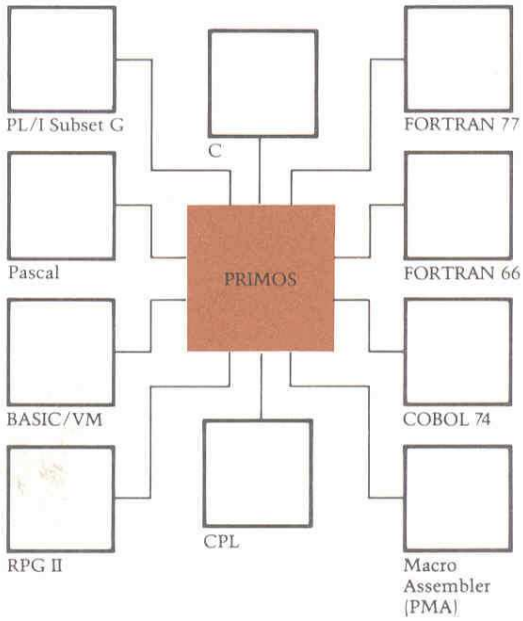
PRIME

PRIMOS® Operating System (Rev. 19)

Features

- Efficient multipurpose, multiprogramming operating system.
- Fully compatible across all Prime 50 Series™ systems.
- Uses the advanced architectural features of Prime 50 Series™ systems.
- Sophisticated virtual memory mechanism employing both segmentation and paging.
- 64Mb maximum program size.
- Embedded re-entrant operating system.
- Hardware memory protection system.
- Shared translators, utilities and libraries.
- Program development support and automatic shared, re-entrant, recursive program execution.
- Dynamic linking to operating system and shared user and system libraries.
- Up to 254 asynchronous terminals and 255 processes.
- Up to eight synchronous lines.
- Up to 16Mb maximum physical memory.
- Over 10Gb maximum mass storage.
- Multilevel hierarchical file structure with access control list protection.
- Sequential and direct file access methods.
- Dynamic file and paging disk space allocation with optional usage quotas.
- Comprehensive, easy-to-use, command line processor and procedure language.
- Exception handling via an ANSI PL/I standard condition mechanism.
- Per-user abbreviation facility.
- Inter-user message facility.
- Online HELP facility
- Advanced communications support.
- Indexed sequential access method and CODASYL-compliant DBMS.
- Full complement of utilities for disk maintenance, system backup and program development.



Description

All Prime® virtual memory computer systems – from the small Prime 2250™ to the Prime 9955™ that supports dozens of concurrent timeshared and queued tasks – use the multifunction PRIMOS operating system. Since each central processor provides a different level of performance and functionality within the Prime systems family, the PRIMOS operating system maximizes the effectiveness of a processor's resources while minimizing operating system overhead.

The PRIMOS operating system further optimizes the high-speed computational ability and exceptionally large memory capacity of Prime central processors by integrating both interactive and batch supervisory services into a single operating system.

PRIMOS is structured to take advantage of the advanced architectural features of Prime 50 Series systems such as virtual memory, process exchange, dynamic linking and hardware memory protection. This means PRIMOS provides a productive and easy-to-use environment for program development and an efficient, secure, reliable environment for program execution. Furthermore, as the same operating system runs on all Prime 50 Series systems, this environment is fully compatible across the product line.

The PRIMOS operating system supports a wide range of communication facilities including seven Remote Job Entry (RJE) products; Prime's Distributed Processing Terminal Executive (DPTX) and Prime/SNA™ for IBM connectibility; and PRIMENET™ networking software. Also supported are MIDASPLUS™ and PRISAM™ indexed sequential file access methods, Prime DBMS, a CODASYL-compliant database management system; and a variety of industry-standard languages. The PRIMOS operating system supports a wide range of peripherals including over 10Gb of mass storage, eight tape drives, four parallel line printers and multiple serial printers, card readers, paper tape and graphics devices.

Compatibility

Since the PRIMOS operating system runs on all Prime 50 Series systems, programs and data files created on one Prime system can be used on any other without modification. This compatibility holds true at both the source language level and at the object code and memory-image levels. Programs running on a small Prime system can run unchanged and substantially faster on larger Prime systems. This preserves software investments and provides an attractive upgrade and growth potential. Large Prime systems can be used for efficient software development, creating programs and databases that are easily transferred to smaller distributed processing systems.

Performance

Prime's system architecture was developed in a unique way – the software was designed first. The hardware was then designed to support the software, with many of the traditional operating system functions implemented in microcode and hardware. Examples include the process exchange mechanism, the hardware procedure-call mechanism and support for dynamic linking. This "software first" design philosophy manifests itself in performance and reliability. The microcode and hardware assist allows PRIMOS to maximize system performance, particularly in a multi-user environment.

Ease-of-use

The PRIMOS operating system provides a productive, easy-to-use environment for the application developer. Advanced features such as command iteration, wildcarding, treewalking, name generation, Command Procedure Language and the abbreviation processor let users customize command environments. The condition mechanism allows an application designer to ensure that users remain within the application. PRIMOS provides an extensive HELP facility for user assistance on all system commands.

The PRIMOS operating system is easy to configure and bring up. During start-up, it automatically configures itself according to the hardware installed on the system. Simple commands, in a configuration file used at system start-up, control additional options that can be configured within the PRIMOS operating system.

Security and Integrity

Hardware, firmware (microprogrammed logic) and software components within PRIMOS monitor the complete hardware/software system to ensure reliable hardware operation and secure process execution. System integrity is maintained by single-bit, main memory, error correction and microverification routines that test the central processor's logic and help determine the cause of any faulty operation. A hardware-implemented ring protection mechanism protects memory from accidental or unauthorized access, preserving the integrity of the operating system, its databases and shared code.

File system security is based on Access Control Lists (ACL). ACLs allow access rights to any directory or file specified on a per-user basis. User access to the system is controlled through a secure log-in mechanism that also records unsuccessful attempts to access the system. The operating system also includes file access integrity features such as forward and backward pointers, and utilities to repair damage or inconsistencies.

Virtual Memory

The advanced virtual memory management in the PRIMOS operating system supports multiple concurrent processes, each with its own private virtual memory space. Additional virtual memory space is shared among all processes. This mechanism takes advantage of both segmentation and paging to provide users with an extremely large address space, eliminating concern over program size limitations (a 64Mb program can be run in a user's private address space). Virtual memory resources are available on systems with as little as 512Kb of main memory. The size of the user's virtual address space is independent of physical memory size. PRIMOS automatically takes advantage of additional increments of main memory (up to 16Mb) to reduce paging as system load increases.

Embedded Operating System

The PRIMOS operating system is exceptionally responsive, providing direct and immediate control because it is effectively embedded in the virtual address space that all users share. It is an integral part of each user's process, and executes for all users at the same time. The shared, re-entrant embedded design of the PRIMOS operating system allows it to service user requests with minimum overhead and delay. Users can access an operating system resource in no more time than it takes a user program to call a subroutine.

Procedure Sharing

Memory utilization is greatly improved with PRIMOS because one user can write procedures that others can automatically share. Prime systems make maximum use of procedure sharing. That means a shared procedure exists no more than once on disk and once in memory regardless of the number of users accessing it. Prime's system architecture provides for separation of "pure" re-entrant code, static storage and dynamic storage. A stack architecture provides a re-entrant, recursive environment for program execution. Prime's shared high-level language translators generate re-entrant code. All high-level language libraries are also shared. Programs are dynamically linked to shared libraries and PRIMOS operating system services at runtime.

Data Communication/Networking

PRIMOS handles all data communication between a Prime system and interactive terminals, other Prime central processors and mainframes. The PRIMOS operating system communicates directly with most currently available asynchronous ASCII terminals operating at speeds of up to 19.2K baud. Communication between Prime systems is handled via PRIMENET Node Controllers (PNCs) for locally connected processors, or through high-speed synchronous lines or public data networks, using the CCITT X.25 standard packet switching protocol.

DPTX

The Prime Distributed Processing Terminal Executive (DPTX) software product allows users to construct interactive communication networks with equipment provided by Prime and IBM. DPTX products conform to IBM 3271/3277 Display System protocols, and can be integrated into new or existing networks containing IBM or IBM-compatible mainframes and terminal controllers without changing application code or access methods.

Prime/SNA

Prime/SNA is a family of subsystems which allows Prime computer systems to coexist with networks based on IBM's System Network Architecture (SNA). The Prime/SNA Interactive Subsystem enables Prime systems to emulate 3274/3278 control units and display stations, while the Prime/SNA RJE subsystem allows Prime systems to support the features of 3776 RJE stations. In addition, Prime/SNA works with PRIMENET to extend Prime/SNA capabilities across multiple Prime systems and networks.

PRIMENET

The PRIMENET distributed networking facility provides complete local and remote network communication services for all Prime systems. In geographically-dispersed network configurations, it allows Prime computers to communicate with other Prime computers, with computers from other vendors, and with terminals and computers attached to packet-switching networks. In local area network configurations, PRIMENET attaches Prime computers to each other via a high-bandwidth, multipoint ring arrangement.

Languages

The Prime family of interactive systems provides a productive, easy-to-use environment for program development in a wide variety of languages. All Prime 50 Series systems support COBOL 74, BASIC/VM, FORTRAN 66 and 77, PL/I Subset G, RPGII V-Model Compiler, Pascal, C and Prime Macro Assembler (PMA).

Prime languages employ a common procedure call architecture. That means programs written in one Prime language can call routines written in another, saving program development time by utilizing existing routines written in other Prime languages.

A Source-Level Debugger is available for most Prime languages, allowing users to step through their source-level code and exert interactive control over all aspects of program execution. They can set or clear breakpoints on any statement, examine or modify variables, evaluate expressions, execute single statements and trace execution at will. The generic capabilities of the Source-Level Debugger are unique and promote user convenience in a multi-language environment.

Data Management

Prime DBMS

Prime DBMS database management system conforms to CODASYL standards for database design, minimizing the need to store and synchronize multiple copies of the same data. Prime DBMS allows users to access a database concurrently, while ensuring that the integrity of the database is maintained. Prime DBMS provides data security, automatic recovery and reorganization facilities. It is ideally suited for data processing environments characterized by interactive data processing applications that require complex data relations and online transaction processing.

Prime DBMS can lead to reduced application programming expenses and shorter development times. Programmers can concentrate on the logic of the application, not the details of data manipulation and file design. When implemented with DISCOVER,™ the complementary nonprocedural query and reporting language, many ad-hoc requirements can be offloaded from the data processing organization to users.

Multikeyed ISAMs

Prime offers two distinct Indexed Sequential Access Method (ISAM) products. MIDASPLUS is a high-performance data manager which allows interactive users to create and maintain their own data structures. Data can be accessed through up to 18 different key paths via all standard Prime programming languages and through the companion query and report writer, PRIME/POWER+.

PRISAM, Prime's Recoverable ISAM, is a unique data manager featuring automatic recovery and concurrency control while maintaining a simple file structure. PRISAM allows interactive users to create and maintain data files with up to 24 different key paths. File access can be gained through any of Prime's standard programming languages, as well as through the companion DISCOVER query and report writer.

Query and Report Writers

PRIME/POWER+ is a user-oriented data query and reporting language which supports MIDASPLUS as well as ASCII, Direct Access and Binary file types. It provides fast, flexible access to data. PRIME/POWER+ offers multi-file query, report generation, text processing, forms writing, data entry and data maintenance.

DISCOVER is a comprehensive data query and reporting language that supports both PRISAM and Prime DBMS file structures. DISCOVER allows ad-hoc queries, report generation and PRISAM record manipulation. It employs relational techniques on traditional files. Because non-programmers can be trained easily to use its powerful facilities, DISCOVER allows data processing personnel to concentrate on more complex tasks.

Through both PRIME/POWER+ and DISCOVER, data is accessed online. Results are immediate and up-to-date, so management decision-making is easier.

Forms Management

FORMS, Prime's Forms Management System, is a set of screen management tools used to help develop applications for interactive, multiterminal and online processing. It permits forms to be designed for a variety of visual display terminals using the FORMS Description Language (FDL), or with FED, the FORMS screen painter. Application programs are created using standard READ/WRITE statements.

Prime INFORMATION

Prime INFORMATION™ is a user-friendly, relational-like, fourth generation, distributed data-management product. It offers a unique, highly productive environment for developing and using software solutions to application problems. Using flexible, variable length technology for all database structures (files, records, fields), it provides fully-integrated facilities for accomplishing many programming duties. These tasks include the ability to develop prototype solutions quickly and easily, design and define files and databases, write production programs, drive applications with menus, perform predefined or ad-hoc queries and reports, change definitions of data easily, enter and verify data easily, and completely customize an application and the user environment.

In addition, over 500 vertical and horizontal application packages run on Prime INFORMATION. There are also many third-party products that further enhance the already high productivity levels of Prime INFORMATION, including application generators complete with automated documentation.

PRIMEWAY

The PRIMEWAY™ Transaction Development and Management System is an integrated set of software products designed to automate and control the development, processing, and maintenance of business applications. Unlike application systems that have been developed using unrelated and often cumbersome software products, PRIMEWAY offers its users a complete, interactive working environment. As a result, application development, processing, and maintenance become more structured, efficient, and productive.

Office Automation

Prime's Office Automation System software combines Word Processing, Management Communication and Support, and Data Processing on one totally compatible system. These components work to improve information handling and productivity for the manager and professional, as well as for administrative personnel.

The Word Processing module includes functions such as text creation and editing, list processing and a boilerplate library. The Management Communications and Support module combines electronic mail; correspondence management, including filing and retrieval capabilities, and management support, providing for electronic scheduling.

The software merges complete office automation and communication functions with the full data processing capabilities of a Prime system.

Utilities

An excellent batch processing subsystem, which is completely compatible with the interactive environment, is supplied as standard software. Command and CPL files created for execution in the interactive environment may be run without modification as batch jobs.

Prime provides a set of disk-to-disk or disk-to-tape utilities as part of standard system software. For system backup, a special feature allows very fast disk-to-disk or disk-to-tape backup of complete disk volumes.

A very powerful print spooling package enables users to share both parallel and remote serial printers. With the versatile administrator and operator controls provided, individual print jobs may be automatically routed to any printer on the system, or to a printer on another network node. This package is also supplied as standard software.

Major Components

Memory Management

The PRIMOS operating system is optimized to make efficient use of the sophisticated virtual memory mechanism available on Prime 50 Series systems. The mechanism takes advantage of paging and segmentation techniques. Demand paging is used to achieve efficient memory use. Segmentation allows easy sharing and access control. These procedures are user-transparent.

When a program refers to a location in virtual memory that is not currently in physical memory, a "page fault" occurs and PRIMOS brings the referenced page (a 2Kb section of virtual memory) into physical memory. If a page has to be overwritten in order to do this, the least-recently used page is chosen. The hardware maintains referenced and modified page bits, which the software uses to reduce disk accesses caused by paging.

Segmentation provides variable-length segments of virtual memory up to 128K bytes. Code and data modules can be loaded into different segments, thus providing an easy method for sharing modules. Access to segments is controlled by per-user segment access rights. This makes it possible for different users to have different access rights to shared data.

Address translation is accelerated by use of a high-speed buffer called the Segment Translation Lookaside Buffer (STLB). The STLB holds recently-used, virtual-to-physical page translations. Prime processors include a cache memory with an access time as short as 40 nanoseconds. The cache is an integral part of the CPU and reduces memory access delays for data residing in the cache. Address translation is overlapped with cache access to further reduce total instruction execution time.

Security

The combination of hardware and software in the Prime 50 Series systems creates a secure multi-user environment.

Security is addressed at the memory, file system and user log-in levels. Segment Descriptor Words describe each segment of a user's virtual address space. Part of this description is per-ring access rights. Rings are levels of access privileges maintained by the hardware. Ring 0 is the most privileged, with full access rights and the ability to execute all instructions. Ring 3 is the least privileged, with no right to execute instructions that alter the system's mode of operation. The PRIMOS operating system enjoys Ring 0 privileges, timeshared users run in Ring 3. This ring privilege is used to validate all memory accesses. A hardware-supported gate mechanism allows Ring 3 users to temporarily gain Ring 0 privileges as they access certain operating system services. This protects data so that it can be accessed only via a controlled, gated call into a more privileged ring.

PRIMOS protects the system against unauthorized use. A user is identified with a 1- to 32-character user name and an optional password which is stored in encrypted form and verified by PRIMOS when the user logs in. Users are further registered as members of projects (an accounting designation) and groups (users with common access rights). If users supply an invalid username, password or project name, the system can record the unsuccessful attempt to log-in.

The system is designed to invoke an optional, installation-supplied log-in procedure which cannot be defeated. This feature allows system administrators to add additional site-dependent log-in processing, such as customized security checking or accounting.

File system security is provided by Access Control Lists (ACLs). ACLs are used to protect files and directories. An ACL is a list of access pairs. Each access pair specifies a user name or group name and the associated access rights. ACLs are a passive mechanism whereby a user's access to an object is determined solely by the access rights associated with the user or group name.

Only users with Protect rights to a directory can create ACLs for objects in that directory. Objects not explicitly protected by their own ACL are implicitly protected by the ACL protecting the directory in which they reside.

An alternative method of file system protection using directory passwords is also available for compatibility with older versions of the PRIMOS operating system.

Process Exchange/Scheduling

The PRIMOS operating system automatically transfers the attention of the central processor from one user or process to another, with little overhead and complete protection. The key is a central processor feature called Process Exchange, a firmware mechanism for context switching. It includes a hardware-implemented priority mechanism and makes use of dual user register sets. A context switch takes as little as nine microseconds. The combination of process exchange and demand paging means that directing the central processor to run a new user is a very low overhead operation. The scheduler takes full advantage of this and can maintain fast response for interactive users as machine load increases.

Command Line Processor

The PRIMOS command line processor improves the user's productivity by reducing the amount of typing necessary to accomplish common and repetitive functions.

The abbreviation facility allows users to create their own synonyms for commands and arguments. Using abbreviations, users can reduce common or lengthy command lines to simple, possibly parameterized, abbreviations. The net result is less typing, fewer errors, and increased productivity.

Users can use global variables for passing strings between command lines and programs. Global variables are created by the user and referenced by using their names in command lines or through subroutines in a program.

Command functions are available which return system data as strings for use in command lines. There are functions providing time and date in multiple formats, file system information, and arithmetic calculations. Additionally, users can write their own command functions.

The command processor also supports features which allow the application of single commands to multiple operands. Command iteration causes a single command to execute once for each of an explicit list of operands. Wildcarding allows a command to operate on a collection of file system objects selected via a mask. Treewalking executes a command over selected parts of a file system subtree. And name generation generates file system names from a given (possibly wildcard) name and pattern. In addition, qualifiers may be applied to selection criteria specifying, for example, file types or date selectors. Finally, all of these PRIMOS features – abbreviations, global variables, command functions, and command processing – may be used together to yield commands of increased functionality.

Command Procedure Language

The Command Procedure Language (CPL) is a powerful programming language available at command level. It is a simple, high-level language; PRIMOS commands, as its primitive statements, make it very powerful. CPL allows sequences of operating system commands and CPL directives to be built into command procedure files for execution. CPL directives provide for parameter passing and validation, error handling and control of statement execution order within the command file. CPL has many unique features which bring the power of command level programming to the user. Some of these are the position-independent argument passing, the interface to the PRIMOS operating system condition mechanism, and the comprehensive set of flow-of-control directives. The use of CPL will significantly enhance user productivity and make the system easier to use.

Condition Mechanism

The PRIMOS operating system provides full support for the ANSI PL/I condition mechanism. A condition is an unexpected event that occurs during program execution. Examples are arithmetic overflow, a "break" from the user's terminal, or a hardware-detected event such as access violation. The condition mechanism allows a specific software module called an "on-unit" to gain control when these events occur, regardless of the execution state within the currently-running program that users can define. When a condition is detected, PRIMOS "signals" that condition and searches the stack history of the running program backward in time for an on-unit for that condition. If a user-defined on-unit is found, it is invoked to perform a user-specified procedure. Users are thus able to trap system conditions and perform user specified actions. Users can also define, signal and trap their own conditions.

File Management System

The PRIMOS operating system file structure is a hierarchical tree structure with a Master File Directory (MFD) at the root of the structure. The file system creates and maintains an MFD for each disk or user-specified portion of a disk. The MFD contains the names and disk addresses of User File Directories (UFDs), segment directories and files. UFDs can be nested to 16 levels. Segment directories contain pointers to files which are addressed by position in the directory rather than by name. They permit rapid access to large collections of data that have an established order but variable size.

File access is through sequential (SAM) or direct (DAM) access methods. In SAM files, each record contains a pointer to the next record in the file for efficient sequential access. In DAM files, pointers to all data records in the file are stored in index records – thus reducing the search time required to retrieve any given data record. Both file types contain redundant pointers which can be used to repair the structure in the event of damage.

Security is ACL-based, meaning that user access rights to files and directories are a function of user identification. Also, file system usage quotas are available for limiting disk usage by directory.

All file I/O is buffered in memory. PRIMOS maintains the most recently-accessed disk records in memory, reducing the effective time for repeated accesses to the same disk record.

The PRIMOS operating system automatically assigns logical files to physical disk records. This feature permits a user to create file structures without concern for the characteristics of the disk on which they are stored.

Batch Processing

Although primarily an interactive operating system, PRIMOS provides a very capable batch-processing facility (BATCH). BATCH is designed for users who want the convenience of sequential job scheduling. With this facility, data processing or computational programs may be submitted for execution at a later time. Jobs submitted to the BATCH processing subsystem are comprised of standard PRIMOS operating system commands, so there is no need for the BATCH user to learn a complicated job control language. BATCH provides operator control features which allow the operator to control the number of job queues and the characteristics of each queue; thus, the operator can optimize the workload balance. BATCH is provided as part of Prime's standard software.

System Backup and Recovery

The Backup/Recovery Management Service (BRMS) is a set of utilities for full and incremental backup and archiving. BRMS is comprised of the BACKUP, BACKUP-RESTORE, ARCHIVE and TRANSPORT utilities.

The BACKUP and BACKUP-RESTORE utilities help ensure that the file system is restored to its state at the time of the last backup. To facilitate recovery of backed-up files, online catalogs record information about each file as it is saved.

In addition to backup and recovery, BRMS also has archiving capabilities. Archiving is available to all users for saving files which will not be needed online for an extended period, or for performing backup of personal files. As with BACKUP, the ARCHIVE utility catalogs file information as each file is saved.

The TRANSPORT command is provided to transfer files between systems. TRANSPORT neither updates catalogs nor saves attributes that may be specific to the system from which the file is saved.

U.S. Offices

Alabama
Birmingham

Alaska
Anchorage

Arizona
Phoenix
Tucson

California
Culver City
Irvine
Mountain View
Sacramento
San Diego
San Francisco
Walnut Creek
Woodland Hills

Colorado
Colorado Springs
Englewood

Connecticut
Windsor
Stamford

Florida
Hollywood
Jacksonville
Tampa
Winter Park

Georgia
Atlanta

Iowa
Iowa City

Illinois
Chicago
Oak Brook
Schaumburg

Indiana
Carmel

Kansas
Overland Park

Kentucky
Louisville

Louisiana
Metairie

Maryland
Rockville

Massachusetts
Framingham

Michigan
Flint
Grand Rapids
Troy

Minnesota
Bloomington

Missouri
Kansas City
St. Louis

Nebraska
Omaha

New Jersey
Parsippany

New Mexico
Albuquerque

New York
Albany
Amherst
Dewitt
Melville
New York
Rochester

North Carolina
Charlotte
Greensboro

Ohio
Cincinnati
Middleburg Heights
Worthington

Oklahoma
Tulsa

Oregon
Portland

Pennsylvania
Bridgeville
Camp Hill
Philadelphia
Wayne

South Carolina
Greenville

Tennessee
Knoxville
Nashville

Texas
Austin
Dallas
Houston

Utah
Salt Lake City

Virginia
Williamsburg

Washington
Bellevue

International Offices

Argentina
* *Buenos Aires*

Australia
Adelaide
Brisbane
Canberra
Hobart
Melbourne
Neutral Bay
* *North Sydney*
Perth

Austria
Vienna

Belgium
Zaventem

Bolivia
La Paz
Santa Cruz

Canada
Calgary
Edmonton
Halifax
London
Montreal
Ottawa
Saint John's
* *Toronto*
Vancouver
Winnipeg

Chile
Santiago

Colombia
* *Bogota*
Medellin

Denmark
Copenhagen

Ecuador
Quito

Finland
Helsinki

France
Aix
Angers
Grenoble
Lyon
* *Paris*
Segres

Greece
Athens

Hong Kong

India
Ahmadabad
Bangalore
* *Bombay*
Calcutta
Madras
New Delhi

Indonesia
* *Jakarta*

Ireland
Dublin

Israel
Tel Aviv

Italy
* *Milan*
Rome
Turin

Jamaica

Japan
Fukuoka
Nagoya
Osaka
* *Tokyo*

Jordan
* *Amman*

Korea
Pusan
* *Seoul*

Kuwait
Hawalli

Malaysia
* *Selangor*

Malta
* *Msida City*

Mexico
Guadalajara
Mexico City

Netherlands
Zoetermeer

New Zealand
* *Auckland*
Christchurch
Wellington
Parnelu

Nigeria
Lagos

Norway
Sandvika

Peru
Lima

Puerto Rico
San Juan

Qatar
* *Doha*

Saudi Arabia
Al Khobar

Singapore

South Africa
Capetown
Durban
* *Johannesburg*
Pretoria

Spain
Madrid

Sweden
Stockholm

Switzerland
Bern
Geneva
* *Zurich*

Taiwan
Taipei

Thailand
Bangkok

Turkey
Istanbul

United Kingdom
Bedford
Birmingham
Bristol
Central Park
City of London
Edinburgh
Feltham
Grange
* *Hounslow*
Leeds
Milton Keynes
Southampton
Stevenage
Syvenham
Warrington
Wilmslow

United Arab Emirates
* *Dubai*

Uruguay
* *Montevideo*

Venezuela
Caracas

West Germany
Dortmund
Düsseldorf
Hamburg
Hannover
München
Stuttgart
* *Wiesbaden*

* Main Office
(1/84)

PRIME®

Prime Computer, Inc.
Prime Park
Natick, Massachusetts 01760