

# Athenian Numerals II\*

Apostolos Syropoulos  
366, 28th October Str.  
GR-671 00 Xanthi, HELLAS  
Email:apostolo@platon.ee.duth.gr

2003/08/24

## 1 Introduction

`\athnum` This L<sup>A</sup>T<sub>E</sub>X package implements the macro `\athnum`. The macro transforms an Arabic numeral, i.e., the kind of numerals we all use (e.g., 1, 5, 789 etc), to the corresponding *Athenian* numeral. Athenian numerals were in use only in ancient Athens. The package can be used only in conjunction with the `greek` option of the `babel` package.

## 2 The Numbering System

The athenian numbering system, like the roman one, employs letters to denote important numbers. Multiple occurrence of a letter denote a multiple of the “important” number, e.g., the letter I denotes 1, so III denotes 3. Here are the basic digits used in the Athenian numbering system:

- I denotes the number one (1)
- II denotes the number five (5)
- Δ denotes the number ten (10)
- H denotes the number one hundred (100)
- X denotes the number one thousand (1000)
- M denotes the number ten thousands (10000)

Moreover, the letters Δ, H, X, and M under the letter II, denote five times their original value, e.g., the symbol  $\text{II}\Delta$ , denotes the number 5000, and the symbol  $\text{II}\Delta$ , denotes the number 50. It must be noted that the numbering system does not provide negative numerals or a symbol for zero.

The Athenian numbering system is described, among others, in an article in Encyclopedia *Δομή*, Vol. 2, page 280, 7th edition, Athens, October 2, 1975.

## 3 The Code

Before we do anything further, we have to identify the package.

```
1 \*package\
2 \NeedsTeXFormat{LaTeX2e}[1996/01/01]
3 \ProvidesPackage{athnum}[2003/08/24\space v1.1]
4 \typeout{Package: 'athnum' v1.1\space <2003/08/24> (A. Syropoulos)}
```

---

\*The documentation of this package is essentially the same as that of the package ‘grnumalt’. The ‘II’ serves as a means to distinguish the two documents.

Next we check to see if the `babel` package is loaded with at least the `greek` option. In case it isn't, we opt to produce an error message.

```

5 \ifpackagewith{babel}{greek}{\relax}
6   \ifpackagewith{babel}{polutonikogreek}{\relax}
7     \PackageError{athnum}{%
8       'greek' option of the 'babel' package
9       package hasn't been loaded}{%
10        The commands provided by this package
11        are specially designed for greek language
12        typesetting with the 'babel' package. Load
13        it with at least the 'greek' option.}\relax
14    }

```

As it is mentioned in the introduction, the Athenian numerals employ some special digits. These digits are included in the `cb` fonts of Claudio Beccari, and so we must provide access commands.

```

15 \DeclareTextCommand{\PiDelta}{LGR}{\char"02\relax}
16 \DeclareTextCommand{\PiEta}{LGR}{\char"03\relax}
17 \DeclareTextCommand{\PiChi}{LGR}{\char"04\relax}
18 \DeclareTextCommand{\PiMu}{LGR}{\char"05\relax}

```

`\@@athnum` Now, we turn our attention to the definition of the macro `\@@athnum`. This macro uses one integer variable (or counter in T<sub>E</sub>X's jargon.)

```

19 \newcount\@ath@num

```

The macro `\@@athnum` is also defined as a robust command.

```

20 \DeclareRobustCommand*\@@athnum[1]{%

```

After assigning to variable `\@ath@num` the value of the macro's argument, we make sure that the argument is in the expected range, i.e., it is greater than zero, and less or equal to 249999. In case it isn't, we simply produce a `\space`, warn the user about it and quit. Although, the `\athnum` macro is capable to produce an Athenian numeral for even greater intergers, the following argument by Claudio Beccari convised me to place this above upper limit:

According to psychological perception studies (that ancient Athenians and Romans perfectly knew without needing to study Freud and Jung) living beings (which includes at least all vertebrates, not only humans) can perceive up to four randomly set objects of the same kind without the need of counting, the latter activity being a specific acquired ability of human kind; the biquinary numbering notation used by the Athenians and the Romans exploits this natural characteristic of human beings.

```

21   \@ath@num#1\relax
22   \ifnum\@ath@num<\@one%
23     \space%
24     \PackageWarning{athnum}{%
25       Illegal value (\the\@ath@num) for athenian numeral}%
26   \else\ifnum\@ath@num>249999%
27     \space%
28     \PackageWarning{athnum}{%
29       Illegal value (\the\@ath@num) for athenian numeral}%
30   \else

```

Having done all the necessary checks, we are now ready to do the actual computation. If the number is greater than 49999, then it certainly has at least one `\Pi` "digit". We find all such digits by continuously subtracting 50000 from `\@ath@num`, until `\@ath@num` becomes less than 50000.

```

31   \@whilenum\@ath@num>49999\do{%
32     \PiMu\advance\@ath@num-50000}%

```

We now check for tens of thousands.

```
33      \@whilenum\@ath@num>9999\do{%
34      M\advance\@ath@num-\@M}%
```

Since a number can have only one  $\text{X}$  “digit” (equivalent to 5000), it is easy to check it out and produce the corresponding numeral in case it does have one.

```
35      \ifnum\@ath@num>4999%
36      \PiChi\advance\@ath@num-5000%
37      \fi\relax
```

Next, we check for thousands, the same way we checked for tens of thousands.

```
38      \@whilenum\@ath@num>999\do{%
39      Q\advance\@ath@num-\@m}%
```

Like the five thousands, a numeral can have at most one  $\text{H}$  “digit” (equivalent to 500).

```
40      \ifnum\@ath@num>499%
41      \PiEta\advance\@ath@num-500%
42      \fi\relax
```

It is time to check hundreds, which follow the same pattern as thousands

```
43      \@whilenum\@ath@num>99\do{%
44      H\advance\@ath@num-100}%
```

A numeral can have only one  $\text{I}$  “digit” (equivalent to 50).

```
45      \ifnum\@ath@num>49%
46      \PiDelta\advance\@ath@num-50%
47      \fi\relax
```

Let’s check now decades.

```
48      \@whilenum\@ath@num>9\do{%
49      D\advance\@ath@num by-10}%
```

We check for five and, finally, for the digits 1, 2, 3, and 4.

```
50      \@whilenum\@ath@num>4\do{%
51      P\advance\@ath@num-5}%
52      \ifcase\@ath@num\or I\or II\or III\or IIII\fi%
53      \fi\fi}
```

**\@athnum** The command `\@athnum` has one argument, which is a counter. It calls the command `\@@athnum` to process the value of the counter.

```
54 \def\@athnum#1{%
55     \expandafter\@@athnum\expandafter{\the#1}}
```

**\athnum** The command `\athnum` is a wrapper that declares a new counter in a local scope, assigns to it the argument of the command and calls the macro `\@athnum`. This way the command can process correctly either a number or a counter.

```
56 \def\athnum#1{%
57     \@ath@num#1\relax
58     \@athnum{\@ath@num}}
59 \</package>
```

## Acknowledgment

I would like to thank Claudio Beccari for reading the documentation and for his very helpful suggestions. In addition, Antonis Tsolomitis spotted a bug in the first version, which is corrected in the present version.

## Dedication

I would like to dedicate this piece of work to my son

Demetrios-Georgios.