Application design for a smart watch with a high resolution display

Chandra Narayanaswami, M. T. Raghunath

Wearable Computing Platforms
IBM TJ Watson Research Center
Yorktown Heights, NY 10598
{chandras, mtr}@us.ibm.com

Abstract

Advances in technology have made it possible to package a reasonably powerful processor and memory subsystem coupled with an ultra high-resolution display and wireless communication into a wrist watch. This introduces a set of challenges in the nature of input devices, navigation, applications, power management, and other areas. This paper describes a platform we have developed and the decisions we have made about how to address these challenges.

1 Introduction

A wrist watch is an attractive form factor for a wearable computer. It has the advantage of always being with you; and it can be instantly viewed with the flick of the wrist. By comparison, devices such as pagers, cell phones, and PDAs are typically worn on belts or kept in pockets and need to be picked up and opened first before they can be accessed. One of the reasons for the success of the Palm was its moving to an *instant-on* paradigm. Wristwatches move us to the next step; to an *instantly-viewable* paradigm.

Because people generally keep watches on their wrists, watches are less likely to be misplaced compared to phones and pagers. For example a hip holster is not the best place to keep a cellular phone while sitting in a car and so people tend to keep them in the car seat and forget them when they leave the car in the parking lot.

The watch form factor requires a relatively small screen size, and there is not much room for input devices or batteries. The value of a wristwatch platform depends on finding good solutions to these issues. To interact with the watch, we need both hands since the hand on which the watch is worn is practically useless for controlling input devices on the watch.

Several smart watches are available commercially today. Personal Information Management (PIM) applications are provided on the Seiko Ruputer™ [1], the onHand PC™ [2], the Casio PC-Unite™ [5] and the Timex DataLink™ [4]. These watches pack an impressive amount of function but have low resolution displays. This limits the amount and type of data that can be displayed on their screens. Some of the above have cumbersome/confusing user interfaces with many buttons leading to limited adoption. For example, on the Seiko Ruputer™ it is quite difficult to get back to the clock face from any application. The Casio WMP-1V™ [5] is a wearable MP3 player, the Casio WQV-1™ [5] is a wearable digital camera, with a 120x120 display, the Casio BP 100™ measures blood pressure and the Casio Satellite Navi™ [6] provides GPS on a watch. The Timex Beepwear™ watch has a built in pager. The Swatch Access™ watch [10] has an RF id tag which can be used as a ski pass at some ski resorts. Swatch is also working on a cell phone in a watch called Swatch Talk™. The NTT PHS Personal Handy Phone System [7] and thc Samsung Watchphone Anycall™ [7] are watches with a telephonc. Polar produces heart rate monitor watches (Smart Edge™, Beat™, Protrainer NV™, Lady Beat™ Target™, Pacer™).

There is plenty of interest in adding function to watches in the industry. Our interest is not in so much in providing a watch, as in providing an open, extensible computing platform in a small form factor. Our objective is to understand the challenges in packaging, hardware design, power management, and embedded software.

2 Wrist watch computer overview

The *wrist watch computer* is a wearable personal information access and alert notification device with both short range and long range RF wireless connectivity. The initial display for the watch is a 96x120 pixel monochrome LCD display. Subsequently, the display will be enhanced to an ultra high resolution (600 dpi) monochrome display. The applications and user interfaces described in this

0-7695-0795-6/00 $10.00 © 2000 IEEE

paper are targeted to the higher resolution display and were prototyped on a commercially available backlit VGA resolution display.

These display and communication capabilities could make the *wrist watch computer* the preferred viewer for other devices that are around you and for your electronic agents running on web, business or e-commerce servers on the intranet or the internet.

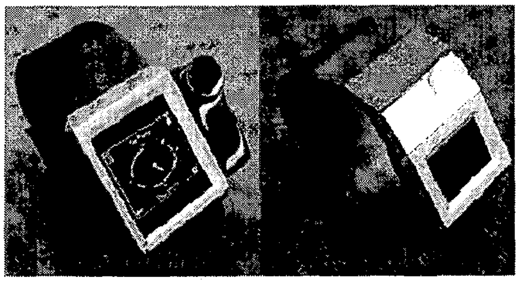

Figure 1: Enhanced Shell and Basic Shell

At the core for the *wrist watch computer* is an ARM 7 based low-power system board that runs Linux. The system board uses state of the art technology such as Surface Laminar Circuit™ (SLC). It is 34.7x27.5x3.0mm in size. Additional communication cards are used to support short range wireless protocols such as Bluetooth. Power is supplied by a rechargeable lithium-polymer battery.

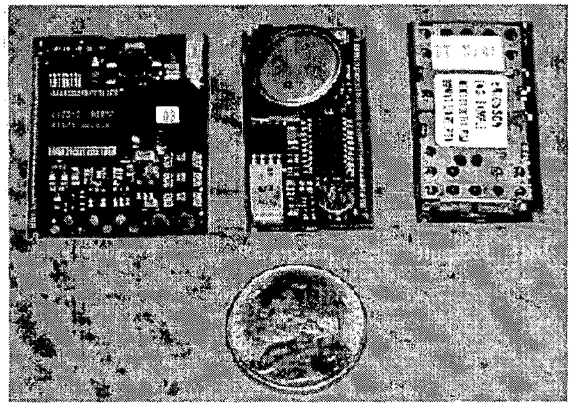

Figure 2: Main card and communication cards

Figures 1 and 2 show initial wrist watch computer prototypes.

The watch must be viewed as a companion to some other key devices such as a PC in an office, a cell phone in a bag, or a wireless beacon in a building. So the watch need not do all the functions a user wants but must communicate and cooperate with other devices so that the strengths of each device can be exploited to the maximum.

3 Selection of input devices

The choice of input mechanisms is critical for ease of use. In our effort to find the right input devices for the wrist watch computer, we first studied the advantages and disadvantages of input devices on a set of commercially available products including the Palm Pilot™, Psion Series 5™, HP Jornada™ and other handheld PDAs, the RexPro 5-DS™, RIM Interactive Pager 960™, Hertz NeverLost™ car navigation system, several digital cameras, remote controls, smart watches, and cell phones. The pros and cons of several choices for input devices are described below:

Keyboards: Though useful for entering large amounts of text, watches do not have the surface area and volume needed for hardware switches such as keyboards. Even if we were able to fit a tiny keypad with a limited number of keys on the wristwatch, it would be difficult to use for most people.

Touch Screen: A touch sensitive screen can be used on the wrist watch computer to provide for soft switches and character input. The relative sizes of a watch face and a human finger limit the number of distinguishable touch zones to four or five, although a stylus could be used on the watch face to accept fine grained input such as graffiti. It was clear to us through experimentation with the Casio VDU 200B™ watch that touch screens frequently showed screens other than the desired time screen due to accidental activation every now and then. Despite this shortcoming we felt that a touch screen would be a good input mechanism to support on a wrist watch since it was more versatile and more elegant compared to buttons. And for usabilitiy reasons, we limit the number of distinguishable zones to the four - one for each corner.

Scrolling: A touch screen provides a good selection mechanism but does not perform too well as a scrolling mechanism. For this purpose we considered a rocker switch, a rotating bezel, and a roller switch. The rocker switch rocks up or down. It can be pushed in to select. The roller wheel and the bezel allow complete revolutions to scroll up or down. The roller wheel may be pushed towards the center to select and the bezel may be pressed down to select. Initial user tests revealed that a rocker switch (fitted on a RexPro 5-DS™) was superior to touch screen actions, but a roller or bezel that allowed complete rotations seemed easier to use and more natural. Since our present watch is rectangular, we chose a roller wheel over the rotating bezel.

We positioned the roller wheel to minimize the chance of accidental activation by bending the wrist. The user has to push in the southwest direction to select.

As shown in Figure 1, the wheel is also positioned in the northeast corner to accommodate both left and right handed users. Right handed users anchor their thumb to the bottom left corner of the watch and use their index finger to roll the wheel, moving the index finger north to south or vice versa depending on the desired direction of rotation. Left handed users who wear the watch on their right wrist tend to anchor their thumb in a similar fashion, but move their index finger along the upper boundary of the watch, in an east-west or west-east direction. Therefore placing the wheel at the northeast corner of the watch was desirable.

Alternatively, for left-handed users we could build a mirror-image watch, where the watch could be reversed so that the roller wheel appears in the northwest corner instead of the northeast corner. However, in this case there would be two distinct types of watches: left-handed and right-handed ones.

External I/O: External input/output devices can interact with the watch over wireless standards such as IrDA and Bluetooth. For example, one could bring the watch near a Bluetooth connected keyboard, perhaps a Twiddler™, to enter information into the watch.

Voice recognition: The wrist watch computer includes a microphone and speaker, so that speech based interfaces are possible. We have found that current technology limits extensive use of this concept, although the capability can be improved using a wireless connection to an adjunct device with more computing power that can be used for doing part of the voice recognition.

4 Application and interaction design

The primary applications of a watch are related to time. Watches started with the ability to tell time. Then alarms were added to help people wake up. The next generation of smart watches will extend this concept by allowing people manage their time, provide personal information management (PIM) functions and also save time in their jobs by receiving just in time information on their wrists by leveraging short range wireless connectivity.

Before designing the applications for the watch, we spent a great deal of time trying to simplify the user interface by taking advantage of the high resolution display and the chosen input devices.

We started a study of existing devices which suggested we focus on navigation between functions. After some study, we settled on the following requirements for navigation between screens on the watch:

i. a quick return to the watch face from any application,
ii. a time-out to the watch face from any function,
iii. one touch deactivation of alarms,
iv. direct access to the main list of applications,
v. user programmable touch screen areas that could be used to access the user's most important applications,
vi. user ability to perform most of the common actions without lifting finger off the wheel,
vii. the ability to easily return to the previous screen. (Our studies indicated that people had become familiar with the browser model and the concepts of following hyperlinks and going back in the browser history stack. Therefore extending the concept of a browser back button to every watch face screen was desirable.).

We wanted the usage of the wrist watch computer to be obvious and avoid the need for a thick user manual for the watch. We started with human computer interaction (HCI) concepts from familiar computing environments such as web browsers, etc., and then employed a user-centered design process to tune the environment.

From our user study, we determined users were typically not aware that the screen was touch sensitive or that the roller could be pushed in to generate a selection. So some documentation to make the users aware of these features was useful.

With the basic input mechanisms on the wrist watch computer, and the four zone limitation we have placed on the touch screen, the following input options are available on the watch.

i. Press any of the four regions on the touch screen
ii. Roll the roller wheel clockwise
iii. Roll the roller wheel counterclockwise
iv. Push the roller wheel in the SW direction to select

In order to facilitate a quick return to the watch face, we decided to dedicate a touch on the top left corner to go back to watch face, irrespective of what the watch face is currently displaying. There is one exception to this rule however; if the watch face is already displaying the time a top-left tap takes the watch to the main-menu screen.

The top-left zone was chosen for this *return-to-home* function, since a right-handed user's right index finger (who wears his watch on the left hand) will almost completely cover the wrist watch computer display when this zone is being pressed. Obscuring the screen in this fashion is not desirable if the user needs to remember some information on the current screen. However, since a top-left tap results in a complete switch to a well-known screen the user is least concerned with the current contents of the display being obscured in this fashion.

In order to support a consistent page-back mechanism we decided to dedicate the bottom left zone to the *go-back* function - in this case 75% of the screen is visible for the right-handed user.

The two right zones are under the control of the application that currently has control of the screen. The application may decide to use them to go to the next page or previous page, or to invoke a list of menus for a hyperlink or an application, or for other purposes.

For left handed users the conventions of functions mapped to the left and right zones may be reversed in software. This is another advantage that results from the use of a touch screen.

The above conventions are suspended when an alarm sounds. In this case the alarm is quickly silenced by pressing any zone on the watch display.

By asking people we determined that many found it easier to set time on an analog watch using the watch stem than it was to set time on digital clocks with buttons. Since this design is for a high resolution display where the hands of the watch could be displayed in an analog fashion we use a circular metaphor and analog representations in our screens whenever desirable. The circular metaphor on the screen works well with the roller wheel.

Rolling the wheel also causes a scrolling action when textual content is displayed on the screen. Clicking the wheel causes a selection action similar to a mouse click. For instance in the case of setting time, rolling the wheel moves the hands and clicking the wheel sets the time to the current time displayed by the hands.

The availability of the back button provides a quick way to undo tasks or revert to previously computed screens and is very convenient and saves time in many instances. For example we found that some users were looking for a person's phone number and then would quickly want to check what time it was at the person's location and would go back to the clock, and in the meanwhile forget the phone number they just looked up. We solve situations like this by allowing the user to jump to the watch face with one touch and then get back to the previous screen with another touch.

The history stack maintained for the back button functionality does not preserve every screen; only a few canonical screens are preserved for any application. So for example it would be quite useless to preserve every screen update that was made during a stopwatch application that counted milli seconds.

We found that touch screens on watches are also activated accidentally by people trying to keep their watch screens clean. To address this, we use a variable time-out mechanism that returns the watch to the time screen from any screen after a time-out interval. Furthermore, the time-out period is initially set to a small value and increased gradually till it reaches an upper limit if the user is actively using the applications. For instance the time-out interval starts at 10 seconds. After the first interaction with the watch (tap on screen/activation of wheel) the watch will time out to the clock face in 10 seconds if there is no further interaction. If there is a second interaction within the 10 seconds the time-out is increased to 20 seconds and so on till it reaches say one minute. If at any point the watch times out to the clock face, the time-out interval is reset to 10 seconds. This mechanism ensures that accidental activations quickly revert the watch to the clock without annoying the user with spurious time-outs when the watch is being used actively.

5 Data Model

Since the wrist watch computer does not have a keyboard, textual content will most likely be created on other devices, such as PCs, and transferred to the watch. One interesting issue is how to represent this textual content on the wrist watch computer to make it most accessible to the user.

We chose to use Wireless Markup Language (WML) as the text format on the watch. Besides being an emerging standard that is expected to be widely accepted, WML has the additional desirable feature of being an XML document type and therefore supports the notion of hyperlinks, an intuitive and well accepted navigation model. We expect WML to be a standard format for publication of content for small form-factor devices like cellular phone, and hope to be able to use such content on the watch without change.

All textual information on the wrist watch computer such as list of appointments, todo lists, email messages, etc., are stored as WML cards. Typically such information will be resident on a PC or other servers. As part of the synchronization software that extracts this information and sends it into the wrist watch computer, the information will be transcoded into WML along with the appropriate hypertext links. For example, a calendar entry can be scanned for people's names and the names are turned into WML hyperlinks that point to the corresponding cards in the address book. In addition, each calendar entry is automatically linked with the previous and next entries to aid in the navigation.

When the watch displays WML content, rolling the wheel causes the current highlight to jump from hyperlink to hyperlink in the appropriate direction. Clicking the wheel in causes an activation of the hyperlink, i.e., navigation to the card pointed to by the hyperlink.

6 Applications

We surveyed prospective wrist watch computer users, who told us they would like to see watch functions, personal information management (PIM) applications (calendar, phone book, to dos), games, and an MP3 music player, on the wrist watch computer. Access to data from the web was cited as an important application as well. This included alerts, stock quotes, sports scores, email headers, headline news, weather and traffic rcports etc.

We have implcmented an prototype application environmcnt for thc wrist watch computer and have created several of the requested applications. The following sections discuss some of the abovc applications. With the power of such a device, one can easily imagine more applications, some highly personalized, some customized to games and trivia, etc.

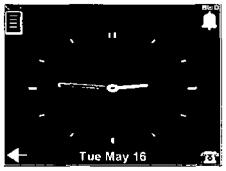

Figure 3: Clock Face

6.1 Clock face

The primary display on the wrist watch is a clock face showing current time and datc (see Figurc 3). This is the default screen to which we come back to when the return-to-home function is invoked by tapping on the top left zone on the touch panel.

On the clock face screen we decided that the top right touch zone should provide access to alarms, and the bottom right region should access the phone book since these are likely to be the most common applications. As discussed earlier, the top left zone in this screen will present the main menu or the application launch pad. Rolling the roller wheel will cycle through the days schedule. Pressing the roller wheel will provide access to the *quiet time* function (discussed below). This allocation allows one action access to key functions of the device - calendar, phone book, alarms, and turning off alarms.

6.2 Icon ring - application selector

The set of applications on a watch is expected to be quite small for any given person, perhaps of the order of a dozen or so. In keeping with the circular metaphor and using the roller wheel for most interactions, we chose to represent the application menu by means of icons that we position around a circle as shown in Figure 4.

The user can roll the wheel clockwise or counterclockwise to move the selection from one icon to another. Once the dcsired icon is sclected, the user clicks the roller wheel in to launch the application.

It is quite difficult to create culture-neutral, self-evident icons that unambiguously convey functional purpose. The small size and limited color depth of the wrist watch computer display makes this cven more difficult. It is generally quite difficult to design such icons unless one has color pixels and a reasonable physical area for the icons. As a result, it is important for functional icons to have captions that explain their function.

Users will initially rely on the caption, but gradually become familiar with the icons and use them directly. Since the screen dimensions are small, we cannot show all icons and their captions at the same time. Though icons can be small, the caption font has to be proportionally quite large to be legible. Therefore we chose to display just the caption for currently highlighted icon in the center of the icon ring. As the user rolls the wheel to move the highlight, the caption changes accordingly.

Based on user experiments, we restricted the number of icons shown on a screen to eight. If there are more than eight applications, we can place them on consecutive menu pages that one can access by touching the top right or bottom right zones of the touch screen to go to the previous page or next page of applications respectively.

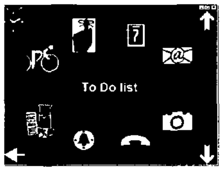

Figure 4: Icon Ring

Initial experience with this environment leads us to believe that users will be able to navigate to the right application page by tapping in corners of the wrist watch computer without looking at the display. For instance if a user has two application screens and knows that thc application of interest is in the second page a top left tap followed by a bottom right tap will get the user to the appropriate menu page.

Once in the right page, the user will have to look at the display while he selects the application icon by rolling the wheel and launches it by clicking the wheel. On any menu screen we put the most commonly used application in that screen as the default highlighted item. In this case, the user does not even have to look at the screen to launch his application, he just has to count the number of screen taps and follow it with a wheel click.

6.3 Alarms

An alarm can be associated with a specific time, a calendar entry, a to do entry, and to software agents either running on the watch or remotely.

Alarms associated with time can be set by adjusting the hour first and then adjusting the minute hand. Alarms can

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**