Engineering for systems using large scale integration

by C. F. O'DONNELL

North American Rockwell Corporation
Anaheim, California

INTRODUCTION

Our experience in designing and producing over 100 different MOS large scale arrays (LSA's) for a variety of systems and customers has led to a number of changes in our pattern of engineering operations. We have found that the major increase in device complexity called for in LSA's requires extensive use of design aids for acceptable development schedules and costs. In turn, the generation and use of these design aids tie device designers and the logic/system designer closely together.

These design aids include the software required for checking logic, for device layout, for the generation of device and system test programs and, in combination with suitable hardware, for device mask generation.

This paper concentrates on the status of interaction between device and logic/system designers and design aids but does include probable short term trends in the evolution of engineering in the LSI area.

To illustrate the capabilities of todays' technology, we describe the Autonetics D200 computer which is built using MOS LSA's. Included in this discussion are the tradeoffs leading to its design, its characteristics and organization, and finally some details of the LSA's used in building it.

Criteria for design

To keep this talk confined to a reasonable time, I will cover only the engineering of digital systems using MOS devices in Large Scale Arrays (LSA's). As can be seen from Figure 1, in my view the change in engineering interaction as we progressed from discrete components through integrated circuits (IC's) to LSA's have been evolutionary rather than revolutionary. There has been more of a change of emphasis on interaction between engineering functions rather than the formation of completely new groups, an adaptive response of engineering to changes in the characteristics of the devices used.

The change from discrete components to integrated circuits, for example, forced logic designers to become involved in the hardware task of board layout. Board interconnection minimization became an essential ingredient of logic partitioning. With board element counts shifting from hundreds to thousands, without new logic design criteria off-board lead requirements easily outstripped physically feasible numbers of connector contacts.

With low-cost IC's available the ability to eliminate every possible resistor, diode or transistor was no longer the most significant criterion in evaluating the efficiency of a logic designer. The ability to partition logic so relatively few expensive interconnections between boards would be required had become dominant.

While LSA's provide low-cost elements, they do intensify other system development problems. One of LSA's weakest points lies in the difficulty of insuring that the first design works, and the expense and schedule slips involved in correcting design deficiencies are minimized. With discrete components, a few overstressed parts could readily be replaced with little impact on system cost or schedules. Even with integrated circuits, jumpering of connections and replacement of one standard packaged unit with another enabled the engineer to take care of the majority of his problems. When an LSA logic design error or local overstress condition occurs, then design corrections must be made, mask sets changed and a new production run of the device made before the system can be checked out.

The increased impact on system costs of logic

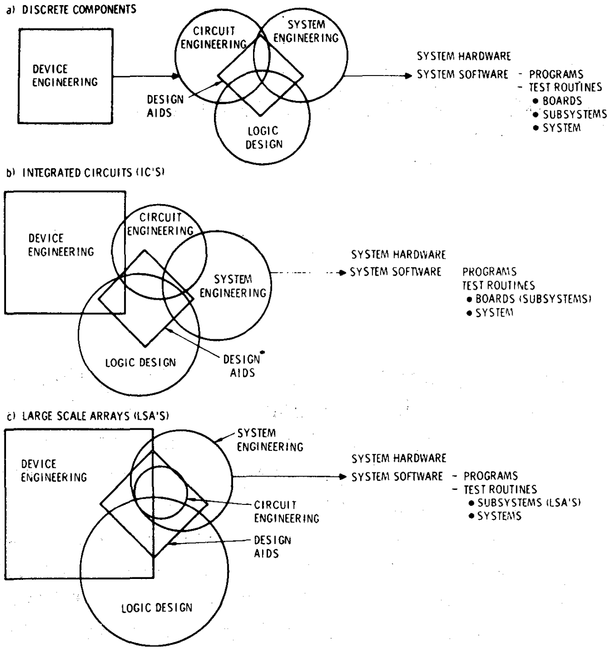


FIGURE 1—Engineering interaction, digital system design and development

errors has involved the logic designer more directly in device engineering. He has far greater impact on the device supplier than was the case in discrete component days. Each time the device supplier has moved more deeply into system fabrication he has lost some of his flexibility. With IC's he was free to design the circuit details as he saw fit provided specified operating characteristics of the device were met. However, his sales depended heavily on whether logic designers decided to use RTL, DTL or TTL circuits in their designs and the range of fan-in and fan-out ratios they required and other obvious choices. With LSA's an additional degree of flexibility has been lost. In many cases the device becomes such a specialized subsystem that it must be tailored to the needs of an individual system manufacturer.

Criteria for user

While there are problems, the incentive of producing low cost, complex electronic systems built with LSA's is sufficiently high to justify a major expenditure of engineering effort to overcome these problems. To make this objective of low cost systems feasible, the engineering system for design and development must be set up to cope with these characteristics of LSA design which differ from discrete component and IC design.

LSA's are not available from manufacturers as off-the-shelf items with the exception of a few standard functions such as shift registers. Seldom is it possible to work around a missing subsystem, which is really what a LSA is. Therefore, these devices when used can become system schedule limiting factors and major contributors to system expense. As an LSA is a subsystem, it represents a higher percentage of total system cost than previous electronic devices. Consequently, errors in original cost estimates can have greater impact on program profits than was the case for previous devices. Certainly high efficiency is called for in the system both for engineering design and development and in production.

These factors all point to the need for engineering excellence in design and accuracy in carrying engineering intent thru into hardware. Both points call for design aids and computer programs to do the detail drugery with high accuracy, and computer controlled precision equipment for tasks such as mask generation and device assembly.

This leads to the engineering configuration shown in Figure 1-C, where the circuit design function has been largely absorbed by device engineering and logic design. The growth of logic design function has been due to its direct involvement in LSA device design and its role in developing many of the design aids used by the remaining engineering functional groups.

This intensification of interaction between device supplier and user is not without its own set of problems. Their nature can be understood by examining what happened when multilayer boards were developed for IC interconnection. These boards were both more compact and uniform than the combination of two-sided printed circuit boards and cabling they replaced. What cross talk, capacitive loading and lead resistance there was remained reasonably uniform from board to board and system checkout yield improved. However, a defective board was a complex problem.

Locating and fixing board errors particularly in internal layer interconnection routings was difficult, time consuming and costly. Because this was done in the system manufacturer's shop, surface jumper connections and other quick fix techniques could be used to avoid severe schedule penalties with their accompanying dollar costs.

In the case of a LSA however such quick fix techniques are not feasible. Furthermore, their turn around time includes the time lag in reaching

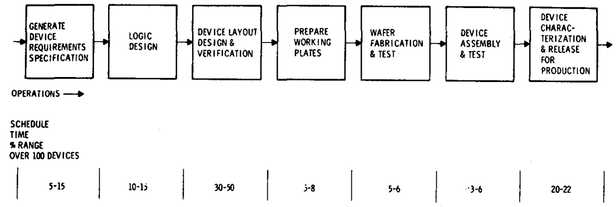

GENERATE DEVICE REQUIREMENTS SPECIFICATION
LOGIC DESIGN
DEVICE LAYOUT DESIGN & VERIFICATION
PREPARE WORKING PLATES
WAFER FABRICATION & TEST
DEVICE ASSEMBLY & TEST
DEVICE CHARAC-TERIZATION & RELEASE FOR PRODUCTION
OPERATIONS
SCHEDULE TIME % RANGE OVER 100 DEVICES
5-15
10-15
30-50
5-8
5-6
3-6
20-22

FIGURE 2—Device design cycle

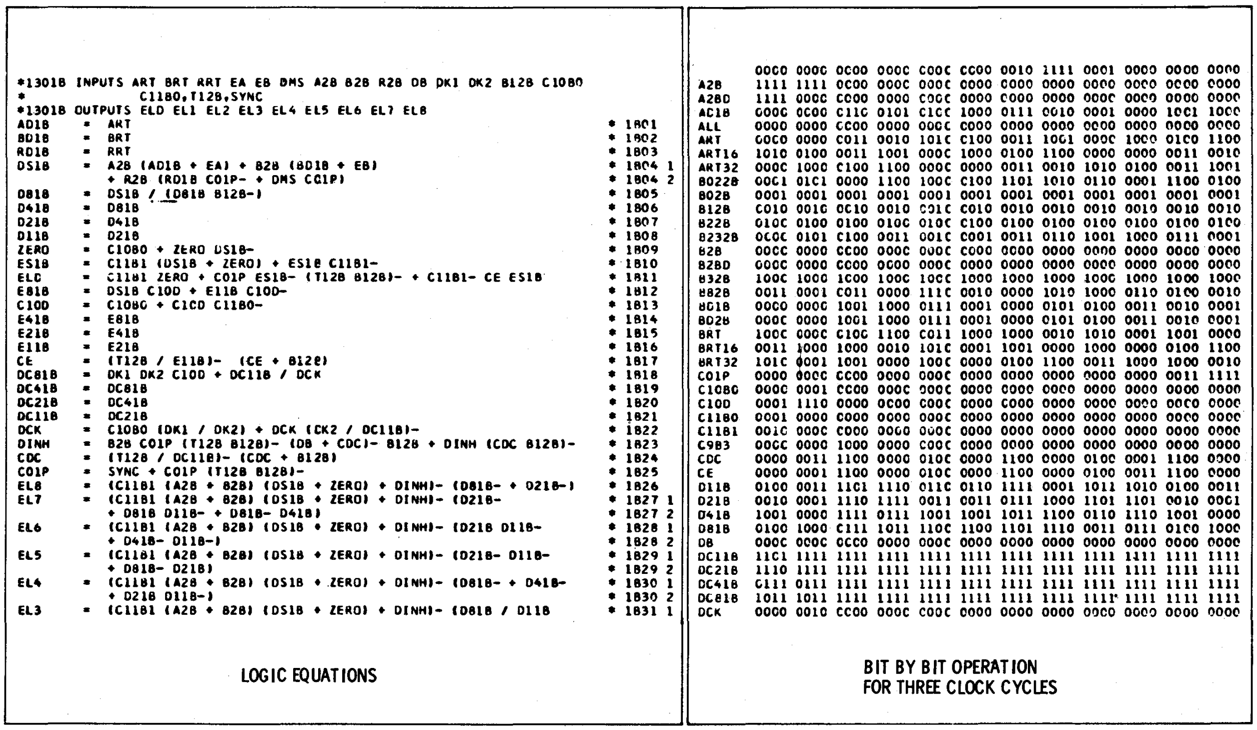
agreement with the device supplier on both the nature of the problem and the fix. It may involve additional problems, such as competing priority projects in that supplier's shop. As a result, some system suppliers have felt it necessary to develop the engineering capability in-house for LSA production. Others have developed a design capability, but propose to use established semiconductor manufacturers for device production. It is still too early to say how the LSA supplier-user interface will stablize, but it is certain that over the next few years a number of problems in this area must be worked out.

Device design cycle

Leaving the supplier-user interface problem for the moment, consider where design aids could be most profitably employed in the device design cycle. The areas of Logic Design and Device Layout show up in Figure 2 as taking half or more of the total cycle time, so must be considered prime areas for design aids. As these aids have a major impact on the skills required of device design engineers, it is worth reviewing them in some detail.

The uses of logic simulation during the early logic design phase are well known, even from discrete component days. It was necessary to write new programs for MOS 4-phase logic. While this technique dramatically reduces circuitry power requirements from dc or 2-phase designs, it does provide a fascinating problem for the logic designer. The lefthand side of Figure 3 shows the logic equations set down in format suitable for entry into the computer. The righthand side shows the situation at the nodes designated at the

FIGURE 3—MOS logic simulation program



```
*1301B INPUTS ART BRT RRT EA EB DMS A2B B2B R2B DB DK1 DK2 B12B C10B0
*               C11B0,T12B,SYNC
*1301B OUTPUTS ELD EL1 EL2 EL3 EL4 EL5 EL6 EL7 EL8
AD1B     =  ART                                                  * 1801
BD1B     =  BRT                                                  * 1802
RD1B     =  RRT                                                  * 1803
DS1B     =  A2B (AD1B + EA) + B2B (BD1B + EB)                    * 1804 1
            + R2B (RD1B C01P- + DMS C01P)                        * 1804 2
D81B     =  DS1B / (D81B B12B-)                                  * 1805
D41B     =  D81B                                                 * 1806
D21B     =  D41B                                                 * 1807
D11B     =  D21B                                                 * 1808
ZERO     =  C10B0 + ZERO DS1B-                                   * 1809
ES1B     =  C11B1 (DS1B + ZERO) + ES1B C11B1-                    * 1810
ELC      =  C11B1 ZERO + C01P ES1B- (T12B B12B)- + C11B1- CE ES1B  * 1811
E81B     =  DS1B C10D + E11B C10D-                               * 1812
C10D     =  C10B0 + C10D C11B0-                                  * 1813
E41B     =  E81B                                                 * 1814
E21B     =  E41B                                                 * 1815
E11B     =  E21B                                                 * 1816
CE       =  (T12B / E11B)-  (CE + B12B)                          * 1817
DC81B    =  DK1 DK2 C10D + DC11B / DCK                           * 1818
DC41B    =  DC81B                                                * 1819
DC21B    =  DC41B                                                * 1820
DC11B    =  DC21B                                                * 1821
DCK      =  C10B0 (DK1 / DK2) + DCK (CK2 / DC11B)-               * 1822
DINH     =  B2B C01P (T12B B12B)- (DB + CDC)- B12B + DINH (CDC B12B)-  * 1823
CDC      =  (T12B / DC11B)- (CDC + B12B)                         * 1824
C01P     =  SYNC + C01P (T12B B12B)-                             * 1825
EL8      =  (C11B1 (A2B + B2B) (DS1B + ZERO) + DINH)- (D81B- + D21B-)  * 1826
EL7      =  (C11B1 (A2B + B2B) (DS1B + ZERO) + DINH)- (D21B-     * 1827 1
            + D81B D11B- + D81B- D41B)                           * 1827 2
EL6      =  (C11B1 (A2B + B2B) (DS1B + ZERO) + DINH)- (D21B D11B-  * 1828 1
            + D41B- D11B-)                                       * 1828 2
EL5      =  (C11B1 (A2B + B2B) (DS1B + ZERO) + DINH)- (D21B- D11B-  * 1829 1
            + D81B- D21B)                                        * 1829 2
EL4      =  (C11B1 (A2B + B2B) (DS1B + ZERO) + DINH)- (D81B- + D41B-  * 1830 1
            + D21B D11B-)                                        * 1830 2
EL3      =  (C11B1 (A2B + B2B) (DS1B + ZERO) + DINH)- (D81B / D11B  * 1831 1
```

LOGIC EQUATIONS

```
        0000 0000 0000 0000 0000 0000 0010 1111 0001 0000 0000 0000
A2B     1111 1111 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
A2BD    1111 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
AC1B    0000 0000 0110 0101 0100 1000 0111 0010 0001 0000 1001 1000
ALL     0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
ART     0000 0000 0011 0010 1010 0100 0011 1001 0000 1000 0100 1100
ART16   1010 0100 0011 1001 0000 1000 0100 1100 0000 0000 0011 0010
ART32   0000 1000 0100 1100 0000 0000 0011 0010 1010 0100 0011 1001
B022B   0001 0101 0000 1100 1000 0100 1101 1010 0110 0001 1100 0100
B02B    0001 0001 0001 0001 0001 0001 0001 0001 0001 0001 0001 0001
B12B    0010 0010 0010 0010 0010 0010 0010 0010 0010 0010 0010 0010
B22B    0100 0100 0100 0100 0100 0100 0100 0100 0100 0100 0100 0100
B232B   0000 0101 0100 0011 0010 0001 0011 0110 1001 1000 0111 0001
B2B     0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
B2BD    0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
B32B    1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
B82B    0011 0001 0011 0000 1110 0010 0000 1010 1000 0110 0100 0010
BD1B    0000 0000 1001 1000 0111 0001 0000 0101 0100 0011 0010 0001
BD2B    0000 0000 1001 1000 0111 0001 0000 0101 0100 0011 0010 0001
BRT     1000 0000 0100 1100 0011 1000 1000 0010 1010 0001 1001 0000
BRT16   0011 1000 1000 0010 1010 0001 1001 0000 1000 0000 0100 1100
BRT32   1010 0001 1001 0000 1000 0000 0100 1100 0011 1000 1000 0010
C01P    0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0011 1111
C10B0   0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
C10D    0001 1110 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
C11B0   0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
C11B1   0010 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
C9B3    0000 0000 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000
CDC     0000 0011 1100 0000 0100 0000 1100 0000 0100 0001 1100 0000
CE      0000 0001 1100 0000 0100 0000 1100 0000 0100 0011 1100 0000
D11B    0100 0011 1101 1110 0110 0110 1111 0001 1011 1010 0100 0011
D21B    0010 0001 1110 1111 0011 0011 0111 1000 1101 1101 0010 0001
D41B    1001 0000 1111 0111 1001 1001 1011 1100 0110 1110 1001 0000
D81B    0100 1000 0111 1011 1100 1100 1101 1110 0011 0111 0100 1000
DB      0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
DC11B   1101 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111
DC21B   1110 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111
DC41B   0111 0111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111
DC81B   1011 1011 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111
DCK     0000 0010 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
```

BIT BY BIT OPERATION
FOR THREE CLOCK CYCLES

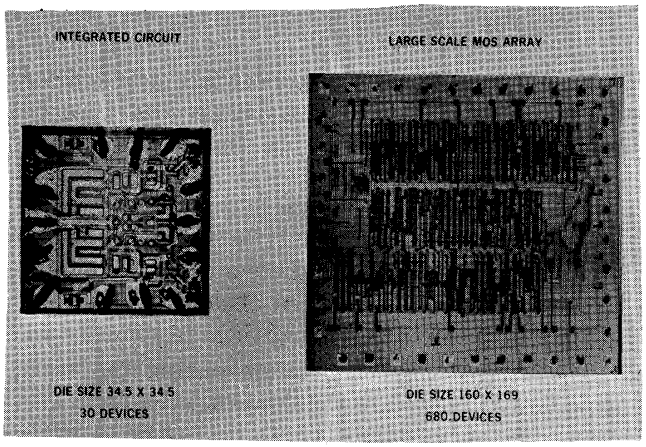


FIGURE 4—Complexity comparison

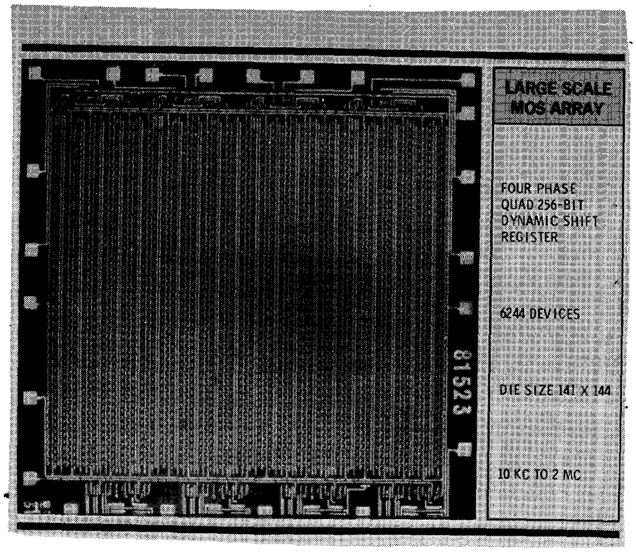


FIGURE 5—Large scale MOS array

lefthand column during three clock-cycles of operation.

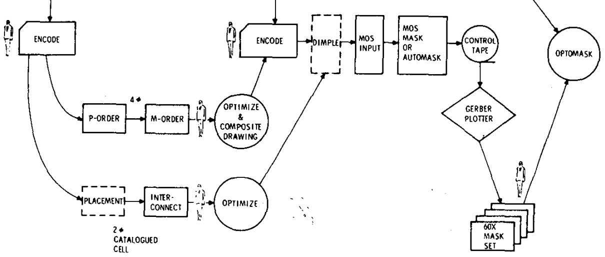
Other than this modification for multiphase clock operation, our logic programs are very similar to ones which have been in widespread use by computer designers for the past decade. Therefore, I propose to leave this area for that of device layout and mask preparation. Here development has been most rapid over the past few years.

Device layout

As IC technology is well developed, why should the change from IC's to LSA's pose such a problem in device layout? The answer is shown strikingly in Figure 4. The increase in device complexity from IC to LSA is so great that problems become different in kind rather than simply different in magnitude. Generating and checking a mask set for the production of 30 element dual quad-NAND gate IC of Figure 4-a is inadequate preparation for checking mask sets for 600 to 6000 element LSA's. The complete +— arithemetic unit in Figure 4-b is a challenge to a designer. Perhaps even more daunting is the example of the latter shown in Figure 5, a 1000 bit shift register employing over 6200 active elements.

An engineer using his detailed knowledge of the dual nand gate bipolar circuit of Figure 4-a could with care verify the completeness of each mask and registration from mask to mask. When he must check for the possible omission of one of several thousand p region windows or of a similar number of interconnections, he faces an impossible task. With care, he can catch all but 1% or so of the errors, but when the points to be checked number in the thousands the resultant possible number of defects is completely unacceptable. This increase in complexity then cannot be coped with by simply being more careful. The engineer is forced to become more heavily dependent on computer programs and other hardware aids for both checking and indeed for the original device design.

The process steps required to do the layout and produce a mask set are shown in Figure 6. The top row shows the manual operations required to produce a device mask set, given the logic equations to be mechanized. In our experience approximately 1100 manhours were required to complete the operations shown for small to medium sized arrays using custom design. For one reason or another attempts have been made from time to time to use this technique to produce masks for truly large arrays in the range of 800 to 1000 elements. These attempts could not be called successful.



EIGURE 6—MOS-FFT LSA computer-aided design system

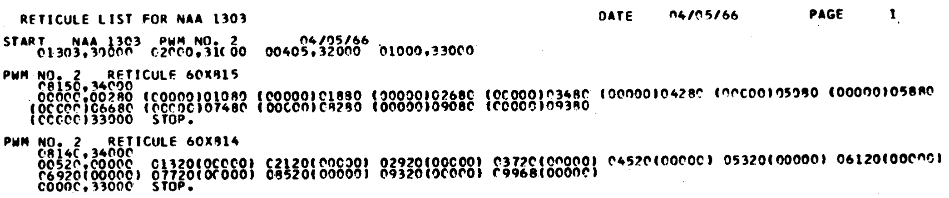
To increase the probability of successful device designs we wrote a series of programs for the engineer to use as aids in doing his original device layout. For 4-phase custom circuitry, we decided that a completely automatic layout program would not be a cost-effective first step.

For the present, dropping to the second line of the flow diagram of Figure 6, we have two programs which act as engineering design aids—the P-ORDER and the M-ORDER programs. The logic equations to be incorporated on the chip are written out in equation form, encoded and used as inputs for these programs. The output of the P-ORDER program is a list of logic equations ordered so as to minimize the total length of interconnecting metal and the number of crossovers. The p regions, each of which represents an equation to be mechanized, appear in sequence so common terms are grouped together. As these terms are connected by metal lines, this grouping minimizes their length.

The resultant ordered equation set is then fed to the M-ORDER program. This program interleaves the interconnecting metal to minimize total chip area. The P-ORDER program has generally provided a fairly long loosely packed array. The M-ORDER program attempts to square up this array and fill the empty spaces reducing the total p length as well as chip area.

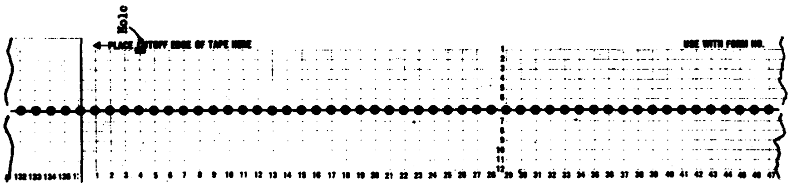
A number of iterative runs are made with these programs. The engineer evaluates the results after the initial 50 or so iterations and intervenes manually to rearrange the sequence of logic terms or their position on the chip whenever it is apparent to him that such a rearrangement will be an improvement. Presently, this requires that a computer printout be delivered to the engineer. The future incorporation of a graphics terminal will let us use the programs in a truly interactive way and so reduce the design process flow time.

The ordered equations are now used in producing a hand drawn composite, like the section shown in Figure 7. This composite is used for encoding the input to the computer programs generating the control tape for the Gerber plotter. The total direct labor hours required for these operations on the average amount to 500 hours. This is a reduction by a factor of 2 from the time required to carry out an all-manual operation. For LSA's of the size we are currently producing these programs make all the difference between a possible and impossible design task.

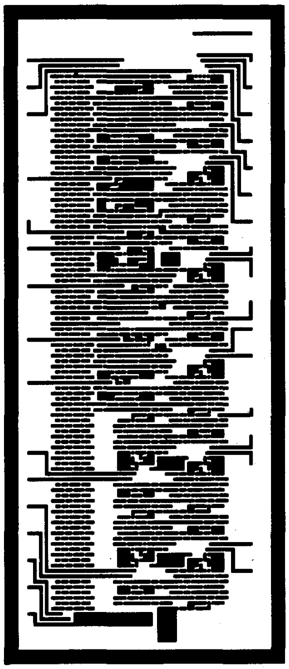


```
  RETICULE LIST FOR NAA 1303                                                  DATE    04/05/66         PAGE     1

START   NAA 1303  PWM NO. 2         04/05/66
    01303,30000  C2000,31C00  00405,32000  01000,33000

PWM NO. 2   RETICULE 60X815
    C8150,34000
    0C00C,00280 (C0000)01080 (C0000)C1880 (00000)02680 (0C000)03480 (00000)04280 (00C00)05080 (00000)05880
   (0CC00)06680 (0CC0C)07480 (00CC0)C8280 (00000)0908C (C0000)09380
   (CCC0C)33000  STOP.

PWM NO. 2   RETICULE 60X814
    C814C,3400C
    00520,C000C  C1320(0CCC0) C2120(00C00) 02920(00C00) C372C(00000) C4520(0000C) 05320(00000) 06120(00000)
    C6920(00000) 07720(0C000) 08520(00000) 09320(0C0C0) C9968(0000C)
    C000C,3300C  STOP.
```

COMPUTER GENERATED RETICULE LIST



COMPUTER GENERATED
PLOTTER CONTROL TAPE



60X MOS"P" MASK
GENERATED BY THE PLOTTER

FIGURE 7—MOS mask generation

The third flow line in Figure 6 shows a different set of design aid programs used for placement and interconnection of standard catalog cells for 2-phase logic circuitry. Design direct labor hours are saved by using these standard catalog functions, as much of the detail chip layout has already been done and is available in the computer data bank. Only the placement of these cells and their interconnection remains to be carried out. There is considerable direct labor time saving using this method. On the average we find it to be about half that required for a 4-phase custom layout or approximately 250 hours. In most cases, however, there is a 10–20% penalty paid in chip size for this use. For this reason, the standard cell approach is often used during development in order to reduce schedule time. A custom design layout is then carried out if a sizeable production order follows.

To reduce human errors in physical mask production, a tape controlled automatic plotter is used. To produce the tape, it was necessary to encode, from a hand-drawn composite of the device, each incremental step of the plotter defining the p and m regions. This took up to 300 hours of technician time.

From this listing the input card deck for the MOS input program was prepared. The output of this program was used as input to the MOS-MASK program which prepared the plotter control tape. This tape, a section of which is shown in Figure 7, specifies the reticle to be used at each XY coordinate of the masks. The mask sets are first plotted at 120 × or 60 × final size. Proof prints are then sent to the design engineer for visual check before making the working plates. If the check is satisfactory, then standard procedures for mask set production are followed.

Three hundred hours of technician time was felt to be excessive for encoding the input to the MOS-MASK program. We then took the next step of writing a simplified intermediate program known as DIMPLE. DIMPLE takes end point encoding and translates it into incremental encoding in a form suitable for use as input to the MOS-MASK program. This relatively simple addition reduced encoding time from 300 hours to 100 hours.

One additional feature to be added is a digitizer to replace the manual listing of end point encoding. This will further reduce errors and save schedule time. It is not so much the two to three days of original encoding time we wish to save, but rather the 2-3 weeks involved in making corrections when an error has been discovered only after a device has been all through the production process.

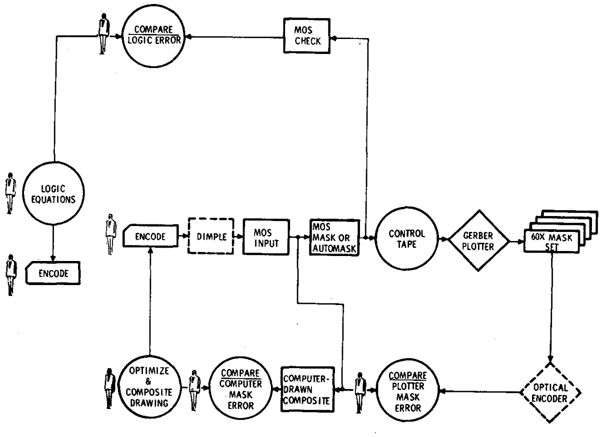


FIGURE 8—MOS-LSA mask checks

Design checks

Even with this extensive use of design aids, it is still possible to have the occasional error creep into the design. As a result we have developed two additional checks, shown in Figure 8, to run thru before the masks enter the optomask system. One is a straightforward check on the logic mechanization. The output data from the MOS INPUT program is fed to the MOS CHECK program. There it is combined with a card deck giving the identification of each gate in terms of the logic equations mechanized. With these inputs the MOS CHECK program regenerates the logic equations mechanized on the chip and types them out. Presently, this printout is checked manually against the original set of logic equations to determine if errors have been made. The program is being modified to do this logic verification within the computer and print out only the errors.

In addition we use a computer printout, shown on the right-hand side of Figure 9, to determine that the plotter control tape is accurate. Computer printout is compared against the hand-drawn composite to be sure that no errors have been made in this translation.

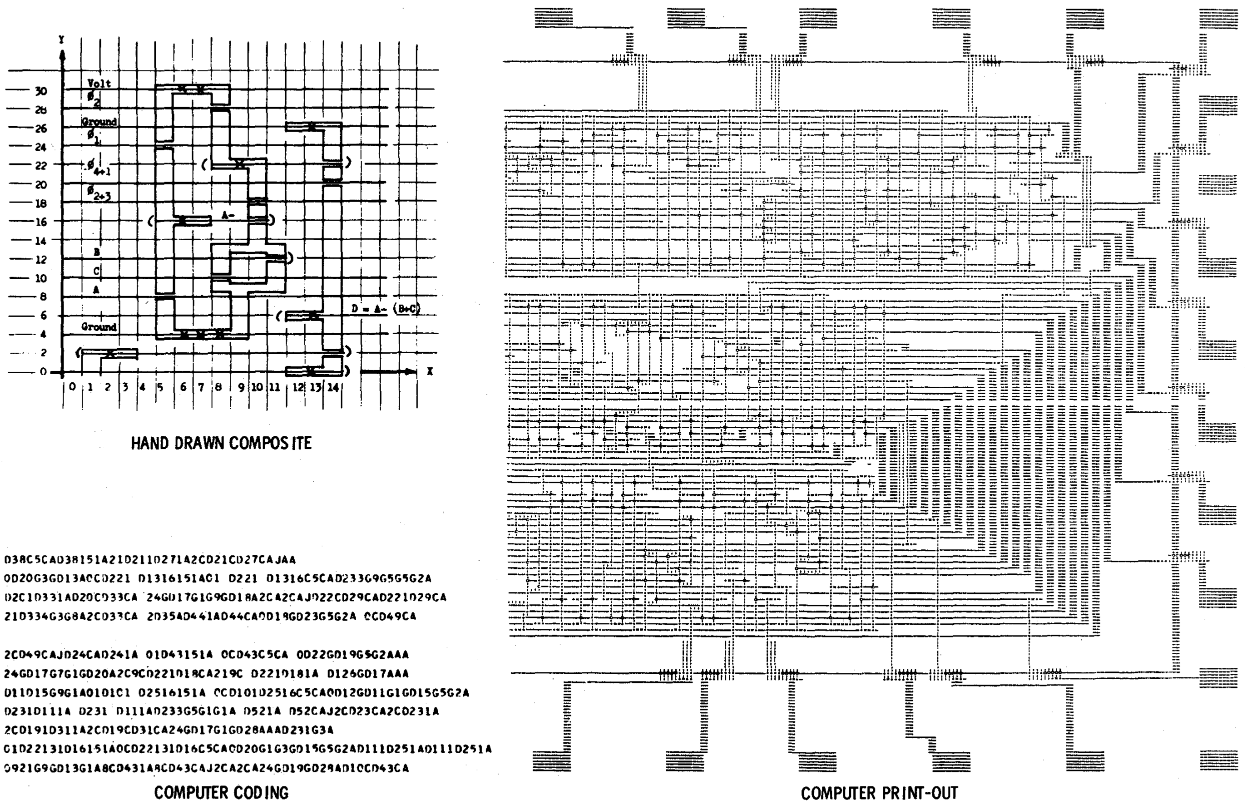
Even if the plotter control tapes are perfect, there is still a possibility that the plotter will malfunction in a way not caught by our monitoring. We have under development an optical encoding equipment which determines the location of all corners on the mask set produced by the plotter. The coordinates of these corners will then be compared with the end point encoding used as entry to the MOS MASK programs as a final check to ensure that the 120 × or 60 × final size original is a faithful translation of the original input to the plotting system. With this implementation we will have the highest confidence of being able to produce working devices functionally correct the first time thru the system.

We still have not exhausted the ways in which design aids can assist in turning out a satisfactory product. Before committing to physical production, the designer needs to know that his device is not only logically correct but that it will operate without errors due to excessive gate loading or noise cross coupling for example. We had previously developed a series of programs for circuit analysis of thin film hybrid circuits and circuits using a combination of integrated circuits and discrete components. The programs exercised the circuits, simulating a variety of stress environments such as temperature extremes, high radiation levels and varying supply voltages.

Capitalizing on this experience we have written a series of programs for the analysis of MOS LSA's. MOS SNAP, for example, is used in doing an overall speed noise analysis. The capacity loading and noise coupling at any given group of nodes in the total nodes set can be determined using this program and supplied as outputs to the engineer for verification of performance.

A separate program called TRAC is used for local speed noise analysis at a restricted group of nodes in the device. Internally, the program forms an indefinite admittance matrix for each node and calculates current and voltage for the equivalent circuit, using an accurate non-linear model for each active MOS element. A representative section of circuitry for analysis, the computer input format and the graphics output to show transient response at a given node, are shown in Figure 10.

The calculations, while straightforward, are far too tedious, error prone and time consuming for an engineer to do by hand with any expectation



HAND DRAWN COMPOSITE

```
D38C5CAD38151A21D211D271A2CD21CD27CAJAA
0D20G3GD13ACCD221 D1316151AC1 D221 D1316C5CAD233G9G5G5G2A
D2C1D331AD20CD33CA 24GD17G1G9GD18A2CA2CAJD22CD29CAD221D29CA
21D334G3G8A2CD33CA 2D35AD441AD44CA0D18GD23G5G2A 0CD49CA

2CD49CAJD24CAD241A 01D43151A 0CD43C5CA 0D22GD19G5G2AAA
24GD17G7G1GD20A2C9CD221D18CA219C D221D181A D126GD17AAA
D11D15G9G1A0101C1 D2516151A 0CD101D2516C5CA0D12GD11G1GD15G5G2A
D231D111A D231 D111AD233G5G1G1A D521A D52CAJ2CD23CA2CD231A
2CD191D311A2CD19CD31CA24GD17G1GD28AAAD231G3A
G1D22131D16151A0CD22131D16C5CA0D20G1G3GD15G5G2AD111D251AD111D251A
0921G9GD13G1A8CD431A8CD43CAJ2CA2CA24GD19GD28AD10CD43CA
```

COMPUTER CODING

COMPUTER PRINT-OUT

FIGURE 9—MOS circuit coding

of useful results. With the TRAC computer program using 200,000 bytes of core memory, with a 2000 FORTRAN statement program, however, significant results can be obtained with only 3 minutes of computer running time for 20 circuit nodes.

Design aids summary

The programs described contribute substantially to device design cost reduction by reducing both the direct labor hours required and the recycles needed due to human errors in the design process. Developing these programs has been expensive and time consuming as you can deduce from the program summary descriptions in Table I, DESIGN PROGRAMS FOR CUSTOM MOS-FET 4-PHASE DEVICE. We feel that it has been worthwhile and indeed essential for providing us with an LSA design and development capability.

This paper has been a progress report on a continuing program of design aids development rather than a final description. In addition to the new programs mentioned in this paper as being currently under development, others will be starting in the near future to reduce obvious inefficiencies in our current operation. We continue to attack each area of the design and development process which has either high direct labor content or is schedule limiting. We pay particular attention to those areas in which our experience shows errors are apt to occur which cause devices to fail on their first run thru the design cycle. Our schedule times to produce and modify LSA's are still too long; our initial device design costs are still too high. From our progress to date, I feel that within the next 1-2 years we will have achieved an optimum system, balancing engineering hours on the one hand with the cost of computer running time and precision design and development aid equipment on the other to minimize both costs and schedule time.

PROGRAM NAME	CORE USED (BYTES)	NO. OVERLAYS	RUNNING (360-65) TIME (MINUTES)	NO. STATEMENTS (PL/1)
'P' ORDER	120K		10 - 15	1250
'M' ORDER	225K	4	20 - 30	2000
MOS INPUT	220K		1/2 - 1	1750
DIMPLE	120K		1 - 2	2500
MOS CHECK	350K	5	10 - 12	2750
MOS SNAP	150K		2 - 4	600
MOS MASK	250K	3	2 - 5	2200
MOS AUTOMASK	300K	4	2 - 5	2750

TABLE I—Computer aided design programs for custom MOS-FET 4-phase devices

Trends

I predict that our current trend in engineering organization will continue. Within the next two years I would expect to see most engineering operations making use of LSA's set up as shown in Figure 11. Logic design and system engineering will have coalesced into a single group. Circuit engineering will have been fully absorbed by device engineering. Design aids, both program and equipment, will have equal rank with these engineering functions. Such an organization will minimize time required from system concept to production acceptance.

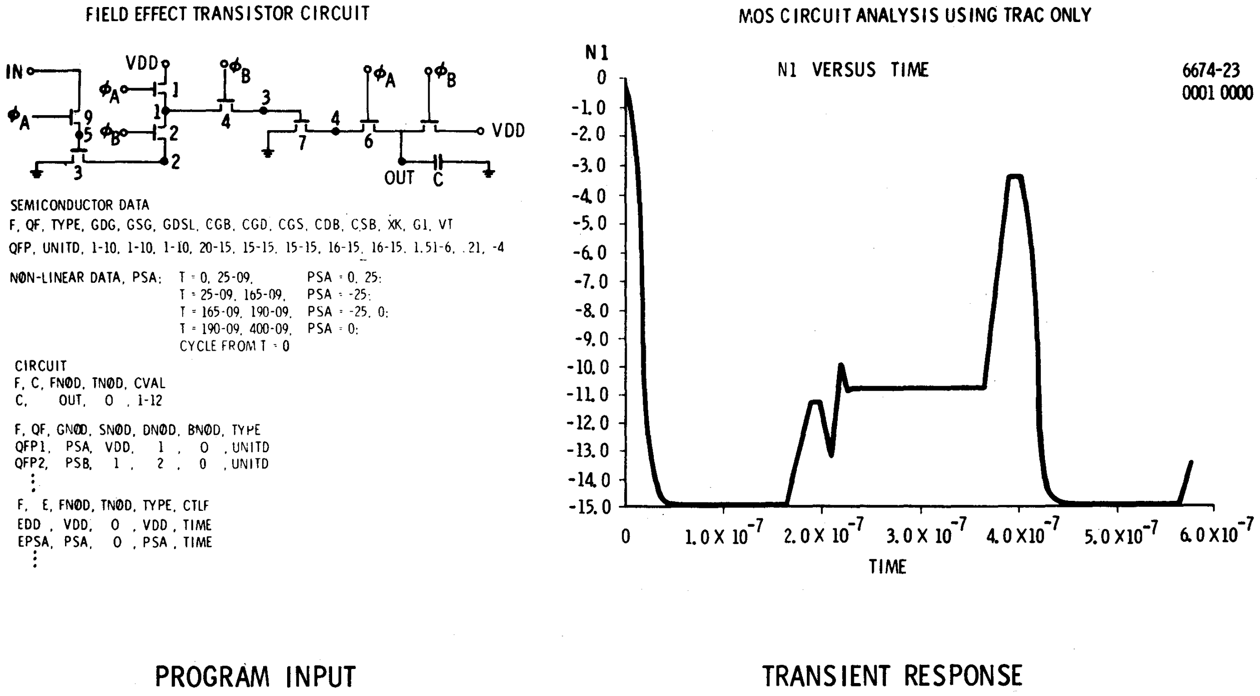
As for the engineering tools used by the organization, I expect a continuing proliferation of software but I also expect that these programs will remain as design aids rather than becoming a completely automated design program. The introduction of graphic terminals with time share computers will accelerate and reinforce this trend. As far as the hardware is concerned, in general it will become more rapid, with some electromechanical devices being replaced by electronic systems. In addition, the accuracy of this precision equipment will increase to allow either greater compaction of devices, therefore reducing costs by having more chips per wafer, or allowing design rules to be relaxed so greater yields can be obtained on chips of today's size.

As I have already pointed out, I expect the device design costs to be reduced sharply over the next two years as an optimum balance is reached by the engineering and by the software and hardware design and development aids.

Spurred in part by the desire to be able to produce larger and larger chips with all elements working, the industry is increasing its knowledge and control of processes. This plus additional experience and the introduction of automation where desirable will lead to a rapid decrease in production costs over this two-year interval as well. Combination of design and production cost reductions will make LSA's a very formidable competitor indeed in the electronic device field by 1970.

The supplier-user interface for LSA's is difficult to predict at this time. My current feeling is that this interface will be quite flexible and econ-

FIGURE 10—Device functional analysis



```
SEMICONDUCTOR DATA
F, QF, TYPE, GDG, GSG, GDSL, CGB, CGD, CGS, CDB, CSB, XK, G1, VT
QFP, UNITD, 1-10, 1-10, 1-10, 20-15, 15-15, 15-15, 16-15, 16-15, 1.51-6, .21, -4

NØN-LINEAR DATA, PSA:  T = 0, 25-09,          PSA = 0, 25:
                       T = 25-09, 165-09,     PSA = -25:
                       T = 165-09, 190-09,    PSA = -25, 0:
                       T = 190-09, 400-09,    PSA = 0:
                       CYCLE FROM T = 0

CIRCUIT
F, C, FNØD, TNØD, CVAL
C,     OUT,   O  , 1-12

F, QF, GNØD, SNØD, DNØD, BNØD, TYPE
QFP1,  PSA,  VDD,   1  ,   O  , UNITD
QFP2,  PSB,   1  ,   2  ,   0  , UNITD
  :
F,  E, FNØD, TNØD, TYPE, CTLF
EDD , VDD,  O  , VDD , TIME
EPSA, PSA,  O  , PSA , TIME
  :
```

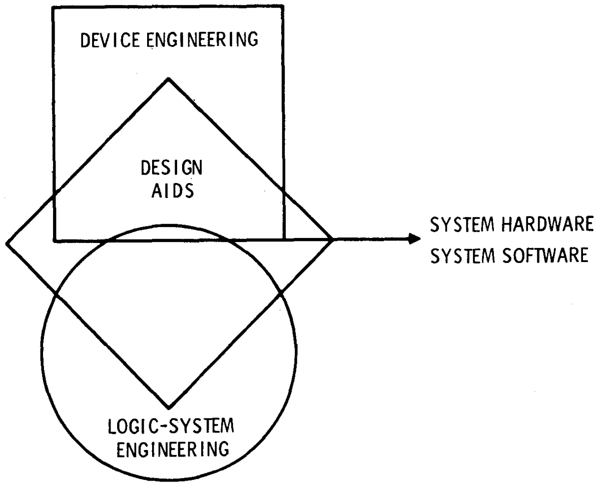


FIGURE 11—Engineering organizational trend, digital systems

omy-determined provided processes are reasonably standardized and both suppliers and users develop or acquire the necessary software and hardware design aids. This would allow users to do their initial design, possibly up to mask production, in-house and then obtain competitive bids for device production from the normal suppliers. On the other hand, if the suppliers also have a well developed device design capability then they could simply supply the LSA specifications and logic equations to the supplier and have him do the complete development job as well as production. It is still too early to say if this desirable state of affairs will in fact come about.

While our current system of designing and producing LSA's is not perfect, we have made considerable progress. Mr. Booher, a rare combination of logic designer and system engineer (though if my predictions are correct, we will see more of this in the future) will describe a General Purpose Parallel Computer developed using his MOS-FET 4-phase logic.

BIBLIOGRAPHY

1 R A RUTMAN
An algorithm for placement of interconnected elements based on minimum wire length
Proceedings of the Spring Joint Computer Conference 1964 pp 477-491

1 H S SCHEFFLER et al
Reliability analysis of electronic circuits
AD 461303 Defense Documentation Center March 1965

3 W HOCHWALD C T KLEINER
Digital simulation of nonlinear electro-magnetic circuits
IEEE Trans on Magnetics Vol MAG-2 No 3 September 1966 pp 532-529

4 *Special report on LSI*
Electronics Vol 40 No 4 February 20 1967 pp 123-182

5 C T KLEINER E D JOHNSON W D ASHCRAFT
A general purpose system of digital computer codes for linear/nonlinear network and system analysis
IEEE International Convention Digest March 1967 pp 424-425

6 R S MILES et al
Design and analysis of electronic circuits
Proceedings of the SHARE and ACM Design Automation Workshop June 1967

7 *Special issue on CAD*
Proceedings of the IEEE Vol 55 No 11 Nov 1967

8 J RHEA et al
Special report on LSI
Areospace Technology January 29 1968 pp 24-46

9 L R McMURRAY et al
Transient radiation analysis by computer program TRAC
Harry Diamond Laboratories June 1968