Spring 1996

Flash ADC using 2μm CMOS P-well technology : design and test

Joseph E. Levinson
New Jersey Institute of Technology

Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

FLASH ADC USING 2μm CMOS P-WELL TECHNOLOGY: DESIGN AND TEST

by
Joseph E. Levinson

This thesis describes the design, implementation and test for a new CMOS analog-to-digital converter IC chip. In designing the analog-to-digital converter in this thesis a radically different comparator design that is only available with CMOS logic. The design utilizes a single CMOS inverter as an ultra-high gain amplifier. This approach reduces the circuit dependence upon matching of the transistors similar to the traditional method. This new design requires less area since the comparator utilizes fewer transistors.

Flash analog-to-digital converters use 2^n - 1 comparators to do a single conversion where n is the number of bits used. These comparators are traditionally made with differential transistor amplifiers in order to obtain matched characteristics. Effects of mismatch in current gains and base emitter voltage is reduced in the new design since the amplifier topology does not depend entirely upon perfectly matched transistors.

FLASH ADC USING 2μm CMOS P-WELL TECHNOLOGY: DESIGN AND TEST

by
Joseph E. Levinson

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements of the Degree of
Master of Science in Electrical Engineering

Department of Electrical and Computer Engineering

May 1996

APPROVAL PAGE

FLASH ADC USING 2μm CMOS P-WELL TECHNOLOGY: DESIGN AND TEST

Joseph E. Levinson

Dr. William N. Carr, Thesis Adviser — Date
Professor of Electrical and Computer Engineering, NJIT
Department of Physics

Mr. Siu Bor Lau — Date
Staff Engineer, AlliedSignal Aerospace, Teterboro, NJ

Dr. Kenneth Sohn — Date
Professor and Associate Chair of Electrical and Computer Engineering, NJIT

BIOGRAPHICAL SKETCH

Author: Joseph E. Levinson

Degree: Master of Science

Date: May 1996

Undergraduate and Graduate Education:

- Master of Science in Electrical Engineering
  New Jersey Institute of Technology, Newark, NJ, 1996

- Bachelor of Science in Computer Science
  Clarkson University, Potsdam, NY, 1989

- Bachelor of Science in Electrical Engineering
  Clarkson University, Potsdam, NY, 1988

Major: Electrical Engineering

Professional Background:

- Senior Engineer
  AlliedSignal Aerospace, Teterboro, NJ, Feb. 1994 to Present

- Engineer
  AlliedSignal Aerospace, Teterboro, NJ, Mar. 1989 to Feb. 1994

- Associate Engineer
  AlliedSignal Aerospace, Teterboro, NJ, June 1988 to Mar. 1989

- Associate Engineering Aid (Co-Op)
  AlliedSignal Aerospace, Teterboro, NJ, Jan. 1987 to Aug. 1987

This thesis is dedicated to
my mother - Dr. Nina Levinson

ACKNOWLEDGMENT

Sincere gratitude to my advisor, Dr. William N. Carr, for his guidance, friendship, patience and understanding throughout this research.

Special thanks to Dr. Kenneth Sohn and Mr. Siu Lau for their time serving as members of the committee.

A warm appreciation and thanks for the people at MOSIS for the fabrication of the device in particular as they were hit by a considerable earthquake at that time.

I wish to thank Dr. Henry Domingos of Clarkson University for my introduction into the world of VLSI. design, simulation and testing.

And last but not least thanks to my many friends and colleagues at AlliedSignal for their friendship, support and understanding with this endeavor.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF FIGURES

Figure **Page**

CHAPTER 1

INTRODUCTION STATE OF THE ART FLASH ADC'S

1. 1 Introduction

ADC's are essential for converting the real world into a form that computers may understand and use data for computations and data logging. For example an automobile's engine computer takes data from sensors on the engine, masks computations and modifies timing and fuel mixtures. The space program also uses ADC's to take data for the purpose of logging it for the future. When a problem arises the data may be analyzed to give an indication of what this problem might have been. These are examples of where an analog-to-digital converter is used. In both cases a low speed analog-to-digital converter can be used. Analyzing video and radar signals require a high speed analog-to-digital converter.

There are many circuit realizations for analog-to-digital converters. There are some low speed types such as single and dual slope converters. There are medium speed devices such as successive approximation and delta sigma converters, and then there are the high speed flash converters.

1. 2 Single and Dual Slope Converters

The low speed single and dual slope converters consist of charging a capacitor and then discharging and comparing these two times. These require a small area on a chip. The conversion is performed by charging the capacitor with a current source. The voltage developed across the capacitor is compared to the unknown input voltage. A

comparator signals when the two are equal. At the same time the capacitor is charging up, a counter is being driven by an oscillator and that counter has a digital value representing the unknown voltage. The single slope converter is very dependent upon the accuracy and stability of the capacitor and an oscillator. This results in a slow and inaccurate quantization of an unknown voltage. A similar technique, the dual-slope integration analog-to-digital converter relies on two current sources to charge and discharge the capacitor. The initial charging of the capacitor uses a fixed time period, while a counter is being run up from an oscillator. The second phase is discharging of the capacitor proportional to the unknown voltage, while running the counter down. The ratio of the difference in time gives the quantization of the unknown voltage. This method does not rely upon the accuracy of the oscillator or the stability of the capacitor and it is used in inexpensive digital multimeters. These converters are typically offered in 7 to 8 bit quantization types; for higher quantization a better type of converter must be used. Both of these types of converters are low cost since they occupy a small area on a silicon chip.[1]

1. 3 Successive Approximation and Delta Sigma Converter

The medium speed analog-to-digital converters are extremely popular in a variety of applications such as data acquisition computer boards and high performance digital multimeters. The successive approximation type of analog-to-digital converter utilizes

---

[1]Horowitz, P. & Hill, W. , 1989, "Digital Meets Analog, Analog/Digital Conversion", *The Art of Electronics* , 2ed, Press Syndicate of the University of Cambridge, NY, NY, pp 621 to 630

a digital-to-analog converter. The counting is done by a successive approximation register that is driven by a clock. The output of this register is fed into a digital-to-analog converter the output of which is given to a comparator that compares that output with that of the unknown. When the two are equal, the value in the successive approximation register is the quantization of the unknown input. This type of converter takes $\log_2 N$ clock pulses where N is the number of bits. The other major type of medium speed analog-to-digital converter is the delta-sigma converter. This type relies upon the charging of a capacitor with pulses of current until the voltage is equal to the unknown input. During that time a counter keeps track of the number of pulses which result in the quantization of the unknown. These devices are available in 8 to 16 bit types; some with 22 bits are becoming available.

1. 4 Flash Converters

This high speed ADC would be a flash converter, meaning that one clock cycle will perform the conversion. Various manufactures produce different varieties of flash ADC's. Datel manufactures a CMOS flash ADC, the ADC-207 and ADC-208. Harris also makes some video flash analog-to-digital converts that utilize a similar architecture to the one presented in this thesis.

> "The CA3306 family are CMOS parallel (FLASH) analog-to-digital converts designed for application demanding both low power consumption and high speed digitization. Digitizing at 15Mhz, for example, requires only about 50mW.
>
> The CA3306 family operates over a wide, full scale signal input voltage of 1V up to the supply voltage. Power consumption is as low as 15mW, depending upon the clock frequency selected.

> The CA3306 types may be directly retrofitted into CA3300 sockets, offering improved linearity at a lower reference voltage and high operating speed with a 5V supply.
>
> The intrinsic high conversion rate makes the CA3306 types ideally suited for digitizing high speed signals. The overflow bit makes possible the connections of two or more CA3306s in series to increase the resolution of the conversion system. A series connection of two CA3306s may be used to produce a 7-bit high speed converter. Operation of two CA3306s in parallel doubles the conversion speed (i.e., increase the sampling rate from 15Mhz to 30Mhz).
>
> Sixty-four paralleled auto balanced comparators measure the input voltage with respect to a know reference to produce the parallel bit outputs in the CA3306. Sixty-three comparators are required to quantize all input voltage levels in this 6-bit converter, and the additional comparator is required for the overflow bit."[2]

The comparator circuit is similar to the design as is presented in this thesis. The Harris device has the same overall block diagram as the device in this thesis, except for the overflow comparator. The Datel device has some differences in the location of a set of latches used to capture the data that has been converted. The Datel devices (ADC207 and ADC208) latches the output of the comparator, while the design within latches the output of the encoding array. This difference in location of the latches would appear to account for the difference in speed of the two Datel designs. The Harris unit will operate up to 15Mhz while the Datel method will operate up to 20Mhz. The device presented in this thesis has a calculated maximum speed of 12Mhz. This concurs with one designed by Harris.

> "The ADC-208 utilizes an advanced VLSI 1.2 micron CMOS in providing 20 MHz sampling rated at 8 bits. The flexibility of

[2]--, 1994, *Harris Semiconductor Data Acquisition 1994*, Harris Semiconductor, Palm Bay, FL, pp 6-16 to 6-30

> the design architecture and process delivers effective bit rates to 30 MHz in the burst mode, one shot mode conversion times of 35 nanoseconds, low power modes to 150mW, latch-up free operation without external components and operation over the full military temperature range.
>
> The ADC-208 has 256 auto-zeroing comparators which are auto-balanced on every conversion to cancel out any offsets due to temperature and/or dynamic effects. These comparators sample the difference between the analog input and the reference voltages generated by the precision reference ladder network. Parallel output data and the overflow pin have Three-State outputs. The overflow pin allows cascading two devices for 9 bit operation."[3]

There are many other types of analog-to-digital converters being designed. Many of these devices are only present in papers presented at conferences such as the International Solid-State Circuits Conference. One of these presented during the 1995 conference was *A 12bit 50Msample/s Two-Stage ADC*[4]. It utilizes a cascaded folding amplifier based on a complementary bipolar process. The use of bipolar technology gives the device an overall advantage in speed. Other designs have also appeared in the Journal of Solid State Circuits. A recent one appeared in the Dec. 1995.

> "*Abstract* -- A CMOS analog to digital converter based on the folding and interpolating technique is presented. This technique is successfully applied in bipolar A/D converters and now also becomes available in CMOS technology. The analog bandwidth of the A/D converter is increased by using a transresistance amplifier at the outputs of the folding amplifiers and, due to careful circuit design, the comparators need no offset compensation. The result is a small area (0.7sq mm in 0.8μm CMOS), high speed (70MS/s), and low-power (110mW at 5V

---

[3]--,1991,*DATEL Databook Volume 1*,Datel Inc, Mansfield, MA. pp 2-5 to 2-8

[4]Gosser, R. & Murden, F., Feb. 17, 1995, *A 12b 50msample/s Two-Stage ADC*, Analog Devices, Greensboro, NC, at the IEEE International Solid-State Circuits Conference

> supply, including reference ladder) A/D converter. A 3.3V supply version of the circuit runs at 45MS/s and dissipates 45mW."[5]

This device utilizes a comparator circuit that may be realized in both CMOS and bipolar technologies. The comparator is a set of matched devices to form a current mirror; this does require the matching of transistors, but results in a very small comparator design. It utilizes an encoding technique that makes it possible to utilize smaller analog-to-digital sections and then encode them together. The smaller comparator would inherently result in smaller area and therefore reduce the propagation times.

This is a look at a few commercially available and recent publications of current analog-to-digital converters available today. There is a wide variety of choice for the designer, depending on whether they want high speed, high accuracy, or low cost. The high cost would be full custom design, while the lowest cost would be a simple device such as a dual slope converter that is commercially available. Their size have become small enough to be incorporated into micro-controllers.

---

[5]Nauta, Bram & Venes, Ardie G. W., Dec. 1995, "A 70-MS/s 110-mW 8-b CMOS Folding and Interpolating A/D Converter", *IEEE Journal of Solid State Circuits*, Piscataway, NJ, Number 12, Volume 30, pp 1302 to 1308

CHAPTER 2

CHIP ARCHITECTURE AND SPECIFICATIONS

The design specification for this chip is as follows:

ADC Type	: CMOS Flash
Output	: TTL Tri-State
Vin	: +1.25V to +3.75V
Vref	: Vref(lo) - +1.25V Vref(hi) - +3.75V
Supply	: 5 Volts (separate Vcc's for Analog (Comparator) section and Digital Sections)
Technology	: 2μm P-Well MOSIS compatible CMOS technology.
Die size	: 2.25mm X 2.25mm
Clock	: 12 MHz
Comparator	: Clocked CMOS comparator
Encoding	: Pass transistor logic and static mask programmed CMOS ROM
Pin-out	: Figure 1
Floor Plan	: Figure 2

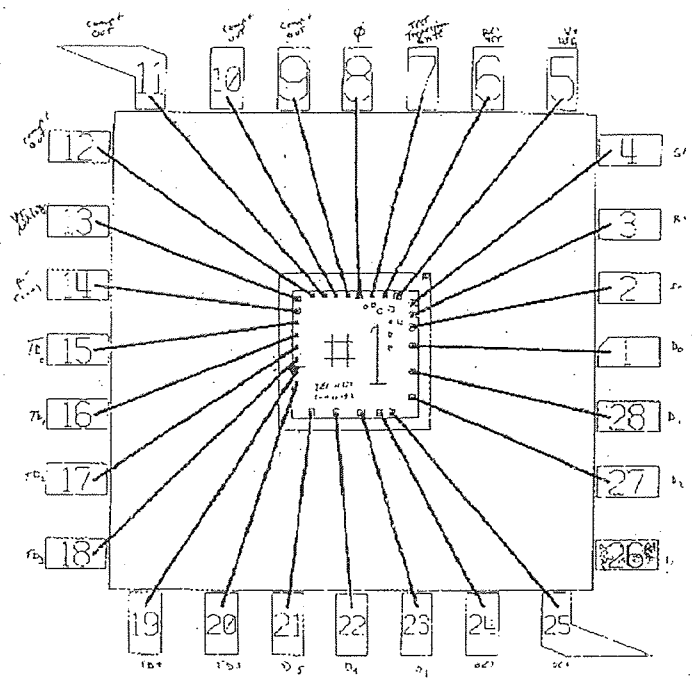

Figure 1

Pin-Out of Die from MOSIS

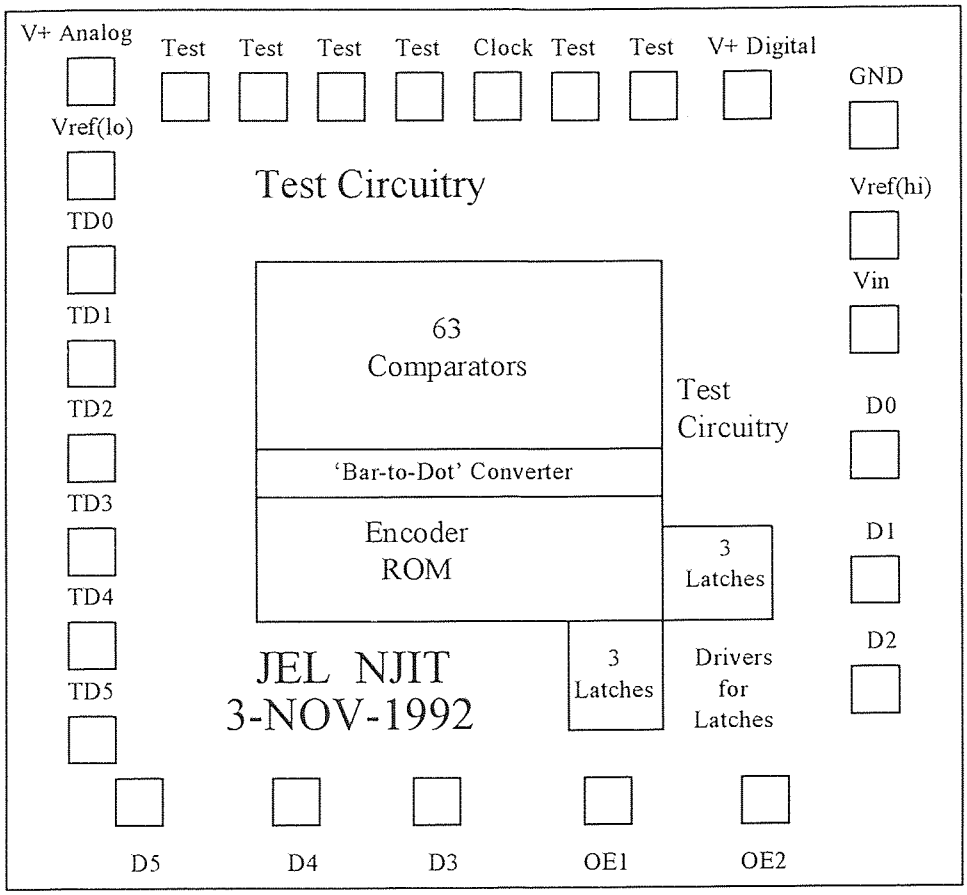

Figure 2

Floor Plan of Chip Layout

CHAPTER 3

CIRCUIT DESCRIPTION AND LOGIC SCHEMATIC

3. 1 Comparators

The main part of the ADC is the comparator. The CMOS comparator does not follow the standard differential transistor design. The comparator makes its 'decision' process by using a capacitor. The capacitor is initially charged with a reference voltage on one of the two plates. The other plate is connected to an inverter used in a linear mode to provide amplification. The clock switches the reference voltage off the capacitor and opens the inverter amplifier. After a moment of delay the unknown voltage is applied to the same side of the capacitor as was used for the reference voltage. The capacitor is actually making the 'decision' whether the unknown voltage is greater than the reference voltage or less. The inverter then amplifies the result of the 'decision'. Previously, when the inverters' output was fed back on it's input the result was to preset or bias the inverter in a region of high gain, resulting in a high speed amplifier that will reach saturation at either rail.[6] The schematic for one of the comparator is shown in Figure 3. There are 63 of these comparators in this ADC design to yield 6 bits. The reference voltage is derived off a resistive ladder.

Since these comparators utilize a single capacitor to make the comparison 'decision', they are not susceptible to device mismatches as in a differential amplifier. The transistors are used for a high-gain amplifier, being driven into saturation. The

[6] --,1991, *DATEL Databook Volume 1*, Datel Inc, Mansfield, MA, pp 2-1 to 2-8

effects of mismatching would only result in a speed degradation. In the differential amplifier, a mismatch in transistor parameters would result in inaccuracies.

3. 2 Encoding

The output of these comparators is fed into a circuit which makes the output with only one high bit and all the other ones low. This is implemented using pass transistor logic. To minimize the circuit area, the actual circuit uses two transistors. Had the circuit been realized with a transmission gate, the circuit would have required three more transistors, a total of five, which would have more than doubled the area. The next part of the circuit allows the one V_{th} drop from the pass transistor due to the level restoration buffers. This circuit I will call a 'bar to dot' converter. The output of the comparator/pass transistor logic is then fed into a ROM encoder. This encoder has six bits for output and 64 bits for input. The circuitry is designed using the contact mask for the actual programming of the ROM.

3. 3 Basic Layout

The actual design/layout of the chip became fairly simple because of the redundancy of a few simple cells. The basic comparator was laid out and optimized. Then four of these comparators with appropriate clock driving circuitry and a four stage resistive divider network were grouped together. Sixteen of these groups of four comparators were then placed onto the actual chip. These cells were designed to be placed next to each other for power, ground and other various input connections. The outputs from

these blocks of four comparators were fed into single 'bar to dot' converters. These converters were designed so that four of them took the same geometric height as the blocks of four comparators. The inputs and outputs were designed/laid-out so that the connections would be there and no manual connection routes would be needed. The encoder ROM array was designed so that one row could be drawn-up and then reproduced 64 times over. The only thing that was omitted on this ROM was the contact mask for encoding. When 64 rows were assembled with the appropriate driving circuitry the encoding, which was the contact to metal 1 layer, had to be added manually. The encoder ROM took the same height the 64 comparators and other previous stages took. This formed the core of the 6 bit analog-to-digital converter shown in Figure 4.

3. 4 Support Electronics

The supporting circuitry also had to be added. This included output latches, clock driving schemes, output drivers, input/output pads, power and ground distribution. The output latches were a simple design utilizing a master and slave latch to create an edge triggered device. The master and slave portions differed in the polarity of their clocks. Each portion utilized two inverters and two transmission gates. The actual 'memory' of this circuit depends upon the storage of charge on a node capacitance. The wiring from the output of the encoder ROM to the 6 latches was done manually with a great expense of time. The clocking was done utilizing a fan-out tree for the comparators to minimize clock skew. The clock signal was also routed to the latches with a small time delay to allow for the propagation delay of the analog-to-digital

converter. The output drivers were a 4 stage set of tri-state buffers. Each buffer was approximately 7 times the gain of the previous stage. Such buffering is necessary to drive large capacitive loads. The transconductance of the transistors for each stage had to be increased with a change in W/L. The final output transistors had a transconductance of 0.064 μ-mhos for the NMOS transistor and 0.052 μ-mhos for the PMOS transistor. The bonding pads have an average capacitance of about 1pF. The driving capacitance that was chosen was to be able to drive about 10pF. The wiring to the pads was also done manually. The final item that had to be done manually was to connect all the power and ground signal to the appropriate bonding pads.

3. 5 Critical Path

The critical path of this circuit is a propagation delay from one stage to the other. The actual propagation delay for one conversion is simple; it involves the delays in the clock driving circuitry that is designed in a fan-out to provide equal clocking for all the comparators. The comparators have their own delays that should be the same for each one. The 'bar to dot' converter has to wait for two comparators to have an output for it to work properly. The final delay is in the encoding ROM. From there the output is latched into six D type latches. The clocking for these latches is skewed by some inverters and physical length of a wire run on the chip so as to capture only 'good' data from the ROM encoder. A partial schematic is shown in Figure 5 with the critical path in heavy lines.

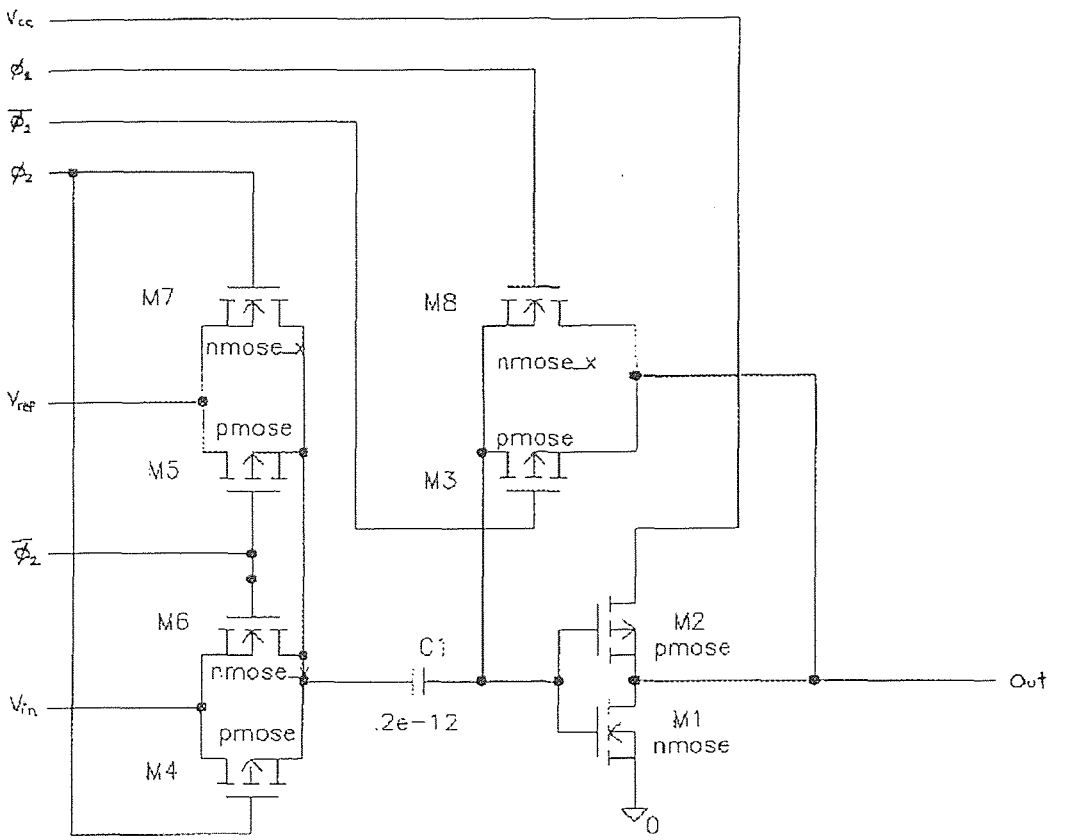

Figure 3

Schematic for the Clocked Comparator Cell

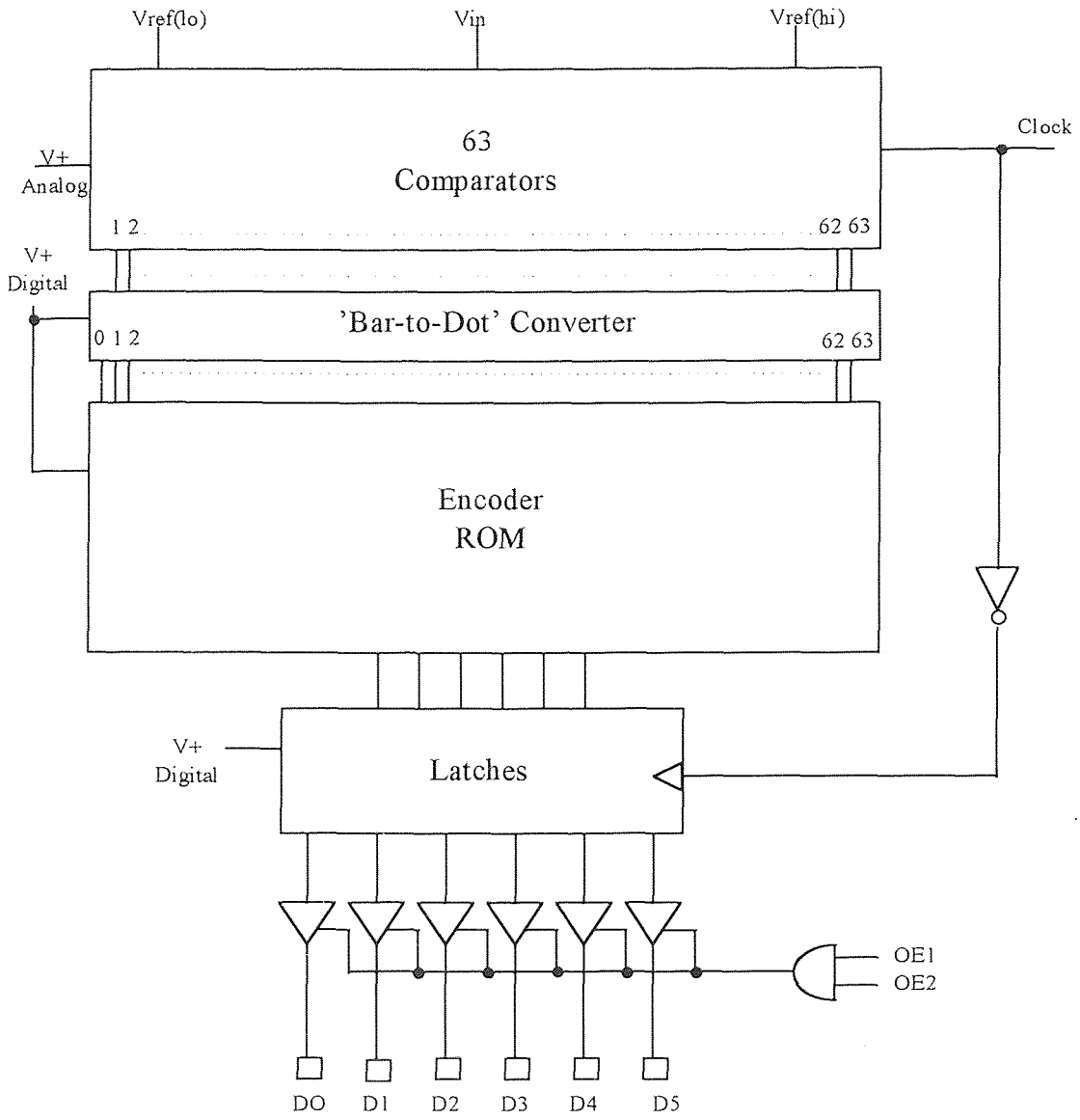

Figure 4

Block Diagram for the 6 Bit Analog-to-Digital Converter

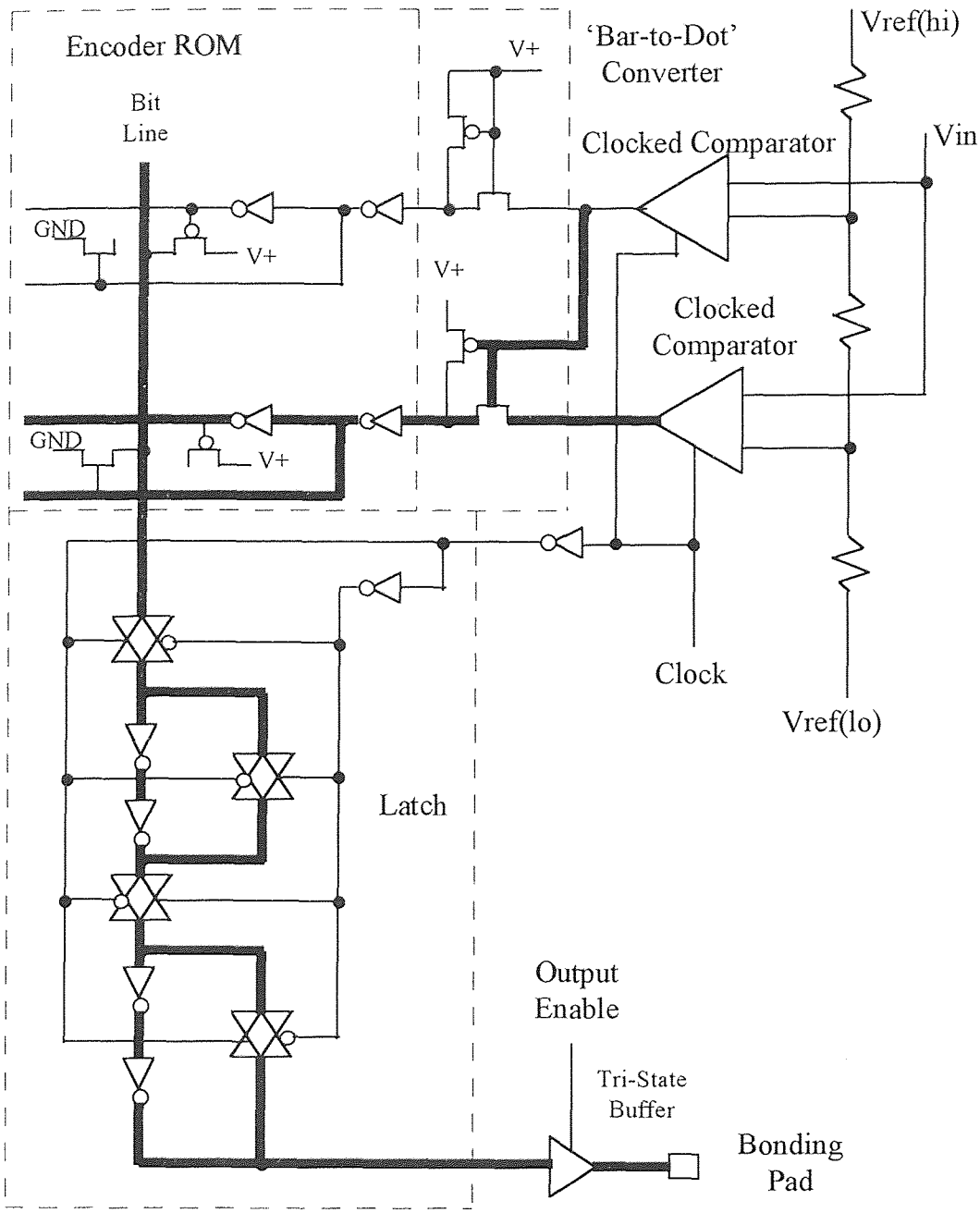

Figure 5

Critical Path

CHAPTER 4
CIRCUIT SIMULATION RESULTS

4. 1 Spice Results

The full 6 Bit ADC contained over 6000 MOS FET's. This would have been too many devices for PSPICE to handle in a reasonable amount of time. In order to do the original design, a student version of PSPICE was used. This version could only handle 10 transistors as a maximum. The circuit was 'cracked' into parts for simulation, a comparator, a few different types of inverters, and a latch. These simulations suggested the ADC should be able to handle at least a clock rate of 12MHz. For a final simulation a full blown version of PSPICE was utilized to handle 218 MOS FET's to simulate a 3 bit analog-to-digital converter. This gave a full picture of the critical path. The maximum clock rate that was obtained with the later simulation was 12MHz with actual process parameters for the transistors. The circuit description, input file, for the SPICE runs was similar in breakup to the actual chip. The first subcircuit that was generated was for a single comparator. Then four of these comparators were grouped together with the appropriate clocking circuitry, the two phase clock. Then these were fed into the two part encoding array and then latched by three latch subcircuits. While looking at internal nodes of the circuit, it became obvious where improvements could be made in the circuit. The outputs from the individual comparators is shown in the plot in Figure 6. The output from the comparators were fed directly into the encoding array; this created a slow path. The output from the two levels of encoders are shown in Figure 7, the output from the 'bar to dot' converter to the encoder ROM is shown in

Figure 8. The output from the ROM and latches is shown in Figure 9. The commercially available device from Datel shows that after the comparator there is a latch to hold the value from the conversion. I think this was done to improve speed/through-put, and is where I would improve the design.

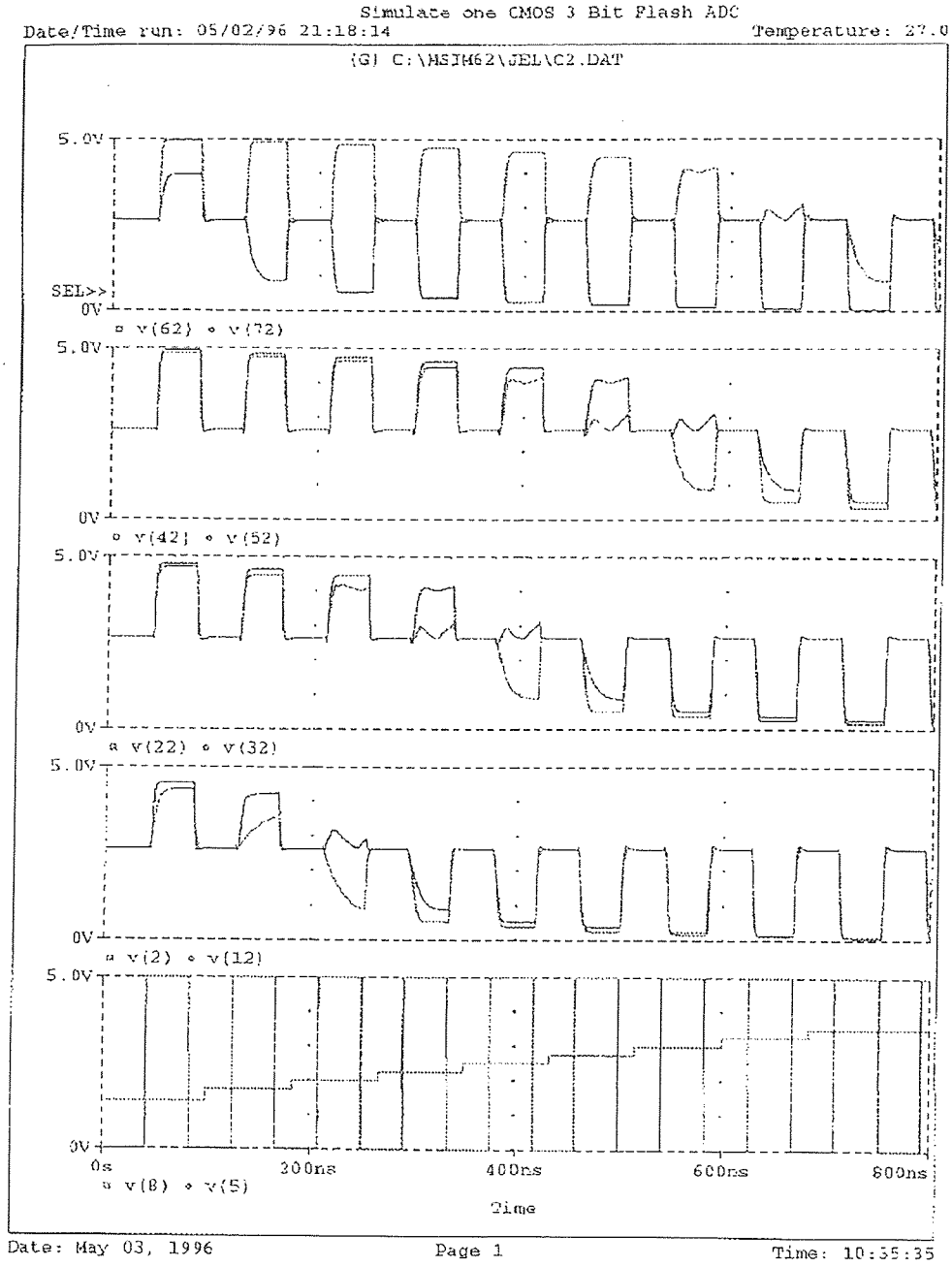

Figure 6

SPICE Simulation - Comparator Outputs

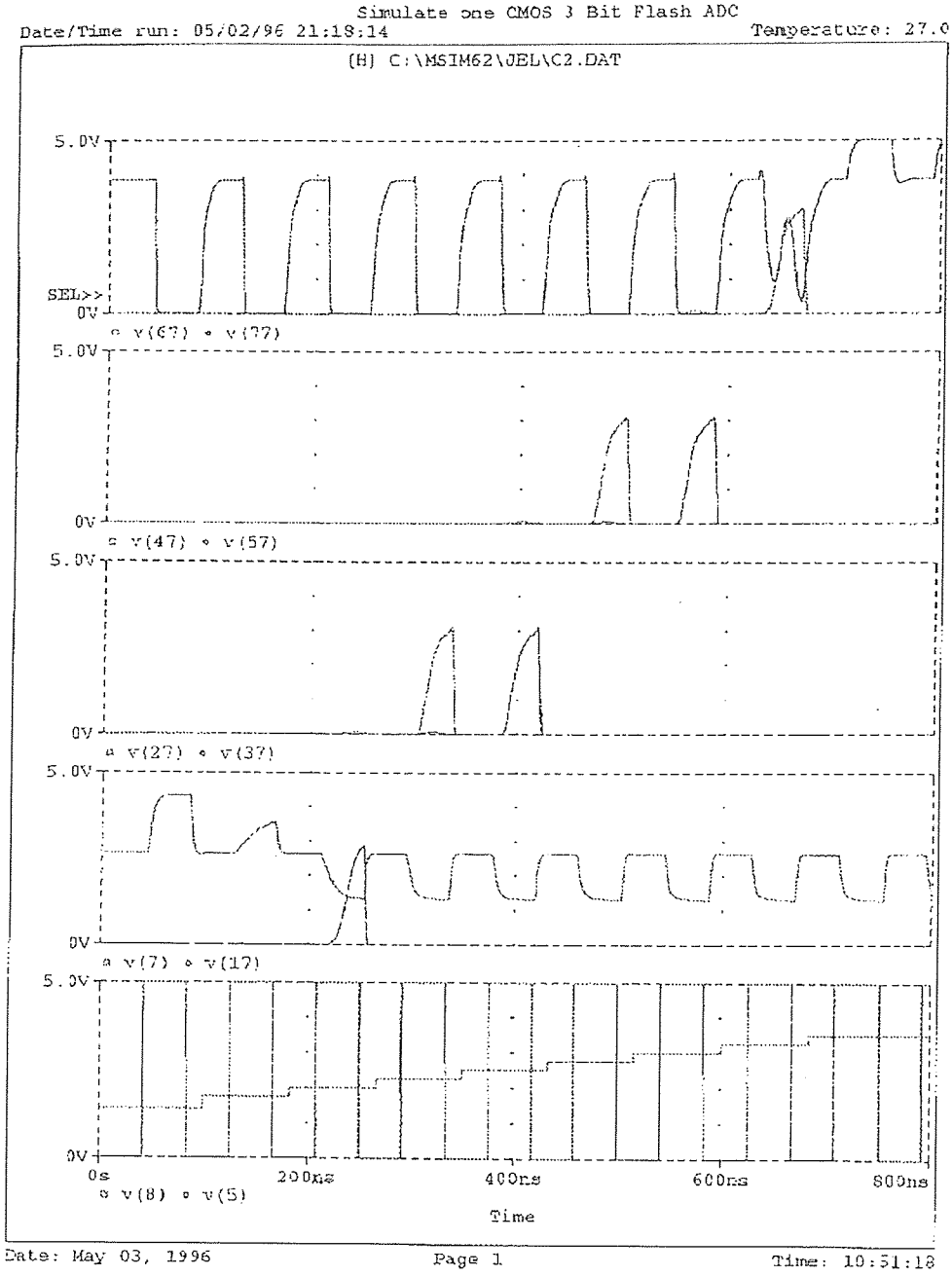

Figure 7

SPICE Simulation - 'Bar to Dot' Converter Outputs

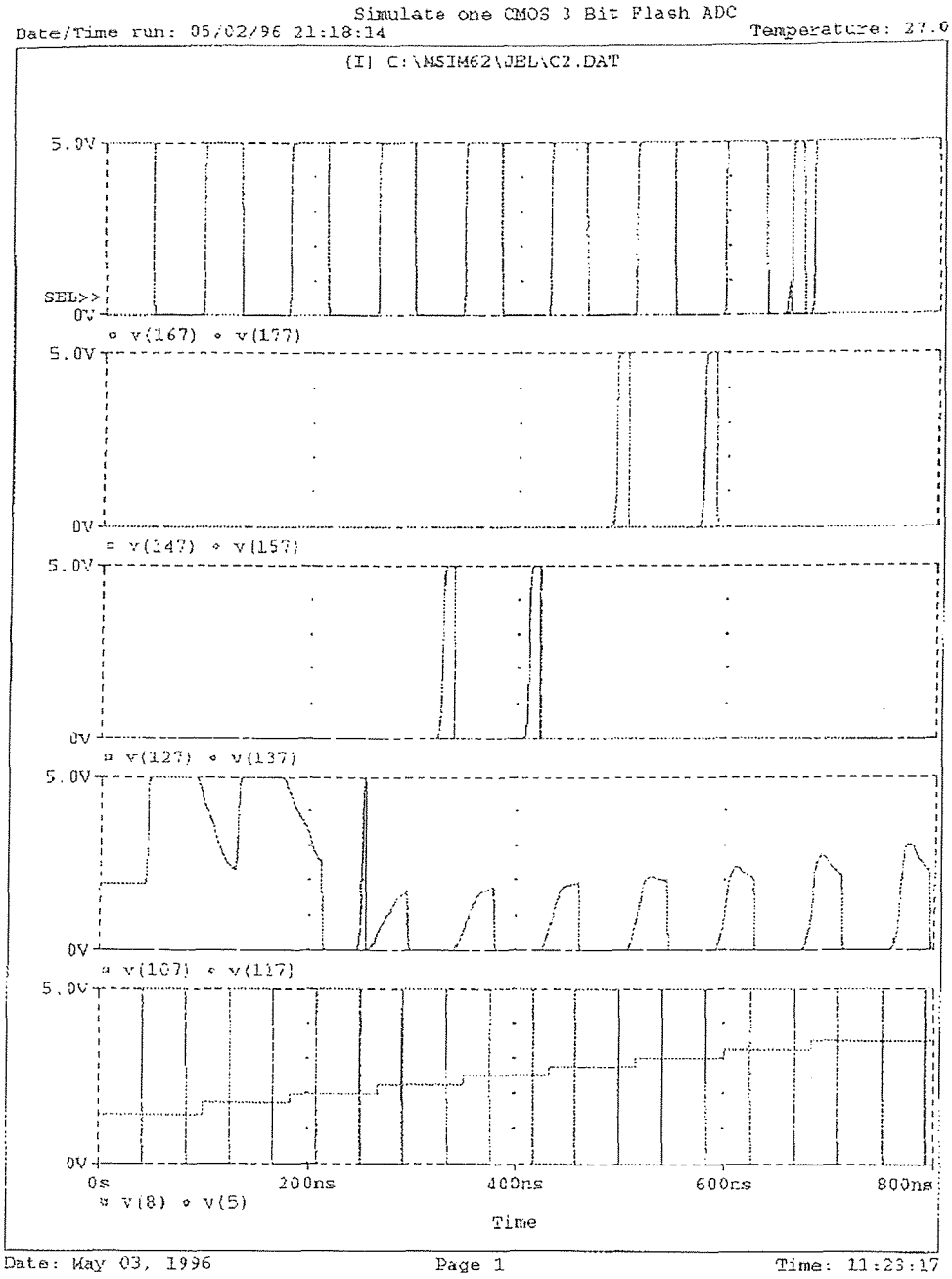

Figure 8

SPICE Simulation - Input to the Encoder ROM

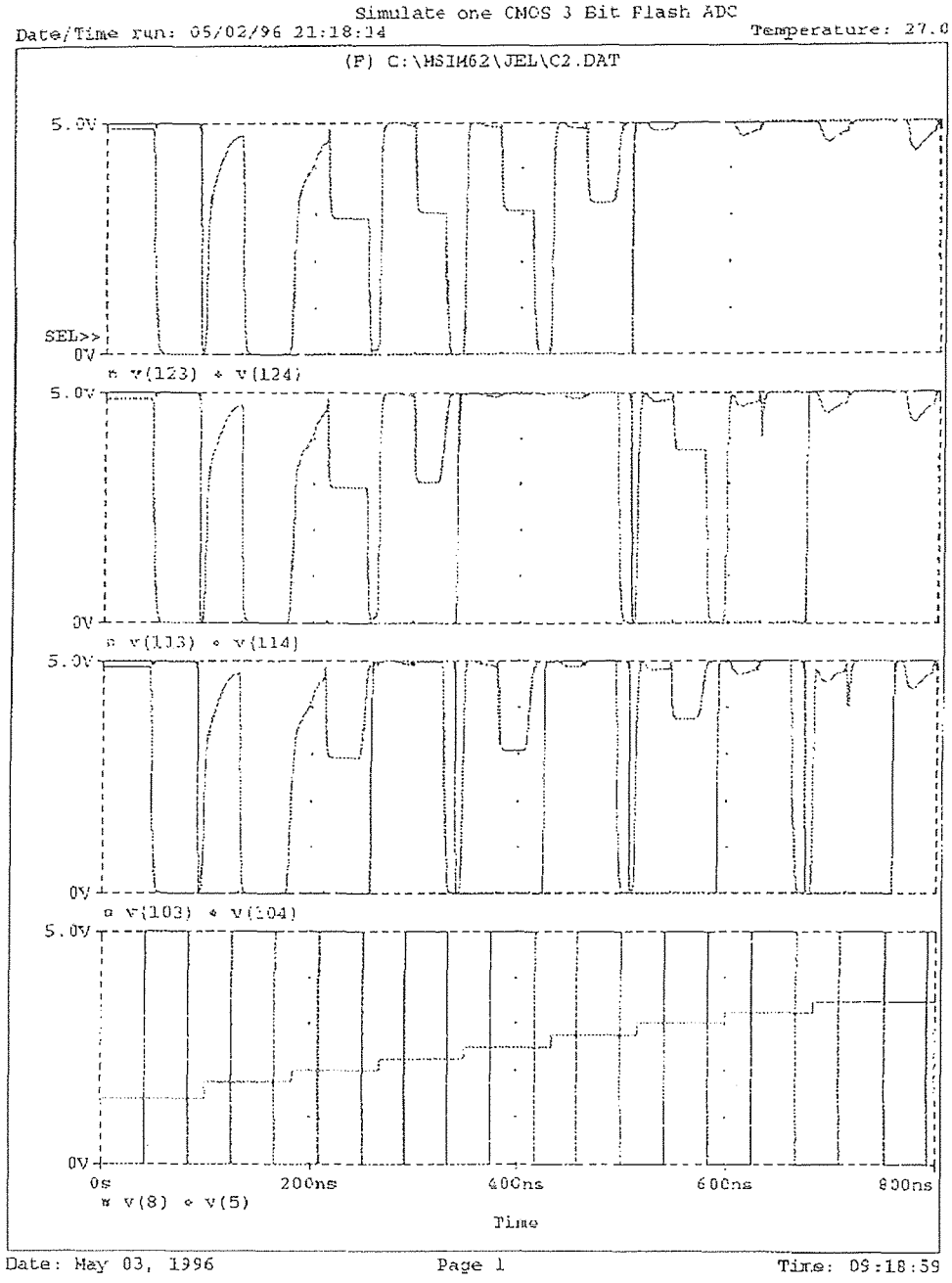

Figure 9

SPICE Simulation - Output From ROM and Latches

CHAPTER 5
PHYSICAL LAYOUT AND PROCESS DISCUSSION

5. 1 Mentor Graphics and Hierarchy

The actual layout for the chip was done using Mentor-Graphics version 7 on HP Apollo workstations. Chipgraph was used for doing the polygon editing. The design utilized a simple comparator. This comparator (Figure 10) was then built into a set of four comparators with the appropriate clocking circuitry and a section of the resistive ladder (Figure 11). The set of four comparators was placed side by side to create a set of 64 comparators. The outputs of each of these comparators was fed into a pass transistor circuit to only have one high and the other circuits low. There were 64 of these circuits that were placed next to the set of 64 comparators. The encoding ROM was set up to be mask programmed on the metal contact layer (Figure 12). A generic row was created without the contact layer. The row included the driving circuitry. This circuit was repeated again 64 times and then the contact layer was added. The output from the encoding ROM was fed into 6 latches (Figure 13) that were also individual cells. As a side note, I have found that the final assembly of a full custom chip takes at least as long as doing the 'inner' workings of the device. A bulk of time was spent designing the comparator to take as little space as possible. The other bulk of time was spent to assemble the chip with bonding pads and general routing of clocking and power lines. From the chipgraph data the design was 'flattened' to remove subcells prior to creating the files to send to MOSIS for fabrication. The bonding pad layout is shown in Figure 14 and Figure 15 and the entire chip is shown in

Figure 16. The large transistors in Figure 15 have a transconductance per area of $6.4x10^{-6}$ μ-mhos/μm^2 for the NMOS transistor and $2.56x10^{-6}$ μ-mhos/μm^2 for the NMOS transistor

5. 2 Mosis Acceptance

The files that MOSIS would accept were either CALMA II files or CIF (California Intermediate Form) files. The CALMA II files were binary files. The CIF files were text, which was the type that was easiest to send over e-mail. The end file occupied approximately 4 Megabytes of characters. I was only permitted to send MOSIS files of 900,000 characters or less. I wrote a C program to split the file up into 5 different CIF sections for MOSIS. A copy of the program is given in appendix B. Once the design passed the few quick tests that MOSIS performed on the source deck, it was up to them to fabricate the device. This device fit into a category that MOSIS called a 'Tiny Chip', a maximum size of 2.25mm by 2.25mm. The fabrication costs were dramatically reduced and the turn-around time was 6 to 8 weeks. The devices were placed into standard 28 pin packages and a bonding diagram was supplied along with the parameters from the processing. From the processing the design rules are created. The design rules tell how close various layers may be to each other; the closeness of transistors; the overlap to ensure connections between layers; and the minimum width and length of certain layers. The following is a brief summary of how the various mask layers are aligned and the order of processing.

5. 3 Design Rules

The layout design rules are provided by MOSIS. They are an adaptation on the CMOS lambda design rules presented in *Principles of CMOS VLSI Design A Systems Perspective* by N. Weste and K. Eshraghian. These rules determine the minimum widths of features such as the minimum width of an active area, P-wells, metalization and poly-silicon layers. The rules also determine the exact size of a contact and via from one layer to another. The minimum spacing between different and similar layers is also determined; also the overlap to assure a contact/via connection between layers. Many factors determine these rules. The first and most obvious factor is how well a mask can be aligned over a preceding layer; this is partly determined by the optics used in an alignment tool and also how the photo-lithography is actually done. Another factor is how the etching of the glass/silicon-dioxide is done and how well it is controlled. The process of etching various layers can cause under-cut or over-cut of material. The amount of doping of the silicon in the various layers also will provide a measure of how susceptible the device is to latchup, which gives the spacing between P-wells and active areas.

5. 4 Processing Parameters

Processing determines the parameters for modeling the device using computer simulations. These models are partly determined experimentally and partly on certain intrinsic properties. Each wafer contains sets of test structures. These structures include various ring oscillators, resistors made from different layers, capacitors made

again from different layers. These parameters must be know to a certain extent prior to the actual design of the chip to ensure a reasonable chance for success. MOSIS provides a summary of these parameters for each of their various production runs. This information then can be used to more accurately simulate the device and verify its properties.

5. 5 Production of a Chip

The first step/layer in processing the silicon wafer is to create the P-wells that contain the NMOS transistors. This step is accomplished by using a mask pattern on the bare silicon wafer.

The second step/layer in processing is to create the active areas where transistors are to be placed. This is accomplished by aligning the mask for the active areas to the P-well areas. The active areas are actually areas of silicon dioxide about 410 Angstroms thick.

The third step/layer is the creation of poly-silicon gates for both N-MOS and P-MOS transistors. This mask layer is aligned to the active areas.

The fourth and fifth steps/layers in processing use the same mask layer. This layer is called the P-select area, but its complement is actually the N-select area. These mask layers are aligned with the active areas.

The sixth step/layer is the second layer of poly-silicon. This layer is aligned with the first layer of poly-silicon.

The seventh step/layer is a series of cuts through the insulating layers of silicon dioxide. These enable the next layer (metal 1) to contact and connect other layers to each other. This layer is aligned with the active areas.

The eighth step/layer is the first level of metalization. This provides interconnects. This layer is aligned with the contact layer.

The ninth step/layer is again a set of cuts through the insulating silicon dioxide, but only down to the first level of metal. This provides a connection between the first and second levels of metal. This is aligned with the first level of metal.

The tenth step/layer is the second layer of metal. This only connects metal one to metal two or provides bonding pads to the outside world. This layer is aligned with the via layer (number ten).

The eleventh step/layer is called passivation. This provides protection to the chip. Holes are cut in this layer to gain access to bonding pads on the second level metal. This layer is aligned with the second level metal layer[7]

The layout and processing shown in Figure 17 and Figure 18 shows the various steps for a single level metalization and single level poly-silicon[8]. To create the processing need for this thesis the processing and layers for the poly-silicon and second level metal and various contact between these layers would have to be added to the steps.

---

[7]Eshraghian, Kamran & Weste, Neil, Oct. 1985,"CMOS Processing Technology", *Principles of CMOS VLSI Design A Systems Perspective*, 1st Ed., VLSI System Series, Addison-Wesley Publishing Company, Reading Ma., pp 63 to 118

[8] Eshraghian, Kamran & Weste, Neil, Oct. 1985,"CMOS Processing Technology", *Principles of CMOS VLSI Design A Systems Perspective*, 1st Ed., VLSI System Series, Addison-Wesley Publishing Company, Reading Ma., pp 72 to 73

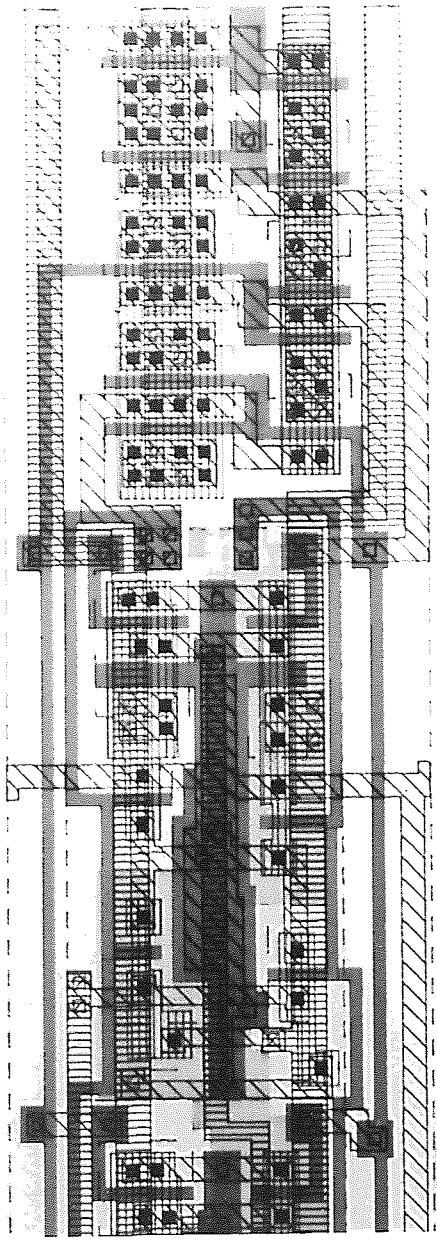

Figure 10

Layout of the Clocked Comparator Cell and Clock Driver

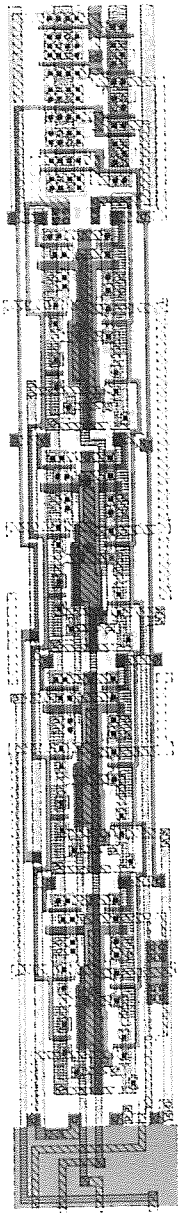

Figure 11

Layout of Four Comparators with Clocking Logic and a Portion of the Resistor Divider

Figure 12

Layout of the Encoding ROM

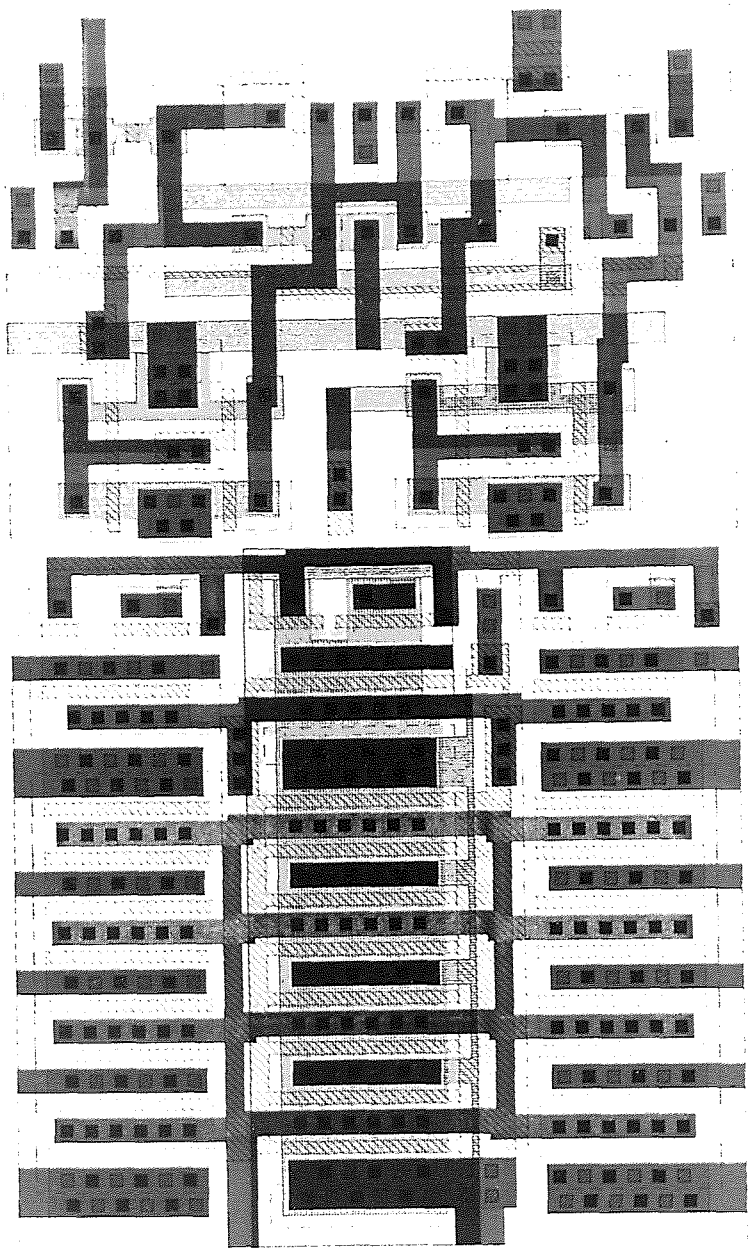

Figure 13

Layout of a Latch

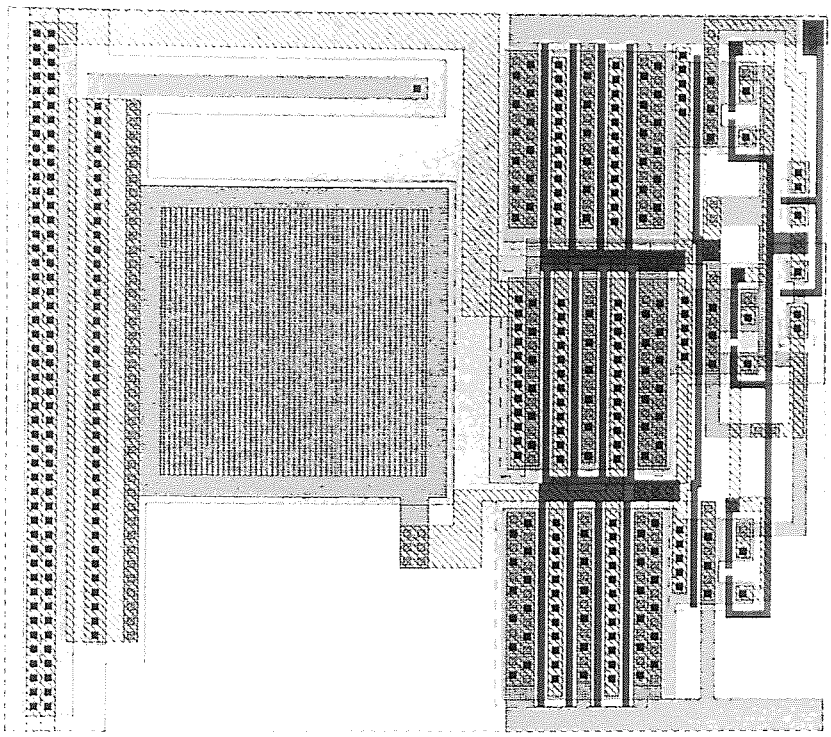

Figure 14

Layout of a Bonding Pad

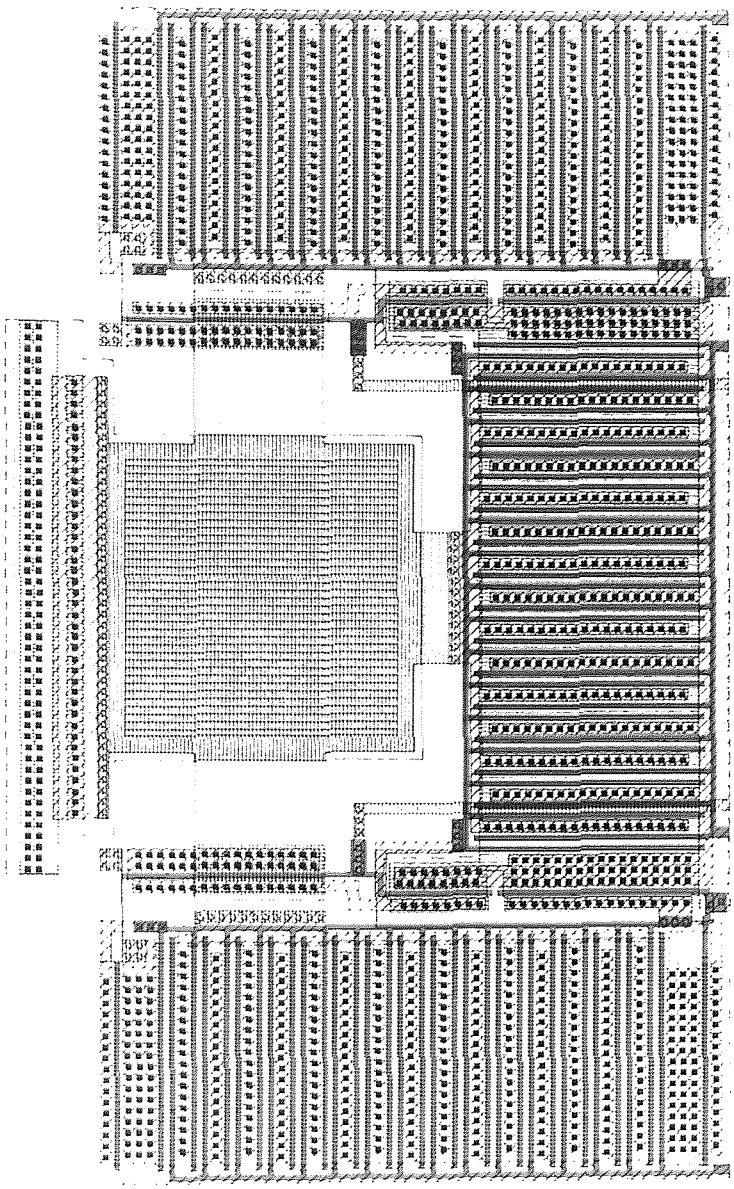

Figure 15

Layout of a Tri-State Bonding Pad

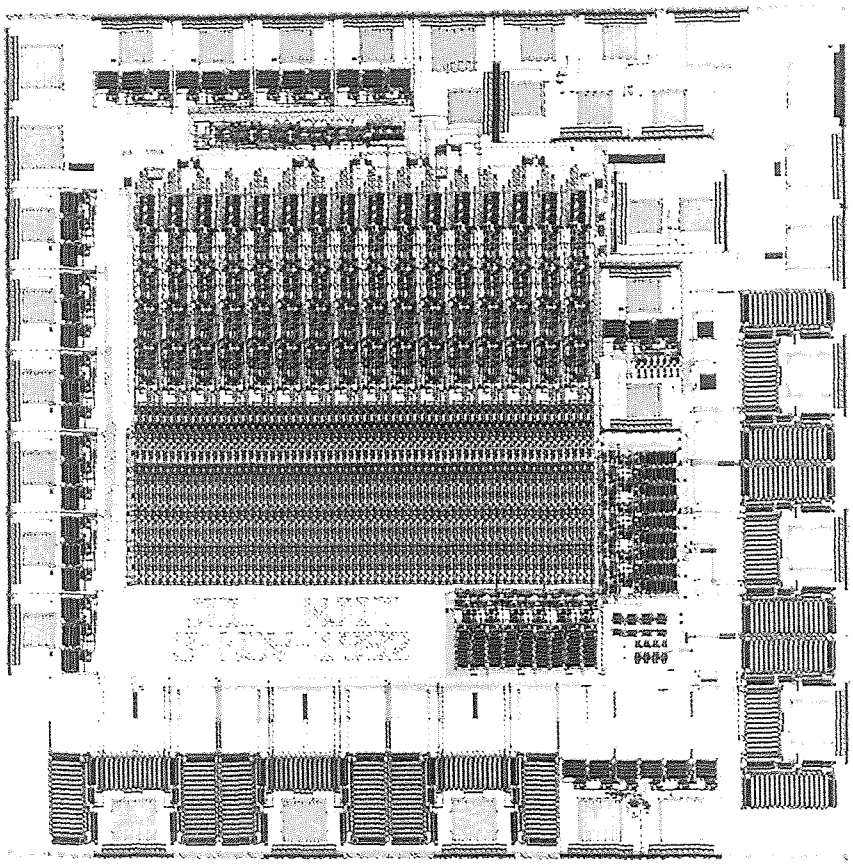

Figure 16

Layout of the Entire 6 Bit A/D Tiny Chip

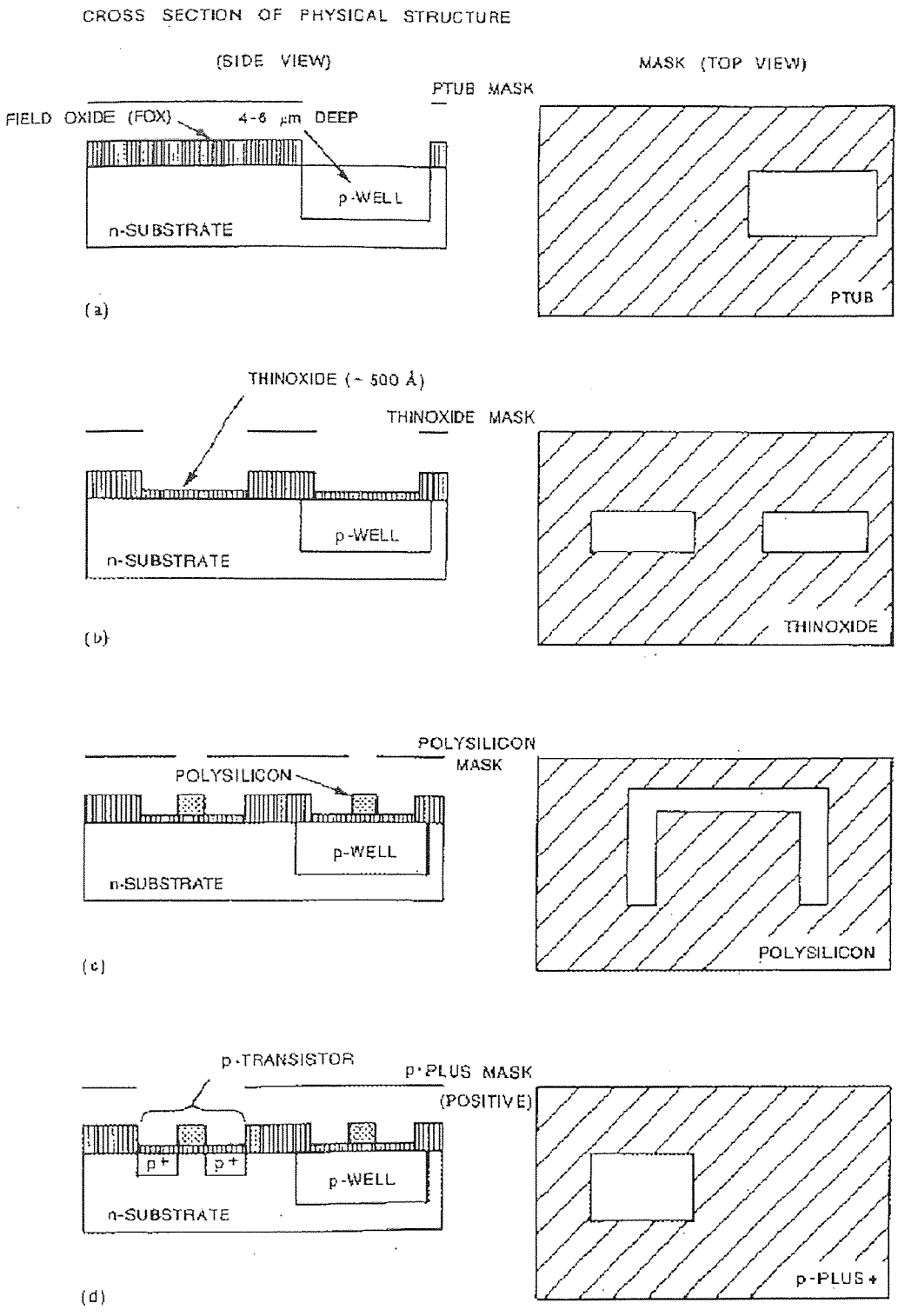

Figure 17

Processing and Mask Layers

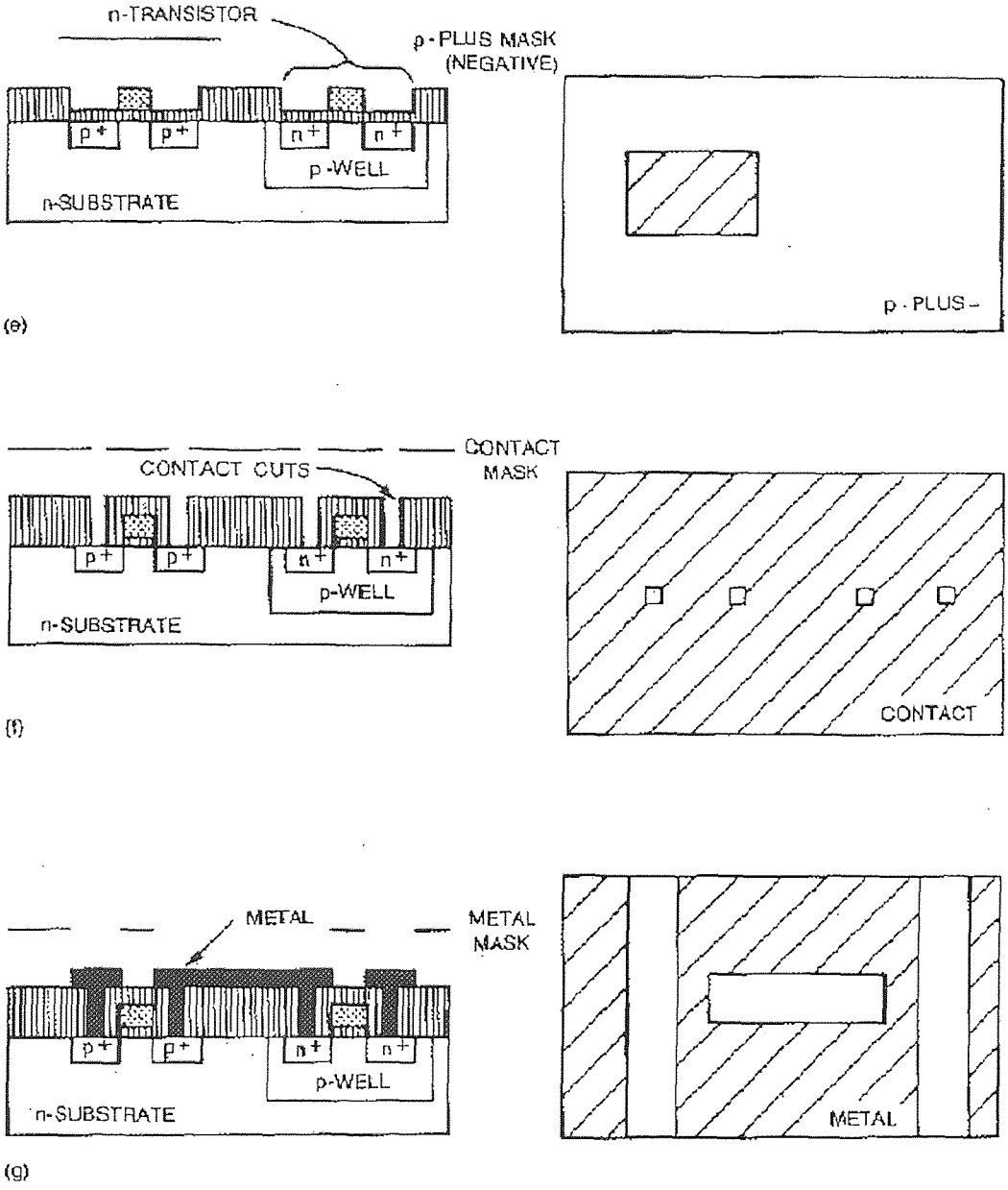

Figure 18

Processing and Mask Layers

CHAPTER 6
DEVICE ENGINEERING TEST

6. 1 Device Testing

Four devices were fabricated by MOSIS on thesis N41Q run during March of 1994. Physical examination, under a microscope, of the four devices did not reveal any blatant fabrication errors that would have prevented the devices from functioning properly. There appeared to be a small defect on one of the power supply metal layers that was either caused by dust or an imperfection in the layer of glass below it. This caused a different texture in one small spot of the run. The resistor voltage divider is 32μm by 1152μm that yields 36 squares. The sheet resistance of the poly-silicon one layer was reported by MOSIS at 22.3 ohms per square, this would make the divider be 802.8Ω. The actual measurement of the divider showed it to be 915Ω. The difference could be attributed to contact resistance combined with the resistance of the bonding wires. The percentage error is just below 14 percent. The devices were then powered with a 5 Volt power supply. Each one drew about 150 mA to 175 mA. The Clock line and the two chip-enable lines were toggled to obtain results. The output results showed that the four devices performed the same, the outputs were low and could not be forced to a valid logic hi. The highest voltage that could be obtained from an output pin was in the range from 0.35 to 0.45 Volts. This would indicate a short internally of some kind. Since all four devices behaved in the same manner, this is believed to be a design flaw in the layout. The micro photograph of the device is shown in Figure 19. The actual

parameters that were derived from the MOSIS run are provided in Appendix C. These parameters were then used to re-run the PSPICE simulation.

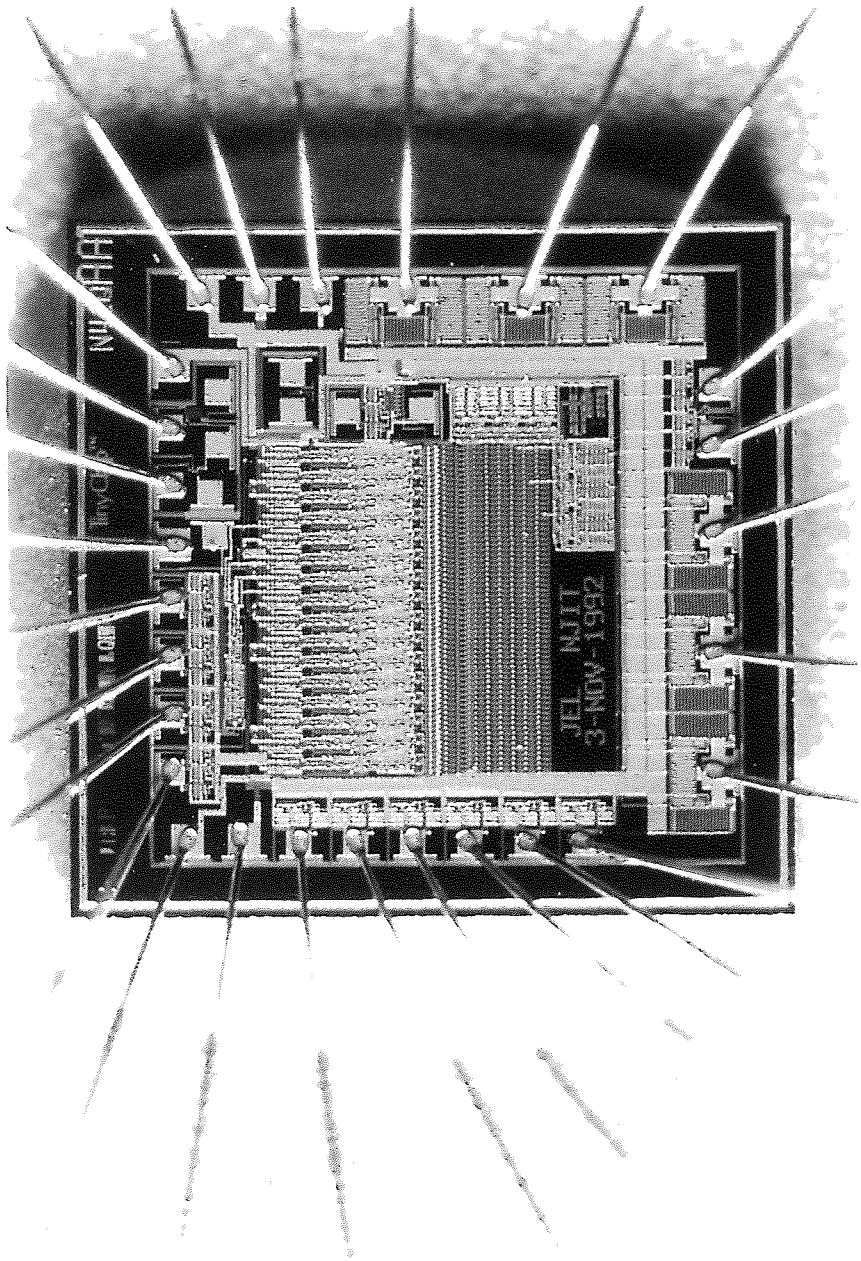

Figure 19

Micro-Photograph of the Actual Tiny Chip

CHAPTER 7
SUMMARY AND CONCLUSIONS

This exercise in designing, simulating, laying out, building and testing of a flash analog-to-digital converter had some limited success. The design and simulation were done without problems. The final simulation did show where improvements could be made for the future. The layout was done using chipgraph from Mentor Graphics version 7, but other design verification tools were not available such as an electrical and design rule checkers. These tools would have aided in the design process by back annotation of the device to simulate it fully, and to have caught any mistakes, either in electrical or design rule errors prior to the actual build. The building of the device was done by MOSIS in California, which yielded four devices to test. The testing revealed that there was a mistake in the layout, but it is undetermined where that mistake lies. This radically different comparator design is in use commercially by Datel and Harris. The Datel devices are the ADC-207 and ADC-208, being a 7 and an 8 bit flash analog-to-digital converters respectively. The Datel device utilizes the improvement that was discovered during a larger simulation using PSPICE. The Harris devices (CA3306 family), also being 6 bit devices, offer similar performance to the device presented in this thesis.

The device presented here is a commercial reality. The design of the comparator utilizing a radically different approach can be taken and does yield better performance due to the absolute matching of transistors as in a traditional differential

transistor design. The performance for a flash analog-to-digital converter rivals other designs available today. There exists no reason that the same comparator could not be used for other types of converters or other applications where a comparator is required as long as a clocked device is acceptable.

APPENDIX A

SPICE SOURCE DECK

The following is a copy of the SPICE program used to run the simulation a 3 bit ADC using PSPICE.

```
            Simulate one CMOS 3 Bit Flash ADC

..MODEL CMOSN NMOS LEVEL=2 PHI=0.600000 TOX=4.2100E-08 XJ=0.200000U
+ VTO=0.9597 DELTA=8.5280E+00 LD=1.8740E-07 KP=4.7466E-05 TPG=1
+ UO=578.7 UEXP=1.8310E-01 UCRIT=8.8450E+04 RSH=1.7240E+01
+ GAMMA=1.0380 NSUB=2.1850E+16 NFS=3.91E+11 VMAX=6.6700E+04
+ LAMBDA=2.4990E-02 CGDO=2.3057E-10 CGSO=2.3057E-10
+ CGBO=4.0753E-10 CJ=3.7086E-04 MJ=0.4303 CJSW=5.6047E-10
+ MJSW=0.384091 PB=0.800000
* Weff = Wdrawn - Delta_W
* the suggested Delta_W is -3.8800E-07

.MODEL CMOSP PMOS LEVEL=2 PHI=0.600000 TOX=4.2100E-08 XJ=0.200000U
+ VTO=-0.6892 DELTA=1.7970E+00 LD=1.5070E-07 KP=1.9218E-05 TPG=-1
+ UO=234.3 UEXP=3.3320E-01 UCRIT=6.6300E+04 RSH=6.6000E+01
+ GAMMA=0.5915 NSUB=7.0900E+15 NFS=3.23E+11 VMAX=9.9990E+05
+ LAMBDA=4.9200E-02 CGDO=1.8541E-10 CGSO=1.8541E-10
+ CGBO=3.4681E-10 CJ=1.6874E-04 MJ=0.4794 CJSW=1.5082E-10
+ MJSW=0.07452 PB=0.700000
* Weff = Wdrawn - Delta_W
* the suggested Delta_W is 6.0720E-09

* Vcc Gnd Nmos I Pmos I Pmos F Nmos F Vref Vin Vout
.subckt comp 1 11 4 5 9 10 8 7 2

MTA0   3   9   2   1 CMOSP (L=2U W=6U  AS=36P PS=48U AD=36P PD=48U)
MTA1   2   3   1   1 CMOSP (L=4U W=12U AS=72P PS=96U AD=72P PD=96U)
MTA2   2   3 11  11 CMOSN (L=4U W=6U  AS=36P PS=48U AD=36P PD=48U)
MTA3   2 10   3  11 CMOSN (L=2U W=3U  AS=18P PS=24U AD=18P PD=24U)
CFB   3 6 .2E-12
MTA4   7   4   6  11 CMOSN (L=2U W=3U AS=18P PS=24U AD=18P PD=24U)
MTA5   6   5   7   1 CMOSP (L=2U W=6U AS=36P PS=48U AD=36P PD=48U)
MTA6   6   4   8   1 CMOSP (L=2U W=6U AS=36P PS=48U AD=36P PD=48U)
MTA7   8   5   6  11 CMOSN (L=2U W=3U AS=18P PS=24U AD=18P PD=24U)

.ends

* Vcc Gnd Clock Vref+ Vref- Vin Vout_1 Vout_2 Vout_3 Vout_4
.subckt comp_4 1 11 3 7 8 13 14 15 16 17

*Node 2 is Nmos F
MTbf1    2   3   1   1 CMOSP (L=2U W=24U AS=36P PS=48U AD=36P PD=48U)
```

```
MTbf2   2  3 11 11 CMOSN (L=2U W=12U AS=36P PS=48U AD=36P PD=48U)

*Node 4 is Pmos F
MTaf1   4  2  1  1 CMOSP (L=2U W=24U AS=36P PS=48U AD=36P PD=48U)
MTaf2   4  2 11 11 CMOSN (L=2U W=12U AS=36P PS=48U AD=36P PD=48U)

*Node 5 is Nmos I
MTai1   5  2  1  1 CMOSP (L=2U W=24U AS=36P PS=48U AD=36P PD=48U)
MTai2   5  2 11 11 CMOSN (L=2U W=24U AS=36P PS=48U AD=36P PD=48U)

*Node 4 is Pmos F
MTbi1   6  4  1  1 CMOSP (L=2U W=48U AS=72P PS=96U AD=72P PD=96U)
MTbi2   6  4 11 11 CMOSN (L=2U W=12U AS=36P PS=48U AD=36P PD=48U)

R0  7 18  6.25
R1 18  9 12.5
R2  9 10 12.5
R3 10 19 12.5
R4 19  8  6.25

* Vcc Gnd Nmos I Pmos I Pmos F Nmos F Vref       Vin Vout

X1 1 11 5 6 4 2 18 13 14 comp
X2 1 11 5 6 4 2  9 13 15 comp
X3 1 11 5 6 4 2 10 13 16 comp
X4 1 11 5 6 4 2 19 13 17 comp

.ends

*D1 D0 Gnd  Vcc Out
.subckt bar_dot 1 2 3 5 4

Mb1  4  2  3  3 CMOSN (L=2U W=3U AS=18P PS=24U AD=18P PD=24U)
Mb2  1  2  4  5 CMOSP (L=2U W=6U AS=36P PS=48U AD=36P PD=48U)

.ends

*Din Gnd  Vcc Out
.subckt Inv 1 2 3 4

Mb1  4  1  2  2 CMOSN (L=2U W=6U  AS=36P PS=48U AD=36P PD=48U)
Mb2  3  1  4  3 CMOSP (L=2U W=12U AS=72P PS=96U AD=72P PD=96U)

.ends

*Clock Din Vcc Gnd Dout
.subckt Latch 1 2 3 4 5

Mlatch1   2 12  6 3 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mlatch2   2 11  6 4 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mlatch3   6 11  8 3 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
```

```
Mlatch4    6 12   8 4 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mlatch5    3  6   7 3 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mlatch6    7  6   4 4 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mlatch7    3  7   8 3 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mlatch8    8  7   4 4 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mlatch11   8 11   9 3 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mlatch12   8 12   9 4 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mlatch13   9 12   5 3 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mlatch14   9 11   5 4 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mlatch15   3  9  10 3 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mlatch16  10  9   4 4 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mlatch17   3 10   5 3 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mlatch18   5 10   4 4 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mlatch20   3  1  11 3 CMOSP (W=12U L=2U AS=144P PS=72U AD=144P PD=72U)
Mlatch21  11  1   4 4 CMOSN (W=6U  L=2U AS=72P  PS=36U AD=72P  PD=36U)

Mlatch22   3 11  12 3 CMOSP (W=12U L=2U AS=144P PS=72U AD=144P PD=72U)
Mlatch23  12 11   4 4 CMOSN (W=6U  L=2U AS=72P  PS=36U AD=72P  PD=36U)

.ends

*In Vcc Gnd Out*
.subckt Inv_4 1 2 3 4

Minv4_1    2  1   4 2 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Minv4_2    4  1   3 3 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

.ends

* Vcc Gnd Clock Vref+ Vref- Vin Vout_1 Vout_2 Vout_3 Vout_4
Xc1 1 0 5 6 3 8 72  2 12 22 comp_4
Xc2 1 0 5 3 4 8 32 42 52 62 comp_4

*D1 D0 Gnd  Vcc Out
Xb0  2  0 0 1  7 bar_dot
Xb1 12  2 0 1 17 bar_dot
Xb2 22 12 0 1 27 bar_dot
Xb3 32 22 0 1 37 bar_dot
Xb4 42 32 0 1 47 bar_dot
Xb5 52 42 0 1 57 bar_dot
Xb6 62 52 0 1 67 bar_dot
Xb7  1 62 0 1 77 bar_dot

*Din Gnd  Vcc Out
Xr0  7 0 1 102 Inv
```

```
Xr1 17 0 1 112 Inv
Xr2 27 0 1 122 Inv
Xr3 37 0 1 132 Inv
Xr4 47 0 1 142 Inv
Xr5 57 0 1 152 Inv
Xr6 67 0 1 162 Inv
Xr7 77 0 1 172 Inv

Xri0 102 0 1 107 Inv
Xri1 112 0 1 117 Inv
Xri2 122 0 1 127 Inv
Xri3 132 0 1 137 Inv
Xri4 142 0 1 147 Inv
Xri5 152 0 1 157 Inv
Xri6 162 0 1 167 Inv
Xri7 172 0 1 177 Inv

*Encoder Rom
Mrom0_0 103 107   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)
Mrom0_1 113 107   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)
Mrom0_2 123 107   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mrom1_0   1 112 103 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mrom1_1 113 117   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)
Mrom1_2 123 117   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mrom2_0 103 127   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)
Mrom2_1   1 122 113 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mrom2_2 123 127   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mrom3_0   1 132 103 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mrom3_1   1 132 113 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mrom3_2 123 137   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)

Mrom4_0 103 147   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)
Mrom4_1 113 147   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)
Mrom4_2   1 142 123 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)

Mrom5_0   1 152 103 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mrom5_1 113 157   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)
Mrom5_2   1 152 123 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)

Mrom6_0 103 167   0 0 CMOSN (W=3U L=2U AS=36P PS=18U AD=36P PD=18U)
Mrom6_1   1 162 113 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mrom6_2   1 162 123 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)

Mrom7_0   1 172 103 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mrom7_1   1 172 113 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)
Mrom7_2   1 172 123 1 CMOSP (W=6U L=2U AS=72P PS=36U AD=72P PD=36U)


*Clock Din Vcc Gnd Dout
```

```
Xout0 91 103 1 0 104 Latch
Xout1 91 113 1 0 114 Latch
Xout2 91 123 1 0 124 Latch


*In Vcc Gnd Out*
Xinv1  5 1 0 92 Inv_4
Xinv2 92 1 0 93 Inv_4
Xinv3 93 1 0 91 Inv_4
Cdelay 92 0 .1P


Vref1 4 0 DC 3.5
Vref2 6 0 DC 1.5

VCC 1 0 DC 5

*2 Mhz
*VCLOCK1  5 0 PULSE(0 5 250.N .05N .05N 250.N 500.N)

*Vin1 18  0 PULSE(1.4 1.75  600.N .01N .01N 6400.N 7000.N)
*Vin2 28 18 PULSE(  0 0.25 1100.N .01N .01N 5900.N 7000.N)
*Vin3 38 28 PULSE(  0 0.25 1600.N .01N .01N 5400.N 7000.N)
*Vin4 48 38 PULSE(  0 0.25 2100.N .01N .01N 4900.N 7000.N)
*Vin5 58 48 PULSE(  0 0.25 2600.N .01N .01N 4400.N 7000.N)
*Vin6 68 58 PULSE(  0 0.25 3100.N .01N .01N 3900.N 7000.N)
*Vin7 78 68 PULSE(  0 0.25 3600.N .01N .01N 3400.N 7000.N)
*Vin8  8 78 PULSE(  0 0.25 4100.N .01N .01N 2900.N 7000.N)
*.TRAN .1N 0.0048M

*4Mhz
*VCLOCK1  5 0 PULSE(0 5 125.N .05N .05N 125.N 250.N)

*Vin1 18  0 PULSE(1.4 1.75  300.N .01N .01N 3200.N 3500.N)
*Vin2 28 18 PULSE(  0 0.25  550.N .01N .01N 2950.N 3500.N)
*Vin3 38 28 PULSE(  0 0.25  800.N .01N .01N 2700.N 3500.N)
*Vin4 48 38 PULSE(  0 0.25 1050.N .01N .01N 2450.N 3500.N)
*Vin5 58 48 PULSE(  0 0.25 1300.N .01N .01N 2200.N 3500.N)
*Vin6 68 58 PULSE(  0 0.25 1550.N .01N .01N 1950.N 3500.N)
*Vin7 78 68 PULSE(  0 0.25 1800.N .01N .01N 1700.N 3500.N)
*Vin8  8 78 PULSE(  0 0.25 2050.N .01N .01N 1450.N 3500.N)
*.TRAN .1N 0.0024M

*8Mhz
*VCLOCK1  5 0 PULSE(0 5 62.5N .05N .05N 62.5N 125.N)

*Vin1 18  0 PULSE(1.4 1.75  150.N .01N .01N 1600.N 1750.N)
*Vin2 28 18 PULSE(  0 0.25  275.N .01N .01N 1475.N 1750.N)
*Vin3 38 28 PULSE(  0 0.25  400.N .01N .01N 1350.N 1750.N)
*Vin4 48 38 PULSE(  0 0.25  525.N .01N .01N 1225.N 1750.N)
*Vin5 58 48 PULSE(  0 0.25  650.N .01N .01N 1100.N 1750.N)
```

```
*Vin6 68 58 PULSE(  0 0.25  725.N .01N .01N  925.N 1750.N)
*Vin7 78 68 PULSE(  0 0.25  900.N .01N .01N  850.N 1750.N)
*Vin8  8 78 PULSE(  0 0.25 1025.N .01N .01N  725.N 1750.N)
*.TRAN .1N .00012M

*10 Mhz
*VCLOCK1  5 0 PULSE(0 5 50.0N .05N .05N 50.0N 100.0N)

*Vin1 18  0 PULSE(1.4 1.75 120.0N .01N .01N 1280.0N 1400.0N)
*Vin2 28 18 PULSE(  0 0.25 220.0N .01N .01N 1180.0N 1400.0N)
*Vin3 38 28 PULSE(  0 0.25 320.0N .01N .01N 1080.0N 1400.0N)
*Vin4 48 38 PULSE(  0 0.25 420.0N .01N .01N  980.0N 1400.0N)
*Vin5 58 48 PULSE(  0 0.25 520.0N .01N .01N  880.0N 1400.0N)
*Vin6 68 58 PULSE(  0 0.25 620.0N .01N .01N  780.0N 1400.0N)
*Vin7 78 68 PULSE(  0 0.25 720.0N .01N .01N  680.0N 1400.0N)
*Vin8  8 78 PULSE(  0 0.25 820.0N .01N .01N  580.0N 1400.0N)
*.TRAN .1N .00100M

*12Mhz
VCLOCK1  5 0 PULSE(0 5 41.667N .05N .05N 41.667N 83.333N)

Vin1 18  0 PULSE(1.4 1.75 100.000N .01N .01N 1066.667N 1166.667N)
Vin2 28 18 PULSE(  0 0.25 183.333N .01N .01N  983.334N 1166.667N)
Vin3 38 28 PULSE(  0 0.25 266.667N .01N .01N  900.000N 1166.667N)
Vin4 48 38 PULSE(  0 0.25 350.000N .01N .01N  816.667N 1166.667N)
Vin5 58 48 PULSE(  0 0.25 433.333N .01N .01N  733.334N 1166.667N)
Vin6 68 58 PULSE(  0 0.25 516.667N .01N .01N  650.000N 1166.667N)
Vin7 78 68 PULSE(  0 0.25 600.000N .01N .01N  566.667N 1166.667N)
Vin8  8 78 PULSE(  0 0.25 683.333N .01N .01N  483.334N 1166.667N)
.TRAN .1N 0.0008M

*16Mhz
*VCLOCK1  5 0 PULSE(0 5 31.25N .05N .05N 31.25N 62.5N)

*Vin1 18  0 PULSE(1.4 1.75   75.0N .01N .01N 800.0N 875.N)
*Vin2 28 18 PULSE(  0 0.25  137.5N .01N .01N 737.5N 875.N)
*Vin3 38 28 PULSE(  0 0.25  200.0N .01N .01N 675.0N 875.N)
*Vin4 48 38 PULSE(  0 0.25  262.5N .01N .01N 612.5N 875.N)
*Vin5 58 48 PULSE(  0 0.25  325.0N .01N .01N 550.0N 875.N)
*Vin6 68 58 PULSE(  0 0.25  412.5N .01N .01N 462.5N 875.N)
*Vin7 78 68 PULSE(  0 0.25  450.0N .01N .01N 425.0N 875.N)
*Vin8  8 78 PULSE(  0 0.25  512.5N .01N .01N 362.5N 875.N)
*.TRAN .1N .0006M

* LIMPTS=200000
.OPTIONS ITL5=0 reltol=.00001 VNTOL=.5U CHGTOL=.005P ABSTOL=.5P
.OPTIONS ACCT EXPAND
.PROBE

.END
```

APPENDIX B

CIF FRAGMENTATION PROGRAM

The following is a copy of the CIF segmentation program used to send the CIF file to MOSIS:

```
/* This program was written in very portable C,      */
/* and tested under VMS 4.2 & 5.0 and 4.3 bsd UNIX. */
/*                                                   */
/* To compile and run this CIF-CHECKSUM program:     */
/*                                                   */
/* FOR VMS and C                                     */
/* $ cc checksum.c                                   */
/* $ link checksum.obj                               */
/* $ define lnk$library sys$library:vaxcrtl.olb      */
/* $ checksum :== $your-disk:[your-full-directory-path]checksum.exe
*/
/* $ checksum your-cif-file                          */
/*                                                   */
/* FOR UNIX and C                                    */
/* % cc -O checksum.c -o checksum                    */
/* % checksum your-cif-file                          */
/*   Modified by Joe E. Levinson                     */
/*          from the mosis program                   */
/*            additions are to split the source file*/
/*            into blocks of approx. 700000 chars    */
/*            also add id and p-password to the      */
/*            files for mosis submission             */
/*                                                   */
/*      THIS PROGRAM IS WRITTEN IN K & R C           */
/*           not ANSI C                              */

#ifdef          VMS
#include        stdio.h
#else
#include        <stdio.h>
#endif

/* standard error routine, print a message then exit */
#define ERROR(str) { printf("%s\n",str); exit(1); }


int main(n, str) int n; char *str[]; {
  int j = 1,       /* number of the file being written       */
      num = 1;     /* number of files to generate            */
  long int i;      /* character count                        */
  char name[15];   /* name of the current output file        */
  int c,           /* character from the current input file  */
```

```
      d;          /* character from the current output file */
  FILE *input,    /* input file                              */
       *tmpfile,  /* temporary file                          */
       *output;   /* output file                            */

/* check input parameters */
    if (n < 2) ERROR("No INPUT file, OUTPUT file, ID or P-PASSWORD
given")
    if (n < 3) ERROR("No OUTPUT file, ID or P-PASSWORD given")
    if (n < 4) ERROR("No ID or P-PASSWORD given")
    if (n < 5) ERROR("No P-PASSWORD given")

/* open input file */
    if ((input = fopen(str[1],"r")) == NULL)
      ERROR("Error opening input file")

/* count number of characters in input file and determine */
/*       how many output files there will be             */
/*        then close input file                          */
    while ((c = fgetc(input)) != EOF)
         { num++; i = 1;
           while ((i++ < 700000L) && ((c = fgetc(input)) != EOF));
           if (c != EOF)
             while (((c = fgetc(input)) != EOF) && (c != '\n'));
           if (c == EOF) break; }
    printf("%d files\n",num-1);
    fclose(input);

/* open input file */
    if ((input = fopen(str[1],"r")) == NULL)
      ERROR("Internal ERROR 1")

/* loop thru each block of the input file */
/*    doing - copy into a temporary file  */
/*          - computing the check sum of  */
/*            the block                   */
/*          - make the correct header for */
/*            the output file and copy    */
/*            the temp file               */
    while ((c = fgetc(input)) != EOF)
         { sprintf(name,"%s%d.cif",str[2],j++);
           i = 1;

           if ((tmpfile = fopen("TMP.tmp","w")) == NULL)
             ERROR("Error opening tmpfile file")
                 fputc(c,tmpfile);
           while ((i++ < 700000L) && ((c = fgetc(input)) != EOF))
                fputc(c,tmpfile);
           if (c != EOF)
                    { while (((c = fgetc(input)) != EOF) && (c !=
'\n'))
                    { i++; fputc(c,tmpfile); }
```

```
            fputc(c,tmpfile); }
          fclose(tmpfile);

          printf("Length = %ld\n",i);
          if ((output = fopen(name,"w")) == NULL)
            ERROR("Error opening output file")

/* write out header to the output file */
          fprintf(output,"REQUEST: SUBMIT\n");
          fprintf(output,"ID: %s\n",str[3]);
          fprintf(output,"P-PASSWORD: %s\n",str[4]);

/* compute the check sum to the output file */
          _compute_cif(output);
          fprintf(output,"CIF-FRAGMENT: %d/%d\n",j-1,num-1);
          if ((tmpfile = fopen("TMP.tmp","r")) == NULL)
            ERROR("Internal Error")

/* copy the temp file to the actual output file */
          while ((d = fgetc(tmpfile)) != EOF)
               fputc(d,output);
          fprintf(output,"REQUEST: END\n");
          fclose(tmpfile); fclose(output);
          if (c == EOF) exit(0);
               }
return 0;
}

#define TRUE 1
#define FALSE 0
static          char mess[100]; /* Buffer for error messages.*/


/*              CKSUM.*/

int _compute_cif(out) FILE *out;
                /* Calculates the CIF-Checksum. */
                /* Reads a CIF file from Standard Input. */
                /* Writes the CIF-Checksum on Standard Output. */
{
static          FILE *in;
long chksum;    /* Checksum accumulates here. */
long nbyte;     /* Byte count accumulates here. */
int sepflg;     /* Boolean for previous separator. */
int c;          /* Gets a character from the file, or EOF. */


                /* Initialization. */

    if ((in = fopen("TMP.tmp","r")) == NULL)
      ERROR("Error opening tmpfile file internal")
```

```
    sepflg = TRUE; /* Set the separator flag. */
    chksum = 32;   /* Initial checksum value. */
    nbyte = 1;     /* Initial byte count. */

                   /* Process the data bytes. */

    while ((c = getc(in)) != EOF)
    {
                c &= 0177; /* Get rid of unwanted bits. */

                if (c > ' ') /* Is this a printing character? */
                {
                    chksum = chksum + (long) c;
                    nbyte++;
                    sepflg = FALSE;
                }

                else if ((c != 0) && (sepflg == FALSE))
                {
                    chksum = chksum + 32L; /* Process the first */
                    nbyte++; /* separator in a row. */
                    sepflg = TRUE;
                }

    }           /* Continue reading file. */

                /* Process the implied trailing separator. */

    if (sepflg == FALSE)
    {
                chksum = chksum + 32L;
                nbyte++;
    }

                /* Print our results. */

    printf ("CIF-checksum = %ld\n", chksum);
    printf ("Byte-count   = %ld\n", nbyte);

    fprintf (out,"CIF-checksum: %ld", chksum);
    fprintf (out," %ld\n", nbyte);

 return ;
}
```

APPENDIX C

MOSIS PROCESS CHARACTERISTICS

The following four pages are a summary of the process sheets provided by MOSIS.

MOSIS PARAMETRIC TEST RESULTS

RUN: N41Q
TECHNOLOGY: SCPE20

VENDOR: ORBIT
FEATURE SIZE: 2.0 microns

INTRODUCTION: This report contains the lot average results obtained by MOSIS from measurements of the MOSIS test structures on each wafer of this fabrication lot. The SPICE parameters obtained from similar measurements on a selected wafer are also attached.

COMMENTS: This looks like a typical Orbit Semiconductor 2.0um p-well run.

TRANSISTOR PARAMETERS	W/L	N-CHANNEL	P-CHANNEL	UNITS
MINIMUM	3/2			
Vth		1.11	-0.75	Volts
SHORT	18/2			
Vth		1.00	-0.71	Volts
Vpt		15.9	-13.0	Volts
Vbkd		15.8	-15.4	Volts
Idss		2398	-1248	uA
LARGE	50/50			
Vth		0.99	-0.75	Volts
Vjbkd		15.7	-18.3	Volts
Ijlk		-16.2	-1.3	pA
Gamma		1.14	0.45	V^0.5
Delta length (L_eff = L_drawn-DL)		0.37	0.30	microns
Delta width (W_eff = W_drawn-DW)		-0.39	0.00	microns
K' (Uo*Cox/2)		25.0	-10.1	uA/V^2

POLY2 TRANSISTORS	W/L	N-CHANNEL	P-CHANNEL	UNITS
MINIMUM	6/4			
Vth		1.09	-0.94	Volts
SHORT	36/4			
Vth		1.03	-0.90	Volts
LARGE	36/36			
Vth		1.00	-0.93	Volts
Delta length (L_eff = L_drawn-DL)		0.18	-0.44	microns
Delta width (W_eff = W_drawn-DW)		0.00	0.00	microns
K' (Uo*Cox/2)		20.4	-8.3	uA/V^2

FOX TRANSISTORS	GATE	N-ACTIVE	P+ACTIVE	UNITS
Vth	Poly	21.5	-11.9	Volts

PROCESS PARAMETERS	N+DIFF	P-DIFF	N+POLY	P+POLY	POLY2	METAL1	METAL2	UNITS
Sheet Resistance	24.4	71.9	20.4	22.3	19.4	0.05	0.03	Ohms/sq
Width Variation (measured - drawn)	0.09	-0.07	-0.20	-0.15	-0.17	-0.09	0.45	microns
Contact Resistance	21.4	70.6	9.1	11.2	10.1		0.03	Ohms
Gate Oxide Thickness	405							Angstroms

CAPACITANCE PARAMETERS	N+DIFF	P+DIFF	POLY	POLY2	METAL1	METAL2	UNITS
Area (substrate)	386	201	62		44	25	aF/um^2
Area (poly)				473	38	22	aF/um^2
Area (poly2)					38		aF/um^2
Area (metal1)						35	aF/um^2
Area (N-active)			825		40	23	aF/um^2
Area (P+active)			816				aF/um^2
Fringe (substrate)	578	128					aF/um
Fringe (N+active)			71				aF/um
Fringe (P+active)			94				aF/um

CIRCUIT PARAMETERS			UNITS
Inverters	K		
Vinv	1.0	2.27	Volts
Vinv	1.5	2.48	Volts
Vlow	2.0	0.00	Volts
Vhigh	2.0	5.00	Volts
Vinv	2.0	2.63	Volts
Gain	2.0	-12.11	
Ring Oscillator			
MOSIS (31 stages)		31.70	MHz

N41Q SPICE LEVEL 2 PARAMETERS

```
.MODEL CMOSN NMOS LEVEL=2 PHI=0.600000 TOX=4.2100E-08 XJ=0.200000U TPG=1
+ VTO=0.9597 DELTA=8.5280E+00 LD=1.8740E-07 KP=4.7466E-05
+ UO=578.7 UEXP=1.8310E-01 UCRIT=8.8450E+04 RSH=1.7240E+01
+ GAMMA=1.0283 NSUB=2.1850E+16 NFS=3.91E+11 VMAX=6.6700E+04
+ LAMBDA=2.4990E-02 CGDO=2.3057E-10 CGSO=2.3057E-10
+ CGBO=4.0753E-10 CJ=3.7086E-04 MJ=0.4303 CJSW=5.6047E-10
+ MJSW=0.384091 PB=0.800000
* Weff = Wdrawn - Delta_W
* The suggested Delta_W is -3.8600E-07
.MODEL CMOSP PMOS LEVEL=2 PHI=0.600000 TOX=4.2100E-08 XJ=0.200000U TPG=-1
+ VTO=-0.6892 DELTA=1.7970E+00 LD=1.5070E-07 KP=1.9218E-05
+ UO=234.3 UEXP=3.3320E-01 UCRIT=6.5300E+04 RSH=6.6000E+01
+ GAMMA=0.5915 NSUB=7.0900E+15 NFS=3.23E+11 VMAX=9.9990E+05
+ LAMBDA=4.9200E-02 CGDO=1.8541E-10 CGSO=1.8541E-10
+ CGBO=3.4681E-10 CJ=1.6874E-04 MJ=0.4794 CJSW=1.5082E-10
+ MJSW=0.07452 PB=0.700000
* Weff = Wdrawn - Delta_W
* The suggested Delta_W is 6.0720E-09
```

```
N41Q  SPICE BSIM PARAMETERS

NM1 PM1 DU1 DU2 ML1 ML2
*
*PROCESS=orbit
*RUN=n41q
*WAFER=10
*Gate-oxide thickness= 421.0 angstroms
*Geometries (W-drawn/L-drawn, units are um/um) of transistors measured were:
* 3.0/2.0, 6.0/2.0, 18.0/2.0, 18.0/5.0, 18.0/25.0
*Bias range to perform the extraction (Vdd)=5 volts
*DATE=03-15-94
*
*NMOS PARAMETERS
*
-7.18833E-01,-8.91684E-02, 9.14746E-02
 7.60096E-01, 3.67344E-24,-1.53167E-23
 1.03199E-00, 1.93365E-01, 3.56115E-01
-4.13719E-02, 1.26819E-01, 2.54641E-02
-3.39164E-03, 9.50544E-03, 1.94026E-02
 5.72474E+02,6.96277E-001,2.65265E-001
 5.38397E-02, 5.03034E-02,-4.52294E-02
 2.29187E-02, 5.12172E-01,-3.06096E-01
 1.43917E+01,-3.16332E-01, 6.45259E+01
-1.26648E-03,-4.39846E-03, 2.87274E-03
-2.47418E-05,-1.45592E-03,-4.51830E-03
 1.47241E-03,-1.34515E-02, 3.99971E-02
 5.48715E-03,-5.13601E-03,-1.49466E-02
 5.51798E-02, 2.91387E+02,-5.99319E+00
 7.12957E-00,-3.32071E+01, 9.23007E+01
-4.07965E+00, 6.28764E+01,-2.49727E+01
-6.42707E-03, 6.67362E-02,-2.25240E-02
4.21000E-002, 2.70000E+01, 5.00000E+00
4.28327E-010,4.28327E-010,4.10748E-010
1.00000E+000,0.00000E+000,0.00000E+000
1.00000E+000,0.00000E+000,0.00000E+000
0.00000E+000,0.00000E+000,0.00000E+000
0.00000E+000,0.00000E+000,0.00000E+000
*
* Gate Oxide Thickness is 421 Angstroms
*
*
*PMOS PARAMETERS
*
-4.73363E-01, 2.74087E-01,-3.90480E-01
 6.84733E-01, 2.40119E-24, 0.00000E+00
 6.32064E-01,-4.35630E-01, 6.24686E-01
 5.92240E-02,-9.35089E-02, 1.31036E-01
-7.65985E-03, 6.09917E-02,-1.56246E-02
 2.41175E+02,3.24954E-001,1.65787E-001
 1.28862E-01, 3.98049E-02,-7.36381E-02
 2.19034E-02, 3.21993E-01,-6.67338E-02
 1.17270E+01,-6.82703E+00, 5.30017E+00
 4.41482E-04,-3.80214E-03,-3.92463E-03
 7.47896E-04,-4.16189E-03, 1.37135E-03
 8.71044E-03,-5.30563E-03, 1.57807E-03
 1.51614E-03,-8.20391E-04, 1.34531E-02
 2.65469E+02, 1.07537E+02, 5.79252E+01
 1.19751E+01,-4.73836E+00, 1.72411E+01
-1.04839E-01, 1.41411E-01,-6.77270E-01
-1.60734E-02,-5.59496E-03, 2.03221E-02
4.21000E-002, 2.70000E+01, 5.00000E+00
1.99901E-010,1.99901E-010,3.86210E-010
1.00000E+000,0.00000E+000,0.00000E+000
1.00000E+000,0.00000E+000,0.00000E+000
```

```
0.00000E+000,0.00000E+000,0.00000E+000
0.00000E+000,0.00000E+000,0.00000E+000
*
*N+ diffusion::
*
24.05,     3.708600e-04,     5.604700e-10,     1.000000e-08,     0.8
0.8,     0.430276,     0.384091,     0,     0
*
*P+ diffusion::
*
72.48,     1.687400e-04,     1.508200e-10,     1.000000e-08,     0.7
0.7,     0.479427,     7.452000e-02,     0,     0
*
*METAL LAYER -- 1
*
5.110000e-02,     2.600000e-05,     0,     0,     0
0,     0,     0,     0,     0
*
*METAL LAYER -- 2
*
2.732000e-02,     0,     0,     0,     0
0,     0,     0,     0,     0
```

REFERENCES

1. Conway, L. A. & Mead, C. A., 1980, *Introduction to VLSI Systems*, 1st Ed., VLSI System Series, Addison-Wesley Publishing Company, Reading, MA.

2. Eshraghian, Kamran & Weste, Neil, Oct. 1985, *Principles of CMOS VLSI Design A Systems Perspective*, 1st Ed., VLSI System Series, Addison-Wesley Publishing Company, Reading, MA

3. Gosser, R. & Murden, F., Feb. 17, 1995, *A 12b 50Msample/s Two-Stage ADC*, Analog Devices, Greensboro, NC, at the IEEE International Solid-State Circuits Conference

4. Horowitz, P. & Hill, W. , 1989, *The Art of Electronics* , 2nd edition, Press Syndicate of the University of Cambridge, NY, NY

5. Nauta, Bram & Venes, Ardie G. W., Dec. 1995, *A 70-MS/s 110-mW 8-b CMOS Folding and Interpolating A/D Converter*, IEEE Journal of Solid State Circuits, Piscataway, NJ, Number 12, Volume 30

6. --,1991, *DATEL Databook Volume 1*, Datel Inc, Mansfield, MA.

7. --, 1994, *Harris Semiconductor Data Acquisition 1994*, Harris Semiconductor, Palm Bay, FL.