

stability control trivial. It can be proved that any signal can be decomposed over the gamma filter basis with an arbitrary small error in L2. Gradient methods to adapt the different parameters are available.

Discrete and continuous time implementations are analyzed. In discrete time is presented a new method to design filters. It utilizes active resistors in CMOS that allows a continuous programming of the filters parameters. Several SC circuits are also analyzed.

In continuous time is presented a capacitor multiplier solution, for very low frequency applications, with MOSFET-C filtering techniques.

For the discrete time topologies, simple models of CFT (Clock Feedthrough) and stray currents are presented. The models allow, in a first order approximation, to determine which are the critical components, and until which point these effects influence the filter performance. These models predict, in a relative simple form, the noise and offset of the different topologies.

Most of the topologies were implemented in a CMOS-VLSI technology to validate the simulations.

Título: Virtualização de Autómatos Programáveis

Title: Virtualization of Programmable Logic Controllers

Autor/Author: Jorge Augusto Fernandes Ferreira

Orientador/Advisor: João Pedro Estima de Oliveira

Data Apresentação/Acceptance Date: 07/94

Palavras Chave: Autómatos programáveis, linguagem de contactos, diagramas de escada, PLC virtual, virtualização, programação orientada por objectos, programação em ambiente Windows, interfaces gráficas, simulação orientada por objectos, simulação em tempo real

Key Words: Programmable logic controllers, ladder diagrams, virtual PLC, virtualization, object oriented programming, programming in Windows, graphic interfaces, object oriented simulation, real time simulation

Mestrado/M.S.

Resumo

Os autómatos programáveis (vulgarmente denominados PLCs, do inglês *Programmable Logic Controllers*) são bastante usados na indústria. Isto implica que um maior número de pessoas deve obter treino suficiente para desenvolver, manter e melhorar aplicações envolvendo PLCs, desde o simples comando de um motor até sofisticados controladores de processos. Neste trabalho foi desenvolvido um sistema (VPC) para o ensino da programação de PLCs, sem a necessidade de laboratórios de custo elevado, e para testar programas antes de usá-los em condições reais.

A ideia básica é a virtualização de um PLC. Isto significa que o PLC é construído por *software*, por exemplo num computador pessoal, de tal modo que as entradas são introduzidas pelo teclado ou pelo rato, e as saídas são visualizadas no monitor.

O sistema VPC contém três blocos principais: o editor de programação, o PLC virtual e o simulador. O editor suporta a conhecida linguagem de contactos (*Ladder*) e fornece as ferramentas necessárias para desenvolver programas para PLCs, utilizando uma interface *point-and-click* baseada em menus e ícones. O PLC virtual e o simulador implementam as operações comuns de um PLC real, tais como contactos abertos e fechados, saídas, temporizadores, contadores e registos de deslocamento. Ligado ao simulador existe um editor de estímulos para definir as entradas do PLC virtual.

O PLC virtual pode executar programas em três modos diferentes: a simulação rápida, onde os sinais aparecem sob a forma de diagramas temporais no editor de estímulos, o qual fornece também informação temporal; a simulação em tempo real, onde as entradas e as saídas são visualizadas sobre uma imagem do PLC real, cujo comportamento está a ser emulado, através dos usuais díodos emissores de luz; e a simulação interactiva, também uma simulação em tempo real, onde as entradas são definidas durante o processo de simulação e os resultados são visualizáveis numa janela específica.

Presentemente, o bloco PLC virtual, do sistema descrito, suporta um autómato específico, mas o uso de técnicas de programação por objectos permite uma fácil modificação, de modo a poder emular outros PLCs comerciais. O uso destas técnicas permite também a fácil adição de instruções *Ladder* mais avançadas. O sistema VPC corre como uma aplicação para o ambiente *Microsoft Windows*.

Abstract

Programmable Logic Controllers (PLC) are widely used in industry. This implies that more and more people must be trained in order to obtain enough knowledge to develop, maintain and upgrade applications involving PLCs, from simple motor commands to sophisticated process controllers. This project presents a software system (VPC) to teach PLC programming, without the need of costly practical class rooms, and to test programs before using them in real conditions.

The basic idea is the virtualization of the PLC. This means that the PLC is software constructed, for instance inside a personal computer, in such a way that the inputs are obtained with the keyboard or the mouse and the outputs are visualized on the screen.

The VPC system has three main blocks: the Editor, the Virtual PLC, and the Simulator. The Editor, with menus, icons and a point-and-click interface, supports the well known relay ladder programming language and supplies the tools needed to develop a PLC program. The Virtual PLC and the Simulator implement the common operations of a real PLC, such as open and closed contacts, outputs, timers, counters and shift registers. Linked to the Simulator there is a stimuli editor to input the conditions into the Virtual PLC.

The Virtual PLC can simulate programs in three different modes: the fast simulation, where the signals appear as a diagram of logic states inside the stimuli

editor, which also provides timing information; the real time simulation, where the inputs and outputs are impressed over an image of the real PLC, whose behaviour is being simulated, with the usual light emitting diodes; and the interactive simulation, also a real time simulation, where the inputs are defined by the user during the simulation process and the results are visualized in a specific window.

Currently, the virtual PLC block of the described system supports a specific PLC, but the use of object oriented techniques makes it easy to change the system, in order to provide support for other commercially available PLCs. The use of these techniques also allows an easy upgrade in order to integrate more advanced ladder instructions, if desired. The VPC system runs as a Microsoft Windows application.

Título: Biblioteca de Software para Cálculo Matemático e Processamento Digital de Sinal

Title: Software Library for Mathematical Calculus and Digital Signal Processing

Autor/Author: Miguel Augusto Mendes Oliveira e Silva

Orientador/Advisor: Francisco Vaz

Data Apresentação/Acceptance Date: 09/94

Palavras Chave: Programação orientada por objectos, Eiffel, C++, cálculo matemático, processamento digital de sinal, linguagens de programação

Key Words: Object-oriented programming, eiffel, C++, mathematical calculus, digital signal processing, programming languages

Mestrado/M.Sc.

Resumo

Neste trabalho é feito um estudo sobre estruturação de software para aplicação em cálculo matemático e em processamento digital de sinal.

Baseado nesse estudo, fez-se o projecto e implementação de uma biblioteca de software. A estruturação desenvolvida para essa biblioteca assentou sobre o paradigma orientado a objectos, contrastando com a estruturação funcional clássica, usada pela maioria das bibliotecas e aplicações matemáticas existentes. Mostra-se que apesar de numa primeira aproximação a estruturação funcional parecer ser a opção mais natural nesta área, ela nem sempre representa adequadamente os conceitos que temos das entidades matemáticas, sendo mesmo a estrutura orientada a objectos a que melhor as representa e implementa.

Um estudo comparativo dos dois tipos de estruturação para a biblioteca de software mostra as fraquezas de cada uma: pouca versatilidade na estrutura funcional; menor desempenho na estrutura orientada a objectos.

Do projecto da biblioteca tiram-se um conjunto de exigências a serem observadas pela linguagem de programação na sua implementação.

Mostram-se as limitações da maioria das linguagens de programação orientadas a objectos existentes, com um

ênfase particular sobre o C++, por ser a linguagem usada na implementação.

Desenvolveu-se uma linguagem de utilização da biblioteca que facilita e resolve alguns problemas do uso directo da biblioteca.

É feita uma aplicação em processamento digital de sinal exemplificando a utilização da biblioteca, onde se salienta as vantagens da estruturação seguida.

Faz-se uma avaliação aproximada do desempenho da biblioteca em relação a versões com estrutura funcional.

Abstract

This work presents a study about structuring application software for numerical calculus and digital signal processing.

Based on this study, a software library was projected and implemented.

The type of structure developed for this library is based on the object-oriented paradigm, contrasting with the classical functional structuring used by most of the existing libraries and mathematical applications.

It is shown that, although in a first glance the functional structure looks like the natural choice in this area, it doesn't always represent in a proper way the concepts that we have about mathematical entities, being the object-oriented approach the one that better represents and implements them.

A comparative study of these two types of structures shows the weakness of each one: little versatility in the functional structure; less efficiency in the object-oriented one.

A set of demands to be observed by the programming language in the library implementation were identified from the library project.

The limitations of most of the object-oriented programming languages are discussed, with a special emphasis on C++, because it was the language used on the implementation.

A library usage language was developed, which simplified and solved some problems in the direct use of the library.

A digital signal processing example was presented to exemplify the utilization of the library, where the advantages on the structure followed are apparent.

An approximate evaluation is made of the performance of the library in relation to functional structure versions.

Título: Sistema de Control SP-LD: Visão Geral de Arquitectura, Hardware e Software de Sistema

Title: SP-LD Control System: Architecture Overview, System Hardware and Software

Autor/Author: António Jaime Ramos

Orientador/Advisor: A. Manuel de Oliveira Duarte

Data Apresentação/Acceptance Date: 09/94

Palavras Chave: SP-LD, sistema de controlo, arquitectura, descrição de software, descrição de hardware