ESD-TR-75-253

Technical Note

1975-53

The Lincoln Digital Voice Terminal System

P. E. Blankenship
E. M. Hofstetter
A. H. Huntoon
M. L. Malpass
S. Seneff
V. J. Sferrino

25 August 1975

Prepared for the Department of the Air Force
under Electronic Systems Division Contract F19628-76-C-0002 by

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LEXINGTON, MASSACHUSETTS

The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology, with the support of the Department of the Air Force under Contract F19628-76-C-0002.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Eugene C. Raabe

Eugene C. Raabe, Lt. Col., USAF
Chief, ESD Lincoln Laboratory Project Office

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LINCOLN LABORATORY

THE LINCOLN DIGITAL VOICE TERMINAL SYSTEM

P. E. BLANKENSHIP
E. M. HOFSTETTER
A. H. HUNTOON
M. L. MALPASS
S. SENEFF
V. J. SFERRINO

Group 24

TECHNICAL NOTE 1975-53

25 AUGUST 1975

LEXINGTON MASSACHUSETTS

ABSTRACT

The Lincoln Digital Voice Terminal (LDVT) is an integrated, ultra-high performance, speech processing system consisting of a specially-designed 55-nsec, 16-bit minicomputer and a set of associated peripherals. Compact, cost effective, practical to replicate, easy to use, and amply matched to the processing loads imposed by current and evolutionary narrowband speech processing algorithms, 18-MHz cycle times have been achieved via an optimum coupling of state-of-the-art technology and special architectural features. To date, four entirely different full-duplex vocoder systems have been programmed. They are a linear predictive vocoder operating at 4.8, 3.6, or 2.4 Kbs, an adaptive predictive vocoder at 8 Kbs, an adaptive residual coding technique at 9.6 and 16 Kbs*, and a new algorithm called TRIVOC operating at 2.4 and 3.6 Kbs. In this report the LDVT hardware is described from the architectural viewpoint, a detailed enumeration of parts, costs, and services is given, and the vocoder algorithms are presented along with implementation details including breakdowns of running times and storage requirements.

*The linear predictive vocoder, the adaptive predictive vocoder and the adaptive residual coding algorithms were developed under the sponsorship of the Defense Communications Agency.

TABLE OF CONTENTS

I. INTRODUCTION

The Department of Defense sponsors an ongoing program of speech compression research with applications primarily in the area of narrowband, secure communications. The effort is coordinated by a consortium whose mandate is to uncover the most efficient, highest quality, lowest cost, minimum-bit-rate, compression systems possible. In the past few years, many systems such as channel vocoders, adaptive-prediction coders, linear-prediction coders, voice-excited linear prediction, adaptive residual coding and continuously varying slope delta modulation have been developed to accommodate varying bit-rate requirements. As more such systems can be expected to evolve, the consortium generally believes that a programmable processor, capable of real-time system implementation, constitutes a valuable experimental tool.

Lincoln Laboratory's initial effort in this direction is the Fast Digital Processor (FDP) computer complex [1], a large, immobile, laboratory-based machine with a highly parallel, pipelined architecture, which has been in operation since 1970. With second-generation emitter-coupled logic, FDP exhibits a 6.7-MHz, instruction execution rate and has borne out the desirability of a flexible, high performance, speech research facility.

Inherent with this large, complex, expensive, one-of-a-kind facility are programming difficulties and the inability to operate in a stand-alone mode. These shortcomings indicated that a second-generation processor was needed. A compact, easy to use, easy to replicate, relatively inexpensive facility capable of stand-alone operation and of equivalent or superior performance capability to FDP became the overall objective.

The Lincoln Digital Voice Terminal (LDVT) was designed to meet this objective . Comprised of a custom designed 55-nsec, 16-bit minicomputer and appropriate integrated peripherals, the LDVT has proven to be well matched to the real-time speech processing problem. In this report a technical description of the LDVT system is presented along with detailed parts/services compilation and costing. The genuine power and versatility of the processor is illustrated via detailed descriptions of 4, fully operational vocoder software packages that have been written for it. These algorithms include Linear Predictive Coding (LPC), Adaptive Predictive Coding (APC), Adaptive Residual Coding (ARC), and the new Triple Function Voice Coder System (TRIVOC).

II. LDVT SYSTEM DESCRIPTION

2.1 Minicomputer Architecture

The high performance minicomputer forms the "heart" of the LDVT processor and accounts for most of the circuitry. To handle the anticipated rigorous real-time processing loads, the machine's architecture had to be sufficiently simple to accommodate maximum rate-cycle times, yet sophisticated enough to permit implementation of a substantially powerful instruction set. At a 50-nsec cycle time, it should be possible to execute a large variety of nontrivial operations in a single machine epoch.

The end result (Figure 1) is a 2's complement, 16-bit, essentially fixed-point processor with software-controlled, extended-precision capability. The major subassemblies are a 512 x 16-bit high speed RAM used exclusively for program data and constants (M_D), a separate 1K x 16-bit RAM strictly for

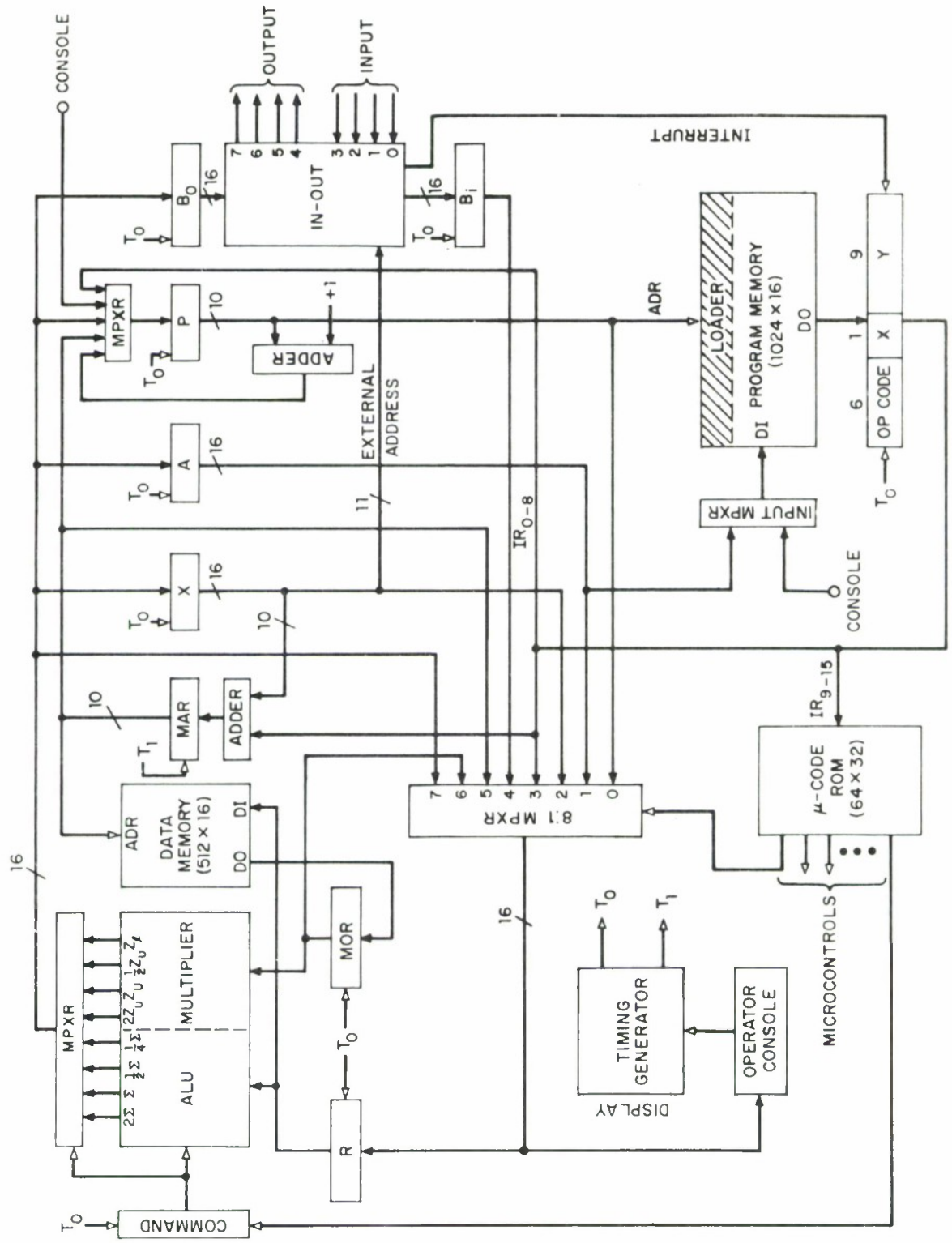


Fig. 1. LDVT minicomputer architecture.

18-2-12506

executable code (M_P), a bused file comprised of four active registers (A, X, P, B), a versatile arithmetic/logic unit (ALU), and an input-output system. In a typical operation an operand selected from M_D and another selected from the register file are operated on in the ALU. The result is returned to the register file. The register file can be loaded from or stored into M_D using the ALU as an intermediary where appropriate.

Each of the four conceptual elements of the register file has special functions. The A register is the primary machine accumulator, but also serves as a bootstrap buffer for code destined for loading into M_P. The X register can be used as an ancilliary accumulator, but serves mostly as an indexing component in M_D address calculation. The P register is actually the machine program counter, hence supplying address information to M_P. Alteration or sequencing of P in response to program status is normally controlled automatically by special hardware. However, its inclusion in the register file facilitates status save/restore operations in subroutine and interrupt handling. The B register is actually a pair of registers that serve as interface buffers for the input-output system. Peripheral in-out traffic handling and initial power-up bootstrapping are effected through this port.

The ALU (Figure 2) is divided conceptually into halves, only one of which can be actuated at a given time. One half consists of the logic necessary to perform the fundamental add/subtract and Boolean operations. Provisions are made for several output scaling options via a selection matrix. Subordinate logic is also included to implement overflow detection and carry status preservation for programmed multiple precision.

18-DO-10512

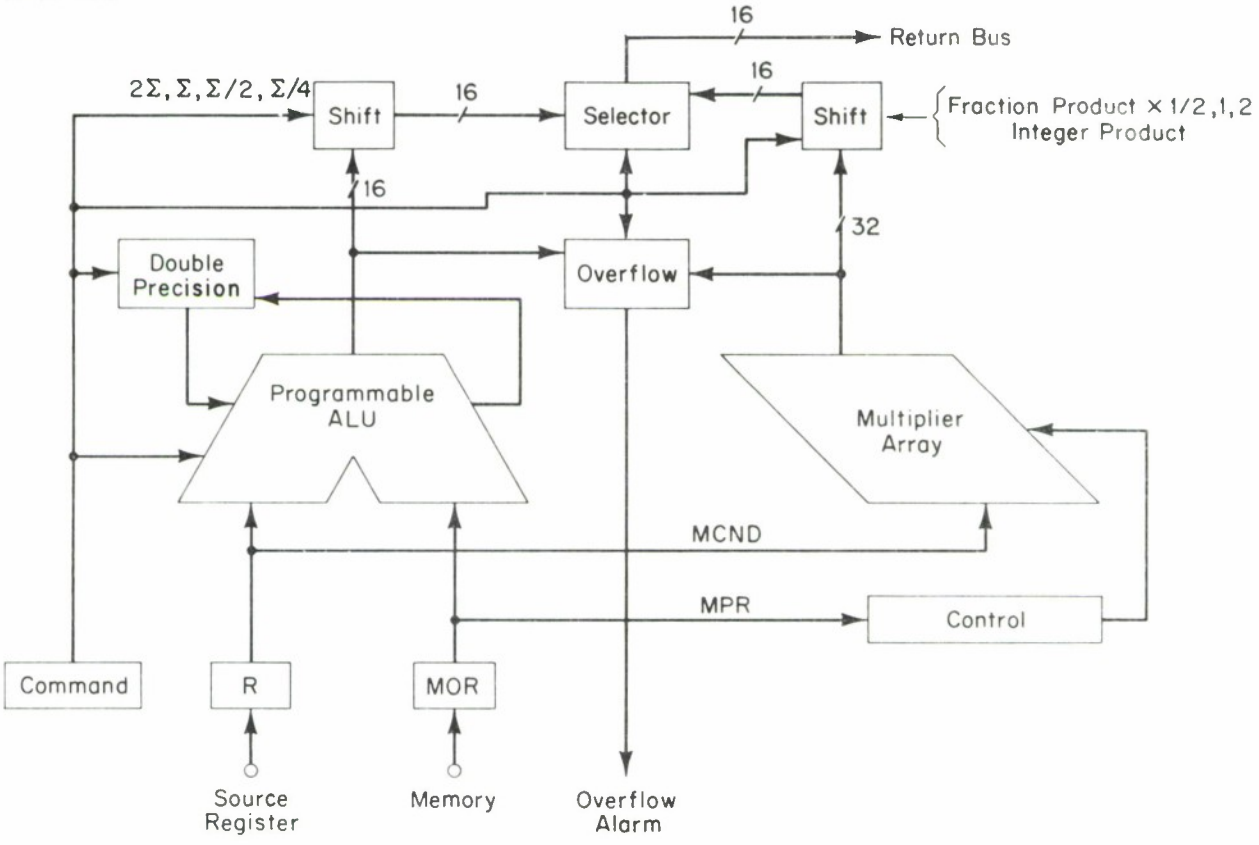


Fig. 2 The LDVT arithmetic unit.

The other half is a 16 x 16-bit multiplication element that forms a 32-bit signed product in 220 nsec. The design is a reentrant/reclocked type consisting of two hardware iterations of Booth's 3-bit multiplier coding algorithm. Four machine cycles are necessary to perform the effective eight iterations required to produce a 32-bit product. Any one of four possible 16-bit multiplier outputs can be selected at a time for transmission to the A register. They consist of the lower product half, upper half, and two shifted versions of the upper half. The lower half is always preserved for future retrieval in cases where the full 32-bit product is desired.

The R and MOR registers serve as intermediate buffers for the operands sourced from the register file and M_D, respectively. They are necessary due to the pipelined timing structure of the processor. The R register serves in a secondary capacity as an input buffer for M_D during data store operations.

The input-output system consists of single, 16-bit input and output channels along with appropriate control. Each of the channels is further multiplexed to four subchannels. Simultaneous input and output may be active, but only one subchannel of each type can be accommodated at a time. Transactions can be conducted on a vector priority interrupt basis, or by using a simple programmed test for completion. Input takes priority over output and only one level of interrupt service routine nesting is permitted, i.e., once an interrupt has been honored, all further interrupts are locked out until the interrupt service is completed. Completion is signalled via a special indirect branch instruction used to terminate the service routine and to return to the main program. Overflow, ALU carry, and program counter

statuses are saved automatically on interrupt. They are restored via the special termination instruction. Active register status must be saved and restored under software control.

2.2 Instruction Formats

To minimize cycle time, it was essential that a control with minimum decoding requirements be designed. For this reason, the LDVT minicomputer is virtually a one-format machine. The format (Figure 3) consists of a 6-bit operation code field, a 9-bit address/constant field (Y), and a single-bit special field (x). With the necessity of differentiating among several formats as a function of OP code eliminated, decoding could be effected efficiently by a fast 32 x 64-bit, micro-code read-only memory (ROM). Though the ROM technique affords the obvious advantage of custom instruction-set tailoring, its primary advantages are compactness and speed. The LDVT control constitutes somewhat of a degenerate case of the classic microprocessor control in that all but one machine instruction can be implemented in a single microstep. Overhead operations such as program counter maintenance and memory address calculation are performed automatically and in parallel with special explicit control logic.

The instruction repertoire that evolved, summarized in Appendix A, can be classified in three basic categories according to the type of action governed. The first of these, the arithmetic/logic class is of the general form:

$$f\ \{[R],\ [M_D\ (\alpha)]\} \rightarrow\ [R]$$

18-2-12507

Fig. 3. Basic instruction format.

where

$$R = A, X, B, P$$

$$\alpha = \begin{cases} Y, & \text{if } x = 0 \\ Y + [X], & \text{if } x = 1 \end{cases} .$$

The 9-bit Y field serves as a base address, capable of spanning all of M_D, which can be modified under control of the x bit by the contents of the X register.

The second class is the memory transfer group and has virtually the same structure as the arithmetic operations. Operations governed are of the general form:

$$[M_D\ (\alpha)] \rightarrow [R]$$

or $$[R] \rightarrow [M_D\ (\alpha)]$$

where

R = A, X, B, P as before. Operations of the form

$$[M_D\ (\alpha)] \rightarrow [P]$$

have the interesting effect of branching the running program. In fact, this is the means by which return-point restoration and indirect jumps are actually implemented.

The most interesting class is the control group and contains all conditional/unconditional branch codes as well as miscellaneous in/out handling instructions. Branches are of the general form

$$Y \rightarrow [P],$$

if conditions are met and

$$[P] + 1 \rightarrow [M_D(1)], \quad \text{if } x = 1.$$

Described verbally, a branch to location Y in M_P can be conditionally or unconditionally effected, and P status (return point) saved optionally in location 1 of M_D. Given a 1K M_P and a 9-bit Y field, branches can take place only within a 512-word page. Page boundaries are crossed using memory transfers into P, as described previously. Condition codes include overflow, input/output status, ALU sign, and sense switch tests. Auto-incrementing/decrementing jumps operating in conjunction with the X register are also included.

2.3 Timing Philosophy

The following sequence of events must occur to fully execute a given instruction:

a. P counter assumes desired state

b. M_P accessed

c. Fetched instruction interpreted, decoded

d. M_D address computed, if applicable

e. M_D and register file read

f. Execution

g. Result recorded.

Assuming the fastest circuit technology available, it would be impossible to accomplish this sequence in 50-nsec unless an utterly simplistic machine structure with very small memories is assumed. Calculations indicate that the above event chain could be segmented in thirds in a well-balanced way yielding a net cycle time on the order of 55-nsec. This implies a triple overlapped, pipelined type of timing arrangement with the usual attendant increase in control complexity. However, experience shows that the overall

package count increases suffered in such cases are usually modest and that the increased cycle time potential justifies the sacrifice.

To clarify details, consider the following symbolic code segment:

$[A] + [M_D] \rightarrow [A]$

$[A] \rightarrow [M_D]$

JPA Y

In this example, the A register is added to a location in M_D, the result is tested, and a branch to M_P (Y) takes place if it is positive. In a timing diagram of this sequence (Fig. 4) three time lines are marked off in units of machine cycles corresponding to M_P activity, decoding and setup, and final execution. The process begins by fetching the "add" instruction from M_P. At the end of the access cycle the instruction is buffered in an instruction register (IR) and M_P is accessed again to fetch the "store" instruction. Simultaneous with the second access, the "add" instruction is decoded and the register file is read. Also, the M_D operand address is computed and M_D is read. At the instant the "store" instruction is loaded into the IR, the operands associated with the "add" instruction are loaded into ALU buffers R and MOR. During the next cycle the "jump" instruction is fetched, the "store" instruction is decoded, and the "add" takes place in the ALU. At this point the three-level pipeline is full.

M_D address calculation requires half a machine cycle (25 nsec). The actual read takes place during the latter half. Rather than leave the memory idle during the first half, it is available for store operations. Therefore the execute portion of a store instruction actually occurs during the first half of the decode epoch of the subsequent operation.

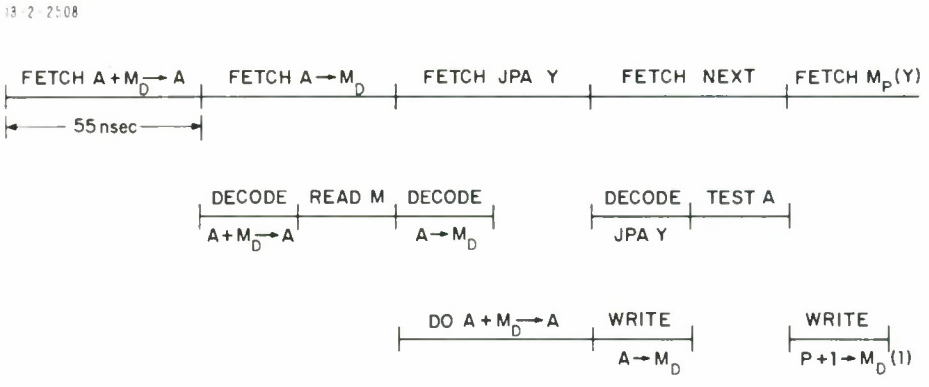


Fig. 4. LDVT timing example.

A curiosity of the pipelined type of timing arrangement involves emptying the pipeline on a branch operation. Because of the overlap, a further instruction is read from M_p before the control realizes a branch is to occur. In essence, a cycle is needlessly lost in emptying the pipe. In the case of the LDVT minicomputer it was decided that this cycle be available for use on an optional basis. That is, each branch instruction can either waste the cycle or not. If the cycle is used, the effect is to perform the next instruction subsequent to the branch irregardless of whether the branch actually takes place. Many programmers find this an exceedingly useful, though somewhat unusual, feature.

2.4 Peripheral System

To make a self-sufficient speech terminal out of what has been described as a general-purpose minicomputer required a wholly integrated set of appropriate peripheral elements. The LDVT peripheral complex (Fig. 5) consists of a 12-bit, analog-to-digital/digital-to-analog converter set, two 16-bit serial-to-parallel/parallel-to-serial converter sets, 4K x 16 ROM, 2K x 16 RAM, and a host computer channel.

The ADC/DAC set serves the obvious purpose of interfacing the local handset. The S-P/P-S sets mediate traffic flow of serialized data out to modems that interface with telephone lines or whatever other transmission medium is desired. The two sets provided include a conferencing capability wherein a given LDVT can transmit from one speaker yet receive from two others.

The host comptuer channel permits program assembling and editing in laboratory-based experimental environments. New software systems are thus easily transmitted to the LDVT. The host computer is also an effective debugging

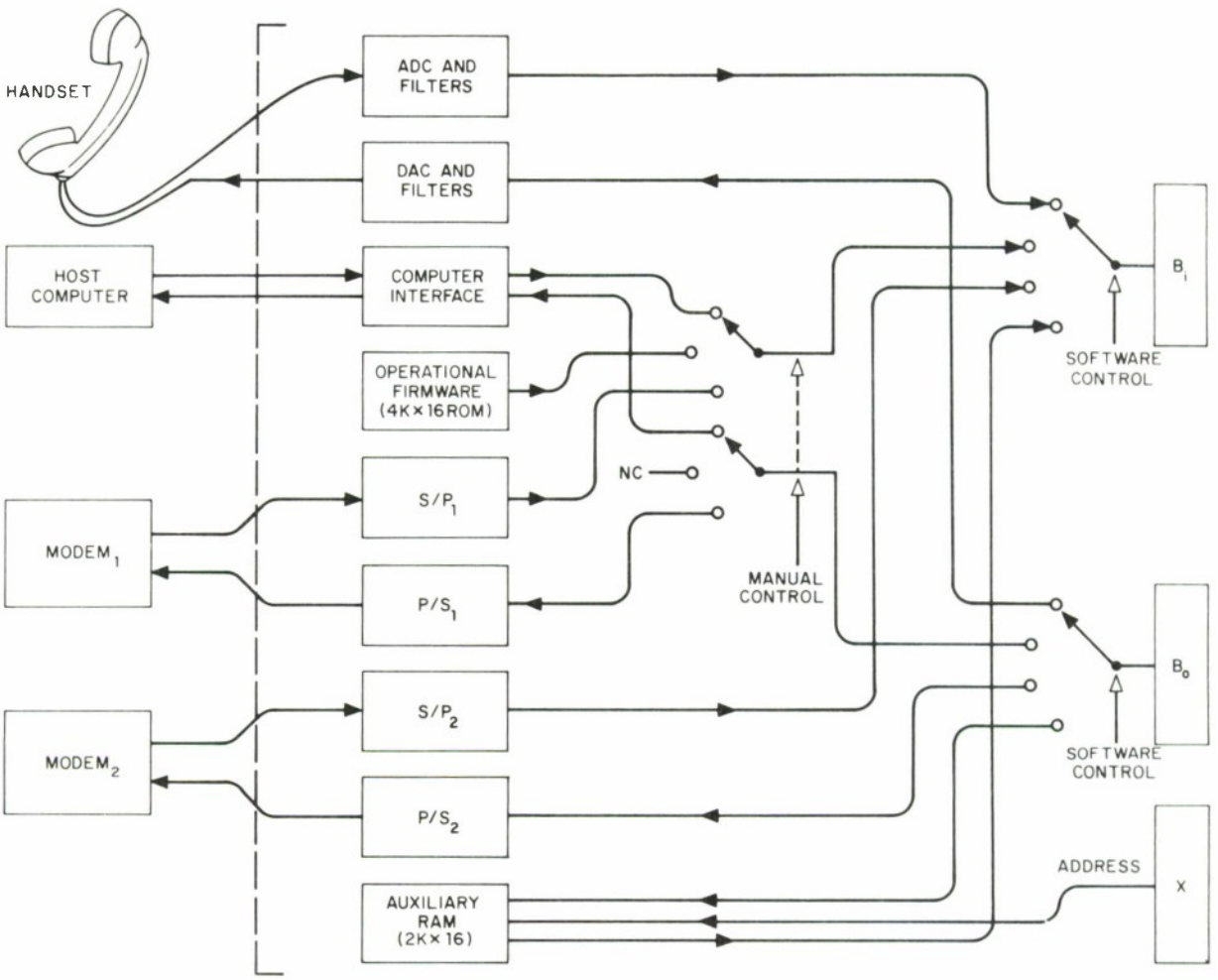


Fig. 5. Input-output complex.

tool to monitor LDVT memory dumps, etc. For stand-alone applications, however, a 4K x 16-bit bootstrap ROM takes the place of the computer channel. In such cases, the ROM contains the necessary operational firmware to personalize the LDVT to whatever speech compression algorithm the user desires. ROM contents are loaded into the minicomputer automatically on power-up controlled by a nonvolatile bootstrap loader in the first few M_p locations. This bootstrap loader can also acquire code from a host computer, if desired.

A high speed 2K x 16 auxiliary RAM (M_X) in the peripheral complex enhances the rather limited memory capacity of the minicomputer. Read/write operations can be streamed at a 200-nsec rate because of the RAM's high performance capability and the way its control is wedded to the computer in-out complex. Address information is supplied through the X register. In a typical operational system, M_X is used to store speech buffers, coding/decoding tables, or perhaps executable code bound for loading in M_p. The latter could occur when the running program is too large to fit into M_p at once, thus necessitating real-time code overlays.

III. ENGINEERING CONSIDERATIONS

3.1 System Fabrication and Packaging

The stringent performance and compactness requirements of the LDVT minicomputer restricted the choice of circuit technology to 10,000-series emitter-coupled logic (ECL 10K), a fully populated 2-nsec MSI family. The lower performance requirements of the peripheral system and outside world compatibility considerations indicated that standard 7400-series TTL could be utilized safely. The minicomputer has 498 ECL packages, all but 12 of which are of the 16-pin DIP configuration. The remainder, used in the ALU, are 24-pin

DIPs. 197 TTL 16-pin DIPs serve the peripheral complex along with a small analog board containing the DAC/ADC system, associated sampling/desampling filters, and miscellaneous audio amplification.

Given the brief development interval allotted, the entire LDVT, except the analog subsystem, was built with wirewrap construction techniques. It is well known [2] that ECL 10K with a 3-nsec rise time can be well controlled in a wirewrap environment as long as proper care is taken in signal path conditioning and DC power distribution. For example, signal paths must be terminated properly to control reflections, and loads must be constrainted carefully in number and physical position to preserve waveform quality. The terminations, ranging typically from 50 to 150 ohms, pose a special problem in that they consume board space and increase dissipation. The usual practice in ECL systems is to provide a special -2V termination voltage in addition the the standard -5.2V supply to conserve power. Since the DC distribution system must exhibit very high capacitance and low inductance in the interests of noise margin preservation, explicit strapping of a -2V supply on a standard, single-voltage, wirewrap board is an extremely dangerous practice. For this reason a special family of wirewrap board, intended for use with ECL systems and currently commercially available, was developed by Lincoln Laboratory. Though essentially similar to standard 180-pack configurations they differ in that a second, buried voltage plane is provided, along with proper decoupling capability, to handle the -2V distribution. In spaces between the 16-pin DIP sockets, special 8-pin, single-inline (SIP) sockets accommodate Cermet termination resistor packs of compatible configuration. The sockets connect directly to the buried -2V plane. In the LDVT system, only two standard

terminator SIP values were necessary: 100 and 150 ohms. By connecting pairs in parallel, values of 50, 60, and 75 ohms could also be achieved.

Four power supplies supplying 225 W of real power for the LDVT, include: 40-A switching supply for the ECL -5.2 V, 10-A linear regulator for the -2 V ECL termination voltage, 9-A linear regulator for the TTL +5 V, and $\pm$ 15 V supply for the analog equipment. Four 3-in., low acoustic noise fans at 50 CFM each provide forced air cooling.

The basic LDVT package (Fig. 6) fits in a 19 x 5 x 22 in. drawer, occupies about 1.25 cubic feet, and weighs 60 pounds. A small outboard box houses the analog equipment and serves as a receptacle for the handset. The LDVT digital electronics, housed on four wirewrap boards arranged in a stack (Fig. 7), open for access much as the pages of a book. Interboard connections are provided by controlled-impedance, flat ribbon cables running along the spine or "binding," obviating the need for a back plane. The bottom three boards are of the special ECL variety and comprise the minicomputer. The topmost board is a standard, single voltage, 180-pack, wirewrap board accommodating most of the peripheral system. Parts purchased for a single drawer [3] totalled about $13,500.

3.2 Parts and Services Compilation

Appendix B contains a detailed tabulation of DVT parts as of 15 July, 1975. The 5 categories delineated are as follows:

1. Integrated circuits
2. P/C cards, W/W boards, power supplies
3. Resistors and capacitors
4. Components
5. Mechanical package parts

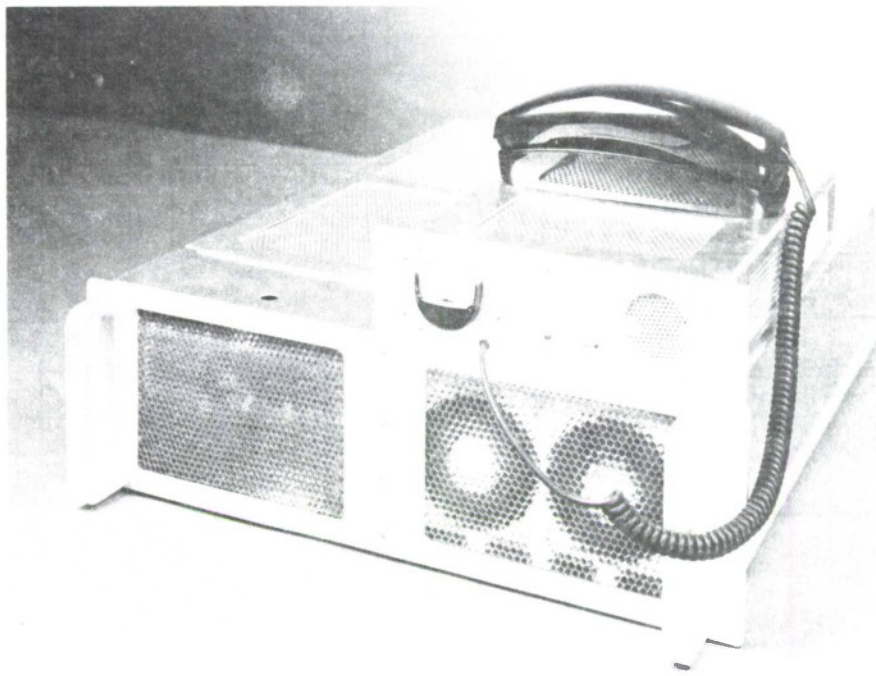


Fig. 6. LDVT ready for use.

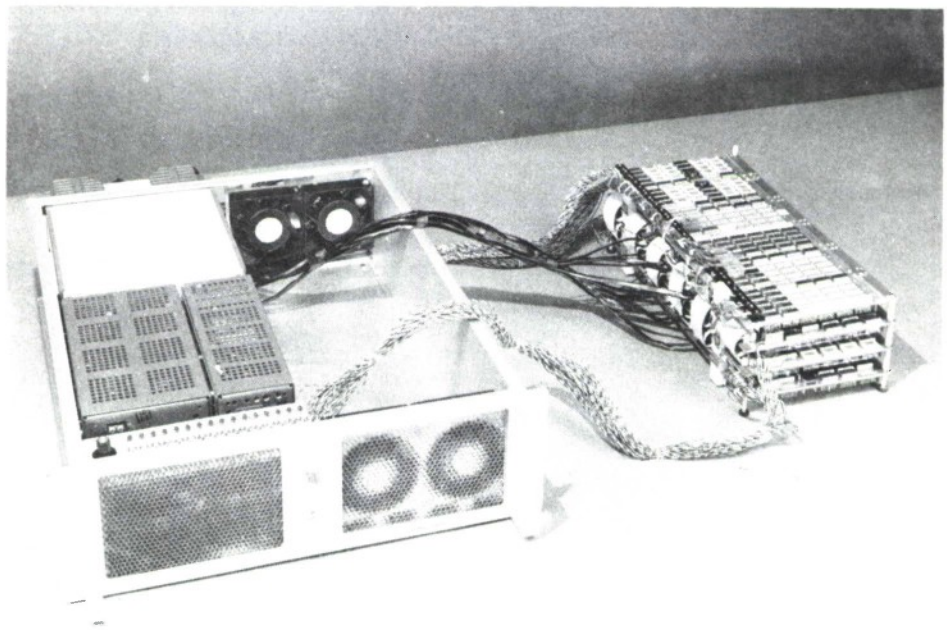


Fig. 7. LDVT open for service.

They are listed separately for the LDVT main drawer and for the external signal conditioner. Manufacturer's part numbers are given along with Lincoln stock numbers if applicable. It should be noted that wire-wrap charges are included and that integrated circuit costs reflect the 4K x 16 bootstrap ROM though this subsystem has not been populated in any of the existing LDVT units. Cost summary information is presented in Table 1.

Applying the extrapolation factors provided by the Narrowband Voice Consortium Hardware Subcommittee for estimation of "cost to produce," the results are as follows:

500 equipments: $32,400 ea.

1000 equipments: $29,700 ea.

10000 equipments $25,110 ea.

IV. VOCODER SOFTWARE IMPLEMENTATIONS

4.1 The Linear Predictive Vocoder

4.1.1 General Description of the Algorithm

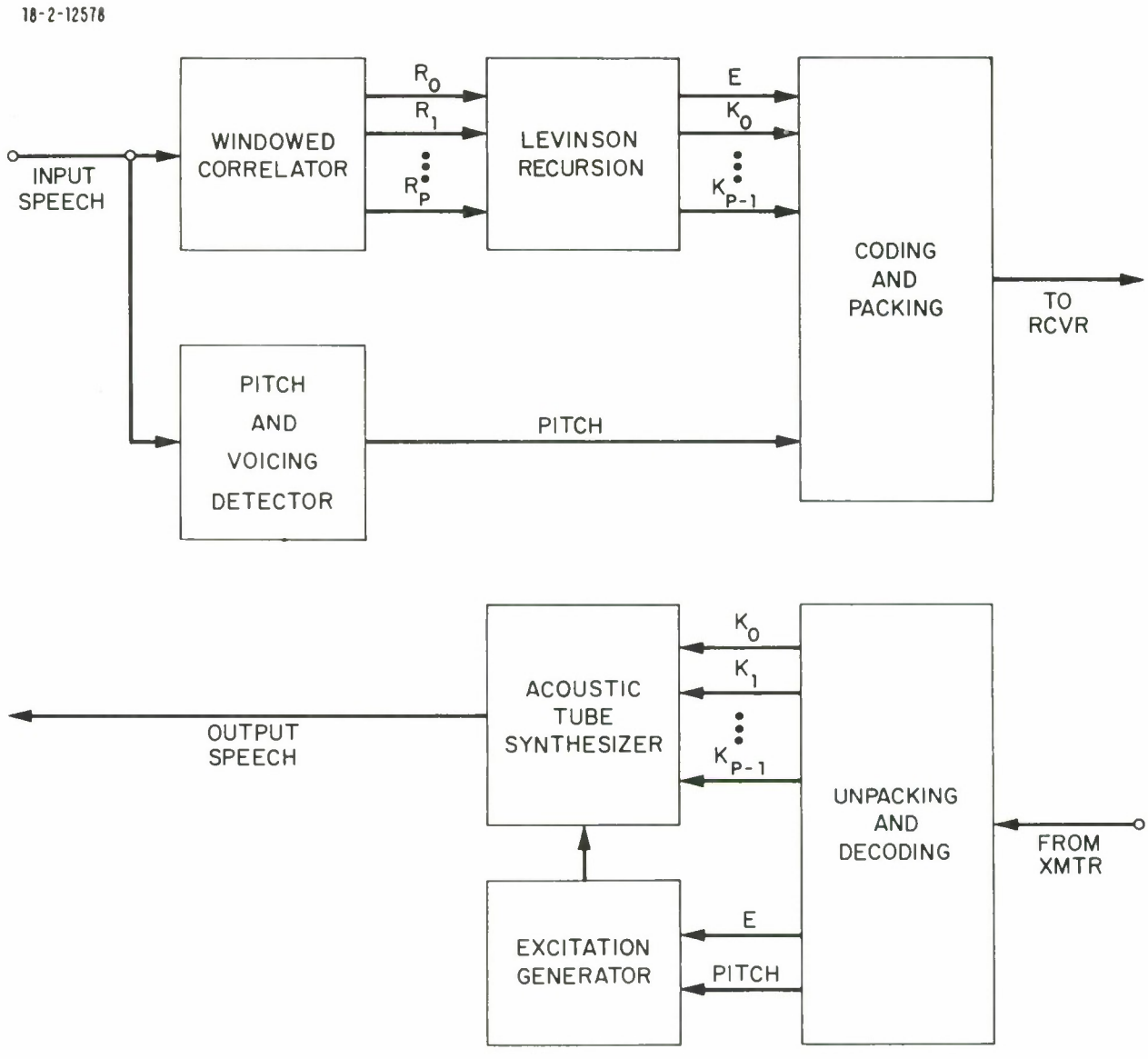
LPC was first described by Atal and Hanauer in 1971 [4]. Since then many variations on this algorithm have appeared in the literature (see bibliography in [5] and [6]). We have chosen to implement the Markel form of the LPC algorithms for reasons detailed in [7].

This algorithm is described in block-diagram form in Figure 8. Speech samples taken every 132 μs are divided into 158 point groups corresponding to approximately 20 ms of data. These groups are multiplied by a Hamming window and then used to form P+1 autocorrelation coefficients $R_o,...R_p$. The parameter P is the order of the filter used to model the vocal tract and ranges from 10 to 12 in current LPC systems.

The autocorrelation coefficients are used as the constants in a set of linear equations that must be solved to obtain the parameters of

CATEGORY	DVT	SIGNAL CONDITIONER	TOTAL
Integrated circuits	6262.15	433.59	6695.74
PC, WW panels, power supplies	3639.99	281.68	3921.67
Resistors, capacitors	510.92	38.77	549.69
Components	1475.26	463.31	1938.57
Mechanical package parts	290.72	84.45	375.17
TOTAL	12179.04	1301.80	13480.84

TABLE 1. DVT Parts Costs Summary



THE LPC VOCODER

Fig. 8. The LPC vocoder.

the vocal tract filter. These equations are solved by means of the Levinson recursion [8] which yields a set of P reflection coefficients $K_o,...K_{P-1}$ and a residual energy E. These reflection coefficients will be used at the receiver to implement the vocal tract filter. The structure chosen for this filter is the acoustic tube filter described in detail in [5]. The residual energy is used at the receiver to generate the amplitude of the excitation for the acoustic tube.

In addition to the processing described above, the raw speech samples are fed to a pitch and voicing detector which produces both a voiced-unvoiced decision and an estimate of pitch. The particular algorithm used for this purpose is the Gold-Rabiner pitch detector which is described in detail in [9].

The parameters produced as described above are next coded by means of a logarithmic-search table-look-up procedure and formed into a serial bit stream for transmission to the remote receiver. The receiver portion of the algorithm accepts such a serial bit stream from the remote transmitter and unpacks it to form the code book addresses of the various parameters. These addresses are then decoded to obtain the actual values of the parameters which are then used to implement the acoustic tube filter and its excitation. The output of the filter is the final synthetic speech.

4.1.2 Details of the LDVT Implementation

The LDVT program for realizing the LPC algorithm consists of two major pieces; a real-time program which is interrupt-driven by the A/D converter and handles those computations that must be made every time a new speech

sample is received, and a non-real-time program which handles those computations that must be made only when a complete frame of speech has been received.

The main task of the real-time program is to update the windowed correlator, the six elementary pitch detectors and the synthesizer filter. The details of the pitch detector update are presented in [9]. The synthesizer update consists of generating a sample of white noise whose amplitude is governed by the residual energy, E, if the frame is unvoiced, or the generation of either a zero or a pitch pulse of appropriate amplitude if the frame is voiced. The resultant excitation is then used to update the acoustic tube algorithm thus producing a synthetic speech sample which is fed to the D/A converter.

The correlation update is somewhat more complicated because of the requirement to produce complete 158 point correlations at a flexible rate. The need for this flexibility will be discussed later; the method used to achieve it was to start a new correlation every 159-S points rather than every 158 points. This means that more than one correlation must be updated at each interrupt but, as long as S is held less than 79, no more than two correlations must be updated at each interrupt. This provides a frame rate flexibility of 96 Hz to 48 Hz which is more than adequate for our needs.

The update of a single correlator is accomplished by first multiplying the incoming speech sample by its appropriate window value. The windowed speech sample is then pushed down on a stack of the previous P+1 such samples. The kth (k=0,...P) running correlation sum is then updated by adding to it the product of the most recent addition to the stack with the

kth entry of the stack. This addition is done with double-precision arithmetic; the full double-length stack product is added to the double-length running correlation sum. This process is facilitated by special LDVT double-precision instructions.

When the correlation routine determines that a 158-point double-precision correlation has been finished, it sets a flag that tells the non-real-time program to start its computation as soon as the real-time program has finished its current updates.

The basic tasks of the non-real-time program are the Levinson recursion, determination of pitch from the current state of the six elementary pitch detectors and, coding and framing. The Levinson recursion is straightforward and the final determination of pitch is described in [9]. The Levinson recursion is done with single-precision arithmetic; however, the necessary correlation coefficients are presented to it in block-floating-point format. A special routine left-justifies the double-precision RØ given it by the correlator and produces single-precision, block-floating-point correlation coefficients. The divisions required by the Levinson recursion are handled by an exact, but fairly slow (5 μs), divide subroutine.

The coding of the parameters produced by the non-real-time analysis, except for pitch which is transmitted as is, is accomplished by a logarithmic-search table-look-up routine. The residual energy is logarithmically coded to 5 bits. The reflection coefficients are coded by means of truncated, log-area ratios in which each reflection coefficient is first clamped to an individually selected interval, transformed by the log-area-ratio function (log [(1-K)/(1+K)]), and finally truncated to the desired number of bits.

After coding, the code-book addresses of the various parameters are packed into 16-bit words and delivered to the output buffer which is emptied by the parallel-to-serial converter and delivered to the transmit modem.

Since the transmit modem absorbs bits at an average rate determined by its internal clock, the analyzer portion of the LPC algorithm must adjust the average rate at which it produces bits accordingly. The latter rate is governed by the number of code bits assigned each frame and the independent A/D converter clock. Equality between these two rates is achieved by dynamic adjustment of the frame rate. This is the reason for the requirement that the real-time correlator be able to produce new correlations at arbitrary intervals.

Frame-rate control is achieved by means of a "bang-bang" servo technique. The locations of the buffer pointers loading and unloading the output buffer are monitored once each frame. The difference between these two pointers determines whether the overlap parameter, S, controlling the frame rate should be left as is or set to produce a higher or lower frame rate. This strategy guarantees that, on the average, the number of bits/second produced by the analysis program matches the number of bits/second being taken by the modem.

A similar tactic is employed by the real-time synthesis portion of the program which must insure that the rate at which it uses up bits matches, on the average, the rate at which the receiver modem is supplying them. Here control is exerted by monitoring the input-buffer loading and unloading pointers and using their difference to determine for how many samples the current

synthesis should continue before a new set of synthesis parameters are derived from the input buffer.

The final details of the LPC algorithm are summarized in Table 2 which depicts the running times and memory requirements of the various components of the algorithm. The salient points to be made here are that, with regards to running time, the machine is at least twice as fast as required for LPC, and just adequate as far as program and data memory requirements are concerned if overlay techniques are to be avoided. The use of overlays; however, enables the LDVT to execute considerably more demanding algorithms as will be illustrated in the later discussion of the TRIVOC algorithm.

4.2 The APC Vocoder

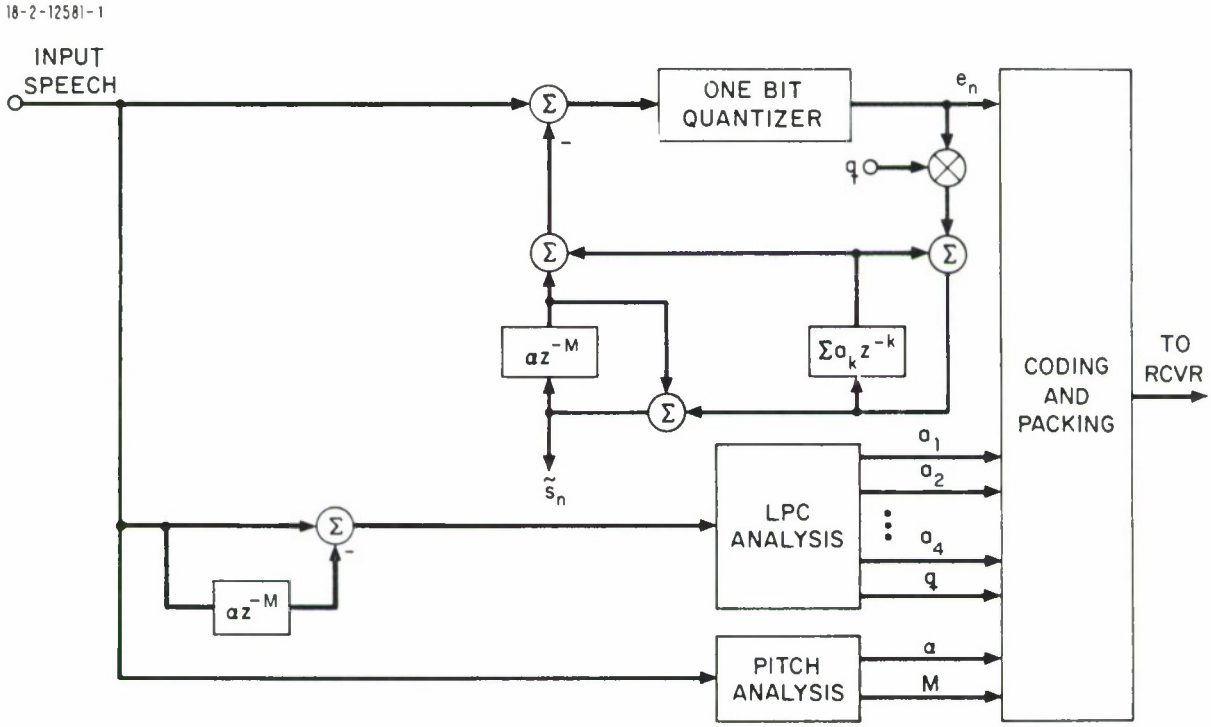
4.2.1 General Description of the Algorithm

The Adaptive Predictive Coding (APC) algorithm which was implemented on the LDVT is an 8K-bit system which recreates the speech waveform from a set of predictor parameters and an error signal. The algorithm was developed by Atal at Bell Laboratories in 1970 [10]. The LDVT version closely follows the modified algorithm described by Goldberg in [11], except that the sampling rate (and therefore the bit rate) is somewhat higher. Included is a fourth-order linear prediction, a pitch prediction, and a non-linear feedback loop.

The APC algorithm is diagrammed in Figure 9. The speech is filtered using a 170-μsec analog filter and sampled at 154-μsec intervals. Processing is begun on a new frame every 25.8 msec (N=168 samples). The first step is to determine the pitch, M, using the simple (but time consuming) AMDF pitch detector [11]. After M has been determined (regardless of whether a frame is voiced or unvoiced) the pitch predictor coefficient α is computed. Both

ALGORITHM	PROGRAM MEMORY	DATA MEMORY	OUTBOARD MEMORY	EXECUTION TIME	PER SAMPLE OR PER FRAME
Double Windowed Correlator	109	88	79	22 μs	PS
Real-Time Pitch	152	95	---	10 μs-23 μs	PS
Acoustic Tube Synthesis	72	67	---	8 μs	PS
Levinson Recursion	142	42	---	220 μs	PF
Pitch Determination	141	50	---	545 μs	PF
Coding/ Decoding	59	56	467	---	PF
Framing/ Deframing	110	32	---	---	PF

TABLE 2. LPC Breakdown

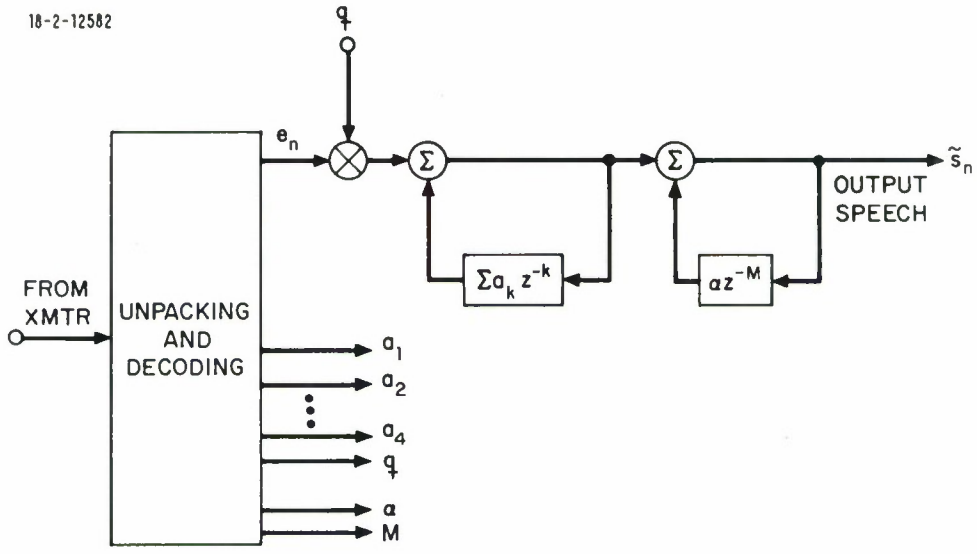


THE APC TRANSMITTER

Fig. 9. The APC transmitter.

M and α are computed using double precision arithmetic. It is now a simple matter to determine the waveform d(n) which is the output of the pitch filter and the input to the LPC analysis. In this analysis, five auto-correlation coefficients R_i are computed double precision, but stored single precision in a block floating point representation as in LPC. The linear prediction coefficients, a_i, reflection coefficients, K_i, and residual error, E, are computed using a fourth-order Levinson Recursion. The parameter q is determined from the residual error by the empirical approximation, $q \simeq .72\sqrt{\frac{E}{N}}$. Translation from E to q is achieved by table look-up in a 5-bit log table, to avoid the necessity of a square root algorithm. The four reflection coefficients, M, α, and q are all coded and the bits are packed and stored in the modem transmit buffer. The reflection coefficients and α are coded using a 5-bit arcsine table, q is coded logarithmically, and M is left uncoded. The coded values of the parameters are used by the analyzer in the feedback loop shown in Figure 9 to generate synthetic speech, $\tilde{s}(n)$, and the error signal, e(n).

A direct form filter, as contrasted with an acoustic tube realizatic is used in the linear prediction component of the speech synthesizer, following a conversion from coded reflection coefficients back to the coefficients a_i. The error is quantized to 1-bit/sample (the sign bit) and packed into the bit stream in the modem transmit buffer. The bits are then shipped across the channel to the other LDVT where, as shown in Figure 10, the synthesizer unpacks and decodes the parameters, and reconstructs synthetic speech which, in the absence of channel errors, is identical to the $\tilde{s}(n)$ previously computed by the analyzer.



THE APC RECEIVER

Fig. 10. The APC receiver.

4.2.2 Details of the LDVT Implementation

The APC implementation, unlike the LPC implementation, has essentially no real-time computation. In the real-time program, a new sample is received from the A/D converter and stored in the input buffer which is located in the LDVT's outboard memory. A new sample is fetched from the output buffer in this memory and sent to the D/A converter, and the modems (both transmit and receive) are checked and serviced if ready. In order to assure that no bits are dropped due to inadequate sampling of the modem, since the bit rate is higher than the sampling rate, the A/D converter is set to twice the sampling rate necessary, and on the odd-numbered interrupts only the modem is serviced.

The problem of drift between the modem clock and the A/D clock is handled quite differently than in the LPC program. Adjustment due to slippage of the pointers in both the analyzer and the synthesizer is done by either skipping an entire frame or repeating an entire frame twice, but only during silence periods. Pointers remain in a danger region for a sufficiently long time that at least one silence frame essentially always occurs before the pointers would collide.

It was decided to incorporate an elaborate synchronization algorithm into the APC program which is capable of resynchronizing in a few seconds should a bit be dropped. Both analyzers send with each frame a 2-bit synchronization code which is verified by each receiver. If a receiver finds 3 frames in which the 2-bit message was incorrect, it assumes that it has lost synchronization and responds by sending a 32-bit special code to the other DVT. When this code is detected, by a matched filter routine which is always

checking, a new frame is started at the end of the 32-bit message and the 32-bit special code is then sent to the first LDVT. Finally, these 32-bits are detected and the other LDVT resynchronizes.

Since there is no real-time processing in the APC implementation, the program structure is a straightforward sequence of subroutines corresponding to the block diagrams in Figures 9 and 10. In Table 3, these subroutines are listed and their memory and time requirements are given. The most costly algorithm in terms of time is the AMDF pitch detection. If time were tight, one could easily cut the AMDF time in half by only allowing even values of pitch, at some slight degradation in quality.

As in the LPC implementation, program memory and data memory are essentially exhausted. Some memory could be gained, if necessary, by replacing the matched filter synchronization algorithm with something simpler.

APC is characterized by a large number of speech buffers because the pitch filter necessitates a delay of M samples. Both the synthetic speech and the input speech are double buffered in the outboard memory. In each case, one buffer is a slave to the A/D-D/A while the other is used in the processing of the next frame. In addition, the analyzer's feedback loop requires a buffer of the previous M samples, but need not be double buffered since the synthetic speech produced in the analyzer is not sent to the D/A converter. The current implementation requires 3 buffers of length N+PMAX (PMAX is the maximum allowable pitch period) and two of length N and uses up 65% of the outboard memory.

ALGORITHM	PROGRAM MEMORY	DATA MEMORY	OUTBOARD MEMORY	EXECUTION TIME
Buffer Handling	62	N+PMAX = 288	0	.84 Msec
Pitch Extraction	37	7	N+PMAX = 288	13.3
Computation of α	56	5	---	.27
Correlation	80	14	---	.84
$d(n) = s(n) - \alpha s(n-M)$	11	0	---	.10
Levinson Recursion	134	20	---	.035
Coding	161	60	64	.12
Analysis Feedback Loop	67	10	N+PMAX = 288	.82
Decoding	87	60	64	.05
Synthesis	61	20	N+PMAX = 288	.72
A/D-D/A, Modem	75	9	2N = 366	.77
Matched Filter, Synchronization	154	8	---	.55
TOTAL	985=96%	507=99%	1328=65%	18.415=74%

N=68
PMAX=1

TABLE 3. APC Breakdown

4.3 The Triple-Function Voice Coder

4.3.1 General Description of the Algorithm

The basic idea behind the TRIVOC algorithm is to divide the speech spectrum into a low-frequency and a high-frequency portion. The low-frequency portion is analyzed and synthesized using LPC techniques and the high-frequency portion is analyzed and synthesized using classical channel vocoder techniques. The two synthesized speech waveforms are then summed to reproduce the final speech output [12].

The Lincoln version of the TRIVOC algorithm uses a sampling interval of 132 μs and divides the 0-3300 Hz speech spectrum into roughly two equal parts, 0-1500 Hz and 1500-3300 Hz. The low-frequency portion is produced by digitally filtering the input speech samples using a sixth-order version of the LPC algorithm described above. This produces a set of six reflection coefficients and a residual energy which is coded, packed and shipped to the receiver. The unfiltered speech is also sent to a Gold-Rabiner [9] pitch detector and the resulting pitch is also packed into the bit stream being sent to the receiver.

The algorithm used for the high-frequency portion of the spectrum is sketched in Figure 11. The input speech is suitably scaled and then passed to a Lerner filter bank consisting of eight filters each having a bandwidth of 225 Hz and spaced to cover the range 1500-3300 Hz. Each Lerner filter consists of a parallel combination of four second-order sections, however, a pole-sharing technique is employed [13] so that only two additional pole-pairs are required for each additional filter in the bank. This results in the entire filter bank realization with a total of eighteen pole-pairs.

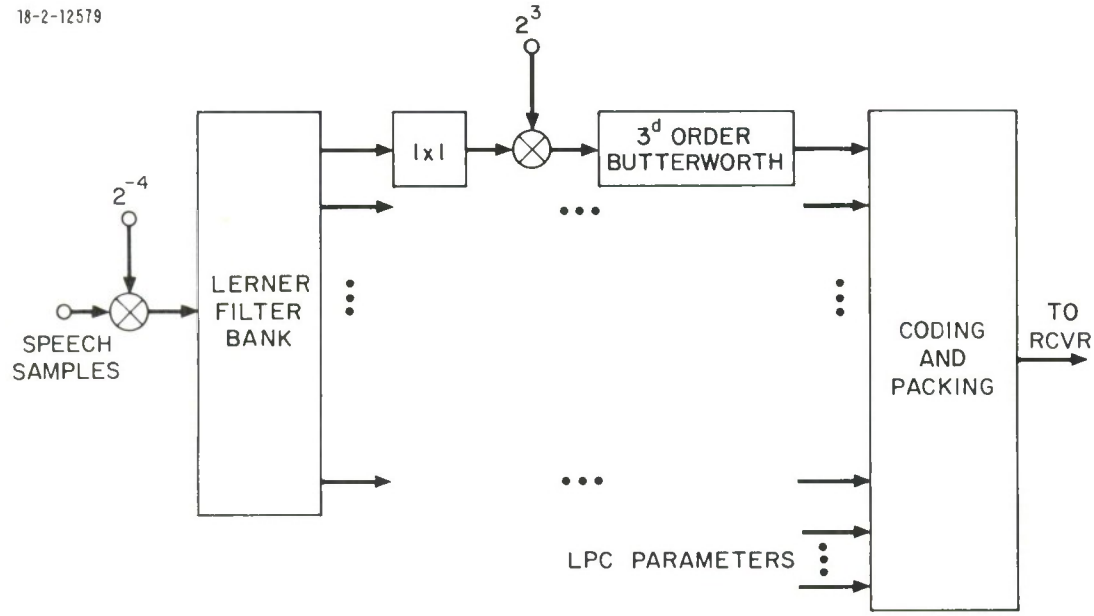


Fig. 11. The TRIVOC transmitter.

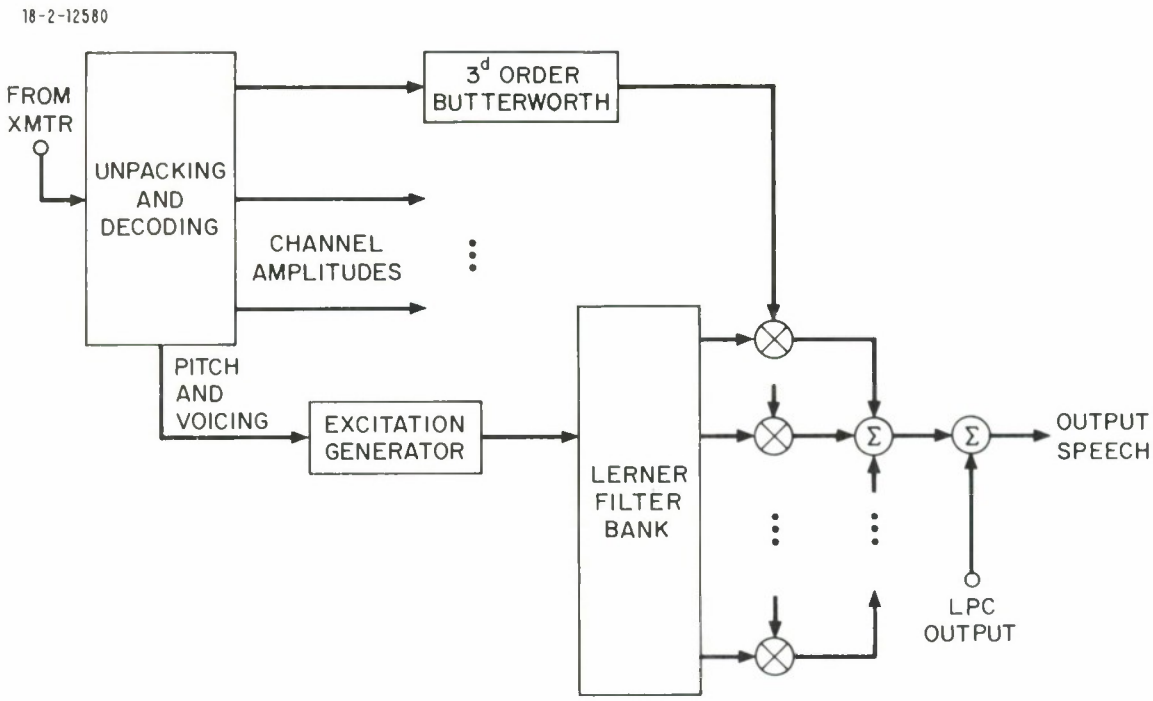
The magnitude of the output of each filter is then taken, the result is scaled and then low-pass filtered using a third-order Butterworth filter. The outputs of these filters are sampled at the end of each frame (roughly every 20 ms) and the resulting values are logarithmically coded, packed and sent to the receiver.

At the receiver, the incoming bit stream is unpacked and the parameters pertinent to the LPC portion of the spectrum are used to generate a low-pass speech waveform. Since this portion of the synthesis was done at half the sampling rate, these output samples must now be upsampled by a factor of two and low-pass filtered before being added to the output of the channel vocoder part of the synthesis.

As depicted in Figure 12, the pitch information coming from the transmitter is used to derive a constant RMS valued excitation (white noise or a periodic pulse train) for a Lerner filter bank that is an exact replica of the one used at the transmitter. The channel amplitudes coming from the transmitter are decoded and smoothed using a third-order Butterworth filter. The outputs of these filters are then used to amplitude modulate the outputs of the filter-bank filters. The resulting outputs are then summed and added to the LPC output to produce the final output speech.

4.3.2 Details of the LDVT Implementation

The TRIVOC program is basically an addition to the LPC program described earlier. All framing, coding, decoding serialization and de-serialization are carried out by the LPC program. The addition of the code to perform the channel vocoder part of the algorithm was quite straightforward except for the fact that, due to the lack of program memory space, program



THE TRIVOC RECEIVER

Fig. 12. The TRIVOC receiver.

overlays were required. This was accomplished by dividing the non-real-time part of the program into three tasks each of which was stored in the LDVT's outboard memory. A control program was then written that, at the appropriate times, read these tasks in from the outboard memory and stored then in program memory where they were then executed. Interrupts are active during the overlay process so that there is no chance of losing data even though the overlay process is quite time consuming. A detailed breakdown of the memory allocations and running times for the channel vocoder part of the TRIVOC algorithm is given in Table 4.

4.4 The Adaptive Residual Coder

4.4.1 Description of the Algorithm

The general algorithm for the Adaptive Residual Coder has been discussed in detail in [14]. The particular algorithm being used by Lincoln Laboratory consists of a second-order fixed predictor and an adaptive error quantizer as shown in Figure 13. Two systems have been implemented on the LDVT, one with a five-level quantizer which transmits at a rate of 9600 bits per second and another with a seven-level quantizer which runs at 16,000 bits per second. The adaptive quantizer has both a slow and fast decaying memory of previous quantization levels, and it determines the unit of quantization. At the kth instant, both the transmitter and receiver update the unit of quantization T(k), using the equations:

$$G'(k) = G'(k-1)*(1-2^{-7}) + f_1(d(k-1)) \geq 0$$

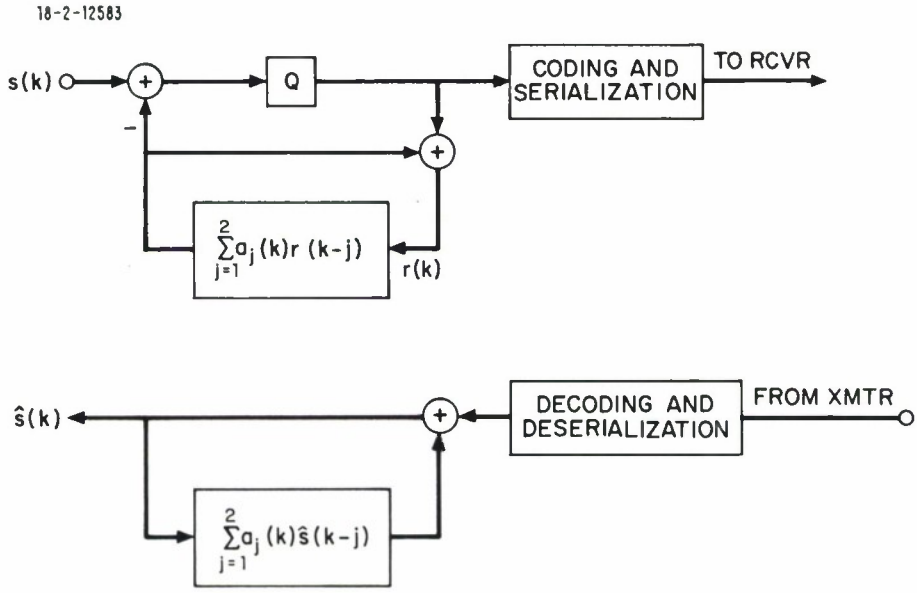
$$C(k) = C(k-1)*(1-2^{-2}) + f_2(d(k-1))$$

$$G(k) = G'(k) + C(k) + GMIN$$

$$T(k) = 2^{G(k)} \quad ,$$

ALGORITHM	PROGRAM MEMORY	DATA MEMORY	OUTBOARD MEMORY	RUNNING TIME	PER SAMPLE OR PER FRAME
LPC Input LPF	26	21	0	5.34 μs	PS
LPC Output LPF	33	9	0	5.91 μs	PS
8th Order Channel Analyzer	76	79	I6	31.82 μs	PS
8th Order Channel Synthesizer	85	70	I6	34.07 μs	PS
Extra Memory Due to Overlay Structure & Channel Additions	37	2	543	---	---

TABLE 4. TRIVOC Subsystem Breakdown



THE ARC VOCODER

Fig. 13. The ARC vocoder.

where f_1 and f_2 are functions of the previous slice level d(k-1) and GMIN is a constant. The function of the quantizer is to increase the quantization unit when the previous errors have reached the outer levels and to decrease the quantization unit when the previous errors have approached the zero level. The quantity G'(k) serves to adjust the quantization unit based upon the long-term behavior of the slice levels. C(k) responds quickly to occurrences of outer slice levels but persists for a shorter period of time. Once the quantization unit has been computed, the transmitter determines the quantizer error Q(e(k)) and the slice level d(k) as shown in Figure 14. The predicted signal at the transmitter plus the quantized error is remembered for later use by the fixed predictor. The receiver adds the quantized error to its predicted value to produce the output signal which is also to be used by its fixed predictor.

4.4.2 Details of the LDVT Implementation

Speech is sampled and outputed at 165-μs intervals via direct interrogation of the analog-to-digital and digital-to-analog converters, and all processing is done in real-time. When the transmitter determines the slice level, (-2 through +2 or -3 through +3), this level is coded into one of five or seven variable length codes and entered into a serial bit stream buffer 512 bits long. The receiver extracts and decodes the next slice level in its 512-bit buffer. At the rates of 9600 and 16,000 bits per second, the modem clock is faster than the A/D clock. A sufficient number of interrogations to the serial-to-parallel and parallel-to-serial converters is made during each 165-μs interval to assure that a modem clock pulse will never be missed.

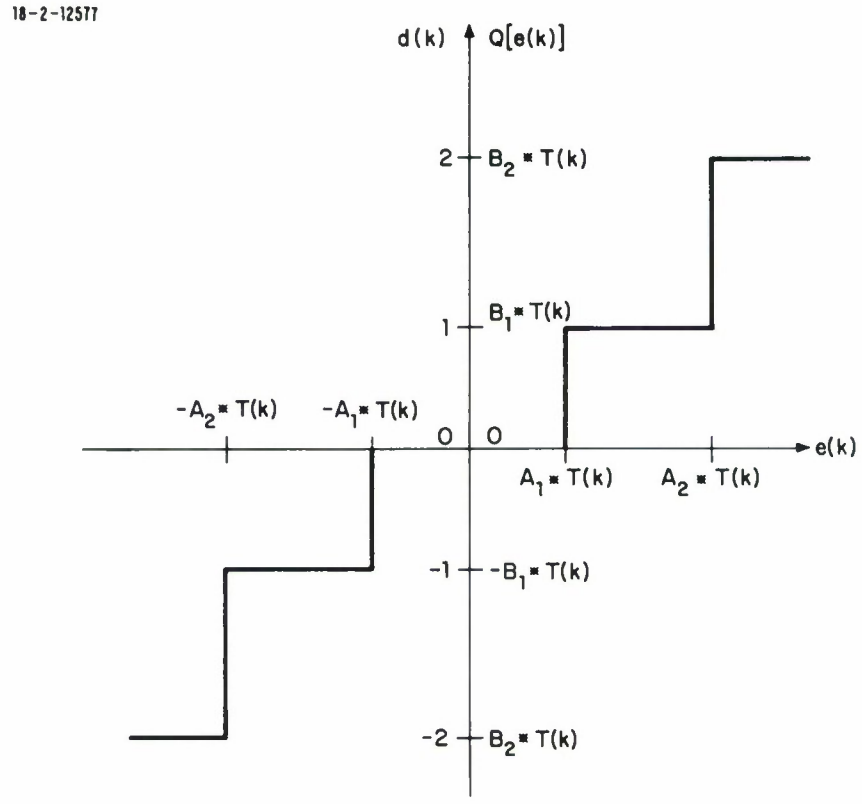


Fig. 14. The ARC quantizer.

To prevent overflow and underflow of either buffer, a bit count is maintained at both the transmitter and receiver. These bit counts are inspected every sample period to maintain stable buffers. The parallel-to-serial and serial-to-parallel converters receive and transmit 16-bit words, and the two buffers must at all times be in a state to accommodate the converters. When the possibility of buffer underflow is detected at the transmitter, a unique filler code is detected at the transmitter buffer which will eventually be discarded at the receiver. If there is danger of buffer overflow at the transmitter, the outer slice levels (which are represented by the longest code words) are truncated to the adjacent inner levels until this danger has passed, thereby degrading the speech signal but maintaining word synchronization. The receiver responds to impending buffer overflow by reading and discarding one additional code word. If the receiver buffer does not contain enough bits to represent a filler code plus the longest code word (i.e., buffer underflow may occur), no bits are read and the zero slice level is used. With an error-free channel and a modem clock with little or no drift, the transmitter buffer should never overflow, and the discard of codes at the receiver should occur only during silence. Channel errors, which may or may not cause loss of word synchronization, will result in degradation of speech quality; but the predictors and the quantizers at the transmitter and receiver will decay during silence, and synchronization of transmitter and receiver parameters should be restored.

The Adaptive Residual Coder as implemented in the LDVT occupies 454 program locations and uses 179 data locations. The worst case estimate of processing time is 64 μs per sample, or less than a fourth percent of real-time.

V. SUMMARY

An easily programmed [16], integrated real-time speech processor was designed and fabricated within 15 months. Five* different speech compression systems, spanning bit rates from 2400 to 16,000 bps, have been implemented successfully and undergone exhaustive test and evaluation. Support software developed includes a full diagnostic system [15]and an offline assembler written in Fortran to maximize compatibility with varying host facilities. The LDVT has proven by direct measurement to be 20 to 60 percent faster than real-time depending on the complexity of the algorithm simulated. It has been fully demonstrated that a compact, high performance processor can be replicated practically and at reasonable cost, making available to the speech research community a potentially exciting new class of experimental tool.

*2, quite different TRIVOC systems have actually been coded.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the contributions of: A.J. McLaughlin and S.D. Pezaris to the minicomputer design; J. Tierney to the analog peripheral system design; V. Sferrino to the digital peripheral system design and overall system fabrication and debugging; J. Drinan and R. Tucker to the support software system preparation; and B. Gold to the various vocoder algorithm designs. We also acknowledge the many helpful discussions we have had with C.P. Smith of the Electronic System Division and Ronald Sonderegger of the Defense Communications Agency.

References

1. B. Gold et al., "The FDP, a Fast Programmable Signal Processor," IEEE Trans. Computers C-20, 33-38 (1971), DDC AD-728092.

2. W.R. Blood, MECL System Design Handbook (Motorola, Inc., 1971).

3. P.E. Blankenship, "Preliminary Investigation of Digital Speech Processor Hardware Implementations," Technical Note 1975-8, Lincoln Laboratory, M.I.T. (5 February 1975), DDC AD-A007062/3.

4. B.S. Atal and S.L. Hanauer, "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave," Acoust. Soc. Am. 50, (1971).

5. J.D. Markel et al., "Linear Prediction of Speech - Theory and Practice," SCRL Monograph No. 10, Speech Communications Laboratory, Inc., Santa Barbara, CA (September 1973).

6. J.D. Makhoul and J.J. Wolf, "Linear Prediction and the Spectral Analysis of Speech," BBN Report No. 2304, Bolt, Beranek and Newman, Inc. Cambridge, MA (August 1972).

7. J.D. Markel and A.H. Gray, Jr., "A Linear Prediction Vocoder Simulator Based Upon the Autocorrelation Method," IEEE Trans. Acoustics, Speech, and Signal Processing ASSP-22, (1974).

8. N. Wiener, Extrapolation, Interpolation and Smoothing of Stationary Time Series (The Technology Press and J. Wiley and Sons, New York, 1957), Appendix B.

9. M.L. Malpass, "The Gold-Rabiner Pitch Detector in a Real-Time Environment," to be presented at EASCON Conference, Washington, DC, 29 September-1 October 1975.

10. B.S. Atal and M.R. Schroeder, "Adaptive Predictive Coding of Speech Signals," Bell Sys. Tech. J. 49, 1973-1968 (1970).

11. A.J. Goldberg and H. Shaffer, "Low Data Rate Voice-Communication Using Small Computers," Proceedings of the IEEE Communications Systems and Technology Conference, Dallas, TX, April 1974.

12. J.E. Roberts, C.P. Smith, and R.H. Wiggins, "Triple-Function Voice Coder (TRIVOC)," Acoust. Soc. Am. 57, Supp. 1, 535 (1975).

13. P.R. Drouilhet and L.M. Goodman, "Pole-Shared Linear-Phase Band-Pass Filter Bank," Proc. IEEE 54, 701 (1966), DDC AD-642206.

14. S.U.H. Qureshi and G.D. Forney, "A 9.6/16 KBPS Speech Digitizer," ICC 75 Conference Record, San Francisco, CA, Session 30.

15. R.I. Tucker, "The Lincoln DVT Diagnostic System," (to be published).

16. R.I. Tucker, "The Lincoln DVT Programming Manual," (to be published).

APPENDIX A

LDVT Instruction List

Mnemonic	Action	Execution Time
LDA/LDAX	$[A] \leftarrow [M_D]$	T
LDB/LDBX	$[B] \leftarrow [M_D]$	T
LDP/LDPX	$[P] \leftarrow [M_D]$	T
LDX/LDXX	$[X] \leftarrow [M_D]$	T
STA/STAX	$[M_D] \leftarrow [A]$	T
STB/STBX	$[M_D] \leftarrow [B]$	T
STP/STPX	$[M_D] \leftarrow [P]$	T
STX/STXX	$[M_D] \leftarrow [X]$	
ADDA/ADDAX	$[A] + [M_D] \rightarrow [A]$	T
ADDP/ADDPX	$[P] + [M_D] \rightarrow [P]$	T
ADDX/ADDXX	$[X] + [M_D] \rightarrow [X]$	T
SUBA/SUBAX	$[A] - [M_D] \rightarrow [A]$	T
SUBP/SUBPX	$[P] - [M_D] \rightarrow [P]$	T
SUBX/SUBXX	$[X] - [M_D] \rightarrow [X]$	T
MULI/MULIX	Bits 0-15 of $[A] \times [M_D] \rightarrow [A]$	4T
MULF/MULFX	Bits 15-30 of $[A] \times [M_D] \rightarrow [A]$	4T
MULD/MULDX	Bits 14-29 of $[A] \times [M_D] \rightarrow [A]$	4T
MULH/MULHX	Bits 16-31 of $[A] \times [M_D] \rightarrow [A]$	4T
STPLA	Lower byte of last product $\rightarrow [A]$	T
AAND/AANDX	$[A] \cap [M_D] \rightarrow [A]$	T
AOR/AORX	$[A] \cup [M_D] \rightarrow [A]$	T
AXOR/AXORX	$[A] \otimes [M_D] \rightarrow [A]$	T
CMPA	$[\overline{A}] \rightarrow [A]$	T
ADDAD/ADDADX	$[A] + [M_D] + C_{save} \rightarrow [A]$	T
SUBAD/SUBADX	$[A] + [\overline{M_D}] + C_{save} \rightarrow [A]$	T
LDAYP	$000000 + IR_{0-9} \rightarrow [A]$	T
LDAYN	$176000 + IR_{0-9} \rightarrow [A]$	T
DBA	$2 \cdot [A] \rightarrow [A]$	T
HVA	$2^{-1} \cdot [A] \rightarrow [A]$	T
QTA	$2^{-2} \cdot [A] \rightarrow [A]$	T
DBX	$2 \cdot [X] \rightarrow [X]$	T
HVX	$2^{-1} \cdot [X] \rightarrow [X]$	T
YIX	$Y \rightarrow [X]$	T
IOS	Initiate I/O transfers	T
STAMP/STAMPX	$[A] \rightarrow [M_P (Y + [X])]$	2T
JP/JPX/JPS/JPKS	$Y \rightarrow [P]$	T
JPZA/JPZAK/JPZAS/JPZAKS	$Y \rightarrow [P]$ if $[A] \geq 0$	T
JNA/JNAK/JNAS/JNAKS	$Y \rightarrow [P]$ if $[A] < 0$	T
JPZX/JPZXK	$Y \rightarrow [P]$ if $[X] \geq 0$, $[X]-1 \rightarrow [X]$	T
JNX/JNXK	$Y \rightarrow [P]$ if $[X] < 0$, $[X]+1 \rightarrow [X]$	T
JIR/JIRK/JIRS/JIRKS	$Y \rightarrow [P]$ if input transfer ready	T
JOR/JORK/JORS/JORKS	$Y \rightarrow [P]$ if output transfer ready	T
JOV/JOVK/JOVS/JOVKS	$Y \rightarrow [P]$ if overflow flag set	T
JSW/JSWK/JSWS/JSWKS	$Y \rightarrow [P]$ if sense switch W set	T
JSV/JSVK/JSVS/JSVKS	$Y \rightarrow [P]$ if sense switch V set	T
IJP	$[M_D(1)] + Y \rightarrow [P]$	T
IOIJP	$[M_D(2)] + Y \rightarrow [P]$	T
HLT	stop execution	T

Notes:

1. T = 55 nanoseconds

2. Suffix X appended to a mnemonic signifies that M_D address is Y + [X], otherwise it is Y + 0.

3. $[M_D\ (0)] = 0$. Thus an "LDA 0" clears A, etc.

4. Suffix S appended to jump code signifies that return point is to be saved, i.e., $[P] + 1 \rightarrow [M_D\ (1)]$.

5. Suffix K appended to jump code signifies suppression of the next subsequent operation. Transfer time is effectively 2T in this case.

6. Machine NO-OP is a "STA 0."

Integrated Circuits

ITEM #	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	4		MC 10100	Motorola		.99	3.96
	40		10101		334-101	.99	39.60
	17		10102		384-102	.99	16.83
	6		10103			.99	5.94
	5		10104			.99	14.85
	9		10105		384-105	.99	8.91
	5		10106		384-106	1.55	7.75
	3		10107		384-107	1.24	3.75
	6		10109		384-109	1.67	10.02
	27		10110		384-110	2.09	56.43
	12		10111		384-111	1.24	14.88
	4		10113			1.24	4.96
	1		10117		384-117	2.09	2.09
	5		10121		184-121	1.24	6.20
	5		10124		384-124	2.66	13.30
	15		10125		384-125	2.66	39.90
	18		10131		384-131	3.33	59.94
	1		10135			5.33	5.33
	8		10139	Signetics		15.97	127.76
	3		10161	Motorola	384-161	3.21	9.63
	44		10164		384-164	3.21	141.24
	1		10171			3.21	3.21
	35		10173			4.74	165.90
	9		10174			3.21	28.89

Integrated Circuits

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	37		MC 10176		384-176	5.28	195.36
	5		10180			13.77	68.85
	12		10181		384-181	18.02	216.24
	7		10210			2.26	15.82
	3		10211			2.26	6.78
	4		10216	Motorola		1.92	7.68
	12		10231			6.78	81.36
	5		10237			14.38	71.90
	64		10405	Fairchild		15.00	960.00
	64		10410			21.00	1344.00
	2		10116	Motorola	384-116	.99	1.98
	32		5304	Intersil		45.00	1440.00
	4		7400		383-000	.26	1.04
	3		7402		383-002	.265	.80
	5		7404		383-004	.325	1.63
	4		7408		383-008	.345	1.38
	2		7410		383-010	.25	.50
	1		7411		383-011	.345	.35
	4		7474		383-074	.592	2.37
	19		74153		383-153	1.48	28.12
	1		74157		383-157	1.024	1.02
	3		74163		383-163	2.92	8.76
	22		74265		383-165	.92	20.24

Integrated Circuits

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	1		MC 74S74		383-774	1.49	1.49
	1		74S138		383-838	6.64	6.64
	32		93415	Fairchild		28.00	896.00
	6		TDR 1084	Engineered Components	735-100	15.10	90.60
							6262.15

PC & Wire-wrap panels, power supplies

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	2	panel, wirewrap, ECL	8136-ECL21-180-2	Augat		558.00	1116.00
	1	panel, wirewrap, ECL	8136-ECL25-162-2	Augat		558.00	558.00
	1	panel, wirewrap, T^{2}L	8136-PG5-180-2	Augat		344.75	344.75
	1	power supply, 5 volt	62605 J	Hewlett-Pac.		445.00	445.00
	1	power supply, 5 volt	LCS-C-5-OV	Lambda		190.00	190.00
	1	power supply, 2 volt	LCS-CC-2	Lambda		240.00	240.00
	1	overvoltage protector	LM-OV-1	Lambda		30.00	30.00
	1	PC board, control panel		LL		17.50	17.50
	4	bus bar, laminated PC		LL		22.50	90.00
	4	PC board, thermostat		LL		3.00	12.00
	7104 wires	wirewrap - wire & instal.		LL		.084	596.74
							3639.99

Resistors, Capacitors

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	9	capacitor, 68 MFD	T330C686M006AS	Kemet	260-268	1.70	15.30
	15	capacitor, 1 MFD	CS13BF105K	Kemet	274-101	.27	4.05
	555	capacitor, .01 MFD Ceram	CK13BX103K	MIL	230-510	.277	153.74
	181	resistor network, SIP	750-81-150	CTS		.90	162.90
	168	resistor network, SIP	750-81-100	CTS		.90	151.20
	2	resistor network, SIP	750-83-100	CTS		1.25	2.50
	2	resistor network, 330/220DIP	898-5-R220/330	Beckman		2.30	4.60
	6	resistor network, 330 DIP	898-1-R330	Beckman		1.25	7.50
	6	capacitor, taut 2.2 MFD	CS13BE225K	Kemet	273-122	.27	1.62
	6	resistor, 1/8W,3.9K,5%	BB3925	Allen-Bradley	133-439	.095	.57
	73	resistor, 1/8W,1K, 5%	BB1025	Allen-Bradley	133-410	.095	6.94
							510.92

Components

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	1	line cord, AC	17408 black	Belden	614-341	1.30	1.30
	1	bushing, strain relief	SR-6P3-4	Heyman	850-634	.0332	.03
	1	fuse holder, 3AG	342004	Littelfuse	469-515	.49	.49
	1	fuse, 3AG 5A SB	313005	Littelfuse	463-850	.172	.17
	1	switch, SPDT moment	1PB5	Microswitch	547-412	2.85	2.85
	1	relay, solid state AC	MSR 100B	Monsanto	502-100	17.00	17.00
	420'	wire, #30 solid twisted	UL#1422	Prestolite	609-222	.02	8.40
	44'	cable, flat braid, 1/4"	1231	Alpha	649-008	.1378	6.06
	4	bumper, rubber 3/4"	2517W	Atlantic India Rubber	990-075	.101	.40
	19	indicator, solid state	5082-4860	Hewlett Packard	407-516	.55	10.45
	4	adapter plug, 16 pin	616-AG1	Augat		1.51	6.04
	4	terhmostat, Klixon	3BTL2-38	Texas Inst.		7.72	30.88
	1	Filter, EMI	5K1	Corcom		6.40	6.40
	1	switch, 5PST Toggle	82600	Arrow-Hart		1.68	1.68
	1	assembly, key lock	EXA-112	Chicago Lock		5.51	5.51
	3	connector, 88 pin, BLKHD	WSS0088P00BN 500	Hughes		12.00	36.00
	3	connector, 88 pin Jackscrew	WSS0088S08BN 001	Hughes		15.93	47.79
	3	Seal, 88 pin interfacial	WAC0088G000	Hughes		5.07	15.21
	300	contacts, female crimp	WS22Y28C000	Hughes		.4387	131.61
	300	contacts, male crimp	WP22Y28C000	Hughes		.2591	77.73
	3	shell, connector	G435-1A	Glen Air		9.24	27.72
	2274'	wire, #28 stranded twisted		Wirecraft		.0535	121.66
	6'	cable, 26 cond. ribbon	3365/26	3M		.2106	1.26
	34	connector, 26 pin ribbon	3399-0000	3M		2.36	80.24

Components

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM	
	2	connector, 26 pin twist pr	3399-1000	3M		2.98	5.96	
	25	interfacing plug, 14 pin	1P14-1	Augat		1.45	36.25	
	12	interfacing plug, 16 pin	1P16-1	Augat		1.55	18.60	
	40'	tubing, PVC 1/4"	FIT-105-1/4	Alpha		.3006	12.02	
	12'	tubing, PVC 3/16"	F1T-105-3/16	Alpha		.241	2.89	
	40'	tubing, PVC 3/4"	F1T-105-3/4	Alpha		.4986	19.94	
	2	connector, 44 pin	H-856	Berg		8.00	16.00	
	1	bus interface card	DR-11C	Digital Equip.		387.00	387.00	
	1	oscillator, crystal	CO-233 MET	Vectron		130.00	130.00	
	43	lead socket, 2-wrap	LSG-1FG10-1	Augat		.104	4.47	
	151	lead socket, no wrap	LSG-1DG4-1	Augat		.065	9.82	
	50	solder clip	8136-592P2	Augat		.035	1.75	
	4	blower, 3 5/8"	76099 XM	Amphenol		22.85	91.40	
	2	finger guard	760-9901-43	Amphenol		1.20	2.40	
	1	switch, 8 pos. rotary	9A45-03-1-8N	Grayhill		21.10	21.10	
	1	knob, skirted	RB-67-0-SK-7-M	Rogan		1.36	1.36	
	3	swithc, DPDT moment toggle	7205SYZBC/7099	C&K		2.36	7.08	
	19	swithc, SPDT toggle	7101SYZBE/7099	C&K		1.31	24.89	
	1 pr.	slides, drawer	110QD-24-1	Jonathan		17.51	17.51	
	1	cable retractor	CRS-25	Jonathan		15.92	15.92	
	1	Hex alum. spacer 3"	8433	H.H. Smith	861-093	.3187	.32	
	3	Hex alum. spacer 2"	8431	H.H. Smith	861-092	.22	.66	
	14	spacer, nylon 3/4x1/4	8158-N-0632	Amatom		.2014	2.82	
	1	switch, DIP, 4 pole, 2 pos	DS16A VARI	ACI	547-120	8.22	8.22	1475.26

Mechanical Package Parts

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	1	control panel		LL		45.00	45.00
	1	front panel				31.00	31.00
	1	rear panel				38.50	38.50
		Honeycomb, Alum	AL-3/16-5052-0.002N-5.7	Hexcel		12.50	12.50
	2	Honeycomb, Filter		LL		10.00	20.00
	1	front blower mount				25.50	25.50
	8	panel hinge block				2.50	20.00
	1	drawer bottom, 3/32 alum				8.75	8.75
		channel alum, 6"				20.31	20.31
		hatch hinge				.88	.88
		perforated alum, 1/32				3.70	3.70
	1	board spacer, G10 glass				2.37	2.37
	4	board spacer tab				1.20	4.80
		misc. parts drafting design, fab. labor		LL		57.41	57.41
							290.72

Integrated Circuits

Signal Conditioner

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM	
	1	operational amplifier	A741	Fairchild	379-041	1.25	1.25	
	1	transistor, PNP	2N3503	Fairchild	367-503	1.52	1.52	
	1	transistor, NPN	2N2219A	Fairchild	367-219	.30	.30	
	6	operational amp, dual	A747	Fairchild		6.75	40.50	
	1	converter, A/D	ADC-12QZ	Analog Dev		129.00	129.00	
	1	converter, D/A	H8-12B	Datel		89.00	89.00	
	1	amplifier, sample & hold	SHA-5	Analog Dev		47.00	47.00	
	3		7400	TI	383-000	.26	.78	
	4		7404		383-004	.325	1.30	
	14		7408		383-008	.345	4.83	
	1		7410		383-010	.25	.25	
	1		7411		383-011	.345	.35	
	3		7414			4.16	12.48	
	4		7474		383-074	.592	2.37	
	4		74163		383-163	2.92	11.68	
	4		74164		383-164	1.47	5.88	
	4		74165			4.50	18.00	
	4		74174		383-174	1.85	7.40	
	3		74265		383-265	.92	2.76	
	1		74390			3.39	3.39	
	1		DM8520	National	383-320	8.75	8.75	
	2		8T15	Signetics		6.75	13.50	
	3		8T16	Signetics		5.40	16.20	
	1		TDR1084	Engineered Components	735-100	15.10	15.10	433.59

PC & Wire-wrap panels, power supplies

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	1	panel, wirewrap	8136-RG4-60	Augat	855-804	85.00	85.00
	1	power supply, 5VG 3A	2BT-3	Powertec	445-503	28.80	28.80
	1	module, overvoltage prot.	OVP-1	Powertec	445-905	7.20	7.20
	1	socket, power supply	MS-7	Datel		3.50	3.50
	1	power supply, ±15V	BPM-15/200	Datel		69.00	69.00
	1	P.C. Board, Audio pwr amp		LL		10.00	10.00
	1	P.C. Board, Analog		LL		15.00	15.00
	2	P.C. Board, filter module		LL		10.00	20.00
	514 wires	wirewrap wire & install.		LL		.084	43.18
							281.68

Resistors, Capacitors

Signal Conditioner

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	1	volume control, 10K	1202-LB-10K	Vishay	170-310	7.41	7.41
	1	resistor, variable, 5K	62PR5K	Beckman	177-450	1.66	1.66
	1	resistor, variable, 20K	62PR20K	Beckman	177-520	1.49	1.49
	2	resistor network, 330 DIP	898-1-R330	Beckman		1.25	2.50
	1	resistor, 1%, 300	MF4C-D-301-F	Mepco/Elec.	153-301	.115	.12
	1	500	-499-F		153-499	.09	.09
	2	600	-604-F		153-604	.115	.23
	2	1K	-1000-F		154-100	.09	.18
	1	1.5K	-1500-F		154-150	.09	.09
	1	5K	-4990-F		154-499	.09	.09
	16	10K	-10.0K-F		155-100	.09	1.44
	1	20K	-20.0K-F		155-200	.09	.09
	4	40K	-40.2K-F		155-402	.09	.36
	2	52K	-52.3K-F		155-523	.228	.46
	1	150K	-150K-F		156-150	.09	.09
	2	17.8K	-17.8K-F		155-178	.16	.32
	2	18.7K	-18.7K-F		155-187	.228	.46
	2	capacitor, Mica 24Pf	DM10C-240-J	EL-Menco	200-224	.12	.24
	2	62Pf	DM10E-620-J		200-262	.13	.26
	4	91Pf	DM10F-910-J		200-291	.152	.61
	4	200Pf	DM10F-201-J		200-320	.20	.80
	2	300Pf	DM15F-301-J		200-330	.15	.30
	4	360Pf	DM15C-361-J		200-336	.33	1.32

Resistors, Capacitors

Signal Conditioner

ITEM	QUANTITY/ SYSTEM	DESCRIPTION		MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	2		430Pf	DM15F-431-J		200-343	.37	.74
	2		620Pf	DM19F-621-J		200-362	.21	.42
	4		750Pf	DM19F-751-J		200-375	.53	2.12
	6	capacitor, Mica	1000Pf	DM19F-102-J	El-Menco	200-410	.26	1.56
	14		1500Pf	DM19F-152-J		200-415	.27	3.78
	2		2200Pf	DM19F-222-J		200-422	.44	.88
	2	Capacitor, Mylar	.01MFD	232A1B103J	Electrocube	244-510	.57	1.14
	2		.015	232A1B153J		244-515	.84	1.68
	4		.03	232A1B303J		244-530	.66	2.64
	1		.47	232A1B474J		244-647	1.00	1.00
	1		.01			251-510	.11	.11
	2		.02			251-520	.21	.42
	3	capacitor, tant	22	CS13BF226K	Kemet	272-222	.40	1.20
	1	capacitor, mylay	1000Pf	MR-330-.001nf	Paktron	243-410	.473	.47
								38.77

Components

Signal Conditioner

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	2	Spacer hinged 1/4 x 1	354	Keystone	861-114	.34	.68
	4	bumper, rubber 7/16"	1925W	Atlantic India Rubb.	990-043	.02	.08
	1	transformer speaker	SO-15P	UTC	701-315	12.34	12.34
	1	indicator, solid state	6039-004-304	Drake	407-505	1.39	1.39
	2	receptable, 2-pin	1074-4	Kings	811-021	3.34	6.68
	2	plug, 2-pin	1075-1	Kings	811-001	2.37	4.74
	1	Bushing, strain relief	SR-30-1	Heyman	850-636	.0332	.03
	1	line cord, AC	2074-4-beige	GE	614-313	1.064	1.06
	1	transformer, wideband	SP-69	Triad	702-169	9.48	9.48
	1	transformer shield	SP-310	Triad	702-910	.70	.70
	204'	wire, #26 solid twisted	UL#1422	Prestolite	609-216	.0244	4.98
	1	fuseholder, 3AG	342004	Littelfuse	469-515	.49	.49
	1	fuse, 3AG 1A SB	313001	Littelfuse	463-810	.252	.25
	2	connector, 88 pin blkhd	WSS0088P00BN500	Hughes		12.00	24.00
	2	connector, 88 pin jackscrew	WSS0088S08BN001	Hughes		15.93	31.86
	2	seal, 88 pin interfacial	WAC 0088 G 000	Hughes		5.07	10.14
	200	contacts, female crimp	WS22428C000	Hughes		.4387	87.74
	200	contacts, male crimp	WP22Y28C000	Hughes		.2591	51.82
	50'	shielding, mesh	05-0860-0602	Metex		.32	16.00
	2	shell, connector	G435-1A	Glen Air		9.24	18.48
	280'	wire, #24 stranded twisted		Harvey		.065	18.20
	2	connector, 15 pin blkhd	17-20150-1	Amphenol		1.95	3.90
	2	connector, 15 pin cable	17-10150-1	Amphenol		2.55	5.10
	2	backshell, metal	17-311-01	Amphenol		1.13	2.26

Components

Signal Conditioner

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	2	assy, jacksocket & screw	17-895	Amphenol		1.68	3.36
	1	handset	Trendline 20000/BA/30M	ITT		54.00	54.00
	1	cord, sprial handset	8409	Belden		1.91	1.91
	1	jack handset	RA1.304NYL U/6.2	Lemo		5.63	5.63
	1	plug, handset	F1.304NYL U/6.2	Lemo		7.06	7.06
	1	Switch, toggle AC	7101 PCBE	C & K		1.80	1.80
	1	meter, VU	MM1	Jewell		32.48	32.48
	1	speaker, 2"	99F60329	Lafayette		1.75	1.75
	208	lead socket, no wrap	LSG-1DG4-1	Augat		.065	13.52
	50	fastener, steel 2-56	S-256-3	Electronic Fasteners		.0364	1.82
	4	spacer, hex alum 1/4x1/4	8421	H.H. Smith	861-082	.08	.32
	4	spacer, rd alum 1/4x1/2	3487	Keystone		.0675	.27
	6	spacer, nylon 1/4x3/4	8158-N-0632	Amatom		.2014	1.21
	2	spacer rd alum 1/4x1	3489	Keystone		.11	.22
	4	spacer rd alum 1/4x1 1/2	3490	Keystone		.15	.60
	1	switch, DIP, 4 pole 2 pos	DS16A VARI	ACI	547-120	8.22	8.22
	27	contact, filter mod, female	NS-441-D1	Robinson-Nugent		.34	9.18
	27	contact, filter mod, male	CP-95	Robinson-Nugent		.28	7.56
							463.31

Mechanical Package Parts

Signal Conditioner

ITEM	QUANTITY/ SYSTEM	DESCRIPTION	MFR. PART#	MFR.	LL#	COST/ UNIT	COST/ SYSTEM
	1	front panel				20.00	20.00
	1	rear panel				38.00	38.00
		perforated alum. 1/32				3.30	3.30
		Misc. parts drafting design, fab. labor				23.15	23.15
							84.45

Distribution List:

Caldwell P. Smith (ESD) (12 copies)

Hardware Subcommittee (12 copies)

Dir. Nat. Security Agency
Fort Meade, Maryland 20755
Attn: S4E

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ESD-TR-75-253	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* The Lincoln Digital Voice Terminal System		5. TYPE OF REPORT & PERIOD COVERED Technical Note 6. PERFORMING ORG. REPORT NUMBER Technical Note 1975-53
7. AUTHOR*(s)* Blankenship, Peter E. Hofstetter, Edward M. Huntoon, Albert H. Malpass, Marilyn L. Seneff, Stephanie Sferrino, Vincent J.		8. CONTRACT OR GRANT NUMBER*(s)* F19628-76-C-0002
9. PERFORMING ORGANIZATION NAME AND ADDRESS Lincoln Laboratory, M.I.T. P.O. Box 73 Lexington, MA 02173		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Program Element No. 33401F Project No. 7820
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Systems Command, USAF Andrews AFB Washington, DC 20331		12. REPORT DATE 25 August 1975 13. NUMBER OF PAGES 74
14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)* Electronic Systems Division Hanscom AFB Bedford, MA 01731		15. SECURITY CLASS. *(of this report)* Unclassified 15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

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

18. SUPPLEMENTARY NOTES

None

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

Lincoln Digital Voice Terminal
speech processing systems
vocoder systems
speech compression

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

An ultra-high performance programmable speech processor, consisting in the main of a custom designed 55-nsec microcomputer, has been designed and constructed. To date, five real-time speech compression programs have been implemented and evaluated.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*