Experience With A Low Power Wireless Mobile Computing Platform

Vijay Raghunathan
Department of EE
University of California
Los Angeles, CA
vijay@ee.ucla.edu

Trevor Pering, Roy Want, Alex Nguyen
Ubiquity SRP
Intel Research
Santa Clara, CA
trevor.pering@intel.com

Peter Jensen
College of Computing
Georgia Institute of Technology
Atlanta, GA
peter@diff.net

ABSTRACT

A detailed power analysis of a multi-radio mobile platform highlights the complex tradeoffs between the computation, storage, and communication subsystems. The particular mobile device, which does not include an LCD or other on-board display, can be used as a source for audio or video media files, or a source/sink for secure data transfers. A version of the device has been augmented with fine-grained power monitoring capability and used to obtain detailed measurements of power dissipation in the various subsystems. Analysis of these measurements sheds light on the power consumption characteristics of different applications, thereby providing hints to system designers about potential areas for optimization. Specifically, this work contrasts the power efficiency of the various wireless technologies supported by the system.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: *design studies*

General Terms

Design, Experimentation, Measurement, Performance

Keywords

Mobile Computing, Wireless Systems, Power Management

1. INTRODUCTION

Mobile computing devices that are powered by a limited battery supply must be highly energy-efficient to maximize operation lifetime. The pressing need for reduced-energy solutions has spurred extensive research in low power design techniques [1, 2, 3], and has resulted in the development of power-frugal system components such as low power SDRAM [4] and processors such as the Intel® XScale™ [5] family (including the popular StrongARM). However, simply making the individual system components low power is not sufficient to support the battery lifetime requirements of emerging ubiquitous usage models. The entire system has to be designed and operated in a *power aware* manner, carefully orchestrating the transitions of the various components to/from their low power states.

This paper presents the design and architecture of the Consus Personal Server (PS) [6] wireless platform, a low power mobile computing device. The platform does not contain any integrated LCD; instead, personal content, such as audio, video, or other documents, is stored on the device and retrieved over the wireless link. The platform architecture includes several components found on traditional PDA devices, such as an embedded processor, memory, and a wireless communication subsystem: except the display. For communication, Consus possesses both Bluetooth (IEEE 802.15.1) and Wi-Fi (IEEE 802.11b) radios, allowing power-performance trade-offs depending on communication requirements. To support long-lasting mobile operation, the platform supports "always-on" operation, which allows the core system to sleep while still being reachable from the outside world.

The Consus platform has been augmented with fine-grained power measurement capabilities to better understand its power profile. Direct measurement of the power consumption of each system component helps identify the power dissipation hot spots in the system, which can then be the target of aggressive optimization. Specifically, the power consumption of the various radio subsystems sheds light on the suitability of specific wireless technologies for a given application, a valuable insight for system designers that helps them choose a system architecture that is best optimized for their needs. For example, Bluetooth is sufficient to supply the bandwidth necessary for high-quality audio playback, and therefore, the high power overhead of Wi-Fi sometimes represents an unnecessary power drain.

2. MOTIVATING APPLICATION

The Personal Server is a mobile device that enables users to readily store and access the data and applications they carry with them through interfaces found in the local environment. Instead of relying on the impoverished user interface found on most mobile devices, it utilizes nearby displays, keyboards, and other IO devices in lieu of ones on the local device. By shifting the interface burden from the small mobile device to the resource-rich environment around it, this platform aims to enable fundamentally new modes of mobile computing. Conceptually, such a device might contain a small display, or even be incorporated into a cell-phone; however, the initial prototype focuses on accessing stored content wirelessly, not allowing for a local display, focusing the research effort on this emerging usage model.

This interaction model addresses two major problems associated with mobile information access: the inherent difficulty of using small user interfaces on handheld devices, and the limited access to personal digital information afforded by public access points. The platform has the capability to be a complete storage and computation hub for the user by providing three major components: wireless communication, local computation, and

ISLPED'04, August 9-11, 2004, Newport Beach, California, USA.

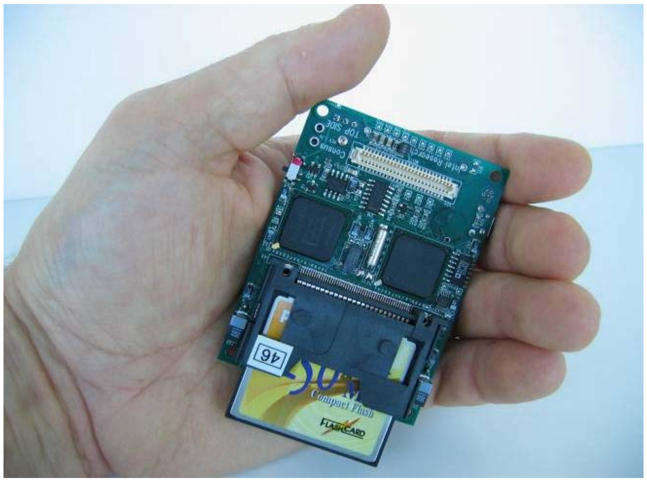

Figure 1. Mobile version of the Consus platform. The board is slightly smaller than a standard deck of cards.

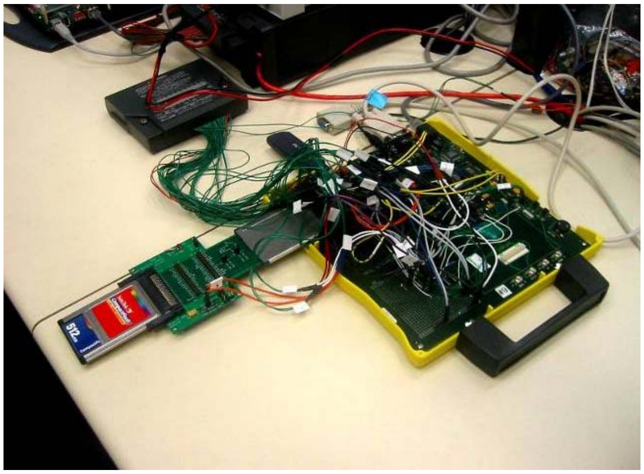

Figure 2. The enhanced Consus board for detailed component-level power measurement.

high-density storage. As a result, a mobile user can enjoy the benefits of a large display and full-sized keyboard (found in the environment) without having to carry a bulky computing platform (such as a laptop). Alternatively, the user could attempt to access their data through the network, placing them at the mercy of the often-unreliable network infrastructure. Further, network access is often complex to setup, due to corporate firewall regulations, *etc*. Fortunately, computing infrastructure is becoming well established in many of the places people wish to use computation. For example, available displays attached to PCs can be found in many homes, businesses, Internet cafes, and even public spaces such as airports and shopping centers. The Personal Server computing model goes beyond the traditional mobile context and strides the gap between the mobile- and ubiquitous- computing worlds, exploring the technologies needed for emerging wireless systems.

The wireless connectivity needed for the Personal Server model is short-ranged, where the user is in close proximity to the I/O device. This model encourages the use of low power radio technologies (such as Bluetooth) that do not necessarily have the full range of their high-power counterparts. In many situations, the range of Bluetooth will be perfectly sufficient: *e.g.,* if a user is standing right in front of a kiosk. Further, with proper antenna design and power amplifiers, the range of Bluetooth can be extended to cover a reasonably large area such as an entire café.

In order to provide an enriching user experience, the Personal Server device should appear to be "always on" – *i.e.,* one can contact it easily through an external display without having to manually turn the device on. This mode of operation is required to provide the seamless wireless experience that is crucial in a mobile context. Otherwise, users are likely to get frustrated due to the inconvenience of using the device and not use it at all. However, this requirement also places considerable strain on the limited power resources of the platform, especially the wireless subsystem. The measurement and optimization of these quiescent states is equally if not more important than the active states, since devices will often spend a great deal of time in these idle states.

3. RELATED WORK

The Itsy research platform designed at Compaq Research is similar to Consus although there are some significant differences. Architecturally, the Itsy has several common features with the Consus (similar StrongARM processor, SA-1111 coprocessor, *etc.*). However, the Consus does not have an LCD or any other display, since it was developed to pursue the ubiquitous computing Personal Server model. Furthermore, the power management features of the Itsy [12] are computation-centric, providing support for dynamic frequency and voltage scaling, but there are no wireless interfaces to manage. Instead, Consus focuses on power management techniques affecting the wireless subsystem, such as making use of hierarchical radios, wake-on-wireless, and passive motion detection. With most (if not all) of today's mobile computing devices employing wireless communication, and since the communication subsystem is often significantly more power hungry than the computation subsystem, focusing on communication-centric power management will significantly prolong battery life. Other devices such as the LART system [13] are similar to the Itsy in many respects.

Wake-on-wireless using multiple-radio systems was first explored in [14], a system that used a secondary highly-impoverished radio channel to control the operation of the primary, high-power, communication system. Our work focuses both more on a detailed power breakdown of the various sub-components and also highlights the use of a fully capable secondary radio channel for both discovery and data transfer operations.

4. SYSTEM ARCHITECTURE

The mobile version of the Consus platform, shown in Figure 1, consists of a small printed circuit board that implements the core functionality of the wireless platform. Additionally, there is a detachable daughter card that provides development support through Ethernet, USB master, JTAG, serial port, *etc*. An enhanced and larger form-factor version of the board, shown in Figure 2, encompasses the same system components but has been instrumented to allow fine-grained power measurement.

The system can broadly be partitioned into four subsystems: computation, communication, storage, and power. The computation subsystem consists of a StrongARM SA-1110 processor running at a nominal clock frequency of 132 MHz. Additionally, the board uses an SA-1111 companion chip to provide USB and Compact Flash (CF) support. The communication subsystem of Consus allows for both a USB

Bluetooth radio and a CF Wi-Fi card. The design incorporates an on-board Silicon Wave 1701/1750 Bluetooth module, which is a low power single-chip solution. The Bluetooth radio interfaces to the SA-1110 processor at the Host Controller Interface (HCI) level through a USB interface. Additionally, Consus has a PC-card slot that is used to attach a NetGear MA-701 Wi-Fi card or any other suitable CF card. The Consus memory subsystem consists of 32 MB of built in SDRAM, 32 MB of on-board flash, and supports additional high-density storage through an external CF memory card (currently available up to 4 GB). Finally, the power supply subsystem uses either an on-board Li-ion battery or an external 4.5V power supply that feeds energy-efficient voltage regulators.

The Consus runs the 2.4.19-rmk7 Linux OS kernel for the ARM processor architecture. It features a complete TCP/IP networking protocol stack, for both the Bluetooth and Wi-Fi subsystems, provided by the BlueZ and HostAP drivers, respectively. Many software packages have been ported to the platform, including a Java Virtual Machine, the Apache web-server, Darwin streaming media server, SSH encrypted communication, *etc.*, allowing the Personal Server to support a rich set of applications.

5. POWER MANAGEMENT TECHNIQUES

Power efficiency is a prime design consideration to enable Consus to provide always-on mobile operation. To address this issue, the platform supports component-level power-saving techniques, system-level communication trade-offs, and advanced wake-up mechanisms. Applying a combination of these techniques not only optimizes the power consumption in the various system-level power states, but also streamlines the transition between states. Transitions into and out of low power sleep states become more important in a truly mobile and ubiquitous usage model where the device always remains active throughout the day, responding to many different kinds of service requests (especially for applications that require location and context awareness). This is different from the standard laptop/PDA model where the device is switched off between episodes of intense use, requiring a cumbersome manual activation process.

5.1 Component Low Power Modes

Most of the components used on Consus support various low power states. The StrongARM SA-1110 has two low power modes, Idle and Sleep, which offer differing levels of power savings as well as state transition overheads. The SDRAM also supports a self-refresh mode, which can be used when the system as a whole is put to sleep. Similarly, the Bluetooth specification allows fine-grained control over scanning and sleep modes, and most Wi-Fi cards feature a shutdown mode with significantly decreased power consumption.

Not all wireless module vendors support the full range of power management capabilities. Therefore, the extent of power savings offered by the low power mode varies greatly depending on the specific wireless card or module: some radios consume a non-trivial amount of power even when in the shutdown mode. Furthermore, the exact card and drivers used for a removable device such as a CF Wi-Fi card are not necessarily under the system designer's control, increasing the variability of system behavior. One option to overcome this limitation is to implement module-level power gating, using which the system could completely cut off power to the module/component, thereby reducing the power consumption to virtually zero. The current platform does not yet support such coarse grain power gating; however, it is definitely desirable given the measured performance of commercial Wi-Fi and Bluetooth modules seen to date.

5.2 Always On Operation

In order to support the desired "always on" operation model, the system must be able to sleep while still being reachable (*i.e.,* appearing to be on) by the nearby environment. Always-on operation enables two usage models: first, the system is capable of always receiving wireless messages (*e.g.,* new email indication, incoming call in voice over IP based systems), and second, the user can readily access their personal content by connecting through a remote display. This always-on functionality is theoretically supported by both Bluetooth and Wi-Fi systems; however, currently only the Bluetooth subsystem on the Consus supports remote-wakeup capability. Datasheet specifications for other existing Bluetooth devices [7] indicate that the expected current consumption for a wake-on-wireless module could be as low as 3 mA, which is small even when compared to the overall power profile of the system in sleep mode. Similarly, recent Wi-Fi chipsets also provide support for a low power wake-on-wireless mode [8], albeit at higher power consumption.

Another way to manage power consumption is through motion detection using a passive mercury switch. Motion detection is useful in two main capacities: activating the device to scan for wireless networks, and as a hint for allowing an external connection. Since most Wi-Fi access points are fixed in the infrastructure, a mobile device does not need to scan for new ones unless it is moving, allowing the device to go to sleep, saving considerable power while motionless. Alternatively, users could intentionally nudge the device (*e.g.,* if it is in their bag) to wake it up when they wish to contact it remotely; although not completely "always on," this technique is far better than turning the device on using a switch or button. A mercury switch, which is either open or closed, depending on the position of a small bead of mercury between two contacts, can be used to detect motion without any active components by connecting it directly to a GPIO pin of the processor core. Normally an orientation-sensitive device, currently available small-bead mercury switches get activated by motion in any direction and can thus sense movement from any position. This mechanism has been implemented and effectively manages transitions between scanning and sleep states. It is also possible to implement a motion-detecting scheme using a micro-controller and accelerometers, increasing the complexity and power consumption, but providing more control over wake-up conditions using threshold, hysteresis, or activity recognition based schemes.

5.3 Hierarchical Radios

The multiple radios supported by the platform are useful for managing the power consumption of wireless discovery as well as supporting basic connectivity between heterogeneous devices and access points. The two different radios on Consus (Wi-Fi and Bluetooth) are significantly different in terms of their data transfer capabilities and power consumption profile. For example, while the Wi-Fi radio is more efficient for bulk data transfers (in terms of energy per bit communicated), it has a high quiescent power drain and state transition overhead. Bluetooth, on the other hand, is optimized for low power consumption but does not support the same high-bandwidth communication. By combining the two

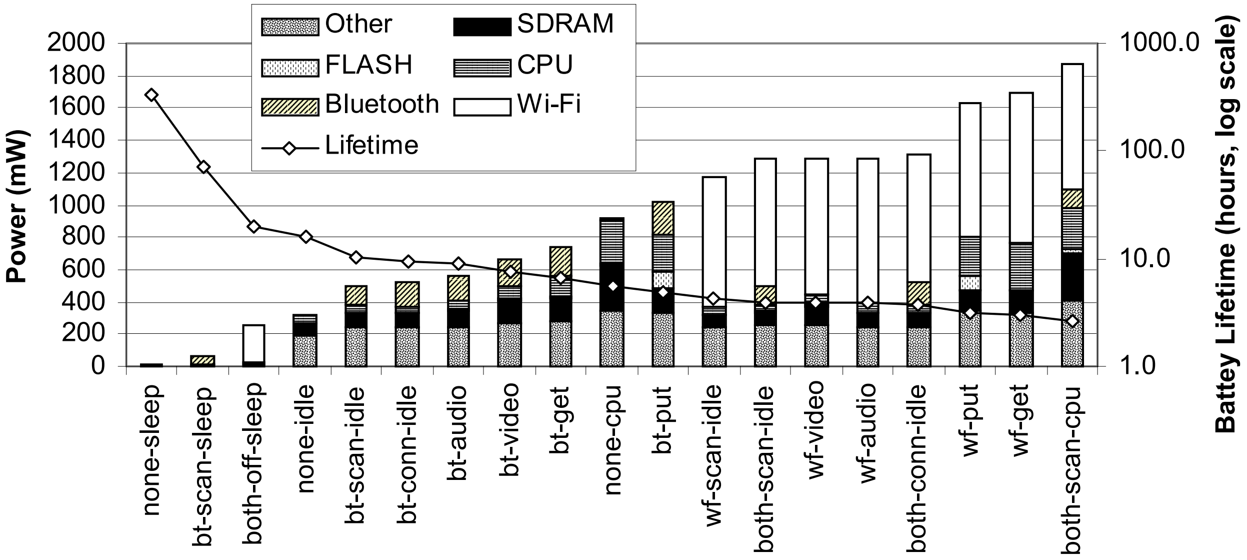

Figure 3. Power consumption overview, and estimated battery life (log scale).

properties of these radios, the system can support high-bandwidth capabilities with low quiescent power consumption.

Consus supports a hierarchical device discovery and connection establishment scheme that exploits the low power scan mode of Bluetooth and then transitions to a high-capacity Wi-Fi link. In this scheme, the Wi-Fi interface is completely shutdown, waiting for the appropriate connection trigger and configuration information to be passed through on the Bluetooth link. After this, the Wi-Fi interface is pre-configured and then powered on. Although not fully discussed in this paper, the motivation for this scheme is evident from the power profiles presented in Section 7.

This dual-radio hierarchical discovery process can be extended to include other radio technologies. Experiments have been conducted using the Mote radio platform [9] in the hierarchy, which is an extremely low power system designed primarily for wireless sensor networks. Although the Mote is inefficient for continuous data transfer, it is ideal for transmitting small packets advertising connection and configuration information for higher-level communication channels since it has a very low overhead for transitioning between active and sleep states. By extending the basic hierarchical radio concept all the way through the connected states (*e.g.*, dynamically transitioning between Wi-Fi and Bluetooth depending on instantaneous application requirements), the overall system power consumption can be further optimized.

6. FINE GRAINED POWER PROFILING

In order to develop effective power optimization strategies for the Consus platform, it was crucial to first determine the power breakdown of the various components. A special version of the Consus board, shown in Figure 2, was developed that was instrumented for fine-grained power analysis. This version is significantly larger and has low-tolerance 10 mΩ resistors placed in series with the power supply to various system components. Monitoring the voltage drop across these resistors, dividing it by the appropriate resistance, and multiplying it by the supply voltage, produces a detailed and accurate breakup of the power consumption. Power numbers are reported for six major categories, with the category “other” including the power (such as register losses, *etc.*) not attributed to other categories.

Measurements were conducted using a Fluke NetDAQ 1645 data acquisition system [10], which provides support for sampling up to 400 input channels in parallel, and is capable of a maximum sampling rate of 1000 samples per second. The NetDAQ system also comes with a visualization package, which enables easy control over the data gathering process. A post-processing program was used to convert the current and voltage samples into an aggregate power dissipation value.

7. EXPERIMENTAL RESULTS

The benchmarks used to profile the Consus have been grouped into subsets to explore the different aspects of system operation. Specifically, comparing the two supported wireless subsystems highlights the performance, power efficiency, and suitability of different wireless technologies for the various applications.

7.1 Benchmarks

Seven different benchmarks to measure the power consumption and performance of the system, exploring various quiescent modes of the processor and typical media-access capabilities:

- *sleep*: low power deep-sleep mode, where the processor is unable to perform any computation. Wake-up is achieved by either a timer event or external interrupt (*e.g.*, wake-on-wireless).
- *idle*: A low power mode that offers less power saving than the Sleep mode, but has a very low time to transition to Active state. The processor is awake, but since the system clock is frozen, it is not executing any instructions. All the I/O subsystems are fully functional.
- *CPU*: A synthetic micro-benchmark designed to exercise the processing and storage subsystems. This benchmark involves the processor performing a search of the local file system and does not use any communication.

- *put* & *get*: These benchmarks use the SSH program to continuously encrypt and transfer a 3 MB file to and from the device, respectively.
- *audio* & *video*: Streaming audio and video from the mobile device to an access point. These benchmarks use the Darwin Streaming Server [11] and standard MP4 encoding at 110 kbps and 410 kbps, respectively.

Each benchmark is run using one of several radio configurations:

- *none*: Both radios are disconnected from the system.
- *bt*: Only the Bluetooth radio is connected; the Wi-Fi radio is physically removed from the system.
- *wf*: Only the Wi-Fi radio is connected; the Bluetooth radio is physically removed from the system.
- *both*: Both radios are connected to the system.

Each radio, when physically connected to the system and not actively transferring data, is in one of the following states:

- *off*: The radio is in a software-controlled shutdown state.
- *scan:* The radio is scanning for nearby devices to connect to.
- *conn:* The radio has established a connection to a nearby device, but is not actively communicating data.

7.2 Measurements and Analysis

Figure 3 shows the complete range of benchmarks for the Consus platform, sorted in increasing order of power consumption. Additionally, it shows an estimated battery lifetime (log scale) from a 1 AH battery. It is easy to see the huge difference between the lowest-power sleep mode and the higher-power consuming benchmark. Additionally, it is apparent that the Wi-Fi radio consumes a disproportionate amount of power in several cases. Several conclusions can be drawn from this data, described below.

Figure 4 shows the impact of the system in the quiescent non-active states. When present, the power consumption of the Wi-Fi radio subsystem can be comparatively large, especially in the lower-power sleep states. Basically, the low power shutdown modes of the Wi-Fi radio are inadequate, arguing for coarse-grain power control to compensate for misbehaving drivers or hardware.

Figure 5 shows the power distribution of the system while using the Bluetooth radio. This kind of breakdown can be used to target further system optimizations. From this graph, it can be seen that the power consumption is fairly well balanced between the major components, implying that it is equally important to optimize across all the system components to achieve the maximum benefit (there is no single power-hog in the system).

Conversely, the power consumption of the various Wi-Fi states, shown in Figure 6, is almost completely dominated by the radio subsystem, which consumes more than 60% of the total power in some cases. This disparity argues that the use of Wi-Fi radios is possibly excessive for many wireless applications, unless the high bandwidth is absolutely necessary. Alternatively, this distribution suggests that the Wi-Fi radio subsystem is a prime candidate for optimization, either through specific power management techniques, or integrated system operation (such as coordinating with the Bluetooth radio subsystem).

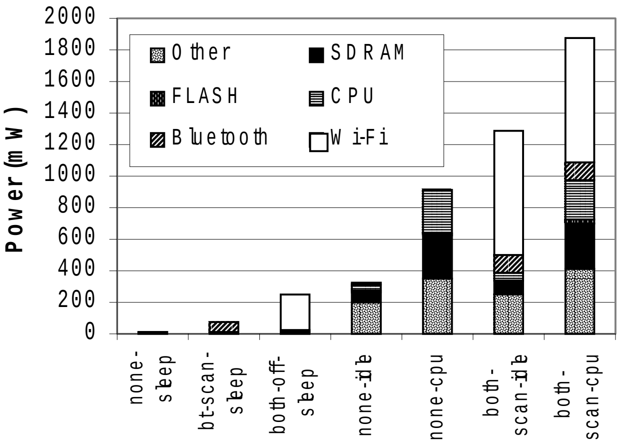

Figure 4. Quiescent state power consumption.

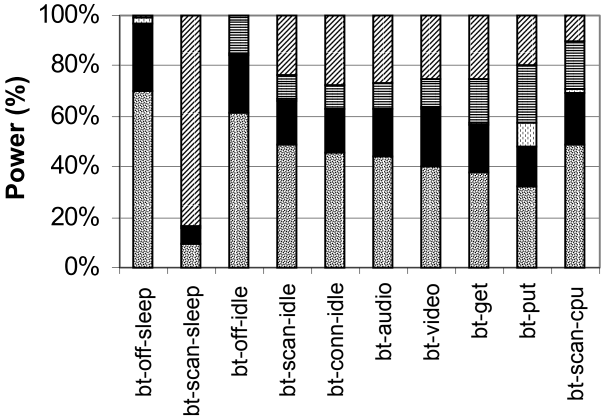

Figure 5. Bluetooth power breakdown.

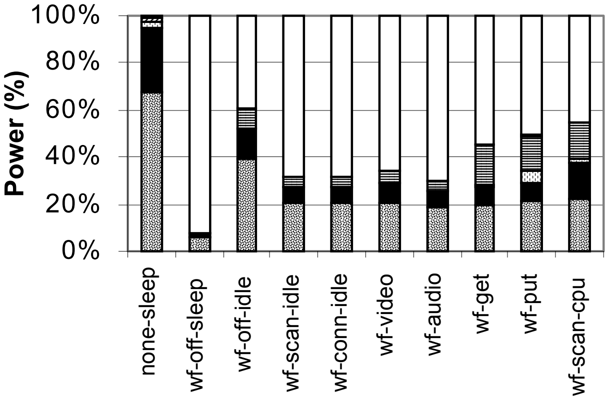

Figure 6. Wi-Fi power breakdown.

The *get* and *put* benchmarks, shown in Figure 7, explore the power consumption for transferring data from and to the device. As expected, Wi-Fi has higher bandwidth and power consumption for the *get* benchmark. However, the *put* benchmark is write-limited, owing to the slow write speed of the on-board flash memory. Therefore, the increased bandwidth of Wi-Fi is almost completely wasted, resulting in a much higher energy/bit ratio for Wi-Fi *put*.

The transfer profile of the streaming benchmarks, shown in Figure 8, resembles that of the *put* benchmark because they are limited by the data rate of the media stream. Specifically, neither high-quality mp3 audio nor moderate-sized movies are sufficient to saturate the Bluetooth radio channel; however, if higher quality video were required, then Wi-Fi would be an appropriate choice.

The power and throughput of the various active wireless transfer modes is summarized in Figure 9, highlighting the different operating zones for the two radios. The area of each bubble indicates the effective energy/bit achieved by one benchmark and radio. As is evident, Bluetooth is often the low power alternative, except when high throughput is required. Further, although Wi-Fi can potentially have a much lower energy/bit, the limited bandwidth requirement of several applications favors Bluetooth.

8. CONCLUSIONS

This paper presented the design and power profiling of the Consus wireless platform, a personal data repository. Consus was built to investigate the Personal Server concept, which explores emerging models of mobile and ubiquitous computing. Several of the novel wireless power management features that are present on Consus support the notion of "always on" computing. The detailed power profile of Consus helps identify the largest power consumers in the system, and provide valuable insights into the power efficiency of several popular wireless technologies in use today. Specifically, Bluetooth offers the more power efficient alternative for several streaming media applications, and Wi-Fi becomes the technology of choice only if extremely high bandwidth is required. Further, for applications that involve writing to flash memory, the data transfer rate is often limited by the flash write speed, negating any benefits of a high-bandwidth wireless link. These observations can directly impact commercially available systems, such as the HP iPAQ, which already support multiple wireless options.

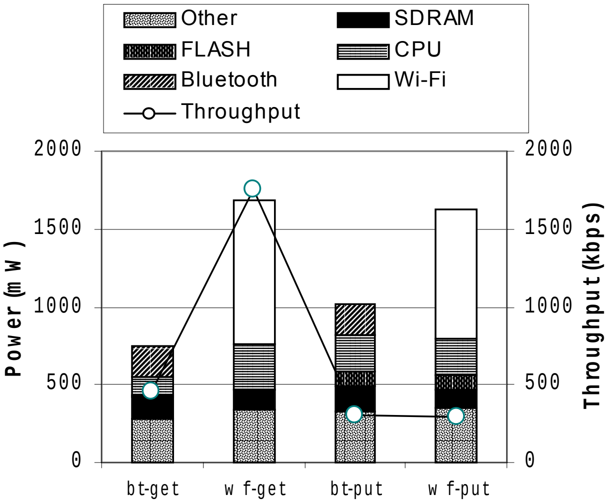

Figure 7. Data transfer benchmarks.

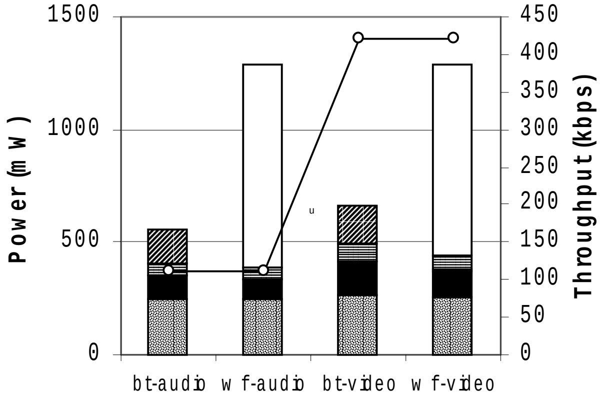

Figure 8. Streaming media benchmarks.

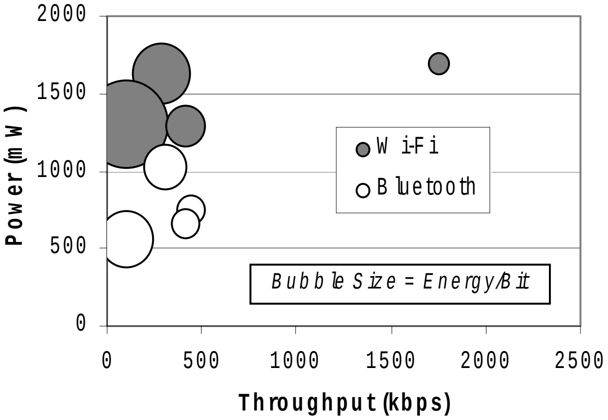

Figure 9. Power/Throughput comparison of Wi-Fi and Bluetooth for the get/put/audio/video benchmarks.

9. REFERENCES

[1] A. P. Chandrakasan and R. W. Brodersen, *Low Power CMOS Digital Design*. Kluwer Academic Publishers, Norwell, 1996.

[2] A. Raghunathan, N. K. Jha, and S. Dey, *High-level Power Analysis and Optimization*. Kluwer Academic Publishers, Norwell, MA, 1998.

[3] L. Benini and G. De Micheli, *Dynamic Power Management: Design Techniques and CAD Tools*. Kluwer Academic Publishers, Norwell, MA, 1997.

[4] Mobile RAM products, Infineon Tech. Inc.

[5] Intel XScale micro architecture (http://www.intel.com/design/xscale).

[6] R. Want, *et al.,* "The Personal Server: Changing the way we think about Ubiquitous Computing", Proc. UBICOMP, 2002.

[7] TC2000 Bluetooth IC, Zeevo Inc.

[8] AR5004 WLAN chipset, Atheros Communications Inc.

[9] J. Hill, *et al.*, "System architecture directions for networked sensors", Proc. of ACM ASPLOS, 2000.

[10] NetDAQ 1445 data acquisition system, Fluke Inc.

[11] Darwin Streaming Server, Streaming Media Inc.

[12] M. A. Viredaz, *et al.,* "Power evaluation of a handheld computer", IEEE Micro, vol. 23, iss. 1, pp. 66-74, Jan 2003.

[13] The LART embedded computer (http://www.lart.tudelft.nl/)

[14] E. Shih, P. Bahl, and M. Sinclair, "Wake on Wireless: an event driven power management strategy", Proc. ACM MOBICOM, pp. 160-171, 2002.