ESD-TR-65-94 TM-03913

THE MUSE SYSTEM: DESCRIPTION AND MANUAL FOR OPERATION

TECHNICAL REPORT NO. ESD-TR-65-94

DECEMBER 1965

A. W. Slawson

OFFICE OF SCIENTIFIC AND TECHNICAL INFORMATION
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

Project 1700

Prepared by

THE MITRE CORPORATION
Bedford, Massachusetts
Contract AF19(628)-2390

ESD-TR-65-94 TM-03913

THE MUSE SYSTEM: DESCRIPTION AND MANUAL FOR OPERATION

TECHNICAL REPORT NO. ESD-TR-65-94

DECEMBER 1965

A. W. Slawson

OFFICE OF SCIENTIFIC AND TECHNICAL INFORMATION
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

Project 1700
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract AF19(628)-2390

FOREWORD

Robert Curtis, Edward Bensley, and Ferrell Sandy all contributed important ideas during the planning and programming of the MUSE Program. James Valentine and Augustine Kish designed and built the digital-to-analog converter. Professor K. N. Stevens of MIT provided very helpful criticism and the use of his sound spectrograph. Drs. F. S. Cooper and A. M. Liberman of Haskins Laboratory lent encouragement and valuable critiques. The author is much in debt to these and many other individuals.

ABSTRACT

The MUSE system, an IBM 7090 computer program and associated conversion equipment, has been designed for use as a sound synthesizer. Concise descriptions of complex sounds including human speech are converted by the MUSE system into sound pressure waveforms. The inputs to the MUSE system are specifications of the changing resonance frequencies of multiple acoustic filter networks and of the changing frequencies and amplitudes of the sources of acoustic energy that excite those networks. The output of the MUSE system is a sampled waveform calculated for each resonance by the solution of a second-order difference equation. The results are summed over a single system of resonances and then the resonance systems are also added together. The resulting string of sampled waveform ordinates is written in digital form on magnetic tape. Conversion to a voltage waveform is accomplished by use of the standard IBM 729 IV tape transport unit and a simple digital-to-analog converter. Although the quality of the sound is somewhat degraded by tape wow and flutter, acceptable and highly intelligible speech has been synthesized.

REVIEW AND APPROVAL

Edward M. Doherty

EDWARD M. DOHERTY
Chief, Scientific and Technical
Information Division

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

SECTION I

INTRODUCTION

A phonetician, wishing to test his model of speech production, often describes some speech utterance in terms dictated by his model and then uses this description to control a speech synthesizer. Provided the speech synthesizer is a good one, the quality of the resulting speech is a good indication of the sufficiency of his model. The model may be too complicated or may not be limited by physiological constraints, but if high quality speech can be synthesized according to the model, it commands attention as a possible means of gaining insight into that most complicated and refined biological system — the human vocal apparatus.

The MUSE system is a computer simulation of a class of sound synthesizers that have been used with success in testing theories of speech production. In spite of some loss of fidelity, the computer simulation can be used in place of this class of sound synthesizers. MUSE, consisting of an IBM 7090 computer program and simple digital-to-analog conversion equipment, translates concise descriptions of a large class of complex sounds, including human speech, into the corresponding analog waveform. This signal can be recorded for later playback or it can be used to drive a loudspeaker for immediate presentation of the sound.

Sound synthesizers can also be used to present information in the form of spoken messages to the human operators of a computer-centered, real-time control system.[1] In such an application, these messages would be stored in computer memory in concise digital form. The main program for the control system would select some appropriate message and send it to a sound synthesizer that would then expand the concise form of the message into the corresponding wide-bandwidth speech signal and present it to the operator in real time. The MUSE system, by simulating the output of various alternative synthesizers, could aid in designing the simplest adequate device for each application.

The MUSE system is itself a weak theory of speech production. MUSE is a weak theory because it leaves unspecified the manner in which the sounds are produced and because, subject to a limited bandwidth, it can reproduce the output of a large class of acoustical devices. The limitations that make it a theory at all are mainly practical ones. MUSE couldn't be used in practice to simulate a symphony orchestra because a single chord could require thousands of statements in the input language. General statements about these limitations are impossible but they will be made explicit by the detailed description of the input language and the operation of MUSE both of which follow.

SECTION II

EQUIPMENT NECESSARY AND OVER-ALL SCHEME

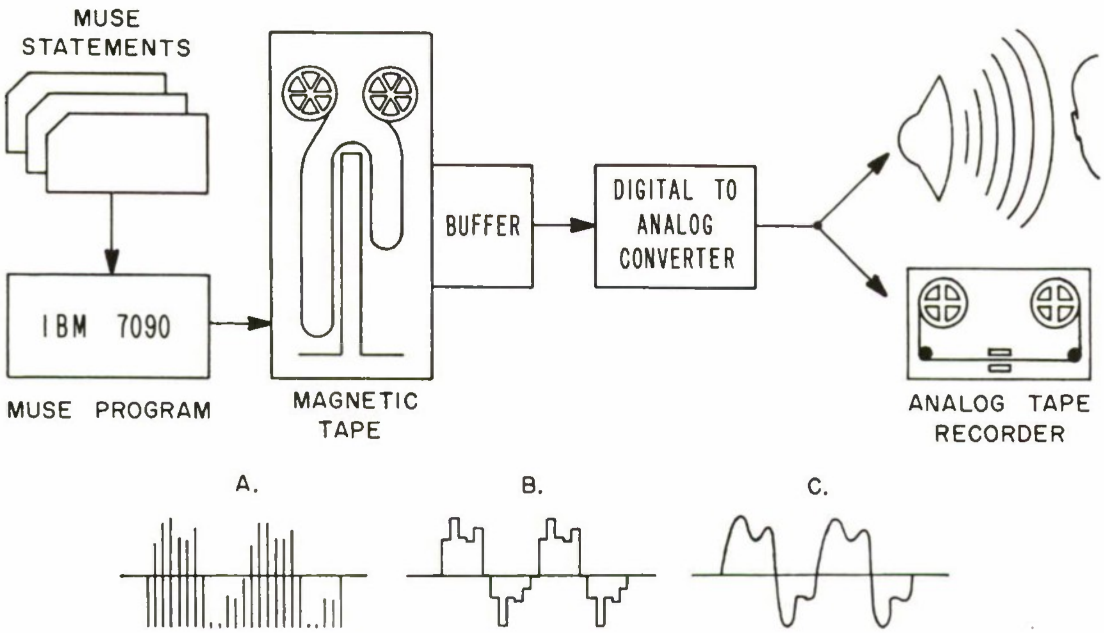
EQUIPMENT NECESSARY

The equipment needed for running the MUSE program consists of a standard IBM 7090 EDPM with at least two data channels and at least one 729 IV tape transport unit, a special purpose digital-to-analog converter, a variable bandpass filter, and any electro-acoustic transduction system or, preferably, a good quality magnetic tape recorder.

OVER-ALL SCHEME

In general, the operation of the MUSE system consists of a translation or calculation phase and a subsequent digital-to-analog conversion phase. In the first phase, the data cards containing the sound specifications are read in as needed and the ordinates of the specified waveform are calculated. These numbers representing the instantaneous pressure of the specified waveform at successive small intervals of time are stored in blocks on magnetic tapes. When calculation is completed for the sound sample or utterance being synthesized, the waveform ordinates are read from the magnetic tape storage into the computer for normalization. These data are then written onto a new tape in which the inter-record gaps between the blocks of numbers are eliminated.

In the conversion phase the tape unit, an IBM 729 IV tape transport unit on which the final output has been written, is disconnected from the data channel and connected to the digital-to-analog conversion device. This device then reads the six-bit (64 levels) waveform ordinates off the magnetic tape, sets them in a buffer register, and converts them to a voltage waveform smoothed by a low-pass filter. The resulting signal can be transduced by a standard loudspeaker system or recorded on magnetic tape as shown in Figure 1.



1. MUSE Statements on IBM cards are read into the computer.
2. The computed waveform ordinates (A) are written onto magnetic tape.
3. The waveform ordinates are converted into a step function (B) representing voltage levels.
4. A low-pass filter smooths the step-function into an analog waveform (C).
5. The output of the digital-to-analog converter is a varying voltage which drives a loudspeaker or tape recorder.

Figure 1. Functional Diagram of the MUSE System

SECTION III

SOUND SPECIFICATION LANGUAGE AND FORMATS

INTRODUCTION

Since the common denominator of all sound synthesis systems is an output waveform, the important differences between these systems lies in the method of specifying the desired sound and in the faithfulness with which these specifications are embodied in the output waveform. The level of sophistication or, in other terms, the degree of bandwidth compression of the system is more or less fixed by the specification language. The MUSE system's specification language consists of statements describing the changing acoustic spectra of the desired sound. Representing the instantaneous frequency response of independent resonating systems at given points in time, these statements contain the resonance frequencies and bandwidths of the several variable resonators that make up these systems.

SOUND SPECIFICATIONS: DATA CARDS

In the description of sound for MUSE, resonators are grouped so that several of them can be excited by the same energy source. These groups are called Spectra. A specification of the states of the resonators in a Spectrum and the excitation function for those resonators at some point in time is called a Statement. Whenever "Spectrum" and "Statement" are used below in this technical sense, they will be capitalized.

Spectrum Specification

The experimenter describes the instantaneous frequency response of a particular Spectrum by supplying the resonance frequency and bandwidth in cycles per second of each resonance contributing to that Spectrum. More explicitly, the resonance frequency refers to the frequency of a pole in a passive electrical network which is analogous to the acoustical system to be simulated. The bandwidth of the resonance controls the real component or the attenuation of that particular pole. The value of a resonance bandwidth is the difference between the two frequencies at which the attenuation of the resonant network is 3 db greater than at the pole frequency under the assumption that the network has only this single resonance.

Excitation Source Specification

Supplying the states of resonances, as described above, can specify an acoustic resonating system at a fixed point in time. If enough points are picked, or if points of inflection are chosen and the program is designed to interpolate between them, a fairly complete description of the continuing response of say the human vocal tract, uttering speech can be made. To excite the resonant system, however, some provision must be made for an excitation source.

In the MUSE language, energy is supplied to the resonant system in the form of a train of shaped pulses. The pulses, whose response characteristics are fixed at the beginning of the run, can excite the system at periodic or pseudo-random intervals. The mode of excitation, buzz or noise, is specified in each Statement. The fundamental frequency of the buzz source and its amplitude are also specified. When the noise option is used, the repetition rate must also be supplied since the pulses are shaped in terms of decibels per octave above the fundamental frequency. The fundamental frequency is given in cycles per second. The amplitude multiplier is a two-digit number specifying the relative amplitude of the source pulse (it is not a logarithmic quantity).

Timing Specification

Having specified the Spectra and their excitations for single points in time, these points must be fixed at particular times by entering in each

Statement the time in hundredths of a second since the beginning of the sound sample.

It has been mentioned above that sparse specification of the variation of a Spectrum through time would suffice if some kind of interpolation is assumed. The translation phase of the MUSE program assumes that all variables in a Statement (except specifications of time and mode of excitation) change linearly between Statements. The values of each variable in successive Statements and the time interval between these Statements determines the rate of this change. If a variable has the same value in successive Statements, that particular variable remains constant throughout the interval between those two Statements.

Sorting the Specification Cards

In the process of running the program, each Statement is read in as needed. Since interpolation between Statements is called for, it is necessary that as the calculations reach the "time" value of one Statement, the next Statement in each Spectrum must be read into the computer. The Spectra are independent of each other so ordering the Statements when specifying multiple Spectra can be fairly involved.

Although it is not necessary for running the system, it is recommended that a sorting field, not read by the computer, be included in the data cards. This sorting field contains in order a single-digit field that is zero ("0") for the first card in a Spectrum and a one ("1") for all other cards, the time value from the previous statement in the Spectrum, and the Spectrum number.

Data Card Formats

The format of the data cards whose fields have been described above are as given in Table I.

TABLE I

FORMAT OF A SOUND SPECIFICATION DATA CARD

Field or Variable Name	Column
Mode of excitation and card type identification: "0" = periodic excitation; "1" = noise excitation	1
Number of Spectra ("1" through "9")	2
Time (in the form XXXX. XX sec)	3-8
Amplitude Multiplier (XX arbitrary units)	9-10
Fundamental Frequency (XXXXX cps)	11-15
Resonance Specifications for Resonance Number i (i = 1, 2 ..., 8), where bandwidth = b_i (XXX cps), frequency = f_i (XXXX cps):	
b_1	16-18
f_1	19-22
b_2	23-25
f_2	26-29
b_3	30-32
f_3	33-36
b_4	37-39
f_4	40-43
b_5	44-46
f_5	47-50
b_6	51-53
f_6	54-57
b_7	58-60
f_7	61-64
b_8	65-67
f_8	68-71
Sorting Field	
"0" for first card in Spectrum, "1" for all others	73
Time from Previous Card in this Spectrum	74-78
Spectrum Number	79

Any resonances that are not used can be left blank.

At this time, a MUSE Statement is presented in its entirety on a single card. Possible future expansions of the input language to multiple cards justify the use of "Statement" as a special term.

CONTROL CARDS

There are three control cards that must be used in the operation of the MUSE program. These are not acoustic data but serve to set up parameters and options for the processing of the sound specifications.

Parameter Control Card

The first control card is identified by a "2" punched in Column 1. This card specifies for the entire set of data which follows it, the number of spectra, the number of resonances in each of these spectra, the response characteristics of the source function, the sampling period of the output waveform and a field controlling program options.

The number of spectra depends upon the number of independently excited resonant systems desired (for speech synthesis the question of number of Spectra is discussed under "Selection of Spectrum Parameters"). In the present version of the program, all spectra have the same number of resonances, hence, only one entry is necessary in the "2" control card. The response characteristics of the source functions in all Spectra are the same and are specified in terms of the slope of their frequency response in db per octave above the fundamental. The sampling period is usually fixed at 44×10^{-6} seconds, the program option field at "00."

The format of the first control card is as given in Table II.

Table II

Format of MUSE Control Card, Type "2"

Field or Variable Name	Column
Identification Field (contains a "2")	1
Number of Spectra ("1" to "9")	2
Sampling Period (in secs x 10^{-6}, usually "000044"	3-7
Number of Resonances ("01 to "08")	8-9
Source Characteristic (in db per octave x 10^{-1}; for instance, "060" is 6db per octave)	10-12
Program Option (usually "00")	13-14

End Cards

There are two kinds of control cards which signal the end of a set of data. Any card with a "4" in Column 1 signals the program that there are no more Statements in this sound segment and that the run is over. A "5" in Column 1 indicates the end of this sound segment and, in addition, sets up the program to accept the specifications for another sound segment preceded by its initial control card. When either of these end cards is read, the program begins processing the computed samples for the conversion phase. About one second of silence is automatically included on the end of each sound segment to erase a section of the output digital tape. The silent interval prevents spurious bits immediately following the waveform samples on the output tape from ruining the synthesized sound segment during the conversion process.

SECTION IV

SPEECH SYNTHESIS WITH THE MUSE SYSTEM: AN EXAMPLE

INTRODUCTION

Although implicit in the description given in Section III, the process of using MUSE for synthesizing sound is best clarified by presenting an example. A complex example, the speech utterance "All's well that ends well," has been chosen so that all features of MUSE can be demonstrated.

This example represents approximately 2 seconds of speech. It took approximately 200 seconds of computer time to synthesize this example.

The first steps in the synthesis procedure are common to all so-called "terminal analog" synthesizers.[2,3,4] First the utterance is recorded and sound spectrograms are made.[5] Amplitude sections, displays of the Spectrum of the speech during some selected short interval of time, can be made to clear up ambiguous areas on the time-frequency-amplitude graph. An additional useful datum is an oscillograph of the utterance with the time scale such that the period of the fundamental and the over-all amplitude can be measured throughout the utterance. These data for the sample "All's well that ends well," from which an analysis of the utterance can be made, are presented in Figure 2.

Selection of Spectrum Parameters

The next step in the analysis procedure is to decide how many Spectra will be necessary to specify the sound sample to be synthesized. Perhaps the most important factor in making this decision for synthesizing speech is a heuristic one; i. e., how many Spectra will best fit the experimenter's view of the speech process? For the example presented here, three spectra are used. Each corresponds to a different mode of operation of the vocal apparatus. The

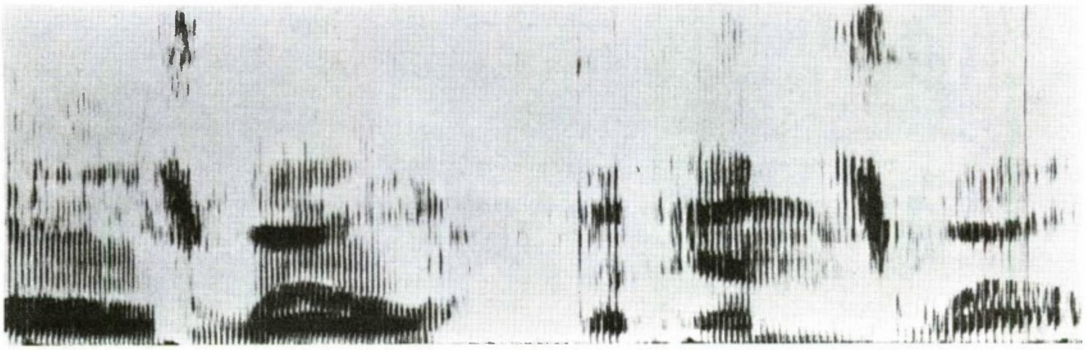


Figure 2a. Sonogram of the Utterance

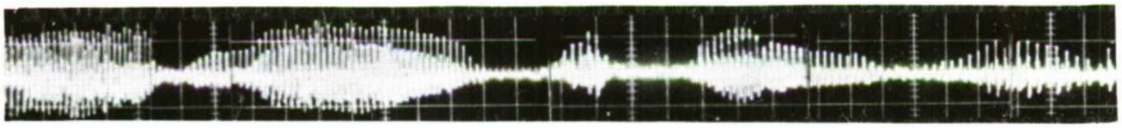


Figure 2b. Oscillogram of the Utterance

Figure 2. Spectrum of the Utterance, "All's well that ends well," Spoken by a Native American.

first spectrum corresponds to the mouth and throat excited by periodic laryngeal pulses. In other words, it is used for all voiced sounds. The second spectrum corresponds to any excitation of the vocal apparatus by noisy sources. Fricatives, aspirates, and some stops call for the use of this spectrum. The third Spectrum represents the contribution of the nasal cavities to nasalized vowels and consonants.

A similar division of the speech process into three more or less independent spectra is implied in the synthesizers built by Fant and K. N. Stevens.* A more concise description of speech utterances can be accomplished using only a single Spectrum with a corresponding degradation in the quality of the synthetic speech.

The other important decision to be made at this point in the analysis involves the number of resonances in the Spectra. Three resonances are a minimum number for intelligible speech. Higher frequency resonances contribute to the realism of the speech and may be important in realizing a particular speaker's voice. Since calculation of these extra resonances takes computer time, some restraint is to be exercised. In the present example, four resonances per Spectrum are specified. The resulting speech is highly intelligible but the speakers are generally not identifiable.

The slope of the power spectrum of the excitation source must be decided upon at this point also. Certain theoretical considerations by Fant[6] give 6 db per octave attenuation as a good empirical approximation to the over-all spectrum slope.

---

*These synthesizers are discussed in References [3] and [4] .

The values chosen for the number of Spectra, the number of resonances per Spectrum, and the slope of the source spectrum are entered into the "2" type control card as specified under "Parameter Control Card," page 9.

SEGMENTATION

Since MUSE interpolates between parameter values on successive statements, it is necessary to select times for the Statements between which the Spectrum parameters change only linearly. This is done by close examination of the spectrograms and oscillograms of the speech utterance. The segmentation is entirely acoustic and has little to do with phonetic divisions. An easy way to place the Statements and find parameter values is to plot the resonance frequencies for each Spectrum as a function of time. Figure 3 contains these plots for the example being considered here. The resonance values can be left constant or can be changed linearly when the amplitude multiplier is zero. (For example, see times 0.00 to 0.06, 1.98 to 0.20 in Figure 3.)

It can be seen from Figure 3 that the Spectra are time-independent; that is, the time segments in difference Spectra do not necessarily coincide. The values of the various parameters can be read from these graphs at the points corresponding in time to the Statements. The parameter values are then entered into a coding form and punched onto IBM cards in the format described under "Data Card Formats," page 7. After sorting the cards on the sorting field, the specification data are only ready for calculation. A printout data for these cards is given in Table III for the sample utterance.

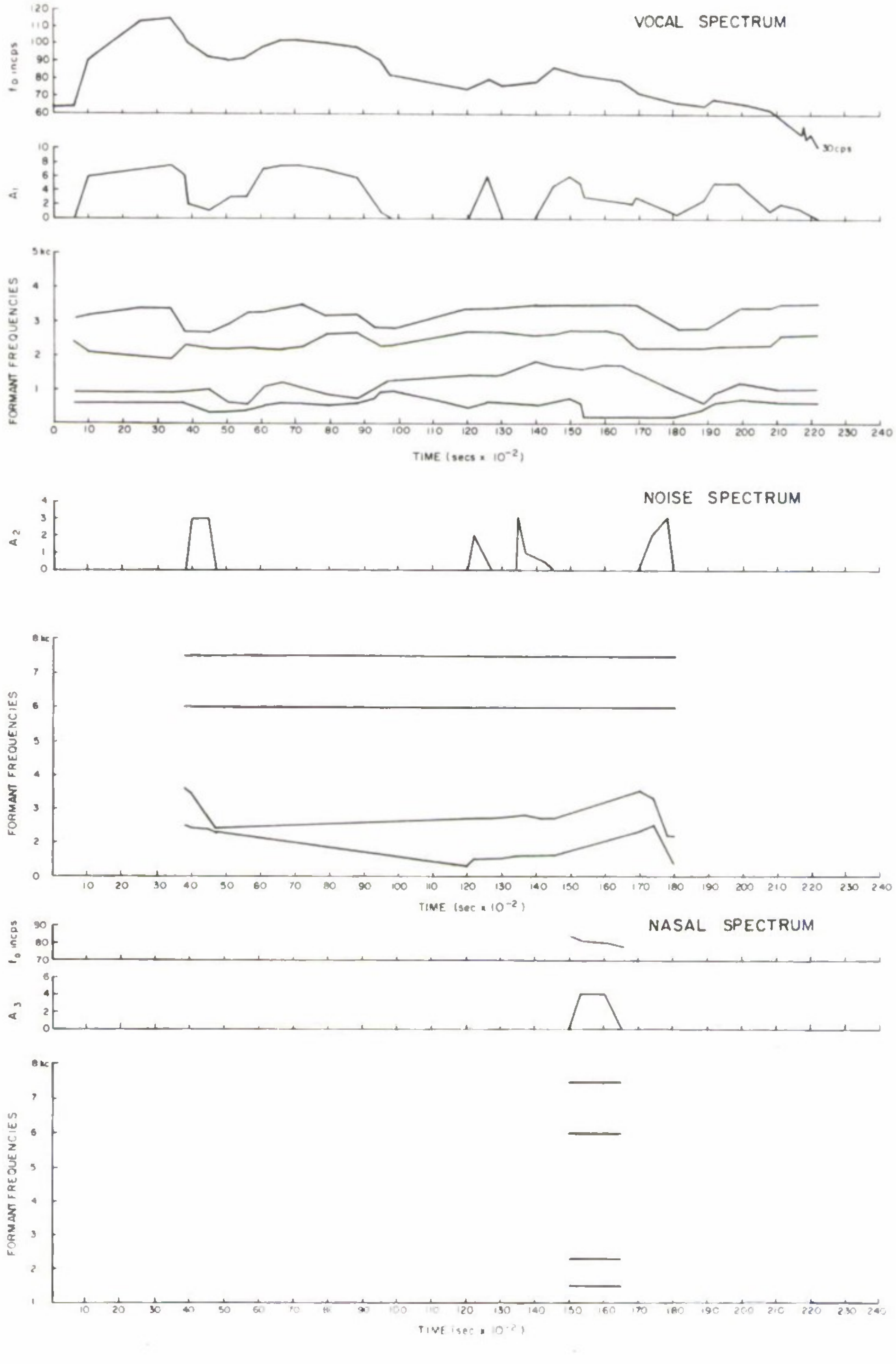


Figure 3. Tracings of the Sonograms (Figure 2a) from which the Input Data to MUSE are Copied. (The Vocal Spectrum Number is "1," the Noise Spectrum is "2," and the Nasal Spectrum is "3.")

Table III

Input Data For The Utterance "All's Well That Ends Well"

ID	Spectrum Number	Time	Amplitude	Fundamental Frequency	b_1	f_1	b_2	f_2	b_3	f_3	b_4	f_4
0	1	0	0	64	50	600	50	900	70	2400	120	3100
1	2	0	0	100	100	2500	60	3600	100	6000	100	7500
0	3	0	0	84	100	1500	90	2300	300	6000	400	7500
0	1	6	0	64	50	600	50	900	70	2400	120	3100
1	2	38	0	100	100	2500	60	3600	100	6000	100	7500
0	3	150	0	84	100	1500	90	2300	300	6000	400	7500
0	1	10	60	90	50	600	50	900	70	2100	120	3200
0	1	25	70	112	50	600	50	900	70	2000	120	3400
0	1	34	75	114	50	600	50	900	70	1900	120	3400
0	1	38	60	104	100	600	100	900	100	2300	200	2700
0	1	39	20	100	100	500	100	900	100	2300	200	2700
1	2	40	40	100	100	2400	60	3400	100	6000	100	7500
0	1	45	10	92	100	300	100	1000	100	2200	200	2700
1	2	45	40	100	100	2400	60	2700	100	6000	100	7500
0	1	51	30	90	70	300	80	600	100	2200	200	2900
1	2	47	0	100	60	2300	60	2400	100	6000	100	7500
1	2	120	0	100	80	1300	80	2700	150	6000	150	7500
0	1	56	30	92	60	400	70	500	100	2200	200	3300
0	1	61	70	98	40	500	50	1100	70	2200	100	3300
0	1	66	75	102	50	600	50	1200	70	2200	100	3400
0	1	72	75	102	50	600	50	1000	100	2300	150	3500
0	1	79	70	100	50	500	50	800	120	2600	200	3200
0	1	88	60	98	60	600	60	700	120	2700	120	3200
0	1	93	30	94	70	700	80	1000	100	2400	100	2800

Table III (cont'd)

ID	Spectrum Number	Time	Amplitude	Fundamental Frequency	b_1	f_1	b_2	f_2	b_3	f_3	b_4	f_4
0	1	95	10	90	70	900	150	1100	100	2300	100	2800
0	1	98	0	82	100	900	150	1200	100	2300	100	2800
0	1	120	0	74	100	500	150	1400	150	2700	150	3400
0	1	126	60	80	50	600	70	1400	90	2700	100	3400
1	2	122	20	100	80	1500	80	2700	150	6000	150	7500
1	2	127	0	100	80	1500	80	2700	150	6000	150	7500
0	1	128	35	78	50	600	70	1400	90	2700	100	3400
1	2	134	0	100	80	1600	80	2800	150	6000	150	7500
0	1	130	0	76	50	600	70	1400	90	2700	100	3400
0	1	140	0	78	50	500	70	1800	90	2600	100	3500
1	2	135	40	100	80	1600	80	2800	150	6000	150	7500
1	2	137	10	100	80	1600	80	2800	150	6000	150	7500
1	2	141	30	100	80	1600	80	2700	150	6000	150	7500
0	1	145	45	86	50	600	50	1700	90	2600	100	3500
1	2	145	0	100	80	1600	80	2700	150	6000	150	7500
0	1	150	60	84	50	700	50	1600	90	2700	100	3500
1	2	170	0	100	60	2300	50	3500	100	6000	100	7500
0	1	153	50	82	60	500	60	1600	90	2700	100	3500
0	3	153	40	82	100	1500	90	2300	300	6000	400	7500
0	1	154	30	82	90	200	100	1600	120	2700	150	3500
0	3	160	40	80	100	1500	90	2300	300	6000	400	7500
0	1	160	25	80	90	200	100	1700	120	2700	150	3500
0	1	165	25	78	90	200	100	1700	120	2600	150	3500
0	3	165	0	78	100	1500	90	2300	300	6000	400	7500
0	1	168	20	74	90	200	100	1600	120	2300	150	3500
0	3	222	0	78	100	1500	90	2300	300	6000	400	7500

Table III (Concl'd)

ID	Spectrum Number	Time	Amplitude	Fundamental Frequency	b_1	f_1	b_2	f_2	b_3	f_3	b_4	f_4
0	1	169	30	72	90	200	100	1500	120	2200	150	3500
0	1	181	5	66	90	200	100	1000	120	2200	150	2800
1	2	174	40	100	50	2500	50	3300	100	6000	100	7500
1	2	178	50	100	50	1700	50	2200	100	6000	100	7500
1	2	180	0	100	50	1400	50	2200	100	6000	100	7500
1	2	222	0	100	50	1400	50	2200	100	6000	100	7500
0	1	189	25	64	90	400	100	600	120	2200	150	2800
0	1	192	50	68	50	600	50	900	90	2300	100	3000
0	1	199	50	66	50	700	50	1200	90	2300	100	3400
0	1	208	10	62	50	600	50	1000	90	2300	100	3400
0	1	211	20	56	50	600	50	1000	90	2600	100	3500
0	1	216	15	50	50	600	50	1000	90	2600	100	3500
0	1	222	0	30	50	600	50	1000	90	2600	100	3500

EVALUATION

The resulting synthetic speech is highly intelligible if somewhat artificial sounding. Recordings have been rather widely demonstrated, with almost total comprehension of the utterances reported by the audience. A spectrogram and oscillogram of the original and synthesized versions of the utterance "All's well that ends well" are given in Figure 4.

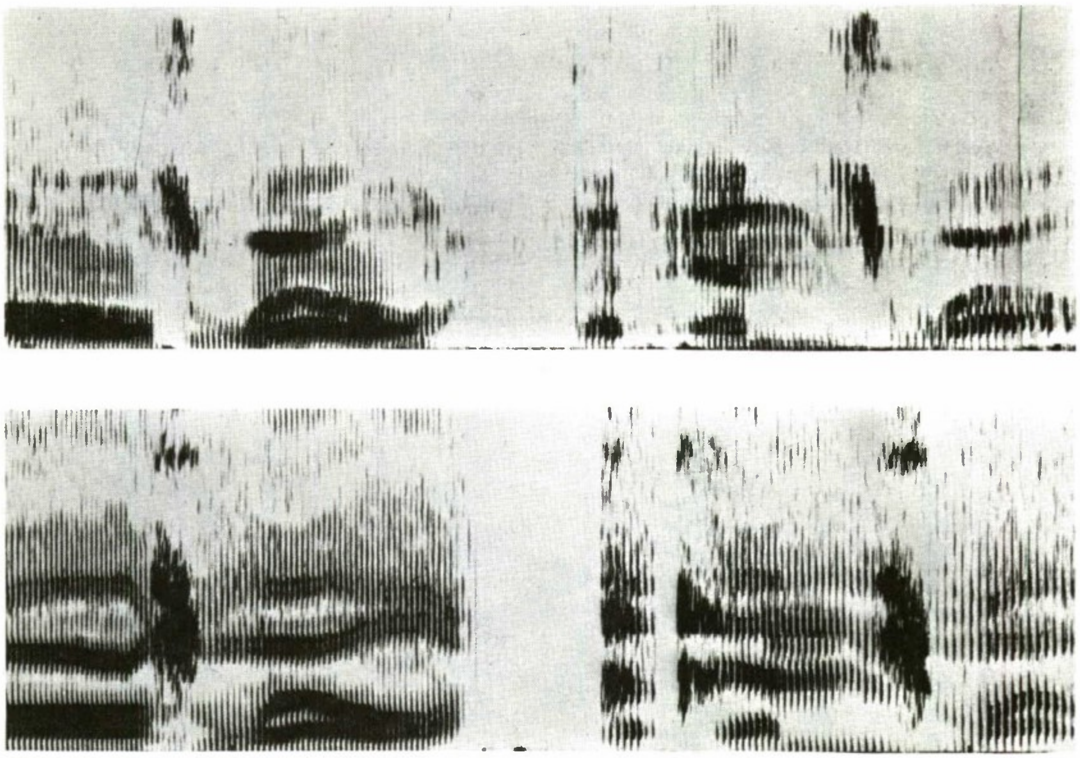


Figure 4a. Sonograms of the Original (Top) and Synthesized (Bottom) Versions of the Utterance

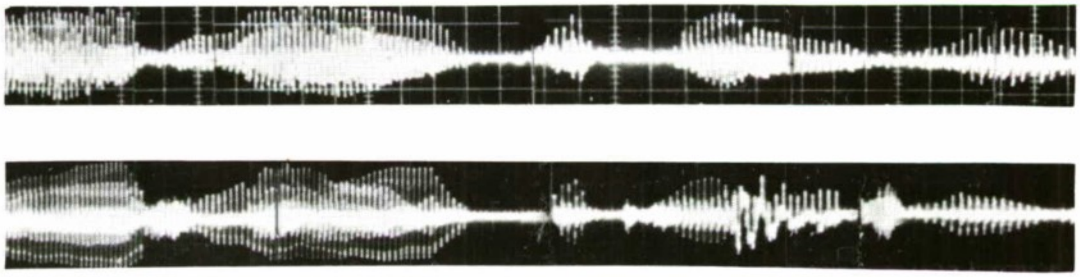


Figure 4b. Oscillograms of the Original (Top) and Synthesized (Bottom) Versions of the Utterance

Figure 4. Comparison of Sonograms and Oscillations of Original and Synthesized Utterance, "All's well that ends well."

SECTION V

FLOW OF THE PROGRAM

INPUT PROCESSING

In order to avoid a "hardware" limitation on the number of data cards or number of spectra, the data cards are read into the computer as needed by an input subroutine. When a new card is read, spectrum parameters are set up and an interpolation increment for each Spectrum parameter is calculated. These increments are added after each period of the excitation pulse. Although "continuous" interpolation (i. e., after each waveform ordinate) would more closely approximate the smooth changes in the vocal tract, substantial savings in computation time accrue if the interpolation is lumped in coordination with the source periods.

CALCULATION PHASE

The two (momentarily) constant coefficients of the following second-order difference equation are the recipients of the interpolation increments:

$$x_t = Ax_{t-\Delta t} + Bx_{t-2\Delta t} + S ,$$

where

$$A = \left[2e^{-2\pi\, b_{ij}\Delta t} \cos(2\pi f_{ij}\Delta t)\right] ,$$

$$B = -\left[e^{-4\pi b_{ij}\Delta t}\right];$$

and where

b_{ij} = bandwidth of the i^{th} resonance in the j^{th} spectrum,
f_{ij} = frequency of that resonance,
Δt = sampling period,

x_t = amplitude at time, t, and
S = relative amplitude of the source function.

Evaluation of this difference equation for successive sampling periods results in a series of ordinates of the waveform of a single resonance excited by a pulse of amplitude, S, once per period of the source function. At each sampling point, t, the ordinates of each resonance are summed, and these sums are added over all Spectra. The resulting over-all ordinate is

$$x_t = \sum_{j=1}^{\ell} \sum_{i=1}^{K} X_{ijt} ,$$

where

i = resonance index,
j = spectra index, and
ℓ and K = upper limits of the indices (the number of spectra and resonances given in the "2" control card).

As the calculation proceeds, the waveform ordinates are stored temporarily in blocks of 12,000 on a magnetic tape.

OUTPUT PROCESSING

When one of the final control cards (with either a "4" or "5" in Column 1) is encountered, the final output routine is begun. The blocks of waveform ordinates stored temporarily on tapes are read into the computer, scaled, packed and converted to a steady stream of six-bit numbers written as a single record in low density on the output tape. If the final control card is a "4," the program is finished and control is returned to the monitor system. If the "5" control card terminates the utterance, the program clears storage space and begins reading in the control and data cards for the next utterance.

SECTION VI

RUNNING THE PROGRAM AND CONVERTING THE OUTPUT DATA

INTRODUCTION

This section describes the preparation of the input deck for computation, the running of the program itself, and the operation of the digital-to-analog converter.

RUNNING THE PROGRAM

The data cards, after sorting on the field described under "Sorting the Specification Cards" on page 7, are preceded by the "2" control card, followed by the "4" or "5" control card and inserted behind the "*DATA" card following the MUSE program binary deck.

Tapes needed in running the program are B5 and B6 and A10. The B-channel tapes provide temporary storage while the final output is written on an A10. Under ordinary operation there are no stops in the program. Any on-line printouts are used by an observer only to keep track of the program's operation. They do not require action from the computer operators.

OUTPUT CONVERSION PROCESS

While the program's operation is quite routine, the output conversion process definitely is not (see Figure 1 for an over-all schematic of the conversion system). It is advisable to make the connections between the converter and the tape drive under the supervision of IBM customer engineers. Computer time is conserved if the tape drive containing the digital output tape is disconnected from the computer during a routine halt between two runs. An extra "terminator" for the detached tape drive must be attached to the converter. The bandpass filter on the output of the converter should be set at a nominal low-pass cutoff of about 10,000 cps.

The tape recorder should be started before the tape drive. A switch on the converter supplies the signals necessary for starting the digital tape drive. The conversion process then continues in real time. When conversion is finished, the digital tape can be stored for later use and the tape drive can be re-attached to the computer.

DIGITAL-TO-ANALOG CONVERSION EQUIPMENT

The digital-to-analog converter that was used in the MUSE system is only one of several circuits that would serve as a conversion device.

An IBM cable attachment plug is part of the converter used. The input to the read amplifiers comes from this plug. The equipment is most conveniently mounted on an ordinary rack with space provided for the bandpass filter used to smooth the output of the converter.

REFERENCES

1. A. W. Slawson, "MUSE — A Sound Synthesizer," Proc. of Int'l Fed. of Information Processing Congress 1962, Amsterdam, North-Holland, 451-455.

2. Franklin S. Cooper, et al, "The Interconversion of Audible and Visible Patterns as a Basis for Research in the Perception of Speech," Proc. Nat. Acad. Sci., 37 (1951), 318-325.

3. Quarterly Progress and Status Report 2/1961, Speech Transmission Lab., Royal Institute of Technology, Stockholm, Sweden, 16.

4. George Rosen, "Dynamic Analog Speech Synthesizer," J. Acoust. Soc. Am., 30 (1958), 201-209.

5. Potter, Kopp, and Green, Visible Speech, New York, Van Nostrand, 1947.

6. C. Gunnar M. Fant, Acoustic Theory of Speech Production, The Hague, Mouton, 1960.

7. Reference Manual, 709/7090 FORTRAN Programming System, Form J28-6087, International Business Machines Corp., White Plains, N. Y.

APPENDIX A PROGRAM LISTING

Following is the FORTRAN program listing of the MUSE Program.[7]

```
      W. SLAWSON   MUSE2, A SOUND SYNTHESIZER                                   4/10/63
C     ****************************************************************
C
      COMMON  FREQ,FAND,SREQ,SAND,DCOA,DCOB,VAL,TOT,STORE,PUTIN,FMIS,   MU200300
     XSMIS,CMIS,FCCA,FCCB,FORCE,PERIOD    ,IIN                          MU200400
     X, ISW2, J, SMALL, ISPEC, LCOUNT, TIME,IRESON, L, IMS2, NAR,
     XGREAT, SAMPER
     X  , STOR
     X,CUTL
     X,PULSE,CPULSE,DB
     X,PERTY
     X, OPTION
      DIMENSION   FREQ(  8, 10), FAND(  8, 10), SREQ(  8, 10), SAND(  8,MU200500
     X 10), DCOA(  8, 10), DCOB(  8, 10), VAL( 2,  8, 10), TOT( 10),    MU200600
     X STORE( 100), PUTIN( 21) ,  FMIS( 4, 10) , SMIS( 4, 10 ),         MU200700
     XCMIS(2,10), FCCA(8,10), FCCB(8,10),  FORCE(8,10)      ,
     X PERIOD ( 10 )     ,  IIN(2)                                      MU200900
     X,STOR(12000)
     X,CUTL(2000)
     X,PULSE(8,10),CPULSE(8,10)
     X,PERTY(10)
       EQUIVALENCE ( STORE, STOR )
C
C     ****************************************************************
C     HOUSEKEEP AND READ CONTROL CARD.                                  MU201100
C     ****************************************************************
C
 100  READ INPUT TAPE 5,10,IIN(1), ISPEC, SAMPER, IRESON, DB,  OPTION
 10   FORMAT  (2I1,F5.0,I2,F3.1 , F2.0         )
       IF ( OPTION - 2. ) 25,15,15
 15    PRINT  11
 11    FORMAT ( 1H1  )
       PRINT 12,IIN(1),ISPEC,SAMPER,IRESON, DB, OPTION
 12    FORMAT   (1H I2, I3, E15.7, I3,2E15.7)
 25    WRITE OUTPUT TAPE 6, 11
       WRITE OUTPUT TAPE 6, 12, IIN(1), ISPEC, SAMPER, IRESON, DB,OPTION
      SAMPER  =   SAMPER  *  10. **(-6        )
      LCOUNT = 0                                                        MU201300
      SMALL  = 0.                                                       MU201500
       GREAT=0.
       REWIND 8
      L      = 1                                                        MU201600
      ISW2   = 2                                                        MU201700
        NAR = 12000
        IF ( OPTION - 5. ) 45, 50, 60
 45    IF ( OPTION - 4. ) 60,57,60
 50    NAR = 999
       GOTO 60
 57    NAR=100
 60    DO  110  J=1,ISPEC
      DO 112  I=1,IRESON                                                MU202000
      VAL(1,I,J)    = 0.                                                MU202100
 112  VAL(2,I,J)    = 0.                                                MU202200
       PERTY(J)  =  0.
 110  TOT(J)        = 0.                                                MU202300
       DO 55 I=1,NAR
```

```
 55     STOR(I) = 0.
       TIME         = 0.                                                MU202400
C
C      *****************************************************************
C                                            END OF HOUSEKEEPING. SET
C                                            UP INITIAL CONDITIONS.
C      *****************************************************************
C
 115     DO 130 J=I, ISPEC
         J-J
        CALL SUB2( IMS2 )
 130     CALL SUB3
         DO 147 J=1, ISPEC
         J=J
        CALL SUB2( IMS2 )
         CALL SUB4
 147     CALL SUB5
        IF (OPTION )  150, 150, 149
 149    PRINT  4020
        WRITE OUTPUT TAPE 6, 4020
 4020    FORMAT  (  12H BEFORE 150.
        CALL  WDUMP
C
C      *****************************************************************
C      BEGIN MAIN FLOW                                                  MU202600
C      *****************************************************************
C
 150   J     = 1                                                        MU205300
 155   TOT(J)           0.0
 600   DO  620  I=1,IRESON                                              MU219800
       TEMP   =     (FCOA(I,J)    *    VAL(1,I,J))   -    FCOB(I,J)     MU219900
      X* VAL(2,I,J)   +  FORCE(I,J)
       FORCE(I,J) = 0.0
       VAL(2,I,J)     =     VAL(1,I,J)                                  MU220200
       VAL(I,I,J)     =    TEMP                                         MU220300
 620   TOT(J)    =      TOT(J)      +     TEMP
 1553   IF(SMIS(1,J) - TIME) 163,163,1552
 1552    IF (PERTY(J) - TIME ) 160, 160, 157
 157   IF  ( ISPEC - J )  159  , 159, 158
 158   J       =   J+1                                                  MU206100
       GO TO  155                                                       MU206200
C
 159   TEMP   =      0.0
       DO 820   J=1,ISPEC                                               MU223900
 820   TEMP     =TOT(J)+    TEMP                                        MU224000
        STOR(L)   =   TEMP
       TIME       =  TIME + 1.                                          MU220500
        L = L+1
        IF    (   NAR   -   L   )      1591, 150 , 150
C
 1591 CALL   SUB7
 1595 L=1                                                               MU206800
      GO TO  150                                                        MU206900
C
C      *****************************************************************
C                                            PERIOD OF THE FUNDAMENTAL
```

```
C                                                   IS OVER. INCREMENT PARAMET-
C                                                   ERS AND START A NEW PERICD.
C     ****************************************************************
 160  DC  920     I=1, IRESCN
      PULSE(I,J) =  PULSE(I,J) + OPULSE(I,J) * PERICO(J)
      FCCA(I,J)  =    FCCA(I,J) + DCCA( I,J )      *PERIOD(J)           MU226400
 920  FCCB(I,J)  =    FCCB(I,J) + DCOB( I,J )      *PERIOD(J)           MU226500
      FMIS(2,J)  =    FMIS(2,J) + OMIS(1,J)        *PERIOO(J)           MU226600
      FMIS(3,J)  =    FMIS(3,J) + CMIS(2,J)        *PERICD(J)           MU226700
 161  CALL SUB5                                                         MU207200
      GC TO 157                                                         MU207300
C
C     ****************************************************************
C                                                   PROCESS A NEW CARD
C     ****************************************************************
C
 163  CALL SUB3                                                         MU207500
 164  CALL SUB2( IMS2 )                                                 MU207600
      IF ( IMS2 - 1 )  167, 166, 165                                    MU207700
 167  CALL SUB4                                                         MU208300
       GC TC 1552
C
C     ****************************************************************
C                                                   ENO CARD HAS BEEN REACHEC.
C                                                   START FINAL PROCESSING.
C     ****************************************************************
C
 165   CC1653  IX = L,NAR
 1653  STCR (IX)  = 0.C
       CALL SUB7
       IF (CPTICN -4. ) 175,1651,1651
 175   CC 1751  IX = 1,NAR
 1751  STCR(IX)     = C.C
       LCCUNT = LCCUNT  +    6
       CC  1752  IX=1,6
 1752  CALL RITE(STCR)
       REWINC 8
       REWINC 1C
       DC 185 IX = 1,LCCUNT
 180   CALL REEC(STCR)
 5002 FORMAT  ( 1H E19.7, 5E20.7 )                                     19M9290
 185  CALL SUB1C (SMALL,GREAT,NAR,STOR,OUTL)
        ENDFILC 1C
 190  CALL SUB11                                                       19M9290
 1651 PRINT 4016
      WRITE CUTPUT TAPE 6, 4016
 4016 FCRMAT (12H END CF JOB.    )
      CALL WCUMP
1652  IF(IMS2-2) 192,192,191
 191   GCTO 100
 192   CALL EXIT
C
C     ****************************************************************
C     ERRCR    RCUTINES
C     ****************************************************************
C
```

```
166  PRINT  4017
     WRITE OUTPUT TAPE 6, 4017
4017 FORMAT ( 41H DATA CARD OUT OF ORDER. STOP EXECUTION.     )
     CALL WDUMP
     CALL EXIT
     END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

```
      SUB2, READ DATA CARD INTO  S(J) REGIONS.                                4/10/63

      SUBROUTINE SUB2(IRRS2    )                                           MU208800
      COMMON  FREQ,FANC,SREC,SAND,DCOA,DCOB,VAL,TOT,STORE,PUTIN,FMIS,      MU208900
     XSMIS,DMIS,FCCA,FCCB,FORCE,PERIOD    , IIN                            MU209000
     X, ISW2, J, SMALL, ISPEC, LCOUNT, TIME,IRESON, L, IMS2, NAR,
     XGREAT, SAMPER
     X   , STOR
     X,CUTL
     X,PULSE,CPULSE,CB
     X,PERTY
     X, OPTION
      DIMENSION   FREQ(  8, 10), FAND(  8, 10), SREQ(  8, 10), SAND(  8,MU209100
     X 10), DCOA(  8, 10), DCOB(  8, 10), VAL( 2,  8, 10), TOT( 10),       MU209200
     X STORE( 100), PUTIN( 21) ,  FMIS( 4, 10) , SMIS( 4, 10 ),            MU209300
     XDMIS(2,10), FCOA(8,10), FCOB(8,10),  FORCE(8,10)      ,
     X PERIOD ( 10 )     ,  IIN(2)                                         MU209500
     X,STOR(12000)
     X,CUTL(2000)
     X,PULSE(8,10),CPULSE(8,10)
     X,PERTY(10)
       EQUIVALENCE ( STORE, STOR )
 200  READ INPUT TAPE 5, 2000,  IIN(1), IIN(2),(PUTIN(M), M=1,19 )
       IF ( OPTION- 2. ) 260, 250, 250
 250   PRINT 2001, IIN(1), IIN(2), (PUTIN(M), M=1,19)
 2001  FORMAT  (1HC2I2, F7., F4., F8., 8(F5.,F6.))
 260   WRITE OUTPUT TAPE 6, 2001, IIN(1), IIN(2), ( PUTIN(M), M=1,19)
 201  IF (  IIN(1) - 2 )  210, 202, 205                                    MU209700
 202   PRINT 2002
       WRITE OUTPUT TAPE 6,2002
 2002  FORMAT (24H MISPLACED CONTROL CARD.        )
       CALL WDUMP
       CALL EXIT
 205  IF (  IIN(1) -3  )  206, 207, 245
 206  CALL WDUMP
      CALL EXIT
 207   PRINT 2003
       WRITE OUTPUT TAPE 6, 2003
 2003  FORMAT (50H CHANGE IN NUMBER OF RESONANCES. TO BE PROGRAMMED.  )
       CALL WDUMP
       CALL EXIT
 245   IF(IIN(1)-4) 208,208,240
 208  IRRS2 = 2                                                            MU210400
 209   RETURN
C     END CARD.                                                            MU210600
 210  IF (  IIN(2)  - J ) 215, 220, 215                                    MU210700
 215  IRRS2 = 1                                                            MU210800
      PRINT 4003
      WRITE OUTPUT TAPE 6, 4003
 4003 FORMAT ( 48H DATA CARDS FROM WRONG SPECTRUM. STOP EXECUTION.)
      CALL WDUMP
       CALL DUMP
 220   SMIS(2,J)  =  PUTIN(2)
 225   SMIS(3,  J)  =  1.  /  (  PUTIN(3)  *  SAMPER  )
       SMIS(1,J)  =  PUTIN(1)  / (100.  *  SAMPER)
      SMIS(4,J)       =     IIN(1)                                         MU211400
      M=4                                                                  MU211500
```

```
     SUB2, READ DATA CARD INTO  S(J) REGIONS.                              4/10/63

     N=5                                                                MU211600
     TWOPI  =  2.   *    3.14159265
     DO 230  I=1, IRESON                                                MU211700
     SREQ(I,J)     =  PUTIN( N    )       *  TWOPI                      MU211900
     SAND(I,J)     =  PUTIN( M )          *TWOPI                        MU211800
     N=N+2                                                              MU212000
 230 M=M+2                                                              MU212100
     IRRS2  = 0                                                         MU212200
      GOTO 209
C    NORMAL RETURN.                                                     MU212400
 240  IRRS2= 3
      GOTO 209
2000 FORMAT  (  2  I1   , F5.0 , F2.0 , F6.0,  8( F3.0,F4.0 )  )        MU212500
     END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

```
     SUB3, MOVE S(J) TO F(J) REGIONS.                                      4/10/63

      SUBROUTINE  SUB3                                                  MU212900
      COMMON  FREQ,FAND,SREQ,SAND,DCOA,DCOB,VAL,TOT,STORE,PUTIN,FMIS,   MU213000
     XSMIS,DMIS,FCOA,FCOB,FORCE,PERIOD  ,IIN                            MU213100
     X, ISW2, J, SMALL, ISPEC, LCOUNT, TIME,IRESON, L, IMS2, NAR,
     XGREAT, SAMPER
     X  , STOR
     X,OUTL
     X,PULSE,CPULSE,CH
     X,PERTY
     X, OPTION
      DIMENSION   FREQ(  8, 10), FAND(  8, 10), SREQ(  8, 10), SAND(  8,MU213200
     X 10), DCOA(  8, 10), DCOB(  8, 10), VAL( 2,  8, 10), TOT( 10),    MU213300
     X STORE( 100), PUTIN( 21) ,  FMIS( 4, 10) , SMIS( 4, 10 ),         MU213400
     XDMIS(2,10), FCOA(8,10), FCOB(8,10),  FORCE(8,10)     ,
     X PERIOD ( 10 )     ,  IIN(2)                                      MU200900
     X,STOR(12000)
     X,OUTL(2000)
     X,PULSE(8,10),CPULSE(8,10)
     X,PERTY(10)
       EQUIVALENCE ( STORE, STOR )
 300  ISW =1                                                            MU213700
       DO 310 M=1,4
 310  FMIS( M,J )  =   SMIS( M, J )                                     MU213900
      DO 320  I=1, IRESON                                               MU214000
      FAND( I,J )  =   SAND( I, J )                                     MU214100
 320  FREQ( I,J )  =   SREQ( I, J )                                     MU214200
      RETURN                                                            MU214300
      END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

SUB 4   COMPUTE COEFFICIENT INCREMENTS

```
      SUBROUTINE SUB4
C     MY END IS MY BEGINNING.  FIDDLE WITH THE MIDDLE COEFFICIENTS.     MU214700
      COMMON  FREQ,FANO,SREQ,SANO,DCOA,DCOB,VAL,TOT,STORE,PUTIN,FMIS,   MU214900
     XSMIS,DMIS,FCOA,FCOB,FORCE,PERIOD ,IIN                             MU215000
     X, ISW2, J, SMALL, ISPEC, LCOUNT, TIME,IRESON, L, IMS2, NAR,
     XGREAT, SAMPER
     X  , STOR
     X,OUTL
     X,PULSE,DPULSE,DB
     X,PERTY
     X, OPTION
      DIMENSION   FREQ(  8, 10), FANO(  8, 10), SREQ(  8, 10), SANO(  8,MU215100
     X 10), DCOA(  8, 10), DCOB(  8, 10), VAL( 2,  8, 10), TOT( 10),    MU215200
     X STORE( 100), PUTIN( 21) ,  FMIS( 4, 10) , SMIS( 4, 10 ),         MU215300
     XDMIS(2,10), FCOA(8,10), FCOB(8,10),  FORCE(8,10)    ,
     X PERIOD ( 10 )    ,  IIN(2)
     X,STOR(12000)
     X,OUTL(2000)
     X,PULSE(8,10),DPULSE(8,10)
     X,PERTY(10)
       EQUIVALENCE ( STORE, STOR )
 400  DTIME     =   SMIS(1,J)    -   FMIS(1,J)                          MU215600
      E =  2.7182818
      DO 410   I=1,IRESON                                               MU215700
      PULSE(I,J) =                FMIS(2,J) *  SINF(  FREQ(1,J) * SAMPER )
     X *(( FREQ(I,J) * ( FMIS(3,J) * SAMPER ) /  6.2831853)) ** (-DB *
     X   .166096  )
      DPULSE(I,J)= (   (          SMIS(2,J) *  SINF(  SREQ(1,J) * SAMPER )
     X *(( SREQ(I,J) * ( SMIS(3,J) * SAMPER ) /  6.2831853)) ** (-DB *
     X   .166096 ) )  - PULSE(I,J) ) / DTIME
      FCOA(I,J)    = (2.0 * E ** (-FANO(I,J) * SAMPER))   *COSF ( FREQ  MU215800
     X(I,J)  *       SAMPER )                                           MU215900
      DCOA(I,J)    =((2.0 * E **(-SANO  (1,J) *  SAMPER)  *  COSF (SREQ MU216000
     X(I,J) *        SAMPER)) -  FCOA(I,J)) /  DTIME                    MU216100
      FCOB(I,J)    =     E ** (-2.0 * FANO (I,J)   * SAMPER)            MU216200
 410  DCOB(1,J)    =  ( E ** (-2.0 * SANO(I,J) * SAMPER) - FCOB(I,J))   MU216300
     X  / DTIME                                                         MU216400
      DO 420  I=1,2                                                     MU216500
 420  DMIS(1,J)    =  (SMIS(I+1,J) - FMIS (1+1 , J) ) / DTIME           MU216600
      RETURN                                                            MU216700
      END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

SUB5, SET PERIOD AND FORCING FUNCTION 4/10/63

```
      SUBROUTINE SUB5
      COMMON  FREQ,FAND,SREQ,SAND,DCOA,DCOB,VAL,TOT,STORE,PUTIN,FMIS,   MU217200
     XSMIS,DMIS,FCOA,FCOB,FORCE,PERIOD ,IIN                             MU217300
     X, ISW2, J, SMALL, ISPEC, LCOUNT, TIME,IRESON, L, IMS2, NAR,
     XGREAT, SAMPER
     X  , STOR
     X,CUTL
     X,PULSE,DPULSE,DB
     X,PERTY
     X, OPTION
      DIMENSION   FREQ(  8, 10), FAND(  8, 10), SREQ(  8, 10), SAND(  8,MU217400
     X 10), DCOA(  8, 10), DCOB(  8, 10), VAL( 2,  8, 10), TOT( 10),    MU217500
     X STORE( 100), PUTIN( 21) ,  FMIS( 4, 10) , SMIS( 4, 10 ),         MU217600
     XDMIS(2,10), FCOA(8,10), FCOB(8,10),  FORCE(8,10)      ,
     X PERIOD ( 10 )     ,  IIN(2)
     X,STOR(12000)
     X,CUTL(2000)
     X,PULSE(8,10),DPULSE(8,10)
     X,PERTY(10)
       EQUIVALENCE ( STORE, STOR )
      IF ( FMIS(4,J) - 1.)  500, 510, 500                               MU217900
500    PERIOD(J)  =  FMIS(3,J)
      GO TO  520                                                        MU218100
510    DEWRAN = RDUN(DUMMY)
       DEWRAN=  (DEWRAN*(1./(1000.*SAMPER)))+(1./( 5000.*SAMPER))
517    PERIOD(J) = DEWRAN
520     PERTY(J)  =  PERTY(J) + PERIOD(J)
       DO 530 I=1, IRESON
530   FORCE(I,J) = PULSE(I,J)
      RETURN                                                            MU218500
      END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

```
      SUBROUTINE 7, WRITE BUFFER AREA                                      4/10/63

      SUBROUTINE SUB7
      COMMON  FREQ,FAND,SREQ,SAND,CCOA,CCOB,VAL,TOT,STORE,PUTIN,FMIS,   MU221100
     XSMIS,DMIS,FCOA,FCOB,FORCE,PERIOD ,IIN                             MU221200
     X, ISW2, J, SMALL, ISPEC, LCOUNT, TIME,IRESON, L, IMS2, NAR,
     XGREAT, SAMPER
     X  , STOR
     X,CUTL
     X,PULSE,CPULSE,DB
     X,PERTY
     X, OPTION
      DIMENSION   FREQ(  8, 10), FAND(  8, 10), SREQ(  8, 10), SAND(  8,MU221300
     X 10), CCOA(  8, 10), CCOB(  8, 10), VAL( 2,  8, 10), TOT( 10),    MU221400
     X STORE( 100), PUTIN( 21) ,  FMIS( 4, 10) , SMIS( 4, 10 ),         MU221500
     XDMIS(2,10), FCOA(8,10), FCOB(8,10),  FORCE(8,10)      ,
     X PERIOD ( 10 )    ,  IIN(2)
     X,STOR(12000)
     X,ISTOR(12000)
     X,CUTL(2000)
     X,PULSE(8,10),CPULSE(8,10)
     X,PERTY(10)
       EQUIVALENCE ( STORE, STOR )
     X,(STOR,ISTOR)
 700  LCOUNT   =  LCOUNT + 1                                            MU221800
       IF (  OPTION  - 4. ) 705, 707, 707
 705   CALL RITE(STOR)
       GOTO 725
 707   IF ( OPTION - 6. ) 720, 710, 720
 710  PRINT  5001
      PRINT  5002,  STORE
 720  WRITE  OUTPUT  TAPE 6,   5001
 5001 FORMAT ( 36H NAREA CONSECUTIVE WORDS OF OUTPUT.    )
      WRITE OUTPUT TAPE 6,  5002 ,  STORE
 5002 FORMAT  (  1H E19.7,   5E20.7    )
 725  DO  730   L=1,   NAR
       IF (  GREAT -STOR(L)  )  740,  750,  750
 740   GREAT  =  STOR(L)
      GO TO  730                                                        MU222300
 750   IF(  SMALL  -  STOR(L)  )  730,  730,  760
 760   SMALL  =  STOR(L)
 730  CONTINUE                                                          MU222600
 770  IF(SENSE SWITCH 5 ) 775,790
 775  DIFF = GREAT - SMALL
      DO 780  L=1,NAR
 780  ISTOR(L) = (((  STOR(L) - SMALL ) / DIFF ) * 512.) + 256.
      PRINT 7001
 7001 FORMAT(120H  THERE IS A REPEATING DISPLAY ON THE CRT. TO STOP IT,
     XPUT SWITCH 5 UP. AFTER HALT, SW5 UP TO STOP TV, DOWN TO SEE MORE.)
 790  RETURN
      END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

```
      SUBROUTINE WDUMP
      COMMON  FREQ,FAND,SREQ,SAND,DCOA,DCOB,VAL,TOT,STORE,PUTIN,FMIS,
     XSMIS,DMIS,FCOA,FCOB,FORCE,PERIOD ,IIN
     X, ISW2, J, SMALL, ISPEC, LCOUNT, TIME,IRESON, L, IMS2, NAR,
     XGREAT, SAMPER
     X  , STOR
     X,OUTL
     X,PULSE,CPULSE,DB
     X,PERTY
     X, OPTION
      DIMENSION   FREQ(  8, 10), FAND(  8, 10), SREQ(  8, 10), SAND(  8,
     X 10), DCOA(  8, 10), DCOB(  8, 10), VAL( 2,  8, 10), TOT( 10),
     X STORE( 100), PUTIN( 21) ,  FMIS( 4, 10) , SMIS( 4, 10 ),
     XDMIS(2,10), FCOA(8,10), FCOB(8,10),  FORCE(8,10)      ,
     X PERIOD ( 10 )    ,  IIN(2)
     X,STOR(12000)
     X,OUTL(2000)
     X,PULSE(8,10),CPULSE(8,10)
     X,PERTY(10)
       EQUIVALENCE ( STORE, STOR )
      WRITE OUTPUT TAPE 6, 3001
 3001 FORMAT (119H                    J                   L
     X  OPTION              ISPEC                 LCOUNT              IRESON)
      WRITE OUTPUT TAPE 6, 3002, ( J,L,OPTION, ISPEC,LCOUNT,IRESON )
 3002 FORMAT(1H I19, I20, F19.7, 3I20  )
      WRITE OUTPUT TAPE 6,3021
 3021 FORMAT (119H                  IMS2                    NAR
     X  TIME                SMALL                 GREAT               DB    )
      WRITE OUTPUT TAPE 6,3022, (IMS2, NAR, TIME, SMALL, GREAT,DB    )
 3022 FORMAT (1H I19,I20,4E20.7 )
      WRITE OUTPUT  TAPE 6, 3025
 3025 FORMAT  (11H IIN(X,J)        )
      WRITE OUTPUT TAPE 6, 3026,  IIN
 3026 FORMAT ( 1H I19, I20  )
      WRITE OUTPUT TAPE 6, 3013
 3013 FORMAT  (11H PUTIN(X)    )
      WRITE OUTPUT TAPE 6, 3004,   PUTIN
 3004 FORMAT  (  1H F19.7,5E20.7  )
      WRITE OUTPUT TAPE 6, 3003
 3003 FORMAT  (11H FREQ(I,J)   )
      WRITE OUTPUT TAPE 6, 3004, FREQ
      WRITE OUTPUT TAPE 6, 3005
 3005 FORMAT  (11H FAND(I,J)   )
      WRITE OUTPUT TAPE 6, 3004, FAND
      WRITE OUTPUT TAPE 6, 3006
 3006 FORMAT  (11H SREQ(I,J)   )
      WRITE OUTPUT TAPE 6, 3004, SREQ
      WRITE OUTPUT TAPE 6, 3007
 3007 FORMAT  (11H SAND(I,J)   )
      WRITE OUTPUT TAPE 6, 3004,  SAND
      WRITE OUTPUT TAPE 6, 3014
 3014 FORMAT  (11H FMIS(I,J)   )
      WRITE OUTPUT TAPE 6, 3004,  FMIS
      WRITE OUTPUT TAPE 6, 3015
 3015 FORMAT  (11H SMIS(I,J)   )
      WRITE OUTPUT TAPE 6, 3004,   SMIS
```

```
      SUBROUTINE WDUMP

      WRITE OUTPUT TAPE 6, 3016
 3016 FORMAT  (11H DMIS(I,J)    )
      WRITE OUTPUT TAPE 6, 3004,     DMIS
      WRITE OUTPUT TAPE 6, 3017
 3017 FORMAT  (11H FCOA(I,J)    )
      WRITE OUTPUT TAPE 6, 3004,  FCOA
      WRITE OUTPUT TAPE 6, 3018
 3018 FORMAT  (11H FCOB(I,J)    )
      WRITE OUTPUT TAPE 6, 3004,   FCOB
      WRITE OUTPUT TAPE 6, 3008
 3008 FORMAT  (11H DCOA(I,J)    )
      WRITE OUTPUT TAPE 6, 3004,  DCOA
      WRITE OUTPUT TAPE 6, 3009
 3009 FORMAT  (11H DCOB(I,J)    )
      WRITE OUTPUT TAPE 6, 3004,  DCOB
      WRITE OUTPUT TAPE 6, 3010
 3010 FORMAT  (11H VAL(X,I,J)   )
      WRITE OUTPUT TAPE 6, 3004,   VAL
      WRITE OUTPUT TAPE 6, 3011
 3011 FORMAT  (11H TOT(J)       )
      WRITE OUTPUT TAPE 6, 3004,  TOT
      WRITE OUTPUT TAPE 6, 3012
 3012 FORMAT  (11H STORE(L)     )
      WRITE OUTPUT TAPE 6, 3004,  STORE
      WRITE OUTPUT TAPE 6, 3019
 3019 FORMAT  (11H FORCE(I,J)             )
      WRITE OUTPUT TAPE 6, 3004,   FORCE
      WRITE OUTPUT TAPE 6, 3020
 3020 FORMAT  (11H PERIOD(J)    )
      WRITE OUTPUT TAPE 6, 3004,  PERIOD
      WRITE OUTPUT TAPE 6, 3023
 3023 FORMAT (15H  DPULSE(I,J)             )
      WRITE OUTPUT TAPE 6, 3004,  DPULSE
      WRITE OUTPUT TAPE 6, 3024
 3024 FORMAT (15H   PULSE(I,J)             )
      WRITE OUTPUT TAPE 6, 3004,   PULSE
         WRITE OUTPUT TAPE 6, 3027
 3027 FORMAT  ( 15H  PERTY(J)
         WRITE OUTPUT TAPE 6, 3004, PERTY
      RETURN
      END(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

```
        SUBROUTINE 10    SCALING AND PACKING ROUTINE

                                                      SCALING AND PACKING SUBROUTINE
                         00002           ENTRY    SUB10

  LINKAGE DIRECTOR
00000   000000000000
00001   626422010060

00002   0634 00 4 00060  SUB10   SXA      XSAVE,4
00003   0634 00 2 00061          SXA      XSAVE+1,2
00004   0634 00 1 00062          SXA      XSAVE+2,1
00005   0500 60 4 00003          CLA*     3,4
00006   0601 00 0 00077          STO      NAR
00007   0500 60 4 00002          CLA*     2,4
00010   0601 00 0 00066          STO      GREAT
00011   0500 60 4 00001          CLA*     1,4
00012   0601 00 0 00065          STO      SMALL
00013   0500 00 4 00004          CLA      4,4            LOCATION OF STOR
00014   0734 00 1 00000          PAX      ,1
00015   1 00001 1 00016          TXI      *+1,1,1
00016   0634 00 1 00037          SXA      L21,1
00017   0500 00 4 00005          CLA      5,4            LOCATION OF OUTL
00020   0734 00 1 00000          PAX      ,1
00021   1 00001 1 00022          TXI      *+1,1,1
00022   0634 00 1 00047          SXA      L23,1
00023   0634 00 1 00027          SXA      L24,1
00024   1 74060 1 00025          TXI      *+1,1,-2000
00025   0634 00 1 00064          SXA      IO1,1
00026   0774 00 2 03720          AXT      2000,2
00027   0600 00 2 50121  L24     STZ      OUTL,2
00030   2 00001 2 00027          TIX      *-1,2,1
00031   0500 00 0 00066          CLA      GREAT                COMPUTE DIFF
00032   0302 00 0 00065          FSB      SMALL
00033   0601 00 0 00067          STO      DIFF
                         *                                     END OF HOUSEKEEPING
00034   0774 00 1 00001          AXT      1,1
00035   0774 00 2 03720          AXT      2000,2
00036   0774 00 4 00006  L22     AXT      6,4
00037   0500 00 1 77462  L21     CLA      NARFA+1,1    SCALE AND PACK THIS RECORD
00040   0302 00 0 00065          FSB      SMALL
00041   0241 00 0 00067          FDP      DIFF
00042   0260 00 0 00072          FMP      SEVSIX
00043  -0300 00 0 00075          UFA      MAGIC
00044   0760 00 0 00011          FRN
00045  -0320 00 0 00074          ANA      MASK1
00046   0767 00 4 00044          ALS      36,4
00047  -0602 00 2 50121  L23     ORS      OUTL,2
00050   1 00001 1 00051          TXI      *+1,1,1
00051   1 00006 4 00052          TXI      *+1,4,6
00052  -3 00044 4 00037          TXL      L21,4,36
00053   2 00001 2 00036          TIX      L22,2,1
00054  +077600002225             OCT      077600002225   SDH   OUTTAP
00055   0766 00 0 02225          WTBB     OUTTAP
00056  -0540 00 0 00064          RCHB     IO1
00057   0061 00 0 00057          TCOB     *
00060   0774 00 4 00000  XSAVE   AXT      **,4
00061   0774 00 2 00000          AXT      **,2
```

```
        SUBROUTINE 10   SCALING AND PACKING ROUTINE

00062  0774 00 1 00000          AXT     **,1
00063  0020 00 4 00006          TRA     6,4
00064 -1 03720 0 44202  IO1     IOCT    OUTL-1999,,2000
00065  0 00000 0 00000  SMALL   PZE
00066  0 00000 0 00000  GREAT   PZE
00067  0 00000 0 00000  DIFF    PZE
00070 +000000000006     SIX     DEC     6
00071  0 00000 0 00000  LOSIX   PZE
00072 +206770000000     SEVSIX  DEC     63.0
00073 +176400000000     PFIVE   OCT     176400000000
00074 +000000000077     MASK1   OCT     77
00075 +233000000000     MAGIC   OCT     233000000000
00076 +000001000000     ONEDEC  OCT     1000000
00077  0 00000 0 00000  NAR     PZE
                 00005  OUTTAP  EQU     5
                 77461  NAREA   EQU     32561
                 50121  OUTL    EQU     20561
                                END
```

```
      SUBRCUTINE 11   WRITE A LONG  RECURD

                 C3722          FNTRY   SUB11
                                                SUBROUTINE TO WRITE A LCNG RECORD

  LINKAGE DIRECTCR
00000  0C0000C0CCCCC
00001  526422201C16C

                 CCC05  OUTTAP EQU     5
                 CCC05  FTAPE  EQU     5
C0002                   AREA   BSS     2000
                 C3720  L      FQU     2C00
03722  0634 00 2 C3751  SUB11  SXA     EXIT,2
03723  0772 C0 0 C2205         REWB    OUTTAP
03724 +0776CC002225            OCT     077600002225   SDHB  OUTTAP
03725 -0030 C0 0 C3726         TEFB    *+1
03726  0762 C0 0 C2225         RTBB    OUTTAP
03727 -0540 00 0 C3757         RCHB    I09
03730  C061 00 0 C3730         TCOB    *
03731 -0022 C0 0 C3732         TRCB    *+1
03732 +0776CC001205            OCT     077600001205   SDLA  FTAPE
03733  0766 00 0 C1225         WTBA    FTAPE
03734  0540 C0 0 C3757         RCHA    I09
03735  0640 00 0 C3756  DELAY  SCHA    T
03736  0534 C0 2 C3756         LXA     T,2
03737 -3 02736 2 C3735         TXL     *-2,2,AREA+L-N
03740  0762 00 0 C2225         RTBB    OUTTAP
03741 -0540 C0 0 C3757         RCHB    I09
03742 -0061 C0 0 03750         TCNB    END
03743 -0640 00 0 C3755         SCHB    S
03744  0534 C0 2 C3755         LXA     S,2
03745 -3 00004 2 C3742         TXL     *-3,2,AREA+2
03746  0544 00 0 03757         LCHA    I09
03747  0C20 00 C 03735         TRA     DELAY
0375C  C060 00 0 C3750  FND    TCOA    *
03751  0774 00 2 CCCC0  EXIT   AXT     **,2
03752 -0030 C0 0 03753         TEFB    *+1
03753  0020 00 4 CCC01         TRA     1,4
03754  0000 00 C CCC01  ERROR  HTR     1
03755  0 00000 C CCCC0  S      PZE
03756  0 C0C00 0 CC00C  T      PZE
                 CC764  N      EQU     500
03757 -1 03720 0 CCC02  I09    IOCT    AREA,,L
                               END
```

```
        COUNT    25

                      00002          ENTRY    REED

   LINKAGE DIRECTOR
 00000   000000000000
 00001   512525246060

 00002   0634 00 2 00012  REED    SXA      EXIT,2
 00003   0500 00 4 00001          CLA      1,4
 00004   0734 00 2 00000          PAX      ,2
 00005   1 50441 2 00006          TXI      *+1,2,-11999
 00006   0634 00 2 00014          SXA      IO,2
 00007   0762 00 0 02221          RTBB     1
 00010 -0540 00 0 00014           RCHB     IO
 00011   0061 00 0 00011          TCOB     *
 00012   0774 00 2 00000  EXIT    AXT      **,2
 00013   0020 00 4 00002          TRA      2,4
 00014 -1 27340 0 00000   IO      IOCT     **,,12000
                                  END
```

```
        SUBROUTINE FOR WRITING A5  RITE(STOR)

                               *      COUNT    25
                      00002           ENTRY    RITE

   LINKAGE DIRECTOR
 00000   000000000000
 00001   513163256060

 00002   0634 00 2 00012  RITE    SXA      EXIT,2
 00003   0500 00 4 00001          CLA      1,4
 00004   0734 00 2 00000          PAX      ,2
 00005   1 50441 2 00006          TXI      *+1,2,-11999
 00006   0634 00 2 00014          SXA      IO,2
 00007   0766 00 0 02221          WTBB     1
 00010 -0540 00 0 00014           RCHB     IO
 00011   0061 00 0 00011          TCOB     *
 00012   0774 00 2 00000  EXIT    AXT      **,2
 00013   0020 00 4 00002          TRA      2,4
 00014 -1 27340 0 00000   IO      IOCT     **,,12000
                                  END
```

```
                    00021          ENTRY   RONN
                    00002          ENTRY   ROUN

  LINKAGE DIRECTOR
00000  000000000000
00001  512445456060

00002  0560 00 0 00016   ROUN  LDQ ROUN+12,0
00003  0200 00 0 00017         MPY ROUN+13,0
00004  0763 00 0 00004         LLS 4,0
00005  0767 00 0 00004         ALS 4,0
00006  0765 00 0 00004         LRS 4,0
00007 -0600 00 0 00016         STQ ROUN+12,0
00010  0400 00 0 00016         ADD ROUN+12,0
00011  0601 00 0 00016         STO ROUN+12,0
00012  0771 00 0 00004         ARS 4,0
00013 -0501 00 0 00020         ORA ROUN+14,0
00014  0300 00 0 00020         FAD ROUN+14,0
00015  0020 00 4 00002         TRA 2,4
00016 +002312421637            OCT 2312421637,1737,200000000000
00017 +000000001737
00020 +200000000000
00021 -0634 00 1 00051   RONN  SXD IR1,1
00022 -0634 00 4 00052         SXD IR4,4
00023  0534 00 1 00044         LXA L20,1
00024  0500 00 0 00020         CLA ROUN+14,0
00025  0601 00 0 00053         STO C,0
00026  0074 00 4 00002         TSX ROUN,4
00027  0761 00 0 00000         NOP
00030  0300 00 0 00053         FAD C,0
00031  0601 00 0 00053         STO C,0
00032  2 00001 1 00026         TIX RONN+5,1,1
00033  0241 00 0 00045         FDP L20+1,0
00034  0500 00 0 00050         CLA L20+4,0
00035  0763 00 0 00043         LLS 35,0
00036  0302 00 0 00046         FSB L20+2,0
00037  0765 00 0 00043         LRS 35,0
00040  0260 00 0 00047         FMP L20+3,0
00041 -0534 00 1 00051         LXD IR1,1
00042 -0534 00 4 00052         LXD IR4,4
00043  0020 00 4 00002         TRA 2,4
00044  0000 00 0 00024   L20   HTR 20,0
00045 +205500000000            DEC 20.,.5,15.49193340,0
00046 +200400000000
00047 +204757573654
00050 +000000000000
00051  0000 00 0 00000   IR1   HTR 0,0
00052  0000 00 0 00000   IR4   HTR 0,0
00053  0000 00 0 00000   C     HTR 0,0
                               END
```

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
The MITRE Corporation
Bedford, Massachusetts

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP

3. REPORT TITLE
THE MUSE SYSTEM: DESCRIPTION AND MANUAL FOR OPERATION

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
N/A

5. AUTHOR(S) *(Last name, first name, initial)*
Slawson, A. Wayne

6. REPORT DATE
December, 1965

7a. TOTAL NO. OF PAGES
47

7b. NO. OF REFS
7

8a. CONTRACT OR GRANT NO.
AF19(628)2390

b. PROJECT NO.
1700

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)
ESD-TR-65-94

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
TM-3913

10. AVAILABILITY/LIMITATION NOTICES
1. Qualified requestors may obtain from DDC.
2. DDC release to CSFTI authorized.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
Office of Scientific and Technical Information,
Electronic Systems Division
L.G. Hanscom Field, Bedford, Mass.

13. ABSTRACT

The MUSE system, an IBM 7090 computer program and associated conversion equipment, has been designed for use as a sound synthesizer. Concise descriptions of complex sounds including human speech are converted by the MUSE system into sound pressure waveforms. The inputs to the MUSE system are specifications of the changing resonance frequencies of multiple acoustic filter networks and of the changing frequencies and amplitudes of the sources of acoustic energy that excite those networks. The output of the MUSE system is a sampled waveform calculated for each resonance by the solution of a second-order difference equation. The results are summed over a single system of resonances and then the resonance systems are also added together. The resulting string of sampled waveform ordinates is written in digital form on magnetic tape. Conversion to a voltage waveform is accomplished by use of the standard IBM 729 IV tape transport unit and a simple digital-to-analog converter. Although the quality of the sound is somewhat degraded by tape wow and flutter, acceptable and highly intelligible speech has been synthesized.

DD FORM 1 JAN 64 1473

UNCLASSIFIED
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Simulation						
Programming (Computers)						
Speech Representation						
Speech						
Voice Communication Systems						
Sound Reproduction Systems						
Sound Generators						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
_______________________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
_______________________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
_______________________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.