EDUCATIONAL
SERVICES

VAX/VMS
Training

VAX/VMS
Device Driver
Student Guide

digital

EY-2278E-SG-0001

VAX/VMS DEVICE DRIVER

STUDENT GUIDE

Prepared by Educational Services
of
Digital Equipment Corporation

First Edition, October 1984

Printed in U.S.A.

The manuscript for this book was created using DIGITAL Standard Runoff. Book production was done by Educational Services Development and Publishing in Nashua, NH.

CONTENTS

COURSE DESCRIPTION

This course is designed for experienced system programmers who need to write or modify UNIBUS or MASSBUS device drivers under the VAX/VMS system. In addition, students will learn about the I/O architecture used by VMS, and will be exposed to several driver-debugging techniques.

PREREQUISITE

Fluency in the VAX-11 MACRO language, and successful completion of the one week VAX/VMS Operating System Internals Course.

COURSE GOALS

Upon completion of this course, the student will be able to:

1. Identify the data structures and routines of a VAX/VMS device driver.
2. Identify a typical sequence of events that take place in the system from the issuance of an I/O request to its completion.
3. Write a UNIBUS or MASSBUS device driver; assemble, link, and incorporate it into the system.
4. Debug a device driver using debugging tools XDELTA and SDA.

RESOURCES

1. Student Package
2. VAX-11 Hardware Handbook
3. VAX Instruction Card
4. Guide to Writing a Device Driver for VAX/VMS

In addition, the following manuals will be available for your reference:

1. VAX-11 Architecture Handbook
2. VAX-11 Software Installation Guide
3. VAX/VMS System Management and Operator's Guide
4. VAX/VMS Release Notes
5. VAX/VMS I/O User's Reference Manual
6. UNIBUS Adapter Technical Description (ED-DW780-TD-OU1)
7. VAX/VMS System Dump Analyzer Reference Manual
8. VAX/VMS System Dump Analyzer Reference Manual

COURSE ORGANIZATION

This course is presented in a lecture/lab format. The lecturer will reference the materials in this student package. Lectures may consist of instructor presentation, class discussions, or directed individual study. The lab time will be used for demonstrations by the instructor, hands-on experience for the students, and the working of exercises and tests.

The course material is structured within modules. Each module is a unique lesson on one or more of the skills required in writing a driver.

A module consists of:

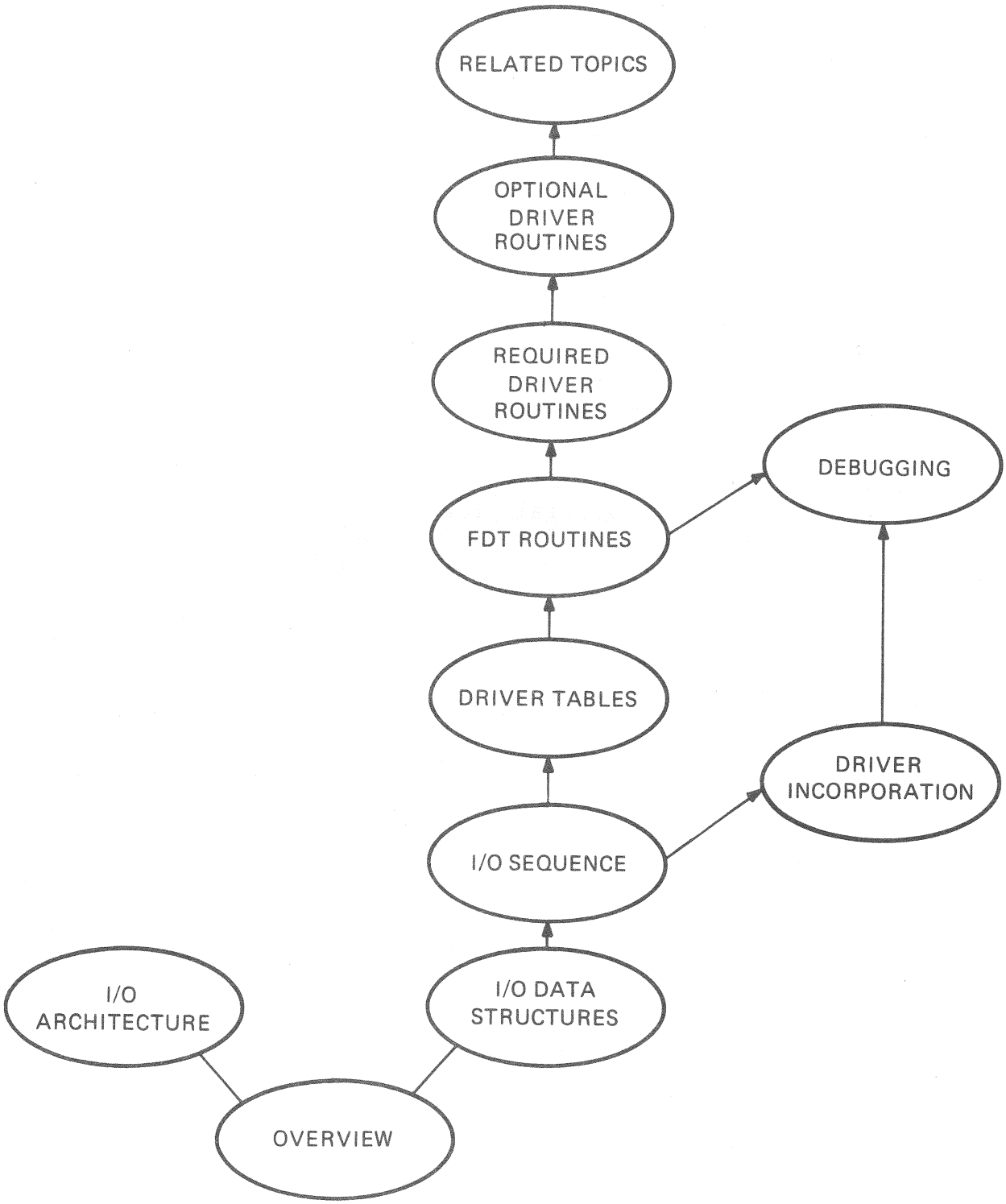
- An **introduction** describing the purpose of the lesson.
- At least one **objective** which states what you will know or be able to do when you complete the module.
- **Additional resources** that provide supplementary reading and/or reference material for the module. You will receive your own copy of most of these. All others will be available for your use during the labs.
- The **module text** consists of examples, reference notes, and copies of any visuals used by the instructor. In terminal printouts, any user input will be underlined.
- A **module test**, which is bound separate from the text. Tests may be paper and pencil, lab-oriented, or both. By comparing your response with the **answer sheet**, you can determine whether or not you have met the objective(s) of the module. If you cannot pass the test, you should consult with your instructor for additional help.

In general, you should familiarize yourself with the reference material and module text prior to attending the related lecture period. Sometimes, an instructor may choose to give a specific assignment.

COURSE MAP DESCRIPTION

The course map shows the relationship among the various modules. Those modules having arrows leading into other modules are defined as prerequisites for that module. You should complete all the prerequisites for a module before you begin studying its material.

COURSE MAP



TK-4866

COURSE OUTLINE

INTRODUCTORY MATERIAL

- Overview
- I/O Data Structures
- I/O Sequence

DRIVER WRITING

- Driver Tables
- FDT Routines
- Required Driver Routines
- Optional Driver Routines

DRIVER TESTING

- Driver Incorporation
- Debugging

SUPPLEMENTARY MATERIAL

- Related Topics
- I/O Architecture

APPENDIX A
Acronyms

ACP = Ancilliary Control Processor

ADP = Adapter Control Block

AST = Asynchronous System Trap

CCB = Channel Control Block

CRB = Channel Request Block

DDB = Device Data Block

DDT = Driver Dispatch Table

DPT = Driver Prologue Table

FDT = Function Decision Table

IDB = Interrupt Data Block

IPL = Interrupt Priority Level

IRP = I/O Request Packet

MBA = MASSBUS Adapter

PCB = Process Control Block

QIO = Queue I/O System Service

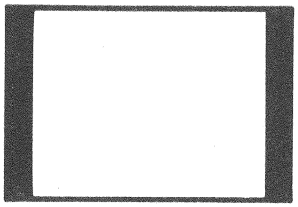
UBA = UNIBUS Adapter

UCB = Unit Control Block

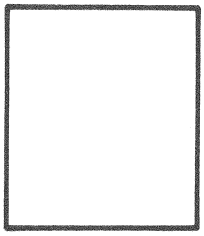
WCB = Window Control Block

APPENDIX B

Symbols

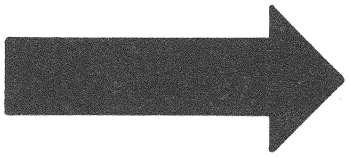


CODE,
MEMORY–RESIDENT ROUTINE
IMAGE



DATA STRUCTURE

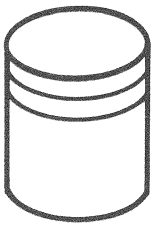
TRANSFER OF CONTROL (JSB/RSB, CALL/RET,
AST, ETC.)



TRANSFER OF DATA

INTERRUPT

POINTER, LINK OR INDEX



FILES, NON–RESIDENT IMAGES, ETC.
(RIGID MEDIA)

TK-4819

digital

EY-2278E-SG-0001
Printed in U.S.A.