DOING MORE WITH LESS: MAGIC ON WINDOWS 95/NT

Danny F. Newport
Electrical Engineering
University of Tennessee
Knoxville, TN 37996-2100
newport@microsys6.engr.utk.edu

ABSTRACT
Since its creation in the mid 1980’s, Magic has been one of the most widely used layout tools for Manhattan (Mead and Conway) style IC layout. This is due to its flexibility, ease of use, and cost (free via ftp). Magic has been ported to many platforms, including the MAC-II and DOS, but mainly Unix-based platforms. The DOS version was limited and not widely used. With the proliferation of Windows 95/NT platforms, it seems only natural that Magic be ported to Windows 95/NT. Yet, this does not appear to be a port to Windows 95/NT available. The reason for this appears to be the difficulty of programming “windows” in this environment. We believe that we have a way around this difficulty. Development is currently underway to port Magic to Windows 95/NT using Tcl and the Tk Toolkit. This should ease the interface to the window system through “generic” routines. As a matter of fact, the same Tcl/Tk display driver should function with X11 Unix-based platforms as well as Windows 95/NT and other platforms with Tcl and Tk toolkits.

1. INTRODUCTION
Although automated synthesis of integrated circuits from high level languages, such as VHDL or Verilog, is fast becoming the method of choice for designing microelectronic systems, there is still the need for low level physical layout. The need for physical layout is evident in analog and mixed-signal IC design, as well as cell design. Many microelectronic system courses involve some form of physical layout. Magic [2] is one of the most widely used layout tools for Manhattan (Mead and Conway) style IC layout in these types of courses.

This paper summarizes the motivation and method of porting Magic to Windows 95/NT. A brief description of Tcl and the Tk Toolkit, which is to be used to do the port, is presented. Finally, further work with the use of Magic and other tools based on Windows 95/NT will be presented.

2. MOTIVATION
Magic has been predominantly available on Unix-based platforms. This has restricted the use of Magic by students to labs with somewhat expensive workstations. Increasing numbers of students have their own computers, typically x86-based platforms. As a matter of fact, several universities are requiring engineering students, if not all students, to have their own computer, again typically x86-based platforms. Even at universities which do not require students to have their own computers, surveys of students indicate that many have personal computers. For instance, at the University of Tennessee, Knoxville, students are not required to have a personal computer, but a recent survey indicated that over 1/3 did have a personal computer.

Even given that all students do not have a personal computer, universities in the current cost-conscious times are purchasing more x86-based platforms for labs than Unix-based platforms. Even major electronic design automation companies which have been Unix-based are moving to support x86-based platforms running Windows 95/NT. With this proliferation of inexpensive x86-based platforms, it seems only natural that Magic and other tools be ported to this platform.

Magic has been ported to Linux of which there is a version available for x86-based platforms. However, Linux installation and system administration can be difficult. An older version of Magic (4.x) was ported to DOS, but the tool was limited and restricted its usefulness in a larger design. Most x86-based platforms use Windows 95 or NT. Yet, there does not appear to be a port of Magic to Windows 95/NT available.

With very good C/C++ compilers available for Windows 95/NT (Borland C++ and Microsoft's Visual C++ for instance) and virtual memory support, one would expect Magic would easily compile since it is very “generic” C code. Compiling Magic, however, is not the difficulty. It appears that the reason that Magic has not been ported to Windows 95/NT is the difficulty of programming “windows” in this environment. There appears to be a means available to reduce this difficulty - Tcl and the Tk Toolkit [1].

3. TCL AND THE TK TOOLKIT
Tcl and the Tk Toolkit provide a programming environment for creating graphical user interfaces (GUIs). Initially created to work under X windows, there are ports of Tcl and the Tk Toolkit available for most platforms, including Windows 95/NT. This eases the interface to any window system through the use of “generic” routines. Tcl and the Tk Toolkit are very easy to learn and use. There are Tk calls which can be substituted for X window calls. Thus, it should be straightforward to convert an existing X window program to Tk.

Magic has an X window display driver. Thus, it should be straightforward to create a Tk display driver for Magic using the existing X window display driver. It may also be possible to eliminate the use of the “X11Helper” program by Magic. This program handles asynchronous input to Magic from the X window server. Mouse button clicks and terminal input are these types of input. The Tk Toolkit handles these inputs inherently. Thus, it may be possible
to not only produce a port of Magic to Windows 95/NT, but to also improve the portability of Magic.

4. FURTHER WORK

Using Tcl and the Tk Toolkit, one should be able to not only port Magic to Windows 95/NT but also other public domain tools such as ext2spice and IRSIM. There are inexpensive or free versions of SPICE available for Windows 95/NT. The author with fellow collaborators are in the process of producing a guide to using Magic for analog and digital design. The guide would include tips and techniques for “good” layout design practices. A copy of Magic and other public domain tools ported to Windows 95/NT would be provided with it. Thus, a “complete” IC physical design kit could be produced for Windows 95/NT that could be used in a microelectronics system course.

REFERENCES
