

Abstract

In this paper, the issues and opportunities involved in using a PDA for a universal remote control are discussed. As the number of home entertainment devices increases, the need for a better remote control interface system becomes apparent. The PDA's flexible input and display mechanisms provide several advantages over traditional physical button-based interfaces. Building remote control functionality in software allows for more automation than traditional fixed remote controls. As a proof of concept, a prototype universal remote control was implemented using the Palm m105 PDA. The steps in creating the interface design are described and the hardware and software issues of the prototype are discussed.

Motivation

Limitations of Current Devices

Currently, when a high-tech entertainment system is put together, the inevitable result is that there will be three to five remotes congregating on the coffee table. Each one can interface with one or more devices, such as televisions, VCRs, DVD players, amplifiers, and game systems like the PlayStation 2. While it is possible and very desirable to trim the number of remotes needed, this process can be difficult, annoying, and time-consuming. In addition, each remote contains a huge array of buttons that can often be confusing.

Configurable universal remotes have been designed to replace the multiple remote controls for an entertainment system with a single device. However, typical universal remotes require a complicated setup procedure that involves digging through the manuals for device code numbers. Since the number and layout of buttons on a universal remote is fixed, all available options must be presented to the user at all times.

Using a PDA to Address These Limitations

A personal desktop assistant (PDA) provides a soft display and input system, facilitating more flexible interaction. A PDA is capable of displaying only pertinent information and keeping track of the user's current state. A PDA can also be easily updated since it is controlled with software stored in the device's mutable memory as opposed to hardware or read-only memory (ROM). In addition, online help can be made available within the system to eliminate the need to hunt and find the manual.

Similar Research and Products

Other products, such as the HP iPAQ H3900 [1] and the Philips TSU2000 [2] have been designed to solve the problems above.

The TSU2000 is basically a PDA that is designed with one purpose: to be used solely as a universal remote. It has built-in support for several devices. It also has a "learning mode" where the user can configure their universal remote to learn signals emitted from other remote controls and assign them to buttons. The main drawback of the TSU2000 is that it requires a large amount of configuration. The user, who probably isn't experienced

in user interface design, must layout buttons and labels for his or her combination of devices. This results in inconsistent interfaces between households.

The H3900 is a standard PocketPC PDA that happens to come with software enabling its use as a universal remote. It is similar to the TSU2000 discussed above, except that it has built-in support for a huge range of devices. It also sports a \$400 price tag.

System Design

Constraints

Using a PDA as a remote control imposes some constraints upon the system. They are enumerated below.

- 1 The software should run on a low-cost PDA. High cost would be a prohibitive factor in widespread adoption.
- 2 The interface should be accessible to people with poor eyesight or motor control. A monochrome, low-resolution display on a low-end PDA might be difficult to read for those with poor eyesight. Similarly, since “soft” buttons cannot be felt, a greater level of motor control is required to use the interface.
- 3 The interface cannot rely on color or backlights since very few low-end PDAs have them. However, backlights might be required for low-light situations.
- 4 Since this system is designed to replace an established and familiar product, its behavior must be consistent with that of current remote controls. It should also

effectively use knowledge of the world by employing idioms and designs that existing remote controls already use.

- 5 The user should always be able to understand what actions are available from the information currently displayed. The number of "modes" should be minimized in order to prevent the user from getting lost in the interface.

User Analysis

Before beginning the interface design, we analyzed potential users of the system and enumerated their likely characteristics, motivations for using the system, probable knowledge, and usage environment.

Since our interface attacks the problem of having too many complex remote controls for a home entertainment system, the user will likely be somewhat technology-savvy. The intended user will probably have at least two or three components of a home entertainment system and will have had experience with a PDA or similar handheld, touch screen-based device. We also made the assumption that the user will be able to read and understand the English language.

The user will expect the universal remote to automate and simplify their common interactions. The user will also expect the remote to behave in a manner similar to other standard remote controls on the market. Nearly everybody has used a traditional remote control and by mimicking their operation, our system would make more sense to new users.

To directly obtain information about potential users, we created a questionnaire. The questionnaire asked questions such as “How many home entertainment devices do you own?” Unfortunately, due to legal issues, we could only distribute the questionnaire to a very specific segment of our general intended audience. A blank copy of the questionnaire is included in the appendix. In total seven questionnaires were returned, responses from six males and one female. Highlights from these responses are:

- 1 All people had more than one entertainment device making a universal remote potentially useful.
- 2 The most common dislikes of their current remote situation is that they have to keep track of multiple different remotes or use one remote with a large array of small buttons crammed together.
- 3 All people included channel change, volume control and power as the most common buttons used. Mute and the number pad were also frequent, appearing on three and five respectively. Three people included fast forward, play, stop and rewind.

In our design, we considered the following environmental concerns: The interface will be used in low-light, loud situations typical in a home theater environment. The interface will also be shared by several people.

Task Analysis

In order to guide our efforts in laying out the design, we enumerated the common tasks an intended user will perform.

Interface Initialization. The first time the interface is turned on, it must introduce itself to the user and walk the user through device detection and setup.

Device Power-On. When the remote control is turned on, it should turn on devices the user probably wants to use. The list of devices to turn on must be configurable.

Device Detection and Setup. The interface has to allow addition of new home entertainment devices when they are available. Supported devices include the television, VCR, DVD player, PlayStation 2 (only for its use as a DVD player), amplifier, and the remote itself. Configuration of each device involves specifying how the devices are connected. A common question would be: is the VCR on channel 3 or an RCA input of the TV? Each device must have its own configuration properties and setup screen. Some devices like amplifiers require more configuration than others, such as the PlayStation 2.

Static TV Interface. Since the television is the central focus of any home entertainment system, the universal remote interface must pay special attention to it. Common TV controls such as channel, volume, and mute should be easy to access and available at almost all times. Because they are so commonly used, it is more important to have these controls in a consistent location.

Media Interface. Currently, most media playback devices exist externally to the television set. The user will want to access them in a consistent way, so the interface must provide a media control interface with play, pause, stop, rewind, and fast forward buttons. The current media -- be it VHS tapes, CDs, or DVDs -- must be selectable.

Intelligent Knowledge of the Amplifier. Sometimes, many devices are connected to the TV through the amplifier. The interface must ask the user for this information so

that the remote behaves intelligently. For example, when the user presses play on the VCR controls, the remote will configure the TV and amplifier so that the VCR is currently displayed on the TV and heard through the speakers connected to the amplifier.

Help Interface. On each screen of the interface, a help menu must be made available. If the user does not understand a particular aspect of a screen, he or she can open that screen's help page to view information about the interface.

Device Automation. The interface should automate interactions with the home entertainment system. To simplify the user's conceptual model of the interactions between devices, the remote will convert commands such as “Play DVD” into all device commands required to start DVD playback on the TV. For example, the TV, and possibly amplifier, must come on and be switched to the right mode for DVD playback. The DVD player must also be turned on and playback must be started.

Persistent State. The interface must remember its state between usage sessions. For example, once a device has been added and configured, it should not be removed when the device is turned off. The user would still be able to explicitly remove the device, however.

Layout

When deciding how to layout the interface, a few important considerations had to be addressed. The first of these was that the amount of information on each screen had to be limited. It was not desired to expect the user to scroll up and down the screen to get to different controls. While this was needed for some of the device specific screens, the main screen displays all the controls without a need for scrolling. Another consideration

was to make the buttons as large as possible to accommodate people with poorer eyesight and lesser motor control. Finally, the user's knowledge of remotes was considered. The graphical buttons for the media devices needed to use the standard symbols that have been appearing on remotes for decades. For example, the play button for a VCR was represented as a triangle pointing right. The initial layout designs for the prototype can be found in the appendix. These layouts are the graphical representation of the device flowchart shown below.

INSERT DEVICE FLOWCHART

Storyboarding

This remote needs to automate as much as possible to make as much of the remote's operation transparent to the user. Once the user gets a brand new remote and fires it up for the first time, it will step the user through an automatic setup phase. In this phase, it will ask the user what kind of devices will be under the remote's control and how these devices ultimately connect to the TV. This step will also ask the user which devices to automatically turn on when the user fires up the remote. Once this setup is done, and on all subsequent startups, the user will be presented the main screen. This screen will provide all of the most common functions desired by users. It will have basic TV controls such a volume, channel select and mute. It will also allow the user to choose among the media devices and gain access to the most common buttons of each. All of these options will be presented on one screen as discussed in the layout section. This eliminates the need for the user to navigate to other screens or scroll down the screen to get at the most useful options. However, the options that are used with less frequency

such as TV brightness control can be found in device specific pages. These pages will present all of the options for the specific device that are not available in the main screen. A menu will be accessible from the main screen that will allow the user to navigate to these device specific screens as well as screens that allow for the addition of new devices, the removal of existing ones, and changing which devices are powered on when the remote is turned on. Finally, each page will present a clear means to get back to the previous screen. At most, the user will only get two levels past the main screen.

Technical Development

Hardware

Our prototype remote control interface was implemented on a Palm m105, a common low-end PDA. It has 8 MB of RAM, 160x160 monochrome LCD touch-sensitive display, and a 16 MHz Motorola CPU, and runs PalmOS. The device itself weighs a mere 4.4 ounces with dimensions of 4.66" x 3.12" x 0.72".

Software

To develop the interface, we used a very limited subset of Java, called MIDP 1.0, suitable for handheld devices. The Java Wireless Toolkit development suite came with everything needed to develop a PalmOS application. In order to test the application, we downloaded an emulator and PalmOS ROM images from the Palm developer site. This way, we didn't have to transfer our application to the actual Palm device in order to test it.

The MIDP software allows software developers to create programs that will run on several wireless devices: PDAs, cell phones, embedded systems, and recently, even the Gameboy Advance. However, this portability comes at a cost: the API is very limited and did not allow us to fully realize the desired page layouts attached in an appendix.

We could have implemented our displays in two ways. The first would have been to create all of the interface controls ourselves and handle all low-level click events and drawing ourselves. The second was to map the essence of our planned screens to the MIDP high-level UI API, which converts abstract user actions into interface elements. In the interest of time, we chose the second method. We had less control over how the final interface was laid out, but it ended up being more portable and vastly quicker to develop, letting us spend more time on user analysis, task analysis, user testing and evaluation.

Device Setup

When the user gets the remote and uses it for the first time, it will automatically take them through setup and configuration steps. There should be no need for the user to read or even look at a manual before beginning this setup procedure. The remote will start by asking which devices the remote should control. Adding a device is as simple as choosing it from a list of available devices. This list enumerates the available devices in general terms instead of by specific make and model. For example, one of the available devices is a VCR, not something specific like Panasonic PV-8450. The reason for this is that listing devices in the general instead of the specific makes it possible to show the entire list of supported devices in one compact and easy to read list. Once the user checks

all of the devices that they want to put under the remote's control, they click the "Add" button on the bottom of the screen.

The user is then taken to device specific setup pages. For example, if the user added a VCR, then they would be taken to the "VCR Setup" page. The user visits these setup pages in the order that the selected devices appeared in the list. An important consequence of this is that since the amplifier appears first, if it is selected it will be setup before the other devices. This behavior is needed because if there is an amplifier, several of the other devices will potentially be plugged into it instead of the TV.

Once the user gets to the device setup page, then the determination of the specific from the general will occur. In the prototype implemented, it assumed that a new technology would be integrated into the entertainment devices. This technology would allow the PDA to send out an IR signal to the device and it would respond with its code for communication. This setup would be ideal, as it would hide all of this from the user. However, if this prototype were to be developed into a viable product, it would be easy to add a question at the top of each device setup page that asks the user to select their particular model from an exhaustive list of supported devices. Selecting their device would then allow the PDA to look in an internal database to find the needed code for communication. It would then execute the code without requiring the user to lookup and enter the code as several universal remotes do now.

There are two types of device setup pages, the amplifier and non-amplifier pages. The amplifier page asks the user to select how many speakers the amplifier controls as well as

if the TV needs to be set to a particular channel or “TV Video” mode. The non-amplifier pages ask different questions depending on if an amplifier was added. If one was, it will ask if the device is connected to the amplifier and if it is, then it will ask what video mode the amplifier needs to be set to for the device in question. If there is not an amplifier, or if the device is just hooked directly into the TV, then the user will be asked if the TV needs to be set to a particular channel or “TV Video” mode.

Once all of the device setup pages have been visited and the devices are configured, one final page is presented to the user. This page presents the devices added in a checklist and asks which of the devices should be powered on when the remote is turned on. This will keep the user from having to go to each device specific page to turn on each one. Once these selections are made, the user has completed remote setup and configuration and will be presented with the main screen.

Main Screen

The single most important screen on the device is the main screen. Its main goal is to provide all of the most common functions desired by users. The experience of the designers, and more importantly, the results of the questionnaires determined what tasks the main screen would need to provide. This functionality includes TV channel selection, TV volume control, and mute. In addition, the main screen offers a method to switch between the media devices added under the remote’s control. This is accomplished without hiding the TV controls. The figure below shows a screenshot of the main screen.

The top half of the screen is devoted to the selected media device controls. A pull down menu just below these controls is used to select between the devices added. Once this is chosen, the controls on the top half of the screen change to present those most important to the selected device. For example, if the VCR is chosen, the controls shown are play, pause, record, stop, fast forward and rewind. In addition, these controls are displayed graphically with symbols common to users such as two parallel vertical lines for pause. The bottom half of the screen always displays the controls for the TV. In this way, the user would never have to leave the main screen to access the most common functions. In addition, due to the system setup, pressing play on the VCR button section would cause the remote to take care of all the steps to get the VHS tape images onto the TV. For example, if the VCR is attached to the amplifier through Video2 and the amplifier requires the TV to be in “TV Video” mode, the simple action of pressing the play button on the remote would set the TV to “TV Video” mode, turn the amplifier to Video2 and then press play on the VCR.

Finally, along the bottom of the main screen are a few options. One of these options is the “Power” button for the remote. Before the remote turns off, it would turn off all devices that it turns on at startup. Another option is the “Menu” button that allows access to the device menu.

Device Menu

The device menu is the pathway to all of the screens on the device. This screen consists of a list of other pages that can be visited. If there are still potential devices to be added, the first option will be “Add Devices”. If there are devices added, the next option will be

“Remove Devices”. The next several options will take the user to specific media device pages. Only the devices added will be included in the list. The final option is the remote startup page. Going to this page allows the user to tell the remote which devices to turn on at startup or to remove devices from automatic startup. The back button at the bottom of the screen takes the user back to the main screen.

Add and Remove Devices Screen

The add screen shows all of the media devices supported by the remote that are not already added to the remote’s control in a check list. A consequence of this design is that only one media device of a particular type is controllable by the remote at a time. This is not considered to be a major problem because most users do not wish to control two different VCRs or DVD players with the same remote. All devices selected in the checklist are added when the user presses the “Add” button at the bottom of the page. After the user clicks add, the user is then taken to the setup pages for the devices selected. These pages are the same ones described in the Device Setup section. A special case occurs when the user adds an amplifier. Since the other media devices already added to the remote may now be connected to the new amplifier instead of directly to the TV, the remote revisits all of the setup pages for the media devices already under control of the remote. This allows the remote to keep track of the system setup. The remove screen presents all the media devices already added in a check list. Any selected devices will then be removed from the remote’s control when the “Remove” button at the bottom of the page is pressed.

Device Specific Screens

The remainder of the screens in the remote is dedicated to a specific media device. It is in these screens that the less common device tasks can be found. For example, the TV screen allows for adjustment of the brightness, contrast and color. Likewise, the amplifier screen allows for the decibel level of each speaker controlled by the amplifier to be adjusted. The VCR screen has a means for the user to input a date and time for the VCR to start and end a recording. These device specific screens should only be visited sparingly and in specialized circumstances.

Future Work

The most important future work on this project is implementing a connection to the IR port of the PDA. With this connection, the remote could be made to actually function as a remote and communicate with the common entertainment devices. The currently implemented prototype is only capable of showing the proof of concept and providing a means to evaluate the user interaction of the device. However, the interface was not implemented with the full graphical polish desired as evident by looking at the layouts in the appendix. The software used to develop the prototype did not directly support advanced interface graphics development. However, given more time, the graphical layout could be implemented to provide the level of visual polish desired. Finally, a full and representative user study is desired for future iterations of the interface design.

Acknowledgements

The authors would like to thank Jon Heseman for finding a great deal on the PDA used for the prototype. Additionally, thanks goes to Jared Knutzon and Jessica Walter for their

input in the task analysis phase of the design. Thanks also goes to Palm Inc. for the PalmOS ROM images used in development.