

# Real-Time Software Design and Analysis of Reconfigurable Multi-Sensor Based Systems

David Bernard Stewart

Submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
in Electrical Engineering

Department of Electrical and Computer Engineering  
Carnegie Mellon University  
Pittsburgh, Pennsylvania 15213-3890

April 1, 1994

Copyright © 1994 Carnegie Mellon University

**ABB Inc.**

**EXHIBIT 1004**

Page 1 of 196



## Table of Contents

<b>List of Illustrations</b> .....	<b>vii</b>
<b>List of Abbreviations and Acronyms</b> .....	<b>ix</b>
<b>Abstract</b> .....	<b>xi</b>
<b>Acknowledgments</b> .....	<b>xiii</b>
<b>Chapter 1</b>	
<b>Introduction</b> .....	<b>1</b>
1.1 Overview.....	1
1.2 Motivation.....	1
1.3 Goals and Contributions .....	4
1.4 Organization of Thesis.....	5
<b>Chapter 2</b>	
<b>A Review of Related Research</b> .....	<b>7</b>
2.1 Introduction.....	7
2.2 Software Reuse for Real-Time Applications .....	7
2.2.1 Software Synthesis.....	7
2.2.2 Interface Adaptation Methods.....	9
2.2.3 Object-based and Object-oriented design.....	10
2.3 Port automaton theory.....	11
2.4 Reconfigurable Real-Time Systems .....	12
2.5 Software Architectures for Robotics.....	13
2.6 Real-Time Systems Theory.....	14
2.7 Real-time operating systems .....	16
2.8 Summary .....	17

*iii*

Page 3 of 196

<b>Chapter 3</b>	
<b>Port-Based Objects.....</b>	<b>19</b>
3.1 Introduction.....	19
3.2 Terminology.....	19
3.3 Port-Based Objects.....	24
3.3.1 Configuration Verification.....	27
3.4 Control Module Integration .....	28
3.5 Generic Framework of a Port-Based Object.....	30
3.6 C-language Interface Specification for Port-Based Objects .....	35
3.7 Automatic Generation of the C-Language Framework .....	42
3.8 Reconfigurable Module Specification: The .rmod file .....	43
3.8.1 Combining Objects .....	46
3.9 Software Libraries.....	47
3.10 Dynamic Reconfigurability.....	48
3.11 Summary .....	52

<b>Chapter 4</b>	
<b>Software Assembly .....</b>	<b>53</b>
4.1 Introduction.....	53
4.2 Structure of SBS master task .....	53
4.3 The Subsystem Definition (.sbs) File.....	54
4.4 Interface Commands .....	55
4.4.1 Command-line interface.....	55
4.4.2 External Subsystem Interface .....	59
4.4.3 Graphical User Interface.....	60
4.4.4 Autonomous Program .....	61
4.5 SBS Subsystem Internals .....	63
4.5.1 SBS Master Task Initialization .....	64
4.5.2 Spawn: creating a new task.....	67
4.5.3 Sending Signals to Tasks .....	68
4.6 Summary .....	69

<b>Chapter 5</b>	
<b>Multiprocessor Real-Time Communication .....</b>	<b>71</b>
5.1 Introduction.....	71
5.2 Review Of A Typical Backplane Configuration.....	72
5.3 Express Mail .....	75
5.3.1 Mailbox Structure .....	77
5.3.2 Interfacing with the Host Workstation.....	82
5.4 Basic IPC .....	84
5.4.1 Dynamically Allocatable Global Shared Memory.....	86
5.4.2 Remote Semaphores.....	87
5.4.3 Prioritized Message Passing .....	88
5.5 Global State Variable Table Mechanism .....	92
5.5.1 Implementation Overview .....	94
5.5.2 State Variable Configuration File.....	96
5.6 Inter-subsystem Communication .....	98
5.6.1 Chimera Implementation of TBUF .....	101
5.7 Summary .....	102
<b>Chapter 6</b>	
<b>Real-time Scheduling for Reconfigurable Systems .....</b>	<b>103</b>
6.1 Introduction.....	103
6.2 Local Real-Time Scheduling .....	104
6.2.1 Rate Monotonic Algorithm .....	104
6.2.2 Earliest-Deadline-First Scheduling Algorithm .....	105
6.2.3 Maximum-Urgency-First Algorithm (MUF).....	106
6.2.4 Considering Data Flow in Scheduling Priority Assignment.....	110
6.3 Timing Failure Detection and Handling .....	113
6.4 Soft Real-Time Tasks .....	116
6.4.1 Implementation .....	118
6.5 Aperiodic Servers.....	119
6.5.1 Aperiodic Servers for the RM Algorithm.....	120
6.5.2 MUF Aperiodic Servers.....	121
6.5.3 Comparison of Aperiodic Servers.....	127

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

## LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

## FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

## E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.