



US 20070123353A1

(19) **United States**

(12) **Patent Application Publication**  
**Smith**

(10) **Pub. No.: US 2007/0123353 A1**

(43) **Pub. Date: May 31, 2007**

(54) **INVITATION SYSTEM FOR ON-LINE VIDEO GAMES**

**Publication Classification**

(75) Inventor: **Darren Smith**, Bellevue, WA (US)

(51) **Int. Cl.**  
*A63F 9/24* (2006.01)

(52) **U.S. Cl.** ..... 463/42

Correspondence Address:  
**NIXON & VANDERHYE, P.C.**  
**901 NORTH GLEBE ROAD, 11TH FLOOR**  
**ARLINGTON, VA 22203 (US)**

(57) **ABSTRACT**

Example methods and systems for inviting users to play video games involve receiving at an invitation server an invitation from a first user logged into the invitation server using a first game system that invites a second user logged into the invitation server using a second game system to play a game. The invitation is forwarded from the invitation server to the second user and an acceptance of the invitation from the second user is received at the invitation server. Invitation acceptance data indicative of the second user's acceptance is stored at the invitation server after the second user logs out of the invitation server. When the second user subsequently logs back into the invitation server, one or both of the first and second users is sent information based on the stored invitation acceptance that enables the first and second users to play the game.

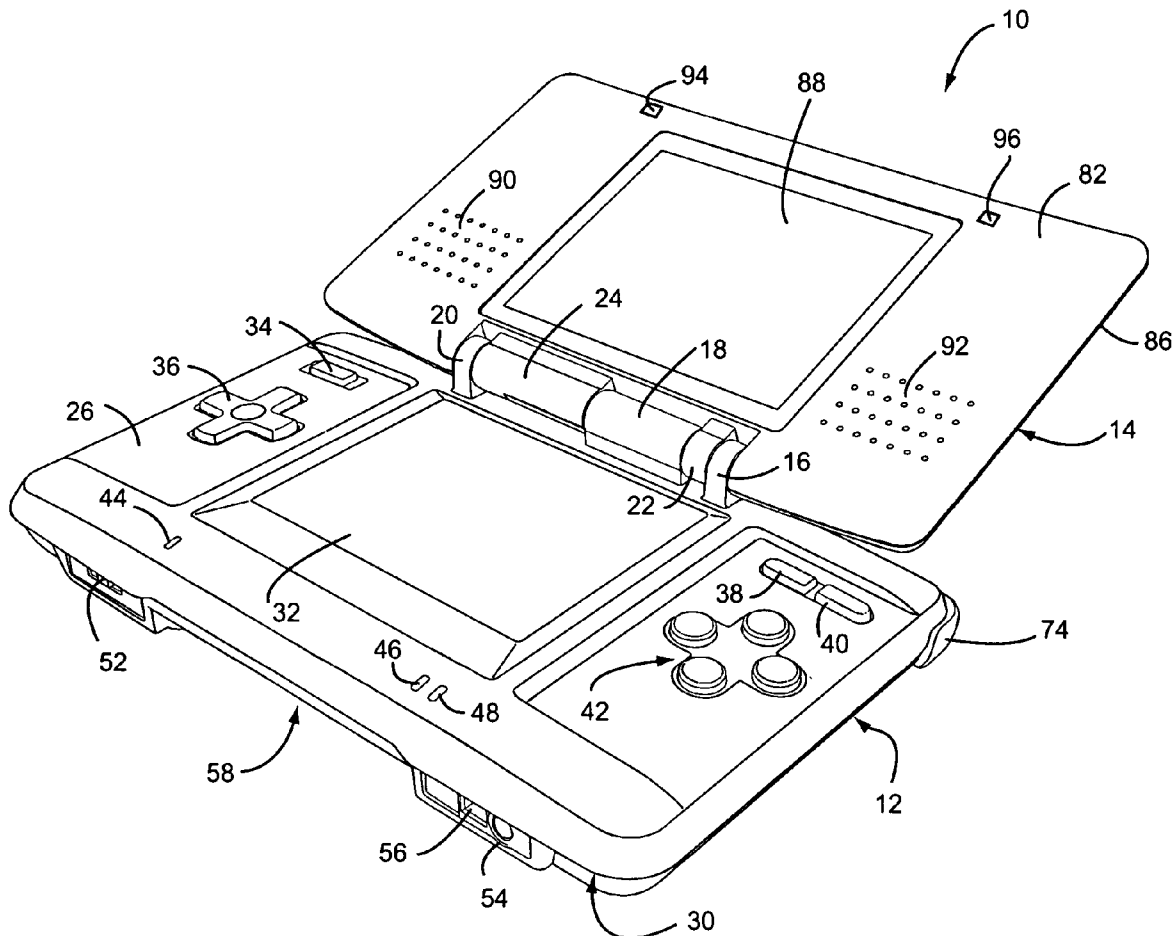
(73) Assignees: **Nintendo Co., Ltd.**, Kyoto (JP); **Nintendo of America Inc.**, Redmond, WA

(21) Appl. No.: **11/507,021**

(22) Filed: **Aug. 21, 2006**

**Related U.S. Application Data**

(60) Provisional application No. 60/709,426, filed on Aug. 19, 2005.





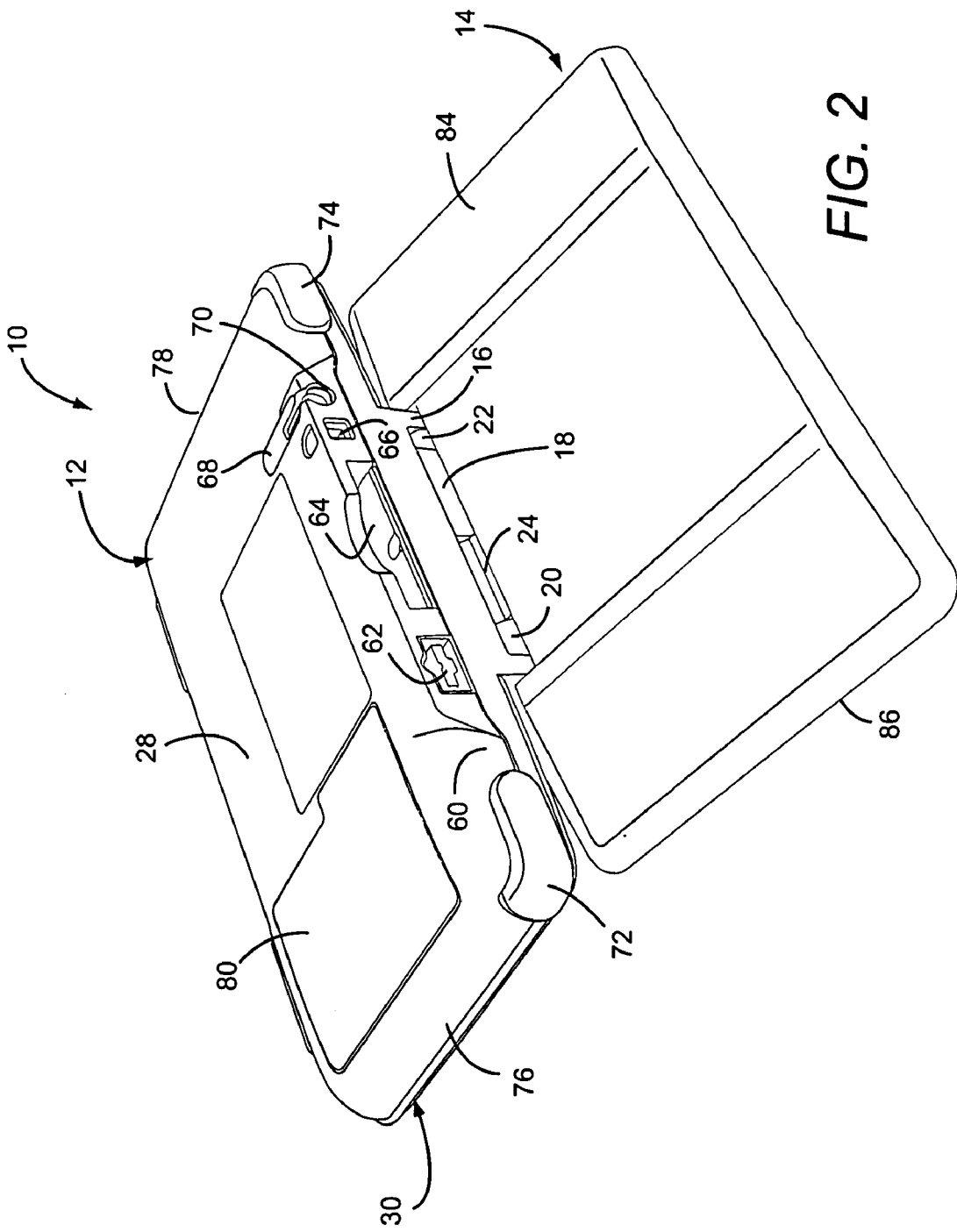


FIG. 2

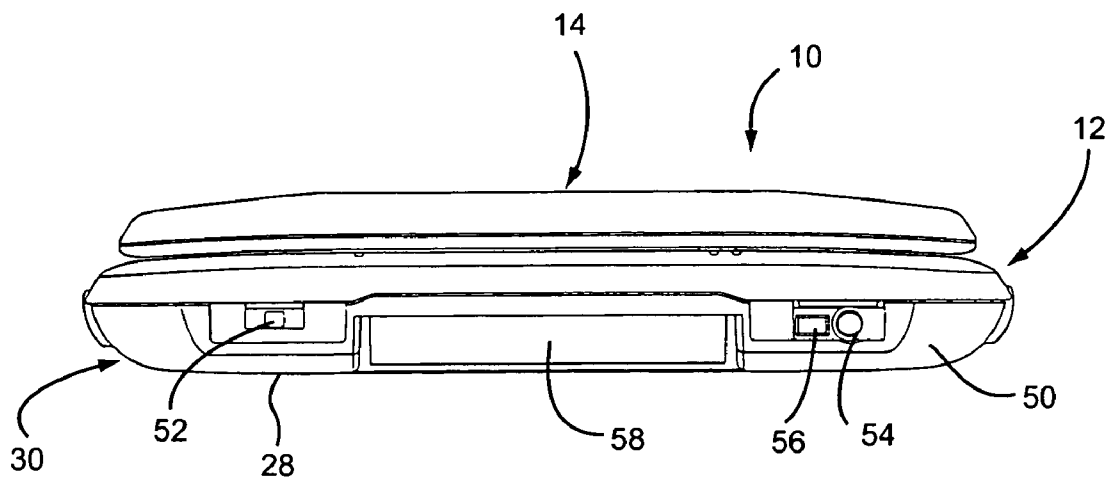


FIG. 3

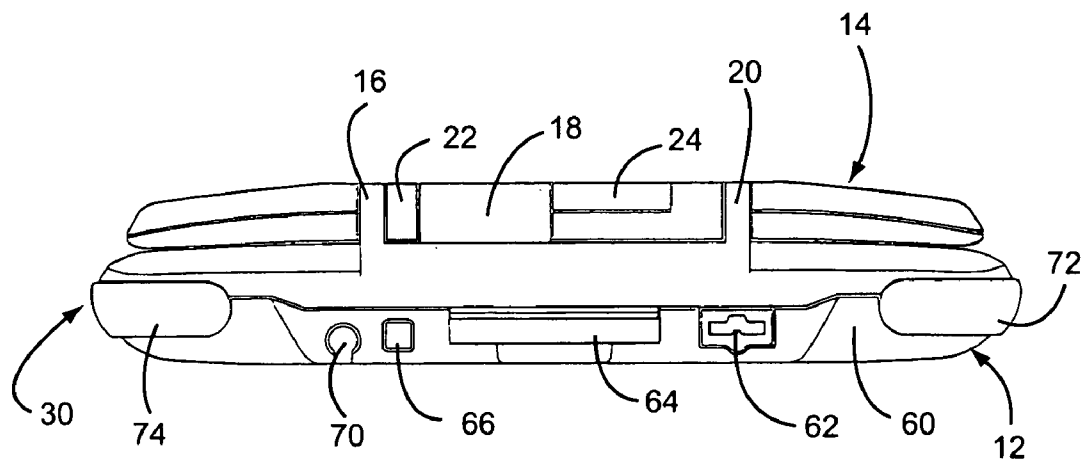
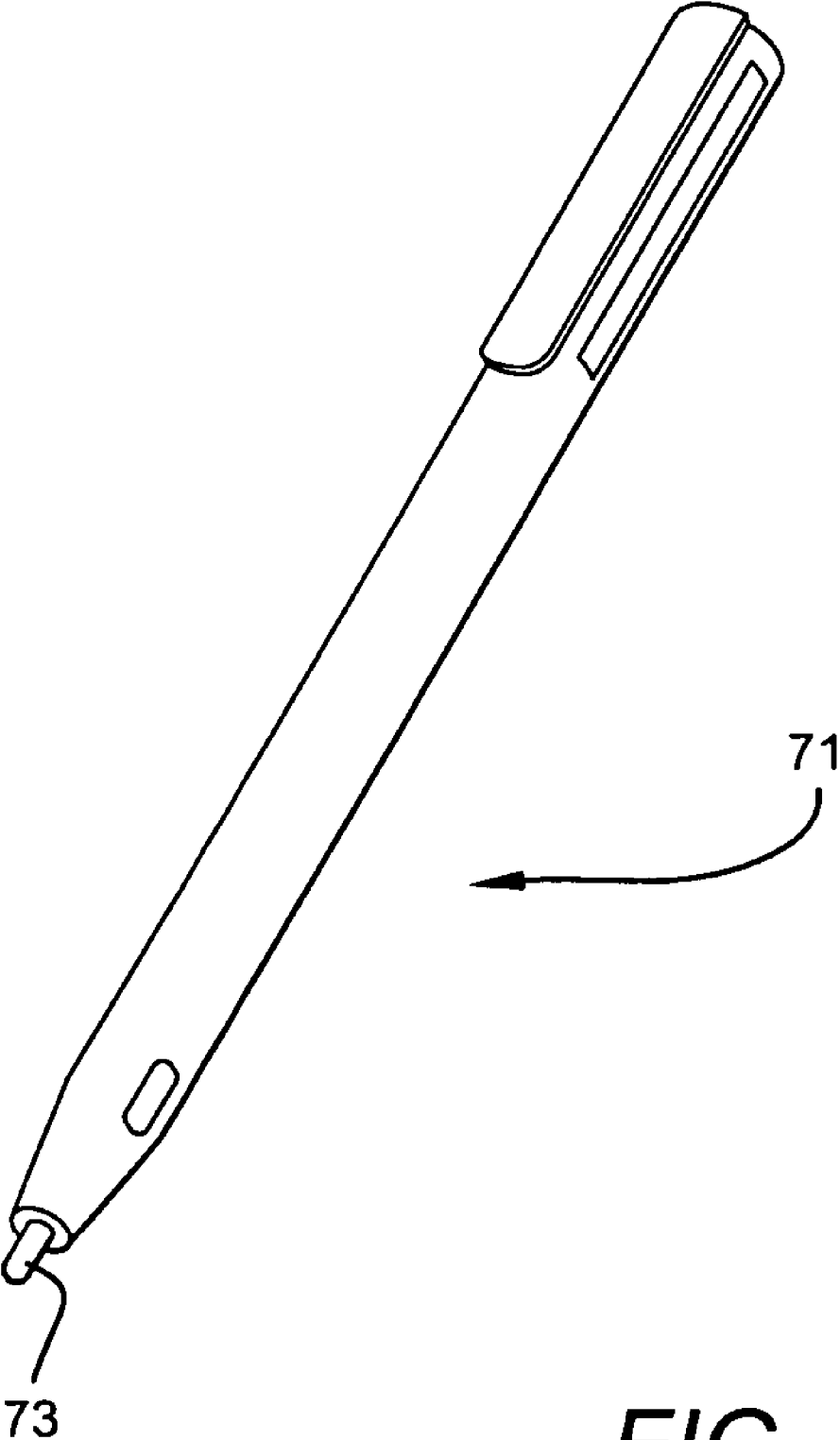


FIG. 4



**FIG. 5**

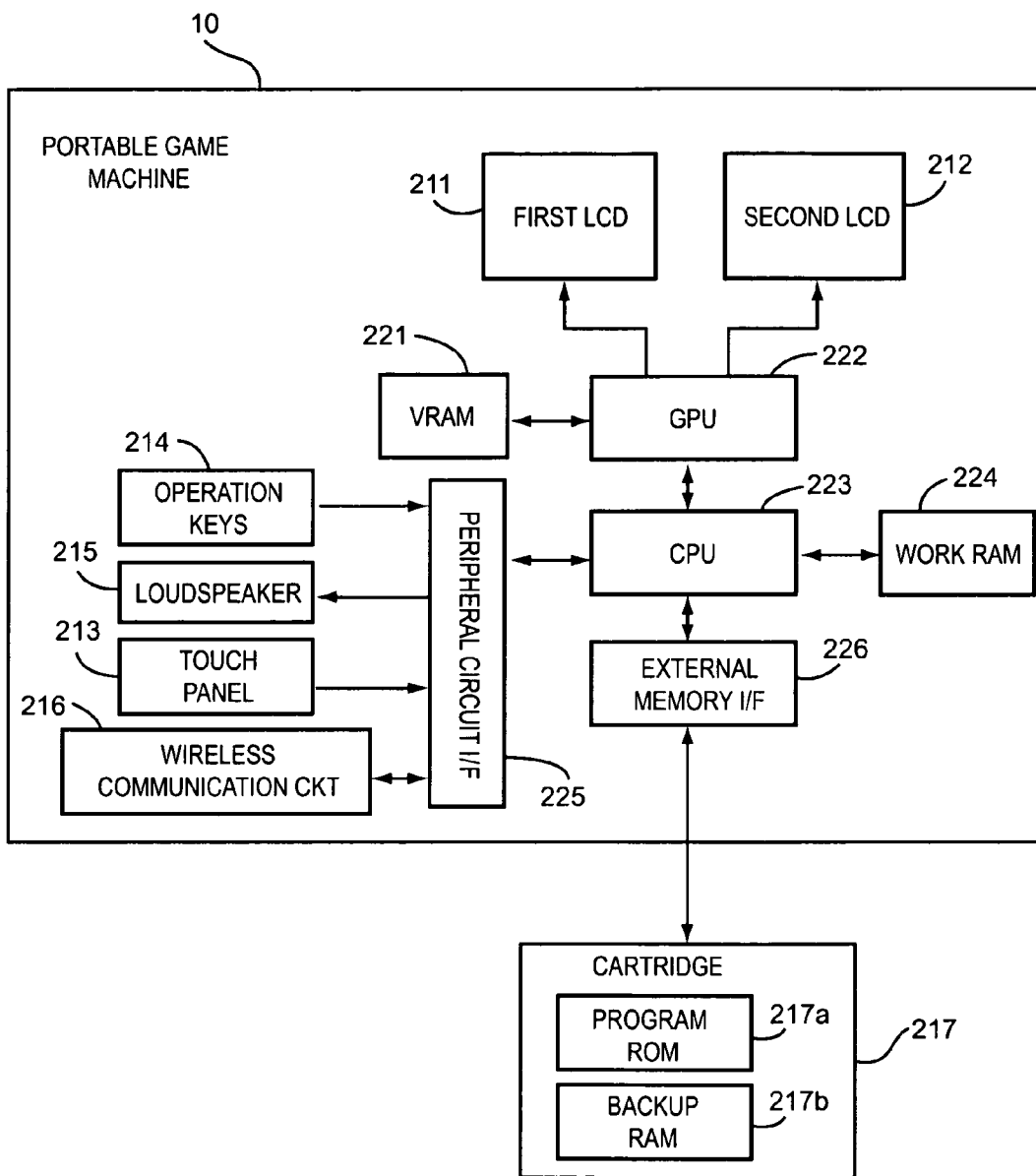


FIG. 6

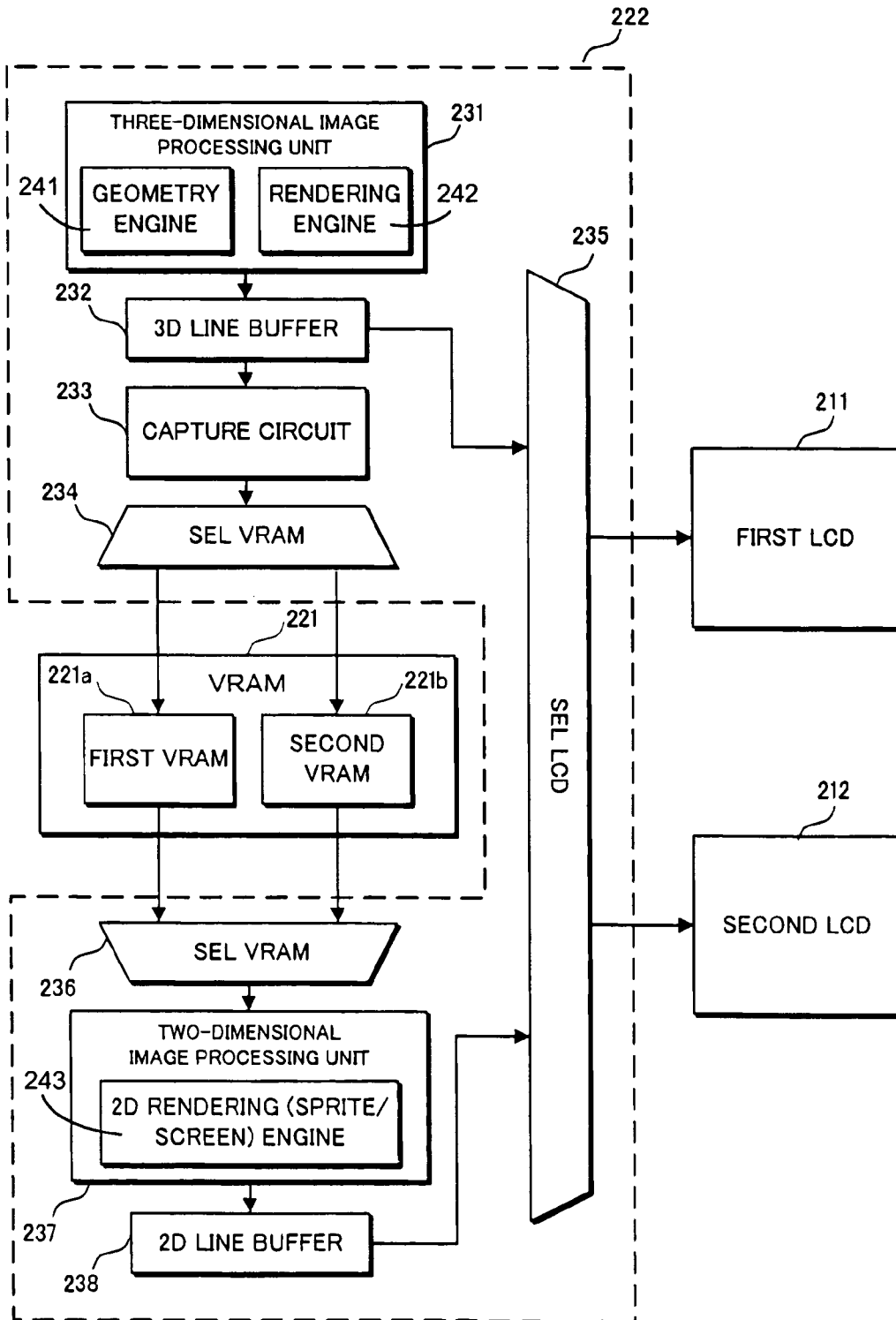


FIG. 7

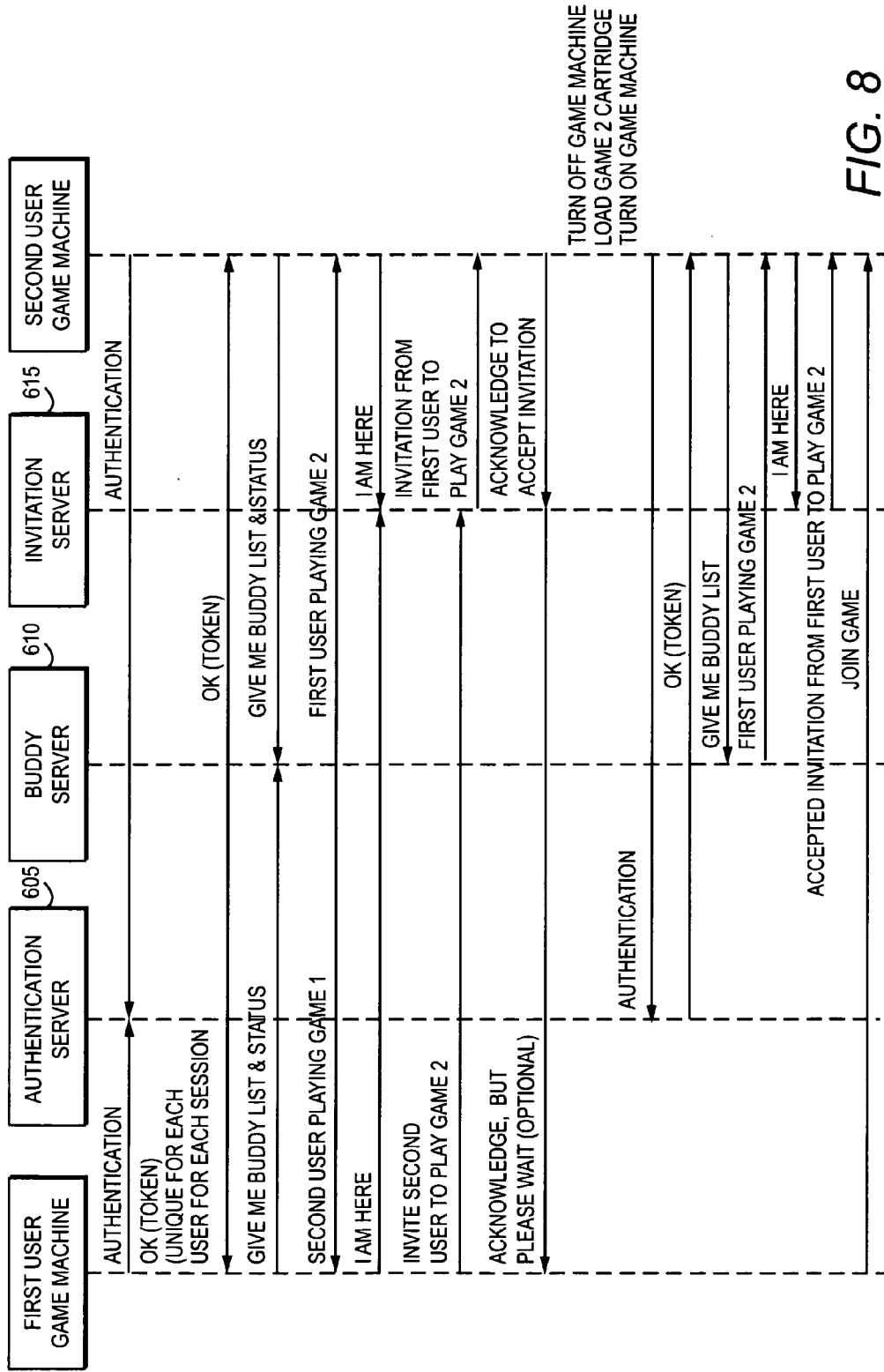


FIG. 8



**INVITATION SYSTEM FOR ON-LINE VIDEO GAMES**

**CROSS REFERENCE TO RELATED APPLICATION**

[0001] This application claims the benefit of provisional Application No. 60/709,426, filed Aug. 19, 2006, the contents of which are incorporated herein in their entirety.

**BACKGROUND AND SUMMARY**

[0002] This application describes an example invitation system that is particularly suitable for, but not in any way limited to, on-line video games.

[0003] As the popularity of on-line gaming increases, it becomes desirable to facilitate the ability of users to invite other users or players to play games.

[0004] In some instances, users may be playing a game and be invited to play another different game. In the case of many handheld and console game systems, to play another game, a user must change media (e.g., cartridge, optical disc, etc.) and reboot his/her game system (e.g., by turning the game system off and then back on). This changing of the media and rebooting of the game system can complicate the process of initiating game play between or among the users because the users must then, for example, navigate in the on-line environment to find each other.

[0005] The systems and methods described herein store certain information when a user accepts an invitation to play a game against another user. This information may stored by way of example without limitation in an invitation server or in a game system(s) used by one or both of the users.

[0006] Certain example methods and systems for inviting users to play video games involve receiving at an invitation server an invitation from a first user logged into the invitation server using a first game system that invites a second user logged into the invitation server using a second game system to play a game. The invitation is forwarded from the invitation server to the second user and an acceptance of the invitation from the second user is received at the invitation server. Invitation acceptance data indicative of the second user's acceptance is stored at the invitation server after the second user logs out of the invitation server. When the second user subsequently logs back into the invitation server, one or both of the first and second users is sent information based on the stored invitation acceptance that enables the first and second users to play the game.

[0007] The systems and methods described herein can simplify the process of extending invitations to play on-line games, particularly where the recipient of the invitation must prepare his/her game system to play a game by turning the game system off and then back on. By providing, among other things, an invitation server that keeps track of accepted invitations, users can be conveniently and, if desired, automatically joined to a game being played by another user, even if a game system is turned off and then back on.

**BRIEF DESCRIPTION OF THE DRAWINGS**

[0008] These and other features and advantages will be better and more completely understood by referring to the

following detailed description of exemplary non-limiting illustrative embodiments in conjunction with the drawings of which:

[0009] FIG. 1 is a perspective view of an example game system, with the system shown in an open, ready-to-use orientation;

[0010] FIG. 2 is a inverted perspective view of the game system shown in FIG. 1;

[0011] FIG. 3 is a front elevation of the system shown in FIG. 1, but with the game system shown in a closed position;

[0012] FIG. 4 is a rear elevation of the system shown in FIG. 3;

[0013] FIG. 5 is a perspective view of an example stylus for use with the game system shown in FIGS. 1-4;

[0014] FIG. 6 is an illustration showing an internal configuration of a portable game system;

[0015] FIG. 7 is an illustration showing an example internal configuration of a GPU 222; and

[0016] FIG. 8 is a diagram for explaining operation of an example on-line invitation system.

**DETAILED DESCRIPTION OF EXAMPLE EMBODIMENTS**

[0017] Referring to FIGS. 1 and 2, in an illustrative embodiment the game system 10 includes a main body 12 and a cover body 14 hingedly connected to each other along an upper edge of the main body 12 and a lower edge of the cover body 14 (references herein to terms such as "upper" and "lower" and "forward" and "rearward" are for ease of understanding and are made relative to an orientation of the game device where the cover body 14 is in an open position and the game is being held by a user in a normal operating position). Hinge elements 16, 18 and 20 on the main body 12 mesh with hinge elements 22 and 24 on the cover body, with a hinge pin (not shown) extending through the aligned hinge elements in conventional fashion. Note that because hinge elements 16, 18 and 20 extend from the upper (or inner) face 26 of the main body 12, the cover body 14 overlies the upper face 26 when the cover body 14 is closed over the main body. When the cover body 14 is in its fully open position, it is substantially parallel to the main body 12 but lies in a substantially parallel, offset plane. The main body 12 also has a lower (or outer) face 28 (FIG. 2) and a peripheral edge 30.

[0018] A first display screen 32 is recessed within the upper face 26 of the main body 12 with dimensions of approximately 2½ inches in length and 1⅞ inches in width, yielding a diagonal screen dimension of 3 inches. The screen in the exemplary embodiment is a backlit (e.g., 40 candelas), color liquid crystal display (LCD) with a display resolution of 256×192 dots (aspect ratio 4:3). This screen is touch sensitive and may be activated by a stylus, described further herein. A power button 34 is located in the upper left corner of face 26 and is used to turn the game on and off. A cross-shaped directional control button 36 is located adjacent and below the power button 34, and is used for game play control.

[0019] More specifically, display screen 32 includes a resistive-membrane touch panel that allows coordinates to be obtained in dot units. The touch panel can be operated with a finger or a stylus.

[0020] In the upper right corner of the main body 12, there are side-by-side “start” and “select” buttons 38, 40, respectively, with X/Y/A/B buttons 42 located adjacent and below the “start” and select buttons. Buttons 38, 40 and 42 are also used for game play control. A microphone 44 (which may be an omni-directional condenser microphone) is located below the left edge of screen 32 for use with specially designed games having a microphone feature. A battery recharge indicator LED 46 and a power indicator LED 48 are also located on the upper face 26, adjacent the lower edge thereof, below the right edge of screen 32.

[0021] With reference now especially to FIG. 3, a lower or forward portion 50 of the peripheral edge 30 (closest to the user) is provided with a volume control slide 52 and headphone and microphone connectors 54, 56 on either side of a first game slot 58. Slot 58 is especially designed for larger game cartridges or cards originally designed for use with the assignee’s Game Boy Advance® game system.

[0022] As best seen in FIG. 2, an upper or rearward portion 60 of the peripheral edge 30 is provided with an external extension connector 62 that permits connection to an AC adapter for recharging the internal battery (not shown), or for operating the game using household power. A second game slot 64 in edge portion 60 is designed for receiving memory or game cards especially designed for this game device. The second game slot 64 is smaller than the first game slot 58, reflecting the different sizes of the game cards. Openings 66, 68 form an elbow-shaped through slot adapted for securing a wrist strap (not shown), thereby enabling the user to secure the game device to the body and thus minimize the potential for losing or misplacing the game. A stylus port or holder, in the form of a blind bore 70 is located adjacent the wrist-strap mount for holding a stylus 71 (FIG. 5) before or after use.

[0023] The stylus 71 is a plastic pencil-shaped device with a rounded tip 73 and is used to activate the touch screen 32.

[0024] A pair of left, right control buttons (or shoulder buttons) 72, 74 are located on the peripheral edge 30, at the corners where the upper portion 60 of the peripheral edge 30 meets the side portions 76, 78 of the peripheral edge. The location of these buttons and the location of previously described buttons 34, 36 and 42 facilitate manipulation game control by the user’s thumbs and index fingers when the game is held with two hands in a natural and intuitive manner.

[0025] The lower (or outer) face 28 of the main body is provided with a battery cover 80 (FIG. 2) for accessing a rechargeable battery pack located within the main body.

[0026] The cover body 14 also has an upper (or inner) face 82 (FIG. 1) and a lower (or outer) face 84 (FIG. 2) connected by a peripheral edge 86. The upper face 82 incorporates a second display screen 88 of substantially the same dimensions as screen 32. Screen 88 is also a backlit color LCD. The cover body 14 also incorporates a pair of stereo speakers, with speaker grills 90, 92 located on opposite sides of the screen 88. Dimples or pads 94, 96 may be located above and laterally of screen 88. The dimples may be made of a compressible polymer or other suitable material and serve to dampen engagement of the inner surface 82 of the cover body 14 with the inner surface 26 of the main body 12 when the cover body is closed over the main body.

[0027] As already noted, the game card slot 58 is sized and adapted to receive a conventional game card designed for the by now well known Nintendo Gameboy Advance System®. Accordingly, the game card per se for slot 58 does not form any part of this invention and need not be described further.

[0028] FIG. 6 is an illustration showing an example internal configuration of the portable game system 10. As shown in FIG. 6, the portable game system 10 includes a CPU (central processing unit) 223, which is an example of a computer for executing the game program, and other components. The CPU 223 is connected to a work RAM (working storage unit) 224, a GPU (graphic processing unit) 222, and a peripheral circuit I/F (interface) 225. The work RAM 224 is a memory for temporarily storing, for example, the game program to be executed by the CPU 223 and calculation results of the CPU 223. The GPU 222 uses, in response to an instruction from the CPU 223, a VRAM 221 to generate a game image for display output to a first LCD (liquid crystal display unit) 211 and a second LCD 212, and causes the generated game image to be displayed on the first display screen 32 of the first LCD 211 and the second display screen 88 of the second LCD 212. The peripheral circuit I/F 225 is a circuit for transmitting and receiving data between external input/output units, such as the touch panel 213, the operation keys 214, the loudspeaker 215, and the wireless communication circuit 216, and the CPU 223. The touch panel 213 (including a device driver for the touch panel) outputs coordinate data corresponding to a position input (specified) with the stylus 71. The wireless communication circuit 216 may be configured for wireless communication in accordance with any of various known wireless protocols such as Bluetooth; any type of 802.11 (Wi-Fi) protocol; HiperLAN/1 protocol; HiperLAN/2 protocol; HomeRF protocol; etc. Although shown as a single block, wireless communication circuit 216 is intended to encompass arrangements in which two or more different protocols are usable or two or more different circuits (e.g., one for short-range communications such as WiFi and another for long-range communications using, for example, wireless telephone protocols) are provided. Of course, the example portable game system is not limited to wireless communication and may alternatively or additionally include circuitry that provides a wired communication link, e.g., to an internet access point.

[0029] To access the internet using the portable game system, wireless communication circuit 216 may, for example, wirelessly connect to an internet access point. Such an access point may be in a public location (e.g., stores, malls, libraries, etc.) or may be in a user’s home (e.g., a cable modem).

[0030] The CPU 223 is electrically connected to the external memory I/F 226, in which the cartridge 217 is inserted via slot 64. The cartridge 217 is a storage medium for storing the game program and, specifically, includes a program ROM 217a for storing the game program and a backup RAM 217b for rewritably storing backup data. The game program stored in the program ROM 217a of the cartridge 217 is loaded to the work RAM 224 and is then executed by the CPU 223. In the present embodiment, an exemplary case is described in which the game program is supplied from an external storage medium to the portable game system 10. However, the game program may be stored

in a non-volatile memory incorporated in advance in the portable game system **10**, or may be supplied to the portable game system **10** via a wired or wireless communication circuit.

[0031] FIG. 7 is a block diagram of an example arrangement for the GPU **222**. The GPU **222** includes two image processing units, that is, a three-dimensional image processing unit **231** and a two-dimensional image processing unit **237**. The three-dimensional image processing unit **231** includes a geometry engine **241** for calculating each vertex of a three-dimensional model based on three-dimensional model data and a rendering engine **242** for generating a game image from the three-dimensional model disposed on a virtual three-dimensional game space. The two-dimensional image processing unit **237** includes a 2D rendering engine **243** for generating a game image based on two-dimensional image data representing characters and two-dimensional image data representing backgrounds. More specifically, the two-dimensional image processing unit **237** disposes a two-dimensional image representing a character on a virtual screen called a “sprite” and a two-dimensional image representing a background on a virtual screen called a “screen”, and then synthesizes these virtual screens to generate a game image to be eventually displayed.

[0032] The three-dimensional image processing unit **231** is connected to the 3D line buffer **232**. The 3D line buffer **232** is a buffer memory for temporarily retaining image data for one scanning line of the first LCD **211** (or the second LCD **212**). The image data generated by the three-dimensional image processing unit **231** is stored in this 3D line buffer **232** sequentially by one line.

[0033] The 3D line buffer **232** is connected to a capture circuit **233** and an LCD selector (SEL LCD) **235**. The capture circuit **233** sequentially reads image data for one line stored in the 3D line buffer **232** and then sequentially stores the read image data in the VRAM **221**, which will be described further below, thereby capturing the game image generated by the three-dimensional image processing unit **231**.

[0034] The capture circuit **233** is connected to a VRAM selector (SEL VRAM) **234**. The VRAM **221** is provided with two VRAMs, that is, a first VRAM **221a** and a second VRAM **221b**. Instead of these two first and second VRAMs **221a** and **221b**, a single VRAM may be used with its two different storage areas being used as the first VRAM **221a** and the second VRAM **221b**. The VRAM selector **234** switches an output destination of the capture circuit **233** between the first VRAM **221a** and the second VRAM **221b**.

[0035] The first VRAM **221a** and the second VRAM **221b** are connected to a VRAM selector (SEL VRAM) **236**. The VRAM selector **236** switches a source of data to the two-dimensional image processing unit **237** between the first VRAM **221a** and the second VRAM **221b**.

[0036] The two-dimensional image processing unit **237** is connected to a 2D line buffer **238**. As with the 3D line buffer **232**, the 2D line buffer **238** is a buffer memory for temporarily retaining image data for one scanning line of the second LCD **212**. The image data generated by the two-dimensional image processing unit **237** is stored in this 2D line buffer **238** sequentially by one line.

[0037] The 2D line buffer **238** is connected to an LCD selector **235**. The LCD selector **235** switches an output

destination of the 3D line buffer **232** between the first LCD **211** and the second LCD **212**, and an output destination of the 2D line buffer **238** between the first LCD **211** and the second LCD **212**. In the present embodiment, the LCD selector **235** performs control such that, when the output of the 3D line buffer **232** is supplied to the first LCD **11**, the output of the 2D line buffer **38** is supplied to the second LCD **212**, and when the output of the 3D line buffer **232** is supplied to the second LCD **212**, the output of the 2D line buffer **238** is supplied to the first LCD **211**.

[0038] Additional information regarding portable game system **10** may be found in application Ser. No. 11/111,985 filed Apr. 22, 2005 and Ser. No. 10/921,957, filed Aug. 20, 2004, the contents of which are incorporated herein in their entirety.

[0039] The example embodiments described herein permit a user or player of a first portable game system to invite a user or player of a second portable game system to play an on-line game. Generally speaking, a user connected to the Internet with the above-described portable game system makes his/her presence known (“logs in”) to an authentication server. The user logging in to the authentication may be playing video games or using other applications being run on the portable game system such as an Internet browser. In an example implementation, the logging in is accomplished by the portable system passing to an authentication server a unique user identifier (e.g., user name, user ID, etc.) and, if desired, a password. Once the user is logged in, he/she is able to receive communications from other users.

[0040] One type of communication that a user might receive from another user is an invitation to play a game. If a first user wants to invite a second user to play Game A, the first user inserts Game A into the portable game system, connects to the Internet and logs into the authentication server as described above. The first user then navigates through the menu on Game A to an option that allows him/her to send a game invitation to another user. At this point, the first user enters information about the invitee, which in this example is the second user. The first user manually enters the second user’s unique user name or selects the second user’s information from a stored list of users (such as from a buddy list or list of users with whom the first user has recently played an on-line game). When the first user sends the invitation, it is sent either directly to the second user or to a server that processes the invitation request.

[0041] If the second user is not online, the server holds the invitation request for a specified period of time (which may be unlimited) until the second user comes online.

[0042] If the second user is online and is playing Game A, the user may accept the invitation and join the first user by connecting to the game host that the first user is connected to for the given online game instance.

[0043] If the second user is online, but not playing Game A, the second user may accept the invitation. However, the second user must change the program running on the portable game system by inserting the storage device (e.g., cartridge) containing Game A. This may involve the second user turning off power to his/her portable game system, removing a currently-connected storage device, and inserting a storage device containing Game A. In accordance with

the example embodiments, when the second user starts Game A, connects to the internet and subsequently logs into the invitation system, Game A will check to determine whether there are any outstanding accepted, but unplayed game invitations for the game. If there are, as in this example, and if the first user is still online waiting for the second user, the second user is immediately joined with the first user by, for example, connecting to a game host to which the first user is connected for the given online game instance. This enables the second user to avoid having to navigate through menu screens to try to locate the first user to join in a game session.

[0044] A more detailed explanation of an example invitation system for on-lines games will be explained with reference to FIG. 8. The example system includes an authentication server 605, a buddy server 610, and an invitation server 615. These servers may, for example, be provided and managed by a video game hardware/software manufacturer. Servers 605, 610 and 615 may be implemented as separate server computer systems or two or more of these servers may be implemented on the same computer system. The processing power needed for the servers will vary depending upon application demands. For example, for certain applications, the authentication server 605 may be any of a wide range of computers including a personal computer (PC). In accordance with one illustrative embodiment, the authentication server 605, buddy server 610 and invitation server 615 contain a dual-core 2 GHz CPU, such as AMD's Opteron, with 8 GB of RAM, and 200 GB hard disk. They may, for example, use a Linux OS, such as Red Hat Enterprise Linux 4, and the various databases may be managed by SQL database management software such as Oracle Database 10 Enterprise Edition.

[0045] In accordance with an example embodiment, the first and second users go on-line and respectively log into authentication server 605. The users go on-line by having their respective portable game systems execute a program having on-line functionality. This functionality may be incorporated in a game or other application executed by the portable game system. The game or other application may include appropriate information (e.g., uniform resource locators (URLs)) for accessing the various servers described below, or this information may be entered manually by the user. For purposes of the following discussion, it is assumed that the game or other application executed by the portable game systems includes appropriate functionality for accessing the various servers and executing the invitation feature described below.

[0046] Generally, the first and second users will log into the authentication server 605 at different times. Authentication server 605, after authenticating the user, grants permission to the user's portable game system to access buddy server 610 and invitation server 615.

[0047] In an exemplary embodiment, each portable game system has an associated unique identifier. For example, each portable game system may be, in an exemplary embodiment, identified by a media access control (MAC) address that is used to uniquely identifying a communication (e.g., Ethernet) chip embodied in the portable game system. The portable game system identifier used herein may be a combination of an identifier associated with the user and the system itself. A user identifier may be generated, for

example, by appending a random number to the MAC address for distinguishing, for example, different users in a household who use the same portable game system.

[0048] The first time a user connects to authentication server 605 a user identifier is created. The user identifier may be created in a portable game system and analyzed by authentication server 605 together with a portable game system-generated user password. Thereafter, when, for example, a user of a portable game system goes on-line, authentication server 605 prompts the portable game system to log on, resulting in server 605 running a user ID and password check.

[0049] Once server 605 authenticates the portable game system, a token is generated. In accordance with an illustrative embodiment, the token is a string of bits that is encrypted with a key shared by the portable game system and authentication server 605, that is digitally signed by authentication server 605. The token may have an associated expiration date that defines the period of time with which the token may be used.

[0050] When a user of the portable game system wants to access buddy server 610 and invitation server 615, as part of an authentication sequence, the portable game system passes the token to these other servers. The token is advantageously used by buddy server 610 and invitation server 615 to permit these servers 610 and 615 to authenticate a user without having to repetitively access a user database. Each of the buddy server 605 and invitation server 610 can determine that authentication server 605 generated the token by analyzing the associated digital signature.

[0051] Associated with authentication server 605 is a user database that stores an array of user IDs, passwords and the authentication status of each user. The authentication status of each user may be utilized, for example, to store status information indicating a user is banned from accessing the system.

[0052] Buddy server 610 is utilized to permit a user to register other users as "buddies." In this fashion, a user may use buddy server 610 to find on-line, one or more friends with whom to play. For example, buddy server 610 may be used to find friends with whom to play a racing game.

[0053] In accordance with one implementation, a first user using a first portable game system and a second user using a second portable game system may exchange user identifiers with each other. Such identifiers may be, for example, during the exchange process entered manually by the users. Thereafter, the first user may register the second user as one of his or her buddies with buddy server 610.

[0054] Buddy server 610 has an associated buddy database. This database stores, for example, for each stored user ID, the number of buddies associated with the user. A buddy user ID is stored for buddy users 0 to n. Various buddy groups 0 to n also may be stored. One buddy group may, for example, consist of a group of eight of a user's friends. Another group may be a group of relatives, while another buddy group may be members of a racing team. An individual may be a member of multiple groups, such as, for example, a family member and a member of a racing team.

[0055] After logging in and receiving a token, the first and second users may each access buddy server 610 to request

a buddy list and the on-line status of the buddies on the list. Generally, the first and second users will access the buddy server at different times. Buddy server **610** then provides the respective users with the requested buddy list and the on-line status of the buddies on the list. In situations in which a particular user has more than one buddy list, that user may be prompted by the buddy server to select one of the groups. In the present example, the first user is on the second user's buddy list and the second user is on the first user's buddy list. Upon receiving the respective buddy lists, the first user is informed that the second user is on-line and is playing, for example, "Game 1" and the second user is informed that the first user is playing "Game 2". In this instance, Game 1 and Game 2 are different games, but of course both users may in fact be playing the same game.

[0056] For purposes of explanation, assume that the first user wants to invite the second user to play Game 2. Using the invitation feature provided as part of Game 2, the first user sends an invitation to the invitation server **615** inviting the second user to play Game 2. Invitation server **615** then sends this invitation to the second user. The invitation may include information such as the user name of the first user, game setting information for Game 2 and a message from the first user (e.g., a challenge).

[0057] As shown in FIG. 8, at some time prior to the sending of the invitation, both the first and second users log into the invitation server. Although this logging in is shown in the FIG. 8 example as taking place immediately before the first user sends the invitation to the second user, this logging in may take place at any time after the user receives the token from the authentication server.

[0058] If the second user does not wish to play Game 2 with the first user, he/she declines the invitation by performing a specified action, e.g., providing a predetermined input to the portable game system. The declining of the invitation is communicated from the invitation server **615** to the portable game system of the first user, which provides an output informing the first user that the second user declined the invitation.

[0059] If the second user wants to play Game 2 with the first user, he/she accepts the invitation by performing a specified action, e.g., providing a predetermined input to the portable game system. The acceptance of the invitation is conveyed to the invitation server **615** and the invitation server **615** thereafter sends an acknowledgment of the acceptance to the first user. Although such an acknowledgment is not required, because there may be some time while the second user prepares his/her portable game system to play Game 2 (e.g., by turning off his/her portable game system, attaching to or inserting in the portable game system the proper game medium (e.g., cartridge, optical disc, etc.) for Game 2, and turning the portable game system back on), the acknowledgment may be desirable to let the first user know that he/she will need to wait for some time period. The invitation server may be adapted to provide short mini-games or other activities in which the first user may engage while he/she waits for the second user to prepare his/her portable game system to play Game 2. Alternatively or in addition, the invitation server may provide information such as advertising, news headlines, sports scores, etc. while the first user waits.

[0060] In response to the acceptance by the second user, the invitation server **615** stores acceptance information in

memory. This information may include, but is not limited to, a flag that the second user has an outstanding accepted invitation, game-related data (e.g., IP address of a game server to be used to play Game 2 with the first user; connection method such as using a password; game settings), and identification of the sender of the invitation. The information may also include a time stamp that can be used to set an expiration period for the accepted invitation.

[0061] After accepting the invitation, the second user prepares his/her portable game system to play Game 2 by, for example, by turning off his/her portable game system, attaching to or inserting in the portable game system the proper game medium (e.g., cartridge, optical disc, etc.) for Game 2, and turning the portable game system back on. The portable game system then logs back into the authentication server **605** (automatically or in response to an input from the second user) and receives a token when the log in process is complete. The second user may then log into the buddy server **610** (automatically or in response to an input from the second user) to request a buddy list. Upon receipt of the buddy list, the second user can confirm that the first user is still on-line. While the logging into the buddy server is optional, as noted, it enables the second user to confirm that the first user is still on-line. In one example implementation, the buddy list server **610** may be updated by information from the invitation server **615** and/or the portable game systems to show that the first user has extended an invitation to the second user and that the second user has accepted the invitation.

[0062] The portable game system of the second user then logs into the invitation server **615**. Using the user identifier information obtained during the log-in process, the invitation server **615** determines whether the second user has an (unexpired) outstanding accepted invitation. If so, the invitation server notifies the portable game system of the second user of the accepted information and provides the necessary information for enabling the second user to play Game 2 with the first user. This information may include an IP address for connecting to a game server (not shown), game settings and the like. The portable game system of the second user uses this information to thereafter join Game 2 with the first user.

[0063] The above system and method simplify the process of extending invitations to play on-line games, particularly where the recipient of the invitation must prepare his/her game system to play a game by turning the game system off and then back on. By providing, among other things, an invitation server that keeps track of accepted invitations, users can conveniently and, if desired, automatically joined to a game being played by another user, even if a game system is turned off and then back on.

[0064] An invitation server is not necessary in all implementations. For example, information relevant to the invitation from the first user could be stored in non-volatile memory of the second user's game system. The Game 2 program could access this stored information when the second user's game system is turned back on.

[0065] Various modifications may be made to the example invitation system. In one such modification, the system may be configured to provided additional options for responding to an invitation other than simply accepting or declining. For example, the recipient of an invitation who does not want to

play the game identified in an invitation, but wants to play some other game with the sender of the invitation, may be permitted to decline the invitation by generating another invitation inviting the original invitation sender to play some other game or engage in some other multi-user on-line activity.

[0066] Moreover, while the example invitation system is described with reference to portable game systems, it will be readily apparent that game consoles such as the Nintendo GameCube may also be used in the system. In addition, depending on the game or the multi-user on-line activity, a user using a portable game system (i.e., a first platform) may extend an invitation to a user using a game console (i.e., a different, second platform). Similarly, depending on the game or activity, invitations may be exchanged between users of platforms provided by different manufacturers.

[0067] Although the invitation system is described with reference to playing games, it may be generally utilized in connection with collaborative activities such as video conferencing.

[0068] Generally speaking, the systems, methods, and techniques described herein may be implemented in digital electronic circuitry, computer hardware, firmware, software, or in combinations of these elements. Apparatus embodying these techniques may include appropriate input and output devices, a computer processor, and a computer program product tangibly embodied in a machine-readable storage device for execution by a programmable processor. A process embodying these techniques may be performed by a programmable processor executing a program of instructions to perform desired functions by operating on input data and generating appropriate output. The techniques may be implemented in one or more computer programs that are executable on a programmable system including at least one programmable processor coupled to receive data and instructions from, and to transmit data and instructions to, a data storage system, at least one input device, and at least one output device. Each computer program may be implemented in a high-level procedural or object-oriented programming language, or in assembly or machine language if desired; and in any case, the language may be a compiled or interpreted language. Suitable processors include, by way of example, both general and special purpose microprocessors. Generally, a processor will receive instructions and data from a read-only memory and/or a random access memory. Storage devices suitable for tangibly embodying computer program instructions and data include all forms of volatile and non-volatile memory, including by way of example semiconductor memory devices, such as Erasable Programmable Read-Only Memory (EPROM), Electrically Erasable Programmable Read-Only Memory (EEPROM), and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and Compact Disc Read-Only Memory (CD-ROM). Any of the foregoing may be supplemented by, or incorporated in, specially-designed ASICs (application-specific integrated circuits). The computer program instructions may also be provided as data signals embodied in a carrier wave or other propagation medium via a communication link (e.g., a modem or network connection).

[0069] While the invitation system and method have been described in connection various embodiments, it is to be

understood that the invitation system and method are not to be limited to the disclosed embodiment, but on the contrary, are intended to cover various modifications and equivalent arrangements.

I claim:

1. A method for inviting users to play video games comprising:

receiving at an invitation server an invitation from a first user logged into the invitation server using a first game system that invites a second user logged into the invitation server using a second game system to play a game;

forwarding the invitation from the invitation server to the second user;

receiving at the invitation server an acceptance of the invitation from the second user;

storing invitation acceptance data indicative of the second user's acceptance at the invitation server after the second user logs out of the invitation server; and

when the second user subsequently logs back into the invitation server, sending to one or both of the first and second users information based on the stored invitation acceptance that enables the first and second users to play the game.

2. The method according to claim 1, wherein the invitation acceptance data is stored for a specified period of time after the second user logs out of the invitation server.

3. The method according to claim 1, wherein the invitation acceptance data comprises game-related data.

4. The method according to claim 1, wherein the invitation acceptance data comprises an IP address of a game server for the game.

5. The method according to claim 1, wherein the invitation acceptance data comprises connection information for connecting to a game server for the game.

6. The method according to claim 1, wherein the invitation acceptance data comprises password information for connecting to a game server for the game.

7. The method according to claim 1, wherein the invitation acceptance data comprises game settings.

8. The method according to claim 1, wherein the invitation acceptance data comprises identification data identifying the first user.

9. The method according to claim 1, further comprising:

sending to the first user an indication that the second user has accepted the invitation.

10. An invitation server for an on-line gaming system, comprising:

communication circuitry;

a storage device; and

processing circuitry for controlling the communication circuitry to forward an invitation from a first user logged into the invitation server using a first game system that invites a second user logged into the invitation server using a second game system to play a game,

wherein the processing circuitry stores in the storage device acceptance data indicative of the second user's

acceptance at the invitation server after the second user logs out of the invitation server.

11. The invitation server according to claim 9, wherein the acceptance data is stored for a specified period of time after the second user logs out of the invitation server.

12. The invitation server according to claim 9, wherein, when the second user subsequently logs back into the invitation server, the processing circuitry controls the communication circuitry to send to one or both of the first and second users information based on the stored invitation acceptance that enables the first and second users to play the game.

13. The invitation server according to claim 9, wherein the processing circuitry controls the communication circuitry to send to the first user an indication that the second user has accepted the invitation.

14. The invitation server according to claim 9, wherein the invitation acceptance data comprises game-related data.

15. The invitation server according to claim 9, wherein the invitation acceptance data comprises an IP address of a game server for the game.

16. The invitation server according to claim 9, wherein the invitation acceptance data comprises connection information for connecting to a game server for the game.

17. The invitation server according to claim 9, wherein the invitation acceptance data comprises password information for connecting to a game server for the game.

18. The invitation server according to claim 9, wherein the invitation acceptance data comprises game settings.

19. The invitation server according to claim 9, wherein the invitation acceptance data comprises identification data identifying the first user.

20. A computer-readable medium having computer readable code embodied therein for use in the execution in an invitation server of a method for inviting users to play video games, the method comprising:

receiving at an invitation server an invitation from a first user logged into the invitation server using a first game system that invites a second user logged into the invitation server using a second game system to play a game;

forwarding the invitation from the invitation server to the second user;

receiving at the invitation server an acceptance of the invitation from the second user;

storing invitation acceptance data indicative of the second user's acceptance at the invitation server after the second user logs out of the invitation server; and

when the second user subsequently logs back into the invitation server, sending to one or both of the first and

second users information based on the stored invitation acceptance that enables the first and second users to play the game.

21. The computer-readable medium according to claim 19, wherein the method further comprises:

sending to the first user an indication that the second user has accepted the invitation.

22. A data signal embodied in a carrier wave and representing a sequence of instructions which, when executed by a processor of an invitation server, cause the processor to perform a method for inviting users to play video games, the method comprising:

receiving at an invitation server an invitation from a first user logged into the invitation server using a first game system that invites a second user logged into the invitation server using a second game system to play a game;

forwarding the invitation from the invitation server to the second user;

receiving at the invitation server an acceptance of the invitation from the second user;

storing invitation acceptance data indicative of the second user's acceptance at the invitation server after the second user logs out of the invitation server; and

when the second user subsequently logs back into the invitation server, sending to one or both of the first and second users information based on the stored invitation acceptance that enables the first and second users to play the game.

23. The data signal according to claim 21, wherein the method further comprises:

sending to the first user an indication that the second user has accepted the invitation.

24. A method for inviting users to play video games comprising:

receiving at a first game system used by a first user to play a first game an invitation from a second user using a second game system inviting the first user to play a second game different than the first game;

storing in nonvolatile memory of the first game system invitation data indicative of the invitation from the first user; and

re-starting the first game system to play the second game, wherein the re-starting comprises using the invitation data stored in the non-volatile memory to enable the first game system to automatically initiate game play processing for playing the second game with the second user.

\* \* \* \* \*