Design of Wireless Portable Systems

E. Brewer, T. Burd, F. Burghardt, A. Burstein, R. Doering,
K. Lutz, S. Narayansaramy, T. Pering, B. Richards, T. Truman,
R. Katz, J. Rabaey, R. Brodersen

*University of California at Berkeley**

The InfoPad project explores the infrastructure and devices required for portable wireless access to the national information infrastructure. The InfoPad model emphasizes high-bandwidth wireless connectivity and moves the computing power of the portable device into the backbone network, where we can provide not only full internet access, but increased computing power as well. By concentrating on I/O for the pad we reduce its cost, weight and power requirements, and increase the effective bandwidth through the greater error tolerance of I/O traffic such as video.

We describe the InfoPad model, its infrastructure, and the results of the first prototype. This prototype proved the feasibility of the basic model and developed key technologies such as low-power design methodology and protocols for wireless connections. We also discuss the next generation InfoPad and our future plans.

1 Introduction

The near future will bring the fusion of four rapidly emerging technologies: high-speed networking and associated services, wireless communications, integrated circuit technology, and multimedia-based applications. Of particular importance in the networking area is the expansion of internet activity which is now taking place. For example, in March 1994 the data traffic over the NSF backbone in the U.S. increased over 20%, from 9 to 11 terabytes, providing new network services of all types to over 20 million users. It is expected that exponential growth will continue as awareness and access to the internet increase. The primary goal of the InfoPad project is to support ubiquitous access to these evolving network services through a wireless, multimedia terminal as shown in Figure 1. As a way of defining the capabilities that must be supported, the we examine the characteristics of present and future services.

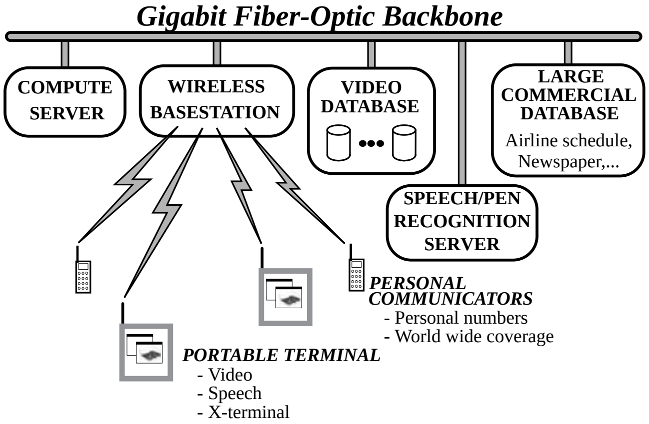


Figure 1: Future infrastructure for information access

Text/graphics Databases: Servers presently provide access to textual and graphic commercial databases that contain a wide spectrum of information, including international and domestic news, weather, financial information, stock pricing, traffic data, transportation schedules, and electronic and voice mail. This will expand in the future to provide educational information that includes on-line textbooks and libraries, health-care information, and databases that support electronic commerce. A key characteristic of this data is that the value of the information is time critical because of its transitory nature, which renders distribution by off-line means such as CD-ROM impractical. Even for relatively constant information, sufficiently large databases, such as digital libraries, will require on-line access.

Video Databases: Future applications will have video databases containing both entertainment and educational media, including lectures, movies, and news clips. Unlike today's television broadcasts, such video would be available in a per-user, on-demand basis. Video will necessarily be stored in a compressed format, for minimization of both storage space and transmission bandwidth, thus requiring that the wireless terminal support real-time video decompression.

Compute Servers: Though it is expected that the primary focus of the network services will be information access, entry and manipulation, there will be also be need for computational support. The placement of this computation is expected to move from the portable unit, where power consumption and cost are an issue, to the backbone network where the computation can be performed in either distributed or centralized compute servers. Client-server computing environments, such as those based on the X-window system have demonstrated that computation need not necessarily be done on a local machine (the display server) that a user is operating; but can be done by programs executing on many remote machines (clients), that simply issue graphic commands to the server to display their results. Many such inexpensive *X terminals* already exist.

Advanced User-Interface Technology: The design of an effective user interface to access the future services is a critical issue, since the general public will require signifi-

---

*: Contact: Eric Brewer, brewer@cs.berkeley.edu. The InfoPad project is supported in part by ARPA Contract J-FBI-93-153.

cantly simpler access mechanisms to a vastly expanded array of services over what is available now. Using audio and pen-based input, supported by speech and handwriting recognition, offers the possibility of considerable improvements over the keyboard and mouse mechanisms. The recognition can be performed by processors on the network, which enables significantly more powerful recognition engines. Context-specific vocabularies will yield further improvements in recognition accuracy.

User Scenarios: The *general public at home* who are becoming aware of the advantages of accessing information over high-bandwidth networks, will ultimately be the majority of users. They will access information for entertainment, educational, and financial purposes as well as for business and personal communication. Ease-of-use requirements are coupled with the need for low cost, including all infrastructure, since the user density at each home is relatively low. It is felt that the investment per user should be less than US$500 and the monthly cost should be comparable to the present costs for cable and telephone services.

Education at all levels also needs a system that is inexpensive and simple to operate. In addition, the density of users is high, but in a closed environment in which the cost of the infrastructure can be shared. On-line textbooks, digital libraries, educational video databases and the ability for students to not only access information, but develop information for others, is a vision of the future that is being promoted by education researchers.

Health care will be impacted in a number of ways by the future networked environment. In hospitals, the density of users is moderate, and again in a relatively closed environment in which a sophisticated infrastructure could be provided. The users are relatively mobile within their closed environment, and need access to a wide variety of information such as patient records, medical journals, and books. They are highly motivated to be as efficient as possible in their activities, and are very receptive to a system that would provide them ubiquitous personal-information access.

Roadmap: After covering the specification and an overview of the our design in Sections 2 and 3, we cover the prototype pad in Sections 4 through 6. Section 7 covers conclusions and future work.

2 System Specification

Some of the critical aspects of a system designed to support access to the high-bandwidth network are the following:

Multimedia Access: The future high-bandwidth communication networks will still deliver voice, but users will also require video, text/graphics and audio. The phenomenal growth of the World Wide Web and associated clients for viewing the multimedia information on the Web indicates the demand for such information. This is particularly amazing in the light of the relatively low bandwidth of most of the Internet, typically 1.5 megabits per second (Mbps) or less.

Terminal Portability: Convenient, ubiquitous access to the network dictates that the I/O device be portable. The resistance to reading large amounts of text on a computer screen, instead of paper, is due in large part to the inconvenience of the usual fixed desktop placement of the screen. However, when multimedia data formats are incorporated into a document the use of paper becomes obsolete, in spite of its historical significance and continuing proponents. Thus the electronic viewing device should have the convenience of paper, yet retain the capabilities of an advanced multimedia desktop unit. This implies the portable device should weight less than a pound and should have a form factor that allows convenient observation, such as an 8x11 inch notepad, with a long battery life (a week a more), while providing color video and audio output and pen and microphone input.

There continues to be rapid progress in the area of LCDs, which usually require a backlight to give sufficient contrast. Fortunately, there has been a recent disclosure from Sharp of reflective color displays that consume only 50 mW for a 5” panel. Although this is too small for our application, it is clear that the continuing enormous efforts in LCD technology will provide a low-power screen solution. We are also investigating backlit active-matrix screens, which are very readable but (currently) expensive and power hungry.

Wireless Communications: The most demanding requirement of the notepad is the requirement that full motion video be transmitted over a wireless downlink. Video quality equivalent to present day television can be compressed to rates on the order of 1–2 Mbps, which sets the minimum data rate for this link. The data on the uplink (from the portable to the base station) is audio and pen data, which can be adequately transferred by a rate of 64 Kbps. Thus the link is quite asymmetrical with a downlink requirement of more than 10 times the uplink rate. Video for teleconferencing may be desirable on the uplink, but compression algorithms for this application are in use that only require data bandwidths of a small multiple of 64 Kbps.

In order to provide high bandwidth (1–2 Mbps) wireless communications to a high density of users, in a reasonable amount of spectrum, extremely efficient use of the wireless media must be employed. Infrared is appealing because of the isolation achieved between rooms, however the composite data rate to support a high user density such as found in a classroom, perhaps 50 users at 2 Mbps each, has yet to be demonstrated. A radio solution must also utilize high levels of reuse, but the inter-cell isolation is not as effective since radio can penetrate non-metal barriers. High levels of reuse result in an interference limited channel in which the primary contributor to bit errors is transmission from other users and other basestations. This is optimized by reducing the size of the cells to the smallest practical size, 5–10 meters, with a reduction in the transmit power levels to well under 1mW.

The wireless link design then becomes a trade-off between capacity and interference, yielding for our design, at *maximum* user density, a bit-error rate (BER) of 10^{-2}–10^{-3}.

This places a significant limitation on the kind of data that can be transmitted over the wireless link since this data must be robust against corruption. A retransmission protocol could ensure end-to-end data integrity, but at the high BER of the wireless link, the reduced throughput and increased latency is unacceptable for real-time I/O. We can also increase the transmission power dynamically to achieve a lower BER when not all of the channels are handling video.

It is interesting to note that as the amount of compression for I/O data is increased, it is often necessary to increase the level of error protection. This results in the need for compression algorithms optimized for error tolerance as well as the amount of compression. This results in a data stream that requires less error protection, which yields higher perceived quality at lower bandwidths.

Backbone Network: The backbone network bandwidth is assumed to be sufficiently high to supply multimedia data to many users, who could potentially be in a relatively small area, such as a classroom. This is not expected to be a major problem, since the wireless link places a severe restriction on the data rates that can be delivered to the user. Thus in a classroom of 50 users that are all served by a single basestation at the maximum data rate, the composite rate would only be 100 Mbps, which is achievable with currently available ATM networks.

The network must also be able to provide routing to a mobile host, while retaining the present inter-networking capability of TCP/IP. Since the cells are small, there will be many transfers (handovers) among basestations, which requires the efficient adaptive routing to these mobile users.

A more difficult challenge is the round-trip latency between the portable terminal and the servers on the backbone network. This includes all the processing required in the portable unit, basestation and servers, along with the network latencies. Latency control is most critical in those applications in which there is a tight interactive loop between user input and display update. The latency that can be tolerated is actually quite similar for the various kinds of I/O data. If the delay between the pen movement and the subsequent screen update is under 30 ms, it will not be detected, since this is the update rate of the display. Experience with long-distance phone connections reveals that delays under 30 ms are also not objectionable to the user.

3 InfoPad Project

The goal of the InfoPad project is to design a prototype system that meets the specifications that have been outlined in the previous section for the future environment described in the introduction. A light-weight portable pad is being designed to provide access to a wide variety of network services over a high-speed backbone, which will run protocols to support mobility and real-time data.

The design of a complex system such as the InfoPad requires careful trade-offs among cost, size, functionality, flexibility, communication bandwidth and reliability. Making these trade-offs requires a design methodology that supersedes the component level and addresses architecture, algorithm and protocol selection as well. Also essential is to include power consumption as an intrinsic element of the design cost, as portability makes power minimization one of the primary design requirements.

The design of the first prototype pad has been completed. This portable pad supports pen and audio input, with text/graphics and audio output and is thus termed the *IPGraphics* pad. A chipset to support video decompression and display was designed and fabricated, but not in time for this first version of the pad. Commercial radios were used in this first design in order to investigate issues relating to data throughput, channel models, the effect of bit errors on the applications and wireless protocols. We also developed network software for a single user and a single basestation, to facilitate measurements of the round-trip latency.

The primary challenge of reducing the weight of the terminal is the power consumption of the circuitry to convert the data from a raw 2 Mbps downlink data stream to the form required by the video and text/graphics display and audio output. The most demanding function is the decompression and associated frame buffering of the video data, which must be performed in real time, and the subsequent conversion to analog signals to drive the LCD.

One important strategy for accomplishing this is to move as much of the processing as possible out of the portable unit into servers on the network. This results in a design in which there is no user-accessible computation in the portable pad, which not only relieves the portable unit of the need for general-purpose operating systems, but also eliminates the need for mass-storage devices, memory, and a high-speed microprocessor, all of which are expensive and power hungry.

By providing only I/O on the pad, we also reduce the need for error-free communication, which allows significantly more bandwidth in practice. Decoupling the pad from the computing resources extends the life of the pad by allowing us to improve its computing power over time.

4 Low-Power Multimedia Terminal Chipset

The computation remaining in the pad was then implemented with the lowest possible power consumption by design of a custom chip set. This chip set supported the functions shown below the dashed line, in the area labeled CORE, in Figure 2. The radio modem and associated data recovery and protocol controller were implemented using commercial off-the-shelf components. The core chip set provides the interface and buffering between the radio section and a commercial speech codec, pen-input circuitry, and an LCD panel for text/graphics. Specifically, the chips provide protocol conversion, synchronization, error correction, packetization, buffering, and D/A conversion at a total power consumption of only 5 mW.

The power reduction achieved by the custom chipset is indicated by comparison with the ARM microprocessor,

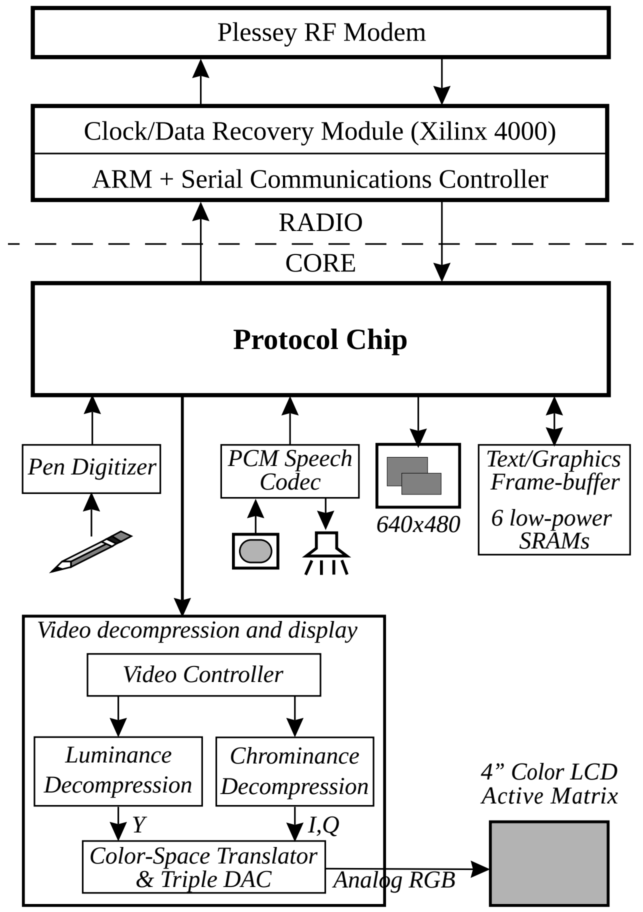


Figure 2: Overview of the InfoPad I/O Terminal

used in the protocol processing. It is one of the lowest-power microprocessors commercially available and one of the very few specifically targeted at low-power applications, yet it still consumes 250 mW and could only perform a fraction of the processing accomplished by the chipset.

All aspects of the chip design were optimized for power. Of most importance, was the use of aggressive supply-voltage reduction, in a strategy that retains throughput by using parallel and pipelined architectures [3]. The chipset was found to operate with a supply as low as 1.1 volts, thus realizing an order-of-magnitude reduction over "low power" 3.3V operation. Also, minimal or near-minimal sized transistors were used whenever possible. Tests indicate that gate and diffusion capacitance still dominate the overall capacitance of the circuit. Therefore, by reducing the transistor widths, the power reduces proportionally, while delays only slightly increase due to the fixed interconnect capacitance. This strategy results in another order-of-magnitude power reduction over a cell library optimized for speed.

An example of the architectural strategies that reduce the power can be seen in an investigation of the text/graphics frame buffers, which only consume 0.5 mW at 1.5V. The memory chips minimize power-consuming switching by first decoding at the block level, and then activating only one of the eight memory blocks. By using minimal column decoding, in this case 2 to 1, only 64 bitlines are charged and discharged each cycle. By pre-charging the bitlines to a voltage less the supply, the bitline swing was reduced to 0.35V. The blocks eliminate glitches on the data bus during reads by initially tristating the outputs and then using a self-timed clock for enabling after the sense amp has settled. The architectural techniques that relaxed the speed requirements on the internal logic were sufficient to allow the frame buffers to operate with supply as low as 1.1 volts.

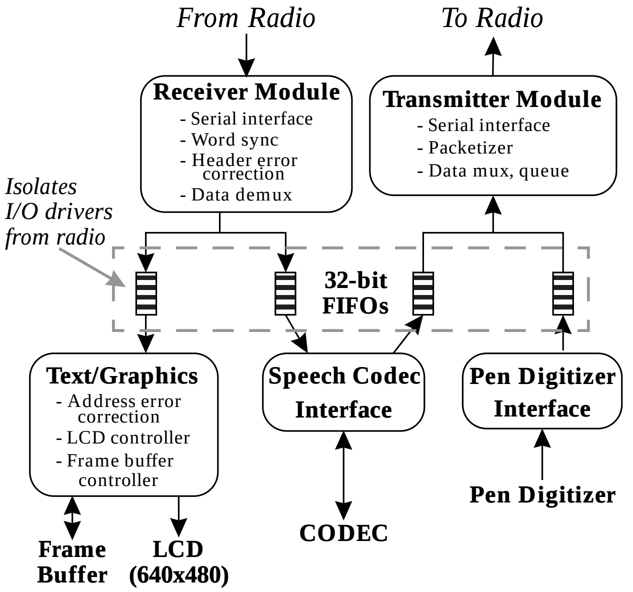


Figure 3: Block diagram of the protocol chip

4.1 Protocol Chip

The Protocol Chip shown in Figure 3 is used to communicate between the various I/O devices in the system. On the uplink, 4-Kbps digitized pen data and 64-Kbps speech data are buffered using FIFOs, arbitrated and multiplexed, packetized, and transmitted in a serial format to the serial communications controller (SCC). On the downlink, serial data from the radio at a 1-Mbps rate is de-packetized and demultiplexed, the header information, which contains critical information such as data type and length, is error corrected and transferred through FIFOs to one of the three possible output processing modules: speech, graphics, and (in the next version) video decompression. This chip makes extensive use of gated clocks to power down unused modules. For example, the error-correction module is powered-up only to process header information and is shut down otherwise. At the layout level, power reduction is obtained by minimizing the length of nets that have high transition activity, such as display data. The power consumption of this chip is less than 2 mW. Within this chip are modules that support the radio interface and the interface to the I/O hardware as seen in Figure 3.

Text/Graphics Module: The graphics module functions as a display controller for the 640x480 active-matrix LCD display. Bit-mapped graphics, generated by a modified X server on the backbone network, are transmitted to the pad beginning with a synchronization pattern, followed by Address, Length and Data fields that are aligned on 32-bit boundaries. The Address and Length fields are error encoded to 32 bits, and indicate the starting address and number of words to write to the display memory. The synchronization pattern is used to realign packet boundaries should the Length field be corrupted.

The 32-bit alignment was a design decision that was made early on with the rationale that the efficiency would be the highest for intensive graphics operations, such as repainting a window. We later came to regret this decision, because during actual operation, repainting the cursor is the most common graphics operation, especially for a pen-based system. The efficiency of this packetization in this case is less than 25%, because each horizontal swath of the cursor occupies at most 1 word, with 1 byte usually being sufficient; thus a redraw of the cursor is a series of horizontal lines and requires between 8 and 16 packets to complete.

The next generation graphics module, which has been submitted for fabrication, has added several features to improve the performance of the system. First, both horizontal and vertical draw commands are supported, which will greatly simplify operations such as painting borders or window frames. Second, painting an arbitrary block of raster data is supported, which will reduce the overhead for cursor painting. Finally, a "protected" mode has been added, which will accept the incoming block only if the CRC indicates that the packet is error free, which allows the pad to display an asymptotically error-free image given periodic screen updates from the display server.

Speech Module: The speech module for the IPGraphics terminal transmits sampled speech from the pad to speech recognition software running on a workstation on the backbone network. This configuration puts the highest bandwidth demand on the InfoPad uplink, but has the lowest computational and power requirements for the portable hardware. We have already developed the software for the speaker-independent, continuous speech recognizer, and are currently developing a programming interface to allow user applications to dynamically change the vocabulary and grammar of the recognizer, thereby using the application's knowledge of its own context to increase the recognition accuracy. This programming interface will be part of an integrated pen and speech user interface for InfoPad.

In the next generation of the InfoPad, improved recognition architectures will be supported. Because of the success of our low-power design strategies, we feel we will be able to include some additional processing without significantly increasing the total power of the pad. In this configuration, the front-end signal processing of the speech recognition algorithm will be performed on custom hardware on the InfoPad, then the output of the front end is sent to the recognizer on the network. This configuration has two main advantages over the first one. First, the output of the front end algorithm has a lower data rate than the raw speech data, so the InfoPad can reduce its uplink bandwidth by about a factor of four when using speech recognition. Second, the A/D on the InfoPad can use a higher sampling rate and take higher resolution samples, which increases the speech recognizer's accuracy without increasing the uplink bandwidth.

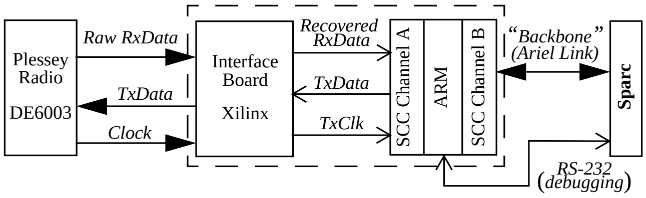
Chip Description	**Area (max)**	**Min Supply**	**Power at 1.5V**
Protocol	9.4 x 9.1	1.1V	1.90 mW
Frame-buffer SRAM (with loading)	7.8 x 6.5	1.1	0.500
Video Controller	6.7 x 6.4	1.1	0.150
Luminance Decompression	8.5 x 6.7	1.1	0.115
Chrominance Decompression	8.5 x 9.0	1.1	0.100
Color-Space Conversion and Triple DAC	4.1 x 4.7	1.3	1.10

Table 1: Chipset Statistics

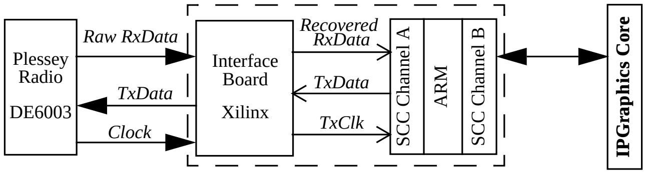
Video Module: The next version of InfoPad will provide support for one-way full-motion video. The video decompression module is realized using four chips that have already been fabricated and tested. They make use of a vector-quantization algorithm, which for decompression simply involves memory look-up operations from a codebook of 256 4x4 pixel patterns. Compressed luminance and chrominance video signals (actually YIQ) is buffered using a ping-pong scheme (one for Y and one for IQ), which provides an asynchronous interface to the radio modem and immunity against bursty errors. The amount of RAM required is reduced by a factor of 32 by storing the video in the compressed format. The YIQ decompressed data is sent to a third chip that converts this data to digital RGB and then to analog form using a triple DAC that can directly drive a 4" active-matrix color LCD. The fourth chip performs the video control functions, which include the synchronization of the various chips and the display, control of the ping-pong memories, and loading of the code-books. It uses an addressing scheme that eliminates the need for an output line buffer. The total power consumption of the chip set is less than 2 mW. Table 1 gives the statistics of all of the low-power chips, which were fabricated in 1.2µm CMOS technology.

5 Wireless Link

In the first-round design of physical (PHY) and media-access (MAC) protocols, the primary goal was to design a platform that would allow us to explore the implications of moving as much processing as possible from the portable unit to the network. To this end, we decided to use a commercially-available RF modem even though it deviates substantially from the long-term plan for the InfoPad wireless link [4][5]. The link for the IPGraphics prototype used a half-duplex, 1 Mbps RF modem: the DE6003, from GEC Plessey.



Base Station Hardware



Pad Hardware

Figure 4: Hardware Block Diagram

It was chosen because it had the highest data rate of commercially available radio modems, in a form factor that we could place in the IPGraphics unit.

The DE6003, shown in Figure 4, provides a demodulated signal to the baseband receiver, implemented in a Xilinx 4000 FPGA, without any explicit timing information. This "raw" received data is passed to the clock-recovery module (the *Interface Board*), which extracts timing information from transitions in the data stream. The interface board next reconstructs the data bits from the bit stream, and passes the recovered clock and data to the ARM board.

Early in the design of the wireless link, the decision was made to use a general-purpose microprocessor or microcontroller for the protocol-support hardware, providing flexibility in the design of the protocol. The ARM610, a 32-bit RISC device, was chosen because it offered a relatively low-power solution for our needs, and provided a mature development environment with a C compiler and a remote debugger. Media-access control, flow control, and tracking error statistics for link management are performed by the ARM, while physical-layer functionality, such as clock and data recovery, line coding, and channel sensing, is implemented in the Xilinx FPGA.

A second issue was how to design the radio system so that it could be used in both the basestation and in the IPGraphics terminal, without having to maintain two versions of the software. This symmetry was achieved by using a commercial serial-communications controller (SCC) as the common interface between the ARM and either the IPGraphics Core or the basestation (Figure 4). The SCC performs serial-to-parallel and parallel-to-serial conversion, and supports two independent full-duplex communications channels. Channel A interfaces the ARM to the radio: data to be transmitted is written to the SCC, which serializes the data and passes it on to the radio. Channel B interfaces the ARM to either the IPGraphics core or the basestation. To support interrupt-based designs, the SCC uses FIFOs to buffer input and output data streams: the transmit FIFOs on each channel interrupt the ARM when they are half empty, while the receive FIFOs interrupt when they are half full.

5.1 Physical Layer

The responsibility of the physical layer is to hide the mechanisms by which bits are passed through the channel. The radios provide to the baseband processing hardware what is essentially a two-level analog signal, from which timing and symbols must be recovered. A reference 10-MHz signal from the radio is divided by 16 to generate the 625 KHz transmit clock, and is also used to drive the baseband receiver hardware.

The clock-recovery algorithm relies on the 10 MHz reference to provide a 16x oversampling of the incoming signal. Because the reference is accurate to within ±20 ppm (1 bit in 25,000) over a 90 degree temperature range, the frequencies of the transmitter and receiver are assumed to be constant during a transmission. This reduces the clock recovery problem to choosing one of 16 phases.

The physical layer prepends a synchronization preamble, which consists of a 32-bit sequence of alternating bits, to each packet that is transmitted. At the receiving end, the recovery circuitry is initially in a search mode, looking for a $0 \rightarrow 1$ transition in the incoming data stream. After the required transition, the receiver looks for a $1 \rightarrow 0$ transition to occur one bit-time later, with a window that accounts for jitter. Spurious transitions that occur outside of this window restart the search algorithm, and as the circuitry converges the window is gradually closed to enhance noise immunity.

After 16 successful transitions in a row, the clock is *locked* and the channel is declared busy until the transmission completes. At this point, the circuitry begins to search for the frame-alignment word, which delimits the beginning of the datagram. Next, the SCC is enabled, serial-to-parallel conversion begins, and the assembled bytes are passed to the ARM for further processing.

5.2 Media-Access Control

The MAC layer is based upon a collision-avoidance model, similar to those proposed by IEEE 802.11. Because carrier sense and collision detection cannot be directly measured, as in Ethernet, the protocol normally uses a 4-way handshake (RTS, CTS, DATA, ACK) to negotiate channel access. The fixed-length RTS (Request-To-Send) and CTS (Clear-To-Send) packets include a field specifying the length of the DATA payload, which provides listening (inactive) nodes with information about how long the channel will be occupied. Under the assumption that all nodes can hear each other equally well, these short control frames provide a virtual collision-detection mechanism. Hearing an RTS requires bystanders to wait long enough for a CTS to be transmitted, while receiving a CTS packet requires them to wait long enough for the DATA frame to be transmitted.

This protocol attempts to force the majority of collisions to occur during the RTS/CTS exchange. By making these control packets very short, it attempts to minimize the time

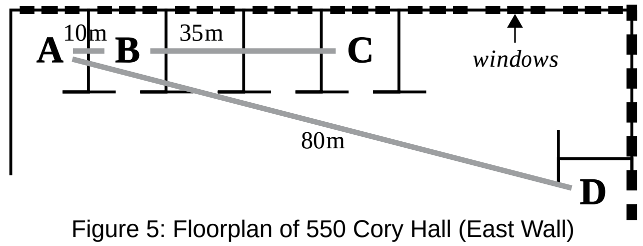


Figure 5: Floorplan of 550 Cory Hall (East Wall)

spent colliding. It does however, rely on the assumption that stations that could interfere with each other have equal visibility, an assumption which we are still investigating.

We discovered that applying this protocol introduced an unacceptable delay, especially for small time-critical packets, such as pen and speech packets. The 5-byte pen packets are the same size as the RTS, CTS, and ACK packets; further, the pen digitizer sends a continuous stream of status information whenever the pen is close to the tablet. Speech packets may be short as well. These short packets, which are transmitted at a regular interval, were handled more efficiently by avoiding the acquisition exchange, and directly sending them. Due to the robustness of this data, lost packets have limited effect on overall pen and speech performance.

5.3 Measurement of Error Characteristics

Because the wireless channels are subject to interference from such a variety of sources, we sought to gain insight on the nature of errors in practice. To simulate an office environment as closely as possible, we conducted our measurements in 550 Cory Hall, a large office space for graduate students. Partitions separate cubicles approximately 8m on a side; the three internal sides are cloth. Measurements were repeated for three different channels, depicted in Figure 5. Each set of measurements consisted of transmitting 1024 1K packets. The received packets were collected and compared to the transmitted data to determine the number of lost packets, as well as the locations of bit errors within the packet.

Table 2 summarizes breakdown of bit errors in the received data stream. The *Path* column indicates the location of the receiver and transmitter. *Raw BER* is the fraction of incorrect bits in the received pattern, and the *Sync Error Percent* and *Random Bit-Error Percent* columns indicate the percentage of errors caused by loss of synchronization versus random bit errors. Finally, the percentage of completely lost packets is shown.

For close placement of the radio transmitter and receiver, within 10m or so, the errors were found to be random, while at greater separations data synchronization is the dominant source or errors. Because the simple synchronization algorithm relies solely on detection of transitions in the data stream to recover the timing information, it is vulnerable to spurious transitions induced by noise.

6 Backbone Network

The first prototype of a basestation and backbone network was developed with two goals in mind. One goal was to develop the criteria for the software architecture that would be required for a fully operational system with multiple users. The other was to investigate the critical issue in the InfoPad architecture, which is the round-trip latency for interactive data. It was felt that if acceptable performance could be accomplished with Ethernet and standard workstations using TCP/IP, then a future system with more sophisticated technology and custom protocols should work well.

Path, meters	**Raw BER**	**Sync BE %**	**Random BE %**	**Dropped Packets**
A→B, 10	$2.6x10^{-4}$	0.00%	100%	0.00%
B→C, 35	$1.2x10^{-3}$	80.1%	19.9%	0.10%
A→D, 80	$8.2x10^{-3}$	98.6%	1.36%	3.90%

Table 2: Error Comparisons. For the short path, A→B, random errors dominate, while for the long path, A→D, sync errors occlude the random errors.

Figure 4 shows a block diagram of the prototype distributed basestation. A Sparc10 workstation equipped with a programmable SBUS card provides a full-duplex serial interface to the wireless link. Five software modules — a customized X-Server, a pen-based notebook application, the pen server, the pad server, and the software gateway — implement the software functionality of the basestation. These modules, which are UNIX processes, can reside on separate machines, and communicate via TCP/IP over Ethernet.

Uplink pen data travels from the *Pad* to the *Gateway*, which forwards the data to the appropriate *Pad Server*, which in turn passes the data to its *Pen Server*.[1] The *Notebook* application then processes the pen event, and computes an appropriate response, such as closing a window. The *XServer* generates the resulting update, which is passed back to the *Pad Server, Gateway*, and finally to the pad.

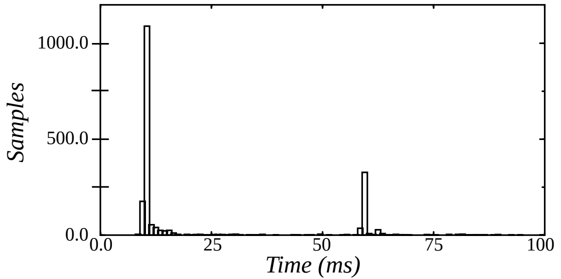
A goal of the design was to provide a maximum round-trip latency of 30 ms from pen event to display update. At first, the round-trip latency was so high that the pen could cross the page before "ink" would appear, with a mean latency of 300 ms. We determined that buffering in the inter-process sockets was contributing most of the latency. With a simple adjustment to a TCP flag to disable local buffering of small packets, we obtained the desired latency.[2]

Although in-transit time still dominates, Figure 6 shows the mean round-trip time is well under 30 ms. The numbers shown are "loopback" latencies, which is the time between pen-to-pad contact and the appearance of "ink" on the display. Figure 6 (a) shows the histogram of round-trip time when the system operates with only pen data. In this case, most delays are under the 30 ms delay target, although the

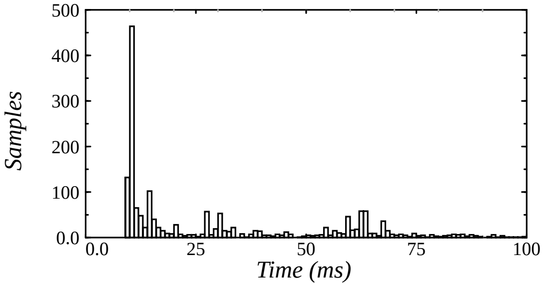
---

[1] 1: We have since taken the Pad Server off of the data path; it now simply tracks all of the services and arranges for the basestation to move data directly to the correct server. This was done primarily to reduce latency.

[2] 2: Specifically, the TCP_NODELAY flag was set on all network sockets to activate the TCP push function.



(a) pen data only



(b) combined pen and video stream

Figure 6: Round-trip latency measurements

peak around 60ms indicates that there is still a problem. One explanation under investigation is that these outliers are due to a collision and subsequent retransmit on the Ethernet.

Figure 6 (b) shows round-trip time when the system operates with both the pen data and full-motion black-and-white video, with the pen data and video streams interleaved in the same *Pad Server*. Here, the majority of the delays are less than 30 ms; however, the distribution is wider than before. It is interesting to note that the two graphs are very similar, indicating that video does not cause noticeable degradation in pen's response time

These results indicate that the InfoPad architecture is both feasible and realizable. However, there are two major architectural limitations in this configuration. The first is an issue of bandwidth: an Ethernet operating at 10 Mbps cannot provide the required 1–2 Mbps for more than a few users. Second, the CSMA/CD protocol used by Ethernet provides no latency guarantees. Under heavy network loading, latencies far greater than 30 ms are common. Both of these problems will be avoided by moving to a switched high-bandwidth network such as ATM.

7 Conclusion and Future Work

The InfoPad project has a long-term goal of providing a system solution to the problem of accessing information from high-bandwidth networks. Optimizations are being made at all levels, extending from the circuit design in the portable units, through to the protocols on the high-speed backbone networks. We are finding that significant advantages can be gained from this unified approach, which could not be achieved by independent efforts at the various levels.

The next generation of InfoPad will support multiple users and cells, full-motion video with audio, higher bandwidth and reduced latency, and real system software and applications. We are also investigating book-quality displays and virtual-reality headsets as alternative output devices. We expect an intermediate prototype to ready by April.

A general conclusion is that the effectiveness of our low-power design methodologies means that we will actually move more functionality into the pad over time. In addition to front-end DSP for speech, we are considering X primitives such as fill rectangle and block transfer that trade pad processing for reduced bandwidth and a faster response time. For example, the fill-rectangle primitive should cut the perceived delay of pop-up menus by almost a factor of two.

The next major step for InfoPad is the software and infrastructure to support large numbers of users in relatively small areas. Although the radio for the basestation can handle 50 users at 2 Mbps each, we still need to develop the rest of the infrastructure, including the servers, to reach that capacity. As a driving application, we are investigating the deployment of 100 pads in Berkeley's Doe/Moffitt library for use as portable search engines and multimedia displays. This would allow pen-based access to the internet and the on-line card catalog from anywhere in the stacks. The library is already moving sources and multimedia *finding aids*, such as representative photographs, on line. Note that a PDA or laptop would be useless in this scenario, since the critical benefit is access to the remote database, which is far too large to fit in any portable device and requires substantial computing power to search.

The key to cost effective computing for large numbers of users is to build fault-tolerant servers based on networks of workstations. The development of fast response-time, highly available servers is an area of ongoing research in conjunction with the Berkeley NOW project [1].

In the longer term, we believe that the right device will be a combination InfoPad and PDA: within the wireless infrastructure it is an InfoPad, but outside it is a PDA with cellular-phone capabilities. Toward this goal, we are developing a single-chip multi-standard RF modem, which would allow one device to speak a wide variety of standards including European, Japanese and US cellular.

[1] T. E. Anderson, D. E. Culler and D. A. Patterson. The Case for NOW. *IEEE Micro*, to appear, 1995.

[2] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power Digital CMOS Design", *IEEE Journal of Solid State Circuits*, 473–484, April 1992.

[3] A. P. Chandrakasan, A. Burstein, and R. W. Brodersen, "A Low-power Chipset for Multimedia Applications", *IEEE International Solid-State Circuits Conference*, 82–83, February 1994.

[4] S. Sheng, A. Chandrakasan, and R. Brodersen, "A Portable Multimedia Terminal," *IEEE Communications Magazine*. 64–75, December 1992.

[5] S. Sheng *et al*., "Integrated CDMA Receiver" *Proceedings of Wireless '94*, 264–275, Calgary, Alberta, Canada, July 1994.