FAST-TRANSIENT CURRENT CONTROL STRATEGY
AND OTHER ISSUES FOR VECTOR CONTROLLED AC DRIVES

DISSERTATION

Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the
Graduate School of The Ohio State University

By
Mongkol Konghirun, M.S.

The Ohio State University
2003

Dissertation Committee:

Professor Longya Xu, Adviser
Professor Vadim I. Utkin
Professor Stephen A. Sebo

Approved by

Department of Electrical Engineering
The sensorless vector controlled drive system of AC motors has been examined by several researchers for over the past fifteen years. Most of this research has been dedicated to the design of flux and speed estimations using advanced control theories such as Sliding Mode Control, Model Reference Adaptive System, and Kalman filter. Only a few publications have focused on the current control strategies for the vector controlled drive of AC motors. Typically, the regular PI controller is chosen because of its ease of implementation. However, under a limited DC-bus voltage, the AC motor drives with PI controller cannot produce the fast current responses with a sudden change of speed reference or load torque, because the applied voltage becomes saturated during the occurrence of a current overshoot. Although the full DC-bus voltage is applied, this phenomenon can also happen when the machine is operated at very high speeds.

This dissertation includes: 1) an investigation and implementation of the generic flux and speed estimations that are applicable for both induction and permanent-magnet synchronous motors; 2) an analysis of the quantization errors inherent in a digital controller with fixed-point digital implementation; and 3) a proposed time-optimal control technique for fast transient current response in the sensorless vector controlled drive of AC motors. Based on the Pontryagin’s maximum principle, it is guaranteed that
the dq-axis current transition from one point to another point is within the minimum-time under a limited DC-bus voltage. Both computer simulation and experimental investigations have been carried out to substantiate the proposed technique. The overall system is implemented in a 32-bit fixed-point digital signal processor based controller (i.e., eZdsp2812 board). Based on the results, the proposed time-optimal controller can achieve the fast current responses without the current overshoots and voltage saturations under a constrained DC-bus voltage.
Dedicated to my parents
ACKNOWLEDGMENTS

I wish to sincerely thank my adviser, Professor Longya Xu, for intellectual support, encouragement, and enthusiasm which made this dissertation research possible.

I also would like to thank Professor Vadim I. Utkin, and Professor Stephen A. Sebo for reviewing the dissertation and being on the examination committee. I wish to thank Professor Donald G. Kasten as well for being on the Candidacy and Master examination committee. I really appreciate their valuable time.

Next, I thank all past and current colleagues in the Power Electronics and Electric Machines Group at The Ohio State University, especially to Dr. Li Zhen, Dr. Changjiang Wang, Dr. Minghua Fu, Dr. Chuanyang Wang, Dr. Shengming Li, Mr. Jingchuan Li, Ms. Jingbo Liu, Mr. Jiangang Hu, Mr. Song Chi, and Mr. Mihai Comanescu.

Finally, I would like to specially thank Duongruitai Nicomrat for encouragements throughout my graduate study. I also would like to express my deepest appreciation to my parents.
VITA

October, 1973 .......... Born – Bangkok, THAILAND

March, 1995 .......... B.Eng., Electrical Engineering,  
                 King Mongkut’s Institute of Technology Thonburi, THAILAND

March, 1999 .......... M.S., Electrical Engineering, The Ohio State University, USA

1997-present .......... Graduate Student, The Ohio State University, USA

PUBLICATIONS

Research Publications

M. Konghirun, L. Xu, and D. Figoli, “Implementation of Generic Sensorless Direct Field  
Oriented Control of AC Motors on a Low-Cost, Fixed-Point Digital Signal Processor”,  
IEEE 17th Applied Power Electronics Conference and Exposition, Vol. 1, pp. 304-309,  
2002.

A. Tessarolo, M. Konghirun, and O. Monnier, “IQmath: A New Approach For Porting  
Floating-Point Algorithms To Fixed-Point DSPs”, Embedded Systems Conference,  
Nuremberg, Germany, 2002.
FIELDS OF STUDY

Major Field: Electrical Engineering

Studies in:

Power Electronics and Electrical Machine Control  Prof. Longya Xu
Control Theory  Prof. Vadim I. Utkin
# TABLE OF CONTENTS

Abstract ............................................................................................................................... ii

Dedication .......................................................................................................................... iv

Acknowledgments ............................................................................................................... v

Vita ...................................................................................................................................... vi

List of Tables .................................................................................................................... xi

List of Figures ................................................................................................................... xii

Chapters:

1. Introduction .................................................................................................................... 1
   1.1 Background .............................................................................................................. 1
   1.2 Problem Statement ................................................................................................. 3
   1.3 Research Objectives and Organizations .................................................................. 3

2. Sensorless Direct Field Oriented Control (DFOC) of AC Motor .............................. 5
   2.1 Introduction ............................................................................................................. 5
   2.2 Principles of Direct Field Oriented Control (DFOC) of AC Motor ....................... 7
      2.2.1 Induction motor ............................................................................................... 8
      2.2.2 Permanent-magnet synchronous motor ......................................................... 9
   2.3 Proposed Flux and Speed Estimations of AC Motor ............................................. 9
   2.4 Evaluation of Flux and Speed Estimations ........................................................... 14

3. Issues on Fixed-Point DSP Based Control and Drive System Implementation ........... 20
   3.1 Introduction ............................................................................................................ 20
   3.2 Hardware and Software Aspects ............................................................................ 21
      3.2.1 Laboratory setup ............................................................................................. 21
      3.2.2 32-bit fixed-point DSP based controller (TMS320x28x series) ..................... 22
      3.2.3 The flexible three-phase voltage source inverter (DMC1500 board) ............ 23
      3.2.4 Low-cost technique for line current and voltage measurements ................. 23
      3.2.5 Easy fixed-point “C” programming by “IQmath” approach ......................... 25
   3.3 Brief Overview of Digital Control System ............................................................. 27
   3.4 Fixed-point Versus Floating-point ......................................................................... 30
B. Solution of Current Dynamics in Vector Controlled Drive of AC Motors..............111

Bibliography.....................................................................................................................113
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Illustration of “IQmath” approach comparing with traditional fixed-point and floating-point approaches in C language</td>
<td>26</td>
</tr>
<tr>
<td>3.2</td>
<td>Smallest, least negative, largest positive numbers, and dynamic range among 16-bit fixed-point, 32-bit fixed-point, and 32-bit floating-point representation</td>
<td>33</td>
</tr>
<tr>
<td>3.3</td>
<td>Theoretical and realized discrete poles with time constants $\tau_1 = 10$ and $\tau_2 = 0.01$ and different sampling periods when using 16-bit and 32-bit word-lengths</td>
<td>39</td>
</tr>
<tr>
<td>4.1</td>
<td>Step response characteristics of dq-axis current control</td>
<td>66</td>
</tr>
<tr>
<td>4.2</td>
<td>Effects of increasing PI gains to the transient responses</td>
<td>70</td>
</tr>
</tbody>
</table>
**LIST OF FIGURES**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Stator flux estimator based on improved back EMF integration</td>
<td>11</td>
</tr>
<tr>
<td>2.2</td>
<td>A dq-axis current control loop system used for stator flux and rotor speed evaluation</td>
<td>14</td>
</tr>
<tr>
<td>2.3</td>
<td>Experimental result - back EMF ($e_{ds}^r$), stator flux ($\lambda_{ds}^r$), and stator flux angle ($\theta_{ls}^r$) from estimator of the IM at $f = 5$ Hz</td>
<td>15</td>
</tr>
<tr>
<td>2.4</td>
<td>Experimental result - back EMF ($e_{ds}^r$), stator flux ($\lambda_{ds}^r$), and stator flux angle ($\theta_{ls}^r$) from estimator of the PMSM at $f = 5$ Hz</td>
<td>15</td>
</tr>
<tr>
<td>2.5</td>
<td>Experimental result - back EMF ($e_{ds}^r$), stator flux ($\lambda_{ds}^r$), and stator flux angle ($\theta_{ls}^r$) from estimator of the IM at $f = 60$ Hz</td>
<td>16</td>
</tr>
<tr>
<td>2.6</td>
<td>Experimental result - back EMF ($e_{ds}^r$), stator flux ($\lambda_{ds}^r$), and stator flux angle ($\theta_{ls}^r$) from estimator of the PMSM at $f = 125$ Hz</td>
<td>16</td>
</tr>
<tr>
<td>2.7</td>
<td>Experimental result – synchronous ($\omega_r$), estimated rotor speed ($\hat{\omega}_r$), and measured rotor speed ($\omega_r$) of the IM at $\omega_e = 0.133 – 0.4$ per unit (base speed = 2250 rpm)</td>
<td>17</td>
</tr>
<tr>
<td>2.8</td>
<td>Experimental result – synchronous ($\omega_e$), estimated rotor speed ($\hat{\omega}_r$), and measured rotor speed ($\omega_r$) of the PMSM at $\omega_e = 0.133 – 0.4$ per unit (base speed = 7500 rpm)</td>
<td>18</td>
</tr>
<tr>
<td>2.9</td>
<td>Experimental result - synchronous speed ($\omega_e$), estimated synchronous speed ($\hat{\omega}_e$), and quadrature detector ($\Delta e$) of the IM at $\omega_e = 0.2 – 0.8$ per unit (base speed = 1800 rpm)</td>
<td>19</td>
</tr>
<tr>
<td>2.10</td>
<td>Experimental result - synchronous speed ($\omega_e$), estimated synchronous speed ($\hat{\omega}_e$), and quadrature detector ($\Delta e$) of the PMSM at $\omega_e = 0.2 – 0.8$ per unit (base speed = 6000 rpm)</td>
<td>19</td>
</tr>
</tbody>
</table>
3.1 Complete overall experiment setup ...........................................................................22
3.2 Viewing resistor based line current measurement ......................................................24
3.3 A typical digital motor control drive system for a three-phase AC motor ...............28
3.4 The quantized signal caused by an ideal A-D converter...........................................29
3.5 The quantization error caused by the finite word-length. (a) plot of variable versus truncated value, (b) plot of error due to truncation, (c) plot of variable versus rounded value, (d) plot of error due to round-off......................................................30
3.6 Formats of 16-bit fixed-point, 32-bit fixed-point, and 32-bit floating-point number representation............................................................................................................32
3.7 Radix point and its bit weights of 16-bit fixed-point and 32-bit fixed-point number representation............................................................................................................32
3.8 Input, output, and output error between floating-point and fixed-point using truncated multiplication in digital implementation of first-order low-pass filter ....41
3.9 Input, output, and output error between floating-point and fixed-point using rounded multiplication in digital implementation of first-order low-pass filter ......42
3.10 A negative feedback loop used for illustration of limit cycle...................................43
3.11 Outputs of the negative feedback loop for floating-point and Q15 fixed-point versions (limit cycle appeared).................................................................43
3.12 Outputs of the negative feedback loop for floating-point and Q24 fixed-point versions (limit cycle nearly disappeared)..................................................................44
3.13 Overall system of flux estimator of IM.....................................................................45
3.14 Simulated sensorless DFOC system of induction motor ..........................................48
3.15 Simulation result – estimated speed ( \( \hat{\omega}_r \) ) and q-axis reference current (\( i^*_q \)) using step speed reference of 0.5 pu among floating-point, 32-bit fixed-point, and 16-bit fixed-point formats............................................................................................................49
3.16 Simulation result – actual (top trace) and estimated rotor flux angles among floating-point (2\(^{nd}\) trace), 32-bit fixed-point (3\(^{rd}\) trace), and 16-bit fixed-point (4\(^{th}\) trace) formats ............................................................................................................50
3.17 Simulation result – estimated d-axis rotor flux produced by current model between 32-bit fixed-point (top trace) and 16-bit fixed-point (bottom trace) formats........51
3.18 Simulation result – estimated d-axis rotor flux produced by voltage model between
32-bit fixed-point (top trace) and 16-bit fixed-point (bottom trace) formats............51

3.19 Simulation result – floating-point format: dq-axis currents/voltages, and estimated
speed (\(\hat{\omega}_r\)) using step speed reference of ±0.5 pu........................................52

3.20 Simulation result – 32-bit fixed-point format: dq-axis currents/voltages, and
estimated speed (\(\hat{\omega}_r\)) using step speed reference of ±0.5 pu.............................53

3.21 Simulation result – 16-bit fixed-point format: dq-axis currents/voltages, and
estimated speed (\(\hat{\omega}_r\)) using step speed reference of ±0.5 pu.............................53

3.22 Real implementation of a sensorless DFOC system of induction motor ...............55

3.23 Experimental result – reference speed (\(\omega^*_r\)), d-axis measured current (\(i^*_{ds}\)), and d-axis
reference voltage (\(v^*_{ds}\)) using step speed reference of ±0.5 pu among (a) 16-bit
fixed-point, and (b) 32-bit fixed-point formats.........................................................55

3.24 Experimental result - q-axis reference current (\(i^*_{qs}\)), q-axis measured current (\(i^*_{qs}\)),
and q-axis reference voltage (\(v^*_{qs}\)) using step speed reference of ±0.5 pu among (a)
16-bit fixed-point, and (b) 32-bit fixed-point formats ...........................................55

3.25 Experimental result - reference speed (\(\omega^*_r\)), estimated speed (\(\hat{\omega}_r\)), and q-axis
reference current (\(i^*_{qs}\)) using step speed reference of ±0.5 pu among (a) 16-bit fixed-
point, and (b) 32-bit fixed-point formats .................................................................56

4.1 Two current control loops of IM (a) q-axis (b) d-axis ........................................63

4.2 Two current control loops of PMSM (a) q-axis (b) d-axis ...................................63

4.3 Approximate two current control loops of IM when \(\frac{1}{\alpha} >> \frac{1}{\gamma}\)
(a) q-axis (b) d-axis .................................................................................................64

4.4 Conventional SISO feedback loop system...........................................................64

4.5 Damping ratio, \(\zeta\), for different \(K_p\) and \(T_i\).........................................................66

4.6 Natural angular velocity, \(\omega_n\), for different \(K_p\) and \(T_i\) .................................67

4.7 Step responses of dq-axis current control for different \(K_p\) when \(T_i = 5e-04\) sec ....67

4.8 Step responses of dq-axis current control for different \(T_i\) when \(K_p = 1\)............68
4.9 Effects of PI output saturation slowing down the responses ........................................69
4.10 Step responses with a responding time be around motor time constant ...............70
4.11 Simulation result - dq-axis current responses of IM using PI controllers with different $K_p$ and $T_i$ ..................................................................................................................72
4.12 Simulation result - dq-axis current responses of PMSM using PI controllers with different $K_p$ and $T_i$ ..................................................................................................................73
4.13 Overall current control loop system using PI controllers .......................................74
4.14 Experimental result - q-axis current responses of IM using PI controllers with different $K_p$ and $T_i$ ..................................................................................................................75
4.15 Experimental result - d-axis current responses of IM using PI controllers with different $K_p$ and $T_i$ ..................................................................................................................76
4.16 Experimental result - q-axis current responses of PMSM using PI controllers with different $K_p$ and $T_i$ ..................................................................................................................77
4.17 Experimental result - d-axis current responses of PMSM using PI controllers with different $K_p$ and $T_i$ ..................................................................................................................78
4.18 State trajectories with different initial conditions when applying the optimal controls (resistance neglected) ..................................................................................83
4.19 Optimal control diagram when resistance is neglected ........................................84
4.20 State trajectories with different initial conditions when applying the optimal controls (resistance included) ..................................................................................85
4.21 Optimal control diagram when resistance is included ........................................86
4.22 Overall current control loop of sensorless vector controlled system using either PI controllers or time-optimal controllers .................................................................89
4.23 Simulation result - dq-axis current and voltage responses using PI controllers ......90
4.24 Simulation result - dq-axis current and voltage responses using time-optimal controller .........................................................................................................................90
4.25 Overall speed control loop system using PI controllers ........................................92
4.26 Simulation result – speed and dq-axis current responses using PI controllers without voltage saturation (full DC-bus voltage) .........................................................93
4.27 Simulation result – speed and dq-axis current responses using PI controllers with voltage saturation (insufficient DC-bus voltage).................................................................93

4.28 Simulation result – speed and dq-axis current responses using time-optimal controller without voltage saturation ..............................................................................................94

4.29 Experimental result - dq-axis current responses of IM (a)-(b) using PI controllers (left column), (c)-(d) using time-optimal controller (right column) .................................95

4.30 Experimental result - dq-axis current responses of PMSM (a)-(b) using PI controller (left column), (c)-(d) using time-optimal controller (right column) .................................96

4.31 Experimental result – speed and dq-axis current responses of IM using PI controllers, (a)-(c) full DC-bus voltage (left column), (d)-(f) insufficient DC-bus voltage (right column) ..........................................................................................98

4.32 Experimental result – speed and dq-axis current responses of IM with insufficient DC-bus voltage, (a)-(c) PI controllers (left column), (d)-(f) time-optimal controller (right column) ..............................................................................................99

4.33 Experimental result – speed and dq-axis current responses of PMSM using PI controllers, (a)-(c) full DC-bus voltage (left column), (d)-(f) insufficient DC-bus voltage (right column) .............................................................................................100

4.34 Experimental result – speed and dq-axis current responses of PMSM with insufficient DC-bus voltage, (a)-(c) PI controllers (left column), (d)-(f) time-optimal controller (right column) ..............................................................................................101
NOMENCLATURE

e_{ds}^s \quad \text{d-axis back EMF in the stationary reference frame (ω=0), V}

e_{qs}^s \quad \text{q-axis back EMF in the stationary reference frame (ω=0), V}

f_c \quad \text{cut-off frequency, Hz}

i_{as} \quad \text{phase-a stator current, A}

i_{bs} \quad \text{phase-b stator current, A}

i_{ds}^e \quad \text{d-axis stator current in the synchronously rotating reference frame (ω=ω_e), A}

i_{qs}^e \quad \text{q-axis stator current in the synchronously rotating reference frame (ω=ω_e), A}

i_{ds}^s \quad \text{d-axis stator current in the stationary reference frame (ω=0), A}

i_{qs}^s \quad \text{q-axis stator current in the stationary reference frame (ω=0), A}

K_i \quad \text{integral gain of PI controller}

K_p \quad \text{proportional gain of PI controller}

L_m \quad \text{magnetizing inductance, H}

L_r \quad \text{rotor inductance, H}
$L_s$  stator inductance, H

$p$  number of poles

$PO$  percent overshoot, %

$R_r$  rotor resistance, $\Omega$

$R_s$  stator resistance, $\Omega$

$t_f$  final time, sec

$t_p$  peak time, sec

$t_s$  settling time to 2% error, sec

$T$  sampling period, sec

$T_e$  electromagnetic torque, N.m

$T_i$  integral time of PI controller, sec

$v_{ds}^e$  d-axis stator voltage in the synchronously rotating reference frame ($\omega=\omega_e$), V

$v_{qs}^e$  q-axis stator voltage in the synchronously rotating reference frame ($\omega=\omega_e$), V

$v_{ds}^s$  d-axis stator voltage in the stationary reference frame ($\omega=0$), V

$v_{qs}^s$  q-axis stator voltage in the stationary reference frame ($\omega=0$), V

$\psi$  co-state variable

$\zeta$  damping ratio

$\tau$  time constant, sec
\( \tau_c \)    time constant of low-pass filter, sec

\( \tau_r \)    rotor time constant of IM, sec

\( \gamma \)    angle between back EMF and stator flux, rad

\( \lambda_{\text{comp}} \)    compensated stator flux, Wb

\( \lambda_{ds,1}, \lambda_{qs,1} \)    dq-axis stator flux computed by approximating integration of back EMF, Wb

\( \lambda_{ds,2}, \lambda_{qs,2} \)    dq-axis stator flux computed by feedback compensated stator flux, Wb

\( \lambda_{dr} \)    d-axis rotor flux in the stationary reference frame (\( \omega=0 \)), Wb

\( \lambda_{qr} \)    q-axis rotor flux in the stationary reference frame (\( \omega=0 \)), Wb

\( \lambda_{ds} \)    d-axis stator flux in the stationary reference frame (\( \omega=0 \)), Wb

\( \lambda_{qs} \)    q-axis stator flux in the stationary reference frame (\( \omega=0 \)), Wb

\( \lambda_{m} \)    constant permanent-magnet flux, Wb

\( \lambda_{dr}^e \)    d-axis rotor flux in the synchronously rotating reference frame (\( \omega=\omega_c \)), Wb

\( \lambda_{qr}^e \)    q-axis rotor flux in the synchronously rotating reference frame (\( \omega=\omega_c \)), Wb

\( \lambda_{ds}^e \)    d-axis stator flux in the synchronously rotating reference frame (\( \omega=\omega_c \)), Wb

\( \lambda_{qs}^e \)    q-axis stator flux in the synchronously rotating reference frame (\( \omega=\omega_c \)), Wb

\( \omega_c \)    cut-off angular velocity, rad/sec

\( \omega_e \)    electrically synchronous angular velocity, rad/sec
\( \hat{\omega}_e \) estimated electrically synchronous angular velocity, rad/sec

\( \omega_n \) natural angular velocity, rad/sec

\( \omega_r \) electrically rotor angular velocity, rad/sec

\( \hat{\omega}_r \) estimated electrically rotor angular velocity, rad/sec

\( \omega_s \) electrically slip angular velocity, rad/sec

\( \theta_e \) electrically rotor flux angle, rad

\( \theta_{ss} \) electrically stator flux angle, rad

\( \Delta e \) quadrature detector, V
ABBREVIATIONS

AC       Alternating Current
ADC      Analog-to-Digital Converter
CCS      Code Composer Studio
DAC      Digital-to-Analog Converter
DC       Direct Current
DFOC     Direct Field Oriented Control
DSP      Digital Signal Processor
EKF      Extended Kalman Filter
EMF      Electromotive Force
FOC      Field Oriented Control
IM       Induction Motor
LSB      Least Significant Bit
MRAS     Model Reference Adaptive System
MSB      Most Significant Bit
PMSM     Permanent-Magnet Synchronous Motor
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PWM</td>
<td>Pulse Width Modulation</td>
</tr>
<tr>
<td>SISO</td>
<td>Single-Input-Single-Output</td>
</tr>
<tr>
<td>TI</td>
<td>Texas Instruments, Inc.</td>
</tr>
<tr>
<td>VCFM</td>
<td>Voltage and Current Flux Model</td>
</tr>
<tr>
<td>ZOH</td>
<td>Zero-Order-Hold</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

1.1 Background

It is well known that the flux vector control is one of most popular schemes in the area of high performance drives of three-phase AC motors. The main idea of this scheme is to imitate the operation of DC motors in which the motor flux and torque are independently controlled by field and armature currents, respectively. In vector control of AC motors, there are two currents referred to as the two stator currents in d-axis and q-axis reference frames. These are orthogonal to each other and synchronously rotating aligned with the rotor flux angle. Consequently, three-phase AC stator currents are necessary to be transformed to the two-phase dq-axis currents through Park transformation using the rotor flux angle. After transformation occurs, the dq-axis currents become DC components making the controllers easier to regulate the currents. Unfortunately, if using the incorrect rotor flux angle, then the vector control scheme fails. As a result, the flux and torque controls are not truly independent of each other. Therefore, identifying the rotor flux angle is most crucial for any vector controlled drive system.
In addition, the overall performance of the system can be affected not only by how accurate the rotor flux angle is for decoupling control of torque and rotor flux, but also how well the controllers regulate the dq-axis stator currents. Even if the rotor flux angle is correct, but the poor system responses can happen when the current controllers are not well designed and/or perform badly. Typically, the currents in a vector controlled drive system of a three-phase AC motor are regulated by PI controllers because of their ease of implementation. However, the normal PI controller cannot satisfactorily perform during transient periods. The overshoot of the current responses and voltage saturation typically occur at such moments when the transient current responses become sluggish. In practice, the voltage saturation is quite evident when the DC-bus voltage is insufficient or an AC machine runs at very high speeds. As a result, the fast transient current responses are not possible by merely using PI controllers.

In this research, both rotor flux angle estimation and the current controller have been examined as well as some practical issues regarding fixed-point digital implementation. In particular, the fast transient current response technique as well as the generic flux and speed estimations that can be applicable for both IM and PMSM were investigated and implemented. The accuracy of the flux estimation itself was verified and a time-optimal current controller was developed for fast transient current responses which could not previously be realized by PI controllers.
1.2 Problem Statement

The dq-axis current control loops in the vector controlled drive of an AC motor plays an important role in determining overall system performance. During a sudden change of speed reference or machine load in the high speed range, a PI controller usually produces the voltage saturation during transient periods because of a limited DC-bus voltage. To prevent the voltage saturations generated by the PI controller, the proportional and integral gains of the PI controller have to be adjusted in such a fashion that the current responses are slowed down. Thus, a normal PI controller cannot provide the fast current responses during high speed operation where DC-bus voltage becomes insufficient. This research was aimed at developing a time-optimal current control technique in a vector controlled drive of AC motors to allow the dq-axis current responses to be controlled as fastest as possible under limited DC-bus voltage.

1.3 Research Objectives and Organizations

The purpose of this research is presented as follows: 1) the principle of a vector controlled drive of AC motor is discussed, 2) the generic flux and speed estimations that are applicable for both IM and PMSM are developed and implemented, 3) the practical issues relating to fixed-point implementation are addressed, and 4) the time-optimal controller for fast transient current response in a vector controlled drive of AC motor is derived. Both computer simulations and experimental results comparing PI controllers
with the proposed time-optimal controller are included. All algorithms are implemented on a 32-bit fixed-point DSP based controller.

This dissertation is organized as follows.

Chapter 2 presents the principle of Direct Field Oriented Control (DFOC) of AC motors. The generic flux and speed estimations are discussed and explained along with their experimental results.

Chapter 3 describes the issues on fixed-point DSP based control and drive system implementation such as hardware and software aspects, overview of digital control system, floating-point and fixed-point representation, quantization effects, and a case study of sensorless DFOC systems of induction motors.

Chapter 4 proposes a fast transient current control strategy of FOC systems for AC motors. A review of conventional PI controllers is presented. Both simulation and experimental results using PI controllers and the proposed time-optimal controller are compared in current and speed control loops.

Chapter 5 summarizes and concludes this dissertation research and includes possible future works.

Appendix A shows the mathematic models of IM and PMSM in both stationary and synchronously rotating reference frames.

Appendix B shows the solution of current dynamics in a vector controlled drive of AC Motor.
CHAPTER 2

SENSORLESS DIRECT FIELD ORIENTED CONTROL (DFOC) OF AC MOTOR

This chapter presents the principles of DFOC systems of both Induction Motor (IM) and Permanent-Magnet Synchronous Motor (PMSM). It also includes proposed flux and speed estimation designs that may be employed for both types of machines. And, finally, the evaluation of flux and speed estimations by experimental results is presented.

2.1 Introduction

For a generic DFOC system of AC motors, the flux and speed estimations are designed such that the DFOC controller can be applied to both IM and PMSM with minimum software configuration effort. During the past decade, several researchers have presented many techniques of flux and speed estimations for the vector controlled drive of AC motors (IM and PMSM) [1-9]. Almost all techniques are designed for either IM or PMSM because the mathematical models of these machines are different. One estimation technique that is possibly applicable to both types of machines employs a priori
knowledge of machine saliencies and requires the signal injections to determine the rotor position [10-11]. In this technique, the machine parameters are not essentially used in estimation, so it is a robust technique insensitive to the machine parameter variation. In addition, this estimation would perform successfully at very low or zero speed where other techniques relying on tracking speed dependent variables (e.g., back EMF) do fail. With this technique, the carrier excitation voltages with high frequency are injected on top of the fundamental command voltages from current regulators. The interaction between the carrier voltage and the machine saliency produces a current at the carrier frequency, precisely the negative-sequence component of carrier a current that contains information related to the position of the rotor saliency. The synchronous reference frame high-pass filters was designed to filter off the fundamental currents and positive-sequence component of the carrier current. Then, the Luenberger observer designed for speed and position estimations takes a corrective position error response driven by the observer controller. The original paper [10] was extended to [11], where different models of machine saliencies (such as rotating sinusoidally distributed saliency, non-sinusoidal saliency, stationary saliency) were introduced and further designed using the same observer structure.

Although this technique seems promising in a DFOC system of AC motors, some requirements may not be practical or difficult to achieve without knowing machine structures and parameters such as carrier frequency selection, saliency models, rotor modification, etc. The carrier frequency should be selected so that it is faster than the stator transient time constant [11], which requires the machine parameters implicitly be taken into consideration. Since the estimation is mainly designed by using a priori
knowledge of machine saliencies, the machine geometry both stator and rotor should be known in detail. For closed rotor slots type of machines, the machining process for opening rotor slots is necessary to produce rotor saliency [10]. In addition to that, other aspects such as harmonics and losses could be expectant to be more influential than other techniques without any signal injections.

2.2 Principles of Direct Field Oriented Control (DFOC) of AC Motor

During the past decade, several textbooks have presented the principles of FOC of AC motors [12-15]. Essentially, the main goal of a FOC system is to independently control torque and flux of an AC motor by using the rotor flux angle to transform the sinusoidal variables (i.e., currents, voltages, flux, etc.) to the DC-type quantities. Such transformation is the so-called “Park” transformation [16]. There is also another transformation, called “Clarke” transformation [17], which is intentionally used as an intermediary of variable transformation from abc-axis stationary reference frame to dq-axis synchronously rotating reference frame. The mathematical equations for these two transformations can be expressed as follows:

Clarke:

\[ i_{ds}^c = i_{as} \]
\[ i_{qs}^c = \frac{1}{\sqrt{3}} (i_{as} + 2i_{bs}) \]

Park:

\[ i_{ds}^e = i_{ds}^c \cos \theta + i_{qs}^c \sin \theta \]
\[ i_{qs}^e = i_{qs}^c \cos \theta - i_{ds}^c \sin \theta \]
In Park transformation, the choice of angle $\theta$ is very important for FOC, which will be depicted in the next subsections.

2.2.1 Induction motor

Within the mathematical model of the IM (in the synchronously rotating reference frame ($\omega=\omega_e$) from Equations (A.1) through (A.5)), when the synchronously rotating reference frame is aligned with the rotor flux angle ($\omega_e=\omega_{\lambda_r}$, meaning that $d$-axis angle is defined on the total rotor flux angle), then the $q$-axis rotor flux becomes zero, $\lambda^{e}_{qr}=0$ (see Appendix A). Thus, Equations (A.1) through (A.5) are reduced to the following equations:

$$\frac{di^{e}_{qs}}{dt} = -\gamma^{e}_{qs} - \omega_{e}i^{e}_{ds} - \beta \omega_{e}\lambda^{e}_{dr} + \beta_{1}v^{e}_{qs} \quad (2.1)$$

$$\frac{di^{e}_{ds}}{dt} = -\gamma^{e}_{ds} + \omega_{e}i^{e}_{qs} + \beta \alpha \lambda^{e}_{dr} + \beta_{1}v^{e}_{ds} \quad (2.2)$$

$$\frac{d\lambda^{e}_{dr}}{dt} = -\alpha \lambda^{e}_{dr} + \alpha L^{e}_{m}i^{e}_{ds} \quad (2.3)$$

$$T_{e} = \frac{3}{2} \frac{p \ L^{e}_{m}}{2 \ L^{e}_{r}} (\lambda^{e}_{dr} i^{e}_{qs}) \quad (2.4)$$

Clearly, the torque Equation (2.4) is simply controlled by $\lambda^{e}_{dr}$ and $i^{e}_{qs}$ where rotor flux $\lambda^{e}_{dr}$ is maintained constant by $i^{e}_{ds}$. Therefore, the FOC of IM can be established because the rotor flux and torque can be independently controlled by $i^{e}_{ds}$ and $i^{e}_{qs}$, respectively.
2.2.2 Permanent-magnet synchronous motor

Unlike IM, the FOC of PMSM is accomplished differently. According to Equations (A.8) through (A.12), if the $i_{ds}^e$ current is forced to be zero, then the stator flux and torque Equations (A.10) through (A.12) become:

\[
\lambda_{qs}^e = L_{s}i_{qs}^e \quad (2.5)
\]

\[
\lambda_{ds}^e = \lambda^e_m \quad (2.6)
\]

\[
T_e = \frac{3}{2} p \left( \lambda_{ds}^e i_{qs}^e \right) = \frac{3}{2} p \left( \lambda^e_m i_{qs}^e \right) \quad (2.7)
\]

Finally, the torque equation of PMSM looks similar to the torque equation of IM in (2.4), therefore the FOC of PMSM is accomplished where total flux appearing in (2.7) is the constant magnet flux and $i_{qs}^e$ independently controls the torque. For convenience, the stator current equations are shown here again.

\[
\frac{di_{qs}^e}{dt} = -\gamma_{qs}^e - \omega_{e} i_{ds}^e - \beta \omega_{e} \lambda^e_m + \beta v_{qs}^e \quad (2.8)
\]

\[
\frac{di_{ds}^e}{dt} = -\gamma_{ds}^e + \omega_{e} i_{qs}^e + \beta v_{ds}^e \quad (2.9)
\]

In summary, to establish FOC of PMSM, not only the rotor flux angle must be known similarly to FOC of IM, but also the $i_{ds}^e$ current must be controlled to zero. For the Direct FOC, the rotor flux estimation is required, which will be explained in following section.

2.3 Proposed Flux and Speed Estimations of AC Motor

Within the literature, many researchers developing the flux and speed estimators, primarily use the machine equations which have been specifically dedicated to either IM
Several flux and speed estimators have been presented with many techniques (e.g., Extended Kalman Filter (EKF) by [1-2], Observer Theory by [3-4], Voltage and Current Flux Model (VCFM) by [5], Model Reference Adaptive System (MRAS) by [6], and Sliding Mode Theory by [7-9]). In theory, these proposed estimators cannot essentially be utilized for both IM and PMSM because they are derived based on the machine mathematical models, which are different from each other for IM and PMSM. Consequently, these flux and speed estimators are the so-called the machine-type dependent schemes.

Additionally, some flux and speed estimators presented in the literature are complex and sometimes not practical for implementation due to their heavy computation. This is especially true if a low cost, fixed-point Digital Signal Processor (DSP) is used. Therefore, these complicated techniques are usually implemented on powerful, expensive floating-point DSP controllers using the high-level languages, where the code optimization, computation bandwidth, and numerical accuracy are not issues. It is evident that the complex techniques may not be suitable and attractive to the industry applications for easy adoption in low cost, fixed-point DSP controllers.

As mentioned earlier, one possible way to develop the flux and speed estimators applicable to both IM and PMSM is to look into the fundamental and common AC motor equations. The motor-type-dependent equations should be avoided as much as possible. An overview of the stator flux estimator is shown in Figure 2.1 [18-19]. The stator flux in the stationary reference frame is mainly computed by means of the improved integration of back EMF’s. By introducing the compensated stator flux generated by PI compensator, the errors associated with the pure integrator (i.e., DC-offset measurement and unknown
initial stator flux) can be carried out. The equations derived for this flux estimator are explained as follows:

Firstly, the back EMF in stationary reference frame (superscript “s”) can simply be calculated from the measured voltages and currents.

\[
\begin{align*}
R_s q_s v_s - R_s d_s i_s &= e_{qs} \\
R_s d_s v_d &= e_{ds}
\end{align*}
\]

Then, the stator flux can be computed by pure integration of back EMF in Equations (2.10) and (2.11) as follows:

\[
\begin{align*}
\lambda_{ds} &= \frac{1}{s} e_{ds} \\
\lambda_{qs} &= \frac{1}{s} e_{qs}
\end{align*}
\]

where \( s \) is the Laplace transform operator.
However, the Equations (2.12) and (2.13) may be equivalently rewritten as the following forms:

\[
\lambda_{ds}^s = \frac{1}{s + \omega_c} e_{ds}^s + \frac{\omega_c}{s + \omega_c} \lambda_{ds}
\]  

(2.14)

\[
\lambda_{qs}^s = \frac{1}{s + \omega_c} e_{qs}^s + \frac{\omega_c}{s + \omega_c} \lambda_{qs}
\]  

(2.15)

where \(\omega_c = 2\pi f_c\) is the cut-off angular velocity (rad/sec).

To accomplish the digital implementation of the pure integration, the quadrature detector is introduced in the compensated stator flux feedback loop. As is known, the back EMF waveform leads the stator flux waveform by 90°. Therefore, the quadrature detector is formulated as follows:

\[
\Delta e = \frac{\left(\lambda_{ds}^s e_{ds}^s + \lambda_{qs}^s e_{qs}^s\right)}{|\lambda_i^s|} = |e_i^s| \cos \gamma
\]  

(2.16)

where \(\gamma\) is the angle between back EMF and stator flux vectors and \(|\lambda_i^s| = \sqrt{(\lambda_{ds}^s)^2 + (\lambda_{qs}^s)^2}\).

The goal is to control the angle \(\gamma\) to be 90° because of the correct integration requirement for stator flux and back EMF. Thus, the quadrature detector must be minimized to be zero by introducing the compensated stator flux computed by PI controller.

\[
\lambda_{comp}^s = K_p \left(\Delta e + \frac{1}{T_i} \int (\Delta e) dt\right)
\]  

(2.17)

where \(K_p\) is the proportional gain and \(T_i\) is the integral time (sec).
As a result, the stator flux shown in Equations (2.14) and (2.15) can be rewritten again as:

\[
\lambda_{ds}^s = \lambda_{ds,1}^s + \lambda_{ds,2}^s = \frac{1}{s + \omega_c} e_{ds}^c + \frac{\omega_c}{s + \omega_c} \lambda_{comp}^s \cos \theta_c
\] (2.18)

\[
\lambda_{qs}^s = \lambda_{qs,1}^s + \lambda_{qs,2}^s = \frac{1}{s + \omega_c} e_{qs}^c + \frac{\omega_c}{s + \omega_c} \lambda_{comp}^s \sin \theta_c
\] (2.19)

where \( \theta_{\lambda s} = \tan^{-1} \frac{\lambda_{qs}^s}{\lambda_{ds}^s} \) is the stator flux angle (rad).

After the stator flux is computed based on Equations (2.18) and (2.19), then the synchronous speed and electromagnetic torque can be easily calculated as follows:

\[
\omega_e = \frac{d\theta_c}{dt} = \left( \frac{\lambda_{ds}^s e_{qs}^c - \lambda_{qs}^s e_{ds}^c}{|\lambda_s^s|^2} \right)
\] (2.20)

\[
T_e = \frac{3}{2} p \left( \lambda_{ds}^s i_{qs}^c - \lambda_{qs}^s i_{ds}^c \right)
\] (2.21)

Once the stator flux is estimated, the rotor flux is simply computed from stator flux and stator current for both IM and PMSM by using a common equation as shown:

\[
\lambda_{dr}^s = -K_1 i_{ds}^s + K_2 \lambda_{ds}^s
\] (2.22)

\[
\lambda_{qr}^s = -K_1 i_{qs}^s + K_2 \lambda_{qs}^s
\] (2.23)

where \( K_1 = \frac{L_s L_r - L_m^2}{L_m} \), \( K_2 = \frac{L_r}{L_m} \) for IM and \( K_1 = L_s \), \( K_2 = 1 \) for PMSM.

Finally, the rotor flux angle can be calculated and used for variable transformation in the FOC. The simple slip speed of IM is additionally added to get the rotor speed.

\[
\omega_{sl} = \frac{2}{3} \frac{2}{p} \frac{T_e R_r}{\left( \lambda_{dr}^s \right)^2 + \left( \lambda_{qr}^s \right)^2}
\] (2.24)
2.4 Evaluation of Flux and Speed Estimations

It is well known that the back EMF decreases as the rotor speed slows. Therefore, the accuracy of estimated stator flux at very low speed range is a major issue and has to be experimentally evaluated for any flux estimation technique based on the integration of back EMF. In this subsection, the stator flux and rotor speed estimation explained previously are evaluated at different speeds by a test setup with dq-axis current control loops as seen in Figure 2.2. No speed control is attempted. In this system, the stator currents are controlled at 2.5 A peak and the synchronous frequency, $f_e$, varies over a wide range. At each $f_e$, the back EMF and stator flux in stationary d-axis, and stator flux angle waveforms are captured with the speed monitored. The testing results are shown in Figures 2.3 through 2.8 for both IM and PMSM.

Figure 2.2: A dq-axis current control loop system used for stator flux and rotor speed evaluation
Figure 2.3: Experimental result - back EMF ($e_{ds}^s$), stator flux ($\lambda_{ds}^s$), and stator flux angle ($\theta_{\lambda_s}$) from estimator of the IM at $f_c = 5$ Hz.

Figure 2.4: Experimental result - back EMF ($e_{ds}^s$), stator flux ($\lambda_{ds}^s$), and stator flux angle ($\theta_{\lambda_s}$) from estimator of the PMSM at $f_c = 5$ Hz.
Figure 2.5: Experimental result - back EMF ($e_{ds}^s$), stator flux ($\lambda_{ds}^s$), and stator flux angle ($\theta_{\lambda_s}$) from estimator of the IM at $f_e = 60$ Hz.

Figure 2.6: Experimental result - back EMF ($e_{ds}^s$), stator flux ($\lambda_{ds}^s$), and stator flux angle ($\theta_{\lambda_s}$) from estimator of the PMSM at $f_e = 125$ Hz.
In Figures 2.3 and 2.4 ($f_c = 5$ Hz), the back EMF is clearly very low, but the stator flux can be successfully estimated with a correct 90° phase difference between the back EMF and the stator flux. This is because the compensated stator flux attempts to reduce the error of the quadrature detector by amplifying itself as the back EMF gets lower. In the high speed range, the compensated stator flux behaves oppositely where the component computed from back EMF is dominant. The results at higher speed range can be seen in Figures 2.5 and 2.6. As expected, the stator flux is correctly estimated without any problem.

Next, the rotor speed estimation is evaluated by using a trapezoidal profile of synchronous speed. Thus, the magnitude of the estimated stator flux is verified by the correct synchronous speed computed by Equation (2.20). The experimental results are shown in Figures 2.7 and 2.8. In the evaluation testing, the synchronous speed varies between 0.133 and 0.4 per unit. Clearly, the estimated rotor speed of both IM and PMSM is quite accurate.

Figure 2.7: Experimental result – synchronous ($\omega_e$), estimated rotor speed ($\hat{\omega}_r$), and measured rotor speed ($\omega_r$) of the IM at $\omega_e = 0.133 – 0.4$ per unit (base speed = 2250 rpm)
Figure 2.8: Experimental result – synchronous ($\omega_e$), estimated rotor speed ($\hat{\omega}_r$), and measured rotor speed ($\omega_r$) of the PMSM at $\omega_e = 0.133 \sim 0.4$ per unit (base speed = 7500 rpm)

To verify the accuracy of stator flux integration, i.e., its correct magnitude and $90^\circ$ phase difference with back EMF, both estimated synchronous speed and quadrature detector for IM and PMSM are captured as shown in Figures 2.9 and 2.10, respectively. In these Figures, the proposed stator flux integration can clearly be achieved over a wide range of speed. The magnitude can be verified by looking at the estimated synchronous speed while the quadrature detector can be maintained at zero, meaning that the stator flux has a $90^\circ$ phase difference with the back EMF.
Figure 2.9: Experimental result - synchronous speed ($\omega_e$), estimated synchronous speed ($\hat{\omega}_e$), and quadrature detector ($\Delta e$) of the IM at $\omega_e = 0.2 - 0.8$ per unit

(base speed = 1800 rpm)

Figure 2.10: Experimental result - synchronous speed ($\omega_e$), estimated synchronous speed ($\hat{\omega}_e$), and quadrature detector ($\Delta e$) of the PMSM at $\omega_e = 0.2 - 0.8$ per unit

(base speed = 6000 rpm)
CHAPTER 3

ISSUES ON FIXED-POINT DSP BASED CONTROL AND DRIVE
SYSTEM IMPLEMENTATION

In this chapter, the experimental setup for both hardware and software aspects are explained as well as an overview of digital control systems. Fixed-point and floating-point representations are also illustrated as are the effects of quantization errors in fixed-point implementation. And, lastly, a case study of simulated sensorless DFOC systems of induction motors is implemented on a eZdsp2812 controller in order to show the accuracy of the 32-bit fixed-point resolution.

3.1 Introduction

In cost-sensitive industrial applications, a fixed-point DSP is usually selected as the first choice because it is much less expensive than a floating-point DSP. In most fixed-point DSPs, the low-level language (i.e., assembly language) is typically used for programming because the code has to be optimized so that it can be executed within the limited number of cycles of an interrupt period. However, several practical concerns related to fixed-point implementation such as quantization errors, scaling of physical
variables (e.g., currents, voltages, etc.) have to be seriously taken into account. These concerns are not issues for floating-point implementation where the high-level language (e.g., C or C++) is usually used for coding. Consequently, the fixed-point programming using assembly language is a challenging task which requires much coding experience. However, these issues for the fixed-point DSP can be improved by increasing the number of bits to extend the dynamic range, by increasing clock speed to reduce the time (to allow more codes to be executed within the same interrupt period), and by providing the efficient supporting C library (to make C programming easier). With these improvements of a fixed-point DSP, it could compete with a floating-point DSP. In the DSP market, one such fixed-point DSPs is a 32-bit TMS320F281x series and its C support library so called “IQmath” from Texas Instruments, which will be explained in the next sections.

3.2 Hardware and Software Aspects

3.2.1 Laboratory setup

Figure 3.1 shows the overall experimental setup used for all experiments in this research work. The function of the isolated transformer is to electrically isolate the power inverter board from the utility. This allows the oscilloscope to probe with any components on the power inverter board without possible shorted neutral paths from the inverter to the utility. The purpose of variac (i.e., auto-transformer) is to safely adjust the DC-bus voltage during algorithm development. In this figure, the DSP controller is directly connected to the power inverter. In the next subsections, more details on the power inverter and DSP controller will be given.
3.2.2 32-bit fixed-point DSP based controller (TMS320x28x series)

Recently, the high-speed, 32-bit fixed-point DSP from Texas Instruments (TMS320x28x series), designed for embedded digital control systems, was introduced to the market. The outstanding features of this DSP are its high-speed operation (up to 300 MHz system clock) and its native 32-bit precision, which allows one to easily develop any complex algorithms of a digital motor control system by using high-level language (e.g., C, C++) without too much concern for the computation bandwidth and numerical resolution problems. The dynamic range of numbers is much improved compared with the conventional 16-bit fixed-point DSP. In this research, all experiments have been implemented on this fixed-point TMS320x28x DSP based controller, so-called
ezdsp2812 board. This DSP board is designed to connect directly with the power inverter board as seen in Figure 3.1.

3.2.3 The flexible three-phase voltage source inverter (DMC1500 Board)

The commercial three-phase voltage source inverter built by Spectrum Digital is conveniently employed to drive the three-phase AC motors rated up to 1-hp, 208 volt (line-to-line). It is compatible with all TI DSP development controller system kits such as F240/F243/LF2407 DSK/EVM, and ezdsp2407/2812. It is also designed to generally drive most types of three-phase AC motors such as induction, permanent-magnet synchronous, DC-brushless, and switched reluctance motors. All experiments in this research were established by this inverter. The technical details of this inverter can be further found in [20].

3.2.4 Low-cost techniques for line current and voltage measurements

In the DMC1500 board, the low-cost solution of current measurement called “viewing resistor” technique is used in experiments. In each switching leg, the small value of resistance (0.04 $\Omega$) is embedded below the lower switch. The requirement of this technique is to trigger the ADC when all upper switches are turning off as seen in Figure 3.2. Then, by measuring the voltage across these resistances, the line currents can be detected at this time. The start of ADC can be programmable within DSP at the beginning of PWM period (timer1 underflow) when all upper switches are turned off.
Typically, current sensors are current transformers or Hall effect types, which are commonly found in any inverter. These common sensors are not limited in terms of a start of ADC, which is required in the current measurement using inexpensive viewing resistor techniques as explained above. Some issues related to current measurements (any kind of current sensors) in digital motor drives using Pulse-Width-Modulation (PWM) operation can be briefly addressed as follow. S.-K. Sul et al. [21] investigated the effects of low-pass filter on current sampling and proposed a technique to compensate the delay of the filter. The current-sampling error due to timing delay error is analytically derived in four cases depending on PWM being turned on or off which activate vectors in Space-Vector PWM strategy. However, one of their assumptions about negligible voltage drop
at stator resistance could be too conservative, therefore their analysis may need to be investigated further for influences of stator resistance voltage drop in smaller-size motors.

For the phase voltage measurement in Y-connected windings, rather than using the direct measurement of voltages in the inverter, the phase voltages are reconstructed from measured DC-bus voltage ($V_{dc}$) and known switching functions ($T_a$, $T_b$, and $T_c$) by using the following equations.

\[
v_{an} = V_{dc} \left( \frac{2}{3} T_a - \frac{1}{3} T_b - \frac{1}{3} T_c \right)
\]

\[
v_{bn} = V_{dc} \left( \frac{2}{3} T_b - \frac{1}{3} T_a - \frac{1}{3} T_c \right)
\]

\[
v_{cn} = V_{dc} \left( \frac{2}{3} T_c - \frac{1}{3} T_a - \frac{1}{3} T_b \right)
\]

where $T_a$, $T_b$, and $T_c$ = either 0 or 1. Based on these techniques, the sensor cost for both current/voltage measurements can be reduced.

3.2.5 Easy fixed-point “C” programming by “IQmath” approach

Texas Instruments has developed the “IQmath” library used in C/C++ language to help programmers easily porting a program written by floating-point to fixed-point format. As a result, the spending time of writing a program can be reduced from days to hours. This library supports all common necessary functions such as multiplication, division, sine, cosine, arctangent, square root, etc. To clearly see how this “IQmath” approach works, a simple equation implementing in three different approaches is illustrated in Table 3.1.
<table>
<thead>
<tr>
<th>Approach</th>
<th>Data type</th>
<th>C code</th>
</tr>
</thead>
<tbody>
<tr>
<td>floating-point</td>
<td>float y,m,x,b;</td>
<td>y = m*x + b;</td>
</tr>
<tr>
<td>traditional fixed-point</td>
<td>long y,m,x,b;</td>
<td>y = (long64)(m*x)\gg\text{GLOBAL_Q} + b;</td>
</tr>
<tr>
<td>“IQmath”</td>
<td>_iq y,m,x,b;</td>
<td>y = _IQmpy(m,x) + b;</td>
</tr>
</tbody>
</table>

Table 3.1: Illustration of “IQmath” approach comparing with traditional fixed-point and floating-point approaches in C language

As seen in Table 3.1, the data type “_iq” is to identify the variables in GLOBAL_Q format. In this case, all variables (i.e., y, m, x, b) are in GLOBAL_Q format. The “_IQmpy(m,x)” function actually replaces “(long64)(m*x)\gg\text{GLOBAL\_Q}” statement. This scaling process for multiplication (i.e., \gg means right shifted operation to align the fractional bits) is time consuming and difficult to debug (or understand) when implementing this traditional fixed-point approach in a complicated algorithm. By using “IQmath” function (in this case, _IQmpy()), the scaling process is hidden to the programmers. In “IQmath” library, an overall fractional scaling parameter so-called GLOBAL_Q (i.e., number of fractional bits) can be adjustable to easily observe the dynamic range and numerical resolution of the overall system. However, the Q format of each variable is not strict to GLOBAL_Q. Each variable can be set at any Q format rather than GLOBAL_Q. The details such as quantization problem investigation, (cycles/code) performance comparison, etc. of this “IQmath” approach between fixed-point 28x DSP and floating-point 3x DSP can be found in [22].
3.3 Brief Overview of Digital Control System

In general, most control algorithms are initially designed in the continuous-time domain. Typically, those control strategies are, however, implemented in the digital controller. Consequently, the continuous-time control algorithms are necessarily transformed to the discrete-time domain. Techniques for such transformation from continuous-time domain (s-domain) to discrete-time domain (z-domain) are summarized as follows:

- Discrete-equivalent via numerical integration such as forward (or Euler), backward, trapezoidal (Tustin, or bilinear), and bilinear approximations with frequency prewarping.
- Zero-pole matching equivalent.
- Hold equivalent such as zero-order-hold and noncausal first-order or triangle-hold.

The details of these techniques can be found in [23-26]. The simplest and most convenient technique is the discrete-equivalent because the discretization can be proceeded from several continuous-time forms such as differential equations, Laplace transfer function, or state-space equation. With a high enough sampling rate, the simple forward or backward approximation could provide a good discrete-time model. Therefore, the forward or backward approximation is often used because of its simplicity.

In practice, the sampling rate is usually selected in a range of 20-40 times of closed-loop system bandwidth [23]. The performance of a digital controller improves with faster sampling rate. However, the higher sampling rate increases cost of the digital controller.
and Analog-to-Digital (A-D) converter because of required higher word-length and less A-D conversion time, respectively. In addition, the stability margin is decreased as sampling rate increases. When the sampling period used in discretization process is getting less, the word size in fixed-point controller has to be increased in order to improve the numerical resolution.

In motor control applications, most control algorithms are also digitally implemented in the computing device such as microcontroller, microprocessor, or digital signal processor (DSP), etc. Figure 3.3 shows a typical digital motor control drive system for a three-phase AC motor.

![Figure 3.3: A typical digital motor control drive system for a three-phase AC motor](image)

In this figure, the continuous signals in the system (i.e., measured signals such as currents, voltages, DC-bus voltage, etc.) are discretized by the A-D converters with a certain bit of resolution (typically, 8-14 bits) before feeding the measured signals in the digital format into the computing device. Then, the computing device does calculations based on the desired control algorithm before sending the proper PWM signals (digital signals) to turn on or turn off the switching devices in the inverter. The function of clock
inside digital controller is to keep timing for A-D triggering and the algorithm scheduling (i.e., interrupt service routine) with a sampling period, T. The range of typical sampling rate of digital controllers for motor control applications is about 10-20 kHz. Consequently, two major sources of quantization found in the system are the A-D converter and the finite word length of computing device [27]. Since numbers representing in the computing device have to fit in digital words with a finite number of bits, the quantization errors happen due to number truncation or rounding-off. Figure 3.4 shows the quantized signal of an ideal A-D converter, consisting of a simpler and a zero-order-hold (ZOH). The quantization errors caused by the finite word length (truncation or rounding-off) can be depicted in Figure 3.5.

Since the quantization always exists in the digital control system, the errors associated with quantization, especially caused by the finite word-length, would affect the system performance and behavior of the digital control system.

![Figure 3.4: The quantized signal caused by an ideal A-D converter](image)
3.4 Fixed-point Versus Floating-point

Technically, the computing device can be categorized into two main types: fixed-point and floating-point. The fixed-point device typically has less complex circuitry compared with the floating-point device, thus resulting in a lower cost. The number representation for 16-bit or 32-bit fixed-point and 32-bit floating-point (IEEE-754 standard) is different as seen in Figure 3.6. In a fixed-point device, the equivalent decimal number is calculated by adding up the bit weights for each bit that is “1”. The bit weights are determined depending on the radix point location within the word. For instance, the
radix point is placed right after the sign bit (most significant bit, MSB). The bit weights for this particular location of radix point for 16-bit and 32-bit fixed-point are shown in Figure 3.7. This number representation is so-called Q15 format for 16-bit fixed-point and Q31 format for 32-bit fixed-point. The number 15 or 31 basically represents the number of fractional bits. Therefore, the maximum decimal number in this case is nearly but less than 1 whereas the smallest one is $2^{-15}$ (for Q15) or $2^{-31}$ (for Q31). Generally, the radix point could be placed in any location within the word. For instance, when the radix point is right shifted 1 bit from one seen in Figure 3.7, then all shown bit weights are multiplied by 2. As a result, the maximum decimal number becomes nearly but less than 2 whereas the smallest one is $2^{-14}$ (for Q14) or $2^{-30}$ (for Q30). Thus, the number format could be varied from Q15 to Q0 for 16-bit fixed-point and from Q31 to Q0 for 32-bit fixed-point, depending on the location of radix point. When the number is in Q0 format, it means that the number represents the signed integer without any fraction. Normally, the selection of Q format depends on the numerical range between the largest and smallest number possibly occurred in the algorithms implemented in the fixed-point digital controller. These algorithms are typically normalized in per-unit system in order to scale all physical variables (e.g., voltage, current, torque, speed, flux, etc.) varying within the same numerical range.

Unlike fixed-point numbers, the radix point is floated by means of mantissa and exponent to represent numbers in the floating-point device. The equivalent decimal number is computed as.

$$\text{value} = \text{mantissa} \times 2^{\text{exponent}} \quad (3.1)$$
As a result of using mantissa and exponent, the floating-point device gives much wider dynamic range than fixed-point device. The dynamic range is defined as a ratio between the largest and smallest numbers that can be represented [28]. The dynamic range in dB unit is formulated as

\[
dynamic\ range = 20 \log \left( \frac{\text{largest number}}{\text{smallest number}} \right) \ dB
\]  

(3.2)
According to the 32-bit floating-point representation (IEEE-754 standard), the equivalent decimal number can be computed by a given 23-bit mantissa (m), 8-bit exponent (e), and 1-bit sign (s) from the following rules:

1) if \( e = 255 \) and \( m \neq 0 \), then value = NAN (not a number)
2) if \( e = 255 \) and \( m = 0 \), then value = \((-1)^s \times \infty\) (infinity)
3) if \( 0 < e < 255 \), then value = \((-1)^s \times 2^{(e-127)} \times (1.m)\)
4) if \( e = 0 \) and \( m \neq 0 \), then value = \((-1)^s \times 2^{-127} \times (0.m)\)
5) if \( e = 0 \) and \( m = 0 \), then value = \((-1)^s \times 0\) (zero)

Table 3.2 summarizes the numerical and dynamic ranges for 16-bit fixed-point, 32-bit fixed-point, and 32-bit floating-point. As seen in this Table, the dynamic range of the floating-point is dramatically high, thus the quantization errors due to the finite word-length are typically unnoticeable when the algorithms are implemented in a floating-point device. Consequently, the floating-point is generally selected in applications where the numerical range is significantly wide and requirement of high precision is needed. On the other hand, the fixed-point device is dominant in the cost-sensitive applications because of its inexpensive architecture compared to the floating-point device.

<table>
<thead>
<tr>
<th></th>
<th>Smallest number (resolution)</th>
<th>Least negative number</th>
<th>Largest positive number</th>
<th>Dynamic range (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-bit fixed-point (Q15)</td>
<td>( 2^{-15} )</td>
<td>-1</td>
<td>1-2^{-15}</td>
<td>90</td>
</tr>
<tr>
<td>32-bit fixed-point (Q31)</td>
<td>( 2^{-31} )</td>
<td>-1</td>
<td>1-2^{-31}</td>
<td>187</td>
</tr>
<tr>
<td>32-bit floating-point*</td>
<td>( \approx 2^{-127} )</td>
<td>( \approx -2^{128} )</td>
<td>( \approx 2^{128} )</td>
<td>1535</td>
</tr>
</tbody>
</table>

* IEEE-754 single-precision standard (23-bit mantissa, 8-bit exponent, 1-bit sign)

Table 3.2: Smallest, least negative, largest positive numbers, and dynamic range among 16-bit fixed-point, 32-bit fixed-point, and 32-bit floating-point representation
3.5 Effects of Quantization Errors

As explained in the previous section, the quantization errors are inherent in any fixed-point device as categorized in the two following ways; coefficient and signal quantizations.

3.5.1 Coefficient quantization

Finite word-length causes the truncated or rounded-off coefficients, which alter the transfer function of the system, the pole-zero locations, and the gain of the system. The quantized coefficients are dependent on the sampling rate, word-length, and technique of discretization. As the sampling rate increases, the pole tends to move toward and cluster around \( z = 1 \), causing the stability of discrete-time system very susceptible to coefficient quantization. To understand this phenomenon better, a transfer-function in continuous-time domain in (3.3) is discretized by using discrete-equivalent via numerical integration method.

\[
H(s) = \frac{K_c (\mu_1 s + 1) \cdots (\mu_n s + 1)}{(\tau_1 s + 1) \cdots (\tau_n s + 1)} \tag{3.3}
\]

By simply using backward approximation, the \( s \)-operator is replaced by \( \frac{z - 1}{Tz} \) [23], thus the transfer function in discrete-time domain with a sampling period (T) becomes

\[
H(z) = \frac{K_d (1 - z_i z^{-1}) \cdots (1 - z_n z^{-1})}{(1 - p_i z^{-1}) \cdots (1 - p_n z^{-1})} \tag{3.4}
\]

where \( K_d = \frac{K_c (\mu_1 + T) \cdots (\mu_n + T)}{(\tau_1 + T) \cdots (\tau_n + T)} \), \( z_i = \frac{\mu_i}{\mu_i + T} \), and \( p_i = \frac{\tau_i}{\tau_i + T} \) \tag{3.5}
As seen in (3.5), when the sampling rate increases, the poles \( p_i \) move toward and cluster about \( z = 1 \), i.e., \( \lim_{T \to 0} p_i = 1 \).

To illustrate the effects of coefficient quantization, the change of poles (or zeros) as a function of the change of coefficients is considered. A model of the theoretical controller transfer function is shown as

\[
H(z) = \frac{a_0 + a_1 z^{-1} + \cdots + a_w z^{-w}}{1 + b_1 z^{-1} + \cdots + b_n z^{-n}} = \frac{(1-z_1 z^{-1}) \cdots (1-z_w z^{-1})}{(1-p_1 z^{-1}) \cdots (1-p_n z^{-1})} \tag{3.6}
\]

However, the implemented transfer function that takes coefficient quantization into account is as

\[
\tilde{H}(z) = \frac{a_0' + a_1' z^{-1} + \cdots + a_w' z^{-w}}{1 + b_1' z^{-1} + \cdots + b_n' z^{-n}} = \frac{(1-z_1' z^{-1}) \cdots (1-z_w' z^{-1})}{(1-p_1' z^{-1}) \cdots (1-p_n' z^{-1})} \tag{3.7}
\]

where \( a_i' = a_i + \Delta a_i \), \( z_i' = z_i + \Delta z_i \), \( b_i' = b_i + \Delta b_i \), and \( p_i' = p_i + \Delta p_i \).

In [24], the change of poles as a function of the change of coefficients is provided as.

\[
\Delta p_i = -\sum_{j=1}^{n} \frac{p_i^{n-j} \Delta b_j}{\prod_{k=1 \atop k \neq i}^{n} (p_i - p_k)} \tag{3.8}
\]

As seen in this equation, the \( \Delta p_i \) is very sensitive to others poles \( p_j \) that are closed to \( p_i \). Therefore, if the theoretical poles in a controller are clustered together, then the truncation or rounding-off of coefficients can cause significant change in the realized poles. To reduced \( \Delta p_i \), the \( \Delta b_j \) has to be minimized by increasing the word-length.

### 3.5.2 Signal quantization

Finite word-length can also cause signal quantization in the three different categories.
• **A-D and D-A converters:** One type of signal quantization happens upon the conversion and representation of a continuous signal into discrete magnitude by an A-D or a D-A converter with a typical word-length of 8-14 bits. Selection of A-D and D-A converters is usually not a major quantization problem when implementing the digital controller. Quantization errors from numerical calculation (i.e., truncation, rounding-off, and overflow) are often mistaken as low resolution in the input/output signal. In a servo control system, if the reference signal has higher precision than the feedback signal, discretized by an A-D converter, then the error will never go to zero, causing a limit cycle [24, 29].

• **Truncation and rounding-off:** The second type of signal quantization occurs when results of signal processing are truncated and rounded-off. As intermediate calculations are carried out, they need higher precision. For instance, a 16×16-bit multiplication requires a 32-bit register to store the result. If only 16 bits are used to keep the result and the lower 16 bits are thrown away, then this is known as truncation error. If the least significant bit (LSB) is rounded before throwing away the lower 16 bits, this is known as rounding-off error. If these errors are fed back recursively, they will accumulate as successive calculations are performed. Truncation and rounding-off introduce bias and noise in the system, which may produce limit cycles because of nonlinearities. In addition, sometimes signals are calculated from some mathematical functions, e.g., trigonometric, exponential functions, etc. These functions are practically realized by using look-up tables with a certain size of table and a word-length of each entry. Therefore, the results
are obviously quantized. To improve the accuracy of result, both size of table and word-length of each entry have to be increased.

- **Overflow or underflow**: The third kind of signal quantization is the overflow or underflow conditions. Successive addition or subtraction can cause registers to overflow or underflow even when the fractional arithmetic is used. This, in turn, will force the contents of associated registers to wrap around and change magnitude from most positive to most negative numbers or from most negative to most positive numbers (in twos complement arithmetic). This is equivalent to changing the direction of the control. In some computing devices, a saturation mode is provided to prevent the contents of registers from wrapping around and changing sign when an overflow or underflow occurs. Overflow or underflow as well as saturation included can also be minimized by the proper selection of scaling factors (i.e., proper base quantities in per-unit system) and by leaving extra guard bits.

In general, the quantization effects due to A-D or D-A converter are not significant and overflow or underflow can easily be manageable through the proper scaling factors. The effects of coefficient and truncation/rounding-off quantizations are most serious and most difficult to handle when implementing complex algorithms in the digital controller. In fact, there are two major effects caused by quantization errors, i.e., stability and limit cycle.
3.5.3 Stability

Stability of algorithms realized in a digital controller is affected by sampling rate and resolution of finite word-length (e.g., 16-bit or 32-bit). Although the transfer function of designed controller is stable, instability could easily happen when the high sampling rate and short word-length are selected for digital implementation, causing the realized poles significantly different from theoretical poles and moved toward around the stability limit at $z = 1$. Generally, the choice of sampling rate is determined by the closed-loop system bandwidth. The word-length of controller should be carefully selected so that the realized poles are still located within the stability limit, $|z| = 1$. In general, when the sampling rate increases, the required word-length is typically increased as well. For instance, the integral output of a digital PI controller is actually an accumulator of a multiplication of error and integral gain $K_i$ (by using backward approximation). As a result of discretization, this gain is usually the product of the sampling period and the gain obtained from original design in continuous domain, and is typically in the order of $\times 10^{-5}$ (for a typical motor drive system with 20 kHz sampling rate). For a 16-bit fixed-point signed representation, the minimum resolution is $2^{15} = 3.05 \times 10^{-5}$ which is on the same magnitude as the integral gain. Therefore, the 16-bit multiplication operation between the error and $K_i$ cannot provide enough accuracy because the quantization errors due to the truncation or rounding-off of both coefficients and multiplication results are relatively significant. Consequently, the integral output of the digital PI controller and gain $K_i$ are typically represented with the 32-bit word-length.

To clearly see the stability effects, an example of 2nd-order transfer function is illustrated as follow.
Continuous-time: \[ H(s) = \frac{1}{(\tau_1 s + 1)(\tau_2 s + 1)} \] (3.9)

Discrete-time: \[ H(z) = \frac{K_d}{(1 - p_1 z^{-1})(1 - p_2 z^{-1})} \] (3.10)

where \[ K_d = \frac{T^2}{(\tau_1 + T)(\tau_2 + T)} \]
and \[ p_1 = \frac{\tau_1}{\tau_1 + T} \] (using backward approximation).

Poles in continuous domain are located at \( p_1 = -1/\tau_1 \) and \( p_2 = -1/\tau_2 \). And poles in discrete domain calculated according to the sampling period \( T \) are \( p_1 = \frac{\tau_1}{\tau_1 + T} \) and \( p_2 = \frac{\tau_2}{\tau_2 + T} \).

Table 3.3 summarizes theoretical and realized discrete poles with the time constants \( \tau_1 = 10 \) and \( \tau_2 = 0.01 \) and different sampling periods when using 16-bit and 32-bit word-lengths.

<table>
<thead>
<tr>
<th>Case</th>
<th>( T ) (sec)</th>
<th>Theoretical discrete poles</th>
<th>16-bit word-length</th>
<th>32-bit word-length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>( p_1 )</td>
<td>( p_2 )</td>
<td>( p_1 )</td>
</tr>
<tr>
<td>1</td>
<td>0.5</td>
<td>0.9523810</td>
<td>0.0196078</td>
<td>0.9523926</td>
</tr>
<tr>
<td>2</td>
<td>0.005</td>
<td>0.999500</td>
<td>0.6666667</td>
<td>0.9995117</td>
</tr>
<tr>
<td>3</td>
<td>0.00005</td>
<td>0.999995</td>
<td>0.9950249</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3.3: Theoretical and realized discrete poles with time constants \( \tau_1 = 10 \) and \( \tau_2 = 0.01 \) and different sampling periods when using 16-bit and 32-bit word-lengths

As seen in this table, when sampling rate is increased, two theoretical discrete poles gradually approach to 1, where the stability is very sensitive. For case 3 \( (T = 0.00005 \) sec.), one realized pole becomes 1 for 16-bit word length due to rounding off. As a result, this 2\textsuperscript{nd}–order transfer function has to be implemented on the word-length higher than 16-
bit in order to maintain stability. In this case 3, the 32-bit word length can provide the exact same poles as the theoretical poles.

3.5.4 Limit cycle

The oscillating signal phenomenon is known as limit cycle. As mentioned previously, the nonlinearities due to the signal quantization errors cause limit cycle in the digital implementation [23, 24, 29, 30]. The magnitude of limit cycle is approximately in the same order of quantization level, \( q = 2^n \) where \( n \) is a number of bits. Another phenomenon due to truncation and rounding-off errors is called “limit constant” [24]. In fact, both limit constant and limit cycle always exist in any fixed-point implementation. To depict clearly, two examples will be provided as follows:

Firstly, the limit constant due to rounding-off and truncation multiplication can be illustrated by a simple first-order low-pass filter that has transfer function as follow:

\[
T(s) = \frac{Y(s)}{U(s)} = \frac{1}{\tau s + 1} \tag{3.11}
\]

where \( \tau = 1/2\pi f_c \) be time constant (sec) and \( f_c \) be cut-off frequency (Hz).

The discrete-time transfer function of this low-pass filter by using backward approximation is then

\[
T(z) = \frac{Y(z)}{U(z)} = \frac{T}{(\tau + T) - \tau z^{-1}} \tag{3.12}
\]

where \( T \) is sampling period (sec).

Then, this discrete-time low-pass filter is implemented on TMS320F2812 eZdsp. Fortunately, the “IQmath” library contains both truncated and rounded multiplications; _IQmpy() and _IQrmpy(). With \( T = 0.00005 \) sec. and \( f_c = 120 \) Hz, the output responses to
a constant input, $u = 0.00001$, are logged out and plotted in Matlab. Figures 3.8 and 3.9 show the results for truncated and rounded multiplications, respectively, comparing with floating-point result.

As seen in Figures 3.8 and 3.9, both truncated and rounded multiplications always give steady-state error, or limit constant. The output, $y$, should be equal to input $u = 0.00001$ at steady state time. By using rounded multiplication, this steady-state error can be reduced about a half as observed in these two figures. The increasing number of fractional bits (or Q format) can be decreased this error as well.

Figure 3.8: Input, output, and output error between floating-point and fixed-point using truncated multiplication in digital implementation of first-order low-pass filter
Figure 3.9: Input, output, and output error between floating-point and fixed-point using rounded multiplication in digital implementation of first-order low-pass filter

In the second example, the limit cycle is identified in a negative feedback loop seen in Figure 3.10. Similarly, this feedback loop system is also implemented within DSP. The plant is a second-order type that has transfer function as shown in the figure. To illustrate the limit cycle, the PID controller is implemented in three formats; floating-point, Q24 fixed-point, and Q15 fixed-point while the plant is implemented only in floating-point version. For a reference, \( r(t) = 0.123 \), \( T = 0.0005 \) sec, and the plant characteristics: \( \zeta = 0.1 \), \( \omega_n = 150 \) rad/sec, the output responses, \( y(t) \), for Q15 and Q24 fixed-point implementations are shown in Figures 3.11 and 3.12, respectively. Both figures compare with the floating-point result. The lower row of these Figures show the output responses at steady state time after time = 0.12 sec. In Figure 3.11, the limit cycle is clearly identified as well as limit constant. As the Q format increased from Q15 to Q24,
the limit cycle is nearly disappeared and the output is nearly identical to floating-point one as seen in Figure 3.12.

Figure 3.10: A negative feedback loop used for illustration of limit cycle

Figure 3.11: Outputs of the negative feedback loop for floating-point and Q15 fixed-point versions (limit cycle appeared)
In summary, all effects of quantization errors, especially stability and limit cycle, can be reduced when the word length is increased. Typically, the 32-bit word length would make the effects of quantization unnoticed [23].

3.6 A Case Study of Sensorless DFOC system of Induction Motor

In this section, the effects of quantization errors are demonstrated in a sensorless DFOC system of IM [31]. The algorithms are implemented in three different formats, 16-bit fixed-point, 32-bit fixed-point, and floating-point, by using the same 32-bit fixed-point DSP controller (i.e., TMS320F2812 eZdsp) and “IQmath” library from Texas Instruments. This “IQmath” library allows one to easily convert the C code written in
floating-point format to 32-bit fixed-point code with adjustable overall Q format (called GLOBAL_Q). Thus, the C language is mainly used for programming. Both simulation and real implementation will be carried out in order to investigate the system performance and behaviors due to quantization errors. The coefficient and truncation/rounding-off quantizations will be focused because these errors are major sources as the recursive computation performed. In the studying sensorless DFOC system, the overall of the flux estimator can be shown in Figure 3.13 [5]. This flux estimation scheme has been reported that it is very effective over a wide range of speed because it utilizes a combination of two flux estimations based on voltage and current models. The rotor flux linkages in the stationary reference frame are mainly computed by means of the integral of back EMF’s in the voltage model. By introducing the compensated voltages generated by PI compensators, the errors associated with pure integrator and stator resistance measurement can be taken care.

Figure 3.13: Overall system of flux estimator of IM
Equations derived for this flux estimator are summarized as follows. In the current model, total rotor flux linkage is aligned into the d-axis component. Thus, the oriented rotor flux dynamics in current model as indicated in Figure 3.13 are as follow.

\[
\frac{d\lambda_{dr}^{e,i}}{dt} = \frac{1}{\tau_r} \left( L_m i_{ds}^e - \lambda_{dr}^{e,i} \right) \tag{3.13}
\]

\[
\lambda_{qr}^{e,i} = 0 \tag{3.14}
\]

where \( L_m \) is the magnetizing inductance (H), \( \tau_r \) is the rotor time constant (sec).

Then, the rotor flux linkages in (3.13) and (3.14) are transformed into the stationary reference frame performed by the inverse park transformation. Next, the stator flux linkages in stationary reference frame are computed from the rotor flux linkages and stator currents as follow:

\[
\lambda_{ds}^{s,i} = L_s i_{ds}^s + L_m i_{dr}^s = \left( \frac{L_s L_{ir} - L_m^2}{L_r} \right) i_{ds}^s + \frac{L_m}{L_r} \lambda_{dr}^{s,i} \tag{3.15}
\]

\[
\lambda_{qs}^{s,i} = L_s i_{qs}^s + L_m i_{qr}^s = \left( \frac{L_s L_{ir} - L_m^2}{L_r} \right) i_{qs}^s + \frac{L_m}{L_r} \lambda_{qr}^{s,i} \tag{3.16}
\]

where \( L_s \) and \( L_r \) are the stator and rotor self inductance (H), respectively.

Next, the stator flux linkages in the voltage model are computed by means of back EMF’s integration with compensated voltages.

\[
\lambda_{ds}^{s,v} = \int \left( v_{ds}^s - i_{ds}^s R_s - v_{comp,ds} \right) dt \tag{3.17}
\]

\[
\lambda_{qs}^{s,v} = \int \left( v_{qs}^s - i_{qs}^s R_s - v_{comp,qs} \right) dt \tag{3.18}
\]

where \( R_s \) is the stator resistance (\( \Omega \)), \( v_{ds}^s, v_{qs}^s \) are stationary dq-axis stator voltages, and the compensated voltages, \( v_{comp,ds}^s, v_{comp,qs}^s \), are computed by the PI control law. The
proportional gain $K_p$ and the reset time $T_i$ are chosen such that the flux linkages computed by current model is dominant at low speed because the back EMF’s computed by the voltage model are extremely low at this speed range. While at high speed range, the flux linkages computed by voltage model is dominant.

Once the stator flux linkages in (3.17) and (3.18) are calculated, the rotor flux linkages based on the voltage model are further computed, by rearranging (3.15) and (3.16), as

$$\lambda^{s,v}_{dr} = \frac{L_s L_r - L_m^2}{L_m} \lambda^{s,v}_{ds} + \frac{L_r}{L_m} \lambda^{s,v}_{ds} \tag{3.19}$$

$$\lambda^{s,v}_{qr} = \frac{L_s L_r - L_m^2}{L_m} \lambda^{s,v}_{qs} + \frac{L_r}{L_m} \lambda^{s,v}_{qs} \tag{3.20}$$

Finally, the rotor flux angle based on the voltage model and estimated speed is computed as follows:

$$\theta_{\lambda_r} = \tan^{-1} \left( \frac{\lambda^{s,v}_{qr}}{\lambda^{s,v}_{dr}} \right) \tag{3.21}$$

$$\omega_r = \frac{\text{d}\theta_{\lambda_r}}{\text{d}t} - \frac{1}{(\lambda^s_r)^2} \frac{L_m}{\tau_r} \left( \lambda^{s,v}_{dr} i^s_{qs} - \lambda^{s,v}_{qr} i^s_{ds} \right) \tag{3.22}$$

In addition, the synchronous speed in (3.22) is necessary to be filtered out by the low-pass filter in order to reduce the amplifying noise generated by the pure differentiator in (3.22). The simple 1st-order low-pass filter is used, then the actual synchronous speed to be used is the output of the low-pass filter, $\hat{\omega}_e$, seen in the following equation. The continuous-time equation of the 1st-order low-pass filter is as

$$\frac{\text{d}\hat{\omega}_e}{\text{d}t} = \frac{1}{\tau_c} (\omega_e - \hat{\omega}_e) \tag{3.23}$$
where \( \tau_c = \frac{1}{2\pi f_c} \) is the low-pass filter time constant (sec), and \( f_c \) is the cut-off frequency (Hz).

### 3.6.1 Simulation results

The 32-bit fixed-point TMS320F2812 eZdsp board and Code Composer Studio (CCS) V2.2 are mainly used to simulate the system. The variables are exported by using “File I/O” feature available in CCS and then plotted by Matlab. The overall block diagram of “simulated” sensorless DFOC system of induction motor is shown in Figure 3.14. In this figure, highlighted blocks are realized in three data formats; 16-bit fixed-point, 32-bit fixed-point, and floating-point. When simulating for 16-bit fixed-point version, the “emulated” induction motor is written in floating-point version. Because the induction motor would be treated as the plant where the quantization errors due to numerical calculation should not be appeared in such model.

Figure 3.14: Simulated sensorless DFOC system of induction motor
To study the numerical accuracy among three data formats, the estimated speed responses and the corresponding q-axis reference currents are monitored as seen in Figure 3.15. Notice that all PI gains, parameters, and base quantities are set the same for all simulations of three data formats. In this figure, the step speed reference is set at 0.5 pu. It is clearly that the 16-bit fixed-point version fails to imitate the responses produced by floating-point version. As a result, the system performance and behavior during transient and steady-state would be affected when implementing in 16-bit fixed-point version. In addition, the estimated rotor flux angle, $\theta_\lambda$, is also monitored because it is very crucial for maintaining FOC condition.

Figure 3.15: Simulation result – estimated speed ($\hat{\omega}$) and q-axis reference current ($i^*_q$) using step speed reference of 0.5 pu among floating-point, 32-bit fixed-point, and 16-bit fixed-point formats
Figure 3.16 shows the estimated $\theta_\lambda$ from 16-bit fixed-point version also fails to imitate ones from floating-point version which is actually estimated very well, comparing with the actual one taken from “emulated” induction motor model (top graph in Figure 3.16). However, the 32-bit fixed-point version can produce the same plots as floating-point ones as seen in Figures 3.15 and 3.16. Thus, lets further examine the rotor flux calculations inside the flux estimator between 32-bit and 16-bit fixed-point. Figure 3.17 and 3.18 show the d-axis rotor flux produced by the current and voltage models of the flux estimator, respectively. In these figures, unlike $\lambda_{dr}^i$, the $\lambda_{dr}^v$ is not the same one between 32-bit and 16-bit fixed-point. Thus, the quantization errors likely come from the voltage model rather than the current model.

Figure 3.16: Simulation result – actual (top trace) and estimated rotor flux angles among floating-point (2nd trace), 32-bit fixed-point (3rd trace), and 16-bit fixed-point (4th trace) formats
Figure 3.17: Simulation result – estimated d-axis rotor flux produced by current model between 32-bit fixed-point (top trace) and 16-bit fixed-point (bottom trace) formats

Figure 3.18: Simulation result – estimated d-axis rotor flux produced by voltage model between 32-bit fixed-point (top trace) and 16-bit fixed-point (bottom trace) formats
Next, the investigations of the system performance and system behavior are studied by applying the step speed reference of ±0.5 pu. The inputs and outputs of three PI controllers for dq-axis currents and speed are monitored and plotted in Figures 3.19 through 3.21 for floating-point, 32-bit fixed-point, and 16-bit fixed-point formats, respectively. In these figures, it is apparent that the current and speed responses during both transient and steady-state are affected by the word length, especially in 16-bit fixed-point version. The 32-bit fixed-point plots can provide the same responses as floating-point ones. The following observations can be addressed for 16-bit fixed-point case when comparing with floating-point or 32-bit fixed-point cases. During transient, the higher oscillating responses of currents and speed with a longer settling time can be observed. Similarly, the responses are also oscillating at the higher magnitude at the steady-state. These quantization effects will be verified by the experimental results in next subsection.

Figure 3.19: Simulation result – floating-point format: dq-axis currents/voltages, and estimated speed (\( \hat{\omega} \)) using step speed reference of ±0.5 pu
Figure 3.20: Simulation result – 32-bit fixed-point format: dq-axis currents/voltages, and estimated speed ($\dot{\omega}_r$) using step speed reference of $\pm 0.5$ pu

Figure 3.21: Simulation result – 16-bit fixed-point format: dq-axis currents/voltages, and estimated speed ($\dot{\omega}_r$) using step speed reference of $\pm 0.5$ pu
3.6.2 Experimental results

The experimental system consists primarily of TMS320F2812 eZdsp and DMC1500 boards from Texas Instruments as a controller and voltage source inverter as seen in Figure 3.22. The interrupt/sampling/PWM frequency is set at 20 kHz with 150 MHz system clock frequency. Similarly, the highlighted blocks in this figure are realized in three data formats. However, since the DSP is fixed-point in nature, the floating-point arithmetics are inefficiently implemented by a run-time supporting library (i.e., rts2800_ml.lib). As a result, the real implementation of floating-point version requires a longer sampling period of 250 µsec. (4 kHz) in order to compute all floating-point modules. For fair comparison, the experimental results will be focused on only between 16-bit and 32-bit fixed-point versions because the system performances could be affected by different sampling period. If the sampling period is not an issue, the similar responses from floating-point version can be expected when comparing with ones from 32-bit fixed-point version as depicted in simulation results of previous section. Notice that all integral terms of PI controllers are always implemented in 32-bit word length even though in 16-bit fixed-point version.

Similar to simulation part in previous sub-section, the step speed reference of ±0.5 pu. is applied to the system and then the inputs and outputs of three PI controllers in both 16-bit and 32-bit fixed-point versions for dq-axis currents and speed are plotted in Figures 3.23 through 3.25, respectively. According to these figures, the responses are corresponding to the simulation ones during both transient and steady-state. The quantization effects can be verified in terms of higher oscillating and longer transient time.
Figure 3.22: Real implementation of a sensorless DFOC system of induction motor

Figure 3.23: Experimental result – reference speed ($\omega^*_r$), d-axis measured current ($i^*_d$), and d-axis reference voltage ($v^*_d$) using step speed reference of ±0.5 pu among (a) 16-bit fixed-point, and (b) 32-bit fixed-point formats
Figure 3.24: Experimental result - q-axis reference current ($i_{qs}^*$), q-axis measured current ($i_{qs}$), and q-axis reference voltage ($v_{qs}^*$) using step speed reference of $\pm 0.5$ pu among (a) 16-bit fixed-point, and (b) 32-bit fixed-point formats.

Figure 3.25: Experimental result - reference speed ($\omega_r^*$), estimated speed ($\hat{\omega}_r$), and q-axis reference current ($i_{qs}^*$) using step speed reference of $\pm 0.5$ pu among (a) 16-bit fixed-point, and (b) 32-bit fixed-point formats.
3.6.3 Conclusions

The effects of quantization errors in a DFOC system of induction motor have been investigated in a 32-bit fixed-point DSP, i.e., eZdsp2812 board. Three data formats of 16-bit fixed-point, 32-bit fixed-point, and floating-point have been implemented. Both simulations and experiments are carried out on the same DSP controller to verify the quantization effects in DFOC system of induction motor in terms of system performance and behavior. Based on simulation results, the current and speed responses produced by the 32-bit fixed-point version can imitate ones produced by floating-point version while they fail in 16-bit fixed-point version. Thus, the numerical accuracy may not be enough in 16-bit fixed-point version. Also, the current and speed responses have the higher oscillation and longer transient time as seen in both simulation and experimental results.
CHAPTER 4

FAST CURRENT CONTROL STRATEGY FOR SENSORLESS VECTOR CONTROLLED DRIVE OF AC MOTOR

In this chapter, the proposed fast current control technique is presented for a sensorless vector controlled drive of AC motor. The conventional PI controllers used in the current control loop of a vector controlled drive are also reviewed. Finally, the proposed fast current control strategy is implemented to validate its effectiveness. The simulation and experimental results of using this fast current control technique are extensively presented as well.

4.1 Introduction

In the vector controlled drive of AC motor, the dq-axis current control plays an important role of determining the overall system performance. Traditionally, the PI controller is employed for such current loops because it is easy to implement without any knowledge of machine model. However, one of the inherent problems of a PI controller is the output saturation causing the sluggish response, especially when the DC-
bus voltage is not sufficient or when the sudden load disturbance or sudden reference is applied. In this research work, a dq-axis current control technique for fast transient response in the sensorless vector controlled drive of AC motor is proposed. The Pontryagin’s maximum principle is primarily used to solve for the optimal-time voltage solution in the dq-axis current dynamical equations. Based on this principle, the technique can guarantee the dq-axis current transition from one point to another point with the minimum-time. Since the optimal-time voltages representing in the synchronously rotating dq-axis, this optimal-time controller can easily replace two PI controllers for the dq-axis current loops in the existing vector controlled drive system without modification of system blocks. Both simulation and experimentation were carried out to substantiate the proposed technique. The overall system was implemented in a 32-bit fixed-point digital signal processor based controller and power inverter from Texas Instruments (TMS320F2812 eZdsp and DMC1500 boards) as previously explained in Chapter 3.

4.2 Literature Reviews in Topic of Current Control Strategies

Since the DFOC drive performance is directly related to how well the dq-axis current regulations perform in the system, the relevant papers related to improvements of these two dq-axis current control loops are reviewed. Lorenz et al. [32-33] proposed two improvement techniques for dq-axis synchronous frame current control loops using PI controllers. Both techniques use the complex vector notation to reduce the complexity of a system, to be easily analyzed and designed by pole/zero cancellation. The cross-
coupling of $L\omega_c i_d$ and $L\omega_c i_q$ terms in q- and d-axis current control loops, respectively, is used in the first technique in order to allow the performance of current regulators independent to the synchronous frequency. The second technique is to modify the integral structure of PI controller so that the controller zero is approximately on the top of the plant pole. These techniques are essentially to move either plant pole to close to controller zero (in first technique) or controller zero to close to plant pole (in second technique). Their experimental results showed that the current regulations have significantly improved by using these two techniques compared with pure PI controllers. Their analysis and design had also generally extended to the synchronous reference frame dq-axis current control loops of IM model where the field oriented condition is not considered. Therefore, further analysis based on their techniques would be fruitful for dq-axis current control loops in a DFOC system of AC motors.

In [34], the parameter tuning using integral outputs of PI controllers in speed and q-axis current control loops is proposed. The waveforms of these integral outputs can give some information on the accuracy of the rotor time constant, stator inductance, and stator transient inductance used in the FOC controller. In the q-axis PI controller, two different feed-forward terms are used to tune for stator inductance, and stator transient inductance whereas the rotor time constant is tuned in the speed control loop. The interesting point of this paper is to show a usefulness of using a feed-forward term added with an output of PI controller to tune some machine parameters off-line.

Next, some selected publications focusing on the fast-response current control strategy in vector controlled drive of AC motors are reviewed [35-38]. S.-K. Sul et al. [35-36] proposed the minimum-time current controller of IM drive in the three-phase
PWM converter. In their approaches, an explicit minimum-time solution of the dq-axis current control for the vector controlled drive of IM is solved. The optimal control voltages are analytically derived by using optimal control theory. The time-optimal controller requires the rotor flux and speed information to calculate back EMF’s. In addition, the minimum time is also required to be calculated. Since this method needs heavy computation, their algorithms were implemented in the floating-point DSP based controller. Consequently, it may be impractical to implement in the fixed-point DSP based controller for low-cost applications.

In [37], a practical time-optimal current controller implemented in the fixed-point DSP is proposed. This controller is derived by using Pontryagin’s maximum principle for a sensored vector controlled drive of PMSM. The optimal control voltages in dq-axis are selected basing on the location of the dq-axis currents in the optimal switching diagram. However, this diagram has been developed with an assumption of stator resistance ignored. This assumption is realistic when PMSM is running at very high speed because the voltage drop across the stator resistance is not dominant in the back EMF equation. As for low speed, this assumption may not be valid and then this strategy has to include the stator resistance in analysis.

S.G. Bosga [38] also proposed a fast current control technique for a vector controlled drive of PMSM. In this technique, the reference dq-axis voltages are directly calculated from the dq-axis voltage equations of PMSM for the given reference dq-axis currents. This controller is actually the feed-forward structure. All machine parameters such as stator resistance, permanent-magnet flux, and dq-axis inductances which appear in these voltage equations have to be precisely known. To solve this problem, the
machine parameter updates are performed at a lower sampling rate in the software. To take care the machine saturation, the dq-axis inductances are updated by means of a look-up table as a function of motor currents. And the stator resistance and permanent-magnet flux are also updated according to the temperature. Although this technique is simple and straightforward, a temperature sensor is needed for update of the stator resistance and permanent-magnet flux. Imprecise information of any machine parameters could directly deteriorate current control performance by computing the incorrect reference dq-axis voltages.

4.3 Reviews of Conventional PI Controllers

In the FOC condition, the stator current dynamics are similar to each other for both IM and PMSM as seen in Equations (2.1) and (2.2) for IM and Equations (2.8) and (2.9) for PMSM. From these equations, one could clearly view each current loop dominated by the simple first-order plant with the $d_{qs}^e$ and $d_{ds}^e$ terms as shown in Figures 4.1 and 4.2. As a result, the PI controller should be sufficient to control the current control loops of FOC system which is characterized by the first order system [39].

For IM, the rotor flux dynamic is usually much slower than the $i_{ds}^e$ dynamic (i.e., typically, $1/\alpha >> 1/\gamma$). Therefore, the rotor flux could be assumed constant while the d-axis current is well controlled. Thus, the dq-axis current loops of IM shown in Figure 4.1 can be approximate as shown in Figures 4.3.
Figures 4.1: Two current control loops of IM (a) q-axis (b) d-axis

Figures 4.2: Two current control loops of PMSM (a) q-axis (b) d-axis
Figures 4.3: Approximate two current control loops of IM when $\frac{1}{\alpha} \gg \frac{1}{\gamma}$

(a) $q$-axis (b) $d$-axis

A general negative feedback loop system including a PI controller, a first-order plant with a disturbance can be presented for the $dq$-axis current loops in the rotor flux oriented condition of IM and PMSM as seen in Figures 4.4. The open-loop transient responds are described by the time constant ($\tau$) and the gain ($K$) for IM and PMSM as follows:

- $\tau = \frac{1}{\gamma}$ and $K = \frac{1}{\sigma \gamma L_s}$, for IM.
- $\tau = \frac{1}{\gamma}$ and $K = \frac{1}{\gamma L_s}$, for PMSM.

where $\sigma$, and $\gamma$ for IM and PMSM are defined in Appendix A.
If the output, \( Y(s) \), due to \( D(s) \), has assumingly been minimized by properly increasing \( K_i = K_p / T_i \) gain of PI controller, then the dominant dynamics of this feedback system could approximately be represented by the closed-loop transfer function from reference, \( R(s) \) to \( Y(s) \), with \( D(s) = 0 \) as follow.

\[
T(s) = \frac{Y(s)}{R(s)} = \frac{KK_p(s + 1/T_i)}{\tau s^2 + (1 + KK_p)s + KK_i}
\]

(4.1)

that gives a zero at \( s = -1/T_i \) and two poles at \( s = \frac{-(1 + KK_p) \pm j\sqrt{4\tau KK_i - (1 + KK_p)^2}}{2\tau} \).

Therefore, the closed-loop poles (which primarily determine the characteristics of closed-loop responses) can be located anywhere in the s-plane by independently adjusting \( K_p \) and \( T_i \) gains of PI controller. In other words, the responding time of current control can be successfully designed to meet any desired specifications. The time-domain dq-axis current step responses of the second-order characteristic equation in (4.1) can be summarized in terms of the machine parameters and PI gains in Table 4.1. Notice that two poles can also be generally expressed in terms of the damping ratio and natural angular velocity as \( s = -\zeta \omega_n \pm j\omega_n \sqrt{1 - \zeta^2} \) where \( \zeta < 1 \).

Figures 4.5 and 4.6 conveniently show the effects of PI gains to \( \zeta \) and \( \omega_n \), which in turn characterize the step responses. Then, Figures 4.7 and 4.8 show the simulated step response to a reference of 0.5 pu with different \( K_p \) and \( T_i \).
<table>
<thead>
<tr>
<th>Step response characteristics</th>
</tr>
</thead>
</table>
| Damping ratio, $\zeta$ | $\frac{1 + KK_p}{2\sqrt{\pi KK_i}}$  
| Natural angular velocity, $\omega_n$ (rad/sec) | $\sqrt{\frac{KK_i}{\tau}}$  
| Settling time to 2% error, $t_s$ (sec) | $\approx \frac{4}{\zeta\omega_n}$  
| Peak time, $t_p$ (sec) | $\frac{\pi}{\omega_n \sqrt{1 - \zeta^2}}$  
| Percent overshoot, PO (%) | $100e^{-\zeta\pi/\sqrt{1-\zeta^2}}$  

Table 4.1: Step response characteristics of dq-axis current control

![Figure 4.5: Damping ratio, $\zeta$, for different $K_p$ and $T_i$](image)
Figure 4.6: Natural angular velocity, $\omega_n$, for different $K_p$ and $T_i$

Figure 4.7: Step responses of dq-axis current control for different $K_p$ when $T_i = 5e-04$ sec
As seen in Figure 4.5, it is apparent that $\zeta$ increases when either $K_p$ or $T_i$ increases, so the resulting PO is reduced as observed in Figures 4.7 and 4.8. In Figure 4.6, $\omega_n$ is significantly increased as $T_i$ decreased. Thus, $\xi$ can be reduced by decreasing $T_i$ but natural frequency will be accordingly increased as seen in Figure 4.8.

In the vector controlled drive of AC motor, the desired fast characteristics of the dq-axis current responses are limited by PI output saturation due to a limited DC-bus voltage. It is unavoidable when the dq-axis currents controlled by regular PI controllers are required to be fastest as possible. In practice, the saturated outputs of PI controller for the dq-axis current loops will be produced with a constraint of $\sqrt{v_{d,pu}^2 + v_{q,pu}^2} \leq 1$. For $K_p = 4$ and $T_i = 5e-04$ sec, if the PI output is limited at 0.5 and 0.2 pu, then the step responses will be slowing down as the saturation level reduced. Figure 4.9 clearly shows this effect of PI output saturation to the step responses using $K_p = 4$ and $T_i = 5e-04$ sec (i.e., fast PI
gain). Note that the saturation level must be larger than the required PI output that successfully brings the current feedback to its reference value.

![Figure 4.9: Effects of PI output saturation slowing down the responses](image)

To avoid PI output saturation, the PI gains must be properly selected such that the responding time of current is not too fast. The responding time could approximate in an order of the motor time constant (i.e., \( \tau = l/\gamma = 0.0047 \) sec for this particular motor). Figure 4.10 shows the step responses using \( K_p = 0.4 \) and \( T_i = 5e-03 \) sec which its settling time is about the motor time constant (0.0047 sec). This selection of \( T_i \) is also suggested by [33]. It is apparent that the PI output is not saturated in this case. In reality, the saturation on the q-axis current loop would be affecting the performance of the d-axis current loop, or vice versa because of the coupled dynamic of dq-axis current. In practice, the gains of both PI controllers for dq-axis current loops are conveniently set the same values.
4.3.1 Rule of thumb for the effects of PI gains

Generally, when the $K_p$ and $T_i$ of a PI controller have increased, then the characteristics of the step transient responses to these gain increases will be affected accordingly as summarized in Table 4.2.

<table>
<thead>
<tr>
<th>Gain increased</th>
<th>Rise time</th>
<th>Settling time</th>
<th>Overshoot</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_p$</td>
<td>decrease</td>
<td>decrease</td>
<td>increase</td>
<td>decrease</td>
</tr>
<tr>
<td>$T_i$</td>
<td>increase</td>
<td>increase</td>
<td>decrease</td>
<td>increase</td>
</tr>
</tbody>
</table>

Table 4.2: Effects of increasing PI gains to the transient responses

In the next subsections, the simulation and experimental results of the transient current responses of IM and PMSM will be presented in terms of different PI gains. The effects summarized in Table 4.2 would clearly be seen from these results.
4.3.2 Computer simulation

In this subsection, the typical dq-axis current control loop system using PI controllers in a sensorless vector controlled drive system of AC motor are simulated. The simulation system uses estimated rotor flux angle from flux estimation explained in Chapter 2 for variable transformations (i.e., Park and inverse Park transformations). The effects of different PI gains to the current responses are illustrated by the simulation results in Figures 4.11 (for IM) and 4.12 (for PMSM). In these figures, the dq-axis current responses and dq-axis voltages are shown. The reference d-axis currents are 0.2 pu for IM and 0 pu for PMSM. For both IM and PMSM, the reference q-axis current are suddenly changed from -0.4 pu to 0.4 pu. As seen from these figures, the reducing $T_i$ causes the natural angular velocity, $\omega_n$, be increased. The longer settling time, $t_s$, and the lower percent overshoot, PO, are also expected as $T_i$ increased. Although $T_i$ plays an important role to determine the dynamic of current responses, the increasing $K_p$ can cause $t_s$ shorter. These simulation results clearly show that the behaviors of dq-axis current responses between IM and PMSM are similar in the vector controlled drive system. The behaviors of current responses due to different PI gains are corresponding to the ones simulated by the single-input-single-output (SISO) feedback system shown in Figure 4.4 (when $D(s)$ is assumed to be zero). In next subsection, the experimental results for current closed loops based on the sensorless vector control drive system of AC motor using the same flux estimation will be represented.
Figure 4.11: Simulation result - dq-axis current responses of IM using PI controllers with different $K_p$ and $T_i$.
$K_p = 1.0$

$K_p = 0.75$

$K_p = 0.5$

$T_i = 0.0002 \text{ sec}$

$T_i = 0.00006 \text{ sec}$

Figure 4.12: Simulation result - dq-axis current responses of PMSM using PI controllers with different $K_p$ and $T_i$
4.3.3 Experimental results

To validate the behavior of current responses due to different PI gains, the system seen in Figure 4.13 is implemented for both IM and PMSM. Experimental results are organized as follows. Figures 4.14 and 4.15 show the q-axis and d-axis current responses of IM, respectively and Figures 4.16 and 4.17 show the q-axis and d-axis current responses of PMSM, respectively. Similarly, the reference d-axis currents are 1 A for IM and 0 A for PMSM. For both IM and PMSM, the reference q-axis current are suddenly changed from -2 A to 2 A. As seen from these figures, the dq-axis current responses with different PI gains behave similarly in terms of PO, ts, and ωn, comparing with the simulation results in Figures 4.11 and 4.12. The current responses could be drastically poor when Ti is not in the proper range. Also, the small Kp could cause ts longer. Typically, the Ti would be selected according to the motor time constant.

![Diagram](image)

**Figure 4.13**: Overall current control loop system using PI controllers
Figure 4.14: Experimental result - q-axis current responses of IM using PI controllers with different $K_p$ and $T_i$
Figure 4.15: Experimental result - d-axis current responses of IM using PI controllers with different $K_p$ and $T_i$
Figure 4.16: Experimental result - q-axis current responses of PMSM using PI controllers with different $K_p$ and $T_i$
Figure 4.17: Experimental result - d-axis current responses of PMSM using PI controllers with different $K_p$ and $T_i$
4.4 Proposed Fast-Transient Current Control Strategy

4.4.1 Induction motor

When controlling IM in the field oriented control, the dq-axis current dynamics seen in Equations (2.1) and (2.2) can be compactly rewritten in a matrix form as follow:

\[
\begin{bmatrix}
\frac{di^e_{qs}}{dt} \\
\frac{di^e_{ds}}{dt} \\
\frac{di^e}{dt}
\end{bmatrix} =
\begin{bmatrix}
\gamma & -\omega_e & i^e_{qs} \\
\omega_e & -\gamma & i^e_{ds} \\
\omega_e & -\gamma & i^e
\end{bmatrix} +
\begin{bmatrix}
-\beta \omega_e \lambda^e_{dr} \\
\beta \alpha \lambda^e_{dr}
\end{bmatrix} +
\begin{bmatrix}
\beta_i \v^e_{qs} \\
\v^e_{ds}
\end{bmatrix}
\]

(4.1)

Then, the matrix can be further simplified to the following form:

\[
\dot{x} = Ax + u
\]

(4.2)

where \( x = \begin{bmatrix} i^e_{qs} \\ i^e_{ds} \end{bmatrix}, A = \begin{bmatrix} -\gamma & -\omega_e \\ \omega_e & -\gamma \end{bmatrix}, \) and \( u = \begin{bmatrix} u^e_{qs} \\ u^e_{ds} \end{bmatrix} = \begin{bmatrix} -\beta \omega_e \lambda^e_{dr} \\ \beta \alpha \lambda^e_{dr} \end{bmatrix} + \begin{bmatrix} \beta_i \v^e_{qs} \\ \v^e_{ds} \end{bmatrix}. \)

Note that the actual control voltage for the inverter is \( v = \begin{bmatrix} v^e_{qs} \\ v^e_{ds} \end{bmatrix}. \) It is also assumed that the synchronous speed, \( \omega_e, \) is slowly varied during the current transition.

Now we are trying to find the control \( u \) that forces the dq-axis current asymptotically stable from an arbitrary initial value to the origin with the minimum time, \( t_f. \) Therefore, the problem can be formulated as

\[
\text{Minimize} \quad t_f
\]

(4.3)

Lets begin with the optimal-time formulation, forming the Hamiltonian function and introducing the co-state \( \psi, \)

\[
H(\psi, x, u) = \psi x = \psi(Ax + u)
\]

(4.4)

The necessary conditions for the minimum-time optimal problem are
\[ x^* = \frac{\partial H}{\partial \psi} = Ax^* + u^* \quad (4.5) \]
\[ \psi = -\frac{\partial H}{\partial x} = -\psi A \quad (4.6) \]

where \( x^* (t) \) is the optimal path with \( t \in [t_0, t_f] \).

According to Pontryagin’s maximum principle [40], the optimal control \( u^* \) that minimizes the objective function should satisfy the following condition (4.7) and such control \( u^* \) is of the form seen in (4.8).

\[
H(\psi(t), x^*(t), u^*(t)) = \max \left\{ H(\psi, x, u); \|u\|^2 \leq U_{\text{max}}^2 \right\} \\
= \max \left\{ |\psi(Ax + u)|; \|u\|^2 \leq U_{\text{max}}^2 \right\} 
\]

\[
u^*(t) = U_0 \text{sign} (\psi(t)) = \begin{cases} +U_0 & \text{if } \psi(t) > 0 \\ -U_0 & \text{if } \psi(t) < 0 \end{cases} \quad (4.8)
\]

where \( U_0 > 0 \) with the constraints of \( u_{ds}^{e*} + u_{qs}^{e*} \leq U_{\text{max}}^2 \) and the solution of the co-state Equation (4.6) is as

\[
\psi(t) = \psi(0)e^\gamma \begin{bmatrix} \cos \omega_c t & \sin \omega_c t \\ -\sin \omega_c t & \cos \omega_c t \end{bmatrix} \quad (4.9)
\]

In fact, the optimal control in (4.8) has been also derived in [41-45]. According to (4.8) and (4.9), it is clearly that the optimal control \( u^* \) is switched between \( \pm U_0 \), depending on the sign of co-state in every half period of \( \omega_c \) (i.e., \( \pi/\omega_c \) second). It is also interesting that the co-state is an unstable variable with an increasing exponential and a sinusoid. As a result, the optimal control satisfying (4.8) would be in one of four possible cases as follows:

1. \( u_{ds}^{e*} = +U_0 \) and \( u_{qs}^{e*} = +U_0 \) \quad (4.10)
2. \( u_{ds}^e = +U_0 \) and \( u_{qs}^e = -U_0 \) \hspace{1cm} (4.11)

3. \( u_{ds}^e = -U_0 \) and \( u_{qs}^e = +U_0 \) \hspace{1cm} (4.12)

4. \( u_{ds}^e = -U_0 \) and \( u_{qs}^e = -U_0 \) \hspace{1cm} (4.13)

In practice, the unstable co-state \( \psi(t) \) may not be calculated in order to find the optimal control in (4.8). Instead, the optimal control \( u^* \) is conveniently selected according to the dynamics of the currents. The current dynamics are solved by substituting these optimal controls (4.10)-(4.13) back into the original motor equation; \( x = Ax + u \). In Appendix B, the solution of dq-axis current dynamics is provided in Equation (B.4). The solution can also be expressed in another form as below:

\[
\frac{[i_{qs}^e(t) + I_{qs}^e]^2 + [i_{ds}^e(t) + I_{ds}^e]^2}{e^{-\gamma t}} = e^{-\gamma t} \left( [i_{qs}^e(0) + I_{qs}^e]^2 + [i_{ds}^e(0) + I_{ds}^e]^2 \right) 
\]  

(4.14)

where \( I_{ds}^e = \frac{1}{\gamma^2 + \omega_e^2} \left( -\gamma u_{ds}^e(t) - \omega_e u_{qs}^e(t) \right) \) and \( I_{qs}^e = \frac{1}{\gamma^2 + \omega_e^2} \left( -\gamma u_{qs}^e(t) + \omega_e u_{ds}^e(t) \right) \).

Thus, the state trajectory in the phase-plane is a spiral rotating in clock-wise direction and having a center at \( -I_{ds}^e, -I_{qs}^e \) with a decayed radius, \( e^{-\gamma t} \sqrt{[i_{qs}^e(0) + I_{qs}^e]^2 + [i_{ds}^e(0) + I_{ds}^e]^2} \).

In fact, the steady state value of current dynamics is at a center \( -I_{ds}^e, -I_{qs}^e \).

To find the optimal controls in the phase-plane, let beginning with the simplified case of resistance neglected (i.e., \( \gamma = 0 \)), when resistance neglected as seen in Equation (B.7), the solution can similarly be expressed in another form as below:

\[
\frac{[i_{qs}^e(t) + I_{qs}^e]^2 + [i_{ds}^e(t) + I_{ds}^e]^2}{e^{-\gamma t}} = \left( [i_{qs}^e(0) + I_{qs}^e]^2 + [i_{ds}^e(0) + I_{ds}^e]^2 \right) 
\]  

(4.15)
where $I_{ds}^e = \frac{-u_{qs}^e(t)}{\omega_e}$ and $I_{qs}^e = \frac{u_{ds}^e(t)}{\omega_e}$.

The state trajectory is simply a circle rotating in the clock-wise direction and having the same center at $(-I_{ds}^e, -I_{qs}^e)$ with a radius of $\sqrt{[I_{qs}^e(0) + I_{qs}^e]^2 + [I_{ds}^e(0) + I_{ds}^e]^2}$. This state dynamics system is so-called as the frictionless or conservative system. Figure 4.18 shows the state trajectories with different initial conditions when applying the optimal controls in (4.10) through (4.13) and neglecting resistance. As synchronous speed increases, the center of circle moves toward the origin but it never across to different quadrants. The radius of circle is determined by the initial condition of currents. In this case, by simply looking at the state trajectories in Figure 4.18, the minimum-time control diagram in dq-axis current plane can easily be realized as shown in Figure 4.19. The switching curves consist of a series of the partial curves of circle with a radius of $\sqrt{I_{qs}^e + I_{ds}^e}$. Finally, the optimal controls $u_{qs}^e$ and $u_{ds}^e$ have the following properties:

- They must switch their values between $+U_0$ and $-U_0$, depending on the sign of co-state as seen in (4.8).
- They must remain constant for every $\pi/\omega_e$ second because co-state is an exponentially sinusoid with a frequency of $\omega_e$.
- There is no upper bound of the number of switchings for the control optimal control $u^*$.
- There is no possibility of the singular control because the co-state never be zero for any time interval where the control optimal control $u^*$ is not defined. Therefore, this time-optimal problem is called “normal”.

82
Figure 4.18: State trajectories with different initial conditions when applying the optimal controls (resistance neglected)
Next, the optimal control diagram in case of the resistance included can be constructed in the similar fashion. Also, four properties of optimal control are still valid. However, the state trajectories are different because the trajectory path is spiral shape. Figure 4.20 shows the state trajectories when applying four possible optimal control voltages. As a result, the switching curves for optimal voltages will be a series of partial curves of spirals rather than ones of circles shown in the case of resistance neglected. In addition, the switching curves are not aligned on the top of the fixed x-y coordinates. Instead, they will be aligned on top of the movable coordinates with a slope of $\gamma/\omega_c$ as seen in Figure 4.21. Because the center of the spiral has to be located at an equal distance on the x- and y- axis of this movable coordinates. This optimal control diagram would be
exactly the same as one seen in Figure 4.19 when the resistances are neglected (i.e., \( \gamma = 0 \)).

(a) \( u_{ds}^* = +U_0 \) and \( u_{qs}^* = +U_0 \)

(b) \( u_{ds}^* = +U_0 \) and \( u_{qs}^* = -U_0 \)

(c) \( u_{ds}^* = -U_0 \) and \( u_{qs}^* = +U_0 \)

(d) \( u_{ds}^* = -U_0 \) and \( u_{qs}^* = -U_0 \)

Figure 4.20: State trajectories with different initial conditions when applying the optimal controls (resistance included)
Likewise, the optimal-time control $u^*$ for the vector-controlled drive of PMSM can be developed in the same fashion as IM because of their similar equations of current dynamics. The dq-axis current dynamics seen in Equations (2.8) and (2.9) can be also compactly rewritten in a matrix form as follow:

$$\begin{bmatrix}
\frac{di^e_{qs}}{dt} \\
\frac{di^e_{ds}}{dt}
\end{bmatrix} = \begin{bmatrix}
-\gamma & -\omega_c \\
\omega_c & -\gamma
\end{bmatrix} \begin{bmatrix}
i^e_{qs} \\
i^e_{ds}
\end{bmatrix} + \begin{bmatrix}
-\beta \omega_c \lambda_m \\
0
\end{bmatrix} + \begin{bmatrix}
v^e_{qs} \\
v^e_{ds}
\end{bmatrix}$$

(4.16)
Then, the matrix can be further simplified to the following form:

\[ \dot{x} = Ax + u \tag{4.17} \]

where \( x = \begin{bmatrix} i_{qs} \\ i_{ds} \end{bmatrix} \), \( A = \begin{bmatrix} -\gamma & -\omega_e \\ \omega_e & -\gamma \end{bmatrix} \), and \( u = \begin{bmatrix} u_{qs} \\ u_{ds} \end{bmatrix} = \begin{bmatrix} -\beta \omega_e \lambda_m \\ 0 \end{bmatrix} + \begin{bmatrix} v_{qs} \\ v_{ds} \end{bmatrix} \).

Note that the actual control voltage for the inverter is \( v = \begin{bmatrix} v_{qs} \\ v_{ds} \end{bmatrix} \). Similarly, it is also assumed that the synchronous speed, \( \omega_e \), is slowly varied during the current transition.

Since the current dynamic matrix Equation (4.17) is same as one of IM (4.2), the analysis of the optimal control will be the same as well. Only difference is that the control \( u \) includes the different terms of back EMF’s.

### 4.4.3 Error dynamics of currents and real optimal control for inverter

In previous sections, the optimal control driving an initial state, \( x(t_0) \) to final state, \( x(t_f) \) at origin with the minimum time is found. The phase plane representing the current trajectories for different optimal controls is also depicted. In general, the final state \( x(t_f) \), however, could be any arbitrary value. The objective of this subsection is to find the optimal control driving to a non-zero arbitrary \( x(t_f) \). Lets define the current error as \( e = x - x(t_f) \) where \( x \) is the real currents and \( x(t_f) \) is the final currents (or constant references).

Then, taking derivative \( e = x - x(t_f) \), yields

\[ \ddot{e} = x - x(t_f) = x \tag{4.18} \]

\[
\begin{bmatrix}
\frac{de_{qs}}{dt} \\
\frac{de_{ds}}{dt}
\end{bmatrix} = \begin{bmatrix}
-\gamma & -\omega_e \\
\omega_e & -\gamma
\end{bmatrix}
\begin{bmatrix}
e_{qs} \\
e_{ds}
\end{bmatrix} + \begin{bmatrix}
e_{qs} + i_{qs} \tau(t) \\
e_{ds} + i_{ds} \tau(t)
\end{bmatrix} + \begin{bmatrix} u_{qs} \\ u_{ds} \end{bmatrix} \tag{4.19}
\]
Equivalently,

\[ \begin{bmatrix} \frac{d e^e_{qs}}{dt} \\ \frac{d e^e_{ds}}{dt} \end{bmatrix} = \begin{bmatrix} -\gamma & -\omega_e \\ \omega_e & -\gamma \end{bmatrix} \begin{bmatrix} e^e_{qs} \\ e^e_{ds} \end{bmatrix} + \begin{bmatrix} -\gamma & -\omega_e \\ \omega_e & -\gamma \end{bmatrix} \begin{bmatrix} i^e_{qs}(t_f) \\ i^e_{ds}(t_f) \end{bmatrix} + \begin{bmatrix} u^e_{qs} \\ u^e_{ds} \end{bmatrix} \] \tag{4.20}

Therefore, the error dynamics are

\[ \dot{e} = Ae + u_e \] \tag{4.21}

where

\[ u_e = \begin{bmatrix} u^e_{c,qs} \\ u^e_{c,ds} \end{bmatrix} = \begin{bmatrix} -\gamma & -\omega_e \\ \omega_e & -\gamma \end{bmatrix} \begin{bmatrix} i^e_{qs}(t_f) \\ i^e_{ds}(t_f) \end{bmatrix} + \begin{bmatrix} u^e_{qs} \\ u^e_{ds} \end{bmatrix} \] \tag{4.22}

\[ \begin{bmatrix} u^e_{q} \\ u^e_{d} \end{bmatrix} = \begin{bmatrix} -\beta \omega \lambda^e_{dr} \\ \beta \lambda^e_{dr} \end{bmatrix} + \beta e \begin{bmatrix} v^e_{qs} \\ v^e_{ds} \end{bmatrix} \] for IM \tag{4.23}

\[ \begin{bmatrix} u^e_{q} \\ u^e_{d} \end{bmatrix} = \begin{bmatrix} -\beta \omega \lambda^e_{m} \\ 0 \end{bmatrix} + \beta e \begin{bmatrix} v^e_{qs} \\ v^e_{ds} \end{bmatrix} \] for PMSM

Notice that the error dynamics (4.21) are valid for both IM and PMSM. As a result, the real optimal control for inverter \( v^* = \begin{bmatrix} v^e_{qs} \\ v^e_{ds} \end{bmatrix} \) can be derived backward from Equations (4.22) and (4.23) once the optimal control \( u^*_e \) in error dynamics (4.21) are found. Such real optimal control is

\[ v^* = \begin{bmatrix} v^e_{qs} \\ v^e_{ds} \end{bmatrix} = \begin{bmatrix} 1 \\ -\beta \end{bmatrix} \begin{bmatrix} v^e_{i} \\ -\beta \omega \lambda^e_{dr} + u^e_{e,qs} \end{bmatrix} \] for IM \tag{4.24}

\[ v^* = \begin{bmatrix} v^e_{qs} \\ v^e_{ds} \end{bmatrix} = \begin{bmatrix} 1 \\ -\beta \end{bmatrix} \begin{bmatrix} v^e_{i} \\ -\beta \omega \lambda^e_{m} + u^e_{e,qs} \end{bmatrix} \] for PMSM

where \( v^e_{i} = \begin{bmatrix} \gamma^e_{qs}(t_f) + \omega_e i^e_{ds}(t_f) \\ \gamma^e_{ds}(t_f) - \omega_e i^e_{qs}(t_f) \end{bmatrix} \).

Note that the \( \lambda^e_{dr} \) in (4.24) for IM could be replaced by \( L_m i^e_{ds} \).
4.5 Simulation Results

4.5.1 Current control loops

The block diagram of the overall current control loop of a sensorless vector controlled system using either PI controllers or time-optimal controllers can be seen in Figure 4.22. In this figure, the PI controllers, which are conventionally used for d- and q-axis current control loops, can be replaced by the proposed time-optimal controller explained in the previous section. The sensorless vector controlled drive system using either PI controllers and time-optimal controller is simulated basing on PMSM. The reference d- and q-axis currents are 0 pu and step changes from -0.4 pu to 0.4 pu, respectively. Figures 4.23 and 4.24 show the dq-axis current responses and their voltages for PI controllers and time-optimal controller, respectively.

![Figure 4.22: Overall current control loop of sensorless vector controlled system using either PI controllers or time-optimal controllers](image-url)
Figure 4.23: Simulation result - dq-axis current and voltage responses using PI controllers

Figure 4.24: Simulation result - dq-axis current and voltage responses using time-optimal controller
In Figure 4.24, the dq-axis current responses using time-optimal controller are obviously faster, compared with ones using PI controllers in Figure 4.23. Even though both PI controllers do not produce the saturated dq-axis voltages, but they cannot provide the fast current responses as the time-optimal controller performed. The dominant current dynamics are governed by the two single feedback systems with coupling to each other (see Figures 4.1 and 4.2), thus the current responses are expected to be exponentially decayed sinusoid with a natural frequency and damping ratio, which rely on PI gains.

4.5.2 Speed and current control loops

Now the speed control loop is added in the system. The conventional PI controller is employed for the speed regulation. The output of this speed PI controller is presented as the q-axis current reference, i.e., torque command. The block diagram of the overall speed control loop of a sensorless vector controlled system can be seen in Figure 4.25. This system is simulated basing on PMSM when the reference speed is a square wave between -0.4 pu and 0.4 pu. Figure 4.26 shows the speed and dq-axis current responses using PI controllers when applying the full DC-bus voltage. Clearly, the dq-axis voltages are not saturated because of the sufficient DC-bus voltage. In Figure 4.27, the speed and dq-axis current responses are simulated when the DC-bus voltage is reduced to about 20% of its full value. In this case, the q-axis voltage is saturated as seen from thick curves in the figure. The dq-axis currents are obviously lost in control during this saturation time even though the integration anti-windup is trying to reduce this saturation effect.

In practice, the saturation effect cannot be avoided in a system employing the regular PI controllers especially when the sudden changes of the reference q-axis current
happen in a system with the insufficient DC-bus voltage. The proposed time-optimal controller can be used to avoid such a situation. Now both PI controllers for dq-axis current in Figure 4.25 are replaced by the time-optimal controller. By using the same amount of DC-bus voltage (20% of its full value), the dq-axis voltages are not saturated when the time-optimal controller is used as depicted in Figure 4.28. The current responses are not lost in control as well. So far, two important benefits of this time-optimal controller over the conventional PI controllers revealed by simulation results are as follows:

- With an enough DC-bus voltage, the current responses are much faster (much shorter settling time).
- With an insufficient DC-bus voltage, the time-optimal controller does not produce the saturated dq-axis voltages which occur in PI controllers.

![Diagram](image)

Figure 4.25: Overall speed control loop system using PI controllers
Figure 4.26: Simulation result – speed and dq-axis current responses using PI controllers without voltage saturation (full DC-bus voltage)

Figure 4.27: Simulation result – speed and dq-axis current responses using PI controllers with voltage saturation (insufficient DC-bus voltage)
4.6 Experimental Results

4.6.1 Current control loops

In this subsection, the current control system in Figure 4.22 is examined. Similar to the simulation part, the reference d-axis currents are set at 1 A for IM and 0 A for PMSM. For both IM and PMSM, the reference q-axis current are a step change from -2 A to 2 A. Figures 4.29 and 4.30 show the dq-axis current and voltage responses of IM and PMSM, respectively. In each figure, current responses from PI controller (at left column) are compared with ones from time-optimal controller (at right column). As expected, the time-optimal controller performs much better than PI controllers for both IM and PMSM. In addition, the d-qaxis current is still well regulated during the reference q-axis current which is suddenly changed from -2 A to 2 A.
Figure 4.29: Experimental result - dq-axis current responses of IM (a)-(b) using PI controllers (left column), (c)-(d) using time-optimal controller (right column)
Figure 4.30: Experimental result - dq-axis current responses of PMSM (a)-(b) using PI controller (left column), (c)-(d) using time-optimal controller (right column)
4.6.2 Speed and current control loops

Now the sensorless vector controlled drive system in Figure 4.25 is implemented. Experimental results are organized as follows. Figures 4.31 show the dq-axis current/voltage responses of IM using PI controllers when the DC-bus voltage is at full value of 320 volt (left column of figure) and at about 20% of full value (right column of figure). Next, Figure 4.32 shows the dq-axis current/voltage responses of IM using PI controllers (left column of figure) and time-optimal controller (right column of figure) when applying the DC-bus voltage about 20% of full value. For PMSM, Figures 4.33 and 4.34 are represented in similar format as seen Figures 4.31 and 4.32 (for IM), respectively. In each figure, the top, middle, and bottom rows show the speed/q-axis current, q-axis current/voltage, and d-axis current/voltage responses, respectively.

In Figures 4.31 (for IM) and 4.33 (for PMSM), by using PI controllers, the dq-axis voltage responses are obviously not saturated when the DC-bus voltage is at full value. Since the required q-axis voltage is increasing because of the reduced DC-bus voltage, PI controllers could produce the voltage saturation. Then, the dq-axis currents are lost in control during the saturation time.

In Figures 4.32 (for IM) and 4.34 (for PMSM), when applying the insufficient DC-bus voltage, the time-optimal controller can effectively control the dq-axis currents without voltage saturation as found in PI controllers. However, the speed responses look similar when currents are regulated by PI and time-optimal controllers as seen on the top row of these figures. Since the mechanical time constant is usually larger than electrical time constant, the speed responses react much slower than current responses even though currents are lost in control during saturation time.
Figure 4.31: Experimental result – speed and dq-axis current responses of IM using PI controllers, (a)-(c) full DC-bus voltage (left column), (d)-(f) insufficient DC-bus voltage (right column)
Figure 4.32: Experimental result – speed and dq-axis current responses of IM with insufficient DC-bus voltage, (a)-(c) PI controllers (left column), (d)-(f) time-optimal controller (right column)
Figure 4.33: Experimental result – speed and dq-axis current responses of PMSM using PI controllers, (a)-(c) full DC-bus voltage (left column), (d)-(f) insufficient DC-bus voltage (right column)
Figure 4.34: Experimental result – speed and dq-axis current responses of PMSM with insufficient DC-bus voltage, (a)-(c) PI controllers (left column), (d)-(f) time-optimal controller (right column)
4.7 Conclusions

In this chapter, the PI controllers for the dq-axis current regulation were reviewed and the proposed time-optimal current controller was derived. The structure of this time-optimal controller is simple and easy to implement because the optimal controls are mainly obtained by means of checking signs of current errors. In addition, this controller can easily replace two PI controllers in a typical vector controlled drive system of AC motors. Both simulation and experimental results were presented to validate the technique. This time-optimal controller can achieve two following key benefits which PI controllers cannot perform:

- The fast current responses are guaranteed. No overshoot of current responses can be seen when time-optimal controller is employed, resulting no sluggish current responses.

- With insufficient DC-bus voltage, the time-optimal controller can regulate dq-axis current without voltage saturation.
CHAPTER 5

CONCLUSIONS

This section summarizes the possible contributions from this dissertation and provides ideas for future research in this area.

5.1 Summary

- The DFOC system is applicable for both IM and PMSM without changing software codes. Therefore, the controller is flexible to use with either IM or PMSM with minimum time of software configurations.

- Commercially available, 32-bit fixed-point TMS320LF2812 DSP is employed to implement the controller. The numerical accuracy is much improved by using this 32-bit fixed-point DSP, compared with conventional 16-bit fixed-point DSP. The “C” language is used for coding.

- The current measurements are obtained by an inexpensive viewing resistor technique, which eliminates more expensive and typically used current sensors.
• Compared with direct measurement of the terminal voltages, the phase voltage reconstruction eliminates the external low-pass filters, by using only the measured DC-bus voltage and switching functions.

• Quantization effects related to the fixed-point DSP implementation are addressed. According to both simulation and experimental results, the performance of the overall system is affected by the number of bits of DSP (e.g., 16-bit versus 32-bit).

• Finally, the development of a time-optimal current controller in a vector controlled drive of AC motors is accomplished on a 32-bit fixed-point DSP based controller. As shown, the fast current responses are achieved without voltage saturations under a limited DC-bus voltage. The principle of this proposed technique is validated by both simulation and experimental results in a sensorless DFOC system for both IM and PMSM.

5.2 Future Research Suggestions

• For the sensorless DFOC of both IM and PMSM, a field weakening algorithm is required for high speed operation under a full DC-bus voltage.

• The time-optimal controller should be examined under the field weakening region where a full DC-bus voltage is applied.

• More indepth analysis of the theoretical aspects of quantization errors in the digital motor control area should be investigated.
• For very low-speed ranges, the accuracy of flux and speed estimations should be investigated as well as the time-optimal controller under a constrained DC-bus voltage.
APPENDIX A

MATHEMATICAL MODEL OF AC MOTOR

A.1 Synchronously Rotating Reference Frame Model

A.1.1 Induction motor [46-47]

The mathematical model of 3-phase IM in a synchronously rotating reference frame ($\omega = \omega_e$) can be described as the following equations:

\[
\frac{di_{qs}^e}{dt} = -\gamma_{qs}^e - \omega_e i_{qs}^e - \beta \omega_e \lambda_{dr}^e + \beta \alpha \lambda_{qr}^e + \beta_1 v_{qs}^e \tag{A.1}
\]

\[
\frac{di_{ds}^e}{dt} = -\gamma_{ds}^e + \omega_e i_{ds}^e + \beta \omega_e \lambda_{qr}^e + \beta \alpha \lambda_{dr}^e + \beta_1 v_{ds}^e \tag{A.2}
\]

\[
\frac{d\lambda_{qr}^e}{dt} = -\alpha \lambda_{qr}^e - \omega_e \lambda_{dr}^e + \alpha L_m i_{qs}^e \tag{A.3}
\]

\[
\frac{d\lambda_{dr}^e}{dt} = -\alpha \lambda_{dr}^e + \omega_e \lambda_{qr}^e + \alpha L_m i_{ds}^e \tag{A.4}
\]

\[
T_e = \frac{3p}{2} \frac{L_m}{L_r} \left( \lambda_{dr}^e i_{qs}^e - \lambda_{qr}^e i_{ds}^e \right) \tag{A.5}
\]

\[
\lambda_{qs}^e = \left( \frac{L_s L_r - L_m^2}{L_r} \right) i_{qs}^e + \frac{L_m}{L_r} \lambda_{qr}^e \tag{A.6}
\]

\[
\lambda_{ds}^e = \left( \frac{L_s L_r - L_m^2}{L_r} \right) i_{ds}^e + \frac{L_m}{L_r} \lambda_{dr}^e \tag{A.7}
\]
where

\[\omega_{si} = \omega_e - \omega_r\]

\[\sigma = 1 - \frac{L_m^2}{L_s L_r}\]

\[\alpha = \frac{R_e}{L_r} = \text{inverse of rotor time constant}\]

\[\beta = \frac{L_m}{\sigma L_s L_r}\]

\[\beta_i = \frac{1}{\sigma L_s}\]

\[\gamma = \frac{L_m^2 R_e + L_s^2 R_s}{\sigma L_s L_r^2}\]

**A.1.2 Permanent-magnet synchronous motor [46-47]**

Similarly, the 3-phase PMSM equations in the synchronously rotating reference frame aligned with rotor flux angle (\(\omega_e=\omega_a\)) can be described as the following equations:

\[
\frac{di_{qs}^e}{dt} = -\gamma_{qs} i_{qs}^e - \omega_e i_{ds}^e - \beta \omega_e \lambda_m + \beta v_{qs}^e \tag{A.8}
\]

\[
\frac{di_{ds}^e}{dt} = -\gamma_{ds} i_{ds}^e - \omega_e i_{qs}^e + \beta v_{ds}^e \tag{A.9}
\]

\[\lambda_{qs}^e = L_s i_{qs}^e \tag{A.10}\]

\[\lambda_{ds}^e = L_s i_{ds}^e + \lambda_m \tag{A.11}\]

\[T_e = \frac{3}{2} \left( \lambda_{ds}^e i_{qs}^e - \lambda_{qs}^e i_{ds}^e \right) \tag{A.12}\]

where
\[ \beta = \frac{1}{L_s} \]
\[ \gamma = \frac{R_s}{L_s} = \text{inverse of stator time constant} \]

**A.2 Stationary Reference Frame Model**

**A.2.1 Induction motor**

The mathematical model of 3-phase IM in the stationary reference frame (\( \omega = 0 \)) can be described as the following equations:

\[
\frac{\text{d}i_{qs}^s}{\text{d}t} = -\gamma^s_{qs} - \beta \omega L_s i_{ds}^s + \beta \alpha L_s \lambda_{dr}^s + \beta \gamma^s_{ds} \]
\[
\frac{\text{d}i_{ds}^s}{\text{d}t} = -\gamma^s_{ds} + \beta \omega L_s \lambda_{qr}^s + \beta \alpha L_s \lambda_{dr}^s + \beta \gamma^s_{ds} \]

\[
\frac{\text{d}\lambda_{qr}^s}{\text{d}t} = -\alpha \lambda_{dr}^s + \omega \lambda_{qr}^s + \alpha L_m i_{qs}^s \]

\[
\frac{\text{d}\lambda_{dr}^s}{\text{d}t} = -\alpha \lambda_{dr}^s - \omega \lambda_{qr}^s + \alpha L_m i_{ds}^s \]

\[
T_e = \frac{3}{2} \frac{p L_m}{2} \left( \lambda_{dr}^s i_{qs}^s - \lambda_{qr}^s i_{ds}^s \right) \]

\[
\lambda_{qs}^s = \left( \frac{L_s L_r - L_{sm}^2}{L_r} \right) i_{qs}^s + \frac{L_m}{L_r} \lambda_{qr}^s \]

\[
\lambda_{ds}^s = \left( \frac{L_s L_r - L_{sm}^2}{L_r} \right) i_{ds}^s + \frac{L_m}{L_r} \lambda_{dr}^s \]

where
\[ \sigma = 1 - \frac{L_m^2}{L_s L_r} \]

\[ \alpha = \frac{R_s}{L_r} = \text{inverse of rotor time constant} \]

\[ \beta = \frac{L_m}{\sigma L_s L_r} \]

\[ \beta_i = \frac{1}{\sigma L_s} \]

\[ \gamma = \frac{L_m^2 R_s + L_s^2 R_s}{\sigma L_s L_s^2} \]

A.2.2 Permanent-magnet synchronous motor

Similarly, the 3-phase PMSM equations in the stationary reference frame (\(\omega_e=0\)) can be described as the following equations:

\[ \frac{d i_{qs}^s}{dt} = -\gamma_{qs}^s - \beta \omega_e \lambda_m^s \cos \theta_e + \beta v_{qs}^s \] (A.20)

\[ \frac{d i_{ds}^s}{dt} = -\gamma_{ds}^s + \beta \omega_e \lambda_m^s \sin \theta_e + \beta v_{ds}^s \] (A.21)

\[ \lambda_{qs}^s = L_s i_{qs}^s + \lambda_m^s \sin \theta_e \] (A.22)

\[ \lambda_{ds}^s = L_s i_{ds}^s + \lambda_m^s \cos \theta_e \] (A.23)

\[ T_e = \frac{3 p}{2} \left( \lambda_{ds}^s i_{qs}^s - \lambda_{qs}^s i_{ds}^s \right) \] (A.24)

where

\[ \beta = \frac{1}{L_s} \]
\[ \gamma = \frac{R_s}{L_s} = \text{inverse of stator time constant} \]
APPENDIX B

SOLUTION OF CURRENT DYNAMICS IN VECTOR CONTROLLED DRIVE OF AC MOTORS

The current dynamics of AC motors in the synchronously rotating reference frame aligned with the rotor flux angle (i.e., vector controlled condition) is shown in the following equation:

\[
\begin{bmatrix}
\dot{i}_{qs} \\
\dot{i}_d
\end{bmatrix}
= 
\begin{bmatrix}
-\gamma i_{qs} - \omega_e i_{ds} + u_{qs} \\
-\gamma i_d + \omega_e i_{qs} + u_{ds}
\end{bmatrix}
\]  \hspace{1cm} (B.1)

where

\[
\begin{bmatrix}
u_{qs} \\
u_{ds}
\end{bmatrix}
= 
\begin{cases}
\begin{bmatrix}
-\beta \omega \lambda_c e \\
\beta \alpha \lambda_c e \\
-\beta \omega \lambda_m e \\
0
\end{bmatrix}
+ \beta_1 \begin{bmatrix}
v_{qs} \\
v_{ds}
\end{bmatrix} & \text{for IM} \\
\begin{bmatrix}
v_{qs} \\
v_{ds}
\end{bmatrix} & \text{for PMSM}
\end{cases}
\]

Equation (B.1) can be written in the state-space form as

\[
\dot{x} = Ax + u
\]  \hspace{1cm} (B.2)

where \(x = \begin{bmatrix} i_{qs} \\ i_d \end{bmatrix}\), \(A = \begin{bmatrix} -\gamma & -\omega_e \\ \omega_e & -\gamma \end{bmatrix}\), and \(u = \begin{bmatrix} u_{qs} \\ u_{ds} \end{bmatrix}\).

Notice that the parameters \(\gamma\), \(\beta\), \(\beta_1\), and \(\alpha\) are defined in appendix A. The solution to equation (B.2) is of the following form:

\[
x(t) = e^{At}(x(0) + A^{-1}u) - A^{-1}u
\]  \hspace{1cm} (B.3)
where $A^{-1} = \frac{1}{\gamma^2 + \omega_e^2} \begin{bmatrix} -\gamma & \omega_e \\ -\omega_e & -\gamma \end{bmatrix}$, and $e^{At} = e^{-\gamma t} \begin{bmatrix} \cos \omega_e t & -\sin \omega_e t \\ \sin \omega_e t & \cos \omega_e t \end{bmatrix}$.

Precisely, the solution of current dynamics can be explicitly shown as follows:

\[
\begin{bmatrix} i_{qs}^e(t) \\ i_{ds}^e(t) \end{bmatrix} = e^{-\gamma t} \begin{bmatrix} \cos \omega_e t & -\sin \omega_e t \\ \sin \omega_e t & \cos \omega_e t \end{bmatrix} \begin{bmatrix} i_{qs}^e(0) \\ i_{ds}^e(0) \end{bmatrix} + \frac{1}{\gamma^2 + \omega_e^2} \begin{bmatrix} -\gamma u_{qs}^e(t) + \omega_e u_{ds}^e(t) \\ -\gamma u_{ds}^e(t) - \omega_e u_{qs}^e(t) \end{bmatrix} - \frac{1}{\gamma^2 + \omega_e^2} \begin{bmatrix} -\gamma u_{qs}^e(t) + \omega_e u_{ds}^e(t) \\ -\gamma u_{ds}^e(t) - \omega_e u_{qs}^e(t) \end{bmatrix}
\]  

(B.4)

If the stator resistance (and rotor resistance for IM) are assumingly negligible, then $\gamma = 0$.

As a result, the current dynamics equation seen in (B.1) simply becomes

\[
\begin{bmatrix} \dot{i}_{qs}^e \\ \dot{i}_{ds}^e \end{bmatrix} = \begin{bmatrix} -\omega_e & u_{qs}^e \\ \omega_e & u_{ds}^e \end{bmatrix}
\]  

(B.5)

And the solution of current dynamics (B.5) with resistance ignored is similar to (B.3) as follows:

\[
x(t) = e^{At} \left( x(0) + A^{-1}u \right) - A^{-1}u
\]  

(B.6)

where $A^{-1} = \frac{1}{\omega_e} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$, and $e^{At} = \begin{bmatrix} \cos \omega_e t & -\sin \omega_e t \\ \sin \omega_e t & \cos \omega_e t \end{bmatrix}$.

Equivalently, the solution of current dynamics (B.6) can be explicitly shown as follows:

\[
\begin{bmatrix} i_{qs}^e(t) \\ i_{ds}^e(t) \end{bmatrix} = \begin{bmatrix} \cos \omega_e t & -\sin \omega_e t \\ \sin \omega_e t & \cos \omega_e t \end{bmatrix} \begin{bmatrix} i_{qs}^e(0) \\ i_{ds}^e(0) \end{bmatrix} + \frac{1}{\omega_e} \begin{bmatrix} u_{qs}^e(t) \\ -u_{ds}^e(t) \end{bmatrix} - \frac{1}{\omega_e} \begin{bmatrix} u_{qs}^e(t) \\ -u_{ds}^e(t) \end{bmatrix}
\]  

(B.7)
BIBLIOGRAPHY


