NASA CR-122372

Final Progress Report
Phase I(a)

MONOLITHIC PARALLEL PROCESSOR

28 January 1970 To 27 September 1971

Contract No. NAS 5-11577

Prepared by

RCA Solid State Division
Somerville, New Jersey

for

Goddard Space Flight Center
Greenbelt, Maryland
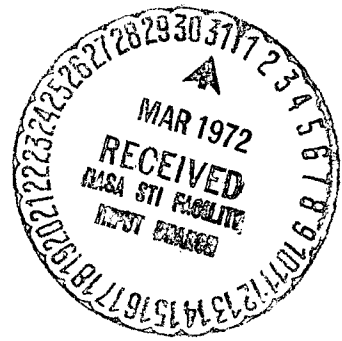
Final Progress Report
Phase I(a)

MONOLITHIC PARALLEL PROCESSOR
28 January 1970 To 27 September 1971

Contract No. NAS 5-11577

Goddard Space Flight Center
Contracting Officer: A.L. Essex
Technical Monitor: F. Link

Prepared by

RCA Solid State Division
Somerville, New Jersey

Project Manager: H. Miiller
Project Engineer: A. Dingwall

for

Goddard Space Flight Center
Greenbelt, Maryland

ABSTRACT

A four-bit parallel processor LSI array was designed and fabricated using COS/MOS integrated-circuit technology. Twenty-five units were delivered to NASA to demonstrate full achievement of Phase I(a) goals and to show the applicability of techniques for high-yield processing. The design features include the provision for interconnecting groups of parallel-processor chips to form an expanded processor of any desired word length. This 800-transistor "computer on a chip" circuit has the logic capability of a medium-size, medium-speed, general-purpose computer suitable for sophisticated scientific data processing.

The ability to fabricate this device repetitively has now been demonstrated.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

LIST OF ILLUSTRATIONS

SECTION I

THE PARALLEL PROCESSOR

A 36-month developmental program has been conducted to design, develop, and fabricate monolithic complementary-symmetry MOS (COS/MOS) large-scale parallel processor arrays[1]. This task was completed successfully and 25 parallel-processor arrays of 800-transistor complexity were designed, fabricated, tested, and delivered to the NASA Goddard Space Flight Center.

The major objective of this Phase I(a) project was to fabricate high-quality photomasks for this "large" chip-size device using computer automated techniques in order to permit reproducible fabrication of this device with extremely low standby leakage levels. This task was successfully accomplished. High performance units of this arithmetic unit can now be repetitively generated to meet NASA's needs.

The original concept of functional operation of the array was conceived by R.J. Lesniewski,[2-4] at that time with the NASA Goddard Space Flight Center, as part of a program to realize an ultralow-power computer that requires a minimal number of array types. The logic was extended and techniques for interconnecting groups of chips were implemented by RCA Airborne Systems Division, Burlington, Massachusetts, and by RCA Solid State Division, Somerville, New Jersey.

Among the unique design features of the parallel processor is its modularity feature. By using external mode controls, a powerful multifunction array capability is achieved which allows one chip to be used in a variety of processing applications, and allows the generation of complete arithmetic units using a single-array type.

The parallel processor has four-bit arithmetic processing and storage capabilities with full functional decoding, expansion capability, and time

1

sharing of data pins. The array has 27 input/output pins with an equivalent logic complexity of 200 two-input gates implemented on a 146- by 155-mil chip containing 775 active devices.

The RCA TA5716, a four-bit, COS/MOS parallel processor, is an example of one building-block of a powerful computer arithmetic unit suitable for use in low-power, medium-speed applications. Because of its unique mode-controlled logic, n-bit arithmetic units of 8-, 12-, 16-, and 32-bit lengths can be constructed by interconnecting several TA5716 processors. Significant design features of the TA5716 include:

- High reliability COS/MOS circuitry

- Look-ahead-carry for higher speed operation of n-bit arrays

- 16-instruction repertoire

- Single-phase clocking

- Easily expandable to n-bit operation

- Bidirectional data buses to minimize interconnections

- Full instruction decoding on chip

- Fully static operation

- Medium-speed operation: Add time for four-bit = 1.3 microseconds (typical), 16-bit = 2 microseconds (typical)

- Low standby power (<10 microwatts typical)

- Low dynamic power (10 milliwatts typical)

- Full military temperature range (-55°C to +125°C)

- High noise immunity: 45 percent of V_{DD} typical over full temperature range

- Operation from a single power supply of 3 to 15 volts

- Single-phase clocking

- High input impedance: 10^{11} ohms typical at 25°C

2

SECTION II

PARALLEL PROCESSOR LOGIC DESIGN

A. GENERAL DESCRIPTION OF FOUR-STAGE PROCESSOR

Figure 1 is a photograph of the parallel processor chip. The four-stage parallel processor basically is a four-stage shift register that has both serial and parallel access. The logic associated with the register allows parallel-two's complement addition, AND, OR, and EXCLUSIVE OR logic operations; and right, left, or right-cyclic shifts.

Figure 2 shows the lead requirements for the four-stage processor. All control lines are encoded, with five leads used for instructions; four leads for control; one lead for timing; two leads for power; and five leads for the following conditions:

    a. negative indication

    b. zero indication

    c. overflow indication

    d. overflow input/output

    e. conditional input

Because information will enter and leave on the same line, six leads are required for the four-stage register to transfer data, with four of the leads used for parallel access and the remaining two leads used for serial access. In addition, four leads are available for expansion to a multiple of four (16 for example) stage processor.

B. GENERAL LOGIC DESCRIPTION

The logic configuration for the four-stage parallel processor is shown in Figure 3. Functional gating is used where possible to achieve a low device count. The hardware requirement is about 750 devices and 27 bonding pads.
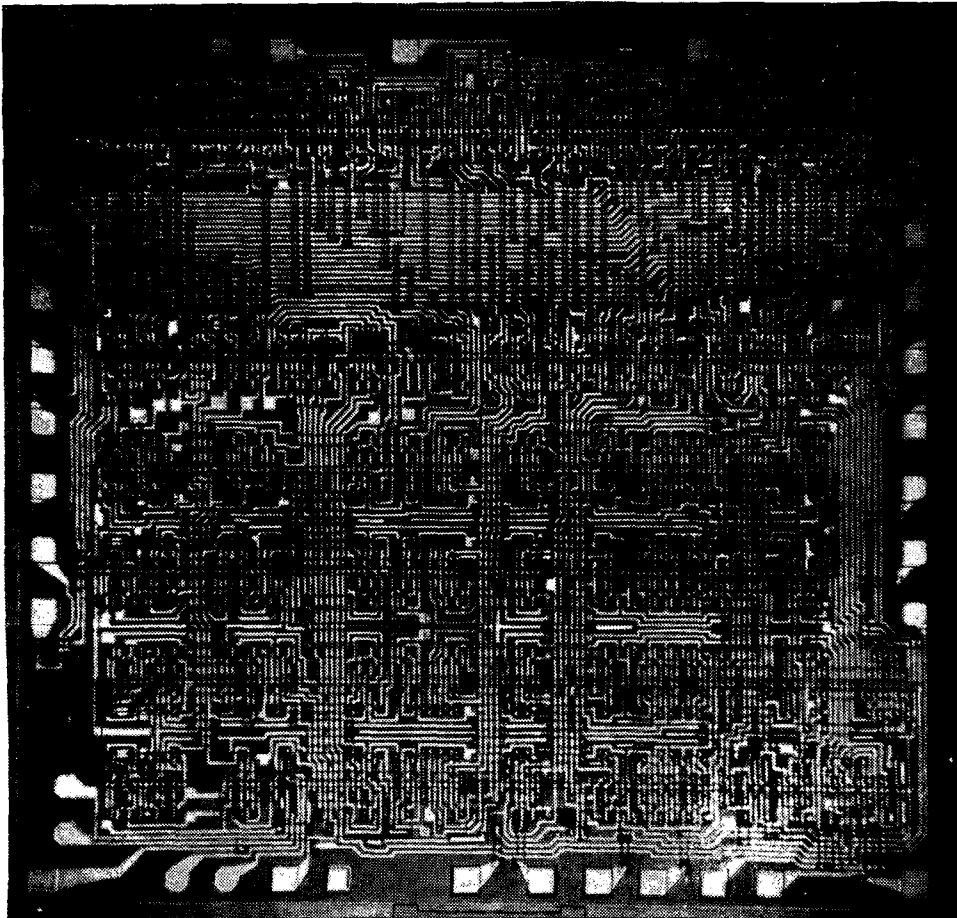
03898 V

Figure 1. Parallel Processor, Array Chip

4

DATA 1	1		R$_{02}$ 28
DATA 4	2	TA5716	DATA 3 27
DATA 2	3	(TOP VIEW)	V$_{DD}$ 26
NEG. IND.	4		GND 25
Z$_i$	5		ZERO IND. 24
c	6		e 23
d	7		a 22
A	8		b 21
C	9		CLOCK 20
RT. DATA	10		B 19
BYPASS	11		LT. DATA 18
Nc	12		OVERFLOW IND. 17
C$_1$	13		OVERFLOW I/O 16
R$_{01}$	14		C$_2$ 15
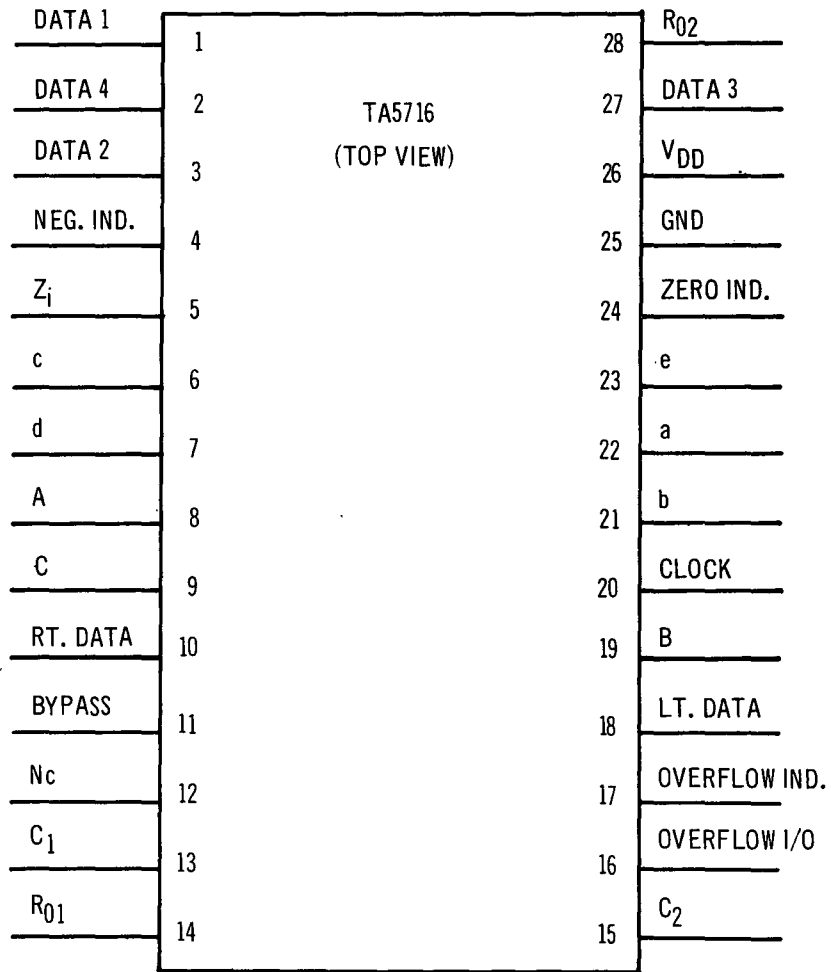
05697L

Figure 2. Terminal Assignment Diagram
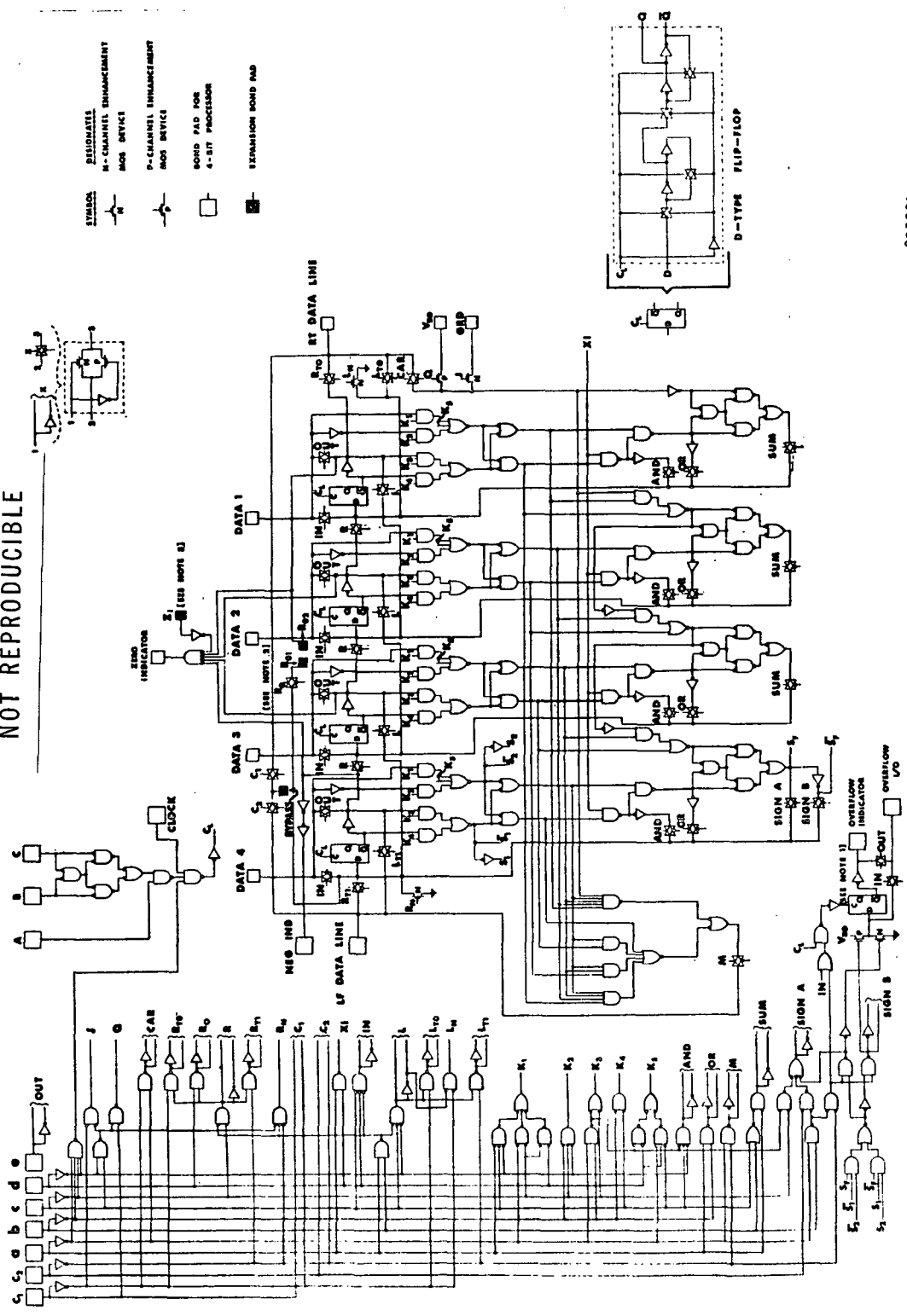
5

NOT REPRODUCIBLE

Figure 3. Logic Diagram

6

C. OPERATIONAL MODES

For this discussion mode is defined as the ability of the parallel pro-
cessor to control the transfer of either serial data or carries due to arith-
metic operations. The parallel processor will be capable of operation in one
of four modes. For simplicity, consider the parallel processor as a strictly
serial device. Serial data can enter or leave either side of the register.
Since there is only one lead on either side of the register, the serial trans-
fer must be bidirectional. The manner in which modes control the serial-data
lines is as follows:

a. Mode 0 (A, Figure 4 - Data can enter or leave from either side.

b. Mode 1 (B, Figure 4 - Data can enter or leave the register on
    the left side during any serial operation.

c. Mode 2 (C, Figure 4 - Data can enter or leave on the right side.

d. Mode 3 (D, Figure 4 - Serial data neither may enter nor leave
    the register, regardless of the nature of the serial operation;
    furthermore, the register is bypassed electrically, i.e., there
    is an electrical bidirectional path from the right serial lead
    to the left serial lead. The most-significant (leftmost) bit
    is used as the sign bit.

D. OVERFLOW DETECTION

A two's complement overflow is defined as having occurred if the signs of
the two initial words are the same and the sign of the result is different
while performing the ADD instruction.

The parallel processor will be capable of detecting and indicating the
presence or absence of an arithmetic two's complement overflow. Overflows
will be detected and indicated only during operation in Mode 2 or Mode 3. In
either mode, only four instructions (AD, SMZ, SM, and SUB) will have the
potential of causing a two's complement overflow. If an overflow is detected
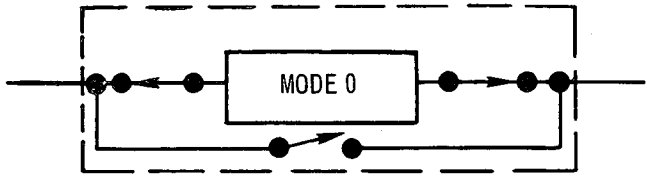
A. MODE 0
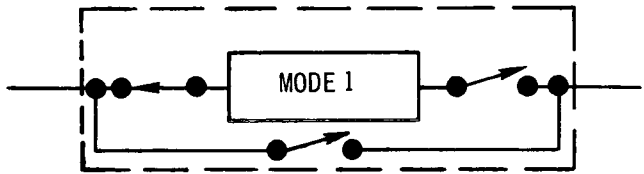
B. MODE 1
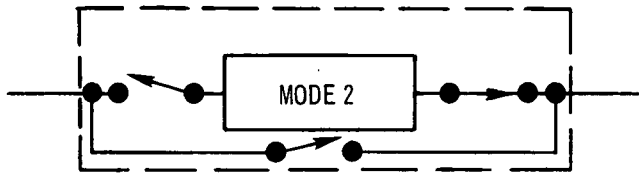
C. MODE 2
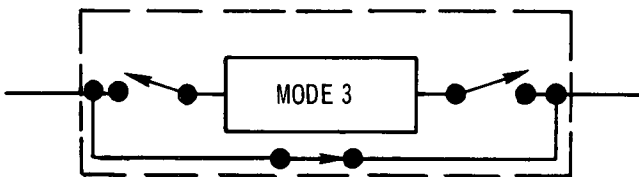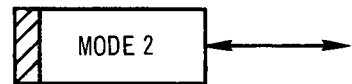
D. MODE 3
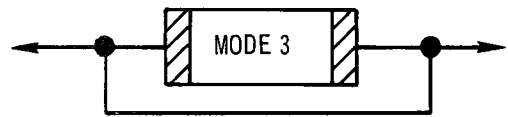
01993L

Figure 4. Operational Modes

and stored by a flip flop, only one of the five instructions (AD, SMZ, SM, SUB or IN) can change the overflow indicator.

Occurrence of a two's complement overflow is represented by a "1" in the overflow flip flop. The absence of an overflow is represented by a "0" in the overflow flip flop. The flip flop will change from zero to one as overflows do not or do occur.

When any one of the three subtraction instructions is used, the sign bit of the data being subtracted will be complemented, and this value will be used in the same manner as one of the initial signs (as in the add instruction) to detect overflows.

If an overflow occurs, the final sign will be one's complemented. This means that the final sign returns to the same polarity as the original sign.

The overflow flip flop will be updated at the same time that the new result is stored in the parallel processor.

E. ZERO DETECTION

The parallel processor will be capable of detecting the condition of all zeros. This operation will be independent of modes. A condition of all zeros will be represented by a "1" on the zero indicator line; otherwise, this line will be zero. If the particular four-bit processor represents the least significant set of bits, ZI should be tied to +V. ZI on all other parallel processor array should be attached to the previous zero indicator line.

F. NEGATIVE DETECTION

The parallel processor will be capable of detecting the presence of a negative number. This operation is independent of modes. If the condition is true, a "1" will appear on the negative indicator line; otherwise, a "0" will appear. A "1" in the most-significant bit position will indicate a negative representation.

9

G. CONDITIONAL OPERATION

Once the instruction and mode have been applied, only the clock pulse will be required to change the state of the register. If this pulse could be inhibited in the ON condition, all instructions would behave as a NO-OP.

The clock pulse can be constrained by using "conditions." A conditional input (C) is compared with a control line (B), and a second control line (A) defines whether or not to test the conditional input line (C). An instruction will be permitted to operate under the following conditions:

    a. Unconditional

    b. The conditional input is positive

    c. The conditional input is negative

H. INSTRUCTION REPERTOIRE

Four encoded lines will be used to represent 16 instructions. A fifth line will be used solely to represent an OUT command. Encoded instructions will be as follows:

NO-OP
Left shift
Right shift
Rotate (cycle) right
Input
Subtract from memory (SM)
Count up
Count down
Clear to zero
Set to one
AND
OR
EXCLUSIVE OR
Subtract from zero (SMZ)
Add (AD)
Subtract (SUB)

I. SERIAL-SHIFT OPERATIONS

    a. Rotate (cycle) right - This operation is internal. The contents of
the register will shift to the right, cyclic fashion, with the left-
most stage accepting data from the rightmost stage, regardless of
mode. Data may leave the register serially on the right data line
only while the register is in Mode 2 or Mode 0. Data may leave the
left data line serially while in Mode 1 or Mode 0.

    b. Right shift - The contents of the register generally will shift to the
right under the following conditions:

        (1) In Mode 0, data may enter serially on the left data line, shift
through the register, and leave on the right data line.

        (2) In Mode 1, data may enter serially on the left data line. The
right data line effectively will be open-circuited.

        (3) In Mode 2, data may leave serially on the right data line. The
left data line effectively will be open-circuited. Vacant spaces
will be filled with zeros.

        (4) In Mode 3, serial data neither may enter nor leave the register;
however, the contents will shift to the right, and vacated places
will be filled with zeros.

    c. Left shift - The contents of the register generally will shift to the
left under the following conditions:

        (1) In Mode 0, data may enter the right data line, shift through the
register, and leave on the left data line.

        (2) In Mode 1, data may leave serially on the left data line. The
right data line effectively will be open-circuited. All vacant
positions will be filled with zeros.

        (3) In Mode 2, data may enter serially on the right data line. The
left data line effectively will be open-circuited.

        (4) In Mode 3, data neither may enter nor leave the register; however,
the contents will shift to the left, and vacated places will be
filled with zeros.

J. PARALLEL COMMANDS

    a. CLEAR - sets register to zero.

    b. SET - sets register to all ones.

    c. OR - processes contents of register with value on parallel-data lines in a logical OR function.

    d. AND - processes contents of register with value on parallel-data lines in a logical AND function.

    e. EXCLUSIVE OR - processes contents of register with value on parallel-data lines in a logical EXCLUSIVE OR function.

    f. IN - loads value on parallel-data lines into register.

    g. OUT - outputs contents of register on parallel-data lines.

    h. SUB:

        (1) In Mode 1, adds to the contents of the register the two's complement of whatever is on the parallel-data lines. Generated carries may leave on the left serial line. The overflow indicator is not altered.

        (2) In Mode 2, adds to the contents of the register the one's complement of whatever is on the parallel-data lines. Carries may enter on the right serial line but may not leave on the left data line. The absence or presence of an overflow is registered.

        (3) In Mode 0, same as Mode 2, except carries may leave on the left data line. The overflow indicator is not altered.

        (4) In Mode 3, same as Mode 1, except carries may not leave on the left data line. The absence or presence of an overflow is registered.

    i. COUNT UP:

        (1) In Mode 1, internally adds one to the contents of the register and permits any resulting carry to leave on the left serial-data line. No data enters or leaves either the parallel lines or the right serial line.

(2) In Mode 2, adds to the contents of the register whatever is on the right serial-data line. No data enters or leaves either the parallel lines or the left serial line.

(3) In Mode 0, adds to the contents of the register whatever is on the right serial line and permits any resulting carry to leave on the left data line. No data enters or leaves the parallel lines.

(4) In Mode 3, internally adds one to the contents of the register. No data enters or leaves the register on any serial-data or parallel-data line.

j. COUNT DOWN:

(1) In Mode 1, internally subtracts one from the contents of the register and permits any resulting carry to leave on the left serial-data line. No data enters or leaves either the parallel lines or the right serial line.

(2) In Mode 2, subtracts one from the contents of the register and adds to this result whatever is on the right serial-data line. No data enters or leaves the parallel lines or the left data line.

(3) In Mode 0, subtracts one from the contents of the register and adds to this result whatever is on the right serial-data line and permits any resulting carry to leave on the left data line. No data enters or leaves the parallel lines.

(4) In Mode 3, internally subtracts one from the contents of the register. No data enters or leaves either the parallel lines or the serial lines.

k. AD:

(1) In Mode 1, adds the contents of the register to whatever is on the parallel-data lines and allows any resulting carry to leave on the left data line. The right serial-data line is open-circuited. The overflow indicator is not altered.

13

(2) In Mode 2, adds the contents of the register to whatever is on the parallel-data lines and the right serial-data line. Any overflows will set the overflow indicator. The left serial-data line is open-circuited. The absence or presence of an overflow is registered.

(3) In Mode 0, adds the contents of the register to whatever is on the parallel-data lines and the right serial-data line. Any resulting carry may leave on the left serial-data line. The overflow indicator is not altered.

(4) In Mode 3, adds contents of the register to whatever is on the parallel-data line. Any resulting carry will set an overflow indicator. The two serial-data lines are open-circuited. The absence or presence of an overflow is registered.

l. SM - same operation as AD, except the contents of the register are two's complemented during addition in Mode 1 and Mode 3. In Mode 0 or Mode 2, the contents of the register are one's complemented and added to whatever is on the right serial-data line and the parallel-data lines. Overflows occurring in Mode 1 or Mode 0 do not alter the overflow indicator. The presence or absence of overflows is registered on the overflow indicator in Mode 2 or Mode 3.

m. SMZ:

(1) In Mode 0, one's complements the contents of the register and adds whatever is on the right serial-data line to the contents of the register. Any resulting carry may leave the left serial line. Any overflow will not alter the overflow indicator.

(2) In Mode 1, two's complements the contents of the register and permits any carry to leave on the serial line. Nothing may enter the right serial line. Any overflow will not alter the overflow indicator.

(3) In Mode 2, one's complements the contents of the register and adds whatever is on the right serial line to the contents of the

register. Carries may not leave the left serial line. The absence or presence of an overflow will alter the overflow indicator.

(4) In Mode 3, two's complements the contents of the register. Serial data neither may enter the right serial line nor leave the left serial line. The overflow indicator will be at zero.

n. NO-OP - The NO-OP condition will inhibit the clock signal before the D-type flip flops.

K. TIMING

Transfer of data is accomplished by using a D-type flip flop which requires one clock pulse to transfer data on the input into the storage element.

The D-type flip flop consists of two double inverters which may feed back on themselves through transmission gates providing a stable state. When the clock is low, transmission gates "1" and "3" are active and gates "2" and "4" are inactive. This state permits the retention of data by the second inverter pair while allowing the incoming data to define the state of the first inverter pair.

When the clock undergoes a low-to-high transition, the states of all transmission gates are changed. During this transition the flip-flop input becomes isolated and the first inverter pair is stabilized by opening the feedback transmission gate holding the information which was on the data line. Meanwhile, the second inverter pair loses its feedback and a path is established from the first stage. For a set of D flip flops in a shift-register configuration, the effect of this transition is to permit the first stage of each flip flop to store information from the output of the previous flip flop before the second stage of the flip flops changes due to the new flip-flop input. During the high-to-low transition, the new data are transferred to the second inverter pair, in a manner similar to the original transfer, and the normal storage mode is assumed.

15

L. MODE-INDEPENDENT SWITCHES

The state of the control lines to the processing logic for the 15 operating instructions is shown in Table I. True data from the parallel inputs are gated into the parallel processor when K_1 is high. The pertinent equation is

$$K_1 = \bar{a}\bar{c}d + ac + c\bar{d} \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (1)$$

Complementary data from the parallel inputs are gated into the processor by K_2, which is given by

$$K_2 = bcd \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (2)$$

True information in the register is gated into the processor by K_3, which is given by

$$K_3 = \bar{a}\bar{b} + c \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (3)$$

and the complementary information is gated by K_4, where

$$K_4 = b\bar{c} \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (4)$$

Control K_5 is used to set all ones into the processor for one operand. Control K_5 can be gated through for a SET or can be used in COUNT DOWN. The pertinent equation is

$$K_5 = \bar{a}\bar{b}\bar{d} + \bar{a}\bar{c}\bar{d} \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (5)$$

The EXCLUSIVE OR can be inhibited when XI is high, which allows the OR operation to be formed. The switching equation is

$$XI = \bar{a} + \bar{d} \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (6)$$

The IN transmission gate is used to load the register in parallel, where

$$IN = ab\bar{c}\bar{d} \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (7)$$

TABLE I.   OPERATIONAL CODE FOR THE PARALLEL PROCESSOR ARRAY

Instruction	a	b	c	d	K_1	K_2	K_3	K_4	K_5	XI	IN	SUM	AND	CR	R_o	R	L	(2,3) R_n	(0,1) R_{T1}	(0,1) L_{T1}	(0,2) R_{T0}	(0,2) L_{T0}	(1,3) L_n	(3,1) Q	(3,1) J	(0,2) CAR	(0,1) M
										Mode Independent								Mode Dependent									
NOP	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
AND	0	0	0	1	1	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1
CNTD	0	0	1	0	1	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
CNTU	0	0	1	1	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
SMZ	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
SM	0	1	0	1	1	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
AD	0	1	1	0	1	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
SUB	0	1	1	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
SET	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
CLEAR	1	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
XOR	1	0	1	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0
OR	1	0	1	1	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
IN	1	1	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
L	1	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0
R	1	1	1	0	1	0	1	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0	1	1	0	0
R_o	1	1	1	1	1	1	1	0	0	0	0	0	0	0	1	1	0	0	1	0	1	0	0	0	0	0	0

Operational Code

17

The SUM transmission gate is used to perform all add-type instructions, where

$$\text{SUM} = \bar{a}\,(b + c) \quad \dots \dots \dots \dots \dots \dots \dots \quad (8)$$

Logic operations are performed by the AND and OR switches, where

$$\text{AND} = \overline{ab}\overline{c}d \quad \dots \dots \dots \dots \dots \dots \dots \dots \quad (9)$$

and

$$\text{OR} = a\bar{b} \quad \dots \dots \dots \dots \dots \dots \dots \dots \dots \quad (10)$$

$\bar{\text{M}}$. MODE-DEPENDENT SWITCHES

Code definitions for the modes were selected such that there is no need to decode. The definitions selected are as follows:

Mode	C_2	C_1
0	0	0
1	0	1
2	1	0
3	1	1

When high or "1," line C_1 or C_2 indicates which side of the processor is inhibited. Lead C_1 corresponds to the right side; lead C_2 corresponds to the left side.

Transmission gate Q (Figure 3) is used to force a "1" into the carry of the least significant adder during a COUNT UP or SUBTRACT operation and during Mode 1 or Mode 3. The equation is

$$Q = \bar{C}_1 + c\bar{d} \quad \dots \dots \dots \dots \dots \dots \dots \dots \quad (11)$$

A zero is forced into the carry by J, which is given by

$$J = C_1 \cdot c\bar{d} \quad \dots \dots \dots \dots \dots \dots \dots \dots \quad (12)$$

During Mode 0 and Mode 2, a carry is brought in from a previous array by the CAR transmission gate, where

$$\text{CAR} = \bar{C}_1 \cdot \bar{a} \quad \dots \dots \dots \dots \dots \dots \dots \dots \quad (13)$$

18

A carry may propagate out during Modes 0 and 1 by using an M, which is given by

$$M = \bar{C}_2 \cdot \bar{a} \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(14)}$$

N. SHIFT SWITCHES

The R and L transmission devices are mode independent gates used to perform the right and left shifts. R_N, R_o, R_{T1}, R_{T0}, L_{T1}, L_{T0}, and L_N are mode dependent shift controls. The equations are summarized as follows:

$$R_N = C_2 \cdot abc\bar{d} \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(15)}$$

$$R_o = abcd \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(16)}$$

$$R_{T,1} = \bar{C}_2 \cdot abc \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(17)}$$

$$L_{T,1} = \bar{C}_2 \cdot ab\bar{c}d \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(18)}$$

$$R_{T,o} = \bar{C}_1 \cdot abc \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(19)}$$

$$L_{T,o} = \bar{C}_1 \cdot ab\bar{c}d \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(20)}$$

$$L_n = C_1 \cdot ab\bar{c}d \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(21)}$$

$$R = abc \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(22)}$$

$$L = ab\bar{c}d \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \text{(23)}$$

Note: R denotes right, L denotes left

O. CONDITIONAL OPERATION

The clock pulse operates on condition. Three control lines will be defined to permit conditional instructions. These lines will be labeled A, B, and C. The following truth table defines interactions among A, B, and C:

C	B	A	Permits Operation
0	0	0	Yes
0	0	1	Yes
0	1	0	Yes
0	1	1	No
1	0	0	Yes
1	0	1	No
1	1	0	Yes
1	1	1	Yes

The truth table reduces to the condition that $(\bar{A} + B \cdot C + \bar{B} \cdot \bar{C})$ data-transfer operation is to take place. This expression combined with the clock pulse accomplishes the data transfer providing the processor is not in the NO-OP state.

P. OVERFLOW INDICATOR

Switch SIGNA operates and puts the truth sum in the register under either one of the following conditions: overflows have not been detected; or the mode of operation and the instruction are such that overflow detection is not needed. This condition can be summarized by

$$SIGNA = \overline{(\bar{S}_f \cdot S_1 \cdot S_2 + S_f \cdot \bar{S}_1 \cdot \bar{S}_2) \cdot C_2 \cdot \bar{a}b}$$

$$+ \bar{C}_2 \cdot \bar{a}b + \bar{a}\bar{b}c \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \quad (24)$$

Switch SIGNB operates when an overflow occurs, and a "1" is placed in the overflow flip flop. The complement of the most significant bit sum output also is placed in the register; hence,

$$SIGNB = (\overline{\bar{S}_f \cdot S_1 \cdot S_2 + S_f \cdot \bar{S}_1 \cdot \bar{S}_2}) \cdot \bar{a}b \cdot C_2 \quad \ldots \ldots \ldots \quad (25)$$

A zero is placed in the overflow flip-flop when the following condition is true:

$$\overline{(\bar{S}_f \cdot S_1 \cdot S_2 + S_f \cdot \bar{S}_1 \cdot \bar{S}_2) \cdot \bar{a}b \cdot C_2} \quad \ldots \ldots \ldots \ldots \ldots \quad (26)$$

The overflow flip flop can only be clocked during ($\overline{ab} \cdot C_2$) or during the IN instruction. Data may be entered or removed from the overflow flip flop, on the OVERFLOW I/O line, and during the IN and OUT commands, respectively.

Q. EXPANSION TO 16-STAGE PROCESSOR

The four-stage parallel processor is designed such that four processors can be interconnected monolithically to form the 16-stage processor. The wafer will be diced such that four operating four-stage processors will form a 16-stage processor. The interconnection scheme is shown in Figure 5. In sections 2 and 3, the C_1 and C_2 mode controls are tied to ground; therefore, these sections are held in Mode 0. In section 1, C_2 is tied to ground; therefore, the section can operate only in Mode 0 or Mode 1. In section 4, lead C_1 is tied to ground; therefore, this section can operate only in Mode 0 or Mode 2. The modes for the 16-stage processor are determined by inputs C_1 and C_2, as shown.

The bypass leads of sections 1 and 4 are connected as shown in Figure 5. When C_1 and C_2 are both "1," indicating Mode 3 the 16-stage register is bypassed; and the left serial-data line of chip 4 and the right serial-data line of section 1 are connected.

Leads R_{o1} of section 4 and R_{o2} of section 1 are connected as shown in Figure 5 to allow the rotate operation. R_o denotes the rotate (cycle shift) function.

The ZERO INDICATOR leads are connected as shown. If a "0" occurs in the 16-stage parallel processor, a "1" will appear on the indicator of the leftmost unit.

R. ELECTRICAL PERFORMANCE

The parallel processor units delivered to NASA met all original electrical specifications with respect to speed and standby leakage power. At standby, the current drain was typically less than 1 microampere.

All units were fabricated with a "standard threshold voltage" process, giving thresholds of approximately 1.9 volts for both n-type and p-type tran-
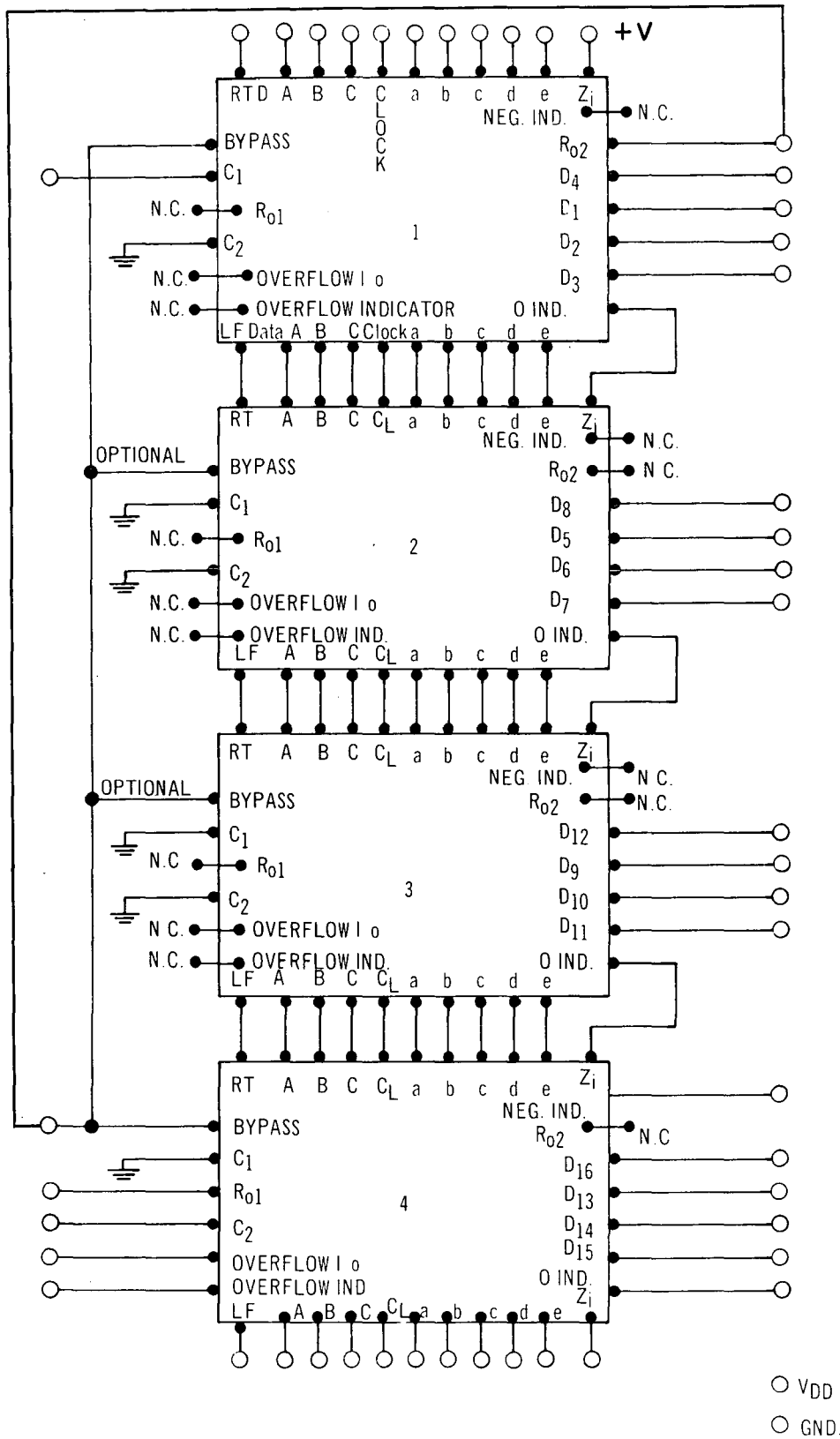
Figure 5. Schematic Interconnection of Four Four-Stage Chips

sistors. As shown in Figure 6, worst-case four-bit add performance at a 10-volt power supply was less than 1 microsecond with most other instructions faster. This speed performance could be further increased by a factor of approximately 2 by using a "low" threshold voltage process and a 12-volt supply.

CLOCK

OUT LINE

DATA 1
DATA 2

DATA 3

DATA 4

LEFT DATA LINE
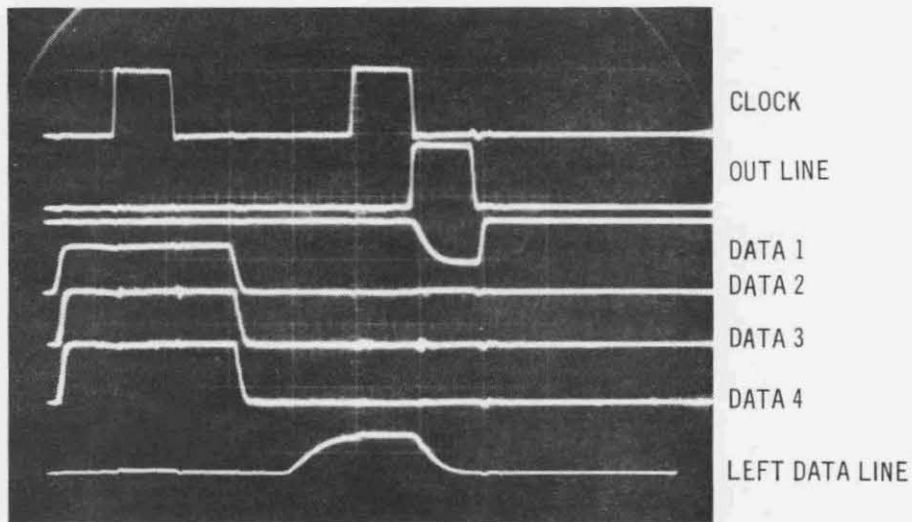
V_{DD} = 10 V   TIME SCALE = $0.5 \mu s/cm$

↑ APPLY DATA AND IN INSTRUCTION

↑ ON RISING EDGE OF CLOCK LOAD DATA (1111)
IN REGISTER

↑ CHANGE DATA TO 1000 AND APPLY ADD INSTRUCTION

↑ LOAD RESULTS OF ADD IN REGISTER

↑ APPLY OUT SIGNAL AND OBSERVE RESULTS OF ADD
ON DATA LINES

THE PHOTOGRAPH DEPICTS THE MINIMUM ADD PERFORM
TIME FOR PROPER CIRCUIT OPERATION.

05698V

Figure 6.   ADD Perform Time

24

SECTION III

IMPROVED PHOTOMASK FABRICATION

A. LIMITATION IN LARGE-CHIP SIZE PHOTOMASK TECHNOLOGY

During Phase I of this program to fabricate an 800-transistor LSI mono-
lithic parallel processor COS/MOS chip for NASA, it became apparent that the
major, and unanticipated, technical difficulty was in obtaining large-chip
size, low-defect-level photomasks of good resolution and dimensional fidelity.
These limitations are associated with old piecing techniques of photomask
fabrication for the large chips, using handcut artwork and multiple photo-
graphic reductions.

Because the large-chip size (0.155 inch by 0.146 inch) of the parallel
processor exceeded the maximum useful capability of the reducing lenses at
that time, it became necessary to prepare each photomask in four parts and
step-and-repeat each of the quadrants individually. In this process, it was
necessary to insert each reticle "blind" into the photorepeater. It was ob-
served that the accuracy with which each quadrant could be located mechani-
cally with respect to its neighbors was such that the desired maximum design
tolerance of 50 microinches could not be consistently maintained, and random
positioning errors greater than 100 microinches within a chip resulted. Such
pattern misregistration cannot be tolerated in the COS/MOS fabrication-sequence
procedure which requires the successive alignment of seven photomasks, each
subject to independent random quadrant location. In three masks n^+, p^+, and
metallization, relative alignment is absolutely critical to avoid shorts and/or
to ensure that the gate metal overlaps both source and drain diffusions for
proper MOS transistor operation.

Extraordinary alignment techniques had been needed to deliver the limited
number of parallel processors fabricated under the original Phase I program.
Such techniques obviously were not adequate to support a reproducible process

for more than a limited supply of parallel processor engineering samples. In
order to fabricate additional large-chip COS/MOS devices in a routine manner
successfully, it was necessary to employ better mask fabrication methods.

B. AUTOMATED PHOTOMASKING EQUIPMENT

To fabricate improved parallel-processor photomasks during Phase I(a) of
the program, RCA used technically superior photomask-making equipment not
available to the industry at the start of the original Phase I program. This
equipment consists of: a digitizer plotter for digital tape preperation, an
automatic reticle generator, and a chrome-master photorepeater.

Automatic artwork generation required that design information be digitized.
Tapes to automatically draft and generate reticles were prepared from digital
data entered mechanically from a digitizer plotter as indicated in Figure 7.
The redesigned parallel processor incorporated a number of minor improvements
which increased its speed.

C. TYPE 1600 10X RETICLE GENERATOR

Once debugged tapes were available, 10X photographic reticles were gener-
ated by the Mann automatic pattern generator; these 10X reticles are used to
make the final photomasks in one step with a special Mann chrome-master photo-
repeater. Figure 8 shows the David W. Mann Company type 1600 reticle generator
with a PDP-8/S central control digital computer. Photomasks fabricated with
this equipment at RCA have shown both superb image detail and dimensional fi-
delity, and represent a very significant advance in photomask art.

Use of this equipment eliminated previous photomask problems, which were
observed in the Phase I parallel-processor program. Here, photographic re-
ductions of pieced 500X artwork presented major technical difficulties in
image definition, distortion, registration within masks, corner rounding, di-
mensional control, and timely delivery of masks. The ability to fabricate 10X
reticles directly solved these problems.

The type 1600 pattern generator is a fully automatic, computer-directed,
highly accurate and reliable system for producing 10X reticles without inter-

A BETTER QUALITY OF THESE

PAGES IS REPRODUCED AT THE END OF

THIS PUBLICATION
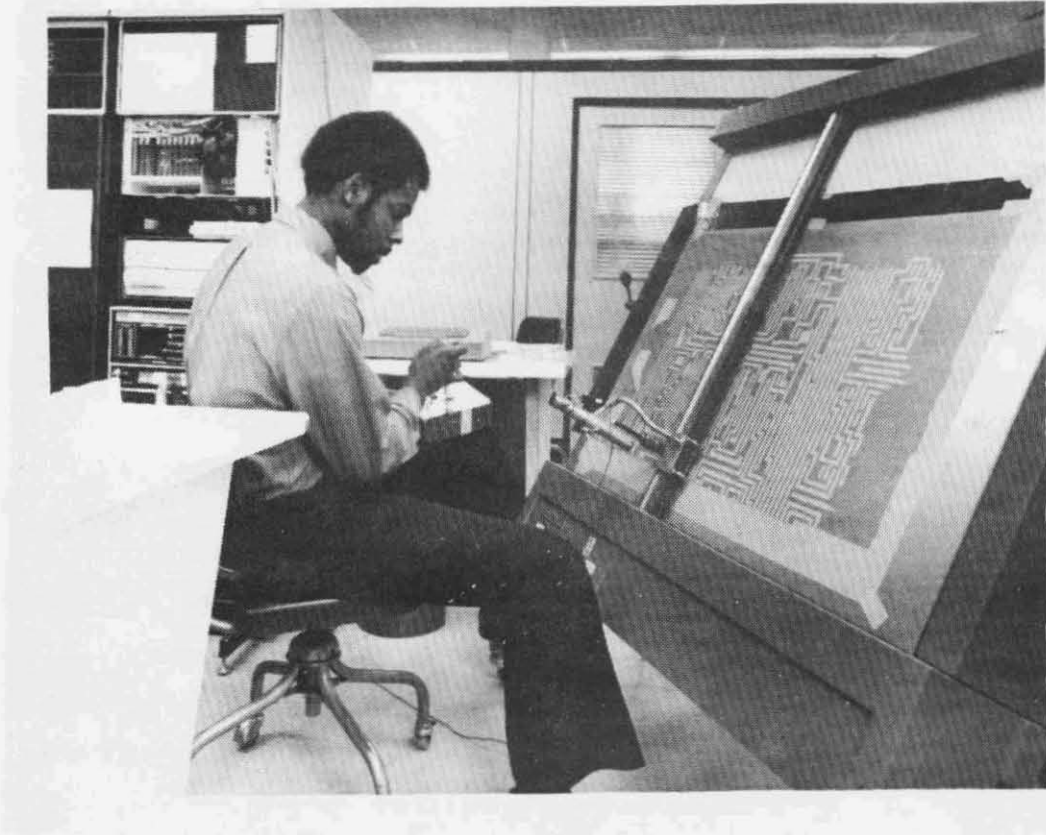
05637 V

Figure 7. Digitizer Plotter System Employed to Digitize Parallel Processor Data
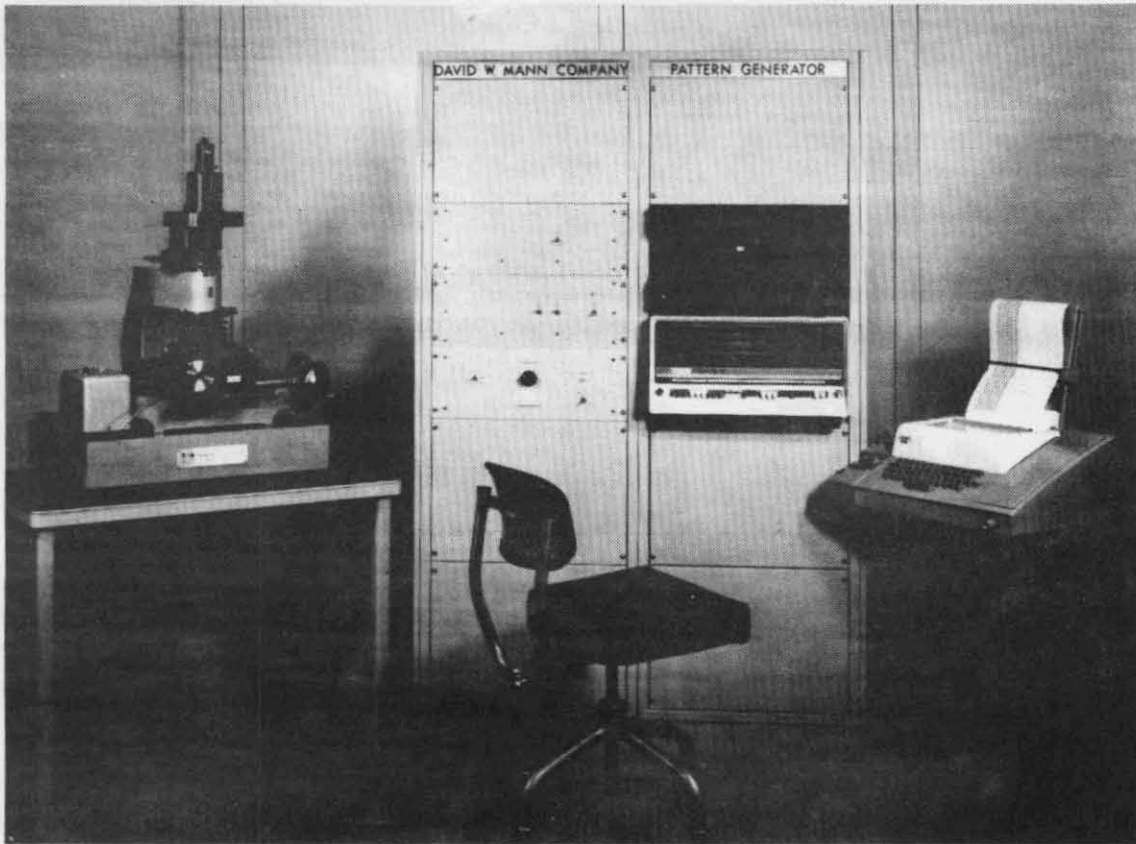
03641V

Figure 8. Type 1600 Automatic Reticle Generator

mediate artwork generation and reduction. Digital input tape controls all automatic functions. Input data on the nine-channel magnetic tape includes X and Y coordinates of the center of exposure, and height and width dimensions of the rectangular exposure. Height and width of the area exposed in a single flash on the 10X pattern may be varied in 240 discrete steps from 0.5 to 120 mils, a total of 57,600 sizes. The microset scale for both the scanning and stepover axes assures positional precision of ± 0.00001 inch on the 10X reticle up to 3 by 3 inches. The guaranteed reticle resolution is superb and more than adequate for the parallel processor, i.e., 650 line pairs per millimeter over the entire circuit pattern area.

D. TYPE 1795 CHROME-MASTER PHOTOREPEATER

Photomasks for this project were prepared from the 10X reticles using a new David W. Mann type 1795, six-head, chrome-master photorepeater. RCA received the first model available and the parallel processor was one of the first projects to utilize this equipment.

The major technical advantages of this equipment are the generation of durable, ultralow-defect level, chromium masters with superior edge acuity; and the exceptionally large, guaranteed chip size (0.250 inch by 0.250 inch) over which precision control of image resolution and dimensional accuracy can be maintained.

The importance of ultralow photomask defect levels in an 800-transistor LSI array is clear. A single imperfection in any of the seven photomasks used in the COS/MOS fabrication sequence will cause the chip in which the defect falls to be inoperative. The enhanced durability and freedom from pinholes of chrome masters will minimize imperfections and ensure that pattern damage does not occur when making contact prints from the master.

The wide 0.25-inch field of the improved lenses used in the 1795 photo-repeater were also of major significance in this program. Maximum useful chip area, which could be usefully photorepeated with older equipment for the parallel-processor program, was less than 0.120 inch by 0.120 inch. This area was smaller than the parallel-processor chip size (0.155 inch by 0.146 inch).

In the previous program, it was necessary to attempt to piece final images by multiple photorepeater runs with resulting registration problems. This class of problem was avoided with the new equipment since the new parallel-processor chip size was well within the limits of the 1795 photorepeater.

E. Processing

As anticipated, the improved photomasks resulted in the capability to make the parallel-processor reproducibly. Yield was observed in the first wafer processed in a high-yield line and delivery of 25 low-leakage units was made to NASA, fulfilling all delivery and performance specifications on this project.

SECTION IV

REFERENCES

1. A.A. Alaspa, and A.G.F. Dingwall, "COS/MOS Parallel Processor Array", IEEE Jr. Solid State Circuits, Vol. SC-5, No. 5, October 1970.

2. R.J. Lesniewski, "A Large Scale Integration (LSI) Computer Concept Utilizing only Five Types of General Purpose Digital Arrays," Master of Science Thesis, University of Maryland; 1970.

3. R.J. Lesniewski and F.J. Link, "A Complementary - MOS Space Craft Data Handling System," Proc. GOMAC Conference, 1969.

4. R.J. Lesniewski and D.H. Schaefer, "Goddard Space Flight Center Ultra Low Power Computer Development Program," Proc. Third NASA Intracenter Microelectronics Conference, February 1968.