

Poorman's Hangul Jamo Input Method

pmhanguljamo.sty

Nova de Hi

2020/01/20 v0.2.1

```
\begin{jamotext}
g@/r@/mi p@/r@/ni sai de/ug h@i/o/,\\
moy/hi pe/re/h@/ni gos/ bi/ci bvr
    bvd/n@n d@s/do/da/.\\
ors bo/mi bon/d@in sdo di/na/ga/n@/ni \\
e/nv na/ri i do/ra/gar h@i/o/.
\end{jamotext}
```

구름미 푸르니 새 더욱 히오,
뫼히 퍼리히니 곳 비치 블 븐느 듯도다.
웁 보미 본딘 쏘 디나가느니
어느 나리 이 도라갈 히오.

차례

1	목적	1
2	사용법	2
2.1	패키지 사용 선언	2
2.2	명령과 환경	2
2.3	성조표지와 음절	3
2.4	알파벳 대응 문자	3
3	폰트	6
3.1	사용가능한 글꼴	6
3.2	폰트의 지정	7
4	贅辭	9
5	변경 이력	10
	부록	11

1 목적

pmhanguljamo.sty는 “Poorman's Hangul Jamo Input Method”에서 따온 이름이다. 알파벳과 몇 개의 기호문자만으로 한글 자모 (첫가끝) 입력을 대신할 수 있게 만든 것이다. 자모 조합 한글을 잘 처리할 수 있는 폰트의 도움을 받아 (3.1절을 보라) ‘모든’ 한글 수백만자를 표현할 수 있다. 한글 패키지 ko.TeX과 함께 쓸 수 있고, 단독으로 사용해도 한글 표현이 가능하다. 외국어를 본문으로 하고 한두 글자의 한글만을 표현할 필요가 있을 때, 별도의 입력기가 없는 상황에서 ‘옛한글’을 포함하는 문자와 문장을 출력하여야 할 때 쓸 수 있다.

X_YLaTeX, LuaLaTeX에서 사용할 수 있고 pdfLaTeX은 지원하지 않는다.

2 사용법

2.1 패키지 사용 선언

```
\usepackage{pmhanguljamo}
```

preamble에 위의 문장을 둔다. 패키지의 옵션은 따로 없다.

2.2 명령과 환경

\jamoword 명령

```
\jamoword{<several words>}
```

한글 자모를 알파벳으로 입력하여 출력한다. 하나 또는 둘 이상의 단어가 올 수 있다. (단 하나의 단어만을 처리하는 \jamotextcmd라는 명령이 있기는 하나 내부적으로 사용되는 것이다.)

```
\jamoword{na/ras;mar:ss@/mi;  
dyuq/guig;ei dar/a;}
```

나·랏·말·쑈·미 둥·꺠에 달·아

<jamotext> 환경

```
\begin{jamotext}  
<text>  
\end{jamotext}
```

<jamotext> 환경은 여러 문단을 포함하는 긴 문장을 식자할 수 있다.

```
\begin{jamotext}  
na/ras;mar:ss@/mi; dyuq/guyg;ey; dar/a;  
  
mun/jj@x;oa;ro; se/rv s@m@s/di;  
a/ni;h@r/ss@i;  
\end{jamotext}
```

나·랏·말·쑈·미 둥·꺠에 달·아
문·쫐·와·로 서르 수·똥·디 아·니·홀·썩

\jmcc 명령

```
\jmcc{<single jamo>}
```

음절이 아니라 한 개의 자모만을 표시할 목적으로 **한글 호환 자모**를 식자하려 할 때 사용하는 명령이다. 호환 자모를 \jamoword 명령의 인자 안에서 쓰는 방법은 2.4절을 보라. 이 명령은 본문 중에서 독립적으로 쓰기 위해 마련되었다. 인자로 단 하나의 자모만이 와야 한다.

```
\jmcc{BSG} \jmcc{@} \jmcc{UEY}
```

ㅂㅅㄱ, ㅏ, ㅕㅔ

2.3 성조표지와 음절

입력에는 로마자 알파벳과 @ 부호, 그리고 세 개의 성조표지(방점) 기호 /, ;, :를 사용한다.

성조표지가 와야 음절이 완성된 것으로 본다. 모든 한글 음절은 반드시 성조표지로 끝난다. 평성이 아닌 성조표지는 (적절한 폰트에서) 한글 음절의 왼편에 찍힌다.

표 1: 성조 표지

/	평성	점 없음
;	거성	점 하나 (·못노·폰 소·리)
:	상성	점 둘

성조 표기를 하지 않는 경우에도 음절의 완성을 나타내는 /를 두어야 한다. 다만 이 경우에 입력 편의를 위한 예외로서 그 뒤에 스페이스가 와서 단어 입력이 종료되는 위치(또는 입력 문자열의 마지막)에서는 /를 생략할 수 있다. 다음 예를 보아라.

<code>\jamoword{naix:jyuq/ G/ so/ri;n@n;}\</code>	:냉중 ㄱ 소·리·논
<code>\jamoword{naix:jyuq G so/ri;n@n;}</code>	:냉중 ㄱ 소·리·논

스페이스 앞에서 /를 생략할 수 있음과 관련하여 스페이스가 아니라면 생략할 수 없음에 주의하라. 예컨대 마침표와 같은 문장부호가 올 때는 그 문장부호 앞에서 /를 생략할 수 없다.

<code>\jamoword{jas ans bo/m@i pvr/oa na/mo/sbun gi/peys/do/da/.}</code>	жат 앓 보미 플와 나모쑤 기펫도다.
--	-----------------------

2.4 알파벳 대응 문자

알파벳 소문자를 이용하여 한글 자모 영역의 문자를 입력한다. 자음과 모음의 대응 규칙은 다음과 같다.

자음

표 2: 알파벳-자모 대응 규칙: 자음

ㄱ	g	ㄴ	n	ㄷ	d	ㄹ	r
ㅁ	m	ㅂ	b	ㅅ	s	ㅇ	x
ㅈ	j	ㅊ	c	ㅋ	k	ㅌ	t
ㅍ	p	ㅎ	h	ㅇ	q	ㅅ	z
ㅇ	f						

- 1) g, n, d, r, m, b, s, j, c, k, t, p, h는 연상이 가능할 것이므로 따로 설명하지 않는다. ㄹ에 대하여 r만을 사용하고 l을 쓰지 않는다.
- 2) ‘ㅇ’(이음)에 대하여 x를 채택하였다. 원래 이것은 오래 전 frktex이라는 패키지에서 첫소리 이음에 대하여 x를 썼던 적이 있으므로 그 유래가 아주 없다고 할 수는 없겠다. 다만 이 패키지는 종성 이음 받침도 x를 쓴다.
- 3) ‘ㅇ’(옛이음)을 q로 한 것은 역시 frktex에서 온 것인데 “꼭지 달린 이음”이라는 글자 모양이 q를 연상시킨다”는 이유였다고 한다. 그럴 법도 하다.
- 4) ‘ㅅ’(반시옷)을 z로 한 것은 납득할 수 있다고 본다. z가 s의 유성음에 대응하기 때문에.

- 5) 가장 곤란했던 것은 ㅎ (여린히읇)이었다. 사용되지 않은 자판 가운데서 선택할 수밖에 없는데 l, v, w, f 정도가 선택 가능한 범위였다. 알파벳이 아닌 부호문자 중에서 선택하는 것도 생각해보았으나 역시 자판의 윗줄로 올라가는 것은 꺼려져서 f로 선택하였다. 순전히 편의에 의한 대응이다.
- 6) ‘ㅅ’나 ‘ㅆ’와 같은 특별한 글자, ‘ㅉㅉㅉㅉㅉㅉㅉㅉ’의 초성 글자들, 즉 훈민정음 언해의 “둥·꺾소·리·엿·니쑈·리” 표기에서 쓰이는 齒頭·正齒音을 구분하기 위한 글자들은 s, j, c에 l, lr을 붙여서 표기한다. l이 왼쪽(치두음), lr이 오른쪽(정치음)이 더 길어지는 글자가 된다.
- 7) ‘ㅃ’, ‘ㅍ’, ‘ㅍ’, ‘ㅍ’와 같은 “입시·울가·비야·불 소리” 표기는 x를 이어써서 표기한다. 각각, bx, bbx, mx, px.
- 8) 겹자음·복자음은 이어쓴다. gg ㄱㄱ, bsg ㅃㅅ, bsd ㅃㅈ, rf ㄹㅍ.
- 9) ‘제로자음’인 초성 이응은 생략할 수 있다. 모음으로 시작하면 초성 이응을 식자해준다. \jamoword{an} 안. 만약 정말로 초성을 비우기를 원한다면(즉 초성 필러를 쓰려 한다면) w를 그 위치에 쓰도록 하라. 이것은 종성에는 적용되지 않는다. \jamoword{wan} ㄴ.

모음

표 3: 알파벳-자모 대응 규칙: 모음

ㅏ	a	ㅓ	e	ㅗ	o	ㅜ	u	ㅡ	v
ㅣ	i	ㅕ	@						
ㅑ	ya	ㅑ	ye	ㅛ	yo	ㅠ	yu	..	@@
ㅓ	ay	ㅓ	ey	ㅖ	oy	ㅜ	uy	ㅜ	vy
ㅕ	yay	ㅕ	yey	ㅙ	yoi	ㅠ	yui	ㅜ	@i
ㅗ	oa	ㅗ	oay	ㅜ	ue	ㅜ	uey		

- 1) 기본 모음 ㅏ, ㅓ, ㅗ, ㅜ, ㅡ, ㅣ, ㅕ (@)는 한 글자로만 표기한다. ㅑ, ㅑ, ㅛ, ㅠ는 y를 앞에 붙인다.
- 2) ‘ㅑ’를 영문자의 자음에 해당하는 v에 대응시킨 것은 이상하게 보일 수 있다. 그러나 윗 항을 충실하게 지키려다 보니 부득이했다. 실로 이 홀소리 글자의 로마자 표기는 혼란스럽다고 생각하고 있는데, 표준인 eu만이 아니라 y, u 등이 있었고, 다른 어떤 제안에서는 w도 있었던 것으로 기억한다. 어차피 마땅한 방법이 없다면 이렇게 약속하고 쓰는 것이 한 방법이라 생각한다. v자를 죽 늘리면 ‘ㅑ’가 되는 것처럼 보이는 연상에서 비롯되었는지 모른다.
- 3) e를 보통 ‘에’라고 읽기 때문에 ‘ㅓ’를 e에 할당한 것이 익숙지 않을 수 있다. 이것은 Yale 표기법에 그 기원이 있다.
- 4) ‘ㅓ, ㅓ’는 단모음으로 간주되기 때문에 ay, ey라고 쓰도록 하였다. 비슷한 이유에서 oy, uy라고 되었다. 그러나 이런 음성학적 지식이 표기에 반드시 필요하다면 그것도 불편하다. 그래서 ‘ai, ei, oi, ui’라고 입력하는 것도 허용하였다. ‘ㅓ’의 경우에도 vy와 vi 둘 다 가능하다.
 \jamoword{sai/say} 새새. \jamoword{sei/sei} 세세.
 \jamoword{soi/soy} 쇠쇠. \jamoword{sui/sui} 쉬쉬.
 \jamoword{gvi/gvy} 괴괴.
 그러나 ㅓ는 단모음으로 인식될 가능성이 없다고 보아 @i만을 취하였다.
 \jamoword{g@i} 괴.
- 5) ‘쌍아래아’는 현대의 제주어 표기에서 이따금 필요하다. 모양대로 @@로 적으며 y@를 취하지 않았다. 현대 국어에서 이 글자의 소리는 잊혀진 것이라서 소리로 연상할 수 없기 때문이다.
 \jamoword{@@/nam/vn} ㄴ남은.

- 6) 표준 로마자 표기법에서 wa, wo, we, wi로 표기할 때 나타나는 반자음 w은 채택하지 않았다. ‘ㅜ, ㅠ, ㄱ’는 생긴 대로 oa, ue, ui로 적는다.
- 7) 이 패키지는 Unicode 4.0의 한글 자모 보충 A, B를 지원한다. 그러므로 한글 자모 영역([U+11XX])에는 없는 ‘ㅚ’와 같은 모음을 표기할 수 있다. \jamoword{sa/guye} 사귀. 물론 폰트가 이를 지원해야 한다.

호환자모 음절

\jamoword나 <jamotext> 범위 안에서 호환 자모를 식자하려면 해당하는 자모를 대문자로, 하나의 음절로 입력한다. 별도로 \jmcc를 쓰지 않는다. 그러나 반드시 하나의 자모가 한 음절이어야 한다.

\jamoword{B/n@n; ib/si/ur;sso/ri;ni;} 바·는 입사·울쏘·리·니

\jamoword{孟m@ix/子j@/I/ g@r/@/sya/d@i}孟밍子즈 | 굴ㄴ샤디

호환 자모 아래아 ‘ㆍ’를 \jamoword 인자 안에서 식자하려 할 때는 자모 아래아 ‘ㆍ’와 구별하기 위해 대문자 W로 표기해야 한다. W와 WI만이 지원된다.

\jamoword{W/n@n; 呑t@n/D/字/jj@x; ga/on;d@is;so/ri; g@;t@;ni/ra;.}

ㆍ는呑툰ㄷ字·죡 가·온·딛소·리·ㄱ·투니·라.

한자와 음절 한글

\jamoword 명령과 <jamotext> 환경 안에 오는 인자는 다음과 같은 문자에 대하여 예약이 되어 있다.

- 알파벳 문자 — 한글 자모 문자에 대응한다.
- ;, :, / — 이 세 개의 부호는 성조표지, 음절완성을 표시하기 위하여 예약되어 있다.
- | — 이 한 개의 부호문자는 내부적으로 사용하도록 예약되어 있다.

이를 제외한 문자는 입력한 대로 찍힌다. 즉 한글 완성형 음절 문자나 한자 및 숫자와 문장부호가 모두 그대로 출력된다. 콜론, 세미콜론, 슬래시는 각각 \ColonMark, \SemiColonMark, \SlashMark 명령을 사용할 수 있게 해두었다.

\jamoword{15 세기 표기\ColonMark{} 國之語
音, 우리나라의 말
이 \$**\Rightarrow**\$ na/ras/mar/ss@/mi}

15 세기 표기: 國之語音, 우리나라의
말이 ⇒ 나랏말싸미

단일 매크로 자체는 그대로 출력된다. 그러나 인자를 취하는 매크로의 인자로 로마자 알파벳이 들어가는 경우에는 이 로마자를 자모 한글로 처리하려 하기 때문에 오류가 발생한다. 알파벳 포함 인자를 취하는 매크로는 이 범위 안에 넣지 않는 것이 좋다.

음절문자뿐 아니라 자모 한글의 입력도 그대로 출력된다. 어디선가 옛한글¹ 텍스트를 복사해와서 넣을 때 유용할지도. 다음 예에서 이 패키지의 입력 방식과 첫가끝 입력 방식이 혼용되어 있는 것을 볼 수 있다.

\jamoword{(/du/si/en/hai dvx/go/)부루미
썩르며 하눌히 눕고 나뵈 뒷푸라미 슬프니}

(두시언해 등고) 부루미 썩르며 하눌히
눕고 나뵈 뒷푸라미 슬프니

¹1933년 이전 표기법의 한글을 가리킨다. 유니코드의 한글 음절 문자 영역에서 표현되지 않는 한글을 통칭하는 것이나 현대 한국어 중에서 제주어를 기록할 때도 쓰이고, 워드 프로세서 한글의 이름을 적을 때도 쓰이기 때문에 그리 ‘옛’ 것은 아니다. 이 글에서는 관행을 따라 옛한글이라고 부르기로 한다.

3 폰트

3.1 사용가능한 글꼴

자모 조합 한글을 출력하려면 이를 해낼 수 있는 글꼴이 필요하다. 소위 ‘옛한글’ 글꼴이라지만 PUA 옛한글 5000여자의 사용자 영역 글자만을 가지고 있는 글꼴로는 이 패키지에서 말하는 옛한글을 식자할 수 없다. 새바탕·새굴림과 같은 한양 옛한글 폰트, 전주완판본체, 제주폰트 등이 모두 옛한글에 사용할 수 없는 폰트에 해당한다.

자모 조합 한글을 구현할 수 있는 몇몇 글꼴을 테스트해보겠다.

```
sa:r@/ma:da; h@i:xye; su:bx; ni/gye;
```

함초롬 바탕

:사롬:마:다 :히:예 :수:비 니:겨

은 바탕

사롬따다 히예 수비 니겨

본명조 또는 Noto Serif CJK

:사롬:마:다 :히:예 :수:비 니:겨

나눔명조옛한글

:사롬:마:다 :히:예 :수:비 니:겨

KoPubWorld 바탕

:사롬:마:다 :히:예 :수:비 니:겨

함초롬 돋움

:사롬:마:다 :히:예 :수:비 니:겨

본고딕 또는 Noto Sans CJK

:사롬:마:다 :히:예 :수:비 니:겨

나눔바른고딕옛한글

:사롬:마:다 :히:예 :수:비 니:겨

맑은 고딕

:사롬:마:다 :히:예 :수:비 니:겨

KoPubWorld 돋움

:사롬:마:다 :히:예 :수:비 니:겨

옛한글을 무리없이 표현하는 것은 현재 이 정도인 것 같다. 각 폰트는 특정한 문자에 대해서 약간의 차이를 보이기도 한다. 참고로 이 문서에서는 본문과 옛한글에 Noto Serif CJK KR을 사용하였다.

LuaTeX으로 컴파일하는 경우에는 위에 든 폰트들, 즉 Script Hangul 속성을 가진 폰트만 사용 가능하다. LuaTeX과 이 패키지의 관계에 대해서 11페이지를 참고하라.

한글 자모 영역의 글리프를 가지고 있는 글꼴이 있다. 성조표지를 잘 처리하지 못하는 경우가 있고 옛한글 글자꼴이 반듯하지 않지만 어쨌든 글자가 나오기는 할 것이다. 이것은 Script Hangul 속성을 갖고 있는 것은 아니며 단지 한글 자모 영역 글리프를 영리하게 배열한 것이다. 방점 없이 테스트해보겠다.

은 자모바탕

사롬따다 히예 수비 니겨

은 돋움

사롬따다 히예 수비 니겨

그러나 자모 영역 자면이 있다 하더라도 글리프가 잘 구성되어 있지 않은 글꼴에서는 다음과 같이 모아쓰기가 되지 않는 형태로 출력된다. 이런 글꼴은 옛한글에 사용할 수 없다.

전주완판본 순L

사르、口마다 ㅎㅇㅇㅋ 수빙 | 니겨

현대 한글 음절문자는 적어도 $X_{\text{La}}\text{TeX}$ 에서는 모두 제대로 표현될 것이다. 자모 코드로부터 한글 음절을 (유니코드 표준을 따라) 구성하는 일을 폰트가 아니라 $X_{\text{La}}\text{TeX}$ 이 하고 있음을 짐작할 수 있다.² 현대 한글만을 표기하려 한다면 폰트의 제약이 줄어들 것이라고 하겠다.

```
o/dvx/vn ja/ey a jo/sen/vi dog/rib/gug/im/goa
```

백묵 바탕

오등은 자에 아 조선의 독립국임과

Adobe 명조

오등은 자에 아 조선의 독립국임과

전주완판본 순L

오등은 자에 아 조선의 독립국임과

제주한라산

오등은 자에 아 조선의 독립국임과

3.2 폰트의 지정

특정 언어 패키지나 입력기의 제약 없이 한글 글꼴을 지정하는 것만으로 현대 한글과 옛한글을 모두 표현할 수 있다는 것이 `pmhanguljamo`의 장점이다. 다만 `ko.TeX` 없이는 글자 사이의 행나눔이나 양끝맞추기 등 한글 문장 식자에 꼭 필요한 몇 가지 기능이 구현되지 않을 것이므로 이런 점은 감안하여야 한다. 대체로 몇 자 정도의 한글을 넣어야 하는 때에 유용하리라 본다.

`ko.TeX` 패키지와 함께 쓸 때와 그렇지 않을 때를 나누어서 간단히 사용법을 보이겠다.

`ko.TeX` 문서

만약 `ko.TeX` (`xetexko` 또는 `luatexko`)과 함께 사용하는 경우라면 본문 한글 글꼴을 다음과 같이 지정한다. [`Script=Hangul`] 옵션을 반드시 주어야 한다는 점을 기억하자.^{3, 4}

```
\setmainhangulfont{Noto Serif CJK KR}[Script=Hangul]
```

본문과 폰트를 달리 쓰려 한다면 `\jamoword` 또는 `<jamotext>` 범위에 다음과 같이 임시로 한글 글꼴을 적용한다.

```
\hangulfontspec[Script=Hangul]{HCR Batang LVT}
```

²이 패키지의 입력 방법으로 $X_{\text{La}}\text{TeX}$ 컴파일하여 만들어진 현대 한글 음절 영역 문자를 pdf로부터 복사하여 편집기에 붙여넣기하면 자모 코드가 아니라 음절 코드로 들어가는 것을 확인할 수 있다.

³이것은 폰트가 가지고 있는 오픈타입 속성을 활성화하여 자모 한글을 제대로 출력하도록 지정하는 역할을 한다. 자모 한글을 처리하는 기능을 가지고 있지 않은 글꼴에서는 아무 역할도 하지 않고 컴파일 시 콘솔에 경고 메시지를 보일 것이다.

⁴`LuaTeX`에서는 약간의 추가 옵션이 더 필요할 수 있다. 이 문서의 이후 [`Script=Hangul`]을 주라고 하는 때에 모두 동일하다. ¹¹페이지의 부록 B를 보라.

`\adnochangulfont` 명령은 이와 동일한 명령이다. 이 폰트의 적용 범위를 설정하거나 하는 것은 문서작성자에게 맡겨져 있다.

ko.TeX이 아닐 때

ko.TeX 문서가 아닐 때는 `fontspec` 패키지를 이용한다. `main font`를 위의 글꼴 중의 하나로 하거나(이 때에도 `[Script=Hangul]` 옵션을 반드시 지정하여야 한다.), `\newfontfamily`로 글꼴 명령을 만들어 쓰거나, 아니면 한글이 필요한 부분에만 `\fontspec`으로 글꼴을 지정해주는 것으로 될 것이다. 그림 1은 `article` 문서의 한 예이다.⁵

`polyglossia`와 함께 사용하는 것도 잘 된다.⁶ 다음은 이 경우의 한 설정 방법이다. (이 보기에서는 한국어를 `other language`로 설정하고 있는데 만약 `main language`로 설정한다면 `<korean>` 환경 안이 아니라도 한글 폰트가 적용된다.) `polyglossia`를 사용하여 `<korean>` 환경에서 쓰게 되면 한국어 조판에 필요한 상당한 지원이 이루어진다.

```
\documentclass{article}
\usepackage{polyglossia}
\setmainlanguage{english}
\setotherlanguage{korean}
\newfontfamily\hangulfont{UnBatang}[Script=Hangul]
\usepackage{pmhanguljamo}

\begin{document}
\begin{korean}
\jamoword{han/gug/e/rvr han/gvr/ro}
\end{korean}
\end{document}
```

만약에 `<korean>` 환경이 시작되면 그 안에서는 (무조건) `pmjamo` 입력 모드가 되도록 하려면 다음 코드를 `preamble`에 두면 된다.

```
\usepackage{etoolbox}
\apptocmd\korean{\begin{jamotext}}{}{}
\pretocmd\endkorean{\end{jamotext}}{}{}

```

이제 다음과 같이 간단히 입력할 수 있다.

```
\begin{korean}
na/ras/mar/ss@/mi
\end{korean}
```

그러나 위의 처방은 이 환경 안의 모든 로마자가 한글이라는 확신이 있을 때만 적용해야 할 것이다. `<korean>` 환경 안에서 로마자를 입력해야 할 일이 있을지 모르기 때문이다.

⁵ 폰트에 부여하는 옵션에 관하여 각주 4를 볼 것.

⁶ `polyglossia`는 `XLaTeX`에서만 사용 가능하다.


```

\documentclass{article}
\usepackage{fontspec}
\newfontfamily\hngljmfont{HCR Batang LVT}[Script=Hangul]
\usepackage{pmhanguljamo}
\newcommand*\jmhangul[1]{%
  \begingroup
  \hngljmfont\jamoword{#1}%
  \endgroup
}

\begin{document}
Say `Hello' in Korean,
\jmhangul{an/nyex/ha/se/yoy}.
\end{document}

```

그림 1: article에서 pmhanguljamo을 쓰는 예

4 贅辭

이 패키지를 만들게 된 사연이 이러하다. Kubuntu 리눅스에서 옛한글이 사용되는 문서를 작성할 일이 있었다. fcitx 입력기가 기본으로 설치되어 있었는데 옛한글 입력이 되는 입력기를 설정하는 문제로 골머리를 썩이는 경험을 하게 된 것이다. 세벌식 자판 배열은 이미 잊은 지 오래. nabi는 크롬이나 QT에서 뭔가 이상하게 동작하거나 아예 한글 입력이 안 되거나……. \LaTeX 문서인데 굳이 입력기가 필요한가 생각이 들어 간단히 로마자 대응 입력(transliteration)으로 한글과 옛한글을 입력하도록 하면 이 문제로 시간낭비하지 않아도 되지 않을까 싶어 시작하게 된 일이었다. (그리고 입력기 설치보다 더 많은 시간과 커⁷를 소비하였다.)

옛한글 입력과 관련하여, Mac OS에서는 구름 입력기가 있더라. 한글 Windows 10은 시스템 자체에 옛한글 입력기가 있는 것 같고 여전히 새나루나 날개셋도 있다. 그러나 어떤 것이든 시스템 설정을 바꾸거나 새로운 앱을 설치하거나 해야 하는 것으로 귀찮음을 감수할 수밖에 없다.

만들어놓고 보니 아예 한글 입력 자체가 불가능한 상황, 즉 시스템 설정을 건드릴 수도 없고 새로운 프로그램을 설치할 수도 없거나 한글 입력기 자체가 없거나, 서버에 접속하여 문서를 작성하는 상황이라도 한글과 옛한글을 포함한 문서를 작성할 수 있게 되었다. 실용적으로 쓸 일이 있을까 싶기는 하지만 버리기도 아까워서 기능을 보완하고 정리해두기로 하였다.

이 “한글 대응 입력” 방식이라는 것은 나름 그 유래가 있다. Yudit이라는 편집기의 옛한글 입력 방식이 그러하고 frktex에서도 역시 이런 식으로 한글을 입력했다. 한글 로마자 입력기(워드 프로세서에 내장된 것도 있고 시스템 입력기의 입력 방식 중의 하나인 경우도 있다)들은 특히 eu, eo 모음 때문에 너무나 많은 키를 눌러야 해서 내가 그리 좋아하지 않았지만.

그리고 옛한글 입력기가 있다 하더라도 두벌식 옛한글 입력이 필연적으로 겪게 되는 음절 경계의 문제는 피할 수 없다. 모음과 모음 사이에 최대 6개의 자음이 올 수 있는 조건에서 음절 경계의 판별을 완전히 자동화하는 것은 불가능하다고 생각한다. 이 패키지에서도 그런 문제를 고려하여 처음부터 음절 경계를 입력하도록 하였다. 약간의 번거로움은 피할 수 없지만 그 때문에 구현이 아주 간단해졌고 예외 처리를 위한 고민을 해소할 수 있었다. 이와 관련하여 종성을 대문자로 입력하는 방안(Yudit 방식)을 유력하게 고려하였지만 쉬프트 누르기를 싫어하는 나는 이를 채택하지 않았다. “쉬프트를 누르느니 슬래시를 한 번 치고 말겠다.”

이 패키지를 만들다보니 참 세상 편해졌다는 생각을 안 할 수가 없었다. Expl3는 내가 원하는 일을 너무나

⁷이 표기를 이태준 (1940), 《文章講話》(경성 종로, 文章社), p. 23에서 보았다. ‘ㅍ’가 아니라 ‘ㅍ’이다. 부록의 그림 2 참조. 인쇄 상태로 미루어보면 ‘ㅍ’자 활자가 있었던 것 같지는 않고 ‘ㅍ’를 식자한 후에 원편에 조그마한 고리를 달아 인쇄한 것으로 보인다.

간단하게 구현할 수 있게 해주었다. $\text{Xe}_{\text{L}}\text{TeX}$, $\text{Lua}_{\text{L}}\text{TeX}$ 엔진의 유니코드 처리 기능은 한글 사용자에게 참 많은 혜택을 주었다고 생각한다. 십수 년 전에 옛한글 폰트와 구현을 만들려고 했던 여러 시도를 기억하는 입장에서 당시 고민했던 문제가 말끔하게 해결되어 있는 현재 상황이 믿어지지 않을 정도이다. 모든 한글을 잘 표현하는 좋은 폰트를 가지게 된 것은 무엇보다 반가운 일이다. 그런 폰트가 더 많아지기를 바란다.

5 변경 이력

- v0.1 (2020/01/17): 첫 릴리스.
- v0.1.1 (2020/01/18): 호환자모 ‘\’ 입력 버그 수정.
- v0.1.2 (2020/01/18): $\text{Lua}_{\text{L}}\text{TeX}$ 컴파일 관련 수정. 문서 갱신.
- v0.2.0 (2020/01/19): pmhanguljamo로 명칭 변경.
- v0.2.1 (2020/01/20): \ColonMark bug fix. thanks to yihoze.

부록 A. 이 문서 컴파일에 필요한 폰트

이 문서의 한글과 한자 부분에 다음과 같은 폰트를 사용하였다. 모두 설치되어 있어야 컴파일할 수 있다.

Noto CJK Noto Serif CJK, Noto Sans CJK, Noto Sans Mono CJK 폰트. [Google Noto CJK](#). Adobe사의 [본명조\(Source Han Serif\)](#) · [본고딕\(Source Han Sans\)](#)과 거의 같다.

함초롬 LVT 함초롬 바탕 LVT, 함초롬 돋움 LVT. 한글과컴퓨터 & 한국텍학회. [KTUG:함초롬체 LVT](#).

나눔 옛한글 나눔명조 옛한글, 나눔바른고딕 옛한글. [네이버 한글한글아름답게](#).

KoPubWorld KoPubWorld 바탕, KoPubWorld 돋움. [한국출판인회의](#).

맑은 고딕 Windows 8 이후 한글 Windows 폰트. 마이크로소프트.

은 글꼴 은 바탕 T_EX Live 배포 서체.

X_YT_EX으로 컴파일하는 상황에서는 다음 폰트들이 더 사용되었다.

은 글꼴 은 자모바탕, 은 돋움.

백묵 글꼴 백묵 바탕. T_EX Live 배포 서체.

전주완판본체 전주완판본 순L. [전주시청](#).

제주서체 제주한라산. [제주특별자치도](#).

Adobe 명조 Adobe Reader 배포 서체.

부록 B. Lua_{HB}T_EX 컴파일을 위한 주석

Lua_{HB}T_EX으로 컴파일하는 경우를 위한 메모.

우선 Lua_{HB}T_EX의 \LaTeX 포맷을 이용하는 것이 좋다. harfbuzz 라이브러리를 활용하므로 더 바람직한 결과를 얻을 수 있을 것으로 기대한다. 이 패키지는 Lua_{HB}T_EX 1.11.2를 기준으로 테스트하였다. T_EX Live 2019에 있는 1.10에서의 올바른 동작은 보장하지 않는다. 이 때에는 X_YT_EX을 쓰도록 하라.

폰트 설정 시에 옵션으로 다음을 부여한다. 아래 RawFeature 부분은 Lua_{HB}T_EX이 아니면 에러를 토할 것이다.

```
[Script=Hangul,RawFeature={mode=harf}]
```

이 설정은 luatexko와 함께 쓰는 경우든 아니든 필요하다.

Lua_{HB}T_EX에서 이 패키지와 함께 사용할 수 있는 폰트는 Script Hangul 속성을 가진 폰트로 제한된다. 이 속성을 가지지 않은 일반적인 폰트로는 자모 입력으로부터 한글 음절 문자를 얻어낼 수 없는 것으로 보인다. [3.1절](#)에서 어떤 폰트가 사용가능한지 확인할 수 있다.

부록 C. 예제

훈민정음 어제 서문을 입력한 예를 보여두겠다.

```
\begin{jamotext}
  世syeyx;宗joq/御ex;製jyeyx;訓hun;民min/正jyeq;音fvm/
  na/ras;mar:ss@/mi; 中dyuq/國guig;ei; dar/a;
```

```

문mun/字jj@x;oa;ro; se/rv s@/m@s/di; a/ni;h@r/ss@i;
i;ren jyen/c@;ro; e/rin; 百b@ig;姓syeq;i; ni/rv/go;jye; horf; bai;
i/sye;do; m@/c@m;nai: jey bdv;dvr; si/re; pye/di;
mod:h@rf no;mi; ha/ni;ra;.

nai; i;r@r; 爲uix;h@;ya; e:yes/bi; ne/gye;
sai;ro; sv;mvr; ye/dvrb; 字jj@x;r@r; m@iq/g@;no/ni;
sa:r@m/ma:da; h@i:xye; su:bxi; ni/gye;
nar;ro; bsu;mey;
便bbyen/安fan/kvi; h@/go;jye; h@rf sd@/r@/mi;ni/ra;.
\end{jamotext}

```

世·생宗중御·영製·정訓·훈민민正·정음음

나·랏·말·ㅁ·미 中·등國·국·에 달·아 文·문·字·종·와·로 서·르 스·못·디 아·니·홀·씩 ·이·런 전·초·로 어·린
百·백·姓·성·이 니·르·고·져 ·흙·배 이·셔·도 무·춤·내 제 ·쁘·들 시·러 퍼·디 ·물·흙 ·노·미 하·니·라.

·내 ·이·를 爲·왕·하·야 ·어·엿·비 너·겨 ·새·로 ·스·를 여·들 字·종·를 밍·그·노·니 ·사·름·마·다 ·히·여
·수·비 니·겨 ·날·로 ·뿌·메 便·뻔·安·한·키 하·고·져 흙·쓰·르·미·니·라.

부록 D. 유니코드의 한글

이 글에서 사용한 다음 용어를 설명한다.

한글 자모 유니코드 블록 [U+1100]–[U+11FF]에 정의된 한글 자모 (*Hangul Jamo*).

한글 자모 확장 A 유니코드 블록 [U+A960]–[U+A97C]에 정의된 한글 자모 (*Hangul Jamo Extended-A*).

한글 자모 확장 B 유니코드 블록 [U+D780]–[U+D7FB]에 정의된 한글 자모 (*Hangul Jamo Extended-B*).

한글 음절 문자 유니코드 블록 [U+AC00]–[U+D7A3]에 정의되어 있는, 한글 음절 문자 (*Hangul Syllables*, Precomposed syllables). 여기에 정의되어 있는 11,172개의 음절 문자는 한글 자모의 조합으로 구성할 수 있어야 하고, 이 문자 각각을 완성형 음절 문자로 간주할 수 있다. 한글 폰트는 이 영역의 글리프만을 가지고 있는 경우가 많고 현대 한국어 문자 생활에서 이 글자만으로 대부분 충족된다.

한글 호환 자모 유니코드 블록 [U+3131]–[U+318E]에 정의되어 있는 한글 자모 (*Hangul Compatibility Jamo*). KS X 1001:1998 표준과의 호환을 위하여 마련된 영역인데 여기에 나오는 한글 자모는 조합에 활용하지 않으며 각 자모를 독립된 문자로 간주한다.

완성형과 조합형 1980년대와 90년대의 이른바 **조합형·완성형 논쟁**은 한글 전산화, 한글 코드의 표준 제정을 둘러싼 논쟁이었다. 유니코드 2.0 이후 이것은 더이상 ‘논쟁’이 아니게 되었다. 오늘날 ‘완성형’이란 한글 표현에 한글 음절 문자 영역의 코드를 사용한다는 의미에 가깝고, ‘조합형’이란 한글 자모 코드를 사용하여 한글을 표현하는 것을 의미하게 되었다. 예를 들어 “한글과 텍”이라는 문자를 음절 문자 코드로 표현하면

[D55C] [AE00] [ACFC] [0020] [AC4D]

처럼 되는데 이것을 자모 조합 코드로 표현하면

[1112] [1161] [11AB] % 한
[1100] [1173] [11AF] % 글
[1100] [116A] % 과

[0020] % space
[1110] [1166] [11A8] % 텍

로 된다. 특히 자음은 초성과 중성에 서로 다른 코드가 할당되어 있어서 이를 두고 “첫가끝(LVT)” 코드⁸라고 부르기도 한다. 자모 조합 코드는 음절 문자 코드에 비하여 몇 배 이상의 정보량을 요구하는 단점이 있으나 (UTF-8로 인코딩하면 그 격차는 더 벌어진다. UTF-8의 3바이트면 한글 음절 문자 한 자를 표현할 수 있지만 초중종성이 모두 있는 글자라면 무려 9바이트가 필요하다.) 음절 문자로는 표현할 수 없는 한글을 기록하기 위해서는 이 방법밖에 없다. 그래서 이른바 ‘옛한글’을 첫가끝 코드로 표현하는 것이다.

폰트 기술의 발전 코드가 그렇게 할당되었다 하더라도 이를 출력(표시)하는 것은 또다른 문제이다. 폰트라는 관점에서 생각하였을 때 한글 음절 영역 코드라면 각 코드에 글리프를 할당하면 되므로 비교적 간단한 편이고 이것이면 문자 생활에 큰 불편이 없기 때문에 대부분의 한글 폰트가 이런 정도의 기능만을 가지고 있다.⁹ 그런데 자모 조합의 경우에는 여러 코드를 조합하여 이를 한 음절로 또는 한 음절처럼 표현하는 능력을 폰트가 갖추고 있어야 한다. 이 글의 본문에서 간간이 “Script Hangul 속성을 갖춘 글꼴”이라는 표현을 썼는데 자모 조합 코드로부터 한글 음절을 식자하는 것이 가능한 폰트를 말하는 것이었다. GSUB/GPOS라는 트루타입 오픈의 속성을 이용하여 이것이 가능하게 된 것은 21세기 들어서의 일이다. 이것을 거의 최초로 가능하게 하였던 글꼴인 “은 바탕”¹⁰의 역사적 의미는 실로 적지 않다 하겠다. 그 후로 함초롬, 본명조와 같은 좋은 폰트들이 많아졌고, 이제 우리는 고품위의 “한글 표현에 제약이 없는” 훌륭한 자유 폰트를 여럿 가지게 되었다.

텍 엔진의 발전 이와 더불어 $\text{T}_{\text{E}}\text{X}$ 에는 $\text{X}_{\text{L}}\text{T}_{\text{E}}\text{X}$ (그리고 $\text{LuaT}_{\text{E}}\text{X}$)의 새로운 시대가 열렸다. 유니코드를 자유롭게 다룰 수 있게 되었다.

이 간단한 패키지는 이러한 기술적 발전을 그 배경으로 하고 있다.

여담: 두벌식과 세벌식 이 문제는 키보드 자판 배열 디자인에 관한 문제이고 여기서 다루는 것과 직접 관련이 있지는 않다. 그러나 가끔 첫가끝 코드와 세벌식을 혼동하는 경우가 있어 여기 간략히 언급하기로 한다. 어쨌든 세벌식은 초성과 종성에 다른 키를 할당하는 것이고 첫가끝 코드는 초성과 종성에 다른 코드를 할당하는 것이니 연관이 없다고 할 수도 없겠다.

세벌식을 선호하는 사람들이 가끔 하는 말 중에 “한글 창제 원리에 맞다”는 것이 있다. 키보드는 익숙함의 문제가 더 큰 것이라서 어떤 것이 더 좋다거나 나쁘다거나 말하는 것이 큰 의미가 없다고 생각하지만 이 말에 대해서는 생각해볼 것이 있다. 아마도 초/중/종성을 구분하여 입력하는 것이, 초성과 중성, 종성을 합하여 한글자를 이룬다는 것과 상통한다는 말인 듯하다. 그런데 훈민정음에서는 “무릇 글자는 모름지기 어울려서 소리가 된다”고 하고 있는 한편, “종성에는 초성자를 다시 사용한다”고도 하고 있다. 이 소위 終聲復用初聲의 원리를 키보드에 (억지로) 적용하면 “종성 입력에는 초성 키를 다시 사용한다”라고 못할 것은 무엇인가? 즉 두벌식이나 세벌식이나와 훈민정음 창제 원리는 아무 상관이 없는 것이다.

내가 생각할 때 세벌식의 가장 큰 장점은 첫가끝 코드와 키보드의 입력 키를 “직결”시킬 수 있다는 점일 듯하다. 코드가 다르면 다른 키를 누른다는 것이다. 매우 안심이 된다. 입력기를 설계하는 입장에서도 세벌식이라면 첫가끝 (자모 코드) 입력을 더 편하게 구현할 수 있을 것 같다. 그러나 사실상 현대 문자 생활에서 첫가끝 코드를 쓰는 일이 거의 없으므로 이 장점이 크게 부각되지 못한 측면이 있다.

⁸LVT라는 것은 leading consonant, vowel, trailing consonant의 첫 자를 딴 것이다.

⁹유니코드의 한글 음절 영역 문자 11,172자를 모두 자면으로 갖춘 폰트가 등장한 것도 실은 오랜 일이 아니다. 그 전에는 소위 완성형 한글이라 불리는 2350자 글리프를 갖춘 것도 많았다. 그런데 생각해 보면 우리의 문자 생활이 과연 만자 정도를 필요로 하는가? 아마 수천 자 안팎이면 거의 문제가 없을 것이라 생각한다. 음절 문자도 충분히 많은 수이다.

¹⁰이를 실험하고 구현한 신정식, 박원규 제씨의 기여는 기억할 가치가 있다.

여기까지 쓰고 보니 소위 “직결식 글꼴”에도 생각이 미치는데 현재 알려진 직결식 글꼴이란 것은 한글 자모 코드를 쓰는 것이 아니라 그저 폰트에서 영문자 알파벳이 놓일 위치에 한글을 가져다둔 것일 뿐이라 별로 언급할 필요를 느끼지 못한다. 다만 공병우 박사가 생전에 이 직결 글꼴의 아이디어로 한글 문자생활을 했다는 이야기가 흥미롭다.

부록 E. 《문장강화》의 한 페이지

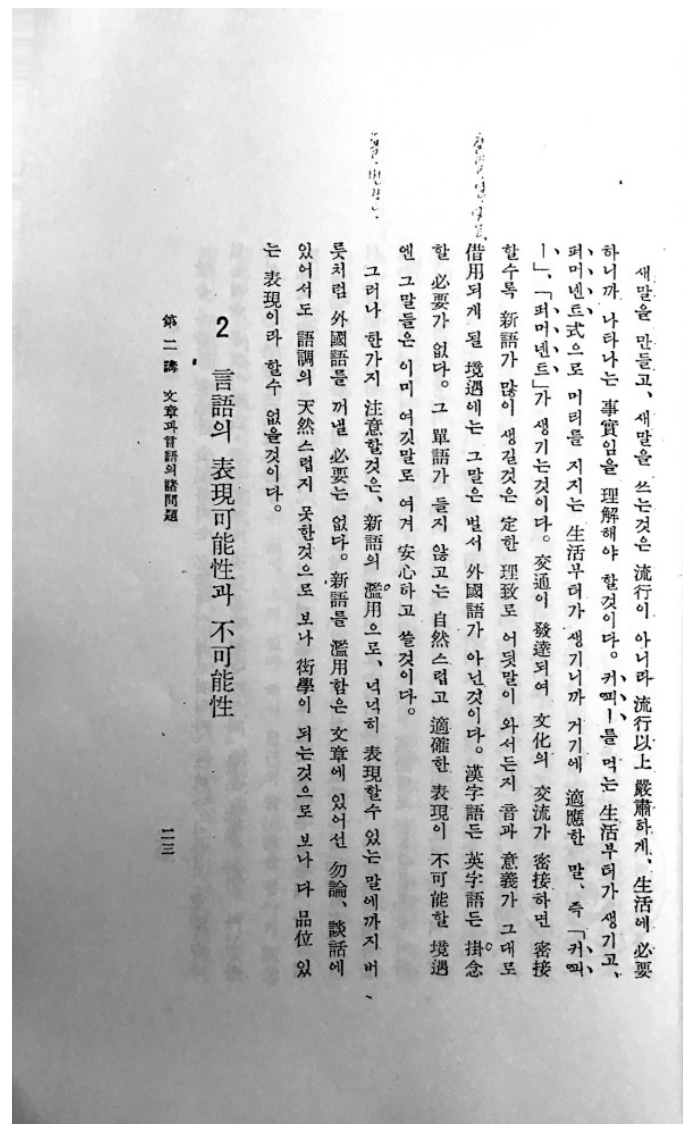


그림 2: 이태준의 《문장강화》의 한 페이지