Part III Systems and Signals

Section 1 Computer-Aided Design

Chapter 1. Custom-Integrated Circuits

Academic and Research Staff

Professor Jonathan Allen, Professor Anantha P. Chandrakasan, Professor Srinivas Devadas, Professor John L. Wyatt, Jr., Professor Berthold K.P. Horn, Professor Hae-Seung Lee, Professor Charles G. Sodini, Dr. Ichiro Masaki, Dr. Joseph F. Rizzo III,[1] Dr. Christopher J. Terman

Graduate Students

Haigian Cheng, Daniel W. Engels, Farzan Fallah, Yajun Fang, Andrew E. Grumet, George I. Hadjiyiannis, Silvina Z. Hanono, Bradley J. Lichtenstein, Nicole S. Love, Shawn G. Kelly, Joshua D. Moss, Pablo Acosta Serafini, Mark G. Spaeth, Zubair Talib, Ching-Chun Wang

Undergraduate Students

Neelima Yeddanapudi

Technical and Support Staff

Susan E. Chafe, Dorothy A. Fleischer

1.1 Interactive Learning Environment for Integrated Circuit Design

Sponsor

IBM Corporation

Project Staff

Professor Jonathan Allen, Dr. Christopher J. Terman

Progress continues on the implementation of an interactive learning environment for VLSI design. The major goal of this research is to design a computer-based learning environment that includes a text with active links to schematic and layout editors, along with extraction and simulation programs. Using this arrangement, the value added over currently available techniques is that several tools and displays are seamlessly integrated, using easy commands, to provide a "low barrier" means to explore the detailed behavior of integrated circuits.

Text chapters are coordinated with an expandable table of contents (much like a PC file system description) so it is easy to jump around in the text in a "non-linear" manner. The figures initially provided with the text chapters have been redrawn in digital form and inserted in the appropriate place within the text. For all the text figures that are circuit schematics (and the circuit level of representation is considered central in this approach), a button is provided in the text next to the figure. When the user clicks on this button, a new environment window, called JADE, is opened. Initially, this window shows the circuit schematic, but it is now the display of a circuit schematic editor. A complete library of circuit primitives is provided, a variety of sources and simulation aids (such as "ammeters") is available, and the user can access several libraries that contain previously designed circuit modules. In this way, the circuit presented in the text can be easily modified. (Of course, the user can also design a circuit from scratch, which can then be compared with other designs that provide the same functionality.)

Once the user has determined a schematic design, it can be "extracted" into the form of a SPICE input deck. It is often useful to inspect this representation of the circuit, and it can be easily edited as a text file. Next, the user can simulate the circuit by using a modified version of SPICE. A waveform display program is provided that seeks to emulate the look and feel (in terms of controls) of an oscilloscope. It is easy to read off-time and voltage (or current) values from waveforms, and to compute time and voltage (or current) differences, which are convenient to have. The display can be scaled and zoom control is provided. The display of the circuit schematic and the corresponding waveform display are shown side by side, so that the user can easily modify the circuit and ascertain the waveform results of the change quickly and easily.

1 Massachusetts Eye and Ear Infirmary, Boston, Massachusetts.

In order to gain a sense of the utility of this learning environment, consider the design of a simple two-input NAND gate in CMOS. While most students readily understand the DC behavior of the NAND gate and the complementary nature of the pullup and pulldown circuits, it is less common to probe the interior nodes of the circuit in order to understand factors that determine the performance of the circuit. For example, if the sequence of inputs is such that both pulldown FETs are ON, and then the bottom one is turned off, the node between them will charge up to one threshold (including body effect) down from the supply voltage, and the upper FET is (just) cut off. In addition, this upper FET has a large body effect, and when the bottom FET goes ON, it is important to pull down this interior node as fast as possible so that the upper pulldown FET goes ON quickly. This can be facilitated by widening the bottom pulldown FET. Thus, the effect on the speed of the NAND gate can be easily seen in this learning environment, and the effect on circuit performance of device parameter changes is easily understood. It is important to realize that even simple circuits, such and the two-input NAND, often have interesting aspects revealed by a detailed simulation, and that a student's confidence in understanding is enhanced by using this learning environment.

Substantial "tuning" of the environment is underway. Also, the layout environment, with editor and display must be completed and integrated into the environment. When this is done, it will be possible to extract the SPICE input "deck" from the circuit schematic, or the layout, or both, so they can be compared. The environment is coded in Java, and hence can run on many platforms, including over the network as an application controlled by a browser. Thus, the environment should be highly portable and accessible.

1.2 Vision Project

Sponsors

W.M. Keck Foundation
Catalyst Foundation

Project Staff

Dr. Joseph F. Rizzo, III, Professor John L. Wyatt, Jr., Michael Socha,[2] Dr. Mohamed Shahin,[3] Dr. Douglas Shire,[4] James Howard,[5] Terry Herndon,[6] Dr. Ralph Jensen,[7] Dr. John Loewenstein,[8] Danielle LeBlanc,[9] Andrew E. Grumet, Bradley J. Lichtenstein, Shawn G. Kelly, Joshua D. Moss, Neelima Yeddanapudi

1.2.1 Description of Progress to Date

The vision project seeks to develop an implantable retinal prosthesis to restore vision to blind patients. Individuals who are born with normal sight but then lose vision slowly from degeneration of their rods and cones are potential candidates. Retinitis pigmentosa and age-related macular degeneration are two important forms of blindness we hope to treat.

Our primary immediate goals are to perform two proof-of-concept experiments. The first consists of human surgical trials lasting only two to three hours to assess what blind patients can see with a simple 10 x 10 microelectrode array placed against the retina. This device will be removed immediately after the experiment, as it is not in a form suitable for chronic implantation. The second is a series of animal surgical trials in which a simple prototype prosthesis is surgically implanted, the rabbit eye is sewn up with no hardware penetrating the eye, the prosthesis is stimulated via laser and the induced activity in the visual cortex is electrically recorded. If successful, this experiment will demonstrate that our concept is feasible in principle, while the first will hopefully demonstrate that perceptual quality is sufficiently high to be of practical use to blind patients. The remaining problem of long-term biocompatibility will require major additional effort.

2 Charles Stark Draper Laboratory, Cambridge, Massachusetts.
3 Massachusetts Eye and Ear Infirmary, Boston, Massachusetts.
4 Cornell Nanofabrication Facility, Ithaca, New York.
5 MIT Lincoln Laboratory, Lexington, Massachusetts.
6 Ibid.
7 Southern School of Optometry, Memphis, Tennessee.
8 Massachusetts Eye and Ear Infirmary, Boston, Massachusetts.
9 Ibid.

Preparation for Human Surgical Trials

Dr. Douglas Shire and Professor John Wyatt have designed a thin, flexible 10 x 10 microelectrode array that is mechanically suitable for surgical implantation in the eye of a blind volunteer for several hours. Dr. Shire has now fabricated a number of these. Electrical tests on the platinum arrays delivered show they are of high quality, with very few open circuits or short circuits among the 100 microelectrodes. However an alternate material, activated iridium, allows roughly ten times more charge than platinum to be applied safely to the retina. We have decided to use activated iridium microelectrodes in the human experiments as soon as they are ready, because the expanded safe stimulation levels increase our chances of collecting useful data from the patients. Dr. Shire has delivered the first two batches of iridium arrays, and Professor Wyatt and graduate student Shawn Kelly have tested and activated (i.e., oxidized) several of these arrays in preparation for the first human trial.

Terry Herndon and James Howard assist as part-time consultants. Mr. Herndon designed the printed circuit connector which attaches to the electrode arrays, as well as the gold weight which attaches to the retinal end of the electrode array to facilitate surgical handling. Mr. Howard has built several of the printed circuit connectors, and their cable harnesses to connect to the stimulator unit, and he has built several gold weights and developed a method for attaching them to the arrays in a biocompatible way.

Graduate students Shawn Kelly and Bradley Lichtenstein are retained as research assistants to develop the electronics and software needed for the human surgical trials. Mr. Kelly has designed and built the stimulator system to be used in the first human tests. The battery-powered system consists of several different units. The digital unit controls timing parameters of the stimulus waveform while an analog unit generates the desired waveform shape. An array of ten current sources and demultiplexers delivers the stimulus current pulses to the appropriate outputs. A switching control grid, consisting of 100 switches, is used to set the stimulus pattern, with each switch determining whether a given electrode will deliver current to the retina. The user interfaces the stimulation system through an array of switches, buttons, and knobs to set the timing and shape parameters of the stimulus, the level of stimulus current, and the configuration of the output. This system is also used in testing the arrays for electrical flaws resulting from the fabrication process. It is now ready for use in the early human surgical trials.

Mr. Lichtenstein has designed a more advanced second-generation stimulator with two new interrelated features for human testing: a user-friendly graphic user interface (GUI) that allows us to intuitively and easily control with a personal computer (PC) the course of a human test in an operating room, and the electronics outside the PC which the PC must respond to and control in order to stimulate the retina. When complete, this system will allow us to present moving images to the patient, to rapidly alter the stimulus in response to patient reaction, to continuously monitor the electrical condition of the stimulating array and to automatically record the entire stimulation history throughout the surgery for later replay.

The retinal surgeon, Dr. John Loewenstein, came to the laboratory on five occasions to learn and practice the surgical techniques for retinal stimulation that we had learned in the rabbit. Practicing on human cadaver eyes, Dr. Loewenstein quickly became comfortable with the techniques. Several small but significant design changes were implemented for the metal weight and post used to place the array on the retina. Also, we added a long suture to the metal weight that will emerge from the eye during the human test, permitting a more secure method of retrieving the array from the eye if any problems develop. Also, new intraocular forceps were ordered to facilitate the procedure.

Despite the fact that only blind surgical volunteers are accepted, we need to be certain that none of the toxic chemicals used in fabricating the microelectrodes remain present in the completed arrays in a form that could further damage the patient's eye during or after surgery. With the help of Ann Hynes, a chemist at Draper Laboratories, we have begun a sequence of experiments in which the electrodes are immersed in aqueous solutions under conditions that closely imitate the planned eye surgery. The solutions are subjected to chemical tests for contaminants. The initial tests have given encouraging results.

Dr. Rizzo has met several times with a blind retinitis pigmentosa patient who has volunteered for the first human trial. His medical history seems appropriate,

and his attitude in conversations with Dr. Rizzo and Professor Wyatt appears interested, alert, and cooperative.

Dr. Rizzo and Professor Wyatt have obtained final approval to proceed with human trials from MIT's Committee on the Use of Humans as Experimental Subjects. They have obtained approval from the Massachusetts Eye and Ear Infirmary Human Studies Committee, conditional on the chemical tests described above.

Biocompatibility Tests

Dr. Rizzo, with assistance from Dr. Mohamed Shahin, has carried out *in vivo* biocompatibility experiments on two rabbits. A microfabricated electrode array, described in the previous section, was placed into the mid-vitreous cavity of one eye of each rabbit. The protocol is the same as that used to assess the safety of six other materials in the eye: electroretinograms (to measure the electrical activity of the retina) are carried out once before surgery and five times after implantation.

The first animal shows no problems after three weeks. The second animal died one week after implantation for reasons not related to the implant. After the pre-operative sedation, used to obtain a baseline ERG, the rabbit became ill, but appeared to the veterinarian to have largely recovered after several days of poor feeding. But the rabbit died hours after the second anesthesia. This type of problem has occurred intermittently at MIT and has prompted a prospective evaluation by the staff veterinarians. They strongly suspect that some rabbits develop a cardiac problem from the anesthetic, which is important because it is the standard anesthetic for rabbit research. The staff veterinarian unequivocally dismisses the possibility that the array was responsible for the death. We have recorded a preoperative ERG from a third rabbit, and implantation of the test array will be made next week.

In Vivo Retinal Stimulation in Rabbits

Drs. Rizzo and Shahin and Mr. Kelly have performed *in vivo* rabbit stimulation studies with platinum and then iridium arrays similar to the array that will be used for the human test. Both arrays gave similar robust cortical signals. We also learned that initial negative-going stimulation pulses consistently give larger cortical responses. This pulse pattern was suggested to us in a meeting of the Keck group.

The immediate goal for our animal studies was described in the second paragraph at the beginning. Toward this goal, Mr. Moss and Dr. Rizzo are developing an intraocular lens to fit in the rabbit eye and serve as a support structure for the photodiode power source. Mr. Moss, working with Michael Socha at Draper, has learned to use a high-powered engineering design software program, the output of which controls a lithography machine. Improvements that he has made include enlarging the central area, to facilitate surgical implantation, and boring extremely fine holes to place "wings" that serve as supporting structures. Unfortunately, the lithography machine cannot make such fine structures. Mr. Moss developed a technique to mount commercial haptics by drilling tiny 4 mil holes without relying on the lithography machine. Working with Mr. Herndon and Mr. Howard, he has mounted a photodiode array on the lens, to which we attached the electrode array using epoxy. In the early designs we also attach two ultrafine wires to allow us to measure current output of the array during preliminary surgical trials.

We are also investigating biocompatible adhesives for safely attaching the microelectrode array to the rabbit retina. We have met several times with a local company to begin testing a material of theirs that we previously assessed and found to have excellent attributes (biocompatible; transparent; can be hardened in the eye by short exposure to UV light, which makes it surgically friendly.) We will begin additional tests within two weeks.

The next goal for our animal studies is chronic implantation of a stimulation system in the rabbit eye. The implant must be encapsulated in materials that protect it from the saline intraocular environment. Within the next week we will encapsulate an intraocular lens/photodiode system using NuSil compounds previously shown by Edell to be effective. Mr. Socha and Mr. Moss have copied a resistive leakage test system for these devices that was developed by Dr. Edell and it is now ready for use in device testing.

In Vitro Retinal Stimulation Studies in Rabbits

Doctoral student Andrew Grumet conducted experiments to characterize retinal responses to microelectrode array stimulation, and made methodological refinements. His first round of in vitro experiments, performed over the summer, revealed interesting physiologic phenomena as well as technical limitations of his experimental apparatus. Neural spike

activity could be produced with stimulation currents as low as 1 microampere, often in trains lasting up to thirty milliseconds after the termination of the stimulus. The long spike latencies suggested a trans-synaptic mechanism, in which delay-producing chemical synapses separate the directly stimulated cell from the cell under study. But in experiments performed this fall, addition of the synaptic blocker cadmium to the bathing solution failed to eliminate the latent activity. An alternate explanation is that the stimulation currents used, though seemingly small, are nonetheless far above the ganglion cell stimulation threshold for producing single spikes.

Thresholds for producing single spikes, which occur within the first few milliseconds following stimulus application, could not be measured due to stimulus artifacts. With the help of Professor Wyatt, Mr. Grumet performed a detailed electrical characterization of the experimental apparatus, with the goal of identifying and reducing the dominant source(s) of the artifact. The effort identified a number of such sources; the major one being leakage paths on the array between lead wires from adjacent electrodes.

Mr. Grumet is now designing new electrode arrays to minimize such paths. Dr. Shire will fabricate them at Cornell, and Mr. Grumet will shortly conduct new experiments to probe the lower limits of current required to produce neuronal activity.

1.2.2 Recent Problems and Unexpected Developments

It is harder than we had hoped to make iridium microelectrode arrays with less than 10% electrode faults, due to iridium sputtering during deposition. The iridium layer also develops thousands of surface cracks with our present deposition technique.

Serious liability and insurance problems at Draper Lab had to be resolved before Draper could help us with chemical analysis of possible contaminants.

During our biocompatibility study, one rabbit died for reasons unrelated to the implant.

1.2.3 Further Progress Expected Within the Next Six Months

We expect to carry out initial versions of the two key proof-of-concept experiments in the next six months:

1. One or more short-term human surgical trials to determine what the patient can perceive with a 10 x 10 microelectrode array.
2. Rabbit trials in which we implant a simple laser-driven stimulator and record a cortical signal.

We also expect to achieve short-term encapsulation of a prosthesis in biocompatible materials and verify the protection it provides against saline leakage using the resistance tests. We believe Mr. Grumet will have made good quality in vitro recordings of rabbit retina response to electrical stimulation using the new microelectrode arrays he has designed that Dr. Shire will have fabricated. We expect Mr. Lichtenstein will have completed a working graphics-driven stimulation system for use in subsequent human trials. Mr. Grumet will have presented a poster session at a major conference (ARVO) on his work, and Mr. Kelly will have completed his M.S. thesis on the stimulator for human trials. We expect that Dr. Shire will have overcome the final difficulties and will have delivered iridium arrays with very few short circuits, open circuits or surface metal cracks for use in future human experiments.

1.2.4 Publications

Rizzo J.F., and J.L. Wyatt. "Prospects for a Visual Prosthesis." *Neuroscientist* 3(4): 251-62 (1997).

Rizzo, J.F., A.E. Grumet, D.J. Edell, J.L. Wyatt, and R.J. Jensen. "Single Unit Recordings Following Extracellular Stimulation of Retinal Ganglion Axons in Rabbits." Annual Meeting of the Association for Research in Vision and Ophthalmology, Fort Lauderdale, Florida, May 1997.

Wyatt, J. "Steps toward the Development of an Implantable Retinal Prosthesis." Charles University, Department of Ophthalmology, Prague, Czech Republic.

Wyatt, J. "Steps toward the Development of an Implantable Retinal Prosthesis." European Congress of Ophthalmology, Budapest, Hungary.

Wyatt, J. "Steps toward the Development of an Implantable Retinal Prosthesis." Allgemeine Krankenhaus, Department of Ophthalmology, Vienna, Austria.

Wyatt, J. "Steps toward the Development of an Implantable Retinal Prosthesis." Eighth Great Lakes Symposium on VLSI, Lafayette, Louisiana.

Wyatt, J. "Steps toward the Development of an Implantable Retinal Prosthesis." Department of Electrical Engineering, Georgia Tech, Atlanta, Georgia.

1.3 Cost-Effective Hybrid Vision Systems for Intelligent Highway Applications

Sponsor

National Science Foundation
Grant MIP 94-23221

Project Staff

Professor Berthold K.P. Horn, Professor Hae-Seung Lee, Dr. Ichiro Masaki, Professor Charles G. Sodini, Professor John L. Wyatt, Jr., Haigian Cheng, Yajun Fang, Nicole S. Love, Pablo A. Serafini, Mark G. Spaeth, Zubair Talib, Ching-Chun Wang

1.3.1 Project Summary

A key concept at MIT's Intelligent Vehicle Research Program is the integration of multiple aspects of the field of intelligent transportation systems (ITS) to produce cost-effective hybrid vision systems for intelligent highway applications. We are trying to integrate research on different scale systems ranging from small novel integrated chips; medium scale projects, such as three-dimensional machine vision systems; and large scale, such as total transportation systems. The component research at each stage indicates what options are available in each frame for a total transportation system architecture. This integration requires the cooperation of researchers in various fields. The following sections detail substantial progress in four subprojects.

1.3.2 Brightness Adaptive Television Camera Chip

For monitoring traffic and detecting obstacles on highways, the intensity dynamic range of 1:10,000 is required for television cameras. Conventional cameras, however, can provide only 1:60 with cheap cameras or 1:1,000 with cooled expensive slow cameras. The goal of this project is to develop a TV camera which has enough intensity dynamic range for ITS applications.

One way to extend the intensity dynamic range is to convert the relationship between the incoming light and the camera output voltage from linear to logarithmic characteristic, as shown in Figure 1. A drawback of this approach is that intensity resolution is decreased.

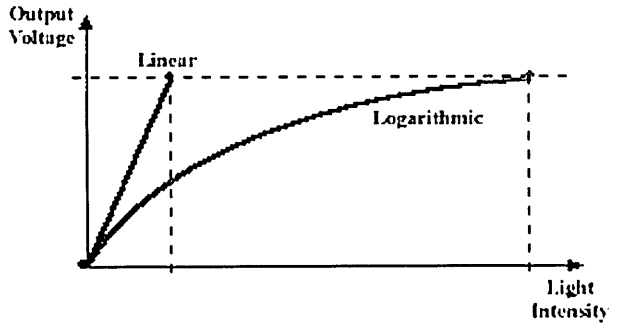

Figure 1. Linear versus logarithmic.

Our approach to dynamically change the relationship between the incoming light and the camera output voltage from frame to frame, depending on the required dynamic range. When the required dynamic range is small, the input-output relationship becomes closer to the conventional linear scheme for better intensity resolution. The logarithmic characteristic is increased as the required dynamic range increases at a cost of intensity resolution.

The first version of 256x256-pixels camera has been developed. This version has two additional features: (1) CMOS technology was used instead of conventional CCD, and (2) parallel 10-bit analog-to-digital converters were included in the imager chip. The pixel size is 22x22 microns. Real-time format converter is under development to provide 30 frames per second. Figure 2 shows that the intensity dynamic range can be extended over 1:10,000 ratio with the compressive mode.

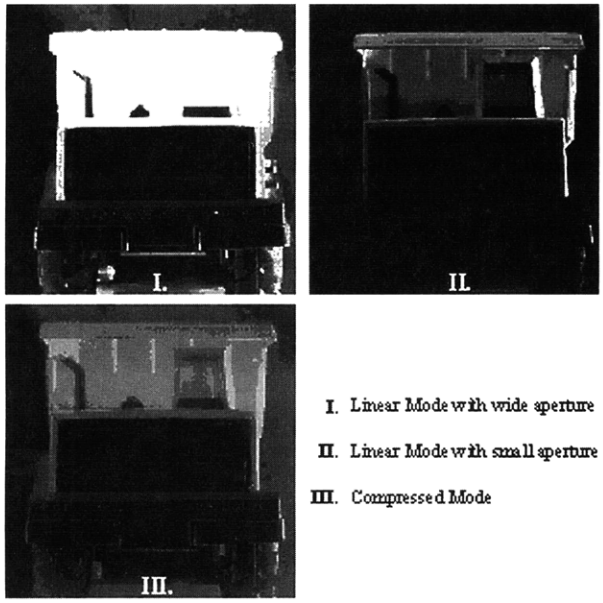

Figure 2. Experimental results (front view of track).

1.3.3 Pixel-Parallel Image Processing System for Intelligent Transportation System Applications

Parallel processing systems are expected to offer more flexibility than application-specific hardware and higher speed than serial processors. However, this type of system is conventionally too expensive and large for intelligent transportation system applications. The challenge is to develop a real-time pixel-parallel image processing system with enough flexibility to handle a wide variety of algorithms at reasonable cost.

An additional problem for this type of system is rapid prototyping without access to hardware for testing. This system includes an application framework in which programs execute entirely within software simulation of the system providing accurate output to allow application debugging before the task is executed in hardware.

Most previous image/radar processing systems used application specific integrated circuits (ASICs), digital signal processors (DSPs), and/or personal computers (PCs). The ASICs are difficult to carry out various algorithms in a time-sharing fashion, and the PCs are too slow in many cases.

Our approach is to develop a programmable parallel processor which carries out various algorithms, such as stereo and optical-flow, in a time-sharing fashion with a reasonable cost and system size. The key was to simplify the processor architecture without sacrificing performance for intelligent transportation applications. The high-density pixel-parallel image processing system allows 256x256 gray scale images to be transferred into the array at frame-rate, with an overhead of slightly more than 1ms.

The array consists of 4 HDPP chips. The HDPP chip was developed in our group; the area of the chip is less than 80 mm^2 with 0.6 μ technology. Each chip is capable of processing 64x64 pixel images, containing one processing element for each pixel. Each pixel-processing element contains 128 bits of memory and a one bit 256 function arithmetic logic unit (ALU) by which to perform the actual processing.

A C++ software framework was developed for the system. The software framework provides a means for application programmers to write and develop high-level programs that can be tested and debugged in a software simulation environment and then compiled to run in the hardware implementation. To preclude the programmer from needing to understand the actual hardware implementation, a higher level C++ based instruction set was developed. The programmer develops an application by constructing sequences of instructions. Instructions exist for many useful operations in the array such as adding, multiplying, comparing, etc.

The current version of the system is a 128x128-pixel processor. Several applications have been tested on the system with the results shown in Table 1.

Table 1: Results of test of applications on the systems.

Application	*Execution Time*
Edge Detection	277 μs
5x5 Median Filtering	274 μs
Template Matching	3.25 ms
Stereo Vision	4.00 ms
Optical Flow	9.90 ms

1.3.4 Compact-Size Array Processor with Multi-Instruction Multi Data (MIMD) Capability

Multi-instruction multi-data (MIMD) array processors have high performance for image processing, radar signal processing, and other signal processing because of their high parallelism and flexibility. Conventional MIMD processors are too large and expensive for intelligent transportation applications. The goal was to develop a compact-size, inexpensive array processor with a MIMD capability.

For signal processing applications which have strict limitations in terms of the system cost and size, single-instruction multi data (SIMD) array processors, digital signal processors (DSPs), or application specific integrated circuits (ASICs) were commonly used.

Each processing element consists of an ALU (Arithmetic/Logic Unit), a data flow control unit, instruction memory, and data memory. Replacing a conventional digital ALU with a mixed-signal ALU reduced the size of the processing element. The mixed-signal ALU utilizes both analog and digital circuits. Its specialized feature is a small silicon area to achieve limited accuracy. The small size helped to implement the MIMD capability at low cost. The size of each processing element of our prototype, fabricated in a 5V 0.8 μm triple metal CMOS process, is 700 μm by 270 μm, and therefore 529 processing elements could be implemented within 1 cm^2 silicon area. The accuracy is comparable to a 7-bit digital processor.

The prototype chip was applied to a stereo vision algorithm successfully in an off-line mode. The algorithm includes gradient calculations, correlations, and subpixel interpolations. A real-time operation is under development.

1.3.5 Cost-Effective Real-Time Three-Dimensional Vision System

A major obstacle in the application of stereo vision to intelligent transportation systems is high computational cost. Stereo vision systems are either too slow or use a large amount of processing hardware. Intelligent transportation systems require real-time results at a reasonable cost.

Our goal is to build a system that achieves an acceptable level of performance using off-the-shelf components. Therefore, we chose off-the-shelf components to minimize the cost of the system as opposed to application specific integrated circuits (ASICs).

Intelligent transportation systems use mainly one or two-dimensional measurements for applications. Conventionally laser radar sensors are used for on-vehicle applications, and loop detectors and closed circuit television are used for on-road applications. Stereo vision systems have been studied for many years but have been too costly for real-time applications. Previous stereo systems have focused on area correlations to achieve three-dimensional measurements which require a large amount of computation making the systems extremely slow or very expensive.

Our approach is an edge-based, subpixel stereo algorithm which is adapted to permit accurate distance measurements to objects in the field of view. The 3D scene information may be directly applied to a number of in-vehicle applications such as intelligent cruise control, obstacle detection, and lane tracking, as well as on-road applications such as vehicle counting, classification, speed measurements, and incident detection.

Our system has three major components: camera assembly, PC-based image processing hardware, and the PC itself. The system uses three standard cameras. The cameras are equally spaced with their optical axes aligned. In our experimental setup, the left and right cameras are 252 mm apart with the center camera located midway between them. The lenses have focal length 35 mm, yielding a horizontal field of view of 10.4 degrees and vertical field of view of 7.9 degrees when used with our cameras.

The images are acquired simultaneously from all three cameras. The first step is to generate the vertical edge gradient for each image, which is accomplished in the image processing hardware as a convolution. Edge points in the left and right images are matched using the center camera to help eliminate false correspondences. For corresponding pixels, edge positions in the left and right images are estimated to subpixel resolutions and used to arrive at a subpixel disparity.

Subpixel resolution is used to achieve greater accuracy in depth computation. Given three gradient amplitudes, quadratic interpolation is performed to arrive at a subpixel edge location. From each subpixel disparity, the depth may be calculated directly from:

$$z = b\frac{f}{x'_l - x'_r}$$

where b is the length of the baseline, f is the focal length, and the x variables are x-coordinates in the left and right images respectively.

Table 2: Summary of performance benchmarks of our system. The performance benchmarks from on-road trials indicate real-time depth recovery.

CPU	100 MHz
Image Size	512x80
Convolution	3x3
Disparity Range	0-400
Speed	15 frames/sec

1.4 Computer-Aided Design Techniques for Embedded System Design

Sponsor

National Science Foundation
Contract MIP 96-1232

Project Staff

Professor Srinivas Devadas, Daniel Engels, George I. Hadjiyiannis, Silvina Z. Hanono

1.4.1 Embedded Systems

Recent years have seen an amazing growth in the use of consumer electronic products such as personal digital assistants, multimedia systems, and cellular phones. Due to their enormous market demand, the manufacturing of electronic systems is very sensitive to cost and time requirements. Moreover, many applications (e.g., cellular phones) have stringent requirements on size and power consumption, for the sake of portability. For these reasons, manufacturers profit from integrating an *entire system* on a single integrated circuit (IC).

As time-to-market requirements place greater burden on designers for short design cycles, programmable components are introduced into the system, and an increasing amount of system functionality is implemented in *software* relative to hardware. Systems containing programmable processors that are employed for applications other than general-purpose computing are called *embedded systems*. An example is a system consisting of a digital signal processor (DSP) core or an application-specific instruction-set processor (ASIP), a program ROM, RAM, application-specific circuitry (ASIC), and peripheral circuitry.

1.4.2 Hardware-Software Co-Design

Rather than designing the software and hardware components of an embedded system separately, *hardware-software co-design* is more cost effective and results in a shorter time to market. In particular, in order to address the constraints of cost, size, and power consumption, it is important to match the architecture of the embedded processor to the application at hand.

In a co-design methodology, designers partition the system functionality into hardware and software. Additionally, a target processor is chosen from existing processor designs, or an ASIP is designed to execute the software. The hardware, software, and ASIP are implemented, and the resulting system is evaluated using a hardware-software co-simulator. The partitioning and processor design are repeated until an acceptable system is developed. With this methodology, tools for *code generation* and *hardware-software co-simulation* are essential for the successful evaluation of candidate architectures.

As the complexity of embedded systems grows, programming in assembly language and optimization by hand are no longer practical except for time-critical portions of the program that absolutely require it. Further, hand coding virtually eliminates the possibility of changing the processor architecture. Thus, an automatic code generation methodology is required for adequate evaluation of candidate architectures and effective architecture exploration. In addition, the automatic code generation methodology will be most useful if it can be easily adapted to generating code for different processors. The same holds for the simulation environment. This property, commonly called *retargetability*, is discussed in the following section. We argue that to be able to explore the processor design space, a set of automatically retargetable design tools including code generators and simulators is essential.

1.4.3 A Machine Description Language for Easy Retargetability

In order to determine the appropriate architecture for the processor being designed, the design space may be explored using iterative improvement. First, an initial architecture is generated and described using a *machine description language*. This machine description is then fed into a retargetable compiler along with the application source code. The retargetable compiler translates the source code to assembly code for the target processor, optimized for speed and code size. The same machine description is also passed to a retargetable assembler that converts the assembly code to binary code for the target processor. Next, a retargetable simulator receives the machine description and the binary code as inputs and executes the code. The simulator generates a set of measurements that can be used to evaluate the architecture and point out possible improvements. If the performance of the architecture does not meet the design specifications, or if the cost, size, or power consumption can be reduced without sacrificing performance, then the appropriate changes are made to the architecture and a new machine description is generated. Once the final architecture has been determined, the machine description can be used to generate an implementation of the architecture.

The machine description language is the focal point of all the tools used for architecture exploration. It is a critical component in the design flow and should be capable of performing the following functions:

- Specify a wide variety of architectures. In particular, it should support very long instruction word) (VLIW) architectures because they are very efficient for custom applications, and because more traditional architectures can be treated as degenerate cases of VLIW architectures.
- Explicitly support constraints that define valid instructions.
- Be easily understandable and modifiable by a compiler writer or hardware architect.
- Support automatically retargetable code generation.
- Support the automatic generation of an assembler and disassembler.
- Support the automatic generation of a cycle-accurate instruction level simulator.
- Provide adequate information to allow for code optimizations.
- Support the generation of an implementation of the architecture from the machine description.

We have developed a machine description language, ISDL instruction set description language (ISDL), which has all of the above features. We have written complete ISDL descriptions for ASIPs, as well as commercial DSP cores; in particular, a powerful ASIP VLIW architecture and the Motorola 56000 DSP.

1.4.4 Retargetable Code Generation

We are developing a retargetable code generator, AVIV, whose inputs are the application program and a ISDL description the target processor. AVIV produces code, optimized for minimal size, that can run on the target processor. Code size is our optimization cost function because we are focusing on embedded processors where the size of the on-chip ROM is a critical issue.

By varying the machine description and evaluating the resulting object code, the design space of both hardware and software components can be effectively explored. AVIV focuses on architectures exhibiting instruction-level parallelism (ILP), including VLIW architectures.

AVIV uses the Stanford University intermediate format (SUIF) and Synopsys, Princeton, Aachen, MIT (SPAM) compilers as a front-end. The front-end performs machine independent optimizations and generates an intermediate representation of the source program which consists of basic block directed acyclic graphs (DAGs) connected through control flow information.

Retargetable code optimization requires the development of parameterizable algorithms for instruction selection, resource allocation, and scheduling. AVIV addresses these code generation subproblems concurrently, whereas most current code generation systems address them sequentially. The main reason why current code generators address these problems sequentially is to simplify decision-making in code generation. However, decisions made in one phase have a profound effect on the other phases.

AVIV converts the intermediate representation generated by the front-end to a graphical (Split-Node DAG) representation that specifies all possible ways of implementing the basic blocks on the target processor. The information embedded in this representation

is then used to set up a heuristic branch-and-bound step that performs functional unit assignment, operation grouping, register bank allocation, and scheduling concurrently. While detailed register allocation is carried out as a second step, required loads and spills due to limits on available registers are generated and scheduled during the first step. This ensures that a valid detailed register allocation can always be found without undoing the chosen operation groupings or functional unit assignments. The heuristics in AVIV are tuned for minimal code size.

The scheduling problems faced in retargetable code generation are complicated by the presence of multiple pipelined datapaths, particularly in high-performance processor cores. Optimally scheduling operations on these multiple datapaths so as to minimize the overall completion time is critical to system performance. Unfortunately, this is a difficult problem due to data dependencies between operations and pipeline interlocks. The pipeline interlocks introduce (possibly varying) distance constraints between data dependent operations, and scheduling so as to avoid interlocks is referred to as the distance-constrained scheduling problem. While scheduling with fixed distance constraints has received some attention, scheduling with arbitrarily varying distance constraints has not.

We have examined the complexity of the arbitrary distance-constrained scheduling problem, finding it to be NP-complete even for simple constraint topologies. Approximation bounds for this problem have been derived, and a heuristic algorithm has been developed.

1.4.5 Retargetable Simulation

The simulator generator uses exactly the same ISDL description of the target architecture. The list of operations and corresponding effects on visible state are used to create a set of functions (one for each operation), each of which modifies the internal representation of the visible state in the same way as the corresponding operation would modify the hardware state. The appropriate functions are then triggered as the simulator steps through the instructions in the binary program. There are two main problems that the simulator must deal with:

- Disassembly: First, the simulator must decompose each VLIW instruction to the corresponding operations. This corresponds to disassembly of the original instruction stream and can be performed upon initialization. This is a non-trivial problem because various operations in the instruction may have overlapping binary images.
- Timing: To be useful, the simulator must be cycle accurate at least at the instruction level. Therefore, the simulator must be able to reproduce effects related to various pipeline features such as bypass logic and stalling. This information is available to the simulator on a per-operation basis.

The simulator contains a disassembler which makes use of the binary images of each operation (also listed in ISDL) and then searches through combinations of these to arrive at the particular group of operations that yielded a given VLIW instruction image. A large number of shortcuts are used to prune the search when there is little or no overlap in the binary images of the operations. The whole program is disassembled when loaded in memory. This results in fast simulations but prevents the use of self-modifying code.

The timing is related to two parameters: cycle cost and latency. Cycle cost is the number of clock cycles it will take for an instruction to execute on hardware. Latency describes how many instructions later the effects of the current instruction will be visible. Between the two, these parameters can describe a wide variety of pipeline architectures including features such as bypass logic and stalling. Since these parameters can vary on a per-operation basis, the simulator can handle pipelines of different lengths for different operations.

The generated simulators can run on the order of 200 Kcycles/sec and offer a user-friendly interface and a wide variety of debugging support. Performance statistics for the target architecture can be generated from an address trace that the simulator creates on command.

1.5 Functional Verification of VLSI Systems

Sponsor

Schlumberger Foundation

Project Staff

Professor Srinivas Devadas, Farzan Fallah

1.5.1 Coverage Analysis

Functional simulation is still the primary workhorse for verifying the functional correctness of hardware designs. Functional verification is necessarily incomplete because it is not computationally feasible to exhaustively simulate designs. It is important therefore to quantitatively measure the degree of verification coverage of the design. Coverage metrics proposed for measuring the extent of design verification provided by a set of functional simulation vectors should compute statement execution counts (controllability information), and check to see whether effects of possible errors activated by program stimuli can be observed at the circuit outputs (observability information). Unfortunately, the metrics proposed thus far, either do not compute both types of information, or are inefficient, i.e., the overhead of computing the metric is very large.

We have developed an efficient method to compute an observability-based code coverage metric (OCCOM) that can be used while simulating complex HDL designs. This method offers a more accurate assessment of design verification coverage than line coverage, and is significantly more computationally efficient than prior efforts to assess observability information because it breaks up the computation into two steps: Functional simulation of a modified HDL model, followed by analysis of a flowgraph extracted from the HDL model. Commercial HDL simulators can be directly used for the time-consuming first step, and the second step can be performed efficiently using appropriate data structures.

1.5.2 Functional Vector Generation

Our strategy for automatic generation of functional vectors is based on exercising selected paths in the given hardware description language (HDL) model. The HDL model describes interconnections of arithmetic, logic and memory modules. Given a path in the HDL model, the search for input stimuli that exercise the path can be converted into a standard satisfiability checking problem by expanding the arithmetic modules into logic-gates. However, this approach is not very efficient.

We have developed a new HDL-satisfiability checking algorithm that works directly on the HDL model. The primary feature of our algorithm is a seamless integration of linear-programming techniques for feasibility checking of arithmetic equations that govern the behavior of datapath modules, and 3-SAT checking for logic equations that govern the behavior of control modules. This feature is critically important to efficiency, since it avoids module expansion and allows us to work with logic and arithmetic equations whose cardinality tracks the size of the HDL model.

1.5.3 Publications

Journal Articles

Hadjiyiannis, G., A.P. Chandrakasan, and S. Devadas. "A Low-Power, Low-Bandwidth Protocol for Remote Wireless Terminals." *ACM Trans. Wireless Comput.* Forthcoming.

Liao, S., S. Devadas, K. Keutzer, S. Tjiang and A. Wang. "Code Optimization Techniques in Embedded DSP Processors." *Des. Automat. Embed. Syst. Int. J.* 3(1): 59-74 (1998).

Liao, S., S. Devadas, and K. Keutzer. "A Text-Compression-Based Method for Code Size Minimization in Embedded Systems." *ACM Trans. Des. Automat. Electron. Syst.* Forthcoming.

Liao, S., K. Keutzer, S. Tjiang, and S. Devadas. "A New Viewpoint on Code Generation for Directed Acyclic Graphs." *ACM Trans. Des. Automat. Electron. Syst.* Forthcoming.

Monteiro, J., S. Devadas and A. Ghosh. "Optimization of Sequential Logic Circuits Using Precomputation." *IEEE Trans. Comput.-Aided Des.* Forthcoming.

Monteiro, J., S. Devadas, A. Ghosh, K. Keutzer, and J. White. "Estimation of Average Switching Activity in Combinational Logic Circuits Using Symbolic Simulation." *IEEE Trans. Comp.-Aided Des.* 16(1): 121-27 (1997).

Conference Papers

Costa, J.C., J. Monteiro, and S. Devadas. "Switching Activity Estimation Using Limited Depth Reconvergent Path Analysis." *Proceedings of the International Symposium on Low Power Electronic Design*, Monterey, California, August 1997, pp. 184-89.

Hadjiyiannis, G., S. Hanono and S. Devadas. "ISDL: An Instruction Set Description Language for Retargetability." *Proceedings of the 34th Design Automation Conference*, Anaheim, California, June 1997, pp. 299-302.

Hanono, S., and S. Devadas. "Instruction Selection, Resource Allocation, and Scheduling in the AVIV Retargetable Code Generator." *Proceedings of the 35th Design Automation Conference*, San Francisco, California, June 1998. Forthcoming.

Liao, S., and S. Devadas. "Solving Covering Problems Using LPR-Based Lower Bounds." *Proceedings of the 34th Design Automation Conference*, Anaheim, California, June 1997, pp. 117-20.

Sudarsanam, A., S. Liao and S. Devadas. "Analysis and Evaluation of Address Arithmetic Capabilities in Custom DSP Processors and ASIPs." *Proceedings of the 34th Design Automation Conference*, Anaheim, California, June 1997, pp. 287-92.