Adding Ethiopic Features To Emacs

ናውዶ ታካሐሺ
Naoto Takahashi, *ntakahas@etl.go.jp*
(Electrotechnical Laboratory, Japan)
ዳንኤል ያዕቆብ መኰንን
Daniel Yacob, *dmulholl@cs.indiana.edu*
(Indiana University)

June 1997

Abstract

The authors have been successfully working on a project whose goal is to add various Ethiopic support to Mule (the Multilingual Enhancement to GNU Emacs) [1].

In addition to the basic editor operations, e.g., displaying, inputting and editing Ethiopic text, Mule supports file I/O in various formats (SERA, TEX, etc.) and reading/writing Ethiopic E-mail over 7-bit networks. As it is multilingual in nature, users can mix Ethiopic text with any other languages supported in Mule (English, French, Japanese, Arabic, etc.)

Our Ethiopic support software runs under the standard UNIX + the X window system environment and requires no special hardware.

As a result of the integration of Mule and GNU Emacs, GNU Emacs 20.1 supports these Ethiopic features by default. Porting to Windows 95 and NT is also on going.

1 Introduction to Emacs and Mule

Mule is a multilingual editor that is used all over the world. In addition to the powerful functions of well-known GNU Emacs, it allows users to use not only English but also many other languages such as German, French, Russian, Japanese, Chinese, Arabic, etc. all at once. Also it is extensible; users can add their own new functions and even new languages.

Mule is accompanied with a printing facility called m2ps. It converts a multilingual file saved in Mule's internal format into the PostScript format with the help of the BDF display fonts. The quality of the output depends on the resolution of the BDF fonts.

Usually, m2ps is invoked via a shell script called any2ps. Any2ps accepts not only Mule's internal format but also other file formats widely used in the Internet community. For the code conversion, a general purpose code conversion library called coco is used. Coco is also included in the distribution set of Mule.

In spite of these capacities, no attempt was made in the first few years to support Ge'ez script (or "Ethiopic" as it is more internationally known) since Mule was developed and is maintained mainly in Japan, geographically and culturally distant from East Africa.

1.1 Origin of the Ethiopic mode

In late 1993 UNIX and VMS were still the prevalent operating systems on the Internet where Emacs would most commonly be found. As the Internet propagated and became more available with each passing month at universities and corporations world wide, so too would grow communication

between increasing numbers of people. The number of people wishing to communicate in Ethiopic script was growing faster than the technology's adaption to the community's needs.

A transliteration system for Ethiopic script materialised when the Ethiopic writing system could not otherwise be transmitted between the increasingly diverse interconnected systems. Haphazard experiments with Ethiopic writing began under the X11 window system. Talk clients and a simple file viewer applying the transliteration system came forth and were set to be merged into an Ethiopic editor by mid 1994.

Fortunately, the X windows application developers then came across the multilingual version of the premier editor and all purpose interface -Emacs! The X11 development efforts were redirected towards adding an "Ethiopic Mode" to version 2.0 of Mule. By early November the Ethiopic mode was introduced and users had an Ethiopic editor, news and mail reader, and web browser.

Development of the Ethiopic mode has been on going since its initial release. Although there are still many things to be done, the Ethiopic mode has advanced to a point where users can perform most of their routine text oriented tasks in Ethiopic script. As the result of the integration of Mule and GNU Emacs, GNU Emacs 20.1 supports these Ethiopic features by default. This means that an Ethiopic editor will be widely spread all over the world in the near future.

2 Supporting the Ethiopic mode

The problem of adding Ethiopic language support to Mule is a matter of implementing the solutions for the following points:

Character set
: By default, Mule-2.3 provides a native character set for the Ethiopic script. But the latest version of Ethiopic support kit uses another character set that is compatible to Unicode/ISO 10646. This kit is available via FTP.

Character attributes
: In addition to "character-syntax", which is a native Emacs feature, Mule supports "character-category" that can be attributed to any kind of characters. All Ethiopic characters have the character category '`e`'. This helps a lot in specifying a regular expression that contains Ethiopic characters. For example, "any one Ethiopic character preceded by ASCII `A` and followed by `B`" can be expressed by '`A\ceB`'.

Font
: Mule-2.3 uses fixed width fonts. An Ethiopic font designed in 16x16 pixels is provided among other fonts.

Input method
: Ethiopic characters are inputted via "quail", a subsystem of Mule.

Language environment
: Mule offers various supports that are specific for Ethiopic, like the command interface and text manipulation functions. Unfortunately the menu bar and message catalogs have not been internationalised yet.

Addressing these areas for Ethiopic often meant devising original solutions when no precedence or standards were available to follow. As the first three items to do not require further discussion, our paper will focus on detailing the methods we implemented in the last two areas.

3 Basic editing features and commands

This section describes how to input Ethiopic characters in Mule. The most basic and important operation of an editor is character inputting. Unlike the case of English, how to input characters is not a trivial problem for languages that use hundreds or thousands of characters. And Ethiopic is in this case. There are more than 300 characters in Ethiopic script, thus it cannot be coded in one byte. At the same time, it is impossible to map Ethiopic characters on a standard ASCII keyboard, which has only about one hundred keys.

3.1 The Quail input method

Mule allows users to input non-ASCII characters from an ASCII keyboard in various ways. In the case of inputting Ethiopic characters, a subsystem called "quail" is used. When the quail system is turned on, the keyboard is configured for a particular language or script (e.g. Ethiopic); when it is turned off, the keyboard is configured for ordinary ASCII. Users can turn on and off quail with a single key stroke.

The behaviour of quail varies depending on the "package" it uses. Each quail package defines an input method for a language or a script. Users can switch to any package at anytime they want. Thus it is easy to mix various languages and scripts in the same text.

In addition to the predefined packages, users can add their own private packages. For this purpose, Mule provides a set of Emacs Lisp functions and macros. In fact, the first version of the Ethiopic input system was implemented as a private quail package. Later, the Ethiopic package became a predefined package in Mule and included in the standard distribution of Mule.

A quail package consists of a package name, a set of translation rules and some other information. The package name is used by users to select the desired package. Translation rules define the correspondences between a key sequence and the inputted character. In Mule-2.3, translation rules are defined with the Emacs Lisp function 'quail-defrule' as shown below. (The way of defining character keystrokes is similar in Emacs-20.1.)

```
(quail-defrule "he" "ሀ")
```

The above reads "to input the Ethiopic character ሀ, type the keys '`h`' and '`e`' in that order". In other words, Ethiopic characters are inputted by typing corresponding mnemonics.

As can easily be seen here, the mnemonic codes are based on phonetic transcriptions. They are similar to the SERA code (discussed in Section 4) but not the same. While the SERA system is intended to be read by humans, these mnemonics are received and analised only by computer. So the ease in typing has greater weight than readability in the mnemonic code. Of course, the mnemonics are defined not only for the Fidel but also for Ethiopic numerals and punctuations.

3.2 Details of Ethiopic syllable input

ሀ፦ Why another keyboard system?

The keyboard input method is arguably the most important level of the user interface for an editor type application. The design of the Mule keyboard for Ethiopic script took into consideration a number of points that bear going over as they lead to a completely new input system. As there are no established standards for an Ethiopic keyboard, Ethiopic software usually provides one or more keyboard systems. Usually some implementation of the traditional typewriter layout is provided along with a phonetic like system.

Of fundamental importance in the input method design would be the text entry speed. This meant that all text elements used in Ethiopic publishing would have to be accessible without leaving the Ethiopic mode. Text entry through control sequences should also be minimised or eliminated.

Since the first version of the Ethiopic mode was targeted towards SERA users, full SERA input was supported so that users could go between Mule and an ASCII resource without modifying their typing system. This was a sound principle, however SERA was meant to be a balance between what is easy to read as well as to type. Mule on the other hand would use SERA as a transfer medium but the user would read Ethiopic and not ASCII. Thus some shortcuts were introduced on top of SERA to enhance the typing speed in Mule. These shortcuts are discussed in the following sections.

አ፦ Entering consonants

In a few instances, as seen on the keyboard in Appendix A, uppercase letters having no Ethiopic assignment (namely `B`, `F`, `J`, `L`, `M`, `R`, `V` and `Y`) were made equivalent to their lowercase counterparts (i.e. `L = l`, `m = M`, etc).

Letters that would not fit on the keyboard through phonetic mapping can be entered by three ways:

1. through double-strike of its phonetic value (ss = ሥ),
2. the phonetic value followed by the number '2' (s2 = ሥ), or
3. by the alternate modifier, backquote, followed by the phonetic value (`s = ሥ).

The typist is left to find the preferable input method. A list of characters entered in this way follows:

sse	=	s2e	=	`se	=	ሠ
hhe	=	h2e	=	`he	=	ኀ
kke	=	k2e	=	`ke	=	ቸ
ee	=	e2	=	`e	=	ዐ
SSe	=	S2e	=	`Se	=	ፀ

When phonetic mapping conflicts occurred, the character that occurs more frequently has the priority (i.e. ሀ vs ኀ).

ሐ፦ Entering vowels

When a key corresponding to a consonant is first struck, the user will see the sixth form of the character appear on the screen with a small underline, 'ም'. The user may then type a vowel to modify the syllabic form of the consonant, or type any non-vowel character to terminate the entry of the letter (the underline goes away). The new character typed will also appear on the screen. A small table for the input system is given now:

me	mu	mi	ma	mE	m	mo	mWa
መ	ሙ	ሚ	ማ	ሜ	ም	ሞ	ሟ

Though the 8^{th} form in Wa is given for SERA compatibility, the character is rendered with W alone to facilitate a faster typing speed.

With the exception of the 8^{th} form Amharic vowel, ኧ, all vowels may be entered with a single keystroke ("u/U" below reads as 'u' *or* 'U') as shown:

e/a	u/U	i	a/A	E	I	o/O	ea
ኧ	ኡ	ኢ	ኣ	ኤ	እ	ኦ	ዐኧ

By default, input of letters follows the regular rules of the Ethiopic writing system used by most languages (Tigrigna, Oromiffa, Guragigna, etc). A Lisp variable may be set for Amharic composers to allow the 'a' key to be mapped to 'ኧ' rather than to 'ኣ'. Amharic key mapping is set with the following function:

```
(setq ethio-primary-language 'amharic)
```

A minor complication arises when the composers wishes to enter a lone vowel following a 6^{th} form consonant. The entry of the consonant may then be terminated with the apostrophe (') as described in SERA and does not render in the buffer:

	l'e	l'u	l'i	l'A	l'E	l'I	l'o
also →		lU		lA		lI	lO
	ልኧ	ልኡ	ልኢ	ልኣ	ልኤ	ልእ	ልኦ

Finally, for the consonant classes having extension to the series of five dipthongs, input again follows as per SERA:

hWe	hWu/hW'	hWi	hWa	hWE
ኈ	ኍ	ኊ	ኋ	ኌ

A note on consonants with 12 forms: In different geographic regions, and at different times within the same region, people have been taught two different sounds for form 9 ('ዀ' in the above). Both phonetical inputs are permitted.

3.3 Ethiopic punctuations vs. ASCII punctuations

All Ethiopic and ASCII punctuations are available in quail-mode with the Ethiopic package. The Ethiopic punctuations are mapped to the keyboard based on equivalence of function as well as appearance to ASCII punctuations. A small table depicts input:

Resembling Glyphs			Resembling Function			
:	→	፡	.	→	።	
::	→	፡፡	..	→	.	(Ethiopic Dot)
:-	→	፦	,	→	፣	
-:	→	፥	;	→	፤	
:|:	→	፧	**	→	፨	
<<	→	«	?	→	?	
>>	→	»	??	→	⁇	
`!	→	¡	`?	→	⁈	

Hitting the same punctuation key repeatedly will replace the Ethiopic version with the ASCII. Since both Ethiopic and ASCII punctuations are commonly used in Ethiopic publishing, this feature is convenient because users can input both types of punctuations without turning on and off the quail-mode.

3.4 Ethiopic numeric input

Both the Arabic (ASCII) and Ethiopic numerals are available in quail-mode with the Ethiopic package. The Arabic numbers are the default when users strike the number keys of the keyboard. Ethiopic numbers may be entered following their SERA names (i.e. `` `1 `` → ፩).

Typing the number keys in Ethiopic numeral mode will render the numerals on the screen. Users may enter Ethiopic numbers like Arabic numbers with 0's to obtain the multiples of ten for numbers 1 - 9 up to a multiple of 1,000,000. This limit is chosen as it will be the largest value that two numbers together may represent (namely ፺፼) before three numbers are required together (i.e. ፩፺፼).

3.5 Word spaces

As shown above, the Ethiopic space (`:`) is always available by typing a colon (`:`). But, Mule provides other ways for its access. The F2 key (written as `[f2]` in Emacs Lisp) toggles whether the ASCII space or the Ethiopic space is inputted by the space bar. The current "space mode" is displayed in the mode-line. Further, `[S-f2]` (shift + F2) works to change all the spaces, either from Ethiopic to ASCII or from ASCII to Ethiopic in the specified region. The Ethiopic space mode has the intelligence not to add additional Ethiopic word spaces following a word space or other punctuation. In that case ASCII spaces are inserted instead. Finally, X11 users of Mule may modify their environment keyboard maps to input `:` with `[S-SPC]` (shift + space bar) which will override the space insertion intelligence in Ethiopic space mode. The details of these functions are documented in the online manual.

3.6 Gemination

It is an often noted deficiency in the writing conventions of Ethiopic languages that there be no orthographic representation of consonant length. Doubling or "gemination" of consonants may change the meaning of a word entirely. Consider ገና pronounced as either *gäna* or *gänna* with meanings of "yet" or "Christmas" respectively.

Linguists and scholars working in Ethiopic apply an Ethiopic gemination symbol, ፕብቅ (*ṭäbq*) to denote a consonant's length. ፕብቅ is available under Mule by entering `C-´` (control + apostrophe) following the character to be geminated. For example: ኣለC-´ → ኣሏ.

A geminated character is realised as a "composite character", which is another extension of Mule's notion of characters. With a macroscopic view, it is a character (`C-f`, `C-d`, etc. work as if it is one character); but with microscopic view, it is a block of characters (each internal character is accessible). Decomposition of a composite character is also possible.

3.7 Vowel modification

By placing the cursor on an Ethiopic syllable and hitting the `[f6]` key and a new vowel, users can modify the syllabic form of the character without changing the consonant. For example, when ለ is selected (appearing like ⌈ለ⌋), `[f6]` is struck, and the user changes the syllable by striking a vowel, 'a' for instance, then ⌈ለ⌋ becomes ላ.

This feature is of convenience to the Ethiopic typist since Ethiopic keyboard conventions abound and the typist, familiar with several systems, may produce errors regularly.

3.8 Ethiopic special characters

Special Ethiopic icons, glyphs, and ligatures are available from the Ethiopic mode. The [f8] key is used to display a menu to select the special characters.

4 ASCII transliteration and character code support for Ethiopic

4.1 SERA and Mule

Until the Unicode specification for Ethiopic completes the ISO stages of officiation, there remains no universally recognised encoding standard for Ethiopic script. The consequence of this absence of an encoding standard over the last few years has lead the Ethiopic language speakers to transcribe and transliterate Ethiopic writing generally into English conventions to assure successful communications on the Internet.

When written systematically, ASCII files (a file composed with the characters available from a common English keyboard) may be interpreted by Mule and viewed as Ethiopic text. Likewise, an Ethiopic document may be written out by Mule into ASCII for transfer and importation elsewhere.

Mule uses "SERA" (System for Ethiopic Representation in ASCII) as its Latin representation system of Ethiopic to provide compatibility with other network and PC softwares, and to allow users without Mule the ability to easily read Ethiopic documents composed with the editor. Though independent from file I/O, SERA has been applied as the input method (IM) for Mule as well, thus much of the system has already been discussed in Section 3.2. Comprehensive and current information on SERA is available online [2].

Mule provides a very rich support of SERA and goes a little further to add typesetting intelligence to make the transliterated documents more pleasant to read. Some functions are available in the Ethiopic Mode for working with SERA buffers. If struck alone, `[f4]` will convert an entire buffer into Ethiopic text. It is assumed that the text begins in Ethiopic. If the beginning of the document should start with Latin script, users can use `C-u [f4]` (control + u, then F4) to make the appropriate conversion. Users may also use `[S-f4]` to convert only a specified region (that has been "marked" or highlighted). `[f5]`, `C-u [f5]` and `[S-f5]` converts back into SERA-Latin with the same rules applying for `[f4]`.

4.2 Customising SERA

Ethiopic is a highly phonetical script set, but not perfect. The clearest example for this is in Amharic where the first form and the fourth form"h"s have the same sound ('ሀ', 'ሐ', 'ኀ' sound the same as 'ሃ', 'ሓ', 'ኃ'). To model in Latin the different sounds associated with a particular Ethiopic member, SERA allows for duplicity of the Latin representation. One such important instance is the choice of '`e`' *and* '`a`' to represent the first lone vowel letter 'አ'. In Amharic, '`a`' is the natural choice and '`e`' is more logical in most other languages where '`a`' is more appropriate for 'ኣ'. Both '`a`' and '`e`' are recognised as 'አ' when converting from Latin to Ethiopic for Amharic; but the user must decide on a choice of '`a`' or '`e`' for converting from Ethiopic to Latin.

The default conversion in Mule is to write 'አ' as '`e`'. Mule may be set to use Amharic rules to always convert 'አ' as '`a`' by the following way:

```
(setq ethio-primary-language 'amharic)
```

As discussed before in Section 3.2, the natural choice of '`Wu`' or '`W'`' for 9th form characters will differ between people. '`Wu`' is the default conversion into Latin, but it may set to '`W'`' by the following way:

```
(setq ethio-W-sixth-always t)
```

SERA also permits the apostrophe (') as a separator between two Latin vowels when the user thinks it aids clarity. The default in Mule is not to insert an apostrophe between vowels, users may change this again in the same way as we have already seen:

```
(setq ethio-quote-vowel-always t)
```

As discussed before in Section 3.4, Ethiopic numerals may be written in several forms and interpreted in the same way. Likewise users have these same choices for SERA output.

For example, consider the Latin sequence `` `10`9`100`80`7 `` for '፲፱፻፹፯'. The Latin form is considered to be in the lowest level of reduction, or "`reduction-level 0`". Users can opt to write out in "`reduction-level 0`" by the following way:

```
(setq ethio-numeric-reduction 0)
```

The next level in reduction is the form `109100807 which can be set by:

```
(setq ethio-numeric-reduction 1)
```

The final level in reduction is the form `10900807 which can be set by:

```
(setq ethio-numeric-reduction 2)
```

These three reduction levels for numbers are offered, because it is not known at this time which of them ultimately becomes the preferred form.

The SERA transcription configuration for the treatment of punctuation is described in the online manual.

5 Ethiopic communication features

5.1 E-mail & News

Mule/Emacs offers various environments in which users can read or write E-mail. We have Ethiopicised the most standard environments, namely rmail-mode (for reading) and mail-mode (for writing). By setting the the hook functions as follows, users can activate the automatic conversion utility for Ethiopic E-mail.

```
(add-hook 'mail-send-hook 'fidel-to-sera-mail)
(add-hook 'rmail-show-message-hook 'sera-to-fidel-mail)
```

The conversion utility automatically converts the incoming E-mail into Ethiopic script if it is written in SERA; it also automatically converts the outgoing E-mail into SERA if it is written in Ethiopic script.

One of the advantages of using SERA is that it can be sent through 7-bit communication protocals. Though the situation is rapidly changing, it is still a good idea to limit the contents of E-mail within 7-bit format. The automatic conversion utility guarantees that the E-mail messages are always transferred in 7-bit code.

The conversion is transparent for users. Not only the messages written in Ethiopic and converted into SERA by Mule, but also the messages directly composed in SERA by the sender appear in Ethiopic script in the receiver's Mule buffer. On the other hand, both kinds of messages appear in SERA if the receiver is using anE-mail reader other than Mule. This feature makes the communication between Mule users and non-Mule users easier. Those who use Mule see messages in Ethiopic script no matter how they were written; those who do not use Mule see messages in SERA no matter how they were written. In both cases users see messages in the same format as they write.

Posting to Internet News groups such as USENET follows the same principals and employs the same Mule mechanisms. To add SERA conversion in News postings simply set the hook as follows:

```
(add-hook 'news-inews-hook 'fidel-to-sera-mail)
```

Users with news readers that have a SERA conversion service will display the text in Ethiopic; users without them may still read and write in SERA transliteration.

5.2 Supported encodings other than SERA

In addition to SERA, Mule also recognises three other encoding systems for Ethiopic script: EthiTEX, Java and HTML.

When a user saves the content of a buffer into a file whose name ends in "`.sera`", "`.tex`", "`.java`" or "`.html`", Mule converts the Ethiopic characters into the appropriate format. For example, ሀ will become "`he`", "`\heG`", "`\u1200`" or "`<sera>he</sera>`", respectively. Likewise, the Ethiopic characters encoded in those encodings appear as real Ethiopic characters when a user opens the file. Again, the conversion is transparent to users; they feel as if raw Ethiopic characters are included in the file.

The `<sera>` HTML markup is recognised by the Emacs "W3" web browser but it may be replaced in the future when a transliteration markup or attribute convention is a part of the HTML specification.

5.3 Ethiopic TEX

Though m2ps and any2ps provide basic printing capability, it is not enough for serious typesetting. For example, users cannot change the font within a text.

One of the most powerful and most widely used desktop typesetting systems is TEX. TEX is extensible and many macros have been written for various languages. There are several Ethiopic TEX systems and Mule can be used as an interface for them. Users can compose TEX sources with Ethiopic characters and Mule converts them into TEX macros when it writes the content of a buffer in a file.

The Ethiopic TEX support in Mule is deserving of special attention. Presently the Ethiopic TEXer has a choice of different Ethiopic packages to choose from. A user may elect to use only the fonts from an Ethiopic package in which case he or she is confronted with incompatible encoding systems between the packages. Considering these options and the forthcoming Unicode extension of TEX, "Omega", the principal in Mule is to support all Ethiopic TEX systems transparently.

This is accomplished by applying a '`.tex`' file hook (mentioned in Section 5.2) where Ethiopic text is converted into TEX type macros. For instance `υ` in a buffer will be converted into `\heG` when the buffer is saved with the file extension '`.tex`'. The Ethiopic TEX macros will be converted into Ethiopic text when the file is re-opened by Mule. To compile the TEX based document, escape mapping files are included with the document. An escape mapping file defines the macros appropriately for the user's preferred Ethiopic TEX system.

Two such Ethiopic TEX packages are the ethiop package for Babel and the EthiTEX package. To use Mule with ethiop or the fonts from the EthiTEX package, users need only include the mapping table in their documents to convert Mule's Ethiopic TEX macros onto the target system. The required mapping tables are not provided with Mule but are available by ftp with example LATEX files [5]. A sample usage of the files is presented in Appendix B.

6 Summary

We have described various Mule's functions concerning the Ethiopic script. Note that all of these functions were implemented by user level extensions. Currently fully supported languages are limited to Tigrigna and Amharic, but other Ethiopic script languages could also be supported by users.

A final noteworthy feature in Emacs and Mule is the powerful online help system. In addition to the accessibility to the description of each mode, function, environment and so on, a hypercard-like documentation system called "info" provides the users all the necessary information in a structured way. Info can be used both as a reference book and as a manual.

At the time of writing this paper, the integrated version of Mule/Emacs is at the final stage of preparation. Its porting to Windows 95 (called MEADOW) is also under its alpha testing. Both of them are supposed to include the Ethiopic supports described in this paper by default.

The Ethiopic mode does require that Mule/Emacs be run in a graphical environment in order to display the Ethiopic fonts. In the near future when Emacs supports Unicode display in terminals this limitation will be overcome. UTF-8 streams will also then be supported for Ethiopic text.

It is hoped the Ethiopic mode in Mule will offer users the capability to perform their basic text processing needs with an interface that is natural and intuitive. Users may apply the Ethiopic support and the full power of Emacs for advanced text processing and programming purposes such as an Ethiopian language spell checker for example. Appendix C provides a summary of variables and functions used in the Ethiopic mode.

References

[1] Handa, Nishikimi, Tomura, Takahashi: Unified and Extensible Mechanism for Multilingual Text Processing, PRICAI-96 Workshop on Future Issues for Multilingual Text Processing, 1996. `http://www.etl.go.jp/etl/donald/mule/mule.html`

[2] Firdyiwek, Y. and Yacob, D., "The System for Ethiopic Representation in ASCII", Bloomington, Indiana, 1993-1997, `ftp://ftp.cs.indiana.edu/pub/fidel/info/sera.ps`

[3] Becker, Dr. Joseph, et.al, "Conclusive Proposal for Encoding of Ethiopic Syllabary, USA (ANSI) & Unicode Consortium (UTC-95-055A)", XSoft/Xerox, December 9, 1995

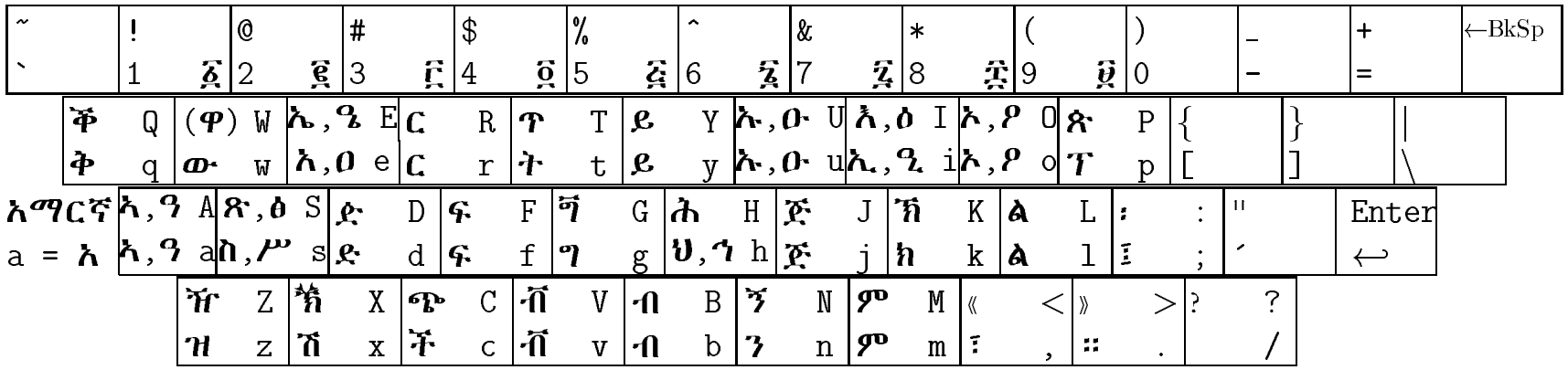
[4] Beyene, B., Kudlek, M., Kummer, O., and Metzinger, J., "Ethiopian Language Support for the Babel Package", Fachbereich Informatik, Universität Hamburg, 1997 `ftp://ftp.dante.de/tex-archive/languages/ethiopia/ethiop/doc/ethiop.ps`

[5] Yacob, Daniel, Ethiopic TEX maps for Mule, `ftp://ftp.cs.indiana.edu/pub/fidel/latex/`

[6] Alamnehe, Abass, EthTEX, `http://www.neosoft.com/~ethiosys/ethtex/ethtexan.html`

[7] Fisseha, Y. and Yacob, D., "IM Principles for an Ethiopic Text Editor", Bloomington, Indiana, 1994-1996, `ftp://ftp.cs.indiana.edu/pub/fidel/info/sera-im.ps`

Appendix A Ethiopic keyboard of Mule



Appendix B Examples of Mule

Ethiopic E-mail

A Ethiopic E-mail message in SERA as composed by Mule viewed in an ASCII E-mail client:

```
Return-Path: ntakahas@etl.go.jp
Date: Fri, 27 Jun 1997 17:56:24 +0300
From: Naoto Takahashi  <ntakahas@etl.go.jp>
To: dmulholl@cs.indiana.edu
Subject: \~eng Emacs 20.1
<sera>
\~amh
selam dan´El:-

sle \~eng Emacs \~amh yemnegreh neger aleN:: do/r handana stalman
be\~eng Cambridge \~amh tesebsbew g`Iz be20\.1 verxn lay medebeNa
Indihon tesmamtewal::

slezih hulet  aynet imaks meketet ayasfelgm
Inam g`Iz lehulum beimaks qes beqes ydrsal::
\~eng Mule\~amh na \~eng Emacs \~amh yemiyawahdew keTqit gzE behWale yeTenaqeqal::

yantew,

  nawto
</sera>
```

The same E-mail message viewed in Mule:

```
Return-Path: ntakahas@etl.go.jp
Date: Fri, 27 Jun 1997 17:56:24 +0300
From: Naoto Takahashi  <ntakahas@etl.go.jp>
To: dmulholl@cs.indiana.edu
Subject: Emacs 20.1

ሰላም ዳንኤል፦

ስለ Emacs የምነግረህ ነገር አለኝ። ዶ/ር ሃንዳና ስታልማን
በCambridge ተሰብስበው ግዕዝ በ20.1 ቨርሽን ላይ መደበኛ
እንዲሆን ተስማምተዋል።

ስለዚህ ሁለት  አይነት ኢማክስ መከተት አያስፈልግም
እናም ግዕዝ ለሁሉም በኢማክስ ቀስ በቀስ ይድርሳል።
Muleና Emacs የሚያዋህደው ከጥቂት ግዜ በኋለ የጠናቀቃል።

ያንተው፣

  ናውቶ
```

Ethiopic TEX

Following is a minimal document with the `ethtex` map for the EthTEX fonts:

```
\documentclass{report}
\input{ethtex}

\begin{document}
  \begin{tabular}{|*{8}{c|}} \hline
    ሀ & ሁ & ሂ & ሃ & ሄ & ህ & ሆ & \\
    ለ & ሉ & ሊ & ላ & ሌ & ል & ሎ & ሏ \\ \hline
  \end{tabular}
\end{document}
```

Following is a minimal document with the `ethbabel` map for the ethiop package with Babel:

```
\documentclass{report}
\usepackage[english,german]{babel}
\usepackage{ethiop}
\usepackage{fontenc}
\input{ethbabel}

\begin{document}
\selectlanguage{ethiop}
  \begin{tabular}{|*{8}{c|}} \hline
    ሀ & ሁ & ሂ & ሃ & ሄ & ህ & ሆ & \\
    ለ & ሉ & ሊ & ላ & ሌ & ል & ሎ & ሏ \\ \hline
  \end{tabular}
\end{document}
```

The ethiop package for Babel and the EthTEX package are available from CTAN archives, information can also be found by WWW [4], [6]:

Appendix C List of variables and functions

Here are the list of user customisable variables and user accessible functions prepared for the Ethiopic support. See the online manual for their usage.

ethio-primary-language [variable]
Symbol that defines the primary language in SERA to FIDEL conversion.

ethio-secondary-language [variable]
Symbol that defines the secondary language in SERA to FIDEL conversion.

ethio-use-colon-for-colon [variable]
Non-nil means associate ASCII colon with Ethiopic colon.

ethio-use-three-dot-question [variable]
Non-nil means associate ASCII question mark with Ethiopic old style question mark (three vertically stacked dots).

ethio-quote-vowel-always [variable]
Non-nil means always put an apostrophe before an isolated vowel (except at word initial) in FIDEL to SERA conversion.

ethio-W-sixth-always [variable]
Non-nil means convert the Wu-form of a 12-form consonant to "`W'`" instead of "`Wu`" in FIDEL to SERA conversion.

ethio-numeric-reduction [variable]
Degree of reduction in converting Ethiopic digits into Arabic digits.

ethio-implicit-period-conversion [variable]
Non-nil means replacing the Ethiopic dot at the end of an Ethiopic sentence with an Ethiopic full stop.

ethio-java-save-lowercase [variable]
Non-nil means save Ethiopic characters in lowercase hex numbers to Java files.

ethio-sera-to-fidel-region [function]
Convert the characters in region from SERA to FIDEL.

ethio-sera-to-fidel-buffer [function]
Convert the current buffer from SERA to FIDEL.

ethio-sera-to-fidel-mail [function]
Convert SERA to FIDEL to read/write mail and news.

ethio-sera-to-fidel-marker [function]
Convert the regions surrounded by <sera> and </sera> from SERA to FIDEL.

ethio-fidel-to-sera-region [function]
Replace all the FIDEL characters in the region to the SERA format.

ethio-fidel-to-sera-buffer [function]
Replace all the FIDEL characters in the current buffer to the SERA format.

ethio-fidel-to-sera-mail [function]
Perform FIDEL to SERA conversion for reading/writing mail and news.

ethio-fidel-to-sera-marker [function]
Convert the regions surrounded by <sera> and </sera> from Fidel to SERA.

ethio-modify-vowel [function]
Modify the vowel of the FIDEL that is under the cursor.

ethio-replace-space [function]
Replace ASCII spaces with Ethiopic word separators in the region.

ethio-input-special-character [function]
Allow the user to input special characters.

ethio-fidel-to-tex-buffer [function]
Convert each FIDEL characters in the current buffer into a fidel-tex command.

ethio-tex-to-fidel-buffer [function]
Convert fidel-tex commands in the current buffer into FIDEL characters.

ethio-toggle-space [function]
Toggle ASCII space and Ethiopic separator for keyboard input.

ethio-insert-space [function]
Insert an appropriate word separator depending on the current situation.

ethio-insert-ethiopic-space [function]
Insert the Ethiopic word delimiter (the colon-like character).

ethio-toggle-punctuation [function]
Toggle Ethiopic punctuations and ASCII punctuations for keyboard input.

ethio-gemination [function]
Compose the character before the point with the Ethiopic gemination mark.