SDC

TM-4652/400/00

THE LEXICON DESIGN FOR THE IBM 360/67

28 May 1971

21

DOCUMENT CONTROL DATA - R & D

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

1. ORIGINATING ACTIVITY (Corporate author) System Development Corporation Santa Monica, California	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE

The Lexicon Design for the IBM 360/67

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Technical Report -- Nov 1970 - May 1971

5. AUTHOR(S) (First name, middle initial, last name)

Iris Kameny

6. REPORT DATE 28 May 1971	7a. TOTAL NO. OF PAGES 15	7b. NO. OF REFS

8a. CONTRACT OR GRANT NO. DAHC15-67-C-0149	9a. ORIGINATOR'S REPORT NUMBER(S) TM-4652/400/00
b. PROJECT NO. ARPA Order #1327, Amendment #3, Program c. Code No. 1D30, and 1P10. d.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. DISTRIBUTION STATEMENT

Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY

13. ABSTRACT

This document provides a detailed description of the lexicon design implemented on the IBM 360/67 for the SDC version of the Vicens-Reddy Speech Recognition System.

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Voice I/O						

UNCLASSIFIED
Security Classification

SERIES	BASE NO. / VOL. / REISSUE
TM	4652 /400/00

TECHNICAL MEMORANDUM

(TM Series)

The work reported herein was supported by the Advanced Research Projects Agency of the Department of Defense under Contract DAHC15-67-C-0149, ARPA Order No. 1327, Amendment No. 3, Program Code No. 1D30, and 1P10.

THE LEXICON DESIGN FOR THE IBM 360/67

by

Iris Kameny

28 May 1971

SYSTEM
DEVELOPMENT
CORPORATION
2500 COLORADO AVE.
SANTA MONICA
CALIFORNIA
90406

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

Distribution of this document is unlimited.

ABSTRACT

This document provides a detailed description of the lexicon design implemented on the IBM 360/67 for the SDC version of the Vicens-Reddy Speech Recognition System.

TABLE OF CONTENTS

BLANK PAGE

1. INTRODUCTION

This document provides a detailed description of the lexicon design implemented on the IBM 360/67 computer for the SDC version of the Vicens-Reddy Speech Recognition System. The Vicens-Reddy System handled a maximum lexicon of 1,000 entries contained in 900,000 36-bit words of core. Not having this size core available under TS/DMS, and looking ahead to future lexicons in excess of 1,000 entries, we have designed a means of storing the lexicon on disc. This design will allow rapid access to a large lexicon.

The programming necessary for setting up a new lexicon, inserting lexicon samples, and retrieving those already inserted was accomplished by additions and modifications to the checked-out Vicens-Reddy Segmentation-Recognition System on the 360/67. This new system is referred to as CWIPER.* The entire system, except for one subroutine (the first and last character hash), is programmed in FORTRAN IV, Version G. CWIPER was compiled in three sections and linked through the TS/DMS link editor to run under TS/DMS on the 360/67.

CWIPER can select a group of possible candidates for a speech sample match. The mapping and evaluation routines necessary to select the best candidate will be implemented in the near future.

2. LEXICON DISC DESIGN

The lexicon is a TS/DMS S-1 file** residing on a 2314 disc pack. It is composed of fixed-length records, each 8,192 bytes long. The fixed-length record size is necessary for defining the lexicon as a direct-access data set under FORTRAN IV (G level). A record size of 8,192 bytes was chosen because it is the length of two core pages.

*Contextual Word In Phrase Extraction Routine

**A single-volume, sequential, variable-record-size file

The 2314 disc pack contains 203 physical cylinders of 20 tracks each. Under TS/DMS, each physical cylinder contains 4 logical cylinders of 5 tracks each. Three of the physical cylinders are used as alternatives should any tracks become defective. Physical cylinder Ø is used by the TS/DMS cataloger and contains information about files that have been created on the pack. The maximum space available to a user of a 2314 disc pack, then, is 199·4 logical cylinders, each containing 5 tracks.

The maximum number of lexicon entries will be limited to 65,535 $(2^{16}-1)$ entries. Each entry necessitates a 16-byte entry in the LXPART table and a variable-length entry in the LXALL table.

The number of LXPART entries per record is calculated as follows:

$$\frac{\text{8,192 bytes per record}}{\text{16 bytes per entry}} = \text{512 entries per record}$$

The number of disc records needed to contain a full LXPART is:

$$\frac{\text{65,535 total number of entries}}{\text{512 entries per record}} = \text{128 disc records}$$

The average length of a LXALL entry is 360 bytes. (The maximum length is 767 bytes.) The number of average LXALL entries per record is calculated to be:

$$\frac{\text{8,192 bytes per record}}{\text{360 bytes per entry}} = \text{22 entries per record}$$

The number of disc records needed to contain a full LXALL of average size entries is:

$$\frac{\text{65,535 total number of entries}}{\text{22 entries per record}} = \text{2,979 disc records}$$

The total number of bytes available on a 2314 disc is 199 physical cylinders · 4 logical cylinders · 5 tracks · 7,231 bytes per track = 28,779,380 bytes. The approximate number of bytes needed for 65,535 average lexicon entries is 27,000,000, which includes 164 bytes per record for SPAM usage and 184 bytes per logical cylinder for SPAM usage.

3. CORE LAYOUT FOR THE INUSE LEXICON

Three main core blocks are used for the lexicon. COMLEX is the lexicon control area, CORPT is the LXPART area, and CORALL is the LXALL area.

3.1 COMLEX

COMLEX is defined as a 2,048-word block located in COMMON/COMLX/. It contains all the control information for lexicon handling and is the first lexicon record on the disc. It contains the following tables and items in sequence:

VFHASH A table of 256 half-word integer pointers that make up the vowel-fricative hash table.

CHHASH A table of 676 half-word integer pointers that make up the first-last character hash table.

LXPPT A table of 128 half-word integer pointers that point to LXPART records.

LXUSE A table of 128 half-word integer indicators representing the core status of corresponding LXPPT entries. A non-zero LXUSE entry indicates the block of CORPT in which the LXPART record resides.

COREA A full-word integer containing the record number of the LXALL record currently in core at CORALL.

CORES A table of 16 half-word integer entries. Each entry represents a block of the CORPT table from 1 to NBUF (see Section 3.2) and is a pointer to the LXPPT and LXUSE table entries for the LXPART record currently in that block.

NREC A full-word integer containing the number of the next available record to be assigned to a new LXPART or LXALL record.

NLEXNO A full-word integer containing the next number for assignment as a lexicon entry number.

NALLRC A full-word integer containing the record number of the last incomplete LXALL record.

NALLWD A full-word integer containing the word position at which the next-to-be-added LXALL entry would begin in the NALLRC record.

NRFLAG A full-word logical that is a flag set in the LOOK subroutine and tested by the INSERT subroutine. If .TRUE. then a new LXALL record is needed; if .FALSE. then a current LXALL record (NALLRC) can be used.

CHTEMP A full-word integer containing the number of the first and last character hash of the print name of the current speech sample to be inserted in the lexicon.

VFTEMP A full-word integer containing the number of vowels-number of fricatives hash of the current speech sample.

3.2 CORPT

CORPT is defined as a NBUF·2,048-word integer table located in COMMON/COMCOR/ CORALL, CORPT. NBUF is defined as a full-word integer and is contained in the COMMON/COMDAT/ area of the Main program and all subroutines. It is currently set to 2 in BLOCK DATA. NBUF indicates the number of 2,048-word blocks available for the simultaneous loading of LXPART records. The minimum size of NBUF is 1 and the maximum size is 16. (The maximum is determined by the size of the CORES table; NBUF may exceed the maximum if the CORES table is increased correspondingly.)

CORPT2 is defined as an NBUF·4096 half-word table and is equivalent to CORPT. It is defined for ease in referencing half-word fields in FORTRAN.

3.3 CORALL

CORALL is defined as a 2,048-word integer table located in the LXALL storage area.

CORAL2 is defined as a 4,096-half-word integer table and is equivalent to CORALL. It is defined for ease in referencing half-word fields in FORTRAN.

CORAL4 is defined as an 8,192-byte LOGICAL*1 table and is equivalent to CORAL2 and CORALL. It is defined for ease in raferencing logical bytes in FORTRAN.

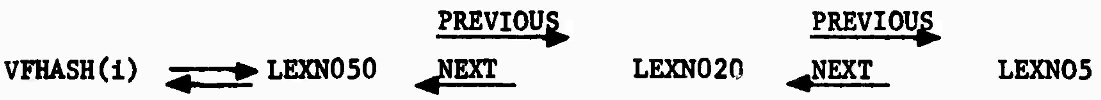
4. DETAILED DESCRIPTION OF THE LEXICON TABLES

4.1 VFHASH TABLE

VFHASH table is the vowel-fricative hash table. The table consists of 256 half-word integers. An entry address in this table is computed by:

16 · FEATURES MATRIX Vowel count + features matrix fricative count + 1.

The 16-bit entry (if non-zero) points to a four-word entry in the LXPART table by lexicon entry number. It is the lexicon entry number of the last entry having the same hash code. When a new lexicon entry is inserted, the lexicon entry number found in the VFHASH entry is used to reference its LXPART entry and change its next VFHASH pointer from zero (the VFHASH table) to NLEXNO (the lexicon number of the current sample to be inserted). The LXPART entry for the new sample (NLEXNO) will use the lexicon number found in the VFHASH entry as its previous VFHASH pointer and zero (the VFHASH table) as its next VFHASH pointer. The VFHASH table entry will be set to NLEXNO.



where: VFHASH(i) is 50.

The next VFHASH pointer for LEXNO50 is 0.

The previous VFHASH pointer for LEXNO50 is 20.

The next VFHASH pointer for LEXNO20 is 50.

The previous VFHASH pointer for LEXNO20 is 5.

The next VFHASH pointer for LEXNO5 is 20.

The previous VFHASH pointer for LEXNO5 is 0.

The vowel-fricative hash is based on a maximum of 15 vowels and 15 fricatives in a speech sample. The hash is based on the low order of bits, and a wrap-around occurs if there are more than 15 vowels or 15 fricatives in a speech sample.

VFHASH Table Layout

Entry Number	16·V + F + 1
1	16·0 + Ø + 1
2	16·0 + 1 + 1
3	16·0 + 2 + 1
4	16·0 + 3 + 1
5	16·0 + 4 + 1
6	16·0 + 5 + 1
7	16·0 + 6 + 1
8	16·0 + 7 + 1
9	16·0 + 8 + 1
10	16·0 + 9 + 1
11	16·0 + 10 + 1
12	16·0 + 11 + 1
13	16·0 + 12 + 1
14	16·0 + 13 + 1
15	16·0 + 14 + 1
16	16·0 + 15 + 1
17	16·1 + 0 + 1
⋮	
32	16·1 + 15 + 1
⋮	
48	16.2 + 15 + 1
⋮	
256	16·15 + 15 + 1

(where V is the features-matrix vowel count and F is the features-matrix fricative count)

4.2 CHHASH TABLE

CHHASH table is the first-last character hash of the print name (i.e., the first and last alphabetic characters). The table consists of 676 half-word integers. Each lexicon entry has a print-name area that can be a maximum of 255 characters long. A hash of the first and last letters yields an entry into the table: 26 · converted 1st letter + converted last letter + 1.

LETTER	EBCDIC REPRESENTATION	CONVERTED NUMBER
A	C1	0
B	C2	1
C	C3	2
D	C4	3
E	C5	4
F	C6	5
G	C7	6
H	C8	7
I	C9	8
J	D1	9
K	D2	10
L	D3	11
M	D4	12
N	D5	13
O	D6	14
P	D7	15
Q	D8	16
R	D9	17
S	E2	18
T	E3	19
U	E4	20
V	E5	21
W	E6	22
X	E7	23
Y	E8	24
Z	E9	25

The conversion rules are:

for:

C1 ≤ character ≤ C9, character = character -C1

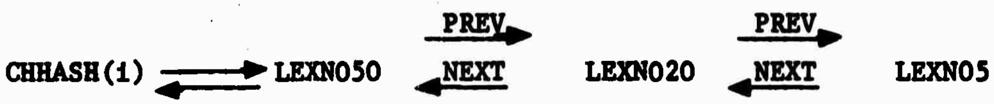
for:

D1 ≤ character ≤ D9, character = character -D1 + 9

for:

E2 ≤ character ≤ E9, character = character -E2 + 18

The 16-bit entry in the CHHASH table pointed to by the first and last letter hash is a pointer to a 4-word entry in the LXPART table (i.e., a lexicon entry number). It is the lexicon entry number of the last entry having the same hash code. When a new lexicon entry is inserted, the lexicon entry number found in the CHHASH entry is used to reference its LXPART entry and change its next CHHASH pointer from zero (the CHHASH table) to NLEXNO (the lexicon number of the current sample to be inserted). The LXPART entry for the new sample (NLEXNO) will use the lexicon number found in the CHHASH entry as its previous CHHASH pointer and zero (the CHHASH table) as its next CHHASH pointer. The CHHASH table entry will be set to NLEXNO.



where: CHHASH(i) is 50.

The next CHHASH pointer for LEXNO50 is 0.

The previous CHHASH pointer for LEXNO50 is 20.

The next CHHASH pointer for LEXNO20 is 50.

The previous CHHASH pointer for LEXNO20 is 5.

The next CHHASH pointer for LEXNO5 is 20.

The previous CHHASH pointer for LEXNO5 is 0.

CHHASH Table Layout

entry number	1st letter	last letter	26·first letter + last letter + 1
1	A	A	26·0 + 0 + 1
2	A	B	26·0 + 1 + 1
3	A	C	26·0 + 2 + 1
4	A	D	26·0 + 3 + 1
5	A	E	26·0 + 4 + 1
6	A	F	26·0 + 5 + 1
7	A	G	26·0 + 6 + 1
8	A	H	26·0 + 7 + 1
9	A	I	26·0 + 8 + 1
10	A	J	26·0 + 9 + 1
11	A	K	26·0 + 10 + 1
12	A	L	26·0 + 11 + 1
13	A	M	26·0 + 12 + 1
14	A	N	26·0 + 13 + 1
15	A	O	26·0 + 14 + 1
.	.	.	. . . .
.	.	.	. . . .
26	A	Z	26·0 + 25 + 1
52	B	Z	26·1 + 25 + 1
.	.	.	. . . .
.	.	.	. . . .
.	.	.	. . . .
676	Z	Z	26·25 + 25 + 1

4.3 LXPPT TABLE

LXPPT table is a table pointing to LXPART records. It contains 128 half-word pointers. Each pointer points to a LXPART record that contains a maximum of 512 four-word entries for a range of lexicon entry numbers.

For example: LXPPT(1) points to the first LXPART record, which can contain entries for lexicon numbers 1 through 512; LXPPT(2) points to the second LXPART record, which can contain entries for lexicon numbers 513 through 1,024; etc., for a total of 128 LXPPT entries given a maximum-size lexicon (65,535 entries).

If the lexicon is not large enough to fill all the LXPART records, then the LXPPT pointers to unused LXPART records will be zero.

The method of finding the correct LXPPT entry and, within the pointed-to record, the correct LXPART entry, is:

$$\text{PPT} = \text{Integer Part} \left(\frac{\text{lexicon number-1}}{512}\right) + 1 = \text{entry in LXPPT table}$$

$$\text{ENPPT} = \text{Fractional Part} \left(\frac{\text{lexicon number-1}}{512}\right) + 1 = \begin{array}{l}\text{the LXPART entry number in}\\ \text{the record pointed to by LXPPT}\end{array}$$

4.4 LXUSE TABLE

LXUSE table contains 128 half-word integer entries. Each entry corresponds to a LXPPT entry and indicates the core status of the respective LXPART record. LXUSE(i) = 0 ... 16; if it is not zero, then it contains the core-block number of CORPT that currently contains the LXPART record pointed to by LXPPT. LXUSE(i) is an index to CORES and vice versa:

CORES (LXUSE(i)) = i CORES(2) = 3
LXUSE (CORES(j)) = j LXUSE(3) = 2

Thus LXUSE(3) = 2 means that the third logical record of LXPART is currently in core beginning at CORPT (2049). If we are checking to see what core is available, CORES(2) = 3 tells us that the second block of CORPT is currently occupied by the third logical record of LXPART. The physical record number of the third logical record of LXPART is given by LXPPT(3).

4.5 LXPART TABLE

For each lexicon entry there is a 16-byte LXPART entry. The LXPART entry identifies the record number and starting byte of the larger LXALL entry. It also contains pointers to the previous and next entries with the same VFHASH, pointers to the previous and next entries with the same CHHASH, and the vowel-fricative map in binary that represents the lexicon entry.

16-byte entry

Pointer to previous entry with same CHHASH	Pointer to previous entry with same VFHASH
Pointer to next entry with same CHHASH	Pointer to next entry with same VFHASH
Vowel-Fricative Pattern Binary	
Record number of LXALL entry	Starting word of LXALL within record

The vowel-fricative pattern is right justified.

Each vowel = Ø1_2

Each fricative = 1Ø$_2$

A pattern--for example, of VFFVVF--is given in binary and hexadecimal as follows:

						V F	F V	V F
binary:	0000	0000	0000	0000	0000	0110	1001	0110
hexadecimal:	0	0	0	0	0	6	9	6

The purpose of the LXPART table is to be able to quickly accumulate a subset of lexicon entries that are possible matches for a particular speech sample. The vowel-fricative count of the speech sample is used to reference a VFHASH entry that (if non-zero) gives the lexicon entry number of the last lexicon entry having the same VFHASH code. This lexicon entry number is first used in referencing the LXPPT and LXUSE tables to be sure the needed LXPART record is in core. When the LXPART record is in core, the proper lexicon entry is found and its vowel-fricative pattern is matched with that of the speech sample. If it matches, then it is saved on the stack as a possible match. Regardless of a match, the LXPART pointer to the previous entry with the same VFHASH yields another lexicon entry number, etc. For a small lexicon (less than 1,024 entries), the entire LXPART table can be core resident.

4.6 LXALL TABLE

The LXALL entry is the compact features matrix of a speech sample, plus identification, lexicon entry number, print name in EBCDIC, number of characters in the print name, and some reserved space for later expansion (perhaps in the syntactic area).

The maximum size of a LXALL entry is:

General Information	32 bytes
8 bytes per row for maximum of 60 rows*	480 bytes
Print name, maximum 255 bytes	255
	767 bytes

*The maximum number of rows in the Vicens-Reddy Features Matrix (1968) is 60; however, this lexicon design would allow up to 255 (2^8-1) rows.

In estimating storage, the average size of 360 bytes was used. This was based on the Vicens-Reddy average of 90 words per lexicon entry.

4.7 THE LXALL ENTRY LAYOUT

<table>
<tr><td colspan="2">∅ = UNUSED FIELD</td><td>NO. of characters in print name</td><td colspan="2">LEXICON ENTRY NUMBER</td></tr>
<tr><td colspan="3">IDSESS = SESSION NUMBER</td><td colspan="2">SAMPLE = SAMPLE NUMBER</td></tr>
<tr><td colspan="3">MANNO = MAN NUMBER</td><td colspan="2">VERSN1 = Preprocessing Version No.</td></tr>
<tr><td colspan="3">VERSN2 = Segmentation Version No.</td><td colspan="2">VERSN3 = Recognition Version No.</td></tr>
<tr><td colspan="3">VERSN4 = Mapping Version No.</td><td colspan="2">∅ = Unused field</td></tr>
<tr><td colspan="3">Beg. Q Matrix Segment Number -1</td><td colspan="2">∅ = Unused field</td></tr>
<tr><td colspan="2">Vowel Count</td><td>Fricative Count</td><td>Row Count + 1</td><td>Vowel 1 Row No. + 1</td></tr>
<tr><td colspan="2">Vowel 2 Row No. + 1</td><td>Vowel 3 Row No. + 1</td><td>Vowel 4 Row No. + 1</td><td>Vowel 5 Row No. + 1</td></tr>
<tr><td>SXT_1</td><td>TYPE_1</td><td>DUR_1</td><td>$\mathrm{A1}_1$</td><td>$\mathrm{Z1}_1$</td></tr>
<tr><td colspan="2">$\mathrm{A2}_1$</td><td>$\mathrm{Z2}_1$</td><td>$\mathrm{A3}_1$</td><td>$\mathrm{A3}_1$</td></tr>
<tr><td colspan="5">. . . .
. Two Words for each row of the features matrix .
. . . .</td></tr>
<tr><td>SXT_i</td><td>TYPE_i</td><td>DUR_i</td><td>$\mathrm{A1}_i$</td><td>$\mathrm{Z1}_i$</td></tr>
<tr><td colspan="2">$\mathrm{A2}_i$</td><td>$\mathrm{Z2}_i$</td><td>$\mathrm{A3}_i$</td><td>$\mathrm{Z3}_i$</td></tr>
<tr><td colspan="5">PRINT NAME IN EBCDIC
FOR MAXIMUM OF 255 CHARACTERS</td></tr>
</table>

Where:

Row Count = $i + 1$

$SXT_i = 0_{16}$ if not a local minimum

$= 2_{16}$ if a local minimum (the local maximums were removed in the recognition process)

Note: If the lexicon entry has been entered by the user from the terminal and not from a live or recorded speech sample, the SESSNO = 0.

5. METHOD OF CHANGING LEXICON DIMENSIONS

COMMON/COMSET/MAXNO, MAXREC, MAXPTR, MAXWDR contains all the variables affecting the lexicon size and usage except for NBUF (the number of 2,048-word blocks available for the simultaneous loading of LXPART records). MAXNO is the maximum number of lexicon entries, MAXREC is the maximum number of records in the lexicon, MAXPTR is the maximum number of entries in a LXPART record, and MAXWDR is the maximum number of words in a LXPART or LXALL record. These are currently initialized in BLOCK DATA as follows:

NBUF = 2
MAXNO = 1024
MAXREC = 60
MAXPTR = 512
MAXWDR = 2048

The dimensions of the lexicon can be changed by redefining these five integer variables in the BLOCK DATA program. Remember, however, that when the new lexicon is opened (and then after), the size of the file parameter and the record size in bytes must correspond to the new definitions.

For example, the current lexicon is opened by:

LEXICON V41003 800000 F R8192

where: "LEXICON" is the file name,
"V41003" is the disc-pack designation,
"800000" is the byte-size of the file,
"F" for fixed-record size, and
"R8192" is the byte-size of the record.

6. INITIALIZING THE LEXICON

When CWIPER asks the user "NEW LEXICON?" and the response is "YES," it then asks for the new lexicon file description and opens up the new lexicon file. The COMLEX core area is cleared to zeros except for the following settings:

NLEXNO = 1 Lexicon entry number 1 will be the first assigned lexicon number.

NALLRC = 3 The current LXALL record will be record 3.

NALLWD = 1 The first LXALL entry will begin at word 1 of record 3.

COREA = 3 The LXALL record 3 is considered to be in core.

LXPPT(1) = 2 The first logical LXPART record is physical record 2.

LXUSE(1) = 1 The LXPART logical record 1 is in the first block of CORPT in core.

CORES(1) = 1 The first block of CORPT contains the physical record pointed to by LXPPT(1).

NREC = 4 The next record to be assigned for LXALL or LXPART will be physical record 4.

COMLEX is then written out as the first record of the new lexicon file.