

## **AC 2010-2252: SCHEMATIC CAPTURE AND TECHNICAL DRAWING SOFTWARE FOR COMPUTER ENGINEERING**

### **Jonathan Hill, University of Hartford**

Jonathan Hill is an associate professor in Electrical and Computer Engineering at the University of Hartford in Connecticut. PhD and MSEE from Worcester Polytechnic Inst. in Worcester MA, and previously a project engineer at Digital Corp. He instructs graduate and undergraduate computer engineering computer courses, directs graduate research, and performs research involving embedded microprocessor based systems. His specific projects involve digital communications, signal processing, and intelligent instrumentation.

# Schematic Capture and Technical Drawing Software for Computer Engineering

## Abstract

Computer engineering students have specific needs involving schematic capture, printed circuit board layout, and block diagrams. The microprocessor devices course that I teach involves the construction of a small embedded microprocessor system and relies heavily on schematic capture and technical diagram software. In addition, many students construct printed circuit boards with their senior project and other projects. To address such needs at my University, I introduced two free software<sup>1</sup> packages, KiCad<sup>2</sup> and Dia<sup>3</sup>, to the curriculum, making the software available in our computer laboratories. Students also install this software on their home computers.

KiCad is software for the creation of electronic schematic diagrams and printed circuit board artwork. It is useful for everybody working in electronic design. In the microprocessor devices course I teach, due to the complexity of the circuits we build, it is practically impossible to draw schematics by hand. Such complexity is due to the detail required to actually construct such circuits. Even in a modest microprocessor system with an 8-bit data bus and a 16-bit address bus, keeping track of pins and pin numbers is problematic. In performing homework and project work alike, students absolutely require a powerful yet easy-to-use schematic capture tool.

Dia is software for the creation of technical diagrams. Unlike schematics that are detail-oriented, technical diagrams such as block diagrams are concept-oriented and play an important role in technical writing, helping to make a specific point or aspect of a system more understandable. Dia can be used to draw many different kinds of technical diagrams. It currently has special objects to help draw entity relationship diagrams, UML diagrams, flowcharts, network diagrams, and many other diagrams. While the interface and features are inspired by Visio, Dia is geared more for casual use.

As an educator I find that KiCad and Dia are useful and valuable to students. The point of free software<sup>1</sup> is that its users are at the core of its development so that you can join the community and get involved. In considering any free software project, look to its community of users and examine the content they provide. A quick Internet search for KiCad and Dia is telling as there are tutorials, documentation, and utilities online. In a nutshell, KiCad and Dia are exceptionally good examples of free software, and both are valuable to computer engineering students.

## Introduction

Computer engineering students have very specific needs in regards to schematics and other technical diagrams. Schematics are drawings that serve a very specific need in electrical and computer engineering, to represent actual circuits that can be constructed. Such drawings are inherently comprehensive in their detail and emphasize the circuit structure. As with a street map, a schematic does not aid much in understanding the behavior of the actual circuit. A schematic may sacrifice interpretability and understandability for comprehensiveness in detail. To summarize, a given schematic may be overwhelming, but in many cases it allows you to build the circuit without knowing how or why it actually works.

In microprocessor system design, due to the complexity of the circuits we build, it is practically impossible to draw schematics by hand. Such complexity is due to the detail required to actually construct such circuits. Even in a modest system with an 8-bit data bus and a 16-bit address bus, there are a seemingly endless number of additional required pins. On the processor, such pins include the interrupt request, reset, pins for the crystal, input and output ports, as well as power and ground. Keeping track of pins and pin numbers is problematic. To perform homework and project work alike, a powerful yet easy to use schematic capture tool is absolutely necessary.

Technical diagrams such as block diagrams and flowcharts are concept-oriented and are particularly useful in technical writing, helping to make a specific point of aspect of a system more understandable. Technical diagrams tend to be easier to draw by hand than schematics, which is of pedagogical value in terms of expressiveness. In teaching microprocessor-based system design, one exercise involves using a block diagram to describe a memory system that students implement by drawing a schematic and using a breadboard. Another exercise involves students interpreting a schematic to produce the corresponding block diagram. In both cases, students may also be asked to draw the corresponding memory map.

Installing KiCad<sup>2</sup> and Dia<sup>3</sup> on a MS Windows computer is easy. For Windows, there is a self-installing executable program for each application. Downloading the latest version may take a few minutes. There are no fees or licensing keys to deal with, and you can choose which directory you want to have the application installed in. With respect to installing in a laboratory, one issue to watch for is that environment variables are assigned for all users and likewise that in the program start menu the links are accessible by all users.

### **The KiCad Package**

KiCad is software for the creation of electronic schematic diagrams and printed circuit board artwork. It is useful for everybody working in electronic design, supporting the design of printed circuit boards with up to 16 copper layers. KiCad runs on GNU/Linux and MS Windows. It is occasionally tested on various versions of UNIX, especially FreeBSD and Solaris, and there are unofficial binaries for Macintosh. I primarily run KiCad on Windows Vista, as well as Windows XP. The project manager, also named KiCad, organizes files in a project and with the exception of Wings3D, launches the remaining tools:

- KiCad – Project manager
- Eeschema – Schematic capture tool
- Cvp pcb – Component selector tools
- Pcbnew – PC board layout tool
- Gerbview – Gerber file viewer tool
- Wings3D – Solid modeling software

Many of the tools include additional attendant software. The schematic capture tool and the PC board layout tool each have an editor for creating or modifying schematic symbols and the corresponding artwork, respectively. The layout tool also includes auto-router software. The

schematic capture tool can also produce several different types of netlist files, useful with other tools, such as in performing simulations.

To use KiCad, first create a new project. Next, produce schematics. The component selector is used to match schematic symbols with the corresponding device artwork or footprint. Artwork produced by the layout tool can be viewed in two or three dimensions. The layout tool also produces industry standard Gerber files and drill files so that you are free to choose your own PC board manufacturer.

Our University recently purchased a PC board milling machine that I intend to make use of with KiCad. Faculty in our Electronic Engineering Technology program also expressed an interest in possibly using KiCad and the milling machine in technology courses.

Figure 1 is of the schematic capture tool showing the detail of a schematic. With buttons along three sides of a window, the tools are neatly organized and uncluttered. The pull-down menus are neatly organized and easy to understand. Pop-up hints describe the function of each button. Additional information is presented along the bottom side of the window in a way that is not distracting. There is also an undo feature.

The layout tool has a rats-nest display feature that helps with initial component placement on the PC board. There are several features such as net highlighting and component dragging that help with manual layout and there is an automatic router as well. Figure 2 is a layout for the schematic in Figure 1, rendered in 3-D. This particular layout was manually routed.

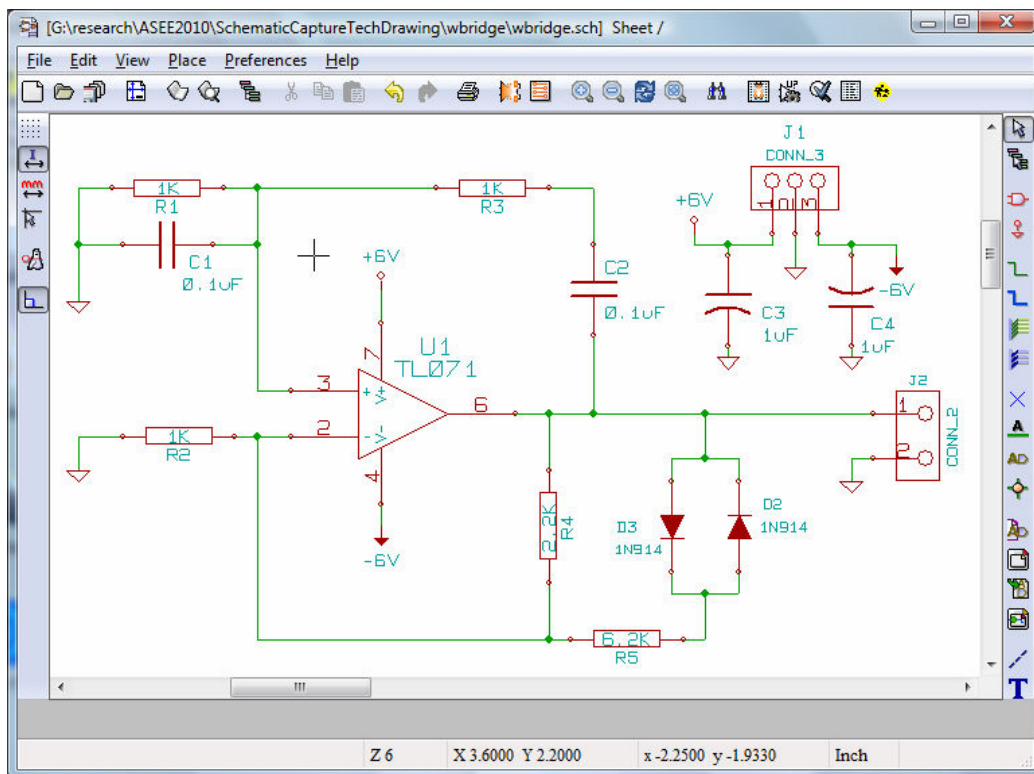


Figure 1: KiCad with Example Schematic

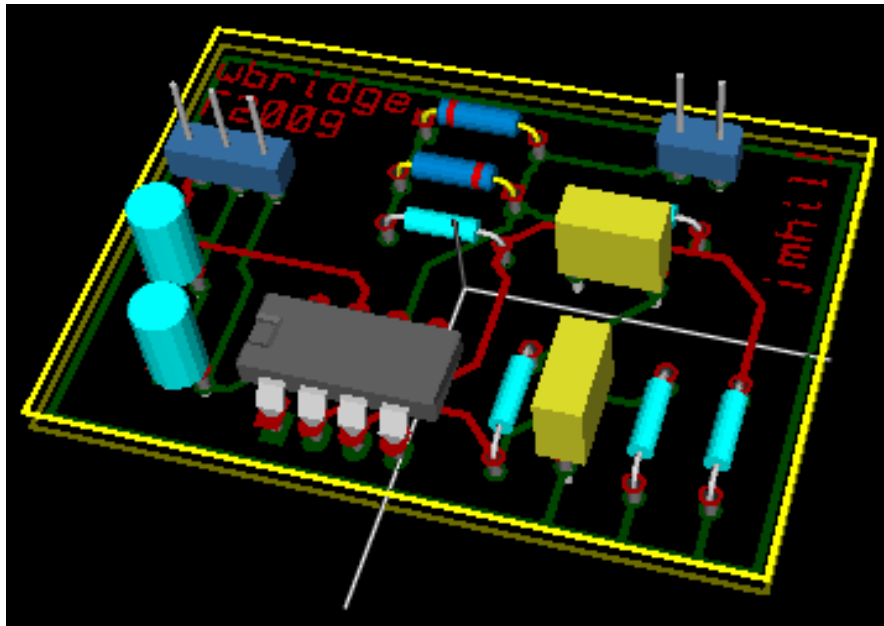


Figure 2: Example KiCad Layout Rendered in 3D

### Technical Diagrams with Dia

Dia is a graphical application for creating many different types of technical diagrams. Dia also supports many file formats. With respect to electrical and computer engineering, Dia can be used to conveniently produce at least the following types of diagrams:

- System block diagrams
- Control theory signal flow diagram
- Memory maps
- Software flowcharts
- Software object relationship diagrams
- Software UML diagrams of various types
- Computer network diagrams.

Using Dia is relatively easy. Starting Dia creates a canvas window for drawing a new diagram. Figure 3 shows Dia in action with a block diagram representing the circuit in Figure 1. The toolbox is the pane to the left; it contains many commonly used object shapes as well as numerous tool drawers that are categories of object shapes unique to a given diagram type. In this case the “Assorted” shapes tool drawer is visible. To add an object to a diagram, select the corresponding shape in the toolbox and click at the desired location in the canvas. Objects can be moved, resized, and have their properties modified. Objects can be connected by lines that will stretch to keep the objects connected. Objects can be organized into layers so that entire groups of objects can be edited without regard to other objects.

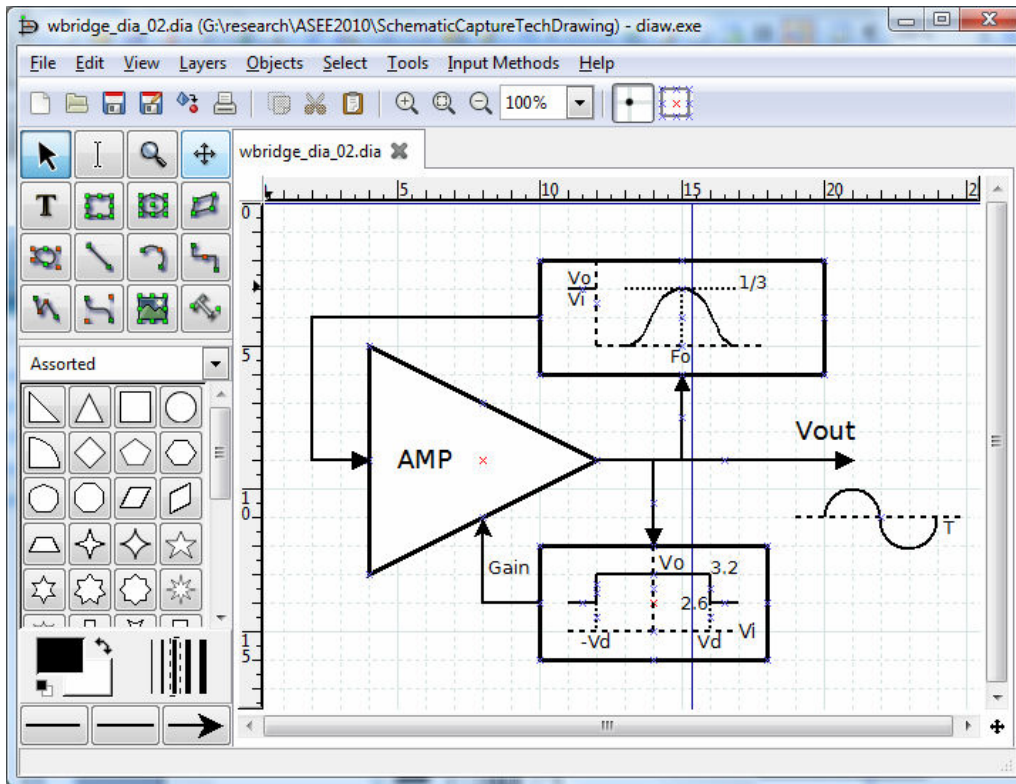


Figure 3: Dia in Action with an Example Block Diagram

In Figure 3 it should be clear that the circuit is the well known<sup>4</sup> Wien bridge oscillator. The upper process box is a band-pass filter centered on  $F_0$  and represents the resistor-capacitor network. The lower process box provides gain stabilization and represents the resistor-diode network. For a sinusoidal oscillation to occur, the overall loop gain must be unity.

### An Example Exercise

The memory system described by the block diagram in Figure 4 was designed for a Motorola 68HC11 microcontroller<sup>5</sup> and is similar to that for other 8/16-bit microcontrollers. This diagram emphasizes that the processor time-multiplexes the lower 8 address bits with data on the AD[7:0] bus. The address latch enable (ALE) signal enables the latch to store only address values. The R/W logic uses the E clock and the read/write signal to produce the read and write enable signals. The address decoder enables a given memory device so that it will appear in one region of the memory map. This memory system is said to use full address decoding as all the address bits are involved in addressing memory.

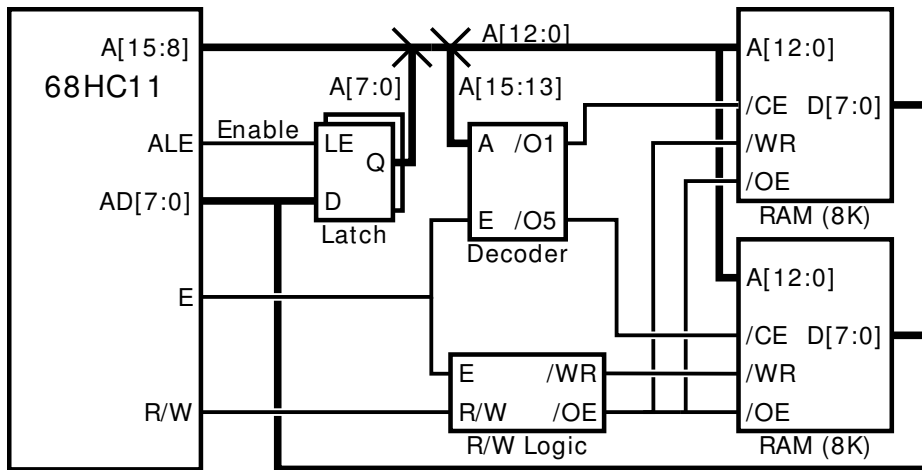


Figure 4: Example Memory System Block Diagram

Note that a block diagram may intentionally disregard certain details. Figure 4 does not indicate that A[15:8] is Port B and that AD[7:0] is Port C and that the processor is in expanded mode. Figure 5 is a schematic of the memory system corresponding to the block diagram in Figure 4. Note that the abstract bus concept used in Figure 4 corresponds in Figure 5 to loose collections of wires. The abstraction that is possible in a block diagram saves the reader from such detail.

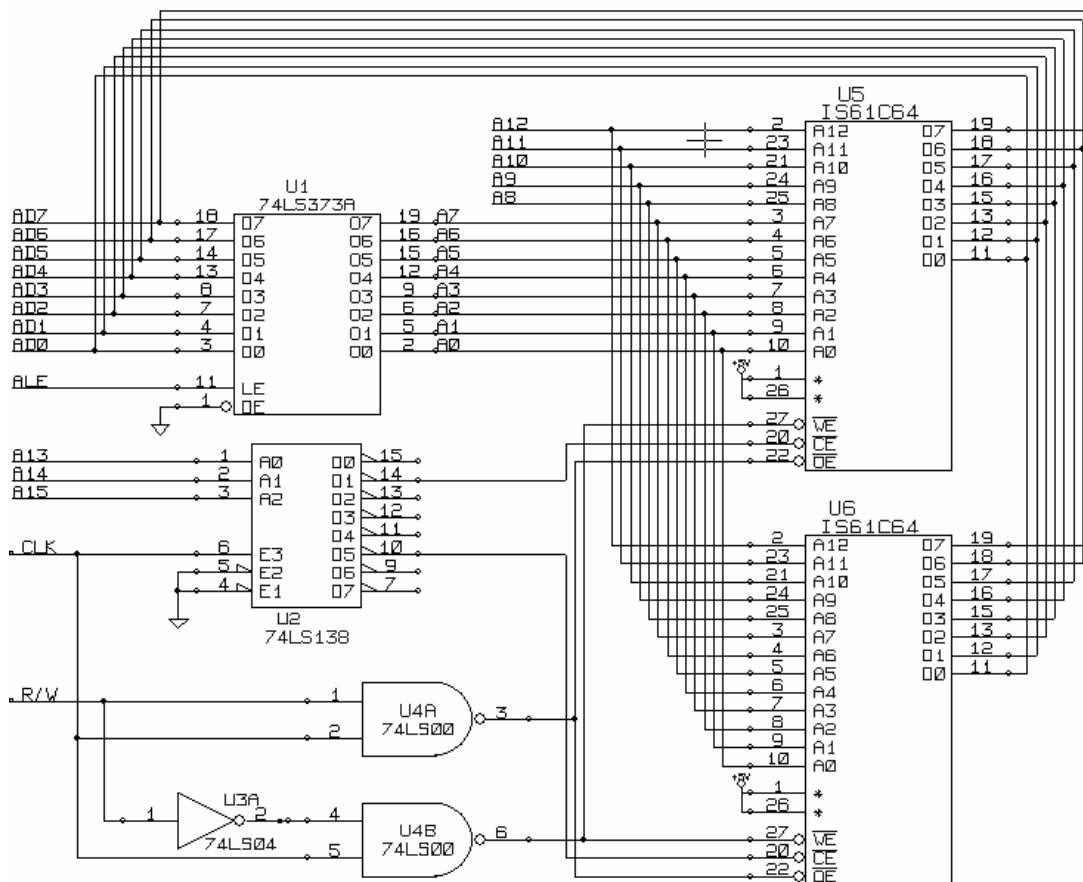


Figure 5: Example Schematic of a Simple Memory System

Once students become familiar with the general idea behind such a memory system then block diagrams can be used to present variations and related topics, such as how to combine the address decode logic with read/write logic, partial address decoding and what causes fold-back regions to appear in a memory map, as well as bank switching.

### Student Feedback

The feedback I receive in talking with students who have used KiCad and Dia is generally positive. I find that students who make full use of KiCad and have produced a PC board see KiCad as a joy to use but the opinion of other students is mixed. Feedback for Dia is generally mixed. To learn more about what students think, I distributed two surveys but received limited feedback. The first survey considers both KiCad and Dia.

- A1.If you have used KiCad to draw schematics, how useful did you find this aspect of the application to be? If you have not used KiCad to draw schematics, select N/A.
- A2.If you have used KiCad to produce a PC board layout, how useful did you find that aspect of the application to be? If you have not used KiCad to draw schematics, select N/A.
- A3.Do you have KiCad installed on a home computer? If you do not have a home computer, select N/A.
- A4.Please make any general comments about your experience with and the usability of KiCad.
- A5.Please make any comments on how you think our use of KiCad can be improved at our University.
- A6.If you have used Dia to draw technical diagrams, how useful did you find this application? If you have not used Dia, select N/A.
- A7.Do you have Dia installed on a home computer? If you do not have a home computer, select N/A.
- A8.Please make any comment about your experience with and the usability of Dia.
- A9.Please make any comments on how you think our use of Dia can be improved at our University.

Table 1 summarizes the results relevant to KiCad. Of those students who have used KiCad, fewer have made full use of its features, most notable PC board construction. It is surprising that not all students have KiCad installed on a home computer.

Table 1: Feedback on KiCad

	Replies	Not Useful	Less So	Only so	Mostly	Very
A1	5				60% (3)	40% (2)
A2	3			33% (1)		67% (2)

	Replies	Not Installed	Yes
A3	6	33% (2)	67% (4)

Question A4 illustrates the mix in feedback. The first comment is from a student who used KiCad at home and at work. I believe the second comment can be addressed by incorporating KiCad further into the curriculum. The third comment might be resolved with tutorial material.



- Used Kicad for the senior project and on some design for my employer.
- KiCad was a powerful tool but required a lot of self learning. A class would have been nice to have...
- It was an easy program to use, but seemed to have problems with floating and open points when doing schematics.

In question A5 students are asking for more support; in particular, they would like more appropriate tutorial material and that KiCad be introduced earlier in the curriculum.

- It would be nice if there was a tutoring lab for Engineering S/W.
- The tutorial on the board layouts and creating the footprints should be improved.
- ...more components for KiCad and an earlier encounter with it would be helpful.

A second survey provides additional feedback with respect to Dia. Question B2 was included as I suspected that Dia is seen as competing with similar software that some students know.

- B1. If you have used Dia to draw technical diagrams, how useful did you find this application? If you have not used Dia, select N/A.
- B2. Please compare Dia and its usability with other technical drawing software that you have used. If you have not used other such software, select N/A.
- B3. Do you have Dia installed on a home computer? If you do not have a home computer, select N/A.
- B4. Please make any comment about your experience with and the usability of Dia.
- B5. Please make any comments on how you think our use of Dia can be improved at our University.

Table 2 shows that feedback received with regard to Dia from both surveys is mixed and is quite different from that for KiCad, in particular a smaller percentage of students have Dia installed on a home computer. The discussion questions provide useful insight.

Table 2: Feedback on Dia

	Replies	Not Useful	Less So	Only so	Mostly	Very
A6 & B1	5		20% (1)	20% (1)	20% (1)	40% (2)

		Much Worse	Not as good	Same	Better Than	Much Better
B2	3	33% (1)			66% (2)	

	Replies	Not Installed	Yes
A7 & B3	7	43% (3)	57% (4)

In questions A8, B4, A9, and B5 I learned that while some students are just being introduced to technical diagram software, other students have very specific tastes. It is most important that we emphasize the need for such software and make sure that at least one free package is available for all of our students to use.

- Very easy tool bar to understand and very easy to use to create diagram[s].
- ...its worlds better than MSPaint or MSWord for block diagrams. However using it was quite frustrating. There HAS to be a better program.
- ...not good! There are tons of free and open source software that we can use that are better than Dia
- It was confusing at first. It takes a little getting used to...
- ...mention this program upfront to upperlevel (and even lower-level) classes... The other ... program people use is Visio, but that costs money, and I personally find [Visio] far more confusing, especially when ... trying to do a simple flow chart.

One weakness in Dia that was not raised by students is that as of version 0.97, it lacks a generalized tool for rotating components. Finally, given the need for more educational and tutorial content, I have applied for a University technology grant to develop new material and to consider new ways to incorporate these packages into the curriculum. I will also make additional components symbols available online<sup>6</sup>.

## Summary and Conclusion

KiCad and Dia are software applications that computer engineering students can use to design, describe, implement, and document electronic circuits. KiCad is software for the creation of electronic schematic diagrams and printed circuit board artwork. Dia is software for the creation of many types of technical diagrams. Unlike schematics that are detail-oriented, technical diagrams such as block diagrams and flowcharts are concept-oriented and play an important role in technical writing, helping to make a specific point or aspect of a system more understandable.

In talking with students and considering a survey, it is found that student feedback regarding KiCad<sup>2</sup> is generally positive but that more support is needed. With regard to Dia<sup>3</sup> I learned that while such software is new to some students, other students have very specific tastes. I believe that the need for such software must be emphasized and that at least one free technical drawing package be made available to students. In closing, both KiCad and Dia are exceptionally good examples of free software<sup>1</sup> and both are valuable to computer engineering students.

## Bibliography

1. Free Software Foundation: <http://www.fsf.org/>
2. KiCad web site: [http://www.lis.inpg.fr/realise\\_au\\_lis/kicad/](http://www.lis.inpg.fr/realise_au_lis/kicad/)
3. Dia web site: <http://live.gnome.org/Dia/>
4. Adel Sedra, Kenneth Smith, *Microelectronic Circuits*, 1998, Oxford University Press
5. Peter Spasov, *Microcontroller Technology, The 68HC11 and 68HC12*, fifth edition, copyright 2004 by Pearson Education, Inc.
6. Jonathan Hill, <http://uhaweb.hartford.edu/jmhill/supnnotes/KiCadDia/index.htm>