

정보 교환용 정음 부호세트의 코드 표준안에 관한 연구

김우선* 이강원*, 김철*

*용인대학교 컴퓨터정보학과

e-mail : swiri55@hanmail.net

A Study on Standardization of the OPA Character set for Information Interchange

, Woo-Sun Kim, *Gang-Won Lee*, Chul Kim*

*Dept. of Computer Science & Information Processing

Yong-In University

요약

본 논문은 현재 한글이 인터넷이나 컴퓨터에서 사용이 어렵고, 자판 구도가 불합리하며, 남북 및 세계 표준이 아직 정해지지 않고있어 정열방식이 불일치와 내부 처리 코드의 상이한 문제점을 가지고 있어 기존의 한글처리로는 부적합한 것들을 종합적으로 해결하기 위하여 새로운 방식을 모색하고, 미래의 방향을 제시할 목적으로 정음부호(OPA : Ortho Phonic Alphabet)의 코드 제안에 따른 기존 KS C 5601과 KS C 5700과의 코드 변환시 문제점과 OPA 표준화를 위해 이 연구를 수행한다.

1. 서론

한글은 자소나 자모, 마디글자까지 표시할 수 있는 양면성과 탁월한 유연성을 발휘할 수 있는 문자로서, 현재 세계공용어로 쓰이는 영어는 말과 글이 일치하지 않아 학습하기에 매우 어려우며 자모 중심이어서 한자나 일본어처럼 마디글자는 구분되지 않아 음절한계가 모호한 특성이 있는 반면에 한글은 세계 공용 문자에 적합하다고 할 수 있다.

정음부호란(OPA: Ortho Phonic Alphabet) 세계의 모듬말을 정음으로 적기위한 음성기호 체계이다. 이와 비슷한 음성기호체계는 현재 통용되고 있는 국제 음성기호이다. 국제 발음기호로 사용될 수 있기 때문에 사라져 가는 언어에 대한 기록수단으로 사용될 수 있다. 로마자나 국제음성기호로의 표기도 가능하지만 학습하기에 어렵고 불편하다. 따라서 국제정음 기호는 새로운 기록수단을 제공할 수 있다.

2. 본론

2.1 부호화 문제

컴퓨터가 영문을 사용하는 서구권에서 개발되어 왔으므로 부호체계도 서구문자를 다루기에 적합하게

개발되어 왔다. 따라서 이제까지 컴퓨터에서 한글을 다루는 부분에 대해서 여러 가지 문제들이 제기되어 왔다. 한글 부호는 2바이트를 사용하고 있으며 로마자의 ASCII는 7비트로 구성되어 있다. 그러나 한글 7 비트는 KS C 5700 부속서 4에 규정하고 있으나 일대일 대응이 되지 못하여 한글의 코드 변환이 어렵게 되어있다. 한편 '모듬이 한글꼴' 모듬은 2바이트 완성형과 조합형으로 구성되어 있다. 일본 JIS를 본뜬 구조이다. 완성형이든 조합형이든 국제 규격을 만족하지 못하고 있다.

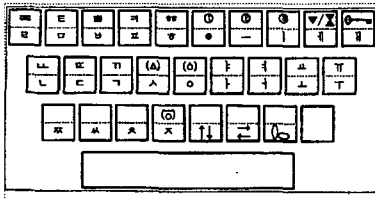
그 외 한글의 내부처리코드가 회사마다 또는 기종마다 틀려서 한글을 다른 프로그램을 통해 읽을 수 정하는 작업이 쉽지 않다. 보편 글 편지를 만드는데 있어서 한글의 내부처리코드의 상이함은 큰 걸림돌이 된다. 따라서 통일된 내부처리 코드가 필요하다

2.2 OPA 자판의 구도 및 코드

정음부호(OPA : Ortho Phonic Alphabet) 자판은 IPA 체계보다 훨씬 간편하고 익히기 쉬워서 우리나라 뿐 아니라 다른 나라 사람들도 편리하게 외국음을 표기할 수 있으며, 기호의 수를 52개 이내로 하여 현행

자판상에서 직접 타자할 수 있도록 하였다. 장점이 있다.

기존 KS C 5715와 국규 9265에 덧붙여 빈자리에 그 추가될 정음 기호를 끼워 넣어야 하지만 세계성을 고려할 때 여러 가지 방안이 있다. 그중 일관된 처리가 가능한 자판을 위해 기능키를 사용하는 방식을 제안하게 되었다. (그림 2)



< (인용 파상표 기능키를 이용하여 입력할 수도 있음을 나타냄) >

(그림 2) OPA 자판배치 제안

2.3 정음부호(OPA)의 7비트 N바이트 구성방법

한글 낱자 부호체계와는 다른 구조이고 자판에 규정된 타건과 1:1로 대응되게 하였다. (그림 2)의 OPA 자판배치(안)을 기준으로 하였다. 여기에서는 '뎨'와 '이음'이란 소리경계를 표시하기 위함이다. 아래 (표 1) 정음 문자의 7비트 부호틀(안)에서 보면 KS C 5601/부속서 4의 코드 1101101 위치의 '나'에 '뎨'를 코드 1101110 위치의 '내'에 '이음'을 정음기호 부호체계(안)에서는 대응시켜 놓았다.

예를 들면 ㄱ + 뎨 + ㅏ 는 '가'와 다르다. 'ㄱ사다'는 ㄱ + 이음 + ㅏ + 이음 + ㅏ 처럼 구성된다.

'N바이트 조합형'은 초성-중성-종성의 세 갈래로 음운을 나누는 글자는 아마 한글 이외에는 없을 것이기 때문이다.(죽은 글자로는 원나라의 파스파 문자가 3벌식 음운 체계를 가졌었다.) ISO-2022를 따르면서, 한 바이트당 94 코드점을 쓸 수 있다. 이 가운데 2비트(32코드점)는 초성/중성/종성/특수목적을 가리키는데 배정하면, 나머지 62 코드점을 쓸 수 있기 때문에, 중요한 옛 음운이나 한자/특수기호까지 포함시키는 것은 쉬운 일이다. (실제로 첫가끝 자모에 들어가 있는 상당수의 초성-중성 음운들은 중국 한자음을 음역할 때나 쓰는 '특수한' 자모로써, 일반적인 자모로 인정하기가 힘들다.)

(그림2) 의 자판배치 제안에서처럼 정음부호에 있는 'ㄴㄴ', 'ㄹㄹ'을 기존의 자판을 이용하여 입력하려면 두 번 타건해야 하기 때문에 2바이트가 입력되

어야 한다. 이것은 사용자로 하여금 혼동을 줄 수 있으므로 동일한 방식으로 하나의 자리를 차지하게 자판을 구성할 필요가 있다.

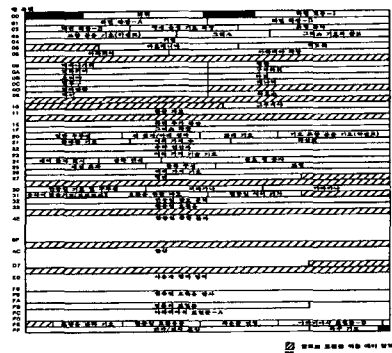
한글 낱자 부호체계 (KS C 5601/부속서 4)		정음부호 (안)		로마자 부호체계 (ISO 5446)										
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
가	나	다	라	사	하	자	차	카	가	나	다	라	사	하
...

(표 1) 정음 문자의 7비트 부호틀(안)

2.4 정음부호의 현재사용되는 코드테이블의 재배치

정음부호(OPA)는 KS C 5700의 한글 자모 테이블의 완성형 부호를 모두 조합할 수 있다. (표 3) 기존의 한글 자모로 되어 있는 자소형 부호계를 유일한 부호계로 두고 조합 방식을 사용하는 방법도 있겠지만 효율적인 활용을 위해서는 한글 완성형 부호계와 호환용 한글 자모 부호계를 없애 공간을 반납하고 정음부호(OPA)의 조합형식을 따라야 한다고 본다.

표2와 표3에서 보듯이 정음기호를 KSC 5601코드 KS C 5700의 테이블로 대치해보았다. KSC 5601에는 없는 'ㅎㅎ', 'ㄴㄴ', 'ㄹㄹ' 등의 문자를 임의대로 넣어 보았다. ISO 646의 코드값은 정음부호 코드값과 거의 일치하므로 이들 서로의 정음부호와 KSC 5601, KS C 5700 그리고 ISO 646의 코드로의 변환을 어떻게 변환할 것인지를 연구한다.



(그림 1) KS C 5700 BMP의 구조와 한글 자모

ㄱ	1001	ㅋ	1104	ㄴ	1002
ㄴ	1004	ㆁ	1106	ㄷ	1008
ㄷ	1007	ㆁ	110A	ㅁ	1013
ㄹ	1009	ㄴ	110C	ㅂ	1016
ㅁ	1011	ㄹ	1112	ㅅ	1019
ㅂ	1012	ㅌ	1113	ㅇ	1100
ㅅ	1015	ㅍ	1117	ㅈ	101F
ㅇ	1017	ㅡ	111A	ㅊ	1101
ㅈ	1018	ㅣ	111C	ㅌ	1118
ㅊ	101A	ㅑ	1107	ㅍ	1119
ㅋ	101B	ㅓ	1103	ㅇ	111C
ㅌ	101C			.	111E
ㅍ	101D				
ㅈ	101E				
ㅊ	1102				

(표 2) 정음부호의 KS 5601 대치 코드표

ㄱ	1100	ㅋ	1163	ㄴ	1101
ㄴ	1102	ㆁ	1165	ㄷ	1103
ㄷ	1103	ㆁ	1167	ㅁ	1108
ㄹ	1105	ㄴ	1169	ㅂ	110A
ㅁ	1106	ㄹ	116D	ㅅ	110D
ㅂ	1107	ㅌ	116E	ㅇ	1114
ㅅ	1109	ㅍ	1172	ㅈ	1158
ㅇ	110B	ㅡ	1173	ㅊ	1119
ㅈ	110C	ㅣ	1175	ㅌ	1141
ㅊ	110E	ㅑ	1166	ㅍ	1159
ㅋ	110F	ㅓ	1162	ㅇ	114C
ㅌ	1110			.	119E
ㅍ	1111				
ㅈ	1112				
ㅊ	1161				

(표 3) 정음부호의 ISO 10646-1 한글자모 TABLE 대치 코드표

2.5 OPA 입력기의 구현

컴퓨터 상에서 한글을 입력하기 위해서는 한글입력기라는 특수한 알고리즘이 필요하다. 한글의 경우 영문과 달리 코드를 조합하여 한 글자가 만들어지는 방식이기 때문이다. 예를 들어 "한"이라는 글자를 완성하려면, 영문 키에서 g-k-s 의 세 키를 눌러야 하죠. 이렇게 눌러진 g-k-s 키의 값들이 어떠한 알고리즘에 의해서 "한"이라는 글자로 변환되는 것입니다. 여기서는 Windows 의 네이티브 한글 입력기를 사용하지 않고 한글을 입력할 수 있는 한글 입력기 알고리즘을 구현해 보려 한다.

먼저 초성-중성-종성 형태의 한글 입력만을 위한 한글 입력기를 생각해 보면 다음 그림3과 같이 나타낼수있다.

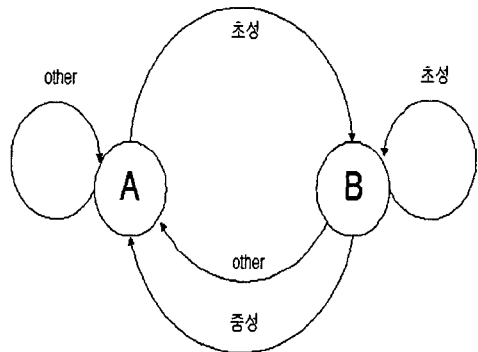


그림 3

그림3에서 A, B가 쓰여 있는 것은 현재 상태를 나타냅니다. 화살표는 현재 상태에서 발생한 사건(event)이고, 화살표가 가리키는 것은 그 사건에 대해서 옮겨지는 상태를 나타냅니다. 초기상태와 최종상태는 A이고 other는 나머지 사건(중성도 포함)들을 제외한 모든 사건을 나타낸다..

예제로 "가리라" 라는 글자의 입력 순서는 아래와같다.

가리라: A > ㄱ > B > ㄹ > A > ㄹ > B > ㅣ > A > ㄹ > B > ㅏ > A

"초성-중성"이 반복되었으므로 현재 상태는 계속 A>B>A>B>... 를 반복하게 된다.

여기서 한글 입력시에는 "조합중인 글자"와 "조합이

끝난 글자”의 차이가 있음을 유념해야 할 것이다.

우리가 한글을 입력할 때 조합중인 글자는 백스페이스로 글자 조합을 변경할 수 있어야한다.

그러나 이미 조합이 끝난 글자는 한 글자 전체가 삭제됩니다. 조합중인 글자는 대개의 경우 실제로 입력창에 삽입되지 않으며, 위의 그림3서 보았을 때 완전히 조합이 끝난 다음에 실제로 삽입됩니다.

위의 그림3을 구현한 코드를 알고리즘 화하면 다음과 같을 수 있다.

```

var
  상태 = A, 초;
// insert(x): 입력창에 x를 삽입한다.
// combine(a, b): 초성a,중성b를 조합한다.
procedure KeyEvent(입력키);
begin
  case 상태 of
    A: case 입력키 of
      초성:
        상태 := B;
        초 := 입력키;
      else
        insert(입력키);
      end;
    B: case 입력키 of
      초성:
        insert(초);
        초 := 입력키;
      중성:
        상태 := A;
        insert(combine(초, 입력키));
      else
        insert(입력키);
      end;
  end;
end;

```

2.6 OPA구현의 문제점

프로그램 구현시 문제가 되는 점은 지금은 사용되지 않는 정음부호(ㅎㅎ, 르르, ㄴㄴ, Δ, ㅎ, ㅇ, ·)를 어떻게 프로그램 할것인가 그들의 적절한 자판배치를 실험하여 자판구도를 다시 정의 해야한다는 어려움이 있다.

3. 결론

정음부호(OPA)의 자판배치(안)(그림 2)과 OPA 7비트 부호들(안)(표 1)을 KS C 5601/부속서4과 비교해 보면 기존 문자 입력시 2바이트가 입력되어야 하는 몇몇 문제점을 정음부호(OPA)를 통해서 해결하고 있다. 또한 (그림 1) KS C 5700의 BMP 구조에서 보면 한글 자모 뒷부분에 앞으로 표준을 위한 예비 영역이 충분히 있기 때문에 표준으로 정음부호(OPA)를 사용함에 있어서 어려움이 없다고 본다.

정음부호(OPA)는 훈민정음의 원리에 따라 음절자를 구성하므로 모든 부호계를 포함한다. 또한 문자 집합이 작으므로 국내외 규격을 만족하고, 모든 한글 부호계의 전체집합이므로 부호간 변환시 호환성이 완벽하게 보장된다. 또한 정음부호는 낱자소 조합으로 음절을 구성하기 때문에 현대 한글은 물론 옛 한글을 자연스럽게 지원한다.

훈민정음 원리에 따르는 정음부호를 기존의 한글처리로는 부적합한 것들을 종합적으로 해결하고, 국어 정보처리와 한글 문자 정보처리의 모든 응용 분야에서 만족할 수 있도록 하기 위해서는 표준화가 필요하다.

참고문헌

- [1] 박성득, “국제정음기호표준 제정의 타당성 연구”, 한국전산원, 1999.12
- [2] 한국표준협회, 정보교환용 부호 (한글 및 한자) KS C 5601 -1987 완성형, 1987
- [3] 한국표준협회, 정보교환용 부호 (한글 및 한자) KS C 5601 -1992 조합형, 1992
- [4] 한국표준협회, 국제 문자 부호계 (UCS : Universal Multiple-Octet Coded Character Set) KS C 5700 - 1995, 1995
- [6] 변정용, “국제 문자 부호계에 포함된 한글 부호계의 개선 방안”, '98 가을 학술발표논문집, 1998
- [7] 한글입력기 구현
<http://www.geocities.com/jounlai/prg/kor/haninput.html>