Representing Myanmar in Unicode
Details and Examples

Martin Hosken[1] and Maung Tuntunlwin[2]

Introduction

One of the first reactions people often have when seeing the Myanmar script block in the Unicode standard is to say that "it doesn't work!" After all, there seem to be many characters missing. Where are all the medials? Unfortunately, people often give up at that point and do not bother to investigate further.

The problem is that people often approach the Unicode standard with a glyph model in mind. This is particularly true for Myanmar where existing fonts follow a glyph model very closely. But Unicode follows a linguistic model whereby the stored text represents the underlying characters rather than the glyphs involved. Thus, there are no separate codes for medials since a medial is simply a consonant following a primary consonant that has been devowelised[3].

This paper aims to show that the Unicode specification for Myanmar does in fact 'work'. In so doing, it will attempt to address some of the loose edges in the specification with regard to some of the more obscure areas of the Myanmar orthography for which there is no clear direction in the existing Unicode standard[4].

Even with the glyph model, there are issues with implementation. Using a linguistic model introduces different implementation issues and these will be discussed. But it must be born in mind that while the Unicode standard endeavours to be implementable, it does not claim that complexity of implementation was a dominant factor in resolving encoding issues. This means that we are not primarily concerned with keying order or whether the encoding makes rendering easy, so long as keying and rendering are ultimately possible.

Another fundamental principle of the Unicode standard is that once something is encoded it will not be removed or changed.[5] This is important otherwise a later version of a standard could break what is currently legal data. The need to update existing data to conform to a new version of the standard is not an option due to the immense problems it would cause for the computing industry. Therefore, the existing specification of the Myanmar script will stand. Only if it can be shown that the Unicode standard cannot successfully store Myanmar text, will any consideration be made of changing the existing standard.

It is hoped, therefore, that this paper will provide useful information for those wishing to implement Myanmar script using Unicode.

Basic Myanmar

The basic consonants and vowels are relatively obvious in how they are encoded. Thus:

ကော် 1000 102C letter

[1] SIL International and Payap University, Chiang Mai, THAILAND
[2] Myanmar World Distribution
[3] Also known as a consonant combination symbol.
[4] This paper is based on the Unicode standard as it stands at version 4.0
[5] The only option open to the Unicode Consortium to fix encoding problems is to encode a new character with the right properties and to deprecate the use of the old character.

Here we show the Myanmar word, the underlying Unicode codes that would be stored to represent this and an English gloss of the word. As this example shows, characters are stored in the order in which they are read.

ခါ	1001 **102C**	to shake

In this example, we highlight the code of interest. Notice how the ါ has the same code as the ာ (U+102C MYANMAR VOWEL SIGN AA) and that it is up to the rendering system to decide which form of the character is to be displayed. The same goes for diacritics. There is only one code for a particular character and it is up to the rendering system to ensure that the diacritic is appropriately placed.

ညို	100A **102F** 102D	brown
တိုး	1011 **102F** 1036 1038	to tie

Here the two forms of ◌ု (U+102F MYANMAR VOWEL SIGN U) are decided by the rendering system.

U+1031 –e vowel

We will see later why the vowels are stored in this relative order. But for now it is important to note that the Unicode standard states that vowels are stored after the consonant, according to how they are read, regardless of where they are rendered. This introduces one of the complexities of implementing Myanmar script:

နေ	1014 **1031**	the sun
ပေါ	1014 **1031** 102C	plentiful

The ေ vowel is rendered in front of the consonant that it is read (and so stored) following. Notice that this says nothing about the relative order for typing, but it does mean that anyone implementing the Myanmar script needs to take special care of this character. In general people are used to and want to type the ေ vowel in front of the consonant, and so implementors need to address issues of keyboarding as well as rendering.

Medials and Syllable Chaining

So much for what is clearly visible on the Unicode chart. What about all those glyphs that are not there? How are words including medials or involve syllable chaining stored?

ပတ္တာ	1015 1010 **1039** 1010 102C	hinge
ဖျား	1016 **1039** 101A 102C 1038	fever
ခြေး	1000 **1039** 101B 1031 1038	grime
မွေး	1019 **1039** 101D 1031 1038	give birth
မှု	1019 **1039** 101F 102F	regard important

In the linguistic model, a medial is formed by devowelising the inherent vowel of the preceding consonant. Likewise, for a syllable chained letter, the inherent vowel at the end of the previous syllable is devowelised. In Unicode this devowelising process is marked using the virama code (U+1039 MYANMAR SIGN VIRAMA). Thus we store a consonant followed by the virama and then follow it with the consonant of interest.

Devoweliser

There are two ways of representing the devowelising process. The first is by creating a medial or syllable chained form, using U+1039 to mark the devowelising. The second is to use the visible virama character (◌်) in conjunction with a base consonant. But if U+1039 is being used to mark medials and syllable chaining, how is the visible character to be represented? The Unicode standard gives the answer. The sequence U+1039

MYANMAR SIGN VIRAMA followed by U+200C ZERO WIDTH NON-JOINER is used to represent a visual virama (◌်).[6]

ထင် 1011 1004 **1039 200C** I think

Kinzi

The remaining issue regarding representation needed for the modern Myanmar language is how kinzi is represented in Unicode. Glyph based encodings give the kinzi its own code. But linguistically, the kinzi is merely a special form of a devowelised nga (U+1004 MYANMAR LETTER NGA). Thus we encode kinzi as U+1004 U+1039.

ပန်း 1005 **1004 1039** 1000 1039 101B 1036 path

Like the –e vowel, kinzi is particularly problematic to implement since people want to type it following the base consonant and it also needs careful handling during rendering.

Diacritic storage order

It is possible for a Myanmar syllable to have a number of diacritics surrounding a base consonant. Since all these diacritics are not spacing, how do we know in which order they should be stored? For example, ည့ can be stored as U+100A U+102D U+102F or as U+100A U+102F U+102D. But what happens if one person stores it one way and then someone searches for that word spelled the other way? It is important that there is a consistent way of storing strings so that applications can work consistently.

The following list gives the relative order that each diacritic should be stored at if it occurs in following a base consonant.

Name	Specification	Example
kinzi	U+1004 U+1039	◌ၼ်
Consonant	[U+1000 .. U+102A][7]	က
Stacked	U+1039 [U+1000 .. U+1019, U+101C, U+101E, U+1020, U+1021]	◌္ဂ
Medial Y	U+1039 U+101A	◌္ယ
Medial R	U+1039 U+101B	◌ျ
Medial W	U+1039 U+101D	◌ွ
Medial H	U+1039 U+101F	◌ှ
E vowel	U+1031	ေ◌
Lower Vowel	[U+102F, U+1030]	◌ု
Upper Vowel	[U+102D, U+102E, U+1032]	◌ိ
A Vowel	U+102C	◌ာ
Anusvara	U+1036	◌ံ
Visible virama	U+1039 U+200C	◌်
Lower Dot	U+1037	◌့
Visarga	U+1038	◌း

Notice the general order of: initial consonant cluster, vowels, tones.

[6] For fallback purposes, U+1039 also displays a ◌် if not followed by a consonant. This is an implementation detail and is not used in spelling words.

[7] Notice the extension of the list here to include independent vowels. The Unicode Standard V4.0 only lists values up to U+1021.

For example:

ပသ္ဌိုး	1015 101E 1039 101A 1039 101F 102F 1038	Malay
မြွာ	1019 1039 101B 1039 101D 1039 101F 102C	segmentalize
သျှောင်	101E 1039 101A 1039 101F 1031 102C 1004 1039 200C	top knot

Advanced Issues

So far we have covered what is explained in the Unicode Standard[8]. In this section we examine some of the more difficult areas of the Myanmar language including some implementation details regarding line breaking and sorting; further examination of the kinzi question; contractions and some issues with respect to Old Myanmar.

Line breaking

Myanmar does not have interword spaces like English. Instead spaces are used to mark phrases. Some phrases are relatively short (two or three syllables, 1.5em, or 2.3 times the width of U+1000 က) while others can be quite long (8.5em or 13 times the width of U+1000 က). A common approach to addressing line breaking issues is to adjust the phrase spacing so that a line breaks at a phrase break. If this approach fails and a phrase must be continued onto a second line, U+200B ZERO WIDTH SPACE may be used to indicate a possible line break point in the text.

The problem with this approach is that when phrases are quite long or a lot of text is to be typeset, the manual adjustment of phrasing or the introduction of zero width spaces can be onerous. A further option is to break lines automatically within phrases when needed. The clearest solution is to have a line break occurring at a word boundary, but since there are no word breaks in Myanmar this is not immediately possible. Most words, though, are mono-syllabic and so a mechanism of breaking lines at syllable boundaries is usually sufficient. From this we can say that a syllable break may occur before a base consonant so long as the consonant:

- is not devowelised with a visible virama and
- has no stacked consonant below it (ignoring true medials: –y –r –w –h) and
- is not a kinzi.

These same syllable breaking rules are used for sorting purposes. For example these phrases show possible inter-syllable line breaks.

ကောင်လေးတွေကျောင်းသို့သွားကြသည်။	1000 1031 102C 1004 1039 200C | 101C 1031 1038 | 1010 1039 101D 1031 | 1000 1039 101A 1031 102C 1004 1039 200C 1038 | 101E 102F 102D 1037 | 101F 1039 101D 102C 1038 | 1000 1039 101B | 101E 100A 1039 200C 104B	the kids are going to school
အိပ်ခန်း**တံခါး**ကို	1021 102D 1015 1039 200C | 1001 1014 1039 200C 1038 | **1010 1036** | **1001 102C 1038** | 1000 102F 102D	to the bedroom door

Notice how in the second example the word 1010 1036 | 1001 102C 1038 is a single word with multiple syllables. Is there some way without a dictionary, that the we can ensure that the word is not line broken? There is a Unicode character that was added for version 4.0: U+2060 WORD JOINER. Previous to this the character U+FEFF ZERO WIDTH NON-BREAKING SPACE was used. Since U+FEFF is most commonly used at the start of a Unicode text file to both identify it as being Unicode data and to indicate the encoding form of the data, U+2060 was added to the standard to take over the function of zero width non-breaking space. The role of this character is to indicate a non-breaking point in a text. Lines should not be broken at that point. Therefore, if we want to ensure that no line-break occurs at the syllable boundary within our poly-syllabic

[8] Version 4.0, 2003

word, we can insert a U+2060 into our data stream between the two syllables and a rendering engine should not break a line at that point. Thus:

အိပ်ခန်းတံခါးကို	1021 102D 1015 1039 200C | 1001 1014 1039 200C 1038 | 1010 1036 **2060** 1001 102C 1038 | 1000 102F 102D	to the bedroom door

In summary, therefore, we propose three levels of line breaking support: breaking at phrase spaces; breaking at syllable breaks and support for polysyllabic words. A rendering engine may choose the sophistication of line breaking support it provides.

Sorting

Sorting Myanmar strings is a complex process involving, significant string transformation and four levels of comparison. The string transformation is a syllable based operation for which the identification of syllable boundaries (but not word boundaries) are required. The same techniques that are used for line-breaking, therefore, may be used for sorting.

Kinzi revisited

Consider the word အင်္ဂ. How can it be represented? The normal encoding we would expect would be 1021 1004 1039 101D 1031. But there are two ways of interpreting this string:

အင်္ဂေါ် 1021 (1004 1039) 101D 1031

အင်္ဂ 1021 1004 (1039 101D) 1031

The question is how different systems will interpret the string. One approach is to say that kinzi above a consonant is rare and that kinzi above one of the for *medial* consonants (U+101A U+101B U+101D U+101F) is very rare, and so we can say that for the sequence U+1004 U+1039 U+10xx we interpret as (U+1004 U+1039) U+10xx if U+10xx is a normal non medial consonant and that we interpret the sequence as U+1004 (U+1039 U+10xx) if U+10xx is one of the medial consonants.

But 1004 1039 represents a kinzi:

အင်္ဂေါ် 1021 1004 1039 101D 1031

အင်္ဂေါ 1021 1004 1039 **200C** 101D 1031

But neither of these representations are the word we want, so how can we represent this word? The main issue is whether we interpret 1021 1004 1039 101D 1031 as 1021 (1004 1039) 101D 1031 or as 1021 1004 (1039 101D) 1031?

There are different approaches we can take, the approach we propose here is to introduce either a U+200C ZERO WIDTH NON-JOINER or a U+200D ZERO WIDTH JOINER between the U+1004 and the following U+1039. The effect of this is to break the kinzi string and to make the U+101D look more like a medial than a main consonant. The sub-sequence 1004 200C 1039 101D results in the rendering we want. But it also marks that there is a syllable break between the U+1004 and the U+101D, while we want the syllable break to occur before the U+1004. So a better solution is: 1004 200D 1039 101D. Here the ZERO WIDTH JOINER indicates that the syllable should be held together and therefore that there is a break before the syllable. Notice that all this talk of syllable breaks makes no difference for rendering (even line breaking). It is only needed for sorting purposes.

If we consider all the sequences that can occur, we can see what the rendering will be and where the syllable break will occur when sorting.

အင်္ဂေါ့်	1021 1004 1039 101D 1031 1037	✓ (good)
အင်္ဂ့	1021 1004 **200C** | 1039 101D 1031 1037	✗ (bad)
အင်္ဂ့်	1021 | 1004 **200D** 1039 101D 1031 1037	✓ (good)

အင်္ဂဝ့	1021 1004 1039 **200C** | 101D 1031 1037	✓ (good)
အင်္ဝ့်	1021 | 1004 1039 **200D** 101D 1031 1037	✗ (bad)

The final column lists whether the syllable breaking is appropriate. Notice that there is no sequence that explicitly marks a kinzi (only the final sequence does this), and that also is appropriate for indicating that the kinzi is associated with the previous syllable. Therefore we have to make the unmarked initial sequence always represent a kinzi (with the syllable break following the kinzi).

The principles presented here form the basis for the solutions for other encoding problems in Myanmar.

Contractions

The Myanmar language has a system of double acting consonants, where a consonant acts as both the final of a syllable and the initial of a following syllable. These are significant for sorting purposes. Double acting consonants are rare, but occur in two common words.

ယောကျ်ား	101A 1031 102C 1000 **1039 200C** 1039 101A 200C 1038	man, husband
ကျွန်ုပ်	1000 1039 101A 1039 101D 1014 **1039 200C** 102F 1015 1039 200C	I (1st person singular)

Notice how the visible virama (1039 200C) occurs immediately after the double acting consonant. This position is not listed in the standard diacritic order, but is most appropriate for a double acting consonant. In order to identify such contractions, we propose that the visible virama be stored immediately following the consonant.

There are also words with double acting consonants which are unmarked. Since these are unmarked, it has been decided that despite their etymology, these words should be sorted as if there were no double acting consonant.

ကလျာဏ	1000 101C 1039 101A 100F	womanly virtues
အဍ္ဍန်	1021 101D 1039 101F 1014 1039 200C	momentum

Old Myanmar

There are a few issues that storing old Myanmar text introduce.

Stacked Ya

There are occasions where a medial ya (U+101A) representation is used for a stacking ya. What is needed is a syllable break between the base consonant and the ya. Thus we propose:

ဥယျာန	1025 101A **200C** 1039 101A 102C 1014	ဥယျာဉ်	garden/orchard

The extra column gives the modern spelling of the word. The use of U+200C ZERO WIDTH NON-JOINER indicates the break in the syllable. It makes no difference to rendering and is only used in Pali sorting.

LaSwe (Medial la)

In some words, a subscript la (U+101C) acts as a medial rather than as the start of a new syllable. We propose this change of sorting be marked using U+200D ZERO WIDTH JOINER between the base consonant and the medial:

လ္ဘ	1000 **200D** 1039 101C	ကျ	drop
လ္ဘျပ်	1000 1039 101A **200D** 1039 101C 1015 1039 200C	ကျပ်	tight

In the second example, while the U+200D is not necessary (since from context it is possible to identify that the U+101C is a medial), it is included for consistency, making implementation easier.

AMyint (Medial A)

Old Myanmar includes a medial form of U+1021 which causes no particular problems since it is just treated as any other medial, and occurs very rarely.

1014 102D 101A **1039 1021** 1039 200C

day

Searching and Comparison

The approach we have taken here to add markers to handle syllable breaking can introduce problems when searching and during string comparison. Since the codes used do not make any change to rendering, it is likely that in many cases the codes will be left out of a text. Therefore there is a need for searching and comparison code to take into account and ignore the extra codes inserted.

The following codes should be ignored when searching and comparing: U+200B ZERO WIDTH SPACE and U+2060 WORD JOINER.

U+200C ZERO WIDTH NON-JOINER and U+200D ZERO WIDTH JOINER are more problematic. At the strictest level of comparison, they should be included but since in many cases they are only used to control syllable breaking for sorting a helpful approach to searching and comparison would only consider these character when they affect rendering. U+200C affects rendering only when it follows U+1039 MYANMAR SIGN VIRAMA. U+200D only affects rendering when it occurs in the sequence U+1004 U+200D U+1039.

Conclusion

This paper has presented an overview of how Myanamar language text may be stored in Unicode in a consistent and conformant fashion. For those coming from a glyph based encoding background, the Unicode model can be amazing. But the linguistic model has its strengths and provides a good compromise between the needs of analysis and of rendering.

The Unicode Standard gives a basic diacritic order. In order to support contractions, we propose a single change to this order that also allows a visible virama to be stored immediately following its base consonant and before any medial or vowel. Thus there are two allowable positions for the visible virama and it is this difference in position that carries the information regarding a contraction.

Mechanisms have also been presented for resolving ambiguities regarding kinzi, stacked ya and la swe. Line breaking has been considered and different levels of line breaking support proposed along with mechanisms to facilitate both line breaking and sorting.

References

Bechert, et al 1979, *Burmese Manuscripts, Part 1* Wiesbaden.

Department of the Myanmar Language Commission 1993, *Myanmar – English Dictionary* Ministry of Education, Union of Myanmar.

Okell, John 1994, *Burmese: An Introduction to the Script* SOAS, London.

The Unicode Consortium 2003, *The Unicode Standard, Version 4.0* Addison-Wesley, Massachusetts.

Appendix

This is a subset of the complete chart of the Myanmar block in the Unicode standard. The other letters are not used.

	100	101	102	103	104
0	က	တ	ဥ	◌ု	ဝ
1	ခ	ထ	ဣ	◌ေ	ဝ
2	ဂ	ဒ		◌့	၂
3	ဃ	ဓ	ဣ		၃
4	င	န	ဤ		၄
5	စ	ပ	ဥ		၅
6	ဆ	ဖ	ဦ	◌ံ	၆
7	ဇ	ဗ	ဧ	◌္	၇
8	ဈ	ဘ		◌း	၈
9	ဉ	မ	ဩ	◌်	၉
A	ည	ယ	ဪ		၊
B	ဋ	ရ			။
C	ဌ	လ	◌ာ		၌
D	ဍ	ဝ	◌ိ		၍
E	ဎ	သ	◌ီ		၎
F	ဏ	ဟ	◌ု		၏