

“GRKT_EXedt” - Tamil T_EX editor for scientific typesetting and for L^AT_EX out put

K. Ganeshamoorthy^a, J. C. N. Rajendra^a and S. Kanaganathan^b

^a Faculty of Science, Eastern University, Chenkalady, Batticaloa, Sri Lanka

^b Faculty of Applied Sciences, University of Jaffna, Vavunia Campus, Sri Lanka

kganeshamoorthy@yahoo.com, jcnrajendra@bellmail.lk, sknathan55@yahoo.com

Abstract

T_EX is an ASCII text-based markup language. In a scheme of automated scientific document preparation L^AT_EX provides the foundation, which is also a markup language created from T_EX. In this work a user-friendly editor was developed integrating the Tamil text-based ASCII characters with the Tamil L^AT_EX characters via character mapping process. These conversions of Tamil text-based fonts have been interfaced with the L^AT_EX engine to produce L^AT_EX out put such as DVI, PDF and PS.

1. Introduction

T_EX is a well-established ASCII text-based markup language introduced by Donald Knuth in late 1970's[1]. It is a computer language designed for the use of typesetting. L^AT_EX is perhaps the mother of all typographic systems in scientific typesetting introduced by American Mathematical Society (AMS) following the invention of T_EX. It makes the authors to concentrate on the content rather than spending more time and effort in managing the complicated and the details of typesetting formats. Although the L^AT_EX engine does not use the WYSWYG input, but it gives the great end result for technical presentation, scientific reports, thesis, publication, etc. In present scientific text writing L^AT_EX is widely used in several languages including Asian languages.

Harold Schiffman and Vasu Renganathan of University of Pennsylvania were the first researchers to design the T_EX Tamil fonts [2]. Thuraiappah Vaseeharan has attempted a first step in Tamil- L^AT_EX [3]. He and his co-workers namely Chinnasamy Nagarajan introduced the AkaramT_EX Tamil Text Editor for T_EX output. It is an extension to the popular Akaram Tamil editor. Chinnasamy Nagarajan developed Akaram. As it is written in TCL/TK, it is platform independent and can be used on Unix, Windows or Mac. Currently TSCII fonts are used displaying Tamil characters. The Tamil characters will be automatically converted to the appropriate L^AT_EX characters. Currently, AkaramT_EX supports the WNTML (Washington University Tamil) fonts [4] for T_EX output.

Although AkaramT_EX is a great package to be used for T_EX output, it has its own drawbacks and difficulties in usage. In this project we have developed an editor, which has the user-friendly environment and at present runs only on windows platform. It has the facility to convert the L^AT_EX characters automatically. Tamil ASCII fonts are used to display Tamil

characters in editor screen. Character mapping concept has been used to integrate to switch between the Tamil ASCII fonts and L^AT_EX characters. Also this can be used as a bilingual editor in L^AT_EX environment. Currently it supports only the available WNTML (Washington University Tamil) fonts [4] for T_EX output. The GRK_TE_Xedt has the special feature to support the window environment to interface and run the appropriate L^AT_EX file without switching to another console. Also within the window environment the conversion of pdfL^AT_EX, dvi2ps, dvi2pdf and DVI out put are all interfaced with L^AT_EX engine. In this project MiK_TE_X [5], which is an up-to-date implementation of T_EX and related programs for Windows, has been used to integrate with the editor.

1. Integration with Tamil LaTeX characters

Several authors, using different text editing modes, have researched Tamil typesetting using L^AT_EX. In this project it has been viewed in a different perspectives to create an independent text editor to handle bilingual features and to have a user-friendly environment.

In achieving this mode of text editing and scientific typesetting in bilingual format, two major resources have been employed. Firstly, the well-established L^AT_EX engine packed in Mik_TE_X [5] was used under Windows platform. The known L^AT_EX Tamil characters called ‘WNTML’ created by Humanities and Arts Computing Centre of the University of Washington, USA were used to create the L^AT_EX output mainly the De_Vice-Independent format (DVI). These can be easily converted to the other formats such as Portable Document Format (PDF) and Post Script format (PS). Secondly the character-mapping concept has been used to map the Tamil ASCII fonts with the Tamil L^AT_EX subroutine to create Tamil characters. This will allow the text to be converted to the format needed by the L^AT_EX engine to produce the L^AT_EX output.

An algorithm has been designed to do the character mapping. The mapping procedure takes ASCII characters one-by-one from the source ASCII document, which is the document in bilingual form written in the GRK_TE_Xedt by the user. The procedure checks whether the font-mode of the ASCII character is in English or in Tamil. If the font-mode is in English the procedure adds the ASCII character to the L^AT_EX source document, which is the T_EX document needed by the L^AT_EX engine to produce the L^AT_EX output. Otherwise, if the font-mode is in Tamil, it will use the ASCII value of the ASCII character as an index to obtain the name of the Tamil L^AT_EX subroutine from a linear array. This linear array contains the names of all Tamil L^AT_EX subroutine in a class file. The corresponding subroutine will be added to L^AT_EX source document. These subroutines display/print Tamil letters when the L^AT_EX engine executes the L^AT_EX source document.

As an initial stage the Tamil ASCII characters for Bamini fonts were used to map with the Tamil L^AT_EX characters. The Bamini fonts were selected purely based on the fact that the most users in Sri Lanka are familiar in using this font style for their Tamil word processing using standard word processing packages such as MSWord.

The common English and Tamil ASCII fonts (in this case Bamini fonts for Tamil) and the usual L^AT_EX commands could be used to create the L^AT_EX source documents as shown in **Figure 2** in the “GRKTeXedt” editor environment.

```
\documentclass{article}
\usepackage{tamil}
\begin{document}
  \begin{enumerate}
    \item \T(மீள்வருவனவற்றின் எல்லைப் பெறுதலானம் காண்க.) \\\
          $\displaystyle{\lim_{x \rightarrow 3} \frac{x^2-3x}{x^2-9}}$

    \item \T(மீள்வருவனவற்றை x குறித்து தொகையிடுக.) \\\
          $\displaystyle{\int_{-\infty}^{\infty} \frac{1}{\pi[1+(x-a)^2]}}$

    \item \T(மீள்வருவனவற்றை x \T(குறித்து வகையிடுக.) \\\
          $\displaystyle{y=2x+\frac{4}{x}}$' \T(எனின்)
          $\displaystyle{x^2 \frac{d^2y}{dx^2}+x \frac{dy}{dx}-y=0}$ \T(என நிறுவுக.)

  \end{enumerate}
\end{document}
```

Figure 2 Typical L^AT_EX bilingual source document

However, if you need to include any Tamil characters the command \T () should be used as shown in Figure 2 and the Tamil characters should be written within the round bracket. Also within the source file the Tamil style file “tamil.sty” supplied by Ross Moore and ITRANS [6] should be called by using the command \usepackage{Tamil} in the initial preamble of the source document.

As mentioned earlier the known Tamil L^AT_EX character WINTML is used to create the L^AT_EX out put. These Tamil characters are only available in three sizes namely 10, 12 and 17 points corresponds to the sizes of the output of \normal, \Large and \huge commands in L^AT_EX.

The characters used in WINTML do not have any numerals or any punctuation characters. Therefore, if you need to use numbers or punctuation marks within Tamil scripts you should change the font before any numbers or punctuation marks. For numbers, it is usually sufficient to use the \$ enclosing scope to make them print correctly, since \$ enters the mathmode and use the math fonts. For punctuation mark, the user needs to change fonts.

When the process is called to mapped the Tamil ASCII fonts with the relevant Tamil L^AT_EX characters it will follow the routine flowchart given in **Figure 3**.

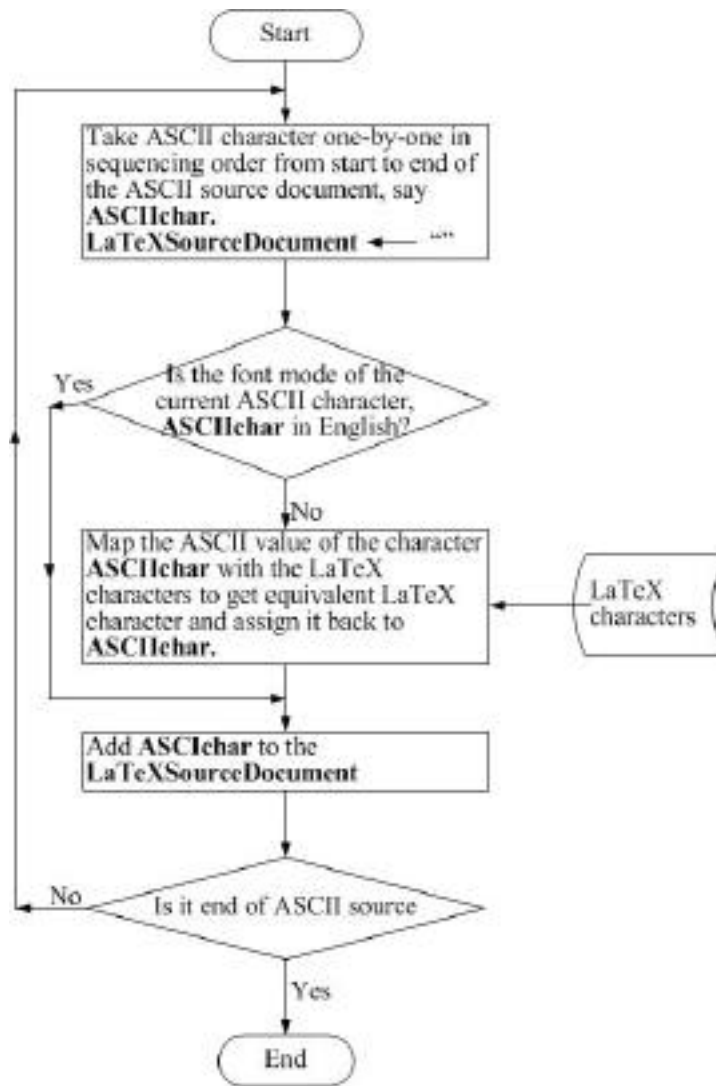


Figure 3 The flowchart to process the character mapping

When the bilingual L^AT_EX source document is prompt to save, this flowchart process will start and create TEX and RTF format files and save under the given file name. Also these processes will be done in a hidden environment leaving the user without any complication. As far as the user is concern, he/she will be in the usual typesetting or word processing environment. The created TEX extension file could be used later to get the L^AT_EX out put either in DVI, PDF or PS formats. The RTF extension file could be used for further modification or typesetting.

2. Description of GRKT_EXedt

The front-end integration of the “GRKT_EXedt” was developed using the Visual Basic programming language. **Figure 4** shows the screen shot of the “GRKT_EXedt.

All
the

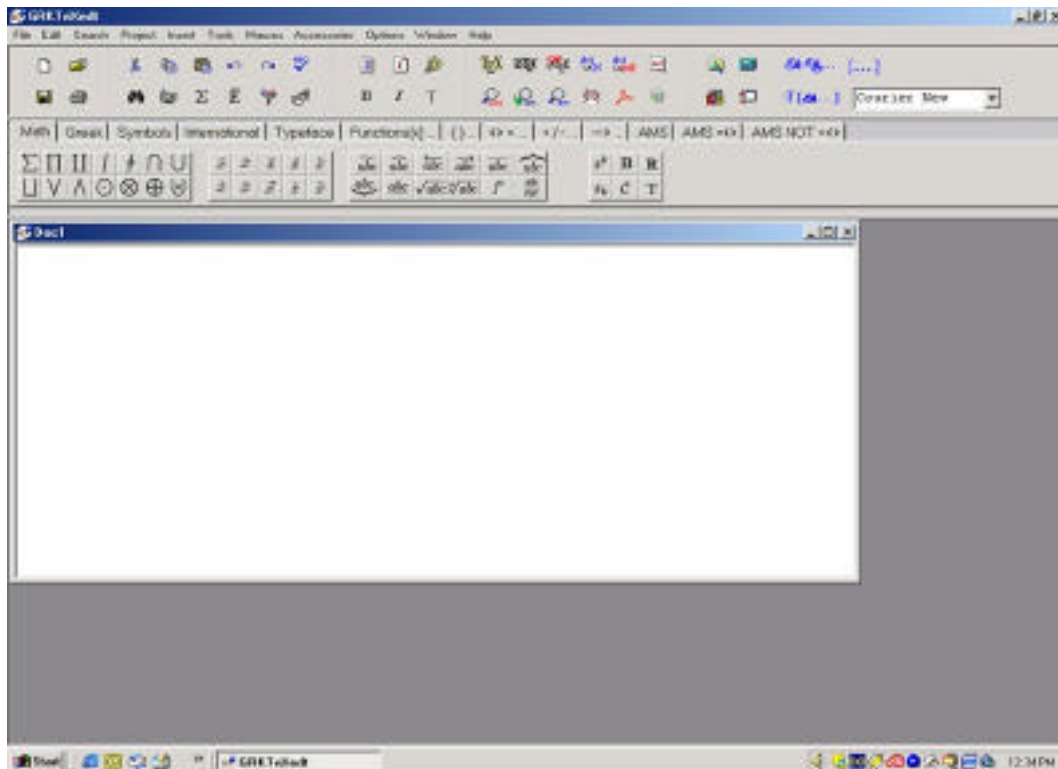


Figure 4 Typical “GRKTeX” editor environment for bilingual typesetting

possible buttons for the use of $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ commands are created and integrated with the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ engine. Bilingual mode between the Tamil and English were also integrated within the editor. Clicking the appropriate button or by pressing the F11 key in the keyboard will swap these two modes. When the bilingual source cord for $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ file creation is written as shown in Figure 2, it has to be saved under a given file name creating the TEX and the RTF files. Then the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ executive file will be called to execute by clicking the appropriate button in the editor, the executive file will take the current version of the TEX file and execute the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ commands under DOS mode and create the DVI file to view the document.

Figure 5 shows the flowchart for the process of executing the bilingual $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ source code shown in Figure 2 under $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ environment.

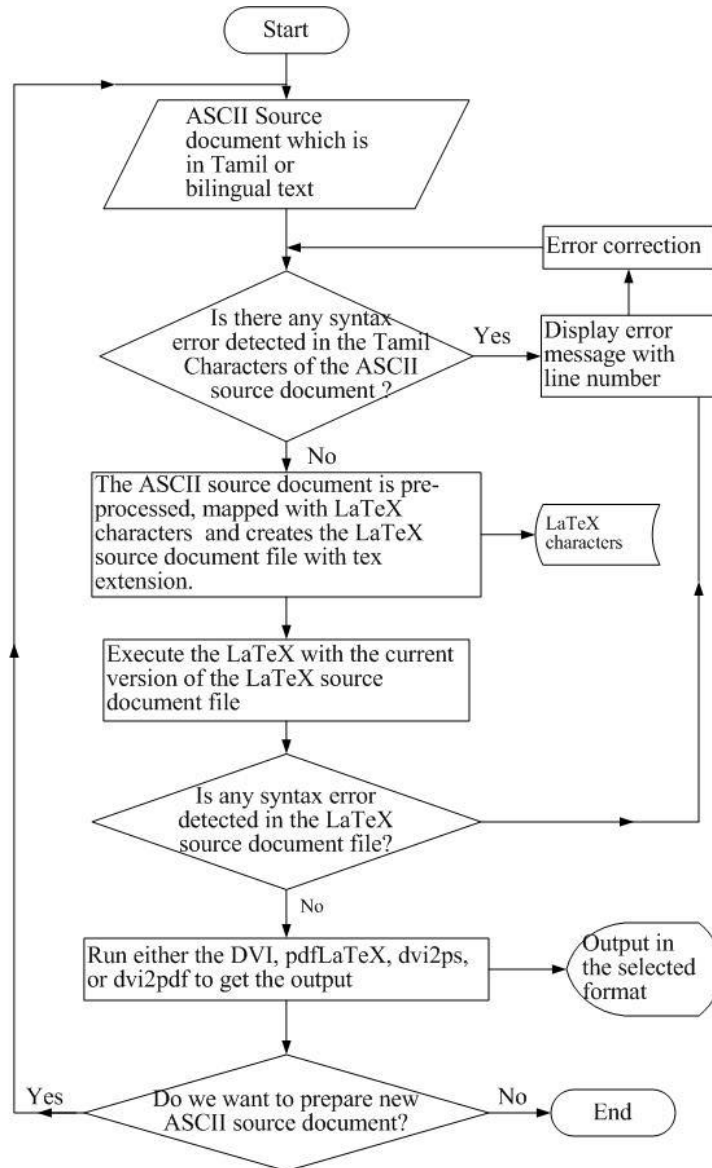


Figure 5 The flowchart for the process of executing the typical bilingual $L^A T_E X$ source code shown in Figure 2 under $L^A T_E X$ environment

The typical DVI out put for the said bilingual $L^A T_E X$ source code shown in Figure 2 is given in **Figure 6**.

1. பின்வருவனவற்றின் எல்லைப் பெறுமானம் காண்க

$$\lim_{x \rightarrow 3} \frac{x^2 - 3x}{x^2 - 9}$$
2. பின்வருவனவற்றை x குறித்து தொகையிடுக

$$\int_{-\infty}^{\infty} \frac{1}{\pi[1 + (x - a)^2]}$$
3. பின்வருவனவற்றை x குறித்து வகையிடுக

$$y = 2x + \frac{4}{x} \text{ எனின் } x^2 \frac{d^2y}{dx^2} + x \frac{dy}{dx} - y = 0 \text{ என நிறுவுக}$$

Figure 6 DVI out put for the bilingual L^AT_EX source code shown in Figure 2

These processes are all done automatically and make the user friendlier with the L^AT_EX environment. The user need not to be worry about any conversion between the normal Tamil ASCII font and the Tamil L^AT_EX characters. All the relevant mapping process, conversion and the relevant DVI, PDF, PS, etc. out put will be done with the one click of the mouse to the relevant button in the editor.

3. Conclusion

GRKT_EXedt has more user-friendly environment and has smooth bilingual and scientific editing facilities. Since the one-to-one mapping process is used in transforming the Tamil ASCII characters to L^AT_EX Tamil subroutine the conversion process and the speed of process level is considerably high. This has an added advantage in this editor to keep the user more flexible and user friendly. Although currently it supports only available fonts for L^AT_EX output, steps have been taken to create more fonts size and styles in Tamil. Also work has been planned to integrate the Unicode, TSCII and TAB fonts in to the editor for universal usage.

References

- [1] <http://www.tug.org/>
- [2] <http://ccat.sas.upenn.edu/~haroldfs/>
- [3] <http://tamil.htmlplanet.com/akaramTeX/aktex.html>
- [4] Anshuman Pandey (1998) An Over view of Indic Fonts for T_EX, *TUGboat*, Volume19(1998), No.2 115-120
- [5] <http://www.miktex.org/reg.html>
- [6] <http://www-texdev.mpce.mq.edu.au/l2h/indic/Tamil/tamildocs/tamildocs.html>,
<http://www.aczoom.com/itrans/>