



## **CRIAÇÃO DE TUTORIAIS E RELATOS DE EXPERIÊNCIAS SOBRE A UTILIZAÇÃO DE *SOFTWARES* LIVRES NO ENSINO DE ENGENHARIA**

**Thaynara L. da Silva** – thaynaraleal.92@gmail.com

**Felipe N. Martins** – felipemartins@ifes.edu.br

**Wagner T. Costa** – wagnercosta@ifes.edu.br

IFES (Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo) - Campus Serra

Coordenadoria de Engenharia de Controle e Automação

NERA - Núcleo de Estudos em Robótica e Automação

Rodovia ES-010 – km 6,5

29173-087 - Serra – ES

**Resumo:** *O presente trabalho propõe a elaboração de tutoriais para facilitar o uso de softwares livres no curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo. Dessa forma, o texto aborda os programas Qucs, Scilab, KiCad e CircuiLab, apresentando uma breve explicação de seus recursos, histórico e funcionalidades. Resultados de algumas experiências com a utilização dessas ferramentas também são descritos. Pretende-se disponibilizar, gratuitamente, os tutoriais elaborados na página do NERA - Núcleo de Estudos em Robótica e Automação - na internet.*

**Palavras-chave:** *Software Livre, Tutorial, Ferramentas de Ensino.*

### **1. INTRODUÇÃO**

Prover um ensino de qualidade é uma das metas do IFES (Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo). Para oferecê-lo à sociedade objetivando suprir as necessidades do mercado por mão de obra qualificada, é necessário, dentre tantas outras variáveis, um investimento em estrutura, seja física (salas e computadores), seja computacional, como: *softwares* de simulação, matemática, modelagem e até mesmo sistema operacional (MORAN *et al.*, 2006).

O curso de Engenharia de Controle e Automação do IFES – Campus Serra é novo: a primeira turma ingressou no segundo semestre de 2007 e deve se formar no primeiro semestre de 2012. O Instituto possui licença para uso do sistema operacional Windows, porém adquirir e manter atualizadas as versões de sistemas operacionais e de todos os *softwares* comerciais usados no desenvolvimento da aprendizagem é um investimento elevado. Pensando na redução de custos e no envolvimento de alunos com outras ferramentas de ensino e de tecnologia que poderão fazer parte da vida deles fora do ambiente escolar, o IFES – Campus

Realização:



Organização:





Serra – passou a utilizar o Ubuntu, sistema operacional baseado em Linux do tipo *open source* (UBUNTU-BR, 2012), em suas máquinas. Foi percebido, infelizmente, que a migração dos alunos para o Ubuntu, em sua maioria acostumada a trabalhar no Windows, foi difícil. Pensando nisso, há uma proposta de a instituição ministrar cursos para que os alunos se adaptem de maneira mais rápida e fácil a essa plataforma computacional.

A instituição iniciou, também, a implantação de alguns outros meios para reduzir esse tipo de custo e fomentar a vivência dos alunos com outras ferramentas computacionais que atendam de maneira satisfatória os objetivos pertinentes à atividade que é proposta pelo professor. Outra ação foi a instalação de *softwares* livres, ou seja, *softwares* com código aberto (*open source*) onde os usuários possuem a liberdade de executar, copiar, distribuir, mudar e melhorar o *software* em edição (GNU PROJECT, 2012). Esse tipo de *software* oferece uma oportunidade dos alunos estudarem o seu código fonte, desenvolverem conhecimentos de programação e elaboração de novos códigos para correção de falhas ou até mesmo criação de novas ferramentas. Tais programas incluem simulação de circuitos elétricos (Qucs), desenho esquemático de circuitos elétricos (Kicad), cálculo numérico (Scilab e Octave), pacote de programas (Libre Office), ambiente de desenvolvimento para o Arduino (Arduino 0023) etc. A escola utiliza versões de demonstração de programas para sistemas supervisórios (Eclipse Scada e InTouch), *drives* de comunicação (RsLogix) etc.

Como esses *softwares* estavam sendo instalados nos computadores dos laboratórios, os professores perceberam certa dificuldade de os alunos conseguirem migrar para essa nova realidade de *softwares* livres. Em função disso, os professores Wagner Teixeira da Costa e Felipe Nascimento Martins, do grupo de pesquisa NERA – Núcleo de Estudos em Robótica e Automação, iniciaram o projeto de confecção de pequenos tutoriais com o intuito de investigar, estudar e descrever os principais fundamentos e comandos, bem como as ferramentas, dos seguintes programas: Scilab, Kicad, Qucs e CircuitLab, além do estudo de outros *softwares*. Esses tutoriais são voltados para aplicações do curso de Engenharia de Controle e Automação. Pretende-se disponibilizá-los gratuitamente na página do NERA na internet, que já possui tutorial para utilização do *software* livre Brixcc destinado à programação NXC para robôs da Lego (NERA, 2012).

## 2. OS SOFTWARES LIVRES UTILIZADOS

Como foi dito, inicialmente, foram escolhidos os seguintes programas para o estudo e consequente elaboração dos tutoriais: Scilab, Kicad e Qucs.

A escolha desses programas foi realizada a partir de um formulário preenchido por cada professor do curso de Engenharia de Controle e Automação do IFES – Campus Serra. Nele foram perguntadas quais disciplinas eram lecionadas pelos professores e quais eram os tipos de *softwares* (simulação numérica, de controle, de dinâmica dos fluidos, de circuitos elétricos, etc.) e tipos de simulação (circuitos analógicos ou digitais, vazão de fluidos, análise de sinais, e outros) que as disciplinas ministradas por ele necessitavam.

Com essa pesquisa, percebeu-se uma demanda maior para programas que realizam simulações numéricas, desenho esquemático de circuitos elétricos e simulação de circuitos. Sabendo disso, a escolha destes três programas em particular, entre tantos outros disponíveis, foi baseada em: experiências com trabalhos acadêmicos desenvolvidos pelos professores; existência de tutoriais e/ou apostilas e/ou livros; possibilidade de instalação no sistema operacional Linux e Windows e utilização ampla de outras instituições de ensino superior com essas ferramentas.



## 2.1. Scilab

Scilab (SCientific LABoratory) é um *software* criado em 1989 por um grupo francês de pesquisadores da INRIA (*Institut National de Recherche en Informatique et en Automatique*) e da ENPC (*École des Ponts ParisTech*) para resolução de problemas que envolvem cálculos numéricos e simulações de sistemas físicos utilizados nas áreas tecnológicas, químicas, medicinais, etc. É uma excelente ferramenta de suporte para linhas de pesquisa onde a utilização de computadores e algoritmos na resolução numérica de problemas é intensivo (PIRES, 2004). A partir de 1994 passou a ser um *software* livre do tipo *open source* mantido por várias empresas francesas por meio do consórcio Scilab. Em sua página na internet (SCILAB, 2012), são disponibilizados arquivos de instalação, manuais e *toolboxes* (bibliotecas). Ele pode ser utilizado como calculadora (quando os comandos são digitados diretamente no seu *prompt* de comando) ou como ambiente de programação (comandos digitados em editores de textos, sendo que ele possui uma linguagem própria de programação). As suas bibliotecas podem ser aumentadas a partir de *downloads* do próprio site do Scilab (LEITE, 2009).

## 2.2. Kicad

O Kicad é um *software* livre onde se cria e gerencia projetos para confecção de placas de circuito-impresso. Ele integra quatro aplicativos (Eeschema, Pcbnew, Cvpcb e Gerbview) para a criação de esquemas eletrônicos e elaboração de desenhos de circuitos impressos. Seu *download* está disponível em sua página oficial (KICAD, 2012).

O projeto Kicad foi criado em 1992 por Jean-Pierre Charras, pesquisador no LIS (Laboratoire d'Images et de Signaux) e professor de eletrônica no Instituto Universitário de Tecnologia de Saint Martin d'Hères em parceria com outros estudiosos. A linguagem de *script* e criação de *toolboxes* do Kicad pode ser o Python, uma linguagem orientada a objetos de alto nível e de uso livre que é executada nas plataformas Windows, Linux/Unix e Mac OS X (PYTHON, 2012).

A criação do projeto é feita no Kicad, como se ele fosse um *dashboard*. Com o Eeschema é feita a edição (há funções de desenho, de controle e de traçados) dos componentes nos esquemas (módulos).

O Eeschema gera o arquivo *Netlist* (contem as conexões elétricas da placa). O Cvpcb atrela cada componente da *Netlist* a outro equivalente na placa de circuito impresso, faz a associação ao módulo físico e adiciona essa informação ao próprio *Netlist*. O Pcbnew (editor de circuitos impressos) usa essa indexação feita pelo Cvpcb para integrar, automática e imediatamente, qualquer modificação do circuito de novos ou antigos módulos, de acordo com as conexões elétricas apresentadas no esquema inicial. Ele também possibilita visualizar o *layout* do circuito em três dimensões. Já o Gerbview é o visualizador de arquivos no formato Gerber.

## 2.3. Qucs

O Qucs (Quite Universal Circuit Simulator) é um simulador de circuitos em constante desenvolvimento desde 2004 por um grupo de pesquisadores liderados por Michael Margraf e Stefan Jahn (a versão mais atual em maio de 2012 é a 0.0.16). Desde o princípio, o programa foi elaborado para ser um *software* livre e, portanto, aberto para sua distribuição e



implementação pelos seus usuários. O GNU/Linux foi o sistema operacional base, contudo, ele é compilado em vários sistemas operacionais como o Solaris, NetBSD, FreeBSD, MacOS, Windows e Cygwin. Seu *download* está disponível na página oficial do projeto (QUCS, 2012).

O Qucs possui modelos de diversos componentes eletrônicos, tanto lineares (resistores, capacitores etc.) como não-lineares (diodos, transistores etc.). Cada modelo possui uma série de parâmetros que são definidos pelo usuário ou pré-definidos a partir de um arquivo de biblioteca.

A estrutura do QUCS é composta por quatro módulos integrados em um único programa, são eles: o editor de esquemático de circuitos elétricos (agrupamento de componentes elétricos e conexões do circuito); o simulador propriamente dito (sua entrada é o arquivo do esquemático); a biblioteca de componentes elétricos tanto lineares como não lineares (utilizadas para desenhar o esquemático); e as ferramentas de síntese e análise (algumas dessas ferramentas possuem ambiente gráfico próprio, mas seus resultados podem ser enviados para o ambiente principal diretamente). Todos os módulos trabalham de forma integrada em um mesmo ambiente gráfico.

#### **2.4. CircuitLab**

CircuitLab é uma ferramenta de uso livre, gratuito e de fácil manipulação de dados para edição esquemática e simulação de circuitos elétricos e eletrônicos que funciona no navegador de *internet*. Logo, não é necessária sua instalação através de um arquivo executável. Ele foi criado em 2011 por dois engenheiros eletrônicos norte-americanos: Humberto Evans e Mike Robbins (CIRCUITLAB, 2012). Um diferencial do CircuitLab é que o usuário abre uma conta, edita seus circuitos e os compartilha com outras pessoas. Isso o torna extremamente didático e aplicável durante o processo de aprendizagem.

#### **2.5. Experiências em sala de aula com *softwares* livres no IFES – Campus Serra**

A adoção de *software* livre (especialmente aqueles que rodam no sistema operacional Linux) requer uma mudança de paradigma e, por vezes, é recebida pelos alunos com muita restrição. Pensando nisso, no segundo semestre de 2011 foi feita uma experiência de utilização de dois *softwares* livres na turma de Introdução à Engenharia de Controle e Automação do IFES: SciLab e Qucs. Dois foram os objetivos principais para tal adoção. Primeiro, pretendeu-se introduzir, logo no início do curso de engenharia, a utilização de ferramentas gratuitas e abertas. Acredita-se que isso facilita a aceitação, por parte dos alunos, da utilização de *softwares* livres no decorrer do curso de engenharia. Segundo, a utilização dos programas SciLab e Qucs objetivou promover a aplicação de conceitos básicos estudados nas disciplinas de programação (Linguagem de Programação e Algoritmos e Estruturas de Dados) e Cálculo I, abordando também conceitos introdutórios de modelagem matemática, circuitos elétricos e controle de processos. Especificamente, os alunos realizaram a modelagem de um sistema simples de controle de nível num tanque, implementando a programação de seu modelo matemático descrito por uma equação diferencial e realizando simulações. Também puderam observar a analogia com o comportamento de um circuito elétrico cuja modelagem é feita pelo mesmo tipo de equação diferencial.

As Figuras 1 e 2 ilustram os sistemas simulados com utilização dos programas Qucs e SciLab, respectivamente. A Figura 1 mostra o circuito simulado e a respectiva variação de tensão  $V_c$  no capacitor após a alimentação do circuito (degrau de tensão). A Figura 2 mostra o





sistema de controle de nível e a respectiva variação de nível  $h$  obtida após uma variação súbita da vazão de entrada  $q_i$  (degrau de vazão).

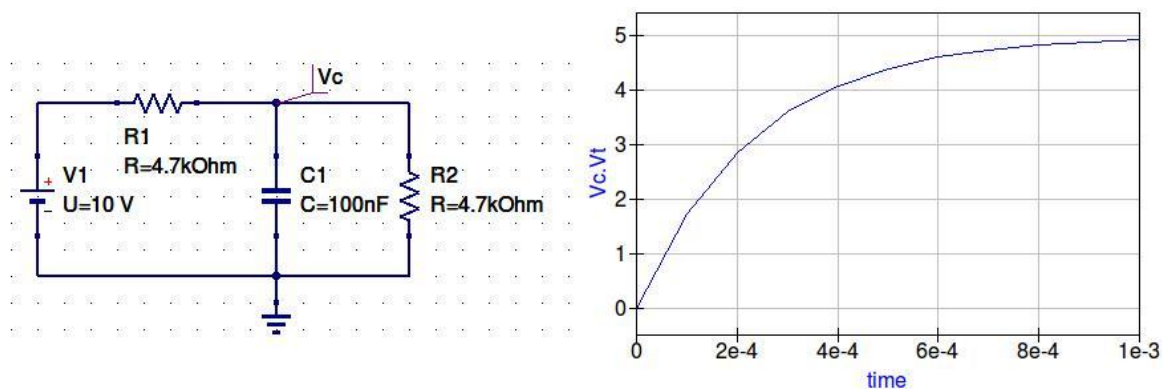


Figura 1. Circuito simulado com Qucs e gráfico de resposta da tensão  $V_c$  no tempo.

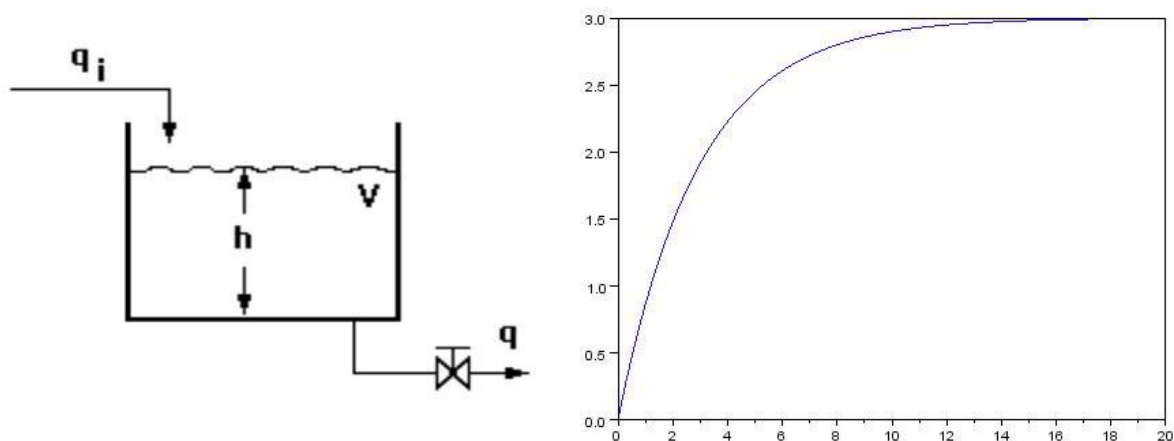


Figura 2. Sistema simulado com SciLab e gráfico de resposta do nível  $h$  no tempo.

Ao comparar os resultados, os alunos podem perceber a semelhança entre os comportamentos das variáveis dos dois processos estudados, apesar de sua distinção física. Notou-se que isso melhora a compreensão sobre os modelos matemáticos de ambos os sistemas. O resultado de tal experiência foi muito positivo e espera-se que os alunos continuem a utilizar as ferramentas aprendidas.

Na disciplina de Eletrônica de Potência o *software* CircuitLab está sendo utilizado para simulação de vários circuitos estudados em sala de aula. Esse *software*, além de ser de livre utilização, não exige instalação em computador, o que facilita bastante a adoção por parte de alunos e professores. Outro aspecto muito interessante do CircuitLab é o fato dele permitir o compartilhamento de circuitos na internet, criando uma página para o circuito construído em que se pode ver o esquema, realizar simulações e colocar comentários sobre o circuito em questão. Esses aspectos estão sendo explorados na disciplina de Eletrônica de Potência da seguinte maneira: o professor constrói o circuito básico, o compartilha, e solicita



aos alunos que realizem simulações para verificarem determinados comportamentos. As simulações, por vezes, exigem que os alunos modifiquem parâmetros ou a própria estrutura do circuito para obterem o resultado solicitado. A Figura 3 mostra um slide apresentado durante a aula, que ilustra a forma como o uso do CircuitLab tem sido estimulado.

## Exercício para casa

- Simular o circuito abaixo para verificar o funcionamento do retificador trifásico e a ocorrência do fenômeno da comutação. Variar os valores de  $L_i$  e da carga, e verificar a distorção provocada nas formas de onda da tensão.
- Circuito disponível em: <https://www.circuitlab.com/circuit/d7uwq/retificador-trifasico-filtro-indutivo/>

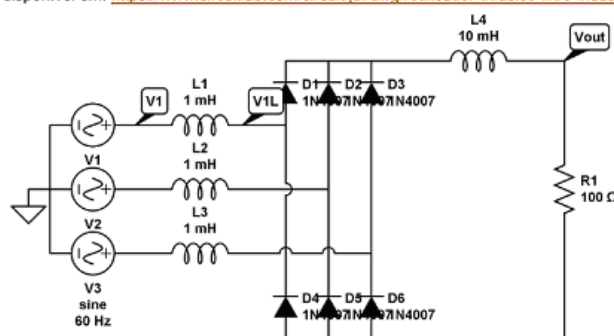


Figura 3: Exemplo de utilização do *software* CircuitLab na disciplina de Eletrônica de Potência.

Embora ainda possua limitações, como reduzido universo de componentes disponíveis, o CircuitLab tem se mostrado muito útil para demonstrar conceitos básicos de circuitos elétricos e de eletrônica. Tem-se observado que a abordagem adotada, que consiste em compartilhar circuitos e passar a tarefa de simulação para casa, estimula a participação do aluno e o estudo. O fato de não exigir instalação e de permitir comentários e compartilhamento de circuitos torna o CircuitLab uma ferramenta muito atrativa.

### 3. CONSIDERAÇÕES FINAIS

Embora sejam muito vantajosos, os *softwares* livres ainda são pouco utilizados e por vezes recebido com certa discriminação pelos alunos, o que demonstra uma grande necessidade de estudo e de divulgação, a começar no meio acadêmico. Em ambientes empresariais privados e federais têm-se usado *softwares* livres com maior frequência. Então começar a ter contato com alguns deles e usá-los durante a graduação ajuda no desenvolvimento das habilidades relativas a essas ferramentas e incentiva a familiarização com eles. O grupo NERA desenvolve, entre outros trabalhos, um estudo e pesquisa sobre a viabilidade de utilização de *softwares* livres nas disciplinas do curso de Engenharia de Controle e Automação no IFES – Campus Serra. Os tutoriais, inicialmente, dos programas Scilab, Qucs e Kicad estão em elaboração e, assim que forem terminados, serão disponibilizados gratuitamente na página do NERA na internet, como foi feito com o tutorial



do *software* livre Brixcc destinado à programação NXC para robôs da Lego. O grupo pretende elaborar outras versões, corrigidas e melhoradas destes tutoriais, além de preparar outros tutoriais sobre outros programas para facilitar o entendimento e possível interesse e iniciativa dos alunos em conhecerem as vantagens e particularidades desses programas.

### ***Agradecimentos***

Os autores agradecem a todos os servidores e professores do IFES – Campus Serra – que contribuíram para o desenvolvimento desse artigo e para a pesquisa e instalação dos *softwares* livres nos computadores da instituição tratados no trabalho.

### **REFERÊNCIAS BIBLIOGRÁFICAS**

CIRCUIT LAB: Página Oficial do Circuit Lab. Disponível em: <<https://www.circuitlab.com/>>. Acesso em: 01 jun. 2012.

GNU PROJCT: **A Definição de Software Livre**. Disponível em: <<http://www.gnu.org/philosophy/free-sw.pt-br.html>>. Acesso em: 31 maio 2012.

KICAD: **Página Oficial do Kicad**. Disponível em: <<http://iut-tice.ujf-grenoble.fr/kicad/>>. Acesso em: 02 jun. 2012.

LEITE, Mário. **Scilab - Uma Abordagem Prática e Didática**. Ass: Ciência Moderna, 2009.

M.MORAN, José; T.MASETTO, Marcos; A.BEHRENS, Marilda. **Ensino e educação de qualidade**. Texto publicado no livro “Novas Tecnologias e Mediação Pedagógica”, Campinas: Ed. Papyrus, 2006. p.12. Disponível em: <<http://www.eca.usp.br/prof/moran/qual.htm>>. Acesso em: 01 jun. 2012.

NERA – Núcleo de Estudos em Robótica e Automação. Disponível em: <<http://nera.sr.ifes.edu.br/>>. Acesso em: 2 jun 2012.

PIRES, Paulo Sérgio da Motta. **Introdução ao Scilab Versão 3.0**. Departamento de Engenharia de Computação e Automação - Universidade Federal do Rio Grande do Norte Natal-RN, Julho de 2004. Disponível em: <<http://www.dca.ufrn.br/~pmotta/sciport-3.0.pdf>>. Acesso em: 02 jun. 2012.

PYTHON: Official Website da Linguagem de Programação Python. Disponível em: <<http://www.python.org/>>. Acesso em: 03 jun. 2012.

QUCS: Página oficial do QUCS. Disponível em: <<http://qucs.sourceforge.net/>>. Acesso em: 02 jun. 2012.

SCILAB: **Página oficial do Scilab**. Disponível em: <<http://www.scilab.org>>. Acesso em: 02 jun. 2012.

UBUNUTU-BR: **Portal da comunidade da plataforma operacional Ubuntu Linux**. Disponível em: <<http://www.ubuntu-br.org>>. Acesso em: 30 maio 2012.



## **PRODUCTION OF TUTORIALS AND REPORTS OF EXPERIENCES ON THE USE OF FREE SOFTWARE FOR ENGINEERING EDUCATION**

**Abstract:** *This paper describes the development of tutorials to facilitate the use of open source software on the Control and Automation Engineering course of the Federal Institute of Education, Science and Technology of Espírito Santo. The text briefly discusses the resources, history and features of the following software tools: Qucs, Scilab, KiCad and CircuiLab. Finally, results of some experiences using these tools are described. The described tutorials will be available for download from NERA's (Núcleo de Estudos em Robótica e Automação) web page.*

**Key-words:** *Free Software, Tutorial, Teaching Tools.*