

Microprocessor System Data Transfer Interface Design: An Expert System Approach Using Signal Timing Behavioral Patterns

by

BENEDIKT THEODOR HUBER

M.Sc., University of Victoria, 1986

B.Sc., University of Victoria, 1983

A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Electrical and Computer Engineering

We accept this dissertation as conforming
to the required standard

Dr. K. F. Li, Supervisor, Dept. of Electr. & Comp. Eng.

Dr. N. J. Dimopoulos, Member, Dept. of Electr. & Comp. Eng.

Dr. E. G. Manning, Member, Dept. of Electr. & Comp. Eng.

Dr. M. H. Van Emden, Outside Member, Dept. of Computer Science

Dr. A. J. Al-Khalili, External Examiner, Dept. of Electr. & Comp. Eng.,
Concordia University

© BENEDIKT THEODOR HUBER, 1998
University of Victoria

*All rights reserved. This dissertation may not be reproduced in whole or in part by
photocopy or other means, without the permission of the author.*

Supervisor: Dr. K. F. Li

Abstract

DAME (**D**esign Automation of **M**icroprocessor-based systems using an **E**xpert system approach) is an expert system for configuring and designing a customized microprocessor systems from original specifications. This work deals with the development of the data transfer interface design module in DAME: the Interface Designer.

The automated Interface Designer is developed by extracting common features, functions and behavior of microprocessor components and representing them using knowledge representation techniques. The design is accomplished through pattern matching, by performing actions and procedures based on recognition of the standard behavior patterns of microprocessor component signals.

The development of the Interface Designer production system is divided into three parts: a hierarchical network of frames that represents the components, a hierarchical network of frames that represents the interface and a set of forward chaining rules that represents the design expertise. Equivalent abstraction levels are developed for the component model, interface model and design rules, allowing the design process to proceed using a top-down methodology.

The component behavior is abstracted at several levels. At the more abstract behavior level, the data transfer behavior is divided into a set of fundamental information transfers, namely the address, data, request, direction, type, delay, size and width information transfers. At the more detailed level, each information transfer is divided into state and timing information transfers, where state information represents the conceptual meaning of the state of a signal, and the timing information specifies when the state information is usable. Finally, the timing information is represented using a set of propagation delay invariant timing patterns. Only a limited number of timing patterns is required, thus allowing a limited number of design rules to accomplish the interface design.

Interface design is carried out by sub-dividing the interface into progressively more detailed interface sub-blocks, until eventually the interface is built up from a set of parameterized primitive circuits that represents the lowest level basic building blocks of an interface. The set of primitive circuits developed gives the Interface Designer the ability to connect signals based on the timing patterns. The timing behavior of the output of the interface is determined as a function of the primitive circuit parameters and the timing behavior of the input of the interface. Once the interface design is complete, the output

timing behavior of the interface is verified to assure that all component input timing constraints are satisfied.

Each of the primitive circuits developed is also given using VHDL. This allows the complete interface to be generated using VHDL code once the design is complete, permitting simulation for verification and synthesis for implementation of the interface. Several small test systems are designed and simulated to check the validity of the Interface Designer.

Examiners:

Dr. K. F. Li, Supervisor, Dept. of Electr. & Comp. Eng.

Dr. N. J. Dimopoulos, Member, Dept. of Electr. & Comp. Eng.

Dr. E. G. Manning, Member, Dept. of Electr. & Comp. Eng.

Dr. M. H. Van Emden, Outside Member, Dept. of Computer Science

Dr. A. J. Al-Khalili, External Examiner, Dept. of Electr. & Comp. Eng.,
Concordia University

Table of Contents

Abstract	ii
Table of Contents	iv
List of Figures	ix
List of Tables	xiii
Acknowledgments	xv
Glossary	xvi
Chapter 1: Introduction	1
1.1 Rationale Behind Microprocessor System Design Using an Expert System Approach	1
1.2 Work Covered in this Dissertation	1
1.3 Dissertation Organization	5
1.4 Trademarks	6
Chapter 2: Background	7
2.1 Microprocessor Systems	7
2.1.1 Microprocessor System Interface Protocols	10
2.1.2 Microprocessor System Component Properties	11
2.1.3 Microprocessor System Components	12
2.1.4 Capabilities of Microprocessor System Components	13
2.1.5 Microprocessor System Summary	14
2.2 Digital Systems Design	14
2.3 Knowledge Based Expert Systems	16
2.3.1 Knowledge Representation	16
2.3.2 Productions Systems	19
2.3.3 Expert System Shells	20
2.4 Design Automation	21
2.4.1 High-Level Synthesis of Digital Systems	21
High Level Description of Digital Circuits	22
High Level Synthesis of Microprocessor Systems and HDL	23
2.4.2 Expert Systems and Artificial Intelligence for Design Automation	24
The XCON Configurer of Computer Systems	24
The DEMETER Design Environment	25
The MAPLE and PECOS Hardware Synthesis Systems	25
The KDMS Hardware/Software Synthesis System	25
The MICON Single Board Computer Designer	26
The DAME Microprocessor System Designer	26
2.5 Summary	27
Chapter 3: Interface Design Expert System Development Issues	29
3.1 Introduction	29
3.2 Data Transfer Interface Example	29
3.2.1 The MC68000 System Interface Example	30
3.2.2 The Timing Diagram of the Example Components	31
Interface of the Address Signals	33
Interface Data Signals	33
Other Control Signals	34
3.2.3 Observations about the Interface Design Example	34
3.3 Approach Used for Development of the Design Automation System	37
3.3.1 Imitating a Human Designer	37
3.3.2 Partitioning of the Interface Design System Knowledge	37

3.3.3	Abstraction of the Design Knowledge Representation.....	37
3.3.4	Design Based on Recognizable Patterns	38
3.4	Representing Components and their Behavior	39
3.4.1	Modelling Capabilities of Components.....	39
3.4.2	Modelling the Capability Protocol	40
	Synchronizing the Protocols between Components.....	40
	Overall Control of a Capability Protocol	41
3.4.3	Modelling Information Transfers	41
3.5	Representing the Interface	42
3.5.1	Partitioning the Interface	42
3.5.2	Hierarchy of the Interface Digital System.....	43
3.6	Representing the Interface Design Knowledge.....	44
3.7	Frame Representation of the Components and Interface.....	46
3.8	Summary	49
Chapter 4: Microprocessor System Component Model.....		52
4.1	Introduction.....	52
4.2	Signals.....	53
4.3	The State of a Signal	54
4.3.1	Compatible States	55
4.3.2	Representing the States of a Signals.....	56
4.4	Using Signal States to Describe Situations.....	57
4.5	State Changes in Signals.....	58
4.5.1	Transitions	58
4.5.2	Events	59
4.5.3	Detectable Events	60
4.5.4	Complementary Events	60
4.6	Modeling Time Relationships Between Events.....	61
4.6.1	The Timing Link Between Events.....	61
4.6.2	Repeated Event Sequences in Timing Diagrams.....	62
4.6.3	Properties of Timing Links.....	63
4.6.4	Timing Links Between Events.....	65
	Causal Timing Links	66
	Non-Causal Timing Links.....	67
4.6.5	Timing Links Between Complementary Events.....	68
4.6.6	Timing Link Summary	70
4.6.7	Notation Used to Represent Timing Links Between Events	71
4.7	Modeling Signal Timings	71
4.7.1	Developing the Concept of Timing Templates.....	71
4.7.2	Propagation Delay Invariance of Timing Templates	73
4.7.3	Developing Propagation Delay Invariant Timing Templates.....	75
4.7.4	The Component Model Timings.....	77
4.7.5	Two Reference Event Timings for Data Transfer.....	78
4.8	The Data Transfer Signal Timings.....	79
4.8.1	Interactive Timings and the Initiate to Terminate Time Interval	82
4.8.2	Multiple Reference Signal Timings.....	84
4.8.3	Signal Timing Summary.....	86
4.9	Modeling Information Transfer	87
4.10	Modeling the Data Transfer Capability	88
4.10.1	Organization of Data Transfer in a Microprocessor Systems.....	88
4.10.2	Classification of the Data Transfer Information Transfers	89
4.10.3	The Request Information	90

4.10.4	The Delay Information	91
Overall Asynchronous Control	91	
Overall Synchronous Control.....	92	
4.10.5	Summary of Information Transfer between Master and Slave.....	92
4.11	Conclusions.....	93
Chapter 5:	Microprocessor System Interface Model.....	94
5.1	The Interface Block	94
5.2	The Information Connection Interface Sub-Blocks.....	94
5.3	Partitioning the Info ISBs	95
5.3.1	The Timing ISBs	97
5.3.2	The State ISBs	98
5.4	Interface Sub-Block Primitive Circuits.....	100
5.4.1	Common ISBPs and their Behavior.....	101
Combinatorial ISBP	102	
D-Flip-Flop Clocked Memory ISBP	103	
Other ISBPs.....	104	
ISBP Timing Simulation	106	
5.5	Interface Representation Summary.....	107
Chapter 6:	The Interface Design Process.....	108
6.1	Introduction.....	108
6.2	Abstraction of the Interface Design Tasks	109
6.3	Overview of the Interface Block Design Terminology and Process	111
6.4	Creating the Interface Block	113
6.5	Partitioning the IB into Info ISBs	114
6.5.1	Rules Used for Connecting Information Signals of the Same Class	116
6.5.2	Rules for Generating Internal Information Ports.....	117
6.5.3	Rules Used for Utilizing Extra Information	119
6.5.4	Rules Used for Generating Missing Information	120
6.5.5	Generating the Goal Information of an Info ISB.....	121
6.6	Creating the State and Timing ISBs.....	125
6.7	Generating the Combinatorial ISBP for the State ISB	126
6.8	Designing the Timing ISB using ISBP	127
6.8.1	Overview of the Timing ISB Design Process	128
6.8.2	Choosing the ISBP to build up the Timing ISB	130
6.8.3	Timing ISBP Timing Propagation	133
D-Latch ISBP Timing Propagation	134	
Leading Edge Delay ISBP Timing Propagation	136	
Summary of Timing ISBP Timing Propagation.....	137	
6.8.4	Combinatorial ISBP Timing Propagation.....	138
Example of Strobe Input Timings for Combinatorial ISBP	139	
Example of Logic Timing Inputs Mixed With Strobe Timing Inputs	142	
Summary of Combinatorial ISBP Timing Propagation	144	
6.9	IB Timing Verification	145
6.9.1	The Connection Timing Constraint Extraction Process	146
Extracting the Timing Constraints	150	
Constraint Extraction Rules	151	
6.9.2	Choosing an Implementation Technology.....	152
6.9.3	Calculating the Initiate-Terminate Delay	153
6.9.4	Timing Constraint Evaluation and Verification	154
6.10	Generating the VHDL Code	158

6.11	Controlling the Design Process	159
6.12	Summary of the Interface Design Process and Representation	160
Chapter 7: Data Transfer Interface Design Implementation and Results		161
7.1	Component Library	161
7.1.1	Prototype Frames	162
7.1.2	Device Frames	162
7.1.3	Components Represented	164
7.1.4	Component Entry Guidelines	164
7.2	Design Rules	166
7.3	Interface Designer Output	167
7.4	Interface Design Example: 68000 to 6116	168
7.4.1	Problem Specification: 68000 to 6116	168
7.4.2	Execution: 68000 to 6116	169
7.4.3	System Schematic: 68000 to 6116	173
7.4.4	Timing Constraint Verification: 68000 to 6116	175
7.4.5	VHDL Code Output: 68000 to 6116	175
7.4.6	VHDL Simulation: 68000 to 6116	176
7.4.7	Validation of the Interface: 68000 to 6116	179
7.5	Timing Verification Failures	180
7.6	Summary of Designs	181
Chapter 8: Conclusions and Future Work		184
8.1	Conclusions	184
8.2	Future Work	188
Bibliography		191
Appendix A: Timing Templates for Modeling Data Transfer		197
A.1	Non-Interactive Timings	197
A.1.1	Strobe Timing	197
A.1.2	Latch Timing	198
A.1.3	Follows Timing	199
A.1.4	Pulse-Latch Timing	200
A.1.5	Follows-Latch Timing	200
A.1.6	Logic Timing	201
A.2	Interactive Timings	202
A.2.1	Handshake Timing	202
A.2.2	Wait Timing	203
A.2.3	Pulse Timing	205
Appendix B: The Component and Interface Frame Hierarchy		206
B.1	The Component Frames	206
B.1.1	The Capability Device Frame	206
B.1.2	A Note About Choosing the Name of a Frame	208
B.1.3	The State-Timing Specification Device Frame	209
B.1.4	The State Specification Device Frame	209
B.1.5	The Timing Specification Device Frame	210
B.1.6	The Signal Device Frame	211
B.1.7	Overview of the Component Organization	212
B.1.8	Examples of Component Frame Hierarchy	213
B.1.9	Examples of Component Frames	213
	Example of a Timing Information Frame	214
	Example of a State Information Frame	218

B.2	The Interface Frames	219
B.2.1	Frame Representation of the Interface Block	220
B.2.2	Frame Representation of an ISBP	222
Appendix C:	VHDL Code for ISBPs	224
C.1	Package Declaration for ISBPs	224
C.2	Entity and Architecture Declaration for ISBPs	225
C.2.1	2 Input AND Entity	225
C.2.2	2 Input OR Entity	225
C.2.3	2 Input XOR Entity	225
C.2.4	Inverter Entity	225
C.2.5	D-Latch Entity	226
C.2.6	D-Flip-Flop Entity	226
C.2.7	Pure Delay Entity	226
D-Flip-Flop Implementation of 50 ns Pure delay	227	
C.2.8	Leading Edge Delay Entity	227
C.2.9	Trailing Edge Delay Entity	228
C.2.10	Tri-Sate Buffer Entity	228
C.2.11	Open Collector Buffer Entity	229
Appendix D:	CRL Frames for Design Example from Section 7.4	230
D.1	CRL Frames for the Motorola MC68000 Microprocessor	230
D.1.1	CRL Frames MC68000 Body	230
D.1.2	CRL Frames MC68000 Timing (8Mhz)	234
D.2	CRL Frames for Component Instances and the Connection Request	236
Appendix E:	VHDL Code for Design Example from Section 7.4	238
E.1	VHDL ISBs for Design Example	238
E.2	VHDL IB for Design Example	250
E.3	VHDL Test Bench for Design Example	254
Appendix F:	Other Interface Design Examples	259
F.1	Interface Design Example: i8086	259
F.2	Interface Design Example: 68020	260
F.3	6809 Interface Example	264
F.4	t32020 Interface Example	266
Appendix G:	The Model Frame	268

List of Figures

FIGURE 1-1.	Data Transfer Interface Design	2
FIGURE 1-2.	Interface Design Expert System.....	4
FIGURE 2-1.	Block Diagram of a Simple Microcomputer.....	8
FIGURE 2-2.	Digital System Design Phases	15
FIGURE 2-3.	Semantic Network for John	17
FIGURE 2-4.	Structure of a Production System	19
FIGURE 2-5.	Abstraction Levels for Digital Systems	22
FIGURE 3-1.	Interface Between MC68000 CPU and MK6116 Static RAM.....	30
FIGURE 3-2.	Timing Diagram of the MC68000 Read Cycle.....	32
FIGURE 3-3.	Timing Diagram for the MK6116 CMOS Static RAM Read Cycle.....	32
FIGURE 3-4.	Example Illegal Glitch Transitions for MK6116 CMOS Static RAM Read Cycle.....	33
FIGURE 3-5.	Structure of the Interface Designer	38
FIGURE 3-6.	Information Embedded in the State of Signals and its Time Reference	41
FIGURE 3-7.	Partitioning a Digital Systems into Sub-systems	43
FIGURE 3-8.	Interface Hierarchy	44
FIGURE 3-9.	X2000 Device Frames.....	46
FIGURE 3-10.	Device and Prototype Frames	46
FIGURE 3-11.	Prototype Hierarchy	47
FIGURE 3-12.	Example Device Frames	48
FIGURE 3-13.	Component Instance Frames.....	49
FIGURE 3-14.	Interface Designer Knowledge Representation.....	50
FIGURE 4-1.	Outline of the Component Model Presentation	52
FIGURE 4-2.	Logic State Hierarchy	54
FIGURE 4-3.	Voltage Levels Associated with Sates.....	55
FIGURE 4-4.	Timing Diagram of the MC68000 Read Cycle.....	61
FIGURE 4-5.	Example of Event Time Relationship	62
FIGURE 4-6.	Repeated Event Sequence Representation	63
FIGURE 4-7.	Possible Event Relationships	65
FIGURE 4-8.	Example of the Always-Accompanied-by Link	67
FIGURE 4-9.	Example of the Accompanied-by Link.....	68
FIGURE 4-10.	Typical Data Write Operation Timing Diagram	68
FIGURE 4-11.	Typical Data Write Operation Timing Links	69
FIGURE 4-12.	Representation of Signal Timing of Non-Multiplexed Signal A3	72
FIGURE 4-13.	Propagation Delay Invariance of Timing Template (Signal is Delayed)	75
FIGURE 4-14.	Propagation Delay Invariance of Timing Template (Reference is Delayed)	76

FIGURE 4-15.	Simple Setup and Hold Time Example.....	76
FIGURE 4-16.	Updated Non Multiplexed Signal Timing Template.....	77
FIGURE 4-17.	Non-interactive Timing Example.....	78
FIGURE 4-18.	Interactive Timing Example.....	79
FIGURE 4-19.	Theoretical Timing Relations.....	79
FIGURE 4-20.	Non-Interactive Timing Templates - Part 1.....	80
FIGURE 4-21.	Non-Interactive Timing Templates - Part 2.....	81
FIGURE 4-22.	Interactive Timing Templates.....	82
FIGURE 4-23.	MC68000 Read Data Transfer.....	83
FIGURE 4-24.	Initiate to Terminate Timing Link Example	84
FIGURE 4-25.	Data Access Timing for a Typical Slave Device.....	85
FIGURE 4-26.	AND-Follows Timing	85
FIGURE 4-27.	Information Transfer Example.....	87
FIGURE 4-28.	Request Information Example	91
FIGURE 4-29.	Overall Asynchronous Control	92
FIGURE 4-30.	Overall Synchronous Control	92
FIGURE 4-31.	Information transfer between master and slave	93
FIGURE 5-1.	Interface Block (IB).....	94
FIGURE 5-2.	Information Connection Interface Sub-Blocks (ISB)	95
FIGURE 5-3.	Timing and State Conversion Order.....	96
FIGURE 5-4.	Details of Information Connection ISB	97
FIGURE 5-5.	Effect of Pure Delay and Clocked Memory Device on a Timing	98
FIGURE 5-6.	Combinatorial State	99
FIGURE 5-7.	Tri-state Buffer.....	99
FIGURE 5-8.	Interface Block Organization	100
FIGURE 5-9.	Behavior Model of Combinatorial ISBP.....	102
FIGURE 5-10.	Behavior Model of Edge Triggered D-Flip-Flop ISBP.....	103
FIGURE 5-11.	Behavior Model of D-Latch ISBP	104
FIGURE 5-12.	Behavior Model of Pure Delay ISBP.....	105
FIGURE 5-13.	Behavior Model of Leading Edge Delay Primitive.....	105
FIGURE 5-14.	Behavior Model of Trailing Edge Delay Primitive	105
FIGURE 5-15.	Behavior Model of Tri-State Buffer Primitive	106
FIGURE 5-16.	Logical Model of Open Collector Buffer Primitive.....	106
FIGURE 5-17.	Simulation of Primitives	106
FIGURE 6-1.	Interface Design Process.....	108
FIGURE 6-2.	Interface Design Task Abstraction Levels	110
FIGURE 6-3.	Design Process Overview and Terminology	112
FIGURE 6-4.	Capability Connection IB Creation.....	114
FIGURE 6-5.	Example Microprocessor / Memory Interface Info ISBs.....	115
FIGURE 6-6.	Example Extra Address Information Merge using three ISBs.....	117

FIGURE 6-7.	Strobe Input Timing Specification Goal Timings	123
FIGURE 6-8.	State and Timing ISB Creation	125
FIGURE 6-9.	State ISB Primitive Circuit Creation.....	126
FIGURE 6-10.	Timing ISBP Design	127
FIGURE 6-11.	Info ISB with Timing ISBs	128
FIGURE 6-12.	Interface Sub-Block example.....	129
FIGURE 6-13.	Example for Info ISB Timing Propagation.....	129
FIGURE 6-14.	Follows Input to Strobe Output Timing Template	131
FIGURE 6-15.	Model of D-Latch ISBP	134
FIGURE 6-16.	Timing for Latch Output if Input is Latch Timing.....	135
FIGURE 6-17.	Model of Leading-Edge Delay ISBP.....	136
FIGURE 6-18.	Logic input and Handshake Output Timing.....	137
FIGURE 6-19.	Model of Combinatorial ISBP	139
FIGURE 6-20.	Timing for Combinatorial ISBP Output for all Strobe Input Timings.....	140
FIGURE 6-21.	Overview of Input and Output Timings for Combinatorial ISBP	142
FIGURE 6-22.	Timing for Combinatorial Output for Logic and Strobe Input Timings.....	144
FIGURE 6-23.	The Interface Output to Component Connection.....	145
FIGURE 6-24.	Example Interface for an Address Signal	147
FIGURE 6-25.	Relative Timing Relationships for Example Interface.....	147
FIGURE 6-26.	Finding Timing TX of A1' relative to CE'	148
FIGURE 6-27.	Contains Interval Operator.....	150
FIGURE 6-28.	Constraint Output and Input Specification.....	151
FIGURE 6-29.	IB Constraint Extraction Rules	152
FIGURE 6-30.	Example Handshake Delay Timing of a Microprocessor	153
FIGURE 6-31.	Delay of a Signal Relative to a Reference	155
FIGURE 6-32.	Delay of a Reference Relative to a Signal	155
FIGURE 6-33.	Example of Addition of a Timing Parameter and a Propagation Delay.....	156
FIGURE 6-34.	Example of Subtraction of a Timing Parameter and a Propagation Delay.....	157
FIGURE 6-35.	Design Phases used for Contexts Limiting	159
FIGURE 7-1.	Class Network of Prototype Frames for Signal Timings	162
FIGURE 7-2.	Motorola 68000 Microprocessor Frame Network	163
FIGURE 7-3.	Interface Designer Output.....	167
FIGURE 7-4.	68000 to 6116 Design Example Specification	169
FIGURE 7-5.	The Example Interface After 8 Rules Have Fired.....	170
FIGURE 7-6.	Request Interface Information Schematic.....	171
FIGURE 7-7.	Completed Interface Design Example Frame Network	172

FIGURE 7-8.	Schematic for Interface Design Example	174
FIGURE 7-9.	68000 Design Example VHDL Simulation	177
FIGURE 7-10.	Simulation Timing Diagram States.....	178
FIGURE 7-11.	IB Signal Naming for Simulation	178
FIGURE A-1.	Strobe Timing	198
FIGURE A-2.	Latch Timing.....	198
FIGURE A-3.	Follows Timing	199
FIGURE A-4.	Pulse-Latch Timing.....	200
FIGURE A-5.	Follows-Latch Timing.....	201
FIGURE A-6.	Logic Timing.....	201
FIGURE A-7.	Logic Timing Example	202
FIGURE A-8.	Handshake Timing (Information Signal is Output)	203
FIGURE A-9.	Handshake Timing (Information Signal is Input)	203
FIGURE A-10.	Wait Timing (Information Signal is Output).....	204
FIGURE A-11.	Wait Timing (Information Signal is Input)	204
FIGURE A-12.	Pulse Timing	205
FIGURE B-1.	The MC68000 Component Device Frame	206
FIGURE B-2.	The MC68000 Capability Device Frame	207
FIGURE B-3.	State Timing Specification	209
FIGURE B-4.	State Information for Address Information Transfer	210
FIGURE B-5.	State Information for MC68000 Type Information Transfer	211
FIGURE B-6.	Example Strobe Timing Information Frame	212
FIGURE B-7.	Event Names for Strobe Timing	212
FIGURE B-8.	Example Signal Frame.....	213
FIGURE B-9.	Prototype, Device and Instance Hierarchy.....	214
FIGURE B-10.	Component Hierarchy for MC68000.....	215
FIGURE B-11.	Component Hierarchy for MK6116.....	216
FIGURE B-12.	Strobe Timing for MC68000 Address Signals.....	217
FIGURE B-13.	Interface Block Organization	220
FIGURE B-14.	Schematic Representation of Example ISBP Frame.....	223
FIGURE F-1.	i8086 System.....	259
FIGURE F-2.	i8086 Design - VHDL Simulation	261
FIGURE F-3.	68020 Design - VHDL Simulation	263
FIGURE F-4.	m6809 Design - VHDL Simulation	265
FIGURE F-5.	t32020 Design - VHDL Simulation	267
FIGURE G-1.	The Model Hierarchy.....	270
FIGURE G-2.	Prototype, Model, Device and Instance of Device frames.....	271

List of Tables

TABLE 2-1.	Semantic Network Frame for John	18
TABLE 4-1.	Compatible States	56
TABLE 4-2.	Opposite States.....	59
TABLE 4-3.	Component Timing Links	70
TABLE 4-4.	Output Specification Timings	86
TABLE 4-5.	Input Requirement Timings	86
TABLE 5-1.	VHDL Behavior Model of 2 Input AND ISBP	103
TABLE 5-2.	VHDL Behavior Model of D-Flip-Flop ISBP	104
TABLE 6-1.	Connections Rules for the Same Information Class	116
TABLE 6-2.	Internal Information Generation Rules	119
TABLE 6-3.	Extra Information Manipulation Rules	119
TABLE 6-4.	Missing Information Generation Rules.....	121
TABLE 6-5.	Internal Information ISB Goal Information.....	122
TABLE 6-6.	Goal Timings.....	124
TABLE 6-7.	Permitted Input / Output Timing Templates for Info ISB.....	130
TABLE 6-8.	Intermediate Timing Templates for Input / Output Timings of Info ISBs.....	132
TABLE 6-9.	Steps for Timing ISBP Timing Propagation	138
TABLE 6-10.	Possible Input Timing for each Output Timing Template for Combinatorial ISBP	138
TABLE 6-11.	Steps for Combinatorial ISBP Timing Propagation.....	145
TABLE 6-12.	Steps for Timing Constraint Extraction	152
TABLE 7-1.	List of Components in Component Library	165
TABLE 7-2.	Rule Design Function Summary	166
TABLE 7-3.	Example Rule for Timing Constr10int Extraction.....	166
TABLE 7-4.	Component Instances and Connection Request for Design Example	169
TABLE 7-5.	Rules fired for Request Information ISB design.....	170
TABLE 7-6.	Internal Request Generation Frame for Design Example	171
TABLE 7-7.	VHDL Request Generation Entity for Design Example.....	176
TABLE 7-8.	68000 Interface Timing Margins	180
TABLE 7-9.	Summary of Designs.....	182
TABLE B-1.	Relations Used to give the State-Timing Frames for Data Transfer Capability.....	208
TABLE B-2.	Example Frame for MC68000 Address Timing Information Frame	217
TABLE B-3.	Frame for Strobe Timing.....	218
TABLE B-4.	Example Frame for the MC68000 Type State Information	219
TABLE B-5.	Interface Block Frame.....	221
TABLE B-6.	VHDL Representation of Example Interface Block Frame	221

TABLE B-7. Combinatorial ISBP	222
TABLE B-8. VHDL Representation of Example ISBP Frame	223

Acknowledgments

I would like to thank Dr. Kin. F. Li for his help and guidance throughout the course of this work. I would also like to thank NSERC for providing financial support for this research.

Glossary

ALU	Arithmetic Logic Unit
ASCII	American Standard Code for Information Interchange
ASIC	Application Specific Integrated Circuit
CAD	Computer Aided Design
CMOS	Complementary Metal Oxide Semiconductor
CPU	Central Processing Unit
CRL	Carnegie Representation Language
CRT	Cathode Ray Tube
DAME	Design Automation of Microprocessor-based systems using an Expert system approach
DIP	Dual In-line Package
DMA	Direct Memory Access
DSP	Digital Signal Processor
EPROM	Erasable Programmable ROM
HDL	Hardware Description Language
IO	Input / Output
IB	Interface Block
ISB	Interface Sub-Block
ISBP	Interface Sub-Block Primitive
LCC	Lead-less Chip Carrier
LSI	Large Scale Integration
MSI	Medium Scale Integration
NMOS	N-Type Metal Oxide Semiconductor
omp	Order of Magnitude Propagation delay
PAL	Programmable Array Logic
PGA	Pin Grid Array
P-M-S	Program-Memory-Switch
RAM	Random Access Memory
RISC	Reduced Instruction Set Computer
ROM	Read Only Memory
SSI	Small Scale Integration
TTL	Transistor-Transistor Logic
UART	Universal Asynchronous Receiver/Transmitter

VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
VLSI	Very Large Scale Integration