United States Patent [19]

Park

[11] **3,852,720**

[45] **Dec. 3, 1974**

[54] **METHOD AND APPARATUS FOR AUTOMATICALLY GENERATING KOREAN CHARACTER FONTS**

[76] Inventor: **Hen Shu Park,** 20707 Anza Ave. No. 10, Torrance, Calif. 90505

[22] Filed: **Feb. 12, 1973**

[21] Appl. No.: **332,013**

[52] **U.S. Cl.** ..... **340/172.5,** 197/1 A
[51] **Int. Cl.** **B41b 9/00,** G06f 3/12
[58] **Field of Search** ..... 340/172.5, 146.3; 197/1 A, 197/1 R; 95/4.5 R; 101/93 C; 199/18, 38, 40

[56] **References Cited**

UNITED STATES PATENTS

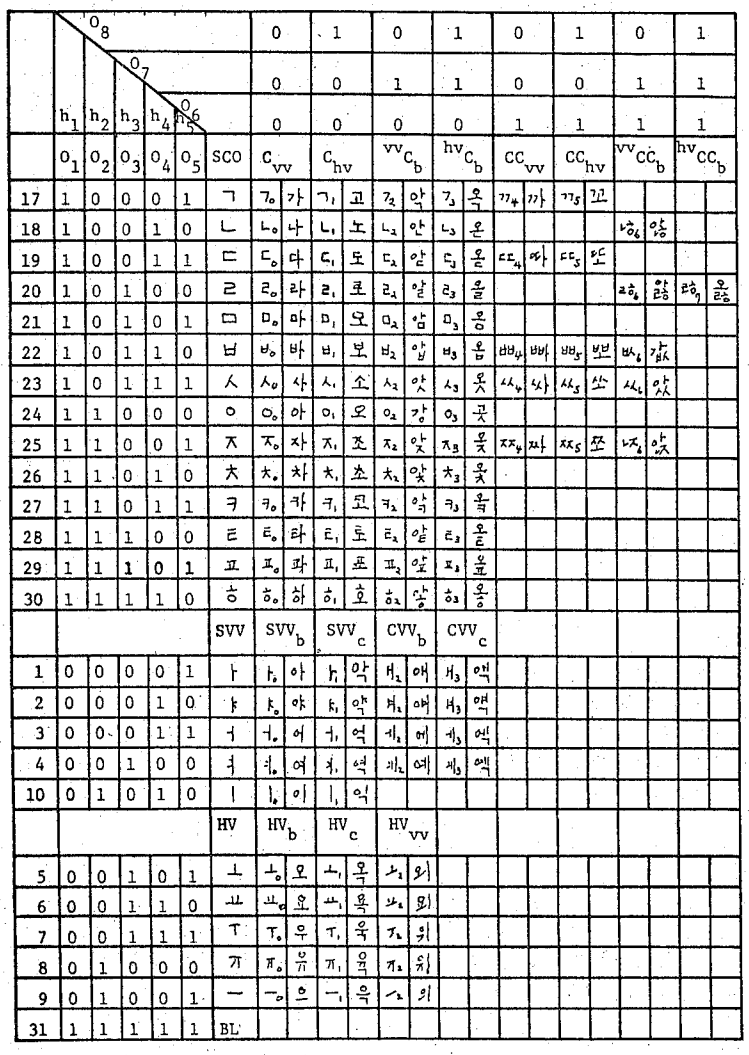
2,728,816	12/1955	Chung-Chin Kao	197/1 A
3,319,516	5/1967	Brown	197/1 A
3,325,786	6/1967	Shashoua et al.	340/172.5
3,422,419	1/1969	Mathews	340/172.5
3,432,844	3/1969	Winston	197/1 R
3,513,968	5/1970	Hanson	197/1 A

Primary Examiner—Gareth D. Shaw
Assistant Examiner—Paul R. Woods
Attorney, Agent, or Firm—Bacon & Thomas

[57] **ABSTRACT**

A method and apparatus for characterizing Korean characters into compositional structure groups so as to provide for the automatic selection of individual symbol fonts used to formulate the various characters. A unique code is provided for each of the basic Korean symbols and logic circuitry is utilized to recognize basic compositional structure groups to provide a supplemental code for each symbol forming the Korean character. The supplemental code identifies the desired font.

9 Claims, 7 Drawing Figures



Korean Alphabet

Vowels	ㅏ	ㅑ	ㅓ	ㅕ	ㅗ	ㅛ	ㅜ	ㅠ	ㅡ	ㅣ										
Consonants	ㄱ	ㄴ	ㄷ	ㄹ	ㅁ	ㅂ	ㅅ	ㅇ	ㅈ	ㅊ	ㅋ	ㅌ	ㅍ	ㅎ						

Fig. 1.

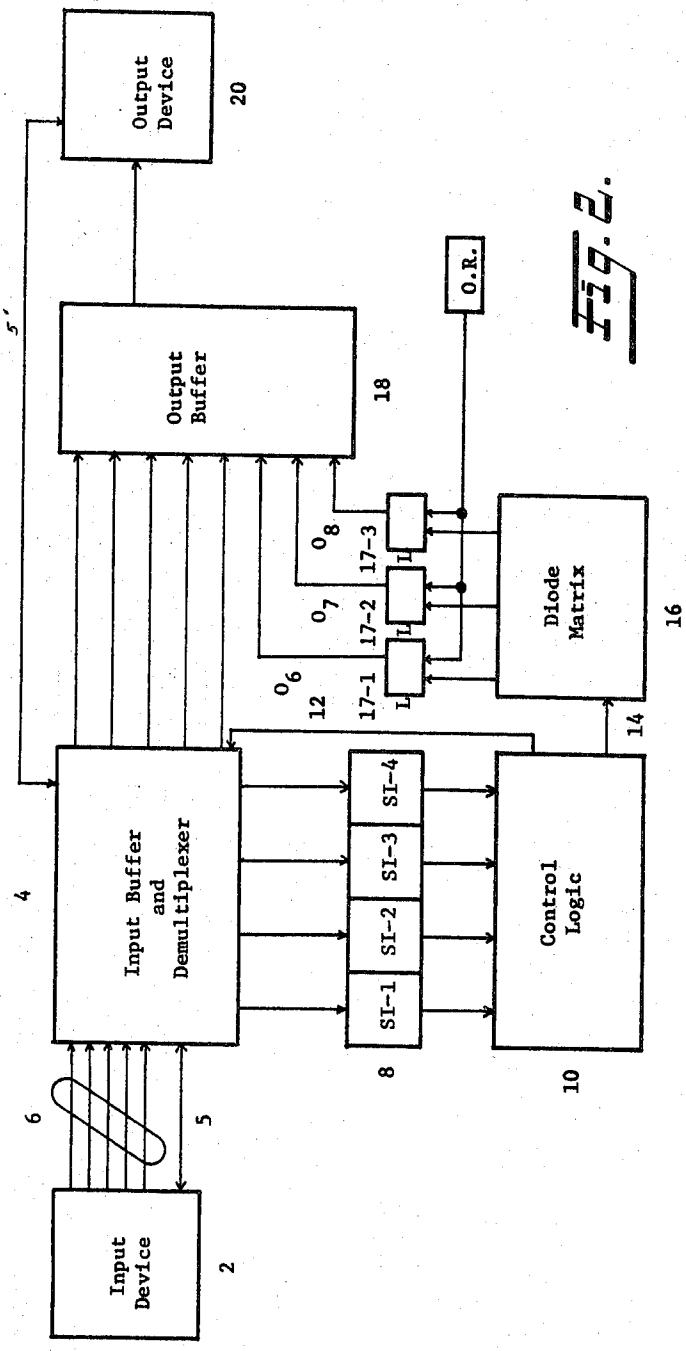


Fig. 2.

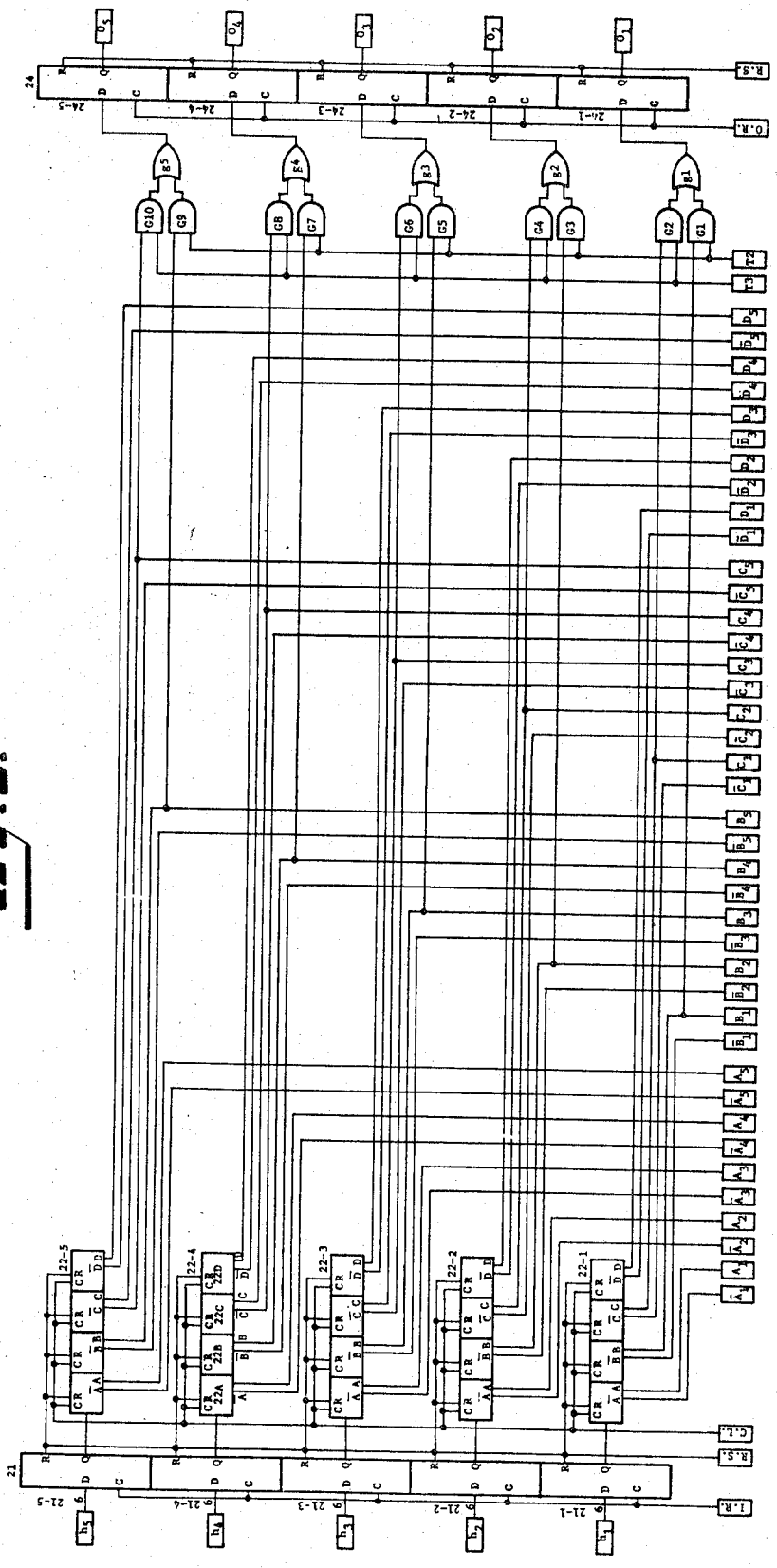
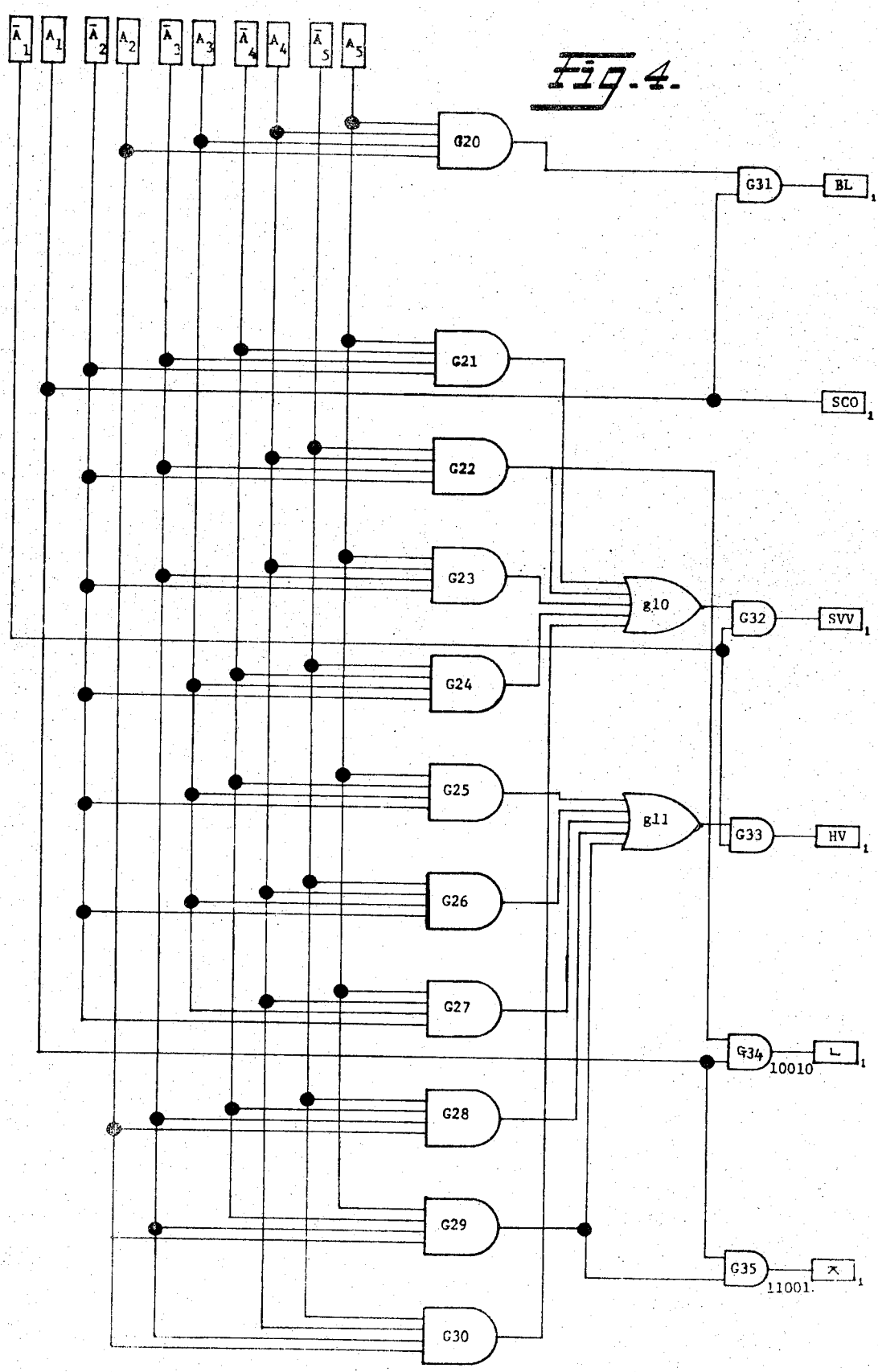


Fig. 3.

Fig. 4.


$\bar{A}_1$
A_1
$\bar{A}_2$
A_2
$\bar{A}_3$
A_3
$\bar{A}_4$
A_4
$\bar{A}_5$
A_5
G20
G21
G22
G23
G24
G25
G26
G27
G28
G29
G30
g10
g11
G31
G32
G33
G34
G35
BL
SCO
SVV
HV
L
10010
11001

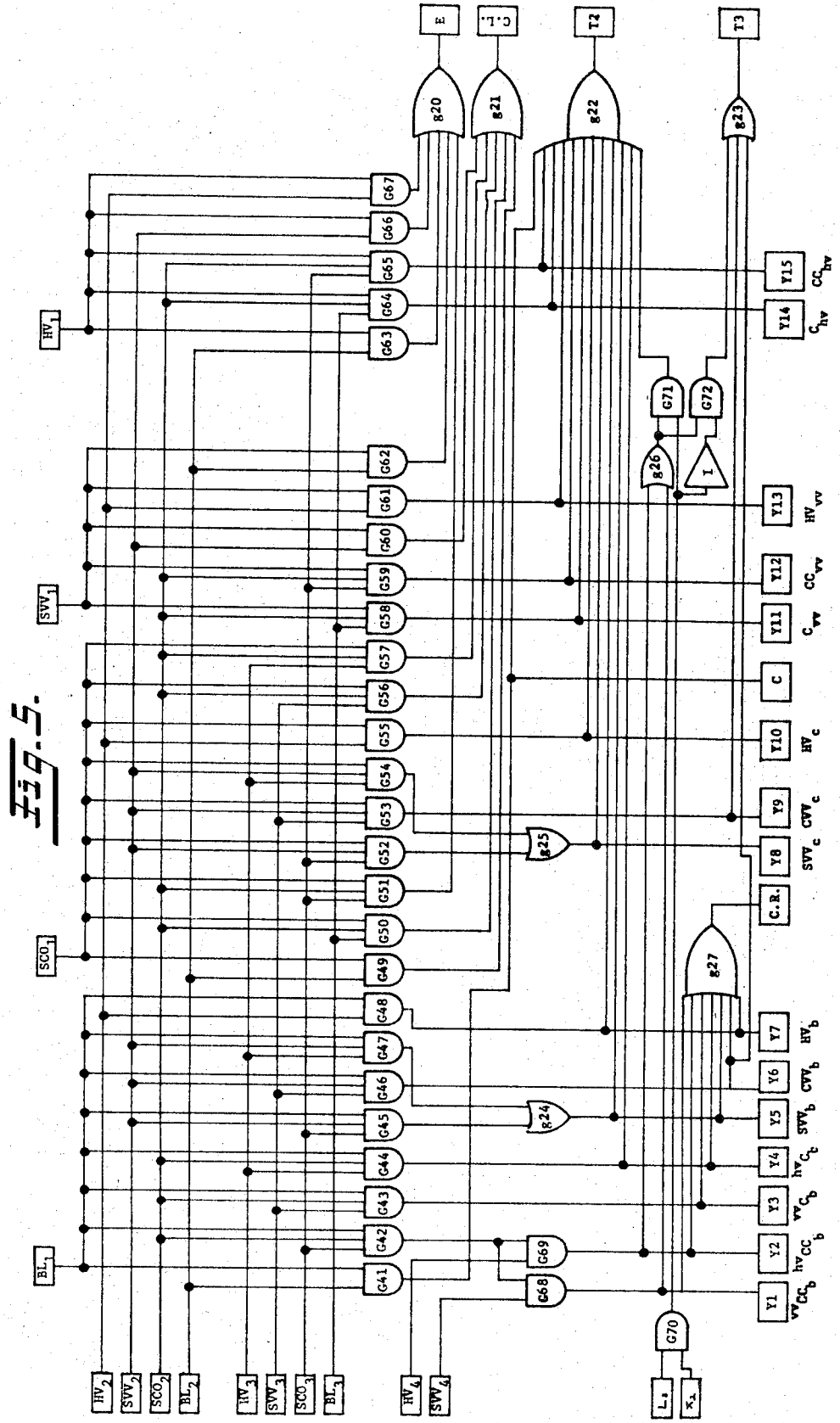

E
C.L.
T2
T3
g20
g21
g22
g23
G67
G66
G65
G64
G63
G62
G61
G60
G59
G58
G57
G56
G55
G54
G53
G52
G51
G50
G49
G48
G47
G46
G45
G44
G43
G42
G41
HV1
SVV1
SCO1
BL1
HV2
SVV2
SCO2
BL2
HV3
SVV3
SCO3
BL3
HV4
SVV4
G69
G68
G70
L1
X1
G71
G72
g26
1
Y15
Y14
Y13
Y12
Y11
C
Y10
Y9
Y8
C.R.
Y7
Y6
Y5
Y4
Y3
Y2
Y1
g25
g24
g27

FIG. 5.

Fig. 6.

	O_8						0		1		0		1		0		1		0		1	
					O_7		0		0		1		1		0		0		1		1	
	h_1	h_2	h_3	h_4	h_5	O_6	0		0		0		0		1		1		1		1	
	O_1	O_2	O_3	O_4	O_5	SCO	C_{vv}		C_{hv}		$^{vv}C_b$		$^{hv}C_b$		CC_{vv}		CC_{hv}		$^{vv}CC_b$		$^{hv}CC_b$	
17	1	0	0	0	1	ㄱ	ㄱ$_0$	가	ㄱ$_1$	고	ㄱ$_2$	악	ㄱ$_3$	옥	ㄲ$_4$	까	ㄲ$_5$	꼬				
18	1	0	0	1	0	ㄴ	ㄴ$_0$	나	ㄴ$_1$	노	ㄴ$_2$	안	ㄴ$_3$	온					ㄴㅎ$_6$	않		
19	1	0	0	1	1	ㄷ	ㄷ$_0$	다	ㄷ$_1$	도	ㄷ$_2$	알	ㄷ$_3$	올	ㄸ$_4$	빠	ㄸ$_5$	뽀				
20	1	0	1	0	0	ㄹ	ㄹ$_0$	라	ㄹ$_1$	로	ㄹ$_2$	알	ㄹ$_3$	올					ㄹㅎ$_6$	랗	ㄹㅎ$_7$	오랗
21	1	0	1	0	1	ㅁ	ㅁ$_0$	마	ㅁ$_1$	오	ㅁ$_2$	암	ㅁ$_3$	옹								
22	1	0	1	1	0	ㅂ	ㅂ$_0$	바	ㅂ$_1$	보	ㅂ$_2$	압	ㅂ$_3$	옵	ㅃ$_4$	빠	ㅃ$_5$	뽀	ㅂㅂ$_6$	밟		
23	1	0	1	1	1	ㅅ	ㅅ$_0$	사	ㅅ$_1$	소	ㅅ$_2$	앗	ㅅ$_3$	옷	ㅆ$_4$	싸	ㅆ$_5$	쏘	ㅆ$_6$	았		
24	1	1	0	0	0	ㅇ	ㅇ$_0$	아	ㅇ$_1$	오	ㅇ$_2$	강	ㅇ$_3$	공								
25	1	1	0	0	1	ㅈ	ㅈ$_0$	자	ㅈ$_1$	조	ㅈ$_2$	앚	ㅈ$_3$	옻	ㅉ$_4$	짜	ㅉ$_5$	쪼	ㅈ$_6$	앉		
26	1	1	0	1	0	ㅊ	ㅊ$_0$	차	ㅊ$_1$	초	ㅊ$_2$	앚	ㅊ$_3$	옻								
27	1	1	0	1	1	ㅋ	ㅋ$_0$	카	ㅋ$_1$	코	ㅋ$_2$	앜	ㅋ$_3$	옼								
28	1	1	1	0	0	ㅌ	ㅌ$_0$	타	ㅌ$_1$	토	ㅌ$_2$	알	ㅌ$_3$	올								
29	1	1	1	0	1	ㅍ	ㅍ$_0$	파	ㅍ$_1$	포	ㅍ$_2$	앞	ㅍ$_3$	옾								
30	1	1	1	1	0	ㅎ	ㅎ$_0$	하	ㅎ$_1$	호	ㅎ$_2$	앟	ㅎ$_3$	옹								
						SVV	SVV_b		SVV_c		CVV_b		CVV_c									
1	0	0	0	0	1	ㅏ	ㅏ$_0$	아	ㅏ$_1$	악	ㅐ$_2$	애	ㅐ$_3$	액								
2	0	0	0	1	0	ㅑ	ㅑ$_0$	야	ㅑ$_1$	약	ㅒ$_2$	얘	ㅒ$_3$	얙								
3	0	0	0	1	1	ㅓ	ㅓ$_0$	어	ㅓ$_1$	억	ㅔ$_2$	헤	ㅔ$_3$	엑								
4	0	0	1	0	0	ㅕ	ㅕ$_0$	여	ㅕ$_1$	역	ㅖ$_2$	예	ㅖ$_3$	옉								
10	0	1	0	1	0	ㅣ	ㅣ$_0$	이	ㅣ$_1$	익												
						HV	HV_b		HV_c		HV_{vv}											
5	0	0	1	0	1	ㅗ	ㅗ$_0$	오	ㅗ$_1$	옥	ㅘ$_2$	와										
6	0	0	1	1	0	ㅛ	ㅛ$_0$	요	ㅛ$_1$	욕	ㅙ$_2$	뫠										
7	0	0	1	1	1	ㅜ	ㅜ$_0$	우	ㅜ$_1$	욱	ㅝ$_2$	워										
8	0	1	0	0	0	ㅠ	ㅠ$_0$	유	ㅠ$_1$	육	ㅞ$_2$	웨										
9	0	1	0	0	1	ㅡ	ㅡ$_0$	으	ㅡ$_1$	윽	ㅢ$_2$	의										
31	1	1	1	1	1	BL																

Fig. 7.

	h_2	h_3	h_4	h_5	h_1 1	h_1 0
					SCO	VV or HV
1	0	0	0	1	ㄱ	ㅏ
2	0	0	1	0	ㄴ	ㅐ
3	0	0	1	1	ㄷ	ㅑ
4	0	1	0	0	ㄹ	ㅒ
5	0	1	0	1	ㅁ	ㅓ
6	0	1	1	0	ㅂ	ㅔ
7	0	1	1	1	ㅅ	ㅕ
8	1	0	0	0	ㅇ	ㅖ
9	1	0	0	1	ㅈ	ㅗ
10	1	0	1	0	ㅊ	ㅘ
11	1	0	1	1	ㅋ	ㅜ
12	1	1	0	0	ㅌ	ㅝ
13	1	1	0	1	ㅍ	ㅡ
14	1	1	1	0	ㅎ	ㅣ

METHOD AND APPARATUS FOR AUTOMATICALLY GENERATING KOREAN CHARACTER FONTS

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention pertains to the automatic characterization and classification of the various Korean symbols within a given character to uniquely select appropriate symbol fonts for each of the basic symbols within the character.

2. Description of the Prior Art

In the written Korean language, each individual character represents a syllable in the spoken language and consists of a combination of several basic symbols from an alphabet of 24 such basic symbols. Each of these basic symbols, however, has several possible fonts or variations. The font used for a particular symbol depends on the position of that symbol within the character and the graphic form of the other symbols comprising the character. For example, the basic symbol "ㄱ," a consonant, has six different types fonts:

a. ㄱ as in 가
b. ㄱ as in 고
c. ㄱ as in 궁
d. ㄱ as in 악
e. ㄱ as in 욱
f. ㄱ as in 괴

Different fonts may be used even for a symbol having the same syntactical meaning within different characters. For example, the symbol "ㄱ" has the same syntactical significance in examples *a*, *b* and *c* above, but different fonts are used. The general rule which determines the particular font of a symbol utilized in a particular situation is based on achieving a composite character having a "balancing appearance." The selection of fonts to utilize within a given character is accomplished by those skilled in the Korean language, and although the basic goal of achieving an overall balance or symmetry for each character has long been realized, no formal rules have been developed for classifying the basic Korean symbols so as to achieve an automated selection of type fonts.

Such an automated classification and selection scheme would be highly desirable since the total number of symbols required in printing or typing Korean characters is much larger than the English set of alphabet letters which have only two fonts for each letter namely, upper case and lower case. As a result, there are considerable difficulties in applying mechanized printing techniques for Korean characters in such devices to photo type-set, line printers or electronic typewriters.

Present day techniques for selecting the proper Korean font for each symbol are primarily manual. These methods necessitate large keyboards usually having over 50 symbols, and require large amounts of training and skill in operation.

Some automated attempts have been made to formulate Chinese characters from a relatively small number of basic symbols. For example, Shashoua et al. describes in U.S. Pat. No. 3,325,786, a Chinese ideograph machine which produces a binary code for each of a plurality of basic strokes. The code is compared to a large number of binary words stored in memory. An optical matrix is then employed to display selected ideographs. The system then utilizes optical scanning for recording the selected image of the optical scanning matrix onto film for later composition into printed form.

The Leban U.S. Pat. No. 3,665,450 shows an encoding and decoding apparatus for Chinese ideograph which also employs an optical printing system together with a complex keying procedure. This procedure necessitates manual classification of characters by the operator as they are keyed onto the system.

The Chinese automated systems utilize a somewhat arbitrary set of basic symbols in formulating the Chinese characters and the symbols have no syntactical significance in and of themselves.

SUMMARY OF THE INVENTION

It is an object of the instant invention to provide a method and apparatus for automatically selecting the proper font for each symbol used in formulating Korean characters. The basic symbols represent the Korean alphabet and have syntactical significance. The invention comprises a logic module which may readily be interfaced between the input and output of an I/O device such as a printing device. The logic module automatically composes the character output in accordance with the logical rules governing the position and type font of the basic symbols comprising the character.

In a broader sense the object of the instant invention is to achieve a compositional classification scheme for groups of basic Korean symbols that permits the selection of type fonts for each basic symbol.

A further object of the instant invention is to provide a relatively simple classification of Korean characters utilizing basic grammar theory.

A further object of the instant invention is to achieve a balanced composite character appearance for Korean characters by selecting appropriate symbol fonts utilizing the basic grammar theory classification.

Another object of the invention is to identify a geometrical structure of each basic Korean symbol and utilize this individual geometrical classification to formulate a composite classification scheme in constructing Korean symbol fonts for Korean characters.

Yet another object of the invention is to provide binary coded signals for each Korean symbol and its identifying type font for use in automatic printing or typing apparatus

BRIEF DESCRIPTION OF THE DRAWINGS

Thesee and other objects of the invention will become clear from the following description of the invention as illustrated by the drawings wherein:

FIG 1 illustrates the 24 basic symbols of the Korean alphabet;

FIG. 2 illustrates a block diagram of the character composer;

FIG. 3 is a schematic diagram of the input buffer and demultiplexer;

FIG. 4 is a schematic diagram of the syntax identifier;

FIG. 5 is a schematic diagram of the control logic;

FIG. 6 is illustrates Table 1 showing the input and output coding; and

FIG. 7 illustrates Table 2 showing a variant of the input coding of Table 1.

DESCRIPTION OF THE PREFERRED EMBODIMENT

Korean Characters — Introduction

The construction of Korean characters from a serial string of basic symbols is described in terms of the formal grammar theory of Chomsky, e.g., N. Chomsky "Formal Properties of Grammar," *Handbook of Mathematical Psychology*, Vol 2, John Wiley and Sons, 1963. In the present system, all of the various type fonts are classified, and the basic grammar theory of Chomsky is applied to these classification groups in the construction of Korean characters for typing or printing.

In order to describe the set of syntax rules governing the construction of a character, the notation of the Backus-Naur Form is adopted, see for example, "The Syntax and Semantics of the Proposed International Algebraic Languages," *UNESCO*, Conference on Information Processing, Paris 1959. This notation is a formal metalanguage used for describing the grammar structure of a language. The following symbols and interpretations are employed:

<X> The object named "X," treated as a unit

:: = "is defined as" or "is formed from"

| "or" (the exclusive "or") this symbol separates alternative definitions or ways of forming the object named on the left-hand side of the :: =.

In the Korean language a word may be composed of a single syllable, or may consist of several syllables. One syllable is written as a character. The syllables may be composed of three elemental phonemes which are the "initial," the "medial" and the "final." The description below applies the the formal language theory to forming a character which transcribes a single syllable.

Since the syllables may be decomposed into three phonemes, the initial (I), medial (M) and final (F), and since a single syllable is transcribed by a character (C), a permissive replacement rule can be established as follows:

$$\langle C\rangle :: = \langle SC\rangle \mid \langle SC\rangle \langle F\rangle \tag{1}$$

In the equation above the metalinguistic element "simple character" (SC) is introduced for the convenience of showing coarse structure of derivations. The simple character is again produced by the following rule:

$$\langle SC\rangle :: = \langle I\rangle \langle M\rangle \tag{2}$$

In addition, the non-terminal vocabularies initial, medial and final are themselves derived from the production rules:

$$\langle I\rangle :: = \langle Consonant\rangle$$
$$\langle F\rangle :: = \langle Consonant\rangle$$
$$\langle M\rangle :: = \langle Vowel\rangle \tag{3}$$

In the expression (3), the consonants and vowels consist of several basic symbols of the Korean alphabet and there are 24 such basic symbols as are illustrated in FIG. **1**. As may be seen in FIG. **1**, the basic 24 symbols are divided into two groups called the simple consonants (SCO) and the simple vowels (SV).

Graphic Structure Classification

In addition to the rather simple classification illustrated in FIG **1**, the simple vowels may be further classified according to the graphic structure of the symbols. Thus, the symbols with a long vertical stroke such as

ㅏ , ㅑ , ㅓ , ㅕ , ㅣ

will be called simple vertical vowels (SVV), and the symbols with a long horizontal stroke such as

ㅗ , ㅛ , ㅜ , ㅠ , ㅡ

are termed horizontal vowels (HV). Furthermore, each of the classes SVV and HV is again divided into two classes. A simple vertical vowel with the tip marks on its right-hand side is called a simple right vertical vowel (SRVV), namely ㅏ, ㅑ ; the SVV with the tip marks on its left-hand side is termed a simple left vertical vowel (SLVV), namely ㅓ , ㅕ . In addition, a HV with the tip marks upward from the horizontal line is termed an up-horizontal vowel (UHV), namely ㅗ, ㅛ ; and the HV with the tip marks downward from the horizontal line are called down-horizontal vowels (DHV), namely ㅜ, ㅠ .

The basic symbol " ㅣ " is called a simple vertical line vowel (SVLV), and the symbol " ㅡ " is termed a horizontal line vowel (HLV).

Thus, according to the above graphic classification scheme the SV is described by the expressions:

$$\langle SV\rangle :: = \langle SVV\rangle \mid \langle HV\rangle$$
$$\langle SVV\rangle :: = \langle SRVV\rangle \mid \langle SLVV\rangle \mid \langle SVLV\rangle \tag{4}$$
$$\langle HV\rangle :: = \langle UHV\rangle \mid \langle DHV\rangle \mid \langle HLV\rangle$$

Compositional Structure Classification — Complex Symbols

A Korean consonant or vowel may consist of several basic symbols taken from the symbols shown in FIG. **1**. However, each of the vowel symbols must belong to one of the classes described by the expression (4) above.

The classification given by the expressions (4) are based on the graphic structure of the basic vowel symbols. In order to describe a Korean character, however, a compositional character structure should be utilized in a classification scheme.

In the class of consonants that are actually used in a Korean character, there are those such as

ㄲ, ㄳ, ㄵ, ㄶ, ㄸ, ㄺ, ㄻ, ㄼ ,

other than simple consonants, and such symbols are termed compound consonants (CCO) which are generated by the expression

$$\langle CCO\rangle :: = \langle SCO\rangle \langle SCO\rangle \tag{5}$$

Then, the class <consonant> in the expression (1) and (2) is completely described by the expression

$$\langle Consonant\rangle :: = \langle SCO\rangle \mid \langle CCO\rangle \tag{6}$$

The expression (6) generates a larger class of consonants than that actually used in a Korean character, but

the expression does suffice to describe all the possible consonants used in any given Korean character.

In the class of vowels that are used in a Korean character, the compositional classifications are also possible and there are two classes which are termed a complex vertical vowel (CVV) and compound vowel (CV).

The CVV is again divided into two classes which are called complex right vertical vowel (CRVV) namely, ㅐ,ㅒ ; and, complex left vertical vowel (CLVV), namely ㅔ, ㅖ. Each of these two classes is composed of the 10 basic vowel symbols. The classification of the CVV is described by the expressions:

<CVV> ::= <CRVV> | <CLVV> (7)
<CRVV> ::= <SRVV> (8)
<CLVV> ::= <SLVV> (9)

The CV is also divided into two classes as shown in the expression:

$$<CV> :: = <CSV> \mid <CCV> \quad (10)$$

where the CSV means the compound simple vowel, and the CCV means the compound-complex vowel. Each of these classes is described by the expressions:

<CSV> ::= <HV> <SVV> (11)
<CCV> ::= <HV> <CVV> (12)

Combining the expressions for SVV, CVV and HV, and considering only the classes of the vowels that are used in a Korean character, the classes CSV and CCV are further classified by the expressions:

<CSV> ::= <CSURV> | <CSDLV> | <CSLV>
<CSLUV> | <CSLDV> (13)
<CSURV> ::= <UHV> <SRVV> (14)
<CSDLV> ::= <DHV> <SLVV> (15)
<CSLUV> ::= <UHV> <SVLV> (16)
<CSLDV> ::= <DHV> <SVLV> (17)
<CSLV> ::= <HLV> <SVLV> (18)

where the CSURV means the complex simple upright vowel such as "ㅚ" the CSDLV is the compound simple down left vowel such as "ㅜㅣ"; the CSLV the compound simple line vowel such as "ㅡ"; the CSLUV the compound simple line up vowel such as "ㅣ"; the CSDLV the compound simple line down vowel such as "ㅗㅣ."

Similarly for the CCV,

<CCV> ::= <CCURV> | <CCDLV> (19)
<CCURV> ::= <UHV> <CRVV> (20)
<CCDLV> ::= <DHV> <CLVV> (21)

where the CCURV means the compound complex up right vowel such as "ㅙ"; and, the CCDLV is the compound complex down left vowel such as "ㅝㅣ".

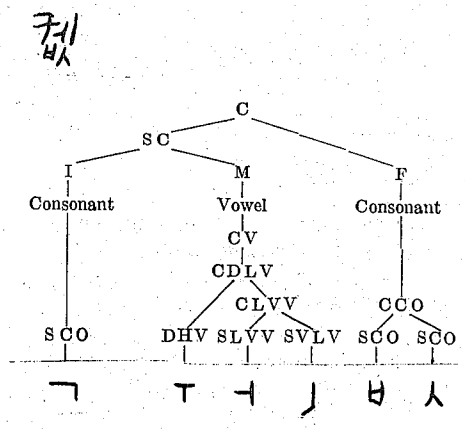
Therefore, all the vowels must belong to one of the classes HV, SVV, CVV and CV. From an observation of the classes HV, SVV, CVV and CV in the directional sense, the class HV is horizontal, the classes SVV and CVV are vertical, and the class CV is neither horizontal nor vertical and is in effect non-directional. Thus, in the directional sense another classification is possible, defined by the expression

$$<VV> :: = <SVV> \mid <CVV> \quad (22)$$

where the VV means the vertical vowel and the class "vowel" in expression (3) is described by the expression

$$<Vowel> :: = <HV> \mid <VV> \mid <CV> \quad (23)$$

Utilizing the above classification scheme, one may readily characterize any given Korean character. An example is shown below.



Type Font Classification Scheme

In printing a given Korean character there is a specific type font for each of the possible Korean symbols used to formulate a character. The selection of the particular font is made on the basis of achieving a "balanced appearance" of the character which attempts to formulate a line of characters which are aligned on a straight line having a uniform character height and width.

Therefor, the set of the Korean fonts for writing from left to right (horizontal writing) is different from the set of fonts for writing from top to bottom (vertical writing). However, since the basic symbols of the Korean alphabet have a graphical structure in both the horizontal and vertical directions, a compositional structure scheme for the characters may be employed in that the balanced appearance depends a great deal on the graphical character structure. Thus, a type font for a given character symbol can be constructed from the class of type fonts based on the compositional structure of characters. The type fonts selected may thus be used to construct characters for horizontal writing as well as vertical writing.

If one wanted to have a complete Korean font type apparatus, no grouping of the basic 24 symbols would be possible since each distinct arrangement of symbols, (seven such symbols for the most complex character) would necessitate a different font for each symbol. In practice, however, many basic symbol combinations are not used and each character itself must obey certain grammar rules with regard to the number of vowels and consonants and their relative position. Thus, for a practical system one can utilize the above defined geometrical classifications as basic building blocks in for-

mulating a compositional classification for a character.

In the following compositional structure classification scheme the geometrical and grammatical classifications SCO, VV, CCO, DHU, CV, UHV, HV, SVV, CVV are utilized. It is to be understood, however, that additional compositional structure classification schemes may be developed employing a more detailed geometrical classification which would also utilize the classifications SRVV, SLVV, SVLV, HLV etc. Although each expanded classification schemes are possible, the practical advantages are small when compared to the cost of implementation and actual font type usage within the Korean language.

The type fonts classes in the compositional structure classification are defined below:

1) C_{srr} — for <SCO>
<C> : : = <SCO> (<SVV>|<SVV><F>)
2) CC_{srr} — for <CCO>
<C> : : = <CCO> (<SVV>|<SVV><F>)
3) C_{crr} — for <SCO>
<C> : : = <SCO> (<CVV>|<CVV><F>)
4) CC_{crr} — for <CCO>
<C> : : = <CCO> (<CVV>|<CVV><F>)
5) C_{dhr} — for <SCO>
<C> : : = <SCO> (<DHV>|<UHV><F>) or
<C> : : = <SCO> <DHV> <F>
6) CC_{dhr} — for <CCO>
<C> : : = <CCO> (<DHV> <UHV><F>) or
<C> : : = <CCO> <DHV> <F>
7) C_{uhr} — for <SCO>
<C> : : = <SCO> <UHV>
8) CC_{uhr} — for <CCO>
<C> : : > <CCO> <UHV>
9) C_{hrr} — for <SCO>
<C> : : = <SCO> (<HV><VV>|
<HV><VV><F>)
10) CC_{hrr} — for <CCO>
<C> : : = <CCO> (<HV><VV>|<HV>
<VV><F>)
11) SVV_b — for <SVV>
<C> : : = <I> <SVV>
12) SVV_c — for <SVV>
<C> : : = <I> <SVV> <F>
13) CVV_b — for <CVV>
<C> : : = <I> <CVV>
14) CVV_c — for <CVV>
<C> : : = <I> (<CVV>|<HV><CVV>)
<F>
15) HV_b — for <HV>
<C> : : = <I> <HV>
16) HV_c — for <HV>
<C> : : = <I> <HV> <F>
17) HV_{rr} — for <HV>
<C> : : = <I> (<HV><VV>|<HV>
<VV><F>)
18) $^{rr}C_b$ — for <SCO>
<C> : : = <I> <VV> <SCO>
19) $^{rr}CC_b$ — for <CCO>
<C> : : = <I> <VV> <CCO>
20) $^{hr}C_b$ — for <SCO>
<C> : : = <I> <HV> <SCO>
21) $^{hr}CC_b$ — for <CCO>
<C> : : = <I> <HV> <CCO>

It is convenient to combine classes C_{dhv}, C_{uhv} and C_{hvv} and classes CC_{dhv}, CC_{uhv} and C_{hvv} and classes C_{svv} and C_{cvv} and classes CC_{srr} and CC_{crr} into single classes defined by

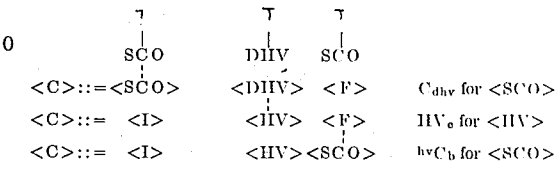
<C_{hr}> : : = <C_{dhr}>|<C_{hrr}>|<C_{uhr}>
<CC_{hr}> : : = <CC_{dhr}>|<CC_{hrr}>|<CC_{uhr}>
<C_{rr}> : : = <C_{srr}>|<C_{crr}>
<CC_{rr}> : : = <CC_{srr}>|<CC_{crr}>

and thus, only 15 separate and distinct classes are illustrated in the preferred embodiment of FIG. 5.

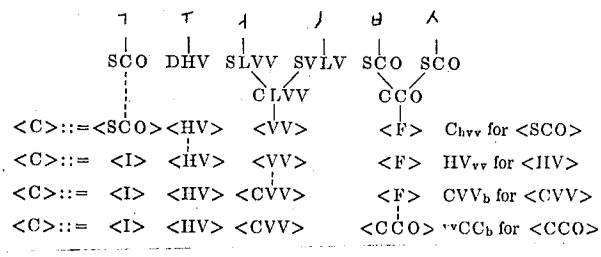
The type fonts for the compound consonants and the complex vertical vowels are not the type of fonts for the separate basic symbols, but rather the fonts for the CCO and the CVV are considered as the basic type fonts for the composite symbol in the character.

In the example below, a serial decomposed string of the symbols of a character will be analyzed by its compositional structure and the type fonts of the symbols are classified according to the classification scheme shown above to form a character for typing or printing.

Example 1



Example 2



The Character Composer

Coding the 24 Korean Characters

FIG. 6 illustrates a binary coding for the 24 basic Korean symbols. The decimal representations of the binary sequences in the table are arranged in ascending order of magnitude for the three basic symbol groups, SCO, SVV and HV. Thus, if the decimal representation of a letter has a smaller numerical value than that of another letter, the former is alphabetically ahead of the latter. The correspondence of the alphabetic order to the order of the magnitude of its decimal representation is preferable for the purposes of sorting and processing.

The literal BL in the table indicates a symbol for the blank or the horizontal space between the characters.

The five input bits $h_1 - h_5$ or output bits $O_1 - O_5$ represent a codeword for one element of the symbols in the column named SCO, SVV or HV. The three output bits $O_6 - O_8$ are used to identify a member of the class according to a given compositional structure classification scheme. For example, a set of a binary values (0, 0, 0) for (O_6, O_7, O_8) represents a character class <C_{vv}> named as C_{vv} in the table. Therefore, the eight output bits having the binar number (O_1, O_2, O_3, O_4, O_5, 0, 0, 0) represents a symbol for the column which belongs to the type font class <C_{vv}>. In other words, it represents a consonant symbol that is followed by a vertical vowel.

In FIG. 6, the symbols in the first column of each of the type font classes named C_{vv}, C_{hv}, $^{vv}C_b$ . . HV_{vv} are subscripted to distinguish between the ones of the classes. The type font in the second column of each of the classes are examples which use a consonant or vowel of the corresponding classes.

For each character, the character composer described below is utilized to recognize the particular binary input code $h_1 - h_5$ assigned to each of the basic symbols and to provide an output signal $O_6 - O_8$ identifying the type font classification.

Character Composer

A block diagram of the character composer is illustrated in FIG. 2. The input device 2 may be a computer or a keyboard which is used to generate a binary code for each of the basic 24 symbols. The code for each given signal is the one utilized in FIG. 6. The binary signal is transmitted to an input buffer and demultiplexer 4 via the five data lines 6. Thus, the first simple consonant having a binary code $h_1 - h_5$ equivalent to (1, 0, 0, 0, 1) is transmitted in parallel along lines 6 to the input buffer and demultiplexer 4. Additional clock and control lines 5 connect the input device to the input buffer and demultiplexer 4. The input buffer and demultiplexer 4 stores the incoming parallel bits and provides output signals to a plurality of syntax identifiers 8. The input buffer stage of the input buffer and demultiplexer stores as many as four separate symbols and each of these symbols is fed to a separate syntax identifier (SI–1 through SI–4). FIG. 4 illustrates a single syntax identifier and it is understood that each of the remaining three syntax identifiers is constructed in an identical fashion. The purpose of the syntax identifier is to classify the incoming parallel binary code according to one of six separate categories. The output of each syntax identifier is then fed to a control logic circuit 10. The function of the logic circuit is to determine from each of the identified character strings the corresponding type font in accordance with the coding groups as shown in FIG. 6. The control logic circuit 10 then feeds control signals to the input buffer and demultiplexer via lines 12 and additional coded signals via lines 14 to a diode matrix 16. The diode matrix may be the HM-050 (Harris Semiconductor) or any of a number of read only memory devices.

The output along lines 14 correspond to the several compositional structure groups as defined above. The diode matrix 16 provides a simple three bit binary code corresponding to these compositional structure groups. The output of diode matrix 16 is fed to flip-flops 17–1, 17–2 and 17–3 which are triggered by an output ready, O. R., signal from the output device. The output of the diode matrix corresponds to the binary code O_6, O_7, O_8 as illustrated in FIG 6. The code output group $O_6 - O_8$ together with signals $O_1 - O_5$ from the input buffer and demultiplexer are fed to an output buffer 18 for later transmittal to an output device such as a printer or computer 20. Thus, the character composer transmits the five bit binary code words $h_1 - h_5$ shown in FIG. 6 from the input device 2 to the output buffer 18. The output signals $O_1 - O_5$ are identical to the input signals $h_1 - h_5$. The output buffer 18 receives three additional signals $O_6 - O_8$ from the diode matrix 16 which characterize the particular compositional structure of the Korean symbol within the Korean character.

In practice, the input device 2 and the output device 20 may be one and the same printer or computer means and may conveniently be called an input/output (I/O) device. FIG. 2 illustrates additional clock and control lines 5′ from the output device 20 to the input buffer and demultiplexer 4. For a single I/O device the lines 5 and 5′ would serve to supply the necessary clock and ready signals for inputing and outputing data. This procedure simplifies the classification scheme and still permits practical coding of all desired characters.

Input Buffer and Demultiplexer

The input buffer and demultiplexer is shown schematically in FIG. 3. The five input signals $h_1 - h_5$ are fed from the input device in parallel form and connected to the D input terminal of J-K flip flops 21–1 through 21–5. The J-K flip-flops (for example, Texas Instruments SN 7475) may comprise the input buffer which stores an incoming data signal until the subsequent storage devices are ready. An input ready (I.R.) signal from the I/O device is applied to the C terminal of each of the J-K flip-flops 20, thus transferring the data signals through the Q output terminals. Each output terminal of the flip-flops is connected to a shift register 22 (for example Texas Instruments SN 7495). Four separate stages of the shift register are labeled 22A–22D. The Q output of the flip-flop is transmitted to the first stage of the shift registers. Shifting between the first and second stages and second and third stages and third and fourth stages of the shift registers is achieved by applying simultaneously to each stage a clock signal along the clock terminal CL. A reset terminal RS is also provided to reset each stage of the shift register (the registers are reset to the binary 1 state). Each stage of the five shift registers 22 have data output terminals $A\bar{A}$, $B\bar{B}$, $C\bar{C}$, and $D\bar{D}$ providing an output signal (A, B, C, D) and its complement ($\bar{A}$, $\bar{B}$, $\bar{C}$, $\bar{D}$). The output signals of each stage of each shift register are grouped together so that the output signals of all of the first stages are connected to terminals $\bar{A}_1$ A_1 – $\bar{A}_5$ A_5 and the second stages of the shift registers are connected to output terminals $\bar{B}_1$ B_1 – $\bar{B}_5$ B_5 etc. The output terminals $\bar{A}A$ – $\bar{D}D$ are each connected to separate syntax identifiers such as illustrated in FIG. 4 and discussed in detail below.

The second and third stages of each of the shift registers are connected to output J-K flip-flops 24–1 through 24–5 via AND gates G1 – G10 and OR gates g1 – g5. The AND gates G1, G3, G5, G7 and G9 are each connected to an input terminal T_2 and to the second stage of one shift register 22. AND gates G2, G4, G6, G8 and G10 are connected to an input terminal T_3 and to the third stages of the respective shift registers 22. The output of each pair of AND gates G1 – G10 is fed to the corresponding OR gate as illustrated in FIG. 3, and the output of each of these OR gates is connected to the D input terminal of the flip-flop 24. The clock pulse for each of the flip-flops 24 is supplied by the I/O device to a clock terminal O.R. which connects to the clock terminal C of each of the flip-flops 24. The output sginals $O_1 - O_5$ for flip-flops 24 are supplied from terminal Q and fed to the output buffer 18 as shown in FIG. 2.

Syntax Identifier

Since the binary code for each Korean symbol is fed in parallel fashion to the four stages of the shift registers 22 of FIG. 3, it is clear that the first stages of all five shift registers contain a single code corresponding to one of the basic symbols of FIG. 6. In addition, the second stages as a group contain a second basic symbol, the third stages a third basic symbol and the fourth stages a fourth basic symbol. Terminals $\bar{A}_1$ A_1 – $\bar{A}_5$ A_5 through $\bar{D}_1$ D_1 – $\bar{D}_5$ are fed to four separate syntax identifiers, one of which is illustrated in detail in FIG. 4. The input signals $\bar{A}_1$ A_1 – $\bar{A}_5$ A_5 are fed to a plurality of AND gates G20 – G30 which are in turn connected as shown in FIG. 4 to OR gates g10 and g11. Additional

AND gates G31–G35 are connected as shown and have outputs connected to output terminals BL_1, SCO_1, SVV_1, HV_1, ㄴ 1, and ㅈ 1. The subscript identifies the syntax identifier. These output terminals from each of the four syntax identifiers are connected to the control logic circuitry as illustrated in FIG. 5.

Control Logic

The logic circuitry comprises AND gates G41 – G72, OR gates *g*20–*g*27 and inverter 1. The input terminals from the four syntax identifiers are marked with subscripts 1–4 which corresponds to the first to fourth syntax identifiers which correspond to the first to fourth stages of the shift registers 22. The output of the control logic circuit as illustrated in FIG. 5 comprises terminals Y1 – Y15 which correspond to the various compositional structure classes as listed in FIG. 6. Additional output terminals C.R., C, E and CL are provided for control purposes. Terminals T_2 and terminals T_3 are connected to the similarly labeled terminals in FIG. 3.

Not all of the output signals from each syntax identifier are utilized in the control logic. For example, only the blank symbols BL_1 – BL_3 are utilized and only the special symbols ㄴ 3 and ㅈ 2 are needed. The blank is used at the input device to separate characters, and the logic recognizes and utilizes this information in the first three stages of the shift register only. The need for only the ㄴ 3 and ㅈ 2 signals result from the limited practical usage of these symbols within the Korean language.

Operation

The operation of the character composer as shown in FIGS. 3–5 is best illustrated by way of an example. A simple example, which was utilized above is the "구". This character is composed of only three basic symbols ㄱ, ㅜ, ㄱ corresponding to the binary bode groups **10001, 00111** and **10001.** Each of these five bit codes h_1 – h_5 are separately transmitted in parallel fashion to the input buffer, J-K flip-flops 21 of FIG. 3, via the data input lines 6. Prior to the transfer of the input characters, the reset signal RS supplied by the I/O device is used to set each stage of the five shift registers to the binary 1 state. On the application of the first clock CL pulse which is supplied by the I/O device after the input code is ready for transmission, the first binary code group corresponding to **10001** is transferred to the first stages of the shift registers **30** (FIG. 3). Thus, the output terminals $\bar{A}_1$ – $\bar{A}_5$ contain the binary code **10001** which will be transferred to the first syntax identifier as shown in FIG. 4. The code complement is supplied along terminals A_1 – A_5. At the same time the remaining three syntax identifiers will be used to decode the second, third and fourth stages of each of the five shift registers 22. However, since these latter stages have all been reset to the binary 1 state, the output code for B_1 – B_5, C_1 – C_5 and D_1 –D_5 are all identical to the "set" or binary 1 state. As can readily be seen by examining the circuit diagram or the table of FIG. 1, the first code bit for each simple consonant is a binary 1, whereas the binary 0 in the first bit position represents either a simple vertical vowel or a horizontal vowel. Thus, the first syntax identifier produces an output signal at terminal SCO. The remaining three syntax identifiers produce output signals at blank terminal BL since the terminal corresponds to the condition or code having all bits set to one. The above signals are fed to the input terminals of the control logic circuit and result in gating on the AND gate G49 which in turn causes a signal along terminal CL to be generated (via gate *g*21). Signal CL is the clock signal of FIG. 3. The fact that only the CL signal (the second such clock pulse in the example) is generated by the control logic simply indicates that no classification of the first simple consonant is yet possible. In fact, additional basic symbols are required since a given font depends upon the other symbols immediately preceeding and following a given symbol.

The clock signal at terminal CL causes an additional transfer from the J-K flip flops 20 to the first stages of the shift registers. In turn, the binary data in the first stages of the shift registers 22 is shifted to the second stages and data in the second stage shifted to the third stage etc. Again the syntax identification occurs for each of the output terminals A_1 – A_5, B_1 – B_5, etc.

After the second CL pulse, the second stages of the five shift registers contain data bits corresponding to a simple consonant, 10001. The SCO_2 terminal of the second syntax identifier will be activated. The first stages, stage A, of the shift registers then contain a code for a horizontal vowel (DHV) code 00111 which activates the HV_1 terminal in syntax identifier SI–1. In addition, the BL_3 terminal of the third syntax identifier is activated since the data shifting causes all of the stages in the shift register to shift one register position to the right. In these circumstances, the input signals to the control logic of FIG. 5 causes gate G64 to be triggered which in turn provides a pulse or a first control signal along terminal T_2 as well as terminal Y14. The signal along terminal Y14 identifies the simple consonant, now stored in the second stages of the shift registers, as belonging to the classification group C_{hv}. since the classification for this particular symbol is now complete, the signal along terminal T_2 gates the corresponding AND gates G1, G3, G5, G7 and G9 of FIG. 3 to transfer data from the second stages of the shift registers to the output J-K flip flops 24. The output of these flip-flops 24 provide the output signals O_1 – O_5 (10001) to the buffer 18. As can be seen in FIG. 2, the output signal Y14 is fed via data channel 14 to a diode matrix 16 which provides corresponding output signals O_6 – O_8 indicative of the classification C_{hv} as shown in FIG. 6. Thus, the eight output signals O_1 – O_8 completely identify the type font to be used with the first simple consonant represented by the symbol 10001 of FIG. 6. It will be noted that only two symbols, the first simple consonant and the following horizontal vowel, were needed in this particular case to determine the font classification of the first simple consonant. However, in order to determine the type font associated with the second symbol, the horizontal vowel, additional symbol data is needed.

After the transfer from the output buffer 18 to the printing or output device 20, the output device provides a clock pulse along lines 5′ to the shift registers 22 to cause another transfer and shifting of data. It will be noted that even though the binary data in the second stage of the shift registers was transferred to the output buffers 24–1 through 24–5 this same data is nevertheless shifted to the third stages of the shift register 22 just as if no classification had occurred. Thus, after the third clock pulse CL has been sent, the first, second and third stages (22A, 22B and 22C) of the shift registers contain data corresponding to the third, second and fourth input symbols respectively. In the example, the first, second and third stages of shift registers 22 contain the codes (10001), (00111) and (10001) re-

spectively. Under this situation, the control logic of FIG. 5 results in the firing of gate G55 and the consequent signal along terminal Y10 and terminal T_2. The signal along terminal Y10 identifies the data in the second stage of the shift register as belonging to the classification HV_c. The signal along terminal T_2 transfers the data in this second stage to the output buffer as before.

The last and remaining identification necessary is for the final character which is a simple consonant. Upon application of a fourth clock pulse, supplied by the I/O device, a blank signal would be transferred to the first stages of the shift registers of FIG. 3 since blank signals separate all of the input characters. Under these conditions, the control logic of FIG. 5 would trigger gate G44 which results in generating terminal Y4 as well as firing OR gate g27 to initiate a pulse along terminal CR. The signal along Y4 identifies the last symbol of the character as belonging to the class $^{hv}C_b$. The signal along terminal CR is used to signal the I/O device to produce the blank needed to separate the just completed character.

Thus, it is seen that the control logic and associated circuitry identifies the single character by first identifying each symbol as a simple consonant, horizontal vowel or vertical vowel and then depending upon the order of the various symbols, the circuitry selects the type font for each symbol.

In the control logic of FIG. 5 additional control signals are provided. The signal at terminal C may be used to produce a fixed margin and is generated by inputting the blank symbol BL two consecutive times, e.g., gate G41 is opened by simultaneous signals at BL_1 and BL_2. The error signal at terminal E is generated whenever certain grammar rules are violated such as inputting a vowel as the first symbol following a blank BL (e.g., gate G63 goes on). The error signal may be used to provide an indication at the I/O device to alert the operator of an incorrect entry. The signal at terminal C.R. is used to generate a blank symbol at the output device and is OR gated to the final symbol in a character represented by the classes at terminals Y1 – Y7.

The signals at T_2 and T_3 are used to transfer the code contents of the second and third stages respectively of the shift registers 22 to the output flip-flops 24. In some cases such as in using terminals $_3$ and $_2$ it is necessary to know the syntax identification of the two symbols following a symbol as well as the symbol preceeding the symbol in question. In these cases the signal, the second control signal, along terminal T_3 is needed.

FIG. 7 illustrates a variant of the input coding which may readily be utilized. In addition to the 24 basic symbols, it is practical to form a separate special input code for commonly used combinations of the basic symbols such as the compound vowels shown in the last column of FIG. 7.

It is to be understood that the above described embodiments are only illustrations of the application of the principles of the invention and that various modifications may be made by those skilled in the art without departing from the spirit and scope of the invention.

I claim:

1. A character converter for generating output signals indicative of a font for the basic symbols of a character from a set of coded input signals identifying each symbol comprising:

a. a plurality of four stage shift registers for storing successive identifying coded signals,

b. geometrical group classifying means connected to said plurality of shift registers for classifying said stored identifying signals into geometrical groups and for providing geometrical group output signals,

c. control circuit means connected to receive the group output signals from said geometrical group classifying means, said control circuit means providing compositional structure output signals associated with each identifying coded signal, and

d. output buffer means connected to receive said compositional structure output signals and said associated identifying coded signals,

said control circuit means comprising:

1. means for generating a first control signal for shifting the stored identifying coded signals in the second stages of the plurality of shift registers into said otput buffer means, and

2. means for generating a second control signal for shifting the stored identifying coded signals in the third stages of said plurality of shift registers into said output buffer means.

2. A character converter as recited in claim 1 wherein said control circuit further comprises a diode matrix.

3. A character converter as recited in claim 1 wherein said control circuit further comprises a read only memory.

4. A character converter as recited in claim 1 wherein the stages of each of said plurality of shift registers are connected to separate geometrical group identifying means.

5. A character converter as recited in claim 4 wherein said shift registers are connected to receive a parallel string of identifying coded signals.

6. A character converter as recited in claim 1 wherein said compositional structure groups comprises the groups:

C_{vv}, CC_{vv}, C_{hv}, SVV_b, SVV_c, CVV_b, CVV_c, HV_b, HV_c, HV_{vv}, $^{vv}C_b$, $^{vv}CC_b$, $^{hv}C_b$, $^{hv}CC_b$, and CC_{hv}.

7. A character converter as recited in claim 1 said converter further comprising:

a. means connected to said storage means for identifying said stored coded signals as representing vowel and consonant symbols,

b. means connected to said vowel and consonant identifying means for detecting a grammatically invalid succession of successively stored vowel and consonant coded signals, and

c. means for generating an error signal upon the occurrence of a grammatically invalid succession of stored coded signals.

8. A character converter as recited in claim 7 wherein said coded input signals comprise a distinct identifying code for distinguishing vowels and consonants.

9. A character converter as recited in claim 1 wherein said compositional structure groups comprise the groups:

C_{svv}, CC_{svv}, C_{cvv}, CC_{cvv}, C_{dhv}, CC_{dhv}, C_{uhv}, CC_{uhv}, C_{hvv}, CC_{hvv}, SVV_b, SVV_c, CVV_b, CVV_c, HV_b, HV_c, HV_{vv}, $^{vv}C_b$, $^{vv}CC_b$, $^{hv}C_b$, and $^{hv}CC_b$.

* * * * *