



Ain Shams University

Faculty of Engineering

Computer and Systems Department

# Model Predictive Control (MPC) On Chip

A Thesis

Submitted in partial fulfillment for the requirements of

Master of Science degree in Electrical Engineering

Submitted by:

**Yasser Shoukry Sakr**

B.Sc. of Electrical Engineering

(Computer and Systems Department)

Ain Shams University, 2007.

Supervised by:

**Prof. Dr. Mohamed El-Shafey**

**Dr. Sherif Ali Hammad**

**Dr. Sahar Hagag**

Cairo 2010

## Acknowledgments

All praise be to Allah the High, “who teacheth by the pen, teacheth man that which he knew not.”, Quran[96:4, 96:5]. I say what Prophet Solomon said: “. . . O my Lord! so order me that I may be grateful for Thy favours, which thou hast bestowed on me and on my parents, and that I may work the righteousness that will please Thee: And admit me, by Thy Grace, to the ranks of Thy righteous Servants.”, Quran[27:19]

I wish to express my deep gratitude to my advisors: Dr. Sherif Hammad, Prof. Dr. Mohamed El-Shafey and Dr. Sahar Hagag for their guide and important remarks on the developed results and the written manuscript and their continuous encouragement.

I hearty thank Dr. Sherif for his suggestion of thesis topic, his invaluable support, guidance and his constructive comments. This thesis would not have been the same without him. He paved me the way to finish it and supported me in presenting my work in all the international conferences in which we have accepted papers. He has undoubtedly been the greatest intellectual influence in my life and a great friend. I will always cherish the time we spent working together.

Special thanks to Dr. Mohamed Watheq El-Kharashi for his constant support, concern, insight, dedication to research, invaluable help, and attention to detail throughout all digital design and hardware arithmetic aspects of this work. He kindly reviewed this thesis and contributed in the published papers in: IFAC 2010 and IEEE-ESL 2010. He proposed the idea of using systolic arrays in accelerating the execution of the control algorithms used and proposed the techniques of embedding processing cores into smaller-sized cores.

I would like to thank my parents for their unfailing help along my age and for their efforts during this thesis development. They offered me much advice and encouragement that was a great source of comfort.

Finally, To my beloved wife: Thank you for your patience, great support and encouragement during the most important stages of my life. Thank you for taking all responsibility of our life providing me with full comfort and concentration in my work. I now have to repay you the countless nights and weekends spent in the working on this thesis.

# Abstract

Applying a computationally intensive control algorithm inside an embedded time-safety-critical application represents a Cyber-Physical challenge. This thesis applies Generalized Predictive Control on automotive active suspension system. The main objective of any automotive active suspension system is to achieve an acceptable behavior, i.e., ride comfort, over a range of working frequencies while minimizing the exerted energy. Thesis objective is to proof the ability of using time-consuming advanced control algorithms on time-safety-critical embedded applications while satisfying tight real-time constraints.

Digital model identification is carried out for an automotive active suspension system, which is then used for numerical simulation of the process and the design of digital controller filters. Frequency and time response of the process are studied in order to emphasize on the challenges of the closed loop control over a networked and embedded control system. An experimental environment based on an automotive CAN channel is used to apply identification and real-time closed loop experiments.

Experimental results show the efficiency of the proposed controller tuning while giving a considerable minimization of the exerted energy. These results are obtained using an embedded software controller implementation, which is tuned offline and verified against a linearized model of the process. In order to tackle the nonlinearities presented in the process, controller filter should be computed online each sampling period. Software results show real-time challenges in the implementation of the online controller. These real-time implementation challenges are highlighted by profiling the software implementation.

Matrix operations consume a large portion of the overall execution time of the online computation of the controller. A hardware coprocessor based on a set of systolic arrays is proposed to meet real-time constraints. The proposed computing system is then integrated inside an ARM-based embedded platform using the state-of-the-art CAP9 technology. The proposed computation system is verified to meet real-time constraints over wide range of tuning parameters.

# Contents

<b>Table of Contents</b>	<b>III</b>
<b>List of Figures</b>	<b>IX</b>
<b>List of Tables</b>	<b>XIV</b>
<b>List of Symbols and Abbreviations</b>	<b>XVI</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	3
1.2 Problem Statement . . . . .	4
1.3 Thesis outline . . . . .	6
<b>2 Model Predictive Control</b>	<b>9</b>
2.1 MPC Strategy . . . . .	10
2.2 Types of MPC controllers . . . . .	13

2.2.1	Simple Model MPC . . . . .	13
2.2.2	Adaptive MPC . . . . .	13
2.2.3	Multivariable MPC . . . . .	14
2.2.4	Robust MPC . . . . .	14
2.2.5	Non-Linear MPC . . . . .	15
2.3	Generalized Predictive Controller . . . . .	16
2.3.1	J-Step Ahead Predictor . . . . .	18
	Step 1: Solving Diophantine Equation . . . . .	18
	Step 2: Forming Predictor Equation . . . . .	20
	Step 3: Forming Predictor Matrix . . . . .	23
2.3.2	Cost Function Optimizer . . . . .	25
2.3.3	Equivalent R-S-T Structure . . . . .	28
2.4	MPC on Chip . . . . .	29
2.4.1	MPC on Chip; Industrial Motivation . . . . .	29
2.4.2	MPC on Chip; Bibliography . . . . .	30
2.5	Conclusion . . . . .	31
<b>3</b>	<b>Automotive Active Suspension System</b>	<b>33</b>
3.1	Types Of Suspension Systems . . . . .	34
3.1.1	Passive Suspension . . . . .	34
3.1.2	Semi-Active Suspension . . . . .	35
3.1.3	Active Suspension . . . . .	35

3.2	Active Suspension System Model . . . . .	36
3.2.1	Quarter Car Model . . . . .	36
3.2.2	Hydraulic Actuator . . . . .	39
3.3	Complete Model . . . . .	42
3.4	System Time Response . . . . .	44
3.4.1	Effect of Actuator Voltage Input . . . . .	44
3.4.2	Effect of Road Profile Disturbance . . . . .	48
3.5	Identifying System Digital Model . . . . .	48
3.5.1	Choosing Sampling Frequency . . . . .	48
3.5.2	Deriving Digital Model $H_v(z)$ . . . . .	51
	Over Sampling . . . . .	52
	Anti-Aliasing Filter . . . . .	52
	Choosing Model Structure . . . . .	52
	Excitation Signal . . . . .	53
	Model Identification algorithm and Model Vali- dation Criteria . . . . .	54
3.5.3	Deriving Digital Model $H_r(z)$ . . . . .	55
3.6	Model Analysis . . . . .	57
3.6.1	Analysis of $H_v(z)$ . . . . .	57
3.6.2	Analysis of $H_r(z)$ . . . . .	60
3.7	Conclusion . . . . .	63



<b>4</b>	<b>GPC Design and Tuning</b>	<b>65</b>
4.1	Control Strategies . . . . .	66
4.1.1	Linear Control Strategies . . . . .	66
4.1.2	Non-Linear Control Strategies . . . . .	68
4.1.3	Intelligent Control Strategies . . . . .	73
4.2	NECS Based Verification . . . . .	75
4.2.1	Active Suspension over NECS . . . . .	75
4.2.2	Test Bench Road Track . . . . .	78
4.2.3	Performance Evaluation Criteria . . . . .	78
4.2.4	Open Loop Response of the Test Bench . . . . .	80
4.3	GPC Controller Design and Tuning . . . . .	80
4.3.1	Internal Model Principle . . . . .	83
	Experiment 1: Prediction Horizon = 0.8 sec, Control Horizon = 0.05 sec . . . . .	83
	Experiment 2: Prediction Horizon = 0.8, Con- trol Horizon = 0.05 sec , Applying In- ternal Model Principle . . . . .	83
4.3.2	Effect Of Prediction and Control Horizon . . . . .	85
	Experiment 2(revisited): Prediction Horizon = 0.8 sec, Control Horizon = 0.05 sec . . .	85

Experiment 3: Prediction Horizon = 0.8 sec,	
Control Horizon = 0.1 sec . . . . .	86
Experiment 4: Prediction Horizon = 1.0 sec,	
Control Horizon = 0.1 sec . . . . .	86
Experiment 5: Prediction Horizon = 3.5 sec,	
Control Horizon = 1.2 sec . . . . .	86
4.4 Conclusion . . . . .	93
<b>5 GPC Co-Processor</b>	<b>95</b>
5.1 Introduction . . . . .	95
5.2 Software Implementation . . . . .	96
5.3 Matrix Multiplication Co-processor . . . . .	101
5.3.1 Multiplication Systolic Array Example . . . . .	101
5.3.2 Hardware Implementation . . . . .	109
5.3.3 Matrix Multiplication Co-processor Performance	112
5.4 Matrix Inversion Co-processor . . . . .	112
5.4.1 Hardware Implementation . . . . .	113
5.4.2 Matrix Inversion Co-processor Performance . . .	122
5.5 Overall Co-processor Performance . . . . .	122
5.6 Conclusion . . . . .	123
<b>6 Implementation Case Study : CAP9 Technology</b>	<b>125</b>

6.1	CAP9 Technology . . . . .	125
6.2	Overall Architecture . . . . .	126
6.3	Matrix Co-Processor Implementation . . . . .	128
6.4	Control unit . . . . .	131
6.5	Profiling Results . . . . .	133
6.6	Conclusion . . . . .	135
<b>7</b>	<b>Conclusion</b>	<b>137</b>
7.1	Summary . . . . .	137
7.2	Future work . . . . .	139
<b>A</b>	<b>Matlab Model for GPC</b>	<b>141</b>
<b>B</b>	<b>Publications</b>	<b>149</b>
B.1	International Conferences . . . . .	149
B.2	Peer-Reviewed Journals . . . . .	150
	<b>References</b>	<b>151</b>

# List of Figures

1.1	MPC control scheme. . . . .	2
2.1	MPC Strategy . . . . .	12
2.2	Digital controller canonical structure . . . . .	29
3.1	Quarter Car Model. . . . .	38
3.2	Physical Schematic for Hydraulic Actuator. . . . .	40
3.3	Quarter Car Model. . . . .	43
3.4	Constant road profile, variable voltage input experi- ment configuration . . . . .	45
3.5	Effect of voltage input varying from 1-10 volt . . . . .	46
3.6	Effect of voltage input varying from 20-100 volt . . . . .	46
3.7	Effect of voltage input varying from 100-250 volt . . . . .	47
3.8	Constant road profile, variable voltage input experi- ment configuration . . . . .	49

3.9	Effect of step road profiles . . . . .	49
3.10	Frequency Response of $H_r(s)$ . . . . .	51
3.11	Anti Aliasing Filter . . . . .	52
3.12	Overall system model. . . . .	53
3.13	System Output . . . . .	54
3.14	Whiteness test results . . . . .	56
3.15	Difference between System Output and Model Output	56
3.16	Frequency response of the identified $B(z)/A(z)$ on a log scale . . . . .	58
3.17	Frequency response of the identified $B(z)/A(z)$ on a linear scale . . . . .	59
3.18	Poles (blue) and zeros (red) of the identified $B(z)/A(z)$ model . . . . .	59
3.19	Frequency response of the identified $B(z)/A(z)$ on a log scale . . . . .	60
3.20	Frequency response of the identified $B(z)/A(z)$ on a linear scale . . . . .	61
3.21	Poles (blue) and zeros (red) of the identified $C(z)/A(z)$ model . . . . .	61
3.22	Frequency response of the identified $H_r(z)$ on a log scale	62

3.23	Frequency response of the identified $H_r(z)$ on a linear scale . . . . .	62
3.24	Poles (blue) and zeros (red) of the identified $H_r(z)$ model	63
4.1	Control Loop over NECS . . . . .	76
4.2	Active Suspension Logical Control Loop . . . . .	77
4.3	Verification environment used to tune controller parameters. . . . .	77
4.4	Test Road . . . . .	79
4.5	Open Loop Results . . . . .	81
4.6	Enhancing the disturbance rejection using internal model principle . . . . .	84
4.7	Results of Experiment 1, $h_p = 0.8, h_c = 0.05$ sec . . . .	88
4.8	Results of Experiment 2, $h_p = 0.8, h_c = 0.05$ sec . . . .	89
4.9	Results of Experiment 3, $h_p = 0.8, h_c = 0.1$ sec . . . .	90
4.10	Results of Experiment 4, $h_p = 1.0, h_c = 0.1$ sec . . . .	91
4.11	Results of Experiment 5, $h_p = 3.5, h_c = 1.2$ sec . . . .	92
5.1	GPC Steps . . . . .	97
5.2	Hardware Schematic of a 4x4 multiplication systolic array	102
5.3	Hardware Schematic of the multiplication cell . . . . .	103
5.4	Systolic Array Multiplication Example: Clock tick = 1	104

5.5	Systolic Array Multiplication Example: Clock tick = 2	105
5.6	Systolic Array Multiplication Example: Clock tick = 3	105
5.7	Systolic Array Multiplication Example: Clock tick = 4	106
5.8	Systolic Array Multiplication Example: Clock tick = 5	106
5.9	Systolic Array Multiplication Example: Clock tick = 6	107
5.10	Systolic Array Multiplication Example: Clock tick = 7	107
5.11	Systolic Array Multiplication Example: Clock tick = 8	108
5.12	Systolic Array Multiplication Example: Clock tick = 9	108
5.13	Dataflow along small-sized systolic array for multi- plication of two 6x6 matrices (first 16 cycles) . . . . .	110
5.14	Location of the small-sized systolic array while multi- plying two 6x6 matrices (first 16 cycles) . . . . .	111
5.15	Boundary Cell Schematic . . . . .	116
5.16	Internal Cell Schematic . . . . .	117
5.17	Systolic array for matrix inversion using QRD . . . . .	118
5.18	Systolic array for matrix inversion using LQD . . . . .	119
5.19	Dataflow along small-sized systolic array for inversion of a 4x4 matrix (first 25 cycles) . . . . .	120
5.20	Location of the small-sized systolic array while inverting a 4x4 matrix (first 8 cycles) . . . . .	121
6.1	Overall architecture of the proposed embedded system.	127