

Adriano Francisco Branco

**Um modelo de programação para RSSF com
suporte à reconfiguração dinâmica de
aplicações**

Dissertação de Mestrado

Dissertação apresentada ao Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio como requisito parcial para obtenção do grau de Mestre em Informática.

Orientador : Prof. Noemi de La Rocque Rodriguez
Co-Orientador: Prof. Silvana Rossetto

Rio de Janeiro
Abril de 2011



Adriano Francisco Branco

**Um modelo de programação para RSSF com
suporte à reconfiguração dinâmica de
aplicações**

Dissertação apresentada ao Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio como requisito parcial para obtenção do grau de Mestre em Informática. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Noemi de La Rocque Rodriguez

Orientador

Departamento de Informática — PUC-Rio

Prof. Silvana Rossetto

Co-Orientador

Departamento de Ciência da Computação — UFRJ

Prof. Luci Pirmez

UFRJ

Prof. Renato Fontoura de Gusmão Cerqueira

Departamento de Informática — PUC-Rio

Dr. Renato Figueiró Maia

Pesquisador — PUC-Rio

Prof. José Eugenio Leal

Coordenador Setorial do Centro Técnico Científico — PUC-Rio

Rio de Janeiro, 6 de Abril de 2011

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Adriano Francisco Branco

Graduou-se em Engenharia Eletrônica pelo Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (CEFET/RJ). Trabalhou mais de 12 anos em consultoria de projetos de integração de sistemas, principalmente nas áreas de automação industrial e telecomunicações. Também tem passagem pela área de P&D, com trabalhos no CERN (Suíça) e CBPF/CNPq. Teve oportunidade de trabalhar em projetos de eletrônica digital de interfaces de comunicação, computadores 8 e 32 bits com CPUs CISC e RISC, sistemas de controles e aquisição de dados. Têm experiência em sistemas operacionais como CP/M, DOS, Windows, VMS e diversas variações de Unix. Trabalhou com integração de sistemas desenvolvendo ferramentas customizadas ou utilizando ferramentas comerciais para integração em tempo real, síncronas e assíncronas. Também tem grande experiência em gerência de projetos e coordenação de equipes de desenvolvimento de sistemas.

Ficha Catalográfica

Branco, Adriano Francisco

Um modelo de programação para RSSF com suporte à reconfiguração dinâmica de aplicações / Adriano Francisco Branco; orientador: Noemi de La Rocque Rodriguez; co-orientador: Silvana Rossetto. — 2011.

98 f. : il. (color); 30 cm

1. Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2011.

Inclui bibliografia.

1. Informática – Teses. 2. Rede de Sensores sem Fio (RSSF). 3. Sistemas Distribuídos. 4. Modelo de Programação. 5. Funções parametrizáveis. 6. Máquina de Estados Finitos (FSM). 7. Reconfiguração Dinâmica. I. Rodriguez, Noemi de La Rocque. II. Rossetto, Silvana. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

Agradecimentos

Aos meus orientadores Professores Noemi Rodriguez e Silvana Rossetto pelo apoio e incentivo para a realização deste trabalho.

Ao CNPq e à PUC-Rio, pelos auxílios concedidos, sem os quais este trabalho não poderia ter sido realizado.

À minha esposa, que me acompanhou todo esse tempo com apoio direto e indireto.

Aos meus pais, irmãos, família e amigos, que me apoiaram mesmo com minha ausência na vida familiar.

A todos os colegas, professores e funcionários do Departamento de Informática da PUC-Rio, pelo companheirismo, aprendizado e auxílio.

Resumo

Branco, Adriano Francisco; Rodriguez, Noemi de La Rocque; Rossetto, Silvana. **Um modelo de programação para RSSF com suporte à reconfiguração dinâmica de aplicações**. Rio de Janeiro, 2011. 98p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Algumas características básicas das redes de sensores sem fio (RSSF) dificultam as tarefas de criação e reconfiguração de aplicações. Nesse trabalho apresentamos um modelo de programação que pretende simplificar essas tarefas. O modelo se baseia no uso conjunto de funções parametrizáveis e de máquinas de estados finitos, e permite a implementação de diferentes tipos de aplicações para redes de sensores sem fio e a configuração remota dessas aplicações. Descrevemos alguns testes para avaliar o quanto esse modelo pode facilitar o desenvolvimento de novas aplicações, o quanto é fácil aplicar novas alterações sobre as aplicações em execução, e o impacto na quantidade de mensagens na rede por conta do uso da configuração remota.

Palavras-chave

Rede de Sensores sem Fio (RSSF); Sistemas Distribuídos; Modelo de Programação; Funções parametrizáveis; Máquina de Estados Finitos (FSM); Reconfiguração Dinâmica.

Abstract

Branco, Adriano Francisco; Rodriguez, Noemi de La Rocque (Advisor); Rossetto, Silvana (Co-Advisor). **A WSN programming model with a dynamic reconfiguration support**. Rio de Janeiro, 2011. 98p. MSc Dissertation — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Some basic characteristics of wireless sensor networks (WSN) make application creation and reconfiguration difficult tasks. A programming model is presented to simplify these tasks. This model is based on a set of parametrized components and on a Finite State Machine, and allows the remote configuration of different applications over the same set of installed components. We describe some tests to evaluate its impact on the development process, and the ease of applying modifications to a running application. We also measure the additional impact of remote configuration on network activity.

Keywords

Wireless Sensor Network (WSN); Distributed Systems; Programming Model; Parametrized functions; Finite State Machine (FSM); Dynamic reconfiguration.

Sumário

| | | |
|-----|--|----|
| 1 | Introdução | 10 |
| 1.1 | Abordagem | 11 |
| 1.2 | Objetivos e contribuições | 12 |
| 1.3 | Trabalhos relacionados | 13 |
| 1.4 | Organização do documento | 15 |
| 2 | TinyOS e NesC | 16 |
| 2.1 | NesC | 16 |
| 2.2 | TinyOS | 17 |
| 3 | O modelo de programação proposto | 21 |
| 3.1 | Padrões de interação para aplicações em RSSF | 21 |
| 3.2 | Conjunto das funcionalidades básicas | 25 |
| 3.3 | Controle de fluxo baseado em FSM | 30 |
| 3.4 | Exemplo simplificado de aplicação do modelo de programação | 32 |
| 3.5 | Reconfiguração | 33 |
| 4 | Implementação | 35 |
| 4.1 | Visão operacional geral | 35 |
| 4.2 | Etapas de operação | 39 |
| 4.3 | Padrão de implementação para as camadas funcionais | 42 |
| 4.4 | Exemplo de um fluxo de execução | 44 |
| 4.5 | Experiências no desenvolvimento e novas funcionalidades | 47 |
| 5 | Avaliação | 52 |
| 5.1 | Implementação das aplicações de referência | 52 |
| 5.2 | Avaliação da facilidade de programação e do impacto da configuração remota | 57 |
| 6 | Conclusão | 66 |
| 6.1 | Trabalhos futuros | 67 |
| 7 | Referências Bibliográficas | 69 |
| A | Detalhamento dos padrões de interação | 72 |
| B | Diagramas das FSMs e Parâmetros de Configuração | 78 |
| B.1 | Aplicação utilizada nos testes | 78 |
| B.2 | Aplicações de Referência | 80 |
| C | Ambiente de execução para os testes | 92 |
| D | Interfaces e Componentes NesC do sistema | 94 |
| E | Formatação dos arquivos de configuração XML | 97 |

Lista de figuras

| | | |
|-----|--|----|
| 3.1 | Camadas da arquitetura de execução | 26 |
| 3.2 | Arquitetura com as funcionalidades básicas | 30 |
| 3.3 | Arquitetura de controle da FSM | 31 |
| 3.4 | Sintaxe FSM e exemplo de tabela de transições | 32 |
| 3.5 | Aplicação exemplo - coleta periódica de temperatura | 33 |
| 4.1 | Módulos utilizados na estação servidora | 36 |
| 4.2 | Arquitetura básica das camadas funcionais | 42 |
| 4.3 | Configuração para uma aplicação de monitoração de temperatura | 44 |
| 4.4 | FSM para monitoração periódica da temperatura média | 45 |
| 4.5 | Exemplo de fluxo de execução para uma aplicação de agregação | 46 |
| 5.1 | Aplicação 1 - FSM para inicialização do mote e controle geral da aplicação | 53 |
| 5.2 | Aplicação 1 - FSM para geração do alarme | 54 |
| 5.3 | Rede de referência | 58 |
| 5.4 | (a) FSM para controle de uma aplicação de coleta e (b) Definição dos respectivos parâmetros. | 59 |
| 5.5 | Distribuição no tempo de cada etapa de operação | 59 |
| 5.6 | (a) FSM para controle de uma aplicação de alarme e (b) Definição dos respectivos parâmetros | 62 |
| B.1 | Aplicação 1 - FSM para Inicialização do Mote e controle geral | 80 |
| B.2 | Aplicação 1 - FSM para geração do alarme | 80 |
| B.3 | Aplicação 2 - FSM para Inicialização do Mote e controle geral | 81 |
| B.4 | Aplicação 2 - FSM para geração do alarme | 81 |
| B.5 | Aplicação 2 - FSM para monitoração periódica | 82 |
| B.6 | Aplicação 3 - FSM para Inicialização do Mote e controle geral | 82 |
| B.7 | Aplicação 3 - FSM para monitoração periódica | 82 |
| B.8 | Aplicação 3 - FSM para reserva de vagas | 83 |
| C.1 | Módulos utilizados para execução dos testes no simulador | 92 |

Lista de tabelas

| | | |
|------|---|----|
| 3.1 | Padrões de interação identificados | 24 |
| 3.2 | Estrutura dos parâmetros de configuração | 29 |
| 3.3 | Campos de um registro da tabela de transições | 32 |
| 4.1 | Principais ações e eventos disponíveis para a aplicação | 40 |
| 4.2 | Biblioteca de funções parametrizáveis | 43 |
| 4.3 | Transições utilizadas para o exemplo de fluxo | 45 |
| 5.1 | Primeiro Teste: Resultado para as métricas da implementação no modelo proposto | 59 |
| 5.2 | Sumarização da quantidade de mensagens enviadas para cada etapa e os respectivos contadores | 60 |
| 5.3 | Quantidade de mensagens enviadas para cada tipo de operação | 60 |
| 5.4 | Primeiro Teste: Resultado para as métricas da implementação alternativa em NesC | 61 |
| 5.5 | Segundo Teste: Resultado para as métricas da implementação no modelo proposto | 62 |
| 5.6 | Quantidade de envios para a operação de reconfiguração remota | 62 |
| 5.7 | Segundo Teste: Resultado para as métricas da implementação alternativa em NesC | 63 |
| 5.8 | Resumo comparativo entre as duas implementações | 65 |
| A.1 | Distribuição dos Padrões de interação X Suporte de comunicação | 73 |
| A.2 | Padrões Identificados | 74 |
| B.1 | Configuração da primeira FSM do primeiro teste | 78 |
| B.2 | Configuração da segunda FSM do primeiro teste | 79 |
| B.3 | Configuração da terceira FSM utilizada no segundo teste | 79 |
| B.4 | Estrutura dos parâmetros de configuração para Aplicação 1 | 83 |
| B.5 | Estrutura dos parâmetros de configuração para Aplicação 2 | 84 |
| B.6 | Estrutura dos parâmetros de configuração para Aplicação 3 | 85 |
| B.7 | Aplicação 1 - Máquina de estados 1 (Inicialização) | 86 |
| B.8 | Aplicação 1 - Máquina de estados 2 (Alarme) | 87 |
| B.9 | Aplicação 2 - Máquina de estados 1 (Inicialização) | 88 |
| B.10 | Aplicação 2 - Máquina de estados 2 (Alarme) | 89 |
| B.11 | Aplicação 2 - Máquina de estados 3 (Monitoração) | 90 |
| B.12 | Aplicação 3 - Máquina de estados 1 (Inicialização) | 90 |
| B.13 | Aplicação 3 - Máquina de estados 2 (Monitoração) e 3 (Reserva) | 91 |
| D.1 | Interfaces das Camadas Funcionais | 94 |
| D.2 | Novos componentes NesC | 96 |