

# 우리말 情報의 入力處理시스템

—IBM機器利用을 中心으로—

李 貴 榮  
(KIET電算室)

〈차 례〉

- I. 背 景
- II. 우리말 情報處理의 特性
- III. 우리말 情報의 入力處理 시스템
- IV. 問題點 및 앞으로의 方向

## I. 背 景

現代文明을 대변하는 가장 대표적인 것이 컴퓨터라 할 수 있다. 거의 모든 分野에 컴퓨터가 利用되고 있어 情報處理에 있어서도 예외는 아니다. 産業이 발달하고 社會가 복잡해짐에 따라 情報量이 急增하여 情報處理의 컴퓨터化가 不可避하게 되었다.

컴퓨터는 원래 數值 데이터 處理를 위하여 만들어 졌으나 英語를 使用하는 先進國에서는 이미 英文字 處理를 實用化하였고, 日本에서도 컴퓨터에 의한 日本語 處理를 완성하였다. 우리나라에서도 한글의 컴퓨터化가 오래전부터 관심의 대상이 되어왔으며 끊임없는 연구개발로 여러가지 한글處理 시스템이 開發 登場하고 있다. 韓國産業經濟技術研究院(KIET)에서는 이미 한글漢字情報處理시스템인 KIPS 1,2,3(Korean Information Processing System)를 開發하여 한글漢字處理의 先導的 역할을 하여왔다. 한글漢字處理의 窮極的 目標은 우리말 데이터 베이스 製作과 對話式 檢索시스템을 開發하여 우리나라도 하루빨리 名實相符한 情報化社會로 發展시키기 위함이다. 보다 효율적인 우리말 檢索시스템을 만들기 위해서는 國家的 次元에서의 한글, 漢字入力 code 表準化, 우리말

索引語 作成, 한글·漢字自動變換시스템 完成 등이 문제로 남아있어 이를 위한 구체적인 開發이 시급한 상황이다. 既 開發된 KIPS 시스템은 1976년에 도입한 S-5300 시스템에 맞게 開發된 것으로서, 여러가지 長點이 있으나 容量 機器를 위한 것이며, 機器도 老朽되어 좀더 普遍化되고 效率적인 處理를 위하여 IBM機器에 맞도록 開發하게 되었다.

## Ⅱ. 우리말 情報處理의 特性

### 1. 한글의 構成

한글은 子音 14字, 母音 10字의 모두 24字의 基本字素로 이루어져 있으며 이러한 基本字素는 字素와 字素의 結合으로 複合字素를 形成한다. 그리고 다시 이들 基本字素 및 複合字素들이 初聲, 中聲, 終聲으로 組合되어 비로소 하나의 音節을 이루게 된다. 이것이 英語, 日本語와 같이 풀어쓰기를 하는 文字와 다른 점이다.

현재 사용되고 있는 한글 字素는 子音字素 30개, 母音字素 21개이며 그 내용은 <表 1>과 같다. 이들 51개의 字素中 初聲으로 사용될 수 있는 것은 基本子音과 쌍자음뿐이고 複字音은 사용되지 않는다. 中聲으로는 모든 母音이 사용되며 終聲은 基本子音과 複子音, 그리고 쌍자음으로는 ㄱ, ㅅ만 사용된다.

<表 1> 한글의 자소

구분	내용	갯수
자음	기본자소 ㄱ ㄴ ㄷ ㄹ ㅁ ㅂ ㅅ ㅇ ㅈ ㅊ ㅋ ㆁ ㆅ ㆆ ㆇ ㆈ ㆉ ㆊ ㆋ ㆌ ㆍ ㆎ ㆏ ㆐ ㆑ ㆒ ㆓ ㆔ ㆕ ㆖ ㆗ ㆘ ㆙ ㆚ ㆛ ㆜ ㆝ ㆞ ㆟ ㆠ ㆡ ㆢ ㆣ ㆤ ㆥ ㆦ ㆧ ㆨ ㆩ ㆪ ㆫ ㆬ ㆭ ㆮ ㆯ ㆰ ㆱ ㆲ ㆳ ㆴ ㆵ ㆶ ㆷ ㆸ ㆹ ㆺ ㆻ ㆼ ㆽ ㆾ ㆿ ㆿ	14
	쌍자소 ㄱㅈ ㄴㅊ ㅂㅅ ㅅㅈ	5
	복합자소 ㄱㅈ ㄴㅊ ㄴㅎ ㄹㄹ ㄹㅁ ㄹㅂ ㄹㅅ ㄹㅈ ㄹㅎ ㅂㅅ	11
모음	기본자소 ㅏ ㅑ ㅓ ㅕ ㅗ ㅛ ㅜ ㅠ ㅡ ㅣ	10
	복합자소 ㅐ ㅑ ㅓ ㅕ ㅗ ㅛ ㅜ ㅠ ㅡ ㅣ	11

따라서 한글은 <表 2>와 같이 初聲 19字, 中聲 21字, 終聲 27字로서 이들의 組合으로 모두 11, 172字 (19 × 21 × 28)를 만들수 있으나 실제 사용하는 字數는 약 1,500字 정도이다.

〈表 2〉

한글자소의 구성내용

구 분	자 모 수	자 소 내 용
초 성	19	기본자음 14, 쌍자음 5
중 성	21	기본모음 10, 복합모음 11
종 성	27	기본자음 14, 복합자음 11, 쌍자음(ㄱ, ㅋ) 2

## 2. 漢字의 併用

우리말 표기에 있어서 漢字의 併用이 불가피하므로 한글과 더불어 漢字의 處理도 필수적이다. 한글은 풀어쓰기가 가능하고 조합된 글자수도 적으나 대량의 漢字 데이터는 入力에 많은 난점을 가지고 있다.

### 3. 2 Bytes 코드 체계

일반적으로 컴퓨터에서는 文字코드를 1 Byte로 表示하며 1 Byte는 8Bits로 構成되어  $2^8=256$ 種을 表現할 수 있다. 英文處理에서는 알파벳, 數字, 特殊記號를 모두 합하여도 256種에 훨씬 미치지 못하므로 1 Byte를 文字의 한 單位로 하였다. 한글은 字素가 51개이나 모아쓴 글자로 使用되고 있는 것이 1,500字 가량되며, 漢字는 選擇方法에 따라 그 보다 많다. 즉 1 Byte로는 모든 글자를 표시할 수 없어 2 Bytes 코드 체계를 택해야 한다.

## Ⅲ. 우리말 情報의 入力處理 시스템

### 1. 시스템의 概觀

S-5300 하드웨어를 중심으로 한 한글·漢字 處理 시스템, KIPS1,2,3로 한글·漢字處理를 어느정도 階도에 올려놓았다. 이 KIPS-2 시스템에서는 入力時에 한글을 풀어쓰기로 入力시킴으로써 off-line Keyboard상의 字盤中에서 해당 문자를 일일이 찾아내야 하는 노고와 시간을 줄일 수 있고, 初聲, 中聲, 終聲의 文字像 合成 시스템으로 文字像 記憶場所(Font Memory)를 절

약할 수 있었다. 文字像合成시스템이란 각 글자코드마다 1:1로 대응되는 文字像(Font Image)을 필요로 하지 않고 각 字素의 文字像을 기억시켜 合成하는 方法인데 완전한 形態의 글자를 合成하기 위해서는 각 字素를 위치별로 여러가지 형태의 文字像이 필요하다. 그래서 모두 712字의 字素를 저장하였는데 이는 實用한글이 1,500여자인데 비해 크게 절약이 안되는 실정이며, 각 字素들의 文字像을 合成하는 데 걸리는 시간도 무시할 수 없었다.

그러나 SP7 Printer 內에는 충분한 容量의 文字像 記憶場所(Font Memory)가 內裝되어 있기 때문에 몇 자의 Memory를 줄이기 위해 복잡한 과정을 거치지 않아도 되도록, 한 글자코드에 하나씩 대응되는 文字像을 기억시켜 놓을 수 있다. 이 文字像 코드들은 하나의 子音에 대해서 母音 21個씩 Match하여 一次 array로 順次的으로 排列되어 있다. 이러한 特性을 이용하면, 풀어쓰기로 入力된 각 字素들의 順序를 가지고 간단히 계산하여 組合된 글자의 文字像이 있는 위치를 찾아갈 수 있다.

入力方式은 Kips-2와 같이 한글은 풀어쓰기로 천공한 후 組合하고, 漢字 및 그 밖의 文字는 Full Keyboard方式을 채택하였다.

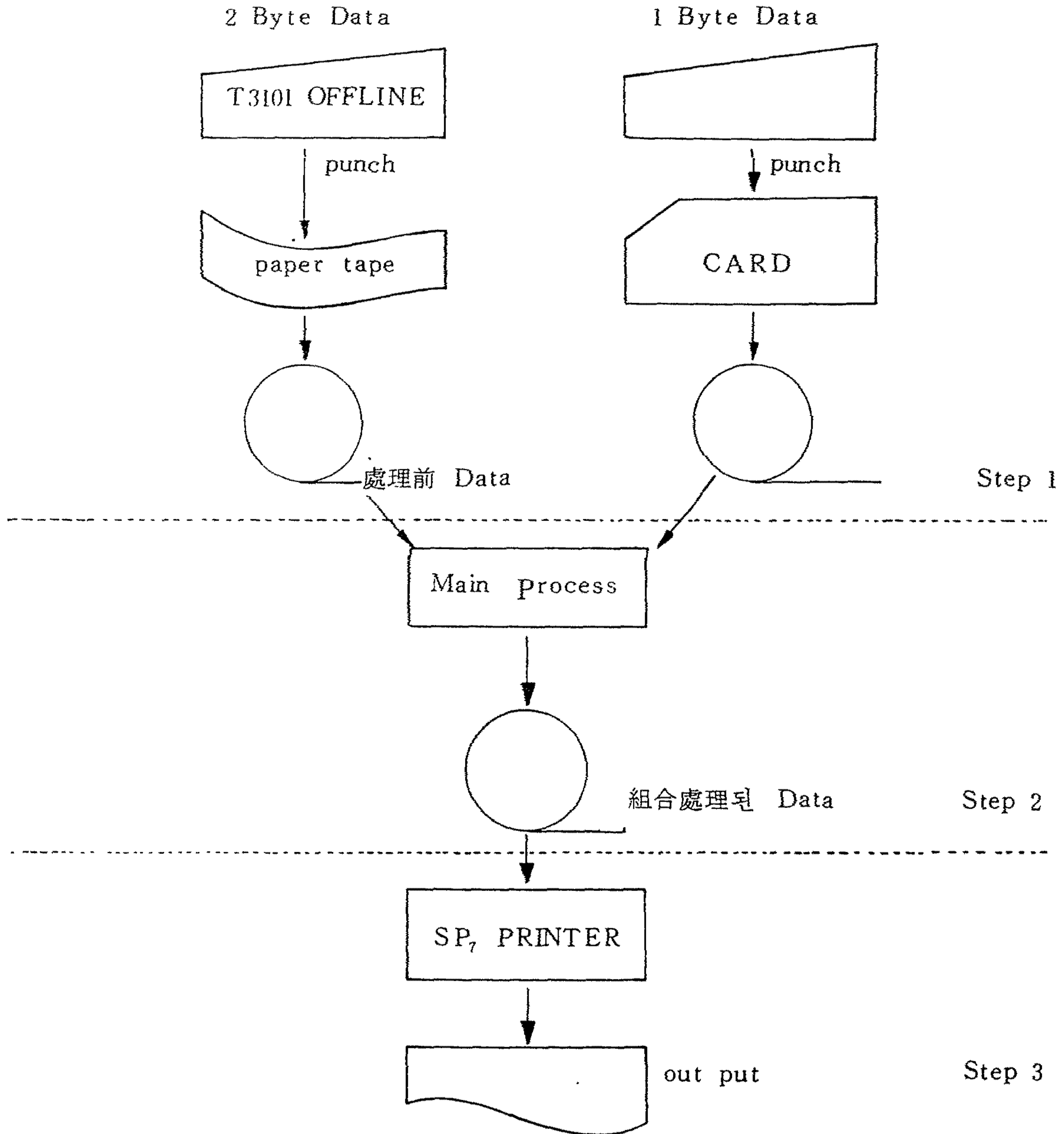
한글을 풀어쓰기로 入力할 경우 각 기관마다 선정한 자판의 字素數는 모두 다른 실정이다. 이에 따라 어떠한 체계의 자판에도 구애되지 않도록, 複合字素에 있어서는 基本字素로 入力하여 組合할 수도 있고 組合된 形態로 入力해도 무관하도록 하였다. 그러나 “깍” 혹은 “뚝” 등 初聲이 二中子音인 경우에는 그 初聲만은 풀어서 入力할 수 없는 制約이 필요하다. 이 制約은 풀어서 入力되는 한글 字素들에서 前字의 終聲과 後字의 初聲을 구별하기 위함이다. 예를들면 “닭고”라는 단어를 풀어서 入力할 경우 풀어쓴 “ㄷ, ㅏ, ㄱ, ㄱ, ㄱ, ㅏ” 중에 어디까지가 前字 “닭”의 終聲이고 어디서부터가 後字 “고”의 初聲인지 위의 制約없이는 가려내기 힘든 일이다.

## 2. General System Flow

全般的인 處理흐름은 <圖 1>에서 보는 바와 같이 크게 3段階로 나눌수 있다. 첫번째 단계인 Step 1에서는 T 3101 off-line Keyboard(2Bytes Data)나 IBM punch Machine(1Byte Data)으로 천공한 데이터를 Magnetic Tape 形態로 변환하는 과정이다.

<圖 1>

일반적인 작업 흐름도



2 Byte Data인 경우, 한글을 제외한 한자, 특수문자등이 모두 文字像(Font Image) 코드와 일치된 상태로 들어있고, 한글은 풀어써진 각 字素의 文字像 코드가 2 Bytes 상태로 들어있다. 이 때 入力機에 따라 코드가 다르므로 필요에 따라 Code Conversion 작업을 수행한다.

1 Byte Data인 경우에는 풀어써진 각각의 한글字素 및 그 외의 文字들이 1 Byte 상태로 들어있는데 한자처리에는 적합하지 않다. 단 漢字가 거의 없

이 드물게 꼭 필요한 경우에는 漢字임을 알려주는 indicator와 함께 文字像 코드를 入力하면 가능하다.

1 Byte 시스템에서 나타낼 수 있는 정보수는 실제로는 94 가지 밖에 안되어 英文字의 大·小文字, 數字, 特殊記號를 넣으면 꼭 차 버린다. 그래서 英文字 및 몇 개의 特殊文字코드와 한글字素코드가 중복으로 부여되어 한글 入力時에는 한글임을 표시하는 indicator가 필요하다.

두번째 단계인 Step 2에서는 Main Process 過程으로 한글 字素들을 組合, 그 글자의 文字像 코드를 찾아낸다.

세번째 단계인 Step 3에서는 Step 2의 결과로 얻어낸 MT File을 SP7 Printer에 걸어 文字像記憶裝置에서 文字像을 불러 프린트한다.

### 3. 프로그램 Step別 機能

#### (1) 2 Bytes Data 處理 프로그램

##### 1) 入力 Data의 取讀 및 分離

종이 tape 形態의 Data를 Magnetic tape 形態로 변환시키는 프로그램에서 Record Size는 1 Block 1 Record로 하여 300 words(600 bytes)씩 만들어 두었으므로 MT File을 읽는 Read Buffer Size는 600 words이다. Read Buffer에 읽혀들어 온 Data들은 2 Byte 단위로 된 element 1개씩 指摘 處理된다.

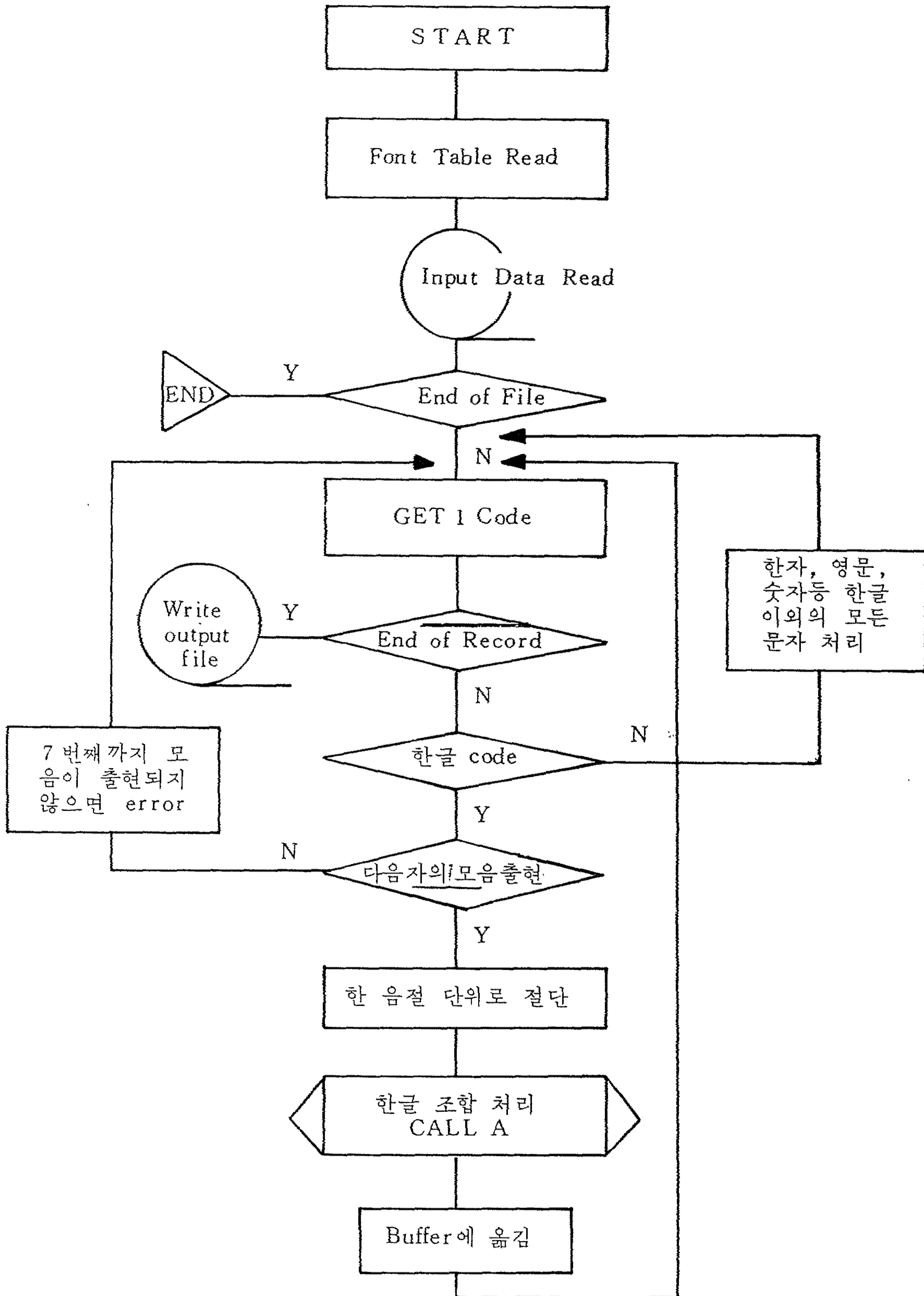
먼저 처음 Data부터 3번째까지 사이에 한글 이외의 文字가 있는지 확인한다. 만약 한글 이외의 文字가 있으면 Direct Routine (한글 以外 Data 처리과정)으로 가고, 없으면 4번째부터 7번째 Data 중 한글 以外의 文字가 있는지 확인하여 같은 方法으로 처리한다.

Direct Routine에서는 바로 앞의 Data가 한글이면 한글처리를 먼저 수행한다. 한글 以外의 Data들은 바로 2 bytes 상태의 文字像(Font image)코드 자체가 入力되었으므로 入力 Data를 그대로 Out Buffer에 옮기면 된다.

4번째부터 7번째 Data 사이에서 한글 이외의 Data가 발견되지 않으면 母音이 있는지 확인하여 母音 바로 前의 子音 앞에서 절단한다. 이것이 바로 한글자를 이루는 단위자소들인데 最少 2個에서 最大 5個로 구성된다. 이들 單

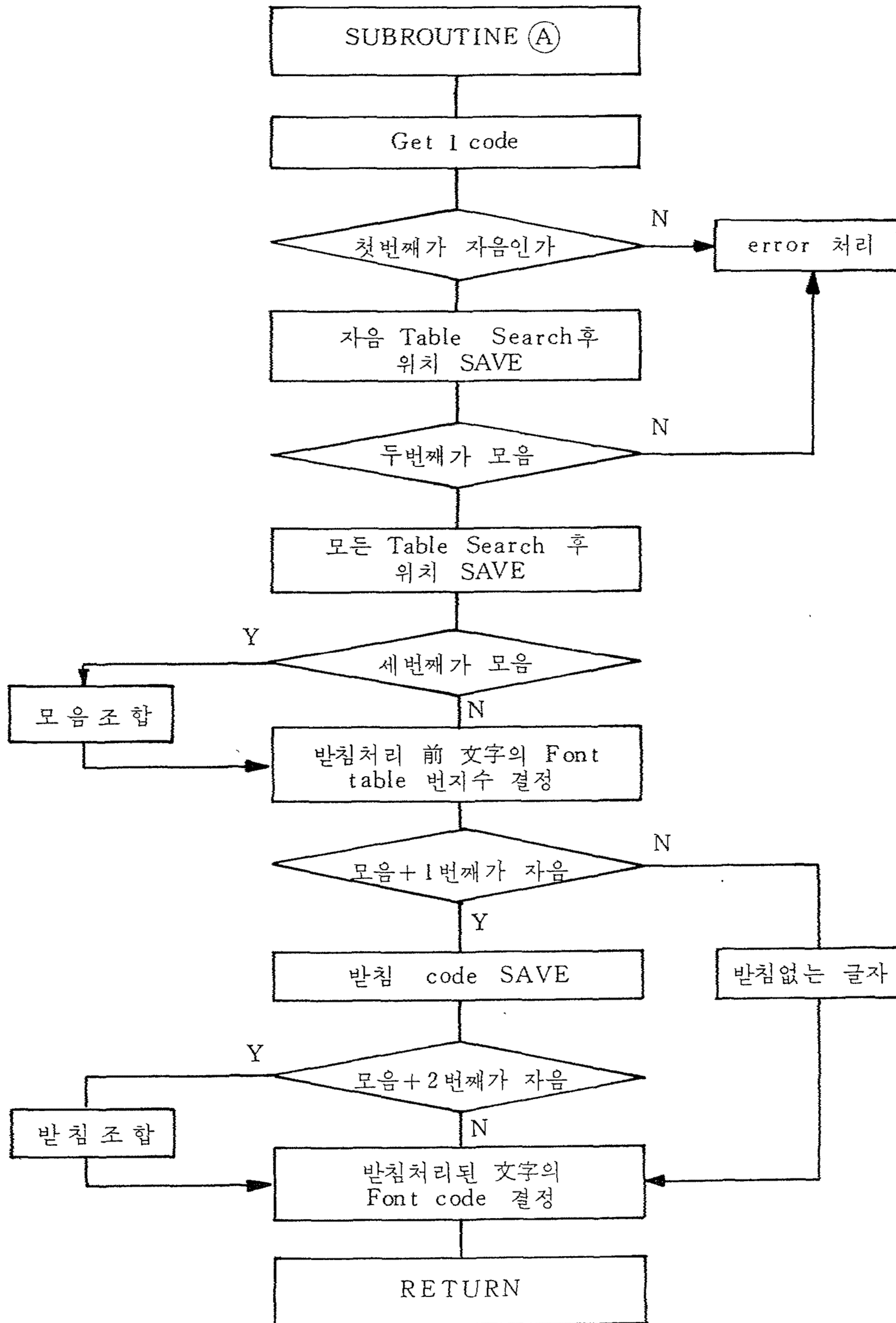
<圖 2>

Step-2 (Main Process 過程)의 處理 흐름도



<圖 3>

한글 組合 Sub-Routine 의 處理 흐름도





位字素들이 組合 Routine 의 入力 Data 가 되며, 組合된 後에는 절단된 바로 다음 Data 로 돌아가서 같은 方法으로 반복된다.

## 2) Element Code 의 組合

### ① 初聲處理

한글을 풀어쓸 경우 초성, 중성, 종성 순으로 排列되므로 첫번째 element 는 항상 子音이어야 한다. 그러므로 첫번째 element 를 프로그램內에 確保된, 文字像코드 table 에서의 子音順과 일치하는 初聲 Table (表 3)에서 Search 하여 그 때의 Index 값 (順序) (I) 을 Save 해 둔다.

〈表 3〉 program內的 子音 Table

Element	ㄱ	ㄲ	ㅋ	ㆁ	ㄷ	ㄸ	ㄹ	ㄴ	ㄷ	ㅁ
Index	1	2	3	4	5	6	7	8	9	10
Element	ㄴ	ㅇ	ㅅ	ㅆ	ㅈ	ㅊ	ㅌ	ㅍ	ㅎ	
Index	11	12	13	14	15	16	17	18	19	

### ② 中聲處理

두번째 element 를, 文字像코드 table 의 母音順과 일치하는 母音 Table 에서 Search 하여 그 Index 값 (J) 을 Save 해 둔다. 組合될 母音들이 풀어서 들어올 수 있으므로 세번째 element 도 母音인지 확인한다. 만약 세번째 element 가 母音이면 母音 Table 에서 Search 하여 그 Index 값을 Save 한 후 두 Index 값을 이용하여 프로그램內에서, 組合된 母音의 새로운 Index 값을 찾아낸다. 즉 첫번째 母音 ( $M_1$ ) 이 “ㄴ” 이고 두번째 母音 ( $M_2$ ) 가 “ㅌ” 이면

㉠ “ㄴ” 의 Table 에서의 위치 ;  $M_1 = 9$

㉡ “ㅌ” 의 Table 에서의 위치 ;  $M_2 = 1$

㉢ 組合된 母音 “ㄴㅌ” 의 위치 ;  $M_3$  는 아래의 과정을 거쳐 “10”

임을 알아 낼 수 있다.

이  $M_3$  의 값이 바로 組合된 母音의 Font Code table 에서의 順序 (J) 가 된다.

<表 4>

Program 内の 母音 Table

Element	ㅏ	ㅑ	ㅓ	ㅕ	ㅗ	ㅛ	ㅜ	ㅠ	ㅡ	ㅣ	ㅞ
Index	1	2	3	4	5	6	7	8	9	10	11
Element	ㅟ	ㅠ	ㅓ	ㅕ	ㅑ	ㅓ	ㅕ	ㅗ	ㅛ	ㅜ	ㅠ
Index	12	13	14	15	16	17	18	19	20	21	
풀어 써진 각 모음의 Index						조합후 Index					
M <sub>1</sub> =	1	&	M <sub>2</sub> =	21	M <sub>3</sub> =	2					
M <sub>1</sub> =	3	&	M <sub>2</sub> =	21	M <sub>3</sub> =	4					
M <sub>1</sub> =	5	&	M <sub>2</sub> =	21	M <sub>3</sub> =	6					
M <sub>1</sub> =	7	&	M <sub>2</sub> =	21	M <sub>3</sub> =	8					
M <sub>1</sub> =	9	&	M <sub>2</sub> =	1	M <sub>3</sub> =	10					
M <sub>1</sub> =	9	&	M <sub>2</sub> =	2	M <sub>3</sub> =	11					
M <sub>1</sub> =	9	&	M <sub>2</sub> =	21	M <sub>3</sub> =	12					
M <sub>1</sub> =	14	&	M <sub>2</sub> =	5	M <sub>3</sub> =	15					
M <sub>1</sub> =	14	&	M <sub>2</sub> =	6	M <sub>3</sub> =	16					
M <sub>1</sub> =	14	&	M <sub>2</sub> =	21	M <sub>3</sub> =	17					
M <sub>1</sub> =	19	&	M <sub>2</sub> =	21	M <sub>3</sub> =	20					

③ 終聲處理

하나의 음절을 구성하는 最少 element 數는 2자이다. 즉 세번째에 element 가 없으면 받침이 없는 글자이다. 이 글자의 Font Code table에서의 위치는 간단한 계산으로 알 수 있다. 받침이 없는 상태의 文字像코드는 <表 5>에서 보는 바와 같이 자음 19개에 대해서 각각 21개의 모음이 대응되도록 만들어져 있다. 받침없는 글자 “너”인 경우 “ㄴ”의 Index I = 3, “ㅏ”의 Index J = 5이며 따라서 組合된 “너”字의 위치 (K)는

$K = (I-1) \times 21 + J = 47$ 이 된다. 즉 文字像코드 table <表 6>에서 47번째의 코드가 바로 “너”字의 코드“4025”가 된다.

세번째 element 가 子音인 경우 終聲으로 간주되며 네번째도 子音인지 확인하여 子音이면 두개의 코드값을 더하여 Save 해 둔다(이 때 入力된 字素의 코드값은 2 byte 체계의 코드값인데 文字像코드 table에 나열된 받침의 코드값은 1 Byte 체계에서의 코드값이므로 바꾸어 줄 필요가 있다). “ㄴ” 받침을 예를 들면 “ㄴ”의 코드값이 217이고 “ㅏ”의 코드값이 227이면 “ㄴㅏ”의 값은 444가 된다.

〈表 5〉 받침없는 글자의 文字像 코드 Table

ㅏ	3728	3888	3994	4118	4223	4297	4393	4394	4602	4668	4809	4885	5070	5193	5263	5342	5437	5527	5612
ㅑ	3746	3899	4010	4134	4236	4307	4407	4507	4612	4683	4819	4900	5082	5205	5276	5350	5447	5538	5623
ㅓ	3756	3908	4020			4316		4620	4693	4827	4909	4921	5091	5284	5359				5632
ㅕ	3759		4024																
ㅗ	3760	3910	4025	4143	4245	4319	4418	4517	4622	4700	4830	4923	5095	5215	5288	5361	5456	5548	5634
ㅛ	3778	3920	4040	4157	4256	4329	4428	4528	4631	4716	4839	4941	5106	5296	5371	5465	5557	5643	5643
ㅝ	3785	3922	4047			4337	4435	4537	4632	4725	4949	4949	5113	5222	5301	5379	5474	5565	5651
ㅟ	3798		4056			4346				4729	4964	4964						5573	5660
ㅛ	3801	3924	4057	4165	4262	4347	4443	4548	4636	4730	4842	4968	5116	5224	5303	5386	5475	5574	5661
ㅜ	3815	3934	4068					4557		4740	4849	4981	5127	5233		5394			5670
ㅠ	3823	3940								4743		4990	5136		5400	5483	5483		5677
ㅡ	3830	3943	4070		4267	4355	4453	4560		4746	4852	4994	5140	5235	5312		5484	5582	5681
ㅓ	3836	3948	4074			4356	4455	4566	4643	4752		5001	5146	5313	5405			5584	5685
ㅕ	3837	3949	4076	4186	4268	4358	4456	4567	4646	4758	4857	5007	5151	5241	5314	5406	5486	5586	5686
ㅗ	3849	3959	4085	4174		4366	4468	4578		4770	4863	5018	5160	5247	5322	5413			5693
ㅛ	3853	3962		4196						4771		5027			5324	5415	5494		5697
ㅝ	3855	3965	4088	4197	4274	4367	4473	4579		4773		5032	5162		5325	5417	5496	5594	5701
ㅟ	3861	3971	4093	4199		4368	4475			4780	4865	5039	5168	5248	5326	5420	5503	5595	5709
ㅡ	3865	3972	4096	4201	4279	4374	4476	4581	4653	4786	4867	5046	5171	5249	5327	5421	5508	5596	5713
ㅓ			4106	4287							4873	5054					5516		5724
ㅗ	3875	3983	4107	4210	4289	4383	4480	4588	4659	4796	4875	5055	5178	5252	5331	5427	5518	5602	5727

< 表 6 >

文字像(Font image)코드 Table

文字像	文字像코드	받침數	받 침 Table																
가	3728	17	217	74	226	197	198	415	193	216	543	227	224	196	230	195	231	229	199
개	3746	9	217	226	9	198	193	216	227	224	196								
갸	3756	2	217	198															
개	3759	1	1																
거	3760	17	217	9	226	197	198	415	9	193	216	227	224	196	230	9	231	229	199
계	3778	6	226	198	193	227	224	196											
겨	3785	12	217	74	226	197	198	193	216	9	224	196	231	9					
계	3798	2	9	227															
고	3801	13	217	226	197	198	391	425	397	193	216	227	196	230	229				
과	3815	7	217	226	198	193	216	227	196										
괘	3823	6	217	226	198	193	216	196											
괴	3830	5	226	198	193	9	196												
교	3836	1	1																
구	3837	11	217	226	197	198	415	391	193	216	227	196	230						
귀	3849	3	226	198	9														
퀘	3853	1	9																
귀	3855	5	226	198	193	216	227	193	216										

文字像	文字像코드	받침數	받 침 Table						
기	3861	3	226	198	193				
그	3865	9	217	226	193	198	115	193	216 226 196
	9999	1	1						
기	3875	12	217	226	197	198	193	216	227 224 196 229 9 9
까	3888	10	217	74	226	198	193	216	224 196 230 231
깨	3899	8	217	226	198	193	216	227	224 196
가	3908	1	217						
~~~~~									
해	5677	2	217	227					
회	5681	3	217	227	196				
효	5685	1	1						
후	5686	6	217	226	198	429	193	227	
휘	5693	3	226	198	193				
혜	5697	3	9	226	196				
휘	5701	7	9	226	198	193	216	9	196
휴	5709	3	198	193	196				
흐	5713	10	217	226	197	198	415	193	216 227 196 231
희	5724	2	226	198					
히	5727	7	217	226	198	193	216	227	196

그런 다음 文字像코드 table을 Search하여 文字像코드 '4025'가 가질 수 있는 받침들 중 'ㄱ'의 코드 444가 몇 번째(n)인지 확인하여 n을 더 해주면 이 코드(4025+n)이 완전히 조합된 글자 '넋'字의 코드(4028)가 되는 것이다.

## (2) 1 Byte Data 處理 프로그램

### 1) 入力 Data의 取讀 및 分離

Read Buffer에 읽혀들어 온 Data들은 1 Byte 단위의 element 1개씩 處理된다.

Data의 분리과정에서 Data가 숫자, 특수문자, 영문 또는 한자인지를 확인하여 각각의 處理 Routine으로 가서 작업을 수행한다. 이 때 英文 및 漢字 Data의 식별은 入力時에 부여한 indicator에 의한다.

한글이면 한글組合 Routine에서 2 Byte 체계의 Data와 같은 方式으로 組合된다.

## 4. OUT PUT list 作成

Main Process를 거쳐서 받아낸 MT에는 2 Byte 形態의 該當 文字像 코드들이 들어있다. SP7 PRINT機에는 文字像 記憶裝置가 內裝되어 있어 文字像을 記憶하고 있는데 이것은 비영구적이거나 Power OFF하기 전에는 남아있다. 먼저 Font-Image (文字像)를 load한 후 위의 tape을 읽혀 프린트하면 해당 코드의 像을 찍어준다. 이 때 프린트方式은 기계적인 Impact Printing方式이 아니라 文字像을 OFT(Optical Fiber Tube)를 통해 感光시키는 光學的인 方式으로서 高品位의 文字像을 出力해 낸다.

## IV. 問題點 및 앞으로의 方向

本 시스템이나 Kips-2에서 해결되지 못한 問題點 중 가장 시급한 것이 入力問題이다. 먼저 한글 풀어쓰기時의 字版의 標準化가 되어있지 않으며, 漢字 入力時 現在 使用하는 Full Keyboard方式은 수천개의 字盤에서 글자를

일일이 찾아야 되므로 入力速度가 매우 느리다.

이러한 난점을 해결하기 위한 방편으로 한글漢字自動變換시스템 ( Kips - 3) 이 發表된바 있으나 아직 實用化되지 못하고 있다.

앞으로 한글漢字自動變換시스템의 完成 및 우리말의 自動索引開發에 박차를 가하여 實用的인 우리말 檢索시스템을 構築해야 할 것이다.

〈表 7〉 데이터의 處理過程

• Input Data의 形態( 1 byte Data )

한글의 入力形	한글漢字自動變換시스템
Keyboard 上의 文字	QHSDLSDMS VUDDIDDMF GKFNAJSWJ CNFQKFGKRH
入力字素別 코 드	DCECDECDE4EECCCCDC4CDCDCDEED4CDCDDCCDDC4 88243244205444944460726511261035682672980

• 組合處理後의 Data 形態

組合된 文字像 코 드	본	인	은	평	양	을	하	루	먼	저	출	발	하	고
2 bytes 文字 코 드	1C	1C	1B00	1C	13	1B00	1E	10	14	1E00	1C	19	1E	0D00
	17	31	3800	54	36	3900	5C	16	14	3700	45	13	5C	E900

• 出力例

- (A) 본인은 평양을 하루먼저 출발하고 후에 가족은 사리원까지 같이 내려오다가 해주로 들어가서 헤어짐.
- (B) 이화여대생의 종교적 신념체계에 관한 연구,, 梨花女大 韓國文化研究院論叢, 25,, 1975  
한국교회 학교성서교재 카리큘럼에 나타난 종교적 개념에 대한 연구,, 梨花女大 韓國文化研究院論叢,  
한국기독교의 사회윤리적 관심의 변화,, 梨花女大 韓國文化研究院論叢, 30,, 1977  
한국기독교교육의 사회윤리적 목적에 관한 한 價値理論的 研究,, 梨花女大 韓國文化研究院論叢, 31,