

UNCLASSIFIED

AD NUMBER
ADB058979
NEW LIMITATION CHANGE
TO Approved for public release, distribution unlimited
FROM Distribution limited to U.S. Gov't. agencies only; Test and Evaluation; Jun 1981. Other requests for this document must be referred to Commander, Rome Air Development Center, Attn: DCLD, Griffiss AFB, NY 13441.
AUTHORITY
AFRL/IFOIP ltr, 1 Jun 2004

THIS PAGE IS UNCLASSIFIED

LEVEL II

BS



RADC-TR-81-77, Vol I (of two)

Final Technical Report

June 1981

**MULTI-RATE SECURE PROCESSOR
TERMINAL ARCHITECTURE STUDY -
TERMINAL ARCHITECTURE**

Harris Corporation

Dr. David M. Bell

DISTRIBUTION LIMITED TO U.S. GOVERNMENT AGENCIES ONLY: TEST
AND EVALUATION; Jun 81. OTHER REQUESTS FOR THIS DOCUMENT
MUST BE REFERRED TO RADC (DCLD), GRIFFISS AFB NY 13441.

DTIC FILE COPY

**ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441**

**DTIC
ELECTE
AUG 25 1981**

D

81 8 24 515

This report contains references to the following classified document:

RADC-TR-81-77, Vol II (of two) dated June 1981, "Multi-Rate Secure Processor Terminal Architecture Study - Terminal Controller (U), SECRET.

RADC-TR-81-77, Vol I (of two) has been reviewed and is approved for publication.

APPROVED:



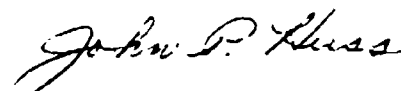
SALVATORE J. NASCI
Project Engineer

APPROVED:



BRUNO BEEK, Acting Technical Director
Communications and Control Division

FOR THE COMMANDER:



JOHN P. HUSS
Acting Chief, Plans Office

SUBJECT TO EXPORT CONTROL LAWS

This document contains information for manufacturing or using munitions of war. Export of the information contained herein, or release to foreign nationals within the United States, without first obtaining an export license, is a violation of the International Traffic in Arms Regulations. Such violation is subject to a penalty of up to 2 years imprisonment and a fine of \$100,000 under 22 U.S.C 2778.

Include this notice with any reproduced portion of this document.

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (DCLD) Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
RADC-TR-81-77, Vol I (of two) AD-B058 9796		
4. TITLE (and Subtitle)		5. TYPE OF REPORT & PERIOD COVERED
MULTI-RATE SECURE PROCESSOR TERMINAL ARCHITECTURE STUDY, A TERMINAL ARCHITECTURE, (1)		Final Technical Report 5 Jun 79 - 15 Dec 79
6. PERFORMING ORG. REPORT NUMBER		
N/A		
7. AUTHOR(s)		8. CONTRACT OR GRANT NUMBER(s)
Dr. David M. Bell		F30602-78-C-0273
9. PERFORMING ORGANIZATION NAME AND ADDRESS		10. PROGRAM ELEMENT PROJECT, TASK AREA & WORK UNIT NUMBERS
Harris Corporation PO Box 37 Melbourne FL 32901		33126F
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE
Rome Air Development Center (DCLD) Griffiss AFB NY 13441		June 1981
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. SECURITY CLASS. (of this report)
Same		UNCLASSIFIED
		15. DECLASSIFICATION/DOWNGRADING SCHEDULE
		N/A
16. DISTRIBUTION STATEMENT (of this Report)		
Distribution limited to U.S. Government agencies only, test and evaluation, June 1981. Other requests for this document must be referred to RADC (DCLD), Griffiss AFB NY 13441		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
Same		
18. SUPPLEMENTARY NOTES		
RADC Project Engineer: Salvatore J. Nasci (DCLD)		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)		
Modem Continuous Variable Slope Delta Modulation (CVSD) Secure Voice Adaptive Predictive Coding (APC) 16 KB/S Linear Predictive Coding (LPC) 9.6 KB/S Multi-Rate Processor 2.4 KB/S		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)		
The objective of this program was to study the technical concept of using a single processor for implementation of wireline modem and speech compression functions. This concept is considered a means to attain cost savings in a multi-rate secure voice terminal over other conventional approaches in which modem and voice processing are implemented separately. The digital signal processing characteristics of modem and speech compression are found to be similar. An application of this similarity is described which		

DD FORM 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

uses a microprocessor for simultaneous execution of both modem (16, 9.6, or 2.4 KB/S) and speech processing (CVSD, APC, or LPC).

Accession For	
NTIS GRA&I	☐
DTIC TAB	☒
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
B	

DTIC
ELECTE
AUG 25 1981
D

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TABLE OF CONTENTS
VOLUME I
TERMINAL ARCHITECTURE

<u>Paragraph</u>	<u>Title</u>	<u>Page</u>
1.0	INTRODUCTION AND SUMMARY.....	1
2.0	TERMINAL DEFINITION.....	3
3.0	TECHNICAL DISCUSSION.....	5
3.1	Terminal Architecture.....	5
3.1.1	Architectural Concepts.....	5
3.1.2	Architectural Design.....	6
3.2	RED Processor.....	12
3.2.1	Processor Features and Capabilities.....	14
3.2.2	Processor Loading and Memory Estimates.....	17
3.2.3	Adaptive Equalizer.....	21
3.2.4	Symbol Timing Recovery.....	27
3.3	BLACK Processor.....	29
3.4	Terminal Controller.....	32
3.5	Mechanical Packaging Concepts.....	34
4.0	UNIT COST.....	39
4.1	Current Cost Data.....	39
4.2	Future Cost Reductions.....	43
5.0	CONCLUSIONS AND RECOMMENDATIONS.....	46
APPENDICES		
I	HMSP Architectural Description.....	47
II	HMSP Voice Processing Loading Estimates.....	64
III	Multi-Rate Terminal Material List.....	82

BLANK PAGE

The Harris Government Communications Systems Division has successfully completed the Multi-Rate Secure Processor Architecture Study in accordance with contract F30602-78-C-0273. This study was initiated to investigate the feasibility of adapting the RADC 16 KB/S modem design to perform voice processing as well as modem functions. Prior approaches have been to separate the voice processing function from the modem function thereby resulting in duplication of performance capability and high terminal cost.

This study was based on the premise that if the modem and speech algorithm can be implemented in the same processor, while maintaining RED/BLACK integrity, an economical solution to the Secure Voice Terminal is possible. The objective of this study was to configure a Secure Voice Terminal capable of operating at 16 KB/S, 9.6 KB/S and 2.4 KB/S which can be produced at a cost of \$5000 (excluding COMSEC) in quantities of 10,000 equipments.

The approach pursued in this study used the RADC 16 KB/S modem as the basic starting point and expanded upon that basic design to handle both the modem and speech processing functions. The 16 KB/S modem was also used as the basis of comparison in developing the multi-rate terminal cost model and projecting the unit cost in quantities of 10,000. The experience and history base of the 16 KB/S modem program was effectively utilized throughout the study in estimating the cost, size and performance capability of the multi-rate terminal. To supplement Harris experience in voice processing, a subcontract was placed with Ketron, Inc. for consultation and an independent assessment of the Harris approach to implementation of LPC-10, APC-4 and CVSD algorithms.

The COMSEC, key distribution and RED/BLACK problems unique to the multi-rate terminal were investigated and preliminary designs developed. A solution to maintaining RED/BLACK integrity for a common processor-based equipment was developed which allows the voice processing and modem receive functions to be accomplished by a single processor located in the RED area of the equipment.

Since this program was primarily an architecture study the clear mode operation of the terminal was not specifically addressed in this report. However, the terminal will provide a clear mode capability where by the processing circuits can be bypassed for non-secure calls thus permitting direct interface of the user's telephone with the Autovon or DDD network.

The results of this study show that the program objectives are achievable. The selected architecture solves the RED/BLACK problem. The cost objective of \$5000 can be met. With current technology and minimum use of LSI the unit cost to the government is \$6,160. With more extensive LSI, plus design simplification, the unit cost goal of \$5,000 can be realized. Costing is based on FY 80 dollars for quantities of 10,000 commercial quality, TEMPEST qualified terminals. The results of this study along with the costing data for the multi-rate terminal are contained in this report.

As the next logical step towards implementation of the low cost terminal, it is recommended that this study effort be continued with breadboard evaluation and demonstration of the concepts derived from the study. This would prove the concepts, both technically and economically, and should provide optimization of cost and performance for the Multi-Rate Secure Voice Terminal.

The multi-rate terminal concept resulting from this study provides the voice processing, encryption/decryption, key distribution, modem and line interface functions for secure voice communications at 2.4 KB/S, 9.6 KB/S and 16.0 KB/S over standard dial-up telephone facilities. All major functions except the key generators (KG) and KG alarms are microprocessor implemented and controlled. The key generators assumed are the Saville type, with two KGs being utilized for transmit and one for receive. As detailed further in this report the design guidelines for the multi-rate terminal were structured to comply with the following:

- Interoperable with existing and planned future systems.
- Based on present and near term new technology (no breakthroughs required).
- Achievable on a reasonable schedule.
- Satisfies TEMPEST and COMSEC control and interface.
- Producing in quantity at low cost.

Some of the more significant performance features of the multi-rate terminal include:

- Multiple Bit Rates
 - LPC-10 at 2.4 KB/S
 - APC-4 at 9.4 KB/S
 - CVSD at 16.0 KB/S

- Half or full duplex operation
 - 2 wire, half duplex
 - 4 wire, full duplex
- Interchangeable (or external) COMSEC module
- Key Distribution Center (KDC) Compatibility
- Key Schemes include
 - KDC
 - NET
 - Dedicated
 - Data
- Call Setup may be
 - Initially clear, or
 - Initially secure
- Size and Weight
 - 19 inches wide X 22 inches deep X 7 inches high
 - Approximately 40 lbs.

3.0 TECHNICAL DISCUSSION

3.1 Terminal Architecture

3.1.1 Architectural Concepts

The main technical concept being addressed in the multi-rate processor study is that of combining modem processing and voice processing together into the same processor. This concept is proposed as a means to attain cost savings in a secure voice terminal over other conventional approaches in which modem and voice processing functions are implemented separately.

Two questions present themselves with regard to this concept:

1. Can the computational aspects of combining the two functions together into the same processor be solved?
2. Can the security aspects relative to such a combination be solved?

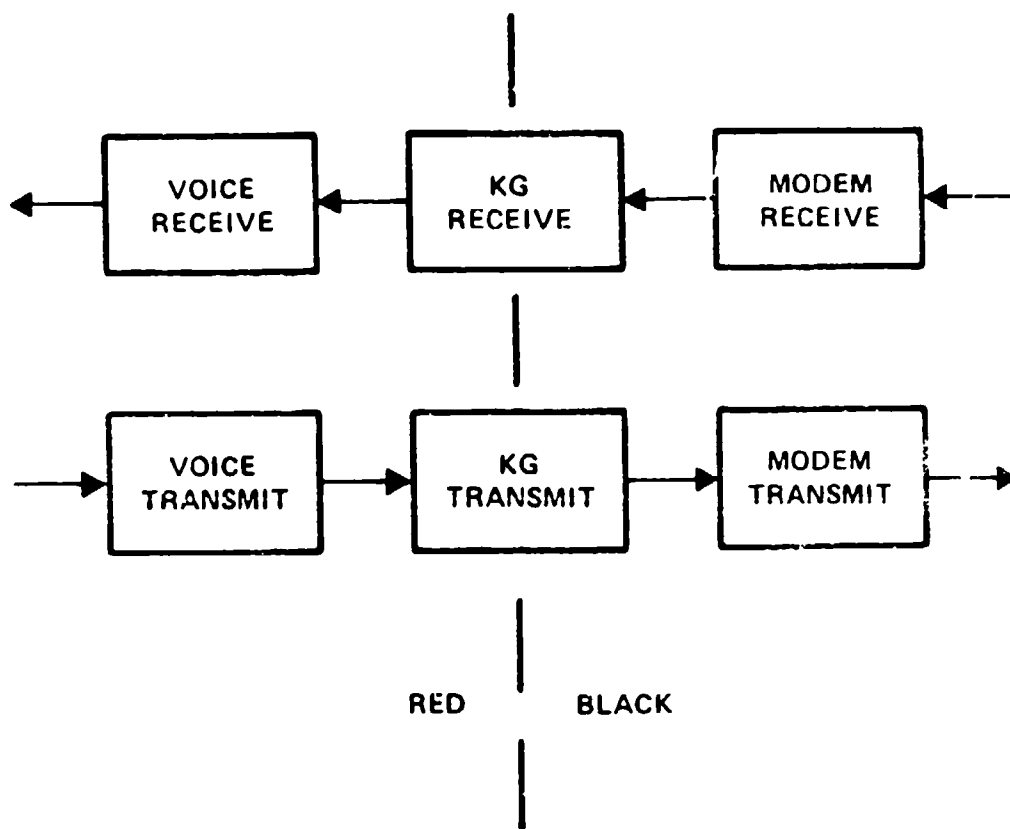
The answer to both these questions is yes, based upon the insight gained during this study. The computational aspects of a multi-tasked processor are not significantly different from those encountered in conventional real-time multi-tasked minicomputers. Thus the solutions possible from a computational aspect are as varied as the many approaches used in present day minicomputers.

The security aspects implied by the combination of a conventionally BLACK function, the modem, and a conventionally RED function, voice processing, are not as straightforward as the computational aspects and, therefore, must be handled in a more

delicate fashion. A block diagram of conventional RED/BLACK partitioning is shown in Figure 3.1.1-1. Transmit and receive functions are shown separately since they are essentially independent operations. In this conventional partitioning, the RED/BLACK dividing line is within the KG's for both transmit and receive. The transmit side is subject to much more stringent security constraints and requirements than the receive side, since the transmit side conveys information to the outside world whereas the receive side conveys information to a secure environment. This leads one to the conclusion that the modem receive function can be migrated from the BLACK to the RED side of the partition with minimal or no impact upon security aspects of the terminal. This is what is done in the modified RED/BLACK partitioning shown in Figure 3.1.1-2, where the modem receive function is migrated into the RED area. The modem transmit function is left on the BLACK side primarily to provide security protection for the transmitted signal. It is felt that if the modem transmit function were combined in a processor with voice (i.e., RED) processing, then a security failure analysis of the system would be of improbable success. Fortunately the modem transmit function is the simplest of the four processing functions under consideration; thus, the cost impact of a separate processor for this function can be minimized by proper sizing of the processor.

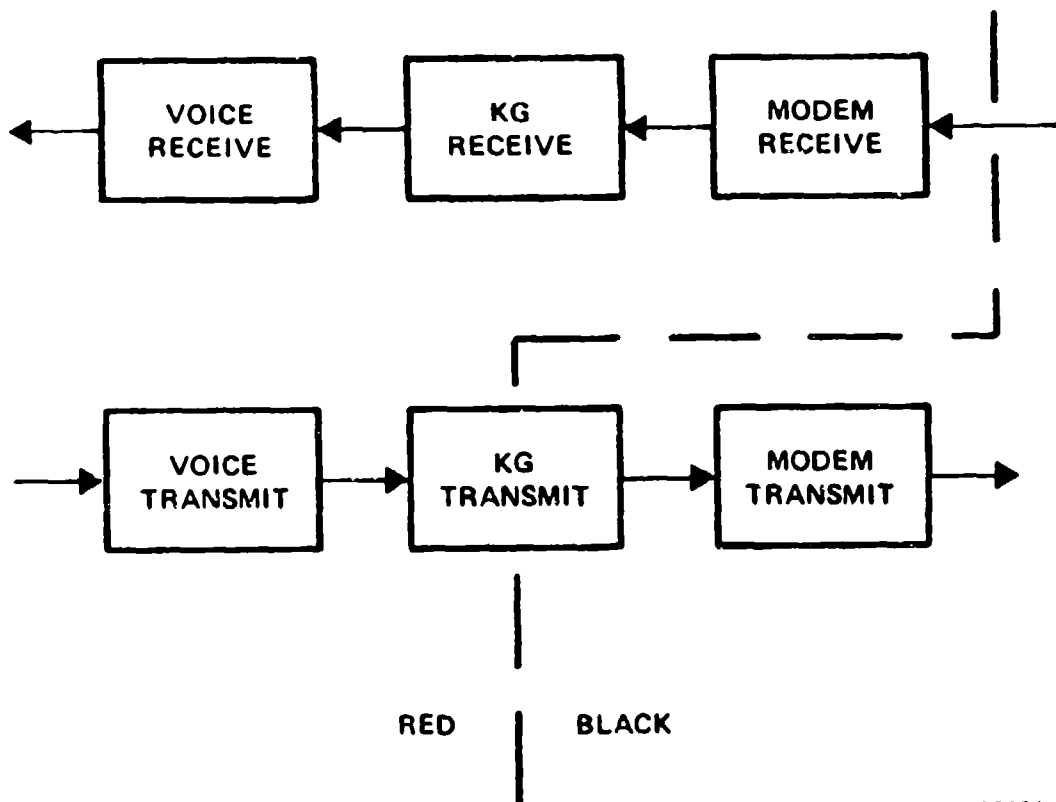
3.1.2 Architectural Design

Based on the RED/BLACK partitioning presented above, an architectural design was developed in the study that has sufficient power and flexibility to satisfy the requirements of the multi-rate secure voice terminal. A major level block diagram of the terminal



A90424-1

Figure 3.1.1-1. Conventional RED/BLACK Partitioning



A90424-2

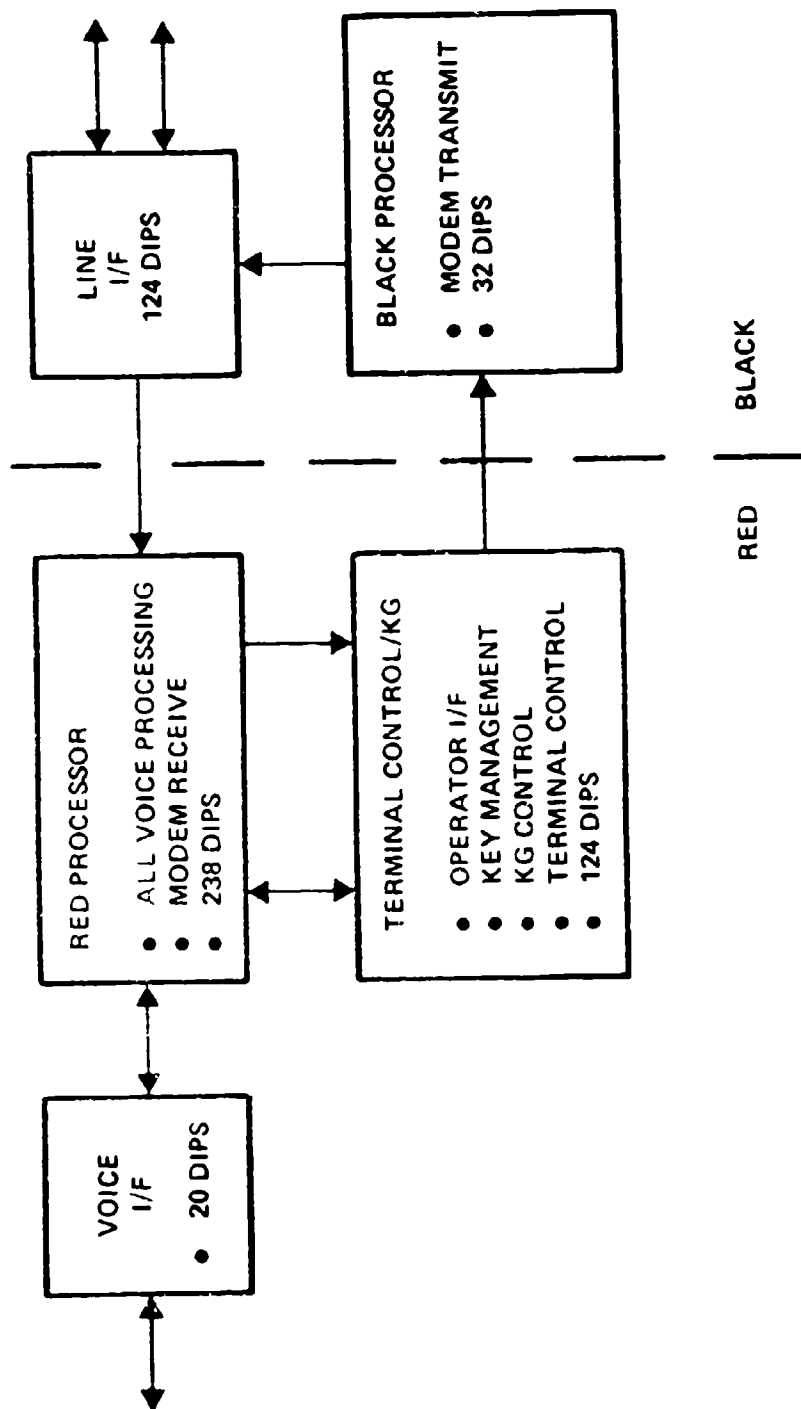
Figure 3.1.1-2. Modified RED/BLACK Partitioning

architectural design is given in Figure 3.1.2-1. As shown in the figure, this design incorporates three processors:

1. RED Processor
2. BLACK Processor
3. Terminal Controller

Together the RED processor and the BLACK processor implement all voice and modem processing. The RED processor implements all voice processing as well as modem receive processing. Thus it is responsible for the bulk of the signal processing accomplished within the terminal and is the largest hardware component in the terminal with an estimated DIP count of 238 chips. The processor includes a hardware equalizer as an outboard component in order to accommodate the 128 tap adaptive equalizer required for the 16 kilobit modem receive function. The processor itself is a 16 bit bit-slice microprocessor developed at Harris GCSD specifically for digital signal processing applications. Further details relating to this processor and the other major functional components within the architecture are presented in subsequent sections to this brief overall description.

The BLACK processor implements the modem transmit function. As discussed previously, this function (modem transmit) is implemented in separate hardware within the BLACK area of the terminal in order to satisfy security protection of the transmitted signal. The cost impact of this action is relatively small as noted by the small dip count (32 dips) estimated for this function. The BLACK processor is implemented with a minimal configuration 8 bit 2903 processor.



A90424-3

Figure 3.1.2-1. Terminal Architecture

The terminal controller and related subsystems are responsible for two main functions.

1. Top level control of all subsystems
2. Encryption/Decryption

The two functions are treated together because of the intimate relationship that must be established between the KG devices and the control of those devices to satisfy security requirements for the terminal. The assumption taken here is that KG components rather than a complete KG are provided GFE for the terminal development. This assumption complicates the design, of course, since alarm, gating and key storage logic must be included, that would normally be furnished in a complete KG.

In addition to KG control, the terminal controller provides the interface to the operator of the terminal. Through this interface call placement, mode selection, BIT and BITE, and other terminal functions are exercised. For control purposes the RED processor and BLACK processor operate as slaves to the terminal controller. Control is exercised by a status and control register interface between processors to communicate simple commands such as start training, start LPC, etc. The terminal controller is implemented with a conventional MOS microprocessor such as the INTEL 8080.

Two interfaces are included in the terminal architecture as depicted in Figure 3.1.2-1. These interfaces connect the terminal to the telephone handset or other voice entry device and to the telephone line(s). The line interface is expected to be the more complex of the two because of isolation, level matching, and switching requirements placed on it.

The RED processor is the major computational component within the multi-rate terminal. Its design is the key to the success of the terminal since it implements the major technical challenge addressed by the study, namely implementing voice and modem algorithms together in the same processor. The RED processor is responsible for executing the following pairs of algorithms:

- a. LPC - 10
2.4 kilobit modem receive
- b. APC - 4
9.6 kilobit modem receive
- c. CVSD
16 kilobit modem receive

Each of these algorithms requires computationally intensive number crunching, as well as fairly complex decision logic. The processing power required, both in terms of instructions per second and the type of instructions needed, is far in excess of that provided by present day mini-computers and microprocessors. This fact has been responsible for the development of many special purpose signal processors dedicated to either speech signal processing or modem signal processing. Fortunately the computational characteristics of signal processing for speech and for modems are similar in many respects, so the combination into one processor is computationally feasible. The biggest problem associated with the implementation of either speech or modem computation in a programmable processor is the adaptive equalizer required for the modem, particularly at the

16 kilobit transmission rate. The 16 kilobit equalizer requires approximately 3 million multiplies per second, a rate that is approximately the capability of large scientific main frames. The use of a hardwired equalizer outboard from the processor in the 16 kilobit Maxi modem, was necessitated by this extremely high computation rate.

During this study an effort was made to discover a processor structure that was capable of implementing the equalizer in software along with the remaining code. Unfortunately, no such structure has been found to date that is economically attractive in terms of dip count. An outboard equalizer, similar in design to that used in the 16 kilobit Maxi modem, has thus been selected for use in the multi-rate terminal. A description of the proposed equalizer is included in a later paragraph.

With the equalizer implemented separately from the processor proper, the computational load on the RED processor is more equally distributed among the three operating rates than would otherwise be the case. The processor thus assumes a characteristic very similar to the 2901 bit slice processor used in the 16 kilobit modem. Regretably the Maxi processor is not capable, in its present configuration, of executing LPC satisfactorily, primarily because it is a 12 bit machine while LPC implementation requires a 16 bit machine as a minimum.

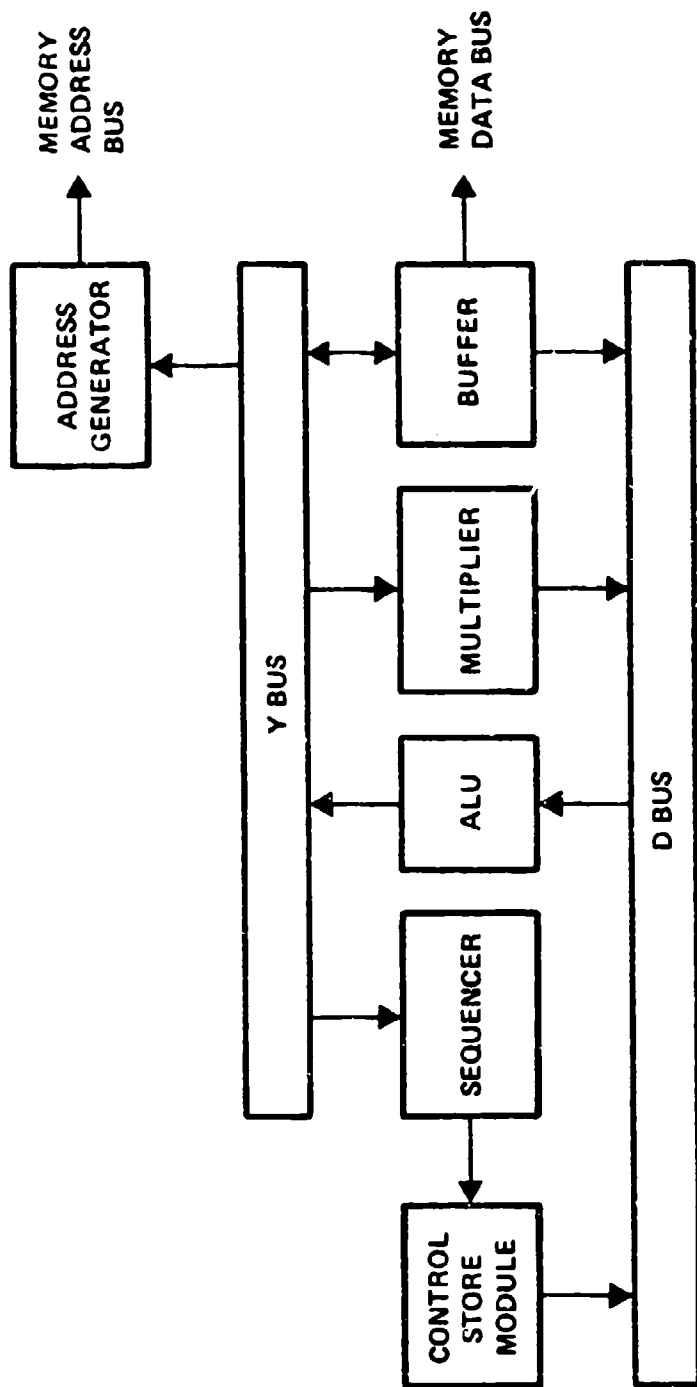
A processor similar to the 16 kilobit processor, that is eminently suited to this application, has been developed at Harris GCSD on IR&D. This processor, a 16 bit microprogrammed bit slice machine, was designed specifically for digital signal processing tasks, such as the speech and modem processing tasks of concern here.

In point of fact, LPC speech compression is under IR&D development at the present time, with very encouraging loading estimates. This special digital signal processor, called the Harris Micro Signal Processor (HMSP) is described in architectural detail in Appendix I to this report. A brief summary of the features and capabilities of the machine are included here, and the interested reader is referred to the appendix for further detail.

3.2.1 Processor Features and Capabilities

The processor proposed here for use in the multi-rate terminal is a 16 bit, microcoded, bit slice microprocessor designed specifically to implement multiply intensive digital signal processing algorithms. A block diagram of the processor is given in Figure 3.2.1-2 showing the major functional components. Significant features of this processor are:

- 16 bit fixed point arithmetic.
- 16 bit TRW LSI multiplier.
- Dual ALU's, one each for
 - Data Computations
 - Address Computations
- Effective Multiply - Add Time of 400 ns
- Microcoded - 64 bit Instructions.
 - 200 ns per Instruction.



A90424-4

Figure 3.2.1-2. RED Processor Block Diagram

- Parallelism allows multiple operations per instruction.
- Memory Fetch/Store
- Memory Address Calculation
- Multiply
- Data ALU Calculation
- Program Branch

In addition to the hardware features described here and in the appendix, extensive software support for the processor is worthy of mention. A very flexible and powerful cross assembler hosted by a PDP-11 is utilized for program development. This assembler, developed at Harris GCSD, is far superior to any known commercial product. It has allowed the development of "assembler like" language coding of the processor, wherein the programmer can concentrate on the operations being performed rather than on the bit field manipulations required to implement those operations.

An extensive debug capability has also been developed for the processor using an 8080 MOS microprocessor to control the 2901 processor. Significant aspects of this debug package are:

- Single Step Mode
- Trace Mode
- Multiple Breakpoints with Break and Continue Option
- Full State Display Under User Control

- State Modification
- Control Store Examination/Modification with Extensive Bit Field Capability
- Data Memory Examination/Modification

3.2.2 Processor Loading and Memory Estimates

Processor load estimation was performed in two separate parts: voice and modem. Estimation of Voice loading was performed by Frank Newdeck of Ketron, Inc., an independent consultant. A report of this activity including details of the estimation procedure, and comments relative to the processor structure, are included as Appendix II to this report. Voice processing loading is summarized in Table 3.2.2-1. Please note that only one of the rates is active at any time, so that each of the three loading rates can be considered independently. The loading figures listed in Table 3.2.2-1 are significant in that they represent a two-to-one speed increase over that attained by current processors.

Table 3.2.2-1 Voice Processing Loading

Rate	Loading (%)
2.4 LPC	29.2 %
9.6 APC	27.0 %
16.0 CVSD	14.2 %

Estimation of the processor load due to modem receive was performed by extrapolation of actual loading figures for the 16 kilobit Maxi modem. Those actuals are listed in Table 3.2.2-2 along with estimates of processor loading of the proposed RED processor at each of the three rates. Three of the software functions, the executive, common receive functions, and post processing, are projected to be invariant to bit rate, and also to execute at approximately the same speed on the RED processor as the Maxi processor. The low pass filter function can be implemented with a two cycle inner loop on the RED processor as opposed to a four cycle inner loop on the Maxi processor. Accounting for overhead decreases the execution rate from 100 microseconds for Maxi to 60 microseconds for the RED processor in 16 kilobit mode. The filter is the only major difference between the RED processor and Maxi at the 1 kilobit rate. The resultant loading is estimated at 170 microseconds out of 375 microseconds available per symbol, or a percentage loading of 45 percent.

Table 3.2.2-2. Modem Loading Estimates

Function	Maxi 16 Kilobit (Actual)	Multi-Rate Terminal		
		16 kb	9.6 kb	2.4 kb
Executive (microseconds)	10	10	10	10
Common Receive Functions (microseconds)	43	43	43	43
Low Pass Filter (microseconds)	100	60	60	110
Post Processing (microseconds)	57	57	57	57
Symbol Timing (microseconds)	0	0	40	40
TOTAL (microseconds)	210	170	210	260
Time Available (microseconds)		375	416	833
Loading (%)		45%	50%	31%

The 9.6 kilobit rate is projected to be identical in execution time to the 16 kilobit rate with the addition of an estimated 40 microseconds per symbol for a symbol timing loop. Symbol timing is discussed in another paragraph. The 40 microseconds budget for symbol timing should allow for the execution of even a very elegant symbol timing structure. The resultant loading is estimated at 210 microseconds out of 416 microseconds available per symbol, or a percentage loading of 50 percent.

The 2.4 kilobit modem operates at half the symbol rate of the 9.6 kilobit modem (1200 baud versus 2400 baud). Otherwise its operation can be construed as being similar to the 9.6 kilobit modem. One would expect that the 2.4 kilobit modem can be structured so that it is a subset of the 9.6 kilobit modem within the software of the multi-rate terminal. The execution time per symbol of the 2.4 kilobit modem is projected slightly higher than the 9.6 kilobit for the low pass filter function because its time span is larger. The resultant loading is estimated at 260 microseconds out of 833 microseconds available per symbol or a percentage loading of 31 percent.

The estimation methodology used here for modem loading is quite reliable but conservative. The methodology is based upon performing modem processing on a symbol by symbol basis. It would be desirable at the 9.6 and 2.4 kilobit rates to perform the modem processing on a frame basis wherein multiple symbols are processed at one time. The frame orientation is desirable at these rates in order to obtain compatibility with the frame oriented computation of LPC and APC voice compression. Computation on a frame basis will also decrease the execution time of the modem code, potentially by a significant amount, because the control portion of the code is generally executed once per frame rather than every symbol. Frame based structure for the modem software will be investigated more fully in the follow-on to this study.

Total processor loading estimates are presented in Table 3.2.2-3 for each of the three rates. Note that the worst case loading is 77 percent, thus leaving substantial growth capacity in the processor for later additions or modifications to the voice or modem software.

Table 3.2.2-3. RED Processor Loading Estimates

Rate	Voice Loading (%)	Modem Loading (%)	Total Loading (%)
2.4 kb	29.2	31	60
9.6 kb	27.0	50	77
16 kb	14.2	45	59

The memory required for the RED processor consists of PROM for program storage (Control store in this processor), PROM for data storage, and RAM for temporary data storage. Estimates for the required memory capacity in the RED processor are presented in Table 3.2.2-4. Listed in the table are the memory estimates for each of the six algorithms individually as well as the total required capacity. An additional line lists the memory capacity used for costing of the terminal as reported later in the report.

The memory estimates listed in the table are in units of K-words (thousand words). Control store in the RED processor is 64 bits wide (8 bytes), while data store in the processor is 16 bits wide (2 bytes). Thus the memory sizes in K bytes used for costing are as follows: 48 K bytes for control store ROM; 20 K bytes for data store ROM; and 8 K bytes for data store RAM.

Table 3.2.2-4. RED Processor Memory Estimates

Function	Control Store ROM (K)	Data Store ROM (K)	Data Store RAM (K)
LPC-10	1.5	2.0	2.0
LPC-4	1.0	1.0	2.0
CVSD	0.3	0.2	0.1
16 kb	1.0	1.0	0.4
9.6 kb	1.1	1.0	0.5
2.4 kb	0.5	0.5	1.5
TOTAL	5.4	5.7	3.5*
Size For Costing	6.0	10.0	4.0

*NOTE: RAM total is max required per rate (16/9.6/2.4)

3.2.3 Adaptive Equalizer

The RED processor requires the use of an outboard automatic equalizer to augment the modem receive function when operating at the 9.6 KB/S rate. The purpose of automatic equalization is to counteract the effects of the transmission channel by forming the channels' basic inverse transfer function. Adapting the equalizer to the transmission line is accomplished during train time by minimizing the mean square error out of a linear transversal filter.

At the lower data rate of 2.4 KB/S the use of an adaptive equalizer is not necessary, but may be desirable. The higher data rate requires a long train time of approximately four seconds. At 9.6 KB/S the train time must be significantly shorter to interface with existing modems.

For the equalizer to function in both the 9.6 KB/S and 16 KB/S modes it must be functionally controllable by the main processor. An overhaul of the previous 16 KB/S equalizer design, with some additional hardware, could be made to handle the requirements of the Universal Terminal. However, microcode controlled processor based design, with parameter passing from the main processor, would not only make the equalizer flexible enough for both 9.6 K and 16 K mode, it would also reduce the chip count over the previous design. In addition, with a processor based equalizer completion of "fast" training in the 9.6 K mode could be accomplished more readily. A discussion of the equalizing technique is necessary before going into the approach for its implementation.

The equalizer functions like a transversal filter storing a number of consecutive data samples in what appears to be a long shift register. On a symbol basis new samples are shifted into, and old samples chopped from the end of the data memory. This data is then weighted and summed to give the estimate of the transmitted value. Thus, the equation for finding an estimate (E) of n data samples ($D_1, D_2, \dots, D_n$) is:

$$E = \sum_{i=1}^n W_i D_i$$

where $W_1, W_2, \dots, W_n$ represent the weights. When this complex equation is broken into its x and y component parts, the estimates are:

$$\text{EQ 3.2.3-1} \quad E_x = \sum_{i=1}^{n/2} W_{in_i} D_{x_i} - \sum_{i=1}^{n/2} W_{cr_i} D_{y_i}$$

$$\text{EQ 3.2.3-2} \quad E_y = \sum_{i=1}^{n/2} W_{in_i} D_{y_i} - \sum_{i=1}^{n/2} W_{cr_i} D_{x_i}$$

Here the weights have been broken into in-channel (W_{in}) and cross-channel (W_{cr}) to distinguish their component parts.

The weights are adaptively set during the train sequence, where a reference is generated, which corresponds exactly to the transmitted sequence to measure the error in the equalizer's output estimate. This error is then unrelated with the stored data values to update the weights. Following is the weight update equation:

$$W_{(new)_i} = W_{(old)_i} + KD_i e$$

where e represents the error, and the k term is the loop gain factor which controls how quickly the weights adapt. The equation when changed from its complex form to its component parts looks like:

$$W_{in(new)_i} = W_{in(old)_i} + KD_{x_i} e_x - KD_{y_i} e_y$$

$$W_{cr(new)_i} = W_{cr(old)_i} + KD_{y_i} e_x + KD_{x_i} e_y$$

The 16 KB/S modem weight update equations executed in order once every symbol are:

$$\text{EQ 3.2.3-3} \quad W_{in(new)_i} = W_{in(old)_i} + e_x D_{x_i} D_{x_i} e_x$$

$$\text{EQ 3.2.3-4} \quad W_{cr(new)_i} = W_{cr(old)_i} + e_x D_{y_i} D_{y_i} e_x$$

$$\text{EQ 3.2.3-5} \quad W_{in(new)_i} = W_{in(old)_i} - e_y D_y D_{y_i} e_y$$

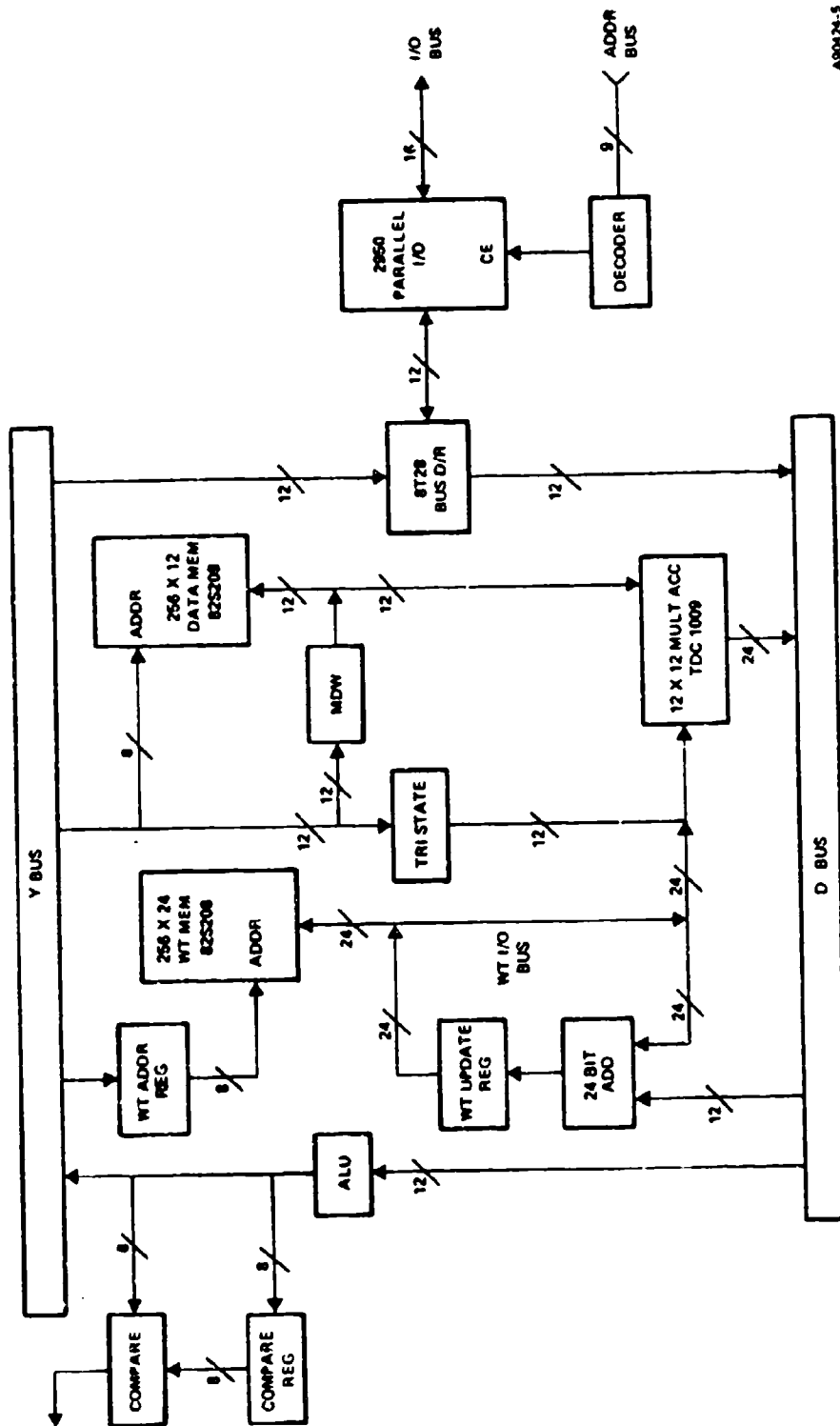
$$\text{EQ 3.2.3-6} \quad W_{cr(new)_i} = W_{cr(old)_i} + e_y D_x D_{x_i} e_y$$

In this way the weight update effectively occurs twice per symbol.

A microprogram controlled equalizer designed to perform equations 3.2.3-1 through 3.2.3-6 efficiently is shown in Figure 3.2.3-1. The two-cycle inner loop is sufficient for executing equation one and two. In every other cycle either a new data or a new weight is addressed by the ALU section which is comprised of three 2901 bit-slice processors. (Using the processors for address generation maximizes addressing capability therefore maximizing memory control.) While the data and weight memories are being stepped through, the multiplier accumulator holds the running sum of their products. A loop execution complete signal is generated by comparison of the weight address with the compare register loaded during set up of the loop. When all the active weights have been accessed, the estimate resides in the multiplier accumulator output register ready to be saved in the ALU section for transfer to the main processor.

The weight update equation can be implemented via a four cycle inner loop. The pipeline effect of the multiplier accumulator along with an external 24 bit adder are used to minimize the instructions in the loop. The ALU portion of the equalizer is again used to step through the weight and data memories.

The equalizer is slave to the main processor by parameter passing through the input/output port. This port is controlled by two AM2950's which contain on-chip flags and I/O holding registers. In addition, there are four spare control lines since the bus to the main processor is 16 bits wide whereas the equalizer bus is



AD0424-5

Figure 3.2.3-1. Adaptive Equalizer Block Diagram

just 12 bits. The only output necessary from the equalizer is the estimate for both x and y in either 9.6 K or 16 K mode. On the other hand, since 9.6 KB/S and 16 KB/S require the equalizer to behave differently, the input controls the section of data and weight memories to be active, brings in the new data with the associated address pointer for that symbol, plus transfers the x and y error information. An estimate of the number of I/O executions necessary is 21 and since the equalizer is twice as fast as the main processor, 40 instruction cycles would handle the I/O requirements.

The control section of the equalizer is composed of a single presetable counter with a status mux for decision making and some program PROM. The 2 K (256 X 8) PROMs with an output register are used for a microword of 40 bits. An external reset is available for initialization. The 256 instruction capability is plenty for handling the equalizer programs which are comprised mainly of two 2-cycle loops and four 4-cycle loops plus a small amount of I/O.

The heaviest use of the equalizer is in 16 K mode when all 242 of its weights are active and the symbol time is the shortest. All six equations with $n = 242$ must be completed in a symbol time of 375 ns (2/2/3 kHz). A total of 2904 cycles are necessary to execute all six equations, using a two cycle inner loop for equations 3.2.3-1 and 3.2.3-2, and a four cycle inner loop for equations 3.2.3-3 through 3.2.3-6. With a 100 ns execution time, 290.4 microseconds are used by the six equations or about 77.4% loading which can be compared to 95% loading of the previous 16 KB/S equalizer. If the 40 additional steps for I/O are added on, the total equalizer loading comes to about 78% loading leaving over 800 execution times for any miscellaneous.

The best use of the power of this equalizer when the modem is in the 9.6 and 2.4 mode requires further study. However, the outcome of this study should not affect recurring cost estimates.

A discussion of modem symbol timing recovery is included here because the 16 kb modem does not presently perform symbol timing, whereas the 2.4 kb and 9.6 kb modems do. Thus this function needs to be integrated in the Maxi design as altered by this study to perform in the multi-rate terminal environment.

In both 2.4 KB/S and 9.6 KB/S modes symbol synchronization is required to interface with existing modems since frequency drift occurs between the transmit and receive clock circuitry. However, when the terminal performs as a 16 KB/S modem a precision oscillator will be used to eliminate the need for recovering timing from the received signal.

A block diagram of a typical analog circuit (Figure 3.2.4-1), designed for timing acquisition, was used for the cost analysis of the multi-rate terminal. Since this circuit contains most of the components essential to any analog symbol timing loop, it should provide adequate basis for the costing effort. Ideally, portions of the timing recovery will be performed by the RED processor to eliminate some hardware therefore reducing the cost.

An alternate solution, which could simplify receiver timing, involves using equalizer weights to detect timing drift. In this method the weights of the equalizer are monitored to detect lateral motion due to symbol timing drift. A VCO, controlling the receiver, will be adjusted periodically to keep the more critical weights centered about an established position. The advantage of this design is letting the equalizer counteract any small changes in symbol timing while the VCO need only control the more significant errors.

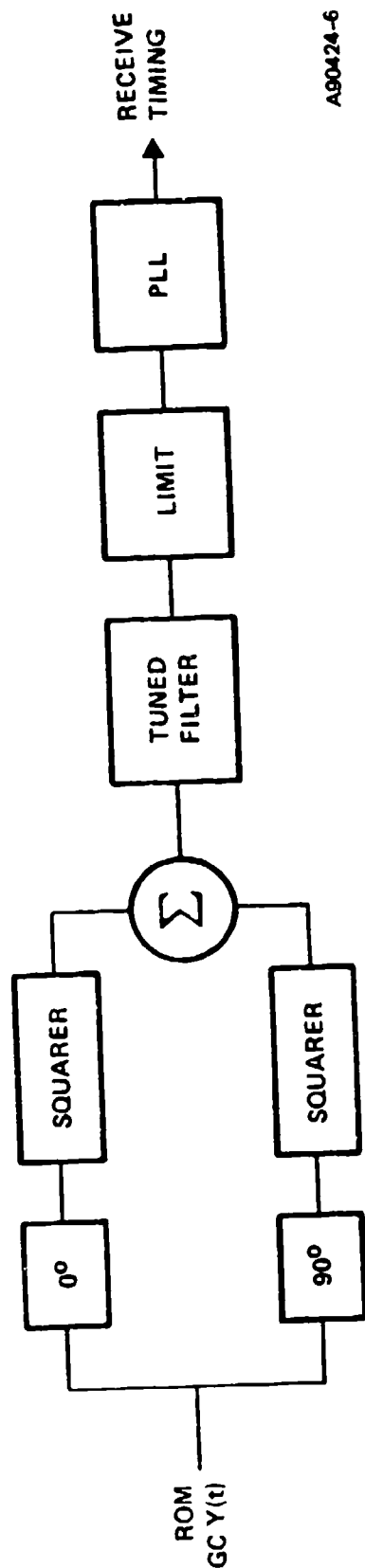


Figure 3.2.4-1. Tentative Symbol Timing Recovery Circuit

However, the problem of the weights ability to adapt fast enough to symbol drift in the 2.4 and 9.6 mode requires further study. The cost of this approach, if it is found suitable should be less than that of the analog approach used as a basis for the costing exercise.

3.3 BLACK Processor

A second processor has been assumed for the Multi-Rate Terminal to alleviate some of the RED/BLACK problems. It will be referred to as the BLACK processor since it has no security restrictions. Its purpose will be to perform the transmit functions which include sending the training sequence and data for the 2.4 KB/S, 9.6 KB/S and 16 KB/S modems. Serial data, received from the main processor through the RED/BLACK security barrier, will be processed and sent to a D/A for transmission over a telephone line. The serial bit stream will be grouped into symbols, grey coded, scrambled, and then sent through a polar to rectangular conversion. The transmit processor will then low pass filter and modulate the data before outputting to the D/A.

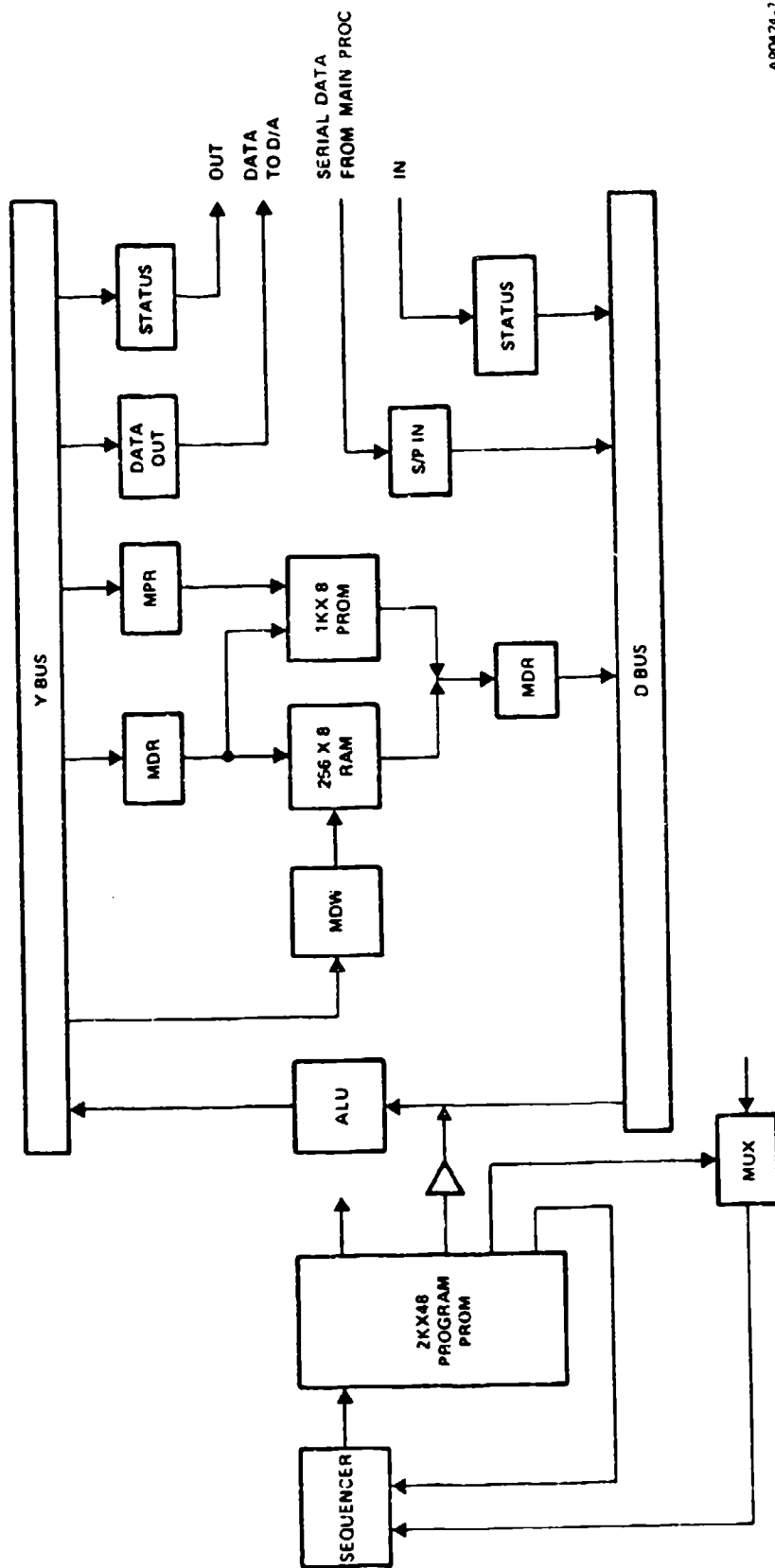
The decision had to be made between the use of either a microprocessor or bipolar bit-slice processor design. One of the important criteria was determining whether a hardware multiplier would be required for implementing the digital filter, which produces the heaviest load on the transmitter. A single microprocessor could not handle the 16 KB/S, low pass, 31 tap, transmit filter, which required 72 multiplications per symbol. With the $2 \frac{2}{3}$ kHz (375 ns) symbol rate for 16 KB/S a 5.2 microsecond multiplication would be required for performing the filter alone. On the other hand, the bipolar bit-slice processor is easily capable of software multiplications for faster than 5.2 microseconds. For this reason and to ensure adequate processing speed for other potential requirements the bit-slice processor was chosen over the microprocessor for the ALU section of the transmitter.

Transmit software from an existing 16 KB/S modem implemented on a bit-slice processor was used to estimate the worse case processing time with a software multiply for filtering. The data rates of 9.6 KB/S and 2.4 KB/S allow more time to perform the same basic transmit functions, therefore they are not as demanding on the BLACK processor. Unlike 16 KB/S Maxi processor where the processor restart occurs every symbol period, this transmit processor will cycle once per sample time to simplify input/output buffering. The upper limit on processing time will be the sample period of 9.6 kHz (104 microseconds) for 2.4 and 9.6 KB/S and $10 \frac{2}{3}$ kHz (93 microseconds) for 16 KB/S.

Estimating the 16 KB/S transmitter at 360 cycle times per output sample with 200 nanoseconds per cycle brings the processing time to 72 microseconds. This leaves a margin of about 20 microseconds per sample period for 16 KB/S or a 77 percent load on the BLACK processor.

The transmit processor (Figure 3.3-1) will be centered around an 8 bit ALU constructed from two bipolar bit-slice 2903s. For the memory, a 256 X 8 RAM will be sufficient to temporarily store processor status, data and flags as demonstrated by the existing 16 KB/S transmitter. However, PROM size for the processor must be larger than required for the 16 KB/S model since most coefficients cannot be shared among the three separate modes of operation. Thus, a 1 K X 8 PROM will be necessary to handle all the tables and coefficients for the transmitter. This PROM has to be divided into four pages accessed by an 8 bit memory address register (MAR) and a 2 bit page register (MPR) in order to comply with the 8 bit data path.

Data from the RED area of the terminal will be converted from serial to parallel at the input port of the BLACK processor. The processing of this data will proceed according to the information obtained at the status input port. This information will



A90474-7

Figure 3.3-1. BLACK Processor Block Diagram

control a 2910 sequencer which steps through the programs stored in 2 K X 48 PROM. The 48 bit microword, addressed by the sequencer, divided into sections each of which will regulate a specified portion of the BLACK processor.

Two buses focused on the ALU handle the data flow of the transmit processor. Inputs to the ALU are received from either memory, the serial to parallel converter, the status port, or directly from the program PROM via the D-bus. Outputs from the ALU pass over the Y-bus to either the MAR, MPR, the memory data write register (MDW), the data output port or the output status register.

3.4 Terminal Controller

The Terminal Controller interfaces between the RED processor and the BLACK processor/ATD to provide all terminal control plus encryption and decryption of the processed digital speech signals. It also establishes the secure communications link based upon the existing key distribution scheme and thus permits direct terminal interface with existing telephone networks which utilize the Key Distribution Center (KDC).

Three KG's are included in the terminal, two in the transmitter and one in the receiver. For the cost analysis discussed in a later section the cost of the KG modules (material) were not included, however, all other costs associated with the terminal controller and key distribution scheme were included.

The RED/BLACK data partition lies between the Terminal Controller and BLACK Processor. All RED data must pass through the Terminal Controller and be encrypted prior to insertion into the BLACK Processor for link transmission. The RED/BLACK data partition

contains the proper TEMPEST design features necessary to meet the specified requirements. These include EMI filtering, power filtering and transmit data RED/BLACK isolation.

Specifically, the Terminal Controller is responsible for the following major functions:

1. Key Distribution Center (KDC) call setup.
 - KDC calls and data format control.
 - Called and calling party calls and format control data.
2. Crypto Management
 - Encryption/Decryption
 - Alarm and Alarm Checks
3. Automatic Telephone Dialing Control
4. User control and display interfacing

These functions are performed by two subsystems within the Terminal Controller, the Terminal Processing Subsystem and the Crypto Management Subsystem. The crypto functions are under direct control of the COMSEC alarm and alarm check hardware. The Terminal Processing Subsystem is a microprocessor controlled subsystem which performs all call setup procedures required prior to transmission of the secure voice data.

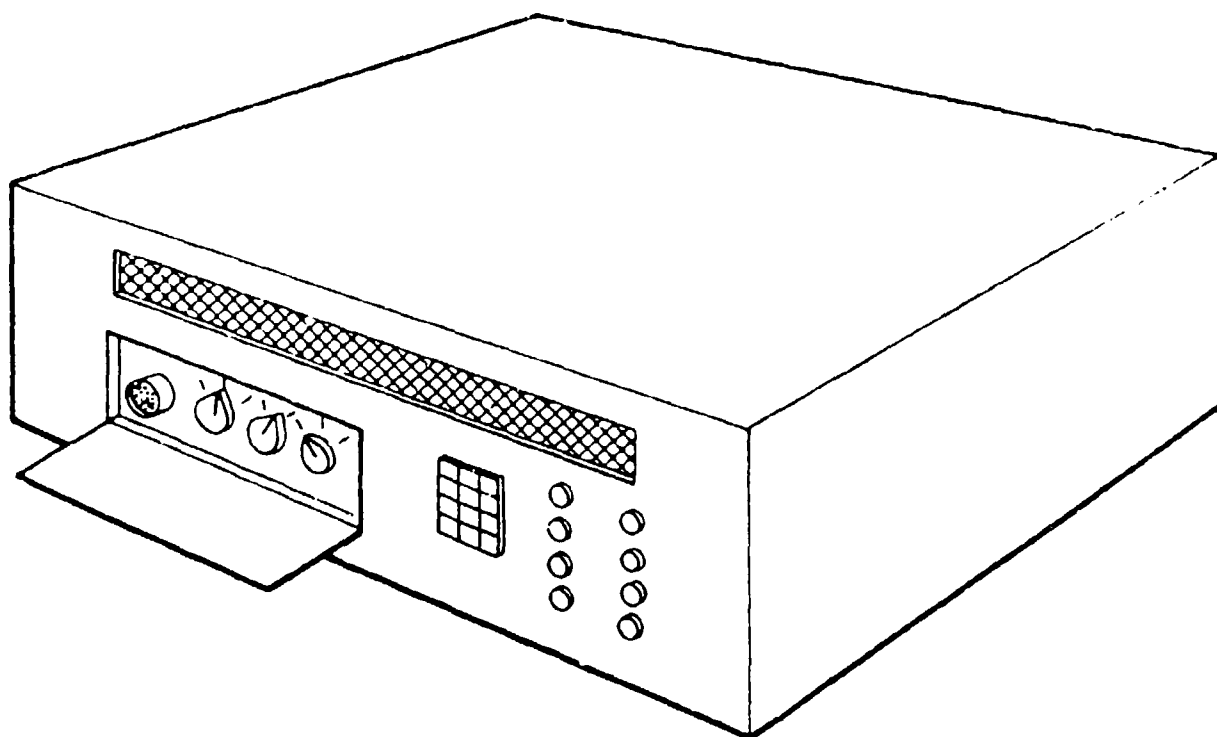
Technical details on the Terminal Controller are classified. A more comprehensive description of the Terminal Controller is contained in Volume II (classified) of this report.

3.5 Mechanical Packaging Concept

The mechanical packaging approach for the Multi-Rate Terminal is envisioned as an expansion of the basic low cost molded chassis utilized for the RADC 16 kbs Modem. This chassis is molded from high strength noryl and coated with a conductive material to provide an RFI-proof enclosure. For the Multi-Rate Terminal the enclosure would be separated into RED and BLACK compartments and RFI gasketing utilized around all panels in the RED areas to provide the RED/BLACK shielding necessary to meet the RFI and TEMPEST requirements.

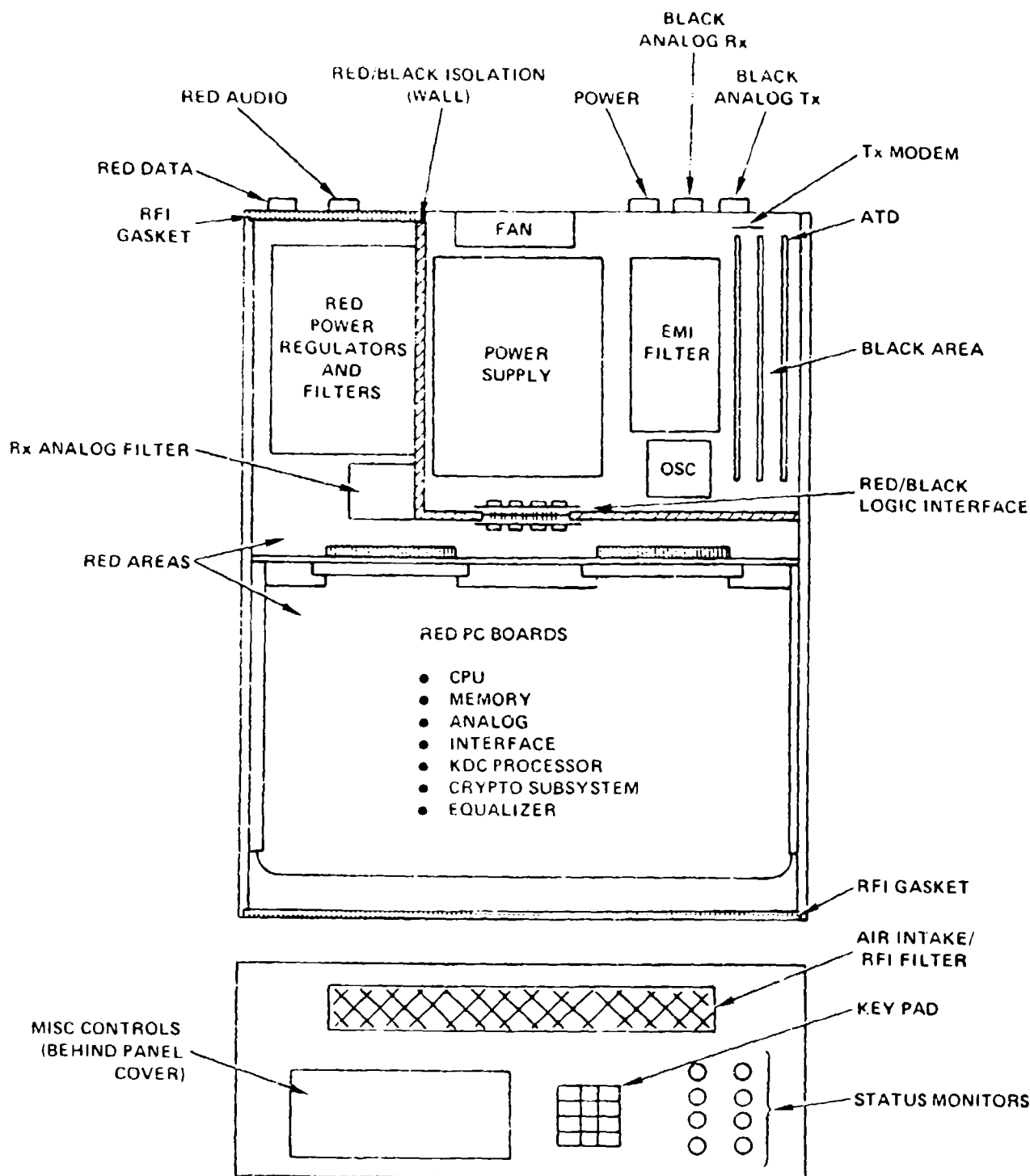
Based on the results of this study, the physical configuration of the Multi-Rate Terminal is shown in Figure 3.5-1. The unit is approximately 19 inches wide, 22 inches deep, and 7 inches high. The front panel is hinged to provide access to the RED logic boards. The front panel, when closed, seals tightly against an RFI gasket around the entire panel to provide a TEMPEST-proof enclosure. Some front panel controls may be located behind a cover plate on the panel to limit access and to provide improved RFI/Tempest isolation without going to special higher cost components.

As shown in Figure 3.5-2, the unit is compartmented into a RED area and a BLACK area. The RED area contains seven pc boards which include the logic for the RED processor, crypto and terminal control functions. This area also contains the power regulators, filters and interface circuitry necessary to provide the RED/BLACK isolation and signal interface between the sealed RED compartment and the outside.



A90424-8

Figure 3.5-1. Multi-rate Secure Terminal



A90424-9

Figure 3.5-2. Multi-rate Secure Terminal Physical Configuration

The pc boards in the RED area are approximately 10 inches by 15 inches in size and are accessible from the front of the unit. These are partitioned such that major circuit functions are accomplished on a single pc board thus reducing the number of interconnects required between boards and simplifying checkout of the overall circuit. The seven pc boards in the RED area incorporate the following major functions.

- CPU
- Memory
- Analog
- Interface
- Equalizer
- KDC Processor/Terminal Control
- Crypto Subsystem

All front panel controls are located in the RED area and interface with the RED terminal control logic. Logic and control signals that must flow between the RED and BLACK areas are handled digitally by the RED/BLACK interface logic mounted on the RED/BLACK compartment wall.

As indicated in Figure 3.5-2, the BLACK area contains the power supply, EMI filters and frequency standard oscillator for the terminal. Current estimates are that this area would also contain three pluggable pc boards. Two of these would incorporate the BLACK

modem transmit processor and related circuits, and a third board would contain the Automatic Telephone Dialing (ATD) device which would provide automatic terminal interface to the telephone network. Signal input, output and power connectors are at the rear of the unit in either the RED or BLACK area as appropriate.

Cooling of the logic and power supply circuits would be provided by forced air circulation through the RED and BLACK areas by a fan mounted on the rear panel.

Further study is needed to fully define the optimum mechanical design for the Multi-Rate Terminal. However, based on the positive results to date utilizing the chassis for the Maxi Modem, we believe that this approach is a very cost effective mechanical package for the Multi-Rate Terminal. The problems associated with RED/BLACK isolation and TEMPEST offer the greatest challenge in the development of a low cost mechanical design. With the RED/BLACK separation of circuit functions, as described previously, plus proper physical and electrical shielding of the RED area by a conductive and RFI gasketed enclosure, the security requirements of the terminal can be fully realized with this low cost approach.

4.0 UNIT COST

4.1 Current Cost Data

The cost to the government for the Multi-Rate Terminal as defined in the preceding paragraphs, is estimated to be \$6,160 per unit in quantities of 10,000. This cost is based on current technology and minimum use of custom LSI. With more extensive use of LSI plus further design simplification and technology advancements, a unit cost in the order of \$5,000 should be achievable.

The cost model and basic design approach for the Multi-Rate Terminal is based on the RADC 16 KB/S modem. In order to cost the terminal, the circuit architecture was partitioned into major subassemblies with logic functions and related circuitry for each subassembly defined. Preliminary circuit designs were then developed sufficient to determine material and labor costs at the subassembly level.

The \$6,160 unit cost for the Multi-Rate Terminal was derived as follows:

1. Based on 10,000 Commercial, TEMPEST, approved units
2. Stated in FY 80 dollars
3. Cost model and design approach based on the RADC 16 KB/S Modem
4. Cost of KG LSI (material only) excluded

5. Material costs were derived by:
 - a. Price history comparisons
 - b. 16 KB/S modem commercial material data in quantities of 10,000 units
 - c. Current 16 KB/S modem quotes
 - d. Engineering and Material Management estimates
6. Labor costs for the Multi-Rate Terminal subassemblies were correlated to 16 KB/S modem and factored accordingly.

For this cost model the Saville KG LSI was not included in the material costs; however, the printed circuit board and labor associated with assembly of the KG LSI into the unit were included.

A computer printout of the parts list is included in Appendix III of this report. Table 4.1-1 is a comparison of the 16 kb Maxi Modem and the Multi-Rate Secure Terminal. This table shows the relative complexity of the two equipments, and indicates the material, labor and total unit costs for each in quantities of 10,000. The cost of material and labor for each major subassembly is listed in Table 4.1-2. The costs include G&A and fee and represent the total estimated unit cost to the government.

Table 4.1-1. Maxi/Multi-Rate Terminal Comparisons

	Maxi Modem	Multi-Rate Terminal
Size	19 inches W x 22 inches D x 5.25 inches H	19 inches W x 22 inches D x 7.0 inches H
Weight	30 lbs	40 lbs
No. PWBS	5	8 (equivalent)
No. Multilayer	2	3
PWB Size	10 x 15	10 x 15
Chassis	Low cost coated noryl	Low cost coated noryl
TEMPEST Partitions	No	Yes
LSI Devices	0	3
No. Components	787	1230
No. IC's	280	583
IC Technology	T ² L	T ² L
Parts Quality	Commercial Ceramic	Commercial Ceramic
Total Parts Cost (Price)	\$2150.80	\$4613.07
Cost/PWB	430.56	576.63
Cost/IC	7.67	7.91
Total Recurring Labor	<u>1464.16</u>	<u>1547.02</u>
Total Cost	\$3616.96	\$6160.09

FY 80 Dollars

Table 4.1-2. Multi-Rate Terminal Cost (\$)

Subassembly	Total Material	Total Labor	Grand Total
Analog	465.92	151.43	617.35
ATD	201.71	69.40	271.11
Chassis	339.60	146.03	485.63
CPU	771.09	239.03	1010.12
Crypto SS	226.54	70.68	297.22
Equalizer	544.08	184.33	728.41
Front Panel	116.37	41.60	157.97
Interface	336.94	106.47	443.41
KDC Proc	402.79	144.50	547.29
Memory	378.38	118.66	497.04
Modem TX	181.24	52.96	240.20
R/B IF-Blk	41.79	14.99	56.78
R/B IF-RED	38.45	14.45	52.90
Rear Panel	<u>568.47</u>	<u>186.49</u>	<u>754.66</u>
Total	4613.07	1547.02	6160.09

The Multi-Rate Terminal study and resulting cost data was based on the use of current technology and minimum use of custom LSI. With further design refinements together with anticipated technology advancements it is possible to achieve significant cost reductions over those presented in this report. Some of the most promising areas of future cost reduction are listed below. These are areas that would be addressed in the next phase of development for the Multi-Rate Secure Terminal.

a. Significant use of custom LSI

In the cost model resulting from this study only three custom LSI circuits were utilized. These were in the crypto alarm, memory and control interface circuitry where several hundred discrete components were replaced by just three LSI circuits. This resulted in considerable savings in material, labor and pc board costs. There are many circuit areas where the component count could be significantly reduced by use of additional custom LSI in the RED processor, terminal processing subsystem and the BLACK processor. For example, a conservative estimate is that the component and IC count could be cut in half by the use of more LSI. This would result in significant reduction in material and labor costs and should result in at least a 25 percent reduction in the unit production recurring cost. Further studies are necessary to accurately identify those areas where LSI could be best applied.

b. Use of Hi-Rel Plastic IC's

The integrated circuits included in the cost model are ceramic commercial grade parts. Cost savings can be achieved by using lower cost IC's such as high reliability plastic encapsulated

circuits. The availability and suitability of these parts for the Multi-Rate Terminal were not evaluated during the study. However, it is anticipated that appreciable cost reductions can be achieved through the use of lower cost components for future designs of the terminal.

c. Technology Advance

The Multi-Rate Terminal costs were based on current technology using available off-the-shelf components. Technology advances in integrated circuits, LSI and microprocessors are continually providing higher performance at lower costs per function. New and improved microprocessor chips are being introduced which provide higher processing speeds, increased computational capability and greater flexibility while consuming less power. Since the Multi-Rate Terminal is a microprocessor based equipment, improvement in microprocessor performance and reduced microprocessor costs will have a direct impact on reducing the cost of the terminal. By taking full advantage of technology advances during the development of the Multi-Rate Terminal it is anticipated that significant cost savings can be realized. This should result in an equipment design with fewer parts, increased performance and reduced power requirements.

d. Design Simplification

The objective of this study was to determine the feasibility of incorporating the functions of the speech processor and the modem into a common processor and thereby reduce the cost of the overall terminal by this shared function approach. This, essentially, involved combining the common elements of the Harris LPC speech processor and the Harris 16 KB/S modem to define the common processor based equipment. With additional study plus detailed design and

breadboarding, further circuit simplification can be achieved. This, together with the use of more LSI and the latest technology components, will result in a terminal design that is significantly lower in cost than the model defined in this report.

In conclusion it can be stated that this study met its technical objective: combining modem and voice processing together in the same processor. The cost objective of \$5,000 per terminal is very nearly met with an estimated cost of \$6,160 per terminal using current off-the-shelf components, and is projected to be met or exceeded by further design refinements and more extensive use of LSI.

It is recommended that this study effort be continued with a breadboard evaluation and demonstration of the concepts proposed in this study. Such a demonstration would prove these concepts both technically and economically and should lead to further optimization of cost and performance as the effort proceeds. In conjunction with the continuing effort, more detailed study will be performed on certain aspects of the terminal that were not completely solved during the present effort. These include:

- a. Symbol timing of the modem receive function
- b. Software structures to minimize coding effort and maintain flexibility and performance in the terminal
- c. Input/output structures for the processor that complement the software structures

All of these aspects have several feasible solutions; it is just a matter of developing the best solution.

APPENDIX I

HMSP ARCHITECTURAL DESCRIPTION

IRDP 4261-60

DAVID BELL

FEBRUARY 1, 1979

I. INTRODUCTION

The HMSP (Harris Micro Signal Processor) is a high speed, bipolar, bit slice microprocessor designed for implementation of digital signal processing algorithms in real time. It is capable of executing most algorithms (such as LPC speech compression) with input sampling rates on the order of 8 kHz and other algorithms with higher sampling rates.

HMSP is a 16 bit fixed point machine with an integral high speed multiplier. A block diagram of the organization of the machine is shown in Figure 1.

A central 2901 array forms the ALU and general register set for the processor. Two generic busses, the Y Bus and the D Bus are provided for data transfer within the machine. An address generator provides for high speed array addressing in parallel with the main ALU operation. The sequencer directs instruction fetch from the 64 bit wide control store, and a single level of pipelining after the control store provides for concurrent instruction fetch and execute. The data memory is organized around a simple two bus scheme. The address bus originates from the address generator. The bidirection memory data bus is formed by combining the Y Bus and D Bus within the processor.

The structure of the machine is such that four major parallel operations may be executed in a single instruction. They are:

- Memory Addressing
- Multiplier Operation

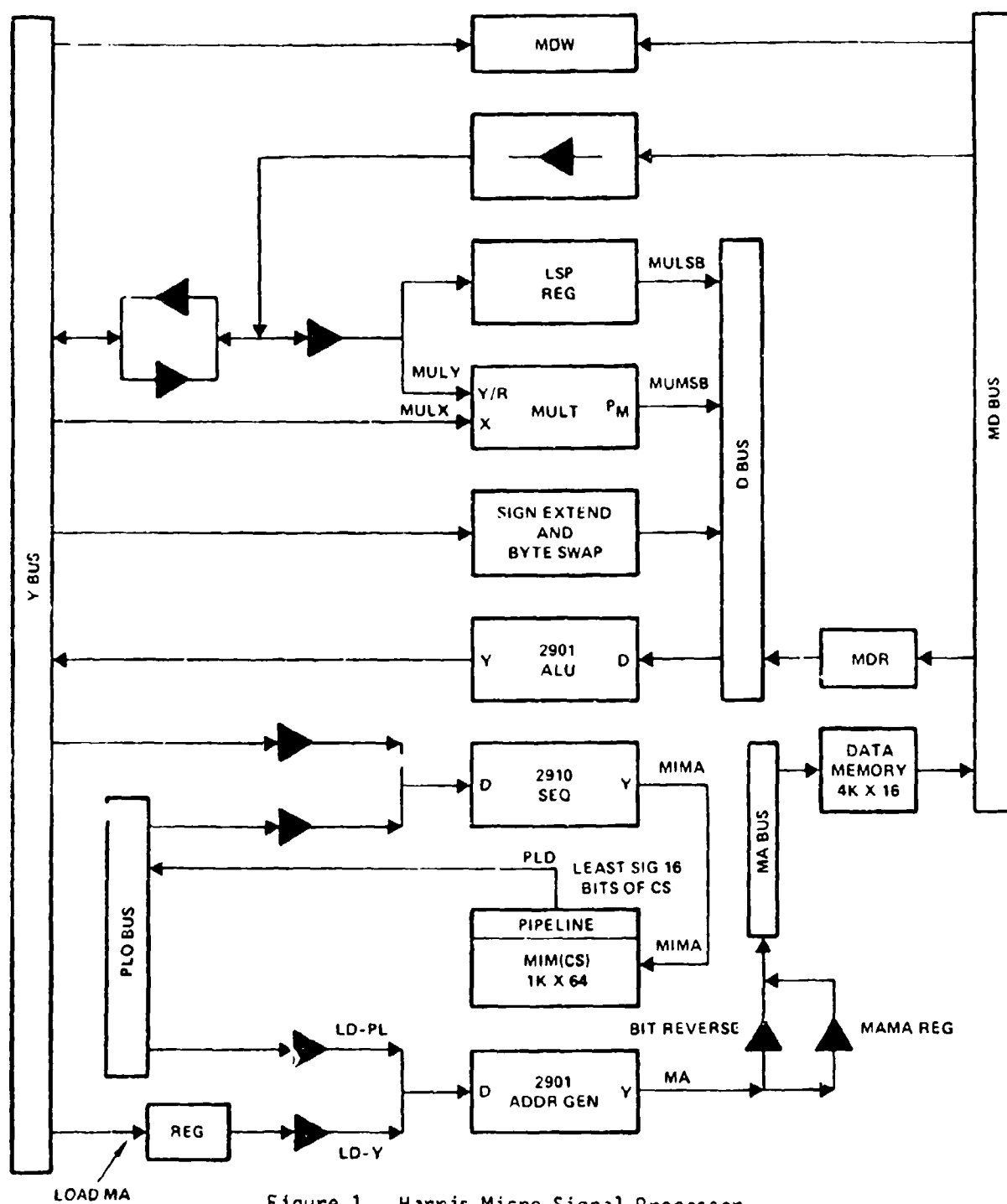


Figure 1. Harris Micro Signal Processor

A90424-11

- ALU Operation
- Sequencer Operation

This parallelism is intended to reduce the size of inner loops required in digital signal processing algorithms. For example, an autocorrelation calculation or FIR filter can be executed with a two cycle inner loop that exercises all four of the major operations listed on the previous page.

Supporting the HMSP is an integral 8080 MOS processor. The 8080 is used for downloading of programs into the HMSP and for debugging assistance while algorithms are in development. Within a target system application, the 8080 may be used for low speed decision and control related tasks.

Input/output for the HMSP is handled via memory mapping of I/O devices. A very simple interrupt scheme is included that requires polling of I/O devices by the CPU. The primary I/O method intended for the HMSP is direct memory access (DMA). A polite form of DMA is utilized which incurs no overhead. This is accomplished by performing DMA during memory cycles that are not used by the CPU. The address generator has a default NULL instruction that asserts a DMA grant line to I/O devices in the cycle before the DMA operation is performed.

II. ALU

The ALU logic organization is shown in Figure 2. Four 2901A elements are concatenated to form a 16 bit ALU. The connections between the 2901A's are as per AMD recommendations. The 2901's have 16 general purpose registers, an ALU and a one bit shifter. Two shift multiplexers on either side of the ALU generate the end conditions for

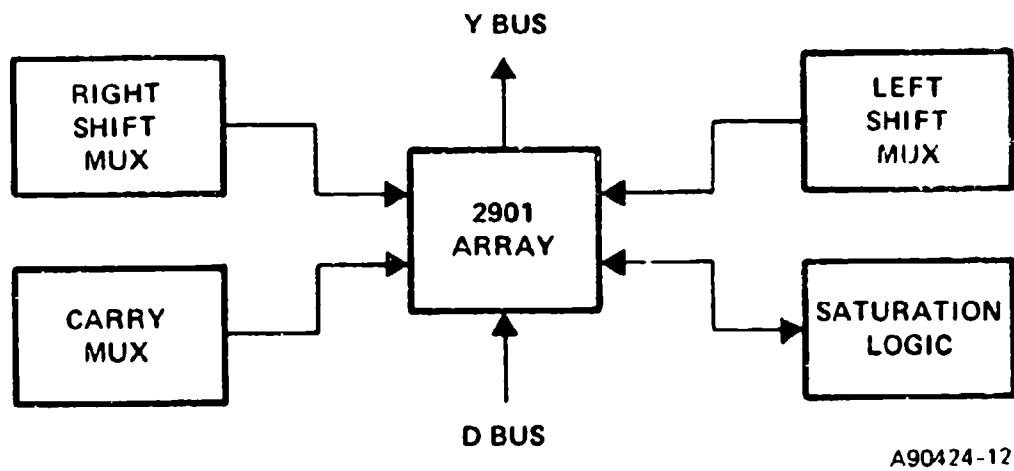


Figure 2. ALU Logic Organization

shift operations. Arithmetic, logical and rotational shifts are provided in both single and double precision. A carry multiplexer provides for zero, one and carry propagate from a previous instruction as carry inputs to the ALU. The saturation logic circuitry provides for hardware saturation or hard limiting of add/subtract operations within the ALU. This logic, under program control, will sense when an arithmetic overflow occurs and then jam the largest positive or negative number into the selected destination. Operation of the saturation logic is transparent to the programmer, other than enabling and disabling it, so that additions/subtractions with saturation can be performed in the same manner as those without.

III. D Bus

The D Bus is a tri-state multiplexed bus that furnishes data input to the ALU. There are seven possible input sources.

1. Immediate data from the pipeline register
2. The Multiplier output the most significant 16 bits of the product
3. The least significant 16 bits of the product from the Multiplier
4. A byte swapped version of the Y Bus Data
5. The Memory data read register
6. A word containing the sign extension of the last value of the D Bus
7. The memory address register

IV.

Y Bus

The Y Bus is a tri-state multiplexed bus that is primarily used to route data outputs from the ALU. There are four program selectable receivers on the Y Bus.

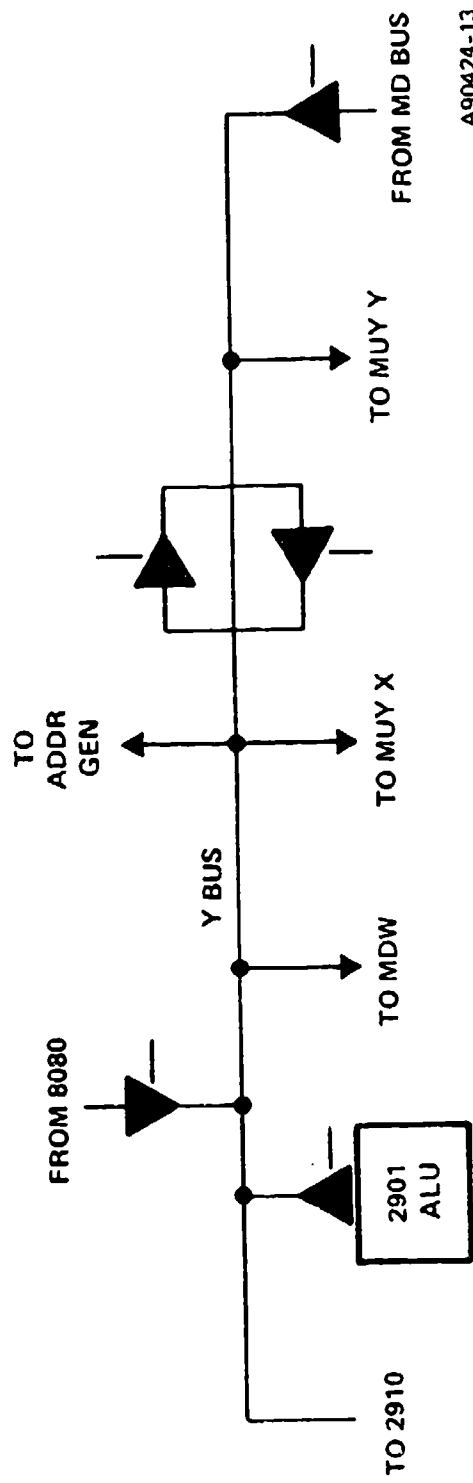
1. The Memory Data Write Register
2. The multiplier x input port
3. The multiplier y input port
4. The register/counter contained within the 2910 sequences

The microcode is organized such that the memory data write, the multiplier x, and the multiplier y registers can all be loaded simultaneously. In addition to the four selectable receivers on the Y Bus, the address generator receives data from the Y Bus under its own control.

In addition to routing ALU output data, the Y Bus is organized such that it can also route data directly from the memory or the 8080. A diagram of this organization is given in Figure 3. There are three selectable sources of data for the Y Bus.

- The 2901 ALU
- The Memory Data Bus
- An 8080 Debug Port

The 8080 Debug port is used to drive the Y Bus only when debugging the HMSP and is not used when the HMSP is running.



A90424-13

Figure 3. Y Bus Organization

The memory data bus is used to drive the Y Bus, so that the multiplier input ports can be loaded directly from the memory. This method of loading the multiplier is considerably faster than using the ALU to route data to the multiplier.

The remaining feature related to the Y Bus is a method whereby the bus is split into two separate busses. This is done so that both multiplier ports can be loaded simultaneously, one port from the ALU and the other port from the memory data bus.

V. Address Generator

The address generator (Figure 4) is designed to provide addressing of array type data in parallel with other operations in the machine. The address generator and Y Bus features of the processor account for at least a 2:1 speed increase over more conventional 2901 architectures when executing digital signal processing algorithms.

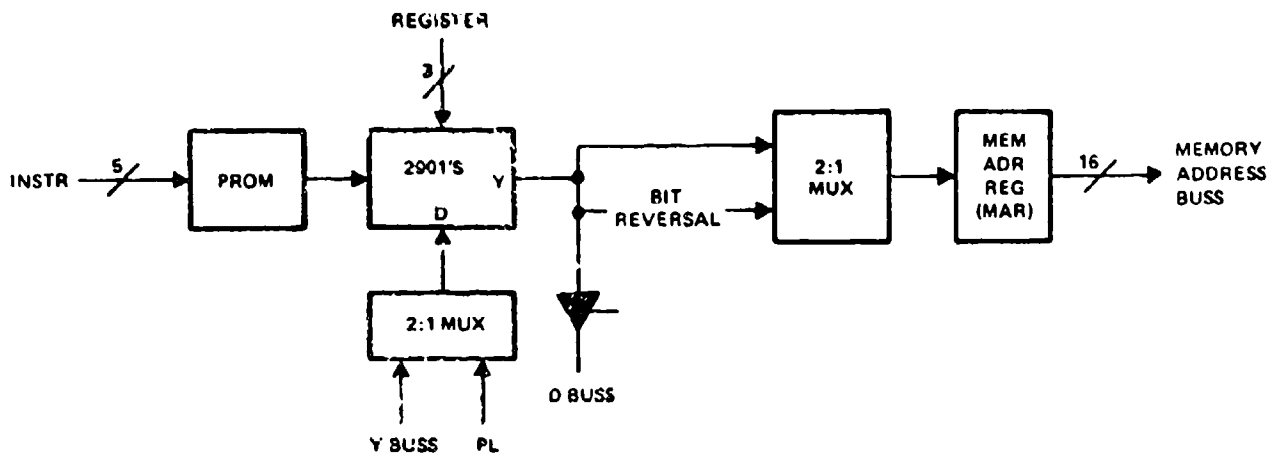
The address generator is built around a 2901 register/ALU set. It executes 32 different instructions under control of a PROM decoder. The PROM is used to limit the number of microcode bits required for programming. The sixteen registers contained within the 2901's are divided into eight address registers (A0 to A7) and eight index registers (I0 to I7). Register selection is under program control. The availability of eight address registers enables nearly all addresses to be computed within the address generator for array type processing. Data inputs to the generator come from two sources: the Y Bus or directly from the Pipeline register. The addresses computed by the 2901's during one cycle are loaded into the memory address register for use by the memory during the following cycle. A 2:1 multiplexer is used as input to the memory address register. It selects between using the 2901's output directly as computed or after bit reversal. Bit reversal is a simple flop between the least

significant and the most significant address bits. It is used in the computation of the FFT. The 2901's output is also routed to the D Bus primarily for examination of the registers during debug.

The address generator also permits the implementation of the polite DMA concept mentioned earlier. One bit out of the PROM decoder is routed out to I/O devices, and serves as a DMA grant line to the devices. This line is asserted for a NULL addressing instruction in which the memory is not needed by the processor. The line is asserted in the cycle before the unused cycle is available, and thus provides advance notice to the I/O device. The greatest advantage of the polite DMA concept is that it requires no overhead for I/O, since unused cycles are utilized.

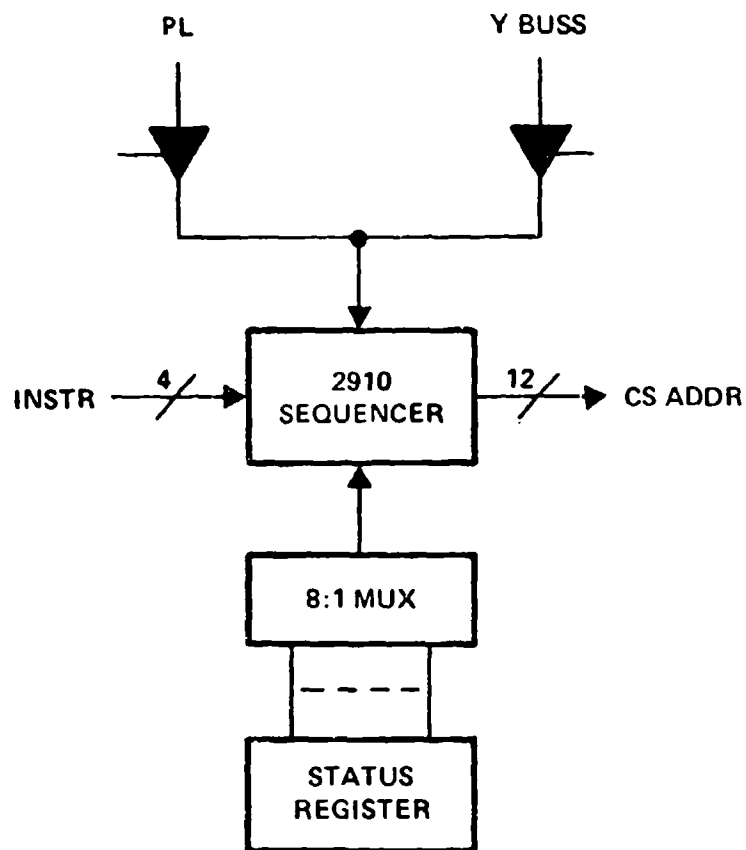
VI. Sequencer

The purpose of the sequencer is to generate addresses to the control store for fetching instruction. The sequencer organization is shown in Figure 5. It consists primarily of the AM2910 single chip micro sequencer. The 2910 executes 16 different instructions including a variety of conditional branches, subroutine calls, and looping operations. It also contains an internal loop counter which is used extensively in signal processing algorithms. Two data inputs are provided for the 2910. The pipeline register input is used for branch addresses and for setting the loop counter. The Y Bus input is for setting the loop counter when the number of loop execution is available. The condition code input for conditional



A90424-14

Figure 4. Address Generator Organization



A90424-15

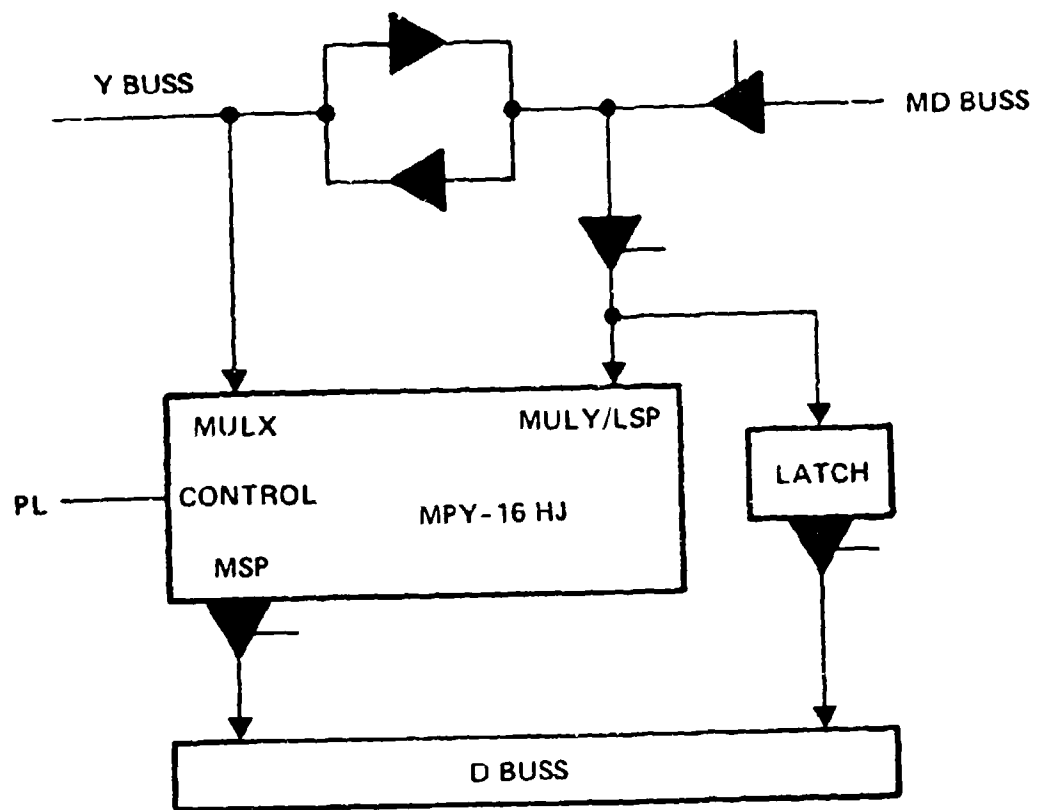
Figure 5. Sequencer Organization

operations is provided by an 8 bit status register followed by an 8:1 multiplexer. An XOR gate is also inserted in the condition code input line so that conditions can be sensed in either a true or inverted fashion. The status register contains 8 conditions:

- Zero
- Sign
- Carry
- Overflow
- Normalize (sensed from ALU output)
- Interrupt
- I/O sense (for polling I/O devices)
- True (for unconditional operations)

VII. Multiplier

The multiplier used in the HMSP is the single chip TRW device MPY-16HJ. This VLSI device executes a 16 x 16 bit multiply in a single clock cycle, and is used extensively in executing digital signal processing algorithms. The HMSP is organized so that the multiplier can be used to the fullest extent possible. The multiplier organization is shown in Figure 6. In this scheme, input data is loaded from the Y Bus, and output data is routed to the D Bus. Since the Y Bus can be driven by the MD Bus, the multiplier can be loaded directly from the memory. The split in the Y Bus is designed so that both multiplier inputs can be loaded simultaneously, one from the ALU,



A90424-16

Figure 6. Multiplier Organization

the other from memory. Because of pin out limitations on the TRW device the least significant product output port and one input port are multiplexed together. These two ports are effectively demultiplexed by the logic shown in Figure 6 so that it appears as a true four port device to the programmer.

Control inputs to the multiplier are provided from the PL register. These inputs are used to select unsigned/signed operation and to enable rounding of the product.

VIII. Data Memory

Data memory is organized around two busses. The memory address bus which originates from the address generator furnishes addresses to memory. The memory data bus routes data to and from memory. It originates from the output of the memory data write register and input to the memory data read register. The memory system operates synchronously with the processor and thus requires memory operations to be performed within a single instruction cycle. A write enable pulse is generated during write cycles by the processor. Read enable to the memory is by default in the absence of the write enable pulse.

IX. Input/Output

Input/Output for the HMSP is handled via a memory mapping of I/O devices. These devices are addressed by the processor as if they were memory locations. There are two methods of handling I/O in the processor, via DMA and interrupt schemes. The interrupt scheme in HMSP is very simple minded. A common "wire-or" interrupt line can be asserted by any device requiring service. The interrupt line is terminated in the status register within the CPU. From that point it can be tested by the running program. It should be noted that assertion of the interrupt line causes no action by itself (other

than changing a bit in the status register). All action relative to interrupts is performed by the application software residing in the HMSP. In order to isolate which I/O device is causing an interrupt, an I/O sense line is included in the architecture. This line is designed to be asserted by a device only when it is requesting interrupt service and its address is present on the memory address lines. This scheme allows for a fairly simple polling structure within the application software.

The primary method of handling I/O is via direct memory access. The DMA scheme used is a polite one which requires no overhead of the processor. This is achieved by letting I/O utilize memory cycles that are not used by the processor as discussed earlier, the address generator issues a DMA Grant to I/O devices during the cycle before an unused cycle is available. The DMA grant line is daisy chained along the memory bus to allow priority selection of devices. An I/O device latches the grant signal synchronously with the system clock and is able to perform its I/O operation during the following cycle. The memory address register is tri-stated to a high impedance state during unused cycles to allow I/O devices to operate.

X. 8080 Controller

An 8080 MOS microprocessor is included within the HMSP architecture. It is configured to serve in several functions:

- a. Low speed decision and control
- b. Program download
- c. Debug assistance

Low speed decision and control tasks can be executed by the 8080 in parallel with 2901 execution, depending on the target system requirements. It is often easier to program front panel, communications protocol and similar tasks within the 8080 than in the 2901.

The 8080 is equipped with hardware facilities to download programs into both the writable control store and data memory of the 2901. This facility is valuable not only for program development but is also desirable for some applications which require changing the program to be executed.

The final function assigned to the 8080 processor is related to debug assistance for the 2901 program. Hardware facilities are provided for examining and modifying register and memory contents within the 2901 as well as controlling execution with single step and breakpoint traps. The facilities afforded by the hardware are made available to the programmer through a CRT terminal. The debug capability obtained with this scheme is comparable to that available on commercial minicomputers and is far superior to that normally available for microcoded machines.

APPENDIX II

HMSP VOICE PROCESSING LOADING ESTIMATES

HMSP VOICE PROCESSING
LOADING ESTIMATES

27 September 1979

TABLE OF CONTENTS

<u>HEADING</u>	<u>PAGE</u>
Introduction.....	67
LPC-10 Loading Estimates.....	68
Pitch Low Pass Filter.....	70
Average Magnitude Distance Function.....	71
Matrix Load.....	72
Matrix Invert.....	72
Synthesis.....	73
HMSP Loading Estimates.....	74
LPC-4 Loading Estimates.....	76
16KB CVSD Loading.....	79
Summary.....	81

Voice Processing Loading Estimates

Introduction

There are currently three separate voice processing algorithms to be included in the Universal Terminal; LPC-10 in the 2400 BPS system, LPC-4 in the 9600 BPS system and CVSD in the 16KB system. In order to correctly size the HMSP for each of these systems it is necessary to estimate the computational loading (i.e., execution time) for each of the system components.

In the following sections the loading for each of the voice processing functions has been estimated for the Harris Micro Signal Processor. The estimates were obtained by first analyzing representative software which was written by Harris Corporation for the HMSP. From this code analysis, KETRON was able to estimate loading by comparing HMSP architecture characteristics to other machine characteristics on which the voice processing algorithms have been programmed.

Since most of KETRON's information on the voice algorithm is based on the KETRON Signal Processor (KSP) hardware, many of the comparisons are illustrated for the KSP and HMSP. Both the HMSP and the KSP contain a separate ALU which handles address arithmetic and removes a substantial load from the main ALU for array oriented subroutines where a significant amount of memory address indexing is done. The main difference between the KSP and the HMSP is that the HMSP is microprogrammed whereas the KSP contains a fixed instruction set in which all instructions, regardless of complexity, execute in a fixed time interval (800 nsec). In the HMSP each microinstruction executes in 200 nsec. For simple type instructions therefore, such as address register loading, incrementing a register, immediate data (PL data) handling, etc., which require only 1 μ cycle the HMSP enjoys a 4:1 advantage in execution time over the KSP. For very complex instruction in which three memory accesses/indexes (source, source, destination) take place the speed advantage is less but still may be as high as 2:1. This occurs since the HMSP address registers can be indexed in the address

arithmetic unit even while register to register arithmetic is being performed in the main ALU. This function cannot be done simultaneously in the KSP. By programming inner loops to take advantage of the HMSP's parallel processing capability, a significant speed advantage over existing processors should be obtained in executing the voice processing functions. The HMSP has basically a wide (64 bits), fast instruction memory common to microprogrammed machines but it also has incorporated a degree of parallel processing which is somewhat unique. The only real disadvantages of the HMSP are that it is more difficult to write and debug code efficiently and it uses a minimum of twice the amount of instruction memory as the KSP. I believe the performance obtained in the HMSP more than outweighs these disadvantages.

In the running time estimates then, the KSP instructions for the subroutines analyzed were broken down into three classes. Class 1 includes simple instructions which could be executed in a single HMSP cycle. Class 2 instructions are those which contain two indexed memory accesses. Class 3 instructions are those which contain three indexed memory accesses, contain a special function unique to the KSP which might require more than 4 HMSP machine cycles, or is a shift instruction. Shifting in the KSP occurs in a single cycle regardless of the direction, type or amount of the shift. In the HMSP shifting typically occurs 1 bit/cycle and may require several cycles to complete.

LPC-10 Loading Estimates

The LPC-10 is a frame oriented algorithm in that "blocks" or frames of speech data are inputted before the analysis begins. The average block contains 180 speech samples which must be processed.

The functional elements and typical loading of the LPC-10 algorithm are listed in Table 1. The major time consuming routines are the Pitch LPF, AMDF, Matrix Load, Matrix Invert and Synthesis (PSYN). Together these five routines account for almost 70% of the vocoder execution time. The execution time of each of these routines is analyzed in detail below to estimate the HMSP execution time.

LPC-10 Vocoder	Function	Subroutine	Type Execution Time (msec)
Transmitter	Voicing/ Pitch Extraction (5.538 msec)	Input Buffering & DC bias removed	.3
		Pitch LPF (11 tap)	2.03
		Normalization	.148
		Pitch Buffer Shift	.087
		Energy Measure	.219
		Voicing	.02
		AMDF	1.68
		Minimum Search	.204
		Seesaw	.53
		Update	.29
		TRACE	.02
		Index to Pitch Conv.	.01
	R.C. Analysis (3.37 msec)	APHASE & Pre-emphasis	.237
		Window & Scaling	.244
		Matrix Load	1.72
		Matrix Invert	.70
		RMS Energy	.169
Receiver	Synthesis (4.68 msec)	Coding & Dig Output	.3
		Digital Sync	.1
		Digital Input & Decoding	.3
		Buffer Updating	.032
		PITSYN (2 calls to PCOEFF)	.914
		PSYN (typical)*	3.34
TOTAL			13.59 msec

* Execution time of PSYN depends on no. of epochs synthesized per frame.

Figure 1. Typical LPC Loading Estimates for Existing Processors

Pitch Low Pass Filter

The pitch low pass filter currently used in the ANDVT software is either a 3-4-5 sum filter, or a fourth order butterworth filter. The 3-4-5 filter was originally developed to minimize the number of multiplications in the filter since in the early LPC processors multiplication was more time consuming than addition. In new processors there is no penalty for multiplication and the 3-4-5 filter can be implemented with a 10 tap transversal filter which is mathematically equivalent to the 3-4-5 filter. Since the HMSP is efficient for this type of structure we will assume an 11 tap filter in the loading estimation to make a direct comparison with the KSP loading.

The filter software program consists of the following steps:

<u>HMSP # Cycles</u>	<u>Step Description</u>
12	Move previous "Tails" to front of input data buffer
1	Setup outer loop counter
2	Initialize input & output data memory addresses
1	Reset inner loop counter to 11.
2·11·N	inner loop → Multiply, index 2 memory addresses, accumulate products, loop
1	Store output data point, index output data memory address
1	outer loop → Set address of tap weights
1	Advance input data memory address to next sample
1	Decrement loop counter and test for end of outer loop
12	Save "Tails" for next frame

Therefore, the loading for filtering a frame of N=180 points is:

$$\begin{array}{rcl} 27 + (5 \cdot 180) & + & (2 \cdot 180 \cdot 11) = 4887 \text{ cycles} \\ \text{Setup + Outer} & & \text{Inner} \\ \text{Loop} & & \text{Loop} \end{array}$$

and the execution time is: $4887 \times .2 = .9774 \text{ msec.}$

Thus, the HMSP executes the pitch low pass filter slightly more than twice as fast as the KSP. It should be noted that the sequential order of the steps in this example are not necessarily optimum but this has no bearing on the loading.

Average Magnitude Distance Function

The bulk of the execution time in the AMDF subroutine occurs in the AMDF kernel itself. The AMDF is defined as follows:

$$AMDF(i) = \sum |S(n) - S(n+T_i)|$$

For the ANDVT there are 60 values of T and the summation includes 32 terms. A conservative estimate of the HMSP steps required for the AMDF kernel is as follows.

<u># HMSP Cycles</u>	<u>Step Description</u>
1	Initialize memory address of S(n+T)
1	Set outer loop counter
1	Initialize Memory Address of S(n)
1	Set inner loop counter = 32, Clear Accumulator
4	Subtract, index 2 memory address registers, take magnitude of result, accumulate, and loop until CTR = 0
1	Generate next memory address for T _i
1	Scale result, store in memory, index output data memory address
1	Decrement loop counter and test for end of outer loop.

Therefore the estimated loading for computing the AMDF kernel (inner and outer loop in the HMSP is:

$$(6 \cdot 60) + (4 \cdot 32 \cdot 60) = 8040 \text{ cycles}$$

outer
inner
loop
loop

Execution Time = 8040 x .2 = 1.608

For this example the HMSP and the KSP execute the AMDF in approximately the same time since the KSP executes the most time consuming part of the AMDF (the inner loop) in a single instruction, or the same time period as the HMSP's 4 instructions. This is a case where a special function included in the KSP equalized the time difference between the two processors.

Matrix Load

The matrix load has been coded and debugged on the HMSP. The execution time for this routine can therefore be considered an "actual" as opposed to estimated and it can be used to verify the assumptions made on the estimated subroutines.

The loading on the HMSP is 4041 cycles or .808 msec for a 179 point window. In actual LPC-10 vocoders a window size of 150 samples per frame is sufficient. Therefore, the loading scales down proportionally to .673 msec. By comparison, the KSP executing the identical autocorrelation calculation requires 1.44 msec. The HMSP is executing 2.13 times as fast as the KSP for the autocorrelation. This difference corresponds well to the difference projected for a process having 2 memory indexes/cycle.

Matrix Invert

The Matrix Invert has also been coded and debugged on the HMSP. The Matrix Invert is less "array oriented" than the Matrix load in that the arrays are shorter and there are more class 1 type instructions occurring in the Levinson recursion. For this reason, the HMSP should approach the 4:1 loading improvement over the KSP which was projected for this type of code.

The loading for the HMSP is 995 cycles or about .2 msec. From Table 1 it can be seen that the HMSP executes the Matrix Invert about 3.5 times faster than the KSP.

Synthesis

The most time consuming part of the synthesis subroutine occurs in the 10 tap direct form recursive filter which represents the vocal tract weighting function. The steps required to execute the synthesis filter for the HMSP are estimated below:

<u># HMSP Cycles</u>	<u>Step Description</u>
1	Set Memory Address to Prediction Coefficients.
1	Load accumulator with excitation sample, index excitation memory address, set loop counter to 10.
2	Inner Loop → Multiply/Accumulate, index two memory addresses, repeat 10 times.
3	Scale Accumulator, store in output buffer, index output memory address.
2	Outer Loop → Save MAX value of output samples.
1	Reset excitation memory address pointer.
1	Decrement loop counter and test for end of outer loop.

The HMSP loading for the synthesis filter, again conservatively estimating the number of cycles, yields:

$$(9 \cdot 180) + (2 \cdot 10 \cdot 180) = 5220 \text{ cycles}^*$$

Outer	Inner
Loop	Loop

* assumes 180 points per analysis frame.

The execution time on the HMSP is 1.04 msec. On the KSP the identical filter loop requires 2.3 msec. Yielding a speed advantage in the HMSP of 2.2.

HMSP Loading Estimates

Based on the analysis of the preceding routines we have estimated the loading on the HMSP for each of the LPC-10 sub-routines using the KSP execution times from Table 1 as a baseline. To generate the HMSP loading the KSP instructions were divided into the three classes as described earlier and each class was assigned a percentage of the total loading for each subroutine. The HMSP loading was then obtained by using the following guidelines: Class I instructions execute 3.5 times faster in the HMSP, Class II instructions execute twice as fast in the HMSP, and for Class III instructions, the HMSP was given no speed advantage over the KSP.

It should be noted that some of the loading estimates in Table 2 are slightly higher than the actual measured loading on the HMSP. These cases serve as a check on the estimation procedure (since they are close to the actuals), and also indicate that the loading estimation is a conservative one.

Also note that no loading for the overhead associated with A/D and D/A has been included in Table 2. A conservative estimate of this overhead would be about 10-15% of the total LPC loading or, at most, an additional 1 msec.

Subroutine	Percentage of Instruction Class			HMSP Execution Time
	I	II	III	
Input Buffering & DC bias removal	50	50	---	.117
Pitch LPF	---	100	---	1.015
Normalization	90	---	10	.045
Pitch Buffer Shift	90	10	---	.0267
Energy Measure	10	90	---	.1046
Voicing	100	---	---	.006
AMDF	---	---	100	1.68
Minimum Search	100	---	---	.058
Seesaw	50	50	---	.193
Update	50	50	---	.105
TRACE	50	50	---	.007
Index to Pitch Conv.	100	---	---	.002
APHASE & Pre-emp.	10	90	---	.1128
Window & Scaling	---	100	---	.122
Matrix Load	---	100	---	.72
Matrix Invert	90	10	---	.215
RMS Energy	100	---	---	.0483
Coding & Digital Output	100	---	---	.086
Digital Sync	100	---	---	.029
Digital Input & Decoding	100	---	---	.086
Buffer Updating	100	---	---	.01
PITSYN	90	10	---	.281
PSYN	25	75	---	1.49
TOTAL HMSP EXECUTION TIME*				6.56 msec

* Exclusive of analog I/O time.

Table 2. HMSP Execution Time for LPC-10

LPC-4 Loading Estimates

The LPC-4 algorithm is also a frame oriented process. It contains several subroutines which are similar in operation to those in LPC-10. The LPC-4 algorithm, however, is currently not as rigidly defined by the DoD as the LPC-10 algorithm. For purposes of the loading estimates we have assumed an identical sampling rate and frame rate for LPC-4 as the LPC-10. The functional elements of the LPC-4 algorithm are listed in Table 3 along with the loading estimates for the HMSP.

The loading estimate for pitch extraction is taken directly from the LPC-10 estimate. The estimates for the Matrix load and Matrix invert are scaled by 5/11 and 5/10 respectively (5/10 was used for the Matrix Invert since the last pass through the Levinson recursion must be completed to obtain the predictor coefficients). The synthesis loading is scaled by 4/10 from the LPC-10 synthesis (using only the loading of synthesis filter itself).

The Error signal calculation is the most time consuming algorithm in the LPC-4 system. Previous loading estimates on processors similar to the KSP indicate that this calculation executes about 2.8 times longer than the synthesizer. It should be noted that the error signal calculation loop actually contains the synthesizer function.

The remaining subroutines contribute only slightly to the loading and were estimated by dividing previous estimates by a factor of 2.75 (the average of the Class I and Class II instruction types). Because the data used for some of the LPC-4 estimates were estimates themselves (with a built-in safety factor) and the 2.75 conversion used is also conservative, there is a double safety factor included in the LPC-4 estimates for these routines.

A rough estimate of the relative complexity of LPC-4 vs. LPC-10 which has been used in the past is that neglecting the pitch extractor, which is common to both, the loading of the LPC-10 is about 1.4 times the loading of the LPC-4. The ratio of LPC-10/LPC-4 loading from Tables 1 and 2, excluding the pitch subroutine, is 1.17. This simply reflects the fact that the LPC-4 estimates are very conservative.

LPC-4 Vocoder		Estimated HMSP Execution Time
Transmitter	Input Buffering	.117
	Pitch Extraction	3.13
	Alpha Calculation	.211
	Reduced Waveform Calculation and Normalization	.254
	Correlation Coefficient Calculation (Matrix Load)	.327
	Prediction Coefficient Calculation (Matrix Invert)	.107
	"Q" Calculation	.038
	Error Signal Calculation	1.16
Receiver	Parameter Coding & Digital Output	.086
	Digital Sync	.029
	Digital Input & Decoding	.086
	Scaling of Previous Frame	.052
	Synthesis	.416
TOTAL HMSP Execution Time*		6.066

* Exclusive of analog I/O time

Table 3 . HMSP Execution Time for LPC-4

16KB CVSD Loading

The CVSD algorithm is the simplest conceptually of the three discussed here. Unlike the LPC algorithms it is not a block oriented algorithm although it can be easily implemented in block form. Since the CVSD will interface with the 16KB modem in the Universal Terminal it would be advantageous to synchronize the CVSD execution to the symbol rate of the modem (2.667KHz). For example, by sampling the speech at 8KHz (symbol rate x 3) and running the CVSD loop twice per sample to generate 16KB, the 16KB modem and CVSD programs can execute in tandem. This approach essentially leads to a block format in which the smallest block (compatible with the modem) would be three samples from which 6 bits (1 symbol) would be generated by the CVSD.

The CVSD loading shown in Table 4 assumes a three sample block and therefore, the execution time is directly comparable to the modem symbol length (375 usec).

Function		Estimated # of HMSP Cycles		Estimated No. of HMSP cycles for 6 Bits (1 symbol)
		Per Sample	Per Bit	
TX	Input Buffering	1	.5	3
	Pre Emphasis	3	1.5	9
	Comparison (Input vs. Feedback)	1	2	12
	Step Size Adaptation Logic	4	8	48
	Step Size Integrator (1st Order)	3	6	36
	Step Direction Generator	1	2	12
	Feedback Integrator (1st Order)	3	6	36
	Output Buffering	2	4	24
RX	Step Size Adaptation Logic	4	8	48
	Step Size Integrator	3	6	18
	Step Direction Generator	1	2	2
	Output Integrator (1st Order)	3	6	18
TOTAL HMSP Loading per Symbol				266 cycles = 53.2 usec.

Table 4. Estimated HMSP Loading for 16KB CVSD

Summary

The loading for each of the voice processing algorithms being considered for the HMSP has been estimated. To insure that the estimates are reasonable, several subroutines were coded and debugged on the HMSP and the execution time calculated. These "actual" times correspond closely to the estimated times for the same routines. Where actual vs. estimated comparisons could be made, the estimated loading was higher than the actual in all cases, indicating that the estimates are very conservative. Table 5 summarizes the results of the voice algorithm loading analysis.

Voice Algorithm	Execution Time	Loading Ratio	% Loading
LPC-10	6.56 msec	6.56/22.5	29.2%
LPC-4	6.066 msec	6.066/22.5	27%
CVSD	53.2 usec	53.2/375	14.2%

Table 5. Summary of HMSP Loading for the
Voice Processing Algorithms

APPENDIX III

MULTI-RATE TERMINAL MATERIAL LIST

#0021

RD-C22104-01 MODULE-PRICING MATERIAL-MASTER DATE 09/26/79 PAGE 1

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
ANALOG-CARD	ANALOG CARD	AA	REV STATUS C	REF32	1
ATD-CARD	ATD CARD	AA	REV STATUS C	REF32	1
CHASSIS-GEN	CHASSIS GEN	AA	REV STATUS C	REF32	1
CPU-CARD	CPU CARD	AA	REV STATUS C	REF32	1
CRYPTO-SS-CRD	CRYPTO SS CRD	AA	REV STATUS C	REF32	1
EQUALIZER	EQUALIZER	AA	REV STATUS C	REF32	1
FRONT-PANEL	FRONT PANEL	AA	REV STATUS C	REF32	1
INTERFACE-CRD	INTERFACE CARD	AA	REV STATUS C	REF32	1
KOC-PROC-CARD	KOC PROC CARD	AA	REV STATUS C	REF32	1
MEMORY-CARD	MEMORY CARD	AA	REV STATUS C	REF32	1
MODEM-TX	MODEM TX	AA	REV STATUS C	REF32	1
R/B-INTFC-BLK	R/B INTFC BLK	AA	REV STATUS C	REF32	1
R/B-INTFC-RED	R/B INTFACE RED	AA	REV STATUS C	REF32	1
REAR-PANEL	REAR PANEL	AA	REV STATUS C	REF32	1
INTERFACE-CRD	INTERFACE CARD	A0C80AG-10	A/D	EA23	1
ANALOG-CARD	ANALOG CARD	A0C80AG-12	INT CKT	EA23	1
ANALOG-CARD	ANALOG CARD	A07521SD	INT CKT	EA23	1
INTERFACE-CRD	INTERFACE CARD	A07533SD	INT CKT	EA23	1
ATD-CARD	ATD CARD	AF108	RECEIVER	EA23	2
MODEM-TX	MODEM TX	AM25LS23DC	INT CKT	EA23	1
CPU-CARD	CPU CARD	AM25507DC	INT CKT	EA23	2
EQUALIZER	EQUALIZER	AM29010C	INT CKT	EA23	3
CPU-CARD	CPU CARD	AM29020C	INT CKT	EA23	2
MODEM-TX	MODEM TX	AM29030C	INT CKT	EA23	2

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
MODM-1A	MODEM TA	AM291CCC	INT CKT	EA23	1
EQUALIZER	EQUALIZER	AM2950DC	INT CKT	EA23	2
ANALOG-CARD	ANALOG CARD	ANALOG-CARD-01	SOLDER	A/R25	1
ATO-CARD	ATO CARD	ATO-CARD-01	PW BOARD	EA23	1
ATO-CARD	ATO CARD	ATO-CARD-02	MISC	A/R25	1
FRONT-PANEL	FRONT PANEL	BJ-27	CONN	EA23	2
REAR-PANEL	REAR PANEL	BS2107F-0-1	FAN	EA23	1
R/B-INTFC-BLK	R/B INTFC BLK	CD4050	INT CKT	EA23	5
R/B-INTFC-RED	R/B INTFACE RED	CD4050	INT CKT	EA23	5
CHASSIS-GEN	CHASSIS GEN	CHASSIS-1	WIRE	A/R25	1
CHASSIS-GEN	CHASSIS GEN	CHASSIS-2	SOLDER	A/R25	1
CHASSIS-GEN	CHASSIS GEN	CHASSIS-3	MISC HWR	A/R25	1
CHASSIS-GEN	CHASSIS GEN	CHASSIS-4	METALLIZATION	A/R25	1
CHASSIS-GEN	CHASSIS GEN	CHASSIS-5	MISC ADHESIVE	A/R25	1
CHASSIS-GEN	CHASSIS GEN	CHASSIS-6	MISC INT CONN	A/R25	1
ANALOG-CARD	ANALOG CARD	CMR05F100JCDM	CAP FXD	EA23	1
ANALOG-CARD	ANALOG CARD	CMR06F102J00M	CAP FXD	EA23	1
CPU-CARD	CPU CARD	CPU-CARD-01	RES FXD 330 OHM	EA23	1

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
CPU-CARD	CPU CARD	CPU-CARD-02	CAP FXD-480 PF EA23	1
CPU-CARD	CPU CARD	CPU-CARD-03	RES FXD 2K EA23	2
CPU-CARD	CPU CARD	CPU-CARD-04	PM BOARD EA23	1
CPU-CARD	CPU CARD	CPU-CARD-05	MISC A/R25	1
CRYPTO-SS-CRD	CRYPTO SS CRD	CRYPTO-SS-03	MISC A/R25	1
CRYPTO-SS-CRD	CRYPTO SS CRD	CRYPTO-SS-01	LSI EA23	1
CRYPTO-SS-CRD	CRYPTO SS CRD	CRYPTO-SS-02	PM BOARD EA23	1
ANALOG-CARD	ANALOG CARD	DAC-80-CBI-V	INT CKT EA23	1
INTERFACE-CRD	INTERFACE CARD	DAC80CBI-V	INT CKT EA23	1
ANALOG-CARD	ANALOG CARD	DG1818A	INT CKT EA23	1
ATO-CARD	ATO CARD	DG307	ANALOG MUX EA23	2
EQUALIZER	EQUALIZER	EQUALIZER-01	PM BOARD EA23	1
EQUALIZER	EQUALIZER	EQUALIZER-02	MISC A/R25	1
FRONT-PANEL	FRONT PANEL	FRONT-PANEL-07	DISPLAY EA23	2
FRONT-PANEL	FRONT PANEL	FRONT-PANEL-1	WIRE A/R25	1
FRONT-PANEL	FRONT PANEL	FRONT-PANEL-2	SOLDER A/R25	1
FRONT-PANEL	FRONT PANEL	FRONT-PANEL-3	MISC HDNR A/R25	1

RD-CR2104-C1		MODULE-PRICING		MATERIAL-MASTER		DATE 09/26/79		PAGE 4	
MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY				
FRONT-PANEL	FRONT PANEL	FRONT-PANEL-4	SILK SCREEN	A/R25	1				
FRONT-PANEL	FRONT PANEL	FRONT-PANEL-5	PAINT	A/R25	1				
FRONT-PANEL	FRONT PANEL	FRONT-PANEL-6	KEY PAD	EA23	1				
INTERFACE-CRD	INTERFACE CARD	F330-100CPF	CAP FXD	EA23	8				
ATD-CARD	ATD CARD	F3643	USART	EA23	2				
ATD-CARD	ATD CARD	F6850	ADAPTOR	EA23	2				
CRYPTO-SS-CRD	CRYPTO SS CRD	F6850	INT CKT	EA23	2				
ATD-CARD	ATD CARD	G1722001	DIALER	EA23	2				
INTERFACE-CRD	INTERFACE CARD	HA-2425-5	INT CKT	EA23	1				
INTERFACE-CRD	INTERFACE CARD	HA-4602-5	INT CKT	EA23	4				
ANALOG-CARD	ANALOG CARD	HA1-2705-5	INT CKT	EA23	2				
ANALOG-CARD	ANALOG CARD	HA1-2735-5	INT CKT	EA23	2				
ANALOG-CARD	ANALOG CARD	HA1-4605-5	INT CKT	EA23	2				
ANALOG-CARD	ANALOG CARD	HA2-2605-5	INT CKT	EA23	2				
CPU-CARD	CPU CARD	HM7603	INT CKT	EA23	2				
CPU-CARD	CPU CARD	M2P15G-10M12	CRYSTAL	EA23	1				
CPU-CARD	CPU CARD	10M2901A-IDC	INT CKT	EA23	8				
KDC-PROC-CARD	KDC PROC CARD	1M6604	INT CKT	EA23	4				

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
INTERFACE-CRD	INTERFACE CARD	INTERFACE-01	L P FILTER	EA23	1
INTERFACE-CRD	INTERFACE CARD	INTERFACE-02	PM BOARD	EA23	1
INTERFACE-CRD	INTERFACE CARD	INTERFACE-03	MISC	A/R25	1
INTERFACE-CRD	INTERFACE CARD	JH-5042	SWITCH	EA23	2
ANALOG-CARD	ANALOG CARD	J1120-5W1	RELAY	EA23	7
KDC-PROC-CARD	KDC PROC CARD	KDC-PROC-01	LSI	EA23	1
KDC-PROC-CARD	KDC PROC CARD	KDC-PROC-02	LSI	EA23	1
KDC-PROC-CARD	KDC PROC CARD	KDC-PROC-03	PM BOARD	EA23	1
KDC-PROC-CARD	KDC PROC CARD	KDC-PROC-04	MISC	A/R25	1
ANALOG-CARD	ANALOG CARD	LF398H	INT CKT	EA23	1
ANALOG-CARD	ANALOG CARD	LH2311D	INT CKT	EA23	6
ANALOG-CARD	ANALOG CARD	LM113H	INT CKT	EA23	3
ANALOG-CARD	ANALOG CARD	LM311H	INT CKT	EA23	1
ANALOG-CARD	ANALOG CARD	LM555CH	INT CKT	EA23	1
A/D-CARD	A/D CARD	MC16409	PULSE CONV	EA23	2
MEMORY-CARD	MEMORY CARD	MEMORY-CARD-01	PM BOARD	EA23	1
MEMORY-CARD	MEMORY CARD	MEMORY-CARD-02	MISC	A/R25	1

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
REAR-PANEL	REAR PANEL	MILF19207/11	FUSE HOLDER	EA23	2
MODEM-TX	MODEM TX	MODEM-TX-01	PW BOARD	EA23	1
MODEM-TX	MODEM TX	MODEM-TX-02	MISC	A/R25	1
CPU-CARD	CPU CARD	MPY16HJ	INT CKT	EA23	1
REAR-PANEL	REAR PANEL	MS14001-222	PWR SWITCH	EA23	1
FRONT-PANEL	FRONT PANEL	MS16562-232	ROLL PINS	EA23	2
REAR-PANEL	REAR PANEL	MS25080-2	STUD,GRND	EA23	2
REAR-PANEL	REAR PANEL	MS27336114F35P	CONN	EA23	1
REAR-PANEL	REAR PANEL	MS27474E14F35S	CONN	EA23	1
FRONT-PANEL	FRONT PANEL	MS9152801U2B	KNOB	EA23	1
MODEM-TX	MODEM TX	MXJ40-2-20.0MHZ	OSCILLATOR	EA23	1
FRONT-PANEL	FRONT PANEL	M19500/519-02	LED	EA23	7
FRONT-PANEL	FRONT PANEL	M19500/520-02	LED	EA23	3
FRONT-PANEL	FRONT PANEL	M19500/521-02	LED	EA23	1
REAR-PANEL	REAR PANEL	M24308/2-4	CONN	EA23	1
REAR-PANEL	REAR PANEL	M24308/22-4	CONN	EA23	1
REAR-PANEL	REAR PANEL	M24308/4-4	CONN	EA23	1

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
ANALOG-CARD	ANALOG CARD	M39003/01-2368	CAP FXD EA23	12
ANALOG-CARD	ANALOG CARD	M39014/01-1201	CAP FXD EA23	2
ANALOG-CARD	ANALOG CARD	M39014/01-1213	CAP FXD EA23	2
ANALOG-CARD	ANALOG CARD	M39014/01-1219	CAP FXD EA23	4
ANALOG-CARD	ANALOG CARD	M39014/01-1452	CAP FXD EA23	2
ANALOG-CARD	ANALOG CARD	M39014/01-1456	CAP FXD EA23	128
ANALOG-CARD	ANALOG CARD	M39014/02-1407	CAP FXD EA23	2
ANALOG-CARD	ANALOG CARD	M39024/11-01	TEST POINT EA23	13
ANALOG-CARD	ANALOG CARD	M39024/11-03	TEST POINT EA23	2
ANALOG-CARD	ANALOG CARD	M83421/01-4065M	CAP FXD EA23	2
ANALOG-CARD	ANALOG CARD	M83421/01-4089M	CAP FXD EA23	4
INTERFACE-CARD	INTERFACE CARD	ME555	INT CKT EA23	2
R/B-INTFC-RED	R/B INTFACE RED	R/B-INTF-01	47UF ELECT EA23	2
R/B-INTFC-RED	R/B INTFACE RED	R/B-INTF-02	.022 UF ELECT EA23	5
R/B-INTFC-RED	R/B INTFACE RED	R/B-INTF-03	1K RESISTOR EA23	2
R/B-INTFC-BLK	R/B INTFC BLK	R/B-INTFC-01	47UF ELECT EA23	2
R/B-INTFC-BLK	R/B INTFC BLK	R/B-INTFC-02	.022UF ELECT EA23	5

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
R/R-INTFC-BLK	R/R INTFC BLK	R/R-INTFC-03	1K RESISTOR	2
ANALOG-CARD	ANALOG CARD	RCR07G100JS	RES FXD	1
INTERFACE-CMD	INTERFACE CARD	RCR07G100JS	RES FXD	18
ANALOG-CARD	ANALOG CARD	RCR07G101JS	RES FXD	3
ANALOG-CARD	ANALOG CARD	RCR07G102JS	RES FXD	24
ANALOG-CARD	ANALOG CARD	RCR07G103JS	RES FXD	18
ANALOG-CARD	ANALOG CARD	RCR07G104JS	RES FXD	1
ANALOG-CARD	ANALOG CARD	RCR07G121JS	RES FXD	16
ANALOG-CARD	ANALOG CARD	RCR07G122JS	RES FXD	1
ANALOG-CARD	ANALOG CARD	RCR07G124JS	RES FXD	1
ANALOG-CARD	ANALOG CARD	RCR07G153JS	RES FXD	4
ANALOG-CARD	ANALOG CARD	RCR07G182JS	RES FXD	1
ANALOG-CARD	ANALOG CARD	RCR07G185JS	RES FXD	2
ANALOG-CARD	ANALOG CARD	RCR07G202JS	RES FXD	3
ANALOG-CARD	ANALOG CARD	RCR07G242JS	RES FXD	1
ANALOG-CARD	ANALOG CARD	RCR07G243JS	RES FXD	1
ANALOG-CARD	ANALOG CARD	RCR07G302JS	RES FXD	2
ANALOG-CARD	ANALOG CARD	RCR07G303JS	RES FXD	1

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
ANALOG-CARD	ANALOG CARD	RCR07G332JS	RES FXD EA23	1
ANALOG-CARD	ANALOG CARD	RCR07G511JS	RES FXD EA23	1
ANALOG-CARD	ANALOG CARD	RCR07G512JS	RES FXD EA23	8
ANALOG-CARD	ANALOG CARD	RCR07G513JS	RES FXD EA23	3
ANALOG-CARD	ANALOG CARD	RCR07G514JS	RES FXD EA23	1
ANALOG-CARD	ANALOG CARD	RCR07G562JS	RES FXD EA23	2
ANALOG-CARD	ANALOG CARD	RCR07G621JS	RES FXD EA23	4
ANALOG-CARD	ANALOG CARD	RCR07G682JS	RES FXD EA23	1
ANALOG-CARD	ANALOG CARD	RCR07G752JS	RES FXD EA23	1
ANALOG-CARD	ANALOG CARD	RCR07G753JS	RES FXD EA23	1
ANALOG-CARD	ANALOG CARD	RCR07G912JS	RES FXD EA23	1
ANALOG-CARD	ANALOG CARD	RCR20G910JS	RES FXD EA23	3
ANALOG-CARD	ANALOG CARD	RCR32G2H7JS	RES FXD EA23	4
REAR-PANEL	REAR PANEL	REAR-PANEL-1	SOLDER A/R25	1
REAR-PANEL	REAR PANEL	REAR-PANEL-2	WIRE A/R25	1
REAR-PANEL	REAR PANEL	REAR-PANEL-3	MISC HOWR A/R25	1
REAR-PANEL	REAR PANEL	REAR-PANEL-4	SILV SCREEN A/R25	1

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
REAR-PANEL	REAR PANEL	RF2230	FILTER,PHR	EA23	3
INTERFACE-CRD	INTERFACE CARD	RJR24FP1024	POT	EA23	2
ANALOG-CARD	ANALOG CARD	RNC55M1000FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M1001FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M1002FM	RES FXD	EA23	7
ANALOG-CARD	ANALOG CARD	RNC55M1152FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M1401FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M1402FM	RES FXD	EA23	2
ANALOG-CARD	ANALOG CARD	RNC55M1472FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M1690FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M1821FM	RES FXD	EA23	4
ANALOG-CARD	ANALOG CARD	RNC55M2001FM	RES FXD	EA23	2
ANALOG-CARD	ANALOG CARD	RNC55M2322FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M2801FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M3742FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55M4020FM	RES FXD	EA23	2
ANALOG-CARD	ANALOG CARD	RNC55M4021FM	RES FXD	EA23	2

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
ANALOG-CARD	ANALOG CARD	RNC55H4752FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55H4991FM	RES FXD	EA23	4
ANALOG-CARD	ANALOG CARD	RNC55H5621FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55H6041FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55H7151FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55H7500FM	RES FXD	EA23	3
ANALOG-CARD	ANALOG CARD	RNC55H7501FM	RES FXD	EA23	4
ANALOG-CARD	ANALOG CARD	RNC55H7502FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55H8061FM	RES FXD	EA23	1
ANALOG-CARD	ANALOG CARD	RNC55H9530FM	RES FXD	EA23	1
INTERFACE-CRD	INTERFACE CARD	RNR55H/J061FM	RES FXD	EA23	25
EQUALIZER	EQUALIZER	SM-A-731254-1	CONN	EA23	2
CPU-CARD	CPU CARD	SN2910J	INT CKT	EA23	1
ANALOG-CARD	ANALOG CARD	SN74LS00J	INT CKT	EA23	6
INTERFACE-CRD	INTERFACE CARD	SN74LS00J	INT CKT	EA23	4
MEMORY-CARD	MEMORY CARD	SN74LS00J	INT CKT	EA23	4
ANALOG-CARD	ANALOG CARD	SN74LS02J	INT CKT	EA23	1
CPU-CARD	CPU CARD	SN74LS02J	INT CKT	EA23	1
ANALOG-CARD	ANALOG CARD	SN74LS04J	INT CKT	EA23	5

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
CPU-CARD	CPU CARD	SN74LS04J	INT CKT	7
INTERFACE-CRD	INTERFACE CARD	SN74LS04J	INT CKT	6
INTERFACE-CRD	INTERFACE CARD	SN74LS05J	INT CKT	2
CPU-CARD	CPU CARD	SN74LS08J	INT CKT	5
INTERFACE-CRD	INTERFACE CARD	SN74LS08J	INT CKT	2
ANALOG-CARD	ANALOG CARD	SN74LS109J	INT CKT	1
CPU-CARD	CPU CARD	SN74LS138J	INT CKT	2
INTERFACE-CRD	INTERFACE CARD	SN74LS138J	INT CKT	4
MEMORY-CARD	MEMORY CARD	SN74LS139J	INT CKT	2
ANALOG-CARD	ANALOG CARD	SN74LS163AJ	INT CKT	3
INTERFACE-CRD	INTERFACE CARD	SN74LS163J	INT CKT	4
ANALOG-CARD	ANALOG CARD	SN74LS174J	INT CKT	2
ANALOG-CARD	ANALOG CARD	SN74LS175J	INT CKT	1
MODEM-TX	MODEM TX	SN74LS244J	INT CKT	1
CPU-CARD	CPU CARD	SN74LS245J	INT CKT	2
INTERFACE-CRD	INTERFACE CARD	SN74LS25J	INT CKT	2
CPU-CARD	CPU CARD	SN74LS27J	INT CKT	1
INTERFACE-CRD	INTERFACE CARD	SN74LS27J	INT CKT	12
ANALOG-CARD	ANALOG CARD	SN74LS273BJ	INT CKT	3

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
ANALOG-CARD	ANALOG CARD	SN74LS32J	INT CKT	1
CPU-CARD	CPU CARD	SN74LS32J	INT CKT	2
MEMORY-CARD	MEMORY CARD	SN74LS32J	INT CKT	4
CPU-CARD	CPU CARD	SN74LS373J	INT CKT	2
ANALOG-CARD	ANALOG CARD	SN74LS374J	INT CKT	4
CPU-CARD	CPU CARD	SN74LS374J	INT CKT	6
INTERFACE-CRD	INTERFACE CARD	SN74LS374J	INT CKT	15
MODEM-TX	MODEM TX	SN74LS374J	INT CKT	3
ANALOG-CARD	ANALOG CARD	SN74LS377J	INT CKT	5
MODEM-TX	MODEM TX	SN74LS377J	INT CKT	3
ANALOG-CARD	ANALOG CARD	SN74LS670J	INT CKT	12
INTERFACE-CRD	INTERFACE CARD	SN74LS73J	INT CKT	2
ANALOG-CARD	ANALOG CARD	SN74LS74J	INT CKT	6
CPU-CARD	CPU CARD	SN74LS74J	INT CKT	3
ANALOG-CARD	ANALOG CARD	SN74LS86J	INT CKT	2
CPU-CARD	CPU CARD	SN74S00J	INT CKT	1
EQUALIZER	EQUALIZER	SN74S00J	INT CKT	2
EQUALIZER	EQUALIZER	SN74S02J	INT CKT	2
CPU-CARD	CPU CARD	SN74S04J	INT CKT	1
EQUALIZER	EQUALIZER	SN74S04J	INT CKT	1
MEMORY-CARD	MEMORY CARD	SN74S04J	INT CKT	12
CPU-CARD	CPU CARD	SN74S10J	INT CKT	1

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
MODEM-TX	MODEM TX	SN74S10J	INT CKT	1
EQUALIZER	EQUALIZER	SN74S126J	INT CKT	2
MEMORY-CARD	MEMORY CARD	SN74S138J	INT CKT	1
MODEM-TX	MODEM TX	SN74S139J	INT CKT	1
CPU-CARD	CPU CARD	SN74S140J	INT CKT	2
CPU-CARD	CPU CARD	SN74S151J	INT CKT	1
EQUALIZER	EQUALIZER	SN74S153J	INT CKT	3
CPU-CARD	CPU CARD	SN74S157J	INT CKT	3
EQUALIZER	EQUALIZER	SN74S163J	INT CKT	2
MODEM-TX	MODEM TX	SN74S175J	INT CKT	1
EQUALIZER	EQUALIZER	SN74S182J	INT CKT	1
MODEM-TX	MODEM TX	SN74S194J	INT CKT	2
EQUALIZER	EQUALIZER	SN74S244J	INT CKT	4
EQUALIZER	EQUALIZER	SN74S251J	INT CKT	2
MODEM-TX	MODEM TX	SN74S251J	INT CKT	2
MODEM-TX	MODEM TX	SN74S253J	INT CKT	2
EQUALIZER	EQUALIZER	SN74S273J	INT CKT	4
EQUALIZER	EQUALIZER	SN74S283J	INT CKT	6

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
CPU-CARD	CPU CARD	SN74S32J	INT CKT	EA23	1
CPU-CARD	CPU CARD	SN74S37J	INT CKT	EA23	1
MODEM-TX	MODEM TX	SN74S37J	INT CKT	EA23	1
CPU-CARD	CPU CARD	SN74S374J	INT CKT	EA23	5
EQUALIZER	EQUALIZER	SN74S374J	INT CKT	EA23	11
MEMORY-CARD	MEMORY CARD	SN74S374J	INT CKT	EA23	8
CPU-CARD	CPU CARD	SN74S74J	INT CKT	EA23	3
EQUALIZER	EQUALIZER	SN74S74J	INT CKT	EA23	1
EQUALIZER	EQUALIZER	SN74S85J	INT CKT	EA23	4
CPU-CARD	CPU CARD	SN74S86J	INT CKT	EA23	1
EQUALIZER	EQUALIZER	SN74S86J	INT CKT	EA23	1
MODEM-TX	MODEM TX	SN74S86J	INT CKT	EA23	1
ANALOG-CARD	ANALOG CARD	SN7406J	INT CKT	EA23	2
ANALOG-CARD	ANALOG CARD	SN754518JG	INT CKT	EA23	10
CPU-CARD	CPU CARD	SN81LS95J	INT CKT	EA23	18
EQUALIZER	EQUALIZER	TDC1009	MULT	EA23	1
ANALOG-CARD	ANALOG CARD	T2220	XFMR	EA23	2
INTERFACE-CRD	INTERFACE CARD	T2220	XFMR	EA23	2
CHASSIS-GEN	CHASSIS GEN	YH725-18-11.520	OSC (MH)	EA23	1
CHASSIS-GEN	CHASSIS GEN	Z99L0577	PAINT	DZ25	16
ANALOG-CARD	ANALOG CARD	1P' -1	DIODE	EA23	9

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
ANALOG-CARD	ANALOG CARD	1N5636A	DIODE	EA23 6
REAR-PANEL	REAR PANEL	10-24-1055-14G	CLAMP	EA23 1
REAR-PANEL	REAR PANEL	102791-003	POWER CORD	EA23 1
EQUALIZER INTERFACE-CRD	EQUALIZER INTERFACE CARD	117985-001	CAP FXD	EA23 25
		117985-001	CAP FXD	EA23 60
CPU-CARD	CPU CARD	118757-001	CAP FXD	EA23 84
ANALOG-CARD	ANALOG CARD	119361-002	CONN,PCB(80PIN)	EA23 2
REAR-PANEL	REAR PANEL	141762-001	PWR SUPPLY	EA23 1
ANALOG-CARD	ANALOG CARD	2N2222A	1STR	EA23 1
ANALOG-CARD	ANALOG CARD	2N2907A	1STR	EA23 1
CHASSIS-GEN	CHASSIS GEN	20-11110	GASKET,EMI	F125 8
ANALOG-CARD	ANALOG CARD	2022-44	SURGE PROT	EA23 2
REAR-PANEL	REAR PANEL	2060-004	CAP,FEED THRU	EA23 10
MEMORY-CARD	MEMORY CARD	2147	INT CKT	EA23 16
CPU-CARD	CPU CARD	25LS253	INT CKT	EA23 3
REAR-PANEL	REAR PANEL	2646-24827-N194	INS WASHER	EA23 2
REAR-PANEL	REAR PANEL	267-102	BINDING POST	EA23 4
REAR-PANEL	REAR PANEL	267-103	BINDING POST	EA23 2

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
ANALOG-CARD	ANALOG CARD	3292P-1-102	RES VAR	EA23	1
ANALOG-CARD	ANALOG CARD	3292P-1-103	RES VAR	EA23	2
ANALOG-CARD	ANALOG CARD	3292P-1-501	RES VAR	EA23	5
ANALOG-CARD	ANALOG CARD	3292P-1-503	RES VAR	EA23	2
ATO-CARD	ATO CARD	34001	INT CKT	EA23	4
CRYPTO-SS-CRD	CRYPTIC SS CRD	34001	INT CKT	EA23	9
KDC-PROC-CARD	KDC PROC CARD	34001	INT CKT	EA23	7
KDC-PROC-CARD	KDC PROC CARD	34001	INT CKT	EA23	3
CRYPTO-SS-CRD	CRYPTO SS CRD	34011	INT CKT	EA23	3
KDC-PROC-CARD	KDC PROC CARD	34011	INT CKT	EA23	5
CRYPTO-SS-CRD	CRYPTO SS CRD	34012	INT CKT	EA23	2
ATO-CARD	ATO CARD	34013	INT CKT	EA23	8
CRYPTO-SS-CRD	CRYPTO SS CRD	34013	INT CKT	EA23	20
KDC-PROC-CARD	KDC PROC CARD	34013	INT CKT	EA23	8
KDC-PROC-CARD	KDC PROC CARD	34013	INT CKT	EA23	9
CRYPTO-SS-CRD	CRYPTO SS CRD	34014	INT CKT	EA23	2
ATO-CARD	ATO CARD	34015	SHIFT REG	EA23	10
CRYPTO-SS-CRD	CRYPTO SS CRD	34015	INT CKT	EA23	2
KDC-PROC-CARD	KDC PROC CARD	340175	INT CKT	EA23	3
CRYPTO-SS-CRD	CRYPTIC SS CRD	34019	INT CKT	EA23	6
CRYPTO-SS-CRD	CRYPTO SS CRD	34019	INT CKT	EA23	2
CRYPTO-SS-CRD	CRYPTO SS CRD	34021	INT CKT	EA23	2

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE QTY
KDC-PROC-CARD	KDC PROC CARD	34023	INT CKT	2
CRYPTO-SS-CRD	CRYPTO SS CRD	34030	INT CKT	4
KDC-PROC-CARD	KDC PROC CARD	34030	INT CKT	8
CRYPTO-SS-CRD	CRYPTO SS CRD	34034	INT CKT	5
ATO-CARD	ATO CARD	34035	SHIFT RCVR	2
KDC-PROC-CARD	KDC PROC CARD	34071	INT CKT	5
CRYPTO-SS-CRD	CRYPTO SS CRD	34075	INT CKT	2
KDC-PROC-CARD	KDC PROC CARD	34075	INT CKT	4
CRYPTO-SS-CRD	CRYPTO SS CRD	34081	INT CKT	4
KDC-PROC-CARD	KDC PROC CARD	34081	INT CKT	5
KDC-PROC-CARD	KDC PROC CARD	34081	INT CKT	6
CRYPTO-SS-CRD	CRYPTO SS CRD	34724	INT CKT	4
KDC-PROC-CARD	KDC PROC CARD	34724	LATCH	6
KDC-PROC-CARD	KDC PROC CARD	3843	USART	2
FRONT-PANEL	FRONT PANEL	399280-AP	SWITCH	1
REAR-PANEL	REAR PANEL	432001	CONN	1
ANALOG-CARD	ANALOG CARD	436578	EXTRACTOR	2
FRONT-PANEL	FRONT PANEL	48-10-403-10	PANL FAST RH	1
FRONT-PANEL	FRONT PANEL	48-20-403-10	PANL FAST LH	1
REAR-PANEL	REAR PANEL	5K5	LINE FILTER	1

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
ANALOG-CARD	ANALOG CARD	555-2007	LED	EA23	2
CHASSIS-GEN	CHASSIS GEN	565070	MOTHERBOARD	EA23	1
FRONT-PANEL	FRONT PANEL	565192	GASKET,EMI	EA23	1
FRONT-PANEL	FRONT PANEL	565822	PANEL,EXTRUSION	EA23	1
FRONT-PANEL	FRONT PANEL	565823	FILTER,AIR	EA23	1
REAR-PANEL	REAR PANEL	62365	HANDLE	EA23	2
CHASSIS-GEN	CHASSIS GEN	624040	CHASSIS,NORYL	EA23	1
REAR-PANEL	REAR PANEL	624084	PANEL,BLANK	EA23	1
ANALOG-CARD	ANALOG CARD	624168	PCB, ANALOG	EA23	1
MEMORY-CARD MODEM-TX	MEMORY CARD MODEM TX	63RA1681 63RA1681	INT CKT INT CKT	EA23 EA23	34 6
EQUALIZER	EQUALIZER	63RA283	INT CKT	EA23	5
INTERFACE-CRD	INTERFACE CARD	645TR	MIKE	EA23	1
REAR-PANEL	REAR PANEL	69-9310600011	FILTER,FAN	EA23	1
FRONT-PANEL	FRONT PANEL	7101SY2Q	SWITCH	EA23	1
ANALOG-CARD	ANALOG CARD	76P808	DIP SWITCH	EA23	1
EQUALIZER	EQUALIZER	8T28	INT CKT	EA23	3
FRONT-PANEL	FRONT PANEL	8018	PUSH BOT CAP	EA23	2

50-COLL-01 MODULE-PRICING MATERIAL-MASTER DATE 09/20/79 PAGE 20

MODULE NUMBER	MODULE NAME	PART NUMBER	PART DESCRIPTION	USAGE	QTY
KDC-PROC-CARD	KDC PROC CARD	8085	INT CKT	EA23	1
ANALOG-CARD	ANALOG CARD	8121A	FB SWITCH	EA23	1
KDC-PROC-CARD	KDC PROC CARD	8155	INT CKT	EA23	1
FRONT-PANEL	FRONT PANEL	8168H2	PUSH BUTTON	EA23	2
EQUALIZER	EQUALIZER	825208	INT CKT	EA23	5
MODEM-TX	MODEM TX	825181	INT CKT	EA23	1
MEMORY-CARD	MEMORY CARD	8304	BUS TRANSCEIVER	EA23	4
KDC-PROC-CARD	KDC PROC CARD	8335	INT CKT	EA23	8
MODEM-TX	MODEM TX	93L422	INT CKT	EA23	2

99 LAST PAGE 99



MISSION of Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control Communications and Intelligence (C³I) activities. Technical and engineering support within areas of technical competence is provided to ESD Program Offices (POs) and other ESD elements. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.



DEPARTMENT OF THE AIR FORCE
AIR FORCE RESEARCH LABORATORY (AFRL)

1 Jun 04

MEMORANDUM FOR DTIC-OCQ

ATTN: Larry Downing
Ft. Belvoir, VA 22060-6218

FROM: AFRL/IFOIP

SUBJECT: Distribution Statement Change

1. The following documents have been reviewed and have been approved for Public Release; Distribution Unlimited:

ADB084552, "Project Birdwatch at Dover AFB", RADC-TR-84-7

ADB191869, "Acousto-Optic Beam Steering Study", RL-TR-94-121

AD0800669, "Use of Commercial Broadcast Facilities for Emergency DoD Communications", RADC-TR-66-392

ADB058979, "Multi-Rate Secure Processor Terminal Architecture Study", RADC-TR-81-77, Vol 1.

ADB053656, "16 KB/S Modem (AN/GCS-38) CONUS Test", RADC-TR-80-89

ADB055136, "VINSON/AUTOVON Interface Applique for the Modem, Digital Data, AN/GCS-8", RADC-TR-80-341

ADB043556, "16 KB/S Data Modem Partitioning", RADC-TR-79-278

ADB029131, "16 Kilobit Modem Evaluation", RADC-TR-78-127.

2. Please contact the undersigned should you have any questions regarding this document. Thank you very much for your time and attention to this matter.

Kelly Cagwin
KELLY CAGWIN
STINFO Officer
Information Directorate
315-330-7094/DSN 587-7094