ADA026111

VOLUME 3 — FINAL REPORT

# ADAPTIVE PROGRAMMABLE SIGNAL PROCESSOR STUDY

APRIL 1976

AEROSPACE GROUPS

## HUGHES

HUGHES AIRCRAFT COMPANY
CULVER CITY, CALIFORNIA

DDC
RECEIVED
JUN 30 1976
A

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER <br> P76-121 | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle) <br> Adaptive Programmable Signal Processor Study. Volume 3. | | 5. TYPE OF REPORT & PERIOD COVERED <br> Final rept. <br> 20 June 1975 to 30 Apr 1976 <br> 6. PERFORMING ORG. REPORT NUMBER <br> D4535 SDN: M-59506 |
| 7. AUTHOR(s) <br> K. E. Myers | | 8. CONTRACT OR GRANT NUMBER(s) <br> F04701-75-C-0241 <br> ARPA Order-2954 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS <br> Hughes Aircraft Company <br> Electro-Optical Division <br> Culver City, California 90230 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS <br> CDRL A012 <br> DI-S-3591/A/M |
| 11. CONTROLLING OFFICE NAME AND ADDRESS <br> Defense Advanced Research Projects Agency <br> Arlington, Virginia 22209 | | 12. REPORT DATE <br> Apr 1976 <br> 13. NUMBER OF PAGES <br> 750 |
| 14. MONITORING AGENCY NAME & ADDRESS(If different from Controlling Office) <br> USAF Space and Missile Systems Organization <br> P. O. Box 92960 <br> Los Angeles, California 90009 | | 15. SECURITY CLASS. (of this report) <br> UNCLASSIFIED <br> 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Distribution limited to DOD agencies only. Other requests for this document must be referred to the Department of the Air Force, Hq Space and Missile Systems organization, P. O. Box 92960, Worldway Postal Center, Attn: DYA, Los Angeles, California 90009.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

HAC-Ref-D4535-Vol-3; HAC-P76-121-Vol-3

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

18. SUPPLEMENTARY NOTES

SAMSO

TR-76-83-Vol-3

DDC RECEIVED JUN 30 1976

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Signal Processing for Satellite Applications
Programmable Signal Processors
Adaptive Signal Processors
Real Time Signal Processing
On-Board Signal Processing

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report presents the results of a study to define an Adaptive Programmable Signal Processor (APSP) suitable for on-board satellite processing of data generated by spaceborne electro-optical surveillance sensors and dual mode radars. The tasks performed included: 1) definition of system requirements based upon mission requirements supplied by SAMSO, 2) definition of processor performance requirements, 3) configuration of a processor architecture to meet those requirements, 4) evaluation of present and projected semi-conductor device technology applicable to APSP development,

# DISCLAIMER NOTICE

THIS DOCUMENT IS THE BEST
QUALITY AVAILABLE.

COPY FURNISHED CONTAINED
A SIGNIFICANT NUMBER OF
PAGES WHICH DO NOT
REPRODUCE LEGIBLY.

5) identification and preliminary design of those devices critical to APSP development, and 6) preparation of a plan for designing, fabricating and testing a feasibility demonstration model of an APSP, including requirement component development.

VOLUME 3

FINAL REPORT

ADAPTIVE PROGRAMMABLE SIGNAL PROCESSOR STUDY

April 1976

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency or the U.S. Government.

Electro-Optical Division

AEROSPACE GROUPS

Hughes Aircraft Company ● Culver City, California

## MASTER CONTENTS

### VOLUME 1

A

## MASTER CONTENTS (Continued)

## MASTER CONTENTS (Continued)

## VOLUME 2

## SECTION IV - RADAR PROCESSOR REQUIREMENTS

C

## MASTER CONTENTS (Continued)

D

MASTER CONTENTS (Continued)

## VOLUME 3

### SECTION V - PROCESSOR ARCHITECTURE

E

MASTER CONTENTS (Continued)

F

MASTER CONTENTS (Continued)

G

## MASTER CONTENTS (Continued)

MASTER LIST OF ILLUSTRATIONS

VOLUME 1

SECTION II - SYSTEM REQUIREMENTS

## MASTER LIST OF ILLUSTRATIONS (Continued)

### SECTION III - PROCESSOR REQUIREMENTS

MASTER LIST OF ILLUSTRATIONS (Continued)

## VOLUME 2

## SECTION IV - RADAR PROCESSOR REQUIREMENTS

MASTER LIST OF ILLUSTRATIONS (Continued)

MASTER LIST OF ILLUSTRATIONS (Continued)

M

## MASTER LIST OF ILLUSTRATIONS (Continued)

### VOLUME 3

### SECTION V - PROCESSOR ARCHITECTURE

## MASTER LIST OF ILLUSTRATIONS (Continued)

## MASTER LIST OF ILLUSTRATIONS (Continued)

MASTER LIST OF ILLUSTRATIONS (Continued)

SECTION VI - TECHNOLOGY SURVEY

MASTER LIST OF ILLUSTRATIONS (Continued)

MASTER LIST OF ILLUSTRATIONS (Continued)

MASTER LIST OF ILLUSTRATIONS (Continued)

## MASTER LIST OF ILLUSTRATIONS (Continued)

## MASTER LIST OF ILLUSTRATIONS (Continued)

### SECTION VIII - DEVELOPMENT PLAN

# ARCHITECTURE STUDY
## FOR
## AN ELECTRO-OPTICAL ADAPTIVE PROGRAMMABLE
## SIGNAL PROCESSOR

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency or the U.S. Government.

| | |
|---|---|
| ARPA Order Number | 2954, Amendment No. 1 |
| Program Code Number | None |
| Name of Contractor | Hughes Aircraft Company |
| Effective Date of Contract | 20 June 1975 |
| Contract Expiration Date | 13 February 1976 |
| Amount of Contract | $498,159 |
| Contract Number | F04701-75-C 0241 |
| Principal Investigator and Phone Number | K.E. Myers, 391-0711, X7598 |
| Project Engineer and Phone Number | K.A. Krause, 391-0711, X2243 |
| Short Title of Work | Processor Architecture for APSP |
| Date of Report | 20 February 1976 |
| Contract Period Covered by Report | 1 September 1975 to 31 January 1976 |

K.E. Myers
Program Manager

## Electro-Optical Division
## AEROSPACE GROUPS
### Hughes Aircraft Company ● Culver City, California

# CONTENTS

CONTENTS (Continued)

# LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

## LIST OF ILLUSTRATIONS (Continued)

## LIST OF ILLUSTRATIONS (Continued)

## 1.0 INTRODUCTION AND SUMMARY

This report presents the results of the Electro-Optical Processor Definition Task, statement of work item 3.3, of the Adaptive Programmable Signal Processor (ACCD$^2$) program.

The work described is based upon information and requirements contained in the program's Mission Requirements, Systems Requirements, and Processor Requirements documents, prepared earlier in the program.

Section 2.0 of this report summarizes the requirements placed upon, and the functions to be performed by, the APSP. This section also discusses the issue of design commonality for electro-optical and radar processors. The conclusion is that there exists considerable device commonality, e.g., both types of processors require high speed multipliers; moderate functional commonality, e.g., high capacity memories are used by both processors; but little architectural commonality, i.e., those common devices are interconnected in completely different ways.

Section 3.0 details the two independent processor architectures which were developed, then merged to obtain the best features of each. The section begins by describing both of the approaches, and concludes with a description of the consolidated architecture.

Section 4.0 contains descriptions of each of the functional modules for the consolidated processor. Included are register level diagrams and signal flow charts. Partitioning of functions between hardware and software is also treated in this section.

Section 5.0 discusses software for the APSP application, particularly development of algorithms for the multi-target track problem. Track initiation, maintenance and termination criteria are treated, along with inter-pixel boundary problems. Estimates of instruction counts, storage requirements and execution times are included.

1-1

The report concludes in section 6.0 with an analysis of projected performance for the APSP, based upon the architecture and mechanizations described in this report, coupled with the device parameters contained in the Technology Survey Report, CDRL A007, and the Critical Device Design Report, CDRL A008, submitted earlier in the program.

# 2.0 REQUIREMENTS AND FUNCTIONS

## 2.1 REQUIREMENTS

The basic functional and performance requirements were delineated in the Performance Requirements Report, dated October 1975, and are summarized in Table 2.1. The APSP accepts data from the focal plane chip at a 164K samples/second ($1.64 \times 10^4$ detectors sampled at a 10 Hz readout rate), and after performing various filter functions, including both temporal and spatial, will track potential targets and output state vectors at the rate of one per second per track.

## 2.2 COMMONALITY BETWEEN THE RADAR SIGNAL PROCESSOR AND THE ELECTRO-OPTICAL PROCESSOR

The primary technical factors which differentiate the radar signal processor task from that of the electro-optical sensors result from the ranging capability of the radar, which is not available in the passive E-O system, and the relative rates of sampling which are required. (If an E-O system were to incorporate active laser ranging, significantly greater commonality would exist in the signal processor.) Radar data must be gathered at intervals determined by the transmitted pulse rate and at time increments compatible with the desired range resolution. In comparison, the MFPA sensors receive data continuously and their output is sampled at rates determined by target and background variations in time.

The radar processor derives considerable capability from the coherent nature of its sensor and receiver and from the ability to resolve almost microscopic variations in the doppler shifts of the target, whereas the optical data processor performs temporal and spatial filtering to reduce the dynamic range caused by noise or clutter background. The adaptive

2-1

## TABLE 2-1. PERFORMANCE REQUIREMENTS SUMMARY

| | |
|---|---|
| Detector channels | $4.2 \times 10^6$ |
| Number of simultaneous tracks | 5000 |
| Transient dynamic range | $10^3$ |
| System dynamic range | $10^7$ |
| Maximum input data rate | 1.6 MHz |
| Clutter rejection | 26 dB at $V_T/V_C = 5$ |
| Velocity discrimination | 0.3 pixels/sec $\Delta V$ at 3 pixels/sec |
| Tracking accuracy ($1\sigma$) | 0.25 pixels |
| Star rejection | 100 percent |
| Output track parameters | X, Y, $V_X$, $V_Y$, J, ID |
| Nominal state vector update | once/sec |
| Target velocity | 0 - 8 pixels/sec |
| 10 year radiation dose | $10^4$ rad(Si) |
| Power dissipation | 128 watts |

features of the electro-optical processor appear strongly in the front end, or at the sensor array itself, whereas the only comparable features in the radar, the AGC and adaptive thresholds, are mechanized further along in the signal processing chain.

Angle tracking circuits in the two processors could utilize similar track files in the main processor memory and similar algorithms in closed tracking loops. The basic angle sensing information in the electro-optical sensor originates in the spatial filtering functions, while the radar, having only a single sensor pointing direction, obtains its basic directional data without any significant tracking filter process. There is no range tracking discriminant function in the electro-optical processor which is comparable to that of the radar.

Doppler filtering is performed in the radar system as a means of excluding broad band noise and clutter and obtain signal-to-noise levels sufficient for purposes of detection. This coherent integration utilizes narrowband filter characteristics with inherently low sidelobe response in order to avoid velocity ambiguity and to obtain unequivocal velocity resolution. The Fourier transform filter is a natural choice for this function. In the electro-optical processor, however, other filter transforms may be useable and relatively advantageous since the transform may be used to narrow the bandwidth of the data processor for a given degree of target detectability. Some forms of the Walsh Hadamard transform (which are unsuitable for radar usage because of undesirable sidelobe characteristics) appear to be advantageous for E-O data processing and target detection.

These transforms may be effective for the MFPA because they are well adapted to the unique character of its data, including broad near-uniform areas of clutter, spacial edges and large dynamic range signals. Target radial velocity has only a secondary influence on the E-O tracker through its effect on the observed brightness, while in the radar system the target velocity doppler effect is the primary factor which permits detection in clutter.

These considerations are important in assessing the degree of commonality between electro-optical and radar processing equipment.

2-3

The AVE (adaptive video encoder) element of the electro-optical processor is functionally integrated and contains on the sensor chip some of the basic temporal filter functions. Thus, its temporal filter function would not be available to the radar, and would not be useable.

The output of the Adaptive Video Encoder provides the input to the layered array (LAP) signal processor. The basic functional blocks of the LAP are shown in Figure 2-1. The point target processor performs video integration, compensation for sensor sensitivity variations for each pixel element, and area correlation of data from neighboring pixels.

The tracking processors execute target tracking algorithms under the general control of the APSP. The computations include a determination of the next probable position for tracks as well as algorithms compensating for the gaps caused by apparent cessation of target motion or gaps in the MFPA chip array. Current and past track histories are also maintained.

The APSP controller exercises supervisory control over the LAP units. It determines the LAP modes, issues commands to the appropriate units, and assigns targets to specific tracking processors. Track and target data for transmission to the earth is selected by the data communication interface.
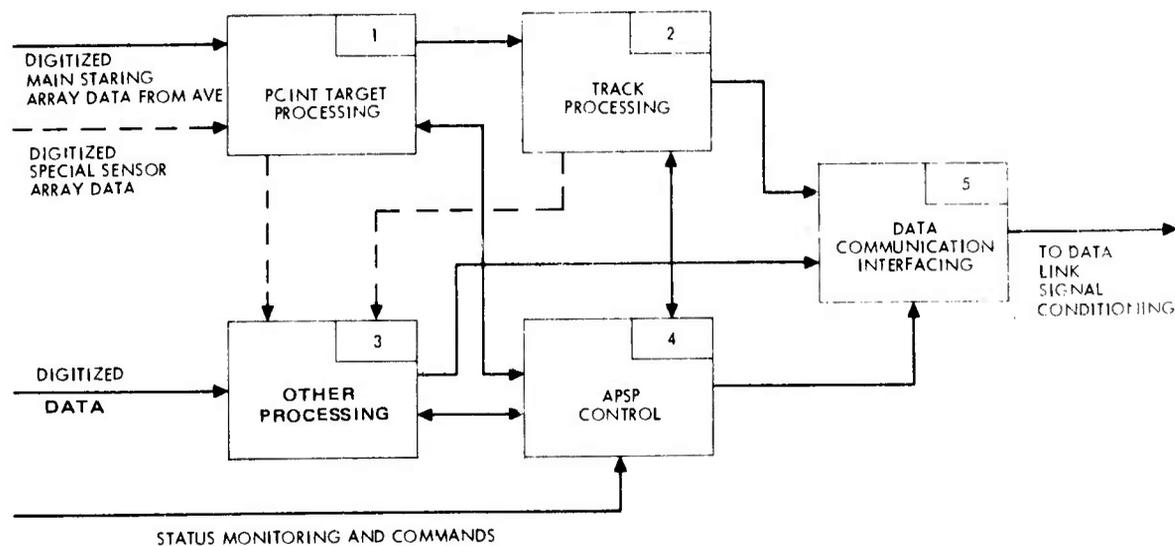


Figure 2-1. Functional diagram of LAP.

Certain portions of the LAP appear to be useful for radar signal processing, with minor changes or additions. Other portions provide functions which benefit only the electro-optical mission, particularly those elements which have specialized functions unique to the MFPA sensor. The point target processing element, for example, is specialized and does not perform a process useful to the radar. Data reduction prior to transmission is not required for the radar sensor, however the angle track processing and data interface might have a significant utility for radar signal processing.

## Conclusion

The functional correspondence between the electro-optical processor and that of the radar is rather limited. The functions of the AVE and the point target processing elements do not resemble the needs of the radar processor. The angle track element may be useful if there is a requirement for radar track files, but at this time, there is no such requirement. The data communication interface and processor control would probably be serviceable for radar functions.

In view of these considerations, the functional correspondence between the requirements (and therefore the architecture) of the two types of signal processing is not great enough to justify a unified design. This conclusion does not apply to the device development aspects of the program since the great preponderance of CCD devices proposed for future design can be utilized in either system.

Excluding the MFPA and the CCD A/D converter, all of the remaining devices for which conceptual designs have been originated appear to be mutually useful. This includes items such as the CCD full adder, the CCD D/A, the on-chip clock driver and input-output semiconductor design efforts. In addition, both processors will make extensive use of low power, high capacity, digital memory devices. Any developments which can meet the specialized requirements of very low power and long life for the optical processor will likely be of substantial benefit to the radar.

Thus, the commonality between the optical and radar is limited in terms of <u>architecture</u> and <u>function</u>, but there appears to be a significant degree of potential joint usage of specialized, custom designs of CCD or other low power devices.

## 3.0 APSP ARCHITECTURE

### 3.1 INTRODUCTION AND BACKGROUND

As the task of developing an architecture for the APSP progressed, it became clear that substantial divergence of opinion existed among competent technical personnel as to what constituted an optimum design. Upon examination, it was apparent that the diversity of concepts really represented variations on only two fundamental approaches. The first approach was to have the track processor request the digitized data for specific detector elements within the projected tracking gate from the point target processor. The second approach was a passive signal processor which performed temporal and spatial filtering on the digitized data from all detector elements, and passed only information which exceeded a threshold to the track processor.

At this point, two independent technical teams were formed, each charged with developing the "optimum" Adaptive Programmable Signal Processor based upon the requirements and information contained in two program documents prepared earlier:

1. The Systems Requirements Report, CDRL A003
2. The Processor Requirements Report, CDRL A004

In mid-November a series of meetings were held with each team presenting, describing, and to some extent, defending its approach. These meetings resulted in a detailed examination and comparison of the two approaches. It became apparent that the first approach required a very complicated switching network and high data rates in the tracker communication network. However, several of the novel concepts from that approach such as

adaptive velocity filters and tracking processor direction of the saturation control logic, have been maintained and incorporated. Both approaches utilized the same Adaptive Video Encoder (AVE) discussed in Section 4.1.

The Adaptive Video Encoder and both of the initial concepts are described, at the system level, in this section of this report in compliance with the contractual requirements that program reports descr┅ "all work performed, knowledge gained, and results achieved". Howev┅, only the amalgamated design was further refined in the register level. Section 3.5 of this report discuss that design.

## 3.2 THE LAYERED ARRAY PROCESSOR (A)

This section describes the configuration proposed by the first of the two independent teams.

Basic Processor Configuration

The basic configuration is shown in Figure 3.2-1. The point target processing function includes uncorrelated pixel processing (1st layer) and correlated pixel processing (2nd layer) for improved clutter rejection. The track processing function implements tracking algorithms. It includes
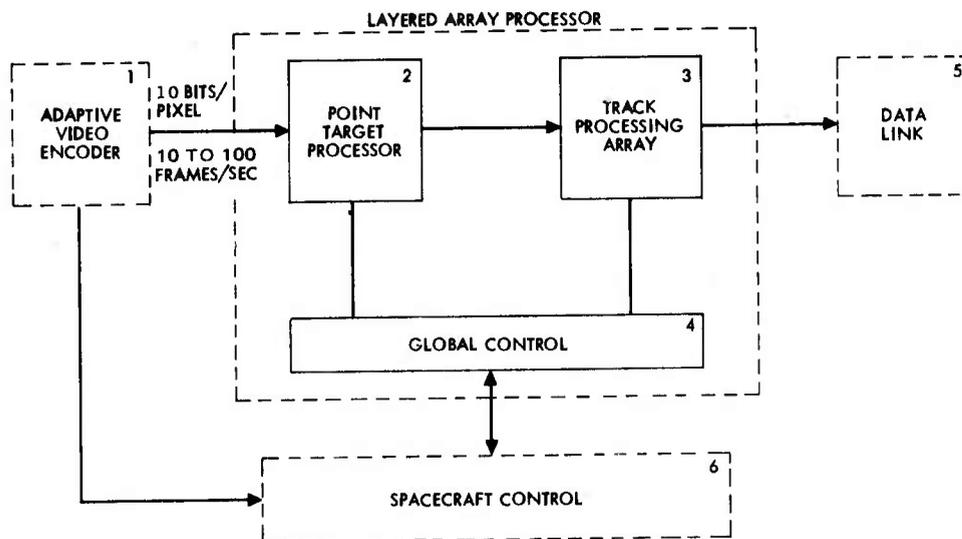


Figure 3.2-1. Processor A functional block diagram.

the data communication interface (3rd layer) and the array of computing elements (4th layer) which execute the individual tracks. The entire layered array processor is under executive control of the computers in the control section which implement le changes and coordinate the actions of individual trackers. Reports oi rent tracks are relayed to the ground through the data link. Primary processor commands are in the spacecraft control section, and alarms and reports on the condition of the processor are reported to spacecraft control.

The designs must be expandable to near term applications of $4 \times 10^6$ pixels with eventual applications of $10^8$ picture elements (pixels). Each pixel has a position (i, j, ), and a magnitude (q) associated with it. The size of a pixel equals the detector instantaneous field of view. Each detector chip is assumed to contain an array of $128 \times 128$ pixels. A small amount of insensitive space (2 to 5 pixels width) is assumed between detector chips. A $16 \times 16$ chip sensor array ($4 \times 10^6$ pixels) can be mounted on a single 8" x 8" substrate. Applications with more pixels will require multiple sensor substrates and larger spaces (10 to 30 pixels width) will be assumed between substrates.

The AVE provides amplitude reconstructed data (10 bit) which has A maximum transfer rate of 100 frames per second is assumed into the point detectors. An algorithm to reduce sensor impulse noise is also in the AVE. A constai transfer rate of 100 frames per second is assumed into the point target procssor. Lower effective frame rates are obtained in the LAP by digital time integration, as appropriate for detection over specific target velocity ranges. The number of hardware data channels used will be selected to be compatible with the degrees of parallelism in the AVE and target processor. The interfaces with spacecraft control and the data link are general purpose computer-type block transfer ports.

Figure 3.2-2 illustrates the LAP from a different viewpoint. Processing for one detector chip ($128 \times 128$ pixels) is emphasized. With the capability for a high frame rate (100 frames/sec.) and the necessity for several frames of digital storage, several first and second layer processing chips are needed for each detector chip. Sixteen first layer and sixteen second layer are shown. One or more extra first layer chips will be
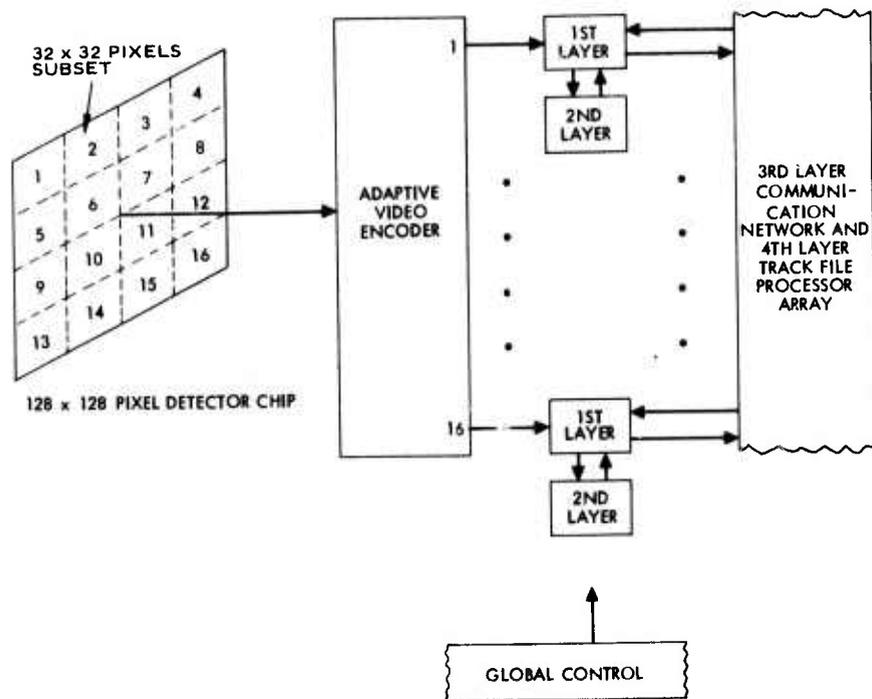
Figure 3.2-2. Processing hardware for a
128 x 128 pixel detector array.

provided for redundancy to improve fault tolerance, while careful layout of
second layer chips should reduce the requirement to eight per detector chip.
The dashed lines show one way that the data can be divided for second layer
processing (32 x 32 pixel squares). However, 8 x 128 pixel rectangles
should prove more efficient.

## Target Processing

Target processing is shown functionally in Figure 3.2-3. Signal
amplitude is input from the AVE. Globally selectable digital time integra-
tion allows the frame integration time to be set to optimize the detection of
expected rapidly changing targets. Data can then be passed through area
correlation (spatial filtering). Area correlation is helpful in distinguishing
point targets from distributed clutter such as clouds or some types of sun
glint. However, it will not help when tracking targets against a star back-
ground. When a specific element of the second layer array fails, operating
the corresponding first layer chip without area correlation is an appropriate
degraded mode of operation.

3-4
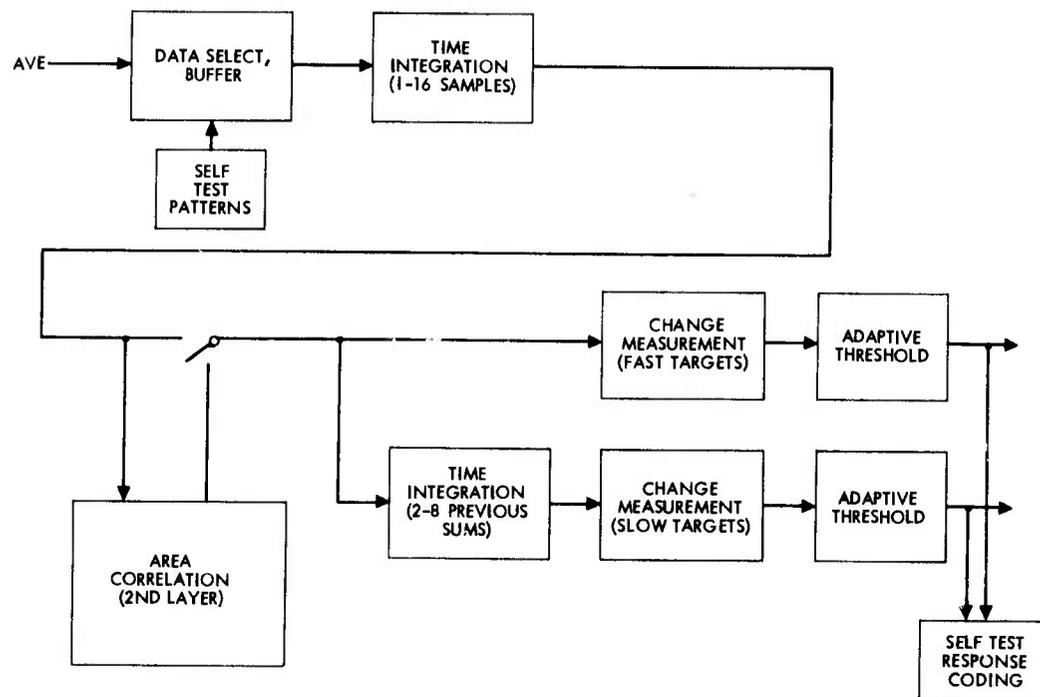
Figure 3.2-3.   Point target processing.

The signal path then divides.   One branch goes through change measurement (time filtering) for fast targets.   This is optimized (via wider bandwidth) to provide maximum sensitivity for fast targets.   The other branch simultaneously provides additional digital time integration to supply data for change measurement for slow targets.   Separate adaptive thresholds are maintained for both fast and slow target detection.   These are determined independently and dynamically for each pixel.   Thresholds are determined from the apparent noise level at the target detector.   The adaptive threshold feature can adjust to changes in target detectability due to different clutter conditions in different parts of the image and due to different hardware conditions such as noisy sensor cells or the absence of sensors on lines between sensor chips.   The diagram also shows some of the special logic necessary for efficient self test.

## Time Integration

Time integration of incoming data is performed to increase the signal to noise ratio and reduce the effect of transients. The integration is performed in two stages to allow detection of both fast and slow targets. To integrate the data for fast target detection, 1 to 16 samples of data for each pixel are summed. The second integration then adds from 2 to 8 of these previous integration summations to allow detection of slowly changing targets. Thus, for slow target integration, summations of up to 128 samples of incoming data are possible. The number of samples summed by each integration is independently selectable under global control. Four guard bits are provided for the first integration and three are supplied for the second to prevent overflow. The integrated sums are rounded off to 16 bits and scaled to assure that the most significant bits are transferred regardless of the number of samples summed. The results of the first integration are used to generate the area correlation data which is used to detect both fast and slow targets.

A functional implementation of the first integration is shown in Figure 3.2-4. Data entering from the input buffer is added to the temporary sum that is kept in the memory for each pixel. The adder is bit serial and represents a minimal amount of extra area for the chip. The select unit is capable of selecting the temporary sum for normal integration, zero for initiating a new sum, or a shifted version of the sum at the completion of an



• 1-16 SAMPLES SELECTED BY GLOBAL CONTROL
• RESULTS SCALED, ROUNDED
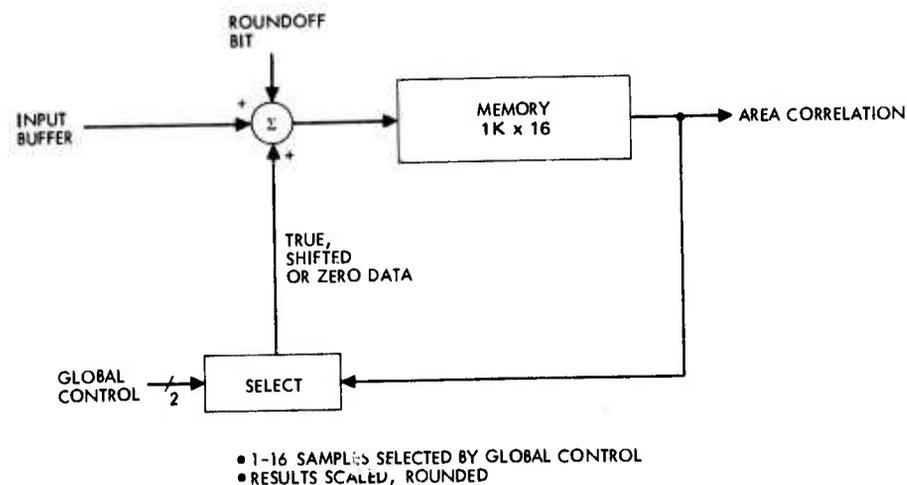
Figure 3.2-4. Time integration I.

integration to accomplish scaling. The selection is performed by global control. The memory contains 1024 words, each of which has 4 guard bits to prevent overflow. At the completion of a summation sequence, round-off is accomplished by adding a roundoff bit to the least significant bit position of the 16 most significant bits produced by scaling. The final summations for each pixel are transferred to the area correlation and change measurement filters for further processing.

Spatial Filtering

Spatial filtering is required to help distinguish moving point targets from changing backgrounds such as those from moving clouds or changing sun glint patterns. The key feature which allows discrimination is that the changing background patterns are correlated over a number of adjacent pixels. This is not the case for point targets.

The processing for spatial filtering considers each pixel along with its eight neighbors as shown in Figure 3.2-5. The neighbors are the 4 pixels on the edges (the E's) and the 4 pixels on the diagonals (the D's). Symmetrical filters are assumed. The $C_{AVE}$ relation is appropriate for computing
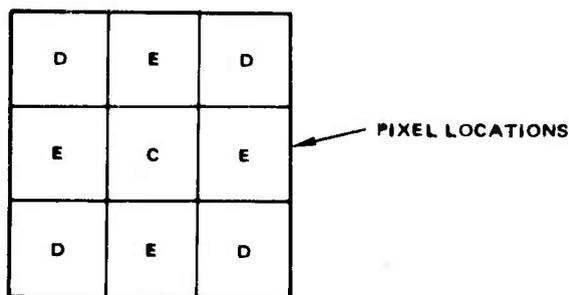


$$C_{AVE} = W_C C + W_E (\Sigma E) + W_D (\Sigma D)$$

$$P_C = C \cdot W_E (\Sigma E) \cdot W_D (\Sigma D)$$

Figure 3.2-5. Spatial filtering
concept.

background references to be used for time change measurement. The $P_C$ relation evaluates a property which could be called peakedness at the central point. This is the value of the central point less an average predicted for that central point based on the eight neighbors. The $W_C$, $W_E$, and $W_D$ are weighting constants which will be selected to provide good filter responses. The calculations are repeated for all pixels as central points. For pixels on the edges of sensor chips, the available neighbors are used with different W's.

Hardware for area correlation, provided as a second layer, is illustrated in Figure 3.2-6. Data for area correlation is stored in up to four 16 bit words for each of 1024 pixels. Inputs to this shift register memory are selected from either of two first layer chips (for fault tolerance) or recirculated for further correlation calculations. Correlation arithmetic is performed in pipelined parallel fashion on the central pixel being processed and eight neighbors. Correlation or averaging of more distant pixels can be
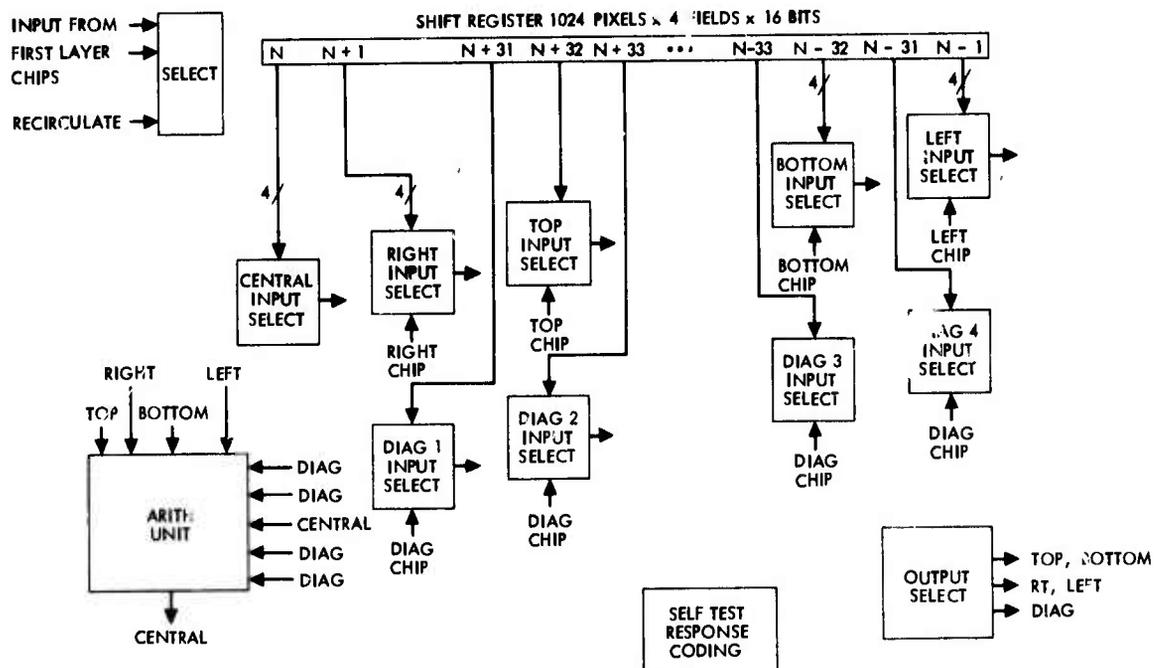


Figure 3.2-6. Second layer block diagram.

accomplished in multiple passes. Each neighbor is automatically selected by connection of shift register taps whose location corresponds to neighbors' array positions. The taps for the 4 data words of each neighbor pixel enter an adder input select multiplexer. This allows any neighbor pixel word (field) to be operated on, selectable by global control. Appropriate input lines from adjacent chips may also be selected where neighbor pixels cross chip boundaries. The corresponding chip output words are also selected for use by neighbor chips as such input data. Compensation for sensor gaps may be accomplished by substituting central pixel or fixed values on the neighbor chip input lines.

Second layer operational capabilities are indicated in Figure 3.2-7. An operation is performed on all pixels in a single pass, with correlation to all 8 neighbors. Use of 4 words per pixel allows prior and newly updated pixel values to co-exist in memory. The first pixel to be processed in an array will require the most recently updated neighbor data from the previous

- PERFORMS ADD, SUBTRACT, MULTIPLY, ARITH RIGHT SHIFT, SET FLAG ON COND.
- OPERATES ON (8) NEIGHBORS AND CENTRAL PIXEL IN PARALLEL NEIGHBORS CAN BE ON ADJACENT CHIPS, 2ND LAYER COMPENSATES FOR SENSOR GAPS
- EACH OPERAND MAY BE ANY OF (4) PIXEL DATA FIELDS
- CONSTANT MAY ALSO BE USED FOR OPERAND
- SELECTS OPERANDS FROM COMMON PROCESSING FRAME TIME
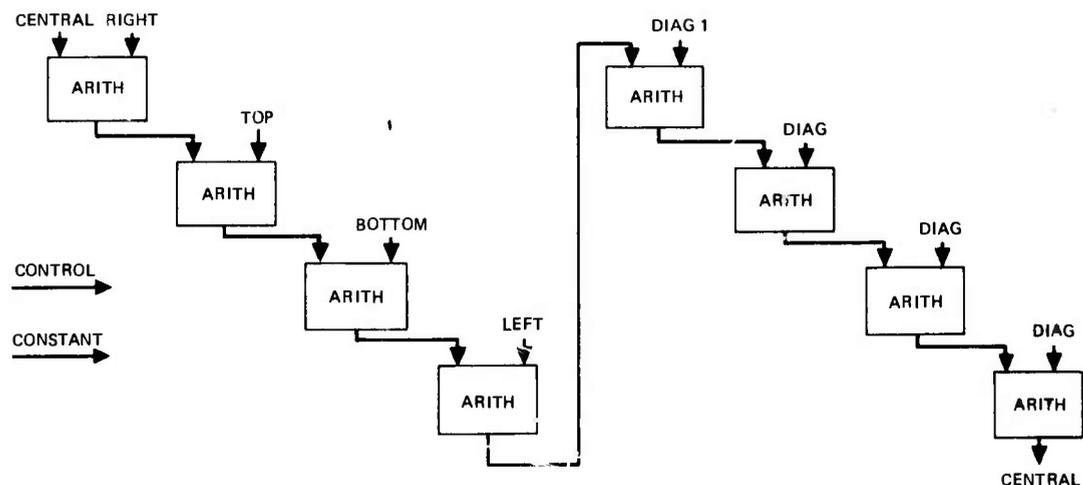
Figure 3.2-7. Second layer arithmetic unit.

3-9

frame. The last pixel to be processed in the frame will still require neighboring data from the previous frame even though updated data is now available also.

First layer data will be input continuously as processing is being performed on data from the previous frame. Thus, when a pixel enters the arithmetic unit (one frame after being input from first layer) all neighbor pixels have also been stored in second layer memory. Data is processed to 16 bit accuracy with rounding and scaling.

Change Measurement

Change measurement determines the differences between time integrated data and predicted values based on time integrated area correlated data. Change measurement is accomplished for each individual pixel. Programming by global control allows changes to be computed for pixel data separated in time by any number of time integration frames. As indicated on the hardware block diagram (Figure 3.2-8), data may be recirculated until the desired time difference for change computation occurs.

The hardware sums past and future area correlated data to provide an estimate of the current background value. Future data is taken from the second layer as needed with no time delay. Past data is delayed for 2 time difference periods by the 2 memories shown. The summed area correlated past and future data (16 most significant bits) are then subtracted from the present value which has been delayed one time difference period. (The present value may be delayed one additional frame to compensate for area correlation delay).

If failures in the second layer cannot be compensated by redundant chips, time integrated data without area correlation may be selected to provide an estimate (without use of second layer data) at somewhat degraded performance levels. This configuration is also useful for deep space tracking where area correlation is unnecessary.

Two change measurement units are provided to allow independent change measurement for fast and slow targets. The number of time integration frames between change values may be independently varied for slow and fast targets. One change detection unit allows additional integration of both

Figure 3.2-8.   Change measurement.

(previously integrated) pixel and area correlated data, providing simultane-
ous separate globally controlled integration times for fast and slow targets.
Integration of up to 128 samples can thus be accomplished.

Adaptive Thresholding

Adaptive thresholding provides a variable threshold for each pixel for
target detection.   The threshold is based on a scaled average of magnitudes
of several previous differences between estimated and measured values.
The threshold may be a summation of 1 to 32 previous difference magnitudes
which are scaled (through division by 1, 1/2, or 1/4) by shifting.  The number
of differences summed and the scaling factor are selectable under global
control.   The threshold may be set in increments of 1/4 difference to any
number of differences up to 8.   Additionally 9 to 32 differences may be used
as a threshold with a coarser increment of threshold selection.   Scaling by
1/8 may be added to provide finer threshold selection of up to 16 differences.
A constant may be selected under global control instead of the previous

3-11

difference sum, or as a minimum level to be used with a previous difference sum.

The adaptive threshold hardware (Figure 3.2-9) stores the previously computed threshold value for use while differences are being summed to compute the new threshold. A target hit (threshold exceeded) will cause threshold summation of the pixel containing the hit to be disregarded. A 1K x 1 bit memory (not shown) stores the hit data. Absolute value is obtained by selecting true or one's complement change detection data and providing a carry in, if necessary, to the difference summation adder.

Independent threshold computations are performed for fast and slow targets. Threshold levels (number of differences and scaling) may be set independently for fast and slow targets.
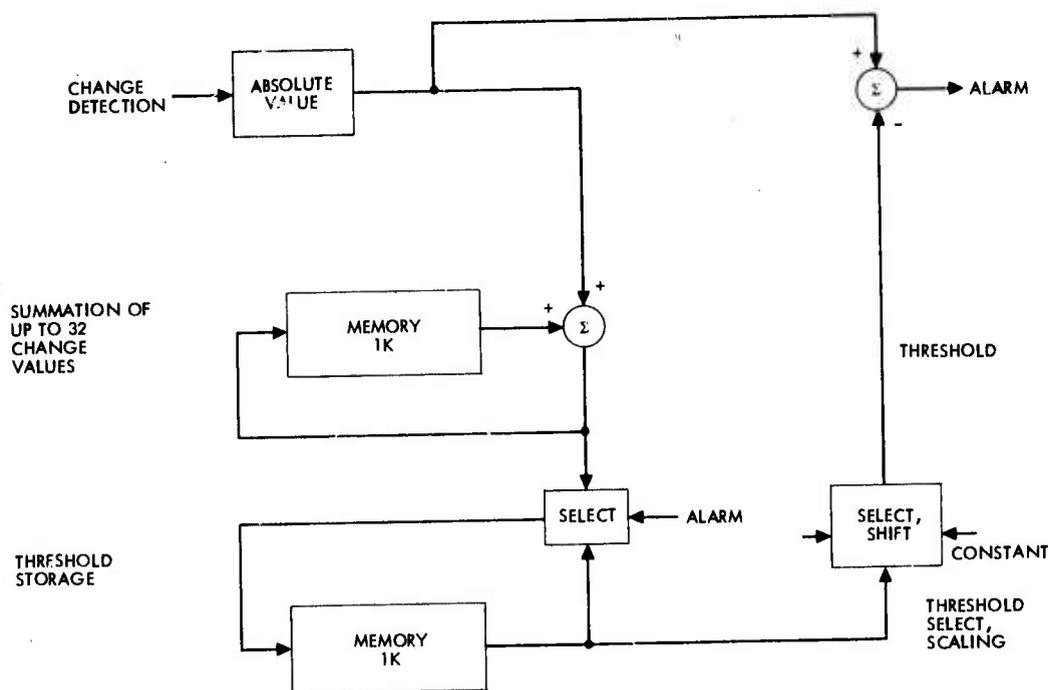


Figure 3.2-9. Adaptive thresholding functional diagram.

3-12

## Burst Location

Burst centroid location can be accomplished in the sequence shown below. In a burst location mode, one or multiple burst centroid locations are determined by parallel second layer processing.

- Burst location mode triggered by burst detector
- Second layer sets flag for each saturated pixel
- Number of saturated pixels in each row, column summed by second layer
- Centroid locations(s) at intersection of row and column with greatest number of saturated pixels
- Events in sensor array gaps may be located by effects on edge pixels

## Point Target Processor Fault Tolerance

Fault tolerance is achieved in the Point Target Processor by periodic tests under global control which isolate any faulty chips. Operational redundant chips are switched by global control to replace those chips found to be faulty. Fault tolerant features for the point target processor are summarized below.

Self test of first and second layer chips is performed using techniques of the Advanced Avionics Fault Isolation System (AAFIS), dev ed under government contract[*]. AAFIS utilizes test pattern generato ontained within units under test to provide self test. est responses (chip outputs) for the entire test sequence are reduced to a single code word which may be compared to the correct coded test response. A test response code checker is provided on each unit (chip) to be isolated. Pseudorandom test pattern generation and response coding are very economically implemented with CCD shift registers, and will constitute less than 0.2 percent of first and second layer chip logic.

Pseudorandom test patterns are generated on each first layer chip by feedback shift register hardware as shown in Figure 3.2-10. A shift

---

[*]N. Benowitz, D.F. Calhoun, G.E. Alderson, J. Bauer, C.T. Joeckel, "An Advanced Fault Isolation System for Digital Logic", IEEE Trans Computers Vol. C-24 No. 5, May 1975, p. 489-497.

16-BIT PSEUDO-RANDOM PATTERNS
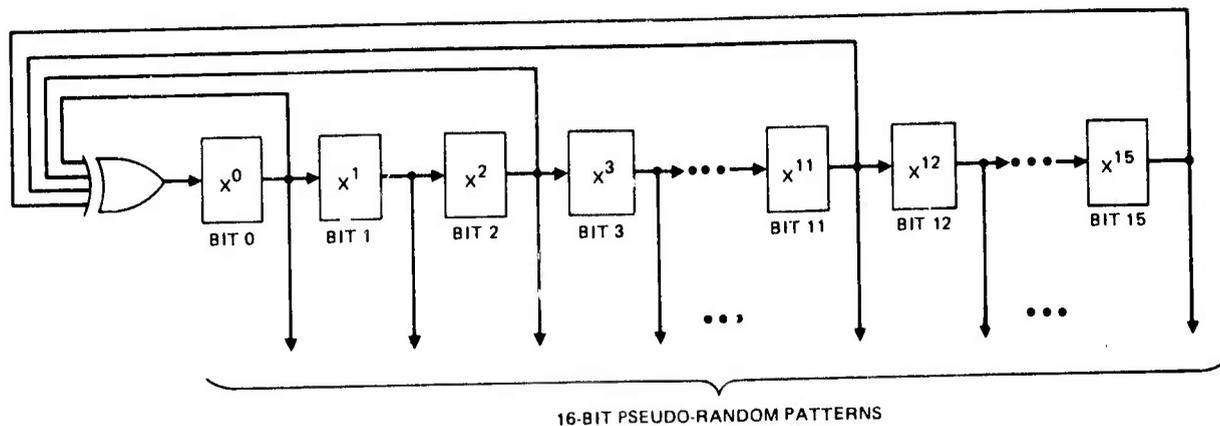
GENERATOR POLYNOMIAL: $1 + X + X^3 + X^{12} + X^{16}$ (16 BITS)

Figure 3.2-10.   Pseudo-random pattern generator.

register of N bits can generate up to $2^N - 1$ pseudorandom patterns of fixed sequence.  Pseudorandom patterns will efficiently and thoroughly test the arithmetic, shift register memory, and select logic implemented in first and second layer chips.  Remaining test inputs will be provided by global control.  Global control inputs to the chips will be varied during the test to verify operation of all globally controlled chip functions.

The patterns thus generated circulate through first and second layer chips.  For isolation purposes, feedback from second to first layer is disabled.

Each first and second layer chip contains a response code generator. The cyclic code generator shown in Figure 3.2-11 is ideally suited to CCD shift register implementation.  All chip outputs are serially entered into the code generator for each test pattern.  The cyclic code checker codes its input data stream by considering this binary data to be a polynominal and dividing it by a polynomial implemented in code checker hardware.  The final code word is the remainder of the division.  Any errors in the data checked, including multiple bit errors, will be detected unless the remainder of the erroneous data stream is the same as that produced by the good data stream. Nearly all erroneous outputs will be detected with only $(1/2)^{N-1}$ of errors undetected for an N bit division polynomial.  The 16 bit cyclic code checker shown in Figure 3.2-11 will detect 99.93 percent of erroneous test responses.

3-14

PRIMITIVE POLYNOMIAL: $1 + x + x^3 + x^{12} + x^{16}$ (16 BITS)
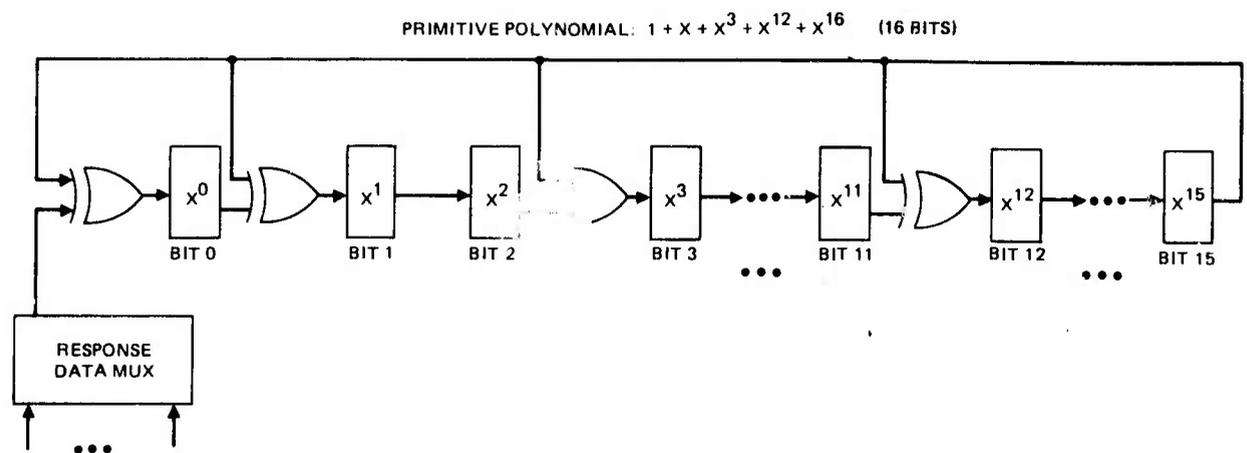
Figure 3.2-11.   16-bit serial cyclic code pattern checker.

If feedback from other chips is eliminated, examination of each chip code response serves to isolate faults to one chip.  As shown in Table 3.2-1, a "fail" first (or second) layer chip test result directly indicates the failed chip if the corresponding second (or first) layer chip has passed the test. If both a first layer chip and its associated second layer chip tests are failed either 1) a first layer chip failure may have propagated erroneous data into a correctly functioning second layer chip, or 2) both first and second layers chips may be faulty.  A second test (Test 2) may then be performed, switching the second layer chip to a known operational first layer chip.  This test will indicate whether both chips or the first layer chip only were faulty.

Upon detection of faulty chips, fault tolerance is provided by electronically substituting redundant operational chips for those which have failed. This is accomplished by fixed interconnection of redundant chips in the first and second layers; e.g., one redundant chip per 4 x 4 array serving an MFPA chip.  As shown in the example of Figure 3.2-12, any faulty chip may be replaced by the chip below it.  Each chip below the faulty chip is switched to handle the processing normally performed by the chip above it, with the redundant chip handling computations normally performed by the last chip.

An additional redundancy feature is the ability to perform degraded accuracy computations in the event of second layer failure.  Here the feedback from the second to the first layer chip is disabled and first layer data substituted for area correlated second layer data.

3-15

## TABLE 3.2-1. ISOLATING TO A FAULTY POINT TARGET PROCESSOR CHIP

|        | First Layer Chip | Second Layer Chip | Faulty Chip |
|--------|------------------|-------------------|-------------|
| Test 1 | Pass | Pass | None |
|        | Pass | Fail | Second Layer |
|        | Fail | Pass | First Layer |
|        | Fail | Fail | Examine Second Test Result |
| Test 2 | - | Pass | First Layer |
|        | - | Fail | First and Second Layer |
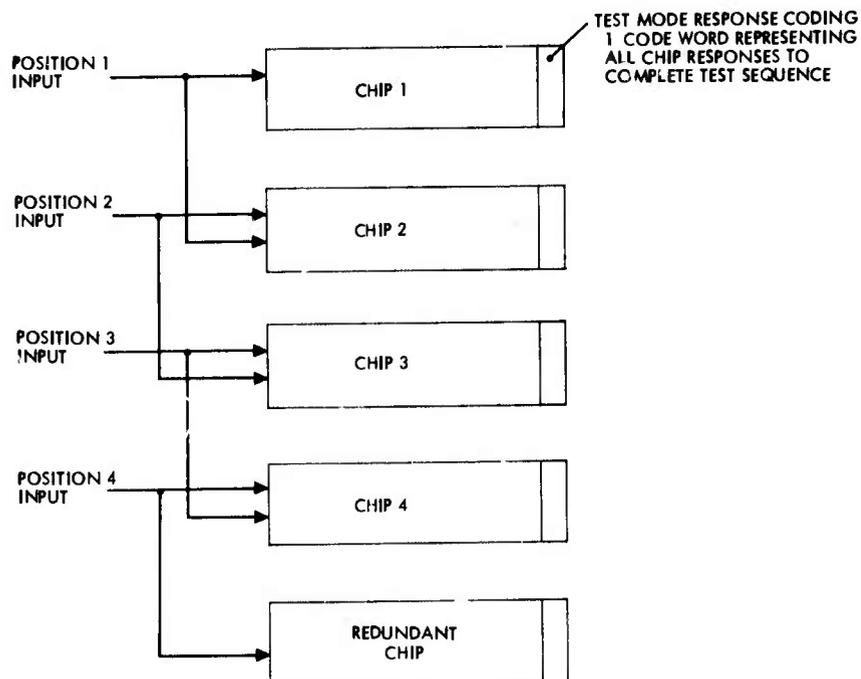


Figure 3.2-12.   Point target processor fault tolerance.

## Hardware Estimates for First and Second Layer

Table 3.2-2 shows the hardware required in a first layer chip capable of performing all time integration, change measurement, and adaptive threshold functions in parallel for fast and slow targets. One chip provides all first level processing for 1024 pixels. The total chip area requirements are within projected capability of 250,000 shift register bits per chip. Only 7 input/output

TABLE 3.2-2. FIRST LAYER HARDWARE REQUIREMENTS
(First Layer Chip to Process 1024 Pixels)

| Processing Function | Shift Register Memory | Memory Bits | Full Adders for Parallel Operation |
|---|---|---|---|
| Input Buffer | 32 x 16<br>1K x 16 | 16,896 | - |
| Time Integration 1 | 1K x 20 | 20,480 | 20 |
| Change Measurement (Fast Targets) | 1K x 16<br>1K x 16<br>1K x 16 | 49,152 | 32 |
| Change Measurement (Slow Targets) Time Integration II | 1K x 19<br>1K x 19<br>1K x 19 | 58,368 | 64 - 76 |
| Adaptive Threshold each of 2 units | 1K x 1<br>1K x 13-17<br>1K x 13-17 | 27,648-<br>35,840 | 21 - 29 |
| Self Test | 1 x ~12<br>1 x ~16 | ~28 | 4 - 6 |
| Total for Each Identical First Layer Chip<br>Tactical (non control) I/O | | 200,220-<br>216,604<br>5+2 | 162 - 192 |

This total is within currently projected CCD capability of 250,000 shift register bits per chip.

lines are needed (including fault tolerance provisions) in addition to control line(s), power ground, and clock lines. A 32 x 32 pixel array offers symmetry; however, a 128 x 8 array on each chip reduces the second layer chip I/Os.

Table 3.2-3 summarizes the hardware required for alternate second layer chips to handle area correlation of 1024 and 2048 pixels, respectively. The number of adders required depends upon the degree of parallelism required. Numbers shown are for fully parallel operation. I/O requirements for tactical

TABLE 3.2-3. SECOND LAYER HARDWARE REQUIREMENTS

(Second Layer Chip to Process 1024 Pixels)

| Processing Function | Shift Register Memory | Memory Bits | Full Adders for Parallel Operation |
|---|---|---|---|
| Area Correlation Burst Location | 1K x 16 4 words | 65,536 | 16 x 8 = 128 |
| Self Test | 1 x ~16 | ~16 | 2 - 4 |
| Total for each identical second layer chip | | 65,552 | 130 - 132 |
| Tactical (non control) I/O | | 13 + 9    or | 6 + 3 |
| Second Layer Chip to Process 2048 pixels | | | |
| Area Correlation Burst Location | 1K x 16 4 words 2 memories | 131,072 | 16 x 8 x 2 = 256 |
| Self Test | 1 x ~16 | ~16 | 2 - 4 |
| Total for each identical second layer chip | | 131,088 | 258 - 260 |
| Tactical (non control) I/O | | 13 + 9    or | 6 + 3 |
| These totals are within projected CCD capability of 250,000 shift register bits/chip | | | |

signals are reduced from 22 to 9 (including fault tolerance) if a 128 x 8 (or 64 x 16) array is used in place of a 32 x 32 pixel array.

Target Tracking

Detected targets are first assigned to a track file which is maintained by a tracker processor. Periodically each track file requests data on its assigned target to update its track parameters. It is assumed for the purpose of analysis that each track file has access to all pixels of data in the first layer of the point target processor. Furthermore, a maximum of 5000 targets (including false trial tracks) must be tracked simultaneously. To enable a preliminary design of the system it is estimated that each tracker processor is capable of handling 10 track files, thus requiring that a total of 500 tracker processors be provided. The medium sized system ($4 \times 10^6$ pixels) which was used for design purposes requires that the track files obtain data from 4000 first layer subarrays. Each subarray contains data for 1024 pixel elements. Since 500 processors must have access to 4000 subarrays during each frame, a communication network must be provided.

The most straightforward design approach is to provide a single bus as shown in Figure 3.2-13. The word transfer rates ( 1 MHz) demand that separate parallel 16-bit buses be provided for the transfer of target addresses to the subarrays and the return of data to the trackers. The basic configuration of each bus network is shown in the figure and includes a data selector array, a bus driver, a bus receiver, a fanout buffer, and a central bus controller. The data selector determines which track file request (or return data) is placed on the bus. The data is transmitted over a 16-bit parallel bus via drivers and receivers. The fanout buffer distributes the data to all subarrays (or tracker processors) so that the appropriate one can identify and receive it. The bus controller is responsible for controlling and sequencing all operations.

An alternate approach to the design of the tracker communication network is a multilevel bus. An example of such a network is given in Figure 3.2-14. The network is composed of several levels of bus elements which form a sort of matrix. The elements in each level are connected only to those in the next level. All buses are serial to reduce the interconnections.

3-19
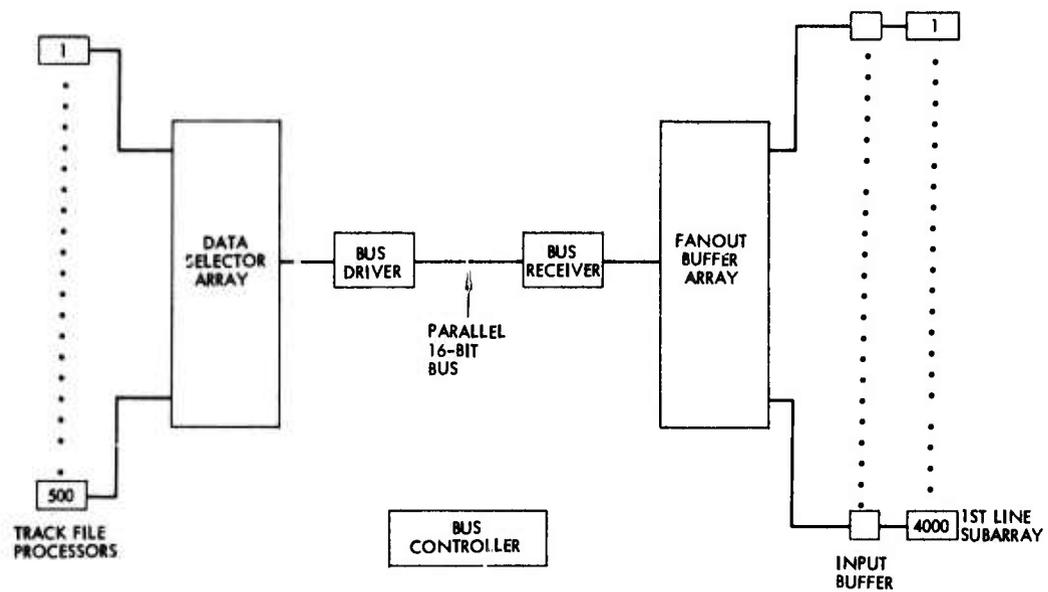
Figure 3.2-13.   Conventional parallel bus approach.
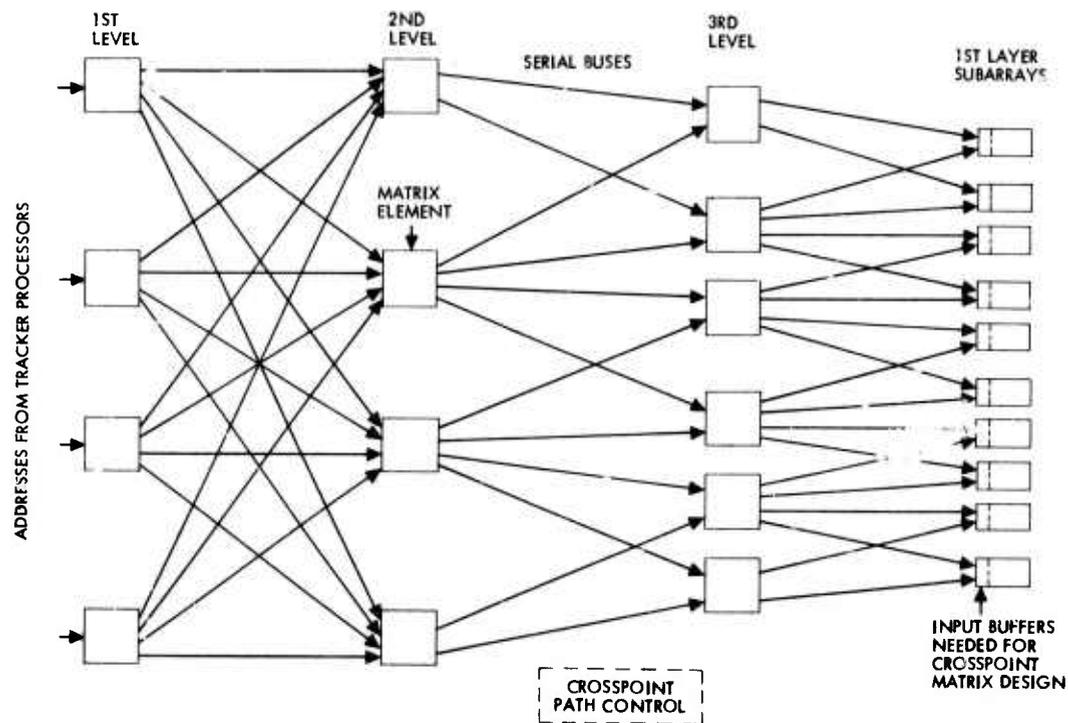
Figure 3.2-14. The multilevel bus approach.

Redundant connections are easily provided to increase the fault tolerance and provide multiple access paths to all subarrays. The latter feature prevents overloading of a subset of buses when target clustering occurs in a small number of subarrays. It also reduces the data rate through any one bus so that serial buses can be used. The bus elements may be either crosspoint switches or store and forward units. A separate multilevel bus network is needed for both the address and return data buses. The number of bus elements required for this type of network is dependent upon the following factors: number of tracker processors, number of first level subarrays, worst case target clustering, maximum number of I/Os permitted per chip, and the amount of redundancy desired.

Table 3.2-4 compares the characteristics of the conventional parallel (CP) and multilevel serial (MS) bus approaches for the design of the tracker communication network. A CP bus design that contains no redundancy requires

**TABLE 3.2-4. COMPARISON OF CONVENTIONAL PARALLEL AND MULTILEVEL SERIAL APPROACHES**

| Conventional Parallel Bus | Multilevel Serial Bus |
|---|---|
| The parallel 16 bit configuration is needed to satisfy the data rates set by the update rate and the number of track files. | Due to the number of identical network elements serial data buses will satisfy the required data rates. |
| To achieve fault tolerance redundant units and multiple parallel buses must be provided. Thus a significant hardware increase is necessary | Fault tolerance can be readily incorporated into the basic network configuration at the expense of nominal extra hardware and interconnections. |
| Input buffers on the subarrays must be provided to handle nonuniform target distribution. | The basic design of the network takes into account nonuniform target distributions. |
| A non-redundant design requires approximately 5000 chips for a $4 \times 10^6$ pixel system. | A fully redundant design requires <5000 chips. |
| 6 chip types are required. | 3 chip types are required. |
| Approximately 200,000 interconnections are needed for a non-redundant design. | Approximately 25,000 interconnections are needed for a fully redundant design. |
| The chips are mostly low to medium complexity. | The chips are medium to high complexity. |
| The functional design utilizes conventional techniques. | In order to optimize the design simulation should be used. |

greater than 5000 chips for a $4 \times 10^6$ pixel system. A MS bus design that is fully redundant requires approximately 3500 chips. A less conservative design requires only half that amount (1750 chips). Two criteria were used in obtaining the chip estimates for both design approaches. The maximum chip complexity was limited to 1000-1500 gate equivalents, and the maximum number of chip I/Os was restricted to 20. (If 2200 maximum targets are assumed for the medium sized system instead of 5000 the following chip estimates for a MS bus design are obtained: 3000-3500 chips for a conservative design and 1500-1750 chips for a less conservative approach.) The

CP bus approach that contains no redundancy requires approximately 200,000 interconnections to connect all the chips, compared to 25,000 interconnections for a fully redundant MS bus design. The chips for the CP design are expected to be of low to medium complexity while those of the MS bus approach are of medium to high complexity. The CP bus approach utilizes conventional design techniques, but in order to optimize the MS bus design a simulation should be used. The simulation program would allow an efficient means of trading off different designs and evaluating the effect of the numerous variables involved. Clearly though, the MS bus approach offers significant advantages over a CP bus design.

A store and forward (S&F) unit provides the same function, but also requires buffer memory and decision logic. A comparison of these two approaches is given in Table 3.2-5 and indicates that, for several reasons, a S&F unit is the superior network element for the design of the multilevel bus tracker communication network.

## Global Control

Global control is used because of its suitability for control of large numbers of identical processing chips operating in parallel and performing the same computation. Global control saves repetition of control logic on each of the processing chips. The global control also can be efficiently implemented with $I^2L$ logic, possibly in the form of 3 or more parallel computers resembling the tracking processors. Local control could result in a less efficient implementation of random control logic gating or read only memory on CCD devices not well suited for these functions.

Global control also provides an intelligent decision making capability which may be dangerous to place on processing chips not protected from chip failure by replicated (e.g., triplicated) logic. It provides a central location for global decision making, system self test, fault tolerant reconfiguration, and task (tracker) assignment.

## TABLE 3.2-5. COMPARISON OF CROSSPOINT AND STORE AND FORWARD APPROACHES

| Crosspoint Switch | Store & Forward |
|---|---|
| Complete path must be established before data can be sent. | Data is sequentially transferred from level to level, thus only one path segment must be available to transfer data to the next stage. |
| A controller is needed to determine the best path of those available. Control of every crosspoint switch is necessary. | Each unit is independent and does not require central control. |
| This approach bottlenecks faster with clustering of target data because complete dedicated paths are necessary for data transfer. Thus more units per level and more connections are required to maintain same data rates. | This approach tolerates greater clustering of target data. Fewer units per level are thus required. |
| The basic unit is a hardware switch that must be controlled by a central control unit. | The basic unit requires more circuitry (however fewer are needed). More flexibility and control are provided. |
| Nonuniform location of targets will cause some time slots to be overloaded. Thus a timing controller must be provided. | Overloading of time slots is prevented by buffering and a time associative queue in each unit. |
| Input buffers for the subarrays are required due to the non-uniform distribution of targets. | Subarray input buffers are not required because each store and forward unit contains a buffer memory. |
| Allows checking of data at destination only. | Allows checking of data at each level of transfer with retry capability. As a result greater fault detection is possible. |
| Statistical design necessary. | Statistical design necessary. |

## 3.3  PROCESSOR B

This section describes processor concept B, the configuration produced by the second of the two independent teams.

### APSP Block Diagram

As shown in Figure 3.3-1 the APSP can be considered to consist of a Signal Processor followed by a Track Processor. The Signal Processor enhances the signal by detecting pixels whose spatial and temporal characteristics indicate the presence of a possible target. Such pixels are referred to as hits, which are correlated over many time periods by the Track Processor. The Track Processor creates files of correlated hits called track files, which contain the address and intensity of each hit. Completed track files may be transmitted to the ground link or be further processed by the spacecraft computer.

### Signal Processor

As shown in Figure 3.3-2 the signal processing function is accomplished by temporal and spatial filtering, and merging of the blur circle. At each step additional clutter is rejected and the data rate is reduced. The MFPA chip must be clocked at varying rates such that it is operating within its dynamic range. The usable frame rates at the MFPA chip are in the range of from 10 to 100 frames per second. MFPA samples are then converted to 10-bit digital words and sent to the temporal filter for further processing.
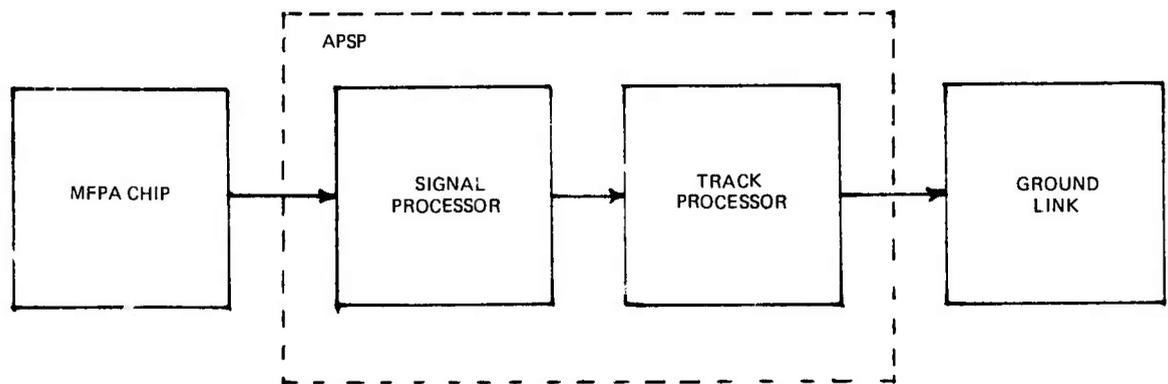


Figure 3.3-1.  APSP block diagram.

3-25
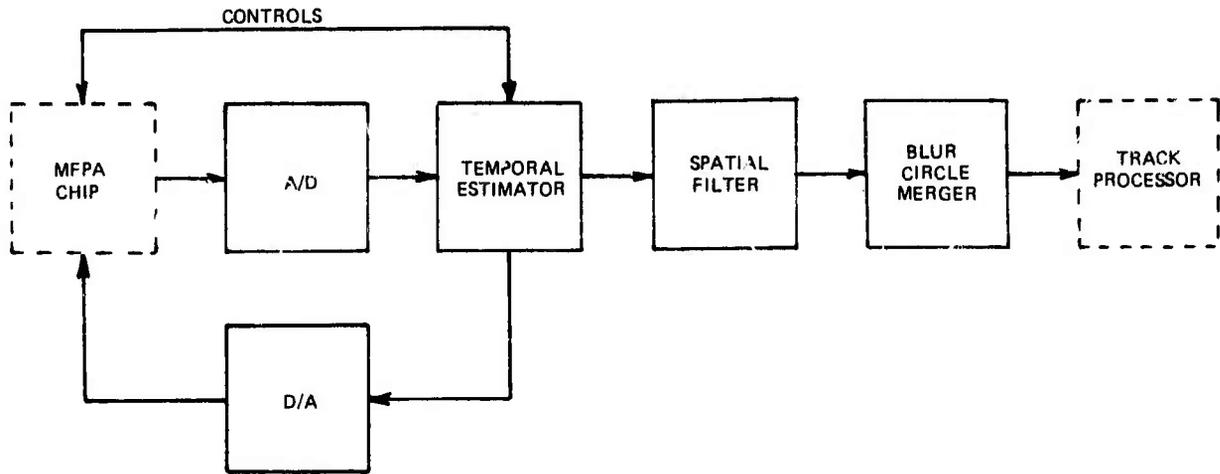
Figure 3.3-2. Signal processor block diagram.

## Adaptive Video Encoder

As discussed in Section VII 3.0, the temporal estimator predicts the intensity of each pixel in the next frame based or a weighted sum of past sample values. The predicted values, $\hat{q}_{ij}$, are converted back to analog form and sent to the MFPA chip. On the MFPA chip, fat zero control is used to subtract $\hat{q}_{ij}$ from the present measured value, $q_{ij}$. The result, $\Delta q_{ij} = q_{ij} - \hat{q}_{ij}$, is the value sent to the A/D converter. The estimator is designed to predict slowly changing clutter and to have a very limited response to moving targets. Ideally, a non-zero $\Delta q_{ij}$ corresponds to the intensity of a target. In addition the temporal estimator performs gain normalization to compensate for frame rate changes and variations in the responsivity of individual detectors. The estimator produces data at a rate equivalent to 100 frames/sec.

## Spatial Filter

Spatial filtering is performed on the data from the estimator to further reduce clutter, and is accomplished by comparing each pixel with its neighbors. Blur circle merging is closely related to spatial filtering. The significance of blur circle merging is described later.

## Signal Processor Architecture

This section describes the signal processor and related items. The two basic areas are the Monolithic Focal Plane Array (MFPA) and the Adaptive Video Encoder (AVE). The functional requirements are as follows:

1. An estimation of the pixel output must be produced based on knowledge of previous outputs.

2. Because the cells on the detector chips are unique in their response to identical inputs, their outputs must be normalized.

3. The change in observed data-versus-predicted data will be presented to the Spatial Filter at a constant rate corresponding to a frame rate equal to or less than 100 F/s (Frames per second).

4. The signal processor must suppress phenomena which cause any of the detector cells to saturate. (Reference Section VII 3.0).

5. The signal processor must acknowledge a valid saturation of a cell or cells and modify the frame rate to remove the cells from saturation such as laser countermeasures. (Reference Section VII 3.0.)

The relationship of these functions is shown in Figure 3.3-3.

## Estimation

The estimator is a temporal filter which makes predictions based upon the past history of a cell. This could be performed in a number of ways. The scheme chosen uses a finite number of past values, along with weighting coefficients, to obtain a prediction that will best locate targets at their earliest appearance. The estimation in weighted sum form is:

$$\hat{q}_{t_{n+1}}(i, j) = \sum_{k=o}^{m} a_k \, q_{t_{n-k}}(i, j)$$

where $(i, j)$ are the coordinates of the cell on a chip and $m$ is chosen arbitrarily to be 4. (Reference Section VII 3.2 on the programmable predictor.) The weights, or gains, $a_i$, will be supplied upon further analysis of this type of estimation scheme.
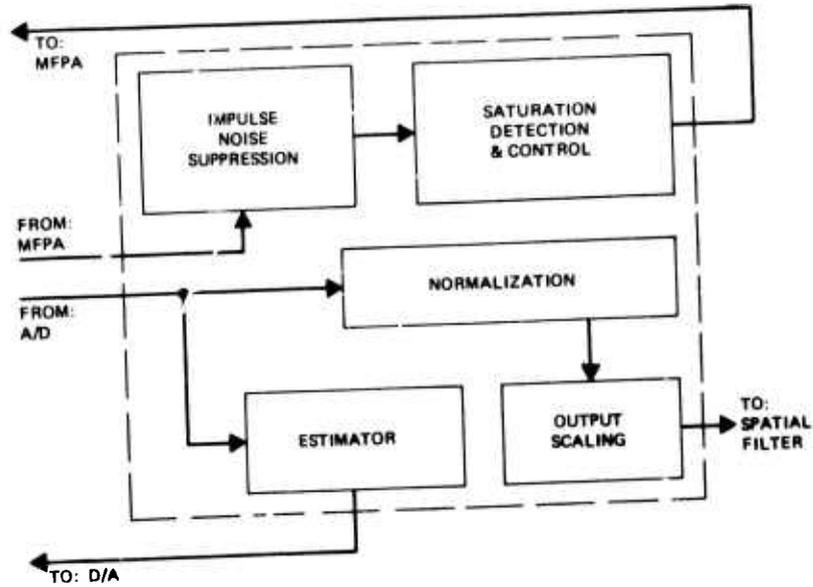
Figure 3.3-3. Temporal filter.

The estimate is then passed through a D/A converter and compared to $q_{t_{n+1}}$ (i, j). The difference (positive or negative) is then digitized and added to the estimate to yield the digital value of $q_{t_{n+1}}$.

Hardware Implementation

To implement this, the maximum data rate must be determined. On a per chip basis: each chip has (128)(128) = 16,384 cells, and the assumed highest frame rate is 100 F/s. Thus a maximum data rate of (100)(16,384) 1.6 MHz is possible. The maximum data rate of 1.6 MHz corresponds to 100 F/s. From this, it is seen that $\Delta q$ samples arrive every 625 nanoseconds. No technology that will allow five multiplications and the summing of five items to be performed at these speeds is foreseeable. Hence, a pipeline or parallel implementation is considered.

In this instance, a parallel scheme will be faster and utilize less hardware than a pipeline scheme. One implementation is shown in Figure 3.3-4 and operates in the following manner.

Step 1)   An estimate is obtained from EQ (Estimate Queue — a CCD memory) and shifted into the $\hat{q}$ register.

Step 2)   The value in $\hat{q}$ is sent to be compared with q from the MFPA.

Step 3)   The difference, $\Delta q$, is returned from the A/D and added to the estimate to obtain the actual value, $q_o$. This $q_o$ is placed into two registers.

Step 4)   a.   $q_O$ is multiplied by its associated gain and temporarily stored.

            b.   $q_O$ is shifted into the $q_{t-1}$ queue. The $q_1$ being shifted out goes into a register and also into the $q_{t-2}$ queue. The $q_2$ being shifted out also goes
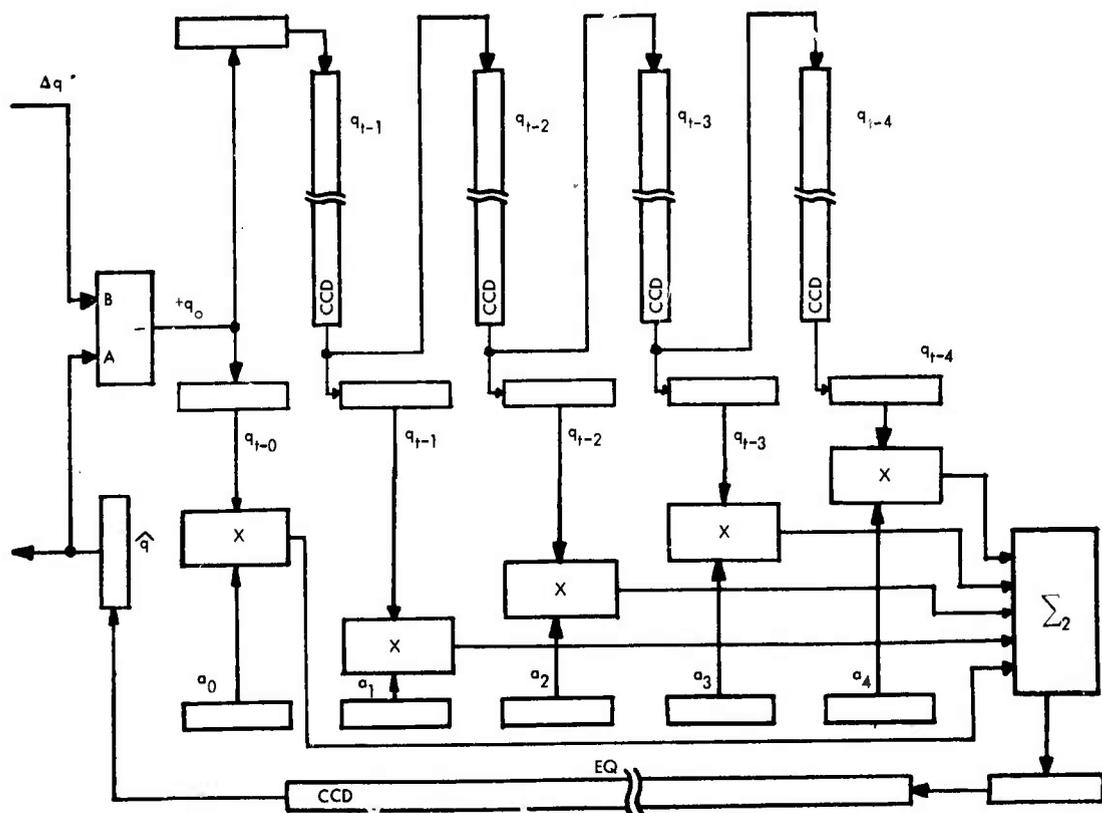


Figure 3.3-4. Mechanization of estimator.

3-29

into a register and into the $q_{t-3}$ queue. The values of $q_3$ and $q_4$ are handled in the same manner except that $q_4$ is not retained for future use.

Step 5) Once these values of $q_i$ are set up in their corresponding registers, four multiplications take place which produce the four quantities, individually: $a_1q_1$, $a_2q_2$, $a_3q_3$, $a_4q_4$. The coefficients $a_n$ are obtained from a ROM. (Reference Section VII 3.0.)

Step 6) After the above products are formed, they are summed along with $a_0q_0$ to yield the next estimate - $\hat{q}$.

Step 7) This estimate is now placed in the EQ to be used after the remaining 16K-1 cells have to be processed.

Notice that increasing the number of past values to use in the estimate is easily done by replicating the last cell as many times as needed.

### Gain Normalization

Gain normalization is performed such that to units beyond the temporal estimator, MFPA data appears to be uniform. One technique is to exercise the MFPA after its construction to determine which cell on the entire array yields the mean output for a constant input. This cell will then have a normalization factor of unity. The remaining cells will have normalization factors different from unity to adjust their outputs to correspond to the weakest cell. However, it is likely that these normalization constants will need to be modified due to changes on the MFPA during its life.

Since the quantity to be normalized is $\Delta q$, and $\Delta q$ is primarily a function of the estimator, the $\Delta q$'s will be directly multiplied by their corresponding normalization constants.

This implementation, along with the ability to update the constants, is shown in Figure 3.3-5. The operation is as follows:

Step 1) While the $\Delta q$ is being digitized in the A/D, its corresponding normalization constant is being shifted into a register and back into the CCD memory.

Step 2) These two values, $\Delta q_D$ and $n\Delta q$, are then transferred to two other registers to allow the next $\Delta q_D$ to immediately follow.

Step 3) $\Delta q_D$ and $n$ are multiplied to yield $\Delta q_{DN}$, a normalized value.

GAIN NORMALIZATION

$\Delta q_D$, FROM: A/D (10 B,T)

TO:
OUTPUT
SCALING

10 BIT X 10 BIT MULTIPLIER
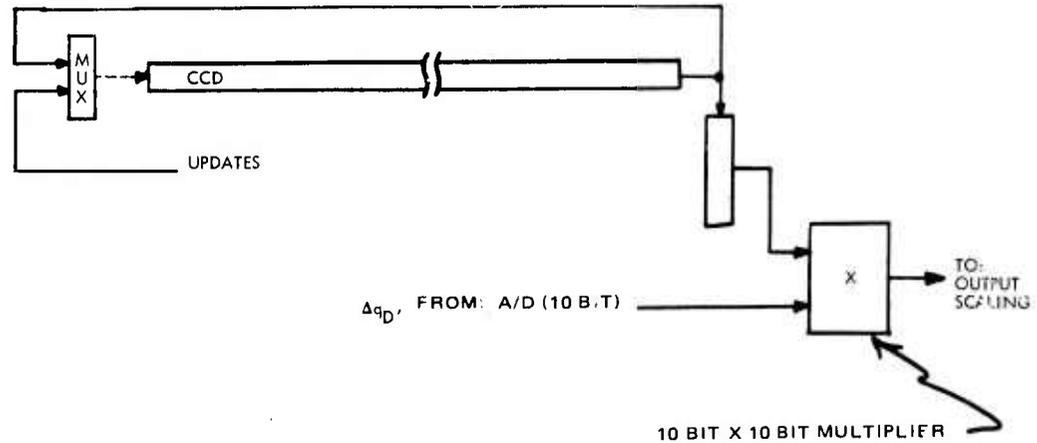
- CCD MEMORY:

$$(16 \text{ K CONSTANTS})(8 \text{ BITS/CONSTANT}) \left( \frac{1 \text{ CHIP}}{16 \text{ K BITS}} \right) = 8 \text{ CHIPS}$$

8 CHIPS

Figure 3.3-5.  Gain normalization.

It should be noted if the MFPA cell with the greatest output for a constant input were given a normalization constant of unity, the weakest cell would then have the largest constant associated with it, and the multiplication could possibly yield a number greater than 8 bits in length.

Along with "normal" operation, the updating of constants is performed via the mux "on top" of the CCD memory.  At the appropriate time, the select line is changed and the new constant shifted in to replace the previous constant.

### Output

The temporal filter output, $\Delta q$, (i, j) is to be presented to the Spatial Filter at a constant rate corresponding to the fixed frame rate.

Since the CCD detector chips are assumed to operate linearly (i.e., at an interval t a bucket will accumulate x photons and at t/2 it will accumulate x/2 photons), using frame rates greater than 10 that correspond to powers of 2 greatly simplifies the output problem. Thus we obtain:

| Frame Rate (F/s) | Multiplier for $\Delta q$ |
|---|---|
| 10 | 1 |
| 20 | 2 |
| 40 | 4 |
| 80 | 8 |

To better understand this simple scheme, let us examine the $\Delta q$'s at higher frame rates. First of all, $\Delta q$ may be positive or negative coming into the A/D. This implies that the estimator overshoots or undershoots the actual q of any detector cell. Over many frames the number of overshoots will equal the number of undershoots implying that the $\Delta q$'s over this range will add to zero, which is what is desired in the absence of targets. Thus, at higher frame rates, the probability of the sum of $\Delta q$'s equaling zero is high.

However, in the worst case, this sum could equal (approximately) $n\Delta q$, where n is given by:

$$n = \frac{\text{HIGHER FRAME RATE}}{10}$$

Beyond this, given an effective estimator, $n\Delta q$ (to a limit on n) should be much less than a target value. Therefore, taking one of the n estimates and multiplying it by n will yield a worst case difference but also cut the hardware to a minimum. Lastly, using n as a power of 2 allows the normalized $\Delta q$ to shifted by 1, 2 or 3 bits to affect the multiplication by 2, 4 or 8, respectively.

The implementation follows easily. Since we are assuming 4 frame rates, all powers of 2, we need only one register to hold $\Delta q_{DN}$ and a 4:1 mux to select the correct shift, as shown in Figure 3.3-6.
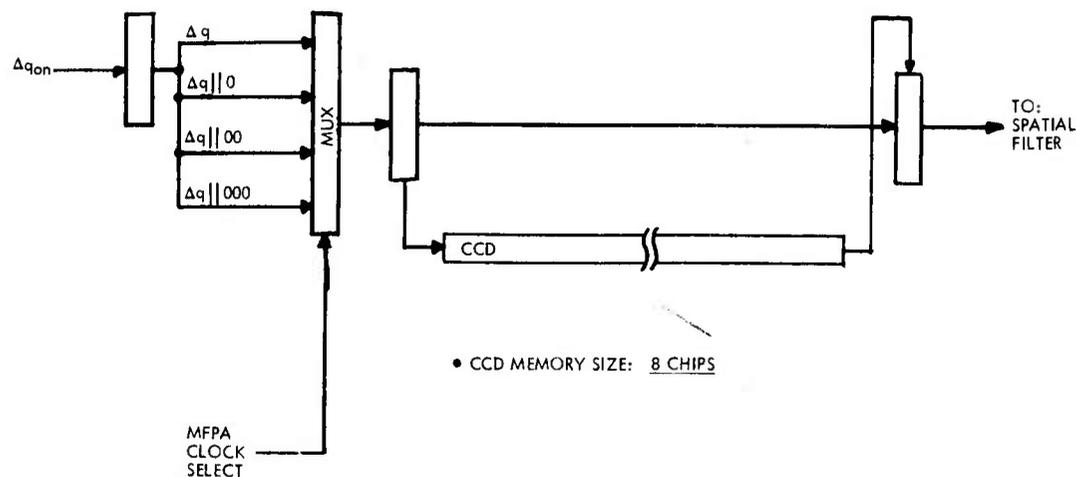
Figure 3.3-6. Temporal filter output scaling.

Two points should be noticed: 1) 2 inputs to $\Delta q_{DNC}$ register and 2) the mux select.

(1) Since data can enter at rates $\geq 10$ F/s, (Frames per second) the $q_{DN}$ register needs 2 inputs; one from the normalization process directly for the 10 F/s rate, and one from a queue in which normalized values are placed for rate $>10$ F/s.

The need for a queue results from the mechanization of handling the higher frame rates. Since a $\Delta q$ is selected and multiplied by a power of 2 to obtain the final $\Delta q$, the remaining N-1 samples are ignored. However, since the first set of samples is presented to the normalization process at rates $>10$ F/s, the normalized values need to be stored so that they can be presented to the Spatial Filter at a rate corresponding to 10 F/s.

(2) The mux select to provide the Spatial Filter with the properly scale data is the same select which operates the select on the clock mux for the MFPA described later.

Due to the effect of data entering the signal processor at $>10$ F/s and the manner in which this data is handled, it must also be equipped with logic to remember how many next $\Delta q$'s to ignore. This is implemented with simple counters.

3-33

## Impulse Noise Suppression

A problem arises when a cell in the MFPA saturates: is it caused by a target or was the cell ionized due to cosmic effects? At the highest tracking rate, 100 F/s, a target would have to have to be moving 1 mile in 10 ms, or 0.36 million miles per hour. Thus we can conclude the following:

CELL (i, j)

| Time | Value |
|------|-------|
| $t_{n-1}$ | Nominal |
| $t_n$ | Saturated |
| $t_{n+1}$ | Nominal |

From this we see that a cell saturating for one frame time is caused by something other than a target. This could be handled as follows: on a per detector chip basis, when one of cells is sensed to saturate, flag the signal processor. The signal processor then takes no action save remembering that a saturation took place. At the next frame time the signal processor looks for a saturation signal. If none arrives, the fact of the previous saturation is forgotten and processing continues normally. However if another saturation signal arrives a target could be present and hence the normal saturation control procedure is involved.

The implementation is a simple logic circuit that looks for 2 consecutive saturation pulses from an MFPA chip as shown in Figure 3.3-7.
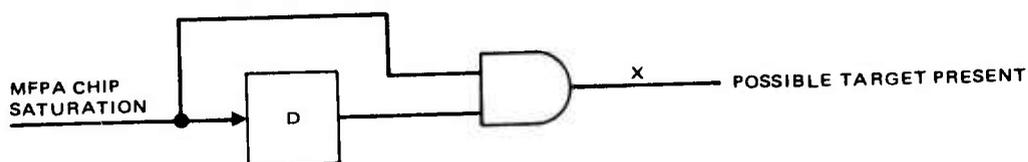


Figure 3.3-7. Impulse Noise Detection.

3-34

The circuit operation is trivial.  A pulse arrives at time $t_n$ and is remembered in the D F/F.  The line x goes active only if another saturation pulse arrives on the following clock, otherwise the first saturation detect is lost.

### Saturation Detection and Control

As was seen in the previous paragraph only 2 time-wise continuous saturation signals from an MFPA chip will cause any action to be taken.  The obvious action is to step up the frame rate of the MFPA.  As was seen previously, the ideal frame rates are powers of 2: 10, 20, 40 and 80 F/s.

Thus, the following approach can be used.  If a valid saturation occurs, select the next higher frame rate.  This will cause one of the following:

$$10 \longrightarrow 20 \ F/s$$
$$20 \longrightarrow 40 \ F/s$$
$$40 \longrightarrow 80 \ F/s$$

with valid targets being incapable of saturating a cell in the MFPA at 80 F/s.

However the reverse situation also exists:  the MFPA running too fast and needing to be slowed.  The saturation detection circuit can be used for this also.  By noting that a frame rate greater than 100 F/s is currently being exercised and that no valid saturations occur, the frame rate can be reduced (by reversing the arrows in the above table).

The implementation is straightforward and a simple technique is pictured in Figure 3.3-8.  The saturation controller looks for a valid saturation signal from the impulse suppressor and utilizes the following logic: if a signal is present, step up the frame rate; if it is not, possibly step down the frame rate.  The reason for possibly stepping down the frame rate is to prevent a thrashing type of operation between a frame rate that causes valid saturations and one that doesn't.

Thus valid saturations speed up the MFPA (and select the appropriately scaled output), and the absence of saturations will eventually slow down the MFPA.
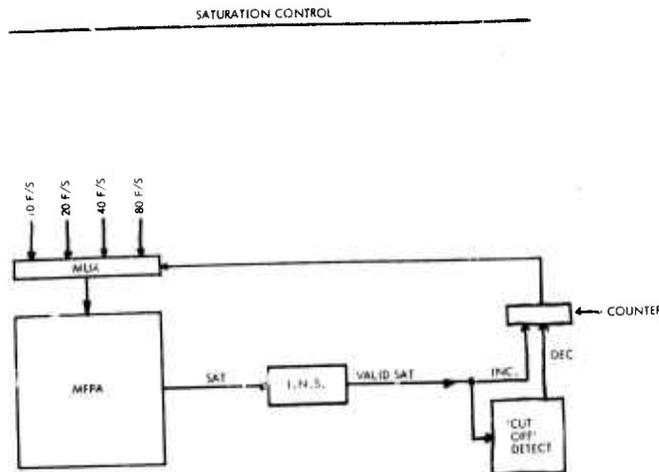
Figure 3.3-8. Saturation control mechanization.

Spatial Filter

The purpose of the Spatial Filter is to determine the locations of pixels which are illuminated by targets. A functional block diagram is shown in Figure 3.3-9. The unit receives sequential $\Delta q$ values for each pixel from the Temporal Filter and after processing this data reports pixel "hits" to the Trackers. The objective is to report the address of the single pixel which most closely represents a target's position. Additionally the difference in amplitude between the target pixel and the average of the adjacent pixels is reported along with the pixel address. To accomplish the above function the following two processes are employed:

1.   Four-direction adjacent pixel comparison

2.   Blur-circle merging

Adjacent Pixel Comparison. The adjacent pixel comparison process is illustrated in Figure 3.3-10. A three by three window centered about the candidate pixel is used to detect the presence of a target. The amplitude of a pixel's illumination is given by $A_{ij}$. If the magnitude of the candidate pixel $(A_{22})$ is greater than the average magnitude in all four directions, it is reported as a "hit."

3-36
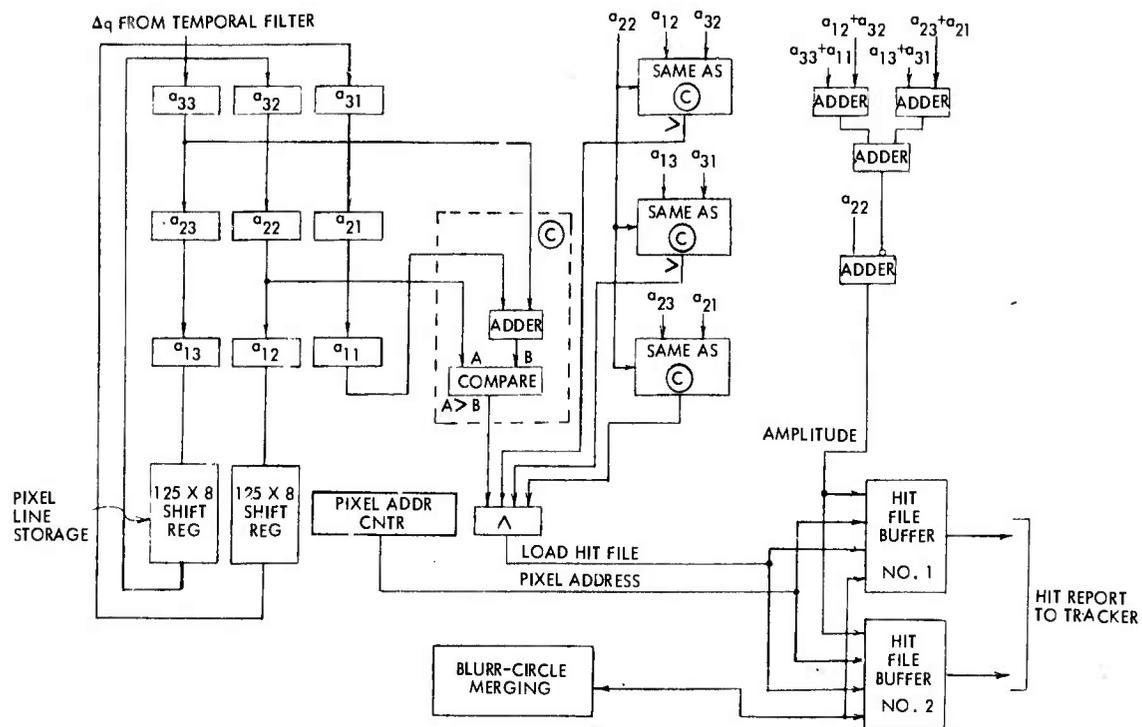
Figure 3.3-9. Spatial filter block diagram.

PIXEL Array

3 x 3 Window

Candidate PIXEL is $A_{22}$

| $A_{11}$ | $A_{12}$ | $A_{13}$ |
| $A_{21}$ | $A_{22}$ | $A_{23}$ |
| $A_{31}$ | $A_{32}$ | $A_{33}$ |
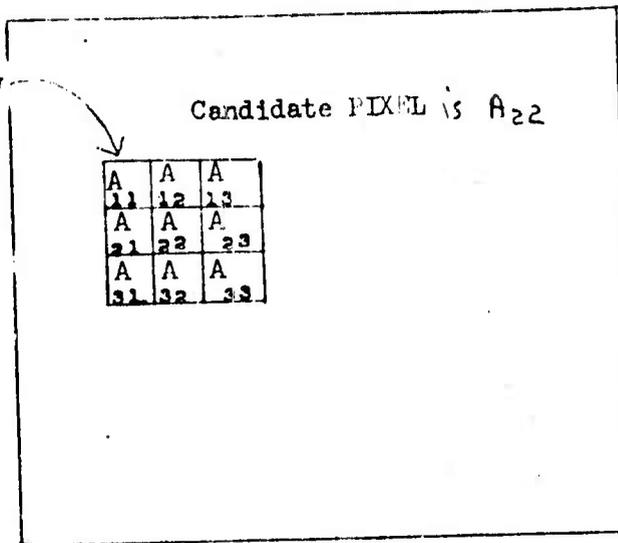
**If**  $[ A_{22} > \frac{1}{2} (A_{11} + A_{33})] \wedge$

$[ A_{22} > \frac{1}{2} (A_{12} + A_{32})] \wedge$

$[ A_{22} > \frac{1}{2} (A_{13} + A_{31})] \wedge$

$[ A_{22} > \frac{1}{2} (A_{21} + A_{23})]$

**then**  Amplitude $\leftarrow A_{22} - \frac{1}{9} [ \sum\limits_{j=1}^{3} \sum\limits_{i=1}^{3} Aij ]$

Add amplitude and pixel address to Hit file.

Figure 3.3-10.  Four direction adjacent
pixel comparison.

Two hit file buffer memories are provided to allow blur-circle merging to be performed concurrently with adjacent pixel comparison. Blur-circle merging is performed on the hit file generated during the nth scan cycle, stored in one buffer, while the hit file generated during the (n + 1)st cycle is being stored in the other buffer.

Blur Circle Merging.  A point target will appear on the MFPA blurred as a circle.  The size of the pixels has been selected to be equal to the blur circle for a point target, for optimum signal-to-noise ratio purposes.

3-38

Because pixels are of the same size as blur circles, a target will almost always be seen in more than one pixel at a time.



**very** rare         **rare**         **common**         **very** common
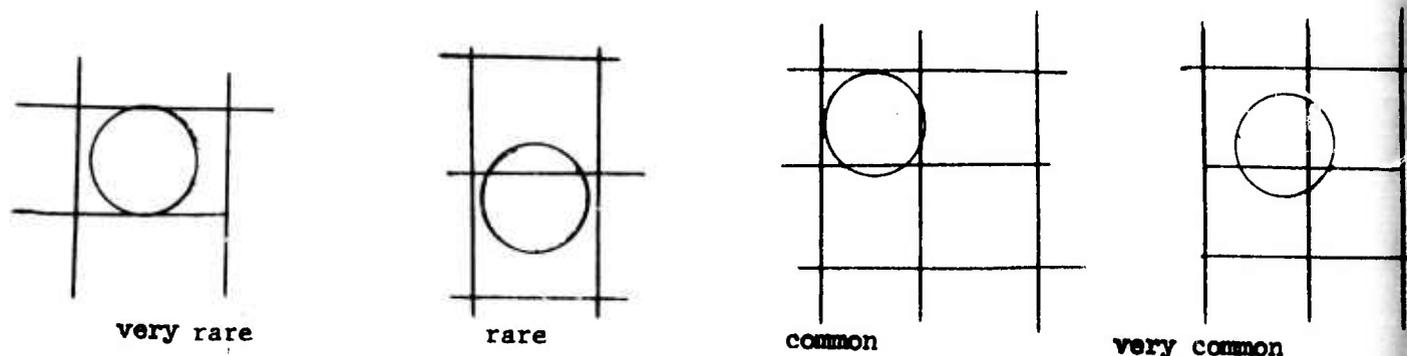
The adjacent pixel comparison process merges some instances of multiple illumination. However, as the blur circle center approaches pixel boundaries the adjacent pixel comparison process is unable to perform the merging and reports more than one hit. Therefore further processing of the hits is required. The adjacent pixel comparator loads the hit file buffer with one data item for each hit recorded. The blur circle merging algorithm identifies and examines clusters of hits. Using intensity and the shape of the cluster as deciding criteria, each clutter is merged into a single hit.

Tracking Processor

From a data processing point of view, tracking in the APSP consists of sorting the continuously incoming hit reports into track files and discarding those hits which do not appear to belong to any track.

The tracking function can be subdivided into the following tasks and subtasks:

1.   Track initiation:        a.   Recognizing potential target

                                    b.   Determining that it is not part of any track

                                      c.   Initiating a microprocessor

2.   Monitoring of a track:    a.   Update state vector at each frame

                                                      b.   Handle special conditions:

- Crossing of chip boundaries

- Missed measurements

- Bifurcations

                                                      c.   Produce track file

3.   Ending a track:           a.   Monitor kinetic properties of tracks

                                                      b.   Identify clutter

                                                      c.   Count missed measurements

                                                      d.   Terminate the tracking if:

- The track is clutter, or

- There are too many consecutive missed measurements

                                                    e.   Transmit track file to ground link.

Given the estimated number of hits per frame that need to be processed and the short time in which this processing has to be done, it is clear that some sort of parallel processing is required. In order to use simple, low power, identical processing elements and to provide the necessary computing power, array processing is best suited. Each processing element in the array has to be assigned a portion of the tracking task. This assignment, which is always the central issue in the design of array processors, is generally referred to as the resource allocation problem.

Another issue generally encountered in array processors, and one particularly acute in this application, is that of data flow. Therefore solutions of the resource allocation and the data flow problems are going to characterize the design of the array processor.

### The One Track per Processing Element Approach

This design enables the hits detected by the spatial filter to be broadcast to all processing elements over a bus. Each track currently being monitored has one processing element assigned to it. When the spatial filter

is broadcasting hits, each micro-processor looks only for hits falling within the tracking gate of the track it is monitoring. As consecutive hits falling into the tracking gate are acquired, a track file containing all the past history information of that track is synthesized.

Each processing element acquires from the bus only information pertaining to the immediate vicinity of the target. Most of the time this amount of information is sufficient to continue the tracking process. At times, however, global information is needed. A special processor, called the supervisor, is used for this purpose.

The supervisor determines which hits in each frame were not picked up by any processing element. All such hits are potential new tracks and an idle processing element must be assigned to each by the supervisor.

Whenever a processing element finds more than one hit within the tracking gate it, must determine whether this is a case of a track crossing a bifurcation, or a new track appearing. Since this decision requires global information, it will have to be made by the supervisor.

Whenever a processing element decides that the track it was monitoring has to be ended it notifies the supervisor which then deactivates that processing element and marks it as available.

Figure 3.3-11 shows the configuration of a processing element in the array. Data is drawn from the Hit-Bus. Among other things, each data item on the Hit-Bus contains sequentially the $(i, j)$ coordinate pair of the hit.

The Mousetrap is a programmable hardware device which when provided with $i_{min}$, $i_{max}$, $j_{min}$, $j_{max}$ will acquire all those hits from the bus whose coordinates $(i, j)$ fall within that rectangle. The data for such hits is passed to the microprocessor. When data items with $i > i_{max}$ appear on the bus, the Mousetrap notifies the microprocessor that no further hits will appear. The microprocessor then processes the hits. Hits falling within the gate will be acknowledged to the supervisor. Tracking information will be pushed onto the Track File Queue. A new gate is computed and the Mousetrap is programmed accordingly. Thereafter the microprocessor waits for new hits to be transmitted by the Mousetrap.
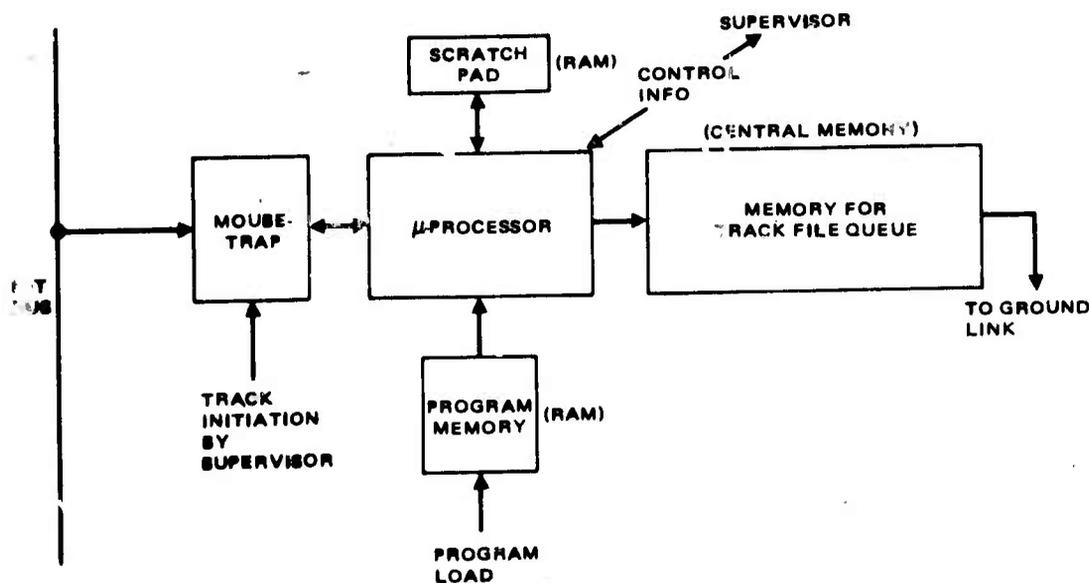
Figure 3.3-11. Processing element configuration.

The Track File Queue is a serial memory where the microprocessor pushes data in from one end while the ground link reads data off the other end. Data is always moving through this memory as through a pipeline.

The Program Memory can be loaded with programs and constants from an external system. Once loaded, it determines the behavior of the microprocessor. From the point of view of the microprocessor, this is a read-only memory.

The Scratchpad is a relatively small memory containing all variables used for tracking in this processing element. The foregoing discussion indicated that each active processing element processed only one track. In fact one microprocessor contains multiple track files, as described in Section 5 (Software) of this report.

This design has two important weak points: the supervisor appears to be very complex and the data rate on the bus is very high.

The data rate on the bus can be reduced by dividing the focal plane into a number of overlapping sections and assigning a separate bus to each.

Each mousetrap is then preceded by a multiplexor which selects one of the buses. The selection is determined by the microprocessor based on the section in which the gate is located. The data rate is cut down by a factor (approximately) equal to the number of buses employed. Thus the data rate on the bus can freely be traded off for added hardware.

Track Initiation Hardware

Hits reported by the spatial filter can be grouped into three categories:

    a.    The next position of the target track

    b.    Clutter that appears to be the continuation of a false track left by clutter

    c.    Hits that do not appear to be the continuation of any track.

The latter kind of clutter will cause a large number of tracks to be initiated and terminated at each frame time. This function represents the largest computational load on the supervisor. In order to accomplish these functions the supervisory task must be divided into several independent functions. Such a division allows for parallel processing at the supervisor level.

Figure 3.3-12 shows the hardware configuration for the track initiation and deletion functions. The control function is distributed throughout the array of mousetraps. Only two subfunctions are performed on a global basis: mousetrap distribution over the four buses and mousetrap chaining to determine the sequence in which mousetraps are assigned to hits.

When a track is terminated, the associated mousetrap is deactivated and placed at the end of the queue of idle mousetraps waiting for new hits. This process is performed in two steps: (1) determine which bus to monitor and (2) determine the number of mousetraps in the mousetrap queue. The first step is performed by a special hardware device which contains four counters containing the current number of idle mousetraps associated with
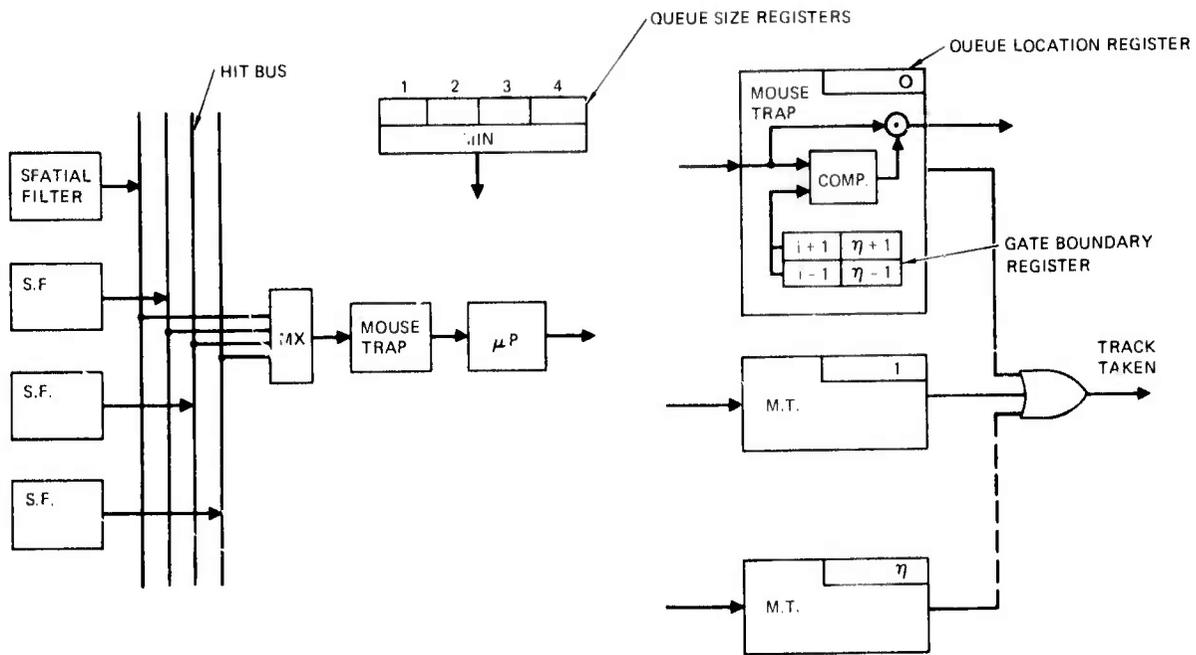
Figure 3.3-12. Track initiation and deletion hardware.

each bus and a logic unit monitoring the four counters and determining which bus has the least number of idle mousetraps. The output of this unit is used by a mousetrap, when it becomes idle, to determine which bus it should monitor. Thus the mousetraps are uniformly distributed over the four buses.

Each mousetrap contains a counter which indicates its position in the mousetrap queue. One additional counter is used to indicate the number of the next empty position at the end of the queue. When a mousetrap is in the process of becoming idle it transfers the contents of this special counter into its own position counter and increments the special counter by one. Mousetraps are assigned to new hits in the following fashion:

1. If a hit is picked up by an active mousetrap this action results in a pulse appearing on the hit taken bus.

2. If a hit is not picked up by any mousetrap the hit is assigned to the mousetrap at the head of the queue and all queue position counters are decremented by one including the end of queue counter.

3. When a mousetrap position counter equals 1 it places itself in the ready state. In this state all hits are picked up from the bus until a hit is assigned to this mousetrap as described in 2 above.

3-44

4.  When a new hit is assigned to a mousetrap the i and j values are placed in the gate boundary register. These registers are incremented and decremented appropriately to form a standard 9 pixel gate for the next frame.

It should be noted that the state of the mousetrap associated with each processing element determines the state of the microprocessor. If the microprocessor is available, then the mousetrap is in the idle queue. When the mousetrap becomes active and picks up a hit, the microprocessor is initialized to start a track.

### The One MFPA Chip per Processing Element Approach

With this design approach each MFPA chip has a dedicated processing element. That processing element monitors all hits and tracks within the MFPA chip. The Hit-Bus is thus eliminated and the resource allocation problem is solved a priori.

As shown in Figure 3.3-13 a bus is used to move the processed track file data to the respective memory. To reduce traffic on this bus, processing elements will not transmit track data pertaining to new tracks until the track has at least 10 valid entries. Track file data items are routed to the respective track file memory by means of an ID-number unique to that track. The assignment of ID-numbers and track file memories to tracks is still done by a supervisor but, since only tracks older than 10 frames receive such allocation, supervisor traffic is much lower.

This approach does not seem very promising because it does not offer dynamic allocation of processing elements and because it seems very doubtful whether fast enough processing elements will be available to process all the tracks that could appear on an MFPA chip in one frame time.

### Tracking Algorithms

In the designs presented, the tracking function is partially implemented in hardware and partially in software. For example the detection of hits on the Hit-Bus that fall within the current gate is a hardware function performed by the mousetrap. Computing the velocity and acceleration of the target is done by executing instructions from the Program Memory in the microprocessor
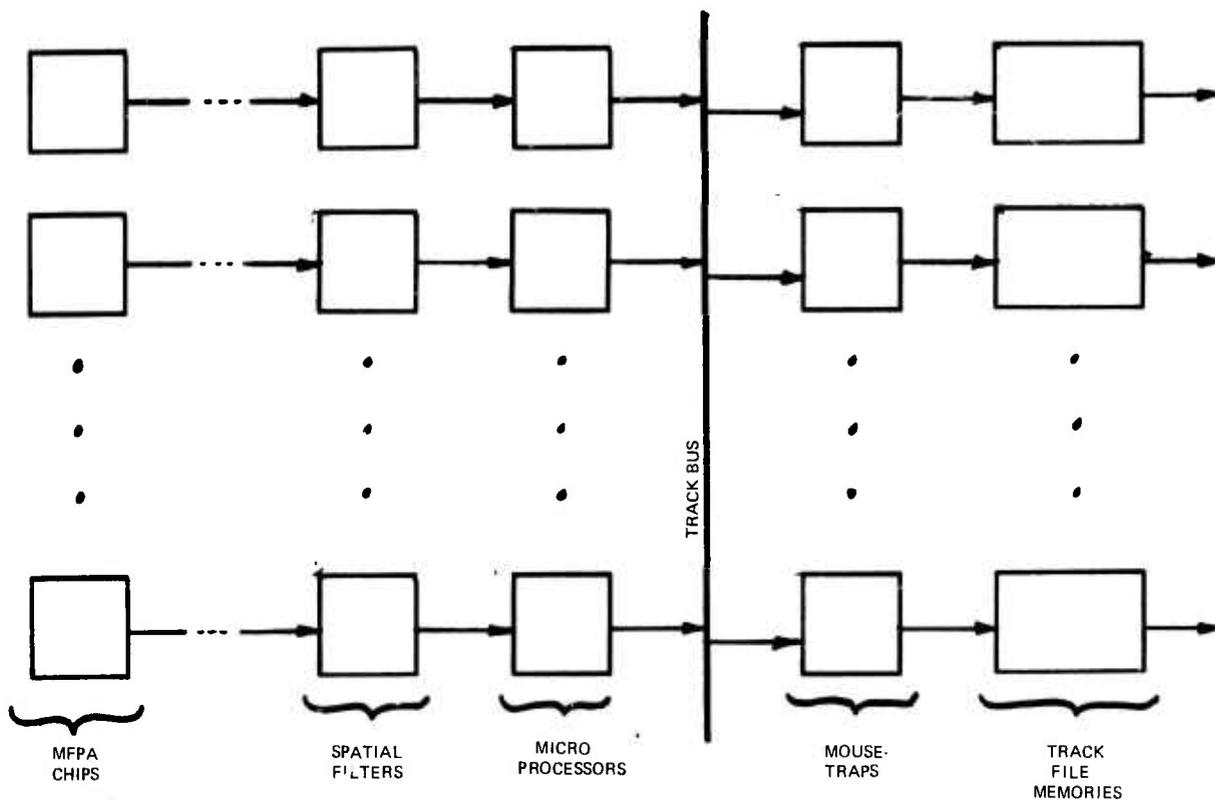
Figure 3.3-13. The one MFPA chip per processing element approach.

and is therefore a software function. Thus the term "tracker" as used in this discussion means a software algorithm.

The tracking software can be viewed as consisting of two independent functions. The purpose of one is to determine the search gate, set up the mousetrap and retrieve the hits from the mousetrap. It also services the Track File Memory.

The other software module is much more complex. It performs the following:

- Rejection of clutter tracks
- Selects hit when more than one appears in gate
- Monitors intensity
- Any other functions desired.

3-46

### Frame Rate

It is desirable to adjust the frame rate in such a way that even for the fastest moving targets $(i_{t+1}, j_{t+1})$ is an immediate neighbor of $(i_t, j_t)$. In other words the target moves at most one pixel on the grid during each frame. Therefore the frame rate will have to be adjusted as a function of the size of a pixel. The apparent velocity of the fastest expected moving target (in pixels/sec) is a parameter in the frame rate adjustment.

Based on the requirement that targets with apparent velocities from 0 to 70 pixels/sec have to be tracked, a range of frame rates between 10 and 80 Hz appears adequate.

Frame rates must be adjustable because too low a rate leads to large gate sizes, whereas too high a rate will cause excessive amounts of redundant tracking data to be output.

### Tracking Algorithm Requirements

Due to the adjustable frame rate, finding the target in the next frame is a relatively easy task.

Tracking will be performed based on the physical laws that govern accelerated motion. By monitoring velocity and acceleration in the state vector for a track it is possible to discriminate between targets and clutter. When clutter is monitored over a period of time it is probable that at some points its velocity will exceed the maximum expected velocity of a target or that its acceleration will surpass a set limit (e. g. , 5 g's).

The tracking algorithm does not have to predict the position of the target in the next frame. That can be done by simply searching the position of the target in the previous frame and the 8 immediately adjacent pixels.

Unlike $\alpha$-$\beta$ filters and Kalman filters which track the target by computing a weighted sum of its predicted and measured positions, tracking in APSP is based solely on measured positions. The reason why such a simple algorithm can be employed is that accurate and frequent measurements are available.

### The Effect of Gaps Between Chips of the MFPA

The frame rate will always be adjusted as a function of the size of the footprint of a pixel in such a way as to make the radius of the gate equal to one pixel or less for all manmade flying objects.

At the points where MFPA chips are joined a number of pixels are missing. For this reason the gate size will have to be increased at that point. The relation between gate radius (R) and gap width (G) is

$$R = G + 1$$

Basically the area of the gate is

$$A = \pi (G + 1)^2$$

Table 3.3-1 shows actual gate size (obtained by counting pixels) as a function of gap size.

TABLE 3.3-1.

| Gap | Maximum Gate Radius $r_{max}$ | $\pi r_{max}^2$ | Maximum Actual Gate Area |
|-----|-----|-----|-----|
| 0 | 1 | 4 | 9 |
| 1 | 2 | 13 | 21 |
| 2 | 3 | 29 | 45 |
| 3 | 4 | 50 | 69 |
| 4 | 5 | 78 | 97 |
| 5 | 6 | 112 | 137 |
| 6 | 7 | 152 | 177 |
| 7 | 8 | 202 | 241 |
| 8 | 9 | 255 | 293 |
| 9 | 10 | 314 | 349 |

Much of the gate area can be eliminated because the target could only reach certain points within the circle (gate) if its acceleration was very high (e.g., greater than 5 g's). It cannot be stated how much of the gate circle can be eliminated without knowing the footprint of a pixel.

Another way to cut down on gate area is to relate it to the speed of the target. The above relation for radius of the gate referred to a target moving at maximum speed (e.g., 70 pixels/second). If the target in fact moves slower, the gate radius can be decreased accordingly.

## 3.4 CONSOLIDATED ARCHITECTURE

After detailed examination of the two proposed architectures, the basic philosophy of the first was used and refined. It is this concept which is now explained. Figure 3.5-1 is a functional partitioning of the APSP. The following is a discussion of the tradeoffs and selection of the temporal filter, spatial filter, detection logic and the track processor.

### Temporal Filter

The filtering provided by the temporal detection filter (TDF) rejects slowly moving or stationary clutter edges while passing moving targets. The system performance is provided by the filter noise equivalent bandwidth and clutter rejection curves.

The filter design philosophy is to provide target detection on a per pixel basis via the hardwired TDF. The trackers use this information to generate track information. An adaptive temporal filter (ATF) is also used to provide refined temporal filter algorithms. For example, an accelerating target requires an ATF which maintains frequency "lock-on" as the target velocity changes.

Prior to filtering by the temporal filter processor, the target has been filtered by the optics and by the detector geometry. Both filters are low-pass filters. The corresponding blue size is approximately equal to the detector element size.
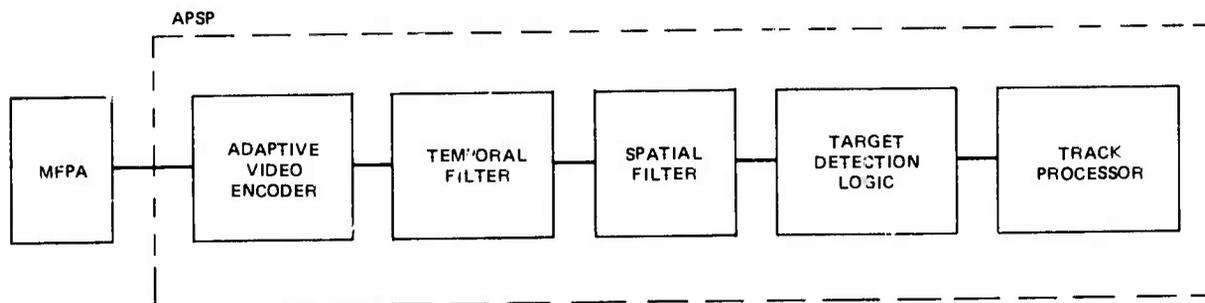
Figure 3.4-1. Functional units of APSP.

As shown in Section 6.2, the temporal detection filter needs to have at least third-order difference filtering capability in order to discriminate effectively against moving cloud edges. Higher order filters provide little performance improvement for the increased cost.

In addition, an effective variable frame time is required at the front end of the temporal filter to provide an optimum match to targets of varying velocity.

Figure 3.4-2 shows the temporal filter functional block diagram. The digital signals from the AVE enter first an N frame accumulator: N = 2, 4, 8, 16. The purpose of the integrator is to provide a match to a range of target velocities for a constant detector/mux array frame rate, nominally 10 Hz. For 5 pixels per second target rate, the 10 Hz frame rate is optimum. On the other extreme, for 0.3 pixels per second, a frame rate of 0.6 Hz is optimum, hence N = 16 frames of integration is required in the latter case.

The third order difference filter in a transversal implementation is also shown in Figure 3.4-2. Three frames of memory, 160K words each, and four multipliers and an adder are used in this implementation. A recirculating recursive filter uses less memory but it may have high round off errors.
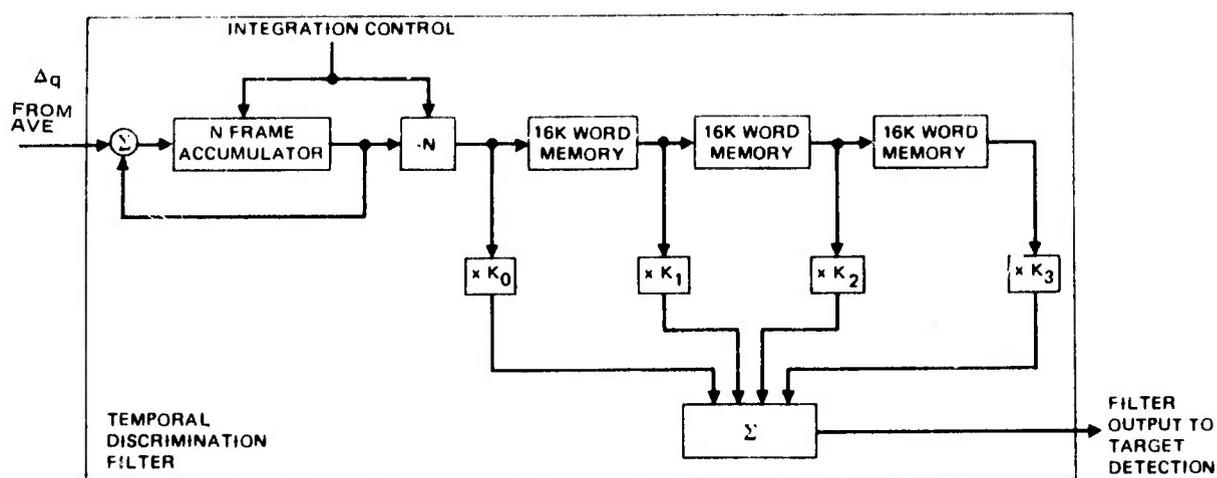


Figure 3.4-2. Temporal discrimination filter block diagram.

3-51

## Adaptive Temporal Filter

Targets that have been acquired by the tracker will be further processed in the Adaptive Temporal Filter (ATF). The principal utility of this processor is to increase SNR, clutter rejection and target location capability. A priori information about the target location and predicted state is furnished by the tracker to the ATF. This results in the detector/mux array being partitioned into smaller arrays around the present target pixel position. Software filtering algorithms can be utilized in this small subarray, resulting in enhanced system performance while properly allocating the track processor resources. A preliminary algorithm is discussed in Section 5.2.

## Star Discrimination Filter

The sensor which looks above the horizon (ATH) has a high density of moving stars in the background. The situation is illustrated in Figure 3.4-3 where the stars are assumed to move at 2.9 pixels/sec for this example. The rate of threshold excessions is estimated to be less than 1 percent in the Galactic plane at a threshold of 5 watts/sr. (Ref. Table 5.4-3.)

A moving track gate is established on the basis of the known star velocity. Those targets which fall within the predicted position and intensity range are declared to be stars after several frames. These targets are then deleted from the track file. All other targets are classified as potentially
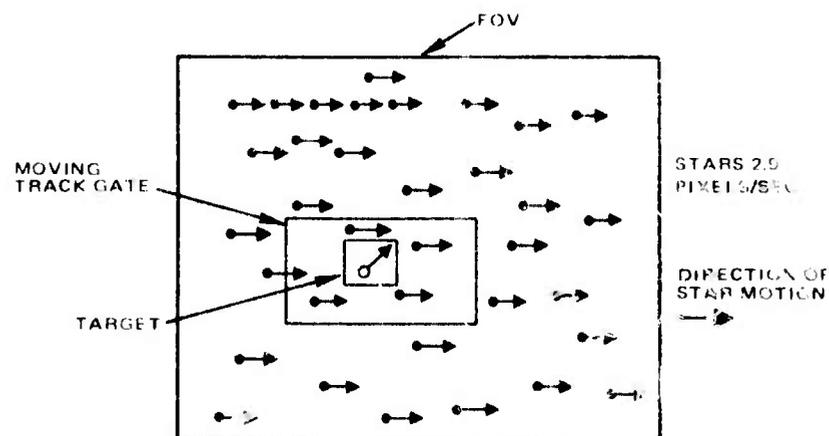


Figure 3.4-3. Star background discrimination.

acceptable targets and transmitted to the module processor for identification. A preliminary algorithm is discussed in Section 5.3.

## Waveform Discrimination Filter

A waveform discrimination filter can be implemented with the temporal filter illustrated in Figure 3.4-4. A first-difference transversal filter is followed by a simple threshold. When a signal of positive polarity is detected, the signal of opposite polarity occurring in the next two integration periods is clocked out and thresholded at about 80 percent with reversed polarity of the original threshold. As can be seen from comparison of an edge waveform and a point target waveform, the undershoot pulse for clutter is either missing, negligible, or greatly delayed (until the edge moves off the cell). This technique is effective in removing extended clutter from further processing and while retaining all targets within a wide range of target velocities.
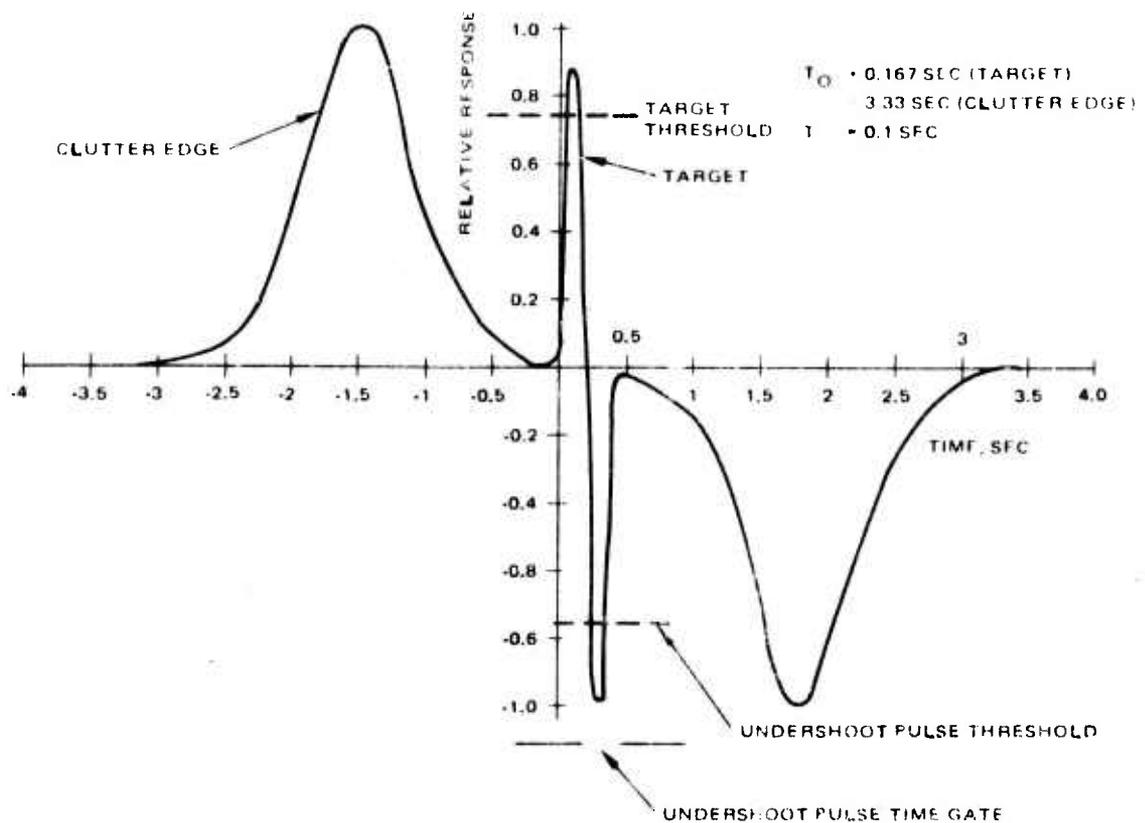


Figure 3.4-4. Waveform discrimination technique.

3-53

## Omni-Directional Time Delay and Integration

A time delay and integration (TDI) filter determines the peak of the signal and successively adds these peaks for each detector crossing. In this manner all uncorrelated noise and clutter is averaged out, while the signal peaks, being highly correlated, add in phase. The result is an enhancement of the signal. The output may be used for both target detection and intensity measurement.

The purpose of the TDI filter is to provide SNR enhancement prior to detection for selected regions and velocities in pixel space. In the handover function from the earth staring sensor to the ATH sensor, approximate track information is available to reduce the computational complexity of the omni-directional TDI function.

## Spatial Detection Filter

Spatial filtering is used to discriminate between targets and clutter based on the relative physical size. The targets are defined to occur in less than two detector elements simultaneously while clutter will generally be present in many contiguous detectors. Since inadequate spatial correlation information (RM-19 data) is available, it is difficult to assess the performance of a spatial filter at this time. However some preliminary data are presented.

Two spatial filter techniques are introduced in this section. The first, local area pixel processing, is a relatively conventional approach toward clutter discrimination. The second, Walsh-Hadamard processing, is felt to be applicable to target enhancement. A system performance/cost tradeoff should be conducted to select the optimum spatial filtering technique for implementation.

## Spatial Filtering Via Local Area Pixel Processing

Local area pixel processing detects the presence of point targets by sliding a small two-dimensional window across the array of pixel amplitudes. The pixel(s) at the center of the window is tested for the presence of a possible target. The window is sequentially moved across the array such that each

pixel is tested. The test is based on the fact that a target will result in an amplitude peak with respect to the neighboring pixels.

The local area pixel processing algorithm for a 5 x 5 window is illustrated in Figure 3.4-5. Two coefficients are computed for a line intersecting the central candidate pixel, one representing target energy and the other representing clutter energy. These coefficients are subtracted to determine if a threshold has been exceeded. The process is repeated for lines intersecting the central pixel(s) in each of four directions: vertical, horizontal and the two diagonals. If the threshold is exceeded in all directions a hit is reported to the track processor.

The spatial filter algorithm is based on the computation:

$$P_{ij} = \sum_k P_k$$

where

$$P_1 = Ba_{13} + Aa_{23} - a_{33} + Aa_{43} + Ba_{53}$$

$$P_2 = Ba_{31} + Aa_{32} - a_{33} + Aa_{34} + Ba_{35}$$

| $a_{11}$ | $a_{12}$ | $a_{13}$ | $a_{14}$ | $a_{15}$ |
|---|---|---|---|---|
| $a_{21}$ | $a_{22}$ | $a_{23}$ | $a_{24}$ | $a_{25}$ |
| $a_{31}$ | $a_{32}$ | $a_{33}$ | $a_{34}$ | $a_{35}$ |
| $a_{41}$ | $a_{42}$ | $a_{43}$ | $a_{44}$ | $a_{45}$ |
| $a_{51}$ | $a_{52}$ | $a_{53}$ | $a_{54}$ | $a_{55}$ |

UNCLASSIFIED

Figure 3.4-5. The 5 x 5 pixel group for local area pixel processing.

$$P_3 = Ba_{11} + Aa_{22} - a_{33} + Aa_{44} + Ba_{55}$$

$$P_4 = Ba_{15} + Aa_{24} - a_{33} + Aa_{42} + Ba_{51}.$$

and where $A = 2^n$ and $B = 2^m$.

Figure 3.4-6 shows the response of the filter to a point target and to a line. In addition to $P_{i,j}$, a "data valid" signal has to be generated. This signal indicates that the pixel $(i,j)$ contains a potential target. This is determined by comparing the four $P_k$ with a threshold value. Only if all four $P_k$ exceed the threshold is the pixel considered to contain a potential target.

In order to carry out the computation of $P_{i,j}$ dynamically, it is necessary to store 4 lines of the image and the intensity of the 24 pixels
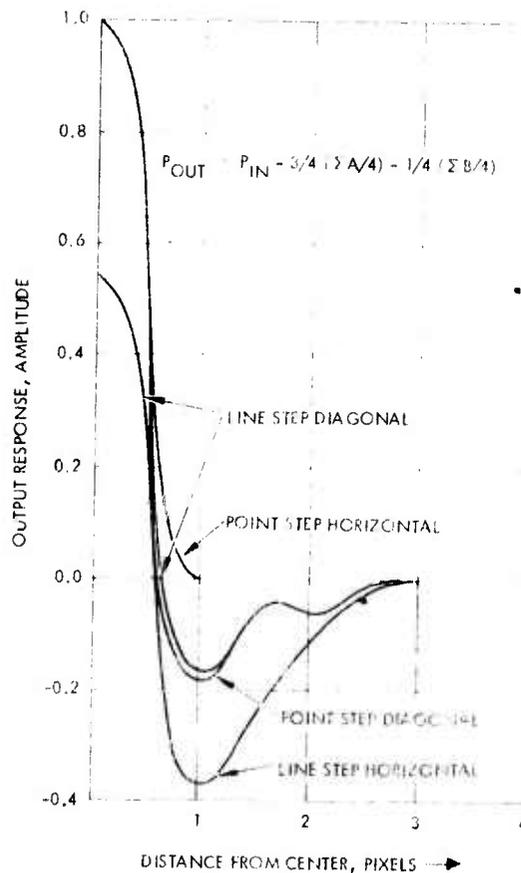


Figure 3.4-6. Local area pixel processing response.

around (i, j) must be in registers that are easily accessible to the arithmetic units. The value $a_{33}$ also has to be available in an analogous manner. Four adders, capable of adding 5 operands each, compute the values $P_k$. Multiplication by the weights A and B is done by binary shifting since A and B are restricted to be exact powers of 2. The spatial filter processor operates as a pipeline.

Knowing that 4 words have to be stored for each potential target (hit) the amount of time required to process the 16,384 pixels of one detector/mux array chip can be found to be;

$$t = 16,384 \frac{\text{pixels}}{\text{chip}} * 4 \frac{\text{words}}{\text{pixel}} * 50 \frac{\text{nsec}}{\text{word}} = 3.28 \frac{\text{milliseconds}}{\text{chip}}$$

Thus several detector/mux array chips may time share one spatial filter, i.e., $(0.1/3.28) * 10^3 = 30$ detector/mux array chips may be processed by one spatial filter processor.

## Digital Walsh-Hadamard Transform Spatial Filtering

The Walsh-Hadamard Transform (WHT) spatial filtering technique is based on transforming a block of pixel amplitudes into sequencies through use of a one-dimensional Walsh-Hadamard transform. Each sequency represents a weighted sum of the detector outputs over the transform block length. Since only the higher order sequencies contain target information, the lower-ordered sequencies can be discarded.

Figure 3.4-7 illustrates the operations required to generate the first eight sequencies of a length 16 Walsh-Hadamard transform by digital techniques. The calculations are similar for length 128. By changing the uppermost row of additional operators to subtractions the upper 8 sequencies can be generated.

A linear combination of sequency coefficients results in additional spatial filtering by eliminating the periodic nature of the Walsh-Hadamard transform. Figure 3.4-8 shows the operations which are required to enhance targets within two detector elements and to suppress clutter.
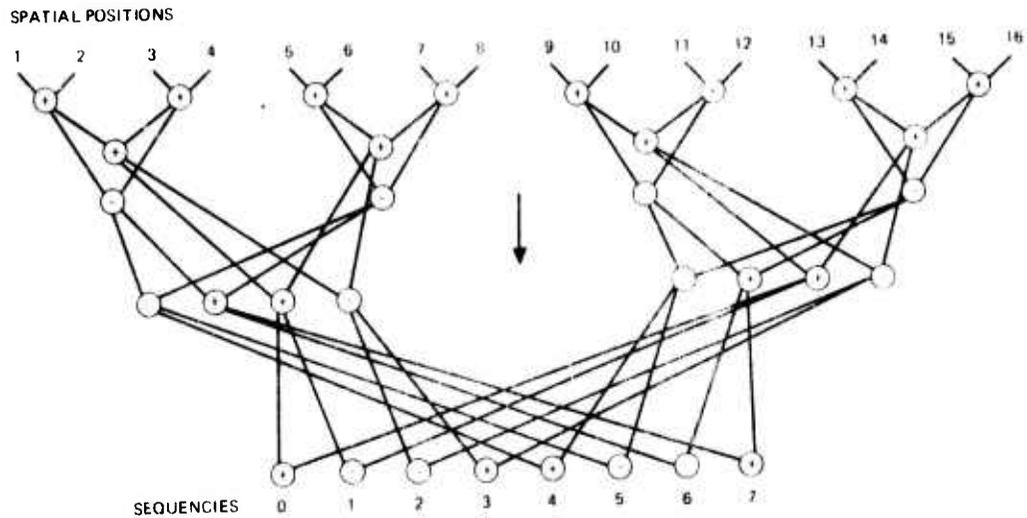
Figure 3.4-7. Forward Walsh-Hadamard transform operations required to map spatial positions into sequencies of block length N = 16.

INPUT POSITIONS

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1,2 | 4 | 4 | 4 | -4 | | | | | | | | | | | | |
| 2,3 | 4 | 4 | 4 | 4 | | | | | | | | | | | | |
| 3,4 | 4 | -4 | 4 | 4 | | | | | | | | | | | | |
| 4,5 | -2 | 2 | -2 | 6 | 6 | -2 | -2 | -2 | | | | | | | | |
| 5,6 | | | | | 4 | 4 | 4 | -4 | | | | | | | | |
| 6,7 | | | | | -4 | 4 | 4 | -4 | | | | | | | | |
| 7,8 | | | | | 4 | 4 | 4 | 4 | | | | | | | | |
| 8,9 | | | | | 2 | -2 | -2 | 6 | 6 | 2 | -2 | -2 | | | | |
| 9,10 | | | | | | | | | 4 | 4 | -4 | -4 | | | | |
| 10,11 | | | | | | | | | -4 | 4 | 4 | -4 | | | | |
| 11,12 | | | | | | | | | 4 | -4 | 4 | 4 | | | | |
| 12,13 | | | | | | | | | 2 | 2 | -2 | 6 | 6 | -2 | -2 | -2 |
| 13,14 | | | | | | | | | | | | | 4 | 4 | 4 | -4 |
| 14,15 | | | | | | | | | | | | | -4 | 4 | 4 | -4 |
| 15,16 | | | | | | | | | | | | | -4 | -4 | 4 | 4 |

POSITION FILTER OUTPUTS

NOTE EACH ROW IN THE ABOVE TABLE CONTAINS THE COEFFICIENTS OF THE INPUT POSITION AMPLITUDES WHICH MUST BE SUMMED TO GENERATE THE POSITION FILTER VALUES DIRECTLY WITHOUT PERFORMING A HADAMARD TRANSFORM

Figure 3.4-8. Walsh-Hadamard spatial filter operations.

Figure 3.4-9 presents some data from a simulation used to develop the concept.

The input data in this example was an FAA clutter map having an amplitude distribution as shown. A low contrast target was superimposed on the clutter. The lower figure shows the enhancement which is achieved when a length .! Walsh-Hadamard spatial filter is used. Note the ability to separate the target from the background by use of threshold detection after transforming.

In summary, a local area pixel processing and a digital WHT spatial filter can be implemented in software in the track processor. The computational loads are quite modest, corresponding to a throughout of 1 MIPS in the track processor.

The spatial filtering will enhance target detection. Further simulations based on real clutter data are required to determine the relative performance of both techniques.

Target Detection Logic

The purpose of the target detection threshold circuits is to generate hits for both positive and negative amplitude signals and to transmit these hits to the trackers. The trackers can handle up to about a 1 percent hit probability (targets and clutter).

Feedback is provided from the trackers to the threshold logic so that a fairly constant hit rate can be maintained even while the characteristics of clutter and sensor noise vary with time and over the field of view.

While adaption is a requirement, the ability to override that adaption is also a requirement. This requirement results from at least two system requirements:

1. The requirement to give priority in surveillance to known launch sites and test areas even in the face of heavy clutter in those areas.

2. The requirement to maximize data collection on established tracks. Thus, during periods when the target intensity dims (i.e., second stage through third stage and post boost thrusting) reducing thresholds in the area of the target track is required.
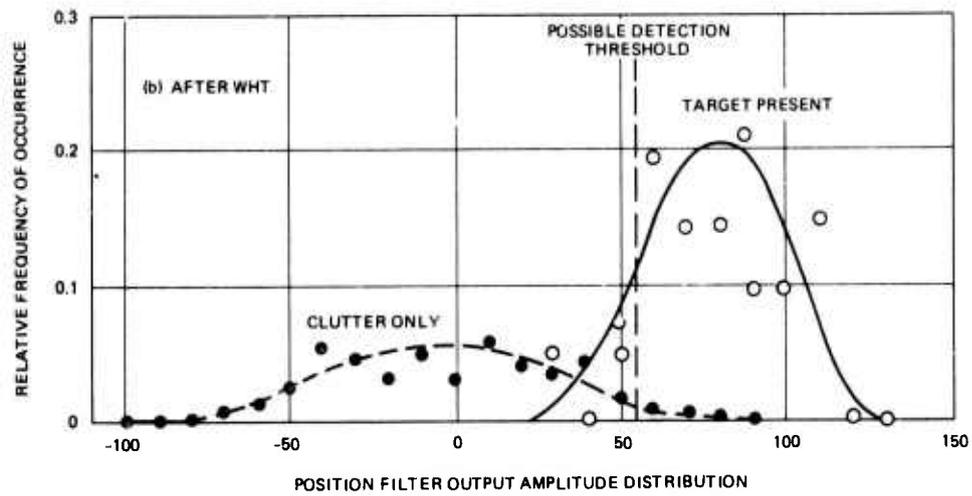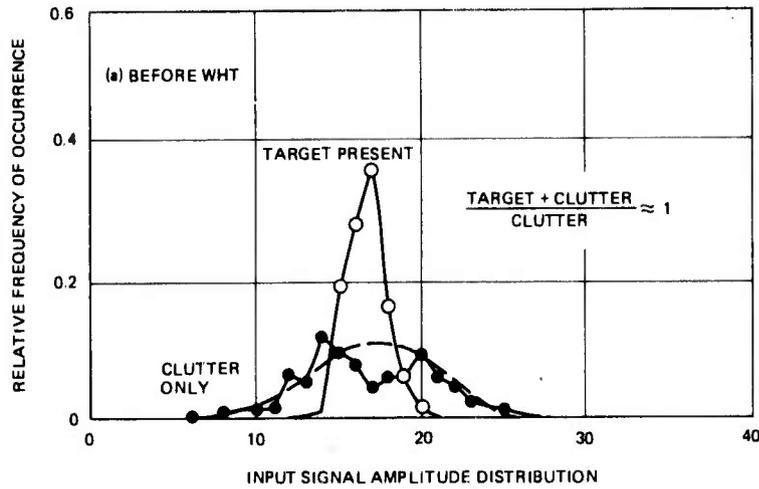
Figure 3.4-9. Radar clutter map histograms.

The above requirements may lead to the following implementation of a three threshold system:

The lowest threshold is called the data rate threshold and is merely used to obtain a count on the number of data points which exceed this threshold each frame.

The intermediate threshold is called the track data threshold and all data points from the filter processing unit which exceed this threshold are available to support the existing tracks in the tracker unit. Those points which are not associated with existing tracks may be discarded if only forward time track formation is implemented.

The highest threshold is called the track initiator threshold and all data points which exceed this threshold and which have not been associated with an existing track are utilized to start new track files if track slots are available.

Figure 3.4-10 is a block diagram of one possible detection threshold multi-level logic implementation.

## Track Processor

Figure 3.4-11 shows a block diagram of the track processor. It can be implemented on five LSI chips using 1985 projected technologies. The track processor is a computer which features a throughput of 8 MIPS, 16 bit word length, specialized I/O ports and priority interrupt structure.
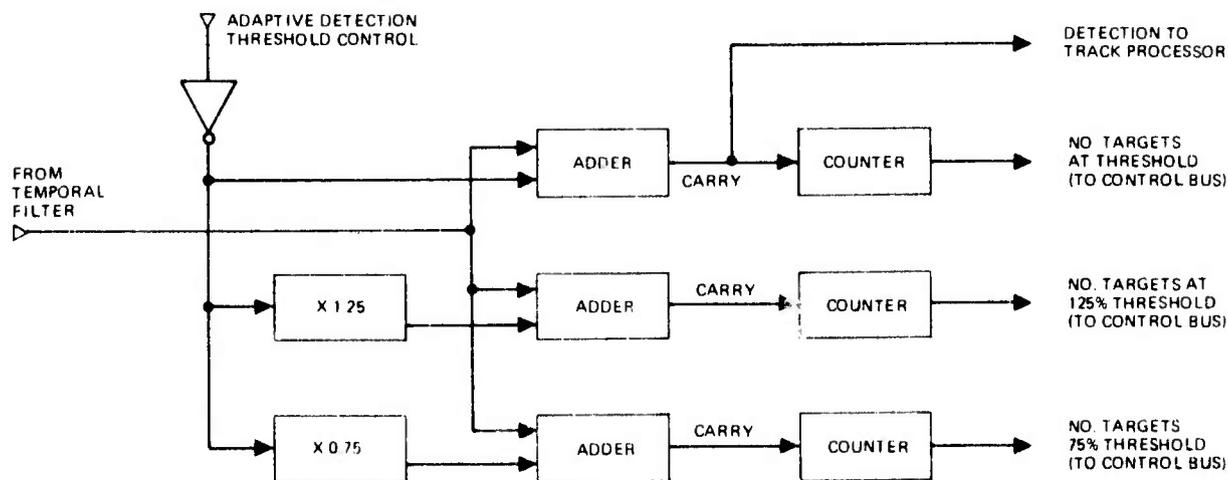


Figure 3.4-10. Target detection logic.
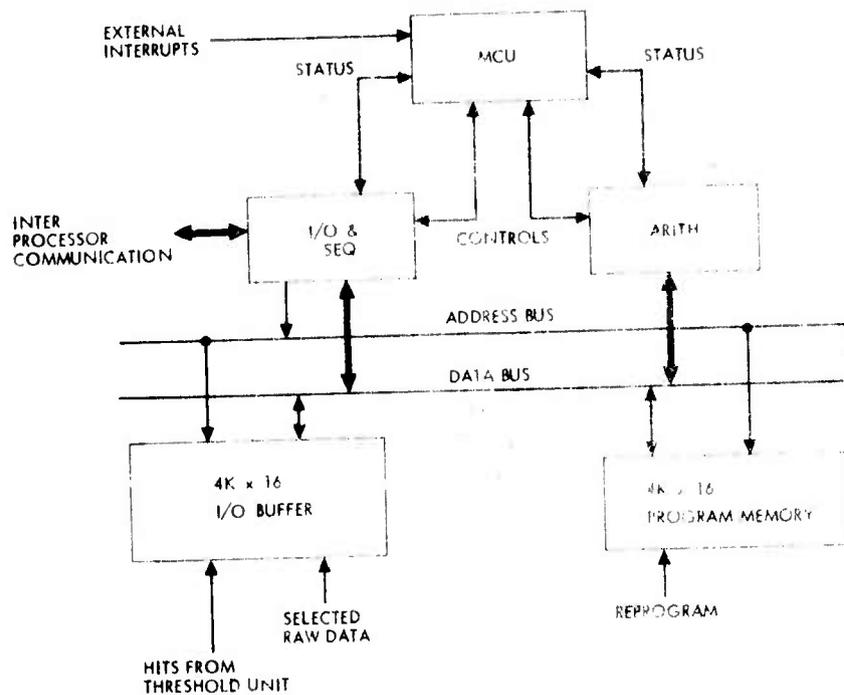
Figure 3.4-11.   Track processor block diagram.

This computer has been specifically designed to implement the tracking function required of APSP.  The instruction set which is also tailored to the tracking function, is shown in Table 3.4-1.  The major functions on the five LSI chips are:  1) Micro-programmed control unit (MCU), 2) arithmetic unit (ARITH), 3) input/output and sequencing (I/O, SEQ) and 4) two random-access memories (RAM) – one memory accepts hit data from the detection logic; the other memory, containing the stored program, could be a PROM.

The arithmetic portion contains 128 general registers, an arithmetic logic unit, a multiply network, and related functional units.  A block diagram of this unit is shown in Figure 3.4-12.  The multiply network allows the parallel multiplication of two 16-bit operands during two machine cycles.

The sequencing and I/O chip, shown in Figure 3.4-13, issues addresses to memory for the purpose of fetching instructions and operands. This unit contains the arithmetic capability to perform address calculations. It also contains the interrupt structure with provisions for seven levels of priority interrupts.  An interrupt stack is provided that allows seven

3-62

## TABLE 3.4-1. THE INSTRUCTION SET OF THE TRACK PROCESSOR

### DATA MOVING

Load
Store
Exchange
Block Load
Block Store
Reg. to Acc.
Acc. to Reg.
Load Immediate

### LOGICAL

And
Or
XOR
1's Complement

### ARITHMETIC

Add
Subtract
Multiply
Divide
Increment Reg.
Decrement Reg.

### SHIFT

Arithmetic
Arithmetic Double
Rotate
Logical O-Ext.
Logical 1-Ext.

### INTERRUPT HANDLING

Set Trap Address
Set Mask Register
Resume

### MISCELLANEOUS

Halt
No-Op
Reset and Start

### BRANCHES

Unconditional
If Reg = 0
If Reg = ACC
If Reg >ACC
If Reg <ACC
Incr Acc. and Branch If ≤ Reg
If OFL on Previous Instr.

### I/O

Output to Vector Buffer
Output to Signal Processor
Output to Neighbor
Input From Neighbor
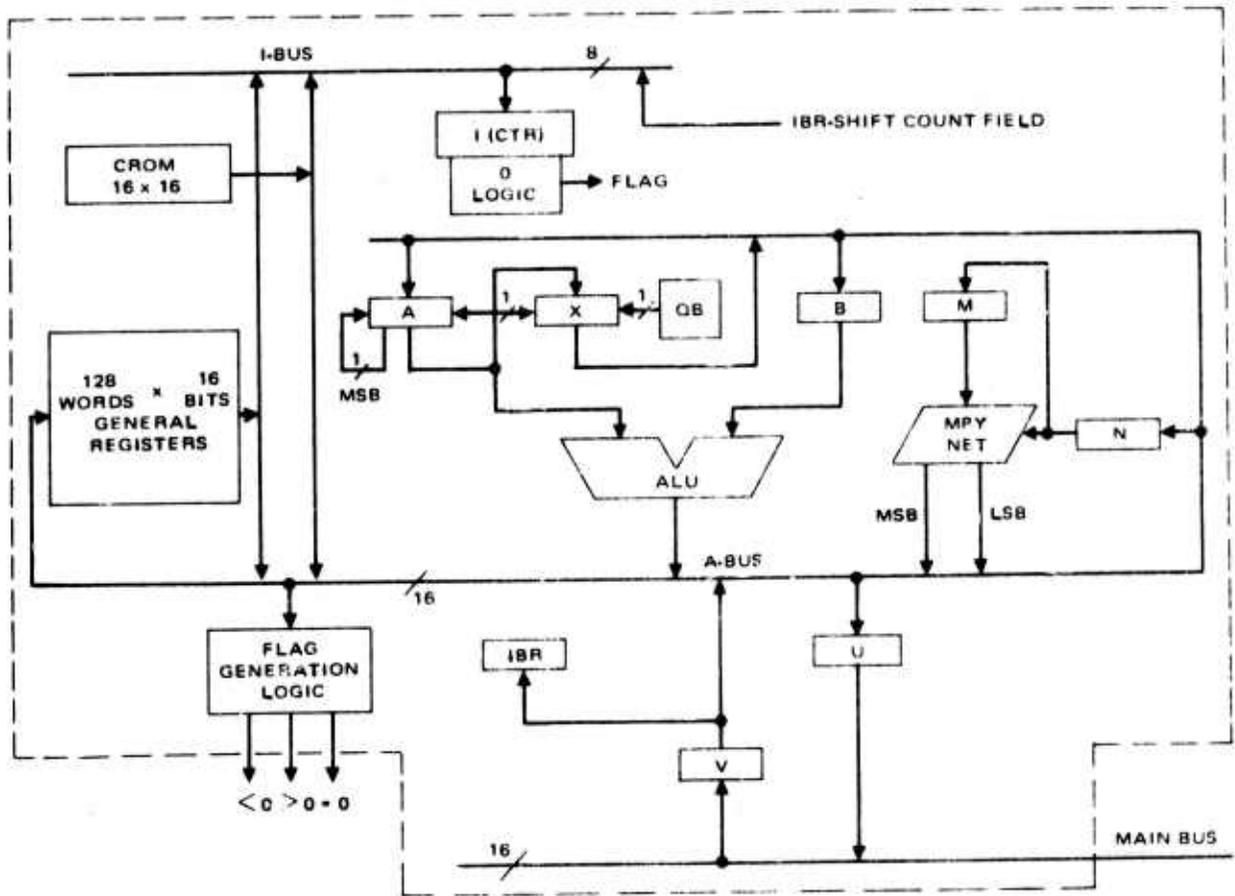
Figure 3.4-12. Arithmetic chip.

levels of nested interrupts. The sequencing and I/O chip also interfaces with the Track-Bus and the filter and detection processor for the purpose of issuing track file data and adaptive thresholds, respectively. The microprogram control chip, shown in Figure 3.4-14, supplies the detailed control signals required to operate the track processor. It consists of a 512 x 32-bit ROM. During each machine cycle one word is fetched from the ROM and placed in the Command Register. Subfields within this register are used directly or decoded to provide the necessary control signals. Logic is also provided to generate the address of the next control word to be fetched from the ROM. This consists primarily of an address multiplexer and flag select logic.
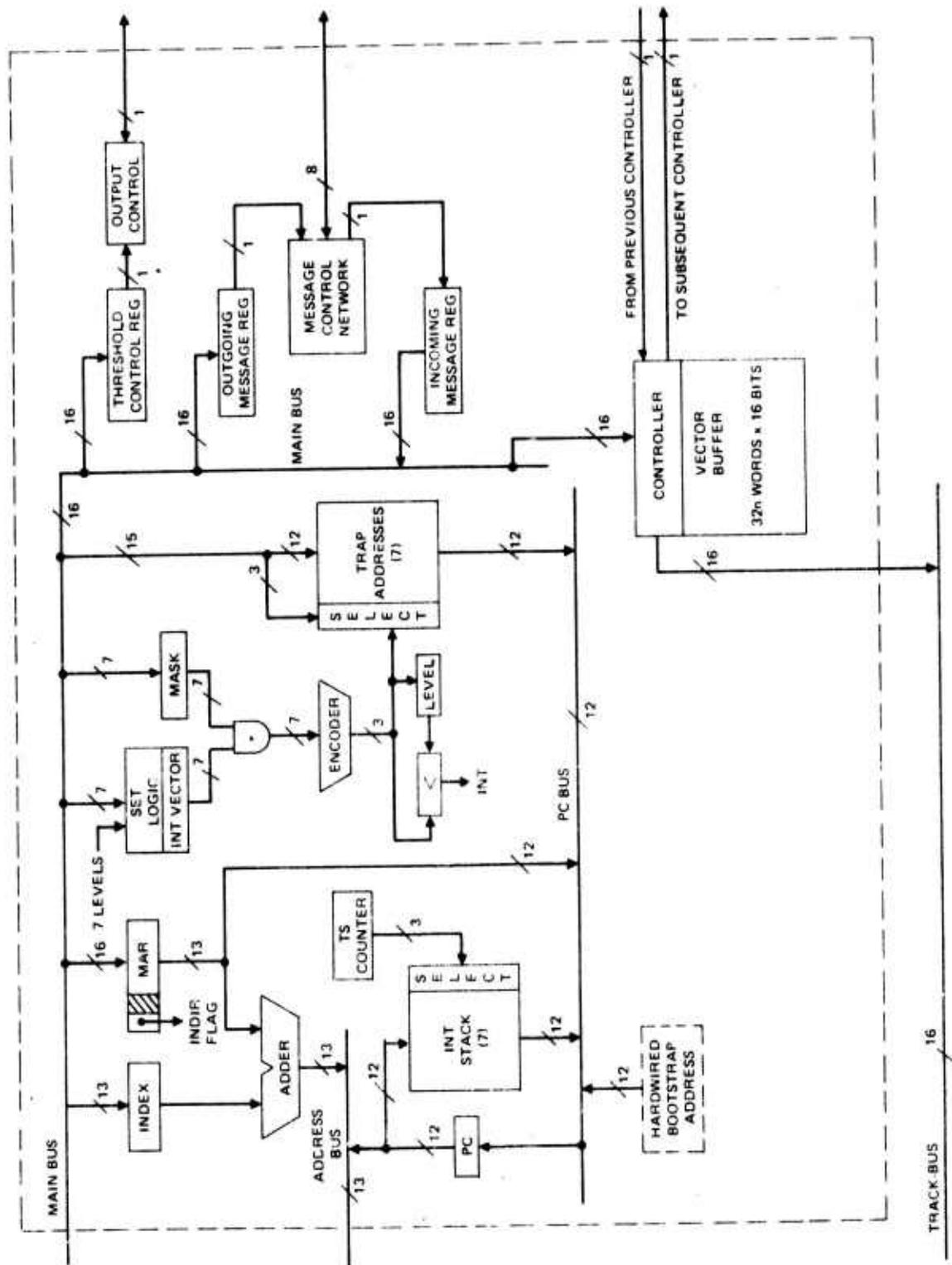
Figure 3.4-13. Sequencing and I/O chip.
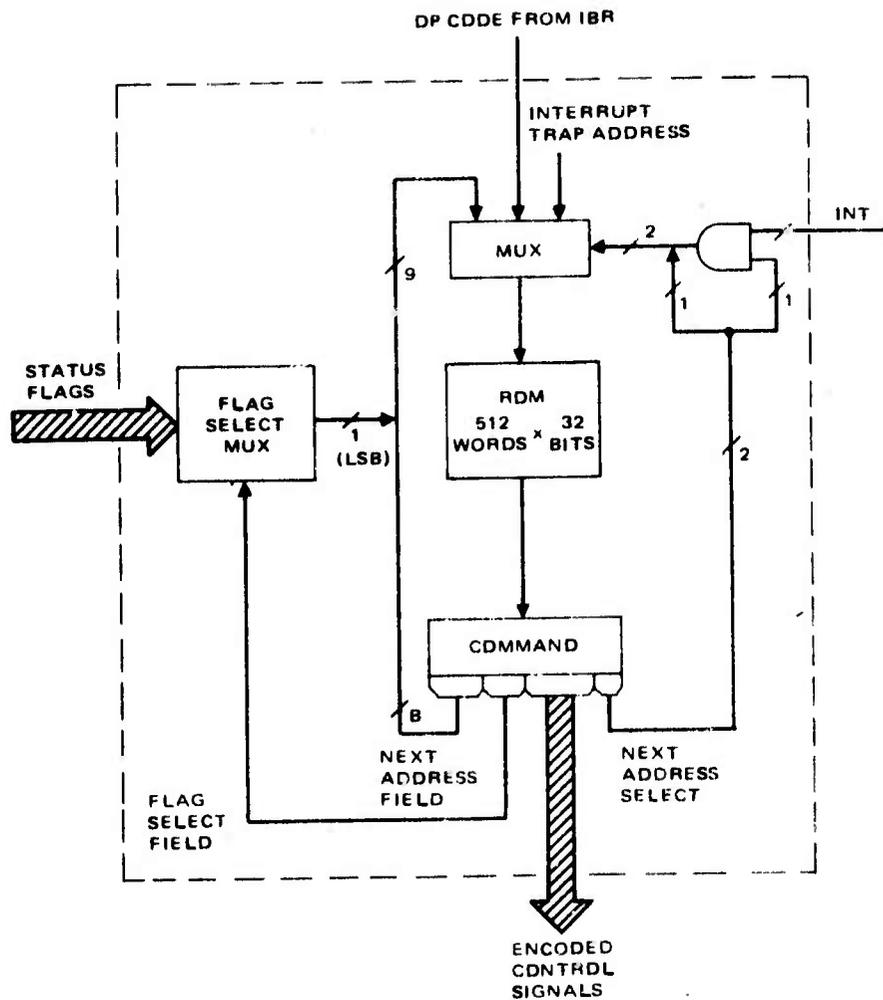
Figure 3.4-14.  Microprogram control unit.

A typical memory is illustrated in Figure 3.4-15.  Three regions of
memory are required for the processor.  First, a region containing 5K
locations is used to store programs, constants, and variables.  Second,
2K locations are used as a double buffer to store target HIT reports.  Third,
1K locations are used to store selected raw data from the filter processor.

The track processor is programmed to perform the following tasks:

- Target tracking
- Clutter tracking and deletion
- Adaptive temporal filtering
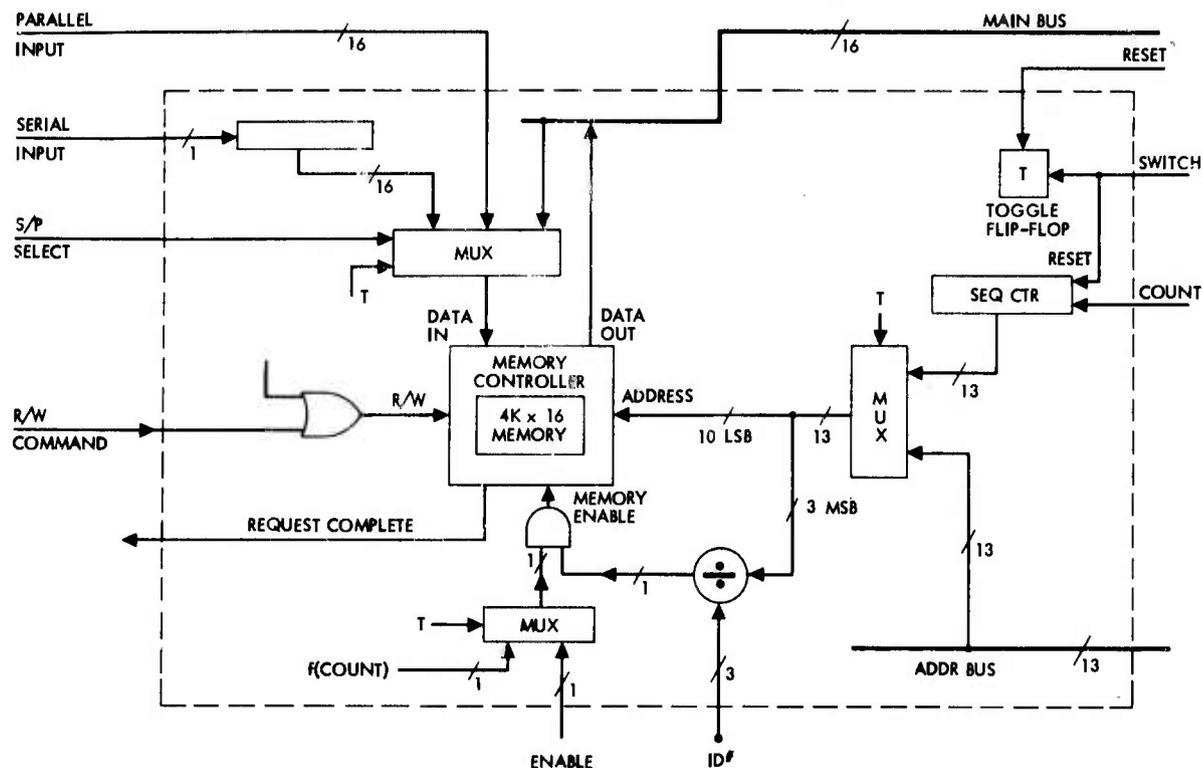
Figure 3.4-15. Memory chip.

- Spatial filtering
- Position space reconstruction from the analog WHT pre-processor
- Control of detection threshold
- Control of AVE dynamic range modes

The use of a dedicated track processor (one per detector/mux array) is the only available technique for most of the adaptive functions in the APSP since all classical filtering algorithms lack the required required flexibility.

## Submodule Processor Chip Partitioning

Based on the projected power and densities of future LSI implementations, the system submodule consists of 12 custom LSI chips interconnected with a hybrid package. The organization of the chips is illustrated in Figure 3.4-16.

3-67
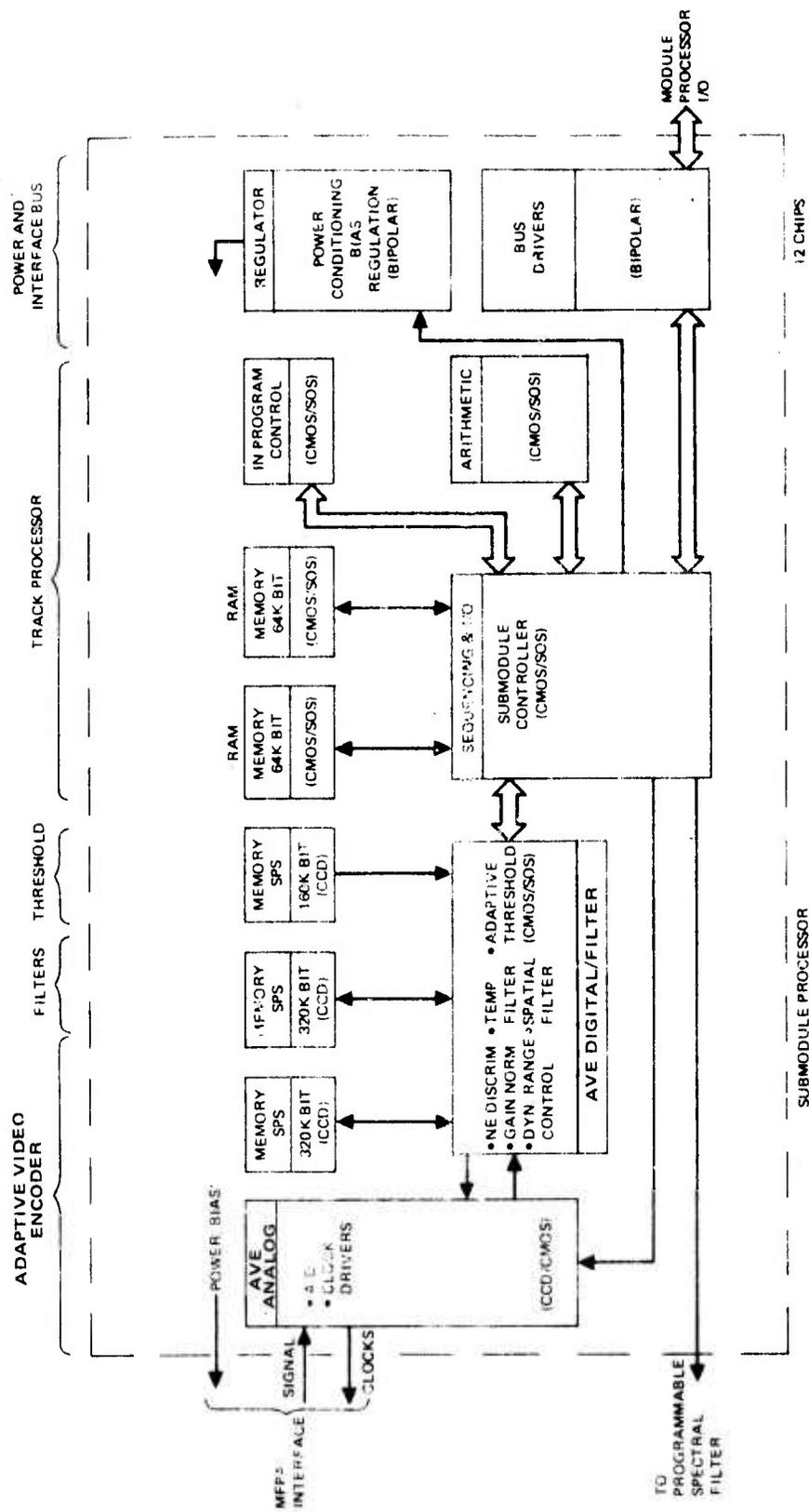
Figure 3.4-16. Submodule chip configuration (advanced technology).

Signals from the detector/mux array are encoded as ten-bit words on a CCD/CMOS chip which interfaces with the digital AVE electronics. Commands are created in the AVE digital chip and are transmitted to the control processor chip which contains the dynamic range algorithms (and over-rides) which optimize the S/N ratio for the various missions.

The temporal and spatial detection filters and adaptive detection threshold logic are combined on one chip with the AVE digital logic. Associated with the large random logic chips are serial (block organized) memories which use serial-parallel-serial-parallel-serial CCD devices for nuclear event suppression and temporal filtering.

The track processor in the APSP is composed of five advanced technology LSI chips with random logic densities approaching 30,000 gates per chip (200 x 200 mil). This density is made possible through high resolution projection photolithography and electron beam microfabrication techniques while power dissipation is held well within thermal power density limits by the use of low power devices.

Table 3.4-2 is a tabulation of the nine special chips that make up the twelve chip submodule processor in a hybrid package. Estimated chip power consumption and number of Input/Output chip pads are shown.

TABLE 3.4-2.  SUMMARY OF THE APSP HARDWARE
CHARACTERISTICS

| Chip Type (Number Required) | Number of Equivalent Gates | Power Consumption Per Chip | Number of I/O Pads |
|---|---|---|---|
| Analog AVE (1) | — | 81 mW | 24 |
| Digital AVE (1) | 9,400 | 5 mW | 28 |
| Serial Memory (3) | 320,000 | 14 mW | 18 |
| Arithmetic (1) | 25,000 | 42 mW | 38 |
| Sequencing and I/O (1) | 16,000 | 53 mW | 79 |
| Microprogram Control (1) | 8,400 | 21 mW | 44 |
| RAM Memory (2) | 82,000 | 15 mW | 59 |
| Bus Driver (1) | 5,000 | 100 mW | 30 |
| Voltage/Bias Regulator (1) | — | 50 mW | 36 |

# 4.0  DESCRIPTIONS OF FUNCTIONAL UNITS

This section describes in detail the functional units of the proposed APSP.  These units are the 1) adaptive video encoder with dynamic range control, 2) temporal detection filter, 3) spatial detection filter using both local area pixel processing and Walsh Hadamard transform processing, 4) target detection logic and 5) the track processor.  A number of symbols are used frequently in the discussions which follow.  For convenience, those symbols are defined below.

## DEFINITION OF SYMBOLS

| | |
|---|---|
| $Q$ | Subscript, represents digital value |
| $P_Q$ | Predicted digital word |
| $P$ | Analog predicted word |
| $S$ | Analog input from detector/mux array |
| $E$ | Analog difference signal |
| $E_Q$ | Digital difference signal |
| $Q_o$ | Quantization level |
| $PC_a$ | Digital predicted corrected value (Output from encoder) |
| $k$ | Discrete time index |
| $n$ | Number of taps |
| $\tau_D$ | Dwell time on detector |
| $T$ | Sample period |
| $m$ | mth derivative |

## 4.1 The Adaptive Video Encoder (AVE)

As shown in Figure 4.1-1, the AVE functions as the APSP interface with the detectors of the Monolithic Focal Plane Array (MFPA). The AVE receives data from the MFPA, encodes the data in digital form, and passes it to the LAP for further processing. The AVE also receives control commands and digital feedback from the LAP, senses saturation in the system, and provides adjustment control and mode change operations.

The AVE encodes the MUXed analog signal from the MFPA chip. The resulting word is a 10-bit representation of the signal. The encoder utilizes a prediction of the next signal value based on previous values (rather than on the statistical properties of the signal) to generate a difference signal which is then encoded and added to the prediction to give the encoded signal. The dynamic range of the encoder is 1023:1 (60 dB).

Figure 4.1-2 is a block diagram of the prediction feedback encoder. The 10-bit predicted word $P_Q$ is converted to its analog counterpart $P$ and subtracted from the MFPA signal $S$ via control of the fat zero level in the CCD channel. A difference signal $E$ is thus generated which is encoded as $E_Q$. This takes place in either channel 'a' or channel 'b'. Assuming MFPA signals between 0 and 1, the $2^5-1$ quantization levels in both A/D converters are determined as follows:

The 10 bit D/A converter generates analog signals between 0 and 1; thus the LSB in the 10-bit word applied to this D/A must represent a value of $1/(2^{10}-1)$. The LSB in the output word formed by channel 'b' of the 2-channel ADC must also represent this value. Since this is a 5-bit A/D converter, the largest value it can convert without exceeding saturation is $(2^5-1)/(2^{10}-1)$. Also since for channel 'b', $E$ is amplified by $2^5$, the saturating value needs to be scaled by the same amount. Thus the channel 'b' A/D can represent a full scale amplitude of $2^5(2^5-1)/(2^{10}-1)$ and because it is a 5-bit A/D its quantization level will be this full scale voltage divided by $(2^5-1)$ or

$$Q_o = \frac{2^5}{2^{10}-1}. \tag{1}$$

4-2
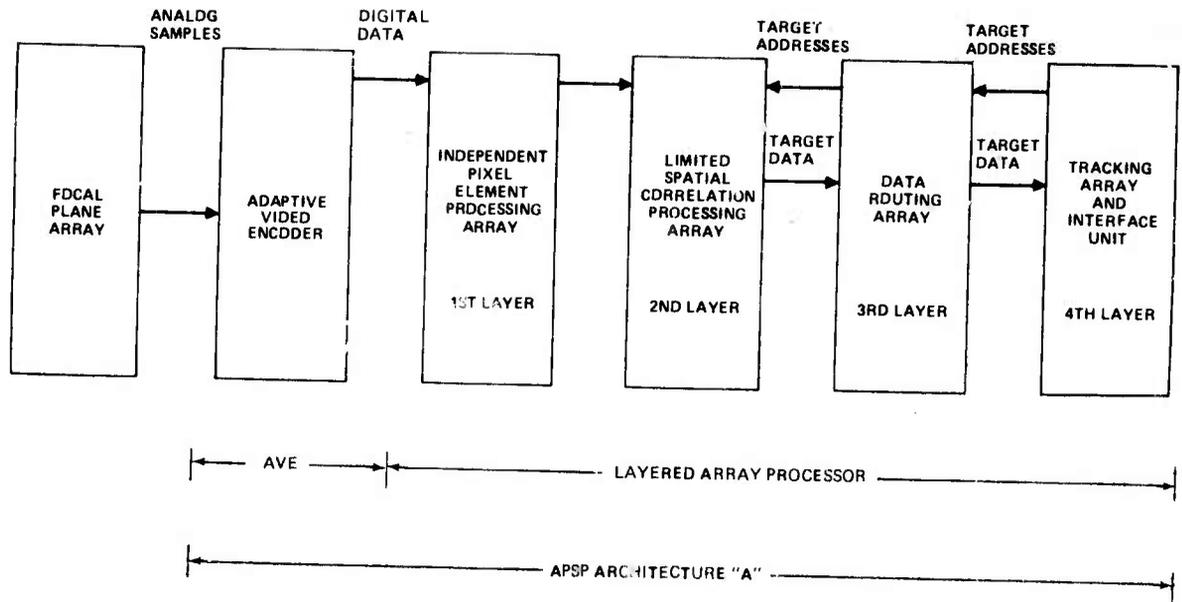
Figure 4.1-1.   APSP
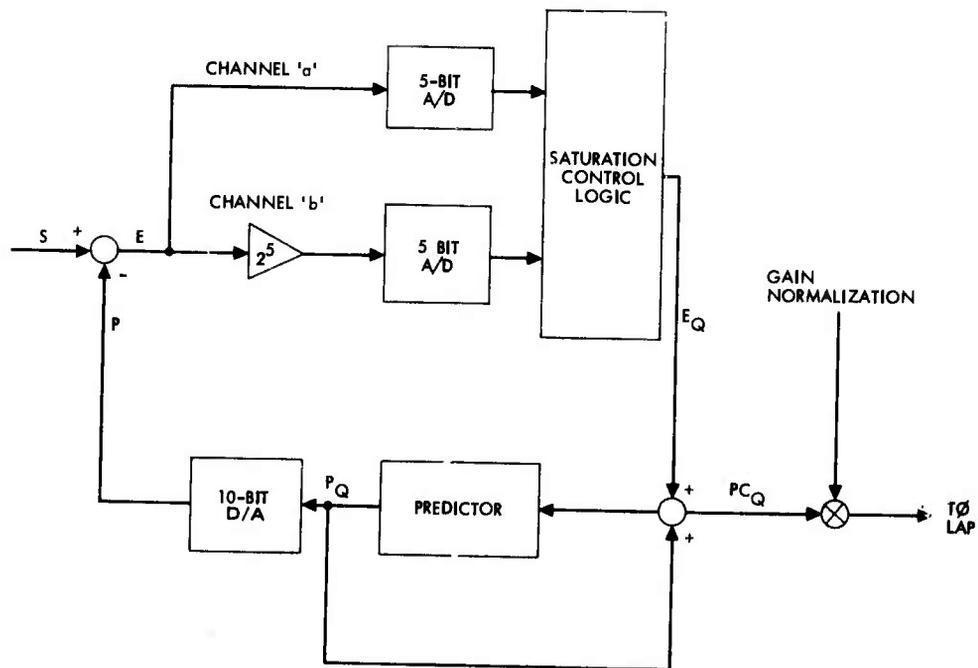


Figure 4.1-2.   Prediction feedback encoder.

4-3

The channel 'a' A/D saturation value is

$$|E|_{SAT} = (2^5 - 1)Q_o.$$

Only the magnitude of E will be applied to the A/D converters so that $E_Q$ must have a sign bit. Now, whenever

$$0 \le |E| \le |E|_{SAT}/2^5$$

the 5-bit word from channel 'b' will be placed in bits 1-5 (LSB) of $E_Q$ (Bits 6-10 will be zero). This is just a normal 5-bit A/D conversion of E. However, when

$$|E|_{SAT}/2^5 < |E| \le |E|_{SAT}$$

the 5-bit result from channel 'a' will be placed in bits 6-10 (MSB) of $E_Q$ (bits 1-5 will be zero). The result will be an approximate representation of E with an error of magnitude no greater than $E_{SAT}/2^5$. This two channel scheme is necessary to allow the encoder to recover from large errors in predicting the signal. $E_Q$ is now added to the 10-bit predicted word $P_Q$ forming the predicted-corrected value $PC_Q$ which is a 10-bit representation of the MFPA signal S and which is then used in predicting the next value of S and as output to the LAP.

The accuracy of the output word $PC_Q$ depends on the accuracy in the A/D conversion of E; the D/A conversion of $P_Q$; and the analog summation S-P. The quantization noise introduced by the conversion of E will dominate if it is assumed for purposes of analysis that the analog summation is ideal.

The optimum predictor will produce difference signals that are no greater than 5-bits such that when $E_Q$ and $P_Q$ are added, the resultant $PC_Q$ will represent the MFPA signal S within the accuracy of the A/D. The predictor must also have few delays since a memory will be required for each pixel element for each delay.

### The Predictor

A number of predictors including geometric feedback, AD PCM (adaptive differential pulse code modulation), and least mean square cubic types were considered during the study and the polynomial fit predictor was found to be optimum among those considered.

From Newton's backward difference formula, for an n-point prediction

$$P_k = \sum_{i=0}^{n-1} \nabla^i S_{k-1-i}$$

where

$$\nabla^0 S_k = S_k$$

$$\nabla^i S_k = \nabla^{i-1} S_k - \nabla^{i-1} S_{k-1}$$

This may be rewritten

$$P_k = \sum_{i-1}^{n} a_i S_{k-i} \tag{2}$$

where

$$a_i = \binom{n}{i}(-1)^{i-1}.$$

The bound on the error magnitude $\left| S_k - P_k \right|$ will be proportional to MAX $\left| S^{(n+1)}(\theta) \right|$, $(k-n)T < \theta < kT$. Thus for rapidly changing signals, the

error can become large. For example, a unit step function can be considered
a worst case situation: for such a signal the predicted values are

$$P_k = \begin{cases} 0 & k \leq 0 \\ \sum_{i=1}^{k} a_i & k = 1, 2, \ldots, n-1 \\ 1 & k \geq n \end{cases}$$

and the resulting difference signal is

$$E_k = S_k - P_k$$

$$= \begin{cases} 1 & k = 0 \\ 1 - \sum_{i=1}^{k} a_i & k = 1, 2, \ldots, n - 1 \\ 0 & \text{elsewhere} \end{cases} \tag{3}$$

Thus the prediction is very far from the signal value beginning at the dis-
continuity takes n samples to recover - a ringing effect essentially.
Table 4.1-1 shows the results of Equation (3) for $1 \leq n \leq 5$.

Figure 4.1-3 shows the implementation of Equation (2). A total of
n-delays, n-multipliers, and an n-input adder are required.

Referring to Figures 4.1-2 and 4.1-3, the z-domain equations can
be written to determine the system frequency characteristics (neglecting
quantization). These are

$$E(Z) = S(Z) - P(Z)$$

$$P(Z) = \sum_{i=1}^{n} a_i PC(Z) Z^{-i} \triangleq H(Z) PC(Z)$$

$$PC(Z) = E(Z) + P(Z) = S(Z).$$

## TABLE 4.1-1. DIFFERENCE SIGNAL RESPONSE TO UNIT STEP INPUT FOR N-POINT PREDICTORS

| k | $E_k$ | | | | |
|---|---|---|---|---|---|
| | n=1 | n=2 | n=3 | n=4 | n=5 |
| 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | -1 | 2 | -3 | -4 |
| 2 | 0 | 0 | 1 | 3 | 6 |
| 3 | 0 | 0 | 0 | -1 | -4 |
| 4 | 0 | 0 | 0 | 0 | 1 |
| 5 | 0 | 0 | 0 | 0 | 0 |
| $\sum_k \lvert E_k \rvert^2$ | 1 | 2 | 6 | 20 | 70 |
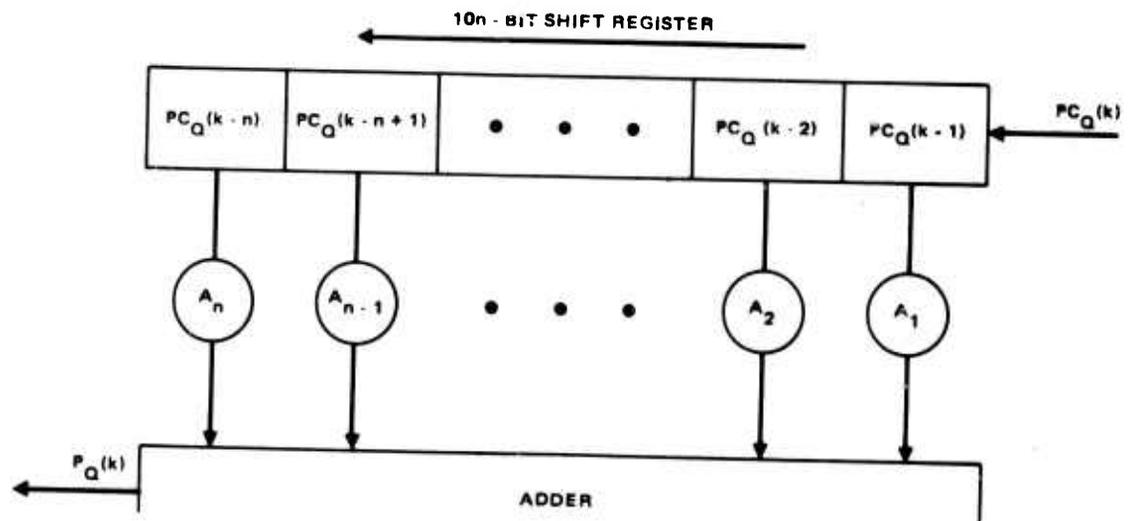


Figure 4.1-3.   n-Point polynomial predictor.

Thus

$$H_D(Z) \triangleq E(Z)/S(Z) = 1 - H(Z)$$

where the predictor transfer function is given by

$$H(Z) = \sum_{i=1}^{n} a_i Z^{-i}.$$

The magnitude characteristic $\left| H(e^{i\omega T}) \right|$ is plotted in Figure 4.1-4.

Now from the binomial theorem and Equation (2), the encoder transfer function is

$$H_D(Z) = \left( 1 - Z^{-1} \right)^n.$$

The magnitude characteristic is

$$\left| H_D(e^{i\omega T}) \right| = 2^n \left| SIN \frac{\omega T}{2} \right|^n$$

and is plotted in Figure 4.1-5. A maximum value of $2^n$ occurs at $\omega T = \pi$ (or $f = 1/2T$). Because of the optical system characteristics, the signal will be approximately band limited to $\pm 1/\tau_D$ where $\tau_D$ is the dwell time (time for a point to move across a detector cell). If $\tau_D > 2T$ to prevent aliasing, then the predictor will not become unstable at large values of n. The tradeoff is between a large n for predictor accuracy (recall that the error is proportional to the (m+1)st derivative of the signal) and a small n to minimize the error for signals with significant energy in the region around $1/\tau_D$. In favor of a smaller n is the better impulse response characteristic (Table 4.1-1) and lower complexity.
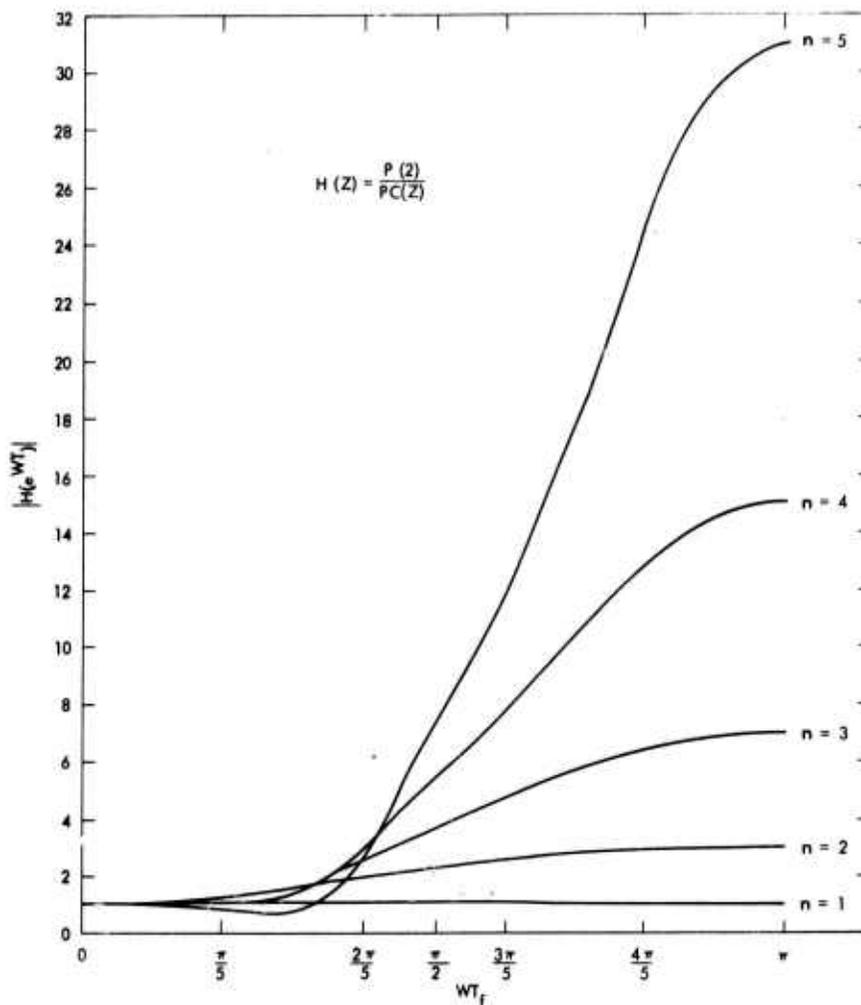
Figure 4.1-4. Frequency characteristic of n-point predictor.

The sum of the squared error is given by

$$\sum_k \left|E_k\right|^2 \triangleq \Delta = \frac{1}{2\pi i} \oint E(Z)E(Z^{-1})Z^{-1}dZ \,.$$

If the worst case signal is assumed to be the unit step then

$$S(Z) = (1 - Z^{-1})^{-1}$$
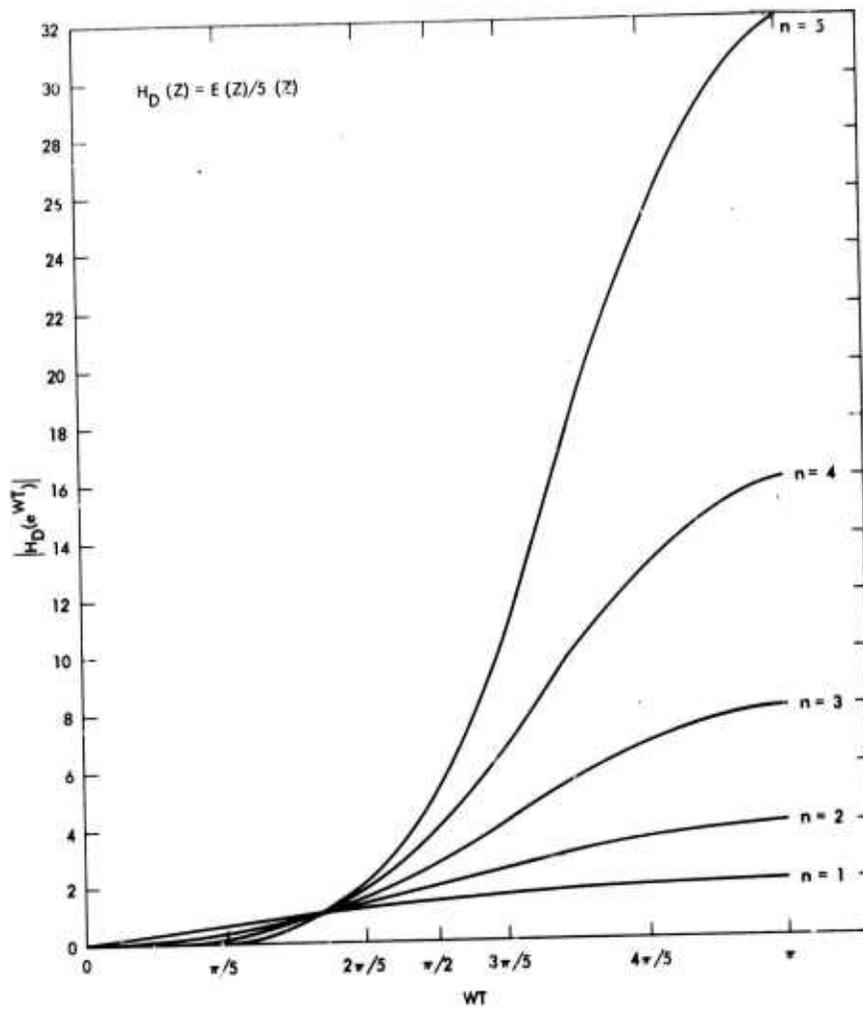
Figure 4.1-5. Encoder frequency characteristic
(n-point predictor).

and

$$\Delta = \frac{(2n - 2)!}{\left[(n - 1)!\right]^2} \quad .$$

This result agrees with the values shown in Table 4.1-1.

## Encoder Definition

The encoder will use the 1 or 2 point polynomial fit predictor. Previous values of the encoded signal $PC_Q$ will be stored in a $10n(n = 1, 2)$ shift register — one for each pixel element on the MFPA. This n-frame memory may be accessed for later processing and for MFPA dynamic range control scaling. By doubling the memory size, two-color input schemes can be readily accommodated.

The predictor processor will also take advantage of a priori knowledge of the MFPA signal dynamic range (0-1) in that all predictions falling outside this range will be clamped at the appropriate boundary. In addition to improving predictor accuracy this insures that the dynamic range of the A/D converter is not exceeded and hence greatly enhances the unit step response of the encoder. Since some predictor weighting coefficients are larger than unity, intermediate weighted sums may exceed 10-bits for maximum signal inputs; thus the predictor adder and result register will require 11 bits. The extra bit will be used to determine any overflow and in that event will set the 10 LSB to all ones and apply the 10-bit result to the D/A and to the $E_Q + P_Q$ adder. At the lower end of the dynamic range, negative predictor values will be set to zero before being applied to the D/A and to the adder.

For a maximum frame rate of 10 frames/second and $2^{14}$ pixels/MFPA, the conversion time of the encoder must be no greater than 6.1 $\mu$sec. The cost of the encoder can be determined from a list of its component parts; for each MFPA these are:

1   2-channel A/D converter ($E$ to $E_Q$)

1   10-bit D/A converter ($P_Q$ to $P$)

1   Analog summer ($S-P$)

1   X32 amplifier

1   10-bit plus sign bit adder ($E_Q + P_Q$)

1   $10n\ (2^{14})$-bit memory/color

1   n-tap digital transversal filter (11 bits) with underflow and overflow logic.

## 4.2 Temporal Filter

The temporal filter (TF) will be realized by a third difference digital filter. This design is based on performance requirements in Section 6.2. The relation between input and output is given by

$$f_{out}(n) = \hat{f}_{in}(n) - 3f_{in}(n-1) + 3f_{in}(n-2) - f_{in}(n-3).$$

This filter requires 3 frames of memory.

For a single velocity range, one implementation of a difference equation is sufficient. For each added velocity range, or integration period, one additional realization of the filter equation must be implemented.

The implementation poses a significant problem: serial versus parallel processing. Parallel processing presents a pin limitation — versus — density dilemma. For 4 inputs and one output of 16 bits each, a minimum of 80 pins is required. However, the logic required is far below 500 gates. This would create a tremendous waste of volume in the Signal Processor. However, handling the data in a serial manner will not only reduce the number of pins per package, but allow many TF's to be fabricated on a single chip. This is also compatible with the serial implementation of the AVE.

The implementation of the TF is as follows and depicted in Figure 4.2.1.

For 3rd difference: 2 Adders $\cong$ 75 gates each, difference $\cong$ 75 gates = 225 gates/3rd difference filter.

For each additional velocity bin, two memory chips plus one section of a TF chip are required.

## 4.3 Spatial Filter

Spatial filtering refers to processing which is done on a single frame of MFPA data for the purpose of locating pixels illuminated by targets. Two classes of spatial filtering algorithms have been proposed: local area pixel processing and Hadamard spatial filtering. Local area pixel processing is the name given to the class of algorithms in which the output corresponding

a) 2ND DIFFERENCE                                   b) MATCHED
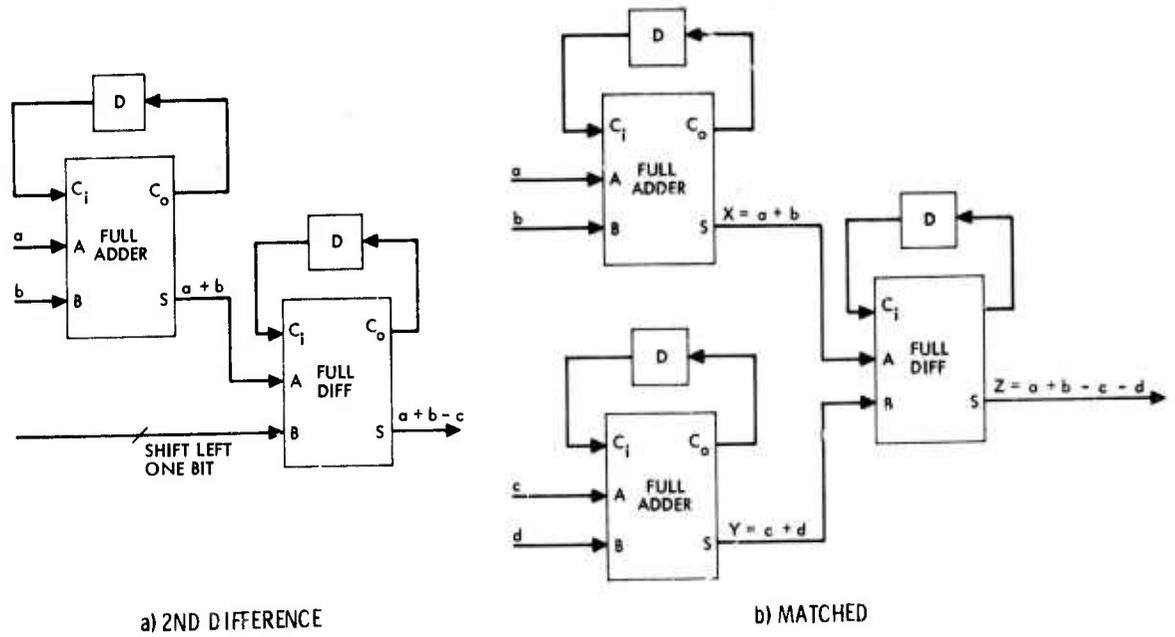
Figure 4.2-1.  Temporal filter (serial).

to a particular pixel is a function of its input amplitude and that of a small
number of nearest neighbors.  This method is discussed first.  Hadamard
spatial filtering is a technique whereby a block of pixel amplitudes are
transformed into the sequency domain, low-order sequences are discarded,
then a type of inverse operation is performed on the retained sequencies to
obtain the output.  The Hadamard algorithm is described and a computation-
ally equivalent algorithm is presented which falls into the category of local
area pixel processing as defined above.

### Local Area Pixel Processing

Local area pixel processing algorithms use a small window which
is moved across the array of MFPA data.  At a given time the small aggre-
gate of pixels contained within the window are used to calculate the output
corresponding to the central pixel.  This concept is implemented by shifting
pixel amplitude data through the spatial filter such that each pixel is pro-
cessed as a central point.  Specific implementations of this type of spatial
filter are described here.

4-13

Method 1 is illustrated in Figure 4.3-1. The four nearest neighbors on the diagonals are summed and multiplied by a weighting factor $C_A$. The four next nearest neighbors on the diagonals are also summed and multiplied by a weighting factor $C_B$. These terms are then subtracted from the central pixel to form the output.

Method 2, shown in Figure 4.3-2, is similar to Method 1. Weighted sums of the nearest neighbors on the edges and on the diagonals are subtracted from the central pixel to form the output.
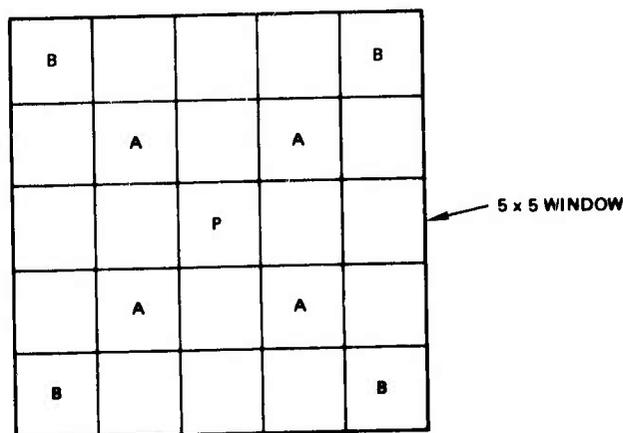


$$P_{OUT} = P_{IN} \cdot C_A \Sigma A \cdot C_B \Sigma B$$

Figure 4.3-1. Spatial filter method 1.
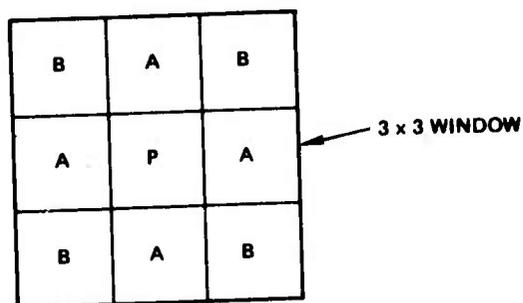


$$P_{OUT} = P_{IN} \cdot C_A \Sigma A \cdot C_B \Sigma B$$

Figure 4.3-2. Spatial filter method 2.

4-14

The algorithm shown in Figure 4.3-3 groups neighboring pixels according to their proximity to the center. Weighted sums of each group are subtracted from the central pixel to form the output.

To obtain a rough idea of the response characteristics of these filtering algorithms, some calculations were performed which simulate their response to simple spatial features. Both a point step and a line step were passed through each filter along the diagonal and along the horizontal as shown in Figure 4.3-4. The resulting response curves are shown in Figures 4.3-5, 4.3-6 and 4.3-7. The coefficients chosen for each of these examples are shown in the figures. Note that each of the filters exhibits a significant response to spatial lines. Thus spatial lines of a given amplitude have the effect of appearing as points of smaller amplitude.

| D | C | C | C | D |
|---|---|---|---|---|
| C | B | A | B | C |
| C | A | P | A | C |
| C | B | A | B | C |
| D | C | C | C | D |

— 5 x 5 WINDOW
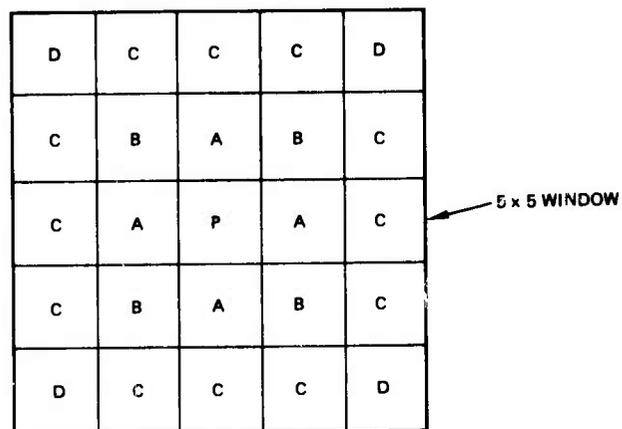
$$P_{OUT} = P_{IN} - K_A \Sigma A - K_B \Sigma B - K_C \Sigma C - K_D \Sigma D$$

Figure 4.3-3. Spatial filter
method 3.
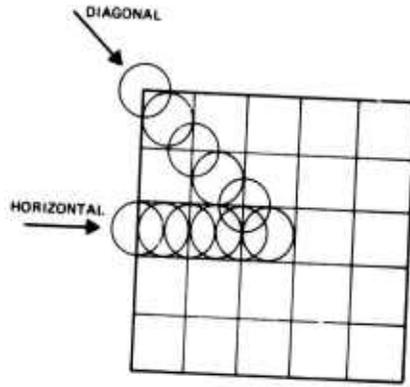
a) POINT STEP RESPONSE
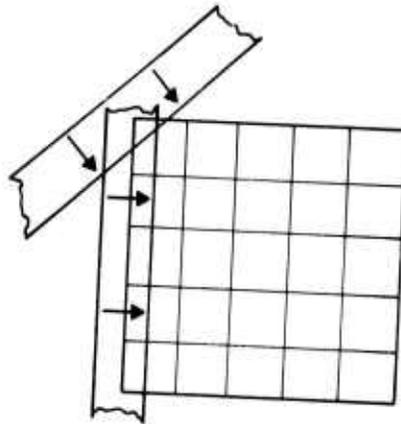
b) LINE STEP RESPONSE

Figure 4.3-4. Filter test cases.

$$P_{OUT} = P_{SN} - \frac{3}{4}\left(\frac{\Sigma A}{4}\right) - \frac{1}{4}\left(\frac{\Sigma B}{4}\right)$$

Figure 4.3-5.   Method 1.

$$P_{OUT} = P_{IN} - \frac{3}{4}\left(\frac{\Sigma A}{4}\right) - \frac{1}{4}\left(\frac{\Sigma B}{4}\right)$$

Figure 4.3-6. Method 2.

$$P_{out} = P_{IN} - .4\left(\frac{\Sigma A}{4}\right) - 3\left(\frac{\Sigma B}{4}\right) - .2\left(\frac{\Sigma C}{12}\right) - .1\left(\frac{\Sigma D}{4}\right)$$

POINT STEP HORRIZONTAL = PSH
POINT STEP DIAGONAL = PSD
LINE STEP HORRIZONTAL = LSH
LINE STEP DIAGONAL = LSD
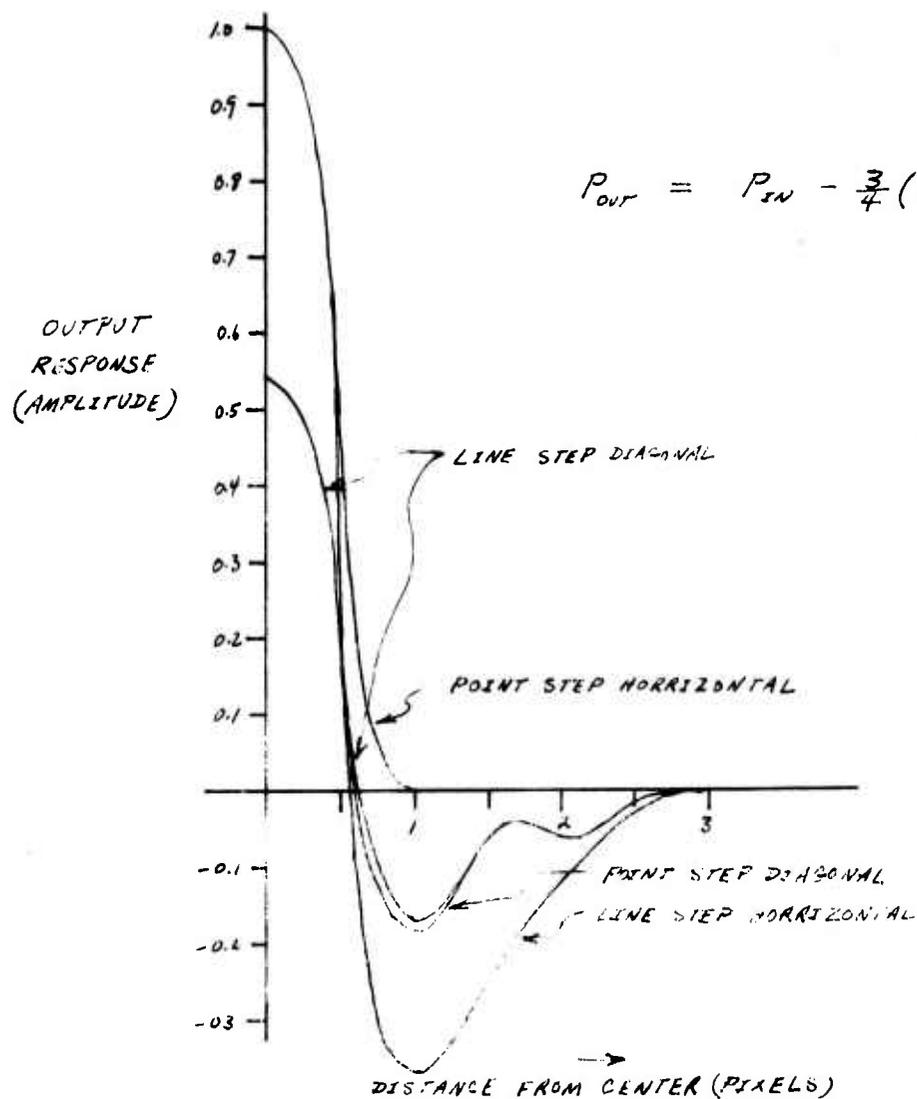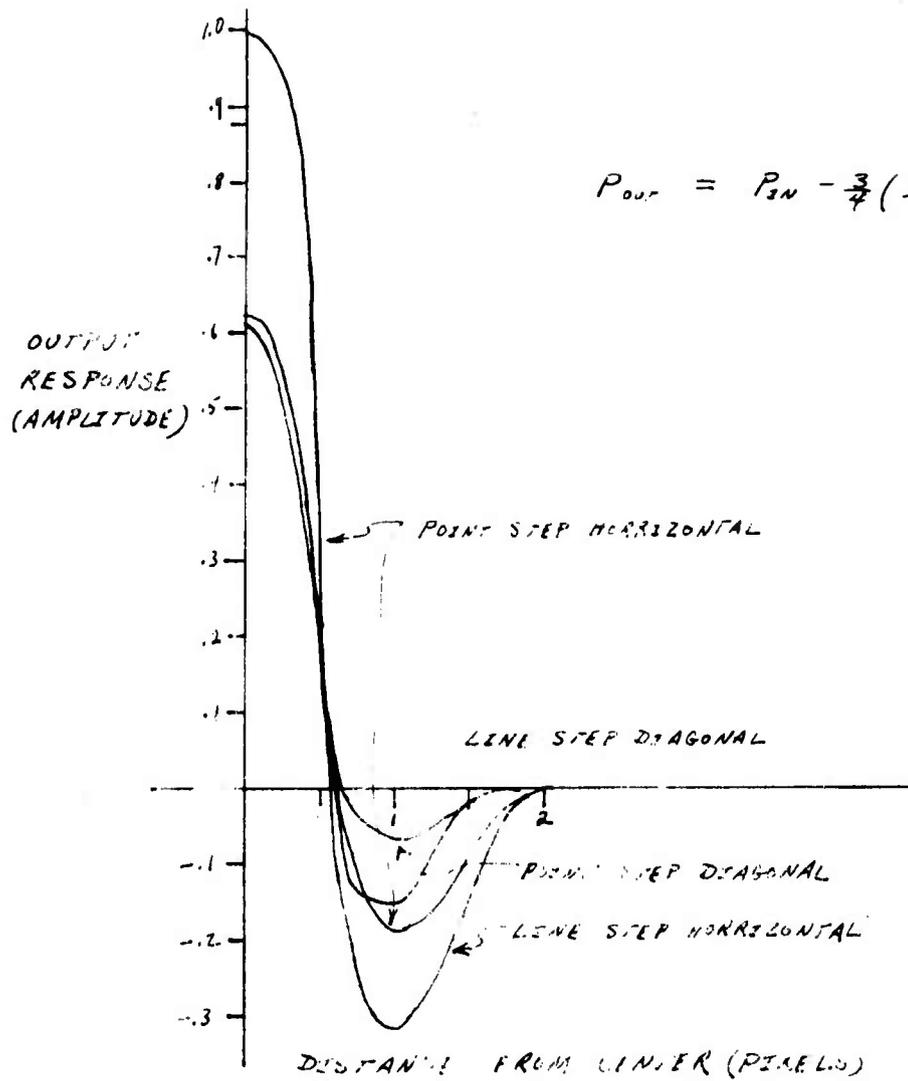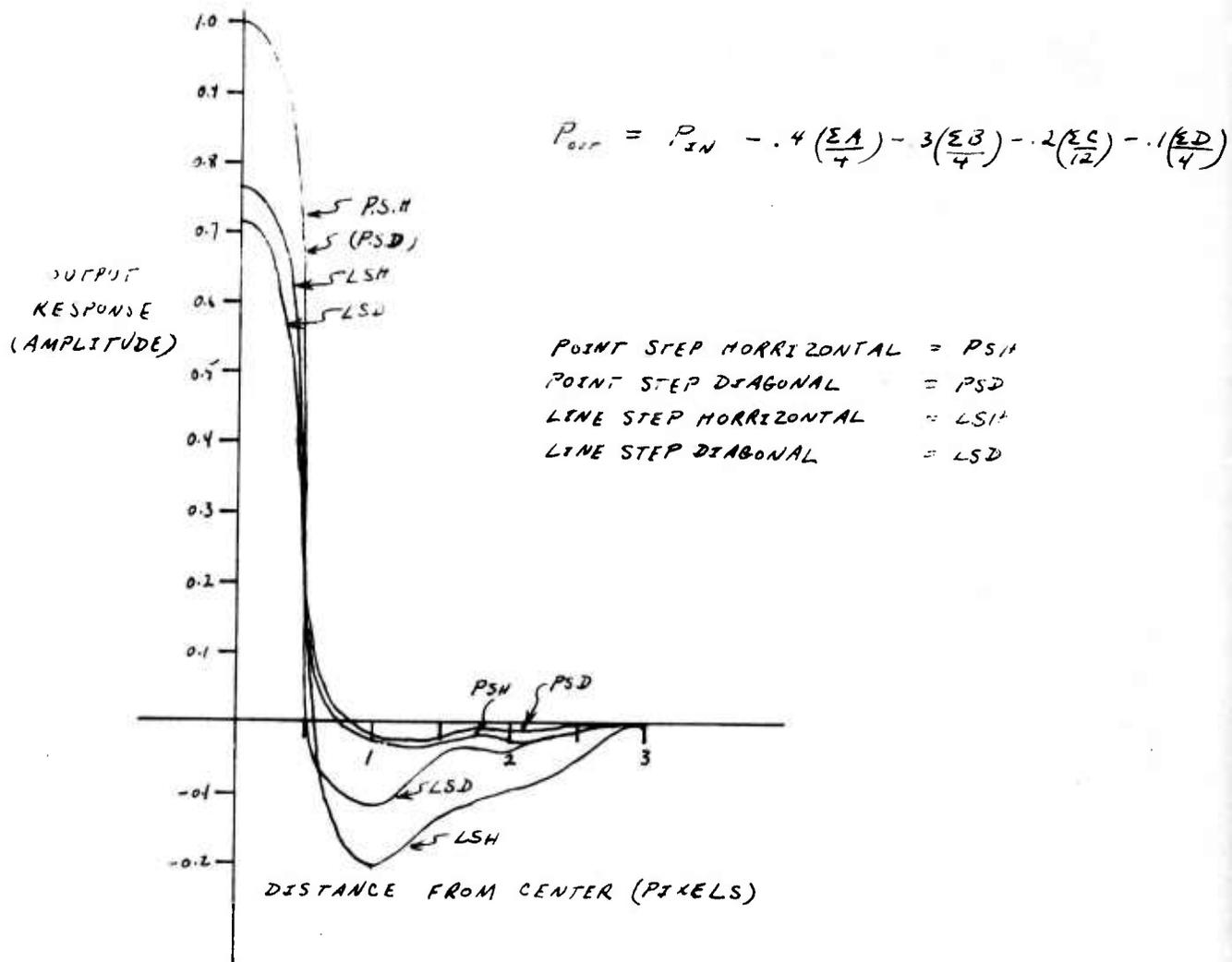
Figure 4. 3-7.   Method 3.

The basic goal of spatial filtering is to detect the presence of targets and to suppress background clutter.  The absolute amplitude of targets must be preserved during this process.  For a given filter, a tradeoff exists between preserving target amplitude and responding only to point targets.  It may not be possible to perform both of these functions satisfactorily with a single filter.  An alternative approach is to provide two filters, one for detection and one for background suppression.  This concept is shown in Figure 4. 3-8.  The output of the detection filter is used to gate the output of the background suppression filter.
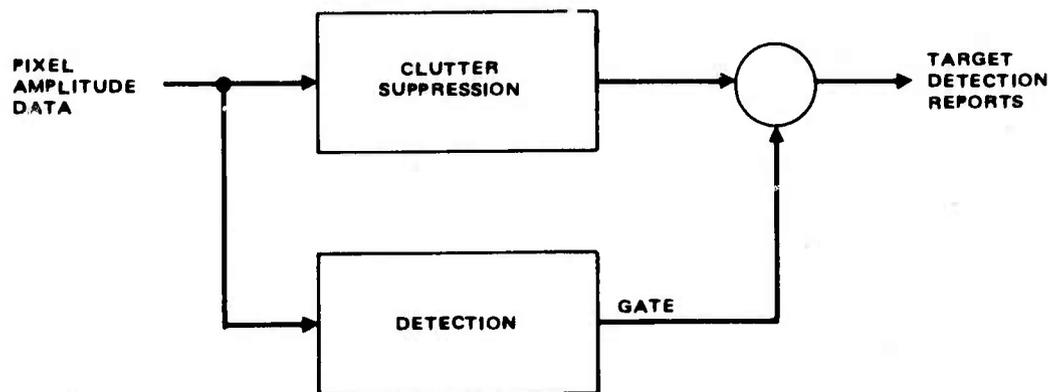
Figure 4.3-8.   Spatial filtering using separate filters
for detection and suppression.

A possible implementation of this scheme is shown in Figure 4.3-9.
Four one-dimensional filters are provided which are aligned along the
horizontal, the vertical and the two diagonals.   The output of each filter is
tested for a threshold excession to detect the presence of an amplitude peak
at the center.   If all thresholds are exceeded then the outputs of the four
one-dimensional filters are combined to form the output corresponding to
the center pixel.   With this implementation both the detection filter and the
suppression filter share the same arithmetic operations.
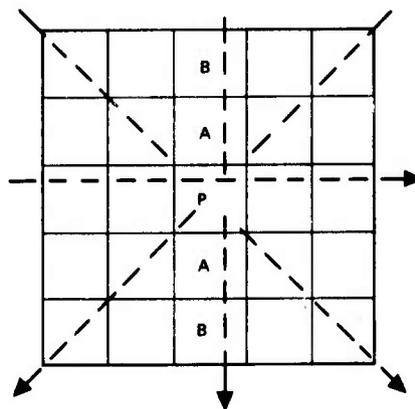


● $P_i = CP_{IN} \cdot C_A \Sigma A \cdot C_B \Sigma B$

WHERE i REPRESENTS DIRECTION
OF ONE-DIMENSIONAL FILTER

● TEST $P_i$ FOR THRESHOLD EXCESSION

● REPEAT FOR i = 1, 4

● $P_{OUT} = \Sigma P_i$

Figure 4.3-9.   Spatial filter
method 4.

4-20

## Hadamard Spatial Filtering

A block diagram of Hadamard spatial filtering is shown in Figure 4.3-10. This discussion is limited to Hadamard spatial filtering in one dimension, although the technique can be extended to two dimensions. First, a forward Hadamard transform is performed on a block of pixel amplitude data. The block length, N, is expected to be equal to the length of a single row of MFPA data which is 128. Spatial filtering is achieved by eliminating the low-order sequencies, so only sequencies 32 through 127 are generated when performing the forward transform. Weighted sums of the sequences are generated by the position filter. These sums represent the target energy incident on a pair of adjacent pixels.

Figure 4.3-11 shows the operations required for a forward Hadamard transform of block length 16. Pixel amplitudes at the top of the diagram are combined through addition and subtraction to form the sequency coefficients. This figure shows only half the operations required for a full transform. The same structure with the top row of addition operators changed to subtraction operations results in sequencies 8 through 15.

Once the sequencies have been generated, a type of inverse operation is performed to determine position amplitudes as shown in Figure 4.3-12. Each node labeled with a double subscript represents target energy incident on the corresponding pair of pixels within the block.

If the diagrams in Figures 4.3-11 and 4.3-12 are combined into a single network and the operations are minimized, Table 4.3-1 is obtained. Each row in this table represents the computationally equivalent operations which must be performed on the input amplitudes to obtain output amplitudes directly without performing a Hadamard transform. For example, to
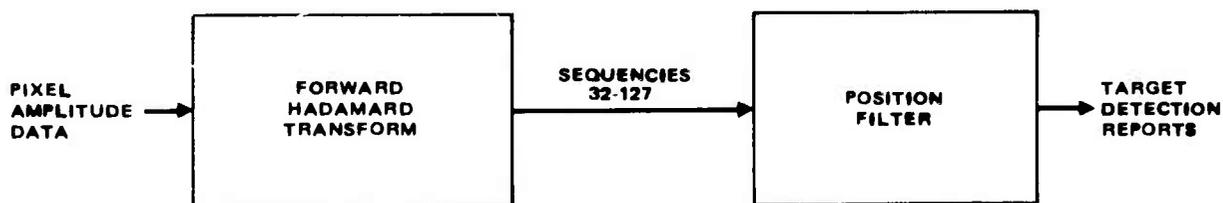


Figure 4.3.10. Hadamard spatial filter.

4-21
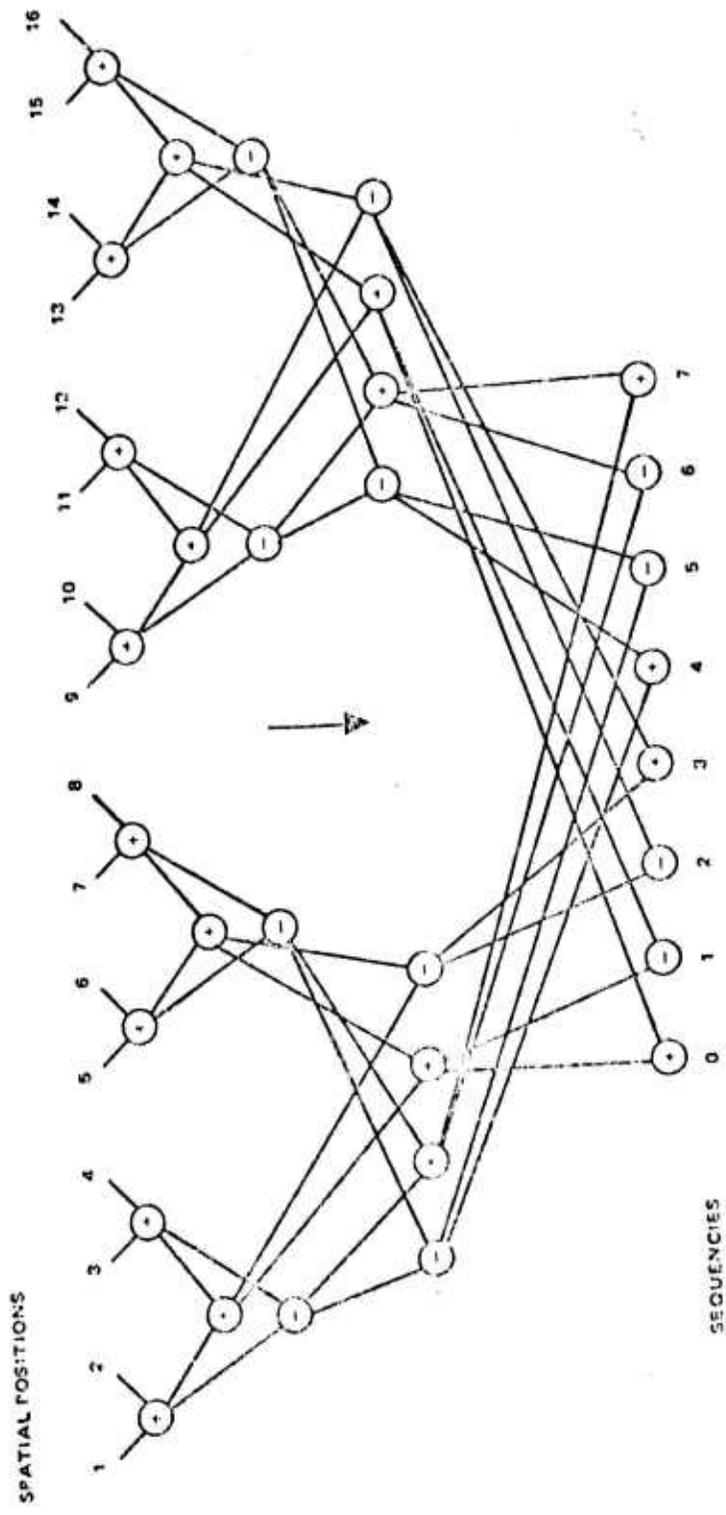
Figure 4.3-11. Forward Walsh Hadamard transform operations required to map spatial positions into sequences of block length N = 16.

OPERATIONS REQUIRED TO GENERATE POSITION
FILTER VALUES FROM SEQUENCES
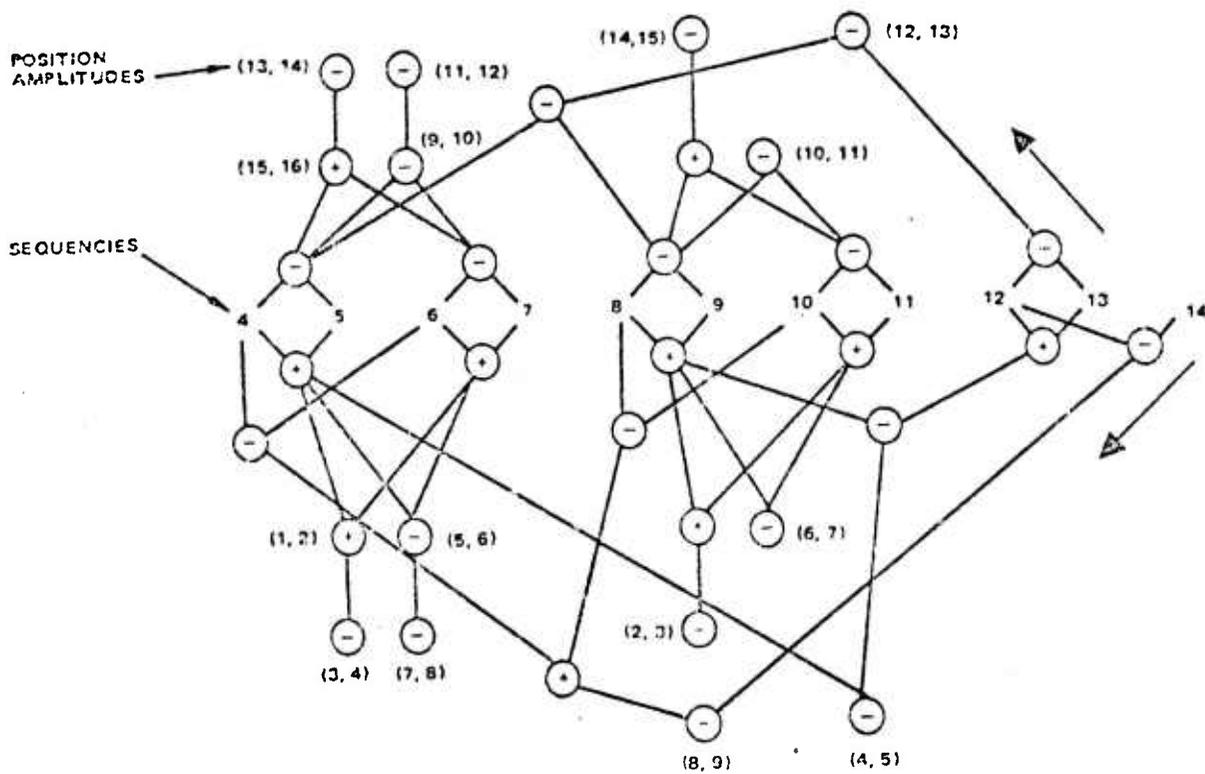
Figure 4.3-12.   Position filter.

TABLE 4.3-1.   WALSH HADAMARD SPATIAL FILTER OPERATIONS

Input Positions

| Position Filter Outputs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1,2 | 4 | 4 | -4 | -4 | | | | | | | | | | | | |
| 2,3 | -4 | 4 | 4 | -4 | | | | | | | | | | | | |
| 3,4 | -4 | -4 | 4 | 4 | | | | | | | | | | | | |
| 4,5 | -2 | -2 | -2 | 6 | 6 | -2 | -2 | -2 | | | | | | | | |
| 5,6 | | | | | 4 | 4 | -4 | -4 | | | | | | | | |
| 6,7 | | | | | -4 | 4 | 4 | -4 | | | | | | | | |
| 7,8 | | | | | -4 | -4 | 4 | 4 | | | | | | | | |
| 8,9 | | | | | -2 | -2 | -2 | 6 | 6 | -2 | -2 | -2 | | | | |
| 9,10 | | | | | | | | | 4 | 4 | -4 | -4 | | | | |
| 10,11 | | | | | | | | | -4 | 4 | 4 | -4 | | | | |
| 11,12 | | | | | | | | | -4 | -4 | 4 | 4 | | | | |
| 12,13 | | | | | | | | | -2 | -2 | -2 | 6 | 6 | -2 | -2 | -2 |
| 13,14 | | | | | | | | | | | | | 4 | 4 | -4 | -4 |
| 14,15 | | | | | | | | | | | | | -4 | 4 | 4 | -4 |
| 15,16 | | | | | | | | | | | | | -4 | -4 | 4 | 4 |

Each row in the above table contains the coefficients of the input position amplitudes which must be summed to generate the position filter output values directly without performing a Hadamard transform.

obtain $P_{(1, 2)}$ positions three and four are subtracted from positions one and two as follows:

$$P_{(1, 2)} = 4(J_1 + J_2) - 4(J_3 + J_4)$$

where

$J_i$ = amplitude of pixel i .

Note that either two or four of the nearest neighbors are required to generate the position amplitudes making this approach similar to local area pixel processing.

Table 4.3-2 shows the number of arithmetic operations required for each of the various phases of Hadamard spatial filtering. A substantial computational savings results from generating position filter values directly, as shown in Table 4.3-1, as opposed to the two phase Hadamard approach.

### Hardware Implementation

Figure 4.3-13 shows a block diagram of a local area pixel processor (spatial filter). This processor accepts digitized pixel amplitude data as input and performs sequential arithmetic operations to generate target detection reports. Detection thresholds are variable and are specified by the track processor. The line storage memory stores several lines of pixel amplitudes so that all pixels within a sliding window are accessible simultaneously. (For a 5 x 5 window, four lines must be stored.) The local storage RAM is a fast access memory used to store temporary variables and target detection records. The arithmetic and logical operations required to execute the spatial filtering algorithms are performed by the arithmetic unit. The threshold detection target detection registers comprise the interface with the track processor and receive threshold settings and report target detections, respectively.

For a frame rate of 10 Hz the processor must perform the calculations associated with a single pixel location within approximately 6 μsec. The algorithms discussed in the previous sections require from 5 to 50 operations per pixel. Assuming an overhead of 100 percent, this results in

4-25

TABLE 4.3-2. DIGITAL OPERATIONS REQUIRED FOR HADAMARD PROCESSING

| n | N (length of row vector) | Operations for full forward Hadamard | Operations for forward Hadamard excluding lower-order seqs. | Operations for Position Filter | Operations for Direct Position Filter |
|---|---|---|---|---|---|
| 1 | 2 | 2 | | | |
| 2 | 4 | 8 | 7 | | |
| 3 | 8 | 24 | 20 | 8 | |
| 4 | 16 | 64 | 52 | 33 | 60 |
| 5 | 32 | 150 | 128 | ~80 | 120 |
| 6 | 64 | 384 | 304 | ~200 | 240 |
| 7 | 128 | 896 | 704 | ~450 | 480 |
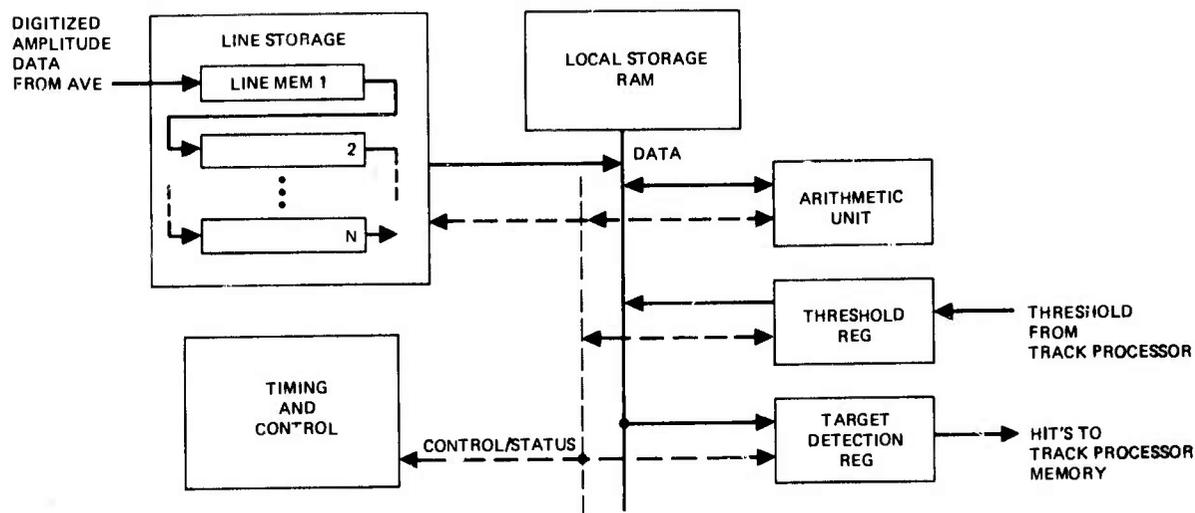
Figure 4.3-13. Spatial filter processor.

a range of 10 to 100 operations per pixel. This estimate imposes a performance requirement for a processor cycle time which ranges from 60 to 600 nsec. if the operations are to be performed sequentially. (Processor cycle time refers to the time required to perform a single operations such as addition or subtraction.)

The hardware complexity of a sequential spatial filter processor of the type shown in Figure 4.3-13 is estimated to be 30,000 to 40,000 gate equivalents. This assumes a 5 x 5 window size. A similar result could be obtained with a CCD implementation.

### Conclusion

A sequential processor capable of performing the spatial filtering associated with a single MFPA chip appears feasible for the algorithms described here. Further analysis is required before an optimum algorithm can be selected.

An appropriate target/clutter model must be formulated so that the optimum coefficients can be obtained and a performance estimate made for each of the algorithms. An algorithm can then be selected on the basis of performance vs. complexity (number of operations per pixel). The spatial filter implementation should minimize the number of operations per pixel by sharing arithmetic operations among the required subfunctions (i.e., clutter suppression, target detection, and thresholding).

4-27

## 4.4 Track Processor

This section describes the tracking multiprocessor system and the individual Microprocessor Trackers (μPT) which comprise the system that is positioned between the APSP Signal Processor and the Data and Control Processor (Figure 4.4-1). The Track Processor receives filtered data in the form of "hits" (potential targets) from the Signal Processor. These data undergo a special sorting procedure (commonly called tracking) in the Track Processor. Hits that appear to be the logical continuations of target tracks are used to update those tracks, while those that are found to be clutter are discarded. Data describing all tracks currently being monitored is sent periodically to the Data and Control Processor. Thresholding and algorithm selection commands may be sent back to the Signal Processor.

The volume of computation for the Track Processor can be estimated from the expected hit rate which in turn is estimated to be about 1 percent (i. e., 1 of every 100 pixels in the MFPA is reported as a hit by the Signal Processor). In the case of the largest contemplated MFPA ($10^8$ pixels), the volume of computation is too large to be handled by a single central processor. For this reason a number of multiprocessing and array processing methods have been examined.
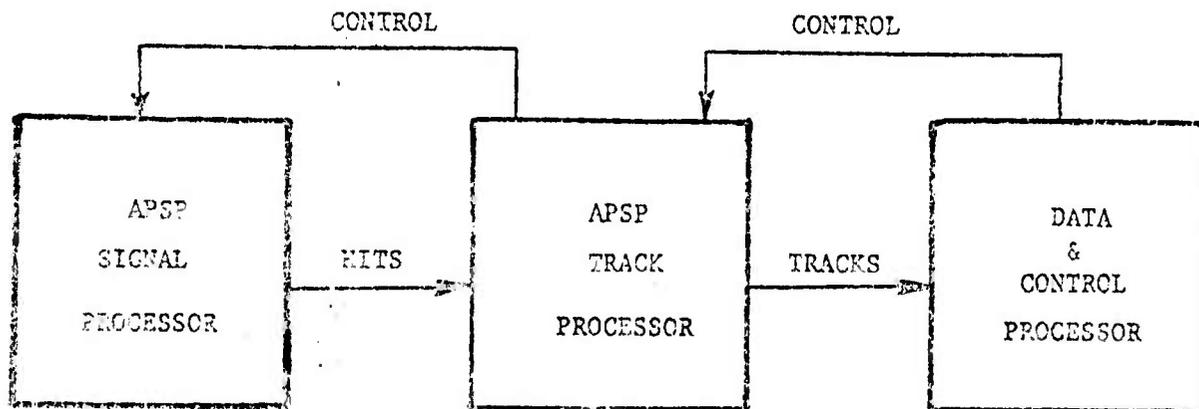


Figure 4.4-1. Position of the tracking multiprocessor in the system.

Two basic approaches have been examined:

1.   dynamic assignment of processors to tracks, and

2.   a priori assignment of processors to fixed regions of the
     focal plane.

While the first of these approaches is attractive because of dynamic resources allocation, it also presents difficultes in the areas of communication and fault tolerance. Such a solution requires universal communication between processor and/or a central supervisor coordinating all processors. The difficulty with this approach arises from the size of the communication networks required and from the lack of fault tolerance of systems with central supervisors. For these reasons the latter approach has been chosen.

### The King-Connected Array Processor

In the King-Connected Array Processor one processing element is assigned to each MFPA chip (128 x 128 pixels). Each processor communicates with 8 neighboring processors (like the moves of a king on a chessboard; hence the name of the configuration) (Figure 4.4-2). The sole purpose of interprocessor communications is to handle tracks that cross MFPA chip boundaries.

Since the Signal Processor is also partitioned into single processing elements for each MFPA chip, it is only necessary for each processing element of the King-Connected Array to communicate with the corresponding element of the Signal Processor.

Periodic updates of track status are sent from each processor element to the Data and Control Processor over a bus referred to as the Track Bus. Each processing element is periodically interrogated, at which time it reports the status of all the tracks it is monitoring.

### The Microprocessor-Tracker (μPT)

Each processing element of the King-Connected Array is a microprocessor and is referred to as a Microprocessor Tracker (μPT).

Figure 4.4-3 shows the connections between a μPT and its environment. In addition to the flows of data, the μPT receives external interrupts

4-29
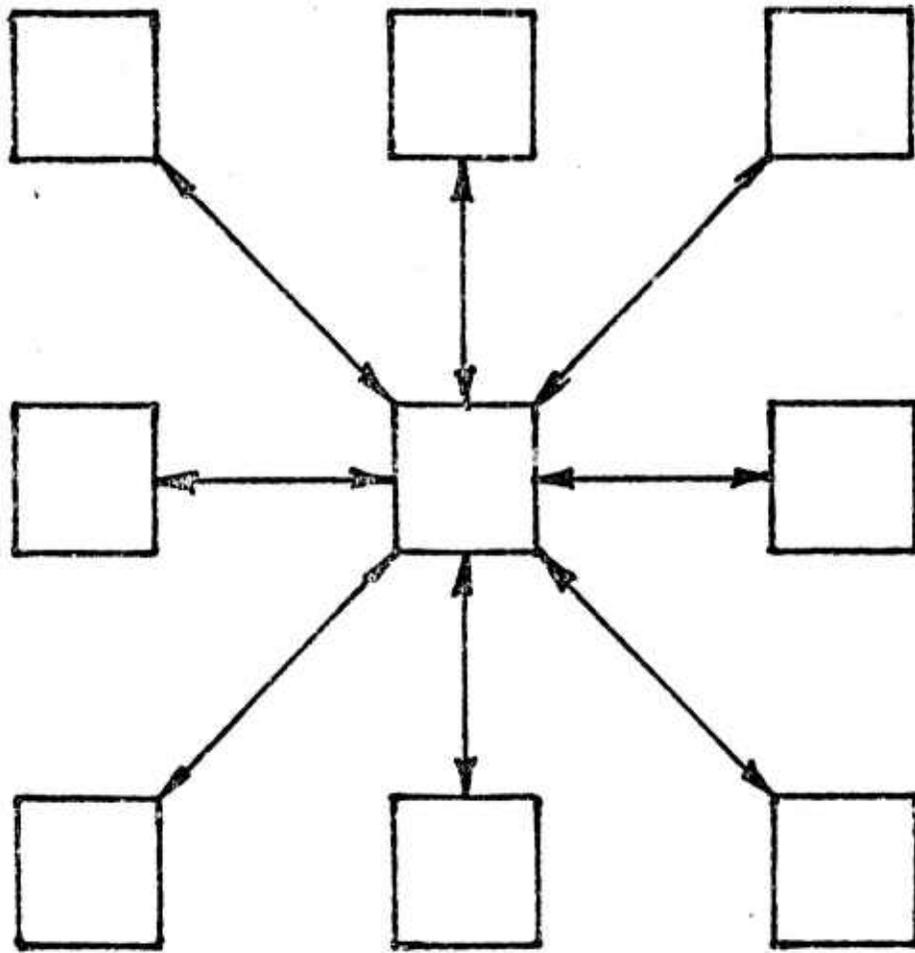
Figure 4.4-2. Interconnections in the king-connected array.
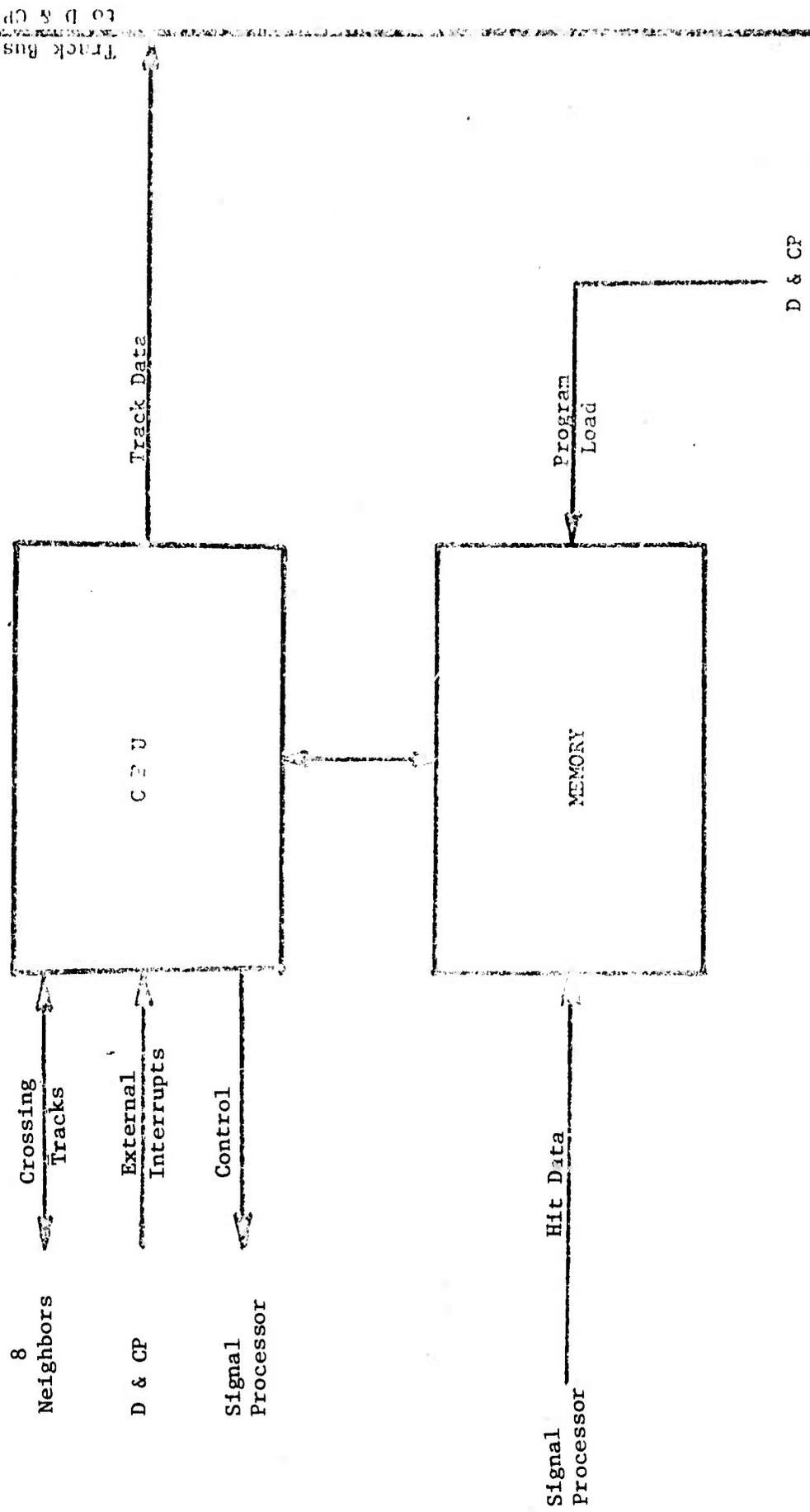
Figure 4.4-3.  Data flow in the μPT.

and can be loaded with a new program from the Data and Control Processor.

The µPT is a microprocessor tailored to the task of implementing one processing element of the Track Processor. For this reason its architecture exhibits special features not found in commercially available microprocessors.

### Special Features

Generally, the µPT is a bus-organized 16 bit processor. The special features of µPT are:

- 128 fast access (1 cycle) registers located on the arithmetic chip,
- 16 x 16 bit fully parallel multiply network (2 cycles),
- a 7-level priority interrupt structure,
- autonomous I/O interfaces, and
- a dual port, automatically switching, partially duplicated memory.

Each of these features is described in detail later in this section.

### Partitioning

The µPT consists of 5 chips (Figure 4.4-4).

a. the arithmetic chip which contains the 128 general registers, the arithmetic-logic unit (ALU), the multiply network, and related functional units;

b. the sequencing and I/O chip which contains the program counter, the interrupt structure, the memory accessing hardware and the autonomous I/O interfaces;

c. 2 memory chips, each containing 4K words, 16 bits each;

d. the microprogram control chip which contains the microprogrammed control unit for the entire µPT.
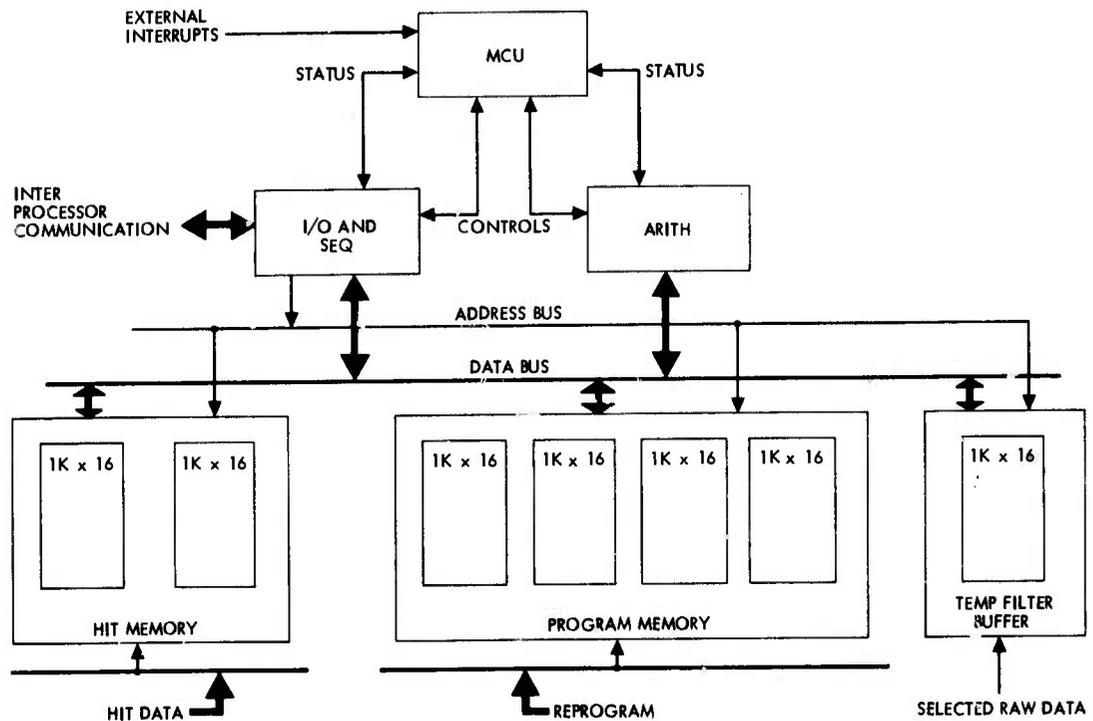
Figure 4. 4-4.   Partitioning of the µPT.

## Detailed Description of the Arithmetic Chip

The architecture of the arithmetic chip is shown in Figure 4. 4-5. Except for the various control and status lines, the only data path leading off this chip is the Main Bus over which memory data will be transmitted and received.

Internally the chip contains two busses: the Arithmetic Bus (A-Bus) which accommodates most register transfers on the chip, and the Iteration Counter Bus (I-Bus) which allows selection of inputs to the Iteration Counter (I). (The I-Bus may subsequently be replaced by a multiplexer if that is advantageous.)

The following is a description of the various functional units on the arithmetic chip.

## The General Registers

A group of 128 16 bit registers is provided for the user program. These registers offer fast access (1 cycle). Since the registers are under
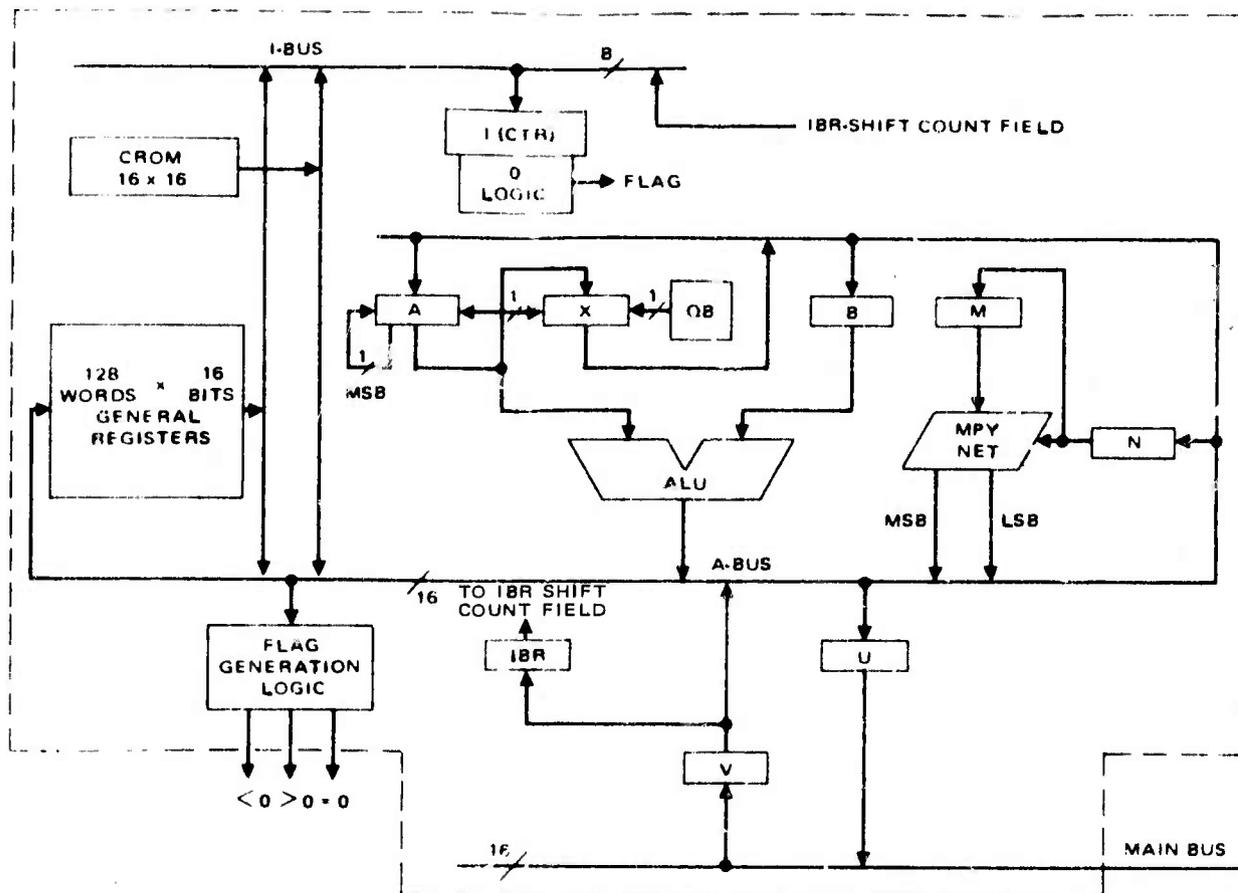
Figure 4.4-5. The arithmetic chip.

the control of the user program, addressing is done exclusively from certain fields of the Instruction Buffer Register (IBR). Data from the General Registers can be sent to the A-Bus or to the I-Bus. The General Registers can be loaded from the A-Bus.

### The Constant ROM (CROM)

This read-only-memory contains certain constants and masks necessary in the interpretation of the instruction set. The CROM is addressed by the microprogram. The CROM is 16 bits wide and its length is estimated to be 16 words. Data from the CROM can be sent to the A-Bus or to the I-Bus.

### The U and V Registers

These two registers are the transfer buffers between the Main-Bus and the A-Bus. Each is 16 bits wide. The IBR can be loaded from the V-register.

### The Iteration Counter (I)

The Iteration Counter is used in the implementation of iterative instructions, such as shifts, block moves, division, etc.; it is an 8-bit up-counter. At the beginning of an iterative algorithm I is loaded with a negative value (-1 to -128) and is then counted up until the zero is reached. Detection of the zero condition is thus reduced to monitoring of the sign bit. The Iteration Counter can be loaded from the I-Bus. For shift instructions, the value of the shift-count field of the IBR is transferred to the Iteration Counter. For block move instructions the length of the block will be transferred from a General Register to the Iteration Counter. For division the initial value of the Iteration Counter will be transferred from the CROM.

### The Arithmetic-Logic Unit (ALU)

The ALU has two 16 bit inputs designated A (left) and B (right). The output of the ALU is one of the following functions of A and B:

A

B

A + B

A - B

A.OR.B

A.AND.B

A.XOR.B

$\overline{A}$

-B

all 0-s

all 1-s

In addition to the 16-bit result, the ALU detects overflow for the operations

A + B

A - B

-B.

The operation to be performed by the ALU is selected by 4 control lines from the microprogram.

The propagation delay through the ALU is very short (under 4 nsec), thus allowing ample time for storing the result during the same cycle.

### The A, X and B Registers

These registers are all 16 bits wide.  The A and B registers serve as the A and B input to the ALU.  Both can be loaded from the A-Bus.  The X-register is used in division and for shift instructions together with the A-register.  The X-register can be loaded only from the A-register and the contents of the X-register can be transmitted directly over the A-Bus.

The A and X registers can be shifted at the rate of 1 bit/cycle.  They can be shifted right or left as one unit.  For left shifts the carry-in into the right end of the X-register is controlled by the Quotient Bit logic network.  For right shifts, sign extension is provided on the left end of the A-register.

### The Multiply Network

The Multiply Network facilitates fully parallel multiplication of two 16-bit numbers.  The result is valid after 2 cycles (50 ns).  The Multiply Network has two 16-bit inputs (multiplicand and multiplier) and two 16-bit outputs (most and least significant bits of the result) to the A-Bus.

### The M and N Registers

These two registers are each 16 bits wide and serve to hold the multiplicand and multiplier for the Multiply Network.  The N-register can be loaded from the A-Bus directly.  The M-register can only be loaded from the N-register.

### The Instruction Buffer Register (IBR)

The IBR holds the instruction currently being processed.  It is 16 bits wide and can be loaded from the V-register.

The IBR is divided into fields as shown in Figure 4.4-6.  The Accumulator pointer field is used to address one of the first eight General Registers (0 through 7).  The General Register pointer field can be used to address any General Register.  The shift count field can be transmitted to the Iteration

OP = Op-code
ACC = Accumulator pointer
REG = General Register pointer
SC = Shift count
R/L = right/left indicator bit
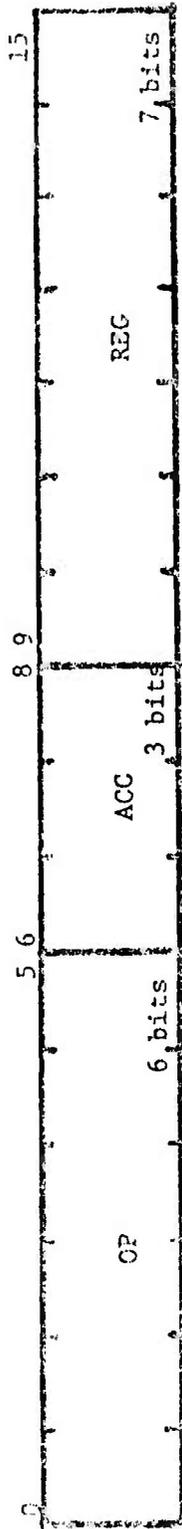


Figure 4-6. Fields in the instruction buffer register.

Counter over the I-Bus. The right/left indicator bit controls the A and X registers' shift direction. The op-code field is used by the Microprogram Control Unit (MCU) in instruction decoding.

### Flag Generation Logic

This logic network monitors the value on the A-Bus. Three flags are generated:

A-Bus = 0
A-Bus > 0
A-Bus < 0.

All three flags are used by the MCU for branching.

### The A-Bus

Table 4.4-1 summarizes the inputs and outputs of the A-Bus.

TABLE 4.4-1. INPUTS AND OUTPUTS OF THE A-BUS

| Inputs | Outputs |
|--------|---------|
| General Reg. | A-register |
| CROM | B-register |
| ALU | N-register |
| Multiply Net MSB | U-register |
| Multiply Net LSB | Flag generation logic[*] |
| V-register | |
| X-register | |
| 7 inputs | 4 + 1 outputs |
| [*]Always receiving. | |

The I-Bus

Table 4.4-2 summarizes the inputs and outputs of the I-Bus.

TABLE 4.4-2.   INPUTS AND OUTPUTS OF THE I-BUS

| Inputs | Outputs |
|---|---|
| General Reg.<br><br>CROM<br><br>IBR | I-Counter |
| 3 inputs | 1 output |

### 4.4.2b   The Sequencing and I/O Chip

The architecture of the sequencing and I/O chip is shown in Figure 4.4-7.   The following data paths are leading off this chip:

a.   Main Bus (16 bits)

b.   Address Bus (13 bits)

c.   Input to Track-Bus (16 bits)

d.   Two-way to neighbors (8 bits)

e.   Output to Signal Processor (1 bit)

f.   Input from previous Vector Buffer Controller (1 bit)

g.   Output to subsequent Vector Buffer Controller (1-bit)

The above list does not include control and status lines.   The Main-Bus is used to transmit and receive memory data.   The Address Bus is used to send memory addresses to the memory chips.   The Track-Bus is not under the control of the μPT.   The data accumulated in the Vector Buffer is periodically sent out over the Track-Bus.

Internally, the sequencing and I/O chip contains another bus, the Program Counter Bus (PC-Bus), which allows selection of inputs to the Program Counter (PC).   (The PC-Bus may subsequently be replaced by a multiplexer if that is advantageous.)

4-39

Figure 4.4-7. The sequencing and I/O chip.

The sequencing and I/O chip can be functionally subdivided into 4 parts:

(1)  the memory addressing function,

(2)  the interrupt structure,

(3)  sequencing mechanism, and

(4)  I/O.

The following is a description of these various functional units on the sequencing and I/O chip.

### (1)  Memory Addressing Function

There are three sources of memory addresses in the µPT:

- the Program Counter,
- addresses in the instruction stream, and
- indirect addresses.

Addresses in the instruction stream may be indexed.

The following sections describe in detail how memory address selection is implemented in the µPT.

### The Memory Address Register (MAR)

The Memory Address Register consists of a 13-bit address and an "indirect" bit. They correspond to the 13 LSB and the MSB of the 16 bit word, respectively. The MAR can be loaded from the Main Bus only.

### The Index

The Index is a 13-bit register used to hold the contents of one of the General Registers, number 1 through 7, when indexed addressing is used. The Index is loaded from the Main-Bus only.

4-41

## The Index Adder

The Index Adder is used for every memory access where the address originates from the MAR (indexed or unindexed). The Index Adder can produce 2 possible results:

> INDEX + MAR

or

> MAR.

The output of the Index Adder is 13 bits wide and can be transmitted to the memory chips via the Address-Bus.

### (2)  Interrupt Structure

## The Interrupt Vector (IV)

The IV consists of a 7-bit register and associated logic used to set the register. The IV can be loaded from the Main-Bus. The principal use of the IV however is to record the occurrence of any of 7 interrupt levels. A "1" in a certain bit position of the IV indicates the occurrence of that interrupt.

## The Interrupt Mask (IM)

This 7-bit register can be used to suppress the servicing of any interrupt levels. The contents of the IM are ANDed with IV before any further decisions are made. The 7-bit number thus obtained is encoded as shown in Table 4.4-3 and the 3-bit code thus obtained is the highest current interrupt level.

## The Level Register

This 3-bit register holds a value equal to level of the interrupt being processed currently. When no interrupt is being processed (normal program flow) the value of the level register is zero.

TABLE 4.4-3. ENCODING OF INTERRUPT LEVELS

| IV. AND. IM | Encoded Value | |
| --- | --- | --- |
| | Decimal | Binary |
| 0000000 | 0 | 000 |
| 0000001 | 1 | 001 |
| 000001X | 2 | 010 |
| 00001XX | 3 | 011 |
| 0001XXX | 4 | 100 |
| 001XXXX | 5 | 101 |
| 01XXXXX | 6 | 110 |
| 1XXXXXX | 7 | 111 |

Interrupts occur whenever the highest current interrupt level is higher than the value in the Level register. For this purpose a comparator continuously compares the value in the Level register with the highest current interrupt level. The signal (INT) thus generated is monitored by the Microprogram Control Unit.

(3) Sequencing Mechanism

The Program Counter (PC)

The PC is a 13-bit up-counter. It can be loaded from the PC-Bus. The PC contains the address of the next instruction to be executed. Addresses from the PC can be transmitted to the memory chips over the Address Bus. The contents of PC can also be pushed onto the Interrupt Stack when an interrupt occurs.

The PC can be loaded from various sources via the PC-Bus:

- from the MAR (for branch instruction),
- from a Trap Address Cell (for interrupts),
- from the Interrupt Stack (for resumption after interrupts), and
- from a hardwired bootstrap address (for cold starts).

### The Trap Address Cells (TAC)

There are 7 Trap Address Cells, each corresponding to one interrupt level. Each TAC contains a memory address and whenever the respective level of interrupt occurs, interrupt handling is started at that address.

The TAC's consist of 7 registers, each 12 bits wide. It has been mentioned that the value of one of the TAC's can be sent to the PC over the PC-Bus. Selection of which TAC to use is based on the value in the Level Register.

The TAC's can be loaded from the Main-Bus. For this purpose a 15-bit quantity consisting of a 3-bit TAC select code and a 12-bit trap address is sent to the TAC's. Hardware associated with the TAC's will load the proper cell.

### The Interrupt Stack

The Interrupt Stack consists of 7 cells (corresponding to the maximum possible 7 nested interrupts).

Each cell consists of a 12-bit resumption address and a 3-bit resumption level. Before an interrupt is serviced, the contents of PC and of the Level register are pushed onto the Interrupt Stack. When the servicing of an interrupt is completed, the value from the top of the Interrupt Stack is popped off and placed into the PC and Level register.

Pushing and popping of the Interrupt Stack is done by means of a Top of Stack Counter (TSC) which is used to address the seven cells. To push, TSC is incremented and then the value is stored. To pop, the value is read off the stack and TSC is decremented.

### (4)  I/O

### Vector Buffer (VB)

The Vector Buffer represents the I/O interface with the Track-Bus. The VB contains one cell of storage for each track. Currently it appears that 32 such cells will be sufficient. Each cell will be composed of a

number (n) of 16-bit words. A controller will allow the following functions to be performed:

a. Receive a command indicating where the next n words are to be placed.

b. Receive n 16-bit words following a command.

c. Receive a pulse from another μPT indicating that all valid data in VB is to be sent out on the Track-Bus.

d. Send out on Track-Bus all valid data in VB and mark entire VB as empty and available.

e. Send a pulse to another μPT indicating that this μPT is done using the Track-Bus.

### Threshold Control Register (TCR)

This register is parallel input/serial output organized. It represents the I/O interface between the μPT and the corresponding Signal Processor.

### Message Control Network

The Message Control Network (MCN) together with the Incoming Message Register (IMR), and the Outgoing Message Register (OMR) form the I/O interface with the 8 neighboring μPT's.

To send a message to a neighbor, the OMR is loaded and thereafter the MCN handles I/O while the processor continues executing the instruction stream.

Incoming messages are handled by the MCN without processor intervention until the message is placed into the IMR. At that point an interrupt is sent to the interrupt structure.

### The Memory Chip

Before discussing the organization of the memory chip it is necessary to restate the functions of the memory as a whole. A memory of 7K words of 16 bits each is required. The first 5K words are dedicated to store programs, constants and variables. The next 1K words of memory comprise a special purpose memory dedicated to the task of acquiring data for software temporal filtering. The last 1K words are dedicated to store the hits of the latest frame and are duplicated. The duplicate memory is isolated

4-45

from the processor address space and is loaded with hit data from the Signal Processor over the alternate memory interface. At the end of the frame time the two 1K-memories exchange rolls.

The first 5K words can also be loaded from the alternate interface. The Data and Control Processor may, at its discretion, load new programs into any or all μPTs. A bit-serial interface is provided for that purpose on the alternate interface.

It should be noted that while accesses to the memory coming from the processor are random, program loading, as well as data loading from the Signal Processor, are both sequential in nature.

The architecture of the memory chip is shown in Figure 4.4-8. The chip was designed so that only one type of chip need be developed. Thus the cost of developing two or more chip types is eliminated.
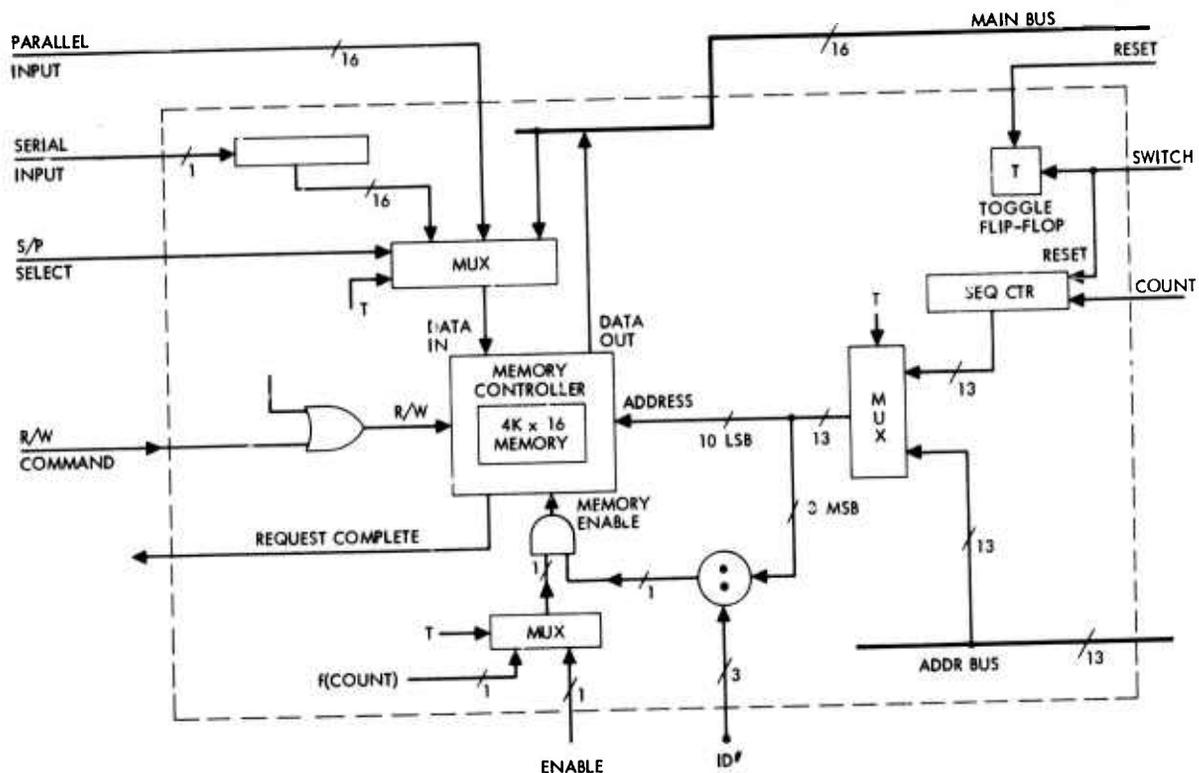


Figure 4.4-8. The memory chip.

### The Memory Proper

Within the memory chip lies the memory storage area and memory controller. The storage area is organized as 4096 words with 16 bits per word. The memory controller is the logic required to coordinate memory operations, such as Read/Write, address decode, request complete indication, etc. Thus, this chip constitutes a complete memory unit.

To access the memory, a predetermined set of procedures must be followed. The signal to read or write must be set up along with the address. Following this, the enable to the chip is activated. The memory then performs the read or write and acknowledges completion of the task. For read operations the data is enabled onto the main bus at the same time the completion signal is generated. To change the address and R/W signals, the chip enable must be disabled to protect the memory contents from being destroyed.

### Chip Identification

Note that the entire address (13 bits) is connected to all memory chips. However, each chip contains only 1K of address space. The 1K space requires 10 bits of address, and the I.D. scheme is as follows: The three MSB of the address are compared to the chip I.D. A match implies that the desired address (13 bits) lies on this chip. A mismatch implies that this chip does not contain the desired address. Hence, each memory chip will have a unique I.D. (with the exception of the hit memories). Since only one chip will match the three MSB at any access, no conflicts will arise. Also notice that the chip I.D. match logic is tied into the chip enable logic to further protect the memory contents.

### Data I/O

There exist three data inputs to the memory chip and one data output. The primary I/O channel is the Main Bus. Data sent across the Main Bus will be stored if so desired, or, if a read was requested, data will be put on the Main Bus.

The two other inputs constitute the alternate interface. One data path is a 16-bit parallel input and is used for signal processor data. The other path is a single line bit serial path to be used for program load. The serial input is converted to a parallel format on the chip and then stored in the memory.

### Addressing

Two address sources are present to the memory, the Address Bus and the Sequence Counter on the chip.

The Address Bus supplies an address to the memory chip from the CPU. The sequence counter is used when one of the write-only ports is being used. Prior to data transmission the counter is reset. Each data word is then stored at an incrementally higher address as specified by the counter. The external system controls the count enable line to the Sequence Counter.

### The 'T' Flip Flop

The 'T' flip-flop controls the current interface to the memory proper. In the zero state, the primary interface to the main bus is enabled and all signals on the other interface are ignored. In the one state, the secondary interface is active while the primary is disabled.

In the program memory, the T is set prior to loading a new program and is under the control of the DCP. In the hit memory, the T is under the control of the unit sending data to the hit memory and will be toggled at the start of each new frame of data.

### The Microprogram Control Unit (MCU)

The architecture of the MCU is shown in Figure 4.4-9. The following data paths are leading off the MCU chip:

a.    Encoded Control Signals to all other chips of the $\mu$PT.

b.    Status Flags from all other chips of the $\mu$PT.

c.    Op-code from IBR on Arithmetic Chip.

d.    INT-flag from Sequencing and I/O chip.
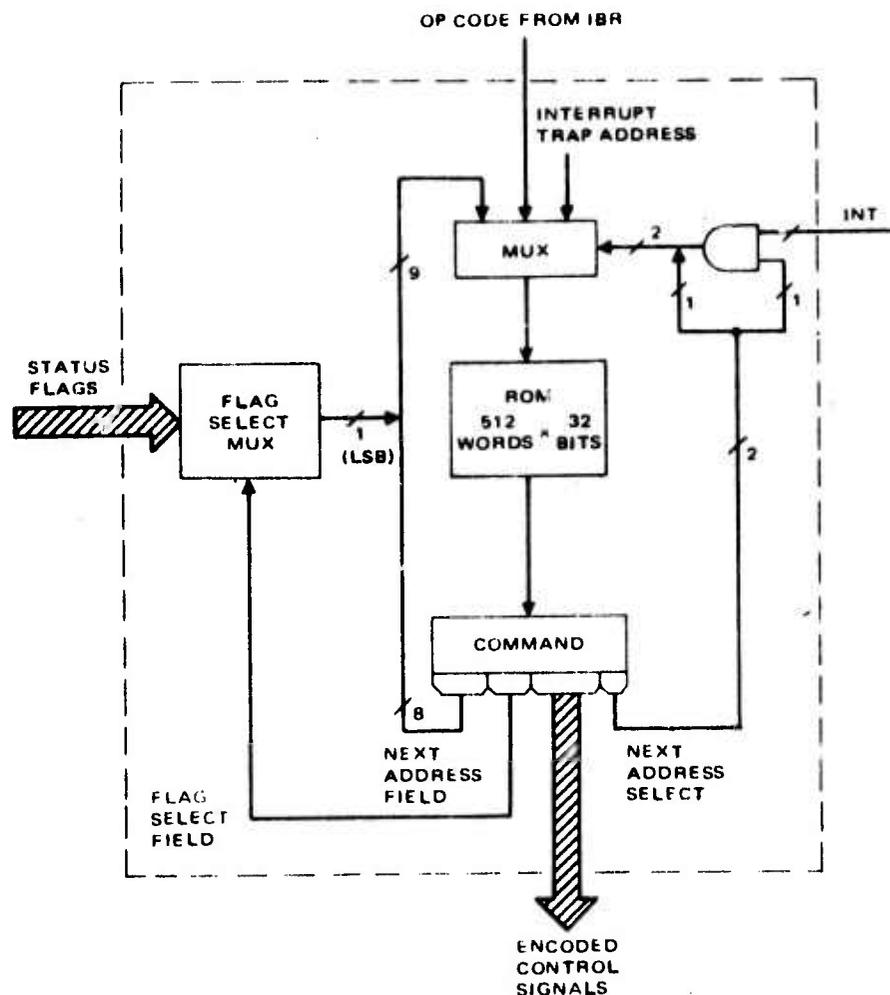
Figure 4.4-9. The microprogram control unit.

### The ROM and the Command Register

The MCU primarily consists of a 512 word x 32 bit ROM and the Command Register which can hold one word from the ROM. During each minor cycle one word is read from the ROM and placed in the Command Register.

### The Address Multiplexer

The 9-bit address of the next ROM word to be fetched can come from one of 3 possible sources:

    a.   8 bits from the Next Address Field of the Command Register, concatenated with one bit based on a status flag; or

b. 6 bits from the Op-code field of the IBR with three zeros as MSB's; used for instruction decoding; or

c. a hardwired address used to branch into a section of the microprogram which is dedicated to trapping interrupts.

Selection between these sources is made by means of the Address Multiplexer. The selection is based on three control bits: one is the INT signal from the Interrupt Structure; the others originate from the Command Register.

Most of the time the address based on the Next Address Field is selected. Only once in the execution of each instruction do the other sources come into play. Whenever a new instruction has to be decoded the INT-signal will select between the Op-code (for instruction decoding of INT=0) or the interrupt trap address (if INT=1).

### The Flag Select Multiplexer

This Multiplexer allows selection of one of the many status flags from the µPT in order to be appended to the Next Address Field. The controls for the multiplexer originate in the Command Register.

### The Instruction Repertoire of the µPT

The µPT has an instruction set consisting of 40 instructions. Most instructions occupy one word in the memory but some are doubleword instructions. Indexing and indirect addressing is available on some instructions.

There are three basic instruction formats as shown in Figure 4.4-10. These are:

- register — register (RR)
- register — memory (RM)
- register — shift (RS).

Figure 4.4-11 presents a summary of the instruction set. The following paragraphs describe the instruction set in detail.
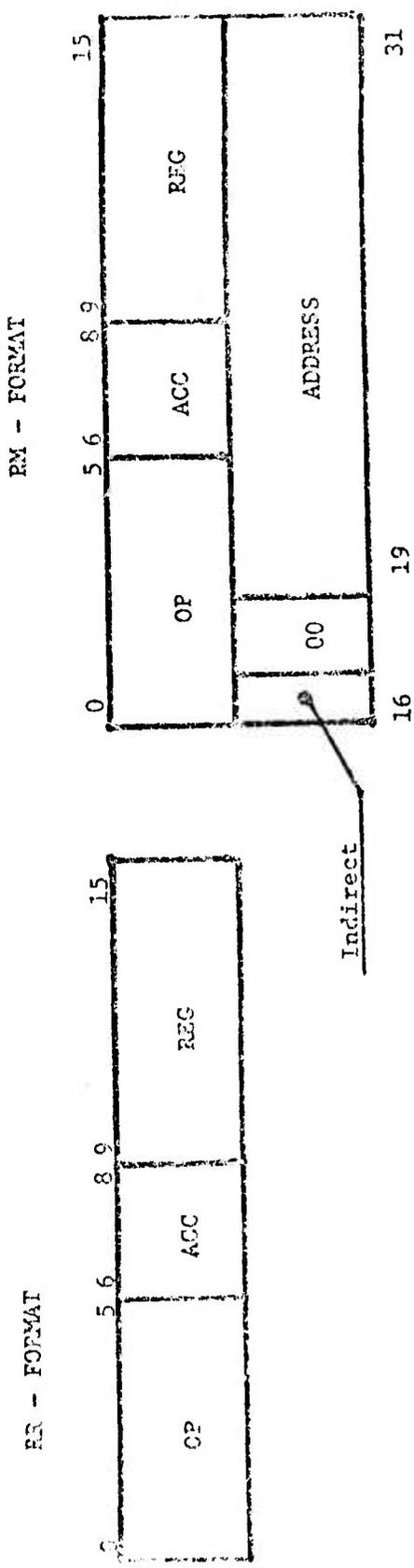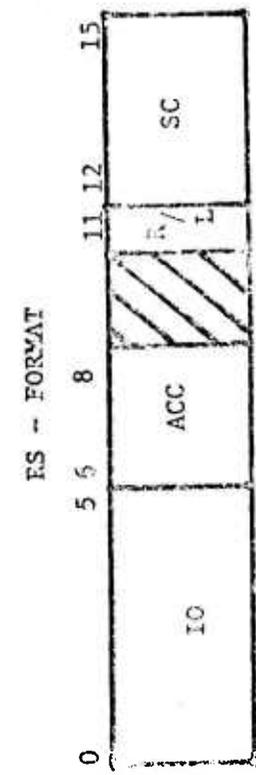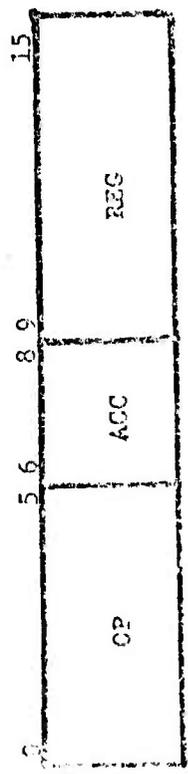
Figure 4.4-10. Instruction formats.

DATA MOVING

LOAD
STORE
EXCHANGE
BLOCK LOAD
BLOCK STORE
REG. TO ACC.
ACC. TO REG.
LOAD IMMEDIATE

BRANCHES
UNCONDITIONAL

IF REG = 0
IF REG = ACC
IF REG > ACC
IF REG < ACC
INCR ACC. & BRANCH IF < REG
IF ERR ON PREVIOUS INSTR.

LOGICAL

AND
OR
XOR
1'S COMPLEMENT

ARITHMETIC

ADD
SUBTRACT
MULTIPLY
DIVIDE
INCREMENT REG.
DECREMENT REG.

SHIFT

ARITHMETIC
ARITHMETIC DOUBLE
ROTATE
LOGICAL 0-EXT.
LOGICAL 1-EXT.

INTERRUPT HANDLING

SET TRAP ADDRESS
SET MASK REGISTER
RESUME

MISCELLANEOUS

HALT
NO-OP
RESET AND START

I/O

OUTPUT TO VECTOR BUFFER
OUTPUT TO SIGNAL PROCESSOR
OUTPUT TO NEIGHBOR
INPUT FROM NEIGHBOR

Figure 4.4-11. The instruction set of the μPT.

### (1) Data Moving Instructions

#### Load (RM)

The contents of the memory address indicated in the second word of the instruction are loaded into the General Register specified by REG. If the ACC is not zero, indexing will be performed using the General Register indicated by ACC as the index. Indirect addressing can be specified by placing a '1' in the most significant bit of the second word of the instruction.

#### Store (RM)

The contents of the General Register specified by the REG field are stored in the memory. The memory address is computed as for LOAD.

#### Exchange (RM)

The contents of the General Register indicated by REG are exchanged with the contents of the memory location specified. The memory address is computed as for LOAD.

#### Block Load (RM)

A block of data from consecutive memory locations is loaded into consecutive General Registers. The beginning of the data in memory is specified by the address in the second word of the instruction, which may be an indirect address. Indexing is not available. The REG field specifies the General Register where loading is to begin. The right half of the General Register indicated by ACC contains the length of the data block to be moved. The length of the block must be between 1 and 256. A length of 256 is indicated by 0. (The left 8 bits of the General Register are ignored.)

#### Block Store (RM)

Like Block Load except the data is stored, not loaded.

#### Load Immediate (RM)

Unlike other RM-format instructions, the second word of the instruction here is data rather than an address. The data from the second word

is loaded into the General Register specified by the REG field. The ACC field is not used.

### Register to Accumulator (RR)

The contents of the General Register specified by the REG-field are copied into the General Register specified by the ACC-field.

### Accumulator to Register (RR)

The contents of the General Register specified by the ACC-field are copied into the General Register specified by the REG-field.

### (2) Arithmetic Instructions

All arithmetic operations assume a fractional 2's complement number system. The binary point is implied between the sign-bit and the bit to its right:

S. XXXX...X.

### Add (RR)

The contents of the General Register specified by ACC are added to the contents of the General Register specified by REG and the result is placed into the General Register specified by REG. Overflow may occur.

### Subtract (RR)

The contents of the General Register specified by ACC are subtracted from the General Register specified by REG and the result is placed into the General Register specified by REG. Overflow may occur.

### Multiply (RR)

The contents of the General Registers specified by REG and ACC are multiplied and the double precision product consisting of two signed 16-bit numbers is placed into the General Register specified by REG and into the one immediately after it.

### Division (RR)

The double precision number from the General Register specified by REG and the one immediately after it are divided by the contents of the General Register specified by ACC. The quotient is a single precision number and is placed into the General Register specified by REG. Overflow may occur.

### Increment (RR)

The value of the General Register specified by REG is incremented. The ACC field is not used. Overflow may occur.

### Decrement (RR)

The value of the General Register specified by REG is decremented. The ACC field is not used. Overflow may occur.

#### (3) Logical Instructions

### And (RR)

The contents of the General Registers specified by REG and ACC are ANDed and the result is placed into the General Register specified by REG.

### Or (RR)

The contents of the General Registers specified by REG and ACC are ORed and the result is placed into the General Register specified by REG.

### Exclusive Or (RR)

The "exclusive or" of the contents of the General Register specified by REG and ACC is computed and the result is placed into the General Register specified by REG.

### 1's Complement (RR)

The contents of the General Register specified by REG are 1's complemented. The ACC field is not used.

(4) Branch Instructions

## Branch Unconditionally (RM)

The next instruction to be executed is the one specified by the address in the second word of the branch instruction. This address may be indirect. The REG and ACC are not used.

## Branch if Register is Zero (RM)

The branch is taken if the General Register specified by the REG field contains zero. Otherwise the instruction following the branch instruction is executed next. The ACC-field is not used.

## Branch if (REG) = (ACC) (RM)

The branch is taken if the contents of the General Registers specified by REG and ACC are equal.

## Branch if (REG) > (ACC) (RM)

The branch is taken if the contents of the General Register specified by REG are greater than the contents of the General Register specified by ACC.

## Branch if (REG) < (ACC) (RM)

The branch is taken if the contents of the General Register specified by REG are less than the contents of the General Register specified by ACC.

## Increment (ACC) and Branch if (ACC) ≤ (REG) (RM)

This instruction allows easy implementation of DO-loops. The contents of the General Register designated by ACC are incremented. The branch is taken unless the new value of (ACC) is greater than (REG).

## Branch if Overflow (RM)

The branch will be taken if the instruction immediately preceding the branch instruction was an add, subtract, or divide instruction and if an overflow had occurred.

### (5) Shift Instructions

Arithmetic Shift (RS)

The contents of the General Register specified by ACC are arithmetically shifted the number of bits indicated by SC, to the right or to the left depending on the state of the R/L bit.

Doubleword Arithmetic Shift (RS)

The doubleword contained in the General Register specified by ACC and the one immediately after it are arithmetically shifted as specified by SC and R/L.

Rotate (RS)

The contents of the General Register specified by ACC are rotated as specified by SC and R/L.

Shift Logical 0-Extended (RS)

The contents of the General Register specified by ACC are shifted as specified by SC and R/L, and the vacated bit positions are filled with zeros.

Shift Logical 1-Extended (RS)

The contents of the General Register specified by ACC are shifted as specified by SC and R/L, and the vacated bit positions are filled with ones.

### (6) I/O Instructions

Output to Vector Buffer (RR)

A track data item consisting of (TBD) words is moved from the General Registers, beginning with the one specified by REG, to the Vector Buffer for output onto the Track Bus. ACC is not used.

### Output to Signal Processor (RR)

A threshold control block consisting of (TBD) words is moved from the General Registers, beginning with the one specified by REG, to the Threshold Control Register and the output operation is initiated. ACC is not used.

### Output to Neighbor (RR)

One outgoing message block consisting of (TBD) words is moved from the General Registers, beginning with the one specified by REG, into the Outgoing Message Register and the output operation is initiated. The ACC field is not used.

### Input from Neighbor (RR)

One incoming message block consisting of (TBD) words is moved from the Incoming Message Register to the General Registers, beginning with the register specified in REG. ACC is not used.

### (7) Interrupt Handling Instructions

### Set Trap Address (RR)

This instruction allows the setting of the Trap Address Cell (TAC) for a certain interrupt level. The contents of the General Register specified by REG must be as follows:

- bit 0-2 contain a level number between 1 and 7,
- bits 4-15 contain a 12-bit address which will be written into the TAC,
- bit 3 is not used.

The ACC field is not used.

### Set Mask Register (RR)

The least significant 7 bits of the General Register specified by REG are moved into the Mask Register in the Interrupt Structure. ACC is not used.

### Resume (RR)

Execution of the Resume instruction causes the Interrupt Stack to be popped and the value of the PC and of the Level Register to be restored. REG and ACC are not used.

### (8) Miscellaneous Instructions

### Halt (RR)

This instruction halts the machine until an external interrupt starts it again.

### No-op (RR)

This instruction is the null-operation.

### Reset and Start (RR)

Resets the state of the machine to the initial state (TBD) and starts execution at the hardwired bootstrap address.

## 5.0  APSP SOFTWARE


This section contains a general discussion of tracking techniques,
followed by a discussion of two algorithms — one for tracking below the
horizon (BTH) and the other for star rejection above the horizon (ATA).

### 5.1  Tracking Techniques

There are two tracking techniques proposed for consideration and
flow diagrams are given in Figure 5.1-1.  A brief comparison of the relative
advantages and disadvantages is given in Table 5.1-1.  The first method
operates basically in a sequential (real time) manner so that one scan of data
is processed at a time.  The second method would operate on three scans of
data when considering existing tracks and would combine the detection and
estimation processes.

### 5.2  Technique 1

The elements of this technique are basically those that have classically
been associated with the multi-target track problem.  The processing is
accomplished in real time so that state variable updates are obtained at the
end of each computational frame.

#### Peak Detection

Peak detection could be used to improve the estimate of target posi-
tion at the sampling interval.  This should lead to some improvement over
the alternative of placing the target measurement at the center of the pixel
where a detection occurred.  The tradeoff between the improvement in
measurement and the required computational complexity should be
investigated.
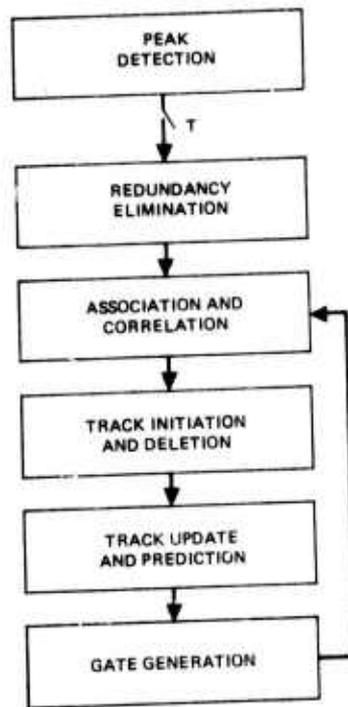
Figure 5.1-1a. Flow chart of tracking technique 1.
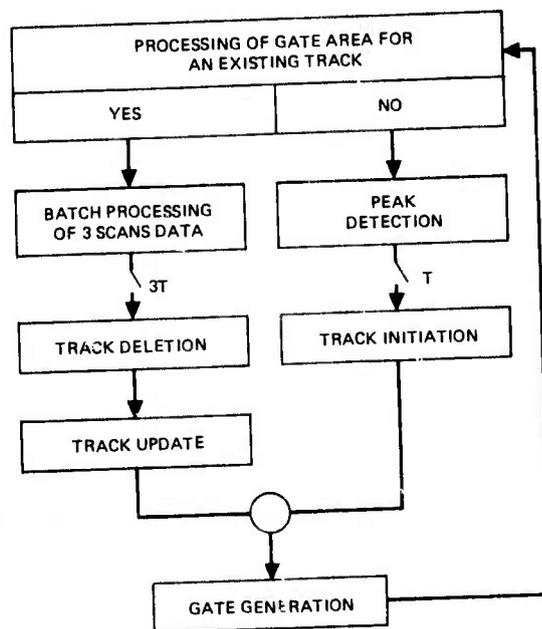


Figure 5.1-1b. Flow chart of tracking technique 2.

TABLE 5.1-1. METHOD COMPARISON

| Method | Advantage(s) | Disadvantage(s) |
|--------|-------------|-----------------|
| 1 | 1. Least computational complexity <br><br> 2. Closest to "classical" method for multi-target track | Lower bound on performance |
| 2 | Best performance | Most computational complexity |

### Redundancy Elimination

During one scan period the same target may produce detections in more than one pixel. Thus, to reduce measurement error and to reduce the probability of more than one track being initiated on the same target, redundancy elimination logic is required. This would involve some type of simple space centroiding of observations received on adjacent pixels.

### Association and Correlation

Standard association and correlation algorithms employ the nearest neighbor technique. For a single track, with gates not overlapping those of any other track, this merely involves finding the observation with the minimum normalized distance from the predicted track position. More complex conflicting situations may occur when track gates overlap and one or more observations are received in the region of overlap. For this more complex situation the use of a correlation matrix with simplified solutions to the classical assignment problem is employed.

### Track Initiation and Detection

Observations which are not associated with existing tracks are used to initiate new tentative tracks. Then, an additional criterion is typically required for a new track to become confirmed. Tracks are deleted when poor quality or no observations are received for update. The deletion criterion is more difficult to satisfy for confirmed tracks. Performance for the presently suggested initiation/deletion algorithms is given in Section 6.3.

### Track Update and Prediction

New observations are incorporated in the track and an updated state variable estimate is formed. Possible candidates for tracking filters in position are the constant coefficient $\alpha$-$\beta$ and $\alpha$-$\beta$-$\gamma$ trackers or a two- or three-state Kalman filter. Section 6.3 provides performance for the $\alpha$-$\beta$ tracker for both aircraft and missiles. The effects of crossing targets is also considered.

### Gate Generation

Gates are formed around the target's predicted state variable estimate. Only those observations found within the gate are considered (in the association and correlation algorithm) for potential track update.

### 5.3 Technique 2

The second technique employs the use of a limited form of batch processing. This would be done in the gated region of the predicted target state variable and would replace the redundancy elimination and the association and correlation functions required for Technique 1. Also, this method provides a velocity measurement so that a higher order filter will be appropriate.

### Batch Processing for Tracking

This tracking technique involves taking in "all" the MPFA data over some extended period of time and performing batch processing. The disadvantage of this technique is the inordinate amount of computational capability required and the significant delay in availability of data. For these reasons, the multiple-target track problem is commonly accomplished in real time as discussed in Technique 1. However, by doing restricted (in time) batch processing, followed by a form of the real time, multiple-target tracking algorithms, it is possible to obtain some of the advantages of both batch and real time. This is accomplished by doing batch processing over a small number of pixels (say 3) and then doing real time tracking at reduced data rates. The track predicted values are used to restrict the amount of batch processing as indicated below.

## Equations for Batch Processing

An illustrative batch track filtering algorithm can be defined by the following equations:

$$F_{m,n,i,j,t} = P_{i-m,j-n,t-1} - W_1 P_{i-m,j-n,t} - W_2 P_{i-m,j-n,t+1}$$

$$- 1/2 P_{i,j,t-1} + P_{i,j,t} - 1/2 P_{i,j,t+1}$$

$$- W_2 P_{i+m,j+n,t-1} - W_1 P_{i+m,j+n,t}$$

$$+ P_{i+m,j+n,t+1}$$

where

$$F_{m,n,i,j,t} = \text{filter output at time } - t$$

$$P_{i,j} = D_{i,j} - W_E(D_{i-1,j} + D_{i+1,j} + D_{i,j-1} + D_{i,j+1}$$

$$- W_D(D_{i-1,j-1} + D_{i-1,j+1} + D_{i+1,j-1} + D_{i+1,j+1})$$

This equation provides nine different filters ($m$, $n = 0$, $\pm 1$) needs only three frames of data storage (the Ps at three different times) and automatically performs the redundancy elimination function of Technique 1. Here

D is data from the sensor,
i and j are integer pixel indexes,
t is an integer time sample index,
$W_E$, $W_D$, $W_1$, $W_2$ are filter constants.

The filter is illustrated in Figure 5.3-1. Such a filter would be velocity sensitive and thus would potentially generate more information than a filter using time and space independently. Figure 5.3-2 shows the average speed response of such a three-dimensional filter tuned to targets
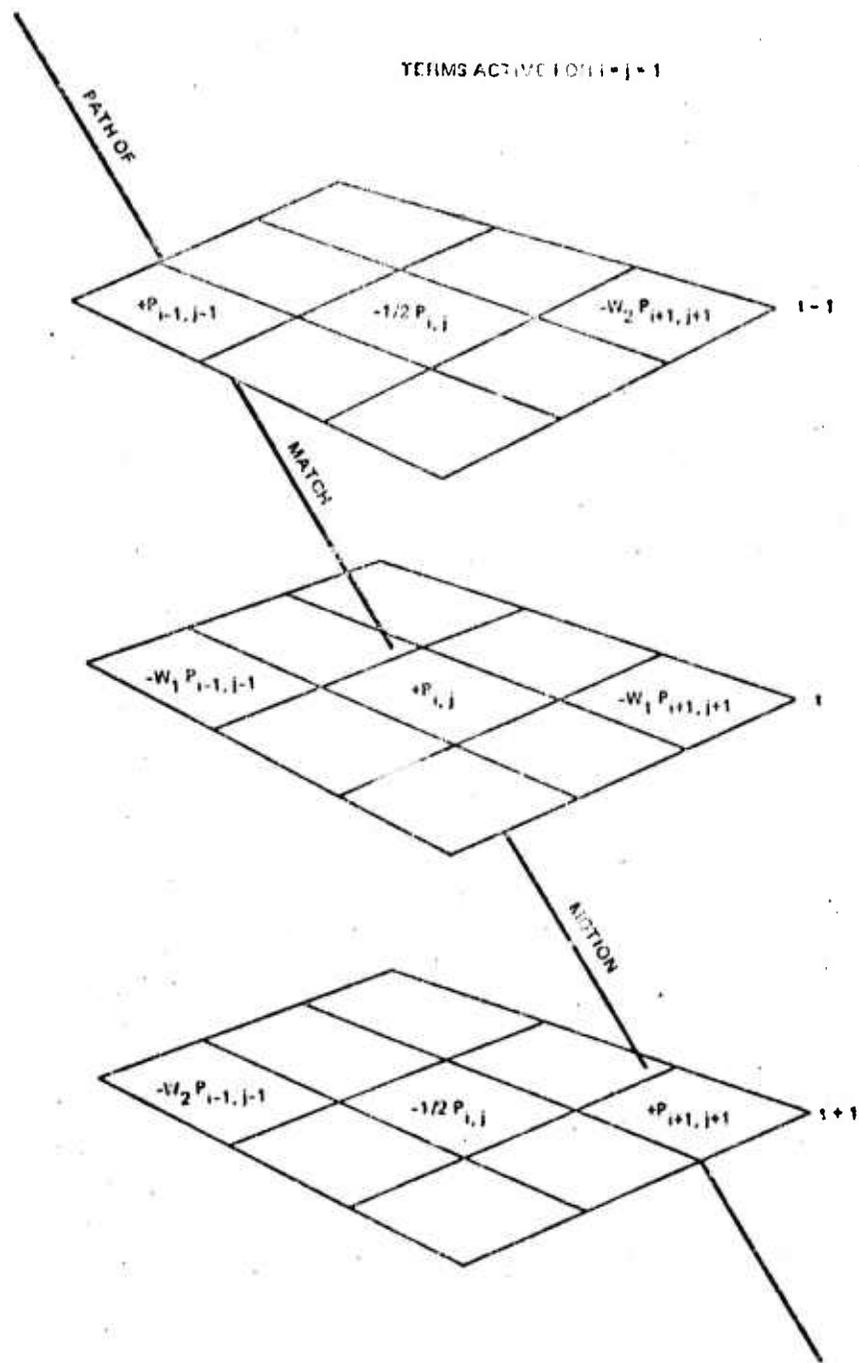
Figure 5.3-1.  Composite time-space filtering.

5-6

PRELIMINARY 3D FILTER
"BEST CASE"
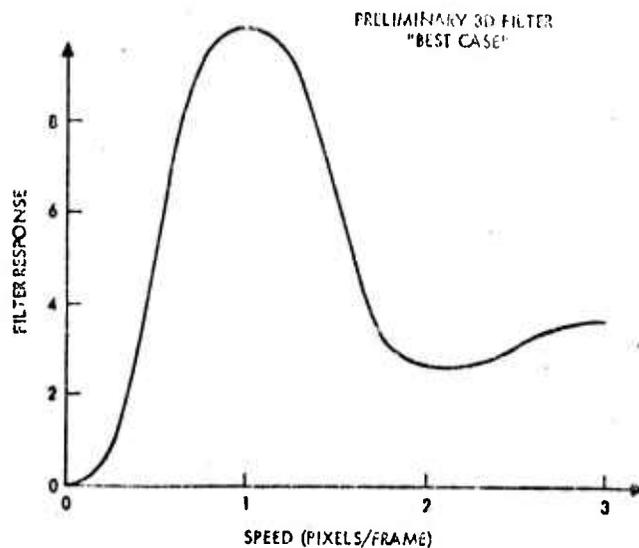
Figure 5.3-2. 3D filter response.

moving in the i-direction at a speed of 1 pixel/frame. Figure 5.3.3 shows the average directional response of the same filter, $0^{\circ}$ being the x-direction. In practice, the x-direction lies along the target's predicted path and the time dependent filter weights are based upon the extrapolated covariance estimates from the tracker. Since the error estimates increase with time, some
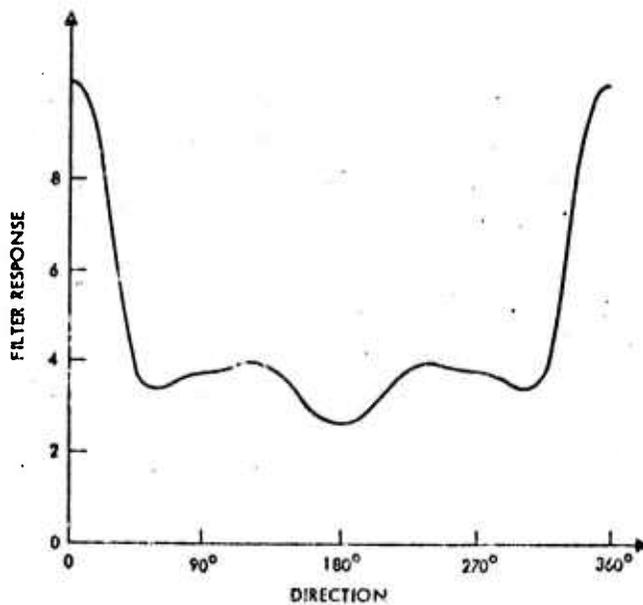


Figure 5.3-3. 3D filter response.

performance degradation occurs which degrades the natural performance improvement of batch over real time and restricts the amount of batch processing (using a single batch filter) that can be accomplished with this technique. It is noted that for effective usage of this technique either

1. Both position and velocity must be obtained from the batch filter. The tracker data rate is then reduced and a set of equations similar to that above is needed.
2. Only position information can be used by the overlapping batch processing window and not reducing the track data rate.

Track Update and Prediction

The proposed batch measurement process will provide a velocity estimate. Thus, the filter should be designed to include this additional measurement. Again standard fixed-coefficient and Kalman filtering algorithms are available. For example, one form of the fixed-coefficient filter incorporating the velocity measurement is:

$$X_s(n) = X_p(n) + \alpha_1 [X_o(n) - X_p(n)] \quad ,$$

$$\dot{X}_s(n) = \dot{X}_p(n) + \alpha_2 [\dot{X}_o(n) - \dot{X}_p(n)] \quad ,$$

$$\ddot{X}(n) = \ddot{X}(n - 1) + \frac{\beta}{T} [\dot{X}_o(n) - \dot{X}_p(n)] \quad ,$$

$$X_p(n + 1) = X_s(n) + T \dot{X}_s(n) + \frac{T^2}{2} \ddot{X}(n) \quad ,$$

$$\dot{X}_p(n + 1) = \dot{X}_s(n) + T \ddot{X}_s(n) \quad ,$$

where

subscripts o, p, and s refer to observed, predicted and smoothed quantities.

## 5.4  APSP Software

### Tracking Algorithm Development

The fundamental aspects of the APSP multi-target track problem are track initiation, gating, association and correlation of observations and tracks, updating of existing tracks, prediction of next observation, and track deletion.

Two preliminary algorithms, one for tracking and another for star rejection have been designed for these functions and will be discussed below.

It is important to minimize the number of false tracks. Thus, the proposed track initiation and gating routines provide safeguards against the initiation and maintenance of a track on observations which are inconsistent with the expected maximum target velocity, maximum acceleration, intensity and rate of change of intensity. In addition, a waveform discrimination routine has been developed. Correlation between the positive and negative peaks of the point target impulse response can be used to discriminate point targets from extended clutter. Also, a sophisticated algorithm which utilizes the vector information is used for the deletion routine.

### Tracker Software Implementation

Figure 5.4-1 shows a top level functional flow diagram of the program used for target tracking. Each of the functional blocks is discussed below.

The program commences at START every frame time ($\sim 0.1$ sec). At the START of the program, the detected hits over the previous frame are assumed to be stored in an observation file consisting of pixel coordinates ($x^*$ $y^*$), and amplitude, $A^*$, for each hit. The first task performed is the initiation function which presets various tables and counters to required values. The track index, i, is initialized to 1. The tracks are numbered from 1 to $N_{MAX}$, with i being the track index. At any one time, N tracks are valid or active and $N_{MAX}-N$ are inactive. Each valid track has associated with it a track file consisting of the state vector and other information for that track. Inactive track files are available for new tracks. For invalid tracks, the main part of the program, track update, is skipped, as shown in the flow diagram.

5-9

START

INITIALIZATION

SET: TRACK INDEX
i = 1

TRACK i VALID ? — NO

YES

OBTAIN THE SUBSET J(r) (r = 1, (M$_i$) , FROM SET OF ALL OBSERVATIONS j, THAT CORRELATE WITH TRACK i

USE $O_{ij} = \dfrac{\left[x^*(j) - x(i)\right]^2}{\sigma_x^2} + \dfrac{\left[y^*(j) - y(i)\right]^2}{\sigma_y^2} + \dfrac{\left[A^*(j) - A(i)\right]^2}{\sigma_A^2} :$ GATE (i)

DETERMINE IF TRACK i IS CONFIRMED, EXTRAPOLATING OR TO BE DELETED

DELETED

X TRAP

$\Delta X = 0$
$\Delta Y = 0$
$\Delta A = 0$

COMPUTE WEIGHTS, p(r), FOR EACH OBSERVATION, J(r)
COMPUTE WEIGHTED INNOVATION FOR EACH DIMENSION:

$$\Delta x(i) = \sum_{r=1}^{M_i} p(r) \left[X^*(J(r)) - X(i)\right]$$

$$\Delta Y(i) = \sum_{r=1}^{M_i} p(r) \left[Y^*(J(r)) - Y(i)\right]$$

$$A(i) = \sum_{r=1}^{M_i} p(r) \left[A^*(J(r)) - A(i)\right]$$

i = i + 1

UNREASONABLE

COMPUTE TRACK i DELTAS AND CHECK FOR REASONABLENESS

DELETE TRACK i

UPDATE TRACK i

SELECT $\alpha, \beta$ FOR NEXT FRAME

TRACK i ACROSS BOUNDARY ? — NO — YES

TRACK i CONFIRMED ? — NO — YES

OUTPUT TRACK i TO SPACECRAFT

BOUNDARY ALGORITHM

OUTPUT TO 1 OF 8 NEIGHBORS

$i \geq N_{MAX}$ ? — NO — YES

CREATE NEW TRACKS FROM UNCORRELATED OBSERVATIONS

CREATE NEW TRACKS FROM CROSSOVERS

ADJUST THRESHOLDS

SELF-TEST (AS TIME PERMITS)

$\Delta T$ INTERRUPT

START

Figure 5.4-1.   Target tracking program.

5-10

For valid tracks, all of the observations are scanned to see which, if any, are within the gate of the track presently under consideration. A generalized distance function can be used, as shown, and the size and shape of the gate can be different for each track. For each track a table, J, containing the index values, j, of observations that correlate with that track, is compiled. Typically zero, one, or two observations will correlate with a given track.

Next, the confirmation, extrapolation or deletion algorithm is executed. A new or tentative track will be confirmed if p hits are received on the first k frames. Otherwise the track is deleted. A track will be extrapolated if no hits are received on this frame. For a track that has reached confirmation, deletion will occur in the event of m consecutive frames of extrapolation. For extrapolating tracks the observational innovations $\Delta X$, $\Delta Y$, and $\Delta A$ are left unchanged.

For each observation, J(r), that correlates with track i, a weighting function, p(r), is computed denoting the quality of that observation. Next the p(r)'s are used to compute innovations for each dimension by summing the products of the p(r)'s and the corresponding observational residuals as shown.

In the next block, the total change in the track i state vector is computed using the innovations weighted by the $\alpha$, $\beta$ gain factors. Reasonableness checks can be performed on the deltas and large enough values can result in track deletion. For reasonable deltas, the track i state vector is updated in the next block. Then the $\alpha$, $\beta$ gain factors are selected for use in the next frame. The maximum p(r) value is first found, and one of three sets of $\alpha$, $\beta$ values is selected based on the value of $p(r)_{MAX}$.

Using the updated state vector, the track i is next checked for having crossed the boundary of the tracker's domain. For confirmed tracks within the boundary, the track file for track i is transferred to the Vector Buffer to be output on the Track Bus. The program then proceeds to the next iteration as shown.

For tracks crossing the boundary, a special algorithm must be executed to determine which of the eight neighboring trackers should receive the track and to what state vector the new tracker should be initialized. One

5-11

technique for accomplishing this is diagramed in Figure 5.4-2 and employs the following strategy:

1. Extend the (x, y) coordinates of the tracker beyond the boundaries of its domain and continue to keep a track file on tracks that cross the boundary using extrapolation without new observations.

2. Continue to extrapolate the track until its predicted position can place it uniquely in one of the eight adjacent trackers. For example, referring to the figure, the track should be assigned to neighbor number 8 if

$$(x > 64 + \delta) \text{ and } (-64 \leq y \leq +64)$$

where

$$\delta = \text{Gap width (in pixel units)} + 1 \text{ pixel}$$

3. Once the neighbor receiving the track has been determined, coordinate conversion into the new tracker's (x, y) frame can be performed.

Figure 5.4-2 shows the domain of a tracker and portions of the domain of the eight nearest neighbors. In the first part of the flow chart the determination of the boundary condition is shown. For tracks that have crossed the boundary, a search for conditions that put the track in one of the neighbors is made. If no such conditions are found, the track is extrapolated. If a unique neighbor is found, the state vector of track i is transferred to that neighbor after coordinate conversion. Track i is then deleted from the tracker from which it was transferred.

After iterating through all i values and updating all valid tracks, the program proceeds to install the new tracks, if any. New tracks first are created from all those observations that did not correlate with any existing tracks. Next, all tracks that were handed over from neighbors are installed as new tracks. All new tracks are tentative and must pass the confirmations criterion before becoming full-fledged tracks. Gates for tentative tracks, particularly crossover tracks, can be larger than in the steady state.

In the following block, the threshold value for the tracker is adjusted. Possible criteria could be total number of observations, the number of active tracks and/or the rate of change of the number of active tracks.
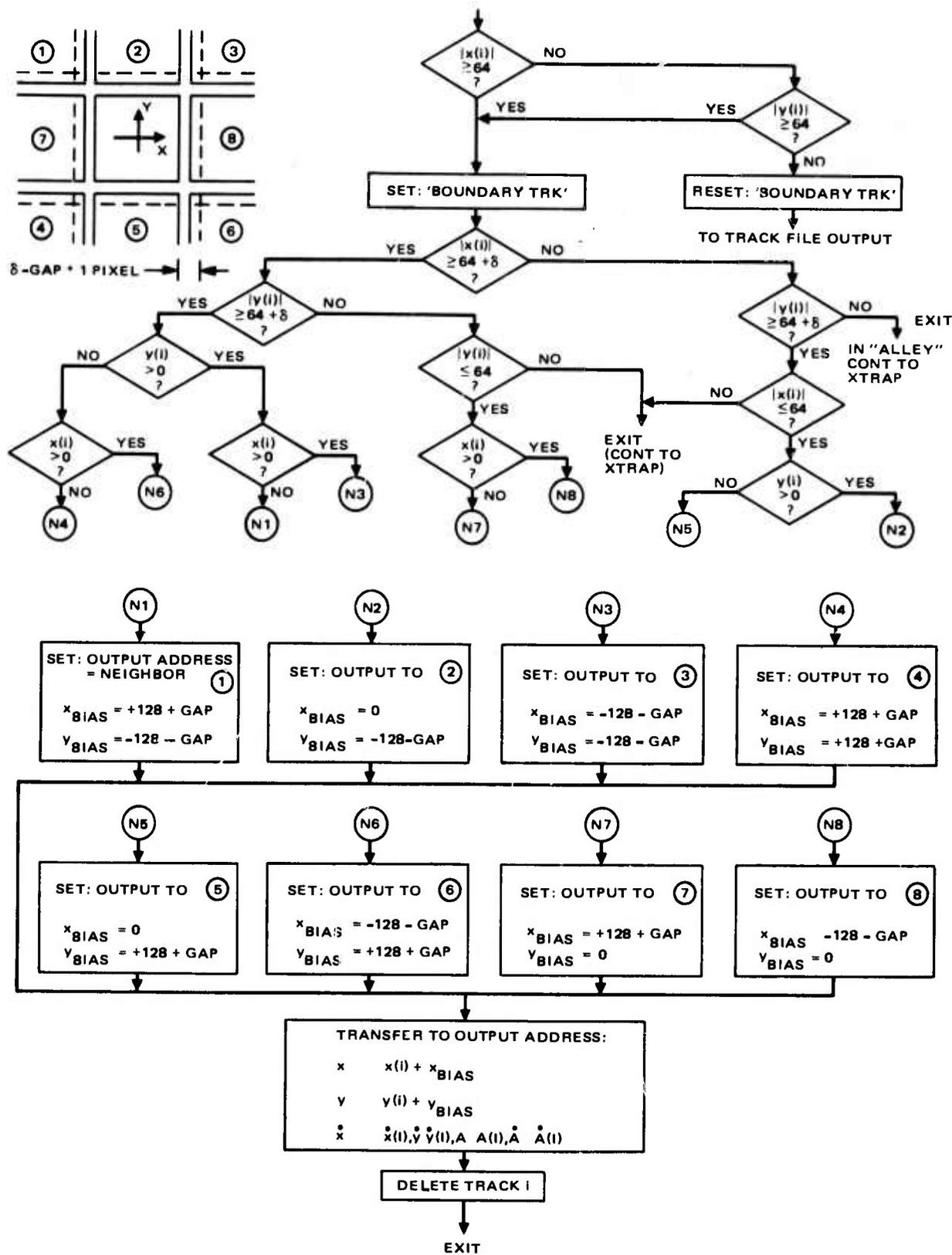
Figure 5.4-2. Boundary algorithm.

5-13

Finally, self test functions can be performed in the time remaining in the frame. The monitoring process will be interrupted by the frame clock which will direct the program back to START at the beginning of the next frame.

A significant part of the tracking software has been refined to the point that tentative instruction counts, storage requirements and execution times can be obtained. Figure 5.4-3 thru 5.4-7 show the flowcharts for the respective parts of the BTH tracking software. Table 5.4-1 gives a detailed account of storage requirements. Table 5.4-2 shows the number of instructions and the number of operations (instruction executions) necessary to implement the flow charts shown in the figures. Also in Table 5.4-2, the execution time for this part of the program is computed under certain plausible assumptions and is found to be under 0.02 seconds. Considering that the typical frame time is 0.1 second and that the 0.02 second execution time represent the processing time for a very significant part of the total tracking software (certainly more than 20%, and probably more than 50%), it can be concluded that one $\mu$PT will be able to handle tracking for one MFPA chip easily. It may even be possible to let one $\mu$PT handle several MFPA chips.

## Star Discrimination Algorithms

The second algorithm used in the microprocessor-tracker eliminates the star background. Utilization of a staring sensor for tracking space objects against the moving background star field (Figure 5.4-8) points up three problem areas for the signal and track processors:

1. The number of detectable stars is a function of sensor sensitivity and can be considered (see Table 5.4-3)

2. With the satellite in synchronous orbit, the star field appears to be moving at a rate 2.9 pixels/sec. This motion must be compensated for in order to cancel star detections.

3. The signal-to-noise ratio (S/N) of a typical satellite target can be less than unity (e.g., for RVs, S/N may be less than 0.5). This complicates the track initiation problem unless some time-delay and integration (TDI) is applied.

5-14

BLOCK 1 — Compute Dij Table (Generalized
Distance from Track i to
Observation j)



For $(i = 1, N)$; $(j = 1, M)$

Do:

$$D_x = X*(j) - X(i)$$

$$D_x = Y*(j) - Y(i)$$

$$D_z = A*(j) - A(i)$$

If $(D_x \geq C_y)$ or $(D_y \geq C_y)$ or $(D_a \geq C_y)$

Then $D(i,j) = +\infty$ (a very large value)

Else $D(e,j) = \dfrac{D_x^2}{\sigma_x^2} + \dfrac{D_y^2}{\sigma_x^2} + \dfrac{D_a^2}{\sigma_a^2}$

SET: $i = 1$
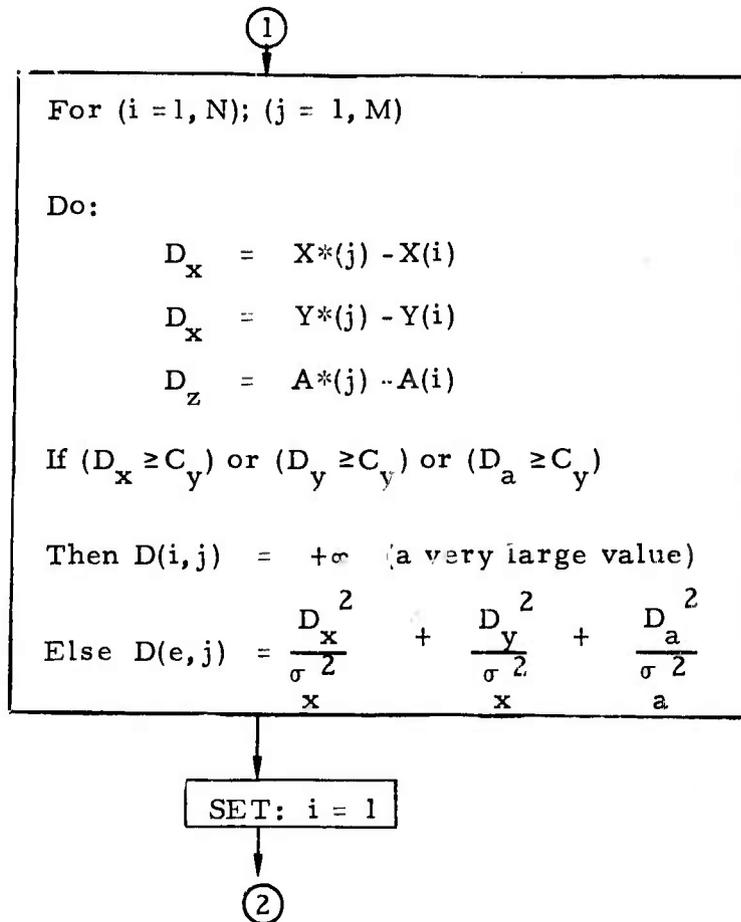
Figure 5.4-3.   BTH block 1.

BLOCK 2 - CREATE GATE FILE AND P'' (r)'s
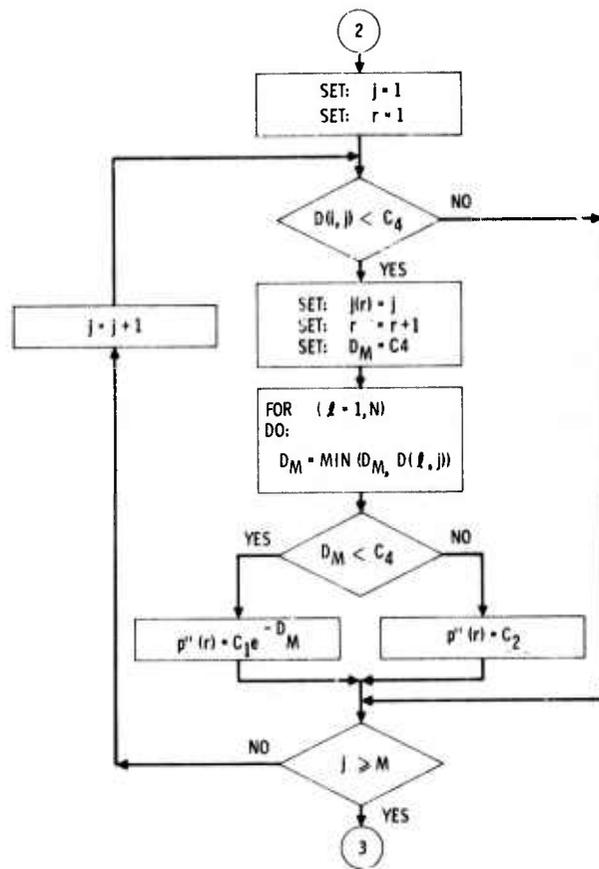


Figure 5.4-4.   BTH block 2.

5-16

BLOCK 3 - COMPUTE GATE FILE p' (r), p(r)
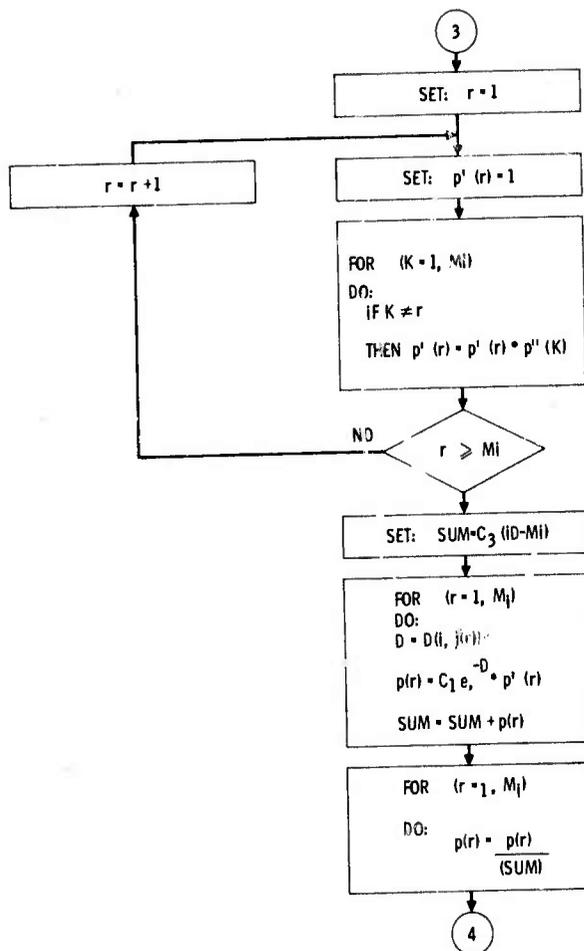


Figure 5.4-5.   BTH block 3.

BLOCK 4 - TRACK UPDATE

1. COMPUTE WEIGHTED INNOVATIONS

2. TRACK UPDATE USING $\alpha$-$\beta$ TRACKER



1)

SET:     $\triangle X(i) = 0$, $\triangle Y(i) = 0$, $\triangle A(i) = 0$

FOR     $(r = 1, MI)$

DO:     $\triangle X(i) = \triangle X(i) + p(r) \cdot \left[ X^\bullet (j(r)) - X(i) \right]$

        $\triangle Y(i) = \triangle Y(i) + p(r) \cdot \left[ Y^\bullet (j(r)) - Y(i) \right]$

        $\triangle A(i) = \triangle A(i) + p(r) \cdot \left[ A^\bullet (j(r)) - A(i) \right]$

2)

COMPUTE:

$\dot{X}(i) = \dot{X}(i) + \dfrac{\beta x(i)}{T} \cdot \triangle X(i)$

$X(i) = X(i) + \alpha x(i) \cdot \triangle X(i) + T \cdot \dot{X}(i)$

$\dot{Y}(i) = \dot{Y}(i) + \dfrac{\beta x(i)}{T} \cdot \triangle Y(i)$

$Y(i) = Y(i) + \alpha x(i) \cdot \triangle Y(i) + T \cdot \dot{Y}(i)$

$\dot{A}(i) = \dot{A}(i) + \dfrac{\beta A(i)}{T} \cdot \triangle A(i)$

$A(i) = A(i) + \alpha A(i) \cdot \triangle A(i) + T \cdot \dot{A}(i)$
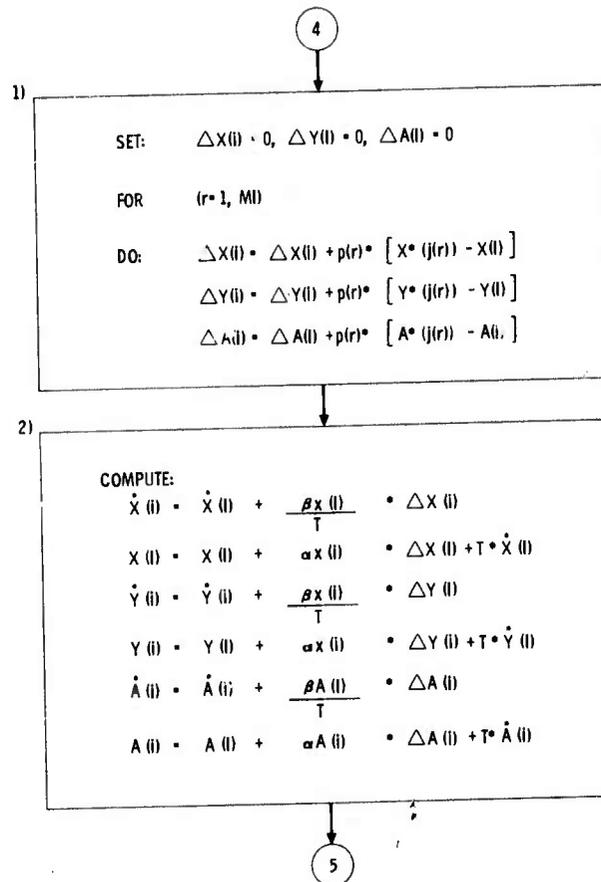
Figure 5.4-6.   BTH block 4.

BLOCK 5 - DETERMINE $\alpha, \beta$ FOR NEXT SCAN

1) DETERMINE MAX p(r)

2) SELECT $\alpha, \beta$ BASED ON MAX p(r)
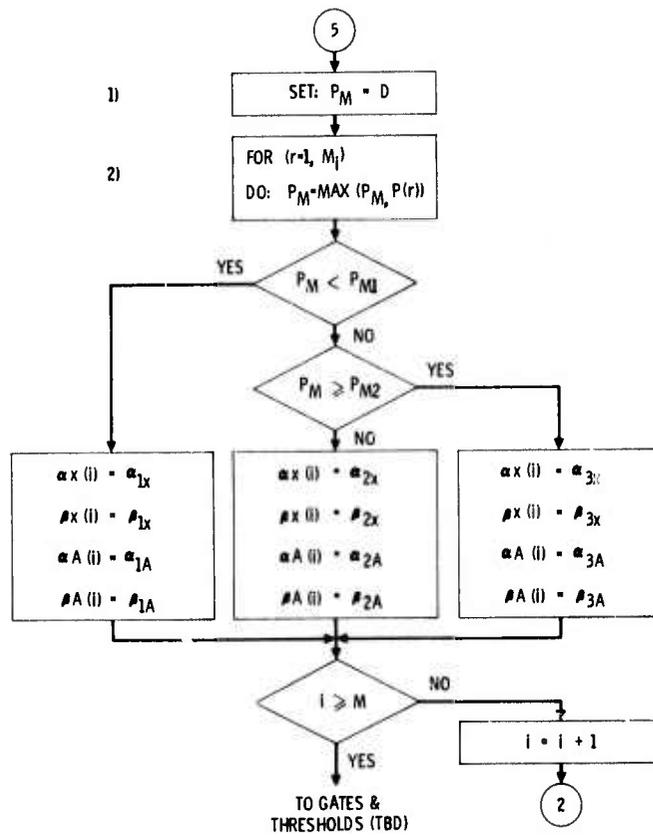


Figure 5.4-7.   BTH block 5.

## TABLE 5.4-1. DATA BASE

TRACK FILE (6 X N)

| | | | | | |
|---|---|---|---|---|---|
| $X(1)$ | $\overset{\circ}{X}(1)$ | $Y(1)$ | $\overset{\circ}{Y}(1)$ | $A(1)$ | $\overset{\circ}{A}(1)$ |
| $X(2)$ | $\overset{\circ}{X}(2)$ | $Y(2)$ | $\overset{\circ}{Y}(2)$ | $A(2)$ | $\overset{\circ}{A}(2)$ |
| $\vdots$ | $\downarrow$ | $\vdots$ | $\downarrow$ | $\vdots$ | $\downarrow$ |
| $X(i)$ | $\overset{\circ}{X}(i)$ | $Y(i)$ | $\overset{\circ}{Y}(i)$ | $A(i)$ | $\overset{\circ}{A}(i)$ |
| $\vdots$ | $\downarrow$ | $\vdots$ | $\downarrow$ | $\vdots$ | $\downarrow$ |
| $X(N)$ | $\overset{\circ}{X}(N)$ | $Y(N)$ | $\overset{\circ}{Y}(N)$ | $A(N)$ | $\acute{A}(N)$ |

OBSERVATION FILE (3 X M)

| | | |
|---|---|---|
| $X*(1)$ | $Y*(1)$ | $A*(1)$ |
| $X*(2)$ | $Y*(2)$ | $A*(2)$ |
| $\vdots$ | $\vdots$ | $\vdots$ |
| $X*(j)$ | $Y*(j)$ | $A*(j)$ |
| $\vdots$ | $\vdots$ | $\vdots$ |
| $X*(M)$ | $Y*(M)$ | $A*(M)$ |

GAINS (4X N)

| | | | |
|---|---|---|---|
| $\alpha_x(1)$ | $\beta_x(1)$ | $\alpha_A(1)$ | $\beta_A(1)$ |
| $\alpha_x(2)$ | $\beta_x(2)$ | $\alpha_A(2)$ | $\beta_A(2)$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $\alpha_x(N)$ | $\beta_x(N)$ | $\alpha_A(N)$ | $\beta_A(N)$ |

## TABLE 5.4-1. DATA BASE (CONTINUED)

D (Generalized Distance) Table (N X M)

| D(1, 1) | D(2, 1) | . . . . . . . . . . . | D(N, 1) |
|---------|---------|-----------------------|---------|
| D(1, 2) | D(2, 2) | . . . . . . . . . . . | D(N, 2) |
| : | . | | : |
| D(1, M) | D(2, M) | . . . . . . . . . . | D(N, M) |

GATE FILE (4 X Mi)

| $j(1)$ | $p''(1)$ | $p'(1)$ | $p(1)$ |
|--------|----------|---------|--------|
| $j(2)$ | $p''(2)$ | $p'(2)$ | $p(2)$ |
| : | : | : | : |
| $j(r)$ | $p''(r)$ | $p'(r)$ | $p(r)$ |
| : | : | : | : |
| $j(Mi)$ | $p''(Mi)$ | $p'(Mi)$ | $p(Mi)$ |

TABLE 5.4-2.  TRACKING STORAGE REQUIREMENTS

| Block | | Operations | Memory Locations |
|---|---|---|---|
| 1. Compute Dij Table | : | $25 * (M * N) + 5 \times N$ | 30 |
| 2. Create GATE FILE $j(r)$'s and $p''(r)$'s | : | $[35 (Mi)_{AVG} + 10Mi] * N$ | 75 |
| 3. Compute GATE FILE $p'(r)$ and $p(r)$ | : | $[5 (Mi)^2_{AVG} + 18(Mi)_{AVG} + 5] * N$ | 55 |
| 4. Track Update | : | $[8* (Mi)_{AVG} + 14] * 3 N$ | 50 |
| 5. New Gains $(\alpha, \beta)$ | : | $[10*(Mi)_{AVG} + 15] * N$ | 45 |
| TOTALS $\longrightarrow$ | | $N * [5 (Mi)^2_{AVG} + 87 (Mi)_{AVG} + 68 + 35 M]$ | 255 |

For M = 200, N = 20, Mi = 1:  $20* [5 * 1^2 + 87 * 1 + 68 + 35 * 200] = 143,200$ ops

At 8 MIPS throughput:  $\dfrac{1.43 \times 10^5 \text{ ops}}{8 \times 10^6 \text{ ops/sec}} = 0.018$ sec

(This table does not include the boundary algorithm.)
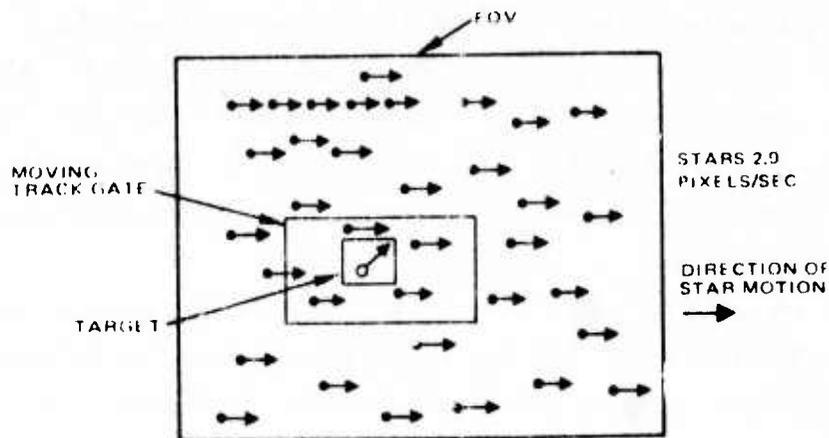
Figure 5.4-8. Star background discrimination.

TABLE 5.4-3. NOMINAL STAR DENSITY

| | Star Density (stars/deg$^2$) | | | |
|---|---|---|---|---|
| Threshold | Out of the Galactic Plane | In the Galactic Plane | Stars per MFPA (16,384 pixels) | Star Density in pixels |
| 30 watts/sr | 3 | 0 | 6 | 1 in 2731 |
| 5 watts/sr | 7 | 2500 (?) | 84 | 1 in 195 |

Assuming that the 8 MIPS* dedicated microprocessor tracker described in Section 4 of this report is available, the anticipated star densities seen by an MFPA chip can be handled. In fact, 110 updated tracks (the worst case of $\mu + 3\sigma$) per frame would allow 7200 instructions per update at a frame rate of 10 Hz. Other considerations, such as anticipated mean star spacing and desired pixel integration times, make a frame rate between 1 and 10 Hz desirable. Current estimates place the number of operations required to process one update between 100 and 800. Hence 8 to 64 MFPA chips may time share a single μPT. Track parameters include, as a minimum, two spatial dimensions, intensity and a track status flag.

---

*(4000 instructions/track x 200 targets/frame x 10 F/S = 8 MIPS)

5-23

Figure 5.4-9 shows a block diagram of this star discrimination preprocessing function. Note that star tracks are not deleted unless several frames (e.g., 5) pass without hits in the predicted positions. Any hit which cannot be correlated with a star is treated both as a new track and as a potential target. Only such hits are passed on to the system tracking algorithms. These algorithms are designed to delete tracks moving with the star velocity vector.

In the exceptional case that targets (ASATs, RVs, etc.) are moving with the star field (in speed and direction) at all times, they cannot reach the sensor, or any other target, and therefore need, not be tracked. If this rule is insufficient for discrimination purposes, changing target amplitude of the target could be used to augment the basic velocity discrimination algorithm.

The standard method for track initiation is to attempt to correlate consecutive HITS. This is accomplished by placing a gate around a new detection and checking the next frame for a HIT within the gate. For low S/N applications, it may be necessary to use the temporal discrimination filter discussed in 5.3. If a target were moving at an unknown velocity in any random direction, TDI requires some a priori tracking information. Thus, the designer faces a dilemma: TDI cannot be used until a track has been initiated, and track initiation itself relies on some sort of TDI due to low S/N. Several approaches should be considered during system design in order to resolve this dilemma.

ATH Software Implementation

The ATH tracking algorithm operates in two modes:

1. Target acquisition mode, in which all threshold excessions are considered and stars are eliminated, and

2. Track maintenance mode, in which tracks designated as targets only are tracked.

The track maintenance procedure is executed every frame time. The target acquisition procedure is executed periodically. Currently, it is contemplated to execute target acquisition once every 3 seconds. Figures 5.4-10 and 5.4-11 show top level functional flow diagrams of the programs for target acquisition and for track maintenance, respectively.
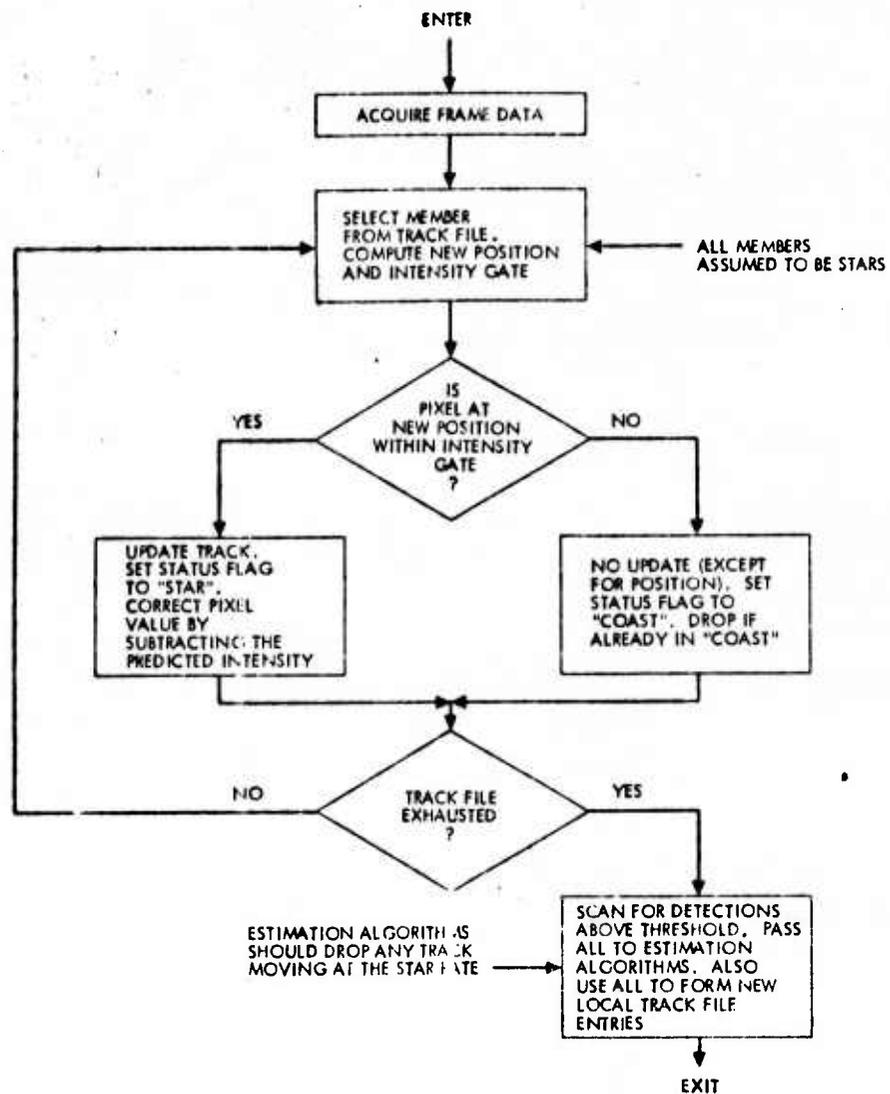
Figure 5.4-9. Star preprocessing.
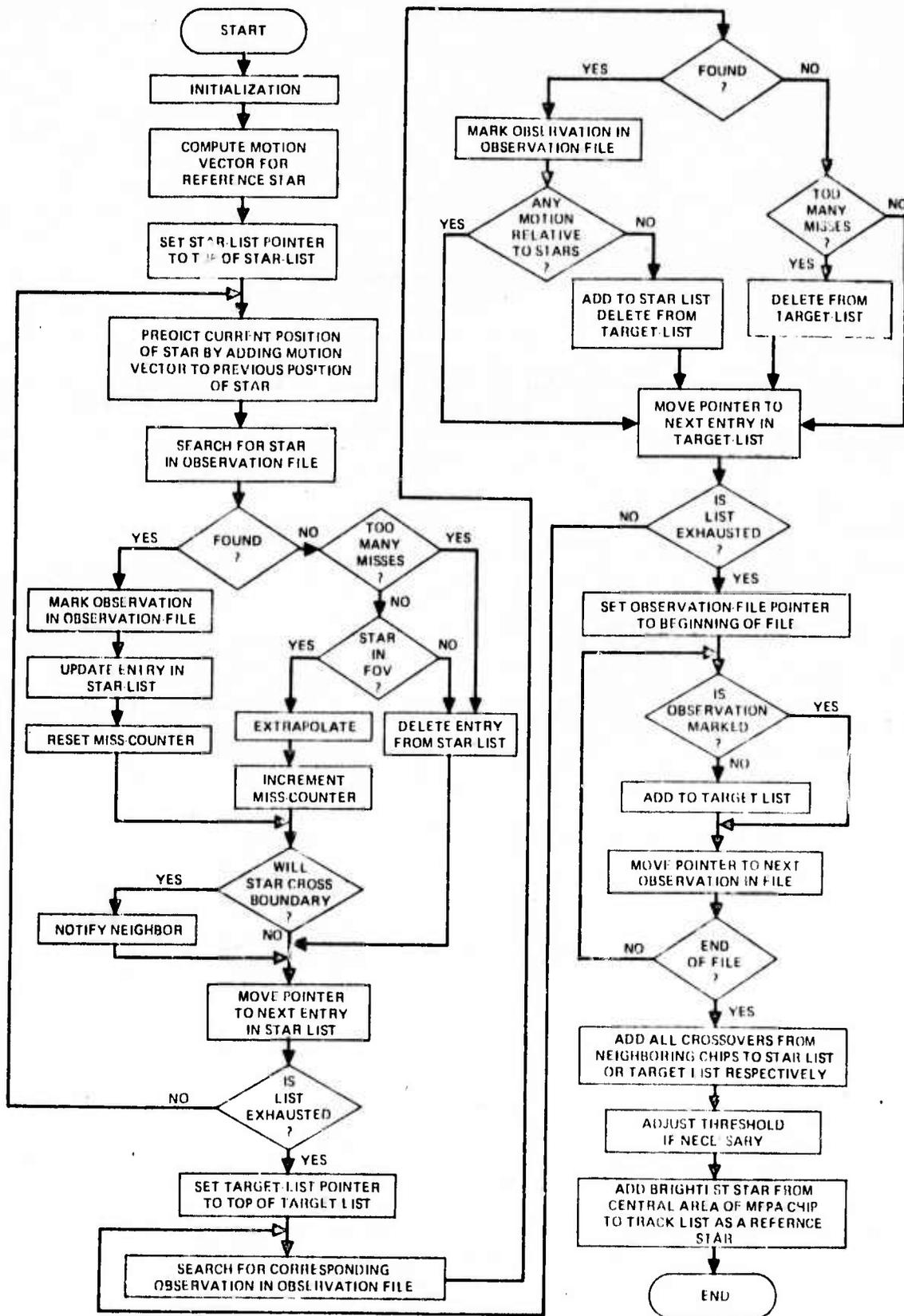
Figure 5.4-10. Target acquisition flow diagram for ATH mode
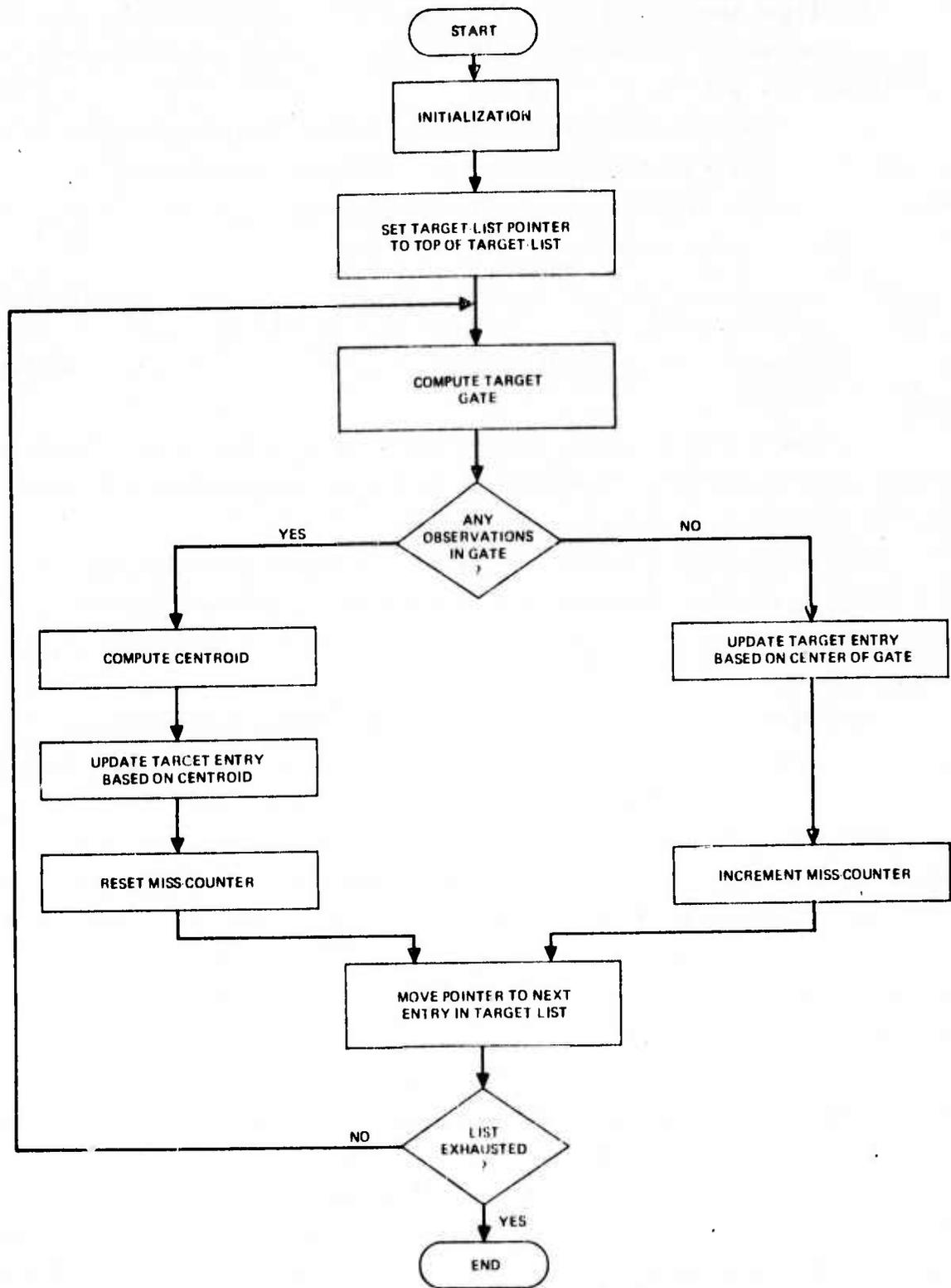
Figure 5.4-11.   Target maintenance flow diagram for ATH mode.

The target acquisition procedure starts out by computing the "motion vector". This vector represents the shift of the frame of reference since target acquisition was last executed.

Next the list of stars compiled during the last target acquisition is scanned. Each star's position is modified by the motion vector and the current observation file is searched for a corresponding entry. If no such entry is found, two conditions have to be checked for:

- the star may not have been observed in many consecutive frames in in which case it is deleted from the star list, or

- The star may have left the field of view, in which case it is also deleted.

If neither condition exists (but the star was not found in the observation file) the star-list entry is updated by extrapolation and the miss-counter (a field of the star-list entry) is incremented.

Usually however, the star will be found in the observation file. In that case the respective observation is marked as having been processed. The entry in the star list is updated based on the observation and the miss-counter is reset.

Last, it has to be checked whether the star being processed will cross into a neighboring chip before the subsequent target acquisition period. If that is the case, the respective neighboring track processor is notified.

The above procedure is repeated for every star in the star list. Next the current list of targets is scanned and the observation file is searched for entries corresponding to each target. If a corresponding observation is found, it is marked as having been processed. If the target's motion was identical to the reference star's motion, then the entry is deleted from the target list and added to the star list.

If no observation corresponding to a given target is found, then the miss-counter of that target (a field in each target list entry) has to be checked. Targets that have not been observed in many consecutive frames are deleted.

The above procedure is repeated for every entry in the target list. Next the observation file is scanned and every unmarked (i.e., unprocessed) observation is added to the target list. Thereafter all crossover tracks from neighboring track processors are added to the star list or target list, as the case may be.

The threshold may be adjusted to maintain the processing load of the track processor constant. (The maximum capability being $10^7$ threshold excessions per second.)

Last, the brightest star from the central area of the detector/mux chip is added to the target list to serve as reference star until the next target acquisition period.

The track maintenance procedure is executed every frame time. No additions or deletions to the target list are made; only continuation points of existing tracks are determined here.

A gate is computed for each track. Typically exactly one observation will fall within the gate. If no observation is found within the gate then the respective target-list entry is updated as if an observation had been found in the center of the gate, and the miss-counter is incremented. If one or more observations fall within the gate, their centroid is computed and the respective target-list entry is updated based on this centroid. The miss-counter is reset.

This procedure is repeated for each entry in the target list.

The execution time for the target acquisition procedure and for the track maintenance procedure are estimated to be less than 0.003 and 0.018 seconds respectively. This indicates that up to 32 detector/mux chips can be handled by one track processor.

In summary, star discrimination at a level 164 stars per detector/mux chip can be handled uniquely with the dedicated on-board trackers by using a simple prediction algorithm which eliminates all targets that move with the star field and do not change in amplitude overtime.

## Omni-directional TDI (OTDI)

Here a nominal target speed is assumed and integrations proceed in "all" directions (say eight). After a definite integration time all results are checked and the largest value selected (see Figure 5.4-12).

This technique consumes a great deal of computing power in the trackers and generally represents "overkill." A hand-over of RV track information from the BTH sensors to the ATH sensors includes an estimated velocity of 6 pixels/sec and a direction of about ±10 degree uncertainty. As

5-29
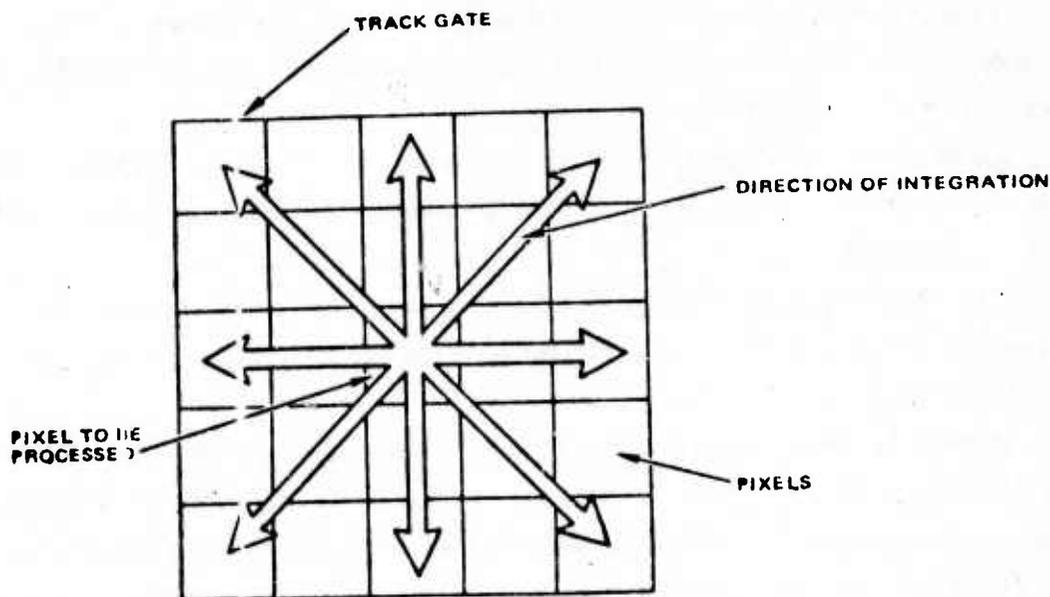
Figure 5.4-12. Omni-directional time delay and integration.

illustrated in Figure 5.4-13, OTDI then needs to be performed only over a limited 20 degree sector and velocity range of ±1 pixels/sec.

Two-dimensional Transform Techniques[*]

A patch around the new potential detection is integrated in time. It is subsequently spatially transformed (WHT or FFT) and the two-dimensional spatial frequency diagram (k-space) examined. In the integrated input patch, tracks should appear as weak line segments which are detectable in the transform domain as resonances at certain points in k-space.

It is recognized that both of these approaches require further study both with respect to their scopes of applicability and their data processing requirements. A tradeoff must be made between the FFT and WHT transform techniques and others.

In summary, star discrimination at a level of 2500 stars/deg$^2$ (84 stars per MFPA chip) can be handled uniquely with the dedicated on-board trackers ($\mu$PT) by using a simple prediction algorithm which eliminates all

---

[*]W.K. Pratt, Image Processing Institute, Summer Course, University of Southern California, Los Angeles, California, 1975.
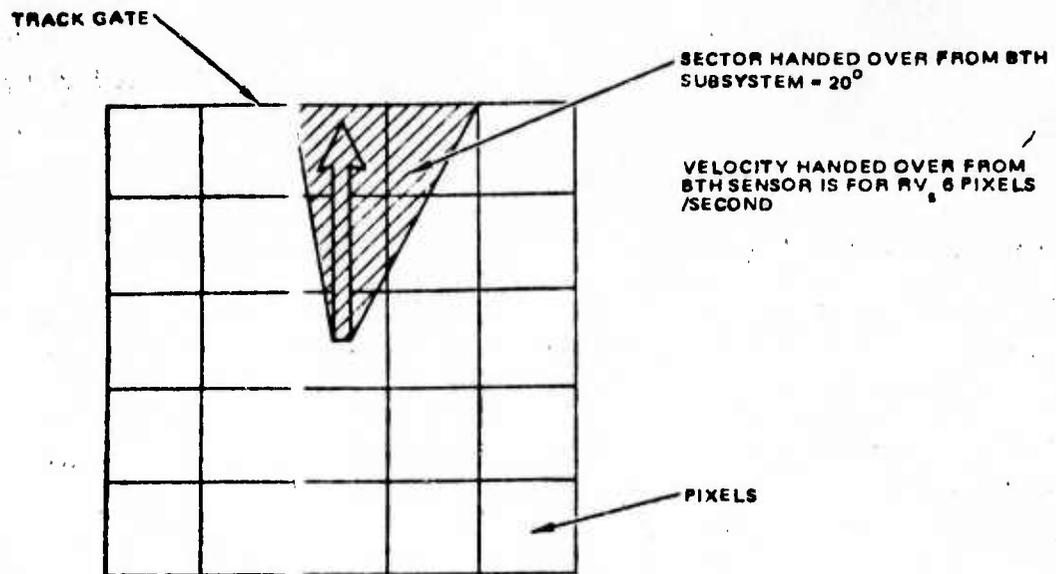
Figure 5.4-13. Selective direction time delay and integration.

targets that move with the star field and do not change in amplitude over time. In addition, for certain targets with low signal-to-noise ratios, an OTDI technique and a transform technique have been introduced as solutions to enhance the detection probability of weak targets (RVs). These algorithms will be implemented in the trackers. Finally, the trackers can be instructed to track designated targets which are handed over from the BTH subsystem. This a priori information to the trackers relaxes the computation load on the $\mu$PT.

# 6.0 ANALYSIS

This section contains an analysis of the adaptive video encoder performance, the temporal detection filter signal-to-noise and signal-to-clutter performance, the Walsh Hadamard processor and the tracker performance. It is shown that a relatively simple $\alpha$-$\beta$ tracker is effective in tracking maneuvering targets with a fraction of a pixel error and in deleting false tracks after a few frame times.

## 6.1 Adaptive Signal Encoder Performance Analysis

This subsection consists of the following items:

(1) Simulation Results

(2) Encoder Noise Analysis.

### Simulation Description

The simulation approach taken in this analysis utilized a two-phase investigation. In the first phase the predictive feedback encoder of Figure 4.1-2 was modeled assuming no D/A or summing (E=S-P) errors. Additionally, a single A/D with 5-bit precision and infinite word length was modeled, instead of a two channel A/D. This yielded an idea of the maximum magnitude of the difference signal E and its dependence upon the input waveform and sample rate. Knowing this, it was possible to determine under what conditions the magnitude of E would exceed a 5-bit word size, i.e. saturation. Eecause saturating values of E result in large encoding errors, the dual channel A/D shown in the block diagram of Figure 4.1-2 was chosen to reduce such errors for large difference signals while retaining the 5-bit accuracy for small difference signals.

Except for the fact that quantization error was included in the model, the first phase might be considered to be the "ideal" case. The second phase modeled the encoder as shown in Figure 4.1-2, considerating saturation effects and A/D quantization error but again neglecting D/A and summing noise. The input-output error and transient characteristics were then determined for specified input signals.

Simulation Results

Two simulations were performed: the first modeled the encoder neglecting A/D saturation and overflow effects in order to look at the difference signal magnitude E and see under what conditions it would saturate a conventional 5-bit A/D converter. It was found that the maximum E generated was a function of both the input signal and the number of samples/dwell time denoted by $\tau_d/T$. For a sampled, convolved-Gaussian, unity maximum amplitude signal input ($\sigma = 0.1283d$)[*] the A/D would saturate at values of $\tau_d/T$ below 150 for n = 1 and below 35 for n = 2. Thus the channel 'b' A/D converter is sufficient to encode the error signals for higher $\tau_d/T$ values with 5-bit accuracy and resolution, while for all lower values of $\tau_d/T$, the channel 'a' A/D must be used with the resultant no greater than 5 LSB error.

The second simulation determined the following characteristics:

1. RMS input-output error-vs-$\tau_d/T_f$ and error in encoding signal peak-vs-$\tau_d/T_f$

2. Transient Response

The rms and peak errors are defined as follows:

$$\epsilon_{rms} = \left[ \frac{1}{K} \sum_{k=1}^{K} \left( S_k - \Gamma C_k \right)^2 \right]^{1/2}$$

---

[*]In this expression d is the detector width and $\sigma$ is a measure of the blur width based on the 60% point of the Gaussian waveform.

where

$$S_k = \text{MFPA signal at time } k$$

$$PC_k = \text{encoder output signal at time } k$$

$$K = \text{number of samples over signal duration}$$

$$= 2\tau_d/T_f + 1 \text{ (see figure above)}$$

and

$$\epsilon_{peak} = \left| S_k - PC_k \right| k = \frac{\tau_d}{T_f} + 1 \quad .$$

Figure 6.1-1 shows $\epsilon_{rms}$ as a function of $\tau_d/T_f$. The non-linear nature of the curves is a result of the non-linearities introduced by the two-channel A/D converter and the prediction clamping (clamps prediction to be between 0 and 1).

At one sample/dwell the difference signal is converted by the coarse A/D because of its large magnitude; this results in large errors due to the type of conversion.

At two samples/dwell the difference signal is in general large but its conversion does not yield the large quantization error as before. The input-output error is mainly a function of the A/D conversion error which is a non-linear function of the difference signal amplitude. Conversion error is determined by the magnitude of the difference signal E: If E is larger than 5 bits, it is encoded by the coarase A/D with a 5-LSB error; an E 5 bits or less is encoded by the fine A/D with 5-bit precision. The error characteristics of the encoder thus depend on the frequency of occurrence of large difference signals E, which depends in turn upon the ability of the predictor to track the input waveform. Thus the behavior of the encoder at $\tau_d/T_f = 2$ is a result of the fact that the difference signal magnitude was such that smaller conversion errors were introduced.

At large values of $\tau_d/T_f$ the curve for n = 2 levels off at the quantization noise of the low-channel A/D converter. This is due to the decreasing magnitude of the difference signals for large $\tau_d/T_f$; such smaller signals can be encoded by the A/D with less quantization error.
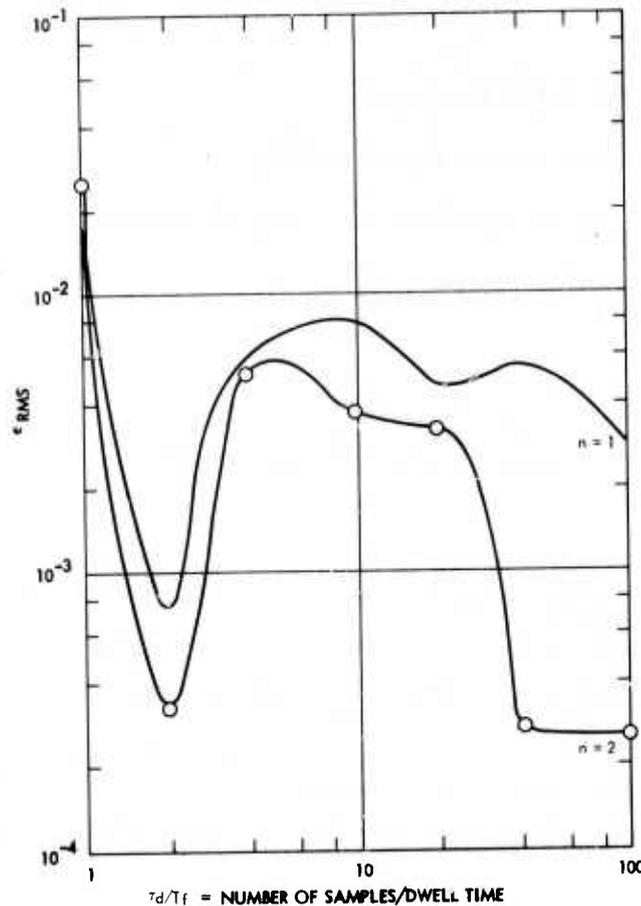
Figure 6.1-1. Rms error.

Except for one sample/dwell, the input-output error remains well below 1 percent.

Figure 6.1-2 is a plot of $\epsilon_{peak}$ as a function of $\tau_d/T_f$. The large error at one sample/dwell time is due to the error in converting the large difference signal with the high-channel A/D. As more samples are taken, the predictions become better in the sense that the smaller difference signals can be encoded by the low-channel A/D with resulting less error. At values of $\tau_d/T$ above 10, the quantization noise of this low-channel A/D dominates and the curves level off.

Again, except for $\tau_d/T = 1$, the error is well below 1 percent.

Figure 6.1-3 and 6.1-4 show the encoder response to a step function of varying amplitude $A_o$. Except for the case $A_o = 0.1$ the curves are the
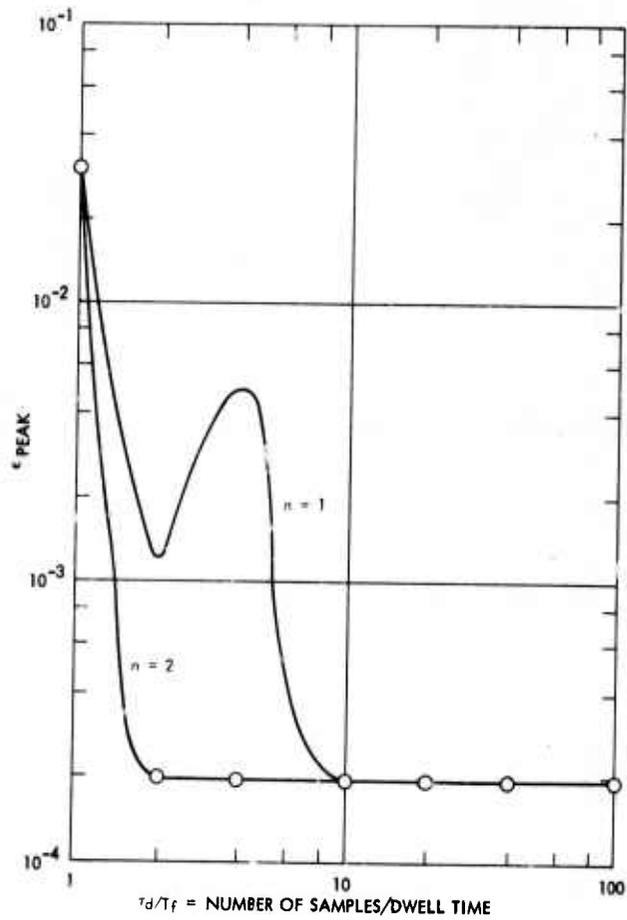
Figure 6.1-2. Peak error.
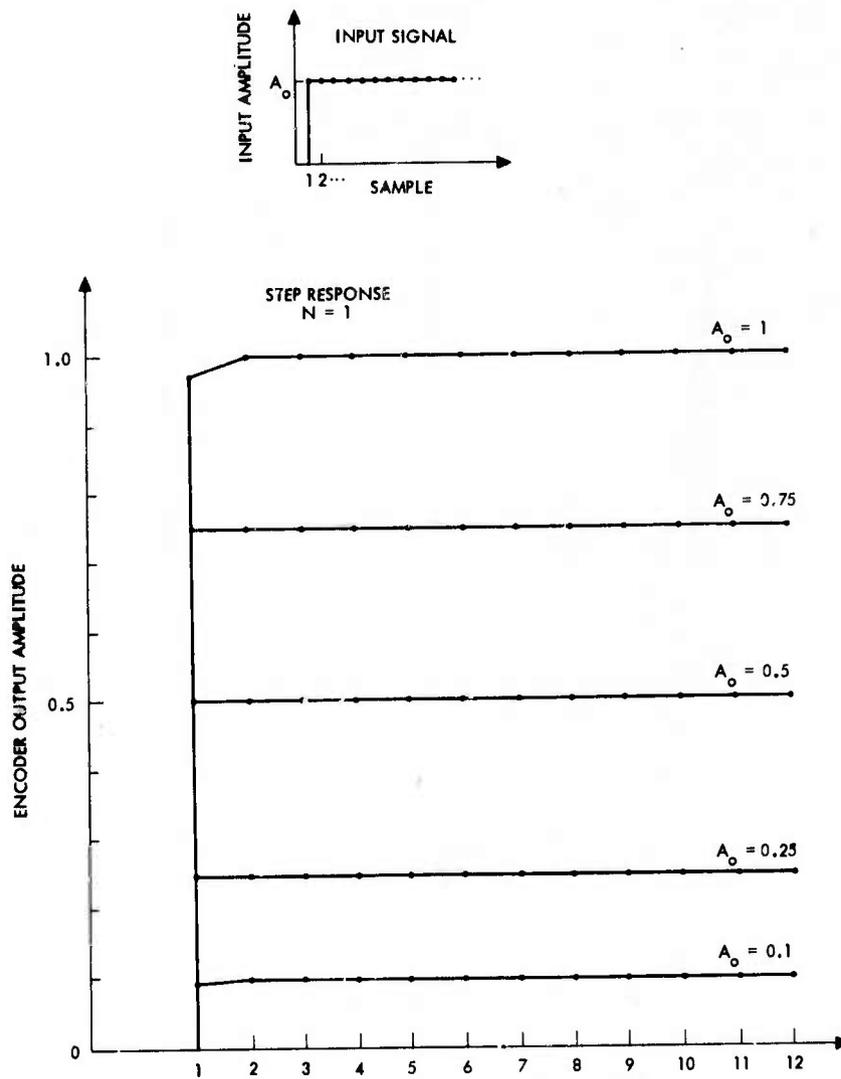
Figure 6.1-3. Encoder step response.
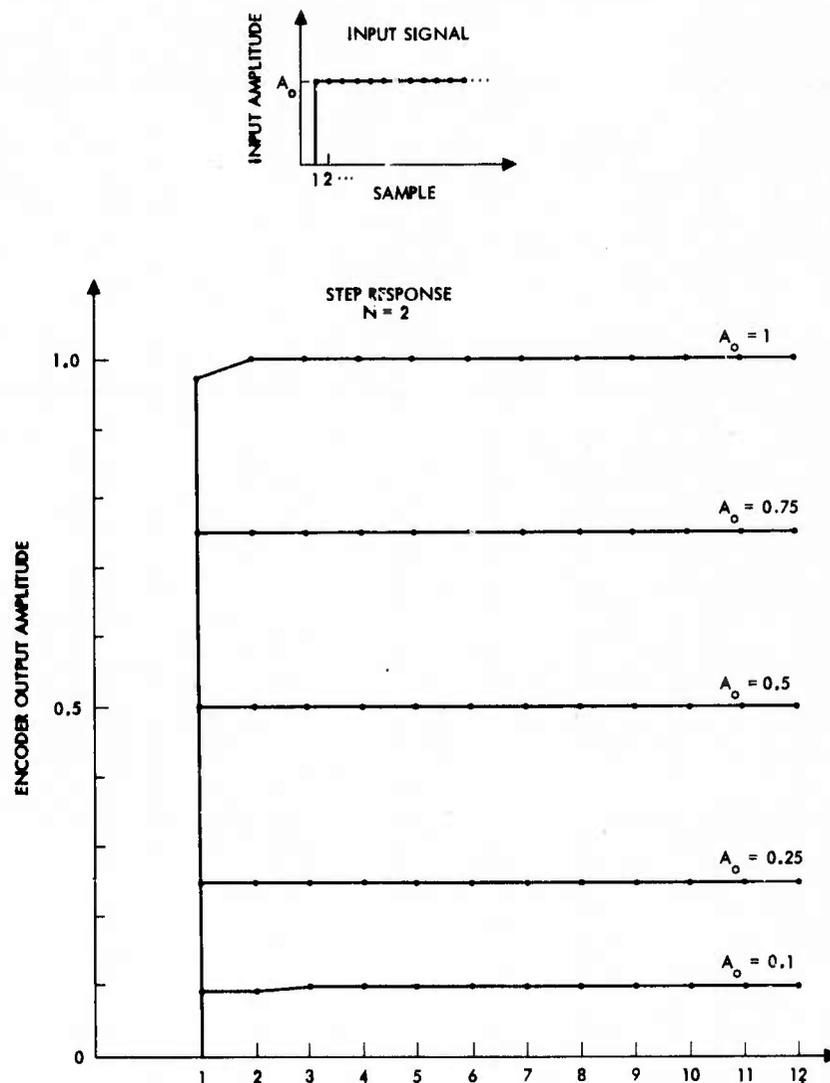
Figure 6.1-4. Encoder step response.

same. No more than n (n = 1 or 2) samples are required to reach steady state in either case.

Table 6.1-1 presents a summary of the encoder characteristics and shows how these compare with a standard 10-bit A/D converter. The results show that the encoder will convert the input signal to a 10-bit word with less than 1 percent error (for $\tau_d/T_f > 1$) over a dynamic range of 1024:1.

TABLE 6.1-1. ENCODER PERFORMANCE SUMMARY

| Parameter | Predictive Encoder (10 bit) | | A/D Converter (10 bit) |
|---|---|---|---|
| | n = 1 | n = 2 | |
| Quantization Noise | <0.08% for $\tau_d/T_f > 1$<br>2.5% for $\tau_d/T_f = 1$ | <0.06% for $\tau_d/T_f > 1$<br>1.8% for $\tau_d/T_f = 1$ | ≤0.028% |
| Signal Peak Error | <0.5% for $\tau_d/T_f > 1$<br>3% for $\tau_d/T_f = 1$ | 0.02% for $\tau_d/T_f > 1$<br>3% for $\tau_d/T_f = 1$ | ≤0.028% |
| Energy (μjoules/conversion) | 0.015 | 0.1 | 0.05 |
| Dynamic Range | 1024:1 | 1024:1 | 1024:1 |
| Conversion Time (μsec) | 6 | 6 | 6 |
| Total Power (m watts) | 2.4 | 16.4 | 8.2 |
| Transient Response (Step Input) | At most one sample time to reach steady state | At most two sample times to reach steady state | No delay |
| | < Function of input amplitude > | | |

## 6.1.2 Noise Analysis

To gain insight into the noise characteristics of the encoder consider the following development. The MFPA output is

$$Q(k) = S(k) + N(k)$$

where $S(k)$ is the signal and $N(k)$ is noise of <u>arbitrary distribution.</u> From this is subtracted the prediction $P(k)$ given by

$$P(k) = P_Q(k) + \epsilon_{D/A}(k)$$

where $P_Q$ is the value of the predicted word and $\epsilon_{D/A}$ is the noise introduced by the D/A converter. The result of the subtraction is the difference signal

$$E(k) = Q(k) - P(k) + \epsilon_{sum}(k)$$

$$= Q(k) - P_Q(k) - \epsilon_{D/A}(k) + \epsilon_{sum}(k)$$

where $\epsilon_{sum}$ is the noise due to the analog summer. Now $E(k)$ is converted to a value

$$E_Q(k) = E(k) + \epsilon_{A/D}(k)$$

where $\epsilon_{A/D}$ is the noise introduced by the A/D conversion process; $E_Q$ may now be rewritten

$$E_Q(k) = Q(k) - P_Q(k) - \epsilon_{D/A}(k) + \epsilon_{sum}(k) + \epsilon_{A/D}(k)$$

and if the encoder noise is defined by

$$\epsilon_{enc}(k) \triangleq \epsilon_{A/D}(k) + \epsilon_{sum}(k) - \epsilon_{D/A}(k)$$

6-9

then

$$E_Q(k) = Q(k) - P_Q(k) + \epsilon_{enc}(k)$$

and the predicted corrected value is

$$PC_Q(k) = E_Q(k) + P_Q(k)$$

$$= Q(k) + \epsilon_{enc}(k) = S(k) + N(k) + \epsilon_{enc}(k)$$

So, the output value is the same as the input value except for the encoder noise introduced. The encoder has done nothing to affect the input noise $N(k)$. If the input noise is assumed to be much larger than the noise introduced by the A/D and D/A converters (except when the 2-channel A/D is operating in channel 'a') then the signal-to-noise ratio at the encoder input will be degraded by any non-negligible noise introduced by the analog summation and/or channel 'a' errors.

## 6.2 Temporal Filter Performance

This subsection presents typical results on the effectiveness of the temporal detection filter. In particular, the signal to clutter ratio before and after filtering is shown as a function of the number of samples per dwell time and the noise equivalent bandwidth for the temporal filter is calculated for the system level performance analysis purposes.

The electrical signal at the detector output is filtered to eliminate the low frequency scene background. Its purpose is to reduce the dynamic range requirement of the CCD registers. The transfer function for this circuit is

$$H_B(S) = \frac{s}{s + j\omega_B}$$

where $\omega_B$ is placed at a value low enough not to reduce the target signals of interest. A reasonable value of $\omega_B/2\pi = 0.1$ Hz was chosen for performance

analysis. This is based on current technology capability and the requirement of passing all target signal frequencies of interest.

The current from this AC coupling filter is integrated and held for T seconds in the CCD storage bucket. The resulting charge samples are multiplexed out to the adaptive video encoder and converted to digital words. The transfer function for the sample and hold circuit is

$$H_T(\omega) = T \left[ \frac{\sin \omega T/2}{\omega T/2} \right]$$

This filter serves as a band limiter which prevents noise folding into the frequency domain occupied by target signals.

A Temporal Discrimination Filter provides each pixel with a dedicated filter which provides background clutter rejection. The discrimination is based on the fact that the targets are moving relative to a stationary or slowly changing clutter scene. Consequently the filter must emphasize the frequencies that contain target energy and suppress those that contain clutter, i.e., the low frequency response must be severely attenuated.

For the simulation case, it has been assumed that a radiance step moves across the detector aperture. After filtering by the optics and the detector convolution, the leading edge of this signal closely approximates a ramp. The slope of the ramp (in time) is proportional to the velocity of the clutter edge. The lowest order digital temporal filter that will give zero response to a ramp is a second differencing filter. It will have a response only at the corners of the ramp. A third difference digital filter provides zero response to a parabolic input and consequently effectively rejects the ramp that is smoothed by the impulse response of the telescope.

The transfer function for the third-order difference filter is

$$H_D(Z) = (1 - Z^{-1})^3 = 1 - 3Z^{-1} + 3Z^{-2} - Z^{-3}$$

in the Z transform domain. The frequency response is given by

$$\left| H_D(e^{j\omega T}) \right| = 2^3 \left[ \sin(\omega T/2) \right]^3$$

6-11

A transversal filter implementation is shown in Figure 3.4-2. A target pulse response is shown in Figure 3.4-4.

The system frequency response is shown in Figure 6.2-1. The system response was obtained for several different frame integration periods each corresponding to the different target velocity windows. In the implementation of the TDF an N-frame accumulator provides this function. The value of N is chosen in accordance with the target velocity. The noise equivalent bandwidth corresponding to the appropriate filter is also shown in Figure 6.2-1.

The curves in Figure 6.2-2 indicate the clutter rejection capability of the third difference TDF. The ordinate is the output to input amplitude ratio. Both the target and the clutter edge have the same signal power within the detector area. The amplitudes have been calculated for a 0.1 second integration time. They are plotted against the dimensionless constant

$$T/\tau_d = (\text{Number of samples per dwell time})^{-1}$$

where

$T$ = integration (frame) time, sec

$\tau_d$ = target or clutter dwell time, sec

From this figure it is apparent that an optimum frame time to dwell time ratio for targets is about 0.5 whereas for clutter, the optimum ratio is zero. Hence optimizing the signal response subject to the constraints of maximum signal-to-clutter ratio yields a value of target frame time to dwell time of 0.2 to 0.3.

## 6.3 Tracking Performance

The purpose of the track processor is to correlate target observations into multiple tracks. The principal algorithms for doing this and their functions are briefly described below.

Association and correlation refers to the techniques by which received observations are assigned to existing tracks.
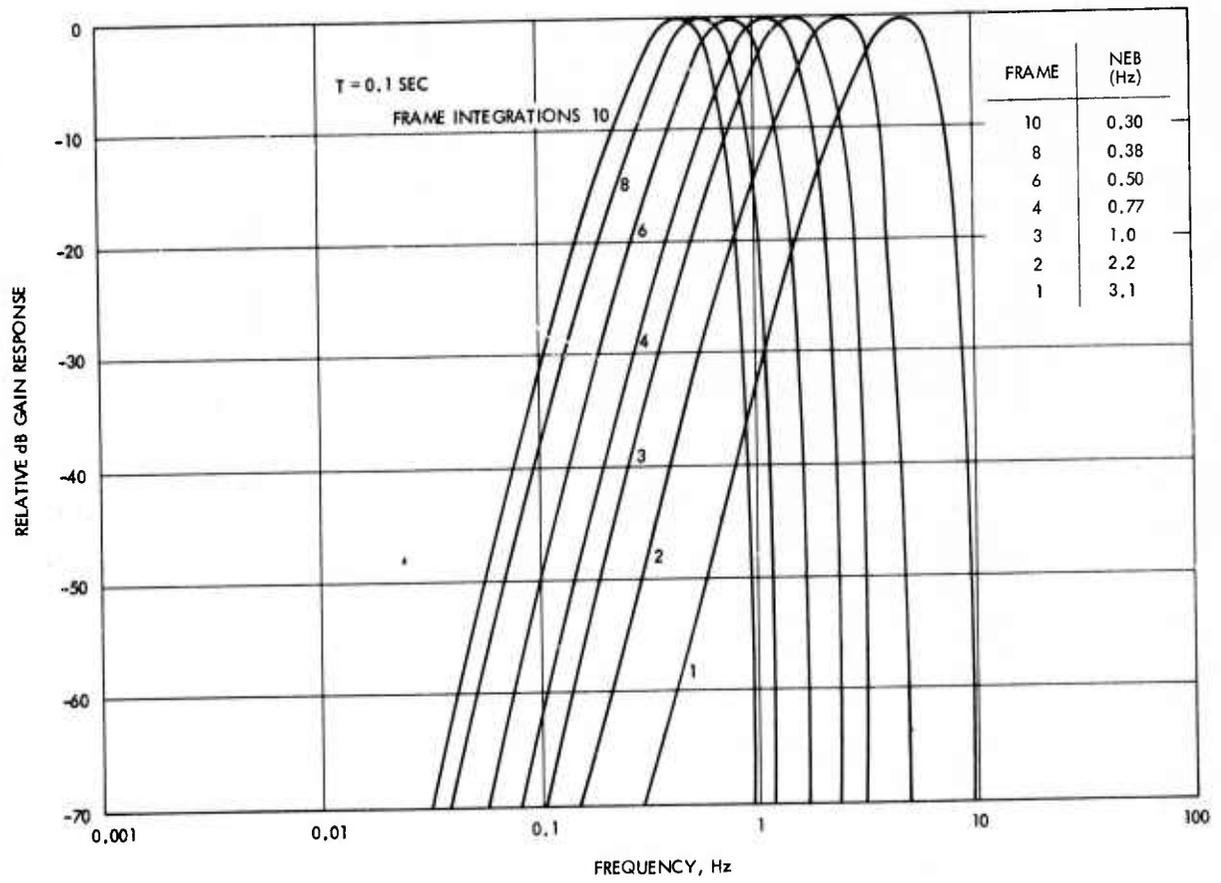
Figure 6.2-1. Normalized system response from temporal discrimination filter.
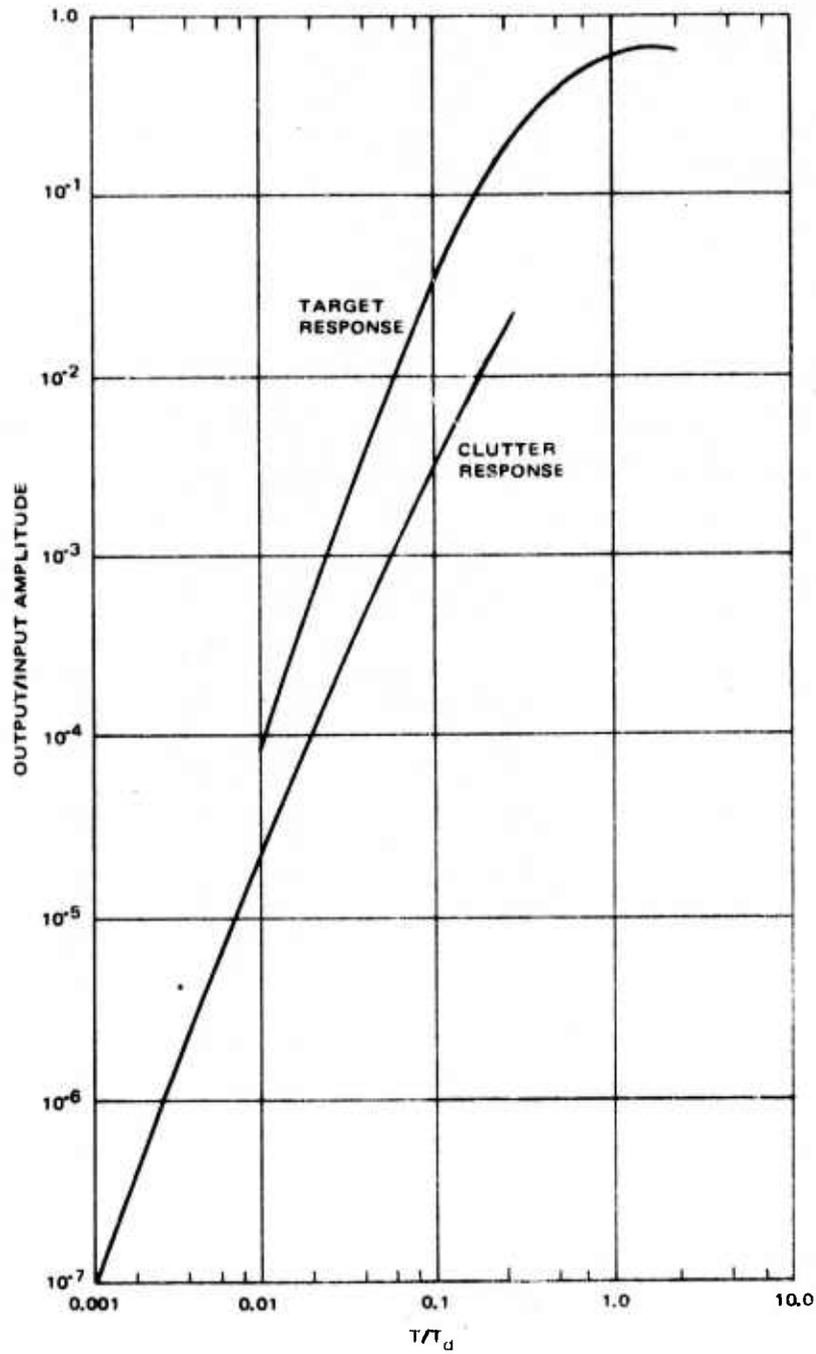
Figure 6.2-2. Third-order difference TDF target and clutter response.

Track initiation is the process of using new observations that are not associated with existing tracks to form new tracks.

Track deletion algorithms are designed to remove low quality tracks from future consideration.

Track update and prediction algorithms are used to incorporate new correlating observations into the existing tracks and form new state variable estimates.

Gates are formed around the tracks' predicted positions and are used to limit the number of observations considered for potential track update.

Measurement Error in Time Centroiding

Time centroiding determines the position of a target image along its direction of motion. A common method for motion-direction centroiding is the determination of the time of the peak signal at the output of the pixel as shown in Figure 6.2-3. Notice that a threshold is shown at about half the level of the expected peak signal. This eliminates most of the ambiguities caused by the presence of the target on more than one pixel.

Errors are introduced by the presence of noise, which distorts the signal enough to give a false indication of the peak. Furthermore, the finite sampling time is a cause of error, since the time of the maximum sample is uniformly distributed over the sampling interval with respect to the peak. Finally, the relationship of the size of the point-spread function with respect to the size of the pixel affects the shape of the output signal.

An upper bound on the centroiding error can be found from the case of no centroiding; when a detection is made on a given pixel, the target image is assumed to be in the center of its path across the pixel. The location of the peak is then a uniformly distributed random variable along the path. For the path in case A of Figure 6.2-3, this yields $\sigma_c^2 = D^2/12$, where $D$ is the pixel dimension. For the path in Case B, $\sigma_c^2 = (\sqrt{2}\,D)^2/12$. Normalizing these errors to the pixel dimension yields $\sigma_c$ (Case A) = 0.289 and $\sigma_c$ (Case B) = 0.408.
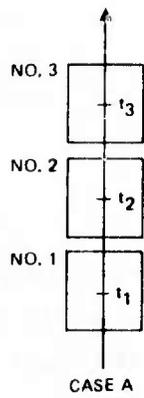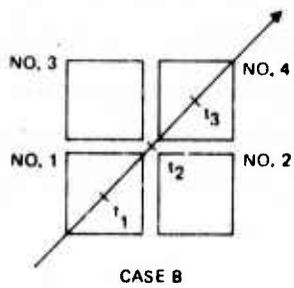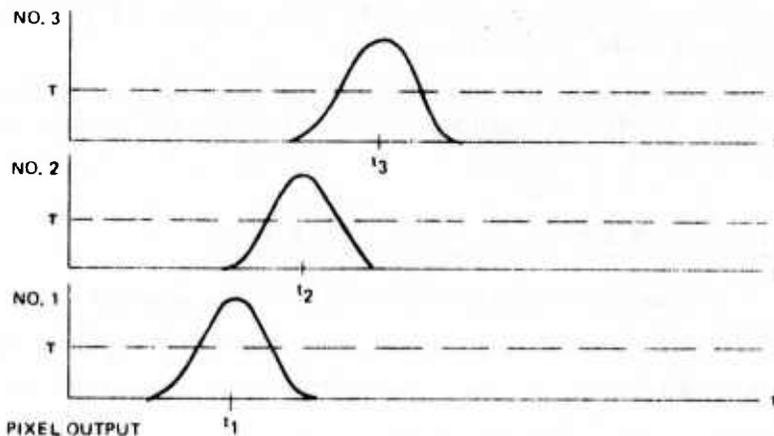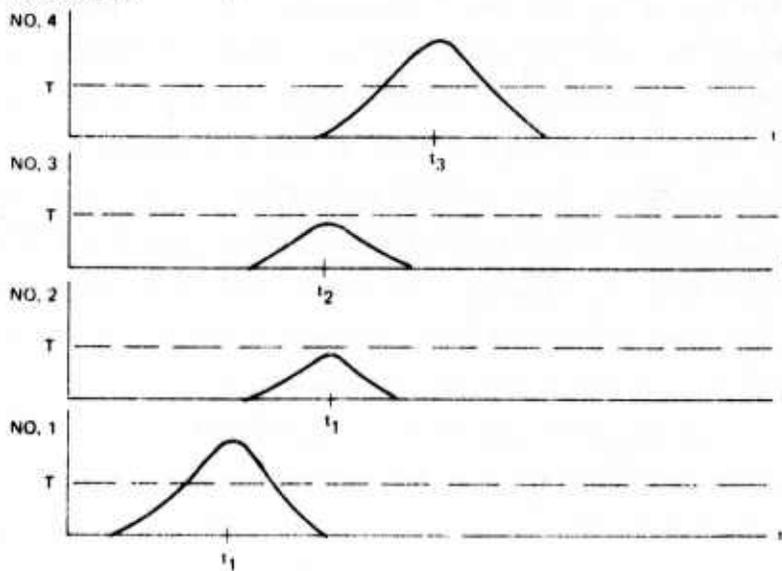
Figure 6.2-3. Time centroiding.

## Track Initiation, Maintenance and Filtering

Following References 1 and 2 and using a maximum likelihood approach a score function ($L_k$) may be defined for the incorporation of K frames of data into $N_k$ tracks

$$L_K = n_K \ln \frac{\beta'}{\lambda'_N} + \sum_{i=1}^{n_K} \left\{ \ln\left[P_{TL}(D_i)\right] + (D_i - m'_i)\ln(1-P_D) \right.$$

$$\left. + \sum_{\ell=2}^{m_i} \left( \ln\left[\frac{P_D}{\lambda'_N (2\pi)^{M/2} \sqrt{|V_{i\ell}|}}\right] - \frac{\delta^2_{i\ell}}{2} \right) \right\},$$

(6.2-1)

where

$M$ = measurement dimensionality,

$D_i$ = track (i) length,

$P_{TL}(D_i)$ = probability of track length $D_i$

$B'$ (n) = true (false) target density,

$P_D \triangleq$ detection probability,

$m_i \triangleq$ total number of target detections including the initial observation

$m'_i \triangleq m_i - 1$,

$n_K \triangleq$ number of tracks formed based upon data received through frame K,

$\tilde{Y}'_{i\ell}$ = residual error for the $\ell$th update of the i th track,

$\delta^2_{i\ell} \triangleq \tilde{y}^T_{i\ell} V^{-1}_{i\ell} \tilde{y}_{i\ell}$,

$V_{i\ell}$ = residual error covariance matrix

6-17

Techniques for using Eq. (6.2-1) to determine track initiation and deletion criteria are discussed in Reference 2. A desirable property of these techniques is that any combination of observations into tracks must lead to a score function greater than zero. As the results given in Reference 2 indicate, this property is particularly convenient for use in defining initiation and deletion criteria so that false tracks are quickly terminated. Also, initiation and deletion criteria can be made adaptive in an optimal manner to the environment.

Ideally, true target density, probability of detection, and track length statistics as well as residual error and false target density should be used. However, these parameters are not presently defined. Thus, to derive preliminary results, tentative algorithms are considered whereby initiation is based upon receiving four correlating observations within five consecutive frames and deletion occurs on six consecutive frames without a correlating observation.

Figure 6.2-4 illustrates the manner in which track initiation and maintenance vary with the detection probability. The probability of having a confirmed track is presented as a function of frame number. Case 1 shows results for the situation where

$$P_D \text{ (case 1)} = \begin{cases} 0.4, \text{ initial detection,} \\ 0.9, \text{ during initiation,} \\ 0.95, \text{ after confirmation.} \end{cases}$$

Case 2 has a cyclic probability of detection with a period of 33 frames

$$P_D \text{ (case 2)} = \begin{cases} 0.95, \text{ first nine frames} \\ 0.94, \ 0.93, \ 0.92, \ 0.91, \ 0.87, \ 0.83, \\ 0.73, \ 0.59, \ 0.42, \ 0.31, \ 0.28, \ 0.27, \\ 0.28, \ 0.31, \ 0.42, \ 0.59, \ 0.73, \ 0.83, \\ 0.91, \ 0.92, \ 0.93, \ 0.94, \text{ frames 10-33.} \end{cases}$$

6-18
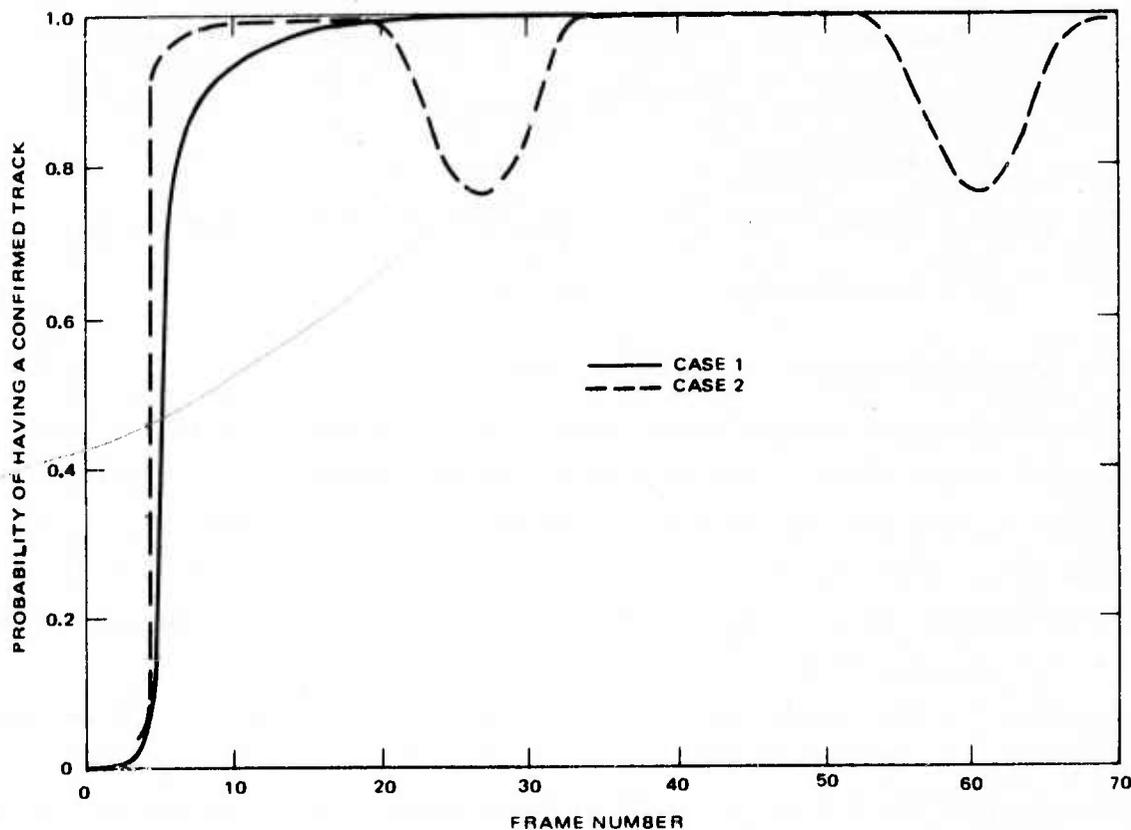
Figure 6.2-4. Comparative track initiation and maintenance.

For track filtering and prediction of both aircraft and missile targets the classical (Reference 3) $\alpha$-$\beta$ tracker is proposed. It is defined by the following equations

$$X_s(n) = X_p(n) + \alpha(n) \, \Delta X(n),$$

$$\dot{X}_s(n) = \dot{X}_s(n-1) + \frac{\beta(n)}{T} \, \Delta X(n), \qquad (6.2-2)$$

$$X_p(n+1) = X_s(n) + T \, \dot{X}_s(n),$$

where, from Reference 3, s = smooth, p = predicted and o = observed,

$$\alpha = \beta^2/(2-\alpha) = \text{coefficient}$$

$$X = \text{state vector}$$

$$\Delta X(n) = X_o(n) - X_p(n) \overset{\Delta}{=} \text{difference between observation and prediction}$$

$$T \overset{\Delta}{=} \text{sampling interval, frame time, sec}$$

### Aircraft Position Tracking Results

A covariance analysis applicable to maneuvering aircraft has been performed to determine prediction error standard deviation as a function of the sampling interval, the detection probability and the prediction (or extrapolation) time. The results, given in Figure 6.2-5, were not found to be a sensitive function of $\alpha$ but are given for the $\alpha$ that minimizes the prediction error standard deviation.

The assumed white measurement noise standard deviation (0.29 pixel) corresponds to a uniformly distributed quantization error. Target maneuver characteristics are defined to be first order Markov with standard deviation 0.25g and time constant of 3.0 sec. Results are given for one step prediction and extrapolation periods of 7.5 and 15 sec. These periods correspond to extrapolation across gaps of 5 to 10 pixels for typical aircraft targets.

### Missile Tracking

The proposed missile sampling interval is 0.1 sec. The probability of detection is expected to be high for missiles during most of their trajectory. Referring to Figure 6.2-5, tracking performance is generally insensitive to detection probability when the probability is 0.7 or greater and when the sampling interval is small (T = 0.25 sec). Thus the results presented below will assume unity probability of detection.

Typical missile position time histories appear to be approximately characterized by a constant acceleration. Given a constant acceleration the bias error $(e_{x_p})$ for an $\alpha$-$\beta$ tracker is given by

$$e_{x_p} = \frac{4 - 2\alpha - \alpha^2}{2\alpha^2} \, aT^2, \tag{6.2-3}$$

6-20
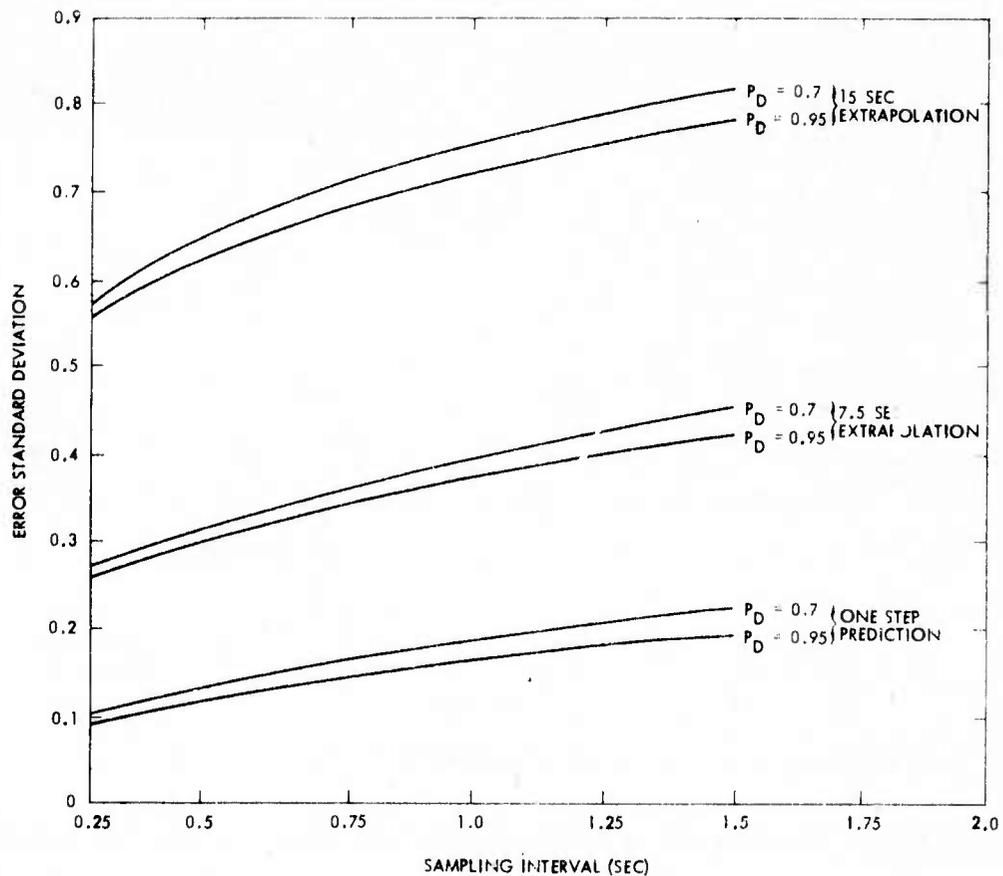
Figure 6.2-5.   Maneuvering aircraft tracking error
standard deviation.

Typical missile accelerations appear to be about 0.03 pixel/sec$^2$ or less.   Thus, using Eq. (6.2-3), for a values of 0.1 or greater and a sampling interval of 0.1 sec., the bias error is negligible.

Figure 6.2-6 shows a typical transient error response using an a of 0.1 when tracking is begun at 20 sec.   These results include the effect of the quantization measurement error for a particular case.   The tracking error shown in Figure 6.2-6 does not quite reach steady state.   In steady state, the prediction error was found to oscillate between 0.04 and 0.14 in agreement with Eq. (6.2-3).   The initial error can be reduced by developing a special initiation technique.
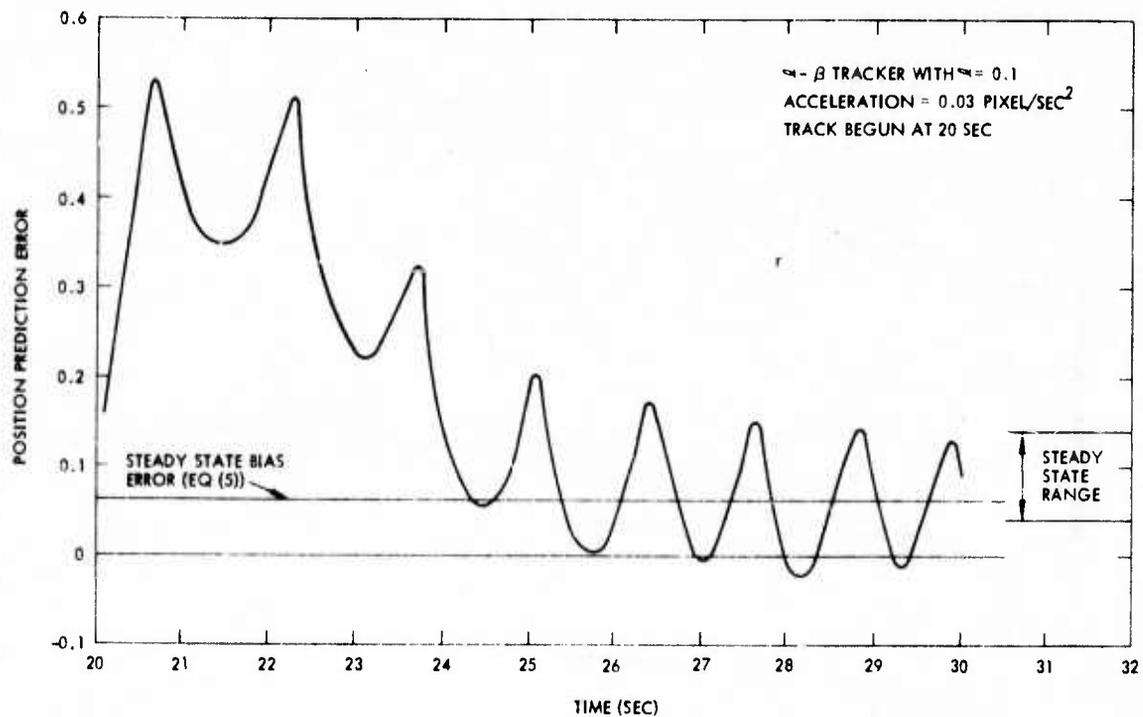
Figure 6.2-6. Transient missile tracking error.

Intensity is considered to be an important discriminant for missile tracking and identification. Preliminary results indicate that an α-β tracker will also suffice for intensity tracking. Figure 6.2-7 shows the mean intensity estimation error for values of α = 0.1 and 0.2 for a typical case. The estimation error is given as a percentage of the true intensity.

Multiple Target Effects

Classically, as discussed in Reference 2, multiple target interactions are handled by using gates around each target's predicted position as a preliminary screening device and a correlation matrix for resolving complex conflict situations. This standard technique allows at most one observation to be assigned to a track. However, other techniques, discussed in References 4 and 5, propose the use of more than one observation for track update. The choice of technique will be studied further when the tracking environment is defined.
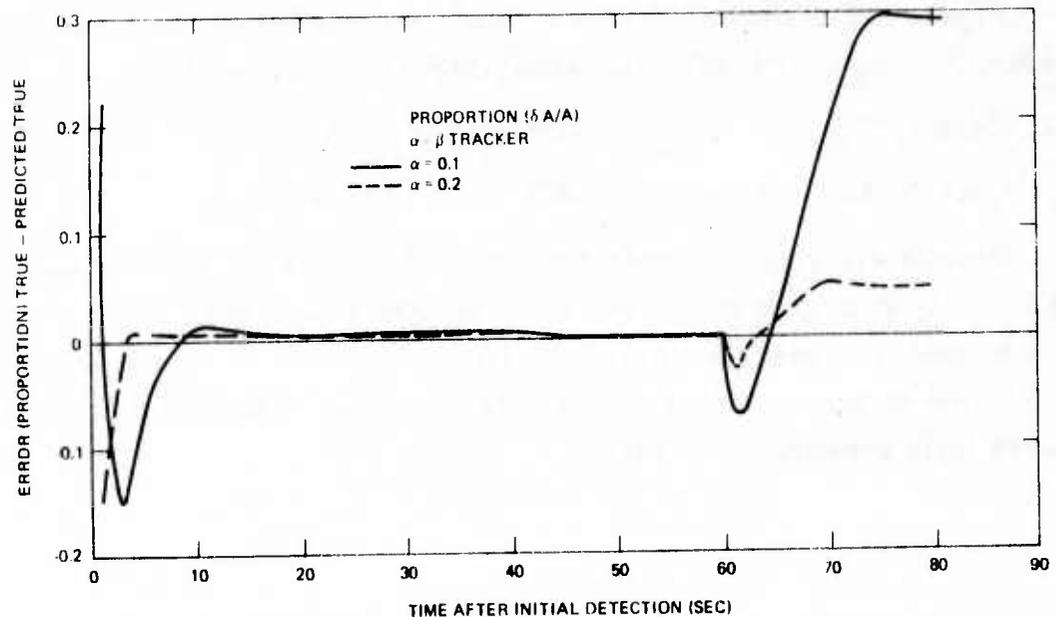
Figure 6.2-7. Intensity tracking error.

## Crossing Tracks

As target tracks come together it becomes more difficult to correctly assign observations to the tracks. In the limit where both targets are within the same pixel there is no unique measurement. Eventually, some type of measurement centroiding logic may be developed using the methods discussed in References 4 and 5. Using this logic, observation may be used for update by both tracks.

One technique for handling a track cross is to predict its occurrence and then to extrapolate the tracks ahead until they again become distinguishable. Then, regular tracking would be reestablished. Results derived using this technique should give a worst case bound on performance because the observations received during the extrapolation period are not used. We assume that extrapolation begins and ends when at least one pixel separation (in either dimension) is assured with probability 0.977 (two standard deviation case).

Figure 6.2-8 shows the error standard deviation at the time immediately preceding track reestablishment for two cases:

Case 1: $P_D = 0.7$, $\sigma_m = 0.25$ g, $T = 0.25$ sec,

Case 2: $P_D = 0.95$, $\sigma_m = 0.1$ g, $T = 1.0$ sec.

Results are given for velocities of 0.35 pixel/sec (approximately Mach 1) for both targets as a function of the interaction angle $\theta$. This simple technique appears adequate for some conditions; but as $\theta$ decreases below around 45 deg., or if the velocities decrease, some form of more advanced logic appears necessary.
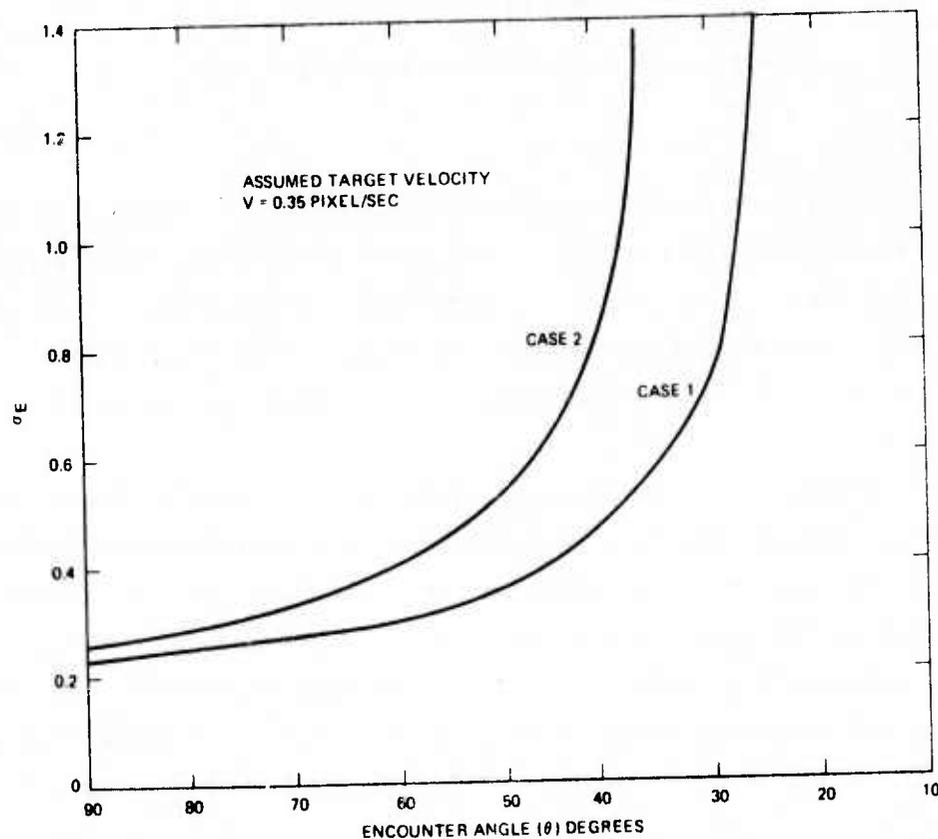


Figure 6.2-8. Worst-case error standard deviation at track re-establishment.

## Issues for Further Consideration

As discussed above it is necessary that the parameters of the tracking environment be specified before efficient tracking algorithms and realistic performance estimates can be defined. This is particularly true of the target detection probability and false target density and distribution.

Further study should be performed on the tradeoffs between computational complexity and tracking performance. For example, in the filtering area, the increased complexity of an $\alpha$-$\beta$-$\gamma$ tracker would allow the tracking of an acceleration with no steady state bias error. However, preliminary results indicate that this may not be necessary. Also, as the detection probability becomes known the use of a Kalman filter may be considered.

Several other techniques that would provide tracking improvements but also require additional computations are being considered. First, following References 1 and 2, a centroided measurement technique is being considered. In the presence of a high false target density and a moderate to large sampling interval ($T \gtrsim 0.5$ sec) preliminary results indicate that this technique should reduce the tracking error and the probability of track divergence. However, this method is computationally complex.

Finally track splicing and handover to other sensors is an important issue in the system context.

## References for Section 6.2

1. R. W. Sittler, "An Optimal Data Association Problem in Surveillance Theory", IEEE Trans. on Military Electronics, vol. MIL-8, pp. 125-139, April 1964.

2. J. J. Stein and S. S. Blackman, "Generalized Correlation of Multi-Target Track Data", IEEE Trans. on Aerospace and Electronic Systems, Vol. AWS-11, November 1975, pp. 1207-1217.

3. T. R. Benedict and G. W. Bordner, "Synthesis of an Optimal Set of Radar Track-While-Scan Smoothing Equations", IRE Trans. on Automatic Control, Vol. AC-7, July 1962, pp. 27-32.

4. R. A. Singer, R. G. Sea, and K. B. Housewright, "Derivation and Evaluation of Improved Tracking Filters for Use in Dense Multi-Target Environments", IEEE Trans. on Information Theory, Vol. IT-20, No. 4, July 1974, pp. 423-432.

5. Y. Bar-Shalom, "Extension of the Probabilistic Data Association Filter to Multi-Target Tracking", Proc. of the 5th Symp. on Nonlinear Estimation, San Diego, California, September 1974, pp. 16-21.

# TECHNOLOGY SURVEY
## FOR
## ADAPTIVE PROGRAMMABLE SIGNAL PROCESSOR

| | |
|---|---|
| ARPA Order Number | 2954, Amendment No. 1 |
| Program Code Number | None |
| Name of Contractor | Hughes Aircraft Company |
| Effective Date of Contract | 20 June 1975 |
| Contract Expiration Date | 13 February 1976 |
| Amount of Contract | $498,159 |
| Contract Number | F04701-75-C-0241 |
| Principal Investigator and Phone Number | K.E. Myers, 391-0711, X7598 |
| Project Engineer and Phone Number | K.A. Krause 391-0711, X2243 |
| Short Title of Work | Technology Survey for APSP |
| Date of Report | 5 December 1975 |
| Contract Period Covered by Report | 25 June 1975 to 21 November 1975 |

K.E. Myers
Program Manager

## Electro-Optical Division
## AEROSPACE GROUPS
## Hughes Aircraft Company ● Culver City, California

# TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

iv

# LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

## LIST OF ILLUSTRATIONS (Continued)

# LIST OF TABLES

# 1.0 INTRODUCTION

This report represents the results of the Device Technology Survey task conducted for the Adaptive Programmable Signal Processor (APSP) program. The purpose of this task was to determine the availability (both present and projected) of semiconductor devices applicable to the APSP design effort.

The task is broadly divided into three portions:

1. A survey of both present and projected availabilities of semiconductor devices applicable to design of the Layered Array Processor (LAP) in the APSP. The survey treats both technologies (e.g., $I^2L$, CMOS, etc.) and devices (e.g., microprocessors, memories, etc.).

2. A brief discussion of the Adaptive Video Encoder (AVE) of the APSP in terms of its function and the associated critical devices required.

3. The fabrication and testing at Hughes, of a variety of Hughes-designed microelectronic devices, including CCD compatible bipolar devices, MOS, integrated injection logic ($I^2L$) and charge coupled devices (CCD's).

Examination of software applicable to the APSP design has been deferred to a more appropriate forthcoming report, E-O Processor Definition. Another forthcoming report, Critical Device Design, will with this study as a basis, examine those devices and processes that are critical to the APSP design, as defined in the currently ongoing E.O. Processor Definition Task; and it will provide preliminary designs, evaluate associated technical risks and supply appropriate development schedules.

## 2.0 DIGITAL TECHNOLOGY STATUS

This section provides a survey of the present status and probable future of those digital technologies applicable to the Adaptive Programmable Signal Processor (APSP). In section 2.1, a brief review of the fundamental limitations on digital devices is presented, and is applicable to both the basic logic gates which make up memory and logic functions, and the microprocessors and associated digital systems utilizing those gates. The development of commercial microprocessors is reviewed in section 2.2, and anticipated future performance predictions are developed.

Section 2.3 reviews the field of high speed low power memory, as required for the APSP concept, and projects its future trends.

The computational capabilities, size, power and cost of the APSP depend upon the characteristics of the digital technology available in the early 1980s. Section 2.4 covers the generic field of logic devices suitable for Large Scale Integration (LSI), and eliminates all except the principal contenders for low cost, low power-delay product and high density capability within the next seven years. The three digital LSI technologies that emerge; $I^2L$, CMOS and DMOS, are discussed in further detail in sections 2.5, 2.6 and 2.7 respectively.

Figure 2.0-1 illustrates the present power-delay products of the various LSI technologies. Ring oscillator circuits are used as the standard of comparison for new technologies. Typically, an order of magnitude degradation in power delay product is experienced when adding the fan-out, interconnects and output devices associated with performing large arithmetic functions with LSI. Anticipated future improvements in the various technologies are summarized in section 5.0, CONCLUSIONS.
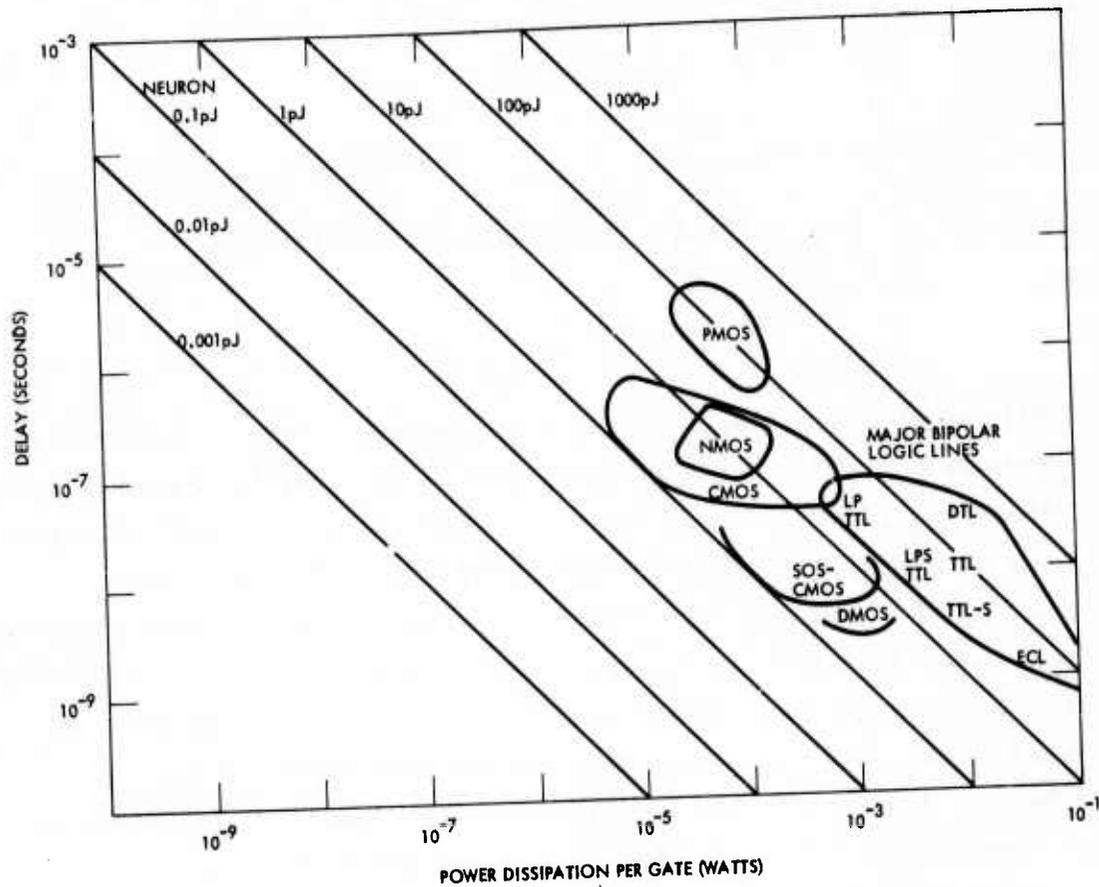
Figure 2.0-1. Mid 1975 LSI technology power delay products

## 2.1 FUNDAMENTAL LIMITATIONS ON DEVICE PERFORMANCE

In evaluating various device technologies for computer hardware applications, numerous parameters must be considered. Those devices that come closest to optimizing these key parameters will emerge as the most suitable. These considerations are categorized into costs and values:

### Costs

1. chip real estate
2. number of processing steps required

### Values

1. speed
2. speed-power product
3. density

4. noise immunity
5. device fan out
6. power supplies required
7. compatibility with existing technologies
8. natural radiation tolerance
9. life/reliability

Of primary importance is speed-power product. If the circuitry can be processed sufficiently small and a low enough speed-power product is obtained, a good deal of the search is over. Today's technology offers a wide selection of logic families, ranging from the excellent speed of emitter coupled logic (ECL) to the very low power of CMOS. The real need, however, is for an optimum compromise of speed, power and silicon real estate. Various MOS technologies (n-MOS, silicon gate n-MOS, CMOS) have been addressing this problem for some time. Recently, bipolar logic has become a true competitor to MOS, with the advent of Integrated Injection Logic ($I^2L$).

It is only a matter of time before sufficient performance will be available to enable an optimum choice. It is possible, even probable, that many of the decisions regarding applications of the various device technologies will soon become obvious. For instance, since the improved bipolar logic ($I^2L$) must constantly draw current, even in the off condition, it might be more applicable to a continuous data system. In constrast, CMOS logic consumes almost no power in the off condition, rendering it suitable for systems where data is being handled only intermittently.

To push speed-power product to its absolute limits, the factors that actually limit logic performance must be considered in detail.

Swanson[2.6-1] discusses a hierarchy of limitations in regard to logic devices. First are the absolute physical limits based on two fundamental laws of physics: thermodynamics and quantum mechanics. Three different approaches involving basic thermodynamic properties arrive at the same conclusion; that the energy consumed by a logic gate is greater than $4kT_a$ ($1.7 \times 10^{-20}$ J). Since power $(P) = \frac{E_t}{\tau_d}$, where $\tau_d$ is the propagation delay through the gate, the thermodynamic limit related in terms of power is:

$$P \geq \frac{4kT_a}{\tau_d}$$

2-3

where

$k$ = Boltzmans constant

$T_a$ = ambient temperature

During a logic state transition it can be assumed that an energy barrier is transversed in a device, such that an energy $E_t$ is dissipated in the transition. Swanson states that the minimum time for this to occur is about $h/E_t$ where h is Planck's constant. Thus the quantum mechanical limit requires:

$$\tau_d \geq \frac{h}{E_t}$$

or

$$E_t \geq \frac{h}{\tau_d}$$

and in terms of power, since

$$P = \frac{E_t}{\tau_d}$$

this yields

$$P \geq \frac{h}{\tau_d^2}$$

These power considerations hold only for maximum switching rates and comparisons between technologies should be made with this in mind. As mentioned previously, CMOS integrated circuits consume virtually no static power. This is also the case in magnetic core memories. Thus duty cycle or standby power factor is another important design consideration.

2-4

There are also limitations due to the properties of the material being used.

Using a potential hill to simulate the channel of a MOST or the base of a bipolar transistor, it is shown that the electric field that can be supported by a particular device limits the speed of the device. How well the medium can carry heat away from the area of importance is another limiting factor, termed the thermal conductivity limit by Swanson. Finally, the length of time required to propagate a signal to interconnected devices also limits speed. All of the above factors impose approximately the same speed limitation on logic circuits; the time it takes to perform a single computation must be greater than $3 \times 10^{-14}$ seconds.

The fundamental limitations are plotted in Figure 2.1-1.

Of course, actual logic device performance is orders of magnitude away from these theoretical limits. Processing techniques and the general technological state-of-the-art in the various device technologies have become the prime performance limiting factors.

The technologies which are promising for high speed - low power operation include $I^2L$, CMOS, DMOS, CCD logic and a few others. Each of these is discussed in detail, following the commercial microprocessor and memory technology surveys.

## 2.2 COMMERCIAL MICROPROCESSOR DEVELOPMENT

A microcomputer, shown in Figure 2.2-1, is a general purpose computer having three basic elements: Memory, Control and a Microprocessor. The memory is used for program storage as well as scratchpad memory. The control electronics interface with peripheral units and acts as monitor and router of data within the microcomputer. The microprocessor contains the central processing unit (CPU) in which all the arithmetic and logical operations are performed.

By use of the LSI technology, it has become possible to place the microprocessor function on a single LSI chip. As semiconductor technology produces still more dense, lower power devices, more and more functions will be incorporated in the basic microprocessor chip.
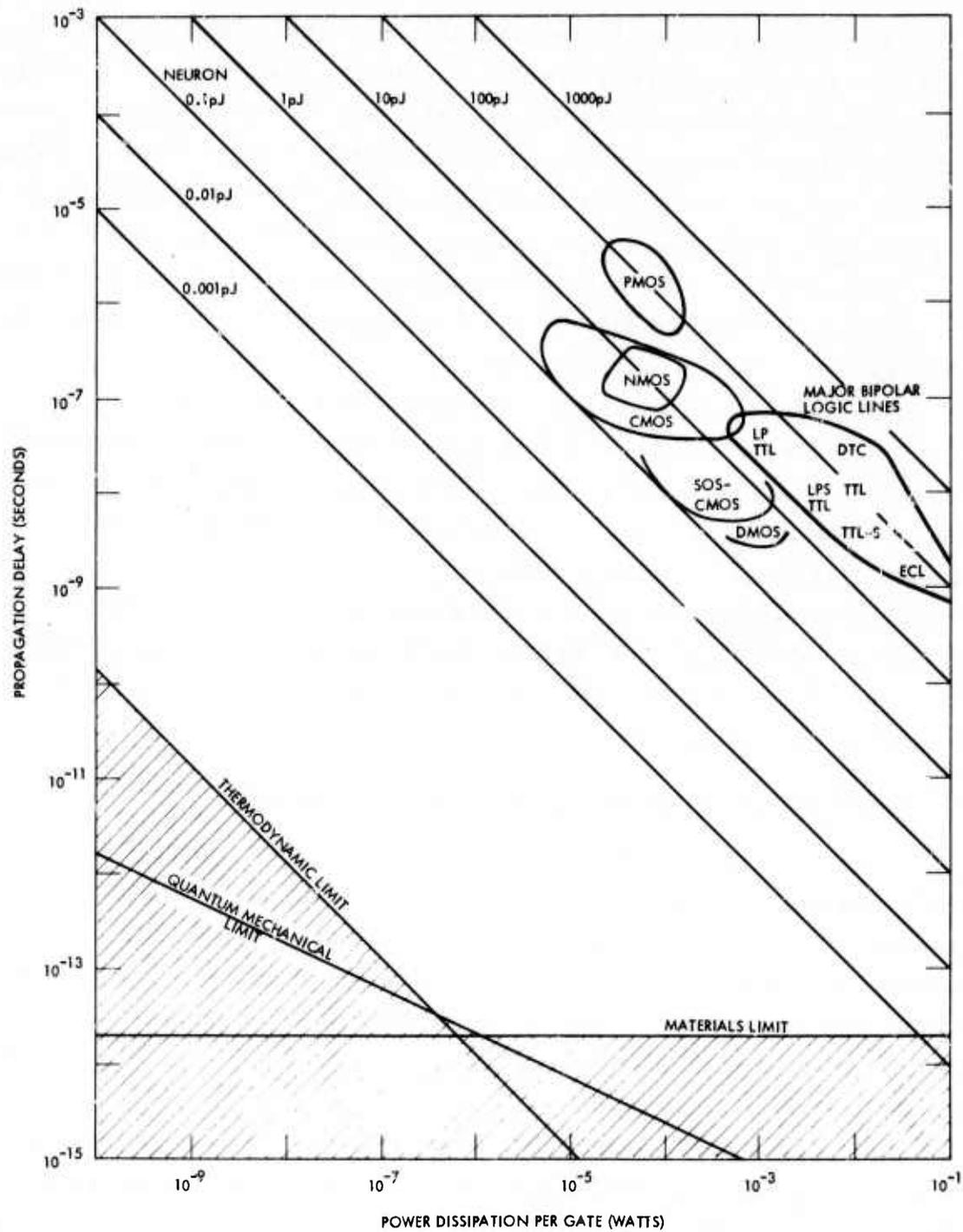
Figure 2.1-1. Fundamental gate logic power and propagation delay limitations.
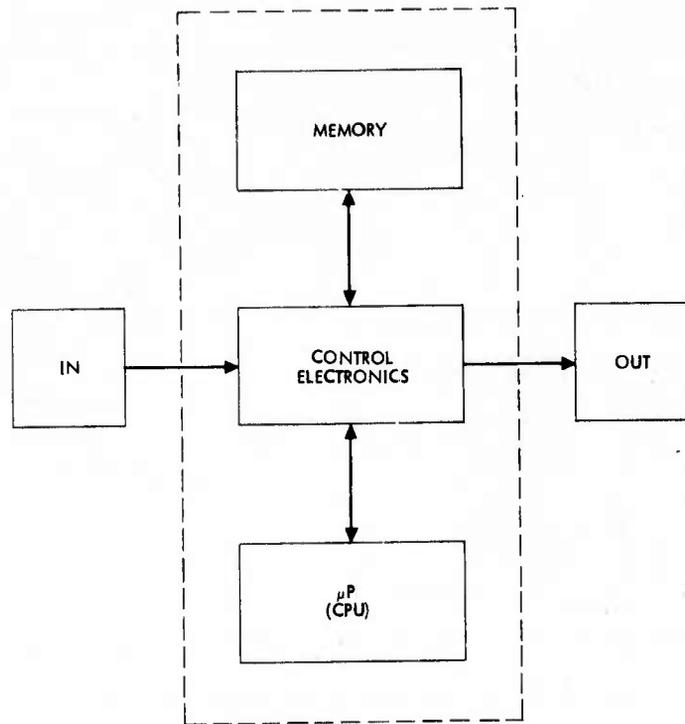
Figure 2.2-1.  A basic microcomputer.

The first microprocessor, introduced in 1971, heralded the beginning of a new field in electronics that has experienced dramatic growth. That first 4-bit PMOS device has since been joined by more sophisticated and faster microprocessors implemented with many other technologies.  By mid-1974, the number of 8-bit microprocessors grew to over twenty, and by the end of 1976 this number may triple.  Figure 2.2-2 is a diagram of micro-processor chronology.

The fact that microprocessors have been introduced so recently and the fact that the field has expanded so rapidly, complicates the study of them and makes many projections of future development somewhat tenuous.  This section shows the current status of the commercial microprocessor field and illustrates its trends of development.  Using these trends, an attempt will be made to display what can be expected to be developed in the next two to five years and thence even further into the future.
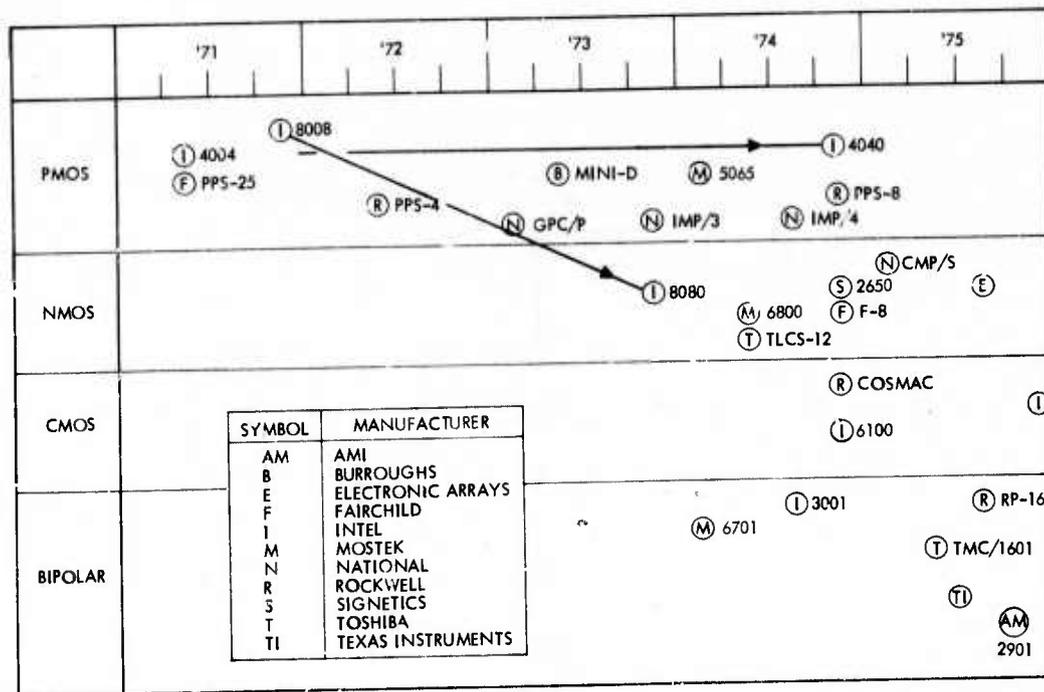
Figure 2.2-2. Microprocessor chronology.

## 2.2.1 Microprocessor Background

Although microprocessors have been available for about five years, their use has not been sufficiently widespread to insure universal understanding of them, the programs that control them, or the systems of which they are a part. Thus it may be useful to describe briefly the general purpose microprocessor as well as the resulting programming language.

A microprocessor is a compact digital processor implemented in LSI technology on one or a small number of semiconductor chips. The microprocessor corresponds to the Central Processing Unit (CPU) of a large computer. The microprocessor typically contains an Arithmetic Logic Unit (ALU) to perform arithmetic and logical operations, one or more accumulators, and registers for temporary storage of data items such as the program counter, instructions, and memory addresses.

Microprocessors presently available are characterized by:

1. PMOS, NMOS, CMOS, TTL Schottky, and $I^2L$ semiconductor device technology.

2. A data word length of 4, 8, 12 or 16 bits.

3. Parallel organization

4. Macroinstruction cycle time from 0.2 μs to 60 μs.

5. Fixed or microprogrammed instruction sets.

6. Memory address capability up to 64K words.

7. Instruction sets having 25 to 100 instructions.

8. Simple input/output structures.

9. Integrated circuits packaged in 16 to 42 pin dual-in-line packages.

10. Low power consumption (<10 watts total)

From an applications viewpoint, the microprocessor can be regarded as an alternative to random logic or custom LSI components. Using a versatile standard microprocessor, a complicated system can now be implemented in a matter of months instead of the years that might be required to design and fabricate a custom LSI device. Instead of random logic, a program in memory is used to control a microprocessor to accomplish the task at hand. Thus, the microprocessor accomplishes jobs previously done sequentially by hardware.

There are two types of microcomputer programming to be understood. The first is an Assembly language type. Microprocessors programmed in this language accomplish discrete tasks as required by the program mnemonic. For example, the instruction Add to Memory would:

1. fetch a word from memory as addressed by the instruction word or a register,

2. place the word on an ALU input,

3. place the output of an accumulator at the other ALU input,

4. place the ALU in an add mode

5. place the result of the add operation into the accumulator.

To the programmer, it would appear that these operations took place simultaneously. However if a check were made of the time required for this instruction, it would be seen that the time is longer than for an Add Immediate instruction. The difference is memory access time. With assembly language type programming, only the required functions are specified and the instruction execution time will always be a function of its complexity.

Recently, microprocessors have been fabricated which have either all or part of their instruction set microprogrammable. This feature allows the user to define his own instructions by writing a program describing the discrete steps that the microprocessor must follow. The user defined word creates the proper multiplexer, ALU, input/output, and memory configurations to accomplish the desired task. This level of programming is called microprogramming and usually the microprogram itself is called firmware. A characteristic of firmware is that each microstep is accomplished in the same amount of time (a microcycle). The advantage of microprogramming is that the microprocessor can be tailored to accomplish a set of desired tasks with maximum speed and efficiency. The disadvantage of this method of programming is that the firmware is totally hardware dependent and the programmer must be familiar with the hardware at a data path level.

## 2.2.2 Computing Power of Microprocessors

The first microprocessors introduced in 1971 were 4-bit PMOS machines. By 1975 numerous 8 and 16-bit machines became available in technologies considered superior to PMOS. Recently several companies have developed microprocessor elements in 2 or 4-bit slices, that can be connected together to produce any reasonable word length that is a multiple of 2 or 4.

The increase in the microprocessor word length is worthy of interest because the alternative of processing double precision data is both and time consuming.

Applications requiring multiplication or division also provide an interesting challenge for the designer using microprocessors. In a computer containing only ALUs and accumulators, multiplication or division can be accomplished by doing a series of add and/or subtract operations. The most straightforward multiply algorithm operates by shifting the multiplicand one bit at a time and either adding or not adding the result to an accumulator depending on the status of a corresponding bit from the shifted multiplier. Numerous algorithms have been developed to reduce the number of necessary shifts so that an N-bit machine does not necessarily require N add operations. However, the multiply is still accomplished

through repetitive adds. The inconvenience of this type of repetitive opera-
tion can be circumvented by the addition of special hardware dedicated to
multiplication or division. Usually with the addition of this special hardware,
the multiply or divide can be accomplished in the same time as an add or
subtract operation. Of course this convenience and speed is gained at the
cost of the additional hardware necessary to implement the multiplication.
Although presently the multiply/divide feature can be had only by the addition
of hardware, the 16-bit microprocessor being developed by Texas Instruments
(reference 2.2-5) has multiply and divide instructions as a part of its instruc-
tion set. In summary:

1. Double precision arithmetic is time consuming and a microproc-
   essor should be selected with a proper word length to avoid
   double precision words most of the time.

2. Multiply and divide operations require additional hardware or
   additional time.

Figure 2.2-3 is a block diagram of the recently announced AM 2901.

The circuit is a four-bit slice cascadable to any number of bits.
Therefore, all data paths within the circuit are four bits wide. The two key
elements in the block diagram are the 16-word by 4-bit 2-port RAM and the
high-speed ALU.

Data in any of the 16 words of the Random Access Memory (RAM) can
be read from the A-port of the RAM as controlled by the 4-bit A address
field input. Likewise, data in any of the 16 words of the RAM as defined by
the B address field input can be simultaneously read from the B-port of the
RAM.

The high-speed Arithmetic Logic Unit (ALU) can perform three
binary arithmetic and five logic operations on the two 4-bit input words R
and S. The R input field is driven from a 2-input multiplexer, while the S
input field is driven from a 3-input multiplexer. Both multiplexers also have
an inhibit capability; that is, no data is passed. This is equivalent to a "zero"
source operand.

The ALU R-input multiplexer has the RAM A-port and the direct data
inputs (D) connected as inputs. Likewise, the ALU S-input multiplexer has
the RAM A-port, the RAM B-port and the Q register connected as inputs.

Figure 2.2-3. Block diagram of AM2901
Microprocessor.

The ALU itself is a high-speed arithmetic/logic operator capable of performing three binary arithmetic and five logic functions. The 3 microinstruction inputs are used to select the one of eight ALU functions.

The ALU data output is routed to several destinations. It can be a data output of the device and it can also be stored in the RAM or the Q register. Eight possible combinations of ALU destination functions are available.

Also included is a microinstruction decode, which on a clock by clock basis, determines the operations of the ALU and the selected data paths within the microprocessor.

## 2.2.3 Present Technology

To enable comparison between microprocessors, standard evaluation criteria were developed. For the purpose of this document, the following characteristics are considered of prime importance for the APSP application:

1. Add time
2. Multiply and divide capability
3. Power dissipation

Additional considerations are:

1. Possibility of nuclear hardening
2. Cost

Before these items are evaluated however, some pitfalls in comparing available data should be discussed. For less complicated ICs, specification sheets contain data that is fairly well standardized from one vendor to another. For microprocessors however, this is not the case; different vendors use different parameters to measure microprocessor capabilities. This lack of standarization makes the selection of a best microprocessor, on the basis of application, difficult.

For example, microprocessor computing speed may be given as the basic cycle time or clock rate. Although this is a valid parameter when evaluating microinstruction (firmware) cycle time, it is misleading if applied to the simple-to-use assembly type instructions available in most microprocessors. Most complicated instructions, like the Add to Memory instruction mentioned earlier, will require many of the basic cycles or clock periods for execution. For example, the INTEL 8080, "instruction cycle" time is given as two microseconds(Reference 2.2-5). However, an 8080 "minor cycle time" is specified as greater than or equal to 500 nanoseconds. This implies that the number of cycle times for most instructions would be four. Examination of the instruction set shows that some instructions require fewer than four minor cycle times while others require more. It can be assumed that the two microsecond cycle time is either:

1. an average execution time for a particular test program, or
2. an average execution time for the entire instruction set.

2-13

Thus it is necessary to evaluate a microprocessor with a view toward its ultimate application. One technique is to construct a "bench mark" program typical of the device's application, and use this program to evaluate microprocessors of interest (reference 2.2-4).

Realizing this caution it is now possible to examine parameters of particular devices. Figure 2.2-4 was reproduced from Reference 2.2-2 dated 15 April 1975, and compares 24 characteristics of 22 different microprocessors. In order to condense the data given by this figure, Figure 2.2-5 summarizes, by technology type, the Memory to Register add times for the various microprocessors listed. As indicated, all add times are in the microsecond range except for the bipolar Intel 3000 series at 300 nanoseconds (register to register). Figure 2.2-6 depicts the power dissipation of devices of various technologies based upon Mid-1975 commercial production. As the figure shows, each technology occupies a fairly specific power/delay characteristic.

Multiplication and division must be implemented with adds, subtracts and shifts, and can be made faster by additional hardware. The soon to be released TI TMS9900 (not shown on the figures) can do 16-bit multiplication in approximately 17 microseconds, without additional hardware (reference 2.2-5).

The add times given are for the word lengths of the particular machine. Again the reader is cautioned about the difficulties of handling double precision words. Thus, although a particular microprocessor may be able to do an 8-bit add in 2 microseconds, in no way should this imply that a 16-bit add would require 4 microseconds. As a rule the time to add the double precision word is more than double that of a single precision word. The added number of instructions might be as many as 10 or 15, to handle the conditions that might occur, such as overflow and carry.

In summary:

1. An N-bit microprocessor can do an N-bit add to memory in about 2 microseconds with current commerical technology.

2. Double precision arithmetic or multiplication can be executed on any microprocessor but at the expense of some hardware or a lot of time.

# MICROPROCESSORS

| Manufacturer | AMI | Fairchild | Gen Instruments | Intel | Intel | Motorola | Intel |
|---|---|---|---|---|---|---|---|
| **Model Highlights** | | | | | | | |
| Model number | S6800 | F8 | CP1600 | 3000 Series | MCS 4004 | M6800 | 4040 |
| 1st shipment | new product | new product | new product | 7/74 | 3/71 | 7/74 | 9/74 |
| Chip technology | NMOS | NMOS | NMOS | bipolar | PMOS | NMOS | PMOS |
| Chips in cpu/pins per chip | 1 chip/40 pins | 2 chips/40 pins | 1 chip/40 pins | 1 chip/28 pins | 1 chip/16 pins | 1 chip/40 pins | 1 chip/24 pins |
| Add time (reg-to-reg) | 2μsec (8 bits) | 2μsec (8 bits) | 2.4μsec (16 bits) | 300nsec (16 bits) | 10.8μsec (4 bits) | not released | 8μsec (4 bits) |
| Microprogrammed | ye- | yes | yes | user microprogram | no | yes | no |
| **Architecture** | 8 bit parallel | 8 bit parallel | 16 bit parallel | 2 bit slice, parallel | 4 bit parallel | 8 bit parallel | 4 bit parallel |
| ALU/logic share chip | yes | yes | yes | no | yes | yes | yes |
| Clock frequency/phases | 1MHz/2-phase | 2MHz/2-phase | 5MHz/2-phase | 8MHz/1-phase | 750KHz/2-phase | 1MHz/2-phase | 1MHz/2-phase |
| Number of instructions | 72 (8, 16, 24bits) | 101 (8 bits) | 68 (16 bits) | variable | 46 (8 bits) | 72 (8 bits) | 60 (8 bits) |
| Reg. load time for instruc. | 2μsec (8 bits) | 2μsec (8 bits) | 2.4μsec (16 bits) | 150nsec (16 bits) | 10.8μsec (8 bits) | 2μsec | 8μsec (8 bits) |
| Reg. to memory add time | 2μsec (8 bits) | 5μsec (8 bits) | 3.2μsec (16 bits) | variable | 10.8μsec (4 bits) | 2μsec | 8μsec (4 bits) |
| **Input/Output** | | | | | | | |
| Data path width | 8 bits | 8 bits | 16 bits | variable | 4 bits | 16 bits | 4 bits |
| Interrupts | yes | chain | nested | vectored priority | no | yes | vectored |
| Peripheral interfaces | through adaptor | – | none | – | display | through adaptors | display |
| **Software** | | unbundled | | unbundled | resident bundled | unbundled | resident bundled |
| Resident assembler | – | ✓ | ✓ | – | ✓ | – | ✓ |
| Cross assembler | – | ✓ | ✓ | cross microassemb. | ✓ | ✓ | ✓ |
| Monitor | – | – | ✓ | – | – | – | ✓ |
| Languages | – | – | – | – | – | – | – |
| Instruction simulator | – | ✓ | ✓ | – | ✓ | ✓ | ✓ |
| Prototyping System | to be announced | yes | no | no | yes | yes | yes |
| **Pricing** | | | | | | | |
| Chips/chip sets (lots of 100) | not released | not released | not released | not released | less than S99 | not released | less than S99 |

| Manufacturer | Intel | Monolithic | Mostek | National Semiconductor |
|---|---|---|---|---|
| **Model Highlights** | | | | |
| Model number | 8080 | 6701 | 5065 | PACE |
| 1st shipment | 12/73 | 7/74 | 1st Q 1974 | 1975 |
| Chip technology | NMOS | bipolar | PMOS | PMOS |
| Chips in cpu/pins per chip | 1 chip/40 pins | 24 chips/40 pins | 1 chip/40 pins | 1/40 pins |
| Add time (reg-to-reg) | 2μsec (8 bits) | 900nsec (16 bits) | 10μsec (8 bits) | 2μsec (16 bits) |
| Microprogrammed | no | user microprogram | no | no |
| **Architecture** | 8 bit parallel | 4 bit slice, parallel | 8 bit parallel | 8 bit parallel |
| ALU/logic share chip | yes | no | yes | yes |
| Clock frequency/phases | 2MHz/2-phase | 5MHz/1-phase | 1.4MHz/3-phase | 1MHz/2-phase |
| Number of instructions | 78 (8 bits) | 22 (16 bits) | 51 (8/16 bits) | 72 (8 bits) |
| Reg. load time for instruc. | 2.5μsec (8 bits) | 1.2μsec (16 bits) | 8.5μsec (8 bits) | 2μsec (8 bits) |
| Reg. to memory add time | 3.5μsec (8 bits) | 1.2μsec (16 bits) | 10μsec (8 bits) | 2μsec (8 bits) |
| **Input/Output** | | | | |
| Data path width | 8 bits | 16 bits | 8 bits | 16 bits |
| Interrupts | vectored | 1 level, priority opt | priority | yes |
| Peripheral interfaces | I/O port | none | none | through adaptors |
| **Software** | resident bundled | bundled | unbundled | unbundled |
| Resident assembler | ✓ | ✓ | ✓ | ✓ |
| Cross assembler | ✓ | ✓ | ✓ | ✓ |
| Monitor | ✓ | – | – | – |
| Languages | PL/M | – | – | – |
| Instruction simulator | ✓ | – | – | ✓ |
| Prototyping System | yes | yes | no | yes |
| **Pricing** | | | | |
| Chips/chip sets (lots of 100) | less than $400 | $95/$600 | S58 | not released |

| Manufacturer | National Semiconductor | National Semiconductor | National Semiconductor | RCA | Rockwell |
|---|---|---|---|---|---|
| **Model Highlights** | | | | | |
| Model number | IMP-4 | IMP-8A/500D | IMP-16 | COSMAC | PPS-4 |
| 1st shipment | 1974 | 3/74 | 1973 | new product | 1973 150K shipped |
| Chip technology | PMOS | PMOS | PMOS | CMOS | PMOS |
| Chips in cpu/pins per chip | 2/24 | 3 chips/24 pins | 5/24 pins | 2 chips/28, 40 pins | 1 chip/42 pins |
| Add time (reg-to-reg) | 12μsec (4 bits) | 4.2μsec (8 bits) | 4.7μsec (16 bits) | 18μsec (8 bits) | 4μsec (8 bits) |

**Table (continued from previous page)**

| | | | | | |
|---|---|---|---|---|---|
| **Peripheral interfaces** | I/O port | I/O port | none | none | through adaptors |
| **Software** | resident bundled | resident bundled | bundled | unbundled | unbundled |
| Resident assembler | √ | √ | √ | √ | √ |
| Cross assembler | √ | √ | √ | √ | √ |
| Monitor | | | | | |
| Languages | PL/M | PL/M | | | |
| Instruction simulator | √ | √ | | | √ |
| **Prototyping System** | yes | yes | yes | no | yes |
| **Pricing** Chips/chip sets (lots of 100) | less than $250 | less than $400 | $95/$600 | $58 | not released |

**Microprocessors (table)**

| Manufacturer | National Semiconductor | National Semiconductor | National Semiconductor | National Semiconductor | RCA | Rockwell |
|---|---|---|---|---|---|---|
| Model Highlights Model number | IMP-4 | IMP-8A/500D | IMP-16 | PACE | COSMAC | PPS-4 |
| 1st shipment | 1974 | 3/74 | 1973 | 1975 | new product | 1973 150K shipped |
| Chip technology | PMOS | PMOS | PMOS | PMOS | CMOS | PMOS |
| Chips in cpu/pins per chip | 2/24 | 3 chips/24 pins | 5/24 pins | 1/40 pins | 2 chips/28, 40 pins | 1 chip/42 pins |
| Add time (reg-to-reg) | 12μsec (4 bits) | 4.2μsec (8 bits) | 4.7μsec (16 bits) | 2μsec (16 bits) | 18μsec (8 bits) | 4μsec (8 bits) |
| Microprogrammed | user microprogram | user microprogram | user microprogram | no | no | yes |
| Architecture | 4 bit slice | 4 bit slice, parallel | 4 bit slice | 16 bit | 8 bit parallel | 4 bit parallel |
| ALU/logic share chip | no | no | no | yes | yes | yes |
| Clock frequency/phases | 500KHz/4-phase | 715KHz/4-phase | 715KHz/4-phase | 2 MHZ/2-phase | 2.67MHz/1-phase | 200KHz/4-phase |
| Number of instructions | 42 | 38 (8 bits) | 43 or 60 | 45 | 59 (8 bits) | 50 (8 bits) |
| Reg. load time for instruc. | 20μsec (4 bits) | 11.2μsec (8 bits) | 10.2μsec (16 bits) | 2μsec (16 bits) | 6μsec (8 bits) | 4μsec (8 bits) |
| Reg. to memory add time | 20μsec (4 bits) | 11.2μsec (8 bits) | 7.7μsec (16 bits) | 2μsec (16 bits) | 6μsec (8 bits) | 4μsec (8 bits) |
| Input/Output | | | | | | |
| Data path width | 4 bits | 8 bits | 16 bits | 16 bits | 8 bits | 4 bits |
| Interrupts | yes | yes | yes | yes | maskable | none |
| Peripheral interfaces | tty, display | tty, display | tty, etc. | tty, etc. | none | display, tty, gp |
| Software | bundled | bundled | bundled | bundled | unbundled | unbundled |
| Resident assembler | √ | √ | √ | √ | √ | √ |
| Cross assembler | √ | √ | √ | √ | √ | √ |
| Monitor | √ | √ | | | | |
| Languages | | | | | | |
| Instruction simulator | | | | | √ | √ |
| Prototyping System | yes | yes | yes | yes | yes | yes |
| Pricing Chips/chip sets (lots of 100) | <100 | $181 | <200 | <200 | not released | approx. $5 |

**Microprocessors (table)**

| Manufacturer | Rockwell | Signetics | Texas Instruments | Texas Instruments | Western Digital |
|---|---|---|---|---|---|
| Model Highlights Model number | PPS-8 | 2650 "PIP" | TMS 1000 | SBP 0400 | CP 1611/1621/1631 |
| 1st shipment | 10/74 new product | new product | 3rd Qtr 1974 | Feb. 1975 | new product |
| Chip technology | PMOS | NMOS | PMOS | $I^2L$ | NMOS |
| Chips in cpu/pins per chip | 1 chip/42 pins | 1 chip/40 pins | 1 chip/40 pins | 1 chip/40 pins | 3 chips/40 pins |
| Add time (reg-to-reg) | 4μsec (8 bits) | 4.8μsec (8 bits) | 12μsec (4 bits) | 2μsec (16 bits) | 300nsec (8 bits) |
| Microprogrammed | yes | no | 30 of 43 instructions | user microprogram | user microprogram |
| Architecture | 8 bit parallel | 8 bit parallel | 4 bit parallel | 4 bit parallel/slice | 8/16 bit parallel |
| ALU/logic share chip | yes | yes | yes | yes | no |
| Clock frequency/phases | 250KHz/4-phase | 1.25MHz/1-phase | internal (300KHz) | 10HZ/1MHz 1-phase | 3.3MHz/4-phase |
| Number of instructions | 100 (8, 16, 24 bits) | 72 (8, 16, 24 bits) | 43 (8 bits) | 43 (8 bits) | over 80 (16 bits) |
| Reg. load time for instruc. | 5μsec (8 bits) | 4.8μsec (8 bits) | 12μsec (8 bits) | 1μsec | 900nsec (8 bits) |
| Reg. to memory add time | 5μsec (8 bits) | 4.8μsec (8 bits) | 12μsec (4 bits) | 1μsec | 1.2μsec (8 bits) |
| Input/Output | | | | | |
| Data path width | 8 bits | 8 bits | 4 bits | variable | 8/16 bits |
| Interrupts | 3x16 daisy chain | 1-level vectored | no | yes | priority, 4 level |
| Peripheral interfaces | display, tty, gp | none | none | none | none |
| Software | unbundled | unbundled | unbundled | unbundled | |
| Resident assembler | √ | √ | √ | | |
| Cross assembler | √ | √ | √ | | |
| Monitor | | | | | |
| Languages | | | | | |
| Instruction simulator | √ | | | | |
| Prototyping System | yes | no | yes | no | to be announced |
| Pricing Chips/chip sets (lots of 100) | approx. $47 | $200 | $10 | ≈$25/slice | not released |

Figure 2.2-4. Microprocessors.

```
                                    FASTEST    -    2  μSEC
                    NMOS
                                    SLOWEST    -    4.8 μSEC
                                    FASTEST    -    5  μSEC
        8-BIT       PMOS
                                    SLOWEST    -    32 μSEC
                    CMOS            -                6  μSEC

                    NMOS            -                2.3 μSEC
                                    FASTEST    -    2  μSEC
        16-BIT      PMOS
                                    SLOWEST    -    10.2 μSEC
                    BIPOLAR         -                1.2 μSEC

                    BIPOLAR         -                300 NSEC (16-BIT REGISTER/
                                                              REGISTER)
    2 OR 4-BIT
    SLICE           I²L             -                2  μSEC (16-BIT REGISTER/
                                                             REGISTER)
```

ALL TIMES ARE REGISTER/MEMORY ADD TIMES UNLESS NOTED

Figure 2.2-5. Microprocessor memory to register Add time comparison by technology.
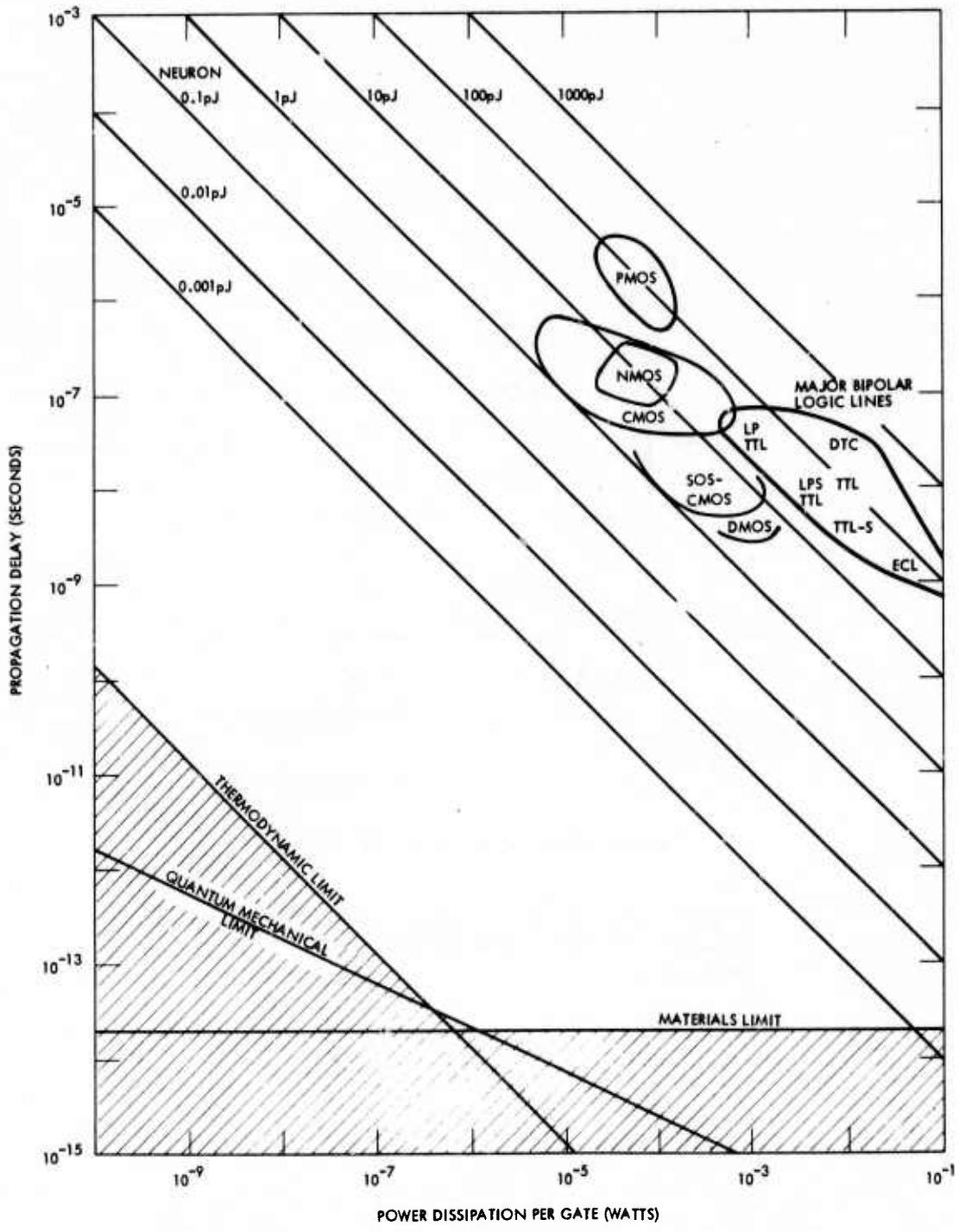
Figure 2.2-6. Speed power product.

3. PMOS is currently being displaced by NMOS and CMOS due to the latter devices reduced power requirement, increased speed and TTL compatibility.

4. 16-bit microprocessors are currently available that operate at about the same speed as 8-bit microprocessors. This implies that it would be unnecessary for an 8-bit device to be applied in a system requiring 16-bit operands.

## 2.2.4 Development Trends and Projections

Trends of execution speed, chip density, and power can be analyzed by looking at them in the past and projecting into the future. It should be noted that since microprocessors have been around less than five years, any observable trends have developed only fairly recently.

Figure 2.2-7 lists various technologies from 1965 to 1974, and projects the technologies of 1980. Each technology is ranked in order of its anticipated predominance, so that while PMOS is the prime commercial technology in 1974, it will nearly disappear by 1980. By 1980 the strong commercial technologies should be:

1. $I^2L$

2. CMOS

3. DMOS

These three technologies will be used in both military and commercial applications. An evaluation of each technology based on the parameters shown in Figure 2.2-9 yields the ranking in Figure 2.2-8. Using power and speed criterion only, it would appear that CMOS/SOS would best serve military needs (see Figure 2.2-8). Note that CMOS/SOS is expected to be the most available technology in 1980. Note that, for 1980, the technology ranked highest on Figure 2.2-8, (CMOS/SOS), is at the bottom of the list of technologies shown in Figure 2.2-7. The reason that Figure 2.2-7 favors what the commercial world will develop due to parameters such as yield and density, while Figure 2.2-8 is based upon the military selection parameters shown in Figure 2.2-9. The most recent advances in DMOS technology, not included in this data, typify the fluidity in the semiconductor field and

| | 1965 | 1974 | 1980 |
|---|---|---|---|
| TECHNOLOGIES | P-MOS | P-MOS | N-MOS, DMOS |
| | | N-MOS. | C-MOS |
| | | C-MOS | ECL |
| | | TTL | $I^2L$ |
| | | ECL | EFL (Triple Diffused) |
| | | EFL (Triple Diffused) | MNOS |
| | | MNOS | CCD |
| | | | C-MOS/SOS |
| Chip Size (MAX) MILS | 100 | 300 | 500 |
| Device Density $(MIL)^2$/gate | 20 | 1 | 0.7-0.4 |
| Speed Power Product (pJoule) | 100 | 5-10 | 0.01-1 |
| Clock Rates (MHz) | 20 | 300 | 2000 |
| Weight Per Gate (lbs) | $5 \times 10^{-4}$ | $1 \times 10^{-5}$ | $1 \times 10^{-7}$ |

Figure 2.2-7. Large scale integration technology, listed in order of projected share of the commercial market as a function of time.

Figure 2.2-8. Comparison of semiconductor technologies (excluding DMOS) for various design parameters.

| TECHNOLOGY | 1975 Availability | 1976 Availability | 1977 Availability | 1978 Availability | 1979 Availability | 1980 Availability | Size (Bits Per Chip) | Power-Delay | Speed (Processing) | Cooling (Min. Req.) | Systems Level EMP | Signal Conducted EMP | Nuclear Transient | Neutron Dose | Ionization – Total Dose | COMSEC/TEMPEST | Jamming (ECM/ECCM) | RINT | Reliability/Gate | Maintainability | EMC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TTL | 3 | 6 | 10 | 9 | 9 | 9 | 10 | 10 | 2 | 9 | 10 | 1 | 8 | 1 | 2 | 9 | 8 | 8 | 1 | 9 | 9 |
| ECL | 4 | 5 | 9 | 8 | 8 | 8 | 8 | 7 | 1 | 10 | 5 | 2 | 9 | 2 | 1 | 8 | 9 | 9 | 9 | 10 | 10 |
| EFL (Triple Diffused) | 5 | 4 | 6 | 7 | 7 | 7 | 4 | 4 | 3 | 8 | 4 | 3 | 7 | 3 | 3 | 4 | 2 | 7 | 3 | 4 | 5 |
| $I^2L$ | 10 | 10 | 7 | 2 | 2 | 3 | 2 | 2 | 5 | 1 | 1 | 4 | 7 | 4 | 4 | 1 | 3 | 2 | 2 | 2 | 2 |
| P-MOS | 2 | 2 | 5 | 2 | 2 | 6 | 9 | 9 | 9 | 5 | 9 | 6 | 2 | 10 | 5 | 10 | 10 | 10 | 8 | 8 | 8 |
| N-MOS | 1 | 1 | 1 | 5 | 6 | 6 | 6 | 8 | 6 | 7 | 8 | 7 | 3 | 8 | 7 | 7 | 6 | 5 | 7 | 6 | 7 |
| C-MOS | 6 | 3 | 2 | 1 | 1 | 4 | 7 | 5 | 8 | 4 | 7 | 9 | 4 | 9 | 8 | 5 | 5 | 4 | 5 | 5 | 4 |
| C-MOS/SOS | 7 | 7 | 3 | 3 | 3 | 1 | 3 | 3 | 4 | 3 | 3 | 10 | 1 | 9 | 3 | 3 | 4 | 3 | 4 | 3 | 3 |
| CCD | 9 | 9 | 4 | 4 | 4 | 2 | 1 | 1 | 10 | 2 | 6 | 5 | 6 | 5 | 10 | 2 | 1 | 1 | 1 | 1 | 1 |
| MNOS | 8 | 8 | 8 | 6 | 5 | 5 | 5 | 6 | 7 | 6 | 2 | 8 | 5 | 6 | 6 | 6 | 7 | 6 | 6 | 7 | 6 |

Legend: 1 = Most important  
10 = Least important

2-21

Nuclear-Effects Survivability

      Systems Level EMP
      Signal Conducted EMP
      Nuclear Transient (Neutron, X and Gamma Rays)
      Neutron Dose
√ Ionization Total Dose (Electrons, X and Gamma Rays)

Electromagnetic Vulnerability

      Communications Security (COMSEC)
      TEMPEST
      Jamming (ECM, ECCM)
      Radiation Intelligence (RINT)

Viability

√√√ Reliability
 √ Availability
   Maintainability
 √ Electromagnetic Compatibility (EMC)

Physical Characteristics

√√ Weight
√√√ Power Consumption
 √ Cooling Requirements
 √ Speeds

Impact Upon
APSP Application

√ Minor
√√ Moderate
√√√ Large

Figure 2.2-9. Characteristics for military uses affecting
selection of LSI technologies.

indicate the risk associated with predictions. DMOS may very well be the dominant LSI technology by 1978 or 79 (see sections 2.4, 2.7), if its considerable promise is realized.

Figure 2.2-10 gives an idea of the trend in circuit densities for integrated circuits. According to this figure, memory bit density in 1978 is expected to be about four times the 1974 value. For CPUs, the density is expected to double in the same period. Basically this implies that the physical area and thus the number of chips required to implement future systems will decrease. This also implies that the amount of hardware that can be cost effectively replaced by microprocessors will increase in the future. Figure 2.2-11 indicates that commercial microprocessor cycle times will drop to about 25-60 nsec by 1982.

As mentioned earlier, evaluation of a microprocessor from its specification sheet may be imprecise due to lack of standardization; benchmark programs should be written in order to test different devices for a specific task. A set of benchmark programs (Figure 2.2-14) was written for a typical application and the results of running these programs on several microprocessors are shown in Figures 2.2-12 and 2.2-13. Although the number of instructions and amount of time varies widely for the different devices, it should be noted that other important selection parameters such as power or physical size of the devices are not treated. The purpose here is not to demonstrate one device's superiority over another, but rather to:

1. Display the data array that is obtained when a benchmark program is run on various microprocessors.

2. Demonstrate that since microprocessors were first introduced tremendous changes have occurred in their characteristics, and that the next few years are expected to yield equally significant changes.

| | Density per Chip | | |
| --- | --- | --- | --- |
| | 1970 | 1974 | 1978 |
| Equivalent Gates per CPU | | | |
| Bipolar | 150 | 1,000 | 2,000 |
| MOS | 1,000 | 5,000 | 10,000 |
| $I^2L$ or $C^3L$ | - | - | 5-10,000 |
| Memory Bits (RAM) | | | |
| Bipolar | 256 | 1,024 | 4,000 |
| MOS | 1,024 | 4,096 | 16,000 |
| $I^2L$ or $C^3L$ | - | - | 8,000 |

| | Price (¢) | | |
| --- | --- | --- | --- |
| | 1970 | 1974 | 1978 |
| Per Logic Gate | | | |
| Bipolar | 10 | 3 | 1 |
| MOS | 1 | 0.3 | 0.1 |
| $I^2L$ or $C^3L$ | - | - | 0.2-0.5 |
| Per Memory Bit | | | |
| Bipolar | 6 | 0.6-1.0 | 0.3 |
| MOS | 2 | 0.3-0.5 | 0.1 |
| $I^2L$ or $C^3L$ | - | - | 0.2 |
| CCD | - | - | 0.01-0.05 |

Figure 2.2-10.  Integrated circuit density and price trends.
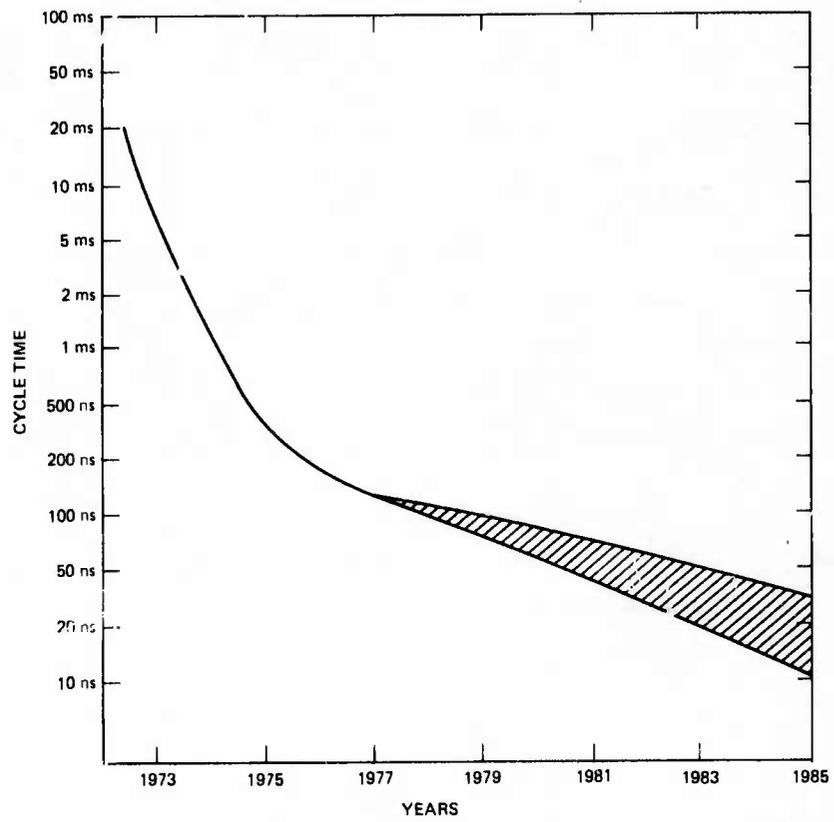
Figure 2.2-11.   Projected cycle times for 8 bit microprocessors.

| | Elec-tronic Arrays | Intel Corp 8008 | Intel Corp 8080 | Mostek Corp 5065 | Motorola Semi M6800 | Nat'l Semi IMP-8 | RCA Corp Cosmac | Rockwell Intl. PPS-8 | Scientific Micro Sys | Signetics Corp 2650 | WDC Marco Level | WDC Micro. Level |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1. Block Data Movement** | | | | | | | | | | | | |
| A. Bytes of Instruction | 13 | 34 | 18 | 40 | 15 | 40 | 22 | 43 | 60 | 10 | 10 | 34 |
| B. Set Up Time | 4 | 170 | 13.5 | 75 | 8 | 75 | 60 | 144 | 2.1 | 4.8 | 7.8 | 3.3 |
| C. Move Time/Char | 24 | 184 | 23.5 | 89 | 20 | 91 | 36 | 16 | 6.9 | 26.4 | 6.0 | 3.0 |
| **2. Servicing An Interrupt** | | | | | | | | | | | | |
| A. Bytes of Instruction | 20 | 51 | 8 | - | 2 | 26 | 15 | 12 | 68 | 13 | 14 | 42 |
| B. Service Time | 44 | 1076 | 42 | - | 22 | 113 | 72 | 38 | 10.2 | 52.6 | 19.5 | 9.0 |
| **3. Add of "N" Decimal Digits and Store** | | | | | | | | | | | | |
| A. Bytes of Instruction | 8 | 70 | 22 | 30 | 17 | 71 | 79 | 16 | 68 | 17 | 16 | 46 |
| B. Set Up Time | 10 | 168 | 17.5 | 7.1 | 7 | 15 | 114 | 40 | 1.2 | 7.2 | 10.2 | 4.2 |
| C. Add Time/Byte | 9 | 1008 | 29 | 147 | 28 | 234 | 216 | 24 | 9.0 | 50.4 | 11.1 | 5.1 |
| **4. Search for Character String** | | | | | | | | | | | | |
| A. Bytes of Instruction | 23 | 30 | 21 | 32 | 22 | 22 | 23 | 20 | 62 | 16 | 24 | 42 |
| B. Set Up Time | 10 | 32 | 5 | 48.6 | 3 | 20 | 36 | 40 | 2.4 | 4.8 | 8.7 | 4.2 |
| C. Search Time/Char | 35 | 360 | 29.2 | 87 | 29 | 40 | 48 | 32 | 4.2 | 40 | 15 | 4.5 |
| **5. Monitors 8 Data Channels** | | | | | | | | | | | | |
| A. Bytes of Instruction | 39 | 43 | 34 | 40 | 30 | 34 | 28 | 36 | 58 | 30 | 8 | 20 |
| B. Set Up Time | 12 | 64 | 8.5 | 42.9 | - | 10 | 24 | 28 | 0.9 | - | - | - |
| C. Through Put | 77 | 704 | 71.1 | 148.7 | 61 | 146 | 84 | 84 | 7.2 | 98.4 | 9.3 | 3.3 |
| **TOTAL** | | | | | | | | | | | | |
| A. Bytes of Instructions | 103 | 228 | 103 | 142 | 86 | 194 | 167 | 127 | 316 | 86 | 72 | 184 |
| B. Total Program Execution Time (μSec) | 225 | 3766 | 249.3 | 645.3 | 178 | 744 | 690 | 496 | 44.1 | 384.8 | 8.6 | 36.6 |

NOTES: Mostek has 3 sets of Reg and has no Interrupt housekeeping.
Signetics, Motorola and WDC have interrupt Polling Schemes to monitor activities.

Figure 2.2-12.  Microprocessor performance comparison.

2-26
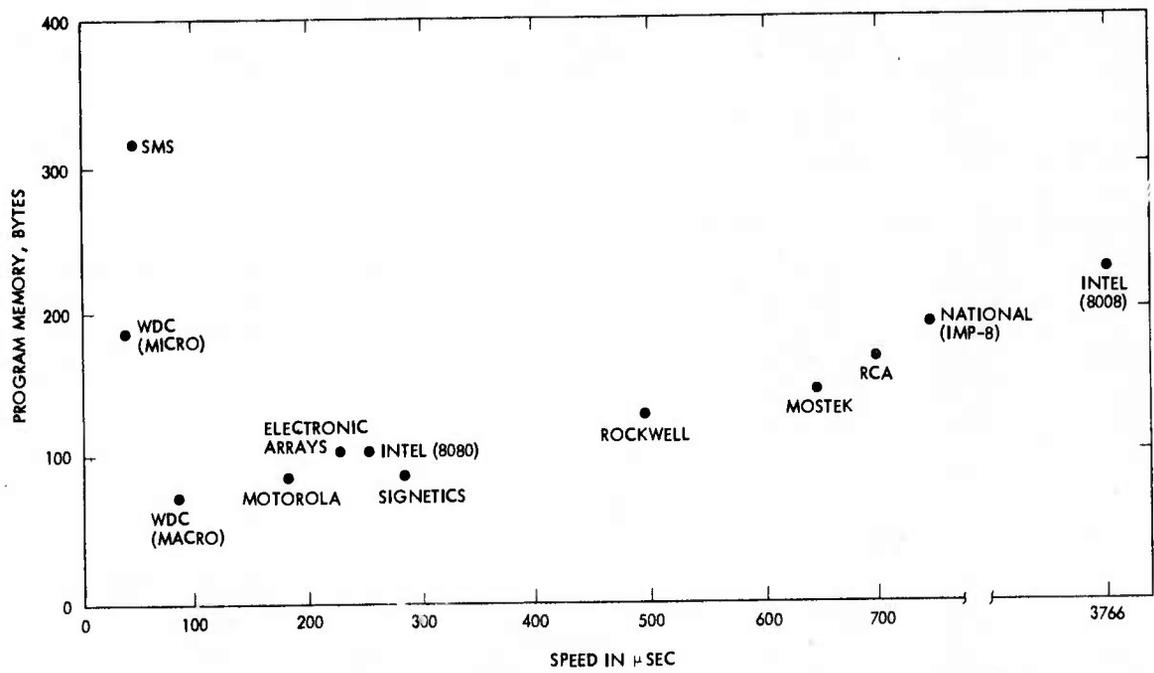
Figure 2.2-13. Program memory versus speed for various microprocessors.

A.   Movement of Blocks of Data

| | | | | | Program Bytes | Set Up Time | Move Time / Character |
|---|---|---|---|---|---|---|---|
| SET UP | { MOV | # Base 1, $R_1$ | | | | | |
| | MOV | # Base 2, $R_2$ | | | | | |
| | MOV | # Char, $R_3$ | | Micro-Level | 34 | 3.3μs | 3.0μs |
| LOOP 1 | MOVB | $(R_1)+$, $(R_2)+$ | | Macro-Level | 10 | 7.8μs | 6.0μs |
| | SOB | $R_3$, Loop 1 | | | | | |
| | EXIT | | | | | | |

B.   Servicing Interrupt

| | | | Program Bytes | Service Time |
|---|---|---|---|---|
| MOV | #INT LOC, $R_1$ | | | |
| MOV± | PC, $(R_1)+$ | | | |
| MOV | ACC, $(R_1)+$ | | | |
| MOV | FLAGS, $(R_1)+$ | Micro-Level | 42 | 9.0μs |
| | | Macro-Level | 14 | 19.5μs |
| - - - - - - - - - - - - - - - - - - - - - | | | | |
| - - - - - - - - - - - - - - - - - - - - - | | | | |
| MOV | $-(R_1)$, FLAGS | | | |
| MOV | $-(R_1)$, ACC | | | |
| MOV | $(R_1)$, PC | | | |

C.   Addition of "N" Decimal Digits and Store

| | | | | Program Bytes | Set Up Time | Add Time / Byte |
|---|---|---|---|---|---|---|
| SET UP | { MOV | # Base 1, $R_1$ | | | | |
| | MOV | # Base 2, $R_2$ | | | | |
| | MOV | #1010, $R_3$ | | | | |
| | MOV | #N, $R_4$ | Micro-Level | 46 | 4.2μs | 5.1μs |
| LOOP 1 | MOV | $(R_1)+$, $R_4$ | Macro-Level | 16 | 10.2μs | 11.1μs |
| *ADD | $(R_2)$, $R_4$ | | | | | |
| MOV | $R_4$, $(R_2)+$ | | | | | |
| SOB | $R_3$, LOOP 1 | | | | | |
| | | | | | | |
| EXIT | | | | | | |

D.   Search for a Character String

| | | | | Program Bytes | Set Up Time | Search Time / Character |
|---|---|---|---|---|---|---|
| | MOV | # Mask, $R_1$ | | | | |
| | MOV | # Char, $R_2$ | | | | |
| | MOV | #0, $R_3$ | | | | |
| LOOP 1 | CMPB | #255, $R_3$ | Micro-Level | 42 | 4.2μs | 4.5μs |
| | BEQ | EXIT | Macro-Level | 24 | 8.7μs | 15.0μs |
| | MOV | $(R_3)+$, $R_4$ | | | | |
| | CMPB | $R_1$, $R_4$ | | | | |
| | BEQ | LOOP 2 | | | | |
| | MOV | # Char, $R_2$ | | | | |
| | JMP | LOOP 1 | | | | |
| LOOP 2 | SOB | $R_2$, EXIT | | | | |
| | JMP | LOOP 1 | | | | |
| | EXIT | | | | | |

E.   Monitor 8 Data Channels

| | | | Program Bytes | Through Put / Character |
|---|---|---|---|---|
| **MOV | INT, $R_1$ | | | |
| MOV | $(R_1)$, $R_2$ | | | |
| INCB | $R_2$ | Micro-Level | 20 | 3.3μs |
| MOV | $R_2$, $(R_1)$ | Macro-Level | 8 | 9.3μs |
| EXIT | | | | |

*Special Macroinstruction composed of AL, ABF, CAD after receiving (R2)
**In response to activity indicated over interrupt line

Figure 2.2-14.   Benchmark programs.

### 2.2.5 Examples of Microcomputer Usage

The following are two recent examples of microprocessor usage in systems design.

The first is the HMC-1820 used as a controller of computer peripherals. Figure 2.2-15 shows the total block diagram of the controller and Table 2.2-1 summarizes the design highlights.

The second example is the Hughes Militarized Microcomputer (MMC) developed as the Central Processor Unit (CPU) for such applications as radar signal processors, and missile guidance computers.

Figure 2.2-16 is the block diagram of the MMC, and Table 2.2-2 contains the major design characteristics.

REFERENCES (2.2)

2.2-1 State-of-the-Art of Hardware and Software for Microprocessors by Arthur D. Little Inc., April 1975.

2.2-2 Microprocessor Technology Forecast through 1978, by Arthur D. Little Inc., April 1975.

2.2-3 Survey of Microprocessor/Microcomputer Applications Executive Summary and Chapter 1, by Arthur D. Little Inc., April 1975.

2.2-4 TMS 9900 Specification Sheet Microprocessor, Texas Instruments Incorporated, October 1975.

2.2-5 INTEL 8080 data book, INTEL Corporation, Santa Clara, CA., 1975.
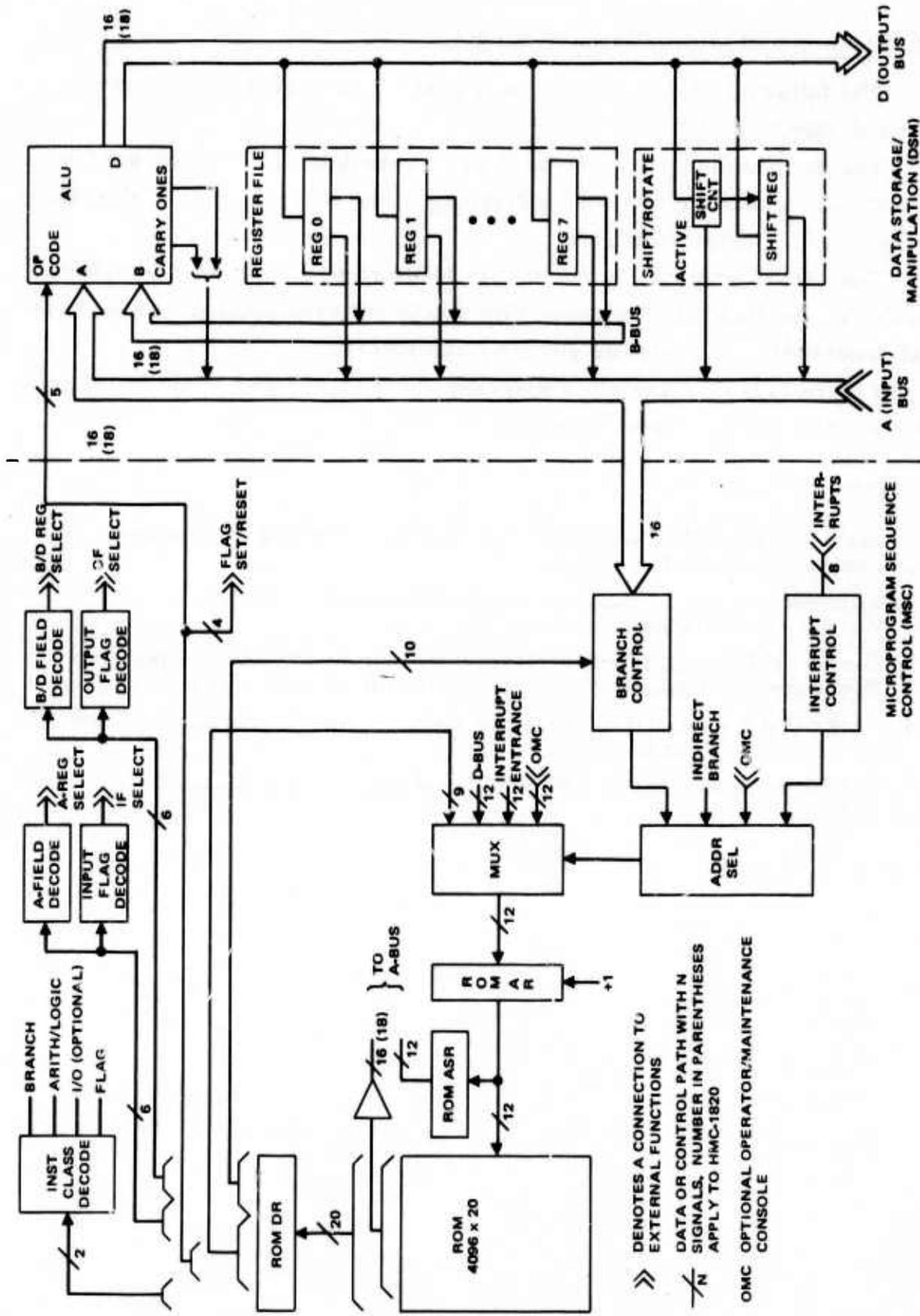
Figure 2.2-15. HMC microcontroller block diagram.

## TABLE 2.2-1. HMC-1820 DESIGN FEATURES

<u>Architecture/Performance</u>

| | |
|---|---|
| Basic Architecture: | General register, microprogrammed |
| Microinstruction Length: | Twenty bits |
| Data Word Length: | Sixteen bits (HMC-1620)/Eighteen bits (HMC-1820) |
| Microinstructions: | Twenty-eight Arithmetic/Logic<br>Twenty-eight Immediate Arithmetic/Logic<br>Four Flag Control<br>Four Shift/Rotate (3 in HMC-1820)<br>Five Conditional Branch<br>Three Unconditional Branch (including indirect and address modified branches)<br>Ten Input/Output (optional) |
| ROM Size: | 512 Words minimum<br>Expandable to 4096 words in 512-word increments |
| Registers: | Eight general purpose<br>One Shift/Rotate |
| Microinstruction Execution Time: | 333 Nanoseconds (250 nsec optional)<br>667 Nanoseconds (500 nsec optional) for immediate operand or if branch taken |
| Interrupts: | Eight priority interrupts, vectored<br>Microprogram controlled interrupt enable<br>One level of interrupt return address storage |

<u>Support</u>

| | |
|---|---|
| Software: | Cross assembler for IBM 360/370 |
| Firmware (option): | Microcontroller Test Program |
| Hardware (option): | Operator console with breakpoint, snapshot and single clock controls to aid in microprogram debugging<br>ROM Simulator for microprogram checkout |

2-31
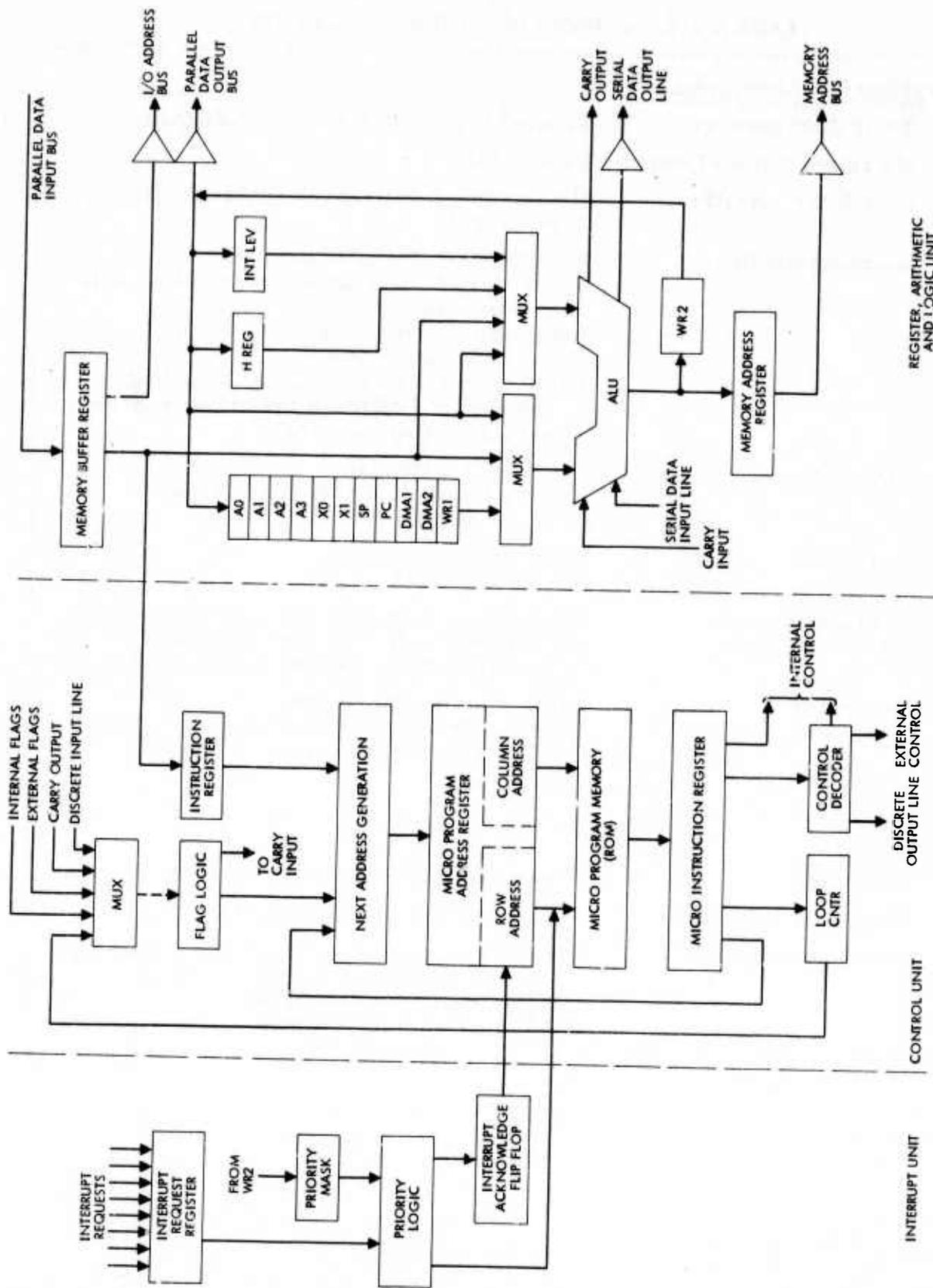
Figure 2.2-16. MMC block diagram.

TABLE 2.2-2. SUMMARY OF MMC PROCESSOR
CHARACTERISTICS

| | |
|---|---|
| Number System | Binary, Fixed Point |
| Data Representation | Fractional 2's Complement |
| Operation | Parallel arithmetic, 16 bits/word |
| Instruction Set | 50 instructions |
| Double Precision Instructions | Add, Shift Left, Shift Right |
| Registers | 8 General Purpose Registers |
| Interrupts | 4 Vectored Priority Interrupts and Power Fail Interrupt |
| Operation Times | Load/Add/Sub 2.2 μsec |
| | Store 2.6 μsec |
| | Multiply 9.0 μsec |
| | Divide 13 μsec |
| | Jump 2.0 μsec (Unconditional) |
| Processing Speed | 400 KOPS (thousand operations per second) |
| Addressing Modes | Direct, Indexed Relative, Relative Indirect, Indirect, Immediate, Register |
| Control Processing | Microprogrammed Control Program Stored in 512 x 32 ROM |

## 2.3 MEMORY TECHNOLOGY

In this section the various key parameters which dictate the appropriate types of memories are discussed and those memory technologies which best fit are enumerated. The particular requirements which are discussed are non-volatile program store, scratch pad, microcode storage (ROM) and large serial memories.

Space borne signal processors have memory requirements which cover a wide spectrum of technologies. If there is to be a general purpose computer with stored program it will require a random access memory (RAM) in which the program resides. In general the contents of this memory are seldom changed, but should be capable of being updated. This requires a non-volatile (i.e., maintains data without power) writable memory. There are presently three candidate technologies: magnetic plated wire, MNOS, and ultra-violet erasable MOS structures. Program storage memory typically will have a radiation hardened specification since it is imperative that the computer program survive (at least critical portions of it) if the general purpose computer is to be useful. Presently 2 mil plated wire appears to be best suited to these above requirements as well as low power and weight. The two semiconductor technologies have the disadvantage of complex erase procedures, and a significant amount of overhead for the write circuitry. The ultra-violet erasable memories presently fatigue after some number of write-erase cycles (~ 100 - 1000).

In contrast to the program store which should be capable of update, the microcontroller which stores the firmware which drives the microprocessor need not be updated. This memory is a read only (ROM), usually less than one thousand words, and of comparatively high speed. The speed of this memory will usually determine the minor cycle time of the processor, thus the memory cycle time must not exceed the register to register data path delays internal to the processor.

Another general class of memories are read/write random access memories used as scratch pad for temporary storage in typical arithmetic computations. These memories should have read access times of about

1 - 3 times the minor cycle time of the processor and in general are not required to be non-volatile. For this class of memories, it is expected that the memory technology will be the same as the logic technology ($I^2L$, DMOS, CMOS/SOS). Table 2.3-1 is a summary chart of basic characteristics of random access memories.

The third class of memories are very large data banks, particularly in the special purpose signal processors where there is a requirement for a significant (16 - 80) bits per pixel. By the nature of the focal plane readout method (pixel serial on an MFPA) the natural memory would be serial.

The memory could be volatile, which implies that during power outage data is lost. The potential size of this memory places highest priority on low power and maximum bit density to reduce weight. The prime candidate technology appears to be CCD memories. (Note that two other memory technologies are not considered viable contenders for the APSP application: (1) magnetic bubble memories, primarily for reasons of speed and bulk, and (2) optical memories for reasons of power and mechanical reliability. Present bubble memories are limited to approximately 100 kHz and have a fundamental materials limit at 1 MHz. The APSP application requires 1.64 MBit capability (Ref. CDRL A006 APSP Architecture Study). The relatively high power requirements of optical memories coupled with reliability concerns caused by their moving elements effectively preclude their use for satellite applications, and restricts their use even in ground-based systems.)

## 2.3.1 CCD Memory Technology

The primary advantages of CCD memory compared to conventional digital shift registers or other digital memory devices are low power dissipation, small element size, and potential low cost per bit. For ground based systems, the power saving and small size are probably of secondary importance, whereas, for space borne equipment, these would be significant factors in large bulk memories.

The serial structure of the CCD does not directly lend itself to random access storage, and the dark current, due to thermally generated carriers, tends to degrade stored data as a function of time. These devices, therefore,

2-35

## TABLE 2.3-1. BASIC CHARACTERISTICS OF MAGNETIC CORE, PLATED WIRE, AND SEMICONDUCTOR MEMORIES, 1975 TECHNOLOGY

| Memory Technology | Storage Element | Non-destructive Readout (NDRO) | Non-volatile | (1) Radiation Tolerance | Speed Range Read or Write Cycle, μs | (2) Power mw/bit | Bit Density/in³ | (3) System Cost (cents per bit) | Power Forms |
|---|---|---|---|---|---|---|---|---|---|
| Magnetic core | Magnetic core | No | Yes, with special circuits | Hard | Large arrays 1 to 2 Small arrays 0.7 to 1 | 0.25 to 0.6 | 1000 to 2000 | 4 to 7 | 3 or 4 |
| Plated wire (5-mil wire) | Magnetic film | Yes | Yes | Hard | 0.5 to 1.0 | 0.1 to 0.2 0.08 to 0.2 | 800 to 1700 | 10 to 30 | 3 or 4 |
| Semiconductor RAMs | Bipolar transistors | Yes | No | Will survive but lose data | 0.025 to 0.100 | 0.5 to 5 | 1000 to 3000 | 5 to 8 | +5 volts only |
|  | Static MOS transistors | Yes | No | Under development | 0.5 to 1.0 | 0.2 to 0.3 | 1000 to 3000 | 3 to 5 | +5 volts only |
|  | $I^2L$ | No | No | Same as static MOS | 0.1 to 0.5 |  | 1000 to 3000 | 1 to 3 |  |
|  | Dynamic MOS transistors | Yes | No | Same as static MOS | 0.5 to 1.0 | 0.05 to 0.2 | 1000 to 10,000 | 1 to 3 | 3 or 4 |
| Semiconductor ROMs | Bipolar transistors | Yes | Yes | Hard | 0.1 to 0.2 | 0.1 to 0.5 | 2000 to 10,000 | 4 to 6 | +5 volts only |
|  | MOS transistors | Yes | Yes | Same as MOS RAMs | 0.8 to 1.5 | 0.05 to 0.2 | 4000 to 40,000 | 1 to 4 | 2 or 3 |
| Semiconductor EAROMs (electrically alterable) | Ovionic amorphous switch | Yes | Yes | Hard | 0.3 to 0.5 | 0.5 | 500 | 10 to 12 | +5 volts only and +12 volts (read only) |
|  | Erasable MOS (MNOS) | Yes | Yes | Moderate | 0.5 to 3.0 Read 10 to 100 Write | 0.3 to 0.4 | 500 | 3 to 5 | 2 |
| Programmable ROM (PROM) | Bipolar | Yes | Yes | Hard | 30 to 100 | 0.2 to 0.5 | 1500 to 100,000 | 5 to 8 | +5 volts only |
|  | MOS | Yes | Yes | Same as MOS RAMs | 0.8 to 1.5 | 0.05 to 0.4 | 1500 to 20,000 | 3 to 6 | +5 volts only |

Notes:
(1) Storage element only. It is assumed that the drive and sense electronics can be designed to meet radiation requirements
(2) Estimate includes storage array and drive and sense electronics
(3) Production quantities-military system. Electronics included. Also includes cost of design, fabrication, assembly, I&C, i.e., completed system. Cost varies widely with memory storage capacity and device type and design requirements.

are dynamic memory, and the data must be regenerated not only as a function of time when the clock frequency is low, as is usually the case in a standby mode, but also after a number of transfers due to charge losses. The CCD digital memory organization providing the highest density on the chip is a "serpentine" arrangement of long chains of serially connected shift registers. This configuration requires a minimum of peripheral circuits which use a relatively large amount of silicon real estate, and provides a highly repetitive (and, therefore, efficient) arrangement for the CCD itself. To reduce access time, the CCDs can be organized in a serial-parallel-serial (SPS) arrangement or in a parallel multiplexed arrangement called line-addressable random access (LARAM). The SPS pattern is very efficient in power requirement. It employs an information flow as shown in Figure 2.3-1. The memory is entered with a high speed serial shift register; when the input register is fully loaded, all of its cells are transferred simultaneously downward to an output register which is read out serially. Because of the parallel transfer, the downward shift occurs only once for each row as a whole, thus the row gates operate at $1/n$ times the frequency of the input register, and the power requirement for the interior rows is relatively small compared to that of the input and output registers. In addition, the M-fold increase in time allowed for the parallel-downward transfer helps to achieve a high charge transfer efficiency and low power.
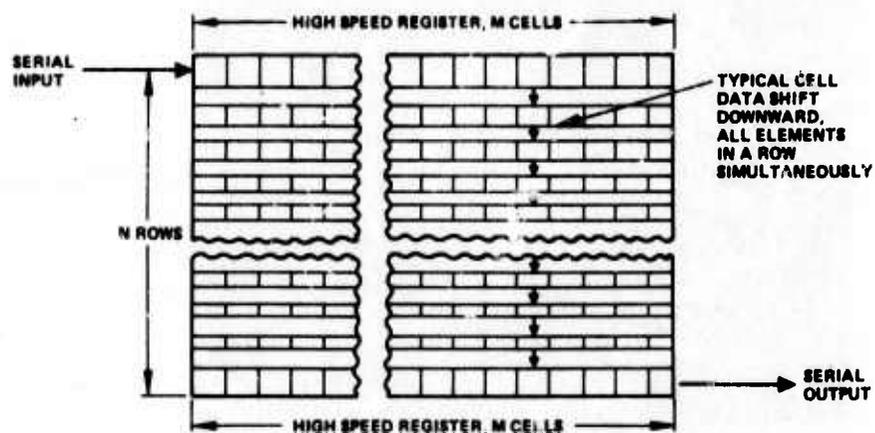


Figure 2.3-1. SPS memory data flow.

A manufacturing problem of the SPS design is that, unlike the ordinary linear register, all the data do not traverse the same path in the CCD. Local imperfections on the surface of the chip may distort an individual portion of the data. The line addressable memory structure provides independent access to each cell of the input and the output register, so that a defective line can be wired out of the circuit by connections inside or outside the integrated circuit package.

The practicality and usefulness of CCD digital memory have become apparent. Several companies are now offering as stock items memory chips with 16 kilobit capacity and line address organization for commercial temperature range. These chips have relatively slow access time but are small and need little power.

An example of more advanced technology is the Hughes SPS type 2069 memory chip, shown in Figure 2.3-2, which has 32 kilobit capacity and requires only 5 mW input power at 1 MHz clock rate. A new version of this memory is being built with a smaller basic cell size and with 64K bits. It
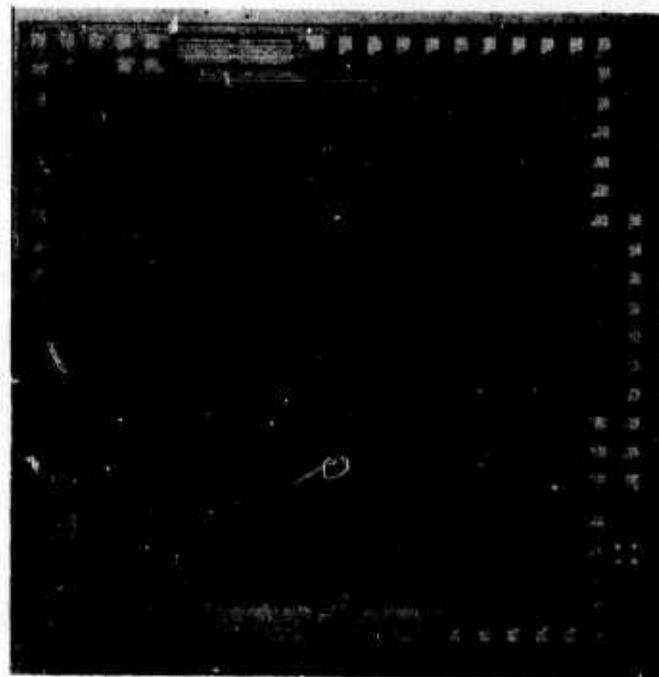


$\ell$ = 0.7 MIL/BIT
CHIP SIZE: 126 x 129 MILS

Figure 2.3-2. Hughes $2^{15}$ (32,768 bit) memory chip 2069.

appears to be quite practical to increase the speed and the chip capacity by orders of magnitude within the coming years. The most modern projection photolithography, mask making, and processing techniques will be required. Table 2.3-2 presents the range of digital memory CCDs now available as experimental or stock items. A future expectation of a single chip with 1 MHz clock rate and $10^6$ bit capacity, organized in very long registers, is quite realistic.

Most of these chips are intended for a commercial market and may not necessarily meet the military temperature requirements. The curve shown in Figure 2.3-3 illustrates the area of application of these digital CCD memories, showing the per bit cost versus speed of operation. If power dissipation, mechanical reliability, and other factors are taken into account the CCD memory may also be competitive with disc memory at the lowest cost level.

## 2.4 DIGITAL LOGIC FAMILIES

In this section the key digital LSI technologies are presented. These separate into four general branches: bipolar, MOS, CCD, and miscellaneous technologies. The individual categories are described and evaluated in terms

### TABLE 2.3-2. CHARACTERISTICS OF SINGLE CHIP CCD MEMORIES

| | Technology | Bits | Clock Speed | Power, mW | | Access Time, μs | |
|---|---|---|---|---|---|---|---|
| | | | | Operate | Standby-Recirculate | Nom | Max |
| *Fairchild CCD 450 (1975) | n Channel | 9 x 1024 | 50 kHz - 3 MHz | 250 | 30 | 168 | 340 |
| *Bell Northern (1974) | | 8 x 1024 | 1 MHz | | | 128 | 256 |
| *Intel 2416 (1975) | | 16 x 1024 | 1.3 - 2 MHz | | | | 200 |
| *Fairchild (1975) | n Channel | 16 x 1024 | 5 MHz | 200 | | | |
| +Westinghouse (1975) | | 2048 | Non volatile memory | | 50 | 12.8 | 25.6 |
| +Fairchild (1975-76) | | 32K | | | | | |
| +Hughes 2068 Low Power Memory Chip | n Channel | 32K | 1.7 MHz-20 MHz | 10 | | | |

*Offered for sale as stock items, 1975
+Experimental or proprietary devices
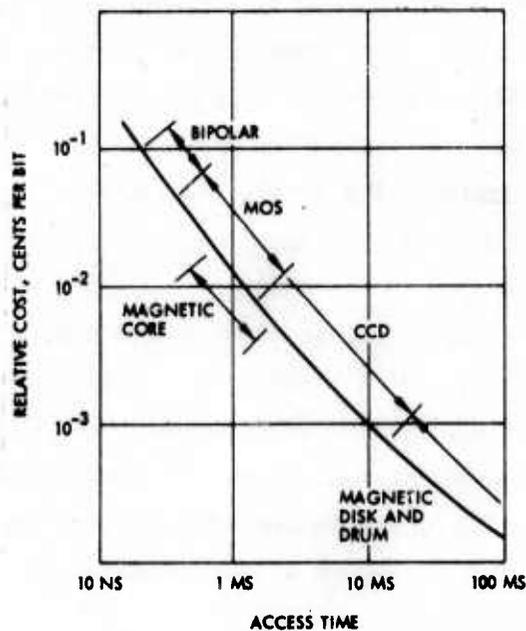
Figure 2.3-3. Application scope of CCD memory devices.

of potential application to the LSI requirements for the APSP. Only those technologies that appear as certain contenders for high speed low power LSI are carried further. $I^2L$, CMOS and DMOS appear to be the most likely technologies to fulfill this role in the early 1980s.

### 2.4.1 Bipolar LSI

Until recently the progress of bipolar LSI has practically been at a stand-still while MOS technology continued to advance and thus has dominated the LSI scene. Bipolar logic has remained the industry workhorse for high speed, but has not evolved effectively into LSI for two basic reasons; size and power. Size: the standard $T^2L$ LSI gate requires 20 sq mils compared with P-MOS with 11 sq mils or N-MOS with 5-1/2 sq mils. Power: in order to maintain gate delays less than 10 ns per gate, power consumption becomes 10 mw. Larger wafer size can accommodate the large per gate area requirement (at the sacrifice of yield) but the high power consumption for a moderate sized 300-gate chip at 10 mw per gate would be 3 watts, requiring external cooling. In order to reduce per gate power, larger resistors are needed to limit current, which increases the area requirement, thereby aggravating the yield problem.

2-40

The "bottom line" is power-delay product. For SSI $T^2L$ the power delay product is approximately 100 pico joules. A 300-gate chip with a total power dissipation of 300 mw would limit per gate power to 1 mw and gate delay would grow to 100 ns. At 100 ns delay per gate bipolar logic loses its speed advantage. Going further to 1000 gate equivalent LSI circuits the problem becomes even worse with very large wafer sizes and even slower logic. Consequently, the growth of bipolar LSI has been very limited.

ECL-10K (Emitter Coupled Logic) is the highest speed logic technology currently available from multiple sources. With this technology, production integrated circuits have been built which exhibit gate propagation delays ranging from 600 pS for a 10 mA switched current to 5 ns for a 1 mA current. This performance can be realized in relatively high yield LSI arrays. Moreover, the technology is sufficiently well established that even in relatively low volume production, LSI components exhibit minimal part failure rates. However, as seen on the power-delay curve of Figure 2.1-1 the power consumed by ECL-10K and related bipolar technologies is high. It appears that the power delay product will not approach the requirements of the space-borne APSP.

Integrated Injection Logic ($I^2L$) or Merged Transistor Logic (MTL) was introduced as a new form of bipolar logic circuit in two papers presented at the 1972 International Solid-State Circuits Conference.[2.4-1][2.4-2] $I^2L$ represents a major advance in high density, low power-delay product bipolar logic, and is one of the promising technologies for the LSI APSP requirements. The evolution, present status and future of $I^2L$ are discussed in detail in section 2.5 of this report.

## 2.4.2 MOS LSI

Current production and future high-speed MOS LSI technologies are described below in essentially chronological order of their development. Diagrams depicting current state-of-the-art device cross sections are included along with each discussion in most cases.

P-MOS -- The first LSI technology, P-MOS, was introduced commercially in 1967. Offering high density and a minimum number of processing steps, it opened up the electronic calculator market by providing highly complex, low cost arithmetic circuits. [2.4-1] P-MOS memory circuits soon followed and have since been produced in greater volume. The production of these ICs, shift registers, read-only- and random-access-memories, formed a foundation of experience on which current, more advanced, MOS memory technologies are built.

The cross section of a typical P-MOS transistor is shown in Figure 2.4-1. a single boron diffusion process step is used to form both source and drain, and channel length is defined implicitly by the distance between diffused areas. [2.4-2] In normal switching circuit operation a negative voltage (larger than a specific threshold voltage) applied to the gate will cause channel inversion (i.e., a p region will form beneath the gate oxide and provide a conductive path between source and drain). Thus, the aluminum gate must overlap both source and drain diffusions by a sufficient margin to ensure that mask alignment errors do not result in the channel
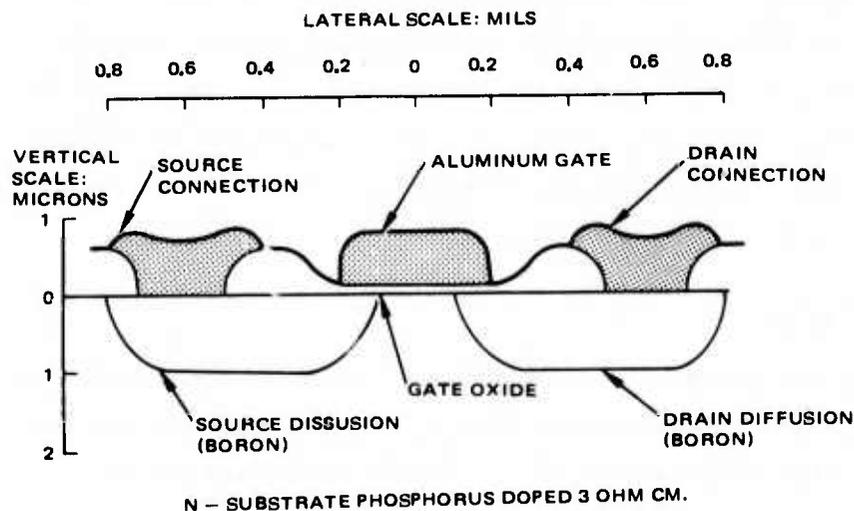


Figure 2.4-1. P-MOS device
cross section.

being partially uncovered. This would have the result that part of the channel could not be inverted and, therefore, the device would be rendered useless.

A basic MOS inverter is shown in Figure 2.4-2. In a P-MOS circuit, the load device is a second P-MOS transistor whose gate is returned to a $V_{GG}$ power supply (where $V_{GG} < V_{DD} < V_{SS}$). Signal levels in the circuit switch range between $V_{SS}$ and $V_{DD}$. When the input equals $V_{SS}$, the inverter transistor is cut off, and the output is pulled to $V_{DD}$ through the load device. If the input changes to $V_{DD}$, the channel of the inverter transistor inverts, forming a p-region between source and drain which allows current to flow from the load device to $V_{SS}$, thereby pulling the output to $V_{SS}$.

P-MOS technology has various shortcomings that limit its speed and density, including:

1.  Low $g_m$/high impedance. Tolerances in photolithography and lateral diffusion require that the channel length be greater than approximately 0.2 mil. Clearly $g_m$ can be made as high as desired and the impedance as low as required simply by lifting all restrictions on gate width. However, parasitic capacitances are increased, as discussed below. Moreover, device sizes and component densities become unacceptably large in the achievement of $g_m$ greater than a few hundred micromhos and impedances less than a few kilohms.

2.  High threshold voltage. The 6-8 volt thresholds of the original P-MOS require logic swings and $V_{DD}$ supply levels of 12-15 volts. Besides requiring special power supplies, the interface between P-MOS and bipolar logic is complicated.

3.  Large parasitic capacitances. The Miller capacitance caused by the gate-to-drain overlap is one of the main reasons why operating rates for circuits built with this technology are limited to
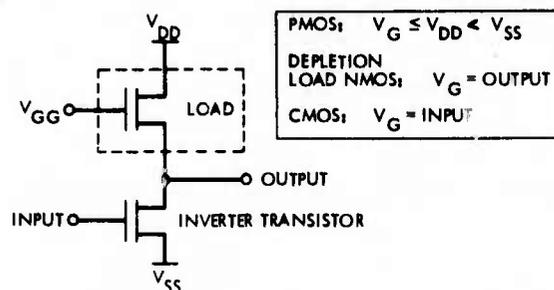


Figure 2.4-2. MOS inverter
circuit.

2-43

approximately 200 kHz.[*] Other limiting capacitances are the
drain-to-substrate capacitance and the channel capacitance which
must be charged before inversion can occur. The low resistivity
substrate used in P-MOS technologies causes these capacitances
to be comparatively large.

Lessening or bypassing these shortcomings has been the goal of all
subsequent MOS technologies.

Self-Aligned-P-MOS -- Major improvements over the original P-MOS
are realized in the self-aligned P-MOS gate structure. A typical cross
section is shown in Figure 2.4-3. As in original P-MOS, source and drain
are again formed by a single diffusion processing step. In the self-aligned
structure, however, these are positioned sufficiently far apart to assure that
the gate metalization does not overlap either diffusion. After the metalization
has been effected, boron ions are implanted to bridge the gaps between the
gate and the source and drain. The resulting device has negligible overlap
capacitance and can be used at clock rates approaching 20 MHz. Moreover,
a smaller device can be made using a self-aligned technique since the over-
lap distance need no longer be included as a part of the channel length.
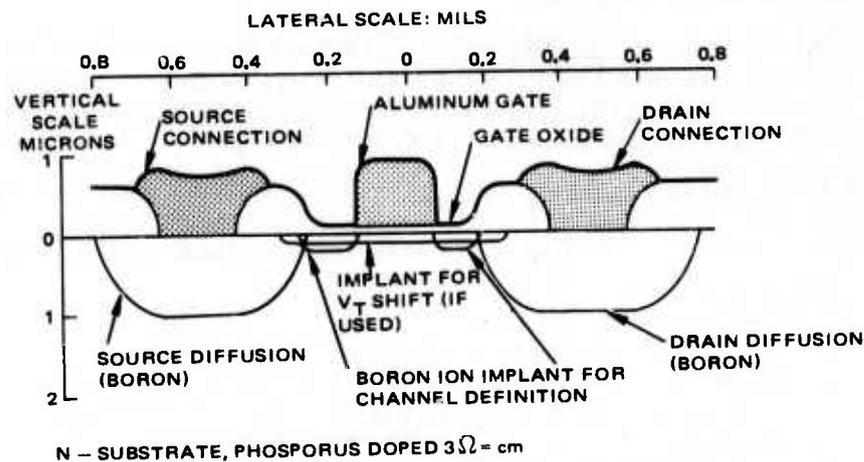
Figure 2.4-3. Self-aligned P-MOS
cross section.

Another improvement achieved using ion implantation was threshold voltage reduction. This technique was utilized after it was found that the high thresholds of the early devices were caused by positive ions trapped in the gate oxide. Subsequently, it was determined that the implantation of a precisely controlled amount of positive dopant into the channel could lower the threshold to any desired voltage.

N-MOS -- N-channel MOSFETs were considered promising from the very beginning of MOS development work because the high carrier mobility of n-type material (three to four times that of p-type) was known to result in higher transconductance and lower resistance for a given geometry. However, the positive ions which are always trapped at the silicon/oxide interface were found to cause the channels in early devices to invert without any applied bias. Worse yet, the same trapped charge, $Q_{ss}$, caused inversions in regions which were not intended to be channels, so that all diffusions were effectively interconnected.

As noted above, the effect of the trapped charges can be counteracted by implanting additional positive ions into the silicon substrate, to assure that the channel remains p-type in the absence of applied gate voltage. Thus ion implantation development was required for N-MOS processing. The cross section of a typical ion implanted N-MOS device is shown in Figure 2.4-4.
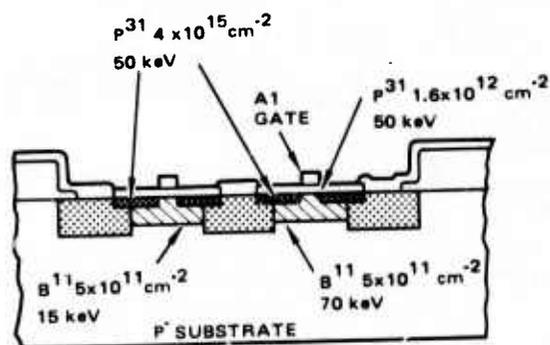


Figure 2.4-4. Ion implanted
self-aligned NMOS gate.

If the implanted positive ions are omitted from the gates of the load transistors (cf., Figure 2.4-2), the resulting devices may be used with their gates and sources connected together, since no $V_{GG}$ bias is required to keep the loads turned on. Such depletion mode[*] loads are used in N-MOS to permit a significant size reduction relative to P-MOS. This reduction stems both from the smaller size of MOS devices and from their simplified interconnection requirements achieved by their use of depletion loads. Moreover depletion mode loads also allow faster circuit operation, since these load devices do not tend to turn off as output voltage levels approach $V_{DD}$. In a recent work[2.4-11] a ring oscillator N-MOS device using ion implantation with a depletion load achieved 115 psec propasation delay and 0.29 pJoule power-delay product, using very small geometries and high substrate doping (1 μm channel length).

Silicon Gate N-MOS -- A considerable improvement over the self-aligned aluminum gate N-MOS device was realized in the silicon gate device as suggested by the cross section shown in Figure 2.4-5. In silicon gate MOS devices, deposited polycrystalline silicon is used instead of aluminum for the gate material. As much as a 50 percent reduction in IC die size can can be realized with silicon gate circuits since the polysilicon can be used as an
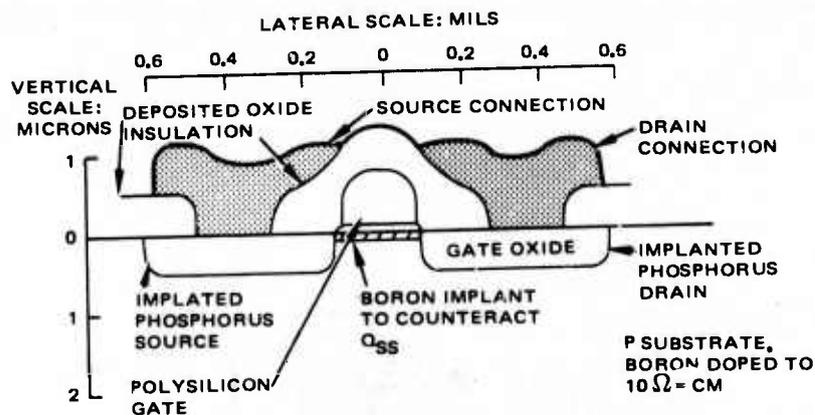


Figure 2.4-5. Silicon gate N-MOS cross section.

---

[*]Depletion mode devices require an applied gate-to-source voltage to turn them off (depleting the current flow), whereas enhancement mode devices require gate-to-source voltage to turn them on.

additional interconnection layer for signals as well as allowing the source and drain connections to overlap the gate. For this reason, silicon gate technology is used widely in large memory arrays.

CMOS -- Complementary MOS devices are attractive for low power applications because both the inverter (n-channel) and load (p-channel) transistors are switched by the input signal with the result that power is consumed essentially only during switching. Individual CMOS transistors are similar to those discussed above, except that N-MOS and P-MOS devices must be separated to prevent spurious conduction paths from forming between parts of the p- and n-channel devices. Heavily doped guard rings or channel stop diffusions typically are placed around groups of devices of each type to achieve this separation. A loss in circuit density, relative to N-MOS, results. However, a substantial reduction in power dissipation is achieved, particularly for those logic systems that are operated at fractions of the highest toggle frequency. Since CMOS represents a viable contender for APSP logic, it is discussed in detail in section 2.6.

CMOS on Sapphire -- Considerable developmental effort has been aimed at producing integrated circuits on insulating substrates (either sapphire or spinel). During the past few years reasonable quality silicon epitaxial layers on these substrates have been produced, into which MOS transistors can be fabricated as separate islands.

Advantages of this technology include greatly reduced capacitances between active elements and the substrate, as well as improved packing density made possible by the simplified isolation between devices. Auto-doping of the epitaxial layer by aluminum ions migrating from the substrate has been a source of problems, though, as has the higher imperfection density caused by mismatch between the silicon and substrate crystal lattices. [2.4-3]

This technology has been used at Hughes in the construction of specialized divider circuits which operate at clock rates as high as 50 MHz. However, the low transconductance which is common to all MOS technologies (except DMOS variants) as discussed above, limits the maximum frequency which can be propagated between packaged devices to approximately 20 MHz.

<u>D-MOS</u> - · One new structure that exhibits a higher $g_m$ and a lower output impedance than any of the MOS devices considered above is D-MOS (or double-diffused MOS). As shown in Figure 2.4-6, an extremely short channel is obtained by first diffusing boron into the source region and then diffusing phosphorus through the same oxide opening. In this process diffusion times are carefully controlled so that the boron diffuses laterally about 0.04 mil (1$\mu$) further than the phosphorus, forming a short p region extending toward the drain. The process is controlled so that the impurity concentration in this region will be high enough to make the device operate in an enhancement mode.

The remainder of the distance between the source and the drain consists of very high resistivity p type material which is inverted by charges trapped in the oxide, thereby providing a "drift" region that serves to increase the source to drain breakdown voltage and to greatly reduce the Miller capacitance. However, the drift region is also affected by gate voltage so that the behavior of the composite device must be modeled as a depletion mode transistor in series with the enhancement mode inverter transistor.[6.4-4]
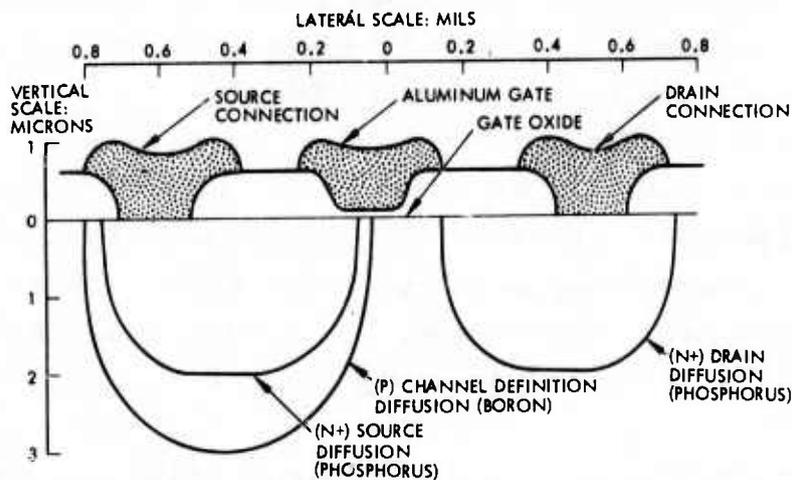


Figure 2.4-6. D-MOS cross section.

When used in logic circuits, the D-MOS device can be treated simply as a very good N-channel MOSFET. Thus D-MOS devices may be connected in series to form NAND gates, and/or in parallel to form NOR gates, and any of the load devices described above may be used in logic circuits. Some form of channel stop diffusion is required, however, in DMOS circuits because of their very light substrate doping.

D-MOS outperforms all of the standard MOS technologies considered above. Using this technology, 2.9 nS gates have been fabricated and an experimental 11-stage ALU which exhibited 32 nS total delay has been realized.[2.4-5] Since D-MOS represents a potential technology for the APSP requirements it is discussed in greater detail (see 2.7).

V-MOS -- Another new short channel MOS structure is V-MOS, shown in Figure 2.4-7. In effect, the device is a vertical (more precisely, an inclined) D-MOS transistor with the substrate as its source. Despite its complex appearance, its fabrication involves no difficult processing steps, (reference 2.4-6) with the possible exception of the nonstandard anisotropic etch required to form the four-sided pyramidal gate.

Since the channel extends completely around the gate opening and is only 1 micron long, extremely efficient use is made of the gate area. In fact, it has been reported that a V-MOS device was made with approximately one-fifth the lateral gate area, one-third the active area, one-half the gate capacitance, and twice the transconductance of a silicon gate N-MOS device produced to the same tolerances in the same laboratory.[2.4-6]



V-MOS CROSS-SECTION          D-MOS CROSS-SECTION
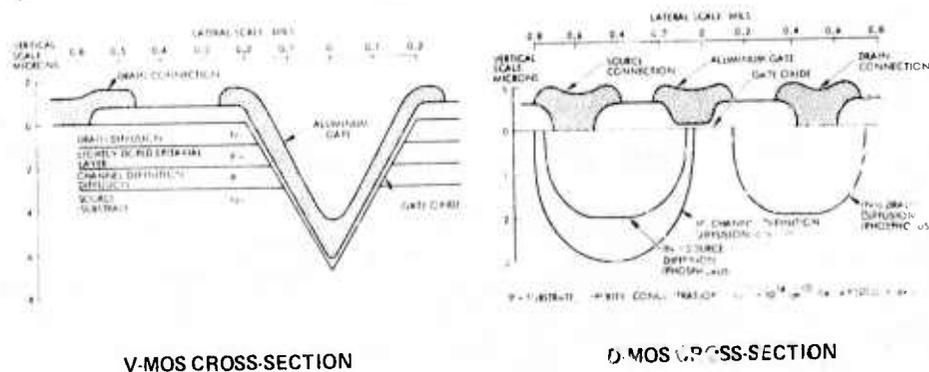
Figure 2.4-7. V-MOS cross section.

V-MOS devices have been used in an experimental counter circuit that toggles at 33 MHz and in an inverter chain that was used to drive two TTL loads at 60 MHz. V-MOS devices are likely to find wide application in future memories but are inefficient in their implementation of random logic circuits. This is true because all V-MOS source nodes must be connected to the substrate, thereby limiting these devices to use in implementing only NOR gates.

### 2.4.3 CCD Digital Technology

Charge Coupled Devices (CCDs) have enjoyed a tremendous growth since their introduction in 1970. This section will briefly introduce the operation of surface and buried channel CCDs and then discuss the status and future of CCDs as logic devices in LSI.

The operation of CCDs is controlled by electrodes that cover the surface of a glass insulating layer on the silicon substrate, as illustrated in Figure 2.4-8, which shows a p type substrate. The positive electrode voltage repels majority carriers (holes) in the silicon, creating a depletion region of negatively charged acceptor sites as shown in Figure 2.4-9. This region extends to a depth in the substrate that increases with magnitude of the gate voltage. When minority carriers (electrons) are present, they tend to collect at the silicon-glass interface and thus decrease the surface potential, which also reduces the extent of the depletion region. The empty cell condition is, in fact, a non-equilibrium condition, since minority carriers are spontaneously generated by thermal effects in the bulk of the silicon, this is usually
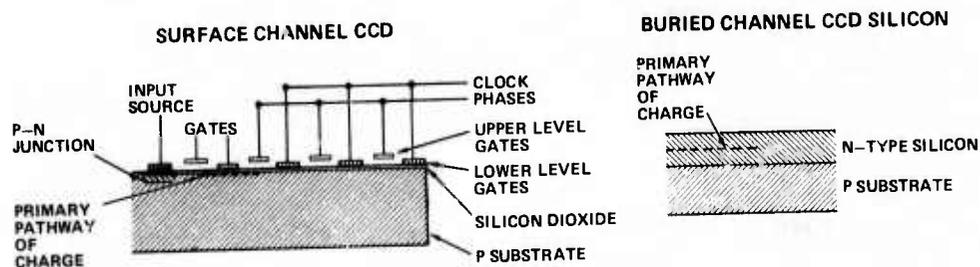


Figure 2.4-8. Typical cross section of CCD.

a. Empty cell

b. Cell with mobile
minority carriers
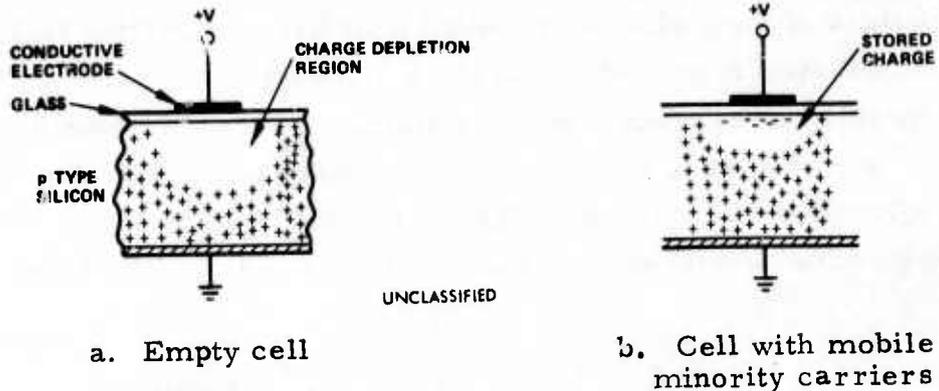
Figure 2.4-9.  Charge distribution in a surface
channel CCD.

called "dark current".  When the register is operating, these minority carriers are swept along and the cell is cleared at every transfer.  The potential of the surface of the p-type silicon is at a minimum beneath the center of the bias electrode.  For a p-type surface channel device the potential varies as shown in Figure 2.4-10.  The electrons are stored at the potential minimum.



$V_G$ = GATE VOLTAGE
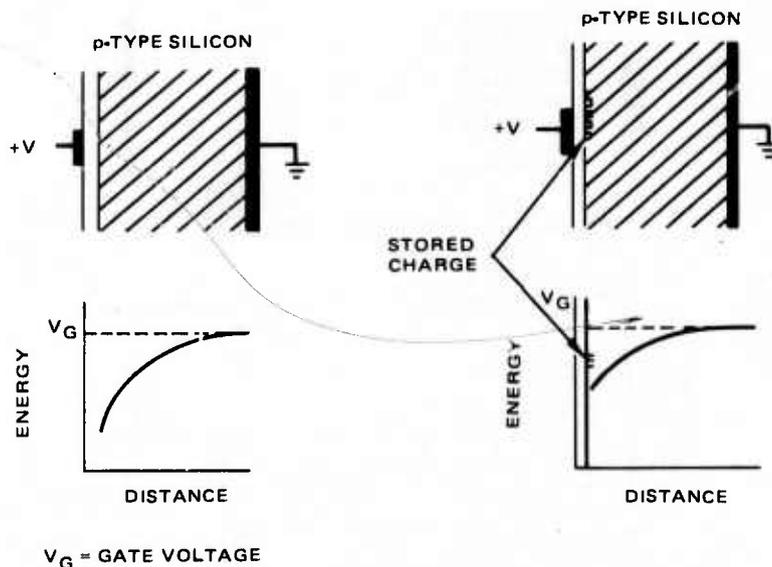
Figure 2.4-10.  Potential distribution in a
surface channel CCD.

In a buried channel structure, a layer of n doped material is introduced on the surface of the p silicon by epitaxial or ion implantation techniques. A typical BCCD structure is shown in Figure 2.4-11.

By this method, the potential minimum under the centerline of the electrode is moved away from the glass-silicon surface into the n type donor layer. The typical potential distribution and the location of the stored charge near the potential minimum are shown in Figure 2.4-12. In the absence of
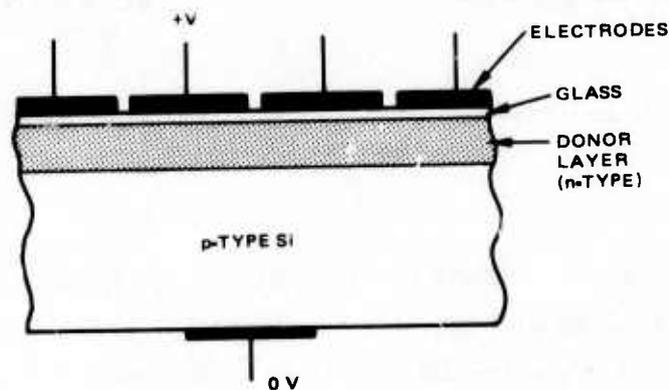


Figure 2.4-11. Buried layer CCD structure.
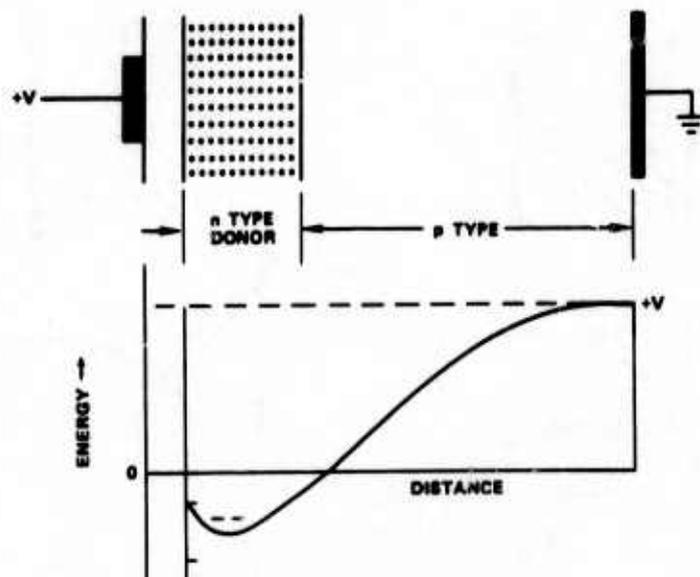


Figure 2.4-12. Energy level beneath the centerline of an electrode in a buried channel CCD.

2-52

stored charge, the donor layer is occupied by its minority carriers; when there is a signal packet of electrons, donor minority carriers are present in layers extending out to the glass interface and to the p-n junction. Thus, the potential minimum remains in the n layer and charges are transferred inside this layer.

When a positive voltage is applied to the electrode, the mobile majority electrons in the surface layer are attracted, leaving bare positively charged donor sites. This positive region acts to create a depletion region in the p type silicon substrate which is quite similar to the distribution within a surface channel CCD substrate. The electric field gradient goes to zero along the centerline under the gate electrode at a point just inside the doped layer; this locates a potential minimum inside the layer, along with injected signal electrons travel. Since there are usually many more charge trapping states on the surface of silicon semiconductors than inside the bulk material, charge is transferred more rapidly and more completely in buried channel devices. Transfer inefficiency values (fraction of charge not transferred) of $10^{-4}$ to $10^{-5}$ are readily attained.

In the surface channel device, the depth of the depletion region of an empty well is a measure of the total well capacity for charge transfer. As indicated in Figure 2.4-9, the depletion region shrinks in depth as more charge is accumulated under the control electrode. When the depletion region approaches the channel surface, the maximum bucket capacity has been reached. In buried channel devices, a similar maximum capacity restriction occurs when the charge in the buried channel becomes so large that the potential minimum that defines the well spreads out to the surfaces of the buried channel.

The charge packet is moved along the structure of the CCD by an electric field which is swept along the line of control electrodes by application of appropriately phased voltages. It should be noted that the difference in level of the electrodes provides the step in surface potential needed to force the charge to transfer in the desired direction. Two-level gate electrodes must be used in one- and two-phase clock sequences, and as a result,

2-53

overlapping of the electrodes is required to reduce the effects of stray fields. With three- or four-phase clocks, the step in surface potential can be obtained even with single layer electrodes. Another technique of transfer gating is to modify the surface doping to create a potential step under the electrode, whi h yields a result similar to that of a two-level gate electrode.

The input, output, and control transfer gates are fabricated of aluminum or doped high conductivity polysilicon. One advantage of overlapping metal gates is that they are inherently shielded from external fields and have low resistivity. However, polysilicon is easier to process and is mechanically somewhat more compatible with the glass insulation layers which are used. Gate structures of metal overlapping polysilicon are frequently employed. The advantages of the two-level overlapping structure are its shielding and relative simplicity. The single layer structure is susceptible to stray fields and requires careful control of the gap spacing.

The rapid advance in the speed of CCD operation offers a future possibility of a high level of circuit integration. The maximum frequency of operation of a CCD depends on its detailed design and transfer efficiency required for a particular application. Furthermore, since the maximum clock frequency depends on the dimensions of the CCD in the direction of charge transfer, the capabilities of the photolithographic process used to fabricate the CCD usually limit high frequency performance. Conventional photolithographic techniques limit the minimum dimensions of a device to 5 to 7.5 $\mu$m. A high resolution projection photolithographic process developed at Hughes allows devices with minimum dimensions of 1.5 $\mu$m (10 to 15 $\mu$m per bit) to be fabricated. Estimated high frequency limits of various CCD structures for conventional and high resolution photolithography are indicated in Table 2.4-1.

To make effective use of the high speed capability of the basic CCD transfer mechanisms, fast support circuits, probably including bipolar or DMOS devices on the CCD chip, will have to be incorporated. The process technology for single chip combinations of bipolar and CCD devices has been demonstrated with laboratory and experimental devices by Hughes. (Reference sections 4.5 and 4.6 of this report.) In the future this capability and the

TABLE 2.4-1. MAXIMUM CCD CLOCK FREQUENCY

| CCD Structure | Conventional Photolithography, $f_c$, maximum, MHz | High Resolution Photolithography, $f_c$, maximum, MHz |
|---|---|---|
| Surface channel | 10 to 20 | 40 to 60 |
| Shallow buried channel | 20 to 30 | 60 to 80 |
| Peristaltic (deep buried channel) | 150 to 200 | 300 to 400 |

high resolution projection photolithography process (which was developed for the fabrication of high density CCD memories) will provide operation above 300 MHz. This operation will occur after establishing the processes and in the solution of other problems, including the technique of applying small size effective on-chip interconnects between the bipolar circuits and the CCDs.

A typical block diagram of a high speed digital CCD chip is shown in Figure 2.4-13. To achieve high speed performance, the cell size must be small, hence control gates and the interconnection lines must also be small, and the devices must be packed densely on the chip. The main bus for distribution of clock signals probably will have to be fabricated of aluminum because of the high current density needed to serve many electrodes.
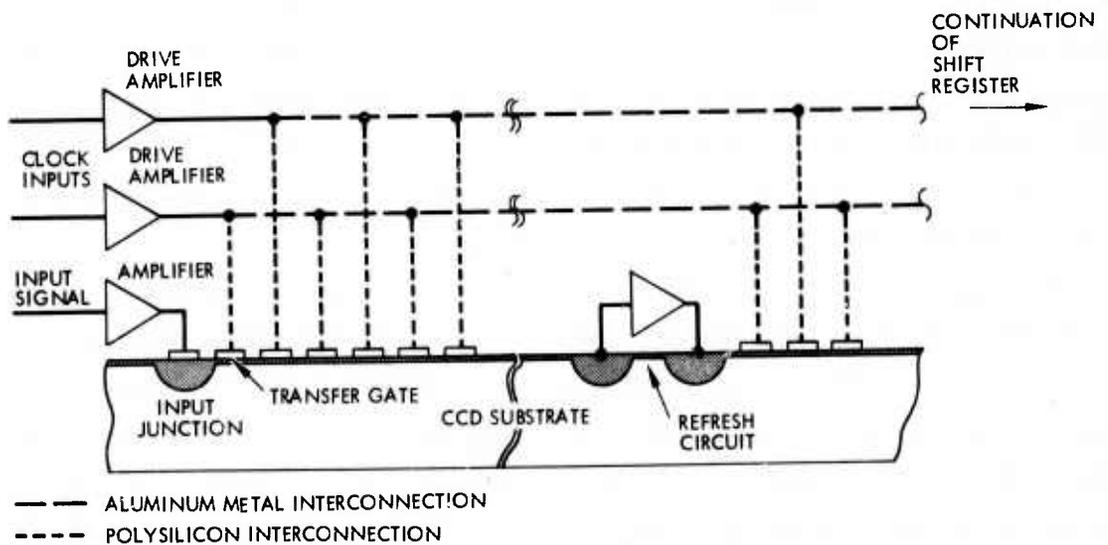


Figure 2.4-13. Bipolar/CCD shift register.

The control electrodes will use local branch connections to be fabricated with low resistance polysilicon; because of the small geometry and close spacing, the resistance of the polysilicon leads may be a design factor.

Although CCDs have been actively developed for imaging, analog signal processing and digital memories, relatively little effort has been devoted to CCD digital logic. The Hughes CRC 100 chip to be tested in early 1976 includes two unique digital adder circuits. The very low power, high density and relatively high frequency possible with CCDs suggests that useful CCD logic structures may be possible. The logic functions are accomplished by transfer of charge under control electrodes which are over-lapped with appropriate transfer or barrier gates. The fundamental oper-ation of binary arithmetic for single digit numbers are accomplished by addition of charge from two sources into a single well and by sensing the overflow, when two digits are simultaneously present behind the barrier gate. Floating electrodes or diffusions can be used to non-destructively sense the resulting charge and control other logic barriers to provide com-plex functions.

Low power CCD memories have been steadily advancing (see sec-tion 2.3) in speed, power and size. However, the complexities of CCD logic in terms of clock generation, bias and control voltage inputs, inter-connects, and regeneration and output circuitry indicate that competition for the ultimate low power-delay product LSI Digital system will be a difficult, uphill battle for CCDs. The competitors ($I^2L$, CMOS, DMOS) are continu-ously reducing their size (capacitance) and output voltage swing. Note that one generally assumes that signal-to-noise ratio remains approximately constant as output voltage swing is reduced, since digital noise is primarily due to other logic devices of the same type in the system. However, swings below approximately one volt do begin to reduce S/N ratio and thereby complicate the tradeoff between noise immunity and power dissipation. Assuming the power delay product of the 1980's is in the region of 0.1 pJ for ring oscillator circuits of other technologies, CCDs will have to achieve (for 2.0 volt clocks) 0.0025 pf/bit or 0.0125 $mil^2$/bit, or very efficient reactive clock generators must be perfected. In addition the ratio of power-delay

performance for LSI logic functions compared to ring oscillator circuits is likely to be considerably larger for CCDs than for other technologies due to the complication of clock line interconnects, reset requirements and output and regeneration devices.

For these reasons it is felt that, though possible, it is not probable that CCD logic will be in the running for the lowest power delay, highest packing density technology. CCD's appear to be appropriate in some circuits (in the AVE for example) to perform simple logic functions. Also, CCD A/D and D/A converters and CCD Serial memories do not have the drawback of complex peripheral circuitry requirements. Hence CCD technology has a unique place in shift register — like applications where there are no competing technologies on the horizon. The APSP system configuration permits these features of CCD's to be fully exploited.

### 2.4.4 Other Logic Technologies

Logic families other than discussed in the previous paragraphs have been developed. In the interest of clarity, the technologies that fall far short of meeting the low power delay product, high speed and high density required for the APSP have not been included. Two interesting advanced technologies however are worthy of consideration, and are discussed below.

Transfer Electron Device (TED) Logic -- The fundamental speed of a logic circuit in LSI is an essential factor in determining the cost effectiveness of the logic circuit. Although not presently amenable to realization in other than SSI configurations, gallium arsenide transfer electron devices (TEDs) can be operated at very high speeds. Therefore, TEDs were evaluated in relation to their possible future applications in low cost signal processing, in spite of this limitation.

TEDs implement Gunn effect electron transport which permits very high performance device operation. The Gunn effect results from the fundamental physical parameters of gallium arsenide. In this regard, its quantum mechanical energy-momentum diagram exhibits satellite valleys,

in addition to the central valley, for electron transport along a vector direction determined by the 1-0-0 edges of the first Brillouin zone. The "bottom" of the first satellite valley is located at a higher energy and momentum state than the "bottom" of the central valley. However, electron mobility is much lower (by approximately a factor of two) at the satellite valley. Thus, as the electric field is increased along the appropriate axis of the crystal, the electron velocity increases and then decreases abruptly as bunching of electrons occurs. (The formation of a high-field-concentration domain results. The remainder of the crystal remains in the state of higher mobility.) The increased voltage drop across this domain reduces the potential drop along the remainder of the conduction path and thus a corresponding overall reduction of the current takes place. The domain travels from the cathode toward the anode where it is absorbed. Then the crystal returns to its initial high current condition, and if the electric field remains constant, a new domain will be formed.

The build-up and annihilation of the domain occurs very rapidly (in a few picoseconds). The transit velocity of the domain is about $10^5$ m/sec. Hence, for example, a 10 micron transit path will correspond to production of pulses of about 100 picosecond duration.

The circuit technique by which the Gunn effect is utilized for logic operation is one of extracting an output voltage pulse in a series resistance (either an external element, or an integral part of the device). In logic circuits, the diode is operated below the threshold field required for domain formation, and a triggering field is applied by a gate electrode placed across the path of current flow. A typical device is shown in Figure 2.4-14.

All important basic logic functions can be realized with a single planar Gunn device similar to the structure shown. The logical AND operation can be accomplished in a structure incorporating a pair of control gates for which critical field is achieved only with both gates activated. The logical OR may be realized by a pair of gates either of which can generate a critical field. Structures for exclusive-OR (and hence inversion), comparator operations, logical-carry generation, and other ingenious arrangements have been devised. These functions are achieved in structures incorporating multiple cathodes, or in structures utilizing lateral domain spreading. [2.4-7,8]
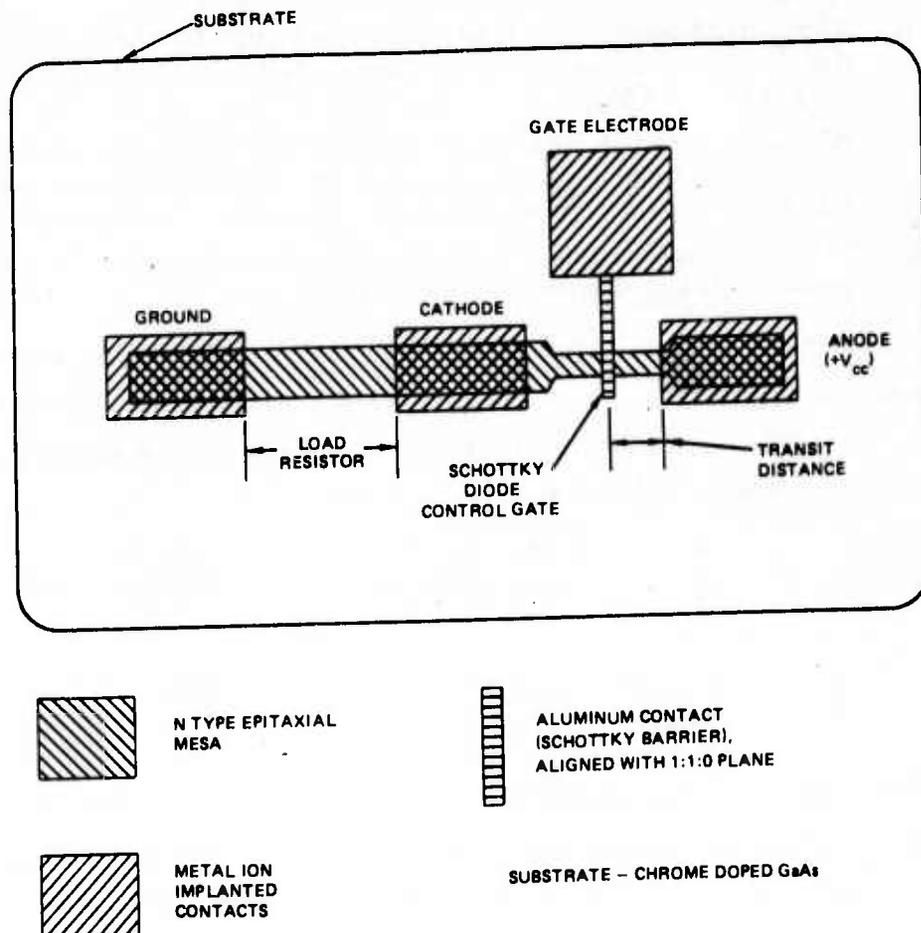
Figure 2.4-14.  Gate controlled Gun diode.


Actually to achieve logical operations suggested above in working TEDs, several device characteristics must be considered.  These include:

1.  With a threshold device, a certain minimum quantity of energy-momentum is required for switching.

2. The fundamental TED switching time is short because the electron transfer mechanism is fast (on the order of picoseconds). As a result, the device switching time is determined primarily by parasitic elements in the external circuitry.

3. The output pulse shape is controlled only by the parameters of the device and input pulse shape is not necessarily related to the output pulse shape.

4. Any additional inputs, after the device has switched, cannot change the output pulse, e.g., a second input pulse cannot initiate a second dipole domain until the first has been extinguished.

5. The pulse propagation velocity is constant since the pulse propagates at the dipole domain velocity.

In addition to the above considerations, there are fundamental constraints on device material parameters and form factors. As reported in the literature, the doping density, n, the device length, $\ell$, and the device thickness, d, have lower limits for stable domain formation given by:

$$n\ell \geq 10^{13}/cm^2$$

$$nd \geq 10^{12}/cm^2$$

Thus, fundamental physical properties of the material are design constraints. The most important of these results in limitations on logic speed and reliability. Maximum logic speed, as noted (pulses per second) at which the device can be utilized, is primarily determined by the pulse propagation time (i.e., the time during which no additional input pulse will cause an output pulse). This pulse propagation time is determined by the domain propagation velocity and the length of the device. Reliability is primarily associated with device operating temperature, which places a restriction on the power dissipated by the device. Since the highest power consumption is at threshold, the threshold power must be minimized sufficiently to give reliable operation. This minimum must be determined somewhat subjectively until reliability testing can establish the appropriate failure mechanisms and predict time-temperature relations.

If Gunn effect devices are to be competitive with other logic elements, they should be usable in LSI arrays. Moreover for high speed operation, the resistance of the circuit should be low and the transit time of the domain from the gate electrode to the anode should be as short as possible. These factors, plus the fact that power consumption decreases while the domain is in transit, indicate that the threshold power consumption should be minimized. A recent analysis of these problems[2.4-7] suggests that a figure of merit for an optimum device can be calculated by using the standard constraints on doping density, dimensions of the gate, the thickness of the device. This figure of merit is the product of threshold power $(P_{th})$ x device resistance $(R_o)$ x transit time $(t_r)$ and may be evaluated as

$$P_{th}R_o t_{tr} = 1.1 \times 10^{39} \, (V^2/n^3) \, \text{sec/cm}^9$$

where

      V is the critical threshold voltage

      n is the doping density

If the doping density is made too large, impact ionization may occur within the nucleated domain; therefore, n cannot be increased indefinitely. The critical voltage cannot be decreased arbitrarily. Hence, an optimum set of design parameters can be determined for dimensions and doping densities representing reasonable values according to current knowledge. For a diode notch width of 10 $\mu$m, a doping level of $10^{16}/\text{cm}^3$, and a gate electrode length of 1 $\mu$m, it was found that the theoretical value of minimum threshold power would be 12 mW. (Experimental devices have exhibited measured dissipations of 25 mW.[*])

Gunn effect logic offers extremely high speed of operation with an apparent capability of providing all the necessary logic functions. One disadvantage is the high threshold power required per gate, which is larger than required for other existing or projected forms of logic and is, in fact, so large that it will create significant thermal problems if TEDs are used

---

[*]There are experimental indications that dimensions of the devices cannot be reduced much below the values used in the calculation above, because of a "dead zone" effect near the cathode in which relaxation effects alter the behavior of the device.

in LSI high density arrays. A secondary problem is that if large scale
integration is not practical for TEDs then significant propagation delay will
be introduced in logic circuits in inter-chip connection networks. Thus, the
inherent high speed advantages of the Gunn effect will be lost. In view of
these difficulties and the lag in gallium arsenide technology development vis
a vis the current state of silicon technology, it is unlikely that TED LSI will
become available in the foreseeable future.

Josephson Junction Logic -- Josephson junction devices operate at
superconducting temperatures (e.g., $4.2^{O}$K). Like Gunn effect logic devices
they are capable of gigahertz frequency operation. In addition, however,
Josephson devices have the apparent advantage of very low power dissipation.
Power delay products of the order of $10^{-17}$ joules and propagation delays in
the 10-50 ps range have been exhibited in laboratory test circuits. The
basic switching circuit is fabricated as overlapping thin lead films separated
by pinhole-free hyper-thin (e.g., 30 Å) oxide tunnel barriers. The junction
is mounted on a superconducting metal ground plane and is controlled by an
overlaid strip of superconducting metal which carries a control current.
Because of their high performance, these devices are being evaluated as
alternatives to semiconductor logic circuits.[2.4-9]

Although the thermal effect of power dissipation is small in super-
conducting devices, Josephson junction logic circuits share problems common
to all ultra-high speed logic including:

1.  Signal delay and lead inductance in inter-chip connections
    increase total effective delay per gating stage.

2.  Bias voltage must be provided through a very low series
    resistance and inductance.

Over and above these common difficulties, the superconducting
Josephson devices have additional requirements.

1.  New techniques must be developed for production fabrication of
    30 Å thick oxide layers over large areas.

2.  New compatible superconducting and insulating materials and
    manufacturing processes must be developed.

3.  New design rules based on superconduction processes rather
    than semiconductor physics must be devised.

4. New LSI testing techniques must be developed since Josephson devices operate only in a liquid helium environment.

On the assumption that all the above difficulties can be overcome, it would appear that Josephson junction logic may have application at some time in the future in large ground based computing systems. Cryogenic cooling is impractical for large spaceborne signal processing systems and, therefore, it is not likely that Josephson logic will be applied as long as other alternative technologies are available and satisfy the APSP requirements.

## 2.4.5 Computing Power Concepts

In addition to power delay product, two characteristics are important inputs for digital LSI technology trafeoff studies, gate density and computing power. Gate density obviously effects yield, interconnect capacitance and single chip functional complexity (thus overall density of required output buffering). Computing power relates to what can be done with the gates or cells in terms of logic efficiency. A key characteristic of ECL logic which is valuable in terms of computing power is the inherent complement output availability in the individual cell. The following discussion briefly outlines an ECL Universal Logic Gate (ULG) [2.4-10] concept which should be applicable to other forms of logic. The concept was developed to minimize stages in very high speed radar signal processing. The important message is that through optimization of logic, improvements in computing power per gate or per $mm^2$ or per p joule can be made. Such studies will be important in the architecture definitions and optimizations for the APSP.

The ULG comprises one-stage arrays of two identical cascode circuits. These ULGs realize all logic functions of four (and fewer) input variables in approximately the same propagation delay as a single ECL current switch emitter follower (CSEF) gate fabricated with the same processing technology. Substantial power and power delay product advantages relative to CSEF arrays have been demonstrated using comparable silicon area for realization of all four-input functions. The ULG was developed for implementing logic arrays with a minimum number of

2-63

gating stages. ULGs permit realization of logic arrays with considerably improved performance which is achieved because logic functions can be factored or decomposed very efficiently using the ULG.

Reduction of the propagation delay through combinational (gating) arrays usually has been achieved by reducing the delay of the individual gates in these arrays. Frequently, gate power dissipation is increased or transistor performance is improved in the gates to achieve this reduction. In conjunction with this approach, more logically efficient gate circuits can be built and used in arrays. The primary objective of the work referenced herein was the development of new circuits for implementing logic arrays with a minimum number of serial gating stages.[2.4-12] Secondary goals were reduction of array power dissipation and silicon area through the use of fewer gate-building blocks. Since the LSI circuits were intended for use in very high speed signal processing, only bipolar technologies were considered.

A universal logic gate (ULG) is a combinational circuit that can be "programmed" to realize any specified function of its input variables. A one-stage ULG realizes any specified function in approximately the same propagation delay as a conventional gate built with the same technology. The single-stage ULG was defined based on a study of selected switching literature.[2.4-11] This research revealed that a ULG having the largest fan-in practical (i.e., three or four) and the same one-stage propagation delay as available (e.g., ECL), gates could be used to realize logic arrays with a minimum number of stages. In a ULG implementation of a specified logic function, further reduction in the number of gating stages in worst case array input/output paths can be achieved only by increasing gate (ULG) fan-in. ULG fan-in of four was selected as a compromise between circuit complexity of the ULG and potential stage-delay reduction in arrays.

Minimum gating stage logic synthesis with ULGs may be illustrated in the design of a 3 x 3-bit binary multiplier. A "conventional" logic design for this circuit, suggested in Figure 2.4-15a, requires four-gating stages in forming the most significant bits of the product.

$$A \cdot B \cdot C = \sum_{k=0}^{5} c_k 2^k$$

where

$$A = \sum_{i=0}^{2} a_i 2^i$$

$$B = \sum_{j=0}^{2} b_j 2^j$$

and

$$a_i, \ b_j, \ c_k = 0, 1.$$

A 2:1 reduction of the delay through this circuit and a correspondingly large reduction in the number of gating blocks are possible. These reductions may be achieved by factoring the logic functions for the individual multiplier outputs more efficiently. A minimum delay logic partition obtained for the two most significant multiplier outputs is shown in Figure 2.4-15b. (The Karnaugh map in each block designates the logical requirements of the block.) Actual realization of a minimum delay partition in general requires one-stage logic blocks capable of realizing any arbitrary logic function of four or fewer inputs, i.e., a one-stage ULG.

In addition to illustrating the utility of a one-stage ULG in reducing network delay, the above multiplier design also suggests the advantages of modular ULG construction. In this regard, it is noted that two and three-input gates would be used along with four-input ULGs if the logic partition of Figure 2.4-15b were implemented directly.

**INPUTS**

$N_1 = a_0 2^0 + a_1 2^1 + a_2 2^2$

$N_2 = b_0 2^0 + b_1 2^1 + b_2 2^2$

**OUTPUT**

$N_1 \times N_2 = N_3 = C_0 2^0 + C_1 2^1 + C_2 2^2 + C_3 2^3 + C_4 2^4 + C_5 2^5$
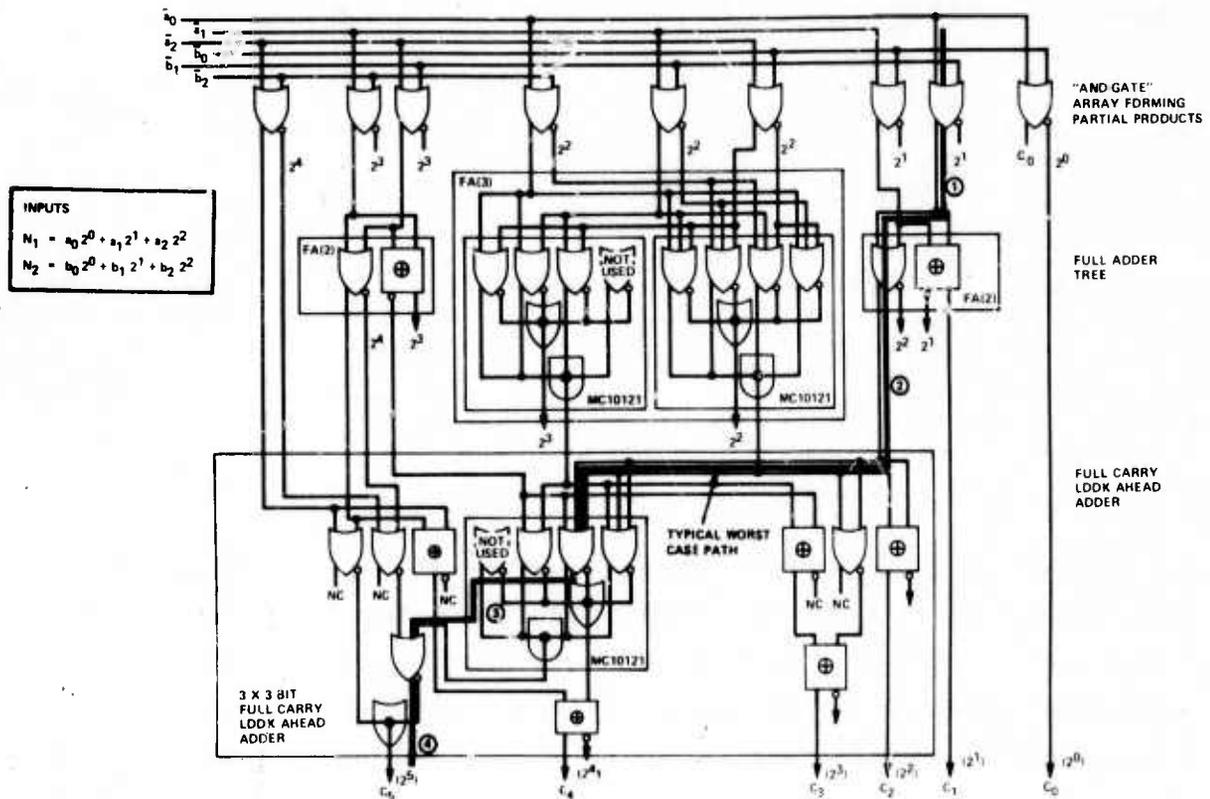
**(A) "CONVENTIONAL" DESIGN BUILT USING ECL-10K**

**(B) MINIMUM DELAY ULG LOGIC PARTITION**
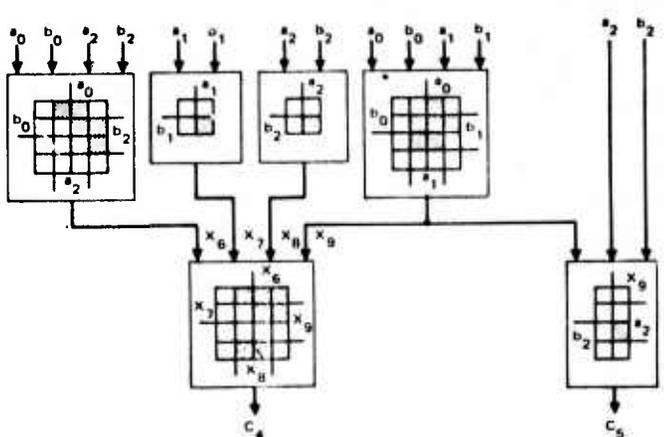
Figure 2.4-15. Logic design improvement using ULGs.

2-66

## 2.4 REFERENCES

2.4-1 Frederico Faggin, "The Role of Technology in Microcomputer Design and Evolution," IEEE Journal of Circuits and Systems, 7, No. 5, February 1975.

2.4-2 William N. Carr and Jack P. Mize, MOS/LSI Design and Application, McGraw Hill, 1972.

2.4-3 McMOS, Motorola Inc., Semiconductor Products Div., C1971, p. 40.

2.4-4 T. John Rodgers, et al., "DMOS Experimental and Theoretical Study," IEEE International Solid-State Circuits Conference, 1975.

2.4-5 Ohta, Juniichi, et al., "DMOS Experimental and Theoretical Study," Self-Aligned Enhancement Depletion MOST," IEEE International Solid-State Circuits Conference, 1975.

2.4-6 T.J. Rodgers and James D. Meindl, "VMOS: High Speed TTL Compatible MOS Logic," IEEE Journal of Solid State Circuits, SC-9, No. 5, October 1974.

2.4-7 K. Mause, et al., "Gunn Device Gigabit Microcircuits," IEEE J. of Solid-State Circuits, SC-10, February 1975.

2.4-8 Tetsuo Nakamura, et al., "Picosecond Gunn Effect Carry Generator," ISSCC Conf. Record, February 1975.

2.4-9 W. Anacker, ISSCC Record, Session XIV, February 1975, p. 162.

2.4-10 Hughes A/C Co, "Current and Future LSI Technologies," Contract No. F33615-74-C-1167, Section 6.0, Phase I Report.

2.4-11 Fang, et al, "High Performance MOS Integrated Circuit Using the Ion Implantation Technique," IEEE Journal of Solid State Circuits, Vol. SC-10, No. 4, August 1975, pp. 205-211.

## 2.5 I$^2$L TECHNOLOGY

This section will first discuss the development of I$^2$L configurations and the significance of this new logic to bipolar LSI. Later the device limitations inherent in the basic configuration and variations of the circuit which improve performance will be discussed. An attempt will be made to present the advantages and disadvantages of the various I$^2$L permutations in tabular form. Finally, the future of I$^2$L will be discussed based on the theoretical limit and projected state-of-the-art.

### 2.5.1 I$^2$L Development

Integrated Injection Logic (I$^2$L) or Merged Transistor Logic (MTL) was developed in an effort to resolve the problems associated with conventional bipolar logic. The I$^2$L structure can be derived from direct-coupled transistor logic (DCTL). Figure 2.5-1 shows the evolution from DCTL to I$^2$L. Figure 1(a) shows three DCTL elements connected together, 1(b) shows the re-allocation of the current supplying resistors. Re-drawing the dotted portion of 1(b) and replacing the two output transistors with a multi-collector transistor results in 1(c). Because of the large area required for 1(c) resistors, the base resistor is replaced with a PNP current source resulting in 1(d). Connecting the PNP base to the NPN emitter results in the integrated injection/merged transistor concept 1(e). The entire I$^2$L gate can be merged into one fabrication region. Figure 2.5-2 shows the resulting I$^2$L structure.

The I$^2$L gate consists of a lateral P-N-P transistor as a current source and a vertical multicollector N-P-N transistor as an inverter. The term integrated injection derives from the fact that the P-N-P transistor is considered to inject current into the N-P-N transistor and is in fact part of the N-P-N structure and may be common to other N-P-N transistors forming an injector strip as shown in Figure 2.5-3.

2-69



(a)  DIRECT COUPLED TRANSISTOR LOGIC
(b)  RELOCATION OF RESISTORS
(c)  EQUIVALENT SINGLE GATE WITH MULTIPLE
     COLLECTORS
(d)  RESISTOR REPLACED WITH CURRENT SOURCE
(e)  FINAL $I^2L$ CONFIGURATION WITH CURRENT
     SOURCE MERGED WITH INVERTER
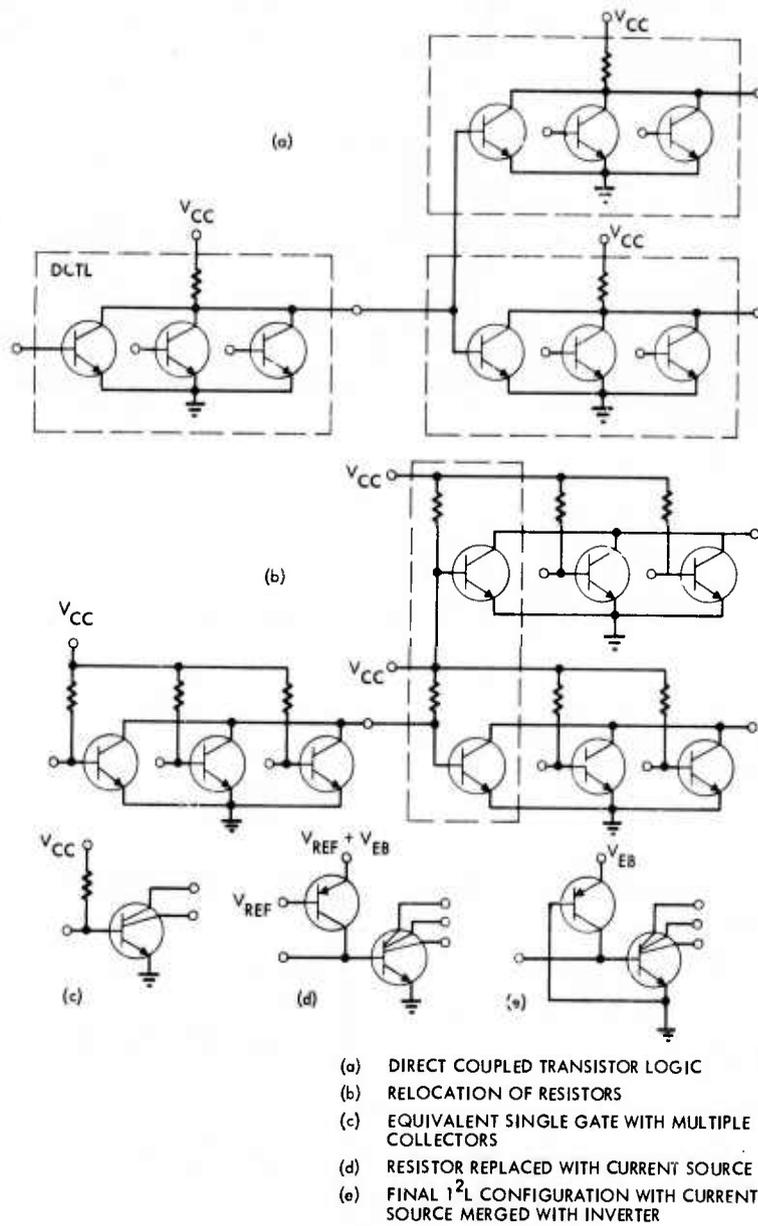
Figure 2.5-1.   Bipolar evolution from DCTL to $I^2L$.
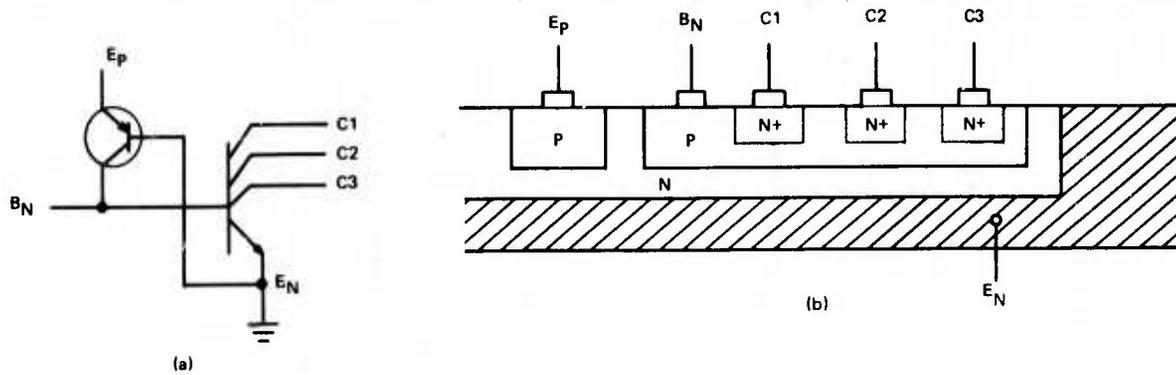
Figure 2.5-2. $I^2L$ gate circuit and structure.
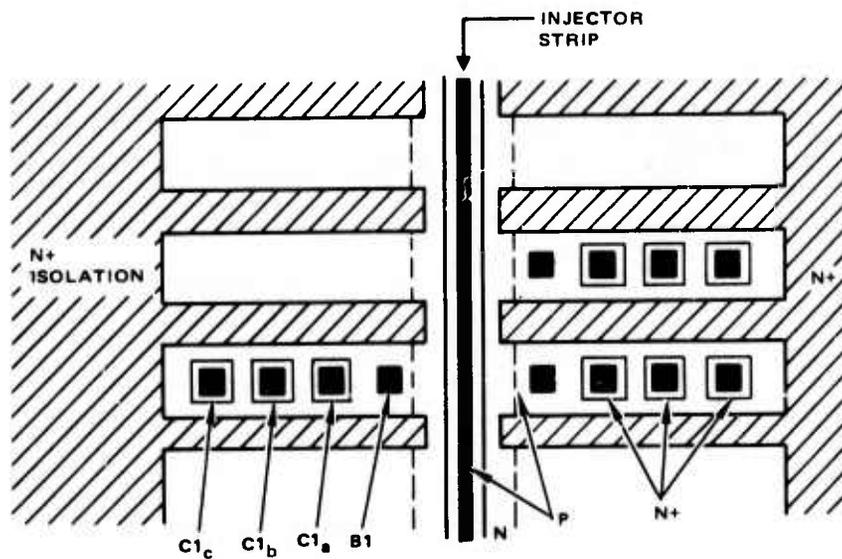


Figure 2.5-3. $I^2L$ layout with injector strip.

### 2.5.2  $I^2L$ Performance Limitations

$I^2L$ is a new digital circuit technique, not a new circuit technology.
$I^2L$ can be manufactured using standard processes and requires only 4 masks
and 2 diffusions, P-MOS being the only device requiring fewer steps.
Because of its unique structure a 4-mil-wide $I^2L$ gate can be constructed on a
5 sq mil area.  Because of the small area the parasitic capacitances are
small.  Low capacitances along with a low logic voltage swing accounts for
the very low power-delay product of $I^2L$.

$I^2L$ has a constant power-delay product for delays from 1 ms through
approximately 100 ns, the propagation delay time being determined pri-
marily by junction and parasitic capacitances.  Minimum propagation delay
is limited by the N-P-N transition frequency $f_T$, which limits the basic $I^2L$
gate to 10-25 ns delay times.  The device limitations inherent to $I^2L$ are a
function of the basic structure.

The standard bipolar fabricating technology used for $I^2L$ necessitates
that the N-P-N transistor be operated in the inverse mode.  The upside down
fabrication has the advantage of automatically isolated collectors and pro-
vides common emitters, but results in poor inverse current gain $\beta_I$, and
poor transition frequency $f_T$.  The poor current gain is caused mainly by
(1) hole injection in the N-type epitaxial layer, which results in low emitter
efficiency, and by (2) base resistance for the multi-collector structure.
Poor $f_T$ is caused by the retarding field in the base and the injected hole
charge in the epitaxial layer.

$I^2L$ devices have been constructed using the basic structure with
power-delay products of 0.25 - 1 pj per gate and minimum delays of
10-25 ns with densities of 250 gates per $mm^2$ with 5 $\mu$m details.

$I^2L$/MTL noise margins can be defined only relative to the magnitude
of the lateral injector current; consequently, the absolute noise margin in
an $I^2L$/MTL circuit is a function of the value of the externally adjustable
injector current.  For a given fixed injector current level, however, both
turn-on and turn-off noise margins may be defined in relation to the circuits
shown in Figure 2.5-4.  As suggested in Figure 2.5-4a, turn-on noise mar-
gin can be defined as the maximum noise current $I_{SN}$ that may be injected

a. Turn-on noise
   margin.

b. Turn-off noise
   margin.

Figure 2.5-4.   Circuits defining $I^2L/MTL$ noise margin.

at a base node without turning on a transistor ($T_2$), which would otherwise
remain turned off.   With the driving transistor ($T_1$) itself driven by current
$I_{BO}$, the maximum collector current which can be absorbed by this tran-
sistor ($T_1$) is given by $\beta_u I_{BO}$.   Thus with transistor ($T_1$), collector current
$I_{BO}$ supplied by the lateral injector associated with the off-transistor ($T_2$)
transistor $T_1$ will absorb additional noise load current up to a level given by
$I_{SNO} = \beta_u I_{BO} - I_{BO}$.   Any component of noise current above this level,
however, will be fed into the base of the off-transistor ($T_2$) where it will be
amplified and will result in a noise output current $I_{CN}$.   Thus for an arbi-
trary $I_{SN}$ noise current level, the output noise signal $I_{CN}$ will be given by

$$I_{CN} = \beta_u(I_{BO} + I_{SN} - \beta_u I_{BO})$$

provided that $I_{SN} \geq (\beta_u - 1)I_{BO}$.

Turn off noise margin may be defined similarly in relation to the
circuit of Figure 2.5-4b where $V_T = kT/q$, k is Boltzman's constant, T is
absolute temp, q is electron charge.   As shown, transistor $T_2$, nominally
turned-on in the absence of noise current will begin to turn off if $I_{SN}$ is
greater than $I_{BO}$.   In this case, the amplified signal noise current will be

given by $I_{CN} = \beta_u (I_{BO} - I_{SN})$. The critical noise voltage corresponding to an allowed output $I_{CN}$ current value is given by

$$V_{NS-CRIT} = V_T \, \ell n \, (\beta_u I_{BO} / I_{CN})$$

$V_T = kT/q$, where $k$ = Boltzman's constant,
$T$ = absolute temperature, $q$ = electron charge

The results above suggested that the $I^2L/MTL$ noise margin at a given injector current level is probably relatively smaller than that exhibited by conventional bipolar saturated logic families. The impedance levels within an $I^2L/MTL$ circuit are also lower so that $I^2L/MTL$ noise margins are adequate to ensure proper operation in the presence of on-chip noise. Moreover, the results above also suggest that the absolute $I^2L/MTL$ noise margin will be increased as a function of the externally controlled injector current level. Noise currents (defined as the difference between nominal and actual injector current levels) will also increase when injection levels are increased due to corresponding increased voltage drops in the distributed injector emitter network. This limitation can be overcome by increasing the size of some of the injectors as appropriate, in an LSI array. Such injector size increases might also be used in any event to provide greater noise margin in I/O interface circuits. Thus $I^2L$ might exhibit less noise immunity than bipolar saturated logic families, but geometry corrections may offset this advantage.

An interesting variation of $I^2L$ known as substrate fed logic (SFL) has been demonstrated by the Plessey Company Limited.[2.5-3] SFL provides improved density and power-delay product at the expense of process complexity. Substrate fed logic is configured as a vertical N-P-N transistor above a vertical P-N-P injector transistor as shown in Figure 2.5-5. The P type substrate is the P-N-P emitter and is connected to the positive supply. The N epitaxial layer is the P-N-P base and N-P-N emitter and is grounded. The resistance of the epitaxial layer is reduced by a mesh like deep N+ diffusion. As a result the entire surface is available for logic interconnection. In addition, it is proposed that the base contact could be replaced with multiple Schottky barrier diodes, thus providing multiple input and output devices as shown in Figure 2.5-6.
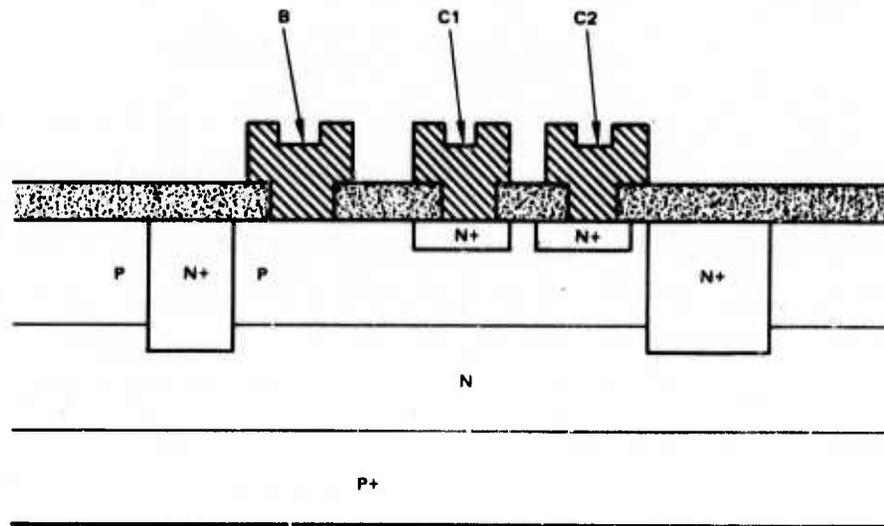
Figure 2.5-5.  SFL structure.
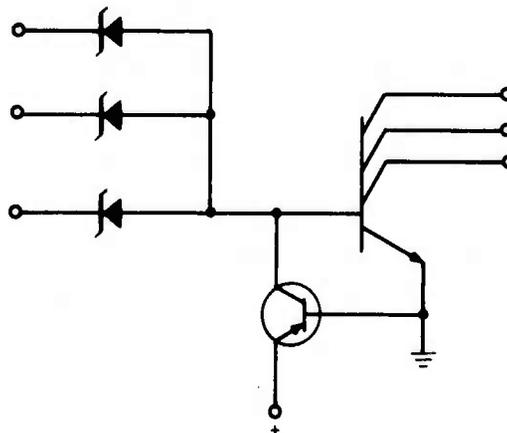


Figure 2.5-6.  SFL gate circuit with
Schottky barrier input diodes.

An experimental device was fabricated without Schottky input diodes and displayed a constant power-delay product of less than 0.05 pj with delays of 110 ns, with a minimum delay of $\approx$50 ns per gate. SFL represents a significant increase in fabrication complexity but the improvement in density and performance is apparent.

The use of Schottky diodes has been proposed in various other configurations. The use of Schottky diodes improves the electrical characteristics of $I^2L$ by reducing logic voltage swings. The power-delay product is proportional to $CV^2$ in the capacitance limited range. Reducing $\Delta V$ increases speed for the same power. In standard $I^2L$

$$\Delta V = V_{BE(ON)} - V_{CE(SAT)} \approx 0.75 - 0 = 0.75 \text{ V}$$

but for Schottky $I^2L$

$$\Delta V = V_{BE(ON)} - \left[V_{CE(SAT)} + V_{S(ON)}\right] \approx 0.75 - \left[0 + 0.45\right] = 0.3 \text{ V}$$

By using Schottky diodes to decouple the output, a single collector can be used which reduces the high inverse gain requirement and eliminates base resistance and current distribution problems. Schottky diode transistor logic (SDTL) can be implemented as Schottky diode decoupling at the output (Figure 2.5-7a) of the ohmic input contacts (Figure 2.5-7b). Logically and electrically it is the same and requires the same area for a given fan-in or fan-out.

The use of another Schottky junction across the collector and base of the inverter transistor has been proposed and called complementary constant current logic, $C^3L$.[2.5-4] Different barrier heights are required since the logic swing is the difference between the Schottky clamp's forward voltage and the decoupling diode's forward voltage. The proposed device uses titanium for the decoupling diodes and the standard combination of platinum
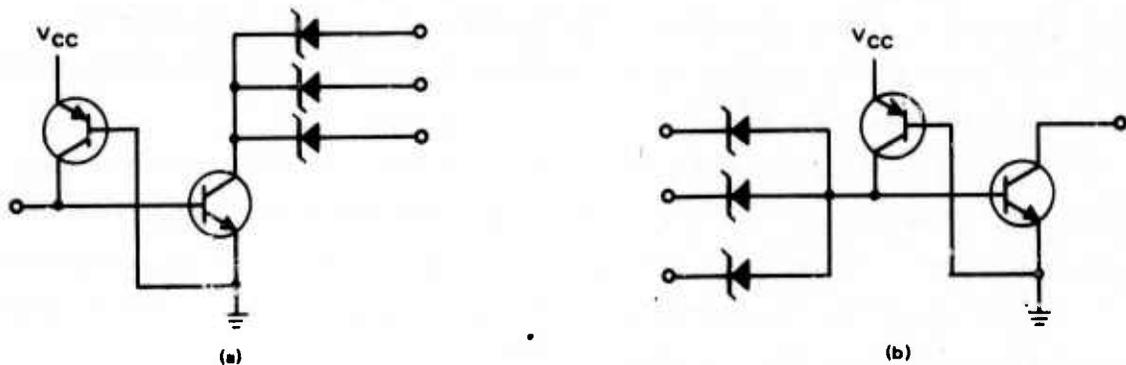
Figure 2.5-7.   SDTL gate circuits with Schottky diodes at
output (a) and input (b).

silicon for the clamp.   Because of the fabrication process the PNP supply
transistor and NPN inverter are physically separate resulting in a lower
density.   Figure 2.5-8 show the $C^3L$ gate schematically.

Schottky transistor logic (STL) using a PNM (M = metal) Schottky
transistor and Schottky input diodes has been proposed.[2.5-5]   Because of the
advanced fabrication technology required,  an experimental device has not
yet been constructed,  but a discrete device simulation was constructed
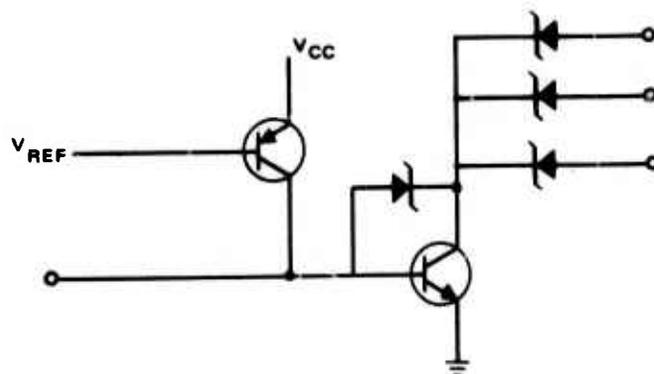


Figure 2.5-8.   $C^3L$ gate circuit.

which showed a 9 ns reduction in gate delay at 100 μA/gate due to the Schottky transistor. Figure 2.5-9a shows the resulting gate, and 2.5-9b the fundamental structures.

## 2.5.3 $I^2L$ Status and Projections

Table 2.5-1 shows a comparison of the existing $I^2L$ structures and present capabilities along with the advantages and disadvantages of each. It should be noted that the best power-delay products are achieved at delays of 100 ns or more and the highest speeds shown require up to an order of magnitude increase in power-delay product. It is also important to consider that much of the experimental data available in the literature quotes power-delay products and maximum speeds based on ring oscillators which are usually single collector gates operating in an optimum situation. Normal combinational logic with multiple collectors and interconnects would be considerably slower.

The future improvements in $I^2L$ will be influenced mainly by advances in process technologies. Ion implantation can be applied to design a bipolar structure more adequate for $I^2L$ circuits, providing higher frequency response and gain. Oxide isolation instead of $N^+$ isolation can be used to reduce parasitic capacitances.

Electron beam technology will soon provide a factor of 5 or more resolution improvement. It is reasonable to assume that $I^2L$ will reach power-delay products of 0.001 - 0.01 pj and speeds in the 1-2 ns range within the next decade, as a result of E-Beam technology and advanced processes.

$I^2L$ is at present the highest density, lowest power-delay product bipolar digital logic circuit available and can be expected to dominate the bipolar LSI field.
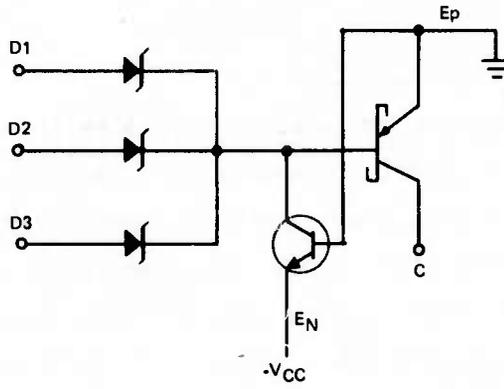
a.
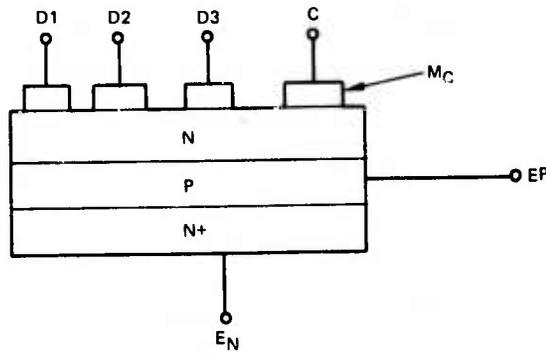


b.

Figure 2.5-9.  STL gate circuit and structure.

TABLE 2.5-1.  I²L VARIATIONS

| | Power-Delay Product | Speed (Min Delay) | Density | Advantages | Disadvantages |
|---|---|---|---|---|---|
| Standard I²L/MTL (Integrated Injection Logic) (Merged Transistor Logic) | 0.1-2 pJ | 10 ns | 250 gates/mm² w/5 μm process | Simple easy process | Limited speed |
| SFL (Substrate Fed Logic) | 0.01-1 pJ | 10 ns | 400 gates/mm² w/5 μm process | Higher density | Limited speed |
| SDTL (Schottky Diode Transistor Logic) | 0.5-2 pJ | 1 ns | 250 gates/mm² w/5 μm process | Higher speed | More complex process |
| C³L (Complementary Constant Current Logic) | 1-2 pJ | 2 ns | 100 gates/mm² w/5 μm process | Higher speed | Lower density more complex process |
| STL (Schottky Transistor Logic) | 0.05-1 pJ | 1-2 ns | 250 gates/mm² w/5 μm process | Higher speed | More complex process |

2-79

## 2.5 REFERENCES

2.5-1 H. H. Berger and S. K. Wiedmann, "Merged-Transistor (MTL)-A Low-Cost Bipolar Logic Concept," IEEE J. Solid-State Circuits, Vol SC-7, 1972, pp 340-346

2.5-2 K. Hart and A. Slob, "Integrated Injection Logic: A New Approach to LSI," IEEE J. Solid-State Circuits, Vol SC-7, 1972, pp 346-351

2.5-3 V. Blatt, L. W. Kennedy, P. S. Walsh, and R. C. A. Ashford, "Substrate Fed Logic-An Improved Form of Injection Logic," Proceeding IEEE Ed Meeting, pp 511-514, Dec. 1974

2.5-4 A. W. Peltier, "A New Approach to Bipolar LSI: $C^3L$," ISSCC Digest of Technical Papers 1975, pp 168-169

2.5-5 H. H. Berger and S. K. Wiedman, "Schottky Transistor Logic," ISSCC Digest of Technical Paper 1975, pp 172-173

2.5-6 N. C. DeTroye, "Integrated Injection Logic - Present and Future," IEEE Journal of Solid-State Circuits, SC-9, No. 5, Oct 1974, pp 206-211

2.5-7 H. H. Berger and S. K. Wiedmann, "The Bipolar LSI Breakthrough, Part 1: Rethinking the Problem," Electronics, pp 89-95, Sept 4, 1975

2.5-8 H. H. Berger and S. K. Wiedman, "The Bipolar LSI Breakthrough, Part 2: Extending the Limits," Electronics, pp 99-103, Oct. 2, 1975

## 2.6 CMOS TECHNOLOGY

As discussed in paragraph 2.4.2, CMOS combines both n- and p-channel MOS devices to form logic inverter stages. The n-channel device has a p-channel device as its load and vice versa. The p-channel MOST can actively pull up in the "1" output and becomes an almost infinite resistance for the logic "0" output. One transistor is off when the other is on, allowing for low power dissipation in either state. Since both devices are operating as common source amplifiers the voltage gain in the active region (during transition) is high. Also, the time to switch from "1" to "0" is the same time it takes to switch from "0" to "1", giving CMOS a symmetry and generally higher speed than either PMOS or NMOS.

Since p-channel and n-channel devices require different substrates, a special process involving additional steps had to be developed to construct CMOS devices. The basic wafer or substrate is an n-type material. A p-type material is then deeply diffused into this substrate forming a p-type well. Heavily doped n+ regions are diffused into the p-well forming an n-channel device and similarly doped p+ regions diffused into the substrate forming a p-channel device. This process is illustrated in Figure 2.6-1.

The basic CMOST (CMOS Transistor) logic circuit is the inverter, which uses a complementary set of devices with the gates tied together. Other logic functions built on this basic concept are shown in Figure 2.6-2.
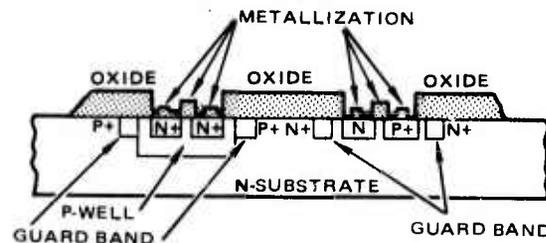
Figure 2.6.1. CMOS structure.

2-81
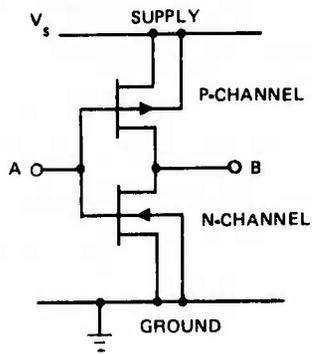
a. INVERTER, B = $\overline{A}$
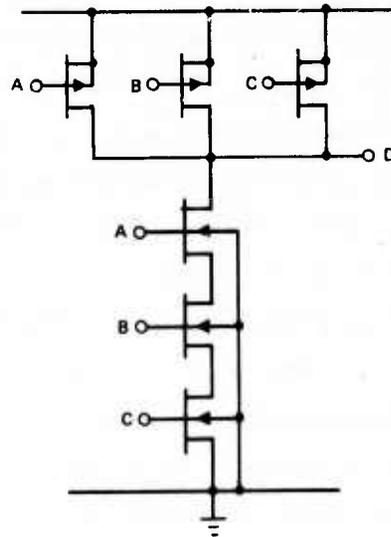
b. NAND, D = $\overline{ABC}$
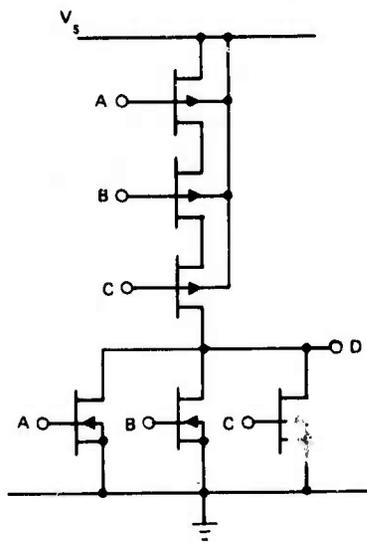
c. NOR, D = $\overline{A+B+C}$
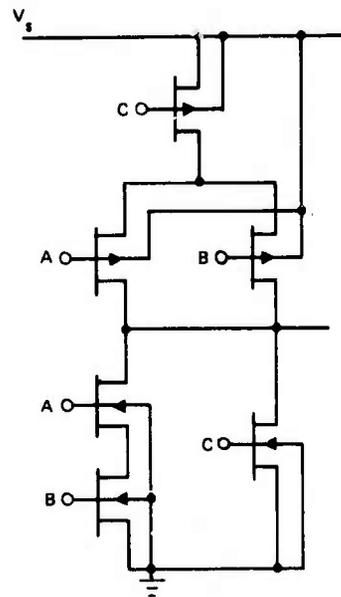
d. COMPLEX, D = $\overline{AB+C}$

Figure 2.6-2. CMOS logic gates.

### 2.6.1 Primary CMOS Logic Considerations

In a CMOS gate during a logic transition the devices that are on and those that are off can each be modeled as one equivalent device. Also, from Swanson, [2.6-1] it is shown that series MOS transistors can be modeled as one equivalent device whose channel width is the sum of the parallel device widths. Thus each CMOS logic gate can be replaced by an equivalent inverter. It is this equivalent device that will be the basis for analysis of CMOS logic. Mathematically since the gain constant of an n-channel device is

$$K = \frac{Z_n}{L_n} \mu_n C_{ox}$$

where

$Z_n$ = channel width

$L_n$ = channel length

$\bar{\mu}_n$ = n channel mobility

$C_{ox}$ = oxide capacitance

then two devices in series have a gain constant of

$$K = \frac{1}{2}\left[\frac{Z_n}{L_n} \bar{\mu}_n C_{ox}\right]$$

and for two in parallel

$$K = 2\left[\frac{Z_n}{L_n} \bar{\mu}_n C_{ox}\right]$$

The same analysis holds for p-channel MOS.

2-83

A theoretical development of the speed-power product can be realized at the gate level using the equivalent inverter. Entire logic networks can be reduced and analyzed as gates in a similar manner. The equation defining the speed-power product for CMOS is developed by Swanson and for two complete (positive and negative) transitions with rise time $\tau_{R1}$ and fall time $\tau_{F1}$ (at the output) is

$$(P = E_{PT} f)$$

$$E_{PT} = E_{RT} + E_{FT}$$

$$= C_L V_S^2 \left[ 1 + \frac{\beta^5}{30} \left( \frac{\tau_{R1}^2}{\tau_n \tau_p} A_\rho + \frac{\tau_{F1}^2}{\tau_n \tau_p} A_n \right) \right] \qquad (2\text{-}1)$$

where $E_{RT}$ and $E_{FT}$ are the individual energies for rise and fall times equal to the total energy (speed-power product) $E_{PT}$

$C_L$ = load capacitance on the equivalent inverter

$V_S$ = power supply voltage

$\tau_n + \tau_\rho$ are the transit times of the n and p-channel devices

$$\tau_n = \frac{2 C_L}{K_N V_S}$$

$$\tau_\rho = \frac{2 C_L}{K_P V_S}$$

$$\beta = 1 - \alpha_n - \rho$$

where $\alpha$ = relative threshold voltage

$$\alpha = \frac{V_T}{V_S}$$

$$A_n = 1 - \left(\frac{1}{7} \frac{\tau_{F1}}{\tau_n} \beta^2\right) - \frac{5}{84}\left(\frac{\tau_{F1}}{\tau_n} \beta^2\right)^2$$

$$A_\rho = 1 - \frac{1}{7}\left(\frac{\tau_{R1}}{\tau_\rho} \beta^2\right) - \frac{5}{84}\left(\frac{\tau_{R1}}{\tau_\rho} \rho^2\right)^2$$

$$\tau_{F_1} = \frac{1.25\tau_n}{1-\alpha_n^2}\left\{1 + (1-\alpha_n)\left[\ell n\left(\frac{1-\alpha_n}{0.2}\right) - 1\right]\right\}$$

$$\tau_{R_1} = \frac{1.25\tau}{(1-\alpha_\rho)^2}\left\{1 + (1-\alpha_\rho)\left[\ell n\left(\frac{1-\alpha_\rho}{0.2}\right) - 1\right]\right\}$$

Thus, the entire equation can be reduced (through numerous substitutions) to one in terms of $C_I$, $K_N$, $K_P$, $V_S$, and $V_T$. Where $K_N$ and $K_P$ are dependent on the actual physical device parameters $\bar{\mu}$, $C_{ox}$, channel length, and channel width. This type of equation can certainly be written into a FORTRAN program, and a fairly straightforward power-speed design process realized.

The calculation of the load capacitance term in this equation also relates directly to the fanout capability of the equivalent MOS inverter. This in itself is an important design consideration. As $C_L$ increases, the speed-power product increases. Figure 2.6-3 illustrates the capacitances that contribute to the load capacitance term. $C_{jp}$ and $C_{jn}$ are the p and n channel drain-junction to substrate capacitance and $C_W$ in the wiring (metalization interconnect) capacitance. The development of the load capacitance equation is fairly straightforward except for the fact that during a negative output transition the p-channel devices connected to the output node
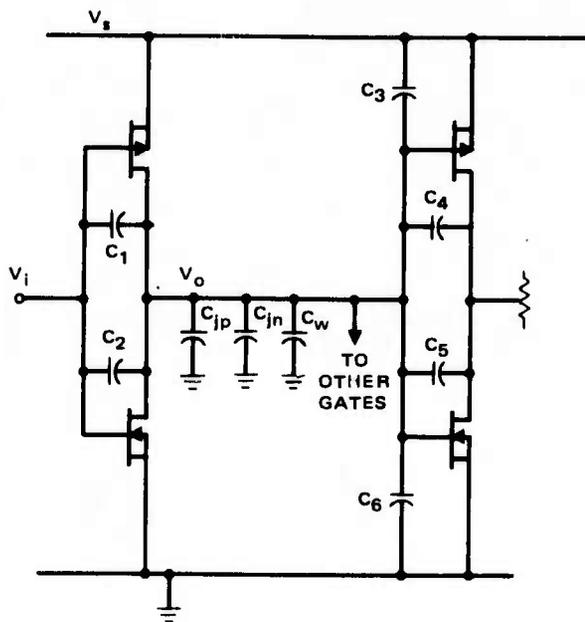
Figure 2.6-3. Various capacitances
connected to the output node of an
equivalent CMOS inverter.

are in their saturation region and the n-channel MOST's are in their linear
region  and vice versa for positive transitions.   Since different capacitances
appear in these two regions the load capacitance must be separated into two
distinct values for + and - going transitions.   Referring to these as $C_L^+$ and
$C_L^-$ the overall load capacitance equation is

$$C_L^\pm = C_{jp} + C_{jn} + C_W + (C_1 + C_2)_{eq}^\pm + \sum_i (C_{4i} + C_{5i})_{eq}^\pm$$

where the summation is over i, the number of gates to which the output is
connected.   Also $(C_1 + C_2)_{eq}^\pm$ and $(C_{4i} + C_{5i})_{eq}^\pm$ are equivalent capacitances
dependent on delay,  rise and fall times associated with the respective
capacitances ($C_1$, $C_2$, $C_{4i}$, $C_{5i}$) during + and - going transitions.   This is
because $C_1$, $C_2$, $C_4$ and $C_5$ are connected to time varying voltage nodes and
represent the Miller feedback effect.

From the previous discussion it can now be seen that equation (2-1) is extremely useful as a design or analysis tool for CMOS logic circuits. The equation basically finds the speed-power product of an equivalent CMOS logic gate, which in itself is extremely important. But on the way to this result other considerations are found. The load capacitance indicates fanout capabilities and limitations through the analysis discussion above. Circuit packing density is also taken into account as the transit times $\tau_n$ and $\tau_p$ are inversely proportional to the n and p channel gain constants $K_n + K_p$. These in turn are directly related to device channel width and length, the limiting factors in MOS chip real estate. Some design considerations regarding circuit power supply levels are related through the $V_s$ term in equation (2-1) as well.

Equation 2-1 would indicate that with sufficiently small supply voltages and geometries, the fundamental logic limitations for power-delay can be overcome. Such devices require extremely small geometries. Swanson derives further restrictions on small MOS logic through a two dimensional analysis, the results of which are shown in Figure 2.6-6. Noise considerations in CMOS circuits as in other logic families are fairly complex. 1/f considerations are developed for MOS transistors by Backensto[2.6-2] and can be extended to CMOS devices. Of prime importance at higher device operating frequencies above the 1/f corner is thermal noise defined as

$$v_{Th} = \left( \frac{8}{3} \frac{kT}{gm} \right)^{1/2}$$

where

$$g_m = \left( 2 \frac{W}{L} \bar{\mu}_p C_{ox} I_D \right)^{1/2}. \tag{2-2}$$

From equation (2-1) it is noted that shorter channel lengths are needed for better power-speed products. This will keep device thermal noise low. However, of prime importance in digital logic circuits is external noise

2-87

sources and how much of this can appear on the input of an inverter without flipping the output from one state to another. The noise immunity of a logic circuit family is strongly dependent on its speed. Noise is capacitively coupled on to signal lines, thus slower logic response means better noise immunity. CMOS typically rejects voltage noise pulses up to 45 percent of the power supply voltage and 30 percent is guaranteed in commercial devices throughout the industry. 2.6-3

Since signal processing in proposed infrared sensor systems involves MOS-CCD multiplexing and A/D schemes, CMOS logic gives an added option of having the logic compatible with the signal processing that comes before it.

Numerous design and analysis techniques enable CMOS to be incorporated well into LSI circuits. Swanson's equivalent inverter and scaling techniques discussed by Keyes 2.6-4 are two methods towards intelligent computer aided techniques. Integrated circuit analysis programs such as SPICE are well developed and directly applicable to CMOS equivalent circuits. A Hughes developed program called ANYMOS specifically accommodates design parameters encountered in CMOS logic.

## 2.6.2 CMOS Status and Projections

Present day CMOS logic devices are well established in the commercial market. Virtually all of the functions obtainable by standard TTL logic are now available in CMOS. However custom designs must be implemented to obtain the highest levels of performance.

Most commercial CMOS devices are constructed on a bulk silicon substrate using the standard metal-oxide-semiconductor processing techniques. The various commercial processes are optimized with regards to yield, speed, power, and density. The limits are not necessarily optimum due to the heavy influence of cost. For a typical CMOS inverter operating at 5 volts, ambient temperature and a load capacitance of 25 pf the power-speed product is about 800 pJ. 2.6-7 This number is not a good figure of merit for LSI circuits since interconnect capacitances are very much lower. However, it does give some feel as to where discrete commercial CMOS is and the significance of large scale integration.

Of particular importance towards the implementation of high speed-low power circuits is the advent of an insulated substrate CMOS process using sapphire. The CMOS/SOS (SOS for silicon on sapphire) greatly reduces the junction to substrate capacitances involved in the input gate capacitance. This allows for an approximate 3:1 reduction in power-speed product over bulk silicon substrates since

$$E_{PT} = \text{power} \times \text{speed} \sim \frac{p}{f} \sim C_L V_S^2.$$

This reduction refers to LSI circuits with minimum geometry devices. In addition, polysilicon (self-aligned) gates may be used to reduce gate overlap capacitance and lower threshold voltage. This allows higher fanout, lower power dissipation, and higher speed. The SOS process is discussed further in paragraph 4.3.

Future trends in CMOS seem to point in certain definite directions. Certainly CMOS/SOS and polysilicon gate techniques will be pursued further for speed-power product reduction. Because the sapphire process is somewhat expensive, some performance tradeoffs may have to made, though, and bulk silicon CMOS will continue in the commercial market.

Shorter channel lengths and reduction of capacity in the channel are certain to come as experience in dealing with small geometry devices grows. Hughes Newport Beach has constructed CMOS devices with 0.2 mil p-channel and 0.3 mil n-channel lengths as well as experimental devices down to 0.1 mil in channel length. Capacitance values are projected to reduce from 0.2 to 0.1 pf/mil$^2$. These two factors in themselves combine for good future speed-power reductions.

An interesting topic that could well prove advantageous is the concept of ion-implanted, buried channel CMOS. Work has been done to characterize buried channel MOSFET's[2.6-9] and to some extent CMOS,[2.6-10] and further investigation seems worthwhile.

A brief explanation of the argument for buried channel CMOS is appropriate. Today, CMOS logic is operated in or near weak inversion (see Figure 2.6-4) in order to maintain low power. However, operation in weak

inversion causes a small effective mobility, as $I_D$ is decreased, in the channel, thus limiting the speed-power product. For a buried channel device weak inversion is essentially a meaningless term, as the inversion is already present (see energy band diagram, Figure 2.6-5). Thus, attempting to better the speed-power product by lowering $I_D$ does not decrease the effective mobility as much as in a surface channel device. The higher mobility in a buried channel device decreases the speed-power product. In addition ion implantation techniques shift the threshold voltage such that the power-speed product is reduced further.

Improved photolithographic techniques are necessary to achieve the high density, lower power-speed devices expected in the 1980's. The increased accuracy obtained by improving this technique would now allow CMOS to approach the ultimate inverter limitations derived by Swanson. These are plotted in Figure 2.6-6. The following conditions are necessary to obtain the best possible performance:

$$L = 500 \text{ Å}$$

$$V_S = 0.1V$$

$$T_{PD} \text{ (pair delay)} = 7.5 \text{ psec}$$

$$\text{Power} = 20 \text{ nW}$$

$$t_{ox} = 100 \text{ Å}$$

with these conditions $E_{PT} = 1.5 \times 10^{-19} J$. A low voltage ring oscillator was fabricated by Swanson using ion implantation techniques and achieved a speed power product of 0.08 pJ with a supply voltage of 0.4V. These numbers approximately reflect future trends of CMOS.

One last point with regards to the use of CMOS logic in space sensor systems is radiation hardening. The Hughes' Newport Beach facility is extensively involved with radiation hardening and testing of CMOS. By ADSP time frame considerable valuable experience and process knowledge in this area will have been obtained and could be used to the benefit of APSP. Since APSP operates in a benign environment, (typically at 104 rads, only a few tenths of a volt threshold shift occurs) this knowledge is not critical, but would be useful in reducing supply voltages below approximately 1-1/2 to 2 volts. MOS devices are majority carrier technologies and do not (as bipolar
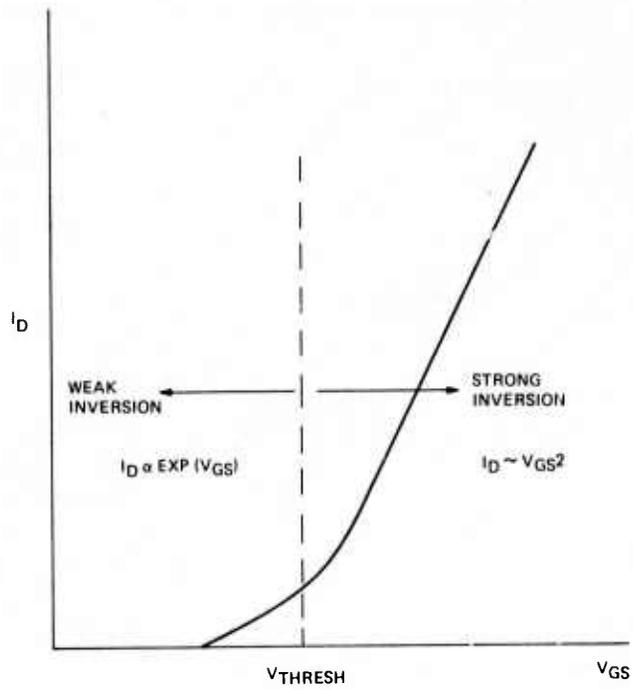
Figure 2.6-4. Inversion regions in a MOSFET.
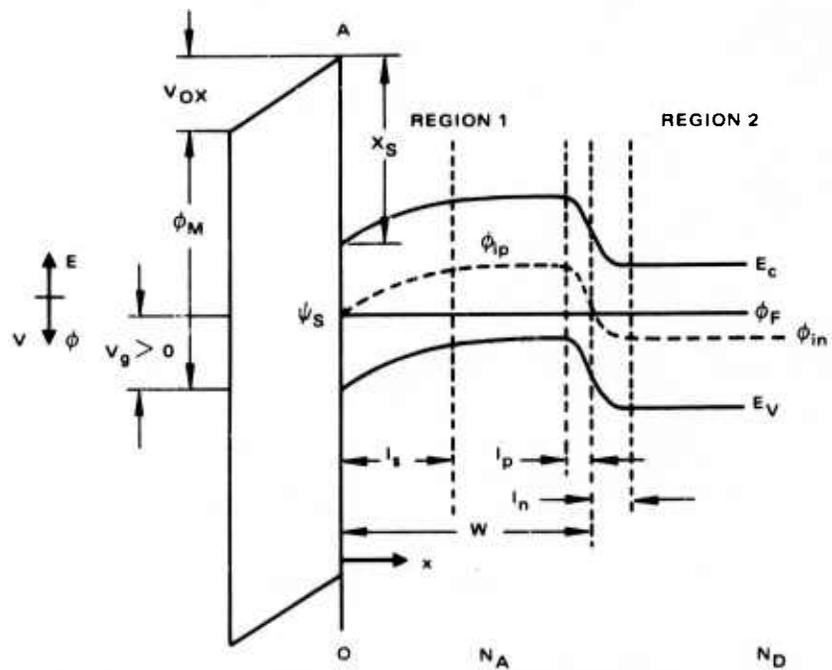


Figure 2.6-5. Band diagram of MOST structure with implanted layer $N_A$ beneath the substrate ($N_A > N_D$).

2-91
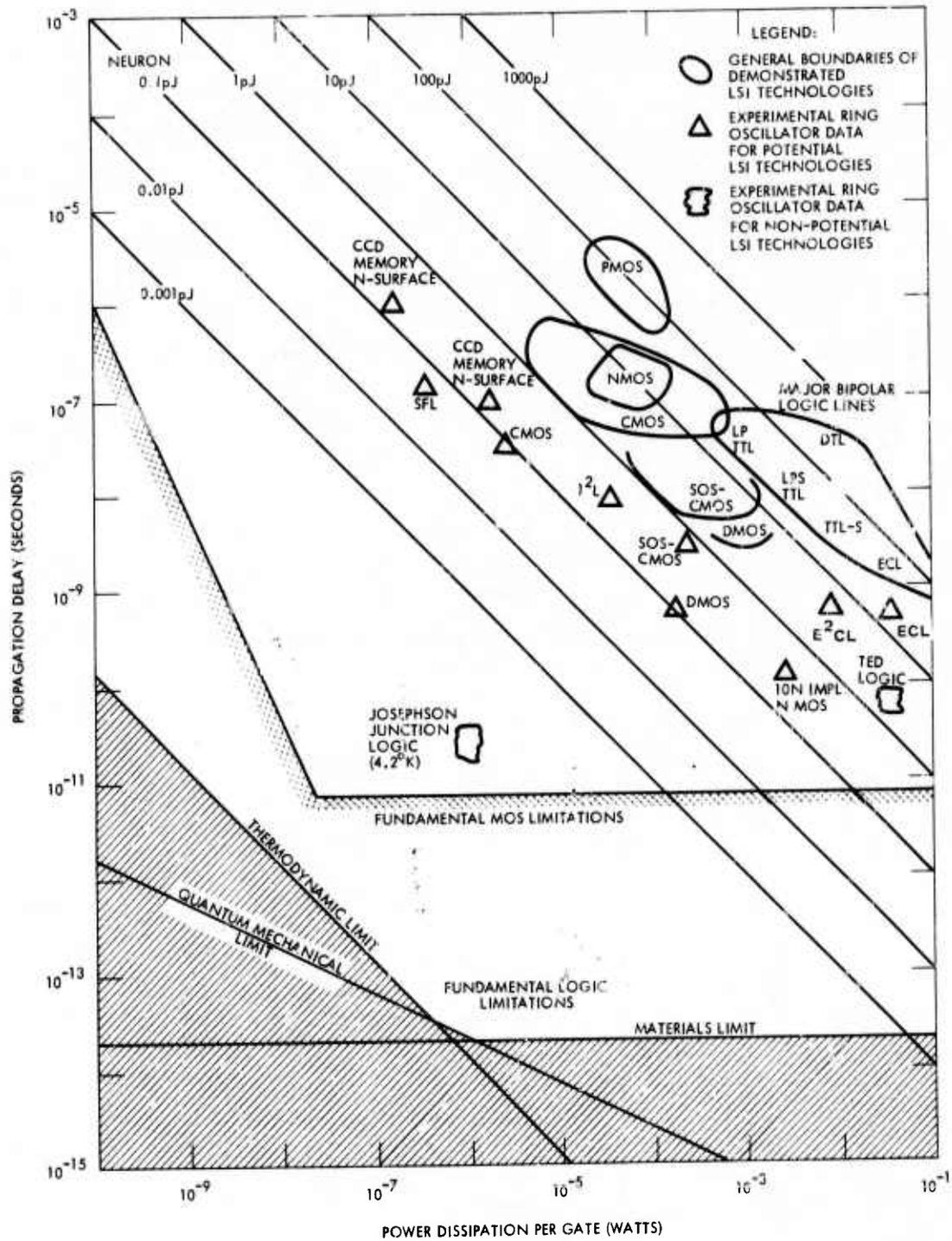
Figure 2.6-6.  Maximum possible performance of CMOS inverters.

$I^2L$) exhibit the loss in gain due to increased recombination rates caused by total $\gamma$ radiation dose. This effect in bipolar base regions degrades $\beta$.

## 2.6 REFERENCES

2.6-1    Swanson, R.M., "Complimentary MOS Transistors in Micropower Circuits," Stanford University Technical Report No. SU-SEL-074-055 prepared for Army Electronics Command, National Institutes of Health. December 1974

2.6-2    Backensto, W.V. and C.R. Viswanathan, "An Improved 1/f Noise Model of an MOS Transistor," IEDM Proceedings 1975.

2.6-3    Motorola "Mc Mos Handbook," Motorola, Inc., 1974 p. 3-26 to 3-35.

2.6-4    Keyes, R.W., "Physical Limits in Digital Electronics," Proceedings of the IEEE, Vol. 63, Nov. 5, May, 1975 Pp. 740-767.

2.6-5    "370 SPICE User's Manual," Hughes Aircraft Company Internal Publication January, 1975.

2.6-6    Ronen, et al, "Recent SOS Technology, Advances and Applications," Solid State Technology, August 1975, pp. 39-46.

2.6-7    Motorola, p. 3-5.

2.6-8    Sze, S.M., Physics of Semiconductor Devices, Wily-Interscience, New York, 1969 Pp. 425-433.

2.6-9    Backensto, W.V. and R.H. Genoud, "Buried Channel MOSFET Model Application to HgCdTe-CCD Buffer Preamp," Hughes Aircraft Interdepartmental Correspondence No. 27721.2/109, 12 Feb. 1974.

2.6-10   Swanson, p. 159-186.

2.6-11   Erb, D.M. and H. Nummedal, "Buried Channel Charge Coupled Devices for Infrared Applications," CCD Applications Conference 18-20 September, 1973, San Diego, Ca. Pp. 157-167.

## 2.7 DMOS

The DMOS technology not only multiplies MOS speed but opens up the entire realm of microwave applications to MOS technology. It involves a two-stage diffusion through a single mask opening, permitting channels of 1-micron lengths to be formed simply and inexpensively.

The result: discrete microwave transistors that exhibit a 10-gigahertz maximum frequency of oscillation, a 7-decibel gain at 2 GHz, and a noise figure of 0.5 dB at 1 GHz - performance usually associated only with bipolar devices. DMOS devices can be switched with subnanosecond speeds, and have the added advantage of high breakdown voltage.

Microwave FETs could only be obtained with a channel length of about 1 micron, which requires 1 micron metal gate widths. But to manufacture devices with such narrow gates it has been necessary to use sophisticated, highly accurate photomasking techniques. Moreover, for digital ICs, high speed is also dependent on highly controlled doping and diffusion steps. The commercial DMOS process, on the other hand, achieves its micron-long channels with metal widths and oxide openings no smaller than 8 microns, demanding photomasking and diffusion tolerances no stricter than those presently used in conventional bipolar IC technology. [2.7-1]

The high performance of DMOS devices results directly from the method of forming the channel as discussed in 2.4.2. In general, the frequency response, or speed, of any MOS transistor is determined primarily by channel length and parasitic capacitance, and improves as they become smaller. Reducing length cuts the transit time for carriers traveling between source and drain, while reducing capacitance decreases the charging time. (In an MOS device, parasitic feedback capacitance exists between gate and drain, $C_{gd}$ as well as between gate and source, $C_{gs}$.)

In the usual MOS devices, unfortunately, a short channel length usually entails large parasitic capacitance, because the separation between source and drain determines the amount of lateral diffusion under the gate for a given L, and the lateral diffused region, which is also highly doped, represents large $C_{gs}$ or $C_{gd}$. These parasitics can be minimized by ion implantation and polysilicon gate processes, which are self-aligning and reduce the overlap between the gate region and the source and drain regions.

The DMOS process eliminates these problems, and results in a device which has a precisely controlled channel length of less than 1 micron, minimal $C_{gs}$, very small feedback capacitance $C_{gd}$, and no restriction on maximum drain breakdown voltage. As Figure 2.7-1 illustrates, the DMOS channel is a narrow region, sandwiched between two opposite-type regions and created by the sequential diffusion of two opposite-type regions and created by the sequential diffusion of two opposite-type dopant impurities under a single mask edge in the source region. Once this edge is formed, the
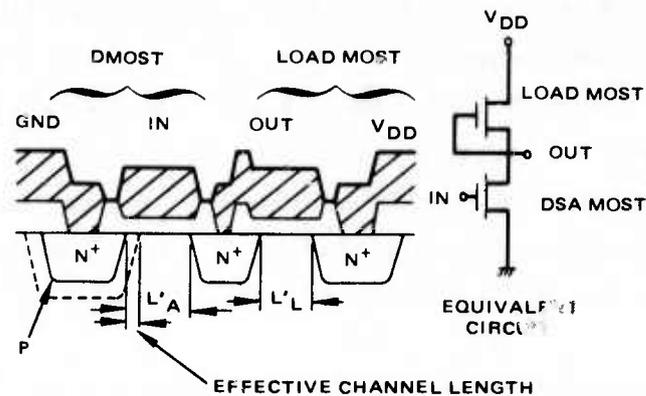
Figure 2.7-1. N channel DMOS with
N channel depletion load forming LSI
inverter stage.

critical distance L is a function only of the diffusion schedule-masking,
exposure and etching errors are eliminated-and since one diffusion edge
follows the other, L can be controlled in much the same way as is the base
width of a bipolar transistor.

Lateral diffusion also controls channel length in the conventional
process. But there it leads to large overlap parasitic capacitance when
narrow channels are required. For the DMOS process, the diffusion lengths
are much smaller, even for very narrow channels, and because overlap
capacitance only exists on the source side for DMOS, it can be kept to a
minimum.

Channel lengths in the range of 0.4 to 2 microns are easily achieved
in DMOS processing, even when the relatively noncritical bipolar diffusion
schedules are used. In contrast, because the variations in mask quality,
exposure and etching can affect channel length by 1 micron or more, typical
MOS transistors in production today are fabricated with 5-micron lengths,
and for this reason are unable to compete in speed with bipolar devices.

Integrating a DMOST driver with a depletion MOS load transistor
provides an exceptionally high speed and low power (by virtue of the low
supply voltage) LSI gate structure (see Figure 2.7-1). [2.7-2] DMOS has a
high gain factor due to the short effective channel length. The threshold

voltage of the DMOS can be precisely controlled using self aligned diffusions. Drain capacitance can presently be made - 1/4 times as low as that of conventional NMOS. The depletion load has efficient driving capability due to its constant current characteristics.

A 0.65 nsec propagation delay time for an integrated ring oscillator circuit has recently been demonstrated, [2.7-2] with a power-delay product of 0.10 pJ using a 2 volt supply voltage. A 510 gate/mm$^2$ density was obtained.

In the same work a 4-bit arithmetic logic unit (ALU) with 48 operations was developed and obtained 2.9 nsec/gate and 2.0 pJ power-delay with 141 gates/mm$^2$ operating with a 5 volt power supply.

Table 2.7-1 is taken from reference 2.7-2. It shows a direct comparison between DMOS, ELC and T$^2$L for similar ALU LSI functions. DMOS is superior in speed-power product and packing density and comparable to ECL in speed.

### TABLE 2.7-1. COMPARATIVE ALU CHARACTERISTICS

| | | DMOS | ECL | TTL |
|---|---|---|---|---|
| $t_{pd}$ per gate | (ns) | 2.9 | 1.5 | 6.0 |
| Power dissipation per gate | (mW) | 0.71 | 15 | 8 |
| $p \cdot t_{pd}$ product per gate | (pJ) | 2.1 | 22.5 | 48 |
| transition time (Including buffer) | (ns) | 32 | 6.5 | 24 |
| | | (11 Stages) | (4 Stages) | (4 Stages) |
| Number of logic stages | | 9 ~ 12 | 4 ~ 6 | 4 ~ 6 |
| Number of logic gates | | 115 | 86 | 64 ~ 87 |
| Chip area | (mm$^2$) | 0.8 | 3 | 7.3 |
| Supply voltage | (V) | 5 | -5.2 | 5 |
| Logic swing | (V) | 4 | 0.8 | 3.3 |
| Device count | | 381 | 632 | 143 |

It appears that with future refinements in photolithographic techniques and DMOS on sapphire technology or even Complementary DMOS on sapphire with low threshold voltages (power supplies in the 1 to 2 volt area), another order of magnitude may be obtained in power-delay product with subnanosecond gate delays for full LSI circuits. DMOS and related technologies have a promising future.

## 2.7 REFERENCES

2.7-1 Cauge, T. P, et al, "Double-diffused MOS transistor achieves microwave gain," Electronics, February 15, 1971.

2.7-2 Ohta, K., et al., "A High-Speed Logic LSI Using Diffusion Self-Aligned Enhancement Depletion MOS IC," IEEE Journal of Solid State Circuits, Vol SC-10, No. 5, Oct. 1975.

2.7-3 Hughes Aircraft Co, "Low Cost Real Time Processor for SAR Systems, Phase 1 Report, Contract No. F33615-74-C-1167, May 1975.

# 3.0 ADAPTIVE VIDEO ENCODER TECHNOLOGY

The Adaptive Video Encoder of the APSP accomplishes the interface between the digital Layered Array Processor (LAP) and the analog detector signals from the Monolithic Focal Plan Array (MFPA). Included are the Moving Target Indication Estimator and video temporal prefilters. In addition, the AVE controls detector bias and clock frequencies on the MFPA, based upon peak signal strength, to adaptively optimize dynamic range. Thus the APSP, in addition to needing high speed logic in the LAP, requires A/D converters (8 bit), D/A converters (14 bit, in some feedback systems under consideration), counters, differential amplifiers, memory, and logic elements (gates, adders, multipliers) for the AVE. The basic word rate requirements vary from 16.4K words/sec to 164K words/sec. This section investigates the present state of the art of the most critical of these devices; the A/D and D/A converters. Detailed descriptions of the AVE and other new devices required for APSP implementation are the subject of a forthcoming report: Critical Device Design.

A company funded development currently under way at Hughes is the CRC 100 signal processing CCD chip. The devices on this chip are specifically tailored to A/D, D/A and digital logic requirements similar to those of the AVE. Table 3.0-1 lists the specific devices and systems on the test chip. In general, a variety of functional components make up each integrated CCD A/D system. Connection pads are provided to allow component testing, characterization and optimization in addition to system operation. With its four A/D's and other devices, the chip provides:

- A/D Division Elements - 2 types
- Comparators - 4 types

- Logic Elements                               - 3 types
- D/A Converters                               - 3 types
- Sample and Hold, Peak Detector, etc.         - 4 types
- CMOS                                         - 2 types

The Critical Device Design document will provide a closer examination of the individual circuits. In general, for each A/D system,

### TABLE 3.0-1.  HUGHES CRC-100 CCD TEST CHIP
### (N TYPE SURFACE CHANNEL)

| Device Number | Circuit/Function | Description |
|---|---|---|
| 1 | Forward A/D System | Serial output successive approximation A/D system with charge comparator |
| 2 | Feedback A/D System | Serial output successive approximation A/D system using digital feedback to D/A |
| 3 | CCD Half Adder | Half Adder building block cell which uses a "carry" sense diffusion to set a potential barrier for input 11 logic correction |
| 4 | CCD Full Adder | Small geometry device which uses a charge trap and regeneration of the "carry" output to correct the input 011 logic state |
| 5 | CCD D/A Converter | An 8 bit charge splitting CCD with parallel digital inputs to select the weighted charge packets |
| 6 | Forward Differential A/D system | Serial output successive approximation A/D system with MOS comparator |
| 7 | Multiplying Feedback A/D System | Serial output successive approximation A/D system using analog feedback |
| 8 | Test Devices | CMOS/CCD process compatibility devices |

the expected resolution is 8 bits with an operating speed of about 1 Mbit/sec while dissipating 10m watts or less (including clock power). The design goal is 10M bits/sec, which is the anticipated performance capability for the burried channel version to be finalized in early 1976.

The following sections examine the status of converter technology, followed by a brief discussion of current analog transform technology.

## 3.1 CONVERTERS

The purpose of this section is to summarize the current state of the art in Analog to Digital and Digital to Analog converters. Units with special features are highlighted and then put into perspective by the use of a comparison table. Major emphasis is on total systems, therefore only those subsystem building blocks which offer valuable and unique features are presented.

Linear converters are presented in Tables 3.1-1 and 3.1-2. Although other types have significant advantages in some applications, they are not appropriate for the AVE task. An example is Precision Monolithics' companding D/A that follows standard nonlinear speech compression laws.

One outstanding new device is the Hughes 4 bit monolithic A.D. encoder which has been operated with a 2.5 nsec converter time; by combining four of these devices, a 6 bit word can be generated. This device dissipates 1.4 watts. A second device is a 6 bit monolithic D.A. converter, which converts in 6 nsec and dissipates 0.7 watts. Another Hughes development is the 6 bit, 200 Mword/sec converter which has been demonstrated (it utilizes 205 watts of input power).

A 5 bit MOS monolithic clockless A/D converter has been described.[3.2-1] It utilized portions of a continuously variable threshold device and achieved conversion times of 2 μsec; power dissipation was not reported.

Another unique device described in reference 3.2-2 is an all MOS successive approximation weighted capacitor A/D conversion technique. It performs a 10 bit conversion in 20 μsec. The acquisition time is 25 μsec; thus the conversion rate is 22 KHz.

## TABLE 3.1-1. COMMERCIAL A/D CONVERTERS

| Co. | Model | Bits | Time, μsec | Power | $* |
|---|---|---|---|---|---|
| Computer Labs | 9000 | 13 | 0.1 | | 13,980 |
| Datel | ADC HY12BC | 12 | .8 | 2 w | 79 |
| | EH12B3 | 12 | 2 | 2.325 w | 299 |
| Analog Devices | 1103 | 12 | 3.5 | 5.1 w | 495 |
| Burr-Brown | ADC 85 | 12 | 10 | 0.45 w | 225 |
| | ADC 60 | 12 | 3.5 | 2.85 w | 395 |
| Computer Labs | 9000 | 12 | 0.1 | | 13,980 |
| Analogic | MP 2712 | 12 | <4 | 3.3 w | 229 |
| ILC Data Device Corp. | ADH-10/1 | 12 | 0.8 | | 990 |
| Teledyne | 4129QZ | 12 | 24 | | |
| | 4132 | 12 | 3.5 | | |
| | 4133 | 12 | 2.5 | | |
| Computer Labs | 9000 | 11 | 0.1 | | 8,200 |
| Burr-Brown | ADC 85 | 10 | 6 | | 185 |
| | ADC 60 | 10 | 1.88 | 2.85 w | 395 |
| Ayden Vector | ADH-10 | 10 | 25 | 1.025 w | |
| Analog Devices | 1103 | 10 | 1.2 | 5.1 w | 484 |
| | 1123 | 10 | 65 | 75 μJ/ conversion | 299 |
| ILC Data Device Corp. | ADH-10/1 | 10 | 0.8 | | |
| Teledyne | 4131 | 10 | 1 | | |
| Datel | ADC CM10B | 10 | 310 | 90 | 159 |
| Datel | M10B | 10 | 1 | 3.3 w ±20 ppm/ | 895 |
| | G10B | 10 | 1 | .1 w ±50 ppm/ °C | 349 |
| Datel | VH8B | 8 | 0.2 | 8.3 w | 895 |
| | UH8B | 8 | 0.1 | 8.3 w | 995 |
| Datel | ADC CM8B | 8 | 250 | 90 | 149 |

(Table 3.1-1, concluded)

| Co. | Model | Bits | Time, μsec | Power | $* |
|---|---|---|---|---|---|
| Analog Devices | 1103 | 8 | 1 | 5.1 w | 473 |
| Micro Networks | 5060 | 8 | 100 | 53 mw | 189 |
| | 5065 | 8 | 100 | 53 mw | |
| | | 8 | 1 | 1.125 w | 295 |
| Intech | A-857-8 | 8 | 0.8 | | 199 |
| Analog Devices | AD75705 | 8 | 35 | | |
| Hughes | | 6 | 0.005 | 205 w | |
| Hughes | | 4 | 0.0025 | 1.4 w | |

*Included as an indicator of relative complexity

At the same conference, (ref 3.2-2), R. B. Craven presented a bipolar LSI, 12 bit D/A converter consisting of a 97 x 180 mil Si-Cr resistor network, and a 79 x 179 chip of active circuitry; power dissipation was not reported.

## 3.2 ANALOG TRANSFORM TECHNOLOGY

The advent and development of CCD recursive and non-recursive (transversal) filter technology opens the door to a wide variety of matched filtering and analog correlation signal processing previously not available for analog design. This section briefly reviews the current status of CCD transversal filters, followed by an example involving Walsh-Hadamard Transforms using transversal filters.

### 3.2.1 CCD Transversal Filter Status

CCD Cross-correlators[3.2-4] provide a convolution between input and reference analog signals. A special case of such a circuit is the CCD Analog Transversal Filter (TVF)[3.2-3] which has a fixed set of appropriately weighted reference coefficients that multiply incrementally delayed signal samples. The sum of the weighted time samples provides the convolution of the reference function and the signal.

TABLE 3.1-2.  COMMERCIAL D/A CONVERTERS

| Co. | Model | Bits | Time, μsec | Power | $* |
|---|---|---|---|---|---|
| Analogic | 1916 | 16 | 3 | 1.05 w | 485 |
| Burr Brown | DAC 70 | 16 | 50 | | 149 |
| Intech | A856 | 16 | 8 | | 1,300 |
| Analogic | 1915 | 15 | 2 | 1.05 w | 440 |
| Analogic | 1914 | 14 | 1.5 | 1.05 w | 395 |
| Dynamic Measurements | | 13 (12, 10, 8) | 350 | | |
| Datel | DAC-HY12BC | 12 | I 0.3 V 3.0 | 1.05 w | 29 |
| FMI | 175-12 | 12 | 3.5 | | 395 |
| Analog Devices | DACH08 AD 563 | 12 12 | 0.150 1.2 | 780 mw 525 mw | 122 42 |
| Micro Networks | MN371 | 12 | 35 | 90 mw | |
| Burr-Brown | DAC 80 | 12 | I 0.3 V 3.0 | 800 mw | 26.50 |
| Micro Networks | MN310-1 | 10 | 3 | 500 mw | 79 |
| Burr-Brown | AD7522 | 10 | 0.5 | | |
| Computer Labs | | 10 | 0.066 | | 1,010 |
| Micro Networks | MN316-1 | 8 | 1.0 | 0.4 | 59 |
| Analog Devices | AD7522 (UP compatible) | 8 | 0.15 | | |
| Hughes | | 6 | 0.006 | 700 | |

*Included as an indicator of relative complexity

3-6

Test chip CCD 2091 shown in Figure 3.2-1 consists of four such matched TVFs with several variations of input gain and sample and hold circuits, differential amplifiers, a charge comparator, and a charge subtractor. The chip measures 0.195 x 0.195 inch and is fabricated by using p channel overlapping aluminum/polysilicon electrode structure and 2:1 projection alignment technology. These devices replace conventional frequency domain analog filters and give significantly better performance.
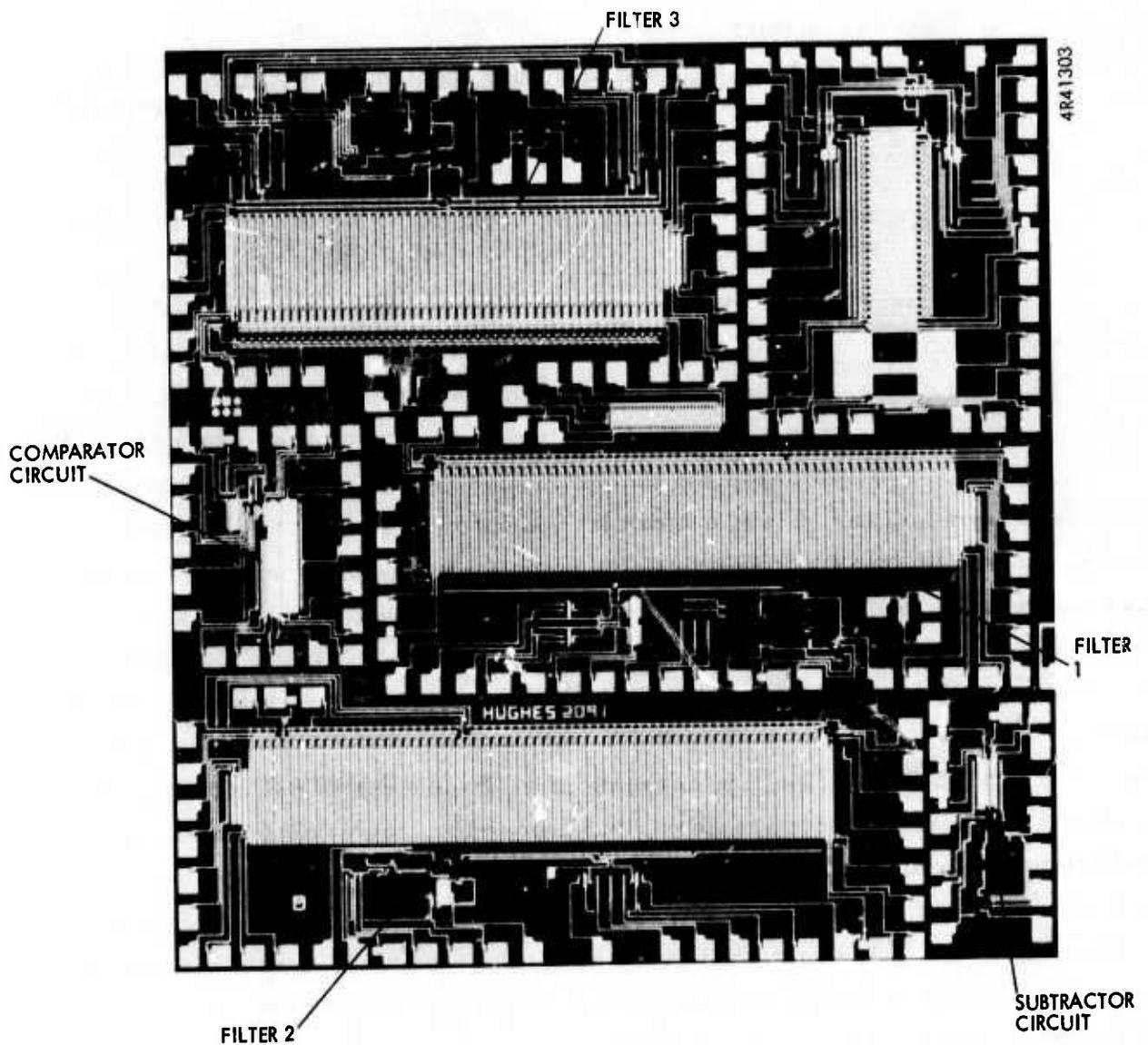


Figure 3.2-1. Hughes 2091 CCD matched filter test chip.

The matched filter concept can be summarized by the functional form of a matched filter transfer function, i.e.,

$$G(j2\pi f) = \frac{kS^*(j2\pi f) \exp (-j2\pi f\Delta)}{\left|N(j2\pi f)\right|^2}$$

where

$k$ = a constant,

$S^*$ = complex conjugate of signal frequency spectrum, which is frequency domain equivalent to time inverse of signal

$\Delta$ = phase factor, which corresponds to an appropriate time shift

$\left|N(j2\pi f)\right|^2$ = noise power density spectrum to which the filter is to be mismatched.

Filter 1 has 19 delay bits followed by 40 weighted bits. This filter has an output differential amplifier that dissipates about 100 $\mu$w and has a bandwidth of 30 kHz and a gain of 3. An on-chip sample and hold circuit eliminates clock feedthrough in the filter output waveforms (see Figure 3.2-2).

Figure 3.2-3 compares the theoretical and experimental frequency responses from a filter (#3) at a clock frequency of 31.2 kHz. The slight discrepancies observed at low signal frequencies are due to capacitive imbalance between the positive and negative sides of the filter, and systematic tap weight errors, which can occur during chip fabrication. Higher frequency differences are caused by a combination of transfer inefficiency and the bandwidth of the filter as measured by the delay from the filter input to the last tap. For a 31.2-kHz clock, the corresponding bandwidth is 637 Hz; therefore, as signal components increase above this frequency, the effects of transfer inefficiency become important. These filters have also operated with as low as a 300-Hz clock rate at room temperature with insignificant deterioration in performance, i.e., a slight shift in fat zero which reduced dynamic range by about 1 dB. This suggests acceptable dark current levels of approximately 10 na/cm$^2$. (Filter #2 was not tested).

The excellent correlation between theoretical and measured frequency response indicates the feasibility of reducing the size of the CCD register

CLOCK FREQUENCY, KHz

FILTER INPUT

ON-CHIP DIFFERENTIAL
AMPLIFIER OUTPUT
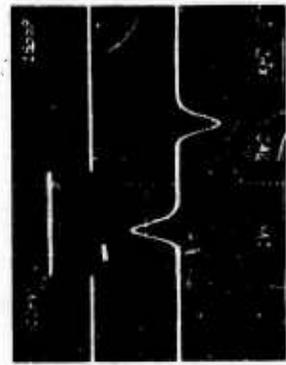
b. FILTER OPERATION WITH SAMPLE/HOLD

CLOCK FREQUENCY, KHz

FILTER INPUT

ON-CHIP DIFFERENTIAL
AMPLIFIER OUTPUT

a. FILTER OPERATION WITHOUT SAMPLE/HOLD

CLOCK FREQUENCY, KHz

EXPANDED VIEW OF ONE
SAMPLE (3.5 MSEC) AT LOW
CLOCK FREQUENCY

c. MEASUREMENT OF DROOP IN SIGNAL

Figure 3.2-2.   Operation of on chip sample and hold circuit of 2091 filter 1.
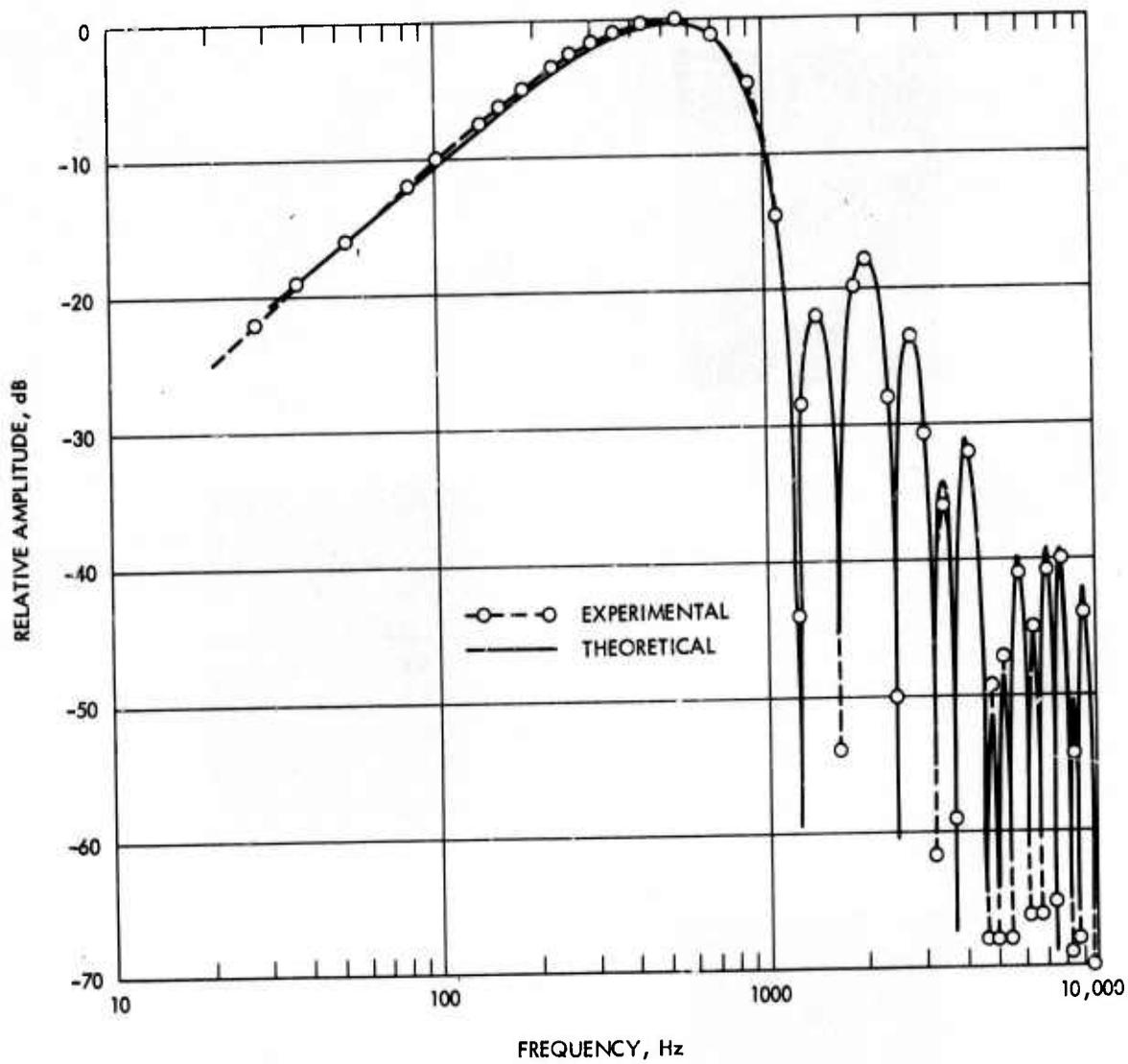
Figure 3.2-3.  Frequency response of filter 3
at a clock frequency of 31.2 KHz.

while maintaining satisfactory accuracy of the tap weights for greater density and higher frequency. Techniques have been developed recently that eliminate entirely the requirement for output differential amplifiers in transversal filters. This is important for high frequency TVF implementation due to the unacceptable common mode rejection and power dissipation of high bandwidth differential amplifiers.

### 3.2.2  Walsh-Hadamard Transform Domain Signal Processing Devices

Two types of signal processors are considered for the APSP application: 1) Real time pixel space processing, and 2) Real time transform domain processing. In order to implement the second type of signal processing (Reference 3.2-1), a CCD Walsh-Hadamard Transform (WHT) filter is required as shown in Figure 3.2-4. A set of 32 Hadamard sequencies (normalized frequencies) is generated with a set of 32 WHT filters. Two of these filters are shown in Figure 3.2-5. The filters are finite impulse response (FIR) transversal filters with binary ($\pm 1$) tap weights.

The output amplitude of each WHT filter with sequency = 0, 1, 2 ... 31 is the projection of the signal vector onto the 32 Walsh basis vectors. This amplitude is encoded using an A/D converter with a number of bits consistent with the dynamic range at the output of each filter. The amplitudes decrease monotonically for increasing sequency if the input signal is an optical photo-generated signal (Lucosz bound, Reference 2).
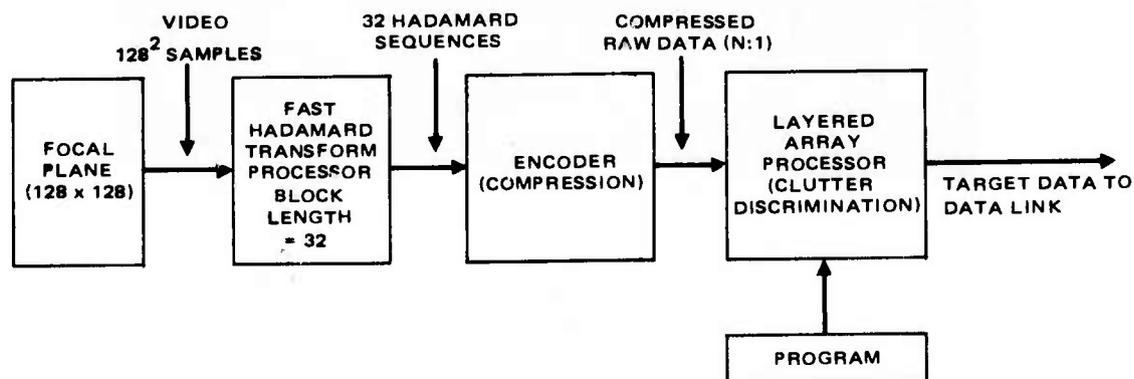


Figure 3.2-4.  Adaptive Hadamard Transform processor.

The amplitude statistics in the sequency domain are generally well balanced Gaussian statistics with zero mean value, except for sequency 0. Conventional classical coding rules can be applied to the encoding of each WHT filter output with a resultant overall reduction in the number of bits, compared to that required for the original signal amplitude. A 3:1 reduction has been obtained for standard TV picture coding, without noticeable degradation. The concept of transform domain processing can be extended with orthogonal functions such as the cosine functions and others. The advantages of the Walsh-Hadamard Transform are its binary characteristics and ease of implementation. The suitability of the WHT for APSP is discussed in the Processor Architecture report CDRL A006.
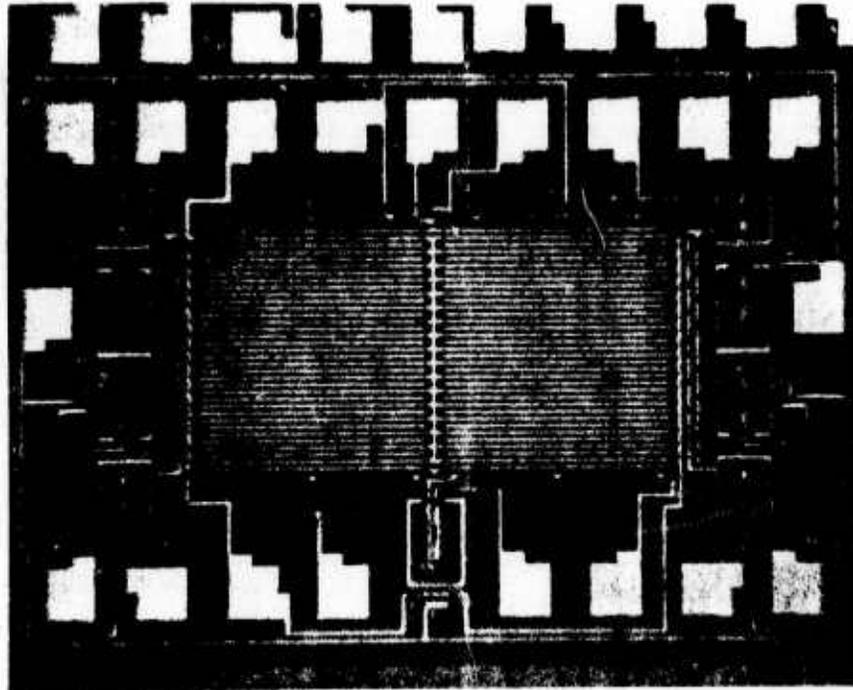


Figure 3.2-5.   Dual 16 Element
Hadamard Filter Chip
No. 2088.

# REFERENCES

3.2-1  Yamaguchi & Sato, IEEE Trans. on Electron Devices, May 1975, page 295.

3.2-2  Baldwin:  1975 IEEE International Solid-State Circuits Conference.

3.2-3  Sekula, et al., "CCD Non-Recursive Matched Filters Using Charge Coupled Devices," IFEE International Electron Devices Meeting, Technical Digest, Washington, December 1974, pp. 244-247.

3.2-4  Harp, et al., "Analog Correlators Using Charge Coupled Devices," 1975 International Conference on the Application of Charge Coupled Devices, October 1975, pp. 229-235.

3.2-5  H. F. Harmuth, "Transmission of Information by Orthogonal Functions," Springer Verlag N.Y., Heidelberg, Berlin Second Edition, 1972.

3.2-6  Lucosz, W., "Ubertragung Nicht-Negativer Signale durch Linear Filter," Optica Octa 9, p. 335-364, 1962.

## 4.0 DEVICE TESTING

As part of this survey, several devices in various technologies designed and fabricated by Hughes were tested in conjunction with the evaluation programs associated with each device. In general these tests involved one or a few of the devices on the chip concerned, and typically only those parameters of interest to the APSP application were evaluated. The data in many cases does not represent optimized process iterations. It does, however, provide an insight into potential performance capabilities and demonstrates the involvement of Hughes in virtually all of the state of the art technologies, generally in considerable depth. The data is presented in a variety of formats, including tabulations, computer print-outs, graphs and oscilloscope trace photographs.

### 4.1 I$^2$L DEVICES

This section provides test data collected on a first iteration I$^2$L process under evaluation at the Hughes, Newport Beach facility.

The chip is identified as the Hughes Model 2100, and contains 32 separate devices. Figure 4.1-1 is a photograph of the 2100 chip with a brief description of the devices. Those evaluated in this report are (a) the Ring Oscillators (R.O.) No. 6 and No. 9, (b) the eight stage shift register No. 17, (c) the ten stage frequency divider No. 15 and (d) the 4 bit adder No. 18. The Ring Oscillator No. 9 is actually two 15 stage oscillators called No. 9 and No. 9B. No. 9B is a smaller geometry version of No. 9 and No. 6.

Two different chips were tested as ring oscillators, one is designated A-7 and the other A-3. A-7 is a standard I$^2$L process called down diffused (or implanted) to distinguish it from the improved process called up diffused

densities. Of the two most attractive candidates, $I^2L$ and CCD, projected for the early 1980's, CCD technology offers a number of advantages over $I^2L$; the most important being small element size and low power dissipation. CCD memories are dynamic devices and consume little power in the standby mode of operation (even with refresh), while $I^2L$ memories consume considerably more power. CCD memory organized in a serial-parallel-serial (SPS) arrangement also offers the greatest bits per chip density and concurrently minimizes peripheral circuitry and access times. An important attribute of the SPS organization is that most of the charge transfers are done at a low frequency which vastly reduces the _effective_ power delay product.

## 4.1 CCD MEMORIES

Present state-of-the-art photolithographic techniques, as used on the Hughes 2069 chip of Figure 4.1-1, have the resolution capability of producing 0.7 mil/bit (18 μm) shift registers. Advanced photolithographic techniques utilizing 4:1 reductions can produce optimal shift registers of 0.4 mil/bit (10 μm). Future processing, however, will utilize high resolution electron beam (E-beam) technology. An internally funded program has



$\ell = 0.7$ MIL/BIT
CHIP SIZE: 126 x 129 MILS
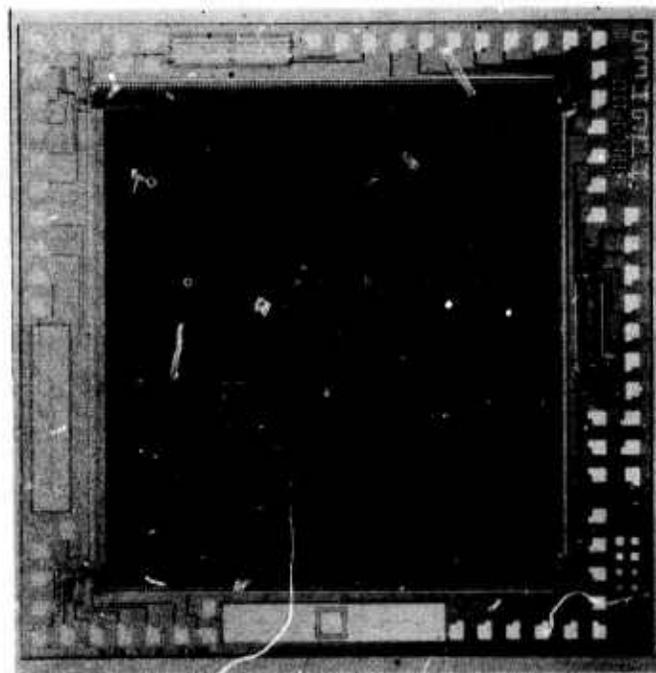
Figure 4.1-1. Hughes 32K bit CCD memory (chip 2069)

which is used in the A-3 chip. Details of this process are described in the
Hughes Patent Disclosure No. 75207. A set of six graphs is shown in Fig-
ures 4.1-2 to 4.1-7. The first three compare the processes at room tem-
perature and the second three show the effect of temperature on the A-3 chip.

Figure 4.1-2 shows time delay per stage as a function of supply
current per stage. Oscillators Number 6 and Number 9 have the same
geometry but Number 9 has improved isolation with less capacitance, result-
ing in lower delay than Ring Oscillator Number 6. The lowest delay is Ring
Oscillator Number 9b as a result of its smaller geometry.

Figure 4.1-3 shows power-delay product versus current, and again
the same trend is evident, except that now oscillator 9 of chip A-3 shows a
substantial improvement over the smaller geometry 9b of chip A-7, depicting
the process improvement. Figure 4.1-4 shows the output signal level for the
same ring oscillators. A 1K load resistor was used to obtain accurate wave
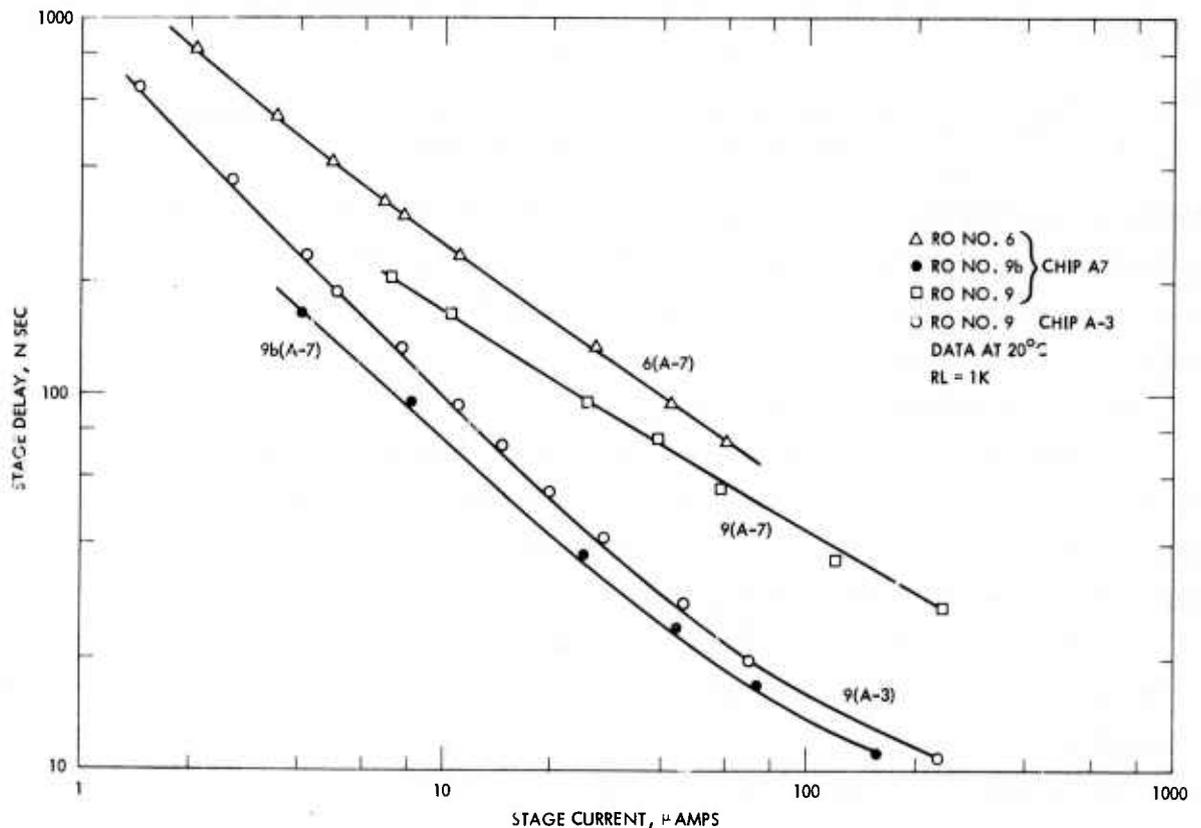


Figure 4.1-2. Ring oscillator time delay per stage versus stage current.

4-3
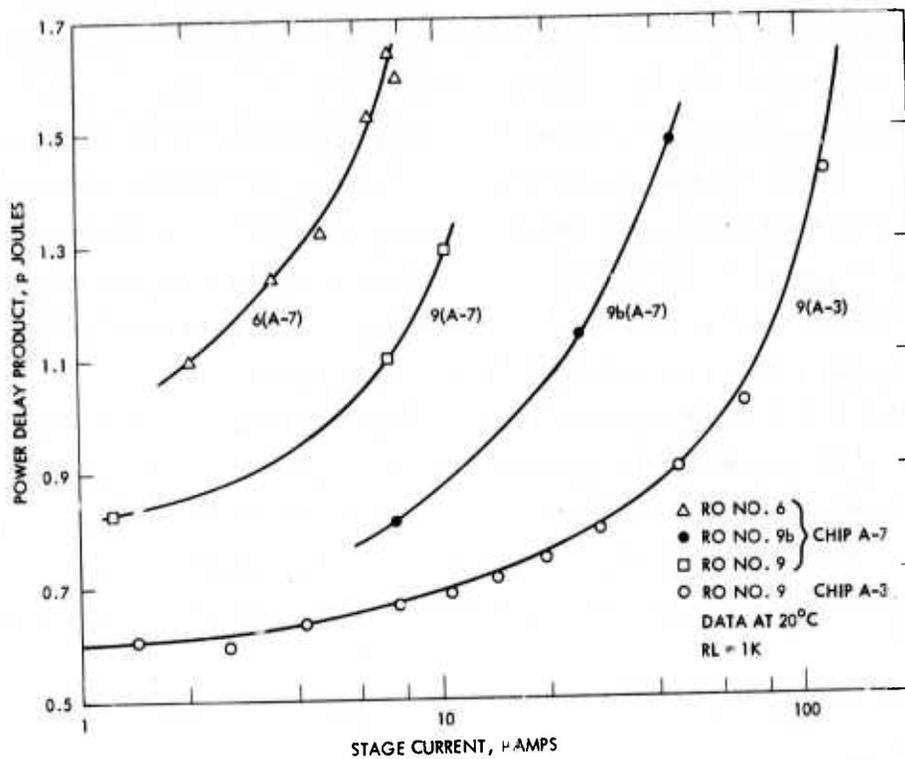
Figure 4.1-3. Power delay product versus stage current for various ring oscillators.

forms at the higher speeds. An increase in output level can be obtained with larger load resistors. In chip A-7 the value of the output supply voltage was somewhat critical, for reasons not completely understood at this time, with 1.75 Volts giving the largest output. A-3 was not as sensitive to variations in this voltage (3 V was used).

The characteristics of A-3 with temperature, shown in Figures 4.1-5, 6, 7 establish clearly that the performance improves with temperature, for the parameters checked. This is due largely to the improvement in beta and the lowered emitter-base voltage drop.

All of the ring oscillators used had fifteen stages connected as shown in Figure 4.1-8. The lateral PNP's have a β of about 10 and must obviously be well matched so that the supply current will be evenly distributed throughout the stages. This does not pose a problem with normal processing. A more practical problem is that of furnishing an efficient power supply. As shown by Figure 4.1-5 the supply voltage is strongly dependent upon temperature.

4-4
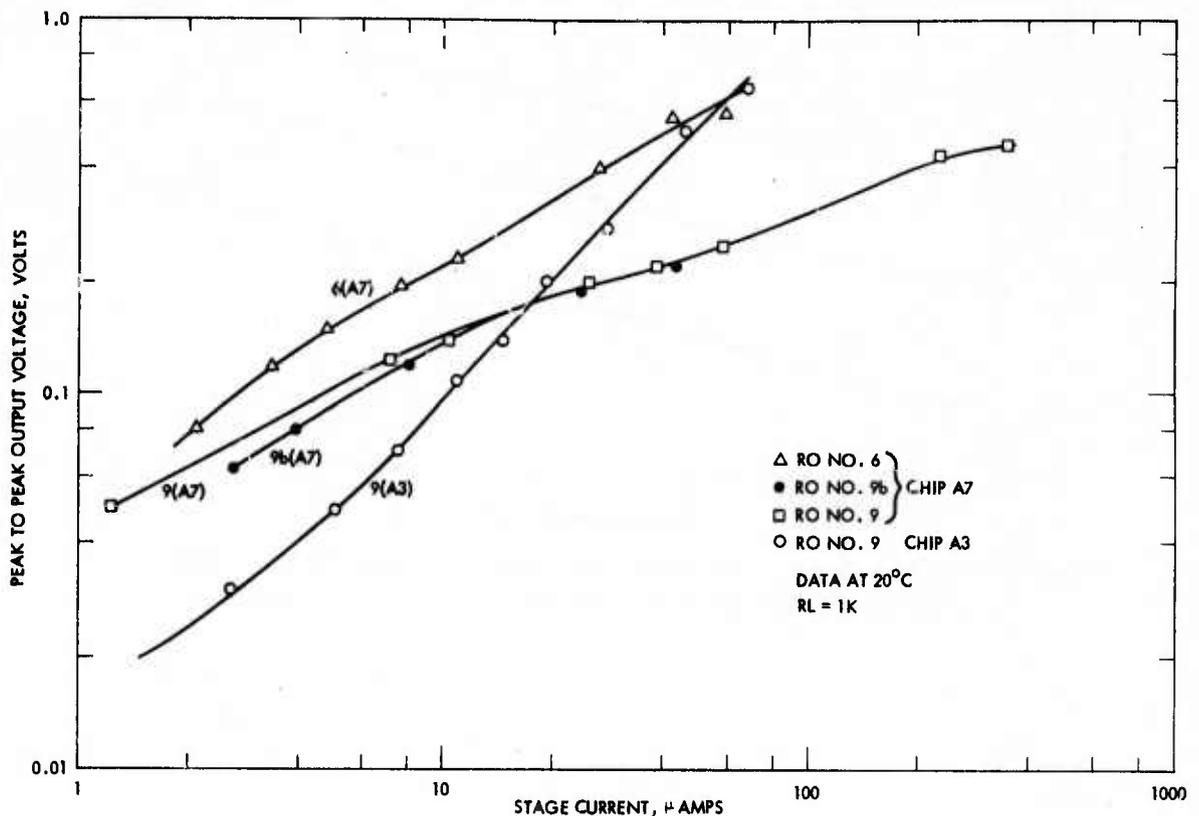
Figure 4.1-4.  Output voltage as a function of stage current for
various ring oscillators.

This comes as no surprise since it is simply the emitter to base drop of the
lateral PNP.  For the purposes of this test, as in practically all other pub-
lished $I^2L$ data, a large resistor (10K to 100K) was used in series with the
device to control the input current, and the power dissipation for the power
delay products of the device were calculated by using the device voltage drop
rather than the much larger supply voltage.  Clearly, when $I^2L$ is compared
with other forms of logic, the unique loading of the power supply and its
losses must be included.  This could well make a significant difference in
the results of the comparison.

Another factor of concern is cross talk or feedback, since there is
no decoupling in the basic $I^2L$ circuit.  This is another factor which leads to
quoting optimistic power levels for $I^2L$, since there is usually a substantial
power loss associated with decoupling in discrete circuits.
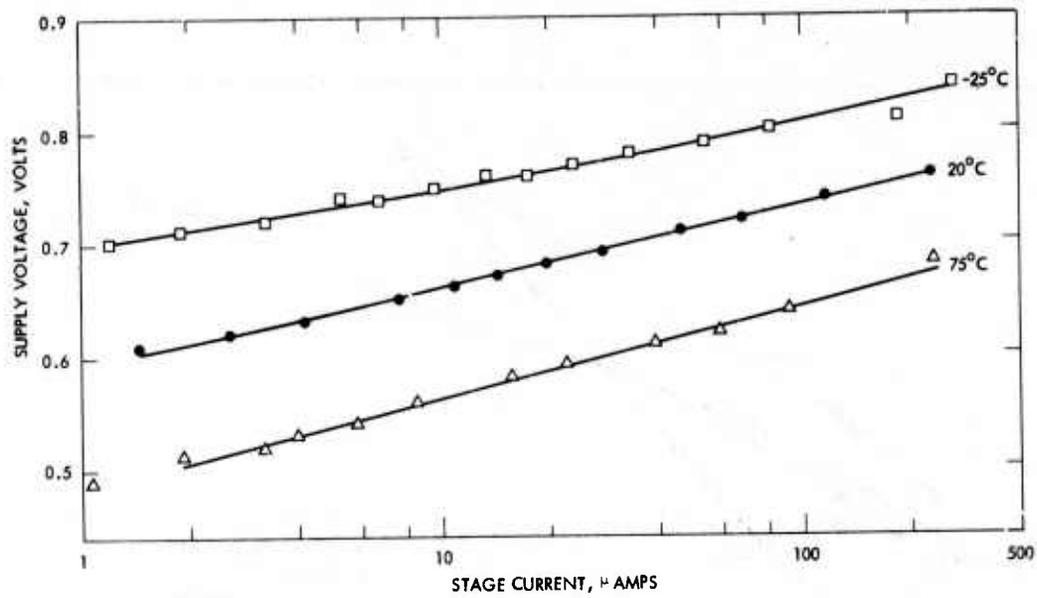
Figure 4.1-5. Ring oscillator A3 supply voltage at the device as a function of device current for three temperatures.
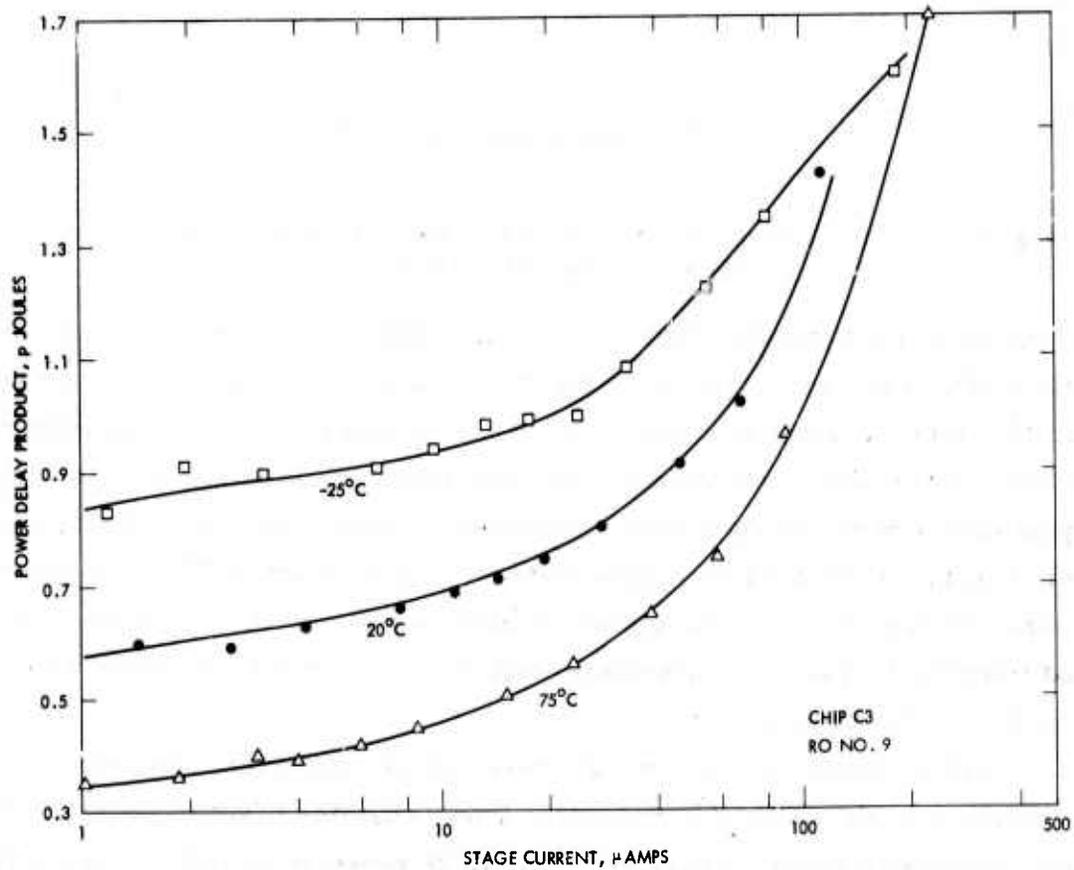


Figure 4.1-6. Ring oscillator A3 power delay product per stage as a function of stage current for three temperatures.
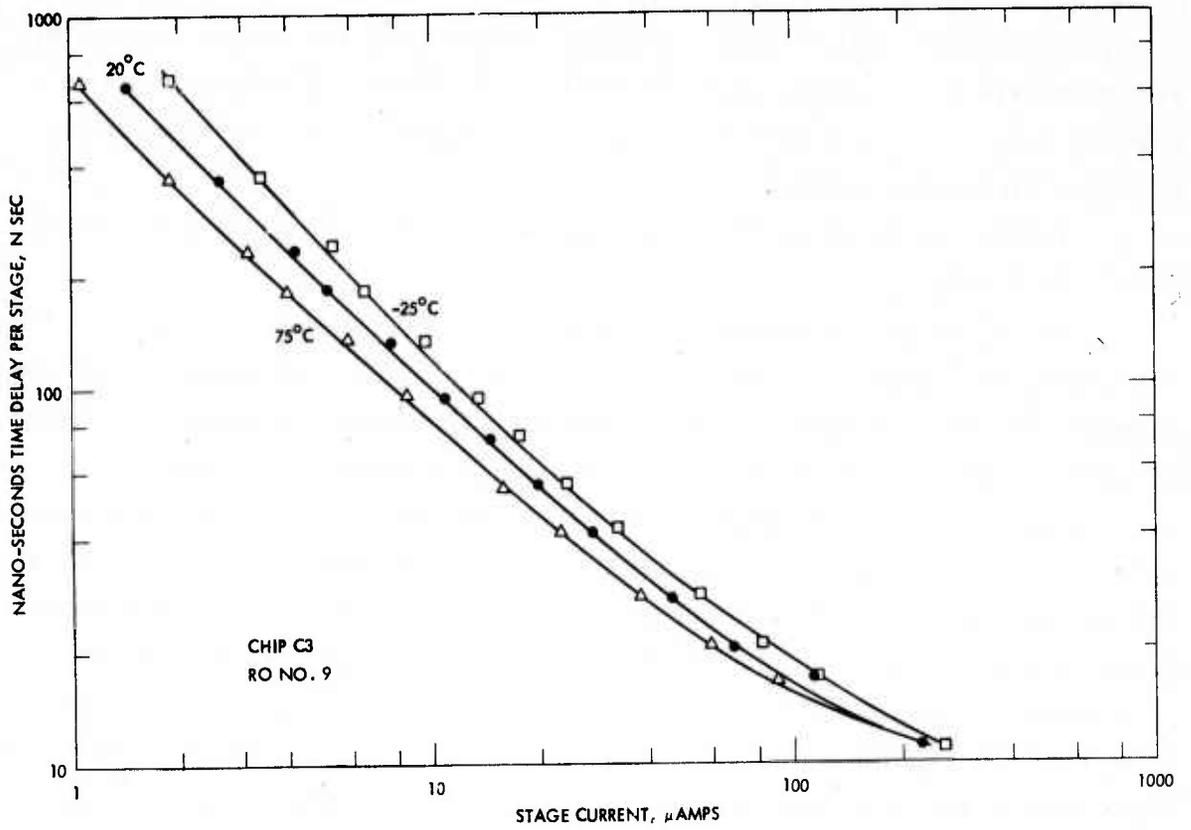
Figure 4.1-7. Ring oscillator A3 stage delay as a function of stage current for three temperatures.
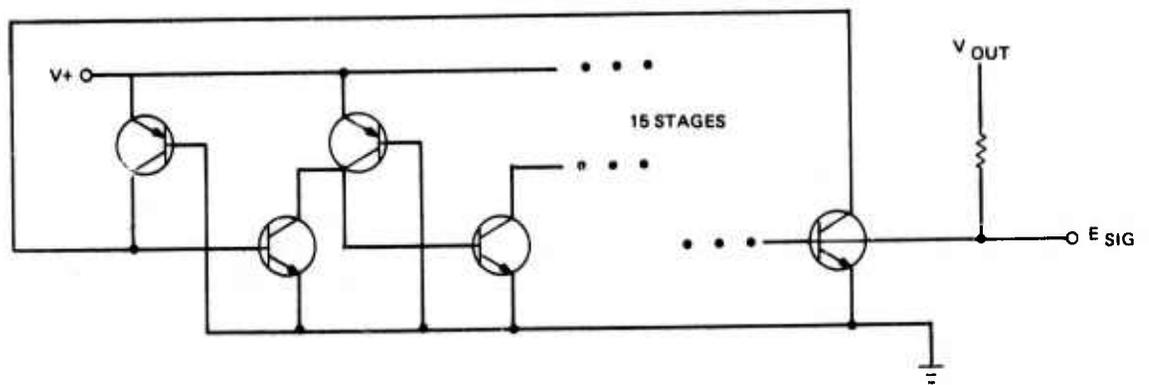


Figure 4.1-8. Ring oscillator schematic.

However, favoring $I^2L$ technology in this respect is the fact that it is limited to digital circuits, and therefore should not be compared to the decoupling requirements of analog devices. Coupling problems will become more critical in large arrays where the ohmic resistance of the interconnects becomes an important factor.

Device A-10 included a shift register which was operated as shown in Figure 4.1-9.

Since the shift register triggers on a high-to-low transition, the circuit shown in Figure 4.1-9 was used to insure that no false triggers affected testing. At the maximum clock frequency of 100 KHz, the output was lagging by almost 1/2 clock cycle (5 $\mu$ sec), which corresponds to a delay of 0.625 $\mu$ sec/stage. The output supply was adjusted to +3.6 V for a maximum output of 0.3 V. The power input was set to +11 V through a 10K resistor. Eleven volts was selected as being in the center of the 9-12.5 V operating range of the device. The actual chip supply was a higher than expected (1.8 volts). Because of the limited operating voltage range, it was not possible to get data on speed versus supply current. The reason for this voltage limitation is not fully understood at this time. In addition, the maximum frequency of operation should have been on the order of 5 Mhz, rather than the observed 100 KHz. The causes of all of these limiting conditions are presently under investigation. In all cases, the above is preliminary data, realized from 1 each of 5 devices on the chip.
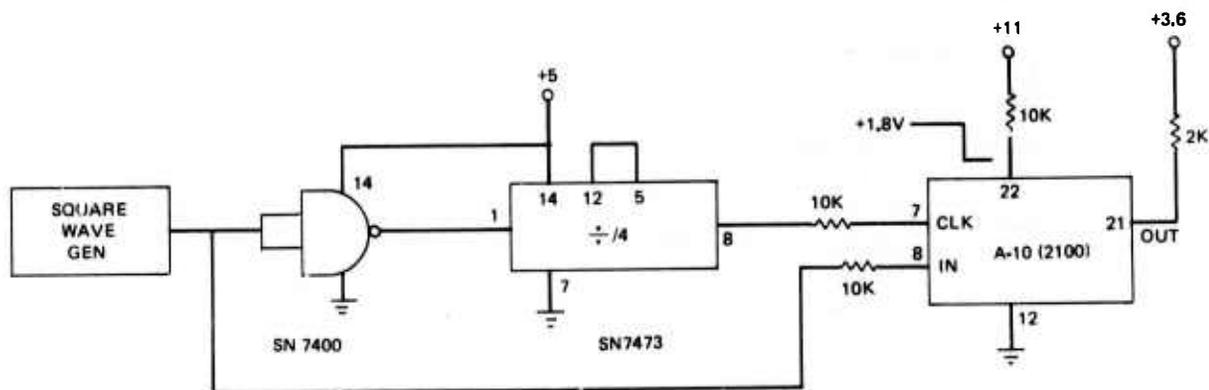


Figure 4.1-9. Test circuit for shift register.

Tests on the 4 bit look ahead adder resulted in a maximum of 1 Mhz operation with a supply current of 2 m.a. at 2.2 volts. This device has 63 transistors. Tests on properly operating frequency dividers have reached 5 Mhz.

## 4.2 PERISTALTIC CCD

The Hughes 2096 test chip includes a peristaltic 64 bit shift register. This section provides test data for an N channel device operating at a 103 MHz clock rate. Figure 4.2-1 is a schematic diagram of the device. Those tested were fabricated on a 3.5 μm thick N expitaxial layer. Gate length is 1.2 mils/bit and gate width is 4.0 mils. The register is clocked at 103 MHz with 2$\phi$, 10 V P-P sinewave clocks illustrated in Figure 4.2-2. The CCD input consists of a modulated 3rd gate with the first two gate electrodes biased to form a current source. The output sense diffusion was connected directly to a 250Ω load resistor. The reset gates were not used.

The device was connected as in Figure 4.2-1 with bias conditions listed in Table 4.2-1. The input gate ($\phi_{in}3$) was modulated with a pulse having 4 ns rise and fall times. The input and delayed output (10 ns x 64 bits = 640 ns) are shown in Figure 4.2-4.

Frequency response and transfer efficiency can be estimated from the rise time of the CCD output wave form. (The source of the output ripple in the early part of the trace has not yet been identified. The observed output
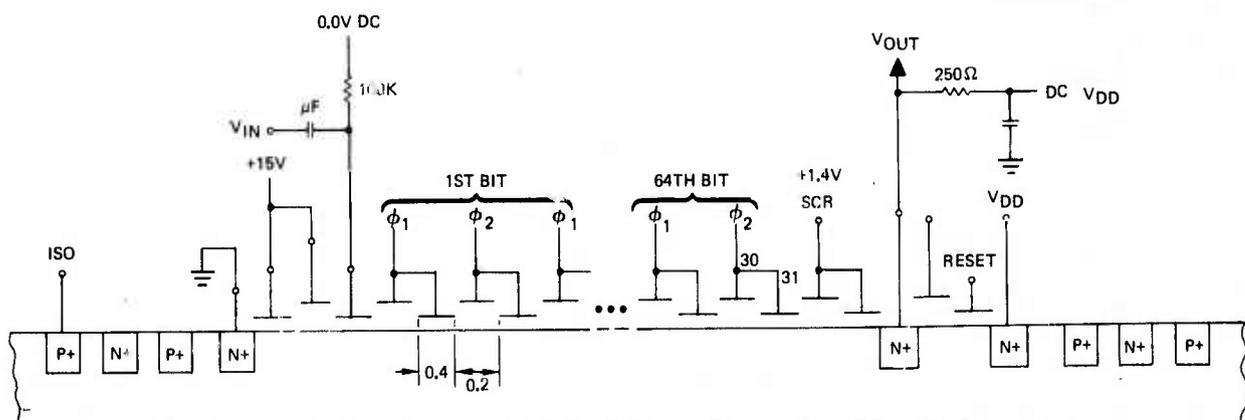


Figure 4.2-1. PCCD 64 bit 4$\phi$/N-channel CCD shift register.

$\phi_1$    5 V/CM

$\phi_2$    5 V/CM

HORIZ: 5.0 NSEC/CM

$f_c$ 103 MHz

Figure 4.2-2.    103 Mhz $2\phi$ PCCD clock.

TABLE 4.2-1.    PERISTALTIC CCD (2096) BIAS CONDITIONS

| Electrode | Value |
|---|---|
| $\phi 1$ | 11 volts P-P, 103 Mhz, sinewave |
| $\phi 2$ | 9 volts P-P, sinewave |
| $\phi in^3$ | 0.0 dc volts; 4 nsec rise input |
| $V_B$ offset | +6 volts |
| $V_S$ offset | 0.0 volts |
| $V_{sub}$ | -11.7 volts |
| $V_{DD}$ | +10 volts |
| $V_{sc_r}$ | +1.4 volts |
| Input diffusion | 0.0 volts |
| $\phi in_{1, 2}$ | +15 V |
| T | $300^oK$ |

4-10

rise time of 13.6 nsec is determined by a) input pulse, b) register, and c) output circuits. The input rise time (4 ns) and output rise time (8.5 ns) are known, therefore the register bandwidth can be estimated from a magnitude response function of three cascaded single pole functions. This calculation leads to a 45 MHz bandwidth, which is consistent with the theoretical sin x/ x response at a 103 MHz clock rate.

The maximum bucket capacity can be estimated from the peak-to-peak output voltage swing across the load resistor. The maximum voltage swing is 37.5 mv., the test circuit output capacitance was about 11 pf; from $\Delta Q = C \Delta V$, $\Delta Q = 0.41$ pc.

Thus

$$\text{Bucket capacity} \approx \frac{0.41 \text{ pc}}{1.6 \times 10^{-19}} \approx 2.6 \times 10^6 \text{ electrons}$$

These results show device operability and represent an approximation of the registers' ultimate performance capability since the test equipment available prevented more accurate determination of performance. Test results indicate transfer efficiency of >0.999 and bandwidth of 45 Mhz at a clock frequency of 103 Mhz. Further evaluation at higher clock frequencies using appropriate test equipment will more accurately determine the limitations of the device. These results compare favorably with data presented by Rockwell International at the 1975 CCD Applications Conference at San Diego (>0.999 at 105 Mhz clock). Phillips Research Labs achieved >0.9999 at frequencies of 100 Mhz in 1973. The 2096 has comparable performance when the I/O circuit effects are normalized out.

### 4.3 CMOS/SOS

Many companies in the industry are investigating the CMOS/SOS process. The Newport Beach facility of Hughes is well along in the development of advanced techniques for implementing CMOS/SOS (Figure 4.2-4). In particular a considerable amount of effort has been invested in the implementation of minimum geometry CMOS/SOS devices with polysilicon gates.

Figure 4.2-3. 2 φ resonant clock driver.



INPUT 1 V/CM

OUTPUT 20 MV/CM

HORIZ:
100 NSEC/CM
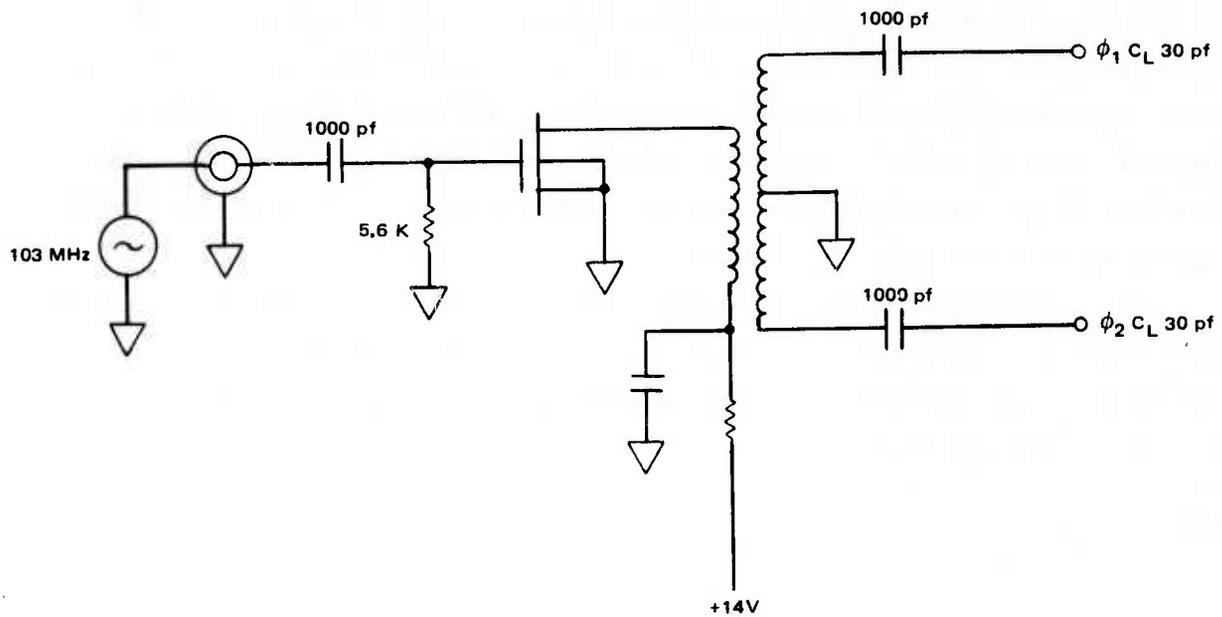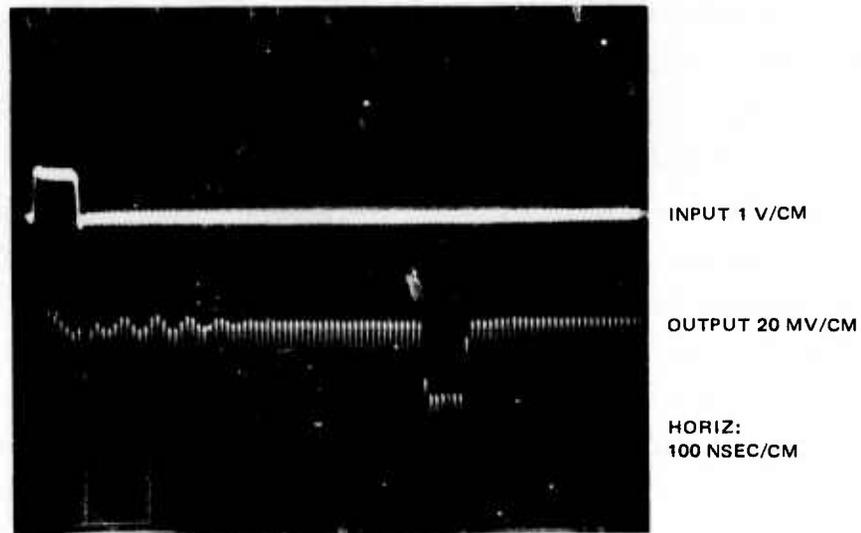
Figure 4.2-4. Peristaltic CCD pulse response
for $f_c$ = 103 MHz.

When standard metal gates are used the overlap of the source and drain diffusions underneath the metal gate area add an extra capacitance factor. In addition the difference in the Fermi levels of the metal and the semi-conductor cause a work function term $(\emptyset_{MS})$ in the threshold voltage equation. If a polysilicon gate is used (as illu trated in Figure 4.3-1 for an n-channel device) the gate material is the same as the semi-conductor, causing the work function to drop to zero and decreasing the threshold voltage. Decreased threshold voltage in turn decreases both the supply voltage and the switching voltage required, improving the speed-power product. In addition there is no overlap capacitance involved in a polysilicon self aligned gate. Polysilicon gates are thus created through a self-aligned process and the devices have better fanout capabilities due to the reduced capacitance. Gate capacitance is reduced about 40% compared to that with bulk silicon. Hughes Newport Beach facility utilizes an electron beam (EBIM) process for accurately aligning the structure and providing the ion implantation used in the process.

The data summarizes the present status of CMOS on sapphire tech-nology (data compiled from Hughes, Newport Beach).

For a minimum size inverter:

| | |
|---|---|
| channel width | = 0. 6 mil |
| channel length | = 0. 3 mil |
| capacitance/mils$^2$ | = 0. 2 pf/mil$^2$ |
| supply voltage | = 10V |
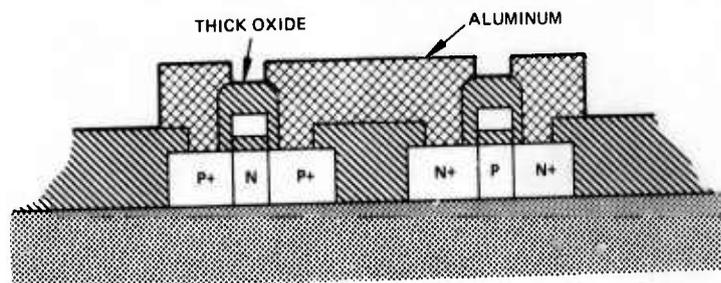


Figure 4. 3-1.   N channel MOST self aligned gate structure.

$$\text{oxide thickness} = 1000 \overset{\circ}{\text{A}}$$

$$E_{pt} = P/f = \text{speed x power} = C V_s^2$$

$$\cong 3.6 \text{ pJ (advanced circuit)}$$

Hughes recently constructed a 256 bit CMOS shift register on a 130 by 138 mil chip containing 4200 transistors (Figure 4.3-2). The basic unit cell building block contained 16 transistors on a 25.6 $\text{mil}^2$ area, to give some idea of the packing density involved. The speed power product of 3.6 pJ is a good figure for contemporary LSI functions. Channel lengths of 0.1 mil were obtained. With some devices, power supply voltages as low as 1.5 volts were demonstrated.

COMPLETE CHIP
4200 TRANSISTORS

UNIT CELL
16 TRANSISTORS



CHIP SIZE: 130 x 138 MILS

1 MIL
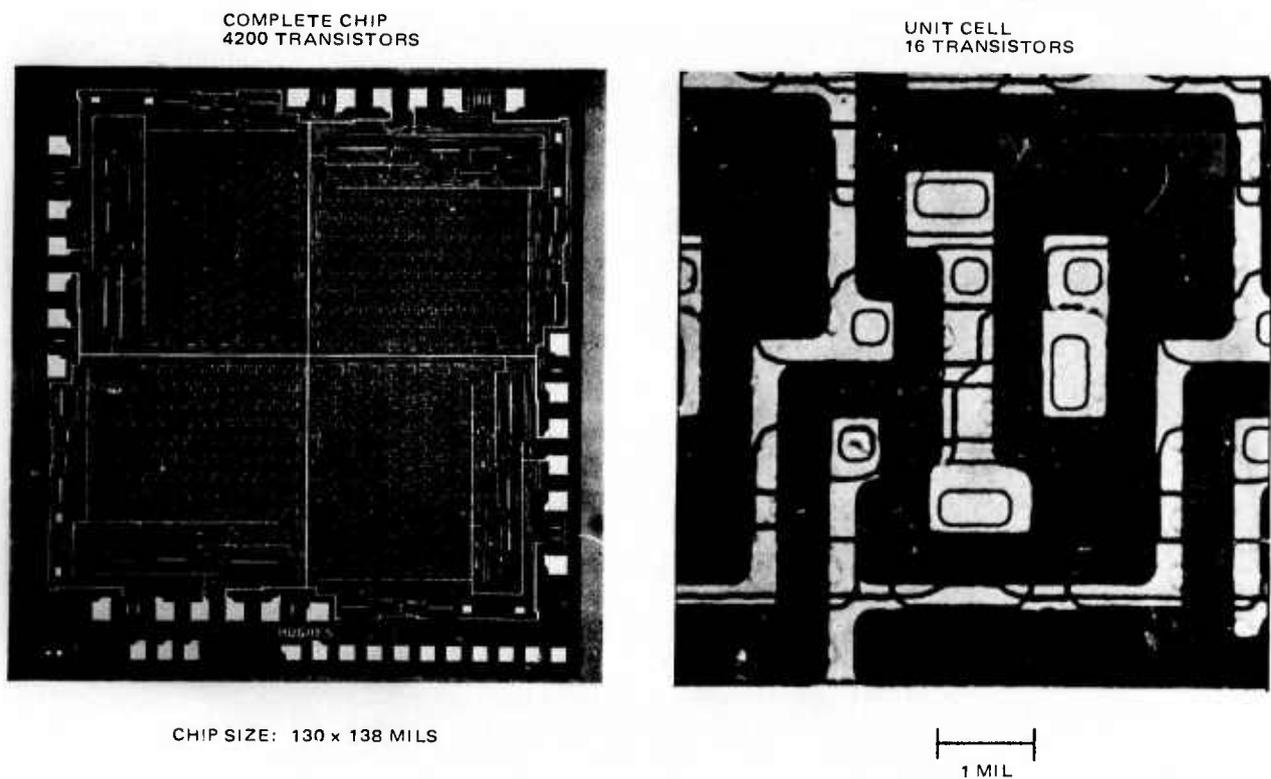
Figure 4.3-2. CMOS/SOS 256 bit static shift register.

## 4.4  WALSH-HADAMARD FILTER

The WHT transversal filters incorporated in the 2088 chip (described in Section 3.2.2) were operated at a 10 MHz clock rate. The filters have a 1.2 mil bit length and are implemented with buried P channel technology. Figure 4.4-1 illustrates the impulse response for a sequency 8 device. The impulse response represents a Walsh function (±1 tap weights) for which the sequency is the number of zero crossings.
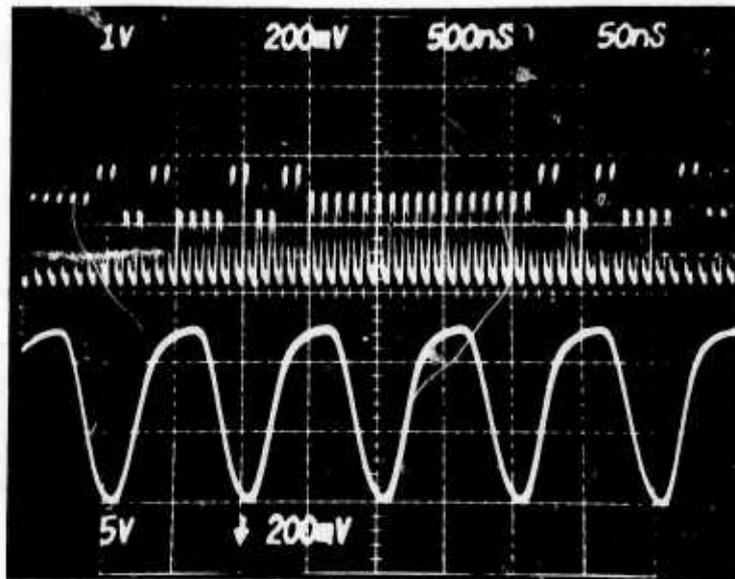


Figure 4.4-1.  Hadamard Filter Impulse Response
Hughes Chip No. 2088, Sequence 8; 10 MHz Clock.

## 4.5 CCD COMPATIBLE BIPOLAR DEVICE

The Hughes 2096 chip contains several interface circuits of the Bipolar-MOS (BIPMOS) configuration to provide information on the capability of bipolar devices with MOS devices and buried channel CCDs. In addition, the BIPMOS circuit can be utilized as an output buffer-amplifier. This section provides detailed characterization of two of the bipolar transistors on the chip. Section 4.6 provides the results of SPICE computer modeling and testing of the bipolar device in conjunction with a MOS common source amplifier.

Devices from lot 9, wafer 8 and lot 14, wafer 10, fabricated using the mask set #2096 were tested in order to characterize the vertical NPN transistor proposed for use in the BIPMOS driver circuit. The #2096-9 version is a non-isolated, N substrate device and the #2096-14 version is an isolated, P substrate device. The emitter and collector saturation currents and emission coefficients were calculated from a least squares curve fit of the DC base to emitter voltage versus the logarithm of the emitter current in both the forward and reverse mode of operation. The low frequency emitter and collector resistances were measured by saturating the transistor with a current source base drive and then calculating the resistances. The results of these tests are summarized in Table 4.5-1.

The procedures used in making the small signal and junction capacitance measurements, summarized in Table 4.5-2, were an automated version of essentially those described in Reference 4.5-1, except as follows:

1.  $C_{JE}$ and $C_{JC}$ curves were obtained by using a new measurement technique (relative to the one discussed in Ref. 4.5-1. The new technique, suggested by Hewlett Packard personnel, requires the use of an HP 4271 LCR meter. As shown in Figure 4.5-1(a), due to the manner in which operational amplifiers are incorporated in the instrument sensing circuitry, the HP 4271 behaves very nearly as an ideal voltage source driving the test capacitance with the current through the test capacitance sensed at virtually zero impedance. This is suggested in the equivalent circuit shown in Figure 4.5-1(b). These characteristics of the HP 4271 nullify the effects of shunt parasitic capacitances $C_{S1}$ and $C_{S2}$ also shown in Figure 1b. (The current through $C_{S1}$ is not measured and no current flows through $C_{S2}$).

TABLE 4.5-1. 2096 VERTICAL NPN CHARACTERISTICS

| MASK SET | $I_{SC}$ AMPS | $I_{SE}$ AMPS | $N_C$ | $N_E$ | $R_E$ $\Omega$ | $R_C$ $\Omega$ |
|---|---|---|---|---|---|---|
| 2096-9 Wafer 8 (N substrate) | | | | | | |
| 1st | 2.7 E-14 | 1.1 E-15 | 1.13 | 1.06 | 11.0 | 12.5 |
| 2nd | 1.8 E-14 | 8.1 E-16 | 1.11 | 1.03 | 12.0 | 11.5 |
| 2096-14 Wafer 10 (P substrate) | | | | | | |
| 1st | 1.5 E-14 | 1.1 E-16 | 1.11 | 1.01 | 15.5 | 11.7 |
| 2nd | 5.1 E-15 | 9.4 E-17 | 1.17 | 1.01 | 15.5 | 15.8 |

MODEL EQUATIONS:

$$I_E = I_{SE} \ (e^{\ V_{BE}/N_E V_T} -1)$$

$$I_C = I_{SC} \ (e^{\ V_{BC}/N_C V_T} -1)$$

$$V_T = kT/q$$

TABLE 4.5-2.  SUMMARY OF 2096 VERTICAL NPN DATA FIGURES

| Measurement | Wafer | Device No. | Figure |
|---|---|---|---|
| $V_{BE}$ vs $I_C$ | 8 | 1 | |
| | 8 | 2 | 4.5-4 |
| | 10 | 1 | |
| | 10 | 2 | |
| Small Signal AC beta (at 1 kHz) vs $I_E$ | 8 | 1 | 4.5-5 |
| | 8 | 2 | 4.5-6 |
| | 10 | 1 | 4.5-7 |
| | 10 | 2 | 4.5-8 |
| $C_{BE}$ vs $V_{JC}$ at 1 MHz | 8 | 1 | |
| | 8 | 2 | 4.5-9 |
| | 10 | 1 | |
| | 10 | 2 | |
| $C_{BC}$ vs V at 1 MHz | 8 | 1* | |
| | 8 | 2* | 4.5-10 |
| | 10 | 1 | |
| | 10 | 2 | |
| $C_{CS}$ vs V at 1 MHz | 8 | 1* | - |
| | 8 | 2* | - |
| | 10 | 1 | 4.5-11 |
| | 10 | 2 | |

*The collectors of the wafer 8 devices were connected to the collector of another lateral PNP transistor and therefore the collector base capacitances are higher than normal.  The collector substrate capacitances were not measured for wafer 8.  For this reason no $f_T$ test was performed on the wafer 8 devices.  The wafer 10 devices had this problem corrected and an $f_T$ test was performed on device number 2.

(a) HP4271 IN CAPACITANCE
    MEASUREMENT

(b) EQUIVALENT CIRCUIT
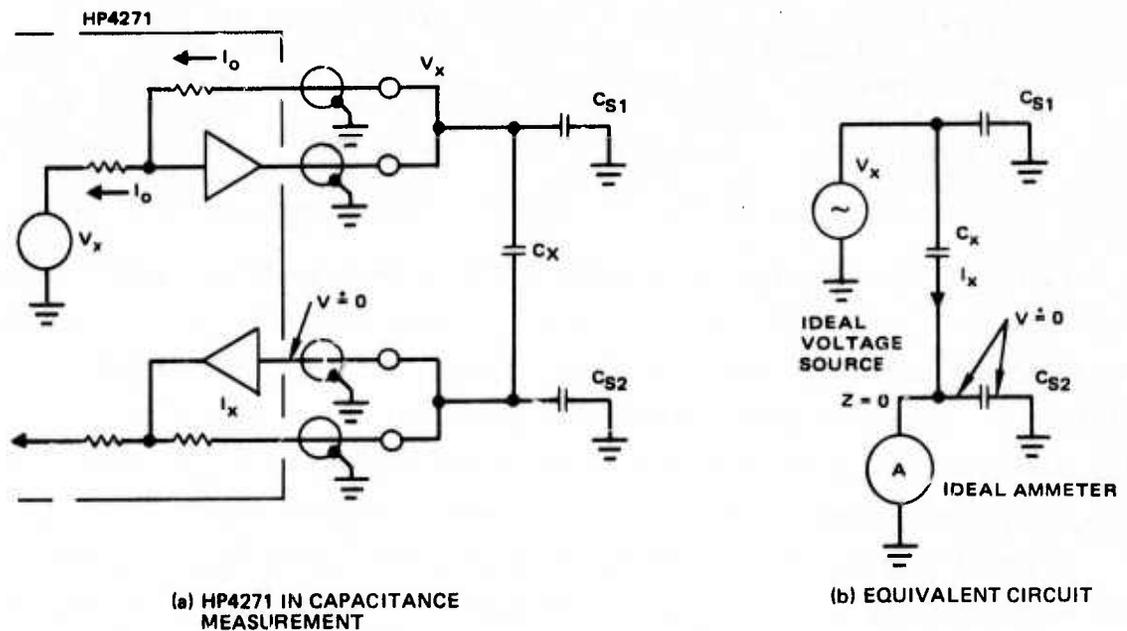
NOTE — CIRCUITRY INTERNAL TO HP4271 FOR APPLYING DC BIAS TO TEST
(E.G. JUNCTION) CAPACITANCE, $C_X$ NOT SHOWN.

Figure 4.5-1. Test configuration for $C_{JE}$ and $C_{JC}$ measurements.

Consequently, $C_{JE}$ and $C_{JC}$ capacitance versus junction voltage curves may be made without corresponding "Pads Only" capacitance measurements. Instead both measurements are made by connecting the HP 4271 directly across the junction of interest. In either case, all other junction and fixed parasitic capacitances will be found to act as shunt capacitances ($C_{S1}$ and $C_{S2}$) to an "exterior" ground. Unfortunately, this technique can NOT be used in measuring $C_{CS}$ versus $V_{CS}$. This is the case because (at least) a constant pad plus header parasitic capacitance appears in parallel with the collector-substrate junction even if the substrate is insulated from the header (e.g., by an insulating epoxy bond). Consequently, a pads only device was used in collector-substrate capacitance measurements.

2.  $f_T$ measurements were made using a pulse response test. $f_T$ measurements are usually made by determining the small signal short circuit current gain of a fixed frequency, $f_o$. This frequency should be at least one octave above $f_\beta$ and at least two octaves below $f_T$. In this case, the $C_T$ approximation to the hybrid-pi model is valid, assuming small (nearly negligible) external and extrinsinc collector resistance in the measurement setup. Second order effects, neglected in the hybrid-pi model,

4-19

make results invalid for $f_o$ near $f_T$. For most ECL transistors tested $f_T \geq 1$ GHz and $f_\beta \leq 10$ MHz. Therefore, our measurements are made at $f_o = 100$ MHz. Special instrumentation for these measurements incorporate tuned circuitry (described in Ref. 1) which will only operate at 100 MHz. Consequently, the instrumentation can not be used for accurate $f_T$ measurements on devices with $f_T \leq 400$ MHz. This is also true for $R_B$ base resistance tests since they require similar instrumentation.

From preliminary measurements made on devices from wafer 10, it was determined that $f_T$ was below 400 MHz. It was therefore necessary to use an alternate technique which involves applying a step voltage signal to the transistor in the common emitter configuration. Then with $R_L + R_C << R_\pi$, the collector waveform may be analyzed to determine $f_T$. The measurement setup used is shown schematically in Figure 4.5-2.

A small (e.g., 200 mV) voltage step is applied with $V_{BE}$ initially biased sufficiently to just make the device active (e.g., 0.7 V). The output voltage, shown in Figure 4.5-3, is then given by:

$$V_o = \frac{\beta V_{IN} R_L}{R_S + R_\pi} (1 - e^{-t/R_p C_T})$$

The final output current can be calculated from Figure 4.5-3 to be 60 mV/25 ohms = 2.4 mA. Given beta = 34 at this current (see Figure 4.5-11) $R_\pi$ can be calculated at the final output current:

$$R_\pi = \beta V_T / I_E = 368 \text{ ohms}$$

The total base resistance and therefore $R_B$ can be calculated from the final voltage gain (e.g., 60 mV/200 mV):

$$A_V = \beta R_L / (R_S + R_\pi)$$

$$R_S = 2K + R_B = \beta R_L / A_V - R_\pi = 2465 \text{ ohms}$$

$$R_B = 465 \text{ ohms}$$

4-20

(a)



$R_P = (R_S + R_B) \parallel R_\pi$

(b)
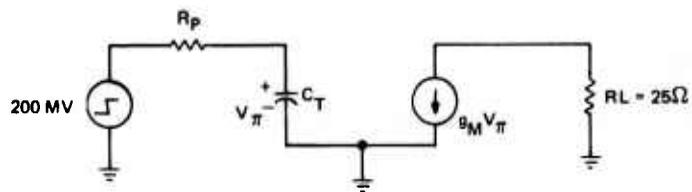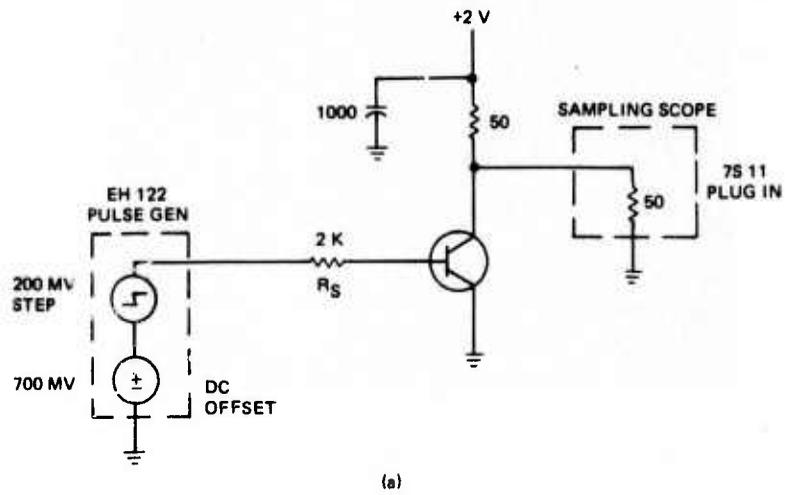
Figure 4.5-2. Test set up for $f_T$ measurement (a), and ac equivalent circuit (b).
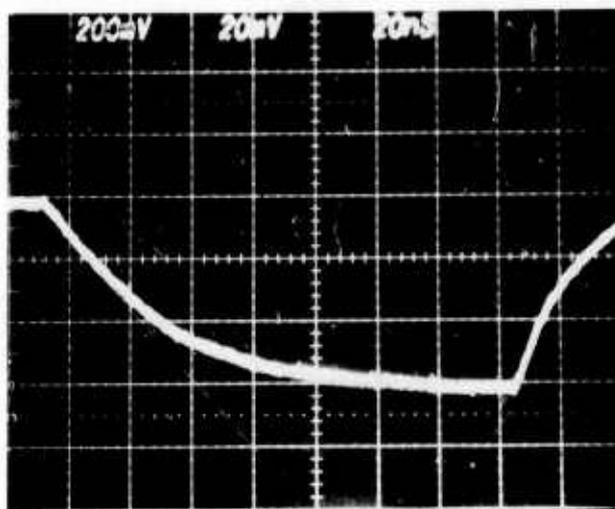
Figure 4.5-3. Output Waveform
for step input.

Note that variations of $R_B$ with output current will be effectively swamped out by the 2K series resistor.

From Figure 4.5-3, $V_0 = 25$ mV at $t = 22$ nsec, and therefore $I_E = 1$ mA. Then using the previous equations:

$$R_\pi = 936 \text{ ohms}$$

$$R_S = 2.47 \text{ K}$$

$$R_p = 679 \text{ ohms}$$

$$e^{-t/R_p C_T} = 0.527$$

$$R_p C_T = 34.3 \text{ nsec}$$

but $$R_p C_T = R_S (R_\pi C_T)/(R_S + R_\pi)$$

$$= \beta R_S / (R_S = R_\pi)(2\pi f_T)$$

Then $$f_T = 121 \text{ MHz at } I_E = 1 \text{ mA.}$$

REFERENCES (4.5)

4.5-1  J. R. Gaskill Jr.  L. R. Weill  J. H. Flint, E. A. Kelley
       J. W. Klinchock, "Maximum Data Rate Logic Array Development."
       AFAL TR-74-221, Dec. 1974  Vol-11  Appendix A  "Transistor
       Models and Parameter Measurement Techniques."
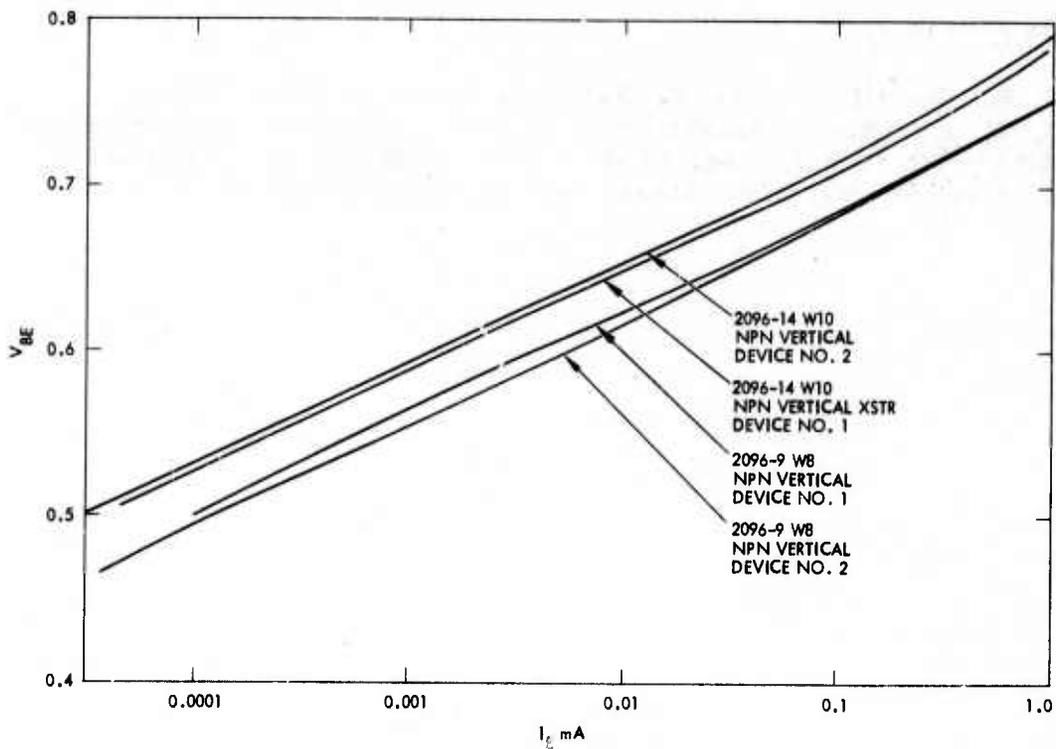
Figure 4.5-4. Transistor base-emitter voltage as a function of collector current for four devices.
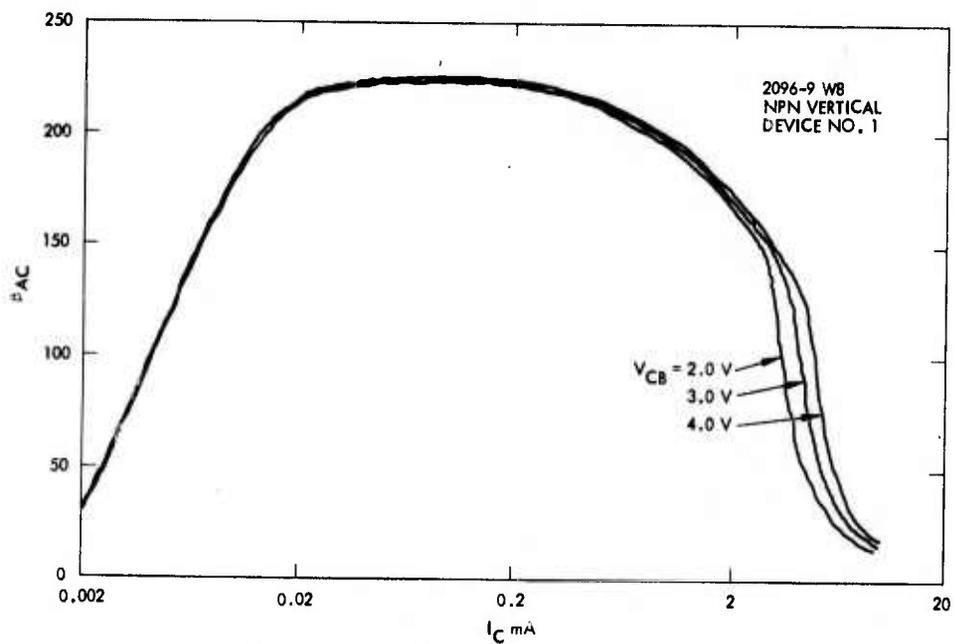


Figure 4.5-5. Gain ($h_{fe}$) vs emitter current.

4-24
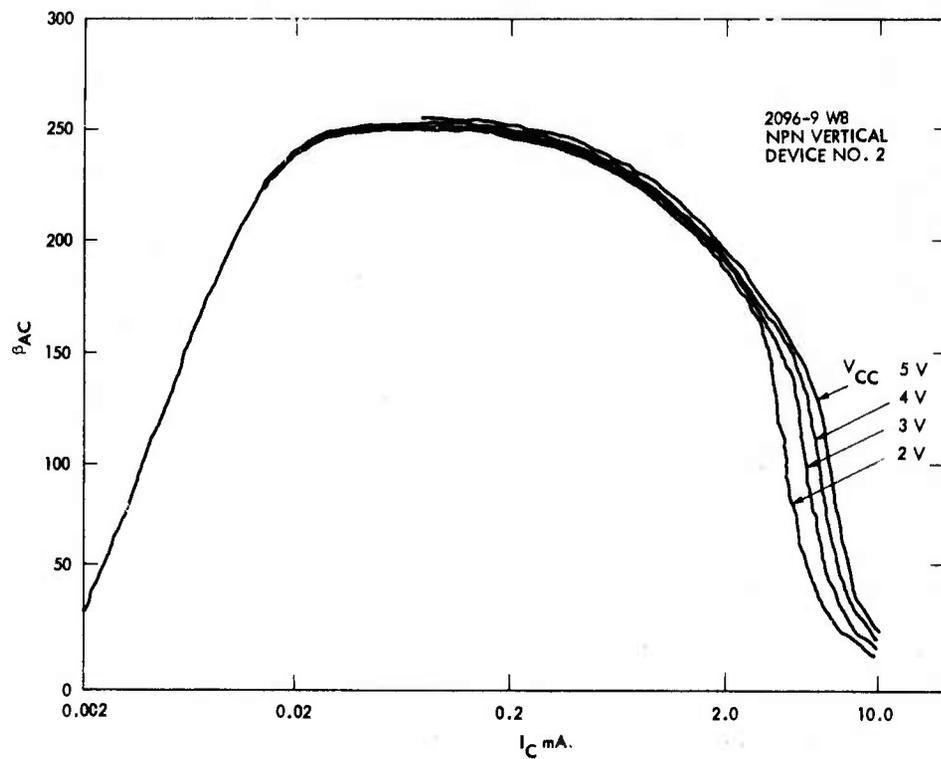
Figure 4.5-6.   Gain ($h_{fe}$) vs emitter current.
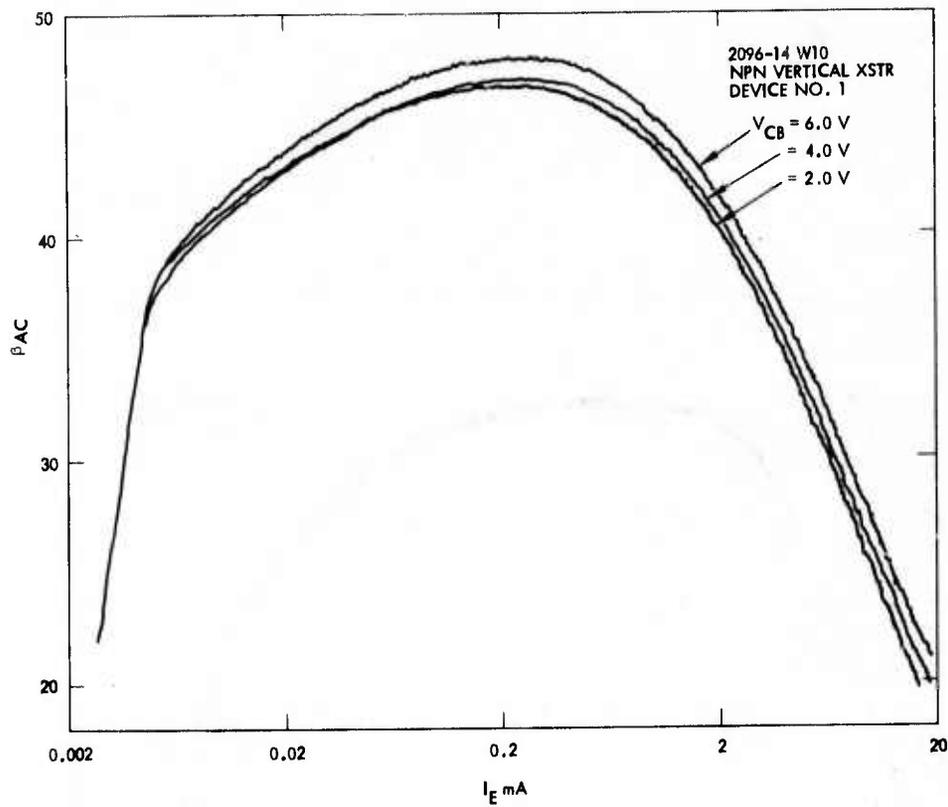
Figure 4.5-7. Gain ($h_{fe}$) vs emitter current.
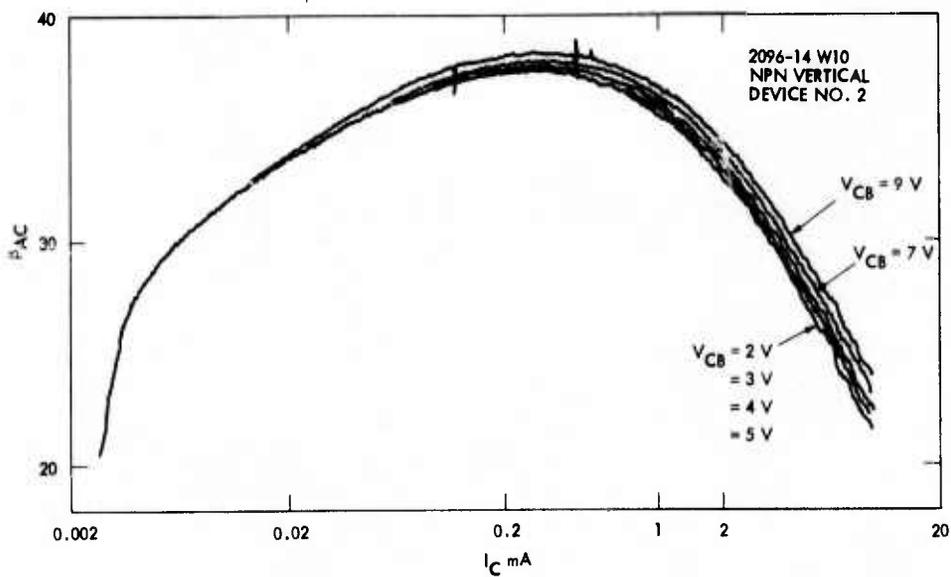


Figure 4.5-8. Gain ($h_{fe}$) vs emitter current.
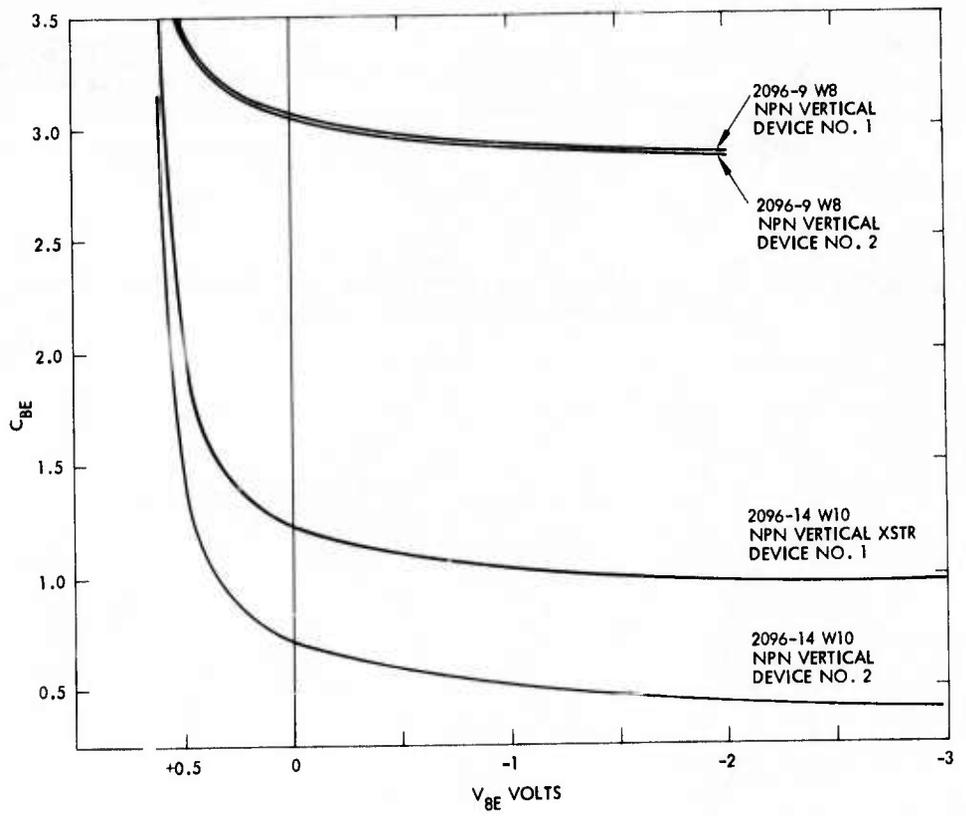
Figure 4.5-9. 1 MHz base-emitter junction capacitance as a function of junction voltage.
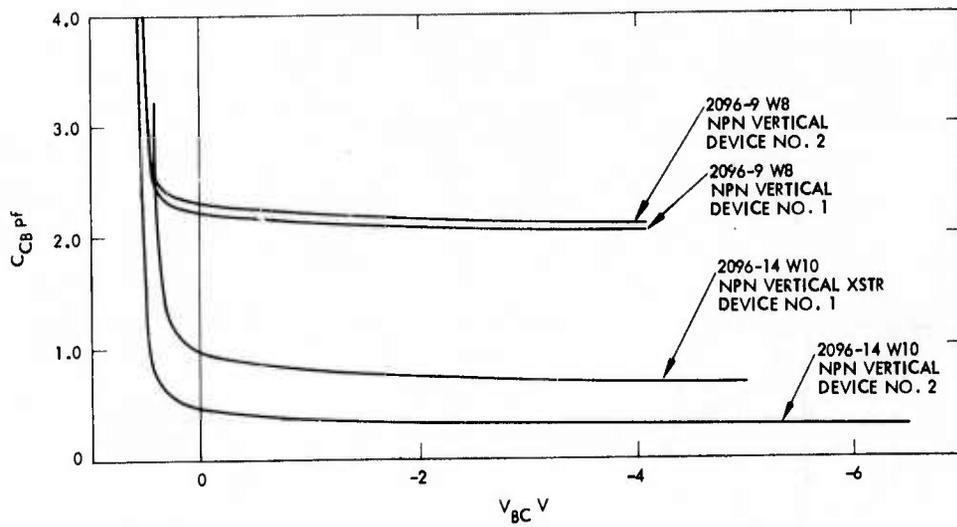
Figure 4.5-10.  1 MHz base-collector junction capacitance
as a function of junction voltage.
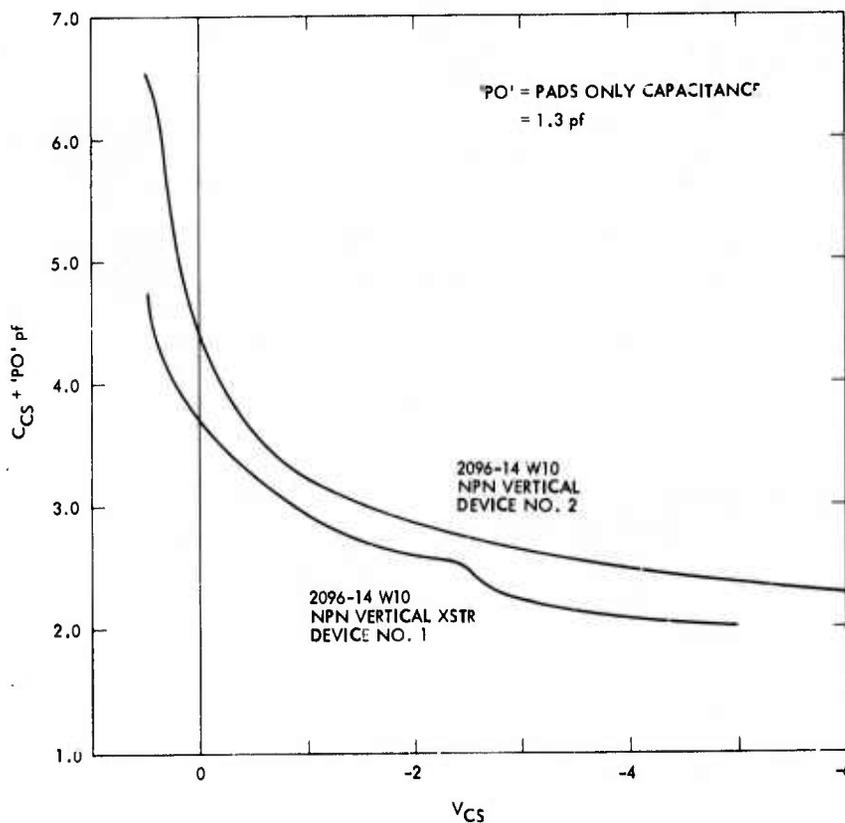


Figure 4.5-11.  1 MHz pad and collector to substrate
capacitance vs junction voltage.

## 4.6 BIPMOS

The 2096 BIPMOS (bipolar-MOS) CCD output buffer demonstrates the compatibility of MOS and bipolar processes and examines one buffer configuration. The device was tested in the lab and the test results were compared with the computer analysis.

The device structure is shown in Figure 4.6-1. It consists of a vertical NPN bipolar output transistor with an N channel MOS transistor. The NPN bipolar is operated as an emitter follower for current gain and the MOS is operated common source for voltage gain. The circuit is shown in Figure 4.6-2. The device achieves the required voltage and current gain necessary for CCD output buffering.
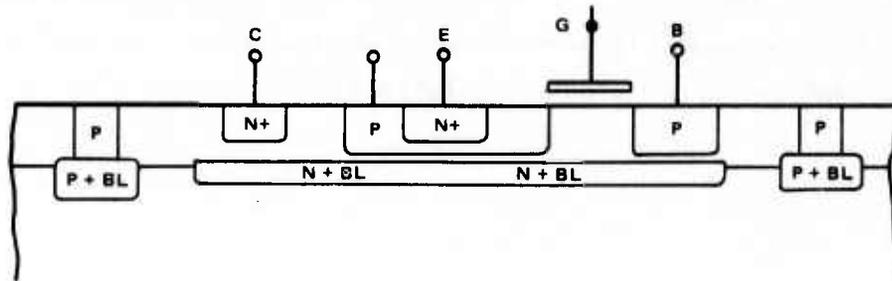
Figure 4.6-1. BIPMOS structure.
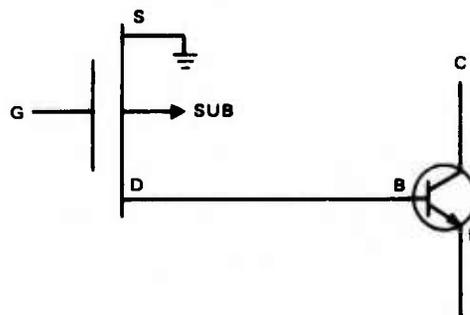
Figure 4.6-2. BIPMOS circuit diagram.

The BIPMOS device was modeled using the SPICE computer aided design program in the configuration shown in Figure 4.6-3. The circuit was analyzed for several values of base resistor using device parameters calculated from physical dimensions and parameters measured on the bipolar transistor as described in Section 4.5.

Tables 4.6-1 and 4.6-2 show the model parameters used for the bipolar and MOS portions of the device. Figure 4.6-4 shows the measured and calculated curves of optimum gate bias and voltage gain versus base resistance. Bandwidth versus base resistance is shown in Figure 4.6-5. Attachment 4.6-1 is a typical computer run showing dc transfer and ac analysis plots.

The results of the computer analysis show reasonable correlation with measured data, indicating a useful model. The BIPMOS device has demonstrated the compatibility of processes and could be a useful output buffer using a high frequency bipolar transistor.
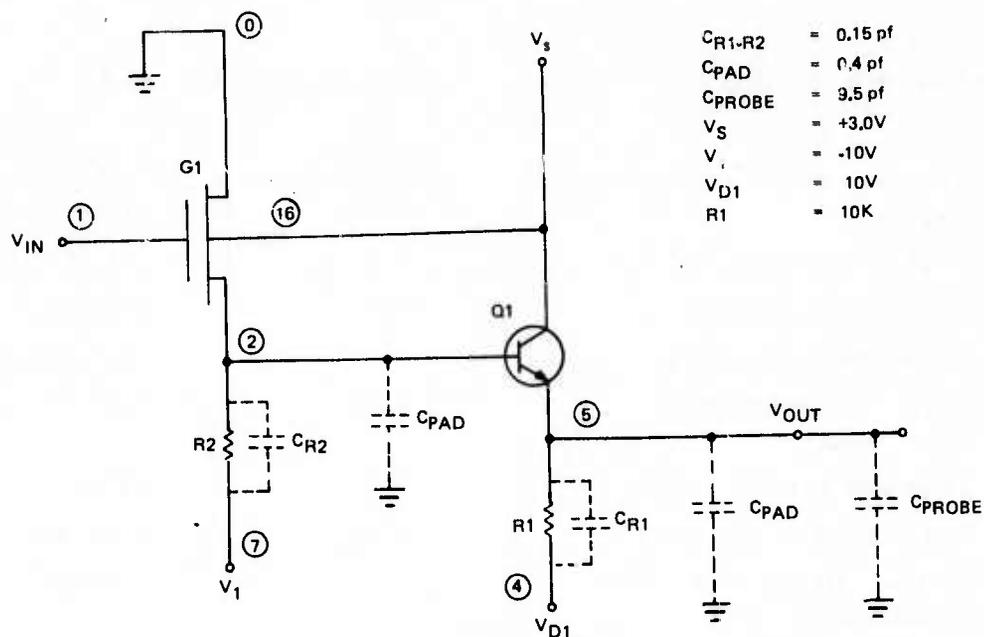
Figure 4.6-3.  2096 BIPMOS model.
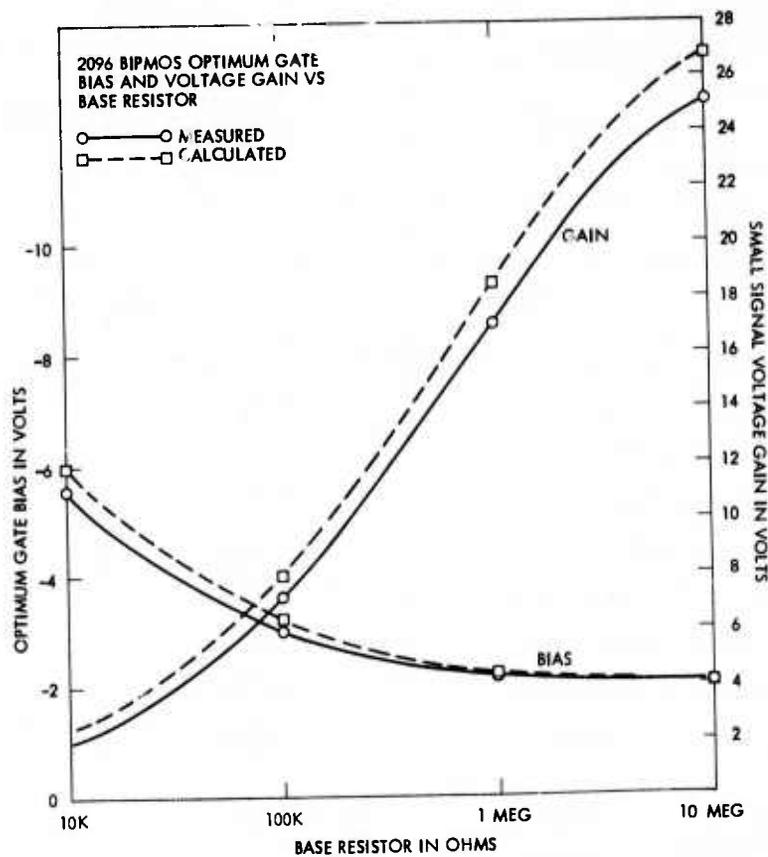


Figure 4.6-4.  2096 BIPMOS optimum gate bias and
voltage gain versus base resistor.

TABLE 4.6-1.  BIPMOS MODEL PARAMETERS — BIPOLAR

| | Parameter | Symbol | Value |
|---|---|---|---|
| 1) | Forward beta | BFM | 210* |
| 2) | Base resistance | RB | 1.6 K ohms |
| 3) | Emitter resistance | RE | 11 ohms* |
| 4) | Collector resistance | RC | 12.5 ohms* |
| 5) | Collector-substrate capacitance | CCS | 3.0 pf* |
| 6) | Forward transit time | TF | 83 ps |
| 7) | Base-emitter junction | CJE | 3.06 pf* |
| 8) | Base-collector junction capacitance | CJC | 2.23 pf* |
| 9) | Forward knee current | IK | 1.12 amps |
| 10) | Base-emitter junction potential | PE | 0.7 V |
| 11) | Base-collector junction potential | PC | 0.7 V |

*Measured values.

TABLE 4.6-2.  2096 BIPMOS MODEL PARAMETERS — MOS

| | Parameter | Symbol | Value |
|---|---|---|---|
| 1) | Threshold voltage | VTO | 1 V |
| 2) | Surface potential | Phi | 0.7 V |
| 3) | Transconductance | Beta | $3.0 \times 10^{-5}$ |
| 4) | Bulk threshold | Gamma | 0.73 V |
| 5) | Gate-source capacitance | CGS | 0.026 pf |
| 6) | Gate-drain capacitance | CGD | 0.013 pf |
| 7) | Gate-bulk capacitance | CGB | 0.46 pf |
| 8) | Base-drain junction capacitance | CBD | 0.09 pf |
| 9) | Base-source junction capacitance | CBS | 0.09 pf |
| 10) | Bulk junction potential | PB | 0.7 V |
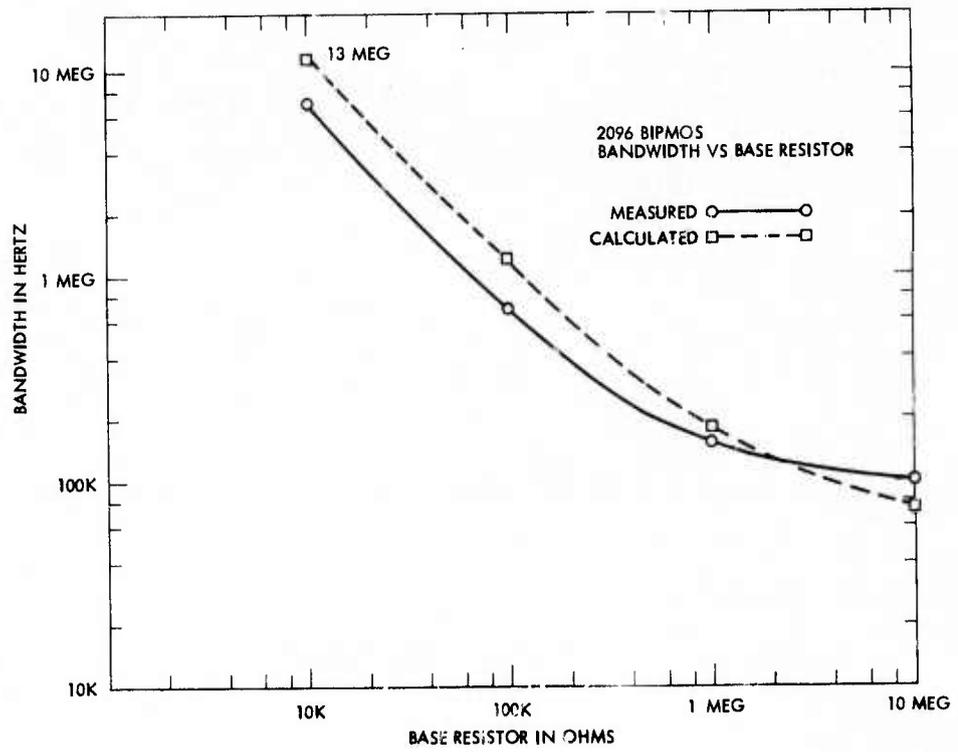
Figure 4.6-5. 2096 BIPMOS bandwidth versus base resistor.

DATE 18-Nov-75    TIME 10:30    VERSION 1A

```
00100  BIPMOS
00200  G1 2 1 0 16 M1
00250  V3 16 0 DC 3
00300  Q1 16 2 5 B1
00400  MODEL M1 PMO VTO=1. PHI=.7 BETA=3.0E-5 GAMMA=.73 CGS=.026P CGD=.
013P CGB=.463P CBD=.09P CBS=.09P PB=.7
00500  MODEL B1 NGP BFM=310 RB=1.6K RE=11 RC=12.50 CCS=3.00P TF=83P CJE
=3.06P CJC=2.23P IK=1.12 PE=.7 PC=.7
00600  R1 5 4 10K
00630  C1 2 7 0.55P
00640  C2 5 4 10.05P
00650  R2 2 7 100K
00670  V1 7 0 DC -10
00700  VIN 1 0 DC -3.20 AC 0.1
00800  VD1 4 0 DC -10
00950  AC DEC 4 1K 1G
01000  NP
01050  DC TC VIN -1 -4    0.25
01100  OUTPUT VOU 5 0 PLOT DC MA 3.162E-2  3.162E.0 PRINT MA
01200  NB
01400  END
```

DC TRANSFER CURVES

```
     VIN        VOU
         -1.000E+01    -7.892E+00    -5.784E+00    -3.676E+00    -1.568E+00
         - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-1.000E+00 +                .             .             .             .
-1.250E+00 +                .             .             .             .
-1.500E+00 +                .             .             .             .
-1.750E+00 +                .             .             .             .
-2.000E+00 +                .             .             .             .
-2.250E+00 .+               .             .             .             .
-2.500E+00 .        +       .             .             .             .
-2.750E+00 .            +.                 .             .             .
-3.000E+00 .                .        +     .             .             .
-3.250E+00 .                .             .        +     .             .
-3.500E+00 .                .             .             .        +     .
-3.750E+00 .                .             .             .             +.
-4.000E+00 .                .             .             .             +
         - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

## Attachment 4.6-1 (continued)

```
AC ANALYSIS
 FREQUENCY        VOU
   (HZ)          MAGN
 1.00E+03      8.004E-01
 1.78E+03      8.004E-01
 3.16E+03      8.004E-01
 5.62E+03      8.004E-01
 1.00E+04      8.004E-01
 1.78E+04      8.003E-01
 3.16E+04      8.001E-01
 5.625E+04     7.996E-01
 1.00E+05      7.979E-01
 1.78E+05      7.927E-01
 3.16E+05      7.768E-01
 5.62E+05      7.338E-01
 1.00E+06      6.295E-01
 1.78E+06      4.658E-01
 3.16E+06      2.985E-01
 5.62E+06      1.759E-01
 1.00E+07      9.932E-02
 1.78E+07      5.409E-02
 3.16E+07      2.743E-02
 5.62E+07      1.204E-02
 1.00E+08      4.290E-03
 1.78E+08      1.242E-03
 3.16E+08      3.174E-04
 5.62E+08      9.526E-05
 1.00E+09      2.332E-05
```

```
AC ANALYSIS
 FREQUENCY    MAGNITUDE OF    VOU
            3.162E-02    9.999E-02    3.162E-01    1.000E+00    3.162E+00
            - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
 1.000E+03 .               .            .            +  .                 .
 1.779E+03 .               .            .            +  .                 .
 3.162E+03 .               .            .            +  .                 .
 5.623E+03 .               .            .            +  .                 .
 1.000E+04 .               .            .            +  .                 .
 1.779E+04 .               .            .            +  .                 .
 3.162E+04 .               .            .            +  .                 .
 5.623E+04 .               .            .            +  .                 .
 1.000E+05 .               .            .            +  .                 .
 1.778E+05 .               .            .            +  .                 .
 3.162E+05 .               .            .             +                   .
 5.623E+05 .               .            .           +     .               .
 1.000E+06 .               .            .        +  .                     .
 1.778E+06 .               .            .     +      .                    .
 3.162E+06 .               .         + .            .                     .
 5.623E+06 .               .     +      .            .                    .
 1.000E+07 .               +            .            .                    .
 1.778E+07 .          +    .            .            .                    .
 3.162E+07 +              .            .            .                     .
 5.623E+07 +              .            .            .                     .
 1.000E+08 +              .            .            .                     .
 1.778E+08 +              .            .            .                     .
 3.162E+08 +              .            .            .                     .
 5.623E+08 +              .            .            .                     .
 1.000E+09 +              .            .            .                     .
            - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

# 5.0 CONCLUSIONS

Existing LSI technologies have been reviewed and state-of-the-art new technologies that have potential for LSI have been identified. These are summarized in Figure 5.0-1, along with projections of the corresponding speed and power-delay products for the early 1980's.

Improvements in fabrication resolution using electron beam technology are expected to increase resolution in the next ten years by approximately an order of magnitude. These improvements will apply to all technologies. Insulating substrates and/or insulator logic cell isolation will further decrease capacitance and increase speed.

Assuming that industry and government funded research and development continue at their present levels, it is highly likely that at least one technology will be available in the early 1980's with logic LSI implementation capable of providing 0.4 nsec gate delays and a 0.08 p Joule power-delay product. For APSP design purposes, these figures are considered adequate. DMOS and CMOS/SOS will emerge as competitors and both may exceed that performance.

Thus more than an order of magnitude improvement in power-delay product and an order of magnitude reduction in gate delay beyond present LSI DMOS should be demonstrated capability by the early 1980's. This does not even consider advanced technologies yet to be conceived or reported. For example, if a combination of low voltage complementary DMOS on an insulator were to be implemented within the decade, utilizing advanced electron beam techniques, LSI gate delays of 200 pico seconds and power-delay products of 0.01 pico Joules might be expected, plus increased circuit density.

Should the potential of CMOS/SOS or DMOS fail to be realized, a backup alternative for APSP digital design using $I^2L$, provides an estimated factor of eight reduction in power-delay product, with about a factor of twenty increase in propagation delay. This is based upon the anticipated $I^2L$ technology of the early 1980's, i.e., 8 nsec delays and 0.011 pico Joule power-delay products.

Not apparent on power-delay curves, however, is the advantage of CMOS operating at lower than maximum speed relative to technologies that draw large standby current. The static power-delay product of present commercial CMOS is 0.002 pico/Joule at 5 volts due to leakage current. Unless the large usage (duty cycle) is very high, the effective power-delay product for systems is greatly decreased for CMOS since the power used is proportional to duty cycle. It becomes clear that very fast CMOS/SOS is preferred. An overall logic duty cycle of 10 percent (not unrealistic) makes 1982 CMOS/SOS competitive with $I^2L$ in effective power-delay with the additional benefit of higher speed (20X) for arithmetic functions requiring it.

Device LSI density will play a major role in the tradeoff. RAM memory will probably follow the same technology. Larger serial memories requiring high duty cycles may follow either CCD or $I^2L$ technologies depending on the overall per chip power.

Table 5.0-1 depicts present and projected densities and projected power-delay products. A figure of merit that combines gate area with power-delay is shown in the last column of the table. Though the figure of merit for CMOS/SOS appears lowest, duty cycle was not included in the figure of merit. As discussed, the duty cycle dependent power saving for complementary technologies (CMOS) is probably going to be the deciding factor. Duty cycle is strongly dependent upon system architecture and cannot be factored into a figure of merit at this time. An all encompasing figure of merit would also include in some manner, the computing power concept discussed in section 2.4.5. Qualitatively, CMOS/SOS leads $I^2L$ and DMOS in the above considerations.

The thermal limitations associated with dense high speed logic form a critical design limitation. Figure 5.0-1 includes a scale which assumes sufficient chip and header thermal conductivity to remove 100 m watts/mm$^2$.

TABLE 5.0-1. PROJECTED GATE DENSITIES AND POWER-DELAY PRODUCTS
FOR LEADING TECHNOLOGIES

| Technology | LSI Density, Gates/mm$^2$ | | LSI Power-Delay Product, pJ/Gate | | Figure of Merit: Density/Power-Delay, Kgates/mm$^2$-pJ | |
|---|---|---|---|---|---|---|
| | TYP. 1975 | Projected 1982 | TYP. 1975 | Projected 1982 | TYP. 1975 | Projected 1982 |
| I$^2$L | 110 | 1320 | 2.0 | 0.06 | 0.06 | 22 |
| DMOS | 140 | 1680 | 2.9 | 0.09 | 0.05 | 21 |
| CMOS/SOS | 90 | 1080 | 1.0 | 0.07 | 0.09 | 15 |
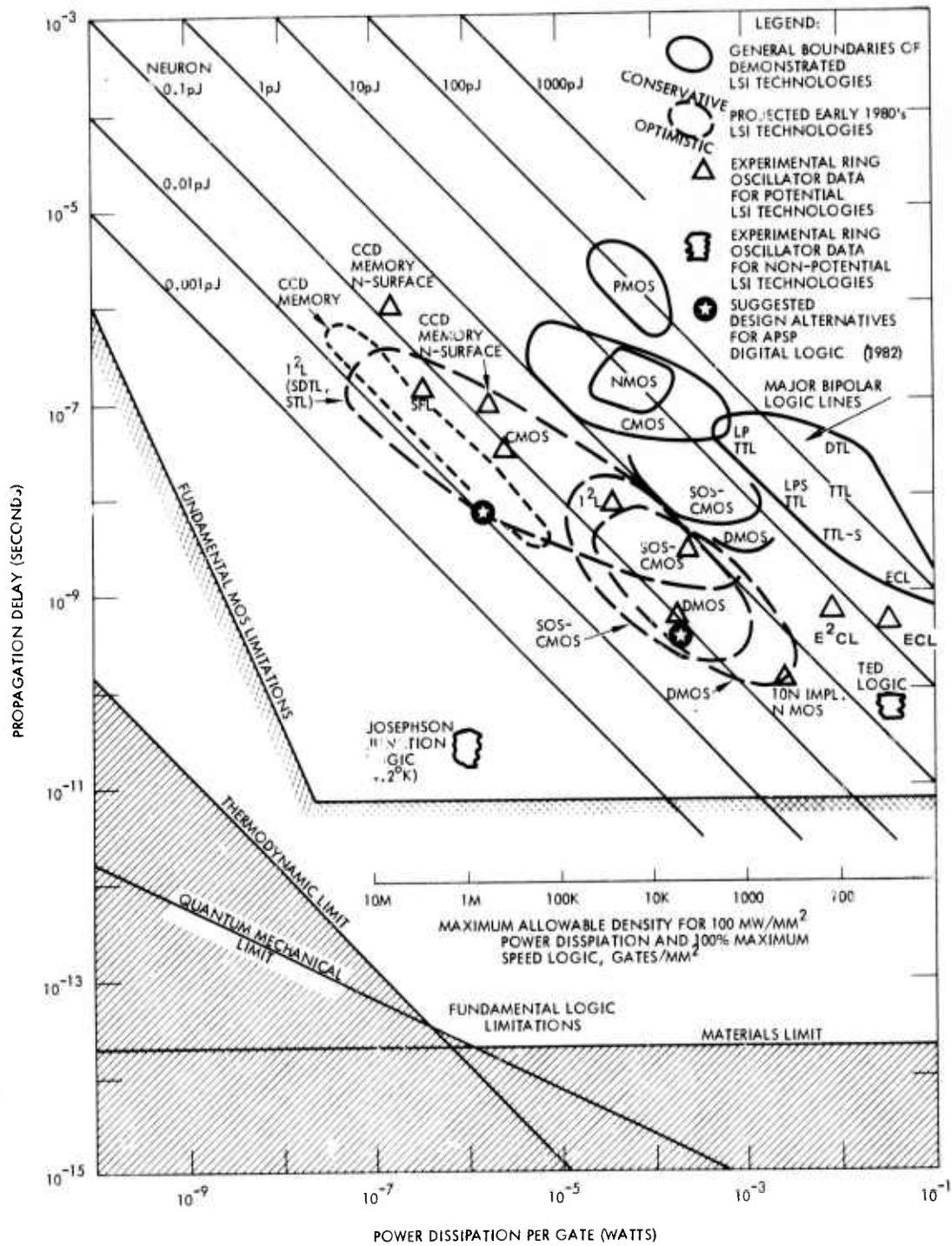
Figure 5.0-1. Power delay products of various technologies showing 1975 LSI, 1975 ring oscillator, and projected 1982 capabilities

This corresponds to 625 m watts for a 100 x 100 mil chip and is probably an upper bound for LSI space applications without liquid coolant. It thus appears that technologies exceeding 1000 gates/mm$^2$ with more than 1 m watt/gate dissipation may be thermally limited without associated advancement in LSI thermal design.

Analog CCD technology is expected to become increasingly popular in signal processing and other applications. The Adaptive Video Encoder is expected to exploit the unique characteristics of CCD's, both analog and digital. CCD logic appears to have application where associated with CCD memory, which has a strong future for serial memory systems.

Microprocessor evolution will continue at a tremendous rate of growth, providing more complex functions and memory per chip, and obtaining 25 to 60 nanosecond cycle times. CMOS technology will be widely used, followed probably by DMOS. CMOS/SOS, though attractive in performance is not likely to be pursued by commercial interests due to the cost of sapphire processing.

# CRITICAL DEVICE DESIGN
## FOR
# ADAPTIVE PROGRAMMABLE SIGNAL PROCESSOR

The views end conclusions conteined in this document are those of tha authors end should not be interpreted as necesserily representing the officiel policies, either expressed or implied, of the Advanced Reseerch Projects Agency or the U.S. Government.

| | |
|---|---|
| ARPA Order Number | 2954, Amendment No. 1 |
| Program Code Number | None |
| Neme of Contractor | Hughes Aircraft Compeny |
| Effective Dete of Contract | 20 June 1975 |
| Contract Expiration Date | 13 February 1976 |
| Amount of Contract | $498,159 |
| Contrect Number | F04701-75-C-0241 |
| Principel Investigator and Phone Number | K.E. Myers, 391-0711, X7598 |
| Project Engineer and Phone Number | K.A. Krause, 391-0711, X2243 |
| Short Title of Work | Critical Device Design for APSP |
| Date of Report | 30 January 1976 |
| Contract Period Covered by Report | 26 September 1975 to 23 January 1976 |

K.E. Myers
Program Meneger

Electro-Optical Division

AEROSPACE GROUPS

Hughes Aircraft Company ● Culver City, California

# CONTENTS

# LIST OF ILLUSTRATIONS

## 1.0 INTRODUCTION

This report presents the results of the Critical Device Design task of the Adaptive Programmable Signal Processor (APSP) program, and is identified as CDRL item A008 of Contract Number F04701-75-C-0241.

The report describes those circuit devices whose existence is critical to the successful development of the APSP. These include digital converters, both D/A and A/D, low-power mass memories, and a high speed (8M instructions per second) microprocessor.

Also included are descriptions and discussions of risk associated with the critical process technologies that need to be developed to permit fabrication of the above devices. These include high resolution, i.e. sub-micron, lithography using electron beams or x-rays in place of light waves, and silicon-on-an insulator technology.

Section seven of the report contains schedules for the design, development and test of several of the critical devices or processes identified earlier.

The report concludes with two appendices (separate cover) containing Hughes-proprietary information on the design and performance analysis of a company-funded CCD A/D converter test chip (CRC-100).

# 2.0 ADVANCED SEMICONDUCTOR PROCESSING

The phenomenal growth in the complexity of silicon integrated circuits during the nearly two decades of their existence can not proceed indefinitely. We have seen an approximate doubling in the number of components per chip each year to the present level of about $10^5$ components per chip. This growth has resulted from a 64-fold increase due to circuit design improvements, 20-fold reduction in linewidth by higher resolution lithography, and by a 12-fold increase in chip area. With the advent of electron lithography during the last few years, linewidths as narrow as 0.045 $\mu$m[1] are possible, (which exceeds the requirements of the APSP) compared to the average linewidth of about 5-7 $\mu$m for present-day integrated circuits. This would suggest that advanced lithography alone could carry us to approximately $10^9$ components per chip. However, Hoeneisen and Mead[2] have predicted that component density growth may flatten out at about $10^7$ components per chip for silicon integrated circuits because of fundamental device physics limits of the MOS field effect transistor. That is, for a MOSFET the minimum separation between source and drain can be no less than about 0.2 $\mu$m; a limitation imposed by source-drain punch through (doping 4 x $10^{17}$ cm$^{-3}$) and gate oxide (50 Å) breakdown. A more recent analysis by Klassen[3] suggests slightly greater minimum dimensions because of avalanche injection at the drain interface. Compared to present integrated circuits, even this remaining 100-fold growth possibility is very attractive for advanced satellite memory and microprocessor system applications, and moreover, attainable by means of electron beam lithography, advanced circuit design and advanced silicon processing methods.

## Lithographies

Advanced lithography and submicron device processing are the pivotal technologies for APSP memory development. The high density, low power memory systems required will depend heavily upon high resolution lithography (electron beam and x ray) and successful projection photolithographic processing of state-of-the-art serial CCD memory chips and CMOS logic circuits.

Because high resolution lithography is so important to the critical devices of the APSP, we next introduce several general methods for improved resolution and discuss image area/resolution limitations of photolithography and electron beam lithography.

Submicrometer lithography requires very short wavelength radiation. Electron, ion and short wavelength photon beams will play important roles in advanced lighography systems and there is even some limited improvement still possible in projection lithography systems by going to shorter wavelength light and smaller fields.

The diagram in Figure 2-1 depicts the variety of ways that electron beam systems are used for microcircuit microfabrication and diagnostics.



3119-1R2
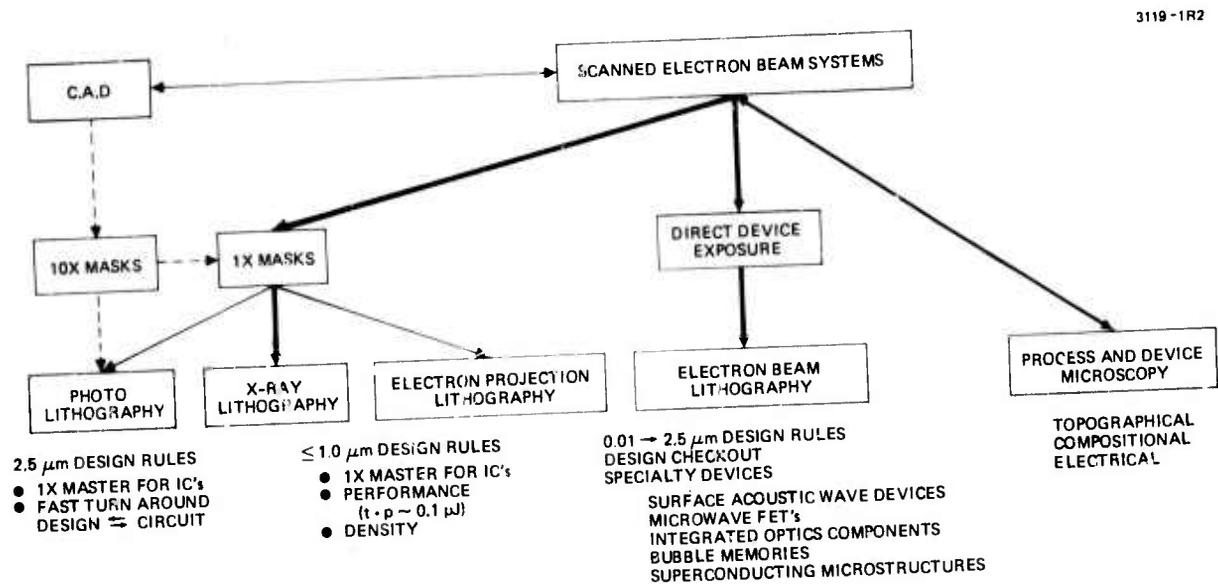
Figure 2-1. Electron beam/microelectronic device technology.

2-2

Electron beam diagnostic techniques, shown in the far right-hand column and only briefly mentioned here, are absolutely indispensable in electron beam lithography. Scanning electron microscopy (including compositional analysis) is the only way to examine critically devices and circuits with passive and active regions that are of the order of one square micrometer or less in surface area.

The utility of direct electron beam device exposure lies in experimental device fabrication for design checkout and in the fabrication of the highest performance (low volume) specialty devices and circuits. If device yields prove to be high, direct electron beam fabrication may be cost effective without replication for certain devices and circuits. Full-field replication by transmission x-ray lithography will probably form a viable basis of a batch production process.

## Image Area/Resolution Requirements

Electron beam/x-ray lithography offer considerable growth potential for higher resolution while convential projection photolithography has very little room for growth.

Because device and IC performance (and probably also fabrication yield) can benefit considerably from the use of higher resolution lithography, let us examine the requirements that such device fabrication places on the lithographic technique. Figure 2-2 diagrams one aspect of these requirements, viz., the pattern area coverage that is necessary in order to fabricate various classes of devices, together with the limits of photo- and electron-beam lithography. The right-hand curve, marked PH, indicates the maximum area that can be covered at a given resolution (minimum linewidth) by the best present-day optical and projection techniques. The region to the right of this line is accessible by photolithography, except that the upper limit shown has been approached only in careful R&D-type work. Production design standards are typically 5 μm over 2 in. The left-hand curve, marked EB, denotes the approximate limit of electron beam lithography, drawn to represent a scan field of 2 mm x 2 mm with 0.1 μm resolution. Above 2 mm the resolution limit is constant at 0.1 μm, due to the estimated

2-3
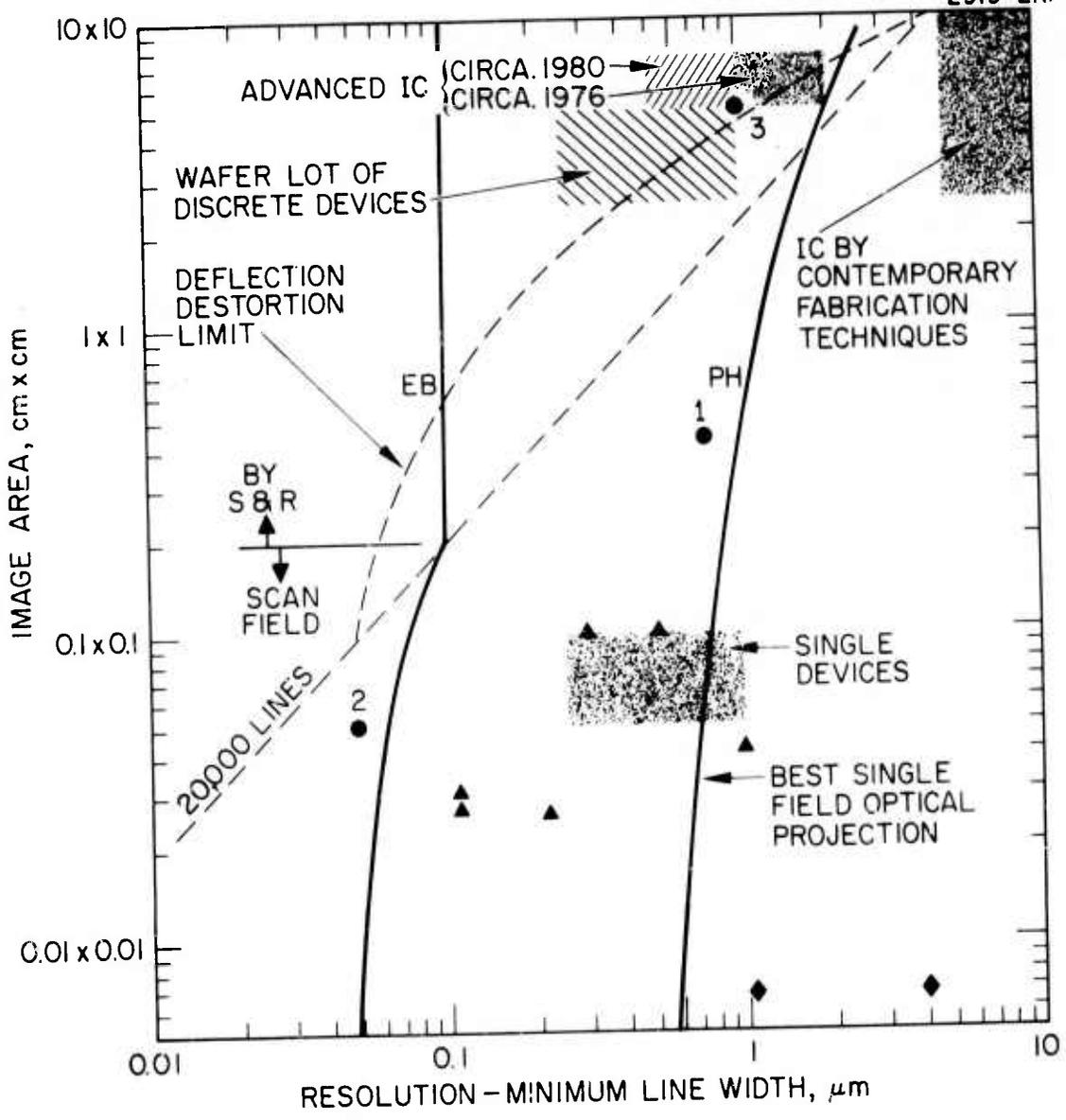
Figure 2-2. Image area and resolution requirements for advanced high resolution IC fabrication.

minimal errors in the step-and-repeat process that is required to obtain areas larger than 2 x 2 mm.

The approximate pattern areas required by discrete devices and integrated circuits are shown shaded, with a number of specific devices that have been built, shown by the points. In production, these discrete devices would be made in wafer-size lots and, therefore, require lithography capability with correspondingly larger area, as shown by the upper shaded regions. The dashed diagonal line represents about the largest single electron beam scan field that appears practical in the near term, due to beam deflection errors, electronic stability of the total system, and electron optics aberrations. It is clear that the regions of the figure representing the most interesting and highest performance devices and circuits of the future are inaccessible by photolithography; this technology is being pushed to its practical limit. However, these regions are well within the boundaries of electron beam lithography. This is an important point, because it indicates that the electron beam technology will not be used at the extremes of its capability, and that has favorable implications for yield.

The triangles on the chart represent acoustic surface wave delay lines made at IBM and Hughes. The diamonds represent MOS circuits fabricated using electron beam lithography, and the circle 2 represents Hughes' very high resolution electron beam resist work.[1]

The main point here is that the electron beam lithography permits patterns with linewidths as narrow as 0.05 μm to be exposed. In light optical techniques, pattern linewidth is limited by diffraction, scattering and interference effects. The narrowest line possible by photolithographic techniques is about 0.4 μm, using a comfortable mask to provide virtually zero spacing between mask and resist. In practice, linewidth values less than about one micron are very difficult to achieve by contact or projection photolithography.

## Electron Beam Lithography

Electron beam lithography is a maskless process utilizing both positive and negative resists that when combined with other beam processes offers submicron three-dimensional device "tailoring." The way in which

a single electron beam is used in microelectronic fabrication (refer to Figure 2-3) is to create a surface mask in resist on the substrate. This resist pattern is then used in any of the subsequent fabrication processes that requires pattern definition.

The basis of this technology is a finely focused electron beam that is deflected over a surface and blanked under digital computer control. The electron beam exposes the resist where it strikes. Subsequent development will either remove the exposed part (positive resist) or remove the unexposed part (negative resist). If this resist pattern is on the device being fabricated, it can then be used directly for any of the subsequent fabrication processes, as diagrammed in Figure 2-3. This process is used for those classes of devices requiring the highest resolution, or for low volume work (e.g., R&D) where replication of production quantities of identical patterns is not required. Alternatively, this pattern generation technique can be used to create a noncontacting replication mask, which can be used either as a shadow mask for x-ray or light exposure on the device substrate, or as the image from which electrons are focused onto the final resist by a suitable large-area electron optical system. Basically, the goal is to deflect, as rapidly and accurately as possible, a well-focused high current electron beam in a randomly addressable manner within as large an area (field) as possible; then mechanically to step and repeat this exposure field with an accuracy of 0.1 $\mu$m independently of or contiguously with similar fields over the entire substrate field, which may be as large as a 3-inch diameter silicon wafer.

## Hughes Programs in High Resolution Lithography (Brief Summary)

For the past seven years, Hughes Research Laboratories has carried out an R&D program on electron beam lithography. We presently have three Cambridge scanning electron microscopes capable of microfabrication and diagnostics. Each of these machines is under dedicated closed-loop minicomputer control for microfabrication; one of the instruments has been modified extensively for higher speed lithography. This facility is housed in a clean room in which most of the other key processes of silicon IC
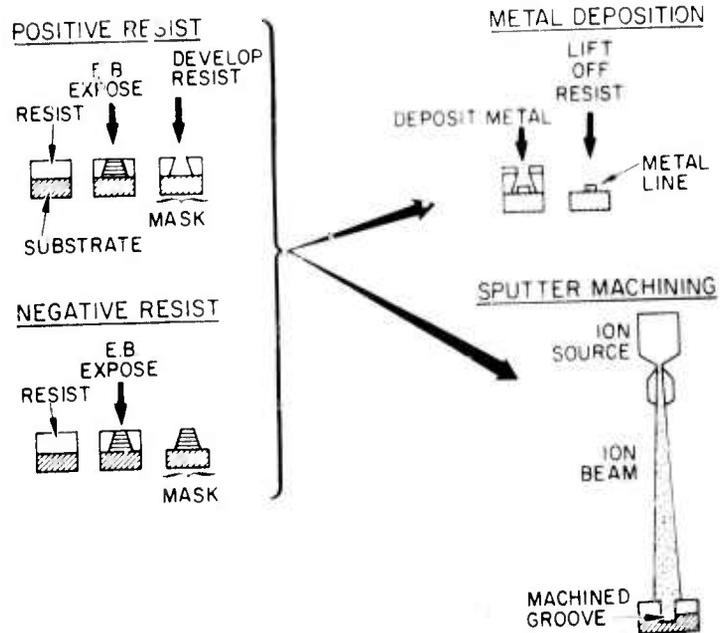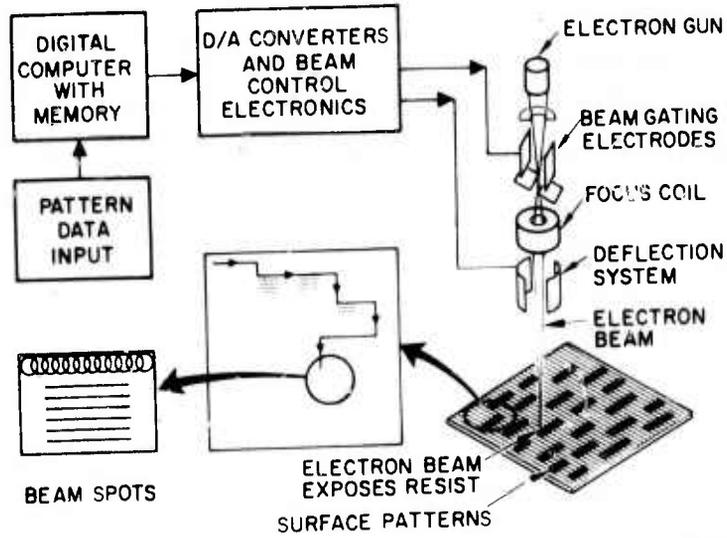
Figure 2-3. Electron beam lithography and beam microfabrication technology.

processing can be accomplished without transfer of the devices through a potentially contaminating environment.

We also have a program in high resolution replication, including both the x-ray and conformable mask methods. These processes are also housed in a clean environment.

The main elements of these programs and the evolutionary growth of our effort are summarized in Table 2-1. Two examples of this high resolution microfabrication which are applicable to submicrometer electrodes and gaps for CCD arrays is shown in Figure 2-4. The aluminum metallization pattern is for a 16-bit, 3-phase shift register, where the single level metal electrode separation is only 0.3 μm. More recently 0.6 μm linewidths have been obtained in polysilicon films for 2-level polysilicon gates. The resist pattern with 460 Å lines on 0.58 μm centers speaks for itself. This work is aimed toward advancing the state of the art in device performance to provide means for improving our advanced electronic systems. The submicrometer lithographic systems presently in use and still under development represent three overlapping generations of high resolution lithography.

The earliest system is a scanning electron microscope (SEM) which is used primarily for diagnostics and for prototype device exploration and system development. The second electron beam system under development is considerably more advanced and is based on our experience with the SEM system. The present x-ray system is serving the purpose of system development and prototype device exploration via x-ray replication. We describe in the following pages the principal features of these experimental lithographic systems and the results obtained to date on these programs.

## TABLE 2-1. SUBMICRON DEVICE LITHOGRAPHY AND PROCESSING AT HUGHES

| R&D Tasks | Specific Topics | Results |
|---|---|---|
| Direction Electron Beam Fabrication | | |
| Device Processing | EB Resists (PMM, PGM, PVP) | 0.1 μm Exposed and developed lines (Ref. 1) |
| | Pattern Registration | Demonstrated ±0.1 μm over 1 mm x 1 mm (Ref. 4) |
| | Combined Beam Processes | JFET fabricated by using implantation, EB lithography, and sputtering (Ref. 5) |
| | Plasma Etch/Strip | Used to fabricate submicrometer structures in polysilicon |
| Device Fabrication | Surface Acoustic Wave Filters | BW 560 MHz: $f_c$ = 1.3 GHz (Ref. 6, 7) |
| | Integrated Optics (Guides/Couplers) | 1 μm guides: 3600 Å gratings (Ref. 8, 9) |
| | SBFETs (GaAs, Si) | Under development for x-hand (Ref 10) |
| | MOSFETs and CCDs | 0.5 to 1.0 μm gate lengths under development |
| New EB System | Deflection Coils, Amplifier | Fabricated, under test |
| | Electrostatic Beam Blanking | Fabricated, under test |
| | Laser/Computer-Controlled Stage | Fabricated, under test |
| Software/Firmware | Pattern Generation Registration Real-Time Process Control System Diagnostics | For devices and processes listed above (Ref. 6) |
| Replication | | |
| Conformable Glass | Mask Fabrication by EB Lithography | Chrome on thin glass with 0.6 μm linewidths |
| | Pattern Replication Contact Fixtures Resist Processing | Submicron lines Made in positive Photoresist |
| | Device Fabrication Substrate Cleaning Al Liftoff | SAW pulse compression filter fabricated with 635 fingers, 0.6 μm wide in each of two transducers |
| X-Ray | Mask Fabrication Silicon (by Contact Photo-lithography and EB lithography | Gold on 2 μm silicon, 2.5 μm linewidths accomplished (Ref. 11). 0.6 μm linewidths fabricated (Ref. 12) |
| | Mylar | In process |
| | Pattern Replication | Good results in PMM and metal acrylate. Fair results in KMNR 747 (Ref. 13) |
| | Registration Piezoelectric Stage Servo Electronics Alignment Mark Evaluation | 130 Å/V sensitivity, 10 μm range, 100 Hz bandwidth built and tested |
| | Device Fabrication SAW Device Microwave FET | In process Under development |

2-9
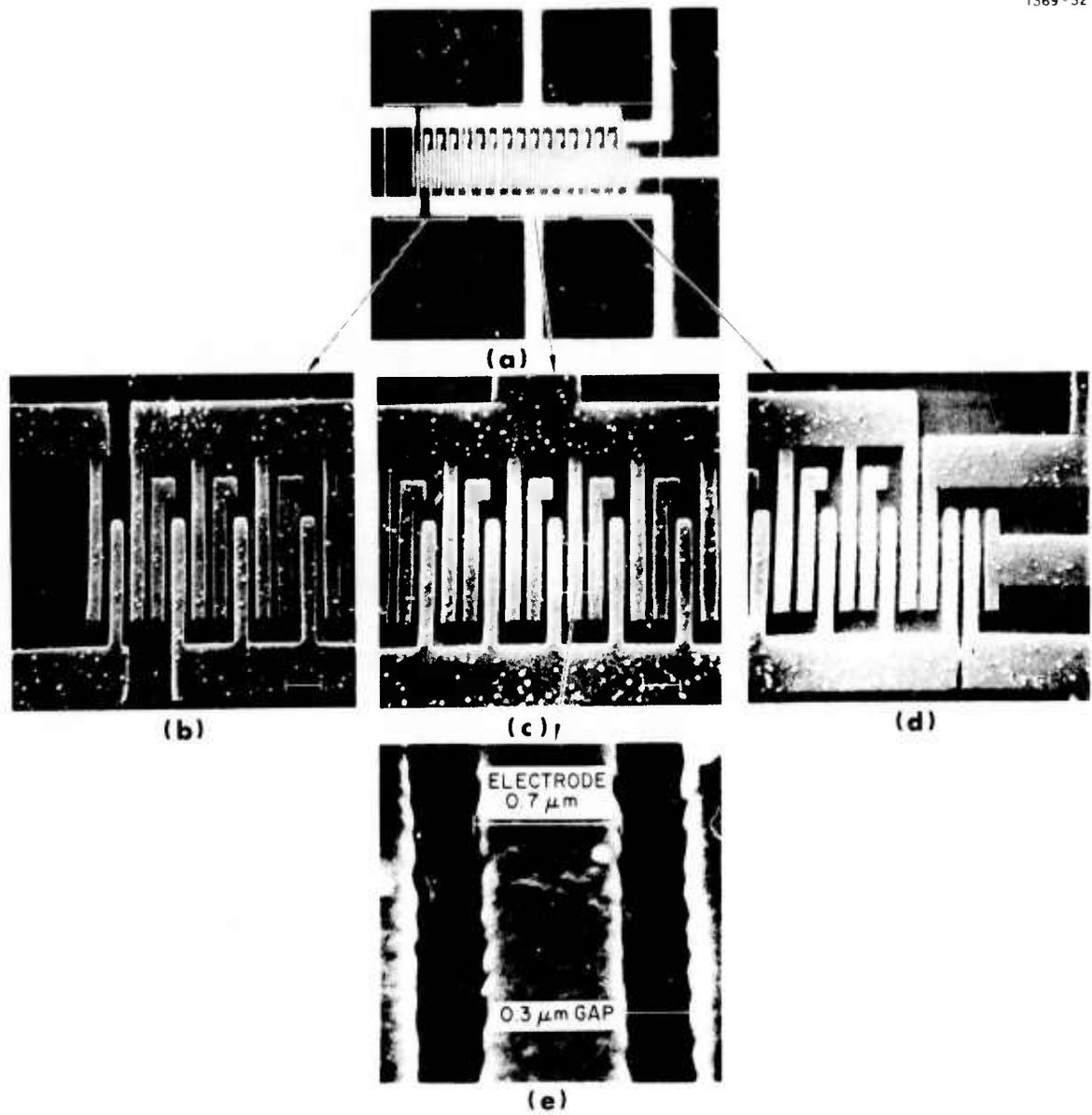
Figure 2-4.  Scanning electron micrographs
of very high resolution electron beam
lithography.  (a) Aluminum metallization
pattern for 3-phase 16-bit CCD; gap is
0.3 μm.  (b) - (e)  Exposed and developed
460 Å wide lines in PMMA.

## Examples of Devices

A number of devices have been fabricated by electron beam techniques. Ion implanted junction FET switches were fabricated first at Hughes by electron beam lithography and ion beam sputtering. Patterns in positive electron resist (PMMA) were used to define areas in the underlying metal layer that was subsequently ion beam sputtered and removed. Then the metal with the sputter etched pattern was used as an ion implantation mask. Figure 2-5 shows the device configuration. The width of the extended source-drain region which was implanted is 1 µm. In subsequent fabrication this width was reduced to 0.4 µm. This device was developed initially as a high conductance, low switching power microwave switch.
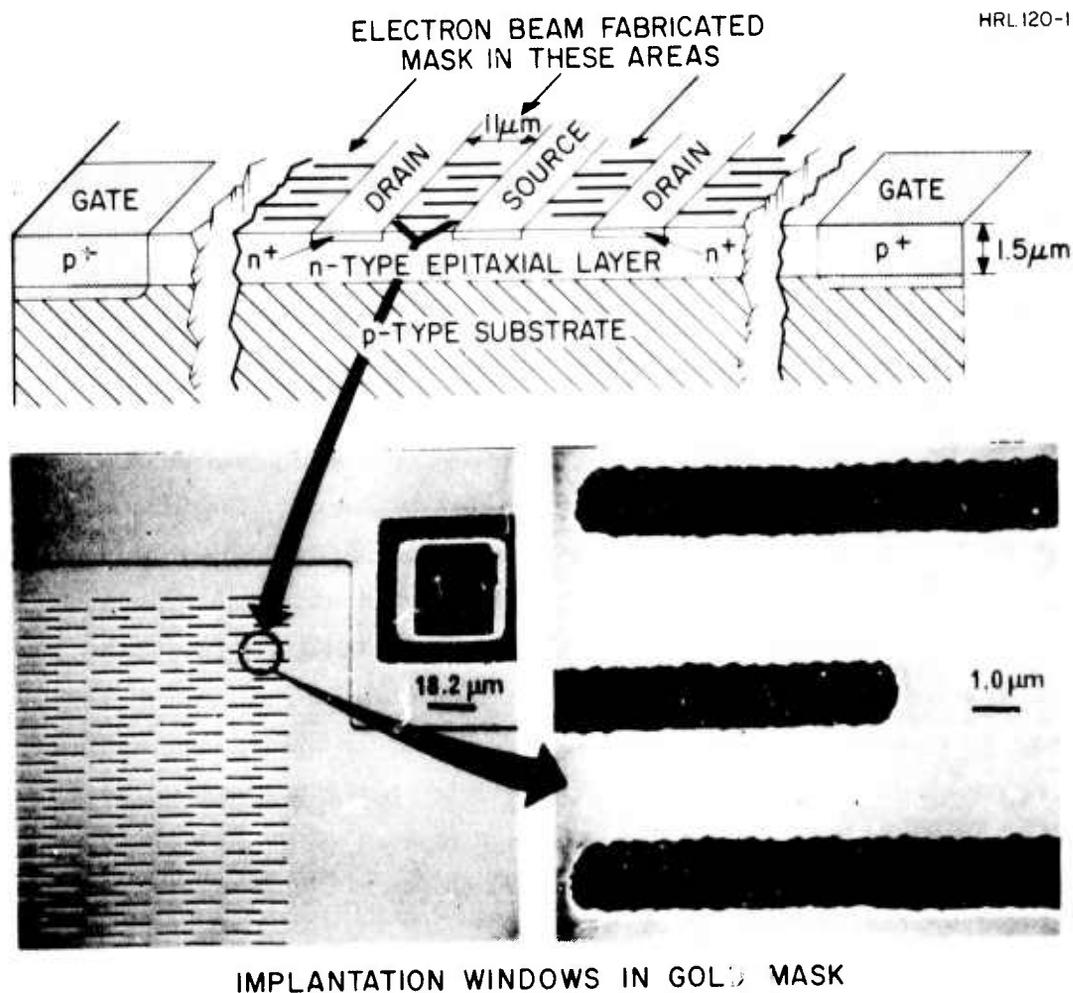


Figure 2-5. Electron- and ion-beam fabricated junction field effect transistor.

Both MOS FETS and CCD shift registers are under study. The Hughes electron beam microfabrication work on acoustic transducers has largely involved work in the 1.0 GHz region on lithium niobate, and as such, has not demanded ultimate resolution (typical metal lines were only 0.6 μm and larger in width). Some recent work has been in the area of pulse-compression filters, an example of which is shown in Figures 2-6 and 2-7. One of the transducers making up the filter is shown in Figure 2-6. The transducer has 634 electrodes covering a span of 0.892 mm. Electrode widths are 5000 Å. Width is held constant to within 8% over the entire array by employing specially developed process control software which determines optimum exposure conditions as a function of adjacent electrode spacing. Figure 2-7 shows the insertion loss for the filter as a function of frequency. The center frequency is 1.3 GHz and the bandwidth is 500 MHz.

## X-Ray Lithography

The x-ray replication technique is shown in Figure 2-8 and is similar to contact microradiography. A mask consists of a semitransparent substrate on which the desired pattern exists in a thin film highly absorbing to x rays. Electron beam lithography is used to generate these high resolution mask patterns. The mask is placed close to a wafer coated with a radiation-sensitive polymer film. A distant "point" source of x-rays, produced by a focused electron beam, illuminates the mask, thus projecting the shadow of the x-ray absorber onto the polymer film. This is the only feasible exposure scheme since efficient x-ray lenses and mirrors for collimation purposes have not yet been developed. The finite size of any real x-ray source leads to some blurring of the image, as illustrated by the insert in Figure 2-8. However, the mask-to-wafer spacing s, the source diameter d, and distance D, can always be chosen so that $\delta$ is small compared with the minimum line-width to be replicated. Limitations of conventional photolithography, such as diffraction and reflection, generally can be neglected since $\lambda \cong 10$ Å; 0.25 μm lines can be resolved for s as large as 60 μm (~2 mil).

Advantages of this approach include:

- Large area parallel exposure
- Mask fabricated by electron beam lithography
- 0.1 μm resolution

REFLECTION PHOTOMICROGRAPH

TRANSMISSION PHOTOMICROGRAPHS
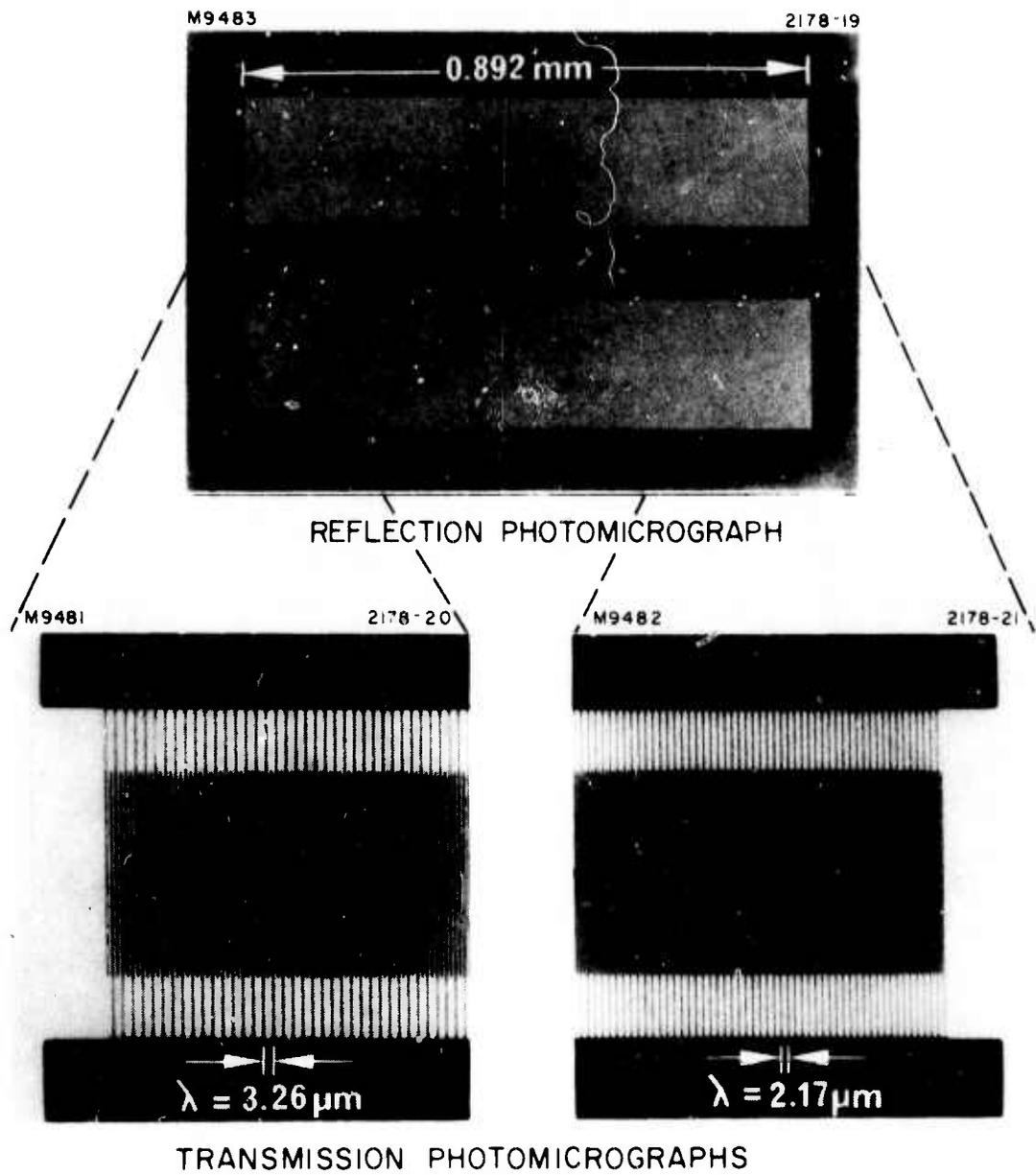
Figure 2-6. Linear fm transducer (Design No. 1).
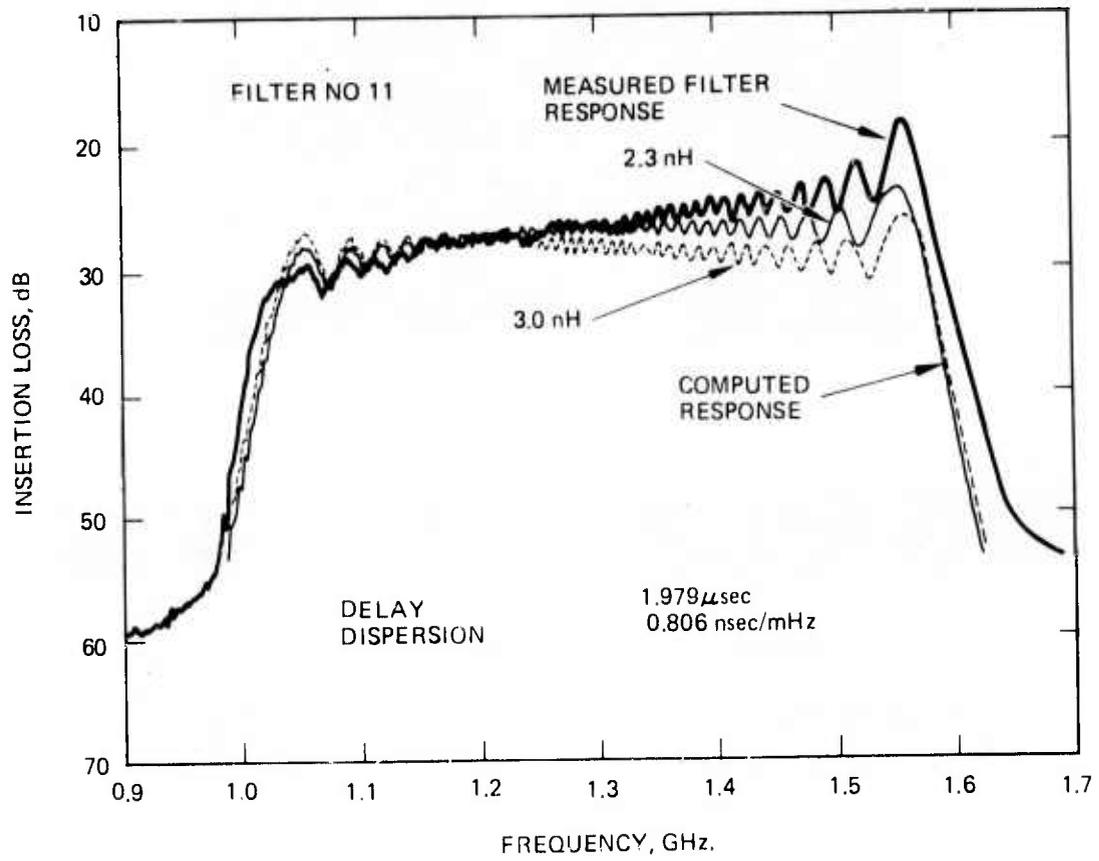
Figure 2-7. Insertion loss for the filter in Figure C-16.
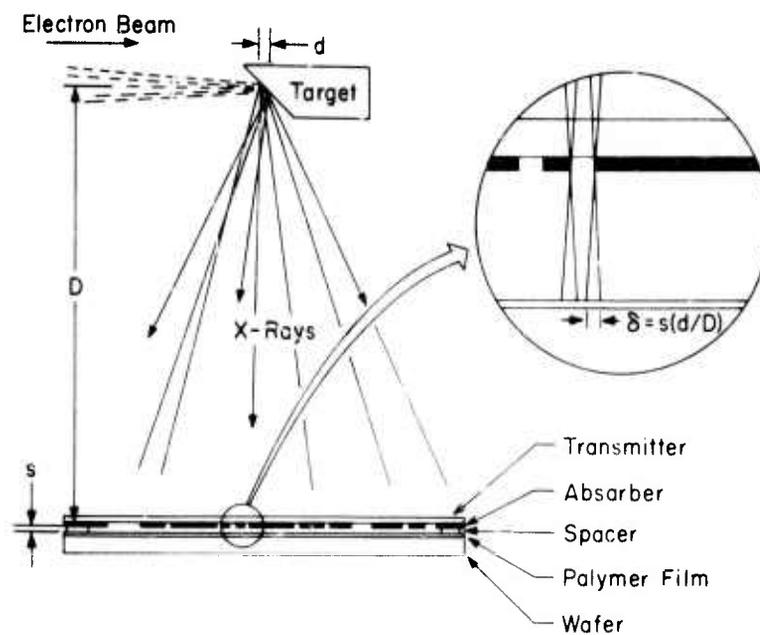
Figure 2-8.   Details of x-ray lithography.

- Off contact exposure (10 μm)
- Insensitive to dust and low atomic number contamination
- Use of positive or negative resists
- Uniform exposure with depth
- No requirement to place the mask or substrate in vacuum

X-Ray Mask

Proper construction of the mask is the key to x-ray lithography.
The substrate for the mask must transmit a reasonable fraction of the x-rays
and yet be self-supporting over the pattern area.   Single crystal silicon

membranes have been fabricated by the procedure shown in Figure 2-9.
X rays are efficiently attenuated only through absorption, a property that is
very sensitive to material and wavelength. A gold absorber has been used
to block x rays from an aluminum target (8.34 A wavelength). As shown in
Figure 2-9, electron beam lithography is used to create the high resolution
pattern in resist and then the pattern is ion beam etched into the gold.

### X-Ray Resists

All of the same characteristics listed above for electron resists are
required for x-ray resists. In addition, because the resist film only absorbs
a small fraction of the total incident x-ray energy, it is necessary to effect
greater absorption within the resist. Toward this end, Hughes effort to
incorporate heavy metal atoms, either as direct additives to the monomers
or as a soluble metal chelate, looks extremely attractive.

4425-3
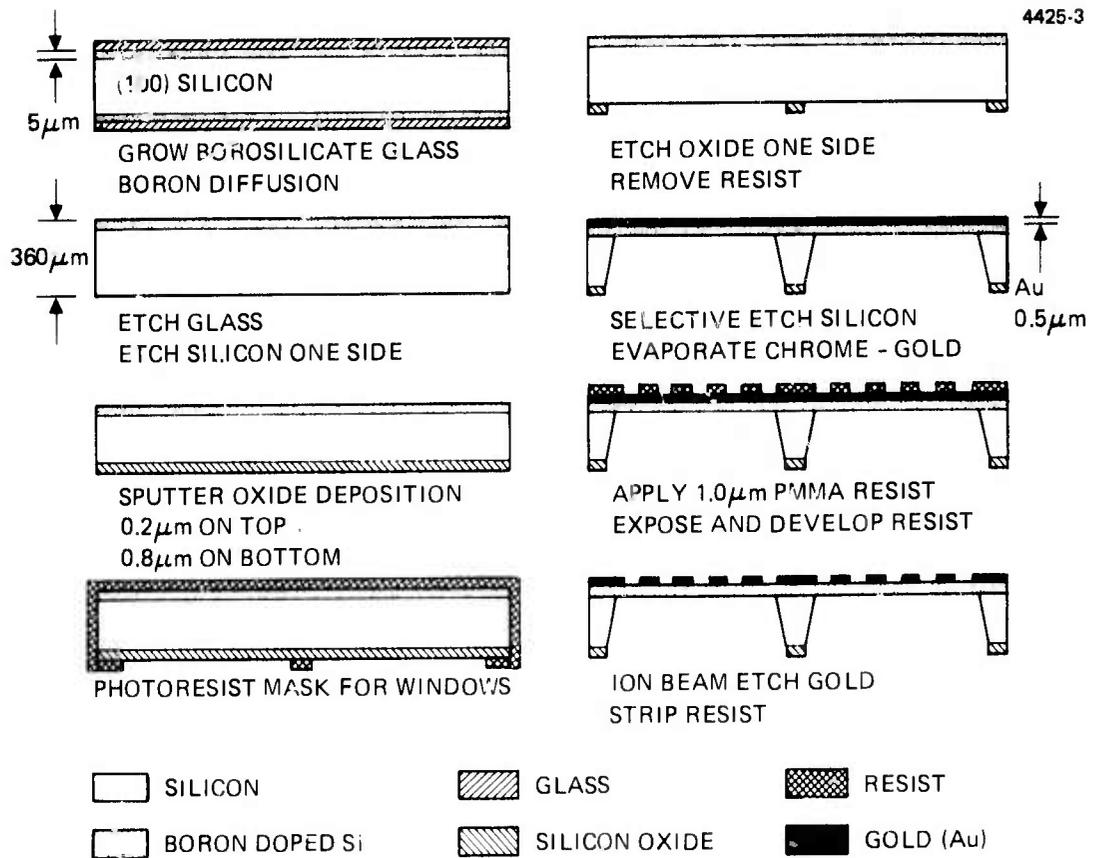
Figure 2-9. High resolution gold patterns fabricated by electron beam lithography and ion beam etching, on a thinned silicon membrane x-ray mask.

# REFERENCES

1. E. D. Wolf, F. S. Ozdemir, W. E. Perkins and P. J. Coane, "Response of the Positive Electron Resist Elvacite 2041 to Kilovolt Electron Beam Exposure," Record 11th Symp. on Electron, Ion and Laser Beam Technology, ed., R. F. M. Thornley, San Francisco Press, Inc., May 1971, pp. 331-336.

2. B. Hoeneisen, Mead, Solid State Electronics, Vol 15 p 819-829, 1972.

3. F. M. Klassen, Late News Paper 3.7, IEDM Meeting, Washington, D.C., 1974.

4. F. S. Ozdemire, E. D. Wolf and C. R. Buckey, "Computer-Controlled Scanning Electron Microscope System for High Resolution Micro-electronic Pattern Fabrication," Record 11th Symp. on Electron, Ion and Laser Beam Technology, ed., R. F. M. Thornley, San Francisco Press, Inc., May 1971, pp. 463-470.

5. E. D. Wolf, L. O. Bauer, R. W. Bower, H. L. Garvin and C. R. Buckey, "Electron and Ion Beam Fabricated Microwave Switch," IEEE Trans. on Electron Devices (Special Issue on Electron Beams in Micro-electronics) ED-17, 466 (1970).

6. E. D. Wolf, F. S. Ozdemir, R. D. Weglein, "Precision Electron Beam Microfabrication of Acoustic Surface Wave Devices," 1972 IEEE Ultrasonics Symposium, Monterey, California, November 1973 (Invited Paper J1).

7. E. D. Wolf, "Electron Beam Fabricated Acoustic Surface Wave Devices," Invited paper BC1 presented in Solid State Physics Symposium on "Synthetic Microstructures," 1974 Annual APS Meeting, Chicago, Illinois, February 4-8.

8. H. L. Garvin and E. D. Wolf, "Ion and Electron Beam Fabrication of Optical Components," Invited paper ThA6, Second Topical Meeting on Integrated Optics, New Orleans, January 1974.

9. E. D. Wolf, "Electron and Ion Beam Microfabrication," Chapter 7 Introduction to Integrated Optics, M. K. Barnoski, ed., Plenum Press, New York 1974.

10. F. S. Ozdemir, G. O. Ladd, E. D. Wolf, W. E. Perkins, N. Hirsch, F. W. Cleary, "Electron Beam Fabricated 0.5 $\mu$m Gate GaAs Schottky-Barrier Field Effect Transistor," 1974 International Electron Device Meeting, Washington, D. C. December 9-11, 1974.

# REFERENCES (Continued)

11.  J. H. McCoy and P. A. Sullivan, "Progress in X-Ray Lithography," presented at the Sixth International Conference on Electron and Ion Beam Science and Technology, San Francisco, May 13, 1974, (Electrochemical Society, Princeton, N. J., to be published 1974).

12.  P. A. Sullivan, "X-ray Lithography System," Final Report, November 1975, Contract F19628-75-C-0105 Air Force Cambridge Research Laboratories.

13.  R. G. Brault, "Properties of Metal Acrylate Compositions as X-Ray Resists," presented at the Sixth International Conference on Electron and Ion Beam Science and Technology, San Francisco, May 13, 1974, (Electrochemical Society, Princeton, N. J., to be published 1974).

# 3.0 ADAPTIVE SIGNAL ENCODER DESIGN

The Adaptive Signal Encoder (ASE) functional block diagram is illustrated in Figure 3.0-1. The ASE utilizes a predictive feedback technique to encode analog detector signals into 10 bit digital words. In addition, the ASE detects and erases samples affected by nuclear events and provides adaptive selection of proper operating modes for the MFPA and programmable spectral filter. ASE outputs are normalized to correct for responsivity variations within the MFPA, and provision is made for periodic automatic calibration.

In the following, each major functional block is described in detail, and critical device considerations are discussed.

## 3.1 DUAL RANGE A/D CONVERTER

As shown in Figure 3.1-1, the analog difference signal is first compared with an analog range threshold, which is usually A/32, where A is the largest signal magnitude in the dynamic range. The comparison result will be latched by a flip-flop which is reset for each word. If the analog difference signal is greater than A/32, switches $S_1$ and $S_2$ select the upper channel and the original signal is encoded by a 5 bit A/D converter. The encoder's digital output then contains only the first 5 MSB's with the 5 LSB's zero. If the analog difference signal is less than A/32, the lower channel is selected. The signal is amplified by a factor of 32, then encoded by the 5 bit A/D converter. A 5 bit shift register is placed after the A/D conversion; thus the first 5 MSB's are zero, only the 5 LSB's contain the signal information. The frequency of the incoming analog signal is 164 K samples/second (for a 10 Hz MFPA frame rate) and the output data rate is 1.64 M bits/sec.

3-1
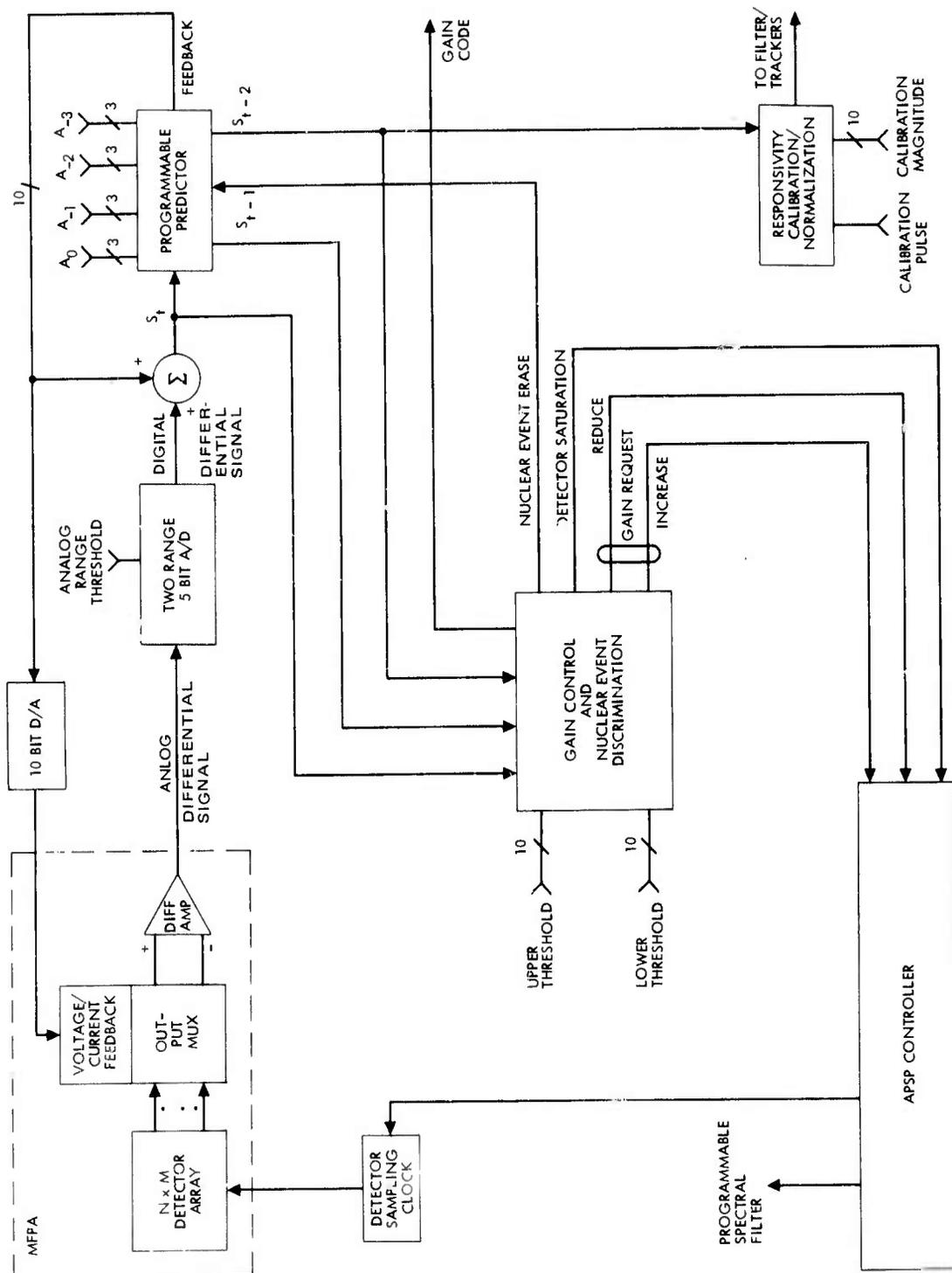
Figure 3.0-1. ASE functional block diagram.

3-2
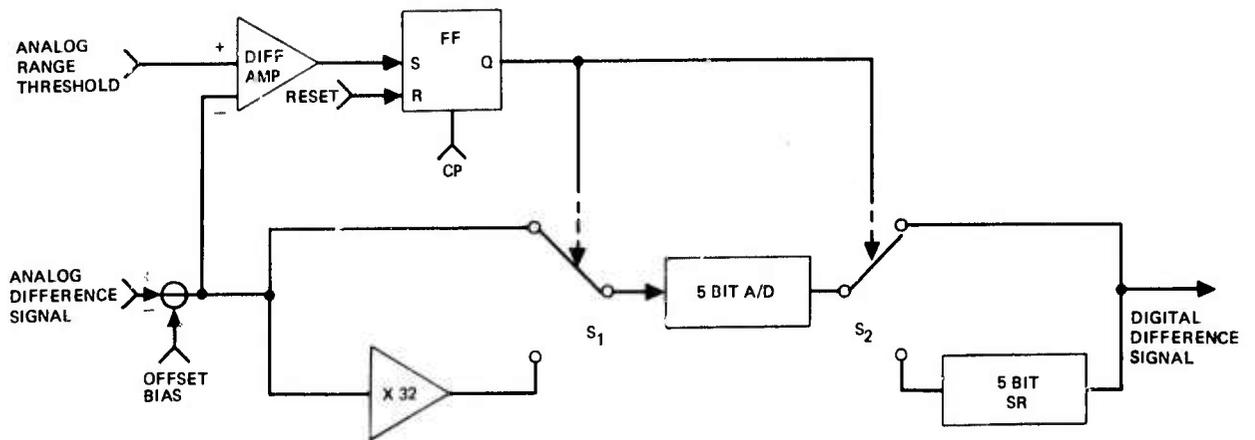
Figure 3.1-1.   Two range, 5 bit A/D converter.

It can be seen that this encoding scheme needs only a 5 bit A/D converter, yet avoids saturation when a relatively large difference signal is present.   For a large difference signal ($>A/32$), resolution is limited to $A/32$; however for typical inputs ($<A/32$), a finer resolution of $A/1024$ can be obtained.   In effect, five bit (32 level) resolution within any of 32 ranges (5 bits) is obtained.   The offset bias is used to accommodate fat zero and to shift the operating potential to allow for both positive and negative differences.

The converter of Figure 3.1-2 consists of a differential amplifier, a flip flop, an analog (X32) amplifier, a 5 bit A/D converter, a 5 bit shift register, and switches.   To perform efficient feedback when a large difference signal is present, it is important to encode the signal with higher accuracy than the A/D resolution indicates.   Thus 7 bit accuracy is required for the 5 bit A/D converter.   CCD A/D converters appear suitable for this application and are discussed in Appendix A and analyzed in Appendix B.

It should be pointed out that the analog range threshold input can be avoided if two 5 bit A/D converters are used as shown in Figure 3.1-2. Here the power consumption may be higher since there are two A/D converters.

3-3
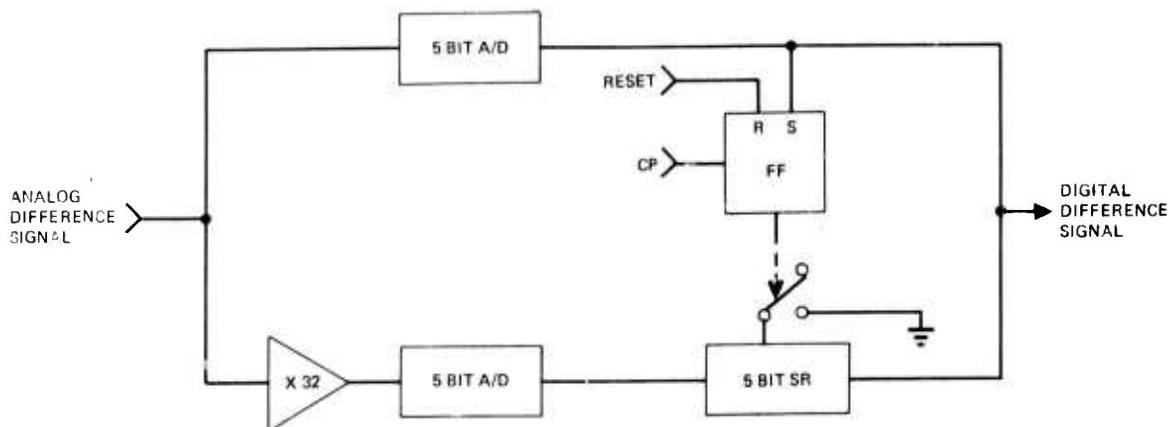
Figure 3. 1-2.   Different approach for dual range A/D converter.

## 3.2  PROGRAMMABLE PREDICTOR

The programmable predictor performs a polynomial data fit using n previous frame samples

$$P = \sum_{k=0}^{n} a_k S_{t-k}$$

As shown in Figure 3.2-1, the signal enters a series of (three shown) 164 K shift register memories.  The required samples for polynomial prediction are obtained by tapping the shift register memories at appropriate points.  The coefficients $a_k$ are programmable.  Table 3.2-1 lists typical values of $a_k$ for 1st order to 4th order predictions.  Simulations have shown all four configurations are stable.  After multiplications and summations the predicted value is sent through a 164 K shift register to the D/A converter and other feedback networks.   Indications are that first order prediction will be satisfactory.  The coefficients are programmable by the APSP controller.

The shift register memory is used to provide inputs for MFPA gain control and nuclear event discrimination.  In case a nuclear event is detected, the erase pulse enables the switch to ignore the output from the memory, which is a contaminated sample, and to replace it with the uncorrupted output from the A/D converter.  This is possible because a nuclear event is assumed to be an isolated saturation signal (see next subsection).
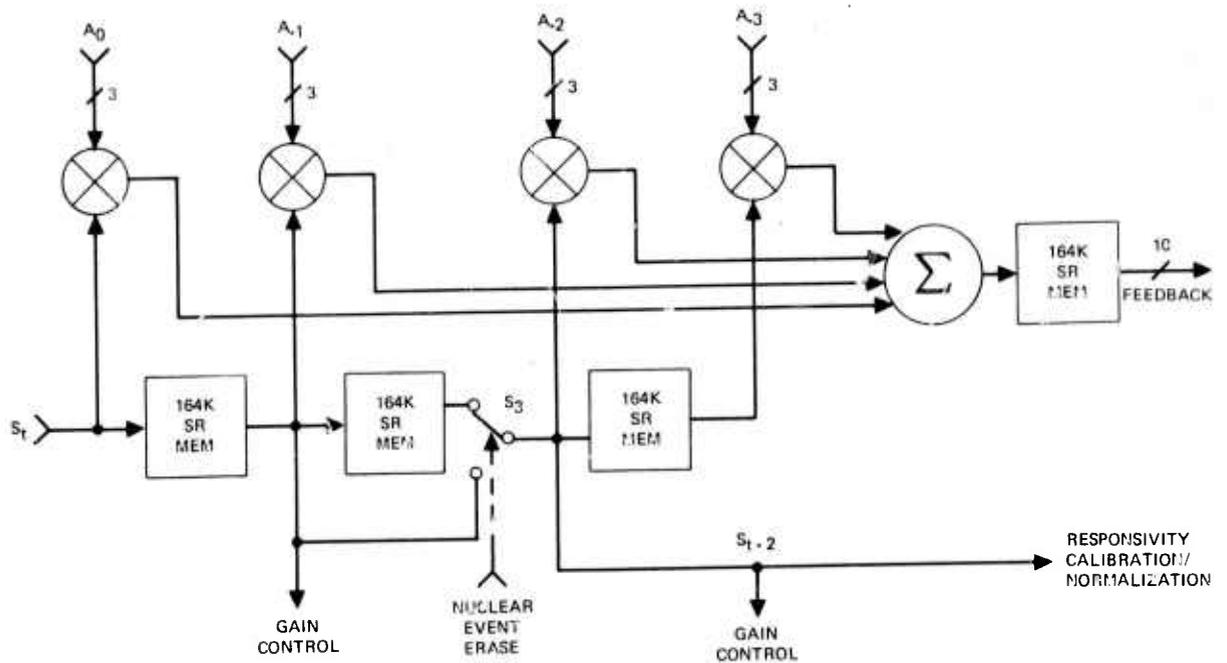
Figure 3.2-1. Programmable predictor.

TABLE 3.2-1. VALUES OF $a_k$

| Coefficients / Predictions | $a_0$ | $a_{-1}$ | $a_{-2}$ | $a_{-3}$ |
|---|---|---|---|---|
| 1st Order | 1 | 0 | 0 | 0 |
| 2nd Order | 2 | -1 | 0 | 0 |
| 3rd Order | 3 | -3 | 1 | 0 |
| 4th Order | 4 | -6 | 4 | -1 |

The design of an optimum predictor requires a knowledge of the input statistics. Here the input process is not well defined and for that reason the programmable predictor is the result of a deterministically-oriented design using the Newton interpolation (backward difference) technique which gives rise to binomial coefficients. As further information about the input characteristics is gathered, it becomes feasible to adapt the coefficients to obtain the best match to the input under a minimum mean square error criterion.

3-5

Note that a large number of shift register memories is required in this functional block unless first order differencing is utilized for the predictive encoder. Because of the serial nature of the data, CCD memories appear to be suitable for the application. Also since the input signal is of serial type, a serial-parallel multiplier appears applicable, which requires only three full adders in this case. The data rate is 1.64 M bits/sec as determined by the A/D conversion. The two range 5 bit A/D is more produceable than a 10 bit A/D and is expected to consume approximately 60% of the power required for a 10 bit A/D converter.

## 3.3 TEN BIT D/A CONVERTER

A CCD D/A converter using a charge division technique will be discussed in Appendix A and analyzed in Appendix B. An extension of the present design (8 bits) to a 10-bit device is required.

## 3.4 GAIN CONTROL AND NUCLEAR EVENT DISCRIMINATOR

In Figure 3.4-1 a gain control and natural nuclear event (i. e. , gamma induced radiation) discrimination system is shown for a first order differencing predictor. Two consecutive frames, $S_t$ and $S_{t-1}$ are compared with saturation and lower thresholds both of which can be programmed. The criteria for nuclear event detection and saturation conditions are as follows:

1. The nuclear event erase will be issued if an isolated frame exceeds the saturation threshold (ST). That is, $S_{t-1} > S_t$ and $S_t$, $S_{t-2} < ST$.

2. A detector is saturated if the last two consecutive frames exceed ST. That is $S_t$ and $S_{t-1} < ST$.

The nuclear event erase pulse enables the switch at the output of the 164 K memory to ignore the memory output, which is a contaminated sample, and to replace it with the uncorrupted one directly from the converter. As shown in the figure, the numbers of single detector saturation, horizontally spreading saturation (3 adjacent detectors in a row) and vertically spreading saturation (3 adjacent detectors in a column) are counted and reported to the controller. In addition, the number of detector samples above the lower threshold and the peak value in a frame are reported for the purpose of optimal operating mode selection to maximize the signal to noise ratio. These

3-6
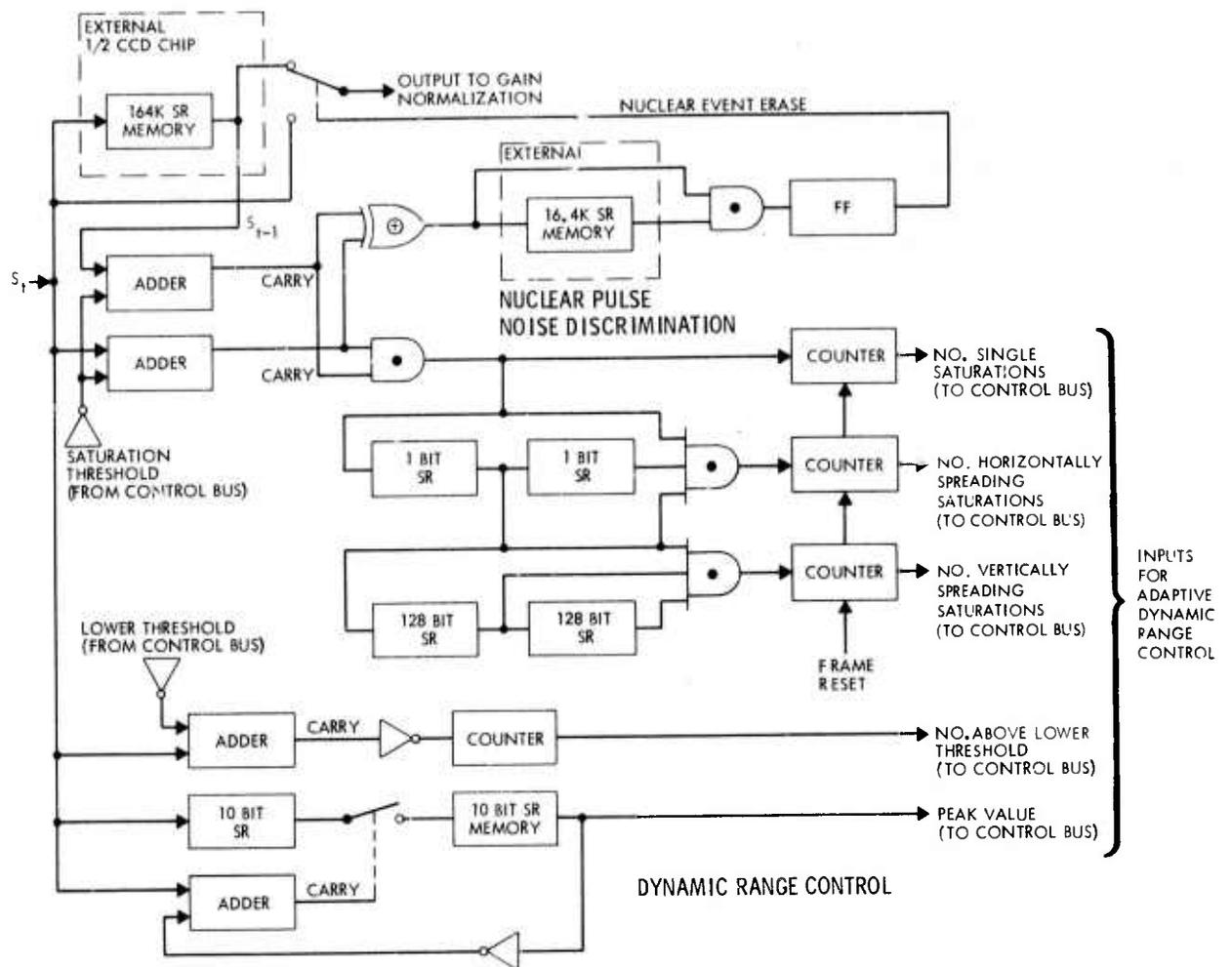
Figure 3.4-1. Gain control and nuclear event discrimination.

are accomplished by the adaptive dynamic range control algorithm pro grammed in the APSP controller.

It can be seen that the circuits involved here are standard digital logic. A 16.4 K shift register memory is needed to perform the discrimination logic; again a CCD register appears suitable. It should be noted that no additional memory is required since the same memory serves the predictor and this functional block.

## 3.5 RESPONSIVITY CALIBRATION/NORMALIZATION

As shown in Figure 3.5-1, the full amplitude (or difference) signal is normalized by multiplying it with the corresponding calibrated responsivity coefficient in the CCD memory, using a hard-wired serial/parallel 10 bit multiplier. The coefficients in the memory can be updated by invoking a calibration mode and correcting for the calibration source nonuniformities.
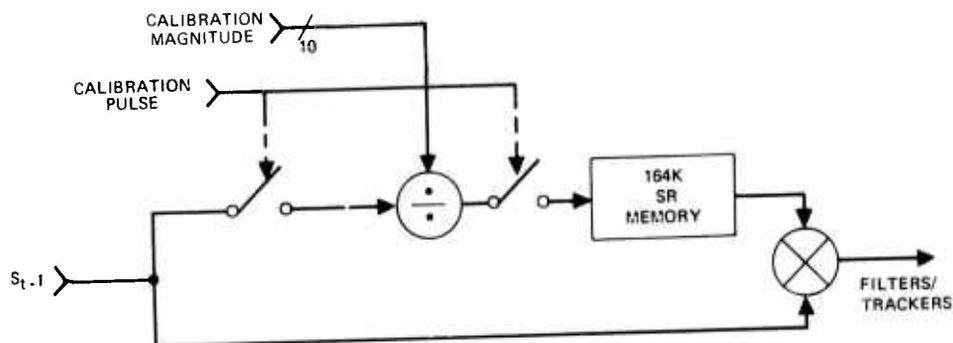


Figure 3.5-1. Responsivity calibration/normalization.

## 3.6 DIGITAL CONVERTER STATUS

The purpose of this discussion is to summarize the current state of the art in Analog to Digital and Digital to Analog converters. Units with special features are put into perspective by the use of a comparison table. The major emphasis is on total systems, therefore subsystem building blocks such as sample and hold circuits, precision ladders, etc. are not discussed. This survey of Analog-to-Digital Converters (A/D's) and Digital-to-Analog Converters (D/A's) includes data from "Technology Survey for Adaptive Programmable Signal Processor" CDRL item A007, previously submitted. However, the present survey covers about 2-1/2 times as many devices as were treated in the corresponding portion of A007.

Figure 3.6-1 is a graphical representation of A/D resolution vs speed. Superimposed over this data is an envelope of the data gathered by Lancaster[1]. The trends are similar but higher performance devices
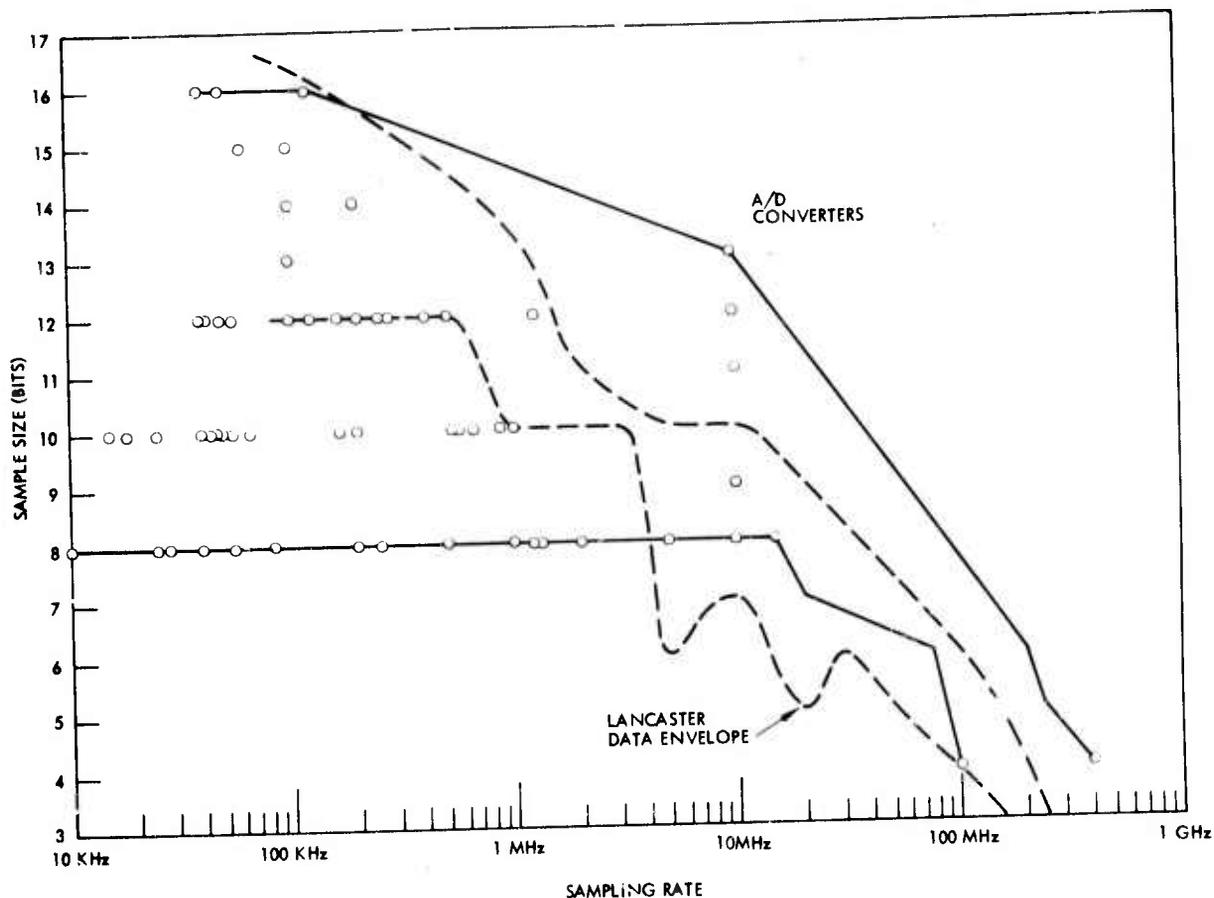
Figure 3.6-1. A/D converter resolution vs speed.

are presented in this (more recent) survey. Figure 3.6-2 depicts D/A's similarly, while Figures 3.6-3 and 3.6-4 show conversion energy vs resolution. Several interesting relationships can be highlighted by observing envelopes of Figures 3.6-1 through 3.6-4.

a. Sample size vs sample rate:

1. The trends for A/D's and D/A's are almost identical.

2. The envelopes are also very similar, which indicates that interfacing and timing considerations are unlikely to limit performance. Compatible speed and resolution requirements seem to be driving both designs.

b. Energy per conversion vs resolution:

1. The development trends are similar, but A/D's require 10 to 30 times more energy per conversion than D/A's for the same resolution.

2. The envelopes show about the same amount of spread in terms of variability of performance.

3-9
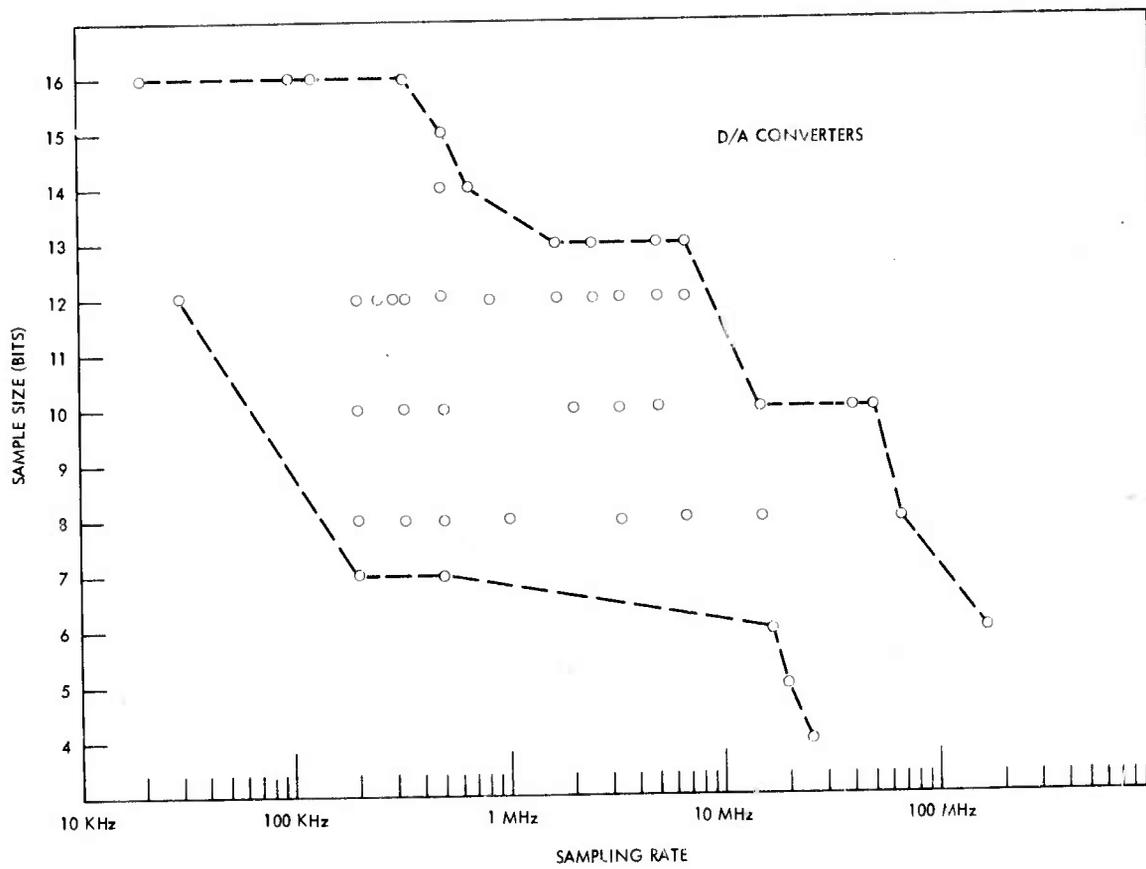
Figure 3.6-2.  D/A converter resolution vs speed.
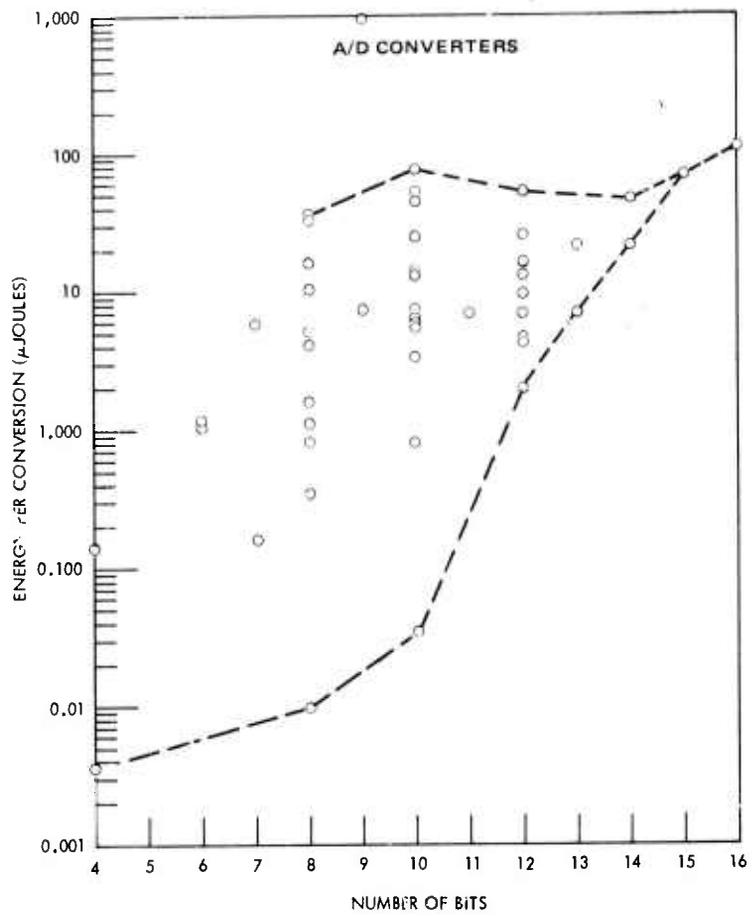
Figure 3.6-3. Conversion energy vs resolution
for A/D converters.
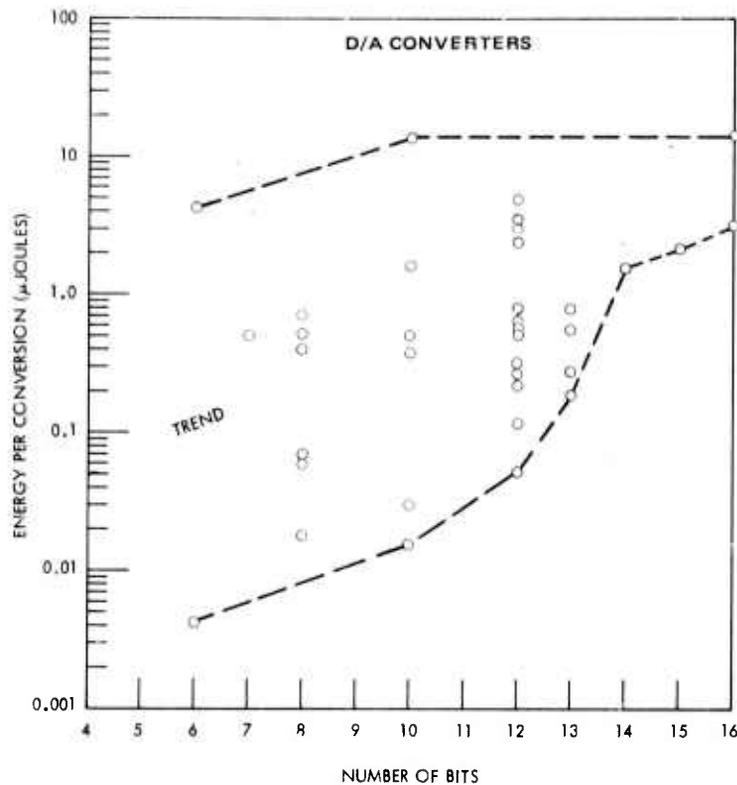
Figure 3.6-4. Conversion energy vs resolution
for D/A converters.

This information is useful in system configuration tradeoffs, and indicates
the benefits of designs utilizing a large number of low resolution devices
instead of one high resolution device; e. g. , One 11 bit A/D typically requires
as much energy as ten 6 bit machines. For encoder designs, concepts
favoring high resolution D/A's and low resolution A/D's would be preferred,
if high encoder resolution is required.

Tables 3.6-1 and 3.6-2 present only linear converters. Although
other types have significant advantages in different applications, they are
not appropriate for the ASE task. An example is the Precision Monolithics'
companding D/A that follows standard nonlinear speech compression laws.

One outstanding new linear device is the Hughes 4 bit monolithic A/D
encoder which has been operated with a 2.5 nsec converter time; by com-
bining four of these devices, a 6 bit word can be generated. This device
dissipates 1.4 watts. A second device is a 6 bit monolithic D/A converter,
which converts in 6 nsec and dissipates 0.7 watt.

3-12

TABLE 3.6-1. 1975 A/D CONVERTER CHARACTERISTICS

| Price | Company | Model | Bits | Time, μs | Sample Rate, $1/\tau$ | Power, w | μJoules | μJoules/Bit | Size |
|---|---|---|---|---|---|---|---|---|---|
| 795 | Analogic | MP 8016 | 16 | 25 | 40 kHz | 4.5 | 112.5 | 7.03 | 3 x 4.5 x 0.35 |
| | Intech | A 856-16 | 16 | 8 | 125 kHz | | | | |
| | Kriz (IEEE Symp Speech) | | 16 | 20 | 50 kHz | | | | |
| | Phoenix Data | ADC 1215 | 15 | 10 | 100 kHz | | | | |
| 695 | Analogic | MP 8015 | 15 | 16 | 62.5 | 4.5 | 72 | 4.8 | 3 x 4.6 x 0.35 |
| 4,170 | Preston Sci | GMAD214B | 14 | 5 | 200 kHz | | | | |
| 595 | Analogic | MP 8014 | 14 | 10 | 100 kHz | 4.5 | 45 | 3.21 | 3 x 4.6 x 0.35 |
| 695 | | MP 2914 | 14 | 10 | 100 kHz | 2.1 | 21 | 1.51 | 2 x 4 x 0.4 |
| 13,980 | Computer Labs | 9000 | 13 | 0.1 | 10 MHz* | 70 | 7 | 0.5385 | |
| | Analogic | MP 2913 | 13 | 10 | 100 kHz | 2.1 | 21 | 1.62 | 2 x 4 x 0.4 |
| 13,980 | | 9000 | 12 | 0.1 | 10 MHz* | 70 | 7 | 0.5833 | |
| 79 | Datel | HY 12BC | 12 | 8 | 125 kHz | 2 | 16 | 1.3333 | |
| 299 | | EH 12B3 | 12 | 2 | 500 kHz | 2.325 | 4.65 | 0.3875 | |
| | Teledyne | | 12 | | | | | | 0.181 x 0.114 |
| | | 4129 QZ | 12 | 24 | 41.7 kHz | | | | |
| | | 4132 | 12 | 3.5 | 285.7 kHz | | | | |
| | | 4133 | 12 | 2.5 | 400 kHz | | | | |
| | D. S. Schover | | 12 | 5 | 200 kHz | | | | |
| | Intersil | | 12 | | | | | | |
| | Intech | A851-12 | 12 | 2.5 | 400 kHz | | | | |
| | Hybrid Systems | | 12 | 20 | 50 kHz | 0.1 | 2 | 0.1667 | |
| | DMC | 2530 | 12 | 6 | 166.7 kHz | 2.6 | 15.6 | 1.3 | |
| | | 2724 | 12 | 18 | 55.6 kHz | 3 | 54 | 4.5 | |
| | Cycon | AD 12Z | 12 | 100 | 10 kHz | 0.25 | 25.5 | 2.12 | |
| | | AD 12QM | 12 | 25 | 40 kHz | | | | 2 x 4 x 0.4 |
| 229 | Analogic | MP 2712 | 12 | 4 | 250 kHz | 3.3 | 13.2 | 1.1 | 2 x 4 x 0.44 |
| 990 | ICL Data Device Corp | ADH 10/1 | 12 | 0.8 | 1.250 MHz | | | | |
| 225 | Burr-Brown | ADC 85 | 12 | 10 | 100 kHz | 0.45 | 4.5 | 0.375 | |
| 47.50 | | ADC 80 | 12 | 25 | 40 kHz | | | | |
| 395 | | ADC 60 | 12 | 3.5 | 285.7 kHz | 2.85 | 9.975 | 0.8312 | |
| 8,200 | Computer Labs | 9000 | 11 | 0.1 | 10 MHz* | 70 | 7 | 0.6364 | |
| 9,220 | | 7000 | 10 | 0.1 | 10 MHz | 75 | 7.5 | 0.75 | |
| | Teledyne | | 10 | | | | | | 0.181 x 0.114 |
| | | 4131 | 10 | 1 | 1 MHz | | | | |
| | McCreary & Gray | MOS | 10 | 20 | 50 kHz | | | | |
| | Intech | A851-10 | 10 | 1.5 | 666.7 kHz | | | | |
| | Hybrid Systems | ADC 580 LP | 10 | 20 | 50 kHz | | | | |
| | General Instr | MEM 5014 | 10 | | | | | | 0.062 x 0.104 |
| | DMC | 2520 | 10 | 5 | 200 kHz | 2.6 | 13 | 1.3 | |
| | | 2722 | 10 | 15 | 66.67 kHz | 3 | 45 | 4.5 | |
| | | 2726 | 10 | 18 | 55.56 kHz | 3 | 54 | 5.4 | |
| 49 | Datel | HY 10BC | 10 | 6 | 166 kHz | | | | |
| | Cycon | AD 10Z | 10 | 55 | 18.2 kHz | 0.25 | 13.75 | 1.37 | |
| | | AD 10GM | 10 | 22 | 45.4 kHz | | | | 2 x 4 x 0.4 |
| 185 | Burr-Brown | ADC 85 | 10 | 6 | 166 kHz | | | | |

(Table 3.6-1, continued)

| Price | Company | Model | Bits | Time, µs | Sample Rate, 1/τ | Power, w | µJoules | µJoules/Bit | Size |
|---|---|---|---|---|---|---|---|---|---|
| $ 45 | | ADA 80 | 10 | 21 | 47.6 kHz | | | | |
| 395 | | ADC 60 | 10 | 1.88 | 531.9 kHz | 2.85 | 5.36 | 0.536 | |
| 895 | Datel | H 10B | 10 | 1 | 1 MHz | 3.3 | 3.3 | 0.33 | |
| 349 | | 610B | 10 | | | 3.1 | . | | |
| 119 | | CM10B | 10 | 310 | 3.2 kHz | 90 | 27,900 | 2790 | |
| | Ayden Vector | ADH-10 | 10 | 25 | 40 kHz | 1.025 | 25.6 | 2.56 | |
| 484 | Analog Devices | 1103 | 10 | 1.2 | 833.3 kHz | 5.1 | 6.12 | 0.612 | |
| 69 | | 75706 | 10 | 1.7 | 588.2 kHz | 0.010 10 mw + comp | 0.034 | 0.0034 | |
| 299 | | 1123 | 10 | 65 | 15.4 kHz | 1.15 | 75 | 7.5 | |
| | | | 10 | 40 | 25 kHz | 0.020 | 0.8 | 0.008 | 0.120 x 0.135 |
| 7,630 | Computer Labs | CLB0910 | 9 | 0.1 | 10 MHz | 75 | 7.5 | 0.9375 | 0.181 x 0.114 |
| | Kindlmann (IEEE) | | 9 | 90 | 11.1 kHz | 11 | 990 | 124 | |
| 150 | Candy (IEEE Comm-22) | | 8 | 0.5 | 2 MHz | | | | |
| 473 | Analog Devices | 1103 | 8 | 1 | 1 MHz | 5.1 | 5.1 | 0.637 | |
| | | AD 7570J | 8 | 35 | 28.6 kHz | 0.01 | 0.35 | 0.044 | |
| | Burr-Brown | ADC 80 | 8 | 5 | 200 kHz | | | | |
| | | ADC 85 | 8 | 4 | 250 kHz | | | | |
| | Teledyne | | 8 | | | | | | |
| | | 4130 | 8 | 0.75 | 1.3 MHz | | | | |
| 189 | Micronetworks | 5060 | 8 | 100 | 10 kHz | 0.050 | 5 | 0.625 | 0.78 x 1 x 0.14 |
| | | 5065 | 8 | 100 | 10 kHz | 0.050 | 5 | 0.625 | |
| 295 | | | 8 | 1 | 1 MHz | 1.125 | 1.125 | 0.1406 | |
| 199 | Intech | A 857-8 | 8 | 0.8 | 1.25 MHz | | | | |
| | Harris | HI 0180/0185 | 8 | 25 | 40 kHz | 1.4 | 35 | 4.38 | 0.113 x 0.124 |
| | DMC | 2510 | 8 | 4 | 250 kHz | 2.6 | 10.4 | 1.3 | |
| | | 2720 | 8 | 12 | 83.3 kHz | 3 | 36 | 4.5 | |
| 895 | Datel | VH8B | 8 | 0.2 | 5 MHz | 8.3 | 1.66 | 0.207 | |
| 995 | | UH8B | 8 | 0.1 | 10 MHz | 8.3 | 0.83 | 0.103 | |
| 49 | | HY8BC | 8 | 4 | 250 kHz | | | | |
| | | CM8B | 8 | 50 | 4 kHz | 90 | 22,500 | 2812.5 | |
| | Cymos | AD8Z | 8 | 40 | 25 kHz | | | | |
| | | AD8QM | 8 | 18 | 55.6 kHz | | | | 2 x 4 x 0.4 |
| 12,500 | Computer Labs | VHS 815 | 8 | 0.0666 | 15 M | 50 | 16.66 | 208 | |
| | Analogic | MP 2908 | 8 | 2 | 500 kHz | 2.1 | 4.2 | 0.525 | 2 x 4 x 104 |
| | Giri & Maxwell 1973 Intl T.M. Conf. | | 7 | 0.05 | 20 MHz | 3 | 0.15 | 0.021 | |
| 12,080 | Computer Labs | VHS 720 | 7 | 0.05 | 20 MHz | 119 | 5.95 | 0.85 | |
| 14,000 | | VHS 675 | 6 | 0.0133 | 75 MHz | 91 | 1.213 | 0.202 | |
| | Hughes | | 6 | 0.005 | 200 MHz* | 205 | 1.025 | 0.171 | |
| | R.E. Fisher (IEEE MTT16 #8) | | 5 | 0.00416 | 240 MHz* | | | | |
| | R.W. Means (A/D Conv by CTD) Navy Case #56,171 | | 5 | | | | | | |
| | Hughes | | 4 | 0.0025 | 400 MHz* | 1.4 | 0.0035 | 0.00875 | 0.121 x 0.164 |
| 2,120 | Computer Labs | 4100 | 4 | 0.01 | 100 MHz | 13.76 | 0.1376 | 0.0344 | |

(Table 3.6-1, continued)

| Price | Company | Model | Bits | Time, μs | Sample Rate, 1/τ | Power, w | μJoules | μJoules/Bit | Size |
|---|---|---|---|---|---|---|---|---|---|
| $ 45 | | ADA 80 | 10 | 21 | 47.6 kHz | | | | |
| 395 | | ADC 60 | 10 | 1.88 | 531.9 kHz | 2.85 | 5.36 | 0.536 | |
| 895 | Datel | H 10B | 10 | 1 | 1 MHz | 3.3 | 3.3 | 0.33 | |
| 349 | | 610B | 10 | | | 3.1 | . | | |
| 149 | | CM10B | 10 | 310 | 3.2 kHz | 90 | 27,900 | 2790 | |
| | Ayden Vector | ADH-10 | 10 | 25 | 40 kHz | 1.025 | 25.6 | 2.56 | |
| 484 | Analog Devices | .103 | 10 | 1.2 | 833.3 kHz | 5.1 | 6.12 | 0.612 | |
| 69 | | 75706 | 10 | 1.7 | 588.2 kHz | 0.010 10 mw + comp | 0.034 | 0.0034 | |
| 299 | | 1123 | 10 | 65 | 15.4 kHz | 1.15 | 75 | 7.5 | 0.120 x 0.135 |
| | | | 10 | 40 | 25 kHz | 0.020 | 0.8 | 0.008 | 0.181 x 0.114 |
| 7,630 | Computer Labs | CLB0910 | 9 | 0.1 | 10 MHz | 75 | 7.5 | 0.9375 | |
| | Kindlmann (IEEE) | | 9 | 90 | 11.1 kHz | 11 | 990 | 124 | |
| 150 | Candy (IEEE Comm-22) | | 8 | 0.5 | 2 MHz | | | | |
| 473 | Analog Devices | 1103 | 8 | 1 | 1 MHz | 5.1 | 5.1 | 0.637 | |
| | | AD 7570J | 8 | 35 | 28.6 kHz | 0.01 | 0.35 | 0.044 | |
| | Burr-Brown | ADC 80 | 8 | 5 | 200 kHz | | | | |
| | | ADC 85 | 8 | 4 | 250 kHz | | | | |
| | Teledyne | | 8 | | | | | | |
| | | 4130 | 8 | 0.75 | 1.3 MHz | | | | |
| 189 | Micronetworks | 5060 | 8 | 100 | 10 kHz | 0.050 | 5 | 0.625 | 0.78 x 1 x 0.14 |
| | | 5065 | 8 | 100 | 10 kHz | 0.050 | 5 | 0.625 | |
| 295 | | | 8 | 1 | 1 MHz | 1.125 | 1.125 | 0.1406 | |
| 199 | Intech | A 857-8 | 8 | 0.8 | 1.25 MHz | | | | 0.113 x 0.124 |
| | Harris | HI 0180/0185 | 8 | 25 | 40 kHz | 1.4 | 35 | 4.38 | |
| | DMC | 2510 | 8 | 4 | 250 kHz | 2.6 | 10.4 | 1.3 | |
| | | 2720 | 8 | 12 | 83.3 kHz | 3 | 36 | 4.5 | |
| 895 | Datel | VH8B | 8 | 0.2 | 5 MHz | 8.3 | 1.66 | 0.207 | |
| 395 | | UH8B | 8 | 0.1 | 10 MHz | 8.3 | 0.83 | 0.103 | |
| 49 | | HY8BC | 8 | 4 | 250 kHz | | | | |
| | | CM8B | 8 | 250 | 4 kHz | 90 | 22,500 | 2812.5 | |
| | Cymos | AD8Z | 8 | 40 | 25 kHz | | | | 2 x 4 x 0.4 |
| | | AD8QM | 8 | 18 | 55.6 kHz | | | | |
| 12,500 | Computer Labs | VHS 815 | 8 | 0.0666 | 15 M | 250 | 16.66 | 208 | 2 x 4 x 104 |
| | Analogic | MP 2908 | 8 | 2 | 500 ... | 2.1 | 4.2 | 0.525 | |
| | Giri & Maxwell 1973 Intl T.M. Conf. | | 7 | 0.05 | 20 MHz | 3 | 0.15 | 0.021 | |
| 12,080 | Computer Labs | VHS 720 | 7 | 0.05 | 20 MHz | 119 | 5.95 | 0.85 | |
| 14,000 | | VHS 675 | 6 | 0.0133 | 75 MHz | 91 | 1.213 | 0.202 | |
| | Hughes | | 6 | 0.005 | 200 MHz* | 205 | 1.025 | 0.171 | |
| | R.E. Fisher (IEEE MTT16 #8) | | 5 | 0.00416 | 240 MHz* | | | | |
| | R.W. Means (A/D Conv by CTD) | | 5 | | | | | | |
| | Navy Case #56,171 | | | | | | | | |
| | Hughes | | 4 | 0.0025 | 400 MHz* | 1.4 | 0.0035 | 0.00875 | 0.121 x 0.164 |
| 2,120 | Computer Labs | 4100 | 4 | 0.01 | 100 MHz | 13.76 | 0.1376 | 0.0344 | |

# TABLE 3.6-2.  1975 D/A CONVERTER CHARACTERISTICS

| Price | Company | Model | Bits | Time, μs | Sample Rate, 1/τ | Power, w | μJoules | μJoules/Bit | Size |
|---|---|---|---|---|---|---|---|---|---|
| $ 1,500 | Analogic | MP 8116 | 16 | 10 | 100 kHz | 1.4 | 14 | 0.088 | |
| | Preston Sci | GMDAC HV4Q-15B | 15 | 2 | 500 kHz | | | | |
| | Cycon | CY2336 | 14 | 2 | 500 kHz | | | | |
| 140 | Dynamic Meas Corp | 2103 | 13 | 0.6 | 1.67 MHz | 1.35 | 0.81 | | 2.5 x 3 x 0.5 |
| 170 | | 2060 | 13 | 0.4 | 2.5 MHz | 1.425 | 0.57 | | 2.5 x 3 x 0.5 |
| 175 | | 2041 | 13 | 0.2 | 5 MHz | 1.425 | 0.285 | | 2.5 x 3 x 0.5 |
| 495 | | 153-075-030 | 13 | 0.15 | 6.67 MHz | 1.25 | 0.188 | | 2.5 x 3 x 0.5 |
| | National Semiconductor | DA 1200 | 12 | 2 | 500 kHz | 2.4 | 4.8 | 0.4 | |
| | Cycon | CY2 6 | 12 | 2 | 500 kHz | | | | |
| | | CYDAC 6912B1 | 12 | 0.3 | 3.33 MHz | | | | |
| | | 12BL | 12 | 5 | 200 kHz | 0.103 | 0.515 | 0.043 | |
| 67 | Dynamic Meas Corp | 2202 | 12 | 0.3 | 3.33 MHz | 0.175 | 0.0525 | | 2 x 2 x 0.4 |
| 115 | | 2102 | 12 | 0.6 | 1.67 MHz | 1.35 | 0.81 | | 2.5 x 3 x 0.5 |
| 155 | | 2030 | 12 | 0.4 | 2.5 MHz | 1.425 | 0.57 | | 2.5 x 3 x 0.5 |
| 160 | | 2031 | 12 | 0.2 | 5 MHz | 1.425 | 0.285 | | 2.5 x 3 x 0.5 |
| 29 | Datel | DAC HY12 | 12 | 0.3 | 3.33 MHz | | | | |
| 37.50 | Burr-Brown | DAC 80 | 12 | 0.3 | 3.33 MHz | | | | |
| | Analogic | MP 2712 | 12 | 4 | 250 kHz | | | | |
| | DMC | 2400 | 10 | 0.025 | 40 MHz | | | | |
| | Computer Labs | MDS 1020 | 10 | 0.020 | 50 MHz | 1.48 | 0.03 | 0.003 | |
| 11 | Analog Devices | AD 7520 | 10 | 0.5 | 2 MHz | 0.03 | 0.015 | 0.0015 | |
| | | AD 7522 | 10 | | | | | | |
| | Cycon | CY2136 | 10 | 2 | 500 kHz | | | | |
| | | CYDAC 4910B1 | 10 | 0.3 | 3.33 MHz | | | | |
| | | 10BL | 10 | 5 | 200 kHz | 0.103 | 0.515 | 0.0515 | |
| 145 | DMC | 2021 | 10 | 0.2 | 5 MHz | 1.425 | 0.285 | | 2.5 x 3 x 0.5 |
| 64 | | 2200 | 8 | 0.3 | 3.33 MHz | 0.175 | 0.0525 | | 2 x 2 x 0.4 |
| | Cycon | CY2036 | 8 | 2 | 500 kHz | | | | |
| | | CY2018 | 8 | 3 | 333 kHz | | | | |
| | | 8BL | 8 | 5 | 200 kHz | 0.103 | 0.515 | 0.064 | |
| | Analog Devices | AD559KD | 8 | 0.3 | 3.33 MHz | 0.232 | 0.0696 | | |
| | Computer Labs | MDS 815 | 8 | 0.015 | 66.67 MHz | 1.26 | 0.019 | 0.0024 | 0.3 x 2.3 x 0.43 |
| | | TVDA 0815 | 8 | 0.066 | 15 MHz | 10.75 | 0.72 | 0.09 | |
| | Cycon | CY2536 | 7 | 2 | 500 kHz | | | | |
| | | 2DL | 7 | 5 | 200 kHz | 0.103 | 0.515 | 0.073 | |
| | | HS 2615 | 6 | 0.060 | 16.67 MHz | | | | |
| | | HS 2520 | 5 | 0.050 | 20 MHz | | | | |
| | | HS 2425 | 4 | 0.040 | 25 MHz | | | | |
| Hughes | | | 6 | 0.0045 | 225 MHz | 1.0 | .0045 | .00075 | .080 x .140 |

Recently several new types of devices have appeared in the literature. The more interesting of these are:

- An 8 bit, 1 GHz A/D converter based on linear electrooptic phase retardation (6) utilizing optical waveguide modulators is described. No actual power or size parameters are reported.

- A 5 bit MOS monolithic clockless A/D converter (7) utilizing portions of a continuously variable threshold device to achieve conversion times of 2 μsec; power dissipation was not reported.

- An all MOS successive approximation weighted capacitor A/D conversion technique (reference 8). It performs a 10 bit conversion in 20 μsec. The acquisition time is 25 μsec; thus the conversion rate is 22 KHz.

- At the same conference (8) R. B. Craven presented a bipolar LSI, 12 bit D/A converter consisting of a 97 x 180 mil Si-Cr resistor network, and a 79 x 179 chip of active circuitry; power dissipation was not reported.

## REFERENCES

1. J. F. Lancaster, D. W. Burlage, R. H. Fletcher Jr., "Analog-to-Digital Converter Technology Preliminary Investigation and Proposal, DA Proj. 1M2-62303-A-214 AD 768771.

2. Kriz, IEEE Symposium on Speech Recognition, April 1974.

3. Candy, IEEE Comm -22 No. 3, March 1974.

4. R. E. Fisher, IEEE MTT16 No. 8, August 1968.

5. Kindlmann, IEEE IM-23 No. 2, June 1974.

6. H. F. Taylor, IEEE Proceedings, October 1975.

7. Yamaguchi & Sato, IEEE Trans. on Electron Devices, May 1975, page 295.

8. Baldwin, 1975 IEEE International Solid-State Circuits Conference.

## 4.0 MEMORY DESIGN

The general microcomputer system shown in Figure 4.0-1 contains three types of memory; a program storage memory (PROM), a read/write random access memory (RAM), and an information storage memory. Improvements in semiconductor technology will enable fabrication of smaller geometry devices, resulting in higher packing densities, an increased number of functions placed on a single chip, and higher operating frequencies. Improvements in microprocessors must be accompanied by corresponding improvements in memory technology.

The information memories (pixel data, star data, etc.) consist of large data banks, and therefore benefit from a serial (block addressed) organization. Because of the large quantity of data in these memories, primary consideration must be given to power dissipation and packing
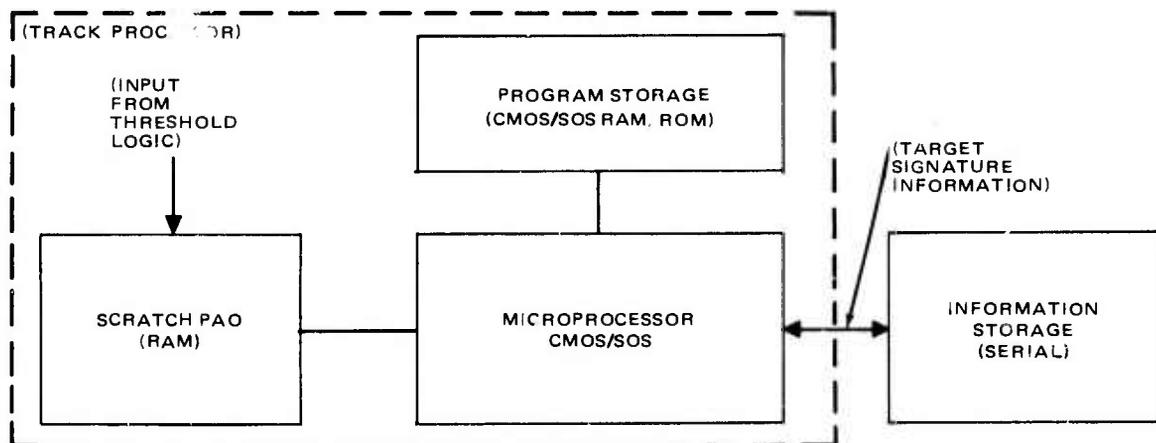


Figure 4.0-1. General microcomputer system.
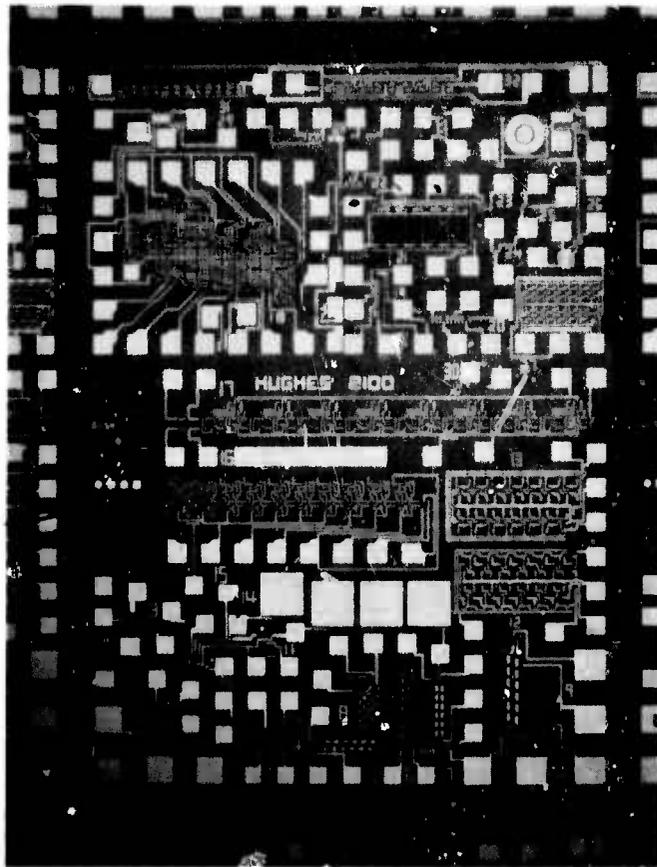
1  Transistor
2  Transistor isoplanar II technology
3  Transistor substrate fed logic
4  Resistors
5  15 stage R.O. substrate fed logic
6  R.O. #6
7  2, 7 stage inverters
8  R.O. 15 stages using isoplanar II technology
9  R.O. 9 and 9B (different isolation from #6)
10  Isolation transistor
11  Isolation transistor
12  ECL 15 stage R.O.
13  EFL 15 stage R.O.
14  Capacitors
15  10 stage freq divider
16  Double metal test device
17  8 stage shift register

18  4 bit adder
19  10 collector transistor
20  10 collector transistor
21  11 stage R.O. with large collector
22  Lateral PNP
23  Lateral PNP
24  Single transistor
25  Single transistor
26  MOS transistor
27  15 stage ECL R.O. isoplanar II technology
28  3 transistors with different isolation
29  Alignment mark
30  Resistor
31  R.O.
32  2 bit shift register
    (R.O. = Ring Oscillator)

Figure 4.1-1.  Micro-photo of Hughes 2100 $I^2L$ chip.

been initiated to establish processing that will yield 4 μm /bit shift registers with minimum lateral dimensions of 0.5 μm. To overcome the problem of aluminum grain size, aluminum gates will be replaced by polysilicon. Although it requires an extra processing step for second level gates, polysilicon enables the construction of much smaller devices. E-beam technology is capable of decreasing the minimum lateral dimension by a factor of 2 or 3 before significant small-geometry problems arise. The concern is that small geometry reduces the bucket size and therefore the amount of stored charge. The limited amount of stored charge introduces threshold voltage variation as a significant problem. This problem is most apparent in relation to refresh amplifiers which then need much more stringent threshold variation tolerances. Another problem of small geometry is that decreases in lateral dimensions also require decreases (and therefore require high control) of vertical dimensions (oxide thickness, diffusions, etc.).

Continual improvements in processing techniques indicate overall device yield will increase to more than 10 percent. Predictions, based on SPS design using Electron Beam lithography and optimized processing, are that a 320K bit serial memory capable of operating in the 20-50 MHz range is quite feasible.

A dual 160K bit serial memory organization (two blocks) is shown in Figure 4-1.2 and provides for two words of storage for all pixels in one MFPA chip (i.e., this is a block addressed memory where one block is a full MFPA word). Chip dimensions, including peripheral circuitry, are approximately 100 x 100 mils and contain ten 32K bit arrays in two SPS blocks. There is one refresher per 32K bit array. Two transfers are needed per bit, therefore the charge transfer efficiency (CTE) necessary to retain 70 percent of the stored information before refresh is given by

$$CTE^{2(M+N)} = 0.7$$

$$M = \text{Number of rows}$$

$$N = \text{Number of columns}$$

Using a 32K bit/refresh design requires a CTE of only 0.9995.

4-3
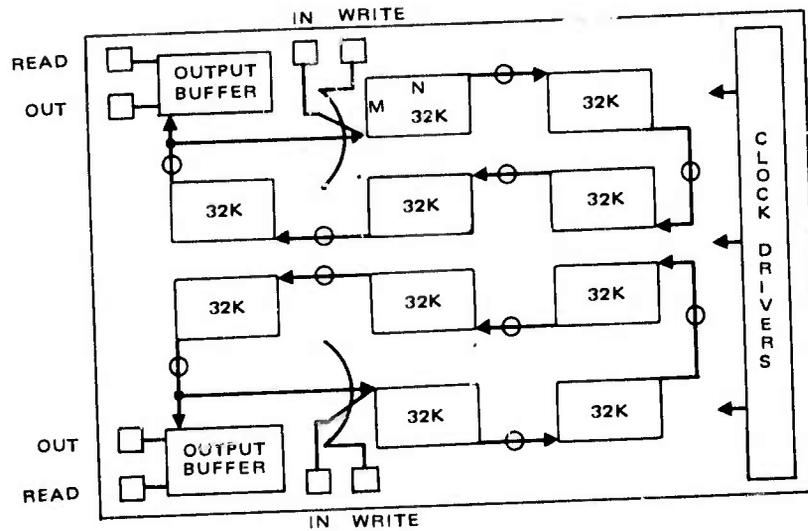
Figure 4.1-2. 320K CCD serial memory
(dual 160K blocks).

Refresher power is given by:

$$P_{Ref} = C_{Ref} V^2 f N_A$$

$C_{Ref}$ = total capacitance of refresh circuitry, clock leads
and pad capacitance

$V$ = PP clock voltage

$f$ = clock frequency

$N_A$ = number of arrays (10)

Assuming four volt clocks operating at a frequency of 2 MHz,

$$P_{Ref} \simeq 1.9 \text{ mwatt}$$

Transfer power dissipation of the CCD memory is given by

$$P_{tr} = CV^2 f(2N + M) N_A$$

Assuming four volt clocks operating at a frequency of 2 MHz (which is con-
servative if advancements in threshold voltage control continue),

$$P_{tr} = 5.6 \text{ mwatts}$$

Based upon an output buffer power requirement of two mwatts and an on chip clock driver of overall 50 percent efficiency (which again is conservative if CCD compatible CMOS is developed).

$$P_{Total} = 17 \text{ mwatts}$$

The _effective_ power delay product of the design is then:

$$E = \frac{17 \text{ mwatts}}{(320K \text{ bits} \times 2 \text{ MHz})} = 0.03 \text{ p-J/bit}$$

For this application of dedicated, serial memory where there is no need for random access and speed is not a critical issue, power dissipation drives the design. Line addressable techniques require additional logic and line drive capability on the common output diffusion. However, a block addressed SPS configuration may provide the optimum trade between power and storage time.

Examples of I/O and refresher circuitry are shown in Figures 4.1-3 and 4.1-4. The refresher circuit is a floating diffusion controlling an input threshold gate. A logic 1 will bias $V_T$ sufficiently to allow the input potential well to refill while a logic 0 will prevent transfer of charge.

Various storage techniques have been devised to provide the refreshing requirement when data is not requested, although for MFPA pixel data the memories will be in continuous use. For star catalog data a standby storage mode will be beneficial to reduce power, assuming a typical storage time of less than 0.3 seconds. The memory can simply be recycled at a reduced clock rate to ensure regeneration within the CCD storage time capability. Regeneration occurs in each of five arrays so that the 1.64 MHz clock (10 Hz MFPA clock) rate can be reduced by a factor of eight without additional chip complication, resulting in a standby power dissipation of about 1.7 mwatts per chip at a frequency of 200 KHz.

The read/write random access memory (RAM) is used as a scratch pad. The main design considerations here are access time (speed) and power dissipation. The two potential candidates for a high speed, high density, low power RAM are CMOS/SOS and the recently introduced application of CCDs in RAMs.

4-5
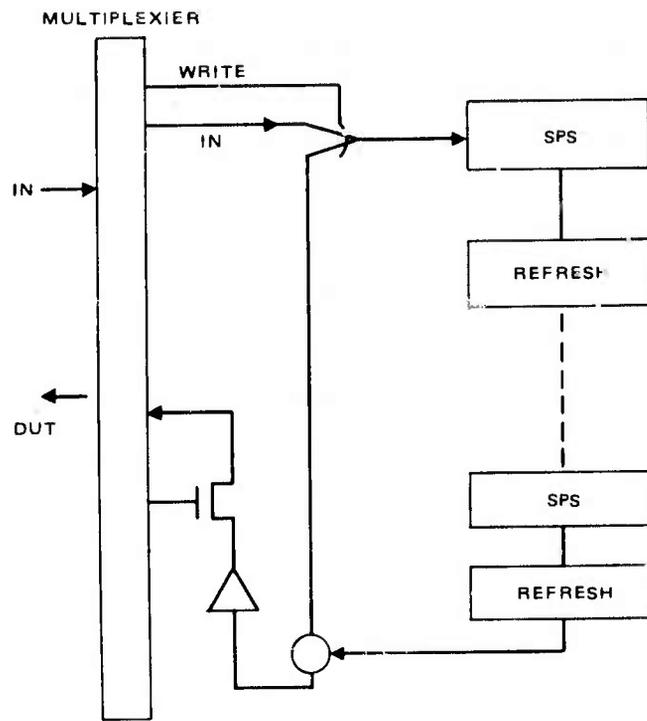
Figure 4.1-3.  I/O circuitry of CCD
serial memory.
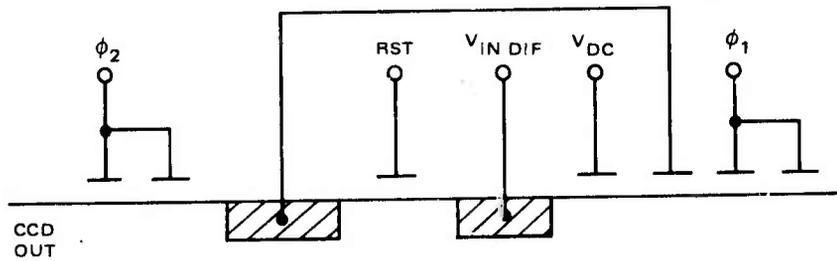


Figure 4.1-4.  CCD refresh circuit (floating diffusion).

4-6

The CCD RAM uses the basic CCD concept of minority carrier storage and transfer between potential wells to provide a nondestructive readout. Access and cycle times are comparable to present MOS memories. The RAM consists of a monolithic array of the CCD unit cells shown in Figure 4.1-5, connected in an N x N array as shown in Figure 4.1-6.
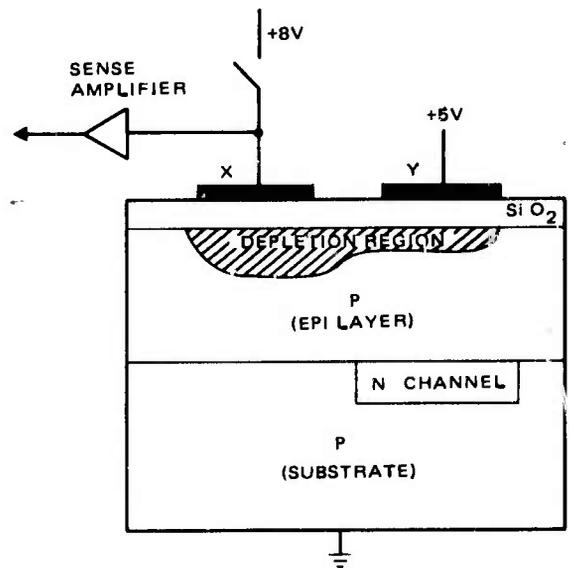


Figure 4.1-5. CCD ram unit cell.
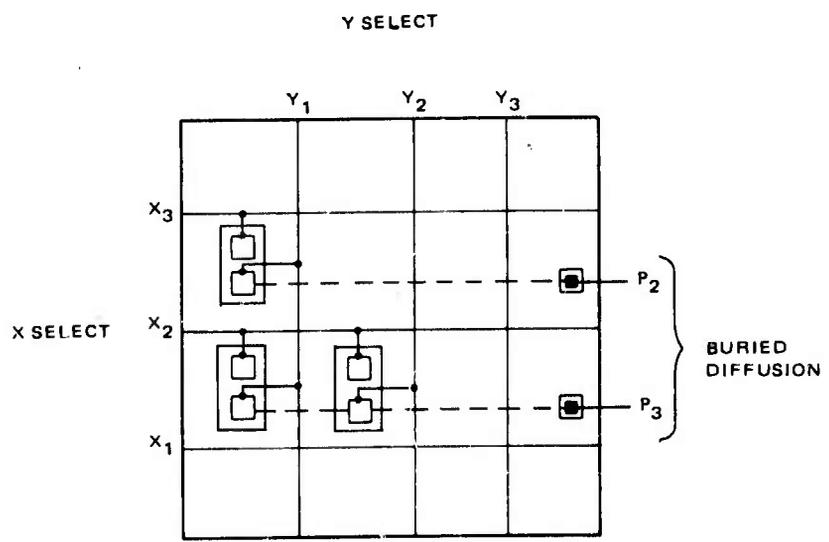


Figure 4.1-6. CCD ram array.

4-7

Each cell has two electrodes, X and Y. A logic 1 is represented by a high density of minority carriers and a logic 0 by a low density of minority carriers in the well. The potential wells are produced by applying a voltage of 8V to all the X (row) lines and 5V to all the Y (column) lines. Thus, when a logic 1 is being stored, most of the minority carriers are beneath the X electrode (n channel device).

To read data stored in the CCD cell, an extra positive pulse is applied to the Y electrode (while disconnecting all X electrode voltage sources), thereby inducing a pulse that drives the X electrode more positive. The induced pulse will be large if the stored bit is a 1, and small if the stored bit is a 0.

To write a logic 1, the N diffusion is momentarily biased, injecting minority carriers into the depletion region. To enter a logic 0, both electrodes are grounded allowing the minority carriers to recombine.

The CCD RAM requires a sense amplifier or comparator for each X select line (row). Also, a periodic refresh is necessary to retain stored information. Although the signal to noise ratio of the CCD RAM is adequate for the APSP application, because of the sense amplifier's power consumption and the necessity of refresher circuitry, a more likely candidate for a high speed, high density, low power RAM is CMOS/SOS.

## 4.2 CMOS RANDOM ACCESS MEMORY

The CMOS/SOS RAM is a static memory consuming power only during the switching of CMOS gates and during the write transient. Stored information is retained indefinitely without the need for refresher circuitry. The introduction of E-beam technology and improvements in device yield will greatly increase the capability of CMOS devices (see Section 5). Applications of E-beam technology should result in resolution capabilities of 0.5 μm and overall device yield will probably reach 10 to 20 percent. CMOS LSI devices are expected to operate above the 100 MHz range with gate delay times of 0.6 nsec or less. Work is currently in progress at the Hughes Newport Beach facility on 4K RAMS. Lithography techniques enabling construction of 64K RAMS probably will be available by the early 1980's. Access times should be 1-3 times the microcomputer minor cycle time.

Present minimum access times for a CMOS RAM are approximately 3-5 times the minimum clock period. Assuming a 0.6 nsec/gate delay and 3 nsec minimum clock period, minimum access times will be approximately 9 to 15 nsec. Write times will be approximately 20 nsec.

The program storage memory will be of the same technology as the RAM and microprocessor (CMOS/SOS) and will therefore have cycle times comparable to those of the microprocessor.

Figure 4.2-1 shows a block diagram of a 64K CMOS/SOS RAM memory. The memory is organized in a 256 by 256 array with a 16 bit
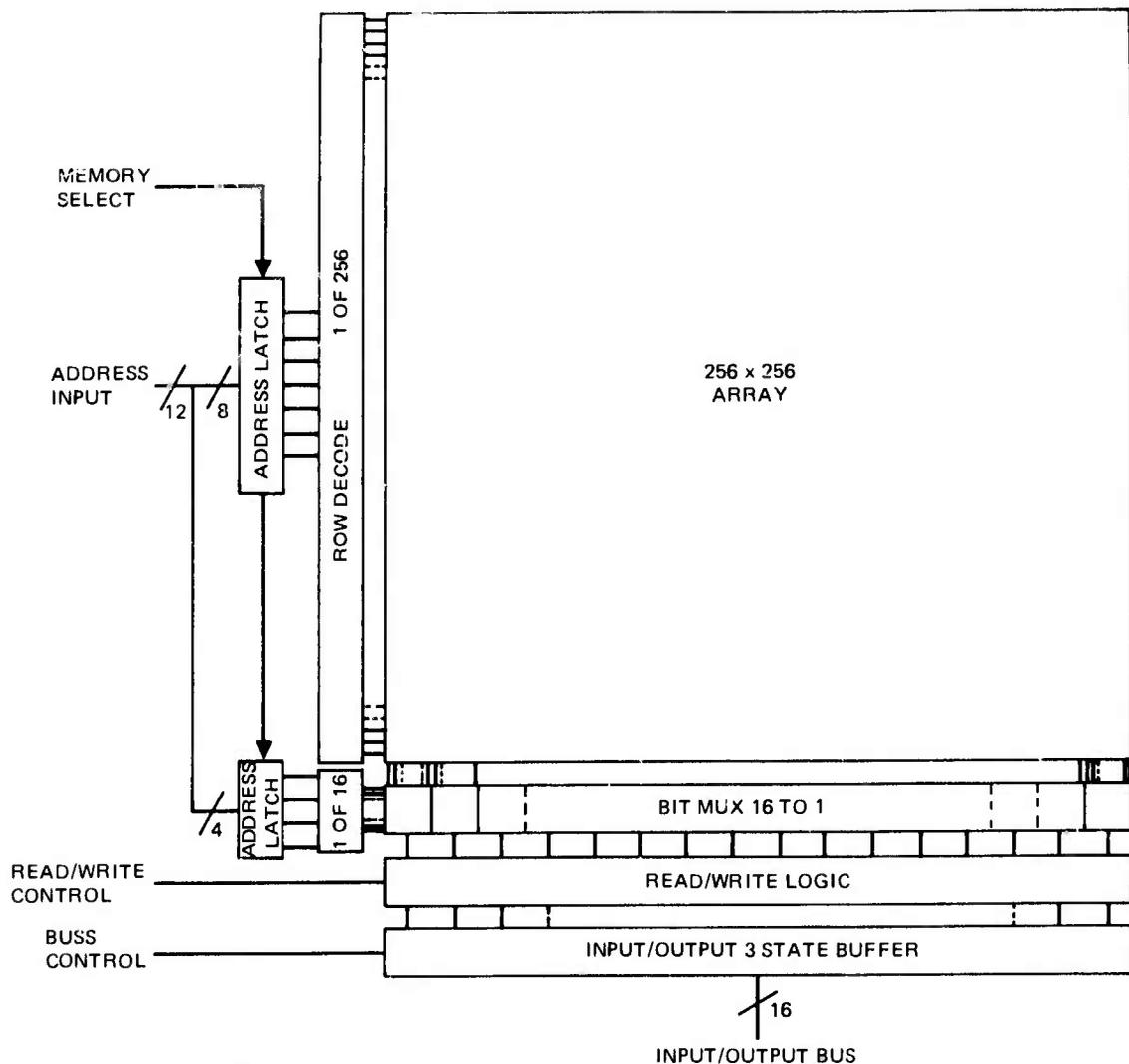


Figure 4.2-1. Block diagram of 64K CMOS RAM memory.

input/output format. The basic memory cell consists of five CMOS transistors, as shown in Figure 4.2-2. The cell is accessed by activating the word select line and sensing the logic state on the bit line in the read mode or by driving the bit line to the desired logic state in the write mode. The 256 bit lines are multiplexed in groups of 16 to provide the 16 bit I/O format. The input/outputs are 3-state bus lines connected to 3 other devices. The necessary read/write logic and timing will be generated on chip to minimize interconnects.

E-beam technology should make it possible to construct the 64K memory on a 200 mil$^2$ chip with good yield. Assuming a 20% access duty cycle and the microcomputer operating at 40 MHz basic clock rate, the memory is expected to consume approximately 15 mw with a standby power of less than 0.3 mw.
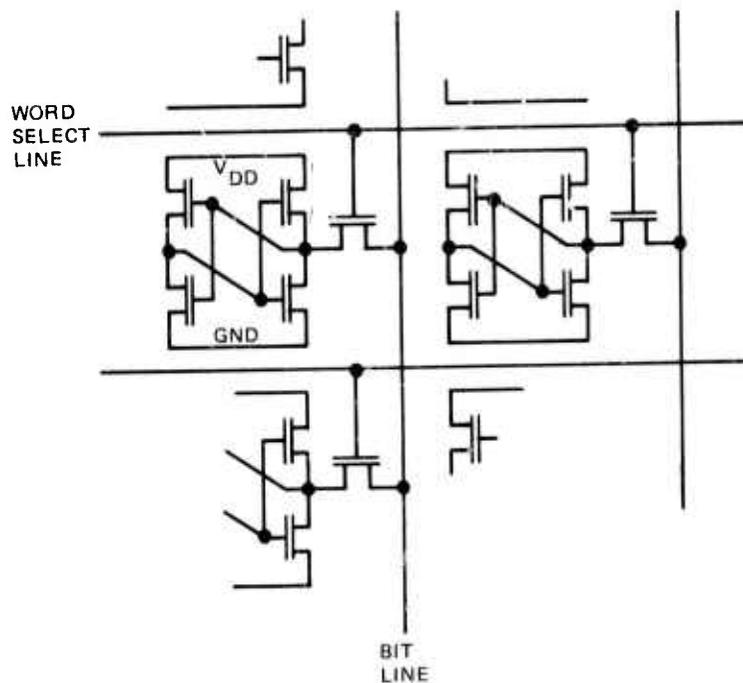


Figure 4.2-2. Basic RAM memory cell configuration.

## 4.3 SUMMARY OF CRITICAL MEMORY DEVICE DESIGNS

In summary, the memory technology likely to be available in 1982 can be expected to meet the needs of the APSP. The critical technology development requirements to realize this capability are:

- Lithography capable of about 0.2 μm line widths over a 100 mil square CCD chip (0.5 μm line widths over a 200 mil square CMOS chip)
- On chip CMOS CCD clock driver development
- Threshold voltage uniformity improvement
- Advances in LSI on chip interconnect technology

The critical memory device characteristics anticipated are:

Block Addressed (Serial Access)

- Technology - monolithic CCDs (n-surface channel) with compatible CMOS clock drivers on chip.
- Dual 160K bit chip (320K bits/chip) for two MFPA-words
- Maximum clock rates >20 MHz
- Chip dissipation - 14 mwatts at 1.64 MHz clock (10 Hz MFPA frame rate), reduced to about 4 mwatts if 2 volt clocks are feasible
- Standby dissipation - 1.7 mwatts at 200 KHz clock

RAM

- Technology - CMOS/SOS
- 64K bit chip
- Chip size - 200 x 200 mils
- Maximum clock rates >100 MHz
- Access time ~12 nsec, write time ~20 nsec
- Standby power dissipation < 0.3 mwatt
- Active power dissipation < 15 mwatts while operating with a 10 pf bus with the 8 MHz read/write rate.

## 5.0  LOGIC DEVICE DESIGN (CMOS/SOS)


Advancements in integrated circuit processing technology have been occurring at a rapid rate over the last few years, and it appears that integrated circuit performance will continue to improve in the years ahead. As these improvements are realized, device speeds will increase, speed-power products will decrease, and the number of devices per given area of chip real estate will increase. This will enable the construction of LSI circuits much more powerful than those in use today. Micro processors operating at clock frequencies in excess of 100 MHz with a speed-power product of less than 0.1 p-j/gate, and a gate density of more than 1000 gates/mm$^2$ should be within the realm of usable technology within six years.

Although there are technologies available today for 200 MHz logic, none operate close to the speed-power product requirements of the APSP. What is required is a technology that consumes little power and has the potential to reach the speeds needed. The most promising is CMOS, because it consumes power only while switching, and has inherently low power consumption, while still allowing high speed operation. As processing technology improves, the devices can be made smaller, which will decrease charge transit time and gate capacitance, with a corresponding increase in device speed. The lower capacitances will mean smaller device currents. This, along with decreased operating voltage, will greatly reduce the power required for CMOS operation. CMOS performance can also be improved by using an insulating substrate; parasitic capacitance is reduced — causing a considerable increase in device speed, and device density is increased since isolation diffusions are not required. CMOS silicon on

5-1

sapphire (SOS) is presently being developed by several manufacturers. This technology has a disadvantage in that it is difficult to grow silicon on sapphire. The silicon tends to have defects in the crystal causing unusually high leakage currents. Improvements in growing silicon have minimized this problem, and further improvements can be expected. Recent work at the Hughes Research Laboratories in Malibu has ... with the possibility of using high resistivity silicon as an insulator. The growing of low resistivity silicon on this substrate would eliminate the crystal defect problem (but creates a substrate leakage problem). The probability of this technology becoming usable in the next few years is high, and would prompt further improvements in CMOS performance.

These technological advancements will enable CMOS to operate with delay times less than 0.4 ns, compared to present CMOS LSI delay times of approximately 5 ns, and less than 0.02 p-J power-delay product compared to present LSI power delay products greater than 1 p-J (with fan outs of 4 or 5). The gate density will be greater than 1000 gates/mm$^2$ for LSI random logic compared to present day LSI gate densities of approximately 90 gates/mm$^2$. These advanced CMOS gates will be used to construct memory cells for Read only Memories (ROM) and Random Access Memories (RAM) as well as LSI for micro processors for high speed, low power, and large data handling capability systems.

Since the key to producing CMOS that meets the criteria stated earlier is reducing device size, an analysis of the problems involved follows. The potential problems are:

1. Approximations used in conventional device modeling do not hold for small geometry devices.

2. Present photographic techniques of mask making and wafer exposing are limited by the wavelength of light.

3. Voltages that can be applied from drain to source are limited due to the reverse bias diode punch through voltage.

4. Threshold voltage becomes harder to predict and control, and more significant as devices and applied voltages get smaller.

These problems are examined individually in the following discussion.

Swanson (5-2) has done an analysis on small geometry MOS devices. In conventional device modeling, it is assumed that there is one dimensional current flow between the drain and the source. This assumption is no longer valid for small geometry devices. Using a two dimensional model for source to drain current, equations for drain current ($I_D$) as a function of gate and drain voltages and pair delay time were derived and are shown below:

$$I_D = \frac{W}{L} \mu_n C_{ox} \left[ (V_G - V_T) V_D - \frac{V_D^2}{2} + \frac{C_{ds}}{C_{ox}} V_D^2 \right]$$

for

$$(V_G - V_T) > \left( 1 - \frac{C_{ds}}{C_{ox}} \right) V_D$$

and

$$I_D = \frac{W}{L} \mu_n C_{ox} \cdot \frac{\left( V_G - V_T + \frac{C_{ds}}{C_{ox}} V_D \right)^2}{2}$$

for

$$(V_G - V_T) < \left( 1 - \frac{C_{ds}}{C_{ox}} \right) V_D$$

The pair delay $T_{PD}$ is

$$T_{PD} = 14.9 \frac{L^2}{\mu_n V_S}$$

when

$$\frac{C_{ds}}{C_{ox}} = 0.2$$

5-3

where:

$W$ = Channel width in cm

$L$ = Channel length in cm

$\mu_n$ = Electron surface mobility in $\text{cm}^2/\text{volt-sec}$

$C_{ox}$ = Oxide capacitance in $\text{f/cm}^2$

$t_{ox}$ = MOST gate thickness in angstroms

$C_{ds}$ = Depletion capacitance in $\text{f/cm}^2$

$V_s$ = Supply voltage in volts

Using these equations, a graph of propagation delay time and power consumption as a function of L and $V_S$ was derived and is shown in Figure 5.0-1.

If a device were constructed with a 0.5 μm channel length and operated with $V_S$ = 0.8 volts, from Figure 5.0-1 the propagation delay would be 0.1 ns and the power delay product would be 0.0003 p-J. After taking into account that this would be the performance obtained from an ideal device without loads and that ring oscillator performance would dissipate more power and that performance would be further degraded when implementing LSI with the associated interconnect capacitances, such devices still appear to meet the criteria stated earlier. CMOS noise immunity is high for high supply voltages and decreases as the supply voltage is lowered. However, the internally generated noise also decreases correspondingly. Sensitivity to external and power supply noise will require careful handling of the filtering and shielding designs.

Present photographic techniques are not adequate to construct a mask with 0.5 μm line width. Typical devices today have 3 to 10 μm line widths, with line widths as small as 1 μm being obtained under ideal conditions. To produce detail as small as 0.5 μm, a technology using wavelengths smaller than light is required. X-ray and/or E-beam technology are the potential candidates for accomplishing this. These methods are both in the advanced development stage and are discussed in detail later.

When voltage is applied to the drain of a MOS transistor, a reverse bias junction exists between the drain and the substrate. A depletion region is formed around the drain and is a function of drain to substrate (and thus drain to source) voltage. As the source to drain spacing is decreased (or the source to drain voltage increased) this depletion region can extend to the source. When this occurs, current that is not a function of the gate

5-4
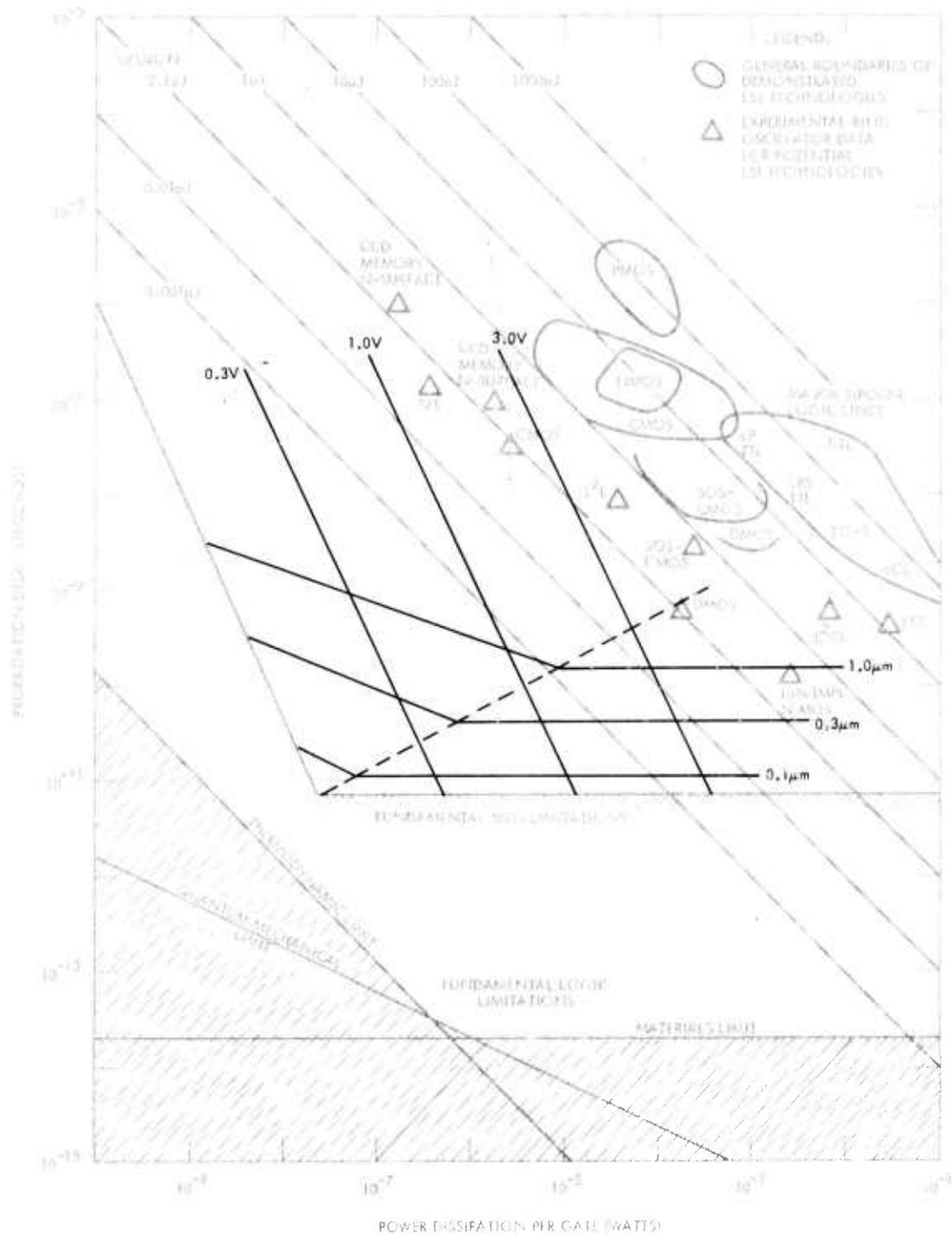
Figure 5.0-1.  Power-delay products showing ideal CMOS
device limitations as a function of gate length and device
voltage (Adapted from Swanson (5-1)).

5-5

the source to drain voltage increased) this depletion region can extend to the source. When this occurs, current that is not a function of the gate voltage will flow from drain to source. This condition is called punchthrough. The equation for punchthrough voltage from a reverse biased diode is given as

$$V_{PT} = \frac{qNL^2}{2\epsilon_s\epsilon_o}$$

where:

$q$ = Electron charge in coulomb

$N$ = Substrate doping concentration in carriers/cm$^3$

$L$ = Channel length in cm

$\epsilon_s$ = Semiconductor dielectric constant

$\epsilon_o$ = Permittivity of free space

Punchthrough voltage for $L$ = 0.5 $\mu$m and 1.0 $\mu$m as a function of substrate concentration is shown in Figure 5.0-2. For $L$ = 0.5 $\mu$m and $V_S$ = 0.8 volts, it can be seen that a minimum substrate concentration of $5 \times 10^{15}$ is required. This is within the range of concentrations used in present CMOS/SOS.

One of the critical aspects of short channel MOS is the threshold voltage variation. The expression for threshold voltage in long channel MOS devices is

$$V_T = -\frac{Q_{SS}}{C_{ox}} - \frac{Q_{FS}}{C_{ox}} + 2|\phi_f| + \frac{1}{C_{ox}}\sqrt{2q\epsilon_s N|2\phi f|}$$
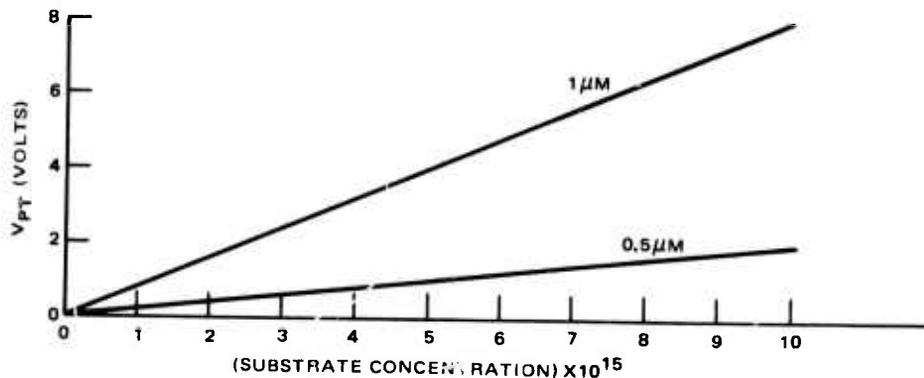


Figure 5.0-2. $V_{punch}$ through $V_s$ substrate concentration.

where:

$Q_{SS}$ = charge at oxide-semiconductor interface in coulomb/cm$^2$

$Q_{FS}$ = fast surface state charge in coulomb/cm$^2$

$\phi_S$ = fermi level referenced by mid gap in volt

$N$ = substrate concentration for carriers/cm$^3$

$C_{ox}$ = $\epsilon_{ox}/t_{ox}$ in f/cm$^2$

$\epsilon_{ox}$ = MOST gate oxide dielectric constant, 3.5 x 10$^{-11}$ farad/cm

$t_{ox}$ = oxide thickness in angstroms.

From this expression, $V_T$ is a function of the substrate concentration and the oxide thickness, but should not be a function of the device width or length. This is not the case however. As short lengths are approached, the threshold voltage drops (5-2) as is shown in Figure 5.0-3. The threshold voltage also is sensitive to device width, but this interaction in small geometry devices is not well documented. From Figure 5.0-3 it can be seen that the threshold voltage in a device with channel length of 0.5 μm will be very sensitive to channel length. This could pose a potential problem in achieving close tolerances on threshold voltages in small geometry devices over a large chip.
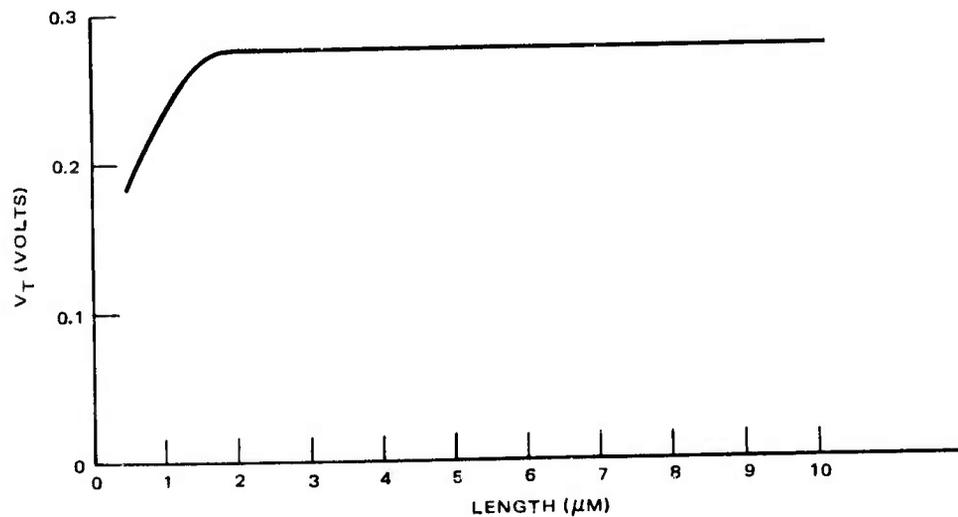


Figure 5.0-3. $V_t$ versus channel length.

Considerable research has gone into the problem of shifting threshold voltages into the ranges needed for low voltage devices (approximately 0.2 to 0.5 volts), through the use of ion implantation. A layer of dopant material is implanted just beneath the gate oxide, with the impurity level and polarity determined by the desired shift in threshold voltage. The maximum shift in voltage is approximately 5 volts, more than enough to move average threshold voltages into the 0.2 to 0.5 volt range.

If scaling down of present CMOS/SOS gate packing densities were all that was required for small geometry devices, present densities could be increased more than two orders of magnitude. But there are several potential problems. Aluminum interconnects tend to have defects that can stop conduction. Present day technology limits aluminum interconnect line widths to 1 μm or larger. Polysilicon can be made in 0.5 μm line widths, but polysilicon has approximately 200 ohms per square resistance and long lines would seriously degrade performance. Fortunately, the isolation areas required for bulk CMOS are not required for CMOS/SOS, so silicon areas can be as close together as lithographic technology permits. A typical current CMOS/SOS layout is shown in Figure 5.0-4. This chip has 7 μm channel length transistors and a packing density of 360 transistors or 90 gates/ $mm^2$. The minimum line width for the aluminum interconnects is approximately 3 μm. By scaling down to 0.5 μm channel length, the gate packing density would be $14^2$ x 90 = 17,640 gates/$mm^2$. However if the aluminum interconnects are limited to 1 μm, the maximum packing density is $3^2$ x 90 = 810 gates/$mm^2$. With the appropriate use of polysilicon for short interconnects, the goal of more than 1000 gates/$mm^2$ can be met. For non-random connected logic (such as RAM), gate densities on the order of 3000 gates/$mm^2$ are anticipated.

Work now under way at the Hughes Newport Beach and Carlsbad facilities may permit even greater packing densities. Methods of using multiple layer metalization are being investigated. Using 2 layers of aluminum would allow greater numbers of transistors to be interconnected in a given area, greatly improving the LSI design flexibility. Both LSI interconnect and logic function organization represent a greater risk than the actual device capabilities.
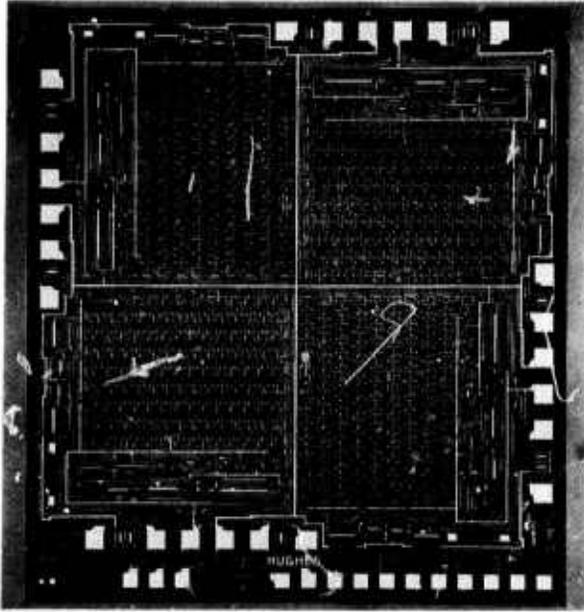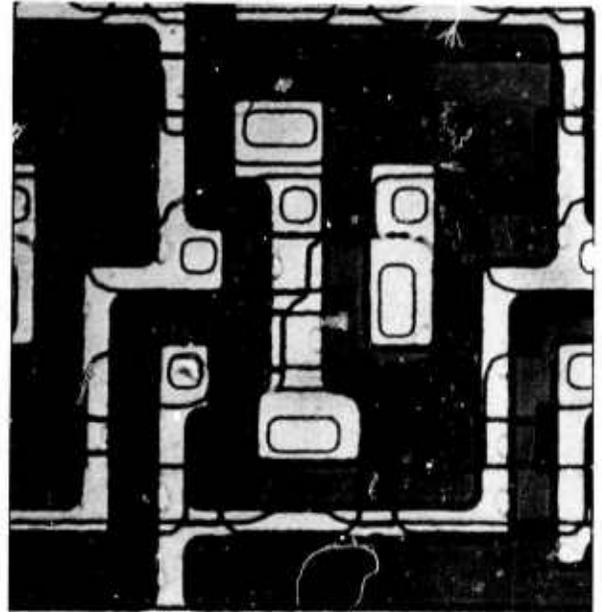
CHIP SIZE: 130 x 138 MILS

1 MIL

Figure 5.0-4.   CMOS/SOS 256 bit static  shift register.

In summary,  this section has shown that a device that can operate in
an LSI logic circuit at clock rates exceeding 100 MHz with a power delay
product of less than 0.1 p-J (possibly <0.01 p-J) and with a packing density
greater than 1000 gates/mm$^2$ should be achievable within six years with low
risk.   Such devices would be CMOS on an insulating substrate,  probably sap-
phire,  and would use channel lengths on the order of 0.5 μm.   How this
device would compare to presently available logic is shown in Figure 5.0-5;
performance would be more than an order of magnitude better than anything
available today.   The critical technology development requirements to rea-
lize this capability are

- Lithography capable of 0.5 μm line widths over a 200 mil
  square chip.

- CMOS on insulator process development with emphasis
  upon threshold voltage uniformity and  reduction of leak-
  age current.

- Advances in LSI on chip interconnect technologies and
  logic organization.
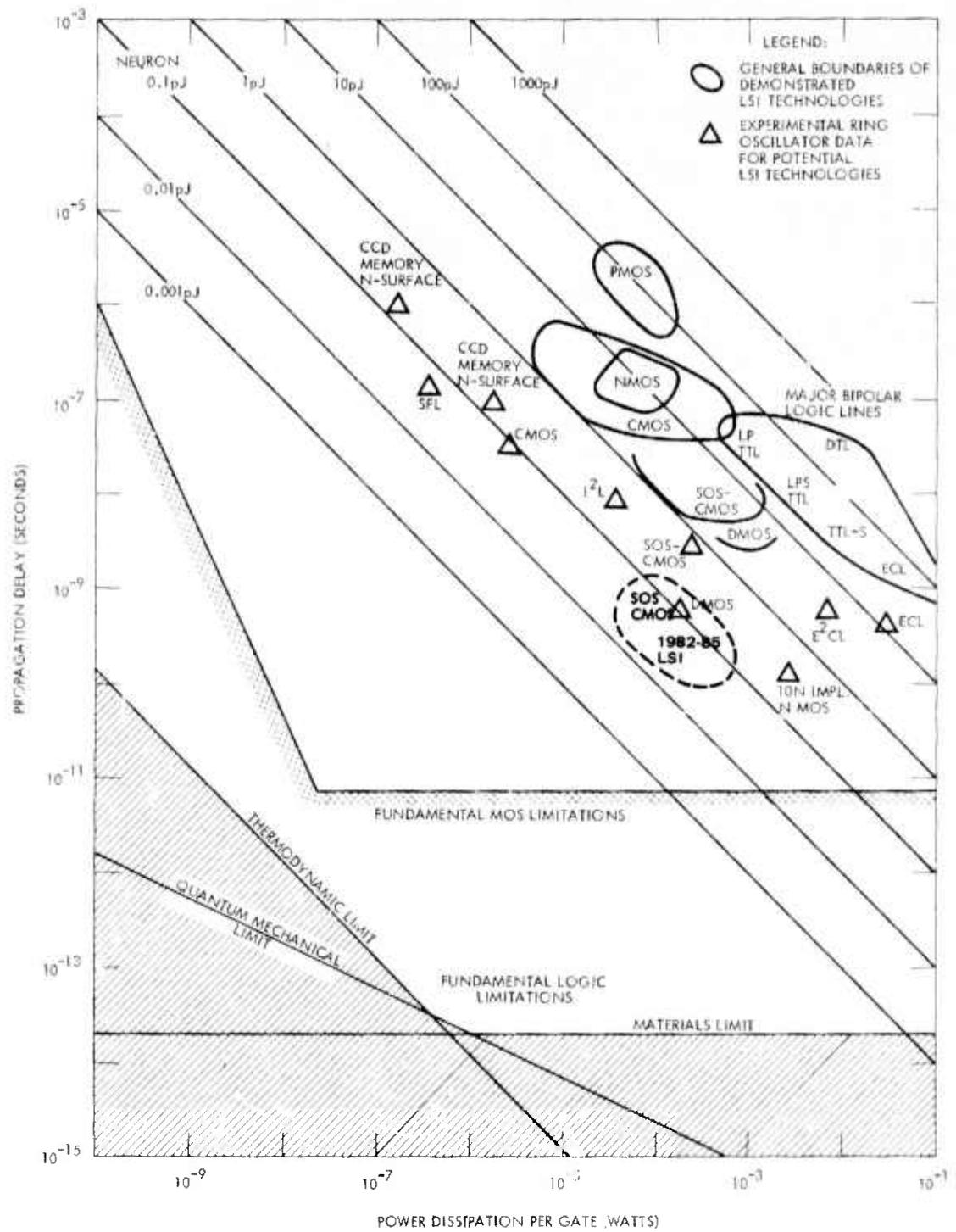
Figure 5.0-5.   CMOS/SOS design LSI performance expectations.

The critical CMOS/SOS gate device characteristics (low risk) anticipated are:

- 0.4 to 0.8 nsec LSI gate delay
- 0.05 to 0.15 p-J/gate LSI power delay product
- >1000 gates/mm$^2$ for random LSI logic (28,000 gates/chip 200 mil$^2$ chip maximum)
- >3000 gates/mm$^2$ for well organized (RAM) LSI logic (84,000 gates/chip 200 mil$^2$ chip maximum)

## References

(5-1)   Swanson, R.M., "Complimentary MOS Transistors in Micropower Circuitry," Stanford University Technical Report No. SU-SEL-074-055. December 1974.

(5-2)   Gaehsslen, F.H., Riedout, V.L., and Walker, E.J., "Design and Characterization of Very Small MOSFETS for Low Temperature Operation," December 1975.

(5-3)   Goser, K., Pomper, M., and Thianyi, J., "High-Density Static ESFI MOS Memory Cells," IEEE Journal of Solid State Circuits, Vol. SC-9, No. 5, October 1974.

# 6.0  LOGIC ARRAYS AND FUNCTIONS

In the following section two examples of critical digital functions are described that need to be implemented in order to achieve the cost/performance goals of the APSP program. The two are: arithmetic units, particularly a high speed multiply function, and a micro processor implemented in high density, low power technologies. Because of its use in the tracking function, the latter has been named the $\mu$PT, for micro processor tracker. To provide the low power and high speed required, the APSP will be packaged in a single hybrid package.

## 6.1  HIGH SPEED MULTIPLY

Due to the large number of multiply instructions expected in tracking computations, it is desirable to have a dedicated portion of the $\mu$PT's arithmetic unit composed of a high speed (parallel) multiplier.

Figure 6.1-1 is a functional diagram of a standard expandable 4 x 4 multiplier made up of Full adders (FA) and appropriate delays ($\tau$). This 4 x 4 array can be stacked to accomplish the 16 x 16 bit multiply used in the $\mu$PT.

To implement the multipliers, it is necessary to develop techniques for generating all the $a_i b_j$ products at the various times they are needed. Using gate equivalents for a full adder and gate equivalents for a half adder it is estimated the above scheme will require approximately 18,000 gate equivalents, which is within the realm of feasibility for 1985 low power technology.

## 6.2  APSP TRACK PROCESSOR

The Track Processor is the link between the Signal Processing elements of the APSP and the Data and Control Processor (reference figure 4.0-1). The Track Processor receives filtered data, in the form of "hits"
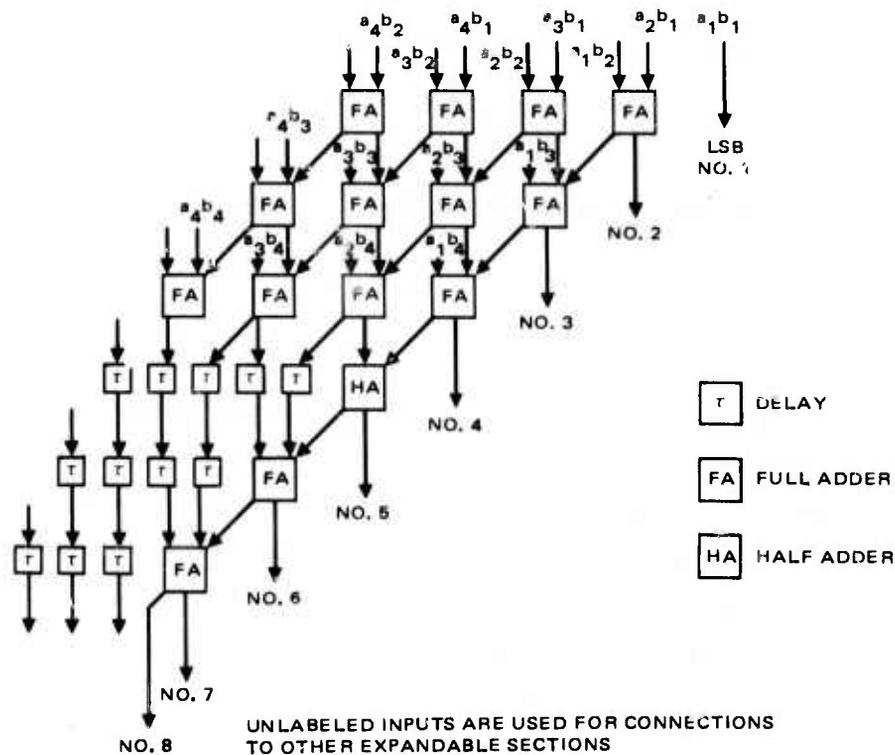
Figure 6.1-1. Expandable 4 x 4 multiplier.

(defined as threshold excessions, i. e., potential targets), from the Signal Processor. These data undergo correlation with previous frame data (commonly called tracking). Hits that appear to be the logical continuations of target tracks are used to update those tracks, while those that are found to be clutter are discarded. Data describing all tracks currently being monitored is sent periodically to the Data and Control Processor. The general requirements which drive the μPT design are summarized below.

## 6.2.1 Requirements

### Throughput Requirement

The computation rate for the Track Processor is derived from the expected hit rate, which is estimated to be a maximum of 1 pixel of every 100 in the MFPA containing a hit. For a single MFPA chip (128 x 128 detectors) this implies at least 160 hits per 0.1 sec frame time. Assuming a Poisson distribution of hits over the entire focal plane, $\mu + 3\sigma \simeq 200$ hits.

Given that the frame time is 0.1 seconds, the processing time for one hit averages 50 msec. The number of instructions executed to process one hit is estimated to be between 2000 and 4000, thereby giving a throughput requirement for the μPT of 4 to 8 MIPS.

## Memory Requirements

A word length of 16 bits is sufficient to represent all measurements in the APSP, as well as allowing overflow bits for arithmetic operations. Sixteen bits allow a powerful instruction format. The quantity of memory of the μPT will be 7K words, 5K for programs, constants and variables, 1K for current hit data and 1K for the alternate hit buffer. The size of the hit buffer is based on the maximum number of hits per frame being less than 200. As described in the processor architecture section, overloading, i.e., more than 200 hits per frame, is accommodated by raising the adaptive threshold under the control of the μPT. Each hit will be transmitted from the Signal Processor as a data block. The size of each data block is estimated to be 4 words. Thus the data input from the signal processor for one frame will be approximately 200 x 4 words, which is less than 1000 words.

## I/O Requirements

The μPT will have the following I/O functions:

1. input of hit data from signal processor,
2. output of thresholding and algorithm selection information to the signal processor,
3. input and output of target chip-boundary crossing information to and from 8 neighbors,
4. output track data onto track bus,
5. program load from Data and Control Processor,
6. receive external interrupts from Data and Control Processor.

The μPT should have dedicated interfaces for each of those I/O functions.

## 6.2.2 Architecture

Generally, the μPT (shown in figure 6.2-1) is a bus-organized 16 bit processor, the special features of which are: 128 fast access registers located on the arithmetic chip, 16 x 16 bit fully parallel multiply network (2 clock cycle full multiply), a 7-level priority interrupt structure, autonomous I/O interfaces, and a dual port, automatically switching memory.

## 6.2.3 Implementation

Given the above requirements as well as requirements for minimum power and size, the technology assumed for implementation in CMOS/SOS, (complementary MOS/Silicon on Saphire). The two key factors that set the μPT apart from all current and near future micro processors are the high gate densities on the chip and the very low logic delay and small access times of memories.

The gate densities expected are in the range of 7000 to 28,000 gates of random logic per chip, on a chip with 40 to 200 pads. For the mid 1980 time



Figure 6.2-1. μPT block diagram.

6-4
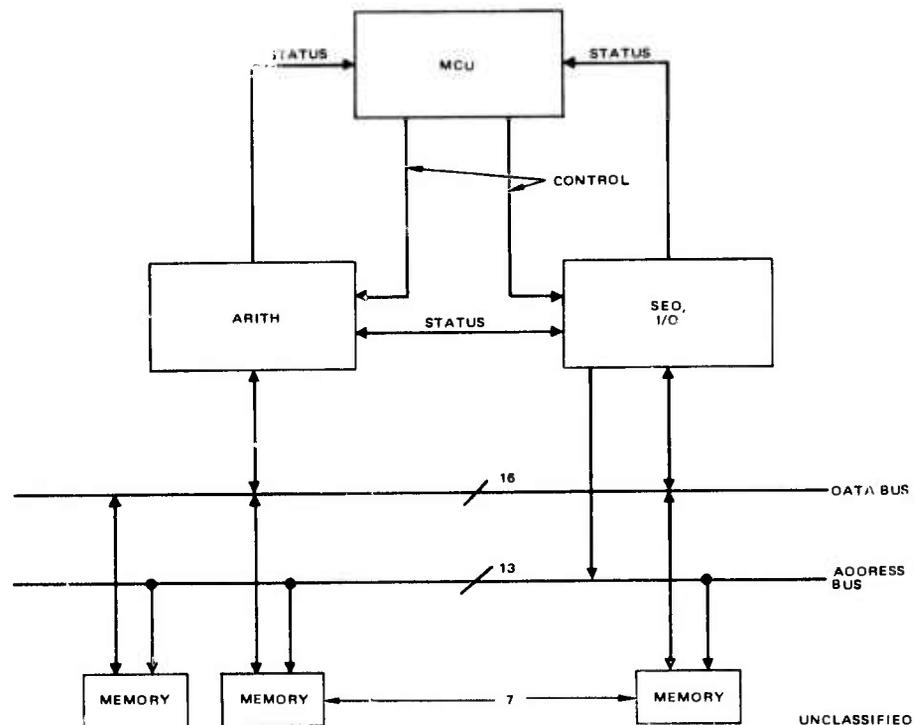
period, this is a low risk requirement. In determining the partitioning of the architecture, the gate equivalent circuits were kept under 25,000 gates. This will allow for undetermined logic increase and less than maximum density on any chip.

The requirement of 8 MIPS imposes a clock rate determined by the average number of minor cycles per instruction. Assuming 5 minor cycles per instruction, the clock rate would be 40 MHz.

The delays assumed for this technology are

| Item | Delay (nanoseconds) |
|---|---|
| Gate delay | 0.4 |
| RAM access | 8 |
| ROM access | 8 |
| Off chip connect | 2 |

Using a 40 MHz clock and 0.4 nsec gate delay, 62 gates is the longest allowable chip path and is more than required. Hence, the gate level design will be constrained to this figure for all single cycle operations.

## 6.2.4  Partitioning

The μPT consists of 9 chips:

1.  An arithmetic chip which contains the 128 general registers, the arithmetic-logic unit (ALU), the multiply network, and related functional units;

2.  A sequencing and I/O chip which contains the program counter, the interrupt structure, the memory accessing hardware and the autonomous I/O interfaces;

3.  Two memory chips, each containing 4096 words, 16 bits each;

4.  A microprogram control chip which contains the microprogram control unit for the entire μPT.

### 6.2.4.1  The Arithmetic Chip

The architecture of the Arithmetic chip is shown in figure 6.2-2. Except for various control and status lines, the only data path leading off this chip is the Main Bus. Memory data will be transmitted and received over the Main Bus.
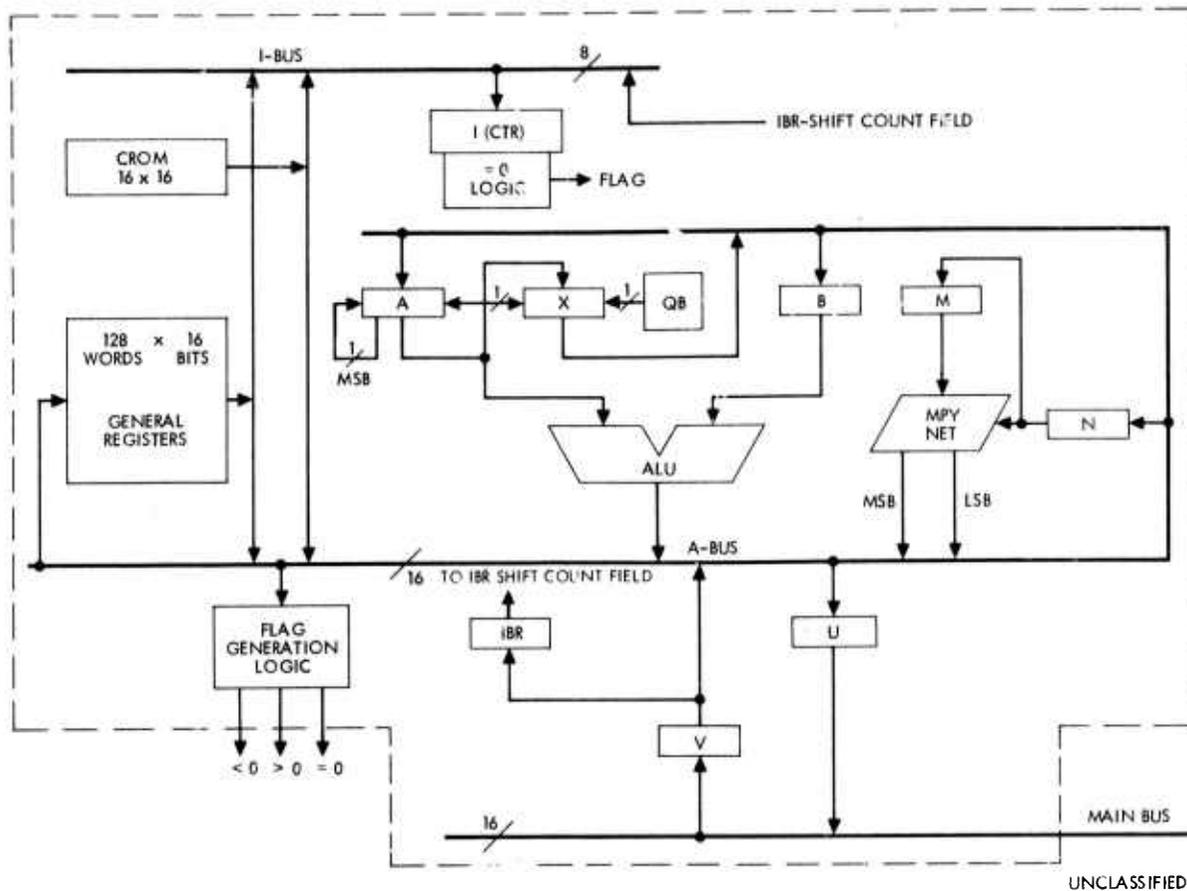
Figure 6.2-2. Register level diagram of arithmetic chip.

The chip contains two internal busses: the Arithmetic Bus (A-Bus) which performs most register transfers on the chip, and the Iteration Counter Bus (I-Bus) which allows selection of inputs to the Iteration Counter (I). A group of 128 16 bit registers is provided for the user program, and offer fast access (1 cycle). The constant read-only memory (CROM) contains certain constants and masks necessary in the implementation of the instruction set. The CROM is addressed by the microprogram. The CROM is 16 bits wide and its length is estimated to be 16 words.

The Iteration Counter is an 8-bit up-down counter used in the implementation of iterative instructions, such as shifts, block moves, and division.

6-6

The ALU has two 16 bit inputs designated A (left) and B (right). The output of the ALU is one of the following functions of A and B:

| | | |
|---|---|---|
| A | A. OR. B | -B |
| B | A. AND. B | all O-s |
| A + B | A. XOR. B | all 1-s |
| A - B | $\overline{A}$ | |

In addition to the 16-bit result, the ALU detects overflow for the operations

A + B, A - B and -B.

The Multiply Network performs fully parallel multiplication of two 16-bit numbers in two clock cycles. The Multiply Network has two 16-bit inputs (multiplicand and multiplier) and two 16-bit outputs (most and least significant words of the result) on the A-Bus. The instruction buffer register holds the instruction currently being processed. It is 16 bits wide and can be loaded from the V-register.

The Flag generation logic network monitors the value on the A-Bus. Three flags are generated: A - Bus = 0, A - Bus > 0 and A - Bus < 0. All three flags are used by the microprogram control unit (MCU) for branching.

### 6.2.4.2 Sequencing and I/O Chip

The sequencing and I/O chip is shown in figure 6.2-3. The data paths leading off this chip are: 1) Main Bus (16 bits), 2) Address Bus (13 bits), 3) Input to Track-Bus (16 bits), 4) Two-way to neighbors (8 bits), 5) Output to Signal Processor (1 bit), 6) Input from previous Vector Buffer Controller (1 bit) and 7) Output to subsequent Vector Buffer Controller (1-bit).

Internally this chip contains another bus, the Program Counter Bus (PC-Bus), which allows selection of inputs to the Program Counter (PC). The Memory Address Register (MAR) consists of a 13-bit address and an "indirect" bit. These correspond to the 13 LSB and the MSB of a 16 bit word, respectively. The MAR can be loaded from the Main Bus only. There are three sources of memory addresses in the μPT: The Program Counter, addresses in the instruction stream, and indirect addresses.
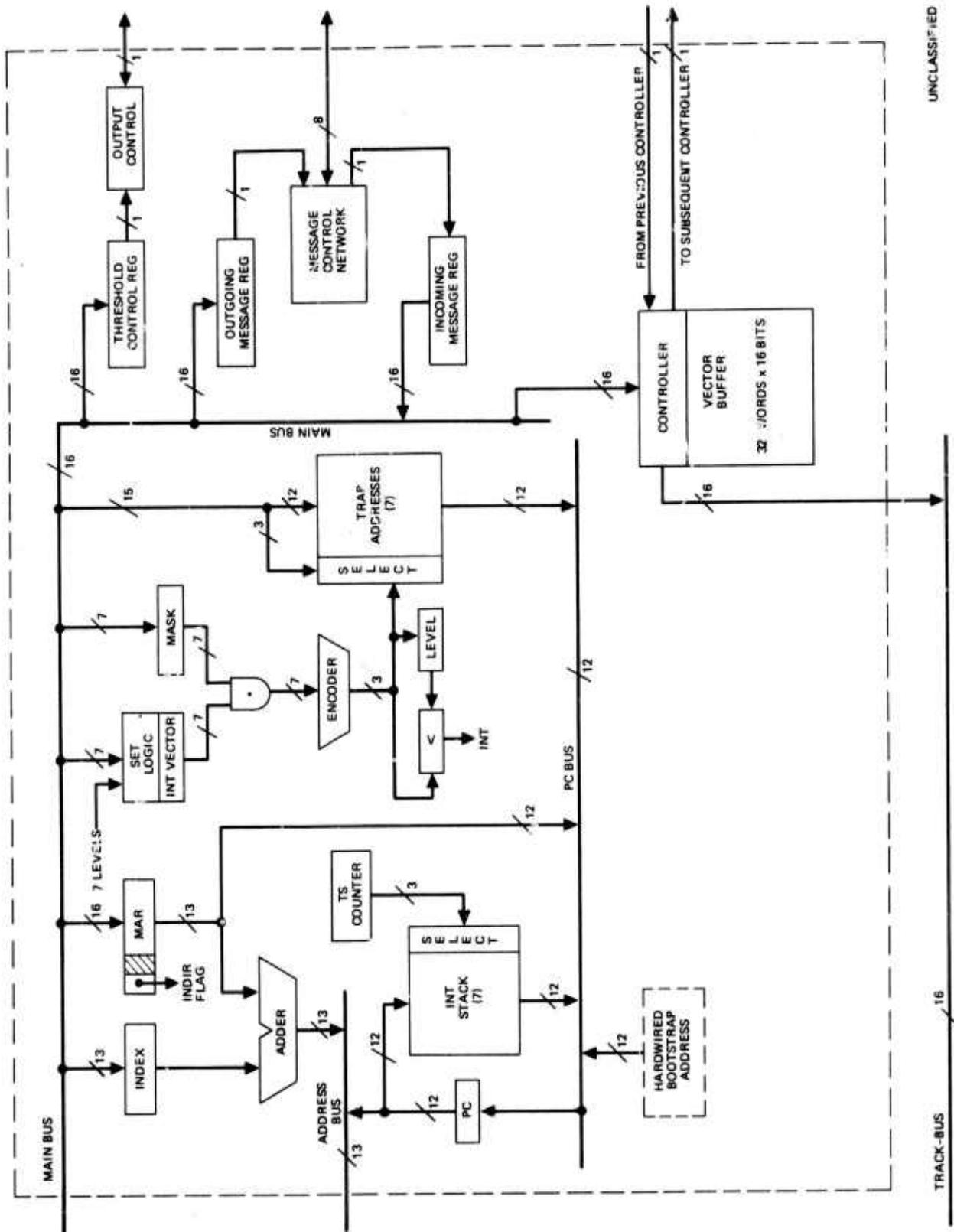
Figure 6.2-3. Sequencing and I/O chip functional block diagram.

The Index register is 13-bits wide and is used to hold the contents of one of the General Registers when indexed addressing is used. The Index Adder is used for every memory access, when the address originates from the MA1 (indexed or unindexed). The Index Adder can produce 2 possible results: INDEX + MAR or MAR. The output of the Index Adder is 13 bits wide and can be transmitted to the memory chips via the Address-Bus. The Interrupt Vector (IV) consists of a 7-bit register and associated logic. The 7-bit Interrupt Mast register (IM) can be used to suppress the servicing of any interrupt levels. The contents of the IM are ANDed with the IV before any further decisions are made.

The Program Counter (PC) is a 13-bit counter loaded from the PC-Bus, and contains the address of the next instruction to be executed. Addresses from the PC can be transmitted to the memory chips over the Address Bus.

There are 7 Trap Address Cells (TAC), a 12 bit wide register each corresponding to one interrupt level. Each TAC contains a memory address, and whenever the respective level of interrupt occurs, interrupt handling is started at that address.

The Interrupt Stack consists of 7 registers corresponding to 7 nested interrupts. Each register consists of a 12-bit resumption address and a 3-bit resumption level. The Vector Buffer represents the I/O interface with the Track-Bus, and contains one cell of storage for each track. Currently, it appears that 32 such cells will be sufficient. Each cell will be composed of a number of 16-bit words.

The Threshold Control Register (TCR) is parallel input/serial output organized, and provides the I/O interface between the μPT and the corresponding Signal Processor. The Message Control Network (MCN) together with the Incoming Message Register, (IMR) and the Outgoing Message Register (OMR) form the I/O interface with the neighboring μPT's.

### 6.2.4.3 Memory Chip

The RAM Memory chip has been discussed in Section 4.1.

### 6.2.4.4 The Microprogram Control Unit (MCU)

The MCU chip is shown in figure 6.2-4. The following data paths lead off the MCU chip:

1. Encoded Control Signals to all other chips of the μPT.
2. Status Flags from all other chips of the μPT.



Figure 6.2-4. The microprogram control unit register level diagram.

6-10

3. Op-code from IBR on the Arithmetic Chip.

4. INT-flag from the Sequencing and I/O chip.

The MCU consists of a 512 word x 32 bit ROM, and a Command Register which can hold one word from the ROM. During each minor cycle one word is read from the ROM and placed in the Command Register.

The 9-bit address of the next ROM-word to be fetched can come from one of 3 sources: 1) 8 bits from the Next Address Field of the Command Register, concatenated with one bit based on a status flag; 2) 6 bits from the Op-code field of the IBR with three zeros as MSB's; or 3) a hardwired address used to branch into a section of the microprogram dedicated to trapping interrupts.

Selection between these sources is made by the Address Multiplexor. The selection is based on three control bits: one is the INT signal from the Interrupt Structure, the others originate from the Command Register.

### 6.2.5 Track Processor Sizing

Given the above architecture, the number of gate equivalents can be estimated to determine the number of chips required to implement this $\mu$PT, and the expected power dissipation. Table 6.2-1 summarizes the gate equivalents for each function. The data from this table is further simplified by assuming 5 gate equivalents per state device, i.e., flip flop.

Further assumptions on gate equivalents are: 1) one bit of ROM corresponds to one gate, and 2) one bit of RAM corresponds to 1.5 gates. Using a figure of 16K bits per memory chip allows space for control, address decode and timing logic to be fabricated on the same chip.

#### 6.2.5.1 Arithmetic Unit

The largest portion of the Arithmetic Unit is the multiply network. The multiply network is fully parallel, and based on common algorithms is

## TABLE 6.2-1.  GATE EQUIVALENTS

| Unit | No. Bits | No. Gates | No. State Devices |
|---|---|---|---|
| ALU | 4 | 63 | -- |
| Adder | 4 | 36 | -- |
| CLA (Carry Lookahead Adder) | 4 Units | 19 | -- |
| Comparator | 4 | 31 | -- |
| Counter | 4 | 28 | 4 |
| MUX | 2:1, Quad | 15 | -- |
| MUX | 4:1, Dual | 16 | -- |
| MUX | 8:1 | 12 | -- |
| MUX | 16:1 | 26 | -- |
| Priority Encoder | 8:3 | 29 | -- |
| Register File | 4 x 4 | 38 | 16 |
| Register -Parallel I/O -Serial I/O | 4 <br> 8 | 27 <br> 47 | 4 <br> 8 |
| Register -Parallel I/O | 4 | 28 | 4 |
| Register -Serial In -Parallel Out | 8 | 4 | 8 |
| Register -Parallel In -Serial Out | 4 <br> 8 | 16 <br> 36 | 4 <br> 8 |

estimated to be on the order of 18,000 gate equivalents. The remainder of the logic sizing is as follows:

1. Registers: A, B, M, N, U, V, X (16 Bits); I (8 Bits)

    a. $(7 \text{ Regs}) \left(16 \frac{\text{Bits}}{\text{Reg}}\right) \left(\frac{1}{4} \frac{\text{Unit}}{\text{Bits}}\right) \left[27 \frac{\text{g.e.}}{\text{Unit}} + \left(4 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] = 1316 \text{ g.e.}$

    b. $(1 \text{ Reg}) \left(8 \frac{\text{Bits}}{\text{Reg}}\right) \left(\frac{1}{4} \frac{\text{Unit}}{\text{Bits}}\right) \left[27 \frac{\text{g.e.}}{\text{Unit}} + \left(4 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] = 94 \text{ g.e.}$

2. ALU/CLA:

$(16 \text{ Bits}) \left(\frac{1}{4} \frac{\text{Unit}}{\text{Bits}}\right) \left(63 \frac{\text{g.e.}}{\text{Unit}}\right) + (4 \text{ Units}) \left(\frac{1}{4} \frac{\text{CLA}}{\text{Units}}\right) \left(19 \frac{\text{g.e.}}{\text{CLA}}\right) \cong 275 \text{ g.e.}$

3. CROM:

$\left(16 \frac{\text{Bits}}{\text{Word}}\right) (16 \text{ Words}) \left(1 \frac{\text{g.e.}}{\text{Bit}}\right) = 256 \text{ g.e.}$

4. General Registers:

$(128 \text{ Words}) \left(16 \frac{\text{Bits}}{\text{Word}}\right) \left(1.5 \frac{\text{g.e.}}{\text{Bit}}\right) = 3072 \text{ g.e.}$

5. Flag generation logic (comparator)

$(16 \text{ Bits}) \left(\frac{1}{4} \frac{\text{Unit}}{\text{Bits}}\right) \left(31 \frac{\text{g.e.}}{\text{Unit}}\right) = 124 \text{ g.e.}$

6. Miscellaneous logic, including I = 0, QB, Command Decode, etc., is estimated at 10%.

0.  19000

1.   1410

2.    275

3.    256

4.   3072

5.    124
     ―――――
     24137

6.   2413   (10%)
     ―――――
     26,550 gate equivalents for the Arithmetic Unit

### 6.2.5.2 Sequencing and I/O Unit Sizing

1.  Registers: INDEX (13), MAR (16), IV (7), MASK (7), TC (16), OMR (16), IMR (16)

a.  16 Bit PI/SO: (TC, OMR)

$$(2 \text{ Regs}) \left(16 \frac{\text{Bits}}{\text{Reg}}\right) \left(\frac{1 \text{ Unit}}{4 \text{ Bits}}\right) \left[16 \frac{\text{g.e.}}{\text{Unit}} + \left(4 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] = 288 \text{ g.e.}$$

b.  16 Bit SI/PO: (IMR)

$$(1 \text{ Reg}) \left(16 \frac{\text{Bits}}{\text{Reg}}\right) \left(\frac{1 \text{ Unit}}{8 \text{ Bits}}\right) \left[4 \frac{\text{g.e.}}{\text{Unit}} + \left(8 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] = 88 \text{ g.e.}$$

c.  16 Bit PI/PO: (MAR)

$$(1 \text{ Reg}) \left(16 \frac{\text{Bits}}{\text{Reg}}\right) \left(\frac{1 \text{ Unit}}{4 \text{ Bits}}\right) \left[28 \frac{\text{g.e.}}{\text{Unit}} + \left(4 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] = 192 \text{ g.e.}$$

d.  13 Reg PI/PO: (INDEX)

$$(1 \text{ Reg}) \left(13 \frac{\text{Bits}}{\text{Reg}}\right) \left(\frac{1 \text{ Unit}}{4 \text{ Bits}}\right) \left[28 \frac{\text{g.e.}}{\text{Unit}} + \left(4 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] = 156 \text{ g.e.}$$

e.   7 Bit PI/PO:  (IV, MASK)

$$(2 \text{ Reg}) \left(7 \frac{\text{Bits}}{\text{Reg}}\right) \left(\frac{1 \text{ Unit}}{4 \text{ Bits}}\right) \left[28 \frac{\text{g.e.}}{\text{Unit}} + \left(4 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] = 168 \text{ g.e.}$$

2.      Counters:  PC (12), TS (3)

   a.   PC:

$$(1 \text{ CTR}) \left(12 \frac{\text{Bits}}{\text{CTR}}\right) \left(\frac{1 \text{ Unit}}{4 \text{ Bits}}\right) \left[28 \frac{\text{g.e.}}{\text{Unit}} + \left(4 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{Unit}}\right)\right] = 144 \text{ g.e.}$$

   b.   TS:

$$(1 \text{ CTR}) \left(3 \frac{\text{Bits}}{\text{CTR}}\right) \left(\frac{1 \text{ Unit}}{4 \text{ Bits}}\right) \left[28 \frac{\text{g.e.}}{\text{Unit}} + \left(4 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{Unit}}\right)\right] = 36 \text{ g.e.}$$

3.      Index Adder:

$$(1 \text{ Adder}) \left(13 \frac{\text{Bits}}{\text{Adder}}\right) \left(\frac{1 \text{ Unit}}{4 \text{ Bits}}\right) \left(36 \frac{\text{g.e.}}{\text{Unit}}\right) = 156 \text{ g.e.}$$

4.      Interrupt Stack:

$$(7 \text{ Words}) \left(16 \frac{\text{Bits}}{\text{Adder}}\right) \left(\frac{1 \text{ Unit}}{4 \text{x} 4 \text{ Bits}}\right) \left[38 \frac{\text{g.e.}}{\text{Unit}} + \left(16 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] = 826 \text{ g.e.}$$

5.      Encoder:

$$(1 \text{ Enc}) \left(\frac{8 \text{ to } 3}{\text{Enc}}\right) \left(\frac{1 \text{ Unit}}{8 \text{ to } 3}\right) \left(29 \frac{\text{g.e.}}{\text{Unit}}\right) = 29 \text{ g.e.}$$

6.      Trap Address Cells:

$$(7 \text{ Words}) \left(12 \frac{\text{Bits}}{\text{Word}}\right) \left(\frac{1 \text{ Unit}}{4 \text{x} 4 \text{ Bits}}\right) \left[38 \frac{\text{g.e.}}{\text{Unit}} + \left(16 \frac{\text{f.f.}}{\text{Unit}}\right) \left(5 \frac{\text{g.e.}}{\text{f.f.}}\right)\right] \cong 620 \text{ g.e.}$$

7.  Message Control Network: ≅ 5000 g.e.

8.  Bus Control: Main = 16 g.e.

PC = (12) (4) = 48 g.e.

Address = (13) (2) = 26 g.e.

9.  Vector Buffer:

a.  RAM: (n = 8) (256) (16) (1.5) = 6144 g.e.

b.  Controller: ≅ 450 g.e.

10.  Output Control: 250 g.e.

11.  Miscellaneous Logic = 10%

| 1.  | 892 |
| 2.  | 180 |
| 3.  | 156 |
| 4.  | 826 |
| 5.  | 29 |
| 6.  | 620 |
| 7.  | 5000 |
| 8.  | 90 |
| 9.  | 6594 |
| 10.  | 250 |
|  | 14637 |
| 11.  | 1463    (10%) |

16, 100 g.e. /SEQ and I/O UMT

### 6.2.5.3 Memory Unit

The assumed organization is 4K words by 16 bits. Because of the large amount of RAM compared to the logic on the chip, an estimate of 80K g.e. for the memory area was used. This is due primarily to the uniformity of the memory versus a random logic array. The logic required on the chip for memory related functions and the switching controller brings the memory chip to a maximum of about 82,000 g.e.

### 6.2.5.4 Micro Program Control Unit

The bulk of the logic in the MCU consists of the ROM and the Command Register (CR). The remaining logic and hardware such as the next address mux, flag select mux, etc., will be treated as a small percentage of the ROM and CR.

ROM: $(512 \text{ words}) (32 \text{ Bits/Word}) (1 \text{ g.e./Bit}) = 16,384 \text{ g.e.}$

CR: $(32 \text{ Bits}) \left( \frac{1 \text{ Unit}}{4 \text{ Bits}} \right) \left[ 27 \frac{\text{g.e.}}{\text{Unit}} + \left( 4 \frac{\text{f.f.}}{\text{Unit}} \right) \left( 5 \frac{\text{g.e.}}{\text{f.f.}} \right) \right] = 376 \text{ g.e.}$

Thus we have 16,760 gate equivalents for the ROM and Control Register. Allowing a 15% expansion factor gives approximately 19,000 gate equivalents for the MCU.

### 6.2.6 Off Chip Connections

Another consideration in designing and partitioning the CPU and memory was the number of pads on each chip. The feasible maximum for this figure is considered about 200.

1. Arithmetic Unit

| Inputs | Pads | |
|---|---|---|
| Command | 8 | (Avg) |
| Main Bus | 16 | |
| Power/Ground | 2 | |
| Clock | 1 | |
| Outputs | | |
| Flags | 5 | |
| OPCODE | 6 | |
| Total: | 38 Pads | |

2. Sequencing and I/O

| Inputs | Pads | |
|---|---|---|
| Interrupts | 7 | |
| Command | 8 | (Avg) |
| Main Bus | 16 | |
| Output Control | 1 | |
| Message Link | 8 | |
| Buffer Controller | 1 | |
| Power/Ground | 2 | |
| Clock | 1 | |

| Outputs | Pads |
|---|---|
| Address Bus | 13 |
| Buffer Output | 16 |
| Buffer Controller | 1 |
| Flags | 5 |

Total: 79 Pads

3. Memory Unit

| Inputs | Pads |
|---|---|
| Parallel Data | 16 |
| Serial Data | 1 |
| S/P Select | 1 |
| R/W Select | 1 |
| Enable | 1 |
| I.D. | 3 |
| Address | 13 |
| Count | 1 |
| $\Delta T$ | 1 |
| $\Delta T$ Reset | 1 |
| Power/Ground | 2 |
| Clock | 1 |

| Outputs | Pads |
|---|---|
| Main Bus | 16 |
| Request Complete | 1 |

Total: 59 Pads

4. Microprogram Control Unit

| Inputs | Pads | |
|---|---|---|
| Status Flags | 16 | (Max) |
| OPCODE | 6 | |
| Interrupt | 1 | |
| Power/Ground | 2 | |
| Clock | 1 | |

| Outputs | |
|---|---|
| Control Signals | 18 |

Total: 44 Pads

## 6.2.7 Power Requirements

Arithmetic Unit

Static: $(25,000 \text{ g.e.}) \left(2 \dfrac{nW}{g.e.}\right) = 0.05 \text{ mW}$

Dynamic: $(4750 \text{ g.e.}) (40 \text{ MHz}) (0.15 \text{ pJ/g.e.}) = 28.5 \text{ mW}$

Output Devices: 13.45 mW

Total:~ 42 mW

Sequencing and I/O

Static: $(16,000 \text{ g.e.}) \left(2 \dfrac{nW}{g.e.}\right) = 0.03 \text{ mW}$

Dynamic: $(6770 \text{ g.e.}) (40 \text{ MHz}) (0.15 \text{ pJ/g.e.}) = 40.62 \text{ mW}$

Output Devices: 12.35 mW

Total:~ 53 mW

Memory:

Static:  (20,000 g. e.)  $\left(2\frac{nW}{g.e.}\right)$ = 0.04 mW

Dynamic:  (2500 g. e.) (8 MHz) (0.15 pJ/g. e.) = 3 mW

Output Devices:  12 mW

Total:~15 mW

Microprogram Control Unit:

Static:  (20,000 g. e.)  $\left(2\frac{nW}{g.e.}\right)$ = 0.04 mW

Dynamic:  (2000 g. e.) (40 MHz) (0.15 pJ/g. e.) = 12 mW

Output Devices:  9 mW

Total:~21 mW

Since the complete μPT will consist of the three CPU chips along with 2 memory chips, power consumption totals approximately 146 mW.

| | |
|---|---|
| MCU | 21 mW |
| ARITH | 42 |
| SEQ and I/O | 53 |
| MEM (2) | 30 |
| Total: | 146 mW |

This assumes the processor is assembled in a large area hybrid, utilizing a dielectric substrate and low capacitance interconnects.

# 7.0 CUSTOM LOGIC CHIP SUMMARY AND SCHEDULES

Preliminary design has been provided for the ASE, $\mu$PT, Memory (SPS and RAM) chip concepts and the basic CMOS/SOS logic device design has been examined. The risks and requirements of Ultra High Density LSI using E-Beam Lithography have been presented. In general the primary areas requiring development are: (1) E-Beam Lithography, (2) CMOS/SOS small device process development, (3) threshold voltage uniformity, (4) interconnect techniques and logic organization for ultra-high density LSI.

Figure 7-1 illustrates a recommended 30 month process development program designed to provide a demonstration of E-Beam CMOS/SOS ring oscillator (0.01 pJ, 0.2 $\mu$sec gate delay) in $\phi$1, and output devices and SSI devices intended to show LSI compatibility in $\phi$2.

Table 7-1 provides an estimate of the development schedule for all APSP critical device development major tasks. It is anticipated that by the second quarter of 1981 all brassboard demonstration chips required for the APSP can be designed, processed and tested.
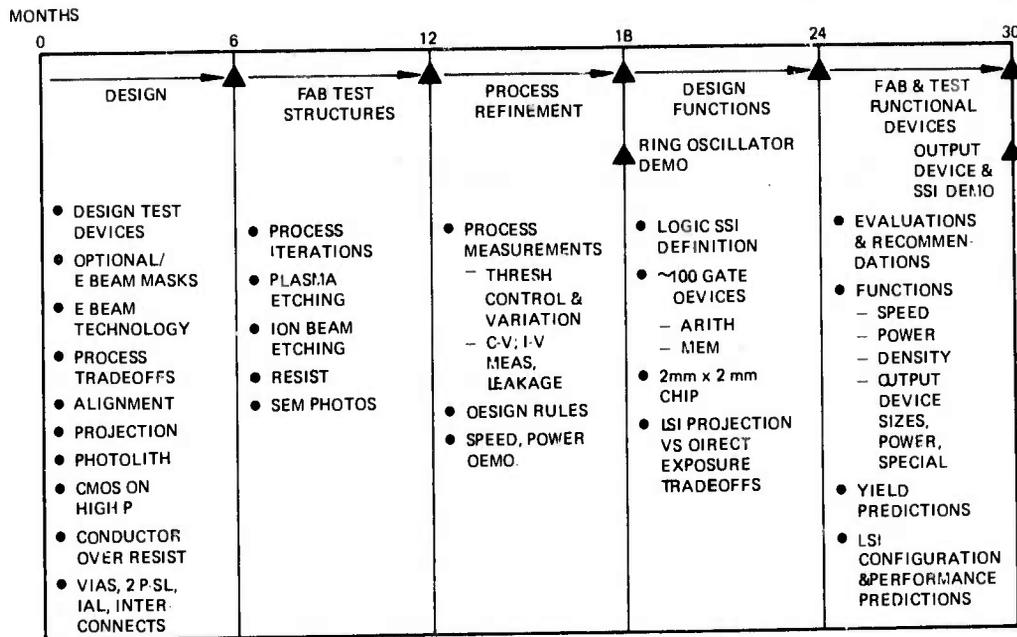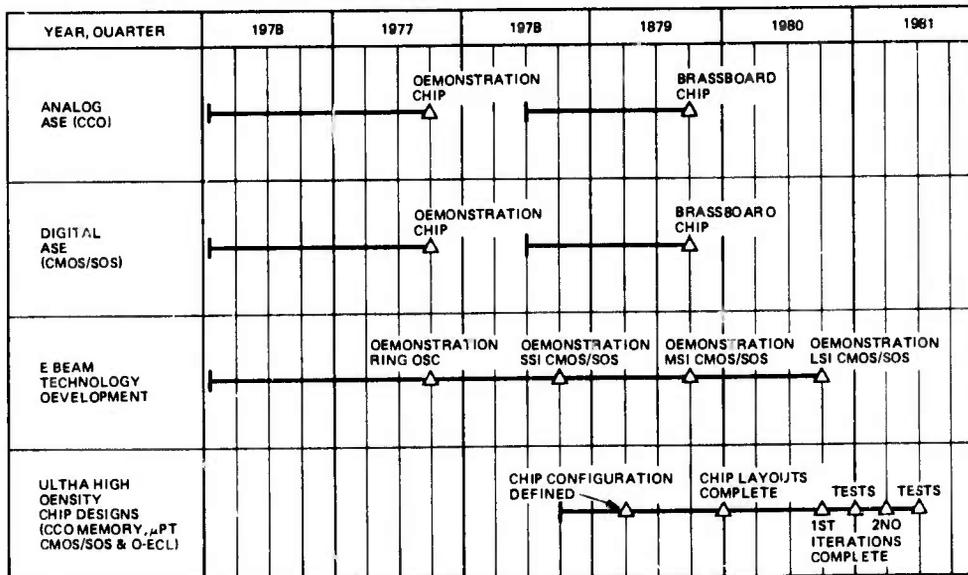
MONTHS



Figure 7-1.  E-Beam CMOS/SOS microfabrication
process development program.

TABLE 7-1.   APSP ESTIMATED OVERALL CUSTOM
LOGIC CHIP SCHEDULES

(Assuming April 1, 1976 start)

DEVELOPMENT AND TEST PLAN FOR AN ADAPTIVE PROGRAMMABLE
SIGNAL PROCESSOR

This section describes a plan to design, fabricate and test a feasibility
demonstration model of an Adaptive Programmable Signal Processor. The
plan (CDRL A009) provides a program having two phases, eight major sub-
tasks, and a thirty-three month duration. Included are development of
hardware, firmware, software, test equipment and critical technologies.

The primary objectives of the plan are to:

- Define a modular adaptive processor having the flexibility and
  programmability needed to provide a multi-mission capability
  for both surveillance and commanded search modes.

- Design and construct a breadboard adaptive programmable
  processor element which, when supplied suitable computer-
  simulated inputs, will be able to provide the performance needed
  for target detection and tracking.

- Develop the technology required for ultra-high-density LSI
  (UHD-LSI) Circuitry in order to verify that the final design is
  capable of meeting the desired performance, density and power
  requirements.

STATEMENT OF WORK

The statement of work for the adaptive programmable signal processor
program consists the eight major tasks diagrammed and costed in Fig-
ure 1. The performance specifications for this processor are listed in
Table 1.

The contractor is to provide personnel, materials, and facilities with
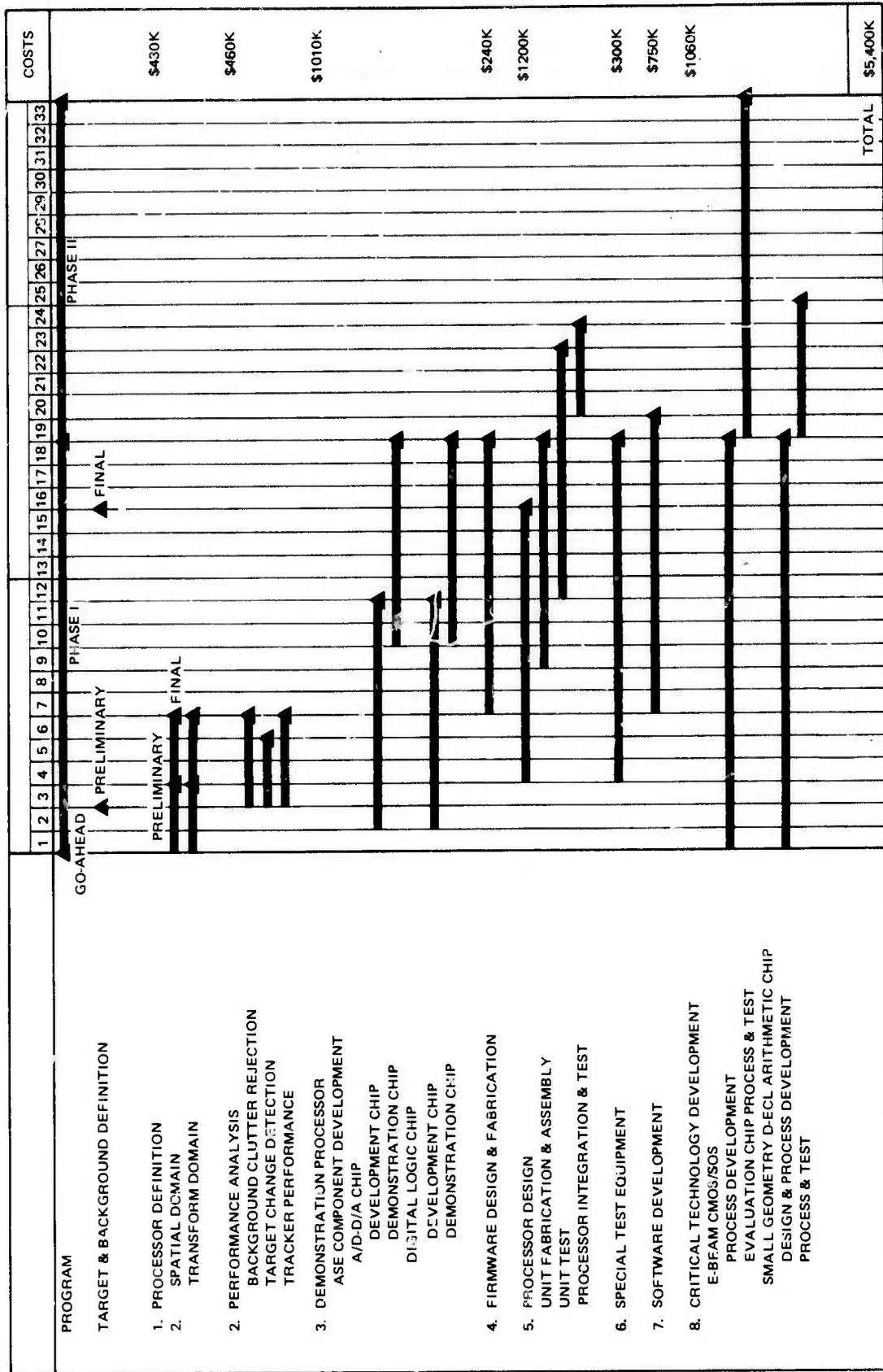the objective to complete the following development and demonstration tasks.

Figure 1. Adaptive programmable signal processor development schedule.

## TABLE 1. DEMONSTRATION ADAPTIVE PROGRAMMABLE PROCESSOR SPECIFICATIONS

| Evaluated Parameters During the Program | Specifications |
|---|---|
| Modes of operation | ATH and BTH |
| Focal plane assembly output data format | 1) Spatial domain |
| | 2) Walsh-Hadamard transform domain |
| Frame rate | 10 Hz |
| Output data format | State vectors of maximum 128-bit word length |
| | 200 state vectors per second |
| Submodule processor output bit rate | 26 Kbps |
| Overall data compression | 256 |
| System input dynamic range with adaptive control | $10^7$ |
| Signal dynamic range | $10^4$ |
| Nuclear event detection and erasure | Yes |
| A/D conversion accuracy | 10 bits |
| Gain normalization | On individual detector element basis with an accuracy of better than 1 percent |
| A/D conversion rate (serial data) | 164 kHz |
| Clutter rejection capability in BTH mode | |
| • Temporal filter | 40 to 60 db |
| • Pixel space spatial filter | 10 to 20 db |
| • Walsh-Hadamard filter | 20 to 30 db |
| Star rejection in ATH mode | 100 percent |
| Tracking accuracy ($1\sigma$ error) | 0.25 pixel |
| Number of simultaneous tracks (false and real) | 50 per track sub-module processor (200 total) |
| Throughput of track processor | 2 MIPS for each submodule processor, 8 MIPS total |

Track Processor Features

Ability to track when target moves from one detector/multiplexer array to another

Multi-target crossing tracks

Track initiation parameters; velocity, acceleration, and number of consecutive hits

Track deletion parameters; number of missed hits, velocity, and acceleration

### Task 1 - Processor definition

Provide a precise definition of two versions of an adaptive programmable signal processor configuration which will meet the requirements summarized in Table 1, assuming limited spatial and Walsh-Hadamard transform domain alalog signal preprocessing in the focal plane assembly. Perform an evaluation, and select one of the two versions for further development. An adaptive video encoder, a temporal filter, adaptive detection logic, and a programmable track processor will follow the readout device on the focal plane. The processor will contain four identical channels for processing signals from four identical detector/multiplexer array chips. The processors will interface with a module processor in a test and control unit. Development of data processing algorithims, identification of the processor/focal plane interface, and definition of the test interfaces are included in this task. It is assumed that preliminary definitions of target and background clutter will be customer-supplied within two months after program go-ahead.

### Task 2 — Performance Analysis

On the basis of the processor definition reached during Task 1 above, with the aid of computer models, analyze the performance capabilities of the processor in the presence of the nominal target and clutter scenes. These analyses will include signal-to-noise ratio, signal-to-clutter ratio, detection, and false alarm probability as well as tracking errors in the presence of multiple crossing tracks. It is assumed that the adaptive threshold will be adjusted to accept no more than 200 real and false targets at a time, for each submodule processor. The modular concept proposed permits arraying of multiple submodule processors, to achieve a several thousand target total capability.

### Task 3 — Demonstration Processor

The demonstration processor will be designed by using (1) off-the-shelf MSI components, (2) special-purpose A/D-D/A CCD/MOS components and (3) a CMOS/SOS digital logic chip in the AVE section. This task will

include design effort for support electronics and the special purpose chip test hardware and software.

## A. Special -Purpose A/D - D/A Chip

Design, build, and test an adaptive video encoder with the A/D and D/A elements, amplifiers, sample-and hold devices, CCD arithmetic functions, and interface devices on the same LSI chip. The processes and design rules needed for the CCD-compatible CMOS clock drivers and logic circuits will be developed. The design will provide optimum conversion accuracy, low power, maximum linearity, low noise, minimum geometry, and complex functional integration and also maintain component independence to ensure flexibility.

## B. CMOS/SOS Digital Logic Chip

Design, build, and test a nuclear event detection and dynamic range selection logic chip for the AVE. This chip will include approximately 2500 CMOS/SOS devices optimized for small geometry, low power, and high speed LSI. The design parameters are 0.8pj, 4-volt logic levels, and 3-nsec gate propagation delays.

The device will be tested to determine the maximum clock frequency, noise immunity, power dissipation, and the operation of the overall logic. The processes and design rules for very small-geometry, low-voltage CMOS/SOS devices and LSI interconnects will also be developed as part of this subtask. All chip design effort will include identification of fault tolerance, nuclear hardening and testability requirements.

## Task 4 — Design and Testing of Firmware

A set of processor firmware will be implemented on the basis of the instruction set selected, and will be designed in detail and tested. The design will include flow charts and microcoding as well as a microprogram simulator. The identification, design and development of the facilities required for firmware development and test are included in this task.

5

## Task 5 — Fabrication and Test of Processor

The demonstration processor will be checked by using standard computer-aided logic test programs and will be assembled on wire-wrapped boards. The volume of a given submodule must not exceed 1 ft$^3$. After unit assembly and test, the submodule will be integrated and tested by applying analog-simulated input signals and observing digital-output state vectors to the module processor. The design will be evaluated by comparing the performance parameters listed in Table 1 to the design goals and determining the limitations. Included in this task is the identification of the hardware and software required for processor evaluation and test.

## Task 6 — Special Test Equipment

The special test equipment identified in Task 5 will be developed and used to test the submodule processor and its various units.

## Task 7 — Software Development

Two major programs will be developed and tested in the submodule: the macro assembler and the tactical software. The tactical software will be a detailed implementation of the track initiation, tracking, and track deletion algorithms for both the Below-the-Horizon mode and the Above-the-Horizon mode.

## Task 8 — Critical Technology Development

### A. E-Beam CMOS/SOS

Design, fabricate, and test one LSI-compatible CMOS/SOS logic function by using electron beam lithography. Develop an E-beam microfabrication process and a compatible wafer process. The design objectives are 0.01 pj per gate, with a 0.2-nsec gate propagation delay for 2-volt ring oscillators, and 0.15 pj per gate, with a 0.6-nsec gate propagation delay with 1000 gates/mm$^2$ equivalent LSI density for 2-volt logic devices. The design rules, processes, their limitations, the interconnect design rules, their characteristics, and limitations will also be developed.

6

## B.  D-ECL Arithmetic Chip

Design, build, and test a D-ECL 8 by 8-bit multiplier.  Make the best use of minimum-geometry devices and interconnects available by means of high-resolution photolithography and thereby achieve lower power consumption and high speed.  The design objective for an overall multiplier delay is 5 to 7 nsec with total power dissipation of 1 to 1.5 watts.  This corresponds to an equivalent internal logic gate delay of approximately 100 psec and an internal equivalent logic gate power dissipation of 1 mw and an internal equivalent logic gate power-delay product of 0.1 pj.

In the course of the multiplier development, design rules for very small geometry D-ECL LSI devices and interconnects all will be developed.  These rules will be available to use with subsequent design and fabrication of LSI components.  Sample quantities of the multipliers built as part of this task will be tested to verify that their logic operates properly and to determine maximum operating throughput rate and amount of power dissipated.

The development of certain fabrication hardware unique to the above technologies also constitutes part of this task.