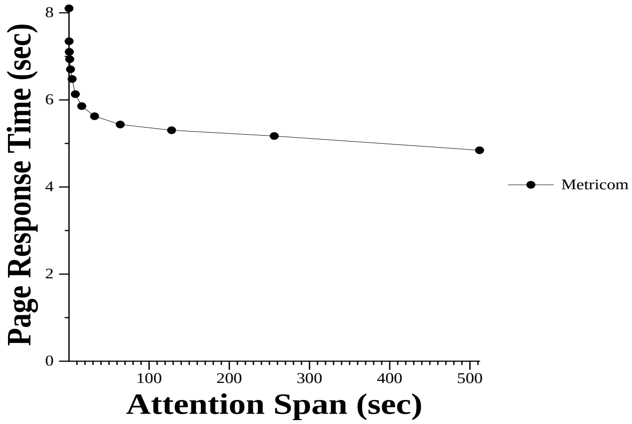

Figure 9. Response time as a function of Attention Span for Metricom

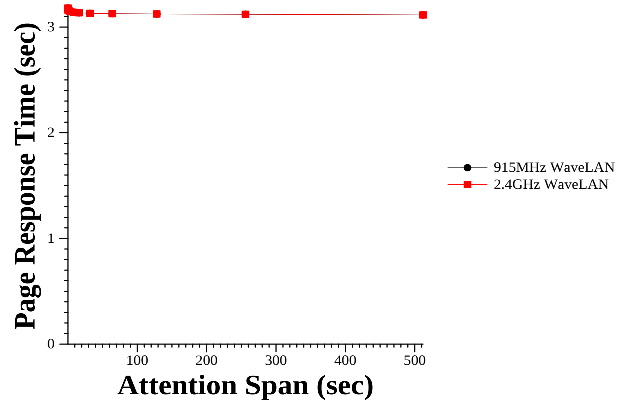

Figure 10. Response time as a function of Attention Span for the Wavelan NIs

(such as TCP's backoff in the presence of wireless losses) wastes power. Even when the protocol is performing correctly, the limited bandwidth of current wireless networks may be the problem; a base station that divides 2 Mb equally among 10 mobiles causes each of them to consume 10 times as much power (100 times as much power total!) as a base station that uses a TDMA scheme to coordinate delivery of data to receivers. As we have learned, the dominant cost comes from an idle network interface, not from receiving data at a higher or lower rate. Even in applications that display real time data such as vic and vat, the receiver may only be able to display data at a rate that is slower than the wireless bandwidth. In these cases, it may be more useful to send short bursts of data at the full wireless bandwidth rather than send at the rate that the receiver can display because the network interface is consuming valuable energy sitting idle while the information is being displayed. The valuable lesson is that network interfaces can consume a significant fraction of the power budget of PDAs, and requires smart software and applications to make sure that battery lifetime is not needlessly shortened.

7 Acknowledgments

Thanks go to Bruce Mah from UC Berkeley for providing us with the WWW traces. This work is supported by DARPA contract DAAB07-95-C-D154 and grants from the California MICRO Program, Hughes Aircraft Corporation, Metricom, and AT&T.

References

[BSAK95] Hari Balakrishnan, Srinivasan Seshan, Elan Amir, and Randy Katz. Improving TCP/IP performance over wireless networks. In *Proceedings MOBICOM 1995*, pages 2–11, San Franscisco, California, November 1995.

[DKM94] Fred Douglis, P. Krishnan, and Brian Marsh. Thwarting the power-hungry disk. In *Proceedings 1994 Winter USENIX Symposium*, 1994.

[DKM+94] Fred Douglis, Frans Kaashoek, Brian Marsh, Ramon Caceres, Kai Li, and Joshua Tauber. Storage alternatives for mobile computers. In *Proc. 1994 Symposium on Operating Systems Design and Implementation*. OSDI, November 1994.

[DM93] Fred Douglis and Brian Marsh. Low power disk management for mobile computers. Technical Report MITL-TR-53-93, Matsushita Information Technology Laboratory, April 1993.

[IVB94] T. Imielinski, S. Viswanathan, , and B.R. Badrinath. Energy efficient indexing on air. In *Proceedings ACM SIGMOD*, 1994.

[JW95] Ravi Jain and John Werth. Airdisks and airraid:modeling and scheduling periodic wireless data broadcast. *Computer Architecture News*, 23(4):23–28, September 1995.

[LKH+94] Kester Li, Roger Kumpf, Paul Horton, , and Tom Anderson. A quantative analysis of disk drive power management in portable computers. In *Proceedings 1994 Winter USENIX*, pages 279–291, San Franscisco, California, Winter 1994.

[Mar93] Brian Marsh. Systems issues for mobile computing. Technical Report MITL-TR-50-93, Matsushita Information Technology Laboratory, Feburary 1993.

[ZFAA94] Stanley Zdonik, Michael Franklin, Rafael Alonso, and Swarup Acharya. Are "Disks in the Air" just pie in the sky? In *IEEE Workshop on Mobile Computing Systems and Applications*, pages 12–19, Santa Cruz, California, December 1994.

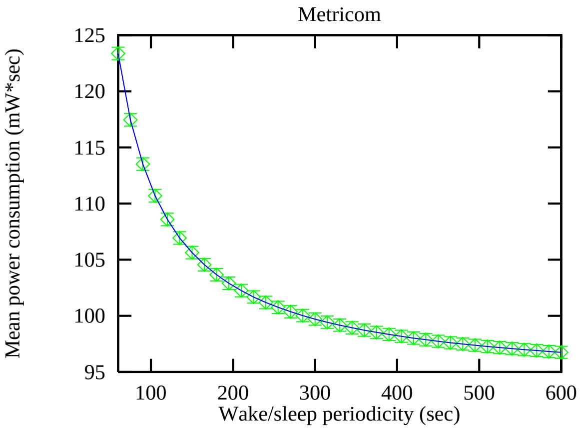

Figure 7. Energy as a function of attention span for Email.

from the interface. For our measurements, we specified a maximum attention span of 5 minutes, after which we considered the machine to have been turned off.

5.3 Simulation Details

5.3.1 Inputs and Outputs

The web simulation uses the parameters measured and derived in Section 2 and Section 3 as well as the traces described above.

The outputs of the simulation are two metrics of performance:

1 Average energy cost, in milliwatt-seconds, of an HTTP page retrieval.

2 The average latency for the initiation of a Web page access. This measures the average amount of time to complete the first HTTP request of a work phase (with the assumption that Web page accesses are a single html document followed by a number of inline images)

5.4 Simulation Results

5.4.1 Energy Per Page

Figure 8 shows the simulation results. (The full paper has results for all of the network interfaces). The results show that the power consumption can be decreased significantly. One interesting feature to notice is that it actually costs less on the Metricom to simply remain on at all times than to use some of the longer (i.e., greater than 64 seconds) attention spans because of the initial wakeup/idle cost.

Figure 9 and Figure 10 illustrate the benefits of a fast transition from the sleeping state to the idle state. The Wavelan's faster sleep to idle transition allows it to use shorter attention

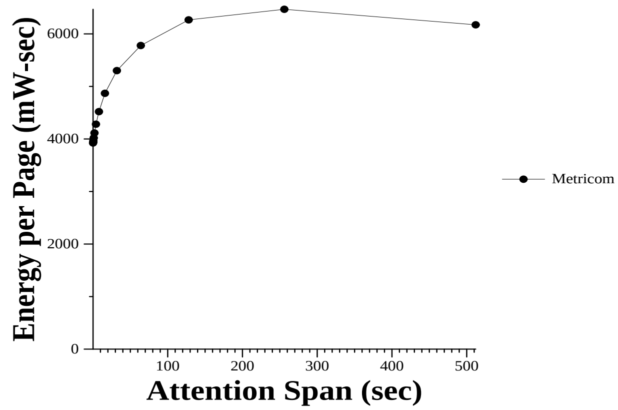

Figure 8. Energy per page as a function of Attention Span for Metricom

spans with virtually no impact on latency, outweighing the higher idle power drain.

6 Conclusions/Recommendations

Our measurements of PDA and Network Interface power consumptions show that Network Interfaces consume a significant fraction of the total power on a PDA, and that the power consumed when the interface is on and idle is more than the cost of receiving packets. For some interfaces, the cost of sending packets may be significant when compared to the cost of being idle, but the applications we examined involved large transfers of data to rather than from the mobile device.

Although the choice of transport layer can have a significant impact on the number of packets sent and received by the mobile device, the actual power difference is minimal. This is because the power consumed simply by keeping the network interface on during the transfer contributes the most to the final energy cost. In the presence of a high packet error rate, however, some schemes may overreact to packet losses, mistaking them for congestion. This slows down the transfer rate, which increases the amount of time that the transfer takes and the amount of power consumption by the network interface.

Simulations show that for email, our approach can lead to a significant decrease in power reduction with a minimal amount of ``staleness''. For web browsing, the Wavelan's quick sleep to idle transition allow very aggressive management of the network interface and significant power savings at a minimum of user-visible latency.

6.1 Recommendations for Future Networks Interfaces and Protocols

The current generation of protocols also may need some tuning to minimize the power cost of network interfaces. Any protocol that leaves a mobile receiver idle unnecessarily

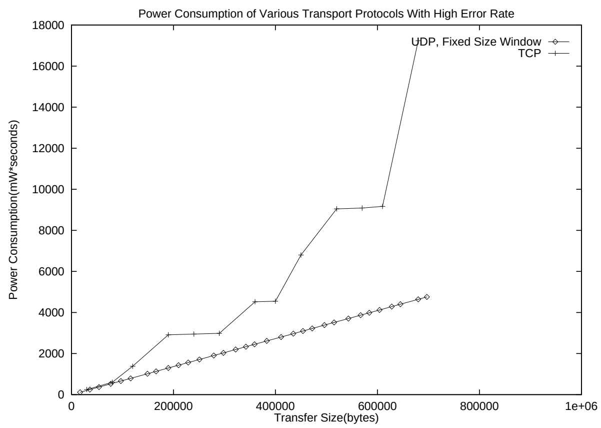

Figure 5. The Effect of Wireless Losses on Energy Consumption.

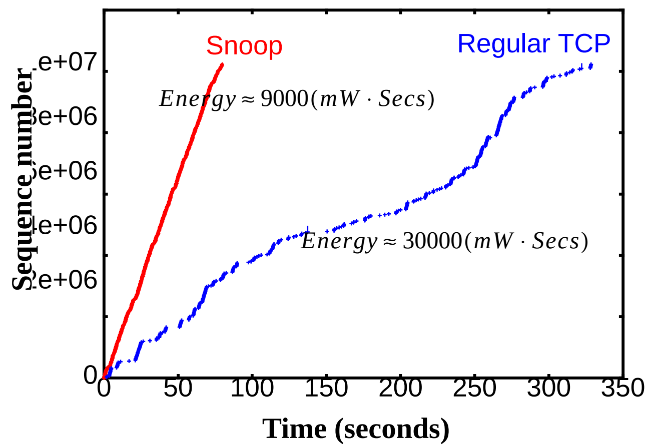

Figure 6. Detailed Example of How Wireless Losses Affect Energy Consumption.

3.4 The Effect of Error Rate on Energy

Section 3.3 indicates that *Idle* makes the greatest contribution to final energy cost, and that for low error rates, the different transport protocols behave similarly. Figure 5 shows the effect of a higher error rate on energy consumption. In the presence of a high packet error rate, the difference is more significant. As shown in [BSAK95], TCP mistakes packet losses for congestion and reduces the transmission rate. From a power standpoint, this decreases the value of B and increases the total energy cost. The UDP schemes we used, however, do not back off as a result of wireless losses. A more detailed example is shown in Figure 6 of sequence number versus time, where a transfer with an increased error rate increases the time of the TCP transfer, reduces the value of B, and leads to a dramatically higher power cost when compared to a transfer using the Snoop protocol, which handles with wireless losses more intelligently.

4 Mail Simulation

4.1 Data Collection

We used the user population of the Computer Science Division at UC-Berkeley to measure mail activity. The arrival times and sizes of mail messages appearing in the Division mail spool was collected.

4.2 A strategy for reducing power consumption

Our power control scheme depends on the fact that the PDA can wake up periodically, bringing its NI from a sleep to idle state and check for new mail. Similar to approaches in [IVB94], [JW95], and [ZFAA94], the availability of new mail is broadcast periodically so the PDA does not have to generate any new messages. Alternatively, the PDA can check for mail itself and download any that has arrived.

4.3 Simulation Results

PDA waits before waking up and checking for new mail. We ran the simulation for attention spans ranging from 60 seconds (1 minute), to 600 seconds (10 minutes), in 15 second increments and measured the average power consumption for the user population and the average lag between the time the mail message enters the mail spool and the time that the PDA notices that the message has arrived.

Figure 7 shows the average power consumption as a function of the attention span. Although not shown here, the ``staleness'' (the time the message arrives and the user is notified) increases as a function of the attention span. (The other networks behaved similarly). The results are quite promising; with an approximate staleness of two minutes, the power consumption drops by 20%. Note that virtually all of the power drain in this case is due to the interface being on. When compared to the alternative scheme of leaving the NI on all of the time, this approach achieves considerable power savings.

5 Web Access Simulation

5.1 Trace Collection and Processing

We used traces of HTTP traffic at UC Berkeley as input to a simulator which experimented with different power savings strategies. For each workstation, we kept track of the start times and transfer sizes for each outstanding HTTP connection. For each user, we divided time into *work* (when at least 1 outstanding connection was outstanding) and *think* (when no connections are active) phases. These post-processed traces form the input to the simulation.

5.2 Power Saving Strategy

The power saving strategy evaluated in this section attempts to reduce effective power consumption during the think time portions of the traces. We turn off the network interface after the user has been in a think phase for more than a certain amount of time (called the *attention span)*. It stays in that state until the user sends data (in this case, a HTTP request)

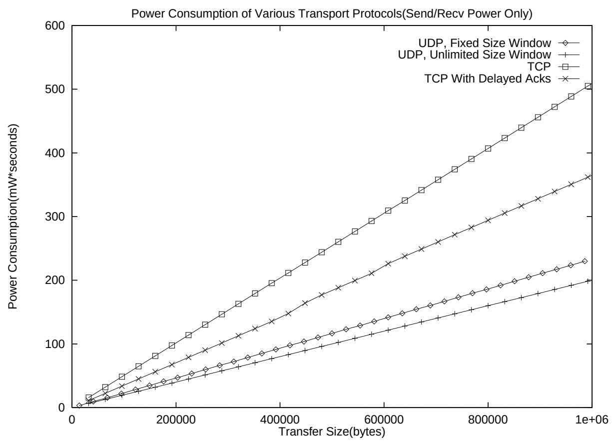

Figure 3. Energy for Different Protocols only including *SendRecv*

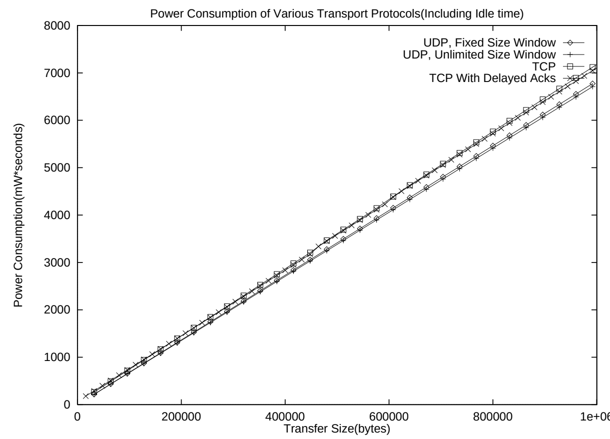

Figure 4. Energy for Different Protocols Including *SendRecv* and *Idle*

2. Sending packet costs more than receiving and can be significant when compared to the cost of being idle, but only if the mobile is sending large amounts of data to the wired network.

3 Transport Layer Simulation

3.1 Breakdown of Transport-layer Power Consumption

We can break down the energy consumed to complete a bulk transfer of b bytes as follows for a fixed data packet size and a fixed acknowledgment size:

$$Energy = SendRecv \times Idle$$

$$SendRecv = aE_a + dE_d$$

$$Idle = I\frac{b}{B}$$

Where a is the number of acknowledgments sent, E_a is the energy cost to send a single acknowledgment, d is the number of data packets sent, E_d is the energy cost to send a single data packet, I is the instantaneous idle power, and B is the effective bandwidth of the transfer.

We compared four different transport layer protocols in terms of the number of acknowledgment packets they generate, the number of packets that they send to the mobile device, and the amount of time necessary to accomplish the transfer. These were:

1. TCP Reno: This protocol generates an acknowledgment for every data packet sent.

2. TCP Reno with delayed acknowledgments: During a continuous transfer, this protocol generates an acknowledgment for every other data packet.

3. Reliable UDP, fixed-size window: This protocol uses a fixed size flow control window of size *w*. We used a window size of 10. This will send on average a little more than one acknowledgment for each **w** packets.

4. Reliable UDP, unlimited window: This is a special case of the above UDP scheme when the flow control window is larger than the number of packets sent.

Note that the TCP schemes use Go-Back-N acknowledgments, which will lead to larger values of a and d than the UDP schemes, which use selective acknowledgments.

3.2 Methodology

The scenario we used was a three node network including a source, base station, and receiver. The source and base station were connected with a high bandwidth, low error rate link, and the base station and mobile were connected with a lower bandwidth, higher error rate link. We simulated the TCP protocols using the Network Simulator (ns) from the networking group at Lawrence Berkeley Laboratory. We simulated the Reliable UDP schemes by deriving formulas that showed the number of packets sent and received for a given bulk transfer size and packet error rate. To compare the protocols, we kept track of the total length of the transfer and the values of a and d. We then used the formulas extracted from the data in Figure 1 and Figure 2 to generate the power drain for each packet sent and received as well as a power drain for the entire transfer.

3.3 Simulation Results

Figure 3 shows the contribution that *SendRecv* makes to the energy cost for a variety of transfer sizes for the 915 Mhz Wavelan. The **x** axis shows the transfer size, and the **y** axis shows the energy cost in mW-seconds. (The other networks behaved similarly). These results show that protocols that send fewer acknowledgments use less power. When the contributions from *SendRecv* and *Idle* to the total energy cost are included, however, (see Figure 4), the total transfer time comes into play. Here, the idle cost dominates the cost to send or receive packets.

Device	Sleep (mW)	Idle (mW)	Wakeup (mW)	Turn-on Time
Wavelan (2.4 Ghz)	177.3284	1318.857	N/A	100ms
Wavelan (915 Mhz)	143.0023	1148.601	N/A	100ms
Metricom	93.50762	346.984	N/A	5 sec
Infrared	N/A	349.607	431.034	100ms
Newton PDA	164.1871	1187.75	N/A	N/A
Magic Link PDA	312.03	700	N/A	N/A
Laptop	N/A	8000	N/A	N/A

TABLE 1. The Steady State Power Consumption for the Network Devices and PDAs

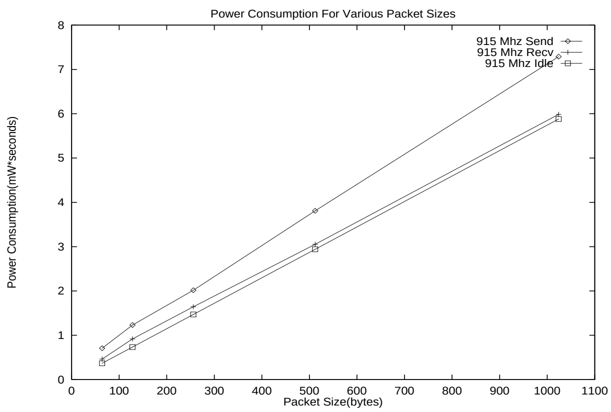

Figure 1. Energy as a function of Packet Size for the 915 Mhz Wavelan NI

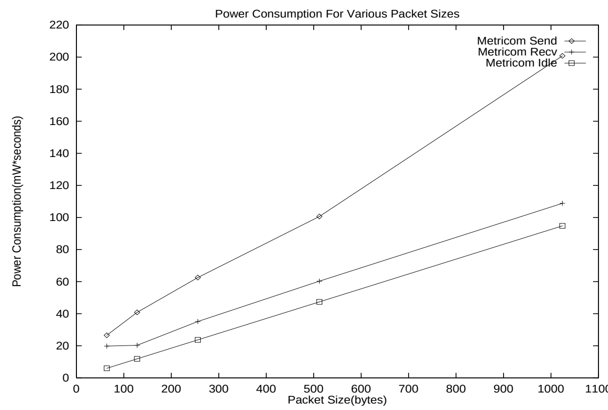

Figure 2. Energy as a Function of Packet Size for the Metricom NI

neous operations such as packet transmission and reception, we made several measurements and averaged these together to get the final energy cost. The digital oscilloscope produced bitmaps of the instantaneous voltage across the series resistor over time. We post-processed the bitmaps to obtain the area under the curve and to determine average voltage drops over the time-frame.

2.2 Measurement Results

Table 1 shows the average power consumption of the two PDAs and the Network Interfaces. The Metricom modem has a Wakeup state; when the Metricom modem turns on, it registers with the network and consumes more power for approximately the first minute seconds of activity. Note that in all cases, although the power consumed by the Network Interface is not a significant part of the power drain for a laptop computer, it is comparable to (or even more than) the power consumed by the PDA. This is a clear indication that power management of the Network Interface is essential.

Figure 1 and Figure 2 show the power consumption for sending and receiving as a function of packet size for the 915 Mhz Wavelan and Metricom device, respectively. (The full paper has results for all of the network interfaces). The **x**-axis shows the packet size, and the **y**-axis shows the energy consumption in milliwatt-seconds. From these measurements, we calculated best-fit linear models for the transport-layer simulations presented in Section 3. One obvious feature in the graph is that receiving packets only costs marginally more energy than simply being idle. Also note that although the cost of sending packets scales with the size of the packet, the cost of being idle for the same amount of time also scales linearly with the packet size. For the Metricom device, the cost of sending is approximately double that of being idle for the same amount of time. For the power consumption to be constantly double that of the idle state, however, the device would have to be sending packets continuously. In any situation where the device is mainly receiving packets (for example, the applications of Section 4 and Section 5), the amount of time that the NI spends sending short acknowledgments is outweighed by the time spent receiving data packets. We believe that these applications or similar ones where the PDA is retrieving rather than sending large amounts of data will be the most common applications on future PDAs.

From these measurements we may conclude that:

1. Receiving packets only costs slightly more than the idle cost.

Reducing Power Consumption of Network Interfaces in Hand-Held Devices (Extended Abstract)

Mark Stemm, Paul Gauthier, Daishi Harada, Randy H. Katz
{stemm,gauthier,daishi,randy}@CS.Berkeley.EDU
Computer Science Division, Department of Electrical Engineering and Computer Science, University of California at Berkeley, Berkeley, CA 94720-1776.

Abstract

An important issue to be addressed for the next generation of wirelessly-connected hand-held devices is battery longevity. In this paper we examine this issue from the point of view of the Network Interface (NI). In particular, we measure the power usage of two PDAs, the Apple Newton Messagepad and Sony Magic Link, and four NIs, the Metricom Ricochet Wireless Modem, the AT&T Wavelan operating at 915 MHz and 2.4 GHz, and the IBM Infrared Wireless LAN Adapter. These measurements clearly indicate that the power drained by the network interface constitutes a large fraction of the total power used by the PDA. We also conduct trace-driven simulation experiments and show that by using application-specific policies it is possible to achieve considerable power savings. We also examine the power drain for different transport level protocols, and find that the critical factor is not the number of packets sent or received but the amount of time that the NI is in an active but idle state.

1 . Introduction

Hand-held devices coupled with wireless network interfaces are emerging as a new way to retrieve information. Tighter cost, weight, and size constraints may make minimizing power consumption even more important for PDAs than for laptop computers. The difficulty lies in the fact that achieving continuous information access requires the addition of a large power consumer: a wireless network interface (NI). For example, the Network Interfaces we measured consumed from 350mW to 1300mW when idle. At these power drains, current network interfaces consume nearly as much as a PDA which is on but inactive (Table 1). Although much work has been done in reducing the power consumption of other peripheral devices such as disks, [DM93] [DKM94] [DKM+94] [LKH+94] [Li94] [Mar93], little work has been done to reduce NI power consumption.

This paper presents detailed measurements of these network interfaces in order to determine the power consumption while in its sleep, idle, send and receive states. We also examine different choices of transport layer protocols, using simulations to examine the relative power trade-offs when sending equal amounts of data from a wired sender to a mobile receiver. Using this information, we then investigate the trade-offs made by various power savings strategies for real applications. We focus on two applications that we expect to be the ``killer apps'' for PDAs: electronic mail and WWW access. We do this by collecting traces of these applications and applying them to simulations where we try alternative (application-specific) policies and determine the resulting power savings and user-visible latencies. Results show that significant power savings can be made with a minimum of user-perceivable latency.

2 Measurements

We measured two PDAs: the Apple Newton Messagepad 100, and the Sony Magic Link (PIC 1000). We also measured four network interfaces (NIs): AT&T's Wavelan PCMCIA card operating at 915 MHz and 2.4 GHz, Metricom's Ricochet Wireless Modem, and IBM's Infrared Wireless LAN card. We measured the PDAs while performing various specific tasks and then averaged the measurements to obtain ``typical'' behavior. We measured the network interfaces while idling, sleeping, and sending and receiving packets of various sizes.

2.1 Methodology

To measure power consumption for steady state behaviors, we required both current and voltage measurements. We used a digital oscilloscope to measure the voltage and current draw of the various devices. The current draw was actually measured by using a small resistor and measuring the voltage drop across the resistor. For the PDAs and Ricochet modem (which has its own external battery), we measured the voltage and current at the battery terminals. (The Metricom NI will soon have a PCMCIA version available, however, so this power drain will soon affect the PDA lifetime directly). For the PCMCIA Network Interfaces, we measured at the power pins coming into the card. For instanta-