A security method for an attached computer module in a computer system. The security method reads a security identification number in an attached computer module and compares it to a security identification number in a console, which houses the attached computer module. Based upon a relationship between these numbers, a security status is selected. The security status determines the security level of operating the computer system.

53 Claims, 14 Drawing Sheets


Board Specifications for DataModule Board, DataStation Backplane, DataStation Front Bezel, and BaseStation Backplane, Rev. E, Anigma, Inc., Mar. 2, 2000, 5 pages.


OTHER PUBLICATIONS


RocketLogix Presentation, RocketLogix, Inc., Apr. 6, 2000, 84 pages.


Email from Emanuel to Juarez including Invalidity Chart for Base Reference IBM BladeCenter, submitted by the Defendants in connection with Case No. 6:09-cv-148-LED, dated Aug. 20, 2010, 54 pages.

Letter from Martiniky to Byers regarding Proposed Amendment to the Invalidity Contentions based on products from Rlx Systems and a description of RLX products, dated Jul. 15, 2010, 2 pages.

“Plaintiff Acqis LLC’s Answer to Defendant Dell Inc.’s First Amended Counterclaims,” Case No. 6:09-cv-00148-LED, filed May 10, 2010, 16 pages.

“Affidavit of Timothy P. Cremen in Support of NEC Corporation of America’s Opposition to Plaintiff’s Motion to Dismiss and Strike,” Case No. 6:09-cv-00148-LED, filed Oct. 9, 2009, 4 pages.


“Defendant NEC Corporation of America’s Opposition to Plaintiffs Motion to Dismiss and Motion to Strike,” Case No. 6:09-cv-00148-LED, filed Oct. 9, 2010, 36 pages.

“Motion to Dismiss Counterclaim Pursuant to Rule 12(b)(6) and Motion to Strike Affirmative Defense Pursuant to Rule 12(f),” Case No. 6:09-cv-00148-LED, filed Sep. 16, 2009, 18 pages.


“Plaintiff Acqis LLC’s Objections and Responses to Defendant’s First Set of Interrogatories,” Case No. 6:09-cv-00148-LED, filed Aug. 5, 2010, 21 pages.


“Plaintiff Acqis LLC’s First Supplemental Objections and Responses to Defendant’s Third Set of Interrogatories,” Case No. 6:09-cv-00148-LED, filed Aug. 5, 2010, 14 pages.


Office Action for RE 41,294, filed Apr. 15, 2008, 10 pages.
Amendment Submitted with RCE for RE 41,294, filed May 27, 2008, 14 pages.
Supplemental Amendment Submitted with RCE for RE 41,294, filed Jul. 8, 2008, 12 pages.
Examiner Interview Summary Record, mailed Jan. 15, 2009, 1 page.
Amendment Submitted with RCE for RE 41,076, filed Nov. 5, 2007, 19 pages.
Amendment Submitted with RCE for RE 41,076, filed Dec. 8, 2008, 44 pages.
Amendment Submitted with RCE for RE 41,076, filed Aug. 4, 2009, 2 pages.
Examiner Interview Summary Record, filed Feb. 24, 2009, 4 pages.
Applicant Summary of Interview, filed Mar. 24, 2009, 1 page.
Examiner Interview Summary Record, filed Sep. 22, 2009, 5 pages.
Examiner’s Amendment for RE 41,092, filed Nov. 4, 2009, 4 pages.
Request for Ex Parte Reexamination of U.S. Patent No. 6,216,185, Control No. 90/010,816, filed Jan. 8, 2010, Exhibit BB Claim Charts with respect to Kobayashi for Anticipation and Obviousness, 128 pages.
Request for Ex Parte Reexamination of U.S. Patent No. 6,216,185, Control No. 90/010,816, filed Jan. 8, 2010, Exhibit CC Claim Charts with respect to Byte for Anticipation and Obviousness, 122 pages.
Request for Ex Parte Reexamination of U.S. Patent No. 6,216,185, Control No. 90/010,816, filed Jan. 8, 2010, Exhibit DD Claim Charts with respect to the Hitachi ‘675 for Anticipation and Obviousness, 119 pages.
Request for Inter Partes Reexamination of U.S. Patent No. 6,718,415, Control No. 95/001,276, filed Dec. 2, 2009, 44 pages.
Request for Inter Partes Reexamination of U.S. Patent No. 6,718,415, Control No. 95/001,276, Replacement Exhibit AA, Claim Charts with respect to QuantumNet for Anticipation and Obviousness, filed Jan. 5, 2010, 91 pages.
Request for Inter Partes Reexamination of U.S. Patent No. 6,718,415, Control No. 95/001,276, Replacement Exhibit CC, Claim Charts with respect to QuantumNet for Anticipation and Obviousness, filed Jan. 5, 2010, 85 pages.


* cited by examiner
FIG. 2
FIG. 3
FIG. 4

<table>
<thead>
<tr>
<th>Security ID</th>
<th>Access Privilege Granted</th>
</tr>
</thead>
<tbody>
<tr>
<td>JST</td>
<td>General Company</td>
</tr>
<tr>
<td>Sunshine</td>
<td>Restricted</td>
</tr>
<tr>
<td>Vacation</td>
<td>General Company</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Owner:</th>
<th>John Doe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Owner Password:</td>
<td>Sunshine</td>
</tr>
<tr>
<td>Business:</td>
<td>JST</td>
</tr>
<tr>
<td>Business Password:</td>
<td>Ether</td>
</tr>
<tr>
<td>Console Type:</td>
<td>Desktop</td>
</tr>
<tr>
<td>Location:</td>
<td>SFB3</td>
</tr>
<tr>
<td>Access privilege:</td>
<td>Private</td>
</tr>
<tr>
<td>Owner Access privilege:</td>
<td>Private</td>
</tr>
<tr>
<td>Others Access privilege:</td>
<td>Company</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Owner:</th>
<th>Mary Li</th>
</tr>
</thead>
<tbody>
<tr>
<td>Owner Password:</td>
<td>Vacation</td>
</tr>
<tr>
<td>Business:</td>
<td>JST</td>
</tr>
<tr>
<td>Business Password:</td>
<td>Ether</td>
</tr>
<tr>
<td>Console Type:</td>
<td>Desktop</td>
</tr>
<tr>
<td>Location:</td>
<td>LAB1</td>
</tr>
<tr>
<td>Access privilege:</td>
<td>Company</td>
</tr>
<tr>
<td>Owner Access privilege:</td>
<td>Private</td>
</tr>
<tr>
<td>Others Access privilege:</td>
<td>Company</td>
</tr>
</tbody>
</table>
Power up

Security code

Match ID?

Access Privilege

No Access

Block all data out

Access Privilege

Full access

Peripheral storage Read-only

Program data control

Hardware takes over

Data out to storage device?

Peripheral Read-only

Program data control

Hardware takes over

Block all data out

FIG. 7
FIG. 11
Figure 13
DATA SECURITY METHOD AND DEVICE FOR COMPUTER MODULES

Matter enclosed in heavy brackets [ ] appears in the original patent but forms no part of this reissue specification; matter printed in italics indicates the additions made by reissue.

Notice: More than one reissue application has been filed for the reissue of U.S. Pat. No. 6,643,777. The reissue applications are U.S. application Ser. Nos. 11/056,604 (a parent reissue application and now U.S. Pat. No. Re. 41,092), 11/545,056 (the present application, which is a continuation reissue of the parent reissue application), and 12/561,138 (which is a continuation reissue of the parent reissue application).

This application is a continuation reissue of U.S. application Ser. No. 11/056,604, which is a reissue of U.S. Pat. No. 6,643,777, which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

The present invention relates to computing devices. More particularly, the present invention provides a method and device for securing a personal computer or set-top box. Merely by way of example, the present invention is applied to a modular computing environment for desk top computers, but it will be recognized that the invention has a much wider range of applicability. It can be applied to other portable or modular computing applications.

Many desktop or personal computers, which are commonly termed PCs, have been around and used for over ten years. The PCs often come with state-of-art microprocessors such as the Intel Pentium™ microprocessor chips. They also include a hard or fixed disk drive including memory in the giga-byte range. Additionally, the PCs often include a random access memory integrated circuit device such as a dynamic random access memory device, which is commonly termed DRAM. The DRAM devices now provide up to millions of memory cells (i.e., mega-bit) on a single slice of silicon. PCs also include a high resolution display such as cathode ray tubes or CRTs. In most cases, the CRTs are at least 15 inches or 17 inches or 19 inches in diameter. High resolution flat panel displays are also used with PCs.

Many external or peripheral devices can be used with the PCs. Among others, these peripheral devices include mass storage devices such as a Zip™ Drive product sold by Iomega Corporation of Utah. Other storage devices include external hard drives, tape drives, and others. Additional devices include communication devices such as a modem, which can be used to link the PC to a wide area network of computers such as the Internet. Furthermore, the PC can include output devices such as a printer and other output means. Moreover, the PC can include special audio output devices such as speakers the like.

PCs also have easy to use keyboards, mouse input devices, and the like. The keyboard is generally configured similar to a typewriter format. The keyboard also has the length and width for easily inputting information by way of keys to the computer. The mouse also has a sufficient size and shape to easily move a cursor on the display from one location to another location.

Other types of computing devices include portable computing devices such as “laptop” computers and the like. Although somewhat successful, laptop computers have many limitations. These computing devices have expensive display technology. In fact, these devices often have a smaller flat panel display that has poor viewing characteristics. Additionally, these devices also have poor input devices such as smaller keyboards and the like. Furthermore, these devices have limited computing platforms to transfer information to and from these devices and other devices such as PCs.

Up to now, there has been little common ground between these platforms including the PCs and laptops in terms of upgrading, ease-of-use, cost, performance, and the like. Many differences between these platforms, probably somewhat intentional, has benefited computer manufacturers at the cost of consumers. A drawback to having two separate computers is that the user must often purchase both the desktop and laptop to have “total” computing power, where the desktop serves as a “regular” computer and the laptop serves as a “portable” computer. Purchasing both computers is often costly and runs “thousands” of dollars. The user also wastes a significant amount of time transferring software and data between the two types of computers. For example, the user must often couple the portable computer to a local area network (i.e., LAN), to a serial port with a modem and then manually transfer over files and data between the desktop and the portable computer. Alternatively, the user often must use floppy disks to “zip” up files and programs that exceed the storage capacity of conventional floppy disks, and transfer the floppy disk data manually.

Another drawback with the current model of separate portable and desktop computer is that the user has to spend money to buy components and peripherals the are duplicated in at least one of these computers. For example, both the desktop and portable computers typically include hard disk drives, floppy drives, CD-ROMs, computer memory, host processors, graphics accelerators, and the like. Because program software and supporting programs generally must be installed upon both hard drives in order for the user to operate programs on the road and in the office, hard disk space is often wasted.

One approach to reduce some of these drawbacks has been the use of a docking station with a portable computer. Here, the user has the portable computer for “on the road” use and a docking station that houses the portable computer for office use. The docking station typically includes a separate monitor, keyboard, mouse, and the like and is generally incompatible with other desktop PCs. The docking station is also generally not compatible with portable computers of other vendors. Another drawback to this approach is that the portable computer typically has lower performance and functionality than a conventional desktop PC. For example, the processor of the portable is typically much slower than processors in dedicated desktop computers, because of power consumption and heat dissipation concerns. As an example, it is noted that at the time of drafting of the present application, some top-of-the-line desktops include 400 MHz processors, whereas top-of-the-line notebook computers include 266 MHz processors.

Another drawback to the docking station approach is that the typical cost of portable computers with docking stations can approach the cost of having a separate portable computer and a separate desktop computer. Further, as noted above, because different vendors of portable computers have proprietary docking stations, computer users are held captive by their investments and must rely upon the particular computer vendor for future upgrades, support, and the like.

To date, most personal computers provide data file security through software only. A wide variety of removable storage media are available for a personal computer. These removable
media do not provide any access security protection in hardware. Data encryption program often must be used for protection. Such program is cumbersome to handle for the user requiring extra cost and time. Data encryption is more commonly used for communication over an unprotected network or the Internet. Having a large number of frequently used files managed by encryption software is not practical. Without software security program, any file can be read and copied illegally from a hard disk drive on a PC or any removable media.

PC architecture generally allows freedom of data flow between memory and peripheral devices within the allowed memory and I/O address spaces. In conventional PC architecture, a peripheral bus, i.e., PCI bus, is used to control all data transactions among peripheral devices. PCI bus allows any device to be a bus master and perform data transaction with another device. Also when a software program is in control, it can move data between any two devices. There is no hardware or protocol security mechanism on a standard peripheral bus such as PCI Bus to detect or block data transactions. Operating system may have individual files read or write protected. These types of special security feature require significant additional user interaction to control. This is too cumbersome for a typical user to manage. There is no mechanism in current PCs to allow access to the primary hard disk drive and yet prevent copying of its content. The conventional PC is a single machine that does not have a mechanism to perform security ID matching in hardware.

Thus, what is needed are computer systems that provide improved security features to prevent illegal or unauthorized access to information.

SUMMARY OF THE INVENTION

According to the present invention, a technique including a method and device for securing a computer module in a computer system is provided. In an exemplary embodiment, the present invention provides a security system for an attached computer module ("ACM"). In an embodiment, the ACM inserts into a computer module bay (CMB) within a peripheral console to form a functional computer. A security program reads an identification number in a security memory device to determine a security level of the ACM according to one embodiment.

In a specific embodiment, the present invention provides a system for secured information transactions. The system has a console (e.g., computer housing) comprising a peripheral controller housed in the console; and a security memory device (e.g., flash memory device) coupled to the peripheral controller. The system also has an attached computer module (i.e., a removable module with memory and microprocessor) coupled to the console. The attached computer module has a host interface controller housed within the attached computer module to interface to the security memory device through the peripheral controller.

In an alternative embodiment, the present invention provides a security protection method for a computer module. The method includes steps or acts of inserting the computer module into a console. Once the module has been inserted, the method initiates a security program in the module to read a security identification of the console and to read a security identification of the computer module. Based upon a relationship of the console identification and the computer module identification, a predetermined security status is determined from, for example, a look up table or the like. The method then selects the predetermined security status, which can be one of many. The method then operates the computer module based upon the security status. In a further alternative embodiment, the present invention provides a method for identifying a user for a computer module. The method includes inserting a computer module into a console; and initiating a security program in memory of the computer module. The method prompts a plurality of input fields corresponding to respective input information on a user interface to be provided by a user of the computer module. Next, the method inputs the input information into the user interface of the computer module. The input information includes a user (e.g., owner) name, a user (e.g., owner) password, a business name, a business password, and a location.

Still further, the present invention provides a system for secured information transactions, e.g., data security, electronic commerce, private communications. The system includes a console comprising a peripheral controller housed in the console. A user identification input device (e.g., keyboard, retinal reader, fingerprint reader, voice recognition unit) is coupled to the peripheral controller. The user identification input device is provided for user identification data of the user. The system has an attached computer module coupled to the console. The attached computer module has a security memory device (e.g., flash memory device) stored with the user identification data.

Numerous benefits are achieved using the present invention over previously existing techniques. The present invention provides mechanical and electrical security systems to prevent theft or unauthorized use of the computer system in a specific embodiment. Additionally, the present invention substantially prevents accidental removal of the ACM from the console. In some embodiments, the present invention prevents illegal or unauthorized use during transit. The present invention is also implemented using conventional technologies that can be provided in the present computer system in an easy and efficient manner. Depending upon the embodiment, one or more of these benefits can be available. These and other advantages or benefits are described throughout the present specification and are described more particularly below.

These and other embodiments of the present invention, as well as its advantages and features, are described in more detail in conjunction with the text below and attached FIGS.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a simplified diagram of a computer system according to an embodiment of the present invention;
FIG. 2 is a simplified diagram of a computer module according to an embodiment of the present invention;
FIG. 3 is a simplified top-view diagram of a computer module according to an embodiment of the present invention;
FIG. 4 is a simplified illustration of security systems according to embodiments of the present invention;
FIG. 5 is a simplified diagram of a computer module in a console according to an embodiment of the present invention;
FIG. 6 is a simplified diagram of a security method for a module according to an embodiment of the present invention; and
FIG. 7 is a simplified diagram of a method according to an embodiment of the present invention.
FIG. 8 is a simplified diagram of a system according to an alternative embodiment of the present application.
FIG. 9 is a block diagram of one embodiment of a computer system using the interface of the present invention.
FIG. 10 is a detailed block diagram of one embodiment of the host interface controller of the present invention.
FIG. 1 is a simplified diagram of a computer system according to an embodiment of the present invention. This diagram is merely an illustration and should not limit the scope of the claims herein. One of ordinary skill in the art would recognize other variations, modifications, and alternatives. The computer system includes an attached computer module (i.e., ACM) 10, a desktop console 20, and other elements. The computer system is modular and has a variety of components that are removable. Some of these components (or modules) can be used in different computers, workstations, computerized television sets, and portable or laptop units.

In the present embodiment, ACM 10 includes computer components, as will be described below, including a central processing unit ("CPU"), IDE controller, hard disk drive, computer memory, and the like. The computer module bay (i.e., CMB) 40 is an opening or slot in the desktop console. The CMB houses the ACM and provides communication to and from the ACM. The CMB also provides mechanical protection and support to ACM 10. The CMB has a mechanical alignment mechanism for mating a portion of the ACM to the console. The CMB further has thermal heat dissipation sinks, electrical connection mechanisms, and the like. Some details of the ACM can be found in co-pending U.S. patent application Ser. Nos. 09/149,882 and 09/149,548 filed Sep. 8, 1998 commonly assigned, and hereby incorporated by reference for all purposes.

In a preferred embodiment, the present system has a security system, which includes a mechanical locking system, an electrical locking system, and others. The mechanical locking system includes at least a key 11. The key 11 mates with key hole 13 in a lock, which provides a mechanical latch 15 in a closed position. The mechanical latch, in the closed position, mates and interlocks the ACM to the computer module bay. The mechanical latch, which also has an open position, allows the ACM to be removed from the computer module bay. Further details of the mechanical locking system are shown in the FIG. 2 below.

FIG. 2 is a simplified diagram of a computer module 10 according to an embodiment of the present invention. This diagram is merely an illustration and should not limit the scope of the claims herein. One of ordinary skill in the art would recognize other variations, modifications, and alternatives. Some of the reference numerals are similar to the previous FIG. for easy reading. The computer module 10 includes key 11, which is insertable into keyhole 13 of the lock. The lock has at least two position, including a latched or closed position and an unlatched or open position. The latched position secures the ACM to the computer module bay. The unlatched or open position allows the ACM to be inserted into or removed from the computer bay module. As shown, the ACM also has a slot or opening 14, which allows the latch to move into and out of the ACM. The ACM also has openings 17 in the backside for an electrical and/or mechanical connection to the computer module bay, which is connected to the console.

FIG. 3 is a simplified top-view diagram of a computer module for computer system according to an embodiment of the present invention. This diagram is merely an illustration and should not limit the scope of the claims herein. One of ordinary skill in the art would recognize other variations, modifications, and alternatives. The layout diagram illustrates the top-view of the module 10, where the backside components (e.g., Host Interface Controller) are depicted in dashed lines. The layout diagram has a first portion, which includes a central processing unit ("CPU") module 400, and a second portion, which includes a hard drive module 420. A common printed circuit board 437 houses these modules and the like. Among other features, the ACM includes the central processing unit module 400 with a cache memory 405, which is coupled to a north bridge unit 421, and a host interface controller 401. The host interface controller includes a lock control 403. As shown, the CPU module is disposed on a first portion of the attached computer module, and couples to connectors 17. Here, the CPU module is spatially located near connector 17.

The CPU module can use a suitable microprocessing unit, microcontroller, digital signal processor, and the like. In a specific embodiment, the CPU module uses, for example, a 400 MHz Pentium II microprocessor module from Intel Corporation and like microprocessors from AMD Corporation, Cyrix Corporation (now National Semiconductor Corporation), and others. In other aspects, the microprocessor can be one such as the Compaq Computer Corporation Alpha Chip, Apple Computer Corporation PowerPC G3 processor, and the like. Further, higher speed processors are contemplated in other embodiments as technology increases in the future.

In the CPU module, host interface controller 401 is coupled to BIOS/flash memory 405. Additionally, the host interface controller is coupled to a clock control logic, a configuration signal, and a peripheral bus. The present invention has a host interface controller that has lock control 403 to provide security features to the present ACM. Furthermore, the present invention uses a flash memory that includes codes to provide password protection or other electronic security methods.

The second portion of the attached computer module has the hard drive module 420. Among other elements, the hard drive module includes north bridge 421, graphics accelerator 423, graphics memory 425, a power controller 427, an IDE controller 429, and other components. Adjacent to and in parallel alignment with the hard drive module is a personal computer interface ("PCI") bus 431, 432. A power regulator 435 is disposed near the PCI bus.

In a specific embodiment, north bridge unit 421 often couples to a computer memory, to the graphics accelerator 423, to the IDE controller, and to the host interface controller via the PCI bus. Graphics accelerator 423 typically couples to a graphics memory 425, and other elements. IDE controller 429 generally supports and provides timing signals necessary for the IDE bus. In the present embodiment, the IDE controller is embodied as a 643U2 PCI-to IDE chip from CMD Technology, for example. Other types of buses than IDE are contemplated, for example EIDE, SCSI, USB, and the like in alternative embodiments of the present invention.

The hard drive module or mass storage module 420 typically includes a computer operating system, application software program files, data files, and the like. In a specific embodiment, the computer operating system may be the Windows98 operating system from Microsoft Corporation of Redmond Washington. Other operating systems, such as WindowsNT,
MacsOS, Unix, and the like are also contemplated in alternative embodiments of the present invention. Further, some typical application software programs can include Office98 by Microsoft Corporation, Corel Perfect Suite by Corel, and others. Hard disk module 420 includes a hard disk drive. The hard disk drive, however, can also be replaced by removable hard disk drive, read/write CD ROMs, flash memory, floppy disk drives, and the like. A small form factor, for example 2.5"., is currently contemplated, however, other form factors, such as PC card, and the like are also contemplated. Mass storage unit 240 may also support other interfaces than IDE.

In a specific embodiment, the present invention provides a file and data protection security system and method for a removable computer module or ACM. ACM contains the primary hard disk drive (HDD) where the operating system, application programs, and data files reside. The security system is used to prevent illegal access and copying of any file residing on the HDD inside ACM. An ACM is a self-contained computing device that can be armed with security software and hardware to protect its owner's private files and data. ACM docks with a computer bay in a wide variety of peripheral consoles. The combined ACM and peripheral console function as a personal computer. A computer module interface bus connects ACM and peripheral device. In some embodiments, all ACM data passes through computer module interface (CMI) bus to reach any device in the peripheral console, i.e. floppy drive, removable media, secondary hard disk drive, modem, and others. CMI bus data transfer is controlled by a pair of interface controllers on either side of the bus. This partitioning of a personal computer offers a way of protecting against illegal access of data residing within ACM by guarding data transaction through the computer module interface bus.

In a specific embodiment, a secured ACM has the enclosure that includes the following components:

1. ACPU,
2. Main memory,
3. A primary Hard Disk Drive (HDD),
4. Operating System, application software, data files on primary HDD,
5. Interface circuitry and connectors to peripheral console,
6. Flash memory used for storing security code and ID,
7. Data detection and control circuitry to manage data flow to peripheral console,
8. Circuit board connecting the above components, and others.

A peripheral console includes some of the following elements:

1. Input means, e.g. keyboard and mouse,
2. Display means, e.g. CRT monitor or integrated LCD display,
3. Removable storage media subsystem, e.g. Floppy drive, CDROM drive,
4. Communication device, e.g. LAN or modem,
5. Computer Module Bay, interface device and connectors to ACM,
6. Flash memory with security ID,
7. Power supply or battery system, and other devices.

The Computer Module Bay (CMB) is an opening in a peripheral console that receives ACM. CMB provides mechanical protection and electrical connection to ACM. The Computer Module Interface bus is made up of 3 bus components: video bus, peripheral data bus, and power bus. Video Bus consists of video output of graphics devices, i.e. analog RGB and control signals for monitor, or digital video signals to drive flat panel displays. Power bus supplies the power for ACM. Peripheral data bus is a high speed, compressed, peripheral bridge bus managed by a Host Interface Controller in ACM and a peripheral Interface Controller in peripheral console. In some embodiments, all peripheral data transaction passes through the interface controllers.

The implementation of the secured ACM generally includes the following elements:

1. A programmable Flash memory controlled by the Peripheral Interface Controller containing the security ID for the peripheral console,
2. A programmable Flash memory controlled by the Host Interface Controller containing hardware specific security code and ID for the computer module,
3. A data detection and control circuitry within Host Interface Controller to detect and manage data going out of ACM, and
4. A low level hardware dependent security code to perform security ID matching, hardware programming to manage data flow,
5. A high-level security program to manage user interface, program security ID, program security level, and other functions with access control.

The hardware and software implementation allow more flexibility in the level of security protection offered to an ACM owner. Some examples of security levels are:

1. No access—Security IDs do not match according to owner’s requirement. The Host Interface Controller blocks all peripheral data traffic between ACM and peripheral console except for keyboard and mouse,
2. Peripheral Read-only—No files can be written to any peripheral devices. All peripheral devices in peripheral console are managed as Read-only devices. The primary hard disk drive in ACM can be accessed freely,
3. Limited access—Certain peripheral devices are allowed read/write access, i.e. modem, and other devices are Read-only, i.e. removable media devices,
4. Full access—No restriction, and others.

Upon power up, the low level security code is executed to compare security ID between the respective flash memory between ACM and peripheral console. Typical security ID can include:

1. User ID
2. User password
3. User Access privilege
4. Business ID
5. Business password
6. Equipment ID
7. Equipment access privilege, and any other security IDs.

The user through the security program can activate different levels of password protection, which can be stored in a look up table. The company through the security program can control different levels of access privilege of a user, a business group, or equipment. The security code then program the security level allowed by the access privilege determined by the security ID matching result. For example, if an unidentified peripheral console is detected upon power up by the low level security code, e.g. a home unit, the access privilege can be set to Peripheral Read-only. With Read-only access privilege for all peripheral devices in peripheral console, the data detection and control circuitry is programmed to monitor all data traffic going to the peripheral console. Any memory block transfer to peripheral console will be detected and blocked. Under this mode, a user can use the computer with free access to the primary HDD in ACM. Any files can be read from other storage media in the peripheral console. But no files from the primary HDD can be copied to another media. The data detection circuitry separately monitors peripheral bus operation type and memory address range being
accessed. A specific address range for memory accesses and for I/O accesses can be programmed for the data detection circuitry to flag a match. A data blocking circuitry is triggered by the detection circuitry when a match occurs, and blank out the data that is being sent to the peripheral console. For the security system to be effective, a temper resistant enclosure must be used to prevent removal of the hard disk drive and the flash memory inside ACM. Further details are shown throughout the present specification and more particularly below.

FIG. 4 is a simplified illustration of security systems 300 according to embodiments of the present invention. This illustration is merely an example, which should not limit the scope of the claims herein. One of ordinary skill in the art would recognize other variations, modifications, and alternatives. The systems show various examples of ways to implement the present invention. Here, a user relies upon certain consoles to access information. A company’s shared, portable console 325 can access general company information 303. Selected security identification information 315 is entered into the shared console to access the information via a network. The information generally includes owner, owner password, business, business password, console type, location, and access privilege information, which is displayed on a user display. The owner is generally the user name. Owner password is the user password. The business is the business unit name and business password is the business unit password. The console type can be portable for laptops, notebooks, and the like. Alternatively, the console type can be a desktop. The location generally specifies the desktop location or address for a networked system. Alternatively, the location can also be a home location. Access privilege can be categorized into many different levels. For example, the user can access general company information, but not information directed to other business units. The user can also be limited to access his/her private information, which is company related. Many other types of information can be restricted or accessed depending upon the embodiment.

Other types of access can be granted depending upon the consoles. For example, various consoles include, among others, a console at a user’s home, e.g., “John Doe’s,” a console in the user’s office 329, a console in a co-worker’s office 331, which the user can access. The access from John Doe’s home console uses security identification 317 and provides restricted access 305. The user’s use of the module 307 can be from a variety of consoles and is accessed using security identification 319. Here, access privilege is private, which allows the user to access private personal information or private company information that the user has created. The user’s access from his office relies upon security identification 321, which grants access to private information and general company information. The co-worker’s console can also be used with security identification 323, which allows the user to access general company information but not private information of John Doe, for example. Depending upon the console used by the user, the security system can provide partial or full access to information on servers via network as well as an attached computer module. Information can also be limited to read only for certain information sources such as a server, a hard drive, a floppy drive, and others.

In a specific embodiment, the present invention also provides a security feature for the ACM 307. Here, the user of the ACM can be granted access to information in the ACM if the correct security identification information 319 is provided to the combination of ACM and console. Once the correct information is provided, the user can access the information on the hard drive of the ACM, which can be for private use. Other levels of access and security can also be provided depending upon the application.

FIG. 5 is a simplified diagram 500 of a computer module in a console according to an embodiment of the present invention. This diagram is merely an illustration which should not limit the scope of the claims herein. One of ordinary skill in the art would recognize other variations, modifications, and alternatives. The block diagram 500 includes an attached computer module 501 and a peripheral console 503, as well as other elements as desired. These elements have a variety of features such as those noted above, as well as others. In the present diagram, different reference numerals are used to show the operation of the present system.

The block diagram 500 illustrates attached computer module 501. The module 501 has a central processing unit 502, which communicates to a north bridge 541, by way of a CPU bus 527. The north bridge couples to main memory 523 via memory bus 529. The main memory can be any suitable high speed memory device or devices such as dynamic random access memory (“DRAM”) integrated circuits and others. The DRAM includes at least 32 Meg, or 64 Meg, and greater of memory, but can also be less depending upon the application. Alternatively, the main memory can be coupled directly with the CPU in some embodiments. The north bridge also couples to a graphics subsystem 515 via bus 542. The graphics subsystem can include a graphics accelerator, graphics memory, and other devices. Graphics subsystem transmits a video signal to an interface connector, which couples to a display, for example.

The attached computer module also includes a primary hard disk drive 509 that serves as a main memory unit for programs and the like. The hard disk can be any suitable drive that has at least 2 GB and greater. As merely an example, the hard disk is a Marathon 2250 (2.25 GB, 2 1/2 inch drive) product made by Seagate Corporation of Scotts Valley, but can be others. The hard disk communicates to the north bridge by way of a hard disk drive controller and bus lines 502 and 531. The hard disk drive controller couples to the north bridge by way of the host PCI bus 531, which connects bus 537 to the north bridge. The hard disk includes computer codes that implement a security program according to the present invention. Details of the security program are provided below.

The attached computer module also has a flash memory device 505 with a BIOS. The flash memory device 505 also has codes for a user password that can be stored in the device. The flash memory device generally permits the storage of such password without a substantial use of power, even when disconnected. As merely an example, the flash memory device has at least 512 kilobits or greater of memory; or 1 megabit or greater of memory. The flash memory device can store a security identification number or the like. The flash memory device is generally non-volatile and can preserve information even when the power is turned off, for example. The flash memory device generally has at least 128 kilobits storage cells or more. The flash memory can be any product such as a W29C020 product made by a company called Winbond of Taiwan, but can also be others. The flash memory cell and user identification will be more fully described below in reference to the FIGS. A host interface controller 507 communicates to the north bridge via bus 535 and host PCI bus. The host interface controller also has a data control 511. Host interface controller 507 communicates to the console using bus 513, which couples to connection 515. Peripheral console 503 includes a variety of elements to interface to the module 501, display 551, and network 553. The console forms around south bridge 571, which couples to
bus 563, which couples to bus 561. Bus 561 is in communication with network card 555, which is a local area network for Ethernet, for example. South bridge also couples through control 569 to peripheral interface controller 567, which also communicates to bus 561. Peripheral interface controller also couples to host interface controller through connection 515 and bus 513. The peripheral controller has a primary removable drive 559 connected to south bridge through bus 575. South bridge also couples to secondary hard disk through bus 577.

In a specific embodiment, the peripheral console also has a serial EEPROM memory device 575, which is coupled to the peripheral interface controller. The memory device can store a security identification number or the like. The memory device is generally non-volatile and can preserve information even when the power is turned off, for example. The memory generally has at least 16 kilobits of storage cells or more. Preferably, the memory device is a 16 kilobit device or 64 megabit device or greater, depending upon the application. The memory cell can be any product such as a X24320 product made by a company called Xicor, but can also be others. The memory cell and user identification will be more fully described below in reference to the FIGS.

FIG. 6 is a simplified diagram of a security method 600 for a module according to an embodiment of the present invention. This diagram is merely an illustration which should not limit the scope of the claims herein. One of ordinary skill in the art would recognize other variations, modifications, and alternatives. The present method uses a combination of software 601 and hardware 603, which is in the computer module. A plurality of external devices can be accessed depending upon the embodiment. These external devices include a secondary hard drive 618, a removable drive 619, a network (e.g., LAN, modem) device 621, and others. A keyboard 623 is also shown, which can act locally.

The software 601 includes an operating system 609, application programs 607, and a data security and initialization program 605. Other programs can also exist. Additionally, some of these programs may not exist. Preferably, the data security and initialization program exists. This data security and initialization program is initiated once the attached computer module is inserted into the console. The program interface and oversees a variety of hardware features, which will be used to control access to the external devices, for example. Of course, the particular configuration of the software will depend upon the application.

Hardware features can be implemented using a primary hard disk 611 coupled to a CPU/cache combination, which includes a main memory. The main memory is often a volatile memory such as dynamic random access memory. Data from any one of the external devices can enter the CPU/cache combination. For example, the secondary hard disk memory and I/O address ranges is transferred 624 to the CPU/ cache combination. The removable drive memory and I/O address ranges can also transfer 625 to the CPU/cache combination. The LAN memory and I/O address range data can also transfer 626 to the CPU/cache combination. Keyboard data can also transfer 627 to the CPU/cache combination. To write data from the module into any one of these external elements, the data security program interfaces with the data detection and control circuit to determine of such data should be transferred to any one of the external elements. As noted, the external elements include, among others, secondary hard disk, and removable drive. Here, the data security program checks the security identification number with other numbers to determine the security access level. There are many other ways that the present invention can be implemented. These methods are described more fully below.

FIG. 7 is a simplified diagram 700 of a method according to an embodiment of the present invention. This diagram is merely an illustration which should not limit the scope of the claims herein. One of ordinary skill in the art would recognize other variations, modifications, and alternatives. The present method begins at power up, which is step 701. The present method reads a security code, which has been entered by a user, for example, in step 703. The security code can be a string of characters, including numbers and letters. The security code is preferably a mixture of numbers and letters, which are at least about 6 characters in length, but is not limited.

The present method reads (step 703) the security code, which has been entered. Next, the security code is compared with a stored code, which is in flash memory or the like (step 705). If the compared code matches with the stored code, the method resumes to step 708. Alternatively, the method goes to step 707 via branch 706 where no access is granted. When no access is granted, all data are blocked from the user that attempts to log onto the system. Alternatively, the method determines if a certain level of access is granted, step 708. Depending upon the embodiment, the present method can grant full access, step 710, via branch 716. The present method allows full access based upon information stored in the flash memory device. Alternatively, the method can allow the user to access a limited amount of information.

Here, the present method allows for at least one or more than two levels of access. In a specific embodiment, the present method allows for the user of the module to access peripheral storage (step 711). The access privilege is read-only. The user can read information on the peripheral storage including hard disks and the like. Once the user accesses the storage, the method data control, step 719, takes over, where the hardware prevents the user from accessing other information, step 721. In a specific embodiment, the method can allow information to be removed from the peripheral storage. If the method allows for data to be removed, step 723, the method goes through branch 731 to let data out, which can occur through the module. Alternatively, the method goes to block data (step 725) via branch 733. Depending upon the embodiment, the method returns to the decision block, step 723. Alternatively, the method traverses branch 714 to a peripheral read-only process, step 712. The read-only process programs data control, step 713. Next, the hardware takes over (step 715). The method blocks all data from being accessed by the user, step 717.

FIG. 8 is a simplified diagram of a system 800 according to an alternative embodiment of the present invention. This diagram is merely an example which should not limit the scope of the claims herein. One of ordinary skill in the art would recognize many other variations, modifications, and alternatives. The system 800 includes an attached computer module 801, which can be inserted into one of a plurality of console devices to create a "plug and play" operation. For example, the console device can be peripheral console 801 or peripheral console 805. Each peripheral console can have similar or different connection characteristics. Peripheral console 803 couples to a local area network using Ethernet 817. Peripheral console 810 couples to a DSL line 827 through a DSL modem 825. Other consoles can also be included to use other types of networks such as ADSL, Cable Modem, wireless, Token Ring, and the like.

As shown, the attached computer module has elements such as a memory region 807, which stores BIOS information, a security code, and a security identification number on a flash memory device or the like. The memory region
couples to a central processing region 809, which can include CPU, chipset, cache memory, graphics, and a hard disk drive, as well as other features. The central processing region couples to a host interface controller, which interfaces the attached computer module to one of the peripheral consoles. Any of the above information can also be included in the attached computer module.

Each peripheral console also has a variety of elements. These elements include a region 813, 821, which has a flash memory device with a security identification number, a password, access information, access privileges, internet service provider access information, as well as other features, which were previously noted. The peripheral console also has an interface controller 815, 823, which couples region 813, 821, respectively to a networking device 817, 825. The networking device can be an Ethernet card 817, which allows communication to the local area network 819. Alternatively, the networking device can be a DSL modem 825, which allows communication to a DSL (or ADSL) phone line. Other types of networking device can also be used, depending upon the application.

Each console provides a selected connection based upon a set of predefined factors. These factors include communication hardware information so that software in attached computer module can read and allow a connection to a network. Here, access information can be provided to the user. Information about connection information will also be included. This connection information includes telephone numbers, account numbers, passwords (local), or a company password. The console and module combination will take care of charges, etc. based upon time bases. Module will have credit card information, but will have security. In a specific embodiment, the module inserts into the console. The module then asks the console which hardware will be used. If the hardware is an Ethernet connect, the module configures connection information to access the Ethernet connection. Alternatively, if the hardware requires a DSL connection, the module configures connection information to access the DSL connection. Other configuration information such as company server information, password, can also be provided.

Embodiments in accordance with the present invention may interface two PCI or PCI-like buses using a non-PCI or non-PCI-like channel. In accordance with embodiments of the present invention, PCI control signals are encoded into control bits and the control bits, rather than the control signals that they represent, are transmitted on the interface channel. At the receiving end, the control bits representing control signals are decoded back into PCI control signals prior to being transmitted to the intended PCI bus.

The fact that control bits rather than control signals are transmitted on the interface channel allows using a smaller number of signal channels and a correspondingly small number of conductive lines in the interface channel than would otherwise be possible. This is because the control bits can be more easily multiplexed at one end of the interface channel and recovered at the other end than control signals. This relatively small number of signal channels used in the interface channel allows using low voltage differential signal (LVDS) channels for the interface. An LVDS channel is more cable-friendly, faster, consumes less power, and generates less noise than a PCI bus channel, which is used in the prior art to interface two PCI buses. Therefore, the present invention advantageously uses an LVDS channel for the hereto unused purpose of interfacing PCI or PCI-like buses. The relatively smaller number of signal channels in the interface also allows using connectors having smaller pins counts. An interface having a smaller number of signal channels has, therefore, a smaller number of conductive lines is less bulky and less expensive than one having a larger number of signal channels. Similarly, connectors having a smaller number of pins are also less expensive and less bulky than connectors having a larger number of pins.

In a preferred embodiment, the interface channel has a plurality of serial bit channels numbering fewer than the number of parallel bus lines in each of the PCI buses and operates at a clock speed higher than the clock speed at which any of the bus lines operate. More specifically, the interface channel includes two sets of unidirectional serial bit channels which transmit data in opposite directions such that one set of bit channels transmit serial bits from the host interface controller (HIC) to the peripheral interface controller (PIC) while the other set transmits serial bits from the PIC to the HIC. For each cycle of the PIC clock, each bit channel of the interface channel transmits a packet of serial bits.

FIG. 9 is a block diagram of one embodiment of a computer system 900 using the interface of the present invention. Computer system 900 includes an ACM 905 and a peripheral console 910, which are described in greater detail in the application of William Y. Y. Chu for "Personal Computer Peripheral Console With Attached Computer Module" filed concurrently with the present application on Sep. 8, 1998, now U.S. Pat. No. 6,216,185, and incorporated herein by reference. The ACM 905 and the peripheral console 910 are interfaced through an exchange interface system (XIS) bus 915. The XIS bus 915 includes power bus 916, video bus 917, and peripheral bus (XPBus) 918, which is also herein referred to as an interface channel. The power bus 916 transmits power between ACM 905 and peripheral console 910. In a preferred embodiment power bus 916 transmits power at voltage levels of 3.3 volts, 5 volts, and 12 volts. Video bus 917 transmits video signals between the ACM 905 and the peripheral console 910. In a preferred embodiment, the video bus 917 transmits analog Red Green Blue (RGB) video signals for color monitors, digital video signals (such as Video Electronics Standards Association (VESA) Plug and Display’s Transient Minimized Differential Signaling (TMDS) signals for flat panel displays), and television (TV) and super video (S-Video) signals. The XPBus 918 is coupled to host interface controller (HIC) 919 and to peripheral interface controller (PIC) 920, which is also sometimes referred to as a bay interface controller. In the embodiment shown in FIG. 9, HIC 919 is coupled to an integrated unit 921 that includes a CPU, a cache, and a north bridge.

FIG. 10 is a detailed block diagram of one embodiment of the host interface controller (HIC) of the present invention. As shown in FIG. 10, HIC 1000 comprises bus controller 1010, translator 1020, transmitter 1030, receiver 1040, a PLL 1050, an address/data multiplexer (A/D MUX) 1060, a read/write controller (RDWR Ctrl) 1070, a video serial to parallel converter 1080 and a CPU control & general purpose output/ output latch/driver (CPU CNTL & GPIO latch/driver) 1090.

HIC 1000 is connected to an optional flash memory BIOS configuration unit 1001. Flash memory unit 1001 stores basic input output system (BIOS) and PCI configuration information and supplies the BIOS and PCI configuration information to A/D MUX 1060 and RDWR Control 1070, which control the programming, read, and write of flash memory unit 1001.

Bus controller 1010 is coupled to host PCI buses, which is also referred to herein as the primary PCI bus, and manages PCI bus transactions on the host PCI bus. Bus controller 1010 includes a slave (target) unit 1011 and a master unit 1016. Both slave unit 1011 and master unit 1016 each include two first in first out (FIFO) buffers, which are preferably
asynchronous with respect to each other since the input and output of the two FIFOs in the master unit 1016 as well as the two FIFOs in the slave unit 1011 are clocked by different clocks, namely the PCl clock and the PCK. Additionally, slave unit 1011 includes encoder 1022 and decoder 1023, while master unit 1016 includes encoder 1027 and decoder 1028. The FIFOs 1012, 1013, 1017 and 1018 manage data transfers between the host PCI bus and the XPBus, which in the embodiment shown in FIG. 10 operate at 33 MHz and 66 MHz, respectively. PCI address/data (AD) from the host PCI bus is entered into FIFOs 1012 and 1017 before they are encoded by encoders 1022 and 1023. Encoders 1022 and 1023 format the PCI address/data bits to a form more suitable for parallel to serial conversion prior to transmission on the XPBus. Similarly, address and data information from the receivers is decoded by decoders 1023 and 1028 to a form manageable for transmission on the PCI bus. The decoded data and address information is passed through FIFOs 1013 and 1018 prior to being transferred to the host PCI bus. FIFOs 1012, 1013, 1017 and 1018, allow bus controller 1010 to handle posted and delayed PCI transactions and to provide deep buffering to store PCI transactions.

Bus controller 1010 also comprises slave read/write control (RD/WR Ctrl) 1014 and master read/write control (RD/WR Ctrl) 1015. RD/WR controls 1014 and 1015 are involved in the transfer of PCI control signals between bus controller 1010 and the host PCI bus.

Bus controller 1010 is coupled to translator 1020. Translator 1020 comprises encoders 1022 and 1027, decoders 1023 and 1028, control decoder & separate data path unit 1024 and control encoder & merge data path unit 1025. As discussed above encoders 1022 and 1027 are part of slave data unit 1011 and master data unit 1016, respectively, receive PCI address and data information from FIFOs 1012 and 1017, respectively, and encode the PCI address and data information into a form more suitable for parallel to serial conversion prior to transmission on the XPBus. Similarly, decoders 1023 and 1028 are part of slave data unit 1011 and master data unit 1016, respectively, and format address and data information from receiver 1040 into a form more suitable for transmission on the host PCI bus.

Control encoder & merge data path unit 1025 receives PCI control signals from the slave RD/WR control 1014 and master RD/WR control 1015. Additionally, control encoder & merge data path unit 1025 receives control signals from CPU CNTL & GPIO latch/driver 1090, which is coupled to the CPU and north bridge (not shown in FIG. 10). Control encoder & merge data path unit 1025 encodes PCI control signals as well as CPU control signals into control information. Control decoder & separate data path unit 1024 decodes these encoded control bits and transmits the merged control bits to the control data path on the data lines PD0 to PD3 and control line PCN of the XPBus. Examples of control signals include PCI control signals and CPU control signals. A specific example of a control signal is FRAME# used in PCI buses. A control bit, on the other hand is a data bit that represents a control signal. Control decoder & separate data path unit 1024 receives control bits from receiver 1040 which receives control bits on data lines PDR0 to PDR3 and control line PCNR of the XPBus. Control decoder & separate data path unit 1024 separates the control bits it receives from receiver 1040 into PCI control signals, CPU control signals and north bridge signals, and decodes the control bits into PCI control signals, CPU control signals, and north bridge signals all of which meet the relevant timing constraints.

Transmitter 1030 receives multiplexed parallel address/data (A/D) bits and control bits from translator 1020 on the AD[3:0] out and the CNTL out lines, respectively. Transmitter 1030 also receives a clock signal from PLL 1050. PLL 1050 takes a reference input clock and generates PCK that drives the XPBuses. PCK is asynchronous with the PCI clock signal and operates at 66 MHz, twice the speed of the PCI clock of 33 MHz. The higher speed is intended to accommodate at least some possible increases in the operating speed of future PCI buses. As a result of the higher speed, the XPBuses may be used to interface two PCI or PCI-like buses operating at 66 MHz rather than 33 MHz or having 64 rather than 32 multiplexed address/data lines.

The multiplexed parallel A/D bits and some control bits input to transmitter 1030 are serialized by parallel to serial converters 1032 of transmitter 1030 into 10 bit packets. These bit packets are then output to the XPBus. Other control bits are serialized by parallel to serial converter 1033 into 10 bit packets and sent out on control line PCN of the XPBus.

The XPBus lines, PD0 to PD3, PCN, PDR0 to PDR3 and PCNR, and the video data and clock lines, VPD and VPCK, are not limited to being LVDS lines, as they may be other forms of bit based lines. For example, in another embodiment, the XPBuses lines may be IEEE 1394 lines.

It is to be noted that although each of the lines PCK, PD0 to PD3, PCN, PCKR, PDR0 to PDR3, PCNR, VPCK and VPD is referred to as a line, in the singular rather than plural, each such line may contain more than one physical line. For example, in the embodiment shown in FIG. 11, each of lines PCK, PD0 to PD3 and PCN includes two physical lines between each driver and its corresponding receiver. The term line, when not directly preceded by the terms physical or conductive, is herein used interchangeable with a signal or bit channel which may consist of one or more physical lines for transmitting a signal. In the case of non-differential signal lines, generally only one physical line is used to transmit one signal. However, in the case of differential signal lines, a pair of physical lines is used to transmit one signal. For example, a bit line or bit channel in an LVDS or IEEE 1394 interface consists of a pair of physical lines which together transmit a signal.

A bit based line (i.e., a bit line) is a line for transmitting serial bits. Bit based lines typically transmit bit packets and use a serial data packet protocol. Examples of bit lines include an LVDS line, an IEEE 1394 line, and a Universal Serial Bus (USB) line.
above description of FIG. 6 for an understanding of the functions of the elements of PIC 1200 having corresponding elements in PIC 600.

As suggested above, there are also differences between PIC 600 and PIC 1200. Some of these differences between PIC 600 and PIC 1100 include the following. First, receiver 1240 in PIC 1200, unlike receiver 640 in PIC 600, does not contain a synchronization unit. As mentioned above, the synchronization unit in PIC 600 synchronizes the PCKR clock to the PCK clock locally generated by PLL 650. PIC 1100 does not locally generate a PCK clock and therefore, it does not have a locally generated PCK clock with which to synchronize the PCK clock signal that it receives from PIC 600. Another difference between PIC 1200 and PIC 600 is the fact that PIC 1200 contains a video parallel to serial converter 1289 whereas PIC 600 contains a video serial to parallel converter 680. Video parallel to serial converter 1289 receives 16-bit parallel video capture data and video control signals on the Video Port Data [0:15] and Video Port Control lines, respectively, from the video capture circuit (not shown in FIG. 12) and converts them to a serial video data stream that is transmitted on the VPD line to the VIC. The video capture circuit may be any type of video capture circuit that outputs a 16-bit parallel video capture data and video control signals.

Another difference lies in the fact that PIC 1200, unlike PIC 600, contains a clock doubler 1282 to double the video clock rate of the video clock signal that it receives. The doubled video clock rate is fed into video parallel to serial converter 1282 through buffer 1283 and is sent to serial to parallel converter 650 through buffer 1284. Additionally, reset control unit 1235 in PIC 1200 receives a reset signal from the CPU CNTL & GPIO latch/driver unit 1190 and transmits the reset signal on the VPD line to the VIC 600 whereas reset control unit 645 of PIC 600 receives the reset signal and forwards it to its CPU CNTL & GPIO latch/driver unit 690 because, in the above embodiment, the reset signal RESET# is unidirectionally sent from the PIC 1200 to the VIC 600.

The XPBUS which includes lines PCK, PDDO to PDD3, PCN, PCKR, PDR0 to PDR3, and PCNR, has two sets of unidirectional lines transmitting clock signals and bits in opposite directions. The first set of unidirectional lines includes PCK, PDDO to PDD3, and PCN. The second set of unidirectional lines includes PCKR, PDR0 to PDR3, and PCNR. Each of these unidirectional sets of lines is a point-to-point bus with a fixed transmitter and receiver, or in other words a fixed master and slave bus. For the first set of unidirectional lines, the VIC is a fixed transmitter/master whereas the PIC is a fixed receiver/slave. For the second set of unidirectional lines, the PIC is a fixed transmitter/master whereas the VIC is a fixed receiver/slave. The LVDS lines of XPBUS, a cable friendly and remote system I/O bus, transmit fixed length data packets within a clock cycle.

In the embodiment shown in FIG. 9, VIC 919 is coupled to an integrated unit 921 that includes a CPU, a cache and a north bridge. In another embodiment, such as that shown in FIG. 13, the CPU 1305 and north bridge 1310 are separate rather than integrated units. In yet another embodiment, such as that shown in FIG. 14, the VIC and PIC are integrated with the north and south bridges, respectively, such that integrated VIC and north bridge unit 1405 includes an VIC and a north bridge, while integrated PIC and south bridge unit 1410 includes a PIC and a south bridge.

Although the functionality above has been generally described in terms of a specific sequence of steps, other steps can also be used. Here, the steps can be implemented in a combination of hardware, firmware, and software. Either of these can be further combined or even separated. Depending upon the embodiment, the functionality can be implemented in a number of different ways without departing from the spirit and scope of the claims herein. One of ordinary skill in the art would recognize other variations, modifications, and alternatives.

While the above is a full description of the specific embodiments, various modifications, alternative constructions and equivalents may be used. Therefore, the above description and illustrations should not be taken as limiting the scope of the present invention which is defined by the appended claims.

What is claimed is:

1. A security protection method for a computer module, said method comprising:
   inserting the computer module into a console;
   initiating a security program in said module to read a security identification of said console and to read a security identification of said computer module;
   determining of a predetermined security status based upon a relationship of said console identification and said computer module identification;
   selecting said predetermined security status; and
   operating said computer module based upon said security status.

2. The method of claim 1 wherein said predetermined security status disables a network access to the computer module.

3. The method of claim 1 wherein said predetermined security status disables a secondary storage of information from said computer module to substantially prevent information to be transferred from a memory of the computer module to said secondary storage.

4. The method of claim 1 wherein said security program is provided in a system BIOS.

5. The method of claim 1 wherein said step of initiating reads said security identification of said computer module from a flash memory device.

6. The method of claim 1 wherein said step of initiating reads said security identification of said console from a flash memory device.

7. The method of claim 1 wherein said console is selected from a desktop home computing device, an office desktop computing device, a mobile computing device, a television set-top computing device, and a co-worker's computing device.

8. A system for secured information transactions, the system comprising:
   a console comprising a peripheral controller housed in the console;
   a user identification input device coupled to the peripheral controller; the user identification input device being provided for user identification data; and
   an attached computer module coupled to the console, the attached computer module comprising a security memory device stored with the user identification data.

9. The system of claim 8 wherein the user identification input device is a finger print reader.

10. The system of claim 8 wherein the user identification input device is a voice processing device.

11. A method for operating a module computer into one of a plurality of network systems, the method comprising:
   providing a computer module, the module comprising a connection program;
   inserting the computer module into a computer console, the computer console having access to a network;
   receiving connection information from the computer console;
configuring the connection program to adapt to the connection information; and establishing a connection between the computer module and a server coupled to the network.

12. The method of claim 11 wherein the connection information comprises a connection protocol for providing the connection.

13. The method of claim 12 wherein the connection protocol is selected from TCP/IP, or mobile IP.

14. A system for secured information transactions, the system comprising:
   a console comprising a network controller housed in the console;
   a user identification input device coupled to the network controller; and
   an attached computer module coupled to the console, the attached computer module comprising a central processing unit,
   a device stored with a user identification data,
   a security program providing password protection for access to the computer module based on the user identification data,
   an integrated interface controller and bridge unit to communicate an encoded serial bit stream of address and data bits of Peripheral Component Interconnect (PCI) bus transaction, the integrated interface controller and bridge unit directly coupled to the central processing unit, and
   a low voltage differential signal channel coupled to the integrated interface controller and bridge unit to convey the encoded serial bit stream of PCI bus transaction,

   wherein the low voltage differential signal channel comprises two sets of unidirectional serial bit channels which transmit data in opposite directions.

15. The system of claim 14 wherein the encoded serial bit stream comprises 10 bit packets.

16. The system of claim 14 wherein the integrated interface controller and bridge unit is coupled to the central processing unit without any intervening PCI bus.

17. The system of claim 16 wherein the integrated interface controller and bridge unit comprises a north bridge and an interface controller integrated with the north bridge, and the low voltage differential signal channel extends from the interface controller to convey the encoded serial bit stream of PCI bus transaction.

18. The system of claim 16 further comprising a peripheral component coupled to the central processing unit through the low voltage differential signal channel.

19. The system of claim 16 wherein the attached computer module further comprises a main memory coupled to the central processing unit through the integrated interface controller and bridge unit.

20. The system of claim 16 wherein the integrated interface controller and bridge unit is configured to output encoded PCI address and data bits in serial form that are conveyed over the low voltage differential signal channel.

21. The system of claim 16 wherein the encoded serial bit stream of PCI bus transaction comprises information to permit decoding to create a PCI bus transaction across the low voltage differential signal channel.

22. The system of claim 16 wherein the low voltage differential signal channel comprises a first plurality of unidirectional, differential signal pairs to convey data in a first direction and a second plurality of unidirectional, differential signal pairs to convey data in a second, opposite direction.

23. The system of claim 16 wherein the network controller comprises an Ethernet controller.

24. A method comprising:
   providing a computer module, the module comprising a central processing unit, a connection program, an integrated interface controller and bridge unit to output an encoded serial bit stream of address and data bits of Peripheral Component Interconnect (PCI) bus transaction, the integrated interface controller and bridge unit coupled to the central processing unit without any intervening PCI bus, and a low voltage differential signal channel coupled to the integrated interface controller and bridge unit to convey the encoded serial bit stream of PCI bus transaction;
   inserting the computer module into a computer console, the computer console having access to a network; receiving connection information from the computer console; configuring the connection program to adapt to the connection information; and establishing a connection between the computer module and a server coupled to the network;
   wherein the low voltage differential signal channel further comprises two sets of unidirectional serial bit channels which transmit data in opposite directions.

25. The method of claim 24 wherein the connection between the computer module and the server comprises an Ethernet connection.

26. The method of claim 24 wherein the connection protocol is TCP/IP.

27. The method of claim 24 further comprising conveying, over the low voltage differential signal channel, the encoded serial bit stream of PCI bus transaction as 10 bit packets.

28. The method of claim 27 wherein conveying the encoded serial bit stream of PCI bus transaction comprises conveying information to permit decoding to create a PCI bus transaction across the low voltage differential signal channel.

29. A system of connecting a computer module to a network, the system comprising:
   a console having access to a network;
   a computer module inserted into the console and powered by the console, the computer module comprising a central processing unit, a peripheral bridge coupled to the central processing unit without any intervening Peripheral Component Interconnect (PCI) bus, the peripheral bridge comprising an interface controller to communicate an encoded serial bit stream of address and data bits of PCI bus transaction, a low voltage differential signal channel extending from the interface controller to convey the encoded serial bit stream of PCI bus transaction, and a connection program receiving connection information from the console, configuring the connection program to adapt to the connection information, and establishing a connection between the computer module and a server coupled to the network;
   wherein the low voltage differential signal channel comprises two sets of unidirectional serial bit channels which transmit data in opposite directions.

30. The system of claim 29 wherein the encoded serial bit stream comprises 10 bit packets.

31. The system of claim 29 wherein the interface controller is integrated with the peripheral bridge as a single integrated unit.
32. The system of claim 31 wherein the peripheral bridge comprises a north bridge.

33. The system of claim 32 further comprising a south bridge coupled to the north bridge through the low voltage differential signal channel.

34. The system of claim 31 wherein the interface controller is coupled to the central processing unit without any intervening PCI bus.

35. The system of claim 31 wherein each of the unidirectional serial bit channels corresponds to a point-to-point link.

36. A system for secured information transactions, the system comprising:
   a console comprising a network communication controller housed in the console;
   a user identification input device coupled to the console, the user identification input device being provided for user identification data; and
   an attached computer module coupled to the console, the attached computer module comprising a central processing unit, a security program, a peripheral bridge directly coupled to the central processing unit, the peripheral bridge comprising an interface controller, the peripheral bridge and the interface controller configured as a single integrated unit, and a low voltage differential signal channel that comprises two sets of unidirectional serial bit channels in opposite directions which transmit data in 10 bit packets, the low voltage differential signal channel coupled to the interface controller; wherein said security program receives the user identification data from the console and determines a predetermined security status, and operating said computer module based upon said security status.

37. The system of claim 36 wherein the data packets comprise an encoded bit stream of address and data bits of Peripheral Component Interconnect (PCI) bus transaction.

38. The system of claim 37 wherein the interface controller is configured to transmit and receive encoded PCI address and data bits over the low voltage differential signal channel.

39. The system of claim 38 wherein the interface controller is coupled to the central processing unit without any intervening PCI bus.

40. The system of claim 38 wherein the peripheral bridge comprises a north bridge.

41. The system of claim 40 further comprising a south bridge coupled to the north bridge.

42. The system of claim 41 wherein the south bridge is coupled to the north bridge through the low voltage differential signal channel.

43. The system of claim 38 wherein the low voltage differential signal channel comprises two sets of unidirectional, multiple serial bit channels to convey data in opposite directions, and each of the unidirectional, multiple serial bit channels corresponds to a point-to-point link.

44. A system for secured information transactions, the system comprising:
   a computer console comprising a network communication controller housed in the console, a low voltage differential signal (LVDS) channel comprising two sets of multiple unidirectional serial channels that transmit encoded address and data bits of Peripheral Component Interconnect (PCI) bus transaction in opposite directions, and
   a user identification input device coupled to the console, the user identification input device being provided for user identification data; and
   an attached computer module that inserts into the console in a "plug and play" operation, the attached computer module comprising a security program providing password protection for data content within said attached computer module, a security memory device stored with the user identification data, and an interface controller coupled to the console through serial bit based lines; wherein the interface controller transfers data between the computer module and the console in Universal Serial Bus (USB) protocol.

45. The system of claim 44 wherein said attached computer module inserts into the console and is powered by the console to form a functional computer.

46. The system of claim 44 wherein the attached computer module has a tamper resistant enclosure.

47. The system of claim 44 wherein the security memory device comprises flash memory.

48. The system of claim 44 wherein the console further comprises an integrated interface controller and bridge unit to transmit and receive encoded serial bits of PCI bus transaction over the LVDS channel.

49. The system of claim 48 wherein the encoded serial bits of PCI bus transaction comprise information to permit decoding to create a PCI bus transaction.

50. A system for secured information transactions, the system comprising:
   a console comprising a network communication controller housed in the console;
   a user input device coupled to the console, the user input device being provided for user identification data; and
   an attached computer module inserted into the console, the attached computer module comprising a central processing unit, a peripheral bridge directly coupled to the central processing unit without any intervening Peripheral Component Interconnect (PCI) bus, the peripheral bridge comprising an integrated interface controller to communicate an encoded serial bit stream of address and data bits of PCI bus transaction, a low voltage differential signal channel extending from the integrated interface controller to convey the encoded serial bit stream of PCI bus transaction and a mass storage unit storing a security program and user identification data; wherein the security program receives the user identification data from the input device, matches the stored user identification data and permits external access to the computer module.

51. The system of claim 50 wherein the security program further determines a predetermined security status, and controls different levels of access privilege to said attached computer module.

52. The system of claim 50 wherein the encoded serial bit stream comprises 10 bit packets.

53. The system of claim 50 wherein the peripheral bridge comprises a north bridge.
54. The system of claim 50 wherein the integrated interface controller is coupled to the central processing unit without any intervening PCI bus, and the integrated interface controller is configured to output encoded address and data bits of PCI bus transaction in serial form that are conveyed over the low voltage differential signal channel.

55. The system of claim 50 wherein the low voltage differential signal channel comprises a first plurality of unidirectional, differential signal pairs to convey data in a first direction and a second plurality of unidirectional, differential signal pairs to convey data in a second, opposite direction.

56. A system comprising:

a console housing a network controller and a low voltage differential signal serial channel for communicating encoded address and data bits of Peripheral Component Interconnect (PCI) bus transaction;

a user identification input device coupled to the console; and

an attached computer module coupled to the console, the attached computer module comprising

a device stored with a user identification data,

a security program providing protection for access to the computer module based on the user identification data, and

an interface controller coupled to the console for communicating data in a serial bit stream, wherein the low voltage differential signal serial channel further comprises two sets of unidirectional serial bit channels which transmit data in opposite directions.

57. The system of claim 56 wherein the encoded address and data bits of PCI bus transaction comprise 10 bit data packets.

58. The system of claim 56 wherein the computer module further comprises a flash memory mass storage device.

59. The system of claim 56 wherein the serial bit stream transmits data packets in Universal Serial Bus (USB) protocol.

60. A system for secured information transactions, the system comprising:

a console comprising a peripheral controller housed in the console;

a user identification input device coupled to the peripheral controller, the user identification input device being provided for user identification data; and

an attached computer module coupled to the console, the attached computer module comprising

a security memory device stored with the user identification data,

a central processing unit,

an integrated interface controller and bridge unit to communicate an encoded serial stream of address and data bits of Peripheral Component Interconnect (PCI) bus transaction as 10 bit packets, the integrated interface controller and bridge unit directly coupled to the central processing unit, and

a low voltage differential signal channel coupled to the integrated interface controller and bridge unit to convey the encoded serial stream of PCI bus transaction, the low voltage differential signal channel comprising two sets of unidirectional serial bit channels which transmit data in opposite directions.

61. The system of claim 60 wherein the integrated interface controller and bridge unit is coupled to the central processing unit without any intervening PCI bus.

62. The system of claim 61 wherein the integrated interface controller and bridge unit comprises a north bridge and an interface controller integrated with the north bridge, and the low voltage differential signal channel extends from the interface controller to convey the encoded serial stream of PCI bus transaction.

63. The system of claim 61 wherein the attached computer module further comprises a main memory coupled to the central processing unit through the integrated interface controller and bridge unit.

64. The system of claim 61 wherein the integrated interface controller and bridge unit is configured to output encoded PCI address and data bits in serial form that are conveyed over the low voltage differential signal channel.

65. The system of claim 61 wherein the encoded serial stream of PCI bus transaction comprises information to permit decoding to create a PCI bus transaction across the low voltage differential signal channel.

66. The system of claim 61 wherein the low voltage differential signal channel comprises a first plurality of unidirectional, differential signal pairs to convey data in a first direction and a second plurality of unidirectional, differential signal pairs to convey data in a second, opposite direction.

* * * * *