ArabTEX
a System for Typesetting Arabic
User Manual Version 3.00 [1] [2]

Klaus Lagally

November 22, 1993

[1] Report Nr. 1993/11, Universität Stuttgart, Fakultät Informatik, Breitwiesenstraße 20–22, 70565 Stuttgart, Germany

[2] This Report supersedes Report Nr. 1992/06

Overview

ArabTEX is a package extending the capabilities of TEX/LATEX to generate the Arabic writing from an ASCII transliteration for texts in several languages using the Arabic script. It consists of a TEX macro package and an Arabic font in several sizes, presently only available in the Naskhi style. ArabTEX will run with Plain TEX and also with LATEX. It is compatible with NFSS, NFSS2 and the EDMAC package; other additions to TEX have not been tried.

ArabTEX is primarily intended for generating the Arabic writing, but the standard scientific transliteration can also be easily produced. For languages other than Arabic that are customarily written in the Arabic script some limited support is available.

ArabTEX defines its own input notation which is both machine, and human, readable, and suited for electronic transmission and Email communication. However, texts in some of the Arabic standard encodings can also be processed.

Please send error reports, suggestions and inquiries to the author:

Prof. Klaus Lagally
Institut für Informatik
Universität Stuttgart
Breitwiesenstraße 20-22
70565 Stuttgart
GERMANY

`lagally@informatik.uni-stuttgart.de`

Contents

1 Activating ArabTEX **5**

2 Input to ArabTEX **6**

2.1 Arabic text elements 6

2.2 Commands in an *Arabic context* 7

3 Language selection **10**

4 Font selection **11**

5 Input coding conventions **12**

5.1 Standard Arabic and Persian characters 12

5.2 Quoting 15

5.3 Ligatures 16

5.4 Vowelization 16

5.5 Verbatim input 17

5.6 Alternate input codings 17

6 Transliteration **19**

6.1 ZDMG transliteration style 19

6.2 Encyclopedia of Islam style 20

7 Support for other languages besides Arabic **21**

7.1 Persian (Farsi, Dari), also Ottoman, Kurdish 21

7.2 Urdu 22
7.3 Pashto (Afghanic) 22
7.4 Maghribi 23
7.5 Other languages 24

8 Miscellaneous features 25
8.1 Automatic stretching 25
8.2 Dots on *yā'* 25
8.3 Additional codings 25
8.4 Progress report 26
8.5 Verbatim copy of the input 27
8.6 Using ArabTEX with EDMAC 27

9 Acknowledgments 28

10 References 29

A Obtaining ArabTEX 32

B Installing ArabTEX 33

C Release history 34

D Sample ArabTEX input 36

E Sample ArabTEX output 37

F Coding examples for Arabic 38

G Coding examples for Persian 45

H Alternate input encodings 48
H.1 ASMO 449 = ISO 9036 48
H.2 ASMO 449E = ISO 8859 - 6 50

I Miscellaneous utilities **52**

I.1 twoblks.sty 52

I.2 abjad.sty 53

I.3 MLS2ARAB 53

Index **54**

List of Tables

5.1 Standard codings for Arabic and Persian. 13
5.2 Additional codings generally available. 14
5.3 Verbatim codings for the carrier of *hamza* 17

7.1 Additional codings for Urdu. 23
7.2 Additional codings for Pashto. 24

8.1 Additional codings for special purposes. 26

H.1 ASMO 449 code table 49
H.2 ISO 8859-6 code table 51

Chapter 1

Activating ArabTEX

With Plain TEX, load the ArabTEX macros by `\input arabtex.tex`. With LATEX, include the option `"arabtex"` in the document header. In both cases some additional files will be loaded automatically.

ArabTEX defines several user commands as indicated below. There is also a large number of (hidden) internal commands which could lead to storage (hash table[1]) overflow in a small TEX implementation. All internal commands contain an "at" sign (`@`) in their names and thus should not interfere with any user defined commands (but could possibly with other TEX extensions we do not know about).

With Plain TEX, the Arabic font by default is only available at the normal 14 point size which ought to cooperate well with the `"cm"` fonts at 10 points. A bold variant is also provided. For other sizes, the user has to change the `\magnification` or to define additional font identifiers himself. To change the default, inspect the file `"arabtex.tex"` and redefine the `\pnash` and/or `\pnashbf` command accordingly. With LATEX, the usual size changing commands will also operate on the Arabic font.

[1] A TEX hash table size of 3000 to 3500 is recommended

Chapter 2

Input to ArabTEX

After activating ArabTEX, select one of the Arabic writing styles, e.g., `\setarab` (see Section 3). Your modified TEX/LATEX system will recognize the following items:

- normal TEX/LATEX text and commands,
- short *Arabic quotations* bracketed by < and > . These must normally fit onto one line of output, except if explicitly broken up by \\ or \| commands (see below). A quotation may also be started with \< except inside a LATEX `{tabbing}` environment.
- longer Arabic texts which are bracketed by `\begin{arabtext}` and `\end{arabtext}`, (even when using Plain TEX!), called *Arabic Environments* in the sequel. An *Arabic Environment* consists of one or more paragraphs separated by blank lines or `\par` commands.

Arabic quotations and *Arabic environments* are called *Arabic contexts* in the sequel.

2.1 Arabic text elements

Every *Arabic paragraph* and every *Arabic quotation* is a sequence of the following kinds of *Arabic items*, separated by blank spaces or newlines:

- isolated punctuation marks, interpreted as the corresponding Arabic punctuation mark;

- "numbers", i.e. character sequences starting with a digit. A "number" will be processed using the normal writing sequence from left to right even if it contains letters and/or special characters; however, if the final character is a punctuation mark, it will be split off and processed separately.
- "Arabic quotes" coded as two left quotes or two right quotes each; they may also be written directly adjacent to a word.
- "words", i.e. character sequences starting with a letter or a special (non-digit) character followed by a letter. A final punctuation mark will be split off and processed separately. The (coded) characters of a word will in the output be arranged from right to left.
- a sequence of words, numbers, and special characters enclosed in curly braces `{` and `}` . This introduces a new level of TEX grouping; otherwise the constituents are processed normally. This feature may be nested.

Output from all items will be arranged from right to left, lines will be broken as necessary.

Inside an *Arabic Environment,* or in an *Arabic quotation,* you may also have:

- ArabTEX commands with or without parameters. These will be executed immediately.
- Some, but not all, TEX/LATEX commands (see below). These will be executed immediately.
- Short mathematical insertions, bracketed by *single* `$` signs. They must fit on one output line and are processed as usual. TEX Display mode within an *Arabic environment* is not provided; if it is required, the user has to leave the *Arabic environment* temporarily.
- short *non-Arabic* ("Roman") *quotations,* containing text and possibly also TEX/LATEX commands, bracketed by `<` and `>` . These must fit on one output line and introduce a new level of grouping, so if they contain any TEX/LATEX assignments the effects of these will be local by default. This feature is not available within an *Arabic quotation.* The alternate notation `\<` is also *not* provided.

2.2 Commands in an *Arabic context*

A control sequence inside an *Arabic context* must be separated from the preceding text item by at least one blank space, newline, or another control sequence, and may be of the following kinds:

- ArabTEX option changing commands. These may also be used outside an *Arabic Context,* and usually follow the TEX grouping rules.
- \\ for a line break; the last line will be padded on the left with spaces.
- \| for a line break; the last line will be aligned. If it comes out very badly spaced, automatic stretching might help (see Section 8).
- `\indent` or `\par` (or a blank line) for a new paragraph, `\noindent` for a new paragraph without indentation; (*not inside Arabic quotations*).
- `\emphasize` *Arabic_item* will put a bar over the *Arabic item.*
- `\emphasize {`*group_of_Arabic_items*`}` will put a bar over the indicated group of *Arabic items.*
- `\setnash, \setnashbf, \setnastaliq` font selection commands, see Section 4.
- size changing LATEX commands like `\large` etc., only if LATEX is used!
- the following commands: `\footnote` (observe that the syntax for Plain TEX and LATEX is different!), `\marginpar` (also with Plain TEX, analogous to the LATEX usage).
- the TEX/LATEX commands `\smallskip`, `\medskip`, `\bigskip`, `\input`, `\hfill`, `\␣` (for a space), `\space` with their usual meaning.
- `\nospace` will place the adjacent items in the output in contact, without any intervening space.
- `\hspace {`*width*`}` will introduce the indicated amount of spacing in the output.
- `\mbox {`*text*`}` puts the *text* into a box that will not be split across a line break.
- `\spreadbox {`*width*`}{`*text*`}` spreads out the *text* to the indicated *width.* This may be useful e.g., when typesetting poetry.
  `\spreadbox {`*width*`}{`*text*`\hfill }` will inhibit the spreading,
  `\spreadbox {`*width*`}{\hfill` *text*`\hfill }` will center the *text* inside the box.
  `\spreadbox {`*width*`}{\hfill }` or `\spreadbox {`*width*`}{␣}` just introduces the indicated amount of horizontal space, as will `\hspace {`*width*`}`.

  If two boxing commands follow each other without any intervening blank space in the input, there will also be no resulting space between the boxes in the output.

- `\centerline {`*text*`}` will start a new line whose contents are centered (*not inside Arabic quotations*).
- `\spreadline {`*text*`}` will start a new line whose contents are spread out over the whole width of the page (*not inside Arabic quotations*). It is approximately equivalent to `\spreadbox {\hsize }{`*text*`}`.
- User defined commands whose expansion produces legal ArabTEX input may be called by `\docommand {`*command and parameters*`}`. The command is expanded exactly once,[1] and the result is processed by ArabTEX again. Any side effects of the expansion will be local.
- Parameter assignments inside an *Arabic context* may be performed by `\doassign {`*parameter*`}{`*value*`}`. The effect is normally local except if the form `\doassign {\global` *parameter*`}{`*value*`}` is used.
- Any non-recognized command will generate an error message and will be echoed verbatim in the output. Even though ArabTEX tries hard to get into synchronization again, additional spurious errors may occur.
- inside an *Arabic Context* no further LATEX or ArabTEX environment may be nested (with the possible future exception of list environments; these are not yet implemented.)

For a list of all available commands, consult the Index to this report. As a reminder, a list of all commands that are valid inside Arabic text will appear in the log file.

[1] This is no strong restriction as the expansion may contain `\docommand` calls again.

Chapter 3

Language selection

The processing of input text to be written in the Arabic script is somewhat language dependent. Thus before the first *Arabic quotation* or *Arabic environment* you have to indicate the desired processing mode by one of the commands `\setarab`, `\setfarsi`, `\seturdu`, `\setpashto`, `\setmaghribi`, or `\setverb` (no special processing; see however Section 5.5). The processing mode may be changed at any time, even inside an *Arabic environment* or an *Arabic quotation.*

After selecting a language, the symbols < and > serve to bracket short insertions in the chosen language. Whereas this is usually convenient, observe that they can thus no more be used for other purposes, except in mathematical mode where they retain their normal meaning as relational operators. To temporarily return them to their normal mode of operation, deselect the language by `\setnone`. *Arabic insertions* may also be started by `\<`.[1]

For further details on supported languages, see Section 7.

[1] Note for advanced TEX users: All language selecting commands except `\setnone` set the character < active. If *Arabic insertions* are not needed, or are always started with `\<`, the user may reuse the command < for other purposes, or deactivate it by `\catcode ‘\<=12` to return it to its normal meaning.

Chapter 4

Font selection

For space economy, only the Naskh font is available by default. With LaTeX, additional fonts can be loaded by the document style options `"nashbf"` (for bold-face) and/or `"nastaliq"` (when available). Users of Plain TeX are considered specialists and have to define and load suitable fonts at the required sizes themselves.

The following font selection commands are available:

- `\setnash` (default) selects the Naskh font.
- `\setnashbf` selects a bold-face version of Naskh.
- `\setnastaliq` selects the Nasta'liq font.

If a font is not available or has not been loaded, the corresponding command will select the default font.

With LaTeX, the size changing commands will also operate on the additional fonts.

Chapter 5

Input coding conventions

The ASCII input notation for Arabic text has been modelled closely after the transliteration standards ISO/R 233 and DIN 31 635. As these standards do not guarantee unique re-transliteration and are also not 7-bit ASCII compatible, some modifications were necessary. These follow the general rules:

- whenever the transliteration uses a single letter, code that letter;
- whenever the transliteration uses a letter with a diacritical mark, put the punctuation character most closely resembling the diacritical mark *before* the letter (and *not* behind it as in some other coding proposals, as otherwise the readability of the input would suffer).
- use capital letters for writing variants

5.1 Standard Arabic and Persian characters

The standard codings for Arabic and Persian are given in Table 5.1 and Table 5.2.

- For long vowels, use the capital letters `<A>`, `<I>`, `<U>` or `<aa>`, `<iy>`, `<uw>`.
- To get the defective writing of long vowels, use `<_a>`, `<_i>`, `<_u>`.
- *'Alif maqṣūra* is `<_A>` or `<Y>`.
- The short vowels *fatḥa*, *kasra*, *ḍamma* are coded `<a>`, `<i>`, `<u>` and need not normally be written except in the following cases:

a	ا	*a*	*'alif*	b	ب	*b*	*bā'*	p	پ	*p*	*pā'*
t	ت	*t*	*tā'*	_t	ث	*t̲*	*t̲ā'*	^g	ج	*ǧ*	*ǧīm*
.h	ح	*ḥ*	*ḥā'*	_h	خ	*ẖ*	*ẖā'*	d	د	*d*	*dāl*
_d	ذ	*d̲*	*d̲āl*	r	ر	*r*	*rā'*	z	ز	*z*	*zāy*
s	س	*s*	*sīn*	^s	ش	*š*	*šīn*	.s	ص	*ṣ*	*ṣād*
.d	ض	*ḍ*	*ḍād*	.t	ط	*ṭ*	*ṭā'*	.z	ظ	*ẓ*	*ẓā'*
‘	ع	‘	*‘ayn*	.g	غ	*ġ*	*ġayn*	f	ف	*f*	*fā'*
q	ق	*q*	*qāf*	v	ڤ	*v*	*vā'*	k	ك	*k*	*kāf*
g	گ	*g*	*gāf*	l	ل	*l*	*lām*	m	م	*m*	*mīm*
n	ن	*n*	*nūn*	h	ه	*h*	*hā'*	w	و	*w*	*wāw*
y	ي	*y*	*yā'*	_A	ى	*ā*	*'alif maqṣūra*	T	ة	*t*	*tā' marbuṭa*

Table 5.1: Standard codings for Arabic and Persian.

  - at the beginning of a word where they generate *'alif*,
  - adjacent to *hamza* where they will influence its carrier,
  - when the transliteration is required,
  - in the `\fullvocalize` mode.

- *Tanwīn* is coded `<aN>`, `<iN>`, or `<uN>`. A silent *'alif*, if required, is supplied automatically; it may also be explicitly written: `<aNA>`. Likewise, a silent *wāw* may be written `<NU>` as in `<‘amruNU>`.

- *hamza* is denoted by a single *right* quote `<'>`. After selecting a language by `\setarab` etc., the *hamza* carrier will be determined from the context according to the rules for writing Arabic words; if that is not wanted, "quote" the *hamza* (see Section 5.2 below). In the `\setverb` mode, the *hamza* carrier is determined by the following letter; see Section 5.5.

- *madda* on *'alif* is generated by a right quote (*hamza*) before `<A>`: `<'A>`.

`c`	ﻉ	*c*	*ḥā'* with *hamza*
`^c`	ﭺ	*č*	*ǧīm* with three dots (below)
`,c`	ﭽ	*ć*	*ḥ̣ā'* with three dots (above)
`^z`	ﮋ	*ž*	*zāy* with three dots (above)
`~n`	ﮒ	*ñ*	*kāf* with three dots (Ottoman)
`~l`	ﻟ̃	*l̃*	*lām* with a bow accent (Kurdish)
`.r`	ﮌ	*ṙ*	*rā'* with two bows (Kurdish)

Table 5.2: Additional codings generally available.

It may also be written `<~A>`; likewise, `<~I>` and `<~U>` will produce *madda* on *yā'* and on *wāw*, as required in some older writing conventions.

- The coding `<‘>` for *‘ayn* is a single *left* quote, beware of confusing it with *hamza*!
- The "invisible consonant" `<|>` may be inserted in order to break unwanted ligatures and to influence the *hamza* writing. It will not show in the Arabic output or in the transliteration. At the beginning of a word it will suppress a following short vowel; otherwise it acts like a consonant.
- The sequence `<||>` will insert a small space, as does `<"|>` (see Section 5.2 below). The adjacent characters will not be connected.
- *Šadda* is indicated by doubling the appropriate letter coding.
- The definite article is separated from the following word by a hyphen. It may be written in the assimilated form (if it exists): `<as-salaamu>`, or always as `<al->`; in that case a subsequent "sun letter" must be doubled: `<al-ssalaamu>`, to receive a *šadda*, and to prevent a *sukūn* on the *lām*. The transliteration in both cases is identical.
- Hyphens `<->` are used for tying words together, or for indicating a connecting vowel in Arabic, or an *iẓāfet* connection in Persian. They may be used freely, and generally do not change the writing, but will show up in the transliteration. Additionally, at the beginning and the end of an

otherwise isolated word they enforce the use of the connecting form of the adjacent letter (if it exists), like e.g. in the date `<1400 h->`.

- A double hyphen `<-->` between two otherwise joining letters will break any ligature and will insert a horizontal stroke (*tatwīl*, *kašīda*) without appearing in the transliteration. It may be used repeatedly. See also Section 8: automatic stretching.

  For special applications, it can also be coded `<B>`; and `<|B>` will behave like an ordinary consonant and may carry vowel indicators, *tanwīn*, *sukūn*, and, in the combination `<|BB>`: *šadda*.

5.2 Quoting

In `\novocalize` mode (see Section 5.4), a double quote `<">` will modify the meaning of the following character as follows:

- if a short vowel follows, the appropriate diacritical mark *fatḥa*, *kasra*, *ḍamma* will be put on the preceding character.
  - If `<N>` follows the short vowel, the appropriate form of *tanwīn* will be generated instead.
  - At the beginning of a word, *'alif* is assumed as the first character.
- if the following character is a single right quote, a *hamza* mark will be put on the preceding character even if in conflict with the *hamza* rules.

  At the beginning of a word, an isolated *hamza* will be generated.
- if the following character is the "invisible consonant" `<|>`, the connection between the adjacent letters will be broken and a small space inserted. This can also be denoted `<||>` instead of `<"|>`.

  At the beginning of a word, *'alif* with *waṣla* will be generated.
- otherwise: a *sukūn* will be put on the preceding character. The following character will be processed again.

The double quote will not show up in the transliteration.

In `\vocalize` mode, (see Section 5.4), quoting will turn a short vowel off; likewise, in `\fullvocalize` mode, quoting will also turn a *sukūn* off. Put differently: quoting will toggle the generation of short vowel indicators and *sukūn* on and off.

5.3 Ligatures

There is no way to explicitly enforce ligatures as a large number of them are generated automatically. The results will not always look satisfactory, so we recommend inspecting the output after the first run. Any unwanted ligature can be suppressed by interposing the invisible character `<|>` between the two letters otherwise combined into a ligature. After `\ligsfalse`, in the middle of a word fewer ligatures will be produced; for some texts this looks better. You can return to the normal strategy by `\ligstrue`.

5.4 Vowelization

There are three modes of rendering short vowels:

- `\fullvocalize`:
  - Every short vowel written will generate the corresponding diacritical mark *fatḥa*, *kasra*, *ḍamma*, except if quoted.
  - If `<N>` follows a short vowel, the corresponding form of *tanwīn* is generated instead.
  - Defective writing: The coding `<_a>` will produce a *Qur'an 'alif* accent (also called *dagger 'alif*) instead of an explicit *'alif* character which would be coded `<A>` or `<aa>`. Likewise, `<_i>` will produce a small *'alif* below the preceding consonant in place of `<I>` (`<iy>`), and `<_u>` will produce an inverted *ḍamma* in place of `<U>` (`<uw>`).
  - If a long vowel follows a consonant, the corresponding short vowel is implied. The long vowel itself carries no diacritical mark.
  - If no vowel is given after a consonant, *sukūn* will be generated except if a double quote precedes the next consonant. The *lām* of the definite article receives no *sukūn* if a double "sun letter" follows.
  - *'alif* at the beginning of a word carries *waṣla* instead of the vowel indicator if the preceding word ended with a vowel.
- `\vocalize`: As above, but *sukūn* and *waṣla* will not be generated except if explicitly indicated by "quoting".
- `\novocalize`: No diacritics will be generated except if explicitly asked for by "quoting".

In all modes, a double consonant will generate *šadda*, and `<'A>` always generates *madda* on *'alif*.

After `<aN>` the silent *'alif* character is generated if necessary. The silent *'alif* may also be explicitly indicated by `<aNA>`, or coded literally as `<A>` in `\novocalize` mode. If a silent *'alif maqṣūra* is wanted instead, write `<aN_A>`, `<aNY>`, `<_A>` or `<Y>`.

The *tanwīn fatḥa* is normally put on the last consonant of the word, even if a silent *'alif* follows. If it is instead supposed to go onto the *'alif* as in some modern Arabic conventions, or in Persian, this behaviour can be achieved by the option `\newtanwin`. The option `\oldtanwin` will restore the classical behaviour.

A silent *'alif* after *wāw* is indicated by `<UA>` or `<WA>` (with a capital `<W>`!).

5.5 Verbatim input

`'a`	أَ	*hamza* on *'alif*	`'i`	إِ	*hamza* below *'alif*
`'w`	ؤَ	*hamza* on *wāw*	`'y`	ئَ	*hamza* on a tooth
`'h`	ۂَ	*hamza* on *hā'*	`'B`	ـٕـ	*hamza* on the line
`'|`	ء	isolated *hamza*	`'A`	آ	*madda* on *'alif*

Table 5.3: Verbatim codings for the carrier of *hamza*

After disabling language specific processing by `\setverb` or `\setnone`, ArabTEX will not use any context information to determine the carrier of *hamza*. Instead the user has to supply this information himself by the next character typed after `<'>`. Generally this character will be used as the carrier; for examples and some exceptions see Table 5.3. A short vowel indicator may follow.

To ease automatic conversion, an initial *'alif* may also be coded `<A>`.

5.6 Alternate input codings

The ArabTEX input notation has been very carefully designed for flexibility, readability, and ease of use for linguists confined to standard 7-bit ASCII equipment for processing and transmitting data. However, it does not make much sense recoding existing machine-readable text files coded according to other standards. Thus, some alternate reading modules have been written (as there

are more than 10 different codings in current use, this is an open-ended activity), and a general code switching procedure has been provided.

An alternate reading module, e.g. `asmo449.sty` for the ASMO 449 code, is installed by adding its name (`asmo449`) as a LaTeX style option, or by `\input asmo449.sty`. Afterwards, a *code_name* (in this case `asmo449`) is defined.

Input coding is switched by the command `\setcode {`*code_name*`}` that changes the coding for *Arabic text* globally, or by the environment `\begin {setcode}{`*code_name*`}` $\cdots$ `\end {setcode}` which follows the normal TeX grouping rules.

Coding may be switched several times in the same document, provided the appropriate reading modules are installed; `\setcode {arabtex}` reverts to the standard ArabTeX notation.

Please observe that *only Arabic text* is affected by `\setcode {`*code_name*`}`; text outside of *Arabic contexts,* and control sequence names, are still assumed to be in 7-bit ASCII. As existing text files presumably do not contain any control sequences or non-Arabic text anyway, we suggest using a small ASCII TeX/LaTeX driver file setting all relevant options and containing any non-Arabic text, and calling the Arabic text files by `\input {`*file_name*`}` from within an *Arabic environment*.

For details on available additional reading modules, see Appendix H.

Chapter 6

Transliteration

6.1 ZDMG transliteration style

In addition to the arabic writing, the standard scientific transliteration may also be obtained from a fully vowelized input text. This mode is activated by `\transtrue` and may be switched off again by `\transfalse`. If *only* the transliteration is wanted, you can deactivate the arabic writing by `\arabfalse`; it can be reactivated by `\arabtrue`. If both modes are active their output will be interleaved line by line.

The transliteration mode assumes that the input text is in the Arabic or Persian language and has been coded according to the rules given above. For words from other languages the transliteration might be in error. For Arabic text, the following special cases are handled:

- after the definite article, a double consonant will be assimilated;
- an initial vowel will be replaced by an apostrophe whenever the preceding word ended with a vowel (in this case a *waṣla* appears in the Arabic writing). If that is not wanted, start with *hamza*.
- a silent *'alif* or *'alif maqṣūra* after `<N>` (*tanwīn*) and `<U>` is omitted in the transliteration. The same happens after *wāw* if it is written as a capital `<W>`.
- To correctly reproduce some historical writings, a silent long vowel after `<_a>` is omitted in the transliteration. For examples, see the Appendix.

For economy of space, the transliteration module is *not* loaded by default. If

you want to use it, add the style option `"atrans"` with LaTeX; and with Plain TeX, say `\input atrans.sty` after loading ArabTeX.

6.2 Encyclopedia of Islam style

For special purposes, the standard transliteration output may be modified by including the LaTeX option `"etrans"`, or by loading the file `"etrans.sty"` when working with Plain TeX. After this modification, the transliteration will follow the style of the Encyclopedia of Islam.

Chapter 7

Support for other languages besides Arabic

ArabTEX is primarily intended for typesetting texts in classical and modern Arabic, but it also provides some support for several other languages that are customarily written in the Arabic alphabet.

In order to switch to the conventions for one of these languages, say `\setfarsi`, `\seturdu`, `\setpashto`, `\setmaghribi`; `\setverb` will switch off any language specific processing. `\setarab` can be used to switch back to the Arabic conventions. After selecting the language, `<` and `>` serve as delimiters for quotations; `\setnone` will, like `\setverb`, deselect any language, and will also return `<` and `>` to their normal TEX meaning.

This part of ArabTEX relies heavily on contributions from the user community; we want to especially mention Ivan Dershanski who completely reimplemented the routines for processing Persian. As we extensively modified these contributions while integrating the system, we are solely responsible for any remaining, or newly introduced, errors.

7.1 Persian (Farsi, Dari), also Ottoman, Kurdish

- All characters needed for writing Farsi are available by default. The short vowels `<e>` and `<o>` are mapped to `<i>` and `<u>`, the long vowels `<E>` and `<O>` to `<I>` and `<U>` without a vowel indicator. `<H>` denotes final silent *hā'*. This *hā'* receives no *sukūn* even in fully vowelized mode.

- For *fatḥa* or *kasra* followed by a final silent *hā'* you can also write `<,a>` or `<,e>` in place of `<aH>` and `<eH>`.
- The *iẓāfet* connection may always be written `<-i>` or `<-e>` (with hyphen); then the correct spelling will be determined from the context. Likewise the *yā'-i-waḥdat* can always be written `<-I>` or `<-E>`.
- The present tense forms of the copula are coded `<-am>`, `<-I>`, `<-ast>`, `<-Im>`, `<-Id>`, `<-and>`. In the output they are written as separate words after a little space.
- The final *yā'* carries no dots. Farsi uses the Nasta'liq font if available, otherwise Naskh.

For further details see Appendix G.

7.2 Urdu

- For Urdu, additional codings are available, see Table 7.1. Some of the given codings also occur in Pashto but with a different meaning, see Section 7.3.
- The short vowels `<e>` and `<o>` are mapped to `<i>` and `<u>`. `<H>`, `<,a>` and `<,e>` are used as in Persian.
- Even in fully vowelized mode, an aspirated consonant before `<h>` receives no *sukūn* since the two are technically a single letter.
- Urdu uses the Nasta'liq font if available, otherwise Naskh.

7.3 Pashto (Afghanic)

- For Pashto, additional codings are available, see Table 7.2. Some of the given codings also occur in Urdu but with a different meaning, see Section 7.2.
- The short vowel `<e>` is indicated by a *zwarakay*, `<o>` by an inverted *ḍamma*.

  Observe also the following codings:

  `<w"'>` *hamza* on *wāw*
  `<h"'>` *hamza* on *hā'*, if not generated by *iẓāfet*

h	ھ	*h*	always denotes the "two-eyed" *hā'*
,h	ہ	*h*	the "wavy" *hā'* letter
,t	ٹ	*t́*	*tā'* with a small *ṭā'* accent
,d	ڈ	*d́*	*dāl* with a small *ṭā'* accent
,r	ڑ	*ŕ*	*rā'* with a small *ṭā'* accent
.n	ں	*ṇ*	*nūn* without a dot
E	ے	*ē*	*ē*, *yā' barī'* in the final position
ae	ۓ	*ae*	the diphtong *ae*
ao	ۄ	*ao*	the diphtong *ao*
O	ھو	*ō*	the long vowel *ō*
U	ہُو	*ū*	the long vowel *ū*

Table 7.1: Additional codings for Urdu.

- The codings `<H>`, `<,a>` and `<,e>` are used as in Persian. The rules for *iẓāfet* and *yā'-i-waḥdat* apply.
- For writing some Pashto words in the Urdu style, write the command `\seturdu` and afterwards switch back to the Pashto conventions by `\setpashto`.

7.4 Maghribi

Nearly like Arabic but using a different writing convention. *fā'* is written with one dot below the letter, *qāf* with one dot above the normal letter form of *fā'*. The three dots of *vā'* are put below the letter.

,t	ټ	*ť*	*tā'* with a small loop
,d	ډ	*ď*	*dāl* with a small loop
,r	ړ	*ŕ*	*rā'* with a small loop
.n	ڼ	*ṇ*	*nūn* with a small loop
g	ګ	*g*	*gāf* with a small loop instead of a bar
,z	ږ	*ź*	*rā'* with one dot above and one below
,s	ښ	*ś*	*sīn* with one dot above and one below
ae	ۍ	*ae*	the diphtong *ae*
Ee	ې	*ey*	the diphtong *ey*
ee	ۍ	*ey*	the diphtong *ey*
E	ې	*ē*	the long vowel *ē*
O	ۏ	*ō*	the long vowel *ō*
U	ۉ	*ū*	the long vowel *ū*

Table 7.2: Additional codings for Pashto.

7.5 Other languages

This is up to experimentation by the user. If `\setarab` or `\setfarsi` will not produce the desired result, try `\setverb` for verbatim mode.

The vowelization and the transliteration cannot generally be expected to be correct, but might work by accident.

In case some character variants not yet provided are needed, feel free to ask the author for help. There is no simple way for the user to modify the script.

Chapter 8

Miscellaneous features

8.1 Automatic stretching

For special purposes, e.g. for headlines and for Arabic paragraphs containing long mathematical or non-Arabic insertions, the connection between adjacent Arabic letters may be made "elastic", if they form no ligature. Thus a *kašīda* is inserted whose length will be adjusted automatically to uniformly fill the output line.

This feature very easily leads to storage overflow during the processing, and should only be used whenever necessary. It is switched on with `\spreadtrue` and switched off again with `\spreadfalse`. Inside an *Arabic Environment,* it will also be switched off automatically at the end of every paragraph.

8.2 Dots on *yā'*

Whether *yā'* in the final position carries dots or not is controlled by the chosen language convention. You can override this, after selecting the language, by `\yahdots` and `\yahnodots`.

8.3 Additional codings

To reproduce exotic, erroneous or archaic texts exactly as they are written, some additional codings are available, see Table 8.1.

`.k`	ک	*k*	*kāf* in the final position without a mark
`^d`	ډ	*ď*	*dāl* with a dot below
`.f`	ف	*f̣*	*fā'* without a dot
`.b`	ب	*b̤*	*bā'* without a dot
`.n`	ں	*n̤*	*nūn* without a dot (not available in Pashto mode)
`Y`	ى	*ā*	*'alif maqṣūra*; *yā'* without dots in all positions

Table 8.1: Additional codings for special purposes.

If further variants are needed, write to the author and indicate:

- the required shape,
- the assumed transliteration,
- a suggestion for the input coding,
- some information on the intended use.

We are willing to consider any suggestion. Adding a new character might be easy, or else it might be impossible. ArabTEX is flexible, but there are some technical limitations.

8.4 Progress report

As ArabTEX is slow, it will produce some terminal output while running to indicate it is still alive. If that is not wanted, e.g., on a very fast system, or while running a batch job, say `\quiet` or `\tracingarab = 0` (outside an *Arabic Environment*; otherwise say `\doassign {\tracingarab }{0}`). `\tracingarab = 1` will only report Arabic paragraphs, a value of 2: Arabic lines and insertions, a value of 3 or more: individual *Arabic items*.

8.5 Verbatim copy of the input

For test purposes, the Arabic input may be reproduced verbatim after `\showtrue` in addition to the normal output; `\showfalse` switches this feature off again. Commands will not usually be shown. The output will generally not look pleasant, and this feature is only provided in order to trace down errors, or to demonstrate the operation of ArabTEX as in the appendix.

8.6 Using ArabTEX with EDMAC

ArabTEX will cooperate with EDMAC, a Plain TEX macro package for critical editions, written by John Lavagnino and Dominik Wujastyk. If EDMAC is already present when ArabTEX is loaded, the EDMAC commands will, after suitable modifications, be available inside an *Arabic environment.* Their arguments are considered Roman text but may contain *Arabic quotations.*

For further details, see the EDMAC documentation.

Chapter 9

Acknowledgments

The development of ArabTEX would not have been possible without the assistance of many people, and it is impossible to acknowledge every individual contribution. Besides our local team, i.e. Udo Merkel and Heribert Schlebbe, helpful advice came, among others, from Chahriar Assad, Benno van Dalen, Ivan Derzhanski, Wolfdietrich Fischer, Ahmed El-Hadi, Yannis Haralambous, Abdelsalam Heddaya, Nicholas Heer, Iqbal Khan, Tom Koornwinder, Eberhard Krüger, Asif Lakehsar, Jan Lodder, Richard Lorch, Pierre MacKay, Eberhard Mattes, Fathy Neamat-Allah, Bernd Raichle, Ulrich Rebstock, Mohamed Saba, Waheed Samy, Annemarie Schimmel, Nariman Shehab, Dominik Wujastyk, and Michio Yano. We also have to thank all users who sent error reports, comments, and suggestions.

Chapter 10

References

B. Alavi, M. Lorenz: *Lehrbuch der persischen Sprache.*
5. Auflage 1988. VEB Verlag Enzyklopädie, Leipzig.

A. A. Ambros: *Einführung in die moderne arabische Schriftsprache.*
1. Auflage 1969. Max Hueber Verlag, München.

ASMO 449: *7-bit coded Arabic character set for information interchange.*
Arabic Standards and Measurements Organization, 1982.

J. D. Becker: *Arabic Word Processing.*
Comm. ACM **30/7,** 600-610 (1987).

T. Borg: *Arabisch für Ausländer. Ein Lehrbuch für modernes Hocharabisch.*
2. Auflage 1979. Verlag Borg GmbH, Hamburg.

J. A. Boyle: *Grammar of Modern Persian.*
Wiesbaden: Otto Harrassowitz, 1966.

B. Comrie (ed.): *The World's Major Languages.*
Croom Helm, London 1987.

DIN 31 635: *Umschrift des Arabischen Alphabets.*
Deutsches Institut für Normung e.V., 1982.

J. Lavagnino and D. Wujastyk: *An Overview of EDMAC: A* `plain` *TEX format for critical editions.*
TUGboat **11/4,** 623-643 (1990).

L. P. Elwell-Sutton: *Elementary Persian Grammar.*
Cambridge University Press, 1963.

C. Faulmann: *Das Buch der Schrift, enthaltend die Schriften und Alphabete aller Zeiten und aller Völker des gesammten (sic!) Erdkreises.*
K. K. Hof- und Staatsdruckerei, Wien 1878.

W.D. Fischer: *Grammatik des Klassischen Arabisch.*
2. Auflage 1987. Verlag Otto Harrassowitz, Wiesbaden.

A. Grohmann: *Arabische Paläographie (Teil I und II).*
Österreichische Akademie der Wissenschaften, Philosophisch-historische Klasse, Denkschriften 94, 1. Wien 1967.

E. Harder, A. Schimmel: *Arabische Sprachlehre.*
15. Auflage 1983. Julius Groos Verlag, Heidelberg.

هاشم محمّد الخطّاط ، قواعد الخطّ العربيّ.
Hāšim Muḥammad al-Ḫaṭṭāṭ: Qawā'id al-Ḫaṭṭi al-'Arabī.
Maktaba an-Nahḍa, Baghdad; Dār al-Qalam, Beirut, 1400/1980.

ISO/R 233 - 1961: *International System for the Transliteration of Arabic Characters.*
International Standards Institution, 1961.

ISO 8859 - 6: *Information processing — 8-bit single-byte coded graphic character sets — Part 6: Latin/Arabic alphabet.*
International Organization for Standardization, 1987.

ISO 9036: *Information processing — Arabic 7-bit coded character set for information interchange.*
International Organization for Standardization, 1987.

D. E. Knuth: *The* METAFONT*book.*
Addison Wesley Publishing Comp., Reading, Mass., 1986.

D. E. Knuth: *The TEXbook.*
Sixth printing. Addison Wesley Publishing Comp., Reading, Mass., 1986.

D. E. Knuth and P. MacKay: *Mixing right-to-left texts with left-to-right texts.*
TUGboat **8/1,** 14-25 (1987).

Ann K. S. Lambton: *Persian Grammar.*
Cambridge University Press, 1953.

L. Lamport: LaTeX, *A Document Preparation System.*
Addison Wesley Publishing Comp., Reading, Mass., 1986.

M. Lorenz: *Lehrbuch des Pashto (Afghanisch).*
2. Auflage 1982. VEB Verlag Enzyklopädie, Leipzig.

P. A. MacKay: *Typesetting Problem Scripts.*
BYTE **11/2,** 201-216 (1986).

H. Ritter: *Über einige Regeln, die beim Drucken mit arabischen Typen zu beachten sind.*
ZDMG **100/2,** 577-580 (1951).

Friedrich Rückert: *Grammatik, Poetik und Rhetorik der Perser.*
Wiesbaden: Otto Harrassowitz, 1966.

C. Salemann, V. Shukovski: *Persische Grammatik.*
4. Auflage 1947. Verlag Otto Harrassowitz, Leipzig.

A. Schimmel: *Islamic Calligraphy.*
E.J.Brill, Leiden, Netherlands 1970.

H.J. Vermeer, W. Akhtar, A. Akhtar: *Urdu-Lautlehre und Urdu-Schrift.*
3. Auflage 1985. Julius Groos Verlag, Heidelberg.

Appendix A

Obtaining ArabTEX

The ArabTEX system is available from the author's institution (by anonymous FTP from `ftp.informatik.uni-stuttgart.de (129.69.211.2)`, in the directory `pub/arabtex`) and from many other common servers, e.g. the CTAN network (Aston, Niord, Stuttgart). The files may be transferred individually or as a package: `arabtex.zip` for PC systems, `arabtex.tar.Z` for U*IX systems; we recommend to get and inspect the `README` file first. Successfull operation on the Apple Macintosh in conjunction with OzTEX has also been reported.

At the time of this writing, version 3.00 is current. The Nasta'liq font is still under development; Naskh will be substituted automatically. Version 2 is downward compatible; the old version 1 is obsolete and should no more be used.

Appendix B

Installing ArabTEX

The installation procedure is strongly system dependent, and we recommend securing the assistance of a local TEXpert. You have to install the `"nash14"` font with its `"*.pk"` and `"*.tfm"` files on the font search path of your TEX system, and the `"*.sty"` files and `"arabtex.tex"` on the source search path (usually TEXINPUT) of your system. Possibly you will also have to rename the `"*.pk"` files according to local conventions, and as a last resort you can try to recreate the fonts from the `"*.mf"` METAFONT sources. Additional fonts, whenever available, are installed analogously.

ArabTEX has been found to cooperate well with TEX versions 3.xxx, LATEX versions 2.09 of 1991 or later, NFSS and NFSS2 (not required), and previewers that can handle fonts of more than 128 characters. TEX-XET or TEX--XET are not required, and their additional features are presently not exploited. The TEX "hash size" should be at least 3000 to 3500, especially when using ArabTEX in conjunction with LATEX, and if the transliteration module is used. Use of a BIG TEX may be necessary when using the NFSS2 due to the latter's high demand on string storage. Space and time requirements are not negligible, and have increased during development; however, ArabTEX currently still runs, albeit slowly, even on a PC XT standard configuration.

Appendix C

Release history

There was a Version 1 which is no more supported.

Version 2 was not fully compatible with Version 1; however, moving to the new version usually caused little problems. Apart from some extensions, most changes were introduced in order to better conform to the transliteration standards, and to have less compatibility problems with TEX and LATEX. Further versions are expected to be upward compatible if no serious problems will turn up.

The main differences between versions 1 and 2 are:

- The font size has increased, so the document layout may change. The old font `"nash10"` can no more be used as the character locations have been assigned differently.
- Some Arabic characters are now coded differently: *'ayn* is denoted by a left quote, and `<c>`, `<^z>`, `<^t>`, and `<.n>` have been assigned new meanings in order to better conform to the standard transliteration.
- There are many more ligatures than before. This normally need not concern the user.
- `\vocalize` will no more generate *sukūn* and *waṣla* except if explicitly indicated by quoting. See `\fullvocalize`.
- Arabic Environments are now always bracketed by the new control sequences `\begin{arabtext}` and `\end{arabtext}` even if only the transliteration is wanted.

We strongly recommend converting any still existing version 1 input files to the new notation. To assist in this migrating procedure, the LATEX option

`"oldarabtex"` and/or the command `\oldarabtex` will switch to a mode where virtually all places where the old conventions are used, will either produce a TEX error message or will be flagged in the output.

The changes introduced since the release of Version 2.00 up to now (Version 3.00) fall into one of two categories: error corrections, and upward compatible extensions. Details are not given here, but are documented in the text file CHANGES that is part of the distribution package of ArabTEX.

Version 3 is upwards compatible with version 2. All supported features are documented in this manual.

Appendix D

Sample ArabTEX input

```
\documentstyle[12pt,arabtex]{article}
\begin{document}

\setarab    % choose the language conventions
\vocalize   % diacritics for short vowels on
\transtrue  % additionally switch on the transliteration
\arabtrue   % print arabic text ... is on anyway
\spreadtrue % spread out caption

\centerline {<^gu.hA wa-.himAruhu>}

\begin{arabtext}
'at_A .sadIquN 'il_A ^gu.hA ya.tlubu minhu .himArahu li-yarkabahu
fI safraTiN qa.sIraTiN wa-qAla lahu:

sawfa 'u`Iduhu 'ilayka fI al-masA'i, wa-'adfa`u laka 'u^graTaN. \\
fa-qAla ^gu.hA:

'anA 'AsifuN ^giddaN 'annI lA 'asta.tI`u 'an 'u.haqqiqa
laka ra.gbataka, fa-al.himAru laysa hunA al-yawma. \\
wa-qabla 'an yutimmu ^gu.hA kalAmahu
bada'a al-.himAru yanhaqu fI i.s.tablihi. \\
fa-qAla lahu .sadIquhu:

'innI 'asma`u .himAraka yA ^gu.hA yanhaqu. \\
fa-qAla lahu ^gu.hA:

.garIbuN 'amruka yA .sadIqI!
'a-tu.saddiqu al-.himAra wa-tuka_d_dibunI?
\end{arabtext}

\end{document}
```

Appendix E

Sample ArabTEX output

جُحَا وَحِمَارُهُ *ǧuḥā wa-ḥimāruhu*

ʾatā ṣadīqun ʾilā ǧuḥā yaṭlubu minhu ḥimārahu li-yarkabahu fī safratin qaṣīratin wa-qāla lahu:

أَتَى صَدِيقٌ إِلَى جُحَا يَطلُبُ مِنهُ حِمَارَهُ لِيَركَبَهُ فِي سَفرَةٍ قَصِيرَةٍ وَقَالَ لَهُ :

sawfa ʾuʿīduhu ʾilayka fī ʾl-masāʾi , wa-ʾadfaʿu laka ʾuǧratan.

سَوفَ أُعِيدُهُ إِلَيكَ فِي المَسَاءِ ، وَأَدفَعُ لَكَ أُجرَةً .

fa-qāla ǧuḥā:

فَقَالَ جُحَا :

ʾanā ʾāsifun ǧiddan ʾannī lā ʾastaṭīʿu ʾan ʾuḥaqqiqa laka raġbataka, fa-ʾlḥimāru laysa hunā ʾl-yawma.

أَنَا آسِفٌ جِدًّا أَنِّي لَا أَستَطِيعُ أَن أُحَقِّقَ لَكَ رَغبَتَكَ ، فَالحِمَارُ لَيسَ هُنَا اليَومَ .

wa-qabla ʾan yutimmu ǧuḥā kalāmahu badaʾa ʾl-ḥimāru yanhaqu fī ʾṣtablihi.

وَقَبلَ أَن يُتِمَّ جُحَا كَلَامَهُ بَدَأَ الحِمَارُ يَنهَقُ فِي اصطَبلِهِ .

fa-qāla lahu ṣadīquhu:

فَقَالَ لَهُ صَدِيقُهُ :

ʾinnī ʾasmaʿu ḥimāraka yā ǧuḥā yanhaqu.

إِنِّي أَسمَعُ حِمَارَكَ يَا جُحَا يَنهَقُ .

fa-qāla lahu ǧuḥā:

فَقَالَ لَهُ جُحَا :

ġarībun ʾamruka yā ṣadīqī! ʾa-tuṣaddiqu ʾl-ḥimāra wa-tukaḏḏibunī?

غَرِيبٌ أَمرُكَ يَا صَدِيقِي ! أَتُصَدِّقُ الحِمَارَ وَتُكَذِّبُنِي ؟

Appendix F

Coding examples for Arabic[1]

The short vowels *fatḥa*, *kasra*, *ḍamma* are denoted, as in the transliteration, by the small letters `a, i, u`:

`mana‘a` مَنَعَ *manaᶜa*, `_dahaba` ذَهَبَ *ḏahaba*, `^sariba` شَرِبَ *šariba*,
`qabila` قَبِلَ *qabila*, `‘a.zuma` عَظُمَ *ᶜaẓuma*, `‘alu` عَلُ *ᶜalu*, `bal` بَلْ *bal*,
`ni‘ma` نِعْمَ *niᶜma*, `yaktub` يَكْتُبْ *yaktub*.

The long vowels *ā, ī, ū* are denoted by capitals `A, I, U` or by `aa, iy, uw`:

`qAtala` قَاتَلَ *qātala*, `nUzi‘a` نُوزِعَ *nūziᶜa*, `lUmI` لُومِي *lūmī*,
`sIrI` سِيرِي *sīrī*; `lawmI` لَوْمِي *lawmī*, `sayrI` سَيْرِي *sayrī*.

Alif maqṣūra is coded as `_A` or `Y`.

`ramY` رَمَى *ramā*, `_dikrY` ذِكْرَى *ḏikrā*, `‘al_A` عَلَى *ᶜalā*, `bal_A` بَلَى *balā*.

Silent *’alif*: The plural suffixes *-ū, -aw* of the verb are denoted `UA`, `aW` or `aWA`:

`katabUA` كَتَبُوا *katabū*, `yaktubUA` يَكْتُبُوا *yaktubū*,
`ramaWA` رَمَوْا *ramaw*, `yalqaW` يَلْقَوْا *yalqaw*.

[1] **Most of the examples are taken from: Wolfdietrich Fischer, Grammatik des Klassischen Arabisch, 2. Auflage, Verlag Otto Harrassowitz, Wiesbaden 1987.**

The defective notation of *ā, ī, ū* can be indicated by `_a`, `_i`, `_u` and leads to the appropriate spelling:

`dAru-h_u` دَارُهُ *dāru-hū*, `ri^gli-h_i` رِجْلِهِ *riğli-hī*,

however: `ramA-hu` رَمَاهُ *ramā-hu*, `yarmI-hi` يَرْمِيهِ *yarmī-hi*;

`_dih_i` ذِهِ *ḏihī*, `h_a_dih_i` هٰذِهِ *hāḏihī*, `tih_i` تِهِ *tihī*, `hAtih_i` هَاتِهِ *hātihī*,

`rabb_i` رَبِّ *rabbī*, `.sAl_i` صَالِ *ṣālī*; `hum_u` هُمُ *humū*;

`qiy_amaTuN` قِيٰمَةٌ *qiyāmatun*, `'il_ahuN` إِلٰهٌ *ʾilāhun*,

`sam_awATuN` سَمٰوَاتٌ *samāwātun*, `_tal_a_tuN` ثَلٰثُ *ṯalāṯun*,

`l_akin` لٰكِنْ *lākin*, `h_a_dA` هٰذَا *hāḏā*, `'al-ll_ahu` أَللّٰهُ *ʾal-lāhu*,

`'al-rra.hm_anu` أَلرَّحْمٰنُ *ʾar-raḥmānu*, `_d_alika` ذٰلِكَ *ḏālika*.

To reproduce the historical writing correctly, a silent long vowel or *ʾalif maqṣūra* after `_a` receives no *sukūn* and is ignored in the transliteration:

`.sal_aUTuN` صَلٰوةٌ *ṣalātun*, `.hay_aUTuN` حَيٰوةٌ *ḥayātun*,

`zak_aUTuN` زَكٰوةٌ *zakātun*, `mi^sk_aUTuN` مِشْكٰوةٌ *miškātun*,

`ar-rib_aU` اَلرِّبٰو *ar-ribā*, `tawr_aITuN` تَوْرٰيةٌ *tawrātun*,

`ram_aYhu` رَمٰهُ *ramāhu*, `sIm_aYhum` سِيمٰهُمْ *sīmāhum*.

The short vowel *u* can be written as a long vowel by `_U`:

`'_UlY` أُولَى *ʾulā*, `'_UlA'i` أُولَاءِ *ʾulāʾi*, `'_UlU` أُولُو *ʾulū*,

`'_UlAka` أُولَاكَ *ʾulāka*, `'_UlA'ika` أُولَائِكَ *ʾulāʾika*.

***Tanwīn*:** The plural suffixes *-un, -in, -an* are written `-uN`, `-iN`, `-aN` or `aNA`. Silent *ʾalif* in *-an* may be indicated by `A` or omitted; if necessary it is supplied from the context.

`ra^guluN` رَجُلٌ *rağulun*, `ra^guliN` رَجُلٍ *rağulin*, `ra^gulaN` رَجُلًا *rağulan*,

`madInaTaN` مَدِينَةً *madīnatan*, `^gamIlaTaN` جَمِيلَةً *ğamīlatan*,

`'i_daN` إِذًا *ʾiḏan*, `samA'aN` سَمَاءً *samāʾan*.

There is a special case:

`ribaNU` رِبًو *riban*; `'amruNU` عَمْرٌو *ʿamrun*, `'amriNU` عَمْرٍو *ʿamrin*,

however: `'amraN` عَمْرًا *ʿamran*.

Tanwīn fatḥa is traditionally put on the last consonant even if a silent *'alif* follows. Some modern conventions, and also Persian practice, require to put it on the *'alif* in this case. This behaviour may be switched on by `\newtanwin`, and off by `\oldtanwin`. `\newtanwin` mode is the default for Persian.

`ra^gulaN` رَجُلاً *raǧulan*, `'i_daN` إِذاً *ʾiḏan*.

A silent *'alif maqṣūra* after *tanwīn* is written `aNY` or `aN_A`:

`hudaNY` هُدًى *hudan*, `fataN_A` فَتًى *fatan*;

compare:

`al-hudY` اَلْهُدَى *al-hudā*, `'al-fat_A` أَلْفَتَى *ʾal-fatā*.

Tā' marbuṭa is denoted by `T`:

`kalimaTuN` كَلِمَةٌ *kalimatun*, `kalimaTiN` كَلِمَةٍ *kalimatin*,

`kalimaTaN` كَلِمَةً *kalimatan*; `fatATuN` فَتَاةٌ *fatātun*,

`fatATiN` فَتَاةٍ *fatātin*, `fatATaN` فَتَاةً *fatātan*.

Hamza is indicated by `'`; the appropriate carrier is determined by the context:

`'amruN` أَمْرٌ *ʾamrun*, `'ibiluN` إِبِلٌ *ʾibilun*, `'u_htuN` أُخْتٌ *ʾuḫtun*;

`ra'suN` رَأْسٌ *raʾsun*, `'ar'asu` أَرْأَسُ *ʾarʾasu*, `sa'ala` سَأَلَ *saʾala*,

`qara'a` قَرَأَ *qaraʾa*; `bu'suN` بُؤْسٌ *buʾsun*, `'ab'usuN` أَبْؤُسٌ *ʾabʾusun*,

`ra'ufa` رَؤُفَ *raʾufa*, `ru'asA'u` رُؤَسَاءُ *ruʾasāʾu*; `bi'ruN` بِئْرٌ *biʾrun*,

`'as'ilaTuN` أَسْئِلَةٌ *ʾasʾilatun*, `ka'iba` كَئِبَ *kaʾiba*, `qA'imuN` قَائِمٌ *qāʾimun*,

`ri'AsaTuN` رِئَاسَةٌ *riʾāsatun*, `su'ila` سُئِلَ *suʾila*; `samA'uN` سَمَاءٌ *samāʾun*,

`barI'uN` بَرِيءٌ *barīʾun*, `sU'uN` سُوءٌ *sūʾun*, `bad'uN` بَدْءٌ *badʾun*,

`^say'uN` شَيْءٌ *šayʾun*, `^say'iN` شَيْءٍ *šayʾin*, `^say'aN` شَيْئًا *šayʾan*;

`sA'ala` سَاءَلَ *sāʾala*, `mas'alaTuN` مَسْأَلَةٌ *masʾalatun*,

`saw'aTuN` سَوْأَةٌ *sawʾatun*, `_ha.tI'aTuN` خَطِيئَةٌ *ḫaṭīʾatun*.

Old *Hamza* convention: In an older writing style that is used, e.g., in some Qur'an editions, the *hamza* is sometimes put below its carrier or on the connecting line. This style may be switched on by `\oldhamza` (and off again by `\newhamza`):

`'as'ilaTuN` أَسْئِلَةٌ *ʾasʾilatun*, `ka'iba` كَئِبَ *kaʾiba*, `qA'imuN` قَائِمٌ *qāʾimun*,

`su'ila` سُئِلَ *suʾila*, `^say'aN` شَيْئًا *šayʾan*, `_ha.tI'aTuN` خَطِيئَةٌ *ḫaṭīʾatun*.

Madda in the context *'ā* is generated automatically:

`'AkiluN` آكِلٌ *ʾākilun*, `qur'AnuN` قُرْآنٌ *qurʾānun*, `ra'Ahu` رَآهُ *raʾāhu*.

To reproduce the historic writing correctly, it can also be explicitly written in other contexts:

`'a.sdiq~A'uh_u` أَصْدِقَآؤُهُ *ʾaṣdiqāʾuhū*;

`ya^g~I'u` يَجِيءُ *yaǧīʾu*, `s~U'ila` سُوئِلَ *sūʾila*.

***Šadda*:** A double consonant must be written twice, even if it is coded by more than one character:

`nazzala` نَزَّلَ *nazzala*, `ba^s^sAruN` بَشَّارٌ *baššārun*, `nawwara` نَوَّرَ *nawwara*,

`sayyiduN` سَيِّدٌ *sayyidun*, `sa''AluN` سَأَّالٌ *saʾʾālun*,

`.sabiyyuN` صَبِيٌّ *ṣabiyyun*, `'aduwwuN` عَدُوٌّ *ʿaduwwun*.

Instead of `iyy, uww` one can also write `Iy, Uw`:

`.sabIyuN` صَبِيٌّ *ṣabīyun*, `'adUwuN` عَدُوٌّ *ʿadūwun*.

Assimilation: the definite article may be always written `al-`; a following "sun letter" must be written twice like in the Arabic spelling. The transliteration and the use of *sukūn* are adjusted accordingly:

`'al-ddAru` أَلدَّارُ *ʾad-dāru*, `'al-rra^gulu` أَلرَّجُلُ *ʾar-raǧulu*,

`'al-ssanaTu` أَلسَّنَةُ *ʾas-sanatu*, `'al-nnAru` أَلنَّارُ *ʾan-nāru*;

`'al-^gAru` أَلْجَارُ *ʾal-ǧāru*, `'al-bAbu` أَلْبَابُ *ʾal-bābu*;

`'al-llaylaTu` أَللَّيْلَةُ *ʾal-laylatu*, `'al-llisAnu` أَللِّسَانُ *ʾal-lisānu*,

`'al-ll_ahu` أَللهُ *ʾal-lāhu*.

The article may also be written in the assimilated form, with identical result:

`'ad-dAru` أَلدَّارُ *ʾad-dāru*, `'ar-ra^gulu` أَلرَّجُلُ *ʾar-raǧulu*,

`'as-sanaTu` أَلسَّنَةُ *ʾas-sanatu*, `'an-nAru` أَلنَّارُ *ʾan-nāru*.

In some special cases the literal spelling must be used:

`'alla_dI` أَلَّذِي *ʾalladī*, `'alla_dIna` أَلَّذِينَ *ʾalladīna*, `'allatI` أَلَّتِي *ʾallatī*;

however:

`'al-lla_dAni` أَللَّذَانِ *ʾal-laḏāni*, `'al-llatAni` أَللَّتَانِ *ʾal-latāni*,

`'al-llawAtI` أَللَّوَاتِي *ʾal-lawātī*.

***Waṣla*:** an auxiliary vowel at the beginning of a word is always written, but in the middle of a sentence generally without *hamza*. If a vowel precedes the word, the auxiliary vowel will be omitted in the transliteration, and the *waṣla* sign will be used in the spelling:

`wa-ismuhu` وَٱسْمُهُ *wa-ʾsmuhu*, `f--a-in.sarafa` فٱنْصَرَفَ *fa-ʾnṣarafa*.[2]

This also works across word boundaries:

`yA ibnI` يَا ٱبْنِي *yā ʾbnī*, `h_a_dA ibnuh_u` هٰذَا ٱبْنُهُ *hāḏā ʾbnuhū*,

`qAla u_hru^g` قَالَ ٱخْرُجْ *qāla ʾḫruǧ*.

An auxiliary vowel at the end of the preceding word may be separated by a hyphen:

`qad-i in.sarafa` قَدِ ٱنْصَرَفَ *qad-i ʾnṣarafa*,

`ra'aW-u al-bAba` رَأَوُا ٱلْبَابَ *raʾaw-u ʾl-bāba*,

`min-i ibnih_i` مِنِ ٱبْنِهِ *min-i ʾbnihī*.

This also works for the article preceding *ʾalif al-waṣl*:

`'al-i-ismu` أَلِٱسْمُ *ʾal-i-ʾsmu*, `'al-i-i^stirA'u` أَلِٱشْتِرَاءُ *ʾal-i-ʾštirāʾu*,

and even if the auxiliary vowel is omitted in the spelling:

`ra^guluN-i ibnatuh_u ^gamIlaTuN` رَجُلٌ ٱبْنَتُهُ جَمِيلَةٌ *raǧulun-i ʾbnatuhū ǧamīlatun*,

`mu.hammaduN-i al-qura^sIyu` مُحَمَّدٌ ٱلْقُرَشِيُّ *muḥammadun-i ʾl-qurašīyu*.

[2] In vowelized writing, it may sometimes be advisable to introduce a *kašīda* to prevent the vowel marks from bumping into each other.

The particles *li-* and *la-* must be combined with the article except before *lām*:

`lil-rra^guli` لِلرَّجُلِ *lir-raǧuli*, `lal-ma|^gdu` لَلْمَجْدُ *lal-maǧdu*;[3]

however:

`li-llaylaTi` لِلَّيْلَةِ *li-llaylati*, `li-ll_ahi` لِلّٰهِ *li-llāhi*.

The Name of God is written with a special ligature if it is recognized from the input sequence `ll_ah`:

`'al-ll_ahu` أَللّٰهُ *ʾal-lāhu*, `ta-al-ll_ahi` تَٱللّٰهِ *ta-ʾl-lāhi*.

Increased spacing (*Tatwīl*) between adjoining characters may be produced by a double hyphen `--`; note the position of the vowel marks:

`qabila` قَبِلَ *qabila*, `qa--bi--la` قَـبِـلَ *qabila*, `q--ab--ila` قَـبِيلَ *qabila*.

`q--a--b--i--la` قَـبِـيـلَ *qabila*, `qa----bi----la` قَـبِـلَ *qabila*

Ties between words are indicated by a single hyphen:

`bi-baladiN` بِبَلَدٍ *bi-baladin*, `ta-al-ll_ahi` تَٱللّٰهِ *ta-ʾl-lāhi*,

`sa-ya'tI` سَيَأْتِي *sa-yaʾtī*, `li-yafra.ha` لِيَفْرَحَ *li-yafraḥa*,

`wa-iswadda` وَٱسْوَدَّ *wa-ʾswadda*, `ba'da-mA` بَعْدَمَا *baʿda-mā*,

`.tAla-mA` طَالَمَا *ṭāla-mā*, `fI-ma` فِيمَ *fī-ma*, `'alA-ma` عَلَامَ *ʿalā-ma*.

A single hyphen at the beginning or end of a word will enforce the use of the joining form of the first resp. the last character, if that form exists (for special uses only):

`s` س *s*, `-s` س *-s*, `-s-` سـ *-s-*, `s-` سـ *s-*

`h` ه *h*, `-h` ه *-h*, `-h-` هـ *-h-*, `h-` هـ *h-*

`d` د *d*, `-d` د *-d*, `lA` لا *lā*, `-lA` لا *-lā*

`1400 h-` ١٤٠٠ هـ *1400 h-*

Digit sequences are written in the natural order:

`1234567890` ١٢٣٤٥٦٧٨٩٠ *1234567890*

[3] The ligature otherwise produced automatically looks ugly and has been broken by `|`.

Ligatures are generated automatically; they can be suppressed by `|`:

`'al-'islAmu` أَلإِسْلَامُ *ʾal-ʾislāmu*;

`'al-^gAru` أَلْجَارُ *ʾal-ǧāru*, `'al|^gAru` أَلْجَارُ *ʾalǧāru*;

`_tumma` ثُمَّ *t̲umma*, `_tu|mma` ثُمَّ *t̲umma*;

`mu.hammaduN` مُحَمَّدٌ *muḥammadun*, `mu|.ha|mmaduN` مُحَمَّدٌ *muḥammadun*.

Abbreviations and emphasis are indicated by `\emphasize`:

`\emphasize .sl'm` $\overline{\text{صلعم}}$ *ṣlʿm*

`\emphasize ab||^g` $\overline{\text{اب ج}}$ *abǧ*

If necessary, use grouping by curly braces:

`\emphasize {'alayhi as-salAmu}` $\overline{\text{عليه السّلام}}$ *ʿalayhi ʾs-salāmu*

Appendix G

Coding examples for Persian[1]

The short vowels *æ (ă)*, *e (ĭ)*, *o (ŭ)* are denoted by the lowercase letters `a`, `e` or `i`, `o` or `u`:

`bar` بَرْ *bar*, `beh` بِهْ *beh*, `bon` بُنْ *bon*.

The long vowels *a (ā)*, *i (ī, ē)*, *u (ū, ō)* are denoted by the capital letters `A`, `I` or `E`, `U` or `O`. *Ælef mædde* is automatically generated for word-initial *a*:

`Ab` آبْ *āb*, `bAd` بَادْ *bād*, `bId` بِيدْ *bīd*, `bUd` بُودْ *būd*.

Note that `I` yields a *ya-ye mæ'ruf* (with *zir*), whilst `E` yields a *ya-ye mæjhul* (without *zir*). Similarly, `U` yields a *waw-e mæ'ruf* (with *piš*), whilst `O` yields a *waw-e mæjhul* (without *piš*):

`tIr` تِيرْ *tīr*, `tE.g` تيغْ *tēġ*; `dUr` دُورْ *dūr*, `zOr` زورْ *zōr*.

The diphthongs *e͡i* and *o͡u* are written `ay` and `aw`:

`pay` پَيْ *pay*, `naw` نَوْ *naw*.

Intervocalic *ḥæmze* is written `'`:

`pA'Iz` پَائِيزْ *pāʾīz*; `miyA'I` مِيَائِى *miyāʾī*, `mIgU'I` مِيگُوئِى *mīgūʾī*;

[1] **We gratefully acknowledge the voluntary help by Ivan Derzhanski who wrote this chapter, *and* implemented the language-specific processing. As we extensively modified his routines during system integration, all responsibility for any remaining, or new, errors rests with us.**

`tawAnA'I` تَوَانَائِی *tawānāʾī*, `zanA^sU'I` زَنَاشُوئِی *zanāšūʾī*.

Silent word-final *waw* is generated by `_U` or `O`:

`t_U` تُو *tu*, `d_U` دُو *du*; `tO` تو *tō*, `dO` دو *dō*.

Waw-e mæ'dul is written `w`; it is omitted in the transliteration and the preceding *xe* receives no *jæzm*:

`_hwAb` خوَابْ *ẖāb*, `_hwI^s` خوِیشْ *ẖīš*, `_hwod` خوُدْ *ẖod*.

Ha-ye hæwwæz-e mæxfi is generated by `H`, or optionally by `,e`, `,a` or `,A`. It does not receive a *jæzm* even in fully vocalised mode and is not joined to a following letter:

`_hAneH` خَانِه *ẖāneh*, `^c,e` چِه *čeh*, `naH` نَه *nah*,

`yal_aH` یَلَهْ *yalāh*, `yal,A` یَلَهْ *yalāh*

`_hAneHhA` خَانِهَا *ẖānehhā*, `_hAneH-hA` خَانِه هَا *ẖāneh-hā*.

Short eḍafe is written `-e` or `-i`:

`ketAb-e U` کِتَابِ اُو *ketāb-e ū*, `rAh-e t_U` رَاهِ تُو *rāh-e tu*,

`nAmeH-i man` نَامِهٔ مَنْ *nāmeh-i man*,

`bInI-e An mard` بِینِیِ آنْ مَرْدْ *bīnī-e ān mard*,

`pA-i In zan` پَایِ اِینْ زَنْ *pā-ʾ īn zan*,

`bAzU-i In zan` بَازُوِی اِینْ زَنْ *bāzū-ʾ īn zan*.

Long eḍafe is written `-_i`:

`dAr-_i man` دَارِ مَنْ *dār-ī man*, `_hU-_i t_U` خُوِی تُو *ẖū-ī tu*.

Ḥæmze* as *ya-ye wæḥdæt/nesbæt/xeṭab is likewise written `-_i`:

`nAmeH-_i` نَامِهٔ *nāmeh-ī*, `sormeH-_i` سُرْمِهٔ *sormeh-ī*,

`gofteH-_i` گُفْتِهٔ *gofteh-ī*.

Ye-ye wæḥdæt is written `-I` or `-E`:

`ketAb-I` کِتَابِی *ketāb-ī*, `rAh-I` رَاهِی *rāh-ī*, `nAmeH-I` نَامِه اِی *nāmeh-ī*;

`dAnA-I` دَانَائِی *dānā-ī*, `pArU-I` پَارُوئِی *pārū-ī*;

`dAnA-I-keH` دَانَائِیکِه *dānā-ī-keh*, `pArU-I-keH` پَارُوئِیکِه *pārū-ī-keh*.

The present tense forms of the verb *budæn* and the pronominal clitics are written as they are spoken:

`rafteH-am` رَفْتِه اَمْ *rafteh-am*, `rafteH-Im` رَفْتِه اِیمْ *rafteh-īm*,

`rafteH-I` رَفْتِه اِی *rafteh-ī*, `rafteH-Id` رَفْتِه اِیدْ *rafteh-īd*,

`rafteH-ast` رَفْتِه اَسْتْ *rafteh-ast*, `rafteH-and` رَفْتِه اَنْدْ *rafteh-and*;

`mard-Id` مَرْدِیدْ *mard-īd*, `asb-etAn` اَسْبِتَانْ *asb-etān*;

`An^gA-st` آنْجَاسْتْ *ānğā-st*, `U-st` اُوسْتْ *ū-st*, `t_U-st` تُوسْتْ *tu-st*;

`ketAb-I-st` کِتَابِیسْتْ *ketāb-ī-st*, `nAmeH-I-st` نَامِه اِیسْتْ *nāmeh-ī-st*.

The preposition *be-* can be written with or without a hyphen:

`be-man` بِمَنْ *be-man*, `be-t_U` بِتُو *be-tu*;

`be-An` بِآنْ *be-ān*, `be-In` بِاِینْ *be-īn*, `beU` بِاُو *beū*.

The components of compounds can be separated by `||` or `"|`:

`.sA.heb||_hAneH` صَاحِب خَانِه *ṣāḥebḫāneh*,

`ta_ht-e-"|_hwAb` تَخْتِ خوَابْ *taḫt-e-ḫāb*;

`pas||andAz` پَس اَنْدَازْ *pasandāz*, `naw||AmUz` نَو آمُوزْ *nawāmūz*,
`bI||_hwod` بِی خوُدْ *bīḫod*.

Appendix H

Alternate input encodings

H.1 ASMO 449 = ISO 9036

The file `asmo449.sty` contains a reading module for the ASMO 449 code (identical to ISO 9036). It is installed by the LaTeX option `asmo449` or by `\input asmo449.sty`. The module is activated by `\setcode {asmo449}` or `\setcode {iso9036}`; all following Arabic text will be considered to be coded according to the ASMO 449 standard. The ArabTeX notation may be reactivated by `\setcode {arabtex}`.

ASMO 449 (see Table H.1) is a 7-bit code, differing from ASCII (ISO 646) mainly by replacing the letters by the Arabic letter characters and diacritical marks; the Arabic digits share their positions with the ASCII digits. The positions of special and control characters in both codes are identical.

A minimal driver file for processing, e.g. a file `asmotext.dat`, could be structured as follows:

```
\documentstyle [arabtex,asmo449]{article}
\begin {document}
\setcode {asmo449}
\begin {arabtext}
\input asmotext.dat

% the preceding blank line is required if "asmotext.dat" did not
% end with a blank line itself; this is strange and embarrassing
\end {arabtext}
\end {document}
```

	0	1	2	3	4	5	6	7
00	NUL	DLE	SP	٠	@	ذ	ـــ	ـِ
01	SOH	DC1	!	١	ء	ر	ف	ـّ
02	STX	DC2	”	٢	آ	ز	ق	ـْ
03	ETX	DC3	#	٣	أ	س	ك	
04	EOT	DC4	$	٤	ؤ	ش	ل	
05	ENQ	NAK	%	٥	إ	ص	م	
06	ACK	SYN	&	٦	ئ	ض	ن	
07	BEL	ETB	’	٧	ا	ط	ه	
08	BS	CAN	)	٨	ب	ظ	و	
09	HT	EM	(	٩	ة	ع	ى	
10	LF	SUB	*	:	ت	غ	ي	
11	VT	ESC	+	؛	ث	]	ـً	}
12	FF	IS4	،	>	ج	\	ـٌ	|
13	CR	IS3	–	=	ح	[	ـٍ	{
14	SO	IS2	.	<	خ	^	ـَ	~
15	SI	IS1	/	؟	د	_	ـُ	DEL

Table H.1: ASMO 449 code table

As texts coded in ASMO 449 are always rendered verbatim the commands `\novocalize`, `\vocalize`, `\fullvocalize` and the language selection commands `\setarab` etc. make no sense and are temporarily disabled.

Texts in ASMO 449 are usually not fully vowelized. Thus the transliteration cannot be expected to be correct. This is especially true for Egyptian texts which commonly do not differentiate between *yā'* and *'alif maqṣūra*.

H.2 ASMO 449E = ISO 8859 - 6

The file `iso88596.sty` contains a reading module for the ISO 8859-6 code (extended ASMO 449 = ASMO 449E). It is installed by the LaTeX option `iso88596` or by `\input iso88596.sty`. The module is activated by `\setcode {iso8859-6}`; all following Arabic text will be considered to be coded according to the ISO 8859-6 standard. The ArabTeX notation may be reactivated by `\setcode {arabtex}`.

ISO 8859-6 (see Table H.2) is an 8-bit code closely related both to 7-bit ASCII and to ASMO 449; whereas the lower 128 positions are identical to ASCII (ISO 646), the upper 128 positions contain the Arabic characters of ASMO 449 in the analogous places, plus a few additional graphic and control characters.

We exploit the close relationship of these codes by reusing the ASMO 449 reading routines, after suitable modification of the input. This only works correctly if the input text does not contain genuine ASCII letters, as we project the Arabic characters onto their locations in ASMO 449. Some of the code switching messages in the log file are spurious; do not worry.

The notes on vowelization and transliteration of ASMO 449 apply also.

The driver file indicated for ASMO 449 will be usable after the obvious modifications; however, your TeX installation must be capable of processing 8-bit data input. This is nowadays usually the case; otherwise you can try to locally find some utility program that will strip the highest order bit off the characters in your file, and process the result via ASMO 449.

	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
00	NUL	DLE	SP	0 ٠	@	P	‘	p			NBSP			ذ	ـ	ـِ
01	SOH	DC1	!	1 ١	A	Q	a	q					ء	ر	ف	ـّ
02	STX	DC2	”	2 ٢	B	R	b	r					آ	ز	ق	ـْ
03	ETX	DC3	#	3 ٣	C	S	c	s					أ	س	ك	
04	EOT	DC4	$	4 ٤	D	T	d	t			¤		ؤ	ش	ل	
05	ENQ	NAK	%	5 ٥	E	U	e	u					إ	ص	م	
06	ACK	SYN	&	6 ٦	F	V	f	v					ئ	ض	ن	
07	BEL	ETB	’	7 ٧	G	W	g	w					ا	ط	ه	
08	BS	CAN	)	8 ٨	H	X	h	x					ب	ظ	و	
09	HT	EM	(	9 ٩	I	Y	i	y					ة	ع	ى	
10	LF	SUB	*	:	J	Z	j	z					ت	غ	ي	
11	VT	ESC	+	;	K	]	k	}				؛	ث		ً	
12	FF	IS4	,	>	L	\	l	|			،		ج		ٌ	
13	CR	IS3	–	=	M	[	m	{			SHY		ح		ٍ	
14	SO	IS2	.	<	N	^	n	~					خ		َ	
15	SI	IS1	/	?	O	_	o					؟	د		ُ	DEL

Table H.2: ISO 8859-6 code table

Appendix I

Miscellaneous utilities

The following packages are not part of ArabTEX proper, and are not supported in any way, but are distributed along with ArabTEX as possibly a convenience to the users. There is no warranty whatsoever.

I.1 twoblks.sty

This LaTeX option will define a command `\twoblocks {#1}{#2}` which will place the two parameters `#1` and `#2`, usually two paragraphs, into two boxes side by side, separated by space of length `\colsep`. If necessary, the resulting boxes will be split across a page boundary.

This feature is useful if two versions of a text are to be compared. They may be in different languages, and one of them might be in Arabic (if enclosed in `\begin {arabtext}` ··· `\end {arabtext}`).

This sentence has been written twice: in the English language and in the Arabic language.

كُتِبَتْ هٰذِهِ ٱلْجُمْلَةُ مَرَّتَيْنِ : بِٱللُّغَةِ ٱلإِنْجِلِيزِيَّةِ وَبِٱللُّغَةِ ٱلْعَرَبِيَّةِ .

Otherwise this command does not depend on ArabTEX in any way, and indeed originated in a completely different context.

Beware that the two "blocks" should each not contain much more than one, not too long, paragraph of text, otherwise TEX's main storage might overflow. There must be no `\verbatim` text inside the parameters of `\twoblocks`, nor any `\catcode` changes; and all TEX groups and `\if` ··· `\fi` sequences must be properly nested.

I.2 abjad.sty

This file, loaded as a LaTeX option, will define a command `\abjad {#1}` usable inside and outside of an *Arabic context.* It profited greatly from suggestions by Dr. Benno van Dalen (Utrecht University).

The command `\abjad {#1}` will convert its argument, which has to be a legal representation of a number between 1 and 1999, to the Arabic *'abğad* notation used in some mediaeval manuscripts. The result of the conversion will not look perfect, and the legal *'abğad* number 0 can presently not be generated.

Improving this routine needs a font revision, which is hard and tedious; whenever this happens, the command might well become part of ArabTeX proper.

I.3 MLS2ARAB

This is an UNIX SED script, written by Prof. Nicholas Heer (University of Washington), and released for free distribution. It will (almost) convert an ASCII file of Arabic text, produced by Multi-Lingual Scholar, to the ArabTeX input notation. The conversion is not perfect so some manual corrections might be necessary.

For operating instructions, see the file itself.

Index

" (quoting), 15
"|, 14, 15
$, 7
--, 15
\␣, 8
\\, 8
\abjad, 53
\arabfalse, 19
\arabtrue, 19
\begin{arabtext}, 6, 34
\begin{setcode}, 18
\bigskip, 8
\centerline, 9
\colsep, 52
\doassign, 9
\docommand, 9
\emphasize, 8
\end{arabtext}, 6, 34
\end{setcode}, 18
\footnote, 8
\fullvocalize, 13, 15, 16, 34
\hfill, 8
\hspace, 8
\indent, 8
\input, 8, 18
\input arabtex.tex, 5
\input atrans.sty, 20
\input etrans.sty, 20
\ligsfalse, 16
\ligstrue, 16
\magnification, 5
\marginpar, 8
\mbox, 8
\medskip, 8
\newhamza, 41
\newtanwin, 17, 40
\noindent, 8
\nospace, 8
\novocalize, 15–17
\oldarabtex, 35
\oldhamza, 41
\oldtanwin, 17, 40
\par, 6, 8
\pnash, 5
\pnashbf, 5
\quiet, 26
\setarab, 6, 10, 13, 21
\setcode, 18
\setcode{arabtex}, 18, 48, 50
\setcode{asmo449}, 48
\setcode{iso8859-6}, 50
\setcode{iso9036}, 48
\setfarsi, 10, 21
\setmaghribi, 10, 21
\setnash, 8, 11
\setnashbf, 8, 11
\setnastaliq, 8, 11
\setnone, 10, 21
\setpashto, 10, 21, 23
\seturdu, 10, 21, 23
\setverb, 10, 21, 24
\showfalse, 27
\showtrue, 27
\smallskip, 8
\space, 8
\spreadbox, 8
\spreadfalse, 25
\spreadline, 9
\spreadtrue, 25
\tracingarab, 26

`\transfalse`, 19
`\transtrue`, 19
`\twoblocks`, 52
`\vocalize`, 15, 16, 34
`\yahdots`, 25
`\yahnodots`, 25
`>`, 10, 21
`\|`, 8
`|`, 14–16
`|`B, 15
`|`BB, 15
`||`, 14, 15

‘ (*‘ayn*), 14
’ (*hamza*), 13

A, 12, 17, 38
’A, 14, 16, 41
,A, 46
_A, 12, 17, 38
˜A, 14
,a, 22, 23, 46
_a, 12, 16, 39
a (*fatḥa*), 12, 38
aa, 12, 38
abbreviation, 44
abjad.sty, 53
’abǧad numbers, 53
Afghanic, 22
‘ayn, 14
al-, 14, 19
’alif, 17
  dagger, 12, 16, 39
  initial, 17
  maqṣūra, 12, 17, 38, 40
    silent, 17, 40
  Qur’an, 16, 39
  silent, 17, 19, 38–40
  small, 16, 39
    below, 16, 39
’Allah (spelling), 43
aN, 13, 17, 39
aN_A, 17, 40
aNA, 13, 17, 39
aNY, 40
Arabic context, 6, 7
Arabic environment, 6
Arabic group, 7
Arabic item, 6
Arabic number, 7
Arabic quotation, 6
Arabic quotes, 7
Arabic word, 7
arabtex.tex, 5
ArabTEX commands, 7, 8
archaic text, 25
ASCII, 48, 50
ASMO 449, 18, 48, 50
aspirated consonant, 22
assignment, 9
assimilation, 14, 16, 19, 41
automatic stretching, 25
aW, 38
aw, 45
aWA, 38
ay, 45

B, 15
be-, 47
boxing commands, 8
breaking connections, 15

code
  7-bit, 48
  8-bit, 50
  arabtex, 18
  ASCII, 48, 50
  ASMO 449, 18, 48, 50
  ISO 646, 48, 50
  ISO 8859-6, 18, 50
  ISO 9036, 18, 48
coding conventions, 12, 34
commands
  ArabTEX, 7, 8
  boxing, 8
  illegal, 9
  internal, 5
  LATEX, 7

overview, 9
size changing, 5, 8, 11
TEX, 7
user defined, 5, 9
compounds, 47
copyright, 0, 32

dagger *'alif*, 12, 16
ḍamma, 12, 15, 16
inverted, 16, 22, 39
Dari, 21
date, 15
default font, 5, 11
defective writing, 12, 16, 39
definite article, 14, 19, 41
Derzhanski, Ivan, 45
diacritics, 16
diphthongs, 45
display mode, 7
dots on *yā'*, 22, 25

E, 21
`-E`, 22
,e, 22, 23, 46
`-e`, 22
EDMAC, 27
emphasis, 44
environment
Arabic, 6, 18
arabtext, 6, 18
setcode, 18
tabbing, 6

Farsi, 21
fatḥa, 12, 15, 16
Fischer, Wolfdietrich, 38
font
bold, 11
default, 5, 11
installation, 33
nash10, 34
nash14, 32–34
nash14bf, 33
naskh, 11, 32, 33
nasta'liq, 22, 32
selection, 11
unavailable, 11

grouping, 7, 44

H, 21–23, 46
h-, 15
hamza, 13, 15, 22, 40, 45, 46
carrier, 17, 40
old style, 41
ḥarakāt, 12, 15, 16, 38, 45
on *tatwīl*, 15
Heer, Nicholas, 53
hyphen, 15, 43

I, 12, 38
`-I`, 22
`~I`, 14
`-i`, 22
_i, 12, 16, 39
i (*kasra*), 12, 38
implementation
Mac, 32
PC, 32
U*IX, 32
iN, 13, 39
input switching, 18
insertion
mathematical, 7
non-Arabic, 7
Roman, 7
installation, 33
internal commands, 5
inverted *ḍamma*, 16, 22
invisible consonant, 14
ISO 646, 48, 50
ISO 8859-6, 50
ISO 9036, 48
iy, 12, 38
iẓāfet, 15, 22, 23, 46

kašīda, 15, 25, 43
kasra, 12, 15, 16

Kurdish, 21

la-, 43
language selection, 10
LaTeX commands, 7
li-, 43
ligature, 16, 34, 44
    breaking, 14–16, 44
lists, 9
long vowels, 12, 16

Macintosh, 32
madda, 14, 16, 45
Maghribi, 23
mathematical insertion, 7
METAFONT, 33
MLS2ARAB, 53
Multi-Lingual Scholar, 53

N, 15, 16, 19
naskh, 11, 32, 33
nasta'liq, 22, 32
nesting, 7, 9
NFSS, 33
NFSS2, 33
non-Arabic insertion, 7
NU, 13, 39
numbers, 43
    'abǧad, 53

O, 21, 46
option
    abjad, 53
    arabtex, 5
    asmo449, 18
    atrans, 20
    etrans, 20
    iso88596, 18
    nashbf, 11
    nastaliq, 11
    oldarabtex, 35
    twoblks, 52
Ottoman, 21

Pashto, 22, 23
PC implementation, 32
Persian, 21
Persian copula, 22
piš, 45
punctuation, 6

quotation
    Arabic, 6
    non-Arabic, 7
    Roman, 7
quoting, 13, 15, 16
Qur'an *'alif*, 16

reading module, 18
Roman insertion, 7

šadda, 14, 16, 41
    on tatweel, 15
short vowels, 12
silent *'alif*, 17, 19
size changing, 5, 8, 11
special codings, 25
stretching, 8, 15, 25
    automatic, 25
sukūn, 15, 16, 22, 34, 46
    on *lām*, 14
    on *tatwīl*, 15
sun letter, 14

T, 40
tabbing environment, 6
tā' marbuṭa, 40
tanwīn, 13, 15–17, 19, 39, 40
    fatḥa, 40
    on *tatwīl*, 15
tašdīd, 14, 16
tatwīl, 15, 43
TeX commands, 7
TeX hash size, 5, 33
text
    archaic, 25
    erroneous, 25
TeX-XeT, 33
transliteration, 12, 19, 34

Encyclopedia of Islam, 20
ZDMG, 19
twoblks.sty, 52

U, 12, 19, 38
_U, 39, 46
~U, 14
_u, 12, 16, 39
u (*ḍamma*), 12, 38
U*IX implementation, 32
UA, 17, 38
uN, 13, 39
unavailable font, 11
Urdu, 22, 23
user defined commands, 5, 9
uw, 12, 38

van Dalen, Benno, 53
verbatim, 17
vowel marks, 16
vowels
long, 12, 16, 38, 45
short, 12, 38, 45

W, 19
WA, 17
waṣla, 15, 16, 19, 34, 42

Y, 12, 38
yā'
dots, 22, 25
yā'-i-waḥdat, 22, 23, 46

zīr, 45
zwarakay, 22