Automatic transcription of keyboard music.

CATTANACH, J.M.

1975

A U T O M A T I C T R A N S C R I P T I O N

O F

K E Y B O A R D M U S I C

J.M. CATTANACH

AUGUST 1975.

THESIS SUBMITTED FOR THE DEGREE

OF M.PHIL. AT ROBERT GORDON'S

INSTITUTE OF TECHNOLOGY.

ACKNOWLEDGEMENT:

I should like to thank my supervisor, Dr. P. Mars, for his guidance. Also I wish to thank A. Brown and D. Simmonds for their assistance in software design.

SUMMARY.

The aim of this system is to produce an automatic transcription of music played on any keyboard instrument.

Work undertaken is divided into 2 sections. Firstly, the keyboard is monitored and information is stored in a memory device. Hardware is developed in order to perform this task. The latter section involves processing the information using software computer programs. Two types of output are obtained, i.e. an alphanumeric printout from a teletype and a staff notation printout from a graph plotter.

CONTENTS:

CONTENTS (Contd.) Page No.

Chapter 1. Introduction and Review

The aim of this project is to generate a printout of the computer's interpretation of music played on a keyboard instrument. Automatic reproduction of a musical score is not a modern concept. In 1876, the first version of an automatic piano was introduced in the form of the pianista as shown in Fig. 1.1. This was the original form of piano player. It was operated by a handle which produced a vacuum to work a set of keys which in turn played the keys of an ordinary piano. A later development was the pneumatic player piano as illustrated in Fig. 1.2. The action of this instrument is explained with the use of Fig. 1.3. The pressure at E with respect to A caused E to be raised, hit the piano action wippen and the hammer to hit the string. With the advent of electric power, the pneumatic version was replaced by the electrical piano player which used electrical pulses rather than air pressure to cause a note to be struck. The construction of this is shown in Fig. 1.4.

In each type of pianola, a music roll is used. Fig. 1.5. shows the layout of such a roll of music. Note pitch and time values are defined, accompanied by the degree of pressure used to strike the note. The soft and sustain pedal dynamics are also defined. The automatic player was not solely restricted to piano. An example of another instrument adapted to an automatic music machine was the violin as illustrated in Fig. 1.6.

The music roll has been quoted as the "film of the music camera" (1), but a piano camera exists (7), the mechanism of which is shown in Fig. 1.7., with a drawing of a photographic output. The action of the camera/

camera is explained with the aid of the upper diagram in Fig. 1.7. Hole F represents the last 12 mms. of hammer travel before hitting the key, as the hammer velocity over the last 12 mms. is practically constant (8). The light source J is under the film D at point E and point C of the hammer tail is focused on E by lens L. When the hammer moves, light shines through hole F then the light is blocked from the film by point A. Once the hammer has struck it falls back slightly but remains in a forward position until sustain is removed. With reference to the lower diamgram of Fig. 1.7. movement of this roll of film is continuous and from right to left. The distance between 2 vertical lines represents 0.04 seconds. For any note struck, black bar A represents the time for the hammer to travel 12 mm. and A plus B the last 24 mm. Bar C represents the time of retreat of the hammer from the string and this, together with the following white bar, gives the time for which the note was held. The final black bar D represents the total return of the hammer and white bar E implies that the note is not being struck. Fig. 1.7. lower also displays the monitor of the damper pedal. At point G the sustain pedal was operated to sustain the chord. The units of A plus B are scaled into 17 steps of 2 decibels each. The intensity, therefore, has the range of 34 dB. Pitch is determined by the horizontal line, one per note, and the total time value derived from distance A to D. From the information derived from the piano camera, it is possible to deduce the transcriptionfor the piece.

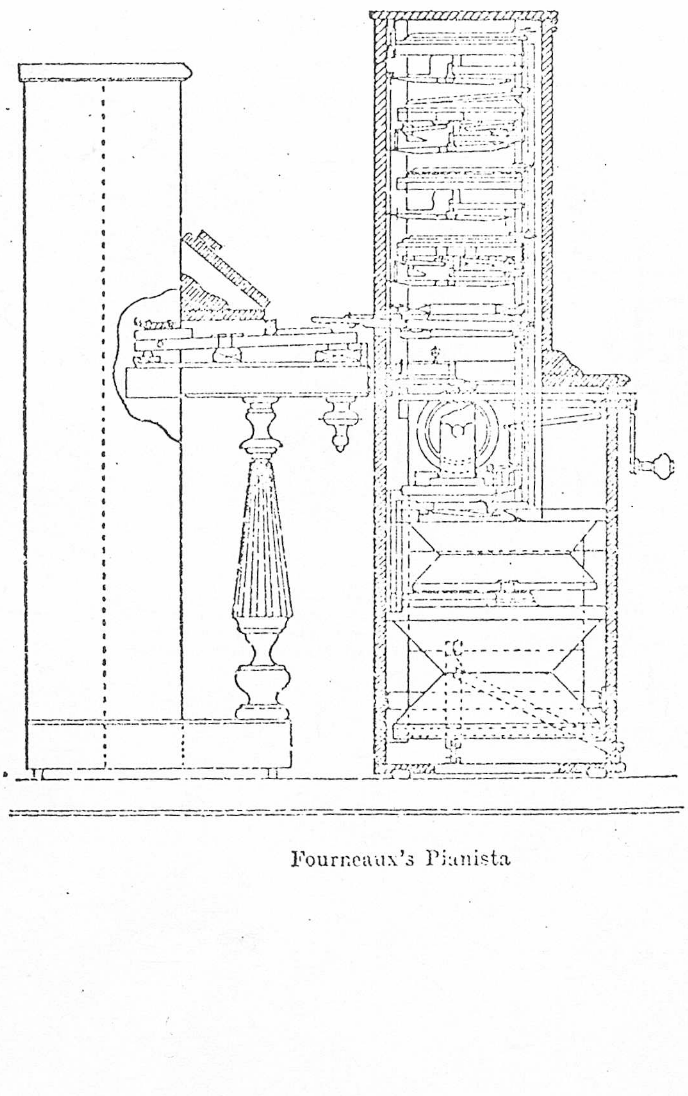
Recently an attempt has been made to elicit the same information using an on line computer at Cambridge. This system enables one to produce transcripts of musical score up to the complexity of simple hymn/

hymn tunes (4). A monitoring system gives data to the main program which stores the note number, start point and duration. From this, the musical note value is determined and is displayed on the visual display unit (V.D.U). An example of this display is given in Fig. 1.8. The system designed at Cambridge is similar to the work described in the present thesis. Both systems must monitor the action of a keyboard and store the information until sufficient data is available for processing. Software design involves the compilation of a program to process the data to give a printout. The main difference between the two systems lies in the data processing technique. Using the interactive approach, data is processed on-line, i.e. during the performance.

The fundamental restriction of the interactive approach is provided by the on-line computing requirements. Hence the practical examples are restricted by computation speeds to the simplicity of simple hymn tunes. However, the system to be described in the present work is not restricted in this manner. This is a direct consequence of the fact that all computations are performed off-line, data being stored on a cassette system during the performance and subsequently processed off line.

At present, work is being performed to interface a pianola with a computer. The object is to replace the music roll by a cassette tape. Although not directly connected to the following system, certain similarities arise and information received (9) has been of great use in designing the system described in the following chapters. The system/

system to be described has potential applications in providing a musical score, musical composition by computer and as a teaching aid. The primary aim is, however, to provide an automatic transcription for music composers and publishers.



Fourneaux's Pianista

FIGURE 1-1

Above is a representative Pneumatic Player Piano. Compare its complicated mechanism of bellows, tubes, valves, screws, levers, buttons and adjustments of various kinds with the simplicity of the Telektra-ly equipped piano shown opposite. Tuning often requires the removal of a part of this mechanism.

FIGURE 1·2

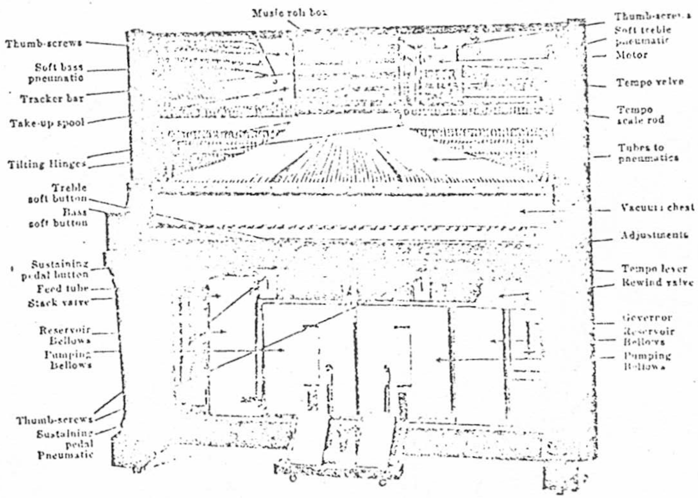

Music roll box
Thumb-screws
Soft bass pneumatic
Tracker bar
Take-up spool
Tilting Hinges
Treble soft button
Bass soft button
Sustaining pedal button
Feed tube
Stack valve
Reservoir Bellows
Pumping Bellows
Thumb-screws
Sustaining pedal Pneumatic
Thumb-screws
Soft treble pneumatic
Motor
Tempo valve
Tempo scale rod
Tubes to pneumatics
Vacuum chest
Adjustments
Tempo lever
Rewind valve
Governor
Reservoir Bellows
Pumping Bellows

DIAGRAM OF PNEUMATIC PLAYER OPERATION

Illustration "D"

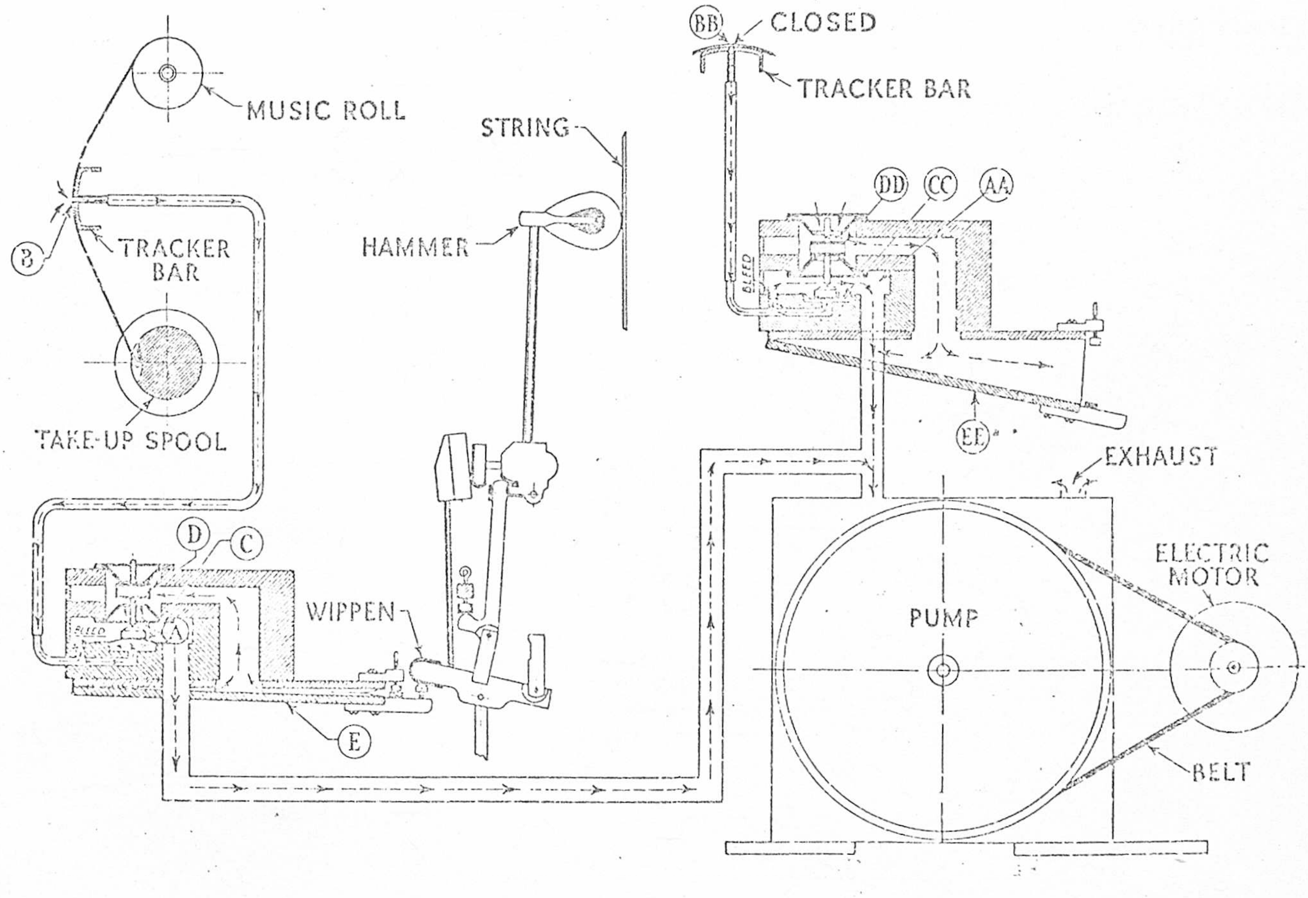


FIGURE 1 - 3

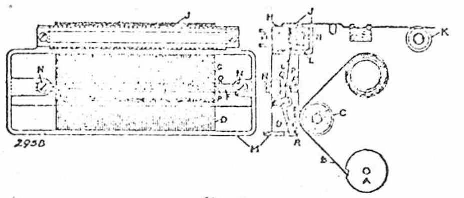


FIG. 1.

A, music-tape case; B, music tape; C, tracker roll; D, reading finger; E, ivory insulating block; F, contact bar; G, contact wire; H, resistance coil; J, comb wire; K, terminal; L, resistance coil frame; M, cradle casting; N, adjusting screws for contact bar; P, platinum strip and contact bar; Q, platinum bands on contact wires; R, swivel rod for reading fingers.

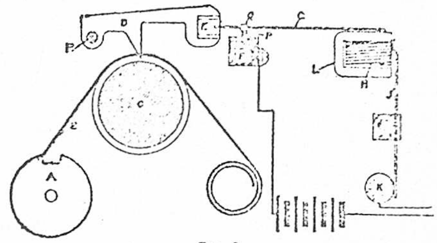


FIG. 8.

A, music-tape case; B, music tape; C, tracker roll; D, reading finger; E, ivory insulating block; F, contact bar; G, contact wire; H, resistance wire; J, comb wire; K, terminal; L, resistance coil frame; P, platinum strip in contact bar; Q, platinum band on contact wire; R, swivel rod for reading finger.

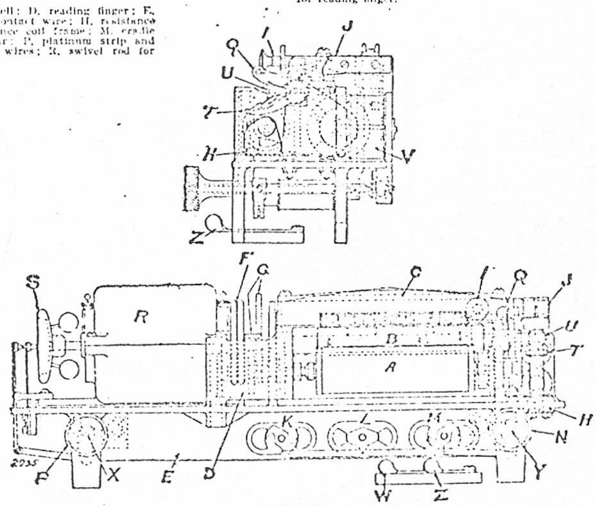


FIG. 2.

A, music-tape case; B, music tape; C, cradle; D, swinging carriage; E, base; F, rubber belt drive; G, gears; H, starting switch; I, expression cut-out knob; J, trip switch; K, forte expression magnet; L, medium forte expression magnet; M, piano expression magnet; N, expression handle; P, tempo handle; Q, carriage latch; R, motor; S, governor; T, lifting lever; U, trip button; V, trip magnet; W, bass subduing switch; X, sustaining pedal button; Y, soft pedal button; Z, treble subduing switch.

FIGURE 1·4

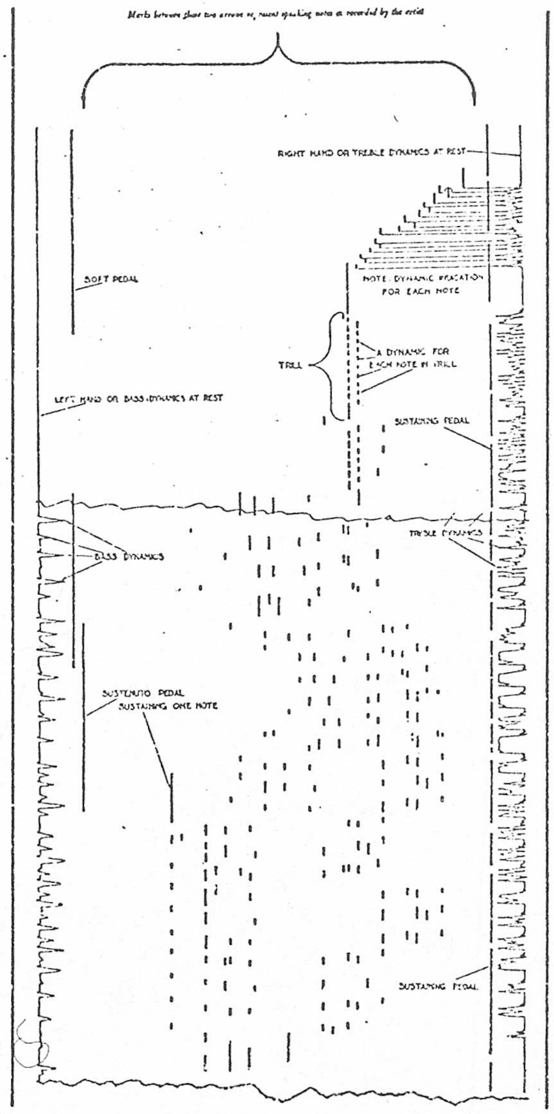
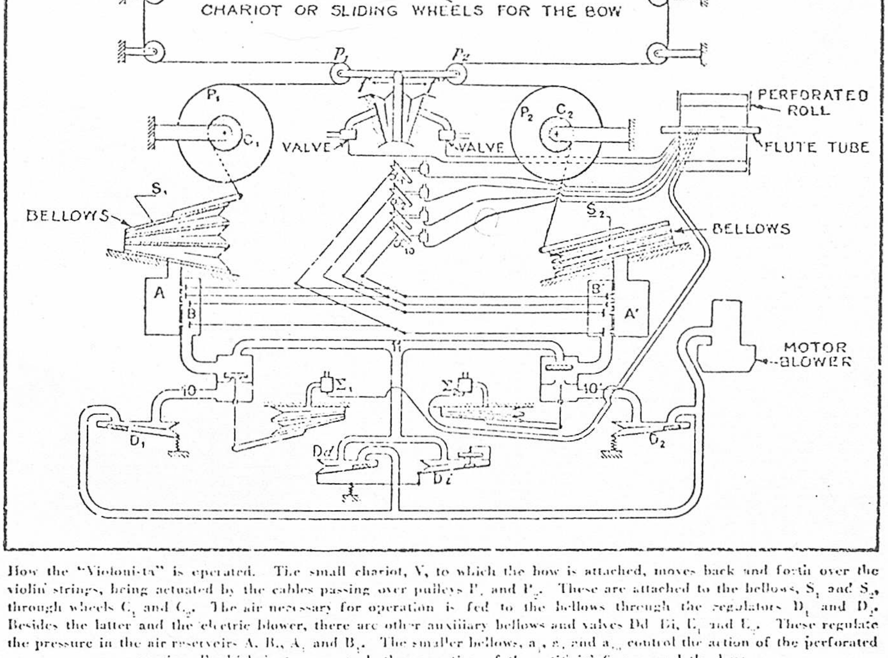
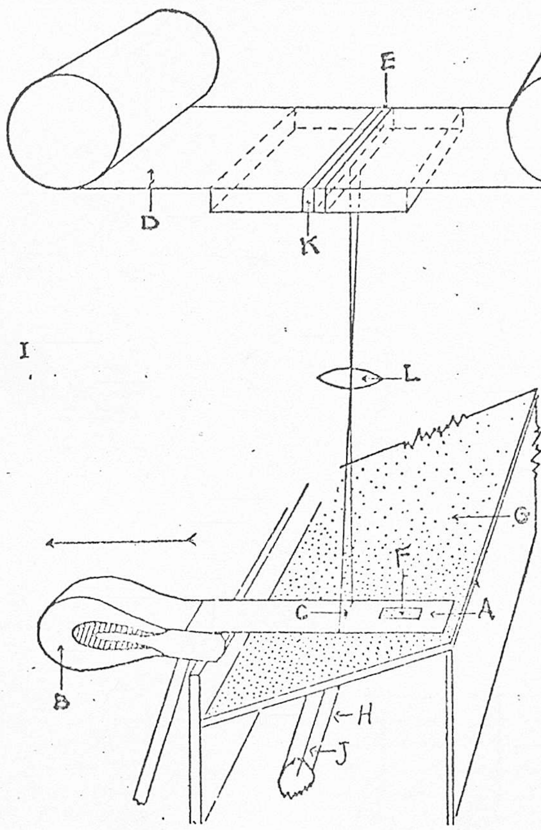

RIGHT HAND OR TREBLE DYNAMICS AT REST
SOFT PEDAL
TRILL
A DYNAMIC FOR EACH NOTE IN TRILL
LEFT HAND OR BASS DYNAMICS AT REST
SUSTAINING PEDAL
TREBLE DYNAMICS
BASS DYNAMICS
SUSTENUTO PEDAL
SUSTAINING ONE NOTE
SUSTAINING PEDAL

FIGURE 1·5

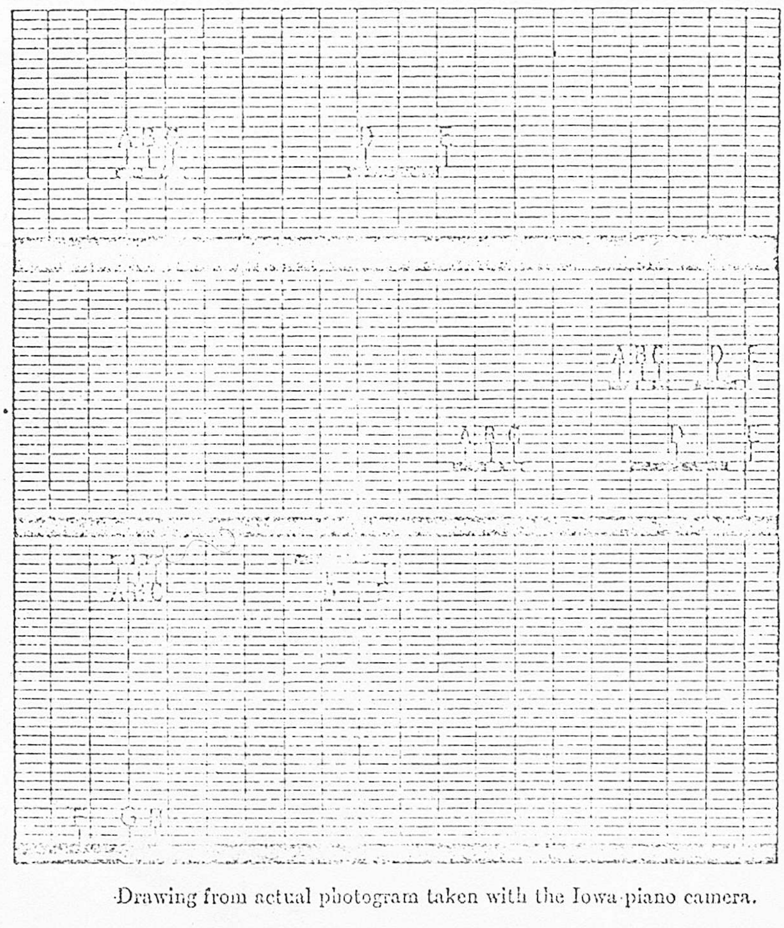


How the "Violonista" is operated. The small chariot, V, to which the bow is attached, moves back and forth over the violin strings, being actuated by the cables passing over pulleys P_1 and P_2. These are attached to the bellows, S_1 and S_2, through wheels C_1 and C_2. The air necessary for operation is fed to the bellows through the regulators D_1 and D_2. Besides the latter and the electric blower, there are other auxiliary bellows and valves Dd Di, E_1 and E_2. These regulate the pressure in the air reservoirs A, B., A_1 and B_1. The smaller bellows, a_1, a, and a_{11} control the action of the perforated music roll which in turn controls the operation of the artificial fingers and the bow

FIGURE 1-6



-Schematic drawing of the mechanism of the piano camera. Explanation in text.



-Drawing from actual photogram taken with the Iowa-piano camera.

FIGURE 1-7

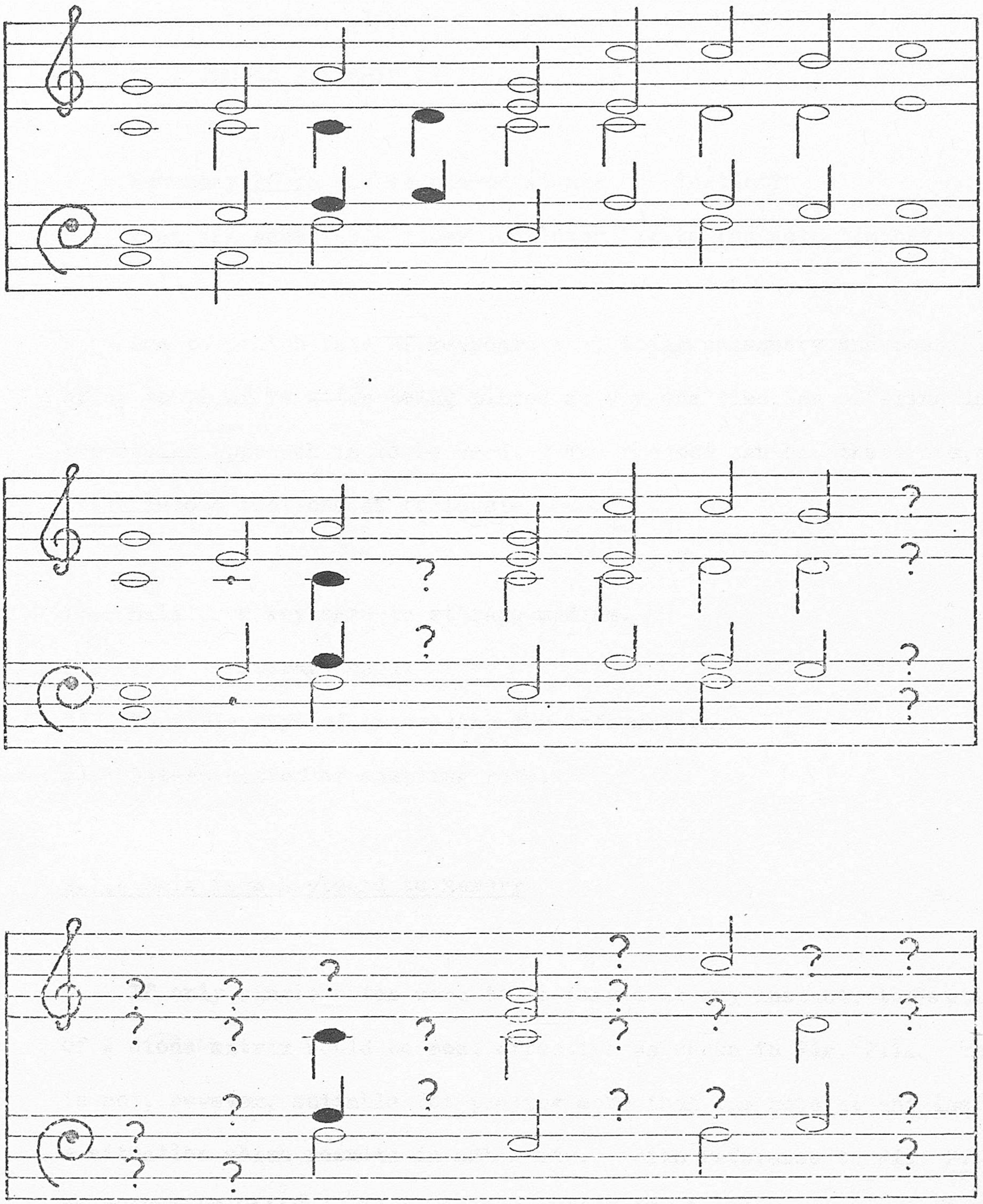



FIGURE 1·8

Chapter 2. Possible Methods of System Design

The object of the project is to produce an automatic transcription of a piece of music, played on a keyboard instrument. Consideration must be given to two main factors:

1). How many notes can be played at any one instant?

2). What are acceptable upper and lower limits for note timing?

Due to a high rate of keyboard scan being necessary and possibility of up to 12 or 14 notes being played at any one time, an off-line data processing approach is to be used. The project can be, therefore, classified into 4 sections as follows:-

1). Data from keyboard to storage medium.

2). Type of storage used.

3). Possible ways of processing the information.

4). Determination of sampling rate.

2.1. Data from Keyboard to Memory

If only single notes were to be played at any instant, the action of a diode matrix would be most effective as shown in Fig. 2.1. This is not, however, suitable for playing more than one note at any instant, a situation which results in ambiguity. With reference to Fig. 2.1., an output of 01111 could represent switch 15 closed, switches 11 & 12 closed or even switches 8, 4, 2 and 1.

In order to monitor the states of multiplicity of keys at any instant in/

in time, two possible approaches exist. The first is to scan along the keyboard at a rate much faster than a performer can move a key through an ON-OFF-ON sequence. Fig. 2.2. shows a data selector, by means of which multiple inputs are enabled sequentially and presented at output Z. The output of the 6 bit counter determines which input is enabled. With reference to Fig. 2.2., 64 inputs are shown and, say, for note number 18 depressed, logic 1 will be present at Z when the counter output enables input No. 18. For all other counter values, the output at Z is logic 0. Another counter monitors the number of scans completed. The information recorded is stored in the following form:

Note Values Scan Number

For example if notes 4 & 5 were played during scan 9 and notes 4 & 7 on scan 10 the following information would be recorded in memory using 6 bit binary code.

000100	000101	001001
binary equivalent for note 4	binary equivalent for note 5	binary equivalent for scan 9
000100	000111	001010

binary equivalent of data for scan 10.

In order to avoid ambiguity when no notes are pressed, dummy notes could be added to indicate the end of one scan.

The second possibility is to monitor the states of all the keys at any one instant as shown in Fig. 2.3. Information is loaded in parallel into a shift register and shifted out serially to the storage device. /

For a keyboard length of 120 notes, say, as in Fig. 2.3., 120 bits of information will be stored for each scan, logic 1 for a note pressed, logic 0 for any note not being played. Again, dummy notes could be added at either end of the scan in order to produce a bit-marking code.

Extending the first possibility, the selection of each key in turn allows its state to be monitored, Given that the maximum number of keys depressed at any interval in time is 10, the available memory could be divided into 10 fields as in Fig. 2.4. For a time interval t, each note present is fed to a separate memory field, and dummy notes added to complete the line of data if necessary. The field line is incremented and on the next scan, the new data is deposited in the next line of memory. Modifications exist in that a 2 x 120 bit memory could be introduced to compare the previous scan with the current scan. If equality exists then a counter is incremented and as no duplication of lines of data are made, a more efficient use is made of the available memory.

Another possibility within data selection is to allocate each note a section of memory. As shown in Fig. 2.5 each note is enabled sequentially and for a note pressed, the contents of the free running counter are deposited in α, thus indicating when the note was pressed. Each time that note is enabled, the contents of the free running counter are deposited in β until the key is no longer depressed. If the note is again depressed then the time of pressing is stored in γ and the time of ceasing to be played is stored in Δ. The data can later be processed to give the following information: at time α, that note was held for duration $(\beta - \alpha)$ and at time γ, it was again pressed, this time its duration was $(\Delta - \gamma)$./

Disadvantages arise in data selection in that the control unit for directing the information to storage becomes extremely complicated. This introduces a high error rate. A lower error rate is found in the method of monitoring the states of all the keys concurrently and transferring the information to storage as shown in Fig. 2.3. In processing the information, data selection could be of great importance if a hardware approach is to be used.

The concept of monitoring the states of all the keys at times t_0, t_1 . . . . t_n is accepted as the more efficient. As in Fig. 2.3. the states of all the key outputs are loaded into the shift registers in parallel then transferred by shifting right into the memory device. Here the information can be stored indefinitely until processing can take place.

2.2. Choice of Memory

The computer to be used, a Pdp8/e, has an 8K store, each word being 12 bits long. Storing data in remaining store once the program is loaded would be equivalent to 20 seconds worth of play, so the store of the computer is insufficiently large. In order to permit information representative of a 10 minute performance, 1.44 Mbits of storage are required. For such a storage capacity, a tape recorder must be used. One further requirement is that the tape recorder can be either played back at a sufficiently slow speed so no information is lost or the tape recorder can be stopped and started without loss of information. It would, therefore, be advantageous to have a multi-speed recorder with a motor step facility, known as forward skip.

2.3. Possible ways of processing the information

The original intention of the project is to obtain alphanumeric output of information, for example, A#2 would be the printout for the note A/

A sharp in octave number 2. To produce this, data is fed to the computer and on receipt of data indicating notes being played, a look-up table is used to give the ASCII code for the relevant characters. In addition, a purely numeric code of the data could be printed out on data tape and used in a transcription program to give a staff notation printout of pitch and musical time value. Both alphanumeric and staff notation printouts are obtained and all software considerations are described in Chapter 5.

2.4. Sampling Rate Considerations

The main factor involved is that no keyboard musician has been recorded scientifically as playing keys at a rate of change greater than 15 per second. (1) In order to monitor all key state transitions, therefore, the keyboard scan rate must be>15 scans per second. The upper limit on scan rate is governed by the maximum data rate of the tape recorder. A digital tape cassette recorder has an upper data rate limit in the order of 2.4 Kbits/second, so for a keyboard length of 120 bits, the upper limit is set at 20 scans per second. Musicians do not always play at high speed and for a slow piece of music, it would also be advantageous to have a slower data rate in order to save wastage of tape. It is already necessary for the tape cassette system to be multi-speed and for slow musical works, more information can be stored on the cassette. In conclusion, data rates will be set as $N_1, N_2, \cdots N_{16}$ determined by the cassette recorder maximum data rate, information will be loaded in parallel into a shift register, shifted serially to a digital tape cassette recorder and stored until the performance ends. In off-line mode, data will be directed to the computer for processing to give:-

a) an alphanumeric printout

b) a staff notation printout

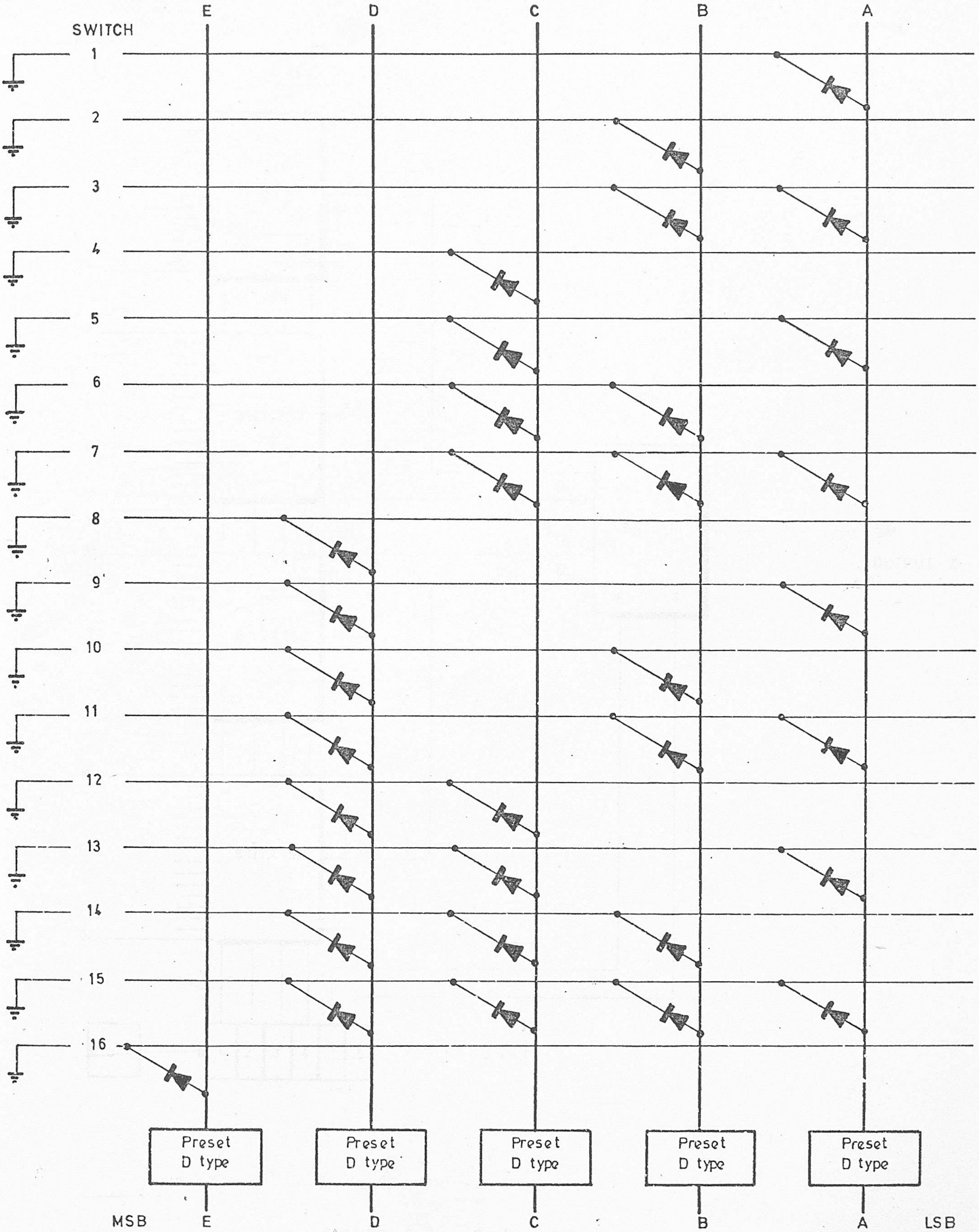


FIGURE 2·1 DIODE MATRIX

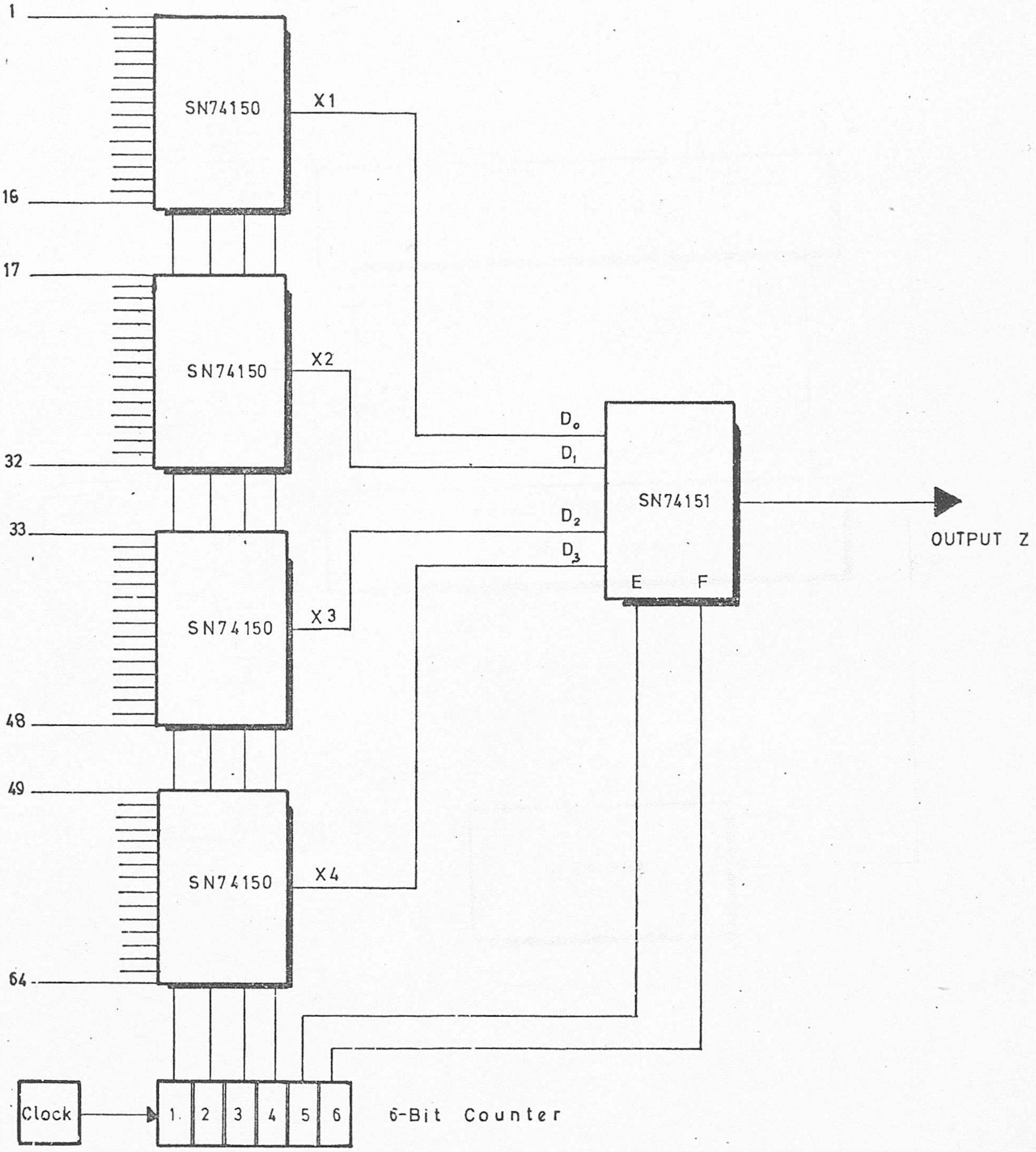


FIGURE 2.2 Data Selector

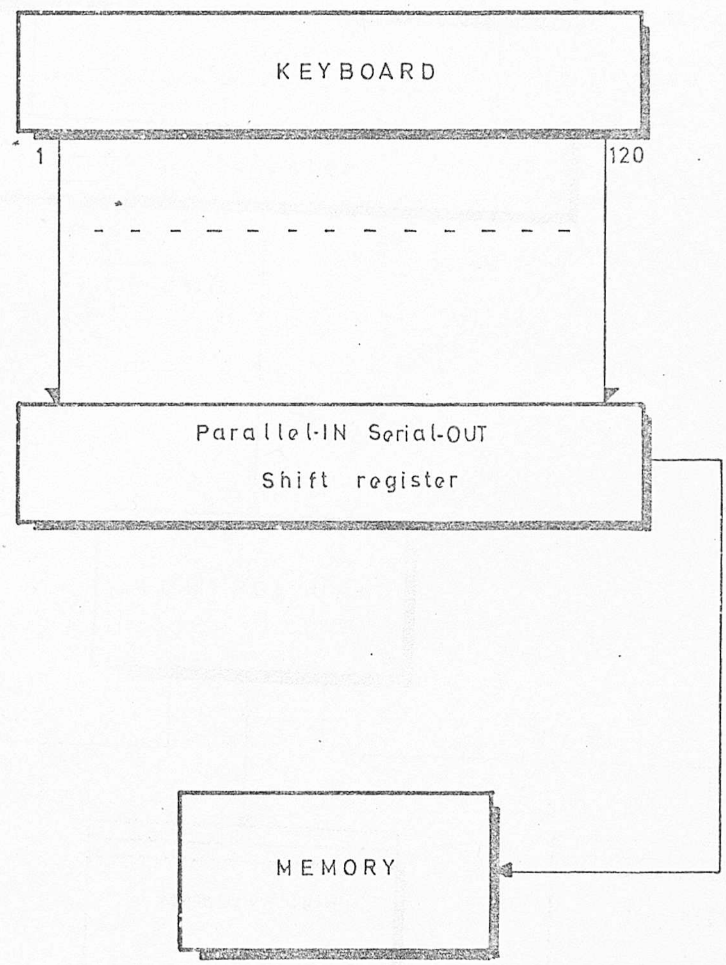

KEYBOARD
1
120
Parallel-IN Serial-OUT
Shift register
MEMORY

FIGURE 2.3

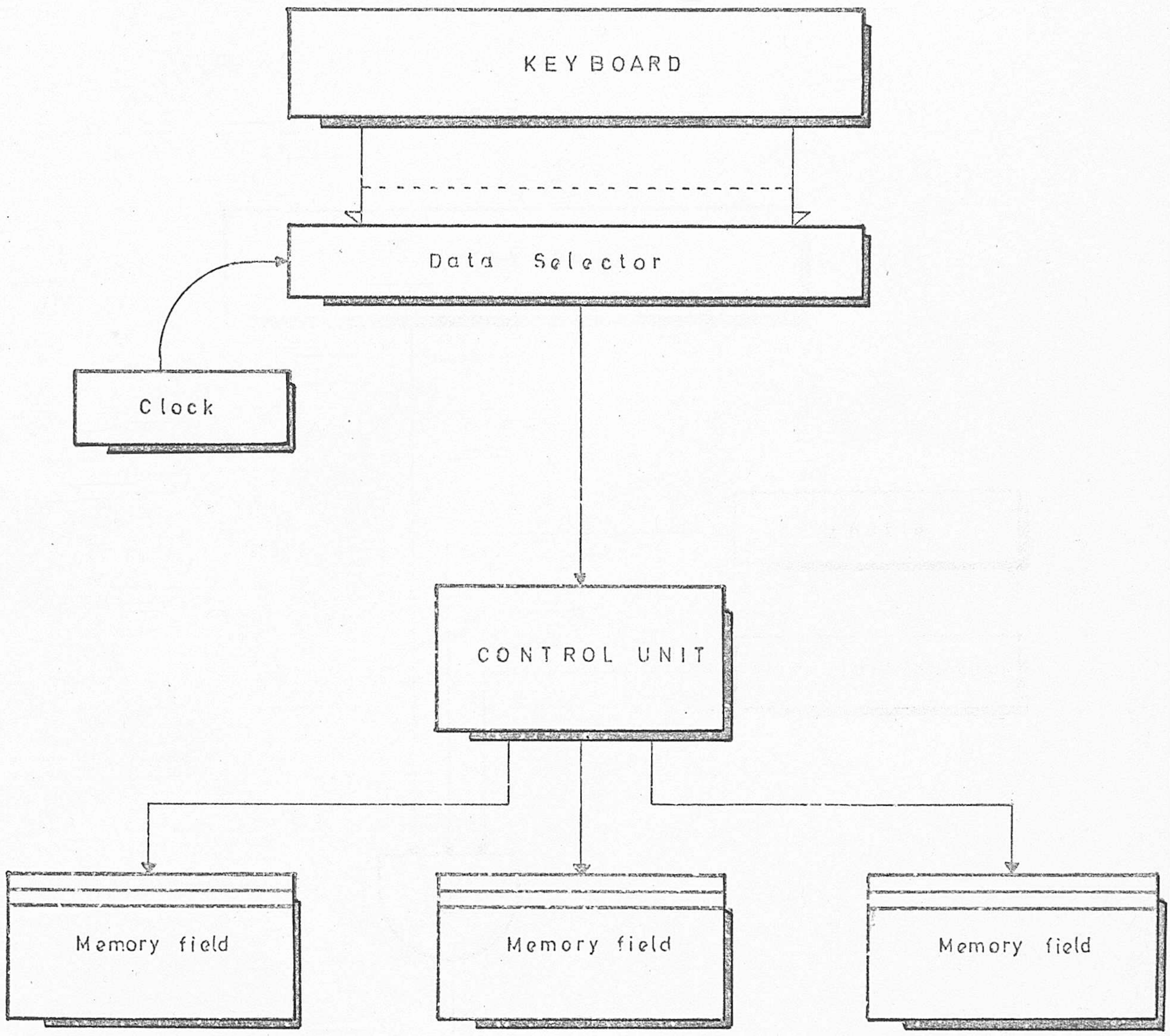

KEYBOARD
Data Selector
Clock
CONTROL UNIT
Memory field
Memory field
Memory field

FIGURE 2.4

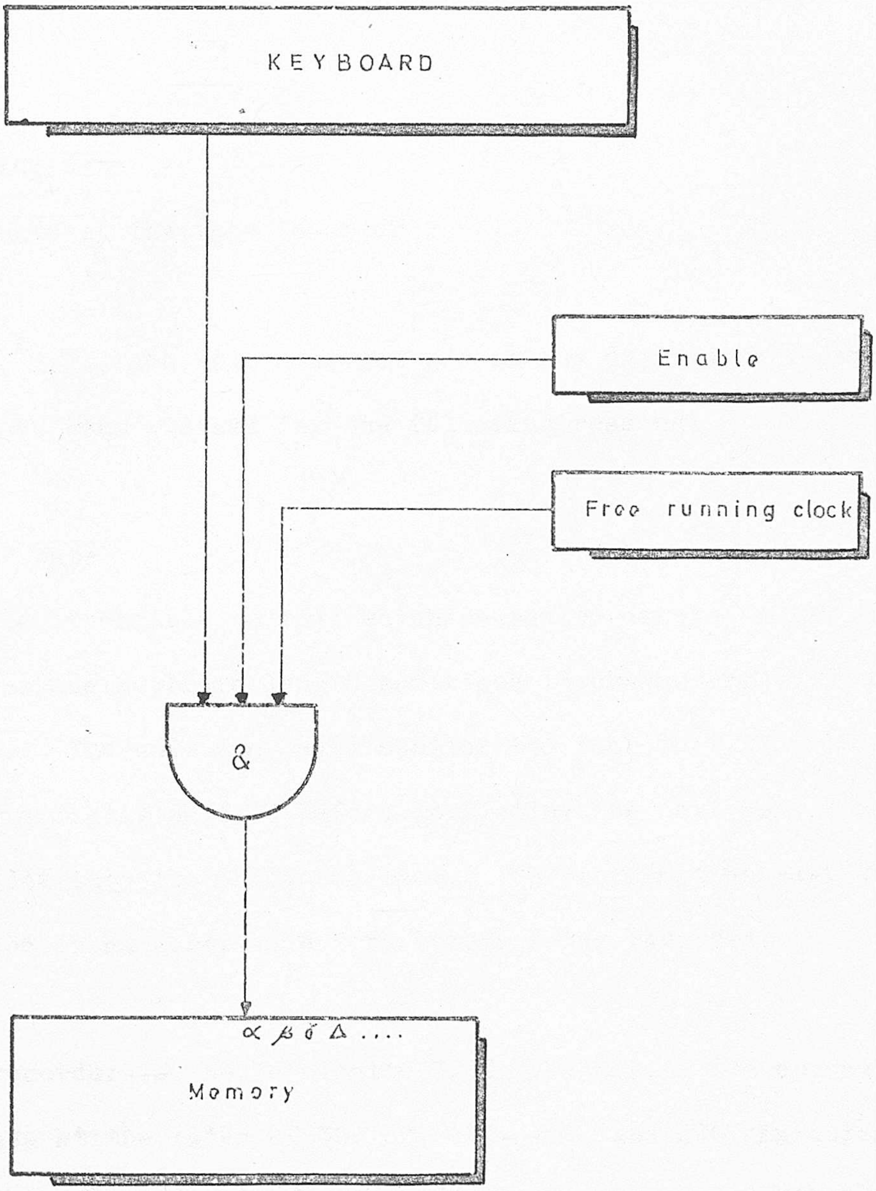


FIGURE 2.5

Chapter 3. General Description of System

The block diagram of the system is given in Fig. 3.1. Information is monitored by the interface unit and recorded on the tape recorder. At a later stage the information is played back off-line, to a computer where it is processed to give a display of the music. The keyboard consists of 120 keys, the action of each of which must be monitored by the interface unit and the data shifted serially to the tape recorder. A master oscillator operates the unit, the master frequency being governed by:

a) the sampling rate

b) the data rate of the tape recorder

As shown in Fig. 3.2., the unit must not permit any data to be recorded, until the note has been pressed for the following reasons:

a) wastage of tape

b) It would be impossible to tell which character denotes which set of notes as one keyboard length comprises 15 characters at 8 notes/character. The unit must also monitor the fact that 120 notes have been serially shifted before permitting the next set of data to be loaded into the shift register. The information must then be arranged in an acceptable form before being recorded.

The tape recorder is the Termicette 3100 by RACAL. Its operation is to accept data at the rates of 30, 60, 120, 180, and 240 characters/second (5). The signal levels for recording information are given as:-
Binary 1/

Binary 1)
Mark) - $-15V < \upsilon < -3V$
Off)

Binary 0)
Space) - $+3V < \upsilon < +15V$
On)

The range ($-3V < \upsilon < +3V$) is the excluded band of the transition region. Operation of the Termicette is in 3 modes. i.e., local, line and line-non-print. For each of the relevant modes, duplexing consists of half or full duplex. The requirements of input and output matching impedances are $R_{in} > 300\Omega$ and $R_{out} > 3K\Omega$ with $C_{out} < 2.5nF$. These values are derived from the RS 2 32 Standard Interface manual (6).

For no note pressed within any character, that character is treated as a "space" by the Termicette and ignored. This factor would cause an error in the recording, so all information is inverted and the error character will only appear if all notes are pressed within the character. Ambiguity might still arise but, musically, the probability of this occuring is very low. In order to record the information, the data character is composed of a start bit (space), the 8 information bits followed by 2 stop bits (marks). Character formation is accomplished by a pulse interrupt system, described in detail in Chapter 4.

Having recorded the information, the data is played back to the computer 15 characters at a time and compared with the previous 15 characters. The computer is a Pdp 8/e. If equality exists then a counter is incremented. When inequality exists, an alphanumeric print-out is made of the notes played followed by a time value N. N is measured in/

in 0.05 second intervals. The keyboard is divided into 10 octaves, each of 12 notes. The first octave is numbered 0, and the second numbered 1 and this continues to octave 10 being numbered 9. By this means, the alphanumeric printout only requires a 1 digit to define the octave number. The notes in any one octave are termed in the following manner:

NOTE:	1	2	3	4	5	6	7	8	9	10	11	12
	C	C#	D	D#	E	F	F#	G	G#	A	A#	B

The reason for chosing D# as apposed to E flat is to remove the problem of finding a teletype with the character "b". As an example of the printout, if a logic 1 were encountered at notes numbered 27, 30, 34 and 37, then the chord $D_{min\ 7}$ is being played and the teletype is instructed to print the following for one time interval:

D2 F2 A2 C3 1

where 1 denotes the number of scans for which that formation of notes had been held. If the same set had been pressed for 0.75 seconds, then at 20 samples per second, this printout would be obtained:

D2 F2 A2 C3 15

Further, by summing the duration of each continuously held note, thereby finding the total time depressed and multiplying by a constant K, a numerical real-time value, t, can be obtained. K is determined by the selected rate of scan and t must fall within a range of values, thereby deciding the value of the note, e.g., crotchet, quaver. This is done automatically in the staff notation transcription program discussed in Chapter 5.

A useful addition to the circuitry is the playback facility. The information/

information is played from the cassette directly back to the keys. An immediate indication of the music content is given to the composer and this facility also permits stopping the tape recorder and editing out mistakes before finally making a transcript.

FIGURE 3.1 Project block schematic

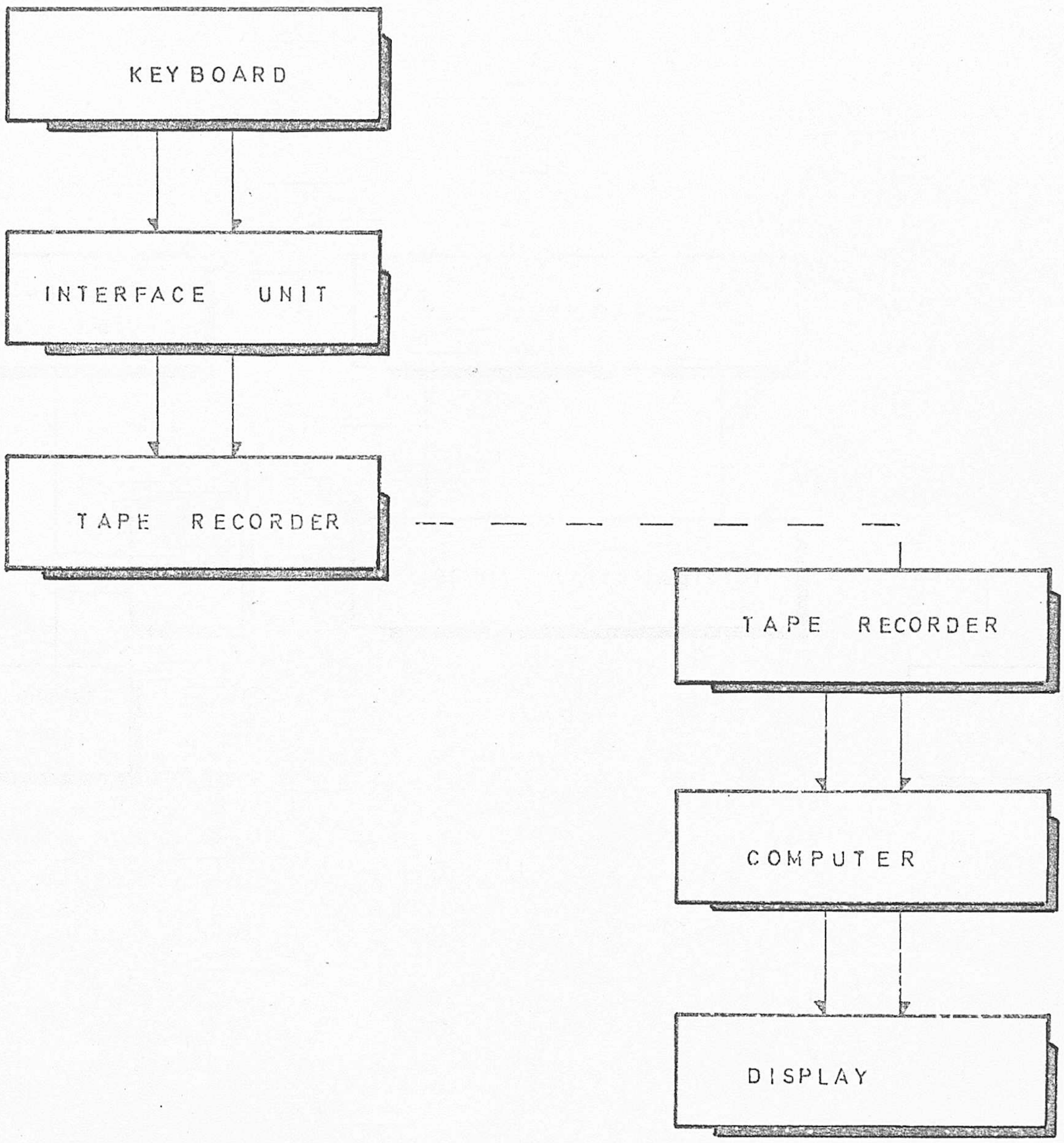


FIGURE 3.1 Project: block schematic

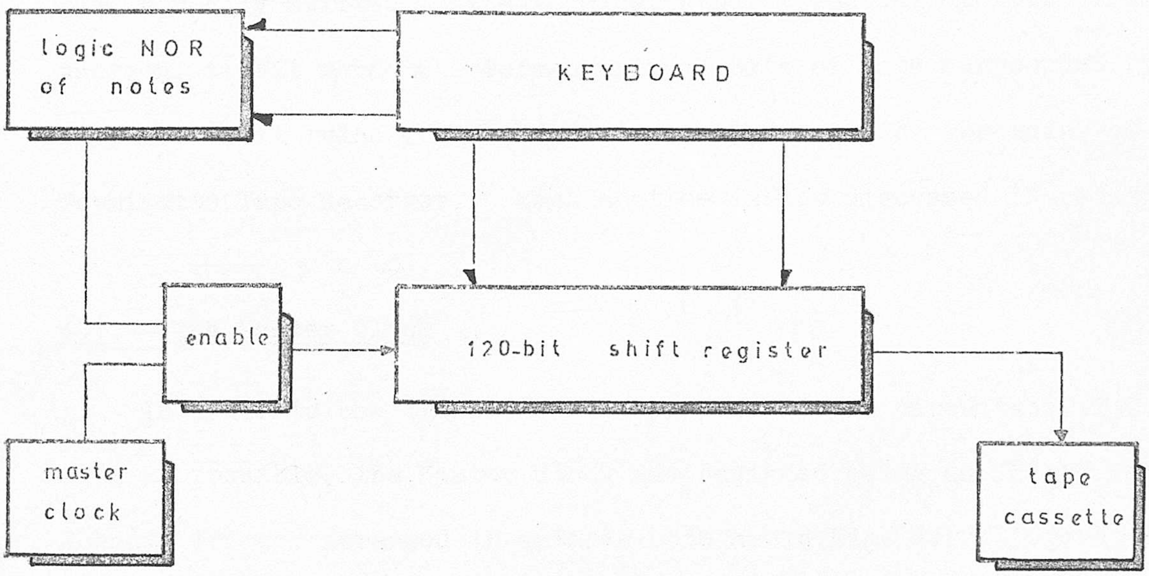

logic NOR
of notes
KEYBOARD
enable
120-bit shift register
master
clock
tape
cassette

FIGURE 3.2

Chapter 4. Detailed Design of The Transcription Unit

The schematic diagram of the transcription unit is given in Fig. 4.1. Information from the keyboard is monitored by two units. Initially, the "NOR" function enables the master clock, by means of a latch circuit and use of the "AND" function. The master clock clocks the ring counter and counters. Information is also monitored by the Latches and fed to a shift register in parallel fashion. The data is then serially shifted to the Termicette pulse routing system. In this section, 11 bit words are formed using 8 bits of data surrounded by 2 stop & 1 start pulses. The data is now prepared for recording on the Termicette Tape Recorder. Each section is now discussed in detail.

4.1. The Master Clock

In order to use Transistor - Transistor logic circuits (T.T.L.) as much as possible, the Master Clock was designed using an SN7413 Nand Schmitt Trigger arranged in astable mode as in Fig. 4.2. Operation of the clock is as follows: on switching on the capacitor voltage A is at 0 volts, thus B is at +5 volts. The capacitor C will charge up through registor R towards +5 volts. As A increases in potential, the Schmitt gate triggers at 1.7 volts and the potential at B falls immediatly to 0 volts. Capacitor C now discharges through R until potential A reaches 0.9 volts, whereupon the gate retriggers and the potential at B rises to +5 volts. Input and output waveforms are shown in Fig. 4.3. The value of C = 91nF and R = 390Ω give a period of 41.66 μsec. i.e., a frequency of 24 KHz. This is exactly 10 times the 2.4 KHz required to monitor the states of 120 notes at a rate of 20 times per second. The value of 24 KHz is necessary as between the end of one scan and the beginning of the next, shift registers and latches must be loaded, counters reset and this requires an extra batch of pulses, as shown in Fig. 4.16.

A master frequency of 10 times the required operational frequency also reduces the error involved in setting the resistor and capacitor values, e.g., an error of 200 Hz in 2.4 KHz implies a maximum of 2.6 KHz, whereas, the same error in 24 KHz, divided by 10 means a final value of 2.42 KHz. In operation the maximum error permissible for the Termicette is + 2%, i.e., within the range of 2.352 KHz and 2.448 KHz.

4.2. The Latches

The latches referred to are resettable data latches. A set of six is available in the SN74118 package. The circuit is shown together with the truth table in Fig. 4.4. The action of the latch is to permit information from A to B ($B = \bar{A}$) if the reset is a logic 0, but, the information at B remains unchanged regardless of A, if the reset is at logic 1.

4.3. Ring Counter

The ring counter is formed from 5 SN7476 dual J-K flip-flops, i.e., 10 bits long. The flip-flops are linked in series as shown in Fig. 4.5. The action of the ring counter is that the flip-flops are initially cleared, thus all $\bar{Q}$ are at logic 1 and the input to J_1, is therefore, at logic 1 (high). When the counter is clocked, Q_1 changes state to high, the input to J_1 is still high so on the next clock pulse, Q_1 again changes state to logic 0 (low). Q_2 is now high so the input to J_1 goes low and remains as such until the high state has been clocked along to Q_{10}. The action is then repeated. This circuit is known as the self-correcting ring counter. It also serves the purpose of providing the required divide by ten. Any pulse number, e.g., pulse No. 6 referred to in the text is derived from the J-K flip-flop in the ring counter, so pulse No. 6 is the Q output of flip-flop 6. All pulses derived from the ring counter are shown in Fig. 4.16.

4.4. Counters

Initially two counters are required. The first is a modulo - 120 to monitor one scan of the keyboard being serially shifted to the memory device. This comprises 2 SN74191 4 - bit binary counters. On a count of 120, the counters are reset to zero after the ring counter latch pulse No. 6 and shift register load pulse No. 7 are enabled. The counter reset is derived from ring counter pulse No. 8 as shown in Fig. 4.16. With reference to Fig. 4.6., the counters are connected such that the second counter is enabled only by the ripple count of the first. They are clocked by pulse No. 5 of the ring counter. When a count of $D_{1.}$ $A_{2.}$ $B_{2.}$ $C_{2.}$ is attained, point A goes low so point B goes high in Fig. 4.6. The Latch pulse is enabled, followed by the shift register load pulse. Both these pulses are later inverted by SN7440 Buffer Nand gates to produce a negative going pulse for the latches and a positive going pulse for loading the shift register. Ring counter pulse No. 8 plus a high at point B provide the counter system with a reset and point B returns to a low, thereby disabling the reset and loading pulses.

The second counter is a modulo-8 unit to monitor 8 bits of information being formed into a character for storage in the Termicette. Pulse No.6. is again used to clock the counter, an SN7491. It is initially set at a count of 8 in order for it to be reset to zero once the system begins to operate but not before, due to the fact that the first piece of information the Termicette must see is a start pulse. Otherwise, a continuous error would be caused in the recording. The manual reset is, therefore, applied to D_{in} as in Fig. 4.7. When the switch is pressed, D_{in} sees a high as the load pin is enabled, so D_{out} goes to high. On releasing the switch, D_{in} sees a low, so on automatic reset D_{out} will be set to zero. The counter is/

is automatically reset using pulse No. 4 of the ring counter.

Two other counters are used. The first is an SN7493 modulo-16 asynchronous counter, clocked by the master clock as in Fig. 4.8. For a master frequency, 10n, possible rates of scan are n,n/2, n/4, n/8 and n/16. For any output chosen from the counter, that output becomes the master frequency and is fed to the ring counter.

The last counter used is an SN7490 modulo-10 counter. This is clocked by the chosen sampling rate master frequency and is used to ensure a correct duration of stop and start pulse levels, before data shifting is enabled as shown in Fig. 4.9. The clock to the modulo-10 counter is indirectly enabled by the output of the modulo-8 counter of Fig. 4.7, and on receipt of 10 pulses, the all-zero state is monitored at A in Fig. 4.9. and two stop pulses are clocked out to the Termicette, followed by the start pulse of the next character. A start pulse "AND" a count of 9 at point B (thus ensuring a sufficiently large pulse width) causes 8 bits of data to be shifted serially to the Termicette, then the system is re-enabled.

4.5. Pulse Interrupt System

The schematic for this is given in Fig. 4.10. The system is pre-set to enable AA to pass pulses to clock the modulo-10 counter. This counts to 9, the duration of the start pulse, the level of which is generated at the divide by 4 unit, composed of 2 J-K flip-flops. Output BB then goes low, clearing flip-flop XX and resetting the modulo-8 counter. The flip-flop XX with Q_{XX} low disables the master clock to the modulo-10 counter and enables the master clock to ring counter circuit. This allows 8 bits of information to be serially fed to the tape recorder, at which point the modulo/

modulo-8 counter output clocks the J-K flip-flop XX. The clock to the modulo-10 counter is restored and the ring counter clock is disabled. Having reset the rest of the circuit already automatically (÷ 4 unit to 01), two high pulses followed by a low level are directed to the Termicette. These constitute the two stop and 1 start pulses. The system then returns to the ring counter enable mode and more data is recorded.

The inverter near the final output is due to the afore mentioned fact that a character of all zero bits is ignored, thus causing an error. Signal levels must finally be changed from logic levels to E.I.A. standard of ± 8 volts. This is accomplished using the circuitry shown in Fig. 4.11. An output of logic 0, 0v. switches T1 "off" which, in turn, switches "off" T2. The potential divider circuit of 2 3.3K resistors, therefore cause the output to be - 8 volts. For an input of logic 1 , +5v, transistor T1 switches "ON" Turning T2 "ON" into saturation. The output is at +8 volts due to $V_{out} = V_{cc} - V_{ce(sat)_2} - V_{(470\Omega)}$. Thus the tape recorder is presented with acceptable voltage levels.

4.6. System Enable

In the main circuit, all clock pulses are disabled until the first note/notes are pressed. This is accomplished by the "NOR" function of 120 notes as in Fig. 4.12. The circuit comprises 3 8-input NAND gates, the outputs of which are inverted then the NAND function is again made of those outputs. Following a final inversion of the data, a logic 1 is seen at A1 for no note pressed, a logic 0 when any note is played. Two octaves are shown, i.e., 24 inputs and 5 such circuits are combined using the "AND" function as shown in Fig. 4.13. and fed to a latch circuit. When any note is pressed, a low is seen at the input to the latch. A high is, therefore, produced at the output/

output which enables the master clock and recording begins.

4.7. The Shift Register

This is built from 30 SN74179 Integrated Circuits (I.C.'s), 4-bit paralled-in, parallel and serial out shift registers. The output of the most significant bit of one shift register feeds the serial-in of the next as shown in Fig. 4.14. Clocking the shift register must not be permitted before the load or shift enables are set. Shift and load enables, are, therefore combined using "OR" logic, then the "NAND" function is made with the master frequency. Final output of the shift register clock arrives after the enable has been set. The shift register receives 120 shift right pulses between each load pulse. Suitable buffering for the clock pulse is achieved using an SN7473 NAND gate I.C.

4.8. Playback

This is accomplished by forward skipping the tape recorder 15 times to give one complete information set and shifting the information into the shift register. Once all 120 bits are present, another set of latches are enabled and the outputs of these are linked to the keyboard keys. The circuit for this is given in Fig. 4.15. In "record" mode, the key information is fed to the following components:

a) the 8-input Nand gates as shown in Fig. 4.12.
b) the Latches (1) as shown in Fig. 4.15.

The Nand gates monitor whether any note has been pressed and once playing commences have no further operation. The latches, however, are reset/

reset at a frequency of 20 Hz, which is sufficient to monitor all the key changes. The outputs of these feed the inputs to the shift registers, information being serially shifted out (between reset pulses to the latches (1).)

In playback mode, information is directed from the tape recorder to the Serial-In of the first shift register and shifted right 120 bits at a time. Once 120 bits have been shifted in, the same latch pulse is used but in this case is directed to latches (2) which causes the data to be fed to the keys. Any logic 0 present at a note input will cause that note to be played.

The inverter and Nand gates to which load, shift enables and master clock are connected form the logic by which the shift register clock pulse arrives after the enables have been set. For "record" facility only, 7 boards are required, i.e., 5 dual-octave boards, each with 24 latches, 24 bit shift register with necessary inverters and buffers form the monitor system for the keyboard. One board comprises the ring-counter, modulo-120 counter with the "AND" logic for the master clock enable and buffered outputs from the ring counter. The final board consists of the Termicette Pulse Interrupt System and Master Clock. Further practical work has been done to form Printed Circuit Boards (P.C.B.) of 3 board types. The masks for the board patching are shown in Appendix E. Suitable housing for the unit was designed for the Automatic Transcription Unit and the final unit, composed of power supply, P.C.B. and relevant connectors is shown in the photograph of Fig. 4.17.

4.9. Interfacing

The/

The transcription unit is interfaced such that it can be linked to any of 3 keyboards. The organ keyboard has T.T.L. compatible switching, so no interface problem is encountered. It is shown with the Termicette and Transcription Unit in the photograph of Fig. 4.18. The dummy keyboard in Fig. 4.19. has been built so that any key pressed will cause the output to fall from logic 1 to logic 0. Concern arises with a piano. The requirement is a portable switch that will be sufficiently sensitive to key movement but not affect the action of the keys. It must also be sufficiently small so as not to hamper the pianist's movement. Such a problem provides a stimulus for future research.

Thus for a keyboard instrument, the states of the keys will be monitored at a maximum rate of 20 per second, the information formed into 8-bit characters and stored in the Termicette Cassette tape recorder. The data is later played back to the computer and a hard copy of the tape contents is made. This involves some hardware, but mainly software and is discussed in Chapter 5.

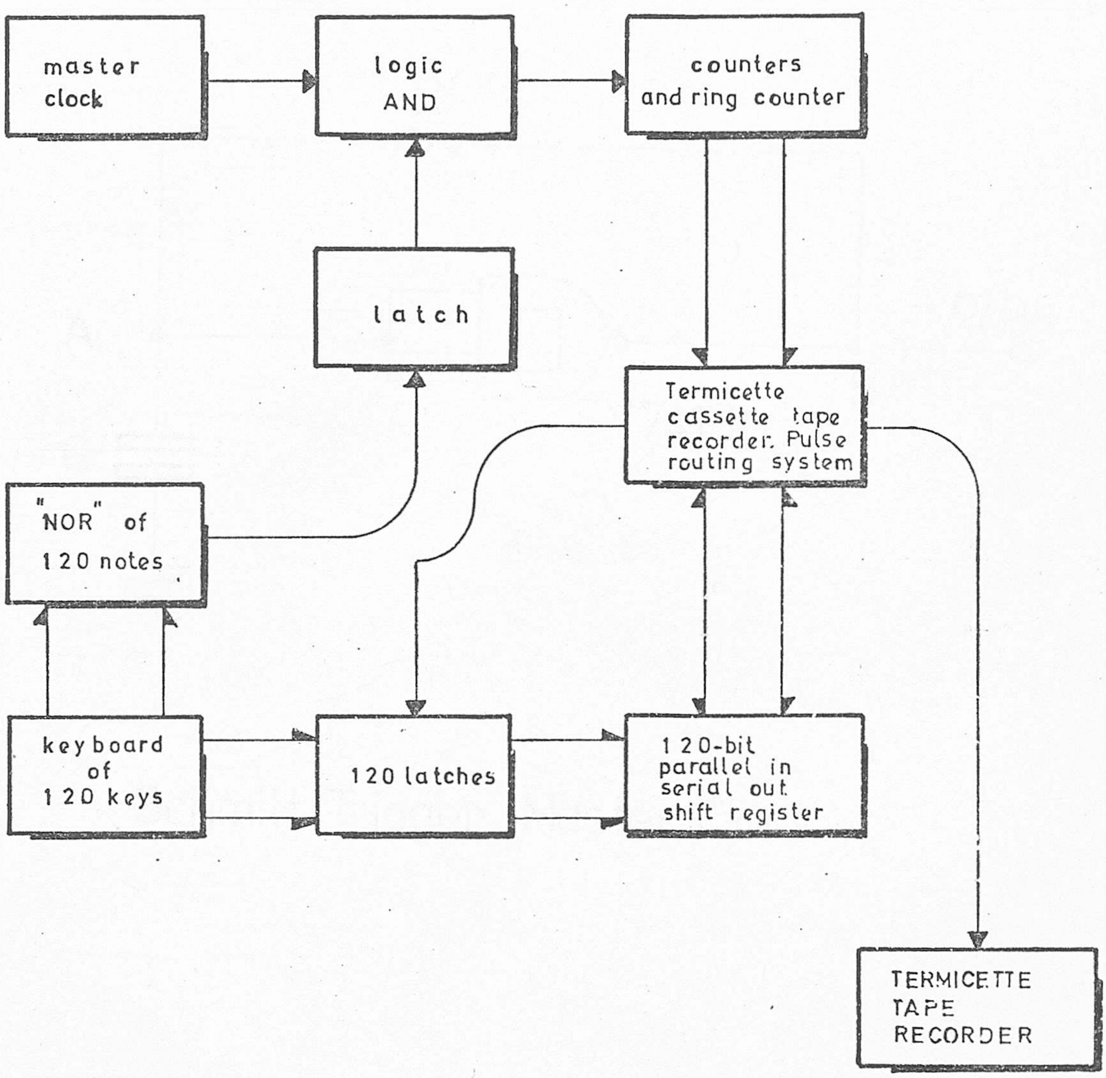
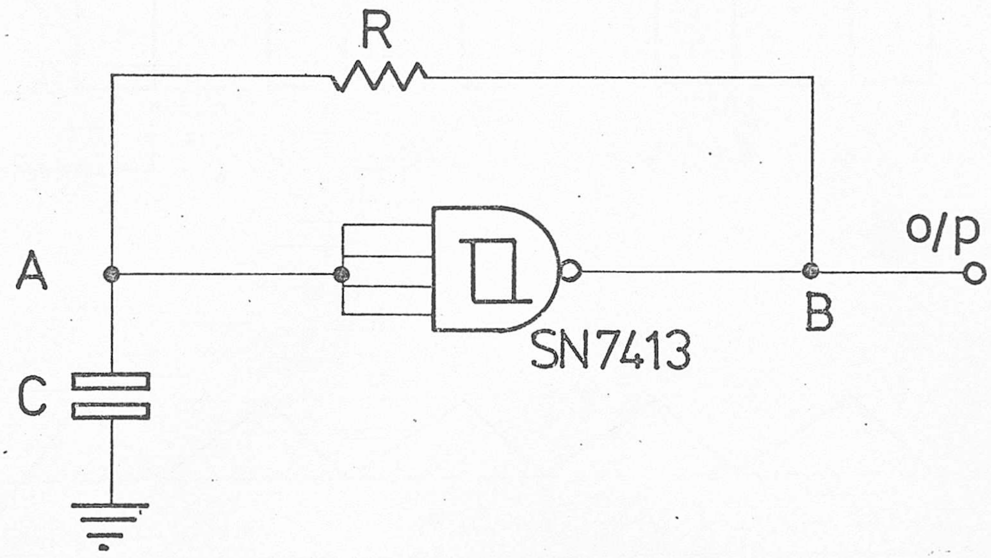


FIGURE 4.1 Project General Schematic



Schmitt Trigger, Master Clock

FIGURE 4.2

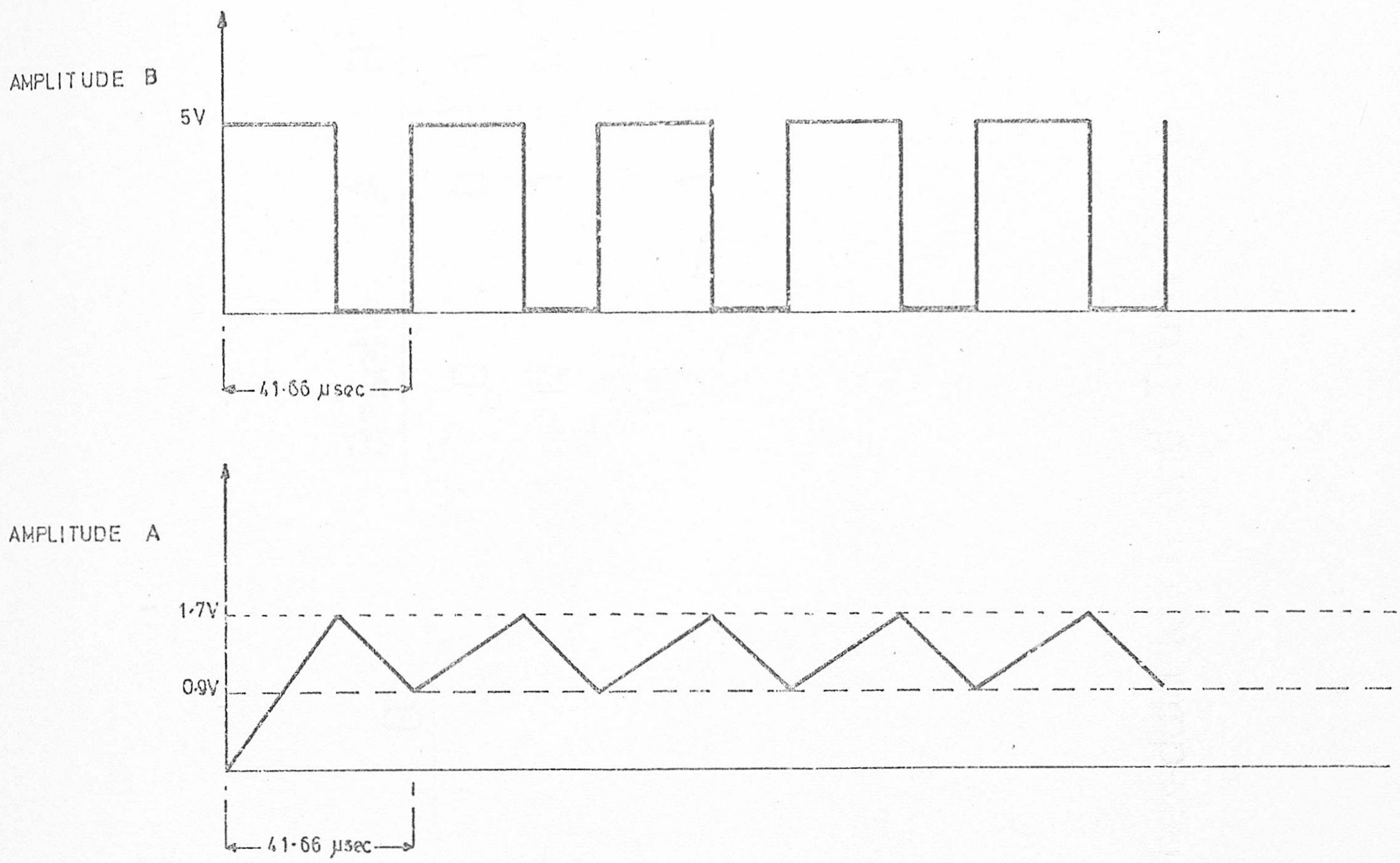
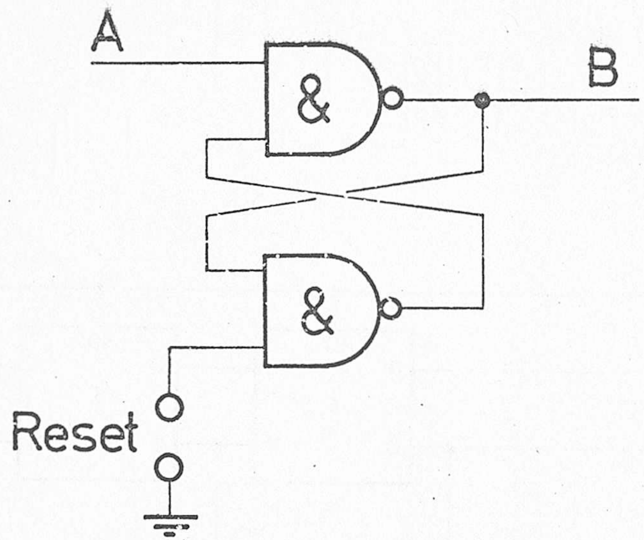


FIGURE 4.3 Schmitt Trigger waveforms



Reset	A	B
0	0	1
0	1	0
1	0	no change
1	1	no change

SN74118 Resettable hex-data latch and Truth Table

FIGURE 4.4

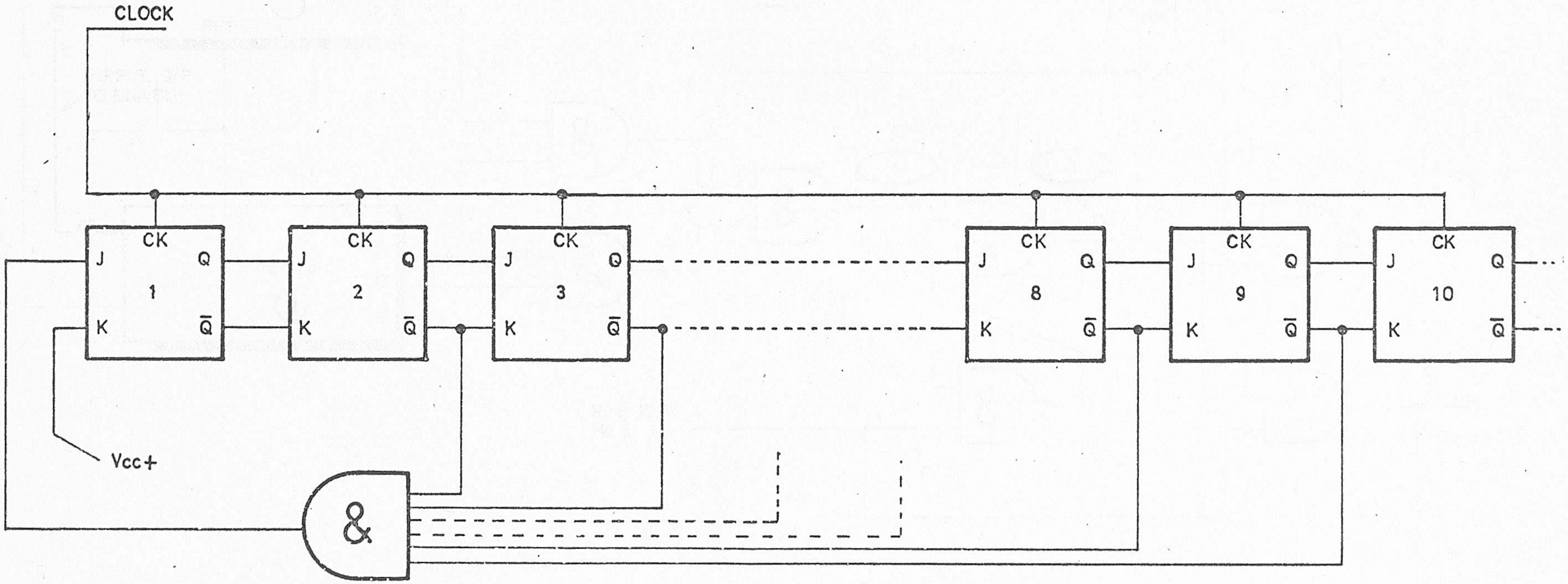


FIGURE 4.5 10-BIT RING COUNTER

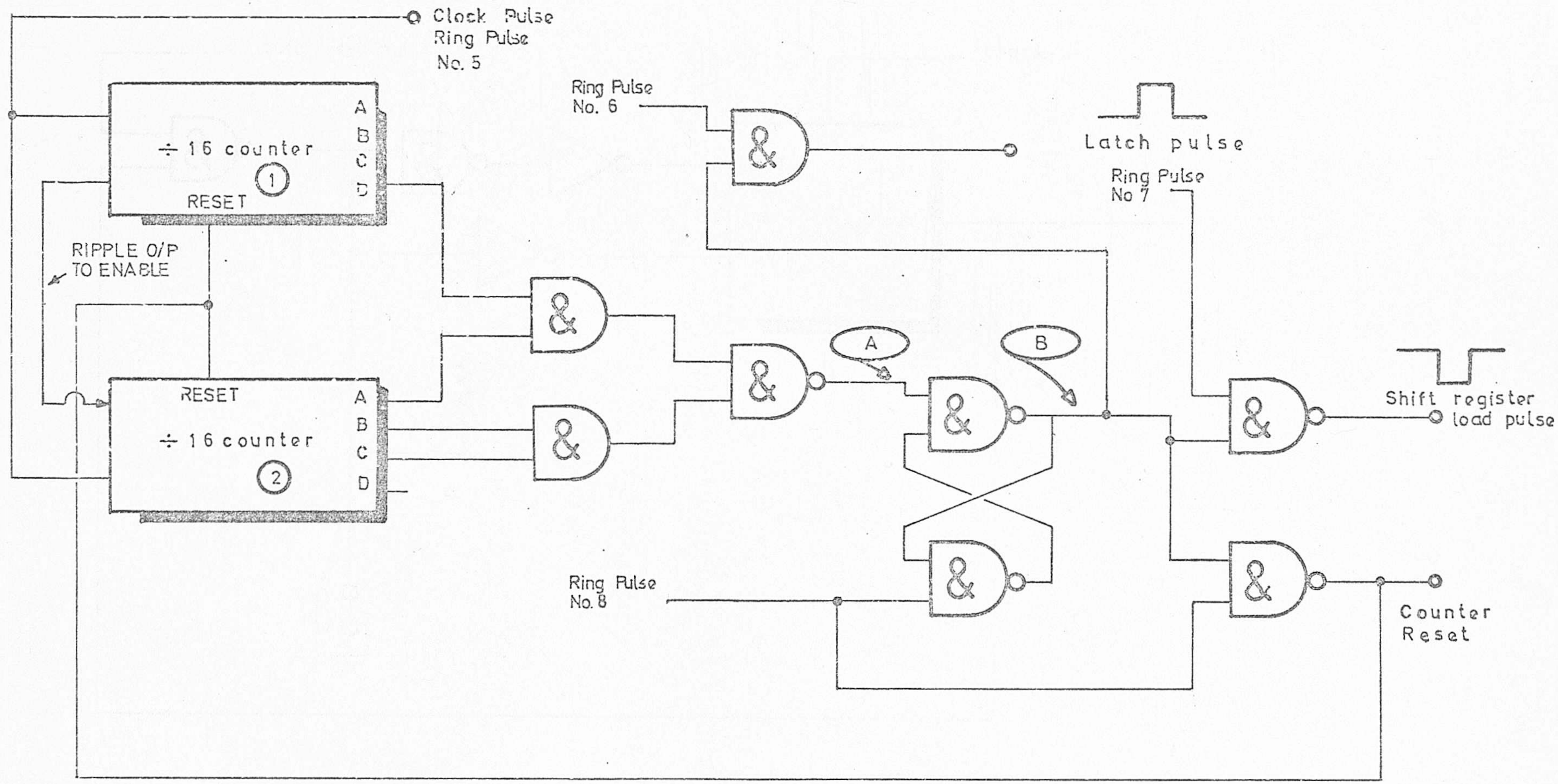


FIGURE 4.6 Counting and Data Loading System

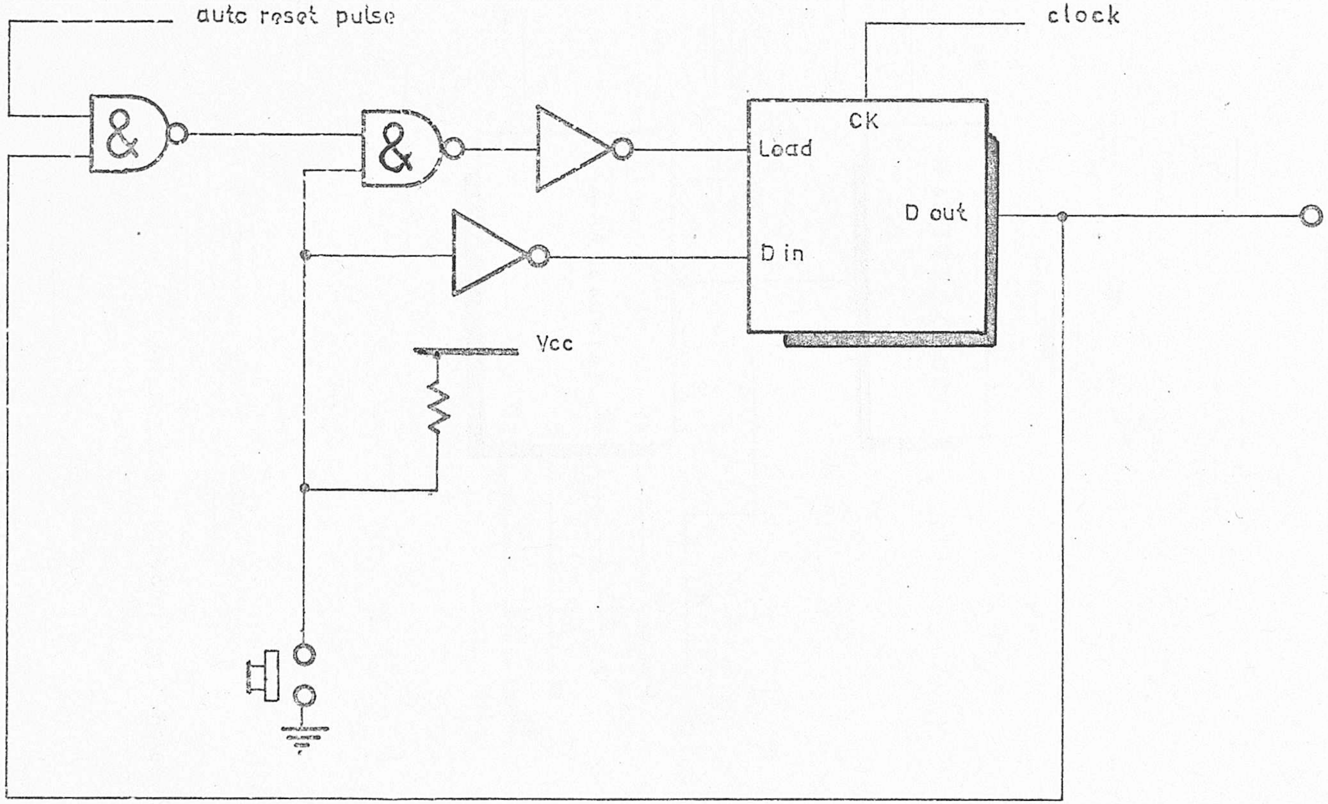


FIGURE 4.7 Modulo 8 counter monitoring Character bit Number

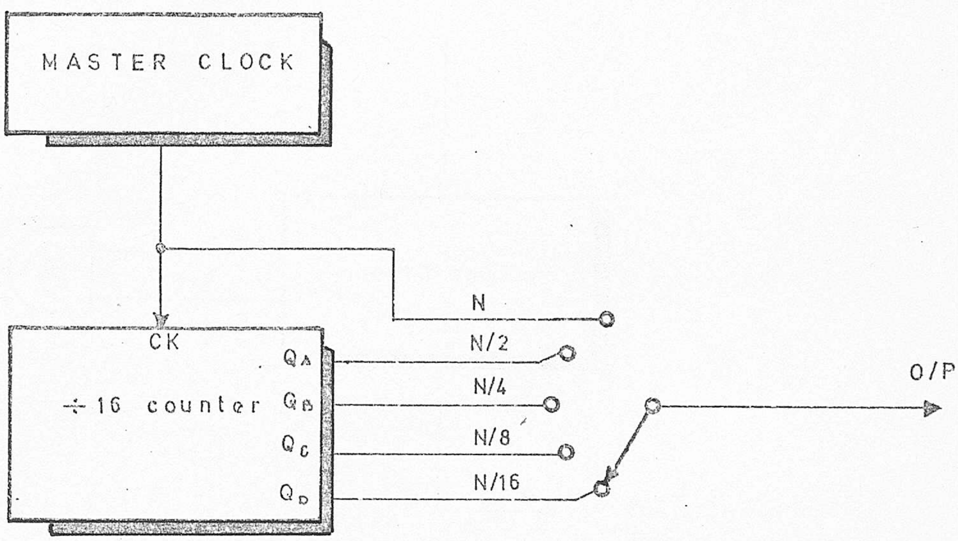


FIGURE 4.8 Sampling Rate Selector

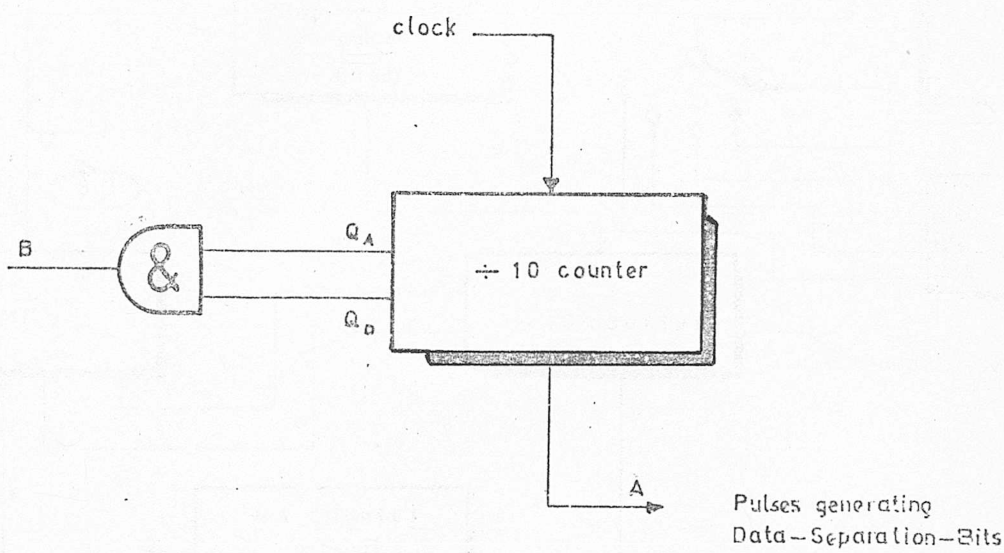


FIGURE 4.9 Pulse Width Regulator

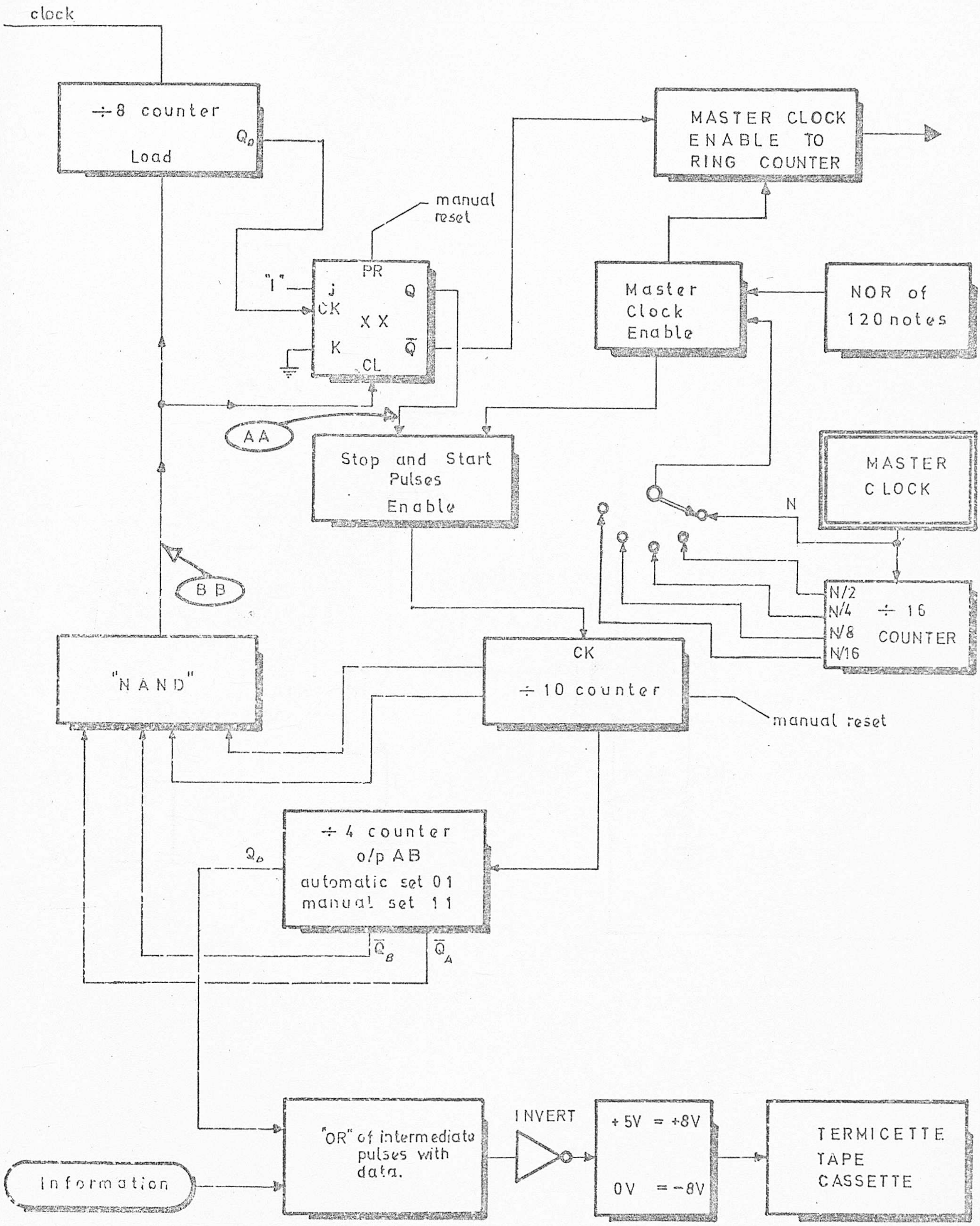


FIGURE 4.10 Pulse Interrupt System Schematic.

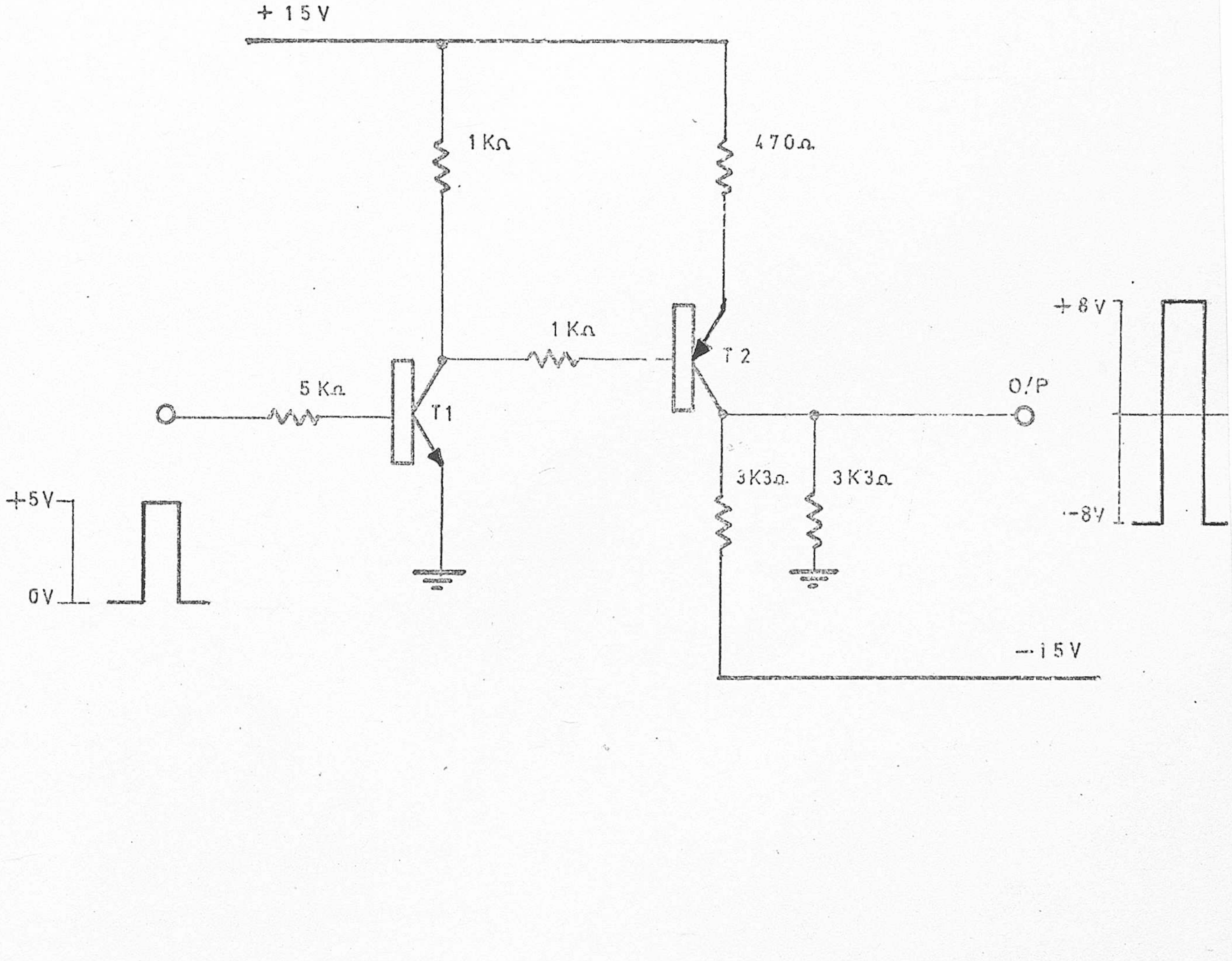
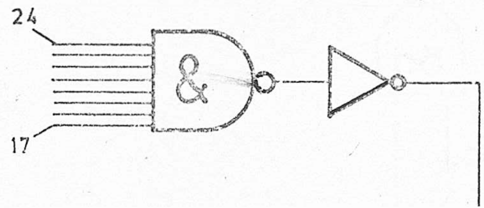


FIGURE 4.11 Voltage Level Shift Circuit


24
&
17

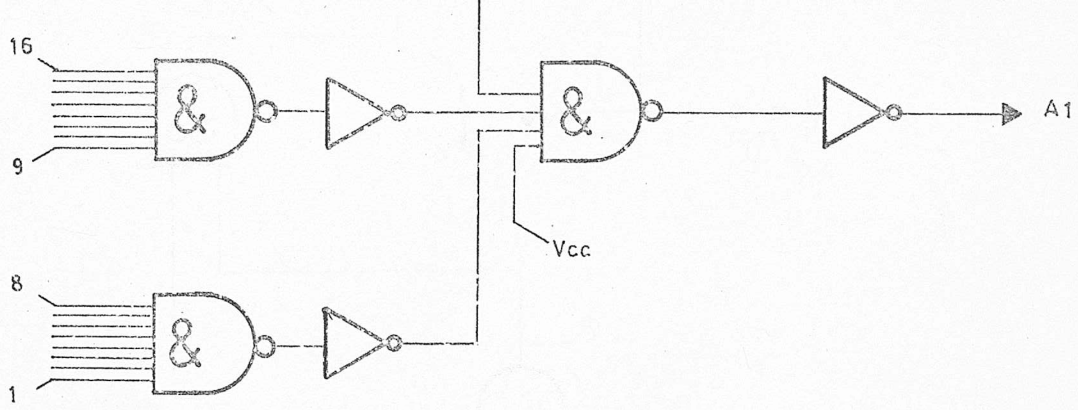


FIGURE 4.12 NOR of Two Octaves

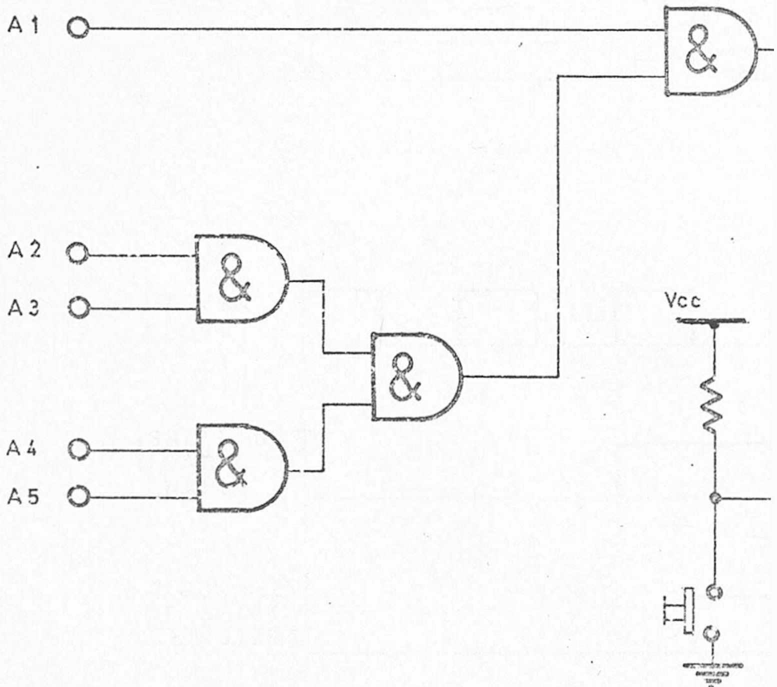
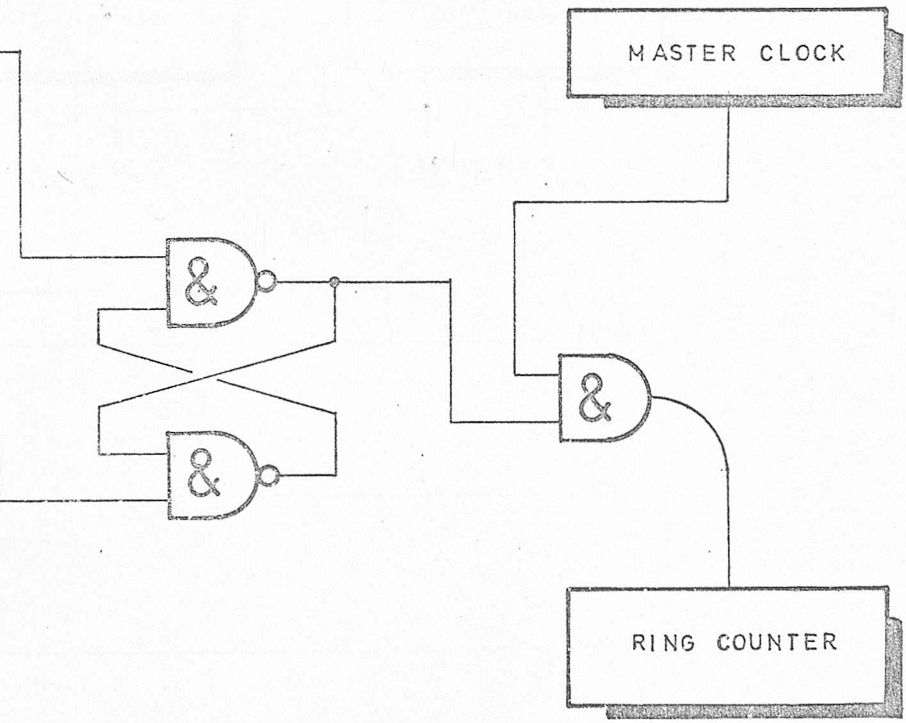


FIGURE 4.13



Automatic Start System

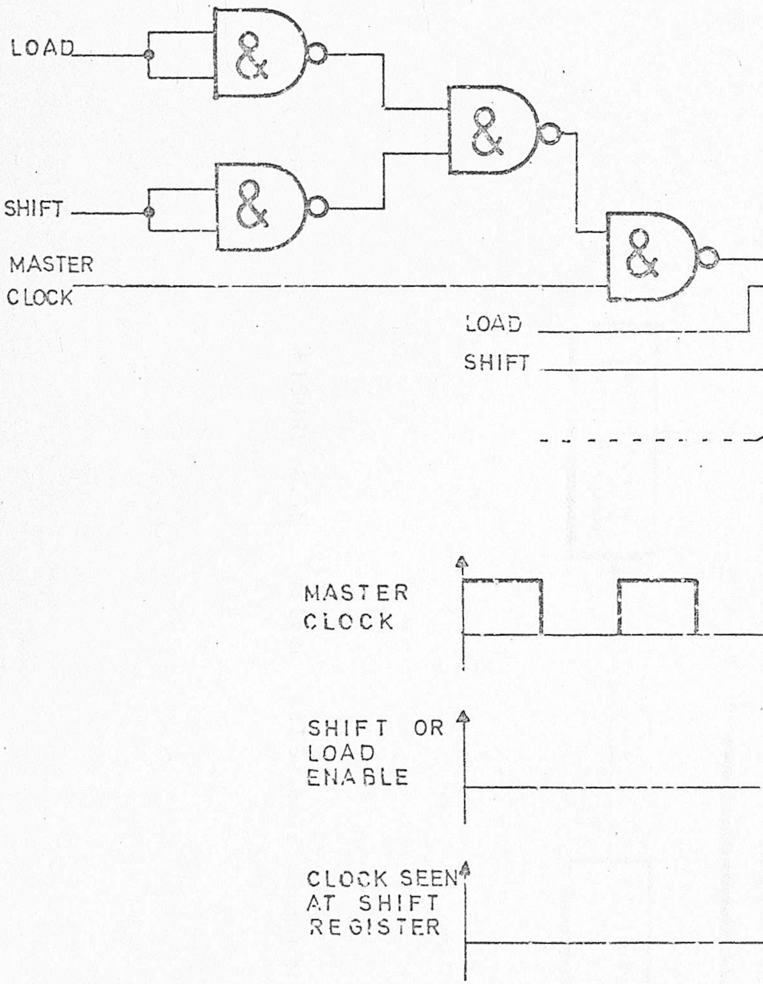
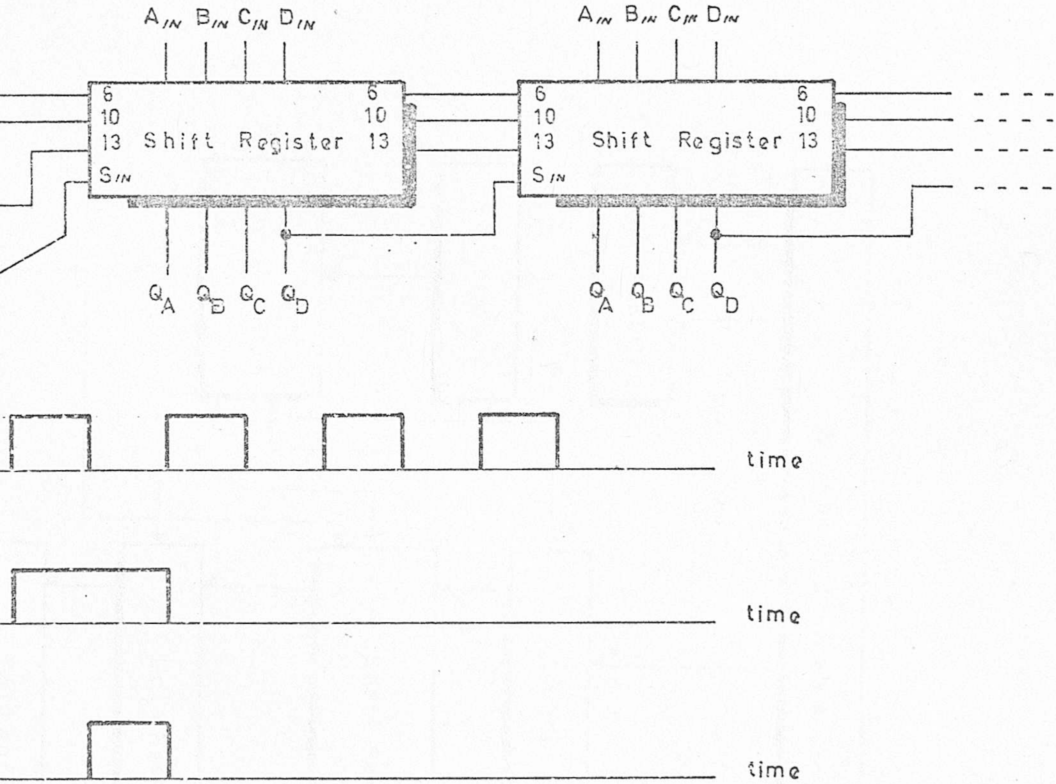

LOAD
SHIFT
MASTER
CLOCK
LOAD
SHIFT
MASTER
CLOCK
SHIFT OR
LOAD
ENABLE
CLOCK SEEN
AT SHIFT
REGISTER

FIGURE 4.14



Shift Register with Clock Waveforms

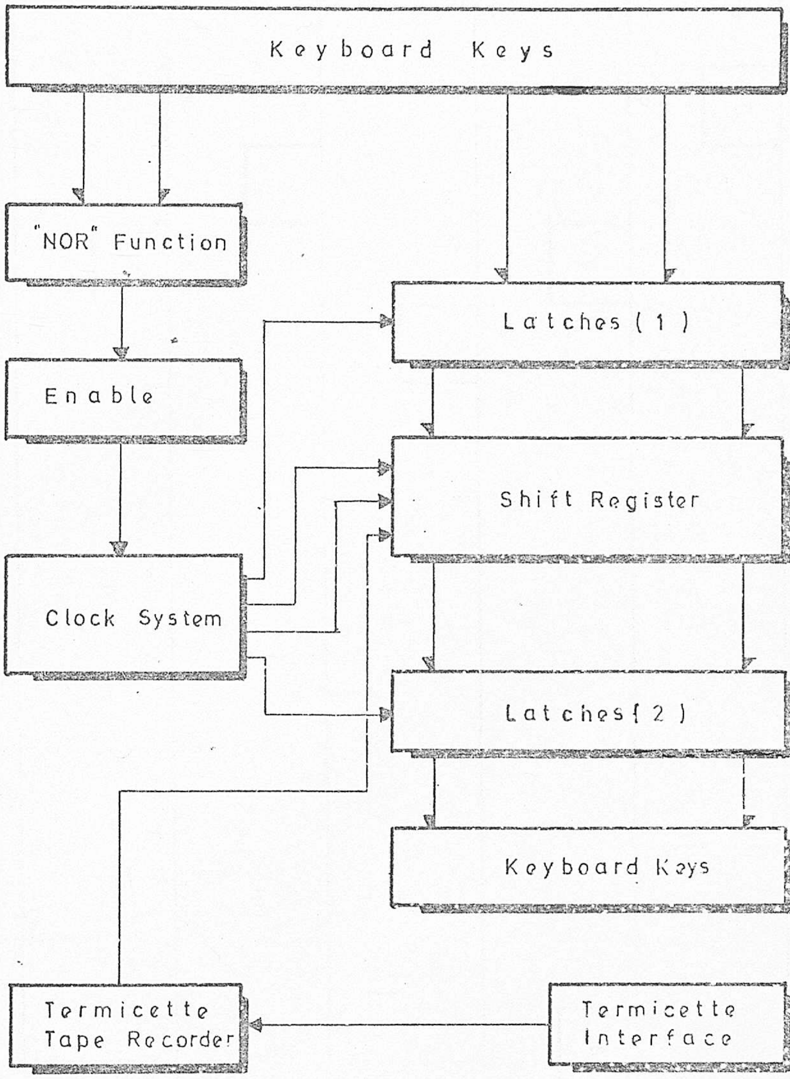


FIGURE 4.15 System Schematic Indicating Playback

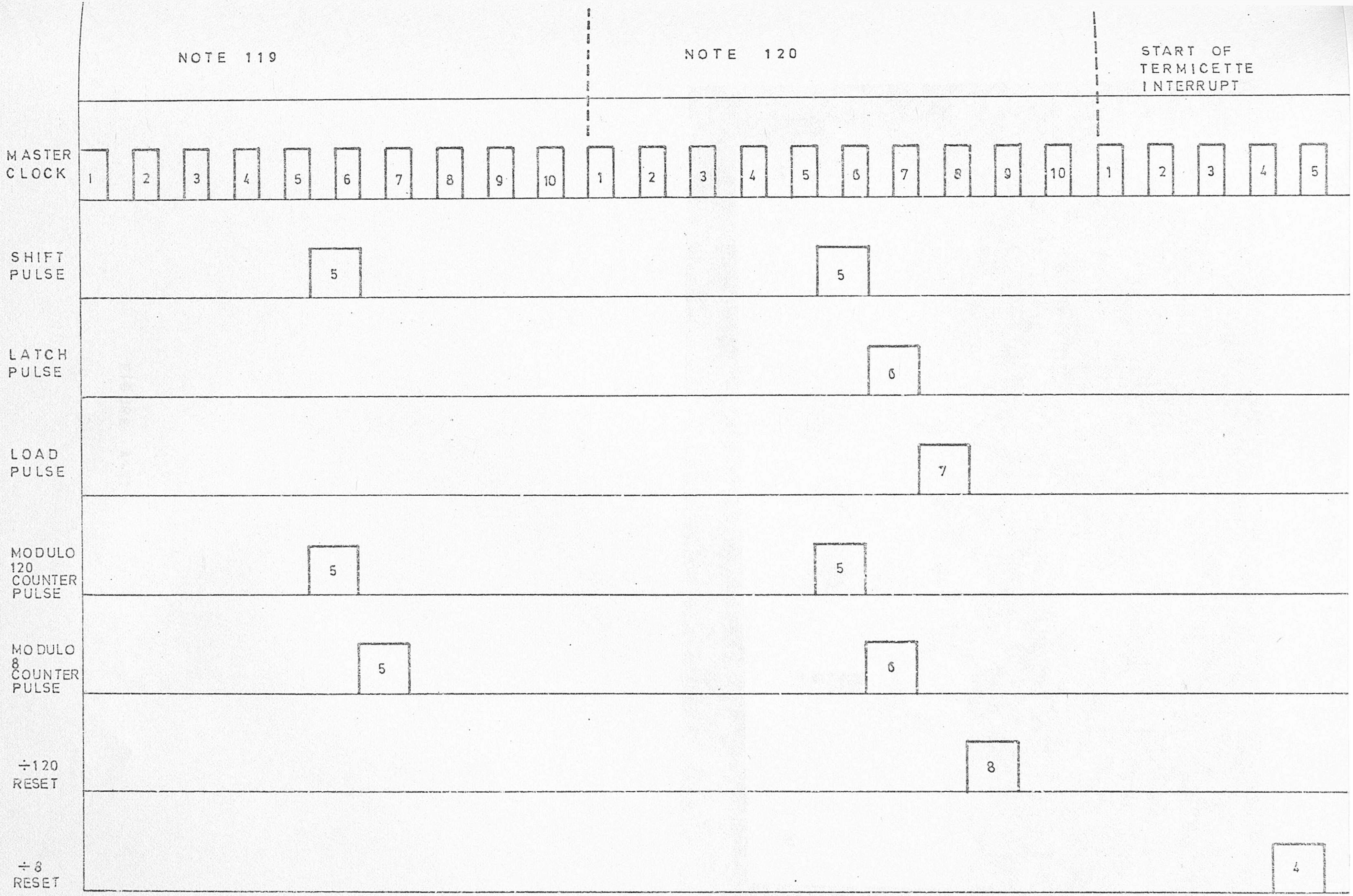


FIGURE 4.16 System Timing Diagram


SAMPLING RATE SELECTOR
N
N/2
N/4
N/8
N/16
AUTO STOP
COUNTER RESET
TERMICETTE RESET
AUTOMATIC TRANSCRIPTION UNIT

FIGURE 4·17

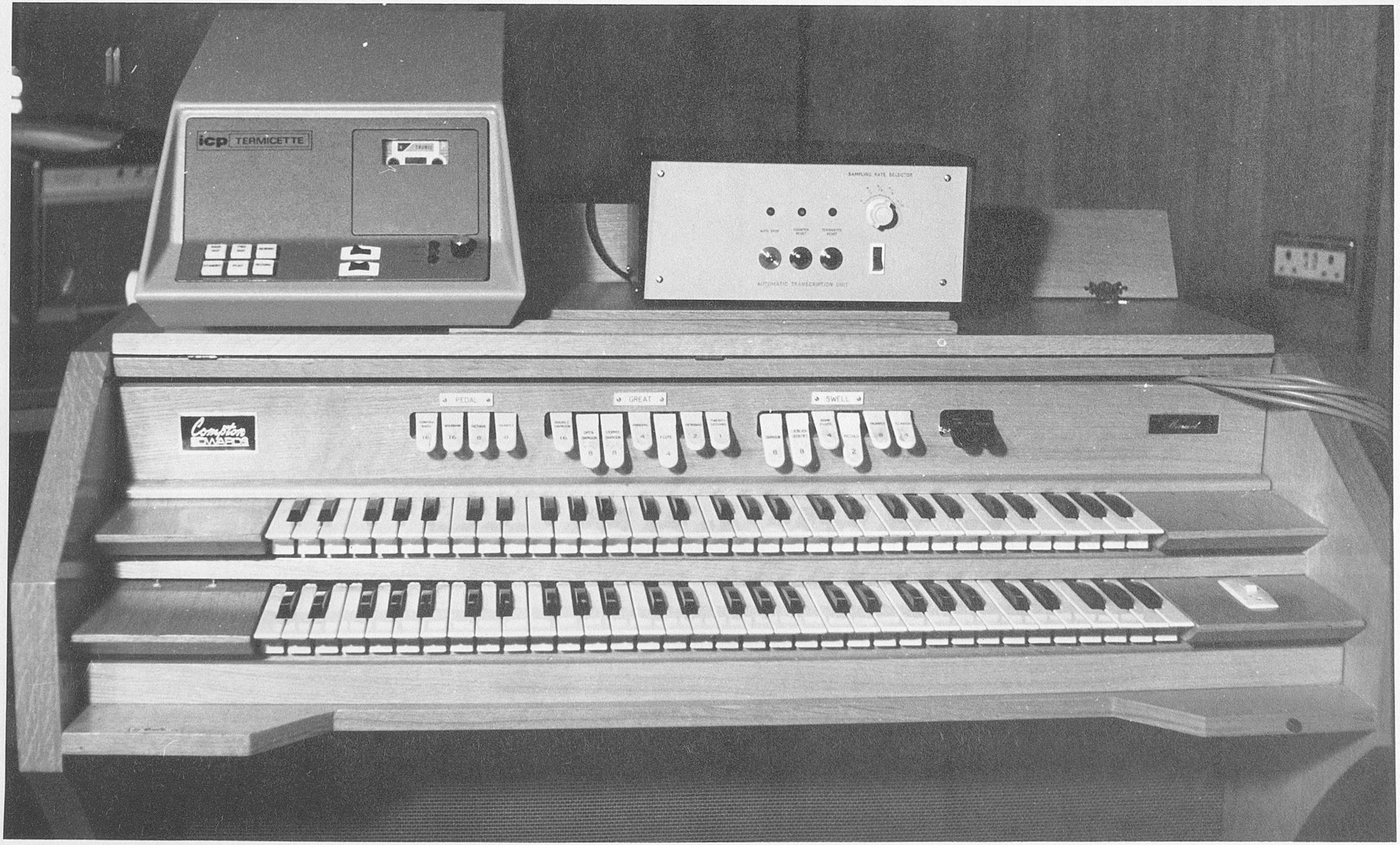

icp TERMICETTE
SAMPLING RATE SELECTOR
AUTOMATIC TRANSCRIPTION UNIT
PEDAL
GREAT
SWELL

FIGURE 4·18

FIGURE 4·19

Chapter 5. Computer Processing of Data

5.1.

The first section deals with the problems involved in data transfer from Termicette to computer. With reference to Fig. 5.1., a signal is sent to the tape cassette which causes an 11 bit word to be presented at the interface of the computer. The signal sent by the computer is a forward skip pulse A, as shown in Fig. 5.2. This causes the Termicette to feed two outputs, XX and YY as shown in Fig. 5.2. Output XX comprises the 8 information bits (D to K), a start bit B and 2 stop bits C. Output YY is a burst of 11 clock pulses used to feed output XX into an 11 bit shift register. The relevant 8 bits (the information bits, D to K) are then buffered using SN7437 Nand Buffer gates and fed to the computer T.T.L. input. For purposes of forming a 120 bit word, 15 such forward skip pulses, A in Fig. 5.2., are sent with a sufficient delay, so as not to cause overlap of any 8 bit word with the next. The 120 bit word is then stored in 10 computer word locations, each word being 12 bits long.

5.2. Alphanumeric Display

The second section in this chapter is concerned with processing the data to give alphanumeric display. As described in the previous section, two 120 bit words are loaded into the computer. The first is read into Array A, the second into Array B as in Fig. 5.3. A check is made for equality between the contents of the arrays. If equality exists, then a counter is incremented by 1 and new information is loaded into Array B. The check for equality continues. Once a state of inequality exists, the contents of array A are printed in alphanumeric form followed by the counter contents. The alphanumeric printout is achieved by measuring along Array/

Array A as in Fig. 5.4. For each logic 1 in the array a note will have been played. Let X equal the corresponding location. Successive subtractions of 12 from X follow until X ≤ 12. The number of subtractions dictates the octave number, N in Fig. 5.4. The resultant X is used in a look up table and X dictates the alphanumeric printout. For example, a logic 1 in locations, 1, 13, 25, 29 and 32 of Array A would result in the following printout:

C0 C1 C2 E2 G2 5

The number 5 represents a possible output from the counter, COUNT in Fig. 5.3. One example of a piece of music transcribed in this form is "SOOZ BLOOZ" by Dudley Moore, and is shown in Fig. 5.5. Other examples are provided in Appendix D. Fig. 5.6. comprises the same example using the staff notation format which will now be discussed.

5.3. Staff Notation Printout

A program has been written in ALGOL for the Elliott 4100 computer in co-ordination with a graph plotter. In order to explain the action of this program a general schematic is given in Fig. 5.7. Procedure STAFF draws the stafflines. Procedure TREBLE plots the treble & bass clefs. Procedure NOTE is responsible for plotting note type and note position. Key signature is plotted using Procedure SGNTRE.

Initially, staff lines are plotted followed by treble & bass clefs. Signature is now added. Note data is read and processed to give a total time value for each note in the first row of data. From this information, the note type is established, its position fixed and the relevant accidental, if any, is added. Procedure NOTE is called for each note in the first row, the notes are plotted and more data is entered. A check is made for the/

the requirement of new staff lines and the process continues.

With reference to Fig. 5.8., Procedure STAFF, when called plots 4 sets of 5 horizontal parallel lines with the relevant spacing between each set in accordance with accepted staff notation. The space between each line within a set is 0.3 cms thus determining the minor diameter of the ellipse to be drawn for each note. With the staff lines plotted, the first two staves require the treble clef sign, the lower pair requiring the bass clef sign. The treble clef is explained with the aid of Fig. 5.9. An arc of 210° is plotted at a radius of 0.2 cms, then from a new centre point another arc of 210° is drawn at a radius of 0.3 cms. This gives the effect of a spiral. A diagonal line is then plotted to the highest staff line in the set. A semi-circle is subsequently plotted, continuing with a vertical straight-line 1.5 cms long to bisect the spiral. With centre D as shown in Fig. 5.9., an arc of 180° is plotted and a small circle centre E is then added and finally filled. The result is the treble clef sign. In order to display the bass clef sign, the following procedure must be used. With reference to Fig. 5.10., a circle of radius 0.1 cms is drawn then filled. At new centre G, a 210° arc is drawn, stemming from the circle. A straight line is plotted at a tangent of 1.5 with respect to the Y axis followed by a line of tangent 0.75 with respect to the Y axis. By this means, the bass clef sign is displayed.

Key signature is now added. With reference to Fig. 5.11. a number X2 is read into the procedure SGNTRE. The number is 0 for key "C". For a sharp key, X2 is the number of sharps and is given a negative sign. Similarly for a flat key, X2 is the number of flats, but is given a positive sign. Each accidental has its own co-ordinates preset in procedure SGNTRE and the/

the variable X2 only governs how many will be plotted. The final procedure is procedure NOTE (K1, K2, K3, K4, YN). It requires the following information. The variable K1 equals the note number ($1 \leq K1 \leq 120$); K2 determines the number of quaver tails to be plotted; K3 is responsible for type of accidental if any; from K4, time plus a half notes are "dotted" and YN determines the distance along the stave. As in Fig. 5.12., note pitch is determined by subtracting 12 from K1 until K1 no longer exceeds 12. The number of subtractions, multiplied by 1.05 cms, plus a value for the resultant K1, gives the pitch value. The value for K1 is taken from a look up array of values which depend on key signature. Thus a set of values for X and Y coordinates are resolved and around this point (XNC, YNC), an ellipse is drawn with major axis radius 0.2 cms, minor axis radius 0.15 cms. From K2, the note type is established and for minim or shorter time value, a tail is drawn. For crotchet or shorter time the note is filled. This is accomplished by plotting (YNC+.2COS(AN),XNC+0.15SIN(AN)) alternately with (YNC+0.2 COS(π-AN),XNC+0.15SIN(π-AN)). The angle, AN, steps from -90° to $+90^\circ$ thereby filling the ellipse. The value of K2 dictates the number of quaver tails, if any, from one for a quaver to four for a semi-demi-semiquaver. Time plus a half notes are "dotted", this being governed by variable K4. Leger lines are added to notes between staves and finally, the relevant accidental is drawn, determined by K3. This is read from an array of accidental values synonymous with PITCH.

In the main part of the program, the first stage is to read the minimum and maximum values for a semi-demi-semiquaver. From these the other note-type values are calculated as shown in Fig. 5.13. The value K6 is then read, this governing the key signature, followed by the note data being entered into arrays for storage. Twenty lines of data are/

held in storage at any one instant and it is considered that no more than twenty notes will ever be played at one instant. The arrays must be formed each 20 x 20, one array A containing the note values, the other B, that relevant note's time value. The former must also contain information regarding the number of notes in any row. In conclusion, Array A is chosen A (1 : 20, 1 : 21) and Array B (1 : 20, 1 : 20) as in Fig. 5.14. A line of data consists of the number of notes, NV, their values and the time value, NT. Nineteen such data lines are read in followed by a separate "READ" statement for the twentieth. Once the first row of data is read it is processed then plotted, followed by all rows A, B (x,y) being stepped up to A,B (x-1,y), the first row is overwritten and new data is introduced into the 20th row. The number 21 is a terminating number and once A (20,21) = 21, no more data is read. Once it appears in A (1,21) then the program has finished as shown in Fig. 5.14.

The data is processed in the following manner. With reference to Fig. 5.15. for each note in the first row, a search is made through the second row for equality of note number in that row. If equality exists, then the time value for that note in the second row is added to that of the first, the note plus time value is deleted and the number of notes becomes one less. For that same note, the process is repeated through the third row on until no equality exists and a total time value for that note's duration lies in the first row of Array B. This is repeated for all the other non-zero values of the first row. These notes are now ready to be plotted. With reference to Fig. 5.16., the time value for each note in Array B (1,Y) is compared with the time intervals as given in Fig. 5.13. giving a time value for K2. K1, the pitch value, equals the note number. K3, the accidental, is derived from a look up Array in conjunction with K1 and K4/

K4 is again decided from the values given in Fig. 5.13. One further set of restrictions prevail; If the note is less than 12 units smaller than the note above then no quaver tails must be drawn and if the note value is less than four units smaller, the note must be stepped to the right hand side of the note tail, the convention chosen being all tails upwards, to avoid confusion on this point of detail. Procedure note is called for each non-zero value in the first row of Array A. At the end of each line of music new staff lines are drawn with clef and signature added. The process continues until all the data has been plotted.

The program listing for staff notation printint is given in Appendix C.

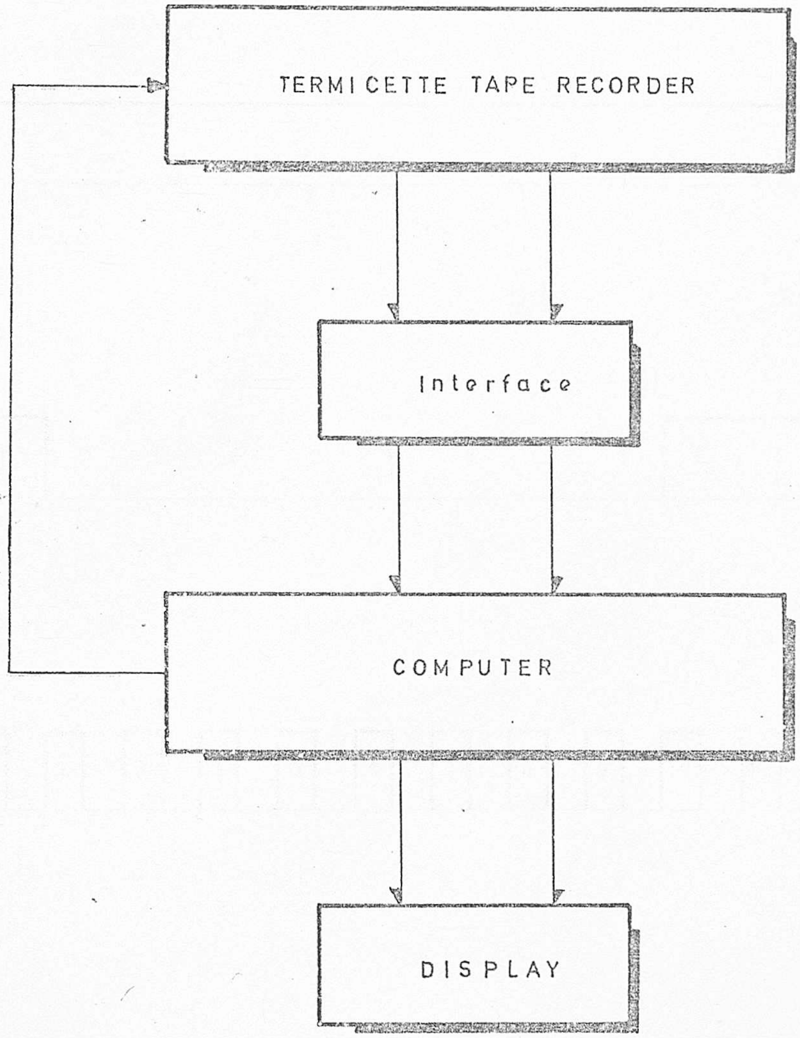


FIGURE 5.1 Schematic of Data Processing

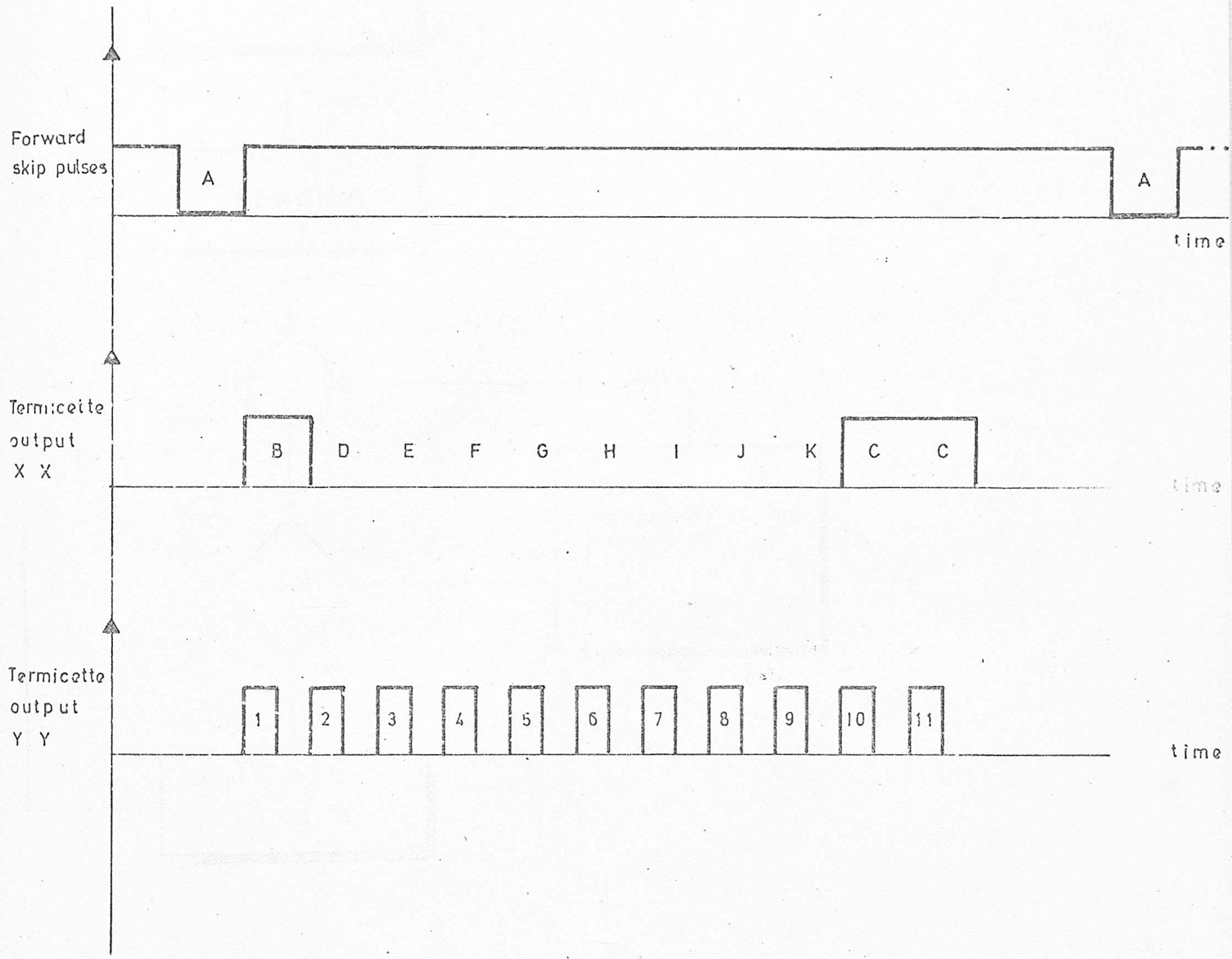


FIGURE 5.2 Waveforms involved in Data Retrieval

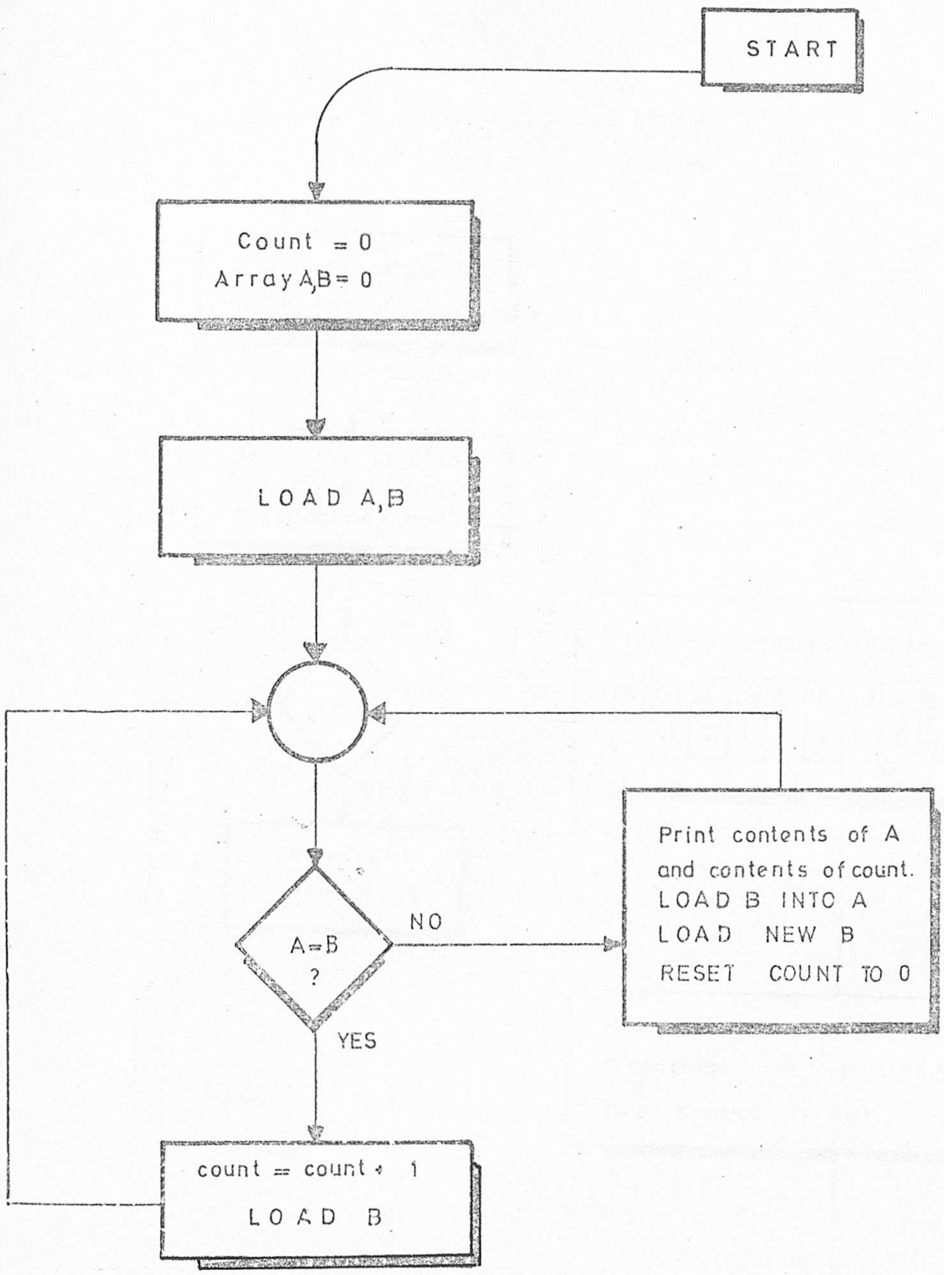


FIGURE 5.3 Process data for Alphanumeric Printout

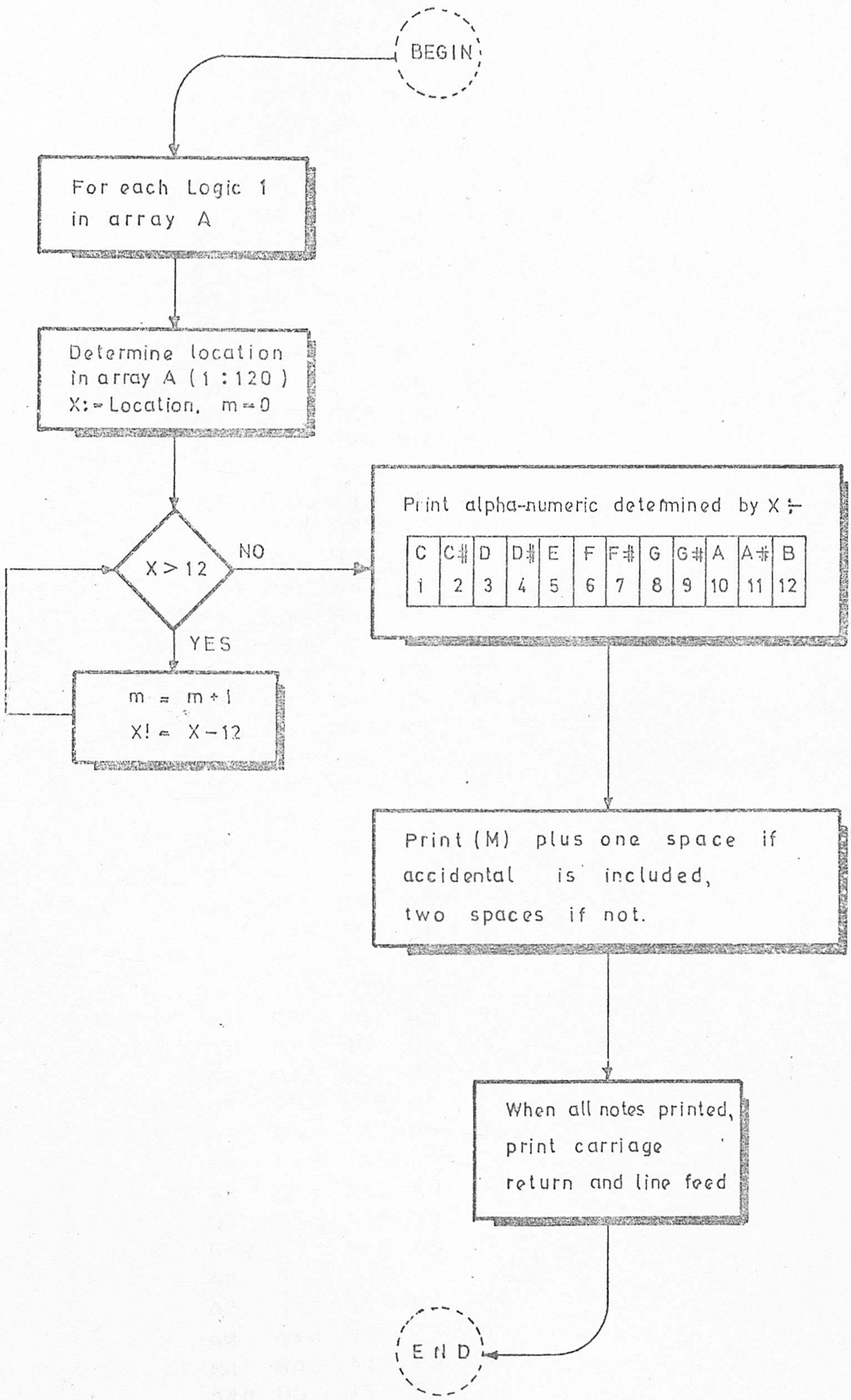


FIGURE 5.4 Alphanumeric Printout

SOOZ BLOOZ. D MOORE.

```
A3   5
A3   G4   B4   E5   5
A3   A#4  D5   G5   5
E4   G4   B4   E5   5
A#4  D5   G5   5
E4   G4   B4   E5   5
A3   A#4  D5   G5   5
E3   A#4  D5   G5   5
A3   B4   C#5  E5   15
E4   B4   C#5  E5   5
B4   C#5  E5   5
E4   B4   C#5  E5   5
A3   B4   C#5  E5   5
E3   B4   C#5  E5   5
A3 5
A3   G4   B4   E5   5
A3   A#4  D5   G5   5
E4   G4   B4   E5   5
A#4  D5   G5   5
E4   G4   B4   E5   5
A3   A#4  D5   G5   5
E4   A#4  D5   G5   5
A3   B4   C#5  E5   15
E4   B4   C#5  E5   5
B4   C#5  E5   5
E4   B4   C#5  E5   5
A3   B4   C#5  E5   5
E3   B4   C#5   E5   5
D3   5
D3   C5   E5   A5   5
D3   D#5  G5   C6   5
A3   C5   E5   A5   5
D#5   G5   C6   5
A3   C5   E5   A5   5
D3   D#5  G5   C6   5
A2   D#5  G5   C6   5
D3   E5   F#5  A5   15
A3   E5   F#5  A5   5
E5   F#5   A5   5
A3   E5   F#5  A5   5
D3   E5   F#5  A5   5
G#2  E5   F#5  A5   5
A2   5
A2   G4   B4   E5   5
A2   A#4  D5   G5   5
E4   G4   B4   E5   5
A#4  D5   G5   5
E4   G4   B4   E5   5
A3   A#4  D5   G5   5
```

FIGURE 5·5

Sooz Blooz

Dudley Moore

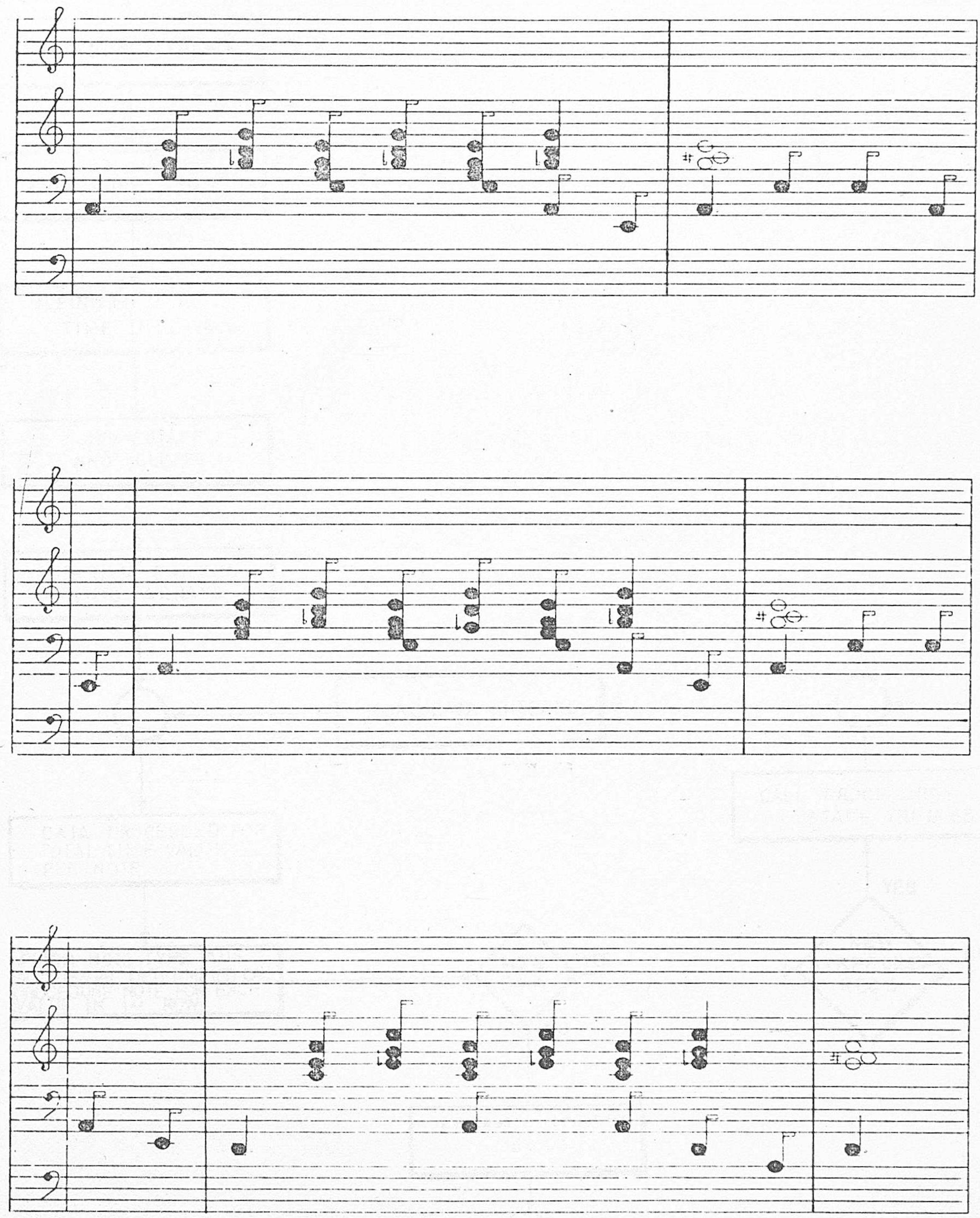



FIGURE 5·6

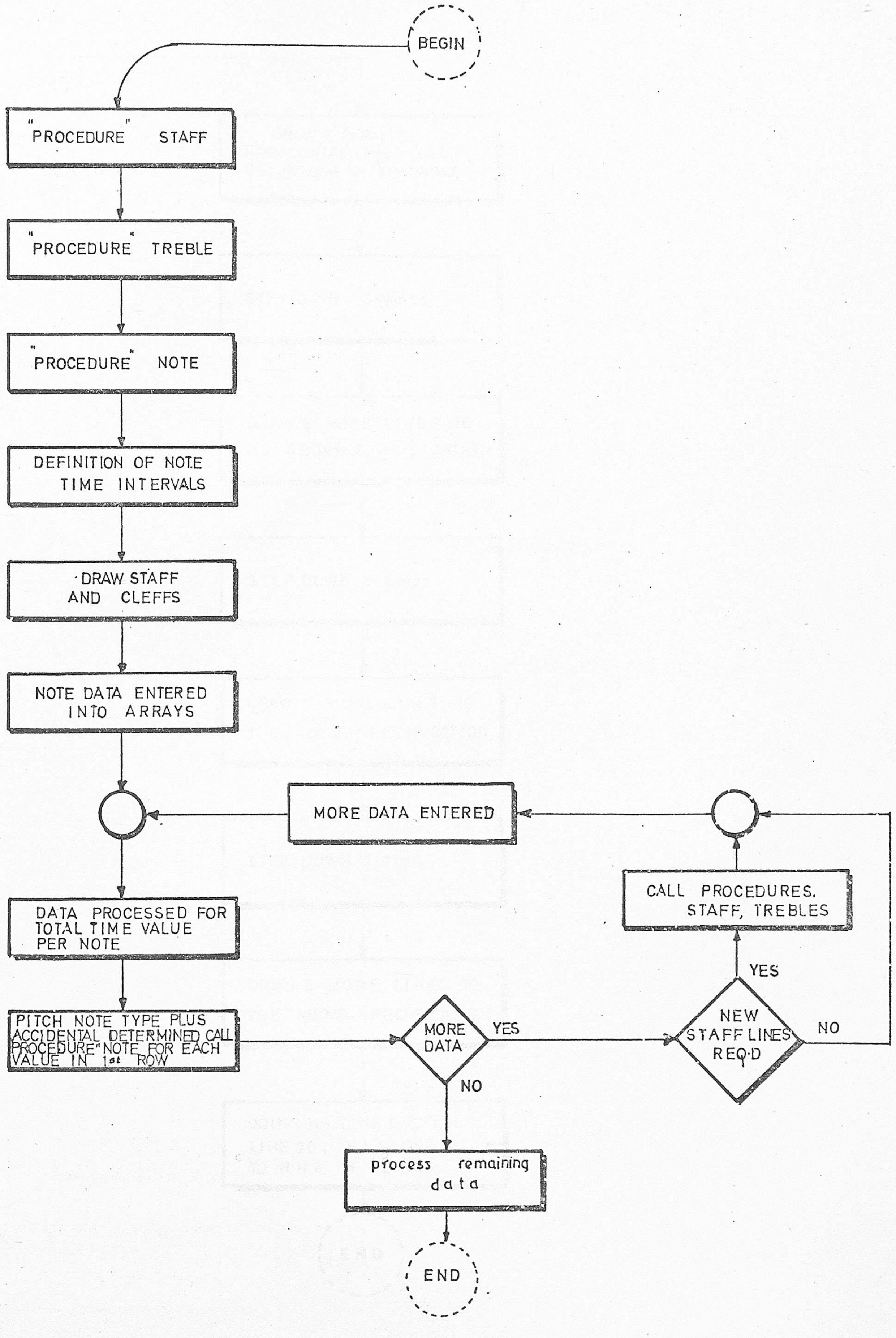


FIGURE 5.7 Program Block Schematic

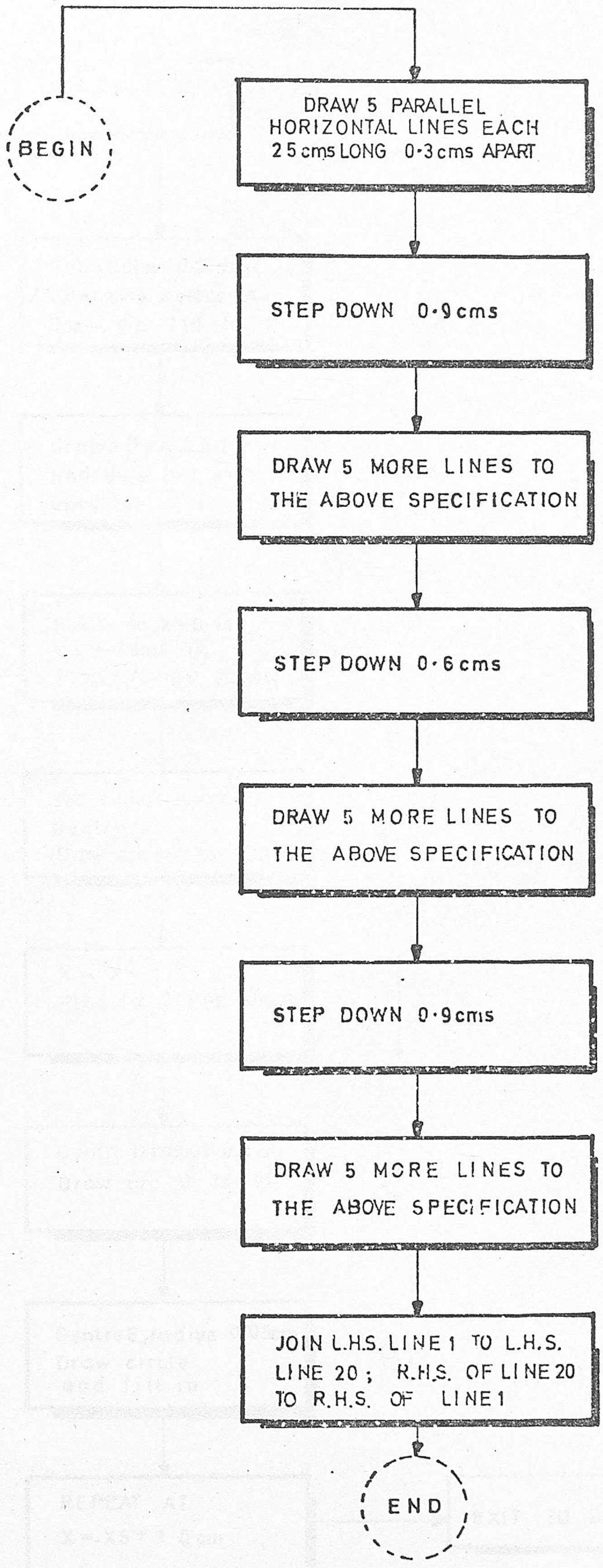


FIGURE 5.8 "PROCEDURE" STAFF

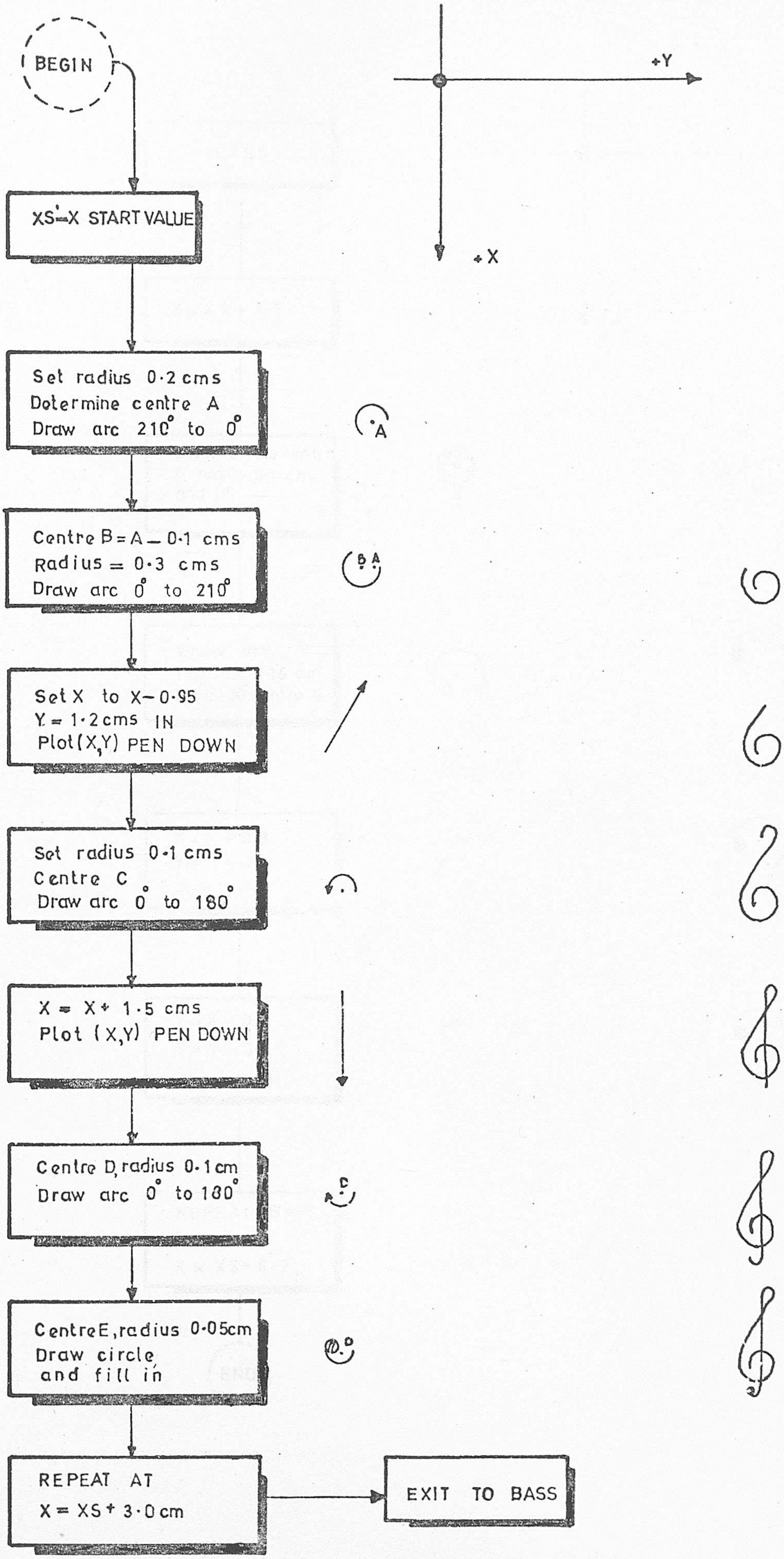


FIGURE 5.9 "PROCEDURE" TREBLE

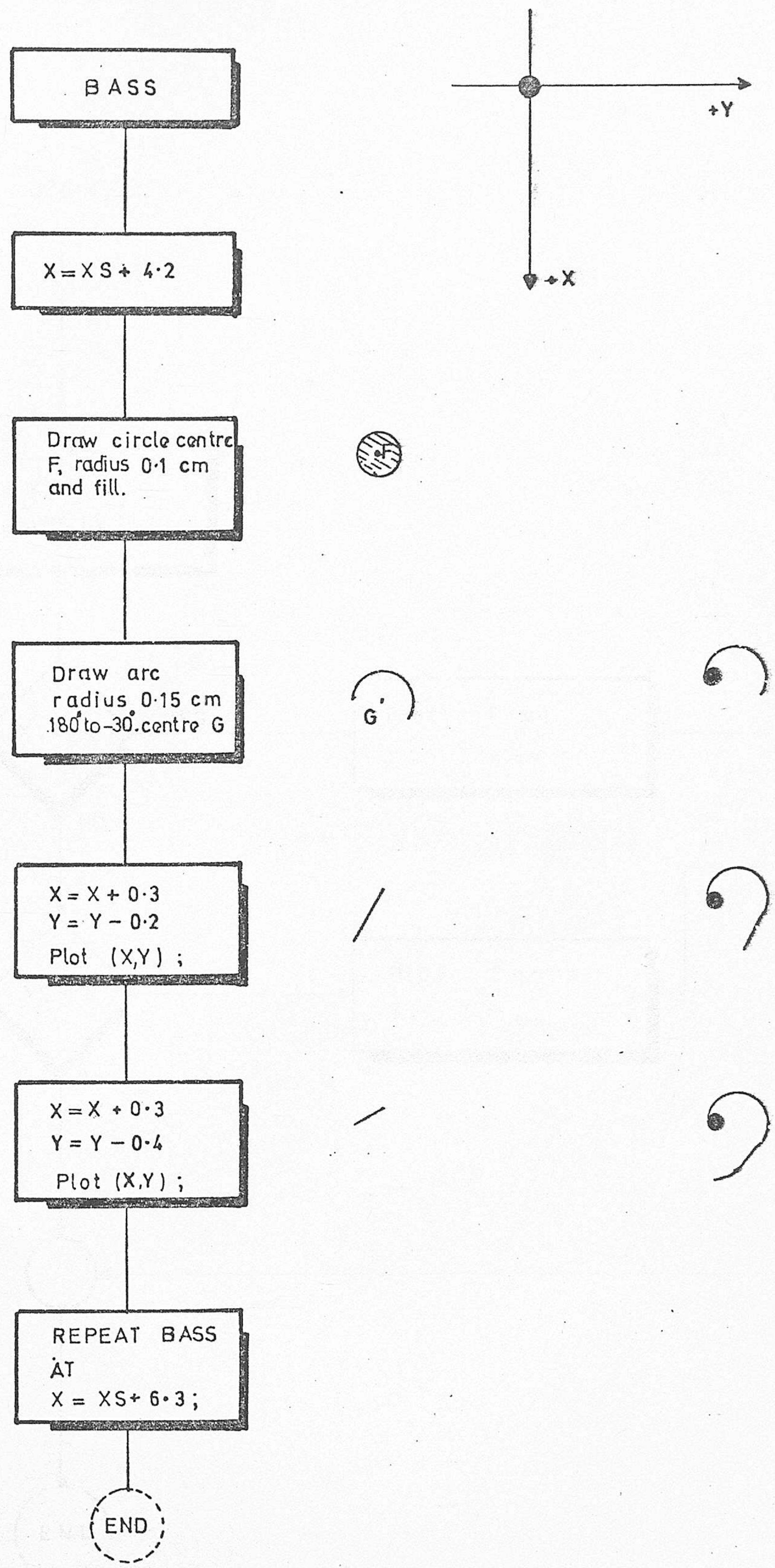


FIGURE 5.10 "PROCEDURE" BASS

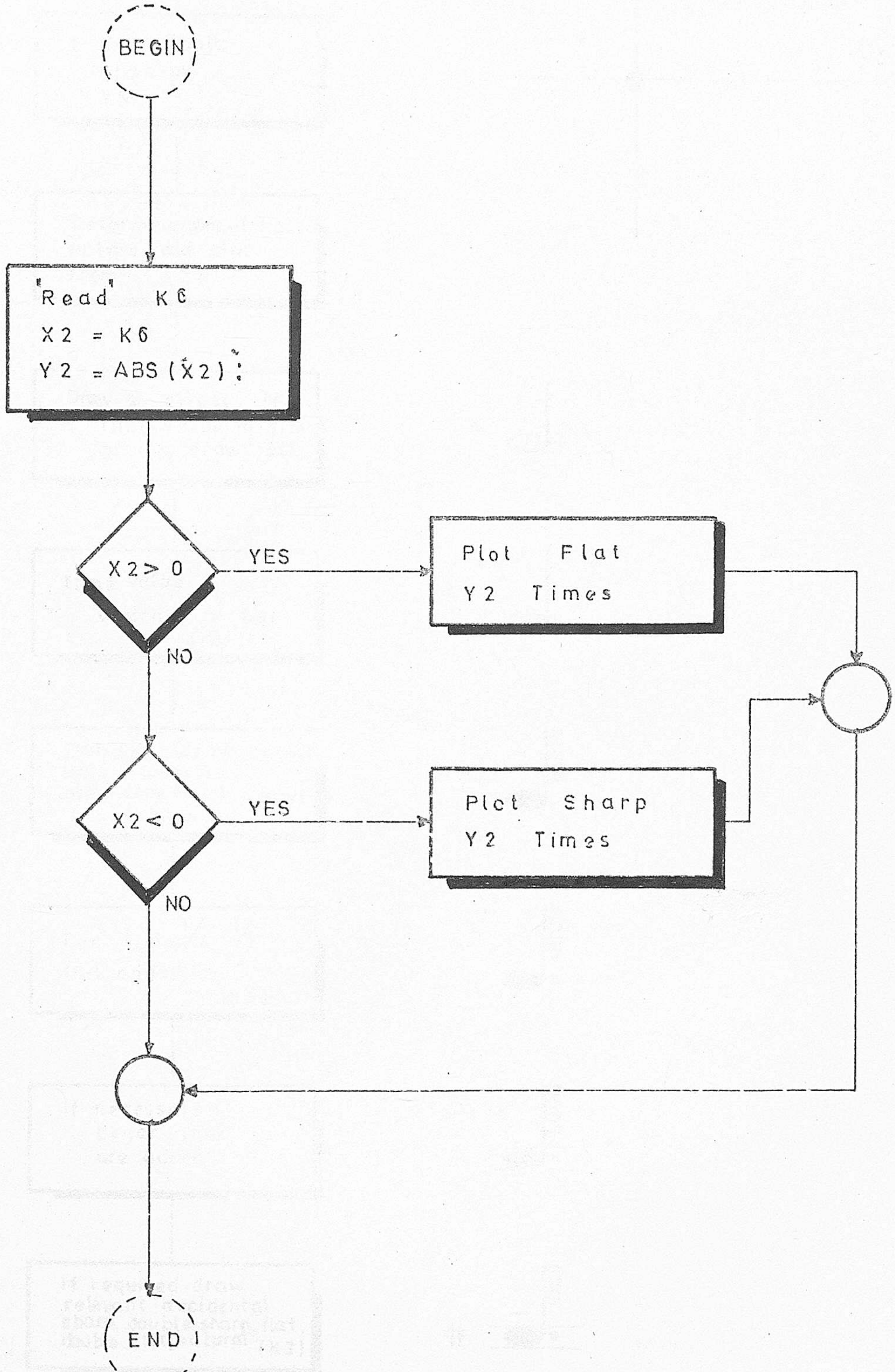


FIGURE 5.11 "PROCEDURE" SGNTRE

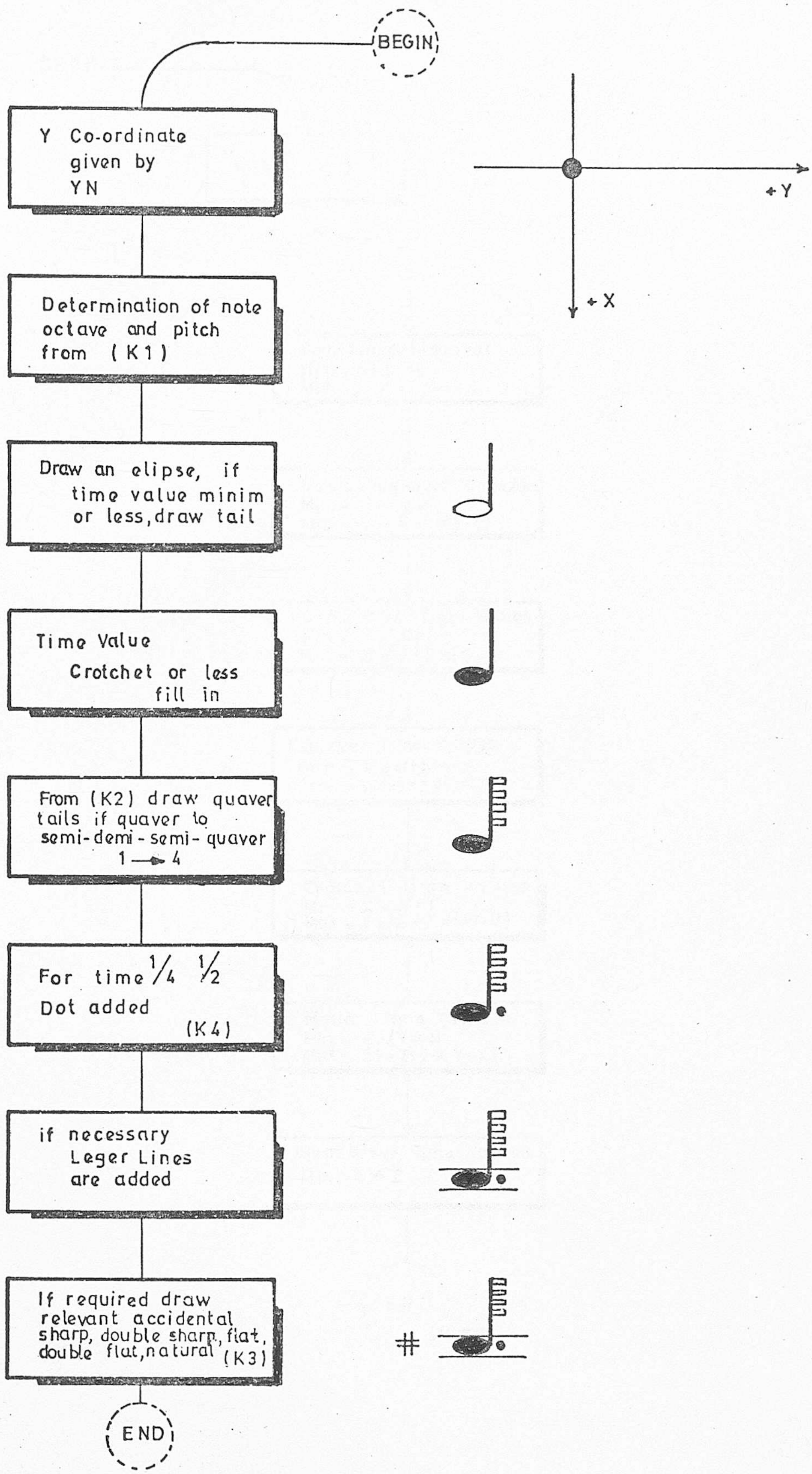


FIGURE 5.12 "PROCEDURE" NOTE

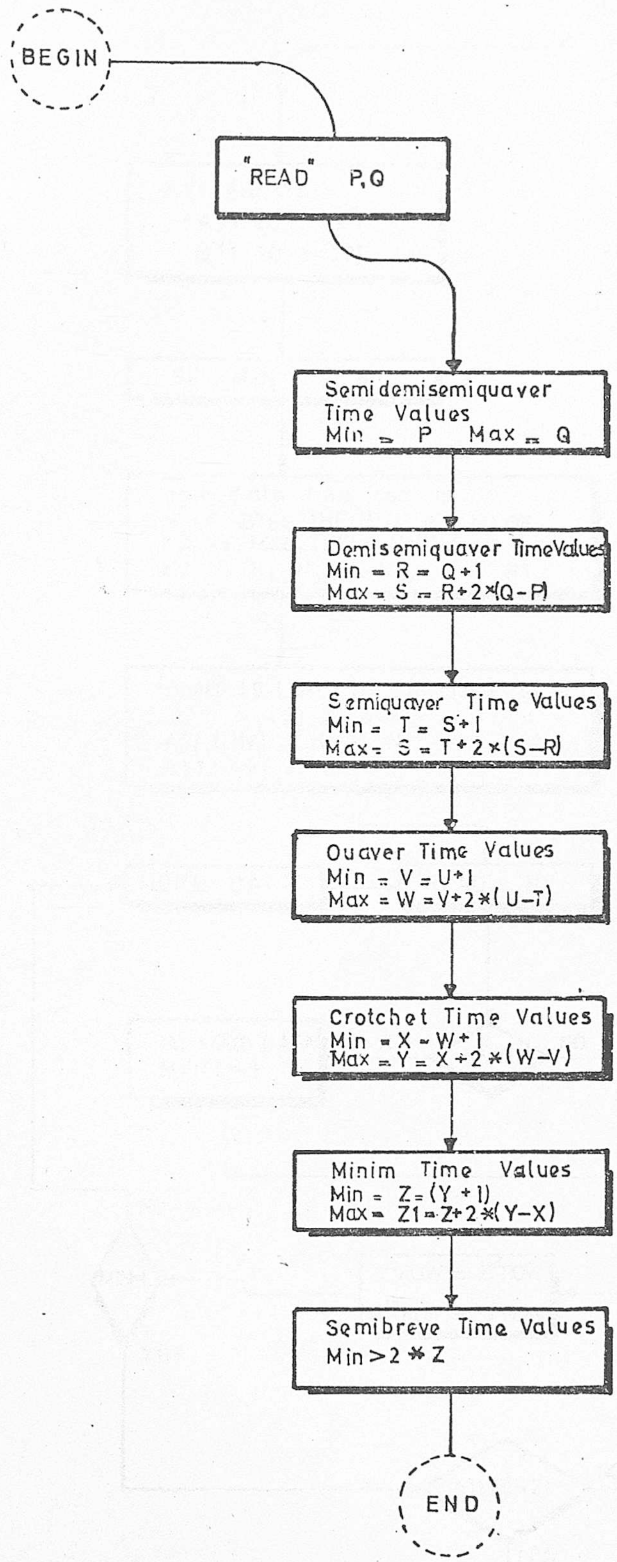


FIGURE 5.13 Calculation of Note–Type Time Values

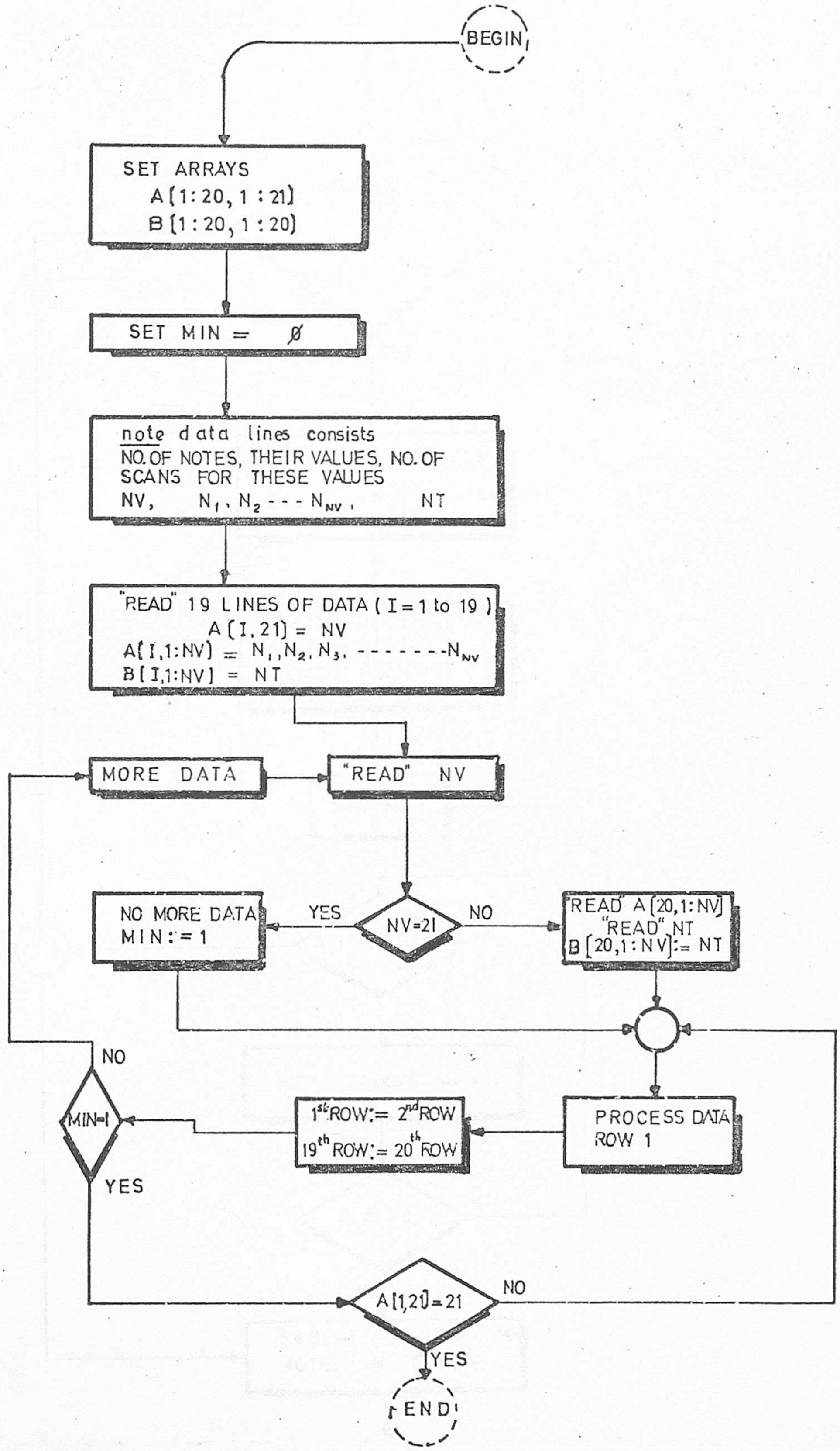


FIGURE 5.14 NOTE DATA ENTERED INTO ARRAYS

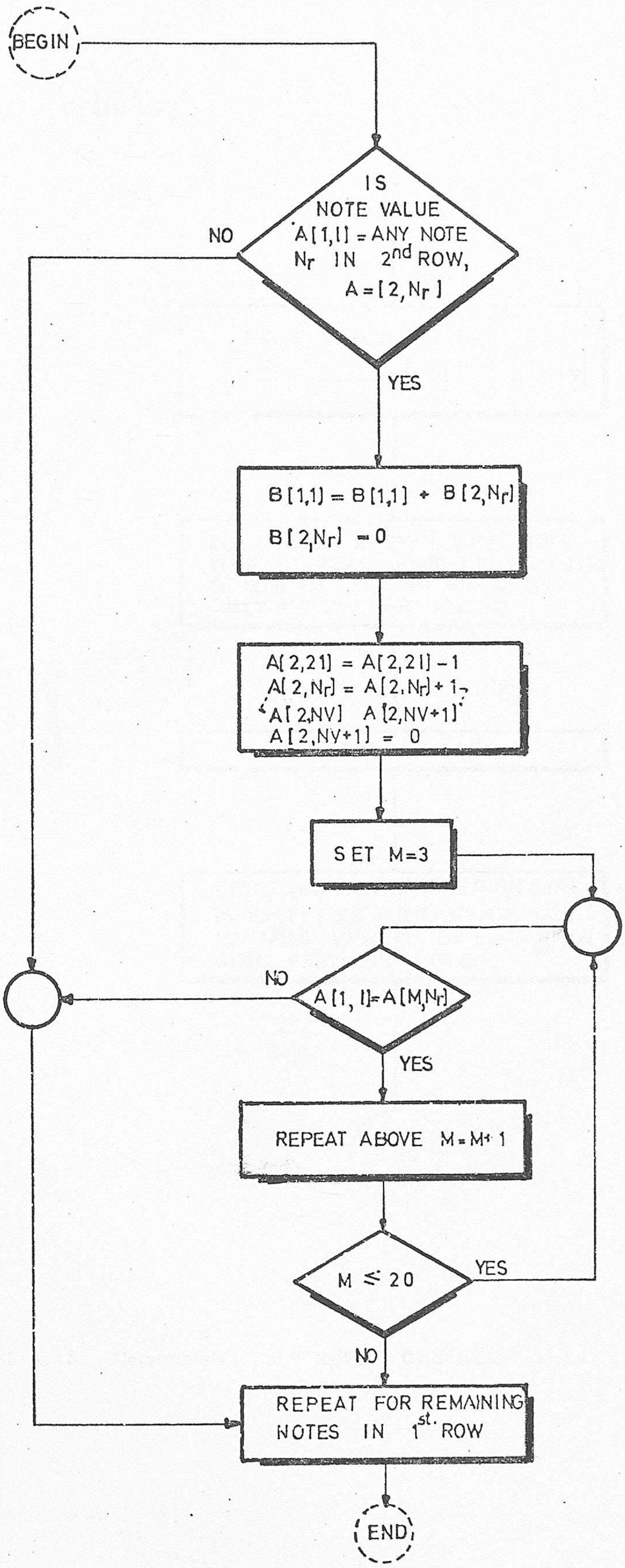


FIGURE 5.15 Data processed for Total Time Values

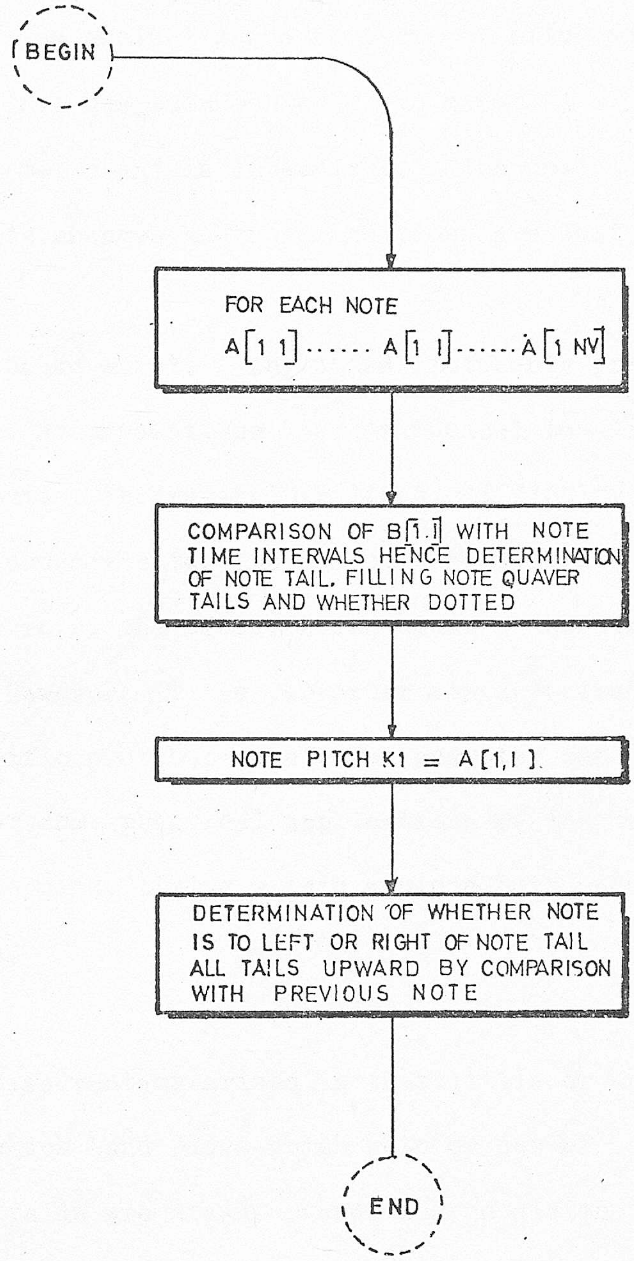


FIGURE 5.16 Determination of Note Characteristics

Chapter 6. Conclusions and Further Work

The system was designed to monitor the action of a keyboard then to give a display of the musical score in terms of pitch and the time value for the notes. This has been achieved and examples of the display are given both in Chapter 5 and in Appendix D. The unit is capable of monitoring all key state changes and no transitions are lost.

Certain drawbacks exist. Any notes mistakenly pressed will also be transcribed since, at present, no laws of musical tonality are used to govern the printout. In drawing bar lines, difficulty might arise as for a given crotchet count/minute, the musician will seldom follow that count exactly. Time errors, therefore, arise causing the bar lines to be incorrectly placed. However, if the performer stays tolerably within the constraint of a specific crotchet count, the computer can draw bar lines. Unfortunately, for some practical applications of the transcription machine, e.g., avante garde jazz, timing within a piece both modulates and can have random variations.

A further disadvantage arises in that little or no discrimination can be made between which hand plays which note or set of notes. For this reason, all note tails are drawn upwards and no distinction is made between lower and upper hand in the printout. For similar reasons, no rests are printed as it is impossible, for a particular piece, to ascertain individual voicings. Thus rests must be added, subsequent to the automatic transcription, by the composer. No expression is added to the display as the system does not monitor the speed at which any note is pressed. The mathematical modelling of expression is an unsolved problem and thus cannot be efficiently computerised. The system does, however, produce an accurate account of the/

the musical score and factors such as bar lines and expression can be added by the composer.

It has been noted previously that the system gives the option of the pre-specification of a key signature. Unfortunately, many compositions involve key modulations. It is not practical to account for these modulations during an on-line performance. Thus this information must be subsequently added off line.

A recent addition to the transcription program is the option of compensating in part for human error. When playing a chord, a musician may press the majority of the notes before the others. If a literal transcription is required, then the notes are printed exactly as played. If compensation is required for this small delay, a search is made for the following condition. The condition is that less notes exist in one line of data than the next, the time value for the first data line is small and the notes in the first line of data exist in the next. If such a state exists, the missing notes are added to the original chord and a printout is made of the full chord.

Future work on the transcription program could involve introduction of the laws of music with ways of determining more sophisticated note types such as triads, quaver strings. Account could be taken of compound beats and more intricate rhythms. An outlet for future research is to reduce the cost of the project by replacing the Termicette with a domestic stereo cassette. The requirement is to be able to stop and start the cassette tape without loss of information. One possible answer is to use switched track recording. On one track, the information is recorded, on the other/

other a clock track. After a time t , where (t x the recording rate) is less than the available core storage, the information and clock tracks are exchanged and recording continues. On exchange, an extra pulse Y is added to the original information track. On playback, information is derived until pulse Y is obtained. Information now in store is processed, during which time the tape is given a slight rewind. When the computer requires more data, playback is enabled, but, no data is permitted until recipt is made of the pulse Y. Control of the cassette automatically would pose few problems and a less expensive storage device would be obtained. The cost would also be reduced with the addition of a matrix plotter to give the display. This would be of use if a hardware approach were used to process the data.

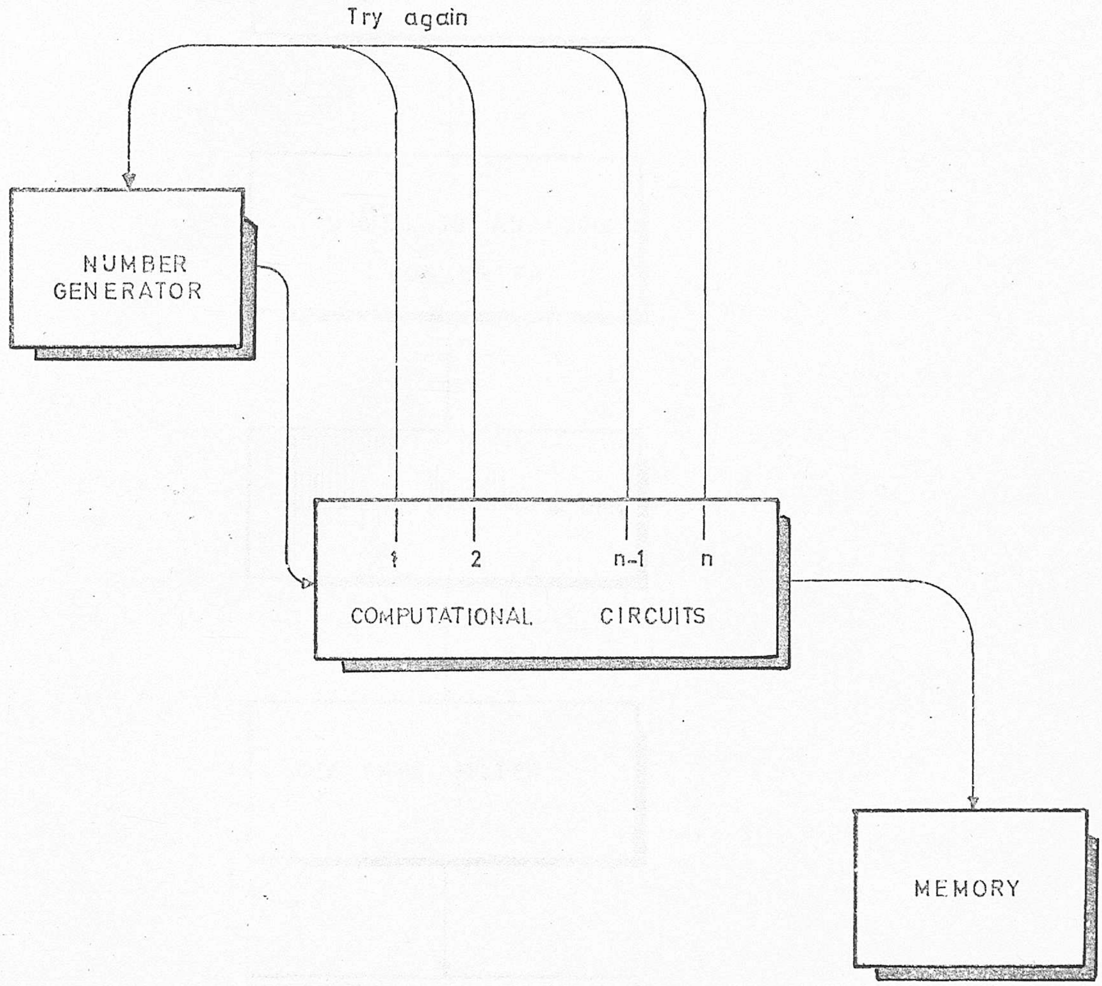
Possible research could be made into linking the Transcription Unit to a Stochastic computer, with aspect to automatic composition, as with the Illiac digital computer (2). Use of randomly generated numbers is ideally suited to the stochastic computer. Two possible approaches exist. One is to set up a random number generator as in Fig. 6.1. The numbers are fed through n computional circuits. If the musical progression is tonal within the laws set, the notes are stored in memory. Otherwise, the try again routine is enabled. The process is continued until the composition is complete. Each number would, in this case, represent a note or chord.

An alternative method is shown in Fig. 6.2. The random number generator outputs digits which are converted to analog samples. On being passed through a low pass filter, the output is an a.c. signal which is amplified and fed into a loudspeaker. In both cases, rules are established, but, the computer is random within these limits.

An/

An extension on the previous suggestions is that the computer could be of use in semi-automated composition. The computer would store a list of possible chord progressions which fit the melodic line and for a given melodic line, different emphasis would be set on notes due to a change in chord structure. Such exploration enters the realm of jazz harmony production.

The system is in itself an aid to the musician and in no way attempts to replace the composer. As a stepping stone to future research, a base has been set for further investigation into the scientific qualities of music.



Try again
NUMBER GENERATOR
1
2
n-1
n
COMPUTATIONAL CIRCUITS
MEMORY

FIGURE 6·1

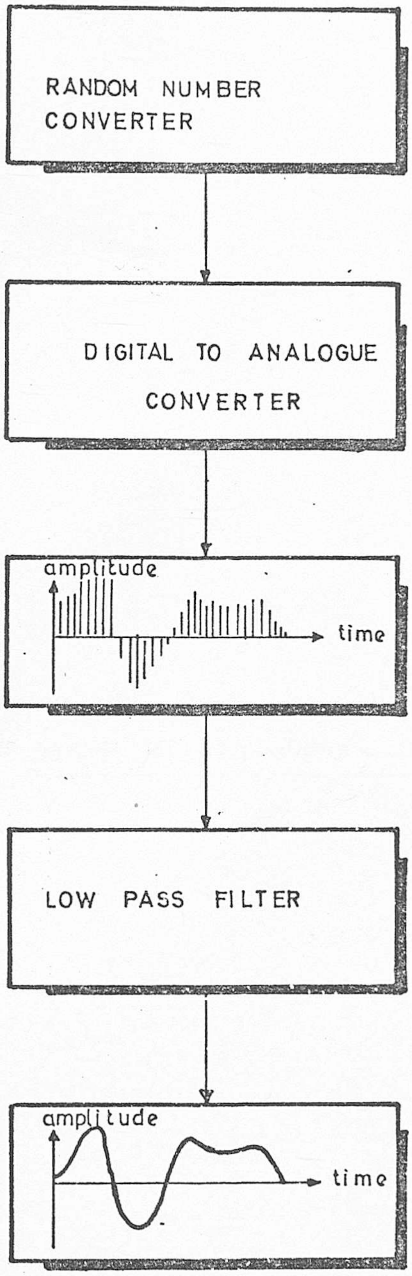

RANDOM NUMBER
CONVERTER
DIGITAL TO ANALOGUE
CONVERTER
amplitude
time
LOW PASS FILTER
amplitude
time

FIGURE 6.2

APPENDIX A

Program for Alphanumeric Printout

```
                *2Ø
ØØ2Ø   7754   M24,      -24
ØØ21   ØØØØ   CNT,      Ø
ØØ22   Ø6ØØ   PNTR,     AST
ØØ23   ØØØØ   PNT,      Ø
ØØ24   4ØØØ   K4ØØØ,    4ØØØ
ØØ25   7766   M12,      -12
ØØ26   761Ø   M17Ø,     -17Ø
ØØ27   ØØØØ   CNT2,     Ø
ØØ3Ø   Ø612   PNTR2,    BST
ØØ31   ØØØØ   PNT2,     Ø
ØØ32   ØØØØ   TIMS,     Ø
ØØ33   Ø212   K212,     212
ØØ34   Ø215   K215,     215
ØØ35   ØØØØ   HLD,      Ø
ØØ36   ØØØØ   PLCE,     Ø
ØØ37   7776   M2,       -2
ØØ4Ø   ØØØØ   CHK,      Ø
ØØ41   Ø254   KCM,      254
ØØ42   Ø26Ø   K26Ø,     26Ø
ØØ43   Ø24Ø   SPACE,    24Ø
ØØ44   7774   M4,       -4
ØØ45   Ø624   OPT,      OCT
ØØ46   Ø421   XDEC,     DECPRT
ØØ47   Ø4ØØ   XPRT,     NPRT
ØØ5Ø   ØØØØ   DEL,      Ø
ØØ51   ØØØØ   TPMS,     Ø
ØØ52   Ø377   K377,     377
ØØ53   Ø611   PTN1,     BST-1
ØØ54   ØØØØ   PTN,      Ø
ØØ55   7773   M5,       -5
ØØ56   ØØØØ   TCNT,     Ø
ØØ57   74ØØ   K74ØØ,    74ØØ
ØØ6Ø   ØØ17   K17,      17
ØØ61   ØØØØ   KDEL,     Ø
ØØ62   ØØØØ   DEL1,     Ø
ØØ63   ØØØØ   SUM,      Ø
ØØ64   7763   M15,      -15
ØØ65   Ø654   XGETNO,   GETNO
ØØ66   ØØØØ   CRLF,     Ø
ØØ67   1Ø34             TAD K215
ØØ7Ø   4Ø74             JMS TYPE
ØØ71   1Ø33             TAD K212
ØØ72   4Ø74             JMS TYPE
ØØ73   5466             JMP I CRLF
ØØ74   ØØØØ   TYPE,     Ø
ØØ75   6Ø41             TSF
ØØ76   5Ø75             JMP.-1
ØØ77   6Ø46             TLS
Ø1ØØ   72ØØ             CLA
Ø1Ø1   5474             JMP I TYPE
Ø1Ø2   ØØØØ   COMMA,    Ø
```

```
Ø1Ø3   1Ø41              TAD KCM
Ø1Ø4   4Ø74              JMS TYPE
Ø1Ø5   55Ø2              JMP I COMMA
Ø1Ø6   ØØØØ   CLK,       Ø
Ø1Ø7   73ØØ              CLA CLL
Ø11Ø   1Ø24              TAD K4ØØØ
Ø111   65Ø5              DBCO
Ø112   65Ø6              DBSO
Ø113   73ØØ              CLA CLL
Ø114   7ØØØ              NOP
Ø115   7ØØØ              NOP
Ø116   2Ø5Ø              ISZ DEL
Ø117   5116              JMP.-1
Ø12Ø   7ØØØ              NOP
Ø121   7ØØØ              NOP
Ø122   65Ø4              DBRI
Ø123   ØØ52              AND K377
Ø124   3Ø51              DCA TPMS
Ø125   1Ø51              TAD TPMS
Ø126   55Ø6              JMP I CLK
              *2ØØ
Ø2ØØ   73ØØ   START,     CLA CLL
Ø2Ø1   3Ø63              DCA SUM
Ø2Ø2   41Ø6   DG,        JMS CLK
Ø2Ø3   764Ø              SZA CLA
Ø2Ø4   52Ø2              JMP DG
Ø2Ø5   2Ø63              ISZ SUM
Ø2Ø6   1Ø63              TAD SUM
Ø2Ø7   1Ø64              TAD M15
Ø21Ø   764Ø              SZA CLA
Ø211   52Ø2              JMP DG
Ø212   6Ø46              TLS
Ø213   1Ø2Ø              TAD M24
Ø214   3Ø21              DCA CNT
Ø215   3Ø32              DCA TIMS
Ø216   1Ø22              TAD PNTR
Ø217   3Ø23              DCA PNT
Ø22Ø   3423              DCA I PNT
Ø221   2Ø23              ISZ PNT
Ø222   2Ø21              ISZ CNT
Ø223   522Ø              JMP.-3
Ø224   1Ø55              TAD M5
Ø225   3Ø56              DCA TCNT
Ø226   1Ø53              TAD PTN1
Ø227   3Ø54              DCA PTN                    /FINISH OF "A" REGISTER.
Ø23Ø   41Ø6   LDIN,      JMS CLK
Ø231   3454              DCA I PTN
Ø232   41Ø6              JMS CLK
Ø233   711Ø              CLL RAR
Ø234   7Ø12              RTR
Ø235   7Ø12              RTR
Ø236   ØØ57              AND K74ØØ
Ø237   1454              TAD I PTN
```

```
0240  3454           DCA I PTN
0241  7240           STA
0242  1054           TAD PTN
0243  3054           DCA PTN
0244  1051           TAD TPMS
0245  7112           CLL RTR
0246  7012           RTR
0247  0060           AND K17
0250  3454           DCA I PTN
0251  4106           JMS CLK
0252  7106           CLL RTL
0253  7006           RTL
0254  1454           TAD I PTN
0255  3454           DCA I PTN
0256  7240           STA
0257  1054           TAD PTN
0260  3054           DCA PTN
0261  2056           ISZ TCNT
0262  5230           JMP LDIN
0263  2032           ISZ TIMS
0264  1022           TAD PNTR
0265  3023           DCA PNT
0266  1030           TAD PNTR2
0267  3031           DCA PNT2
0270  1025           TAD M12
0271  3021           DCA CNT
0272  1423  COMP,    TAD I PNT
0273  7041           CIA
0274  1431           TAD I PNT2
0275  7640           SZA CLA
0276  5304           JMP UNEQU
0277  2023           ISZ PNT
0300  2031           ISZ PNT2
0301  2021           ISZ CNT
0302  5272           JMP COMP
0303  5224           JMP LDIN-4
0304  7300  UNEQU,   CLA CLL
0305  4465           JMS I XGETNO
0306  7300           CLA CLL
0307  1022           TAD PNTR
0310  3023           DCA PNT
0311  1025           TAD M12
0312  3021           DCA CNT
0313  3036  NEWLOC,  DCA PLCE
0314  1423           TAD I PNT
0315  7104  DMORE,   CLL RAL
0316  7430           SZL
0317  4447           JMS I XPRT
0320  2036           ISZ PLCE
0321  7440           SZA
0322  5315           JMP DMORE
0323  7300           CLA CLL
```

```
Ø324  2Ø23          ISZ PNT
Ø325  2Ø21          ISZ CNT
Ø326  5313          JMP NEWLOC
Ø327  1Ø22          TAD PNTR
Ø33Ø  3Ø23          DCA PNT
Ø331  1Ø3Ø          TAD PNTR2
Ø332  3Ø31          DCA PNT2
Ø333  1Ø25          TAD M12
Ø334  3Ø21          DCA CNT
Ø335  1423  AGN,    TAD I PNT
Ø336  3431          DCA I PNT2
Ø337  2Ø23          ISZ PNT
Ø34Ø  2Ø31          ISZ PNT2
Ø341  2Ø21          ISZ CNT
Ø342  5335          JMP AGN
Ø343  1Ø32          TAD TIMS
Ø344  4446          JMS I XDEC
Ø345  4Ø66          JMS CRLF
Ø346  7ØØ1          IAC
Ø347  3Ø32          DCA TIMS
Ø35Ø  523Ø          JMP LDIN
            *4ØØ
Ø4ØØ  ØØØØ  NPRT,   Ø
Ø4Ø1  7421          MQL
Ø4Ø2  73ØØ          CLA CLL
Ø4Ø3  1Ø22          TAD PNTR
Ø4Ø4  7Ø41          CIA
Ø4Ø5  1Ø23          TAD PNT
Ø4Ø6  71Ø6          CLL RTL
Ø4Ø7  3Ø35          DCA HLD
Ø41Ø  1Ø35          TAD HLD
Ø411  71Ø4          CLL RAL
Ø412  1Ø35          TAD HLD
Ø413  1Ø36          TAD PLCE
Ø414  7ØØ1          IAC
Ø415  4446          JMS I XDEC
Ø416  41Ø2          JMS COMMA
Ø417  75Ø1          MQA
Ø42Ø  56ØØ          JMP I NPRT
Ø421  ØØØØ  DECPRT, Ø
Ø422  3272          DCA VALUE
Ø423  3273          DCA DIGIT
Ø424  3275          DCA CKS
Ø425  1264          TAD CA
Ø426  3274          DCA CB
Ø427  1263          TAD ADR
Ø43Ø  3235          DCA ARR
Ø431  741Ø          SKP
Ø432  3272          DCA VALUE
Ø433  71ØØ          CLL
Ø434  1272          TAD VALUE
```

```
Ø435  1265  ARR,     TAD TENPWR
Ø436  743Ø           SZL
Ø437  2273           ISZ DIGIT
Ø44Ø  743Ø           SZL
Ø441  5232           JMP ARR-3
Ø442  72ØØ           CLA
Ø443  1273           TAD DIGIT
Ø444  1275           TAD CKS
Ø445  765Ø           SNA CLA
Ø446  5256           JMP CON
Ø447  1273           TAD DIGIT
Ø45Ø  1271           TAD AKO
Ø451  6Ø41           TSF
Ø452  5251           JMP.-1
Ø453  6Ø46           TLS
Ø454  72ØØ           CLA
Ø455  2275           ISZ CKS
Ø456  3273  CON,     DCA DIGIT
Ø457  2235           ISZ ARR
Ø46Ø  2274           ISZ CB
Ø461  5234           JMP ARR-1
Ø462  5621           JMP I DECPRT
Ø463  1265  ADR,     TAD TENPWR
Ø464  7774  CA,      -4
Ø465  6Ø3Ø  TENPWR,  -175Ø
Ø466  7634           -144
Ø467  7766           -12
Ø47Ø  7777           -1
Ø471  Ø26Ø  AKO;     26Ø
Ø472  ØØØØ  VALUE,   Ø
Ø473  ØØØØ  DIGIT,   Ø
Ø474  ØØØØ  CB,      Ø
Ø475  ØØØØ  CKS,     Ø
            *6ØØ
Ø6ØØ  ØØØØ  AST,     Ø;
Ø6Ø1  ØØØØ  Ø;
Ø6Ø2  ØØØØ  Ø;
Ø6Ø3  ØØØØ  Ø;
Ø6Ø4  ØØØØ  Ø;
Ø6Ø5  ØØØØ  Ø;
Ø6Ø6  ØØØØ  Ø;
Ø6Ø7  ØØØØ  Ø;
Ø61Ø  ØØØØ  Ø;
Ø611  ØØØØ  Ø;

Ø612  ØØØØ  BST,     Ø;
Ø613  ØØØØ  Ø;
Ø614  ØØØØ  Ø;
Ø615  ØØØØ  Ø;
Ø616  ØØØØ  Ø;
Ø617  ØØØØ  Ø;
Ø62Ø  ØØØØ  Ø;
Ø621  ØØØØ  Ø;
Ø622  ØØØØ  Ø;
Ø623  ØØØØ  Ø;
```

```
Ø624  Ø3Ø3  OCT,   3Ø3        /C
Ø625  ØØØØ         Ø
Ø626  Ø3Ø3         3Ø3
Ø627  Ø243         243        /C#
Ø63Ø  Ø3Ø4         3Ø4        /D
Ø631  ØØØØ         Ø
Ø632  Ø3Ø4         3Ø4
Ø633  Ø243         243        /D#
Ø634  Ø3Ø5         3Ø5        /E
Ø635  ØØØØ         Ø
Ø636  Ø3Ø6         3Ø6        /F
Ø637  ØØØØ         Ø
Ø64Ø  Ø3Ø6         3Ø6
Ø641  Ø243         243        /F#
Ø642  Ø3Ø7         3Ø7        /G
Ø643  ØØØØ         Ø
Ø644  Ø3Ø7         3Ø7
Ø645  Ø243         243        /G#
Ø646  Ø3Ø1         3Ø1        /A
Ø647  ØØØØ         Ø
Ø65Ø  Ø3Ø1         3Ø1
Ø651  Ø243         243        /A#
Ø652  Ø3Ø2         3Ø2        /B
Ø653  ØØØØ         Ø
Ø654  ØØØØ  GETNO, Ø
Ø655  73ØØ         CLA CLL
Ø656  3Ø27         DCA CNT2
Ø657  1Ø22         TAD PNTR
Ø66Ø  3Ø23         DCA PNT
Ø661  1Ø25         TAD M12
Ø662  3Ø21         DCA CNT
Ø663  1423  NL,    TAD I PNT
Ø664  71Ø4  DM,    CLL RAL
Ø665  743Ø         SZL
Ø666  2Ø27         ISZ CNT2
Ø667  744Ø         SZA
Ø67Ø  5264         JMP DM
Ø671  2Ø23         ISZ PNT
Ø672  2Ø21         ISZ CNT
Ø673  5263         JMP NL
Ø674  1Ø27         TAD CNT2
Ø675  4446         JMS I XDEC
Ø676  5654         JMP I GETNO
            DBCO=65Ø5
            DBSO=65Ø6
            DBRI=65Ø4
            MQA=75Ø1
            MQL=7421
```

APPENDIX B

Program for Numeric Printout

```
                *20
0020   7754   M24,      -24
0021   0000   CNT,      0
0022   0600   PNTR,     AST
0023   0000   PNT,      0
0024   4000   K4000,    4000
0025   7766   M12,      -12
0026   7610   M170,     -170
0027   0000   CNT2,     0
0030   0612   PNTR2,    BST
0031   0000   PNT2,     0
0032   0000   TIMS,     0
0033   0212   K212,     212
0034   0215   K215,     215
0035   0000   HLD,      0
0036   0000   PLCE,     0
0037   7776   M2,       -2
0040   0000   CHK,      0
0041   0254   KCM,      254
0042   0260   K260,     260
0043   0240   SPACE,    240
0044   7774   M4,       -4
0045   0421   XDEC,     DECPRT
0046   0400   XPRT,     NPRT
0047   0000   DEL,      0
0050   0000   TPMS,     0
0051   0377   K377,     377
0052   0611   PTN1,     BST-1
0053   0000   PTN,      0
0054   7773   M5,       -5
0055   0000   TCNT,     0
0056   7400   K7400,    7400
0057   0017   K17,      17
0060   7772   KDEL,     7772
0061   0000   DEL1,     0
0062   0000   SUM,      0
0063   7761   M17,      -17
0064   0624   XGETNO,   GETNO
0065   0000   TEMP,     0
0066   0000   HLD1,     0
0067   7771   M7,       -7
0070   0000   BCT,      0
0071   0200   K200,     200
0072   0177   K177,     177
0073   0000   CRLF,     0
0074   1034             TAD K215
0075   4101             JMS TYPE
0076   1033             TAD K212
0077   4101             JMS TYPE
0100   5473             JMP I CRLF
0101   0000   TYPE,     0
0102   3065             DCA TEMP
0103   1065             TAD TEMP
0104   0072             AND K177
0105   3066             DCA HLD1
```

```
Ø1Ø6  1Ø67            TAD M7
Ø1Ø7  3Ø7Ø            DCA BCT
Ø11Ø  3Ø27            DCA CNT2
Ø111  1Ø65  MRE,      TAD TEMP
Ø112  1Ø27            TAD CNT2
Ø113  3Ø27            DCA CNT2
Ø114  1Ø65            TAD TEMP
Ø115  711Ø            CLL RAR
Ø116  3Ø65            DCA TEMP
Ø117  2Ø7Ø            ISZ BCT
Ø12Ø  5111            JMP MRE
Ø121  1Ø27            TAD CNT2
Ø122  7Ø1Ø            RAR
Ø123  72ØØ            CLA
Ø124  1Ø66            TAD HLD1
Ø125  753Ø            SZL CLL
Ø126  1Ø71            TAD K2ØØ
Ø127  3Ø66            DCA HLD1
Ø13Ø  76Ø4            LAS
Ø131  771Ø            SPA CLA
Ø132  5141            JMP PNCH
Ø133  1Ø66            TAD HLD1
Ø134  6Ø41            TSF
Ø135  5134            JMP.-1
Ø136  6Ø46            TLS
Ø137  72ØØ            CLA
Ø14Ø  55Ø1            JMP I TYPE
Ø141  1Ø66  PNCH,     TAD HLD1
Ø142  6Ø21            PSF
Ø143  5142            JMP.-1
Ø144  6Ø26            PLS
Ø145  72ØØ            CLA
Ø146  55Ø1            JMP I TYPE
Ø147  ØØØØ  COMMA,    Ø
Ø15Ø  1Ø41            TAD KCM
Ø151  41Ø1            JMS TYPE
Ø152  5547            JMP I COMMA
Ø153  ØØØØ  CLK,      Ø
Ø154  73ØØ            CLA CLL
Ø155  1Ø24            TAD K4ØØØ
Ø156  65Ø6            DBSO
Ø157  65Ø5            DBCO
Ø16Ø  73ØØ            CLA CLL
Ø161  1Ø6Ø            TAD KDEL
Ø162  3Ø61            DCA DEL1
Ø163  2Ø47            ISZ DEL
Ø164  5163            JMP.-1
Ø165  2Ø61            ISZ DEL1
Ø166  5163            JMP.-3
Ø167  65Ø4            DBRI
Ø17Ø  ØØ51            AND K377
Ø171  3Ø5Ø            DCA TPMS
```

```
0172  1050           TAD TPMS
0173  5553           JMP I CLK
            *200
0200  7300  START,   CLA CLL
0201  3062           DCA SUM
0202  4153  DG,      JMS CLK
0203  7640           SZA CLA
0204  5202           JMP DG
0205  2062           ISZ SUM
0206  1062           TAD SUM
0207  1063           TAD M17
0210  7640           SZA CLA
0211  5202           JMP DG
0212  6046           TLS
0213  6026           PLS
0214  1020           TAD M24
0215  3021           DCA CNT
0216  3032           DCA TIMS
0217  1022           TAD PNTR
0220  3023           DCA PNT
0221  3423           DCA I PNT
0222  2023           ISZ PNT
0223  2021           ISZ CNT
0224  5221           JMP.-3
0225  1054           TAD M5
0226  3055           DCA TCNT
0227  1052           TAD PTN1
0230  3053           DCA PTN              /FINISH OF "A" REGISTER.
0231  4153  LDIN,    JMS CLK
0232  3453           DCA I PTN
0233  4153           JMS CLK
0234  7110           CLL RAR
0235  7012           RTR
0236  7012           RTR
0237  0056           AND K7400
0240  1453           TAD I PTN
0241  3453           DCA I PTN
0242  7240           STA
0243  1053           TAD PTN
0244  3053           DCA PTN
0245  1050           TAD TPMS
0246  7112           CLL RTR
0247  7012           RTR
0250  0057           AND K17
0251  3453           DCA I PTN
0252  4153           JMS CLK
0253  7106           CLL RTL
0254  7006           RTL
0255  1453           TAD I PTN
0256  3453           DCA I PTN
0257  7240           STA
0260  1053           TAD PTN
0261  3053           DCA PTN
0262  2055           ISZ TCNT
0263  5231           JMP LDIN
0264  2032           ISZ TIMS
0265  1022           TAD PNTR
0266  3023           DCA PNT
```

```
Ø267   1Ø3Ø             TAD PNTR2
Ø27Ø   3Ø31             DCA PNT2
Ø271   1Ø25             TAD M12
Ø272   3Ø21             DCA CNT
Ø273   1423   COMP,     TAD I PNT
Ø274   7Ø41             CIA
Ø275   1431             TAD I PNT2
Ø276   764Ø             SZA CLA
Ø277   53Ø5             JMP UNEQU
Ø3ØØ   2Ø23             ISZ PNT
Ø3Ø1   2Ø31             ISZ PNT2
Ø3Ø2   2Ø21             ISZ CNT
Ø3Ø3   5273             JMP COMP
Ø3Ø4   5225             JMP LDIN-4
Ø3Ø5   73ØØ   UNEQU,    CLA CLL
Ø3Ø6   4464             JMS I XGETNO
Ø3Ø7   4147             JMS COMMA
Ø31Ø   1Ø22             TAD PNTR
Ø311   3Ø23             DCA PNT
Ø312   1Ø25             TAD M12
Ø313   3Ø21             DCA CNT
Ø314   3Ø36   NEWLOC,   DCA PLCE
Ø315   1423             TAD I PNT
Ø316   71Ø4   DMORE,    CLL RAL
Ø317   743Ø             SZL
Ø32Ø   4446             JMS I XPRT
Ø321   2Ø36             ISZ PLCE
Ø322   744Ø             SZA
Ø323   5316             JMP DMORE
Ø324   73ØØ             CLA CLL
Ø325   2Ø23             ISZ PNT
Ø326   2Ø21             ISZ CNT
Ø327   5314             JMP NEWLOC
Ø33Ø   1Ø22             TAD PNTR
Ø331   3Ø23             DCA PNT
Ø332   1Ø3Ø             TAD PNTR2
Ø333   3Ø31             DCA PNT2
Ø334   1Ø25             TAD M12
Ø335   3Ø21             DCA CNT
Ø336   1423   AGN,      TAD I PNT
Ø337   3431             DCA I PNT2
Ø34Ø   2Ø23             ISZ PNT
Ø341   2Ø31             ISZ PNT2
Ø342   2Ø21             ISZ CNT
Ø343   5336             JMP AGN
Ø344   1Ø32             TAD TIMS
Ø345   4445             JMS I XDEC
Ø346   4Ø73             JMS CRLF
Ø347   7ØØ1             IAC
Ø35Ø   3Ø32             DCA TIMS
Ø351   5225             JMP LDIN-4
```

```
                *400
0400  0000  NPRT,    0
0401  7421           MQL
0402  7300           CLA CLL
0403  1022           TAD PNTR
0404  7041           CIA
0405  1023           TAD PNT
0406  7106           CLL RTL
0407  3035           DCA HLD
0410  1035           TAD HLD
0411  7104           CLL RAL
0412  1035           TAD HLD
0413  1036           TAD PLCE
0414  7001           IAC
0415  4445           JMS I XDEC
0416  4147           JMS COMMA
0417  7501           MQA
0420  5600           JMP I NPRT
0421  0000  DECPRT,  0
0422  3274           DCA VALUE
0423  3275           DCA DIGIT
0424  3277           DCA CKS
0425  1266           TAD CA
0426  3276           DCA CB
0427  1265           TAD ADR
0430  3235           DCA ARR
0431  7410           SKP
0432  3274           DCA VALUE
0433  7100           CLL
0434  1274           TAD VALUE
0435  1267  ARR,     TAD TENPWR
0436  7430           SZL
0437  2275           ISZ DIGIT
0440  7430           SZL
0441  5232           JMP ARR-3
0442  7200           CLA
0443  1275           TAD DIGIT
0444  1277           TAD CKS
0445  7650           SNA CLA
0446  5253           JMP CON
0447  1275           TAD DIGIT
0450  1273           TAD AKO
0451  4101           JMS TYPE
0452  2277           ISZ CKS
0453  3275  CON,     DCA DIGIT
0454  2235           ISZ ARR
0455  2276           ISZ CB
0456  5234           JMP ARR-1
0457  1277           TAD CKS
0460  7640           SZA CLA
0461  5621           JMP I DECPRT
0462  1273           TAD AKO
0463  4101           JMS TYPE
0464  5621           JMP I DECPRT
```

```
0465  1267  ADR,     TAD TENPWR
0466  7774  CA,      -4
0467  6030  TENPWR,  -1750
0470  7634           -144
0471  7766           -12
0472  7777           -1
0473  0260  AKO,     260
0474  0000  VALUE,   0
0475  0000  DIGIT,   0
0476  0000  CB,      0
0477  0000  CKS,     0
            *600
0600  0000  AST,     0;
0601  0000  0;
0602  0000  0;
0603  0000  0;
0604  0000  0;
0605  0000  0;
0606  0000  0;
0607  0000  0;
0610  0000  0;
0611  0000  0;

0612  0000  BST,     0;
0613  0000  0;
0614  0000  0;
0615  0000  0;
0616  0000  0;
0617  0000  0;
0620  0000  0;
0621  0000  0;
0622  0000  0;
0623  0000  0;

0624  0000  GETNO,   0
0625  7300           CLA CLL
0626  3027           DCA CNT2
0627  1022           TAD PNTR
0630  3023           DCA PNT
0631  1025           TAD M12
0632  3021           DCA CNT
0633  1423  NL,      TAD I PNT
0634  7104  DM,      CLL RAL
0635  7430           SZL
0636  2027           ISZ CNT2
0637  7440           SZA
0640  5234           JMP DM
```

```
0641  2023          ISZ PNT
0642  2021          ISZ CNT
0643  5233          JMP NL
0644  1027          TAD CNT2
0645  4445          JMS I XDEC
0646  5624          JMP I GETNO
            DBCO=6505
            DBSO=6506
            DBRI=6504
            MQA=7501
            MQL=7421
```

APPENDIX C

Staff Notation Program

```
JCATP;
     "BEGIN""REAL"XS,YN,XO,PI,P,Q,R,S,T,U,V,W,X,Y,Z,Z1;
"INTEGER"L,J,NV,M,MIN,E,F,X1,X2,X3,X4,X5;
     "INTEGER"Y1,L;
     "INTEGER" NT,X6,X2,Y2;
     "INTEGER" "ARRAY" A,B[1:2Ø,1:21];
                         "INTEGER""ARRAY"IBUF[Ø:5ØØ];
                "LIBRARY"PLOTS,PLOT;
"PROCEDURE"STAFF(XS);
                "REAL"XS;
                     "BEGIN""INTEGER"I,J,K;"REAL"XA,XN,YN,AN,X,Y;
                     XA:=XS;
                  I:=Ø;
"COMMENT"     STAFF  LINES   DRAWN;
     STAR:     "IF"I>4"THEN""GOTO"DELT;      J:=Ø;
WYE:    PLOT(XA,Ø,3);PLOT(XA,25,2);J:=J+1;
     "IF"J=5"THEN""BEGIN"I:=I+1;
     "IF"I=1"THEN"XA:=XA+Ø.9;
     "IF"I=2"THEN"XA:=XA+Ø.6;
     "IF"I=3"THEN"XA:=XA+Ø.9;
     "GOTO"STAR;            "END";
     PLOT(XA,25,3);PLOT(XA,25,3);
XA:=XA+Ø.3;PLOT(XA,25,3);PLOT(XA,Ø,2);J:=J+1;
       PLOT(XA,Ø,3);       PLOT(XA,Ø,3);
                XA:=XA+Ø.3;"GOTO"WYE;
DELT:    I:=I;
        XA:=XA-7.2;
     PLOT(XA,25,2);    XA:=XA+7.2;
         PLOT(XA,Ø,3);    XA:=XA-7.2;PLOT(XA,Ø,2);
             "END"   OF   STAFF;
  "PROCEDURE"REST(XS,YN);
     "REAL"XS,YN;
   "BEGIN""REAL"X,Y;
        X:=XS+Ø.6;
        Y:=YN-Ø.2;PLOT(X,Y,3);
      X:=XS+Ø.4;PLOT(X,Y,2);
        Y:=YN+Ø.2;PLOT(X,Y,2);
        X:=XS+Ø.6;PLOT(X,Y,2);
      X:=XS+2.7;
        Y:=YN-Ø.2;PLOT(X,Y,3);
        X:=XS+2.5;PLOT(X,Y,2);
        Y:=YN+Ø.2;PLOT(X,Y,2);
        X:=XS+2.7;PLOT(X,Y,2);
        X:=XS+4.5;
        Y:=YN-Ø.2;PLOT(X,Y,3);
        X:=XS+4.3;PLOT(X,Y,2);
        Y:=YN+Ø.2;PLOT(X,Y,2);
        X:=XS+4.5;PLOT(X,Y,2);
        X:=XS+6.6;
        Y:=YN-Ø.2;PLOT(X,Y,3);
        X:=XS+6.4;PLOT(X,Y,2);
        Y:=YN+Ø.2;PLOT(X,Y,2);
        X:=XS+6.6;PLOT(X,Y,2);
   "END"  OF   REST;
  "PROCEDURE"TREBLE(XS);
    "REAL"XS;
```

```
        "BEGIN""REAL"AN,X0,Y0,X,Y;
     "INTEGER"I,K;
    X0:=XS+0.9;I:=1;
RET:       I:=I;
    AN:=210*PI/180;Y0:=1.0;
    X:=X0-0.2*SIN(AN);
    Y:=Y0+0.2*COS(AN);PLOT(X,Y,3);
          "FOR"K:=210"STEP"-15"UNTIL"0"DO"
    "BEGIN"AN:=K*PI/180;X:=X0-0.2*SIN(AN);
          Y:=Y0+0.2*COS(AN);PLOT(X,Y,2);
          "END";
    Y0:=Y0-0.1;    "FOR"K:=0"STEP"-15"UNTIL"-210"DO"
    "BEGIN"AN:=K*PI/180;X:=X0-0.3*SIN(AN);
        Y:=Y0+0.3*COS(AN);PLOT(X,Y,2);
    "END";X:=X-0.05;Y:=1.2;PLOT(X,Y,2);
    Y0:=1.1;X0:=X;
          "FOR"K:=0"STEP"15"UNTIL"180"DO"
    "BEGIN"AN:=K*PI/180;X:=X0-0.1*SIN(AN);
       Y:=Y0+0.1*COS(AN);PLOT(X,Y,2);
          "END";
    X:=X+1.5;PLOT(X,Y,2);Y0:=Y-0.1;
    X0:=X;      "FOR"K:=0"STEP"-15"UNTIL"-180"DO"
    "BEGIN"AN:=K*PI/180;X:=X0-0.1*SIN(AN);
      Y:=Y0+0.1*COS(AN);PLOT(X,Y,2);
     "END";Y0:=Y0-0.05;
          "FOR"K:=-90"STEP"15"UNTIL"90"DO"
          "BEGIN"AN:=K*PI/180;
     Y:=Y0+0.05*COS(AN);X:=X0+0.05*SIN(AN);
            PLOT(X,Y,2);
     Y:=Y0+0.05*COS(PI-AN);X:=X0+0.05*SIN(PI-AN);
            PLOT(X,Y,2);
          "END";
      I:=I+1;   "IF"I=2"THEN""BEGIN"X0:=XS+3.0;"GOTO"RET;"END";
    X0:=XS+4.2;
BASS:        Y0:=1.0;
    PLOT(X0,Y0,3);
          "FOR"K:=-90"STEP"15"UNTIL"90"DO"
          "BEGIN"AN:=K*PI/180;
    Y:=Y0+0.1*COS(AN);X:=X0+0.1*SIN(AN);PLOT(X,Y,2);
   X:=X0+0.1*SIN(PI-AN);    Y:=Y0+0.1*COS(PI-AN);PLOT(X,Y,2);
    "END";Y0:=1.15;
          "FOR"K:=180"STEP"-15"UNTIL"-30"DO"
   "BEGIN"AN:=K*PI/180;Y:=Y0+0.15*COS(AN);X:=X0-0.15*SIN(AN);
            PLOT(X,Y,2);
"END";Y:=1.2;X:=X0+0.3;PLOT(X,Y,2);
    X:=X+0.3;Y:=Y-0.39;PLOT(X,Y,2);
    I:=I+1;    "IF"I=4"THEN""BEGIN"X0:=XS+6.3;"GOTO"BASS;"END";
        "END"  OF  TREBLE;
   "PROCEDURE"SGNTRE(XS,X2,Y1,Y2);
    "REAL"XS,Y1;    "INTEGER"X2,Y2;
    "BEGIN"
    "REAL"X,Y,X1,AN,XN,YN,PI;       "INTEGER"I,J;
        PI:=4*ARCTAN(1);
    X:=XS;Y:=1.5;"IF"X2>1"THEN""GOTO"FLAT;    "IF"X2=0"THEN""GOTO"SAN;
```

```
SHRP:      "FOR"I:=1"STEP"1"UNTIL"4"DO""BEGIN"
    PLOT(X,Y,3);X:=X-0.2;PLOT(X,Y,3);
    X:=X+0.4;PLOT(X,Y,2);Y:=Y-0.1;PLOT(X,Y,3);
    X:=X-0.4;PLOT(X,Y,2);
    Y:=Y-0.1;   X:=X+0.15;PLOT(X,Y,3);
    Y:=Y+0.3;PLOT(X,Y,2);
    X:=X+0.15;PLOT(X,Y,3);      Y:=Y-0.3;PLOT(X,Y,2);     X:=X+2.0;Y:=Y+0.2;
  "END";
   "IF"Y2=2"THEN""BEGIN"Y2:=10;X:=XS+0.45;Y:=1.6;
                              "GOTO"  SHRP;
                      "END";
   "IF"Y2=3"THEN""BEGIN"Y2:=2;   X:=XS-0.15;Y:=1.7;
                              "GOTO"  SHRP;
                      "END";
   "IF"Y2=4"THEN""BEGIN"Y2:=3;   X:=XS+0.3;   Y:=1.3;
                              "GOTO"  SHRP;
                      "END";
   "IF"Y2=5"THEN""BEGIN"Y2:=4;X:=XS+0.75;;   Y:=1.9;
                              "GOTO"  SHRP;
                      "END";
   "IF"Y2=6"THEN""BEGIN"Y2:=5;   X:=XS+0.15;   Y:=2.0
         ;
                              "GOTO"  SHRP;
                      "END";
   "IF"Y2=7"THEN""BEGIN"Y2:=6;   X:=XS+0.6;Y:=2.1;
                              "GOTO"  SHRP;
                      "END";
  "GOTO"SA1;
FLAT:      X:=XS+0.6;    Y:=1.5;
NOE:       "FOR"I:=1"STEP"1"UNTIL"4"DO""BEGIN"
    PLOT(X,Y,3);   X:=X+0.1;   PLOT(X,Y,2);YN:=Y;
    X:=X-0.45;PLOT(X,Y,2);    XN:=X+0.2+0.15;
    "FOR"J:=90"STEP"-15"UNTIL"-90"DO"
    "BEGIN"AN:=J*PI/180;
            X:=XN+0.1*SIN(AN);
            Y:=YN+0.1*COS(AN);
            PLOT(X,Y,2);
    "END";   X:=X+2.2;
                                                  "END";
    "IF"Y2=2"THEN""BEGIN"Y2:=10;X:=XS+0.15;Y:=1.6;
                              "GOTO"  NOE;
                      "END";
    "IF"Y2=3"THEN""BEGIN"Y2:=2;  X:=XS+0.75;Y:=1.7;
                              "GOTO"  NOE;
                      "END";
    "IF"Y2=4"THEN""BEGIN"Y2:=3;   X:=XS+0.3;   Y:=1.3;
                              "GOTO"  NOE;
                      "END";
    "IF"Y2=5"THEN""BEGIN"Y2:=4;X:=XS+0.9;   Y:=1.9;
                              "GOTO"  NOE;
                      "END";
    "IF"Y2=6"THEN""BEGIN"Y2:=5;   X:=XS+0.45;   Y:=2.0;
                              "GOTO"  NOE;
                      "END";
```

```
    "IF"Y2=7"THEN""BEGIN"Y2:=6;   X:=XS+1.05;   Y:=2.1;
                              "GOTO"  JOE;
                    "END";
SA1:        Y2:=Y2;
    "END"  OF  SGNTRE;
    "PROCEDURE"NOTE(X1,X2,X3,X4,YN,Y1);
"INTEGER"X1,X2,X3,X4,Y1;
  "REAL"YN;
 "BEGIN""INTEGER"G,H,I,J,K;
    "INTEGER" "ARRAY" PITCH[0:1,1:12,1:15];
          "REAL" XN,X,Y,XNC,YNC,AN,XA,XY;
  "IF"  Y1=1"THEN" "GOTO"AAA;
   "FOR"I:=1"STEP"1"UNTIL"15"DO"
   "BEGIN"  PITCH[0,1,I]:=1;   PITCH[0,2,I]:=1;
        PITCH[0,3,I]:=2;   PITCH[0,4,I]:=2;
      PITCH[0,5,I]:=3;   PITCH[0,6,I]:=4;
      PITCH[0,7,I]:=4;   PITCH[0,8,I]:=5;
      PITCH[0,9,I]:=5; PITCH[0,10,I]:=6;
      PITCH[0,11,I]:=6;   PITCH[0,12,I]:=7;
  "END";
      PITCH[0,5,2]:=3;
      PITCH[0,1,1]:=0;   PITCH[0,4,3]:=PITCH[0,5,1]:=PITCH[0,5,2];
      PITCH[0,11,3]:=7; PITCH[0,5,15]:=4;
      PITCH[0,12,14]:=PITCH[0,12,15]:=8;
    "FOR"I:=9"STEP"1"UNTIL"15"DO"
      "BEGIN"  PITCH[0,2,I]:=2;   PITCH[0,4,I]:=3;   PITCH[0,7,I]:=5;
      PITCH[0,9,I]:=6; PITCH[0,11,I]:=7;
      "END";
    "FOR"I:=1"STEP"1"UNTIL"15"DO"
    "FOR"J:=1"STEP"1"UNTIL"12"DO"
        PITCH[1,J,I]:=0;
  "FOR"I:=2"STEP"1"UNTIL"6,14"STEP"1"UNTIL"15"DO"
    PITCH[1,1,I]:=5;
 "FOR"I:=1"STEP"1"UNTIL"4,12"STEP"1"UNTIL"15"DO"  PITCH[1,3,I]:=5;
 "FOR"I:=10"STEP"1"UNTIL"14"DO"PITCH[1,5,I]:=5; PITCH[1,5,1]:=5;
 "FOR"I:=3"STEP"1"UNTIL"7,15"STEP"1"UNTIL"15"DO"  PITCH[1,6,I]:=5;
 "FOR"I:=1"STEP"1"UNTIL"5"DO"  PITCH[1,8,I]:=5;
    "FOR"I:=13"STEP"1"UNTIL"15"DO"PITCH[1,8,I]:=5;
 "FOR"I:=1"STEP"1"UNTIL"3,11"STEP"1"UNTIL"15"DO"PITCH[1,10,I]:=5;
    "FOR"I:=9"STEP"1"UNTIL"13"DO"PITCH[1,12,I]:=5;   PITCH[1,12,1]:=5;
    PITCH[1,2,7]:=PITCH[1,2,8]:=1;
    PITCH[1,4,5]:=PITCH[1,4,6]:=PITCH[1,4,7]:=1;
    PITCH[1,7,8]:=PITCH[1,9,6]:=PITCH[1,9,7]:=1;
    "FOR"I:=4"STEP"1"UNTIL"7"DO"PITCH[1,11,I]:=1
    PITCH[1,2,9]:=PITCH[1,2,10]:=PITCH[1,2,11]:=2;
    PITCH[1,4,8]:=PITCH[1,4,9]:=2;
    "FOR"I:=9"STEP"1"UNTIL"12"DO"  PITCH[1,7,I]:=2;
    PITCH[1,9,8]:=PITCH[1,9,9]:=PITCH[1,9,10]:=2;
    PITCH[1,11,8]:=2;
AAA:      Y1:=1;
            H:=0;    G:=0;
          K:=X1;       XY:=X3;
"COMMENT"  DETERMINATION OF NOTE PITCH ;
TRY:        "IF"X1>12"THEN"
```

```
            "BEGIN"N:=N+1; K1:=K1-12;
                  "GOTO"TRY;
        "END";
     G:=PITCH[Ø,K1,K6+3];      K3:=PITCH[1,K1,K6+3];
     XN:=XS+9.Ø-(1.Ø5*N+Ø.15*G);
      XNC:=XN; X:=XN; YNC:=YN; Y:=YN+Ø.2;
  PLOT(X,Y,3);
"COMMENT"ELIPSE  DRAWN;
"FOR"I:=Ø"STEP"3Ø"UNTIL"36Ø"DO"
        "BEGIN"AN:=I*PI/18Ø;
            X:=XN+Ø.15*SIN(AN);
Y:=YN+Ø.2Ø*COS(AN);
             PLOT(X,Y,2);
   "END";
  "IF"K4"GE"2Ø"THEN""BEGIN"K4:=K4-2Ø; "GOTO"DOT; "END";
    "IF"K2"GE"1Ø"THEN""BEGIN"K2:=K2-1Ø;
      Y:=YN-Ø.2;   X:=XNC;   PLOT(X,Y,3);   X:=X-Ø.6; PLOT(X,Y,2);
    "GOTO"FILL; "END";
"COMMENT"NOTE TAIL  DRAWN;
UP:    X:=X-Ø.9; PLOT(X,Y,2);
"COMMENT"  QUAVER  TAILS  DRAWN;
            "IF"K2>2"THEN""BEGIN"
"FOR"I:=1"STEP"1"UNTIL"K2-2"DO"
"BEGIN"Y:=Y+Ø.3; PLOT(X,Y,2);
     X:=X+Ø.1; PLOT(X,Y,2);
     Y:=Y-Ø.3; PLOT(X,Y,2);
        X:=X+Ø.1; PLOT(X,Y,2);
  "END""END";
   PLOT(XNC,YNC,3);
"COMMENT"NOTE  FILLED  OR  NOT;
FILL:      "IF"K2>1"THEN"
     "BEGIN"X:=XNC; Y:=YNC;
        PLOT(X,Y,3);
        "FOR"I:=-9Ø"STEP"1Ø"UNTIL"9Ø"DO"
       "BEGIN"AN:=I*PI/18Ø;
              Y:=YNC+Ø.2Ø*COS(AN);
             X:=XNC+Ø.15*SIN(AN);
             PLOT(X,Y,2);
           Y:=YNC+Ø.2Ø*COS(PI-AN);
              X:=XNC+Ø.15*SIN(PI-AN);
                PLOT(X,Y,2);
        "END";
    "END";
"COMMENT"  DOTS  ADDED;
DOT:          "IF"K4>Ø"THEN"
     "BEGIN"X:=XNC;  Y:=YNC; Y:=YNC+Ø.32; PLOT(X,Y,3);
      "FOR"J:=1"STEP"1"UNTIL"K4"DO"
             "FOR"I:=Ø"STEP"3Ø"UNTIL"36Ø"DO"
     "BEGIN"AN:=I*PI/18Ø;
X:=X+Ø.Ø2*SIN(AN);
     Y:=Y+Ø.Ø2*COS(AN); PLOT(X,Y,2);
  "END"; Y:=Y+Ø.1;
    PLOT(X,Y,3);
  "END";
    "IF"K=1"OR"K=2"THEN""BEGIN"
    X:=XNC-Ø.15; Y:=YNC-Ø.2; PLOT(X,Y,3);
```

```
Y:=YNC+Ø.2; PLOT(X,Y,2);
      X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);
X:=X-Ø.3; PLOT(X,Y,3); Y:=Y+Ø.4; PLOT(X,Y,2);
      X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);
X:=X-Ø.3; PLOT(X,Y,3); Y:=Y+Ø.4; PLOT(X,Y,2);
"END";
"IF"K=15"OR"K=39"OR"K=33"OR"K=34"OR"K=1Ø7"OR"K=1Ø3"THEN""BEGIN"
      X:=XNC+Ø.15; Y:=YNC-Ø.2; PLOT(X,Y,3);
Y:=YNC+Ø.2; PLOT(X,Y,2);
      X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);              "END";
"IF"K=11"OR"K=12"THEN""BEGIN"
X:=XNC-Ø.15; Y:=YNC-Ø.2; PLOT(X,Y,3);
Y:=YNC+Ø.2; PLOT(X,Y,2);
      X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);              "END";
"IF"K=3"THEN""BEGIN"
X:=XNC-Ø.15; Y:=YNC-Ø.2; PLOT(X,Y,3);
Y:=YNC+Ø.2; PLOT(X,Y,2);
      X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);
X:=X-Ø.3; PLOT(X,Y,3); Y:=Y+Ø.4; PLOT(X,Y,2);
"END";
"IF"K=4"OR"K=5"THEN""BEGIN"
X:=XNC-Ø.15; Y:=YNC-Ø.2; PLOT(X,Y,3);
Y:=YNC+Ø.2; PLOT(X,Y,2);
      X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);
X:=X-Ø.3; PLOT(X,Y,3); Y:=Y+Ø.4; PLOT(X,Y,2);
      X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);              "END";
"IF"K=111"THEN""BEGIN"
X:=XNC+Ø.15; Y:=YNC-Ø.2; PLOT(X,Y,3);
Y:=YNC+Ø.2; PLOT(X,Y,2);
X:=X+Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);          "END";
"IF"K=114"OR"K=115"THEN""BEGIN"
X:=XNC+Ø.15; Y:=YNC-Ø.2; PLOT(X,Y,3);
Y:=YNC+Ø.2; PLOT(X,Y,2);
X:=X+Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);
X:=X+Ø.3; PLOT(X,Y,3);    Y:=Y+Ø.4; PLOT(X,Y,2); "END";
"IF"K=117"OR"K=113"THEN""BEGIN"
X:=XNC+Ø.15; Y:=YNC-Ø.2; PLOT(X,Y,3);
Y:=YNC+Ø.2; PLOT(X,Y,2);
X:=X+Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);
X:=X+Ø.3; PLOT(X,Y,3);    Y:=Y+Ø.4; PLOT(X,Y,2);
X:=X+Ø.3; PLOT(X,Y,3); Y:=Y-Ø.4; PLOT(X,Y,2);          "END";
"IF"K=8"THEN""BEGIN"
X:=XNC; Y:=YNC+Ø.4; PLOT(X,Y,3);
Y:=YNC-Ø.4; PLOT(X,Y,2);
"FOR"I:=1"STEP"1"UNTIL"4"DO"
"BEGIN"
X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-((-1)↑I)*Ø.8; PLOT(X,Y,2);
"END""END";
"IF"K=6"OR"K=7"THEN""BEGIN"
X:=XNC; Y:=YNC+Ø.4; PLOT(X,Y,3);
Y:=YNC-Ø.4; PLOT(X,Y,2);
"FOR"I:=1"STEP"1"UNTIL"3"DO"
"BEGIN"
X:=X-Ø.3; PLOT(X,Y,3); Y:=Y-((-1)↑I)*Ø.8; PLOT(X,Y,2);
```

```
"END""END";
"IF"K=9"OR"K=10"THEN""BEGIN"
X:=XNC;Y:=YNC+0.4;PLOT(X,Y,3);
Y:=YNC-0.4;PLOT(X,Y,2);
"FOR"I:=1"STEP"1"UNTIL"2"DO"
"BEGIN"
X:=X-0.3;PLOT(X,Y,3);Y:=Y-((-1)↑I)*0.3;PLOT(X,Y,2);
"END""END";
"IF"K=13"OR"K=14"THEN""BEGIN"
X:=XNC;Y:=YNC+0.4;PLOT(X,Y,3);
Y:=YNC-0.4;PLOT(X,Y,2);
X:=X-0.3;PLOT(X,Y,3);Y:=Y+0.3;PLOT(X,Y,2);"END";
"IF"K=16"OR"K=17"THEN""BEGIN"
X:=XNC;Y:=YNC+0.4;PLOT(X,Y,3);
Y:=YNC-0.4;PLOT(X,Y,2);
"END";
"IF"K=37"OR"K=38"OR"K=40"OR"K=41"OR"K=61"OR"K=62"OR"K=31"OR"K=32
"OR"K=35"OR"K=36"OR"K=105"OR"K=106"THEN""BEGIN"
X:=XNC;Y:=YNC+0.4;PLOT(X,Y,3);
Y:=YNC-0.4;PLOT(X,Y,2);
"END";
"IF"K=109"OR"K=110"THEN""BEGIN"
X:=XNC;Y:=YNC+0.4;PLOT(X,Y,3);
Y:=YNC-0.4;PLOT(X,Y,2);
X:=X+0.3;PLOT(X,Y,3);            Y:=Y+0.3;PLOT(X,Y,2);"END";
"IF"K=112"OR"K=113"THEN""BEGIN"
X:=XNC;Y:=YNC+0.4;PLOT(X,Y,3);
Y:=YNC-0.4;PLOT(X,Y,2);
"FOR"I:=1"STEP"1"UNTIL"2"DO"
"BEGIN"
X:=X+0.3;PLOT(X,Y,3);Y:=Y-((-1)↑I)*0.3;PLOT(X,Y,2);
"END""END";
"IF"K=116"THEN""BEGIN"
X:=XNC;Y:=YNC+0.4;PLOT(X,Y,3);
Y:=YNC-0.4;PLOT(X,Y,2);
"FOR"I:=1"STEP"1"UNTIL"3"DO"
"BEGIN"
X:=X+0.3;PLOT(X,Y,3);Y:=Y-((-1)↑I)*0.3;PLOT(X,Y,2);
"END""END";
"IF"K=119"OR"K=120"THEN""BEGIN"
X:=XNC;Y:=YNC+0.4;PLOT(X,Y,3);
Y:=YNC-0.4;PLOT(X,Y,2);
"FOR"I:=1"STEP"1"UNTIL"4"DO"
"BEGIN"
X:=X+0.3;PLOT(X,Y,3);Y:=Y-((-1)↑I)*0.3;PLOT(X,Y,2);
"END""END";
"IF"KY>5"THEN""BEGIN"KY:=0;YNC:=YNC-0.5;"END";
"COMMENT" DRAW ACCIDENTALS;
"COMMENT" SHARP;
"IF"K3=1"THEN"
"BEGIN"X:=XNC-0.15;Y:=YNC-0.3;
PLOT(X,Y,3);    X:=X+0.3;PLOT(X,Y,2);   Y:=Y-0.1;PLOT(X,Y,3);
X:=X-0.3;PLOT(X,Y,2);Y:=Y-0.1;X:=X+0.1;PLOT(X,Y,3);
Y:=Y+0.3;PLOT(X,Y,2);X:=X+0.1;PLOT(X,Y,3);
```

```
        Y:=Y-Ø.3; PLOT(X,Y,2);
    "END";
"COMMENT"FLAT OR DOUBLE FLAT;
  "IF"K3>1"THEN"
        "BEGIN""IF"K3<4"THEN"
          "BEGIN"Y:=YNC-Ø.4; X:=XNC;
  PLOT(X,Y,3);
             "FOR"I:=1"STEP"1"UNTIL"K3-1"DO"
               "BEGIN"  X:=X+Ø.Ø5; PLOT(X,Y,2); X:=X-Ø.4; PLOT(X,Y,2);
        XN:=X+Ø.35; PLOT(X,Y,3);
        "FOR"J:=9Ø"STEP"-15"UNTIL"-9Ø"DO"
      "BEGIN"AN:=J*PI/18Ø;
       X:=XN+Ø.Ø5*SIN(AN); Y:=YNC-Ø.4+Ø.1*COS(AN);
            PLOT(X,Y,2);
     "END"; Y:=YN-Ø.3;   X:=XNC; PLOT(X,Y,3);
"END""END""END";
"COMMENT"  DOUBLE  SHARP;
        "IF"K3=4"THEN"
  "BEGIN"X:=XNC-Ø.2; Y:=YNC-Ø.5; PLOT(X,Y,3);
         X:=X+Ø.4;
     Y:=Y+Ø.2; PLOT(X,Y,2);
     X:=X-Ø.4;
    PLOT(X,Y,3); X:=X+Ø.4; Y:=Y-Ø.2; PLOT(X,Y,2);
     "END";
"COMMENT"  NATURAL;
    "IF"K3=5"THEN"
    "BEGIN"
    X:=XNC-Ø.2;  Y:=YNC-Ø.4; 'PLOT(X,Y,3);
    X:=X+Ø.5;  PLOT(X,Y,2);
    X:=X-Ø.1; Y:=Y-Ø.1; PLOT(X,Y,3);
    X:=X-Ø.5;  PLOT(X,Y,2);
    X:=X+Ø.1;  Y:=Y+Ø.1;  PLOT(X,Y,3);
    X:=X+Ø.1; Y:=Y-Ø.1;  PLOT(X,Y,2);
    X:=X+Ø.3;  PLOT(X,Y,3);
    X:=X-Ø.1;  Y:=Y+Ø.1; PLOT(X,Y,2);
PLOT(X,Y,3);      PLOT(X,Y,3);
 "END";
PLOT(X,Y,3);      PLOT(X,Y,3);
"END"       OF  NOTE;
   PLOTS(IBUF,5ØØ,3);
   READER(1);
"COMMENT"  NOTE  TIME  INTERVALS  DEFINED;
    Y1:=Ø;
      "READ"P,Q; PI:=4*ARCTAN(1); MIN:=Ø;
  R:=Q+1;      S:=R+2*(Q-P);
  T:=S+1;      U:=T+2*(S-R);
  V:=U+1;      W:=V+2*(U-T);
  X:=W+1;      Y:=X+2*(W-V);
  Z:=Y+1;      Z1:=Z+2*(Y-X);
  YN:=2.Ø;    XS:=Ø;   K3:=Ø;
    "READ"K6;
    "READ"L;
     STAFF(XS);
       TREBLE(XS);
```

```
    "IF"K6<0"THEN"K2:=1;    "IF"K6>0"THEN"K2:=2;
    Y2:=ABS(K6);    "IF"K6=0"THEN"K2:=0;
    SGNTRE(KS,K2,Y1,Y2);
"COMMENT"  NOTE  DATA  ENTERED  INTO  ARRAYS;
  "FOR"I:=1"STEP"1"UNTIL"19"DO"
   "BEGIN""READ"NV; A[I,21]:=NV;
"FOR"J:=1"STEP"1"UNTIL"NV"DO"
   "READ"A[I,J]; "READ"NT;
   "FOR"J:=1"STEP"1"UNTIL"NV"DO"
     B[I,J]:=NT;
   "FOR"J:=NV+1"STEP"1"UNTIL"20"DO"
 "BEGIN"A[I,J]:=0; B[I,J]:=0;
"END""END";
"COMMENT"  MORE  DATA  ENTERED;
MORE: "READ"NV;                  A[20,21]:=NV;
    "IF"NV=21"THEN""BEGIN"MIN:=MIN+1; "GOTO"DONE; "END";
    "IF"NV=0"THEN""BEGIN""READ"B[20,1];    A[20,1]:=0;
          "FOR"J:=2"STEP"1"UNTIL"20"DO"A[20,J]:=B[20,J]:=0;
    "GOTO" DONE ; "END";
   "FOR"J:=1"STEP"1"UNTIL"NV"DO"
  "READ"A[20,J]; "READ"NT;
  "FOR"J:=1"STEP"1"UNTIL"NV"DO"
  B[20,J]:=NT;
  "FOR"J:=NV+1"STEP"1"UNTIL"20"DO"
   A[20,J]:=B[20,J]:=0;
"COMMENT"  DATA  PROCESSING  FOR  TOTAL  TIME  VALUE;
DONE:       "IF"A[1,21]=0"THEN" "GOTO"OFF;
    "COMMENT"  LAZY  FINGER  COMPENSATION;
    "IF"L=1"THEN"
    "BEGIN" I:=A[2,21]-A[1,21];
    "IF"I<3"AND"I"NE"0"THEN"
    "BEGIN"J:=1;    M:=1;    K5:=1;
SIC:        "IF"A[2,J]=A[1,K5]"THEN" "BEGIN"J:=J+1; "GOTO"SIC; "END";
    "IF"A[2,J]<A[1,K5] "THEN"
    "BEGIN" "FOR"I:=A[1,21]"STEP"-1"UNTIL"K5"DO"
           A[1,I+1]:=A[1,I];
         A[1,K5]:=A[2,J];    A[1,21]:=A[1,21]+1;
     "END";
    K5:=K5+1;
    "IF"K5"LE"A[1,21]"THEN" "GOTO"SIC;
      M:=A[1,21]+1;
    A[1,M]:=A[2,J];;  A[1,21]:=M;    J:=J+1; K5:=1;
    "IF"A[1,21]<A[2,21]  "THEN" "GOTO" SIC;
  "END";   "END";
    I:=1;
EK:   M:=2;
WIS:   J:=1;
BACK:     "IF"A[1,I]=A[M,J]"AND"A[1,I]"NE"0"THEN""GOTO"OUT;
   J:=J+1;
  "IF"J"LE"20"THEN""GOTO"BACK;
NEW:     I:=I+1; M:=2;
   "IF"I"LE"20"THEN""GOTO"WIS;
   "GOTO"OFF;
OUT:      B[1,I]:=B[1,I]+B[M,J];
```

```
        B[M,J]:=0; A[M,21]:=A[M,21]-1;
          "IF" A[M,21]>0"THEN""BEGIN"
     "FOR"K5:=J"STEP"1"UNTIL"A[M,21]"DO"
     "BEGIN"  A[M,K5]:=A[M,K5+1];
          B[M,K5]:=B[M,K5+1];
     "END""END";
      K5:=A[M,21];
      B[M,K5+1]:=0;
       A[M,K5+1]:=0;        M:=M+1;
     "IF"M"LE"20"THEN""GOTO"WIS;
      I:=I+1; "IF"I"LE"20"THEN""GOTO"EX;
OFF:        "IF"A[1,21]=21"THEN""GOTO"FIN; E:=A[1,21];
"COMMENT"  DETERMINATION  OF NOTE TYPE PLUS  ACCIDENTAL;
     K5:=0;
     "IF" E=0 "THEN" "GOTO" NUNN;
     "FOR"I:=E"STEP"-1"UNTIL"1"DO"
       "BEGIN"K1:=A[1,I]; K4:=0; K2:=0;
         "IF"B[1,I]"LE"Q"THEN""BEGIN"
       "IF"B[1,I]"GE"P"THEN""BEGIN"
       K2:=6;
       "END"; "END";
         "IF"B[1,I]"LE"S"THEN"
       "BEGIN""IF"B[1,I]"GE"R"THEN""BEGIN"
            K2:=5;
      "END""END";
       "IF"B[1,I]"LE"U"THEN"
       "BEGIN""IF"B[1,I]"GE"T"THEN""BEGIN"
        K2:=4;
       "END""END";
       "IF"B[1,I]"LE"W"THEN"
       "BEGIN""IF"B[1,I]"GE"V"THEN""BEGIN"
         K2:=3; "IF"B[1,I]>(3*W+V)/4"THEN"K4:=1;
       "END""END";
       "IF"B[1,I]"LE"Y"THEN"
       "BEGIN""IF"B[1,I]"GE"X"THEN""BEGIN"
        K2:=2;    "IF"B[1,I]"GE"(3*Y+X)/4   "THEN"K4:=1;
       "END""END";
       "IF"B[1,I]"GE"Z"THEN"K2:=1;
    "IF"B[1,I]>2*Z"THEN""BEGIN"K4:=K4+20; "GOTO"OVER; "END";
     "IF"B[1,I]>(3*Z1+Z)/4"THEN"K4:=1;
     "IF"B[1,I]>Z1"THEN"
     "BEGIN""IF"B[1,I]-Z1<Z"THEN""GOTO"OVER;
     F:=A[2,21]; F:=F+1; A[2,F]:=A[1,I];
     B[2,F]:=B[1,I]-Z1;     A[2,21]:=F;
     "END";
OVER:        "IF"I<E"THEN""BEGIN"
       "IF"ABS(A[1,I+1]-A[1,I])<12"THEN"
     "BEGIN""IF"K2>2"THEN"K2:=2; "END";
     "IF"ABS(A[1,I+1]-A[1,I])<4"THEN""BEGIN"
     "IF"K5=1"THEN" "BEGIN"K5:=0; NOTE(K1,K2,K3,K4,YN,Y1); "GOTO"AWAY;  "EN
         D";
     K5:=1; K2:=K2+10; K3:=K3+10; YN:=YN+0.4;  NOTE(K1,K2,K3,K4,YN,Y1);
     YN:=YN-0.4;   "GOTO" AWAY;  "END";
     "END";
```

```
     NOTE(K1,K2,K3,K4,YN,Y1);
     K5:=Ø;
AWAY:       "END";
       NIL:        YN:=YN+2.Ø;
NUNN:      YN:=YN;
"COMMENT"INTRODUCTION  OF  NEW  NOTE  DATA  SET;
"FOR"I:=2"STEP"1"UNTIL"2Ø"DO"
       "BEGIN""FOR"J:=1"STEP"1"UNTIL"21"DO"
         "BEGIN"A[I-1,J]:=A[I,J];
                 B[I-1,J]:=B[I,J];
         "END";
       "END";
     "FOR"J:=1"STEP"1"UNTIL"2Ø"DO""BEGIN"A[2Ø,J]:=Ø;B[2Ø,J]:=Ø;"END";
"COMMENT"NEW  STAFF  LINES  DRAWN;
          "IF"YN>24.Ø"THEN"
     "BEGIN"XS:=XS+12.Ø;
              STAFF(XS);YN:=2.Ø;
        TREBLE(XS);
     Y1:=1;
      Y2:= ABS(K6);
     SGNTRE(XS,K2,Y1,Y2);
       "END";
       "IF"A[1,21]=21"THEN""GOTO"FIN;
     "IF"MIN=1"THEN""GOTO"DONE;
          "GOTO"MORE;
FIN:      PLOT(X,Y,3);PLOT(X,Y,3);PLOT(X,Y,999);
"END";
```

APPENDIX D

Examples of Printout

PRELUDE IN C MINOR -- CHOPIN.

C3 C4 G4 C5 D#5 G5 24
1
F2 F3 G#4 C5 D#5 G#5 24
1
G2 G3 G4 B4 D#5 G5 12
G2 G3 G4 B4 D5 F5 12
1
C3 G3 C4 D#4 G4 C5 D#5 24
1
G#2 G#3 D#4 G#4 C5 D#5 24
1
C#2 C#3 F4 G#4 C#5 F5 24
1
D#2 D#3 C#4 D#4 G4 C5 D#5 12
D#2 D#3 C#4 D#4 G4 A#4 D5 12
1
G#2 G#3 C4 D#4 G#4 C5 24
1
G2 G3 D4 F4 B4 D5 24
1
C2 C3 E4 G4 A#4 C5 E5 24
1
F2 F3 G#4 C4 G5 12
F2 F3 G#4 C4 F5 12
1
C3 C4 G4 C5 D#5 24
1
D3 A3 D4 F#4 C5 D5 24
1
G2 G3 G4 B4 D5 G5 24
1
D2 D3 C5 D5 F#5 B5 12
D2 D3 C5 D5 F#5 A5 12
1
G3 G4 B4 D5 G5 24
1
C3 C4 D#5 G5 D#6 24
1
C4 C5 D#5 G5 D#6 24
1
B3 B4 D5 G#5 D6 12
G#5 6
1
B3 B4 D5 F#5 D6 7
1
A#3 A#4 D5 G5 D6 24
1
A3 A4 C5 G5 C6 24

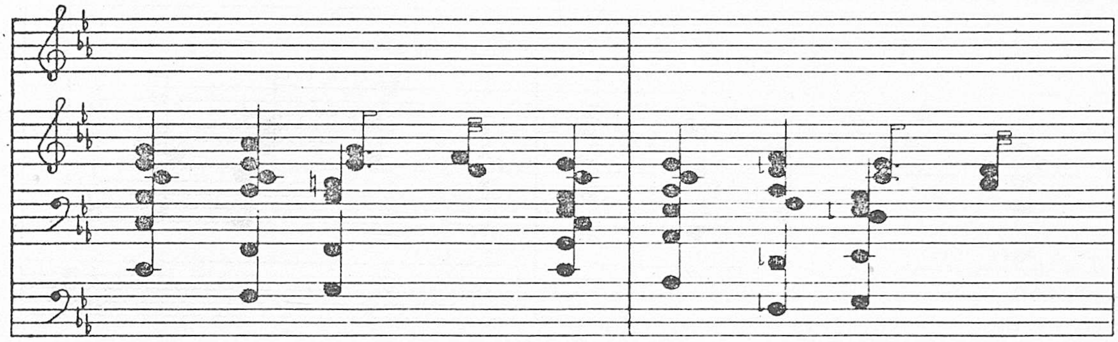
```
1
G#3  G#4  C5   D5   F#5  D6   24
1
G3   G4   D5   G5   B5   18
G3   G4   C5   A5   7
1
F3   F4   B4   D5   G5   24
1
D#3  D#4  C5   G5   C6   24
1
F3   F4   G#4  C5   G#5  24
1
B2   B3   G4   D5   G5   18
B2   B3   G4   D5   F5   7
1
C3   C4   G4   C5   D#5  24
1
G#2  G#3  D#4  G#4  C5   D#5  24
1
C#2  C#3  F4   G#4  C#5  F5   24
1
G2   G3   F4   G4   B4   D#5  18
G2   G3   F4   G4   B4   D5   7
1
C2   C3   D#4  G4   C5   24
1
C3   C4   D#5  G5   D#6  24
1
C4   C5   D#5  G#5  D#6  24
1
B3   B4   D5   G#5  D6   12
G#5  6
B3   B4   D5   F#5  D6   7
1
A#3  A#4  D5   G5   D6   24
1
A3   A4   C5   G5   C6   24
1
G#3  G#4  C5   D5   F#5  D6   24
1
G3   G4   D5   G5   B5   18
G3   G4   C5   A5   6
1
F3   F4   B4   D5   G5   24
1
D#3  D#4  C5   G5   C6   24
1
F3   F4   G#4  C5   G#5  24
1
B2   B3   G4   D5   G5   18
B2   B3   G4   D5   F5   6
1
C3   C4   G4   C5   D#5  24
```

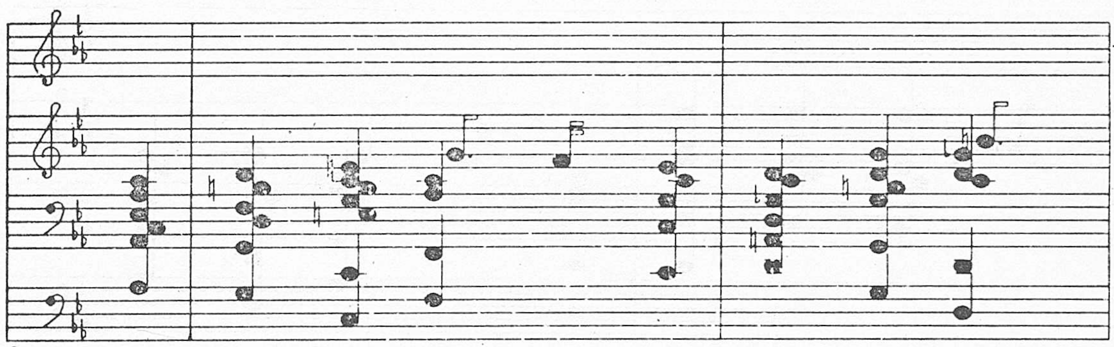
```
1
G#2  G#3  D#4  G#4  C5   D#5  24
1
C#2  C#3  F4   G#4  C#5  F5   24
1
G2   G3   F4   G4   B4   D#5  18
G2   G3   F4   G4   B4   D5   6
1
C2   C3   D#4  G4   C5   24
1
C4   G4   C5   D#5  G5   C6   100
```

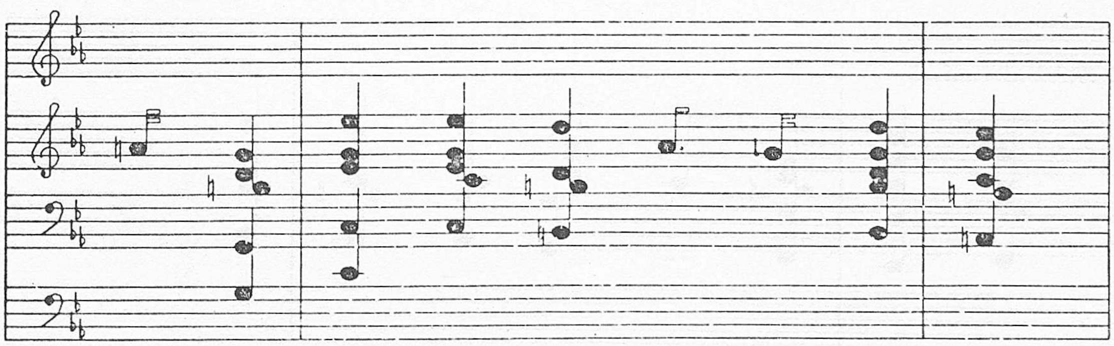
Prelude in C Minor

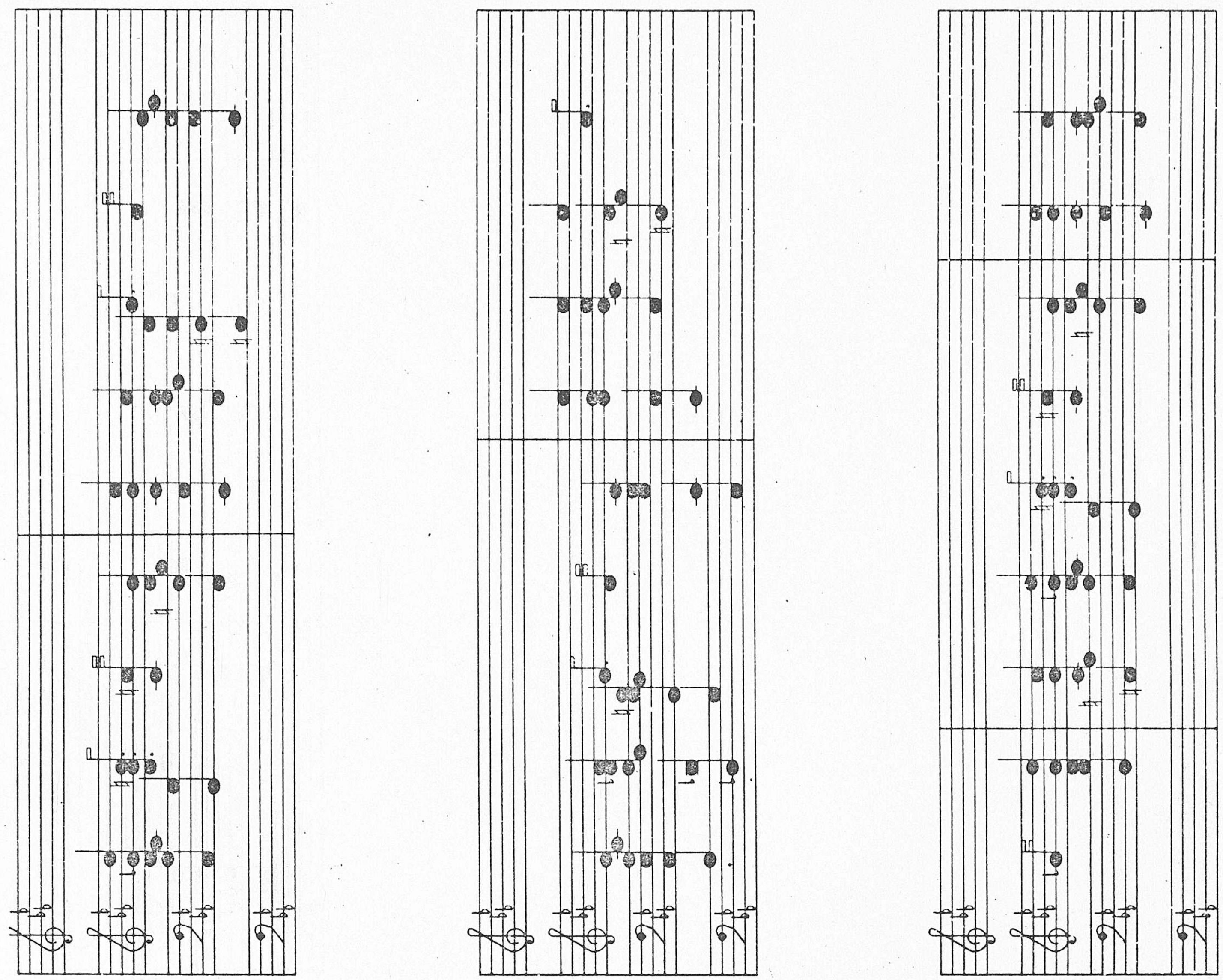
F Chopin

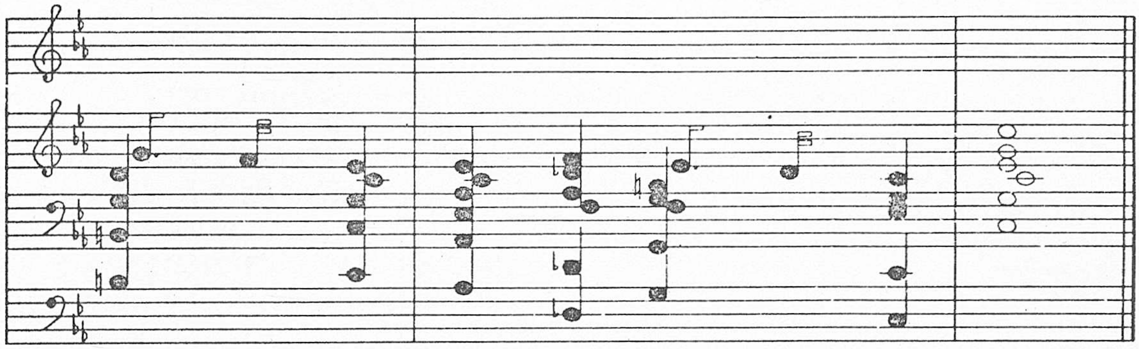
Op 28 , No 20











I'M IN THE MOOD FOR LOVE.   --SHEARING.

```
C4   G4   C5   E5   10
B3   G#4  D5   F5   5
B3   G#4  D5   F5   5
A3   E4   B4   D5   10
G3   D4   A#4  C5   5
G3   E4   A#4  C5   5
F3   C4   G4   A4   D5   10
F3   C4   F4   A4   D5   10
E3   B3   E4   G4   B4   D5   10
D#3  A#3  D#4  F#4  A#4  D5   10
D3   A3   D4   F4   A4   C5   D5  7
D3   A3   D4   F4   A4   C5   E5   7
F5  7
G3   F4   B4   E5   G5   5
G3   F4   5
G3   F4   B4   D#5  16
A5   10
C4   G#4  D#5  F#5  B5  10
C4   A4   E5   G5   C6   10
D4   C5   F5   A5   C6   10
D4   C6   2
D4   B4   F5   G#5  C6   10
E4   D5   G5   B5   D6   10
1
F4   D#5  G#5  B5   5
F4   D#5  G#5  D6   5
E4   D5   G5   B5   10
D#4  C#5  F#5  A#5  10
D4   C5   F5   A5   10
D4   B4   E5   G5   C6   10
D4   A4   D5   F5   C6   12
D4   G4   C5   E5   C6   10
G3   F4   C5   E5   A5   10
G3   E4   B4   D5   G5   5
G3   E4   B4   D5   A5   5
G3   D4   A4   C5   G5   10
G3   F4   G#4  B4   10
1
E4   B4   D5   G5   10
1
D#4  A#4  C#5  G5   10
1
G#3  D#4  C5   G5   10
1
C#4  B4   F5   G5   10
C4   G4   C5   E5   10
B3   G#4  D5   F5   5
B3   G#4  D5   E5   5
A3   E4   B4   D5   10
1
G3   D4   A#4  C5   5
```

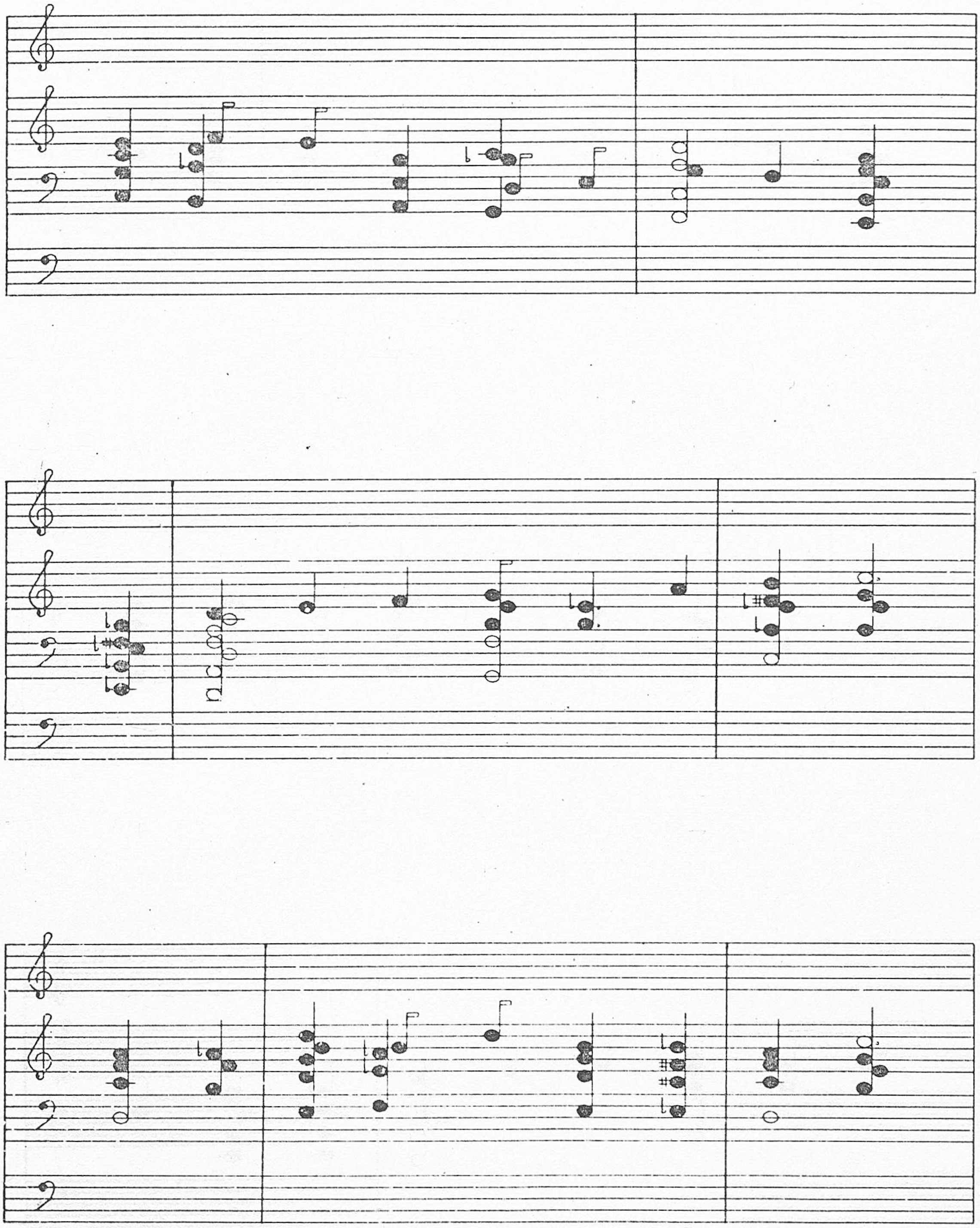
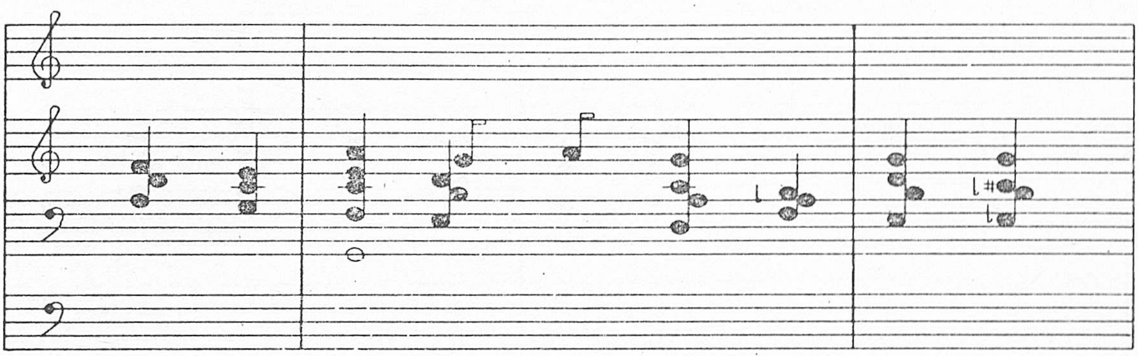
```
G3   E4   A#4  C5   5
F3   C4   G4   A4   D5   10
F3   C4   F4   A4   D5   10
E3   B3   E4   G4   B4   D5   10
D#3  A#3  D#4  F#4  A#4  D5   10
D3   A3   D4   F4   A4   C5   D5   10
D3   A3   D4   F4   A4   C5   E5   10
F5  10
1
G3   F4   B4   E5   G5   5
G3   F4   5
G3   F4   B4   D#5  16
A5   10
C4   G#4  D#5  F#5  B5   10
C4   A4   E5   G5   C6   10
D4   C5   F5   A5   C6   10
D4   C6   2
D4   B4   F5   G#5  C6   10
E4   D5   G5   B5   D6   10
1
F4   D#5  G#5  B5   5
F4   D#5  G#5  D6   5
1
E4   D5   G5   B5   10
D#4  C#5  F#5  A#5   10
D4   C5   F5   A5   10
D4   B4   E5   G5   C6   10
D4   A4   D5   F5   C6   12
D4   G4   C5   E5   C6   10
1
G3   F4   C5   E5   A5   10
G3   E4   B4   D5   G5   5
G3   E4   B4   D5   A5   5
G3   D4   A4   C5   G5   10
F3   F4   G#4  B4   10
C4   E4   G4   A4   C5   10
1
A#3  F4   A4   D5   F5   5
A#3  F4   A#4  D5   F5   5
A3   E4   G4   C5   E5   10
1
G3   D4   F4   C5   5
G3   D4   2
G3   D4   E4   A#4  C5   5
F3   C4   A4   D5   10
E3   B3   G4   B4   E5   5
E3   B3   G4   B4   F5   5
D3   A3   F4   C5   E5   10
G3   F4   G#4  B4   D5   10
C4   B4   D5   E5   G5   10
1
B3   A4   C#5  D#5  G5   10
A#3  G#4  C5   D5   G5   12
A3   G4   B4   C#5  G5   10
```

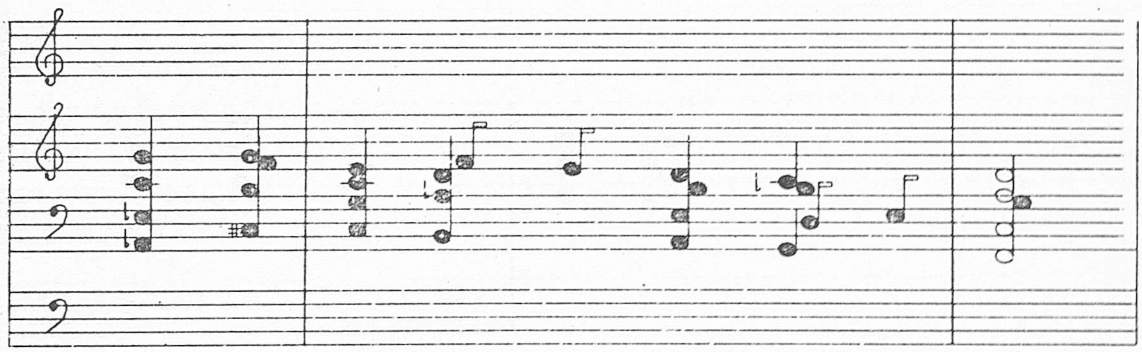
D4 F4 A4 C5 D5 10
D4 F4 A4 C5 E5 10
F5 10
1
G3 F4 G#4 B4 E5 10
G3 F4 G#4 B4 D5 10
C4 E4 B4 D5 G5 40
1
B4 C5 E5 F#5 10
1
A4 C5 E5 G5 5
1
G4 C5 E5 A5 5
1
F#4 A#4 E5 G5 10
B3 A4 D#5 F5 10
E4 A4 B4 D#5 B5 10
E4 2
E4 G4 C#5 E5 B5 10
E4 F#4 D#5 F#5 B5 10
B5 2
E4 B4 E5 G5 B5 10
1
D#4 B4 G5 B5 10
D#4 B4 G5 C6 10
B5 10
1
B3 A#4 C5 F#5 A#5 D6 10
B3 A4 C5 F#5 A5 D6 10
1
G3 F4 A4 C5 D5 10
G3 F4 A4 C5 E5 10
D5 10
1
G#3 G#4 A#4 C#5 F5 10
G#3 G#4 A4 C5 F5 10
C#4 G4 C#5 F5 10
B3 F#4 B4 D#5 F#5 5
B3 F#4 B4 D#5 F5 5
1
A#3 F4 A#4 D#5 10
G#3 D#4 B5 C#5 5
G#3 F4 B5 C#5 5
F#3 C#4 G#4 A#5 D#5 10
F#3 C#4 F#4 A#5 D#5 10
F3 C4 F4 G#4 C5 D#5 10
E3 B3 E4 G4 B4 D#5 10
D#3 A#3 D#4 F#4 A#4 D#5 10
1C#4 F4 A#4 C#5 F5 5
C#4 F#4 A#4 C#5 F#5 5
1
C4 G#4 A#4 D#5 G#5 10
1
C4 A#4 D#5 F#5 A#5 10

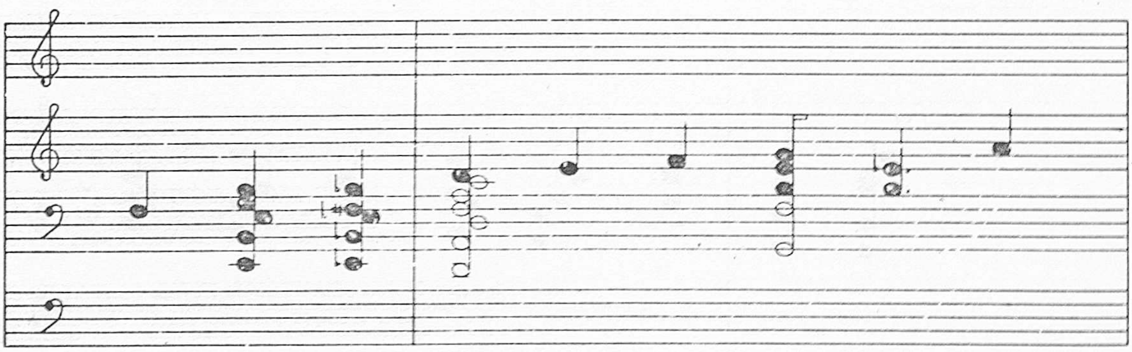
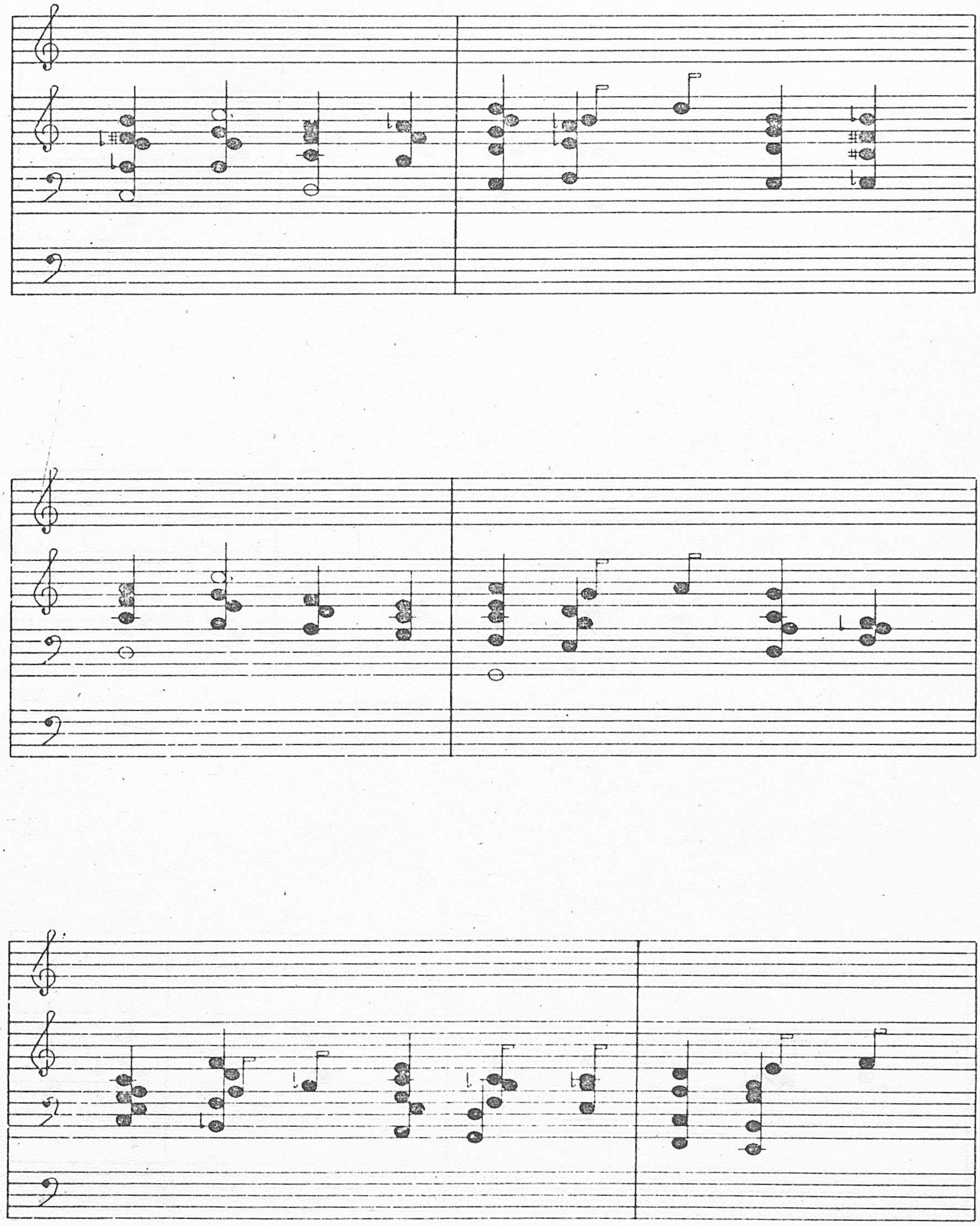
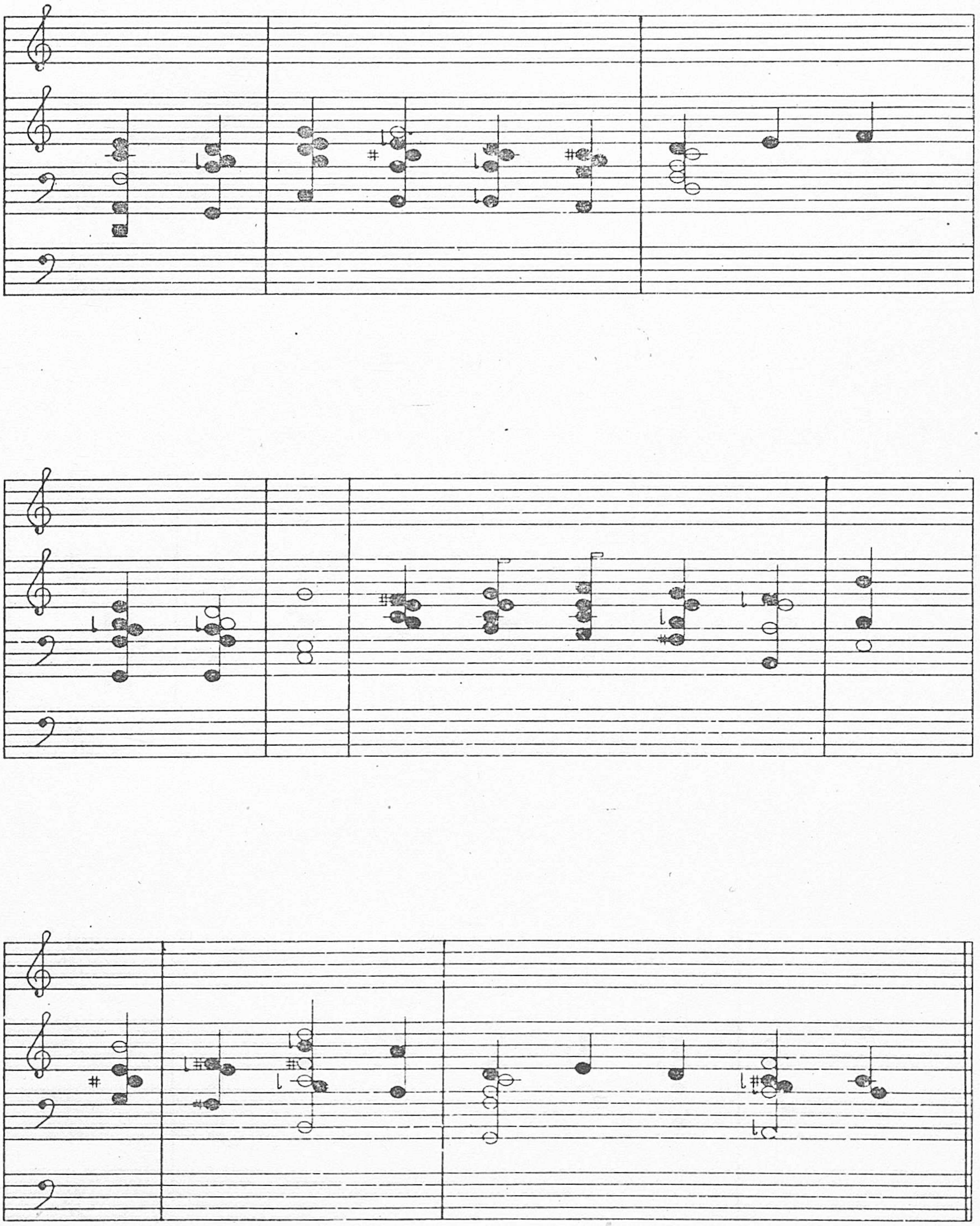
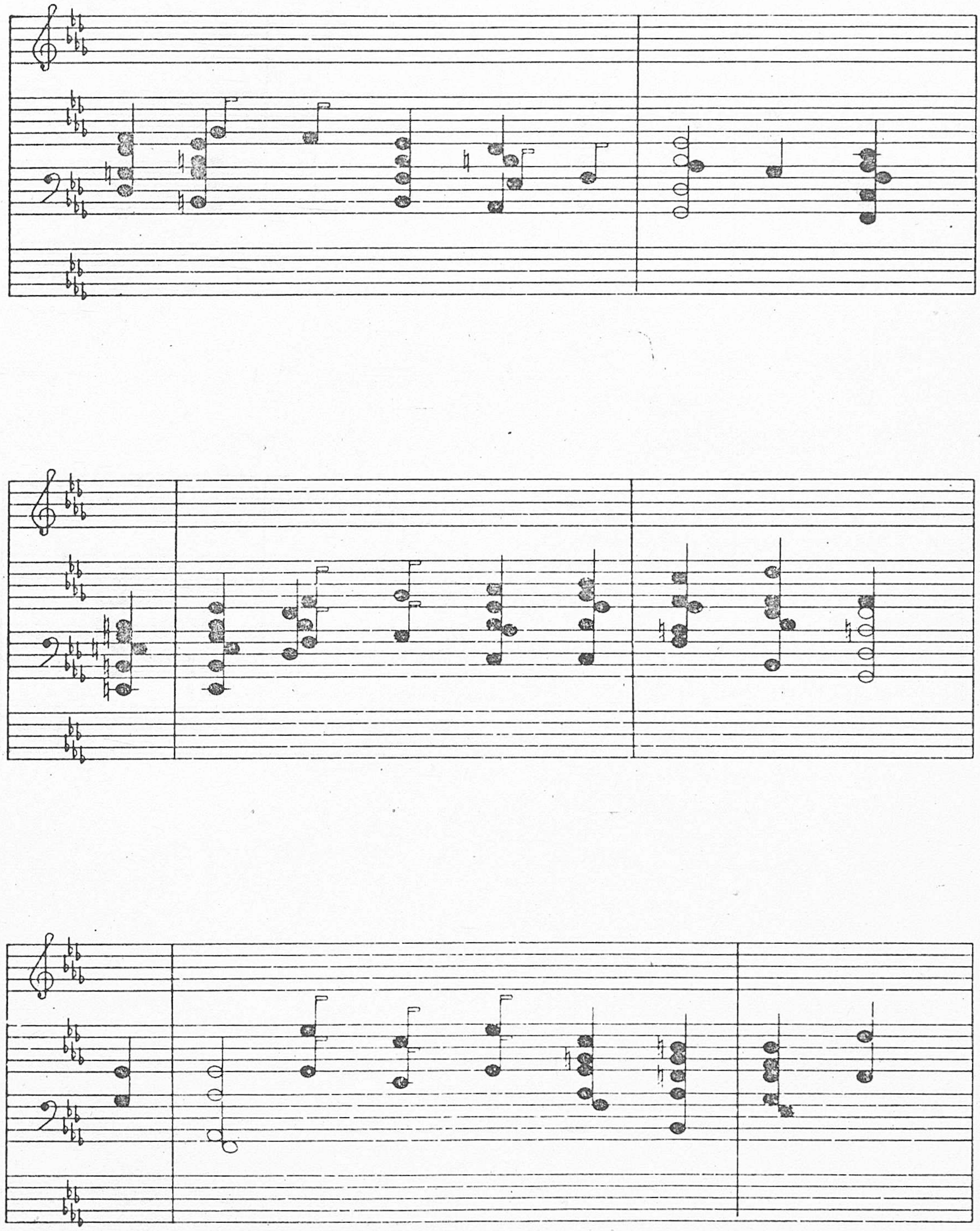
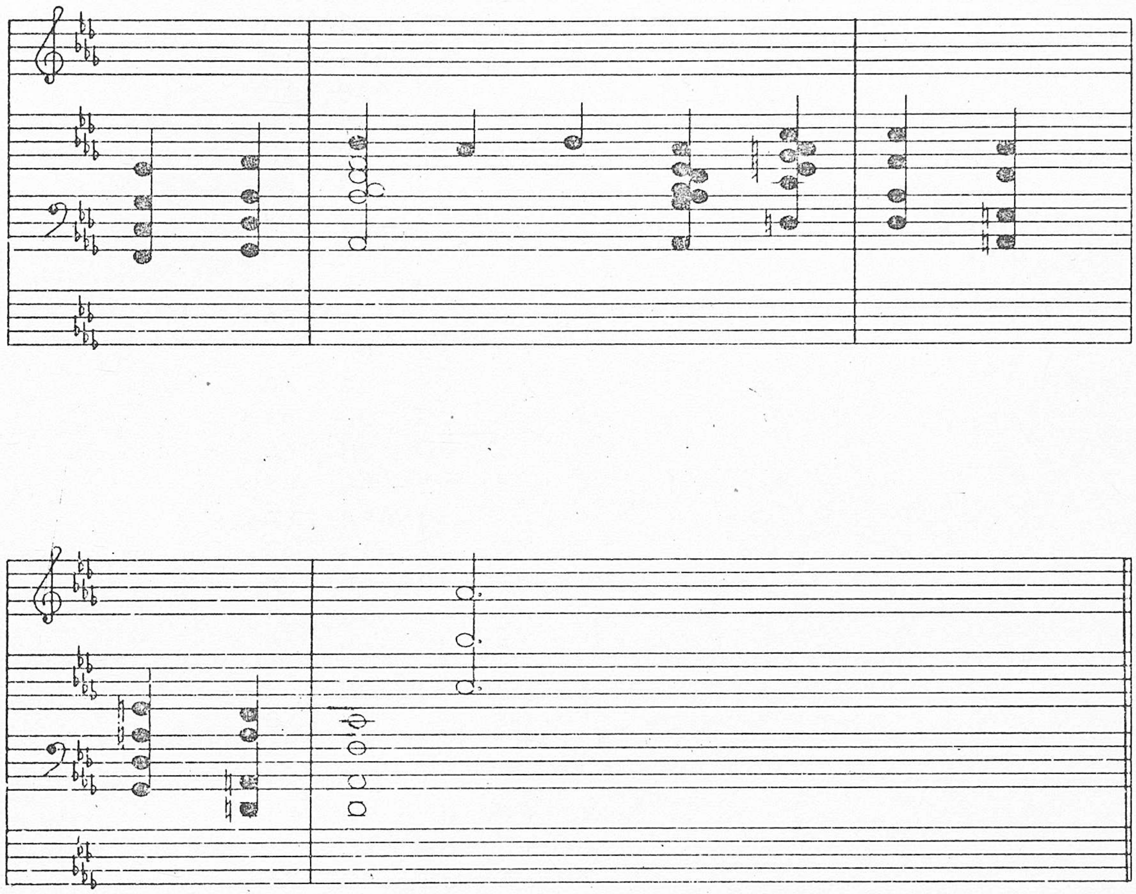
```
1
F4   A4   D#5  F5   C6   10
1
A#3  A#4  C#5  F5   C#6  10
1
F#3  C#4  A4   C#5  F5   10
F#3  C#4  F#4  A4   C#5  D#5  10
1
F3   G#3  G#4  D#5  20
1
D#5  D#6  6
C5   C6   6
D#5  D#6  6
1
F4   G#4  D#5  G5   C6   10
1
A#3  G#4  D5   F#5  B5   10
D#4  F#4  C#5  F5   A#5  10
1
C#5  C#6  10
F3   C4   G#4  D#5  10
F#3  C#4  A#4  F5   10
1
G#3  G#4  A#4  C#5  F5   A#5  10
G#3  G#4  A#4  C#5  F5   G#5  10
A#5  10
1
G#3  F#4  G#4  A#4  C#5  D#5  G#5  10
1
D4   C5   E5   F#5  A5   C6   10
C#4  G#4  F5   C#6  10
A3   E4   C#5  G#5  10
F#3  C#4  A4   E5   10
D3   A3   F#4  C#5  10
C#3  G#3  F4   C5   40
G#5  G#6  G#7  32
```

I'm in the Mood for Love

McHugh and Fields

Arrt. G Shearing







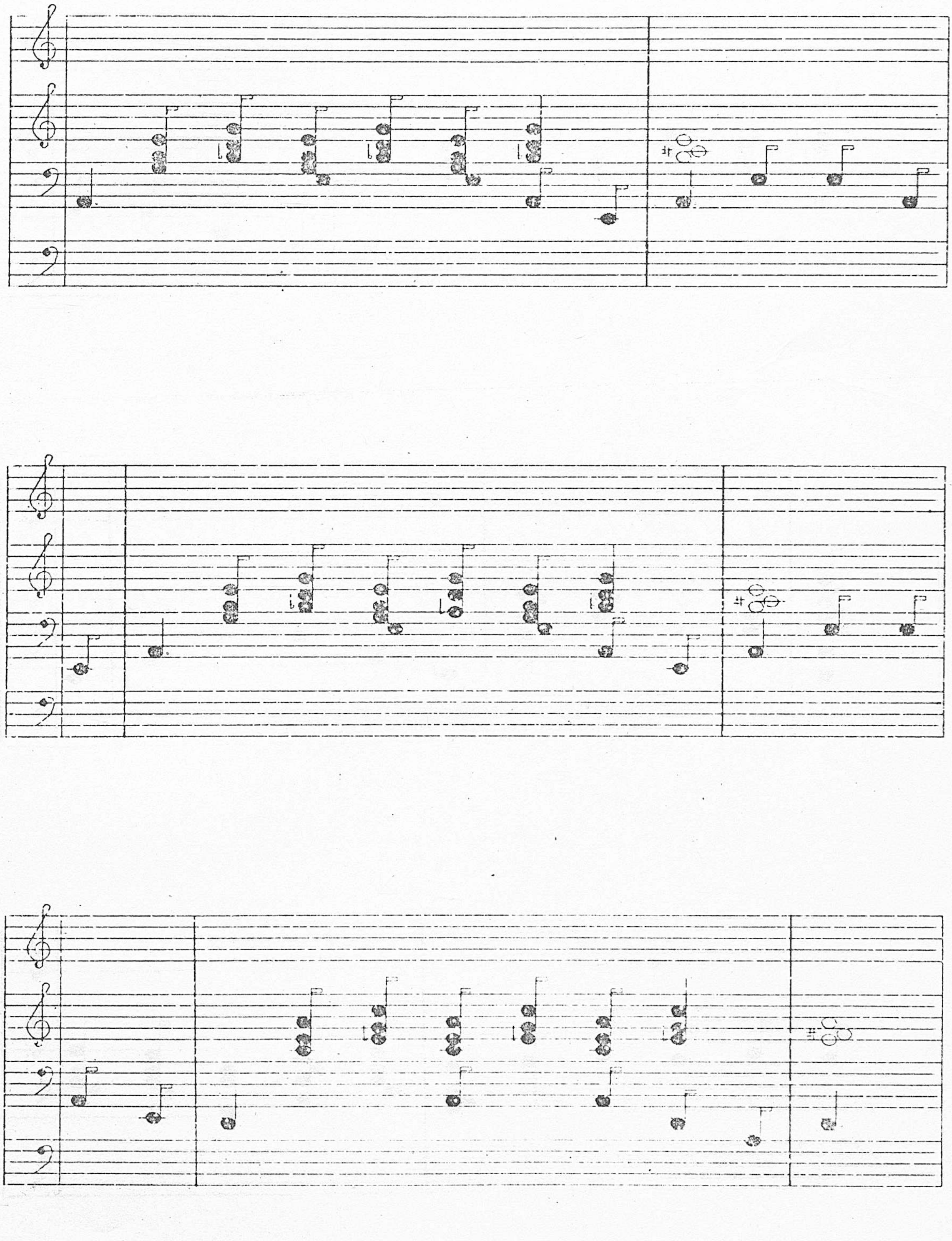
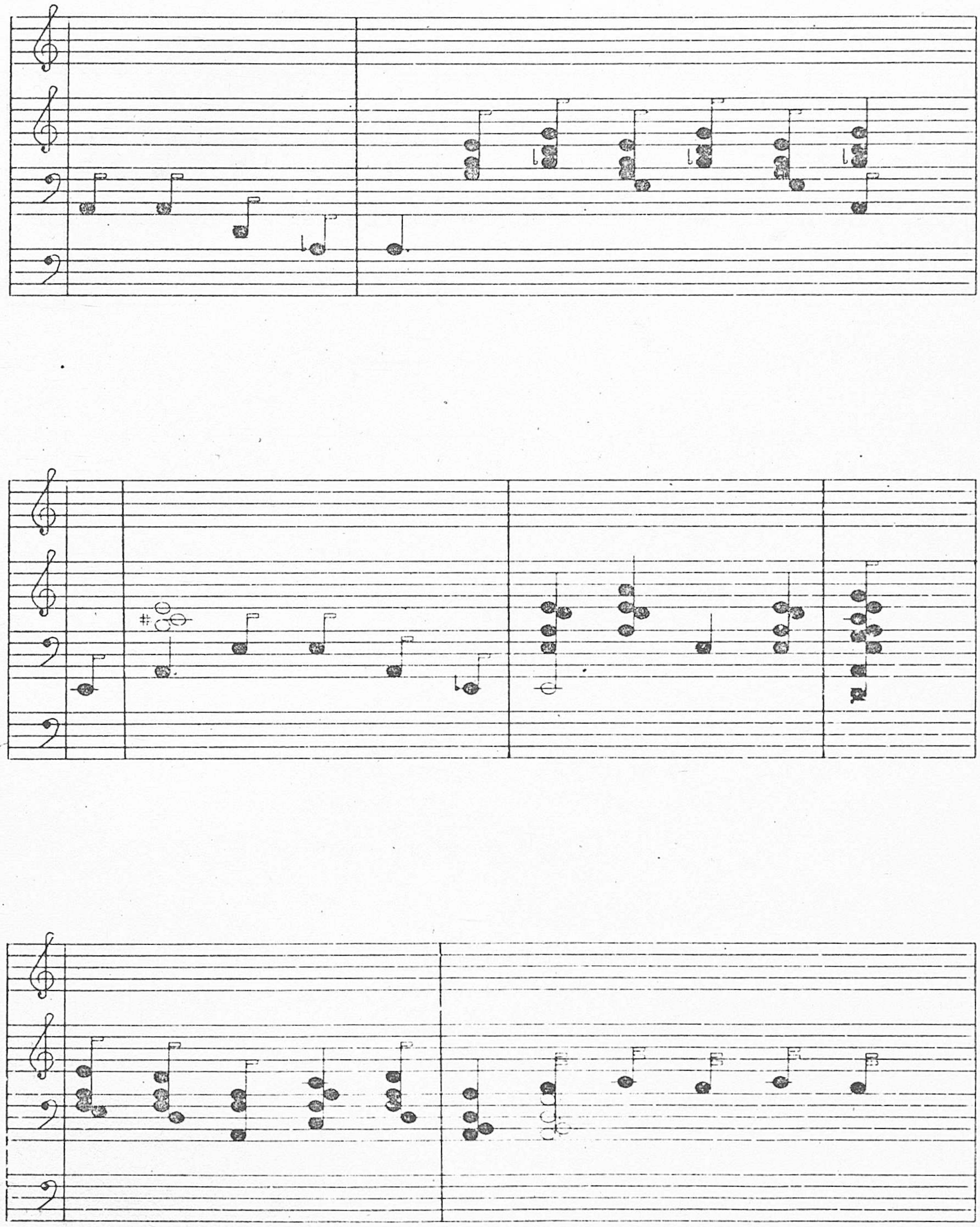
SOOZ BLOOZ.   D MOORE.

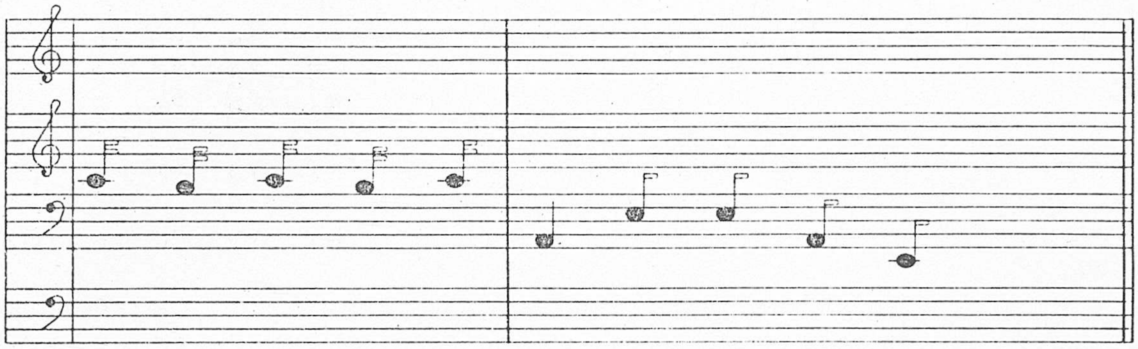
A3	5			
A3	G4	B4	E5	5
A3	A#4	D5	G5	5
E4	G4	B4	E5	5
A#4	D5	G5	5	
E4	G4	B4	E5	5
A3	A#4	D5	G5	5
E3	A#4	D5	G5	5
A3	B4	C#5	E5	15
E4	B4	C#5	E5	5
B4	C#5	E5	5	
E4	B4	C#5	E5	5
A3	B4	C#5	E5	5
E3	B4	C#5	E5	5
A3	5			
A3	G4	B4	E5	5
A3	A#4	D5	G5	5
E4	G4	B4	E5	5
A#4	D5	G5	5	
E4	G4	B4	E5	5
A3	A#4	D5	G5	5
E4	A#4	D5	G5	5
A3	B4	C#5	E5	15
E4	B4	C#5	E5	5
B4	C#5	E5	5	
E4	B4	C#5	E5	5
A3	B4	C#5	E5	5
E3	B4	C#5	E5	5
D3	5			
D3	C5	E5	A5	5
D3	D#5	G5	C6	5
A3	C5	E5	A5	5
D#5	G5	C6	5	
A3	C5	E5	A5	5
D3	D#5	G5	C6	5
A2	D#5	G5	C6	5
D3	E5	F#5	A5	15
A3	E5	F#5	A5	5
E5	F#5	A5	5	
A3	E5	F#5	A5	5
D3	E5	F#5	A5	5
G#2	E5	F#5	A5	5
A2	5			
A2	G4	B4	E5	5
A2	A#4	D5	G5	5
E4	G4	B4	E5	5
A#4	D5	G5	5	
E4	G4	B4	E5	5
A3	A#4	D5	G5	5

E4 A#4 D5 G5 5
A3 B4 C#5 E5 15
E4 B4 C#5 E5 5
B4 C#5 E5 5
E3 B4 C#5 E5 5
A3 B4 C#5 E5 5
D#3 B4 C#5 E5 5
E3 A4 D5 E5 1Ø
E3 E4 A4 D5 E5 A5 5
E3 5
D#4 2
E4 8E
E4 A4 D5 E5 1Ø
D3 A3 E4 G4 A4 C5 E5 G5 5
E4 F4 A4 E5 5
D4 F4 A4 D5 5
A3 F4 A4 5
C4 F4 A4 C5 1Ø
D4 F4 A4 D5 5
5
A3 B3 D3 A4 1Ø
A3 B3 D3 G4 B4 3
A3 B3 D3 G4 C5 3
A3 B3 D3 G4 B4 3
A3 B3 D3 G4 C5 3
A3 B3 D3 G4 B4 3
A3 B3 D3 G4 C5 3
A3 B3 D3 G4 B4 3
A3 B3 D3 G4 C5 3
A3 B3 D3 G4 B4 3
A3 B3 D3 G4 C5 3
A3 15
E4 5
5
E4 5
A3 5
E3 5

Sooz Blooz

Dudley Moore



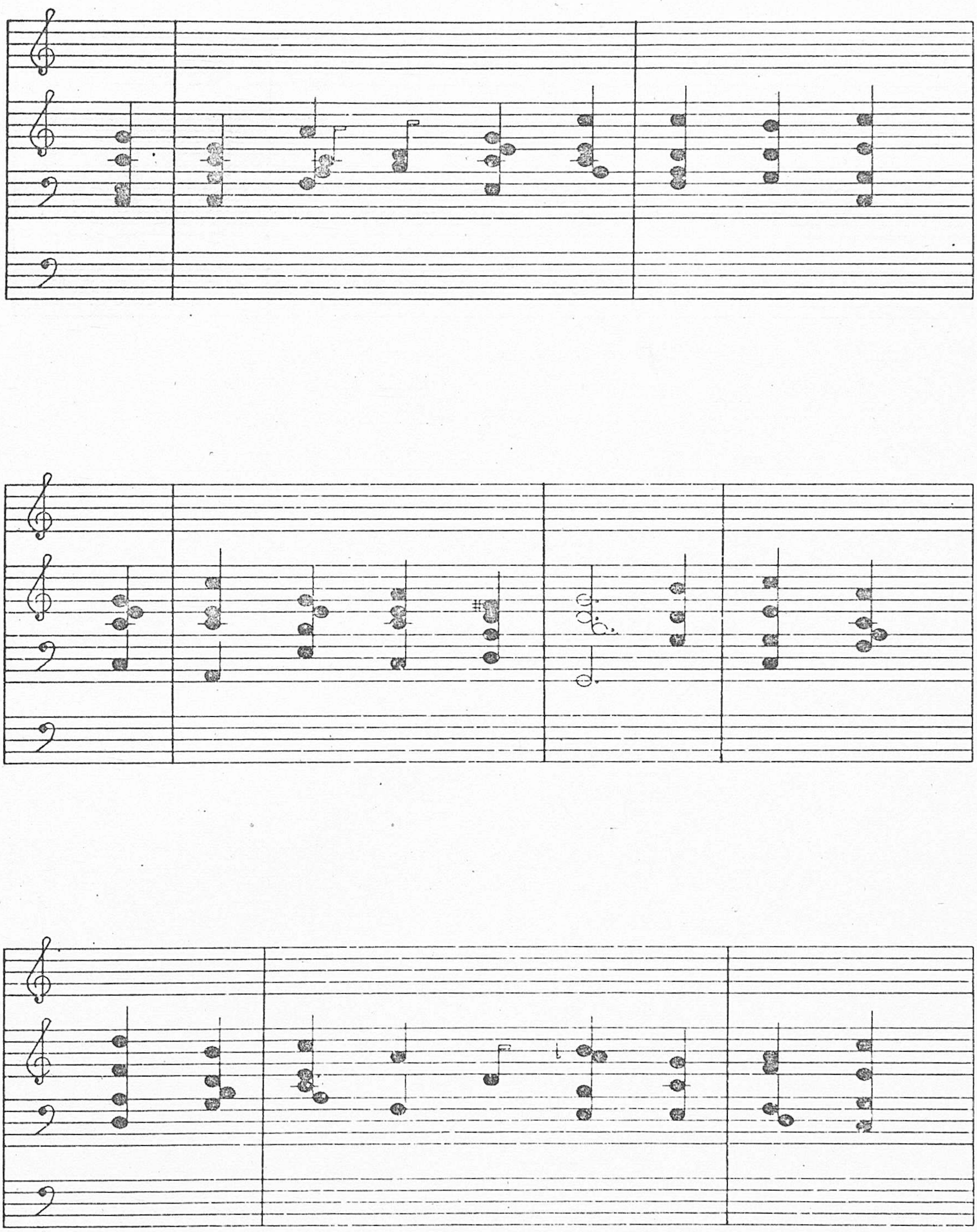
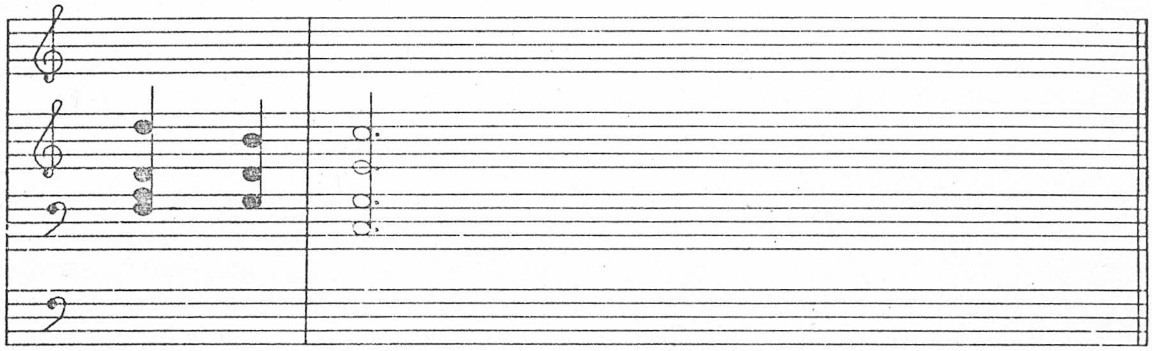


O GOD , OUR HELP IN AGES PAST.

```
C4   E4   C5   G5   1Ø
1
C4   G4   C5   E5   1Ø
1
F4   A4   C5   A5   5
F4   B4   D5   A5   5
E4   C5   E5   G5   1Ø
1
A4   C5   E5   C6   1Ø
1
F4   A4   D5   C6   1Ø
1
G4   D5   B5   1Ø
C4   G4   E5   C6   1Ø
1
C4   C5   E5   G5   1Ø
1
A4   C5   E5   C6   1Ø
1
E4   B4   E5   G5   1Ø
1
C4   C5   E5   A5   1Ø
D4   A4   D5   F#5  1Ø
1
G3   B4   D5   G5   28
1
G4   D5   B5   1Ø
1
C4   G4   E5   C6   1Ø
F4   A4   C5   A5   1Ø
1
D4   A4   F5   D6   1Ø
G4   B4   D5   B5   1Ø
A4   C5   E5   C6   1Ø
F4   E5   A5   6
F4   D5   A5   5
E4   B4   G#5  B5   1Ø
1
E4   C5   G5   1Ø
D4   F4   F5   A5   1Ø
C4   G4   E5   C6   1Ø
F4   A4   D5   D6   1Ø
1
G4   D5   B5   1Ø
1
C4   G4   E5   C6   34
```

O God, our help in Ages Past

W Croft

1671 - 1727



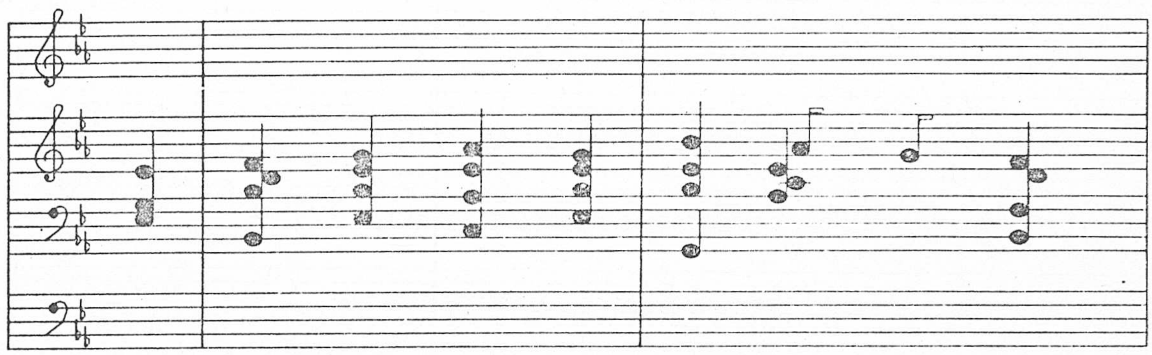
THERE IS A GREEN HILL FAR AWAY.

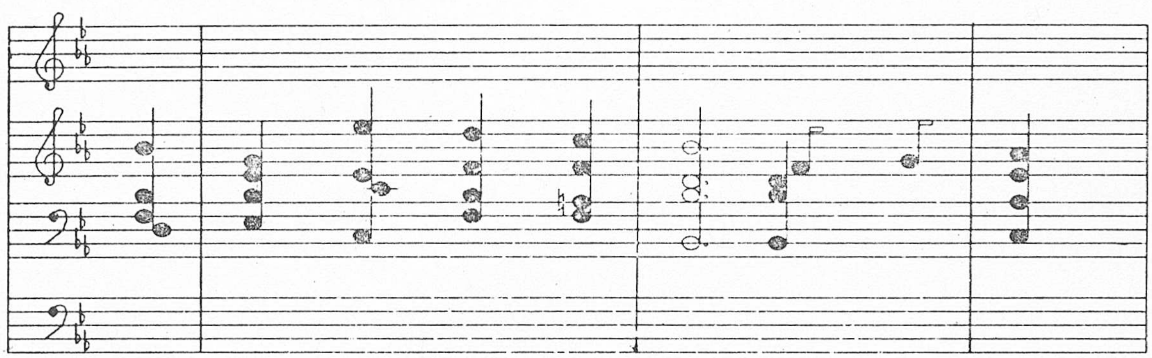
D#4 G4 D#5 10
A#3 A#4 D5 F5 10
1
D#4 A#4 D#5 G5 10
1
C4 G#4 D#5 G#5 10
1
G3 A#4 D#5 A#5 10
1
G#3 C5 D#5 G#5 5
G#3 C5 D#5 G5 5
A#3 F4 D5 F5 10
1
D4 F4 A#4 A#5 10
1
D#4 A#4 D#5 G5 10
1
C4 C5 D#5 D#6 10
F4 A#4 F5 D6 10
1
F4 A4 F5 D6 10
1
A#3 A#4 D5 A#5 34
1
A#3 A#4 D5 F5 5
A#3 A#4 D5 G5 5
C4 G#4 D#5 A5 10
1
C4 C5 D#5 G#5 10
1
D4 A#4 F5 G#5 10
1
D#4 A#4 G5 10
1
G#4 C5 G5 C6 10
1
G#4 C5 F5 C6 10
G4 D5 G5 B5 10
1
G4 D5 G5 B5 10
G#4 C5 D#5 C6 10
1
G#3 C5 F5 G#5 10
A#3 A#4 D#5 G5 10
A#4 6
A#3 D5 F5 10
G#4 5
D#4 G4 D#5 34

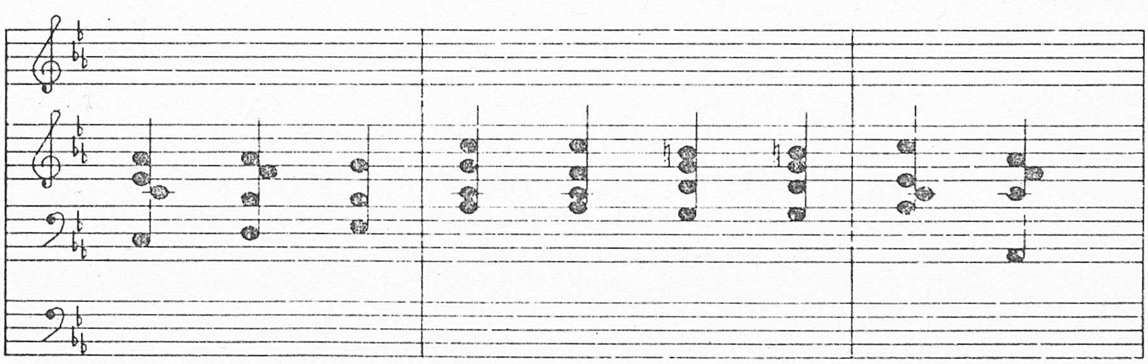
There is a Green Hill Far Away

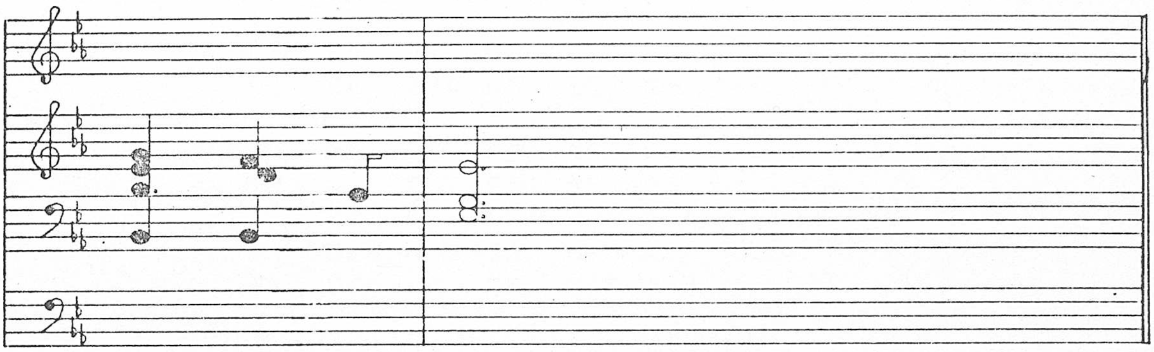
W Horsley

1774 - 1858









THAT OLD FEELING.   FAYNE.

```
G3   D4   A#4  D5   F5   A5   24
G3   D4   A#4  D5   F5   C6   24
1
C3   E4   A#4  D5   E5   A5   24
C3   E4   A#4  D5   E5   C6   24
1
F3   A4   C5   E5   A5   24
F3   A4   C5   E5   F5   24
1
F3   D#4  A4   C5   D#5  A5   24
F3   D#4  A4   C5   D#5  F#5  C6   24
1
A#3  C#5  F5   G#5  C6   D#6  48
1
D#4  G4   A#4  D#5  F5   G5   C6   D#6  48
1
A3   C5   D#5  G5   B5   D6   24
A3   C5   D#5  C5   A5   24
1
D4   F#4  A4   C5   D#5  F#5  A5   53
1
G3   A#4  D5   F#5  A#5  24
G3   A#4  D5   F#5  D6   24
1
A3   C5   D#5  A5   24
D#5  31
A3   F#4  C5   G#5  D6   24
1
G3   D4   A#4  D5   F5   A#5  24
G3   D4   A#4  D5   F5   G5   24
1
C3   E4   A#4  D5   E5   A#5  24
C3   E4   A#4  D5   E5   C6   24
1
G#3  B4   C#5  F#5  B5   C#6  46
1
C#4  F4   G#4  B4   C#5  F5   B5   C#6  47
G3   A#4  D5   F5   A5   C6   24
G3   A#4  D5   F5   G5   24
1
C4   E4   G4   A#4  C#5  E5   G5   51
1
D3   F4   A4   C5   D5   24
D3   F4   A4   C5   F5   24
1
G3   F4   A4   C#5  F5   24
G3   F4   A4   C#5  24
F5   3Ø
1
F#3  G4   B4   D5   F#5  A5   24
F#3  G4   B4   D5   F#5  25
```

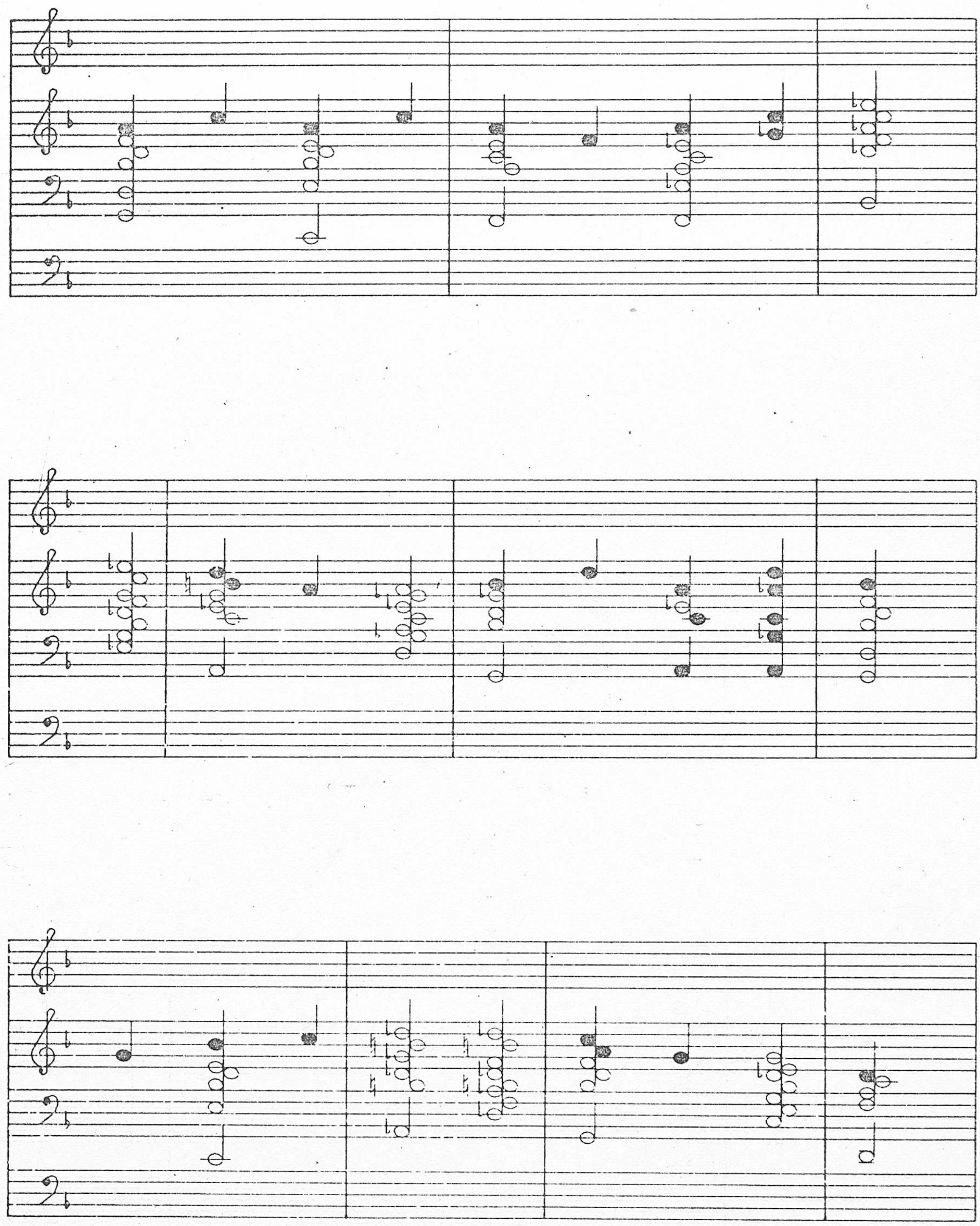
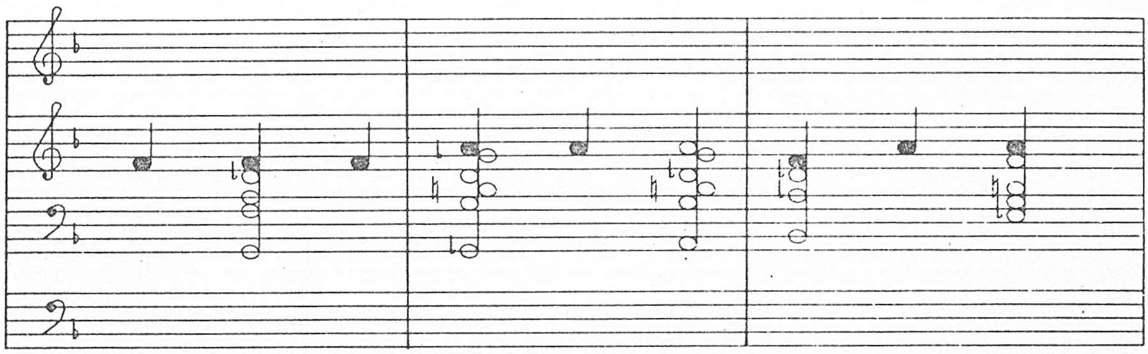
A5 24
1
F#3 G4 B4 D5 F#5 A5 24
F#3 G4 B4 D5 F#5 24
A5 24
1
A3 G4 B4 C#5 G5 A5 41
1
A#3 G#4 C#5 F5 24
A#3 G#4 C#5 A5 24
1
D#4 G4 B4 F5 A5 24
D#4 G4 B4 F5 24
A5 24
1
A3 G4 C5 E5 G5 B5 D6 52
1
D3 F#4 C5 F5 D6 47
1
D3 F4 C5 E5 A5 24
D3 F4 C5 E5 D6 24
1
G3 F4 D5 F5 B5 D6 24
G3 F4 D5 F5 B5 24
D6 24
D3 F4 D5 G#5 C6 E6 24
D3 F4 F5 G#5 C6 24
E6 24
1
G3 F4 F5 G#5 B5 E6 53
1
A#3 F4 C#5 F5 G#5 24
A#3 F4 C#5 F5 F6 24
1
D#4 G4 A#4 C#5 F5 A5 C6 F6 24
D#4 G4 A#4 C#5 F5 A5 C6 24
F6 31
G3 F4 A#4 D#5 F5 G5 50
1
C3 E4 A#4 D5 E5 F#5 51
1
F3 C4 A4 C5 E5 A5 24
F3 C4 A5 C5 E5 C6 24
1
G3 D4 A#4 D5 F5 A5 24
G3 D4 A#4 D5 A5 C6 24
1
A3 E4 C5 D5 E5 A5 24
A3 E4 C5 D5 E5 F5 24
1
A3 D#4 C5 D#5 A5 24
A3 D#4 C5 D#5 F#5 C6 24
1
E4 D5 E5 G5 B5 D#6 45

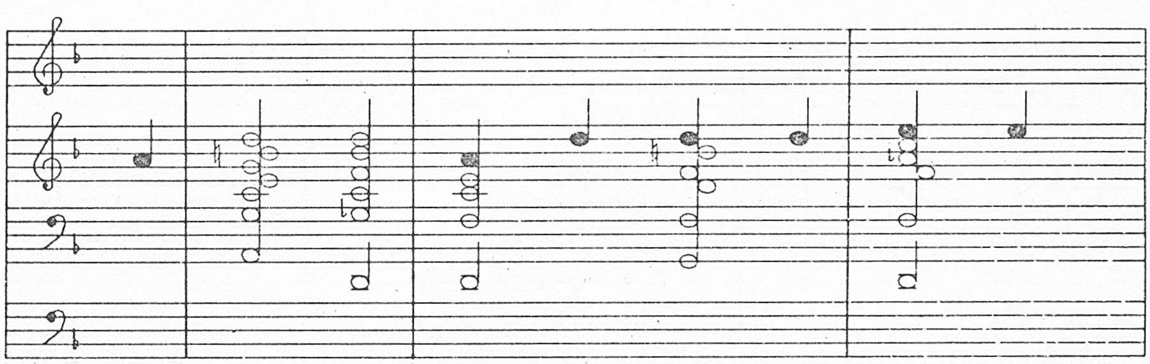
1
A3 C#5 D#5 G5 A#5 E6 46
1
A3 C5 D#5 G5 A#5 D6 24
A3 C5 D#5 G5 A5 24
1
D4 F#4 A4 C5 D#5 F#5 A5 52
1
G3 A#4D5 F5 G5 A#5 24
1
G3 F#4 D5 F#5 A#5 24
G3 F#4 D5 F#5 D6 24
5
G3 F4 A#4 C#5 F5 A#5 24
G3 F4 A#4 C#5 F5 G5 3Ø
4
C3 A4 C#5 E5 F#5 A#5 24
C3 E4 A#4 C#5 E5 F#5 D6 24
1
C4 G4 E5 G5 A#5 D6 F6 48
2
E4 G#4 B4 D5 F5 G#5 E6 54
3
A#3 C#5 F5 G#5 24
A#3 C#5 F5 D#6 24
2
D#4 G4 A#4 C#5 F5 G5 C6 E6 24
D#4 G4 A#4 C#4 F5 G5 C6 24
E6 24
1
A3 C4 D#5 G5 A#5 D#6 24
A3 C5 D#5 G5 24
A#5 D6 14
A5 C6 14
2
D4 F#4 A4 C5 A5 C6 14
D4 F#4 A5 C5 G5 A#5 14
D4 F#4 A5 C5 D#5 A5 24
1
G3 D4 A#4 D5 F5 24
G3 D4 A#4 D5 A5 C6 24
2
C4 A#4 E5 A5 C6 24
C4 A#4 E5 G#5 B5 24
G5 A#5 24
2
G3 D4 F5 A#5 C#6 F6 47

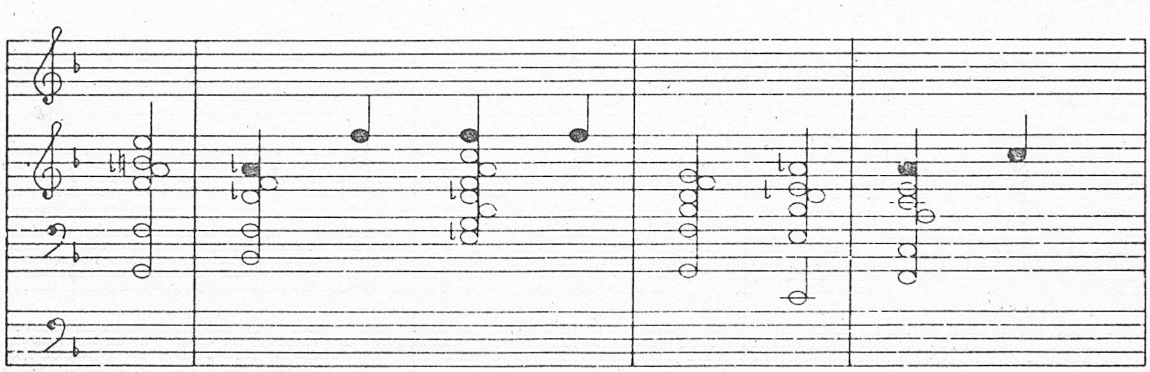
1
C4 A#5 C#5 F5 24
C4 A#5 24
C#5 E5 G5 24
2
D3 D#4 A4 C#5 F5 A5 46
1
A3 G4 A4 C#5 E5 A5 24
A3 G4 24
A4 C#5 E5 A5 24
1
D3 D#4 A4 C5 D5 A5 24
D3 D#4 D5 F5 A5 D6 24
D#5 G#5 B5 D#6 24
2
G3 A#4 D5 F5 A5 D6 24
G3 A#4 24
F5 24
1
C3 E4 A#4 C#5 F5 A5 24
C3 E4 24
F5 14
A#4 C#5 E5 A5 24
2
F3 C4 G4 A4 C5 D5 F5 94

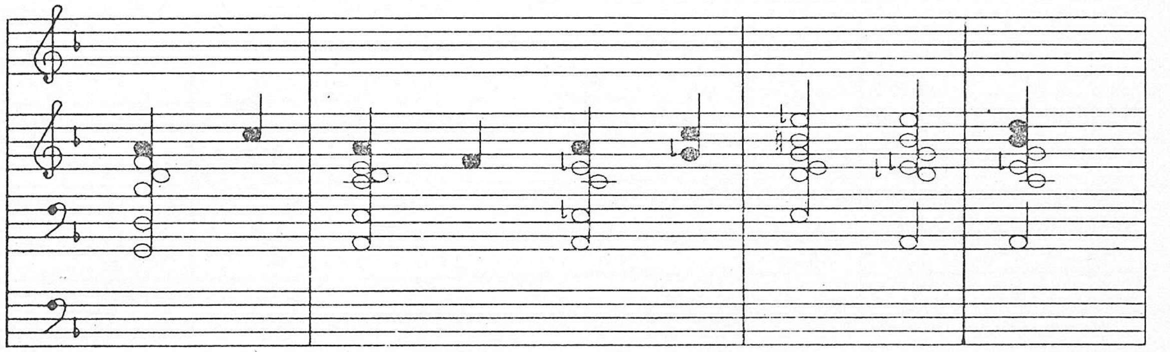
That Old Feeling

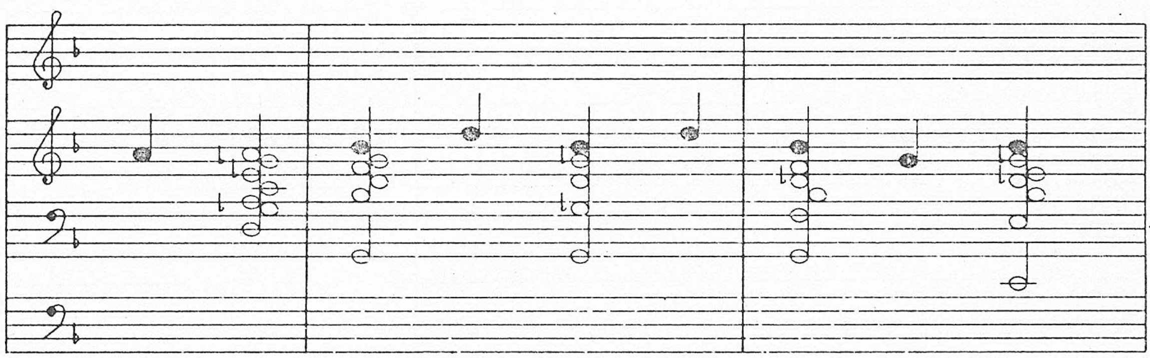
L Brown

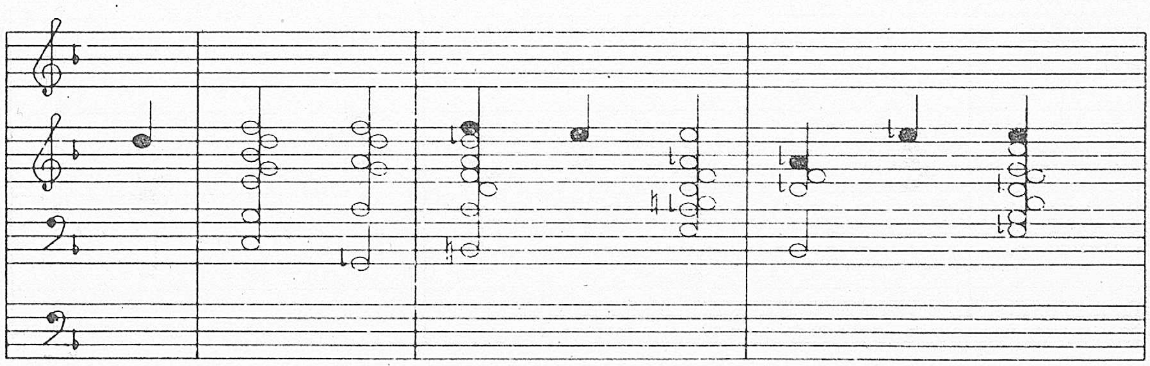
and S Fain



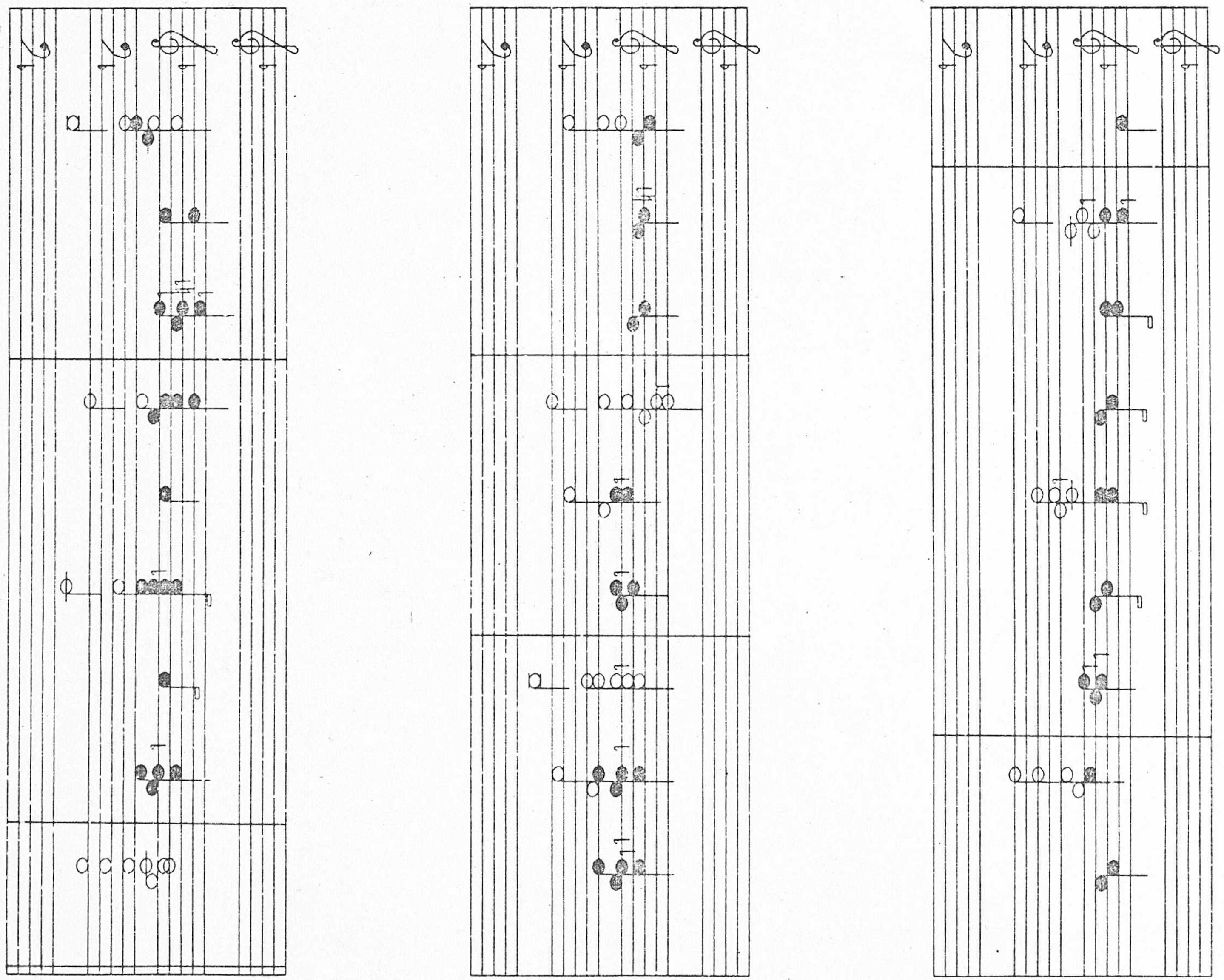






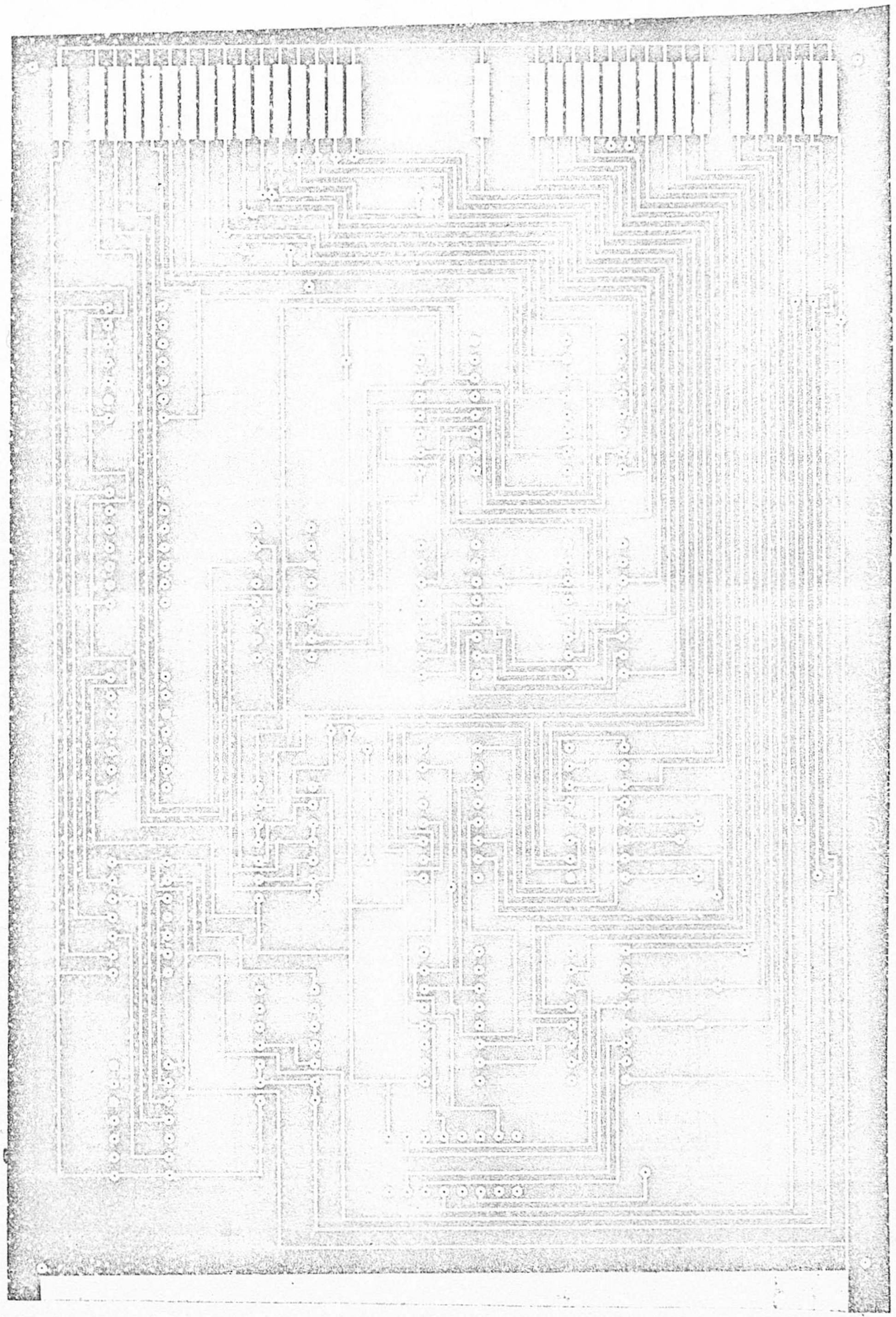




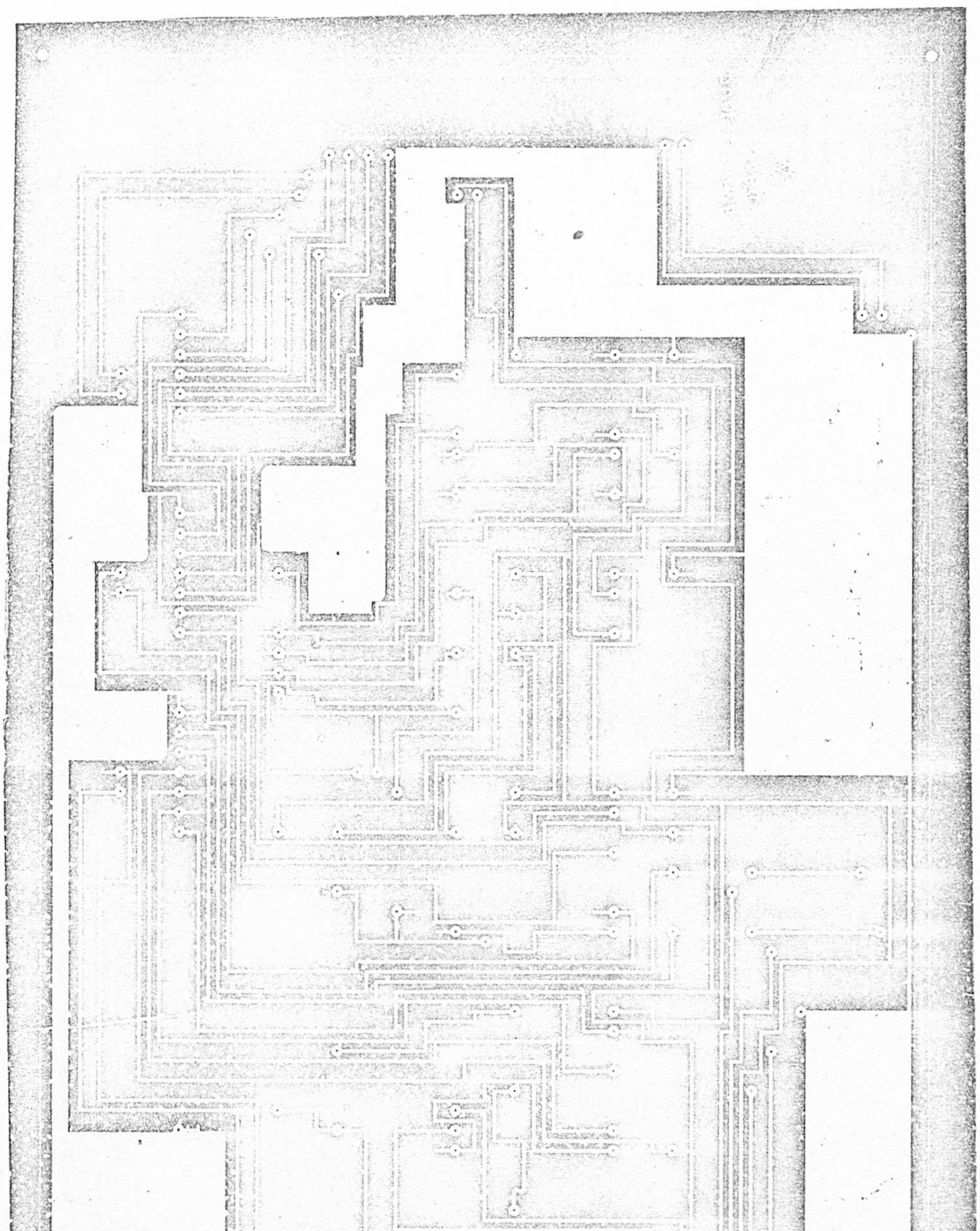


APPENDIX E

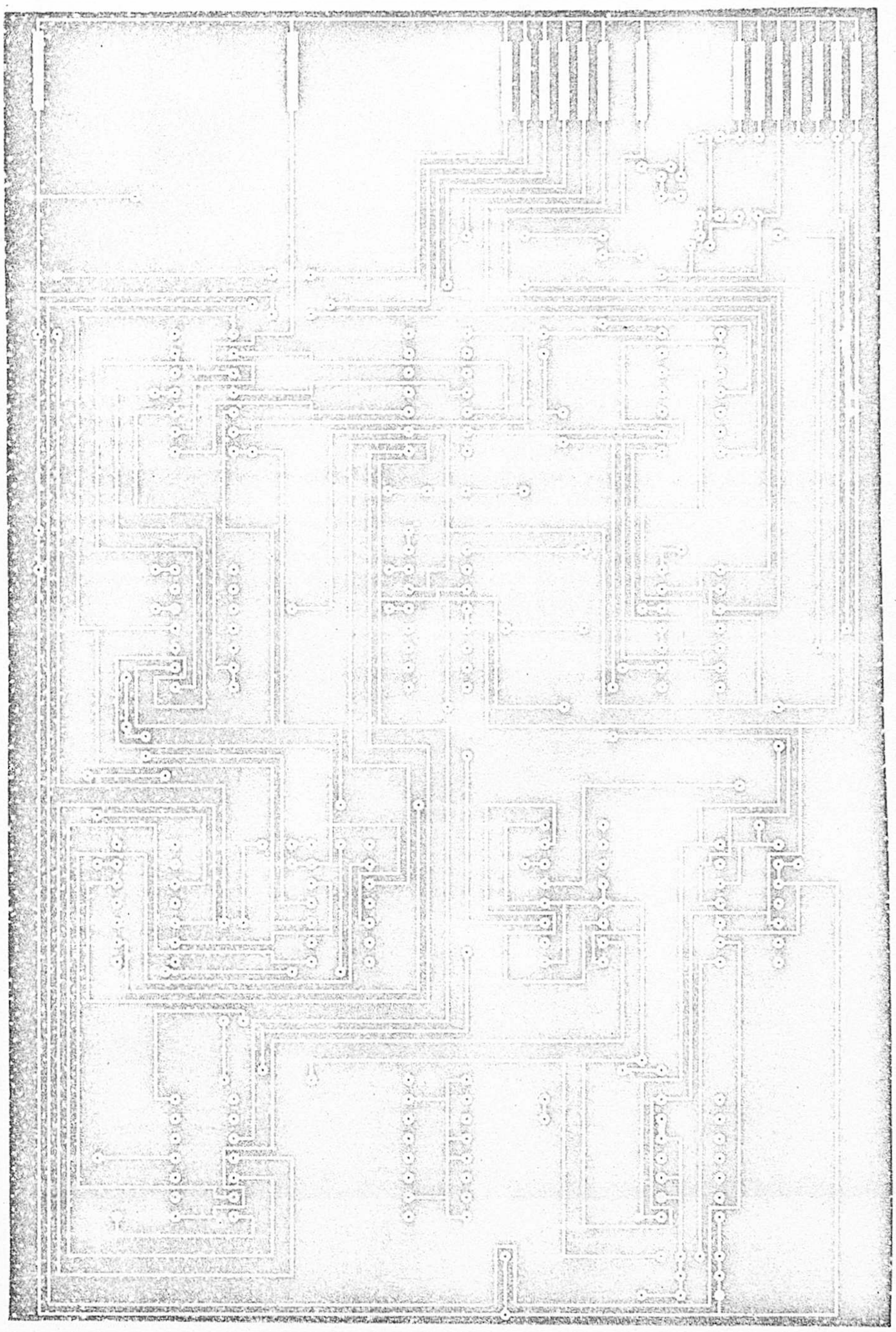
Printed Circuit Board Masks



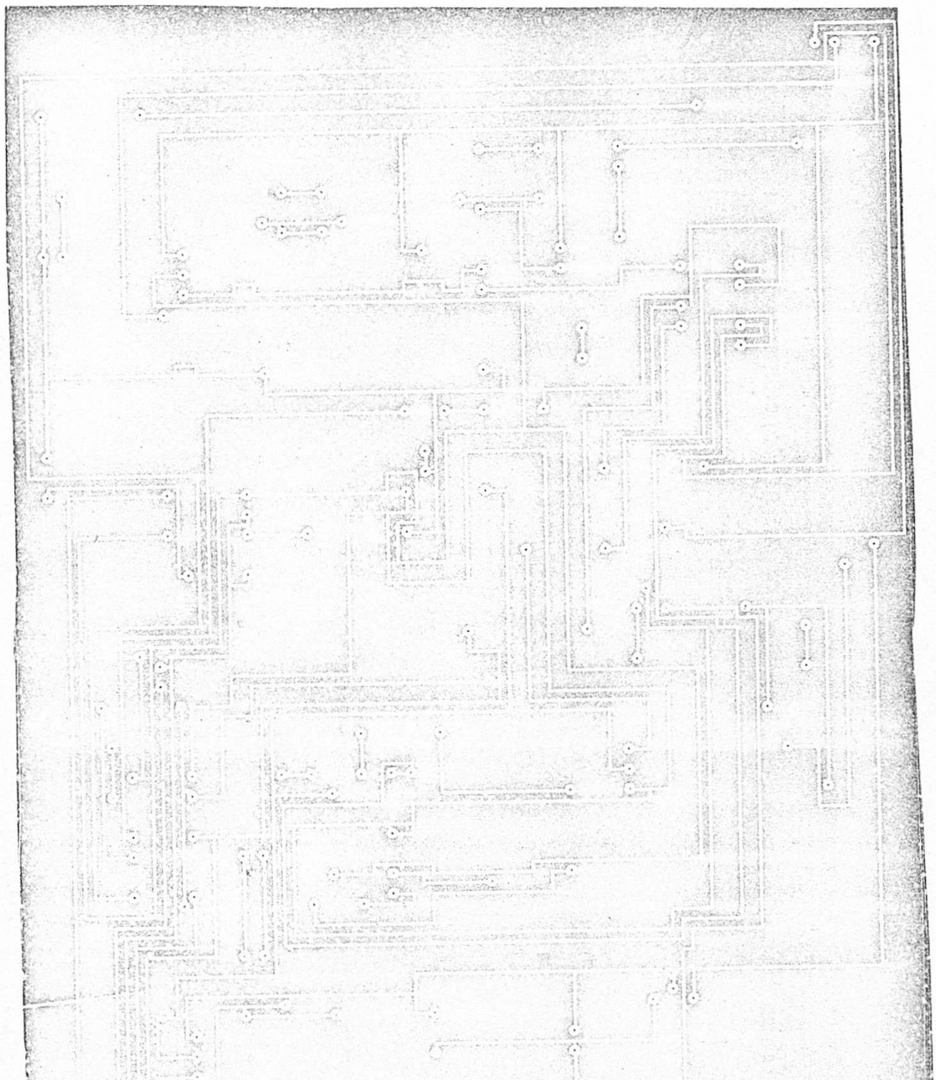
Dual Octave board, Side A



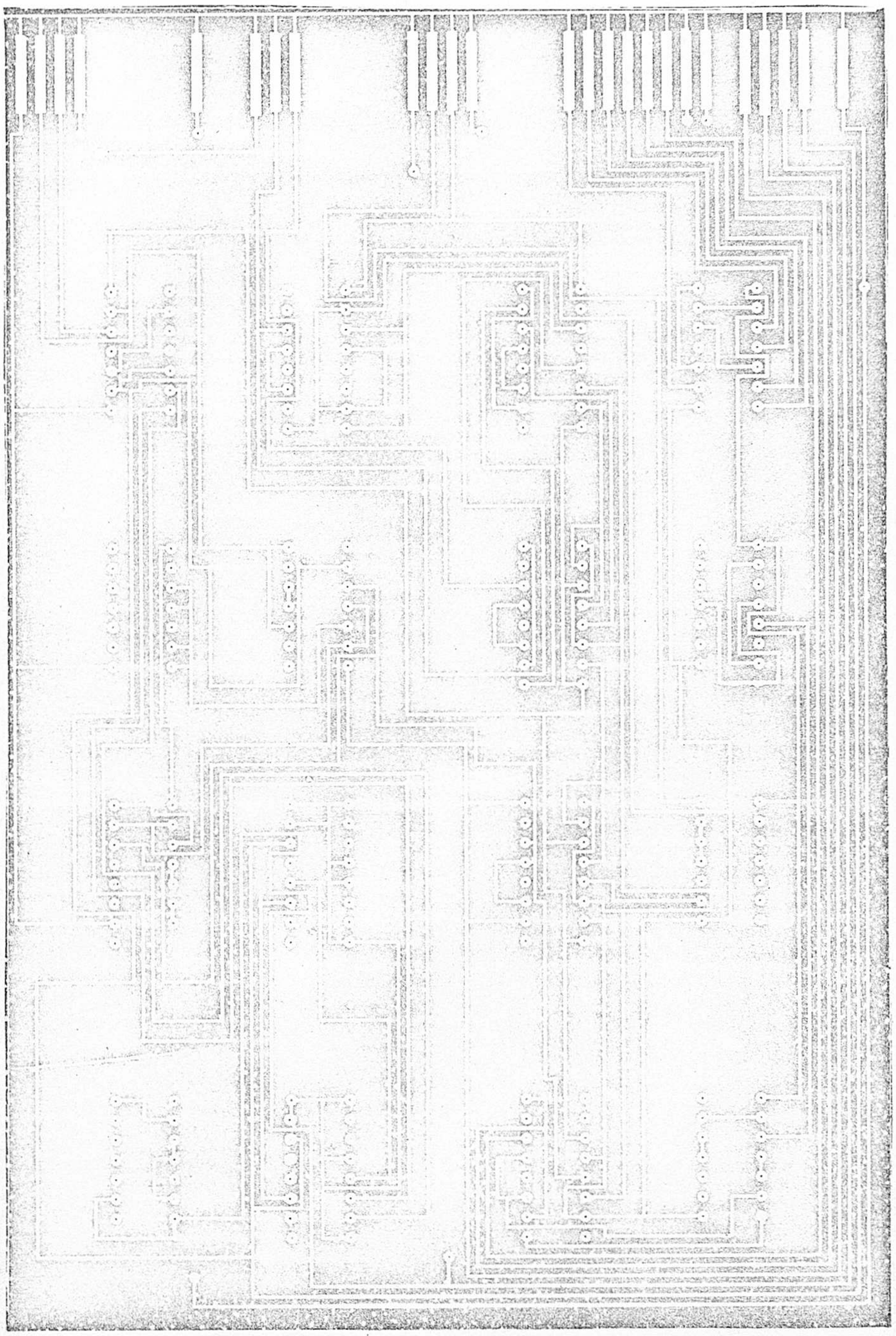
Dual Octave board, Side B



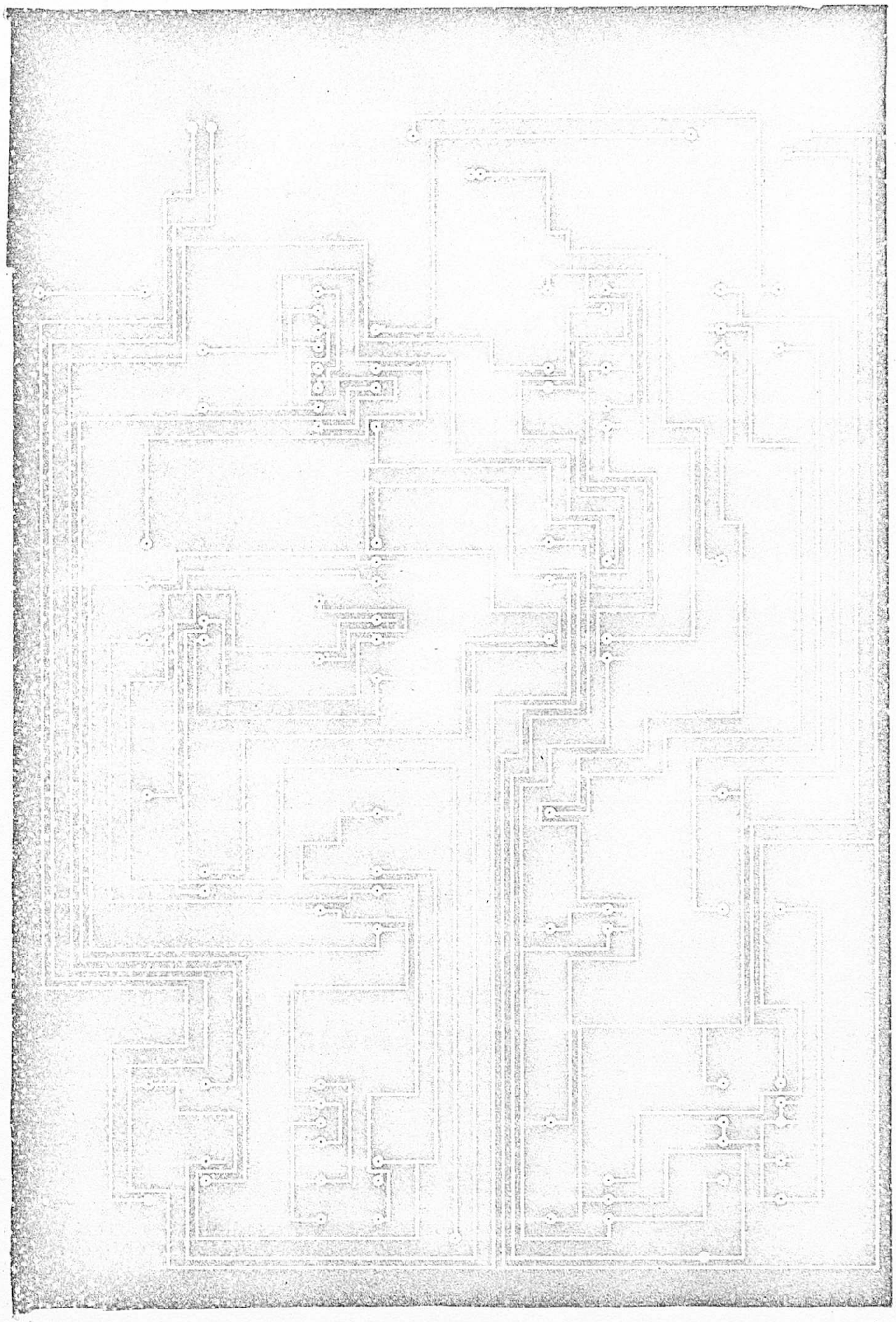
Termicette Interface and Master Clock, Side A



Termicette Interface and Master Clock, Side B



Ring Counter board, Side A



Ring Counter board, Side B

REFERENCES:

1. Piano Player Treasury
Harvey N Roehl. 1961.

2. Computer Music. Scientific American abstract.
Lejaren A. Hiller. December 1959.
pp.113 - 122.

3. Technology of Computer Music
M.V. Mathews.
Chapters 2,3.

4. Ph.D. Abstract
B. Styles Cambridge 1974.

5. Termicette Technical Manual ICP 011 - 103.
RACAL
Sections 2,3,4.

6. RS 232 Standard Interface Manual
RACAL August 1969.
Sections 2.3, 2.4.

7. Psychology of Music
Carl E. Seashore 1967
pp.233 - 248.

8. Accoustic Spectrometer
C.N. Hickman 1934.
pp.108 - 111.

9. Private Communication.
F.G. Heath.