

# LuaTeX-ko 간단 매뉴얼

Dohyun Kim <nomos at ktug org>      Soojin Nam <jsunam at gmail com>  
<<http://github.com/dohyunkim/luatexko>>

Version 2.7    2020/03/14

For a summary introduction in English, please see [README](#) file.

v2.6 —

- `\hangulbyhangulfont` 및 `\hanjabyhanjafont` 가 지역적 효력만 가짐

v2.5 —

- LuaHBTeX 지원 (세로쓰기 제외)

v2.2 —

- 글꼴옵션 `InterCharStretch` 제공

v2.0 —

- 수식 한글 글꼴을 자동으로 잡아주지 않음
- `\actualtext` 명령 작동 않음. 대신, 옛한글 텍스트 추출 자동 지원<sup>1</sup>
- 글꼴옵션 `CompressPunctuations`, `RemoveClassicSpaces`, `InterCharacter` 추가
- `\registerbreakableafter`, `\registerbreakablebefore` 명령 추가

## 1 부르기

**TeX Live 2020 이상** (LuaTeX v1.12 이상)을 사용해야 한다.

LuaTeX-ko 부르기: `\usepackage{luatexko}` 혹은 `\usepackage{kotex}`.<sup>2</sup>

입력은 원칙적으로 UTF-8으로 한다. BOM (Byte Order Mark)은 있어도 좋고 없어도 좋다. CP949 인코딩에 관해서는 제 17 절 참조.

## 2 패키지 옵션

패키지 옵션으로 `hangul` 과 `hanja`가 제공된다.<sup>3</sup> 행간격 조정이 행해지며 장 제목이나 표·그림

`hangul`  
`hanja`

<sup>1</sup>PDF 리더 어플리케이션이 ActualText 기능을 지원해야 가능하다. 어도비 리더는 확실히 가능할 것이다.

<sup>2</sup>플레인텍에서는 `\input luatexko.sty` 혹은 `\input kotex.sty`.

캡션 따위가 한글 또는 한자 방식으로 표시된다. 장(chapter) 제목과 편(part) 제목에만 “제(第)”가 붙어 “제 1 편” “제 3 장”과 같은 방식으로 표시되며 절 제목 이하에는 붙지 않는다.

### 3 글꼴명령

LuaTeX-ko를 부르면 fontspec 패키지도 자동 불러온다. 글꼴 설정은 fontspec 문서 참조.

한국어 글꼴을 위해 새로 마련한 명령은 다음과 같다. 첫 두 줄, 즉 main/sans 글꼴들에는 Ligatures=TeX 옵션이 자동으로 주어진다.<sup>4 5</sup>

<code>\setmainhangulfont</code>	<code>\setmainhanjafont</code>	<code>\setmainfallbackfont</code>	<code>\setmainhangulfont</code>
<code>\setsanshangulfont</code>	<code>\setsanshanjafont</code>	<code>\setsansfallbackfont</code>	<code>\setsanshangulfont</code>
<code>\setmonohangulfont</code>	<code>\setmonohanjafont</code>	<code>\setmonofallbackfont</code>	<code>\setmonohangulfont</code>
<code>\newhangulfontfamily</code>	<code>\newhanjafontfamily</code>	<code>\newfallbackfontfamily</code>	<code>\newhangulfontfamily</code>
<code>\newhangulfontface</code>	<code>\newhanjafontface</code>	<code>\newfallbackfontface</code>	<code>\newhangulfontface</code>
<code>\addhangulfontfeature</code>	<code>\addhanjafontfeature</code>	<code>\addfallbackfontfeature</code>	<code>\addhangulfontfeature</code>
<code>\hangulfontspec</code>	<code>\hanjafontspec</code>	<code>\fallbackfontspec</code>	<code>\hangulfontspec</code>

### 4 글꼴 대체

원칙적으로 LuaTeX-ko는 지시된 글자가 영문폰트에 없으면 한글폰트에서, 한글폰트에도 없으면 한자폰트에서, 한자폰트에도 없으면 fallback 폰트에서 글자를 찾아 찍는다. 한글폰트가 지정되지 않았고 기본폰트에 한글 글리프가 없다면 **은바탕, 은돌움 트루타입**을 자동으로 불러온다.<sup>6</sup>

하지만 `\hangulbyhangulfont=1`을 선언하면 한글은 우선 한글폰트로 식자한다. 또한 `\hanjabyhanjafont=1`을 선언하면 한자는 우선 한자폰트로 식자한다. 두 경우 모두 0을 선언하면 원래 방식으로 되돌아간다. 문단 중간에서 사용해도 동작한다. 그러나 루아코드가 실행되므로 텍의 그룹에 의해 영향받지 않는 전역적 효과를 가진다.

일정한 영문 문장부호들은 한글 폰트로 식자된다. 즉, `\hangulpunctuations=1`이 기본으로 작동하고, 0을 지시해야 이 기능이 비로소 꺼진다. verbatim 코드를 식자할 때는 이

<sup>3</sup>[nofontspec] 옵션은 v2.0부터 더는 제공되지 않는다.

<sup>4</sup>`\adhochangulfont` `\adhochanjafont` `\adhocfallbackfont`는 마지막 줄 명령들의 다른 이름들이다.

<sup>5</sup>플레인텍에서 한글 글꼴 설정은 다음과 같이 한다:

```
\hangulfont=UnDotum\relax
\hanjafont=UnDotum at 14pt
\fallbackfont=HanaMinB at 12pt
```

<sup>6</sup>은글꼴 트루타입은 T<sub>E</sub>X Live에 포함되어 있다.

기능이 작동하지 않으며,<sup>7</sup> 문장부호가 고정폭 글꼴일 때에도 작동하지 않는다.<sup>8</sup> 영향 받는 문장부호들의 기본값은 다음과 같다:

[0x21]	!	[0x27]	'	[0x28]	(	[0x29]	)
[0x2C]	,	[0x2E]	.	[0x3A]	:	[0x3B]	;
[0x3F]	?	[0x60]	`	[0xB7]	·	[0x2014]	–
[0x2015]	—	[0x2018]	‘	[0x2019]	’	[0x201C]	“
[0x201D]	”	[0x2026]	...	[0x203B]	※		

다음과 같은 명령으로 이 목록에 문자를 추가하거나 제거할 수 있다. 인자는 콤마로 분리된 숫자 형식으로서 유니코드 코드포인트를 뜻한다.

`\registerpunctuations`  
`\unregisterpunctuations`

```
\registerpunctuations{45, "2D, `-}
\unregisterpunctuations{"2D, "2015}
```

이상 문장부호 관련 명령들은 문단 중간에서도 쓸 수 있지만, 전역적 효과를 가진다.

## 5 줄바꿈 허용

어떤 글자 앞이나 뒤에서 줄바꿈을 허용하고 싶을 때 아래와 같이 명령을 준다. 이들 명령을 문단 중간에서 사용하면 문단의 처음부터 전역적 효력을 가진다.

`\registerbreakableafter`  
`\registerbreakablebefore`

```
\registerbreakableafter{"2460, "2461}
\registerbreakablebefore{"2460, "2461}
```

LuaTeX-ko는 다분히 보수적으로 줄바꿈을 허용하고 있다. 예시한 유니코드는 ‘①’과 ‘②’인데 기본적으로 이들 문자 앞뒤 모두에서 줄바꿈이 되지 않는다.

## 6 글꼴 옵션

fontspec의 글꼴 옵션 외에 LuaTeX-ko가 따로 제공하는 것들이 있다.<sup>9 10</sup>

**InterHangul** 한글 글자 사이의 자간. 아래는  $-0.04\text{em}$  만큼 한글 자간을 지시한다.<sup>11</sup>

`InterHangul`

<sup>7</sup>정확히 말하자면 `\language=\l@nohyphenation`, 즉 하이픈레이션이 허용 안 되는 상황에서는 작동하지 않는다. 부연하건대, `\language=\l@nohyphenation` 상황에서 작동하지 않는 그밖의 기능은 다음과 같다: 줄바꿈 허용, `InterCharacter`, `InterHangul`, `InterLatinCJK`, `CompressPunctuations`, `RemoveClassicSpaces`, 한글·한자 폰트 문자 다음의 공백 크기 조정 등.

<sup>8</sup>또한 플레인텍에서는 영문글꼴을 트루타입/오픈타입으로 지정해야만 작동한다.

<sup>9</sup>옛한글 식자를 위해서는 적어도 `Script=Hangul` (플레인텍이라면 `script=hang`)을 주어야 한다. 옛한글이 아니더라도 한글을 쓴다면 이 옵션과 `Language=Korean` (플레인텍은 `language=KOR`)을 주는 것이 좋다.

<sup>10</sup>`PunctRaise`, `NoEmbedding` 옵션은 v2.0부터 더는 제공하지 않는다.

<sup>11</sup>플레인텍에서는 `interhangul=<dimen>`.

**世·생宗宗御·잉製·쟁訓·훈民民正·정음음**

製·쟁·논·글지·슬·씨·니御·잉製·쟁·논·님·금·지스·산·그리·라訓·훈·은·고루·칠·씨·오民民·은·百·  
·빅姓·성·이·오音음·은·소·리·니訓·훈民民正·정音음·은·百·빅姓·성·고루·치시·논正·정훈소·리·라

**國·국之장語·앙音음·이**

國·국·은·나·라히·라之장·논·임·겨지·라語·앙·는·말·쓰·미·라  
나·랏·말·쓰·미

**異·잉乎·중·中·國·국·하·아**

異·잉·는·다·를·씨·라·乎·중·논·아·모·그·에·하·논·겨·체·쓰·는·字·중· | ·라·中·國·국·은·皇·황·帝· Deng·겨·  
신·나·라·히·니·우·리·나·랏·常·쌍·談·담·애·江·강·南·남·아·라·하·하·니·라  
中·國·국·에·달·아

그림 1: 옛한글 조판 보기

[InterHangul=-0.04em]

**InterLatinCJK** 한글 또는 한자와 Latin 문자 사이의 자간을 설정한다.<sup>12</sup> 한글·한자 다음에 라틴 문자가 오는 경우 원래는 줄바꿈이 되지 않지만 이 옵션을 주면 줄바꿈이 가능해진다.

InterLatinCJK

[InterLatinCJK=0.25em]

**InterCharacter** CJK 글자들 사이의 자간을 지시한다.<sup>13</sup> fontspec의 LetterSpace 옵션과 기능이 유사하지만, 옛한글을 깨뜨리는 등의 오작동을 하지 않는다. 전술한 자간 옵션들보다 후순위로 동작한다.

InterCharacter

[InterCharacter=.125em]

**InterCharStretch** CJK 글자간 가변공백(글루)의 stretch 값을 지시한다.<sup>14</sup>

InterCharStretch

[InterCharStretch=0.5pt]

**CharRaise** 글자의 세로 위치를 조절할 수 있는 기능이다. 이로써 주변에 식자되는 다른 글꼴과 조화를 이루게 한다.<sup>15</sup>

CharRaise

<sup>12</sup>플레인텍에서는 `interlatincjk=<dimen>`.

<sup>13</sup>플레인텍에서는 `intercharacter=<dimen>`.

<sup>14</sup>플레인텍에서는 `intercharstretch=<dimen>`.

<sup>15</sup>플레인텍에서는 `charraise=<dimen>`.

[CharRaise=2pt]

세로쓰기에서는 모든 글자들을 지정한 값만큼 오른쪽으로 이동시킨다. 세로쓰기에서 이 옵션 값을 고정길이(가령 pt 단위)로 주면 크기가 다른 폰트들간에 중앙정렬이 이루어진다. 세로쓰기에서 이 옵션을 주지 않으면 0.5ex가 기본값으로 동작한다.

**RemoveClassicSpaces** 고문헌 조판시에 CJK 글자들 사이의 공백을 없애준다.<sup>16</sup>

RemoveClassicSpaces

**CompressPunctuations** CJK 구두점 (낫표 따위)의 글자폭을 반각으로 만든다. v2.0부터는 사용자가 지시하지 않는 한 자동으로 글자폭을 줄여주지 않는다.<sup>17</sup> 이 옵션은 `CharacterWidth=AlternateHalf` 내지 (세로쓰기의 경우) `Vertical=HalfMetrics`와 거의 같은 기능을 한다.<sup>18</sup>

CompressPunctuations

**Protrusion** 특정 글자가 행 끝에 왔을 때 판면 바깥으로 끌어내는 기능이다. LuaTeX은 기본값으로 온점 반점 등을 완전 글자내밀기 한다. `Protrusion`은 `Protrusion=default`와 같은 뜻이다.<sup>19</sup> 자신만의 설정을 만들어 지정할 수 있다.<sup>20</sup>

Protrusion

**Expansion** 판면의 균일한 조판을 위해 글자들을 미세하게 늘이거나 줄이는 기능이다. `Expansion`은 `Expansion=default`와 마찬가지로 뜻이다.<sup>21</sup>

Expansion

## 7 고문헌

고문헌 조판을 위해 `typesetclassic` 환경을 제공한다.<sup>22</sup> 그림 2 참조.

typesetclassic

v2.0부터는 고문헌 조판에서 한글·한자와 영문자가 붙어 나오더라도 자동으로 간격을 잡아주지 않으므로 이런 경우 사용자가 `InterLatinCJK` 옵션을 지정해야 한다. 또한 불필요한 공백이 오더라도 자동으로 제거해주지 않으므로 이런 경우 사용자가 `RemoveClassicSpaces` 옵션을 주어야 한다.

`typesetmodern` 환경은 고문헌 조판 중에 잠깐 현대 한국어를 조판하는 데 사용한다.<sup>23</sup>

typesetmodern

고문헌 조판 중 LuaTeX-ko가 글자 사이에 삽입하는 미세간격을 사용자가 영(zero)으로 강제하기 위해선 `\inhibitglue` 명령을 이용한다.

\inhibitglue

<sup>16</sup>플레인텍에서는 `+removeclassicspaces`.

<sup>17</sup>플레인텍에서는 `+compresspunctuations`.

<sup>18</sup>플레인텍에서는 각각 `+halt` 및 `+vhal`이다. 이들과 완전 동일하진 않다. 가령 U+3002 (。) 처리의 경우.

<sup>19</sup>플레인텍에서는 `protrusion=default`.

<sup>20</sup>default 설정을 고치는 방법은 이를테면 다음과 같다 (이 매뉴얼의 소스 앞부분을 참고하라):

```
\directlua{ fonts.protrusions.setups.default[0x201C] = { 1, 0 } }
```

<sup>21</sup>플레인텍에서는 `expansion=default`.

<sup>22</sup>플레인텍에서는 (문서 전체에 적용하지 않는다면 그룹 안에서) `\typesetclassic`을 사용한다. `japanese`, `chinese`, `Schinese`, `Tchinese`, `korean` 따위의 환경은 지난 버전과의 호환성을 위해 당분간 제공되고 있을 뿐이다.

<sup>23</sup>플레인텍에서는 (문서 전체에 적용하지 않는다면 그룹 안에서) `\typesetmodern`을 사용한다.

子曰：「學而時習之，不亦說乎？有朋自遠方來，不亦樂乎？人不知而不愠，不亦君子乎？」

有子曰：「其爲人也孝弟，而好犯上者，鮮矣！不好犯上，而好作亂者，未之有也！君子務本，本立而道生；孝弟也者，其爲仁之本與？」

子曰：「巧言令色，鮮矣仁！」

曾子曰：「吾日三省吾身：爲人謀，而不忠乎？與朋友交，而不信乎？傳，不習乎？」

子曰：「道千乘之國，敬事而信，節用而愛人，使民以時。」

子曰：「弟子入則孝，出則弟；謹而信，汎愛衆；而親仁，行有餘力，則以學文。」

그림 2: 고문헌 조판 보기. typesetclassic 환경을 이용했다. 글꼴에 `CompressPunctuations` 옵션을 주었다.

## 8 세로쓰기

세로쓰기는 폰트의 고급 오픈타입 속성을 이용하므로 폰트가 이를 지원해야 가능한 일이다. 폰트에 `Vertical=Alternates`와 `RawFeature=vertical` 옵션을 준다.<sup>24 25 26</sup> 세로쓰기에서 `RawFeature=vertical CharRaise` 옵션의 효과에 대해서는 제6절을 참조할 것.

세로쓰기에서는 `\typesetclassic`의 아종인 `\typesetvertical`이 기본으로 셋팅되어 있다. 일종의 고문헌 조판으로 보는 것이다.

문서의 일부를 세로쓰기하려면 `\begin{vertical}{<dimen>} ... \end{vertical}` 환경을 `vertical` 이용한다. `<dimen>`으로 세로쓰기 박스의 높이를 지시한다.<sup>27</sup> 그림 3, 4 및 5 참조.

문서 전체를 세로쓰기한다면 이 환경을 쓰는 대신 `\verticaltypesetting` 명령을 전처리 부에 선언한다. 이때 면주는 가로로 식자되며 면주 폰트의 설정은 사용자의 몫이다. `\verticaltypesetting`

세로쓰기 도중에 문서의 일부를 가로쓰기하려면 `\begin{horizontal}{<dimen>} ... \end{horizontal}` 환경을 이용한다. `<dimen>`은 가로쓰기 박스의 너비를 지시한다.<sup>28</sup> 가로쓰기 영역의 폰트 설정은 사용자의 몫이다.<sup>29</sup> `horizontal`

<sup>24</sup>이는 플레인텍에서 `vertical;+vert` 옵션을 주는 것과 같다. 사실 `vertical`을 선언하면 `vert`는 자동으로 켜지도록 해 두었다.

<sup>25</sup>`vmx` 정보를 가지고 있지 않은 글꼴은 세로쓰기에 적합치 않은 글꼴이다. `otfinfo -t <파일>` 명령으로 글꼴에 들어있는 테이블 정보를 알 수 있다.

<sup>26</sup>폰트 옵션 `Renderer=OpenType` 또는 `Renderer=HarfBuzz` 등 (플레인텍에서는 `mode=harf` 옵션)을 주고 컴파일 하여 `Harfbuzz` 모듈을 이용하려는 경우 세로쓰기가 현재로서는 지원되지 않는다. 세로쓰기를 하려면 이 옵션을 빼고 디폴트 값인 `Renderer=Node` (플레인텍에서는 `mode=node`)를 사용하라.

<sup>27</sup>플레인텍에서는 `\vertical{<dimen>} ... \endvertical`.

<sup>28</sup>플레인텍에서는 `\horizontal{<dimen>} ... \endhorizontal`.

<sup>29</sup>폰트 지시는 `horizontal` 환경 안에서 행하는 것을 권장한다.

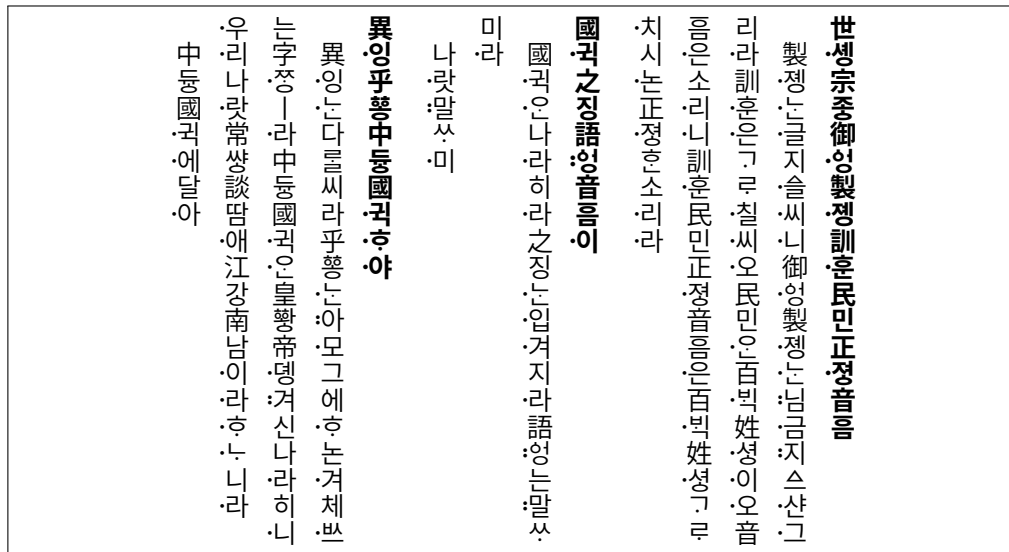


그림 3: 세로쓰기의 예. 박스 높이 20em을 지시했다.

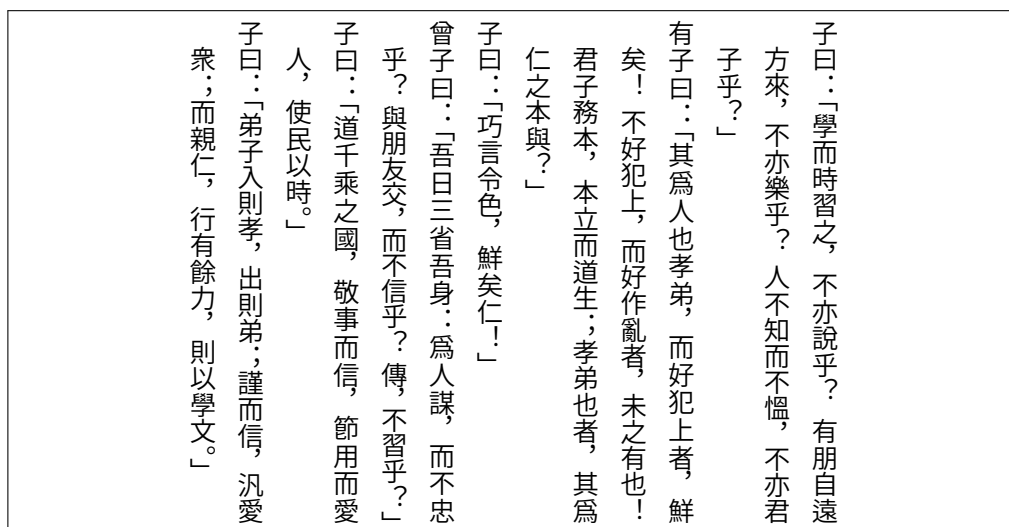


그림 4: 세로쓰기의 다른 예. 박스 높이 19em을 지시했다. 글꼴에 CompressPunctuations와 더불어 InterCharStretch=1pt를 옵션으로 주었다. 공백이 없는 문서는 자간늘이기 값을 넉넉하게 주는 것이 좋다.

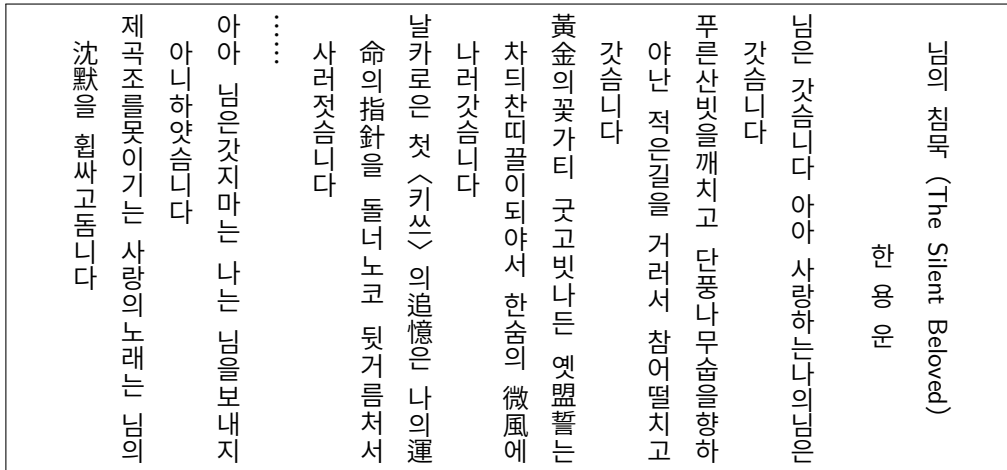


그림 5: 세로쓰기의 또 다른 예. 박스 높이 17em을 지시했다. Noto Sans CJK 글꼴의 세로쓰기 공백은 원래 1em, 즉 전각크기로 매우 크기 때문에, 여기서는 `\spaceskip` 명령을 이용하여 반각 크기로 강제했다.

## 9 드러냄표

`\dotemph` 명령으로 드러냄표를 이용한 강조를 할 수 있다. 기본은 글자 위에 점을 찍는 형태이나 다음과 같이 명령을 주어 개인적으로 선호하는 기호를 드러냄표로 쓸 수 있다.

- ① `\def\dotemphraise{0.4em}`: 드러냄표를 피강조 글자 위로 끌어올리는 길이 `\dotemphraise`
- ② `\def\dotemphchar{\bfseries ^^02d9}`: 드러냄표 기호 자체를 정의. `\dotemphchar`

## 10 루비

루비를 달 수 있다. ruby 패키지가 이미 존재하지만 LuaTeX-ko와 궁합이 잘 맞지 않아 새로 매크로를 제공한다.

`\ruby{漢字}{한자}` ⇒ 漢字 `\ruby`

이처럼 글자별로 따로 루비를 달 필요가 없다. 관련 설정은 다음처럼 한다.

- ① `\rubyfont`: 루비를 식자할 폰트를 지시해 둔다. 기본값은 현재 폰트<sup>30</sup> `\rubyfont`
- ② `\def\rubysize{0.6}`: 루비 글자 크기를 본문 글자 크기에 대한 비율로 지정<sup>31</sup> `\rubysize`
- ③ `\def\rubysep{0.1ex}`: 루비와 본문 글자 사이의 간격을 지정 `\rubysep`

<sup>30</sup>루비는 글자 크기가 작기 때문에 본문 폰트보다 약간 굵은 폰트로 지정하는 것이 좋다.

<sup>31</sup>플레인텍에서는 루비의 글꼴크기를 따로 조정해주지 않는다. 사용자가 `\rubyfont`를 설정할 때 크기도 함께 지시하도록 한다. 가령:

`\font\rubyfont=UnBatang at \rubysize em`



④ `\rubynoooverlap`: 루비의 폭이 본문 글자의 폭보다 클 때 루비가 이웃 글자들 위로 빠져나가지 못하게 한다. 본문 글자의 흐름을 중시하여 `\rubyoverlap`을 기본값으로 하였으므로 이는 따로 선언할 필요가 없다. `\rubynoooverlap`  
`\rubyoverlap`

한편, 연속된 본문 글자들에 각각 한 글자씩 루비를 달고자 한다면 `\xxruby{...}{...}` 명령을 사용한다. `\xxruby`  
글자들 사이에 줄바꿈이 허용된다. 두 인자의 글자 수가 동일해야 한다.

## 11 밑줄긋기

ulem 패키지가 LuaTeX-ko와 궁합이 맞지 않아(줄바꿈에 문제가 있음) 명령을 따로 제공한다.

<code>\uline{밑줄을 그을 수 있다}</code>	⇒	밑줄을 그을 수 있다	<code>\uline</code>
<code>\sout{취소선을 그을 수 있다}</code>	⇒	취소선을 그을 수 있다	<code>\sout</code>
<code>\uuline{밑줄을 두 줄 긋는다}</code>	⇒	밑줄을 두 줄 긋는다	<code>\uuline</code>
<code>\xout{빋금으로 취소할 수 있다}</code>	⇒	빋금으로 취소할 수 있다	<code>\xout</code>
<code>\uwave{물결표로 밑줄을 삼는다}</code>	⇒	물결표로 밑줄을 삼는다	<code>\uwave</code>
<code>\dashuline{대시로 밑줄을 삼는다}</code>	⇒	대시로 밑줄을 삼는다	<code>\dashuline</code>
<code>\dotuline{밑줄을 점선으로 긋는다}</code>	⇒	밑줄을 점선으로 긋는다	<code>\dotuline</code>

관련하여 다음 설정을 할 수 있다.

- ① `\def\ulinedown{0.5ex}`: 밑줄을 베이스라인 아래로 끌어내리는 정도 `\ulinedown`
- ② `\def\ulinewidth{0.04em}`: 밑줄의 굵기 `\ulinewidth`

사실 위 밑줄 양식들을 만드는 데는 예외없이 `\markoverwith` 명령이 사용되었다. 따라서 사용자는 이를 이용해 원하는 양식을 만들 수 있다. 가령 노란색 마커펜을 흉내내려면,<sup>32 33</sup>

```
\def\ymarker{%
  \markoverwith{%
    \hbox{{\color{yellow!50}\rule width 1pt height 9pt depth 2pt}}%
  }}

```

`\ymarker{노란색을 칠해서 강조한다}` ⇒ 노란색을 칠해서 강조한다.

## 12 자동조사

자동조사는 ko<sub>TeX</sub>과 동일하게 `\은 \는 \이 \가 \을 \를 \와 \과 \로 \으로 \라 \이`라 따위를 사용한다. 버전 1.3부터는 `\josaignoreparens=1`이 선언되어 있으면 자동조사는 괄호 부분을 건너뛰고 그 앞 글자에 매칭한다. 0이 선언되면 원래 방식으로 돌아간다. `\josaignoreparens`

<sup>32</sup>이 명령의 인자로 들어갈 수 있는 것은 box와 rule에 국한된다. 예시에서는 박스를 넣었다.

<sup>33</sup>별표 붙은 명령 `\markoverwith*`는 `\xleaders`를 실행하고, 예시처럼 별표가 없으면 `\cleaders`를 실행한다.

```

\josaignoreparens=1
홍길동(1992)\는 ⇒ 홍길동(1992)은
홍길동(2001)\로 ⇒ 홍길동(2001)으로

\josaignoreparens=0
홍길동(1992)\는 ⇒ 홍길동(1992)는
홍길동(2001)\로 ⇒ 홍길동(2001)로

```

## 13 루아 모듈의 제거 및 복원

루아 코드 `luatexko.deactivateall()`은 이상 언급한 LuaTeX-ko의 거의 모든 기능을 무력화한다. `luatexko.reactivateall()`은 이를 복원한다.<sup>34</sup> 다른 패키지와 충돌할 때 시도해 볼 수 있다. 전자의 함수에는 문자열을 인자로 줄 수도 있는데, 해당하는 다른 패키지의 모듈들을 잠시 제거할 수 있다.

## 14 수식 한글

```
$가^{나^다}$ ⇒ 가나다
```

수식 한글 폰트를 설정하면 수식 모드에서도 한글을 (hbox로 감싸지 않고) 직접 입력할 수 있다.<sup>35</sup> v2.0부터는 자동으로 수식 한글을 잡아주지 않는다.

`\setmathhangulfont`

```

\setmathhangulfont{Noto Sans CJK KR}[
  SizeFeatures={
    {Size=-6, Font=* Medium},
    {Size=6-9, Font=* Regular},
    {Size=9-, Font=* DemiLight},
  } ]

```

현재 한글만 쓸 수 있게 설정되어 있다. 한자도 수식에 직접 입력하려면 사용자는 다음 명령으로 유니코드 블록을 추가 지정해야 한다.

`\setmathhangulblock`

```
\setmathhangulblock{4E00}{9FC3}
```

<sup>34</sup>수직 모드 (문단과 문단 사이)에서 사용할 것. 수평 모드에서는 정상작동을 보장하지 못한다.

<sup>35</sup>플레인텍에서는 `\mathhangulfont=UnBatang`과 같이 설정한다. 또는 좀 더 그럴듯하게 하려면:

```

\font\texthangul="Noto Sans CJK KR DemiLight" at 10pt
\font\scripthangul="Noto Sans CJK KR Regular" at 7pt
\font\scriptscripthangul="Noto Sans CJK KR Medium" at 5pt
\setmathhangulfonts\texthangul\scripthangul\scriptscripthangul

```

## 15 한국어 항목 번호

ko<sub>TeX</sub>과 동일하게 항목 번호를 한국어 기호로 붙일 수 있다.<sup>36</sup>

## 16 입력 변환

`\luatexhangulnormalize=1`이라 지시하면 첫가끝 자모를 완성형 음절로, 2라면 완성형 음절을 첫가끝 자모로 인코딩 변환한다. 0이 할당되면 인코딩 변환 기능이 꺼진다. X<sub>TeX</sub>의 `\XeTeXinputnormalization` 명령과 유사하나, 오직 한글과 일부 한자에 대해서만 정규화가 작동할 뿐이다.

## 17 UHC 입력 인코딩

권장하지 않지만 불가피하게 입력 인코딩이 UHC (Unified Hangul Code)<sup>37</sup>로 되어 있는 파일을 처리할 때는 `\luatexuhcinputencoding=1`을 선언한다. 0을 할당하면 다시 UTF-8 입력으로 간주한다. X<sub>TeX</sub>의 `\XeTeXinputencoding` 명령과 유사하나, 오직 한국어 문자만 처리할 수 있다.<sup>38</sup>



---

<sup>36</sup>`\jaso \gana \ojaso \ogana \pjaso \pgana \onum \pnum \oeng \peng \hnum \Hnum \hroman \hRoman \hNum \hanjanum` 따위를 사용한다.

<sup>37</sup>CP949라고도 하며 EUC-KR을 포함한다.

<sup>38</sup>윈도 운영체제에서의 한글 파일이름 불러오기 기능은 v2.0부터는 제공하지 않는다. 대신 `luatex -cmdx ...`, `lualatex -cmdx ...`와 같이 컴파일시 명령행에 `-cmdx` 옵션을 주면 된다고 하니 테스트해 보시기 바란다.