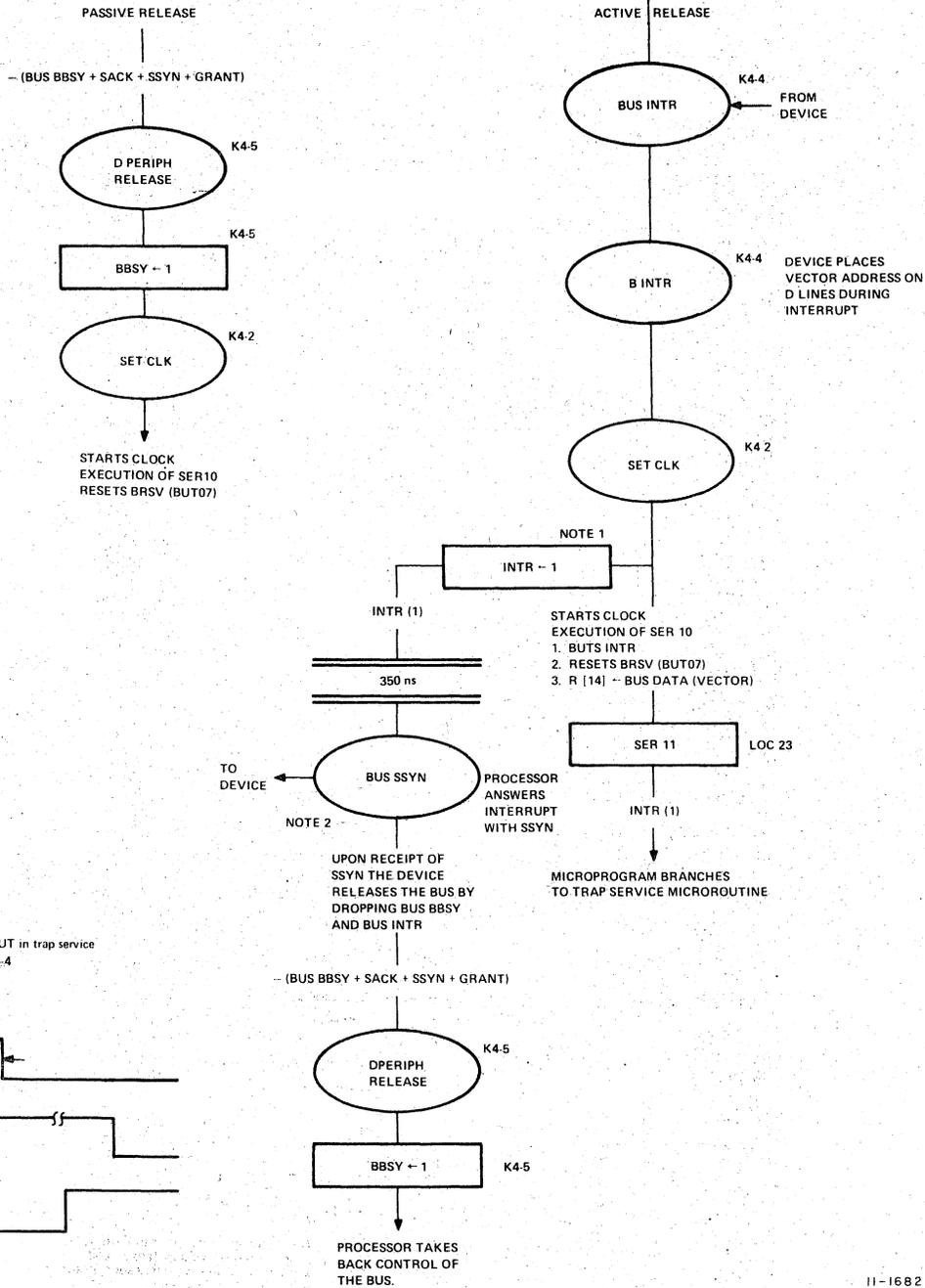
KD11-A Priority Transfer Timing and Control for NPRs



11-1681

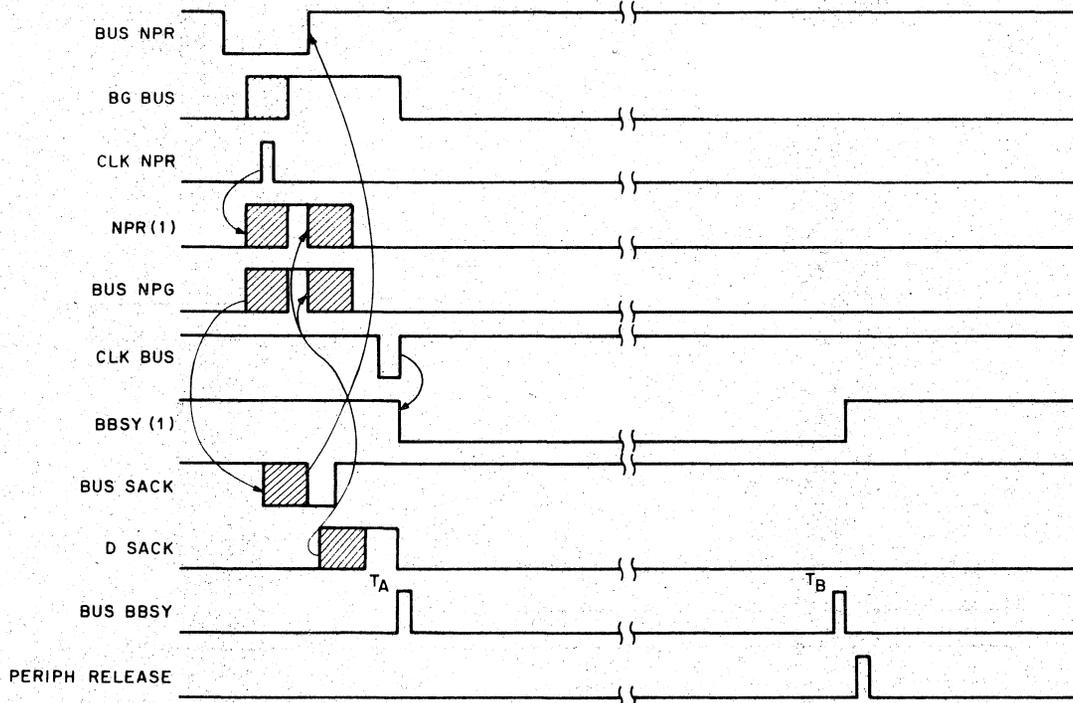
KD11-A Priority Transfer Timing and Control for BRs (Sheet 1 of 2)

DEVICE HAS CONTROL OF THE UNIBUS AS A RESULT OF A BR TRANSFER



11-1682

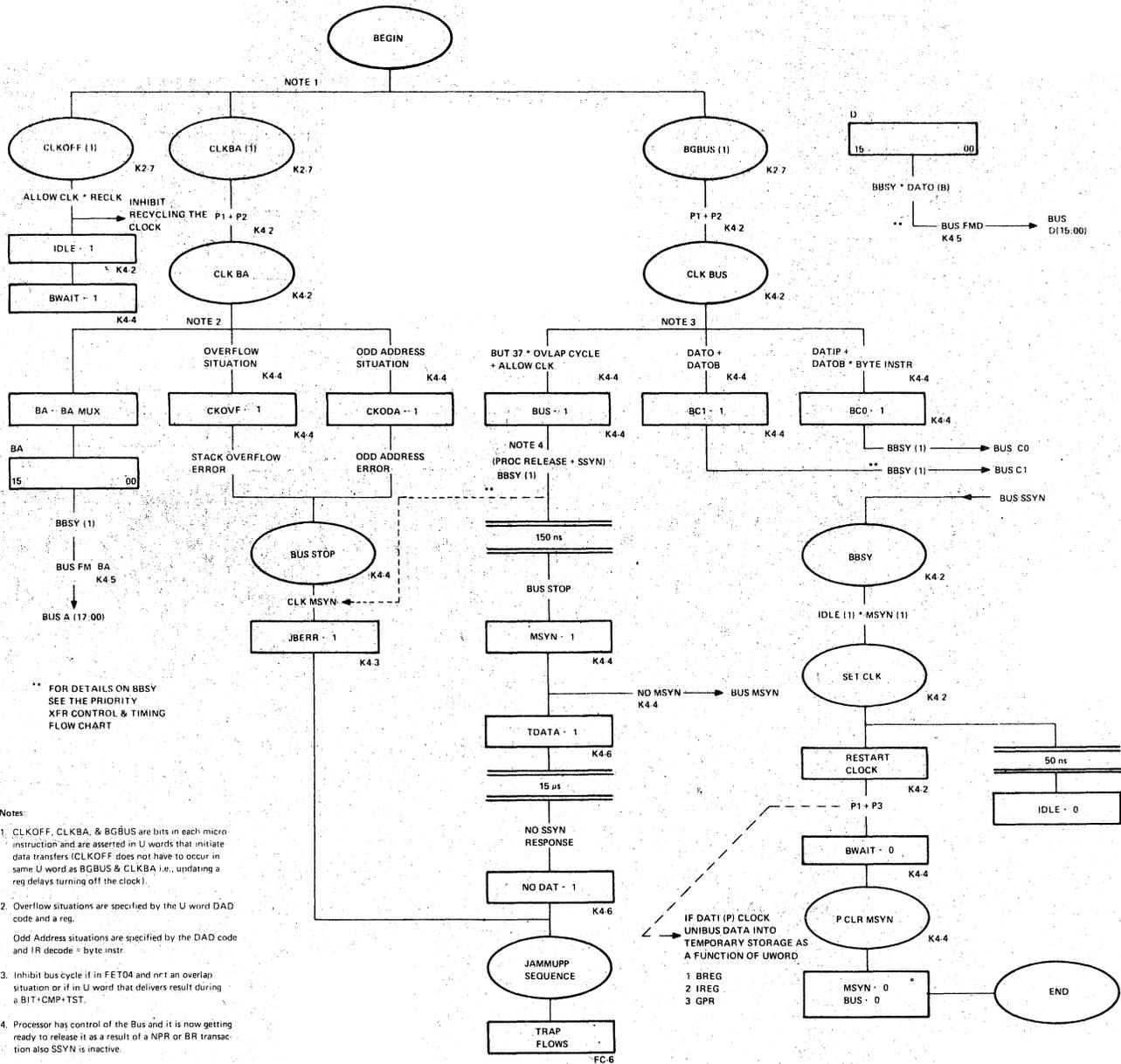
KD11-A Priority Transfer Timing and Control for BRs (Sheet 2 of 2)



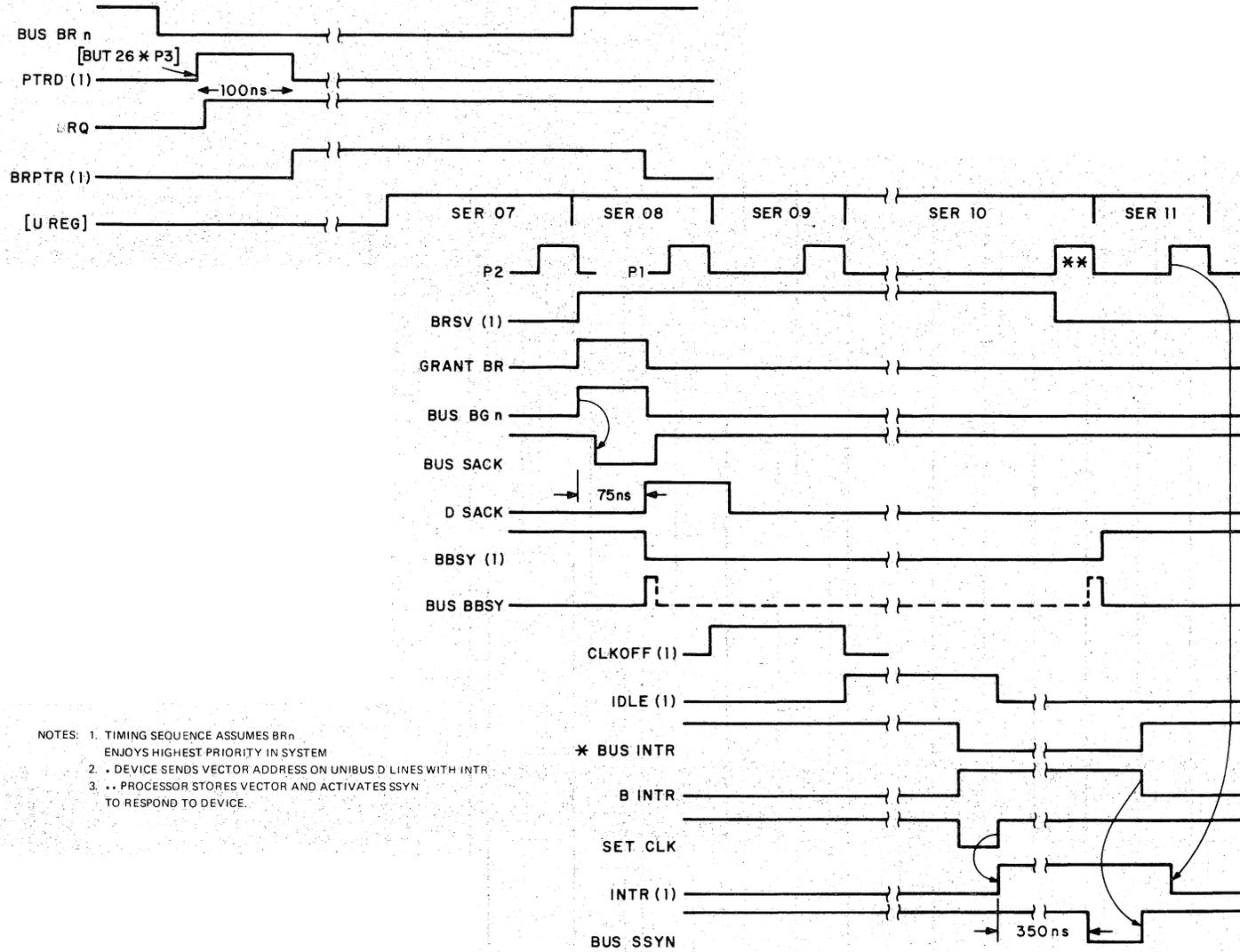
NOTE:  
 1. Setting NPR conditional upon - (DATIP+B AC LO)  
 2. From  $T_A$  TO  $T_B$  the NPR device has control of the Unibus

11-2125

Figure 3-3 NPR Priority Transfer Timing Sequence



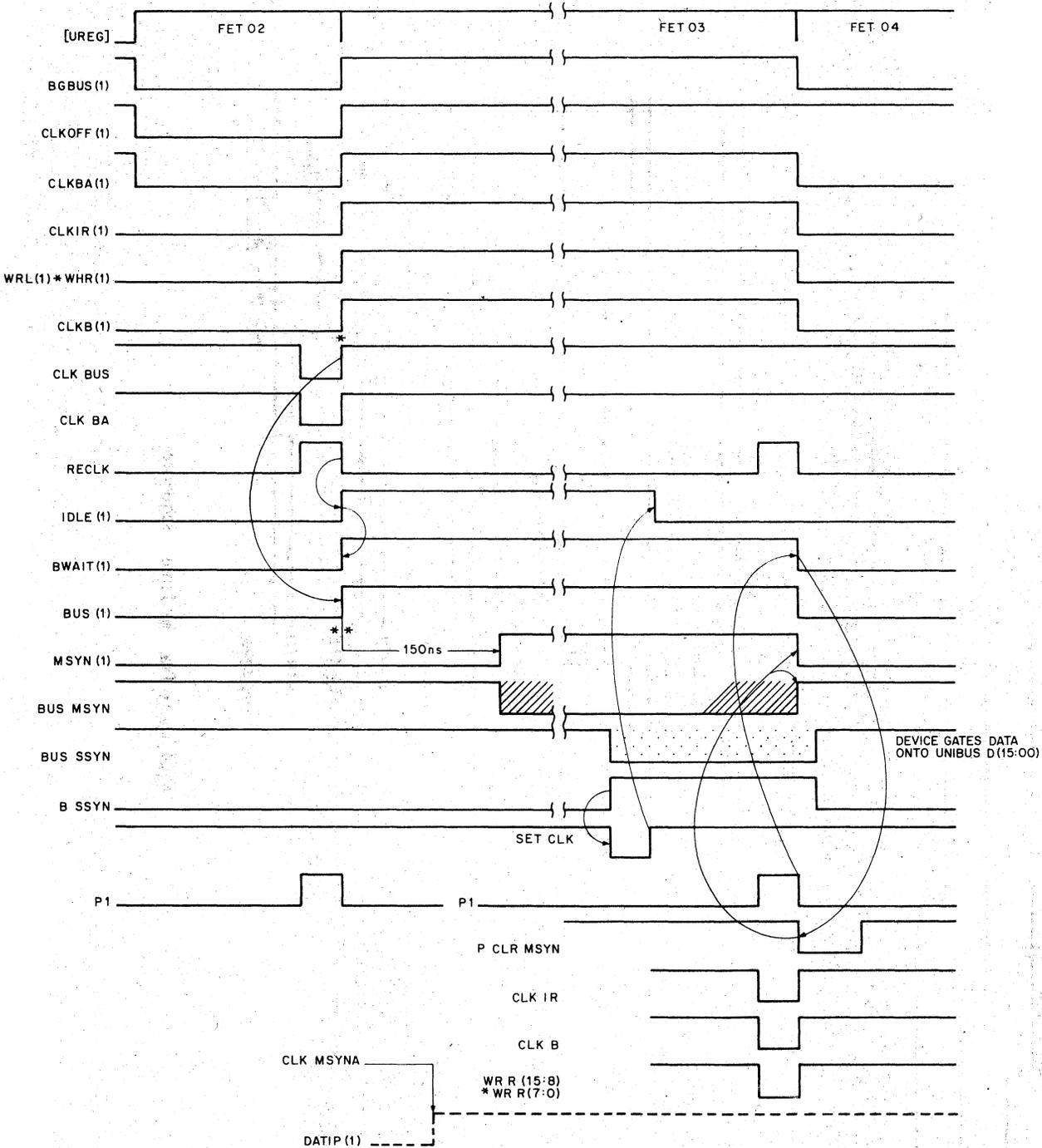
KD11-A Bus Data XFER Timing and Control



- NOTES: 1. TIMING SEQUENCE ASSUMES BR<sub>n</sub> ENJOYS HIGHEST PRIORITY IN SYSTEM  
 2. \* DEVICE SENDS VECTOR ADDRESS ON UNIBUS D LINES WITH INTR  
 3. \*\* PROCESSOR STORES VECTOR AND ACTIVATES SSYN TO RESPOND TO DEVICE.

11-1676

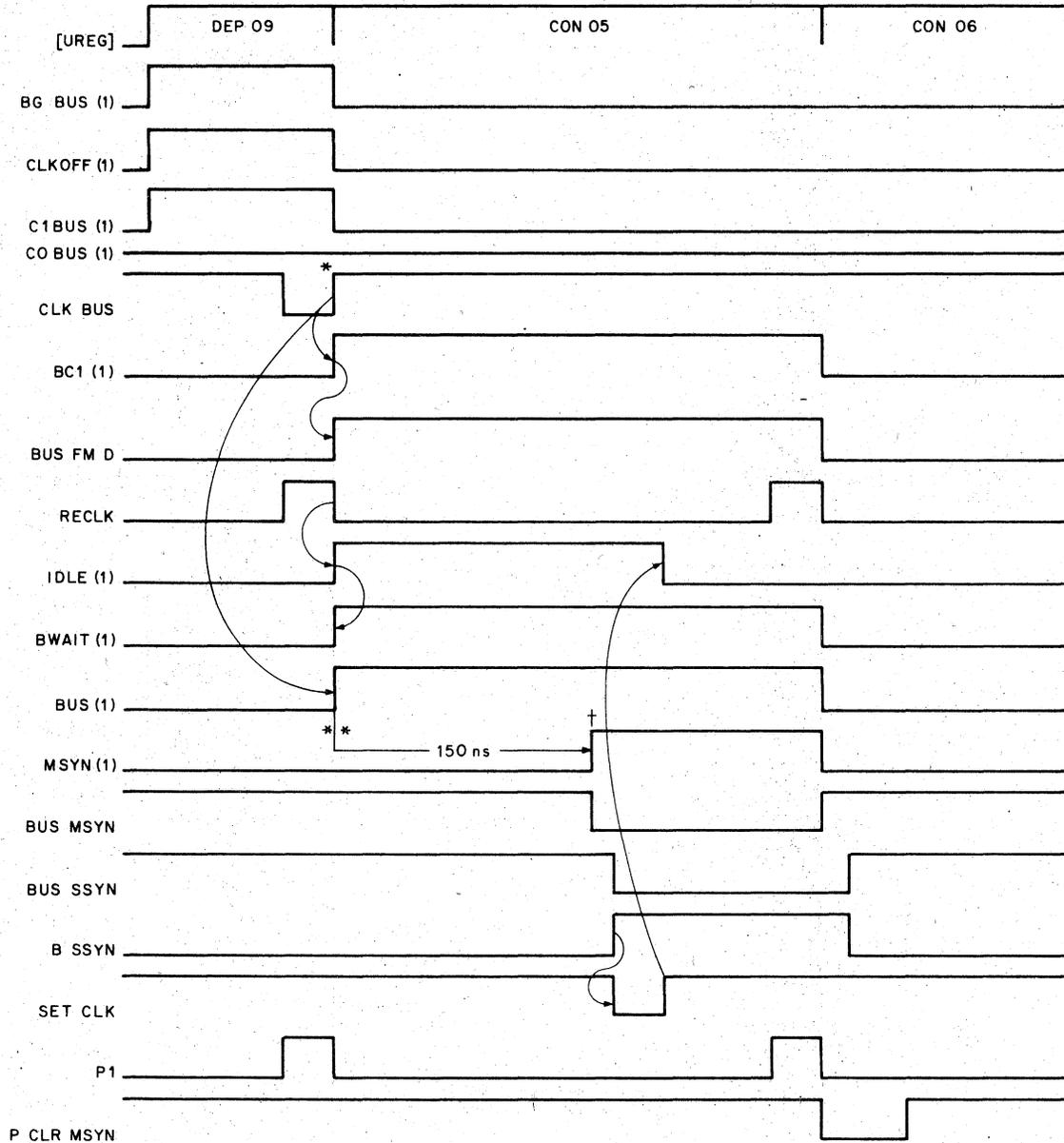
BR Priority Transfer Timing Sequence



- NOTES: 1. EFFECTS OF GATE DELAYS AND F.F. RESPONSE NOT SHOWN  
 2. ASSUMES KD11-A HAS CONTROL OF THE UNIBUS [BBSY (1)]  
 3. \* CLOCK NPR REQUESTS: IF NPR (1) THEN BBSY -- 0  
 4. \*\* TRIGGERING DELAY TO CLOCK MSYN CONDITIONAL UPON MACHINE STATE: -- (PROC RELEASE + SSYN) \* BBSY (1)  
 5. MSYN BEING SET TRIGGERS 15 USEC BUS TIMEOUT DELAY  
 6. INHIBIT SETTING MSYN IF BUS STOP ACTIVE: (RED ZONE OVFL + ODA ERR) TRAP  
 7. SCALE: 1 INCH ≈ 100 NSEC.

11-1677

KD11-A DATI(P) Bus Transaction Timing Diagram



- NOTES: 1. EFFECTS OF GATE DELAYS AND F.F. RESPONSE NOT SHOWN  
 2. ASSUMES KD11A HAS CONTROL OF THE UNIBUS [BBSY(1)]  
 3. \* CLOCK NPR REQUESTS. IF NPR (1) THEN BBSY = 0  
 4. \*\* TRIGGERING DELAY TO CLOCK MSYN CONDITIONAL UPON MACHINE STATE: - (PROC RELEASE + SSYN) + BBSY (1)  
 5. † MSYN BEING SET TRIGGERS 15 USEC BUS TIMEOUT DELAY  
 6. INHIBIT SETTING MSYN IF BUS STOP IS ACTIVE: [RED ZONE OVFL + ODA ERR] TRAP  
 7. SCALE: 1 INCH = 100 NSEC.

11-1678

KD11-A DATO(B) Bus Transaction Timing Diagram



11/40 - KD11A - 090

LEARNING ACTIVITY UNIT

PDP 11/40 LABORATORY PROJECT I

MAINTENANCE MODULE USAGE

1/21/74



RATIONALE

In the previous units you have analyzed the 11/40 Block Diagram and then the Flow Diagrams to understand how the processor accomplished the execution of machine instructions. With your general knowledge, the Block Diagram and the microprogram worksheets you were able to determine what exact steps the processor went through and what happens at each of these.

In this unit, then, you will verify that your previous determinations were in fact correct. You can do this by using the KM11A Maintenance Module. This maintenance aid allows you to execute one microinstruction at a time and then stop, one microinstruction and stop, etc.

The only way to determine if the sequence of microinstructions is correct and the data is correct is through proper use of the KM11A. To trouble shoot the PDP 11/40 effectively you must be able to correctly operate the Maintenance Module and analyze the information it provides for you.



PREREQUISITES

Successful completion to date of all previous units.

OBJECTIVES

Given one of the various console functions you must verify that the sequence taken and the data displayed is correct. After a console function do a typical machine instruction to verify its operation. To demonstrate successful completion of this unit you must correctly determine the sequence and the Data of a machine instruction given to you by the instructor, and record it on a microprogram worksheet.

PRETEST

To be filled in when a fully programmed environment is in effect.

LEARNING ACTIVITIES

I. LECTURE

Attend a short lecture where the instructor will define the indicators and switches of the KM11A as it applies to the PDP 11/40. Then a brief overview of its usage in the KD11A. All procedures concerning its use and purpose will be outlined. Questions will be answered and discussion permitted during this lecture.

II. PROJECTS/QUESTIONS

A. Before going to Lab

1. Review the procedure taken to check out equipment.
2. The instructor will make team and machine assignments as necessary.
3. Take the necessary material, to include:
  - (a) Picture of overlay.
  - (b) Processor Clock Block Diagram.
  - (c) Several Micorprogram Worksheets.
  - (d) Paper and pencil.
  - (e) Processor Engineering Drawings.
  - (f) KM11 Maintenance Module Set.

B. After arriving in Lab

1. Go to your assigned machine and make sure the power is off.
2. Assemble and install the KM11 in slot F1 of the KD11A backplane.

3. Turn the MclkEnab switch to the OFF position and put the HALT switch on the console to the Halt position and power up.
4. Once in the console mode dynamically Load Address to 500 and deposit a MOV %0, %1.
5. Now activate the Kml1 and single clock through an Examine of the instruction you just deposited.
6. When back in the Console Mode depress start and execute the previously deposited instruction to verify its operation.
7. Continue to practice using the Maintenance Module until such time that the instructor gives you your Post Test. Use your examples that you took through the Flows previously for additional practice.

C. Before leaving the Lab

1. Power down your systems and remove the Maintenance Module.
2. Replace panels, close doors, etc. to properly secure the system you were working with.
3. Straighten chairs, tables, etc. as necessary.

III. SELF EVALUATION

- A. Dynamically deposit the following information then single clock this instruction through. Check your results with those in the Appendix.

5000/BICB #377, -(3)

R3 = 3000  
2776/177777

Start at FET 05 and give final results.

#### IV. ADDITIONAL READING

Paragraph 7.3 and corresponding figures and tables on pages 7-4 - 7-8 in the KD11A Processor Maintenance Manual contain a detailed description of the K11A.

#### WHAT'S NEXT

After completing this unit, you have completed the second major portion of the 11/40. (First the Block Diagram, then the Flow Diagrams, and finally the Engineering Drawings which will be started next.) You are well on your way to obtaining the necessary knowledge and understanding to do an acceptable maintenance job with the PDP 11/40. In the next four (4) units you will identify and analyze the 11/40 Engineering Drawings so that you obtain a working knowledge of major circuits and the prints in general.

#### SUPPLEMENTARY INFORMATION

If you want more practice with the Maintenance Module, arrange some extra Lab time with your instructor.

APPENDIX

INDICATORS							SWITCHES	
PUPP 6	PUPP 3	PUPP 0	BUPP 6	BUPP 3	BUPP 0	C		
PUPP 7	PUPP 4	PUPP 1	BUPP 7	BUPP 4	BUPP 1	V		
PUPP 8	PUPP 5	PUPP 2	BUPP 8	BUPP 5	BUPP 2	Z		
		TRAP	SSYN	MSYN	T	N		

**NOTE:**

Maintenance module set plugs into slot F 1 of the KD11-A

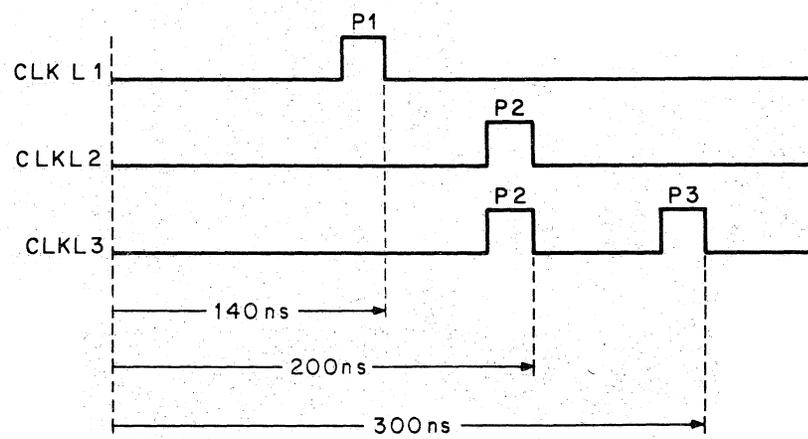
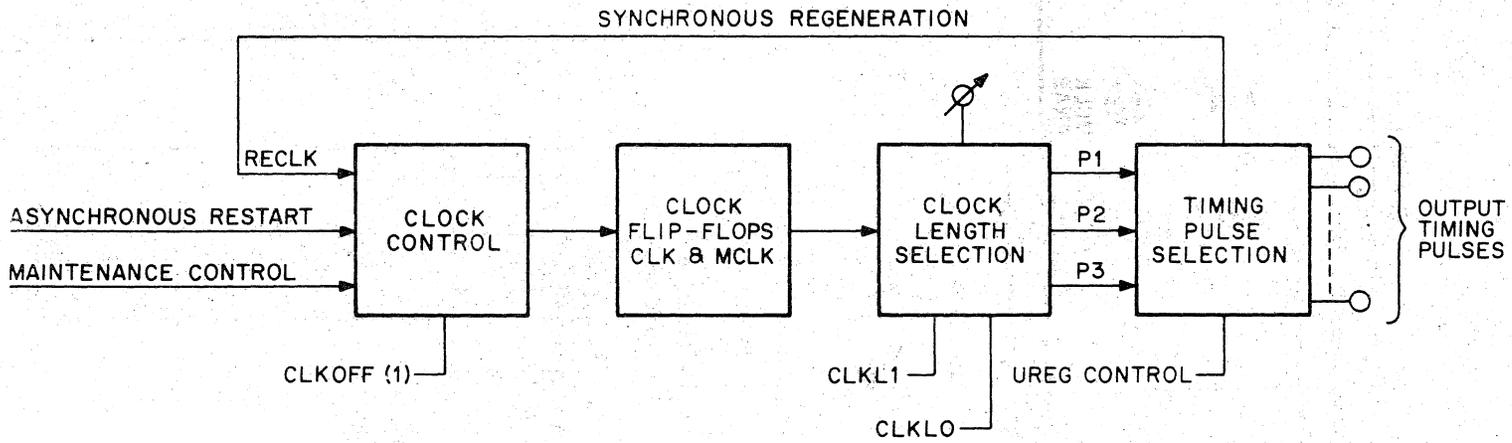
11-1662

PUPP = Address of the microinstruction stored in the Ureg.

BUPP = Address of the next microinstruction being read out.

1. MSTOP — Allows stopping the processor clock when BUPP = SR (8:0)
2. MCLK ENAB — disables processor clock and allows single clocking the microprogram with MCLK
3. MCLK — single clock switch

**KD11-A Maintenance Module Overlay**



KD11-A Processor Clock, Block Diagram

11-1661

APPENDIX

11/40 - KD11A - 090

-5a-





11/40 - KD11A - 100

LEARNING ACTIVITY UNIT

PDP 11/40 LOGIC DIAGRAMS I

FORMAT - SPECIAL CIRCUITS - DATA PATHS

1/21/74



RATIONALE

In the Learning Activity Units up to this point, you have learned the operation of the 11/40 utilizing the Block Diagram and the Flow Diagrams. Along with your general knowledge and these two things it became possible to determine how the 11/40 executed the 11 instruction set.

Now that you know how the CPU does its' thing, the last several units will deal with the Logic Diagrams (Engineering Drawings). You will be given an overview of the prints and shown the major circuits and key signals. An attempt will be made to provide you with the 'ability to gate chase' should it become necessary in the future. Identification, location, description and understanding will be our main goals during the next five units.

If at some time you find it necessary to trouble shoot the Processor down to chip level, these units will prove invaluable to you. A basic understanding of the prints will reinforce those concepts and ideas that you may have had difficulty with to date in the course.



PREREQUISITES

Satisfactory completion to date of all previous units.

OBJECTIVES

Given ten signals and/or circuits concerning the Data Paths, you must locate them in the Print set and state the coordinate and page number. To demonstrate successful completion of this unit you must correctly locate and state eight (8) of these signals/circuits. A time limit of 20 minutes will be allowed.

PRETEST

To be filled in when fully programmed environment is in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend a forty-five (45) minute lecture where the instructor will introduce the Engineering Drawings. He will discuss the format of each print, identify the conventions and present a brief overview of the Data Paths, module M7231.

Some special circuit chips will be analyzed and pointed out to you in the logic diagrams. Make notes on your prints and jot down any questions you may have and ask them after the lecture.

## II. PROJECTS/QUESTIONS

With your Block Diagram as an aid:

- A. Analyze the D mux on the K1-2 print and identify the four possible inputs. List the binary configuration of the SDM bits and the corresponding inputs below and on your block diagram.

SDM1,0	INPUT TO D MUX FROM
1.	
2.	
3.	
4.	

- B. Determine the output of BMUX BIT0 on the K1-2 print when the SBML BITS are equal to two (10)
- C. At coordinate B8 on Print K1-4 there are some 74H04 inverters at component location E29. What is the DEC part number?

- D. Locate the ALU (07:04), determine the inputs to A and B and the origin of the control bits to the ALU.
  
- E. The BA MUX is contained on \_\_\_\_\_ separate integrated circuit chips and is found at coordinate \_\_\_\_\_ on the K1 Print sets.
  
- F. The D register is a \_\_\_\_\_ type chip and gets its input from the \_\_\_\_\_.
  
- G. On K1-6, B1 you find FO4P2 state all you know about this.
  
- H. The D(C) Flip Flop is on K1-5 at what time is it clocked?
  
- I. The BA register is located on K1-6. Analyze the AND Gate at D6 (E5) and describe its purpose.
  
- J. On the K1-7 Print determine the first decoded address to give you the signal BOVFL stop H.

- K. The GPRs on K1-8 are type 3101A chips. What does it take to:
1. Address a GPR
  2. Read a GPR
  3. Write a GPR
- L. On the K1-9 Print a comparison of the SR and BUPP takes place. What two (2) outputs does this circuit provide?

Check your answers to the preceding questions in the Appendix. Get help from your instructor if you do not understand any of the above or if you want a more detailed explanation of any of the circuits.

### III. ADDITIONAL READING

Chapter 5 in the KD11A Processor Maintenance Manual contains the Logic Diagram Description. Pages 5-1, 5-2, 5-3 are concerned with the general format and pages 5-3 through 5-19 with a more detailed description of the M7231 Data Paths Module. All of this reading will be of a benefit to you now and in the future.

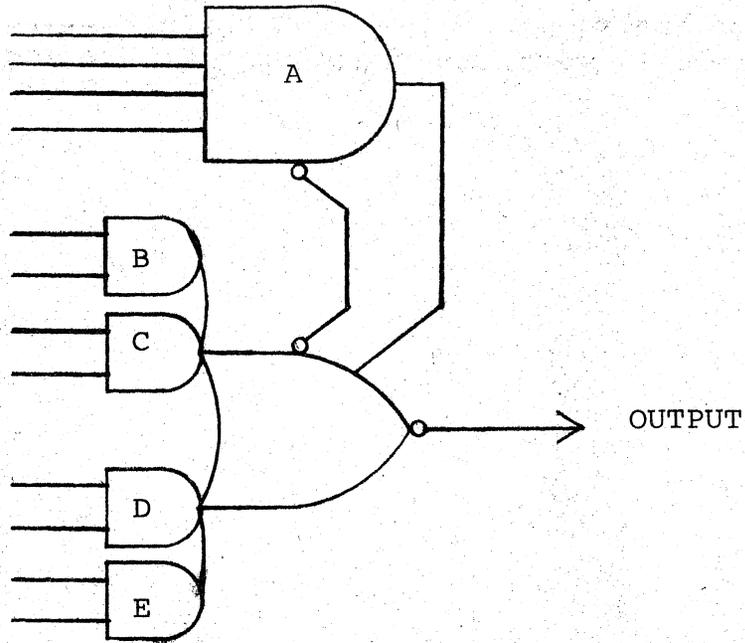
#### WHAT'S NEXT

After completing this unit you will continue with your analysis of the logic diagrams and go on to the rest of the Engineering Drawings. This group of LAU's consists of four (4) units.

SUPPLEMENTARY INFORMATION

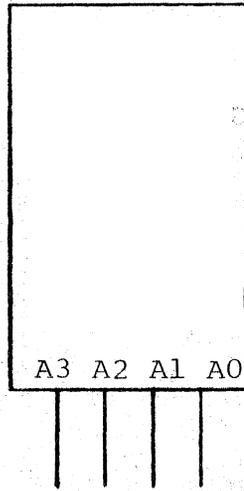
Consult your instructor if you want more practice or additional projects with this first unit of the logic diagrams.

APPENDIX

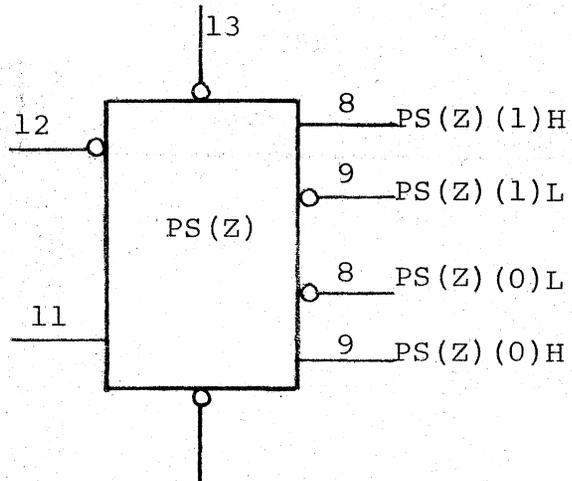
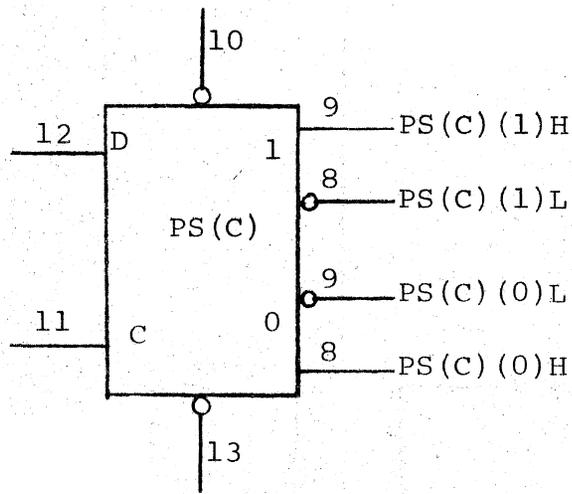


EXPANDED ADD/NOR. ALL "AND GATES" MAKE OUTPUT REACT IN SAME MANNER.

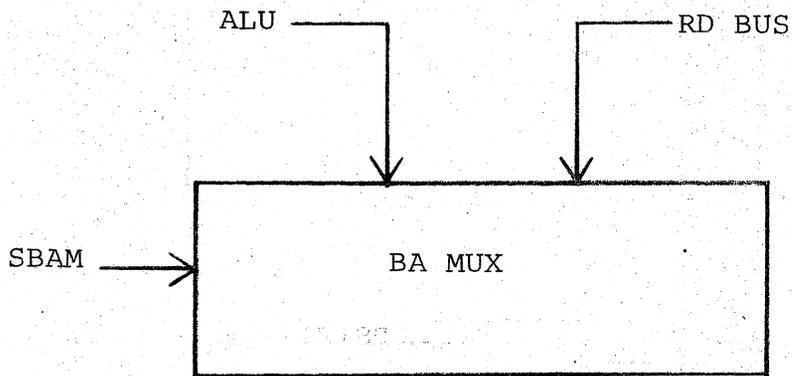
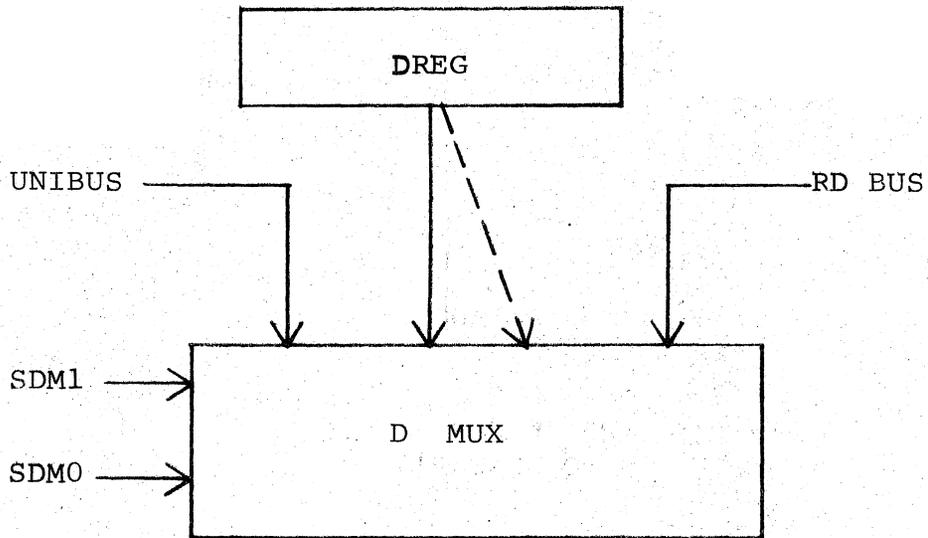
APPENDIX



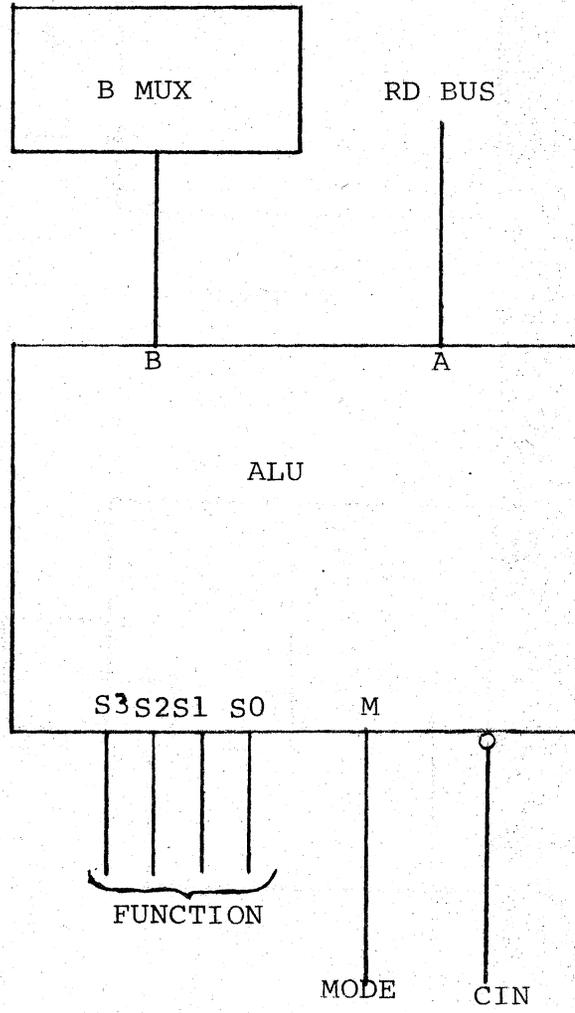
ALL LOWS TO ADRS R17



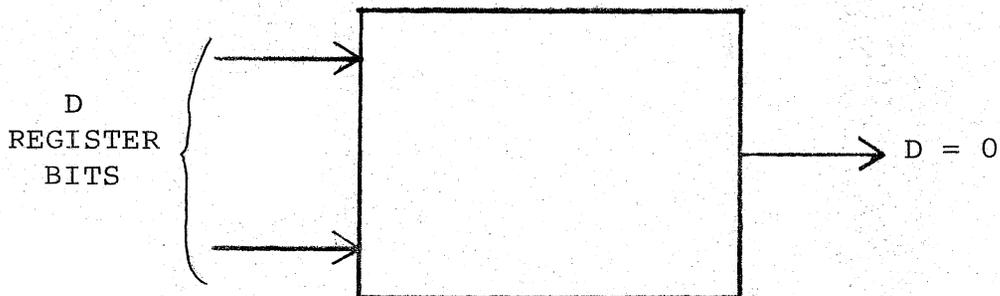
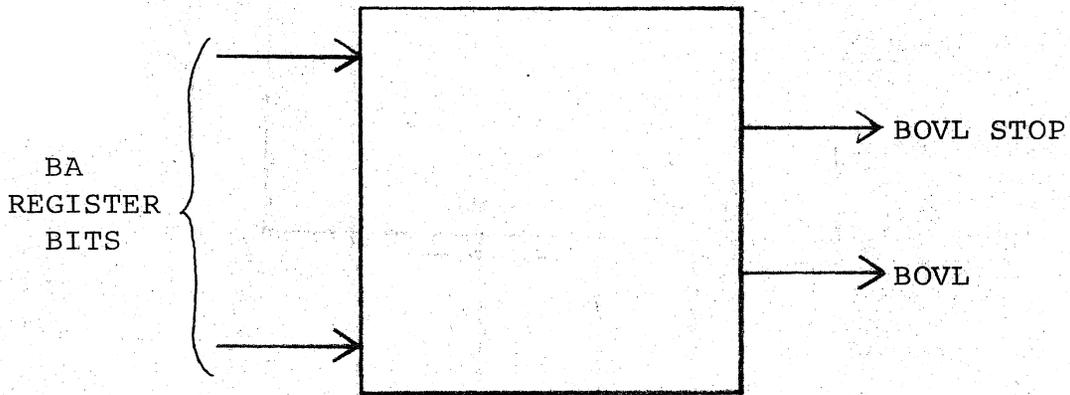
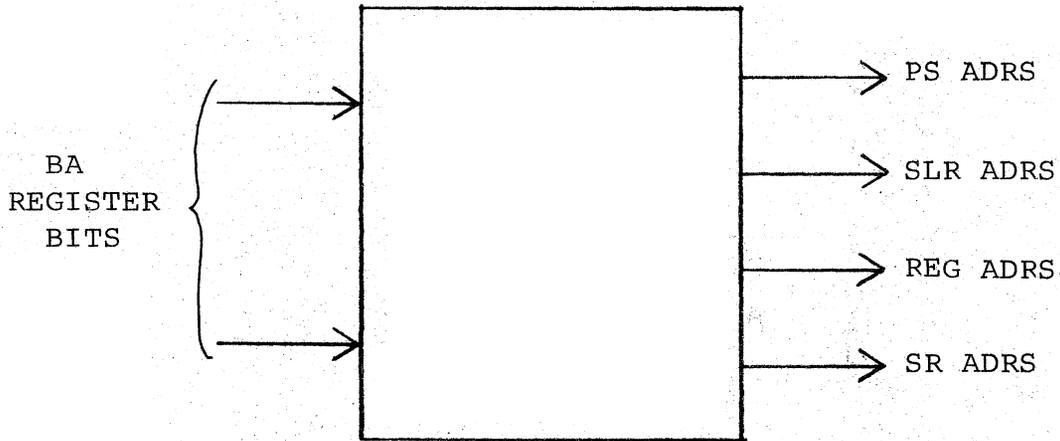
APPENDIX



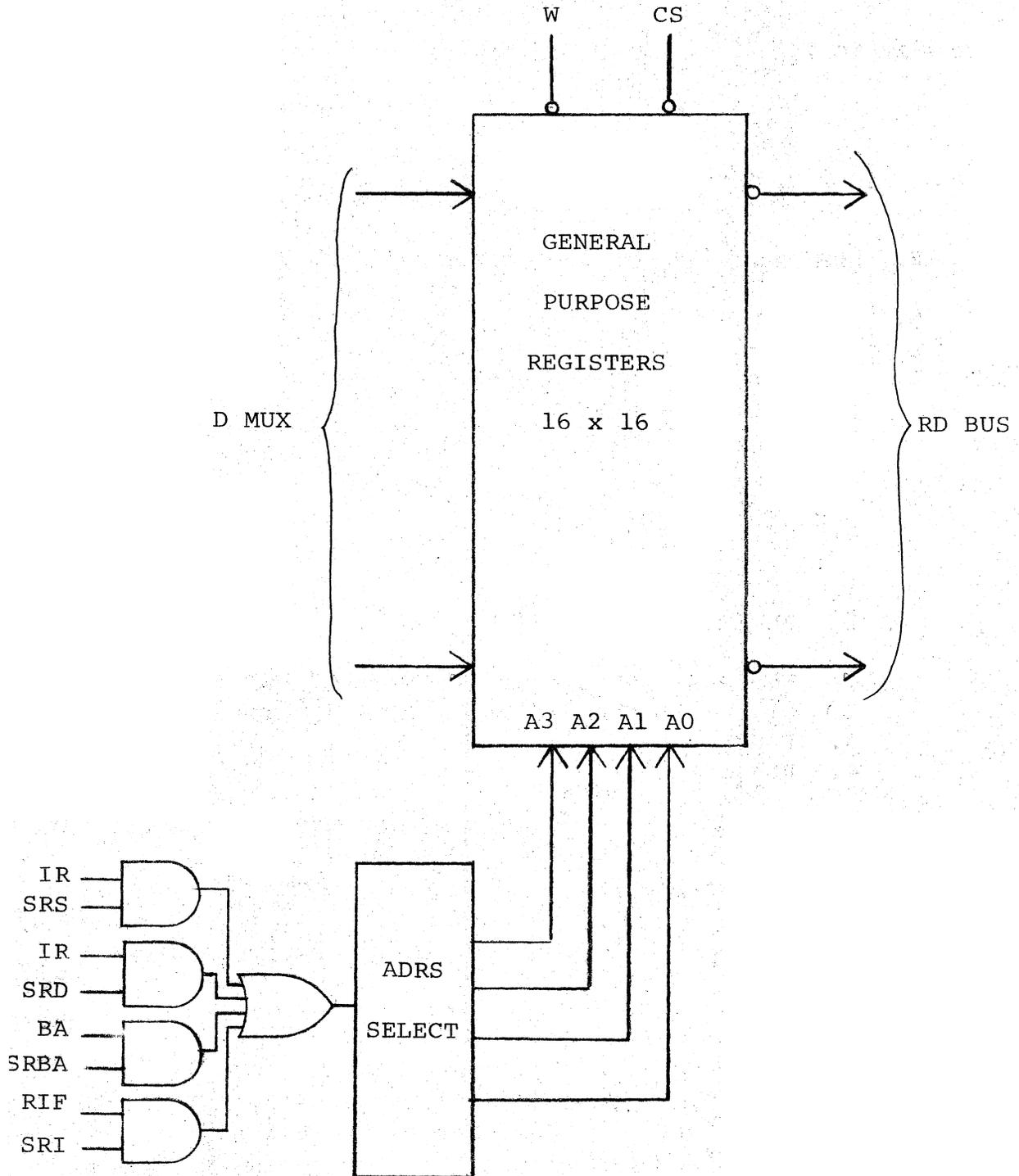
APPENDIX



APPENDIX



APPENDIX



APPENDIX

## ANSWERS TO PROJECTS AND QUESTIONS

- II. A. 1. 00, RD BUS  
 2. 01, UNIBUS DATA  
 3. 10, D REGISTER  
 4. 11, D REGISTER RIGHT SHIFTED
- B. B08 (swapping or duplicating bytes)
- C. 1909931 (K1-1)
- D. 1. A input = RD BUS  
 2. B input = B MUX  
 3. Control bits from K3-8
- E. 1. four (4)  
 2. B2
- F. 1. 74174  
 2. ALU
- G. 1. Level F (bottom) of a HEX height module  
 2. Slot 4 of the 9 slot KD11A Backplane  
 3. Pin P  
 4. Row 2
- H. When the D register is clocked (CLKD & P2 pulse)
- I. It is this And Gate that monitors BITS 13, 14, 15 of the BA register so that Bit 16, 17 become active low on the address lines whenever the attempted address is  $\geq 160000$ .
- J. 336
- K. 1. One of the four (4) select signals, SRI, SRBA, SRS, SRD, and their associated bits.  
 2. Same as above  
 3. Same as above plus a write signal.
- L. 1. UPP Match  
 2. P Match

11/40 KD11A 110

LEARNING ACTIVITY UNIT  
LOGIC DIAGRAMS II  
UWORD AND IR DECODE

1/21/74



RATIONALE

In the previous unit, you were introduced to the format of the logic diagrams and you took a relatively close look at the Data Paths module (M7231). You were able to identify and locate major circuits and key signals. You then applied that knowledge to your previous knowledge of the Block Diagram so that you could make a meaningful correlation between the two.

Perhaps this unit contains the two most important modules in the PDP11/40, UWord (M7232) and IR Decode (M7233). The ROM of course, being the heart of the Processor and the BUT providing the streamlining of the Microprogram. Our major goal in this unit will also be to identify and locate major circuits and key signals so that we recognize them in the future if the need arises. You will be able to discuss and illustrate the major areas of these two modules when you have completed this unit.

The knowledge and skills that you acquire in this unit will enable you to effectively identify problems within these modules. In a subsequent unit, you will be required to troubleshoot some live problems concerning these two modules and successful completion of this unit should enable you to properly isolate them.



PREREQUISITES

Satisfactory completion to date of all previous units.

OBJECTIVES

Given five multiple choice questions regarding circuit operations/failure symptoms, you must choose the most correct answer from those distractors given. To demonstrate successful completion, you must correctly answer four of the five questions. You will be allowed to use all available references but a time limit of 20 minutes will be in effect.

PRETEST

To be filled in when fully programmed environment is in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend a short lecture where the instructor will give a brief overview of the UWord M7232 and IR Decode (M7233) Logic Diagrams. Pay close attention and take any notes you feel are pertinent. Jot down your questions to ask after the lecture. Correlate your prints to the Block Diagram whenever possible to provide for a better and more thorough understanding.

## II. PROJECTS/QUESTIONS

## A. K2-2 Print

1. Analyze the UPP Flip Flops down the center of the print. Note how their state can be affected. Let us look at UPP3.
  - a. Direct set or clear by JAMUPP Logic
  - b. Clocked at (CLK(UPP \* PUPP))
  - c. Data Input qualified
    - (1) Output of ROM E15
    - (2) BUS U03 - from expansion ROM
    - (3) BUBC - Basic Micro branch Control
2. Locate the PUPP register and identify what the inputs are and when the clocking takes place

## B. K2-3 Print

1. Locate the address input to the ROM chip at the left side of the print & define this. (the UPP register output supplies the address to the ROM, verify on your Block Diagram)
2. Completely describe the clocking and data qualifications for the UPP8 Flip Flop. Compare your results with those given in the Appendix.
3. Note that this print completes the UPP and PUPP registers and that position of the ROM that contains the UPF.

C. K2-4 through K2-8 Prints

1. These diagrams contain the remainder of the UWord. Analyze these and identify the ROM chips and the corresponding register Flip Flops.
2. On the K2-8, note the three (3) Berg connectors. These connectors carry the expansion ROM bits if the EIS option is installed. They are left blank on machines without the KELL-E

D. The ROM (Micro instruction) listings are the final two sheets before the IR Decode, it gives the contents of each location in the KD11-A's ROM. Review your knowledge of these fields and format with the simplified description and diagram in the appendix.

E. K3-2 Prints

Analyze each of the six (6) multiplexer chips found here and determine the following:

1. Which BUT's activate each chip
2. Which Bits can each control
3. Where do the outputs from these multiplexers go
4. If a particular butted condition is true - what level leaves this page

When you have made your determinations verify your findings with those in the appendix

F. K3-3 Print IR & Decode

1. Analyze the four (4) IR register chips found on this print and note the inputs, outputs and clocking. Correlate this print with your Block Diagram.
2. Inspect the IR decoders to insure your comprehension. These should not be new to you. Check the truth table on this print and the appendix to verify your thoughts. Note the type outputs you get from these decoders.

G. K3-4 Print IRD & Overlap

1. More IR decoding found at the top of this print. Note what it takes to enable decoder E34.
2. Overlap combinational decoder. Decodes those various conditions under which we start a Fetch in micro instruction FET04. Cross reference the Boolean Expression of the overlap conditions found in note 3, Flow Diagram 1, to this logic. Verify as many conditions as you feel necessary.

H. K3-5 Print to modify Base Address 100 during a BUT 37. In other words, the logic that facilitates getting off Flow Diagram 1.

1. Let us analyze how the WAIT instruction would condition the appropriate bits such that Base Address 100 when doing a BUT 37 would become modified to a 114. BITS 2 & 3 are the ones we will concern ourselves with. Start at coordinate C5 with the translation WAIT 6.
  - a. Immediately fires two gates; E70 pin 8 and E69 pin 8
  - b. From E70 pin 8 through E29 pin 10 through E35 pin 03 to become BUBC3 H.
  - c. From E69 pin 8 through E54 pins 1 & 2 to become BUBC2 H
  - d. These two signals feed to sheet K3-2 E98 and E90. When a BUT 37 is active these two multiplexer chips invert and send a low to K2-2, E5 pin 5 and E7 pin 11 such that the UPP Flops can and will become set.
2. Now you do the same as above for the Halt instruction. When you are finished, check your results with the sketch in the appendix

I. K3-6 Print

More IR decoding just note general layout and the culmination of the decoding down the right hand side of print

J. K3-7 Print

Decoding for most of the other BUTS. For example, byte instructions, odd byte, SUB etc. There is an important piece of logic across the top of the page the Service Logic. When you arrive at the end of any machine instruction, as you recall, you BUT Service and if Service is high then you go to the Service Micro Flow. Investigate those conditions that will result in Service and list them below including a brief explanation Check your list against the appendix when you have completed

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.

K. K3-8 Print

ALU control. Locate and identify the following circuits.

1. The CIN multiplexer
2. The Cout multiplexer control
3. The ALU control multiplexer

L. K3-9 Print

Data conditioning inputs for the two processor status flip flops V and C

1. Analyze E26 at coordinate B2 (lower) note that during rotates and shifts to the right this gate enables PS(c) to accept D00.
2. Identify which gate is used when doing a SET condition code instruction.

III. ADDITIONAL READING

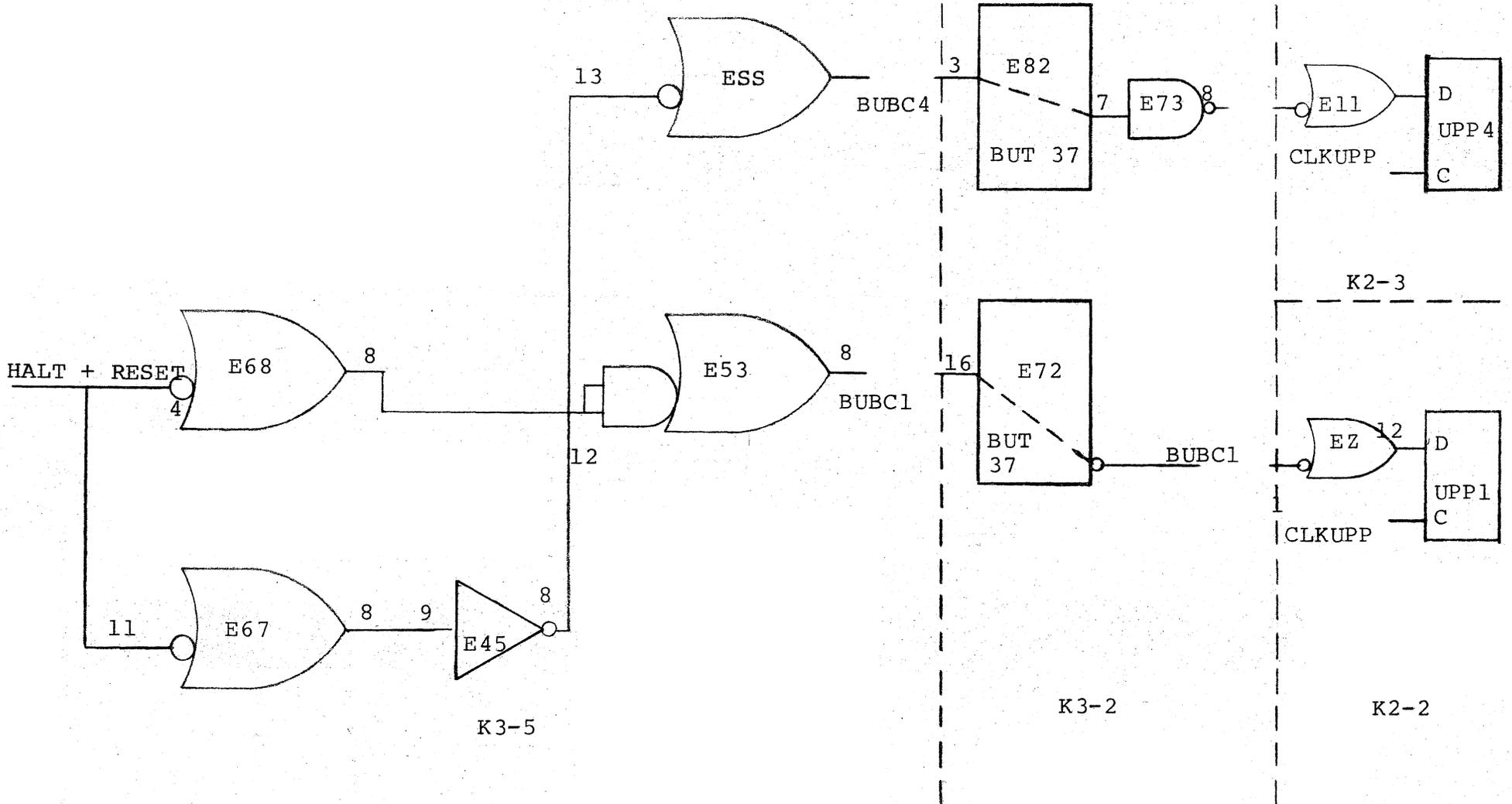
To broaden and reinforce your understanding of the logic just covered, it is recommended that you read and study the detailed logic descriptions in the KD11A Processor Maintenance Manual, pages 5-21 through 5-57

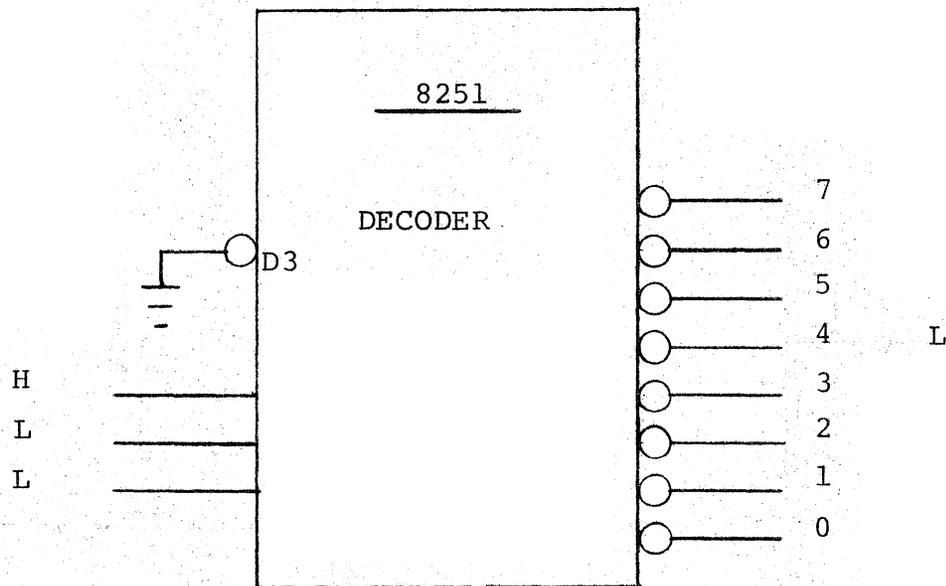
WHAT'S NEXT

After completing this unit, you should have developed the logic/technique necessary to master these prints. You will get more practice with the remaining prints of the processor.

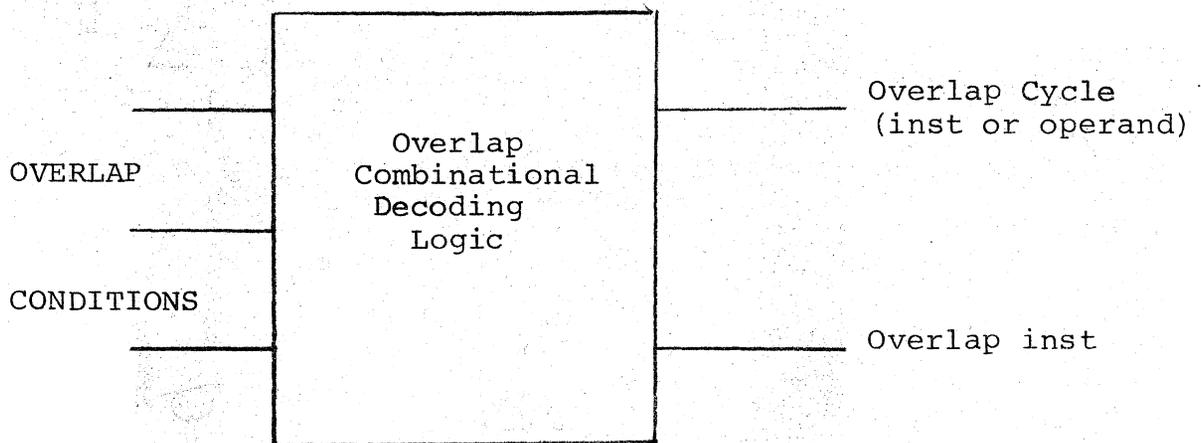
SUPPLEMENTARY INFORMATION

If you want more practice, consult your instructor for additional assignments.

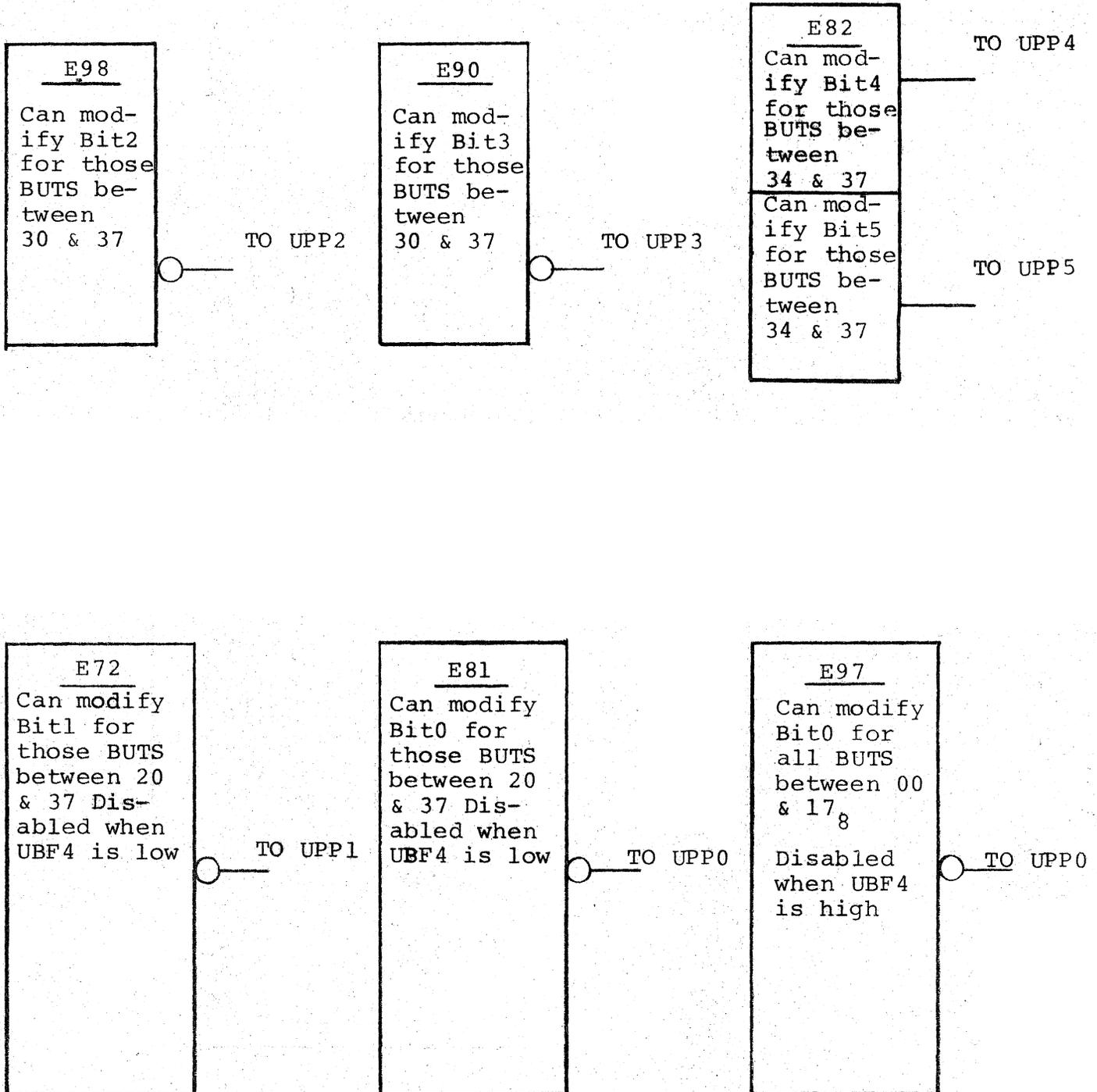




The Binary value of D2, D1, D0 activates one of eight possible outputs. D3 must be low to enable chip.  
 Note example.



APPENDIX



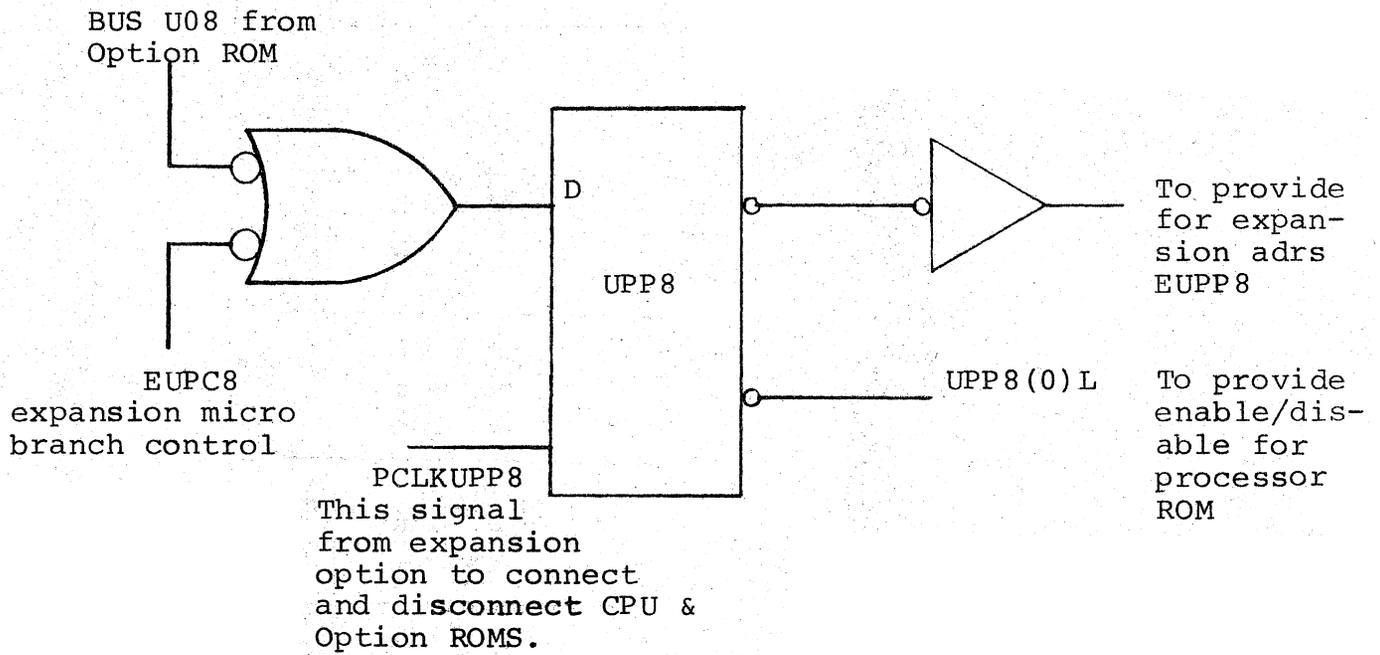
## UWORD LISITNG FORMAT

ALU	SBC	SBM	SDM	SBA	UBF	SRX	RIF	UPF
5 Bit selects mode & con- trols ALU func- tion	4 Bit se- lects a bin- ary con- stant	4 Bit set up the BMUX	2 Bit select 1 of 4 inputs thru the D MUX	1 Bit set up the BA MUX	5 Bit micro- branch control field	4 Bit what enable to pro- vide GPR Adrs	4 Bit in <sup>M</sup> instr to adrs a gpr	8 Bit the next adrs field

## UWORD LISTING FORMAT

FLWS	STATE	ARD	CLK	CIR	WR	CB	CD	CBA	BUS	DAD	SPS
Flow Page	Mnemonic name of micro inst	8 Bit Octal Add- ress	3 Bit Clock Code	1 Bit the IR Regis- ter	2 Bit Write the GPR's	1 Bit Clock the B Regis- ter	1 Bit Clock the D Regis- ter	1 Bit Clock the BA Regis- ter	3 Bit Bus Trans- action code	4 Bit Dis- crete alter- ation of data code	3 Bit select Pro- cessor status code

APPENDIX



## APPENDIX

SERVICE CONDITIONS	DESCRIPTION
1. DM = 0 * PS ADRS	You have just done something with the PS, you may have changed priority go back through Service just in case.
2. CBR (1) L	Console Bus Request. Halt switch is depressed.
3. PS (T) * -RTT	Go through Service on way to trap flows.
4. BERR	Go through Service on way to trap flows.
5. BOUFLW	Go through Service on way to trap flows.
6. PWRDN	Go through Service on way to trap flows.
7. BRP	Go through Service on way to trap flows.



KD11A 120 000

LEARNING ACTIVITY UNIT

KD11A LOGIC DIAGRAMS

CLOCK, JAM CLK, BUS DELAYS, MISC.

1/23/74



RATIONALE

By this time, you should be pretty good at finding your way around the prints. You have studied the major portions of the 11/40 which are represented on the Block Diagram. You have also seen how the prints are laid out and the types of circuits and nomenclature used.

Now you will study the logic that is used to control the KD11A. You will see the timing and clock pulse generation, the JAM CLK function and the power fail control. Then you will pick up some of the miscellaneous but essential circuitry, such as: the status bits, BUT Decoder, flags and the B Constant generator.

This unit is important in that this circuitry is usually the place where you have got to start when the machine is completely dead. Often the trouble is the JAM CLK logic not operating properly or the clock not generating the right pulses, etc.



PREREQUISITES

Successful completion of all units to date.

OBJECTIVES

Given some failure symptoms and a list of components, to demonstrate successful completion of this unit you will be required to pick the component which, if faulty, would most likely generate each of the symptoms.

LEARNING ACTIVITIES

I. LECTURE

Attend the lecture in which the instructor will give an overview of the following prints:

K4-2, K4-3, K5-8, K5-2, K5-3, K5-4, K5-5

The LAU will be your study guide for the first three of these. The lecture will cover all you need to know about the other four. So, pay close attention and take careful notes of all key signals and components.

II. PROJECTS/PROBLEMS

A. K4-2 CLOCK

1. Analyze how and when P1, P2 and P3 pulses are generated. Fill in the table below indicating all the possible combinations of clock length bits for each pulse.

<u>P1</u>	<u>P2</u>	<u>P3</u>
-----------	-----------	-----------

CLKL0

CLKL1

2. Analyze gate E62 output pin 06 at C-3. When is a RECLK generated? Which timing pulse does not generate a RECLK?

---

Observe how RECLK clocks the CLK flop at C-5, 6

3. Gate E88 output pin 08 at C, D-6 turns the clock on (This is the input called ASYNCHRONOUS on the block diagram). Inputs pins 01, 02, 03, 04 are for restarting the clock after a normal bus cycle. What are the conditions which qualify this gate. \_\_\_\_\_
-

4. E87 output pin 08 and E 82 output pin 08 at D-6, 7 are for the other cases the clock is started asynchronously. Note on your prints the following functions:
  - a. E87 pin 09 - from EIS Option
  - b. E87 pin 10 - to restart clock at the end of a RESET instruction.
  - c. E82 pins 09, 10 - after a BR in which data is being transferred  
- not used by any DEC equipment.
  - d. E82 pins 02, 03 - after a BR in which a vector address has been transferrred  
- this is the normal BR use
  - e. E82 pins 01, 13 - for starting after all JAMs except the first JAM of a power up.
  - f. E82 pins 04, 05, 06 - for starting after an NPR cycle.

Make note of these on your prints. We will see them again and discuss them further as we study JAMUPP later in this unit

5. Find the IDLE flop at C-6. This is set to indicate that the clock is turned off. Analyze how the gate which enables the data input to set the IDLE flop also causes the CLK flop not to set. The conditions which set the IDLE flop are as follows:
  - a. E77 pins 09, 10 - for the beginning of a JAM sequence
  - b. E77 pins 02, 03 - for a normal bus cycle  
- ALLOW CLK L tells us we are doing the last DATI to get the destination operand for a BIT, CMP or TST instruction  
- why is that signal here?

- c. E77 pins 13, 01 - for a Match Stop  
- remember, this is when the contents of the UPP is the same as the contents of the switch register and the Match Stop switch is enabled on the maintenance board.
  - d. E77 pins 04, 06, 05 - for a FETCH OVLAP  
- note this will stop the clock if we are in an OVLAP situation and the clock length bits are 0, 0.
6. Observe how the MCLK flop can simulate the clock when doing maintenance functions.
- a. Does the maintenance board inhibit the operation of the IDLE flop?  

---
  - b. Does the maintenance board inhibit the operation of the CLK flop?  

---
7. Observe the enabling gates down the page underneath the number 2.
- a. Which four timing pulses generate CLK U and CLK (UPP \* PUPP)?  
    - (i) \_\_\_\_\_
    - (ii) \_\_\_\_\_
    - (iii) \_\_\_\_\_
    - (iv) \_\_\_\_\_
  - b. Notice the clocking signals for the various register. Analyze to find out what pulses clock each of the following:  
    - (i) B \_\_\_\_\_
    - (ii) GPR \_\_\_\_\_
    - (iii) IR \_\_\_\_\_
    - (iv) BA \_\_\_\_\_
    - (v) D \_\_\_\_\_

Do these match what you previously noted on your Block Diagram?

B. K4-3 CLK/JAM

1. The 2  $\mu$ sec JAMUPP one-shot is really the heart of this circuitry. It is fired by qualifying gate E80 output pin 06 at D-5. The conditions which will do this are:
  - a. E80 pin 01 - JBERR flop which is set on an odd address error or a red zone stack overflow
  - b. E80 pin 02 - JPUP flop which is set for START switch activation in the HALT mode or for Power Up Restart
  - c. E80 pin 05 - NODAT flag is for SSYN time out for non-existent bus address errors
  - d. E80 pin 04 - PWRUP INIT is for a special JAM used during the power up sequence
2. Study the following sequence using an Odd Address Error as an example.
  - a. ODA ERR will generate BUS STOP and then CLK MSYN will set the JBERR flop.
  - b. This fires off the JAMUPP one shot
  - c. JAMUPP sets the IDLE flop on K4-2 and stops the processors clock.
  - d. JAMUPP anded with ODA ERR L directs sets and clears the UPP flops to put 002 in the UPP on K2-2, 3
  - e. A 100 nsec delay is allowed for 002 to be addressed in the ROM and have its contents sitting at the input to the U REG and its UPF bits on the inputs to the UPP.

- f. After the delay, JAM CLK is true generating CLK U and CLK (UPP \* PUPP) on K4-2

What will now be contained in the following registers? Use your flows to help you

(i) PUPP \_\_\_\_\_

(ii) UPP \_\_\_\_\_

(iii) U REG \_\_\_\_\_

- g. JAM CLK also fires the 100 nsec one-shot called JAMSTART. This will restart the processors clock a K4-2 and the processor continues on its sequence in Service to service the error.

All the other conditions which cause JAMs work in the same way, each causing a unique address to be jammed into the UPP. The only one which is slightly different is a power up JAM, and we will study this next.

C. K5-8 BUS Delays

1. Study the following power up sequence

- a. Power comes on ie, +5v. is true
- b. BUS DC LO L goes high dropping the direct clear on the PWRUP INIT one shot at B-3 causing it to fire. This generates several INIT signals which clear various registers throughout the CP and also out on the Unibus
- c. PWRUP INIT also fires off the JAMUPP one shot on K4-3. This forces a 377 into the UPP which, if you look at the ROM bit map, you will see reads all zeroes out of the ROM and these are clocked into the U REG and UPP.

- d. The processor's clock is not started on K4-2 because PWRUPP INIT is still active (20 msec). Therefore, so far what has been accomplished is:

The CP has been initialized

The Unibus has been initialized

The UPP and UREG have been zeroed out

- e. Sometime during the 20 msec. BUS AC LO L went high. Therefore, when PWRUP INIT times out it will fire the 70 msec. POWER RESTART one shot at C, D-3.
- f. When this times out it will set the JPUP flop a K4-3 initiating another JAM sequence. This time it will jam either 337 or 030 depending upon the HALT switch and will start the clock (remember PWRUP INIT is gone) and the CP is off and running again.
- g. JPUP will fire off the DELAY POWER DOWN one shot on K5-8 which will prevent the power down sequence from occurring for 3 nsec. This allows the programmer time to set up everything properly before he has to power down again. Further information can be found in the JAMUPP timing sequence in Appendix A and in Fig. 3-8 on page 3-11 of the KD11A Maintenance Manual.

2. Read and study the power down sequence on the last paragraph of page 3-6 and figure 3-9 on page 3-11 of the KD11A Processor Maintenance Manual
  - a. Answer True or False. On the power down sequence, BUS AC LO sets the LOWAC flop, generating CLK PWR ON on a P1 or P3 which jams a 337 into the UPP. \_\_\_\_\_
  - b. How much time does the program have to save the volatile information and to prepare itself for the power going off?

- 15 msec
- 7 msec
- 22 msec
- 2 msec
- 9 msec

D.

1. Analyze the RESET and RESET RESTART one shot.
  - a. During which microinstruction do they both get fired? \_\_\_\_\_
2. Note during this microinstruction that we have a CLKOFF. What signal will restart the clock?
  - a. K5-8 P ENDRESET L
  - b. K5-8 RESET RESTART L
  - c. K5-8 INIT \* RESET H
3. How long will this be after the clock was stopped?

KD11A 120 000

- a. 70 msec
- b. 20 msec
- c. 90 msec

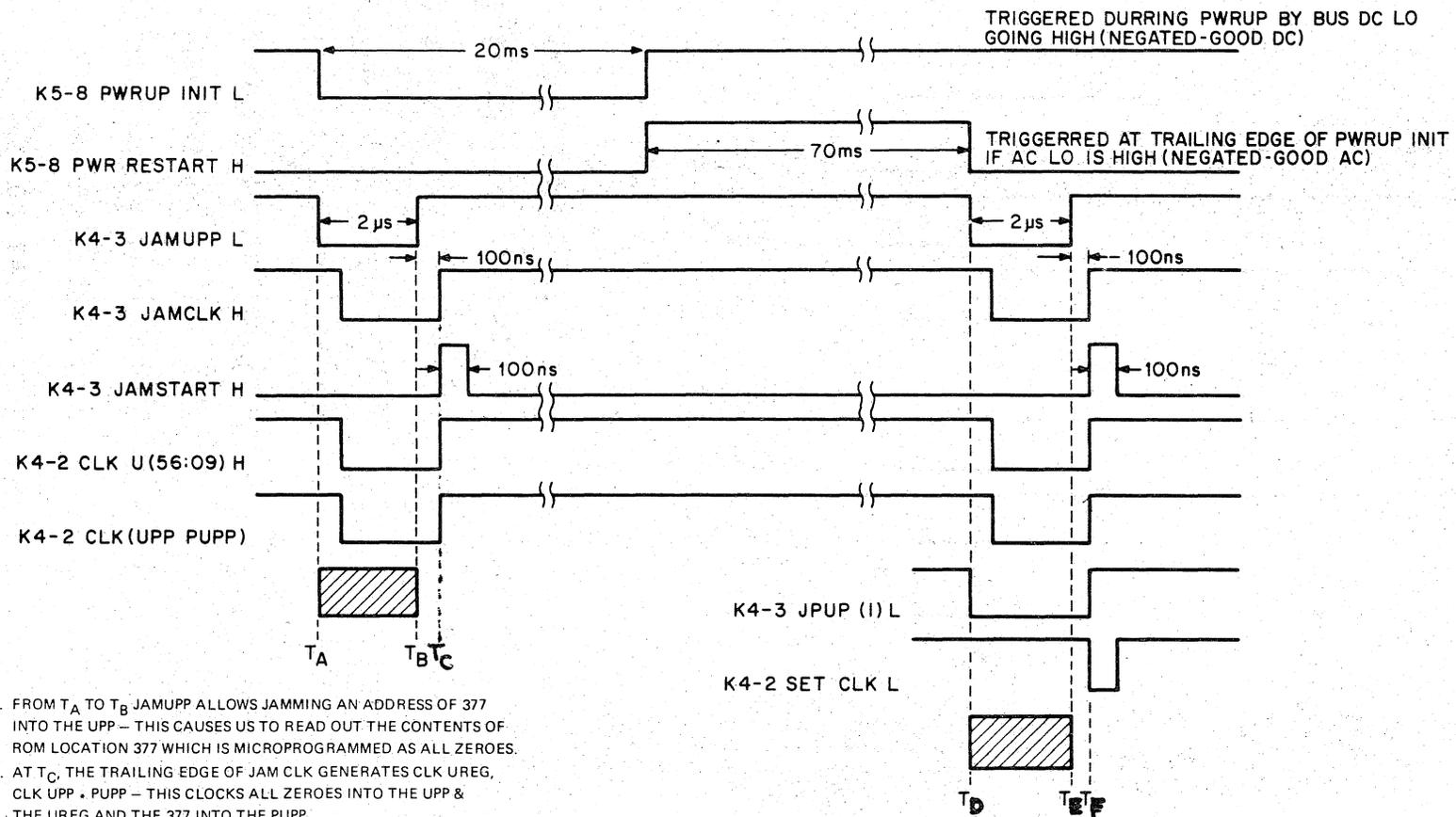
## SELF EVALUATION

Match the symptom listed below with the component that would most likely cause the symptom. The problem could be on any of the seven prints studied in this unit. Check you answers with the correct answers in Appendix B.

1. You deposit 177777 into location 777776, but when you examine the same location the data displayed is 000353. \_\_\_\_\_
2. You observe that every microinstruction which specifies a CL3 generates only a P3 and no P2. Microinstructions specifying CL2 operate properly. \_\_\_\_\_
3. Whenever a RESET instruction is executed, the clock stops and never gets restarted. The Unibus is not initialized. \_\_\_\_\_
4. When doing sequential examines or deposits, the contents of the BA never increments. Everything else works properly. (ie all Maindecs run)  
\_\_\_\_\_

LIST OF POSSIBLE FAULTY COMPONENTS

- a. E66 pin 02 failed high
- b. E26 pin 09 failed high
- c. E74 pin 11 failed high
- d. E56 pin 09 failed low
- e. E92 pin 08 failed low
- f. E30 pin 08 failed high
- g. E72 pin 11 failed high
- h. E66 pin 01 failed high
- i. E69 pin 06 failed high
- j. E60 pin 06 failed low



1. FROM  $T_A$  TO  $T_B$  JAMUPP ALLOWS JAMMING AN ADDRESS OF 377 INTO THE UPP - THIS CAUSES US TO READ OUT THE CONTENTS OF ROM LOCATION 377 WHICH IS MICROPROGRAMMED AS ALL ZEROES.
2. AT  $T_C$ , THE TRAILING EDGE OF JAM CLK GENERATES CLK UREG, CLK UPP + PUPP - THIS CLOCKS ALL ZEROES INTO THE UPP & THE UREG AND THE 377 INTO THE PUPP.
3. THE JAMSTART PULSE IS INHIBITED FROM STARTING THE CLOCK BY PWRUP INIT STILL BEING ACTIVE.
4. SINCE THE UREG CONTAINS ALL ZEROES WE HAVE FORCED THE MACHINE TO A KNOWN STATE - ALL CONTROL SIGNALS INACTIVE (LOCATION 000 IN THE ROM IS BEING READ OUT, BUT THE SUBSEQUENT JAM SEQUENCE WILL ALTER THE UPP PRIOR TO CLOCKING THE UREG AGAIN)
5. AT  $T_D$  THE TRAILING EDGE OF PWR RESTART SETS THE JPUP FLOP
6. JPUP TRIGGERS THE JAMUPP ONE-SHOT AND FROM  $T_D$  TO  $T_E$  WE JAM A NEW ADDRESS INTO THE UPP. [IF HALT; UPP = 30 (CON05)]  
[IF - HALT; UPP = 337 (TRP17)]

7. ASSUME THE HALT SWITCH WAS ON; AT  $T_F$  THE TRAILING EDGE OF JAM CLK GENERATES CLKU, CLK(UPP+PUPP) - THIS CLOCKS THE CONTENTS OF LOC 30 INTO THE UREG, 315 INTO THE UPP, AND 30 INTO THE PUPP - WE NOW HAVE THE MICROPROGRAM IN THE CONSOLE ROUTINES.
8. THIS TIME, THE JAMSTART PULSE GENERATES THE SET CLK SIGNAL TO TURN ON THE PROCESSOR CLOCK - THE MICROPROGRAM SHOULD NOW SEQUENCE TO CON06 AND START LOOPING, WAITING FOR A CONSOLE CONTROL SWITCH.

11-1664

KD11-A JAMUPP Timing Sequence

APPENDIX B

A.

7

b

- (i) B, P1 or P3
- (ii) GPR, P1 or P3
- (iii) IR, P1 or P3
- (iv) BA, P1 or P2
- (v) D, P2

B. 2

f

- (i) PUPP - 002
- (ii) UPP - 015
- (iii) U REG - SER01

C.

2

a

FALSE

b

(ii) 7 msec

D.

1

a On P3 of RST01 on Flow Diagram page 6

2

a K5-8 P ENDRESET L

3

a 70 msec

## ANSWERS TO PROBLEMS

A

1		P1	P2	P3
	CLKL0	0, 1	0, 1	1
	CLKL1	0	1	1

2 P2 when CLKL0(1) \* CLKL1(1)  
i.e. CL3

3 B SSYN H \* IDLE (1) H \* no clock pulse \*  
NO MCLK ENABLE H \* MSYN (1) H

5

b Inhibits stopping the clock since we do  
not want to do a bus cycle for these  
situations.

6

a No

b Yes

7 a

(i) P1  
(ii) P2 on a CL2  
(iii) P3  
(iv) JAM CLK

APPENDIX B

ANSWERS TO SELF EVALUATION

1. E56 pin 09 failed low (K5-2)
2. E66 pin 01 failed high (K4-2)
3. E30 pin 08 failed high (K5-8)
4. E74 pin 11 failed high (K5-5)



11/40 - KELLER - 160

LEARNING ACTIVITY UNIT

EXTENDED INSTRUCTION SET OPTION I

INTRO - INSTRUCTIONS - BLOCK DIAGRAM

1/21/74



RATIONALE

In the previous units you have accomplished the objectives of learning the operation and maintenance of the PDP 11/40. A thorough knowledge of the 11/40 Block Diagram, you will find, will be a big advantage to you as you progress through these units concerning the Extended Instruction Set Option.

During this and the subsequent units of this group you will undertake the task of learning the KELLE. The degree to which we will address ourselves will be such that when completed, we can effectively and efficiently maintain the EIS option. This first unit will deal with introducing you to the option, learning and using the four (4) new instructions, and an overview of the Block Diagram.

This unit will provide a knowledge foundation so that you can see the 'big picture' as you go through the remaining units. An understanding of how this option fits into the overall system is probably the most important aspect of this unit.



PREREQUISITES

Successful completion of all previous units to date.

OBJECTIVES

Given six (6) questions concerning the Extended Instructions and the Block Diagram, you must be able to choose the most correct answer from those given or determine if the statement is true or false. To demonstrate successful completion you must answer five of the six correctly. Open book of course, however, a twenty minute time limit must be observed!

PRETEST

To be filled in when a completely programmed environment is in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend a short lecture where the instructor will introduce the Extended Instruction Option. He will discuss the whys and hows. He will also format for you and briefly discuss the four (4) new instructions and give you an overview of the Block Diagram of the KELLE. Take notes and jot down questions you have during the lecture so that you can ask them when the lecture has been completed.

## II. PROJECTS/QUESTIONS

A. Answer the following questions with the aid of your KELLE Block Diagram.

1. The DR register is part of which section of the EIS option?
2. Two (2) control signals feed the DR register because of the type register it is. What type is it?
3. The data route from the BR register to the CPU is via the RD Mux and into the processor on the \_\_\_\_\_.
4. The Expansion BUT MUX is controlled by \_\_\_\_\_.
5. What is the technique utilized to make 44 bits become 56 bits so that the expansion ROM can be used to control the processors operation?
6. What key signal controls which ROM is being used when this option is installed?
7. The counter can count up or down - True or False.

Refer to the Appendix to verify your answers to these questions concerning the EIS Block Diagram.



### III. ADDITIONAL READING

The Introduction, General Description, and Programming in the KELLE Instruction Set Options Manual should prove very useful. There are examples of the instructions and a good general description of the option. Be careful to only concern yourself with the EIS explanation at this time.

#### WHAT'S NEXT

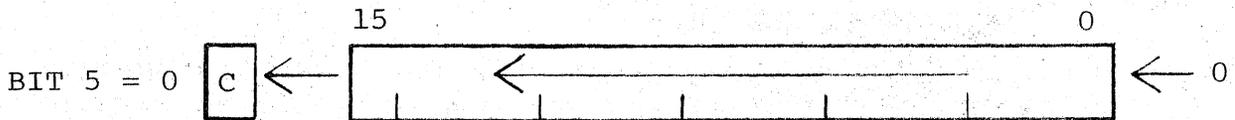
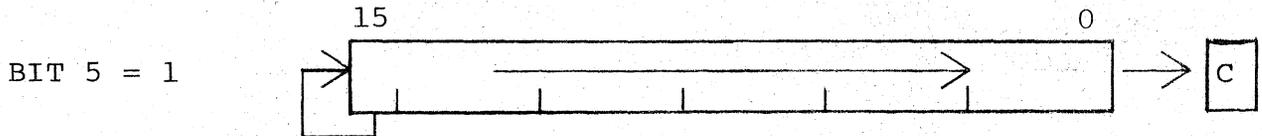
After completing this unit, you should understand how the option ties into the KD11A and the basic job it has to perform. You will need this knowledge as you progress to the subsequent units to discover how the KELLE does its thing.

#### SUPPLEMENTARY INFORMATION

Your instructor will assign additional practice problems if you feel the need for more exercise and examples.

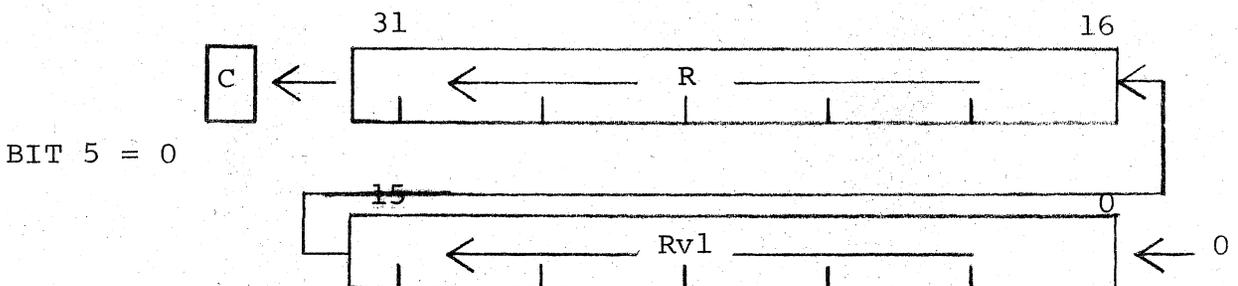
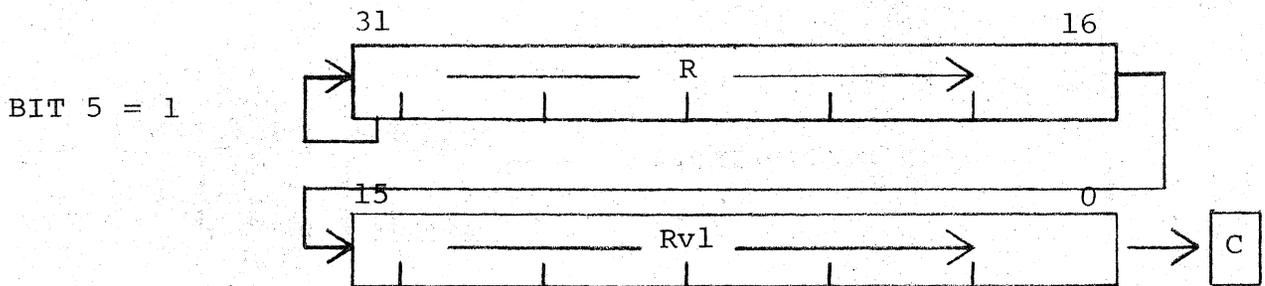
ASH

ASSEMBLER SYNTAX ASH S, R WHERE R IS THE REGISTER TO BE SHIFTED  
 MACHINE OR OCTAL 072RSS NOTE REVERSAL  
 THE SHIFT COUNT IS TAKEN AS LOW ORDER 6 BITS OF SOURCE OPERAND  
 WHERE BIT 5 DETERMINES DIRECTION AND 4-0 AMOUNT. TOTAL +31, -32.



ASHC

ASSEMBLER SYNTAX ASHC S, R WHERE R & Rv1 ARE THE REGS TO BE SHIFTED  
 MACHINE OR OCTAL 073RSS NOTE REVERSAL  
 THE SHIFT COUNT IS SAME AS ABOVE, HOWEVER, TWO REGISTERS ARE AFFECTED



MUL

ASSEMBLER SYNTAX MUL S, R WHERE R IS MULTIPLIED BY THE SOURCE MACHINE OR OCTAL 070RSS NOTE REVERSAL THE CONTENTS OF R AND THE SOURCE ARE MULTIPLIED TOGETHER AND STORE IN R AND Rv1.

<u>MUL</u>		
MULPLICAND	- R	16 BITS
x MULTIPLIER	- SS	16 BITS
PRODUCT	- R AND Rv1	32 BITS

EX. 1

```
MOV #400, R1
MUL #10, R1
```

↙ MULTIPLICAND (R)

↘ MULTIPLIER (SS)

RESULT R1 = 4000

EX. 2

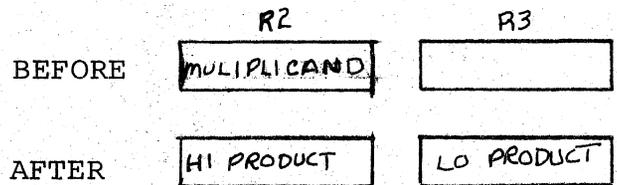
R1 = 1000

```
MUL (1), %2
```

↙ MULTIPLICAND

↘ R1(1000)

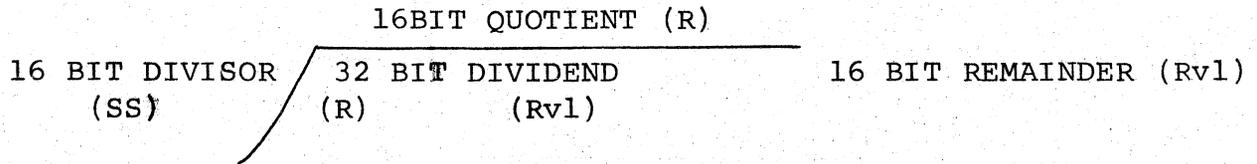
1000/MULTIPLIER



APPENDIX

DIV

ASSEMBLER SYNTAX DIV S,R WHERE R AND Rv1 GET DIVIDED BY SOME SOURCE  
 MACHINE OR OCTAL 071RSS NOTE REVERSAL  
 THE DOUBLE REGISTER R AND Rv1 GETS DIVIDED BY SOURCE ANSWER TO  
 R AND REMAINDER TO Rv1.



EX. 1

```
CLR R0
MOV #20001, R1
DIV #2, R0
```

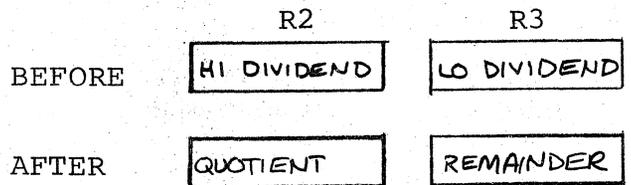
→ WITH R1, DIVIDEND  
 → DIVISOR

RESULT - QUOTIENT (R0) = 010000  
 REMAINDER (R1) = 000001

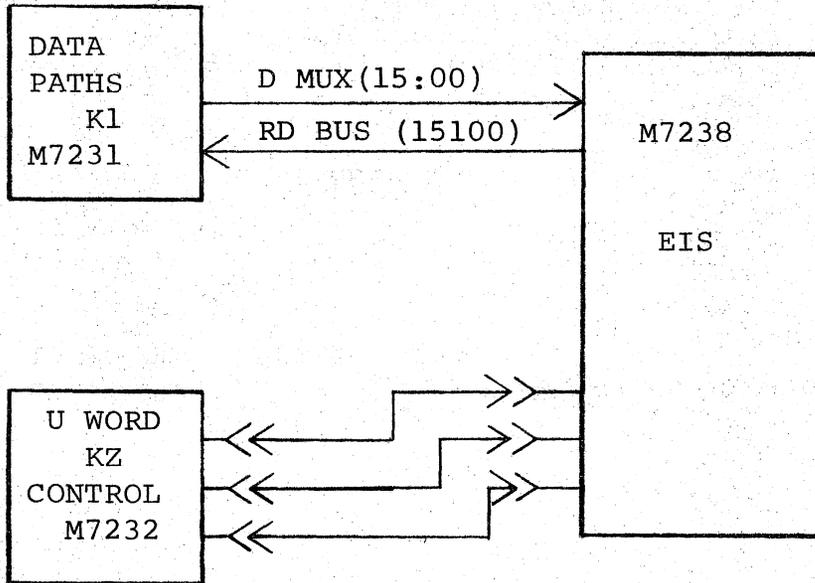
EX. 2

```
R1 = 1000
DIV (1), %2
```

→ DIVIDEND  
 → R1 (1000)  
 1000/DIVISOR



KE 11-E EIS



BASIC IDEA - INSERTS NEW ROM.

APPENDIX

ANSWERS TO PROJECTS/QUESTIONS

A. Relative to the Block Diagram

1. Data Paths
2. Shift Register
3. RD Bus
4. EUBF (3:0)
5. Encoding
6. EUPP 8 (1)
7. True

B. Relative to the Execution of the EIS Instructions

1. 0, 0  $15_8$  Shifts Right of R3
2. 0, 5336, 0  $16_8$  Shifts Right of R0, R1
3. R2 = 0, R3 = 430
4. R0 = 5 (Quotient), R1 = 11 (Remainder)



11/40 - KELLE - 180

LEARNING ACTIVITY UNIT

EXTENDED INSTRUCTION SET OPTION III

TO AND FROM AND SHIFT FLOW SUMMARY

1/21/74



RATIONALE

The first two units of this section were concerned with getting to know the EIS. You discussed the instructions and the Block Diagram and then analyzed to some extent how the option could execute the instructions assigned.

This unit will provide you with the information and the capability of getting from the Processor Flows to the EIS Flows and then back again. In other words, the routes taken when cutting in and cutting out the option. This unit will also serve as an introduction to the EIS Flows and a summary of the Shift Flows will be made. A brief analysis of those "extra" control bits will also be made.

To effectively trouble shoot this option, there are some techniques that can be employed by merely knowing how to activate and deactivate this option; you will discuss these during this unit. The analysis of the Shift Flows will give you the proper foundation to effectively relate the multiply and divide to those Algorithms discussed.



PREREQUISITES

Successful completion to date of all previous units

OBJECTIVES

At the end of this unit you will be asked to answer five (5) questions. To demonstrate successful completion of this unit, you will be required to correctly answer four of the five. You will be allowed to use your reference material, however, a 20 minute time limit will be in effect.

PRETEST

To be filled in when a fully programmed environment is in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend a 30 minute lecture where the instructor will very briefly overview four aspects of the EIS pertinent to this unit. First, how to get to EIS; secondly, how to get from EIS back to CPU; thirdly, the expansion ROM Format (80:57), and finally, a summary of the Shift Micro Flows. He will attempt to proceed as rapidly as possible, so jot down any questions you may have and ask them at the conclusion of the lecture.

## II. PROJECTS/QUESTIONS

A. Analyze more thoroughly those steps taken to activate the EIS option, include:

1. Translating a ASH with and without option
2. The Clocking and Data qualifications for UPP8
3. The BUT EIS INSTR.

Summarizing: the activation of the EIS option is caused by Fetching and Translating an EIS instruction thereby clocking and setting UPP8.

B. Analyze more thoroughly those steps taken to deactivate the EIS option; include:

1. Translating a Floating Instruction with only EIS installed
2. The clocking and clearing of UPP8
3. The exit to Service

Summarizing: the deactivation of EIS is caused by clocking and clearing the UPP8 Flip Flop which a direct function of the EIS ROM.

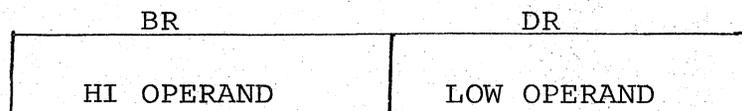


Summary for ASH:

1. ASH 0  
Puts the shift count of 3 into the counter and checks to see if it is a left or right shift.
2. ASH 1  
Puts number to be shifted (75) into the BR register and in our case decrements the counter by 1.
3. ASH 5  
Checks to see if the count is zero. If the shift count had been one, it would be zero now and only one pass thru ASH 6 would be made. When count = zero, it is inhibited from being decremented any more, otherwise, the count is decremented once more in this micro instruction.
4. ASH 6  
The shifting takes place here. Note where it is accomplished and that it complies with your general knowledge of the ASH operation.
5. ASH 15 & 21  
Takes care of status and routes you to SERVICE.

Summary for ASHC:

1. ASH 7  
Puts the shift count in the counter, moves the low operand (R3) to BR and checks if right or left shift.
2. ASH 8  
Moves the low operand to DR and puts the hi into BR and increments the counter. (Shift is right). Operand position shown here and in the Flows.



3. ASH 10  
Serves the same function as ASH 5 in the previous example. Relate if required.
4. ASH 11  
Shifting taking place. Note where the operands are shifted. The low operand gets shifted \_\_\_\_\_, and the hi operand gets shifted \_\_\_\_\_.
5. ASH 12  
Moves the low answer from DR to the D reg in preparation for storing.
6. ASH 13  
Stores the low answer in Rv1.
7. ASH 19  
Or's the high and low answer together on the RD Bus to test for zeroness in preparation for setting the condition codes.
8. ASH 15 & 21  
Moves Expansion Processor Status Bits to Processor Status and Exit back to the CPU.

### III. ADDITIONAL READING

Paragraph 4.7.3, p. 4-26 - 4-30 of the EIS Options Manual provide an extremely fine explanation and description of the work we covered so far in the EIS Flows; it would be to your advantage to read this.

### WHAT'S NEXT

After completing this unit you have the necessary basis to go on to the Multiply and Divide Flow which will be handled in a similar manner; summarizing the Flows. As you will see, the time spent on flow diagrams to date can and will be put to good use in the remaining EIS Flows.

SUPPLEMENTARY INFORMATION

If you feel you need more practice with the topics covered in this unit, arrange with your instructor for additional assignments or projects.

APPENDIX

ANSWERS TO PROJECTS/QUESTIONS

II. C.

1. Routes DR 15 to carry in of ALU
2. Divide Quit (Divide Fault)
3. Enables to counter to count
4. Control the RD MUX
5. Is this an EIS instruction

D. 2.

- a. ASH - shift count
- b. ASHC - shift count
- c. MUL - multiplier
- d. DIV - divisor

E. 4.

- a. in DR (EIS Option)
- b. in Processor (D Mux facility)



11/40 - KELLE - 190

LEARNING ACTIVITY UNIT

EXTENDED INSTRUCTION SET OPTION IV

MULTIPLY FLOW SUMMARY

1/21/74



RATIONALE

In the previous units, you have determined what is EIS, why EIS, and how it fits into the system. You have examined the Expansion Control Bits and analyzed the basic data manipulation capabilities of this option.

This unit will use those fundamental things we have observed to date to perform the Multiply Instruction. Let us keep in mind, as we proceed with this unit, that the important thing to learn will be what steps EIS takes to accomplish a successful multiply, not the proof of; if we take these steps, will the answer be correct. The difference, of course, is between learning the Algorithm (a recursive computational procedure) and proving that the Algorithm is in fact correct. Take my word for it, the Algorithm works. This unit, therefore, will deal with a Summary of the Multiply Flow Diagram.

The knowledge and ideas you obtain in this unit will enable you to effectively troubleshoot the Multiply Instruction, if necessary, and provide you a firm foundation from which you can analyze the next unit (the Divide Instruction).



PREREQUISITES

Successful completion to date of all previous units.

OBJECTIVES

There are five (5) simple questions relating to the operation of the Multiply Instruction at the conclusion of this unit; to demonstrate successful completion, you must correctly answer four of the five. Your notes, flows, and any other material may be used, however, a twenty (20) minute time limit will be in effect.

PRETEST

To be filled in when a fully programmed environment is in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend a short lecture where the instructor will review the ideas and concepts of how multiplication in the 11/40 is accomplished. He will then give you a brief overview of the Multiply Flow Diagrams in the EIS Engineering Drawings. He will discuss some troubleshooting techniques similar to those concerning the ASH and ASHC. Pay close attention and question the instructor if you are having difficulty understanding the basic ideas.

## II. PROJECTS/QUESTIONS

Let us go through and describe and summarize each microinstruction or group of microinstructions as they would relate to any multiply. We will not work with any particular instruction, but rather with the operands.

	Multiplicand	-	R(SF)
x	<u>Multiplier</u>	-	BR, B R(DEST)
	Product		R and Rv1
	Hi		Low

This is the condition as we enter Flow Diagram 3.

A. A summary of the Multiply microinstructions follows:

1. MUL 0  
MUL 1 - Set up count, test multiplier for negative number, 2's complement multiplier in case it is the most negative number. 100000g.
2. MUL 7  
MUL 8 - If multiplier is positive, move multiplicand to DR and zero to BR.
3. MUL 2  
MUL 3 - If multiplier is negative, move multiplicand to DR, test multiplier for most negative number.

4. MUL 19 - If multiplier is  $\neq 100000_8$ , then the operands are in position. Decrement count before going into loop (because PUPP = UPP = UPF) and go to loop.
5. MUL 4  
MUL 5 - If multiplier =  $100000_8$ , then make it the multiplicand by putting it in the DR and zero the BR. Test the original multiplicand for a negative number.
6. MUL 6 - If the original multiplicand is positive, put it in the B reg (it is now the multiplier) decrement the count and go to the Loop.
7. MUL 21  
MUL 22  
MUL 23 - If the original multiplicand is negative, test it for the most negative number. Prepare to generate the answer "special case."
8. MUL 24  
MUL 25  
MUL 26 - If the original multiplicand is also  $100000_8$ , generate the answer, set CC's and store.
9. MUL 27 - If the original multiplicand  $\neq$  the most negative number, clock it into the B reg; it is now the multiplier. The original multiplier, now the multiplicand, is in DR with 0's in BR (MUL 5). Decrement count and go to the Loop.

Now, before analyzing the Multiply Loop, let us see if I can summarize the operation so that there is no doubt as to what has happened to this point.

There are six (6) possible combinations of the multiplier - multiplicand as I see it. Let us list them in a table to simplify things.

MULTIPLIER	MULTIPLICAND	ACTION TAKEN
1. Positive	Do not care	Position operands, to loop
2. Negative	Do not care	If not most neg., to loop
3. Most negative	Do not care	Swap operands; multipli- cand now becomes multiplier
4. Positive		Position operands, to loop
5. Negative		If not most neg., to loop
6. Most negative		Generate answer

10. MUL 9 - The operands are shown on the Flow and in the Appendix. The SHIFT will be RIGHT and the TECHNIQUE is the STRING OF ONES.

The Loop: As you ENTER a string of ones, subtract the least significant power of two of the string from the partial product. As you LEAVE a string of ones, ADD the most significant power-plus-one to the partial product. WITHIN a string of ones or zeroes, just shift the partial product right without adding or subtracting the multiplier. When count = 0, pass through the Loop once more, then out.

11. MUL 10  
MUL 20  
MUL 11 - Store the hi product and test the low product for sign. Also test the hi product for future setting of the EPS (C) bit.
12. MUL 16  
MUL 17  
MUL 18 - Store the low product. If the low product is positive and the hi product = 0's, clear EPS (C) indicating single word precision. If low product is positive and the hi product  $\neq$  0's, set EPS (C) indicating double word precision.

13. MUL 12

MUL 13

MUL 14

MUL 15 - Store the low product. If the low product is negative and the hi product = 1's, clear EPS (C) indicating single word precision. If the low product is negative and the hi product  $\neq$  1's, set EPS (C) indicating double word precision.

Once the Multiply loop is left, these remaining micro instructions store the product. In addition, so that the programmer can meaningfully use the condition codes, the microprogram goes through some manipulations to determine the sign of the answer.

Now you move those condition codes to the Processor's Status Register and go to service.

B. Now that you know what it takes to accomplish the Execution of a Multiply Instruction, list some troubleshooting techniques that you would use having been given the following symptom: The Multiply doesn't work.

- 1.
- 2.
- 3.
- 4.

Check the Appendix to see how your ideas compare with those given.

### III. ADDITIONAL READING

To further enhance your understanding of the Multiply operation, it is suggested that you read the Multiply write-up, pages 4-30 - 4-33 in the EIS Options Manual.

#### WHAT'S NEXT

After completing this unit, you should have very little trouble with the next one. Basically, only the names will be changed as we tackle the Divide. All of the ideas,

concepts, and techniques learned to date will be able to be put to use in the next unit.

SUPPLEMENTARY INFORMATION

If you want more practice or you want to take a bit by bit example through to prove the Algorithm is correct, check with the instructor - there may be some simplified example already worked out.

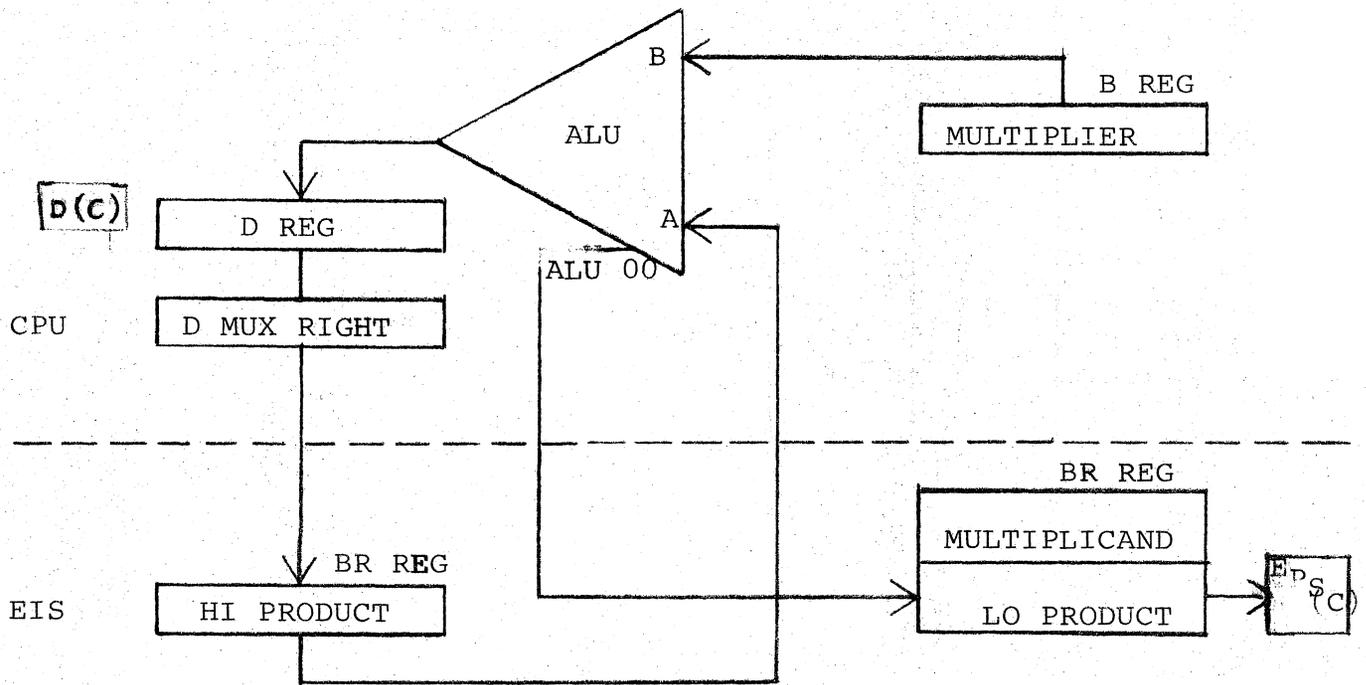
APPENDIX

ANSWERS TO PROJECTS/QUESTIONS

- B. Troubleshooting tips
  - 1. Insure EIS activation
  - 2. Operands = 100000
  - 3. Multiplier small positive  
Multiplicand = 0
  - 4. Two small positive operands

MULTIPLY LOOP KELL-E

ALU CONTROL f(DR00 & EPS(C))		
DR00	EPS(C)	ALUF
0	0	BR
0	1	BR PLUS B
1	0	BR MINUS B
1	1	BR



11/40 KELLE 200

LEARNING ACTIVITY UNIT

EXTENDED INSTRUCTION SET OPTION V

DIVIDE FLOW SUMMARY

1/21/74



RATIONALE

The previous units of this EIS group dealt with the introduction to EIS and the Flow Diagrams of the ASH, ASHC and MUL. The reasons for EIS and the implementation of the option was discussed, also.

In this unit, you will want to gather all of your previous knowledge and skills so that you can effectively use them while analyzing the Divide Flow Diagram. Again in this unit an attempt will be made to understand an Algorithm; not prove its correctness. This time, of course, it's the Divide. The main thrust of the unit will be directed toward a summary of the microprogram flows. Very similar to the way we tackled the multiply flow.

The information obtained during this and the other units to date will provide the necessary background to effectively analyze the Engineering Drawings which will be the next and the last unit in this section. You will realize that a thorough understanding of the flows is what is necessary to be effective in the maintenance of this option.



PREREQUISITES

Successful completion to date of all previous units.

OBJECTIVES

At the end of this unit, there will be five (5) simple questions relating to the operation of the Divide instruction; to demonstrate successful completion, you must correctly answer four of the five. As is the standard case, all notes, flows, reference manuals, etc. can be used, However, you will be limited to 20 minutes for completion.

PRETEST

To be filled in when a fully programmed environment is in progress.

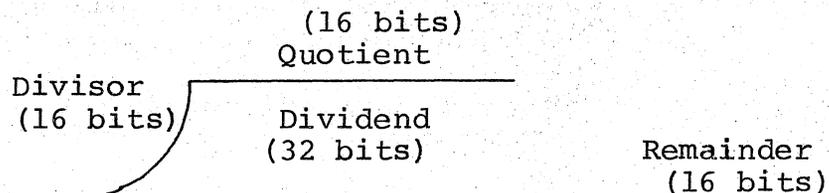
LEARNING ACTIVITIES

## I. LECTURE

Attend a short lecture where the instructor will review the method the 11/40 uses to accomplish the Divide instruction. He will again identify the Divide operands and the limitations of the operation (Divide Quit Conditions). A brief overview of the Divide Flow Diagrams will be given and typical troubleshooting techniques discussed. Pay close attention during the lecture and question the instructor if you are having difficulty with the basic ideas.

## II. PROJECTS/QUESTIONS

Let us go through the Divide Flow Diagrams summarizing micro instructions or groups of micro instructions. We will deal with the Divide in general terms and with the operands instead of some specific values or members. Keep in mind the technique here is left shift and subtract, subtract, subtract until you over subtract, then add the divisor back in. The first subtract must fail and the sign of the remainder follows the sign of the dividend.



- A. Our entry is to DIV 0 top left of Flow Diagram 4. The operands are located:

```

      Divisor -BR, B R(DEJT)
HI  Dividend - R
LO  Dividend - RV 1
  
```

Now, let us summarize the flow

1. DIV 0  
 DIV 1  
 DIV 2  
 DIV 3 - Load count, get high order dividend  
 test for the divisor = 0.
2. DIV 4 - If the divisor = 0, let local CC's  
 go to MOVE EPS.
3. DIV 5  
 DIV 6 - If divisor  $\neq$  0, load low dividend,  
 test sign of high dividend, complement  
 low dividend in case dividend is neg-  
 ative. The Algorithm requires a  
 positive dividend.
4. DIV 7  
 DIV 8  
 DIV 9  
 DIV 10 - If the dividend is negative  
 complement it and test for most  
 negative # (100000). Set local  
 CC's just in case.
5. DIV 11 - If dividend = most negative #,  
 quit.
6. DIV 13  
 DIV 14  
 DIV 15 - If the dividend is positive, set  
 up high and low dividends and  
 divisor for 1st (trial) division  
 step
7. DIV 16  
 DIV 17  
 DIV 18 - 1st division step is a trial subtract  
 (it must fail); if the divisor  
 goes into the high order dividend  
 its overflow, therefore
8. DIV 12 - Set local CC's, go to move EPS
9. DIV 19  
 DIV 20 - If the trial subtract resulted in  
 a carry out (successful), shift  
 the dividend left and subtract the  
 divisor again. Continue in this  
 manner until either count = 0 or  
 the sign of the dividend changes  
 (no carry out)

9. DIV 20 "Cont"  
Shift the dividend left and add the divisor back in. Continue in this manner until count = 0 or original sign of dividend is restored.
10. DIV 21  
DIV 22 -When count = 0, get ready to correct and store the remainder and test the last add/sub operation.
11. DIV 23  
DIV 24 -A positive divisor and last subtract operation resulted in a carry out (good); complement remainder in case original sign of dividend was negative. Test original sign of dividend.
12. DIV 25 -If original sign of the dividend was positive, every thing is OK as is. Move the quotient into the D Reg and BR. Get ready to set condition codes and leave.
13. DIV 27  
DIV 28 -A positive divisor and the last subtract operation did not result in a carry out (bad), add the divisor back in to compensate for having over subtracted. Then store this corrected remainder and go to DIV 23.
14. DIV 33  
DIV 34 -A negative divisor and the last addition operation resulted in a carry out (good); complement the remainder in case original sign of dividend was negative. Test original sign of dividend.
15. DIV 35  
DIV 36 -If the original sign of the dividend was negative, the quotient is OK but the remainder must be complemented. Go to DIV 25.

- 16 DIV 31  
DIV 32 -A negative divisor and the last addition operation did not result in a carry out (bad), subtract, the divisor from the remainder to compensate for having over added. Then store the corrected remainder and go to DIV 33.
17. DIV 37  
DIV 38  
DIV 39 -The original sign of the dividend was negative and now the quotient must be complemented to compensate. Test the quotient for most negative # (100000).
18. DIV 40  
DIV 41  
DIV 42 -Set local condition codes. If quotient = most negative #, quit, otherwise:
19. DIV 26 -Store the quotient, set codes and go to service.
- B. The Divide Quit expression describes the Fault monitoring with the Boolean equation. Make a meaningful explanation of each member here and check your analysis with that in the appendix to see if you concur.
1.  $D \neq 0 * DR00$
  2.  $B15 * EPS (N) * DR00$
  3.  $-B15 * -EPS (N) * DR00$
- C. If time permits at this point construct a simple divide instruction, and verify its operation and sequence through the flows.

### III. ADDITIONAL READING

A very good description of the Divide operation exists in the reference manual, EIS option manual- pages 4-33 thru 4-36. It certainly would be a benefit to you to read this especially if you desire more detailed information on the Divide. It is not very lengthy, and we recommend it.

### WHAT'S NEXT

After completing this unit, you have finished the EIS option with the exception of analyzing the Logic Diagrams. One more unit will be required to accomplish that and that is what follows. Your knowledge gained in the previous units will be extremely useful in the identification of the logic circuits, and you should be able to easily relate the circuits and key signals to the Block Diagram and the Flows.

### SUPPLEMENTARY INFORMATION

If you want more practice or possibly some bit by bit examples, check with your instructor. There may be some lengthy examples already worked out that you could analyze.

APPENDIX

DIVIDE CAPSULE SUMMARY

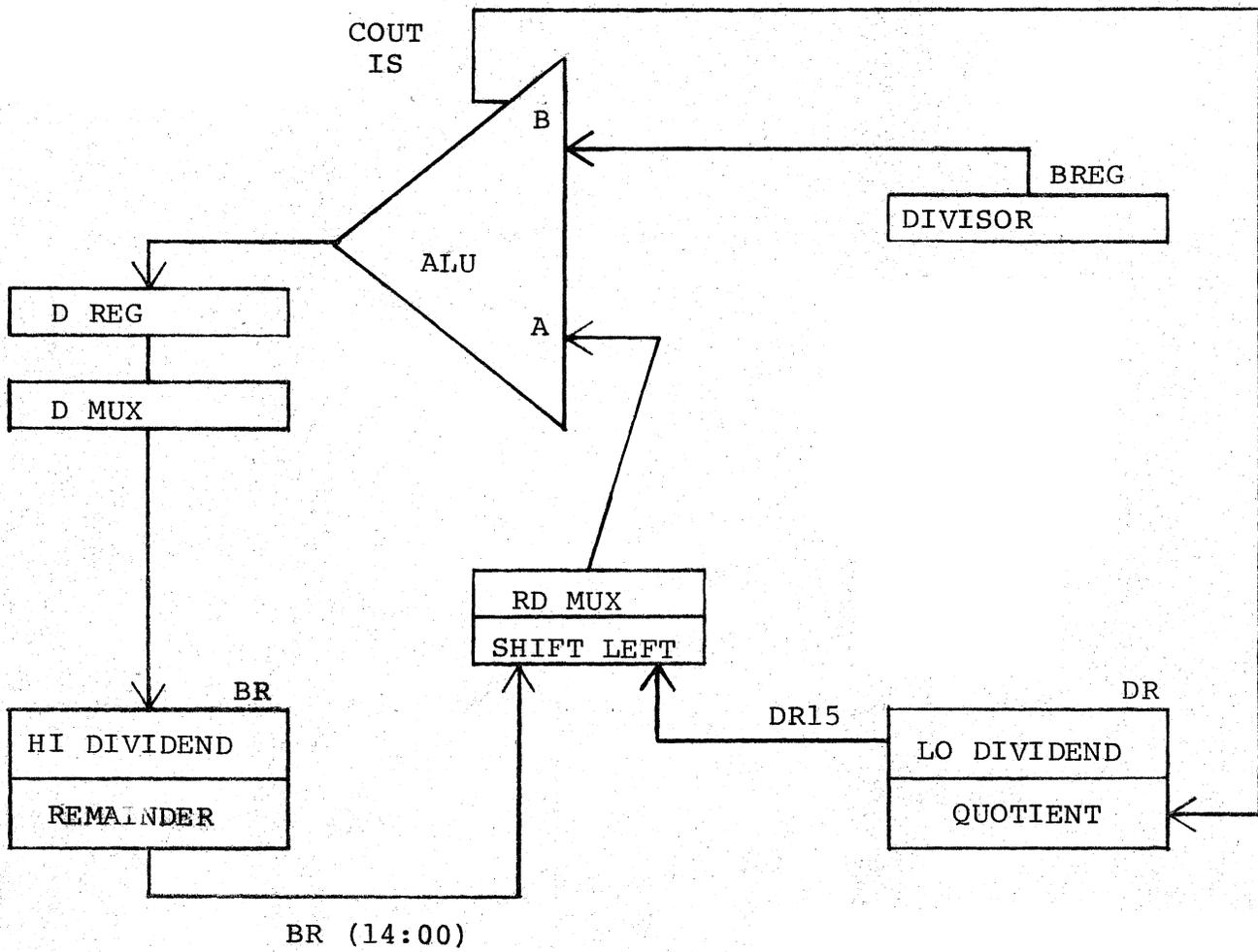
XXX/DIV (1), R(SF) ← must be even

↳ R1 = A A/Divisor



1. Get Divisor (load SC = 17<sub>8</sub>)
- ← 2. Test Divisor for all zeroes
3. Get Dividend, test for negative
- ← 4. If the Dividend negative, test for most negative
- ← 5. Is Dividend = to 0
6. Make Dividend positive prior to coop
- ← 7. Test if quotient will be > 16 bits
8. Divide Loop count = 0
9. Correct answers
10. Store Answers
- ← 11. Complement = 100000
12. Store Flags
- 13. Exit

DIVIDE LOOP (EIS)



## APPENDIX

## ANSWERS TO PROJECTS/QUESTIONS

B 1.  $D = 0 * DR00$

This takes care of a situation where you attempt a divide of a large positive number into a relatively small negative number. And a large negative number into a small positive number. It is conceivable that maintaining the proper sign could be impossible. Divide quit.

2.  $B15 * EPS (N) * DR00$

This makes sure that a positive divided by a positive yields a positive, if not, Divide Quit

3.  $-B15 * -EPS (N) * DR00$

This makes sure that a negative divided by a negative yields a positive, if not, Divide Quit.



11/40 - KELLE - 210

LEARNING ACTIVITY UNIT

EXTENDED INSTRUCTION SET OPTION VI

LOGIC OVERVIEW

1/21/74



RATIONALE

All of the previous units you learned in this set had to do mainly with the Block Diagram and the Flow Diagrams of EIS. Most of the understanding and knowledge was obtained with those facilities. As you may remember from the KD11-A, the logic diagrams just show how the Block and Flows are implemented.

In this unit, we will investigate the EIS Logic Diagrams. As you will very rapidly see, there is not much to them. Our analysis will focus upon the location and identification of the key signals and important circuits. Some of the Flow and Block Diagram techniques will now become more apparent to you when you analyze these diagrams.

This is the last unit in the PDP 11/40 section of this course. Knowing the 11/40, as you will find, will be a tremendous asset in your quest to master the 11/45, which is next. Some areas of the processor will require only the names to be changed, other areas will be changed however slightly. In all cases, you will be able to apply those ideas and concepts you have learned to date to good advantage as you progress through the course.

Good Luck.



PREREQUISITES

Successful completion to date of all previous units.

OBJECTIVES

Five (5) simple questions concerning the EIS Logic will be given at the end of this unit. To successfully complete this unit, you will be required to answer four of the five correctly. You can use any notes or reference material but must complete within a twenty (20) minute time limit.

PRETEST

To be filled in when a fully programmed environment in progress.

LEARNING ACTIVITIES

## I. LECTURE

Attend a short lecture where the instructor will give you a brief overview of the EIS Logic Diagrams. He will stress the major signals and key circuits. Pay close attention and jot down questions that you have so that you can ask them when the lecture has been completed.

## II. PROJECTS/QUESTIONS

A. KE-2 Print. This print has the BR and DR registers on it which are the major items in the Data Paths portion of the EIS, analyze, identify, or verify the following:

1. BR input
2. DR input
3. Data Shift input to DR00
4. Data Shift input to DR15
5. DR's truth table

Check your findings with those in the Appendix.

B. KE-3 Print. This Print contains the RD Mux. The path (DATA) between the EIS option and the processor's RD Bus. Analyze and verify the following:

1. Signal to allow RDMUX to RDBUS.
2. Information through the RDMUX on
  - (a) A input
  - (b) B input
  - (c) C input
  - (d) D input

Check your answers with those in the Appendix.

C. KE-4 Print. Expansion BUT MUX, EUBC4-1 are able to be activated under various conditions. Identify when multiplexer chips are active and verify the Divide Quit conditions.

1. E20
2. E14
3. E24
4. E37

Check your answers with those in the Appendix.

D. KE-5 Print. Control page implements EIS activation/deactivation by controlling UPP8. Also on this page, counter control and ALU control.

1. Note the three possible ALU control function by analyzing the table in the upper right corner of this print.
2. E26 pin 3 at coordinate B3 goes to data input of UPP8 - state the boolean translation.
3. E46 pin 6 coordinate C6, provides the clocking to UPP8 - state the boolean expression for this.
4. Analyze the gates that provide for the loading and clocking of the counter at coordinates C7 and B7 and give boolean expression.
5. Note the circuit across the bottom of the sheet - provides for Expansion NPR clocking give analysis.

Check your answers with those given in the Appendix.

E. KE-6 Print. Expansion Processor Status Flops and the counter. Analyze the two (2) Status Multiplexers truth tables at the top of this page and verify their selections. Analyze the counter and answer the following questions.

1. Counter inputs
2. Determines direction
3. Counter counts up when
4. A monitoring circuit (E57 pin 12, B5) is provided to inhibit any more counting when count = \_\_\_\_\_.

Check the Appendix to verify your answers.

F. KE-7, 8, 9 Prints. The Expansion ROM, first the 24 extra bits on KE7 then the remaining 44 bits that get encoded to 56 for expanded controlling of the processor on KE-7 & 8. Analyze these three prints and answer the following questions.

1. The clock for Expansion U Word Register
2. Expansion ROM Bit 39 becomes \_\_\_\_\_ as it goes to the CPU.
3. When EIS is activated what are bits 14, 17, 18 a function of?

Check the Appendix to verify your answers.

G. KE-10 - 17 Print. Expansion ROM Listing. The listing and the end of the Logic Diagrams is of course, very similar to the ROM listing in the CPU. I would like to point out the differences and have you note them.

1. To get address 000 (all 80 bits) on one page, they had to split up the listing.
  - (a) CPU top half of each page
  - (b) EIS (24 bits) the bottom half of page

2. All addresses, to be correct, must have 400 added to them.

### III. ADDITIONAL READING

A complete detailed logic description exists in the EIS options manual starting on page 4-48 and continues through 4-57. It would be to your advantage to read this to reinforce those learning activities experienced during this unit.

#### WHAT'S NEXT

After completing this unit, other than some lab projects, you have completed the 11/40 portion of the 11 Processor course. The next processor you tackle will be the 11/45. The units you have covered to date concerning the 11/05 and the 11/40 will become your foundation as you go on. All of the concepts and ideas you have will be put to good use in the future. Good Luck as you continue on.

#### SUPPLEMENTARY INFORMATION

The areas we did not specifically look at in the options manual contain some very good information. This information will be useful to you should you desire more knowledge about the EIS.

APPENDIX

ANSWERS TO PROJECTS/QUESTIONS

A. KE-2 Print

1. DMUX
2. BR REG
3. (a) 0 when doing a ASHC  
(b) COUTI5 when doing a DIV
4. (a) when not MUL, BR00  
(b) when MUL, ALU00
5. (a) 00 - no op  
(b) 01 - right shift  
(c) 10 - left shift  
(d) 11 - load

B. KE-3 Print

1. STRDM(1) , Strobe the RD MUX
2. (a) 00 - EIS Status  
(b) 01 - DR (15:00)  
(c) 10 - BR (14:01), DR15  
(d) 11 - BR (15:00)

C. KE-4 Print

1. Expansion BUT 14-17, for Bits 3 & 4
2. Expansion BUT 10-17, for Bit 2
3. Expansion BUT 10-17, for Bit 1
4. Expansion BUT 0-7, for Bit 1

APPENDIX

ANSWERS TO PROJECTS/QUESTIONS (CONT'D)

D. KE-5 Print

1. (a)  $f = A$   
(b)  $f = A \text{ plus } B$   
(c)  $f = A \text{ minus } B \text{ minus } 1$
2. Reserved instruction \* UPP8 clear
3. (a) Part P end \* BUT 37 (activating EIS)  
(b) Expansion clock \* CLK UPP8 (ROM Bit)  
(deactivating EIS)
4. (a) P2 \* LCNT (ROM Bit) to load counter  
(b) PEND \* count  $\neq 0$  \* ECNT (ROM Bit) to count counter.
5. PEND \* BUT 10

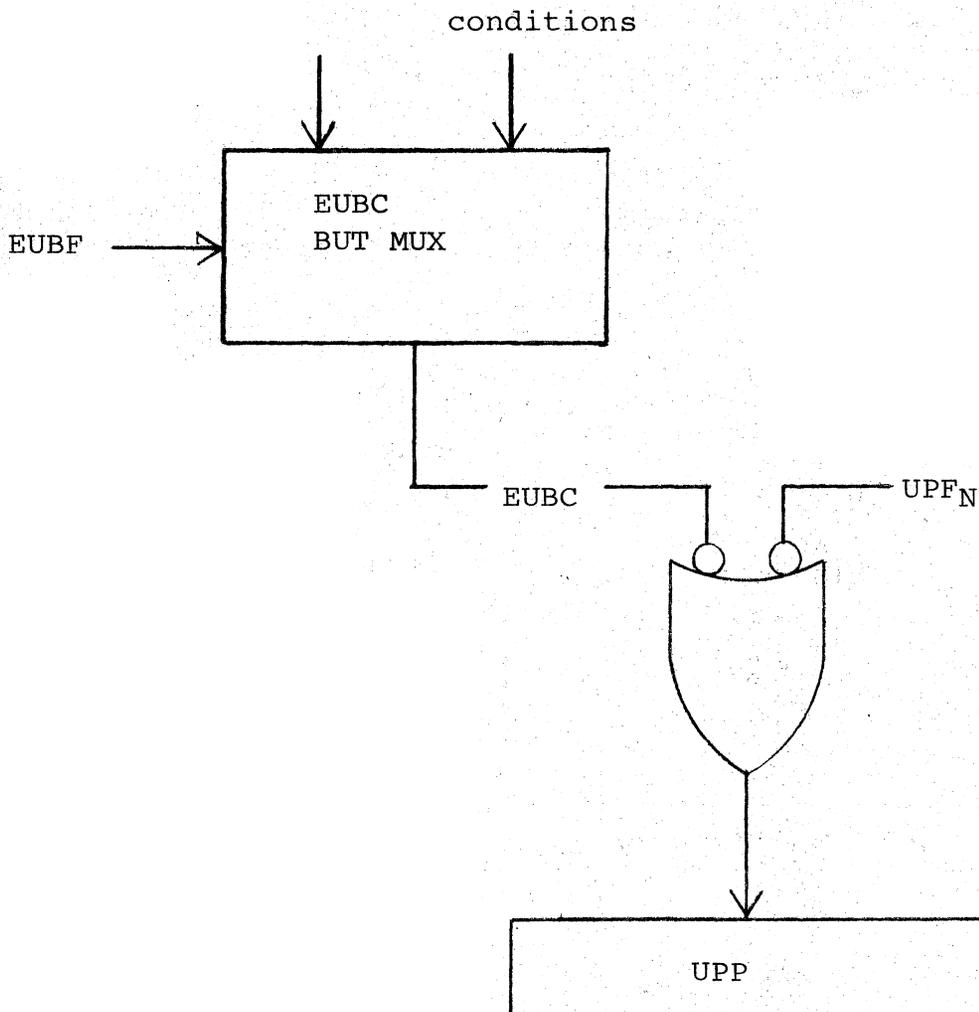
E. KE-6 Print

1. BR Register
2. BR05
3. BR05 = 1, shift direction right
4. 0

F. KE-7, 8, 9 Print

1. CLK EU (88:57)
2. 40 and 39
3. B EUPP8 (is EIS activated)

EIS M7238



LEARNING ACITVITY UNIT

Title: Unibus Interfacing; M105 and M 821

Date: 01/23/74



Rationale

In the previous units you have learned the functions and operations of the 11 processors. You learned the operations of the memories, both 8K and 16K and memory management. Now you have a solid background in the 11 processors.

Of course, the processors can't function very well without input/output devices - the peripherals.

In this unit you will learn how the peripherals are interfaced on to the Unibus. To do this, you will be going over the operations of the M105 address select module and the MF821 interrupt control. These are standard interface modules. You will learn the function and operations of these two modules. You will also learn how to isolate malfunctions in this area.

The knowledge and skills that you acquire in this unit will prove invaluable to you in isolating Unibus interfacing problems dealing with the M105 and M7821.



Prerequisites

Before entering this unit, you must have demonstrated successful completion to this point.

Objectives

Given a list of four failure systems, you must determine the signal most likely to cause that system. To demonstrate successful completion you must answer, three out of the four correctly. You will be allowed to use all available references, but must complete the test in 20 minutes.

## LEARNING ACTIVITIES

## I. Lecture

Attend the lecture on Unibus Interfacing where the Instructor will give an overview on Unibus interfacing modules, M105 and MF821. Pay close attention and take careful notes to highlight key points. Take down any questions you may have and ask them after the lecture.

## II. Projects/Questions

A. M105 address Selector. Refer to the KL11-A Teletype control Eng. Drawings, C-CS-M105-0-1.

1. Analyze the inputs along the bottom of the M105 print. Note the decoding of lines A(12:03) is determined by jumpers. When a given line contains a jumper, the address selector looks for a zero on that line. If there is no jumper, the address selector looks for one.
2. What jumpers would be installed if this M105 were the address selector for the TTY?  
\_\_\_\_\_
3. Line A00 is used for byte control.
4. Line A01 and A02 are decoded to select one of the four addressable device registers.
  - a. If input lines A (02:01) are 01 what select line is asserted?  
\_\_\_\_\_
  - b. If A(02:01) = 11, what select line is asserted?  
\_\_\_\_\_
5. Inputs A00 and BUS C1 and BUS C0 are used to bring up the gating signals. Analyze the logic.
  - a. If C (1:0) = 00, and A00 = 0 what gating control signal comes true?  
\_\_\_\_\_
  - b. If C (2:0) = 10, and A00 = 1, what gating signals come up?  
\_\_\_\_\_

6. These gating signals along with the Select signals generate the functions dealing with the device registers.
    - a. If you were doing DATO to the address of the TKS (777560). You would bring up Select 0 and out low and out high. This would load the TKS with the data.
    - b. What signals would be up for a DATOB at Address 777566 (TPB)?
  7. Analyze the SSYN logic. Note that SSYN is returned to the master 100 ns ec. after the Select line becomes true.
- B. M7821 Interrupt Control Logic. Refer to Drawing D-CS-M7821-0-1, sheet 2 of 2 in the KL11A teletype control Eng. Drawings.
1. Analyze the logic on this print. This logic has the facilities to make bus requests and to become bus master.
  2. Note that the module contains two independent request and grant circuits (A and B)
  3. Locate the signal BUS A BRL at co-ordinate 5D. What signals are used to enable the request?  
\_\_\_\_\_
  4. When the master device (Arbitration logic) recognizes the request and issues a BG, it is input at co-ord 8D. Note how the Grant is propagated through if this is not the requesting device.
  5. Analyze the SACK logic. Note how SACK is asserted when a grant is received.
  6. Analyze the BBSY logic. Notice how the BBSY F/F clocking input is controlled by BUS BBSYL. When the BBSY F/F sets it enables the vector address on the bus and also the BUS INTR signal.
  7. Note the vector address logic in the lower right of the print. You can assign the vector address generated by the uses of jumpers. Install for a 1, remove for a 0.
  8. Notice the jumper at co-ord 3D. This jumper should only be installed if the system is experiencing NPR latency problems.
  9. Analyze the logic and answer the following questions:

- a. What jumpers would be installed to generate vector address 6Ø?  
\_\_\_\_\_
  - b. What state does the BBSY F/F have to be to generate the vector address?  
\_\_\_\_\_
  - c. Does the GRANT F/F set or clear to propagate the BG to the next device if this logic is not the requesting device?  
\_\_\_\_\_
  - d. What signal clocks the BBSY F/F?  
\_\_\_\_\_
10. Check your answers in this unit with the correct answers in the appendix. If you don't understand why you got any questions wrong, consult your instructor. If you feel confident with the material ask your instructor for the POST TEST.

### III Additional Reading

You will find supplementary information regarding this unit in the Peripherals handbook, Chapter 6.

#### What's Next

Now that you are familiar with the basic Unibus interface modules and their concepts of operation; you are ready to investigate the same concepts in an actually interface device, the DL11-A for the TTY. The concepts you learned in this unit will provide the foundation for the next unit.

#### Supplementary Information

If you feel you need more practice, or a better understanding of the M1Ø5 or the M7821, ask your instructor. He will assist you.

APPENDIX

## ANSWERS TO PROJECTS/QUESTIONS

- II.    A     2     Jumper bit F and 3
- 4a    Select 2
- b     Select 6
- 5a    IN H
- b     Out low, out high
- 6b    Select 6
- Out Low
- B     3     A Request 1H
- A Request 2H
- 9a    Jumper 4 and 5
- b     Set
- c     Set
- d     BUS BBSY L going High



11/40 DL11A 240

LEARNING ACTIVITY UNIT  
DL11A THEORY OF OPERATION

1/24/74



PREREQUISITES

Before entering this unit you must have successfully completed the previous units.

OBJECTIVES

Given five statements describing a signal or function of the DL11A, you must be able to match the statement with the answer (signal, register etc) that it best describes. To demonstrate successful completion, you must match four of the five statements correctly. You will be allowed to use all available references, but must complete the test in 20 minutes.

LEARNING ACTIVITIES

I. LECTURE

Attend the short lecture where the instructor will give you an introduction to the theory of operation of the DL11A asynchronous line interface. Pay close attention and take careful notes to emphasize key points. Write down any questions you may have and ask them at the end of the lecture.

II. PROJECTS/QUESTIONS

A. DL11A Teletype Control Block Diagram  
(Refer to fig. 2-3, page 2-8, in DL11 Asynchronous Line Interface Manual)

1. Locate the address selection logic block.

What are the inputs to this block?

---

---

---

2. What register can be selected by this block?

---

---

---

---

3. This block's function is similar to what modules that you covered in the previous unit?

---

4. Locate the interrupt control logic block.

What are the Unibus inputs and outputs of this block?

---

---

---

---

---

---

5. This block's function is similar to what module covered in the previous unit?

---

6. Locate the receiver Status Register and the Receiver Buffer blocks. These blocks would be used when the processor is \_\_\_\_\_ (sending, receiving) information from the TTY.

7. What format is the data sent to the Receiver Buffer from the TTY?

---

8. This is converted to \_\_\_\_\_ data for transmission to the Unibus.

9. Locate the transmitter status register and the transmitter buffer. These blocks control the transfer of \_\_\_\_\_ (parallel, serial) Bus Data to \_\_\_\_\_ (parallel, serial) data for transmission to the TTY.

B. Address Selection - Simplified Diagram  
(Refer to fig. 5-1, page 5-5 in the DL11 Manual)

1. Analyze the simplified diagram for the address selection logic.

Notice that this logic function is similar to the M105 logic operation.

2. The address selection output signals are used to select one of the four registers and determine whether data is to be gated into or out of master device. Refer to table 5-3, page 5-6 in the DL11A Manual for a list of these signals.
3. What function is selected when the processor does a DATI on the RCSR address (777560)?  

---
4. What function is selected when the processor does a DATO to the XBUF address (777566)?  

---
5. What function is selected when the processor does a DATOB to the high byte of the XBUF?  

---

C. Interrupt Control - Simplified Diagram.

1. Analyze the simplified diagram for the Interrupt control, refer to fig. 5-3, page 5-8 in the DL11 Manual.
2. The interrupt control logic permits the DL11A to gain control of the bus and perform an interrupt operation. The Vector Address for the DL11A is 060 and 064. The BR level is 4.
3. What signals are used to enable a transmitter interrupt?  

---

---

4. What signals are used to enable a Receiver Interrupt?
- 
- 

5. Once the interrupt sequence is started, it is the same as in the M7821 module.

D. Device Register Functions  
(Refer to table 5-5, page 5-11 in the DL11 Manual)

1. What register holds the data received from the processor for transfer to the TTY?
- 

2. What register holds the data received from the TTY for transfer to the processor?
- 

3. What register provides information on the status of the receiver logic?
- 

E. Device Register Bit Assignments

1. Receiver Status Register (RCSR)  
(Refer to fig. 4-1, page 4-2 in DL11 Manual)

NOTE:  
Bits 15-12, 10-8, 5-1 are not used in the DL11A

2. Receiver Buffer Register (RBUF)  
(Refer to fig. 4-2, page 4-4 in DL11 Manual)

NOTE:  
Bits 15-8 are not used in the DL11A

3. Transmitter Status Register (XCSR)  
(Refer to fig. 4-3, page 4-6 in DL11 Manual)

NOTE:  
Bits 15-8, 5-3, and 1-0 are not used in DL11A

4. Transmitter Buffer Register (XBUF)  
(Refer to fig. 4-4, page 4-6 in DL11 Manual)

NOTE:

Bits 15-8 are not used

F. Universal Asynchronous Receiver/Transmitter (UART)

1. The UART is a full duplex receiver/transmitter. The receiver section accepts serial binary characters and converts them to a parallel format for transmission to the Unibus. The transmitter section accepts parallel binary character from the bus and converts them to a serial output with starts and stop bits added.
2. The data format for the DL11A consists of a START bit, eight DATA bits, and two STOP bits. (Refer to fig. 2-1, page 2-4 in the DL11 Manual) Parity is not used in the DL11A.  
Refer to table 2-3, page 2-5 in DL11 Manual
3. Complete the table below for jumpering the DL11A for a START bit, 8 DATA bits, 2 STOP bits, and no parity. Indicate IN or OUT.

JUMPERS	NP	EP5	2SB	J9	J11	J10	NB1	NB2

G. UART Receiver - Block Diagram  
(Refer to fig. 5-5, page 5-23 in DL11 Manual)

1. Read paragraph 5. 7. 1 on Receiver operation and analyze the block diagram
2. Note that on the DL11A, the overrun, parity, and frame error are not used.
3. How does the UART Receiver know when the entire data character has been shifted into the register?

4. After the character is in the shift register it is then transferred in \_\_\_\_\_ (parallel, serial) form to the holding register and the \_\_\_\_\_ flag set.

H. UART Transmitter - Block Diagram

Refer to fig. 5-6, page 5-24 in DL11 Manual

1. Read paragraph 5.7.2 on Transmitter operation in DL11 manual and analyze the block diagram.
2. What signal loads the parallel data bits in to the data buffer?  
\_\_\_\_\_
3. What block determines the format of the serial data?  
\_\_\_\_\_
4. What signal prevents the timing generator from loading another character into the shift register while the present character is being shifted into the TTY?  
\_\_\_\_\_
5. What flip flop indicates when the transmitter is ready for a character?  
\_\_\_\_\_

I. Clock Logic fig. 5-7, page 5-26 in DL11 Manual

1. Analyze the frequency divider logic simplified diagram.

NOTE:

The DL11A uses an 8.44.8KH crystal oscillator, so turn to page 2-3 and substitute the correct baud rates for this crystal and the appropriate switch positions in the fig. 5-7. The standard baud rate for the teletype is 110 for both receiver and transmitter so the correct switch position is no. 3.

2. The frequency selected by the transmitter switch is \_\_\_\_\_ signal which is used to generate the transmitter clock input.

3. The frequency selected by the receiver switch is \_\_\_\_\_ and is applied to the receiver clock input.
4. For a detailed description of the clock logic, read paragraph 5.8, page 5-25 in the DL11 Manual.
5. Check your answers in this unit with the correct answers in the appendix. If you do not understand why you got any questions wrong, consult your instructor. If you feel confident with the material, ask your instructor for the POST TEST.

### III. ADDITIONAL READING

You will find supplementary information regarding this unit in the DL11 Asynchronous line interface manual.

#### WHAT'S NEXT

Now that you have the theory of operation of the DK11A under your belt, you are ready to investigate the detailed logic prints. The theory you learned in this unit will provide the background for detailed analysis of the DL11A logic.

#### SUPPLEMENTARY INFORMATION

If you feel you need more practice, or a better understanding of the material covered in this unit, see your instructor. He will help you to enhance your understanding of the material.

APPENDIX

ANSWERS

- II A 1 A<17:00>,C<1:0>, MSYN
- 2 Receiver Status Register  
Receiver Buffer Register  
Transmitter Status Register  
Transmitter Buffer Register
- 3 M105
- 4 BBSY, SSYN, SACK, BR, BG, INTR
- 5 M7821
- 6 Receiving
- 7 Serial
- 8 Parallel
- 9 Parallel, Serial
- B 3 Receiver status to the bus
- 4 Bus Data to transmitter buffer
- 5 No function, must be a DATOB to low byte only
- C 3 XMIT INT ENB (1)H  
XMIT DONE H
- 4 RCVR INT ENB (1)H  
RCVR DONE H
- D 1 XBUF
- 2 RBUF
- 3 RCSR
- F 3

NP	EPS	2SB	J9	J11	J10	NB1	NB2
OUT	*(IN OR OUT)	OUT	OUT	OUT	IN	OUT	OUT

\* DOES NOT MATTER SINCE NP IS OUT

APPENDIX

ANSWERS

- G. 3. Occurance for a STOP bit
- 4. Parallel, Data available
  
- H. 2. DATA STROBE
- 3. Control Logic block
- 4. END OF CHARACTER
- 5. TBMT F/F
  
- I. 2. XMIT CLK H
- 3. RCVR CLK H

11/40-DL11-A-250

LEARNING ACTIVITY UNIT

DL11-A LOGIC PRINTS

01/25/74



RATIONALE

In the previous unit you learned the theory of operation of the DL11-A asynchronous line interface. You saw how the interface handled data conversion from parallel to serial and from serial to parallel for input to and output from the Teletype.

In this unit you will learn how the logic was designed and set-up to implement the operation. You will learn the names and locations of key signals into and out of these logic areas. You will also learn how to isolate problems in these areas.

The knowledge and skills that you acquire in this unit will prove invaluable in isolating and solving problems in the DL11-A and related interfaces.



PREREQUISITES

Before entering this unit you must have demonstrated successful completion to this point.

OBJECTIVES

Given a list of four failure symptoms, you must be able to determine the correct malfunction that would cause the symptom. To demonstrate successful completion of this unit you must get three of the four questions correct.

LEARNING ACTIVITIES

## I. LECTURE

Attend the 20 minute lecture where the instructor will give an overview of the DL11-A prints. Pay close attention and take careful notes to highlight the key signals in the logic. Take down any questions you have and ask them after the lecture.

## II. PROJECT/QUESTIONS

## A. DL-2 Print

1. Analyze the Base Data Receivers at co-ord FCD. These gates gate the bus data D 7:00 into the DL11. Note that there is no gating signal.
2. Analyze the Bus Drivers at co-ord 5CD. These gates gate the contents of the Read Buffer to the Bus. What is the gating signal that performs this function? \_\_\_\_\_
3. Analyze the Bus Drivers at co-ord 3CD. These gates applied the Transmitter Status Register to the Bus. What is the gating signal that performs this function? \_\_\_\_\_
4. Analyze the Bus Drivers at co-ord 2CD and 2B. These gates applied the Receiver Status Register to the Bus. What is the gating signal to do this? \_\_\_\_\_
5. Analyze the logic at co-ord 5-6B. This logic is used to gate the Receiver Buffer to the bus. However, in the DL11-A jumper J8 is out, so this logic is disabled.
6. Also, note jumper JF at co-ord 4C. This jumper is out for the DL11-A.

## B. DL-3 Prints

1. Analyze the crystal oscillator logic at co-ord 7C. In the DL11-A the crystal is 844.8 KHz. Note that the output of the crystal Y1 is applied to four IC chips that function as frequency dividers to provide the eight different frequencies fed to the rotary switch.
2. Refer to Fig. 5-7 in the DL11 manual, page 5-26 for the simplified diagram of the frequency dividers.

3. The DL11-A function with the rotary switch in Position 3 for both XMIT CLK and RCVR CLK. This gives you a band rate of \_\_\_\_\_.
4. Note the signal BERG SERIAL IN L at co-ord 8B. This is the serial input from the TTY.
5. Note the signal SERIAL OUT H at co-ord 8A. This is applied to a circuit (2-3B) that converts the line to a bipolar level required by the TTY. This sends the serial data to the TTY.
6. Note the priority plug at co-ord 2D. This plug assigns the BR4 level to the interrupt logic.

#### C. DL-4 Print

1. Locate chip E44 co-ord 8B. This is a four bit shift register that checks for a legal start pulse during receiver operation. A legal start pulse would be low and that low shifted in at one-half the 567 user rate, until all the register bits are 0's then the RCVR ACT F/F (co-ord 7c) sets.
2. Analyze the UART at co-ord 5C. Note the serial in, pin 20, this is the serial data input.
3. What are the inputs to the XBUF?  
\_\_\_\_\_
4. What are the output of the RBUF?  
\_\_\_\_\_
5. Locate chip E62, co-ord 3C. This is the RDONE bit.

#### D. DL-5

1. The logic on this print functions similar to the M105 logic.
2. What jumpers would be installed to assign address 777560 thru 777566?  
\_\_\_\_\_
3. What signal is used to load bus data into the XBUF?  
\_\_\_\_\_

E. DL-6 Prints

1. The logic on this print is the interrupt logic. It's function is similar to the M7821.
2. Note that the vector addresses are jumpered for A "1".
3. Check your answers in this unit with the correct answers in the appendix. If you have any questions, consult your instructor. If you feel confident with the material take the POST TEST.

III. ADDITIONAL READING

You will find supplementary information regarding this unit in the DL11-A manual.

WHAT'S NEXT

Now you should have a good background in the DL11-A interface. The concepts you learned in **this** unit will prove invaluable in isolating interface malfunctions. In the next unit you will be covering the KW11 Real-Time clock.

SUPPLEMENTARY INFORMATION

If you feel you need more help on anything covered in this unit, consult your instructor.

APPENDIX

Answers to Project/Questions

II.

- A. 2. RBUF to BUS H  
3. XCSR to BUS H  
4. RCSR to BUS H
- B. 3. 11Ø
- C. 3. BD 7:Ø  
4. RD 7:Ø
- D. 2. A3, A7  
3. BUS to XBUF H





## digital

DIGITAL EQUIPMENT CORPORATION, Maynard, Massachusetts, Telephone: (617) 897-5111 • ARIZONA, Phoenix • CALIFORNIA, Sunnyvale, Santa Ana, Los Angeles, San Diego and San Francisco (Mountain View) • COLORADO, Engelwood • CONNECTICUT, Meriden • DISTRICT OF COLUMBIA, Washington (Riverdale, Md.) • FLORIDA, Orlando • GEORGIA, Atlanta • ILLINOIS, Northbrook • INDIANA, Indianapolis • LOUISIANA, Metairie • MARYLAND, Riverdale • MASSACHUSETTS, Cambridge and Waltham • MICHIGAN, Ann Arbor and Detroit (Southfield) • MINNESOTA, Minneapolis • MISSOURI, Kansas City and Maryland Heights • NEW JERSEY, Fairfield, Metuchen and Princeton • NEW MEXICO, Albuquerque • NEW YORK, Huntington Station, Manhattan, New York, Syracuse and Rochester • NORTH CAROLINA, Durham/Chapel Hill • OHIO, Cleveland, Dayton and Euclid • OKLAHOMA, Tulsa • OREGON, Portland • PENNSYLVANIA, Bluebell, Paoli and Pittsburgh • TENNESSEE, Knoxville • TEXAS, Dallas and Houston • UTAH, Salt Lake City • WASHINGTON, Bellevue • WISCONSIN, Milwaukee • ARGENTINA, Buenos Aires • AUSTRALIA, Adelaide, Brisbane, Crows Nest, Melbourne, Norwood, Perth and Sydney • AUSTRIA, Vienna • BELGIUM, Brussels • BRAZIL, Rio de Janeiro, Sao Paulo and Porto Alegre • CANADA, Alberta, Vancouver, British Columbia; Hamilton, Mississauga and Ottawa, Ontario; and Quebec • CHILE, Santiago • DENMARK, Copenhagen and Hellerup • FINLAND, Helsinki • FRANCE, Grenoble and Rungis • GERMANY, Cologne, Hannover, Frankfurt, Munich and Stuttgart • INDIA, Bombay • ISRAEL, Tel Aviv • ITALY, Milano • JAPAN, Osaka and Tokyo • MEXICO, Mexico City • NETHERLANDS, The Hague • NEW ZEALAND, Auckland • NORWAY, Oslo • PHILIPPINES, Manila • PUERTO RICO, Miramar and Santurce • REPUBLIC OF CHINA, Taiwan • SCOTLAND, West Lothian • SPAIN, Barcelona and Madrid • SWEDEN, Solna and Stockholm • SWITZERLAND, Geneva and Zurich • UNITED KINGDOM, Birmingham, Bristol, Edinburgh, London, Manchester, Reading and Warwickshire • VENEZUELA, Caracas