INTERACTIVE

A Kodak Company

VP/ix Technical Guide

First printing (August 1988) PF073-00-0

VP/ix Technical Guide

CONTENTS

VP/ix Technical Guide

1. INTRODUCTION

This document provides a technical description of the VP/ix™ Environment which has been jointly developed by INTERACTIVE Systems Corporation and PHOENIX Technologies Ltd. It is intended for use by OEM customers who wish to understand the internals of the VP/ix Environment, or to customize it. For a higher-level, less technical description of the VP/ix Environment, see the *VP/ix Product Brief*. It is assumed throughout this document that the reader is familiar with the architecture of the Intel® 80386™ microprocessor.

1.1 Functional Description of the VP/ix Environment

The VP/ix Environment provides users of Intel 80386–based computer systems with the ability to run most MS-DOS® (referred to in this guide as DOS) applications under UNIX®. Initial versions of the product operate under UNIX System V Release 3 on 80386 hardware environments that are otherwise compatible with the IBM® PC AT®. However, the VP/ix Environment is expressly designed to be portable to *any* single-user or multiuser 80386–based UNIX system. A MULTIBUS™ version of the product is also available for customization by OEM's. That version is fully functional except for certain hardware-dependent features, such as diskette copy protection support, that are likely to change from system to system. Graphics applications require the appropriate display hardware, and are currently supported only on the PC AT version of the product. However, text-only applications will run under the VP/ix Environment on serial terminals as well as on memory-mapped video displays. Any serial terminal with 24 or 25 visible lines is supported.

Multiple DOS applications, including multiple instances of the same application, can run concurrently with conventional UNIX processes under the VP/ix Environment. The UNIX kernel provides the same efficient interrupt-driven scheduling services to DOS applications and UNIX applications alike, making the VP/ix Environment a true multitasking system.

The VP/ix combination of DOS and UNIX System V.3 provides the following features:

- Support for all facilities and applications normally available under UNIX.
- Support for all facilities and applications normally available under DOS, including support for the unusual hardware accesses required by *ill-behaved* programs.
- Transparent integration of the DOS and UNIX file systems. Any program, regardless of whether it is a DOS or a UNIX program, can access (and share) any DOS or UNIX file without concern for the boundaries between file systems or disk partitions. No partitioning is required.
- The ability for each user to run DOS and UNIX sessions simultaneously in a paged virtual memory environment. Each session runs in its own secure virtual address space.
- The ability to run DOS programs from a UNIX session and UNIX programs from a DOS session.
- Access to DOS applications from any serial terminal, not just the system console.
- Virtual screen support on the PC console permits multiple DOS and UNIX applications for the same user to run as though each one were in full control of its own screen. Up to eight applications running on virtual screens may share a physical screen. The user switches the physical screen from one virtual screen to another with a keystroke. Virtual screens are not supported on serial terminals.
- Built-in emulation support for standard IBM PC AT peripherals, including graphics adapters, mouse, modems, printers, etc.
- Simulated direct access to IBM–compatible peripherals for ill-behaved DOS programs on non–compatible systems, via a well-defined generic driver interface that facilitates the porting of the VP/ix Environment to new systems.

- Optional provision of 2MB of virtual EMS (Lotus®-Intel-Microsoft®'s *E*xpanded *M*emory *S*tandard) to any DOS application, regardless of a system's actual physical memory configuration.

1.2 Components of the VP/ix Environment

The VP/ix Environment consists of several functional areas. These areas are outlined briefly here, and discussed in detail in the remainder of this document.

1.2.1 VP/ix Emulation Control Process Support. The VP/ix Environment allows DOS and its applications to run under UNIX by creating a virtual machine in which all the standard layers of a real PC, i.e., the 8086 microprocessor, the surrounding hardware environment, and the BIOS, are available in either real or emulated form. A VP/ix virtual machine consists of a user-level program called `vpix` and some supporting modifications in the UNIX kernel. The bulk of the emulation in the VP/ix Environment is done in `vpix` rather than in the kernel.

`vpix` runs as a special kind of UNIX process called a *dual-mode* process. A dual-mode process spends part of its time in Virtual 8086 mode running DOS and its applications, and the rest of its time in Protected mode acting as the virtual machine monitor for its "other self." Each half of the process is a separate task in the 80386 hardware sense, in that each has a separate TSS. Virtual machines and dual-mode processes are one-to-one.

The technical details of dual-mode process support are discussed in section 2.

1.2.2 Access to Devices. A complete virtual PC must provide virtualized hardware support for the standard IBM PC environment. This includes such things as the keyboard, displays, serial lines, and parallel ports, but if ill-behaved programs are to be supported, it must also include timers, interrupt controllers, diskette controllers, and DMA controllers. The VP/ix Environment emulates all standard PC hardware, and runs all popular DOS applications, ill-behaved or not.

Device emulation is shared between `vpix` and kernel drivers.

There is also a class of devices to which it is safe to allow DOS drivers direct access, without any intervening emulation or UNIX driver. The VP/ix Environment provides a means to allow such access.

The VP/ix Environment also provides a way to add emulation for more complex non-standard devices such as high-resolution displays and network cards. Adding device emulations to VP/ix closely parallels (and sometimes requires) adding device drivers to the UNIX kernel; thus it is a task for the OEM or other hardware manufacturer, rather than the end user.

The details of device access, including how to package a set of drivers for a non-standard device for user installation and customization of the VP/ix Environment, are discussed in section 3.

1.2.3 Channels. The VP/ix Environment provides a mechanism whereby DOS applications which "know" they are running under VP/ix may communicate with the virtual machine monitor, and through it, with other UNIX processes. Thus an OEM may add communications code to a DOS application they are writing, and similar code in a pseudo-device emulation module they add to the virtual machine monitor, and use this communications channel to pass information back and forth to other UNIX processes using normal UNIX inter-process communications facilities. Or the channel may be used for the DOS application to request services from the virtual machine monitor.

The specification of this mechanism is given in section 4.

1.2.4 User Interface. The VP/ix pop-up User Interface Menu allows users to communicate directly with `vpix`. This is discussed in section 5.

1.2.5 Cross-Execution Facilities. The VP/ix Environment permits users to execute DOS programs directly from the Bourne or C shells, and UNIX commands from the DOS shell (`COMMAND.COM`). These facilities are discussed in section 6.