The Korean Writing System

This text is not intended as a tutorial to help in learning the Korean alphabet. Instead, it is written for web designers and other people who need a superficial knowledge how to encode Korean characters in an HTML document. Basic knowledge of Unicode character representation and HTML character references is necessary to understand this text. If you don't know about Unicode, Character Sets, Encodings and I18n, please read some tutorial, e.g., Character Set Issues by A.J. Flavell.

Introduction

Each Korean character (*hangul*, 한글) represents one syllable. Although the Hangul look superficially similar to Chinese logograms, there is no close relation between the two except the roughly quadratic shape. Historically, Chinese characters were used to write Korean until the Hangul were introduced in the 15.th century. Since then, the usage of Chinese characters to write Korean (*hanja*, 한자) has declined drastically. Contemporary Korean writing has abolished Hanja to a large extent: In newspapers, for example, the Hanja typically amount to less than 5% of the characters used, and may even be completely absent. Nevertheless, scientific and scholary texts often use up to 50% Hanja, especially in matters that were historically influenced by Chinese concepts.

The Hangul are composed of letters (*jamo*, 자모) in a rather systematic way. The Jamo represent sounds similar to the way how Latin letters represent sounds. Although there are 11172 different Hangul, their individual appearances need not to be memorized; rather, one has to learn the 68 different Jamo shapes and the rules governing the construction of Hangul from Jamo. Moreover, even the Jamo shapes are not arbitrary: For example, the Jamo ㅆ (SS) looks like a duplicated ㅅ (S), and the Jamo ㄵ (NJ) is a juxtaposition of ㄴ (N) and ㅈ (J).

Some Jamo shapes are obviously mnemonic, e.g., ㅁ (M) which symbolizes a closed mouth articulating the sound /m/. It is now fairly established that this mnemonic character applies to all Jamo consonants even if they look rather arbitrary: The key is the shape and position of the tongue when producing the consonant sound.

This document discusses the relation between Hangul and Jamo, primarily from a Unicode point of view. You will be given a mathematical formula to construct (i.e., calculate the codepoint of) a Hangul from its constituent Jamo, from the Unicode names of constituent Jamo or from the Unicode name of the Hangul itself; and you will also be enabled to do the reverse analysis starting from a given Hangul character (i.e., its code point). A JavaScript form is offered to perform such calculations quickly in the browser. Writing with pencil and paper, however, is not covered here (I'm a programmer, not a kalligraph).

The Korean Letters: Jamo

A Korean syllable consists of a lead consonant, a medial vowel and a tail consonant. To write syllables with an initial vowel, a special sign for a mute lead consonant must be used. In open syllables (syllables ending in a vowel), the tail consonant is omitted. Isolated vowels can be considered regular syllables with a mute initial and a missing tail.

Number	Lead	Jamo		Character reference
1	G	ㄱ	ㄱ	&#x1100;
2	GG	ㄲ	ㄲ	&#x1101;
3	N	ㄴ	ㄴ	&#x1102;

Number	Vowel	Jamo		Character reference
1	A		ㅏ	&#x1161;
2	AE		ㅐ	&#x1162;
3	YA		ㅑ	&#x1163;

Number	Tail	Jamo		Character reference
1	G	ㄱ	ㄱ	&#x11A8;
2	GG	ㄲ	ㄲ	&#x11A9;
3	GS	ㄳ	ㄳ	&#x11AA;

4	D	ㄷ	ㄷ	ᄃ
5	DD	ㄸ	ㄸ	ᄄ
6	R	ㄹ	ㄹ	ᄅ
7	M	ㅁ	ㅁ	ᄆ
8	B	ㅂ	ㅂ	ᄇ
9	BB	ㅃ	ㅃ	ᄈ
10	S	ㅅ	ㅅ	ᄉ
11	SS	ㅆ	ㅆ	ᄊ
12		ㅇ	ㅇ	ᄋ
13	J	ㅈ	ㅈ	ᄌ
14	JJ	ㅉ	ㅉ	ᄍ
15	C	ㅊ	ㅊ	ᄎ
16	K	ㅋ	ㅋ	ᄏ
17	T	ㅌ	ㅌ	ᄐ
18	P	ㅍ	ㅍ	ᄑ
19	H	ㅎ	ㅎ	ᄒ

4	YAE		ㅒ	ᅤ
5	EO		ㅓ	ᅥ
6	E		ㅔ	ᅦ
7	YEO		ㅕ	ᅧ
8	YE		ㅖ	ᅨ
9	O		ㅗ	ᅩ
10	WA		ㅘ	ᅪ
11	WAE		ㅙ	ᅫ
12	OE		ㅚ	ᅬ
13	YO		ㅛ	ᅭ
14	U		ㅜ	ᅮ
15	WEO		ㅝ	ᅯ
16	WE		ㅞ	ᅰ
17	WI		ㅟ	ᅱ
18	YU		ㅠ	ᅲ
19	EU		ㅡ	ᅳ
20	YI		ㅢ	ᅴ
21	I		ㅣ	ᅵ

4	N	ㄴ	ㄴ	ᆫ
5	NJ	ㄵ	ㄵ	ᆬ
6	NH	ㄶ	ㄶ	ᆭ
7	D	ㄷ	ㄷ	ᆮ
8	L	ㄹ	ㄹ	ᆯ
9	LG	ㄺ	ㄺ	ᆰ
10	LM	ㄻ	ㄻ	ᆱ
11	LB	ㄼ	ㄼ	ᆲ
12	LS	ㄽ	ㄽ	ᆳ
13	LT	ㄾ	ㄾ	ᆴ
14	LP	ㄿ	ㄿ	ᆵ
15	LH	ㅀ	ㅀ	ᆶ
16	M	ㅁ	ㅁ	ᆷ
17	B	ㅂ	ㅂ	ᆸ
18	BS	ㅄ	ㅄ	ᆹ
19	S	ㅅ	ㅅ	ᆺ
20	SS	ㅆ	ㅆ	ᆻ
21	NG	ㅇ	ㅇ	ᆼ
22	J	ㅈ	ㅈ	ᆽ
23	C	ㅊ	ㅊ	ᆾ
24	K	ㅋ	ㅋ	ᆿ
25	T	ㅌ	ㅌ	ᇀ
26	P	ㅍ	ㅍ	ᇁ
27	H	ㅎ	ㅎ	ᇂ

There are 19 different lead consonants, including the mute consonant. The following table gives these consonants in their canonical order, and their Unicode values. Consonant number 12 is the mute consonant.

The vowels number 21 in Korean. Note that some of these vowels would be classified as diphthongs in other languages; also, some vowels contain an Y as part of the vowel.

The total number of tail consonants is 27; some of them are very rarely used in modern Korean. Most of the tail consonants can also appear as leads; the Jamo for these consonant pairs look very similar. Tail consonant 21 ㅇ (NG) corresponds to the mute lead consonant ㅇ (12). This correspondence is purely graphical and has no deeper meaning; some fonts will distinguish the two characters by a small vertikal stroke attached to the NG letter, while other will render them identically as a (squashed) circle.

Actually, Unicode has another range of Jamo characters called *Hangul Compatibility Jamo*, starting at U+3100. These represent the same letters, but they have no conjoining behaviour as described in the next section. The compatibility Jamo are rather unlikely to appear in a real Korean document, but they can be used if isolated Jamo must be shown in a text, for example an instruction of the writing system (like the one you are currently reading).

In this document, I use compatibility Jamo almost everywhere, as they tend to render more cleanly. In the tables on the right side, each Jamo is given twice: First as conjoinig Jamo and then as compatibility Jamo, while the hexadecimal codepoint refers to the former. The differences (if any) can be demonstrated here: Jamo GG ᄁ (head) and ᆩ (tail) and Compatibility Jamo GG ㄲ. What you will see depends on your operating system, your browser, your default font and even the font size. While the Compatibility Jamo will probably look all right, the

isolated conjoining Jamo may appear identical, smaller (and raised or lowered), overlapping their neighbours, or even empty.

Korean Syllabary: Hangul

Jamo are combined to Hangul by vertical stacking. If the vowel has a more or less horizontal shape, the three Jamo are stacked from top to bottom. An example is the Hangul NOG 녹 composed of the Jamo N ㄴ, O ㅗ and G ㄱ. If the vowel, however, extends vertically, then the upper left quarter of the Hangul displays the lead, the upper right quarter the vowel and the lower half the tail. An example is NAES 냇 which breaks down to the Jamo N ㄴ, AE ㅐ and S ㅅ. These rules are applied in a flexible way in the construction of some Hangul to yield calligraphically optimal results

Therefore, a Korean text can be seen as a sequence of Hangul, each of which represents one spoken syllable. In this view, Korean script would be seen as a syllabary comparable to East African Ge'ez script, and to lesser degree, Japanese (*kana*) scripts. On the other hand, one could understand Korean writing without reference to Hangul at all; according to this perspective, Korean writing is as alphabetical as the Latin script, but uses complicated typographic rules to determine the placement of any Jamo relative to its predecessor and successor.

The latter view also reminds to Indic scripts of the Brahmi family, where an arbitrary number of consonant signs plus a vowel are graphically combined into a syllable glyph; the combination rules for Indic scripts are, however, much more involved. It has been argued that the Indic model influenced the construction of the Hangul script via the Tibetean Phagspa script. Phagspa was a short-lived script and is now extinct; it is not the predecessor of the modern Tibeten script.

Encoding Hangul in Unicode

Theoretically, it would be possible to encode Korean texts with Jamo only and leave the task of Hangul construction to the font renderer. In Unicode, this is allowed, as a well-formed sequence of Jamo is canonically equivalent to the corresponding Hangul. Yet due to lack of support in current renderers, this is not recommended practise. To see whether your browser supports composition of Hangul from Jamo, compare the following three representations of the syllable HWEOLH: Precomposed Hangul 훓, three conjoining Jamo (H+WEO+LH) 훓, the same three Jamo enclosed in some markup to prevent their joining ᄒ ᅯ ᇜ and three Compatibility Jamo ㅎ ㅝ ㅀ. Ideally, only the first two should render identically as compound Hangul.

Canonical equivalence even extends to mixed cases of Hangul HWEO 훠 plus tail Jamo LH ㅀ, see here 훓 for a live example (support for this construction is much worse than for the previous). However, there is no canonical nor compatibility equivalence that would allow you to decompose a complex Jamo like LH into its constituents (L and H); therefore, you cannot repesent the HWEOLH syllable by something like Jamos H+WEO+L+H or by Hangul HWEOL plus tail Jamo H.

The common way of coding Korean text is to use the precomposed Hangul syllables that do not explicitly reference the underlying Jamo characters. Isolated Jamo are rarely found in Korean texts. The Unicode Standard assigns an individual code point to each Hangul. To calculate the code point of a Hangul from its Jamo components, the following formula may be used:

$$\text{Code point of Hangul} = \textit{tail} + (\textit{vowel}-1)*28 + (\textit{lead}-1)*588 + 44032$$

In this formula, *lead*, *vowel* and *tail* refer to the small integer numbers given in the above tables (if there is no tail consonant, use the value 0). The Hangul syllabary occupies the Unicode range from AC00 (decimal 44032) to D7A3 (decimal 55171). In UTF-8, each Hangul needs three bytes (the same is also true for the Jamo, which is another reason why they are almost never used for encoding Korean text).

In the other direction, the phonetic value of a Hangul can be calculated from its code point. It is convenient to use the modulo function mod(*a*,*b*), which yields the remainder of the quotient *a*/*b*, and the integer function int(*a*) which yields the integer part of *a*.

$$tail = \text{mod}\ (\text{Hangul codepoint} - 44032, 28)$$
$$vowel = 1 + \text{mod}\ (\text{Hangul codepoint} - 44032 - tail, 588) / 28$$
$$lead = 1 + \text{int}\ [\ (\text{Hangul codepoint} - 44032)/588\ ]$$

To illustrate the formulae, let us consider the writing of the words *jamo* and *hangul* in Hangul. The Hangul neccessary are called JA, MO, HAN and GEUL in Unicode.

	JA	MO	HAN	GEUL
lead consonant	J ㅈ (13)	M ㅁ (7)	H ㅎ (19)	G ㄱ (1)
vowel	A ㅏ (1)	O ㅗ (9)	A ㅏ (1)	EU ㅡ (19)
tail consonant	– (0)	– (0)	N ㄴ (4)	L ㄹ (8)
Hangul code point (dec)	51088	47784	54620	44544
Hangul code point (hex)	C790	BAA8	D55C	AE00
Hangul character	자	모	한	글

As an inverse problem, we now analyse the two Korean words 서울 and 평양:

Hangul	서	울	평	양
Hangul code point (hex)	C11C	C6B8	D3C9	C591
Hangul code point (dec)	49436	50872	54217	50577
Code point − 44032	5404	6840	10185	6545
tail consonant	– (0)	L ㄹ (8)	NG ㅇ (21)	NG ㅇ (21)
vowel	EO ㅓ (5)	U ㅜ (14)	YEO ㅕ (7)	YA ㅑ (3)
lead consonant	S ㅅ (10)	– ㅇ (12)	P ㅍ (18)	– ㅇ (12)
Hangul	SEO	UL	PYEONG	YANG

So the the two words actually stand for the capitals of South and North Korea, respectively. These are usually rendered in Latin script as Seoul and Pyeongyang, although other romanizations are possible (e.g., *Sŏul* and *P'yŏngyang* or *Pyongyang*).

Note that Unicode contains also obsolete or archaic Jamo that are absent from standard writing. They might still be used in reproducing historical texts, writing Korean dialects or transcribing Chinese words. Unicode does not offer precomposed hangul with these rare letters; instead, syllables containing them must be coded by jamo letters (if only the tail of a syllable is archaic, then the mixed representation with an open-syllable hangul followed by the archaic tail is also possible). An example is the archaic Z ㅿ appearing in the hangul ZIZ or GOZ or 고ㅿ (unlikely to render correctly)

Quick Hangul construction or decomposition

The formulae in the previous section should enable you to analyze a Hangul encountered in some dark corner, or to construct a Hangul out of its Unicode name. To illustrate the formulae, and to make things easier to use, I offer a Hangul Construction Form that allows you to create Hangul from their components, or to decompose

them. You might also construct the Hangul for fun, or to study how the visual appearance depends on the size and shape of the Jamo constituents.

You can select constituent Jamo (by Unicode name) from the dropdown menues, or enter data into the text fields. The text fields will accept *either* valid Unicode names of the right kind (e.g., "N" in the first, "YE" in the second, "LB" in the third or "BBWEOBS" in the last field) *or* true Korean characters of the right kind (combining or compatibility Jamo in the first three fields, Hangul in the last). In the course of the calculation, text field contents will be normalized into true Korean characters (actually, Compatibility Jamo in the first three fields), irrespective of the type of input. Input parsing is tolerant on case and blanks, but make sure you delete previous field contents.

G ㄱ A ㅏ (–) Hangul = 가 AC00 (decimal 44032)

Clicking the codepoint will display the current form data permanently in a table, which is useful if you wish to study the Hangul shapes in a series with varying Jamo components. Jamo shown are Compatibility Jamo, but Jamo codepoints refer to true combining Jamo.

As this form runs entirely inside the client, it depends on JavaScript. For the same reason, it will keep running if you download this file and open if offline. There seems to be no major browser issue, but subtle differences exist how browsers handle submission, selection and focus. Calculation should start immediately after you select a dropdown item, or change any text field content.

Romanization of Korean

Korean writing system, due to its phonetic nature, represented the sounds of spoken Korean very well at the time of its introduction. Since then, more than half of a millennium has passed. In this time, sound shifts have occurred in spoken Korean, but the spelling still has not been reformed much since. Therefore, written and spoken Korean do no longer correspond closely to each other. When Korean language is to be written in the Latin alphabet (“romanized”), the question arises whether the spoken or the written language should be followed in devising a romanization scheme. Both appoaches have their merits, and both are actually used.

The Unicode names of Jamo and Hangul closely follow the Revised Romanization of Korean, which has official status in Korea. The revised system basically maps each Jamo to one letter (or a polygraph) of the Latin alphabet, thus creating a faithful representation of written Korean in Latin script. It does not, however, represent the actual pronunciation very well.

Outside of Korea, the older McCune-Reischauer System continues to be popular. It takes into account certain assimilation phenomena that occur on syllable boundaries, and thus comes closer to the actual pronunciation of Korean. On the other hand, McCune-Reischauer romanizations cannot be constructed trivially from sequences of Jamo. The remainder of this section will describe the procedure briefly.

The vowels are transliterated in a rather straightforward way. As special characters, u and o with the breve accent (ŭ, ŏ) are used to denote short reduced vowels. The diacritics are often omitted when publishing on the web.

Vowel name	A	AE	YA	YAE	EO	E	YEO	YE	O	WA	WAE	OE	YO	U	WEO	WE	WI	YU	EU	YI	I
Korean	ㅏ	ㅐ	ㅑ	ㅒ	ㅓ	ㅔ	ㅕ	ㅖ	ㅗ	ㅘ	ㅙ	ㅚ	ㅛ	ㅜ	ㅝ	ㅞ	ㅟ	ㅠ	ㅡ	ㅢ	ㅣ
McCune-Reischauer	a	ae	ya	yae	ŏ	e	yŏ	ye	o	wa	wae	oe	yo	u	wŏ	we	wi	yu	ŭ	ŭi	i

The romanization of consonants is significantly more involved. The complication arises from the fact the tail of any syllable may be assimilated to the lead of the following syllable. Therefore, there are no separate

pronunciations of the tail and the lead of consecutive syllables, but both of them are to be pronounced as one single unit. The McCune-Reischauer romanization acknowledges this fact by also romanizing them as a unit. The following table gives the transliteration of tail/lead combinations involving the most common Jamo.

lead=	ㅇ –	ㄱ G	ㄴ N	ㄷ D	ㄹ R	ㅁ M	ㅂ B	ㅅ S	ㅈ J	ㅊ C	ㅋ K	ㅌ T	ㅍ P	ㅎ H	word-final
tail= ㄱ G	G	KK	NGN	KT	NGN	NGM	KP	KS	KCH	KCH’	KK’	KT’	KP’	KH	K
ㄴ N	N	N’G	NN	ND	LL	NM	NB	NS	NJ	NCH’	NK’	NT’	NP’	NH	N
ㄹ L	R	LG	LL	LT	LL	LM	LB	LS	LCH	LCH’	LK’	LT’	LP’	RH	L
ㅁ M	M	MG	MN	MD	MN	MM	MB	MS	MJ	MCH’	MK’	MT’	MP’	MH	M
ㅂ B	B	PK	MN	PT	MN	MM	PP	PS	PCH	PCH’	PK’	PT’	PP’	PH	P
ㅇ NG	NG	NGG	NGN	NGD	NGN	NGM	NGB	NGS	NGJ	NGCH’	NGK’	NGT’	NGP’	NGH	NG
ㅅ S	S	TK	NN	TT	SL	NM	TP	TS	TCH	SCH’	SK’	ST’	SP’	TH	T
any vowel		G	N	D	R	M	B	S	J	CH’	K’	T’	P’	H	
initial		K	N	T	R	M	P	S	CH	CH’	K’	T’	P’	H	

The apostrophe is used to disambiguate N’G (sequence of Jamo N + Jamo G) from NG (Jamo NG) and to mark aspirated plosives. This sign is often omitted from documents published on the web.

To illustrate the use of that table, we will romanize the National Motto of South Korea 널리 인간을 이롭게 하라 “Bring benefit to all people” using the McCune-Reischauer System.

Hangul	널	리		인	간	을		이	롭	게		하	라
Hangul Name	NEOL	RI		IN	GAN	EUL		I	ROB	GE		HA	RA
Romanization	nŏlli			in’ganŭl				iropke				hara	

Note that in the combination GAN-EUL (간을), the second hangul starts with an empty lead, thus the transcription value for the N must be taken from the first column of the table. Likewise, in I-ROB (이롭), the first hangul has no tail which means that the penultimate row applies. This is admittedly complicated, but not unreasonably so.

Table of Hangul

The following (very long) table lists all possible Hangul. Note that the loading of the table is not finished unless the last line in your browser shows the lead H and the vowel I. You may move the mouse over a character to see its Unicode code point in the status bar (if your browser allows it), and a click should show you additional information in a popup.

Note: Some browsers have problems displaying this large table. Therefore, its display is now switched off. You may enable it at your own risk (this can take minutes for some versions of the Chrome browser, though Gecko and Safari need only a few seconds).