

The HitType Package

Jonathan Walls

February 3, 2025

1 Introduction

This package aims to allow platform-agnostic typing in Hittite cuneiform using LaTeX. It is based on the work done by the Hethitologie Portal Mainz at www.hethport.uni-wuerzburg.de, using (with written permission) the Ullikummi fonts developed by Prof. Sylvie Vanséveren for Hittite cuneiform. This package was developed for use with LuaLaTeX, functionality with other compilers is not guaranteed.

2 Setup

The system aims to be maximally user-friendly (to the extent that LaTeX ever is), allowing more-or-less free typing using Latin characters as LaTeX commands which are then converted to cuneiform text in the PDF. To begin, in the project folders, one must include:

1. The Fonts folder containing the necessary fonts
2. The `hitttype.sty` file containing the commands necessary for the Unicode conversions.

In the preamble of the `main.tex` file, one must also include the following line to import the package:

```
\usepackage{hitttype}
```

This will incorporate the fontspec package allowing switching to the Unicode font, bring in the HitType functionality from the .sty file, and create the

```
\Hitt
```

command which will allow easy switching to the cuneiform typing mode.

3 Usage

3.1 Basic Cuneiform Typing

To use the package, in a given section, you can trigger the cuneiform writing mode by using the `\Hitt` command as listed above. Each character is designated by an uppercase command (to differentiate it from other stock LaTeX commands), and subscripts for sign variants are given in brackets. Typing is made fairly easy by simply turning on caps lock. An example of use is shown below:

```
\Hitt\UM\MA \AN\UD\CI \DIC\HAR\CI\HittB\LI\Hitt
\LUGAL\GAL \LUGAL \KUR \HittB\HA\Hitt\AD\TI
\UR\HittB\SAG

\setmainfont{Charis SIL}\normalsize
```

This displays the cuneiform text corresponding to:

UMMA ^dUTU-ŠI ^mMur-ši-li LUGAL.GAL LUGAL KUR *Ḫa-at-ti* UR.SAG


Note that you can use the commands

```
\HittB
\HittC
```

as shown above to switch to different scribal variants of certain signs or to create certain compound signs. To change between GÍR and GÌR, use the following syntax:

```

\GIR[2]
\GIR[3]
%the base GIR character can be had with
\GIR

```



All commands follow the listing as found in the HittiteSignList PDF included in the package Documentation folder. Note that `<C>` is used in place of `<Š>` for ease of typing (likewise `<H>` for `<Ḫ>`), and that Auto-text commands as shown in the document which correspond to numbers, i.e., 1, 2, etc., are replaced in the LaTeX code with

```

\ONE
\TWO

```

as LaTeX does not allow numeric characters in command names. Also note that any compound signs which require specifications such as `<ekisim5xamac>` in the PDF use roman numerals and a lowercase `<x>` in the LaTeX command, e.g.

```

\EKISIMvxAMAC

```

3.2 Formatting Hittite Transcriptions

This package also contains a number of commands which are useful for formatting transcriptions of Hittite, which follows conventions that are often tedious to work through. Some of these will create special characters often used in such transcriptions, as in the following table.

Command	Effect
<code>\badeq</code> (“bad equal”)	≠
<code>\funh</code> (“fun h”)	ḥ
<code>\funH</code>	Ḫ
<code>\hl</code> (“half left”)	┌
<code>\hr</code> (“half right”)	┐

Certain other commands are not explicitly included in this package but may be used without special inclusion, which are `\v{s}` which generates

š, \=V which will generate some vowel \bar{V} with a macron for plene spelling, and \d{S} which places a dot below a letter as in Š.

Some other commands are included which serve to facilitate formatting, as shown in the following table.

Command	Effect
<code>\sumer{}{}{}</code>	^D GĀŠAN=IA
<code>\tsup{}</code>	^M
<code>\tsub{}</code>	₂
<code>\bracks{}</code>	[<i>nu</i>]

`\sumer` takes three arguments (all optional). The first of these will be superscripted, the second will be in the normal font, and the final will be italicized, allowing for easy grouping and formatting of words which contain classifiers, sumero- or akkadograms, and Hittite phonetic complements without cluttering the reading.

`\tsup` and `\tsub` create super- and subscripts, respectively, and `\bracks` will put non-italicized square brackets around and italicize internal sections to make it easy to display broken sections of tablets where the text is inferred.

3.3 Working with expex

If one is using the expex package for interlinear translation, the easiest thing to do is to define a new glw level as shown in this example:

```
\defineglwlevels{cun}

\beginogl[glwordalign=center,glhangstyle=none, aboveglftskip=0pt,
everyglcun=\Hitt]
\gla \sumer{GI\v{S}}{}{lu-ut-ta-a-u\v{s}} kam-ma-ra-a-a\v{s} I\d{S}-
BAT \sumer{}{\'E}{-er} \hl t\'u\funh-\funh u-i\v{s}\hr\
\bracks{\'u-i-\v{s}}\textit{u-\hl ri-ya-ta\hr-ti}//
\glcun {\GIC\LU\UD\TA\A\UC} {\KAM\MA\RA\A\AC} {\GIC\BAD} {\E[2]\IR}
{\TAH\HU\IC} {\U[2]\I\CU\RI\IA\TA\TI}//
\glb \tsup{GI\v{S}}\lutt\=au\v{s} kammar\=a\v{s} I\d{S}BAT
\'E-er tu\funh\funh ui\v{s} wi\v{s}uribatati//
\glc windows.acc mist.nom seized.3sg house.acc smoke.nom stifled.3sg//
```

```
\glft Mist seized the windows, smoke stifled the house.\newline//
\endgl
```

Note the section at the beginning which defines a “cun” (cuneiform) glw level, which needs to be done only once at the beginning of the document, and note also the `everylcun=\Hitt` specification in the parameters which assigns the entire line to be in Hittite cuneiform. You may of course omit one of the transcription levels, in which case you can simply add the specification to the a, b, or c level instead. This code will produce the following output:

^{GIŠ} <i>lu-ut-ta-a-uš</i>	<i>kam-ma-ra-a-aš</i>	<i>IŠ-BAT</i>	<i>É-er</i>
^{GIŠ} <i>luttāuš</i>	<i>kammarāš</i>	<i>IŠBAT</i>	<i>É-er</i>
windows.acc	mist.nom	seized.3sg	house.acc
^ʽ <i>túh-ḫu-iš</i> ^ʽ	^ʽ <i>[ú-i-š]u-ri-ya-ta</i> ^ʽ <i>ti</i>		
tuḫhuiš	wišuriyatati		
smoke.nom	stifled.3sg		

Mist seized the windows, smoke stifled the house.

Happy Hittite!