SIGDA NEWSLETTER

A Quarterly Publication of the Special Interest Group on Design Automation

VOLUME 20 NUMBER 1 JUNE 1990

CONTENTS PAGE

****

SPECIAL DESIGN AUTOMATION CONFERENCE ISSUE
Attend the SIGDA Annual Meeting at DAC—Sunday 7:00 pm

****

SIGDA BOARD OF ADVISORS

Nominating Committee Chair
Richard C. Smith
National Semiconductor, Inc.
2900 Semiconductor Drive, D3677
Santa Clara, California 95051
(408) 721-4794 FAX (408) 773-0978
nsc!parns!ricardo

High School Scholarship Program
Charlotte F. Acken
Sandia National Laboratories, Div. 4853
P. O. Box 969
Livermore, California 94551
(415) 294-3248 FAX (415) 294-1539
cacken@sandia.llnl.gov

Bibliography & Textbook Projects
Waldo G. Magnuson, Jr.
Lawrence Livermore National Lab.
P. O. Box 808, L-153
Livermore, California 94550
(415) 422-9950 FAX (415) 423-5080
magnuson@icdc.llnl.gov

Workshops/Conferences (except DAC)
Paul B. Weil
Nextwave Design Automation, Inc.
5339 Prospect Road, Suite 240
San Jose, California 95129
(408) 255-6458
pweil@icdc.llnl.gov

IEEE/CS DATC Chairman
Ron Waxman
University of Virginia, Thornton Hall
Charlottesville, Virginia 22901
(804) 924-6086 FAX (804) 924-8818
waxman@virginia.edu

Outstanding Member Recognition
Charles E. Radke
IBM Corporation, AR1
Route 52
Hopewell Junction, New York 12533
(914) 892-2957 FAX (914) 892-3039

European Activities
H. Gordon Adshead
International Computers Ltd.
Wenlock Way, West Gorton
Manchester M125DR, ENGLAND
(011-44-61) 223-1301

Asian Activities
Akihiko Yamada, Chief Engineer
Information Processing Group
NEC Corporation, Daito Tamachi Bldg.
14-22 Shibaura 4-Chome, Minato-Ku
Tokyo 108, JAPAN
(011-81-3) 456-5750
NEFAX (011-81-3) 456-7495

Travel Grants
James P. Cohoon, Ph.D.
Department of Computer Science
University of Virginia, Thornton Hall
Charlottesville, Virginia 22903
(804) 982-2210 FAX (804) 982-2214
csnet: cohoon@virginia.edu
uucp: !uvacs!cohoon

Scholarships
Herschel Loomis, Jr.
Naval Postgraduate School - EE Dept.
Code 62Lm
Monterey, California 93943
(408) 646-3214
loomis@cs.nps.navy.mil

University Faculty Activities
Mary Jane Irwin
Pennsylvania State University
Department of Computer Science
University Park, Pennsylvania 16802
(814) 865-3176
mji@psuvaxl.cs.psu.edu

University Booth Project (DAC)
Martin Wong
Department of Computer Science
University of Texas
Austin, Texas 78712
(512) 471-7316
wong@cs.utexas.EDU

SIGDA OFFICERS

Chairman & ACM DAC Representative

Charles A. Shaw
Cadence Design Systems, Inc.
2455 Augustine Drive
Santa Clara, CA 95054-3082
(408) 987-5321
shaw@cadence.com

Secretary/Treasurer

Michael J. Lorenzetti
Microelec. Center of North Carolina
P.O. Box 12889
Research Triangle Park, NC 27709
(919) 248-1915
mikel@mcnc.org

Vice Chmn & IEEE DATC Nwsltr Coord

Bryan T. Preas
CADLAB
Bahnhofstrasse 32
4790 Paderborn, West Germany
011 49 52 51 28 41 02 main
011 49 52 51 60 20 67 secy
preas.@cadlab.cadlab.de
preas%cadlab.uucp@seismo.css.gov

EDITOR'S NOTES

Welcome to the summer edition of the *SIGDA Newsletter*, the DAC edition. It contains a real mix of articles from how to get the most from your trip to the DAC to thermal expansion joints (circa 1970). Also, there are summaries of the first-ever European DA conference (EDAC-90) held in Glasgow.

We also have a special feature in this edition courtesy of Chuck Radke. We have reprinted the first *SIGDA Newsletter* (Vol. 1, No. 1). Chuck has archived all the *Newsletters*, and he kindly sent me copies of the early ones.

Everyone seems to have found the new Membership Benefits section, and we are getting a large number of orders for manuals. The ordering is going smoothly, except for EDIF manuals, which have been very difficult to obtain. And of course, conference proceedings will not be available until after the conference date. Thank you for your patience. In contrast to the response for the manuals, there has been a slow response to our membership survey. We have had less than 50 responses. This is out of a membership of over 3000. I hope the tide will turn on this so we can get some meaningful data (for our email directory, etc.).

We are fortunate to have two very good articles in this edition; one on CAD Tool Interchangeability by Steve Meyer, and one on ECAD Data Integration by G. Kaufman. Both focus on CAE/CAD database and tool integration issues (areas I like to call enabling technology). Good enabling technology allows the DA user to mix and match diverse tools for their special individual attributes. Mr. Meyer takes the somewhat controversial stand that netlist translators running under the basic Unix operating system may be a sufficient framework for integration. Mr. Kaufman suggests a database approach (ultimately object oriented) for linking diverse DA applications. Of course, equal time will be given to any CFI advocate who wishes to comment.

If you have an opinion or article, send it in to the *Newsletter*. And please, send in your survey forms. **THANK YOU.**

Patrick M. Hefferan - SIGDA Editor
1681 Princeton Ave.
St. Paul, MN 55105
(612) 690-5451
hefferan@frith.entmoot.cs.psu.edu
76350.54@compuserve.com

LETTER FROM THE CHAIR

Activities are picking up in preparation for the 27th DAC in Orlando. You won't want to miss it! If you aren't convinced already that you need to go, you will be after you've read this issue of the Newsletter!

SIGDA starts out right on opening day. Our ANNUAL MEETING is SUNDAY, June 24, from 7 to 9 p.m., in Room 6 of the Convention Center. The usual beverages and groaning table of finger food will take care of any rumbles in the tummy. The reports of our many activities, and the chance to discuss them and provide your directors with input and authorization to continue, should make this another interesting, stimulating meeting. Come and give your support.

In the meeting, we'll have reports on the first EDAC European Conference, which a number of your officers and directors attended and met with their counterparts in Europe. I want to congratulate Gordon Adshead, General Chair of EDAC in Glasgow this year on a stellar conference. Hopefully he'll attend our Sunday night session - he misses very few SIGDA events - and we can give him our congratulations in person.

While you tour the exhibit floor at DAC, make it a special point to stop by the SIGDA University Booth. Martin Wong has a much expanded and even more dynamic version of Mary Jane Irwin's booth from last year. The enthusiasm and skill of these University students, as they demo their software, should make you feel that our profession will be in good hands in the coming years.

The landmark CD-ROM project will share the University Booth. See what you'll be able to do in literature search when your SIGDA project is completed - the whole world's literature on Design Automation literally at your finger tips, with easy-to-use software to help you find the references you really need. Bryan and Kathy Preas are driving the field of literature retrieval!

Our candidates for High School Scholarships will be at DAC, with their mentors. You'll get a chance to meet them Sunday night. Take time during the week to talk with them. See what a great program Charlotte Acken and Regina Pistilli have going - it'll make you proud that your SIGDA is doing such good things. Help us honor Regina, as she leaves us to go for her PhD, and welcome Donna Couch in her place.

The awards and scholarships will be announced and presented. We hope you'll agree with the selection committee that these are richly deserved.

I'll conclude with a Chairman's constant plea - **GET ACTIVE in SIGDA** if you aren't already. We have a rich variety of projects, and an open mind toward ideas for new ones. Dynamic organizations need infusions of ideas, talent and energy. The excitement you see around you in SIGDA's activities at DAC this year should inspire you to want to join in yourself!

SEE YOU AT DAC!

Charles A. Shaw
SIGDA Chairman

SIGDA ADVISORY BOARD MEETING MINUTES

The SIGDA Advisory Board held its meeting on Saturday, January 20, 1990, at the Hyatt Regency at the Dallas-Fort Worth airport. The meeting was called to order by SIGDA Chairman, Chuck Shaw at 9:30 a.m. In attendance were Charlotte Acken, Jim Cohoon, Pat Hefferan, Mary Jane Irwin, Hersch Loomis, Mike Lorenzetti, Bryan Preas, Dick Smith, Ron Waxman and Paul Weil.

Shaw presented the agenda, which was approved. Those agenda items appear as boldface headers in these minutes.

A. Approval of Minutes

Minutes from the November 4 Board Meeting were distributed by SIGDA Secretary/Treasurer, Mike Lorenzetti. There were no amendments (the minutes had been distributed earlier via email and Board comments and corrections incorporated).

MOTION: The minutes be approved as presented. *(Preas, Weil, unanimous)*

The minutes will be printed in the next issue of the Newsletter (vol 20, no. 1, February 1990).

B. Budget

Mike Lorenzetti reported that the FY '91 budget (which covers July 1990 to June 30, 1991) has been submitted. In addition to distributing copies to the Board members, he presented a summary in terms of expenses which recur annually versus those which are one time expenditures. These are summarized in the following tables:

SIGDA Recurring Expenses:

Travel Grant	125,000
Officer/Board Travel	40,000
Secretarial	10,000
Newsletter Help	10,000
Office & Printing overhead	21,400
Mailing	8,000
SIG Discretionary Fund	200,000
HQ Allocation	10,000
Undergraduate Scholarships	60,000
Conference Capital Equip.	30,000
Textbooks	25,000
Graduate Scholarships	120,000
Library Grants	20,000
Benchmarks	50,000
Other Misc. Expenses	2,300
University Booth	90,000
TOTAL	821,700

SIGDA Non-Recurring Expenses

Communications	10,000
Promotion & Advertising	20,000
Local Support	10,000
CD-ROM	350,000
Video Tape Production	100,000
TOTAL	490,000

The important point is that due to recent increases in SIGDA Board activity levels, the recurring expenses now total slightly in excess of our annual revenues. Given our current reserves (approximately $1,500,000) this is not cause for alarm, but indicates we are finally applying our reserves to meaningful ways to further the DA profession.

C. Member Address List

Pat Hefferan has a complete address, phone, FAX and email list for the Board, which he updates and publishes on the inside front cover of the Newsletter. The question was raised whether we should expand this listing service to include all members. It was pointed out that mailing labels for members are available from ACM Headquarters, should we need to reach the membership by direct mailing. After some discussion, including concerns about privacy and whether members want us to do this, the issue was tabled without action.

D. Newsletter

Hefferan reported on the success of the coupon program, wherein SIGDA members could mail in coupons printed in the Newsletter, along with a $5 fee and receive

copies of the VHDL and EDIF manuals (with SIGDA picking up the balance of the cost). To date, he has received over 60 responses, mostly for the VHDL manual. The next Newsletter (vol. 20 no. 1) goes to press in 2 weeks. Hefferan asked whether we should reprint coupons.

MOTION: Reprint coupons for EDIF and VHDL manuals in the next Newsletter issue and continue offer. *(Preas, Weil, unanimous)*

Should we expand the program to include Proceedings such as DAC or ICCAD? Jim Cohoon checked with ACM about this previously. There is no problem with Proceedings discounts (we buy from ACM at the regular price). Coupons could be an alternative to the two classes of membership proposed by Cohoon at the November Board Meeting. One problem is estimating the number of extra Proceedings to print and passing this information on to DAC in a timely manner. For DAC and ICCAD we need to ask our members in advance so we can give projections to ACM and IEEE and they can adjust the number of copies they print. DAC needs to know by mid-March. Jim Cohoon volunteered to check prices with Fred Aaronson of ACM.

MOTION: SIGDA Newsletter print coupons which allow members to order DAC and ICCAD Proceedings for coming year (1990) at a cost of $5 per Proceedings. *(Hefferan, Lorenzetti, unanimous)*

MOTION: SIGDA Newsletter print coupons which allow members to order 25 Years of Electronic Design Automation for $5 per copy. *(Preas, Cohoon, unanimous)*

Hefferan requested more articles for the Newsletter. The Newsletter seems to be a good place to print system overview articles. Such articles are of high interest to the members and are often rejected by conferences and journals as not theoretical enough. Currently, EDIF, VHDL and CFI all submit minutes and reports to our Newsletter.

The idea of publishing the tables of contents of CAD-related journals and magazines was suggested. After some discussion it was agreed that this was a good idea and we should restart it.

MOTION: Include the SIGDA Newsletter in the European Design Automation Conference (EDAC) handouts. *(Weil, Hefferan, 10,1,0)*

E. Email System

Mary Jane Irwin announced that the 800 number was installed in mid-December and Board members may now dial into one of the SIGDA University Booth machines to read news and mail. A 2400 baud modem is required and details on how to log in and use the system were distributed. This service will be down during DAC or whenever they use the machine for demos off site (usually only one other show). Announcements will be made in advance of these downtimes.

The costs for this service were $340 for installation and $45 per month plus a per-call charge for the phone line. The $1000 per year budget should be adequate.

The question of whether we should open this service to all SIGDA members was raised. Disk cost could be a problem. Also, it could end up being used for more than DA and SIGDA purposes. There was general agreement not to pursue it at this time.

The question was raised whether we can we link in the Asian representative to this system. This led into a discussion of Asian representation in general. It was pointed out that although Akihiro Yamada is an excellent contact and a hard worker, his responsibilities do not always allow him time to attend SIGDA meetings and perform other volunteer functions. Should we pursue a replacement? Ron Waxman volunteered to contact Yamada regarding both issues. He also agreed to contact Richard Newton about suggesting an Asian representative.

On the subject of Asian activities, it was suggested that we fund translation of the VHDL manual to Japanese. Hefferan reported that translation of technical material costs approximately $100/page. It was pointed out during the discussion that perhaps Japanese professional societies or DATC should pay for this. However, UHDL is a competing Japanese language so Japanese societies may not be interested. Also, if we do this for Japanese, what about German, French and all the others. The suggestion was tabled without action.

F. CD-ROM

Bryan Preas distributed a report from the contractor which included sample screens, and a description of the capabilities. Three contractors are working on the design:

Data Capture	Saztec
Service Bureau	Reference Technology
Retrieval Software	Knowledge Set

Contractors are yet to be determined for documentation, packaging and CD pressing. They are working on a prototype consisting of the 26th DAC Proceedings. Evaluators should have ROM's in March. Cooperation from societies has been good. Both ACM and IEEE-CS press are excited about it and are planning a large publicity campaign. Our cost is approximately $11 per page. Adding the Proceedings of the International Test Conference is currently being discussed. There will be six CD-ROMs total (instead of two or three, as originally estimated). All the DAC Proceedings will fit on three ROMs. ICCAD and Transactions on CAD will each be on one ROM.

Preas summarized the approximate costs as follows:

Design Phase	30,000
Prototype	35,000
Production	305,000
Press 1100 copies	15,000
Documentation	20,000
Administration	97,000
TOTAL	502,000

This works out to about $11/page, including mark-up, keying in the papers (twice) and QA of the results.

The question of whether the SIGDA Newsletter should be added was raised. Although it is not a reviewed publication like the others, it has much useful information and it is our own publication.

MOTION: Establish a budget of $25,000 for inclusion of the SIGDA Newsletter in the CD-ROM project. *(Weil, Loomis, unanimous)*

The software will run on MAC, PC, some Unix machines. It will require lots of memory and a CD-ROM reader (which cost about $700-800). Users will be able to order the ROMs from ACM and IEEE. Unit costs will be $70 for license fee plus (approximately) $150 for overhead, advertising and profit (this is currently being negotiated). Preas will report in the future on costs to members.

Kathy Preas is administrating the project. She has been paid roughly $2,000 total so far.

MOTION: SIGDA pay Kathy Preas's expenses to attend future SIGDA Board Meetings to report on the status of the Project. *(Loomis, Weil, unanimous)*

G. DAC Management Contract Status

Dick Smith reported on contract status. The process has been much more prolonged than anticipated, despite great efforts by Dan Schweikert and Smith himself. The final draft has been approved by legal counsel on both sides. It has been sent to ACM and IEEE-CS. The contract covers three years (92, 93 and 94). Subsequent contracts will be put to competitive bid.

H. University Booth 1990

Mary Jane Irwin reported on the University Booth for DAC. SIGDA has paid for space and a schedule is being formulated. Professor Martin Wong of The University of Texas is the contact person. He is sending out direct mail invitations to faculty and students who have published at DAC and ICCAD during the last two years. A demonstration of the CD-ROM will be included in the University Booth, if available.

I. University Booth 1991

Jim Cohoon will organize the University Booth for the 1991 DAC. There is nothing to report at this time.

J. High School Scholarship

Charlotte Acken reported that she has been meeting with MESA and High School counselors from the San Francisco area to set up the 1990 program.

The final selections for the 1989 program are about to be made. Mentors are reviewing the progress of their students. They also use the visit to the high school to talk to classes about DA careers. The Mentors will meet in the middle of February for final selection.

The question of how to disburse the money was raised. The winners are awarded $4000 per year. Should we work through financial department of the school? Since rules vary for different schools it was decided that this should be left up to the Mentors.

Charlotte is looking for Mentors between now and May. She feels it is important to match minorities of Mentor and student. It was suggested that she recruit a pool of Mentors and make selections once the students are chosen.

Charlotte reported that Regina Pistilli is going back to school this fall and may not be able to continue. We may be able to hire MPA staff to replace her, however, staff local to Bay Area may be better. Can MESA do it for a fee? The approximate salary range is $20 per hour (administrative assistant). The decision was left up to Charlotte.

Charlotte suggested a token of appreciation for Regina. A plaque combined with a small cash award would be appropriate.

MOTION: SIGDA award $500 plus a plaque to Regina Pistilli for her tremendous work on behalf of the High School Scholarship program. *(Loomis, Lorenzetti, unanimous)*

Charlotte volunteered to make the arrangements for the award.

K. Graduate Scholarships

Herschel Loomis reported that at the 1989 Design Automation Conference three new scholarships at $10,000 each, were awarded along with three renewals at $7000 each. This year there will be ten scholarships total (new and renewal), at $12,000 each. The Graduate Scholarship Committee consists of Loomis, Shaw, and Radke. Publicity is done by direct mail to ACM student branches, IEEE student branches, SIGDA members and DATC members. Loomis showed a sample of the flier he will mail. It was suggested that he add the list of university people who submitted papers to DAC and ICCAD which is used for University Booth announcements.

The idea of expanding the program further was raised, however, it was pointed out that $120,000 is a large portion of the budget already. It was agreed that we should leave it as is for now and that library grants can continue to be used as consolation prizes. It was suggested that Loomis send reminders to recipients for Newsletter articles describing their programs.

L. Petty Cash Fund

Lorenzetti announced that a SIGDA checking account has been opened for petty cash expenses. The maximum amount in the account is $10,000. He requested that Board members limit requests to $5000 and send larger bills to ACM as before.

M. Video Tape Plans

Mary Jane Irwin distributed a letter from University Video Communications (UVC) which included a proposal for SIGDA support of CAD video tape production. At Irwin's request, UVC interviewed a list of industry and academic experts in the CAD community regarding the need for CAD related videos. As a result of these interviews, UVC proposed the production of two tapes:

(1) Introduction to CAD, geared to sophomore/junior university curricula,

(2) First in a series of in-depth CAD tutorials, geared to the professional community, university and graduate students.

Some of the research and speaker fees contain variable amounts, but the total budget is not to exceed $96,000. Discussion centered around whether this is in keeping with our goals and a good use of our funds. Should we include a VHDL tutorial? This could also be good for educating management and non-technical people on the importance of CAD. An introduction tape that grabs the interest of a CEO would help the profession. One of the concerns of the Electronic Design Automation Companies (EDAC) is the perception of the industry. Perhaps they are willing to be partners in this project. Smith volunteered to contact EDAC to assess their interest. Tutorials from DAC could be taped at a separate site. Slides of a selected tutorial could be reformatted for video and the filming done at a site with a studio. This sort of project must be done a short time after DAC while the information from the tutorial is still fresh and interest of the speaker is high.

MOTION: SIGDA fund one introductory tape targeted at management personnel who are not familiar with CAD. *(Preas, Smith, 10, 1, 0)*

Discussion then turned to a tutorial tape. It was pointed out by Ron Waxman that DATC may be pursuing this in the near future. The issue was tabled until next meeting. Charlotte Acken volunteered to pursue the matter further and make a proposal at the next meeting.

N. SIG Discretionary Fund

Lorenzetti reviewed the ACM SIG Discretionary fund, to which ACM's 30 SIGs donate funds used to support ACM-wide interests. He announced that our contribution is worded the same as last year: "Subject to an absolute ceiling of $200,000, SIGDA will contribute an amount equal to the higher of the following two numbers: the largest contribution of any other single SIG or half the contribution of all the other SIGs combined." This is only a one-year commitment and can be changed for FY'92. Lorenzetti asked for guidelines from the Board in representing SIGDA at the meeting to decide disbursements for the Discretionary Fund in February. The consensus of the Board was that service directly to members of ACM is a higher priority than to the profession as a whole.

O. CSC Meetings

Lorenzetti, who will be representing SIGDA at the SIG Board and Conference Board meetings at ACM, asked for input on any other issues regarding DAC which are expected to come up. Smith and Shaw reported that no such issues exist in light of recent progress on the DAC management contract.

P. University Booth Equipment Purchase

Cohoon brought up the idea of equipment grants to. He reported that we can get a better discount in bulk purchases than in giving individual awards. However, individual faculty members can do better in negotiating matching funds if the grants are cash and we should not endorse a single workstation with these awards. Time is now short for this new project so we should not start until the '91 conference. Cohoon will assemble a detailed proposal, including a committee to administer the awards, and bring it to the

Board. One aspect that needs to be addressed in that proposal is the criteria for the awards (need vs. merit).

Q. SIGDA at EDAC

Preas reported on plans for publicizing SIGDA at the upcoming European Design Automation Conference (EDAC). He plans to distribute the following material: overview of ACM (material from ACM headquarters), brochures and membership forms. He will make a presentation giving an overview of SIGDA projects at one of the evening sessions.

R. Brochure

Shaw reported for Chuck Radke on the preparation of a SIGDA brochure. He distributed sample brochures to the Board and invited comments/corrections. Preas reminded the group that he needs printed brochures for the March 11 presentation at EDAC.

S. Travel Grants

Cohoon reported that he is currently spending his entire budget. $104,000 has already been expended for this year. Cohoon reviews airfares and per diem charges to insure they are reasonable. It was suggested that he hire a travel agent to help shop for the best airfares. However, this was left to his discretion, as is the monitoring of proposals and funds.

T. Proceedings Copies Response

See item D above.

U. Modern Journal (Video) Proposal

Preas discussed his idea for a "Video Journal" which features filmed presentations on VCR tape or CD-ROM rather than printed papers. It should have a shorter lead-time than current journals and would be a modern medium not previously utilized for this purpose. The rejection rate at DAC and ICCAD is very high, and there is room for other means of "publishing" good work. Furthermore, system overview papers are no longer published because they are rejected in favor of algorithm papers and this could be an outlet for such papers. The Board agreed that it is an interesting idea that we should keep in mind for later action. It was felt that we should evaluate the response to the Video Tape project (see item M above) before proceeding with this one.

V. Future Meeting Plans

Shaw announced that the General Membership Meeting at DAC will be held Sunday night after the close of the technical presentations in the convention center. There will be a Saturday morning Board Meeting at DAC as well. Informal Board discussion will take place Friday night in one of the hotel suites.

Shaw opened a discussion on whether we need to hold another Board Meeting prior to DAC. Weil proposed that we hold one at EDAC. This could help establish our image as international organization and open doors to more European cooperation. He pointed out that the cost is low for a one-week stay in Glasgow. There is a conflict with the Technical Program Meeting for the MCNC workshop on Layout Synthesis which involves two Board members (Cohoon and Lorenzetti). The discussion was tabled while one of the Board members called to check airfares.

W. Textbook Report

Since Waldo Magnuson was not able to attend the meeting, the report on textbooks was postponed until next meeting.

X. DATC Report

DATC chair, Ron Waxman, brought us up to date on DATC activities. They are holding an Executive Committee meeting January 24 at Scottsdale, Arizona. They are looking for ways to increase circulation of Design & Test Magazine. They plan to increase the number of pages that DATC Newsletter purchases from D&T.

IEEE-CS now has 106,000 members, 34% of IEEE members are CS members.

Waxman distributed copies of the February issue of the DATC Newsletter which summarizes DATC activities. A VHDL video tutorial is in the planning stages and we may wish to make this a cooperative effort with SIGDA (see item M above). The VHDL users group, sponsored by DATC, has agreed to an experiment to plan their 1990 activities as an entity of the Computer Society. DATC is working with DASS to set up a DASS meeting on VHDL maintenance by way of international teleconference between US and France.

For more information on DATC activities, see the DATC Newsletter in the February issue of Design & Test Magazine.

Y. Outstanding Member Recognition

Shaw reported for Chuck Radke that Board members should send nominations for this year's award directly to Radke.

Z. Benchmarks

Lorenzetti reported that the benchmark distribution program is continuing well within budget. Requests to MCNC for benchmarks occur at a rate of 8-10 per month, with higher rates around the time of workshops.

It was pointed out that Hal Carter is putting together a repository of VHDL benchmarks and we may want to investigate merging those into the SIGDA set.

a. Letter of Commendation from ACM

Shaw distributed copies of a letter he received from Jack Esbin, Treasurer of ACM, commending SIGDA on the scope and depth of our programs. He commended both the Executive Committee and advisory Board "for the effort and concern placed on constructive use of your resources to benefit others."

b. Workshops

Paul Weil reported a dramatic increase in requests for SIGDA sponsored workshops. He receives about two requests per month and has turned down only three of them. Two could not produce a two-page proposal and were turned down for that reason. Another was not well organized enough to be successful.

Workshops we will be sponsoring in the near future include:

* The High Level Synthesis Workshop in Germany (March 3-6, 1991),

* A Workshop on Timing Issues in British Columbia (August 1990), and

* Formal Methods in VLSI Design organized by Subrahmanyan of AT&T

A workshop on Solid Modeling is being organized by Joshua Turner of Rensselaer and has requested SIGDA co-sponsorship (along with SIGGRAPH). Weil asked if we should pursue this since all our other activities focus on Electronic Design Automation. How many SIGDA members would attend this workshop? After some discussion it was decided to refer them elsewhere (to a mechanical engineering society).

Weil reported that the workshop on Logic Level Modeling had lower than expected attendance, but broke even financially.

Weil asked the Board whether we will be holding a Physical Design Workshop this year. Preas informed him that there may not be one this year because the MCNC workshop on Layout Synthesis (for which we have "in cooperation with" status) is covering physical design this year.

There was a discussion of what happens when workshops are co-sponsored. Division of services between ACM and the other society are negotiated on a per-workshop

basis. ACM usually handles registration and other bureaucratic details. This is very helpful to volunteers who organize workshops and are not familiar with organizing such activities.

Weil asked for extra Newsletters to give to workshop organizers for distributions. Hefferan agreed to request 200 extra copies per issue. He also requested two gross of SIGDA brochures, when they become available.

c. Bylaws

Lorenzetti reported that the bylaw changes have been sent to ACM for approval and we are awaiting their response.

d. Bulletin Board

Smith reported that comp.lsi.cad works well as a bulletin Board medium, but is not heavily used. He asked for ideas on how we can publicize it further. Do our members have access to the network? Smith suggested that we run a survey in the Newsletter to determine who has access. The same survey can ask for email addresses and other information to help us determine the needs of our constituents. Hefferan suggested that we offer some prize or some other inducement to encourage return of the survey, such as a pocket electronic diary. We could either send a small award (such as a SIGDA pen) to all respondents or have a drawing among those who return it for a larger award.

MOTION: SIGDA Newsletter publish a survey with a prize awarded by drawing among those who participate. The prize is to be determined at the Newsletter Editor's discretion, but should not exceed a cost of $500. *(Loomis, Smith unanimous)*

e. Publicity

John Acken has ordered pencils with "SIGDA" printed on them to be distributed at workshops and conferences sponsored by us.

f. ACM Regional Representative Candidates

Shaw reminded the Board that, although the ACM council has been reorganized to give more seats to SIGs and less to regions, it is a phased transition and the Nominating Committee is looking for nominations. To place a name in nomination requires a petition signed by some percentage of ACM members. For the North Central Region, 47 signatures are required. For the Pacific, 91 are needed. He asked if SIGDA should consider nominating someone to these posts. No action was taken on this issue.

g. New Projects, Suggestions

Cohoon suggested pursuing production of a tutorial or interactive paper. Such a system would run on a PC, or on an X-based system, and allow the user to walk through the information, skipping over parts he was not interested in and viewing very detailed information for parts of high interest. Computer-aided instruction tools currently exist to do this. The CMU Tutor is such a system, another is available from a firm in Minnesota. These systems come with an authoring system.

MOTION: Budget $5,000 to look into computer-aided instruction kits as a means for publishing CAD papers or tutorials. The money is to be used to get copy of the necessary software. *(Waxman, Preas, unanimous)*

Preas encouraged project leaders to write a one-two page article at least once per year describing their activities. Smith added that these should also be posted to news.

Lorenzetti requested that money be allocated to allow the Secretary/Treasurer to purchase a portable computer for taking minutes at these meetings, keeping electronic copies of bylaws and other SIGDA functions.

MOTION: Budget up to $5,000 for purchase of a portable computer for use by the SIGDA

Secretary/Treasurer. *(Weil, Hefferan, unanimous)*

h. Next Meeting of the Advisory Board

It was reported that a seven day stay in Glasgow is needed to reduce airfare from $1400 to $600. SIGDA could pay for the flight and three nights lodging. A strong presence at EDAC would help to solidify our European ties. ACM is unknown in Europe and we need to make a splash. An information booth at EDAC would help. We could have an Advisory Board Meeting there on Sunday before the conference or just send a subcommittee and hold the meeting at a later date in the US.

MOTION: SIGDA send subcommittee (appointed by the Chairman) to EDAC, and hold the Advisory Board Meeting at a later date in the US. *(Lorenzetti, Smith, 10-one-0)*

MOTION: To adjourn. *(Cohoon, Loomis, unanimous)*

Respectfully submitted,

Michael J. Lorenzetti
SIGDA Secretary/Treasurer

After the meeting adjourned, it was suggested that a budget of $5,000 be established to purchase a portable computer for the Newsletter Editor. All present (Shaw, Weil, Waxman, Acken, Hefferan, Loomis, Lorenzetti) agreed.

MEMBERSHIP BENEFITS

As a special benefit to SIGDA members, certain standards manuals, proceedings, and other documents of general interest will be made available at a nominal rate. Availability of the documents may vary, so allow several weeks for delivery. Proceedings will only be issued after the conference sales have been satisfied. ***You must be a member of SIGDA to quality.***

MEMBERSHIP COUPON - DOC.

-SIGDA Membership Sponsored Manual/Proceedings -

Name _____

Address _____

_____

_____

_____

ACM Membership Number _____

SELECTIONS (please check box(es))

- ☐ EDIF 2.0.0 Standards Manual
- ☐ VHDL Language Reference Manual
- ☐ 25 Years of Electronic Design Automation
- ☐ 1990 DAC Proceedings
- ☐ 1990 ICCAD Proceedings

Please include a check for $5.00 (made out to ACM/SIGDA) to cover the cost of shipping. ($5.00 per document). Please allow several weeks for shipping and handling. DAC & ICCAD Proceedings mailed only after conference is over. Also, allow extra time for EIA EDIF Publications.

Please send coupon to:
Patrick M. Hefferan
1681 Princeton Avenue
St. Paul, MN 55105

SIGDA MEMBERSHIP SURVEY

In order to better serve the SIGDA membership, we are taking a survey of your wants and needs. Please take a few minutes to fill out the survey form on the next page and send it in. It has a preprinted address on the back, so just cut it out and fold it up with the address facing outward. Then tape (please do not staple) it shut and mail it in. You will need to provide 25 cents postage, as we cannot use our bulk mailing permit for this.

To make this more interesting and provide an incentive to get the surveys filled out, those who participate will be eligible for a drawing for a portable, electronic address book. We want to hear from you - so take a few minutes and fill it out. THANK YOU!!

PLEASE FILL OUT AND SEND IN ---------->

THE WINNER OF THE ELECTRONIC ADDRESS BOOK THIS QUARTER IS DAVID J. ELLIS FROM LONDONDERRY, NEW MEXICO!! THERE WILL BE ANOTHER WINNER ANNOUNCED IN THE NEXT SIGDA NEWSLETTER.

SIGDA SURVEY
January, 1990

This is a general membership survey.
Participants will be entered in a drawing for an electronic address book.

Name _____ EMAIL Address _____
(May we publish this address? _____)

Address _____ Company _____

_____ Occupation _____

_____ Job Function _____

Phone - Home _____ Education _____
(Highest attained)

Work _____

_____
(Signature)

How long have you worked as a DA professional? _____

Computers used in work _____

Do you have a PC at home? _____ If yes, what kind _____

Do you have a modem? _____ *At home?* _____ *At work?* _____ *Speed?* _____

Are you connected to a network (Arpa, CIC, INTERNET?) _____

In what professional societies are you a member? _____

_____

What CAD/CAE functions are you responsible for?

_____ Schematic capture	_____ Simulation	_____ Layout
_____ DRC	_____ Test	_____ Management
_____ Design	_____ Specification	_____ Evaluation
_____ Mechanical	_____ Other _____	

What additional benefits would you like from SIGDA?

_____ DAC Proceedings	_____ DAC/ICCAD Admissions
_____ ICCAD Proceedings	_____ Network Access for EMAIL/BBS
_____ Standards Manuals	_____ Continuing Education _____
_____ Tutorial Tapes	_____ Other _____
_____ CD-ROM Tapes	

Would you pay extra for them? _____ *Yes* _____ *No*

The SIGDA High School Student Mentor Program is designed to increase under-represented groups in the DA profession. The target groups for this program are: Hispanic, Black, Women (all races), Disabled, and American Indian. We are requesting that the Mentors be from the target groups. Also, for administrative purposes, the Mentors must be from the San Francisco Bay Area. For further details, contact Charlotte Acken (address on front cover).

Would you like to be a Mentor? _____ *Yes* _____ *No*

Of which target group are you a member? _____

put 25 cent
stamp here

SIGDA Membership Survey
c/o Patrick M. Hefferan
1681 Princeton Avenue
Saint Paul, Minnesota 55105

UNIVERSITY OF VIRGINIA
CHARLOTTESVILLE

OFFICE OF THE PRESIDENT

March 5, 1990

Mr. Charles A. Shaw
Cadence Design Systems, Inc.
2455 Augustine Drive
Santa Clara, California 95054-3082

Dear Mr. Shaw:

It was a pleasure to learn of SIGDA's recent grant to the University of Virginia for computer hardware to be used at the Design Automation Conference next year in San Francisco. Our computer scientists find this support deeply gratifying indeed.

The University must rely increasingly on private support to meet many of its special funding needs. The workstations and associated equipment which you will make available to us both before and after the conference will be of great assistance to our researchers as they develop innovative software packages. We are exceedingly appreciative of SIGDA's continued generosity, and we hope that our information processing program continues to merit your confidence.

Very sincerely,

Robert M. O'Neil
President

RMO:slb

MADISON HALL BOX 9011 22906-9011 TELEPHONE 804-924-3337

acm

the society for computing and information processing

March 20, 1990

Ron Oliver (SIG Board Chair)
Dept. of CS
CAL POLY
San Louis Obispo, CA 93407

Stewart Zweben (Chairman, Constitutions & Bylaws Committee)
Dept. of Computer & Information Science
Ohio State University
2036 Neil Avenue Mall
Columbus, OH 43210

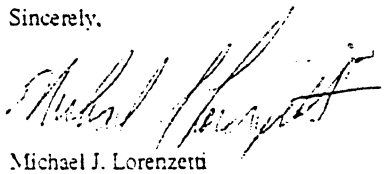
Dear Sirs:

Enclosed for your approval are proposed changes to the SIGDA Bylaws. I have included both a new draft and an annotated copy of the current bylaws, so you can see where the changes were made. The changes were precipitated by the recent changes to ACM Bylaw 7. The SIGDA Advisory Board has elected to make some additional (minor) changes during the revision process. Namely:

- A more precise formalization for the approval of large expenditures (Article 5.b.iii).
- Increase the suggested size of the Advisory Board to better fit our current operating mode and the magnitude of our ongoing projects (Article 8a). It was also felt that "Board of Directors" was a misnomer for the members-at-large and it was deleted.
- Change the suggested location of the annual membership meeting to the Design Automation Conference, in keeping with current SIGDA practice.
- Minor changes as suggested by Stu Zweben of the Constitutions and Bylaws (a copy of his suggestions is attached)

These changes have been approved by the SIGDA Advisory Board at their March 12th meeting.

Sincerely,



Michael J. Lorenzetti
SIGDA Secretary/Treasurer

MJL/myd

cc: Fred Aaronson, (Manager, SIG Activities)
Pat Hefferan, (SIGDA newsletter editor)
Bryan Preas, (SIGDA vice-chairman)
Charles Shaw, (SIGDA chairman)

Enclosures

ACM Press Database and Electronic Products

A Publication Series of the Association for Computing Machinery • 11 West 42nd St., NY, NY 10036, TELEX 421686
ACM HQ Liason, Bernard Rous, Assoc. Dir. of Pub., 212/869-7440, FAX 212/944-1318, rous-cr@acmvm.bitnet
Editor-in-Chief, Edward A. Fox, Dept. of Computer Science, VPI&SU, Blacksburg VA 24061-0106, 703/231-5113
Internet: fox@vtopus.cs.vt.edu, BITNET: foxea@vtcc1, FAX 703/231-7826
Hypertext Editor, Gary Marchionini, 4121C Hornbake, CLIS, Univ. Maryland, College Park MD 20742, 301/454-3235
Multimedia Editor, Scott M. Stevens, Software Eng. Inst., Carnegie-Mellon Univ., Pittsburgh PA 15213, 412/268-7700

Memorandum

To: All SIG Chairs and SIG Newsletter Editors
Date: December 20, 1989
Subject: ACM CSC Meeting, Tuesday 10-12,* about Electronic Submissions
(tentatively, in Arlington Room at the Sheraton Washington Hotel)
From: Edward A. Fox, Editor ACM Press Database and Electronic Products

You are invited to a meeting to hear about and help with planning for ACM allowing electronic submissions of documents sent in for publication. This topic is especially important as we move toward making articles available for hypertext browsing or computer-aided searching. It can also help with re-use of articles, which might be prepared for a conference, reprinted in a Transaction, and also later included in an ACM Press Books manuscript.

This topic relates to SIG methods for: accepting submissions to newsletters, handling of accepted papers for conference proceedings, and possibly also to receiving and distributing notes for tutorials. Therefore we ask that you and/or others in your SIG attend this session if at all possible.

The meeting will include presentations on national and international standards for document markup, discussion of difficult problems dealing with graphics and compatibility between various formatters and WYSIWYG editors, and issues relating to media or network transmission. We hope that you also will consider helping us involve volunteers so that authors and editors can have versatile tools to make electronic submission and subsequent processing be feasible and perhaps even somewhat convenient.

My current suggestion is that we have volunteers prepare "Author Kits" so that someone using troff or LaTex or GML Script or Microsoft Word, etc. will be able to follow a set of guidelines, using some set of macros or electronic style sheets that we provide. Then they can work in their own environment and yet allow us to map their article into the "ACM Internal Document Standard Form" which I believe should be based on SGML (Standard Generalized Markup Language -- ISO 8879-1986) and its application in ANSI/NISO Z39.59-1988, the American National Standard for Electronic Manuscript Preparation and Markup. Meanwhile, I look forward to alternative or additional suggestions, and to discussing realistic plans with you in February!

* 2/20/90

DEPARTMENTS

The following pages contain updates and information about various projects and activities funded by the SIGDA. Contact information (email, phone, and addresses) for the individuals directing the programs is available from the front inside cover.

European Activities ..... H. Gordon Adshead

DATE: March, 1990

FROM: Gordon Adshead

SUBJECT: EDAC '90

EDAC '90 is apparently being enthusiastically received. The EDAC committee currently plans EDAC91 on February 25-28 in Amsterdam and EDAC92 in February/March in Brussels. We plan to promote a good series of high quality technical forums for the DA community moving steadily around Europe. As such we respectfully request the support and involvement of DATC, SIGDA, and the DAC Committee.

An IFIP 10.5 group headed by Gerry Musgrave has some interesting suggestions to extend the scope, size, time and place of a possible "Mega-event" in 1992. Some of the ideas and suggestions make good sense and some seem not. I would like to stress that at this point in time there is unfortunately no agreed position of the European DA Community and most of the EDAC committee demand a lot more discussion.

The current plan is to meet constructively at EDAC in Glasgow on Wednesday, March 14, agree on a policy and get an advance EDAC '92 committee moving.

I look forward to a healthy co-operation between DAC and EDAC and welcome your comments.

DATE: April 2, 1990

FROM: Gordon Adshead

SUBJECT: EDAC - DAC

It was very good to see SIGDA in force at EDAC in Glasgow. Many of us felt that we have at last achieved the basis of a good technical, relevant DA event in Europe - 450 paying attendees and 520 people involved totally!

EDAC '91 will be in Amsterdam in February, and we are pleased to have the cooperation of DATC and SIGDA! **GREAT!!**

The real problem is EDAC '92. Everybody has ideas. Everybody wants a piece of the action. There was a lot of misunderstanding created when TEN of YOU (our friends from the U.S.) turned up at a meeting in Glasgow and gave some of us the impression that the U.S. DAC Committee was in some sense trying to exert influence or even take over EDAC. I know this is not the case, but the real problem is that there are still

some European views to be reconciled before we are clear on exactly what sort of help and involvement we would like from the U.S. DAC.

I cannot, of course, speak for "Europe" - this is impossible.

However, I would like to make it absolutely clear what the position of the EDAC Association is regarding EDAC '92:

1. Significant high quality technical international DA event.
2. Held in Europe but fitting the world context.
3. Best possible harmonious relationship with the U.S. DAC.
4. Target March 1992 in Brussels or Paris.
5. 1,000 to 1,500 attendees.
6. Significant exhibition with 100 vendors.
7. A real DA Conference.

I would greatly appreciate any views, comments or suggestions from SIGDA Board Members on how we can best set out to achieve the best possible relationship between EDAC and DAC in a way that makes real sense when viewed from both sides of the Atlantic.

Workshops/Conferences.....Paul B. Weil

SIGDA sponsors many Workshops and Conferences in the Electronic CAE/CAD arena. During 1989 and 1990, SIGDA will either sponsor, co-sponsor or work in cooperation with other professional societies on a dozen professional meetings, including the Design Automation Conference (DAC), International Conference on CAD (ICCAD), Logic Level Modeling of ASICs WS, Formal Methods WS, Physical Design WS on Model Generation, High Level Synthesis WS, Timing Issues WS, International WS on Logic Synthesis, C Hardware Description Lang & Applications WS, International WS on Layout Synthesis.

We are always developing new Professional Activities to meet the needs of our 2,000+ membership. SIGDA does its utmost to assist the organizing committees through comprehensive support activities. For the smaller meetings, SIGDA funds ACM and others who provide direct assistance with site selection, budgeting, advertisement, and registration. This leaves the organizers with time to develop a superior technical program.

If you would like more information about the SIGDA Workshop and Conference support activities, E-mail or write the Workshop/Conference Chair (see inside the front cover of this Newsletter).

DATE: February 8, 1990

FROM: Paul Weil

SUBJECT: SIGDA Workshops & Conferences

SIGDA SPONSORED CONFERENCES SYMPOSIUM AND WORKSHOPS APPROVAL PROCESS

1. **PROPOSAL:** Two page summary of:
   The Meeting's Technical Objectives
   Background of this and similar meetings on this Subject
   Potential Locations and Dates
   Meeting Size and Format
   Organizing Committee and brief background of Chairs

2. **INITIAL ORGANIZATION:** ACM will assist the Committee with:
   Site Selection
   Budgets
   Timeline of Activities leading up to the Meeting

3. **MEETING BETWEEN ACM/SIGDA REPRESENTATIVE AND CHAIRPERSON(s)**

4. **FORMAL APPROVAL BY SIGDA**

5. **SIGDA WILL ARRANGE FOR ACM TO PROVIDE ASSISTANCE IN:**
   Local Arrangements
   Call of Papers/Participation
   Registration
   Advanced Program
   Publications
   On-site Registration
   Post Conference Accounting

ACM/SIGDA encourages the organizers to consider in-cooperation or co-sponsorship with other Professional Societies. (ACM/SIGDA has a close working relationship with the IEEE as well as other Electronic Engineering Societies.)

PROFESSIONAL ACTIVITIES DEVELOPMENT

1. **PROVIDE ATTENDEES** with copies of the latest SIGDA Newsletter and other handouts including ACM membership information

2. **SEND SIGDA** four copies of any material that is generally published for the attendees. If any portion of the material is not for public release, please indicate that on appropriate section of the material.

AFTER THE MEETING:

1. **IMMEDIATELY FOLLOWING THE CONFERENCE/SYMPOSIUM/WORKSHOP** hold a meeting that will discuss future events and seek volunteers to run these events.

2. **STATUS REPORT** - Within a few days of the meeting transmit to SIGDA a brief status report including the meeting's attendance, financials, how the meeting met its technical objectives and any outstanding problems that may have developed.

3. **MEETING SUMMARY** - Within a few weeks of the meeting provide a summary (1 to 5 pages) of the meeting's activities for publication in the SIGDA Newsletter along with plans for future events. Emphasize the technical aspects of the CONFERENCE/SYMPOSIUM/WORKSHOP.

Asian Activities .....Akihiko Yamada

REPORT OF ASIAN ACTIVITIES
March 7, 1990

1. DAC SUBCOMMITTEE

DAC Subcommittee, chaired by T. Kozawa of Hitachi, has been organized for the first time following the recommendation of the DAC Executive Committee. As a result, the acceptance rate of Japanese papers has greatly improved - from 16% (5 papers) in 1989 to 56% (13 papers) in 1990. The Subcommittee is supported by SIGDA of IPSJ (Information Processing Society of Japan).

2. SIGDA OF IPSJ

SIGDA of IPSJ has about 400 members. It holds small workshops every two or three months, mostly one-day sessions. The SIGDA Committee, chaired by K. Hirakawa of Oki Electric, consists of 32 members from industries and universities. It also sponsors a DA Workshop every summer with about 100 attendees. We sometimes invite speakers from the States for panel discussions. This year, it will be held from August 30 to September 1 in Hakone, Kanagawa.

3. AFFILIATE RELATIONSHIP BETWEEN SIGDA OF ACM AND SIGDA OF IPSJ

I am proposing to IPSJ SIG Board to establish an affiliate relationship between SIGDA of ACM and SIGDA of IPSJ. The joint membership agreement between ACM and IPSJ has been established for several years, and they are sister societies. The DAC Subcommittee is supported by the SIGDA of IPSJ. It will be useful to have an affiliate relationship between the two SIGDAs to promote further cooperation.

4. EFIP WORKSHOP ON DESIGN & TEST OF ASICS

This workshop, chaired by Professor Kozo Kinoshita of Osaka University, will be held in Hiroshima, Japan, June 11-12, 1990. It is sponsored by IFIP WG10.5 and IPSJ. Twenty-seven abstracts have been

accepted: five from the United States and Canada, eleven from Europe, and twelve from Japan. Seventy participants are expected. The topics will include core microcomputer design methodology.

5. INFOJAPAN '90

IPSJ holds InfoJapan '90 international conference on information technology on October 1-5, 1990, in Tokyo, commemorating its 30th anniversary. Steven Jobs will be invited as a keynote speaker. They are expecting 1,200 attendees.

6. ADEE JAPAN '90

In January of this year, ADEE Japan '90, sponsored by Cahners Exposition Japan, was held in Tokyo. Forty-five United States and Japanese EDA vendors exhibited their latest products. There were 14,000 attendees.

CD-ROM Project ..... Bryan T. Preas, Director

DATE: April 4, 1990

FROM: Kathy Preas

SUBJECT: Report to Board of Directors
March 30, 1990

Although we just saw many of you at EDAC and presented a report on the CD-ROM Project at that time, we believe that it is important to keep you informed of the progress of this project. And these days, progress occurs daily. The project requires a lot of time from both of us now.

1. PROTOTYPE PROJECT

Data capture is complete for the prototype (the 26th DAC). Our contractors are working well together, and we expect that the Sparc and Mac versions of the prototype will be available for evaluation around the end of April. The Windows version of the retrieval engine is not complete, which will delay prototype evaluation on the IBM. Happily for the prototype evaluation schedule, we have a significant number of evaluators who are working on either Mac or Sparc machines, so we should be able to get enough feedback based on those versions for progress to continue on the final design, even as we await the IBM version. The IBM version will have the same feel, but a slightly different look. (Although KRS has not released the Windows version, they are using our data to demonstrate it at trade shows.) We are confident that we can obtain a good evaluation of the project from the prototype versions for Mac and Sparc stations.

The rapidly approaching date for testing of our prototype means that we must order the evaluation CD-ROM drives immediately. We have gotten very few returned information sheets and evaluation agreements, but Bryan has been personally contacting evaluators to get their system requirements so the evaluation will not be unduly delayed.

IEEE CS and ACM are anxious to evaluate the product in preparation for marketing, sales and distribution. Gene Falken (IEEE CS) will get the editors of various IEEE magazines to review it (in both

prototype and final form), so we are hopeful of some good publicity. Mark Mandelbaum and Bernard Rouse of ACM are also going to act as evaluators.

We will have our prototype on Jim Cohoon's SparcStation at the University Booth at DAC for demos. Thanks, Jim!

The names of those who have agreed orally to act as evaluators, and their machines, are:

Jim Cohoon .....Sparc
Jonathan Rose .....Sparc
Chuck Shaw, Cadence.....Sparc
Dick Smith, National.....Sparc
Bryan Preas .....Sparc

John Acken.....Mac II
Mark Mandelbaum .....Mac II
(ACM publications)
Gene Falken .....Mac II
(IEEE CS publications)

Gordon Adshead .....IBM
Univ. of Paderborn/CADLAB .....IBM
Kathy Preas .....IBM
Pat Hefferan.....IBM

Mike Lorenzetti.....Mac II
Rhonda Gaede.....Mac II, Mac+

2. ACM/IEEE

Bryan spent a day at ACM headquarters talking with Mark Mandelbaum, and has had a long telephone conversation with Gene Falken of IEEE CS. Both organizations have "agreed to agree" regarding their positions on the CD-ROM Project. Their enthusiasm for the project is most welcome and encouraging to us - they genuinely seem interested in becoming involved and working to the benefit of the Project. Several issues which have been agreed to in principle relate to marketing, advertising, packaging and cost of the CD-ROMs. It is important that the CD-ROMs, as distributed, look the same to the end user, no matter from which Society they are purchased. Similarly, an identical price structure is necessary. Both ACM & IEEE agree that pricing should not be based on the value of the information; that would make the ROMs so expensive that no one would buy them. Rather, a reasonable price is called for, so as to encourage widest possible distribution of the ROMs. Pricing currently under discussion is $300 for members and $600 for non-members. Perhaps we should have a price break for SIGDA/DATC members ($300 for SIGDA/DATC, $450 for ACM/IEEE, and $600 for others).

Both ACM and IEEE will have machines running the prototype at DAC, as well as advertising, flyers, preliminary order forms, etc.

3. DOCUMENTATION

ACM is actively working on the problems of production, advertising, packaging, and distribution of the CD-ROMs. Bryan and Mark have agreed that ACM will take the lead in providing packaging for the product. Kathy will be responsible for writing the documentation and ACM will supply a copy editor to assist her. Kathy will submit camera ready copy to ACM. The documentation will be on paper, as well as being available on each of the ROMs. Kathy will prepare the documentation for data base build (SGML tags). ACM will handle the printing, collating, packaging and shrink wrap.

4. PRODUCTION

ACM wants to do several other CD-ROM projects, using this one as a model. They want to establish relationships with disk-pressing houses, and have asked Bryan and Kathy to evaluate the pressing facilities. We will begin this task shortly.

ACM is currently working on the question of how many copies to press. SIGDA should consider the number of copies that we would like to have to distribute for: evaluators, library grants, rewards for those who participated in the project, distribution to magazine editors for review, etc.

5. DISTRIBUTION & USER SUPPORT

ACM and IEEE will stock the CD-ROMs just as they do printed material. They have tentatively decided to have our contractor, Reference Technology, provide user support. Bryan and Kathy concur in this, at least in the beginning. As a topic of discussion for the SIGDA Board, how does this impact our consideration of building up a position at ACM headquarters for our use? Possibly support for this project could be included in that position.

Support can be a big variable in the cost of the product. The CD-ROM Project may have to fund support, at least partially.

6. ON-LINE SERVICES

The SIGDA Board has discussed putting the CD-ROM databases on-line, but the discussion was tabled because of concerns about the impact of such a service on ACM income. It turns out that ACM has an ongoing relationship with three on-line services, STN, Dialog and FIZ Karlsruhe, and earns a royalty from on-line use of its databases. ACM has asked Bryan and Kathy to talk with these services about how our CD-ROM database might be used. Two of these services are based in Germany, so we will be talking with them over the next few months.

7. NEW MATERIAL FOR ROMS

At the Dallas meeting Bryan was authorized to include the SIGDA Newsletter on the CD-ROMs. Many thanks to Chuck Radke, who has collected all of the back issues for us.

We also propose to include the European Design Automation Conference Proceedings on the CD-ROM, so as not to isolate our European friends and colleagues.

We can include the 27th DAC on the first ROM set instead of the update ROM. We assume everyone wants us to do so.

We have direction to include the SIGDA Newsletter from the Dallas meeting. We are proceeding under the assumption that we should include EDAC and the 27th DAC. Like the Newsletter, there will be a cost increase in the project proportional to the page count, as discussed at Dallas. We still don't know the exact figures, so we don't propose to update the budget at this time. At the close of the prototype evaluation we will get hard figures from our contractors, based on all of the information available. At that time the budget can be updated.

8. FOLLOW-ON PROJECTS

We have discussed providing advanced retrieval capabilities beyond those possible with our retrieval engine by using KRS' Open Retrieval Interface. Examples are cite checking, forward and backward referencing, perhaps a key word index if we find that word searching is not powerful enough. (ACM has a program to extract key words from text and would like to be a part of this project.) The potential exists to fund university students to do part of this work. The new capabilities would be put on the first yearly update disk. We need to get Board member reaction now because we have the most leverage for negotiation for the Open Retrieval Interface licenses now.

We have found a company that can put digital data (pictures and text), as well as audio on the CD-ROMs. We installed a demo on the SIGDA CD-ROM Project computer -- very nice. This seems ideal for courses for undergraduates. Graphics can show slide material, algorithm operation, audio could be lecture/tutorial (maybe even some 'heavy metal' to keep the interest level up!)

9. CURRENT HOT TOPICS

We are, at this very moment, working on the following:

* Completion of prototype design, ordering drives, instructions for evaluators, making sure the prototype will work.
* Royalty agreement with KnowlegeSet: agreed orally, next need a written version; we are working with the ACM lawyer on this.
* Production contracts
* Prototype evaluation program
* Documentation

10. PROBLEM AREAS

We still have no authorization to include the Transactions on CAD.

The FAX machine at CADLAB is unreliable. This is causing delays in communication.

********ACTION ITEMS********

1. *The Board must decide how many copies of the CD-ROMs are wanted for library grants, Board members, evaluators, etc.*
2. *Consider position at ACM for support, SIGDA business.*
3. *Budget increase proportional to page count for EDAC, 27th DAC: speak up if this is a problem.*
4. *Any input regarding pricing of the CD-ROMs?*
5. *Reaction to follow-on projects (#8 above):*
   A. *Only the first part is critical. We need to know how hard to push for Open Retrieval Interface?*
   B. *The Multimedia ROM is not critical to anything we are doing now.*

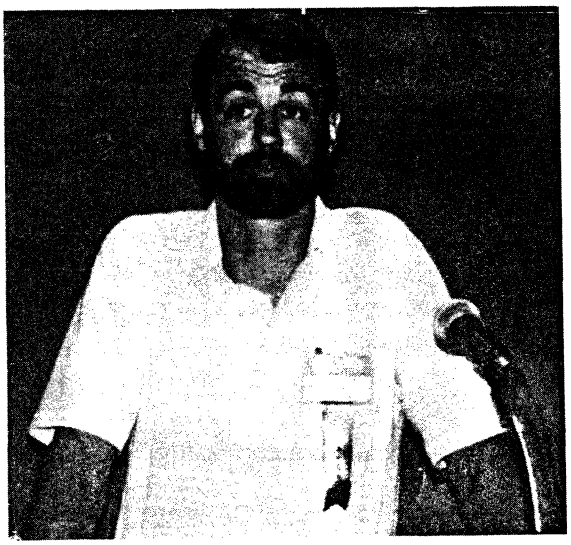
Just something to think about.

Board of Advisors..... Michael J. Lorenzetti, Secretary/Treasurer

DATE: March, 1990

FROM: Michael J. Lorenzetti

SUBJECT: SIGDA Bylaws



I received the following from Stu Zweben, Chairman, Constitution and Bylaws Committee, ACM regarding the SIGDA Bylaws. The SIGDA Advisory Board has been having an email discussion of the bylaw suggestions that were made. There is general agreement that the suggestions are excellent and serve to improve our bylaws. I will enter these changes and have them approved at the Advisory Board Meeting. The Constitution and Bylaws Committee will then be prepared to state that our proposed bylaws are consistent with the ACM Constitution and Bylaws, and that should clear the way for approval by the ACM Executive Committee.

The Constitution and Bylaws Committee has met to discuss the proposed SIGDA bylaws changes. We have the following comments for your consideration.

Inconsistencies:

1. Article 3. The dissolution of a SIG can take place in a manner other than Council vote (per Bylaw 7). Therefore, we suggest that the phrase "by the Council of the ACM" be deleted from Article 3.

2. Article 6. Section b should state that the "SIG Board Chairman" rather than the "SIG Board" fills vacancies if there's no other provision in the SIGDA bylaws.

3. Article 11. There are references to both a "nominating" committee and an "election" committee, when it appears that only one committee is in fact intended. We suggest that the references to "an election" committee in section a, and to "the election" committee in section c, be changed to "a nominating" and "the nominating" committee, respectively.

4. Article 11. In section a, there is no election of ANY officers if the SIG Board decides to exercise its option to extend the terms of office, according to PPG 17.4.2. Therefore, the middle sentence should be changed to something like "This committee will nominate at least two candidates, who consent to serve, for each elective office."

5. Article 12. In section a, there is reference to approval by the SIGDA "Advisory Board" before an amendment of section 1 can go to the SIG Board. However, ACM Bylaw 7 states that the SIG's "Executive Committee" must approve of such an amendment. So we suggest that you change the words "Advisory Committee" to "Executive Committee" in section a (alternatively, you can have both approvals, but you must require that the Executive Committee be in the loop). Also in Article 12, section b, you mention the "Advisory Board" rather than the Executive Committee, in approving amendments to

go to the SIG's membership. This is not in conflict with ACM's Bylaws, but is this what you want, given the change you must make to section a?

6. Use of the word "Chair." The Constitution and Bylaws Committee takes the position that references to ACM positions should be made consistently throughout ACM's documents. ACM refers to its positions using the title "Chairman" and recognizes this use as a non-gender-specific use (you may be interested to know that several years ago Council "cleaned up" its Bylaws to remove the use of gender-specific pronouns, but deliberately chose to keep the word "Chairman" and not use "Chairperson" or "Chair" instead). While we recognize that your executive committee takes a different position, we are still recommending the use of Chairman wherever the word "Chair" appears in the bylaws. Perhaps we will need to let the ACM Executive Committee decide this issue.

Clarifications and wording suggestions:

1. There are several examples of the use of future tense in the SIGDA bylaws, where present tense would seem more appropriate. Examples are in Articles 1a, 2a, 2b, 5b, and 8a where "will" or "will be" can be changed. This is, of course, a minor point.

2. The use of the phrase "executive committee" should be consistently capitalized or consistently non-capitalized in the document. For example, it is not capitalized in Article 4bi (where we also recommend inserting the modifier "SIGDA" in front of executive committee, since the Executive Committee hasn't yet been defined), Article 7a and Article 8c.

3. Use of gender-specific pronouns. In Article 6a, the phrase "offices he has appointed" would be better phrased "positions appointed by the Chair(man)" since the Chairman doesn't appoint offices (as SIGDA has defined its officers) and this would avoid the use of "he (or she)." In Article 8, sections a and b, the word "his" should be "his or her."

4. Avoid the use of the word "Group" since it wasn't defined in section 1. Instead, use "SIGDA." Relevant places are Article 8b and Article 14.

5. In Article 9, preface the second sentence with "As provided in ACM Bylaw 7," for clarification, since the rest of the first sentence (until the semicolon is verbatim from Bylaw 7); however, the word "voters" should be changed to "votes" to be consistent with Bylaw 7's wording.

6. In Article 11, section a, do you really want to state that the report of the nominating committee must be presented "in the SIGDA newsletter"? Perhaps the timing of the publication of the newsletter is such that you can't get it out in order to meet this provision. You might instead want to do a special mailing to all members of SIGDA to announce candidates. To allow such flexibility, the phrase "in the SIGDA newsletter" can be changed to "to the SIGDA membership". However, this is entirely your call.

7. There are capitalization omissions that, in the final writing, should probably be fixed. Examples are Article 4dii, Article 10 (second sentence), and Article 12c (last sentence). Also, readability would be improved if commas were inserted in i) Article 9b, last sentence, after the word "SIGDA"; ii) Article 14, first sentence, after the word "year".

On the following pages are the Bylaws of the Special Interest Group on Design Automation of the Association for Computing Machinery, Inc. These were adopted on October 27, 1979 and revised on October 5, 1990.

BYLAWS
of the
Special Interest Group
on
DESIGN AUTOMATION
of the
Association for Computing Machinery, Inc.

Adopted 10-27-79
Revised 03-5-90

Article 1. Name and Scope

(a) This organization is called the Special Interest Group on Design Automation (SIGDA) of the Association for Computing Machinery, Inc: (the "ACM").

(b) The scope of SIGDA's specialty is to enhance the utility of computers as engineering tools in the design, fabrication, and test of equipment systems and structures.

Article 2. Purpose

(a) SIGDA is organized and operated exclusively for educational, scientific, and technical purposes in design automation.

(b) The purpose of SIGDA and its activities includes:

(i) Collecting and disseminating information in design automation through a newsletter and other publications;

(ii) Organizing sessions at conferences of the ACM;

(iii) Sponsoring conferences, symposia, and workshops;

(iv) Organizing projects and working groups for education, research, and development;

(v) Serving as a source of technical information for the Council and subunits of the ACM; and

(vi) Representing the opinions and expertise of the membership on matters of technical interest to SIGDA or ACM.

Article 3. Charter

SIGDA will exist until dissolved as provided in Bylaw 7 of the ACM.

Article 4. Officers

(a) SIGDA officers are the Chairman, the Vice-Chairman, and the Secretary-Treasurer. The officers are elected for two-year terms beginning July 1 of odd-numbered years.

(b) The Chairman is the principal officer, being responsible for leading SIGDA and managing its activities. The duties of the Chairman are:

(i) Calling and presiding at SIGDA Executive Committee and business meetings;

(ii) Conducting all of SIGDA's activities in accordance with the policies of the ACM; and

(iii) Making all appointments as authorized herein.

(c) The duties of the Vice-Chairman are:

(i) Assisting the Chairman in leading and managing SIGDA; and

(ii) Presiding at meetings when the Chairman is absent.

(d) The duties of the Secretary-Treasurer are:

(i) Maintaining the records and correspondence of SIGDA;

(ii) Keeping and distributing the minutes of business and Executive Committee meetings; and

(iii) Managing SIGDA's finances according to the Financial Accountability Policy of the ACM. This includes preparing the annual budget, monitoring disbursements for adherence to the annual budget, and preparing financial reports as required.

Article 5. The Executive Committee

(a) The Executive Committee comprises the officers, the Past Chairman, the Editor of the SIGDA newsletter, and the chairman (if any) of the Advisory Board. No person may hold two positions on the Executive Committee.

(b) The general duty of the Executive Committee is to advise the Chairman on all matters of interest to SIGDA. Specific duties or responsibilities may be specified in these Bylaws or assigned by the Chairman. All the major management policy decisions of SIGDA must be approved by the Executive Committee. Specific duties of the Executive Committee include:

(i) Approval of bylaw amendments before submission to members.

(ii) Approval of annual dues for SIGDA.

(iii) Approval of the annual budget and review all expenditures in excess of $50,000 on a quarterly basis.

(iv) Approval of conferences, symposia, workshops or sessions sponsored, co-sponsored or held in cooperation with SIGDA.

(c) A quorum is a majority of the members of the Executive Committee and approval requires a majority vote of those present. Approval by mail ballot requires a majority vote.

(d) All members of, or candidates for, the Executive Committee must be voting Members of ACM and of the SIG.

Article 6. Vacancies and Appointments

(a) Should the Chairman leave office before his term expires, the Vice-Chairman will assume the duties of Chairman. Should any other elective office (including Past Chairman) become vacant, the Chairman of the SIG Board may, on nomination of the SIGDA Chairman, fill the vacancy. The Chairman may fill vacancies in positions appointed by the Chairman, according to the procedures for making the original appointments as provided herein.

(b) Should a vacancy be unfilled, either because of inadequacy of these bylaws or because of a dispute or for any other reason, the SIG Board Chairman may fill it (as provided in Bylaw 7 of the ACM).

(c) All appointments expire automatically when the Chairman's term of office expires.

Article 7. The Newsletter

(a) SIGDA will publish a newsletter at regular intervals as determined by the Executive Committee. The newsletter will be distributed to all members. Newsletter subscriptions may be sold to nonmembers.

(b) With the advice of the other officers, the Chairman will appoint the Editor of the Newsletter, who will become a member of the SIGDA Executive Committee.

Article 8. The Advisory Board

(a) The Advisory Board includes the officers and the Past Chairman. It also includes members-at-large who are appointed by the SIGDA Chairman. The Chairman normally appoints ten members-at-large to the Advisory Board for his or her term of office.

(b) The SIGDA Chairman serves as Chairman of the Advisory Board, or at his or her option the Advisory Board may select one of its members-at-large to be its chairman who will then be a member of the SIGDA Executive Committee.

(c) The purpose of the Advisory Board is to allow members outside the Executive Committee to participate in setting policy and direction for, and assist in the operation of, SIGDA.

Article 9. Membership, Dues, and Voting Privileges

(a) SIGDA is a restricted SIG as defined in Section 6 of Bylaw 7. Membership in the group is open to any member of the ACM, or to any person whose major professional allegiance is outside the computing field.

(b) An eligible person becomes a member only after enrolling and paying the required dues. The dues for SIGDA are determined by SIGDA Executive Committee with the approval of the Chairman of the SIG Board. All dues, and any fees for activities and services of SIGDA, must be lower for members of the ACM.

(c) All members of SIGDA may vote in any ballot conducted with SIGDA. As provided in ACM bylaw 7, on any ballot, the votes cast by non ACM members of SIGDA will, if necessary, be prorated downward so that their effective total cannot exceed 50% of the eligible votes. When it applies, the proration factor will be specified on the ballot.

Article 10. Reports and Records

SIGDA Chairman is responsible for filing reports about SIGDA as required by the SIG Board. These include:

(i) An annual report on the activities during the previous year;

(ii) All reports required by the Financial Accountability Policy of the ACM; and

(iii) Closing reports on conferences and symposia.

The membership records of SIGDA will be maintained by ACM headquarters.

Article 11. Elections

(a) The Chairman shall appoint a nominating committee in the autumn of each even-numbered year, unless the SIG Board has exercised its option (under ACM Bylaw 7) to extend the terms of all SIGDA's current officers for an additional two years. This committee will nominate at least two candidates, who consent to serve, for each of the elective offices. A report of the nominating committee must be presented to the SIGDA membership before an election can be held.

(b) A petition from at least ten of the voting members of SIGDA will place other consenting candidates on the ballot. Petitions must be received by the Secretary-Treasurer no later than April 15.

(c) The election will be conducted among eligible voters by ballot mailed by the nominating committee or by ACM Headquarters, following the election procedures of the ACM. Of all the ballots returned in an election, the candidates receiving the largest number of effective votes win. The SIG Board will resolve ties.

Article 12. Amendments

(a) These bylaws may be amended by a majority vote of the ACM Executive Committee, or by a vote of SIGDA's members as provided below. With the approval of the SIGDA Executive Committee, the SIGDA Advisory Board and the Executive Committee of the ACM, 2/3 of all the members of the SIG Board may amend Article 1 of these bylaws without a referendum of the members.

(b) Amendments to these bylaws may be proposed by the SIGDA Executive Committee, the SIGDA Advisory Board, the SIG Board, or a petition from 10 of the voting members of SIGDA. All proposed amendments must be approved, prior to being submitted for a

vote of the membership, by the Advisory Board and by the Chairmen of both the SIG Board and the Constitution and Bylaws Committee of ACM after the Executive Director of ACM has provided his advice.

(c) The ballot on the proposed amendment(s) will be conducted among the eligible voters by ACM Headquarters following the procedures of the ACM for voting bylaw amendments, unless a different procedure has been approved by the SIG Board. The proposal is adopted only if at least 2/3 of the effective votes of returned ballots approve it, and only if at least 1/4 of the ballots are returned. The Secretary-Treasurer will send a clean copy of the amended bylaws to the Executive Director of ACM and to the Chairman of the SIG Board.

Article 13. Dissolution

Should SIGDA be dissolved, control of its assets will revert to the ACM.

Article 14. Meetings

SIGDA will conduct at least one business meeting each year, normally in conjunction with the annual Design Automation Conference. All meetings sponsored by SIGDA must be open to all members of the ACM. SIGDA may hold meetings only in places that are open to all classes of members of the ACM.

Article 15. Consistency

The Constitution, Bylaws, and policies of the ACM and of the SIG Board take precedence over any conflicting provisions of these bylaws or internal policies of SIGDA.

High School Scholarship Program.....Charlotte Acken

The ACM (Association of Computing Machinery) and SIGDA (Special Interest Group on Design Automation) are sponsoring a scholarship program for high school seniors from under-represented groups in engineering (Blacks, Hispanics, American Indians, women, or disabled) who wish to pursue careers in computer science or electrical engineering. These scholarships are being offered to Bay Area students only. Recipients will be chosen May 18th; two students will be awarded $4000, renewable scholarships and two will receive one time only awards of $1000.

We are soliciting CS/EE professionals who will serve as mentors for scholarship recipients. Duties of the mentor are as follows:

Give a career oriented presentation to an area high school to help publicize the scholarship and interest other students in pursuing careers in engineering. Arrangements for this presentation will be done through CA MESA, an extensive outreach organization which works closely with kids from underrepresented groups encouraging students to pursue careers in math and science.

- Assist in selecting recipients from a pool of applicants on May 18th in Santa Clara, Expenses for this meeting will be reimbursed by SIGDA. Recipients will be paired with mentors at this meeting according to similar background. If you are paired with a student your duties will be as follows:

  Attend the Design Automation Conference, June 24-28, in Orlando, Fl, and show your student around the exhibits on the keynote day June 25. Travel, Conference registration, lodging and per diem for this trip will be reimbursed by SIGDA.

  During the student's college career, provide encourgement and assistance where possible.

  Act as a role model for the student and help him/her to understand more about their chosen career.

If you are interested in becoming involved in this program please contact:

Charlotte Acken
Director, ACSEE Scholarship
(415) 294-3248
e-mail: cacken@sandia.llnl.gov

Scholarships.....Hersch Loomis

Here's a brief status report on the scholarships. We have received a total of 51 applications and they have been forwarded to Chuck Shaw and Chuck Radke who with me will read and evaluate them. We will report on the results and notify winners and also-rans on 1 May.

Regarding the overhead question, everyone who replied was in agreement that we NOT allow overhead and/or indirect costs. Our policy statement reflects that. It was also felt by most responders that we should continue to be flexible in allowing almost any other costs, on a case-by-case basis. Thanks for your interest, all!

ED. There was some lively debate by email on whether the scholarship program (as administered by Hersch Loomis) should include payment of overhead in its grants. Here are some excerpts from that debate.

From: C.RADKE
Subject: SCHOLARSHIPS--DISAGREEMENT

HERSCH, I HAVE TO DISAGREE WITH YOU. I HAVE BEEN ON THE SCHOLARSHIP COMMITTEE (I BELIEVE THAT I AM ON AGAIN THIS YEAR FOR DATC AND SIGDA, BOTH) FOR A NUMBER OF YEARS. HECK , AT SOME TIME BACK I INITIATED THE FIRST ONE. AS LONG AS I CAN REMEMBER, WE HAVE HAD PROPOSALS IN WHICH THE PROFESSORS HAVE 1)SPLIT UP THE AMOUNT AMONG STUDENTS, IN-

CLUDED EQUIPMENT, AND EVEN INCLUDED SETTING UP SPECIAL PROGRAMS. WE HAVE OFTEN (WELL AT LEAST SEVERAL TIMES) GIVEN THE "SCHOLARSHIP" TO ONE OF THOSE PROPOSALS BECAUSE IT FIT IN WITH OUR DESIRES TO INITIATE NEW THINGS. I FEEL THAT WE SHOULD LEAVE ROOM, PLENTY OF ROOM TO ALLOW THE SUBMITTER TO COME UP WITH JUST ABOUT ANYTHING NEW. IN FACT, AT THE NOVEMBER BOARD MTG WE APPROVED OFFICIALLY THAT THE FUNDS COULD BE USED FOR EQUIPMENT . I BELIEVE, THAT WAS THE INTENT. I WILL NEED TO LOOK UP THE EXACT WORDING.

I WOULD PROPOSE THAT WE INDICATE THAT UNLESS OTHERWISE STATED THAT IT WILL BE EXPECTED THAT THE FULL $12,000 GOES TO THE STUDENT FOR THE STUDY; HOWEVER, WE ARE ENCOURAGING INNOVATION IN HOW THE MONEY CAN BE USED TO FURTHER THE EDUCATIONAL ENVIRONMENT. I DO AGREE THAT THE MONEY SHOULD NOT GO FOR OVERHEAD, BUT IN SOME STATE SCHOOLS THE SCHOLARSHIP MONEY IS USED FOR TUITION. I FEEL THAT THE BEST WAY IS TO LET THE SUBMITTER DETERMINE WHAT HE WANTS TO PUT IN THE PROPOSAL. THE EMPHASIS IS THAT SIGDA IS CONCERNED WITH HAVING THE MONEY BE PUT DIRECTLY TO FURTHERING THE EDA EDUCATION.

ANY BODY OUT THERE AGREE WITH ME?????? CHUCK <Radke>

TO: Chuck <Radke>

I believe the policy of not paying overhead/indirect costs is consistent with everything you said in your message. We would still allow any reasonable DIRECT costs such as student stipend, equipment, etc. So, I guess I don't see how you disagree.

- Hersch

TO: Chuck <Radke>,

This is part 2 of my comments. They occurred to me after I sent the previous message.

I am only suggesting we eliminate overhead. We need a formal statement that overhead is excluded because without that statement, some schools require that it be included in all proposals.

As far as tuition is concerned, the student is free to use whatever portion he or she gets in any way including paying tuition.

- Hersch

TO: SIGDA Board

I would not pay overhead. Paying would effectively reduce the award by approximately 1/3 to 1/2 at most schools. I think it was a different thing to hire a student for a SIGDA project as Mike did. These scholarships should be considered gifts and gifts don't normally generate overhead.

- Jim Cohoon

TO: SIGDA Board

I agree with Jim Cohoon.
- Ron Waxman

TO: SIGDA Board

I, too, agree with Jim. University overhead should not be charged against the scholarship. At PSU overhead is 49% - at some private schools its close to 70%. Most reasonable Universities will agree to waving overhead for this kind of award. We certainly had not problem getting PSU to agree to this last year. Anyway, the faculty and the student never get any direct benefit from the money - it goes to pay the administrators over in old main, keep the buildings heated, etc. --janie *irwin*

TO: SIGDA Board

No overhead on scholarships and grants. These are not projects and should not generate any additional overhead for the university.

-Dick Smith

TO: SIGDA Board
Subject: SCHOLARSHIP--AGREEMENT; I MEAN AGREEMENT

WHERE WOULD YOU BE IF EVERYONE AGREED ALL THE TIME. I AGREE ON THE OVERHEAD, BUT WOULD HOPE THAT THERE IS ENOUGH FLEXIBILITY TO ALLOW INNOVATION--AND IN FACT ENCOURAGE IT.

-CHUCK <Radke>

ACM Meetings at CSC'90

Michael Lorenzetti

I represented SIGDA at several ACM level meetings held at the Computer Science Conference (CSC90) held February 20-21 in Washington DC. This report is a summary events of those meetings of interest to SIGDA. (NOTE: these are not intended as minutes, since only selected topics are included).

Electronic Submissions Workshop

This was a meeting chaired by Edward Fox, ACM Publications board. Attendees included representatives from SIGs, Fred Aaronson from HQ, newsletter editors, and some industry representatives in area of electronic publishing.

Fox is beginning work on standards for formats of submission, storage and distribution for ACM pubs. His goal is to have an Electronic Library to include video tapes, hypertext, papers in electronic form and other electronic media.

The first agenda item was a review of current practices at ACM Publications. Mark Mandelbaum, ACM director of Publications, told us that all SIG newsletters are being transferred to ACM publication department. For othe publications, reviews (but not articles) are submitted electronically. Bernard Rous, Associate Director of Publications, informed us that they are bringing in a Sun-based system for editing. The output is postscript.and the internal format is similar to SGML. They can do edits, produce camera-ready copy and archive the publications. They cannot now accept external electronic submissions. They expect to have the system installed by the end of March. ACM Communications are edited electronically, Transactions are not. Experiments have been done with TEX submissions but this is not the normal mode of operation.

Betsy Kiser, from the Electronic Publishing Special Interest Group (EPSIG), explained the Standard General Markup Language (SGML). The Electronic Manuscipt Project (1983-1986) developed an application of SGML for publishing known as *Standard for Electronic Manuscript preparation and Markup*. This has since been approved as an ANSI standard. Fox advocates formulating a document type definition for each ACM publications using SGML.

J. Sperling Martin, from the ATLIS Consulting Group, discussed experiences in electronic publishing. For other organization looking into electronnic

publishing, the driving force is not electronic submission, but rather the publication database and the editorial process.

Gary Marchianni, University of Maryland, is assembling a compendium of hypertext articles on behalf of ACM. They are open to other suggestions for hypertext publications (After the meeting, I gave him Jim Cohoon's name as a person to contact regarding the SIGDA activity in this area. There may be some room for cooperation here, as we have just budgetted $5000 to look into this media.)..

The floor was then open for discussion Tools for conversion between formats are important because there are many accepted publishing systems. Authors will write papers before they decide where to submit them, so we can't just impose the ACM style and must accept several formats.

Fox presented a list of reasonable formats (in his own order of preference).

SGML
TEX or LATEX
Microsoft Word or Word Perfect
Mathmatica Files
Plain ASCII if others are not possible

It was agreed that *troff* should be added to this list. Fox put out a call for SIG involvement in the process of defining standards. It was agreed to leave the first pass in the hands of ACM HQ. HQ could Poll contributors as to what editing tools were used. Should there be a volunteer committee to formulate plans and supervise implementation of the plan for electronic submissions? No decisions were made at this meeting. Comments on how to proceed should be directed to Fox.

SIG Discretionary Fund

The SIG Discretionary Fund (SDF) is a means for SIG Board to support worthwhile projects that might not be otherwise be supported by a single SIG. Individual SIGs make annual (voluntary) contributions to the fund. The purpose of this meeting is to review the proposals submitted for funding and allocate funds among them.

SIG Board Chair, Ron Oliver, reviewed the criteria for funding: (1) it addresses some aspect of technical excellence and (2) must be one-time funding, not an on-going program. This year, they added the concept of out-of-cycle proposals. Requests for funds sometimes are presented to SIG

Board mid-year, and often there are circumstances which do not allow them to wait until the start of the next fiscal year. SIG Board now has the authority to award funding for these out-of-cycle projects, provided there are sufficient funds and they feel that the circumstances warrant it.

SIGDA, because of its large surpluses, remains the largest contributor to the fund. In order to insure that the fund remains a cooperative effort, SIGDA worded its contribution as follows:

> Subject to an absolute ceiling of $200,000, SIGDA will contribute an amount equal to the higher of the following two numbers: the largest contribution of any other single SIG or half the contribution of all other SIGs combined.

This is the third year of the SIG Discretionary fund. Including the carryover from previous years, the total available funds available for FY'91 is $635,380. 15 proposals were presented, each followed by a short question and answer session. Each proposal was then discussed in turn and a vote was taken on each. The results are summarized below.

SDF Proposals Approved for Funding

Conference on Critical Issues in CS	40,000
International Activities	40,000
NRC study on Scope and Direction of CS	30,000
NRC study on Human Resources in CS	10,000
Publication of ISSAC'90 Proceedings	15,000
Role of Disabled in Computer Professions	12,400
21st SIGCSE Symposium	15,000
SIGCAS Conference (an additional 22,500 for awards was turned down)	19,000
2-year computer curriculum development	59,700
ACP Exam for Associate Rehabilitation Projects in Data Processing	30,000
TOTAL	271,100

SDF Proposals Invited to Resubmit Out-of-Cycle

Data Collection / Survey Management (proposal not yet complete)	150,000
Revision of ACM Code of Ethics (does not have definitive ethics recommendation as output)	94,000
ACM Student Publication (proposal not yet complete)	30,000
TOTAL	274,000

SDF proposals that were Turned Down:

Blue Ribbon Panel on Intellectual Property Rights	154,000
Computer Chess Documentary	111,700
Hypertext version of Self Assessment Procedures	23,066
TOTAL	288,766

The table on the next page is a summary of the SIG contributions to the discretionary fund for the past 3 years. The fifth column is the fund balance of the SIG as of the close of FY'89 and the sixth column is the percentage of that fund balance that their FY'91 contribution represents. The total contibutions to the FY'91 fund works out to 6.1% of the total FY'89 fund balance. It is my opinion that SIGDA needs to take an active role in encouraging other SIGs to contribute to the fund.

Joint SIG Board / SIG Chair Meeting

There was a Presentation by The Southwest Regional Representative on behalf of California State University at Northridge. They are beginning a new organization to deal with access to computers for people with handicaps. Looking for support of some kind from ACM. After some discussion, it was proposed that they become a local SIG of SIGCAPH (computers and the Physically Handicapped).

It was proposed that we formalize the SDF proposal review process. Ideas included a formal call for proposals, more specific selection criteria, a review committee to pre-screen proposals and make recommendations (this was a point of contention). Ron Oliver agreed to appoint a Committee to make a detailed proposal on this. (Note: SIGDA needs to have representation here since we are the largest single contibutor to the fund. I have volunteered to join this committee.).

SIG Contributions to SDF

SIG	SDF FY'89	SDF FY'90	SDF FY'91	Fund Bal 4Q89	% of FB for FY'91
ACT	5000	10,000	10,000	144,400	3.5%
Ada	10,000	15,000	15,000	334300	4.5%
APL	0	1000	2000	71,000	2.8%
ARCH	0	2000	4000	274,000	1.5%
ART	5,000	5,000	5000	89,500	5.6%
BDP	0	800	800	70,400	1.1%
BIO	500	0	0	28,363	0.0%
CAPH	0	0	0	31,480	0.0%
CAS	0	100	100	22,700	0.4%
CHI	25,000	25,000	0	400,600	0.0%
COMM	10,000	10,000	10,000	155,700	6.4%
CPR	0	0	500	20,000	3.2%
CSE	100	150	200	25,000	0.8%
CUE	0	1000	1000	32,000	3.1%
DA	200,000	200,000	135,350	1,447,900	9.3%
DOC	500	0	0	70,900	0.0%
FORTH	-	0	100	(10,200)	*.*%
GRAPH	150,000	50,000	100,000	2,245,600	4.4%
IR	500	1,000	1,000	17,500	5.7%
METRICS	0	1,000	1,000	85,400	1.2%
MICRO	0	0	0	(700)	0.0%
MOD	0	0	0	22,500	0.0%
NUM	0	1,000	2,000	37,500	5.3%
OIS	0	0	2,000	62,700	3.1%
OPS	500	1,000	1,000	105,600	0.9%
PLAN	100,000	200,000	100,000	817,800	12.2%
SAC	0	3,000	3,000	46,168	6.5%
SAM	0	0	0	3,500	0.0%
SIM	0	0	0	37,500	0.0%
SMALL/PC	1,000	2,000	2,000	66,500	3.0%
SOFT	5,000	4,000	0	39,500	0.0%
UCCS	10,000	10,000	10,000	97,200	10.3%
other	500	-	-		
TOTAL	523,600	543,050	406,050	6,649,628	

ACM president, Brian Kocher, addressed group. His expressed his disappointment that the computer bowl proposal was turned down when he proposed it for SDF funding in the fall and again when he submitted it to the SIG Board out-of-cycle. Kocher pointed out that the computer bowl provides high publicity for ACM and fits with the strategic plan. A discussion ensued about the purpose of the fund and how to improve cooperation between ACM executive committee and SIG Board. No action was taken.

Joe DeBlasi reviewed the ACM HQ reorganization. They are forming an office of SIG services, managed by Pegotty Cooper. It consists of 6 Liaison managers, each assigned to a group of SIGs to service. SIGDA is included with SIGMETRICS, SIGMICRO, SIGOPS, SIGSAC, SIGSIM and SIGSAM in a single group. Interviews to fill these Liaison Manager positions are in progress and candidates look very good, many have experience in professional organizations.

Hal Berghel of the University of Arkansas presented a proposal for a new SIG on Applied Computing (SIGAPP). This is an outgrowth of the annual Symposia on Applied Computing (SAC). There was some concern about overlap with many other SIGs. Their areas of interests include CAD/CAM among many others. The proposal will be formally submitted to SIG Board after the necessary signatures are gathered at their upcoming April Symposium. Plans for a new SIG on Two-Year Colleges are also being formulated but no formal proposal is yet available.

Ron Oliver announced that this is his last meeting as SIG Board Chair. He reminded us that 68% of the ACM cash flow is through the SIGs. SIGs are gaining a larger voice in ACM Council by the changes in Council structure. The next few years are very important in increasing SIGs voice in ACM.

SIG Board Meeting

Conferences Board proposed a joint sponsorship agreement for co-sponsored conferences and workshops which must be signed prior to approval of the TMRF. They will provide a template for such an agreement. The idea is to address issues of which society is responsible for what in advance of the conference. Freidman announced they are very flexible regarding what is included in the agreement, they are only concerned that there be advance agreement. A single agreement can cover several years. SIG Board voted to endorse the proposal, which now must go to ACM Council.

SIGDA bylaw amendments were on the agenda, but were postponed to give us time to incorporate the comments from the Constitution and Bylaws Committee. They will be put on the agenda for the fall SIG Board meeting.

DAC Insider's Guide

The 27th Design Automation Conference (DAC) will be held in Orlando, Florida this year from June 24 through June 28. The main event is the excellent technical program along with the commercial tradeshow exhibit. Also, there are a number of professional society meetings, including the SIGDA annual meeting, the IEEE/CS DATC meeting, and meetings of various standards groups. There are also a series of tutorials on Thursday, June 28th.

TECHNICAL SESSIONS

The technical sessions of the DAC are always of high quality and very well attended. Leading off the technical sessions will be the keynote address by Bill Joy. Bill was (and still is) instrumental in founding Sun Microsystems. He is well known for his design of the BSD UNIX (TM AT&T) operating system, and later for his efforts on the Network File System (NFS) and the SPARC (TM Sun Microsystems) chip. His topic will be "Engineering the Future."

There are 43 sessions ranging from "HDL Validation and Intermediate Format" to "Ideas in Testing." All of the papers submitted for publication at the DAC were reviewed by the Program Committee, and about one fourth were accepted. In addition a block of time from 2:00 pm to 7:00 pm on Sunday has been set aside for exhibitor technical/marketing presentations. Here exhibitors will make product announcements, discuss product strategies, and make other announcements of note. The press will be there to cover this event.

EXHIBITIONS

One hundred and twenty companies with thousands of marketing, sales, and support personnel will be exhibiting their wares in halls B, C, & D (like - Binary Coded Decimal) of the Orlando/Orange County Convention Center. The exhibit times are:

SUNDAY (24 June) ..... 2:00 - 7:00 pm
MONDAY (25 June) ..... 9:30 - 6:00 pm
TUESDAY (26 June)..... 9:30 - 6:00 pm
WEDNESDAY (27 June)..... 9:30 - 3:00 pm

These exhibits will be a good place to get a feel for the "state- of-the-art" in CAE/CAD. There will also be some other interesting noncommercial booths in the exhibit area. The CFI (CAD Framework Initiative) will be showing their framework standards on a number of platforms with a number of vendors.

Also, the SIGDA-sponsored UNIVERSITY BOOTH will again be in the exhibit area. It will be bigger and have more equipment than last year. This is the third year of this project, and it has grown in popularity and size every year. Dr. Martin Wong of the University of Texas is the coordinator of this year's UNIVERSITY BOOTH project. Just as the commercial exhibits are a place to view todays tools, the UNIVERSITY BOOTH is the place to see the CAE/CAD tools of tomorrow.

CD-ROM DEMONSTRATED

The CD-ROM that Bryan and Kathy Preas have been working on will be demonstrated at the UNIVERSITY BOOTH. The CD-ROM project will take all of the available electronic CAE/CAD literature (DAC Proceedings, ICCAD Proceedings, etc.) and put them on a set of optical disks (CD-ROMS). See the report on page 22 of this Newsletter for details.

Please come by an look at this exciting new research tool. People will be on hand to help you perform a search - perhaps for your own papers and citations.

TUTORIALS

The tutorials are always very popular and fill up very fast. But if you have not registered in time, check on the bulletin board for cancellations. And if you are registered for a tutorial, but for some reason can not attend, post your opening.

The tutorials tend to be very topical, emphasizing emerging trends and technologies. This year there is a tutorial on designing with VHDL, two on logic synthesis, one on object- oriented programming for CAD applications, one on design for test, and one on parallel processing for CAD applications.

SOCIAL EVENTS

Yes, there are parties, free dinners, snacks, and drinks. Many are provided by exhibitors, some by professional societies (like the SIGDA). You shouldn't go hungry or thirsty at the DAC. Most companies have suites where you can go eat, drink, and be merry - and perhaps pick up their brochure. Signs are typically posted in the hotel lobbies showing open times and suite numbers. Also, there will be a few major parties, also sponsored by exhibiting companies.

BIRDS OF A FEATHER MEETINGS

These are meetings after the DAC that focus on specific areas of interest, often too small or too specialized to get a formal session at the DAC. Currently scheduled is a meeting for the CFI. Others will be posted, and a schedule should be available at the information booth.

SUMMER SIGDA MEETING

What ever you do, don't miss this. This is typically the largest SIGDA meeting. It's a chance to meet the SIGDA officers and board members, express your feelings on the direction of the society, and socialize with your peers. Also, there are drinks and enough food for a meal (don't book a reservation at a restaurant on Sunday night). And it's free. See you there!

Pat Hefferan - SIGDA Newsletter Editor

SPECIAL FEATURE - Reprint of First SIGDA Newsletter

This is the 20th year of publication of the SIGDA Newsletter. And to commemorate that fact, we are publishing the first recorded edition of the Newsletter (Vol. 1, No. 1). The SIGDA Chairman was Chuck Radke who in his first act as SIGDA Chair, appointed his defeated rivals (Gerhard Paskusz and Walter Samek) to the position of Newsletter Editors. I hope you enjoy this look at the past.

Thanks to Chuck Radke, all of the SIGDA Newsletters will be recorded on the CD-ROM. Chuck archived all of the Newsletters and passed them onto Bryan Preas for capture on the SIGDA CD- ROMs. Currently, Chuck is a SIGDA Board member in charge of outstanding member recognition, publicity, IEEE representative to the DAC Executive Committee, and active member of the SIGDA Advisory Board (see pages 34 & 35). THANKS CHUCK...

Chuck Radke

ASSOCIATION FOR COMPUTING MACHINERY · ACM · FOUNDED 1947 ·

SIGDA NEWSLETTER

SPECIAL INTEREST COMMITTEE ON DESIGN AUTOMATION

Vol. 1, No. 1 January 1971

Message From Your Chairman

Is SIGDA for real? I feel that a large percentage of the membership (83 out of around 200) has shown that it is.

In a letter dated June 10, 1970, Jean Sammet as Chairman of ACM Committee on SIG's and SIC's sent each of us a letter concerning the impending dissolution of SIGDA. Robert Hitchcock of IBM Research tried to reverse the trend by organizing a meeting of interested SIGDA members (and some non-members) at the 1970 DA Workshop in San Francisco. The result was that a list of interested people and potential candidates was generated. The SIGDA Nominating Committee, under chairmanship of J. B. O'Neill, met and increased this list of interested volunteers by four additional people. You received the list of eight candidates, and a whopping 40% of you voted. I don't know on what basis you made your choice, but I do know that we had eight interested and talented people running. And as you can see from my list of appointments, I am making good use of them.

To put first things first, I consider the most important item on the agenda to be one of communication with the membership. Our only means of achieving this at the present time is through a newsletter. I have appointed an Editorial Board headed by Walter Samek, whose interests lie in the area of Mechanical DA. Jerry Paskusz, who is active in theory of design and educational courses in design and DA, has agreed to be associate editor. As a third member of the board I have asked Larry Margol, whose interests lie in the long established DA area of electronics and computer design, and who is also your Vice Chairman. The Editorial Board is broad not only in geography but also in disciplines. My intent is not just to cover the geography of ACM, but also the disciplines affected by the increasing use of DA.

Although our first issue is small and meets only a few of the objectives which we think a Newsletter should serve, it is a start. In 1971 we plan to publish at least 3 issues. To a large extent our plans depend upon your interests; that is why I am including a survey in this initial newsletter. I'll leave it up to Walt to give you the pitch on some of the plans and objectives of the SIGDA Newsletter.

Charles E. Radke
SIGDA Chairman

Present SIGDA Organization

Executive Committee:

Charles E. Radke - IBM, Poughkeepsie, New York - Chairman

Lawrence Margol - Microdesign, Anaheim, California - Vice Chairman

John Hanne - Texas Instruments, Dallas, Texas - Secretary/Treasurer

Editorial Board

Walter Samek - Combustion Engineering, Windsor, Connecticut - Editor

Gerhard Paskusz - University of Houston, Houston, Texas - Associate Editor

Lawrence Margol - Executive Committee Representative

Official Representative of SIGDA on 1971 Design Automation Workshop Committee - C. E. Radke

Editor's Note

A new administration is taking office at SIGDA, consisting of a Chairman, a Vice Chairman, and a Secretary-Treasurer. The election, which brought these people into office, produced the following tally:

For Chairman:
Charles E. Radke..... 33 votes
Gerhard F. Paskusz..... 29 votes
Walter J. Samek..... 21 votes

For Vice-Chairman:
Lawrence Margol..... 37 votes
Stephen P. Krosner..... 25 votes
Marco Somalvico..... 21 votes

For Secretary-Treasurer:
John R. Hanne..... 44 votes
Stephen A. Szygenda..... 39 votes

Therefore Radke and Margol have it by plurality and Hanne by majority.

As his first act as newly elected chairman Chuck Radke has appointed his defeated rivals Editor and Associate Editor of the SIGDA Newsletter. A neat trick, if I ever saw one, he gets the glory, and we do the work.

As your new editor I am herewith informing you, that the SIGDA Newsletter, which has heretofore been a good publication, will be maintained at its high standards. However, a few remarks are in order.

Only 83 people, out of a total of more than 200, took part in the election. This low participation may well have been caused by the fact that the candidates were not known to the membership. This anonymity, in turn, is at least partly due to lack of activity on the part of SIGDA, and lack of communication among its members. This newsletter will endeavor to remove the second of these lacks. The methods it will employ to achieve this goal are currently planned to be as listed below. Changes will be made to these methods whenever new ideas are brought forward and convincingly presented. Readers are invited to participate in this game. The newsletter is your servant, not your master, so don't hesitate to tell it what to do.

The SIGDA Newsletter will attempt to accomplish the following tasks:

1. Provide a chronicle of SIGDA and other DA-related activities during the period immediately preceding its publication.

2. Serve as a medium for the dissemination of Chairman's messages, Executive Committee decisions, election results, announcements of upcoming meetings, appointments and other items concerning SIGDA business.

3. Serve as a medium for the dissemination of news concerning SIGDA members, report their achievements, publications, patents and personal data.

4. Serve as a medium for the dissemination of technical information and newsworthy items related to DA.

5. Serve as a forum for the expression of opinions related to items previously published.

6. Exercise a unifying influence on the activities of SIGDA. This editorial policy can be put into a nutshell by stating that DA is a multidisciplinary activity, and that SIGDA is trying to crystallize the common ingredients from the various approaches reported by its members.

Walter J. Samek

Technical papers appearing in this issue are unrefereed working papers.

Attention all Authors

Authors of DA related papers, talks, articles or books are herewith invited to submit a one-paragraph abstract of their work for publication in the SIGDA Newsletter. Please note that all company rules on proprietary material, all government regulations on classified information, and all publishers' rules on abstracts must be observed by the authors. SIGDA accepts no responsibility for any unauthorized disclosure or other violation of such nature.

Attention all SIGDA members

The SIGDA Newsletter depends on you for information concerning newsworthy items for its pages. So keep those cards and letters coming to any one of the editors, whose names, addresses and phone numbers are listed below.

Walter J. Samek
Combustion Engineering, Inc.
Dept 617-3
Windsor, Conn. 06095
(203)688-1911 X128

Gerhard F. Paskusz
University of Houston
Cullen College of Engineering
Houston, Texas 77004
(713)748-6600 X516

Lawrence Margol (RC01)
N. American Rockwell
Micro Electronics Co.
3370 Miraloma Avenue
Anaheim, Calif. 92803
(714)632-8565

Design and Automation

Designautomation

Design Automation System

By C. E. Radke

At the recent ACM 70 Conference, Bob Hitchcock chaired the SIGDA meeting. There were four of us in attendance: Bob (IBM), John Hanne (Texas Instruments), Murray Freeman (Philco), and myself (IBM). Later we caught a passer-by, Bob Brover (UCLA student). Although Bob was not acquainted with SIGDA or its purposes, at our insistence he did come up with a definition of Design Automation (DA). His on-the-spot definition was:

DESIGN - An iterative, decision-making process that seeks to optimize the value of society's resouces.

AUTOMATION - Selecting of procedures for doing work with less effort.

To me this definition sounds very much oriented toward the DA user (i.e., the engineer who is looking for a "tool" to aid him in his developing his design).

What happens if one asks a DA Development group (i.e., those individuals who generate the computer programmed procedures) that same question? One might expect a definition for the entire phrase: "Design Automation, why it's really not automation, its computer-aided design." The key word, of course, refers to the person's main interest - "computer". Let's now ask the Information Systems man what DA is. He might say: "First, it isn't Design Automation, it is DA System; and a DA System is the software required to enter all design information into a computer document it, keep records, and turn out manufacturing data. A course, today all this must be terminal oriented.

Which function do you serve; what is your definition? Whichever it is, SIGDA will want to serve you.

We went through another exercise at the SIGDA meeting at the ACM 70 Conference in New York City. We formed a two-dimensional array with "Discipline" on the Y-axis and "DA Function" on the X-axis. For example, for discipline one might have:

Aerospace Design	Electronics Circuitry Design
Civil Engineering	Computer Design
Mechanical Engineering	Optical System Design
Automative Engineering	Communication System Design

And for function one might list:

Simulation and Modeling

Testing

Fault Diagnostics

Design Verification

Packaging and Physical Realization

Synthesis

Input of Design Data

File Structure Definition and File Updating

Documentation

Process Automation

Records

Languages

Manual Input/Override

At a particular X-Y coordinate intersection one might find either a blank, or a full history of the effort expended to fulfill the particular function by DA for the selected discipline. For example, one might expect a small amount of effort to generate test procedures to test an automobile design, but would find that much more effort was required to aid in fault detection of a computer system and its parts. As you can visualize, the number of combinations of discipline, function, and degree of effort is large, and that is another reason for you to let us know where you fit in the matrix.

The following item is included in this issue to illustrate one DA application to mechanical design.

Design Automation of Expansion Joints

An expansion joint is a part of a gas duct placed between various boiler components to absorb thermal expansion without transmitting forces. Its shape is similar to that of an accordion.

The computer used is an IBM 7070, conversion of the program to the IBM 360 is imminent. Input consists of length and width of the rectangular opening to be covered (typically 10 x 20 feet), and the number of pleats desired (this may be anywhere between 1 and 9).

The program selects the proper plate design from 14 standard shapes, adjusts certain dimensions to make the plates fit the desired size of opening, and produces:

a. a parts list showing the different plates used and how many of each,

b. a pictorial representation of every plate by means of a printed plot,

c. a perforated paper tape for every plate.

These paper tapes are used on a Wiedematic punchpress to produce the necessary plates from 10 gage steel, which are afterwards folded to generate the pleats, and welded together. No conventional method can produce better fitting expansion joints.

ED NOTE: If you understand this last page, please drop me a note.

CAD Tool Interchangeability through Net List Translation

Steve Meyer

Pragmatic C Software
220 Montgomery Street, Suite 925
San Francisco, CA 94104

Abstract: The argument for electronic circuit logic design tool interchangeability by means of source-destination specific net list translation within a framework of Unix operating system commands is presented. After describing the information necessary for design verification using an assortment of diverse tools, various design tool interchangeability alternatives are considered. Discussion of net list translator examples, Unix like design movement programs, and translation speed complete the argument.

1. Introduction

The number of available electronic circuit logic design and verification tools has increased in recent years. This tool availability has created a problem in tool interchangeability. From the circuit designers viewpoint, there is great advantage in the ability to mix and match ones favorite tools. It is also valuable to be able to use tools with special capabilities even though such tools may not be general purpose enough to replace, say, a company wide standard simulator. The advantages of tool interchangeability have been shown in various case studies. See [22] for an example in which two gate level simulators were used to verify a gate array design. Each simulator found timing problems missed by the other. This trend toward multiple design tools has recently been acknowledged by established CAD tool providers in what they call open systems or frameworks.

Most recent electronic circuit design methodology papers argue that design should be accomplished at a higher level, and that designers should be shielded from design verification tool details (see [3] for example). According to this view, not only design objects but also tools should be opaque objects within an object oriented design framework. This paper expresses the contrary viewpoint. It argues that logic design should move to a lower, more design specific, more tool specific, more concrete, and more accurate level. According to the view expressed here, the widely available Unix operating system [23] [2] provides as much framework as necessary. Source tool to destination tool net list language translation is then used for design movement between tools. Data is stored in normal ascii or binary files and maintained with Unix framework commands (cf. make [8] and SCCS [1][26][2, vol. 3]).

Net list translation plays the same role in design verification that Unix filters play in Unix text precessing. Translators are filters with optional configuration and mapping libraries to control the filtering. Translation is used to move designs between the various verification tools classes: schematic editors, timing calculators, simulators, timing verifies, and layout systems. This paper can alternatively be viewed as a commentary on various CAD opportunities and problems that arise from net list translator development.

The most obvious scheme for moving electronic design descriptions from the language format required by one tool to another tool's format is by means of a computer program. That computer program is called a net list translator. Since net list translators are low level tools that require careful attention to detail and considerable development effort, various alternative approaches to the tool interchangeability problem have been proposed. Before discussing the various alternative proposals, it is necessary to define the information required to characterize an electronic circuit and to describe the net list translation task.

2. The Four Kinds of Design Information

The main circuit description is the net list itself. It specifies cell-port-net connections, I/O ports, and primitive logic function types used in a circuit. Test patterns describe circuit behavior from one viewpoint and are the legal circuit definition when a circuit is moved to an ASIC vendor. Behavioral models and programs describe circuit behavior from another viewpoint. The fourth information class

includes primitive type element definitions (called models or model libraries). Primitives can be macro cells for ASIC designs, TTL parts for PCB designs, or transistor models for custom design. Since primitive type models are either represented as behavioral programs or net lists that are expanded before tool application, this fourth class either fits into the net list or the behavioral information categories and will not be considered separately.

Test pattern movement from one tool to another is usually easy and can be accomplished with an editor or macro language script. There is currently no general solution to behavioral module translation from one language into another since the problem is no easier than translating from one computer programming language into another. This requirement for manual recoding is usually not serious since before manufacture, any hardware system must be decomposed into a primitive based net list containing no behavioral descriptions. The net list form rather than the behavioral form component model description can be moved to the new tool. In early design stages when only behavioral models exist, it makes no sense to apply tools that do not support behavioral circuit description. Another use for behavior modeling is in test scaffolding and circuit debugging. In these cases one would expect to redo the behavioral descriptions for the new tool since the point to moving to a new tool is to take advantage of its different capabilities. The different test scaffolding can help to test the assumptions made in the original testing and debugging programs. Another possibility is to execute the behavioral test scaffolding on the original system while monitoring I/O ports. The sampled values are used as test pattern to drive the destination tool.

Notice this comprehensive design information movement paradigm allows all destination tool features to be utilized. If the object approach were used, only circuit description elements already built into the black box framework could be utilized by the destination tool. Of course, conventions and computer programs such as those supplied by Unix that improve user interface consistency and ease of learning are valuable.

3. Net List Translator Functional Requirements

A net list translator must preserve circuit meaning so that both structural gate level connectivity and semantic function are identical on the two systems providing the target system implements the necessary connection pattern functionality. When the source and destination function do not correspond, the translation needs to generate a net list that will have equivalent functionality for the destination tool's intended application. When construct mapping is ambiguous, a means for the user to specify translator mapping is required.

Optional translation libraries need to be provided to allow different translation levels such as exact timing versus unit delay timing and to allow primitive name mapping in order to facilitate destination net list use with with pre-existing libraries (precoded models). A translator should support all conceptual design aids: vectored wires (busses), signal wire concatenations, vectored part instances, synonyms, strength modeling, automatic name conflict repair, property mapping, timing checks, and even perhaps minor rewiring. A translator should map any feature that does not correspond exactly in the destination format into something as close as possible using the same coding style. A translator should never translate something that will have different destination tool semantic meaning unless errors or warnings are emitted.

4. Current Translator Limitations

One reason for scepticism toward net list translators is that many current translators are not complete. Many translators are just net-port or cell-port connection reformatters. They often use an automatic parsing system such as the lex/yacc combination [15][12]. This method is almost always unable to deal with the context sensitive nature of net lists and can not translate commonly occurring formats in which a primitive's type determines not only syntactic form but also lexical structure. For example, the entire TDL language [28] requires a part type recognition pass before the DEFINE section instances can be parsed. Hilo has various primitives such as **capacitor** that change element parameter interpretation. Language specific recursive descent (usually without any need for recursion) is advantageous for translating such irregular formats. Parsing routines can be passed the current type and then use it to determine tokenization.

It is not well understood that there is more to gate level circuit translation than simple net connection translation. Translation to insure identical cell-port-net and net-port-cell connections is not sufficient. Circuit descriptions also contain implicit semantic information that is built into the source tool verification algorithm and source language models. For example, some simulators model mos style gates by building strengths into the primitive models (see [18] for example) while others use general purpose primitives for which each instantiation must specify the required rise and fall strengths (see [10] for example). For output instances the mapped to type may need to include various instance parameters added by the translator.

Many net list translators handle special source language features by producing a destination format net list that is syntactically correct but expresses the wrong meaning. This is worse than not emitting anything since when a designer finds a problem during destination tool use, problem location is difficult. For example, since the Valid system simulator tracks and stores both logic polarity rails, net lists use a property (called BUBBLED) to indicate which rail to select during simulation. For destination tools that do not support dual rail logic, it is necessary to add inverters. Translators that do not add the extra logic cause incorrect destination tool results. The LSI Logic TDL dialect called NDL [17] allows modules that have outputs that are effectively power supply generators. If translated modules do not have output I/O ports connected to the supply net, when the module is expanded (flattened) before simulation, the higher level signal nets will not be correctly tied to the power supply. Some output formats allow outputs to be tied to power supplies while others require logic gate addition.

5. Net List Translation Alternatives

Before considering the various possible disadvantages of net list translation, the adoption of one universal design standard is considered. Standardization eliminates any need for net list translation because all electronic circuits are stored in the one standard format. This section concludes by considering two proposals that while still requiring net list translation, claim to simplify tool interchangeability. One proposal would standardize on one universal intermediate language. Net list translators would then be written to translate into and out of the standard form. The other proposal uses net list translation within a tool framework. The tool framework then supposedly simplifies circuit design through abstraction and detail hiding.

5.1 One Universal Standard

If one universal standard could be agreed upon, it would solve the tool interchangeability problem by standardizing it out of existence. CAD tool development would become implementation of the one standard tool set. The main problem with universal standards is that it is virtually impossible in areas of design methodology to achieve the consensus necessary for standard adoption. See for example the proposed Japanese alternative to VHDL discussed at the 1989 Design Automation Conference [30]. Even in the unlikely event that one design verification standard could be agreed upon, there are still inherent problems with attempting to standardize in areas involving human problem solving. At first glance, a universal standard seems to have numerous advantages such as elimination of designer retraining and model recoding. In practice, any standard tool set modeling weakness would cause errors associated with the weakness to appear in every circuit using the technology for which the weakness applies. Also, standardization would stop progress in design tool development. There would be no possibility of discovery outside the assumptions of the standard. The only possible progress would be improvements in simulation speed or design representation implementations of the one standard tool set. It is not difficult to imagine potential large improvements in current design methods and tools. What would happen if someone discovers a new conceptual framework for circuit simulation that is as accurate as spice but allows simulations to run as fast as current gate level simulations? There would be no way to test it. It could not be developed, and its adoption as the new standard would be prohibitively difficult.

The current primary standardization candidate, VHDL [7][16][4] which defines both verification semantics and language format has potential difficulties. VHDL is based on another standard (the ADA programming language) that itself is not being adopted as rapidly as had been hoped. Two additional problems were mentioned at the 1989 DAC panel on standardization [9]. The view that programming language abstract data types were not appropriate for hardware design was expressed. If hardware

design is programming, why bother building circuits, just write microprocessor programs. Problems with the VHDL timing model were mentioned. VHDL proposes the unlikely combination of a digital gate timing model with programming language task semantics. The standard precludes a design group from, for example, preferring the traditional system simulation style task independent time token movement model. VHDL makes sense as a design environment alternative and is worth developing if for no other reason than the popularity of its data type model in computer programming.

5.2 Source-Destination Specific Translation

Given that net list translators correctly map all features and semantic meaning such that designers trust the destination net list format veracity (see section 3), they are a workable solution to tool interchangeability. The main objection to net list translators has been the seemingly unbounded effort required in implementing one translator for every source-destination format combination. In practice, required programming effort is not a problem. There are only around 5 to 10 widely used formats. No one organization needs to provide translators between every combination. Any given tool provider only needs to provide "into" translators for its net list format or formats. All tool providers taken together provide the complete translator set. Programming effort can be minimized by using source code from a related finished translator as the stem or template for new translators. Translation from one source to one destination format is concrete and well understood. Of course, translator programming requires experienced programmers since tools with large user communities always seem to contain numerous complicated special purpose features. The expertise required to handle large programs and to understand net list features that are similar but not identical across different translators must be present. Treating program routines as opaque objects is probably not sufficient for the subtle semantic differences between formats.

5.3 One Universal Intermediate Form

Since it is unlikely that a universal standard will be adopted, another approach to tool interchangeability retains net list translation but attempts to simplify the translation task by using one common intermediate form. The EDIF [6][25] design interchange format is currently the main example of this approach. The possible advantage is that for each tool specific net list language only an "into" and an "out of" translator are needed. This idea has practical problems. Since the standard format must be completely general, it will handle the details of no format exactly, and even very close languages such as TDL [28] and its NDL dialect [17] become as difficult to translate as completely unrelated formats. The mapping problem that involves translating from a general intermediate language whose original source net list language is unknown into a specific format is difficult. What happens is that the syntax (lisp s-expressions for EDIF) stays universal but properties are created that are added by the "into" translator and by convention recognized and processed by the "out of" translator.

Since any standard must try to fit every case, no matter how comprehensive such a standard is, there will be cases that do not fit. The net list view part of EDIF is not comprehensive enough to represent all net list features. This causes the EDIF **property** and **userdata** constructs ([6], p. 2-313, 2-383) to be used with the effect of moving back to the source-destination specific net list translator scheme. Many net list languages support high to low and low to high bus ranges that do not start at zero, but EDIF arrays are unidirectional and always start at zero ([6], p. 2-11). Therefore some property is required. EDIF defines a simple delay calculator model ([6], 2-295, 2-69 through 2-72), but, at least in the case of semiconductor vendor ASIC libraries, most timing algorithms do not fit the EDIF model. The EDIF signal strength scheme allows a strength order relation ([6], p. 2-87, 2-356, 2-405), but this is a mere shadow of the complicated semantic and behavioral meaning of strengths in real simulators.

5.4 Frameworks

Frameworks do not solve the tool interchangeability problem by themselves since within the framework, either a universal standard or net list translation are required. Given that net list translation is still required, frameworks are valuable to the extent they provide conventions and shells to simplify learning and using tool interfaces. As long as the standard is flexible enough to allow both tool and design growth, frameworks are positive. Frameworks are detrimental to circuit design if they express the modern tendency to believe design problems are solved by simply providing a meta level design environment. Systems that implement abstraction and detail hiding often result in designs for which details are not well handled. If a framework is too narrow and standardized, it precludes utilization of a

tool's special features, and any framework that makes design verification model assumptions (such as the VHDL or EDIF standards) will probably be narrow in exactly this sense. For example, a tool's ability to handle general signal name synonyms requires that the framework have a synonym capability, but then that complicated and probably slow synonym mechanism must be present even for tools that do not use synonyms.

The Unix operating system itself may be the best CAD tool framework. Its computer interface model has gained wide acceptance during twenty years of use. It allows individual design verification tools to use the most appropriate net list formats and interfaces. [27] describes the Vivid system framework implementation based on Unix (see also [11][19]).

6. Examples

There are certain location within the electronic system design cycle at which the most benefit from alternative or supplementary tools can be gained. This section discusses net list translation at those points.

6.1 Graphic Editor Output Translation

Since schematic editor choice depends upon widely varying individual taste, it is possible for a project or company to mix a schematic editor from one tool set with verification tools from another. There are two common approaches to schematic level translation. One translates the raw schematic editor output which is usually expressed as geometric drawing commands. This approach has the advantage that it requires no intermediate processing on the source graphics editor system, but has the disadvantage that it is difficult to reconstruct the implicit semantic data that are assumed by the schematic editor. Examples are unexpanded macros, multiple fan-in signal nets, and properties that are removed by the first stage circuit compiler. Any translator that translates raw schematic data must effectively implement the source system's schematic interpretation algorithm.

Circuit compiler output is another entry point. This has the disadvantage that the source system circuit compiler may be slow but has the advantage that the output is in net list format and that the format is usually well defined. Examples are the NDL dialect of TDL that is output by the LSED schematic editor [17], and the Valid cmpexp.dat file [18]. If the source tool compiler is used to additionally expand a circuit down to built in primitives, a new tool can be used after coding only those few primitives in the destination tool format. After more models are available, unexpanded source drawings can be used to move hierarchical modules to the destination tool.

6.2 Translation at the Manufacturing Interface

Since net lists are the normal entry point to a manufacturing system, at least for ASIC vendors, any circuit that is not designed on the vendor's system requires translation into the vendor's verification tool format. Completed circuits can be translated a second time to move the circuit back into the original design system. The second translation both assures that the manufactured circuit exactly matches the original circuit and allows further verification at the system level. Semiconductor interface net list translations are among the easiest since the vendor requires that circuits be limited to its general purpose macro cells (mask patterns need to exist), and does not require timing translation since timing is built into the vendor's sign off simulation system. Translation back from a vendor's system is frequently more difficult since timing information including exact wire delays needs to be moved.

6.3 Translation for Mixed Gate and Behavioral Verification

Development of verification systems that combine behavioral and gate level simulation has increased the potential benefits from net list translation. Net list translators can improve system simulation speed. Simulators that mix accurate time modeling including timing checks and delay calculator delays, unit delay modeling, behavioral simulation, and programming language system control and debugging allow fast and accurate system level verification.

In one verification method, net list translation is used twice. Initially, unfinished parts or even subsystems are modeled behaviorally. The behavioral model may only implement the circuit's bus interface protocol or it may model complete functionality with a program. Once the gate level representation is completed, perhaps using some automated method, the previous incomplete behavioral model is replaced by the completed gate model. Usually the completed model will be the signed off

model translated back from the semiconductor vendor's system. Since by the time a part is ready for fabrication, complete system level tests would exist, those tests can be rerun with the previous behavioral model replaced by the exact timing gate model. The simulation will run more slowly but any problems in the gate level implementation can be found and corrected. The most common problems found at this point are timing problems and function misunderstanding between system and IC designers. For timing critical parts such as vector pipelines and bus interfaces, it may be necessary to run simulations with various combinations of behavioral and gate model instances in various locations to detect subtle timing interaction problems. Once system function is verified using exact timing, the same part can then be retranslated using unit delays so large system level simulation will run faster. A unit delay gate model with a large percentage of random logic will usually simulate faster than a behavioral model that implements full functionality. In fact, software simulation on a general purpose work station using mixed model timing detail can simulate faster than hardware simulators that use exact timing gate models for an entire system. The mixed level software method achieves at least as accurate timing and allows much better monitoring and debugging. Finally, behavioral, unit delay, and "real" delay simulation together can find circuit problems that one simulation method alone would not find.

6.4 Translation to Reduce the Tool Change Learning Curve

When an organization decides to change design tools and design methodology, the change over process can be traumatic. If net list translators are available to move old designs from the previous format into the new one and to move circuits back from the new format to the old, the transition can be made less difficult. For experienced designers the ability to work with familiar circuits simplifies learning the new system. Inexperienced designers can use the capability to move circuits from the new tool to the old to make a small design step and then move back to familiar ground to assure the small design step was correct. This has great psychological benefit for designers reticent about making the plunge into a new environment all at once.

6.5 Translation to Change Model Packaging

When a net list translation step is included in a circuit development edit-compile-translate-simulate loop, the translate step can be used to control model packaging and net list style within the destination tool. Translator configuration and mapping library files are normally used to map names (i.e. equivalent models named differently) and to allow user translation feature control for ambiguous or not exactly mapable features, but translation can also be used to change output style. Instance and vector wire splitting versus recombination can change destination net list part packaging. Parts can be removed from the output net list. Related instances can be concatenated into one instance with a different type for which the I/O ports from each instance are concatenated into one large instance I/O port list. Type or instance specific delays can be selected. One design group wanted the capability to leave certain vector widths unsplit but to split others depending on instance vector size since the different sizes were modeled differently.

7. Unix Framework Tools For Net List Translation

Various programs the are similar in function to Unix commands can be used in conjunction with net list translators to improve circuit design. The complicated output mapping described in section 6.5 can benefit from mapping library maintenance tools. One such tool builds library templates from source format net lists. Since translators can automatically determine type elements I/O ports from the input net list, translation libraries are optional, but the ability to start from default map templates simplifies map library preparation for those parts that need special mapping. Another useful utility is a library difference finder that is library syntax and semantics specific. It is like the Unix **diff** program but specialized to a given translator library format and can therefore deal with different orderings and codings that are equivalent in effect. The library difference finding program usually also has **lint** [13] like consistency checks.

Other programs are possible. A program to build library map entries for TTL part libraries on different CAD systems that were originally coded from different data book versions is useful. It reads both the source and destination language model I/O port lists and produces a translator map library that maps the port names. A good algorithm uses a dynamic programming string to string mapping algorithm

to match ports [29]. The port name matching problem is easier than matching natural language words since port names have no related letter combinations and frequencies [21].

Utilities that generate **make** program make files [8] which automatically execute circuit compilation and net list translation can be written. Normally, **make** does not work well when rules for making things can not be expressed as file name suffix transformations. For example on the Valid system, a program can read the drawing hierarchy and determines design tree dependencies. The program then writes a make file containing usually hundreds of literal rules, one rule for each module type. The make file generator utility can be run after each schematic editor change for completely automatic translation, but it only needs to be run if new user module or library types are added to some drawing. See the Vivid system description for a program that builds a make file from a manually prepared design description script [27]. The utility speeds up simulation preparation time since small circuit changes require only local translation and probably also local compilation.

8. Translation Speed

If an extra net list translation step adds significant time to the edit-simulate loop, it would reduce the advantages of tool interchangeability through net list translation. A net list translation step normally takes about twice the time it takes to read and tokenize the input net list language plus the disk write time to write the output format (see [20] for translation speed measurements). This duration is enough faster than even good circuit compilation (expansion or flattening) programs so that a translation step is usually not a noticeable time factor in the edit-simulate cycle. See [5] for circuit expansion speed measurements.

Recent work by Jones [14] attempts to show that circuit compilation can be executed must faster than has been thought. If this is true, the extra time used by a net list translation step would become a more significant factor. I believe those results have the following three flaws which suggest circuit compilation can not be significantly speeded up without functionality loss. The speed measure presented was the number of different modules (types) used plus the number of after expansion nodes (instances) per unit time ([14], section 6). Since this measure is I/O port number independent, it does not seem to correlate with expansion time for real circuits. Certainly an expansion of a circuit containing only modules with hundreds of I/O ports will be significantly slower than one in which all modules have less than 10 I/O ports. Since the schematic net list storage form used by the possibly faster method uses a "connected to" relation that is searched linearly and attached to the module type (p. 823), the nodes in the measured circuits are probably small. Second, all results assume no disk I/O. If in memory only operation were feasible, it would speed up processing, but some circuit description form needs to be stored on disk. If all in memory processing were practical, net list translation and other flattening algorithms would be speeded up by an equal amount. Third, the circuit model is so simplified that it may not be useful in practice. The extra details would probably make the DAG representation scheme unusable (p. 824). There is no fully qualified name storage. Imagine trying to debug with a simulator for which each net and instance lacks any name hierarchy but is identified by only a number. There is no multiple fan in (wired) net handling. This would at least require a more complicated DAG node connectivity representation scheme. There is no way to assign instance specific properties. This means no user assigned instance names. There is no way to traverse the net list to determine connectivity and fan-out for delay calculators without repeated relation searches.

There are actually situations where net list translation can reduce the total time needed to prepare a circuit for simulation. In the NDL net list format, any net renaming or bus merge-demerge connectors produce equivalence statements in the NDL net list. Modules with wide busses and bit rearrangement can sometimes contain five times as many equivalence statements as instances. A translator can execute an equivalence class computation and select a single base name that is then used in the destination format. The computation can be executed more quickly when processing individual models during net list translation than it can during circuit expansion. If all synonym names must be preserved in the flattened circuit, base name selection is not applicable.

9. Conclusions and Open Problems

Even though the argument for net list translators seems compelling, no design method can be adopted from arguments alone. The acceptance must depend on experience in design laboratories. Net list translators deserve further study and development to determine their actual utility. Meta level design methodology problems need to be studied. Many tool developers as opposed to framework developers believe that a new tool should have full flexibility in interface command choice, net list format, and functionality. They believe the extra design flexibility allows more rapid tool growth. They would claim tool growth depends on design form following function and would point out the Unix growth pattern successes. This view is contrary to nearly everything published in recent CAD literature (see [3] [4] for example). Studies that explain the discrepancy and determine which is superior would be interesting. The argument presented in this paper can be viewed as an argument against design standards. Since standards are obviously valuable in hardware interfaces, standard applicability guidelines would be useful. Finally, an argument against the whole idea of tool development was expressed at the 1989 IFIP conference [24]. Development as tool building as opposed to theory development and scientific problem solving was repeatedly criticized. It is not obvious to me what the alternative to building tools would be, but the anti tool building argument seems worth pursuing.

10. References

1. Allman, E. An Introduction to the Source Code Control System. Unix Berkeley 4.1BSD System Documentation, UC Berkeley, 1980.
2. Bell Telephone Laboratories, *UNIX Programmer's Manual*. Holt Rinehart, 1983, Vol. 1-5.
3. Daniell, J., and Director, S. W. An object oriented approach to CAD tool control within a design framework. Proceedings 26th Design Automation Conference, 1989, 197-202.
4. Dewey, A., and Gadient, A. VHDL Motivation. *IEEE Design and Test of Computers*, 5, 2(April 1986).
5. Diss, W. C. Circuit compilers don't have to be slow. Proceedings 25th Design Automation Conference, 1988, 622-627.
6. Electronic Industry Association. EDIF Electronic Design Interchange Format Version 2 0 0. Technical Report, 1987.
7. The IEEE, *IEEE Standard VHDL Reference Manual*, New York, 1987.
8. Feldman, S. I. MAKE - A Program for Maintaining Computer Programs. *Software Practice and Experience*, 9, 4(April 1979), 256-265.
9. Gajski, D., et. al. Why can't we agree on a single standard? (panel), Proceedings 26th Design Automation Conference, 1989, 49.
10. Genrad Corporation, Hilo III Hardware Description Language Reference Manual. 1985.
11. Hardwick, M., and Yakoob, N. Using a database and UNIX to author CAD applications. Proceedings IEEE ICCAD-85, 1985, 53-55.
12. Johnson, S. C. YACC -- Yet Another Compiler Compiler. Computer Science Technical Report 32, Bell Laboratories, Murray Hill, July 1975.
13. Johnson, S. C. Lint, a C Program Checker. Computer Science Technical Report 67, Bell Laboratories, Murray Hill, 1977.
14. Jones, L. G. Fast online/offline netlist compilation of hierarchical schematics. Proceedings 26th Design Automation Conference, 1989, 822-825.
15. Lesk, M. E. Lex -- A Lexical Analyzer Generator. Computer Science Technical Report 39, Bell Laboratories, Murray Hill, October 1975.
16. Lipsett, R., Marshmer, E., and Shahdad, M. VHDL - The Language. *IEEE Design and Test*, 3, 2(April 1986).

17. LSI Logic Corporation, NDL Reference Manual. 1987.

18. McWilliams, T. M., and Widdoes, L. C. SCALD: Structured computer aided logic design. Proceedings 15th Design Automation Conference, 1978, 271-277.

19. Mehmood, Z., Singer, D., Singhal, A., and Wu, K. A design data management system for CAD. Proceedings IEEE ICCAD-87, 1987, 220-223.

20. Meyer S. A data structure for circuit net lists. Proceedings 25th Design Automation Conference, 1988, 613-616.

21. Morgan, H. L. Spelling Correction in System Programs. *CACM*, 13, 2(Feb. 1970), 90-94.

22. Rappaport, A. Data Transfer and Chip Layout Ready IC for Production. *EDN Magazine*, 3 May 1984, 280-288.

23. Ritchie, D. M., and Thompson, K. L. The Unix Time Sharing System. *CACM*, 17, 7 (July 1974).

24. Ritter, G. (ed.). Foundations of Software Engineering - the Silver Bullet (panel), *Information Processing 89*, North Holland, Amsterdam, 1989, 953-956.

25. Roberts, M. CEDIF: A data driven EDIF reader. Proceedings 26th Design Automation Conference, 1989, 818-821.

26. Rochkind, M. J. The Source Code Control System. *IEEE Trans. on Software Engineering*, 1, 12 (December 1975).

27. Rosenberg, J. The making of Vivid. Proceedings 24th Design Automation Conference, 1987, 74-81.

28. Szygenda, S. A. TEGAS 2 - Anatomy of general purpose test generation and simulation system. Proceedings 9th Design Automation Conference, 1972, 116-127.

29. Wagner, R. A., and Fischer, M. J. The String-to-String Correction Problem. *JACM*, 21, 1(January 1974), 168-173.

30. Yasuura, H., and Ishiura, N. Semantics of a hardware design language for Japanese standardization. Proceedings 26th Design Automation Conference, 1989, 836-839.

Pragmatic ECAD Data Integration

G. Kaufman

Hewlett-Packard Company
Electronic Design Division
3404 E. Harmony Rd., ms 238
Fort Collins, CO 80525-9599
gjk%hpfcla@hplabs.HP.COM

Abstract

Data integration refers to the ability of applications to share data and the ease with which this data is shared. Data integration is one of the fundamental missing pieces of functionality in ECAD systems created with tools from multiple tool vendors. Current systems meet neither the user's nor the tool supplier's needs for sharing data.

A data integration solution is composed of five parts: semantics, syntax, storage, translation, and mechanism. There are only a handful of data objects transferred among ECAD applications. The key to obtaining data integration is to develop a data model that is common to all applications for each of these objects.

The goals for a data integration architecture are: use one data model, meet tool performance and resource requirements, provide an extensible model and format, accommodate tools from any supplier, provide a single data interface, and permit portability.

An ECAD data integration architecture to meet these goals consists of: a system context data model, a data model for each system data object, one format and access method for each object (the canonical form), and a data controller to control access to and translation of the data objects.

An initial implementation of this architecture might consist of a system context data model and data models and canonical forms for schematic capture symbols and simulation netlists. In the short term a very simple integration mechanism is acceptable; in the long term an object management facility holds the most promise as a flexible and general integration mechanism.

Introduction

This is a tutorial on data integration for ECAD systems composed of tools from multiple tool suppliers (heterogeneous ECAD systems). It explains what data integration is, why it is important, and what must be done to meet the tool users' needs. Simply put, data integration refers to the ability of applications to share data and the ease with which this data is shared. Data integration is one of the fundamental missing pieces of functionality in current heterogeneous ECAD systems.

We start, logically enough, by examining the user's and tool supplier's requirements for data integration. Next, we set the context by looking at the sort of data that is shared among ECAD applications. Once the context is set, we break the data integration problem into its five components and examine each in turn.

With this background, we then move on to set the requirements for an ECAD data integration architecture. We wind up with a proposal for an architecture that meets the requirements and a plan for an initial implementation of that architecture.

User needs

ECAD tool users really have only four requirements for data integration. Unfortunately, most heterogeneous ECAD systems meet none of them:

Users should be able to easily exchange CAD data among any arbitrary set of ECAD tools from any set of CAD tool suppliers.

Users should be able to easily exchange data among purchased applications and user created applications (an open system).

The transfer of information from one application to another should be at most a one step process (better is an automatic process), should be reliable, and should occur at an appropriate speed.

Users should have to enter data into the CAD system only once.

Tool supplier needs

The current common practice of information transfer via links from one application's semantics and syntax directly to another's causes five major inefficiencies in CAD tool development and maintenance:

Applications are tightly coupled. Changes in one application invariably require changes to all the links to that application. In any system of more than a few applications this leads to a maintenance nightmare, since applications change at arbitrary times.

Link reliability is poor. Most links between applications require semantic translation in the link. Semantic translation is a difficult and error prone task. The difficulty is in matching data that mean different things in different applications, especially since this meaning is often not clearly specified. This leads to convoluted translation schemes and subtle translation errors that are as likely to be found by the users as by the link creators.

Link development is forced to be in series with application development, since links cannot be designed until the applications are stable and available.

The addition of an application to a CAD system often requires several pair-wise links to be written. Because this is costly, the minimum number of links are usually written, thus limiting the flexibility of information interchange among applications.

Companies that choose to specialize on a small set of CAD functionality must spend considerable effort choosing and linking to all the ancillary tools their applications require, instead of concentrating on their area of expertise.

A data-centric view of a CAD system

To understand how these user and tool supplier needs can be met, let's first set the context by looking at data shared among a typical set of ECAD applications. Tables 1 and 2 were created by examining an existing heterogeneous ECAD system that was thought to be typical. Table 1 is a list of the links between pairs of applications and the data transferred over these links. Table 2 is a list of the information various applications need about the components they manipulate. Notice in table 1 there are a relatively small number of distinct pieces of information that are shared among multiple applications. The parts information in table 2, on the other hand, consists mostly of information that is different for each application; there is little part information that is shared among applications. From a data integration perspective, it is natural to think of the data as central objects that are shared among the applications. Figures 1 through 6 show this view for some of the data objects identified in the tables. These figures are a vision of ideal data integration. They should be contrasted with figure 7, which is a diagram of the system used to derive the data in tables 1 and 2.

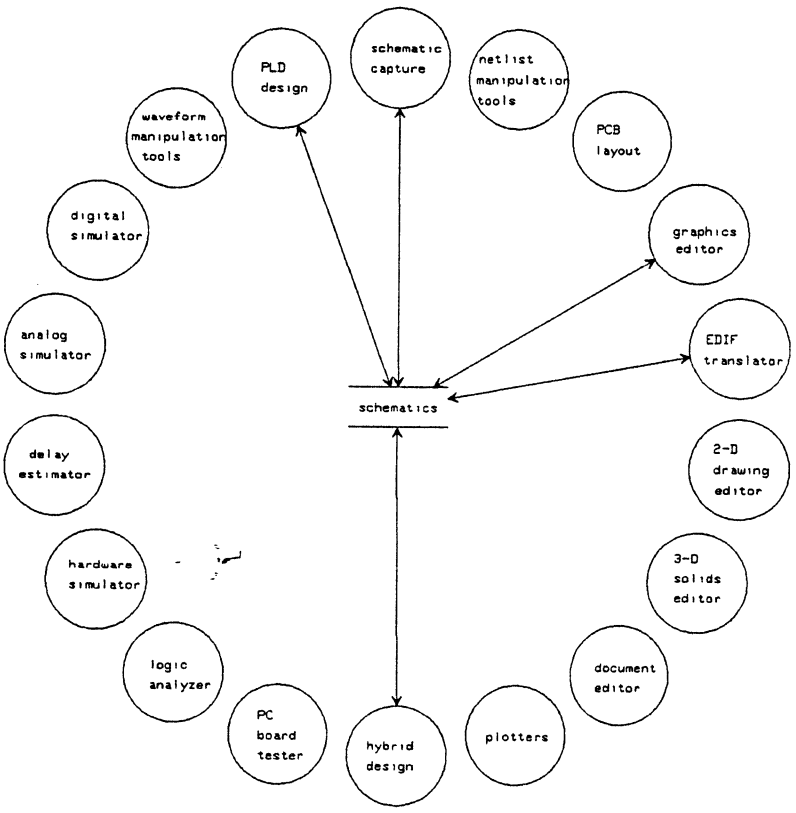
The chunks of data we have been considering may be thought of from an ECAD system perspective as "atomic data objects", that is, they are passed among applications as whole pieces and are never

transfer	file name	information
from schematic capture to:		
PCB layout	design file	netlist
	parts file	pinout
EDIF netlist	EDIF netlist	netlist
EDIF schematic	EDIF schematic	schematic
analog simulator	circuit file	netlist, waveform
digital simulator	netlist file	netlist
	stimulus file	waveform
	control files	simulator control
	memory model files	memory patterns
hardware simulator	stimulus file	waveform
	capture file	waveform
PLD design	netlist file	netlist
PC board test	PCF file	waveform
to schematic capture from:		
PCB design	delta design file	back annotation
EDIF schematic	EDIF schematic	schematic
analog simulation	circuit file	waveform
digital simulation	capture file	waveform
hardware simulation	capture file	waveform
	memory contents file	memory patterns
PLD design	PLD connection file	subcircuit connectivity
	PLD definition files	gate connectivity
from PCB layout to:		
PC board test	board config file	netlist
2-D drawing editor	drawing file	board blank
to PCB layout from:		
EDIF netlist	EDIF netlist file	netlist
2-D drawing editor	design file	board blank
from many tools to:		
2-D drawing editor	HPGL file	drawing
document editor	HPGL file	drawing
plot spooler	HPGL file	drawing

TABLE 1. Design Data Transfers

application	information
PCB layout	package physical pinout pin, gate swap silkscreen graphics autoinsert parameters
schematic capture	symbol graphics pinout
digital simulation	digital functionality
analog simulation	analog functionality
PLD design	PLD composition
PC board test	test values
2-D drawing editor	package outline

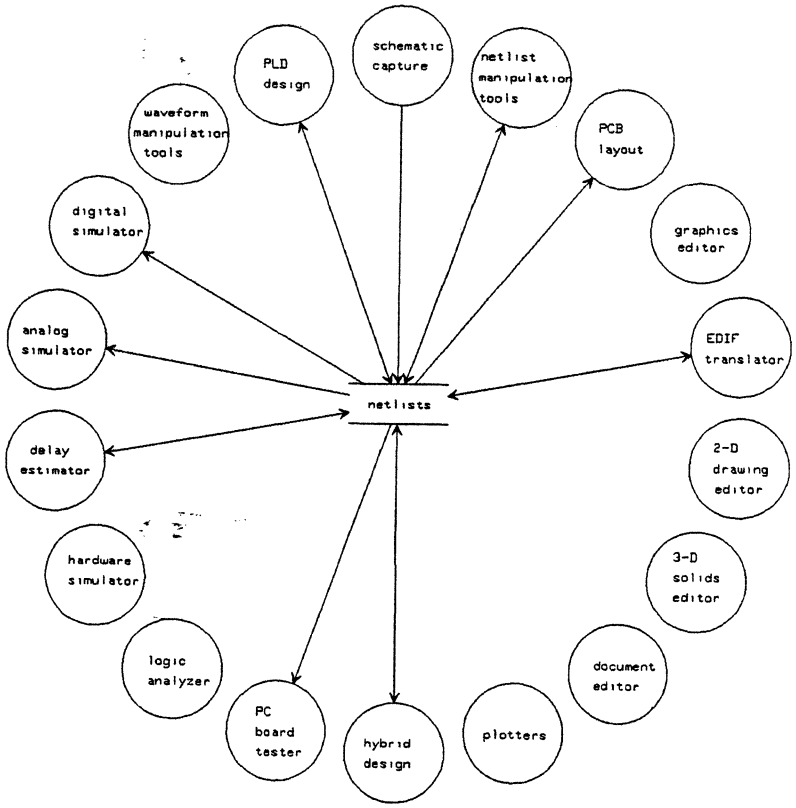
Table 2. Parts Data Transfers



dca/cfi/arch/pdi/figl_d

Figure 1

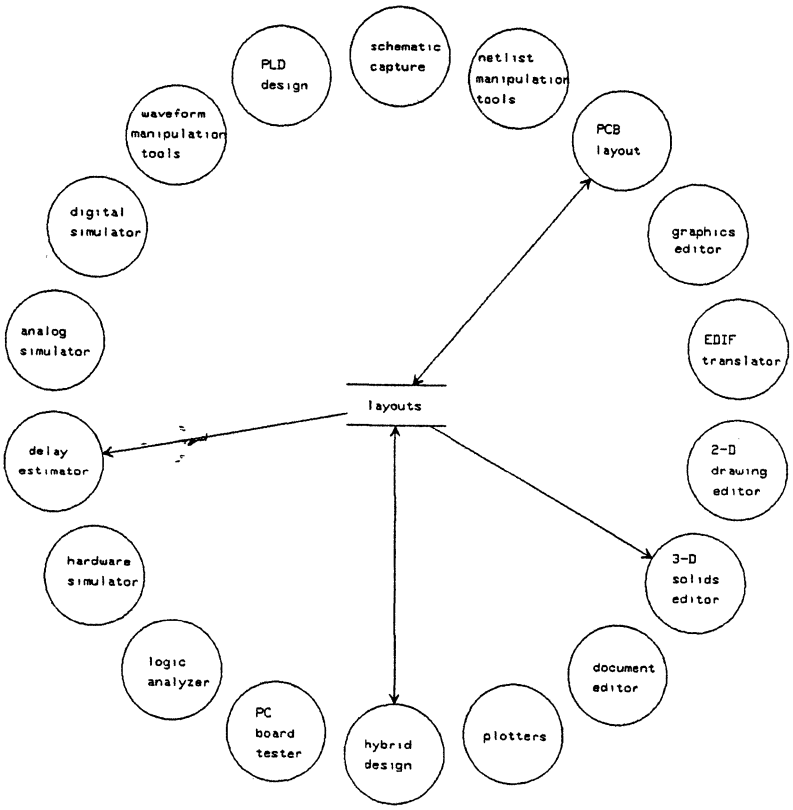
gjk
11/28/89



dca/cfi/arch/pdi/fig2_d

Figure 2

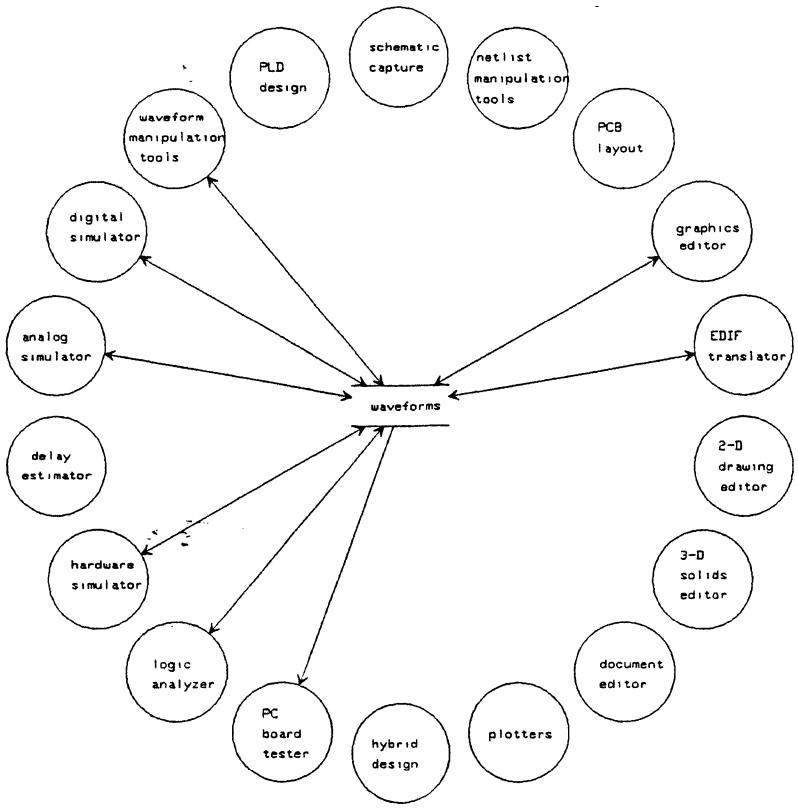
gjk
11/28/89



dca/cfi/arch/pdi/fig3_d

Figure 3

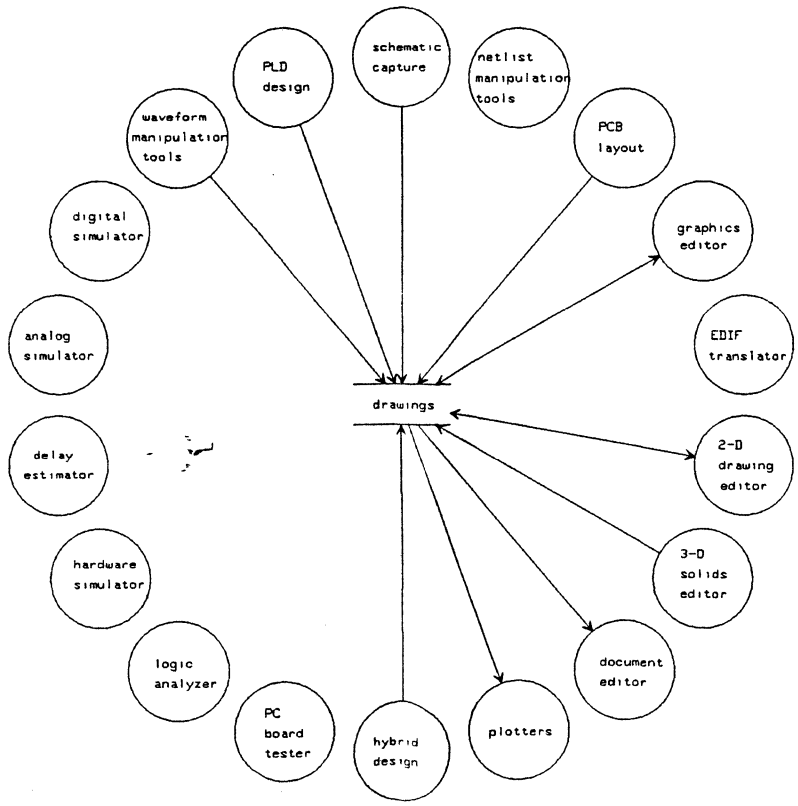
gjk
11/28/89



dca/cfi/arch/pdi/fig4_d

Figure 4

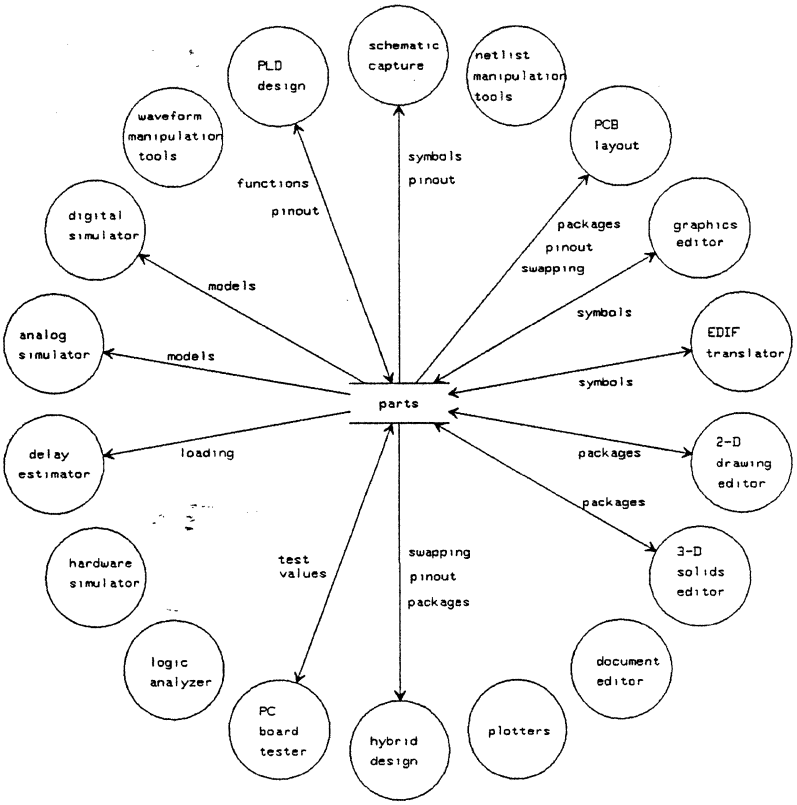
gjk
11/28/89



dca/cfi/arch/pdi/fig5_d

Figure 5

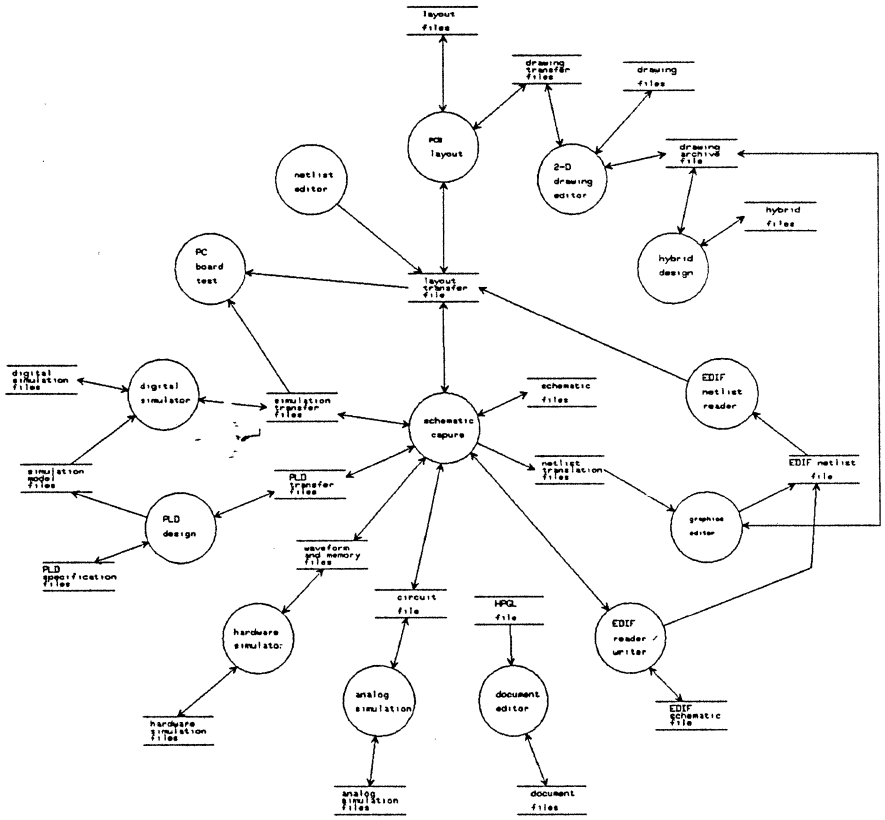
gjk
11/28/89



dca/cfi/arch/pdi/fig6_d

Figure 6

gjk
11/28/89



dca/cfi/arch/pdi/fig7_d

Figure 7

gjk
12/1/89

decomposed into smaller pieces outside an application. They may, however, be combined into larger objects; this is likely to be useful for managing the information.

The pieces of the puzzle

It is useful to break the data integration problem into five pieces:

semantics: Semantics (or data modeling) is concerned with the meaning of data. When we say a part number uniquely identifies a part and this part has a package that contains gates and has pins, we are talking of the semantic definition of a part.

syntax : Syntax (or format) is the representation or expression of data. When we say a part number is an ASCII string of at most 9 characters, we are talking of the syntax of a part number.

storage : Storage is the form in which data is saved and the way it is accessed. An object oriented data base, a file, and a relational data base are examples of forms of storage. Each has its own access mechanism: OSQL, read/write, and SQL.

translation: Translation is the process that changes data from one form to another. Changing an EDIF netlist to a Spice deck is an example of translation.

mechanism: Mechanism is the method used to get the proper shared data to an application in the form it needs, when it needs it. The current data integration mechanism for most heterogeneous CAD systems is manual invocation of translators.

System designers, when they think of improving data integration, often tend to focus on the mechanism for some reason (perhaps because that's where all the new, whizzy technology is), but most of the problems are caused by the first two pieces. Indeed, without a shared meaning for the shared data, a common syntax and a transfer mechanism are of no use at all.

Procedural interfaces for data integration (as proposed by the European CAD Integration Project [1] and the CAD Framework Initiative [2]) are an example of a data integration solution that contains elements of all of the above five components. Procedural interfaces contain some semantic information, e.g., the call GetCellInstName(id) implies the existence of a cell instance, and that a cell instance has a name. They contain some syntactic information: the specification for GetCellInstName (id) contains the data types for both the cell name and cell id. They are obviously an access method: GetCellInstName takes a cell id and returns the cell's name. They provide translation from the source tool's internal data representation to an external representation. And finally, they use an interprocess communication mechanism to transfer data from one tool to another.

Let us examine each of these five components of data integration in turn.

Semantics

The semantics of data are described by a data model [3]. Theory tells us it is impossible in the general case to translate data from one data model to another. Experience bears this out. (See appendix A for some simple examples.) Currently, in order to exchange data among existing applications from different sources, link creators are forced to make semantic translations, since each application's data semantics were developed independently of the others. These translators work only under limited conditions and are very fragile.

It is axiomatic that sharing data requires one model of that data, one meaning for that data. A data model that is common to all applications is the foundation upon which data integration is built.

Data integration, by its very nature, requires data coupling among applications. If the model of data shared among applications is required to change, then all the applications that use the data must change. Data integration demands a high degree stability in the definition of the shared data. The design of data

models is in many ways analogous to the design of code; many of the same techniques used to design code may be profitably employed in the design of data models that are robust and meet the needs of the users of the data. In particular, frequent design reviews by tool designers and consideration of a representative sample of tools are important.

Syntax

Unlike semantic translation, syntactic (format) translation is relatively straightforward. Presently, format translation is a necessary part of data interchange in heterogeneous CAD systems, since, just as with semantics, the formats used by applications were specified independently of each other.

Although the elimination of this format translation by defining a single syntax for data shared among applications is not, strictly speaking, a necessity, it certainly is desirable. Format translation provides no benefit to the user, but does extract a cost. Translators consume processing time and require maintenance. Multiple formats of the same data are one form of data redundancy and thus require synchronization to keep all copies current. There must also exist some mechanism to invoke the translators to create the correct formats of the data.

With syntax, unlike semantics, conforming applications may evolve toward a common syntax. A common syntax may be defined, and in the short term translated to an application's format. Then, at a convenient time, the application developer may change the application to operate directly from the common format. As with semantics, the syntax must be stable to prevent forcing changes to applications that use the format directly.

An often overlooked, but nonetheless desirable, characteristic in a common format is extensibility. An extensible format allows the possibility of adding new information to a data object without the need to convert old data to the new format. Coupling rational, stable data models with stable but extensible formats has the potential of allowing new revisions of applications to read old data without translation and also allowing old applications to read data from new applications without translation. I say potential, since changes to the data model will require data translation, but extensions to the data model to provide new application functionality may well be accommodated without data translation. Since data migration from one revision of a tool to another has often been a significant problem for CAD users, extensibility should be considered by system designers.

Storage

The phrase "common data base" is often used as a surrogate for improved data integration. This is confusing, since it implies the use of only one storage method to store all CAD information. Although this is a desirable goal, and one worth striving to achieve, it may not be possible, given the diversity of ECAD information and the different performance and space requirements of the different data objects. The ability to select a storage method to match the requirements of a data object and the applications that use it is one of the most powerful tools in the system designer's arsenal. If the CAD system designer is restricted to one storage method for all data objects his ability to design the highest performance systems is severely constrained.

A special purpose data store, for example, is well suited to the storage and access requirements for applications that use waveforms. A grammar in an ASCII file may be well suited to the requirements applications have for netlists. And a relational data base seems a good choice for storing the part information used by schematic capture and PCB layout tools (although perhaps not so good for all part information, such as simulator models).

It is essential that there be one and only one access method for each data object. This allows, for example, a simulator to read a netlist from any source (a netlist editor, a schematic capture package, a PLD synthesis tool). Although there may be different access methods for different data objects (the access method for a schematic may be different from that for a digital simulator model, for example), it is both useful and feasible to strive for one access method for all design data. A consistent and uniform interface

to design data can decrease development time and design errors, since the designer need understand only one interface. It provides implementation flexibility and a level of isolation from the storage of the data. If the interface reflects the semantics of the data, not just the format, design errors will be reduced even further by reducing misunderstanding of the meaning of data and incorrect accesses.

Translation

The translation of data from one form to another is always undesirable. Given CAD systems will always contain applications that do not conform to a data interchange standard, a data integration architecture must allow and support translation of data to have an open CAD system.

Nevertheless, the need for semantic translation (the transformation of the meaning of data) must be eliminated for conforming applications. Semantic translation is at best difficult and error prone. Syntactic translation (the transformation of the format of data), although relatively straightforward, consumes resource for no useful purpose and should also be eliminated where possible.

Mechanism

There are several integration mechanisms that may be used to get the correct information, in the correct form, at the correct time, to an application. A messaging service, an object management system, a distributed, object oriented data base, a central data dispatcher, a data manager, or manual invocation of translators will all work. The key is to have one mechanism all applications in a system can use. This implies an application may use the mechanism via encapsulation. That is, the integration mechanism must be able to be used by an application without changing the application's code. Encapsulation also gives the application developer some flexibility in how and when he modifies his applications to use the integration mechanism.

An integration mechanism does not have to be complex. A universal manual invocation scheme is an acceptable first step. More esoteric mechanisms, such as object management systems, provide useful features not found in simpler systems, but are still in the early stages of development.

Data management

Data integration and data management are two separate and distinct functions, both of which are required in CAD systems. Since these two functions are sometimes confused with each other, let's take a minute to look at data management and see how it differs from, yet complements, data integration.

Data integration is the ability of one application to share data with another. It comprises the five elements enumerated above. Data management is the control and organization of design information. It comprises the following functions:

concurrency control

security control

process control

data organization (configuration control)

revisioning / versioning

In a good architecture, data management and data integration functions are kept separate from each other and do not overlap, but cooperate to provide the user a powerful and flexible environment to organize, control, and share CAD data. In particular, by making data organization and revisioning / versioning part of data management and not data integration, the user gains flexibility. He is free to organize the data as he sees fit, and to add data from his internal tools to the data organization.

Even with a good data integration architecture, there are three forms of data redundancy the architecture must be able to handle. This responsibility often falls on the data manager. They are: multiple formats of the same information, multiple copies of the same information, and semantic overlap. The first two are self-explanatory and well understood (if not easy to solve); the third is often forgotten about, but from a data integration perspective is important. For example, connectivity information is contained in at least three data objects: the schematic, the netlist, and the board layout. When a change is made to one, the others may or may not still be synchronized, depending on what was changed in which data object. This is mainly a versioning problem and part of the data manager requirements. It is important a CAD system handle all three forms of data redundancy, since they cannot be eliminated.

Architecture goals

Now let's reduce the above discussion into a set of goals for an architecture for data integration in a CAD system:

One data model. Semantic translations must be eliminated for conforming applications.

Meet application performance, efficiency, and resource requirements. The architecture must not constrain the tools in these areas.

An extensible model and format. Maximum flexibility must be provided to improve an application's functionality without requiring data migration and without changing any other application or link. The user must be able to add his own data to a data object.

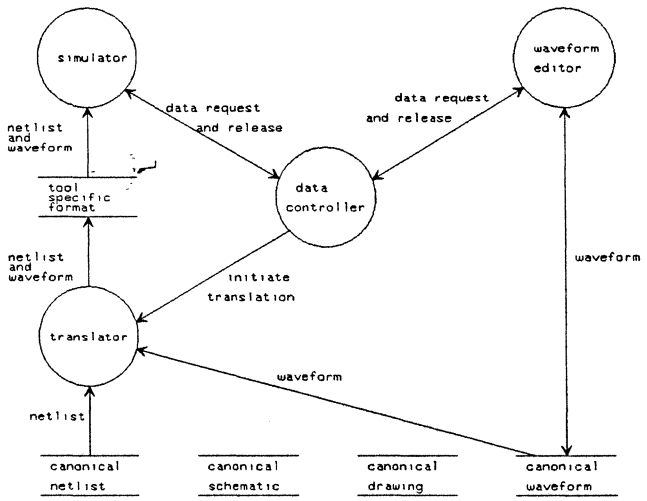
Accommodate applications from any source. Users want to integrate tools from every CAD supplier as well as their internal tools.

A single interface for data access by programs that captures the data's semantics. This improves development efficiency and reduces design flaws.

Portable. The data integration architecture must be able to be used on a wide variety of operating systems and machines.

An architecture for data integration

Figure 8 is a data flow diagram [4] of an architecture that achieves a balance among these sometimes conflicting requirements. Data shared among applications is stored in canonical forms. These canonical



dca/cfi/arch/pdi/fig8_d

Figure 8

gjk 12/21/89

forms comprise a single data model, format, and storage method. Each type of data may have a unique canonical form, but each instance of a type has the same form. The applications request access to the data they need through a central data controller. The left path of figure 8 shows the data controller initiating a translation process that packages the required data for the simulator in the correct form (by combining and translating the canonical forms) before releasing it to the simulator.

The data controller is also responsible for initiating a process to unpack data from tools back into the canonical forms. The right path in figure 8 shows applications (in this case a waveform editor) may directly access the canonical forms of the data, after they have been released by the data controller. Direct access is preferred to translation, of course. An application may also use some combination of the two access methods. Notice the data does not flow through the controller; the controller only receives requests, triggers translation, and permits access.

The data controller may be any of the mechanisms discussed above. In the short term, only the simplest of functionality is required. The key to data integration architecture is not, however, the mechanism. The mechanism is secondary. Data integration succeeds or fails based on the data models, formats, and storage forms that make up the canonical data forms. This architecture is not sophisticated code and new technology. It is insightful and precise definition, partitioning, and design of the shared data.

Data modeling for ECAD system data integration needs to be tackled at at least two levels. Figure 9 shows a high level view of ECAD data in IDEF1X notation [3, 5]. It's a much simplified example of what a complete ECAD data model might look like. This view shows the system level atomic data objects as individual entities (the boxes). Its purpose is to both enumerate the data objects and show the relationships among them. The data objects should be as disjoint as possible, that is, they should have few relationships to other objects. This provides maximum isolation of one object from the others and reduces the chance that a change to one object will cause changes to others. The relationships must be precisely defined and the properties they represent must be consistent among the entities to achieve data integration across data objects. This model sets the ECAD system data context, but although it is necessary, it is not sufficient.

Figure 10 is an example of a data model diagram for a netlist. This diagram is replicated in appendix B along with its associated data glossary. The diagram and the glossary together comprise the data model. Both are essential for a complete data specification. Figure 10 is an expansion of one entity from figure 9. Data models such as this are the definitions of the data objects. They are required to achieve data sharing of objects among tools. Notice in the lower left of the diagram the two links to the parts data model. In this case, in order to achieve good data integration there must be agreement on the part_name and port_name properties between the parts data model and the netlist data model.

Both the ECAD system model (figure 9) and the object models (figure 10) are important for data integration and open systems. The system model is needed to show the relationships among the data objects; these relationships must be well defined to achieve data integration across data objects. The object models are needed to precisely define the data shared among the applications; these definitions are necessary for an open system as well as for data sharing among conforming applications.

In summary then, the ECAD system data model sets the data context. It:

Defines the system's atomic data objects.

Shows the relationships among the atomic data objects and defines the properties shared among them.

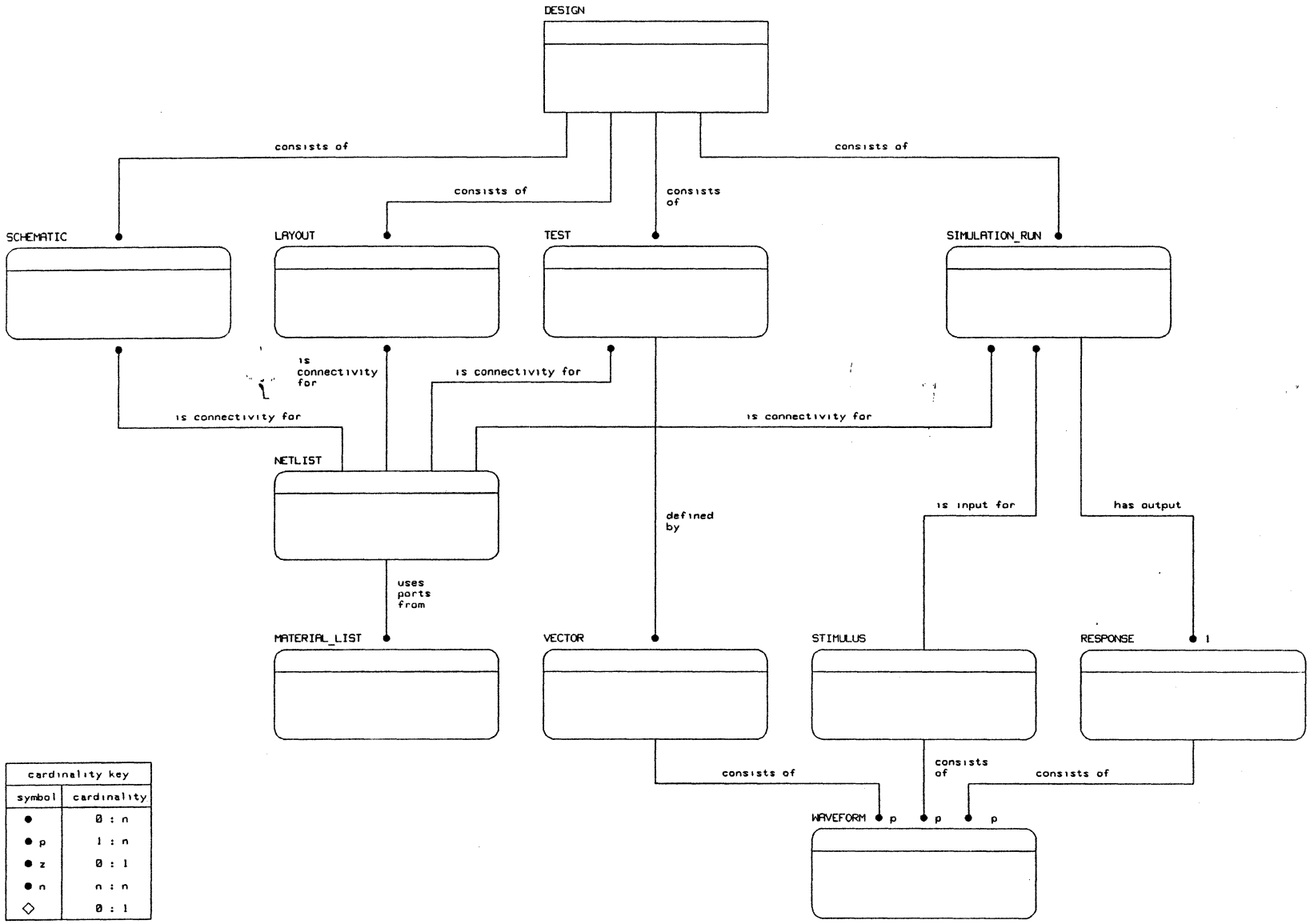
Provides a context for the object models.

Helps to identify and reduce semantic overlap among data objects.

Is necessary documentation for tool integrators and users.

The object models:

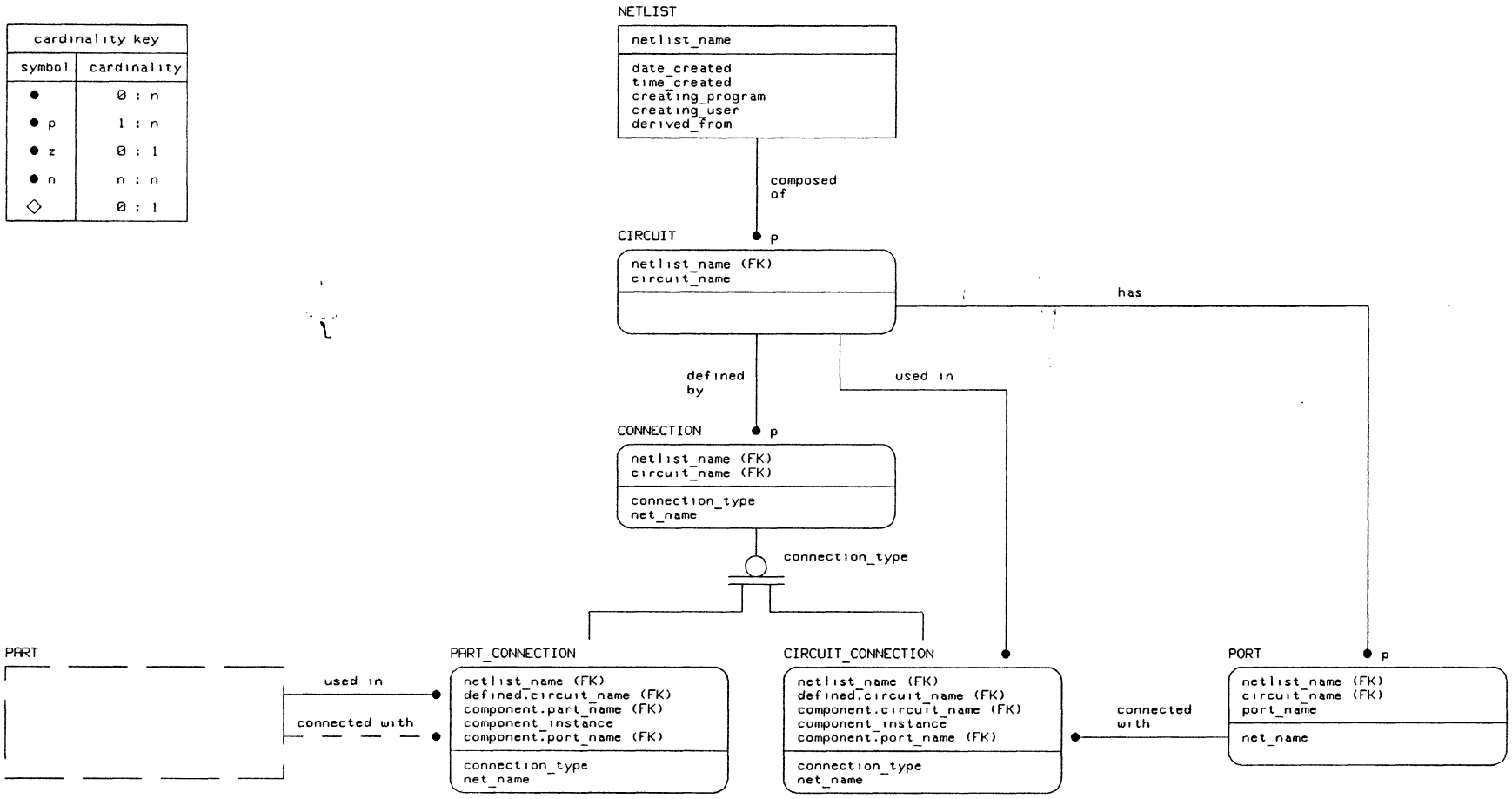
Precisely define the atomic data objects shared among applications.



dca/cfi/arch/pdi/fig9_d

Figure 9

gjk
12/18/89



dca/cfi/arch/pdi/figl0_d

Figure 10

gjk
12/15/89

Verify the relationships among the data objects identified in the system model.

Provide common definitions for the shared properties that create the relationships among data objects.

Help to identify and reduce semantic overlap among data objects.

Are necessary documentation for tool integrators and users.

In this architecture there is only one syntax and access method for the canonical form of each data object, but the syntax and access method may be different for each object. That is, waveforms must all use one format and one access method, but this format and access method may (if necessary) be different from the format and access method used for netlists. The format and access method for each object should be chosen to match the needs of that object.

Also in this architecture, data transfers always go through the canonical form. If two tools both operate directly off the canonical form, they may share data with no translations. If one tool operates directly off the canonical form and the other has its own form (model / format / storage), then one translation step is necessary. If both tools have their own unique forms, then two translation steps are necessary, from unique form 1 to the canonical form and then from the canonical form to unique form 2. This double translation may end up being slower than a direct translation, and so needs to be watched carefully to make certain user needs for translation speed are met.

In return for the development investment to design the canonical forms and restricting data to flow through them, several of benefits are gained:

The data translations among tools may be incrementally reduced. Tool developers have the choice of continuing to run off unique data forms or of running directly off canonical forms (or some combination). The change from unique forms to canonical forms may occur at the tool developer's convenience.

Canonical forms provide a uniform and consistent method of packing and unpacking data to and from non-conforming applications. They also help identify semantic discrepancies and semantic translations.

Coupling among applications is reduced over pair-wise links, since a change to one application's unique form requires changes only to one translator and not all the other translators and applications that use the data object. Well thought out data models for the data objects that meet the needs of many applications are also likely to be far more stable than data models developed for only one application.

The effort to add a new tool is decreased. With a drawing data object, for example, adding support for a new plotter or adding a new documentation tool allows any tool that creates drawings to send drawings to that plotter or documentation tool.

The possibility of increased and serendipitous data sharing is enhanced, since any tool that manipulates a data object may now share data with any other tool that manipulates that data object.

With design data stored in canonical forms with extensible formats, the data migration problem is reduced for a large class of tool changes.

How to do it

An incremental approach to the implementation of this architecture is best. It provides benefit with minimum risk and minimum investment. This is important, since the architecture is unproven; a real evaluation of the cost and benefits with as little investment as possible is only prudent. Also, small data models are much easier to create than large models. An incremental approach gives the opportunity to apply the lessons learned from the integration of one data object to the next. On the other hand, enough must be done to make a significant dent in the problem. Consider a five part initial project:

Develop the ECAD system data model to set the system data context.

Develop a netlist data model and canonical form and use it as a data sharing path between schematic capture and simulation.

Develop a schematic symbol data model and canonical form. This is the natural first step in sharing component data and is related to the netlist data.

Develop a simple, common mechanism to coordinate the sharing of the data objects identified by the system context data model.

Begin working toward a general object management facility as a long term data integration mechanism and the core of an ECAD framework.

These five steps represent achievable goals for the near term. Although they may be criticized as being too small a step forward, achievement of these goals would fill an existing need and represent a significant and useful improvement in the data integration of heterogeneous ECAD systems. They also provide a stepping stone for solving the more complex problems.

Acknowledgments

Thanks to my colleagues at the Electronic Design Division of the Hewlett-Packard Company who participated in our many ad hoc discussions of CAD data integration. The analysis and proposal in this paper grew from these discussions and are an amalgam of the ideas of many people. Bias in the selection of ideas, and errors, are mine.

References

1. "SPI version 2.4", Cockx, Johan, European CAD Integration Project, 18 Sept. 1989.
2. "The CFI DAC '90 Scalar Netlist Procedural Interface Specification As revised at the 11-9-89 CFI Meeting", Goldfein, Arny, CAD Framework Initiative, 11-22-89.
3. "The Database Book", Loomis, Mary E. S., Macmillan Publishing Company, 1987, Chapter 3 (IDEF1X tutorial).
4. "Structured Analysis and System Specification", DeMarco, Tom, Prentice-Hall, 1978, Part 2.
5. "MISS - Integrated Information Support System Information Modeling Manual (IDEF1X)", ICAM Project 6201, Subcontract #013-078846, USAF Prime Contract #F33615-80-C-5155, prepared by D. Appleton Company, Manhattan Beach, CA.

Appendix A: Translation Between Data Models.

To understand some of the difficulties in translating from one data model to another, consider this simple example. Let us imagine two relational data bases, one maintained by everyone's favorite governmental agency, the Internal Revenue Service, the other maintained by your company's personnel department. For simplicity we will consider only one table in each database. Table 3 lists the columns in the corresponding tables of the two databases.

IRS database	Company database
social_security_number (key field)	employee_number (key field)
first_name	name
middle_initial	
last_name	
day_of_birth	birth_date
month_of_birth	
year_of_birth	
job_classification	job_classification
income	income

TABLE 3. Two database tables

Since both of these tables contain similar information about people, one might expect it to be a simple task to transfer information from one database to the other. We will show this is not the case because the tables are implementations of different data models. In so doing we will also illustrate some of the information about data a good data modeling methodology will specify.

Let's start by trying to determine if information about an individual exists in both databases. Since names are not unique, each database uses a unique identifying number as a surrogate (the key field). But these numbers are different, and there is no mapping between them. A social security number uniquely identifies an individual in the IRS database and an employee number uniquely identifies an individual in the company database, but one cannot be determined from the other. We are defeated before we begin. (This is not a specious example. Its equivalent exists in heterogeneous ECAD systems.) One obvious way of approximating an alternate key is to concatenate fields (for example, name and birth date) to obtain a surrogate that has a reasonable chance of being unique, but the reliability of such an alternate cannot be guaranteed.

Now let's look at the other fields. The IRS database stores name information in three fields, the company database uses one. Transfer of a name from the IRS database to the company database seems simple, concatenate the three name fields and store the result in the one company name field. But what do we do if the IRS database uses 15 character fields for the first and last names and the company database uses a 25 character field for the full name?

Transfer of a name from the company database to the IRS database poses a greater challenge. Now we must in some way parse the full name into its component names. Parsers to solve this problem tend to be complex, and experience teaches us there are more forms to translate than are dreamt of by designers of translators. How many forms of names are there to recognize and parse? "Gerald J Kaufman" is simple enough, but what about "Gerald J. Kaufman", or "Kaufman, Gerald J". Did you consider "Gerald John Kaufman"? (Notice in this case there is an information loss in going from the company database to the IRS database; the middle name is lost and only the middle initial remains. In this case a translation to the IRS database and back to the company database cannot preserve the original data.) What do you do with "Gerald Kaufman, Jr."? Will your parser handle "Elizabeth Yanko-Kaufman" correctly?

Birth_date is an example of a translation that may be easy or difficult, depending on the specification. If birth_date in the company database is specified as an ASCII string of the form mm/dd/yy, left justified,

no leading zeros (i.e., 1 Jan. 1990 is stored as 1/1/90), then the parsing and translation of the date to the IRS database is simple. But if any human readable form of date is allowed, we must once again try to anticipate all the possible ways dates can be written.

Job_classification looks trivial. Let's suppose that both are enumerated types encoded as two digits. So just transfer the value. Well, what if the IRS uses job classifications such as:

00 - unemployed
01 - clerk
02 - secretary
03 - engineer
04 - manager
05 - self employed

and so on, while the company uses:

00 - assembly worker
01 - shipping worker
02 - receiving worker
03 - electrical engineer
04 - mechanical engineer
05 - first level manager

The encoded values do not correspond to the same thing. Further, there is little correspondence between the categories in the two databases. A rough translation may be possible in some cases (the more specific engineer categories in the company database may be translated to the one engineer category in the IRS database, for example), but even then information is lost.

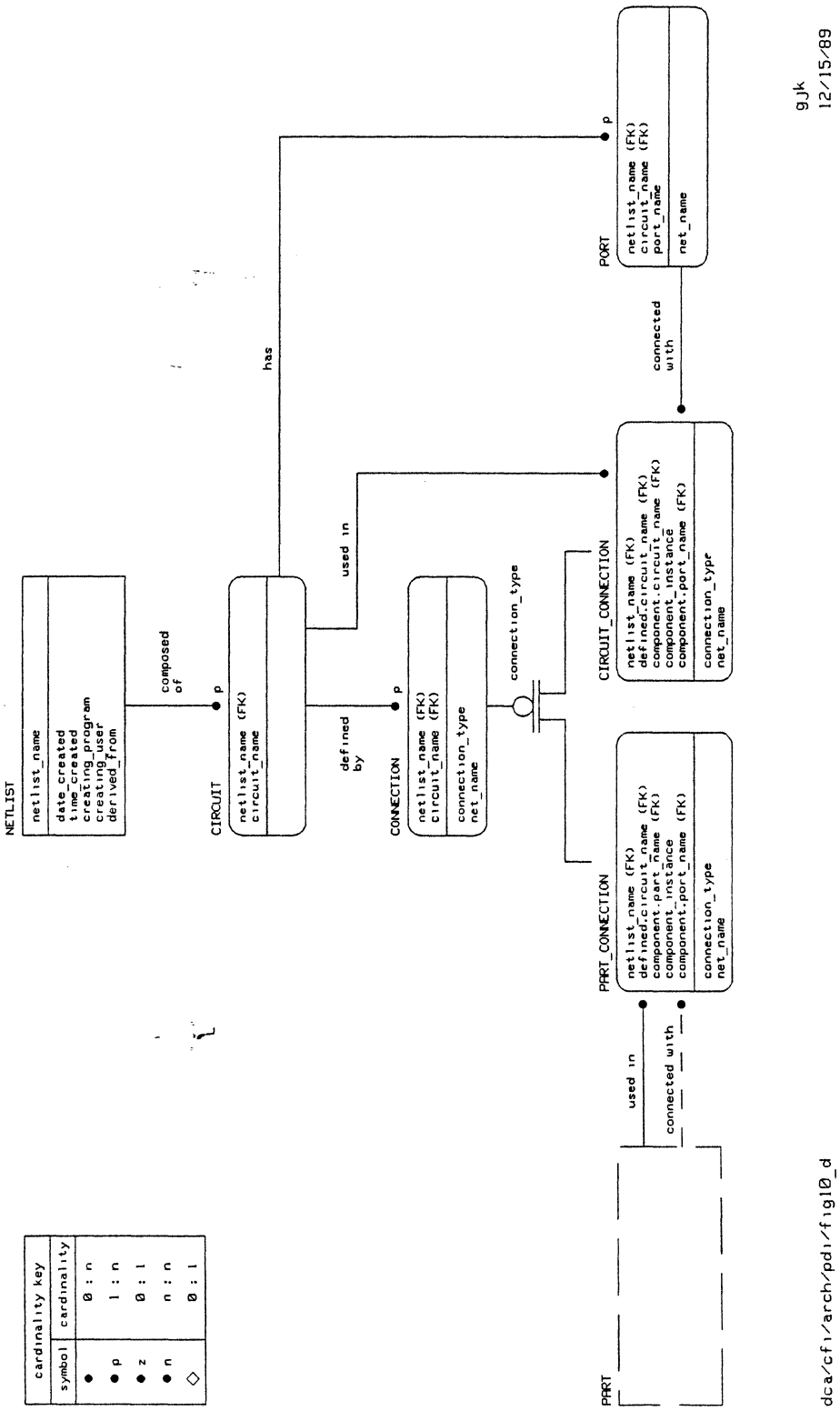
One last example. Assume the income fields have the same format (32 bit positive integers representing cents, say). Now there is no translation, just a transfer of the value. But in the IRS database "income" means total income from all sources (salary, savings interest, stock dividends, and so on), while in the company database "income" means total income paid by the company (salary, bonuses, profit sharing, and so on). Although we can transfer the value, it is an error to do so. Indeed, translation is not possible, since the information required by the translation algorithm is not available. It is quite common in existing heterogeneous ECAD systems to encounter two databases with different field that have the same name but different meanings.

The above examples are all illustrations of problems encountered in translating single valued data items from one data model to another. These problems can be identified only if the data models contain sufficient information about the meaning and format of the data they describe. In IDEF1X [3, 5], this information is contained both in the diagram and, more importantly, in the glossary. Data model diagrams do not contain sufficient information to completely describe data; text descriptions are essential. In the netlist data model in appendix B, for example, the diagram can be almost completely constructed from the glossary, but the glossary cannot begin to be constructed from the diagram. Both are essential to a complete specification of the netlist.

Beside the problems of translation of single valued data, there is another class of problems involving the structure of data. A complete explanation is beyond the scope of this appendix, but consider this simple example:

Assume the IRS database contains the value of a person's salary, and further, it contains the salary for each year a tax return was filed (salary for 1986, salary for 1987, ...). Now let the company database also contain the value of a person's salary, but only the value of the current salary. So even if the meaning and format of these two data items are the same, there are translation difficulties, since the IRS database contains a list of values, but the company database allows only one value. There are may other (and far more difficult to translate) structural differences possible between two data models. The ambitious reader may wish to consider the problem of translating a netlist from the CFI "5 box model" [2] to the netlist model in appendix B of this paper.

IDEF1X diagram



NETLIST properties

A netlist describes the connectivity of a set of parts from the parts library. It is organized as a hierarchy of circuits. Each circuit consists of other circuits and parts. Circuits and parts have ports, which are connected to the ports of other circuits and parts in nets. A net is the set of ports that are connected together.

netlist_name	meaning:	A name to uniquely identify a netlist.
	format:	standard name format.
date_created	meaning:	The date the netlist was created. The value is from the system clock on the machine that ran the process that wrote the netlist.
	format:	standard date format.
time_created	meaning:	The time the netlist was created. The value is from the system clock on the machine that ran the process that wrote the netlist.
	format:	standard time format.
creating_program	meaning:	The name of the application program that created the netlist.
	format:	standard program id format.
creating_user	meaning:	The UNIX user id of the user that ran the creating program.
	format:	standard user id format.
derived_from	meaning:	The object this netlist was derived from. This is usually a schematic name. For netlists created by a netlist editor, the value is null.
	format:	standard name format.

CIRCUIT properties

A circuit is a component of a netlist. It is composed of other circuits and parts connected to nets. It describes the connectivity of its components by enumerating the nets the ports of each of its components are connected to.

netlist_name	see NETLIST.	
circuit_name	meaning:	A name to uniquely identify a circuit.
	format:	standard name format.

CONNECTION properties

A Connection describes the connection of a port of a component of a circuit to a net in that circuit. A port may be connected to one and only one net.

netlist_name	see NETLIST.	
circuit_name	see CIRCUIT.	
connection_type	meaning:	The type of component being connected. A connection may be to either another circuit in the netlist or to a part from the parts library.
	format:	enumerated type represented as a 7 character ASCII string. Allowed values:
		'part' => this connection is to a path.
		'circuit' => this connections is to another circuit in the circuit dag in this netlist.

net_name	meaning: The name of the net the component port is connected to. Net names must be unique within a circuit. format: standard name format.

PART CONNECTION properties

A part connection describes the connection of a port of a part to a net in a circuit. It is a connection subtype.

netlist_name	see NETLIST.
defined.circuit_name	meaning: The name of the circuit the part and net are in. format: standard name format.
component.part_name	meaning: A name to uniquely identify in the part library the part being connected. format: standard name format.
component_instance	meaning: A name to uniquely identify the part being connected in this circuit. format: standard name format.
component.port_name	meaning: The name of the port of the part being connected. format: standard name format.
connection_type	see CONNECTION. Value is 'part' for this connection subtype.
net_name	see CONNECTION.

CIRCUIT CONNECTION properties

A circuit connection describes the connection of the port of a component circuit to a net in a circuit. It is a connection subtype.

netlist_name	see NETLIST.
defined.circuit_name	meaning: The name of the circuit the component circuit and net are in. format: standard name format.
component.circuit_name	meaning: The name of the circuit being connected. format: standard name format.
component_instance	meaning: A name to uniquely identify the component circuit being connected. format: standard name format.
component.port_name	meaning: The name of the port of the component circuit being connected. format: standard name format.
connection_type	see CONNECTION. Value is 'circuit' for this connection subtype.
net_name	see CONNECTION.

PORT properties

A port defines the allowed points at which external connections may be made to a circuit.

netlist_name	see NETLIST.
circuit_name	see CIRCUIT.
port_name	meaning: A name that uniquely identifies a port of a circuit. format: standard name format.

net_name
meaning: The name of the net in the circuit that connects to this port of the circuit.
format: standard name format.

standard formats

std date
meaning: A date in standard US format.
format: 8 char string in the form MM/DD/YY. MM and DD do not have leading zeros. YY is always two digits. So Jan. 2, 1988 is 1/2/88, not 01/02/88. But Jan. 2, 2000 is 1/2/00, not 1/2/0.

std time
meaning: Greenwich mean time to the nearest millisecond in 24 hour format.
format: 9 character ASCII string in the form 'hh:mm:ss.msc'. All fields always have leading zeros, so 1 minute after 1 am is: 01:01:00.000. One second after 1 pm is: 13:00:01.000. Ranges:

hh	00 :	23 hours
mm	00 :	59 minutes
ss	00 :	59 seconds
msc	000 :	999 milliseconds

std name
meaning: A human readable identifier for an object.
format: 16 character NLS string.

std user id
meaning: A UNIX user id.
format: 32 bit integer. Range: 0 : 99 999.

std program id
meaning: An identifier intended to uniquely identify a tool and its revision.
format: 16 character ASCII string. Only printing characters (' ' through '~') allowed.

Report on EDAC 1990

Neerav Berry

LSI Logic Corporation
1551 McCarthy Blvd., MS E-192
Milpitas, CA 95035

I would like to thank SIGDA for providing me a grant to attend the first European Design Automation Conference (EDAC) in Glasgow, Scotland, March 12-15, 1990. A large percentage of the attendees where from Europe, though several other countries were represented. The U.S. accounted for 15% of the total attendance. There were 120 technical papers that covered most aspects of electronic design automation. It was heartening to learn that some excellent work in this field is being done by the universities and the industry in Europe.

My primary interest is in High Level Synthesis, and I was quite pleased to see the attention it got at this conference. There were two sessions on Scheduling and Allocation, and one on High Level Synthesis of systems. Although there were a number of good papers in these sessions, the ones I found particularly interesting are:

2A.2 "A Branch-and-bound method for optimal transformation of data flow graphs for observing hardware constraints" by W. Grass

A branch-and-bound method is proposed to produce a data path from a given behavioral description while observing hardware constraints and maximizing the speed. The method starts with a maximal parallel flow graph, and modifies it in a way that hardware restrictions are fulfilled for all possible schedules of the new graph. Among the advantages of the technique are: it explores the entire design space; and, any scheduling algorithm can be used for the resulting graph to optimize hardware costs within the constraints.

3A.1 "A neural net based Self Organizing Scheduling Algorithm" by A. Hemani and A. Postula

A scheduling algorithm is presented based on Kohonen's rule for self organization. The algorithm has an inherent hill climbing mechanism, can be given a comprehensive set of constraints and can be implemented on parallel structures. It will be interesting to see extensions of this approach that allows the optimization of registers, busses and multiplexers.

Report on EDAC-90

Linda Kristoffersen
Department of Electrical Engineering and Computer Science
Division of Physical Electronics
N-7034 Trondheim - NTH, Norway

The 1'st European Design Automation Conference, EDAC – 90, was held at the Hospitality Inn and Convention Centre in Glasgow, Scotland, from 12th to 15th of March, 1990. The conference attracted about 400 attendees from more than 20 countries throughout Europe, but also from USA and from Japan.

I am a Ph.D. (Dr.ing.) student at the Norwegian Institute of Technology at the University of Trondheim, and my main interest is testing of VLSI devices. I gave a poster presentation at EDAC, and I had a lot of interesting questions and comments on the work I presented.

Since I am from a rather small country, with very few people working in the area of testing, it is inspiring to meet people from other parts of the world working in the same area. Thus, I welcomed the opportunity to exchange research ideas with fellow participants, and made professional contacts I hope will be beneficial in the future. At the conference, I went to most of the presentations on testing, and my impression is that both the papers and the presentations were of a high quality, presenting interesting work going on in the area. The sessions covering testing spanned from *Fault Modeling* to *Tools for Testing.*

Of the papers presented, I would like to to mention W. *T. Chen and J. H. Patel, "PROOFS: A super fast fault simulator for sequential circuits"*. A recently introduced concept of differential fault simulation was combined with parallel fault simulation and careful fault selection to achieve very fast fault simulation.

The conference also contained Vendor exhibitions and demonstrations. I took the opportunity to take a look at some of the CAD – tools, which was an interesting experience.

Attending the conference has given me a wider perspective on my own work, and I would like to thank ACM/SIGDA for providing a travel grant covering my airfare to the conference.

Sincerely,

Linda Kristoffersen

Linda Kristoffersen.

Report on EDAC-90
(Glasgow, March 12-15, 1990)

Andrzej Kraśniewski
Warsaw University of Technology
Institute of Telecommunications
Nowowiejska 15/19
00-665 Warszawa, Poland

The European Design Automation Conference, a new event on the Design Automation scene, started rather successfully. Out of 303 submissions from 31 countries, 120 papers were selected for verbal presentations in 3 parallel sessions; besides, there were 30 poster presentations. In addition to the technical presentations, the Program Committee organized a panel discussion "Why Develop CAD in Europe?", a number of fringe meetings for special interests groups, and a tutorial day. Also, several CAD suppliers demonstrated their products and services in vendor suites.

A lot of papers presented at the EDAC'90 are worth to be mentioned, but I was particularly impressed by some presentations on design for testability. There was an excellent paper on design of testable finite state machines authored by V. D. Agrawal and K.-T. Cheng. Two presentations, "Tools and Devices Supporting the Pseudo-Exhaustive Test" by S. Hellebrand and "PEST - A Tool for Implementing Pseudo_Exhaustive Self Test" by E. Wu, addressed practical problems associated with implementation of pseudo-exhaustive testing. The poster "A Synthesis Approach to Reduce Scan Design Overhead" presented by B. Eschermann was another interesting contribution in the area of designing testable FSMs. In general, the poster sessions drew a lot of interest from the audience; I hope that they will become a tradition for EDAC events.

Now, about the other side of the conference. The organizers deserved an applause for the social program. The get-together party, the reception at the City Chamber, the banquet, and even the coffee breaks - all these meetings were well organized and enjoyable.

UNIVERSITY OF MINNESOTA
TWIN CITIES

Department of Computer Science
4-192 EE/CSci Building
200 Union Street S.E.
Minneapolis, MN 55455

December 3, 1989

Dr. James P. Cohoon
SIGDA Travel Grant coordinator
Department of Computer Science
University of Virginia
Thornton Hall
Charlottesville, Virginia 22901

Dear Dr. Cohoon:

Thank you for the award of a travel grant from SIGDA to present a paper at ICCAD 89. I found the conference to be very interesting and very well organized. Participation was quite international, with presentations from thirteen countries including the US. The technical level of the presentations as well as the many questions asked by the floor were excellent.

Though I have interest in several CAD areas, my area of active research is timing analysis and prediction of timing requirements of VLSI's. I was very pleased to see three sessions devoted to timing related topics. Overall, there were 11 papers related to timing issues in IC designs. The paper I had the opportunity to present was on the critical path issue in VLSI's. I was pleased with the interest of the floor in the subject of my paper and timing in general.

I had the opportunity to attend sessions related to other CAD areas, such as physical design, synthesis, and modeling. As a VHDL user, I enjoyed very much Session 4c on the VHDL language, especially the two presentations on the use of VHDL for switch level modeling and analog modeling. I also enjoyed the presentation by Dr. Murdocca from Rutgers University on Digital Computing and the need for CAD tools in that area of engineering.

I am most pleased with the opportunity to meet with so many of the active researchers in the CAD community, both from Industry and Academia.

Sincerely,

Habib Youssef

ICCAD '89

Johnson Chan Limqueco
Department of Computer Science
University of Illinois at Urbana-Champaign
1304 W. Springfield, Urbana, IL 61801

This year's ICCAD was slightly different from the previous year's. First of all, the tutorials were held on the last day of the conference, instead of the first day. Secondly, presentors of papers in the technical sessions were all required to use slides, instead of having the option of using transparencies. As a result, the technical sessions were better organized, ran more smoothly, and looked more professional.

My research work being in Logic Synthesis, the sessions I attended were mostly in this or related areas. One thing that stood out to me from attending all those sessions was the intimate relationship between logic synthesis and testing. Researchers were finding out more and more on how this can be exploited to produce optimal yet testable designs.

Hachtel et al., in their paper, "On Properties of Algebraic Transformations and the Multifault Testability of Multilevel Logic," showed that algebraic factorization can be applied to a minimized two-level circuit which is completely single-fault (and therefore, also multifault) testable to produce an area-optimized, completely multifault testable multilevel circuit. Furthermore, all multifault tests of the synthesized circuit can be derived from the single-fault tests of the original two-level circuit. In a similar vein, Geer and Brayton, in their paper, "Consistency and Observability Invariance in Multi-level Logic Synthesis," showed that most of the algebraic and some Boolean operations commonly used in logic synthesis preserve the testability of all but a single node in a network.

Cheng and Agrawal presented two interesting papers on finite state machine synthesis addressing the problem of testability. In their first paper, "State Assignment for Initializable Synthesis," they addressed the problem of conventional synthesis methods producing designs which may be uninitializable during simulation or test generation. Their new state assignment algorithm finds a synchronizing/initialization sequence for the finite state machine to be

synthesized, derives a set of state groups from this sequence, and transforms these into a set of constraints to be considered during state encoding. In their second paper, "Design of Sequential Machines for Efficient Test Generation," they introduced *SACRED*, a state assignment algorithm that attempts to minimize the number of feedbacks (or cycles) in the synthesized circuit. Their approach is to assign state variables in multiple passes, assigning those variables that depend only on primary inputs and the smallest number of previously assigned state variables in each pass. The algorithm resorts to a conventional state encoding procedure when each remaining unassigned state variable depends on all unassigned variable. Designs produced by *SACRED* have a pipeline–like structure which is easily analyzed by a sequential circuit test generator. As a result, test generation times for these were much faster, fault coverage was improved, and surprisingly, circuit area was also reduced.

In a different direction, Jacoby et al., in their paper, "New ATPG Techniques for Logic Optimization," explored the use of automatic test pattern generation techniques, such as implication, contrapositive relations, unique sensitization, etc., to identify and remove redundancies in a network. They gave improved algorithms and reported comparable to superior results with one to two order of magnitude speedup in execution time.

Two papers describing multilevel logic optimization procedures using the powerful yet unpublicized Transduction Method based on permissible functions, "SYLON–DREAM: A Multi–Level Network Synthesizer," by Chen and Muroga, and "Multi–Level Logic Optimization Using Binary Decision Diagrams," by Matsunaga and Fujita, were also presented. They showed favorable results compared to existing multilevel network optimizers.

The conference has been an interesting and valuable learning experience for me. My thanks and congratulations to all those who made it the success that it was, from the executive committee to the presentors, and also to SIGDA for giving me the opportunity to be a part of it.

THE INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN 1989

Kaushik Roy
Computer Systems Group
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign.

The International Conference on Computer Aided Design 1989 (ICCAD-89) was held at the Convention Center in Santa Clara, California, from November 5 to November 9. It was surprising to note the smoothness with which such a large conference was organized. Three sessions ran in parallel in three large rooms with very good audio-visual facility.

There were 36 sessions in which 126 papers were presented. The conference had a strong emphasis on both high-level and logic synthesis and synthesis for testability. The other areas of interest were timing simulation, automatic test pattern generation and logic and fault simulation.

I presented my paper in session 9C (Synthesis for testability). It addressed the issue of synthesizing circuits which are robust delay fault testable. The other papers in the session looked at manipulating Boolean functions to maintain multi-fault testability criteria. Session 5C covered Finite state machine synthesis. The first paper in that session considered Boolean minimization and algebraic factorization procedures for fully testable sequential machines. The second paper had interesting ideas about state assignment for initializable synthesis. Session 11B (High performance simulation) was also of considerable interest to me. Paper 1 described hierarchical compiled event-driven logic simulation. Paper 2 considered the theoretical issues of parallel logic simulation, while Paper 3 showed the implementaion of a parallel logic and fault simulator.

The exhibition booths from the industries were in Double Tree Hotel rooms. The major companies involved in Design Automation were there to provide information about their latest products and developments.

It was a great opportunity and learning experience for me to be at the conference. The panel sessions were very helpful in getting insights into the philosophies of different aspects of Computer-Aided Design. I certainly would like to attend the ICCAD-89.

An Aspect of ICCAD-89: An Attendee's Report

Ankan K. Pramanick

Department of Electrical and Computer Eng.
The University of Iowa
Iowa City, IOWA - 52242

I had an opportunity to attend this year's International Conference on Computer Aided Design, held at Santa Clara, California, from the 6th. to the 9th. of November, where I was scheduled to present a paper. Being primarily interested in testing and Design For Testability (DFT), I was interested to find a relatively high number of technical sessions dealing with such issues. There were five sessions entirely devoted to testing, and several others that addressed in part many different issues in testing.

From these sessions, and also from the tutorial on "New Trends in Testing and Verification" that I attended, an important aspect that seems to be evolving rapidly is Synthesis For Testability (SFT). With the advances in automated circuit design and IC technology, the problems of testing have become much more difficult because of the increase in functionality per chip, larger numbers of hierarchical design levels, and lowered observability at the accessible chip pins. Traditionally, synthesis has been aimed at optimizing chip area/delays. This year we saw several presentations that aimed at using aspects of synthesis specifically meant to increase the ease of testing of the designed chip, namely, the further evolution of SFT. These presentations demonstrated how fresh synthesis approaches can produce both combinational and sequential circuits with very high levels of testability. These include design methodologies for the multifault testability of multilevel logic, design for testability of sequential machines at the logic synthesis level, and synthesis of delay fault testable combinational circuits.

This trend of DFT starting at a relatively early level in the synthesis process, as opposed to commonly used DFT methodologies that rely on post-synthesis logic modifications, appears to be an important and rapidly evolving new aspect in logic design and testing. I think we can look forward to many interesting developments in SFT in the future.

Report on the 1989 IEEE International Conference on Computer-Aided Design - ICCAD-89

L. N. Kannan
Department of Electrical & Computer Engineering
University of Cincinnati
Cincinnati, OH 45221

The Seventh IEEE International Conference on Computer-Aided Design (ICCAD-89) was held between November 5-9 at the Santa Clara Convention Center in California. There were approximately 125 technical papers presented over the three day period and there were several impressive exhibits by various industries in the area.

In addition to the technical papers, there were several informative tutorials and panel discussions. The papers were presented in three parallel technical sessions. Therefore, although there were several technical papers that were of my interest, I could attend only a few of them. I have described below an aspect of the conference I found interesting.

A trend that was reflected in the papers presented in ICCAD-89 was in the area of design for testability. Instead of trying to develop efficient test procedures for complete fault coverage (ATPG), we try to synthesize the circuit so that it is 100% testable even with existing testing techniques. This sometimes resulted in some unexpected bonuses!

An interesting paper was one by Cheng and Agrawal which gave a state assignment procedure that tries to reduce the number of cycles in the implemented finite state machine. This resulted in remarkable improvement in fault coverage and test generation. As an interesting side effect, it also resulted in lower logic complexity in some cases! Roy et. al. presented a paper on the synthesis of combinational logic that is delay fault testable. Hachtel et. al. presented a number of results regarding the relationship between algebraic transformations used for area optimizations and the testability of combinational logic circuits.

The emphasis was on integration of design procedures with testing schemes. Overall my attendance at ICCAD-89 was a rewarding experience. I would like to thank the organizers of the conference for an outstanding job and SIGDA for providing the travel grant which made my attendance at ICCAD-89 possible.

DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

The Technological Institute
Northwestern University
Evanston, IL 60208

TEL: (312) 491-7378
FAX: (312) 491-4133
majid@eecs.nwu.edu

November 14, 1989

SIGDA, Travel Grants

Dear Travel Grant Committee:

I would like to thank you for providing me with a travel grant to attend ICCAD-89. I presented a paper entitled "*A Powerful Global Router: Based on Steiner Min-Max Trees*" (joint work with C. Chiang and C.K. Wong). We proposed a novel formulation of the Steiner tree problem, called Steiner min-max trees. This is the first class of efficiently solvable Steiner tree problem, in general graphs. An effective global router, based on Steiner min-max trees, were proposed.

I enjoyed the conference. It was a unique opportunity to discuss both theoretical and practical aspects of VLSI layout. Researchers and designers are trying to integrate their expertise in mathematical analysis and implementation to attack complex problems.

With best regards,

M. Sarrafzadeh
Assistant Professor

Computer Science Department
University of Minnesota
EE/Csci 4-196
Minneapolis, Minnesota 55455

November 28, 1989

Dr. James P. Cohoon
SIGDA Travel Grant coordinator
Department of Computer Science
University of Virginia
Thornton Hall
Charlottesville, Virginia 22901

Dear Dr. Cohoon:

Thank you for the award of a travel grant from IEEE SIGDA to attend ICCAD 89. I found the conference to be very interesting and it afforded me the opportunity to determine the direction of research in those areas with which I am familiar, as well as acquaint myself with an overview of current research in some related areas outside the scope of my own research.

I was a pleasure to attend several of the presentations related to parallel simulation since I am currently working in this area. The presentations were usually clear and enjoyable. There seemed to be more questions from the floor this year than there were last year. These questions from the attendees helped highlight trouble areas and also clarified more difficult aspects of the work.

I also attended several interesting talks on VHDL. Although not well-versed in this area, I nonetheless found that the excellent presentations provided me with a better understanding of the problems being studied there. But the most enjoyable and instructive part of the conference was meeting people doing research similar to my own. Such exchanges of information and ideas help everyone. Thanks you once again for the grant.

Sincerely,

Patrick L. Jarvis

New Frontiers in Logic Synthesis: A Report on IWLS 89

Sharad Malik

University of California, Berkeley

This year the International Workshop on Logic Synthesis was held at the Research Triangle Park in North Carolina from May 23-26. One of the primary results of this confluence of research is to clearly identify the research frontiers.

Based on the presentations, the panel sessions and the discussions that ensued three areas of current and future significance emerged. These are: **performance driven synthesis, synthesis for testability** and **sequential synthesis**. I will be briefly reviewing each of these.

When asked to identify as to what they would like logic synthesis to deliver to them, the wish list of nearly all the industrial participants was identical. The desirable synthesis system was one which would first meet the performance constraints, then have a high fault coverage, and finally, it would be nice if it resulted in significant area saving. Given the fact that most existing logic synthesis efforts are directed towards area optimization, this was a clear indicator that a change in direction was due. There has been a wealth of papers presented in area optimization for combinational logic over the past few years. However, the literature in performance driven optimization is scarce. [10, 2] represent the state of the art in papers that have been presented in performance optimization. But even these authors admit that significant work still needs to be done in this area. In this direction, Richard Rudell from Synopsys Inc. presented a paper ([9]) on technology mapping for performance optimization at this workshop. It showed how the conventional tree mapping algorithms for technology mapping (e.g. [6]) could be extended for performance optimization. By considering a multi-pass method, a minimum area solution that meets the delay constraint can be obtained by this method (if one exists). This work represents a significant step in the direction of technology mapping for delay. However, the open question that still remains is: What are the technology independent

resynthesis operations that may be done for performance optimization?

With increased circuit complexity, reliability is becoming of increasing concern. It is critical that faulty circuits be detected before they are shipped. This puts the burden on testing engineers. An alternative and perhaps preferred scenario would be if a synthesis process could guarantee the testability of the circuits and provide the test vectors. For the single stuck-at model, synthesis techniques have been presented for both combinational (e.g. [1]) and sequential circuits (e.g. [4]) that promise this. The next step is to consider the multiple stuck-at fault model. At the workshop, a paper by Hachtel *et al.* ([5]) presented a technique for the synthesis of multi-fault testable combinational circuits for which a single fault test set detects all multiple-faults. Questions that still need to be answered here are: How relevant is this fault model, and how critical is the area penalty imposed by this synthesis technique?

Combinational logic synthesis has attained significant maturity over the past five years, at least in the area of area optimization. However, there have been little or no efforts in the area of sequential circuit synthesis. This workshop marked the emergence of the first significant efforts in this area. The author, in collaboration with other researchers at Berkeley, presented a paper ([8]) that considers optimizing sequential circuits by first migrating all the registers in a sub-circuit to the boundary of the sub-circuit, then applying combinational optimization techniques to this sub-circuit, and finally replacing the registers in an optimal way. The most interesting aspect of this work is that it pushes well known combinational optimization techniques to their limits in the sequential setting. Giovanni DeMicheli from Stanford presented a paper ([3]) that had a similar motivation, i.e. considering combinational optimization techniques beyond latch boundaries. In addition, an interesting alternative solution to performance optimization by retiming was presented in this paper. In an alternative approach, Bill Lin from Berkeley considered optimizations that could be obtained by first extracting the behavior of the circuit in terms of its state transition graph and then considering optimizations at the state transition graph level ([7]).

In conclusion, to reiterate the main points, the workshop recognized that area optimization for combinational logic was reasonably well handled by current synthesis tools, and made a case for pushing in the direction of performance optimization, synthesis for testability, and sequential optimization.

References

[1] D. Bostick, G. D. Hachtel, R. Jacoby, M. R. Lightner, P. Moceyunas, C. R. Morrison, and D. Ravenscroft. The Boulder Optimal Logic Design system. In *Proceedings of the International Conference on Computer-Aided Design*, 1987.

[2] G. DeMicheli. Performance-oriented synthesis of large scale domino CMOS circuits. *IEEE Transactions on Computer-Aided Design*, CAD-6(5):751–765, September 1987.

[3] G. DeMicheli. Synchronous logic synthesis. In *Proceedings of the International Workshop in Logic Synthesis*, 1989.

[4] S. Devadas and K. Keutzer. Boolean minimization and algebraic factorization procedures for fully testable sequential machines. In *Proceedings of the International Conference on Computer-Aided Design*, 1989.

[5] G. Hachtel, R. Jacoby, K. Keutzer, and C. Morrison. On the relationship between area optimization and multi-fault testability of multi-level logic. In *Proceedings of the International Workshop in Logic Synthesis*, 1989.

[6] K. Keutzer. Dagon: Technology binding and local optimization. In *Proceedings of the Design Automation Conference*, 1987.

[7] B. Lin and A. R. Newton. Restructuring state machines and state assignment: relationship to minimizing logic across latch boundaries. In *Proceedings of the International Workshop in Logic Synthesis*, 1989.

[8] S. Malik, E. M. Sentovich, R. K. Brayton, and A. Sangiovanni-Vincentelli. Retiming and resynthesis: Optimizing sequential networks with combinational techniques. In *Proceedings of the International Workshop in Logic Synthesis*, 1989.

[9] R. Rudell. Technology mapping for delay. In *Proceedings of the International Workshop in Logic Synthesis*, 1989.

[10] K. J. Singh, A. R. Wang, R. K. Brayton, and A. Sangiovanni-Vincentelli. Timing optimization of combinational logic. In *Proceedings of the International Conference on Computer-Aided Design*, 1988.

Conference Report of the International Workshop on Logic Synthesis 1989

Michael Neher

Computer Science Research Center (FZI)
at the University of Karlsruhe
W. Germany

Introduction: The International Workshop on Logic Synthesis was held on May 24 - 26, 1989 at the Microelectronics Center of North Carolina in Research Triangle Park, N.C. The over 100 attendees came mainly from US with very low European participation. The 11 sessions covered multilevel combinational logic synthesis, sequential logic synthesis, control FSM synthesis, behavioral synthesis systems and complexity issues in synthesis. One session was dedicated to benchmark and poster presentations, chaired by R. Lisanke, who is responsible for the MCNC benchmark distribution.

I attended the conference with support from SIGDA and want to point out the highlights of the conference:

Control FSM synthesis: There were four papers presented on state assignment. G. Saucier (INPG/CSI, Grenoble) concentrated on the generation of coding constraints for the state assignment. T. Villa (ucb) uses symbolic logic minimization for the generation of coding constraints in his NOVA system. He developed an exact, but time consuming algorithm for embedding the coding constraints. All coding constraints are taken into account, so the length of the state code field is eventually not minimal. From his exact algorithm he derived different very good and fast heuristic algorithms for embedding the most promising coding constraints with minimal code length. G. Rietsche and M. Neher also presented a new heuristic embedding algorithm in their control synthesis system CASTOR. The state assignment results and cpu times are comparable to NOVA. CASTOR generates furthermore appropriate pre- and postprocessing structures with multiplexers and decoders for the kernel FSM. M. Benshop (Philips Research Labs) uses a Min Cut algorithm for embedding the coding constraints.

Sequential logic synthesis: Three talks were presented, introducing a global minimization approach for mixed combinational and sequential logic circuits. The central idea is to shift the registers to the border of the circuit and then to minimize the combinational logic. Afterwards a retiming technique is used to reinsert the registers, so that the gate delay between registers is balanced. Results and cpu times were not presented.

Multilevel logic synthesis: The most interesting talk in this area was presented by T-T. Hwang and R. Owens (Penn State Univ.). Unlike the usual multilevel logic synthesis systems they use the communication complexity of circuits (i.e. the wire count) instead of the literal count as the primary

design goal. The results for at least some kinds of circuits (with arithmetic units) are very promising and far better then MIS (ucb) results. The system is able to generate a carry ripple adder structure from an adder function table. There has to be done some work to minimize the amount of needed cpu time and memory.

Benchmark presentations: There exists now quite a large set of benchmarks for two and multilevel logic and for finite state machines, which consists not only of "university circuit descriptions", but also of some industrial examples and an example cell library with transistor count and area/time measures for each cell. So the benchmark set can be quite useful in comparing multilevel logic synthesis systems or state assignment algorithms. More or less complete results for the benchmark set were presented by:

G. Saucier (INPG/CSI, Grenoble), M.Neher/G.Rietsche (Univ. of Karlsruhe), T. Villa (ucb) for single PLA FSM´s; N. Benshop (Philips Research Labs) for random logic FSM´s; Brayton et.al. (ucb), T-T. Hwang/R. Owens (Penn State Univ.) and others for multilevel logic synthesis.

The Second Oxford Workshop on CAD Accelerators : A Report

C.P. Ravikumar

October 5, 1989

BACKGROUND

The second International Workshop on CAD Accelerators was held at the University of Oxford during 20th and 21st September, 1989. It was attended by about 30 research professionals from Universities as well as Computing Industry. The participants were mainly from U.S.A, Canada, and Europe (U.K, West Germany, The Netherlands, Italy, France and USSR). The workshop was jointly organized by the departments of External Studies and Engineering Science at the Oxford University. It is one of the series of technical workshops held every year as part of the *Continuing Professional Development Programme* at Oxford. The first workshop on CAD Accelerators was held in 1987. A technical digest of the papers presented at the 1987 workshop has been published [2]. The aim of the workshop, as stated by Tony Ambler and Will Moore (who were the chief organizers) is to bring together workers who are involved in the design and use of hardware accelerators, novel CAD algorithms, and related topics.

CAD Accelerators

Judging from the material presented at the workshop, it is clear that different definitions exist for the term "CAD Accelerator". The conventional definition of a CAD Accelerator (also known as a *Hardware Accelerator* or a *Point Accelerator*) is that of a special-purpose machine built to execute a single CAD algorithm. The Machester Routing Machine described by Edwards [5] and the Munich Simulation Engine described by Jacoby [6] fit into this category. The former is a routing engine developed at the University of Manchester. It is based on the Lee-Moore algorithm. The Munich Simulation Engine is being developed at Siemens, Germany to handle circuit simulation at several levels – behavioral level, logic level, and electrical level.

Agrawal discussed a *programmable* accelerator called MARS which can handle several useful applications such as logic simulation and fault simulation [1].

A radically different view of the term *hardware accelerator* is held by several other attendees of the workshop. With the proliferation of general-purpose parallel machines, the time has come to re-evaluate the relative merits and demerits of general-purpose machines and special-purpose machines for CAD algorithms. Since general-purpose machines are flexible, they can be used to run more than one CAD application. They provide a programming environment which is complete with command interpreters, languages, com-

pilers, and debuggers. Therefore software development on general-purpose machines is much more easy an rapid. Hence, this situation encourages experimentation. If an entirely new algorithm is proposed for some CAD application, the general-purpose machine can be readily reprogrammed. Already a large number of researchers are developing CAD algorithms for parallel computers [8, 11] and their results have been promising. It is certain that in the coming years general-purpose machines will win over point accelerators both in terms of cost and in terms of performance. This view was expressed by several people at the conference. Ravikumar and Sastry presented parallel algorithms for layout [9, 10]. Briner presented a parallel algorithm for mixed-level simulation [3]. Rob Smith, who presented a novel scheme to build CAD accelerators, clearly expressed his concern about the future of point accelerators. Since VLSI technology allows the designer to integrate more and more functionality into chips, general-purpose systems are likely to evolve rapidly. The "Experimental Systems" division at MCC (Austin,Texas) has developed a general-purpose hardware module which can be used as a building block to construct scalable parallel architectures. Through efficient packaging of these modules, Smith and his associates have successfully engineered several parallel machines. Since the performance improvement obtainable by improving the *hardware* of the design is likely to reach its limit fairly quickly, Smith felt that further research must focus on developing efficient parallel algorithms.

There is also a growing interest in using a network of computers (or transputers) to accelerate CAD tasks. An entire session was devoted to papers which described the use of transputers for logic simulation, design rule check, and mixed mode simulation. Since a local area network of workstations has gained a great deal of popularity, there is a natural tendency to make use of the "idle CPU-cycles" to run a *distributed algorithm.* As a result, an emerging field of research is that of distributed CAD algorithms. Steven Smith described an attempt at building a distributed logic simulation algorithm to run on a network environment.

MAIN ISSUES

The workshop was organized in seven sessions. The main issues discussed in these sessions may be classified as follows.

- History of CAD Accelerators
- Advances in Simulation Accelerators
- Advances in Layout Accelerators
- Novel computing platforms
- Future Directions

A HISTORICAL PERSPECTIVE

Tim Saxe [12] identified some interesting history associated with hardware accelerators. Accelerators were first designed by CAD professionals who realized that logic simulation of large circuits cannot be completed in real time if conventional VAX-like machines were used. When simulation accelerators became a commercial success, other CAD applications were also considered for acceleration. Special-purpose machines were built for design rule verification, circuit simulation, switch level simulation, layout, and fault simulation. Certain problems were soon observed with the building of *point accelerators.* Since a point accelerator executes a single algorithm, the question is which algorithm should be selected

for implementation. It may not pay off to accelerate "conventional" algorithms which have been used on VAX-like machines. As a result, entirely new algorithms may have to be invented for accelerators.

SIMULATION ACCELERATORS

There are two sources of concurrency in the circuit simulation problem : data parallelism and control parallelism. Data parallelism comes from the fact that a large circuit can be partitioned into smaller subcircuits which can be handled concurrently. Control parallelism is due to algorithmic concurrency. Several papers presented at the Oxford workshop attempted to exploit both the sources of parallelism.

David Lewis [7] described a hardware accelerator called Awsim-3 for speeding up *compiled simulation.* Simulators are of two basic types, event driven simulators and compiled simulators. The latter technique has been used for gate level and switch level simulation. It is less suitable for circuit level simulation, since model evaluation phase is compute-intensive. Awsim-3 tries to overcome this problem by using dedicated hardware for device model evaluation.

The aim of the Awsim project is to build a single accelerator which can handle a variety of simulations. It uses a general-purpose processor (GP) and a set of special-purpose processors (SP). The GP gathers terminal voltages and other parameters into a register block and initiates a transfer to an SP. The SP evaluates the device model and returns the results to GP over a high speed bus. Each SP is built around a data structure known as the 2^N tree. It is a table data structure which allows more than one analytical model for the same device.

The architecture of GP is tuned to execute the *compiled simulation* algorithm efficiently. The most frequent operation in a circuit simulation consists of the following three steps : (1) Gather parametric values from an array into a set of consecutive registers, (2) evaluate a device model, and (3) scatter the results back to an array. The compiled simulator unrolls the loop consisting of the above operations. The resulting straight-line code can be run efficiently on a deep pipeline. Furthermore, the unrolled code exhibits a considerable amount of parallelism which can be detected statically. The GP consists of five independent "slices", each of which can handle an independent instruction stream. Up to five independent operations can take place in one cycle. Crossbar network is used to exchange data among the slices. Each slice has a 12 stage instruction pipeline. Lewis estimates that the GP can execute up to 90 "RISC like" instructions per clock cycle. The GP uses a VLIW (very large instruction word) format to keep all the five instruction streams active. A program called *packer* is used to generate the object code for GP. The packer looks at the unrolled code and packs those instructions which can be executed concurrently.

There are several performance issues related to Awsim-3 or any other hardware accelerator. Simulation of Awsim-3 has provided useful feedback on many of these questions. What is the amount of instruction-level parallelism which can be detected? To answer this question, Lewis built a packer program which assumes an infinitely parallel machine. Three different simulation algorithms were tried on this virtual machine. These were *direct simulation*, *point relaxation*, and *block relaxation* algorithms. It was observed that the average parallelism is highest with point relaxation algorithms and lowest with block relaxation algorithms. The second question is, what should be the degree of parallelism in an actual machine? Simulation shows that for block relaxation, processor utilization decreases rapidly after the machine parallelism increases beyond 5 or so. On the other hand, the direct method and the point relaxation algorithm can efficiently utilize a machine parallelism of about 100.

PROGRAMMABLE ACCELERATOR

Prathima Agrawal described the architecture of MARS, a microprogrammable accelerator for rapid simulations [1]. It is a multiprocessor-based system intended to be a programmable accelerator which can support a variety of CAD applications. However, MARS is ideally suitable for *event driven* simulation of circuits, delivering a peak performance of 1 million gate evaluations per second.

MARS consists of a number of *clusters* connected by a Hypercube communications network. Each cluster consists of 14 processing elements (PEs). The PE architecture is microprogrammable. Each PE has a local memory bank. The 14 PEs, a housekeeping

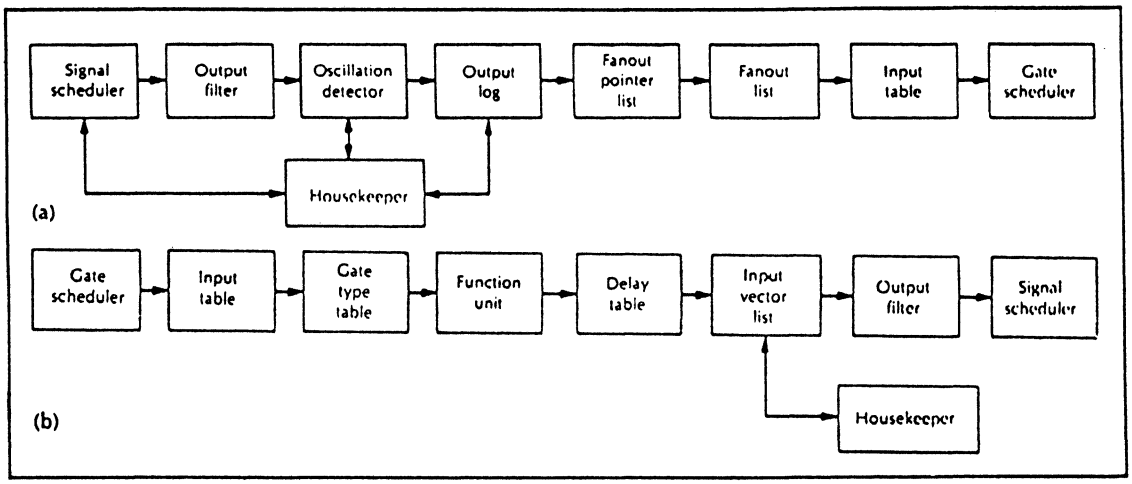


Figure 1: The Simulation Pipeline in MARS – The figure is borrowed from Ref. [1]

processor, and the communications manager are connected by means of a 16×16 crossbar network. In a simulation environment, each PE is programmed as one stage of a simulation pipeline (see Figure 1(a)).

The first stage in the pipeline, called signal scheduler, schedules changes in signal values. The second stage keeps track of pending signals and filters out unnecessary events. The third stage detects zero-delay oscillations. For signals which are being observed, an *output logger* records events in its data memory. The next three stages are responsible for handling important data structures associated with logic simulation, namely, fanout pointer list, fanout list, and input table. The *fanout pointer list* receives a 'gate/value' message from the output logger, looks up a pointer into the gate's fanout list, and passes the pointer and the value along the fanout list. The *fanout list* takes the 'pointer/value' message and uses the pointer to look up the fanout list for the gate. The *input table* maintains a list of input values for each gate. Upon receiving a 'gate/value' message from the fanout list, it updates the input table and forwards a message containing the gate identifier to the *gate scheduler*. The gate scheduler schedules events corresponding to those gates whose inputs have changed.

A pipeline is also used for gate evaluation, as shown in Figure 1(b). The gate scheduler

pops an event off its stack and forwards it to the *input table*. The input table looks up the input values for the gate and sends them as a 'gate message' to the *gate type table*, which appends the type information to the gate message. The *function unit* evaluates the gate. The *delay table* looks up the appropriate gate delay. The *input vector list* receives the delay information and converts it into time. The output filter detects new events and sends them to the signal scheduler.

MARS has been used with a circuit compiler which translates a circuit description into appropriate data structures needed by different PEs of the accelerator. The compilation is quite fast, requiring about 100 ns on a VAX 780 for a 52,000-transistor circuit. MARS has provided a peak performance close to 1 million gate evaluations per second. A fully configured MARS system will have 256 MARS clusters connected by an 8-dimensional Hypercube network. Such a system is predicted to have a performance of 100 million gates per second.

LAYOUT ACCELERATORS

A parallel approach to three layer channel routing was presented by Ravikumar [10]. On a theoretical plane, the paper considers parallel execution of channel routing on an abstract machine known as EREW PRAM. The abbreviation stands for a *p*arallel *r*andom *a*ccess *m*achine which supports *e*xclusive *r*ead and *c*xclusive *w*rite operations. There are two important results in this work. Given a channel with N two point nets, it is shown that the nets can be routed in the optimal number of tracks in three layers in $O(\log N)$ parallel time. The scheme uses $2N$ processors, all of which have random access to all the locations in shared memory. The paper shows that the lower bound on routing time is also $\Omega(\log N)$.

A second source of parallelism in the channel routing problem comes from routing several channels concurrently. A channel order graph can be constructed to reflect the routing dependences among channels. Given a floor plan with M modules, the channel order graph has $M-1$ nodes, one corresponding to each channel. If we restrict ourselves to *sliced* floorplans, the channel order graphs are *trees*. All the channels at any level of the tree can be routed concurrently. Therefore, the depth of the channel order tree is a measure of the parallelism in the routing problem. It is known that for large M, the average depth of a tree with M nodes grows asymptotically as $\sqrt{M}$. It may therefore be expected that a chip with M modules and N nets per channel can be routed in $O(\sqrt{M}\log N)$ time.

Ravikumar and Sastry presented a layout accelerator based on a novel parallel architecture known as the *reduced array* or *orthogonal array* [9]. The architecture consists of a linear array of processors which share a two dimensional array of memory. Processor i is connected by a high speed bus to all the memory modules in row i and all the memory modules in column i. There are two memory access modes, called *row access* and *column access*. During a row access, processor i can read or write to a memory module in row i. A column access is similarly defined. There cannot be an access conflict between two processors due to this "orthogonal access" property. The reduced array was originally conceived for image processing applications. Ravikumar and Sastry have described how to perform Mincut placement and Cut-and-Paste global routing on the reduced array.

The Min-cut placement algorithm and the Cut-and-Paste routing algorithm both have a tree structure. For example, consider Min-cut placement. Given a set of logic modules, they are first partitioned into two equal-sized subsets such that the number of connections across the partition is minimized. The partition process is recursively applied to each of the subsets until a subset has no more than a single module. Each partition is associated with an imaginary cutline on the chip. Following a partition, the modules in the two subsets are placed on either side of the cutline. The placement procedure is conveniently described by means of a process tree. This binary tree maps onto a tree architecture in a straight forward manner. However, a tree machine implementation will suffer from poor processor utilization. By making two observations, the mincut process tree can be mapped nicely to a

linear array of processors. First, notice that processes at any one level of the tree are active at any time step. Since there are M nodes at the leaf, M being the number of modules, at most M processors are required. The granularity of the processes increases as we proceed from leaf level to root level. This suggests that we can do the placement in $\log M$ steps, where each step corresponds to the following computations : (1) use a circuit bisection program to partition the circuit into two equal parts, (2) pass the left subcircuit to the left subtree and vice versa. There being 2^i nodes at level i, each of which has no more than $M/2^i$ modules, an M-processor array can be used to implement the min-cut procedure. The Cut-and-Paste algorithm for global routing has a similar *process tree* formulation and can therefore be implemented on the reduced array using a similar technique.

NOVEL COMPUTING PLATFORMS

The transputer appears to be attracting many CAD professionals as a cost effective and fast solution to many CAD problems. Alan Gray described an effort in INMOS to build a VLSI workstation based on the transputer [4]. The cost of a fully configured machine, which includes separate alphanumeric and graphics terminals, up to 8MB of application-dedicated memory, a transputer-based communications network, and dedicated peripheral chips, is about $12,000.

Jacoby discussed *ESimAc*, a simulation accelerator which uses transputer as a basic building block [6]. *ESimAc* is being jointly built by Siemens AG and Brunel University, UK. The accelerator implements an event driven mixed-mode simulation algorithm. A VLSI circuit is decomposed into several subcircuits, each of which is simulated on a different processor known as EFMP. EFMP stands for an *event flow macro processor* and handles all the activity associated with the firing of an event. The authors subdivide an even flow into nine different processes, which include the initialization of event data structures, element evaluation, fanout determination, and updating the event list. Within the EFMP, functional pipelining is used to obtain increased performance. Each stage of this functional pipeline is implemented on a T800 transputer. The prototype consists of 16 EFMP nodes connected by a two dimensional end-around mesh network.

FUTURE DIRECTIONS

In future, CAD accelerators will play a major role in the field of design automation. According to Tim Saxe, today's fabrication facilities can handle much larger designs than can be designed using CAD tools. This situation can only be improved by employing CAD accelerators.

Saxe [12] felt that future research in CAD must focus on designing at a higher level of abstraction. Even to this day, the typical designer in CAD industry tends to think in terms of TTL gates and MSI logic devices. This situation needs to be changed. Saxe identified three interesting ways in which this could materialize. First, accelerators should focus on high level descriptions such as those allowed by VHDL. Accelerated simulation must be carried out at the instruction set level as well. Finally, recognizing the importance of synthesis in compiling high level designs to low level circuit descriptions, acceleration efforts must concentrate on the area of VLSI synthesis.

References

[1] P. Agrawal et al. Modeling and simulation of functional memory blocks in the MARS accelerator. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper C1, 1989.

[2] W. Moore A.P. Ambler, P. Agrawal, editor. *Hardware Accelerators for Electrical CAD.* Adam Hilger, 1988.

[3] J.V. Briner et al. Parallel mixed level simulation using virtual time. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper H2, 1989.

[4] S. Christian and A. Gray. A transputer-based VLSI workstation. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper E1, 1989.

[5] D. Edwards. Manure2 : A second generation accelerator for PCB routing. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper G1, 1989.

[6] K. Jacoby et al. ESIMAC : a hardware accelerator for mixed mode simulation. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper E4, 1989.

[7] D. Lewis. Performance issues in a compiled hardware accelerator. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper B2, 1989.

[8] C.P. Ravikumar. Parallel Routers. In G. Zobrist, editor, *Progress in Computer Aided VLSI Design Vol, VI.* Ablex Publishing Corporation, 1990.

[9] C.P. Ravikumar and S. Sastry. Lara : A Layout Accelerator based on Reduced Array architecture. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper G2, 1989.

[10] C.P. Ravikumar and S. Sastry. A Parallel Approach to Three-layer Channel Routing. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper H1, 1989.

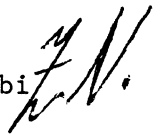
[11] C.P. Ravikumar and S. Sastry. Placement Acclerators. In G. Zobrist, editor, *Progress in Computer Aided VLSI Design, Volume V.* Ablex Publishing Corporation, 1990.

[12] T. Saxe and P. Offredi. Changing the Hardware Acceleration Game. In *Proceedings of the 2nd Oxford Workshop on CAD Accelerators*, page Paper A1, 1989.

Report on VHDL Methods Workshop

by

Dr. Zainalabedin Navabi



At the VHDL workshop is Charlottesville, Virginia, presentations were made by university and industrial researchers.

Papers and presentations made by researchers at the Universities indicated that much work on development of VHDL based tools is being done at the universities.

Several papers were presented on synthesis from VHDL. Most of these talks centered around synthesis from a subset of VHDL. The subsets from which synthesis are being done differ, but they all share certain constructs of the language that can be categorized as "dataflow". Researchers at the University of Virginia presented talks on special applications of VHDL and VHDL based tools that have been developed at Virginia.

Several papers were presented on the subject of education and methods of teaching VHDL. The presentations from universities indicated that VHDL is being taught in association with hardware. At Virginia Polytechnic Institute, VHDL is being taught and used in a modeling course; while at Northeastern University, VHDL is being taught and used in a CPU design course. At IBM, hardware aspects of VHDL are taught to digital system designers in a short course.

A paper from Vantage Corporation was particularly interesting in that it presented advanced applications of bus resolution functions. In this talk methods for better utiliation of resolution functions were suggested.

Overall the workshop was both instructive and informative.

A Workshop on Applied Formal Methods for VLSI

A Trip Report

Zheng Zhu,
Department of Computer Science
Indiana University
Bloomington, Indiana USA

The International Workshop on Applied Formal Methods for Correct VLSI Designs was held in Leuven, Belgium from Nov. 13-16, 1989. The workshop was a success. There were more than 150 attendants from 19 countries. The workshop had three major events: regular paper presentation, poster presentation and system demonstration. 30 papers were presented in regular paper presentation. 20 papers were presented in a two-hour poster presentation. Over 20 systems were displayed in demonstration sessions. The proceedings of the workshop will be published by Elsevier Science Publishers B.V. in 1990.

The prevailing theme of the workshop was that the traditional CAD methodologies were not enough to cope with the fast growing complexity of VLSI designs therefore formal methods should play more important role in VLSI design.

There are two major categories of the papers presented in the workshop. One is formal verification of designs and another is formal design methodology. Verification and verification oriented papers outnumbered those of formal design methodology.

Verification of Designs

Two major types of verification research were represented at the workshop. Verification of combinational circuits and that of sequential circuits. The first type of verifications were mainly accomplished by tautology checking technique. Benchmark results of those tautology checkers were very impressive. For example, checking the correctness of a 32-bit adder, those tautology checkers usually take a magnitude of 10 seconds. The second type of verifications were mainly accomplished by using some well developed theorem provers such as Boyer-Moore theorem prover (e.g. [Pierre][1] and [Verkest]) and HOL system (e.g. [Gordon] and [Herbert]). In addition to these, timing analysis and protocol verification research are also reported in the workshop.

Formal Design Methodology

[1]Throughout of this report, the occurrences of "[...]" represent papers whose primary authors' names are quoted between "[" and "]". They can be found in the proceedings of the workshop.

Formal design methods were discussed in the workshop. Most of the methods discussed had specific problem domains such as regular structure architectures [Luk], pipelined architecture [Gopalakrishnan]. Generalized formal approach to circuit design were discussed in [Hanna] and [Finn]. The formal synthesis system based on the theory in [Finn] were demonstrated at the workshop. An interesting feature of the system is that it can find, in the example demonstrated, an appropriate operation unit to implement the abstract function symbol in specification through second order unification. [Knapp] presented a system whose specification has three types of separate but linked representation for behavior, timing and structure. Since the specification has redundant information, the system can detect incorrect design decisions.

Other Research

[Brock] presents a hierarchical, occurrence-oriented, combinational hardware description language which is formalized by the Boyer-Moore logic. A combinational circuits are represented by list of constants in the Boyer-Moore logic. The approach allows the direct verification of circuit specification, as well as allowing the verification of circuit generating functions.

Acknowledgment

I am grateful to ACM/SIGDA for its providing me a travel grant to attend the workshop.

Report on HICSS-22 Experience

by
Yi Cheng

I found the exchange of information at the conference most rewarding. From these discussions I found new insights into the complexity model for my paper as well as new ideas about the compaction algorithm in my paper.

I found the "Logical Modelling" and the "VHDL" tutorials to be very informative.

I wish to express my gratitude to SIGDA for giving me the opportunity to attend the HICSS-22 Conference to present my paper entitled: "A New Method for Two Dimensional Symbolic Compaction of IC Layout".

DA STANDARDS ACTIVITIES

CFI Newsline

FEBRUARY 1990 *Published by: CAD Framework Initiative, Inc.* Vol. 2, Issue 1

1989 - THE YEAR IN REVIEW

by Andy Graham, Motorola, Inc.

For many contributors to the CFI, last year represented a significant turning point. Continued high levels of participation and a focus on solving practical tool integration problems have given members confidence in developing a workable set of guidelines. As a result, our mission was refined as follows- "To create a free market model for EDA tools and their supporting framework environments, via development of effective industry guidelines that remove barriers to integration." In keeping with this mission, key highlights of the past year are summarized below:

- Architecture Technical Subcommittee (A-TSC) formed - The formation of the TSC following the initial work of the original six TSC's resulted in a more clearly defined agenda of interdependencies to be resolved by CFI as a whole.
- Requirements Specifications Published - These initial documents published at ICCAD in November will be the basis of evaluating candidates for adoption in the first draft of the guidelines.
- DAC '90 Integration Project Planned - This first experiment aimed at solving the practical problem of passing netlist data through a common procedural interface has drawn commitment from over 18 CFI member companies. Excellent work from the MCC-CFL, Design Representation and Architecture TSCs has had the contagious effect of accelerating our overall progress.
- Worldwide Cooperation Model Defined - An inaugural TCC meeting in Europe resulted in a plan for effective International contributions of outstanding work from a number of framework related efforts. In addition, Tom Rhyne visited Japan in November to initiate discussions on supporting active participation of our growing number of members there.

Looking forward to 1990, our challenges will be to conclude our Integration Project at DAC '90 successfully and to deliver Version 1.0 of the CFI guidelines by ICCAD in November of this year. In addition, we will be funding programs to get timely information to our members and to present an accurate view of our progress to the interested public. The accomplishments last year notwithstanding, 1990 will be the pivotal year in determining the future success of the CFI. With the support of our membership continuing on the current trajectory, I remain confident in this success.

JAPANESE CFI MEMBERS HOST CFI OFFICER

by Tom Rhyne, MCC

Tom Rhyne, the CFI Treasurer, represented CFI in a recent visit to Tokyo. He met with representatives of the eight Japanese CFI member companies from December 5 - 8, 1989. The CFI Japanese member companies present for this first group meeting were: Sony Corp., NEC Corp., Matsushita Electric Industry Co., Fujitsu Microelectronics, Seiko Instruments, Mitsubishi Electric Corp., Sharp Microelectronics, and Toshiba. Discussion focused on ways for the Japanese companies to become more involved in the technical activities of the CFI Technical Subcommittees. An annual CFI technical meeting in the Far East is a possible approach, much as the CFI has already decided to establish a CFI meeting in Europe each year.

Atsushi Akera of MCC accompanied Mr. Rhyne to Japan, and Mr. Takahide Inoue of the Sony Corporation arranged their visit. While in Japan, Tom also presented an overview of the CFI at a meeting of the SIG-DA Committee of the Information Processing Society of Japan and to a group of Japanese CAD vendors. They also met with CAD specialists at Sony and at NEC, where he was hosted by Yoshishige Kitamura.

The attendees at the IPSJ meeting were surveyed regarding their knowledge of and participation in CFI. CFI awareness in this meeting was 79%, and all expressed interest in keeping abreast of future CFI activities. Most of the attendees, 91%, expressed an interest in the creation of a Japanese access point for the CFI. All agreed that framework technology was important to CAD developers and users and that the CFI is playing an important role in that area.

IMPORTANT DATES

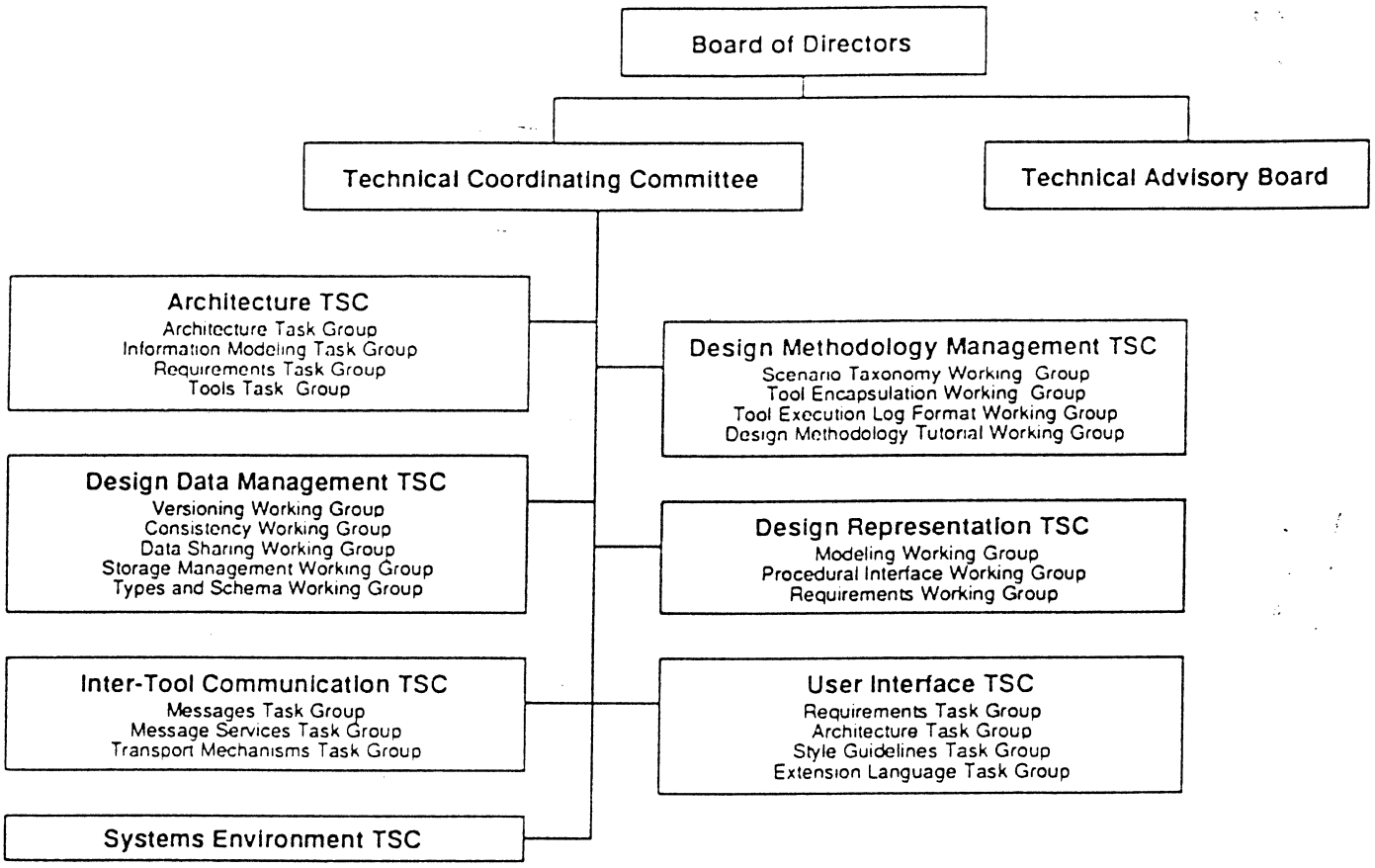
April 3, 1990: Austin, TX - DAC '90 Project Dry Run at MCC.

April 17 - 19, 1990: Denver Marriott Hotel, CO - Architecture subcommittee will meet Tues., April 17th. All other subcommittees will meet Wed., April 18th and the first half of Thurs. April 19th. Architecture will recap the second half of Thur., April 19th.

June 27 - 29, 1990: Stouffers Hotel, Orlando, FL - Architecture subcommittee will meet Wed., June 27th, 7:00pm. All other subcommittees will meet Thur., June 28th, 8:00am - 5:00pm. Board of Directors meeting will take place Fri., June 29th, from 8:00am - 2:00pm(approximate adjournment time).

CFI Leafs Out!

Last February, when the CFI technical work started, there was a Board of Directors, Technical Advisory Board, Technical Coordinating Committee and six Technical Subcommittees (TSC): Design Data Management, Design Representation, Design Methodology Management, Inter-Tool Communication, Systems Environment and User Interface. Since then, the Architecture TSC was added and many of the TSC's have formed working groups or task groups. The structure of the CFI technical committees is very dynamic. The following structure is accurate as of the beginning of the February 1990 Board of Directors meeting.



Architecture TSC

Chair: Paul Magelli, AT&T Bell Laboratories, gilligan@ihlpe.att.com, (708) 979-7172

Architecture Task Group: Define a reference model for a CAD framework.
Chair: Paul Magelli, AT&T Bell Laboratories

Information Modeling Task Group: Investigate information modeling methodologies and recommend one for use by CFI.
Chair: Glen Fullmer, Motorola

Requirements Task Group: Identify and document the users, goals, objectives, and requirements for a CAD framework.
Chair: Paul Painter, MCC

Tools Task Group: Investigate the requirements that CAD tools place on a CAD framework and the 'tool to tool' and 'tool to framework' communication in a framework.
Chair: John Dawson, Teradyne

Systems Environment TSC

Chair: Keith Davis, Hewlett-Packard, khd@hpfela.hp.com, (303) 229-6067
Secretary: Michael Gzowski, Siemans Components, intersil!siecomp!gzowski

Design Data Management TSC

Chair: Earl Ecklund, Mentor Graphics, earle@mentor.com, (503) 626-7000
Vice Chair: Debbie Lienhart, Hewlett-Packard, debbie@hpfcla.hp.com, (303) 229-4227

Versioning Working Group: Versions and configurations.
Chair: David Hogan, Texas Instruments

Data Sharing Working Group: Workspaces, visibility of data, concurrency, libraries.
Chair: Steve Banks, Valid Logic Systems

Consistency Working Group: Data integrity, rules, policies, dependency, notification.
Chair: Mark Hartoog, VLSI Technology

Types and Schema Working Group: Definition of meta-data.
Chair: Gordon Landis, Object Design

Storage Management: Specify the interface to the Storage Manager, a general purpose mechanism providing access to data entities and relationships.
Chair: Andrew Wade, Objectivity

Design Methodology Management TSC

Chair: Kenneth Fiduk, MCC, fiduk@mcc.com, (512)338-3518

Scenario Taxonomy Working Group
Chair: Manny Gonzales, Mentor Graphics

Tool Encapsulation Working Group
Chair: Eileen Perez, Digital Equipment Corporation

Tool Execution Log Format Working Group
Chair: Merrill Cornish, Texas Instruments

Design Methodology Tutorial Working Group
Chair: unassigned

Design Representation TSC

Chair: Mike Meyer, Cadence Design Systems

Modeling Working Group: Extend the netlist schema in the following areas:
Versions, Configurations and Libraries: Chair: Ted Manahan, Hewlett-Packard
Buses and Bundles: Chair: Arny Goldfein, Cadence Design Systems
Views and Schematics: Chair: Tom Smith, Digital Equipment Corporation
Hierarchical Structure: Chair: Robert Griffith, IBM

Procedural Interface Group: Define language bindings for access to the models defined by the Modeling Group
Error handling for DAC '90: Chair: Reid Madsen, Silicon Compiler Systems
DAC '90 PI Review: Chair: Ron Mazdra, Cadence Design Systems

Requirements Working Group: Refine the Design Representation requirements.
Chair: Jim Wilmore, Hewlett-Packard

Inter-Tool Communication TSC

Chair: Ed Guy, Digital Equipment Corporation, guy@mpgs.dec.com, (315 443-4228
European Chair: Leo L. M. Nabben, Phillips Components, mcvax!phcoms.seri.philips.nl!nabben@uunet.uu.net, 31 40 72 42 04
Vice Chair: George Tatge, Hewlett-Packard, gt@hpfcses.hp.com, (303) 229-3585

Messages Task Group
Chair: unassigned

Message Services Task Group
Chair: unassigned

Transport Mechanisms Task Group
Chair: unassigned

User Interface TSC

Chair: Jean M. Brouwers, EDA Systems Inc., decwrl!eda!jean@eda.com, (408) 986-9585
Vice-Chair: Tina Timmerman, Texas Instruments, ttim@aaet.csc.ti.com, (512) 250-7343

Requirements Task Group: Document requirements for CAD user interfaces and the CFI user interface subsystem
Chair: Tina Timmerman, Texas Instruments

Architecture Task Group: Propose an architecture for the user interface subsystem
Chair: Jeff Markham, Cadence Design Systems

Style Guidelines Task Group: Propose style guidelines for CAD user interface developers.
Chair: Larry Mikkelson, AT&T Bell Laboratories

Extension Language Task Group: Propose an extension language for the CFI framework.
Chair: Mac Michaels, MCC

CALENDAR OF EVENTS

Below is a list of upcoming DA events prepared by Dr. Sunil Das of the University of Ottawa. Dr. Das is also editor of the IEEE *VLSI Technical Bulletin.* He is an associate editor of the SIGDA Newsletter

Sunil Das

International Workshop on Algorithms and Parallel VLSI Architectures, June 10-16, 1990, Pont-a-Mousson, France. *Cosponsors* : IEEE, Eurasip.

Contact : Alle-Jan van der Veen, Electrical Engineering Department, Delft University of Technology, 2628 CD Delft, The Netherlands. *Phone* : (31)1578-1442.

Ninth International Conference on Analysis and Optimization of Systems (INRIA), June 12-15, 1990, Antibes, France.

Contact : Conference Secretariat, INRIA, Service des Relations Exterieures, Domaine de Voluceau - BP 105, 78153 Le Chesnay Cedex, France. *Phone* : 33(1)-39-63-5500.

10th International Symposium on Protocol Specification, Testing, and Verification (IFIP), June 13-15, 1990, Ottawa, Ontario, Canada.

Contact : Luigi Logrippo, Computer Science Department, University of Ottawa, Ottawa, Ontario, Canada K1N 6N5.

ICPR 90, 10th International Conference on Pattern Recognition (IEEE Computer Society), June 17-21, 1990, Atlantic City, NJ.

Contact : Herbert Freeman, CAIP Center, 605 Hill, Rutgers University, New Brunswick, NJ 08903. *Phone* : (201)932-4208.

Workshop on Computer-Aided Verification, June 18-20, 1990, Princeton, NJ.

Contact : E. M. Clarke, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213-3890 ; R. P. Kurshan, AT & T Bell Laboratories, Room 2C-353, Murray Hill, NJ 07974 ; A. Pnueli, Weizmann Institute, Rehovot, Israel ; or J. Sifakis, LGI-IMAG, BP 53X, 38041 Grenoble Cedex, France.

IASTED International Symposium on Applied Modelling and Simulation (IASTED), June 18-21, 1990, Lugano, Switzerland.

Contact : Janet Bodner, IASTED Canadian Secretariat, Suite 201, 4 Parkdale Crescent N.W., Calgary, Alberta, Canada T2N 3T8. *Phone* : (403)270-3616.

DAC 90, 27th ACM/IEEE Design Automation Conference (IEEE Computer Society, ACM, SIGDA), June 25-29, 1990, Orlando, FL.

Contact : Pat Pistilli, MP Associates, 7490 Clubhouse Road, Suite 102, Boulder, CO 80301. *Phone* : (303)530-4333.

FTCS 20, 20th International Symposium on Fault-Tolerant Computing (IEEE Computer Society), June 26-28, 1990, Newcastle upon Tyne, England. *Cosponsors* : Centre for Software Reliability, British Computer Society, IEE.

Contact : Neil Speirs, Computing Laboratory, University of Newcastle upon Tyne, Newcastle upon Tyne, NE1 7RU, U.K. *Phone* : 44(91)232-8511.

SIGGraph 90, 17th Conference on Computer Graphics and Interactive Techniques (IEEE Computer Society), August 6-10, 1990, Dallas, TX. *Cosponsor* : ACM.

Contact : ACM, 11 West 42nd Street, New York, NY 10036. *Phone* : (212)869-7440.

33rd Midwest Symposium on Circuits and Systems (University of Calgary, University of Alberta), August 12-14, 1990, Calgary, Alberta, Canada. *Cosponsors* : IEEE Circuits and Systems Society *et al.*

Contact : R. M. Rangayyan, University of Calgary, Electrical Engineering Department, 2500 University Drive, N.W., Calgary, Alberta, Canada T2N 1N4. *Phone* : (403)220-6745.

Fourth Digital Signal Processing Workshop (IEEE Signal Processing Society), September 16-19, 1990, New Paltz, NY.

Contact : K. S. Arun, Coordinated Science Laboratory, University of Illinois, Urbana, IL 61801.

Micro 23, 23rd Symposium and Workshop on Microprogramming and Microarchitecture (IEEE Computer Society), November 27-29, 1990, Orlando, FL. *Cosponsor* : ACM.

Contact : Chris Papachristou, Case Western Reserve University, Computer Engineering and Science Department, Cleveland, OH 44106. *Phone* : (216)368-5277.

CASE 90, Fourth International Workshop on Computer-Aided Software Engineering (IEEE Computer Society), December 5-8, 1990, Irvine, CA.

Contact : Elliott J. Chikofsky, Radius Systems, 75 Lexington Street, Burlington, MA 01803. *Phone* : (617)494-8200.

SIGDA SUMMER MEETING

acm SIGDA

at DAC-90

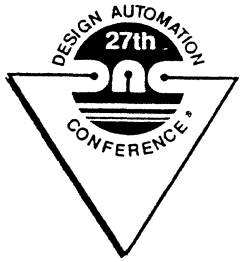
Sunday Evening, June 24 1990

Room 6 of the Convention Center

7:00 to 9:00 PM

In Orlando, Florida

Food and beverages (alcoholic and non-alcoholic) will be served



IEEE INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN

ICCAD-90

A CONFERENCE FOR THE EE CAD PROFESSIONAL

NOVEMBER 11-15, 1990
SANTA CLARA, CA

sponsored by
IEEE CIRCUITS AND SYSTEMS SOCIETY
THE IEEE COMPUTER SOCIETY

CALL FOR PAPERS

The 1990 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN will be held November 11-15, 1990. ICCAD is orlented towards Electrical Engineering CAD professionals, concentrating on CAD for Electronic Circuit Design.

AREAS OF INTEREST

Original technical papers on (but not limited to) the following topics are invited:

1)HIGH LEVEL SIMULATION: Functional, Behavioral, Architectural, Mixed-mode, Fault Simulation, HDL

2)SIMULATION: Timing, Circuit, Device, Process Simulation, Modeling, Manufacturability, Hardware Acceleration

3)HIGH LEVEL SYNTHESIS: Functional, Behavioral, Architectural Synthesis, Silicon Compilation, DSP Synthesis, System Design, HDL

4)LOGIC SYNTHESIS: Combinatorial, Synchronous, Asynchronous Logic Synthesis, Technology Mapping, Verification, Finite State Machine Synthesis, Optimal Clocking

5)LAYOUT VERIFICATION/ANALOG CIRCUIT DESIGN: Circuit Extraction/Verification, DRC, ERC, Symbolic Design and Compaction, Module Generation, Analog Circuit Synthesis

6)PLACEMENT AND FLOORPLANNING: Placement, Floorplanning, Partitioning, Area Estimation, Cell Layout, Layout Systems, Hardware Acceleration

7)ROUTING: Routing for LSI, PCB and Multichip Substrate, Hardware Acceleration

8)TESTING: Design for Testability, ATPG, BIST, Fault Diagnosis

9)CAD FRAMEWORKS: Tool Integration, Data Conversion, User Interfaces, DataBases, Design Languages, CASE

AUTHOR INFORMATION

Authors should submit 12 COPIES of:

1) a one-paragraph abstract.

2) a more detailed description not to exceed 18 double-spaced pages, figures and tables included. Excessively long submissions and previously published papers will be returned to the authors.

FORMAT

The ONE-PARAGRAPH ABSTRACT, typed on one separate page, should clearly and precisely state what is new and point out the significant results. Succinctness is required since this paragraph may be included in the Advance Program. In the detailed description, the author must objectively address why the proposed contribution is superior to prior work or what the significance of the contribution is, if breaking new ground. Demonstration of superiority in algorithms and strategies with heuristics is required through a description of the programming implementation and application to "real" problems. Additional mathematical proofs are welcome. The contribution should address an area of current technical interest to the CAD professional. A clear description of the new contribution, status of the work and significant examples and results should be given.

COVER PAGE REQUIREMENTS

Submissions should include, on the cover page: the title of the paper; the category 1 - 9, which most clearly matches the paper's content (see Areas of Interest); the full name, complete return address, telephone number and affiliation of each author; and clear identification of the individual to whom all communication should be addressed. In giving your return address, please consider that the communications for paper acceptance and mailing of the author's kit occur in the month of July.

AUTHOR'S SCHEDULE

Deadline for submissions:	**April 27, 1990**
Notification of acceptance:	**June 29, 1990**
Deadline for final version:	**August 10, 1990**

SEND TO: ICCAD-90 Secretary
MP Associates, Inc.
7490 Clubhouse Rd., Suite 102
Boulder, CO 80301
Telephone: (303) 530-4562

GENERAL CHAIRPERSON
Alberto Sangiovanni-Vincentelli
Department of EE and CS
University of California, Berkeley
Berkeley, CA 94720
(415) 642-4882

PROGRAM CHAIRPERSON
Satoshi Goto
C & C Info. Tech. Research Labs
NEC Corporation
1-1 Miyazaki 4-Chome
Miyamae-ku, Kawasaki, 213 Japan
011-81-44-856-2170

PUBLICATIONS CHAIRPERSON
Louise Trevillyan
T. J. Watson Research Center
IBM Corporation
Yorktown Heights, NY 10598
(914) 945-1507

CALL FOR PAPERS

INTERNATIONAL JOURNAL OF COMPUTER AIDED VLSI DESIGN

is announcing a

SPECIAL ISSUE

scheduled for early 1991

on

VLSI TESTING

With the advent of new high density electronic technologies like VLSI, the problems of testing and diagnosis of devices have been substantially intensified. To ensure the quality and reliability of such high density electronic devices evidently requires meticulous efforts directed towards testing.

This *special issue* is being planned as a forum for the dissemination of the current research and application information in the important evolving area of VLSI testing. Topics of interest include, but are not strictly limited to, the following:

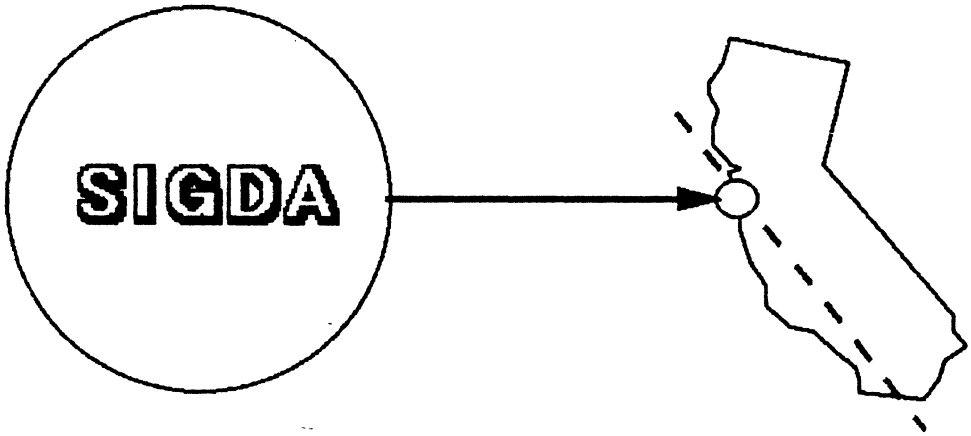
- Testing of analog and digital electronic circuits and VLSI devices
- Testing of microprocessors, memories and signal processing devices
- Fault modeling, fault simulation and test generation
- Testability analysis, design for testability and built-in-self-test (BIST)
- Fault tolerance
- AI methods and expert systems in diagnosis
- Automatic test equipment (ATE), CAD tools
- Testing wafer-scale integration devices
- Performance modeling and evaluation
- Economics of testing
- Failure mode analysis and process improvement
- Manufacturing yield and design for yield improvement
- Communication networks and protocols
- Software reliability and data structures

Prospective authors are *invited* to submit five copies of complete manuscript of high-quality research contributions that would not require *major revision* by July 31, 1990 to the *Guest Editor*:

Professor Sunil R. Das
Department of Electrical Engineering
Faculty of Engineering
UNIVERSITY OF OTTAWA
Ottawa, Ontario K1N 6N5
CANADA

Phone: (613) 564-3374
E-mail: *das@uotelg01 or srdpb@uottawa - Bitnet*
Fax: (613) 564-7681 or 564-6882

The Silicon Valley Chapter of



home of the movers and shakers in our field

First Tuesday of the Month Lunch Meetings

at
St James Infirmary
390 Moffett Blvd.
Mountain View
12:00 noon

for more info: E-mail pweil@icdc.llnl.gov

Logic Level Modeling for ASICs
Monterey Beach Hotel, Monterey, California
August 12-14, 1990
Fee: $420 (includes Registration, Hotel, and meals)
Sponsored by SIGDA of ACM
in cooperation with IEEE Circuits and Systems Society

CALL FOR PARTICIPATION

The Logic Level Modeling for ASICs Workshop will provide an interchange of ideas between people working on ASIC logic level models. The workshop will include people from ASIC foundries, CAE companies, Universities, and System designers (ie ASIC users.) There will be three major technical sessions, and two lighter evening sessions. Each technical session with be discussion oriented rather than presentations. Topics include:

-TIMING MODELS FOR ASICS
-SIGNAL REPRESENTATIONS
-TESTING ISSUES
-MODELING LARGE CELLS
-ACCURACY/ROBUSTNESS/PERFORMANCE/FEATURES
-AUTOMATION OF MODEL GENERATION (SYNTHESIS)

To maintain the workshop atmosphere and to encourage open discussions no recorders or cameras will be allowed. However, notes will be taken and a workshop report will be published in the SIGDA newsletter. The attendance will be strictly limited to 40 people, based upon first paid, first served basis with a limit of 2 attendees per company. In case of over subscription attendance preference will be given to those actively working in the field and willing to make a significant technical contribution to the workshop. Each attendee is expected to participate actively in the workshop and will be assigned some questions to be answered before the workshop, and the results will be presented during the sessions. The Sunday evening session will provide a light supper, kickoff discussions, and a chance to meet the participants.

For more information please contact any member of the Workshop Committee:

Workshop Chair	Program Chair	Arrangements Chair
Mark Glasser	Steve Bush	Colleen Matteis
Valid	VLSI Technology	
(408) 998-2430	(408)944-4811	(408)434-7624

Session Chair	Assistant Lexicographer	Lexicographer
Dr. Robert Mathews	Dr. John M. Acken	Bill Hobson
Zycad	Intel	Valid
(415) 688-7410	(408) 765-4265	(408)944-4823

acm SIGDA

IEEE Circuits and Systems Society

Please fill out and mail this application before July 6,1990; you must include payment.

Name:_____Telephone:_____

Company Affiliation:_____

Address:_____ email:_____

_____ Fax #:_____

_____

My specific current interest is _____ .

My viewpoint is that of (check one):

_____ ASIC vendor _____ CAE vendor _____ ASIC User
_____ University _____ Other, specify: _____

I am most interested in contributing and helping on the topic of(1=first choice, 2=second,etc):

_____ Timing models for ASICs
_____ Signal representations
_____ Testing Issues
_____ Modeling Large Cells
_____ Accuracy/Robustness/Performance/Fea'ures
_____ Automation of Model Generation (Synthesis)

Checks should be made out to: ACM/SIGDA Logic Level Modeling Workshop

Please return the application to:

Colleen Matteis
990 W. Taylor Street
San Jose, CA 95126
(408) 998-2430

Colleen will take care of **ALL** hotel reservations, registration, and any special requests - Please contact her and **NOT** the hotel.

Fifth International Workshop on High-Level Synthesis

Bühlerhöhe, Black Forest, West Germany
March 3-6, 1991

Co-sponsored by the ACM/SIGDA and IEEE/DATC
with the cooperation of IFIP/WG 10.2 and 10.5
and the GI/ITG Special Interest Group on Synthesis and Verification

CALL FOR PARTICIPATION

The Fifth International Workshop on High-Level Synthesis will be held on March 3-6, 1991 in Bühlerhöhe. The workshop is oriented towards design automation professionals and covers issues in the automated design of electronic circuits from high-level specifications. Areas of interest include:

- Control/Data path synthesis
- Finite state machine synthesis
- Specification languages
- Synthesis and verification
- Synthesis for testability
- Hardware/Software trade-offs
- Design constraints
- Special purpose synthesis: DSP, parallelism, pipelining, processors, communication, interfaces
- Design representation
- Design process issues: modeling, planning, optimization

To encourage a free exchange of ideas, no formal proceedings will be published nor will recording devices be permitted. Voluntarily submitted materials will be compiled and distributed to all participants. Sessions will be informal with adequate time for each presentation to support technical discussions. Workshop attendance will be limited to 70 persons.

Proposal Submission

Participants wishing to deliver a regularly scheduled presentation should submit 12 copies of an extended summary including references, significant results and contributions. Submissions should not exceed 1500 words. Participants wishing to take part in a poster session should submit 12 copies of the material to be posted on letter-size paper. An area of approximately 1 sqm will be available for poster sessions. Proposals should be submitted to the technical program chairman.

Benchmarks

To facilitate objective comparisons of tools and techniques, authors are encouraged to apply their work to a set of benchmarks. These benchmarks can be obtained from the workshop's electronic clearing house at "IILSW@decwrl.dec.com" (InterNet) or "ucbvax!decwrl!IILSW" (UseNet). If you don't get a response after a period of time, include a line like the following in the body of your message: "Net-Address: <your return path>". In general, though, it is best not to include this line. Persons without access to these networks may contact the benchmark coordinator directly.

A mailing list of parties interested in the workshop is also being maintained. To be added or deleted from this list send mail to "IILSW-request@decwrl.dec.com". To distribute mail to everybody on the list, use "IILSW-people@decwrl.dec.com".

Deadlines

Deadline for submission:	October 8, 1990
Notification of acceptance:	January 1, 1991
Pre-registration deadline:	January 25, 1991

Steering Committee

Workshop Chairperson
Raul Camposano
IBM T.J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598
phone no. (914) 945-3871
raulc@ibm.com
fax no. (914) 945-2141

Technical Program Chairperson
Wolfgang Rosenstiel
Forschungszentrum Informatik
an der Universitaet Karlsruhe (FZI)
Haid-und-Neu-Str. 10-14
D-7500 Karlsruhe, FRG
phone no. (721) 6906-402
rosenstiel@ira.uka.de
fax no. (721) 6906-909

Benchmark Coordinator
Robert N. Mayo
Western Research Laboratory
Digital Equipment Corporation
100 Hamilton Ave.
Palo Alto, CA 94301
phone no. (415) 853-6630
mayo@decwrl.dec.com
fax no. (415) 321-6953

Local Arrangements
Diana Reiter
Forschungszentrum Informatik
an der Universitaet Karlsruhe (FZI)
Haid-und-Neu-Str. 10-14
D-7500 Karlsruhe, FRG
phone no. (721) 6906-401
reiter%fix@ira.uka.de
fax no. (721) 6906-909

Technical Program Committee

Robert K. Brayton
Univ. of California at Berkeley

Daniel D. Gajski
Univ. of California at Irvine

Satoshi Goto
NEC Corporation

Giovanni De Micheli
Stanford Univ.

Gabrielle Saucier
Univ. of Grenoble

Michael McFarland
Boston College and AT&T, Bell Labs

Gert Goossens
IMEC - Leuven

Egon Hoerbst
Siemens Munich

Jukihiro Nakamura
NTT Corporation

Donald E. Thomas
Carnegie Mellon Univ.

Preliminary Announcement/Call for Papers

INTERNATIONAL WORKSHOP ON FORMAL METHODS IN VLSI DESIGN

Puerto Rico, Jan 9-11, 1991

ACM SIGDA, in cooperation with IFIP WG 10.2 - WG 10.5, IEEE TC VLSI

Background

There is increasing interest, both in academia and industry, in the application of formal methods to the design of integrated systems. Some of this interest has been motivated by the urgency of improving the reliability, testability and robustness of designs. The aim of this series of workshops is to bring together researchers interested in the application of formal techniques to the hardware design process. The emphasis of this year's meeting is to provide an opportunity for synergistic interaction between researchers in "traditional" CAD and those interested in formal approaches to design.

Focus

Papers describing original work in all aspects of formal hardware design methods are invited. Topics include, but are not limited to:

- formally based automated/interactive synthesis methods
- formal hardware verification methods
- models for timing specification and verification
- high level specification techniques (with well defined semantics)
- hardware description languages
- use of theorem provers for verification
- design for verifiability
- correctness preserving transformations
- formal approaches to design/synthesis for testability
- microprogram verification
- practical experiences
- formal models for design

Prior Workshops

This workshop is intended as a series in North America to complement a corresponding series held in Europe. The latter series has been organized within the scope of IFIP WG 10.2 (System Description and Design Tools) and IFIP WG 10.5 (Very Large Scale Integration); the latest of these workshops was held at Houthalen, Belgium in November 1989.

Participation/Registration

Those interested in presenting their work should submit 8 copies of a paper (or an extended abstract) to the workshop chairman. If you would like to participate in the workshop, please submit 4 copies of (1) an abstract (1-2 pages) summarizing your research projects, (2) a provocative position statement indicating directions in which you believe the field is (or should be) headed; and (3) (optionally) a reprint of your most relevant publication for the symposium.

Attendance at the workshop will be restricted in order to promote increased interaction. Since considerable interest has been expressed, preference will be given to participants who express their interest in participating early. *Please include a phone/FAX number and an electronic mail address on the manuscript in addition to your regular mailing address.*

Authors' Schedule

- **Deadline for submission of papers: August 1, 1990.**
- **Notification of acceptance: October 15, 1990.**
- **Camera-ready paper for circulation at the workshop: November 15, 1990.**

Revised versions of selected papers may be considered for publication in a special issue of an archival journal.

Program Committee

L. Berman (IBM), D. Borrione (IMAG), R. Bryant (CMU), R. Camposano (IBM), S. K. Chin (Syracuse), L. Claesen (IMEC, European co-chair), S. Devadas (MIT), D. Dill (Stanford), H. Eveking (Darmstadt), M. Fourman (Edinburgh), W. Hunt (Computational Logic Inc.), K. Keutzer (AT&T), G. Milne (Strathclyde), P. Prinetto (Torino), A. Sangiovanni-Vincentelli (Berkeley), P. A. Subrahmanyam (AT&T, Workshop Chair).

Mail submissions to: P.A.Subrahmanyam, Rm 4E-530, AT&T Bell Labs, Holmdel, N.J. 07733. Phone: (201)-949-5812. e-mail: subra@vax135.att.com. Fax: 201-949-3697 (also 201-949-9118).

TAU '90

1990 ACM International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems
August 15 - 17, 1990
The University of British Columbia
Vancouver, British Columbia, Canada

Call for Papers

Timing issues have recently risen to the forefront of concern in all areas of the synthesis and analysis of digital integrated circuits. This is prompted by increased industrial interest in synthesis for performance, and because it is becoming more apparent that temporal and functional behavior cannot be effectively separated. Research in these issues is also complicated by the lack of accurate delay models sufficient for rapid computation and near optimal synthesis.

Contributions are sought describing work in all areas of temporal behaviour of digital systems, particularly in the following areas:

- **Timing Specification:** Petri nets, various forms of temporal logic, general methods of specifying timing information at the combinational, sequential, and behavioral levels.
- **Synthesis for Performance:** packaging issues, system-level partitioning, architectural transforms, asynchronous techniques, retiming, logic resynthesis for delay, transistor sizing/buffering, technology mapping for delay/area, optimal clocking methods.
- **Testing and Verification:** timing verification and its interaction with testing, testability for delay, delay faults, delay-irredundant faults, the false path problem.

Authors should submit, by **April 1, 1990**, 6 copies of draft papers not exceeding 15 pages. At the workshop, a proceedings will be distributed which includes a final version of each paper, but will not be separately published. Submissions should be sent to:

Robert K. Brayton,
Chair, Technical Program Committee, Tau '90
Department of Electrical Engineering and Computer Science,
University of California at Berkeley,
Berkeley, CA 94720

Authors will be informed of acceptance by **May 15, 1990.** The workshop is organized by the Computer Science Department of the University of British Columbia and sponsored by the Association for Computing Machinery/Special Interest Group on Design Automation. The conference chairman is Patrick C. McGeer of the Computer Science Department of the University of British Columbia.

Technical Program Committee
Robert K. Brayton, UC-Berkeley (chair)
Jacob Abraham, University of Texas at Austin
Gaetano Borriello, University of Washington
Randall Bryant, Carnegie-Mellon University
Jochen Jess, Eindhoven University
Kurt Keutzer, AT&T Bell Laboratories
Michael R. Lightner, University of Colorado
Patrick C. McGeer, University of British Columbia
Teresa Meng, Stanford University
A. Richard Newton, UC-Berkeley
Nicholas Rumin, McGill University
Alberto L. Sangiovanni-Vincentelli, UC-Berkeley
Louise Trevillyan, IBM TJ Watson Research Center

FIRST CALL FOR PAPERS

Advanced Research in VLSI Conference

University of California, Santa Cruz

26-28 March 1991

The field of VLSI (Very Large Scale Integration) involves the design, manufacture, and use of systems constructed from highly complex integrated circuits. The field comprises a number of interrelated technical disciplines including semiconductor devices and processing, circuit and system design, computer-aided tools and design automation, and systems architecture. The most successful advances in any of these areas come from an understanding of the other related areas. The Advanced Research in VLSI Conference has always been a multi-disciplinary conference, with papers covering all of the above fields.

This conference is the thirteenth in a series that has been held at Caltech, MIT, University of North Carolina, and Stanford. This time it will be held at the University of California in Santa Cruz and co-sponsored by U.C. Berkeley. As in the past, the main goal of the conference is to promote interaction among researchers in the various disciplines listed above. However, in 1991, we would like to focus the conference around the main theme of "Systems Design and Integration."

We thus welcome in particular original research papers describing theory or practice relating to one or more of the following areas:

Systems: Architectures that cater to the special characteristics of VLSI technology and packaging, systems on a chip; systems integration issues such as partitioning, interfaces, clocking, and synchronization; fault tolerance, simulation, testing, and formal verification methods.

Theory: Models of computation suitable for implementation in VLSI; massively parallel computational algorithms; metrics for evaluating the complexity of algorithms or systems;

Tools: Systems modeling tools that permit experimentation with different partitionings and architectures; languages to specify or describe systems interfaces, methods and tools for behavioral synthesis, design data bases and data management frameworks.

Technology: Innovative sensor and actuator devices and circuits; integration of analog and digital circuits; power consumption, distribution, and cooling; wafer-scale integration and packaging.

But, as always, we look forward in particular to papers presenting exciting new ideas on any aspect of VLSI that may not fit into a specific category and which may even open up new problem domains for people to work on. The conference is not intended as a showcase for established systems and CAD tools, but as a forum for people with new ideas. We are interested in sparking the imaginations of the top researchers in VLSI.

The breadth of the conference guarantees that for any particular paper there will be both experts and novices in the audience. It is thus essential that papers be presented with enough background to be understood by the majority of the attendees and with enough exciting new material to please the experts.

Send 5 copies of draft papers (not to exceed 15 pages) by November 1, 1990 to

Prof. Carlo H. Sequin
University of California
CS Division, 529B Evans Hall
Berkeley, CA 94720

GENERAL CHAIRS

Dr. Vishwani D. Agrawal
AT&T Bell Labs (Rm 2C-476)
600 Mountain Avenue
Murray Hill, NJ 07974, USA
(201) 582-4349, 582-5857 (FAX)
va@research.att.com

Dr. A. Prabhakar
Microelect. & Comp. Div.
Indian Telephone Industries
Bangalore 560016, India
(0812) 567021, 572824 (FAX)

ORGANIZING COMM. CHAIR

Asoke K. Laha
Cadence Design Systems, Inc.
Advanced CAE Division
2 Lowell Research Center Dr.
Lowell, MA 01852-4995, USA
(508) 458-1900, 836-3981 (FAX)
uunet.UU.NET!cadence!gda!laha

PROGRAM CHAIRS

Dr. Lalit M. Patnaik
Comp. Sci. & Automation
Indian Inst. of Science
Bangalore 560012, India
(0812) 342451
!uunet!shakti!turing!lalit@shakti.uu.net

Dr. Adit D. Singh
Elect. & Comp. Eng. Dept.
Univ. of Massachusetts
Amherst, MA 01003, USA
(413) 545 0188, 545-0724 (FAX)
singh@ecs.umass.edu

TUTORIALS CHAIRS

Dr. Anshul Kumar
Comp. Sci. & Eng. Dept.
Indian Inst. of Tech.
New Delhi 110016, India
(011) 656272
!uunet!shakti!vikram!netearth!anshul

Dr. Vason P. Srini
Comp. Sci. Div., EECS (Evans Hall)
Univ. of California
Berkeley, CA 94720, USA
(415) 642-8900 or 527-7183
srini@ernie.berkeley.edu

EXHIBITS CHAIR

G. H. Visweswara
Microelect. & Comp. Div.
Indian Telephone Industries
Banglore 560016, India

PUBLICITY CHAIRS

Dr. Yashwant K. Malaiya
Computer Sci. Dept.
Colorado State Univ.
Ft. Collins, CO 80523, USA
(303) 491-7031, 491-2293 (FAX)
malaiya@ravi.cs.colostate.edu

Debashis Roy Chowdhury
Gateway Design Auto. (India)
SDF#A-1
NOIDA export processing zone
PO. NEPZ, NOIDA 201305
U.P. India
(05736) 62342, 62343 (FAX)

CALL FOR PAPERS AND PARTICIPATION

VLSI DESIGN '91

THE FOURTH CSI/IEEE INTERNATIONAL SYMPOSIUM ON VLSI DESIGN

Hyatt Regency, New Delhi, India

January 5-8, 1991

In Cooperation with:

IEEE COMPUTER SOCIETY
Technical Committees on Design Automation and VLSI

IEEE CIRCUITS AND SYSTEMS SOCIETY

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

Sponsored by:

COMPUTER SOCIETY OF INDIA (CSI)

DEPARTMENT OF ELECTRONICS (GOVERNMENT OF INDIA)

The symposium is a forum for engineers and researchers to present and discuss various aspects of VLSI design. The four-day program will consist of regular paper sessions, posters, tutorials, and industrial CAD exhibits. There will be enough opportunities for informal exchange of ideas. The papers will be published as a hard-cover book that will be available at the symposium.

TOPICS OF INTEREST (Partial List): CAD/CAE Systems, High Level Synthesis, Logic Synthesis, Fault Modeling, Test Generation, Design For Testability, Layout, Routing, Logic Simulation, Circuit Simulation, Timing Verification, Application Specific Devices, Microarchitecture, Regional/Global Perspectives, Economic Issues, and India's ASIC design environment.

PAPERS: **Six (6)** copies of previously unpublished papers should reach either of the **program chairs** by **July 15, 1990.** A manuscript should clearly state the originality of its contribution, significant results and applications. It should not exceed **ten** double-spaced pages including figures and references. The authors should identify the presenting author and include the complete address, telephone and/or FAX numbers and, when possible, email address. The papers will be selected through a review process and the authors will be notified of acceptance by September 15, 1990. Camera-ready manuscripts (limited to *six* book pages) will be due on November 15, 1990.

TUTORIALS: The symposium has run a highly successful tutorials program in the past. Four full-day tutorials are being planned. Although the final choice of topics is open at this time, speakers on analog VLSI design, design for testability, hardware description languages, and synthesis, are sought. Proposals may be submitted to either of the **tutorials chairs.** A tutorial may include a software demonstration.

EXHIBITS: The symposium provides a unique opportunity for CAD/CAE systems vendors to display their products. Since the available space may be limited, those interested should contact the **exhibits chair** as soon as possible.

First International Conference on Artificial Intelligence in Design

25–27 June 1991
Royal Museum of Scotland, Edinburgh, UK

A new international conference series on artificial intelligence in design has been established to provide a forum for the presentation and discussion of the state-of-the-art and cutting-edge research and development results in design theory and methodology.

Design is becoming a major research topic in engineering and architecture. It is the key to economic competitiveness and the fundamental precursor to manufacturing. However, our understanding of design as a process and our ability to model it are still very limited. This conference series aims to provide an international forum for developments in artificial intelligence in design.

Papers will be rigorously refereed by an international board and accepted papers will be published in book form. The conference will be structured to provide adequate time for both presentation and discussion.

Papers on the following areas related to artificial intelligence in design will be considered.

- Representing designs
- Cognitive models
- Machine learning
- Design processes
- Design creativity
- Knowledge representation
- Knowledge-based systems
- Integrated environments
- User interfaces
- Applications

A Call for Papers will be issued in mid-1990; full papers are due 16 November 1990.

Conference Chair:
John Gero, University of Sydney

Vice Chairs:
UK: **Tim Smithers, Edinburgh University**
Ken MacCallum, Strathclyde University
USA: **Barbara Hayes-Roth, Stanford University**
Duv Sriram, MIT
Japan: **Tetsuo Tomiyama, University of Tokyo**
France: **Khaldoun Zreik, CIMA**

The conference series is being sponsored by the major international journals *Computer-Aided Design* and *Knowledge-Based Systems* published by Butterworth Scientific Ltd.

For paper submissions, please contact:

John Gero, Department of Architectural and Design Science, University of Sydney, NSW 2006, Australia

Tel: 61-2-692 2328 Fax: 61-2-692 3031
E-mail: john@archsci.arch.su.oz
john%archsci.su.oz@**uunet.uu.net**

For registration details and further information, please contact:

Helen Hodge or Tom Whiting, Butterworth Scientific Ltd, Westbury House, Bury Street, Guildford, Surrey GU2 5BH, UK

Tel: 0483 300966 Fax: 0483 301563 Telex: 859556 SCITEC G

Sponsored by the
VHDL Users' Group
IEEE Computer Society

In Cooperation with the Electronic Industries Association

CALL FOR PAPERS

Fall VHDL User's Group

VHDL: A SYSTEMS DESIGN PERSPECTIVE

October 14-17, 1990
Claremont Resort
Oakland, California

The fall VHDL User's Group Meeting will be held in the beautiful Claremont Resort in Oakland, California. Our theme this fall will be the use of VHDL in designing large systems. The technical presentations for this User's Group Meeting will focus on the different ways of using VHDL in designing systems. Any presentations concerning the use of VHDL in designing, modeling, and/or documenting systems at a high level of abstraction are welcome. Presentations concerning the synthesis of systems level descriptions from gate level VHDL descriptions are especially encouraged.

Areas of interest include, but are not limited to:

Abstract Modeling	Behavioral Specifications
Verifying System Models	Validating Behavioral Models
Tools for Systems Modeling	Validating Systems Models
Formal Methods	Large Team VHDL Models
Transition to Other Tools	Testing Systems Models
Standards for Systems Models	High-Level Synthesis
Sample Systems Models	Actual Uses of VHDL for Systems

Important dates:

Abstracts Due:	July 1, 1990
Acceptance Notification:	July 15, 1990
Presentations Due:	September 1, 1990

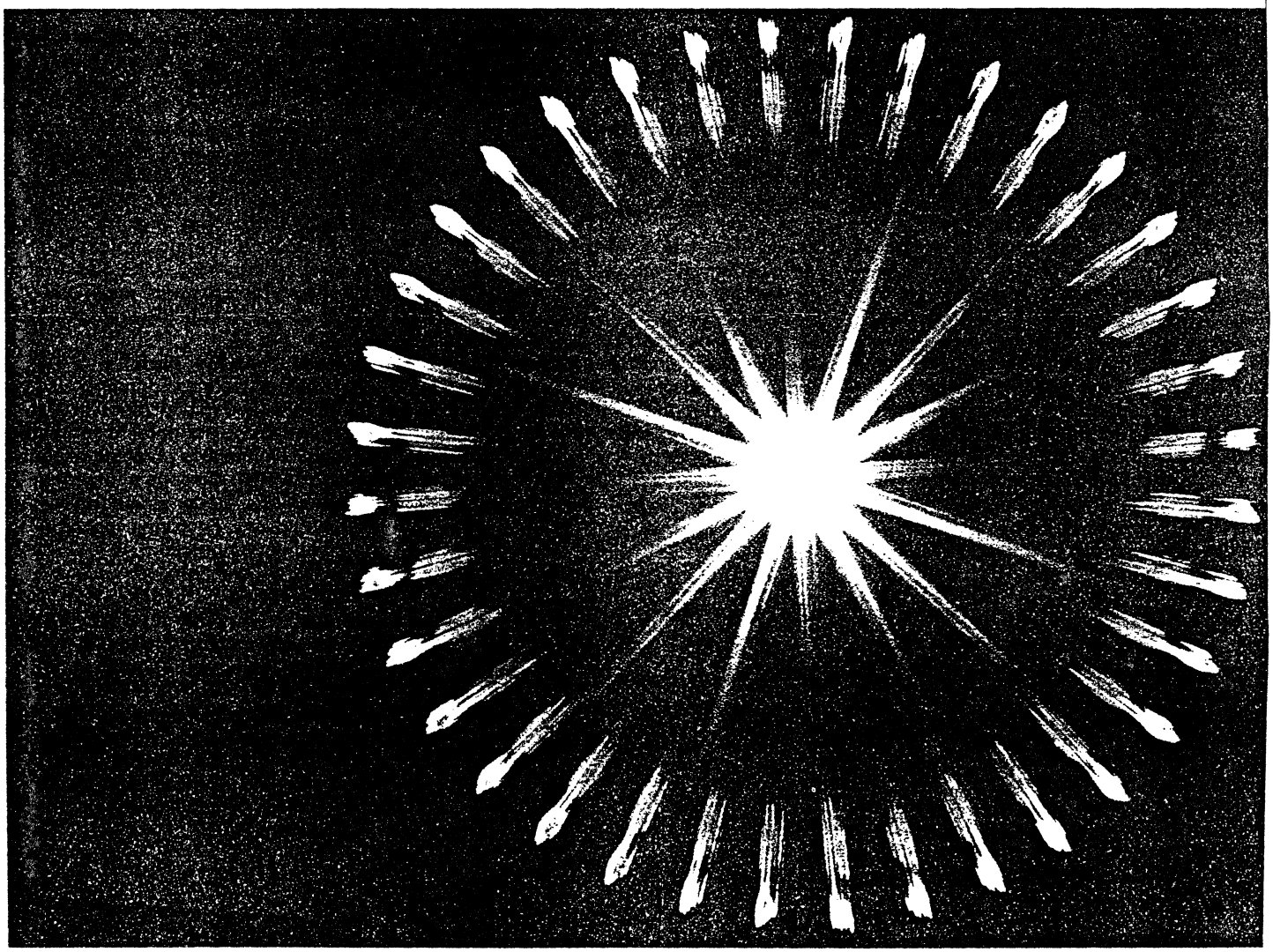
Abstracts may be sent to the program co-chairs:

David Barton
Intermetrics, Inc.
4733 Bethesda Ave.
Bethesda, MD 20814
(301) 657-3775
(301) 657-9290(fax)
barton@i2wash.com

Doug Perry
Synopsys, Inc.
1098 Alta Ave.
Mountain View, CA 94043
(415) 962-5000
(415) 965-8637(fax)

Massachusetts Institute of Technology
July 23-27, 1990
Special Summer Program 6.08s

Computer-Aided Design of Integrated Circuits: Synthesis, Simulation and Testing Techniques



Important Information

Office of the Summer Session
50 Ames Street, Room E19-356
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Telephone (617) 253-2101 (9:00 am to 4:30 pm)
Dormitory (617) 253-6561 (after 4:30 pm)
Telex 92-1473 MITCAM
FAX (617) 253-8042
FAX Verification (617) 253-2101

LATE NEWS: Report on Logic Modeling for ASICSs Workshop

Final Report for the 1989 Workshop on Logic Level Modeling for ASICs

John M. Acken, Intel Corporation
Rob Mathews, Zycad Corporation

Introduction

The goal for the Workshop on Logic Level Modeling for ASICs was to provide an interchange of ideas between people working on ASIC logic level models. The workshop discussions centered upon topics related to simulation and analysis based upon discrete logic values and discrete time steps. Participants included people from ASIC foundries, CAE companies, and System design houses, i.e., ASIC users.

LLMFA Workshop Logistics

The Logic Level Modeling for ASICs Workshop was held from December 3 through December 5 of 1989, in the Monterey Beach Hotel in Monterey, California. The workshop opened on Sunday evening with a dinner session that included introductions, definitions, and ground rules for the workshop. Three intense, half-day, technical discussions where spread over Monday and Tuesday. A less formal, but no less intense, open discussion session was held Monday evening. After Tuesday's lunch a session was held to wrap up the 1989 workshop and plan for 1990.

The workshop's success hinged upon active participation by every attendee. A wide variety of viewpoints were represented. Participating organizations were Apple, ExperTest, Fujitsu, Hitachi, Honeywell, Independent Consulting CADonomist, Intel, Mentor, Naval Post Graduate School, NEC, Next Wave, Santa Clara University, Synopsys, Texas Instruments, Valid Logic Systems, Vantage Analysis, VLSI Technology, and Zycad. In order to maintain a very candid workshop atmosphere, no formal papers or direct quotes were recorded. However, the notes that are the basis for this report were recorded by scribes from the Naval Post Graduate School and Santa Clara University.

The LLMFA 89 workshop organizing committee comprised: Dr. Robert Mathews from Zycad (Workshop Chair), Dr. John M. Acken from Intel (Program Chair), Colleen Matteis (Arrangements Chair), and Bill Hobson from Valid Logic Systems (Session Chair).

Technical Sessions

The technical sessions were freewheeling discussions, led and moderated by a session chair. Consequently, the actual discussion moved back and forth among topics; here, we will summarize each session subject by subject, instead of chronologically. Attendees answered homework questions for at least one of the sessions, and the session chair incorporated the results into the topics in the session. We tried to reach consensus when we could; some of the significant points of consensus are noted below.

Timing Models for ASICs

The Monday morning session focused on modeling timing for ASICs. The goal of the session was to explore what needs to be modeled and how to package it for design and

for simulation. As in most of the sessions, we focused on the internals of MOS ASICs, which provided more than enough material.

As it was the first full technical session, we invested a fair amount of time in learning to work together, but we nevertheless managed to cover a lot of material. Topics we discussed or at least touched on include: constraints on modeling; sources of delays, e.g., driver delays, net delays, slew, etc.; derating; accuracy and precision of models; and how to combine timing and function. This review will concentrate on areas where we achieved good consensus, surfaced interesting ideas, or got into debates.

First, we discussed the "pure" aspects of timing — sources of delays and logical models for them. Cells, wires, and I/Os each contribute their own effects. Particularly, models of interconnect timing are getting quite complicated for sub-micron processes, and that trend has strong implications for simulation systems. Derating affects structures differently, overlaying more complexity on the basic timing models. Along with intra-chip timing variations, it has implications for the accuracy and precision of timing models.

Constraints on Timing Models

The session began with the observation that modeling is intrinsically a compromise: models must be simple and efficient enough to use, but elaborate enough to assure working parts at speed. Designers can't handle all the timing information at a detailed level. To gain simplicity and to separate logical and physical design, we surrender some speed and capability at the margin. Simulators can't handle all the algorithms we'd like to use, so we surrender some accuracy in checking designs. Races and hazards imply compromises among modeling ease, efficiency, and accuracy.

Attendees contributed several other observations. First, a model is only interesting if it is supported by enough simulators. Lacking such support, it is unlikely to be widely embraced by vendors nor much used by users. Also, a model must fit with the design tools that surround it. For example, if the router treats all functionally equivalent cell inputs as equivalent, the timing model shouldn't model timing differences among these inputs.

Modeling must also be timely. Models must be simple enough that they don't hold up the release of a new process. The ASIC vendor must trade off time and effort to market against tweaks in the models to show off his process to best advantage.

Users must take care to use models in a timely way. The user who tweaks his design to fine-tune it for a particular process typically ties his design more and more tightly to the process. He may well be better off optimizing his design less, getting it done more quickly, and porting it to the next generation process to get higher performance.

Sources of Delays in MOS ASICs

The bulk of the session covered the sources of delays, models for them, and their importance. To distill the discussion, the group listed what models are required and reached consensus on a "90% solution" for MOS (see Fig. 1):

(1) fixed intrinsic delays for rise and fall times, with no load and fixed slews;

(2) capacitive load dependence, including lumped net delay;

(3) derating by overall scaling as a function of temperature, voltage, and process; and

(4) input slew rate correction.

We kicked timing constraints, such as setup and hold times, off of the 90% list for the pedantic reason that they are not models of sources of delay, not because they are unimportant. Owing to lack of time, we discussed them only a little.

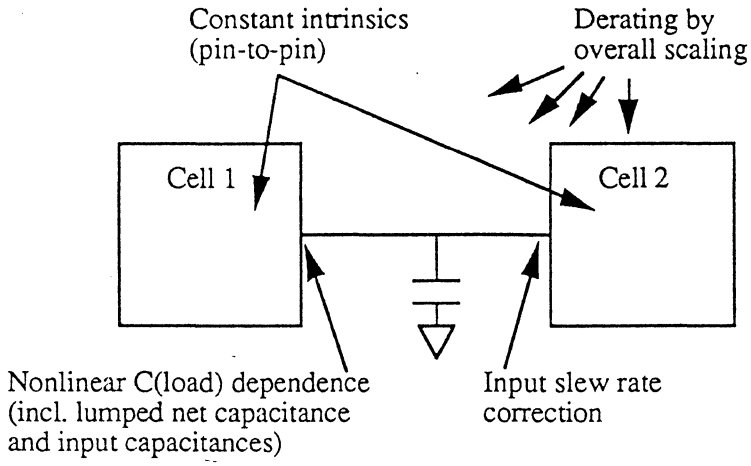


Fig. 1. The consensus "90% solution" for modeling sources of delay in MOS ASICs.

We also listed or discussed other effects that we decided not to include in the 90% solution:

- multiple simultaneously switching inputs;
- RC tree and pin-to-pin models for nets;
- varying input thresholds;
- dynamically changing capacitive loading;
- unbuffered cells;
- complex derating;
- state-dependent timing; and
- packages connected to pads.

We excluded these effects from the 90% model based more on the feeling that they were needed infrequently or were intractable to implement than because they had a minor effect on timing. For example, simultaneously switching inputs, dynamic load changes, and state dependencies can all be major effects in practice, changing a delay 50-100%.

We discussed a number of these "secondary" effects in some depth. An early topic was dynamically changing loading. A driver can see widely different capacitances at different times if it drives into pass transistors, resulting in big changes in delay times. The first modeling option is to model these dynamics in logic simulation. The trouble is that the model is inefficient and non-portable, violating our modeling constraints. So a second option is to model with circuit simulation, instead of logic simulation. Then the obvious trouble is deciding which paths to model, since full-chip circuit simulation is currently impractical. The third and most popular option is to preclude the problem by buffering pass transistors. Then there is no dynamic loading to model external to cells. This solution works best for gate arrays and worse as the implementation becomes more fully customized.

The group had a variety of opinions on modeling pads and packages. The sense of the group was that while packaging issues are increasingly important, it was unclear what effects should be modeled in logic simulation. For example, ground bounce and noise are not modeled by logic simulation; instead, you need to use circuit simulation. This area warrants more discussion.

Derating received considerable attention. Today's typical derating is an overall scaling, and its effects are large enough that it must be included in the 90% solution. One participant took the radical position that overall scaling is so coarse an approximation as to be useless; most participants disagreed. However, many did agree that overall scaling needn't be a simulation issue, since it does not change the behavior of the design. (N.B.: our discussion was limited to single-chip simulation; derating clearly is a simulation issue for multi-chip simulation with different or independent deratings for different chips.)

For submicron processes, the general feeling was that overall derating is in fact too coarse. The 90% solution for submicron derating likely involves either multiple libraries characterized at typical values of the derating parameters or derating specialized by cell. Either way, derating becomes much more complex that what is typically supported today.

Accuracy and Precision

In the Sunday introductory session, we spent some time refreshing everyone on what "accurate" and "precise" mean. The goal of simulation modeling is to provide a description that describes the behavior of a device at some level of abstraction. The accuracy of the model is measured with respect to the actual device: an accurate model describes the behavior at some level of abstraction that matches the device behavior measured at that same abstraction level. Precision, on the other hand, is measured with respect to the value representation for the level of abstraction: a precise model describes behavior with no ambiguity or calculates with many significant digits.

For ASICs, modeling more effects may lead to models that are more precise, but not necessarily any more accurate. Since high precision is pointless without corresponding accuracy, we tried to quantify accuracy for ASIC modeling.

The group agreed that delays within a die vary by about ±5% due to processing variations, setting a bound on accuracy. The variation seems to be growing as feature sizes shrink. This number implies about 1 1/3 digits of precision, i.e., delays should be less than 2 digits. Clearly, many databooks imply greater precision, yet a design that depends on more precision is likely to have problems in production.

There was no consensus on overall modeling accuracy, but one participant noted that his deisgns routinely exceed the speed predicted by simulation. Several people noted that there was a great deal of padding in overall times, probably 100% in MOS and 60% in ECL. One suggestion for contolling padding was to pad only once, rather than at each level in the modeling. However, each piece of padding is there to guarantee that some speed spec will be satisfied, so unfortunately there is no obvious way to reduce the padding without a better understanding of the sources of errors and how to account for them.

Mixing timing and function

As time ran short, we turned briefly to how to combine timing and function in models. There was no consensus on what constitutes a sufficient model, although the proposal shown in Fig. 2 seemed to come close. Most agreed that lumping delays at inputs, say, net propagation times, into pin-to-pin times for models is unacceptable because it changes the event orderings that the function sees. There was some debate as to whether the pin-to-pin times in the model needed to be state-dependent. A radical position was that the function could not in general be zero-delay: timings had to be distributed across low-level primitives, because it is too hard to characterize accurately the behavior of larger cells any other way.

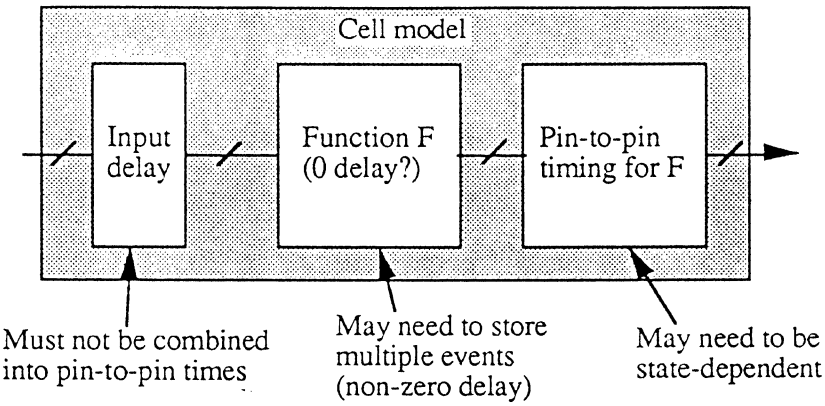


Fig. 2. An attempt to define a sufficient cell delay model. Most agreed that separate input delays are necessary. It was not clear what constituted adequate pin-to-pin timing.

Signal Values, Primitives, and Operations

The next technical session discussed signal values, primitives, and operations. The values that a signal in a logic model may assume are selected from among a fixed set known to the target simulator. The set of values made available should be consistent with requirements imposed by the types of logic primitives and operations supported by the simulator and by the user's requirement for accuracy.

We spent the majority of this session on signal representations. There has been a great deal of discussion of this topic in various VHDL working groups, but relatively few workshop participants were up to date on these.

In many cases four value logic (1, 0, X, and Z) may suffice, so we started with a simple definition for these four basic states. However, when modeling with such primitives as tristate drivers, pass transistors, and resistors, even large and complicated value systems may not be sufficient.

The discussion centered around strengths and the need for ranges of values. We discussed, or at least listed, a large number of logic systems, including the obvious 2, 3, 4, 7, 9, 12, and 15 state systems, and 46 and 136 state strength-interval systems (see Fig. 3). No consensus was reached on the minimal set of logic values. However, there was general agreement that if multiple standard sets of logic values were adopted, each larger set should be a proper superset of all smaller sets.

We purposely took a very tolerant attitude toward all possible representations for this session, leaving open the question of what representation we really felt we should endorse. We plan to address that question directly at the next workshop.

Model Verification

The session on model verification was the last of the technical sessions. The previous sessions covered some characteristics of ASICs that should be modeled; this session covered the requirements and methods for verifying those characteristics.

For purposes of this discussion, model verification meant checking that a simulation model produced expected results. Logic-level model verification meant determining that the output of a model (both the logic values and the times they occur) are correct for

a given set of inputs.

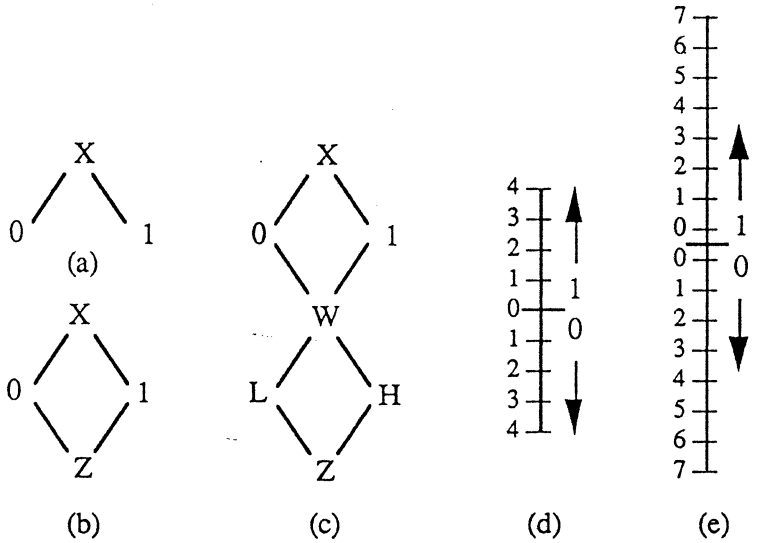


Fig. 3. Five of the many possibilities for signal states. (a), (b), and (c) are 3-, 4-, and 7-state systems; bus resolution is the common ancestor lowest in the lattice. (d) and (e) are 45- and 136-state interval systems; bus resolution involves finding the least interval that dominates.

Most participants agreed that the most common source of expected results is from a "golden" or trusted simulator. In most cases, the technology characterization people come up with very accurate SPICE models for individual cells. The circuit simulation using these cells is used to derive logic level models on the golden simulator, which defines the golden chip model. Other simulation models are compared to this model using extensive simulation. Verifying correct behavior requires a level of tolerance for simulator inconsistencies and timing differences. We were not able to formalize hard and fast rules for acceptable tolerances.

One verification task is to check that the actual ASIC matches its model. In practice, no one admitted to spending much time doing so. The ASIC customer is no longer interested in the model once the silicon is available, as long as the ASIC functions correctly and at least as fast as the model predicted. If the customer is happy, the vendor is happy and isn't likely to spend a lot of time understanding why the model is inaccurate (pessimistic).

Another verification problem is completeness, i.e., checking that the model describes all the characteristics of interest, allowing rigorous design specification. Both performance and function must be modeled and verified. A big open issue here is how to verify or compare the model's response to unknown states, since they don't exist in a real circuit (or in SPICE). They nevertheless convey important information about the model and need to be systematically verified somehow.

Finally, we discussed how to determine systematically that a model had been verified. Participants indicated that verification is done by testing a model, not by proof or other formal means. One suggestion was to use fault simulation as a means of determining whether a test exercised a design sufficiently to verify it. Clearly, good fault coverage doesn't guarantee good verification of the design, but it does provide a potentially reasonable basic standard.

The Next LLMFA Workshop

Most participants felt strongly that SIGDA should sponsor the workshop again. The next Logic Level Modeling for ASICs Workshop will be held in the Monterey Beach Hotel, Monterey, California. The workshop start Sunday evening, August 12, and continue through Tuesday afternoon, August 14. The basic format of the workshop will be the same as LLMFA 1989. The workshop will be sponsored by SIGDA of ACM, and because of the topic, will be held in cooperation with IEEE Circuits and Systems Society. The topics for LLMFA-90 include timing models and signal representations again, but also modeling large cells; the relationship among accuracy, robustness, performance, and features of models; and automatic model generation (model synthesis).

The 1990 LLMFA workshop committee includes: Mark Glasser, Steve Bush, Colleen Matteis, Dr. Robert Mathews, Dr. John M. Acken, and Bill Hobson. For information about registration for the 1990 Logic Level Modeling for ASICs Workshop, please contact the Registration Chair, Colleen Matteis, 990 W. Taylor Street, San Jose, CA 95126 (408/998-2430).

A Brief Bibliography

We asked participants to suggest references that were relevant to the workshop. Only a few contributed. Here is the list:

"DA Standards Activities" (includes several letters on signal representations), *SIGDA Newsletter*, Volume 19, Number 1, April 1989.

"Automatic Generation of Gate Level Models with Accurate Timing," Steve Bush, *IEEE International Conference on Computer-Aided Design* (ICCAD-87), November 1987.

"VHDL Model Portability," G. M. Nurie and P. J. Mencini, *High Performance Systems*, July 1989.

"ZYGEN Timing Model Generator," J. J. Harrison, *IEEE ASIC Seminar and Exhibit*, September 1989.

"Delay Modeling in Logic Simulation," John M. Acken and Lawrence H. Goldstein, *IEEE International Conference on Circuits and Computers* (ICCC 80), October 1980.

Acknowledgements

Many people make a workshop possible, but Dr. Paul Weil, SIGDA Workshop Coordinator, was instrumental in helping us to secure SIGDA sponsorship and to cope with administrative details. Our thanks go to Paul and to ACM SIGDA for their assistance. The entire workshop extends a special thanks to Prof. Loomis at the Naval Post Graduate School and Prof. Mourad at Santa Clara University for contacting and providing the scribes for the workshop. We also thank the scribes: Shen Quan Zheng, David C. Stuart, Brian Pooler, and Hal Skoog. Finally, we would like to thank the attendees for their active and enthusiastic participation in the technical discussions.

ACM/SIGDA High School Scholarship Program

In 1988, a high school scholarship program was suggested to ACM/SIGDA which would encourage minority advancement in the fields of electrical engineering and computer science. With the approval of the ACM/SIGDA Board, in 1989 the first Advancement in Computer Science and Electrical Engineering (ACSEE) Scholarship Program was implemented. This pilot program was intended to provide experience for future development.

This is the second year of this scholarship outreach effort and based on the experience gained in administering the pilot program, it was decided that ACSEE should work in conjunction with community early intervention programs for advancement in the engineering sciences. These community outreach organizations work to tap the talent of every segment of our population by "recruiting and tracking" junior and senior high school students into math and science career curriculums. Recruiting from an often ignored segment of our student population accesses new talent-- a necessary effort launched to maintain America's position as world economic/engineering leader. To further concentrate administrative and outreach efforts, ACSEE is working with students and organizations in the San Francisco Bay Area.

ACM/SIGDA made available in the Bay Area four ACSEE Scholarships to high school seniors from underrepresented groups (Blacks, Hispanics, American Indians, women, or disabled) who wish to pursue careers in computer science or electrical engineering. Two of the scholarships are $4000 per academic year awards, renewable for up to 5 years; the other two scholarships are $1000, one time only awards offered to the top runner-ups in the scholarship competition. Recipients may plan to attend colleges anywhere in the continental United States.

Selection for awards is based on: high school grades, performance on standardized tests, recommendations and personal characteristics, financial need, and potential. CS and EE professionals from the Bay Area, who have offered to act as student mentors, will assist in selecting recipients from a pool of applicants. After selection, recipients will be paired with mentors according to similar background. The four recipients will be awarded scholarships during the Keynote Ceremonies at the 27th Design Automation Conference, this year in Orlando, Florida. On Keynote day the mentors will show the students around the 27th DAC exhibits.

Last year, mentors proved to be great assets to ACSEE, the students, and the high schools ACSEE worked with in the Las Vegas area. Not only did the mentors act as support for the student, assisting in formulating college plans, encouraging the student to meet college application deadlines, take college entrance exams, etc. the mentors also visited the student at his/her school and spoke to math and science classes about careers in CS and EE. Mentor Robbie McClellan, of DASIX, tells stories of students following him out of classrooms, down hallways and even escorting him to his car in zealous efforts to ask more questions. And mentor Meyleen Beichler, also of DASIX, found her knowledge of elementary PASCAL handy as students in computer classes discussed their programming problems with her.

In keeping with what we hope will develop into a strong tradition, mentors will continue to take an active role in both the student's career goals and community outreach. This year's mentors will be giving career oriented presentations to Bay Area high schools to help publicize the scholarship and interest other students in engineering. Additionally, after meeting at DAC, mentor and student will keep in contact during the student's college years. Mentors will provide encouragement and assistance where possible, act as a role model, and help the student to understand more about their chosen career.

If you are interested in becoming a mentor, please contact: Charlotte Acken, ACSEE Director, (415) 294-3248, e-mail cacken@sandia.llnl.gov.

1989 Recipients

Maria de los Angeles Nolasco	Moapa Valley High School	$4000/renewable
Kim Tuyen Dinh	Valley High School	$4000/renewable
Jon Eugene Duran	Chaparral High School	$1000

Many thanks to the 1989 mentors:

Meyleen Beichler	DAZIX
Al Jimenez	Nextwave Design Automation
Robbie McClellan	DAZIX
Anita Whichard	Lawrence Berkeley Labs

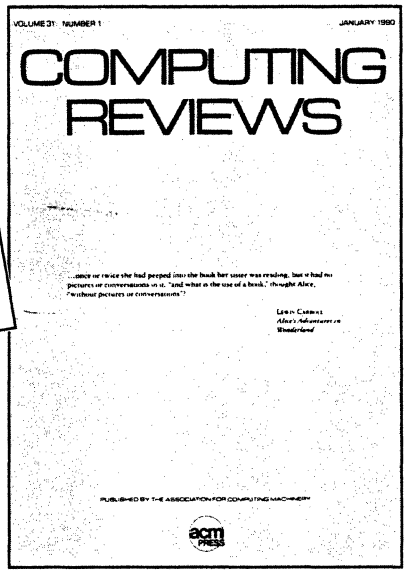
An easy and convenient way to keep up with new developments...

Computing Reviews

(Monthly)

Available Online:

Now you can access and search *Computing Reviews* online through STN's COMPUSCIENCE or AMS' Math\Sci.



Editor-in-Chief: **Aaron Finerman,** *University of Michigan,* Ann Arbor, MI

Computing Reviews aims to furnish computer oriented persons in mathematics, engineering, the natural and social sciences, the humanities, and other fields with critical information about current publications in any area of the computing sciences, and to further, thereby, the development of the computing sciences as a discipline, as an art, and as a tool for revolutionizing our technology and our patterns of thinking.

Computing Reviews can be a great help to the librarian who is building a collection of books, journals, and nonprint media in computer science. Each month, *Computing Reviews* brings to its readers critical commentaries on the most important articles published in over 300 computer science journals (included in the list of 450 journals scanned for *Computing Reviews* and *The ACM Guide to Computing Literature*). In addition, thousands of books, proceedings, and conference papers are scanned each year, and the most important contributions to the literature are selected and reviewed in *Computing Reviews.*

The following categories are routinely covered:

- general literature
- hardware
- computer systems organization
- software
- data
- theory of computation
- mathematics of computing
- information systems
- computing methodologies
- algebraic manipulation
- artificial intelligence
- computer graphics
- image processing
- pattern recognition
- simulation and modeling
- text processing
- computer applications
- computing milieux

Computing Reviews enables computer professionals, researchers, scientists, mathematicians, engineers, social scientists, educators, librarians, and students to monitor new developments quickly, reduce information and source overload, keep up to date on specific topics, locate citations of relevant topics, evaluate potential source materials, and compile a productive reading list for personal or professional use. For the librarian, it is an excellent tool in collection development.

Each review includes a complete citation, and consists entirely of the reviewer's opinions.

Volume 31 (1990) ISSN: 0010-4884
• ACM Order No.: **103000** • Monthly.
About 720 pages a year.

Nonmembers	$115.00
ACM Member	$32.00
Single Issues: Nonmembers	$14.00
ACM Members	$7.00
Back Volumes: Nonmembers	$168.00
ACM Members	$84.00
Student Members	$27.00

Please Send All Orders and Inquiries to:
P.O. Box 12115
Church Street Station
New York, NY 10249

The Society for Computing and Information Processing

Membership Application

Mail to: ACM, P.O. Box 12114, Church Street Station, New York, NY 10257

You may use this ACM Membership Application to: 1) Join ACM and take advantage of the special Publication and SIG rates for ACM members; 2) Join the Special Interest Group(s) of your choice at the SIG Member (non-ACM member) rates on the reverse side; individuals joining SIGs using these rates are not entitled to ACM membership privileges.

Personal Information

■ Please Print ■

First, Middle Initial, Last Name

Address

Address

Address

City/State/Zip or
City/Country/Postal Code

Title

Address above is: ☐ Business ☐ Home

Day Telephone Number

For Office Use Only

ACM Membership

ACM Membership includes a subscription to the monthly *Communications of the ACM*

Check member class desired

☐ **$71.00 Voting Member Applicants**
You must subscribe to the Purposes of ACM; have attained professional stature as demonstrated by intellectual competence and ethical conduct in the arts and sciences of information processing; and must satisfy at least one of the requirements at the right.

1. Bachelor's Degree. Institution: _____
2. Equivalent level of education. Institution: _____
3. Four full-time years of experience (attach a brief statement).

I attest the above is correct _____
Signature

☐ **$71.00 Associate Member Applicants**
You must subscribe to the purposes of ACM. Associate Members may convert to Voting Member status at any time by writing ACM Headquarters for a "Self-Certification" form.

☐ **$21.00 Student Member Applicants**
You must be registered in an accredited educational institution and a Faculty Member must certify your full-time status.

Institution: _____
Faculty Member's Signature _____
Expected Graduation Date: ____/____
MM YY

Joint Membership Applicants

☐ **$66.00 Members of the IEEE-CS**
receive a $5 dues discount.
(not Affiliates with "N9" member #'s)

Institution: _____
Affiliation: _____

☐ **$57.00 Members of the following overseas computing societies**
ACS (Australia), AFCET (France), AICA (Italy),API (Portugal),BCS (United Kingdom), BIRA/IBRA (Belgium), CIPS(Canada),CSZ(Zimbabwe), GI (Germany), HKCS (Hong Kong), ICS (Ireland), IPA(Israel), IPSJ(Japan), NGI (Netherlands), NZCS (New Zealand), SCS (Shanghai).

☐ **Spouse Member Applicants**
Spouses who are both members may elect to receive only one subscription to *CACM* and pay the following reduced rates:

Voting Members		Student Members	
1st person + *CACM*	$71.00	1st person + *CACM*	$21.00
2nd person, no *CACM*	$48.00	2nd person, no *CACM*	$14.00

☐ **$35.00 Retired Member Applicants**
Your annual income from part time and consulting work must not exceed $2,500; your age plus years of ACM membership must exceed 75.

SIG Membership

Non-ACM Members Only

☐ **SIG Membership only. (Non-ACM Members only).**
Membership in special interest group(s) of your choice includes a newsletter subscription. SIG non-ACM members are not eligible for ACM SIG member rates or ACM membership privileges. *See "SIG Member (non-ACM) Rates" on reverse.*

Mailing List

Optional

ACM occasionally makes its membership list available to companies and societies for computer-related mailings. If you wish to restrict the use of your name for these purposes, please check one of the following:

☐ACM announcements only.
☐ACM and sister society announcements only.

For Office Use: Recruit/Promotion Code
S 0 6 0 0 1 3

Purposes of ACM

Signature Required

To advance the sciences and arts of information processing; to promote the free interchange of information about the sciences and arts of information processing both among specialists and among the public; and to develop and maintain the integrity and competence of individuals engaged in the practice of information processing.

I hereby affirm that I subscribe to the purposes of ACM and understand that my membership is not transferable.

_____ _____
Signature *Date*

Note: Membership dues include $30.00 ($20.00 for students) toward a subscription to *Communications of the ACM.*

Effective 7/90

Publications

Definitions:

Air Options: Available to overseas members. Circle desired rate(s) and add to publication subscription and/or SIG membership rate(s).

Partial Air Service-Air freight to Amsterdam and Dutch surface mail. Available only to Europe, India, Africa, and Mideast.

Full Air Service-Air service from U.S. Available to all overseas locations including Hawaii.

Circle appropriate rate(s) and indicate subtotal. Membership includes a subscription to ACM's flagship publication Communications of the ACM. If air service is desired for member copy of CACM, circle appropriate Air Option.

Add Appropriate Overseas Air Option(s) to Voting/Associate or Student Rate(s)

	Code #	Voting/ Associate	Student	Partial	Full Air	Total
Communications of the ACM (monthly) addt'l subscriptions only	101	$24.00	$19.00	+$30.00	$60.00	=...$ _____
Membership includes a subscription to *CACM*						
Journal of the ACM (quarterly)	102	21.00	16.00	+.20.00	25.00	=...$ _____
Computing Surveys (quarterly)	103	14.00	9.00	+.15.00	30.00	=...$ _____
Computing Reviews (monthly)	104	33.00	28.00	+.20.00	50.00	=...$ _____
Collected Algorithms, Vols I, II, III, IV, V & 1 yr's quarterly updating supplements	105	210.00	185.00	+.10.00	20.00	=...$ _____
ACM Guide to Computing Literature (annual)	160	86.00	81.00	no air offered		=...$ _____
You will receive the current published edition.						
Transactions on: (all quarterlies)						
Mathematical Software/TOMS	108	25.00	20.00	+.10.00	20.00	=...$ _____
Database Systems/TODS	109	24.00	19.00	+.10.00	20.00	=...$ _____
Programming Languages and Systems/TOPLAS	110	23.00	18.00	+.10.00	20.00	=...$ _____
Graphics/TOG	112	29.00	24.00	+.10.00	20.00	=...$ _____
Information Systems/TOIS	113	25.00	20.00	+.10.00	20.00	=...$ _____
Computer Systems/TOCS	114	25.00	20.00	+.10.00	20.00	=...$ _____
Software Engineering Methodology/TOSEM	115	22.00	17.00	+.10.00	20.00	=...$ _____
Modeling & Computer Simulation/TOMACS*	116	30.00	25.00	+.10.00	20.00	=...$ _____

Expected Publication Date January, 1991

PUBLICATION SUBTOTAL $ _____

Special Interest Groups (SIGs)

Note: SIG members who are not ACM members do not receive a subscription to *Communications of the ACM.*

Definitions:

Air Options: Available to overseas members. Circle desired rate(s) and add to publication subscription and/or SIG membership rate(s).

Partial Air Service-Air freight to Amsterdam and Dutch surface mail. Available only to Europe, India, Africa, and Mideast.

Full Air Service-Air service from U.S. Available to all overseas locations including Hawaii.

Circle appropriate rate(s) and indicate subtotal.
SIG Membership includes a Newsletter subscription.

Add Appropriate Overseas Air Option(s) to Voting, Associate, Student or SIG Member Rate(s)

	Code #	ACM Members Voting/ Associate	ACM Members Student	SIG Mbrs Non-ACM Mbr. Rates	Partial	Full Air	Total
SIGACT (Automata and Computability Theory)	001	$11.00	$5.50	33.00	+.$11.00	$22.00	.= $ _____
SIGADA (ADA)	037	15.00	10.00	37.00	+...22.00	44.00	.= $ _____
SIGAPL (APL)	032	15.00	7.50	40.00	+.....8.00	20.00	.= $ _____
SIGARCH (Computer Architecture)	002	20.00	10.00	45.00	+...17.00	47.00	.= $ _____
SIGART (Artificial Intelligence)	003	15.00	8.00	37.00	+....11.00	22.00	.= $ _____
SIGBDP (Business Data Processing & Management)	004	12.00	6.00	34.00	+.....6.00	9.00	.= $ _____
SIGBIO (Biomedical Computing)	005	20.00	7.00	30.00	+...10.00	16.00	.= $ _____
SIGCAPH (Computers and the Physically Handicapped-Print Edition)	006	10.00	5.00	32.00	+.....5.00	7.00	.= $ _____
SIGCAS (Computers and Society)	007	12.00	6.00	34.00	+.....7.00	10.00	.= $ _____
SIGCHI (Computer and Human Interaction)	026	20.00	10.00	42.00	+...12.00	16.00	.= $ _____
SIGCOMM (Data Communication)	008	22.00	15.00	50.00	+...10.00	40.00	.= $ _____
SIGCPR (Computer Personnel Research)	010	15.00	5.00	17.00	+.....7.00	10.00	.= $ _____
SIGCSE (Computer Science Education)	011	15.00	7.00	37.00	+...10.00	23.00	.= $ _____
SIGCUE (Computer Uses in Education)	012	15.00	7.00	50.00	+.....9.00	20.00	.= $ _____
SIGDA (Design Automation)	013	3.00	3.00	5.00	+.....Free	Free	.= $ _____
SIGDOC (Documentation)	033	12.00	2.00	34.00	+.....6.50	11.00	.= $ _____
SIGFORTH (Forth)	039	20.00	11.00	42.00	+.....8.00	11.00	.= $ _____
SIGGRAPH (Computer Graphics)	015	20.00	13.00	49.00	+...12.00	28.00	.= $ _____
SIGIR (Information Retrieval)	016	12.00	6.00	56.00	+.....5.00	11.00	.= $ _____
SIGMETRICS (Measurement & Evaluation)	019	16.00	6.00	38.00	+.....7.00	18.00	.= $ _____
SIGMICRO (Microprogramming)	020	12.00	7.00	34.00	+.....9.00	14.00	.= $ _____
SIGMOD (Management of Data)	014	20.00	12.00	23.00	+....11.00	34.00	.= $ _____
SIGNUM (Numerical Mathematics)	021	15.00	7.50	23.00	+.....8.00	11.00	.= $ _____
SIGOIS (Office Information Systems)	027	15.00	7.50	37.00	+.....8.00	17.00	.= $ _____
SIGOPS (Operation Systems)	022	15.00	8.00	35.00	+...13.00	26.00	.= $ _____
SIGPLAN (Programming Languages)	023	20.00	8.00	42.00	+...29.00	60.00	.= $ _____
FORTRAM FORUM (Fortran) Newsletter Only	038	6.00	3.00	16.00	+.....6.00	8.00	.= $ _____
LISP POINTERS (LISP) Newsletter Only	040	8.00	4.00	18.00	+.....7.00	12.00	.= $ _____
OOPS MESSENGER (Object-Oriented Programming Systems) Newsletter Only	041	6.00	3.00	14.00	+.....6.00	8.00	.= $ _____
SIGSAC (Security, Audit & Control)	036	12.00	4.00	34.00	+.....6.00	11.00	.= $ _____
SIGSAM (Symbolic & Algebraic Manipulation)	024	15.00	5.00	20.00	+.....8.00	12.00	.= $ _____
SIGSIM (Simulation)	025	12.00	5.00	34.00	+.....9.00	16.00	.= $ _____
SIGSMALL/PC (Small & Personal Computing Systems & Applications)	031	12.00	8.00	32.00	+.....8.00	17.00	.= $ _____
SIGSOFT (Software Engineering)	034	16.00	10.00	38.00	+...17.00	40.00	.= $ _____
SIGUCCS (University & College Computing Services)	028	10.00	5.00	32.00	+.....Free	Free	.= $ _____

SIG SUBTOTAL $ _____

Special ACM SIGSOFT OFFER!

Join ACM and SIGSOFT and save 50% if you subscribe to TOSEM. Pay only $11.00/8.50 (Voting and Associate/Student rates) for the first year!

Payment Information

Mail this application to:
ACM
P.O. Box 12114
Church Street Station
New York, NY 10257

ACM accepts payment by personal check, money order, cash or credit card. For overseas residents, payment must be in U.S. dollars drawn on a U.S. bank. Please make checks payable to ACM, Inc.

I wish to pay by: ☐ Check (make payable to ACM, Inc.) ☐ Credit Card

☐ American Express ☐ Master Card ☐ Visa

Card #: _____ Card Exp. Date: _____

Signature: _____

ACM Member Dues ***(from reverse side)*** **$** _____

Publication Subtotal $ _____

SIG Subtotal $ _____

GRAND TOTAL $ _____

APPLICATION FOR SIG MEMBERSHIP*

ASSOCIATION FOR COMPUTING MACHINERY
11 West 42nd Street, New York, New York 10036

Please enroll me as a member of the SPECIAL INTEREST GROUP ON DESIGN AUTOMATION.

Membership includes Newsletter subscription. Please make checks payable to:
ACM, INC., Post Office Box 12115, Church Street Station, New York, New York 10249

_____
Name (Please print or type)

_____
Company

_____
Mailing Address

_____
City State Zip

Signature _____

☐ New address. Please change my ACM record.

Annual Membership Dues:

$3.00 for ACM Members
$5.00 for Non-ACM Members
$3.00 for ACM Student Members

☐ **ACM MEMBER:** ACM Member Number _____. *(Send no money now. Dues are payable when ACM membership is renewed.)*

☐ **ACM STUDENT MEMBER:** ADM Student Member Number _____. *(Send no money now. Dues are payable when ACM membership is renewed.)*

☐ **NON-ACM MEMBER:** Bylaw 7 Section 6 of the ACM Constitution states that Non-ACM Members may be members of a SIG provided that their major professional allegiance is in a field other than information processing. *(Enclose annual dues payment of $5.00.)* Please state your major professional field: _____

☐ **SUBSCRIPTION TO SIGDA NEWSLETTER ONLY:** Enclosed is payment of $5.00 for annual subscription. *(Note: Subscription is included with membership dues.)*

☐ Please send information on ACM Membership.

---

* For ACM members renewing within the next three months, do not use this application. Simply add this SIG to your renewal invoice when you receive it and return with the appropriate additional payment.

ASSOCIATION FOR COMPUTING MACHINERY
11 WEST 42ND STREET, NEW YORK, NY 10036