A Manual of Brāhmī Keyboard Layout for Mac OS X

Prachya Boonkwan

National Electronics and Computer Technology Center (NECTEC), Thailand

kaamanita@gmail.com, prachya.boonkwan@nectec.or.th

Introduction

This manual and keyboard layout are designed to help you make use of the Brahmi keyboard layout (v1.0). This keyboard layout is developed to facilitate typing of Brahmi texts including vowel diacritics (*saras*), conjunct consonants (*saṃyoga*), and Brahmi numbers. It works with Mac OS X 10.10 onwards and is made available for downloads from Prachya Boonkwan's Github (https://github.com/kaamanita/brahmi). The keyboard layout was designed by Prachya Boonkwan as a hobby using *Ukelele Keyboard Layout Designer* (available on App Store) and is distributed under the MIT License.

Note that the keyboard layout is based on Unicode character encoding. It cannot be installed on older operating systems that do not support Unicode and will not work with non-Unicode applications that rely on 8-bit character set. This manual is typeset using the Unicode-enabled font *Google Noto Sans Brahmi*.

Instruction for Installation

Brahmi Keyboard Layout is packaged in a disk image file with the DMG extension. Follow the instruction below to install the keyboard layout.

1. Double click the file to open the disk image. A window for the disk image will show up.
2. Double click the Keyboard Installer app.
3. Go back to the window of the disk image. Drag the file *Brahmi.keylayout* to the app window.
4. Select a method of installation: (1) install for the current user, or (2) install for all users. Click on the corresponding icon to install the keyboard layout.
5. When finished, the keyboard installer app will close down automatically.
6. Unmount the disk image.
7. Add this keyboard layout to the system. Go to *System Preferences* > *Keyboard*.
8. In the *Input Sources* pane, push the + button. Under the *Others* language, click *Brahmi* and push the *Add* button.
9. Log out and relog in to refresh the memory.

Using the Keyboard Layout

There are four states of pressing a keystroke in this keyboard layout.

Normal state: The Brahmi keyboard layout attempts to match each keystroke to the Sanskrit romanization system as much as possible. In most cases, consonants and vowels can be typed in quite straightforwardly. For example, *K* is mapped to the

consonant + /k/, and *S* to the consonant ഛ /s/. Consonants and vowels are combined automatically, e.g. typing + [K] and ◌ി[I] will result in ഴി /ki/. There are certain cases where consonants and vowels cannot be mapped to their positions due to the linguistic dissimilarity between English and Sanskrit. For example, the consonant ം /ṅ/ is mapped to the position of apostrophe ['], while the independent vowel ഒ /ʔa/ to the position of [Q]. A conjunct consonant can be typed in using the *virāma* sign (◌്) at the position of [F]. For example, ഛ /s/ + *virāma* + ഩ /t/ + *virāma* + ാ /r/ results in a conjunct consonant ഛ്ര്യാ /str/. Finally, note that the independent forms of vowels are distributed throughout the keyboard layout.

Shift state: The shift key is pressed for four specific purposes. First, it makes a consonant aspirated. For example, pressing Shift + ഩ /g/ yields ഖ /gh/. Second, it lengthens a vowel. For example, pressing Shift + ◌ി/i/ produces ◌ീ/ī/. Third, it upgrades the class of a long vowel. For example, pressing Shift + ◌െ /e/ will result in ◌ൈ /ai/, while pressing Shift + ◌ൊ/o/ will result in ◌ൌ/au/. Fourth and last, it converts a consonant into a diacritic sign. For example, Shift + ഹ /h/ yields a visarga ◌ഃ /ḥ/, and Shift + ഠ /m/ yields an anusvāra ◌ം /ṃ/.

Alternate state: The alternate key is pressed for four specific purposes. First, it gives the independent form of a vowel. For example, pressing Alt + ◌ൊ/o/ yields ഌ /ʔo/. Second, it produces an alternative form of a consonant. For example, pressing Alt + ഺ /l/ will give us the alternative form ഽ /l/. Third, it produces the non-digit numeral system of Ancient India. For example, pressing Alt + minus [-] yields ൗ (one hundred). Fourth and last, it converts a consonant into a diacritic sign. For example, Shift + ഠ /m/ yields a candrabindu ◌ഁ /ṁ/.

Shift and alternate state: This key combination is pressed for two purposes. First, it yields the independent form of an upper class long vowel. For example, pressing Shift + Alt + ◌െ /e/ will result in ഈ /ʔai/.

Keyboard Mapping

Consonants

Most consonants are mapped to their equivalent Sanskrit romanization on the keyboard layout. However, eight consonants are displaced due to linguistic dissimilarity between Sanskrit and English, as highlighted by the red bold face. As shown in the table below, consonant aspiration is regularly marked by pressing the shift key.

	Plosive				Nasal	Approximant		Fricative
Guttural	⊦ /k/ [K]	ᒣ /kh/ Shift+[K]	ᐱ /g/ [G]	ᓄ /gh/ Shift+[G]	**ᑦ /ṅ/** **[']**	ᓗ /h/ [H]		
Palatal	ᑯ /c/ [C]	ᑉ /ch/ Shift+[C]	ᕀ /j/ [J]	ᕮ /jh/ Shift+[J]	**ᖏ /ñ/** **Shift+[Y]**	ᔪ /y/ [Y]		**ᐱ /ś/** **[W]**
Retroflex	**ᑕ /ṭ/** **[Z]**	**ᑐ /ṭh/** **Shift+[Z]**	**ᓭ /ḍ/** **[X]**	**ᓅ /ḍh/** **Shift+[X]**	ᐃ /ṇ/ Shift+[N]	ᐸ /r/ [R]	ᓬ /l/ Shift+[L]	**ᑲ /ṣ/** **Shift+[W]**
Dental	ᐠ /t/ [T]	⊙ /th/ Shift+[T]	ᕗ /d/ [D]	ᗪ /dh/ Shift+[D]	ᒪ /n/ [N]		ᓪ /l/ [L]	ᓱ /s/ [S]
Labial	ᒐ /p/ [P]	ᑲ /ph/ [Ph]	□ /b/ [B]	ᗰ /bh/ Shift+[B]	ᘐ /m/ [M]	ᓂ /v/ [V]		

Vowels

Almost all vowels are placed on the keyboard layout with respect to their equivalent Sanskrit romanization. However, the vocalic r and l are positioned at the keys [[ ] and [ ]], respectively.

As seen, vowel lengthening is marked by pressing the shift key. The independent form of a vowel can be achieved by pressing the Alt key. The consonant allophones can also be achieved by pressing the Shift and Alt keys.

One final remark to make here is that the short vowel /a/ does not have any glyphs. This is because each consonant has an inherent /a/ sound; therefore, one can simply type [K] to achieve the sound /ka/. However, the virtually empty keystroke [A] is introduced into the keyboard layout for ease of understanding of Sanskrit beginners who may sometimes accidentally type [K] + [A] in order to achieve the sound /ka/.

	Dependent form		Independent form	
Guttural	/a/ [A]	◌-/ā/ Shift+[A]	ʔa /ʔa/ Alt+[A]	ʔā /ʔā/ Shift+Alt+[A]
Palatal	◌ि/i/ [I]	◌ी/ī/ Shift+[I]	ʔi /ʔi/ Alt+[I]	ʔī /ʔī/ Shift+Alt+[I]
Labial	◌ु/u/ [U]	◌ू/ū/ Shift+[U]	ʔu /ʔu/ Alt+[U]	ʔū /ʔū/ Shift+Alt+[U]
Retroflex	◌ृ /ṛ/ **[[ ]**	◌ॄ /ṝ/ **Shift+[[ ]**	ʔṛ /ṛ/ **Alt+[[ ]**	ʔṝ /ṝ/ **Shift+Alt+[[ ]**
Dental	◌ॢ /ḷ/ **[ ]]**	◌ॣ /ḹ/ **Shift+[ ]]**	ʔḷ /ḷ/ **Alt+[ ]]**	ʔḹ /ḹ/ **Shift+Alt+[ ]]**
Palato-guttural	◌े /e/ [E]	◌ै /ai/ Shift+[E]	ʔe /ʔe/ Alt+[E]	ʔai /ʔai/ Shift+Alt+[E]
Labio-palatal	◌ो-/o/ [O]	◌ौ-/au/ Shift+[O]	ʔo /ʔo/ Alt+[O]	ʔau /ʔau/ Shift+Alt+[O]
Consonant allophones	◌ं /aṃ/ Shift+[M]	◌ँ /aṁ/ Alt+[M]	◌ः /aḥ/ Shift+[H]	

Numbers

There are two numeral systems in Brahmi script: the digit one and the older non-digit one.

The digit system is based on the decimal numbers, similar to the current Hindu-Arabic numbers.

·	𑁒	𑁓	𑁔	𑁕	𑁖	𑁗	𑁘	𑁙	𑁚
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]

Note that the number zero is represented by a single middle dot. These Brahmi digits can be achieved by pressing their equivalent keystrokes in English.

The non-digit system, on the other hand, is not based on the decimal numbers because each numeral value is represented by a distinct glyph. These non-digit number glyphs can be achieved by pressing Alt + number.

𑁛	𑁜	𑁝	𑁞	𑁟
Alt+[1]	Alt+[2]	Alt+[3]	Alt+[4]	Alt+[5]
𑁠	𑁡	𑁢	𑁣	
Alt+[6]	Alt+[7]	Alt+[8]	Alt+[9]	

The values from 10 to 90 can be achieved by pressing a combination of Shift + Alt + (first digit).

𑁤 10	𑁥 20	𑁦 30	𑁧 40
Shift+Alt+[1]	Shift+Alt+[2]	Shift+Alt+[3]	Shift+Alt+[4]
𑁨 50	𑁩 60	𑁪 70	𑁫 80
Shift+Alt+[5]	Shift+Alt+[6]	Shift+Alt+[7]	Shift+Alt+[8]
𑁬 90	𑁭 100	𑁮 1,000	
Shift+Alt+[9]	Alt+[-]	Alt+[=]	

The non-digit system is preserved in the keyboard layout only for the purpose of ancient text transcription. Its use is rather discouraged because it under-represents the decimal system.

Punctuation Marks

There are seven punctuation marks in Brahmi script. Each of them is graded according to the levels of segmentation.

𑁇	[.]	End of clause (*single ḍanda*)
𑁈	Shift+[.]	End of verse (*double ḍanda*)
𑁍	[;]	Comma
𑁎	Shift+[;]	Colon
𑁏	[/]	Dash
𑁐	Shift+[/]	End of section
𑁑	[`]	End of chapter

Orthographical System

Combination of Consonants and Vowels

When a consonant is combined with a dependent vowel, the vowel becomes a diacritic mark attached to the consonant. The shape of the vowel diacritic mark depends on the shape of the consonant and the attachment position of the vowel (top or bottom).

Consonant shapes: In Brahmi script, we characterize the shape of a consonant by its top and bottom parts. Each part is either straight, round, flat, or angled. For example, the following characters have distinct shapes of the top part.

𑀓 /k/	𑀣 /th/	𑀜 /ñ/	𑀕 /g/
straight	round	flat	angled

On the other hand, the following characters have different shapes of the bottom part.

𑀔 /kh/	𑀧 /p/	𑀩 /b/	𑀢 /t/
straight	round	flat	angled

There is only one exception for the character 𑀪 /bh/, presumably derived from the capital Pi (Π) in Ancient Greek. In this case, we consider the top part as flat, and the bottom part as straight. This is because the vowels will be attached to the right-hand-side pillar.

Vowel attachment: We characterize any vowel diacritic mark by its attachment position: top-left, top-right, bottom-right, top, and outside. The following chart classifies each vowel diacritic into five attachment positions as follows.

top-left	top-right	bottom-right	top	outside
◌𑁂 /e/	◌𑀸 /ā/	◌𑀼 /u/	◌𑁄 /o/	◌𑀾 /ṛ/
◌𑁃 /ai/	◌𑀺 /i/	◌𑀽 /ū/	◌𑁅 /au/	◌𑀿 /ṝ/
	◌𑀻 /ī/			◌𑁀 /ḷ/
				◌𑁁 /ḹ/
				◌𑀁 /aṃ/
				◌𑀀 /aṁ/
				◌𑀂 /aḥ/

The vowels in the first four categories are attached to a consonant, while those in the last category (outside) are written separately. When attached to a consonant, a vowel may change its orientation with respect to the following rotation rules to comply with the shape of the consonant.

Rotation rules: Once we can identify the shape of a consonant and the attachment position of a vowel diacritic mark, we can now determine the shape of the latter by the following rotation rules.

Rule 1: If the shape of the attached part is straight or angled, the vowel can be directly attached to that part.

+ /k/ + ◌̄ /e/ = ∓ /ke/	∧ /g/ + ◌̄ /e/ = ∧̄ /ge/
+ /k/ + ◌-/ā/ = ∔ /kā/	∧ /g/ + ◌-/ā/ = ∧ /gā/
+ /k/ + ◌̨ /u/ = ± /ku/	∧ /g/ + ◌̨ /u/ = ∧ /gu/
+ /k/ + ◌̄-/o/ = ∓ /ko/	∧ /g/ + ◌̄-/o/ = ∧ /go/
+ /k/ + ◌̇ /aṃ/ = +̇ /kaṃ/	∧ /g/ + ◌̇ /aṃ/ = ∧̇ /gaṃ/

Rule 2: In the case where the shape of the attached part is round, consider rules 2.1 and 2.2.

Rule 2.1: If the attachment position is top-left, top-right, top, or outside, we sometimes move the vowel slightly towards the center to avoid overlapping.

⊙ /th/ + ◌̄ /e/ = ⊙̄ /the/	⅂ /kh/ + ◌̄ /e/ = ⅂̄ /khe/
⊙ /th/ + ◌-/ā/ = ⊙- /thā/	⅂ /kh/ + ◌-/ā/ = ⅂ /khā/
⊙ /th/ + ◌̄-/o/ = ⊙̄ /tho/	⅂ /kh/ + ◌̄-/o/ = ⅂ /kho/
⊙ /th/ + ◌̇ /aṃ/ = ⊙̇ /thaṃ/	ᒐ /p/ + ◌̇ /aṃ/ = ᒐ̇ /paṃ/

Rule 2.2: If the position is bottom-right, we rotate the vowel before attaching.

⊙ /th/ + ◌̨ /u/ = ⊙̨ /thu/	ᒐ /p/ + ◌̨ /u/ = ᒐ̨ /pu/

Rule 3: In the case where the shape of the attached part is flat, consider rules 3.1 and 3.2.

Rule 3.1: If the position is top-left, top-right, or top, we sometimes move the vowel slightly towards the center to avoid overlapping.

ɦ /ñ/ + ◌̄ /e/ = ɦ̄ /ñe/	⊏ /ṅ/ + ◌̄ /e/ = ⊏̄ /ṅe/
ɦ /ñ/ + ◌-/ā/ = ɦ /ñā/	⊏ /ṅ/ + ◌-/ā/ = E /ṅā/
ɦ /ñ/ + ◌̄-/o/ = ɦ /ño/	⊏ /ṅ/ + ◌̄-/o/ = E /ṅo/
ɦ /ñ/ + ◌̇ /aṃ/ = ɦ̇ /ñaṃ/	⊏ /ṅ/ + ◌̇ /aṃ/ = ⊏̇ /ṅaṃ/

Rule 3.2: If the position is bottom-right, we rotate the vowel before attaching.

⊥ /n/ + ◌̨ /u/ = ⊥̨ /nu/	□ /b/ + ◌̨ /u/ = □̨ /bu/

The rotation rules aforementioned can be summarized as follows.

Shape	Part	Action
straight	any	Attach
angled	any	Attach
round	top	Move towards center
	bottom	Rotate
flat	top	Move towards center
	bottom	Rotate

Remarks on Vowel Attachment

Remark 1: As aforementioned, the character 𑀪 /bh/ is an exceptional case for vowel attachment. Since the vowels will be attached to the right-hand-side pillar, we obtain the following attachment scheme.

top-left	top-right	bottom-right	top	outside
𑀪𑁂	𑀪𑀸	𑀪𑀼	𑀪𑁄	𑀪𑀁
/bhe/	/bhā/	/bhu/	/bho/	/bhaṃ/

Remark 2: The *candrabindu* ◌𑀀 /aṁ/ becomes very similar to the anusvara ◌𑀁 /aṃ/ when combined with a consonant, e.g. 𑀓𑀀 /kaṁ/ vs. 𑀓𑀁 /kaṃ/. The candrabindu is articulated as a nasalized sound /an/ in Modern Hindi while the anusvara still preserves the original pronunciation /aṃ/. In the case of the character 𑀪 /bh/, the candrabindu is moved towards the center 𑀪𑀀 /bhaṁ/ while the anusvara is placed above the character 𑀪𑀁 /bhaṃ/.

Conjunct Consonants

Two consonants can be combined to form a conjunct consonant using the virāma sign (◌𑁆), which cancels the inherent /a/ sound. For example, 𑀲 /sa/ + *virāma* + 𑀢 /ta/ + *virāma* + 𑀭 /ra/ = 𑀲𑁆𑀢𑁆𑀭 /stra/; therefore 𑀰𑀸𑀲𑁆𑀢𑁆𑀭𑀸 is articulated as /śāstrā/. In the earliest attestation of Brahmi script (the Edicts of King Aśoka, circa 500 BC), conjunct consonants were written consecutively. But this practice evolved to a stacked conjunct consonant for orthographical compactness in the following Gupta Dynasty era (circa 200 BC). Therefore, 𑀰𑀸𑀲𑁆𑀢𑁆𑀭𑀸 /śāstrā/ is written as 𑀰𑀸𑀲𑁆𑀢𑁆𑀭𑀸 in the latter period. The font *Google Noto Sans Brahmi* supports the Aśoka's version, while Microsoft's font *Segoe UI Historic* supports the Gupta Dynasty version.

Word Boundaries and Sentence Boundaries

The earliest attestation of Brahmi script (the Edicts of King Aśoka) portrays an extensive use of space as the word delimiter. It is assumed that this practice was influenced by Ancient Greek's orthography in which space is used for the word delimiter. Generally, phrase and sentence boundaries are implicitly marked with (1) enclitic conjunction 𑀘 /ca/ 'and', (2) direct-speech particle 𑀇𑀢𑀺 /iti/ 'thus', and (3) subordinate conjunction 𑀢𑀺 /ti/ 'because'. In proses and poetry, the *single ḍanda* (𑁇) was used for clause delimiters, while the *double ḍanda* (𑁈), as verse delimiters.

License Message