
**TECHNOLOGIES
FOR
WIRELESS COMPUTING**

edited by

Anantha P. Chandrakasan
Massachusetts Institute of Technology

and

Robert W. Brodersen
University of California, Berkeley

Reprinted from
a Special Issue of
JOURNAL OF VLSI SIGNAL PROCESSING SYSTEMS
Vol. 13, Nos. 2 & 3
August/September, 1996

KLUWER ACADEMIC PUBLISHERS
Boston / Dordrecht / London

JOURNAL OF VLSI SIGNAL PROCESSING SYSTEMS

Volume 13—1996

Special Issue on Technologies for Wireless Computing

Guest Editors' Introduction	<i>Anantha P. Chandrakasan and Robert W. Brodersen</i>	1
Hardware-Software Architecture of the SWAN Wireless ATM Network	<i>Prathima Agrawal, Eoin Hyden, Paul Krzyzanowski, Mani B. Srivastava and John A. Trotter</i>	3
An Integrated Testbed for Wireless Multimedia Computing	<i>Charles Chien, Sean Nazareth, Paul Lettieri, Stephen Molloy, Brian Schoner, Walter A. Boring IV, Joey Chen, Christopher Deng, William H. Mangione-Smith and Rajeev Jain</i>	21
Design of a Low Power Video Decompression Chip Set for Portable Applications	<i>Benjamin M. Gordon, Ely Tsern and Teresa H. Meng</i>	41
IC Implementation Challenges of a 2.4 GHz Wireless LAN Chipset	<i>M. Chian, G. Croft, S. Jost, P. Landy, B. Myers, J. Prentice and Doug Schultz</i>	59
Flat Panel Displays for Portable Systems	<i>Kalluri R. Sarma and Tayo Akinwande</i>	81
Threshold-Voltage Control Schemes through Substrate-Bias for Low-Power High-Speed CMOS LSI Design	<i>Tadahiro Kuroda and Takayasu Sakurai</i>	107
Processor Design for Portable Systems	<i>Thomas D. Burd and Robert W. Brodersen</i>	119
Instruction Level Power Analysis and Optimization of Software	<i>Vivek Tiwari, Sharad Malik, Andrew Wolfe and Mike Tien-Chien Lee</i>	139
Low-Power Architectural Synthesis and the Impact of Exploiting Locality	<i>Renu Mehra, Lisa M. Guerra and Jan M. Rabaey</i>	155
Techniques for Power Estimation and Optimization at the Logic Level: A Survey	<i>José Monteiro and Srinivas Devadas</i>	175

Distributors for North America:

Kluwer Academic Publishers
101 Philip Drive
Assinippi Park
Norwell, Massachusetts 02061 USA

Distributors for all other countries:

Kluwer Academic Publishers Group
Distribution Centre
Post Office Box 322
3300 AH Dordrecht, THE NETHERLANDS

Library of Congress Cataloging-in-Publication Data

A C.I.P. Catalogue record for this book is available
from the Library of Congress.

ISBN-13: 978-1-4612-8633-2 e-ISBN-13: 978-1-4613-1453-0
DOI: 10.1007/978-1-4613-1453-0

Copyright © 1996 by Kluwer Academic Publishers

Softcover reprint of the hardcover 1st edition 1996

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, mechanical, photo-copying, recording, or otherwise, without the prior written permission of the publisher, Kluwer Academic Publishers, 101 Philip Drive, Assinippi Park, Norwell, Massachusetts 02061

Printed on acid-free paper.

Guest Editors' Introduction

ANANTHA P. CHANDRAKASAN
Massachusetts Institute of Technology

ROBERT W. BRODERSEN
University of California, Berkeley

Research over the last decade has enabled high-performance systems such as powerful workstations, sophisticated computer graphics, and multimedia systems such as real-time video and speech recognition. A significant change in the attitude of users is the desire to have access to this computation at any location, without the need to be connected to the wired power source. This has resulted in the explosive growth of research and development in the area of wireless computing over the last five years.

This special issue deals with several key technologies required for wireless computing. The topics covered include reliable wireless protocols, portable terminal design considerations, video coding, RF circuit design issues and tools, display technology, energy efficient application specific and programmable design techniques, energy efficiency metrics, low-voltage process technology and circuit design considerations, and CAD tools for low-power design at all levels of abstraction.

The first three papers deal with low-power wireless terminal design, protocols, and system infrastructure. Agrawal et al., presents a wireless ATM network (SWAN) that provides end-to-end connectivity to mobile end-points equipped with RF transceivers for wireless access. This paper describes the design and implementation of the ATM-based wireless last-hop, including the air-interface control, the MAC, and low-level ATM transport signaling. Chien et al., presents a testbed to evaluate node architectures that support multimedia applications and services across a wireless network. A low bitrate subband video compression algorithm is evaluated for video networking across bandwidth-limited RF channels. Gordon et al., describes the design of a low-power video decompression chipset for portable applications. A error resilient algorithm is used based on subband decomposition and pyramid vector quantization. A variety of power reduction techniques are presented for application specific designs including low-voltage operation, computation vs. memory trade-offs, programmability vs. dedicated hardware, etc.

Chian et al., describes the IC implementation challenges of a 2.4 GHz wireless LAN chipset developed at Harris Semiconductor. The technology considerations, CAD methodology, and the manufacturing considerations are presented. The lessons learned from designing this chipset are presented. Sarma and Akinwande review the flat panel technologies available for portable systems. They review display requirements, and propose metrics to evaluate display technologies. Current day as well as emerging technologies are evaluated.

The Kuroda and Sakurai paper presents some key technology and circuit considerations for low-voltage high-performance system design. They propose a standby power reduction technique where the threshold voltage of the devices are raised to lower idle leakage power. They also propose feedback circuits to adjust the substrate bias to reduce fluctuations in threshold voltage. Burd and Brodersen present techniques for energy efficient programmable processor design. A key contribution in this paper is the definition of energy efficiency metric for various user modes including fixed throughput maximum throughput, and burst throughput modes. Tiwari et al., presents techniques to analyze and optimize power dissipation of software. A measurement based instruction level power analysis approach is used to provide an accurate power cost for software. The ability to model power dissipation of software is key to finding energy efficient programmable implementations.

The final two papers address CAD tool issues for low-power design. Mehra et al., presents various architectural and behavioral approaches for power minimization. A key new idea emphasized involves algorithm partitioning to preserve locality in the assignment of operations to hardware units. This not only reduces implementation area, but

also the number of accesses to high capacitance interconnect. Monteiro and Devadas review power estimation and optimization techniques at the logic level. Simulation-based as well as probabilistic approaches are described for switching activity estimation in sequential circuits. Various power reduction techniques are described including a data dependent logic level power down approach called precomputation.



Anantha P. Chandrakasan received the B.S., M.S., and Ph.D. degrees in Electrical Engineering and Computer Sciences from the University of California, Berkeley, in 1989, 1990, and 1994 respectively. Since September 1994, he has been the Analog Devices career development assistant professor of Electrical Engineering at the Massachusetts Institute of Technology, Cambridge. He received the NSF Career Development award in 1995, the IBM Faculty Development award in 1995 and the National Semiconductor Faculty Development award in 1996. He received the IEEE Communications Society 1993 Best Tutorial Paper Award for the IEEE Communications Magazine paper titled, "A Portable Multimedia Terminal". His research interests include the ultra low power implementation of custom and programmable digital signal processors, wireless sensors and multimedia devices, emerging technologies, and CAD tools for VLSI. He is a co-author of the book titled "Low Power Digital CMOS Design" by Kluwer Academic Publishers.



Robert W. Brodersen received Bachelor of Science degrees in Electrical Engineering and in Mathematics from California State Polytechnic University, Pomona, California in 1966. In 1968 he received the Engineers and M.S. degrees from the Massachusetts Institute of Technology, (MIT) Cambridge, and he received a Ph.D. in Engineering from MIT in 1972.

From 1972–1976, Brodersen was with the Technical Staff, Central Research Laboratory at Texas Instruments, Inc., Dallas. He joined the Electrical Engineering and Computer Science faculty at the University of California at Berkeley in 1976, where he is currently a professor. In addition to teaching, Professor Brodersen is involved in research inclusive of new applications of integrated circuits, focused in the areas of low power design and wireless communications.

He has won conference best paper awards at Eascon (1973), International Solid State Circuits Conference (1975) and the European Solid State Circuits Conferences (1978).

Professor Brodersen received the W.G. Baker award for the outstanding paper in the IEEE Journals and Transactions (1979), Best Paper Award in the Transactions on CAD (1985) and the Best Tutorial paper of the IEEE Communications Society (1992).

In 1978 Professor Brodersen was named the outstanding engineering alumnus of California State Polytechnic University. He became a Fellow of the IEEE 1982. He was co-recipient of the IEEE Morris Libermann award for "Outstanding Contributions to an Emerging Technology," in 1983. And he received Technical Achievement Awards from the IEEE Circuits and Systems Society in 1986 and in 1991 from the IEEE Signal Processing Society.

Professor Brodersen was elected a member of the National Academy of Engineering in 1988. In September of 1995, he was appointed the first holder of the John R. Whinnery Chair in Electrical Engineering at University of California, Berkeley.

Hardware-Software Architecture of the SWAN Wireless ATM Network

PRATHIMA AGRAWAL, EOIN HYDEN, PAUL KRZYZANOWSKI, MANI B. SRIVASTAVA
AND JOHN A. TROTTER

Lucent Technologies, Bell Laboratories, 600 Mountain Avenue, Murray Hill, NJ 07974

Received October 26, 1995; Revised April 18, 1996

Abstract. The SWAN (Seamless Wireless ATM Network) system provides end-to-end ATM connectivity to mobile end-points equipped with RF transceivers for wireless access. Users carrying laptops and multimedia terminals can seamlessly access multimedia data over a backbone wired network while roaming among room-sized cells that are equipped with basestations. The research focus on how to make ATM mobile and wireless distinguishes SWAN from present day mobile-IP based wireless LANs. This paper describes the design and implementation of the ATM-based wireless last-hop, the primary components of which are the air-interface control, the medium access control, and the low-level ATM transport and signalling.

The design is made interesting by its interplay with ATM; in particular, by the need to meaningfully extend over the wireless last-hop the service quality guarantees made by the higher level ATM layers. The implementation, on the other hand, is an example of hardware-software co-design and partitioning. A key component of the wireless hop implementation is a custom designed reconfigurable wireless adapter card called FAWN (Flexible Adapter for Wireless Networking) which is used at the mobiles as well as at the basestations. The functionality is partitioned three-way amongst dedicated reconfigurable hardware on FAWN, embedded firmware on FAWN, and device driver software on a host processor. Using an off-the-shelf 625 Kbps per channel radio, several of which can be supported by a single FAWN adapter to provide multiple channels, per-channel unidirectional TCP data throughput of 227 Kbps (or, 454 Kbps bidirectional) and per-channel unidirectional native ATM data throughput of 210 Kbps (or, 420 Kbps bidirectional) have been obtained.

1. SWAN Network for Mobile and Wireless ATM

Progress in wireless access, multimedia processing, high-speed integrated service wired networking, and low-power electronics promises to provide mobile users with ubiquitous multimedia information access in the near future. Wireless technology will soon make per-user data rates of several Mbps possible, at least indoors. Even the current generation of 2.4 GHz Industrial, Scientific, and Medical (ISM) band can provide around 1 Mbps per channel while supporting a reasonable user density when used in a spatially-multiplexed pico-cellular environment. Advances in multimedia processing, such as improved compression algorithms, can allow transmission of packetized video and audio to a mobile. Emerging high-

speed integrated service networking technologies, such as ATM and reservation-based IP, enable multimedia communications by allowing end-to-end service quality guarantees. Low-power circuit, display, and packaging technologies will enable portable multimedia end-points that seamlessly integrate into a user's networked computing environment.

The SWAN (Seamless Wireless ATM Network) system [1] is being developed to explore the synergy among the above technology trends. It provides indoor mobile users carrying heterogenous multimedia end-points with continual audio, video, and data connectivity to the network. A distinguishing feature of SWAN is that it uses end-to-end ATM connectivity as the underlying communication mechanism. This choice is in contrast to mobile-IP based wireless LANs, such as

WaveLAN [2], that extend TCP/IP data LANs into the wireless and mobile domain. It appears likely that the core of networks providing ubiquitous and tetherless access to multimedia information will be based on the emerging ATM cell switching networks that provide integrated transport of voice, video, data, and other multimedia traffic. ATM, with its virtual circuit (VC) paradigm, makes it possible for service quality guarantees to be given to a specific connection. SWAN, with its focus on multimedia communication, uses an ATM network as its wired backbone. It, therefore, appears to make sense that the ATM virtual circuit paradigm be extended over the wireless hops of SWAN as well, thereby allowing the system to give service quality guarantees end-to-end over both the wireless and the wired links.

The high level network model in SWAN is based on extending the backbone wired ATM network over the wireless last hop via special ATM switching nodes, called basestations, at the periphery of the wired network. The basestations are equipped with one or more radio ports, and provide wireless links to geographically nearby mobile hosts, which are also equipped with radio adapters. The geographical area corresponding to a basestation is called its cell, and the various basestations are distributed in room-sized cells. All ATM virtual circuits terminating at a mobile pass through its basestation, and network connectivity is continually maintained as a mobile moves from the vicinity of one basestation to that of another. Mobiles in SWAN must have the capability to participate in the necessary ATM signalling and data transfer protocols. Otherwise, there are no restrictions on the mobility or functionality of the mobiles. They include “smart” PDAs and laptops, “dumb” multimedia terminals, and infrequently moving wireless entities such as cameras and printers.

The architecture of SWAN gives rise to many technical problems. At a high level, the control and signalling in the network needs to be able to establish, reroute [3] and tear-down VCs to and from mobile hosts; provide service guarantees for these VCs in the presence of mobility; and, provide call-back to context-adaptive applications [4] which have registered interest in mobility and other network context related events. Virtual circuits carrying audio and video must, as far as possible, be immune from disruptions when a mobile host *hands-off* from one basestation to a neighboring one. Novel algorithms, protocols, and their implementation, for these higher level aspects of SWAN have been described elsewhere [5].

Of more interest to this paper are the lower layers of SWAN that provide the air-interface, medium access control, and wireless ATM data link functionality over the wireless last hop. These layers must be cognizant of and cooperate with the higher mobile ATM layers to enable efficient wireless bandwidth utilization, and minimal latency and traffic disruption during hand-off. After presenting the wireless last hop functionality, the paper describes its implementation at the two ends of a hop: the basestations and the mobiles. The implementation is presented as a three-way partitioning between dedicated hardware, embedded software (firmware), and host kernel software. The first two are resident on a custom designed wireless adapter card called FAWN (Flexible Adapter for Wireless Networking), while the third is embodied as device driver software on the host CPU.

To put the rest of the paper in perspective, Fig. 1 illustrates the SWAN system at a high level. A mobile is shown communicating over a wireless link with a basestation using radio frequency (RF) ATM adapters at each end. The basestation, which is functionally an ATM switch, is also equipped with a wired ATM adapter to allow it to communicate with a wired ATM network, which in turn may be interconnected to the Internet. Thus, SWAN hardware provides the necessary low level physical connectivity across wired and wireless links to allow mobile to communicate with other SWAN mobile and wired hosts, as well as to Internet hosts. Using this, the upper layers of SWAN provide two modes of data transport capabilities to applications.

The first mode is *mobile and wireless ATM*. How best to make ATM mobile and wireless is the key technical advance provided by SWAN. Using mobile and wireless ATM, applications get end-to-end ATM transport with the attendant benefits of virtual circuits (VCs) for which quality of service (QoS) parameters such as bandwidth can be negotiated by the applications, and for which individualized policies (as opposed to generic policies for all data traffic) can be used for data packet scheduling, error control, and hand-off. These benefits of mobile and wireless ATM are used in multimedia applications which require audio, video, and other real-time traffic streams to be transported across the network. For example, as shown in Fig. 1, the *nv* video player application has been modified by us to use SWAN’s ATM mode so that it can establish a connection of a specified data rate and delay to a live camera feed at a workstation. Protocols in SWAN, such as those embodied in the Connection Manager modules

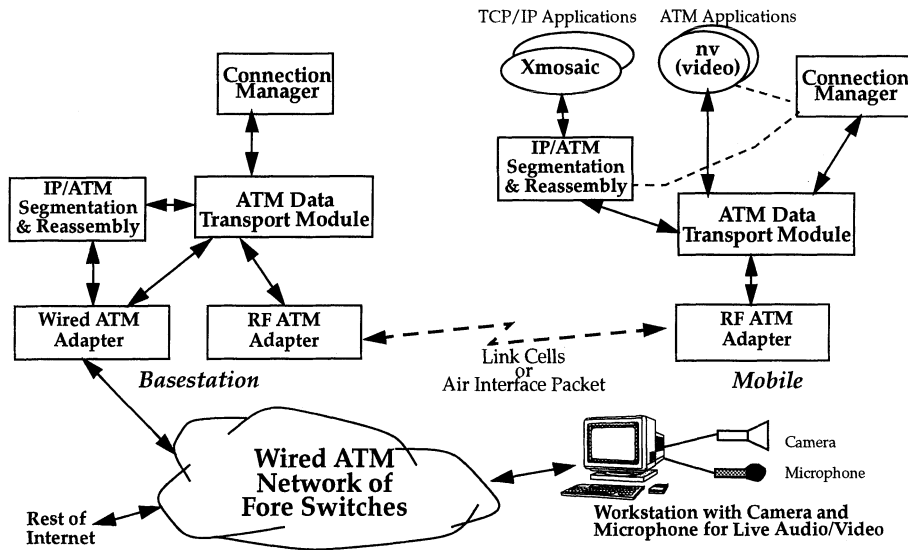


Figure 1. ATM connectivity in SWAN for multimedia applications.

shown in Fig. 1, reroute virtual circuits and attempt to maintain their quality of service parameters as a mobile moves from basestation to basestation. If enough resources are not available, the application is notified allowing it to adapt.

The second data transport mode in SWAN is *mobile IP* connectivity, which is provided to support legacy TCP/ IP based internet applications. In this mode, as with any IP network such as the Internet, SWAN acts as a best effort network with no notion of quality of service for a connection. To provide IP data transport, *IP/ATM segmentation-reassembly* modules are used at the basestations and the mobiles, as shown in Fig. 1. This allows IP packets to be transported over the wireless link. *IP forwarding* modules at the basestations route packets between the wireless link and the rest of the IP-based internet, so that the basestation acts as an IP router as well.

2. Related Work

While wireless networks with end-to-end ATM are still the subject of research, cellular telephone networks, indoor wireless data LANs such as WaveLAN [2], and outdoor cellular wide-area and metropolitan-area data networks such as CDPD and Metrocom's Ricochet are three broad categories of wireless networks that exist. Cellular telephone networks are connection oriented, and use either frequency division multiple access (in older analog networks), or use time or code division

multiplexing (in newer digital networks). These networks provide only voice bit rate connections but with a rigid guaranteed bandwidth.

Closer to SWAN's domain are the indoor wireless data LANs such as WaveLAN from Lucent and RangeLAN from Proxim. The radios used in these networks are typically ISM band radios (like SWAN's) and may be either frequency hopping spread spectrum based, or direct sequence spread spectrum based. Frequency hopping based radios are relatively recent, and smart algorithms for the control of frequency hopping are still proprietary. These wireless LANs, unlike SWAN, offer no notion of quality of service. Therefore, while such wireless LANs can carry mobile IP traffic (which too does not have the ability to specify bandwidth, delay, etc.), they are inadequate for carrying mobile and wireless ATM. The medium access control and physical control layers in these wireless LANs are the subjects of the upcoming IEEE 802.11 standard. Broadly speaking, all these networks operate in a peer-to-peer fashion with the mobiles and the wired network access points operating as peers in a shared broadcast channel. This is similar to what goes on in an ethernet, and actually the medium access control layers in these wireless LANs are also ethernet variants in that the multiple access is based on CSMA (carrier sense multiple access) enhanced with collision avoidance and handshaking [6–9]. Polling based medium access control [7] have also been proposed for use in wireless data LANs. Metropolitan-area data networks such as CDPD and Metrocom's Ricochet provide capabilities

similar to wireless LANs, except at a wider geographical scope and with lower data rates. SWAN is distinguished from these networks by its use of ATM to provide a multimedia-oriented mobile and wireless integrated service network with a quality of service environment for individual connections.

Mobile and wireless ATM networks are still in research stages, although certain aspects of wireless last hops in such networks have previously been explored in the literature. Chandler et al. [10] address the problem of ATM transmission over a CDMA based wireless network, and in particular describe the mapping between ATM cells and air-interface packets. Raychaudhuri and Wilson [11] explore an ATM-based architecture for next generation multiservices personal communication networks. While providing philosophical underpinnings for the mobile and wireless ATM networks, these previous studies have been at an abstract level.

Only recently have a few research groups begun to explore algorithm, protocol, architecture and implementation issues in realizing mobile and wireless ATM. Besides our and other [12, 13] efforts within Bell Labs, NEC's WATMNet [14] and ORL Cambridge's wireless ATM system [15] are the two other concurrent wireless research systems that are implementing wireless and mobile ATM, though with different approaches and scope. Compared to other wireless and mobile ATM efforts, the SWAN system uses novel low latency VC rerouting algorithms based on performance triggered rebuilds [16], custom reconfigurable and miniature wireless ATM adapter hardware [17], and support for heterogeneous end-systems ranging from laptops to dumb multimedia terminals [18]. In addition, SWAN

is among the first to have a functional and portable prototype system hardware.

3. SWAN's ATM-Based Wireless Last Hop Architecture

Figure 2 details functional blocks in the wireless last hop of SWAN. Following a basestation-centric model, all wireless communications in SWAN is between a mobile and its basestation. Therefore, the wireless last hop in SWAN is really a number of wireless links between mobiles and their basestations, with the various wireless links sharing the air resources. The primary function of the basestation is to switch cells among various wired and wireless ATM adapters attached to the basestation, so that the basestation can be viewed as an ATM switch that has wireless (RF) ATM adapters on some of its ports. In SWAN, however, generic PCs and Sun workstations are used as basestations by plugging in a wired ATM adapter card and one or more RF wireless ATM adapter cards. The cell switching functionality is realized in software using a kernel-space-resident *ATM Data Transport (DT)* module, and a user-space-resident *Connection Manager (CM)* signalling module. The use of PCs and workstations for basestations allows them to act as wired hosts as well, running application processes. In essence, basestations in SWAN are nothing but computers with banks of radios. Placing the basestation signalling (CM) and switching (DT) functionality on host computers has the advantage of allowing one to leverage on cheap PC hardware and software technology.

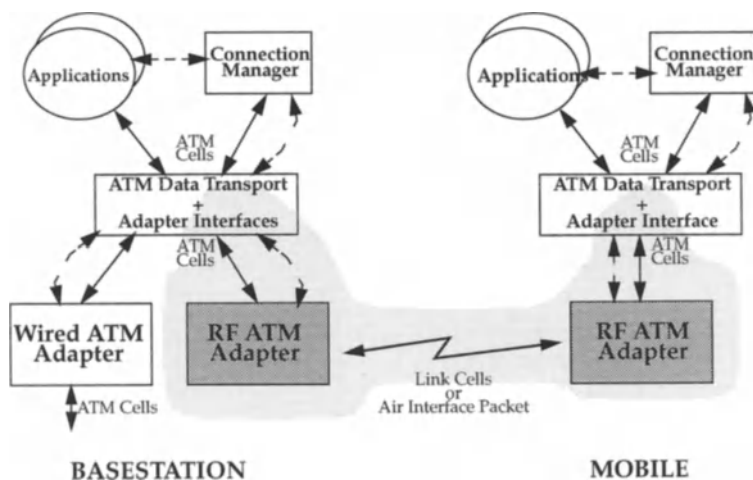


Figure 2. The last hop in a wireless ATM network.

At the other end of the wireless last hop is a mobile that too has a RF wireless adapter, a connection signalling manager module, and a module that routes cells among various agents within the mobile. Although pictorially the mobile may look like a basestation with no wired adapter and only one wireless adapter, this is not entirely true. The Connection Manager at the mobile is different—for example, it does not have to provide a switch-like functionality, and implements different protocol state machines for VC establishment and rerouting. In addition, mobiles such as dumb terminals may have only hardware agents acting as sinks as sources of ATM cells, as opposed to software processes.

The subset of this last hop that is of interest to us is the shaded area in the picture—a stream of ATM cells from the higher level ATM layers needs to be transported across the wireless link between a mobile and its basestation. There are three primary functions that need to be realized at the basestations and the mobiles to support the ATM wireless link: ATM interface, medium access control, and air-interface. The following sub-sections describe these functions.

3.1. *ATM Interface*

At the highest level, the wireless ATM link needs to interface with the upper layers of the ATM protocol stack, in particular with the Connection Manager (CM) which is responsible for VC establishment, rerouting, and tear-down at the highest level. The use of ATM, with its provision of end-to-end per-VC service quality guarantees, means that the wireless link in SWAN cannot be a dumb best-effort sub-system that just handles physical and data link layer details of ATM cell transmission over the air. Rather, it must be cognizant of the ATM layers and continually interact with the upper layers to provide support for per-VC service quality. When the CM requests a new VC to be established, the wireless hop must go through a process of admission control based on the available wireless resources. In SWAN, the CM specifies the bandwidth needed for a VC over a time period (specified by the channel time T_1 needed for this VC over every period of time T_2), and the wireless link ensures that the desired data rate can be supported with the available resources before admitting the VC. Later on, the wireless link must coordinate with CM to enable soft hand-offs of mobiles from neighboring basestations, and participates in signalling to reroute the VCs. Similarly, the wireless link must inform the CM of events, such as hand-offs or

drop in available VC bandwidth, in which a context-adaptive application might have registered interest in. Obviously, providing lower level ATM support for interaction with higher level ATM protocols is an integral part of SWAN's wireless link control functionality. In addition, however, the data transport within the wireless link must also be aware of ATM. For example, it must provide support for any necessary ATM cell queuing, segmentation-reassembly into ATM cells at end-points, and per-VC error control.

Per-VC error control is a novel aspect of SWAN. Since air is a noisy medium (with bit error rates as high as 10^{-5} to 10^{-3}), it is often desirable that there be an appropriate link level error control mechanism. Previous research has shown that relying solely on end-to-end retransmission capability of transport protocols such as TCP is often detrimental to performance in the presence of wireless link errors. Link level error control customized to the requirements of individual connections solves this problem in a transport protocol independent fashion. At the time of VC setup, the CM informs the wireless link whether to do forward-error correction (FEC), or link-level retransmission (ARQ), or both. This allows an application to select a suitable error control and recovery mechanism for a VC depending on the nature of the data traffic. For example, VCs carrying delay critical real-time traffic such as voice typically use FEC, whereas VCs carrying file data transfer tend to rely on ARQ so as to get good throughput at the cost of more delay and delay variance.

Per-VC error control also provides a tangential benefit—a suitable error control mechanism may be selected depending on channel conditions to reduce electrical power consumption when battery capacity is at a premium. For example, FEC schemes require higher up front computation power for encoding and decoding, as well as higher up front communication power due to packet bloat. But the packet error rate is improved due to the FEC encoding. On the other hand, ARQ schemes defer extra electrical power consumption to the time a retransmission is needed. While we have not yet implemented the selection of link level error control to minimize electrical power consumption, this example does point to the possibilities enabled by SWAN's capability of per-VC error control.

3.2. *Medium Access Control*

Like in any other wireless system, the medium access control (MAC) function in SWAN's wireless hop deals

with generic problems of dividing available bandwidth into channels, distributing channels among basestations, regulation of access to a shared channel, and hand-off of mobiles from one basestation to another. In addition, the MAC in SWAN must provide support for per-VC allocation of air bandwidth, otherwise the quality of service guarantees given to applications by the higher layers will be voided.

The details of the MAC strategy depend, partly, on the details of the radio transceiver used. Briefly, SWAN's off-the-shelf radio transceiver, which is described later in Section 5.2, is a 2.4 GHz ISM band slow frequency hopping radio. While communicating, this radio must frequency hop (with its peer) according to a pseudo-random hop sequence at a fast enough rate to satisfy FCC regulations. The frequency hop sequences correspond to communication channels, and pairs of radios that use sufficiently orthogonal hop sequences can operate in the same geographical vicinity with minimal interference. As detailed in Section 5.2, there are 22 distinct channels in SWAN. The current bandwidth sharing strategy in SWAN is to distribute these channels among the basestations in a three-way spatial multiplexing, so that there are up to 7–8 channels per basestation. This is quite sufficient for the initial use of SWAN in individual offices and small meeting rooms, as opposed to large conference rooms. All communicating radios on a channel in a cell hop in synchrony according to the hop sequence, although all data traffic is only between a mobile and its basestation. The time between two frequency hops on a channel is called the *hop frame*, which is sub-divided into *link cells* or *air-interface packets* of fixed length.

Access to a channel is regulated by the basestation which uses the service quality information associated with the VCs to schedule data transfers. To keep things simple while still allowing per VC allocation of shared channel bandwidth, SWAN uses a simple token passing mechanism arbitrated by the basestation. The token represent the privilege to transmit. The MAC modules at the basestations and the mobiles maintain per-VC cell transmit queues, with the cell transmission being driven by a cell scheduler that operates when it has the token. There is a time limit on duration for which the token can be held, and time-outs are used to detect a lost token. In case the token is lost, the basestation takes control and resets the token passing protocol. The token passing scheme uses two fields reserved in the headers of the link cells being transmitted over the air. Details of SWAN's MAC algorithm are available in [19].

3.3. Air-Interface

A physical layer controller, or air interface controller, accepts link *cell* or *air-interface packet* data units from the medium access controller. Using the SDLC (synchronous data link control) protocol, which is widely supported by serial controller chips, the air interface controller packs one or more link cells into SDLC frames before sending them over the air to the receiver. The type and format of Link Cells depend on the MAC protocol in use. One Link Cell type, ATMLC, is used to carry encapsulated ATM data cells. The other Link Cell types are defined to carry MAC protocol signalling messages, and are collectively called MACSIGLCs. The token passing MAC protocol mentioned above defines six different types of MACSIGLCs: CRLC for connection request by a mobile that powers up, HRLC for hand-off request by a mobile, SYNCLC for idle channel, and CHRLCACK1, CHRLCACK2, and CHRLCACK3 for handshake during mobile registration at a basestation. All link cells have a 6 byte header, the format of which is depicted in Fig. 3. The header has an 8-bit statically assigned basestation radio port id (BS_RPID) field, an 8-bit dynamically assigned mobile host radio port id (MH_RPID) field, a 1-bit CELL_TYPE field indicating whether the Link Cell is of type ATMLC or not, 7 bits defined by the MAC protocol, and a 24-bit FEC field that uses a (8,4) linear code to forward error correct the preceding 24 bits. In the case of a non-ATMLC Link Cell, the token passing MAC protocol uses 3 of the 7 reserved bits to disambiguate among the six MACSIGLC signalling Link Cell subtypes. The basestation radio-port id BS_RPID is a logical id statically assigned at set-up such that no two radio-ports in radio vicinity have the same id. This logical radio-port id is mapped by the basestation to the wired network address of the basestation, and the radio-port id within the basestation. Similarly, the mobile host radio-port id MH_RPID is a logical id that is assigned to a mobile host by a basestation radio port when the mobile registers at that basestation radio port. MH_RPID is unique among the mobiles registered on the same basestation radio port. The link cell data payload depends on the type of the link cell, and is the largest in the case of ATMLC link cell where the body contains the 53-byte ATM cell together with error control information. The MACSIGLC link cells have much smaller data payloads.

Efficient mapping of ATM cells to link cells is an important problem in the air-interface part of the wireless

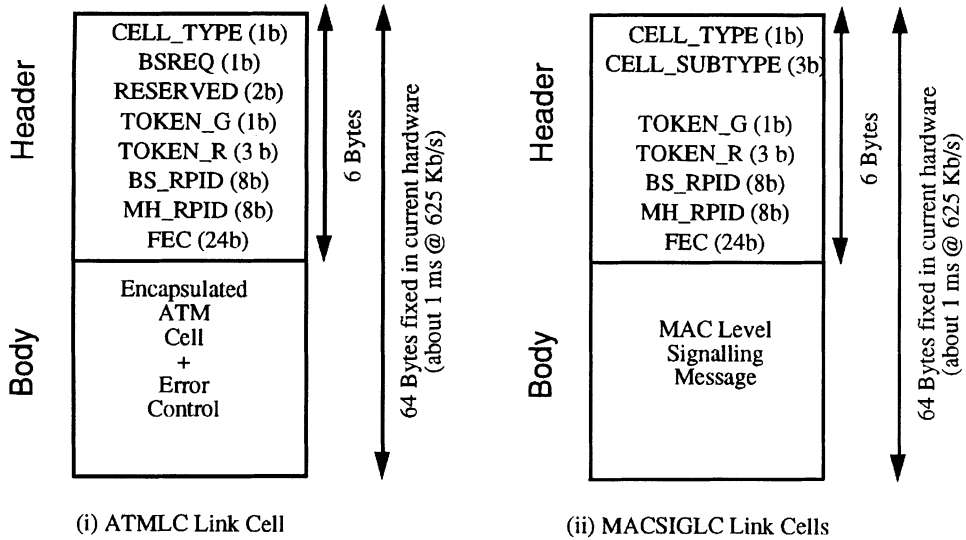


Figure 3. Format of link cells.

hop. While issues such as the ratio of header overhead to data payload are important, the primary constraint comes from hardware. Current SWAN hardware, described later, however provides for fixed 64-byte size link cells so that some bytes are unused when transporting encapsulated ATM cells and small signalling cells.

4. Hardware/Software Partitioning of Wireless Link Functions

The wireless link subsystem at SWAN basestations and mobiles need to incorporate the ATM interface, medium access control, and air-interface functions described in the previous section. While parts of the subsystem, such as the radio and low-level physical control, must be in hardware, the remaining functions which could be implemented either in dedicated hardware, or in embedded software (firmware), or in software on the host CPU (if any). This gives rise to a range of architecture and implementation alternatives for the wireless link subsystem spanning the cost and performance spectrum. Figure 4 shows some of the implementation possibilities with different hardware-firmware-software partitioning.

At one extreme would be a simple and cheap design which kept hardware to the minimum, and which implemented most of the functionality, such as the MAC algorithm and the ATM transport and

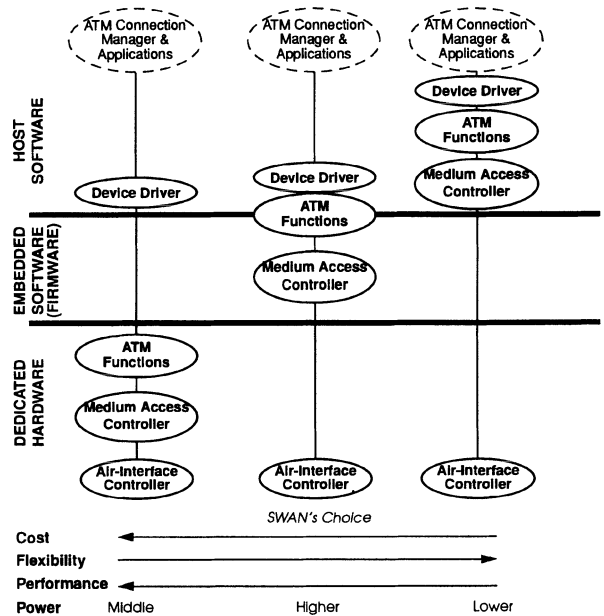


Figure 4. Some hardware/software partitioning alternatives for the wireless link functions.

signalling interface, in software on the host CPU. But CPU horsepower, interrupt overhead, and memory bus bandwidth limitations would hinder the performance of this alternative. At the other extreme would be a design which uses dedicated hardware and ASICs for all the functions, including the necessary ATM signalling and transport. Cost and lack

of flexibility would argue against such an approach, particularly in an experimental system like SWAN where the algorithms themselves are subject to frequent changes.

Fundamentally, however, the wireless link design problem can be viewed as a three-way hardware-software partitioning task. The wireless link functions can be implemented at one of the three places:

1. As kernel-mode or user-mode software on the basestation CPU or the mobile CPU, if any.
2. As embedded software on a processor on a custom wireless link adapter card.
3. As dedicated hardware on a custom wireless link adapter card.

The goal, in the case of SWAN, was to have a low-cost flexible implementation of the wireless ATM link that is not limited by any hardware, software, or operating system bottlenecks in the path to the ATM applications and Connection Manager. The particular hardware-software partitioning chosen in SWAN was driven particularly by the flexibility requirements and the relatively low wireless link data rates of at most a few Mbps. As the following sections describe, the air-interface controller is implemented in reconfigurable hardware on a custom wireless adapter card, the MAC function is implemented on the adapter card as embedded software (firmware) supported by state-machines in reconfigurable hardware, and the ATM functionality is implemented as a mix of adapter card firmware and host software. ATM functions that are closely tied to the MAC, such as VC admission control, cell scheduling, and hand-off signalling reside in firmware on the adapter, whereas ATM Connection Manager functions reside in the host software.

5. Hardware in SWAN's Wireless Hop

From a hardware perspective there are four primary components in the current implementation of SWAN's wireless link: a custom reconfigurable wireless adapter card, the radio transceiver, the basestation (which uses the custom adapter), and the mobile hosts. Generic PCs and Sun workstations are used as basestations by plugging in a wired ATM adapter card, and one or more custom-designed RF wireless ATM adapter cards. The mobiles, at the other end of the wireless hop, include portable computers with an adjunct ATM wireless adapter or multimedia terminals with an embedded ATM wireless adapter. The software driving the wireless link hardware is called *Etherware*, and consists of software modules running on the basestation CPU, on the mobile host CPU, and on an embedded CPU on the wireless adapter card. Following subsections present the various aspects of the hardware implementation focussing on the custom wireless ATM adapter, while the next section describes the software aspects.

5.1. FAWN: A Custom Reconfigurable Adapter Card for Wireless ATM

The wireless hop hardware in SWAN is based around the idea of a single reusable ATM wireless adapter architecture, shown in Fig. 5, that interfaces to one or more digital-in digital-out radio transceivers on one side, to a standard data bus on the other side, and has a standard core module sandwiched in between providing field-programmable hardware resources and a software-programmable embedded compute engine to realize the necessary data processing. Multiple implementations of this basic architecture could be made with differing form factor, different bus interfaces, and

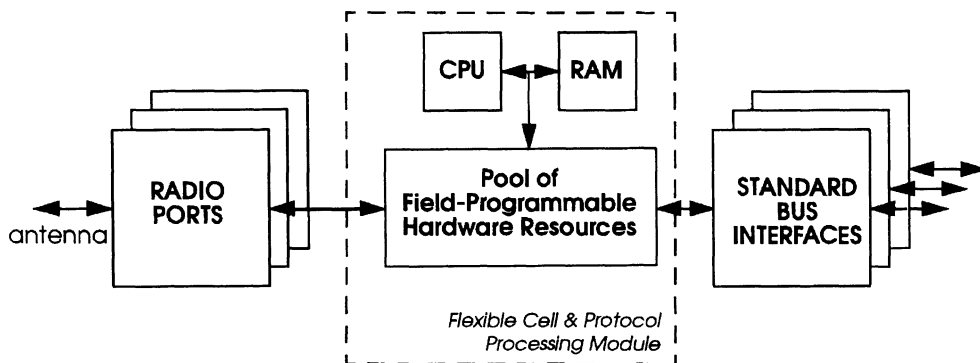


Figure 5. Reusable ATM wireless adapter architecture template.

different radios, but all with the same core data processing module. This provides a uniform mechanism for making devices *SWAN-ready*. Implementations could range from PCMCIA adapter cards that are adjunct to laptop computers, to small-form factor cards for embedding in a wireless terminal, and to higher speed adapters with multiple radios for use in basestations. The adapter could be configured for algorithms by reprogramming the embedded software, and by reconfiguring the field-programmable hardware. System level board synthesis tools with interface synthesis and parameterized library capabilities (such as the SIERA system from Berkeley [20]) can be used to easily generate variations of the basic adapter architecture for different busses and radios.

At present there exists one implementation of our standard architecture in the form of a card called FAWN, for *Flexible Adapter for Wireless Networking* [17]. FAWN interfaces one or more 2.4 GHz band slow frequency hopping radios on one side, to a PCMCIA bus (to interface to laptop computers and to basestations) and a generic peripheral expansion data bus (to interface to peripherals in a dumb terminal setting) on the other side. Another implementation, using MCM technology, is being planned.

The FAWN adapter is responsible for receiving data over the air, processing that data then presenting it to the host computer. Since the planned implementation supported ATM over the wireless link it was decided to present the host computer with queues of data that represent different VCs of an ATM link. The host computer would then be responsible for executing the AAL5 ATM adaptation layer and for routing packets to other FAWN boards or to the backbone ATM network. The FAWN adapter is responsible for processing the data as it goes from the RF interface to a packetized queue that is made available to the host computer. The packet is first received by an RF modem which demodulates the data and provides a serial bitstream. After suitable clock recovery and synchronization, the serial stream is presented as a synchronized bit or byte stream. The packet then has to be reassembled from the bit or byte stream and presented to ATM queue management software which forwards packets to various queues as well as implementing any forward error control, retransmission and error detection algorithms. The queues then make the data available to the next level of processing, the adaptation layers which execute on the host computer. At each stage of the implementation, hardware/software

co-design decisions were made, and these are detailed below.

RF Modem and Serial Data. The packet first has to be received and converted from an RF signal to a bit stream. There are several embedded radios available and we chose to use one of these rather than undertake a design ourselves. By choosing such a radio, improvements in radio technology can be incorporated into latest embedded design. This approach allows a large degree of flexibility in the choice of the frequency band, and initially we chose to use the unlicensed band at 2.4 MHz.

The slow frequency hopping spread spectrum (FHSS) radio chosen demodulated the data and provided a raw bitstream to the remainder of the circuit. Once the bitstream is made available it is necessary to extract the clock so a synchronized version of the bitstream is presented to a circuit that can detect the beginning of the packet. It is possible to implement this function either in a UART or in some programmable logic. While the programmable logic approach allows the use of flexible data encoding schemes that ensure enough transitions to reliably extract the clock, a UART implements acceptable encodings as well as extracts the clock and converts the bitstream into bytes. Therefore we chose to use a UART that implements SDLC encoding, clock recovery and a level of bit stuffing needed to guarantee enough transitions for the RF modem.

Packetization of Bytes. In order to transfer complete ATM cells over the wireless interface the byte stream (from the UART) has to be assembled into packets. One approach is to interrupt the CPU every time a byte is available then the CPU can construct a packet in a memory buffer. At 1 Mbps there are about 125 kbytes/sec (ignoring start bits, sync bits and stuff bits), which means that the CPU would have to handle 1 byte every 8 ms. If the CPU dealt with each byte the interrupt handler would have to deal with an interrupt every 8 ms, which for a 20 MHz RISC CPU would leave 160 instructions to service the interrupt as well as providing cycles to perform the remainder of the tasks required of the CPU. An alternative approach is to implement a buffer in some programmable resources that reads in a complete packet before interrupting the CPU. If 64 bytes are buffered, which is 53 bytes for an ATM cell plus 11 bytes for error correction and MAC headers, the CPU would be interrupted every 512 μ s. A complete

packet can be processed at one time, avoiding the need to read out bytes of data and providing a longer uninterrupted time for the CPU to deal with other functions implemented on the adapter.

Consequently, a packet buffer was implemented in the programmable logic that could read a complete packet before interrupting the CPU. In addition, a second buffer was provided that allowed a second packet to be read while the first was processed, relaxing the constraint of having to read a complete packet out of the buffer within the 8 μ s it would take for a new byte to become available. The resources used to implement this function were provided by an FPGA, and were both flexible enough to allow cell size and format to be changed easily, and fast enough to deal with the data rate from the UART. A final version would probably implement this function as part of an ASIC, avoiding the relatively high cost of an FPGA.

Reading Packets and Error Checking. Once the data is available in a buffer it is read out and processed. Processing includes checking and correcting errors if possible, then placing the packet in the appropriate output queue. The processor has about 0.5 ms in which to process a cell. We chose a reasonably fast CPU, the ARM 610 which is a RISC processor running at 20 MHz to execute the software necessary for these functions. The CPU provides about 10 k cycles in which to process the ATM cell. In addition, the clock of the ARM610 can be slowed down to an arbitrarily low frequency, allowing spare time to be directly turned into a power saving. Conversely, if more cycles are needed, for instance if one was planning to run an embedded application, then the processor clock speed can be increased.

MAC Considerations. The FAWN card is designed to be an embedded unit, so the MAC function was implemented directly on the card rather than on the host. There is still the choice of either implementing the MAC in software or hardware. Because the MAC was being developed as part of the project a high degree of flexibility was needed, and we chose to implement it in software running on the CPU. However, some MAC specific functions were implemented in hardware to improve efficiency, in particular the received signal strength subsystem was implemented in part of the FPGA and interrupts the CPU if the signal strength of the radio transceiver rises above or falls below a value in a register. Because the MAC is quite complex future implementations would also be implemented on the CPU of the embedded processor in the FAWN card.

FAWN to Host Interface. Once the cells have been processed they can be made available to the host CPU. The interface has to support the 1 Mbps data rate as well as be an interface that is available on both portable and desktop machines. This approach allowed the design of just one adapter that can be used to interface to both basestations as well as mobile computers. This made the choice relatively easy since PCMCIA (now PC Card) was the only interface that was readily available everywhere and that gave the required data rate to the machine. Another issue was how to interface the data between the FAWN CPU and the host CPU. The PCMCIA interface is 16 bits wide, and if we interrupt the CPU during a data transfer, assuming a 1 Mbps data rate and ignoring all the extra packing bits surrounding the ATM cell, the CPU would be interrupted once every 16 μ s. If blocks of data were transferred from the memory on FAWN, then the CPU would be interrupted once every 0.5 ms. The alternative that we considered was a dual port RAM (DPR) system that would completely decouple the PCMCIA interface from the FAWN CPU. The DPR also would allow the implementation and manipulation of queues directly in the memory, minimizing the amount of interaction that would have to occur between the CPU and the host system. This approach gave us the greatest flexibility in the software system. While a DPR IO system may be overdesign in a final version it gave us a great deal of flexibility which merited using it in the prototype.

Architectural Constraints. One of the major additional constraints imposed on the adapter design was the requirement that we should be able to use the same hardware in a base station as well as in a mobile host. The PCMCIA standard uses a thin credit-card like form factor which would be an ideal final housing for the FAWN card. However, the prototype uses the PCMCIA interface for power and data, and an external version of the FAWN adapter which makes it easier to debug. Even though an external card was used, the circuit was designed to fit comfortably into the bay of a removable floppy drive.

Because we prefer to assign one radio channel per user, the base station needs to be equipped with a corresponding number of radio transceivers and their support circuitry. This coupled with the requirement that the same design be used in the base station as well as the mobile host imposed a further constraint on the design. The base station can support several FAWN boards using several PCMCIA interfaces, but in addition the CPU on the FAWN card can be used to support

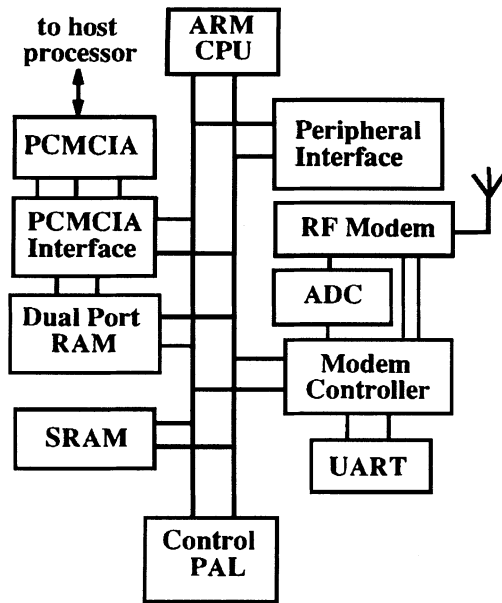


Figure 6. Architecture of the FAWN wireless adapter.

several radio transceivers and the associated packet buffering logic. Implementing the packet buffers in an FPGA provided a natural place to partition the circuit into a CPU and PCMCIA interface part as well as the radio and packet buffer part. Implementing the data buffer and radio modem on a separate card allowed several of these cards to be plugged into a single FAWN CPU card, increasing the number of users that a single PCMCIA interface in a base station could support. Figure 6 shows the architecture of the FAWN adapter, and the photograph in Fig. 7 shows the FAWN card's two parts—the CPU part which contains the

Table 1. Characteristics of FAWN radio adapter hardware.

Dimensions	10.8 cm (W) × 1.9 cm (H) × 11.4 cm (D)
Power of main FAWN cards	2.0 W
Power of radio transceiver	0.6 W (receive)/1.8 W (transmit)
Software resources	20 MIPS, 4 MByte
Reconfigurable hardware resources	1000 Gates equivalent

CPU, PCMCIA interface and memory, and the radio modem part which contains the radio transceiver and the packet buffer logic, which is implemented in an FPGA and is accessible in the address space of the CPU.

Table 1 summarizes the main characteristics of the FAWN hardware.

The power consumption of 2.0 W for the two custom cards constituting FAWN is dominated by two sources: the programmable logic (FPGAs and PLDs), and the dual-port RAM in the host interface. The former reflects the penalty of choosing a flexible and reconfigurable design, attributes considered desirable in SWAN. The latter allowed us to experiment with several different schemes of organizing ATM cell buffers between the host and the FAWN. An implementation that uses ASICs and a customized host-interface memory structure can reduce the power consumption by around one watt.

5.2. Off-the-Shelf Slow Frequency Hopping Radio Transceiver

The FAWN card can be configured to support any radio with serial data in and out capability. Currently, we use

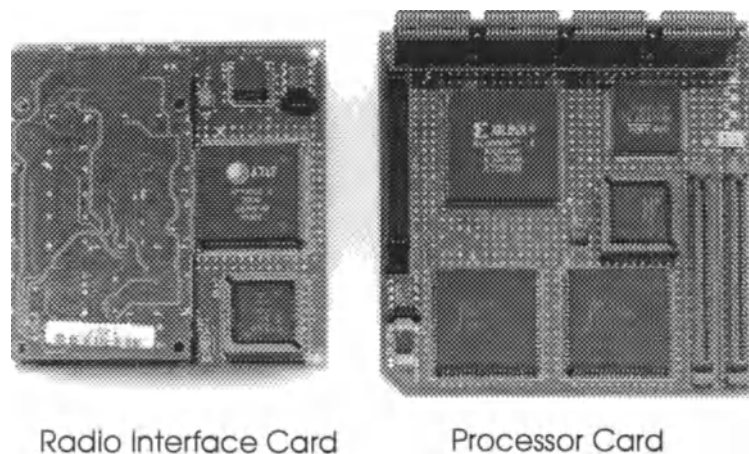


Figure 7. Photographs of the FAWN radio interface card and FAWN processor card.

an off-the-shelf 2.4 GHz ISM band radio with a data rate of 625 Kbps. Although it does not affect the SWAN system per se, the radio is based on slow frequency hopping to multiplex multiple users. Further, the radio has two power levels, and has two selectable radio antennas. Legal requirements dictate that the radio must be operated in such a fashion that it hop pseudo-randomly among at least 75 of the 83 available 1 Mhz wide frequency slots in the 2.400 to 2.4835 MHz region such that no more than 0.4 seconds are spent in a slot every 30 seconds. Communicating transceivers hop according to a pre-determined pseudo-random hopping sequence that is known to all of them.

The slow frequency hopping mechanism suggests that a *channel* in SWAN's wireless hop naturally corresponds to a hopping sequence, or a specific permutation of 75 to 83 frequency slots. Channels co-located in the same geographical area should use hopping sequences such that the chances of two different channels being in the same frequency slot at the same time is minimized (we call such hopping sequences to be *weakly orthogonal*). In SWAN, 22 distinct channels are defined with their own hopping sequences and these channels are then statically distributed among the basestations in various pico-cells. More than one channel can be allocated to a basestation, and a basestation needs to have a separate radio for each channel assigned to it. The same channel cannot be assigned to two basestations in cells that can mutually interfere. The mobiles have only one radios, and at any given time operate in a specific channel.

The hop sequences used in SWAN are of length 79, corresponding to the 1 MHz wide frequency slots number 0 through 78, centered at 2.402 GHz, 2.403 GHz, . . . 2.480 GHz. Following the draft IEEE 802.11 proposal for frequency hopping radios, the 22 weakly orthogonal hopping sequences are defined under the following constraints: (1) the two adjacent frequency slots (at distance $k = 1$) are considered to interfere, and (2) the radio must jump over at least $F = 6$ frequency slots on each hop. The family of 22 hop sequences is given by:

$$\begin{aligned} F_j &= \{f_j(0) \cdots f_j(78)\} \\ f_j(i) &= (i \times j) \bmod (79) \\ i &= 0 \cdots 78 \\ j &= 7, 10, 13, 16, \dots, 67, 70 \end{aligned}$$

In fact, two other families of 22 sequences each also exist corresponding to $j = 8, 11, 14, 17, \dots, 68, 71$ and

$j = 9, 12, 15, 18, \dots, 69, 72$, but only one of three families can be used in the system because while sequences within a family are mutually weakly orthogonal, the property does not hold true across sequences.

The radio provides a bit error rate (BER) of $1E-5$ maximum for operation in SWAN's environment. This translates into a probability of less than 0.5% that an ATM cell will be lost due to noise. While being a much larger loss probability compared to what is easily available on the wired backbone, this cell loss probability is overshadowed due to frequency slot collision in two co-located channels. For example, Monte Carlo simulations show that if $N = 2 \cdots 22$ mobiles in a geographical neighborhood in SWAN are assigned different hop sequences (channels) but arbitrary phases (phase = $0 \cdots 78$ is the starting point in the cyclic hop sequence at time 0), then the percentage bandwidth loss due to frequency slot collision, even when adjacent channel interference is not considered, are: 1.2% for $N = 2$ mobiles, 3.7% for $N = 4$ mobiles, 10.8% for $N = 10$, and 23.5% for $N = 22$. Obviously, cell loss due to frequency collision dominates the loss due to noise, even for $N = 2$ mobiles. Techniques such as information spreading across frequency slots and smart hopping synchronization are to the first order more crucial in SWAN's wireless hop than forward error correction techniques targeted at errors only due to noise. To put it differently, cell loss due to frequency collision is visible to applications as noise with a very different and bursty characteristic compared to the regular RF and circuit noise.

Also relevant to the wireless hop design are the some of the timing parameters associated with the radio transceiver used in SWAN. The radio has a maximum limit of 10 ms on the duration of a continuous transmission, and two periods of such continuous transmissions must be separated by at least $88 \mu s$. This suggests that, at the data rate of 625 Kbps, a maximum of 6250 bits (or 781.25 bytes) can be transmitted in one burst. Therefore, the maximum size of an air interface packet is 6250 bits. Further, the overhead time to switch from receive to transmit mode is $5.8 \mu s$ maximum, and for the switch from transmit to receive mode is $30 \mu s$ maximum. Compared to the $88 \mu s$ separation between two continual transmissions, these two numbers suggest that from efficiency perspective it is better that a transceiver switch its direction after the 10 ms maximum transmission burst. Another timing parameter that results in overhead is the $80 \mu s$ time taken by the radio to hop from one frequency slot to another.

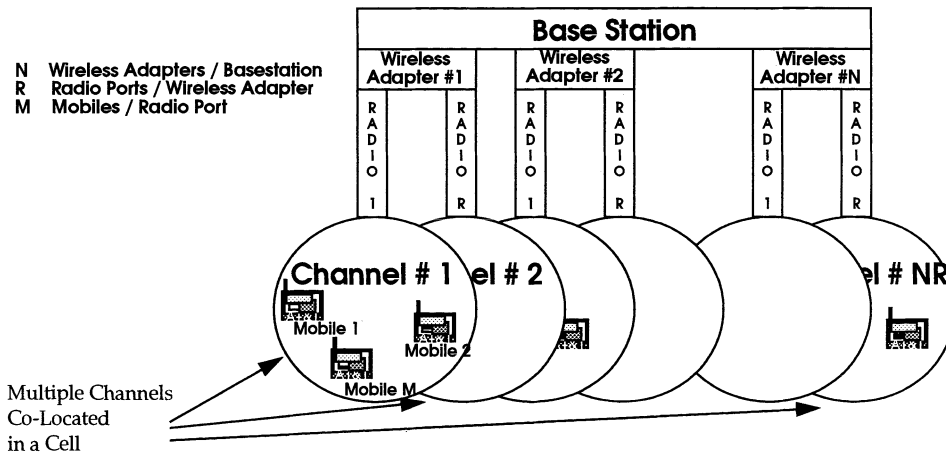


Figure 8. Abstract view of a basestation in SWAN.

5.3. Basestation Architecture

Figure 8 shows the abstract architecture of a typical basestation in SWAN. A basestation consists of multiple wireless ATM adapter cards plugged into its backplane, with each card handling multiple radio transceivers. Each radio transceiver is assigned a channel (frequency hopping sequence) that is different from channels assigned to a radio in the current or neighboring basestation. Typically, in SWAN, a basestation has fewer than 3–5 radios per basestation. The preceding basestation organization results in a cellular structure where each cell is covered by multiple co-located channels. A mobile in a cell is assigned to one of the radio ports on the base station, and frequency hops in synchrony with it.

5.4. Mobile Host Hardware

There are two types of mobiles in the current SWAN hardware. The first type are smart hosts that are built by connecting FAWN cards to off-the-shelf laptops via PCMCIA interface. These hosts have substantial local general purpose computing resources. The FAWN card is designed to fit in the floppy drive bay of generic PC laptops. More interesting are the second type of mobile hosts which are multimedia terminals [18] that follow the dumb terminal philosophy advocated by Berkeley's Infopad [21] and Zenith's CruisePad product. No general purpose computation is done locally. Instead, the functionality, features, and services of the terminal are decided by network based servers in a context dependent fashion. These terminals, called *Personal*

Multimedia Terminals or *PMTs*, have been built by attaching an LCD display subsystem, an audio in/out subsystem, a bar code scanner, and three control buttons to the FAWN via the peripheral expansion bus on FAWN. In addition to the medium access control and air-interface related firmware, the ARM processor on the FAWN card embedded in a PMT also has firmware to terminate ATM VCs at the data source and sink peripherals. Besides being used as a phone and as a multimedia messaging device, the PMT with its bar code scanner is also usable in applications requiring database access such as patient monitoring in hospitals, and inventory management in warehouses.

The necessary circuitry for PMT peripherals reside on a small card that plugs into the peripheral expansion connector of FAWN. Together, the entire assembly together with the radio card is then embedded in the PMT package that holds the LCD screen, buttons, bar code scanning LED, audio ports, and batteries. In addition to the 2.0 W consumed by the embedded FAWN card, and the 0.6 W/1.8 W consumed by the radio (in receive/transmit modes), the PMT consumes 0.57 W.

6. Firmware and Software in SWAN's Wireless Hop

As described in Section 4.0, the implementation of the wireless link in SWAN was viewed as a three-way hardware-software co-design task where the functionality is implemented at one of three locations: as software on the basestation CPU or the mobile CPU, as embedded software on the wireless adapter, and on field programmable hardware on the wireless adapter.

In the case of a dumb terminal with an embedded wireless adapter, there is no CPU in the terminal, so that the entire functionality is on the wireless adapter itself. While the air interface control functions are implemented as dedicated hardware structures on FAWN's field-programmable hardware resources, the remaining functionality of the wireless hop is in firmware and software. The two key software components in the wireless link are the embedded software (firmware) that runs on the ARM610 processor on the FAWN card, and the system software that runs on the host CPU.

The embedded software on FAWN is responsible for the MAC function, and part of the low level ATM queuing, transport, and signalling functions. In the case of the PMT dumb multimedia terminal, the embedded software is also responsible for ATM connection management and PMT peripheral management functions. One constraint on this software is that the parts of MAC functions which interface to the air-interface control hardware must meet real-time requirements—for example, replying to a link cell within a specified time. These constraints are of the order of tens of microseconds to about a millisecond (which is approximately the transmit time for one 64-byte link cell). On the other hand, the constraints on the ATM part of the software are relatively soft and can be taken care of by adequate buffering. With the above requirements in mind, a custom real-time multi-threaded scheduler was used as the underlying platform for the embedded software on the ARM processor. This scheduler supports two classes of threads: interrupt

mode-threads, and user-mode threads, and provides inter-thread communication primitives. The interrupt mode threads are driven by the interrupt events from the air-interface controller, from the dual-port RAM interface to the host, and peripherals (in the case of PMT). The scheduler does not really control the invocation of the interrupt-mode threads, which are used for short and frequent or latency sensitive high priority tasks such as reading or writing ATM cells to the link cell buffer in the air-interface controller. These interrupt-mode threads can also in turn generate less time critical events for the user mode threads, in effect acting as event buffers.

Using the multi-threaded scheduling kernel, the embedded software on the FAWN adapter is organized as shown in Fig. 9. The finite state machines (FSMs) corresponding to the MAC protocol at each radio port are instantiated as interrupt-mode threads, one for each radio port. The MAC FSMs communicate with a main ATM thread that runs in the user mode and handles ATM VC management, cell queue management, and dispatching/scheduling of ATM cells to the MAC FSMs on one side, and to other threads or to the basestation/mobile CPU on the other side. The inter-thread communication is done using queues of pointers, with the ATM cells themselves being stored in a shared memory area. In the case of PMT terminals, with no CPU of their own, the embedded software on the FAWN card also runs an ATM Connection Manager thread which takes care of ATM signalling, and threads that source or sink ATM cells to the PMT hardware.

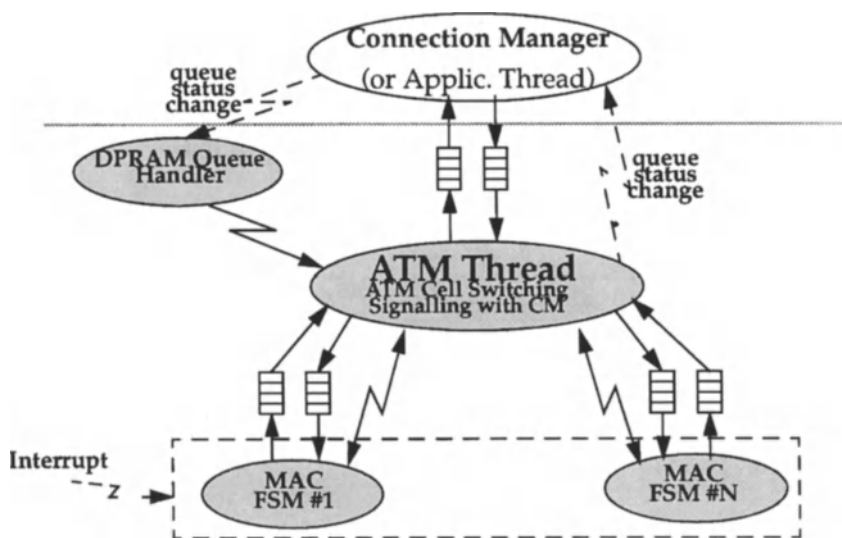


Figure 9. Organization of embedded software on wireless adapter.

To support the embedded software development, a run-time environment is provided in SWAN under Linux and SunOs. Four special device file, `/dev/fawn/mem`, `/dev/fawn/ctl`, `/dev/fawn/cons`, and `/dev/fawn/atm` provide an interface to the ARM memory, a command-based interface to the FAWN control/status registers, a console for `printfs` in the embedded code, and interface to the ATM cells. The first three are meant for debugging and run-time control, while the third is used by the ATM Connection Manager and the ATM applications to receive and send the ATM cells on selected VCs.

There are two ATM specific modules also running on the host CPU: a Connection Manager (CM) module which implements the ATM layer signalling, and a Data Transport (DT) module which implements the ATM cell transport functionality in software. The DT functionality is split between kernel-resident software for ATM cell switching, and user-mode libraries for ATM adaptation layer functionality. In the case of PMTs, which have no local host CPU, the CM and DT functions are realized in simplified form as threads running in the FAWN-based embedded software.

The CM and the DT modules, in interaction with the MAC and air-interface subsystem, implement the necessary address resolution, mobile address allocation, VC establishment, VC rerouting, quality of service support, ATM cell switching, segmentation/reassembly of ATM cells to AAL and IP, and reliable data delivery. The CM communicates, via signalling messages on a predetermined VC, with CM modules on other basestations and mobiles, and also with the local MAC and air-interface module on another predetermined VC. The CM also manipulates the cell routing table in the wireless link software, as VCs are established, rerouted, and torn down, and manages VCs in the Fore ATM switches in our wired backbone via an IP-based RPC interface exported by the switches. Details on CM and DT, and the VC establishment and rerouting algorithms incorporated in them can be found in [5].

7. Results and Experience

Performance Measurements

The wireless ATM link hardware and software described in this paper is functional, together with initial implementation of higher level ATM signalling protocols. Various aspects of the system, such as the cell

scheduling algorithm and error control schemes continue to be refined. Nevertheless, we have ample experience with initial use of the wireless link hardware and software, both with IP-based multimedia applications such as *mv*, *vat*, and *xmosaic*, and with native ATM mode applications such as *netperf* and *mv_atm* (a specially ported version of *mv*). The wireless link has a raw half-duplex bandwidth of 625 Kbps per channel. This is of course a function of the radio being used. In IP-based applications, where we use the spare bytes in FAWN's 64-byte link cells for data as well, we get reliable TCP throughputs of 227 Kbps in each direction (or, 454 Kbps total) in a single mobile case. This throughput measurement was done using the *ttcp* tool using a time division duplex MAC which transferred frames of 10 FAWN data cells. In fact, as shown in Fig. 10, the MAC frame size is a parameter that affects the TCP throughput. The frame size corresponds to the size of the transmission burst between two MAC entities, and smaller frame sizes lead to larger overhead in switching radio modes between transmit and receive. We found that a frame size of 10 cells/frame is quite acceptable with the 2.4 GHz ISM band radio we used; transfer rates drop off rapidly for smaller frames, while much larger frames do not lead to significantly higher data rates.

In true native mode ATM, the wireless link delivers a raw user data throughput (excluding MAC and ATM headers) of 210 Kbps in each direction (or, 420 Kbps total). With header, the throughput is 280 Kbps each way, or 560 Kbps total. Our still un-optimized higher level ATM software is unable to keep up with the 210 Kbps each way user data rate possible on the wireless link, and we get 190 Kbps in each direction on SPARCstation 10 basestations. These user level throughput numbers are end-to-end application level numbers. SDLC

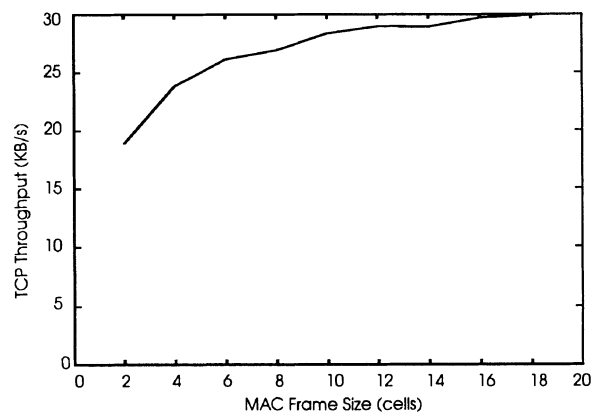


Figure 10. TCP throughput vs. MAC frame size.

frame overhead, signalling protocol implementation in user mode processes, and hardware factors such as fixed 64-byte size link cell buffer in FAWN are responsible for some of the wasted bandwidth reflected in these throughput numbers. The cell loss due to noise was quite negligible, less than 0.25 percent, with a stationary mobile at a distance of 4–5 meters from the basestation. These measurements were done using the performance measurement tool *netperf*.

Architecture Evaluation

It is not possible to directly and meaningfully compare the SWAN system implementation against other implementations because there are few groups who have undertaken an implementation of wireless and mobile ATM, and no one has published hardware and software implementation details. As mentioned earlier in Section 2.0, besides our and other [12, 13] efforts within Bell Labs, NEC's WATMNet [14] and ORL Cambridge's wireless ATM system [15] are the two other concurrent wireless research systems that are implementing wireless and mobile ATM. Comparing research prototypes built with different assumptions and approaches is not meaningful. We, therefore, resort to the following retrospective evaluation of our system architecture.

Architecturally, our approach of using a universal wireless ATM adapter that could be used to make basestations and a range of mobile devices work in SWAN was very successful. In addition, the reconfigurability provided by the firmware and FPGA hardware was instrumental in allowing new protocols and modem controllers to be downloaded without the penalty of host-based software implementations.

However, our implementation approach highlighted several problems. First, the choice of using dual-port RAM based host interface in the wireless ATM adapter proved to be expensive in terms of dollar cost, electrical power and board area, and in addition provided relatively small amounts of ATM cell buffering capability. Since the lower layers of the wireless and mobile ATM protocol in SWAN require maintaining memory hungry per-VC ATM cell queues, the small available buffer space in the dual-port RAM meant having to copy cells into queues maintained in the larger local RAM, thus wasting CPU cycles. An architecture based on the local FAWN RAM being directly shared between the FAWN CPU and the main host CPU via a bus arbiter would be more effective. Second, the FPGAs on

FAWN, while providing reconfigurability and the ability to provide custom data paths to process the ATM cell stream, is expensive and power hungry. Since we have now identified many common and required air interface and MAC functions, it would be better to migrate them to an ASIC. In fact, using the ARM processor cores available from some ASIC vendors, it is feasible to implement the FAWN functionality except for the radio and the RAM on a single chip. Using this, the power consumption could be easily reduced two to four fold. Third, the bandwidth of the PCMCIA bus, while sufficient for the mobile, is not enough on the basestation where a single FAWN card may have several radios. The lack of DMA capability in PCMCIA makes it hard to achieve even the peak bandwidth with programmed I/O. In future, we envisage wireless ATM adapter card optimized for basestation usage by using a faster bus. Fourth, the current implementation has inadequate hardware support to do forward error correction, which is currently done in software. Fifth, we have found that the approach of using fixed 64-byte link cells, while simplifying the double packet buffer hardware, extracts a substantial bandwidth penalty in the transport of small MAC signalling cells, as well as in the transport of individual ATM cells. Support for variable sized link cells, or at least support for a selected smaller link cells sizes is extremely desirable.

Clearly the performance of our system is also limited by the off-the-shelf radios available to us. The current radios are 625 Kbps per channel, and fairly representative of the state of the art in ISM band radios. However, our architecture based on a 32-bit processor and FPGA-based hardwired datapaths is capable of comfortably supporting several such radios, or alternatively a single higher speed radio. Using the latest higher clock rate versions of the ARM processor, the architecture can easily support the higher speed 20 to 25 Mbps ISM band radios expected to emerge in 5 GHz band. As an existence proof of the viability of the processor based architecture at higher speeds, one should note that many 155 Mbps wired ATM adapter cards are also based on processor supported by ASICs. However, reasons such as need for low power may argue for a hardware intensive ASIC approach.

8. Conclusions

In the paper we described the wireless hop of an ATM wireless network called SWAN that has been implemented at Bell Labs. While ATM allows virtual con-

nections to be exploited for meaningful allocation of wireless resources under end-to-end quality of service constraints, it also places demands on the medium access control and physical layer control subsystem. Latency of hand-off becomes crucial, as does the need to schedule ATM cells belonging to different virtual circuits. The architecture of the wireless link, incorporating low level ATM functions, medium access control and air-interface functions, as implemented using SWAN's wireless adapter, was described. The implementation uses a mix of hardware structures in field-programmable hardware and software threads running on the wireless adapter and on the mobile/basestation CPU. On-going work is extending SWAN to accommodate ad hoc networks where no a priori basestation is present.

Acknowledgments

We would like to acknowledge the help and contributions of Abhaya Asthana, Mark Cravatts, and Partho Mishra. Mark helped with the fabrication of FAWN, while Abhaya and Mark created the PMT. Partho has addressed the higher level ATM signalling in SWAN.

References

1. P. Agrawal, A. Asthana, M. Cravatts, E. Hyden, P. Krzyzanowski, P. Mishra, B. Narendran, M. Srivastava, and J. Trotter, "A testbed for mobile networked computing," in *Proceedings of 1995 IEEE International Conference on Communications (ICC'95)*, June 1995.
2. B. Tuch, "Development of WaveLAN, an ISM band wireless LAN," *AT&T Technical Journal*, pp. 27–37, July/Aug. 1993.
3. K. Keeton, B. Mah, S. Seshan, R. Katz, and D. Ferriari, "Providing connection-oriented services to mobile hosts," in *Proceedings of the USENIX Symposium on Mobile and Location-Independent Computing*, Cambridge, Massachusetts, Aug. 1993, pp. 83–102.
4. B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," *Workshop on Mobile Computing Systems and Applications*, Dec. 1994.
5. P.P. Mishra and M.B. Srivastava, "Network protocols for wireless multimedia access," *Proceedings of Workshop on Principles of Multimedia Information Systems*, Washington, D.C., Sept. 1995, pp. 28–30.
6. V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LANs," in *Proceedings of SIGCOMM'94*, 1994.
7. K.-C. Chen, "Medium access control of wireless LANs for mobile computing," *IEEE Network*, pp. 50–63, Sept./Oct. 1994.
8. W. Diepstraten, G. Ennis, and P. Belanger, "Distributed foundation wireless medium access control," *IEEE P802 11-93/190*.
9. P. Karn, "MACA—A new channel access method for packet radio," in *ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, Sept. 1990.
10. D.P. Chandler, A.P. Hulbert, and M.J. McTiffin, "An ATM—CDMA air interface for mobile personal communications," in *Proceedings of PIMRC'94*, 1994.
11. D. Raychaudhuri and N.D. Wilson, "ATM-based transport architecture for multiservices wireless personal communication networks," in *IEEE Journal on Selected Areas in Communications*, Vol. 12, No. 8, pp. 1401–1414, Oct. 1994.
12. Condon et al., "Rednet: A wireless ATM local area network using infrared links," in *Proceedings of the First International Conference on Mobile Computing and Networking*, Nov. 1995.
13. K.Y. Eng et al., "BAHAMA: A broadband ad-hoc wireless ATM local area network," in *Proceedings of 1995 IEEE International Conference on Communications (ICC'95)*, June 1995, pp. 1216–1223.
14. L. French and D. Raychaudhuri, "The WATMnet system: Rationale, architecture, and implementation," in *Proceedings of IEEE Computer Communication Workshop*, pp. 18–20, 1995.
15. J. Porter and A. Hopper, "An overview of the ORL wireless ATM system," *IEEE ATM Workshop*, Washington D.C., Sept. 1995.
16. P.P. Mishra and M.B. Srivastava, "Call establishment and rerouting in mobile computing networks," *Private Communication*, Sept. 1994.
17. J. Trotter and M. Cravatts, "A wireless adapter architecture for mobile computing," in *Proceedings of 2nd USENIX Symposium on Mobile and Location Independent Computing*, April 1995.
18. A. Asthana, M. Cravatts, and P. Krzyzanowski, "An indoor wireless system for personalized shopping assistance," in *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications*, Dec. 1994.
19. M.B. Srivastava, "Medium access control and air-interface subsystem for an indoor wireless ATM network," in *Proceedings of the Ninth International Conference on VLSI Design*, Bangalore, India, Jan. 1996.
20. M.B. Srivastava, B.C. Richards, and R.W. Brodersen, "System level hardware module generation," *IEEE Transactions on VLSI Systems*, March 1995.
21. B. Barringer, T. Burd et al., "Infopad: A system design for portable multimedia access," in *Wireless 1994*, Calgary, Canada, July 1994.



Prathima Agrawal heads the Networked Computing Research Department at Bell Laboratories in Murray Hill, New Jersey.

She presently leads the Seamless Wireless ATM Network (SWAN) research team.

Her previous positions include Distinguished Member of the Technical Staff and Supervisor of AT&T's Microprocessor Design Methodology Group. She led the MARS and PACE multiprocessor architecture and applications projects.

Her research interests are computer networks, mobile computing, multimedia, parallel processing architectures, and VLSI CAD (simulation and test.) She currently serves on the editorial boards of the *Journal of Parallel and Distributed Computing* and the *International Journal of Wireless Personal Communications*.

She was the Program Chair for the 1987 IEEE International Conference on Computer Design (ICCD'87) and the General Chair of the ICCD'88. Dr. Agrawal holds B.E. and M.E. degrees in Electrical Communication Engineering from the Indian Institute of Science, Bangalore, India, and a Ph.D. degree in Electrical Engineering from the University of Southern California.

She has published over 100 papers and has received seven U.S. patents. She is a Fellow of the IEEE.



Eoin Hyden received the B.Sc., B.E. and M.Eng. Sc. degrees from the University of Queensland, Australia, and the Ph.D. from the University of Cambridge Computer Laboratory in England. While with the Systems Research Group in the Computer Laboratory, he worked on operating systems, high speed networks and multimedia systems.

Currently, he is a Member of Technical Staff in the Networked Computing Research department at AT&T Bell Laboratories, Murray Hill, where he enjoys working on mobile computing and multimedia systems



Paul Krzyzanowski received the B.E. degree in electrical engineering in computer science from the Cooper Union in 1985, the B.E.

degree in computer science from New York University in 1985, and the M.S. degree from Columbia University in 1987. He joined the Unix System laboratory at AT&T Bell Labs in 1985 where he worked on distributed file systems. In recent years, he has been working on mobile ATM in the Networked Computing Research Department at Bell Labs. His research interests are in computer architecture, operating systems, and mobility.



Mani Srivastava is a Member of Technical Staff in the Networked Computing Research Department at Bell Laboratories, Murray Hill, NJ. Prior to joining Bell Labs in 1992, Mani received his Ph.D. and M.S. from U.C. Berkeley and B.Tech. from I.I.T. Kanpur. His research interests are in various aspects of mobile and multimedia information systems, including networking technology, end-point architectures, digital signal processing, power management and optimization, and computer-aided design techniques.



John Trotter is a Member of Technical Staff in the Networked Computing Research Department, Bell Laboratories, Lucent Technologies at Murray Hill, NJ. Prior to joining Bell Labs he obtained his B.Sc. degree from Brunel University, U.K. in 1986 and his D.Phil. degree from Oxford University, U.K. in 1990, both in Electronic Engineering. His research interests are in the area of ubiquitous information access and include wireless networked computing, mobile computing and mobile information access. He is also interested in fault tolerance in distributed computer systems.

An Integrated Testbed for Wireless Multimedia Computing

CHARLES CHIEN, SEAN NAZARETH, PAUL LETTIERI, STEPHEN MOLLOY, BRIAN SCHONER,
WALTER A. BORING IV, JOEY CHEN, CHRISTOPHER DENG, WILLIAM H. MANGIONE-SMITH
AND RAJEEV JAIN

*Integrated Circuits and Systems Laboratory, Department of Electrical Engineering, University of California,
Los Angeles*

Received December 12, 1995; Revised April 9, 1996

Abstract. A testbed has been constructed to evaluate node architectures that support multimedia applications and services across a wireless network. Using this testbed, a low bitrate subband video compression algorithm has been prototyped in a field programmable gate array (FPGA) and evaluated for video networking across bandwidth-limited RF channels. A radio interface has been prototyped in an FPGA and a common applications programming interface (API) has been developed to allow experimentation with multiple radios. This testbed has been used to evaluate node performance under two different wireless applications: 1) simultaneous video and data networking (VTALK) and 2) TCP/IP utilities such as FTP and telnet. Based on this evaluation, the design of a battery-operated high throughput wireless multimedia node is presented.

1. Introduction

Nodes for wireless networking require signal processing techniques to achieve bandwidth efficient multimedia communications in the presence of potentially high channel interference and limited bandwidth. Traditionally, system simulation tools have proven extremely useful in evaluating the trade-offs in the design and implementation of these signal processing functions. However, with existing tools, simulation of an entire multimedia node consisting of video codec, digital transceivers, RF front-ends, and network protocols is time consuming. This results in a long design cycle for the node architecture.

Testbeds offer the means to evaluate the performance of wireless nodes in a reasonable amount of time. Testbeds for experimental systems such as SWAN and INFOPAD have been reported in [1] and [2], respectively. Each of these systems represents an important wireless extension to the evolving telecommunication infrastructure: namely, wireless ATM networking and information access on the high speed backbone.

Complementary to these are portable systems that use the existing TCP/IP protocols for internetworking and have sufficient local signal processing power to support not only information access but also multimedia networking over the wireless channel. This paper focuses on a testbed constructed to allow the evaluation of video compression algorithms, radio technology, and network protocols suitable for this type of wireless system. The evaluation is aimed at investigating the affects of adaptation parameters such as compression ratio and data rate on system performance as well as at identifying the key architecture parameters that limit the system performance. Section 3 will summarize the experimental results obtained from the testbed by running two benchmarks—a video-talk application and a data transfer application. Based on the measured results, a node architecture design for the implementation of a high throughput battery-powered mobile computing terminal is presented in Section 4. To facilitate the discussion that follows, we will first describe the functional requirements and the construction of the testbed.

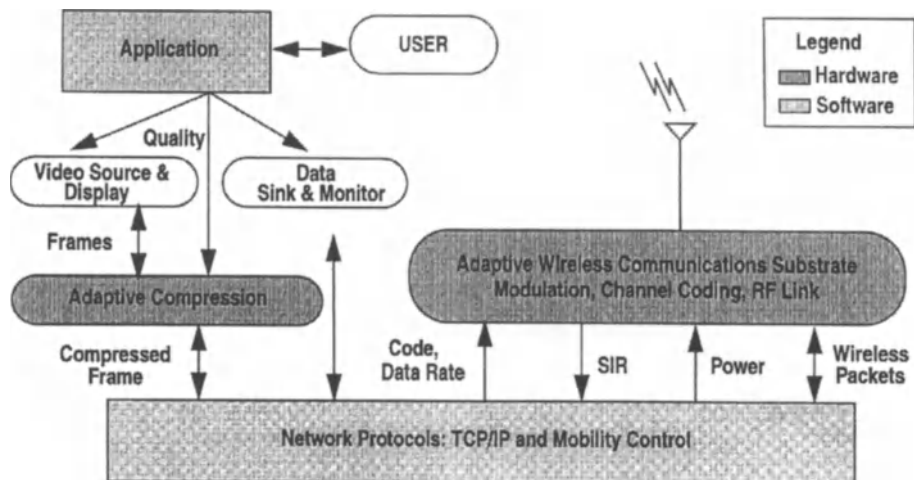


Figure 1. System partition and signal flow graph of the wireless multimedia terminal.

2. Construction of an Integrated Testbed

To evaluate different approaches to the design of a wireless computing node, the testbed must support the basic functionality and signal flow as illustrated in Fig. 1. The top-level block in the node is the application. Through the application, the user executes specific capabilities of the wireless node. For example, to set up a peer-to-peer video link, the application initiates the capture of an image frame and sends it to the compression hardware. The compressed frame is then sent to the TCP/IP protocols for packetization and the packets are subsequently transmitted by the radio sub-system. To enable networking with multiple users, the network protocols sets parameters in the radio sub-system such as spreading codes and frequency channels. In return, the radio sub-system assists the network protocols in mobility control by feeding back signal-to-interference ratio (SIR) estimations of the received signal. The SIR estimates also provide the video application an indication of the channel quality. Depending on the available bandwidth, the video application sets the compressed video bitrate to adapt to the current network capacity and quality of service. To achieve this adaptivity in bandwidth, the underlying compression algorithm must support the ability to adapt the compression ratio while maintaining the picture quality. Similarly, to provide bandwidth adaptivity, the radio sub-system must provide programmability in the data rates.

Figure 2 shows the wireless testbed which is built on a PC platform and consists of a network operating

system (NOS), a radio sub-system, a video sub-system and interfaces to enable the performance measurement of the network while exercising the adaptation capabilities in the compression and radio hardware. The NOS provides the software framework to support the evaluation of network protocols and applications for the bandwidth constrained and interference limited wireless channel. A set of common applications programming interface (API) have been developed to transfer packets and to control adaptation parameters in the radio and video hardware through a set of device drivers. The radio and video hardware are connected to a common bus along with the CPU motherboard and peripherals, such as the keyboard, video display, video graphics adaptor (VGA), and hard disk drive. The bus provides the common medium through which signals and data from the radio and video sub-systems can flow in accordance with Fig. 1.

The radio sub-system of the testbed consists of an adaptation interface prototyped in a FPGA which provides controls to the radio for the selection of transmitted power, spreading codes, frequency channels, and data rate. It also implements an SIR estimator to provide channel quality indicators to the network protocols. As part of the interface, the radio bit stream is packetized/depacketized by a commercial serial interface (SI) card. This interface allowed us to evaluate networking performance with a direct-sequence spread-spectrum radio that has a synchronous serial interface. Commercial radios which have the host interfaces built-in can also be inserted into the testbed via

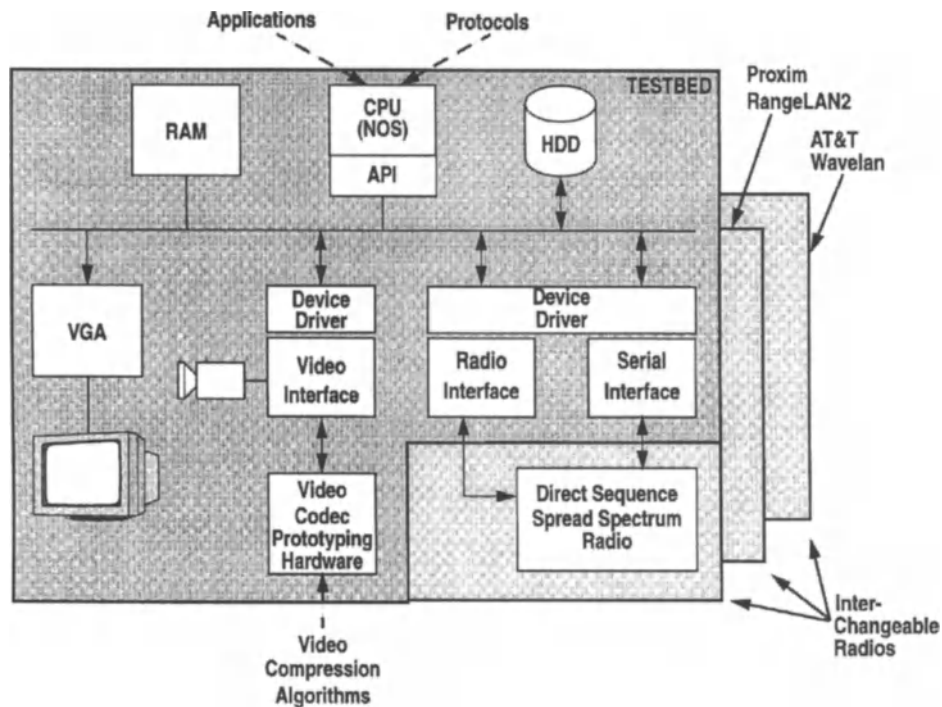


Figure 2. System architecture of the wireless testbed.

the PC bus. The radio interface design in the testbed will be described in more detail in Section 2.2 and the performance of wireless applications evaluated using several different radio technologies will be reported in Section 3.1.

To support the performance evaluation of video coding algorithms, the video subsystem of the testbed contains an FPGA used to prototype video coding algorithms and a video interface used to digitize and buffer image data. Both a commercial and a custom video interface card will be described in Section 2.3 and the performance of a wireless video-talk application that uses an adaptive subband-based video compression algorithm prototyped on the FPGA hardware will be reported in Section 3.2.

2.1. Network Operating System

As shown in Fig. 1, in addition to the hardware required for the two subsystems, software modules consisting of applications and network protocols are needed to harness the functional capabilities provided by the hardware for integral system operation. In the testbed, these software modules are integrated in a NOS derived from the KA9Q operating environment, which traditionally

has supported TCP/IP over HAM radios. This environment as shown in Fig. 3 is flexible and modular, to support rapid prototyping and experimentation with new multimedia and wireless technologies. The NOS provides a socket interface to bridge the gap between the applications and the TCP/IP protocols which combine source/destination headers with user data into packets for transmission via the radio hardware. The mobility control protocols insert additional control information, for example spreading codes for code division multiple access (CDMA) and synchronization headers for time-division multiple access (TDMA). The hardware controls to the video and radio subsystems are executed in NOS through a common software interface provided by APIs to the underlying device drivers.

2.2. Radio Interface Design

Figure 4 shows the application interface between NOS and the radio hardware. The radio API in NOS allows the application to manage the radio link in a hardware-independent fashion. The API encapsulates in software the details of the hardware control functions such as setting the code (*SetCode*) and setting the transmit power (*SetPower*). Device independence is achieved

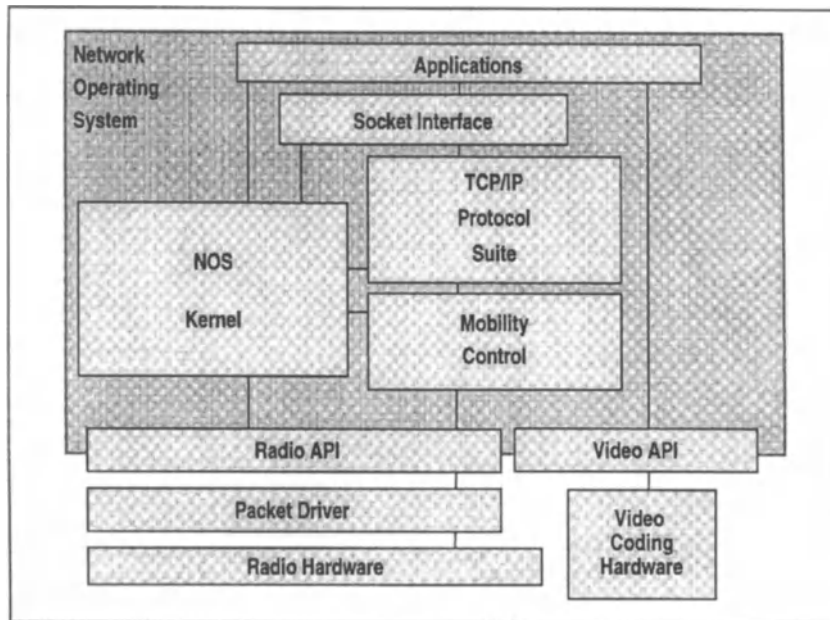


Figure 3. Network operating system.

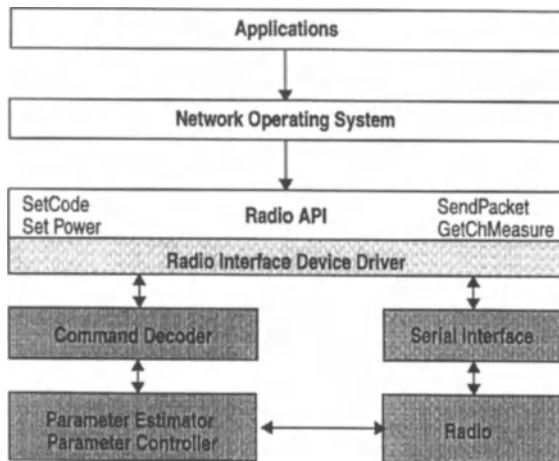


Figure 4. Radio sub-system software interfaces.

through a table that maps hardware specific controls available through the device drivers to the function calls in a ‘C’ interface library. The device drivers are software routines that are customized to meet the control requirements of a specific hardware device. In the testbed, three hardware devices are used to interface with a direct-sequence spread-spectrum radio described in Section 3.1: a command decoder, a parameter estimator/controller (PEC), and a serial interface.

Control data are written to or read from the host bus by the application or the network protocols through the API. Since multiple interface controllers can be on the bus, the data associated with a particular controller is assigned an unique device address. The higher order bits of the control data are used for the device address space and the lower order bits represents the port data. The command decoder decodes the control data based on the device address and writes/reads the port data to/from the appropriate controller. This decoding scheme reduces the complexity of the interface by minimizing the number of control lines between the host and the device. The command decoder requires 30% of a 3 K gate Xilinx FPGA.

The parameter controller in the PEC further decodes the control port data to write the control setting to the specified control port in the radio. The control port data is structured with a 4-bit port address which identifies the control port and an 8-bit data value. This structure requires very little coding/decoding logic and its address space can be easily extended to accommodate radios with more control ports. The parameter controller uses 12% of an 8 K gate Xilinx FPGA.

The parameter estimator in the PEC computes the received SIR based on the soft-decision values from the digital modem within the radio. Soft-decisions are the received signal values before the data decision

has been made in the receiver. The SIR is calculated based on a statistical estimation technique described in [3]. Since the computation requires a 12 bit soft-decision value at a rate of 100–800 kHz, implementing the SIR calculation in software can substantially limit other processes running on the host due to frequent bus interrupts. Therefore, the SIR estimator has been implemented in hardware instead and uses 60% of an 8 K gate Xilinx FPGA.

The command decoder and PEC together provide an interface to the radio for sending controls and estimating channel quality. An interface is also required to send the data packets generated from the TCP/IP protocol stack to the radio for wireless transmission. In the testbed, the data interface is implemented with a commercial SI card which provides synchronous serial data to the radio. The SI card also supports the synchronous data link control (SDLC), CRC error detection, and carrier-sense multiple access (CSMA).

The SDLC and CRC provide the logic link control (LLC) and the CSMA provides medium access control (MAC) necessary for data networking. Commercially available radios are typically built with their own SI, LLC, MAC, and control interfaces. These radios can also be inserted into the testbed provided that their corresponding device driver commands are mapped in the API interface function calls. In Section 3.1, network performance is evaluated in the testbed using both a

commercial radio by Proxim as well as a custom direct-sequence spread spectrum radio.

2.3. Video Interface Design

The adaptation interface for the radio has been designed to allow interfacing with multiple radios for the performance evaluation of different wireless technologies on the testbed. Similarly, the video interface is constructed to allow rapid-prototyping of video coding algorithms on the testbed. Figure 5 shows the video API and device drivers that allow applications to execute commands on the video interface hardware and the video codec. The video interface API provides device independent function calls such as *Capture* to capture an image and *CompressFrame* to send the captured image to the video codec. The device drivers translate the API calls into hardware dependent CPU instructions to control the FPGA video codec prototyping hardware, the frame grabber interface, and the command decoder. In the video interface, the command decoder is used to control the compression ratio in the codec according to bandwidth requirements set by the network through the NOS.

2.3.1. FPGA Prototyping Hardware for Video Compression Algorithms. The FPGA prototyping hardware relies on an external 256 kB SRAM to simplify

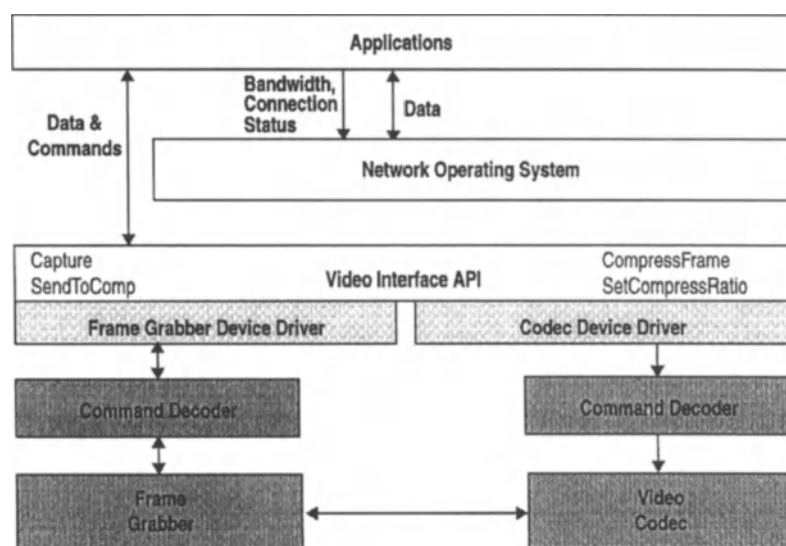


Figure 5. Video sub-system software interfaces.

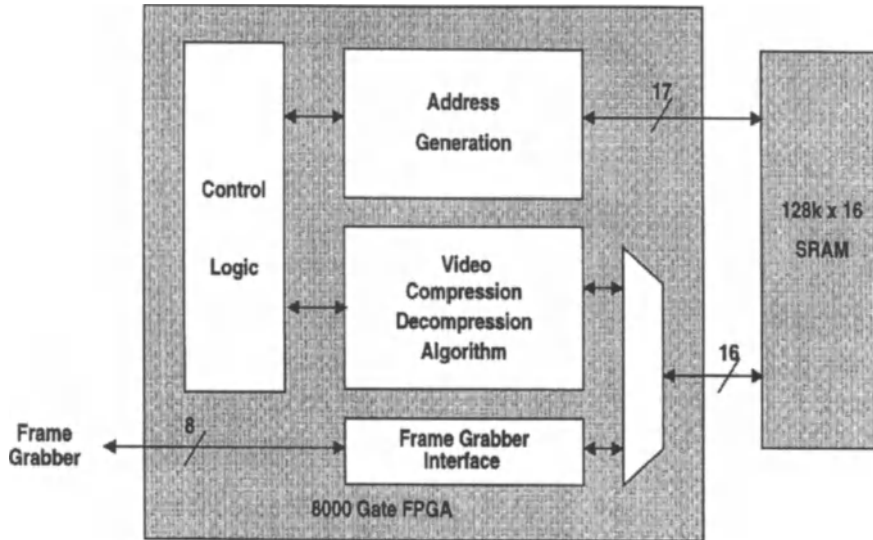


Figure 6. FPGA hardware for prototyping video compression algorithms.

the interface of the video codec to the host, camera, and display. The architecture for the FPGA prototyping hardware is shown in Fig. 6. The SRAM interfaces to the host via a frame grabber interface which consists of a simple handshake mechanism using a strobe and ready a signal to transfer 1 Byte of image data at a time. The specific processing associated with the compression algorithm is transparent to the peripherals that must interface to the codec. The address generation block produces a 17-bit address to the portion of the SRAM used by the compression/decompression algorithm. Any processed data is written back into the SRAM two bytes at a time to a location specified by the address generation block. The control logic is a finite-state machine that sequences through the operations specific to the video compression algorithm.

2.3.2. Frame Grabber Interface. The external SRAM allows the storage of a 256 kB of image data, large enough to store a 256×256 8-bit grayscale image from the frame grabber. Depending on the application, the frame grabber needs to support three types of operations: image capture, transfer of compressed image data to NOS, and display the uncompressed image. A flexible architecture which supports any combinations of the above operations is shown in Fig. 7. The frame grabber accepts a standard analog video input and produces digitized video data. The performance related variables include the number of frame buffers and the mechanisms available to access the buffers. For the testbed, both a commercial and a custom frame grabber

have been used. The two implementations differ in the management of the video buffers, the speed of the operations, and the control overhead. These differences result in significantly different system performance as described in Section 3.3.1.

3. Performance Measurements

The NOS, video and radio interface described in Section 2 have all been integrated into a testbed shown in Fig. 8. Two spread-spectrum radios and an adaptive subband video codec have been evaluated in the testbed based on their performance in supporting an FTP and a video-talk application (VTALK). These experiments have been performed by inserting analysis code in NOS to record timing statistics of the relevant node functions. The measured results are used to characterize 1) the effect of network processing and host CPU processing overhead on packet throughput, 2) the effect of MAC protocol overhead on throughput, 3) the performance of adaptive video compression, 4) the limitations imposed by the transfer time in the time shared PC bus, 5) the effect of memory copies, and 6) power dissipation of the overall system.

3.1. Radio Link and Medium Access Control Performance

Radio transmission has traditionally been implemented with narrow-band techniques which tend to be sensitive

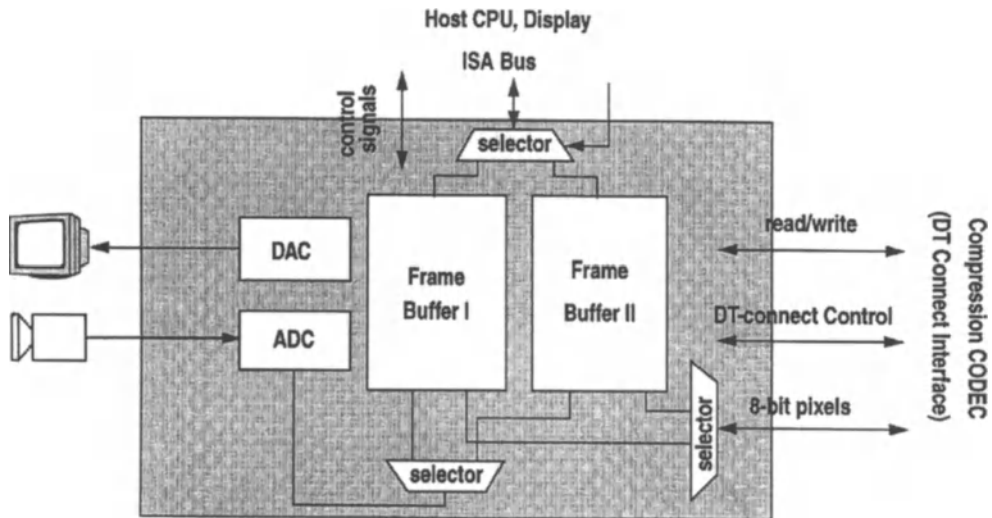


Figure 7. An architecture for frame grabbers.

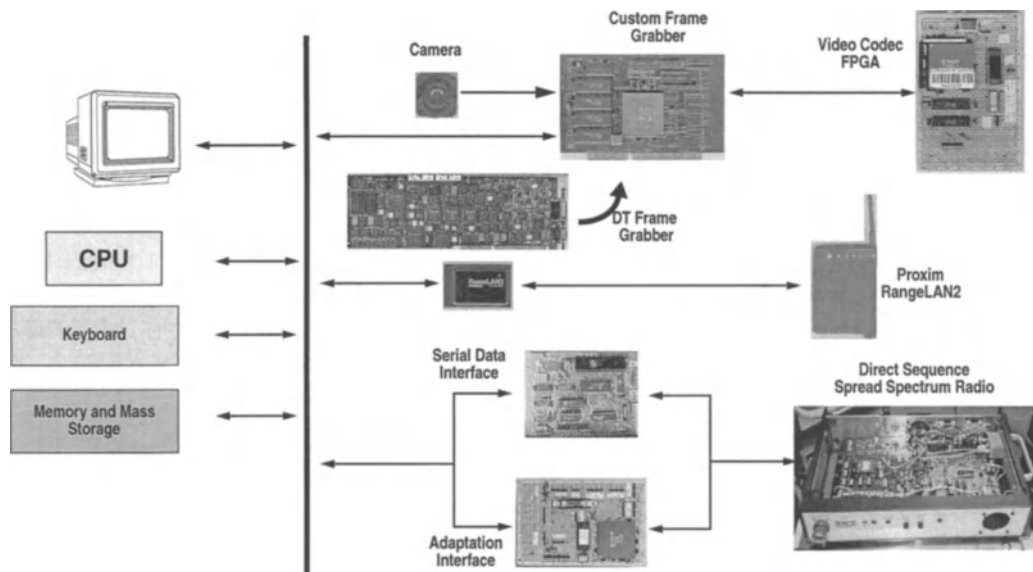


Figure 8. The integrated testbed.

to co-channel interference and multipath fading commonly experienced in a wireless channel. Spread-spectrum has emerged as a technique which promises to alleviate the multipath fading and co-channel interference [4, 5]. It uses a pseudo-noise (PN) spreading code, which is uncorrelated with the transmitted data, to provide increased separation among users in a multiple access system to mitigate degradations due to multipath fading. In the testbed, network performance in the physical and MAC layers has been evaluated using

two spread-spectrum radios: (1) a commercial Proxim frequency-hopped radio and (2) a custom built direct-sequence radio.

Figure 9 shows the direct-sequence radio architecture which consists of a single-chip all-digital direct-sequence spread-spectrum modem [6] and a custom RF front-end built from *commercial-off-the-shelf* (COTS) technology [7]. The transmitter modulates the bit-stream with a Gold sequence in the digital transceiver IC and directly upconverts the bandlimited *binary*

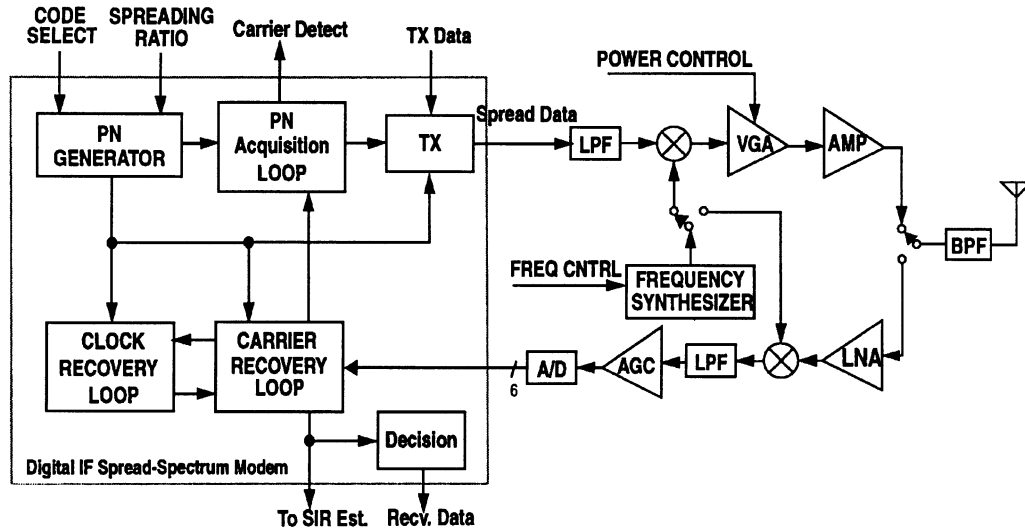


Figure 9. Architecture of a direct-sequence spread-spectrum radio.

phase shift keyed (BPSK) baseband signal to the 900 MHz ISM band. The receiver down-converts the RF to IF which is digitized and processed digitally by the modem IC. The all-digital architecture makes the adaptation of codes, spreading factor and bitrate very simple. Figure 9 highlights the control points to vary the spreading codes, spreading factor, bitrate, power setting, and frequency channels. The following section discusses the effect on network performance as these parameters are varied.

3.1.1. Network Performance with Varying Bitrate and Spreading Ratios. The spreading ratio and bitrate in a direct-sequence spread spectrum radio is related by the chip rate:

$$f_{\text{chip}} = N_c R_b \quad (1)$$

where N_c is the length of the PN-code spanning over a data bit duration and R_b is the bitrate. For a BPSK modulation scheme, N_c equals the spreading ratio which measures the effectiveness of spread-spectrum against channel interference. Measured in dB, it is commonly known as the processing gain (P_G). The capacity of a spread-spectrum system using CDMA is

$$C = K 10^{0.1(P_G - E_b/N_0)} \quad (2)$$

where K is a constant that depends on system implementation [8].

The processing gain determines the ability to adapt to varying channel *bit-error rate* (BER) according to (2) and capacity requirements and the chip rate determines the range of bitrates that can be achieved according to (1). A higher processing gain increases capacity and/or reduces the BER at a given data rate. To adapt to different BER requirements for information with different bitrates, the spreading ratio, N_c , can be changed to vary the processing gain while keeping the chip rate constant. For instance, BER can be traded for bitrate. This approach allows video applications, which require low latency and tolerate higher BER (e.g., 10^{-4}), to use short PN-codes, while data transfers, which is insensitive to latency but requires lower BER (e.g., 10^{-5}), to use longer sequences. Table 1 summarizes the adaptation parameters for the direct-sequence spread-spectrum radio. Note, the chip rate is selected to be 1 Mchips/sec to split the 26 MHz bandwidth available in the 900 MHz ISM band into ten 2 MHz frequency channels.

The number of channels directly affects the capacity gain constant (K). To improve capacity CDMA can be

Table 1. Adaptation parameters for the direct-sequence spread-spectrum radio.

Radio parameters	Value
Chip rate	1 Mchip/sec
Burst data rate	$1/N_c$ MHz
Chips per bit (N_c)	15, 31, 63, 127

utilized. According to (2), for a processing gain of 21 dB and $K = 1$, the capacity is only 12.7. To improve the capacity, a combination of TDMA and FDMA can be used. With sufficient processing gain, the time slots and frequency can achieve a re-use factor of approximately one [8]. Therefore, K becomes $N_s N_f$ where N_s is the number of time slots and N_f is the number of frequency channels. With four time slots and ten frequency channels, the direct-sequence radio can provide a maximum system capacity of 500.

Throughput Analysis. The performance of the network throughput due to variations in the spreading ratio and bitrate has been evaluated in the testbed. Using a file transfer application based on FTP, analysis is performed by tracing the activities in the software as well as hardware associated with the transmission and reception of a wireless packet as shown in Fig. 10. The FTP and the radio API are part of NOS and the Packet Driver is the device driver which provides the direct software interface to the radio hardware. The received bit stream is put in host memory by the SI card and the packet driver sends a completion flag when the transfer completes. The *Buffer Time* in Fig. 10, represents the time required for the radio interface API to queue the received packet stored in memory from the time the completion flag is issued by the packet driver. The *App Time* is the time spent by FTP to process a packet off the receive queue and to push an acknowledgment or a data packet in the transmit queue.

Table 2. Breakdown of FTP time usage.

	8 kbps (127 cpb)	16 kbps (63 cpb)	32 kbps (31 cpb)	64 kbps (15 cpb)
Buffer	0.02%	0.05%	0.09%	0.14%
Application	0.18%	0.34%	0.71%	1.15%
Send	0.06%	0.16%	0.27%	0.43%
Preamble	5.68%	3.23%	1.98%	0.98%
Header				
Trailer	3.44%	3.50%	3.37%	2.61%
Pkt. loss	0.1%	1%	5%	25%
Propagation	90.52%	91.72%	88.58%	69.69%
RTT	4491 ms	2186 ms	1121 ms	698 ms

The *Send Time* is the time required for the packet driver to begin sending packets from the time the API generates a request to send a packet. The *Propagation Time* is the time spent in the channel. Table 2 shows the relative duration for the *App Time*, *Send Time*, *Buffer Time*, and *Propagation Time* as well as overheads incurred by retransmission due to packet loss, headers/trailers used for control, and the preamble used to allow for synchronizing on the PN-code. The round trip time (RTT) is the sum of all the times in Table 2. Each item in the table is an average per transmit/receive cycle over all the packets transmitted during an FTP session. The packet size used in this experiment is 4 kB.

As expected, the measured results show that as data rates decrease with increasing spreading ratio, the

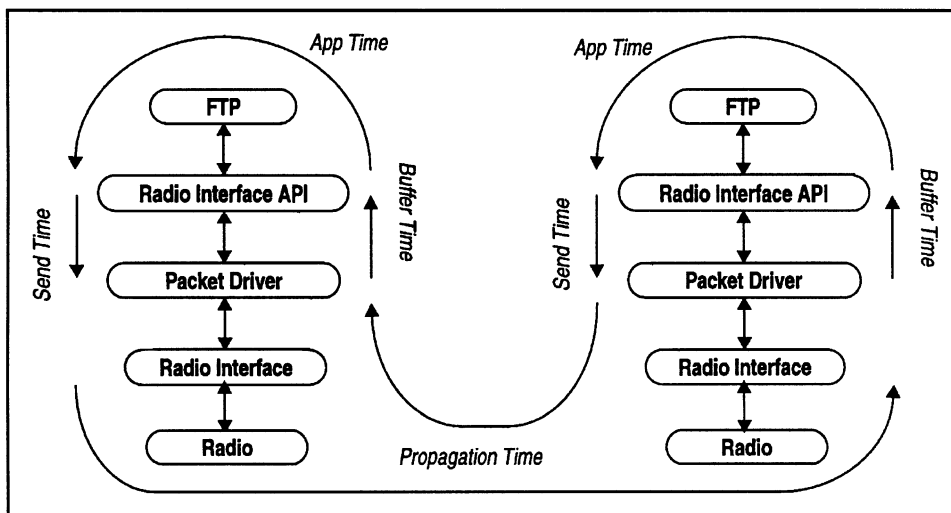


Figure 10. Layers in the FTP transfer.

system becomes more robust to bit errors and the applications and network overheads diminish relative to the longer time required to transmit at lower bit rate. The application and network overheads are the *Buffer Time*, *App Time*, and *Send Time*. The headers and trailers contain controls for the radio parameters in Table 1 and link layer CRC. Since the control information is in general fixed, the overhead incurred decreases as the data rate decreases. The synchronization preamble, however, shows the opposite trend because the serial acquisition correlator used in the modem has an acquisition time which is proportional to the spreading ratio [9]. For non-real time applications, this can be compensated by having a longer packet. However, for real-time applications this overhead must be shortened either through a matched filter at the cost of more hardware or a hybrid serial-parallel search scheme which compromises acquisition time for less increase in hardware.

Throughput Analysis Using Proxim RangeLan2. The testbed has also been used to evaluate the network performance with a Proxim RangeLan2 radio that uses a different spread-spectrum technique, slow frequency hop. The purpose is not to compare the two different spreading techniques but rather to analyze the overhead incurred by applications and network processing at a substantially higher bit rate of 1.6 Mbps. The results show that the average RTT is 38 ms with 1.5 kB of payload per RTT which results in 316 kbps throughput or 20% of the available bitrate. The detailed breakdown of the overhead is not possible because the internals of neither the packet driver nor the radio is accessible. Measured results show 10% of the overhead is spent in the application and 70% in the built-in CSMA/CA MAC protocol, and the LLC controls. Therefore, the overhead exacted by the CSMA/CA MAC protocol can be quite significant even for a peer-to-peer link of only two users.

3.2. Subband Video Codec Performance

To achieve networking of high bandwidth video data in a wireless system characterized by a limited throughput and a noisy channel requires a compression algorithm which achieves low bitrate while being able to adapt to the varying available bandwidth. Traditional compression techniques based on image-domain vector quantization (VQ) have been studied extensively and implemented in the wireless system reported in [10]. VQ results in simple receivers but requires high

bitrate for high quality video. Using the block DCT and motion estimation, H.263 provides good quality at bitrates below 64 kbps. However, the use of motion estimation, leads to complex, high-power implementations [11], as well as error propagation across frames due to the interframe nature of the coding. Subband coding is a promising technique for providing adequate image quality when intraframe-only coding is used at low bitrates, and is the basis for the systems described in [12] and [13]. The mode of degradation at low bitrates is an overall softening and loss of detail in the image, rather than the more objectionable blocking seen with intraframe DCT (JPEG) and image-domain VQ schemes. In the testbed, a full-frame multi-resolution adaptive subband video codec has been evaluated for its network performance in a video-talk application.

3.2.1. FPGA Implementation. Figure 11 shows the subband codec prototyped in an 8000-gate Xilinx FPGA operating at an 8 MHz clock rate. The FPGA can be reprogrammed while the system is operational which allows one device to support both image compression and decompression. An ISA bus interface allows the computer to set the compression ratio and the compress/decompress mode through the command decoder. Video images are acquired from a camera through a frame grabber and stored in the SRAM.

The FPGA implements a four level subband decomposition which divides the image into 13 frequency regions (i.e., subbands). The subbands stored in the SRAM are transformed by recursively applying high- and low-pass filters in both the horizontal and vertical directions. Address generation circuitry manages access to individual subbands within the SRAM. Each transformed subband is then quantized and sent to the run-length encoder (RLE) as depicted in Fig. 12. Run-length coding compresses the quantized subbands.

The RLE simply counts the number of zeros between non-zero quantized values. The hardware guarantees that the pair of run-length and quantized values will be 8-bits long. For example, if a subband has 4 bits allocated for quantized values, it will have 4 bits left for run-length coding, and thus a maximum run-length of 15. The choices of fixed quantization step size, thresholding, and fixed 8-bit pairs are all made to reduce hardware complexity. A total of eight settings are available to adjust the quantization step sizes to adapt the compressed bitrate given the available bandwidth.

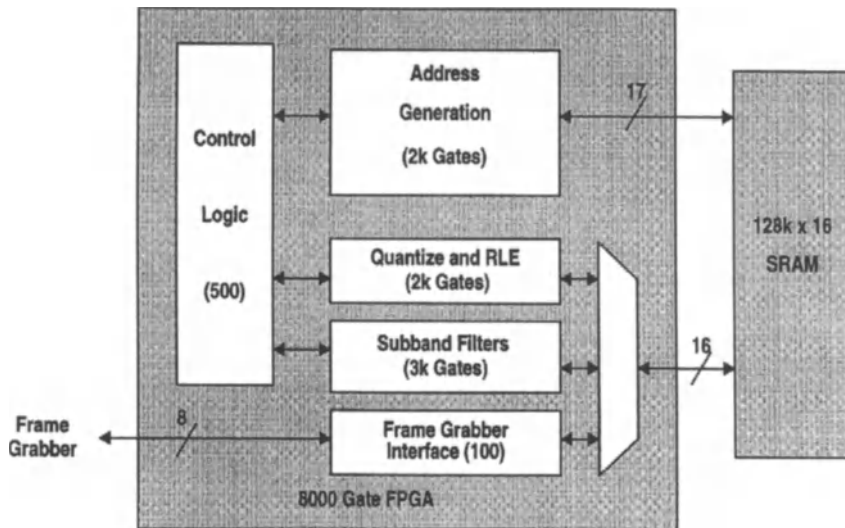


Figure 11. FPGA prototype of the adaptive subband video codec.

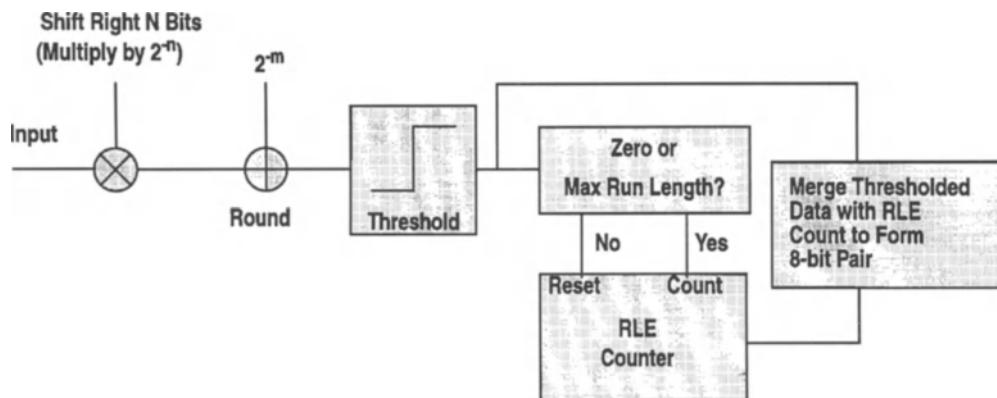


Figure 12. Quantization and run-length coding.

3.2.2. Performance Measure of Adaptive Compression. The performance of the bitrate adaptation for the subband algorithm has been evaluated on real-time captured image sequence. Figure 13 shows the compression range for a sequence of images captured in an indoor office environment. Each data point represents a snap shot of the office as the camera is moved randomly. The upper curve shows the compression ratio for the highest compression setting and the lower curve for the lowest compression setting. As expected the compression ratio varies with image content. However, the compression ratio averaged over a slowly varying scene can show fairly distinct compression ratios for the eight different rate settings as shown in Fig. 14. This set of

data is obtained by moving the camera slowly in an office environment. The monotonic trend between the compression setting and compression ratios makes it possible to change the output bitrate by adjusting the compression ratios based on the average of compressed image size from previous frames. Compression ratio varies from 10 to 20 over the eight control settings.

Since the image quality scales with the compression ratio, the bitrate adaptation requires that the picture quality scales with the compression ratio as well. A monotonic relationship between bitrate and picture quality allows the network to maintain a graceful variation in the picture quality as it adapts the bitrates to maximize the quality of service. A non-subjective

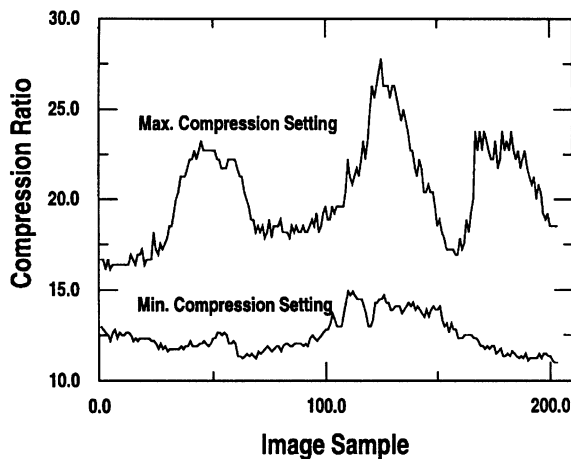


Figure 13. Range of compression ratios for a rapidly varying image source.

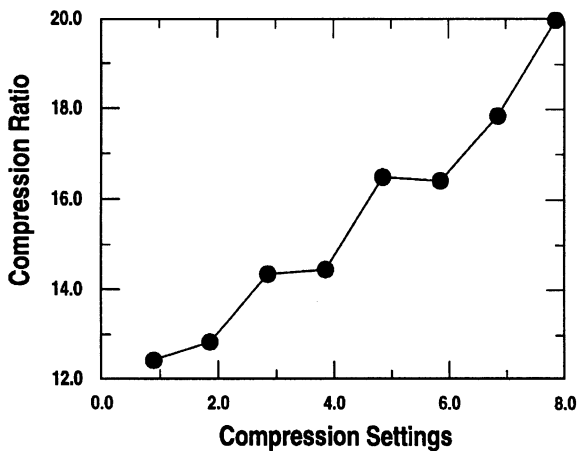


Figure 14. Averaged compression ratio for different compression settings.

measure of picture quality is the peak signal-to-noise ratio (PSNR) which measures the mean squared error between the original image and the reconstructed image. Figure 15 shows the PSNR of a standard test image and a captured image of a talking head. The trend is generally monotonic and ranges between 26 dB to 33 dB for the eight settings.

The prototype has served to evaluate the performance of a subband codec with a variable bitrate on real-time captured images. Since only eight compression settings are available, the incremental step size in the compression ratio and PSNR are rather coarse. Additional improvements using entropy coding and fine grained adaptive quantizations can be implemented in a working video codec to provide two-fold increase in compression ratio and fine stepsizes for adaptation.

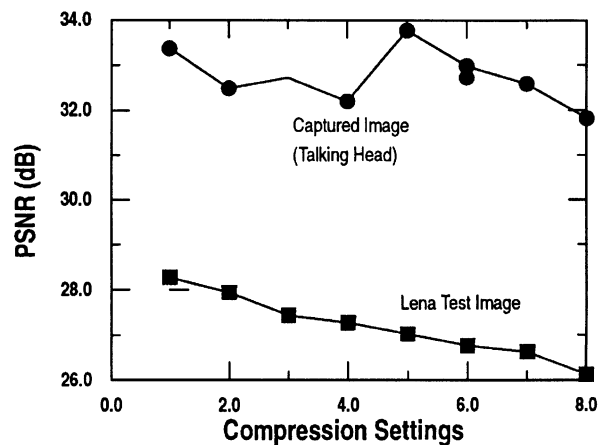


Figure 15. PSNR of two test images as a function of compression settings.

3.3. Video and Data Networking Performance

The overall system performance for VTALK using the prototype codec has been evaluated in the testbed. VTALK allows peer-to-peer networking of real-time video together with a text-based talk in a *time division duplexed* (TDD) fashion. The UDP protocol is used to send and receive video frames, while the TCP protocol is used for the text and data transfer. The frame grabber captures an image from the camera and sends it to the video codec. The compressed image is passed back from the video codec to the frame grabber and finally to the host memory. Once in the main memory, the compressed data is passed down the network protocol stack to the packet driver which packetizes the compressed image in the SI card. The radio then transmits the packets. The reverse sequence of operations occurs during the receive mode.

The time required for each operation which occurs in VTALK has been measured. Table 3 lists the percentage time utilization for these operations using a commercial and a custom frame grabber. Since in this analysis, the main concern is in characterizing the video hardware and software time utilization, the effect of radio is not shown but has been evaluated in a separate analysis described in Section 3.1.

With the commercial frame grabber, the required time for a complete send/receive cycle is 682.9 ms and with the custom frame grabber this improves slightly to 531.9 ms. With an ideal transmission rate of zero, the first results in a frame rate of 1.5 fps while the latter lead to 1.9 fps. Both cases are clearly too slow compared to real-time video frame rate at 15–30 fps. The

Table 3. Time utilization per frame during VTalk image processing.

Operations	DT interface	Custom interface	Speed up
Bus transfer	26.34%	17.30%	1.96
Memory copy	11.13%	14.29%	1.0
Mode changes	29.29%	37.60%	1.0
Frame capture	13.18%	5.08%	3.33
Display to screen	6.44%	8.27%	1.0
Compress	4.25%	5.45%	1.0
Decompress	4.25%	5.45%	1.0
Frame to packet driver	0.79%	1.00%	1.0
Generate statistics	4.33%	5.56%	1.0
Total time	682.9 ms	531.9 ms	1.51

reasons for the reduction in frame rate are discussed below.

3.3.1. Shared Bus Transfer. The most costly operation is the bus transfer which consumes 17% to 26% of the time. In the time-shared bus architecture, the control and data must pass through the ISA Bus as is shown in Fig. 16. The measured result accounts for the time required to move large blocks of image data across this shared bus between the frame grabber, the host memory, and the VGA. The sequence of operations performed by the transmit node is shown in Fig. 17. The bus transfer occurs in the display of the captured

image where the frame grabber (FG) captures a 64 kB image and transfers it to the host memory and to the VGA for display. To transmit the compressed image, the FG transfers the 2 kB compressed data to the host memory for network processing. Similar transfer operations occur in the receive node as shown in Fig. 18.

Although the ISA-bus is rated at 8 MBps peak transfer rate, actual throughput across the bus is much lower because of the control overhead of processes running on the host. For the commercial data translation (DT2867) frame grabber, the average transfer rate is 1.2 MBps which is only 15% of the peak rate. The general-purpose architecture of the DT frame grabber also contributes to the low peak rate. For example, DT2867 offers the application control over which buffer to use for captures, transfers, and displays. However, this flexibility requires additional cycles in the bus transfers resulting in a slower frame rate.

Overlap of bus transfers can alleviate the performance degradations due to slow transfers. The transmitter operation sequence (Fig. 17) shows that an overlap of the *FG-VC-compress* operation with the *FG-host RAM-VRAM* bus transfer can improve the performance of the application. However, since the DT interface does not allow simultaneous transfer out from its port, the bus transfer required for display cannot be overlapped with the memory copy required by the VC. A custom frame grabber interface (Section 2.3) has been designed to overcome this problem. Figure 19 shows the overlapped bus transfer. This improves the

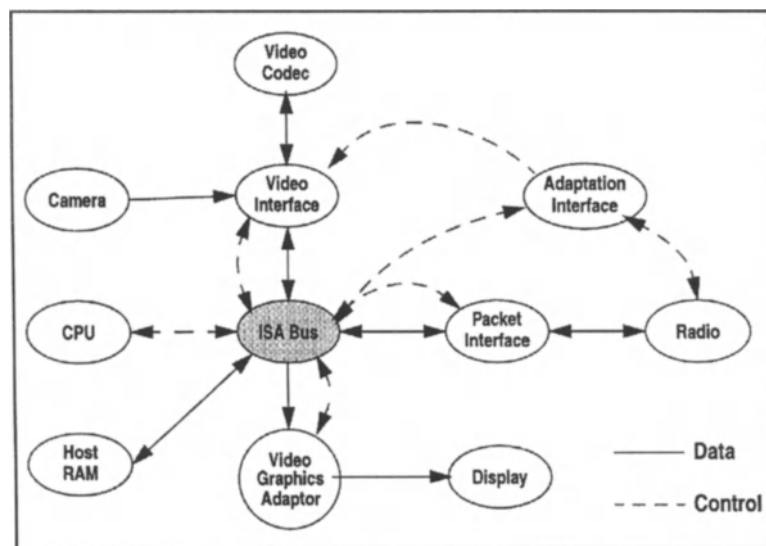


Figure 16. Control and data flow in the ISA-bus architecture.

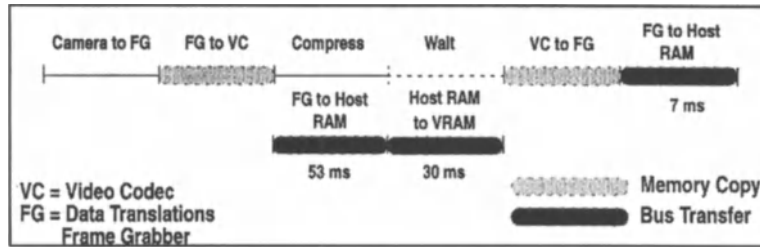


Figure 17. Video operations in the transmit mode for DT frame grabber.

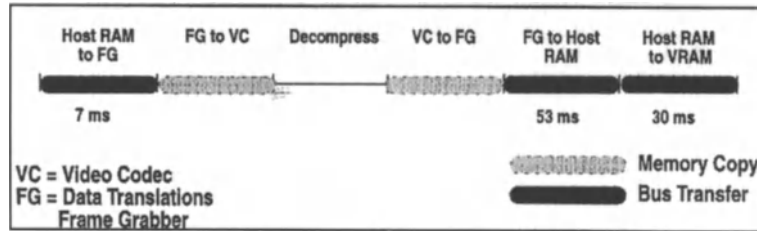


Figure 18. Video operations in the receive mode for DT frame grabber.

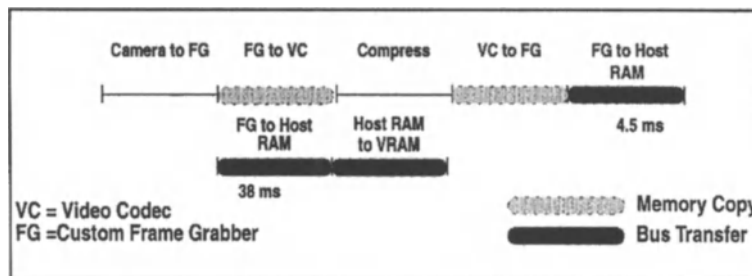


Figure 19. Video operations in transmit mode with custom frame grabber.

overall performance by 9% as indicated in the second and third column of Table 3. The transfer nevertheless remains a serious performance limitation. Section 4 describes an integrated video interface design that eliminates the bus transfer altogether.

3.3.2. Memory Copies. The bus transfer involves moving data from hardware components to the host memory across a shared bus. The transfer rate between components on the bus can be improved by moving some of the hardware components onto local busses to provide a direct connection among blocks which have critical timing. In the testbed, local busses are employed to connect FG to VC, the camera to FG, and VGA to the display. The transfer rate in this case is limited only by the processing speed of the hardware blocks connected to the local bus.

Figures 17–19 show that transfers across the camera-FG and FG-VC local busses still take relatively long time. This limitation is due to the memory transfers that occur on the bus for image data stored on local memory buffers in the FG and the VC. For instance, to capture and compress an image, the FG hardware first stores the digitized image in its local buffers and then copies it to the VC memory buffer from which compression is performed. The compressed image is then copied back to the FG buffers and transferred to the host memory for network processing. This store-and-process scheme is appropriate for the testbed implementation since it allows a simple standardized interface for different hardware components to be plugged in for evaluation. However, from a performance standpoint, this interface design leads to degradation in frame rate. From Table 3, memory copies use 11% to 14% of the overall

frame processing time. To remove the memory copies requires an integrated interface design that allows data to be streamed into the various processing blocks in a pipelined fashion while local buffer processing still occurs but is done in parallel. An architecture which implements these techniques is presented in Section 4.

3.3.3. Reconfiguration Time. Aside from memory copies and bus transfers, the frame rate is substantially limited by the reconfiguration time of the FPGA in programming the coder and decoder architectures between mode changes during a VTALK session. The reconfiguration time is incurred once in the full transmit and receive cycle. As the receive node switches to the transmit mode the FPGA must be reconfigured from the decoder to a coder. Results in Table 3 show that the reconfiguration time exacts 29% to 37% of the total frame time. This overhead depends solely on the FPGA technology used. By using a more advanced FPGA such as ATT's ORCA2C10, the performance hit due to reconfiguration time can be reduced to less than 0.3%.

3.4. Power Distribution

Portable video applications require not only high frame rate but also long battery lifetime. Using a 24 Watt-hour NiMH battery, the current testbed has a battery life of just slightly more than one hour when connectivity is enabled. In order to evaluate the leading factors contributing to the short battery life, a detailed power consumption study has been conducted. Each subsystem is measured in isolation and the results are shown in Fig. 20. The most power consuming component of the wireless testbed is the data translation board used for image capture, accounting for 29% of the active power consumption for the entire system. The PC/104 motherboard consumes the second largest portion of the active current drain, accounting for 19% of all power drain. The hard disk subsystem consumes the third largest fraction of active current drain, approximately 14%. The fourth largest power consumer is the display subsystem, at approximately 9%. The display hardware consists of the LCD panel controller, the actual active matrix LCD, and a backlight module. By far, the interfaces and host CPU dissipates much more power than the basic signal processing and communication functions. In Section 4.0, a system architecture is described which allows these components to be powered off when they are not needed to the operation of active tasks, resulting in an increased battery life.

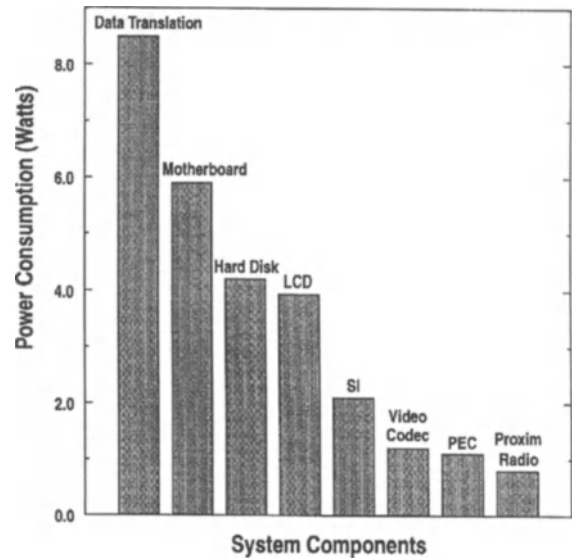


Figure 20. Relative power consumption in the wireless testbed.

4. Architecture for Low Power High Throughput Multimedia Terminal

Performance evaluation using the testbed has revealed the relative importance of the overhead incurred by the application and network protocols as well as the signal processing in the video and radio hardware. For a high performance node, the overheads due to bus transfers, memory copies, and network processing need to be minimized. Solutions to minimize each of these overheads will be addressed in this section.

Bus transfer is the main source of limitations to system throughput for applications requiring movement of large blocks of data across the system bus. Throughput is degraded due to contentions between host processes and data transfers. Thus, to improve the performance of the system, an architecture must perform these transfers independent of the host.

To achieve this, application-specific processing must be controlled locally requiring minimal processing overhead from the host. In general, most tasks performed by an application are initiated by commands from the host but are actually performed by other hardware blocks in the system such as the video codec or the radio. The task of scheduling the execution of these functional blocks should be performed by a separate controller. The host only needs to communicate with this controller occasionally, allowing the CPU to attend to other processes running on the host. The control and

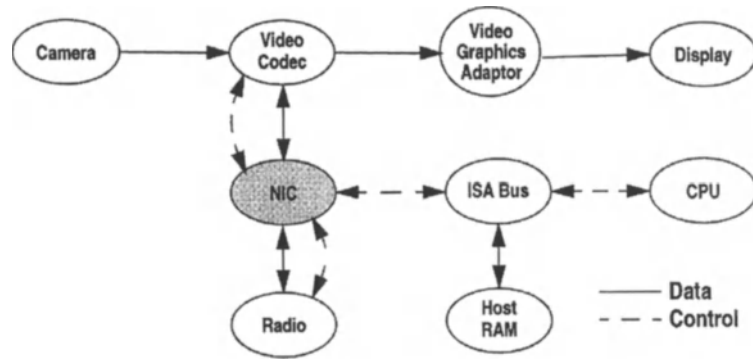


Figure 21. Control and data flow in the high performance node architecture.

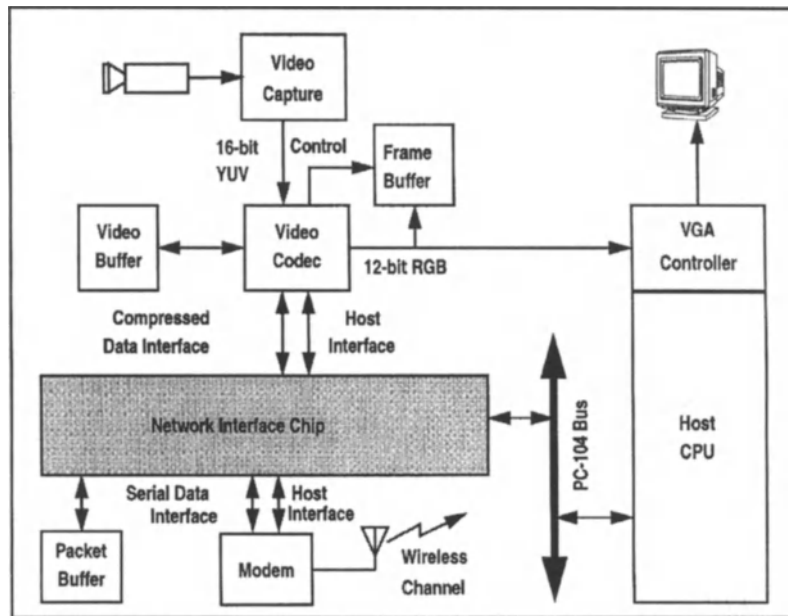


Figure 22. A high performance node architecture.

data flow in such an architecture is depicted in Fig. 21. Here, a network interface controller (NIC) controls the flow of data throughout the system. Comparing Fig. 21 to Fig. 16, the ISA bus is only used to send control information to the NIC and is no longer used to transfer data.

The architecture which implements the control flow diagram is depicted in Fig. 22. The three key components in this architecture are the NIC, the video codec, and the modem/radio. Memory copies are eliminated in both the video data path as well as in the packet interface by means of a stream-like processing of the image data and overlapping it with the packet processing. The interface of the video codec is designed to

decouple all video transfers from the host bus. For video capture, the codec has a direct interface to a digital CCIR-601 camera source which can be connected to commercial single-chip video digitizers such as the Brooktree Bt819. Therefore, raw video is passed directly to the CODEC without traversing the PC's ISA bus. For video display, the codec has a direct connection to a commercial flat-panel VGA controller capable of *overlaying* the codec's RGB output directly on the LCD screen without traversing the ISA bus or being copied to/from system or video memory. Similarly, compressed data transfers are kept off the host bus by a direct connection to the NIC. The NIC packetizes

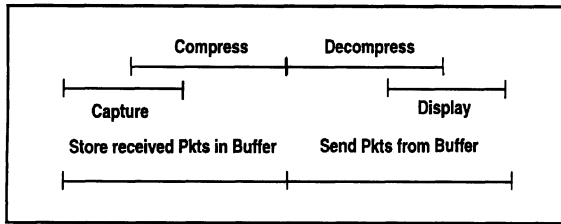


Figure 23. Timing diagram of operations performed in the high performance node.

or depacketizes compressed data that is received from or transmitted to the modem. The NIC also directly connects to the codec's host interface, through which the codec is instructed to encode, decode or vary the amount of compression.

The operations executed in this architecture is shown in Fig. 23. Note the capture and compression are overlapped since as pixels are generated from the video capture IC, they are immediately processed in the video codec and stored in the video buffer for subsequent passes of the compression processing. Meanwhile, packets are received by the radio and depacketized in the NIC. The received data is stored in the packet buffer and is routed to the host or video codec depending on the type of data. For video data, the contents of the packet buffer is transferred to the video codec for decompression. Concurrently, compressed data are packetized by the NIC and stored in the packet buffer for transmission through the radio. All these operations can be performed in 60 ms to achieve a frame rate of 15 fps for a duplex video link.

4.1. System Power

From Fig. 20, the DT interface consumes the highest power in the wireless testbed. By implementing a custom frame grabber design specific to the codec requirements, the power dissipation has been reduced by 90% for this functional block. A substantial amount of power is still being dissipated by the interfaces—approximately 5 Watts total. The power dissipation is due to using FPGAs in the interface implementation. For a testbed, this is tolerable. However, for a high performance node, these interfaces must be integrated with the associated signal processing functions in ASICs to reduce the active power further.

For low power battery operation power shut-down capabilities must be built into the video and radio hardware to allow subsystem powerdown along with fast wakeup modes. The current testbed does not support

this approach, primarily because the underlying PC based technology does not permit processes to be running during sleep mode though processes related to maintaining connectivity must be kept alive at all times. Thus, for the testbed standby power consumption is equivalent to active power consumption which totals to 20 Watts.

The worst-case power dissipation under ideal power management can be estimated by multiplying the percentage utilization in Tables 2 and 3 with the power dissipated for the corresponding blocks in Fig. 20. A 65% improvement over the original 20 Watts total system power can be achieved. The above estimate assumes 100% active cycle in which the nodes are always fully on. With shorter active cycle, further power saving is achievable.

5. Conclusions

A testbed has been constructed to enable the evaluation of key components required for wireless multimedia networking: spread-spectrum radios, video compression algorithms, TCP/IP-based network protocols, and applications. Performance evaluation of the FTP and VTALK applications underscores the importance of adaptivity in the radio as well as the video compression. The ability to change the spreading ratio and data rate in the radio and compression ratios in the video codec enables the network to adapt to different channel conditions and bandwidth requirements. Measured results show that the direct-sequence radio achieves an effective throughput of 70% to 90% as the chipping rate varied from 15 cpb to 127 cpb corresponding to data rates from 64 kbps down to 8 kbps. Degradation in throughput arises from higher packet loss due to higher BER at the lower chipping rates. For the RangeLan2 frequency-hop radios by Proxim, measured results show an effective throughput 20% of the 1.6 Mbps raw data rate. The limited throughput in this case is due mainly to the CSMA/CA MAC protocols implemented by the Proxim radios. Under the limited throughput, the subband video codec is able to achieve a compression ratio from 10:1 to 20:1 with a corresponding PSNR ranging from 33 dB down to 26 dB.

Measure of multimedia performance on the testbed show two important overheads which must be minimized in a high performance node design: 1) bus transfer and 2) memory copies. The former expends 17% of the system throughput while the latter expends 11%. An architecture is proposed which improves the packet

throughput by off-loading the control flow management for multimedia and network processing to a network interface controller. Bus transfer is eliminated in this architecture and the CPU processing time is also improved with the assistance of the network interface controller. The memory copies are eliminated by a tight integration of the interface functions with the associated signal processing blocks. Current testbed dissipates a total of 20 W and has an hour of battery life. Based on measured data, under ideal power management with the node in active mode, the total active power dissipation can be reduced by as much as 65%.

Acknowledgment

We would like to thank E. Roth for his assistance in assembling the testbed and J. Short for the initial work on the network operating system. S. Spurrier for the graphics in the figures. This work is supported by ARPA/CSTO under contract J-FBI-90-091 and the FBI under contract J-FBI-93-117.

References

1. P. Agrawal et al., "A testbed for mobile networked computing," *Proceedings IEEE Communication Conference*, Seattle, WA, June 1995.
2. B. Barringe et al., "Infopad: A system design for portable multimedia access," in *Wireless 1994*, Calgary, Canada, July 1994.
3. J.K. Holmes, "Coherent spread spectrum systems," Wiley, New York, 1982.
4. Pickholtz et al., "Spread spectrum for mobile communications," *IEEE Trans. on Vehicular Technology*, Vol. 40, No. 2, pp. 313–321, Dec. 1992.
5. M.K. Simon et al., *Spread Spectrum Communications*, Vol. 3, pp. 3–15, 1985.
6. C. Chien et al., "A 12.7 Mchips/sec all-digital BPSK direct-sequence spread spectrum IF transceiver," *IEEE Journal of Solid-State Circuits*, Vol. 29, No. 12, Dec. 1994.
7. P.T. Yang, "Design and implementation of a spread-spectrum radio," Ph.D. Dissertation, University of California at Los Angeles, 1993.
8. A. Salmasi and K. Gilhousen, "On the system design aspects of code division multiple access applied to digital cellular and personal communications networks," *Proc. of the IEEE Veh. Tech. Conf.*, pp. 57–62, May 1991.
9. B. Chung et al., "Performance analysis of an all-digital BPSK direct-sequence spread-spectrum IF receiver architecture," *IEEE JSAC*, Vol. 11, pp. 1096–1107, Sept. 1993.
10. A. Chandrakasan et al., "A low power chipset for portable multimedia applications," *IEEE International Solid-State Circuits Conference*, San Francisco, CA, Feb. 1994.
11. M. Harrand et al., "A single chip videophone encoder/decoder," *IEEE International Solid-State Circuits Conference*, San Francisco, CA, Feb. 1995.
12. B. Belzer et al., "Adaptive video coding for mobile wireless networks," *IEEE International Conference on Image Processing*, Austin, TX, Nov. 1994.
13. E. Tsern et al., "Video compression for portable communication using pyramid vector quantization of subband coefficients," *IEEE Workshop on VLSI Signal Processing*, 1993.



Charles Chien received the B.S.E.E. degree from the University of California, Berkeley, in 1989 and the M.S.E.E. and Ph.D. degrees from the University of California, Los Angeles, in 1991, and 1995, respectively.

In 1988–1989, he worked at Bell Communications Research, Red Bank, NJ, where he worked on a transmission system capable of carrying digital SONET-like HDTV on fiber at the rate of 622 Mbps. From 1989 to 1995, he worked as a Graduate Student Researcher at the Department of Electrical Engineering in UCLA, where he is currently the Principal Development Engineer in the Wireless Multimedia Networking Laboratory. His research interests are in digital communications, digital signal processing, high-speed integrated circuit designs, and VLSI implementation of DSP algorithms and communication systems.

Dr. Chien is a member of Tau Beta Pi and Eta Kappa Nu. He has received the Outstanding Master Student Award in the School of Engineering and Applied Sciences at UCLA in 1991 and the AT&T Bell Laboratories Doctoral Scholarship for the duration of his Ph.D. study.



Sean Nazareth was born in Montclair, New Jersey. He received the B.S.E.E. and M.S.E.E. degree from the University of California, Los Angeles, in 1992 and 1994 respectively. Since 1994, he has been a Graduate Student Researcher with the Department of Electrical Engineering at the University of California, Los Angeles, where he is presently pursuing the Ph.D. degree. His current

research is in the design of high performance terminals for wireless multimedia.



Paul Lettieri received the B.S.E.E. magna cum laude from Rensselaer Polytechnic Institute in Troy, NY in 1995. While working on this degree, he did an internship with Pitney Bowes, designing and evaluating embedded systems. Mr. Lettieri is currently working on the M.S.E.E. at the University of California, Los Angeles, where his focus is Integrated Circuits and Systems. For the last year, he has been involved in the system level design of wireless multimedia terminals at UCLA.



Stephen Molloy received the B.S.E.E. degree magna cum laude from Rensselaer Polytechnic Institute in 1991. He received the M.S.E.E. degree from the University of California, Los Angeles in 1993, where he is currently pursuing the Ph.D. degree. His research interests include digital video, architectures and integrated circuits for video signal processing and compression, and low power design. Mr. Molloy is a member of the IEEE and SMPTE.

Brian Schoner developed several innovative video processing systems while a graduate student at UCLA. He received his M.S.E.E. degree in 1994 for his work on a reconfigurable ASIC for video processing. In 1994 he implemented a real time wavelet transform-based video compression system on a single 10k gate FPGA. As a research assistant in 1995, he worked on a team to use dynamic computing techniques to create an entire video codec and transceiver based on a single 5k gate FPGA. Mr. Schoner is currently a design engineer at LSI Logic in Fremont where he continues to work on efficient digital video products.



Walter A. Boring IV was born and raised in California. He received the B.S. in Computer Science from the California State University, Chico. He worked for Apple computer for 4 years as an engineer before he moved to Los Angeles to be a Research Engineer for the Electrical Engineering department at UCLA. He is the main software/systems engineer for the department's efforts in wireless multimedia networking.



Joey Chen received the B.S. degree in Electrical Engineering and Nuclear Engineering from University of California, Berkeley, in 1994. He is currently involved in the development of a video subsystem for wireless terminals in the Electrical Engineering Department at University of California, Los Angeles. His technical interests are in VLSI design and signal processing. Mr. Chen is a member of Eta Kappa Nu.



Christopher Deng received the B.S.E.E. degree from University of California at Los Angeles in 1995. He is currently in the M.S.E.E.

program at UCLA in the department of Integrated Circuits and Systems. His study emphasizes digital communication circuits, especially spread spectrum systems. Mr. Deng is a student member of IEEE and life member of Eta Kappa Nu and Tau Beta Pi.



William Mangione-Smith earned his Bachelors, Masters and Doctorate from the University of Michigan. He was employed by Motorola for three years, on various projects involving low power portable computing systems with integrated wireless data channels. In 1995 he joined the electrical engineering department at the University of California at Los Angeles. His current research interest focus is on low power computer engineering, involving hardware, communications protocol, and systems design issues.



Rajeev Jain received the B.Tech. degree in Electrical Engineering from the Indian Institute of Technology in 1978 and the Ph.D. degree in Electrical Engineering from the Katholieke Universiteit, Leuven in 1985. Currently, he is an Associate Professor at UCLA.

He has worked at Siemens AG, in Munich where he developed CAD tools for automatic programmable signal processor code-generation to implement digital filters. He has been a research group leader at IMEC, Leuven where he was responsible for the development of the Cathedral-I CAD system for automatic design of DSP circuits using bit-serial architectures, which has resulted in a commercial product called DSPStation. As a Research Engineer at U.C. Berkeley he worked on the development of the LagerIV design system for bit-parallel signal processing circuits.

His current research efforts at UCLA are concentrated on CAD tools for design of high-performance signal processing architectures and the development of ASICs for spread-spectrum modems and image compression. Rajeev Jain is the recipient of the 1991 Northrop Outstanding Junior Faculty Research award.

Design of a Low Power Video Decompression Chip Set for Portable Applications*

BENJAMIN M. GORDON, ELY TSERN AND TERESA H. MENG
Center for Integrated Systems, Stanford University, Stanford, CA 94305

Received October 30, 1995; Revised April 12, 1996

Abstract. This paper describes the design process of a chip set which performs real-time video decompression for wireless portable applications and concentrates on four critical aspects of the design: compression algorithm development, control complexity, programmability, and throughput. For each of these design areas, this paper evaluates the design trade-offs between low power, compression efficiency, and throughput, which are the three main requirements for wireless portable video. The chip set consists of a subband reconstruction chip and a pyramid vector quantization (PVQ) decoder chip and requires no external memory support or frame buffer. For portable applications with a resolution of 176 pixels wide, 240 lines, and 30 frames per second color video, the chip set, operating at a 1.35 V supply, dissipates less than 9 mW.

1. Introduction

Video has rapidly become an integrated part of our information exchange and to meet this need, a growing number of computer systems are incorporating multi-media capabilities for displaying and manipulating video data. This interest in multi-media, combined with the popularity of portable computers and phones, provides the impetus for creating a portable video-on-demand system. Because of the large bandwidth requirement, for both storage and transmission, data compression must be employed, requiring real-time video decoding in the portable unit. The key design consideration for portability is the reduction of power consumption to extend battery life.

Designing systems for low power requires a high level approach to estimating power consumption to allow easy evaluation of differing schemes. In digital CMOS implementations, the power is dominated by the switching power given by $P = \alpha CV^2 f$ where α is an activity factor indicating the percentage of nodes switching, C is the capacitance, V is the supply volt-

age, and f is the frequency of operation [1]. For a higher level of insight, the power can alternatively be viewed as

$$P = \sum_{\text{ops}} \frac{\text{energy}}{\text{op}} \times \frac{\text{ops}}{\text{sec}}$$

the sum over all operations of the energy per operation times the operation throughput. Now algorithms can be analyzed according to the type and quantity of operations needed to implement them. The energy per op can be estimated for the largest power consuming operations including computation (addition, multiplication, etc.), memory accesses (internal and external), and I/O. Table 1 shows the spice simulated estimates for the layout cells in a $.8\mu$ CMOS process operating at 1.5 V. The estimate for the external memory is based on commercially available 4 Mbit SRAM running at 3.3 V with 16 bit data access. Within this framework, various trade-offs can be evaluated by how they change operation throughputs given fixed energy per ops. For instance, the quality of the algorithm may be degraded to achieve reductions in computation or memory requirements, with the amount of tolerable loss dependent upon the exact system goals. Further trade-offs may be made by introducing greater complexity in the

*This research was supported by JSEP contract number DAAH04-94-G-0058.

Table 1. Energy per operation ($.8\mu$ CMOS, 1.5 V).

Operation (16 bit)	Energy/Op (pJ)
Add	7
3-2 Add	2
Multiply	40
Latch	1.8
Internal read	36
Internal write	71
I/O	80
External memory	16000

implementation's controller to reduce energy expensive operations. The controller typically requires little power, and any increases are outweighed by the other savings. The main cost incurred is the increased design and implementation time required to develop a more complex system. Computation and memory may also be traded for each other. Memory can be very power expensive, so numerous computations may be substituted to achieve an overall reduction. Also, excess throughput can be exchanged for lower power by reducing the supply voltage. This reduces the energy per operation, while the operation throughputs remain fixed. The voltage may be reduced, increasing the circuit delay, until it just meets the throughput constraints. Finally, programmability, which introduces overhead and inefficiencies, should be discarded to achieve the lowest possible power consumption. However, some programmability in the controller may be tolerated since its overall power contribution is small.

In this paper, we evaluate many of these trade-offs in the design of a video decompression chip set for portable video applications. The two-chip set implements the decoding process for a video compression algorithm using subband decomposition, which decorrelates a video frame using filters, and pyramid vector quantization, a lattice-based quantization scheme optimized for subband data. The PVQ chip performs decompression by converting pyramid vector quantization codewords into subband data values. The subband chip reconstructs four levels of hierarchical subband structures for color images. The paper's four major sections discuss the design areas of compression algorithm development, control complexity, programmability, and throughput. For each of these areas, we emphasize the trade-offs between low power, compression efficiency, and throughput.

2. Compression Algorithm Development

Specifying the algorithm is the most important step in designing a low power system. The largest power savings can be attained through algorithm recasting specifically targeted for energy conservation. The challenge is obtaining maximum energy savings while maintaining good compression efficiency and image quality. This section discusses our algorithm design process, which includes trade-offs between inter vs. intraframe processing, selecting and optimizing a data transform, and choosing a quantization technique.

2.1. Inter vs. Intra Frame Compression

An intra-frame scheme, which operates only on a single frame, was selected due to the high power consumption required for inter-frame methods. These inter-frame techniques, such as motion estimation or $3d$ subband filtering require storage of typically three frames of data. For a CIF size image (352×288) at 30 frames per second (fps), MPEG style motion compensation uses 13.68 Maccesses per second from a 3.65 Mbit storage for 3 YUV frames (current, previous, and future frames). Given an energy access of 1 nJ per bit for the commercial SRAM, this totals 110 mW, which is much higher than our target power of under 10 mW. Future memory technology using a reduced supply voltage and wider access widths may eventually provide sufficiently low power to warrant inter-frame compression [2].

Several inter-frame schemes were, in fact, evaluated that operate on compressed base-frames to reduce the memory size and bandwidth requirements, but were found to offer small compression gains of under 1.5 times improvement while still requiring relatively large power. With these approaches, the stored base frame must be uncompressed, motion-compensated, and recompressed into the next base frame. The most basic scheme stores a sub-sampled version of the base frame, making the decompression and compression very simple while reducing the memory size by 4 or 16 [3]. Another approach, for motion-compensated subband data, discards the larger subbands thereby using only 1/4 of the storage. The largest reduction is achieved with a conditional replenishment (CR) scheme, where only the compressed intra-frame data is stored and new compressed values simply replace the previous ones. This scheme works well with a fixed-rate intra-frame code so that the exact position

of values is known, thus making substitution straightforward. However, all these reduced memory schemes greatly diminish the effectiveness of the original full frame motion-compensation algorithm. The compression gains become small while the power is still large compared to the intra-frame power.

2.2. Data Transform

Most intra-frame compression algorithms include a data transform, such as the discrete cosine transform (DCT) or subband filtering. This section describes why subband filtering was selected over the DCT based upon both power and quality issues. The DCT, used in JPEG and MPEG, is a block-based transform, typically operating on 8 by 8 pixel blocks. The DCT is a real valued version of the discrete Fourier transform, converting pixel data into spatial frequency information. For a Markov-1 process with $\rho \rightarrow 1$, the DCT optimally decorrelates the input data, compacting the information into the minimal energy. The one-dimensional DCT is defined as:

$$X(m) = \sqrt{\frac{2}{N}} k_m \sum_{n=0}^{N-1} x(n) \cos \left[\frac{(2n+1)m\pi}{2N} \right]$$

$$k_m = \frac{1}{\sqrt{2}}, \quad (m = 0)$$

$$= 1, \quad (m \neq 0)$$

where $x(n)$ is the input data, $X(m)$ is the transform coefficients, and N is the block length.

Many image and video compression algorithms and implementations utilize the 2D inverse DCT (IDCT) for decompression. Typically, an 8 point 1D IDCT is applied to 8 rows of data, followed by an IDCT

on the columns of the results, totalling 16 IDCTs for an 8 by 8 block. The minimum computational requirements based on the efficient Lee’s algorithm [4], consists of 12 multiplies and 29 adds per 8 point 1D IDCT, resulting in 3 multiplies per pixel and 7.25 adds per pixel. However, many implementations require higher computations to achieve a regular and more readily achievable DCT design. Here, as shown in [5, 6], 32 multiplies and 40 adds are needed per 8 point IDCT, resulting in 8 multiplies and 10 adds per pixel. With further optimizations some of the multiplications can be made trivial, resulting in only 5 multiplies per point [7]. However, the multiply coefficients for the IDCT are complex, requiring full multipliers or large, irregular shift and add implementations.

In contrast to the DCT algorithm, subband coding uses filtering and decimation to compact image information into smaller bands. The image passes through both low and high-pass filters, vertically and horizontally, creating four subbands: low-pass vertical/low-pass horizontal (LL), low-pass vertical/high-pass horizontal (LH), high-pass vertical/low-pass horizontal (HL), and high-pass vertical/high-pass horizontal (HH), as shown in Fig. 1. After the filtering, the bands are decimated by two in each direction, as shown by the down arrows in Fig. 1. For reconstruction, the data is upsampled, illustrated by the up arrows, filtered, and combined.

Because of the filtering, the bands have reduced spatial frequency content, allowing decimation without loss of information. Each band is one-fourth the size of the original image but contains different spatial frequency information. The LL band appears as a subsampled version of the original, while the other bands hold the high frequency detail. Since most of the image

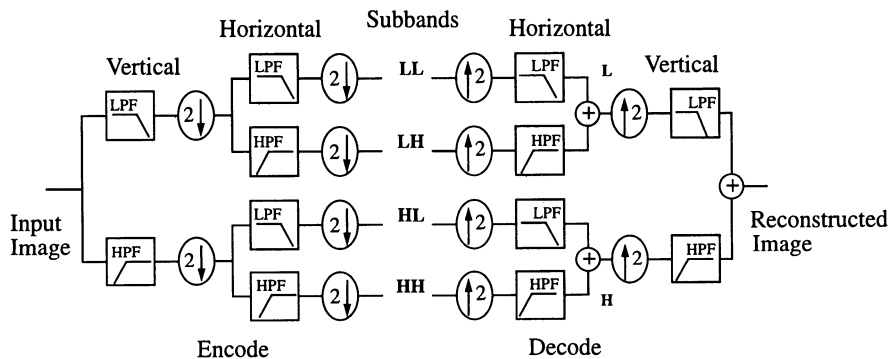


Figure 1. Two-dimension subband filtering structure.

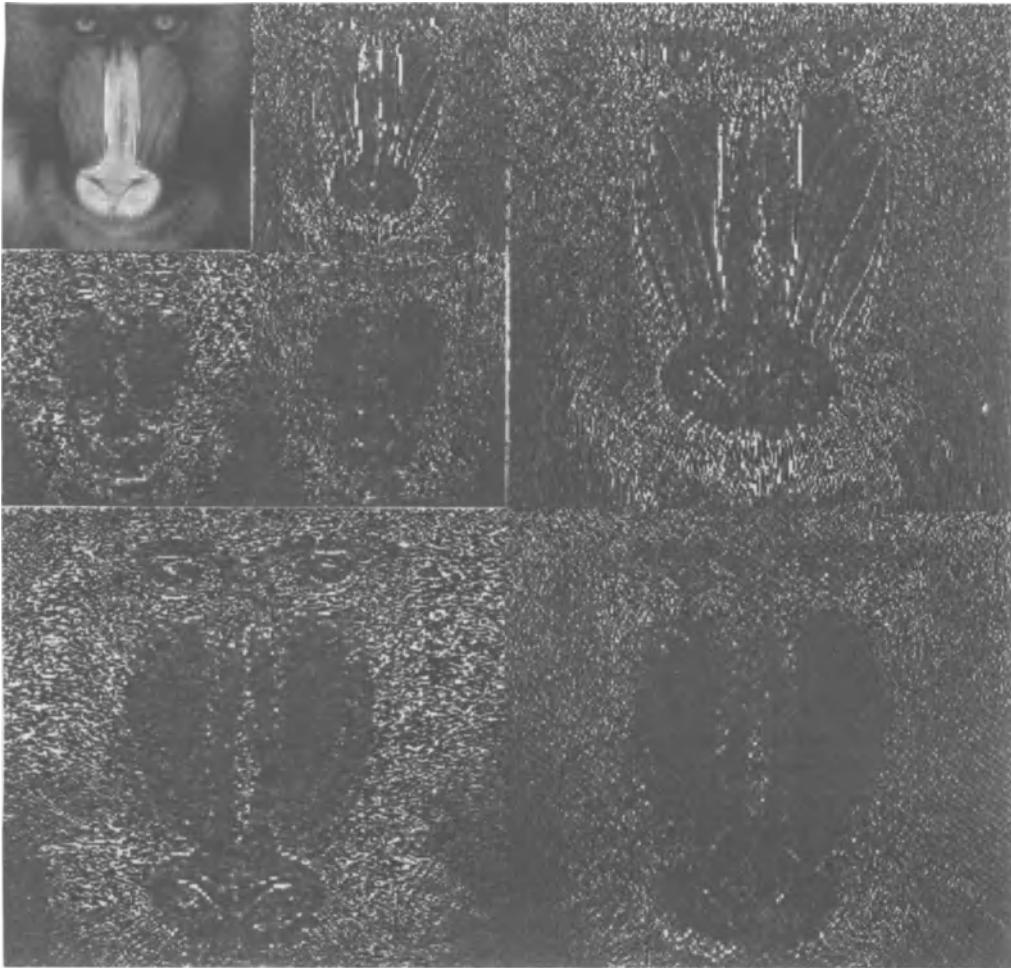


Figure 2. Subband decomposed image.

information is contained within the LL band, the subband filtering process is re-applied to this band creating another level of 4 subbands as illustrated in Fig. 2. A three-level subband decomposition is shown in Fig. 3, which, for the lowest frequency component, has the same spatial frequency resolution of the 8×8 DCT. Each DCT coefficient has a resolution of $1/64$ of the 2D spatial frequency domain while the subband levels have $1/4$, $1/16$, and $1/64$ respectively.

In subband coding the amount of processing depends upon the type and length of the chosen filter. The total computation for a single level, non-symmetric filter is $2 * N$ multiplies and adds, where N is the filter length. Three levels of subband processing increases this by a factor of 1.375 ($1 + 1/4 + 1/16$). With symmetric even filters, a polyphase configuration reduces the number of

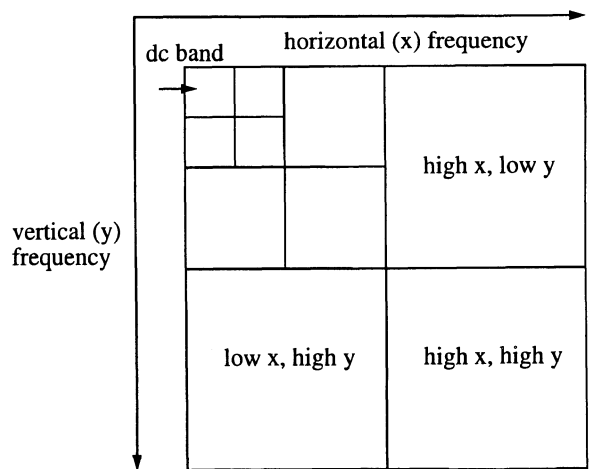


Figure 3. Luminance 3-level band decomposition.

Table 2. Memory and computation estimates.

Filter name	Coefficients (Un-normalized)	Memory lines	Adds	Multiplies
DCT (min.)		16	7.25	3
DCT (typ.)		16	10	5-8
h2	1, 1	3.5	5.5	2.75
w4	3, 6, 2, -1	7	11	11
db4	2.73, 4.73, 1.27, -.73,	7	11	11
w8	-8, 8, 64, 64, 8, -8, 1, 1	14	22	22
qmf9	.02, -.04, -.05, .29, .56, .29, -.05, -.04, .02	15.75	24.75	13.75
qmf12	-.004, .019, -.003, -.085, .088, .48, .48, .088, -.085, -.003, .019, -.004	21	33	16.5
qmf16	.001, -.005, -.003, .03, -.01, -.1, .1, .48, .48, .1, -.1, .03, -.003, -.005, .001	28	44	22
bi53	-1, 2, 6, 2, -1 1, 2, 1	8.75	13.75	6.875
bi97	.04, -.02, -.11, .38, .85, .38, -.11, -.02, .04 -.065, -.04, .42, .79, .42, -.04, -.065	15.75	24.75	12.375

multiplies by $2x$. With symmetric odd filters, the data can be folded since they share the identical filter coefficients, also reducing the multiplies by $2x$. Another factor in choosing a scheme is the memory requirements. The IDCT itself only needs an 8×8 transpose memory to store the 1D row IDCT results. However, the outputs are not in raster-scan order and require a minimum of 16 lines of memory to double buffer the output results. Subband implementations typically require storage of $N/2$ lines of horizontal filter results for each subband level, resulting in $N * 1.75$ for a 3 level structure. Additional memory may be required for storage of the final output results. The computation and memory requirements for various filters and the IDCT are summarized in Table 2. The short filters (2, 4, or 5 taps) compare favorably to the IDCT in both memory and computation while medium length (8 or 9 taps) have slightly higher requirements.

The final criteria involves the relative compression performance of the DCT versus the subband algorithm. The main function of the data transform is to decorrelate or “compact” the image information into as few values as possible. An energy compaction, or coding gain, measure is defined by

$$G = \frac{\sigma_x^2}{\prod_{k=1}^M (\sigma_k^2)^{1/M_k}}$$

where σ_x^2 is the input image variance and the bottom term is the geometric mean of the variance of the subbands or DCT coefficients. M_k equals 64 for an 8×8 DCT or 4, 16, 64, and 256 for the subbands depending upon the size of the band.

Table 3 gives this measure, averaged over a set of images for various subband filters, using 3 and 4 level decomposition, relative to the 8×8 DCT value. As seen in the table, the subband performance is comparable to the DCT, especially with longer filters. Additionally, the visual artifacts created by the DCT are subjectively more noticeable, especially at lower bit rates (below .5

Table 3. Relative energy compaction.

Filter	4-Level	3-Level
h2	.60	.60
w4	.80	.79
db4	.79	.77
w8	.87	.85
qmf8	.86	.84
qmf9	.90	.89
qmf12	.91	.88
qmf16	.92	.90
bi53	1.03	1.02
bi97	.98	.96

bpp). The DCT creates blockiness due to mismatches between the 8×8 coded blocks, while subband filtering does not suffer from this problem since it filters over the entire image. The visual superiority of subband filtering over DCT is noted by many case studies [8] and leads us to use subband filtering given the potentially attractive computational and memory requirements.

Given the subband scheme, the filter and decomposition must then be determined. Based on the energy compaction measure, four levels do not appear to give much advantage over three levels. However, in actual quantization schemes, the additional level provides higher compression efficiency and only adds a small amount to the power. The computational requirements increases by $1/64$, from 1.3125 ($1 + .25 + 0.0625$), a gain of only 1.5%, while the memory increases by $1/8$ from 1.75, a 7% gain.

2.3. Filter Selection

To determine which filter to use, the performance of various filters [8–12] were analyzed in actual compression schemes using scalar quantization/entropy coding and pyramid vector quantization (PVQ), described in Section 2.4.2, resulting in the choice of the 4 tap wavelet filter (w4). Several categories and lengths of subband filters are analyzed. One class is the wavelet filter which has perfect reconstruction (i.e., all aliasing caused by up and downsampling is perfectly cancelled), but is an asymmetric filter. For symmetric filters based upon one kernel filter, a near perfect reconstruction filter is used, such as the quadrature mirror filter (QMF). For both symmetric and perfect reconstruction filters, two filter kernels are required as used by the biorthogonal filters. For different filters, Fig. 4 illustrates the peak signal to noise ratio (PSNR) versus entropy using scalar quantization, while Fig. 5 shows the PSNR versus bits per pixel using pyramid vector quantization (PVQ) of two different images (girl with hat (LENA) and aerial view of LA airport (LAX)). The 9/7 tap bi-orthogonal kernel filters (b97) performed best in all cases. Of the QMF filters, the nine tap kernel (qmf9) outperformed both the 12 (qmf12) and 16 tap (not shown). The maximum difference between filters, ignoring the Haar (h2) which performs very poorly, is 1.5 dB. For Lena, with both scalar and pyramid quantization, the spread ranges from 1 to 1.5 dB, while for LAX the spread is much lower, staying under 0.5 dB. Thus

the choice of filters, barring the Haar, will not have a significant visual impact upon the image quality. The 4 tap wavelet (w4) filter and the 5/3 tap bi-orthogonal filters (b53) both offer excellent choices of low memory and computation. The b53 performs very well, but the four tap has slightly lower memory requirements, which was the deciding factor. Additionally, both these filters have very simple coefficients making their implementation extremely efficient compared with the QMF filters as well as the DCT.

Another consideration in filter selection is the effect on boundary handling. Wrap-around at the image edges will give correct results for any filter. Reflection can also be used for linear phase (symmetric) filters and has the added benefit of not introducing an artificial discontinuity at the border. The easiest approach is replication of the edge value, though this will not generate perfect results. All of these approaches cause a pipeline break due to the extra values that must be loaded in, leading some to propose wrapping or reflecting to the next line of data [13]. This does eliminate the break but assumes that all the lines are processed consecutively, re-creating the entire subband level.

The boundary problem is further complicated by the multi-level processing technique used in the subband chip's reconstruction processing. This more memory-efficient approach for multi-level subband decomposition requires moving between levels instead of completing each level in one pass. This requires storage of the next line's initial values for each subband, which may be cumbersome depending upon the exact implementation. Generating the values for wrap-around and reflection can also be problematic. Wrap-around may need values from the end of the line at the front and re-uses values from the front at the end. Reflection is simpler but requires re-ordering of the first and last values. Only repetition is easy to implement but does compromise quality. Shorter filters have the advantage of extending less over the image edge. In fact, the w4 does not extend over the front at all and extends only one value over the end. This makes a wrap-around implementation relatively easy, requiring storage of only the first lines input value to be re-used at the end. The nine tap overlaps by two values in front and three at the end while the b53 uses one value in the front and end.

The filter boundary problem is further compounded when multiple chips are used to decompress a single image. Each chip works on its own vertical slice

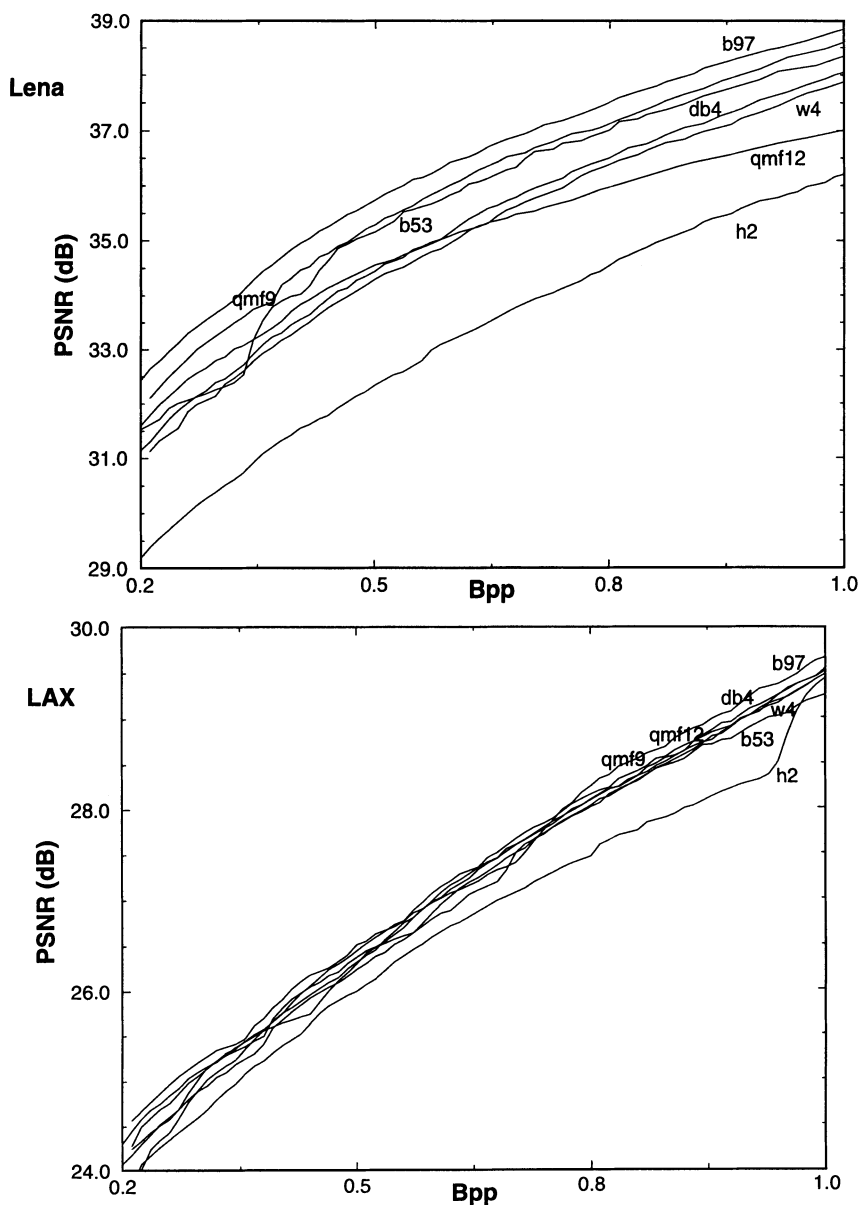


Figure 4. Filter performance for scalar quantization.

of the image up to a maximum width determined by its on chip storage. If the chips perform their operation solely on their own slice, a vertical line will appear at the slice boundaries. By sharing values across the boundary, this artifact can be eliminated. Filters which are 5 taps or shorter require only two values from the adjacent slice. For longer filters, within a multi-level decomposition, the number of additional values grows with the number of levels. A 9 tap filter requires 2 values at the top level but 5 values at the lowest level. These edge handling

considerations provide additional motivation for using the w4 filter.

2.4. Quantization of Subbands

To achieve compression, each subband is quantized, and the number of bits allocated to each subband set according to energy content. The quantization scheme largely determines compression efficiency and has a large impact on the hardware complexity

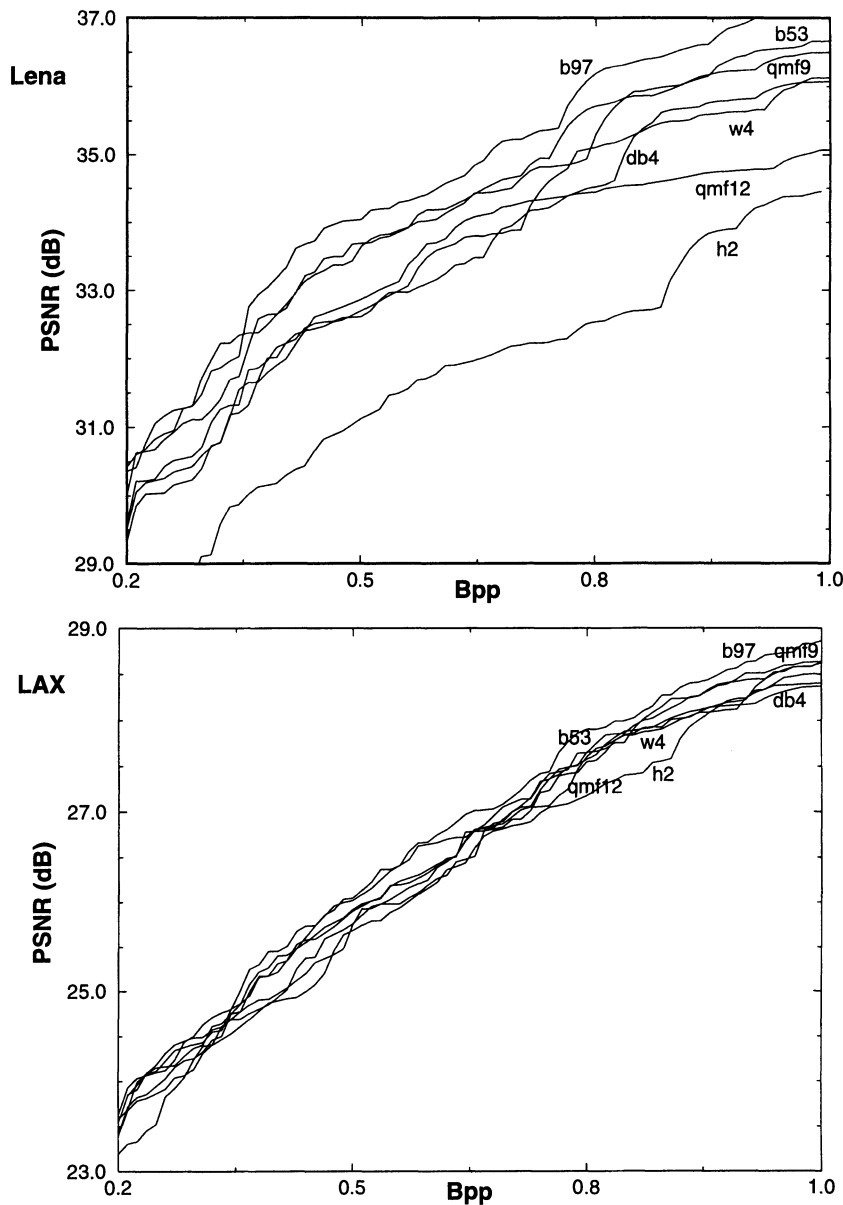


Figure 5. Filter performance for PVQ.

and error-resiliency, two important design criteria for portable wireless systems. Using these criteria, this section compares variable-rate quantization and various fixed-rate schemes with the fixed-rate PVQ coding scheme. The PVQ scheme is used due to its superior performance in the presence of bit errors as will be shown in this section.

2.4.1. Variable-Rate Coding vs. Fixed-Rate Coding.

Variable-rate coding is a common, effective approach of quantization for subband data. This approach

combines either scalar or vector quantization with entropy coding, such as arithmetic coding, Huffman coding, or zero run-length coding. Because entropy coding takes advantage of data redundancy and compresses varying amounts depending on the original data, variable-rate codes usually achieve better compression than fixed-rate codes, but the compressed bit rate is not constant.

Variable-rate codes also has drawbacks, which may make it unsuitable for wireless portable applications. They result in greater hardware complexity, requiring

additional buffering and synchronization control in decoding. Variable-rate codes are also highly susceptible to bit errors, since a single bit error can cause error propagation and produce totally erroneous decoded results until the next synchronization point.

In fixed-rate coding schemes, the compressed bit rate remains fixed, as do the word and block boundaries. Since these boundaries are pre-defined, fixed-rate bit streams are much less susceptible to bit errors, since error propagation is limited to corruption within the boundaries. A single bit error can cause deviation of decoded values, but does not affect the critical data position information. With an optimized fixed-rate scheme, the compression algorithm becomes inherently error-resilient, so that if channel distortion does occur, its effect will be a gradual degradation of video quality. With this approach, we do not pay a high premium in bandwidth overhead using error-correction codes, when protection is not needed (under low distortion conditions) and still deliver reasonable quality video when these codes fail under severe distortion condition. The drawback with fixed-rate codes is that they typically exhibit less compression efficiency compared with variable-rate codes. The goal is to develop a fixed-rate scheme with compression efficiency comparable to variable-rate codes, while maintaining the benefits of error-resiliency and less hardware complexity.

2.4.2. Pyramid VQ. A variety of fixed-rate quantization schemes exist, including a family of LBG-based vector quantization schemes, lattice-based vector quantization, and specialized scalar quantizers optimized for different data distributions [14]. For subband data, a statistical analysis confirmed that the data has a distribution similar to Laplacian (double-sided exponential) [15]. First introduced by Fischer [16], PVQ is a fixed-rate, lattice-based quantization method optimized for Laplacian sources. Its codebook is formed from points on the cubic lattice which lie on the surface of an N -dimensional pyramid. PVQ is a form of geometric source coding, which means that its codebook consists of a subset of lattice points that closely matches the source's high probability region. Its fixed-rate nature makes it less sensitive to bit errors, and its compression performance has been shown to approach the performance of entropy-coded scalar quantization.

In practice, PVQ, in combination with subband decomposition, can perform as well as variable-rate

schemes, such as that used in JPEG, the industry standard for image compression. Figure 6 shows the comparable image quality, measured by PSNR, of subband/PVQ vs. JPEG, over a wide range of bit rates [17]. Furthermore, the subband/PVQ algorithm exhibits greater error-resiliency as illustrated in Fig. 7 which compares the two algorithms at 1.0 bpp over a range of bit error rates (BER) [18]. Notice first that the raw JPEG bitstream, without error correcting codes, degrades rapidly with increasing bit errors. With error correction, (73, 45) Weldon difference-set, JPEG maintains good image quality until the BER reaches over 10^{-2} , where the correction code starts to fail catastrophically. The subband/PVQ curve gradually degrades with increasing bit errors which results in two nice properties. First, the quality is higher under low channel distortion. Since there is no error correction used with subband/PVQ, there is, in this case, 40% additional bandwidth available to code the image and improve image quality over the error-corrected JPEG. Secondly, under the high BER conditions (10%) which may occur with serious fading in the channel, the subband/PVQ scheme is over 6 dB better.

Pyramid VQ also has distinct hardware advantages over other fixed-rate VQ schemes [19]. Many VQ schemes have large memory requirements, because a "codebook", which is the collection of the representative quantization points, must be stored for the decode look-up process. The size of a codebook is a function of the bit rate and vector dimension:

$$\text{codebook size} = 2^{NR}$$

where N = vector dimension and R = bit rate.

Ideally, for the best compression efficiency, the largest possible vector dimension should be used to exploit larger multi-dimensional quantization space and result in a more accurate data representation. From the equation above, however, the codebook size exponentially increases with vector dimension, and as a result, physical memory limits the vector dimension (typically <16) for practical VQ-based systems.

Pyramid VQ has a computation-based decoding process, which performs arithmetic operations to decode quantization points. Because this algorithm has minimal memory requirements, PVQ allows for very large vector dimensions and large codebook sizes in a practical system without being bound by physical memory limitations. The use of vector dimensions up to 128 further improves the compression efficiency of

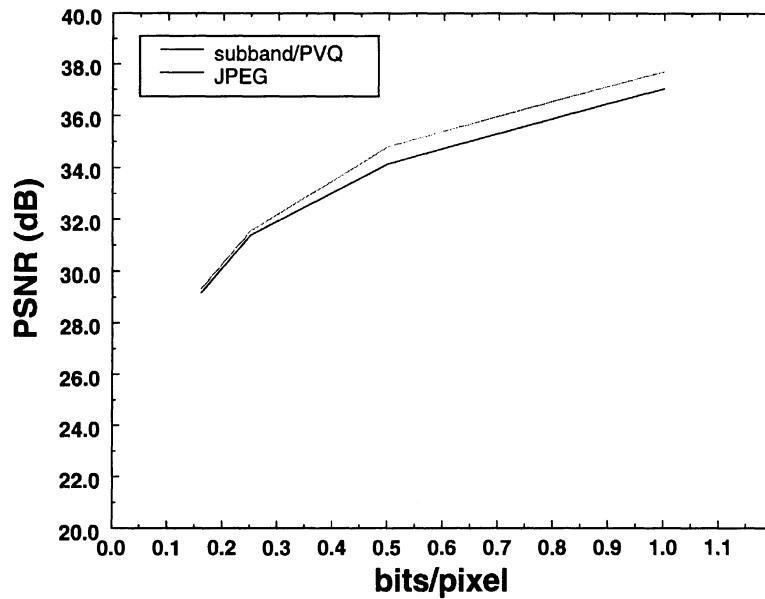


Figure 6. PSNR vs. bit rate for JPEG and subband/PVQ.

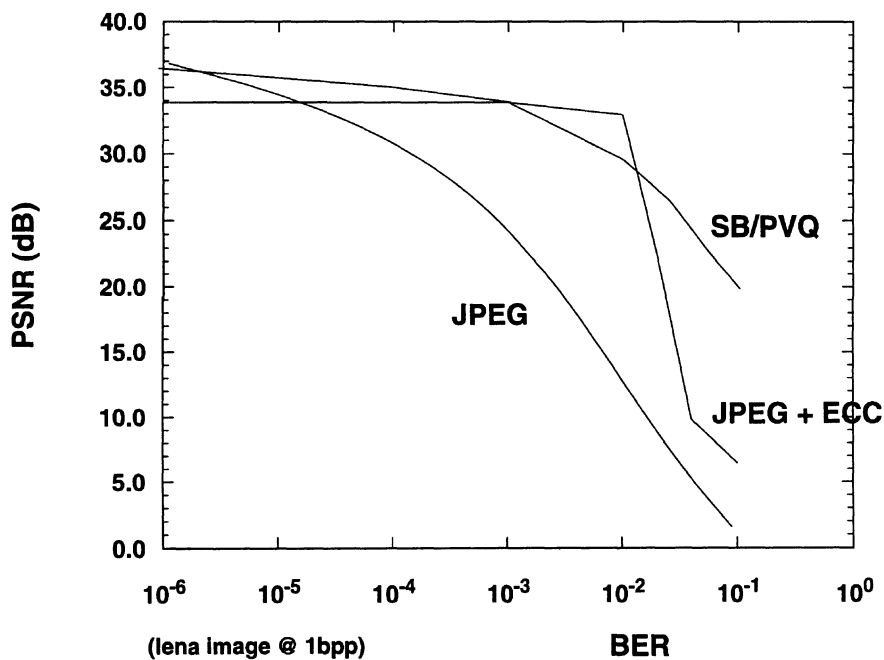


Figure 7. PSNR vs. bit error rate for JPEG and subband/PVQ.

this approach. As an example, to code one subband at a bit rate of 2 bits/pixel and a vector dimension of 64, the PVQ codebook size is 3.9×10^{38} . A codebook size of this magnitude results in very good compression performance but would clearly be impractical

to implement for memory-based VQ schemes. Furthermore, for codebook sizes that can reasonably be implemented with large memories, codebook accesses still consume greater power than computing results.

3. Programmability vs. Hard-wired

Given a particular algorithm, a certain amount of programmability in the implementation may be desired. However, this leads to increased power and should, in our opinion, be avoided except in the controller which usually has a small overall contribution to the total power. This section illuminates the programmability which was removed from the subband chip design for implementation efficiency, along with the programmability which was left in due to its importance and low overhead.

3.1. Filter

For the compression algorithm we have selected a single fixed filter suitable for a low-power design. A hard-wired shift and add implementation, commonly used for area and performance optimization, delivers the full benefit of this selection. Figure 8 shows the actual 4 tap filter implementation of the high and low pass filters, up sampling, and combining. A single 3-2 adder implements each pair of filter coefficients. The same hardware performs both the low and high pass filtering by reversing the input data and negating the (3, 2) pair result. A 4-2 adder and a carry-propagate adder combine the low and high pass values forming the reconstructed result. The implementation requires very little computation and

thus consumes much less power than a full multiplier version.

3.2. Precision

With any implementation arises the question of data precision width. Most programmable DSP processors have a 16 or 32 bit width to support a wide range of applications. Here we can pick the minimum precision that does not degrade the image quality. Precision specifications are needed for the input and output data as well as internal *Id* filtering results and constant coefficients. The input and output precisions should match since the outputs are fed back to the inputs for multi-level reconstruction. These precisions determine the datapath and memory width, which directly increase both the energy and delay. Therefore, the smallest widths should be chosen which do not significantly impact the resulting quality. An input/output precision of ten bits provides no significant loss in quality for this application while any less begins to have an impact. The precision of the internal filter results also affects the implementation. A precision of 10 bits has no additional loss for the chosen 10 bit input precision when rounding is used, whereas truncation performance suffers as shown in Fig. 9. The power and area cost of rounding is more than compensated by the reduction in memory storage and subsequent datapath computation. The coefficient accuracy

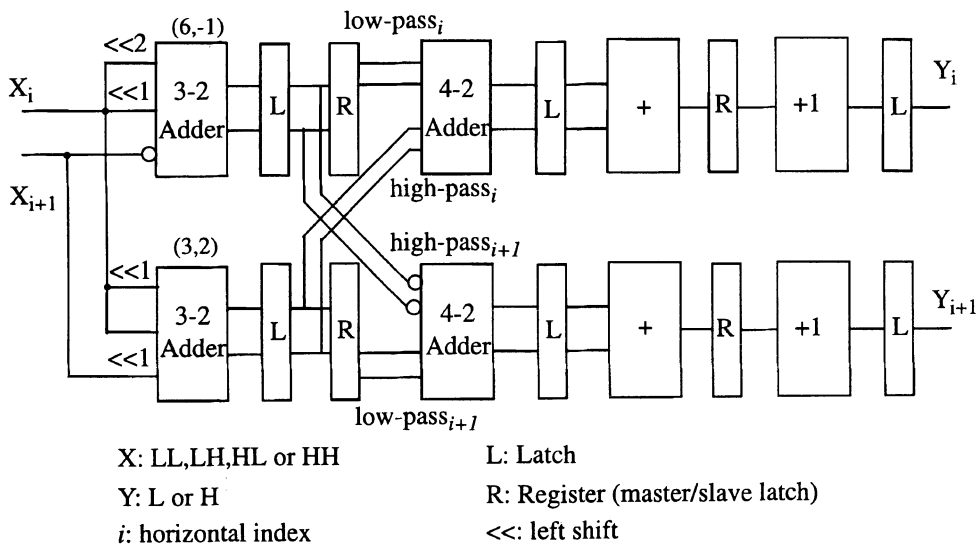


Figure 8. Filter implementation.

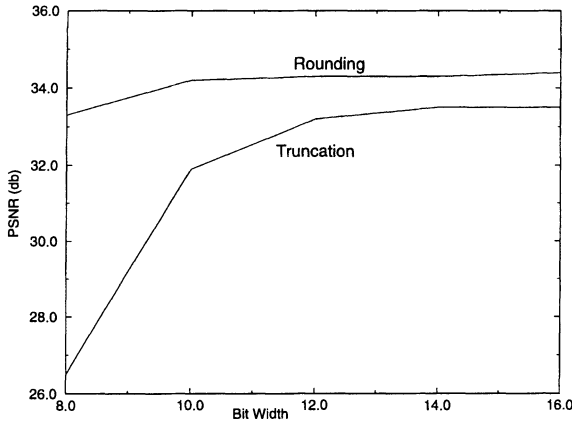


Figure 9. Subband data precision.

is not an issue since the four tap can be exactly represented. A required normalization of $4/100$ is approximated by $41/1024$, implementable with a single 3-2 adder and CPA ($32x + 8x + x$) with an accuracy within 0.1%.

The accuracy of other coefficients for the chip's color space (YUV to RGB) conversion must also be specified. All the coefficients require only a single CPA but have no visual image quality degradation [20]. The exact conversion matrix and its approximated counterpart are shown below:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.402 \\ 1 & -0.34414 & -0.71414 \\ 1 & 1.772 & 0 \end{bmatrix} \times \begin{bmatrix} Y \\ U - 128 \\ V - 128 \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.5 \\ 1 & -0.375 & -0.75 \\ 1 & 1.5 & 0 \end{bmatrix} \times \begin{bmatrix} Y \\ U - 128 \\ V - 128 \end{bmatrix}$$

3.3. System Configuration

Other issues in programmability arise from a system wide perspective. First, the system performs only decoding and not encoding. Many data transform implementations leverage the similarity between the forward and inverse transform to perform both functions in the same chip. This, however, incurs some computational overhead, as well as limits the space of possible optimizations. Since the chip set

is dedicated to decode only, this additional overhead was eliminated.

One programmability aspect included on the subband chip was the image width and video timing parameters. This allows various displays to be supported while only impacting the controller thereby limiting the power increase. In the controller, loadable counters using programmable values were used instead of fixed value counters to keep track of the position within a data line, thus providing a low overhead solution to support different image widths.

4. Control Complexity

Another technique for power savings increases the controller complexity to reduce the number of computation and memory operations. Typical DSP solutions have a fixed pipeline and processing algorithm. However, by including additional decision making into the controller, unnecessary computations can be avoided. Further, complex architectures requiring sophisticated control, provide a more power efficient implementation, as demonstrated in the subband and PVQ decoder designs.

4.1. Subband High Frequency Data

Subband data from the three highest frequency bands contain a large percentage of zero values which can be exploited to further reduce the power. Our method explicitly reduces the number of operations by representing the high-frequency data with a zero run-length code. Logic circuitry on the PVQ decoder computes the distance between non-zero values, and stores only the run-length and non-zero values. This significantly reduces the buffer sizes and accesses on the PVQ chip. The output FIFO buffers are reduced by a factor of 3, the number of on-chip FIFO buffer accesses by up to a factor of 10, and off-chip accesses by 3 times. Further, the I/O power used in transferring the data to the subband chip is halved because the non-zero values do not always return to zero as would occur if the zeros were transmitted. A signed magnitude numbering system would further reduce the activity [21]. For a Laplacian data source, which approximates subband data, with a variance of 10, the switching activity is reduced by a factor of 2 for 10 bit data. However, this numbering scheme requires additional hardware to convert back to 2's complement

before computation can be performed. The extra area and power for the conversion versus the overall additional savings made the scheme less attractive and was not implemented.

4.2. Skip Processing

The run-length encoding also enables the subband decoder to detect zero data and perform skip processing. Skipping occurs when four HH and HL values are zero (two horizontal and two vertical). The skipping saves the horizontal filtering, the combining of the HH and HL data, and the result storage into the *ld* line buffer. Also, reading the H value from the *ld* line buffer and its vertical filtering are eliminated. The amount of processing cycles per pixel is reduced from 1.98 for all levels and color components to 1.33 resulting in a measured 15% power reduction in the subband decoder chip.

The price of this scheme is increased control complexity. The controller must keep track of zero values within the previous line of subband data. Further, the pipeline now has two types of processing cycles, one with four cycles alternating between LL, LH, HL, and HH data and the other alternating between LL, LH, LL, LH. The sequence type must be passed down the pipeline so each logic unit will function according to the specified cycle type. The overall control flow with these functions is shown in Fig. 10.

4.3. Clock Gating

Both the PVQ and subband decoders take advantage of clock gating to maximize power savings from unused processing cycles. The PVQ decoder has four independent processors, each performing a different task. Each processor is separated by FIFO buffers and only processes when its input FIFO is not empty and its output FIFO is not full. Otherwise, the processor stalls, the controller enters a power-conserving stand-by state, and the clock to that unit turns off. More than half of the chip's total clock capacitance lies in gated clocks. With the exception of the vector decoding unit, whose idle time is typically 10%, the other processing elements are typically idle 50% to 60%. These high idle times lead to savings in total clock power dissipation by a factor of 2 [22].

In the subband decoder, skip processing decreases the number of cycles required to decode the frame thereby increasing the number of idle cycles. An additional 20% of the cycles are idle due to the vertical and horizontal synchronization and blanking times in the video stream. With gated clocks, the power used by the idle cycles will be reduced to near zero.

4.4. Data Interleaving

An important consideration in any subband design is the ordering of the input data stream which greatly impacts the implementation. Previous designs operated

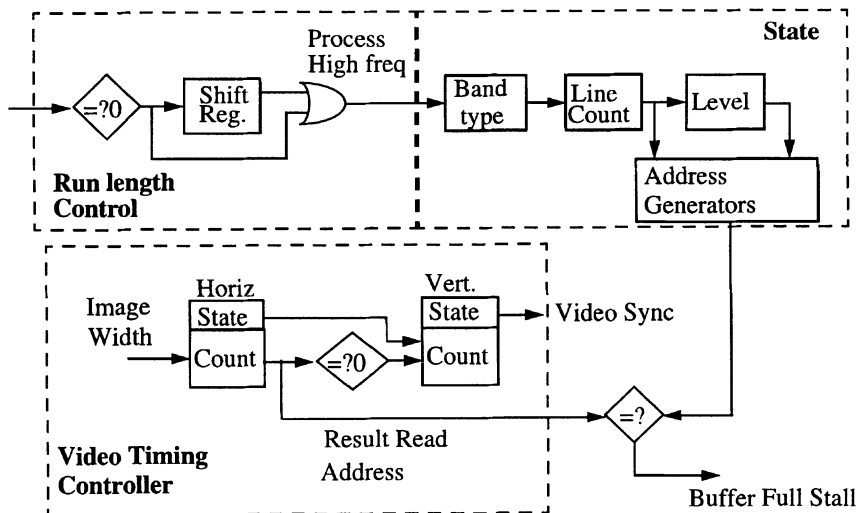


Figure 10. Subband control flow.

on either a single subband level for a 2-D image, or multi-levels for 1-D data [23–26], while this decoder handles multi-levels for a 2-D image, as well as separate YUV color components. The data ordering is designed to eliminate all external memory and minimize the internal storage requirements, allowing it to remain on-chip, to minimize the large power required by memory accesses. To perform multi-level processing, lines of subband data from different levels are interleaved to allow for line-by-line processing. The processing of a single line of data from the four subbands from a given level results in two lines of output due to the upsampling. The interleaving processes only one of these new lines at the next level and does not process the other until all the higher level result lines have been processed. In this way, only two lines of results per level are required. The final output buffer is increased to four lines to prevent underflow. Each color component requires its own storage, and their processing is interleaved so that all components remain synchronized. Because the YUV output is formed in raster scan order, no output frame buffer is needed.

For storage, an SRAM was chosen over a shift-register for several reasons. First, for a programmable image width, the shift register would require a variable shift, while the memory needs no additional adjustments. Further, separate shift registers would be needed for each level and component, while they can be merged into a single memory block. Finally, a memory has additional power advantages by using reduced voltage swings in the bit lines and by not clocking all the storage elements for every access. The memory does require additional control hardware in the form of address generators, though this power is not significant especially given the low switching activity of a counter.

When used in tandem with the subband reconstruction chip, the PVQ decoder chip must incorporate additional control to track the data interleaving between subband levels. Since the chip has four independent datapaths with separate controllers, all four controllers must keep track of the interleaving data with additional counters and logic. The additional control circuitry was considered a good system trade-off for the significant power savings achieved by eliminating the frame buffer.

5. High Throughput Implementation

A final important design approach trades off excess throughput for lower power by reducing the supply

voltage. The energy per op is proportional to V^2 , while the gate delay is proportional to $V/(V - Vt)^2$. Thus, the voltage can be reduced until the frequency of operation just delivers the required operation throughput while dramatically reducing the energy/op. With increased parallelism and pipelining, lower frequencies will provide the same throughput with reduced power [27]. However, the parallelism and pipelining have additional energy overhead which will eventually compensate for the voltage scaling gains, especially at low voltages near $2Vt$ where the delay begins to increase dramatically. Therefore, the algorithm should be mapped to provide high throughput with low overhead.

5.1. PVQ Decoder

The PVQ decoder chip design incorporates parallelism, pipelining, and algorithm recasting to achieve greater performance. The chip has four independent processing elements (stream parser, index pre-decoder, vector decoder, multiplier), each with its own independent control and each individually pipelined (shown in Fig. 11). Because the PVQ decoding algorithm is inherently non-deterministic, i.e., the number of processing steps to decode a PVQ index depends on the vector data, the latency in the index parser and vector decoder units is also non-deterministic. Dividing the chip into four processors, separated by FIFO buffers, decouples the various function blocks and maximizes the chip throughput.

The stream parser controls input dataflow into the chip and parses incoming 16-bit words into PVQ indices, scaling factors, and scalar-quantized values. Next, the index pre-decoder (16-bit datapath) decodes each PVQ index into four intermediate subindices which characterize the vector to be decoded. This includes the number of non-zero vector values, the position of the non-zero elements, and relative magnitude of the non-zero elements, and finally the sign of the non-zero values. The vector decoder takes these four characteristics from the index pre-decoder and generates a data vector. Finally, a 6×8 -bit pipelined multiplier generates the final output by multiplying the decoded vector elements with the scaling factor accompanying each PVQ index.

The critical path of the PVQ decoding, found in the vector decoding unit, is optimized for higher throughput. Increasing the performance of this unit directly increases chip throughput, allowing for lower voltage operation, and also reduces the amount of output buffering

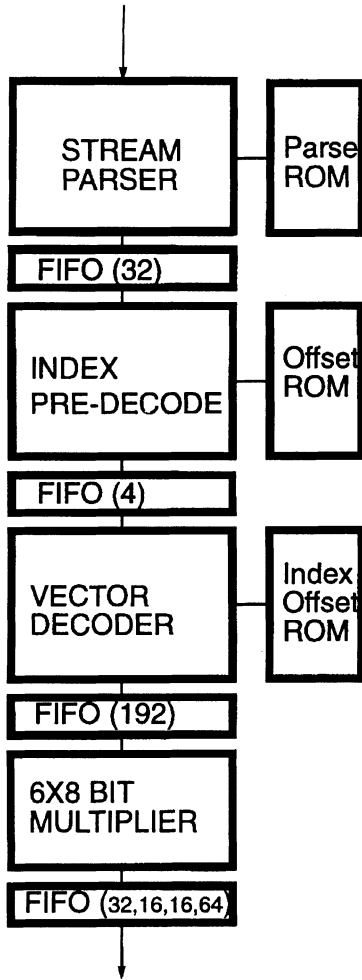


Figure 11. PVQ decoder architecture.

required to meet the real-time constraints. Direct implementation of the PVQ algorithm requires a linear search to locate and compute the correct index offset and decode the vector. Improved throughput is achieved by searching and processing a block of 4 combinatorial offsets at a time [22]. For typical image data, this reduces the average number of search iterations from 15 to 3, and halves the number of processing cycles and the amount of output buffering.

These optimizations led to increased performance at low voltages and low clock rates [28]. The chip operates at 1.35 V at 6.4 MHz to perform real-time video decoding (30 frames/sec) for a display size of 176×240 YUV pixels operating at a peak computation rate of 21 MOPS (operations include shifts, subtracts, multiplies, compares, ROM accesses, and adds.) Figure 12 shows the measured power dissipation of the PVQ decoder chip at maximum operating frequencies for a range of

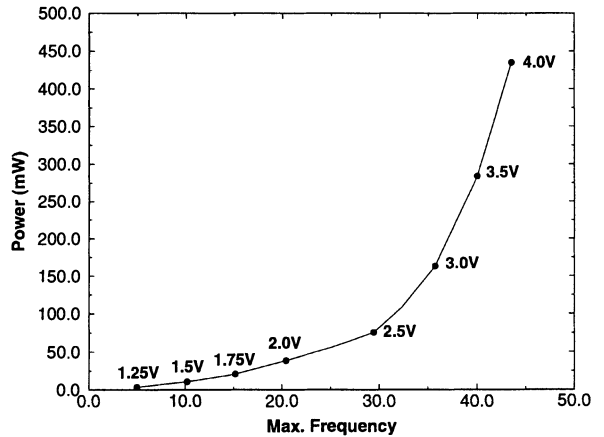


Figure 12. PVQ decoder measured power vs. voltage.

supply voltages. As the graph shows, the chip operates at 1.35 V at 6.4 MHz to perform real-time video decoding while dissipating 6.7 mW.

5.2. Subband Decoder

The subband decoder takes advantage of the natural parallelism inherent in the algorithm, as well as pipelining to improve throughput. Figure 13 illustrate the overall architecture of the subband decoder chip. A small input buffer stores the subband data and passes it to the horizontal filter unit. The line delay memory stores the horizontal filter output and sends them to the vertical filter followed by the scale unit. Lower level results go to the intermediate result memory where they are passed back to the input buffer for reconstruction of the next level. Top level subband results are stored in the final result memory buffer before conversion from YUV to RGB color space. The RGB results are sent off chip to a digital to analog convertor (DAC) and then to the display.

The architecture includes three concurrent processing units consisting of the horizontal and vertical filters and the color converter. The filters simultaneously create two outputs from the input data due to the upsampling. The even filter coefficients generate one output, while the odd coefficients generate the other. Furthermore, the computation operations are deeply pipelined to perform an add per one and a half cycles through cycle stealing. The three separate memories all operate independently and concurrently, matching the output of the filters. This scheme provides significant throughput, allowing the real-time constraints to be met with a voltage as low as 1 V as shown in Fig. 14. Figure 15

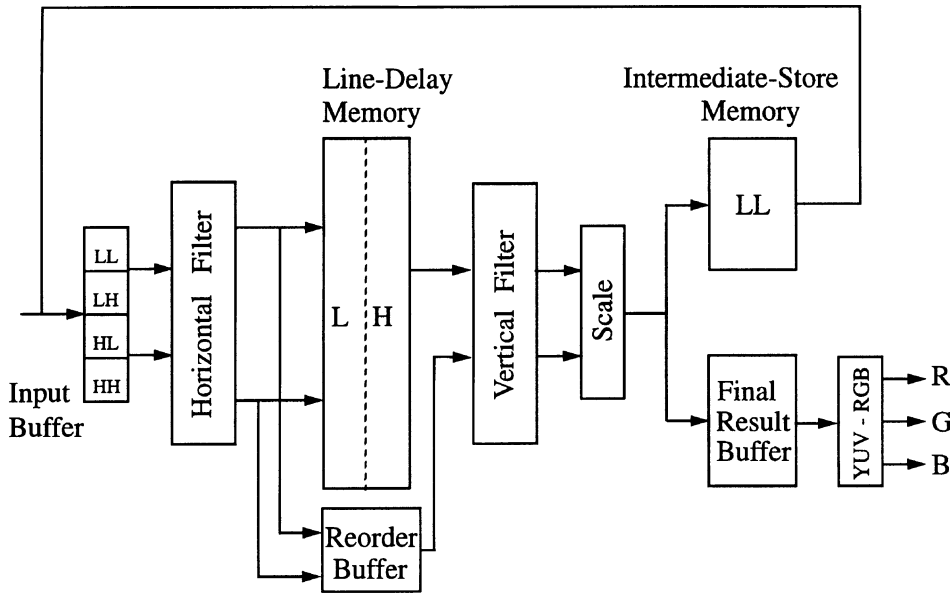


Figure 13. Subband decoder datapath architecture.

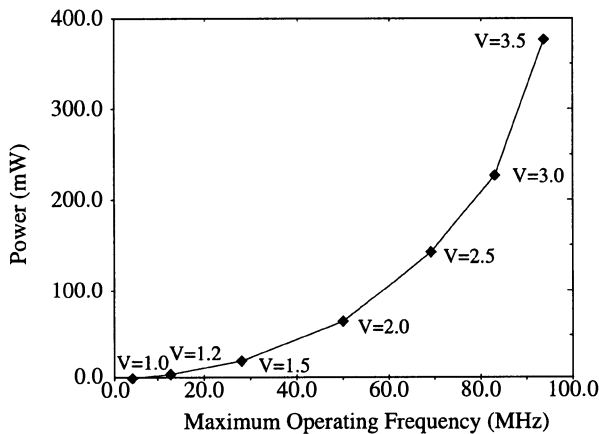


Figure 14. Frequency vs. power for subband decoder.

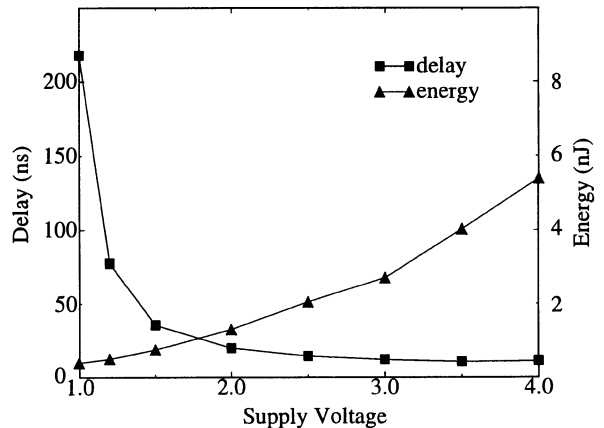


Figure 15. Energy and delay vs. voltage for subband decoder.

illustrates how, as the supply voltage drops, the chip delay increases while the energy to perform the function drops. The voltage is dropped until the delay just meets the throughput constraints, giving the minimum energy to perform the necessary task.

6. Conclusion

In the design of this low power system for portable video decoding, we used high level power estimation to guide the entire design process. Decisions were made by evaluating their effect upon the type

and frequency of operation requirements. Table 4 illustrates the subband decoder's operation throughputs and resulting power estimates, along with the measured power consumption [29]. The measured results closely match the estimates validating the high level, operation based design approach. As predicted, most of the power goes into computation and storage (memory and FIFOs) while control power, despite the additional complexity, remains a small overall factor.

The custom algorithm design delivers the majority of the power reductions. Various trade-offs between compression performance and power consumption were

Table 4. Estimated and measured power for subband decoder.

Operation	Mops/ sec	Energy/op (pJ)	Estimated power (mW)	Measured power (mW)	% of total
Total datapath			0.35	0.34	29
Add (16 bits)	17.7	7	0.12		
3-2 add (16 bits)	25	2	0.05		
Latching (16 bits)	100	1.8	0.18		
Internal memory			0.26	0.39	33
Internal read (16 bits)	2.4	36	0.09		
Internal write (16 bits)	2.4	71	0.17		
External access (16 bits)	2.7	80	0.22	0.38	31
Control			0.13	0.09	7
Total			1.0	1.2	100

made. Inter-frame coding was rejected as too power hungry despite the higher compression ratios. Subband coding provided a low power solution with higher subjective quality than the DCT. A small amount of quality was sacrificed for less power with the selection of a low complexity, highly efficient filter. In the quantization scheme computation was chosen over the more power expensive memory by using PVQ. Further savings resulted from the architecture and implementation. By increasing the controller complexity, operations and memory size and accesses were reduced by skip processing and data ordering. Programmability and overhead were kept to a minimum and only in sections which contributed little to the overall power such as the video controller. Finally, the implementation delivered high throughput, through parallelism and pipelining, which was traded-off for lower power by reducing the supply voltage. This portable video-decoder demonstrates the effectiveness of these comprehensive power optimizations, ranging from algorithm design to implementation, required for designing a very low power system.

References

1. N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*, Reading, Addison-Wesley, 1993.
2. B. Amrutur and M. Horowitz, "Techniques to reduce power in fast wide memories," *Proc. 1994 Symposium on Low-Power Electronics*, Vol. 1, pp. 92-93, Oct. 1994.
3. S. Molloy and R. Jain, "System architecture optimizations for low power MPEG-1 video decoding," *1994 IEEE Symposium on Low Power Electronics*, Vol. 1, pp. 26-27, Oct. 1994.
4. B.G. Lee, "A new algorithm for the discrete cosine transform," *IEEE Trans. Acoust., Speech, and Signal Process.*, Vol. ASSP-32, pp. 1243-1245, Dec. 1984.
5. S. Uramoto et al., "A 100-MHz 2-D discrete cosine transform core processor," *IEEE Journal of Solid State Circuits*, Vol. 27, pp. 492-499, April 1992.
6. A. Madisetti and A. Willson, "A 100 MHz 2-D 8×8 DCT/IDCT processor for HDTV applications," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 5, pp. 158-165, April 1995.
7. P. Ruetz and P. Tong, "A 160-Mpixels/s IDCT processor for HDTV," *IEEE Micro*, pp. 28-32, Oct. 1992.
8. J. Woods (Ed.), *Subband Image Coding*, Kluwer Academic Publishers, Boston, 1991.
9. A. Akansu, "Multiplierless suboptimal PR-QMF design," *SPIE*, Vol. 1818, pp. 723-734, Nov. 1992.
10. T. Senoo and B. Girod, "Vector quantization for entropy coding of image subbands," *IEEE Transactions on Image Processing*, Vol. 1, pp. 526-533, Oct. 1992.
11. J. Villasenor et al., "Wavelet filter evaluation for image compression," *IEEE Transactions on Image Processing*, Vol. 4, pp. 1053-1060, Aug. 1995.
12. P. Vaidyanathan, "Multirate systems and filter banks," PTR Prentice-Hall, Englewood Cliffs, 1993.
13. A. Lewis and G. Knowles, "VLSI architecture for 2-D daubechies wavelet transform with-out multipliers," *Electronic Letters*, pp. 171-173, Jan. 1991.
14. A. Gersho and R.M. Gray, "Vector quantization and signal compression," Kluwer Academic Publishers, Boston, 1992.
15. H. Westerink et. al., "Subband coding of color images," Chap. 5 from *Subband Image Coding*, Kluwer Academic Publishers, Boston, 1991.
16. T. R. Fischer, "A pyramid vector quantizer," *IEEE Trans. Inform. Theory*, IT-32, pp. 568-583, July 1986.
17. E.K. Tsern and T. Meng, "Image coding using pyramid vector quantization of subband coefficients," *Proceedings ICASSP 1994*, Vol. 5, pp. 601-604, April 1994.
18. E.K. Tsern, A.C. Hung, and T. Meng, "Video compression for portable communication using pyramid vector quantization of

subband coefficients," *1993 IEEE Workshop on VLSI Signal Processing*, pp. 444–452, Oct. 1993.

19. T. Meng, B. Gordon, E. Tsern, and A. Hung, "Portable video-on-demand in wireless communication," *Proceedings of the IEEE*, Vol. 83, pp. 659–680, April 1995.
20. B. Gordon, N. Chaddha, and T. Meng, "A low power multiplier-less YUV to RGB converter based on human vision perception," *1994 IEEE Workshop on VLSI Signal Processing*, pp. 408–417, Oct. 1994.
21. A. Chandrakasan and R. Broderickson, "Minimizing power consumption in digital CMOS circuits," *Proceedings of the IEEE*, Vol. 83, pp. 498–523, April 1995.
22. PE. K. Tsern and T. Meng, "A low power video-rate pyramid VQ decoder," *1996 ISSCC Digest of Technical Papers*, Vol. 39, pp. 162–163, Feb. 1996.
23. M. Winzker et al., "VLSI chip set for 2 D HDTV subband filtering with on-chip line memories," *IEEE Journal of Solid-State Circuits*, Vol. 28, pp. 1354–1361, Dec. 1993.
24. G. Van Der Wal and P. Burt, "A VLSI pyramid chip for multiresolution image analysis," *International Journal of Computer Vision*, pp. 177–189, Sept. 1992.
25. M. Vishwanath and C. Chakrabarti, "A VLSI architecture for real-time hierarchical encoding/decoding of video using the wavelet transform," *Proceedings ICASSP 1994*, Vol. 2, pp. 401–404, April 1994.
26. J. Kowalczyk et al., "A VLSI filter architecture for digital HDTV codecs," *1992 IEEE International Symposium on Circuits and Systems*, Vol. 3, pp. 1077–1080, May 1992.
27. A. Chandrakasan et al., "Low-power CMOS digital design," *IEEE Journal of Solid-State Circuits*, Vol. 27, pp. 473–484, April 1992.
28. E.K. Tsern and T. Meng, "A low power video-rate pyramid VQ decoder," to appear in *Journal of Solid State Circuits*.
29. B. Gordon and T. Meng, "A 1.2 mW video-rate 2 D color subband decoder," *IEEE Journal of Solid-State Circuits*, Vol. 30, pp. 1510–1516, Dec. 1995.



Benjamin M. Gordon was born in State College, PA in 1965. He received the B.S. degree in electrical engineering from MIT in 1987 and the M.S. degree from Stanford University in 1992. From 1987

to 1991 he worked as a systems engineer for Advanced Processing Labs, Inc. in San Diego on real-time signal processing systems. He is currently finishing the Ph.D. degree at Stanford University where he is working on low power video compression for portable applications.



Ely K. Tsern was born in Park Ridge, IL, on January 29, 1966. He received the B.S.E.E. degree from University of California, Berkeley, in 1987. After working at Hewlett Packard in digital microwave from 1988 to 1989, he attended Stanford University, where he received the M.S. degree in 1991 and Ph.D. in 1996 in electrical engineering. His research focused on video compression for portable applications, signal processing, and low power VLSI design. He is currently a member of the technology group at Rambus, Inc., working on high-speed DRAM architectures.



Teresa H. Meng joined the faculty of the Stanford University in 1988, where she is an Associate Professor in the Department of Electrical Engineering. Her current research activities include video compression, wireless communication, and low-power design. Among the awards she has received are the IEEE Signal Processing Society's Paper Award in 1989, the 1989 NSF Presidential Young Investigator Award, the 1989 ONR Young Investigator Award, and a 1989 IBM Faculty Development Award. She received a B.S. degree from National Taiwan University in 1983 and an M.S. and Ph.D. from U.C. Berkeley in 1984 and 1988 respectively. She was co-program chair of the 1992 Application Specific Array Processor Conference and of the 1993 HOTCHIP Symposium, and the general chair of the 1996 IEEE Workshop on VLSI Signal Processing.

IC Implementation Challenges of a 2.4 GHz Wireless LAN Chipset

M. CHIAN, G. CROFT, S. JOST, P. LANDY, B. MYERS AND J. PRENTICE
Harris Semiconductor, Melbourne, FL 32901

DOUG SCHULTZ
Integrated RF Solutions, Palm Bay, FL 32909

Received and Revised May 17, 1996

Abstract. The recent introduction of IC technologies offering high frequency transistors with f_t greater than 10 GHz has opened new opportunities for higher integration of wireless communication systems. Fast silicon IC devices make possible the integration of many RF subsystems on a single die and offer a total solution to mixed frequency (low frequency and RF) and mixed signal (analog and digital) systems. This paper describes IC implementation challenges of a 2.4 GHz wireless LAN chipset developed at Harris Semiconductor.

1. Introduction

The rapid growth of the wireless market has created a dynamic environment with an on-going search for lower cost, smaller size, and higher performance RF and IF products. Traditional discrete designs are quickly reaching the physical limits of size, parasitics, and electrical performance. A solution that integrates many RF and/or IF subsystems on a single die promises dramatically smaller size, greater manufacturability, and in many cases higher performance. This paper will review some of the IC design and manufacturing considerations encountered during development of a fully integrated wireless local area network (WLAN) chipset. An overview of the system requirements for WLAN based on the IEEE 802.11 standard is provided, followed by some of the partitioning and technology considerations necessary for IC realization. Following the system overview, some specific design and manufacturing considerations for critical components of the chipset are provided. Finally, some of the computer aided design tools (CAD) necessary for the chipset development will be reviewed.

2. System Overview and Technical Requirements

The emerging IEEE 802.11 WLAN standard allows for interoperability of wireless products, and utilizes two spread spectrum modulation techniques, frequency hopping (FH) spread spectrum and direct sequence (DS) spread spectrum. The data rate of FH systems is 1 MBPS, with 2 MBPS optional. The data rate of DS systems is both 1 and 2 MBPS. Both spread spectrum options operate in the 2.4 GHz Industrial Scientific Medical (ISM) band.

FH systems change the carrier frequency in a pseudorandom (PN) fashion. A typical FH system consists of a transmitter with a frequency shift keying (FSK) data modulator, followed by a synthesizer that centers the narrow-band FSK waveform on the desired frequency channel. The synthesizer is driven by a PN code generator. An FH receiver includes a synthesizer, PN code generator, and an FSK demodulator. The transmitter and receiver share a common, synchronized PN code. By utilizing different PN codes with orthogonal hopping sequences, multiple systems can operate simultaneously over the same bandwidth. Narrowband

noise affects the system only during the dwell time at that frequency.

In contrast to FH systems, DS systems spread their energy by rapidly changing phase, so that the signal is continuous for brief time intervals called chips. These chips are several times shorter than the actual data bits. A second modulation for the data is used, and is typically phase modulation. If the bandwidth of the spreading, or chipping signal, is large relative to the bandwidth of the data, the transmission bandwidth is dominated by the spreading signal and is nearly independent of the data signal. Therefore, the energy of the DS system is not confined to the bandwidth of the original phase shift keyed (PSK) narrow band signal, but is distributed over the spreading bandwidth defined by the chipping rate.

DS has several operational benefits as compared to FH. For example, due to its lower power spectral density, the DS system can operate at, or below the noise floor of a given environment. One implication of this is DS waveforms are not as disruptive to other communications on the same frequency. Also, multiple DS systems can operate simultaneously in the same channel if each system uses a different spreading code. The characteristics of the spreading code are critical because the network code set must exhibit low cross-correlation characteristics among its individual codes. For the sake of interoperability in the IEEE 802.11 DS standard, only one 11 bit spreading code is used. Another benefit in DS is the received signal to noise ratio (SNR), which is a function of the ratio between the data rate and the chip rate, can be negative. The higher the

chip rate is with respect to the data rate, the lower the signal level, and the more negative the spread signal's SNR will become. This ratio of the two rates is referred to as processing gain (PG). PG means a DS receiver can still recover a signal even when the interfering signal has a higher SNR than the desired spread waveform. Using DS, not only is the signal of interest amplified, but also every signal that does not correlate with the spreading code is attenuated.

To successfully despread the received signal, the receiver not only needs to use the identical PN code, but also needs to phase-align it with the incoming received code. Autocorrelation measures the degree of similarity of the signal with its phase-shifted replica over all phases. A PN replica is multiplied with the incoming waveform and the result is integrated to obtain the correlator output. The reference code is shifted in phase until the output of the integrator is maximum.

Architectural Considerations and Trade-offs

There are several issues to consider when selecting a WLAN architecture. Because many systems will be used with notebook computers in a PC Card environment, size (including thickness), power consumption, and cost are all key drivers. Performance parameters such as range, throughput, and interference immunity are also important.

The DS architecture implemented in Harris Semiconductor's Prism™ chipset [1] is shown in Fig. 1. It features a single conversion receiver, with limiting IF processing, and analog demodulation to base-band.

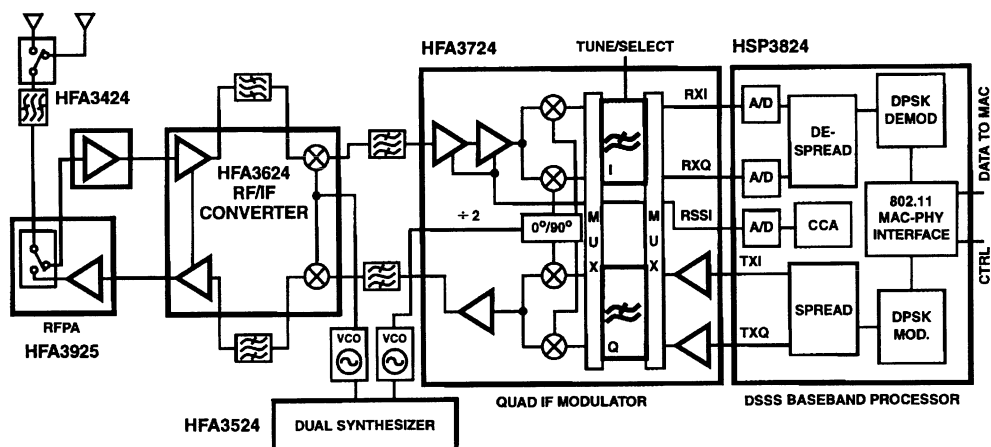


Figure 1. System architecture.

Digital signal processing (DSP) is used to despread and demodulate the signal.

The single conversion architecture was chosen to minimize the complexity of the chipset. More than one conversion would require additional local oscillators and subsequent IF filter and amplifier stages. The front-end gain is sufficient to accommodate moderate IF filter losses. The IF frequency range that can be accommodated reaches up to 400 MHz, easing the front-end image filtering requirements.

Limiting IF amplifiers may be used with DS systems, and are especially suited for systems with short PN codes. With the 11 bit code specified by the IEEE 802.11 standard, the processing gain is modest, resulting in positive signal to noise ratios at the limiter input while operating. As a consequence, limiting IF systems actually perform slightly better than automatic gain (AGC) IF systems in most environments. The use of limiting IF amplifiers lowers system cost and complexity, because AGC IF systems are difficult to implement in a WLAN environment where fast acquisition (less than 1 μ sec) of packets is required.

The transition between analog signal processing and digital signal processing was carefully chosen to exploit the advantages of DSP, while avoiding the high supply current penalty commonly associated with IF based DSP. Using DSP techniques at baseband reduces system cost. For example, the second LO, used to provide quadrature demodulation in the receive path, is not part of a carrier recovery loop. Instead, a novel PSK demodulator achieves performance levels close to those of a complex coherent system, without the burden of an external feedback loop.

Other DSP techniques that improve system performance include the despreading and demodulation scheme, which is optimized to exploit the impact of the limiting IF amplifiers. This results in improved receiver sensitivity at a given bit-error rate (BER). Furthermore, a DSP based leveling circuit is used to provide control of the received baseband signals, maximizing the dynamic range of the A/D converters.

The transmit processing follows a similar single conversion flow. Starting at baseband, quadrature digital PSK modulated signals are generated, and then spread. The baseband signals are then low pass filtered to meet the transmit spectral mask specified in the IEEE 802.11 standard. After this filtering, the remaining transmit components must be operated in a linear fashion to avoid spectral regrowth. Quadrature IF modulation is used to convert the baseband signals to the IF frequency.

A further up conversion then shifts the carrier up to the desired channel in the 2.4 GHz ISM band. To develop the required output power, an integrated RF power amplifier is used. As previously discussed, this power amplifier must be operated in a linear manner, with a fixed back-off from its 1 dB compression point to avoid spectral regrowth.

System Requirements

Because IEEE 802.11 compatible systems operate in the 2.4 GHz ISM frequency band (2.4–2.483 GHz), no license is required for operation. This allows for simple installations, and ease in changing existing installations. Although the IEEE 802.11 specification is complex, a few of the key physical layer requirements merit consideration.

On the receive side, the required sensitivity is -80 dBm for a frame error rate (FER) of $8 \cdot 10^{-2}$, using differential quadrature (DQPSK) modulation. With a required bit to noise energy ratio (E_b/N_0) of approximately 16 dB to meet the required FER, an 18 dB front-end noise figure is needed to meet this specification. In this implementation however, the goal for receiver sensitivity was -90 dBm, requiring a front-end noise figure of 8 dB. Since there is approximately 3.5 dB of loss before the active receive components, the receive noise figure requirement at the low noise amplifier is actually 4.5 dB.

On the transmit side, the desired output power is $+16.5$ dBm. Because there is approximately 3.5 dB of loss after the power amplifier, the amplifier must operate at $+20$ dBm. After allowing for 3 to 4 dB of back-off from its 1 dB compression point, the required amplifier compression point is $+23$ to $+24$ dBm. Also, the transmit spectral mask requires the first side-lobe of the PSK signal to be suppressed at least 30 dB with respect of the main-lobe. An unfiltered PSK signal would have the first side-lobe only 13 dB below the main-lobe. This requirement drives the choice of the baseband transmit low pass filter, and sets the amount of back-off from compression allowable in the transmit chain.

System Partitioning and Process Technology Considerations

After the architectural issues are finalized, the task of partitioning the integrated circuits is addressed. The available process technologies determine system

partitioning. In this case, the processes available that are appropriate for this chipset are, a high speed bipolar process, a mixed signal BiCMOS process, and a digital CMOS process. The high speed bipolar process (UHF1X) is best suited for small to medium scale integration of RF and IF functions, and features an NPN f_{\max} of 13 GHz, and a PNP f_{\max} of 7 GHz. This complementary process is built using dielectrically isolated, bonded wafer techniques. By controlling the features of the active and handle wafers, the high frequency losses commonly associated with silicon can be improved. The BiCMOS process (HBC10) is best suited for medium to large scale integration of IF and baseband analog functions, with a small digital content. HBC10 features complementary bipolar and MOS devices, with NPN f_T of 4 GHz. The digital CMOS process is a high density 0.6 micron triple level metal technology ideal for DSP integration.

Besides process choice, other significant factors affecting system partitioning are passive components requiring interfacing and isolation issues. For example, if off-chip filters are required, it may be appropriate to partition circuits around these external components, especially in the case of filters needing high isolation, such as the receive IF channel selection filter. High gain stages also require high isolation. Limiting IF amplifiers, with over 80 dB of gain, require isolation from output to input of significantly greater than 80 dB. These type of isolation requirements affect chip partitioning as well as packaging options.

As this system operates in a half-duplex mode, transmit and receive functions can be integrated together with little concern for isolation. Similar frequency processing is grouped together, as the process most appropriate for a given frequency can then be partitioned optimally.

The RF/IF converter is partitioned separately from the IF sections to allow for high isolation in the receive IF filter. The off-chip image filters associated with the RF/IF converter require only modest isolation for proper operation. The local oscillator (LO) is shared between both receive and transmit sections of the RF/IF converter. The RF/IF converter is fabricated with the high speed UHF1X bipolar process.

The IF QMOD is packaged in an 80 TQFP package that was selected to achieve high isolation for the limiting IF amplifier sections. The IF QMOD is a four chip assembly, with all the die sharing a common leadframe. The IF limiter function is split into two identical die to allow for an external interstage filter, and to assure

stable operation. The limiter die are fabricated with the high speed UNF1X bipolar process. The quadrature up and down converter functions of the IF QMOD are implemented with a third UHF1X die. The high speed of the UHF1X process is needed to achieve good phase balance at frequencies up to 400 MHz. The baseband low pass filter functions of the IF QMOD are implemented with the fourth die, using the HBC10 mixed signal process. As these filters are moderately complex, and the frequency is relatively low, the mixed signal process is more appropriate for this function.

The RF power amplifier is partitioned as a separate IC due to thermal issues, as well as the large signal nature of the circuit. The baseband processor is partitioned as a separate IC due to the large digital content of this function, and is fabricated with the digital CMOS process.

3. Description of Major Blocks

RF/IF Converter

The integrated RF/IF converter incorporates on-chip spiral inductors and MOS capacitors to provide 50 ohm internal matching on all high frequency ports, as well as higher impedances for the IF ports, thus supporting simple connection to IF filters. No IF baluns are required. One LO input is needed, with internal connections between the transmit and receive mixers. The IF passband extends well beyond 400 MHz.

In the receive path, a two-stage LNA establishes the receiver noise figure. An optional external image rejection filter can enhance overall system sensitivity. The single-balanced receive mixer is optimized for high conversion gain, low noise figure, and high third-order intercept. The IF output is a differential structure that supports IF impedance matching networks. Optionally, the IF output can be used in a single-ended fashion.

In the transmit path, the RF/IF converter uses a double-balanced up-conversion mixer to minimize the amount of LO leakage in the transmit output. The chip allows the use of an external sideband selection filter, with characteristics similar to those in the receive image-reject filter. An on-chip two-stage exciter amplifier eases RF power amplifier gain requirements.

IF Filtering

Both the receive and transmit IF paths use filtering. A highly selective SAW filter provides receiver

selectivity. At the recommended 280 MHz IF, and an 11 Mcps chipping rate, the filter bandwidth should be approximately 17 MHz. Insertion loss should be less than 15 dB, and differential group delay should be less than 100 ns.

Frequency Synthesis

In both transmit and receive modes a dual-frequency synthesizer provides the LO signal for both the RF/IF Converter and the phase splitter in the IF/QMOD. By maintaining identical IF frequencies in both transmit and receive paths there is no frequency switching and therefore no settling time requirements. This enhances transmit/receive turnaround time, a key issue in carrier sense multiple access (CSMA) data applications.

RF Power Amplifier

The linear RF power amplifier provides matched 50 ohm characteristic impedances and provides +23 dBm at 1-dB compression point. To limit spectral regrowth, the amplifier operates 3 dB below the 1-dB compression point. Assuming 3.5 dB insertion loss for the antenna diversity scheme, +16.5 dBm of transmit power is available at the antenna.

IF QMOD

In the IF QMOD, a two-stage limiting amplifier in the receive path provides sufficient gain and bandwidth to exhibit a -84 dBm limiting sensitivity at frequencies up to 400 MHz. The limited output is typically 200 mV, and is compensated over temperature. An internal RSSI (Received Signal Strength Indicator) provides linear temperature-compensated performance. The RSSI signal is routed to the internal 6-bit RSSI A/D converter on the baseband processor, where it is used for Clear Channel Assessment (CCA).

Following limiting, the IF signal is routed to a quadrature demodulator featuring an internal, quadrature LO network that achieves accurate phase performance. The quadrature network utilizes a divide-by-two approach to achieve broadband operation. Feedback circuits maintain phase accuracy over a wide range of LO input levels and duty cycles. The demodulator input compression point exceeds 1 V_{pp}, making it suitable for use with the limiter output, or with any external AGC, should system designers wish to bypass

the on-board limiters. The I/Q baseband signals exhibit ± 0.6 degree phase balance and ± 0.2 dB amplitude balance. On the transmit side, the quadrature modulator utilizes the same quadrature LO network and provides an accurate IF output from 10 to 400 MHz. As a result of the excellent phase and amplitude balance, sideband suppression in a single sideband (SSB) operating mode is typically 33 dB.

The IF QMOD also provides programmable baseband *I* and *Q* low pass filtering. Dual fifth-order Butterworth filters are internally multiplexed between the transmit and receive channels. These filters offer four digitally-selectable cutoffs: 2.2, 4.4, 8.8, and 17.6 MHz. These cutoffs correspond to DS chip rates of 2.75, 5.5, 11, and 22 Mcps. In addition, the filters may be tuned up to 20% above or below the fixed cutoffs by changing an external resistor. The filter response was selected to meet the transmit spectral mask requirements. Specifically, the first side-lobe attenuation of the transmitted spread DBPSK and DQPSK signal is typically -35 dBc relative to the main-lobe. In receive mode, the demodulated, filtered *I* and *Q* signals are routed to the baseband processor. In transmit mode, the digital *I* and *Q* signals from the baseband processor feed to the IF QMOD. To avoid spectral regrowth, once the transmit single-bit inputs are filtered by the fifth-order Butterworth filters, the rest of the transmit chain must operate linearly. In other words, all further transmit elements must be operated backed-off from their 1 dB compression points. Despite this characteristic of BPSK and QPSK modulation, the improved receiver performance over simpler non-coherent modulations such as Gaussian frequency shift keying (GFSK) results in an overall system performance advantage, especially at high data rates.

Baseband Processor

In the baseband processor, the analog *I* and *Q* signal outputs from the lowpass filters are digitized by 3-bit A/D converters at 22 Msps, twice the chip rate. The quantized *I* and *Q* baseband paths are correlated against a reference PN code, using separate matched filter correlators. The reference PN code is programmable from 11 to 16 chips. The correlators despread information of interest back to its original data rate while spreading interfering signals and noise. After the despreading, the *I* and *Q* signals are converted to polar form, and the phase information is subsequently processed by the DPSK demodulator, which supports both

DBPSK and DQPSK. A digital phase locked carrier tracking loop allows coherent DPSK data processing. In the transmit mode, the baseband processor functions as a DPSK modulator, including a data scrambler, and a BPSK spread modulator.

4. RF/IF Converter Design Challenges

With the system considerations complete, it is interesting to visit some of the specific challenges associated with two key components of the WLAN chipset. Both the RF/IF converter and the IF QMOD parts presented particularly challenging design and manufacturing goals.

RFIC Paradigm Shift

With IC applications approaching several GHz, IC designers are advancing into RF design. On the other side, the appeal of monolithics is encouraging the traditional RF designer to use GaAs and silicon foundries for the design of ASICs. For the silicon IC designer, perhaps the biggest challenge involves learning the language and customs of RF design. For the RF designer, the challenges relate to understanding the advantages and the limitations of the IC process while adopting design techniques appropriate for the new media.

Many of the limitations of IC processes are well known. These involve the inability to economically integrate high value capacitors and resistors, relatively poor (>10%) absolute error tolerances of component parameters, and limited choice of device types. In addition to these limitations, the RFIC designer must also be concerned with parasitics associated with the substrate, the general lack of high quality passive components, and package parasitics. While somewhat formidable, these challenges can be met head on by understanding and working around the limits of the present technologies and by driving the development of new and better technologies.

Substrate Parasitics

Historically, integrated circuits that operate at microwave frequencies have been implemented on semi-insulating monocrystalline Gallium Arsenide or on insulating hybrid substrates. Both technologies are expensive and generally produce circuits of lower device

density compared to the cost and density of silicon planar technology.

The availability and performance of microwave IC's implemented in silicon has been limited in part by the high losses that occur in the silicon substrate at microwave frequencies [2]. These losses limit transistor performance and also greatly reduce the Q -factor of integrated inductors. More recently, highly resistive float-zone silicon substrates (HRS), have been shown to have losses nearly as low as GaAs and have been successfully applied at multi GHz frequencies [3]. However, the wafers are expensive and only available in small diameters that are incompatible with modern silicon fabrication facilities.

As mentioned earlier the RF and IF circuits described in this work were implemented in UHF1X, a conventional bonded oxide process. Since the active silicon area is isolated from the silicon substrate by a layer of oxide, it is possible to select both the thickness and the resistivity of the substrate to reduce parasitic effects. The resulting loss and isolation characteristics of the substrate is adequate for circuits operating up to several GHz.

For higher frequencies, improved bonded oxide silicon substrates comparable to GaAs and HRS have been proposed [4, 5]. For example in [5] a so called Dual-Resistivity Substrate is described in which the substrate is divided in two layers with the resistivity and thickness of the two layers optimized to reduce resistive losses and capacitive coupling (crosstalk). Simulation results show the resulting substrate has resistive losses nearly as low as HRS, and lower crosstalk than HRS-while retaining the much lower cost of conventional bonded wafers.

Spiral Inductors

Planar inductors have been used for many years in circuits with insulating or semi-insulating substrates. In the early development of Si IC's, planar inductors were investigated [6], but large chip areas due to lithography limitations, low Q 's limited by substrate losses, and low frequencies of operation led to the conclusion that integrating inductors on chip was impractical. In 1990, the first of three papers [7–9] by Nguyen and Meyer were published showing that it was possible to make usable inductors on silicon integrated circuits. Subsequent papers, Chang et al. [10], and Ashby et al. [11], are illustrative of some of the development work (much of it unpublished) that is now going into the design and

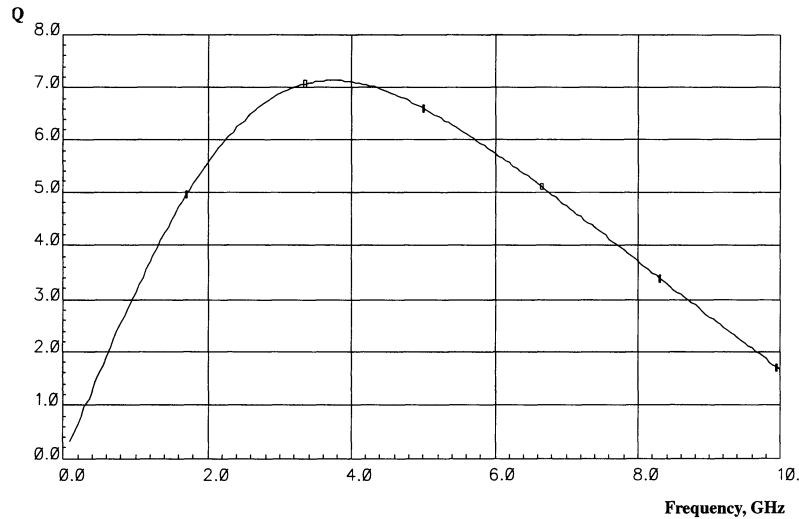


Figure 2. Plot of Q vs. frequency for a 4 nH spiral inductor.

modeling of spiral inductors for use in silicon IC wireless applications.

Several features of the UHF1X process including two level metal, a thick local oxide layer, a 12 μm metal pitch, and low loss bonded oxide substrate, allow for compact, high Q , high frequency inductors. For example, a typical inductor of 4 nH takes up an area of 0.038 mm^2 , has a peak Q of 7 at 3.5 GHz, and its self resonance frequency exceeds 10 GHz, as shown in Fig. 2.

Spiral inductors ranging in value from 0.4 nH to 8.0 nH were used extensively in the design of the 2.4 GHz RF/IF Converter (shown in Fig. 3) for impedance matching and as collector loads. For low voltage operation, the inductor loads not only provide impedance matching, but also allow for voltage swing above the positive supply. To support the design of this and other circuits using spiral inductors, scalable models with associated parasitics and substrate loss terms were developed based on electromagnetic (EM) simulation and lab measurement results. In addition, a parametric layout utility was written and integrated into the Harris Fastrack Design System to make generation of the spiral geometries automatic.

Package Parasitics

Low cost packaging is essential for moderate and high volume commercial wireless products. In many cases the RF performance of standard small outline dual in-line or thin quad flatpack, surface mount packages is adequate (though often not optimum) for circuits up

to several GHz. Customization of the package adds cost and is generally to be avoided unless it can be demonstrated that the standard package alternative is not up to the task.

Given some knowledge of the parasitics for a particular package, it is often possible to assign pin functions and/or to modify the design and layout of the circuit to minimize the impact and in some case to take advantage of the parasitics. For example, to reduce crosstalk between signal leads, it is common practice to assign one or more ground leads between the leads in question and/or to assign the leads to opposite sides of the package. To lower ground inductance, it is also common to assign multiple leads to ground and in some cases to make use of the package die attach paddle as a ground plane. Finally for impedance matching applications, the clever designer will often utilize bondwire or package lead parasitics in the design of matching networks to save on internal and external components.

To quantify these effects it is essential to have accurate electrical models for the package and for the network of bondwires that connect the die to the package. When low ground impedance is an objective, there may be multiple connections (downbonds) between the die and the package leads to the die attach paddle. In this case it may also be important to model the distributed nature of the package paddle, so that the ground inductance and interaction between ground connections is accounted for. These models can then be used in circuit simulations where various trade-offs involving pin assignment, circuit design, and bondpad placement can be analyzed.

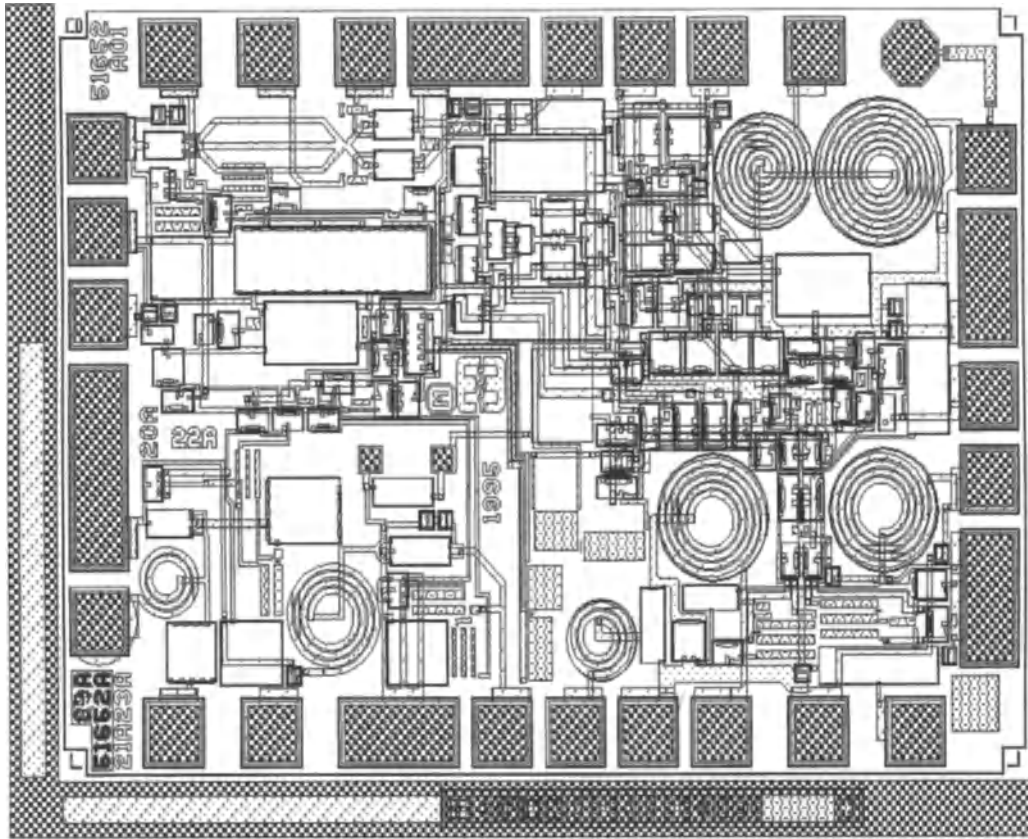


Figure 3. Die plot of the 2.4 GHz RF/IF Converter.

Much of the published and unpublished work involving package models involves utilizing electromagnetic simulation in conjunction with measurements to verify and fine tune the resulting model [12–14]. Proceeding in this manner it is possible over time to build up a library of models for standard packages.

5. IF QMOD Design Challenges

The IF portion of the wireless LAN chipset is a single 80 in TQFP comprised of four independent integrated circuits to make up the processing chain (see Fig. 1 for reference). The operating range for the IF QMOD is 10–400 MHz on the IF side, with programmable data rates up to 4 Mbit/s on the base-band side. The primary technical challenges in the IF network revolved more around operating conditions and precision than high frequency concerns. Wide supply range (2.7–5.5 V), isolation, low distortion, excellent gain/phase balance, and high gain are key

requirements that are difficult to achieve. Part of the solution was partitioning into a multi-chip architecture, but this adds its own set of challenges. These considerations are evident by exploring some of the design challenges for the individual integrated circuits.

Limiters

The receive side of the IF processor utilized a two die implementation for the limiter function. The limiter circuitry must provide most of the receiver gain (<80 db), low noise figure (<7 db), low output limiting level variation (<0.5 db), and an accurate receive signal strength indicator (RSSI).

The most difficult design challenge was to implement the 80 db of gain at 400 MHz and maintain stability. The first task was to find a packaging option which would give the needed isolation (>100 db) at 400 MHz. It was determined through extensive RF testing of different packages and pinout options that an 80

pin TQFP, with limiter inputs and outputs on opposite sides and multiple grounds, was capable of achieving the required isolation. Physical die size and bond wire limits, meant that to keep the limiter inputs and outputs on opposite sides of the package, required two dice. Other benefits of a two dice approach includes less gain per die for increased stability, and allowance of interstage filtering. Because of the large limiter gain and required external components, it is inevitable that the output will be fully limited on noise. For best performance, it is important that the noise causing the limiting be primarily from the RF front-end and not the first limiter. A low limiter noise figure is a must, but the wide band-width (400 MHz of the first limiter compared to the RF front-end 20 MHz SAW) has the potential to degrade input sensitivity. Hence, interstage filtering can be used to narrow the first limiter's bandwidth and improve sensitivity. The narrower bandwidth also has the consequence of increasing the effective gain of the second limiter which will improve the logarithmic response of RSSI as discussed later.

Gain stability and sufficient overall gain are obvious limiter needs, but gain variability due to temperature and process variation is also important. Too much gain will eventually lead to stability issues, but more constraining is the requirement for an accurate receive signal strength indicator (RSSI). RSSI gives an absolute value indication of input signal strength at the limiter input. It is used as one sensor for a carrier sense multiple access (CSMA) networking scheme. By monitoring RSSI, a clear channel assessment of the environment can be made to determine when it is feasible to transmit. Multiple thresholds may be used for RSSI, thus it must be accurate and its slope constant. The baseband processor uses a 6 bit A/D to convert the RSSI signal from the limiter. Noise limiting and offsets will set the minimum RSSI level, while practical A/D constraints within the 2.7 V supply range will limit the full scale level. For this reason, RSSI must compress the large limiter dynamic input range. This logarithmic converter function can be implemented using successive detection techniques, where cascaded limiting stages and detector circuitry are used to produce the compressed signal. The complete details of log converter design will not be addressed, but some of the design challenges are readily apparent including the need for accurate gain.

It can be shown that by cascading a set of limiting stages of fixed gain, and detecting and summing the outputs of each, a logarithmic approximation of the

input signal is generated. Further, the logarithmic accuracy of the approximation is determined solely by the gain chosen. Lower gain results in a more smooth, accurate approximation. However, with lower gains it takes more stages to cover a given dynamic range, which will result in higher power and usually lower bandwidth. The very high frequency capabilities of the UHF1X process allow more stages of lower gain for high accuracy, while not compromising power or bandwidth. The issue of improved RSSI response with a narrow filter between limiters can also now be understood. The increased gain of the second limiter occurs because fewer stages are saturated and these added stages contribute to an improved logarithmic approximation. The accuracy of the gain not only effects the log conformity, but also the slope of the log function. To minimize gain variations to the degree needed for RSSI, it is necessary to implement more complicated circuitry than would ordinarily be needed for just a limiter function or a coarse RSSI. High gain bandwidth requirements warrant an open loop configuration for limiting stages, but accuracy can be worse for these configurations and is further compounded by the lower limit supply range of 2.7 V. The costs and test issues associated with a trimming option are unattractive. Therefore, the actual circuitry did implement an open loop topology where design techniques were found to minimize variations.

The design of the limiter implemented balanced differential signal paths throughout to minimize the effects any internal or external coupling might have on stability. DC coupling is used throughout eliminating the need for external coupling capacitors between stages. Stable DC operating points are an issue because the large gain in the presence of device offsets could result in some stages limiting. A negative feedback loop was developed to stabilize DC offsets. This same loop was bypassed and effectively open for AC signals. Offsets cause another issue associated with fast switching from transmit to receive. When in transmit mode the limiter is powered down, upon switching to receive mode the limiter must be active in 2 microseconds to handle the fast packetized data being processed. During this switching and differences in power down to power up operating points, along with offsets, must be overcome. The charging and discharging of large external bypass capacitors can cause excessively long time constants during this transition. For this reason, care was taken to reduce voltage differences and minimum capacitors across differential nodes were utilized to achieve the required switching

speed. The final important specification for the limiter is output limiting level variation. All processing following the limiter, quadrature demodulation and low pass filtering, linearly amplifies the signal with minimal variation. This is necessary to maintain an effective dynamic range for the A/D converters that follows. Output limiting variations are minimized in the limiter through proper biasing and circuit techniques.

Quadrature Modulator/Demodulator

The third die in the IF chain, referred to as the QModem, provides the modulation and demodulation function. A common set of quadrature local oscillator (LO) clocks drive the I and Q mixers of both the transmit and receive sections. There are two separate baseband signals, I and Q , encoded 90° out of phase with each other making up the single IF signal. So the QModem encodes/decodes the two base band signals and provides frequency translation.

The design challenges stem from the requirements for accurate signal manipulation over a broad frequency range and with wide power supply limits. There are also constraining requirements such as limits on cost, i.e., die size, supply current and the number of package pins. Accuracy means that the I and Q channel have matched gain and phase with no DC offsets, and that the LO clocks are precisely 90° in phase at all frequencies. Accuracy also implies the absence of unwanted signals.

Accuracy is achieved by choosing circuits such that the gain matching, phase matching and DC offsets depend predominately on component matching. To achieve 0.5 dB of gain matching requires total interstage matching of better than 6%. To achieve 30 dBc of carrier suppression requires the DC offset to be less than 3% of the applied signal level. To achieve 2° of phase match requires that the LO clock circuits match to within 0.6%. The best component matching, a few tenths of percent, is obtained with resistors, so while the transistors are a necessary element for amplification, by design, resistors set the gain. In addition, extensive use of automated layout tools were used to verify that the parasitic capacitance of the interconnect wiring matched to tens of femtofarads.

A divide-by-2, digital flip-flop, circuit generates the 90° phase shifted clock operating over the 400 MHz bandwidth. A divide-by-4 generally is more accurate, however, this would require an input clock running at 1.6 GHz from a more expensive synthesizer. The

divide-by-4 approach would have increased the supply current by several milliamperes. The divide-by-2 is sensitive to the quality of the input clock signal. Therefore, an additional analog feedback loop was developed to correct for minor distortions of the input clock.

All the internal stages are directly coupled together since it is impractical to use coupling capacitors between successive stages. As is common with most integrated circuits of this type, differential stages are used since they are easily coupled. What is a little different is the absence of level shifting stages between successive gain stages. Common practice is to operate the transistors with positive collector-base voltage, usually the amount of one transistor's V_{BE} . This positive voltage reduces the base-collector and collector-substrate capacitances. A subsequent emitter follower stage then shifts the operating point bias back to the original level and passes on the signal. The emitter follower stage increases the drive current so the preceding stage can operate at reduced current. The disadvantages of using emitter follower level shifters are the additional source of mismatches, added die area, and the reduction in voltage available for biasing circuits such as current sources.

The bonded wafer process chosen (UHF1X) has low base-collector capacitance and very low collector-substrate capacitances with no voltage dependency. Therefore, it was possible to operate each stage with the base and collector at the same DC potential and direct couple without any level shifter stages. The current in each stage was increased by the amount saved from the eliminated level shifter stage and the load resistor values reduced to decrease the node impedance. The net result was a small increase in the bandwidth and a significant die area savings from smaller resistors as well as from the eliminated stages. The saving in overhead voltage was used to increase the impedance and matching of the biasing current sources to achieve the required signal accuracy.

Both transmit and receive baseband signals and the IF input signal are differential. Differential signals don't corrupt the power supply or ground paths and are also relatively immune to noisy supplies or grounds. They also require no special interface circuits since the filter and limiter use differential signal processing stages. However, the extra pins used for differential signals are not available for ground pins. Separate ground pins are required for each section to keep the signals from mixing. The solution was to turn the package paddle into a ground plane. Each section has a

ground bond down to the paddle freeing up package pins. The limiter with its 80 dB of gain has to be well isolated from the paddle; this is no problem with the bonded wafer process chosen.

Two of the outputs, an LO output clock for the synthesizer and the transmit IF output, are not differential and are high level so they couple into the ground and supply wiring as well as adjacent package pins. Any LO output coupling into the IF output degrades the carrier suppression specification. It is impossible to filter externally because its frequency is so close to the IF. Three design choices minimize this problem. First, the LO output pin is sandwiched between the two supply pins which are AC coupled to the external ground plane. Second, internally the LO output clock has its own supply pin and wiring. Third, the user is given the choice to disable this function. Likewise, the IF output pin is sandwiched between the power enable pins isolating it from other more sensitive pins.

Lowpass Filter

The final die in the IF processor is the programmable lowpass filter network (LPF). In receive mode, the LPF removes high frequency products generated during the IF down-conversion, leaving only baseband energy. In the transmit mode, the LPF provides filtering for data pulse shaping prior to modulation. The simplified block diagram of the LPF is shown in Fig. 4. Since the transceiver operates in half duplex mode, the in-phase and quadrature filters are multiplexed between transmit and receive modes to save on silicon area. Referring to Fig. 4, this die consists of an input network which interfaces to the QModem during receive and

the baseband processor during transmit. Similarly, an output interface network connects the filters to either the QModem during transmit, or the baseband network during receive. The filter portion consists of two identical 5th order Butterworth filters slaved to a central tuning network. The filters are programmable from 2.2 MHz to 17.6 MHz cutoff and in addition may be tuned $\pm 20\%$ around the selected cutoff frequency by way of an external resistor. Due to the relatively high frequency of the LPF cutoff, a transconductance-capacitor ($g_m - C$) architecture was selected. This removes the need for active amplifiers with high gain-bandwidth requirements and minimizes power dissipation. The LPF is fabricated in a BiCMOS technology.

The $g_m - C$ methodology along with the requirement for low voltage operation, introduces some challenging design considerations. Headroom is the primary consideration while operating at the lower supply limit of 2.7 V. To maximize headroom requires active networks with few stacked devices. For a $g_m - C$ filter, this implies a simple transconductance stage. Hence, simplicity is a key consideration leading to the differential transconductance stages shown in Fig. 5. The current outputs drive a low impedance cascode stage (not shown) which includes common mode feedback for bias stability. Hence the transconductor is fully differential to exploit the extra 6 dB implicit in such architectures. This simple differential stage uses bipolar transistors with resistor degeneration as the voltage to current converter. The low V_{be} drop compared to a MOSFET device aids in headroom conservation, and in addition, the NPN devices have better matching properties than corresponding MOS devices, yielding transconductance stages with lower intrinsic offset. For large order filters, offset propagation could seriously

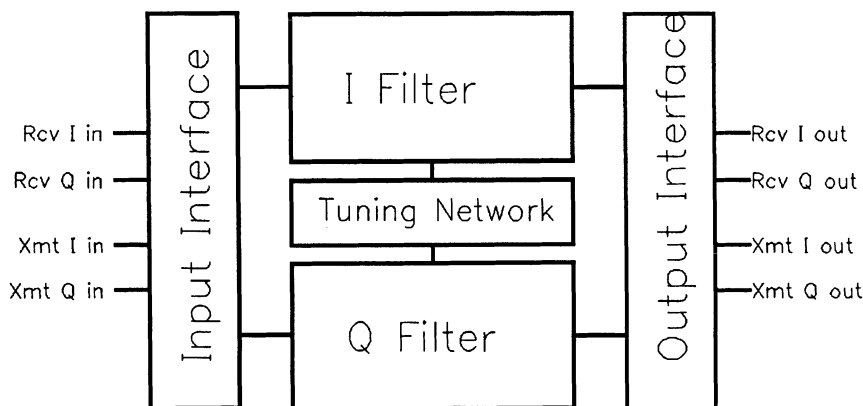


Figure 4. Baseband filter block diagram.

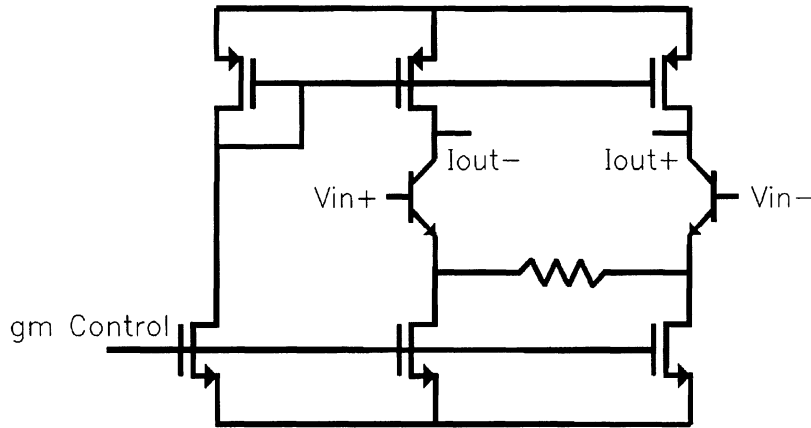


Figure 5. Transconductor schematic.

degrade the available dynamic range, hence reducing offset at the source is critical. As the network is differential, common mode offset must be considered as well. The bipolar devices improve differential offset, but common mode offset is only as good as the common mode feedback (CMFB) which controls it [15]. As a result, care must be taken in the design of the common mode circuitry. High gain in the CMFB stage assures that the DC level of the common mode operating point is within a few millivolts of the intended value. This is important to assure that bipolar devices in the g_m stages operate in an active region, and mosfets remain saturated. Low voltage operation also complicates the programmability of the LPF. The actual transconductance value the g_m stage assumes is determined by the tail current of the differential pair. To achieve three octaves of cutoff control and also achieve a relatively large linear region of operation would be very difficult. As a result, cutoff control required switching in different capacitor banks. This relatively simple solution makes 3 volt operation much more straightforward.

In addition to low voltage considerations, signal isolation between I and Q channels is of concern. Even at the relatively low frequencies of 2.2 MHz to 17.6 MHz, frequency dependent crosstalk can be substantial. This is due to electrical effects such as common circuitry between channels, and physical crosstalk due to parasitic capacitances coupling through adjacent lines or via the substrate. The relatively large die area of the LPF compared to its RF counterparts makes parasitic paths more likely and does degrade performance at the higher passband frequencies. Methods to combat these effects include total electrical isolation between channels. For example, independent bias networks for I and

Q are necessary to preclude coupling through that path. Experimental data taken at 1 MHz on networks with common bias, show up to a 20 dB degradation compared to networks with independent bias. Further, since the tuning network is common between both channels, maximum isolation is achieved by providing mirrored control currents to each channel and doing local I to V conversion, as opposed to generating one tuning voltage common to all filter sections. The current mirrors add a layer of isolation otherwise not obtained. Ideally, the current mirrors would be cascaded to provide yet another layer of isolation, but in low voltage networks, this is not practical. Physical isolation is maximized by separating channels as much as practical and using star supply and ground routing. Guard rings for both majority and minority carrier collection are helpful, and buried layer or well regions placed under all capacitors and active devices prevents injection or coupling into the substrate [16]. These techniques combine to give over 40 dB isolation.

Finally, a unique challenge to the LPF is the layout constraints. Because the die is one of four making up the IF processor, the pinout is restricted to only two sides since the die-attach is in the corner of the TQFP package (Fig. 1). This constraint makes a rectangular aspect ratio desirable. This leads to a partitioning of I and Q filters on the top and bottom of the die with the tuning network and output interface between. The input interface occupies the right side of the die as shown in Fig. 6. This partitioning permits a general signal flow from right to left without the need for crossing of input and output lines. The centralized tuning network improves matching between I and Q channels while also providing separation between channels for reduced

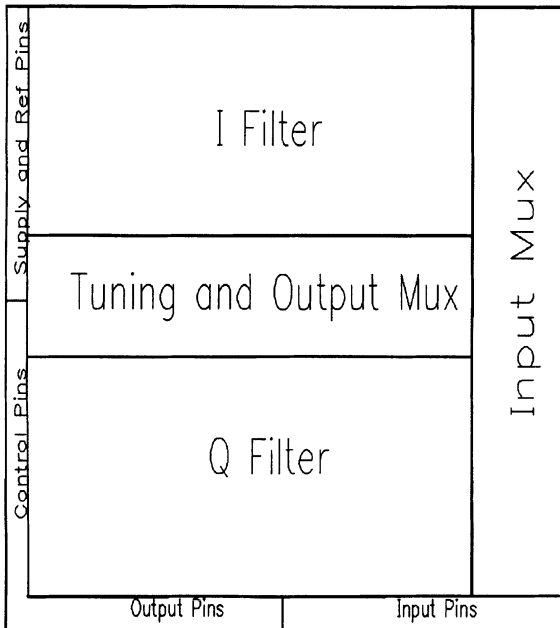


Figure 6. LPF layout floorplan.

crosstalk. To maximize matching between channels, transconductance stages are physically placed in proximity to one another. Further, all transconductance cells are placed in the same absolute orientation to prevent errors caused by shadowing effects during ion implantation. This greatly improved DC offset control and general channel matching.

6. Manufacturing Challenges

Designing and fabricating a set of integrated circuits to implement a 2.4 GHz spread spectrum radio is certainly in itself a difficult technical challenge. However, when the requirements are expanded to include high volume manufacturing with an acceptable profit margin, the challenges begin to multiply rapidly. The following are a few of the manufacturing questions that had to be addressed during the development and early manufacturing stages of this project.

- 1) Are the production areas technically capable of producing these devices and if not, what new capabilities need to be developed?
- 2) Are the production areas properly equipped and staffed to handle the projected sales volume?
- 3) What specific operations in the manufacturing flow will restrict production and what actions need to be taken to avoid these constraints?

- 4) What is the most efficient manufacturing flow both from a cost and a quality standpoint?
- 5) Are there reasonable design or specification changes that can be implemented to improve the manufacturing yields and reduce costs?

Many of these questions are common to the development of any integrated circuit. However, the intent of this discussion is to focus on specific manufacturing issues that arose due to the technical requirements or architectural design of this chipset. The typical steps in an IC manufacturing flow are wafer fabrication, wafer test, assembly, and package test. The 2.4 GHz WLAN chipset presented some unique challenges in each of these areas.

Wafer Fabrication

At the start of this project Harris's UHF1 process had an NPN Fmax of 7 GHz, while at least 13 GHz was needed to make this chipset feasible. The new Fmax goal was achieved primarily by improving the photolithographic process, by switching from a projection aligner to a stepper. The wafer throughput of the stepper was considerably less than the projection aligner thus only critical levels used the stepper.

Wafer Probe

For the IF QMOD, the main manufacturing concern was caused by the architectural decision to place four die in a multichip module. A simple DC wafer probe screen has an efficiency of around 90%. This means that 90% of the die that make it through the screen are actually good while 10% are actually bad. Assuming a simple yield model, the package test yield for a single chip device would be around 90%. However, if four of that same circuit were placed in a multichip module the odds of getting all of the die to function at the same time would be 0.9 to the fourth power or 65.6%. Therefore, for this example, the package test yield decreased by 25% solely due to the multichip module assembly approach.

Early in the development of this device it was decided that the maximum manufacturing effort would be placed on trying to screen out as many of the bad die as possible at wafer probe. This meant not only doing the typical DC testing at wafer probe, but also doing full AC testing despite the increased cost and complexity of the probe screen. This approach would

make it possible to increase the probe efficiency much closer to the goal of 100%. Therefore, full DC and AC probe capabilities were developed for all three of the IF die.

Probing of the RF/IF Converter die also posed a manufacturing challenge. Simple DC testing was hindered by high frequency oscillations induced by probe parasitics. Compensation networks were designed by modeling and simulation of the circuit and the probe fixture. Circuit simulations were then run on the device in this configuration and new DC test limits were generated.

Assembly

One of the more obvious problems posed by this chipset was the complexity of assembling four die in one package. The assembly diagram for the IF module is shown in Fig. 7. Bondwire lengths were kept as short as possible to minimize series inductance. This was accomplished by positioning each of the four die as close as possible to its respective corner of the common die attach paddle. The series ground inductance was minimized by connecting the die attach paddle to external ground through parallel bondwires and lead fingers.

In addition to adding complexity to the assembly process, the multichip module approach also presented some difficult logistic problems for production planners. A multichip module cannot be assembled until

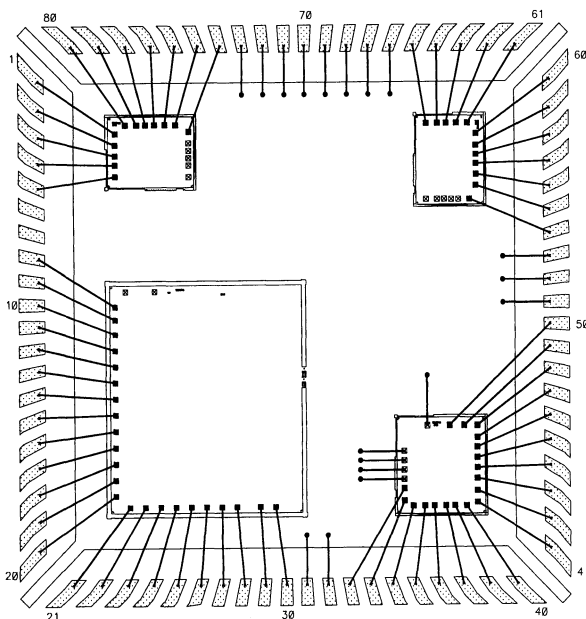


Figure 7. Assembly diagram for the IF QMOD.

all of the individual die needed to build it are available. As a result the planning activity is complicated due to the uncertainties inherent in the manufacturing process. The solution was to create an excess die bank for each of the individual die in the multichip module. However, this approach had to be used with some restraint because of the cost associated with creating excess inventory.

Assembly of the RF/IF Converter die was also challenging. The main issue was series ground inductance. As in the IF device, the ground plane for the RF/IF Converter was the die attach paddle. In addition to multiple downbonds, the leadframe itself was altered to leave several of the ground leads fused directly to the die attach paddle, thereby reducing series inductance.

A final assembly challenge posed by this chipset was the maximum package height requirement, for a Type II PC Card form factor. Thin packages make the assembly process more difficult. The loop height of the bondwires must be closely controlled, and the die thickness must be closely maintained. Thin wafers are more susceptible to breakage, and thin packages are more susceptible to moisture intrusion. As a result, the manufacturing flow was setup to avoid excessive exposure of packaged units to moisture.

Package Test

AC performance of the IF QMod is not fully guaranteed by AC probing of the individual die. Cascaded performance is impossible to guarantee by wafer probing because some parameters are affected by coupling between die or the package parasitics. Also, due to the statistical nature of this problem, the optimum limits at probe are less stringent than worst case. However, this guarantees that a small percentage of packaged units will fail and must be screened out. Therefore a data collection and analysis system was established to permit the optimization of total yield.

The package test plan for the RF/IF Converter specified full 2.4 GHz testing of various RF parameters such as gain, compression point, VSWR, and return loss. Implementation of these RF tests in a production environment required special RF measurement hardware installed on a commercially available production tester. However, the interfacing of this RF test equipment with the device under test was the main problem. The necessary test socket must have low inductance, be compatible with an automated handler, and rugged enough to survive many insertions. Further, the interconnections

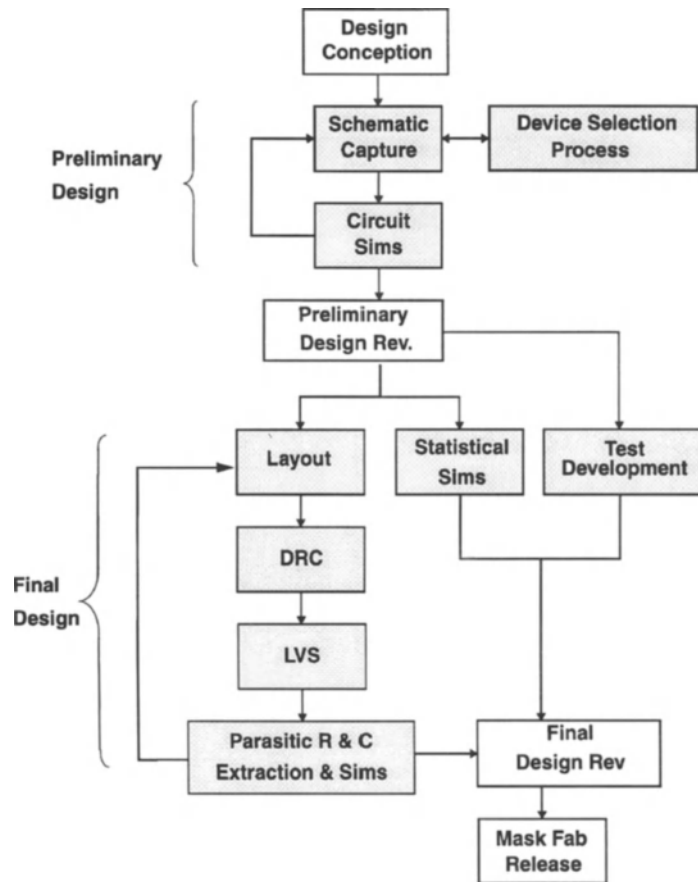


Figure 8. A simplified RFIC design flow.

from the tester to the socket must be de-embedded. These techniques made it possible to implement full 2.4 GHz RF testing using an automatic tester coupled to a production handler.

7. CAD Methodology

The IC implementation of the wireless LAN chipset required a new set of sophisticated CAD tools to cover all aspects of the development process. Historically, IC and RF designers have used different design goals, design methodologies, and practices [17]. Traditional analog designers have enjoyed an integrated front-to-back IC design system. On the other hand, RF designers, backed by a discrete design background, have typically used board level CAD design tools. As the boundary between IC and RF design blurs, both IC and RF designers are compelled to design in each other's domain. Designers in both areas are beginning to recognize the need for a CAD system that supports the

design tools of both domains. A typical RFIC design flow is shown in Fig. 8, where the shaded boxes represent steps where the designer must use traditional analog and RF design tools.

After an extensive evaluation of the CAD requirements for the RF portion of the wireless LAN chipset, the approach taken was to enhance and modify the user interface, simulation engine, and data analysis tools in the Fastrack IC design system [18] to provide RF specific design and data representation capabilities. With this methodology, the simulation data base, device models, and cell libraries are the same, regardless of the type of application (RF or IC). Embedding RF tools in an IC design system is an enabling factor for the transition from discrete based RF designs to RFIC. It enables high frequency IC designers to easily traverse to the RF domain and it provides a traditional RF-design-system-like environment for RF designers. Regardless of IC or RF design, the user interface, simulation engine, and device models are the same.

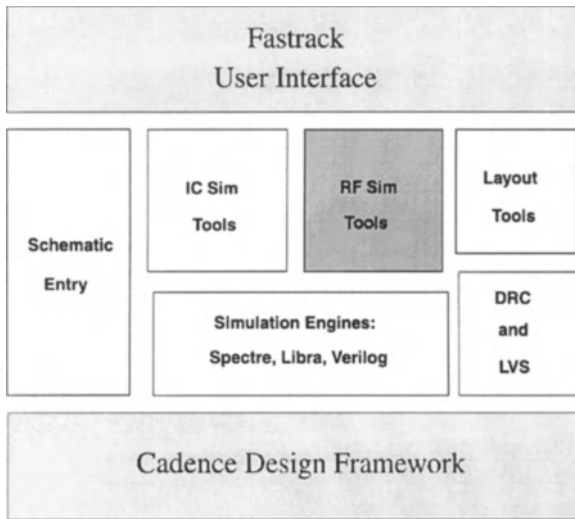


Figure 9. The structure of Fastrack, showing the embedded RF design tools.

Most RF specific design tools fundamentally use the same mathematical basis and numerical algorithms found in typical analog IC design systems. The procedure to extract the required data can be different and it may require a special set up and a controlled environment, but the basic tools are the same [19–21]. The structure of the CAD system used for the development of the Wireless LAN chipset is shown in Fig. 9.

The following sections elaborate on challenging methodologies and tools designed to replicate RF design capabilities in an IC design system.

Nonlinear Analysis

RF designs are frequently interested in the nonlinear performance of a circuit in the frequency domain. This will enable the designers to observe harmonic distortion for single tone circuits and the intermodulation products for multi-tone circuits. There are two basic approaches to nonlinear analysis in the frequency domain: 1) harmonic balance and 2) time domain based simulation with time to frequency conversion.

Harmonic balance based simulators are typically more efficient than the time based simulators for applications in which the circuit

- Contains a small number of nonlinear elements.
- Takes a long time to reach steady state.

- Operates in a multi-tone mode where the beat frequency is many orders of magnitude (5 or higher) smaller than the tone frequencies.
- Element count is dominated by linear elements (e.g., package models).
- Contains many parts/blocks represented by S parameters.

The standard approach to time based simulation for analysis of nonlinear effects in the frequency domain is to use Spice based transient analysis followed by an Fast Fourier Transform (FFT) of the results. This method generally produces 60–80 dB of dynamic range which is unacceptable for most applications. A new method based on internally controlled and automated nonlinear transient analysis and followed by a dedicated FFT, has shown to produce a dynamic range limited only by the floating point accuracy of the computing machine (up to 260 dB). The following elaborates on this methodology.

- Determine the frequencies of the input sources.
- Determine the Beat frequency, f_b . It can be shown that the beat frequency is the largest common divisor of the input frequencies.
- Determine the starting point in time for the FFT sampling, t_{init} . This is the point (user specified) at which we assume that the circuit has reached satisfactory steady state.
- Determine the end point for the FFT sampling, t_{end} . Where

$$t_{\text{end}} = t_{\text{init}} + \frac{1}{f_b}$$

- Determine the largest non-negligible frequency content of the desired signal, f_{max} . This is the frequency at which the spectrum of the desired signal effectively dies out. Even though exact knowledge of this frequency is not required, it plays an important role in eliminating aliasing.
- Determine the number of FFT sampling points, N_{fft} . Where

$$N_{\text{fft}} \geq \frac{2f_{\text{max}}}{f_b}, \quad N_{\text{fft}} = 2^n, \quad \text{and } n \equiv \text{integer}$$

- Determine the FFT sampling points, t_i . Where

$$t_i = t_{\text{init}} + \frac{i}{f_b N_{\text{fft}}} \quad i = 0, \dots, (N_{\text{fft}} - 1)$$

- Run the simulator and force it to step onto the FFT sampling points.
- Perform an FFT of the time domain results from t_{init} to t_{end} using N_{fft} sampling points.

While harmonic balance based simulators become extremely inefficient for anything but very small circuits, the above method is as efficient as typical SPICE transient analysis for larger circuits.

Backend Design. Verification of the completed IC layout is accomplished using Cadence’s Diva Design Rule Checking (DRC) tool to ensure that the layout conforms to all manufacturing specifications (metal width, metal spacing, etc.), and a Layout Versus Schematic (LVS) tool to validate the electrical functionality of the layout with respect to the schematic. Once the layout is verified, parasitic resistances and capacitances can be measured and back-annotated to the schematic netlist for simulation of the parasitic effects on circuit performance.

A key part of the physical design is based on the device layouts designed to fully implement the variable geometry structures that were modeled during simulation; they are not limited to discrete values. In effect the parameterized cells completely remove the numerous IC device level design rules from the layout process, without loss of freedom or functionality.

The device layouts are automatically synthesized from the schematic by placing the parameterized cells relative to their schematic positions and applying the corresponding model parameters to size the geometries. In this way, the transistor in Fig. 10 can be thought

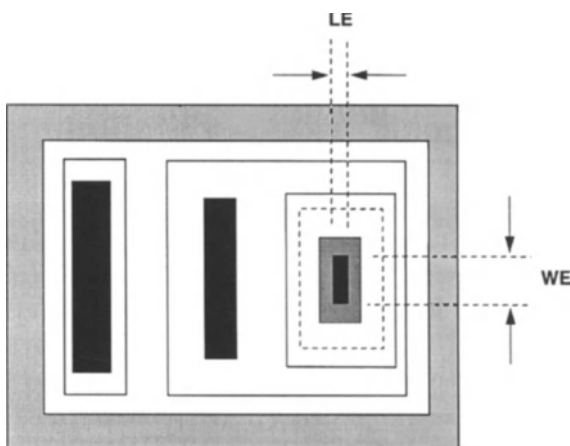


Figure 10. Simplified top view of Harris UHFN1 transistor showing emitter length (LE) and emitter width (WE) electrical parameters.

of as a virtual black box, as shown in Fig. 11, with terminals for connecting the collector (C), base (B), and emitter (E). The device can now be thought of as having stretch lines that bisect the layout structure to adjust all of the internal geometries as a function of the electrical parameters (e.g., emitter length, emitter width, etc.). A wide range of layout optimizations are also supported by simply changing device parameters; for example, parallel and serpentine resistor structures, trimmable thinfilm (as a function of trim range and trim sensitivity), and capacitor aspect ratio. This allows the user to concentrate on just the circuit level interconnections.

Design For Packaging

The configuration of bondwire networks and the locations of down bonds (if used) to the die attach on the other hand is unique to the particular design and layout. Tools which automate the task of generating models for these unique cases can be extremely useful in allowing quick exploration of the impact of various bonding configurations.

One such set of tools [22] combines a point and click graphical user interface for generating bond diagrams with an electromagnetic (EM) simulation engine to generate a fully coupled lumped element model for the bondwire network and die attach downbond configuration. The conventional two-dimensional bond diagram is constructed and then augmented with information about die thickness, cavity depth and bond angles to allow the generation of a three dimensional representation for the bondwire network as shown in Fig. 12. This three dimensional representation is used as the input for the EM simulator. Post processing is then performed on the EM simulator output (inductance and capacitance matrixes) to result in a lumped equivalent element SPICE [23] netlist. Finally, a symbol representing the netlist as a “black box” with input and output connections is generated for subsequent placement and hookup to rest of the circuit schematic.

8. Lessons Learned

One of the privileges of membership in the human race is the ability to learn from one’s own mistakes and successes and to know the difference. In most projects of this scope, the company gains intangible assets in its employee knowledge base. In a well managed organization, this knowledge is used to reduce costs and/or time for future products developments.

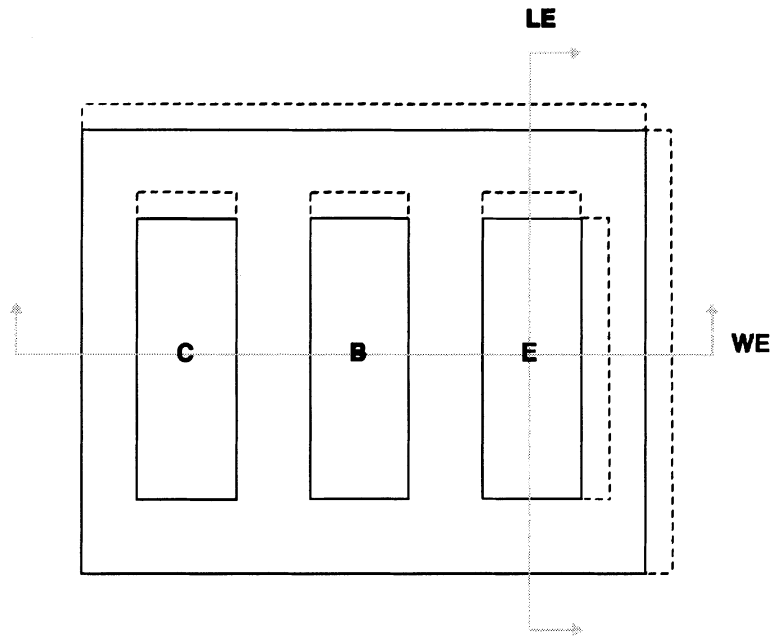


Figure 11. Effective parameterized cell of Harris UHFN1 transistor showing stretch lines that adjust internal geometries based on electrical parameters.

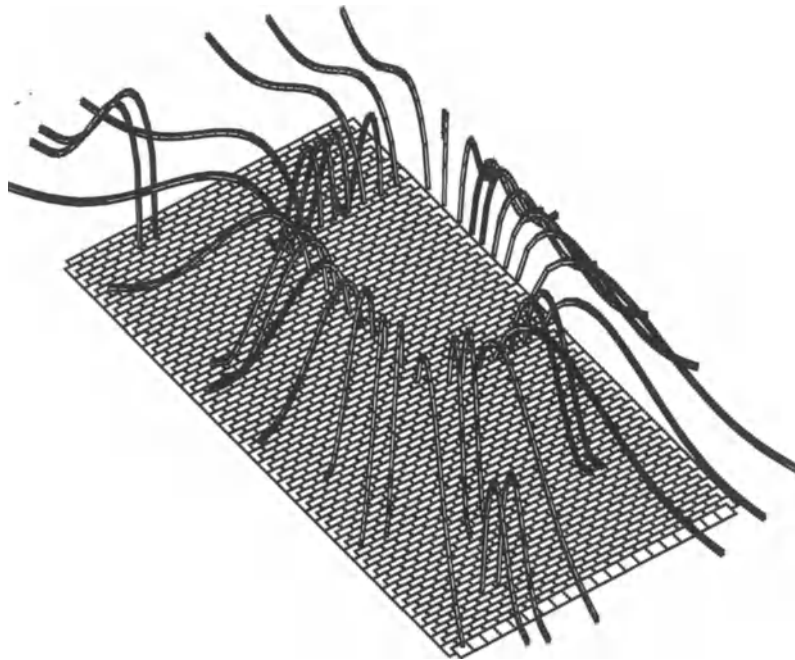


Figure 12. Three dimensional bond wire network.

Manufacturing issues need to be considered early in the concept and design phases of an IC development project to insure a smooth transition of the product into production. High volume RF IC testing in a manufacturing environment is difficult to implement.

Great care must be taken in the choice of test hardware and the design of the test fixtures. Unforeseen manufacturing problems will occur during production ramp up. Manufacturing and Development engineering need to work closely together to resolve these problems

quickly. This requires a commitment from both product development and manufacturing engineering organization to plan for the “unplanned” resource needs during the early project planning stage.

With a project that incorporates the development of several IC's, the entire manufacturing flow must be scrutinized to minimize costs. Sometimes it is beneficial to increase the costs of certain manufacturing steps or circuits to reduce the costs of the complete system. In addition, it is sometimes expeditious to schedule in delays or stagger development activities such that two circuits are not competing for the same limited resources.

Multichip modules are an effective way to provide functionality not possible with a single IC. However, they can become expensive if assembly and package test yields are not excellent. Very stringent testing of the individual circuits at wafer probe and coordination of individual test limits are necessary to insure high yield at package test. Statistical methods may be useful for optimizing final cost. The communications with the assembly site, more difficult when outside the national boarders, must be timely and clear.

Better CAD tools often result in simulating more parameters over a longer design cycle instead of doing the same simulations more quickly. This may be acceptable if increasing design cycle times are viewed as risk compensation for longer, fixed manufacturing cycle times. A timely design cycle requires a simulation strategy which recognizes that undetected design problems usually result not from poor models but from what was not even considered and thereby not simulated. It is an art determining when to go to silicon and stop the design. With too little simulation, major problems may be overlooked resulting in extra manufacturing cycles. Too much simulation delays finding the completely unexpected. The improvements in computer aided design tools should never lull a design team into relaxing its requirements for peer design reviews.

At the RF frequencies, even the distributed nature of the package paddle needs to be modeled if it is used as a ground plane. Such effects were observed in the RF/IF converter circuit as a reduction in gain and a shift in the impedance match vs. frequency (i.e., return loss vs. frequency). Therefore, the size of the circuit may increase tenfold when these extra parasitics are added. This creates a demand for faster simulation tools to get acceptable run times for circuits with complex package, bondwire and layout parasitics.

Finally, hiring qualified consultants is a good way to add missing experience to a design team or to gain a fresh perspective. This may be particularly important when the product is conceived as novel or prospective

customer feedback is limited by their expectations or present implementations.

References

1. W. Kilgore, A. Petrick, and D. Schultz, “Four-chip set supports high-speed DSSS PCMCIA applications,” *RF Design*, Oct. 1995.
2. S.R. Taub and S.A. Alterovitz, “Silicon technologies adjust to RF applications,” *Microwaves & RF*, pp. 60–74, Oct. 1994.
3. A.K. Agarwal et al., “MICROX-An advanced silicon technology for microwave circuits up to X-band,” *Proceedings of the 1991 IEEE Int'l Electron Devices Meeting*, pp. 687–690.
4. S.A. Campbell, “The possibility of semi-insulating silicon wafers,” *1995 IEEE MTT-S International Microwave Symposium—Silicon RF Technologies Workshop WFFA*, pp. 1–10.
5. R. Lowther et al., “Substrate parasitics and dual resistivity substrates,” to be published, in *IEEE Transactions on Microwave Theory & Techniques*.
6. R.M. Warner and J.N. Fordemwalt (eds.), *INTEGRATED CIRCUITS Design Principles and Fabrication*, McGraw-Hill, New York, p. 267.
7. N.M. Nguyen and R.G. Meyer, “Si IC-Compatible inductors and LC passive filters,” *IEEE Journal of Solid-State Circuits*, Vol. 25, No. 4, pp. 1028–1031, Aug. 1990.
8. N.M. Nguyen and R.G. Meyer, “A silicon bipolar monolithic RF bandpass amplifier,” *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 1, pp. 123–127, Jan. 1992.
9. N.M. Nguyen and R.G. Meyer, “A 1.8-GHz monolithic LC voltage-controlled oscillator,” *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 3, pp. 444–450, March 1992.
10. J.Y.-C. Chang, A.A. Abidi, and M. Gaiten, “Large suspended inductors on silicon and their use in a 2- μm CMOS RF amplifier,” *IEEE Electron Device Letters*, Vol. 14, No. 5, pp. 246–248, 1993.
11. K.B. Ashby et al., “High Q inductors for wireless applications in a complementary silicon bipolar process,” *1994 Bipolar/BiCMOS Circuits & Technology Meeting*, pp. 179–182.
12. F. Ndagijimana et al., Frequency Limitations on an Assembled SO8 Package, 0569-5503/93/0000-0530, 1993 IEEE, source publication unknown.
13. T. Luk and M. Hosseini, Characterization of Electrical Packages via Simulation and Measurement, 10th Biennial University/Government/Industry Microelectronics Symposium, 1993 IEEE.
14. S. Diamond and B. Janko, “Extraction of coupled SPICE models for packages and interconnects,” *Tektronix IPA 310 Applications Information*.
15. J.E. Duque-Carrillo and P. Van Peteghem, “A general description of common-mode feedback in fully differential amplifiers,” *Proc. IEEE Int. Symp. Circuits and Systems*, pp. 3209–3212, 1990.
16. K. Joardar, “A simple approach to modeling cross-talk in integrated circuits,” *IEEE Journal of Solid State Circuits*, Vol. 29, pp. 1212–1219, Oct. 1994.
17. M. Chian and D. Chian, “Merging RF and IC design tools for ASIC development,” *RF Design*, Oct. 1993.
18. “Harris fastrack design system,” Harris Semiconductor, Release 3.6, Sept. 1993.

19. R.A. Anderson, "S-parameter techniques for faster, more accurate network design," *Hewlett-Packard Journal*, Vol. 18, No. 6, Feb. 1967.
20. J. Ortiz and C. Denig, "Noise figure analysis using spice," *Microwave Journal*, April 1992.
21. G. Gonzalez, *Microwave Transistor Analysis and Design*, Prentice-Hall, Inc., New Jersey, 1984.
22. S. Majors, "Generating bond diagrams for packaged IC verification," *Harris Semiconductor Technical Journal*, Vol. 1, No. 1, pp. 19-23, 1995.
23. L.W. Nagel, SPICE2, A Computer Program to Simulate Semiconductor Circuits, Technical Report ERL-M520, UC Berkeley, May 1975.



Mojoy C. Chian received B.S.E.E., M.S.E.E., M.S. in applied math., and Ph.D.EE from Florida Institute of Technology (FIT) by 1988. From 1980 to 1988, he was a teaching and research assistant with the EE/CP department at FIT where he worked on circuit simulation, tabular macromodels, closed form solution of linear systems and numerical algorithms. He is presently an adjunct professor in the EE/CP department at FIT.

He joined Harris Semiconductor, CAE in 1988. His activities included RF CAD tools, mixed signal and mixed level simulation, switched capacitor analysis, and macro/behavioral modeling for circuit level simulation. He is currently Director of Design Systems.



Gregg D. Croft received the B.S.E.E. degree from Carnegie Mellon University in 1981 and the M.S.E.E. degree from the University of South Florida in 1993. In 1978, he began his career as a technician for the Solid State Research Laboratory at Carnegie Mellon University. From 1981 to 1987 he was employed by Advanced Micro Devices where he worked on a variety TTL, ECL and CMOS integrated circuits. In 1987 he joined Harris Semiconductor as a Staff Engineer for the Analog Signal Processing group. His primary responsibilities in this group included the development and manufacturing of

high speed analog integrated circuits for video and RF applications. He is presently a Sr. Principal Engineer in the Advanced Process Development group.



Steve Jost received the B.S. degree in electrical engineering from the University of California at Davis in 1984. He received an M.B.A. degree from Florida Tech in 1994.

He joined the analog product development group of Harris Semiconductor in 1984. Between 1984 and 1993 he designed numerous high performance linear standard products including analog multipliers, comparators, and several families of current feedback amplifiers. Since 1993 he has been developing RF ICs for wireless applications. Since 1992 he has been a group leader for analog RF product development.



Patrick J. Landy was born in New York City on September 12, 1963. He received the B.S.E.E. degree from the University of South Florida in 1985, and the M.S.E.E. degree from Florida Institute of Technology in 1989. He has been an IC circuit designer with Harris Semiconductor since 1986. His interests are in analog, mixed signal, and RF IC design.



Brent Myers received the B.S. degree in electrical engineering from Purdue University in May 1979, the M.S. degree from Virginia Tech in Dec. 1982, and the Ph.D. degree from the Florida Institute of Technology in Dec. 1994.

Since 1985 he has been with Harris Semiconductor where he is presently a Senior Scientist in the Mixed Signal Product Development group. He is also an adjunct professor at the Florida Institute of Technology in the area of analog and mixed signal circuit design. His interests include monolithic integrated filter techniques, PLL and frequency synthesis networks, and analog design techniques in general.



John Prentice received a B.S. degree in electrical engineering in 1971 from Cornell University and a M.S. degree in ocean engineering in 1975 from the University of Miami. Since 1975 he has been employed by Harris Semiconductor with reliability, advanced process development and circuit design engineering responsibilities. His current technical interests include designing analog IC circuits and components for rf wireless and power products. He holds 12 patents.



R. Douglas Schultz received the B.S. degree in electrical engineering from Penn State University in 1981. From 1981 to 1983 he was with Alpha Inc., Optimax Division, Colmar PA, where he worked with thick film RF amplifiers. From 1983 to 1986 he was with American Electronics Labs, Lansdale PA, where he worked with RF receiving systems. From 1986 to 1992 he was with TRW, Warner Robins Avionics Lab, Warner Robins GA, where he worked with radar and ECM systems.

From 1992 to 1994 he was with Harris Corp., Melbourne FL, where he worked with silicon monolithic for commercial wireless applications. In 1994, he founded Integrated RF Solutions Inc., Palm Bay FL, and specializes in commercial radio development and custom integrated circuit design.

Flat Panel Displays for Portable Systems

KALLURI R. SARMA

Honeywell Technology Center, 21111 N. 19th Ave., Phoenix, AZ 85036

TAYO AKINWANDE

Massachusetts Institute of Technology, Cambridge, Massachusetts

Received ; Revised

Abstract. Flat panel display technologies for portable and personal information systems are reviewed. The display sub-system performance requirements, and the metrics for evaluating display technologies for portable systems are discussed. The current display technology choices for high performance portable systems are active matrix liquid crystal display (AMLCD) and field emitter display (FED). AMLCD is at the forefront at an advanced state of development, and it is already in mass production for notebook computer applications. Because of the huge market size, AMLCD technology continues to be developed at an aggressive pace to address the needs of the future portable systems. On the other hand, FED technology is not currently in mass production, but it is being developed at rapid pace; Impressive technology capabilities and demonstration displays have already been shown. This review focuses on the current status and future development trends in both the these display technologies for application to portable systems. The current status of the reflective LCDs and their future development trends are also reviewed.

1. Introduction

The pervasive nature of computing and communication is bringing about a paradigm shift in the direction of merging these to provide mobile or nomadic information systems. The vision of mobile computing systems is to provide anytime, anywhere access to information [1]. The ultimate goal is to provide a system capable of moving information to and from people at all locations through an advanced computer/communication network including high-speed wireless links between sources of information and users of information [2]. A consequence of the mobile computing paradigm is the increased importance of information systems with reduced power consumption. Central to the success of any portable system is battery lifetime or time between charges. This new paradigm is manifesting itself as users travel to different locations with laptops, personal digital assistants (PDAs), cellular telephones, pagers, cordless telephones, etc. [3]. The advent of battery-powered portable information systems with

significant computational and communication capability is driving the development of low-power technologies. The growth of the market for battery-powered portables is the current dominant trend in information technology. This in turn has led to increased effort to develop portable systems that are lighter and smaller, have enhanced performance, increased functionality and longer battery life [1–3].

The term portable information system covers a broad range of computer/communication systems. It includes palm-top, notepad, notebook, laptop and sub-notebook computers and personal digital assistants (PDAs) that utilize direct view displays. Some portable systems may utilize head mounted displays (HMD) and body mounted displays, using small very high resolution image sources (displays). In this review, the main focus is on larger area, direct view displays. The most common of the portable computer systems is the letter size, A4 format notebook computer. Today's typical notebook computer includes a microprocessor, a hard drive or PCMCIA, a floppy disk drive, memory, power supply,

graphics controller, and a display. Essentially all of the current notebook computer displays are based on passive-matrix or active-matrix liquid crystal display (LCD) technologies. The typical display is 10.4" dia. in size, with a VGA (640 × 480) resolution, and operates in a monochrome or full color mode with a brightness of 20 fL. Because the display is the most important human-machine interface, it is a critical sub-system of the portable system. It is the primary visual source for transmitting text, graphics, images and video data to humans. Commensurate to its importance, it presently represents about 30% of the cost of most portable systems and about 50% of the power dissipation. The availability of affordable flat panel displays stimulated the development of portable computer and communication systems.

One of the anticipated devices or appliances is the wireless multi-media terminal. The proliferation of applications such as the World Wide Web (WWW) and the Internet Multi-media Backbone (the MBone) has resulted in vast amounts of image and video data on the Internet. This is because of the recent advances in several technological areas that are leading to large scale databases of visual and multi-media information. Such databases are finding ready application in a wide range of fields such as advertising and marketing, education and training, entertainment, medicine and remote sensing. All these require increasingly sophisticated viewing and authoring tools that are mobile.

The InfoPad developed at the University of California by Chandrakasan et al., [2] is an example of the future multi-media terminal that will be required for the mobile or nomadic information systems described above. The InfoPad is a portable multi-media terminal that is intended for untethered access to fixed multi-media information servers on the Web. It is designed to transmit audio and pen input data from user to the network which contains the database and the necessary computation power through a wireless uplink. The InfoPad also receives audio, graphics and compressed video from the network on a downlink. The rich data type and the requirement for full motion color video will increase the performance requirements of the displays. It is expected that SXGA (1280 × 1024) resolution will be required for future multi-media terminals. Furthermore, the brightness requirement is expected to increase in order to accommodate a variety of ambient lighting conditions.

The key consideration for future information systems remains the battery life, portability and the display

technology. The display is particularly important because it is the most important human/machine interface, and it continues to be a major power consumer in the portable system.

2. Display Requirements for Portable Systems

The display sub-system requirements of portable information systems are largely driven by (i) application, (ii) the data types and (iii) the environment. This is illustrated using the following examples:

A personal digital assistant (PDA) does not necessarily require a high information content, high brightness or a full motion video color display. However, such a portable system requires a low power display. Depending on the intended use of the PDA, it may require only a 1/2 VGA display that relies on ambient light that a reflective monochrome display can provide. If the PDA on the other hand is intended for accessing still photo images or full motion video sporting action clips (such as a football game), a higher performance display sub-system will be required. Another example is the use of body-worn personal electronic systems for aircraft or tank maintenance. Such a system will require a full color display with high information content and dimmable brightness that adjusts to the background illuminance of the environment. For this application, full motion video capabilities will not be necessary; however, high resolution and high pixel matrix are essential. These requirements should be contrasted to those of personal information systems that are designed to view battle field situations at a remote location either sent by a laser range finder or an unmanned aerial vehicle (UAV). This system should be capable of showing live video of battle scenes to a soldier, remote intelligence officer or a battle field commander. Such a portable system for battlefield use will require high information content, full color displays capable of showing images and full motion video and operating in almost all ambient lighting conditions. Another example is the display for distance learning information systems in which remote students can participate in classroom discussions. In such a system, high information content and full color are essential, and the display system should be bright, consume very low power and be capable of full motion video.

Central to all the examples given above is the issue of battery life. Power consumption is a major consideration for all portable system displays. It is expected that the multi-media terminal will have the most demanding

display requirements because of the rich data sets that includes, text, graphics and full motion color video. The generic requirements of a display-subsystem for future portable information systems include:

- Full Color (8 bits/color)
- High Resolution (e.g., 160 dots/inch)
- High Pixel Matrix (1280 × 1024)
- Adequate Brightness (15–100 fL)
- Full Motion Video (80 fps)
- Wide Viewing Angle ($> \pm 45^\circ$ in horizontal and vertical directions)
- High Contrast Ratio ($> 100:1$)
- Light Weight
- Small Volume (small depth)
- Low power consumption/high luminous efficiency
- Low cost

There is no electronic display that meets the above requirements at the present time. Most portable notebook computers currently use the LCD because it is the closest to meeting the requirements listed above. However, much work needs to be done in order to bridge the gap between the requirements for today's notebook computers and tomorrow's multi-media terminals. A major consideration in the design of the display subsystem of today's notebook computer is the power consumed by the display subsystem. Table 1 compares the power requirements of today's (1995) notebook computer using a 10.4" dia. VGA display, with those of tomorrow's (1999) multi-media terminal, using a 13.3" dia. SXGA display. From the table we observe that the display sub-system consumes about 50% of the total energy today. Hence it is very important to develop technologies that reduce the power consumed by the display sub-system as means for reducing the power consumed by the multi-media terminal to extend its

Table 1. Power budget for a 1995 notebook computer and 1999 multi-media terminal.

Sub-system	Note book computers 1995	Multi-media terminal 1999
Display	4 W (10.4" dia. VGA)	2.5 W (13.3" dia. SXGA)
Logic/memory	2 W	1.0 W
Communications	0.5 W	0.25 W
Storage	1 W	0.5 W
DC Power Supply	0.5 W	0.25 W
Total	8 W	4.5 W

battery lifetime. As the table shows the current development efforts are targeted at reducing the display system power consumption substantially, while at the same time improving its size, resolution, brightness and image quality.

3. Review of Flat Panel Display Technologies

While the dominant display technology continues to be the CRT, its many deficiencies make it unsuitable for portable applications. To overcome the limitations of the CRT, various flat panel display technologies have been developed over the years for application to portable systems, as well as other systems where the deficiencies of the CRT are an issue. These technologies include:

- Liquid Crystal Display (LCD)
- Electroluminescent Display (ELD)
- Vacuum Fluorescent Display (VFD)
- Plasma Display Panels (PDP)

Development of these technologies spans over many years. They are relatively mature and currently in production for specific applications. Further, these technologies continue to be improved due to continuing development efforts. The LCD has been the most successful display technology for portable systems because it met the critical requirement for portability, low power, and ability to generate full color. Currently, LCD represents over 90% of the flat panel display market. Furthermore, it has also been able to give the highest luminous efficiency of all the display technologies that meet the portability requirements, leading to the longest battery lifetime for the portable system.

In the present notebook computers, the largest single drain on battery continues to be the flat-panel display sub-system. Most of the flat panel display systems described above are rather inefficient at converting electrical power to optical (visible) radiation. Typical conversion factors are of the order of 1 lumen/watt or less. The inefficiency is common to both emissive displays (ELD, VFD, PD) and the backlit light valve displays (transmissive LCD). The observed conversion efficiency should be contrasted with what is possible in some physical processes. For example, photoluminescence which occurs in a fluorescent lamp has a luminous efficiency (conversion efficiency) of 50 lumen/watt, while cathodoluminescence which occurs in a CRT has an intrinsic luminous efficiency of 25

lumen/watt. More striking is the fact that green light at 550 nm with a 100% electrical-to-optical conversion will have a luminous efficiency of 680 lumen/watt [4].

Below we describe the operating principles of each of the flat panel display technologies and give a short summary of their status and limitations. In the next section we propose metrics for evaluating the display technologies which explains why the LCD is the dominant display technology today.

3.1. Liquid Crystal Displays

Liquid crystals are typically organic chemicals that exhibit a meso-phase with anisotropic physical properties. These anisotropic properties include viscosity, dielectric permittivity, magnetic susceptibility and refractive index. Two important properties critical to the electro-optic effect exhibited by the liquid crystals are (i) optical anisotropy or birefringence meaning that the index of refraction is different when measured parallel and perpendicular to the optical axis, and (ii) dielectric anisotropy, which allows the LC molecule to be aligned with the electric field.

The liquid crystal display (LCD) is essentially a spatially addressable light valve that utilizes polarized light. The light valve is a “sandwich” composed of two sheets of glass with patterned transparent conductors (ITO) and a liquid crystal between them. Polarized light from an external source is sent through the light valve. The transmitted light intensity depends on the voltage applied across the liquid crystal. The LCD can be constructed to function in a transmissive mode using an external backlight, or in a reflective mode using a reflector behind the back polarizer [5]. In most instances, the reflective mode works with ambient light. Further, LCDs can be categorized in to two major types—passive matrix (PMLCD) and active matrix (AMLCD). Currently, AMLCD and super twisted nematic type passive matrix LCD (STN LCD) are most developed and widely used. We will discuss the LCD technology in greater detail in Section 5.

3.2. Electroluminescent Displays

Unlike the LCD which is a light valve, the electroluminescent display (ELD) is an emissive display. Figure 1 shows the basic structure of an ELD. It creates light by the excitation of a powder phosphor or a thin-film phosphor layer. There are four types of ELDs, based on the phosphor type and the excitation method used.

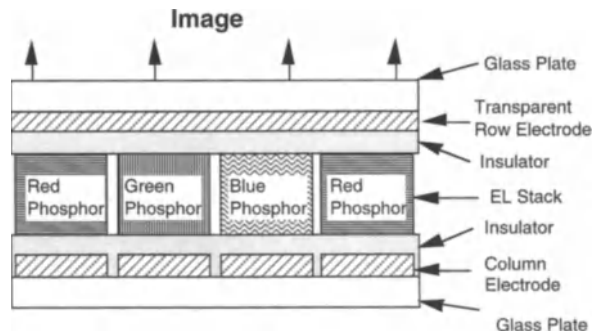


Figure 1. Device structure of the electroluminescent display (ELD).

These are: (i) dc powder ELD, (ii) ac powder ELD, (iii) dc thin-film ELD, and (iv) ac thin-film ELDs. For color displays, the red, green, and blue phosphor layers are patterned into pixel format and are encapsulated by two electrodes on either side. Parallel conductors form electrodes across the phosphor. They are fabricated in a row and column format on opposite sides of the phosphor as in the LCD [4, 6].

The ac thin-film ELD is the most viable of the four technologies. It consists of three layer sandwich structure—two insulating layers on either side of a thin-film phosphor. The row and column electrodes with the top layer being transparent are on either side of the sandwich. The basic EL device structure is purely capacitive as no current passes through the phosphor from the exciting circuit. The central thin-film phosphor layer emits light when a large enough electric field is applied across it. The required field is of the order of 1.5×10^6 V/cm. The high electric field means that the phosphor can be easily destroyed by any local imperfection in the thin film. Dielectric thin-films are thus added on either side of the phosphor to limit the current that can flow through the phosphor and prevent destructive short circuit of the phosphor. The insulating layers store charge and only allow displacement current through them to the phosphor.

When a voltage is applied across the basic EL structure, a high electric field appears across the phosphor. When the internal phosphor voltage reaches a threshold, a current flows in the phosphor layer and excites the EL center causing light emission. Typical phosphor used for ELD consist of a host material doped with an activator which is the light emission center. To be an efficient EL phosphor, the light emitting center must have a large cross-section for impact excitation mechanism but must be stable in high electric fields present in the phosphor [6].

Recently active matrix electroluminescence displays (AMELDs) for head mounted display applications have been reported [7, 8]. A 2560×2048 AMELD image source with $12 \mu\text{m}$ pixel pitch has been demonstrated. The display size is 1-inch diagonal. It uses PMOS and LDMOS transistors for local memory and driving the EL cells. Lateral DMOS transistors have the high voltage drive capabilities required for exciting the EL cells. The transistors are fabricated on single crystal silicon layers on insulators. The EL layers are deposited conformally on the transistor layers by atomic layer epitaxy.

The light production process of the ELD is rather inefficient. ELDs suffer from three major disadvantages—(i) low luminous efficiency (<1 lumen/watt), (ii) high voltage drive leading to high cost electronics, and (iii) lack of good blue phosphor making full color ELDs difficult.

3.3. Plasma Display Panels

Plasma displays operate by glow discharge of a noble gas typically Ne, mixed with a small amount of Xe or Ar. The plasma display panel consists of two glass substrates separated by dielectric spacers to form a Ne containing chamber. Figure 2 shows the typical structure of a plasma display panel. Each of the substrates has a set of transparent parallel conductors. The conductors on the two substrates are orthogonal to each other, thus forming a row and column addressing matrix. The gas is confined between the transparent row and column electrodes (on the top and bottom) and glass seal spacers on the sides. This is to isolate the plasma discharge and provide electrical isolation. When a high voltage is applied between the electrodes, the intersection produces a plasma from which light is emitted, and the

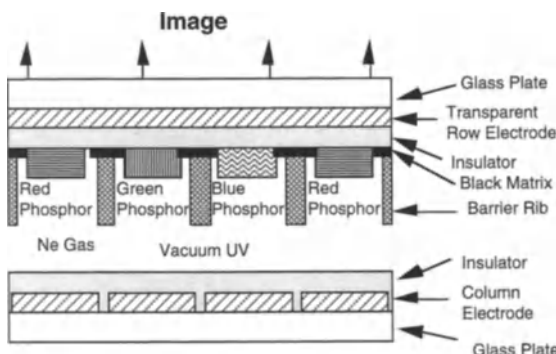


Figure 2. Device structure and operating principles of plasma display panels.

brightness of the display is related to the current of the plasma [9]. For a more detailed information on the recent developments in the plasma display technology, the reader is referred to [10–12].

In general, plasma displays are grouped according to how the current through the display is limited to prevent the destruction of the display. AC plasma displays use capacitors formed by placing a dielectric material between the plasma gas and the transparent electrode to limit the current. DC plasma displays use an external resistor to limit the current. Plasma displays have a number of desirable properties. These are (i) sharp turn-on voltages for matrix-addressing, (ii) high brightness and (iii) luminous efficiency (for monochrome displays).

The principle of operation of the full color plasma display is the same as in a fluorescent tube. A voltage applied between two electrodes creates an electric field which ionizes a gas at low pressure at a certain applied voltage. This leads to the emission of vacuum ultra violet light. The UV light is used to excite phosphors to produce visible light as in a fluorescent tube. While the plasma display has a simple basic structure and abrupt IV characteristics, it has two major disadvantages. The first disadvantage is the significant volume of gas that must be present to create sufficient brightness. This essentially places a limitation on how small the pixel can be which has implications for (i) resolution and (ii) portability—two important parameters for portable systems. The second disadvantage is the omni-directional nature of the light generated which leads to cross-talk between pixels and hence places a limitation on resolution.

3.4. Vacuum Fluorescent Displays (VFD)

The green emitting VFD is the most commonly used display device in cash registers, entertainment devices and automobile dashboards [13]. The VFD uses the same light production principle as the CRT, namely, Cathodoluminescence. Figure 3 shows a typical structure of a VFD. It uses a broad-area thermionic source of electrons typically made out of heated wires to create a virtual large area cathode. Electrons from the large area cathode are then density modulated by a group of x - y addressable grids. The electrons are accelerated towards a phosphor screen where light is produced by cathodoluminescence. Typically they operate at an anode voltage of less than 50 V. There are three basic problems with the VFD. The first problem is that there

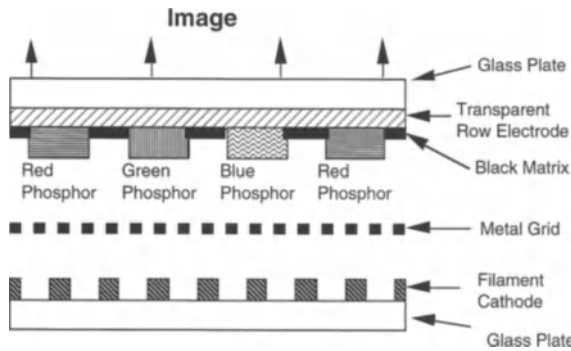


Figure 3. Device structure and operating principles of the vacuum fluorescent display (VFD).

are few known acceptable low-voltage phosphors with very high luminous efficiency to generate full color displays. The second problem is that electrons are emitted all the time from the broad area cathode which results in unnecessary power dissipation. The third issue is that matrix addressing of VFDs often result in reduced brightness due to a decrease in the duty cycle. To overcome the third problem, recently efforts have been directed towards increasing the brightness by using active matrix addressing techniques [14–16]. Troxell et al., recently reported an active matrix vacuum fluorescent display (AMVFD) technology. Thin film transistors were used to drive vacuum fluorescent displays. In this approach, a local memory is used at each pixel to achieve a 100% duty cycle, similar to an AMLCD [14, 15]. A 64×40 pixel, $450 \mu\text{m}$ pitch AMVFD was demonstrated with a spot luminance of 2500 fL at an anode voltage of 40 V.

4. Performance Metrics for Flat Panel Displays

Pankove [4] defined the ideal display many years ago thus: “the ideal device would modulate ambient light when it is abundant, but would emit bright light in the dark; it would be capable of producing saturated colors at will, be visible from all angles, have high resolution, respond in microseconds but retain image indefinitely if so desired, have a contrast ratio of over 50:1, 64 levels of gray, and consume negligible power at a low voltage”. However as he noted, there are many display devices that have subsets of these qualities, but there is none that has all these virtues. We shall go about defining the display performance metrics with Pankove’s vision in mind. To characterize and compare different display technologies, relevant performance measures and their metrics need to be defined. The relevant per-

formance measures include: luminance, contrast ratio, grayscale, color, viewing angle, power, luminous efficiency, frame rate, resolution, screen size, depth, pixel matrix, lifetime and temperature, humidity and vibration/shock resistance. In the following we will define these performance measures, discuss their significance, and describe their metrics for evaluating the available display technologies for portable system applications.

Luminance is a measure of the brightness of the display or the luminous energy emanating from the display surface. It is the luminous flux emitted per solid angle per unit area from the surface of the display. Its units are nit (candela/m²) or foot-Lamberts (1 fL = 3.426 nit).

Contrast Ratio characterizes the dynamic range of the display luminance. It is defined as the ratio between the maximum brightness to the minimum brightness of the display. High contrast ratio is essential for high quality video images.

Gray Scale is defined as the number of distinguishable steps in the display luminance. For adequate number of distinguishable graylevels, a luminance dynamic range covering at least ten $\sqrt{2}$ changes in brightness is believed to be necessary. The grayscale is measured in number of bits per primary color. Present AMLCDs typically employ 6 bit column drivers (i.e., 6 bits per color). However, future developments are aimed at 8 bit grayscale (per color), to achieve true color rendition.

Color quality is characterized by the color gamut achieved. For best color rendition, the color gamut should be as broad as possible. Color performance is characterized by the CIE chromaticity coordinates of the white, black, and primary and secondary colors of the display, and their stability as a function of graylevel and viewing angle.

Viewing Angle is the ability to view the display from a direction that is at angle to the display normal without degrading the image contrast and quality. The metric for viewing angle is the size of the viewing cone in angles in the horizontal and vertical directions.

Power is the electrical power in Watts consumed by the image source and the driver circuitry needed to drive the image source.

Luminous Efficiency is the optical power or luminous flux generated per unit electrical power required to generate it. It measures the efficiency of the display in converting electrical power to luminous flux. It is measured in lumens/watt. As a benchmark, 1 W of

radiant power at 550 nm is equivalent to 680 lumens of luminous flux.

Frame Rate measures how often an image is refreshed. Typically, it is anywhere between 30 Hz and 80 Hz. In some displays, flicker or smearing of fast action images may be observed if the frame rate is not high enough.

Resolution of packing density of information depends on how small each picture element (pixel) can be made. It is a measure of fineness and visibility of detail. The metric for resolution is lines per inch (lpi) or dots per inch (dpi).

Screen Size is a measure of the display size, and is usually given as the size of the screen diagonal. For example, most notebook computers have 25 cm to 30 cm diagonal displays.

Display Depth is the physical depth of the display in the direction orthogonal to the screen. It is a measure of display bulkiness or portability.

Pixel Matrix is the number and format of pixels in the display. The standard VGA display has 640×480 pixel array, while the SVGA has 800×600 pixel array, XGA has 1024×768 pixel array and the SXGA has 1280×1024 pixel array. We expect that future multi-media applications will use the SXGA resolution displays.

Lifetime is an important parameter and depends on the physics of the operative display device processes. Typically, an operational life of over 2×10^4 hours is usually targeted in the display design.

Temperature, Humidity, Vibration/Shock Resistance are measures of the stability or ruggedness of the display to environmental and operational conditions.

Table 2, adapted from reference [17], shows a comparison of the performance attributes of various flat panel display technologies in relation to a CRT. The

Table 2. Comparison of flat panel display technologies (Table adapted from reference [17]).

Performance parameters	CRT	Plasma DC/AC	Electro-luminescent DC/AC	VFD	STN/LCD	AM/LCD
<i>Display visual parameters</i>						
Pixel density	High	High	High	Medium	Medium	High
Screen resolution	High	Med/High	Medium	Medium	Medium	High
Raster distortion	Yes	No	No	No	No	No
Flicker propensity	Yes	Yes/No	Yes	Yes	Yes	Yes
Luminance	High	Medium	Medium	Medium	Medium	High
Dimming range	High	Medium	Medium	Medium	Medium	High
Contrast	Medium	Low/Med	Med/High	Medium	Medium	High
Gray shades						
(Intrinsic)	High	Medium	Medium	Medium	Low	High
Viewing angle	High	High	High	High	Low	Medium
Ambient contrast	Low	Medium	Low/High	Low	Medium	High
Color capability	High	Med/High	Med/High	Medium	Medium	High
Screen update time	Fast	Fast	Fast	Fast	Slow	Fast
<i>Display system parameters</i>						
Power	High	Medium	Medium	High	Low	Low
Luminous efficiency (Lum/Watt)	0.5	1	1	0.5	2	2
Temperature range	Wide	Med/Wide	Wide	Wide	Narrow	Narrow
MTBF	Medium	High	High	High	High	High
RFI emanations	High	Medium	Medium	Medium	Low	Low
Vibration endurance	Low	Medium	Medium	Medium	High	High
Volume	High	Low	Low	Low	Low	Low
Weight	High	Medium	Low	Medium	Low	Low

attributes are categorized into visual and system performance parameters. From the visual performance perspective, AMLCD can be seen to have essentially all the best possible attributes, with the exception of unlimited viewing angle. More importantly, from a system performance perspective, again AMLCD can be seen to have essentially all the best attributes, with the exception of narrow temperature range. It should be noted that the narrow temperature range of the current AMLCD is not a major limitation for its use in portable systems, which in general are not expected to require a broad temperature operation such as automotive and avionic displays. Similarly, the viewing angle limitation of the current AMLCD is outweighed by its superior overall system performance attributes for portable applications. With an adequate display visual performance, the figure of merit (FOM) of a display for portable applications is essentially based on power (luminous efficiency, LE), and size characterized by weight, W , and volume (depth, D). We can define the FOM as:

$$\text{FOM} = \text{LE} / W * D$$

Higher luminous efficiency, and smaller display weight and display depth, will result in a higher figure of merit. Based on this metric, currently AMLCD can be seen to have the highest FOM of all the flat panel technologies in Table 2, and thus the best choice for portable applications requiring high performance displays.

5. Liquid Crystal Display (LCD) Technology

LCD is one of the enabling technologies for the acceptance and rapid growth of the present portable systems such as the note book computers. The acceptance of LCDs for these systems is primarily based on their low power, and full color capabilities. While LCD technology development has a long history, its rapid pace and wide market acceptance based on price and performance criteria, started happening only since the past few years. Because LCD is a nonemissive display, in contrast to the competing technologies which are emissive displays, its intrinsic power consumption can be very low. While there are many types of liquid crystal materials such as smectics, nematics, and cholesterics, and display modes using these materials, twisted nematic (TN) display mode is the most advanced and popular. For a general overview of liquid crystal technology, the reader is referred references [18, 19]. Figure 4

shows the basic principle of operation of a twisted nematic display in a normally white mode. The incoming light from a back-light source, which is typically a fluorescent lamp with a diffuser, is linearly polarized using a sheet polarizer. In the voltage-off state, the 90° twisted nematic liquid crystal pixel rotates the polarization direction of the incoming linearly polarized light by 90°, which then goes through the exit polarizer which is set in a crossed polarizer configuration, thus creating the bright (white) state of the display pixel. In the voltage-on state, the liquid crystal molecules tilt so that their director orientation is parallel to the field, due to the positive dielectric anisotropy of the LC material. This allows the incoming linearly polarized light to go through the liquid crystal with its polarization state unaltered. When this linearly polarized light encounters the crossed exit polarizer, it is blocked, thus creating a dark (black) state for the display pixel. A full color display can be created by incorporating red, green, and blue color filters at the pixels. Figure 5 shows the transmission (luminance) versus applied voltage characteristic for a typical TN LCD. Grayscale can be achieved by varying the voltage applied across the LCD using the pixel transmission versus voltage characteristic shown in Fig. 5. Note that the transmission-voltage curve shown in Fig. 5 is for a normal viewing angle. Unfortunately, the transmission-voltage curve is viewing angle dependent as shown in Fig. 6, which leads to grayscale errors and color shifts in a display when it is viewed from significant angles to the display normal. We will discuss the current solutions to this problem later in this section.

This display shown in Fig. 4 can be made to operate in a normally black mode simply by changing the polarizers to a parallel orientation. There are performance and cost trade-offs for selection between a normally black mode and normally white mode of operation. The majority of the present applications use a normally white mode; they are easier to manufacture, but provide a limited viewing angle. Normally black displays require tighter manufacturing tolerances, but provide an enhanced viewing angle particularly when used in combination with multi-gap [20] and halftone grayscale [21]. Presently only specialized applications such as avionics requiring a wide viewing angle, superior color and grayscale performance use normally black mode [22]. In the rest of the following discussion, we will focus on the normally white AMLCD that is typically employed in the present portable systems. It remains the configuration of choice for the

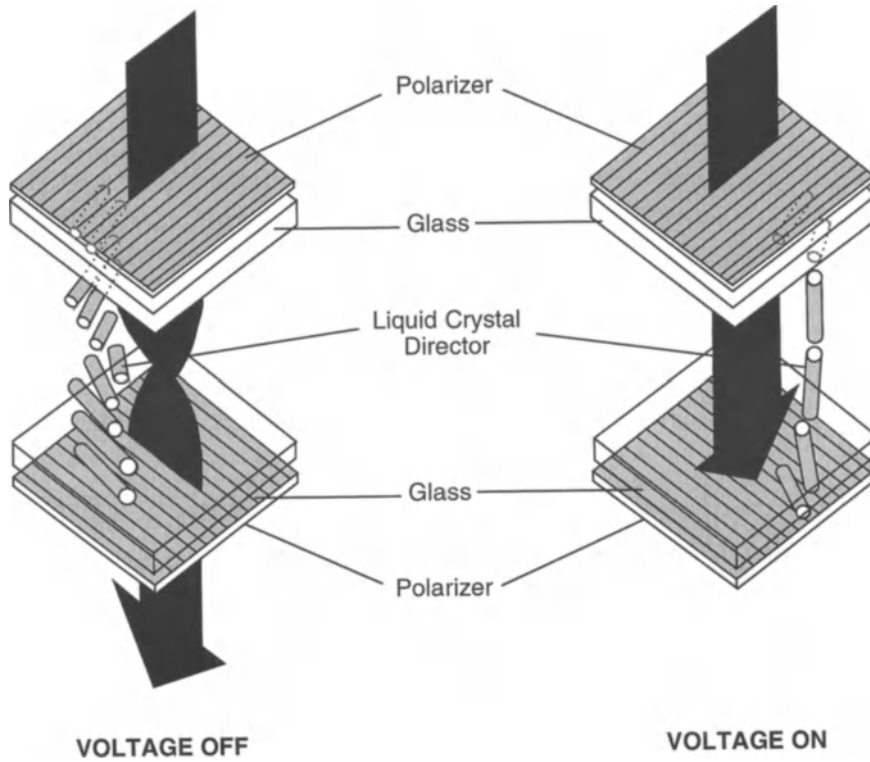


Figure 4. Principle of operation of a twisted nematic LCD in a normally white mode.

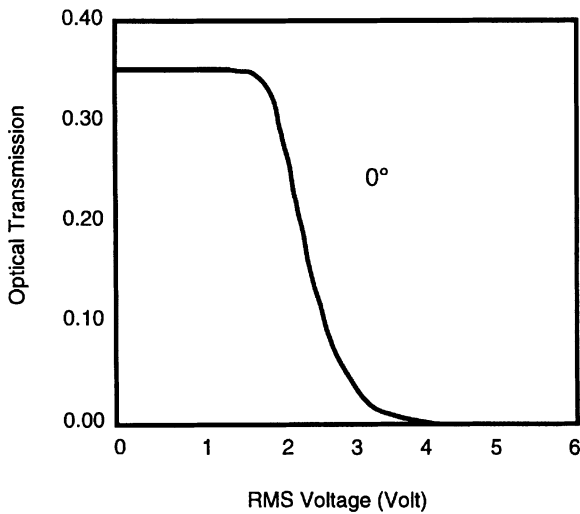


Figure 5. Transmission versus voltage for a typical TN LCD, for normal viewing angle ($0^\circ = 0^\circ$).

portable systems based on the projected cost and performance.

There are two broad categories of LCDs, namely passive matrix LCDs (PM LCD) and active matrix LCDs

(AMLCD). A passive matrix display comprising a liquid crystal between a matrix of transparent conducting row and column electrodes, is the simplest and least expensive liquid crystal display to manufacture. In a PM LCD, the row voltages are scanned in succession with a voltage, V_r , while all the columns in a given row are driven in parallel, during the row time, with a voltage of $\mp V_c$ depending on whether the pixel is selected to be ON or OFF. This approach is acceptable for low resolution displays. As the resolution increases, the difference between the select and non-select voltages (V_{on} , and V_{off}) decreases due to cross-talk between select and non-select pixels. The cross-talk is a result of the sneak paths in the row/column electrode matrix, which allows the non-select pixels to receive part of the applied voltage, thus degrading the contrast ratio of the display. Alt and Pleshko [23] analyzed this situation and derived an expression for the voltage ratio between the select and non-select pixels as a function of number of rows in the display:

$$\frac{V_{on}}{V_{off}} = \sqrt{\frac{(\sqrt{N} + 1)}{(\sqrt{N} - 1)}}$$

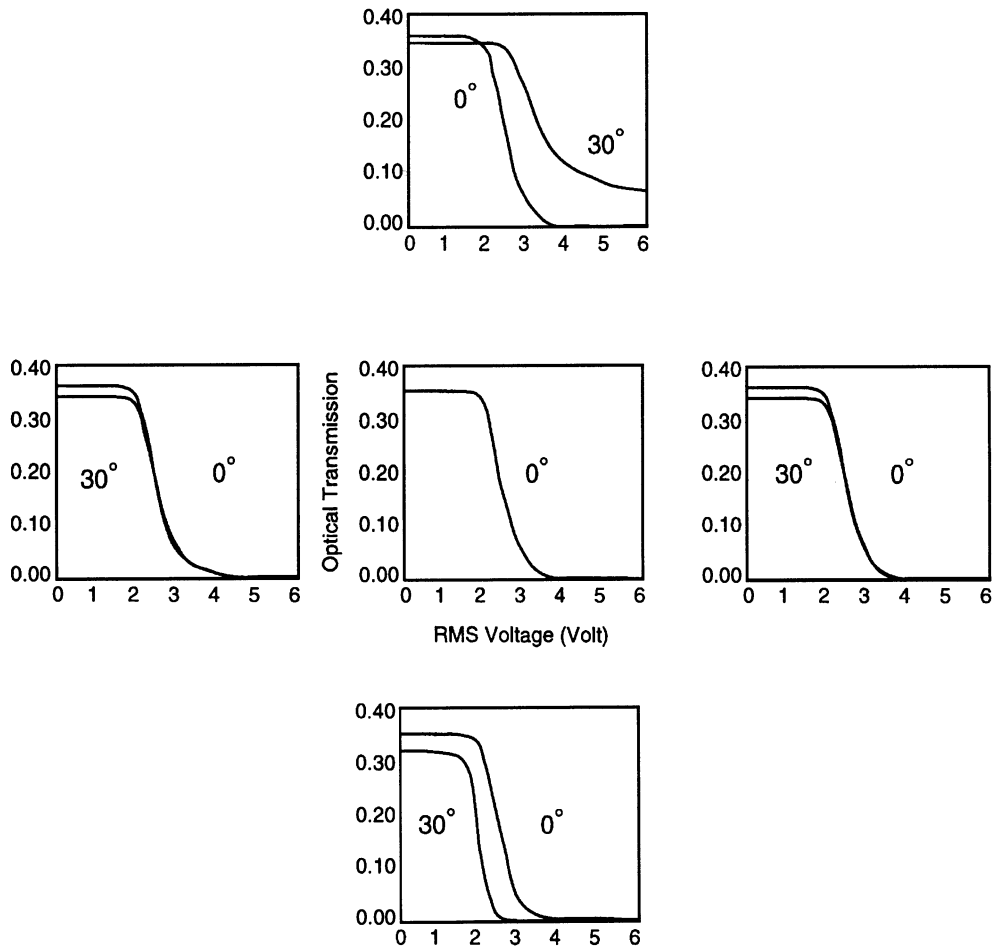


Figure 6. Viewing angle dependence of transmission versus voltage for a typical TN LCD. The figure shows the transmission for on-axis and 30° off-axis in four directions.

where N is the number of rows in the display. A transmission—voltage curve (Fig. 5) with a shallow slope requires a large voltage ratio, $V_{\text{on}}/V_{\text{off}}$, for an acceptable contrast ratio, thus limiting the display size to a small number of rows. To some extent, the steepness of the luminance-voltage curve of the TN displays can be somewhat improved by optimizing liquid crystal material parameters. However, passive matrix TN displays are still not suitable for fabrication of high information content displays requiring large number of rows and columns. The steepness of the luminance-voltage curve can be dramatically improved using the supertwisted nematic (STN) approach [24], employing a much higher twist angle such as 270°. Thus STN approach allows fabrication of LCDs with large number of rows for high information content display applications. The relative ease of the STN display manufacturing and their low cost, has created a huge

demand for these displays for note book computer applications, in spite of their marginally acceptable performance. To enhance the operating margin for improved contrast ratio, DSTN (dual scan STN) configuration is used. In a DSTN, the display is separated into two halves, and the rows in each half are scanned simultaneously and synchronously, to essentially double the duty ratio of the “on” pixels to increase the contrast ratio.

One of the major shortcoming of the STN display is the slow response time of the liquid crystal which is of the order of 150 m Sec. This low response time is not adequate for video applications and barely fast enough for a graphical interface of a computer. The response time of the STN LCDs can be improved by active addressing or multi-line addressing techniques [25, 26]. These techniques involve simultaneous addressing of several rows of a display to suppress the frame response

problems of conventional STNLCDS. The cost and performance of the active addressed PMLCDs is expected to lie in between that of a conventional STNLCD and an AMLCD. The active addressed STNLCDS are more suited to the medium information content display applications requiring video response rate. Even with active addressing the STNLCDS will still be limited by the multiplexibility limit of the PMLCDs [23].

In addition to the slow response time, the performance of a STNLCD is inferior to that of an AMLCD with respect to contrast ratio, grayscale, viewing angle, and color gamut. AMLCDs offer a significantly higher potential for meeting the performance requirements for the future high information content portable systems. Further, the recent major development efforts have reduced the cost difference between a STNLCD and an AMLCD considerably, making AMLCD the most likely technology of choice for future portable systems. AMLCD is most suitable for applications requiring high image quality such as from a CRT, but having a flat profile, lower power consumption, lower weight, and viewability under a variety of ambient lighting conditions. During the past ten years AMLCD technology has progressed from being a laboratory novelty to having commercial viability for applications ranging from

hand held TVs to portable notebook computers, engineering workstations, and avionic displays. AMLCDs are in current use for large area direct view displays as well as small high resolution light valves for helmet/head mounted displays (HMD) for military and commercial applications as well as large area projection displays. AMLCD with sizes of up to 28" diagonal have been demonstrated for direct view applications [27]. Similarly AMLCDs with high resolutions such as 1440×1024 and a pixel size in the range of $25 \mu\text{m}$ [28] have been demonstrated for HDTV projector and HMD applications. In the following we will discuss the active matrix technology, AMLCD components, display module electronics, optical performance and efficiency characteristics, and future developments.

5.1. Active Matrix Technology

Active matrix addressing removes the multiplexing limitations [23] of the PMLCDs, by incorporating a nonlinear control element in series with each pixel, and provides 100% duty ratio for the pixel, using the charge stored at the pixel during the row addressing time. Figure 7 illustrates an active matrix array with

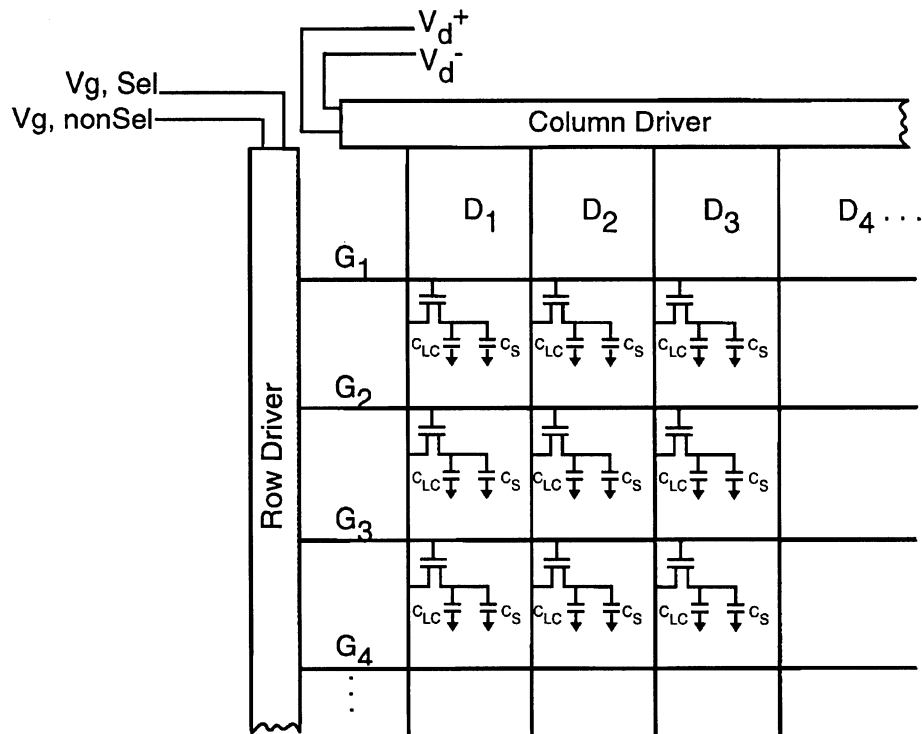


Figure 7. Schematic of a TFT active matrix array.

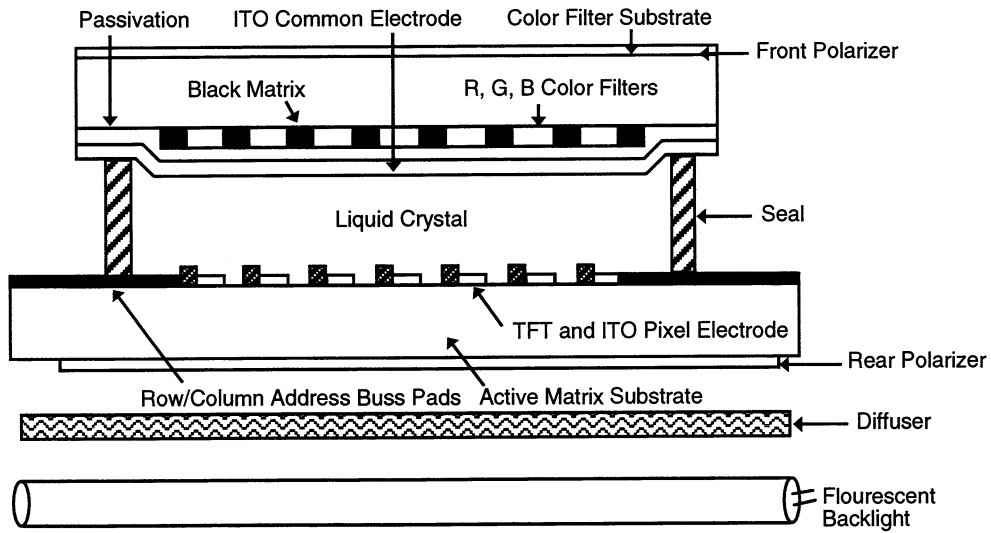


Figure 8. Schematic cross-section of an AMLCD.

row and column drivers. In the figure C_{LC} and C_s represent the pixel capacitance and the pixel storage capacitance. Figure 8 shows the cross section through an AMLCD illustrating various elements of the display. Figure 9 shows a typical AMLCD pixel, showing the gate and data busses, a thin film transistor (TFT), ITO pixel electrode, and the storage capacitor buss.

Fabrication of the active matrix substrate is one of the major aspects of the AMLCD manufacturing. Both two terminal devices such as back to back diodes [29], and metal-insulator-metal (MIM) diodes [30] as well as three terminal thin film transistors (TFTs) are developed for active matrix addressing. While 2-terminal devices are simple to fabricate, and cost less, their lim-

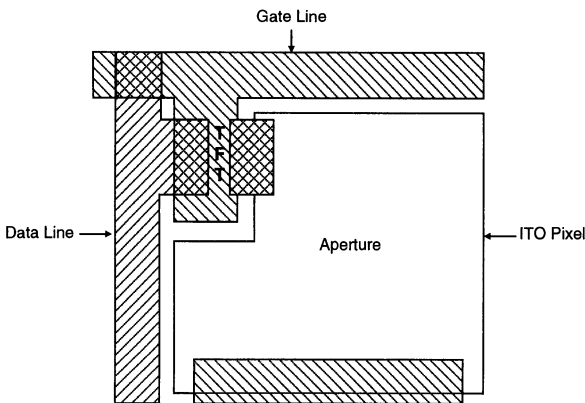


Figure 9. Typical TFT-LCD pixel layout.

itations include difficulty in achieving uniform device performance (breakdown voltage/threshold voltage) over a large display area, and lack of total isolation of the pixel when neighboring pixels are addressed. As a result, most of the current AMLCDs use TFT for the active matrix device, which provide a complete isolation of the pixel from the neighboring pixels.

Figure 10 shows the electrical equivalent of a TFT-LCD pixel, the display drive wave forms and the resulting pixel voltage. As in most matrix addressed displays with a line-at-a-time of addressing, the rows (gates) are scanned with a select gate pulse, $V_{g,sel}$, during the frame time t_f , while all the pixels in a row are addressed simultaneously with the data voltage $\pm V_d$ during the row time $t_r (= t_f/N)$. During the row time the select gate voltage, $V_{g,sel}$, “turns-on” the TFT and charges the pixel and the storage capacitor to the data voltage V_d . After the row time, the TFT is “switched-off” by application of the non-select gate voltage, $V_{g,non-sel}$; the voltage (charge) at the pixel is isolated from the rest of the matrix structure until the next frame time. Note that the LC pixel must be driven in an AC fashion with $+V_d$ and $-V_d$, during alternate frame periods, with no net DC across the pixel. A net DC voltage across the pixel results in flicker and image sticking effects [31]. Large and sustained DC voltages degrade the LC material due to electrolysis. The shift in pixel voltage, ΔV_p shown in Fig. 10, at the end of the row time is due to the parasitic gate-to-drain capacitance, C_{gd} , of the TFT. When the gate voltage is switched, the distribution of the charge from the TFT gate dielectric causes

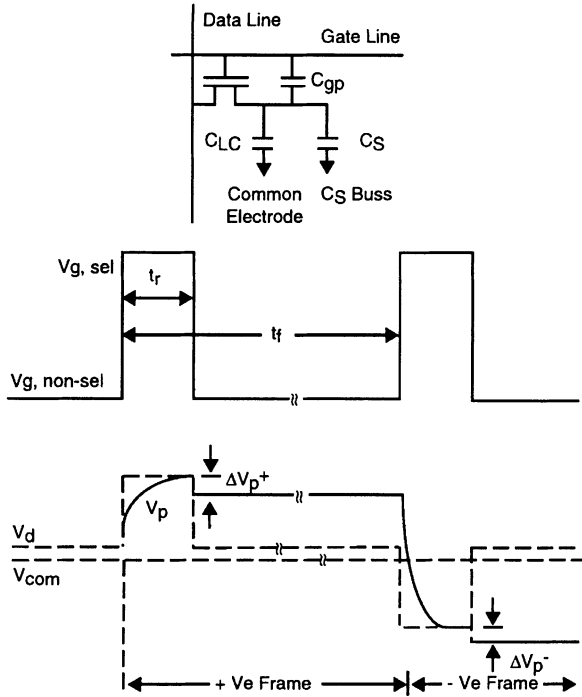


Figure 10. Electrical equivalent of a TFT-LCD pixel, and its operation.

the pixel voltage shift, ΔV_p , given by:

$$\Delta V_p = V_{g,sel} * C_{gd} / (C_{gd} + C_{lc} + C_s)$$

For the n -channel enhancement mode TFT switching device used, this voltage shift ΔV_p is negative for both the $+ve$ and $-ve$ frames, and thus it helps pixel charging in the negative frame and hinders it in the $+ve$ frame. Further, due to increased gate bias during the $-ve$ frame, the pixel attains the data voltage much more rapidly during the addressing period. Hence, the TFT is designed for the worst case $+ve$ frame conditions. ΔV_p is reduced by minimizing C_{gd} by decreasing the source drain overlap area of the TFT, and by using a storage capacitor, to minimize DC voltage shift across the pixel. Further ΔV_p is compensated by adjusting the common electrode voltage V_{com} as shown in Fig. 10. Note that C_{lc} is a function of the V_p (V_{lc}) due to the dielectric anisotropy of the LC, and hence adjustment to V_{com} alone does not eliminate DC for all graylevels. Either special addressing techniques [31] or modification of the grayscale voltages is required to compensate for the dielectric anisotropy of the LC.

Presently, most of the AMLCDs are fabricated using amorphous silicon (a-Si) TFTs. The advantages

of the a-Si TFTs include low processing temperatures (compatible with the use of glass substrates), large area deposition capability, and compatibility with the well established silicon IC industry. a-Si TFT has a typical mobility of $0.5 \text{ Cm}^2/\text{V.Sec.}$ which is adequate for active matrix devices. However, this low mobility is not suitable for the fabrication of row and column driver circuitry for a high information content displays, which require large bandwidth drivers, which in turn requires higher mobility TFTs. In addition to lower mobility, a-Si TFTs are characterized by a higher threshold voltage (typically 3–4 Volts), threshold voltage instabilities due to gate bias stress particularly at elevated temperature operation, and difficulty in fabricating self aligned source/drain-to-gate structure. These limitations have motivated the development of polysilicon, and single crystal silicon TFT technologies. We will discuss these technologies below under future developments.

5.2. Display Components

The major display components include fluorescent backlight, diffuser, rear polarizer, liquid crystal cell assembly comprising the active matrix substrate and the color filter substrate sandwiching the liquid crystal material, and front polarizer (see Fig. 8). The row and column drivers are attached to the liquid crystal cell assembly (not shown in Fig. 8) either by a TAB or chip-on-glass assembly techniques. The front polarizer of the display may include antireflective and EMI coatings. The display may further include compensation films between the display glass and polarizers for enhanced viewing angle performance [32].

The optical efficiency of the AMLCD may be examined by considering the optical losses at each of the components. Starting from the diffuser, the rear polarizer transmits only a maximum of 50% of the incoming light. This transmission is then reduced by the aperture ratio of the pixel, and the transmission through the color filters. Using a typical aperture ratio of 50%, and R , G , B color filter transmission of about 33%, the maximum transmission of the light from the diffuser can be calculated to be about 8%. In reality, due to absorptive losses and reflection losses the polarizer transmits only about 43%. Also, due to mismatch of the backlight spectrum and the color filter transmission spectrum, the transmission of the color filters is only about 25%, and the reflection losses at each optical interface in the display, reduce the optical transmission by another 10%. This results in a typical display transmission to about 4.5%.

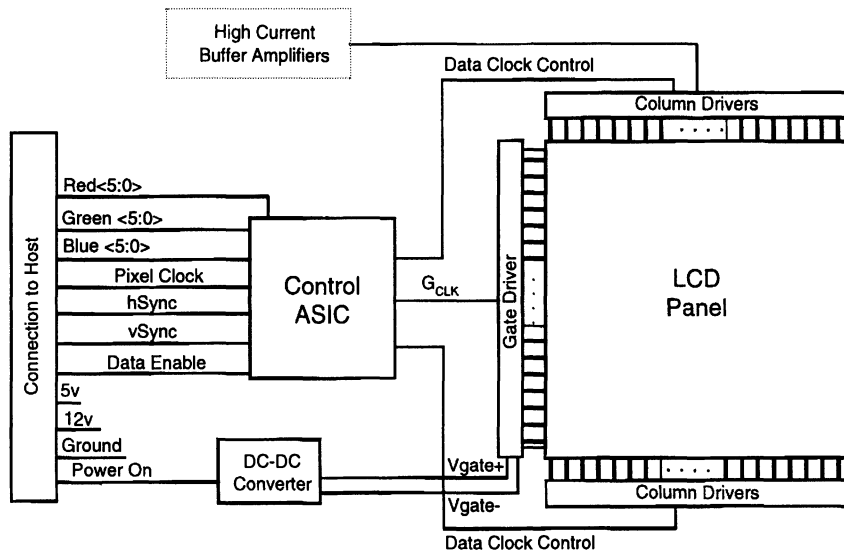


Figure 11. AMLCD module electronics block diagram.

On top of this, when we consider that the lamp to diffuser coupling is typically about 50% efficient, it can be seen that AMLCD is optically a very inefficient device. However, there are many opportunities for improving the optical efficiency of an AMLCD in the future. These will be discussed under the future developments.

5.3. Display Module Electronics

Figure 11 shows an example of a block diagram for an AMLCD module electronics [33]. The row and column drivers are typically mounted on a TAB which is interconnected to the row and column electrodes on the display glass. In some cases the row and column drivers are mounted directly on the row and column electrodes of the display glass (chip-on-glass). The control block and power supply generation means are separately mounted on a PC board and connected to the row and column drivers on one side and to the host controller on the other. The control block may include level shifters, timing generators, and analog functions in some cases. Essentially, the purpose of the control block is to take in digital data from the host system, which is typically a graphics controller chip and convert it into timing and signal levels required by the row and column drivers. The architecture and design of the module electronics encompassing row and column drivers have a significant impact on not only the display system cost and power consumption, but also the image quality.

The liquid crystal material typically requires about 5 V to achieve optical saturation (see Fig. 5). Considering the need for an AC drive (Fig. 10), the required voltage swing across the LC material is about 10 V. To achieve this 10 V swing across the LC material, the column drivers typically use 12 V power supplies. The requirement for the gate voltage driver outputs $V_{g,sel}$ and $V_{g,non-sel}$ is as follows: The $V_{g,sel}$ must be higher than the most positive column voltage by at least the TFT threshold voltage (V_T). The $V_{g,non-sel}$ must be lower than the lowest column voltage by at least the TFT threshold voltage. This ensures that TFT stays “turned-on” to charge the pixel to the desired column voltage during the addressing period, and stays “turned-off” during the non-addressing period to hold the pixel charge. Column driver voltage can be reduced by using V_{com} modulation drive method. In this method, the V_{com} node (which is connected to all pixels in the display) is driven above and below a 5 V range of the column drivers. Each and every row time, the V_{com} node is alternated between a voltage above and a voltage below the 5 V output range of the column drivers. This achieves 10 V across the LC material using 5 V column drivers. This method requires additional components and consumes additional power due to the oscillation of the V_{com} node. In addition, to avoid capacitive injection problems, the row drivers usually have their negative supply modulated with the same frequency as the V_{com} node. Note however, that compared to 10 V column drivers, 5 V column drivers

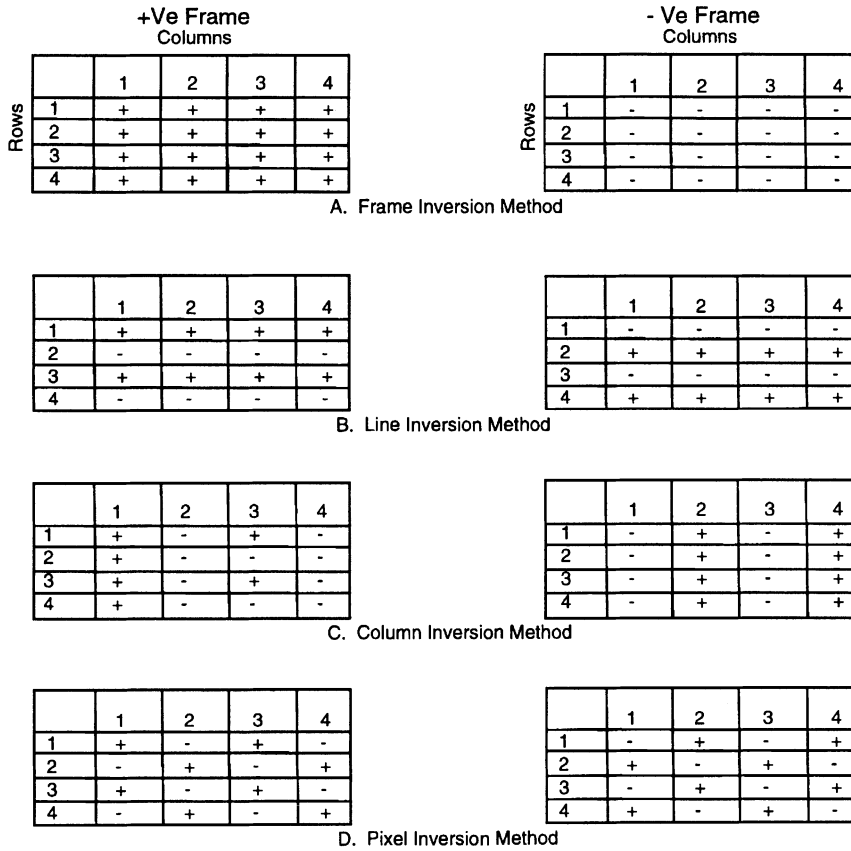


Figure 12. Typical AMLCD Polarity Inversion Methods.

consume less power, and are simpler to design and fabricate using small geometry CMOS.

Polarity inversion method is used to eliminate DC voltage across the liquid crystal, as well as to eliminate the influence of pixel flicker on the display image quality. The type of polarity inversion method used has an impact on the power consumption. Figure 12 shows the four widely used polarity inversion methods. In the frame inversion method, all pixels in one frame period are driven to +ve polarity in one frame period, and then all of them are driven to the -ve polarity during the next frame period. This method consumes the lowest driver power. However, it is sensitive to flicker due to slight transmissivity mismatch between the +ve and -ve polarities. It is also sensitive to horizontal and vertical cross-talk. As a result, this method is not generally employed when high image quality is required. In the other methods, flicker is eliminated by spatial averaging of the adjacent pixels with +ve and -ve polarities with slight transmissivity mismatches. In the line inversion method, the polarity

of pixels in the adjacent rows is alternated. This method is compatible with the V_{com} modulation drive scheme. It has much reduced sensitivity to vertical cross-talk, and more propensity for horizontal cross-talk. Also, it consumes more power than column inversion method, because the capacitance of all the row busses is charged and discharged every row time. In the column inversion method, alternate columns are driven with +ve and -ve polarities. This method is not compatible with the V_{com} modulation drive scheme and requires higher voltage column drivers. It has greatly reduced horizontal cross-talk, and low-power operation. Pixel inversion method is the ultimate scheme for spatial averaging of the +ve and -ve polarity pixels. In this scheme, the polarity of each pixel is inverted from the polarity of each of its neighboring pixels, by a combination of simultaneous row and column inversion. This produces the highest quality image, by total elimination of flicker and cross-talk. This method is incompatible with the V_{com} modulation drive scheme, and thus requires high voltage column drivers. Also, it consumes

more power due to the row inversion component. For a typical 10.4" diagonal VGA display the power dissipation for driving a global V_{com} node can be estimated to be in the range of 200 mW–350 mW [33]. This has to be balanced against the simplicity and lower cost of the low voltage column drivers.

The current trends in display module electronics developments are aimed at going to higher gray shades (going to 8 bit column drivers from the current 6 bit drivers) to achieve true colors, full motion video, lower power consumption (going to 0.5 W for SVGA from the current 1.5 W for driver power), higher levels of integration (higher column driver output count), and smaller bezel width for maximizing the display size for a given portable system size.

5.4. *AMLCD Optical Performance, and Efficiency Characteristics*

The relevant optical performance measures for an AMLCD include:

- Luminance range
- Luminance contrast ratio
- Color gamut
- Viewing angle
- Graylevel stability across the viewing angle
- Chromaticity stability as a function of graylevel and viewing angle
- Response time, and
- Reflectivity

For the display to be viewable comfortably under a variety of ambient lighting conditions ranging from dark to full sun light, it should have a controllable (dimnable) luminance range extending up to 100 fL. This luminance range can be presently achieved with a fluorescent backlight and a diffuser with appropriate control circuitry. Note that in a transmissive LCD, the luminance can be arbitrarily increased by merely using a brighter backlight (provided thermal issues are manageable). Unfortunately, use of brighter backlights increases the power consumption, and thus is not a preferable approach for portable systems. As a trade-off between acceptable luminance and lower power consumption, the present AMLCDs for notebook computers typically use a maximum display luminance of about 25 fL. The present AMLCDs achieve an on-axis luminance contrast ratio in excess of 100:1 (in a dark ambient) which is acceptable.

Color gamut is a function of the spectral characteristics of the backlight used and the color filter transmission characteristics. By suitable choice of these parameters, AMLCD can achieve a color gamut comparable to a that of a high quality CRT. However, in many of the present note book computer displays, the color filter transmission characteristics are chosen to achieve a higher transmission and thus higher display luminance to reduce power at the expense of color gamut. This results in unsaturated display colors.

LCDs have been notorious for having a very limited viewing angle. In general, portable systems do not require a very wide viewing angle such as $\pm 60^\circ$, that avionic displays with cross-cockpit viewing, and multi-viewer television displays characteristics. However, it is desirable for portable systems to have viewing angle up to $\pm 45^\circ$, for comfortable viewing in a variety of situations. LCDs have intrinsically poor viewing angles compared to emissive displays. The asymmetry of the LC director orientation around the display normal, and the variation of retardation of the liquid crystal cell with viewing angle result in variation of the off-state luminance with viewing angle. This leads to a decreased contrast ratio, as the viewing angle is increased. Figure 13 shows the contrast ratio as a function of horizontal and vertical viewing angles in a typical note book computer display today. A significant drop in contrast ratio at large viewing angles can be clearly seen in Fig. 13.

In addition to a high contrast ratio, grayscale displays must have stable graylevel luminances across the viewing angle. Figure 14 shows the luminance variation as a function of viewing angle for 8 graylevels in a typical AMLCD used in the present note book computers. While the graylevel luminances are reasonably stable in the horizontal direction, they vary significantly in the vertical direction, becoming brighter in the upper direction and darker in the lower viewing direction. Notice the graylevel inversion at larger viewing angles. This instability of graylevel luminances leads to chromaticity tolerances problems and color shifts that are not acceptable in a high quality display. Thus, presently the viewing angle of an AMLCD is limited by the lower contrast ratio, graylevel inversion, and unacceptable color shifts. However, several approaches are being developed to overcome these limitations which will be discussed under the section "future developments".

Due to the advances in the liquid crystal materials and the AMLCD drive techniques, the response time

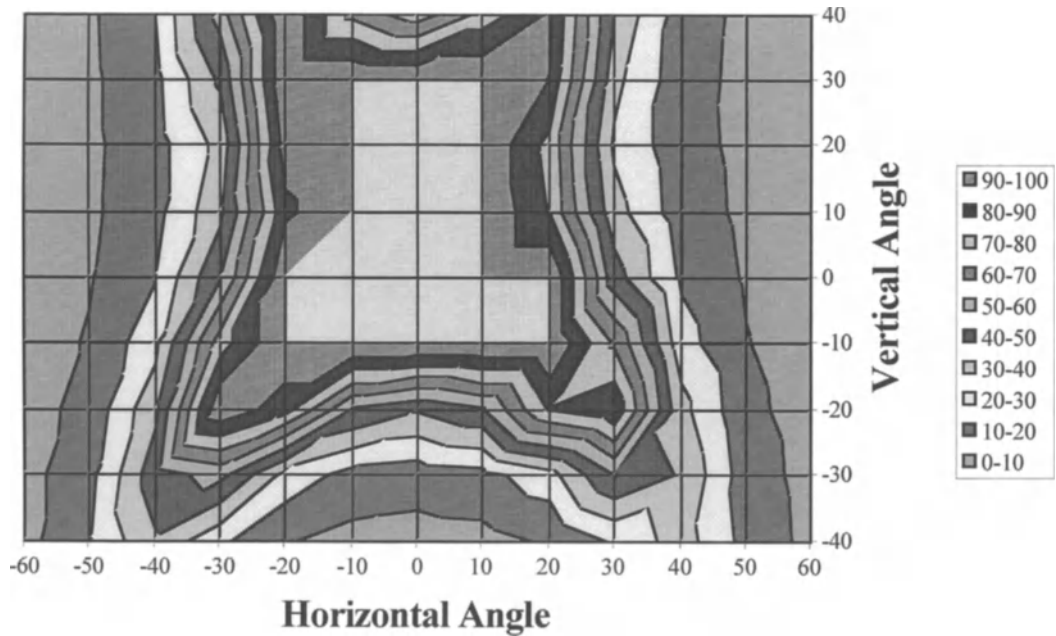


Figure 13. Contrast ratio as a function of horizontal and vertical viewing angles in a typical normally white AMLCD.

(raise time + fall time) of the AMLCDs used in the current notebook computers is less than about 40 m sec which is adequate for full motion video at 72 fps.

For viewability under high ambient lighting conditions, the reflectivity of the display must be low. Reflection of the ambient light from the display panel reduces the contrast ratio. Presently, a typical AMLCD used in a note book computer has a reflectivity of about 2%. This is only marginally acceptable now, and too high to meet the requirement of a display for the future portable multi-media systems. It must be reduced to less than 0.75% by use of improved AR films, and by reducing reflectivity of various layers in the display panel.

5.5. Future Developments

Presently, there is much development activity aimed at reducing the deficiencies of AMLCDs for high information content portable systems with low power consumption. One of the major activities is the development of a high mobility TFT (on glass). This allows fabrication of row and column driver circuitry on glass, to eliminate the cost of externally connected silicon driver chips, improve reliability, and reduce package size and weight. The ultimate vision

of this development is to fabricate other functions—logic, memory and control circuitry—also on the display glass (system-on-glass). Quartz substrates and high temperature TFT device processing can be used for fabricating single crystal silicon thin film transistors [34, 35] with a typical mobility of over 600 $\text{cm}^2/\text{V} \cdot \text{sec}$, for small high resolution displays with integrated drivers for HMD applications. However, lower temperature TFT device processes are required for large area displays. Recently, significant progress has been made on low temperature polysilicon TFTs, and mobility in excess of 100 $\text{cm}^2/\text{V} \cdot \text{sec}$ has been demonstrated. Sieko Epson has reported a 10.4" diagonal SVGA display with integrated 6 bit data drivers, using low-temperature (425°C) polysilicon TFTs [36]. Driver voltage reduction is another future development trend. Because power consumption is proportional to the square of the voltage (V^2), significant reduction of the panel power can be achieved by this approach. While 5 V drivers are more common now, the conversion to 3.3 V has already been started. This requires development of new LC materials with a low switching voltage, as well as advances in the TFT technology.

Weight is a significant parameter for portable systems. While mostly 1.1 mm thick glass has been utilized for the AMLCD fabrication now, it is being replaced with 0.7 mm thick glass. Use of 0.5 mm thick

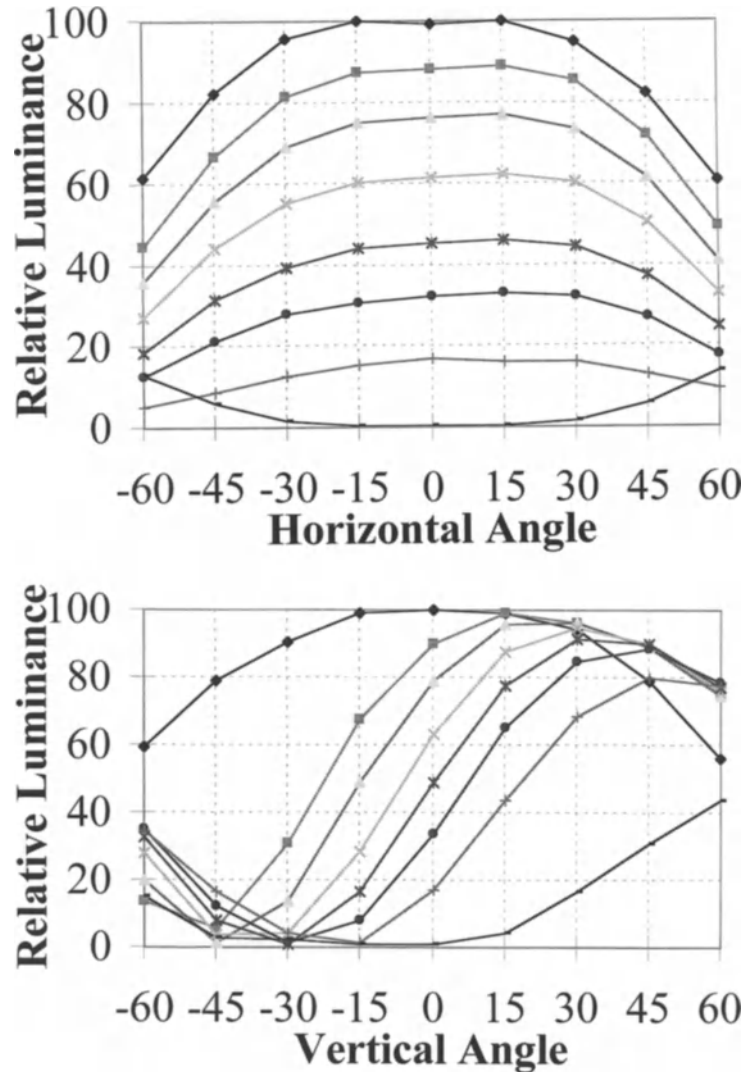


Figure 14. Graylevel luminance stability as function of horizontal and vertical viewing angles.

glass is being considered for future displays. For a 10.4" dia. display, replacing the 1.1 mm glass with 0.7 mm glass will reduce the display weight from 225 gms to 145 gms. Plastic substrates are also being developed as a replacement for glass substrates for further weight reduction and improved durability [37].

As discussed earlier, power consumption is a major factor for portable systems. Two major areas for reducing AMLCD power consumption are: 1) increasing the aperture ratio of the pixel and 2) use of a rear reflective polarizer [38, 39] instead of the absorptive polarizer. By using field shielding techniques [40, 41] and by use of smaller design rules the active area of the pixel can

be maximized by decreasing (or eliminating) the inactive space between the pixel electrodes and the gate and source buss lines. Sharp [41] has demonstrated 82% aperture ratio for 10.4" diagonal AMLCDs with SVGA resolution. Conventional absorptive linear polarizers reduce the optical efficiency by at least 50% by absorbing the light that is polarized perpendicular to the pass-axis of the polarizer. A reflective polarizer would reflect this light of wrong polarization for polarization rotation and re-imaging to potentially double the efficiency of the LCD. Two approaches [38, 39] have been reported for reflective polarizers. One approach is to use retro-reflective sheet polarizers [38].

This microprism based retroreflector performs satisfactorily at normal incidence but exhibits color shifts off-axis. The second approach for the reflecting polarizer is based on the use of a cholesteric LC film as a circular polarizer, and a quarter wave film to generate linearly polarized light. Using this approach Merck has reported 80% improvement in brightness for the same power consumption [39].

Several options are available and being developed to enhance the viewing angle of a display. The various viewing angle enhancement techniques include: halftone grayscale [21], dual domain technique [42], compensation films [32], SpectraVue™ backlight collimation and front viewing screen method [43], OCB (optically compensated bend) mode [44], and IPS (in-plane switching) mode [45]. Some of these approaches may not be suitable for use in portable systems due to their impact on the display transmission and thus its power consumption. For example, the IPS mode may not be suitable for portable systems because of its very low transmission, even though it provides excellent viewing angle and image quality. Based on the rapid progress being made on the viewing angle enhancement methods driven by the workstation display market, it is expected that AMLCD viewing angle will not be a major limitation for its use in the future high performance portable systems.

Reflective displays offer the most potential for reducing the power consumption. Unlike selectively modulating the transmitted light in a backlit LCD, reflective LCDs selectively reflect the ambient light, eliminating the need for backlight power. When ambient light is not adequate, front lighting can be provided to make these displays viewable [46]. Reflective color displays using guest-host mode [47], and PDLC mode [48, 49] have been reported. One of the major requirements for a high quality reflective display is a high level of spectral reflectance, to achieve good color selectivity and acceptable reflected luminance. Conventional polarizer and color filter based LCD approaches described above are not suitable for a reflective mode, because of the very low transmitted light intensity (typically less than 5%) that is available for reflection. Both the guest host mode and PDLC mode do not require polarizers, and thus can provide brighter reflective displays. Recently, a holographically formed [49] liquid crystal polymer reflective display has been reported. It consists of a liquid crystal (LC) and polymer multi-layer structure that reflects light at specific wavelength and transmit the light at other wavelengths.

The reflection intensity can be controlled electrically. This device does not use either polarizers or color filters, and hence, can provide bright color images.

Monochrome, reflective LCDs have also been developed using cholesteric liquid crystal materials [50]. Polymer stabilized cholesteric texture (PSCT) type reflective LCDs show a potential for use in electronic book, electronic newspaper and document viewing applications. PSCT-LCDs are bistable, the stable states being reflecting planar state and non-reflecting focal conic state. The bistability allows the display to retain the image without consuming any power. A 14" diagonal high information content, monochrome, passive matrix reflective cholesteric display has been demonstrated [50] with a resolution of 200 dpi, and a pixel matrix of 2240×1728 pixels. These displays can be bright because no polarizer is needed and the pixel aperture ratio can be very high. The present major drawbacks of the reflective cholesteric liquid crystal technology are lack of full color and video speed.

The current development efforts in reflective LCDs are focused on achieving improved color performance, resolution, grayscale, viewing angle, and brightness.

6. Field Emitter Display Technology

The field emitter display (FED) is essentially a flat CRT. It consists of a two-dimensional array of matrix-addressable electron sources that is proximity focused on another two-dimensional array of phosphor dots. The FED is an emissive flat panel display based on cathodoluminescence-similar to the CRT; however, unlike the CRT which relies on the sequential addressing of phosphor dots by a single electron source, the FED uses multiple electron sources per pixel which are proximity focused on the phosphor dots. In one respect it bears some similarity with AMLCD because the two-dimensional array of electron sources are matrix addressed. The electron sources are on a base-plate while the phosphor dots are on a face-plate. The face-plate is separated from the base-plate by a vacuum gap using physical standoffs or spacers [51–56].

A typical field emitter display is shown in Fig. 15. The electron sources are sharp cones that emit electrons by quantum mechanical tunneling in the presence of a high electric field. The application of a positive voltage to the gate electrode with respect to the emitter results in an intense electric field at the emitter surface leading to electron tunneling or cold cathode electron emission. The electrons are emitted normal to the base-plate and

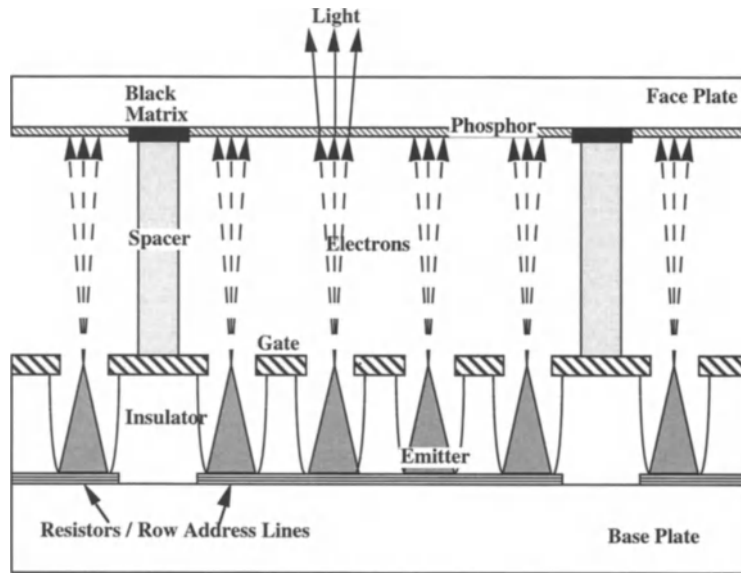


Figure 15. Device structure and operating principles of a field emitter display (FED).

travel to the face-plate which is biased at a voltage higher than the base-plate. The electrons bombard the phosphor and create a luminous image which is seen by the viewer.

The key features of a field emitter display are:

- Two-dimensional array of electron sources on a base plate
- Two dimensional array of phosphor dots on a face-plate
- Proximity focusing of emitted electrons on phosphor
- Redundant electron sources per pixel.

6.1. FED Operating Principles

The base plate of the FED consists of a two-dimensional array matrix-addressable electron sources. There are several approaches to generating electron emission and addressing the electron emitters. Similar to all matrix addressing schemes, the FED relies on the non-linear current voltage characteristics of field emitters. Figure 16 shows the current-voltage characteristics of a typical field emitter array. One approach to addressing the field emitter arrays for a video-display application is to use the following scheme: Each pixel consists of multiple field emitters. Each field emitter has an emitter and a gate (extraction) electrode. Rows of pixels consisting of multiple field emitters are electrically connected and are placed in parallel with other

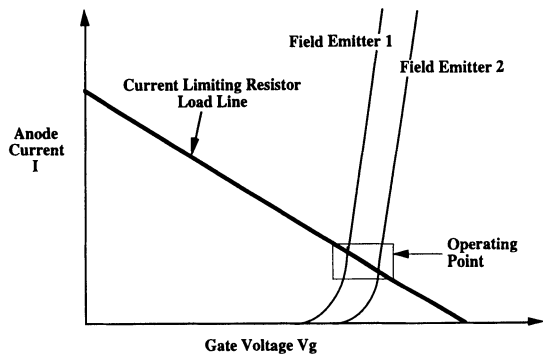


Figure 16. Current-Voltage characteristics of a field emitter display. The figure indicates the role of a resistive load line in established an operating point.

rows of pixels. On the other hand, the gates (extraction grids) of the field emitters in each pixel are electrically connected in parallel columns which are orthogonal to the rows of emitters. In this configuration, the emitter array associated with each pixel is uniquely defined by the intersection of a specific emitter row and a specific gate (extraction) grid column. The pixel is selected by electrically addressing the row corresponding to the pixel while simultaneously addressing the corresponding column [57].

Gray scale and color rendition are two important issues for video-display applications. The brightness of a pixel is proportional to the total electron charge delivered by the field emitter array to the phosphor of an

individual pixel within a frame. The charge delivered by the electron sources to the phosphor within a given frame can be varied by changing the time period within the frame at the activated site or changing the emission current produced during the activation. These two techniques are referred to as time division multiplexing or analog voltage gray scale methods. Cathodoluminescence also has the property that phosphor continues to emit photons after electron excitation has stopped. This is the key property that allows sequential addressing of phosphors in a CRT. Photon emission persists for a duration of the frame period even though the phosphor is only addressed for a brief portion of the frame period. Another factor is the nature of the human visual system. The human visual system allows the eye to perceive color by either spatially integrating the photons from the sub-pixels or temporally integrating the photons from the sub-pixels.

6.2. Field Emission

The electron sources in FEDs are field emitter arrays. Field emitters work on the principle of electron tunneling from a metal or semiconductor surface into vacuum when a high electric field is applied [58, 59]. Field emission occurs when the energy barrier width at the surface is of the order of the electron wavelength ($\sim 10\text{--}20 \text{ \AA}$). Typical metal and semiconductor surfaces have an energy barrier height of about 4.5 eV that prevents the electrons from escaping into vacuum. This implies a field of about $4.5 \times 10^7 \text{ V/cm}$ is required on the emitting surface. Obviously, this high electric field will be difficult to generate with reasonable voltage using a parallel plate capacitor. However, from electrostatics we know that high electric fields can be generated on surfaces with reasonable voltages if the surface has a small radius of curvature. The electric field at the tip of the curvature is

$$F_{\text{tip}} = \frac{V_g}{kr_0}$$

where F is the electric field, V_g is the applied gate (extraction) voltage, r_0 is the radius of curvature and k is constant that depends on the aperture width and cone height. “ k ” has a typical value of 2–5. For example, a field emitter with a radius of curvature of 50 \AA will be able to generate a field of $4.5 \times 10^7 \text{ V/cm}$ with an applied gate voltage of about 20 to 100 V.

Advances in integrated circuit technology and micro-machining technology have made the fabrication

of small field emitter structures with nanometer-range radius of curvature (1–10 nm) feasible. There are several approaches to fabricating these microstructures with small radius of curvature. They fall into the following broad categories:

- Point Emitters (Cone Field Emitter Arrays [60, 61])
- Line Emitters (Vertical Edge [62], Ridge [63], Lateral Thin-Film-Edge Field Emitter Arrays [64, 65])
- Micro-textured broad area emitters (Amorphous Diamond Field Emitters [66]).

In general, they are three terminal devices; however, two terminal and four-terminal devices are possible. Electrons are emitted from the cathode (emitter) by the application of a positive voltage to the gate (extraction) electrode with respect to the emitter. Emitted electrons are collected by the anode or phosphor screen.

In the simplest form, the terminal I-V characteristics are given by

$$I_s \sim I_e = a_{\text{FN}}(V_G)^2 \exp\left[-\frac{b_{\text{FN}}}{V_G}\right]$$

where a_{FN} and b_{FN} are constants that depend on the material properties and geometry [60], I_s and I_e are the screen and emitter currents respectively, and V_G is the applied gate voltage. The non-linear and essentially exponential I-V characteristics makes the field emitter array suitable for matrix addressing, without having to use a non-linear switching element such as a TFT in AMLCD.

6.3. Phosphor Screen

The faceplate of the FED is the phosphor screen. It is patterned into stripes or dots of red, green and blue phosphors. Two types of phosphors are used in FEDs. The majority of FEDs currently use low voltage phosphors. These phosphor screens have transparent conductor backing between the glass and the phosphors [51, 56]. Low voltage phosphors are used for two reasons: (i) the need to keep the beam size small requires that the distance between the base plate and the faceplate be kept to a minimum, and (ii) the concern about dielectric breakdown in the vacuum envelope when the distance between the faceplate and the base plate is too small, requires that the screen voltage be kept low. Typical separation between the face-plate and the base-plate is $200 \mu\text{m}$ and the screen voltage is 500 V. Low voltage phosphors have moderate luminous efficiency of about 5 lumen/watt.

Some FEDs use high voltage aluminized phosphors as in a CRT [67–70]. This requires an increase in the distance between the face-plate and the base-plate, due to dielectric breakdown considerations. Typical anode voltage bias is about 10,000 V for a typical base-plate to face-plate spacing. The phosphor is covered on the vacuum side by a thin Al film. High voltages allow the electrons accelerated toward the screen to penetrate the thin Al layer. High voltage phosphors have higher luminous efficiency (~ 25 lumen/watt).

For a given brightness, low voltage phosphors require higher cathode current density than high voltage phosphors. Phosphor lifetime is determined by total Coulomb charge build up. As a result of this high voltage phosphors have a longer life than low voltage phosphors because they (i) require lower cathode current densities for a given brightness and (ii) have higher luminous efficiency.

6.4. *FED Demonstrations*

PixTech of France has demonstrated a 6-in dia. FED with a brightness >100 fL, an intrinsic contrast ratio $>100:1$ and a response time of 2 ms. The average screen power was 0.01 W/cm², and the display had viewing angle characteristics of a CRT [Vaudaine 1991]. The display uses a low voltage phosphor and demonstrated a luminous efficiency of about 5 lumen/watt. Variants of the PixTech technology have been adapted for high brightness application [68] and low power application. These FEDs are based on Mo-cones fabricated by e-beam evaporation.

Micron Display Technology demonstrated an 0.7-in diagonal full color display for Camcorder and head mounted applications. The power consumption of the display is a factor five lower than an equivalent AMLCD display operating at 15 fL. The luminous efficiency is greater than 3 lumen/watt. The Micron FED display is based on Si-cone technology. It was fabricated using reactive ion-etching and chemical-mechanical polishing. A 14" diagonal display is under development [58].

Raytheon adapted the classical cone emitter developed by LETI for high brightness displays by using high voltage phosphors. This was accomplished by increasing the base-plate/face-plate separation and adding a focus grid between the face plate and the base plate. The grid focuses electrons emanating from the emitter on the phosphor. This reduces the spot size and improves resolution. This approach allows

Raytheon to use 10 KV screens and attain very high brightness and luminous efficiency. Brightness of up to 4200 fL flat field and 3500 fL video have been demonstrated with a luminous efficiency of 20 lumen/watt with P53 phosphor. The display is 512×512 , 4" per side monochrome display. They are working on a 512×512 , 6" per side color version [68]

The technologies described above rely on lithography technologies that have 1–1.5 μm gate aperture and require gate voltages of 100 V for device operation. The dynamic power consumed by the addressing circuitry for the display is proportional to the square of the voltage. Furthermore, the high voltage ICs required for driving the FEDs are expensive. The cost of FEDs and driver circuitry can significantly be reduced if the gate operating voltage is reduced. Silicon Video Corporation [69] and FED Corporation [70] have recently demonstrated small displays based on ultra-small gate aperture FEAs which require low gate operating voltage. Furthermore, they use high screen voltage.

The Silicon Video demonstrated a full color 2.4" dia., 1/4 VGA display with 80 lines/inch resolution, and a 120×140 pixel matrix. This display demonstrated a brightness of 20 fL (all white, 50% filter), 20% brightness uniformity, and 16 gray level operation. It uses high voltage aluminized phosphors (P22) operated at 4 kV to achieve high quality color, low power operation and long life. Ion tracking lithography technique has been used to define molybdenum cone emitters with 200 nm gate apertures [69].

The FED Corporation has demonstrated a 2.54" dia., 1/4 VGA display using interference lithography to fabricate ultra-small gate aperture Mo cones. Like the Silicon Video Corp. demonstration display, it uses high-voltage aluminized phosphors. It is a 512×512 -pixel, 256 level gray-scale, 700 cd/m² display consuming only 0.25 W. The luminous efficiency is over 30 lumen/watt [70].

6.5. *Research Activities in Field Emitter Displays*

Despite recent advances in FED technology, there are a number of issues that need to be resolved before the technology can reach the market or its full potential. These issues revolve around (i) gate operating voltage, (ii) current uniformity, reproducibility and reliability, (iii) emitter surface conditioning, (iv) luminous efficiency, (v) spacer technology, (vi) vacuum sealing and packaging, (vii) low voltage phosphors and (viii) obtaining high resolution with high voltage phosphors.

Bozler et al. demonstrated the reduction of gate operating voltage by reducing the gate aperture of FEAs [71]. The gate aperture of the devices was 160 nm and was defined using interference lithography (IL). Gate turn-on voltages of 25 V were demonstrated and this was reduced to about 10 V when the FEAs were flashed with Cs. Further reduction in gate operating voltage can be obtained if the aperture is decreased further.

There are several research groups that are developing high aspect ratio field emitter arrays to reduce the capacitance of the field emitters arrays and obtain the nearly ideal field emitter structure proposed by Utsumi [72]. Hori et al. demonstrated a high aspect ratio, tower structure field emitter array with large emission current and low gate voltage operation [73]. The device has high aspect ratio.

Other research activities focus on integrating a focusing electrode with the field emitter array. The purpose of the focus electrode is to reduce the spot size when high voltage phosphors are used and the faceplate-baseplate distance is increased [74–78]. The results of Palevsky et al. indicate that the integrated focusing electrode will be important for high resolution and high efficiency FEDs [68].

7. Summary and Conclusions

Advances in computing and communications are bringing about a paradigm shift in the direction of merging these technologies to provide anytime anywhere access to information, which is driving the development of high performance portable systems such as notebook computers, and wireless multi-media terminals such as InfoPad. These portable systems require thin, light weight, low power displays with a high image quality including high resolution, true color, and full motion video. The display is expected to handle a variety of data types such as text, graphics, and high resolution true color full motion video. Display is a critical subsystem of the portable system because it is the primary man-machine interface, and has a major impact on the system power consumption, and thus the battery life between charges. Currently, backlit AMLCD is the best display choice for a high performance portable system such as the present notebook computer, or the future wireless multi-media terminal. However, significant improvements in AMLCD performance are required to meet the requirements of the future portable systems. The areas of required improvements include power consumption, brightness, viewing angle,

grayscale, color rendition, cost etc. Meeting these future display requirements of reduced power consumption and cost, while significantly increasing the resolution, size, optical performance and functionality is challenging. Innovative light management techniques, reflective polarizers, reflective color filters, low voltage display driving, and integration of driver electronics on display glass are needed to address future performance requirements. Reflective LCDs offer a very low power display solution for portable systems where low power is of paramount importance. The current emphasis on the reflective color LCD development effort is expected to lead to improvement in the image quality of these displays, for application to certain niche applications. Reflective LCDs are not expected to match the image quality of the backlit TFT-LCD under a wide range of ambient lighting conditions.

Field emission display (FED) development is coming out of the research stage, and into the proto-type development phase. FED technology has made significant progress in recent years, and the performance of proto-type FEDs continues to improve rapidly. This technology continues to show potential for low power displays with high image quality which is required for the future portable systems. Longer term, FEDs may be an attractive alternative to LCDs for portable applications.

References

1. R. Bargodia et al., "Vision, issues and architecture for nomadic computing," *IEEE Personal Communications*, p. 14, Dec. 1995.
2. A.P. Chandrakasan et al., "Minimizing power consumption in digital CMOS circuits," *Proceedings of the IEEE*, Vol. 83, No. 2, p. 498, April 1995.
3. E.P. Harris et al., "Technology directions for portable computers," *Proceedings of the IEEE*, Vol. 83, No. 2, p. 636, April 1995.
4. J.I. Pankove, *Display Devices*, Springer-Verlag, ISBN-3-540-09868-2, 1980.
5. H.J. Plach et al., "Liquid crystals for active matrix displays," *Solid State Technology*, p. 186, June 1992.
6. C.N. King, "Electroluminescent displays," *Conference Record of the 1994 International Display Research Conference*, p. 69, Oct. 1994.
7. R. Khormaei, et al., "A 1280 × 1024 active matrix EL display," *Digest of Technical Papers, International Display Symposium of the Society of Information Display*, p. 891, May 1995.
8. L. Arbuthnot et al., "A 2000 lpi active-matrix EL display," *Technical Digest of International Symposium of the Society for Information Display*, p. 374, May 1996.
9. P.S. Friedman, "Materials issues related to large size color plasma displays," *Conference Record of the 1994 International Display Research Conference*, p. 69, Oct. 1994.

10. T. Nakamura et al., "Drive for 40-in.-diagonal full-color ac plasma display," *Technical Digest of International Symposium of the Society for Information Display*, p. 807, May 1995.
11. H. Doyeux et al. "A high resolution 19-in. 1024 × 768 color ac PDP;" *Technical Digest of International Symposium of the Society for Information Display*, p. 811, May 1995.
12. J.L. Deschamps, "Recent developments and results in color-plasma-display technology," *Technical Digest of International Symposium of the Society for Information Display*, p. 315, May 1994.
13. R.W. Schumacher, "Automotive display trends," *Technical Digest of International Symposium of the Society for Information Display*, p. 9, May 1996.
14. J.R. Troxell et al., "TFT-addressed high-brightness reconfigurable vacuum fluorescent display," *Technical Digest of International Symposium of the Society for Information Display*, p. 153, May 1996.
15. J.R. Troxell et al., "Thin-film transistor fabrication for high brightness reconfigurable vacuum fluorescent displays," *IEEE Transactions on Electron Devices*, Vol. 43, No. 5, p. 706, May 1996.
16. K. Kinoshita et al., "Active-matrix VFD with phosphor on memory chip," *Technical Digest of International Symposium of the Society for Information Display*, p. 452, May 1996.
17. P. Pleshko et al., "Overview and status of information displays," SID 1992 Seminar Lecture Notes, Vol. 1, M-0, pp. 1–76, Boston, USA.
18. B.S. Scheuble, "Liquid crystal displays with high information content," SID'91 Seminar Notes, Vol. II, F-2, 1991.
19. T.J. Scheffer, "Super twisted nematic (STN) LCDs," SID'95 Seminar Notes, Vol. I, M-2, 1995.
20. C.H. Gooch and H.A. Tarry, *J. Phys. D: Appl. Phys.* Vol. 8, p. 1575, 1975.
21. K.R. Sarma, H. Franklin, M. Johnson, K. Frost, and A. Bernot, "Active matrix LCDs using grayscale in halftone methods," *Proc. of the SID*, Vol. 31, No. 1, p. 7, 1990; K.R. Sarma et al., *SID'91 Digest*, p. 555, 1991.
22. E. Haim, R. Mc Cartney, C. Penn, T. Inada, T. Unate, T. Sunata, K. Taruta, Y. Ugai, and S. Aoki, "Full-color grayscale LCD with wide viewing angle for avionic applications," *SID'94 Applications Digest*, p. 23, 1994.
23. P.M. Alt and P. Pleshko, *IEEE Trans. Electron Dev.*, Vol. ED-21, p. 146, 1974.
24. T.J. Scheffer and J. Nerring, *Applied Physics Letters*, Vol. 45, p. 1021, 1984.
25. T. Scheffer and B. Clifton, "Active addressing method for high-contrast video-rate STN displays," *SID'92 Digest*, p. 228, 1992.
26. H. Muraji et al., "A 9.4-in. color VGA F-STN display with fast response time and high contrast ratio by using MLS method," *SID'94 Digest*, p. 61, 1994.
27. Japan Electronics Show, Osaka, Oct. 1995; Sharp Corp. Exhibited 28" Diagonal AMLCD.
28. S. Higashi et al., "A 1.8-in Poly-Si TFT-LCD for HDTV projectors with a 5 V fully integrated driver," *SID'95 Digest*, p. 81, 1995.
29. Z. Yaniv et al., *SID'86 Digest*, p. 278, 1986.
30. M. Toyoma et al., "A large-area diode-matrix LCD using SiNx layer," *SID'87 Digest*, p. 155, 1987.
31. Y. Nano et al., "Characterization of sticking effects in TFT-LCD," *SID'90 Digest*, p. 404, 1990.
32. H.L. Ong, *Japan Display'92*, p. 247, 1992.
J. Mukai et al., "A viewing angle compensator film for TFT-LCDs," *Asia Display'95*, p. 949, 1995.
33. A. Erhart, "Module electronics for flat-panel displays," SID'95 Application Seminar Notes, 1995.
34. J.P. Salerno et al., "Single crystal silicon transmissive AMLCD," *SID'92 Digest*, p. 555, 1992.
35. K.R. Sarma et al., "Silicon-on-quartz (SOQ) for high-resolution liquid-crystal light valves," *SID'94 Digest*, p. 419, 1994.
36. S. Inoue et al., "425°C Poly-Si TFT technology and its applications to large size LCDs and integrated digital data drivers," *Asia Display'95*, p. 339, 1995.
37. A. Stein et al., "Plastic LCD substrates that combine optical quality and high use temperature," *SID'96 Applications Digest*, p. 11, 1996.
38. M.F. Weber, "Retroreflective sheet polarizer," *SID'93 Digest*, p. 669 1993.
39. D. Coates et al., "High-performance wide-bandwidth reflective cholesteric polarizers," *SID'96 Applications Digest*, p. 67, 1996.
40. S.S. Kim, et al., "High aperture and fault-tolerant pixel structure for TFT-LCDs," *SID'95 Digest*, p. 15, 1995.
41. "Japan Electronics Show," Oct. 1995, Sharp exhibited AMLCDs with VGA resolution with a pixel aperture ratio of over 82%.
42. K. Takatori, et al., *Japan Display'92*, p. 591, 1992.
43. S. Zimmerman et al., "Viewing angle enhancement system for LCDs," *SID'95 Digest*, p. 793, 1995.
44. T. Konno, et al., "OCB-cell using polymer stabilized bend alignment," *Asia Display'95*, p. 581, 1995.
45. M. Ohta, et al., "Development of super TFT-LCDs with in-plane switching mode," *Asia Display'95*, p. 707, 1995.
46. C.Y. Tai et al., "A transparent front lighting system for reflective-type displays," *SID'95 Digest*, p. 375, 1995.
47. S. Mitsui, Y. Shimada, K. Yamamoto, T. Takamatsu, N. Kimura, S. Kozaki, S. Ogawa, and T. Uchida, SID'92 Digest of technical papers, p. 437, 1992.
48. J.W. Doane, D.K. Yang, and Z. Yaniv, *Proc. 12th Intl. Display Research Conference*, p. 73, 1992.
49. K. Tanaka, K. Kato, S. Tsuru, and S. Sakai, *J. Society for Information Display*, Vol. 4, p. 37, 1994.
50. Z. Yaniv et al., "Electronic news paper display," *Asia Display'95*, p. 113, 1995.
51. R. Meyer et al., "Microtips fluorescent display," *Japan Display*, p. 513, 1986.
52. C.A. Spindt et al., "Field emitter arrays applied to vacuum fluorescent display," *IEEE Transactions on Electron Devices*, Vol. 36, No. 1, p. 225, Jan. 1989.
53. C.A. Spindt et al., "Field emitter arrays for vacuum microelectronics," *IEEE Transactions on Electron Devices*, Vol. 38, No. 10, p. 2355, Oct. 1991.
54. C.A. Spindt et al., "Field emitter array development for microwave applications," *1995 IEEE International Electron Device Meeting Technical Digest*, p. 389.
55. P. Vaudaine et al., "Microtips fluorescent display," *IEEE IEDM Technical Digest*, p. 197, 1991.
56. A. Ghis, et al., "Sealed vacuum devices: Fluorescent microtip displays," *IEEE Transactions on Electron Devices*, Vol. 38, No. 10, p. 2320, Oct. 1991.

57. F. Leroux et al., "Microtips display addressing," *Technical Digest of International Symposium of the Society for Information Display*, p. 437, May 1991.
58. R.H. Fowler et al., "Electron emission in intense electric fields," *Proceedings of the Royal Society, London, Series A*, Vol. 119, p. 173, 1928.
59. R.H. Good et al., "Field emission," in *Handbuch der Physik*, Springer, Vol. XXI, 1956.
60. C.A. Spindt et al., "Physical properties of thin-film field emission cathodes with molybdenum cones," *Journal of Applied Physics*, Vol. 47, No. 12, p. 5248, Dec. 1976.
61. H.F. Gray et al., "A vacuum field effect transistor using silicon field emitter arrays," *IEEE-IEDM Technical Digest*, p. 7776, 1986.
62. H.H. Busta, "Volcano-shaped field emitters for large area displays," *IEEE International Electron Device Meeting Technical Digest*, p. 405, 1995.
63. C.A. Spindt et al., "Field emission cathode array development for high-current-density applications," *Applications of Surface Science*, Vol. 16, p. 286, 1993.
64. A.I. Akinwande et al., "Nanometer scale thin-film-edge emitter devices with high current density characteristics," *1992 IEEE IEDM Technical Digest*, p. 367.
65. A.I. Akinwande et al., "Field-emission lamp for avionic AMLCD backlighting," *Technical Digest of International Symposium of the Society for Information Display*, p. 745, May 1996.
66. N. Kumar et al., "Development of nano-crystalline diamond-based field-emission displays," *Technical Digest of International Symposium of the Society for Information Display*, p. 43, May 1994.
67. D. Cathey, "Field emission displays," *Proceedings of the International Symposium on VLSI Technology, Systems and Applications*, pp. 131-136, 1995.
68. A. Palevsky et al., "Field emission displays: A 10,000-fL high-efficiency field-emission display," *Technical Digest of International Symposium of the Society for Information Display*, p. 55, May, 1995.
69. Presentations and Demonstrations by Silicon Video Corporation at the ARPA High Definition Systems Information Conference, Arlington, VA, April 15-18, 1996.
70. Presentations and Demonstrations by FED Corporation at the ARPA High Definition Systems Information Conference, Arlington, VA, April 15-18, 1996.
71. C.O. Bozler et al., "Arrays of gated field emitter cones having 0.32 μm tip-to-tip spacings," *J. Vac. Sci. Tech.*, Vol. B 12, p. 626, 1994.
72. T. Utsumi, "Keynote address vacuum microelectronics: What's new and exciting," *IEEE Transactions on Electron Devices*, Vol. 38, No. 10, p. 2276, Oct. 1991.
73. Y. Hori, et al., "Tower structure Si field emitter arrays with large emission current," *IEEE International Electron Device Meeting Technical Digest*, p. 393, 1995.
74. W.D. Kesling et al., "Field emission display resolution," *SID'93 Digest*, pp. 599-602, 1993.
75. W.D. Kesling et al., "Beam focusing for field-emission flat-panel displays," *IEEE Transactions on Electron Devices*, Vol. 42, No. 2, p. 340, Feb. 1995.
76. C.-M. Tang et al., "Theory and experiment of field-emitter arrays with planar lens focusing," *Eighth International Vacuum*

Microelectronics Conference, Portland Oregon, July 30-Aug. 3, 1995, p. 77.

77. Y. Toma, "Electron beam characteristics of double-gated Si field emitter arrays," *Eighth International Vacuum Microelectronics Conference*, Portland Oregon, July 30-Aug. 3, 1995, p. 9.

78. J. Itoh, et al., "Fabrication of double-gated Si-field emitter arrays for focused electron beam generation," *J. Vac. Sci. Technol.*, Vol. B 13, No. 5, p. 1968, Sept./Oct. 1995.



Kalluri Sarma received his Ph.D. degree in Materials Science from the University of Southern California, Los Angeles, in 1976. From 1977 to 1985 he was a Member of a Technical Staff at Motorola, Inc. During this time he performed research in silicon ribbon growth technologies, plasma processing, laser recrystallization of silicon, laser materials processing and solar cell technologies. He joined Honeywell Technology Center in 1986 as a Senior Research Staff Scientist. Since then he has been involved with the flat panel display research and development including a-Si, poly-Si and single crystal silicon TFT technologies, TFT-LCD design, AMLCD viewing angle enhancements, and display characterization. He is an author of over 30 technical publications, and holds 20 U.S. Patents. He served as a member of the SID Program Committee and Executive Committee. He is a member of SID, and IEEE.



Akintunde Ibiatyo (Tayo) Akinwande joined the Department of Electrical Engineering and Computer Science at MIT in January 1995 as an Associate Professor. His research interests are in the areas of vacuum microelectronics, flat panel displays, micromachining, and wide bandgap semiconductors.

Prior to joining MIT, Professor Akinwande was a Staff Scientist at the Honeywell Technology Center in Bloomington, MN. He led Field Emitter Display Development efforts at Honeywell. He pioneered the development of thin-film-edge field emitter arrays. While at Honeywell, he also worked on surface and bulk micromachining and the application to pressure sensors and accelerometers. His earlier work at Honeywell included the development of GaAs complementary heterostructure FET technology (a GaAs analog of Si CMOS),

and a 500 MHz DFT Processor based on E/D heterostructure FET technology.

He received the Sweatt Award—Honeywell's highest technical award in 1989 for his work on the DFT processor. He is also a recipient of the National Science Foundation Faculty Early Career Development (CAREER) Award.

He received the B.Sc. degree (First Class Honors) in Electrical and Electronic Engineering from the University of Ife, Nigeria, in 1978. He received his M.S. and Ph.D. both in Electrical Engineering from Stanford University, California in 1981 and 1986 respectively.

He has over 50 conference presentations and journal publications.

Threshold-Voltage Control Schemes through Substrate-Bias for Low-Power High-Speed CMOS LSI Design

TADAHIRO KURODA AND TAKAYASU SAKURAI
Semiconductor Device Engineering Laboratory, Toshiba Corp.

Received October 27, 1995; Revised March 26, 1996

Abstract. Lowering supply voltage, V_{DD} , is the most effective means to reduce power dissipation of CMOS LSI design. In low V_{DD} , however, circuit delay increases and chip performance degrades. There are two means to maintain the chip performance: 1) to lower the threshold voltage, V_{th} , to recover circuit speed, or 2) to introduce parallel and/or pipeline architectures to make up for slow device speed. The first approach increases standby power dissipation due to low V_{th} , while the second approach degrades worst case circuit speed caused by V_{th} fluctuation in low V_{DD} . This paper presents two circuit techniques to solve these problems, in both of which V_{th} is controlled through substrate bias. A Standby Power Reduction (SPR) scheme raises V_{th} in a standby mode by applying substrate bias with a voltage-switch circuit. A Self-Adjusting Threshold-Voltage (SAT) scheme reduces V_{th} fluctuation in an active mode by adjusting substrate bias with a feed-back control circuit. Test chips are fabricated and effectiveness of the circuit techniques is examined. The SPR scheme reduces 50% of the active power dissipation while maintaining the speed and the standby power dissipation. The SAT scheme improves worst case circuit speed by a factor of 3 under a 1 V V_{DD} .

1. Introduction

For many years a 5-volt power-supply was employed in digital equipment. During this period power dissipation of CMOS LSI chips such as digital signal processors and microprocessors increased as a whole four-fold every three years [1]. It was a foregone conclusion of the constant voltage scaling.

A 3.3-volt power-supply is recently used for submicron CMOS VLSI designs and lower voltages are studied for future ULSI designs [1–7]. Reduction of supply voltage has been primarily driven by two factors; reliability of gate oxides [2] and reduction of power dissipation [3–7]. The principle driver of this trend is emerging portable digital media such as Personal Digital Assistance (PDA) and digital communication. Chip power dissipation should be held down to milliwatt levels for battery life. Standby power dissipation should also be saved as much as possible. According to the forecast by the Semiconductor Industry Association (SIA) [8] supply voltage for battery-operated products is to be 0.9 volts (end-of-life battery voltage) by 2004.

Another motivation is a tight budget for consumer products such as a set-top box where an inexpensive plastic package is indispensable. Permitted limit of the chip power dissipation should be a little over 3 watts. Above the criterion, an expensive ceramic package is necessary. The SIA's roadmap predicts that the main stream of supply voltages for desktop products will be at 2.5 volts in 1998 and 1.5 volts in 2004.

Lowering supply voltage brings about a quadratic improvement in CMOS power dissipation, and therefore, is the most effective means. This simple solution to low-power design, however, comes at the cost of a speed penalty [3]. High-speed and low-power features are both required for portable multimedia equipment which delivers giga operations per second (GOPS) data processing performance for digital video use.

There are two different approaches to maintain the chip performance in low V_{DD} : 1) to lower the threshold voltage, V_{th} , to recover circuit speed [4, 5, 7], or 2) to introduce parallel and/or pipeline architectures to make up for slow device speed [3, 6]. The first approach increases standby power dissipation due to low V_{th} [9],

while the second approach degrades worst case circuit speed caused by V_{th} fluctuation in low V_{DD} [10].

The focus of this paper is to address these problems in low V_{DD} or low V_{th} , and to present solutions by circuit techniques. Two circuit schemes are presented to solve these problems, in both of which V_{th} is controlled through substrate bias, V_{BB} . A Standby Power Reduction (SPR) scheme [9] raises V_{th} and cut off leakage in a standby mode by applying deep V_{BB} with a voltage-switch circuit. A Self-Adjusting Threshold-Voltage (SAT) scheme [10] reduces V_{th} fluctuation in an active mode by adjusting V_{BB} with a feed-back control circuit.

The SPR scheme is presented in Section 2, and the SAT scheme is discussed in Section 3. In both sections, problems in low- V_{DD} or low- V_{th} circuit design are addressed, followed by descriptions of the proposed schemes, details in their circuit designs, and reports on experimental results. Section 5 is dedicated for conclusions.

2. Standby Power Reduction (SPR) Scheme

2.1. Problems

In order to understand circuit delay and power dissipation dependence on V_{DD} and V_{th} , a 2-input NAND

gate with fanout of 5 is simulated assuming a $0.3 \mu\text{m}$ CMOS technology. The fanout condition corresponds to the statistical average of gate loads in typical logic LSI designs. The same V_{th} is chosen for $n\text{MOS}$ and $p\text{MOS}$. Gate width of all the MOSFETs is $10 \mu\text{m}$.

Figures 1(a)–(c) show the simulation results by SPICE. Delay contour lines are projected on the V_{DD} - V_{th} plane in Fig. 1(a). If V_{th} is reduced to 0.3 V , V_{DD} can be lowered down to 2 V while maintaining the speed at $V_{th} = 0.7 \text{ V}$ and $V_{DD} = 3 \text{ V}$, typical operation condition for high-speed LSI design. Figure 1(b) shows a simulated power dependence on V_{DD} and V_{th} . The power includes subthreshold leakage current, crowbar current, and CV^2 component. At $V_{DD} = 3 \text{ V}$ and $V_{th} = 0.7 \text{ V}$ the power dissipation is 0.107 mW , while at $V_{DD} = 2 \text{ V}$ and $V_{th} = 0.3 \text{ V}$ it is reduced to 0.048 mW . This corresponds to the active power reduction of more than 50% while maintaining the speed. The energy-delay (ED) product is also plotted in Fig. 1(c) as a function of V_{DD} and V_{th} . Reducing the ED product is a good direction for optimizing LSI design for portable use, since the ED product reflects the battery consumption (E) for completing a job in a certain time (D) [5]. The ED product is also reduced from $5.18 \times 10^{-24} \text{ J} \cdot \text{s}$ to $2.32 \times 10^{-24} \text{ J} \cdot \text{s}$.

The only drawback of choosing 0.3 V V_{th} is the increase in standby power dissipation. In order to solve

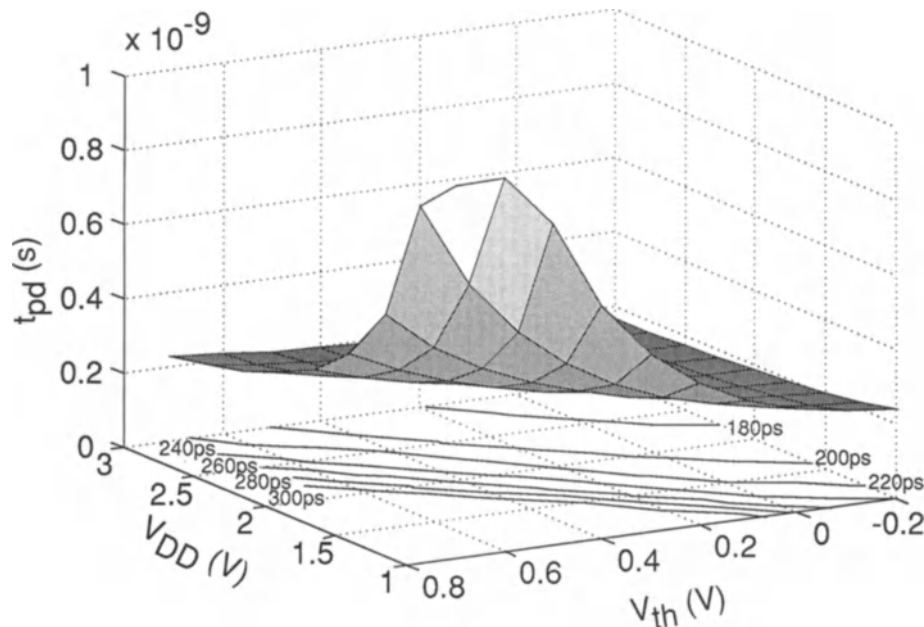


Figure 1a. Simulated delay dependence on V_{DD} and V_{th} by SPICE. V_{th} signifies the absolute value of the threshold voltage of MOSFETs. Same V_{th} is set for $n\text{MOS}$ and $p\text{MOS}$.

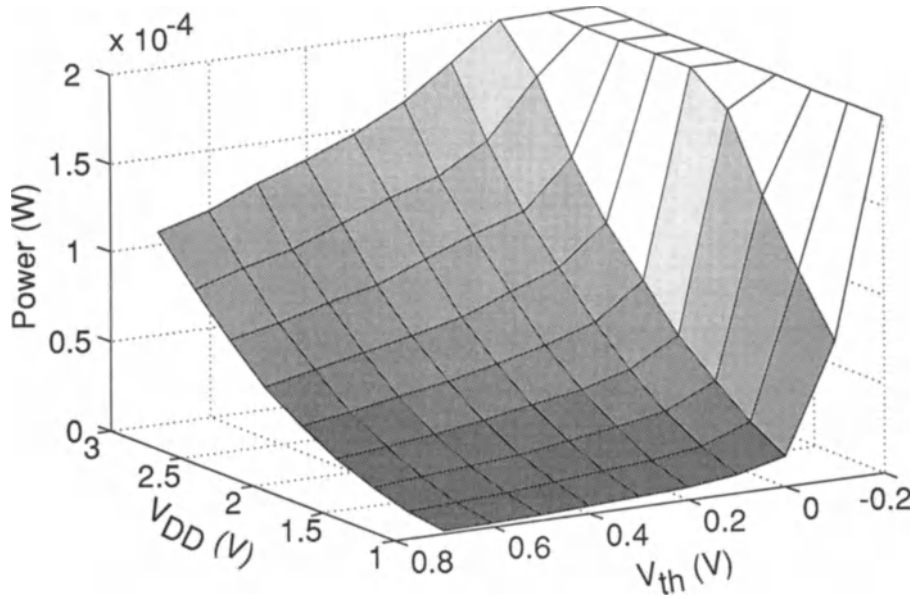


Figure 1b. Simulated power dissipation dependence on V_{DD} and V_{th} by SPICE. Activation ratio of 0.3 and cycle time of $30 \times t_{pd}$ are assumed. The power includes subthreshold leakage current, crowbar current, and CV^2 component.

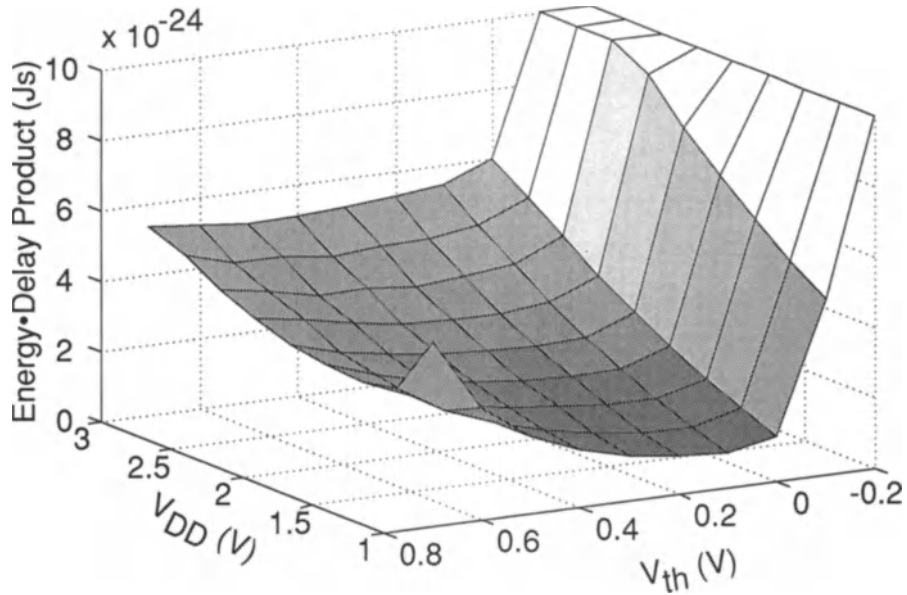


Figure 1c. Simulated dependence of energy-delay (ED) product on V_{DD} and V_{th} by SPICE.

the standby power problem, multithreshold-voltage CMOS technology was proposed [11] where two different V_{th} MOSFETs were employed; low V_{th} for fast circuit operation and high V_{th} for providing and cutting internal supply voltage. This scheme, however, requires very large transistors for the internal power-supply control to impose area and yield penalties, otherwise degrading circuit speed. Furthermore it cannot be applied to memory elements.

A new standby power reduction scheme is proposed in the next section which dynamically changes V_{th} in the active and standby mode by applying substrate bias. This scheme does not impose those penalties in area and speed, nor the limitation in usage.

2.2. Circuit Design

The main idea of this standby power reduction (SPR) scheme is that substrate bias is applied in the standby mode to increase V_{th} and cut off leakage current, while in the active mode the substrate bias is not applied to assure high-speed operation. Figure 2 shows a measured I_{DS} - V_{GS} characteristics of the $0.3 \mu\text{m}$ $n\text{MOS}$ transistor. By applying V_{BB} of -2 V , V_{th} can be increased by

0.4 V . This means that if V_{BB} of -2 V is applied in the standby mode, V_{th} is increased from 0.3 V to 0.7 V , and thus realizes the same standby current as the design in $V_{th} = 0.7 \text{ V}$.

Figure 3 depicts a circuit diagram of the proposed SPR scheme. Figure 4 shows the simulated waveforms of the circuit. The circuit consists of a level-shifting part and a voltage-switching part. When chip enable, CE, is asserted in the active mode, the n -well bias, V_{NWELL} , becomes equal to V_{DD} ($= 2 \text{ V}$), and the p -well bias, V_{PWELL} , becomes V_{SS} . When CE is disabled in the standby mode, V_{NWELL} equals V_{NBB} which is set at 4 V , and V_{PWELL} becomes V_{PBB} , which is -2 V . A standby-to-active mode transition and an active-to-standby mode transition take less than 100 ns . Power dissipation of the SPR circuit in the standby mode is less than $0.1 \mu\text{A}$. V_{NBB} , V_{PBB} , V_{DD} and V_{SS} are applied from external sources, but power supply lines for V_{NBB} and V_{PBB} need to supply only $0.1 \mu\text{A}$ or less current. The diodes in the circuit are built using a junction-well structure through which current flows only in the active mode.

In designing the circuit, care is taken so that no transistor sees high-voltage stress of gate oxide and junctions. Figure 5(a) shows V_{GS} - V_{GD} trajectories of

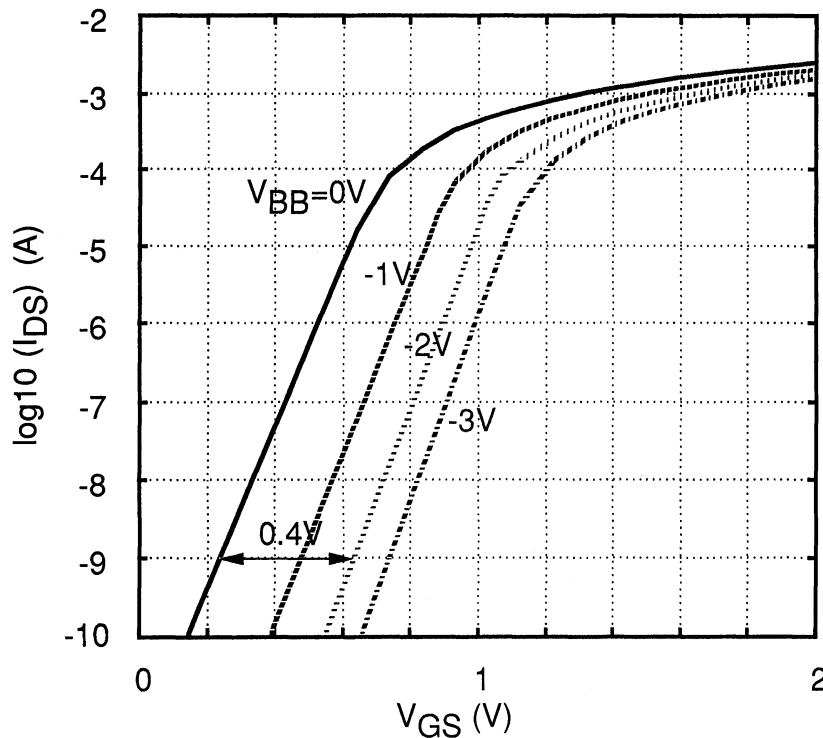


Figure 2. Measured I_{DS} - V_{GS} characteristics of $0.65 \mu\text{m}$ $n\text{MOS}$ transistor. By applying V_{BB} of -2 V , V_{th} can be increased by 0.4 V .

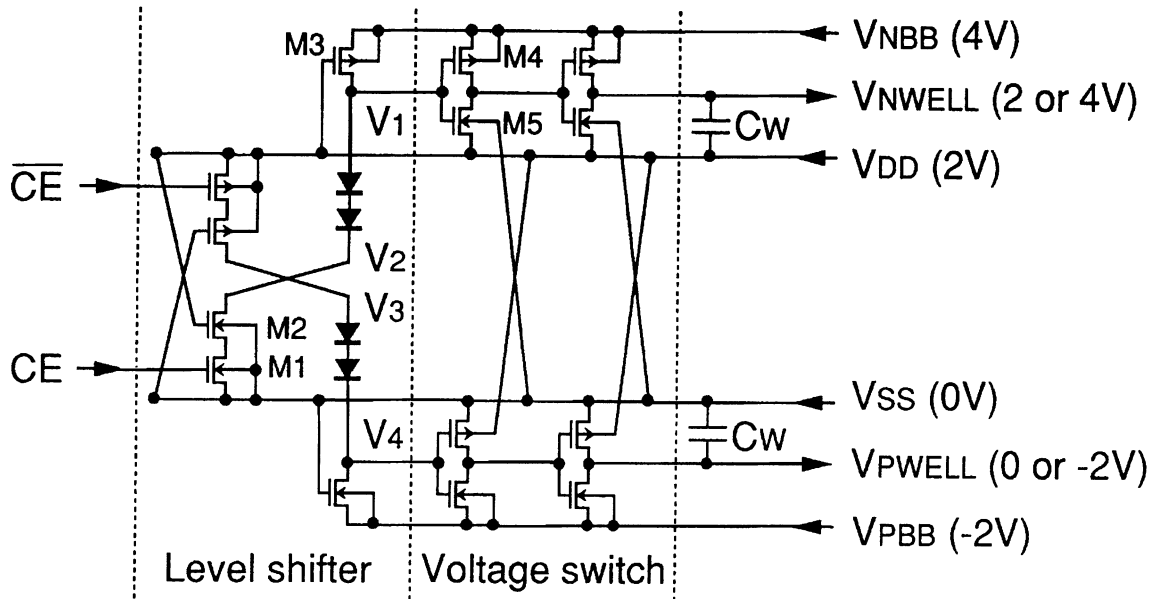


Figure 3. Circuit diagram of the SPR circuit. Well capacitance, C_w , is assumed to be 1000 pF.

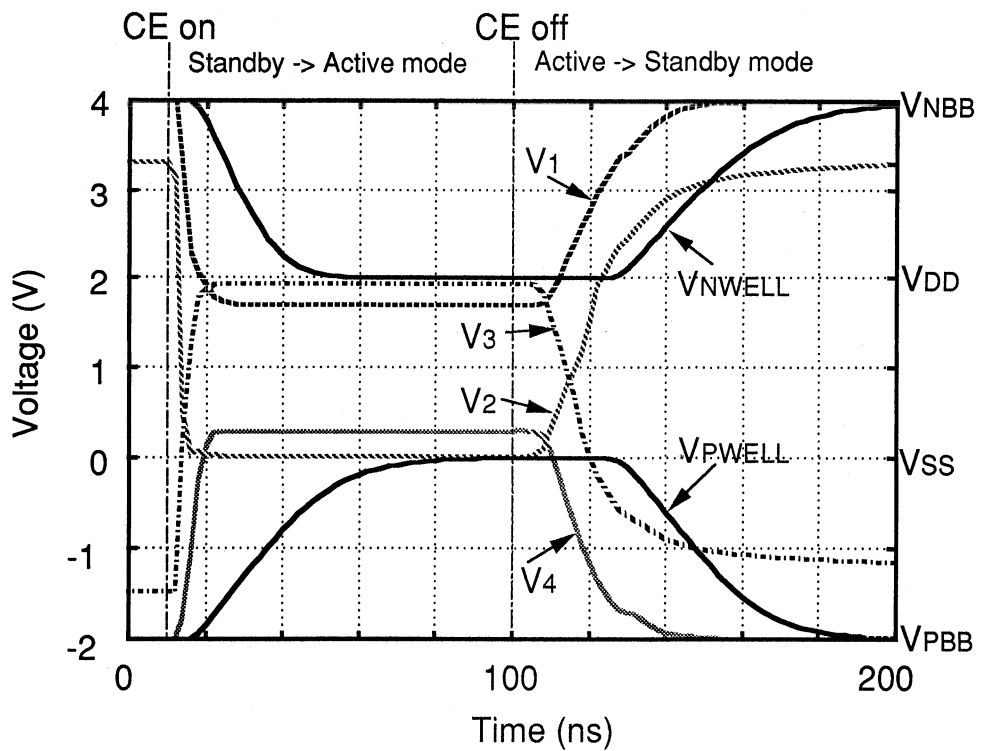


Figure 4. Simulated waveforms of the SPR circuit. Standby-to-active mode transition and active-to-standby mode transition take less than 100 ns.

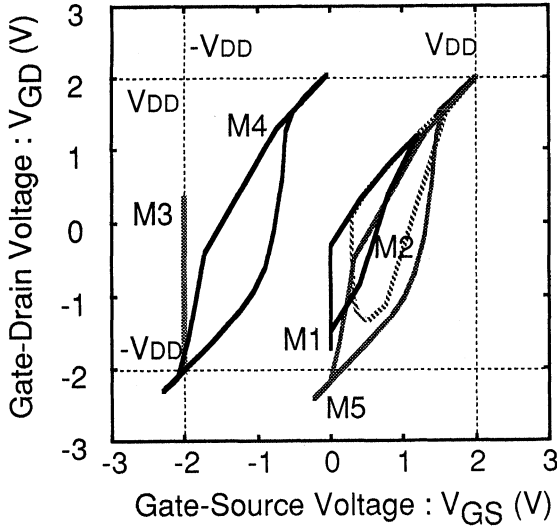


Figure 5a. V_{GS} - V_{GD} trajectories of MOSFETs used in the SPR circuit. The trajectory does not go beyond $\pm(V_{DD} + \alpha)$. Notations from M1 through M5 correspond to transistor names in Fig. 3.

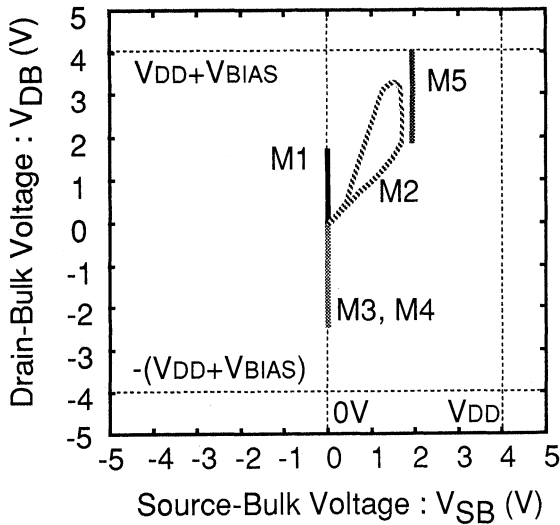


Figure 5b. V_{SB} - V_{DB} trajectories of MOSFETs used in the SPR circuit. The trajectory does not go beyond $\pm(V_{DD} + V_{BIAS})$. Notations from M1 through M5 correspond to transistor names in Fig. 3.

MOSFETs used in the SPR circuit. The trajectories do not go beyond $\pm(V_{DD} + \alpha)$, which assures sufficient reliability of gate oxide. Figure 5(b) depicts V_{SB} - V_{DB} trajectories of MOSFETs used in the SPR circuit, where V_{SB} represents the source-bulk voltage and V_{DB} represents the drain-bulk voltage. The trajectories do not go beyond $\pm(V_{DD} + V_{BIAS})$, where V_{BIAS} signifies the larger voltage of $|V_{NBB} - V_{DD}|$ and $|V_{SS} - V_{PBB}|$. This voltage is applied to junctions, but the junction

breakdown voltage of the $0.3 \mu\text{m}$ MOSFETs is more than 9 V, and hence, junction breakdown does not occur for any MOSFET.

2.3. Experimental Results

Figure 6 shows a micrograph of a test chip. A ring oscillator constructed with 49 stages of 2-input NAND gates and the SPR circuit are implemented using the $0.3 \mu\text{m}$ process technology. The SPR circuit occupies $2500 \mu\text{m}^2$ for either n -well or p -well bias circuit. In cases where n MOS circuit mostly determines the speed as in n MOS pass transistor logic, only V_{th} for n MOS should be lowered and hence only p -well bias circuit is needed. If both of the n -well and p -well bias circuits are required as in Fig. 3, $5000 \mu\text{m}^2$ Si area is occupied and a triple-well technology is to be used. The standby current of less than $0.1 \mu\text{A}$ is measured on the test chip in the standby mode. In the active mode the standby current is measured larger by three orders. The speed of the 2-input NAND gate of 300 ps is achieved at $V_{DD} = 2$ V. Setting time of the substrate bias is less than 100 ns.

The proposed SPR scheme is fully compatible with the existing CAD tools including automatic placement and routers. As for standard cell library, cell layouts should be modified to separate substrate bias lines and power supply lines. The substrate bias lines, however, can be as narrow as possible and can be scaled. The area overhead to the total chip is estimated to be less than 5%.

3. Self-Adjusting Threshold-Voltage (SAT) Scheme

3.1. Problems

Figure 7 shows how much the V_{th} fluctuation affects gate propagation delay, t_{pd} , in various V_{DD} regime. Alpha-power law MOSFET model [12] is used to estimate t_{pd} whose expression is written as follows:

$$t_{pd} = \frac{C_L V_{DD}}{(V_{DD} - V_{th})^\alpha} \left[\left(\frac{1}{2} - \frac{1 - V_{th}/V_{DD}}{1 + \alpha} \right) \times \left(\frac{0.9}{0.8} + \frac{V_{D0}}{0.8 V_{DD}} \ln \frac{10 V_{D0}}{e V_{DD}} \right) + \frac{1}{2} \right],$$

where V_{D0} is a drain saturation voltage and C_L is a load capacitance. Typical parameter values that $\alpha = 1.3$ and $V_{D0}/V_{DD} = 0.5$ are employed in this estimation.

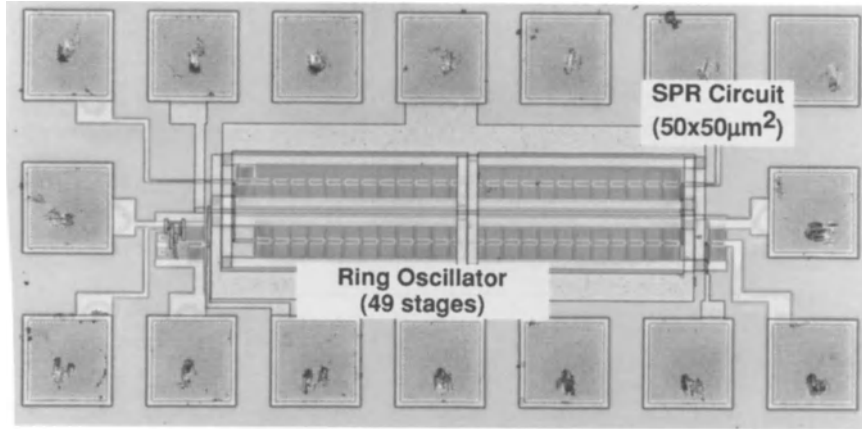


Figure 6. Micrograph of SPR test chip. The SPR circuit occupies $2500 \mu\text{m}^2$ for either n -well or p -well bias circuit.

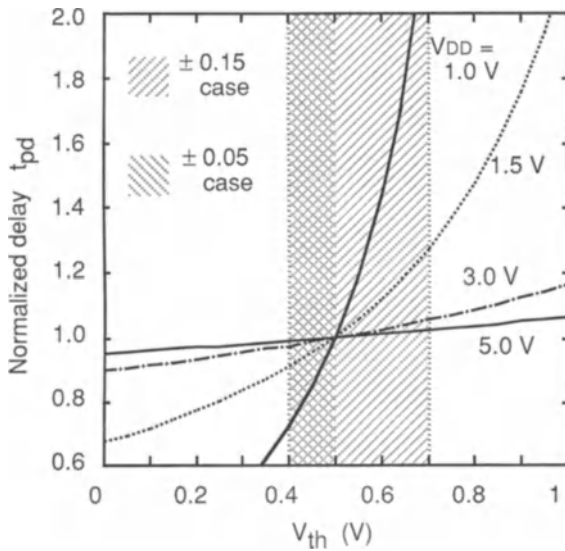


Figure 7. Calculated t_{pd} dependence on V_{th} for various V_{DD} .

The minimum V_{th} in the distribution is determined by a total leakage current of a chip. If the V_{th} is too low, the power dissipation of a chip surmounts the specified maximum power dissipation. For example, if it is specified that 0.4 V is the minimum V_{th} and the V_{th} fluctuation is ± 0.15 V, then the worst chips show the threshold voltage of 0.7 V. In order to achieve high performance yield, the speed of these worst chips determines the speed specification of the product. The V_{th} distribution of this case is suggested by a hatched region in Fig. 7.

On the other hand, if the V_{th} fluctuation can be reduced to ± 0.05 V, the worst V_{th} becomes 0.5 V.

The situation is indicated by another hatched region in Fig. 7. The speed difference of the worst cases is a factor of 1.3 at 1.5 V V_{DD} and a factor of 3 at 1 V V_{DD} .

3.2. Circuit Design

Figure 8 illustrates a block diagram of the proposed Self-Adjusting Threshold-Voltage (SAT) scheme to reduce the V_{th} fluctuation down to ± 0.05 V. A leakage sensor senses leakage current of a representative MOSFET and outputs a control signal, V_{cont} , to Self-Substrate-Bias (SSB) circuit. V_{cont} is controlled so that it triggers the SSB only when the leakage is higher than a certain preset level. Suppose an n MOS case (a p MOS case is conceptually the same). The SSB, when triggered, draws charge from p -wells to lower substrate bias, V_{BB} . The lowered V_{BB} in turn increases V_{th} of n MOS and lowers leakage current. The V_{BB} is distributed to all the p -wells on a chip.

Thus, V_{th} is controlled to make the leakage current equal to the specified value, that is, V_{th} is set to the lowest possible value that satisfies the power specification. Consequently, the speed is optimized. Substrate bias is also good for reducing junction capacitance to further improve the performance of a chip. Process target of V_{th} should be low enough so that SSB can tune V_{th} to whatever value that is necessary. V_{th} of p MOS can be controlled in the same way at the same time.

Figure 9(a) shows a measured subthreshold I_{DS} - V_{GS} characteristics of an n MOS and Fig. 9(b) shows a measured I_{DS} dependence on V_{th} . The I_{DS} can be written as

$$I_{DS} \propto \exp\{(V_{GS} - V_{th})/s\},$$

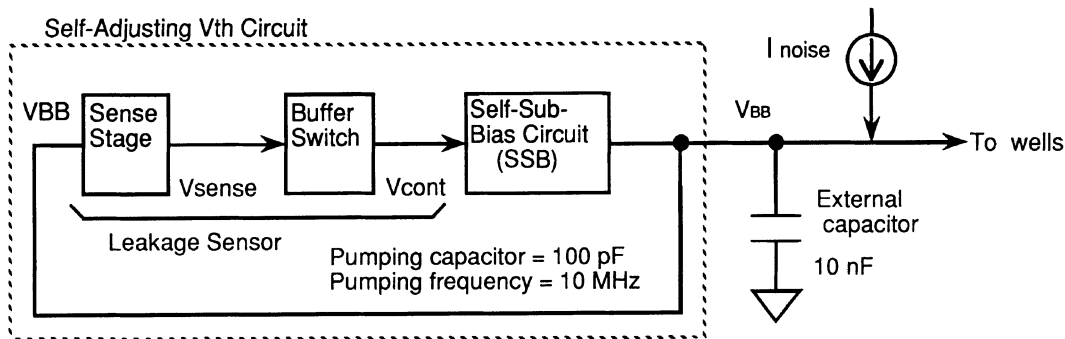


Figure 8. Block diagram of Self-Adjusting Threshold-Voltage (SAT) scheme.

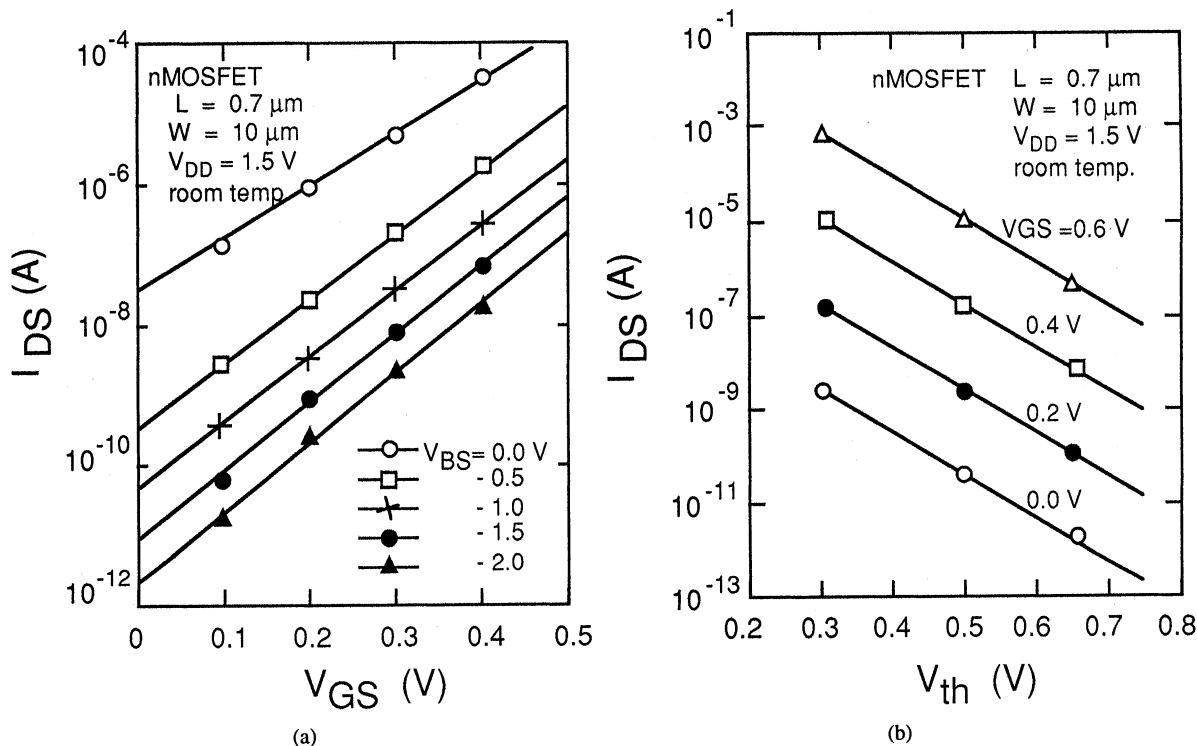


Figure 9. (a) Measured sub-threshold I_{DS} - V_{GS} characteristics for various V_{BS} . (b) Measured sub-threshold I_{DS} - V_{th} characteristics for various V_{GS} .

where s is called the s -factor and is about 110 mV/decade when V_{BS} is zero and becomes 90 mV/decade when V_{BS} is less than -1 V . This suggests that with the substrate bias, V_{th} can be set lower than in the case without the substrate bias with maintaining the leakage current the same.

Figure 10 is a circuit diagram of the leakage sensor. The leakage current through the representative MOSFET M1 can vary by a factor of as much as 10 when V_{th} is changed by only 0.1 V because of the

exponential dependence of the leakage current on V_{th} . The leakage current is amplified by the load L . The load can be either a resistor made by a well diffusion of about $1 \text{ M}\Omega$ or $p\text{MOS}$ whose process and environmental (temperature and V_{DD}) variation is within a factor of 3. This corresponds to V_{th} controllability of $\pm 0.02 \text{ V}$.

V_G is generated by dividing V_{DD} and is set around 0.2 V. This finite V_G is necessary to enhance the leakage current and shortens the dynamic delay of the sensor.

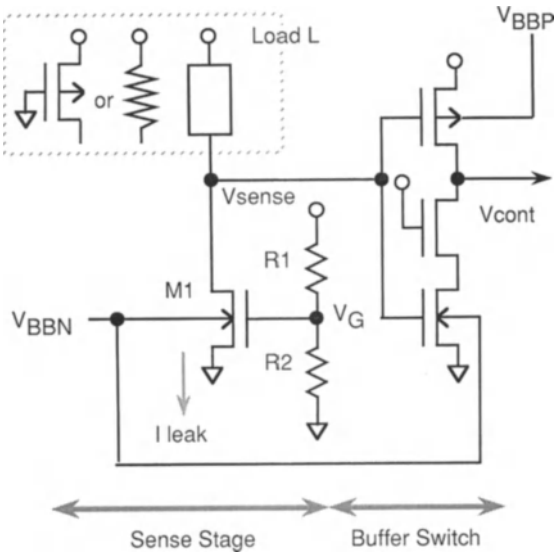


Figure 10. Leakage sensor in SAT scheme.

Since V_{DD} can be controlled within $\pm 5\%$, the fluctuation of V_G is ± 0.01 V. This value should be added to ± 0.02 V mentioned above, totaling ± 0.03 V of static controllability. The fluctuation of the switch buffer gives negligible effects to the controllability.

Figure 11 depicts dynamic behavior of the circuit. A large capacitor of 10 nF is connected to V_{BB} in external of a chip. The current noise is assumed to be 1 mA which corresponds to hot-carrier current generated by 10 A of drain channel current. The delay of the sensor introduces dynamic controllability which is additive to the static controllability. Overall V_{th} controllability including static and dynamic effects is ± 0.05 V.

3.3. Experimental Results

A test chip is fabricated by a standard 0.7 μm CMOS process, whose micro-photograph is shown in Fig. 12. The size is 1.5 mm \times 0.7 mm including SSB and the leakage sensor. The chip includes only a p -well bias generator for controlling V_{th} of $n\text{MOS}$ so that the size should be doubled if threshold voltages of both $p\text{MOS}$ and $n\text{MOS}$ are to be controlled. The implemented SSB circuit employs a conventional configuration. Figure 13 shows a measured V_{th} static controllability which is shown to be less than ± 0.025 V.

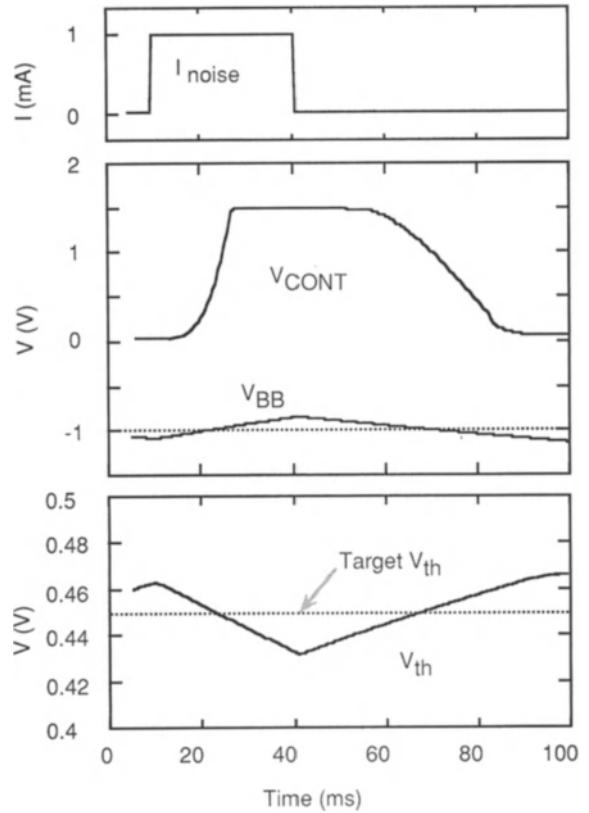


Figure 11. Simulated waveforms of V_{BB} and V_{th} in SAT scheme.

4. Conclusions

Two circuit techniques have been studied for low-power high-speed CMOS LSI design, in both of which V_{th} is controlled through substrate bias. The Standby Power Reduction (SPR) scheme raises V_{th} in the standby mode by applying substrate bias. It reduces 50% of the active power dissipation while maintaining the speed and the standby power dissipation. The Self-Adjusting Threshold-Voltage (SAT) scheme reduces V_{th} fluctuation in the active mode by adjusting substrate bias. It improves worst case circuit speed by a factor of 3 under a 1 V V_{DD} .

The SPR scheme is mainly used for low V_{DD} and low V_{th} design for portable use, while the SAT scheme is primarily employed for low V_{DD} and standard V_{th} design. The two schemes, therefore, can take different approaches; in the SPR the substrate bias is applied from external sources with the voltage switch circuit, while in the SAT the substrate bias is generated internally with the SSB circuit. If V_{DD} is reduced below 1 V,

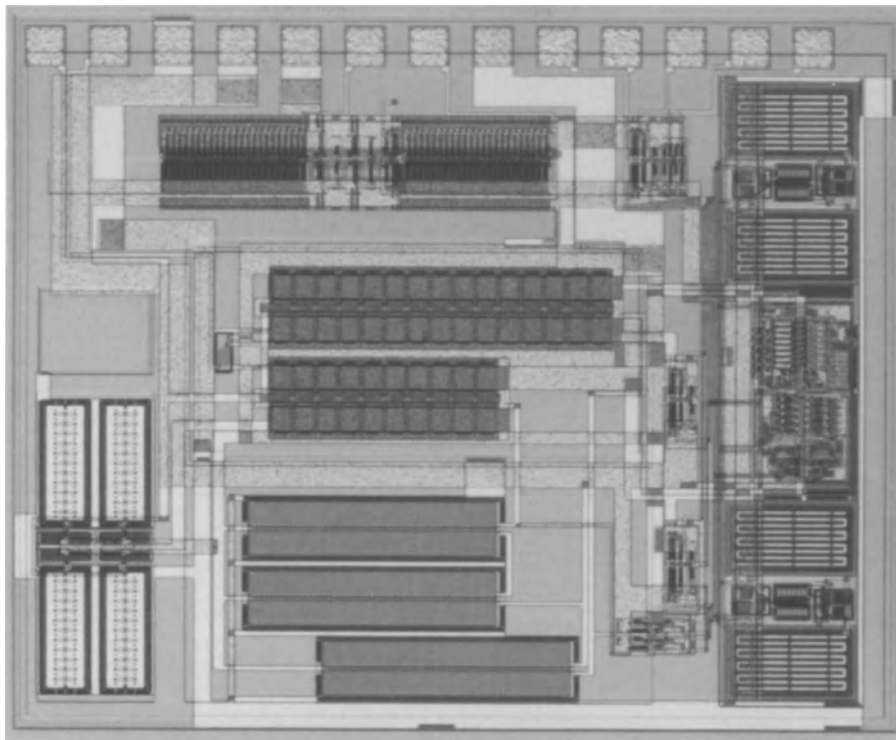


Figure 12. Micrograph of SAT test chip.

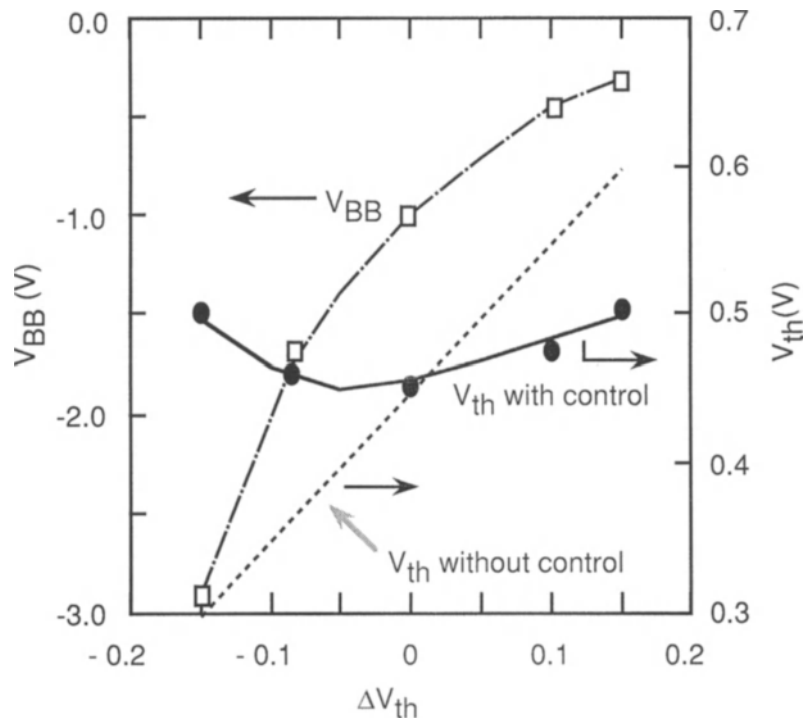


Figure 13. Measured V_{th} controllability dependence on process fluctuation ΔV_{th} .

however, the speed variation due to the V_{th} fluctuation cannot be ignored even in low V_{th} . A unified scheme which can solve the two problems in a unified way will be required in the future and should be studied.

Acknowledgment

The authors would like to acknowledge the encouragement of J. Iwamura, O. Ozawa, and Y. Unno throughout the work. Assistance provided by T. Kobayashi, H. Hara, K. Seta in circuit design, and by M. Kakumu in test chip fabrication is also appreciated.

References

1. T. Kuroda and T. Sakurai, "Overview of low-power ULSI circuit techniques," *IEICE Trans. Electron.*, Vol. E78-C, No. 4, pp. 334–344, April 1995.
2. M. Kakumu and M. Kinugawa, "Power supply voltage impact on circuit performance for half and lower submicrometer CMOS LSI," *IEEE Trans. Electron Devices*, Vol. 37, No. 8, pp. 1902–1908, Aug. 1990.
3. A.P. Chandrakasan, S. Sheng, and R.W. Brodersen, "Low-power CMOS digital design," *IEEE J. Solid-State Circuits*, Vol. 27, No. 4, pp. 473–484, April 1992.
4. D. Liu and C. Svensson, "Trading speed for low power by choice of supply and threshold voltages," *IEEE J. Solid-State Circuits*, Vol. 28, No. 1, pp. 10–17, Jan. 1993.
5. J.B. Burr and J. Scott, "A 200 mV self-testing encoder/decoder using stanford ultra low power CMOS," in *ISSCC Dig. Tech. Papers*, pp. 84–85, Feb. 1994.
6. A. Chandrakasan, A. Burstein, and R.W. Brodersen, "A low power chipset for portable multimedia applications," in *ISSCC Dig. Tech. Papers*, pp. 82–83, Feb. 1994.
7. M. Izumikawa, H. Igura, K. Furuta, H. Ito, H. Wakabayashi, K. Nakajima, T. Mogami, T. Horiuchi, and M. Yamashina, "A 0.9 V 100 MHz 4 mW 2 mm² 16b DSP core," in *ISSCC Dig. Tech. Papers*, pp. 84–85, Feb. 1995.
8. The Semiconductor Industry Association (SIA), "The national technology roadmap for semiconductors," 1994 revision.
9. K. Seta, H. Hara, T. Kuroda, M. Kakumu, and T. Sakurai, "50% active-power saving without speed degradation using standby power reduction (SPR) circuit," in *ISSCC Dig. Tech. Papers*, pp. 318–319, Feb. 1995.
10. T. Kobayashi and T. Sakurai, "Self-adjusting threshold-voltage scheme (SATS) for low-voltage high-speed operation," in *Proc. of IEEE CICC'94*, pp. 271–274, May 1994.
11. S. Mutoh, T. Douseki, Y. Matsuya, T. Aoki, S. Shigematsu, and J. Yamada, "1-V power supply high-speed digital circuit technology with multithreshold-voltage CMOS," *IEEE J. Solid-State Circuits*, Vol. 30, No. 8, pp. 847–854, Aug. 1995.
12. T. Sakurai and A.R. Newton, "Alpha-power law MOSFET model and its application to CMOS inverter delay and other formulas," *IEEE J. Solid-State Circuits*, Vol. 25, No. 2, pp. 584–594, April 1990.



Tadahiro Kuroda was born in Mie, Japan on February 20, 1959. He received the B.S. degree in electronic engineering from the University of Tokyo, Tokyo, Japan, in 1982.

In 1982 he joined Toshiba Corporation, Japan, where he was engaged in the development of CMOS design rules, and CMOS gate arrays and standard cells. From 1988 to 1990 he was a visiting scholar at the University of California, Berkeley, doing research in the field of computer-aided design of VLSI's. In 1990 he was back at Toshiba and involved in the development of BiCMOS ASIC's and ECL gate arrays. Since 1993 he has been with Semiconductor Device Engineering Laboratory at Toshiba, where he has been engaged in the research and development of multimedia CMOS LSI's. His research interests include high-speed, low-power, low-voltage circuit techniques in CMOS, BiCMOS, and ECL technologies.

Mr. Kuroda is serving as a program committee member for the Symposium on VLSI Circuits. He is a member of the IEEE and the Institute of Electronics, Information and Communication Engineers of Japan.



Takayasu Sakurai was born in Tokyo, Japan in 1954. He received the B.S., M.S. and Ph.D degrees in electronic engineering from University of Tokyo, Tokyo, Japan, in 1976, 1978, and 1981, respectively.

In 1981 he joined the Semiconductor Device Engineering Laboratory, Toshiba Corporation, Japan, where he was engaged in the research and development of CMOS dynamic RAM and 64 Kbit, 256 kbit SRAM, 1 Mbit virtual SRAM, cache memories, and BiCMOS ASIC's. During the development, he also worked on the modeling of interconnect capacitance and delay, new memory architectures, hot-carrier resistant circuits, arbiter optimization, gate-level delay modeling, alpha/nth power MOS model and transistor network synthesis. From 1988 through 1990, he was a visiting scholar at Univ. of Calif., Berkeley, doing research in the field of VLSI CAD. He is currently back in Toshiba and managing multimedia LSI development. His present activities include low-power designs, media processors and video compression/decompression LSI's.

Dr. Sakurai is a visiting lecturer at Tokyo University and serving as a program committee member of CICC, DAC, ICCAD, ICVC, ISPLED and FPGA Workshop. He is a technical committee chairperson for the '97 VLSI Circuits Symposium. He is a member of the IEEE, the IEICEJ and the Japan Society of Applied Physics.

Processor Design for Portable Systems

THOMAS D. BURD AND ROBERT W. BRODERSEN
Department of EECS, University of California at Berkeley

Received January 24, 1996; Revised April 3, 1996

Abstract. Processors used in portable systems must provide highly energy-efficient operation, due to the importance of battery weight and size, without compromising high performance when the user requires it. The user-dependent modes of operation of a processor in portable systems are described and separate metrics for energy efficiency for each of them are found to be required. A variety of well known low-power techniques are re-evaluated against these metrics and in some cases are not found to be appropriate leading to a set of energy-efficient design principles. Also, the importance of idle energy reduction and the joint optimization of hardware and software will be examined for achieving the ultimate in low-energy, high-performance design.

1. Introduction

The recent explosive growth in portable electronics requires energy conscious design, without sacrificing performance. Simply increasing the battery capacity is not sufficient because the battery has become a significant fraction of the total device volume and weight [1–3]. Thus, it has become imperative to minimize the load on the battery while simultaneously increasing the speed of computation to handle ever more demanding tasks.

One successful approach is to move the data processing (e.g., handwriting and speech recognition, data encoding and decoding, etc.) to specialized DSP integrated circuits, which can achieve orders of magnitude improvement in energy efficiency [4]. However, there exists a large amount of control processing (e.g., operating system, network control, peripheral control, etc.) that cannot be suitably implemented on dedicated architectures. New approaches to energy-efficient design are required for reducing the energy consumption of programmable processors that are needed for this kind of processing.

There are three main elements to energy-efficient processor design. First, the best available technology should be used, since energy efficiency improves quadratically with technology [5]. Next, design for energy efficiency should be considered from the outset,

rather than retrofitting an existing design. In doing so, the portable device must be optimized as a complete system, rather than optimizing individual components. Finally, energy-efficient design techniques should be aggressively utilized in the processor implementation.

A framework for an energy-efficient design methodology suitable for a processor used in a portable environment will be presented. First, the unique demands on a processor in this environment will be examined. Starting from simple analytic models for delay and power in CMOS circuits, metrics of energy efficiency for the processor will be quantified. These metrics will then be applied to develop four important principles of energy-efficient processor design. This paper will conclude with a survey of both hardware and software techniques, and their relative impact on processor energy efficiency.

2. Operation in a Portable Environment

Since portable devices (e.g., notebook computers, PDAs, cellular phones, etc.) are single-user systems, their usage is typically bursty; that is, the useful computation is interleaved with periods of idle time. Also, portable devices demand minimum energy consumption to maximize battery life, whereas desktop computers require minimum power dissipation to minimize

heat dissipation. The growing single-user “green” PC movement, however, has the same demands on performance and energy consumption as portable devices since it is total energy which is being limited.

2.1. Processor Usage Model

Understanding a processor’s usage pattern in portable applications is essential to its optimization. Processor utilization can be evaluated as the amount of processing required and the allowable latency before its completion. These two parameters can be merged into a single measure. Throughput, T , is defined as the number of operations that can be performed in a given time:

$$\text{Throughput} \equiv T = \frac{\text{Operations}}{\text{Second}} \quad (1)$$

Operations are defined as the basic unit of computation and can be as fine-grained as instructions or more coarse-grained as programs. This leads to measures of throughput of MIPS (instructions/sec) and SPECint92 (programs/sec) which compare the throughput on implementations of the same instruction set architecture (ISA), or different ISAs, respectively.

Figure 1 plots a sample usage pattern which shows that the desired throughput falls into one of three categories. Compute-intensive and minimum-latency processes (e.g., spreadsheet update, spell-check, scientific computation, etc.) desire maximum performance, which is limited by the peak throughput of the processor, T_{MAX} . Background and relatively high-latency processes (e.g., screen update, low-bandwidth I/O control, data entry, etc.) do not desire the full throughput of the processor, but just a fraction of it as there is no intrinsic benefit to exceeding their latency requirements.

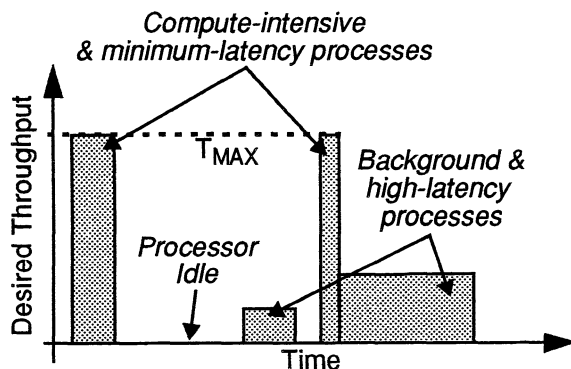


Figure 1. Processor usage model.

When there are no active processes, the processor idles and has zero desired throughput.

2.2. What Should Be Optimized?

Any increase in T_{MAX} can be readily employed by compute-intensive and minimum-latency processes. In contrast, the background and high-latency processes do not benefit from any increase in T_{MAX} above and beyond their average desired throughput since the extra throughput cannot be utilized. Peak throughput is the variable to be maximized since the average throughput is determined by the user.

To maximize the amount of computation delivered per battery life, the energy consumed per operation should be minimized. Only the time it takes to completely discharge the battery is of interest, and not the incremental rate of energy consumption (i.e., power). Thus, the average energy consumed per operation should be minimized, rather than the instantaneous energy consumption.

These are the optimizations to target in energy-efficient processor design: maximize peak deliverable throughput, and minimize average energy consumed per operation. Metrics will be developed in Section 4 to quantitatively measure energy efficiency.

2.3. InfoPad: A Portable System Case Study

The InfoPad is a wireless, multimedia terminal that fits a compact, low-power package in which as much as possible of the processing has been moved onto the backbone network [4]. An RF modem sends/receives data to/from five I/O ports: video input, text/graphics input, pen output, audio input, and audio output. Each I/O port consists of specialized digital ICs, and the I/O device (e.g., LCD, speaker, etc.). In addition, there is an embedded processor subsystem used for data flow and network control. InfoPad is an interesting case study because it contains large amounts of data processing and control processing, which require different optimizations for energy efficiency.

The specialized ICs include a video decompression chip-set which decodes 128×240 pixel frames in real-time, at 30 frames/second. The collection of four chips takes in vector quantized data and outputs analog RGB directly to the LCD and dissipates less than 2 mW. Implementing the same decompression in a general purpose processor would require a throughput of around 10 MIPS with hand-optimized code. A

processor subsystem designed with the best available parts would dissipate at least 200 mW. This provides a prime example of how dedicated architectures can radically exploit the inherent parallelism of signal processing functions to achieve orders of magnitude reduction of power dissipation over equivalent general-purpose processor-based systems.

The control processing, which has little parallelism to exploit, is much better suited towards a general purpose processor. An embedded processor system was designed around the ARM60 processor, which combined with SRAM and external glue logic dissipates 1.2 W, while delivering a peak throughput of 10 MIPS. It is this discrepancy of almost 3 orders of magnitude in power dissipation, which can result in the processor sub-system dominating the total energy consumption, that leads to the current objective of substantially reducing the processor energy consumption.

2.4. The System Perspective

In an embedded processor system such as that found in InfoPad, there are essential digital ICs external to the processor required for a functional system: main memory, clock oscillator, I/O interface(s), and system control logic (e.g., PLD). Integrated solutions have been developed for embedded applications that move the system control logic, the oscillator, and even the I/O interface(s) onto the processor chip leaving only the main memory external.

Figure 2 shows a schematic of the InfoPad processor subsystem, which contains the essential system components described above. Interestingly, the processor does not dominate the system's power dissipation; rather, the SRAM memory dissipates half the power. For aggres-

sive low-power design, it is imperative to optimize the entire system and not just a single component; optimizing just the processor in the InfoPad system can yield at most a 10% reduction in power. While the work presented here focuses just on processor design, in the implementation of a complete system it is important to optimize all components as necessary.

High-level processor and system simulation is generally used to verify the functionality of an implementation and find potential performance bottlenecks. Unfortunately, such high-level simulation tools do not exist for energy consumption with the result that simulations to extract energy consumption are typically not made until the design has reached the logic design level. At this time it is very expensive to make significant changes, with the result that it is difficult to make system optimizations for energy consumption through redesign or repartitioning. Only recently has this issue been addressed [6].

It is important to understand how design optimizations in one part of a system may have detrimental effects elsewhere. An example is the effect a processor's on-chip cache has on the external memory system. Because smaller memories have lower energy consumption, the designer may try to minimize the on-chip cache size to minimize the energy consumption of the processor at the expense of a small decrease in throughput (due to increased miss rates of the cache). The increased miss rates affect not only the performance, however, but may increase the system energy consumption as well because high-energy main memory accesses are now made more frequently. So, even though the processor's energy consumption was decreased, the total system's energy consumption has increased.

3. CMOS Circuit Models

Power dissipation and circuit delays for CMOS circuits can be accurately modeled with simple equations, even for complex processor circuits. These models, along with knowledge about the system architecture, can be used to derive analytical models for energy consumed per operation and peak throughput.

These models will be presented in this section and then used in Section 4 to derive metrics that quantify energy efficiency. With these metrics, the circuit and system design can be optimized for maximum energy efficiency. These models hold only for digital CMOS circuits, and are not applicable to bipolar or BiCMOS circuits. However, this is not very limiting since CMOS

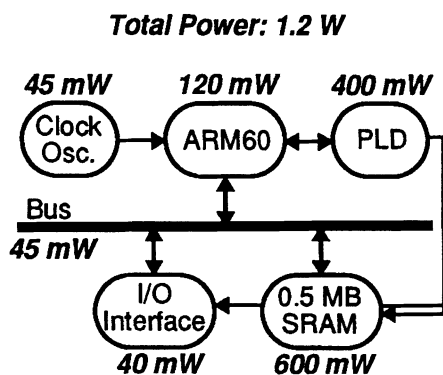


Figure 2. InfoPad processor subsystem.

is the common technology of choice for low power systems, due to its minimal static power dissipation, and high level of integration.

3.1. Power Dissipation

CMOS circuits have both static and dynamic power dissipation. Static power arises from bias and leakage currents. While static gate loads are usually found in a few specialized circuits such as PLAs, their use has been dramatically reduced in CMOS designs focussed on low power. Furthermore, careful design of these gates can make their power contribution negligible in circuits that do use them [7]. Leakage currents from reverse-biased diodes of MOS transistors and from MOS subthreshold conduction [8] also dissipate static power but are also insignificant in most designs that dissipate more than 1 mW.

The dominant component of power dissipation in CMOS is therefore dynamic. For every low-to-high logic transition in a digital circuit, the capacitance on that node, C_L , incurs a voltage change ΔV , drawing an energy $C_L \Delta V V_{DD}$ from the supply voltage at potential V_{DD} . For each node $n \in N$, these transitions occur at a fraction α_n of the clock frequency, f_{CLK} , so that the total dynamic switching power may be found by summing over all N nodes in the circuit:

$$Power = V_{DD} \cdot f_{CLK} \cdot \sum_{i=1}^N \alpha_i \cdot C_{L_i} \cdot \Delta V_i \quad (2)$$

Aside from memory bit-lines and low-swing logic, most nodes swing a ΔV from ground to V_{DD} , so that the power equation can be simplified to:

$$Power \cong V_{DD}^2 \cdot f_{CLK} \cdot C_{EFF} \quad (3)$$

where the effective switched capacitance, C_{EFF} , is commonly expressed as the product of the physical capacitance C_L and the activity weighting factor α , each averaged over the N nodes.

During a transition on the input of a CMOS gate both p and n channel devices may conduct simultaneously, briefly establishing a short from V_{DD} to ground. In properly designed circuits, however, this short-circuit current typically dissipates a small fraction (5–10%) of the dynamic power [9] and will be omitted in further analyses.

3.2. Circuit Delay

To fully utilize its hardware, a digital circuit should be operated at the maximum possible frequency. This maximum frequency is just the inverse of the delay of the processor's critical path.

Until recently, the long-channel delay model suitably modeled delays in CMOS circuits [8]. However, minimum device channel lengths, L_{MIN} , have scaled below 1 micron, degrading the performance of the device due to velocity saturation of the channel electrons. This phenomenon occurs when the electric field (V_{DD}/L_{MIN}) in the channel exceeds 1 V/ μm [10].

$$\begin{aligned} Delay &\cong \frac{C_L}{I_{AVE}} \cdot \frac{\Delta V}{2} \\ &\cong \frac{C_L \cdot V_{DD}}{k_V \cdot W \cdot (V_{DD} - V_T - V_{DSAT})} \quad (4) \end{aligned}$$

The change in performance can be characterized by the short-channel or velocity-saturated delay model shown in Eq. (4). I_{AVE} is the average current being driven onto C_L , and is proportional to device width W , technology constant k_V , and to first-order, V_{DD} . V_T is the threshold voltage. For large V_{DD} , V_{DSAT} is constant, with typical magnitude on order of V_T . For V_{DD} values less than $2V_T$, V_{DSAT} asymptotically approaches $V_{DD} - V_T$. The important difference between the two delay models is that in the latter, current is roughly linear, and not quadratic, with V_{DD} .

3.3. Throughput

Throughput was previously defined as the number of operations that can be performed in a given time. When clock rate is inversely equal to the critical path delay, throughput is equal to the amount of computational concurrency (i.e., operations completed per clock cycle) divided by the critical path delay:

$$T = \frac{\text{Operations}}{\text{Second}} = \frac{\text{Operations per clock cycle}}{\text{Critical path delay}} \quad (5)$$

The critical path delay can be related back to the velocity-saturated delay model by summing up the delay over all M gates in the critical path:

$$\text{Critical path delay} \cong \frac{V_{DD}}{k_V \cdot (V_{DD} - V_T - V_{DSAT})} \cdot \sum_{i=1}^M \frac{C_{L_i}}{W_i} \quad (6)$$

Making the approximation that all gate delays are equal, Eq. (6) can be simplified if N_{gates} is used to indicate the length of the critical path (i.e., number of gates), and average values for C_L and W are used. Throughput can now be expressed as a function of technology parameters, supply voltage, critical path length, and operations per clock cycle:

$$T \cong \frac{k_V \cdot W \cdot (V_{DD} - V_T - V_{DSAT})}{N_{gates} \cdot C_L \cdot V_{DD}} \cdot \frac{Operations}{Clock\ cycle} \quad (7)$$

As mentioned earlier, typical units for operations per clock cycle are MIPS/Mhz, and SPECint92/MHz when operations are respectively defined as instructions and benchmark programs.

3.4. Energy/Operation

A common measure of energy consumption is the power-delay product (PDP) [11]. This delay is often defined as the critical path delay, so PDP is equivalent to the energy consumed per clock cycle ($Power/f_{CLK}$). However, the measure of interest is the energy consumed per operation which can be derived by dividing the PDP by the operations per clock cycle. The energy consumed per operation can now be expressed as a function of effective switched capacitance, supply voltage, and operations per clock cycle:

$$\frac{Energy}{Operation} \cong \frac{V_{DD}^2 \cdot C_{EFF}}{Operations/Clock\ cycle} \quad (8)$$

3.5. Technology Scaling

Although it is usually beyond the control of the IC designer, it is worth noting the impact of technology scaling on throughput and energy/operation. Capacitances and device width, W , scale down linearly with minimum channel length L_{MIN} ; transistor current is approximately independent of L_{MIN} if V_{DD} remains fixed; technology constant k_V scales approximately inversely with L_{MIN} . So, as L_{MIN} is scaled down, the throughput scales up and the energy/operation is reduced, thus yielding the conclusion that technology scaling is an important strategy for improving energy efficiency.

4. Energy Efficiency

While the energy consumed per operation should always be minimized, no single metric quantifies energy

efficiency for all digital systems. The metric is dependent on the system's throughput constraint. We will investigate the three main modes of computation: fixed throughput, maximum throughput, and burst throughput. Each of these modes has a clearly defined metric for measuring energy efficiency, as detailed in the following three sections. While portable devices typically operate in the burst throughput mode, the other two modes are equally important since they are degenerate forms of the burst throughput mode in which the portable device may operate.

4.1. Fixed Throughput Mode

Most real-time systems require a fixed number of operations per second. Any excess throughput cannot be utilized, and therefore needlessly consumes energy. This property defines the fixed throughput mode of computation. Systems operating in this mode are predominantly found in digital signal processing applications in which the throughput is fixed by the rate of an incoming or outgoing real-time signal (e.g., speech, video, handwriting).

$$Metric|_{FIX} = \frac{Power}{Throughput} = \frac{Energy}{Operation} \quad (9)$$

Previous work has shown that the metric of energy efficiency in Eq. (9) is valid for the fixed throughput mode of computation [11]. A lower value implies a more energy-efficient solution. If a design can be made twice as energy efficient (i.e., reduce the energy/operation by a factor of two), then its sustainable battery life has been doubled and since the throughput is constant its power dissipation has been halved. For the case of fixed throughput, minimizing the power dissipation is equivalent to minimizing the energy/operation.

4.2. Maximum Throughput Mode

In most multi-user systems, primarily networked desktop computers and mainframes, the processor is continuously running. The faster the processor can perform computation, the better, yielding the defining characteristic of the maximum throughput mode of computation. Thus, this mode's metric of energy efficiency must balance the need for low energy/operation and high throughput, which is accomplished through the use of the Energy to Throughput Ratio, or ETR given

in Eq. (10),

$$\text{Metric}|_{\text{MAX}} = \text{ETR} = \frac{E_{\text{MAX}}}{T_{\text{MAX}}} = \frac{\text{Power}}{\text{Throughput}^2} \quad (10)$$

where E_{MAX} is the energy/operation, or equivalently power/throughput, and T_{MAX} is the throughput in this mode.

A lower ETR indicates lower energy/operation for equal throughput or equivalently indicates greater throughput for a fixed amount of energy/operation, satisfying the need to equally optimize throughput and energy/operation. Thus, a lower ETR represents a more energy-efficient solution. The Energy-Delay Product [5] is a similar metric, but does not include the effects of architectural parallelism when the delay is taken to be the critical path delay.

Throughput and energy/operation can be scaled with supply voltage, as shown in Fig. 3 (the data for Figs. 3–5 is derived from Eqs. (7) and (8), and suitably models sub-micron processes); but, unfortunately, they do not scale proportionally. So while throughput and energy/operation can be varied by well over an order of magnitude to cover a wide dynamic range of operating points, the ETR is not constant for different values of supply voltage.

As shown in Fig. 4, V_{DD} can be adjusted by a factor of almost three ($1.4V_T$ to $4V_T$) and the ETR only varies within 50% of the minimum at $2V_T$. However, outside this range, the ETR rapidly increases. Clearly, for supply voltages greater than 3.3 V there is a rapid degradation in energy efficiency, as well as for supply voltages that approach the device threshold voltage.

To compare designs over a larger range of operation for the maximum throughput mode, a better metric is

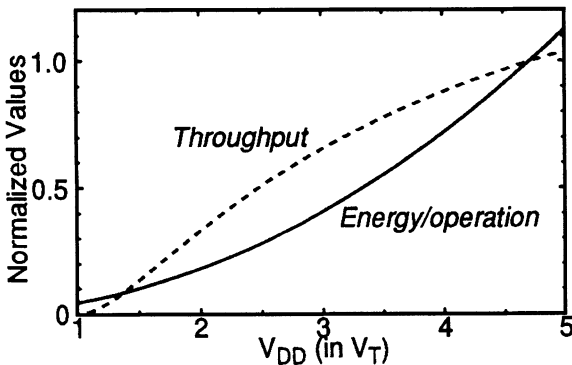


Figure 3. Energy/operation, throughput vs. V_{DD} .

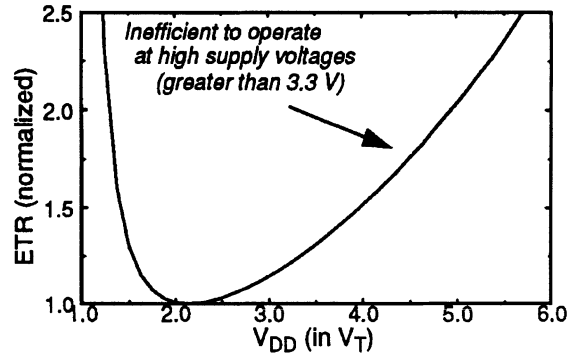


Figure 4. ETR as a function of V_{DD} .

a plot of the energy/operation versus throughput. To make this plot, the supply voltage is varied from the minimum operating voltage (near V_T in many digital CMOS designs) to the maximum voltage (2.5–5 V, depending on the technology), while energy/operation and throughput are measured. The energy/operation can then be plotted as a function of throughput, and the architecture is completely characterized over all possible throughput values.

Using the ETR metric is equivalent to making a linear approximation to the actual energy/operation versus throughput curve. Figure 5 demonstrates the error incurred in using a constant ETR metric, which is calculated at a nominal supply voltage of 3.3 V for this example. For architectures with similar throughput, a single ETR value is a reasonable metric for energy efficiency; however, for designs optimized for vastly different values of throughput, a plot may be more useful, as Section 5.1 demonstrates.

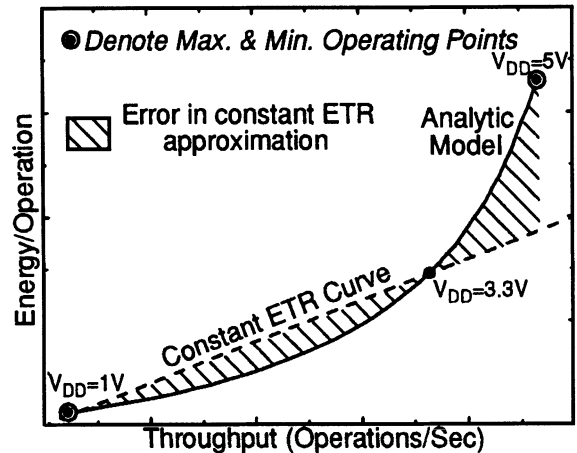


Figure 5. Energy vs. throughput metric.

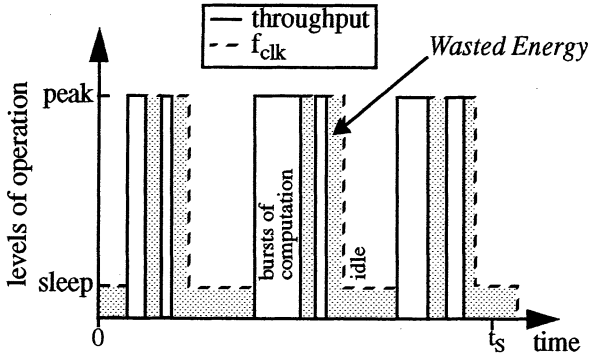


Figure 6. Wasted energy due to idle cycles.

4.3. Burst Throughput Mode

Most single-user systems (e.g., stand-alone desktop computers, notebook computers, PDAs, etc.) spend a fraction of the time performing useful computation. The rest of the time is spent idling between processes. However, when bursts of computation are demanded, the faster the throughput (or equivalently, response time), the better. This characterizes the burst throughput mode of computation in which most portable devices operate. The metric of energy efficiency used for this mode must balance the desire to minimize energy consumption, while both idling and computing, and to maximize peak throughput when computing.

Ideally, the processor's clock should track the periods of computation in this mode so that when an idle period is entered, the clock is immediately shut off. Then a good metric of energy efficiency is just ETR, as the energy consumed while idling has been eliminated. However, this is not realistic in practice. Many processors do not have an energy saving mode and those that do so generally support only simple clock reduction/deactivation modes. The hypothetical example depicted in Fig. 6 contains a clock reduction (sleep) mode in which major sections of the processor are shut down. The shaded area indicates the processor's idle cycles in which energy is needlessly consumed, and whose magnitude is dependent upon whether the processor is operating in the "low-power" mode.

$$E_{MAX} = \frac{\text{Total energy consumed computing}}{\text{Total operations}} \quad (11)$$

$$E_{IDLE} = \frac{\text{Total energy consumed idling}}{\text{Total operations}} \quad (12)$$

Total energy and total operations can be calculated over a large sample time period, t_s . T_{MAX} is the peak

throughput during the bursts of computation (similar to that defined in Section 4.2), and T_{AVE} is the time-averaged throughput (total operations/ t_s). If the time period t_s is sufficiently long that the operation characterizes the "average" computing demands of the user and/or target system environment yielding the average throughput (T_{AVE}), then a good metric of energy efficiency for the burst throughput mode is:

$$\text{Metric}|_{BURST} = \text{METR} = \frac{E_{MAX} + E_{IDLE}}{T_{MAX}} \quad (13)$$

This metric will be called the Microprocessor ETR (METR); it is similar to ETR, but also accounts for energy consumed while idling. A lower METR represents a more energy-efficient solution.

Multiplying Eq. (11) by the actual time computing = [$t_s \cdot$ (fraction of time computing)], shows that E_{MAX} is the ratio of compute power dissipation to peak throughput T_{MAX} , as previously defined in Section 4.2. Thus, E_{MAX} is only a function of the hardware and can be measured by operating the processor at full utilization.

E_{IDLE} , however, is a function of t_s and T_{AVE} . The power consumed idling must be measured while the processor is operating under typical conditions, and T_{AVE} must be known to then calculate E_{IDLE} . However, expressing E_{IDLE} as a function of E_{MAX} better illustrates the conditions when idle energy consumption is significant. In doing so, E_{IDLE} will also be expressed as a function of the idle power dissipation, which is readily calculated and measured, as well as independent of t_s and T_{AVE} .

Equation (12) can be rewritten as:

$$E_{IDLE} = \frac{[\text{Idle power dissipation}][\text{Time idling}]}{[\text{Average throughput}][\text{Sample time}]} \quad (14)$$

With the Power-Down Efficiency, β , defined as:

$$\beta = \frac{\text{Power dissipation while idling}}{\text{Power dissipation while computing}} = \frac{P_{IDLE}}{P_{MAX}} \quad (15)$$

E_{IDLE} can now be expressed as a function of E_{MAX} :

$$E_{IDLE} = \frac{[\beta \cdot E_{MAX} \cdot T_{MAX}] \cdot [(1 - T_{AVE}/T_{MAX})t_s]}{[T_{AVE}] \cdot [t_s]} \quad (16)$$

Equation (17) shows that idle energy consumption dominates total energy consumption when the

fractional time spent computing (T_{AVE}/T_{MAX}) is less than the fractional power dissipation while idling (β).

$$METR = ETR \left[1 + \beta \left(\frac{T_{MAX}}{T_{AVE}} - 1 \right) \right], T_{MAX} \geq T_{AVE} \quad (17)$$

The METR is a good metric of energy efficiency for all values of T_{AVE} , T_{MAX} , and β as illustrated below by analyzing the two limits of the METR metric.

Idle Energy Consumption is Negligible ($\beta \ll T_{AVE}/T_{MAX}$): The metric should simplify to that found in the maximum throughput mode, since it is only during the bursts of computation that energy is consumed and operations performed. For negligible power dissipation during idle, the METR metric in Eq. (17) degenerates to the ETR, as expected. For perfect power-down ($\beta = 0$) and minimal throughput ($T_{MAX} = T_{AVE}$), the METR is exactly the ETR.

Idle Energy Consumption Dominates ($\beta \gg T_{AVE}/T_{MAX}$): The energy efficiency should increase by either reducing the idle energy/operation while maintaining constant throughput, or by increasing the throughput while keeping idle energy/operation constant. While it might be expected that these are independent optimizations, E_{IDLE} may be related back to E_{MAX} and the throughput by β since T_{AVE} is fixed:

$$\frac{E_{IDLE}}{E_{MAX}} \cong \frac{P_{IDLE}/T_{AVE}}{P_{MAX}/T_{MAX}} = \beta \cdot \frac{T_{MAX}}{T_{AVE}} \quad (18)$$

Expressing E_{IDLE} as a function of E_{MAX} yields:

$$METR \cong \frac{\beta \cdot E_{MAX}}{T_{AVE}}, \text{ (idle energy dominates)} \quad (19)$$

If β remains constant for varying throughput (and E_{MAX} stays constant), then E_{IDLE} scales with throughput as shown in Eq. (18). Thus, the METR becomes an energy/operation minimization similar to the fixed throughput mode. However, β may vary with throughput, as will be analyzed further in Section 7.

4.4. Energy Efficiency for Portable Systems

As mentioned earlier, the METR metric measures the energy efficiency of portable systems. Unfortunately, information on the system's average throughput (T_{AVE}) is required to utilize this metric and is very application specific. Thus, the METR metric cannot be used

to describe the energy efficiency of a processor in general terms. It is only useful when a target application (or class of related applications) has been specified. An example application is the InfoPad, as described in Section 2.3. The processor system is responsible for packet-level network control on the pad and has an average throughput requirement of 0.8 MIPS. If the video decompression was implemented by the processor rather than the custom chip-set, then the average throughput would increase to approximately 11 MIPS.

So that energy-efficient design techniques can be discussed independent of the final application, the METR metric's subcomponents, ETR and E_{IDLE} , will be discussed individually. Section 6 discusses design techniques for optimizing ETR, and Section 7 discusses techniques for minimizing idle energy consumption.

5. Design Principles

Four examples are presented below to demonstrate how energy efficiency can be properly quantified. In the process, four design principles follow from the optimization of the previously defined metrics: a high-performance processor can be an energy-efficient processor; idle energy consumption limits the energy efficiency for high-throughput operation; reducing the clock frequency is not energy efficient; and dynamic voltage scaling is energy efficient.

5.1. High Performance is Energy Efficient

Table 1 lists two processors that are available today—the ARM710 targets the low-power market, and the R4700 targets the mid-range workstation market, and both are fabricated in similar 0.6 um technologies, facilitating an equal comparison. The measure of throughput used is SPECint92. The typical metric for measuring energy efficiency is SPECint92/Watt (or SPECfp92/Watt, Dhrystones/Watt, MIPS/Watt, etc.). The ARM710 processor has a SPECint92/Watt five times greater than the R4700's, and the claim then follows that it is "five times as energy efficient". However, this metric only compares operations/energy, and does not weight the fact that the ARM710 has only 15% of the performance as measured by SPECint92.

The ETR (Watts/SPECint92²) metric indicates that the R4700 is actually *more* energy efficient than the ARM710. To quantify the efficiency increase, the plot of energy/operation versus throughput in Fig. 7 is used because it better tracks the R4700's energy at the low

Table 1. Comparison of two processors [12, 42].

Processor	SPECint92 (T_{MAX})	Power (Watt)	Supply voltage, V_{DD} (volts)	SPECint92/Watt ($1/E_{MAX}$)	ETR (10^{-3})
R4700	130	4.0	3.3	33	0.24
ARM710	20	0.12	3.3	167	0.30

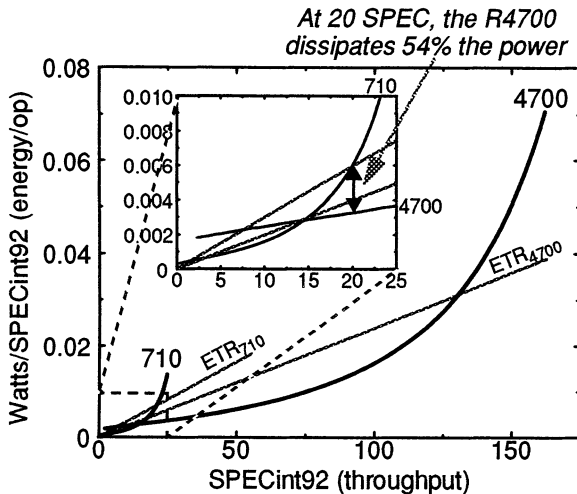


Figure 7. Energy vs. throughput of R4700 and ARM710.

throughput values. The plot was generated from the throughput and energy/operation models in Section 3.

According to the plot, the R4700 would dissipate 65 mW at 20 SPECint92, or about 1/2 of the ARM710's power, despite the low V_{DD} ($1.5V_T$) for the R4700. Conversely, the R4700 can deliver 30 SPECint92 at 120 mW ($V_{DD} = 1.7V_T$), or 150% of the ARM710's throughput.

This does assume that the R4700 processor has been designed so that it can operate at these low supply voltages. If the lower bound on operating voltage is greater than $1.7V_T$, then the ARM710 would be more energy efficient in delivering the 20 SPECint92 than the R4700. Typically, a processor is rated for a fixed standard supply voltage (3.3 V or 5.0 V) with a $\pm 10\%$ tolerance. However, many processors can operate over a much larger range of supply voltages (e.g., 2.7–5.5 V for the ARM710 [12], 2.0–3.3 V for the Intel486GX [13]). The processor can operate at a non-standard supply voltage by using a high-efficiency, low-voltage DC-DC converter to generate the appropriate supply voltage [14].

While the ETR correctly predicted the more energy-efficient processor at 20 SPECint92, it is important to note that the R4700 is not more energy efficient for all values of SPECint92, as the ETR metric would indicate.

Because the nominal throughput of the processors is vastly different, the Energy/Operation versus Throughput metric better tracks the efficiency, and indicates a cross-over throughput of 14.5 SPECint92. Below this value, the ARM710 is more energy efficient.

5.2. Fast Operation Can Decrease Energy Efficiency

If the user demands a fast response time, rather than reducing the voltage, as was done in Section 5.1, the processor can be left at the nominal supply voltage, and shut down when it is not needed.

For example, assume the target application has a T_{AVE} of 20 SPEC, and both the ARM710 and R4700 have a β factor of 0.2. If the processors' V_{DD} is left at 3.3 V, The ARM710's METR is exactly equal to its ETR value, which is 3.0×10^{-4} . It remains the same because it never idles. The R4700, on the other hand, spends 85% ($1 - T_{AVE}/T_{MAX}$) of the time idling, and its METR is 5.0×10^{-4} . Thus, for this scenario, the ARM710 is nearly twice as energy efficient.

However, if the R4700's β can be reduced down to 0.02, then the METR of the R4700 becomes 2.66×10^{-4} , and it is once again the more energy-efficient solution. For this example, the cross-over value of β is 0.045.

This example demonstrates how important it is to use the METR metric instead of the ETR metric if the target application's idle time is significant (i.e., T_{AVE} can be characterized and is significantly below T_{MAX}). For the above example, a β for the R4700 greater than 0.045 leads the metrics to disagree on which is the more energy-efficient solution. One might argue that the supply voltage can always be reduced on the R4700 so that it is more energy efficient for any required throughput. This is true if the dynamic range of the R4700 is as indicated in Fig. 7. However, if some internal logic limited the value that V_{DD} could be dropped, then the lower bound on the R4700's throughput would be located at a much higher value. Thus, finite β can degrade the energy efficiency of high throughput circuits due to excessive idle power dissipation.

Table 2. Impact of clock frequency reduction on energy efficiency.

Operating conditions	Compute energy consumption dominates	Idle energy consumption dominates	
		β independent of throughput	β inversely proportional to throughput
Throughput	Decreases	Decreases	Decreases
Energy	Unchanged	Decreases	Unchanged
Energy efficiency (METR)	Decreases	Unchanged	<i>Decreases</i>

5.3. Clock Frequency Reduction is NOT Energy Efficient

A common fallacy is that reducing the clock frequency f_{CLK} is energy efficient. When compute energy consumption dominates idle energy consumption, it is quite the opposite. At best, it allows an energy-throughput trade-off when idle energy consumption is dominant. The relative amount of time in idle versus maximum throughput is an important consideration in determining the effect of clock frequency reduction.

Compute Energy Consumption Dominates ($E_{MAX} \gg E_{IDLE}$): Since compute energy consumption is independent of f_{CLK} , and throughput scales proportionally with f_{CLK} , decreasing the clock frequency increases the ETR, indicating a drop in energy efficiency. Halving f_{CLK} is equivalent to doubling the computation time, while maintaining constant computation per battery life, which is clearly not optimal.

Idle Energy Consumption Dominates ($E_{IDLE} \gg E_{MAX}$): Clock reduction may trade-off throughput and energy/operation, but only when the power-down efficiency, β , is independent of throughput such that E_{IDLE} scales with throughput. When this is so, halving f_{CLK} will double the computation time, but will also double the amount of computation per battery life. If the currently executing process can accept throughput degradation, then this may be a reasonable trade-off. If β

is inversely proportional to throughput, however, then reducing f_{CLK} does not affect the total energy consumption, and the energy efficiency drops.

5.4. Dynamic Voltage Scaling is Energy Efficient

If V_{DD} were to track f_{CLK} , however, so that the critical path delay remains inversely equal to the clock frequency, then constant energy efficiency could be maintained as f_{CLK} is varied. This is equivalent to V_{DD} scaling (Section 4.2) except that it is done dynamically during processor operation. If E_{IDLE} is present and dominates the total energy consumption, then simultaneous f_{CLK} , V_{DD} reduction during periods of idle will yield a more energy-efficient solution.

Even when idle energy consumption is negligible, dynamic voltage scaling can still provide significant wins. Figure 8 plots a sample usage pattern of desired throughput, with the delivered throughput superimposed on top. For background and high-latency tasks, the supply voltage can be reduced so that just enough throughput is delivered, which minimizes energy consumption.

For applications that require maximum deliverable throughput only a small fraction of the time, dynamic voltage scaling has a significant win. For the R4700 processor, the peak throughput is 130 SPECint92. Given a target application where the desired throughput is either a fast 130 SPECint92 or a slow 13 SPECint92, Table 3 lists the peak throughput, average

Table 3. Benefits of dynamic voltage scaling.

Throughput	Time spent operating in:			T_{MAX} (SPEint92)	E_{AVE} (W/SPECint92)	ETR (10^{-6})	Normalized battery life
	Fast mode	Slow mode	Idle mode				
Always full-speed	10%	0%	90%	130	0.031	237	1 hr.
Sometimes full-speed	1%	90%	9%	130	0.006	45.0	5.3 hrs.
Rarely full-speed	0.1%	99%	0.9%	130	0.003	25.8	9.2 hrs.

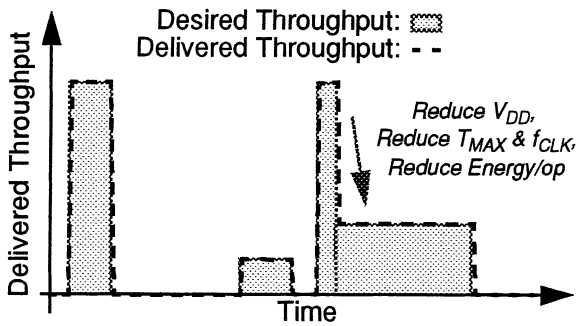


Figure 8. Dynamic voltage scaling.

energy/operation, and effective ETR depending on the fraction of time spent in the fast mode. For each category of throughput the total number of operations completed are the same so that the relative changes in battery life can be evenly compared. When that fraction becomes small, the processor's peak throughput is still set by the fast mode, while the average energy consumed per operation is set by the slower mode. Thus, the best of both extremes can be achieved. For simplicity, this examples assumes that idle energy consumption is always negligible.

6. Energy-Efficient VLSI Design

When idle energy consumption of the processor is negligible, ETR is a valid energy-efficiency metric. A variety of low-power and low-energy design techniques have been published but are not energy efficient in the ETR sense, if the power/energy savings came at the expense of throughput. Various design techniques drawn from the literature and original research are discussed below as well as their impact on ETR.

The techniques can be categorized into one of three areas of processor design: instruction set architecture, microarchitecture, and circuit design. However, these are not entirely independent; design decisions in one area may impact design decisions made in the other areas.

6.1. Instruction Set Architecture

Typically, instruction set architectures (ISA) are designed solely with performance in mind. High-level performance simulators allow the architect to explore the ISA design space with reasonable efficiency. Energy is not a consideration, nor are there high-level

simulators available to even let the architect estimate energy consumption. Simulation tools exist, but require a detailed description of the microarchitecture so that they are not useful until the ISA has been completely specified. Processor's targeted towards portable systems should have their ISA designed for energy efficiency, and not just performance.

Many processors have 32-bit instruction words and registers. Register width generally depends on the required memory address space, and cannot be reduced; in fact, more recent microprocessors have moved to 64-bits. For low-energy processors, 16-bit instruction widths have been proposed. Static code density can be reduced by 30–35%, while increasing the dynamic run length by only 15–20% over an equivalent 32-bit processor [15, 16]. Using 16-bit instructions reduces the energy cost of an instruction fetch by up to 50% because the size of the memory read has been halved [17]. In system's with 16-bit external busses, the advantage of 16-bit instructions is further widened [16, 18]. Since instruction fetch consumes about a third of the processor's energy [19, 20], total energy consumption is reduced by 15–20%, which is cancelled out by the 15–20% reduction in performance, giving approximately equal energy efficiency. The available data indicates that this technique is significantly energy efficient only if the external memory's energy consumption dominates the processor's energy consumption, or if the external bus is 16 bits.

The number of registers can be optimized for energy efficiency. The register file consumes a sizable fraction of total energy consumption since it is typically accessed multiple times per cycle (10% of the total energy in [19]). In a register-memory architecture, the number of general purpose registers is kept small and many operands are fetched from memory. Since the energy cost of a cache access surpasses that of a moderately sized (32) register file, this is not energy efficient. The other extreme is to implement register windows which is essentially a very large (100+) register file. The energy consumed by the register file increases dramatically increasing total processor energy consumption 10–20%. Unless this increase in energy is compensated by an equivalent increase in performance, register windows is not energy efficient. One study compared register files of size 16 and 32 for a given ISA, and found that for 16 registers, the dynamic run length is 8% larger [21]. The corresponding decrease in processor energy due to a smaller register file is on the order of 5–10%. There appears to be a broad optimum

on the number of registers since the energy efficiency is near equal for 16 and 32-entry register file.

The issue of supported operation types and addressing modes has been a main philosophical division between the RISC and CISC proponents. While this issue has been debated solely in the context of performance, it can also have an impact on energy consumption. Complex ISAs have higher code density, which reduces the energy consumed fetching instructions and reduces the total number of instructions executed. Simple ISAs typically have simpler data and control paths, which reduces the energy consumed per instruction, but there are more instructions. These trade-offs need to be analyzed when creating an ISA.

The amount of hardware exposed (e.g., branch delay slot, load delay slot, etc.) is another main consideration in ISA design. This is typically done to improve performance by simplifying the hardware implementation. Since the scheduling complexity resides in the compiler, it consumes zero run-time energy while the simplified hardware consumes less energy per operation. Thus, both the performance is increased and the energy/operation is decreased, giving a two-fold increase in energy efficiency. A good example of radically exposing the hardware architecture are very long instruction word (VLIW) architectures, which will be discussed in more detail in the next section.

6.2. Microarchitecture

The predominant technique to increase energy efficiency in custom DSP ICs (fixed throughput) is architectural concurrency; with regards to processors, this is generally known as instruction-level parallelism (ILP). Previous work on fixed throughput applications demonstrated an energy efficiency improvement of approximately N on an N -way parallel/pipelined architecture [11]. This assumes that the instructions being executed are fully vectorizable, that N is not excessively large, and that the extra delay and energy overhead for multiplexing and demultiplexing the data is insignificant.

Moderate pipelining (4 or 5 stages), while originally implemented purely for speed, also increases energy efficiency, particularly in RISC processors that operate near one cycle-per-instruction. Energy efficiency can be improved by a factor of two or more [22], and is essential in an energy-efficient processor.

Superscalar Architectures: More recent processor designs have implemented superscalar architectures, either with parallel execution units or extended pipelines, in the hope of further increasing the processor concurrency. However, an N -way superscalar machine will not yield a speedup of N , due to the limited ILP found in typical code [23, 24]. Therefore, the achievable speedup will be less than the number of simultaneous issuable instructions and yields diminishing returns as the peak issue rate is increased. Speedup has been shown to be between two and three for practical hardware implementations in current technology [25].

If the instructions are dynamically scheduled in employing superscalar operation, as is currently common to enable backwards binary compatibility, the C_{EFF} of the processor will increase due to the implementation of the hardware scheduler. Also, there will be extra capacitive overhead due to branch prediction, operand bypassing, bus arbitration, etc. There will be additional capacitance increase because the N instructions are fetched simultaneously from the cache and may not all be issuable if a branch is present. The capacitance switched for un-issued instructions is amortized over those instructions that are issued, further increasing C_{EFF} .

The energy efficiency increase can be analytically modeled. Equation (20) gives the ETR ratio of a superscalar architecture versus a simple scalar processor; a value larger than one indicates that the superscalar design is more energy efficient. The S term is the ratio of the throughputs, and the C_{EFF} terms are from the ratio of the energies (architectures are compared at constant supply voltage). The individual terms represent the contribution of the datapaths, C_{EFF}^{Dx} , the memory sub-system, C_{EFF}^{Mx} , and the dynamic scheduler and other control overhead, C_{EFF}^{Cx} . The 0 suffix denotes the scalar implementation, while the 1 suffix denotes the superscalar implementation. The quantity C_{EFF}^{C0} has been omitted, because it has been observed that the control overhead of the scalar processor is minimal: $C_{EFF}^{C0} \ll C_{EFF}^{D0, M0}$ [19].

$$ETR|_{RATIO} = \frac{S(C_{EFF}^{D0} + C_{EFF}^{M0})}{(C_{EFF}^{C1} + C_{EFF}^{D1} + C_{EFF}^{M1})} \quad (20)$$

Simulation results show that C_{EFF}^{C1} is significant due to control overhead and that C_{EFF}^{M1} is greater than C_{EFF}^{M0} due to un-issued instructions negating the increase due to S . Since C_{EFF}^{C1} increases quadratically as the number

of parallel functional units is increased, the largest improvement in energy efficiency would be expected for moderate amounts of parallelism. In this best case, however, the superscalar architecture yields no improvement in energy efficiency [22].

Superpipelined Architectures: These architectures also exploit ILP and offer speedups similar to those found in superscalar architectures [26], but their performance is lower because the number of stall cycles increases with the depth of the pipeline due to data dependencies. While these architectures do not need as complex hardware for the dynamic scheduler (C_{EFF}^{Cx} is lower), they do need extra hardware for more complex operand bypassing (C_{EFF}^{Dx} is higher). The net differences in speedup and capacitance should give superpipelined architectures an energy efficiency similar to superscalar architectures.

VLIW Architectures: These architectures best exploit ILP by exposing the underlying parallelism of the hardware to the compiler's scheduler which minimizes the complexity of the hardware. A good compiler is necessary to fully utilize the hardware. One such implementation from Multiflow gives a speedup factor, S , between 2 and 6 [27]. Because the parallelism is visible, VLIW processors do not require aggressive branch prediction, dynamic schedulers, and complex bus arbitration, so that the energy consumed per operation is roughly the same as for the scalar processor. The main additional energy cost is for the communication network that connects the autonomous functional units that comprise the VLIW processor, and executing the instructions that shuffle data between them. Even assuming a worst case energy per operation increase of 50%, the VLIW processor's energy efficiency increases anywhere from 33% to 300%.

On-chip caches reduce off-chip communication that is both slow and energy consuming. Caches consume around a third of the processor's energy consumption (50% in [19]). Designing the cache in a sectored (or sub-banked) manner, such that only one part of the SRAM array is activated per memory access reduces energy/access and increases throughput [4]; this is recommended technique for any memory larger than one kilobyte.

For split caches, the instruction cache consumes up to four times the data cache's energy consumption since loads and stores do not occur every instruction. For the instruction cache, an instruction buffer (or Level

0 cache) can dramatically increase energy efficiency by exploiting the spatial locality of instructions [17]. When a cache line is accessed, it is placed into a buffer, and the instruction cache is not accessed again until the instruction buffer takes a miss. For a 32-byte wide buffer, the hit rate is around 80% [17]; this reduces the instruction cache energy consumption up to 80%. If the buffer is designed to have no penalty on a miss so that performance is unchanged, the processor energy efficiency can be improved by 15–25%. Further techniques have been proposed to reduce the accesses to the instruction cache's tag array by exploiting this same spatial locality, increasing processor energy efficiency of 5–10% [28].

The processor control typically knows which pipeline stages are being used each cycle. Those pipeline stages not used in a given cycle should have their clock disabled for that cycle. This is particularly important to do in superscalar architectures that typically have only a fraction of the entire processor being utilized in any given cycle. With only a small overhead cost, this technique increases processor energy efficiency by 15–25% (estimated that 40–50% of the processor is disabled 40–50% of the time) [30]. To maximize the benefit of clock-gating, NOP instructions should be suppressed. In many microarchitectures, NOP instructions are mapped to real instructions. Although NOPs write to a null register, they consume more than half the energy of a normal instruction, as demonstrated by empirical measurements described in [30]. Instead, NOPs should be detected by a comparator in the instruction decode stage, and later stages executing on the NOP should be disabled. Similarly, pipeline stalls and/or bubbles should not inject NOP instructions into the pipeline but should instead cause subsequent pipeline stages to be disabled during the appropriate cycle.

Correlation of data is often exploited for energy efficiency in signal processing circuits. While processors do not exhibit the same level of correlation as found in DSP circuits, high amounts of correlation can be found in calculating the effective address which is typically offset from a high-valued stack pointer. In most scalar processors, a single ALU calculates the effective addresses and all integer additions. By partitioning these two types of additions onto separate adders, the signal correlation increases by 16%, decreasing the adder's energy consumed per addition by an equivalent 16%. Total processor energy efficiency is then increased by 3–7%.

The ETR metric can be used to evaluate other microarchitectural design decisions for their relative energy efficiency. For those decisions with more than one feasible approach, the relative ETRs can be compared to select the most energy-efficient alternative.

6.3. Circuit Design

A variety of energy-efficient design techniques exist at the circuit design level. Many were developed in earlier research targeted towards custom DSP (fixed throughput) design, and some remain applicable to general-purpose processor design [31]. The ETR metric can be used to determine which of these “low power” techniques are also energy-efficient design choices. Of the three levels of the processor design hierarchy, the circuit design level has by far the largest amount of previously results from which to draw.

For example, the topologies for the various macrocells (e.g., adder, register file, etc.) should be selected by their ETR, and neither solely for speed nor solely for energy. A variety of studies have been made to study the relative energy consumption and speed of various macrocells, which can be used to aid in making design decisions [32, 33]. Similar studies have also been with respect to various logic design styles [34].

Transistor level optimizations can be made, such as minimizing all devices not in the critical path(s). This typically requires have a fast and slow versions of the same cell, and the cell selection is based on whether it is in the critical path(s) or not [29]. Low-voltage swing circuits for large capacitive nodes, such as those found in memories and global busses, can significantly drop energy consumption, while improving speed at the same time [7].

7. Minimizing Idle Energy Consumption

As demonstrated in Section 2.1, when the processor is not actively computing on user or background tasks, the desired throughput is zero. Any throughput delivered by the processor in this idle mode needlessly consumes energy. The METR metric is revisited to understand when idle energy consumption is important followed by a survey of design techniques to minimize this energy.

7.1. Optimizing METR

Equation (17) shows that when the fractional time spent computing (T_{AVE}/T_{MAX}) is less than the fractional

power dissipation while idling (β), idle energy consumption dominates total energy consumption. Then the METR optimization is to minimize β and E_{MAX} as shown in Eq. (19). Furthermore, the exact optimization depends on whether β changes as the throughput is varied as shown below.

β is Independent of Throughput: This is the case when the processor has no power-down mode. If the clock frequency remains the same, or proportional, during both the computation and idle periods, then idle power dissipation tracks compute power dissipation. Idle energy consumption cannot be optimized independent of throughput and compute energy consumption. If throughput increases, the compute power dissipation increases, and the idle power dissipation and energy consumption increases proportionally. Minimizing the compute energy consumption will have a proportional decrease in the idle energy consumption.

β Varies with Throughput: This case occurs for processors that implement idle power down modes in which idle power dissipation is independent of compute power dissipation. It is energy efficient to maximize throughput, since idle energy consumption will remain constant and dominate compute energy consumption. In practice, β will be less than inversely proportional to throughput (e.g., due to latency switching between operating modes) so that idle energy consumption is not entirely independent of throughput. However, energy efficiency will continue to increase with throughput until idle energy consumption is no longer dominant.

7.2. Power Down Modes

Unless there is specific hardware support to externally disable a processor’s clock to turn off the processor when it is not being utilized, the processor typically executes a busy wait loop, which consists of NOP instructions. The processor hardware has an intrinsic, moderately-valued β which can be estimated or measured as the ratio of the power dissipated executing a NOP instruction to the power dissipated executing a typical instruction. Even if the clock is gated to those pipeline sections executing NOP instructions, the instruction-memory access per cycle will continue to consume energy. For a laptop computer in which average throughput is on the order of 1 SPECint92 (high estimate for user’s average operations/second) and β is reasonably estimated as 0.2, increasing the peak throughput of the processor beyond 5 SPE Cint92

reduces the processor energy efficiency. This is equivalent to a 386-class processor. To deliver a more tolerable response time to the user, energy efficiency will have to be degraded.

An alternative to degrading energy efficiency is to implement power down modes. To achieve their full benefit requires an energy-conscious operating system that utilizes them. Then, β can be decreased by one or more orders of magnitude. β will typically become a function of throughput since the operating system can decouple the compute and idle regimes' power dissipation. There may also be multiple values of β , one for each power down modes.

The design of the PowerPC 603 processor provides a good demonstration of useful power down modes to include [29]. A doze mode stops the processor from fetching instructions, but keeps alive snoop logic for cache coherency and the clock generation and timer circuits, giving a β of 0.16 for this mode. A nap mode disables the snoop logic, only keeping alive the timer logic, dropping the β to 0.06. Lastly, there is a sleep mode which only keeps alive the PLL and clock. The β for this mode is 0.05, while the processor can be up and running at full speed within 10 clock cycles, and a cache flush. Further power reduction can be achieved by disabling the PLL in the sleep mode, which drops the β down to 0.002, but at the cost of several thousand cycles (up to 200 usec) to return to full speed.

It is important to notice how much the PLL, which is found on most microprocessors, limits the reduction of idle energy consumption. Frequently turning off the PLL is not a viable approach due to the large overhead of restarting it. Techniques for improving the energy efficiency of PLLs in power down modes are needed.

While most microcontrollers and some embedded processors have power down modes, only a few microprocessors have them. It is an important technique to include in energy-efficient processors. The actual energy savings, though, depends more on how well the operating system can utilize these modes.

7.3. Transition Time

There is an energy cost associated with entering and leaving power down modes. When entering a mode, the processor will continue to operate at full throughput and energy consumption for a number of cycles while cleaning up state in the processor. This creates an energy penalty. When leaving a mode, there is usually a latency incurred to restart the processor, which creates

an effective throughput penalty. Restarting a PLL can cost several thousand clock cycles.

A metric has been proposed to measure this cost, the Cycles Per Stop Ratio, or CPSR, which includes the entry, exit, and processing overhead of entering a power down mode [35]. This is useful for first-order comparisons of various power down methods, but does not accurately measure the energy consumed per operation and performance.

To activate the power down modes for the PowerPC 603, the processor must handshake with the system logic via external control lines. In many microcontrollers and embedded microprocessors, instructions have been added to the ISA to directly activate power down modes. The benefit of this is to reduce the amount of time it takes to transition between modes, and in the best case, an instruction added to the ISA to shutdown the processor can take effect with one cycle of latency.

Included in the turn-off or restart time is the number of cycles it takes to save the internal processor state to memory. To save state in a consistent manner, it is best to allow the operating system to invoke the processor shut down instructions.

The cost of restarting the processor includes a one or two cycle delay for synchronously un-gating the clock, and a number of cycles delay equal to the pipeline depth to restart the pipeline. The PowerPC 603 is reported to have a start-up time of under 10 cycles for all the power down modes that leave the PLL running. The biggest latency cost is to restart the PLL, which typically takes 10–100 usec, to lock, and in the case of the 603 processor, can be as high as 200 usec, [29]. However, this is for an analog PLL. A digital PLL has been implemented with a reported lock time of under 2 usec., drastically reducing the start-up cost when in a fully powered-down mode (i.e., clock generation disabled) [36].

8. Energy-Efficient Software

Power down modes and halt instructions provide no benefit unless they are effectively used by the software running on the processor. Thus, an energy-efficiency minded operating system is crucial in portable systems. Other energy reductions can be achieved through variable-performance schedulers and optimizing compilers.

8.1. Operating System

The processor should be completely disabled during idle periods to minimize idle energy consumption.

Only the operating system has knowledge when there are no more pending events to process, and can invoke processor halt instructions to disable the processor. The operating system is central to system power management for a portable system.

Since the operating system is also aware of the peripheral hardware components' usage (e.g., disk drive, LCD, network controller, etc.), it should be given the ability to switch the power on and off to these devices as well. This is common practice in most notebook computers today and can reduce energy consumption by up to 50% [37]. With more aggressive design, such as a proposed technique for predictive shutdown of system components [38], this should be able to be reduced further.

Intel and Microsoft has put forth a specification called Advanced Power Management (APM) [39]. This specification defines an interface between hardware-specific power management software, which resides in the BIOS, and a hardware-independent operating system power management driver. This driver can manage APM-aware applications, by notifying them of impending processor state changes, and it provides an API that allows applications to directly employ power management. In a multi-tasking operating system, the driver will also negotiate conflicting power management requests. This vertical approach to power management shows great promise for further reductions in energy consumption.

8.2. Variable Performance Scheduling

Software processes have different performance and latency demands as shown in Section 2.1. Not every process needs the peak throughput of the processor. The supply voltage, along with the clock frequency, can be reduced to just meet the required throughput for those processes with lower performance demands, yielding a reduction in energy consumption, as described in Section 5.4.

The operating system can set the performance level at the time of a process context switch, with the level proportional to the priority level of the process. Most operating systems have the concept of process priority levels and the granularity of the performance settings increases with number of priority levels.

Another approach is to use predictive scheduling in which CPU performance is incrementally changed over finite intervals [40, 41]. The amount of performance delivered in the current time interval is set by evaluating

CPU activity in previous intervals, using a variety of averaging algorithms. This technique dynamically trades off throughput and energy with no knowledge of what process is being executed.

8.3. Algorithms and Compilers

Algorithms have always been tuned and optimized for maximum performance. These same techniques have a large impact on energy efficiency, as well. By using an algorithm implementation that requires fewer operations, both the throughput is increased, and less energy is consumed because the total amount of switched capacitance to execute the program has been reduced. A quadratic improvement in ETR can be achieved [5]. This same improvement holds for optimizing compilers which also try to minimize a program's dynamic run length, as demonstrated empirically in [30].

This does not always imply that the program with the smallest dynamic instruction count (path length) is the most energy efficient, since the switching activity per instruction must be evaluated. The work presented in [30] demonstrates through empirical measurements that the energy consumed per cycle is roughly constant, so that by minimizing execution time of the program, the energy consumption will be minimized. This implies that when considering the energy overhead for each cycle (e.g., clocks, instruction fetch, etc.), the key parameter to minimize is cycle count, and not instruction count.

9. Conclusions

Processors used in portable systems have a usage pattern in which the desirable throughput varies. Compute-intensive processes desire maximum throughput and high-latency processes desire less than maximum throughput to sufficiently complete. When no processes are pending, the processor idles and yields zero throughput. The important optimizations for these processors are to maximize throughput, which minimizes the response latency of the system, and minimize average energy consumed per operation, which maximizes the computation delivered over the life of the battery.

Metrics for energy efficiency have been defined for the three modes of computation that characterize typical processor operation. In particular, an energy efficiency metric, called is the Microprocessor Energy Throughput Ratio, or METR, was defined which

describes typical processor usage in a portable system. In addition to the energy consumed while computing it includes the energy consumed while idling which can dominate total energy consumption in user-interactive applications. When the idle energy consumption is negligible, METR degenerates to the Energy Throughput Ratio or ETR. Because of the variation of ETR with supply voltage, a better metric though less convenient, is the complete curve of energy/operation versus throughput for which ETR is just a linear approximation.

Four important design principles were developed to aid in energy-efficient design. High performance design was shown to be similar to energy-efficient design. Actually operating at high speeds, however, may not be energy efficient if idle energy consumption becomes dominant. Clock frequency reduction which is generally believed to be a method of improving battery life can actually be detrimental in some circumstances. However, if this reduction is coupled with an equivalent reduction in supply voltage and is performed dynamically depending on the performance requirements, then it becomes energy efficient.

A variety of new and existing design techniques were evaluated for energy efficiency. Some techniques that are low-energy such as 16-bit instructions were shown to not be energy efficient since the reduction in energy came at the expense of too much a reduction in throughput. Other techniques such as pipelining and cache sectoring were shown to be indispensable for energy-efficient design.

Decreasing the idle energy consumption is critical to the design of an energy-efficient processor and complete shut down of the clock while idling is optimal. If this cannot be accomplished, then it is imperative that the operating system implement a power down mode so that the idle power dissipation becomes independent of the computing power dissipation. Then the METR optimization will maximize the throughput delivered to the user in an energy-efficient manner. Otherwise, if idle power dissipation is proportional to the compute power dissipation, achieving energy-efficient operation requires the throughput to be minimized.

Acknowledgments

This research is sponsored by ARPA. We would like to thank Arthur Abnous, Andy Burstein, Dave Lidsky, Trevor Pering, Tony Stratakos, and the reviewers of this paper for their invaluable input.

References

1. S. Kunii, "Means of realizing long battery life in portable PCs," *Proceedings of the IEEE Symposium on Low Power Electronics*, pp. 12–13, Oct. 1995.
2. M. Culbert, "Low power hardware for a high performance PDA," *Proceedings of the Thirty-Ninth IEEE Computer Society International Conference*, pp. 144–147, March 1994.
3. T. Ikeda, "ThinkPad low-power evolution," *Proceedings of the IEEE Symposium on Low Power Electronics*, pp. 6–7, Oct. 1995.
4. A. Chandrakasan, A. Burstein, and R.W. Brodersen, "A Low power chipset for portable multimedia applications," *IEEE Journal of Solid State Circuits*, Vol. 29, pp. 1415–1428, Dec. 1994.
5. M. Horowitz, T. Indermaur, and R. Gonzalez, "Low-power digital design," *Proceedings of the IEEE Symposium on Low Power Electronics*, pp. 8–11, Oct. 1994.
6. D. Lidsky and J. Rabaey, "Early power exploration—A world wide web application," *Proceedings of the Thirty-Third Design Automation Conference*, June 1996.
7. T. Burd, *Low-Power CMOS Cell Library Design Methodology*, M.S. Thesis, University of California, Berkeley, UCB/ERL M94/89, 1994.
8. S. Sze, *Physics of Semiconductor Devices*, Wiley, New York, 1981.
9. H.J.M. Veendrick, "Short-circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits," *IEEE Journal of Solid-State Circuits*, pp. 468–473, Aug. 1984.
10. R. Muller and T. Kamins, *Device Electronics for Integrated Circuits*, Wiley, New York, 1986.
11. A. Chandrakasan, S. Sheng, and R.W. Brodersen, "Low-power CMOS digital design," *IEEE Journal of Solid State Circuits*, pp. 473–484, April 1992.
12. Advanced RISC Machines, Ltd., *ARM710 Data Sheet*, Technical Document, Dec. 1994.
13. Intel Corp., *Embedded Ultra-Low Power Intel486™ GX Processor, SmartDie™ Product Specification*, Dec. 1995.
14. A. Stratakos, S. Sanders, and R.W. Brodersen, "A low-voltage CMOS DC-DC converter for portable battery-operated systems," *Proceedings of the Twenty-Fifth IEEE Power Electronics Specialist Conference*, pp. 619–626, June 1994.
15. J. Bunda et al., "16-Bit vs. 32-Bit instructions for pipelined architectures," *Proceedings of the 20th International Symposium on Computer Architecture*, pp. 237–246, May 1993.
16. Advanced RISC Machines, Ltd., *Introduction to Thumb*, Developer Technical Document, March 1995.
17. J. Bunda, W.C. Athas, and D. Fussell, "Evaluating power implications of CMOS microprocessor design decisions," *Proceedings of the International Workshop on Low Power Design*, pp. 147–152, April 1994.
18. P. Freet, "The SH microprocessor: 16-Bit fixed length instruction set provides better power and die size," *Proceedings of the Thirty-Ninth IEEE Computer Society International Conference*, pp. 486–488, March 1994.
19. T. Burd and B. Peters, *A Power Analysis of a Microprocessor: A Study of an Implementation of the MIPS R3000 Architecture*, ERL Technical Report, University of California, Berkeley, 1994.
20. J. Montanaro et al., "A 160 MHz 32b 0.5W CMOS RISC microprocessor," *Proceedings of the Thirty-Ninth IEEE International Solid-State Circuits Conference—Slide Supplement*, pp. 170–171, Feb. 1996.

21. J. Bunda, *Instruction-Processing Optimization Techniques for VLSI Microprocessors*, Ph.D. Thesis, The University of Texas at Austin, 1993.
22. R. Gonzalez and M. Horowitz, "Energy dissipation in general purpose processors," *Proceedings of the IEEE Symposium on Low Power Electronics*, pp. 12–13, Oct. 1995.
23. D. Wall, *Limits of Instruction-Level Parallelism*, DEC WRL Research Report 93/6, Nov. 1993.
24. M. Johnson, *Superscalar Microprocessor Design*, Englewood, Prentice Hall, NJ, 1990.
25. M. Smith, M. Johnson, and M. Horowitz, "Limits on multiple issue instruction," *Proceedings of the Third International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 290–302, April 1989.
26. N. Jouppi and D. Wall, "Available instruction-level parallelism for superscalar and superpipelined machines," *Proceedings of the Third International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 272–282, April 1989.
27. P. Lowney et al., "The multiflow trace scheduling compiler," *The Journal of Supercomputing*, Kluwer Academic Publishers, Boston, Vol. 7, pp. 51–142, 1993.
28. R. Panwar and D. Rennels, "Reducing the frequency of tag compares for low power I-Cache design," *Proceedings of the International Symposium on Low Power Design*, pp. 57–62, April 1995.
29. S. Gary et al., "The powerPC 603 microprocessor: A low-power design for portable applications," *Proceedings of the Thirty-Ninth IEEE Computer Society International Conference*, pp. 307–315, March 1994.
30. V. Tiwari et al., "Instruction level power analysis and optimization of software," *Journal of VLSI Signal Processing*, this issue.
31. A. Chandrakasan, *Low Power Digital CMOS Design*, Kluwer Academic Publishers, Boston, 1995.
32. C. Nagendra et al., "A comparison of the power-delay characteristics of CMOS adders," *Proceedings of the International Workshop on Low Power Design*, pp. 231–236, April 1994.
33. T. Callaway and E. Swartzlander, "Optimizing arithmetic elements for signal processing," *VLSI Signal Processing*, Vol. 5, IEEE Special Publications, New York, pp. 91–100, 1992.
34. K. Chu and D. Pulfrey, "A comparison of CMOS circuit techniques: Differential cascode voltage switch logic versus conventional logic," *IEEE Journal of Solid State Circuits*, pp. 528–532, Aug. 1987.
35. T. Biggs et al., "A 1 Watt 68040-compatible microprocessor," *Proceedings of the IEEE Symposium on Low Power Electronics*, pp. 8–11, Oct. 1994.
36. J. Lundberg et al., "A 15-150 MHz all-digital phase-locked loop with 50-Cycle lock time for high-performance low-power microprocessors," *Proceedings of the Symposium on VLSI Circuits*, pp. 35–36, June 1994.
37. J. Lorch, *A Complete Picture of the Energy Consumption of a Portable Computer*, M.S. Thesis, University of California, Berkeley, 1995.
38. A. Chandrakasan, M. Srivastava, and R.W. Brodersen, "Energy efficient programmable computation," *Proceedings of the Seventh International Conference on VLSI Design*, pp. 261–264, Jan. 1994.
39. Intel Corp. and Microsoft Corp., *Advanced Power Management (APM): BIOS Interface Specification*, Technical Document, Feb. 1996.
40. M. Wieser et al., "Scheduling for reduced CPU energy," *Proceedings of the First USENIX Symposium on Operating Systems Design and Implementation*, pp. 13–23, Nov. 1994.
41. K. Govil, E. Chan, and H. Wasserman, "Comparing algorithms for dynamic speed-setting of a low-power CPU," *Proceedings First ACM International Conference on Mobile Computing and Networking*, pp. 13–25, Nov. 1995.
42. Integrated Device Technology, Inc., *Enhanced Orion 64-Bit RISC Microprocessor*, Data Sheet, Sept. 1995.



Thomas D. Burd received the B.S. and M.S. degrees in Electrical Engineering and Computer Science from the University of California, Berkeley in 1992 and 1994, respectively. He is currently working towards the Ph.D. degree at Berkeley.

His research emphasis is on energy-efficient processor system design. His research interests include energy-efficient design methodologies for portable systems, CMOS IC design, computer architecture, computer-aided design, and the world-wide web.

Thomas Burd is a member of Eta Kappa Nu and Tau Beta Pi.



Robert W. Brodersen received Bachelor of Science degrees in Electrical Engineering and in Mathematics from California State Polytechnic University, Pomona, California in 1966. In 1968 he received the Engineers and M. S. degrees from the Massachusetts Institute of Technology (MIT), Cambridge, and he received a Ph.D. in Engineering from MIT in 1972.

From 1972–1976, Brodersen was with the Technical Staff, Central Research Laboratory at Texas Instruments, Inc., Dallas. He joined the Electrical Engineering and Computer Science faculty at

the University of California at Berkeley in 1976, where he is currently a professor. In addition to teaching, Professor Brodersen is involved in research inclusive of new applications of integrated circuits, focused in the areas of low power design and wireless communications.

He has won conference best paper awards at Eascon (1973), International Solid State Circuits Conference (1975) and the European Solid State Circuits Conference (1978).

Professor Brodersen received the W.G. Baker award for the outstanding paper in the IEEE Journals and Transactions (1979), Best Paper Award in the Transactions on CAD (1985) and the Best Tutorial paper of the IEEE Communications Society (1992).

In 1978 Professor Brodersen was named the outstanding engineering alumnus of California State Polytechnic University. He became a Fellow of the IEEE 1982. He was co-recipient of the IEEE Morris Libermann award for "Outstanding Contributions to an Emerging Technology," in 1983. And he received Technical Achievement Awards from the IEEE Circuits and Systems Society in 1986 and in 1991 from the IEEE Signal Processing Society.

Professor Brodersen was elected a member of the National Academy of Engineering in 1988. In September of 1995, he was appointed the first holder of the John R. Whinnery Chair in Electrical Engineering at University of California, Berkeley.

Instruction Level Power Analysis and Optimization of Software

VIVEK TIWARI, SHARAD MALIK AND ANDREW WOLFE

Dept. of Electrical Engineering, Princeton University, Princeton, NJ 08540

MIKE TIEN-CHIEN LEE

Fujitsu Labs. of America Inc., 3350 Scott Blvd., Bldg. 34, Santa Clara, CA 95054

Received October 27, 1995; Revised April 29, 1996

Abstract. The increasing popularity of power constrained mobile computers and embedded computing applications drives the need for analyzing and optimizing power in all the components of a system. Software constitutes a major component of today's systems, and its role is projected to grow even further. Thus, an ever increasing portion of the functionality of today's systems is in the form of instructions, as opposed to gates. This motivates the need for analyzing power consumption from the point of view of instructions—something that traditional circuit and gate level power analysis tools are inadequate for. This paper describes an alternative, measurement based instruction level power analysis approach that provides an accurate and practical way of quantifying the power cost of software. This technique has been applied to three commercial, architecturally different processors. The salient results of these analyses are summarized. Instruction level analysis of a processor helps in the development of models for power consumption of software executing on that processor. The power models for the subject processors are described and interesting observations resulting from the comparison of these models are highlighted. The ability to evaluate software in terms of power consumption makes it feasible to search for low power implementations of given programs. In addition, it can guide the development of general tools and techniques for low power software. Several ideas in this regard as motivated by the power analysis of the subject processors are also described.

1. Motivation

The increasing popularity of power constrained mobile computers and embedded computing applications drives the need for analyzing and optimizing power in all the components of a system. This has forced an examination of the power consumption characteristics of all modules—ranging from disk-drives and displays to the individual chips and interconnects. Focussing solely on the hardware components of a design tends to ignore the impact of the software on the overall power consumption of the system. Software constitutes a major component of systems where power is a constraint. Its presence is very visible in a mobile computer, in the form of the system software and application programs running on the main CPU. But software also plays an even greater role in general digital applications, since an ever growing fraction of these

applications are now being implemented as embedded systems. Embedded systems are characterized by the fact that their functionality is divided between a hardware and a software component. The software component usually consists of application-specific software running on a dedicated processor, while the hardware component usually consists of application-specific circuits. In light of the above, there is a clear need for considering the power consumption in systems from the point of view of software. Software impacts the system power consumption at various levels of the design. At the highest level, this is determined by the way functionality is partitioned between hardware and software. The choice of the algorithm and other higher level decisions about the design of the software component can affect system power consumption in a big way. The design of system software, the actual application source code, and the process of translation into

machine instructions—all of these determine the power cost of the software component. In order to systematically analyze and quantify this cost, however, it is important to start at the most fundamental level. This is at the level of the individual instructions executing on the processor. Just as logic gates are the fundamental units of computation in digital hardware circuits, instructions can be thought of as the fundamental unit of software. This motivates the need for analyzing power consumption from the point of view of instructions. Accurate modelling and analysis at this level is the essential capability needed to quantify the power costs of higher abstractions of software, and to search the design space in software power optimizations.

In spite of its importance, very little previous work exists for analyzing power consumption from the point of view of software. Some attempts in this direction are based on architectural level analysis of processors. The underlying idea is to assign power costs to architectural modules such as datapath execution units, control units, and memory elements. In [1, 2] the power cost of a module is given by the estimated average capacitance that would switch when the given module is activated. More sophisticated statistical power models are used in [3, 4]. Activity factors for the modules are then obtained from functional simulation over typical input streams. Power costs are assigned to individual modules, in isolation from one another. Thus, these methods ignore the correlations between the activities of different modules during execution of real programs.

Since the above techniques work at higher levels of abstraction, the power estimates they provide are not very accurate. For greater accuracy, one has to use power analysis tools that work at lower levels of the design—physical, circuit, or switch level [5–7]. However, these tools are slow and impractical for analyzing the total power consumption of a processor as it executes entire programs. These tools also require the availability of lower level circuit details of processors, something that most embedded system designers do not have access too. This is also the reason why the power contribution of software and the potential for power reduction through software modification has either been overlooked or is not fully understood.

1.1. Instruction Level Power Analysis

The above problems can be overcome if the current being drawn by the CPU during the execution of a program is physically measured. An instruction level

power analysis technique based on physical measurements has recently been developed [8]. This technique helps in formulating instruction level power models that provide the fundamental information needed to evaluate the power cost of entire programs. This technique has so far been applied to three commercial, architecturally different processors—the Intel 486DX2 (a CISC processor), the Fujitsu SPARCite 934 (a RISC processor), and a Fujitsu proprietary DSP processor. The purpose of this paper is to provide a general description of the instruction level power analysis technique, based on its application for these three different processors.

The power models for the subject processors are described and interesting observations resulting from the comparison of these are highlighted. Other salient observations resulting from the analysis of these processors are summarized and these provide useful insights into power consumption in processors in general. Instruction level analysis of each processor helps to identify the reasons for variation in power from one program to another. These differences can then be exploited in order to search for low power alternatives for each program. The information provided by the instruction level analysis can guide higher-level design decisions like hardware-software partitioning and choice of algorithm. But it can also be directly used by automated tools like compilers, code generators and code schedulers for generating code targeted towards low power. Several ideas in this regard as motivated by the power analysis of the subject processors are also described.

2. Applications of Instruction Level Power Analysis

The previous section described the primary motivation for power analysis at the instruction level. There are several additional applications of this analysis and it is instructive to list the important ones here:

- The information provided by the analysis is useful in assigning an accurate power cost to the software component of a system. For power constrained embedded systems, this can help in verifying if the overall system meets its specified power budget.
- The most common way of specifying power consumption in processors is through a single number—the average power consumption. Instruction level analysis provides additional resolution about power consumption that cannot be captured through just

this one number. This additional resolution can guide the careful development of special programs that can be used as power benchmarks for more meaningful comparisons between processors.

- The proposed measurement based instruction level analysis methodology has the novel strength that it does not require knowledge of the lower level details of the processor. However, if micro-architectural details of the CPU are available, they can be related to the results of the analysis. This can lead to more refined models for software power consumption, as well as power models for the micro-architecture that may potentially be more accurate than circuit or logic simulation based models.
- The additional insight provided by an instruction-level power model also provides directions for modifications in processor design that lead to the most effective overall power reduction. Instructions can be evaluated both in terms of their power cost as well as frequency of occurrence in typical compiler or even hand-generated code. This combined information can be used to prioritize instructions that should be re-implemented to be less expensive in terms of power.

3. Analysis Methodology

This section describes in greater detail the measurement based technique that was referred to in the previous sections. This technique has so far been applied to three commercial processors:

- Intel 486DX2-S Series, 40 MHz, 3.3 V (referred to as the 486DX2). A CISC processor based on the x86 architecture. It is widely used in mobile and desktop PCs [9, 10].
- Fujitsu SPARCliteMB86934, 20 MHz, 3.3 V (referred to as the '934). A 32-bit RISC processor based on the SPARC architecture. It has been specially designed for embedded applications [11, 12].
- Fujitsu proprietary DSP, 40 MHz, 3.3 V (referred to as the DSP). A new implementation of an internal Fujitsu DSP architecture. It is used in several embedded DSP applications.

The basic idea that allows the use of the measurement based technique in the development of instruction level power models of given processors will also be described in this section. But first, we have to clarify the distinction between “power”, a term that we have

been using so far, and the term “energy”. The average power consumed by a processor while running a certain program is given by: $P = I \times V_{CC}$, where P is the average power, I is the average current and V_{CC} is the supply voltage. Power is also defined as the *rate at which energy is consumed*. Therefore, the energy consumed by a program is given by: $E = P \times T$, where T is the execution time of the program. This in turn is given by: $T = N \times \tau$, where N is the number of clock cycles taken by the program, and τ is the clock period.

Energy consumption is the primary concern for mobile systems, which run on the limited energy available in a battery. Power consumption, on its own, is of importance in applications where cooling and packaging costs are a concern. Energy consumption is the focus of attention in this paper. While we will attempt to maintain a distinction between the two terms, we may sometimes use the term power to refer to energy, in adherence to common usage. It should be noted, nevertheless, that power and energy are closely related, and the energy cost of a program is simply the product of its average power cost and its running time.

3.1. Current Measurement

As can be seen from the above discussion, the ability to measure the current drawn by the CPU during the execution of the program is essential for measuring the power/energy cost of the program. The different current measurement setups used in our work point to some of the options that can be used.

3.1.1. Board Based Measurements. In the case of the 486DX2 study, the CPU was part of a mobile personal computer evaluation board. The board was designed for current measurements and thus the power supply connection to the CPU was isolated from the rest of the system. A jumper on this connection allows an ammeter to be inserted in series with the power supply and the CPU. The ammeter used is a standard off the shelf, dual-slope integrating digital ammeter. Programs can be created and executed just as in a regular PC. If a program completes execution in a short time, a current reading cannot be visually obtained from the ammeter. To overcome this, the programs being considered are put in infinite loops. The current waveform will now be periodic. Since the chosen ammeter averages current over a window of time (100 ms), if the period of the current waveform is much smaller than this window, a stable reading will be obtained. The limitation of this

approach is that it cannot directly be used for large programs. But this is not a limitation, since the main use of this technique is for performing an instruction-level power analysis, and as discussed in the next section, short loops are adequate for this. This inexpensive current measurement approach works very well here. The current drawn by the external DRAM chips is also measured in a similar way. A similar measurement technique is also used in the case of the Fujitsu DSP. However, the DSP board had not been laid out with current measurements in mind. Therefore, the power pins of the CPU had to be lifted from the board in order to create an isolated power supply connection for them.

3.1.2. Tester Based Measurements. A suitable board was not available for the '934. Therefore, an alternative experimental setup, consisting of a processor chip and an IC tester machine was used. The program under consideration was first simulated on a VERILOG model of the CPU. This produces a trace file consisting of vectors that specify the exact logic values that would appear on the pins of the CPU for each half-cycle during the execution of the program. The tester then applies the voltage levels specified by the vectors to each input pin of the CPU. This recreates the same electrical environment that the CPU would see on a real board. The current drawn by the CPU is monitored by the tester using an internal digital ammeter.

It should be stressed that the main concepts described in this paper are independent of the method used to measure average current. The results of the above approaches have been validated by comparisons with other current measurement setups. But if sophisticated data acquisition based measurement instruments are available, the measurement method can be based on them, if so desired. Interestingly, instruction level power power analysis can be conducted even for unfabricated CPUs. Instead of physical current measurements, current estimates can be obtained through simulations on low level design models of the CPU.

4. Instruction Level Power Models

The instruction level analysis scheme described in the previous section has been applied to all three subject processors. Instruction level power models have been developed based on the results of these analyses. The key observations are summarized in this section. Separate references provide greater detail for each individual processor [13–15]. The basic components of each

power model are the same. The first component is the set of *base costs* of individual instructions. The other component is the power cost of *inter-instruction effects*, i.e., effects that involve more than one instruction. This includes the effect of circuit-state, and other effects like stalls and cache misses. These components of the power models are described below:

4.1. Instruction Base Costs

The primary component of the power models is the set of *base costs* of instructions. The base cost of an instruction can be thought of as the cost associated with the basic processing needed to execute the instruction. The experimental procedure used to determine this cost requires a program containing a loop consisting of several instances of the given instruction. The average current drawn during the execution of this loop is measured. The product of this current and V_{CC} is the base power cost of the instruction. The base power cost multiplied by the number of non-overlapped cycles needed to execute the instruction is proportional to its base energy cost. Table 1 presents a sample of the base costs of some instructions for the 486DX2 and the '934. The measured average current, number of cycles, and the base energy costs are also shown. The base energy costs are derived from the formula shown in Section 3.

There are some points to be noted with regard to the assignment of base costs to instructions:

- The definition of base costs follows the convention that the base costs of instructions should not reflect the power contribution of effects like stalls and cache misses. The programs used to determine the base costs have to be designed to avoid these effects. The power costs of these effects are modelled separately.
- The program loops used to determine the base costs should be large enough to overcome the impact of the jump instruction at the bottom of the loop. But they should not be so large so as to cause cache misses. Loop sizes of around 200 have been found to be appropriate.
- It has been observed that, in general, instructions with similar functionality tend to have similar base costs. This observation suggests that similar instructions can be arranged in classes, and a single average cost can be assigned to each class. Doing so speeds up the task of power analysis of the given processor. Table 2 illustrates the application of instruction grouping in the case of the DSP. The commonly

Table 1. Subset of the base cost table for the 486DX2 and the '934.

No.	Instruction	Intel 486DX2			Instruction	Fujitsu SPARClite '934		
		Current (mA)	Cycles	Energy ($\times 8.25 \times 10^{-8}$ J)		Current (mA)	Cycles	Energy ($\times 16.5 \times 10^{-8}$ J)
1	nop	276	1	2.27	nop	198	1	3.26
2	mov dx, [bx]	428	1	3.53	ld [%i0], %i0	213	1	3.51
3	mov dx, bx	302	1	2.49	or %g0, %i0, %i0	198	1	3.26
4	mov [bx], dx	522	1	4.30	st %i0, [%i0]	346	2	11.40
5	add dx, bx	314	1	2.59	add %i0, %o0, %i0, %i0	199	1	3.28
6	add dx, [bx]	400	2	6.60	mul %g0, %r29, %r27, %i0	198	1	3.26
7	jmp	373	3	9.23	srl %i0, 1, %i0, %i0	197	1	3.25

Table 2. Average base costs for instruction classes in the DSP.

	LDI	LAB	MOV1	MOV2	ASL	MAC
Current range (mA)	15.8–22.9	34.6–38.5	18.8–20.7	17.6–19.2	15.8–17.2	17.0–17.4
Average energy ($\times 8.25 \times 10^{-8}$ J)	0.160	0.301	0.163	0.151	0.136	0.142

used instructions have been grouped into 6 classes as shown.

- The base cost of an instruction can vary with the value and address of the operands used. While appropriate measurement experiments can give the exact cost if the operand and address values are known, in real situations these values are often unknown until runtime. The alternative is to assign a single average cost as the base cost of an instruction. This is justified, since extensive experimentation reveals that the variation in operands leads to only a limited variation in base costs. The DSP, which was the smallest of the three processors, exhibited the maximum variation. But even this was less than 10% for most instructions. Therefore, the inaccuracy due to the use of averages will be limited.

4.2. Effect of Circuit State

The switching activity, and hence, the power consumption in a circuit is a function of the change in circuit state resulting from changes in two consecutive sets of inputs. Now, during the determination of base costs, the same instruction executes each time. Thus, it can be expected that the change in circuit state between instructions would be less here, than in an instruction sequence in which consecutive instructions differ from one another. The concept of *circuit state overhead* for a pair of instructions is used to deal with this effect. Given any two instructions, the current for a loop consisting

of an alternating sequence of these instructions is measured. The difference between the measured current and the average base costs of the two instructions is defined as the circuit state overhead for the pair. For a sequence consisting of a mix of instructions, using the base costs of instructions almost always underestimates the actual cost. Adding in the average circuit state overhead for each pair of consecutive instructions leads to a much closer estimate.

While the above effect was observed for all the subject processors, it had a limited impact in the case of the 486DX2 and the '934. In the case of the 486DX2, the circuit state overhead varied in a restricted range, 5–30 mA, while most programs varied in the range of 300–420 mA. In the case of the '934, the overhead was less than 20 mA between integer instructions, and in the range 25–34 mA between integer and floating point instructions. In contrast, most programs themselves vary in the range 250–400 mA. The explanation for the limited impact may lie in the fact that in large complex processors like the 486DX2 and '934, a major part of the circuit activity is common to all instructions, e.g., the clocks, instruction prefetch, memory management, pipeline control, etc. Circuit state can certainly result in significant variation within certain control and data path modules. But the impact of the variation on the net power consumption of the processor will be masked by the much larger common cost.

It should also follow from the above that if instruction control and the data path constitute a larger fraction

Table 3. Average pairwise circuit state overhead costs for the DSP (in mA).

	LDI	LAB	MOV1	MOV2	ASL	MAC
LDI	3.6	13.7	15.5	6.3	10.8	6.0
LAB		2.5	1.9	12.2	20.9	15.0
MOV1			4.0	18.3	10.5	3.8
MOV2				25.6	26.7	22.2
ASL					3.6	8.0
MAC						12.5

of silicon, the impact of circuit state should be more visible. This indeed happens in the case for the DSP, a smaller, more basic processor. Table 3 shows the average overhead costs between different classes of instructions. Considering the fact that for most programs the average current is in the range 20–60 mA, several numbers in the table are significantly large.

4.3. Other Inter-Instruction Effects

The final component of the power model is the power cost of other inter-instruction effects that can occur in real programs. Examples are prefetch buffer and write buffer stalls [10], other pipeline stalls, and cache misses. Base costs of instructions do not reflect the impact of these inter-instruction effects. Separate costs need to be assigned to these effects through specific current measurement experiments. The basic idea is to write programs where these effects occur repeatedly. This helps to isolate the power costs of these effects. For example, in the case of the 486DX2, an average cost of 250 mA per stall cycle was determined for prefetch buffer stalls [8]. The average cost for a cache miss was 216 mA per cache miss cycle. Multiplying the power cost of each kind of stall or cache miss by the number of cycles taken for each, gives the energy cost of these effects.

4.4. Overall Instruction Level Power Model

The previous subsections described the basic components of the instruction level power models of the subject processors. These models form the basis of estimating the energy cost of entire programs. For any given program, P , its overall energy cost, E_P , is given by:

$$E_P = \sum_i (B_i \times N_i) + \sum_{i,j} (O_{i,j} \times N_{i,j}) + \sum_k E_k \quad (1)$$

The base cost, B_i , of each instruction, i , weighted by the number of times it will be executed, N_i , is added up to give the base cost of the program. To this the circuit state overhead, $O_{i,j}$, for each pair of consecutive instructions, (i, j) , weighted by the number of times the pair is executed, $N_{i,j}$, is added. The energy contribution, E_k , of the other inter instruction effects, k , (stalls and cache misses) that would occur during the execution of the program, is finally added.

The base cost and overhead values are obtained as shown in the previous sections. As described in Section 4.2, circuit state varies in a limited range in the case of the 486DX2 and the '934. This suggests a more efficient and yet fairly accurate way of modelling this effect for these processors. Instead of a table of pairwise overhead values, a *constant* value is used for all instruction pairs. For, e.g., 15 mA and 18 mA in the case of the 486DX2 and the '934 respectively. A table is still needed for the DSP, since this effect has a significant impact and greater variation, in the case of this processor.

The other parameters in the above formula vary from program to program. The execution counts N_i and $N_{i,j}$ depend on the execution path of the program. This is dynamic, run-time information. In certain cases it can be determined statically but in general it is best obtained from a program profiler. For estimating E_k , the number of times pipeline stalls and cache misses occur has to be determined. This is again dynamic information that can be statically predicted only in certain cases. In general, this information is obtained from a program profiler and cache simulator. A software power/energy estimation framework based on the above model is described in [8].

The 486DX2 program shown in Table 4 will be used to illustrate the basic elements of the estimation process. The program has three basic blocks as shown in the figure (A basic block is defined as a contiguous section of code with exactly one entry and exit point. Thus, every instruction in a basic block is executed as many times as the basic block). The average current and the number of cycles for each instruction are provided in two separate columns. For each basic block, the two columns are multiplied and the products are summed up over all instructions in the basic block. This yields a value that is proportional to the base energy cost of one instance of the basic block. The values are 1713.4, 4709.8, and 2017.9, for B1, B2, and B3 respectively. B1 is executed once, B2 four times, and B3 once. The `jmp main` statement has been inserted to put the program

Table 4. Illustration of the estimation process.

Program	Current(mA)	Cycles
;Block B1		
main:		
mov bp,sp	285.0	1
sub sp,4	309.0	1
mov dx,0	309.8	1
mov word ptr -4[bp],0	404.8	2
;Block B2		
L2:		
mov si,word ptr -4[bp]	433.4	1
add si,si	309.0	1
add si,si	309.0	1
mov bx,dx	285.0	1
mov cx,word ptr _a[si]	433.4	1
add bx,cx	309.0	1
mov si,word ptr _b[si]	433.4	1
add bx,si	309.0	1
mov dx,bx	285.0	1
mov di,word ptr -4[bp]	433.4	1
inc di,1	297.0	1
mov word ptr -4[bp],di	560.1	1
cmp di,4	313.1	1
j1 L2	405.7(356.9)	3(1)
;Block B3		
L1:		
mov word ptr _sum,dx	521.7	1
mov sp,bp	285.0	1
jmp main	403.8	3

in an infinite loop. Cost of the `j1 L2` statement is not included in the cost of `B2` since its cost is different depending on whether the jump is taken or not. It is taken 3 times and not taken once. Multiplying the base cost of each basic block by the number of times it is executed and adding the cost of the unconditional jump `j1 L2`, we get a number proportional to the total energy cost of the program. Dividing it by the estimated number of cycles (72) gives us an average current of 369.1 mA. Adding the circuit state overhead offset value of 15.0 mA we get 384.0 mA. This program does not have any stalls, and thus, no further additions to the estimated cost are required. If in the real execution of this program, some cold-start cache misses are expected, their energy overhead will have to be added. The actual measured average current is 385.0 mA. Thus, the estimate is within 0.26% of the measured value.

An interesting extension of the above ideas is the development of power profilers for given processors. The above instruction level power model suggests that this can easily be done by enhancing existing performance based profilers with the power costs of instructions and inter-instruction effects. Using this data, the profilers can generate a cycle by cycle profile of the power consumption of given programs.

When average values are used for base costs, etc., the accuracy of the energy estimate given by the model described in Eq. (1) is limited to some extent by the range of variation in the average and the actual costs. However, the accuracy of the energy estimate is primarily limited by the accuracy in determining the dynamic information regarding the program. Other than this the model is very accurate. For example, for the 486DX2 and the '934, for instruction sequences where the dynamic information was fully known, the maximum difference between the estimated and the measured cost was less than 3%.

It should also be mentioned that in certain applications, e.g., speech processing, some statistical characteristics of the input data are known [16]. Incorporating this knowledge into the power model can lead to more accurate power estimates. This may be specially beneficial in the case of the DSP, which shows greater sensitivity to data based power variations than the other two processors.

4.5. Impact of Internal Power Management

An examination of the base costs of the '934 in Table 1 reveals that the cost for different operations like OR, SHIFT, ADD, or MULTIPLY does not show much of a variation. It may well be the case that the differences in the circuit activity for these instructions are much less relative to the circuit activity common to all instructions. Thus, these differences may not be reflected in the comparisons of the overall current cost. Nevertheless, the almost complete lack of variation is somewhat counter-intuitive. For instance, it is expected that the logic for an OR should be much less than that for a MULTIPLY, thus leading to some variation in the overall current drawn for these instructions. The reason for the similarity of the costs most likely has to do with the way ALUs are traditionally designed. A common bank of inputs feeds all the different ALU modules, and thus all the modules switch and consume power, even though on any given cycle, only one of the modules computes useful results. This observation motivates

a power reduction technique called *guarded evaluation* [17]. Under this, the modules that are not needed for the current ALU operation are disabled. Thus, it can be expected that if this technique were to be used, the power costs of the different ALU operations will show a variation depending upon their functionality.

The above idea is actually an extension of the principles of *power management*, which refers to the dynamic shutting down of modules that are not needed for a given computation. Power management is gaining popularity as an effective power reduction technique, and has been implemented in recent processors like the Low Power Pentium, PowerPC 603 [18], and others [19]. Logic level techniques based on the power management idea have also been proposed recently [17, 20, 21]. An aggressive application of power management in a processor may have interesting ramifications for the instruction level power analysis of the processor. First, the base costs of different instructions may show greater variation than they do now. Variations due to differences in data may also increase, both due to the presence of data dependent power management features and due to a general decrease in the overall power consumption. The overall reduction in power may also make the effect of circuit state overhead more prominent. Some power management features may get activated depending on the occurrence of specific sequences of instructions, and these may require special handling.

A related effect was observed in the case of the DSP. The inputs to the on-chip multiplier on the DSP are latched. Thus, the change in the circuit state in the multiplier occurs only for multiply instructions. This change in circuit state is observed even if multiply instructions are not consecutive, and due to the relatively large power contribution of the multiplier for this processor, this effect can actually get reflected in the power cost of instruction sequences. An accurate way to deal with the effect is to add in the exact circuit state overhead for consecutive multiply instructions, even when they are not adjacent in the instruction execution order. An easier but approximate alternative is to enhance the base cost of the multiply instruction with an average value for this overhead. This assumes an unknown state for the multiplier on each multiply instruction, but eliminates the need to keep track of the preceding multiply. While this effect was observed only in the specific case of multiply instructions in the DSP, and for none of the larger processors, aggressive use of power management may mean that the basic power

model described in Section 4.4 may need to be adapted in certain cases. And finally, if the mechanism of the major power management features is not described in public domain data books, greater experimental effort may be needed in order to conduct a comprehensive power analysis of the processors. These issues will be investigated further as part of future work.

5. Software Energy Optimization Techniques

It is generally accepted that there is a great potential for energy reduction through modification of software. However, very little has been done to effectively exploit this potential. This has largely been due to the lack of practical techniques for analysis of software energy consumption. The instruction level analysis technique described in the previous sections overcomes this deficiency. Application of this technique provides the fundamental information that can guide the development of energy efficient software. It also helps in the identification of sources of energy reduction that can then be exploited by software development tools like compilers and code generators and schedulers. Several ideas in this regard as motivated by our analysis of the subject processors are described below. Some of these ideas have general applicability for most processors. Others are based on specific architectural features of the subject processors.

5.1. Reducing Memory Accesses

An inspection of energy costs reveals an important fact that holds for all three processors—instructions that involve memory accesses are much more expensive than instructions that involve just register accesses. For example, for the 486DX2, instructions that use register operands cost in the vicinity of 300 mA per cycle. In contrast, memory reads cost upwards of 400 mA, even in the case of a cache hit. Memory writes cost upwards of 530 mA. Every memory access can also potentially lead to cache misses, misaligned accesses, and stalls. These increase the number of cycles needed to complete the access, and the energy cost goes up by a corresponding factor. The energy consumption in the external memory system adds an additional energy penalty for cache misses, and for each write in case of write-through caches (as in the 486DX2 and the '934).

These observations point to the large energy savings that can be attained through a reduction in the number

of memory accesses. This motivates the need for development of optimizations to achieve this reduction at all levels of the software design process, starting from higher level decisions down to the generation of the final assembly code. At the higher level, some ideas for control flow transformations [22] and data structure design for signal processing applications have been proposed [23] by other researchers. Our experiments provide physical data to analyze these ideas quantitatively.

Attempts can also be made to reduce memory operations during generation of the final code. This can be done automatically if compilers are used, but the basic ideas are applicable even if the assembly code is created manually. This is the level that we explored further using the instruction level analysis technique. The technique provides the guiding information as described above, and is also used to quantify the effectiveness of different ideas.

During compilation, the most effective way of reducing memory operands is through better utilization of registers. The potential of this idea was demonstrated through some experiments in the case of the 486DX2 [24] and the results are also shown in Table 5. The first program in the table is a *heapsort* program ("sort" [25]). `hlcc.asm` is the assembly code for this program generated by `lcc`, a general purpose ANSI C compiler [26]. The sum of the observed average CPU and memory currents is given in the table above. The program execution times and overall energy costs are also reported. The generated code for the main routine is shown on the left in Table 9. While `lcc` generates good code in general, it often makes tradeoffs in favor of faster compilation time and lesser compiler complexity. For example, register allocation is performed only for temporary variables. Local and global variables for the program are normally not

allocated to registers. Optimizations were performed by hand on this code, in order to facilitate a more aggressive use of register allocation. The final code is shown on the right in Table 9. The energy results are shown in Column 3 of Table 5. There is a 40% reduction in the CPU and memory energy consumption for the optimized code. Results for another program (`circle`) are also shown in Table 5. Large energy reduction, about 33%, is observed for this program too.

It should be noted that register allocation has been the subject of research for several years due to its role in traditional compilation. The results of our study show that this research also has an immediate application in compilation for low energy. Further, it also motivates the aggressive use of known techniques, and the development of newer techniques in this direction.

On a related note, an interesting RISC vs. CISC power tradeoff is suggested by the following observation. In the 486DX2, a memory read that hits the cache is about 100 mA more expensive than a register read. This difference is only 10 mA in the case of the '934 (compare entries 2 and 3 for the two processors in Table 1). The smaller difference can be attributed to the larger size of the register file in the '934, which leads to a higher power cost for accessing registers. The '934 has 136 registers, as opposed to only 8 in the 486DX2. A large register file is characteristic of RISC architectures. Availability of more registers can help to reduce memory accesses, leading to power reduction. But on the other hand, a larger register file also means that each register access itself will be costlier.

5.2. Energy Cost Driven Code Generation

Code generation refers to the process of translating a high-level problem specification into machine code. This is either done automatically through compilers, or in certain design situations, it is done by hand. In either case, code generation involves the selection of the instructions to be used in the final code, and this selection is based on some cost criterion. The traditional cost criteria are either the size or the running time of the generated code. The main idea behind energy cost driven code generation is to select instructions based on their energy costs instead. The instruction energy costs are obtained from the analysis described in the previous sections.

An energy based code generator was created for the 486DX2 using this idea. An existing tree pattern based code generator selected instructions based on the

Table 5. Energy optimization of `sort` and `circle` for the 486DX2.

Program <code>sort</code>	<code>hlcc.asm</code>	<code>hfinal.asm</code>
Avg. current (mA)	525.7	486.6
Execution time (μ sec)	11.02	7.07
Energy ($\times 10^{-6}$ J)	19.12	11.35
Energy reduction		40.6%
Program <code>circle</code>	<code>clcc.asm</code>	<code>cfinal.asm</code>
Avg. current (mA)	530.2	514.8
Execution time (μ sec)	7.18	4.93
Energy ($\times 10^{-6}$ J)	12.56	8.37
Energy reduction		33.4%

number of cycles they took to execute. It was modified to use the energy costs of the instructions instead. Interestingly, it was found that the energy and cycle based code generators produced very similar code.

This observation provides quantitative evidence for a general trend that was observed for all the subject processors. This is that energy and running times of programs track each other closely. It was consistently observed that the difference in average current for sequences that perform the same function is never large enough to compensate for any difference in the number of cycles. Thus, the shortest sequence is also invariably the least energy sequence. Since this observation holds for all the subject processors, each of which represents a distinct architecture style, it is reasonable to expect that it will also hold for most other processors that exist today.

This is a very important observation, and something that has not been addressed in previous literature. It can be considered as empirical justification for a powerful guideline for software energy reduction for today's processors—as a first step towards energy reduction, do what needs to be done to improve performance. Potentially large energy reductions can be obtained if this observation is used to guide high-level decisions like hardware-software partitioning and choice of algorithm. It should be noted that this guideline is motivated and justified by the results of our instruction level analysis. Without the physical corroboration provided by the results, we would not have been able to put forth this guideline.

It also bears mentioning that it is possible that there may be certain application specific processors where this observation may not hold in general. It is also possible that aggressive use of use of power management and other low power design optimizations may also lead to situations where the fastest code may not always be the least energy code. While these cases remain to be identified, code generation based on energy costs will be useful in its own right for these cases.

5.3. Instruction Reordering for Low Power

Reordering of instructions in order to reduce switching between consecutive instructions is a method for energy reduction that does not involve a corresponding reduction in the number of cycles. An instruction scheduling technique based on this idea has been proposed in another work [27]. In this, instructions are scheduled in order to minimize the estimated switching

Table 6. Effect of instruction reordering in the '934.

No.	Instruction	Register contents
1	<code>fmul s %f8,%f4,%f0</code>	<code>%f8=0,%f4=0</code>
2	<code>andcc %g1,0xaaa,%i0</code>	<code>(%g1=0x555)</code>
3	<code>faddd %f10,%f12,%f14</code>	<code>(%f10=0x123456,%f12=0xaaaaaa)</code>
4	<code>ld [0x555],%o5</code>	
5	<code>sll %o4,0x7,%o6</code>	<code>(%o4=0x707)</code>
6	<code>sub %i3,%i4,%i5</code>	<code>(%i3=0x7f,%i4=0x44)</code>
7	<code>or %g0,0xff,%i0</code>	
	Sequence	Current (mA)
a	1, 2, 3, 4, 5, 6, 7	227.5
b	1, 3, 5, 7, 2, 4, 6	224
c	1, 4, 7, 2, 5, 3, 6	226
d	2, 3, 7, 6, 1, 5, 4	228
e	5, 3, 1, 4, 6, 7, 2	223.5

in the control path of an experimental RISC processor. Our experiments, however, indicate that in terms of net energy reduction for the entire processor, instruction reordering may not always be effective. It has been observed to have very limited impact in the case of the 486DX2 and the '934. Table 6 illustrates this with an example. As can be seen, different reordering of the given sequence of instructions lead to very little change in the measured average current. The idea behind reordering instructions can be seen as an attempt to reduce the overall circuit state overhead between consecutive instructions. But as seen in Section 4.2, this quantity is bounded in a small range and does not show much variation in the 486DX2 and the '934.

In the case of the DSP, however, this quantity is more significant and does show relatively greater variation (refer to Section 4.2 and Table 3). Thus, instruction reordering is more beneficial for this processor. A scheduling algorithm that uses the measured overhead costs was developed for this processor [15]. The data in Table 7 illustrates the effectiveness of this algorithm. This table shows the impact of different software energy optimization techniques that are applicable for the DSP ("packing" and "swapping" will be discussed later). Five standard signal processing programs were used for the experiment. `FJex1` and `FJex2` are real Fujitsu applications for vector preprocessing. `LP_FIR60` is a length-60 linear phase FIR filter. `IIR4` is a fourth-order direct form IIR filter, and `FFT2` is a radix-2 decimation-in-time FFT butterfly. The

Table 7. Results for different energy optimization techniques for the DSP.

Benchmark		Original	Packing	Scheduling	Swapping
FJex1	Energy ($\times 10^{-8}$ J)	2.79	2.46	2.12	
	Energy reduction		12.0%	24.0%	
FJex2	Energy ($\times 10^{-8}$ J)	3.91	3.14	2.83	
	Energy reduction		19.7%	27.7%	
LP_FIR60	Energy ($\times 10^{-8}$ J)	57.60	30.80		25.60
	Energy reduction		46.6%		55.6%
IIR4	Energy ($\times 10^{-8}$ J)	10.10	7.47	6.78	6.37
	Energy reduction		26.3%	33.1%	37.2%
FFT2	Energy ($\times 10^{-8}$ J)	9.59	9.35	8.97	8.64
	Energy reduction		3.4%	7.4%	10.9%

last three programs were taken from the TMS320 embedded DSP examples in [28] and translated into native code for the target DSP processor. Column 2 shows the initial energy consumption of the programs. Columns 3, 4, and 5 show the energy consumption and the overall percent energy reduction after the application of each technique. The three techniques are applied one after the other, from left to right. The percent by which the values in Column 4 are lower than those in Column 3 quantifies the effectiveness of instruction scheduling alone. As shown, up to 14% reduction in energy (for FJex1) has been observed using this algorithm. Table 10 shows the initial code for IIR4, and the final code after all three optimizations. For this example, instruction scheduling alone leads to a 9.3% reduction in energy.

Switching on the address and data pins is a specific manifestation of the effect of circuit state. Software transformations to reduce this switching are believed to be a possible energy reduction method. The large capacitance associated with these pins can indeed lead to greater current when these pins switch. However, there are some practical considerations that should be noted in this regard. First, the presence of on-chip caches greatly reduces external traffic. In addition the traffic becomes unpredictable making it harder to model the correlation between consecutive external accesses. Second, real systems often use external buses and memories that are slower than the CPU, necessitating the use of “wait states”. This implies that, on the average, pins switch less often. Thus, for instance, in the case of the 486DX2 system, switching on the address and data pins had only a limited impact for most programs—even for back to back writes, the impact of greater switching on the address lines

was less than 5%. Finally, even for processors without caches, it is difficult to model this switching for general programs. The necessary information is fully available only at run-time. However, reasonable models may be feasible for more structured applications like signal processing, and this bears further investigation.

5.4. Processor Specific Optimizations

Instruction level power analysis of a given processor can lead to the identification of features specific to that processor that can then be exploited for energy efficient software. We identified such specific features for each of the subject processors. Some of the more noteworthy examples are briefly described below.

5.4.1. Instruction Packing. The DSP has a special architectural feature called *instruction packing* that allows an ALU instruction and a memory data transfer instruction to be packed into a single instruction. The packed instruction executes in one cycle, as opposed to a total of two for the sequence of two unpacked instructions. Interestingly, we found that the use of packing always leads to large energy reductions, even though a packed instruction represents the same functionality as a sequence of two unpacked instructions. Figure 1 illustrates this graphically. The average current for a certain sequence of n packed instructions is only marginally greater than for the corresponding sequence of $2n$ unpacked instructions. Therefore, since the unpacked instructions complete in twice as many cycles, their energy consumption (proportional to the area under the graph) is almost twice that of the packed instructions. Thus, instructions should be packed as much as possible.

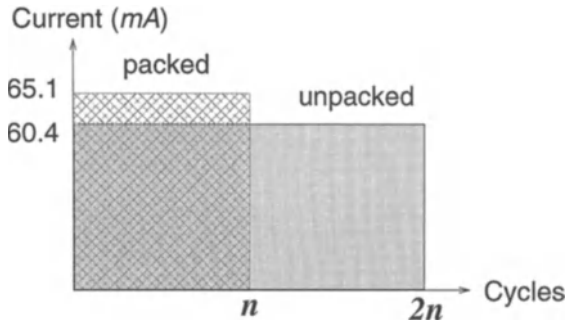


Figure 1. Comparison of energy consumption for packed and unpacked instructions.

Table 10 illustrates the application of packing for the example IIR4. Instructions with two opcodes separated by a colon are packed instructions, e.g., MUL:LAB. The use of packing leads to large energy savings for real programs (e.g., 26% for IIR4 and 47% for LP_FIR60, as shown in Column 3 of Table 7). The substantial savings attainable also make it worthwhile to develop program transformation and scheduling techniques that can lead to better utilization of instruction packing.

5.4.2. Dual Memory Loads. The Fujitsu DSP has two on-chip data memory banks. A special dual load instruction can transfer two operands, one from each memory, to registers in one cycle. The same task can also be attained by two single load instructions over two cycles. However, we found that the average current for the latter was only marginally lower, and thus, doubling of execution cycles implies a corresponding increase in energy consumption. The large energy difference also justifies the use of memory allocation techniques that can lead to better utilization of dual loads. A static memory allocation technique based on *simulated annealing* was developed for this purpose [29]. Application of this technique led to a 47% energy reduction over the case where data is assigned to only bank for LP_FIR60. Our observations also suggest that other memory allocations techniques developed from the point of view of improving performance can also find direct application for energy reduction [30].

It should be noted that both the above features are not unique to the Fujitsu DSP, but are also provided by several other popular DSP processors, e.g., the Motorola 56000 series. The above observations are likely to be valid for these other processors too.

5.4.3. Swapping Multiplication Operands. The results of our analysis of the Fujitsu DSP indicate that the on-chip multiplier on this processor is a major source of energy consumption for signal processing applications. This motivated a more detailed analysis of power consumption for multiply instructions. It was discovered that similar variations in the values of the two operands lead to different degrees of variations in the power consumption of multiply operations. This is reasonable, since the multiplier is based on the Booth multiplication algorithm, which treats the two operands in very different ways. We found that an appropriate swapping of the operands, in order to exploit this asymmetry, leads to up to 30% reduction in multiplication energy costs. This can translate into appreciable energy reduction for entire programs, as shown in Column 5 of Table 7. For example, for LP_FIR60, the use of operand swapping reduces the energy consumption of the packed code by an additional 16%.

5.4.4. Software Controlled Power Management. The '934 provides a software mechanism for powering down parts of the CPU. By setting appropriate bits in a system control register through a specified sequence of instructions, the clock inputs to certain modules can be enabled or disabled. We were able to quantify the effectiveness of this mechanism by using our analysis technique. Table 8 shows the measured power reductions attained for an OR instruction, when some combinations of the SDRAM interface (SDI), DMA module, floating-point unit (FPU), and floating-point FIFOs are powered down. It is evident from the results, that power management, i.e., powering down of unneeded modules can lead to significant power savings. It should also be noted that *automatic* power management will be a more effective and more generally applicable power reduction technique. The energy overhead associated

Table 8. Software controlled power management in the '934.

Instruction: or %i0, 0, %i0		
Units powered down	Current (mA)	% Energy reduction
None	198	0.0
SDI	185	6.6
FPU	176	11.1
DMA, FPU	172	13.1
FIFO, FPU	163	17.7
SDI, DMA, FIFO, FPU	154	22.2

Table 9. 486DX2 software energy optimization example: sort.c.

Compiler generated code			Energy optimized code		
sort:		mov ebx,dword ptr [ebx]	sort:		
push ebx		mov edi,dword ptr 4[edi][esi]	push ebp		
push esi		cmp ebx,edi	mov edi,dword ptr 08H[esp]		
push edi		jge L14	mov esi,edi		
push ebp		mov edi,dword ptr -4[ebp]	sar esi,1		
mov ebp,esp		lea edi,1[edi]	inc esi		
sub esp,24		mov dword ptr -4[ebp],edi	mov ebp,esi		
mov edi,dword ptr 014H[ebp]		mov ecx,edi			
mov esi,1		L14:	L3:		
mov ecx,esi		mov edi,dword ptr -12[ebp]	cmp ebp,1		
mov esi,edi		mov esi,dword ptr -4[ebp]	jle L7		
sar esi,c1		lea esi,[esi*4]	dec ebp		
lea esi,1[esi]		mov ebx,dword ptr 018H[ebp]	mov esi,dword ptr 0cH[esp]		
mov dword ptr -20[ebp],esi		add esi,ebx	mov edi,dword ptr [edi*4][esi]		
mov dword ptr -8[ebp],edi		mov esi,dword ptr [esi]	mov ebx,edi		
L3:		cmp edi,esi	jmp L8		
mov edi,dword ptr -20[ebp]		jge L16	L7:		
cmp edi,1		mov edi,2	mov edi,dword ptr 0cH[esp]		
jle L7		mov esi,dword ptr 018H[ebp]	mov esi,dword ptr 4[edi]		
mov edi,dword ptr -20[ebp]		mov ebx,dword ptr -16[ebp]	mov ebx,dword ptr [ecx*4][edi]		
sub edi,1		mov ecx,edi	mov dword ptr [ecx*4][edi],esi		
mov dword ptr -20[ebp],edi		sal ebx,c1	dec ecx		
lea edi,[edi*4]		add ebx,esi	cmp ecx,1		
mov esi,dword ptr 018H[ebp]		mov ecx,dword ptr -4[ebp]	jne L8		
add edi,esi		mov dword ptr -24[ebp],ecx	mov dword ptr 4[edi],ebx		
mov edi,dword ptr [edi]		mov ecx,edi	jmp L2		
mov dword ptr -12[ebp],edi		mov edi,dword ptr -24[ebp]	L8:		
jmp L8		sal edi,c1	mov edi,ebp		
L7:		add edi,esi	mov edx,edi		
mov edi,dword ptr 018H[ebp]		mov edi,dword ptr [edi]	add edi,edi		
mov esi,dword ptr -8[ebp]		mov dword ptr [ebx],edi	mov eax,edi		
lea esi,[esi*4]		mov edi,dword ptr -4[ebp]	jmp L12		
add esi,edi		mov dword ptr -16[ebp],edi	L11:		
mov ebx,dword ptr [esi]		mov esi,edi	jge L14		
mov dword ptr -12[ebp],ebx		add esi,edi	mov esi,dword ptr 0cH[esp]		
mov edi,dword ptr 4[edi]		mov dword ptr -4[ebp],esi	mov edi,dword ptr [eax*4][esi]		
mov dword ptr [esi],edi		jmp L12	cmp edi,dword ptr 4[eax*4][esi]		
mov edi,dword ptr -8[ebp]		L16:	jge L14		
sub edi,1		mov edi,dword ptr -8[ebp]	inc eax		
mov dword ptr -8[ebp],edi		lea edi,1[edi]	L14:		
cmp edi,1		mov dword ptr -4[ebp],edi	mov esi,dword ptr 0cH[esp]		
jne L8		L12:	cmp ebx,dword ptr [eax*4][esi]		
mov edi,dword ptr 018H[ebp]		mov edi,dword ptr -4[ebp]	jge L16		
mov esi,dword ptr -12[ebp]		mov esi,dword ptr -8[ebp]	mov edi,dword ptr [eax*4][esi]		
mov dword ptr 4[edi],esi		cmp edi,esi	mov dword ptr [edx*4][esi],edi		
jmp L2		jle L11	mov edx,eax		

(Continued on next page)

Table 9. (Continued.)

Compiler generated code		Energy optimized code	
L8:	mov edi,dword ptr -16[ebp]	add	eax,ecx
mov	edi,dword ptr -20[ebp]	lea	edi,[edi*4]
mov	dword ptr -16[ebp],edi	mov	esi,dword ptr 018H[ebp]
lea	edi,[edi*2]	add	edi,esi
mov	dword ptr -4[ebp],edi	mov	esi,dword ptr -12[ebp]
jmp	L12	mov	dword ptr [edi],esi
L11:	jmp L3	cmp	eax,ecx
mov	edi,dword ptr -4[ebp]	L2:	jle L11
mov	esi,dword ptr -8[ebp]	mov	esp,ebp
cmp	edi,esi	pop	ebp
jge	L14	pop	edi
lea	edi,[edi*4]	pop	esi
mov	esi,dword ptr 018H[ebp]	pop	ebx
mov	ebx,edi	ret	ret
add	ebx,esi		

Table 10. DSP software energy optimization example: IIR4.

Portion of original code		After energy optimizations	
LDI	coefa,X0	LDI	coefa,X0
LDI	xn,X2	LDI	coefb,X1
MOV	(X2),C	LDI	xn,X2
LDI	datanm1,X3	LDI	datanm1,X3
LAB	(X3+1),(X0+1)	LDI	datan,X4
MUL:		MOV	(X2),C
LAB	(X3+1),(X0+1)	LAB	(X0+1),(X3+1)
MSMC:		MUL:LAB	(X0+1),(X3+1)
LAB	(X3+1),(X0+1)	MSMC:LAB	(X0+1),(X3+1)
LDI	datan,X4	MSMC:LAB	(X0+1),(X3+1)
MSMC:		MSMC:	
LAB	(X3+1),(X0+1)	MSMC:	
MSMC:		MOV	C,(X4)
LDI	coefb,x1	RESC:LAB	(X1+1),(X4+1)
MSMC:		MUL:LAB	(X1+1),(X4+1)
MOV	C,(X4)	MSMC:LAB	(X1+1),(X4+1)
RESC:		MSMC:LAB	(X1+1),(X4+1)
LAB	(X4+1),(X1+1)	MSMC:	
MUL:		MSMC:	
LAB	(X4+1),(X1+1)	MOV	C,(X4)
MSMC:			
LAB	(X4+1),(X1+1)		
MSMC:			
LAB	(X4+1),(X1+1)		
MSMC:			
MSMC:			
MOV	C,(X4)		

with power management will be much less if it is controlled by logic internal to the CPU, rather than through a sequence of instructions. The temporal resolution of the power management strategy will also be much finer, since it can then be applied on a cycle by cycle basis.

6. Future Directions

There are several directions in which we would like to extend this work. The first of these would be to extend the analysis methodology to processors whose architecture and implementation style is significantly different from the processors studied here. We would specially like to analyze processors that are based on superscalar and VLIW architectures. These seem to be the architectures of choice for high performance processors in the near future, and with ever increasing integration and clock frequencies, the power problem will become even more acute for these processors. We would also like to continue to work on avenues for power reduction through software optimization, and development of automated tools where applicable. The results of our work show that a number of ideas from existing literature on traditional software optimization can be used here, but new techniques will also be developed. The ability to evaluate the power cost of the software component of an embedded system can also be used as a first step towards ideas and tools for hardware-software co-design for low power. Finally, the software perspective is essential in understanding the power consumption in processors. This additional

perspective can help guide us in the search for more power efficient architectures, and this issue will be explored in the future.

7. Conclusions

The increasing role of software in today's systems demands that energy consumption be studied from the perspective of software. This paper describes a measurement based instruction level power analysis technique that makes it feasible to effectively analyze software power consumption. The main observations resulting from the application of this technique to three commercial processors were presented here. These provide useful insights into the power consumption in processors. They also illustrate how a systematic analysis can lead to the identification of sources of software power consumption. These sources can then be targeted through suitable software design and transformation techniques. The ability to quantitatively analyze the power consumption of software makes it possible to deal with the overall system power consumption in an integrated way. A unified perspective allows for the development of more effective power reduction techniques that are applicable for the entire system.

References

1. T. Sato, M. Nagamatsu, and H. Tago, "Power and performance simulator: ESP and its application for 100MIPS/W class RISC design," in *Proceedings of 1994 IEEE Symposium on Low Power Electronics*, San Diego, CA, Oct. 1994, pp. 46–47.
2. P.W. Ong and R.H. Yan, "Power-conscious software design—a framework for modeling software on hardware," in *Proceedings of 1994 IEEE Symposium on Low Power Electronics*, San Diego, CA, Oct. 1994, pp. 36–37.
3. P. Landman and J. Rabaey, "Black-box capacitance models for architectural power analysis," in *Proceedings of the International Workshop on Low Power Design*, Napa, CA, April 1994, pp. 165–170.
4. P. Landman and J. Rabaey, "Activity-sensitive architectural power analysis for the control path," in *Proceedings of the International Symposium on Low Power Design*, Dana Point, CA, pp. 93–98, April 1995.
5. L.W. Nagle, "SPICE2: A computer program to simulate semiconductor circuits," University of California, Berkeley, No. ERL-M520, 1975.
6. A. Salz and M. Horowitz, "IRSIM: An incremental MOS switch-level simulator," in *Proceedings of the Design Automation Conference*, pp. 173–178, 1989.
7. C.X. Huang, B. Zhang, A.C. Deng, and B. Swirski, "The design and implementation of PowerMill," in *Proceedings of the International Symposium on Low Power Design*, Dana Point, CA, April 1995, pp. 105–110.
8. V. Tiwari, S. Malik, and A. Wolfe, "Power analysis of embedded software: A first step towards software power minimization," *IEEE Transactions on VLSI Systems*, Vol. 2, No. 4, pp. 437–445, Dec. 1994.
9. Intel Corp., *Intel486 Microprocessor Family, Programmer's Reference Manual*, 1992.
10. Intel Corp., *i486 Microprocessor, Hardware Reference Manual*, 1990.
11. Fujitsu Microelectronics Inc., *SPARClite Embedded Processor User's Manual*, 1993.
12. Fujitsu Microelectronics Inc., *SPARClite Embedded Processor User's Manual: MB86934 Addendum*, 1994.
13. V. Tiwari, S. Malik, and A. Wolfe, "Power Analysis of the Intel 486DX2," Technical Report Princeton Univ., Dept. of Elect. Eng., CE-M94-5, June 1994.
14. V. Tiwari and Mike T.C. Lee, "Power analysis of a 32-bit embedded microcontroller," accepted for publication, *VLSI Design Journal*.
15. T.C. Lee, V. Tiwari, S. Malik, and M. Fujita, "Power analysis and low-power scheduling techniques for embedded DSP software," in *Proceeding of the International Symposium on System Synthesis*, Sept. 1995, Cannes, France.
16. P. Landman and J. Rabaey, "Power estimation for high level synthesis," in *Proceedings of the European Design Automation Conference*, Paris, Feb. 1993, pp. 361–366.
17. V. Tiwari, S. Malik, and P. Ashar, "Guarded evaluation: Pushing power management to logic synthesis/design," in *Proceedings of the International Symposium on Low Power Design*, Dana Point, CA, April 1995, pp. 221–226.
18. S. Gary et al., "PowerPC 603, a microprocessor for portable computers," *IEEE Design & Test of Computers*, pp. 14–23, Winter 1994.
19. A. Corrae, "Overview of the power minimization techniques employed in the IBM PowerPC 4xx embedded controllers," in *Proceedings of the International Symposium on Low Power Design*, Dana Point, CA, April 1995, pp. 75–80.
20. M. Alidina, J. Monteiro, S. Devadas, A. Ghosh, and M. Papaefthymiou, "Precomputation-based sequential logic optimization for low power," *IEEE Transactions on VLSI Systems*, pp. 426–436, Dec. 1994.
21. L. Benini and G. De Micheli, "Transformation and synthesis of FSMs for low power gated clock implementation," in *Proceedings of the International Symposium on Low Power Design*, Dana Point, CA, April 1995, pp. 21–26.
22. S. Wuytack, F. Franssen, F. Cathoor, L. Nachtergaele, and H. De Man, "Global communication and memory optimizing transformations for low power systems," in *Proceedings of the International Workshop on Low Power Design*, Napa, CA, April 1994, pp. 203–208.
23. S. Wuytack, F. Cathoor, and H. De Man, "Transforming set data types to power optimal data structures," in *Proceedings of the International Symposium on Low Power Design*, April 1995, Dana Point, CA.
24. V. Tiwari, S. Malik, and A. Wolfe, "Compilation techniques for low energy: An overview," in *Proceedings of 1994 IEEE Symposium on Low Power Electronics*, San Diego, CA, Oct. 1994, pp. 38–39.
25. Press et al., *Numerical Recipes in C*, Cambridge Univ. Press, 1988.
26. C.W. Fraser and D.R. Hanson, "A retargetable compiler for ANSI C," *SIGPLAN Notices*, pp. 29–43, Oct. 1991.

27. C.L. Su, C.Y. Tsui, and A.M. Despain, "Low power architecture design and compilation techniques for high-performance processors," in *IEEE COMPCON*, Feb. 1994.
28. Texas instruments, *Digital Signal Processing Applications—Theory, Algorithm, and Implementations*, 1986.
29. T.C. Lee and V. Tiwari, "A memory allocation technique for low-energy embedded DSP software," in *Proceedings of 1995 IEEE Symposium on Low Power Electronics*, Oct. 1995, San Jose, CA.
30. A. Sudarsanam and S. Malik, "Memory bank and register allocation in software synthesis for ASIPs," in *Proceedings of the International Conference on Computer-Aided Design*, Nov. 1995, San Jose, CA.



Vivek Tiwari received the B.Tech. degree in Computer Science and Engineering from the Indian Institute of Technology, New Delhi, India in 1991. Currently he is working towards the Ph.D. degree in the Department of Electrical Engineering, Princeton University. He will join Intel Corporation, Santa Clara, CA, in fall 1996.

His research interests are in the areas of Computer Aided Design of VLSI and embedded systems and in microprocessor architecture. The focus of his current research is on tools and techniques for power estimation and low power design. He has held summer positions at NEC Research Labs (1993), Intel Corporation (1994), Fujitsu Labs of America (1994), and IBM T.J. Watson Research Center (1995), where he worked on the above topics.

He received the IBM Graduate Fellowship Award in 1993, 1994, and 1995, and a Best Paper Award at ASP-DAC'95.

vivek@ee.princeton.edu.



Sharad Malik received the B.Tech. degree in Electrical Engineering from the Indian Institute of Technology, New Delhi, India in 1985 and the M.S. and Ph.D. degrees in Computer Science from the University of California, Berkeley in 1987 and 1990 respectively.

Currently he is on the faculty in the Department of Electrical Engineering, Princeton University. His current research interests are: design tools for embedded computer systems, synthesis and verification of digital systems.

He has received the President of India's Gold Medal for academic excellence (1985), the IBM Faculty Development Award (1991), an NSF Research Initiation Award (1992), a Best Paper Award at the IEEE International Conference on Computer Design (1992), the Princeton University Engineering Council Excellence in Teaching Award (1993, 1994, 1995), the Walter C. Johnson Prize for Teaching Excellence (1993), Princeton University Rheinstein Faculty Award (1994) and the NSF Young Investigator Award (1994). He serves/has served on the program committees of DAC, ICCAD and ICCD. He is on the editorial boards of the Journal of VLSI Signal Processing and Design Automation for Embedded Systems.



Andrew Wolfe received a B.S.E.E. in Electrical Engineering and Computer Science from The Johns Hopkins University in 1985 and the M.S.E.E and Ph.D. in Electrical and Computer Engineering from Carnegie Mellon University in 1987 and 1992 respectively. He joined Princeton University in 1991, where he is currently an Assistant Professor in the Department of Electrical Engineering. He served as Program Chair of Micro-24 and General Chair of Micro-26 as well as on the program committees of several IEEE/ACM conferences. He has received the Walter C. Johnson award for teaching excellence at Princeton. His current research interests are in embedded systems, instruction-level parallel architectures and implementations, optimizing compilers and digital video.

awolfe@ee.princeton.edu.



Mike Tien-Chien Lee received his B.S. degree in Computer Science from National Taiwan University in 1987, and the M.S. degree and the Ph.D. degree in electrical engineering from Princeton University, in 1991 and 1993, respectively.

He has been working at Fujitsu Laboratories of America, Santa Clara, CA, as a Member of Research Staff since 1994. Before then he was a Member of Technical Staff at David Sarnoff Research Center, Princeton, NJ, working on video chip testing. His research interests include low-power design, embedded DSP code generation, high-level synthesis, and test synthesis. He received a Best Paper Award at ASP-DAC'95.

lee@fla.fujitsu.com.

Low-Power Architectural Synthesis and the Impact of Exploiting Locality

RENU MEHRA, LISA M. GUERRA AND JAN M. RABAEY
Department of EECS, University of California at Berkeley

Received November 17, 1995; Revised April 3, 1996

Abstract. Recently there has been increased interest in the development of high-level architectural synthesis tools targeting power optimization. In this paper, we first present an overview of the various architecture synthesis tasks and analyze their influence on power consumption. A survey of previously proposed techniques is given, and areas of opportunity are identified. We next propose a new architecture synthesis technique for low-power implementation of real-time applications. The technique uses algorithm partitioning to preserve locality in the assignment of operations to hardware units. Preserving locality results in more compact layouts, reduced usage of long high-capacitance buses, and reduced power consumption in multiplexors and buffers. Experimental results show reductions in bus and multiplexor power of up to 80% and 60%, respectively, resulting in 10–25% reduction in total power.

1. Introduction

High-level synthesis is steadily making an inroad into the digital design community. So far, most of the work has focused on techniques for area and speed optimization. In recent years, there has been significant interest in low-power issues due to excessive heat dissipation in increasingly complex digital systems and rising popularity of portable devices, where extending battery life is a primary design objective. Most of the work in design automation for low power has focused at the logic, circuit, and layout levels. Relatively little research has been devoted to high-level techniques, where the impact of design decisions is much greater [1–3].

Previously proposed techniques include optimizations that enable voltage scaling [1] and those that preserve data correlations [4–7]. Until now, however, optimization of interconnect (i.e., buses, buffers and multiplexors) power has not been addressed. Interconnect optimization is important because interconnect power may be a substantial percentage of the total power and it can be affected significantly by synthesis optimizations.

In this paper, a synthesis technique for optimizing interconnect power for real-time applications is presented. The technique uses algorithm partitioning to

preserve the locality of operations in their assignment to hardware. Our experiments have shown that preserving locality results in reduced interconnect power due to more compact layouts, reduced usage of long high-capacitance buses, and lower multiplexor and buffer power. In Section 2 we present a breakdown of the parameters affecting power (i.e., voltage, frequency, physical capacitance, number of resource accesses, and data correlations) and for each, determine the synthesis tasks that can most directly impact its value. Section 3 explains how exploitation of locality during the synthesis process reduces interconnect power. Previous approaches for partitioning and the details of our low-power partitioning methodology are given in Section 4. In Section 5, we present our synthesis techniques which have been integrated into the *Hyper-LP* system. A new model for estimation of bus power in partitioned designs is described in Section 5.2 and the experimental results are summarized in Section 6.

2. Synthesis Tasks and Power Consumption—An Analysis

In this section we discuss the effect of various synthesis tasks on the power consumption of the different components of the chip. The focus is on synthesis for

ASIC implementation of real-time DSP applications. First we present a brief overview of the synthesis tasks and the different architecture-level techniques for low-power design.

2.1. Architecture Synthesis: Background

Architecture synthesis is concerned with deriving an architectural implementation of a given algorithm. The input is a behavioral description of the algorithm and the synthesis process involves deciding how the operations in the algorithm will be mapped onto a set of hardware resources. A good tutorial on the main high-level synthesis tasks is presented in [8] and a number of available CAD systems for high-level synthesis are described in [9]. Though terms are defined slightly differently in different systems, the basic tasks are the same. The following paragraphs define some terms as they will be used in this paper.

The main tasks in the architecture-synthesis process include module selection, allocation, assignment, and scheduling. Module selection involves selecting specific hardware modules that implement the operations specified by the algorithm. Allocation refers to the task of deciding how many instances of each hardware resource are needed. Assignment binds each of the operations to specific hardware instances and scheduling decides when each operation will be executed. Both allocation and assignment are performed for each of the different resource types (functional units, registers, and buses) in the system.

For memory intensive applications, an important synthesis task is memory management which includes generation of addresses, deciding whether variables should be stored in registers or background memory, allocating the number and size of the memory blocks, and assigning variables to specific memory blocks.

Algorithm selection and transformations are also important tasks in high-level synthesis. Algorithm selection involves choosing the best algorithm from a set of algorithms for the same application. Transformations modify the algorithm structure without altering its input-output relationship. Common transformations include pipelining, retiming, algebraic manipulations, loop merging, and loop folding.

2.2. Architecture-Level Power Reduction Techniques

The sources of power consumption on a chip are dynamic power, short-circuit power, and leakage power.

At the algorithm and architecture levels, only dynamic power is targeted for optimization. This is because short-circuit and leakage currents, (i) can be reduced to less than 15% of the total chip power by smart circuit design techniques [10], and (ii) are influenced mainly by the circuit design style used. At the algorithm and architecture level, therefore, the power dissipated can be described by the following equation:

$$Power = C_{eff}(V_{sw} \cdot V_{DD})f \quad (1)$$

where f is the frequency of operation, V_{sw} is the switched voltage, V_{DD} is the supply voltage, and C_{eff} is the effective capacitance switched. C_{eff} is in turn dependent on C , the physical capacitance being charged/discharged, and α , the activity factor:

$$C_{eff} = \alpha C \quad (2)$$

Due to the quadratic effect of voltage on power, voltage scaling results in large power reductions, at the expense, however, of increased delays. A common approach to power reduction, therefore, is to first increase the performance of the design and then reduce the voltage as much as possible. Proposed methods to do this include increasing algorithm concurrency, pipelining and retiming for speed, using faster units, and increasing the number of hardware units used [1]. For instance, the critical path of the third-order FIR filter can be reduced from 3 clock cycles (Fig. 1(a)) to 2 by retiming (Fig. 1(b)). Since the throughput of the application is fixed, this speed-up can be used to scale the voltage from 5 to 3.2 Volts, thereby reducing the power from 136.3 to 53.2 mW, a 61% reduction. Speed-up techniques such as these typically result in increased silicon area and therefore are commonly referred to as “trading area for power”.

Another way to reduce power is to operate at reduced speeds since the power dissipation is directly proportional to the frequency. As this work targets real-time applications that have fixed timing constraints, frequency reduction is not considered. Though the external data rate is fixed, the internal clock speed can be varied such that the time is maximally utilized by the hardware. Design space exploration for clock selection is presented in [2, 11].

The effective capacitance can be reduced by avoiding wasteful computations. Rules for avoiding waste at the

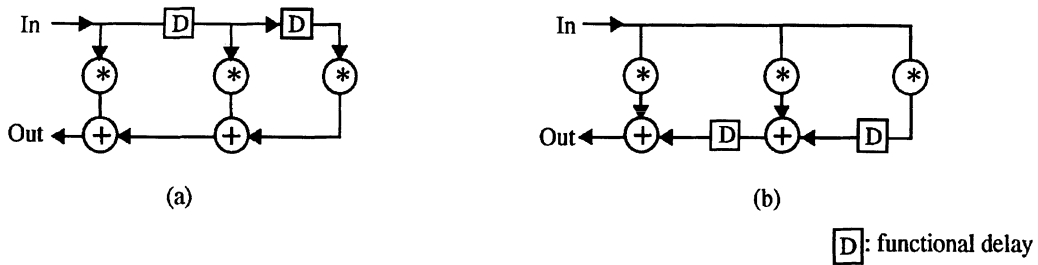


Figure 1. Using speed-up transformations to reduce power: (a) original structure, (b) after retiming.

architecture level can be classified along the following lines:

- *Preservation of Data Correlations:* Switching activity is dependent on correlations between successive data inputs and increasing correlations results in large power savings.
- *Distributed Computing/Locality of Reference:* Accessing global computing resources (control, datapath, memory, I/O) is expensive: the time-sharing nature of these resources requires a high switching rate, and the shared nature of such a resource typically incurs a capacitive overhead. Distributing the accesses over many resources relieves both the switching requirements and the overhead. For example, accesses to long global buses are costly and keeping data local reduces power consumption in data communications.
- *Application-Specific Processing:* Specialized units consume less power than general-purpose ones due to simpler structure and reduced control required to support programmability.
- *Demand-Driven Operation:* To avoid wasteful transitions, it is important to perform operations only when needed. Power down of memory and functional units when they are not in use is the most popular technique in this category.

These power reduction concepts will recur in the next section, where we analyze how the different synthesis tasks impact the effective capacitance and the power reduction techniques they use.

2.3. Effect of Synthesis Tasks on Switched Capacitance, C_{eff}

The capacitance switched by each resource type — functional units, memory (including register files), interconnect (buses, multiplexors, and buffers), and control — depends on three factors: the resource’s physical

capacitance, the number of times it is accessed, and the correlation of the data that it operates on (the latter two determine the activity factor). While all three factors should be reduced to lower power consumption, the impact of reducing any one may depend on the values of the other factors. For example, it is more effective to reduce accesses to resources if they have a high physical capacitance.

Each of the above C_{eff} factors for a resource is affected by decisions made during synthesis. In this section, we analyze these effects and describe the relevant research efforts. While the synthesis tasks are highly inter-dependent and thus may limit or enhance each other’s effects, we present only those tasks which influence each capacitance factor most directly. Table 1 summarizes these influences and the remainder of this section elaborates upon them. Note that algorithm selection and transformations can affect all factors of effective capacitance.

Let us consider first the tasks affecting physical capacitance (Table 1, column 1). The physical capacitance of functional units, memory, registers, multiplexors, and buffers depends on the selection of modules from the hardware library. In general, faster, more capacitive units may be needed in timing critical situations; less capacitive units are better for cases where the timing requirements are not so critical. For example, as shown in [2], at higher voltages, a ripple-carry adder leads to more energy-efficient designs than a carry-select adder (Fig. 2). At lower voltages, however, the ripple-carry adder may not be fast enough to meet the speed requirement and the carry-select adder can be used. Goodby et al. [12] used module selection to speed up non-pipelineable paths to meet the timing constraint, using cheaper units for less time-critical paths.

Another important trade-off in module selection lies in the use of specialized units instead of programmable ones. For example, it may be worthwhile to use a specialized adder instead of an ALU if there are a large number of additions in the algorithm. Specialized units

Table 1. Synthesis tasks that most directly affect the different factors of effective capacitance.*

	Physical capacitance		Resource accesses	Data correlation
Functional units	Module selection	—	—	Memory management Assignment (functional units, registers, buses) scheduling
Memory		Memory management	Memory management	
Registers		Register allocation Memory management	Register assignment Memory management	
Muxes/buffers		Functional unit assignment	Functional unit assignment Bus assignment	
Buses	Placement and routing [†] Bus assignment Allocation (functional units, registers, buses)		Bus assignment	
Control and control wiring	Assignment, scheduling Logic synthesis [†] Placement and routing [†]			

* Algorithm selection and transformation can affect all factors of effective capacitance.

[†] Lower level synthesis tasks.

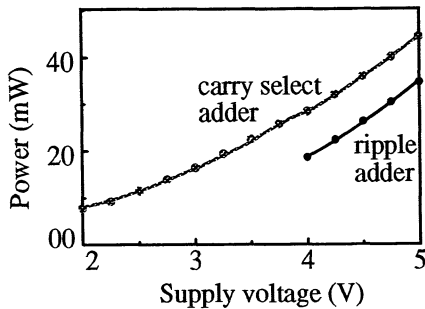


Figure 2. Relative power dissipation for an application running at a fixed throughput using different adders.

consume less power for performing a particular operation but, on the other hand, do not have the flexibility to perform as many types of operations.

For memory units, the size, and therefore, the physical capacitance being switched per access, is affected by memory management. Similarly, the main task that influences the size of register files is the allocation of registers. Since memory management decides whether variables should be stored in registers or background memory, it also influences the size of the register files and their physical capacitance. Previous works have explored the use of algorithm transformations for memory size reduction [13]. Consider the loop shown in Fig. 3(a). Arrays *A* and *C* are already available in memory; when *A* is consumed another array *B* is generated; when *C* is consumed a scalar value, *D*, is produced. Memory size can be reduced by executing the *j*

```

for l := 1 to N do
  B(l) := f(A(l));
for j := 1 to N do
  D := g(C(j), D);
(a)
for j := 1 to N do
  D := g(C(j), D);
for i := 1 to N do
  B(i) := f(A(i));
(b)

```

Figure 3. Loop transformations for reducing memory power — loop interchange.

loop before the *i* loop (Fig. 3(b)) so that *C* is consumed before *B* is generated and the same memory space can be used for both arrays.

The physical capacitance of buses is directly related to their lengths, which are mainly determined by the number and size of the hardware units and their placement and routing. In general a large number of units with a lot of connections between them will lead to long buses. The number of hardware units is determined by resource (functional unit, register, and bus) allocation. Bus assignment (also called bus merging) affects the physical capacitance since it can affect their lengths and the capacitive loading on them. For example, the length of a bus may be increased if it is merged with another bus that has different sources and destinations. Bus lengths can be reduced by exploiting locality of the operations in the algorithm. We will show that careful partitioning of a design into small, well-connected parts can reduce the lengths (and capacitance) of a large fraction of the buses.

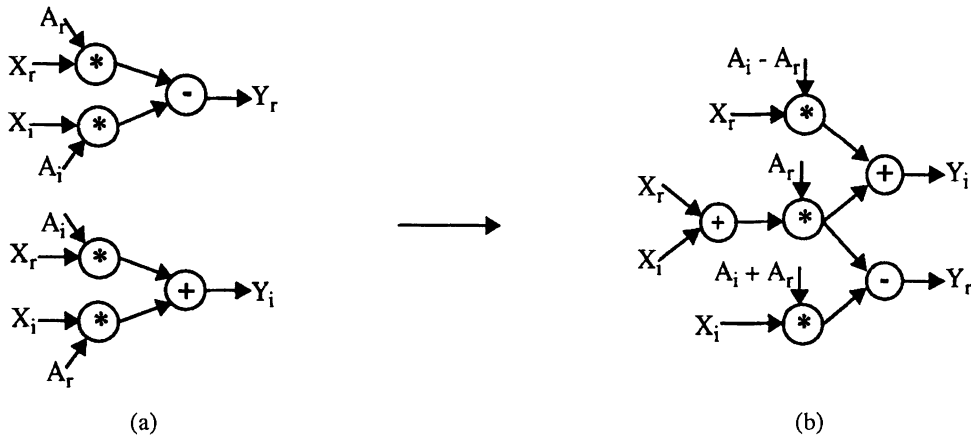


Figure 4. Reduction of multiply operations through transformations — distributivity, redundancy manipulation, and common sub-expression elimination: (a) 4 multiplications, (b) 3 multiplications.

Consider next the tasks affecting the number of accesses (Table 1, column 2). The number of accesses to each hardware component is influenced by the algorithm. This can be changed by using transformations. Several algorithm transformations, such as operation reduction and operation substitution, that reduce the accesses to functional units have been proposed [1]. In Fig. 4, the number of multiply operations is reduced by applying distributivity, redundancy manipulation, and common sub-expression elimination. Notice that, for this example, this comes at the cost of an increase in critical path.

Accesses to memory are governed by the behavioral description and the memory management techniques used. During memory management, arrays are assigned to either registers or memory. In general, accessing a value from the register file is cheaper since the size of the register file is smaller. In the example of Fig. 5, the array A is the input to the loop and the array C is the output; array B stores intermediate values. Since only one value of B needs to be alive at a given time, the array can be stored in a register eliminating the related memory accesses.

Another way to reduce memory accesses is to use loop-based transformations as was proposed by Catthoor et al. [13].

```

for j := 1 to N do
    B(l) := f(A(l));
    C(l) := g(B(l));
end
    
```

Figure 5. Simple loop illustrating memory access reduction.

For register files, the accesses depend on the architecture model being used. For example, in a single centralized register file scenario, writes are determined by the algorithm (exactly equal to the number of variables) whereas for distributed register files, a single variable may need to be stored in more than one place. For a given architecture model, the number of reads from and writes to registers depends on the register assignment and the schedule.

Accesses to buffers are determined primarily by the algorithm, since often each data transfer is buffered. The multiplexor accesses, on the other hand, are independent of the algorithm, and are instead, determined by the synthesis tasks. Assignment affects the amount of time-multiplexing of the functional units, which in turn affects the multiplexing of data transfers. Accesses to multiplexors is further affected by bus assignment — if a unit needs data from two or more sources, a multiplexor may or may not be required at the inputs depending on whether the corresponding data transfers are merged onto the same bus.

Accesses to buses depend on the total number of data transfers in the algorithm. Bus assignment further affects the accesses since if a single variable needs to be transferred to more than one destination, one or more bus transfers may result based on whether the connections to the two units are merged into one bus. In this context, it is important to note that, if the buses are not all the same size, all accesses to buses do not consume the same amount of power and it is more important to reduce accesses to the longer buses. Preserving locality during bus assignment can reduce accesses to long global buses.

For a clocked resource, power is consumed for clocking even when it is not being accessed. Power down techniques are especially popular for reducing power consumption in memories during idle cycles. Farrahi et al. [14] presented a memory segmentation algorithm for this purpose. The main idea is to partition the memory space so that memory accesses that are temporally close to each other are in the same block. In this way, only one block needs to be active for a given period of time and the other memory blocks can be shut off.

The controller generates signals to control reads from and writes to registers, tri-stating of buffers, amount of shift for shifters, etc. The control related power includes the power consumed by the control wiring and the control logic. Wiring power depends on the length of buses which is determined by the placement and routing. The power consumed by the control logic depends on the assignment, schedule, and logic level optimizations. It is difficult to relate the controller power to high-level parameters, and therefore, to account for this component during architecture synthesis. One attempt to relate control power to high-level parameters is reported in [2]. At the logic level, the problem of estimating and optimizing controller power has been well studied.

Finally, consider the data correlation component of power (Table 1, column 3). Input correlations of all the components are affected by the allocation, assignment, and schedule. However the effect of these tasks on the correlations cannot be easily determined during synthesis because the correlations also depend heavily on the input data. Some research has been done to minimize switching activity during hardware assignment and scheduling. For assignment, the objective is to bind operations onto hardware so that the input signal activity is minimized. Raghunathan and Jha [4] proposed an assignment scheme to minimize the average number of bit transitions on the signal inputs to hardware units (obtained from simulations).

Musoll and Cortadella [5] minimize the bit transitions for constants during scheduling. Consider, for example, the FIR filter of Fig. 6. There are four multiplications with constants — c_0, c_1, c_2, c_3 — which can be scheduled on a single multiplier such that the transition activity at the right input of the multiplier is minimized. For the values of the constants given in the figure, the schedule $c_0 \rightarrow c_1 \rightarrow c_2 \rightarrow c_3 \rightarrow c_0$ results in 26 transitions for a 12-bit implementation whereas the schedule $c_0 \rightarrow c_1 \rightarrow c_3 \rightarrow c_2 \rightarrow c_0$ results in 34 transitions. Other methods suggested in [5] for increasing

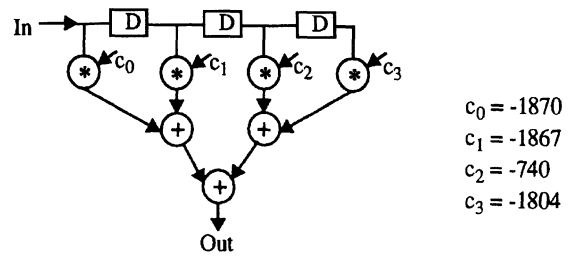


Figure 6. Scheduling to minimize the transitions.

signal transitions include operand sharing (executing operations with common inputs in successive cycles on the same hardware), loop interchange and operand reordering.

Chatterjee and Roy [6] studied the effect of operand activity on the power consumption of additions and multiplications and used it for appropriate graph transformations. While the above techniques have focussed on increasing the correlations for functional units, Chang and Pedram [7] proposed a register assignment scheme that reduces the activity for register files.

In this section we have presented the architecture-synthesis tasks that most directly affect the different factors comprising effective capacitance. It is important to notice that the tasks are highly interdependent and, therefore, the impact of each on power may be constrained by other tasks. For example, though the physical capacitance of buses depends on bus assignment, the effect of this task can be limited by functional unit assignment. If functional units are assigned such that number of destinations of each unit are low, bus assignment can produce in better solutions.

As another example, consider number of accesses to registers in a distributed register file model. Assume that each hardware unit has a dedicated register file that stores its inputs. Though the number of writes to registers is determined by the assignment of variables to specific register files, variable assignment to registers is in turn determined by whether the operations that need these variables are assigned to corresponding functional units.

The rest of the paper focuses on one approach to low-power synthesis. We explore the impact of exploiting spatial locality for reduction of interconnect power and provide synthesis techniques for this purpose.

3. The Impact of Exploiting Locality

In this section we examine the impact of preserving spatial locality in time-shared ASIC implementations.

We begin with an example emphasizing the importance of interconnect power, and then show how exploiting spatial locality can be used to reduce it.

3.1. Importance of Interconnect Power

While for area optimization, high resource utilization through hardware sharing is one of the main goals, for power optimization, reduced hardware sharing often gives better results. Consider Wu's comparison of an automatically-generated maximally time-shared and a manually-generated fully-parallel implementation of a QMF sub-band coder filter [15]. In the manual design, a number of optimizations were used to obtain power savings in the various components. The power consumption of both versions is documented in Table 2. For the same supply voltage, an improvement of a factor of 10.5 was obtained at the expense of a 20% increase in area.

Note that the interconnect elements (buses, multiplexors, and buffers) consume a large percentage of the total power — 43% and 28% in the time-shared and parallel versions, respectively. Moreover, large improvement factors were obtained for these components — 16.9, 15.1, and 12.5 for buses, multiplexors, and buffers, respectively — mainly due to dedicated communication and reduced usage of multiplexors and buffers. Clearly, there is a large opportunity for decreasing power consumption through interconnect power reduction. Typical designs from the *Hyper* synthesis system [16] have shown that buses may alone consume 5 to 40% of the total power and the interconnect elements together may contribute 25–50% of the total power.

While in the above example, the fully-parallel implementation resulted in large power gains with low area overhead, this may not always be the case. Paral-

Table 2. Power consumption (mW) in the maximally time-shared and fully-parallel versions of the QMF sub-band coder filter.

	Time-shared	Fully-parallel	Improvement factor
Functional units	8.52	1.03	8.3
Registers	9.76	1.08	9.0
Buses	23.69	1.40	16.9
Multiplexors	3.77	0.25	15.1
Buffers	4.36	0.35	12.5
Others	23.99	2.92	8.2
Total	74.09	7.03	10.5

lel implementations may be too area intensive and may not necessarily result in reduced interconnect power. In general, area can be traded-off to a certain extent to reduce the power consumption in buses, multiplexors, and control. If the area overhead is too high, the increase in the required bus lengths may offset the power gains due to other factors.

In this work, techniques are presented to achieve low-power designs by reducing the interconnect power while incurring low area overhead. The approach aims to capture some of the optimizations of the above example in an automated way while maintaining a balance between the maximally time-shared and the fully-parallel implementations. The next section illustrates the main idea behind our proposed low-power synthesis technique.

3.2. Exploiting Spatial Locality for Interconnect Power Reduction

The main idea behind our approach is to synthesize designs with localized communications. We achieve this by dividing the algorithm into *spatially local clusters* and performing a *spatially local assignment*. A spatially local cluster is a group of algorithm operations that are tightly connected to each other in the flowgraph representation. Two nodes are tightly connected if the shortest distance between them, in terms of number of edges traversed, is low. A spatially local assignment is a mapping of the algorithm operations to specific hardware units such that no operations in different clusters share the same hardware. Partitioning the algorithm into spatially local clusters ensures that the majority of data transfers take place within clusters and relatively few occur between clusters. The spatially local assignment restricts intra-cluster data transfers to buses that are local to a subset of the hardware (local buses); thus only inter-cluster data transfers use buses that are shared by all resources (global buses). In general, since intra-cluster buses are localized to a part of the chip, they are shorter than the buses in the original designs, while the global buses in the original and partitioned designs may be comparable in length. The combined result is that local buses which are used more frequently are shorter, and longer highly-capacitive global buses are used rarely. The partitioning information is passed to the architecture-netlist generation and floorplanning tools which place the hardware units of each spatially local cluster close together in the final layout.

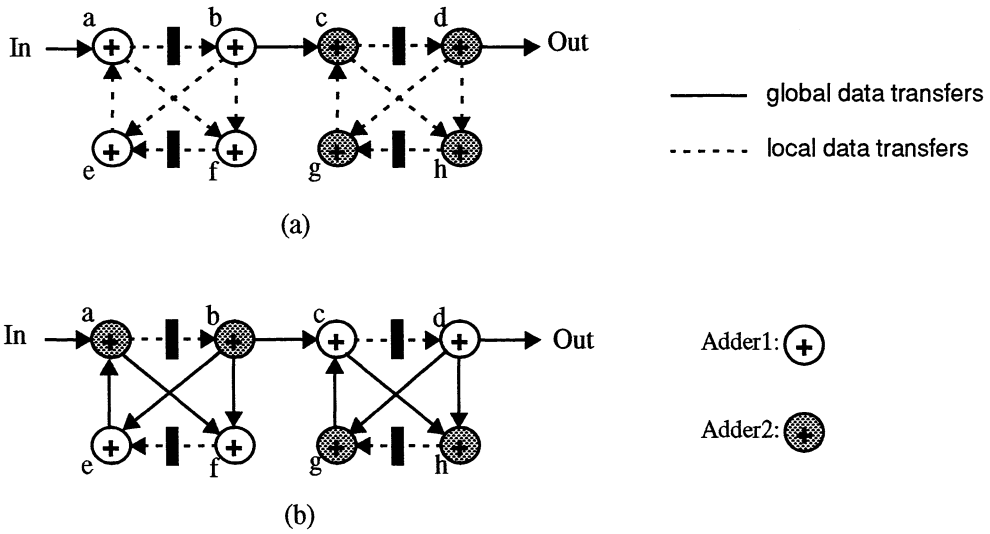


Figure 7. Example of spatially local and non-local assignments of a given graph: (a) local assignment, (b) non-local assignment.

Consider the example of Fig. 7, showing two alternative mappings of a single flowgraph to a hardware configuration consisting of two adders. In Fig. 7(a), all operations of a tightly-connected group are mapped to the same hardware (for example, *a*, *b*, *e*, and *f* are all mapped to adder 1). This does not hold in the assignment of Fig. 7(b). Considering data transfers in which a given adder outputs data to its own inputs as *local*, and those in which it outputs data to the other adder as *global*, we find that assignments of Fig. 7(a) and 7(b) have 1 and 9 global data transfers, respectively

(excluding input and output connections). Since global buses are long and highly capacitive compared to local ones, reducing accesses to global buses reduces the power dissipation.

As another example, consider the two different assignments for maximum throughput implementation of a fourth-order parallel-form IIR filter shown in Fig. 8. Indicated beside each operation is the hardware resource assigned to it (A_i are adders and M_i are multipliers). In Fig. 8(a), the graph is divided into two spatially local clusters and the operations in each cluster

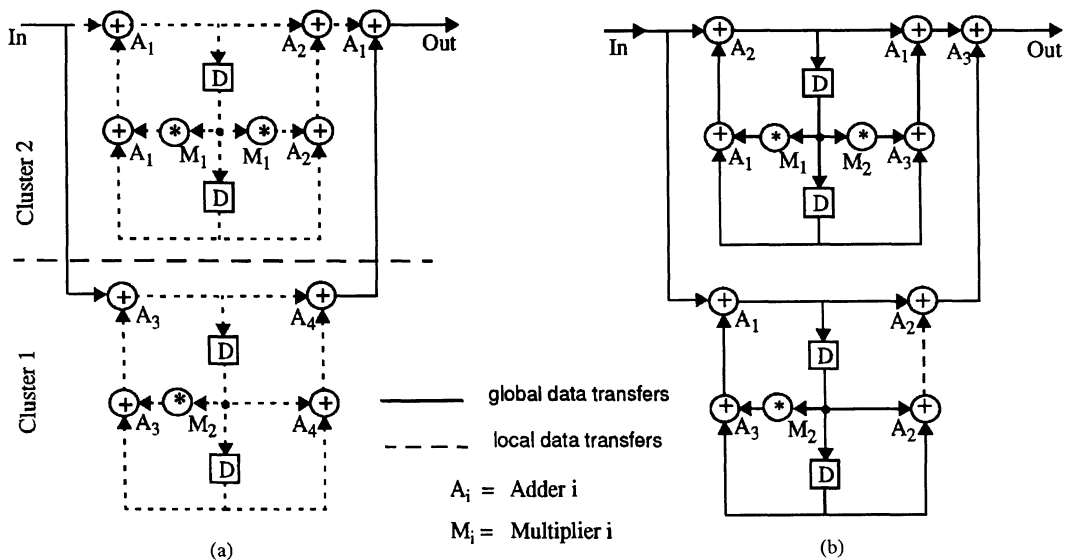


Figure 8. A fourth-order parallel-form IIR filter: (a) local assignment, (b) non-local assignment.

are mapped to mutually exclusive sets of hardware resources (A_1 , A_2 , and M_1 are used for operations in cluster 1 and A_3 , A_4 , and M_2 are used for those in cluster 2). As a result, a large number of the communications are restricted to only a subset of the hardware. In Fig. 8(b), however, the hardware is not partitioned and all communications are global. The number of global data transfers (shown with solid lines in both cases) for the local and the non-local assignments are 2 and 20, respectively.

Several factors may affect the quality of partitioned designs. Firstly, the local assignment may come at the cost of extra hardware. In the above example, the local version needs 4 adders and 2 multipliers whereas the non-local assignment requires just 3 adders and 2 multipliers. However, this increase in the number of functional units does not necessarily translate into a corresponding increase in the overall area since localization of interconnect makes the design more conducive to compact layout. Secondly, reduced hardware sharing results in additional power savings in multiplexors and buffers. Thirdly, varying the number of clusters trades off local and global bus power. In particular, as the number of clusters is increased, the number of inter-cluster communications increases but the local bus lengths decrease. Our partitioning methodology takes these factors into consideration by providing techniques for evaluating the power-saving potential of proposed partitions.

4. Partitioning for Low Power

The core of our proposed low-power synthesis approach involves partitioning the algorithm into spatially local clusters of computation. In this section we present previous work in partitioning (Section 4.1), review the concepts of spectral partitioning (Section 4.2), describe our the algorithm representation and the hyperedge model (Section 4.3), and present the overall partitioning methodology (Section 4.4).

4.1. Previous Work in Partitioning

Previous works in partitioning for high-level synthesis have targeted area minimization, with a significant portion of the gains resulting from interconnect reduction. In the BUD system [17], pairs of nodes are repeatedly merged based on three criteria: number of common connections, possibility of executing them at the same time, and possibility of executing them on the same

hardware type. In the APARTY system [18], clustering is performed in multiple stages: each stage clusters nodes based on a particular criterion such as reducing the number of control transfers, increasing hardware sharing, and decreasing data transfers.

In partitioning for low power, the goal is to reduce total chip power by reducing interconnect power. One way in which this is done is by maximizing the number of accesses to short local buses relative to long global ones. Note that in area minimization, the number of global buses is minimized. In power minimization, however, it is better to have two global buses each accessed twice rather than one bus accessed six times.

Assuming that global buses are only used for data transfers between partitions, the number of accesses to global buses is equal to the total number of edges cut by a partition. For area optimization in high-level synthesis, the cuts do not translate exactly into the number of buses required because of hardware sharing between units. However, partitioning techniques for lower-level CAD (e.g., layout) minimize the number of edges cut. For power purposes, therefore, it makes sense to look at partitioning techniques used in circuit-level and layout-level CAD.

A variety of techniques have been used for partitioning at the logic, circuit and layout levels. These include iterative improvement methods such as Kernighan and Lin [19], Fiduccia and Mattheyses [20], and simulated annealing [21]; bottom-up aggregative algorithms such as [22]; top-down recursive bi-partitioning [23, 24]; and spectral partitioning techniques [25–32].

We have developed a new behavioral-level partitioning method for low-power. The basic idea is to derive an ordering of the nodes by using the spectral properties of the graph and then heuristically partition this ordering. The theoretical results that form the basis of this technique are presented in the next section.

4.2. Key Ideas in Spectral Partitioning

Spectral methods use eigenvectors of the Laplacian of the graph to extract a one-dimensional placement of graph nodes which minimizes the sum of squares of edge lengths. This placement is then heuristically partitioned. The key result that forms the basis of this technique was presented by Hall [33], and is given below.

Problem Statement: Find a one dimensional placement $\vec{x} = (x_1, x_2, \dots, x_n)$ of the nodes of a given weighted-edge graph that minimizes the weighted sum of squares of the edge lengths.

Solution: Let A be the weighted adjacency matrix of the graph, where A_{ij} is the weight of the edge between nodes i and j ; $A_{ij} = 0$ if there is no edge between i and j . The cost function, z , that needs to be minimized is given below.

$$z = \frac{1}{2} \sum_i^n \sum_j^n (x_i - x_j)^2 A_{ij} \quad (3)$$

The following constraint is used to normalize the placement between -1 and 1 .

$$|\bar{x}| = \sqrt{\bar{x}^T \bar{x}} = 1 \quad (4)$$

Define a degree matrix, D , as the diagonal matrix in which each diagonal element is the sum of the weights of all the edges connecting to the corresponding node. The cost function z can be rewritten as $\bar{x}^T (D - A) \bar{x}$. The matrix $Q = (D - A)$ is called the Laplacian of the graph. The constrained cost function is given by the Lagrangian, L , as

$$L = \bar{x}^T Q \bar{x} - \lambda (\bar{x}^T \bar{x} - 1) \quad (5)$$

Setting the derivative of the Lagrangian, L , to zero gives Eq. (6).

$$(Q - \lambda I) \bar{x} = 0 \quad (6)$$

The solutions to Eq. (6) are those where λ is the eigenvalue and \bar{x} is the corresponding eigenvector. The smallest eigenvalue, 0 , gives a trivial solution with all nodes at the same point. The eigenvector corresponding to the second smallest eigenvalue minimizes the cost function while giving a non-trivial solution. \square

4.3. Algorithm Representation

The input algorithm is represented internally as a data-flow graph. The nodes represent operations and the

edges represent data dependencies. Strictly speaking, the edges are “hyperedges” since a node may have several fanouts. Conditionals are implemented in the datapath — all branches are executed and the conditional test is used to select the appropriate result. The representation can be hierarchical, that is, a node may itself be a graph having nodes and edges. However, our current implementation of the partitioning methodology does not handle hierarchy.

While connections are represented in the algorithm as hyperedges, the partitioning technique requires a representation in terms of edges in the strict sense (an edge is a connection between only two nodes). Several models to replace the hyperedge by edges were examined. All of them replace the hyperedge by edges between pairs of nodes to form a clique but differ in the weight assigned to the resulting edges.

The hyperedge problem is similar to the one discussed in layout partitioning where a net may connect several pins. In that case, the uniformly weighted clique model [34] has been widely used. This model assigns a weight of $1/(k - 1)$ to all edges in the clique, where k is the number of nodes in the clique (Figs. 9(a) and (b)). If a hyperedge is cut, its contribution to the weight of the cut is exactly one. At the layout level the sum of weights of the edges cut by a partition under this model exactly corresponds to the number of nets cut.

At the high level, however, the hyperedge does not correspond to a bus. Consider the hyperedge shown in Fig. 9(a). If a cut removes node z from the rest of the clique, exactly one global data transfer will be required. However, if the cut isolates node a from the rest of the clique, the number of data transfers between clusters will be anywhere from 1 to 3 depending on how the data transfers are assigned to buses. Therefore, a model that weights the edges between nodes x , y , and z less than the edges connecting a to the nodes x , y , or z may result in better solutions. We propose two new models that do this. In the first model, we assign a lower weight

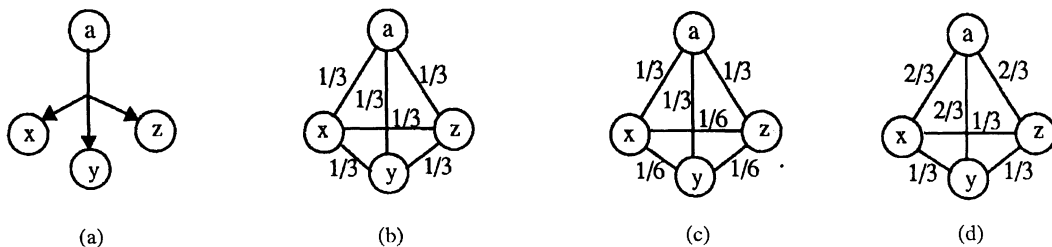


Figure 9. Hyperedge models: (a) initial hyperedge, (b) uniformly weighted clique model, (c) clique model with lower weights on edges between destination nodes, (d) clique model with higher weights for edges connected to the source node.

$1/2(k-1)$ on edges between the destination nodes leaving the weights on the other edges unchanged. In the second model, we increase the weight of the edges that join the source to the destination nodes to $2/(k-1)$ while the weights on edges between the destination nodes is unchanged. These two models are shown in Figs. 9(c) and (d), respectively. Our experiments showed that different models give the best result in different situations. In our clustering methodology, we try all three hyperedge models and select the best one.

4.4. Overall Low-Power Partitioning Methodology

The goal of the partitioning methodology is to generate a single promising partition for the purposes of low-power synthesis. The methodology is implemented in two phases. In Phase I, several candidate graph partitions are generated. In Phase II, these partitions are evaluated and the most promising one is selected.

Phase I reduces the space of possible partitions to a few partitions that are balanced and localized. Multiple solutions are generated by using the different hyperedge models and by varying the maximum allowed number of clusters. As the number of clusters increases, the number of global accesses increases while the size of local buses reduces, resulting in lower local-bus and higher global-bus power. Generating several solutions with different number of clusters explores this trade-off. An overview of the partitioning methodology is shown in Fig. 10.

4.4.1. Phase I: Finding Good Partitions. The goal of this phase is to propose a few promising partitions. The eigenvector placement obtained as described in Section 4.2 forms the nucleus of the approach. It provides an ordering in which nodes tightly connected to

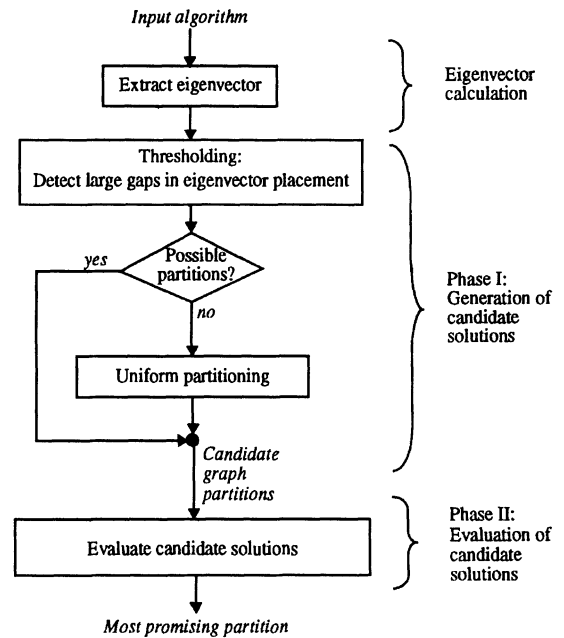


Figure 10. Overview of the partitioning methodology.

each other are placed close together. Furthermore, the relative distances is a measure of the tightness of connections. We use the eigenvector ordering to generate several partitioning solutions. The spectral technique is specially suited to our needs since the eigenvector is computed only once and generating partitions from it is computationally inexpensive.

Two main techniques are used to generate partitions from this ordering. The first technique uses the relative distances between the nodes to detect clusters for graphs of the type shown in Fig. 11(a) while the second uses the ordering to partition algorithms of the type shown in Fig. 11(b). The example shown in Fig. 11(a) has two distinct clusters and this is also reflected in the

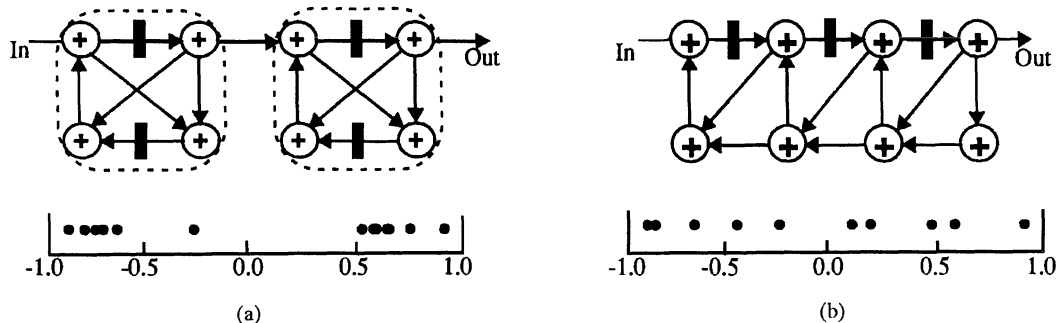


Figure 11. Two different examples and the corresponding eigenvectors: (a) example with natural clusters, (b) partitionable example with no natural clusters.

corresponding eigenvector. The second example, however, does not have any distinct clusters and the nodes are uniformly spaced in the eigenvector placement.

The first technique detects natural clusters inherent in the algorithm. Large gaps in the eigenvector placement are used to indicate good points for partitioning. Since the nodes are always placed between -1 and 1 , the absolute values of distances vary depending on the total number of nodes in each example. By a “large gap” we mean the distance between two adjacent nodes in the placement that is large relative to distances between other nodes in the same example. The threshold for detecting these gaps is, therefore, relative to the distances in the same example. Several different thresholds for identifying large gaps — $m + \sigma$, $m + 2\sigma$, and $m + 3\sigma$ — where m is the mean of the distances and σ is the standard deviation, were evaluated. Preliminary experimentation showed that although the $m + 3\sigma$ threshold found most of the clusters, some clusters were only detected using $m + 2\sigma$. In our methodology, we therefore try the $m + 3\sigma$ threshold first. If no clusters are detected the threshold is reduced to $m + 2\sigma$. Smaller thresholds may be tried for a more exhaustive exploration of the design space.

Solutions targeting up to 2, 3, 4, and 8 clusters are generated. This is done by varying the constraint on the number of nodes allowed in each cluster. Notice that fixing the smallest cluster size to have $(\lceil n/x \rceil + 1)$ nodes, where n is the total number of nodes in the graph, limits the maximum number of clusters to $x - 1$.

Partition points are inserted in the 1-dimensional placement to mark large gaps based on the $m + 3\sigma$ or $m + 2\sigma$ metric. These points mark boundaries between different groups of nodes. If a group detected has less nodes than allowed by the size constraint, it is merged with one of the neighboring groups. The decision regarding which neighboring group to merge with, is based on the size of the gap between the groups (as calculated from the eigenvector). The gap between two adjacent groups in the eigenvector placement is the distance from the right-most node of the left group to the left-most node of the right group. Clusters are thus identified using thresholding and subsequent merging.

The main goal of the above technique is to detect natural clusters inherent in the algorithm. However, not all algorithms have a clearly clustered structure. A non-clustered algorithm may still be very partitionable in that it may be possible to partition it so that few edges are cut relative to the total number of edges. For example the graph shown in Fig. 11(b) is not clustered

but is partitionable since only three edges are cut when it is partitioned into two equal parts. Another good example is an FIR filter structure which has no distinct groups of clustered operations but can be easily partitioned.

If no clusters are found by thresholding, a second technique is applied in which the eigenvector placement is evenly divided into clusters. Again solutions targeting 2, 3, 4, and 8 clusters are generated. Depending on the quality of solutions required and the time that the user wants to spend, a greater number of partitioning solutions may be tried. The best solution is selected using the evaluation criteria of Phase II.

4.4.2. Phase II: Evaluation. The generation of candidate partitions is based on minimizing the number of global bus accesses. The underlying assumption is that intra-cluster buses in the partitioned implementation will be significantly shorter than the buses in the original non-partitioned one. This assumption may not hold for designs in which the area of any one cluster is too large. In the evaluation phase we first prune out unpromising partitions based on area estimates and compare the effectiveness of the remaining ones based on an estimate of the bus power.

The area estimates are based on distribution graphs [35]. A distribution graph displays the expected number of operations executed in each time slot. Figure 12(a) shows a simple algorithm and the corresponding distribution graph. For an algorithm with different types of operations, the total weighted distribution graph is obtained by summing up the distribution graphs of each operation type weighted by the area of the corresponding hardware. For clustered designs, distribution graphs are constructed for each cluster. We use the maximum height of the total weighted distribution graph as an estimate of the area. Using this metric, we see that though the number of edges cut by the partition (2 edges) is the same in Figs. 12(b) and (c), the area penalty is higher in the first case.

The first test prunes out candidate partitions that have an area larger than a user defined multiple of the area of the original design. For example, the partition shown in Fig. 12(b) would be pruned since the area of the second cluster is the same as the area of the original design (2 units).

The remaining candidate solutions are then compared, and the most promising one is selected. The comparison is based on a measure of the total bus power. For each cluster, the number of local data-

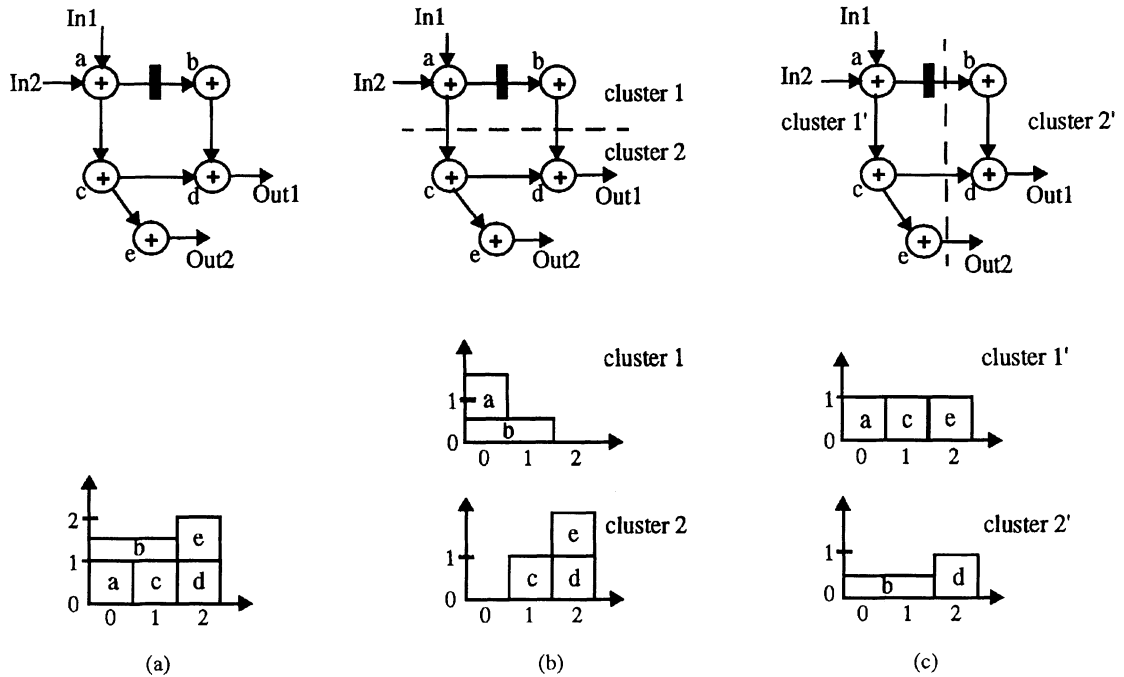


Figure 12. Distribution graphs for different candidate partitions of a given algorithm: (a) unpartitioned, (b) candidate partition 1, (c) candidate partition 2. Each operation's contribution is labeled on the distribution graphs.

transfers times the area of the cluster is a measure of the cluster's local bus power. Similarly, the number of global data transfers times the total area is a measure of the global bus power. These can be combined to define a measure of the total bus power. As discussed before, increasing the number of clusters trades-off global bus power for reduced power consumption in local buses. The bus power measure defined above evaluates this trade-off.

5. Low-Power Synthesis System

In this section we describe a new synthesis system based on the concept of spatially local assignment. We also present our models for estimating the bus power consumption for clustered designs.

5.1. Partitioning Based Synthesis — *Hyper-LP*

In this section we present our synthesis strategy which has been incorporated into the *Hyper-LP* system. While the basic flow is the same as that of the *Hyper* system [16], the core algorithms have been modified to incorporate the new partitioning based synthesis methodology for low power. Since we will use the

Hyper synthesis tools for comparisons with and evaluations of the *Hyper-LP* system, it is useful to briefly describe *Hyper*'s design flow and basic algorithms.

Given an algorithm and a throughput constraint, the *Hyper* system implements an architecture that minimizes the area. The *assignment* strategy uses a random initial assignment with iterative improvement and the *scheduling* uses an enhanced list-based strategy. The assignment and scheduling processes are repeated several times changing the allocation each time, until a feasible schedule is obtained and no area reduction is possible. The *allocation* starts with a minimal number of units and reallocates based on the results of the assignment and scheduling phase, adding or removing hardware based on a “badness” measure. The “badness” is a measure of how a given resource type (adder, multiplier, etc.) affects the scheduling difficulty (for further details see [36]). The basic idea is to add units of the resource type with the highest badness (they are most responsible for failures in the assignment/scheduling phase) and to remove those whose badness is the lowest. To optimize area the allocation sacrifices smaller units to save on larger ones. *Hyper*, however, does not optimize the interconnect.

Register and *bus assignment* use a graph coloring algorithm. For example, in the case of buses, timing

conflicts between different data transfers are represented in a conflict graph and a simple graph coloring heuristic is used for bus assignment. Once the schedule, assignment and allocation are fixed, the graph is fed into the hardware mapper which generates the final architecture description (e.g., a structural VHDL netlist).

The core of the *Hyper-LP* system is the partitioning methodology described in Section 4.4. Once the partitioning is complete, all operations have an associated cluster number. Also, each data transfer is classified as either global or local. The assignment technique is based on the random initial assignment with iterative improvement approach of the *Hyper* system with the added constraint that there is no hardware sharing between clusters. The scheduling algorithm is unchanged.

The concept of “badness” in the *Hyper* allocation scheme is extended to define a badness measure for each cluster. For a given type of unit and cluster, the “cluster-badness” is defined as the sum of the badnesses of all nodes of that type in the cluster. When the scheduling/assignment process fails, new units are allocated based on the badness measure. Once a unit is chosen for addition, the cluster to which it is added is selected based on the highest cluster-badness. Similarly, the cluster with the lowest cluster-badness is used when resources are being removed.

Bus assignment is also modified to account for the partitioning. In the construction of the conflict graph, conflict edges are added to account for partitioning constraints in addition to timing conflicts. A given pair of data transfers can only be merged onto the same bus if they are either both global transfers, or both local transfers in the same cluster.

The partitioning information is passed to the hardware mapping and floorplanning tools which place hardware units of a given partition close together in the final layout.

After the architectural netlist is generated, architectural-level power estimation is performed using the SPA tool [37]. SPA’s bus models were modified for computing the bus power in partitioned designs. This model is presented in the next section.

5.2. Bus-Length Estimation Models

The power consumption in the interconnect wiring is proportional to the number of accesses to the buses times the capacitance switched per access. The

number of times each bus is accessed is determined during SPA’s functional simulation step. The capacitance switched per access directly depends on the bus length, which is not determined until after placement and routing.

Buses fall into two main categories—those that connect units in different datapaths and those that connect units in the same datapath using over-the-cell routing. SPA uses two different models to estimate the lengths of the two types of buses. The estimation is performed hierarchically, estimating the intra-datapath lengths first, and using this information to calculate of the global bus lengths. Note that, in un-partitioned designs, units are merged into datapaths in a random fashion mostly governed by their sizes. In the partitioned designs, however, the merging, as also the floorplanning, is dictated by the partitioning. We first explain the bus lengths estimation model in SPA and then present our modifications to account for the partitioning.

For connections within a datapath, SPA estimates the lengths using a stacked datapath model. This model assumes that a given set of units are stacked into a single datapath and over-the-cell wiring is used for communication between them. It estimates the average length of over-the-cell connections as half the cumulative height of the units in the datapath. This is based on the assumption that routing between units increases the actual height of the datapath by approximately 20–30% and that most wire lengths are about 30–40% of the datapath height. The datapath area is simply the sum of the areas of the units.

SPA estimates the average global bus length as the square root of the estimated chip area. The chip area is based on an empirical model presented in [2]. The model is derived from the active area, A_{act} , (calculated by summing up areas of all the hardware units) and the total number of wires (number of buses between datapaths, N_{bus} , times the wordlengths, N_{bits}) as follows:

$$\text{Area} = \alpha_1 + \alpha_2 A_{act} + \alpha_3 N_{bits} N_{bus} A_{act} \quad (7)$$

The three terms in the model represent white space, active area of the components, and wiring area, respectively. Notice that the wiring area depends on the total number of wires and also on the active area. The coefficients, α_1 , α_2 , and α_3 , are derived statistically.

In the *Hyper-LP* system, hardware units are explicitly partitioned into clusters. The floorplanner places units that are in the same cluster physically close to each other. In particular, all units in a cluster, besides multipliers, are merged into a single datapath and

multipliers are placed nearby. We have modified the bus length estimation models in SPA to account for this floorplanning strategy.

Since multipliers cannot be stacked onto datapaths, the wire lengths for clusters with multipliers is calculated using a hybrid of the stacked and statistical models. For buses between any units other than the multipliers, lengths are estimated using the stacked datapath model. For the connections to the multiplier, lengths are given as the square-root of the estimated cluster area. The cluster area is estimated using the statistical model Eq. (7), where the active area is the sum of the areas of all the units in the cluster and the number of buses includes all buses that transfer data from the multiplier to any other unit in the same cluster. As in SPA, the global bus lengths are derived from the square-root of the chip area Eq. (7).

6. Results

In this section we present the results of our partitioning-based synthesis scheme. We compare implementations from the *Hyper* system with those from the new *Hyper-LP* system.

6.1. Cascade Filter

Figure 13 shows an eighth-order cascade IIR filter and the corresponding eigenvector placement. The spacings between the points in the placement clearly indicate the four clusters that are evident in the structure. Using $m + 3\sigma$ as the threshold to decide the points of

partition, we obtain clusters delimited by the arrows. It is interesting to note that $m + 2\sigma$ also works as a good threshold for this example.

The following experiment shows the gains achieved by exploiting the locality thus identified in the graph. Notice that all the multiplications in the design are multiplies with constant factors and can be converted into shift-and-add operations to avoid the use of area-intensive multipliers. The critical path of the resulting graph is 19 clock cycles, with both shifters and adders taking one clock cycle to execute. The speed constraint is set to be 21 clock cycles.

The *Hyper* implementation uses 4 adders and 3 shifters, whereas the *Hyper-LP* implementation uses 4 adders and 4 shifters (one adder and one shifter for each cluster). Table 3 compares the power dissipated in the two implementations. The bus power reduced

Table 3. Comparison of the power consumed for the *Hyper* and *Hyper-LP* implementations.

Component	<i>Hyper</i> implementation: power (mW)	<i>Hyper-LP</i> implementation: power (mW)	Percentage reduction
Functional unit	2.7	2.5	7.4
Register	5.3	4.2	20.8
Bus	2.0	0.4	80.0
Multiplexor	3.7	1.5	59.5
Buffer	1.0	0.9	10.0
Clock	0.7	0.7	0.0
Total	15.4	10.2	33.8

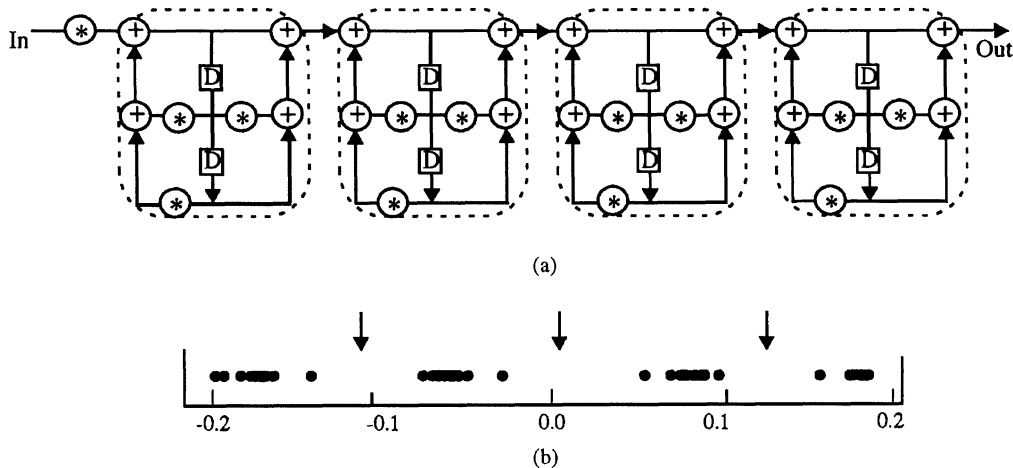


Figure 13. A eighth-order cascade-form IIR filter: (a) the structure, (b) corresponding eigenvector.

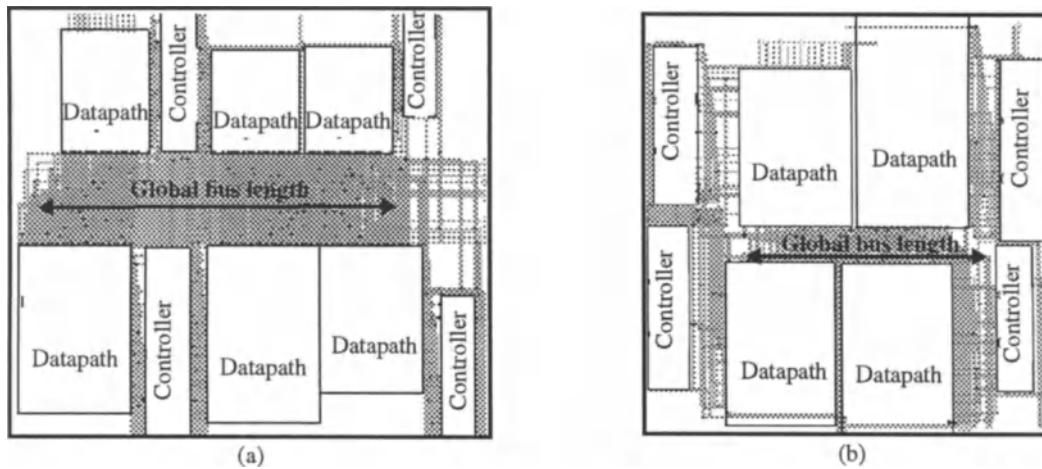


Figure 14. Layouts of the cascade filter: (a) non-local implementation from *Hyper*, (b) local implementation from *Hyper-LP*.

5-fold, from 2 mW to only 0.4 mW. The multiplexor power reduced by 60% since the partitioned design has less time-sharing of units. For this example, functional unit power reduced since localized data references resulted in improved data correlations. Buffer and register power also decreased due to factors such as improved data correlations, reduced physical capacitance, and fewer accesses. A 34% reduction in the total power consumption was realized. Notice that the contribution of interconnect (buses, multiplexors, and buffers) to the total power dissipation was reduced from 30% to 17%.

Magic layouts of the two implementations (Fig. 14) were obtained using the *Lager* silicon compiler and the *Flint* placement and routing tool [38]. The partitioning aids in the localization of computation enabling more compact layouts to be obtained. In the *Hyper* implementation, there are seven functional units that communicate heavily with each other (the layout tool has merged two units into the same datapath resulting in 6 datapaths). The *Hyper-LP* implementation has eight units divided into four clusters, with heavy

communication within each cluster and few connections between the clusters. All units in the same cluster are merged into one datapath.

The average length of the global buses is reduced by approximately 55%, from 2100 to 950 microns. A comparison of the estimated and measured bus lengths is presented in Table 4. Notice that the new models are conservative, overestimating the lengths of the buses.

6.2. Other Examples

In this section we present results of our partitioning based scheme for a set of DSP filter and transform examples. Some are in their original form (DCT, FFT, and parallel-form IIR) and others are transformed using either constant multiplication expansion (cascade-form IIR, direct-form IIR, and wavelet) or retiming (wave digital filter).

Table 5 shows the number of bus accesses, number of multiplexor accesses, and the estimated bus lengths in designs from the two systems. The accesses to global

Table 4. Comparison of the estimated and measured average bus-lengths.

	<i>Hyper</i>	<i>Hyper-LP</i>				Global
		Cluster 1	Cluster 2	Cluster 3	Cluster 4	
Estimated length	2.82	0.41	0.45	0.54	0.44	2.73
Measured length	2.10	2.23	0.29	0.46	0.32	0.95
Percentage difference	25.5	43.4	36.0	14.4	27.4	65.3

Table 5. Comparison of bus accesses and lengths in *Hyper* and *Hyper-LP* designs.

Name	<i>Hyper</i>			<i>Hyper-LP</i>						
	Bus accesses	Bus length*	Mux accesses	Number of clusters	Global bus accesses	Local bus accesses	Global bus length*	Average local bus length*	Bus length's ratio	Mux accesses
Cascade	106	2.97	279	4	9	93	2.73	0.46	0.17	164
Direct form	202	8.95	1319	3	6	197	9.04	2.86	0.32	757
Wavelet	71	4.25	247	2	3	69	4.49	1.79	0.40	195
Wave digital	57	2.33	156	2	2	54	2.54	0.70	0.28	110
DCT	58	6.19	119	2	4	54	5.89	2.89	0.49	85
FFT	38	6.32	58	2	3	35	6.51	1.62	0.25	44
Parallel IIR	40	4.67	82	2	2	39	5.42	2.43	0.45	69

*Bus lengths are in millimeters for a 1.2 micron technology.

Table 6. Comparison of power consumption (mW) in *Hyper* and *Hyper-LP* designs.

Name	<i>Hyper</i>				<i>Hyper-LP</i>				Percentage reduction		
	Bus Power	Mux Power	Buffer Power	Total power	Bus power	Mux power	Buffer power	Total power	Bus power	Mux power	Total power
Cascade	2.0	3.2	1.0	21.3	0.4	1.5	0.9	16.3	80.0	59.6	25.9
Direct form	29.8	38	4.8	144.6	10.3	21.1	4.5	110.4	65.4	44.5	23.7
Wavelet	3.6	7.7	1.8	42.3	2.6	5.1	1.7	37.4	53.6	33.8	11.6
Wave digital	1.5	3.2	0.9	20.4	0.5	1.7	0.8	18.0	33.3	46.9	11.8
DCT	9.0	3.8	1.3	41.5	4.5	2.3	1.9	37.2	50.0	39.5	10.36
FFT	17.6	4.7	2.1	48.6	5.2	3.8	2.5	36.8	69.7	19.1	24.3
Parallel IIR	15.1	2.8	1.3	57.5	3.2	2.2	2.0	48.8	78.7	21.4	15.1
								Average	61.5	37.8	17.5

buses is reduced drastically for all examples with very little change in the lengths of these buses. Exploiting spatial locality moves almost all of the bus accesses to intra-cluster buses whose lengths are 50 to 75 percent shorter than those of the global buses. In general, due to reduced hardware sharing, there is a decrease in the multiplexor accesses for all examples except one. The total number of buffer accesses is unchanged because a buffer is used to drive every data transfer.

Table 6 shows the resulting power consumption for the examples. The *Hyper-LP* implementations uniformly dissipate lower power than the *Hyper* implementations. Power consumed by buses is reduced drastically in all examples (up to 80%). Furthermore, the power dissipated in multiplexors is reduced — up to 60% reduction in multiplexor power is seen due to reduced and more localized hardware sharing. The total chip power reduces up to 30%.

We expect buffer power to decrease since smaller buffers can be used to drive the data transfers occurring on short local buses. However, our architecture-netlist generation tool currently uses minimum-sized buffers for all data transfers, regardless of bus length, and therefore, our results show negligible change in buffer power. With necessary modifications, buffer power should contribute toward further reduction in total power.

These experiments have demonstrated that restricting hardware sharing to behaviorally localized operations in the algorithm results in a large reduction in the interconnect power. However, this does not necessarily come free of cost and in several cases an area penalty must be paid. An increase in the number of functional units does not necessarily translate into an equivalent increase in overall area. Since the communications across the chip are localized, the design

Table 7. Area penalty.

Name	<i>Hyper</i>					<i>Hyper-LP</i>					Percentage reduction	
	Units			Estimated active area (mm ²)	Estimated total chip area (mm ²)	Units			Estimated active area (mm ²)	Estimated total chip area (mm ²)	Active area	Total area
	*	+	»			*	+	»				
† Cascade	0	4	3	8.79	14.88	0	4	4	7.46	7.83	-15.1	-47.4
Direct form	0	15	12	11.77	80.20	0	23	12	12.00	81.71	+2.0	+1.9
Wavelet	0	6	6	3.88	18.07	0	9	6	4.34	19.95	11.8	+10.4
Wave digital	0	3	2	1.46	5.46	0	4	3	1.71	6.46	17.1	+18.3
DCT	8	12	0	7.18	38.36	6	14	0	6.13	34.75	-14.6	-9.4
FFT	2	7	0	6.31	39.95	2	8	0	6.42	42.39	+1.7	+6.1
Parallel IIR	4	3	0	6.04	21.85	5	6	0	7.52	29.43	+24.5	+34.7
										Average	+3.9	+14.6

† Area numbers for this example are from layouts (not estimated).

is more conducive to compact layout. This is due to not only reduced global connections and smaller local buses but also due to fewer overhead elements such as multiplexors and buffers. As was seen for the cascade filter of Section 6.1, the new layout may actually be smaller than the original. Table 7 shows the estimated area penalty obtained in the *Hyper-LP* designs. It is seen that the impact of the increase in functional units is not proportionately reflected in the total area. In most cases, therefore, the total chip area is marginally affected. For one example, the parallel-form IIR filter, about 34.7% area penalty is seen. In two of the examples, the area is reduced by 47 and 9%. In fact, for the DCT example, the number of components required was also reduced! This is because partitioning the example into localized regions serves as a guidance to the assignment tool, and it is able to find a better solution. Note that the assignment tool is heuristic and, therefore, not guaranteed to find the global minimum.

The examples used in these experiments are small and have been divided into few clusters (two or three). We feel that the advantage will be higher for larger examples which can be divided into more clusters.

In summary, exploiting locality of algorithms during the high-level synthesis process greatly reduces interconnect power. For most examples a significant reduction in total chip power is obtained. Though the number of functional units required is increased due to restricted hardware sharing, the penalty in the total chip area is marginal.

7. Conclusions

The architecture synthesis process can have a large impact on the power dissipated in a design. We have analyzed the effect of various synthesis tasks on the power consumption of the different components of a chip: the functional units, memory, interconnect, and control. In the process, previous work done in the field was surveyed and opportunity areas for research identified.

We have presented a new technique for power reduction based on exploiting the locality in a given application. It was seen that preserving the locality improves the implementation in a variety of different ways. The predominant effect is the reduction of accesses to highly capacitive global buses. Our results showed up to 80% improvement in the power consumed in buses. Additionally, restricting hardware sharing led to reduced usage of multiplexors. Though the power savings can come at the cost of increased area, this effect is marginal. The techniques have been integrated into the *Hyper-LP* system.

The concept of preserving locality is a special case of a more general class of techniques referred to as distributed computing. In general, accesses to global computing resources — controllers, buses, memory, I/O — are expensive due to increased capacitance. Dividing these resources reduces the capacitance being switched per access. This work can therefore be extended to more general applications such as memory partitioning and processor partitioning.

Acknowledgments

The authors are grateful to Bruce Hendrickson and Rob Leland of the Sandia National Laboratories for providing their implementation of the Lanczos method for calculating eigenvectors [32] and to anonymous reviewers for their helpful comments. Ms. Mehra is supported by the ARPA grant J-FBI 93-153 and Ms. Guerra is supported by scholarships from AT&T and the Office of Naval Research. Their support is gratefully acknowledged.

References

1. A. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey, and R.W. Brodersen, "Optimizing power using transformations," *IEEE Transactions on CAD*, pp. 12–31, Jan. 1995.
2. R. Mehra and J.M. Rabaey, "Behavioral level power estimation and exploration," *Proceedings of the International Workshop on Low-Power Design*, pp. 197–202, April 1994.
3. K. Keutzer and P. Vanbekbergen, "Impact of CAD on the design of low power digital circuits," *Symposium on Low Power Electronics*, pp. 42–45, Oct. 1994.
4. A. Raghunathan and N.K. Jha, "Behavioral synthesis for low-power," *Proceedings of the International Conference on Computer Design*, pp. 318–322, Oct. 1994.
5. E. Musoll and J. Cortadella, "High-level synthesis techniques for reducing the activity of functional units," *Proceedings of the International Symposium on Low-Power Design*, pp. 99–104, April 1995.
6. A. Chatterjee and R. Roy, "Synthesis of low power linear DSP circuits using activity metrics," *Proceedings of the International Conference of VLSI Design*, pp. 265–270, Jan. 1994.
7. J.M. Chang and M. Pedram, "Register allocation and binding for low power," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 29–35, June 1995.
8. D.D. Gajski, *High-Level Synthesis: Introduction to Chip and System Design*, Kluwer Academic Publishers, Boston, 1992.
9. R.A. Walker and R. Camposano, *A Survey of High Level Synthesis Systems*, Kluwer Academic Publishers, Boston, 1991.
10. H.J.M. Veendrick, "Short-circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits," *IEEE Journal of Solid-State Circuits*, pp. 468–473, Aug. 1984.
11. A. Raghunathan and N.K. Jha, "An iterative improvement algorithm for low power data path synthesis," *Proceedings of the International Conference on CAD*, pp. 597–602, Nov. 1995.
12. L. Goodby, A. Orailoglu, and P.M. Chau, "Microarchitectural synthesis of performance constrained, low-power VLSI designs," *Proceedings of the International Conference on Computer Design*, pp. 323–326, Oct. 1994.
13. F. Catthoor, F. Franssen, S. Wuytack, L. Nachtergaele, and H. De Man, "Global communication and memory optimizing transformations for low-power signal processing systems," *VLSI Signal Processing Workshop*, pp. 178–187, Oct. 1994.
14. A.H. Farrahi, G.E. Tellez, and M. Sarrafzadeh, "Memory segmentation to exploit sleep mode operation," *Proceedings of ACM/IEEE Design Automation Conference*, pp. 36–41, June 1995.
15. S. Wu, "A hardware library representation for the hyper synthesis system," Masters' Thesis, University of California, Berkeley, Memorandum No. UCB/ERL M94/47, June 1994.
16. J.M. Rabaey, C. Chu, P. Hoang, and M. Potkonjak, "Fast prototyping of datapath-intensive architectures," *IEEE Design & Test of Computers*, pp. 40–51, June 1991.
17. M.C. McFarland and T.J. Kowalski, "Incorporating bottom-up design into hardware synthesis," *IEEE Transactions on CAD*, Vol. 9, No. 9, pp. 938–949, Sept. 1990.
18. E.D. Lagnese and D.E. Thomas, "Architectural partitioning for system level synthesis of integrated circuits," *IEEE Transactions on CAD*, Vol. 10, No. 7, pp. 847–860, July 1991.
19. B.W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *Bell System Technical Journal*, Vol. 49, pp. 291–307, Feb. 1970.
20. C.M. Fiduccia and R.M. Mattheyses, "A linear-time heuristic for improving network partitions," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 175–181, June 1982.
21. S. Kirkpatrick, C.D. Gelatt, and M.P. Vecell, "Optimization by simulated annealing," *Science*, Vol. 220, No. 4598, pp. 671–680, May 1983.
22. G. Vijayan, "Partitioning logic on graph structures to minimize routing cost," *IEEE Transactions on CAD*, pp. 1326–1334, Dec. 1990.
23. W.E. Donath, "Logic partitioning," *Physical Design Automation of VLSI Systems*, (Eds.) B. Preas and M. Lorenzetti, Benjamin/Cummings, pp. 65–86, 1988.
24. D.G. Schweikert and B.W. Kernighan, "A proper model for the partitioning of electrical circuits," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 57–62, 1972.
25. Y.C. Wei and C.K. Cheng, "Ratio cut partitioning for hierarchical designs," *IEEE Transactions on CAD*, Vol. 10, pp. 911–921, July 1991.
26. L. Hagen and A.B. Kahng, "New spectral methods for ratio cut partitioning and clustering," *IEEE Transactions on CAD*, Vol. 11, No. 9, pp. 1074–1085, Sept. 1992.
27. C.J. Alpert and A.B. Kahng, "Geometric embeddings for faster and better multi-way netlist partitioning," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 743–748, 1993.
28. E.R. Barnes, "An algorithm for partitioning the nodes of a graph," *Siam Journal of Algorithms and Discrete Methods*, Vol. 3, No. 4, pp. 541–549, 1994.
29. P.K. Chan, M.D.F. Schlag, and J. Zien, "Spectral K -way ratio-cut partitioning and clustering," *IEEE Transactions on CAD*, Vol. 13, No. 9, pp. 1088–1096, 1994.
30. J. Frankle and R.M. Karp, "Circuit placement and cost bounds by eigenvector decomposition," *Proceedings of the International Conference on CAD*, pp. 414–417, 1986.
31. H.D. Simon, "Partitioning of unstructured problems for parallel processing," *Computing Systems in Engineering*, Vol. 2, No. 2/3, pp. 135–148, 1991.
32. B. Hendrickson and R. Leland, "The Chaco user's guide, V. 1.0," Tech. Report SAND93-2339, Sandia National Lab., Oct. 1993.
33. K.M. Hall, "An r -dimensional quadratic placement algorithm," *Management Science*, Vol. 17, No. 3, pp. 219–229, Nov. 1970.
34. T. Lengauer, *Combinatorial Algorithms for Integrated Circuit Layout*, Wiley-Teubner, Chichester, U.K., 1990.

35. P.G. Paulin and J.P. Knight, "Force-directed scheduling for behavioral synthesis of ASIC's," *IEEE Transactions on CAD*, Vol. 8, No. 6, pp. 661-679, June 1989.
36. M. Potkonjak and J.M. Rabaey, "Scheduling algorithms for hierarchical data control flow graphs," *International Journal of Circuit Theory and Applications*, Vol. 20, No. 3, pp. 217-233, May-June 1992.
37. P.E. Landman and J.M. Rabaey, "Architectural power analysis: The dual bit type method," *IEEE Transactions on VLSI Systems*, Vol. 3, No. 2, pp. 173-187, June 1995.
38. R.W. Brodersen, (ed.), *Anatomy of a Silicon Compiler*, Kluwer Academic Publishers, Boston, 1992.



Renu Mehra received the Bachelor of Technology (B.Tech.) degree in Electrical Engineering in 1991 from the Indian Institute of Technology, Kanpur, India. She received the M.S. degree in Electrical Engineering and Computer Science with a specialization in IC design from the University of California, Berkeley where she is currently a Ph.D. candidate.

Currently, she is working on high-level estimation, synthesis and methodologies for low-power design. Her research interests include low-power techniques at all levels of design including the algorithm, architecture and circuit levels. She is also interested in the IC design, high-level and system-level synthesis and architectures for DSP systems.



Lisa M. Guerra received the B.S. degree in electrical engineering from Stanford University in 1990. She is currently a doctoral

candidate at the University of California, Berkeley. In 1991, she was awarded doctoral fellowships from the Office of Naval Research and AT&T. Here research interests include design methodologies, computer-aided analysis and estimation, and high-level and system-level synthesis.



Jan M. Rabaey received the EE and Ph.D. degrees in applied sciences from the Katholieke Universiteit Leuven, Belgium, respectively in 1978 and 1983. From 1983 till 1985, he was connected to the University of California, Berkeley as a Visiting Research Engineer. From 1985 till 1987, he was a research manager at IMEC, Belgium, where he pioneered the development of the CATHEDRALII synthesis system for digital signal processing. In 1987, he joined the faculty of the Electrical Engineering and Computer Science department of the University of California, Berkeley, where he is now a professor.

Jan Rabaey authored or co-authored more than 100 papers in the area of signal processing and design automation. He received numerous scientific awards, and has been on the technical program committees of conferences, such as ISSCC, ICCAD, EDAC. He is currently serving on the executive committee of the DAC conference. He has served as associate editor of the *IEEE Journal of Solid State Circuits*.

His current research interests include the exploration of architectures and algorithms for digital signal processing systems and their interaction. He is furthermore active in various aspects of portable, distributed communication and computation systems, including low power design, networking and design applications.

Techniques for Power Estimation and Optimization at the Logic Level: A Survey

JOSÉ MONTEIRO AND SRINIVAS DEVADAS
Department of EECS, MIT, Cambridge, MA 02139

Received October 27, 1995; Revised March 25, 1996

Abstract. We present a survey of state-of-the-art power estimation methods and optimization techniques targeting low power VLSI circuits. Estimation and optimizations at the circuit and logic levels are considered.

1. Introduction

Rapid increases in chip complexity, increasingly faster clocks, and the proliferation of portable devices have combined to make power dissipation an important design parameter. The power dissipated by a digital system determines its heat dissipation characteristics as well as battery life. Power reduction techniques have been proposed at all levels—from system to device. These techniques require efficient and accurate power estimation methodologies. In this paper we present a survey of estimation and optimization techniques used in the design of low-power VLSI circuits, focusing primarily on logic level abstractions of circuit behavior.

2. Power Estimation

For power to be used as a design parameter tools are needed that can efficiently estimate the power consumption of a given design. As in most engineering problems we have tradeoffs, in this case between the accuracy and running time of the tool.

Accurate power values can be obtained from circuit-level simulators such as SPICE [1]. In practice, these simulators cannot be used in circuits with more than a few thousand transistors, so their applicability in logic design is very limited—they are essentially used to characterize simple logic cells.

A good compromise between accuracy and complexity is switch-level simulation. Simulation of entire chips can be done within reasonable amounts of CPU

time [2, 3]. This property makes switch-level simulators very important power diagnosis tools. After layout and before fabrication these tools can be used to identify *hot spots* in the design, i.e., areas in the circuit where temperature may exceed the safety limits during normal operation.

At gate-level, a more simplified power dissipation model is used, leading to a fast power estimation process. Although detailed circuit behavior is not modeled, the estimation values can still be reasonably accurate. Obtaining fast power estimates is critical in order to allow a designer to compare different designs. Further, for the purpose of directing a designer or a synthesis tool for low power design, rather than an absolute measure of how much power a particular circuit consumes, an accurate relative power measure between two designs will suffice.

2.1. Power Dissipation Model

The sources of power dissipation in CMOS devices are summarized by the following expression [4]:

$$P = \frac{1}{2} \cdot C \cdot V_{DD}^2 \cdot f \cdot N + Q_{SC} \cdot V_{DD} \cdot f \cdot N + I_{leak} \cdot V_{DD} \quad (1)$$

where P denotes the total power, V_{DD} is the supply voltage, and f is the frequency of operation.

The first term in Eq. (1) corresponds to the power involved in charging and discharging circuit nodes. C represents the node capacitances and N is the switching

activity, i.e., the number of gate output transitions per clock cycle (also known as *transition density* [5]).

The second term in Eq. (1) represents the power dissipation during output transitions due to current flowing directly from the supply to ground. This current is often called *short-circuit current*. The factor Q_{SC} represents the quantity of charge carried by the short-circuit current per transition.

The third term in Eq. (1) is related to the static power dissipation due to leakage current I_{leak} . The transistor source and drain diffusions form parasitic diodes with bulk regions. Reverse bias currents in these diodes dissipate power. Subthreshold transistor currents also dissipate power.

These three factors for power dissipation are often referred to as *switching activity power*, *short-circuit power* and *leakage current power* respectively.

It has been shown [6] that for well designed CMOS circuits the switching activity power accounts for over 90% of the total power dissipation. Most optimization techniques at different levels of abstraction target minimal switching activity power. The model for power dissipation for a gate i in a logic circuit is thus simplified to:

$$P_i = \frac{1}{2} \cdot C_i \cdot V_{DD}^2 \cdot f \cdot N_i \quad (2)$$

The supply voltage V_{DD} and the clock frequency f are defined prior to logic design and the capacitive load C_i can be extracted from the circuit. Therefore, the problem of logic level power estimation reduces to obtaining an accurate estimate of the number of transitions N_i for each gate in the circuit. In the remainder of this section we present existing techniques for efficient computation of switching activity in logic circuits.

2.2. Switching Activity Estimation

The techniques we present in this section target *average* switching activity estimation. This is typically the value used to guide optimization methods for low power.

Some work has been done on identifying and computing conditions which lead to *maximum* power dissipation. In [7] a technique is presented that implicitly determines the two input vector sequence that leads to maximum power dissipation in a combinational circuit. More recently, in [8] a method for computing the multiple vector cycle in a sequential circuit that dissipates maximum average power is described.

2.2.1. Simulation-Based Techniques. A straightforward approach to obtain an average transition count at every gate in the circuit is to use a logic simulator and simulate the circuit for a *sufficiently large* number of randomly generated input vectors. The main advantage of this approach is that existing logic simulators can be used directly and issues like glitching and internal correlation are automatically taken into account by the logic simulator.

The most important aspect of simulation-based switching activity estimation is deciding how many input vectors to simulate in order to achieve a given accuracy level. A basic assumption is that under random inputs the power consumed by a circuit over a period of time T has a Normal distribution. Given a user-specified allowed percentage error ϵ and confidence level α , the approach described in [9] uses the Central Limit Theorem [10] to compute the number of input vectors with which to simulate the circuit with. With $(1 - \alpha) \times 100\%$ confidence, $|\bar{p} - P| < z_{\frac{\alpha}{2}} s / \sqrt{N}$, where \bar{p} and s are the measured average and standard deviation of the power, P is the true average power dissipation, N the number of input vectors and $z_{\frac{\alpha}{2}}$ obtained from the Normal distribution. Since we require $\frac{|\bar{p} - P|}{\bar{p}} < \epsilon$, it follows that

$$N \geq \left(\frac{z_{\frac{\alpha}{2}} s}{\epsilon \bar{p}} \right)^2 \quad (3)$$

For a typical logic circuit and reasonable error and confidence levels, the numbers of vectors needed is usually small, making this approach very efficient.

A limitation of the technique presented in [9] is that it only guarantees accuracy for the *average* switching activity over all the gates. The switching activity values (N_i in Eq. (2)) for individual gates may have large errors and these values are important for many optimization techniques.

In [11] the authors augment this method by allowing the user to specify the percentage error and confidence level for the switching activity of individual gates. Equation (3) is used for each node in the circuit, where instead of power, the average and standard deviation of the number of transitions in the node is the relevant parameter. The number of input vectors N is obtained as the minimum N that verifies the equations for all the nodes.

The problem now is that gates which have a low switching probability, *low-density nodes*, may require a very large number of input vectors in order for the

estimation to be within the percentage error specified by the user. The authors solve this problem by being less restrictive for these gates: an absolute error bound is used instead of the percentage error. The impact of possible larger errors for low-density nodes is minimized by the fact that these gates have the least effect on power dissipation and circuit reliability.

Other methods [12] try to compute a more tight bound on the number of input vectors to simulate. Instead of relying on normal distribution properties the authors assume the number of transitions at the output of a gate to have a multinomial distribution. Yet this method has to make a number of empirical approximations in order to obtain the number of input vectors.

Simulation-based techniques can be very efficient for loose accuracy bounds. Increasing the accuracy may require a prohibitively high number of simulation vectors.

2.2.2. Issues in Probabilistic Estimation Techniques.

Given some statistical information of the inputs, such as static and/or transition probabilities, probabilistic methods propagate these probabilities through the logic circuit obtaining static and/or transition probabilities at each node in the circuit. Only one pass through the circuit is needed making these methods potentially very efficient. However modeling issues like correlation between signals can make these methods computationally expensive.

The *static* probability P_s of a logic signal x is the probability of x being 0 or 1 at any instant (we will represent this, respectively, as $P_s(\bar{x})$ and $P_s(x)$). *Transition* probabilities are the probability of x making a 0 to 1 or 1 to 0 transition, staying at 0 or staying at 1 between two time instants. We will represent these probabilities as $P_t^{01}(x)$, $P_t^{10}(x)$, $P_t^{00}(x)$ and $P_t^{11}(x)$, respectively. Note that we always have $P_t^{01}(x) = P_t^{10}(x)$. The probability that signal x makes a transition is $P_t(x) = P_t^{01}(x) + P_t^{10}(x)$. Relating to Eq. (2), $N_x = P_t(x)$.

Static probabilities can always be derived from transition probabilities:

$$\begin{aligned} P_s(x) &= P_t^{11}(x) + P_t^{01}(x) \\ P_s(\bar{x}) &= P_t^{00}(x) + P_t^{10}(x) \end{aligned}$$

Derivation in the other direction is only possible if we are given the correlation coefficients between successive values at an input. If we assume they are inde-

pendent then:

$$\begin{aligned} P_t^{11}(x) &= P_s(x)P_s(x) \\ P_t^{10}(x) &= P_s(x)P_s(\bar{x}) \\ P_t^{01}(x) &= P_s(\bar{x})P_s(x) \\ P_t^{00}(x) &= P_s(\bar{x})P_s(\bar{x}) \end{aligned}$$

In the case of dynamic precharged circuits, exemplified in Fig. 1(a), the switching activity is uniquely determined by the applied input vector. If both x and y are 0, then z stays at 0 and there is no switching activity. If one or both of x and y are 1, then z goes to 1 during the evaluation phase and back to 0 during precharging. Therefore, the switching activity at z will be twice the *static* probability of z being 1 ($N_z = 2P_s(z)$).

On the other hand, the switching activity in static CMOS circuits is a function of a two input vector sequence. For instance, consider the circuit shown in Fig. 1(b). In order to determine if the output f switches we need to know what value it assumed after the first input vector and to what value it evaluated after the second input vector. Using static probabilities one can compute the probability that f evaluates to 1, $P_s(f)$, for the first and second input vectors. Then:

$$\begin{aligned} P_t(f) &= P_{s,1}(f)P_{s,2}(\bar{f}) + P_{s,1}(\bar{f})P_{s,2}(f) \\ &= P_s(f)(1 - P_s(f)) + (1 - P_s(f))P_s(f) \\ &= 2P_s(f)(1 - P_s(f)) \end{aligned}$$

since $P_{s,1}(f) = P_{s,2}(f) = P_s(f)$ and $P_s(\bar{f}) = 1 - P_s(f)$.

By using static probabilities in the previous expression we ignored any correlation between the two vectors in the input sequence. In general ignoring this type of correlation, called *temporal correlation*, is not a valid assumption. Probabilistic estimation methods work with transition probabilities at the inputs, thus introducing the necessary correlation between input vectors.

Another type of correlation is *spatial correlation*. The probability of two or more signals being 1 may not be independent. Spatial correlation of input signals, even if known, can be difficult to specify, so most probabilistic techniques assume the inputs to be spatially independent. In Subsection 2.2.5 we review methods that try to take into account input signal correlation.

Even if independence is assumed for input signals, logic circuits with reconvergent fanout introduce spatial correlation between internal signals. Consider the

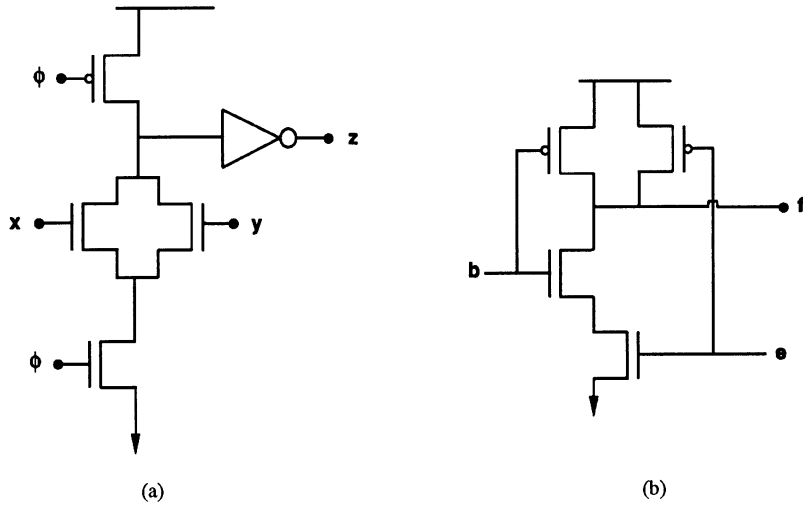


Figure 1. Dynamic vs. static circuits.

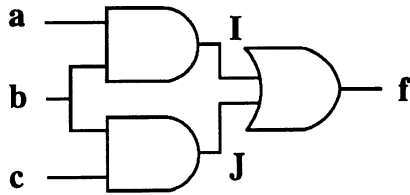


Figure 2. Spatial correlation between internal signals.

circuit in Fig. 2. Assuming that inputs a , b and c are uncorrelated, the static probability at \mathcal{I} is $P_s(\mathcal{I}) = P_s(a)P_s(b)$ and at \mathcal{J} is $P_s(\mathcal{J}) = P_s(b)P_s(c)$. However, $P_s(f) \neq P_s(\mathcal{I}) + P_s(\mathcal{J}) - P_s(\mathcal{I})P_s(\mathcal{J})$ because \mathcal{I} and \mathcal{J} are correlated ($b=0 \Rightarrow \mathcal{I}=\mathcal{J}=0$).

To compute accurate signal probabilities, we need to take into account this internal spatial correlation. One solution to this problem is to write the Boolean function as a disjoint sum-of-products expression where each product-term has a null intersection with any other. For the previous example, we write f as:

$$f = (a \wedge b) \vee (b \wedge c) \\ = (a \wedge b) \vee (\bar{a} \wedge b \wedge c)$$

Then $P_s(f) = P_s(a)P_s(b) + P_s(\bar{a})P_s(b)P_s(c)$.

A more efficient method is to use Binary Decision Diagrams (BDDs) [13]. The static probabilities can be computed in time linear in the size of the BDD by traversing the BDD from leaves to root, since the BDD implements a disjoint cover with sharing. The previous example is illustrated in Fig. 3.

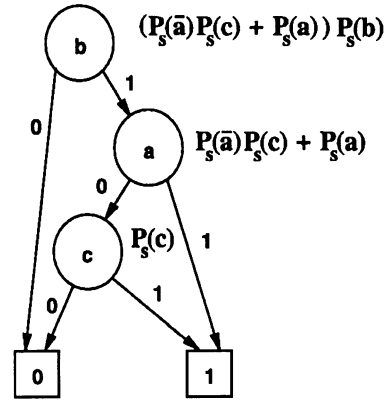


Figure 3. Computing static probabilities using BDDs.

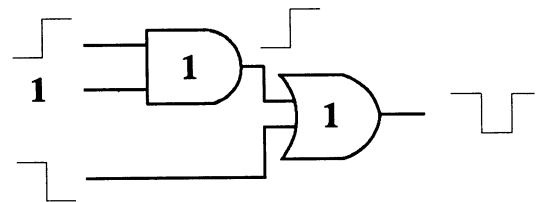


Figure 4. Glitching due to different input path delays.

Yet another issue is spurious transitions (or glitching) at the output of a gate due to different input path delays. These may cause the gate to switch more than once during a clock cycle, as exemplified in Fig. 4. Studies have shown that glitching cannot be ignored as it can be a significant fraction of the total switching activity [14, 15].

2.2.3. Probabilistic Techniques. An approach to computing switching activity using probabilities was presented in [16]. Static probabilities of the input signals are propagated through the logic gates in the circuit. In this straightforward approach, a zero delay model is assumed thus glitching is not computed and since static probabilities are used no temporal signal correlation is taken into account. Further, spatial correlation is also ignored as signals at the input of each gate are assumed to be independent.

In [5] a technique is presented that propagates *transition densities* (which for a zero-delay model is equivalent transition probabilities, $(D(x) = P_i(x))$). The authors show that the transition density at the output f of a logic gate with n *uncorrelated* inputs x_i can be computed as:

$$D(f) = \sum_{i=1}^n P_s \left(\frac{\partial f}{\partial x_i} \right) D(x_i)$$

where $\frac{\partial f}{\partial x_i}$ are the combinations for which the value of f depends on the value of x_i and is given by:

$$\frac{\partial f}{\partial x_i} = f|_{x_i=1} \oplus f|_{x_i=0}$$

That is, the switching activity at the output is the sum of the switching activity of each input weighted by the probability that a transition at this input is propagated to the output.

Implicit to this technique is also a zero delay model. An attempt to take glitching into account is suggested by decoupling delays from the logic gate and computing transition densities at each different time point where inputs may switch.

A major shortcoming of this method is the assumption of spatial independence of the input signals to each gate. [17] extends the work of [5] by partially solving this spatial correlation problem. The logic circuit is partitioned in order to compute accurate transition densities at some nodes in the circuit. For each partition, spatial correlation is taken into account by using BDDs.

A similar technique, introduced in [18], uses the notion of *transition waveform*. A transition waveform, illustrated in Fig. 5, represents an average of all possible signal waveforms at a given input. The example of Fig. 5 shows that there are no transitions between instants 0 and t_1 and that during this interval half of the possible waveforms are at 1. At instant t_1 a fraction of 0.2 of the waveforms make a 0 to 1 transition,

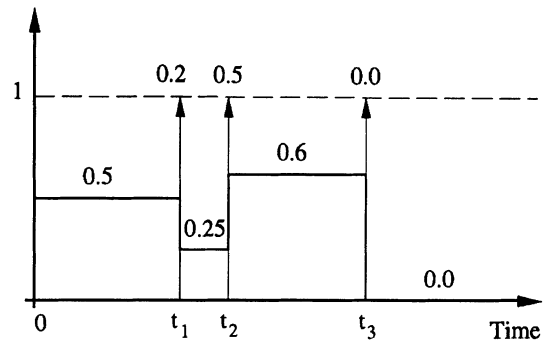


Figure 5. Example of a transition waveform.

leaving a quarter of the waveforms at 1 (which implies that a fraction of 0.45 of the waveforms make a 1 to 0 transition). A transition waveform basically has all the information about static and transition probabilities of signals and how these probabilities change in time. Their main advantage is to allow an efficient computation of glitching. Transition waveforms are propagated through the logic circuit in much the same way as transition densities.

Again, transition waveform techniques are not able to handle spatial correlations. Another method based on transition waveforms is proposed in [19] where *correlation coefficients* between internal signals are computed beforehand and then used when propagating the transition waveforms. These coefficients are computed for pairs of signals (from their logic AND) and are based on steady state conditions. This way some spatial correlation is taken into account.

More recently, a new approach for probabilistic switching activity estimation was presented in [20]. In this work, a numerical expression is obtained for the correlation between the input signals of a logic gate. The switching probability of the output is computed approximately by using the first order terms of the Taylor expansion of the correlation expression, which can be done efficiently. Higher accuracy can be achieved by using higher order terms of the Taylor expansion at the cost of longer computational time.

A different switching activity estimation technique based on *symbolic simulation* is presented in [21]. A symbolic network is built which has the Boolean conditions for all values that each node in the original network may assume at different time instants given an input vector pair. If a zero delay model is used, the symbolic network corresponds to two copies of the original network, one copy evaluated with the first input vector and the other copy with the second. EXOR gates

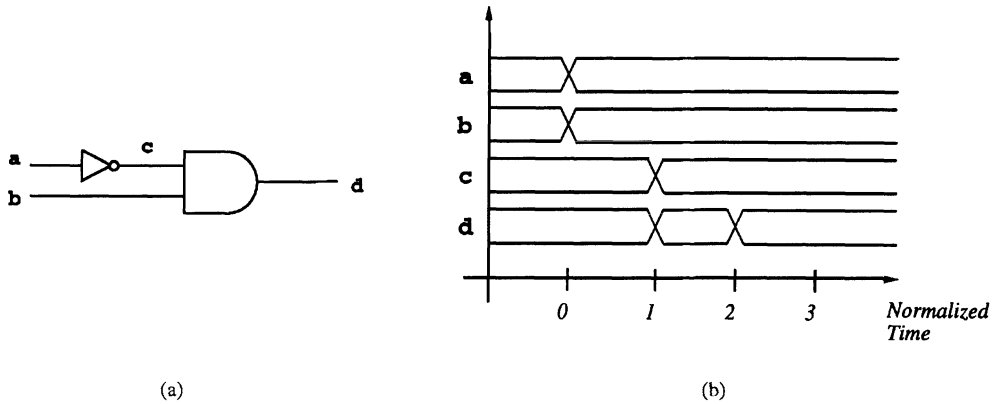


Figure 6. Example circuit for symbolic simulation.

are added between pairs of nodes, one from each copy, that correspond to the same node in the original circuit. The output of an EXOR evaluating to a 1 indicates that for this input vector pair the corresponding node in the original circuit makes one transition (since it evaluates to a different value for each of the input vectors).

If unit or general delay models are used, the symbolic network will have nodes corresponding to all intermediate values that each signal may assume. In this case, the EXOR gates will be connected to nodes corresponding to consecutive time instants and relating to the same node in the original circuit. To exemplify how the symbolic network is built, consider the simple logic circuit depicted in Fig. 6(a). If inputs a and b change at instant 0, and assuming a unit delay model, node c may change at instant 1 and node d may change at instants 1 and 2, as shown in Fig. 6(b). The symbolic network corresponding to this circuit is presented in Fig. 7. We have inputs a_0 and b_0 from the first input vector and inputs a_t and b_t from the second input vector. Nodes c_0 and d_0 are the initial values of nodes c and d respectively. At instant 1, node c will have the value c_{t+1} and d the value d_{t+1} . $c_0 \oplus c_{t+1}$ evaluates to 1 only if node c makes a transition at instant 1. Similarly for node d . At instant 2, node d will assume the value d_{t+2} . Again EXORing d_{t+1} and d_{t+2} gives the condition for d to switch at instant 2. Thus the total switching at d will be the sum of d_{x1} and d_{x2} .

Once the symbolic network of a circuit is computed, the method uses the static probabilities of the inputs to obtain the static probabilities of the output of the EXORs in this network evaluating to 1. This probability is the same as the switching probability of the nodes in the original circuit.

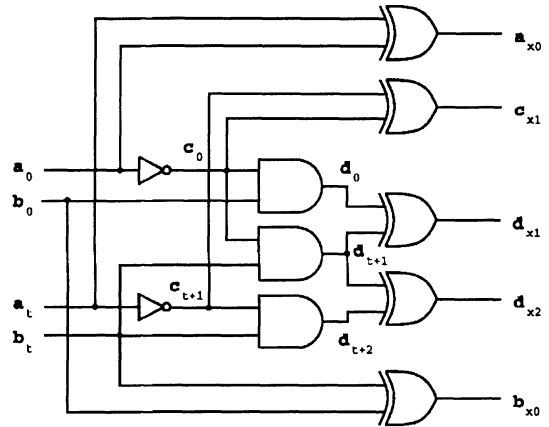


Figure 7. Symbolic network.

This method models glitching accurately and, if BDDs are used to compute the static probabilities, exact spatial correlation is implicitly taken into account. Temporal correlation of the inputs can be handled during the BDD traversal by using the probabilities of pairs of corresponding inputs, e.g., (a_0, a_t) , which are the transition probabilities [22]. A disadvantage of the method is that, for large circuits, the symbolic network may be very large and BDDs cannot be created, requiring the use of approximation schemes.

2.2.4. Switching Activity in Sequential Circuits. The methods described previously apply to combinational logic blocks. We now present some techniques that target issues particular to sequential circuits. Figure 8 represents a generic sequential circuit.

There are two main issues in the switching activity estimation of sequential circuits. One is correlation in time: some of the inputs to the combinational logic

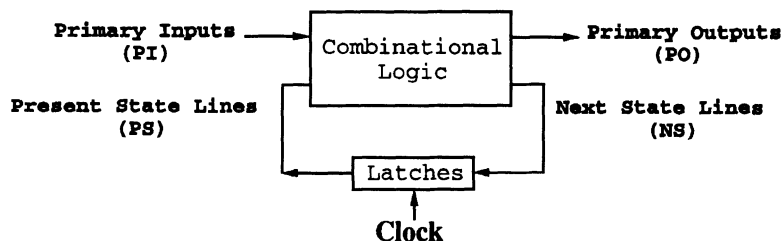


Figure 8. A synchronous sequential circuit.

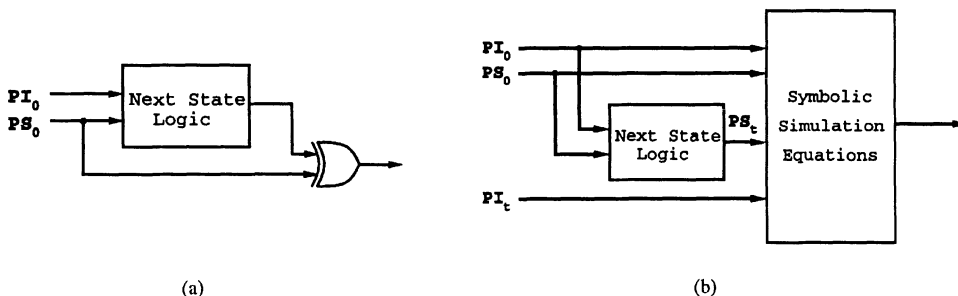


Figure 9. Generating temporal correlation of present state lines.

(the *present state lines*) are uniquely determined by the logic circuit and the previous input vector. The second issue is state line probabilities: the sequential circuit may have different probabilities of being in each state.

The first of these issues can be solved by using the *next state logic* block (the part of the combinational logic of Fig. 8 that computes the next state lines) as shown in Fig. 9. Transition probabilities can be calculated by computing the static probabilities of the outputs of the circuit shown in Fig. 9(a). Figure 9(b) shows how the next state logic can be used directly in the symbolic simulation method [21] to introduce the correct temporal correlation between the two input vectors. In any case, we are left with the problem of computing the probabilities of the state lines.

In general the state lines of a sequential circuit are correlated and, therefore, exact methods cannot use probabilities of individual state lines. Instead probabilities of each combination over all present state lines, i.e., *state probabilities*, have to be used. An exact method of computing state probabilities is to extract the *State Transition Graph* (STG) from the sequential circuit and solve a linear system of equations—the Chapman-Kolmogorov equations [10]. The problem is that the number of unknowns (state probabilities) is exponential in the number of state lines (n state lines $\Rightarrow 2^n$ states) thus cannot be applied to large circuits. However, in [23] the authors report solving the

Chapman-Kolmogorov system of equations for large Finite State Machines using Algebraic Decision Diagrams [24].

In [25] a method is proposed that computes individual state line probabilities directly. The authors show that over a large set of examples ignoring the correlation between state lines leads to an average error of 3%. The method involves the solution of a non-linear system of equations of size n . The next state lines are a Boolean function of the present state lines and the primary inputs and can be represented as:

$$\begin{aligned} ns_1 &= f_1(i_1, i_2, \dots, i_m, ps_1, ps_2, \dots, ps_n) \\ ns_2 &= f_2(i_1, i_2, \dots, i_m, ps_1, ps_2, \dots, ps_n) \\ &\vdots \\ ns_n &= f_n(i_1, i_2, \dots, i_m, ps_1, ps_2, \dots, ps_n) \end{aligned}$$

In terms of probabilities:

$$\begin{aligned} \text{prob}(ns_1) &= \text{prob}(f_1(i_1, \dots, i_m, ps_1, \dots, ps_n)) \\ \text{prob}(ns_2) &= \text{prob}(f_2(i_1, \dots, i_m, ps_1, \dots, ps_n)) \\ &\vdots \\ \text{prob}(ns_n) &= \text{prob}(f_n(i_1, \dots, i_m, ps_1, \dots, ps_n)) \end{aligned}$$

In steady state, $\text{prob}(ps_i) = \text{prob}(ns_i) = p_i$ and since we know the probability values for the primary inputs,

the system of equations above can be written as:

$$\begin{aligned}
 p_1 &= g_1(p_1, p_2, \dots, p_n) \\
 p_2 &= g_2(p_1, p_2, \dots, p_n) \\
 &\vdots \\
 p_n &= g_n(p_1, p_2, \dots, p_n)
 \end{aligned}$$

where the g_i 's are non-linear functions. To exemplify this, consider the Boolean function f_1 that generates ns_1 :

$$f_1 = (i_1 \wedge ps_1 \wedge \overline{ps_2}) \vee (i_1 \wedge \overline{ps_1} \wedge ps_2)$$

Assuming $\text{prob}(i_1) = 0.5$, the corresponding non-linear equation g_1 is:

$$g_1 = 0.5 \cdot (p_1 \cdot (1 - p_2) + (1 - p_1) \cdot p_2)$$

Iterative methods, such as Newton-Raphson or Picard-Peano, are used to solve the non-linear system of equations.

Recently a simulation-based technique to compute state line probabilities has been presented [26]. N logic simulations of the sequential circuit are done starting at some initial state S_0 and the value of each state line is checked at time k . N is determined from the confidence level α and allowed percentage error ϵ . k is the number of cycles the circuit has to go through in order to be considered in steady state. In steady state, the probabilities of the state lines are independent from the initial state, thus N parallel simulations are done starting from some other state S_1 . k is determined as the time at which the line probabilities obtained from starting at state S_0 and from S_1 are within ϵ .

2.2.5. Modeling Input Correlation. One basic assumption of all the previous techniques is the statistical independence of the primary inputs. This assumption can introduce some error in the switching activity estimation and is assumed simply because very often the correlation coefficients are not known and even if known, are difficult to specify as input to the estimation method.

Two recent works try to introduce some degree of information about correlation between inputs. The estimation method of [27] permits the user to specify pairwise correlation of inputs as *static* (SC) and *transition* (TC) *correlation coefficients*. These are defined as:

$$\begin{aligned}
 SC_{ij}^{xy} &= \frac{P_s(x = i \wedge y = j)}{P_s(x = i)P_s(y = j)} \\
 TC_{ij,kl}^{xy} &= \frac{P_t(x_{i \rightarrow k} \wedge y_{j \rightarrow l})}{P_t(x_{i \rightarrow k})P_t(y_{j \rightarrow l})}
 \end{aligned}$$

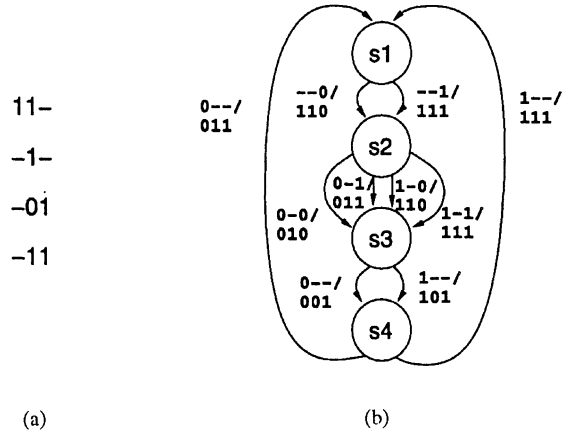


Figure 10. State transition graph to model an incompletely specified input sequence.

These coefficients are then propagated through the logic circuit and similar coefficients for internal signals are obtained.

A different technique is presented in [28]. Given a completely or incompletely specified input sequence, a Finite State Machine (named *Input Modeling Finite State Machine*—IMFSM) is built generating this sequence of inputs and feeding it to the original sequential logic circuit. Figure 10(a) shows an incompletely specified input sequence and the State Transition Graph (STG) generated for this sequence is represented in Fig. 10(b). A logic implementation of this STG is obtained after encoding the states (the result is independent of the particular encoding as the switching in the IMFSM will be ignored) and the outputs of the IMFSM are connected to the inputs of the original logic circuit. This is illustrated in Fig. 11 where M is the original circuit. The input to the IMFSM are probabilities of the -'s (unknowns) in the incompletely specified sequence being 1 or 0. Any of the techniques for estimating switching activity in sequential circuits can be used on the entire circuit of Fig. 11 to obtain the power dissipation of M over a particular set of input sequences. A limitation of this method is that long input sequences lead to large IMFSMs, thus increasing the complexity of the circuit the sequential power estimation method as to handle.

2.3. Summary

There are two main approaches for switching activity estimation at the logic level: simulation-based and probabilistic techniques. In both the tradeoff is

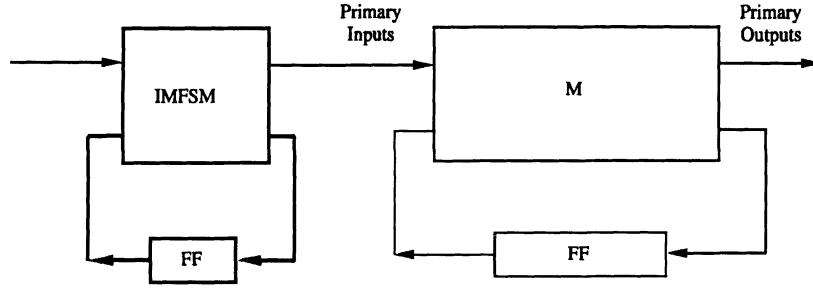


Figure 11. Input modeling finite state machine feeding the original circuit M .

accuracy vs. run-time. In simulation-based methods, the higher the accuracy requested by the user (translated in terms of lower allowed error ϵ and/or higher confidence level α) the more input vectors that have to be simulated. In probabilistic methods, we have methods like the transition density propagation method [5] that are very fast but ignore some important issues like spatial correlation, to methods like symbolic simulation [21] that model correlation and glitching correctly but are much slower and limited in the size of circuits that can be handled.

3. Power Optimization by Transistor Sizing

We describe an important optimization method for low power: transistor sizing. While strictly this is not a gate level optimization technique, its importance has led to the incorporation of transistor sizing into logic synthesis systems.

Power dissipation is directly related to the capacitance being switched, cf. Eq. (2). Low power designs should, therefore, use minimum sized transistors. However, there is a performance penalty in using minimum sized devices. The problem of *transistor sizing* is computing the sizes of the transistors in the circuit that minimizes power dissipation while meeting the delay constraints specified for the design.

Transistor sizing for minimum area is a well established problem [29]. There is a subtle difference between this problem and sizing for low power. If the critical delay of the circuit exceeds the design specification and thus some transistors need to be resized, methods for minimum area will focus on minimizing the total enlargement of the transistors. On the other hand, methods for low power will first resize those transistors driven by signals with lower switching activity.

A technique for transistor resizing targeting minimum power is described in [30]. Initially minimum sized devices are used. Each path whose delay exceeds the maximum allowed is examined separately. Transistors in the logic gates of these paths are resized such that the delay constraint is met. Signal transition probabilities are used to measure the power penalty of each resizing. The option with least power penalty is selected. A similar method is presented in [31]. This method is able to take false paths into account when computing the critical path of the circuit.

In [32] the authors note that the short-circuit currents are proportional to the transistor sizing. Thus the cost function used in [32] also minimizes short-circuit power.

These methods work on local optimizations. A global solution for the transistor sizing problem for low power is proposed in [33]. The problem is modeled as:

$$\tau_g = \tau_{\text{intr}} + k \frac{C_{\text{wire}} + \sum_{i \in \text{fanout}(g)} S_i C_{\text{in},i}}{S_g} \quad (4)$$

$$T_g = \tau_g + \max_{i \in \text{inputs}(g)} T_i \quad (5)$$

$$P_g = N_g \left(C_{\text{wire}} + \sum_{i \in \text{fanout}(g)} S_i C_{\text{in},i} \right) \quad (6)$$

where S_g , N_g , P_g , and τ_g are respectively the sizing factor, switching activity, power dissipation and delay of gate g . τ_{intr} and k are constants representing respectively the intrinsic delay of the gate and ratio between delay and the capacitive load the gate is driving. T_g is the worst case propagation delay from an input to the output of g . C denotes load capacitances.

The solution to the optimization problem is achieved using Linear Programming (LP). A piecewise linear approximation is obtained for Eq. (4). The constraints

for the LP problem are:

$$\begin{aligned} \tau_g &\geq k_{1,1} - k_{1,2} S_g + k_{1,3} \sum_i S_i C_{in,i} \\ &\vdots \\ \tau_g &\geq k_{n,1} - k_{n,2} S_g + k_{n,3} \sum_i S_i C_{in,i} \end{aligned} \quad (\text{from Eq. (4)})$$

$$S_{min} \leq S_g \leq S_{max}$$

$$\begin{aligned} T_g &\geq T_j + \tau_g \quad \forall j \in \text{fanin}(g) \\ T_{max} &\geq T_g \end{aligned} \quad (\text{from Eq. (5)})$$

and the objective function is:

$$P = \sum_{\text{over all gates } i} P_i$$

where $k_{i,j}$ are constants computed such that we get a best fit for the linearized model.

As devices shrink in size, the delay and power associated with interconnect grow in relative importance. In [34] the authors propose that wiresizing should be considered together with transistor sizing. Wider lines present less resistance but have higher capacitance. A better global solution in terms of power can be achieved if both transistor and wire sizes are considered simultaneously.

4. Combinational Logic Level Optimization

In this section we review techniques that work on restructuring combinational logic circuits to obtain a less power consuming circuit. The power dissipation model used is the one presented in Eq. (2) and V_{DD} and f are assumed fixed. The cost function to minimize is $\sum_i C_i \times N_i$, which is often called *switched capacitance*.

The techniques we present in this section focus on reducing the switched capacitance within traditional design styles. A new design style targeting specifically low power dissipation is proposed in [35]. It is based on *Shannon circuits* where for each computation a single

input-output path is active, thus minimizing switching activity. Techniques are presented on how to keep the circuit from getting too large as this would increase the total switched capacitance.

4.1. Path Balancing

Spurious transitions account for a significant fraction of the switching activity power in typical combinational logic circuits [14, 15]. In order to reduce spurious switching activity, the delay of paths that converge at each gate in the circuit should be roughly equal. Solutions to this problem, known as *path balancing*, have been proposed in the context of *wave-pipelining* [36]. One technique involves restructuring the logic circuit, as illustrated in Fig. 12. Additionally, by selectively inserting unit-delay buffers to the inputs of gates in a circuit, the delays of all paths in the circuit can be made equal (Fig. 13). This addition will not increase the critical delay of the circuit, and will effectively eliminate spurious transitions. However, the addition of buffers increases capacitance which may offset the reduction in switching activity.

4.2. Don't-Care Optimization

Multilevel circuits are optimized by repeated two-level minimization with appropriate don't-care sets. Consider the circuit of Fig. 14. The structure of the logic circuit may imply some combinations over nodes A, B and C never occur. These combinations form the *Controllability or Satisfiability Don't-Care Set* (SDC) of F. Similarly, there may be some input combinations for which the value of F is not used in the computation of the outputs of the circuit. The set of these combinations is called the *Observability Don't-Care Set* (ODC).

Traditionally don't-care sets have been used for area minimization [37]. Recently techniques have been proposed (e.g., [14, 38]) for the use of don't-cares to reduce the switching activity at the output of a logic gate. The

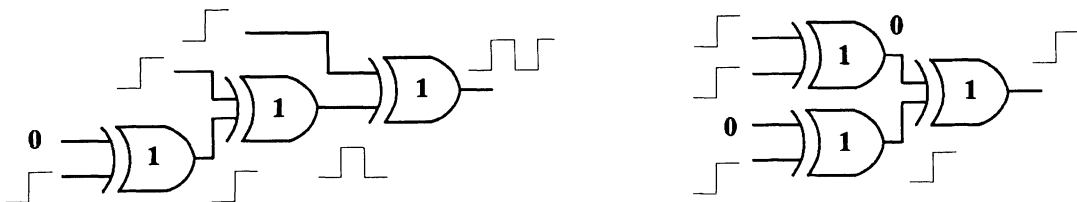


Figure 12. Logic restructuring to minimize spurious transitions.



Figure 13. Buffer insertion for path balancing.

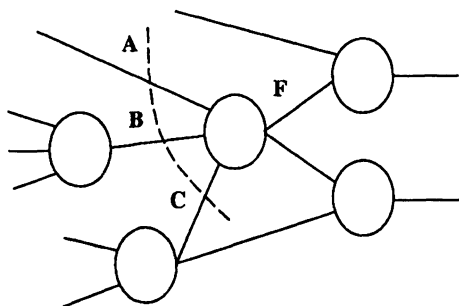


Figure 14. SDCs and ODCs in a multilevel circuit.

transition probability of a static CMOS gate is given by $P_t(f) = 2P_s(f)(1 - P_s(f))$ (ignoring temporal correlation). The maximum for this function occurs when $P_s(f) = 0.5$. The authors of [14] suggest including minterms in the don't-care set in the ON-set of the function if $P_s(f) > 0.5$ or in the OFF-set if $P_s(f) < 0.5$. In [38] this method is extended to take into account the effect the optimization of a gate has in the switching probability of its transitive fanout.

4.3. Logic Factorization

A primary means of technology-independent optimization is the factoring of logical expressions. For example, the expression $a \cdot c + a \cdot d + b \cdot c + b \cdot d$ can be factored into $(a + b) \cdot (c + d)$ reducing transistor count considerably. Common subexpressions can be found across multiple functions and reused. Kernel extraction is a commonly used algorithm to perform multilevel logic optimization for area [39]. In this algorithm, the kernels of the given expressions are generated and kernels that maximally reduce literal count are selected.

When targeting power dissipation, the cost function is not literal count but switching activity. Even though transistor count may be reduced by factorization, the total switched capacitance may increase. Consider the example shown in Fig. 15 and assume that a has a low transition probability $p_a = 0.1$ and b and c have each

$p_b = p_c = 0.5$. The total switched capacitance in circuit (a) is $(2p_a + p_b + p_c + p_1 + p_2 + p_3)C = 1.378C$ and in (b) is $(p_a + p_b + p_c + p_4 + p_5)C = 1.551C$. Clearly factorization is not always desirable in terms of power. Further, kernels that lead to minimum literal count do not necessarily minimize the switched capacitance.

Modified kernel extraction methods that target power are described in [40–43]. The algorithms proposed compute the switching activity associated with the selection of each kernel. Kernel selection is based on the reduction of both area and switching activity.

4.4. Technology Mapping

Technology mapping is the process by which a logic circuit is implemented in terms of the logic elements available in a particular technology library. Associated with each logic element is an area and delay cost. The traditional optimization problem is to find the implementation that meets some delay constraint and minimizes the total area cost. Techniques to efficiently find an optimal solution to this problem have been proposed [44].

As long as the delay constraints are still met, the designer is usually willing to make some tradeoff between area and power dissipation. Consider the circuit of Fig. 16(a). Mapping this circuit for minimum area using the technology library presented in Fig. 16(b) yields the circuit presented in Fig. 17(a). The designer may prefer to give up some area in order to obtain the more power efficient design of Fig. 17(b).

The graph covering formulation of [44] has been extended to use switched capacitance as part of the cost function. The main strategy to minimize power dissipation is to hide nodes with high switching activity within complex logic elements as capacitances internal to gates are generally much smaller. Although using different models for delay and switching activity estimation, techniques such as those described in [45–47] use this approach to minimize power dissipation during technology mapping.

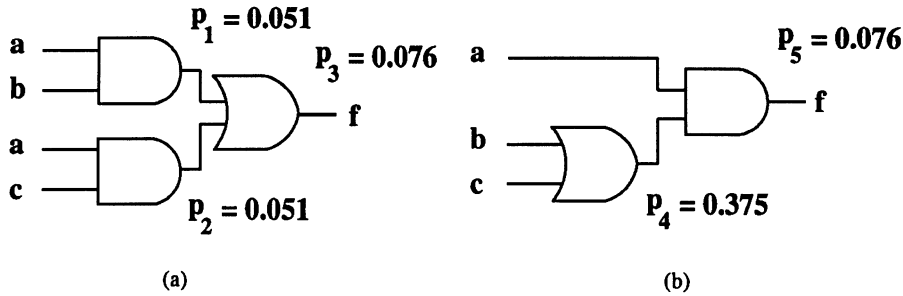


Figure 15. Logic factorization for low power.

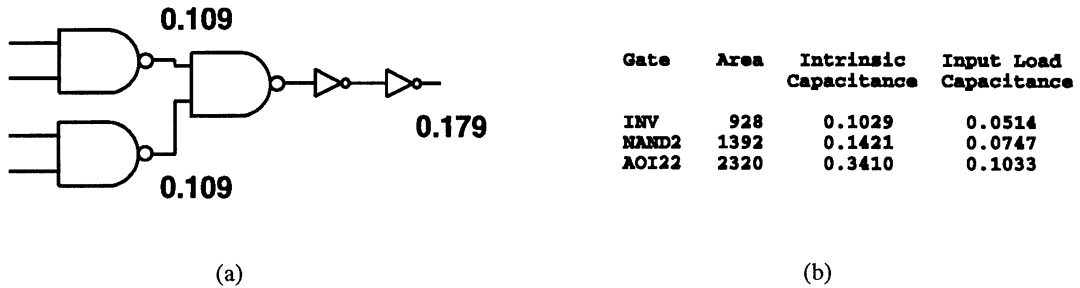


Figure 16. (a) Circuit to be mapped. (b) Information about technology library.

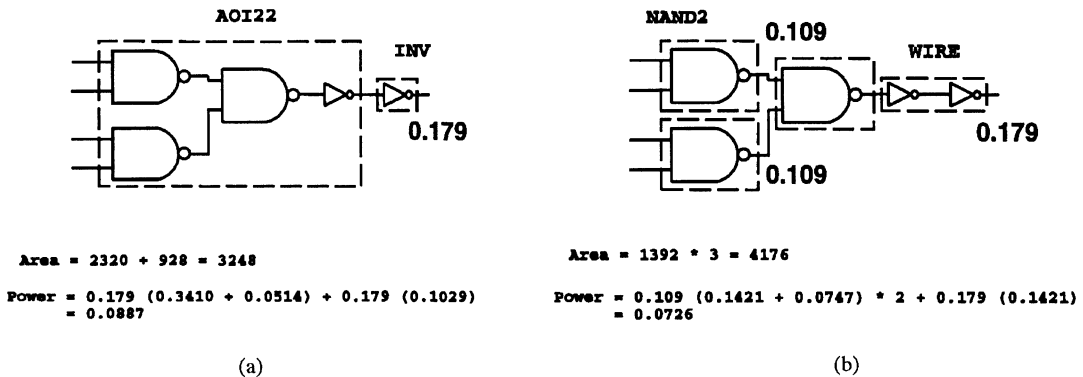


Figure 17. (a) Mapping for minimum area. (b) Mapping for minimum power.

Most technology libraries include the same logic element with different sizes (i.e., driving capability). Thus, in technology mapping for low power, the choice of the size of each logic element such that the delay constraints are met with minimum power consumption is made. This problem is the discrete counterpart of the transistor sizing problem of Section 3 and is addressed in [30, 48, 49].

5. Sequential Logic Level Optimization

We now focus on techniques for low power that are specific to synchronous sequential logic circuits. A

characteristic of this type of circuits is that switching activity is easily controllable by deciding whether or not to load new values to registers. Further, at the output of registers we always have a clean transition, free from glitches.

5.1. State Encoding

State encoding is the process by which a unique binary code is assigned to each state in a Finite State Machine (FSM). Although this assignment does not influence the functionality of the FSM, it determines the complexity of the combinational logic block in the FSM implementation (cf. Fig. 8).

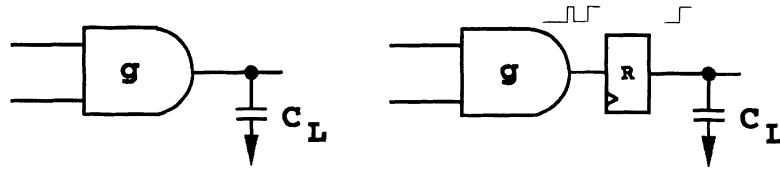


Figure 18. Filtering of glitching by adding a register.

State encoding for minimum area is a well-researched problem [50]. The optimum solution to this problem has been proven to be NP-hard. Heuristics that work well assign codes with minimum Hamming distances to states that have edges connecting them in the State Transition Graph (STG). This potentially enables the existence of larger kernels or kernels that can be used a larger number of times.

Targeting low power, the heuristics go one step further: assign minimum Hamming distance codes to states that are connected by edges that have larger probability of being traversed. The probability that a given edge in the STG is traversed is given by the steady-state probability of the STG being in the start state of the edge times the static probability of the input combination associated with that edge. Whenever this edge is exercised, only a small number of state lines (ideally one) will change, leading to reduced overall switching activity in the combinational logic block. This is the cost function used in the techniques proposed in [40, 51, 52].

In [53], the technique takes into account not only the power in the state lines but also in the combinational logic by using in the cost function the savings relative cubes possible to obtain for a given state encoding.

5.2. Encoding in the Datapath

Encoding to reduce switching activity in datapath logic has also been the subject of attention. A method to minimize the switching on buses is proposed in [54]. In this technique, an extra line E is added to the bus which indicates if the value being transferred is the true value or needs to be bitwise complemented upon receipt. Depending on the value transferred in the previous cycle, a decision is made to either transfer the true current value or the complemented current value, so as to minimize the number of transitions on the bus lines. For example, if the previous value transferred was 0000, and the current value is 1011, then the value 0100 is transferred, and the line E is asserted to signify that the value 0100 has to be complemented at the other end. Other methods of bus coding are also proposed in [54].

Methods to implement arithmetic units other than in standard two's complement arithmetic are also being investigated. A method of one-hot residue coding to minimize switching activity of arithmetic logic is presented in [55].

5.3. Retiming for Low Power

Retiming was first proposed in [56] as a technique to improve throughput by moving the registers in a circuit while maintaining input-output functionality. In [57] retiming is used to allow optimization methods for combinational circuits to be applied across register boundaries. The circuit is retimed so that registers are moved to the border of the circuit, logic minimization methods are applied to the whole combinational logic block and lastly the registers are again redistributed in the circuit to maximize throughput.

The use of retiming to minimize switching activity has been proposed in [58], based on the observation that the output of registers have significantly fewer transitions than the register inputs. In particular, no glitching is present. Consider Fig. 18. Since the spurious transitions are filtered by the register, $N_R \leq N_g$. For a large load capacitance C_L adding the register may actually reduce the total switched capacitance: $N_g C_R + N_R C_L < N_g C_L$. Further, moving registers across nodes by retiming may change the switching activity at several nodes in the circuit. In the top circuit of Fig. 19 the switched capacitance is $N_0 C_R + N_1 C_{L1} + N_2 C_{L2}$ and the switched capacitance in its retimed version, shown at the bottom of the same figure, is $N_0 C_{L1} + N_1 C_R + N_2 C_{L2}$. One of these two circuits may have significantly less switched capacitance. The technique of [58] uses heuristics to place registers such that nodes driving large capacitances have reduced switching activity.

5.4. Gated Clocks

Large VLSI circuits such as processors contain register files, arithmetic units and control logic. The register

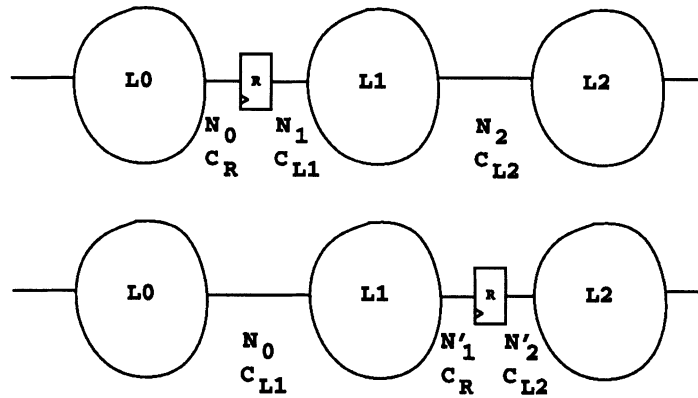


Figure 19. Retiming for low power.

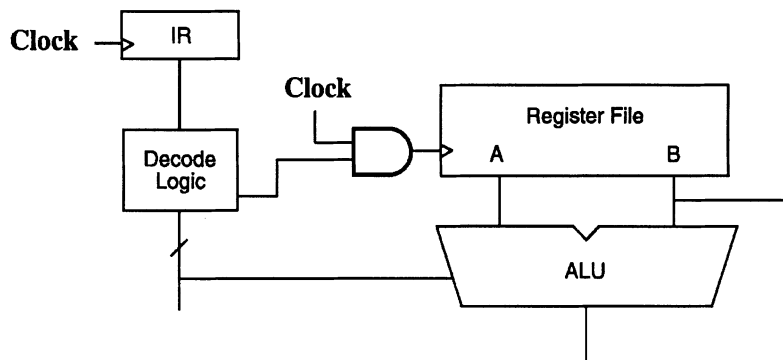


Figure 20. Reducing switching activity in the register file and ALU by gating the clock.

file is typically not accessed in each clock cycle. Similarly, in an arbitrary sequential circuit, the values of particular registers need not be updated in every clock cycle. If simple conditions that determine the inaction of particular registers can be determined, then power reduction can be obtained by gating the clocks of these registers [59] as illustrated in Fig. 20. When these conditions are satisfied, the switching activity within the registers is reduced to negligible levels.

The same method can be applied to “turn off” or “power down” arithmetic units when these units are not in use in a particular clock cycle. For example, when a branch instruction is being executed by a CPU, a multiply unit may not be used. The input registers to the multiplier are maintained at their previous values, ensuring that switching activity power in the multiplier is zero for this clock cycle.

In [60] a gated clock scheme applicable to FSMs is proposed. The clock to the FSM is turned off when the FSM is in a state with a self loop waiting for some external condition to arrive.

5.5. Precomputation

A technique called precomputation, originally presented in [61], achieves data-dependent power down at the sequential logic or combinational logic level. In the sequential precomputation architecture, the output logic values of a circuit are selectively precomputed one clock cycle before they are required, and these pre-computed values are used to reduce internal switching activity in the succeeding clock cycle. The architecture proposed in [61] is shown in Fig. 21. Functions g_1 and g_2 are a function of a subset of the inputs to block A. g_1 or g_2 evaluate to 1 when their inputs are enough to determine the output of A: $g_1 = 1 \Rightarrow f = 1$; $g_2 = 1 \Rightarrow f = 0$. In this situation, all other inputs to A are disabled and we will have reduced switching activity in A in the next clock cycle. The objective is to obtain simple functions g_1 and g_2 , since this is extra logic, but at the same time maximize the number of input combinations for which the other inputs are disabled. In [61] it is proved that obtaining g_1 and g_2 from

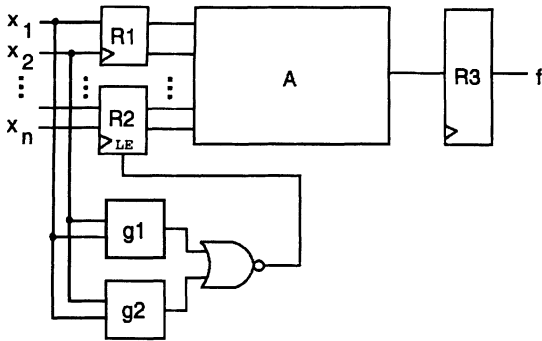


Figure 21. Subset input disabling precomputation architecture.

the universal quantification of f over the inputs not in g_1 and g_2 maximizes the number of input combinations.

For example, in using precomputation for a n -bit comparator we make $g_1 = C \langle n-1 \rangle \wedge D \langle n-1 \rangle$ and $g_2 = \overline{C} \langle n-1 \rangle \wedge \overline{D} \langle n-1 \rangle$. That is, if the most significant bit of the two numbers are different then we can disable all other inputs since we already know the value of f .

g_1 and g_2 are kept simple by making them a function of a small subset of the inputs to A . This can be a limitation of this technique. To overcome this, a new architecture is proposed in [62] where g_1 and g_2 can be a function of any number of inputs. When one of them evaluates to 1, all inputs to A are disabled, implying no switching activity in A in the next clock cycle, and the output f is set directly. In this architecture the input combinations that are included in g_1 and g_2 has to be monitored carefully to prevent these functions from becoming too complex.

A precomputation architecture for combinational circuits is also presented in [62]. Again g_1 and g_2 functions are generated and transitions are prevented from propagating by using latches or pass-transistors. The

main advantage of this combinational architecture is that precomputation can be done at any point in the circuit, as illustrated in Fig. 22.

In the same lines, a technique called *guarded evaluation* is presented in [63]. Instead of adding extra logic to generate the disabling signal, this technique uses signals already existing in the circuit to prevent transitions from propagating. Disabling signals and subcircuits to be disabled are determined by using observability don't-care sets.

6. Summary

We have reviewed techniques for power estimation of combinational and sequential logic circuits. A spectrum of techniques exist which make different assumptions regarding logic behavior and signal correlations. Frameworks for the estimation of power in sequential circuits which model internal and input correlations have been developed.

We have also reviewed recently proposed optimization methods for low power that work at the transistor and logic levels. Shut down techniques such as those presented in Sections 5.4 and 5.5 have a greater potential for reducing the overall switching activity in logic circuits. Other techniques that focus on reducing spurious transitions, such as those described in Sections 4.1 and 5.3, are inherently limited as they do not address the zero-delay switching activity. However these techniques are independent improvements and can be used together with the other optimization techniques.

Techniques that work at higher (system and architecture) and lower (layout) levels exist. The techniques we presented in this paper can be used on a system/architecture optimized circuit and layout optimization can be done after logic optimization has been

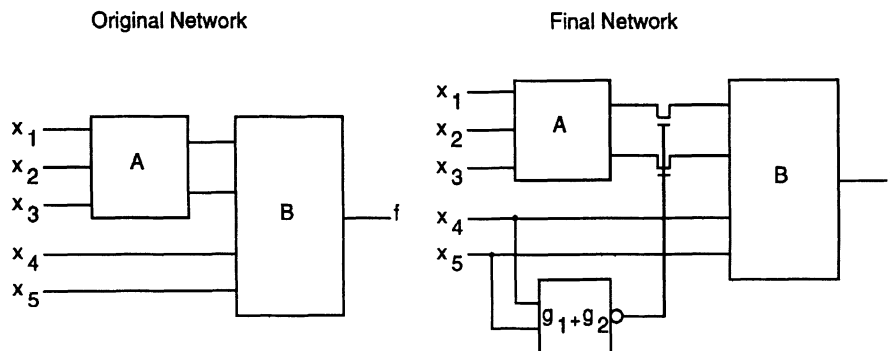


Figure 22. Precomputation in a combinational circuit.

performed, thus obtaining a design that is made more power efficient at all levels of abstraction.

Acknowledgments

This research was supported in part by the Advanced Research Projects Agency under contract DABT63-94-C-0053, and in part by a NSF Young Investigator Award with matching funds from Mitsubishi Corporation.

References

1. T. Quarles, "The SPICE3 implementation guide," Technical Report ERL M89/44, Electronics Research Laboratory Report, University of California at Berkeley, Berkeley, California, April 1989.
2. R. Tjarnstrom, "Power dissipation estimate by switch level simulation," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 881–884, May 1989.
3. A. Salz and M. Horowitz, "IRSIM: An incremental MOS switch-level simulator," in *Proceedings of the 26th Design Automation Conference*, pp. 173–178, June 1989.
4. L. Glasser and D. Dobberpuhl, *The Design and Analysis of VLSI Circuits*, Addison-Wesley, 1985.
5. F. Najm, "Transition density: A new measure of activity in digital circuits," *IEEE Transactions on Computer-Aided Design*, Vol. 12, No. 2, pp. 310–323, Feb. 1993.
6. A. Chandrakasan, T. Sheng, and R.W. Brodersen, "Low power CMOS digital design," *Journal of Solid State Circuits*, Vol. 27, No. 4, pp. 473–484, April 1992.
7. S. Devadas, K. Keutzer, and J. White, "Estimation of power dissipation in CMOS combinational circuits using Boolean function manipulation," in *IEEE Transactions on Computer-Aided Design*, pp. 373–383, March 1992.
8. S. Manne, A. Pardo, R. Bahar, G. Hachtel, F. Somenzi, E. Macii, and M. Poncino, "Computing the maximum power cycles of a sequential circuit," in *Proceedings of the Design Automation Conference*, pp. 23–28, June 1995.
9. R. Burch, F. Najm, P. Yang, and T. Trick, "A Monte Carlo approach to power estimation," *IEEE Transactions on VLSI Systems*, Vol. 1, No. 1, pp. 63–71, March 1993.
10. A. Papoulis, *Probability, Random Variables and Stochastic Processes*, McGraw-Hill, 3rd edition, 1991.
11. M. Xakellis and F. Najm, "Statistical estimation of the switching activity in digital circuits," in *Proceedings of the Design Automation Conference*, pp. 728–733, June 1994.
12. A. Hill and S. Kang, "Determining accuracy bounds for simulation-based switching activity estimation," in *International Symposium on Low Power Design*, pp. 215–220, April 1995.
13. R. Bryant, "Graph-based algorithms for Boolean function manipulation," *IEEE Transactions on Computers*, Vol. C-35, No. 8, pp. 677–691, Aug. 1986.
14. A. Shen, S. Devadas, A. Ghosh, and K. Keutzer, "On average power dissipation and random pattern testability of combinational logic circuits," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 402–407, Nov. 1992.
15. M. Favalli and L. Benini, "Analysis of glitch power dissipation in CMOS ICs," in *International Symposium on Low Power Design*, pp. 123–128, April 1995.
16. M.A. Cirit, "Estimating dynamic power consumption of CMOS circuits," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 534–537, Nov. 1987.
17. B. Kapoor, "Improving the accuracy of circuit activity measurement," in *Proceedings of the 1994 International Workshop on Low Power Design*, pp. 111–116, April 1994.
18. F.N. Najm, R. Burch, P. Yang, and I. Hajj, "Probabilistic simulation for reliability analysis of CMOS VLSI circuits," *IEEE Transactions on Computer-Aided Design*, Vol. 9, No. 4, pp. 439–450, April 1990.
19. C.Y. Tsui, M. Pedram, and A. Despain, "Efficient estimation of dynamic power dissipation under a real delay model," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 224–228, Nov. 1993.
20. T. Uchino, F. Minami, T. Mitsuhashi, and N. Goto, "Switching activity analysis using boolean approximation method," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 20–25, Nov. 1995.
21. A. Ghosh, S. Devadas, K. Keutzer, and J. White, "Estimation of average switching activity in combinational and sequential circuits," in *Proceedings of the Design Automation Conference*, pp. 253–259, June 1992.
22. P. Schneider and U. Schlichtmann, "Decomposition of boolean functions for low power based on a new power estimation technique," in *Proceedings of the 1994 International Workshop on Low Power Design*, pp. 123–128, April 1994.
23. G. Hachtel, E. Macii, A. Pardo, and F. Somenzi, "Probabilistic analysis of large finite state machines," in *Proceedings of the Design Automation Conference*, pp. 270–275, June 1994.
24. R. Bahar, E. Frohm, C. Gaona, G. Hachtel, E. Macii, A. Pardo, and F. Somenzi, "Algebraic decision diagrams and their applications," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 188–191, Nov. 1993.
25. C.-Y. Tsui, J. Monteiro, M. Pedram, S. Devadas, A. Despain, and B. Lin, "Power estimation for sequential logic circuits," *IEEE Transactions on VLSI Systems*, Vol. 3, No. 3, pp. 404–416, Sept. 1995.
26. F. Najm, S. Goel, and I. Hajj, "Power estimation in sequential circuits," in *Proceedings of the Design Automation Conference*, pp. 635–640, June 1995.
27. R. Marculescu, D. Marculescu, and M. Pedram, "Efficient power estimation for highly correlated input streams," in *Proceedings of the Design Automation Conference*, pp. 628–634, June 1995.
28. J. Monteiro and S. Devadas, "Techniques for the power estimation of sequential logic circuits under user-specified input sequences and programs," in *Proceedings of the International Symposium on Low Power Design*, pp. 33–38, April 1995.
29. S. Sapatnekar, V. Rao, P. Vaidya, and S. Kang, "An exact solution to the transistor sizing problem for CMOS circuits using convex optimization," *IEEE Transactions on Computer-Aided Design*, Vol. 12, No. 11, pp. 1621–1634, Nov. 1993.
30. C.H. Tan and J. Allen, "Minimization of power in VLSI circuits using transistor sizing, input ordering, and statistical power estimation," in *Proceedings of the International Workshop on Low Power Design*, pp. 75–80, April 1994.
31. R. Bahar, G. Hachtel, E. Macii, and F. Somenzi, "A symbolic method to reduce power consumption of circuits

- containing false paths,” in *Proceedings of the International Conference on Computer-Aided Design*, pp. 368–371, Nov. 1994.
32. M. Borah, R. Owens, and M. Irwin, “Transistor sizing for minimizing power consumption of CMOS circuits under delay constraint,” in *International Symposium on Low Power Design*, pp. 167–172, April 1995.
 33. M. Berkelaar and J. Jess, “Computing the entire active area/power consumption versus delay trade-off curve for gate sizing with a piecewise linear simulator,” in *Proceedings of the International Conference on Computer-Aided Design*, pp. 474–480, Nov. 1994.
 34. J. Cong and C. Koh, “Simultaneous driver and wire sizing for performance and power optimization,” *IEEE Transactions on VLSI Systems*, Vol. 2, No. 4, pp. 408–425, Dec. 1994.
 35. L. Lavagno, P. McGeer, A. Saldanha, and A. Sangiovanni-Vincentelli, “Timed shannon circuits: A power-efficient design style and synthesis tool,” in *Proceedings of the Design Automation Conference*, pp. 254–260, June 1995.
 36. T. Kim, W. Burleson, and M. Ciesielski, “Logic restructuring for wave-pipelined circuits,” in *Proceedings of the International Workshop on Logic Synthesis*, 1993.
 37. K. Bartlett, R.K. Brayton, G.D. Hachtel, R.M. Jacoby, C.R. Morrison, R.L. Rudell, A. Sangiovanni-Vincentelli, and A.R. Wang, “Multi-level logic minimization using implicit don’t cares,” in *IEEE Transactions on Computer-Aided Design*, pp. 723–740, June 1988.
 38. S. Iman and M. Pedram, “Multi-level network optimization for low power,” in *Proceedings of the International Conference on Computer-Aided Design*, pp. 371–377, Nov. 1994.
 39. R. Brayton, R. Rudell, A. Sangiovanni-Vincentelli, and A. Wang, “MIS: A multiple-level logic optimization system,” in *IEEE Transactions on Computer-Aided Design*, pp. 1062–1081, Nov. 1987.
 40. K. Roy and S. Prasad, “Circuit activity based logic synthesis for low power reliable operations,” *IEEE Transactions on VLSI Systems*, Vol. 1, No. 4, pp. 503–513, Dec. 1993.
 41. R. Murgai, R. Brayton, and A. Sangiovanni-Vincentelli, “Decomposition of logic functions for minimum transition activity,” in *Proceedings of the 1994 International Workshop on Low Power Design*, pp. 33–38, April 1994.
 42. S. Iman and M. Pedram, “Logic extraction and factorization for low power,” in *Proceedings of the Design Automation Conference*, pp. 248–253, June 1995.
 43. R. Panda and F. Najm, “Technology decomposition for low-power synthesis,” in *Proceedings of the Custom Integrated Circuit Conference*, 1995.
 44. K. Keutzer, “DAGON: Technology mapping and local optimization,” in *Proceedings of the 24th Design Automation Conference*, pp. 341–347, June 1987.
 45. V. Tiwari, P. Ashar, and S. Malik, “Technology mapping for low power,” in *Proceedings of the 30th Design Automation Conference*, pp. 74–79, June 1993.
 46. C.-Y. Tsui, M. Pedram, and A.M. Despaigne, “Technology decomposition and mapping targeting low power dissipation,” in *Proceedings of the 30th Design Automation Conference*, pp. 68–73, June 1993.
 47. B. Lin, “Technology mapping for low power dissipation,” in *Proceedings of the International Conference on Computer Design: VLSI in Computers and Processors*, Oct. 1993.
 48. R. Bahar, H. Cho, G. Hachtel, E. Macii, and F. Somenzi, “An application of ADD-based timing analysis to combinational low power synthesis,” in *Proceedings of the 1994 International Workshop on Low Power Design*, pp. 39–44, April 1994.
 49. Y. Tamiya, Y. Matsunaga, and M. Fujita, “LP-based cell selection with constraints of timing, area and power consumption,” in *Proceedings of the International Conference on Computer-Aided Design*, pp. 378–381, Nov. 1994.
 50. P. Ashar, S. Devadas, and A.R. Newton, *Sequential Logic Synthesis*, Kluwer Academic Publishers, Boston, Massachusetts, 1991.
 51. E. Olson and S. Kang, “Low-power state assignment for finite state machines,” in *Proceedings of the 1994 International Workshop on Low Power Design*, pp. 63–68, April 1994.
 52. G. Hachtel, M. Hermida, A. Pardo, M. Poncino, and F. Somenzi, “Re-encoding sequential circuits to reduce power dissipation,” in *Proceedings of the International Conference on Computer-Aided Design*, pp. 70–73, Nov. 1994.
 53. C.-Y. Tsui, M. Pedram, C.-A. Chen, and A.M. Despaigne, “Low power state assignment targeting two- and multi-level logic implementations,” in *Proceedings of the International Conference on Computer-Aided Design*, pp. 82–87, Nov. 1994.
 54. M. Stan and W. Burleson, “Limited-weight codes for low-power I/O,” in *Proceedings of the International Workshop on Low Power Design*, pp. 209–214, April 1994.
 55. W.A. Chren, “Low delay-power product CMOS design using one-hot residue coding,” in *Proceedings of the International Symposium on Low Power Design*, April 1995.
 56. C.E. Leiserson, F.M. Rose, and J.B. Saxe, “Optimizing synchronous circuitry by retiming,” in *Proceedings of 3rd CalTech Conference on VLSI*, pp. 23–36, March 1983.
 57. S. Malik, E. Sentovich, R. Brayton, and A. Sangiovanni-Vincentelli, “Retiming and resynthesis: Optimizing sequential circuits using combinational techniques,” in *IEEE Transactions on Computer-Aided Design*, pp. 74–84, Jan. 1991.
 58. J. Monteiro, S. Devadas, and A. Ghosh, “Retiming sequential circuits for low power,” in *Proceedings of the International Conference on Computer-Aided Design*, pp. 398–402, Nov. 1993.
 59. A. Chandrakasan, *Low-Power Digital CMOS Design*, Ph.D. Thesis, University of California at Berkeley, UCB/ERL Memorandum No. M94/65, Aug. 1994.
 60. L. Benini and G. De Micheli, “Transformation and synthesis of FSMs for low power gated clock implementation,” in *Proceedings of the International Symposium on Low Power Design*, pp. 21–26, April 1995.
 61. M. Alidina, J. Monteiro, S. Devadas, A. Ghosh, and M. Papaefthymiou, “Precomputation-based sequential logic optimization for low power,” *IEEE Transactions on VLSI Systems*, Vol. 2, No. 4, pp. 426–436, Dec. 1994.
 62. J. Monteiro, J. Rinderknecht, S. Devadas, and A. Ghosh, “Optimization of combinational and sequential logic circuits for low power using precomputation,” in *Proceedings of the 1995 Chapel Hill Conference on Advanced Research on VLSI*, pp. 430–444, March 1995.
 63. V. Tiwari, P. Ashar, and S. Malik, “Guarded evaluation: Pushing power management to logic synthesis/design,” in *International Symposium on Low Power Design*, pp. 221–226, April 1995.



José Monteiro received the Engineer's and Master's degrees in Electrical and Computer engineering in 1989 and 1992 respectively, from Instituto Superior Técnico at the Technical University of Lisbon. He has been at the Massachusetts Institute of Technology since 1993 working on his Ph.D. which he is about to conclude. The Ph.D. thesis is entitled *A Computer-Aided Design Methodology for Low Power Sequential Logic Circuits*. His research interests are in the area of synthesis of VLSI circuits, particularly on optimization methods for low power consumption. He received the 1996 IEEE Transactions on VLSI Systems Best Paper award.



Srinivas Devadas received a B.Tech. in Electrical Engineering from the Indian Institute of Technology, Madras in 1985 and a M.S. and

Ph.D. in Electrical Engineering from the University of California, Berkeley, in 1986 and 1988 respectively. Since August 1988, he has been at the Massachusetts Institute of Technology, Cambridge, and is currently an Associate Professor of Electrical Engineering and Computer Science. He held the Analog Devices Carrier Development Chair of Electrical Engineering from 1989 to 1991. His research interests span all aspects of synthesis of VLSI circuits, with emphasis on optimization techniques for synthesis at the logic, layout and architectural levels, design for low power, testing of VLSI circuits, formal verification, hardware/software co-design, design-for-testability methods and interactions between synthesis and testability of VLSI systems. He has received six Best Paper awards at CAD conferences and journals, including the 1990 IEEE Transactions on CAD and the 1996 IEEE Transactions on VLSI Systems Best Paper awards. In 1992, he received a NSF Young Investigator Award. He has served on the technical program committees of several conferences and workshops including the Int'l Conference on Computer Design, and the Int'l Conference on Computer-Aided Design. Dr. Devadas is a member of IEEE and ACM.