LinuxFocus article number 322
http://linuxfocus.org

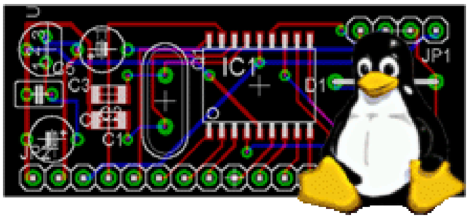
by Aleksandar Milovac
<aleksandar.milovac(at)gmail.com>

About the author:

Aleksandar has recently graduated from the Faculty of Technical Sciences in Novi Sad, Department of electronics and telecommunication. He has been using Linux since 1999. For the last two years he has been using Debian GNU/Linux.

Translated to English by:
Aleksandar Milovac
<amilovac(at)ptt.yu>

Let's do some electronics



Abstract:

When I was an electrical engineering student (that was not so long ago), I've often had a chance to work with commercial software such as: Protel, OrCAD, various comercial compilers... When I discovered Linux (early '99), I asked myself: "Would it be possible to do all this on Linux?" After four years of using it I'm proud to say:" Yes! It is." In this article I will introduce you to the software you need for Linux.

Introduction

Linux started as a hobby project in 1991 but things have changed since then. Today it's a modern multiuser, multitasking OS, which has the raputation of being very robust a reliable. These facts are crucial wheather you are an electronics engineer or just a hobbyist.

But one of the most important things about Linux is that it is an open source project. What has that to do with electronics? If you are an engineer working on some project you could try doing it on Linux. Since GPL lets you acquire other people's ideas and share yours with them, you could use parts of code that are already written and save lots of time. Then you could concentrate on designing your hardware – for example. Also, Linux and many other programs in open source community are very well documented and have an excellent technical support service. You just send email to the author or join some of the news groups – they will be glad to help you. These are only few of the benefits of using Linux for your job.

Some of my friends were not convinced by that and they said: "But why would we leave something that we're familiar with and switch to this great unknown?" They might be right, but you will never find out why if you don't try it. So, let us try this and see what we'll come up with?

What to choose?

Everyone who tries to make some electronic device first starts with the schematics, and then makes a printed circuit board (PCB) which is later used to assemble all the components of some specific device. Thanks to the modern computers we could do all of this in our room with some beer and crackers next to us. We all know the programs that are used for this: Protel, Pcad, OrCAD... But they are to expensive for us home users.

Fortunately there is a good program which let's you do all this but without paying any money. The name of this program is EAGLE which stands for Easily Applicable Graphical Layout Editor. This program is being developed by Cadsoft (www.cadsoftusa.com) but unfortunately it is just freeware and not GPL software. Thanks to the people from Cadsoft you can use it freely on your Linux box, but with few constraints. The PCB is 100mm by 80mm and it can be only done in two layers. Even with this you can make good PCB's. After few months of using EAGLE, I must say it is really an excellent software. It's small and not so memory consuming like some other programs. You will need some time to adopt its logic, but that is not so hard. The interface is clean and simple. Special thanks to the library manager which is simple and very intuitive.

Besides Eagle there are programs that are used on universities, and in the industry. One of them is the famous Matlab (mathworks.com). It's practicly a standard in numerical computation, dsp, and system modelling. Although there is a version for Linux − it's too expensive. If you don't have a lot of money you can use a program called Scilab (scilabsoft.inria.fr). It will not cost you anything and you can download source code or binaries. Practically it can do anything that Matlab does: numerical computation, dsp... It has a command syntax that is similar to Matlab's. It is written for X and you will need Tcl library in order to install it. It even has a toolbox that is analogue to Matlab's Simulink and it's called Scicos (www.scicos.org)

What about programming micocontrollers and other stuff? Can this be done on Linux. Of course! Linux is known for excellent support for many programming languages. You can write programs for your beloved microcontroller and compile them. You can even program it under Linux. No need for expensive compilers and other stuff. If you must program a lot of different microcontrollers, EEPROM's..., I would recommend a programmer called PonyProg (www.lancos.com/prog.html). With it, and suitable hardware, you can program PIC's, AVR's, lots of different EEPROM's etc. It runs on X, and it's simple and efficient.

If you are just AVR fan then you can use a programmer called SP12. It's a command line programmer and it comes with schematics for an ppropriate hardware programmer, which is very simple and easy to use. I've used it for a project at university and it worked just fine. Of course this is not the only one serial programmer for AVR. There are many good AVR programmers out there. You can use C or assembler to write programs for it. If you choose C then you should consider using AVR−GCC compiler. Guido Socher wrote (LF article 231) about it a few years ago, so I will just mention it. Read his article if you want to know more about it. It's very good.

For all of you who are PIC fans here is something good. I use Debian GNU/Linux on my box. When I browsed the aptitude to see is there something about PIC's I found a couple of programs. The first is called Picasm and it is an assembler for PIC microcontrollers. The second is simulpic and obviously it is a PIC simulator. I didn't test these programs yet, since I didn't use PIC's in the past, but you can try it . As for the programmers for PIC's, I would recommend Picprg (www.brianlane.com) by Brian Lane. It's a simple and easy to use program.

Since we're talking about electronics, programming and similar stuff on Linux, I should mention one more interesting thing. Recently Linux has been introduced into areas such as embedded systems design and real time OS's. If you are interested to use Linux as a real time OS which will be used to monitor some industrial processes or to control machines you should check out two most popular real time extensions for Linux:

RTLinux (www.fsmlabs.com) and RTAI (www.rtai.org). Both are available under GPL and very well documented.

Conclusion

This has been a short presentation of free/open source software for everyday (and advanced) electronics. If you thinking about doing some electronics you should try Linux. As you can see, there are a lot of programs which you can use.

In the past four years I've searched for free/open source software available for Linux that will replace all programs that I used on university or at home on my Windows box. I managed to replace everything that I used on Windows. I still keep on searching for better software. But things are much more easier now with Linux.

References

- Debian GNU/Linux
- Eagle software
- Scilab
- PonyProg programmer
- SP12 programmer
- Picprg programmer
- RTLinux web page
- RTAI web page
- AVR−GCC libc
- LF article : how to use and install the AVR−GCC

Webpages maintained by the LinuxFocus Editor team © Aleksandar Milovac "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org	Translation information: sr −−> −− : Aleksandar Milovac <aleksandar.milovac(at)gmail.com> sr −−> en: Aleksandar Milovac <amilovac(at)ptt.yu>

2005−03−27, generated by lfparser_pdf version 2.51