12-1993

Application of Fuzzy Logic for Performance Enhancement of Drives

Gilberto Costa Sousa

University of Tennessee - Knoxville

Recommended Citation


https://trace.tennessee.edu/utk_graddiss/3384
To the Graduate Council:

I am submitting herewith a dissertation written by Gilberto Costa Sousa entitled "Application of Fuzzy Logic for Performance Enhancement of Drives." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Engineering Science.

Bimal K. Bose, Major Professor

We have read this dissertation and recommend its acceptance:

Frederick W. Symonds, Walter L. Green, J. M. Bailey, B. R. Upadhyaya

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)
To the graduate Council:

I am submitting herewith a dissertation written by Gilberto Costa D. Sousa entitled "Application of Fuzzy Logic for Performance Enhancement of Drives". I have examined the final copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.

We have read this dissertation and recommend its acceptance:

Bimal K. Bose, Major Professor

[Signatures]

Accepted for the Council:

[Signature]

Associate Vice Chancellor
and Dean of The Graduate School
APPLICATION OF FUZZY LOGIC FOR PERFORMANCE
ENHANCEMENT OF DRIVES

A Dissertation

Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Gilberto Costa D. Sousa

December 1993
ACKNOWLEDGEMENTS

This work has been made possible due to the generosity of several people and institutions. I would like to initially thank my employer, The Federal University of Espirito Santo (UFES), and the Brazilian Council of Scientific and Technological Development (CNPq) for their invaluable financial support. I am also grateful to Research Triangle Institute (RTI) for funding part of this work.

I wish to express my deep appreciation to my advisor, Prof. B. K. Bose for his guidance and constant challenges that helped me mature in this new field. Thanks are in order to Profs. J. M. Bailey, W. L. Green, F. W. Symonds, and B. R. Upadhyaya, for their kindness in serving as my supervisory committee.

A special thank is reserved for Dr. K. S. Kim, of Goldstar Industrial Systems Co., Korea, for designing the basic vector control system and introducing me to the practical aspects of drive technology. I am also grateful to Prof. M. G. Simoes for the preliminary study on converter modeling. The support of R. L. Lambert and other Power Electronics Application Center staff members has been of great help.

Finally, I would like to express my gratitude to my wife, Loise C. P. Drumond, for her unconditional emotional support, as well as for helping me in the preparation of this manuscript.
ABSTRACT

Fuzzy logic shows enormous potential for advancing power electronics technology. Its application to DC and AC drives control is discussed here.

Initially, a phase-controlled bridge converter DC drive was considered. Analysis of converter performance at continuous and discontinuous conduction modes was first conducted. Fuzzy control was used to linearize the transfer characteristics of the converter in discontinuous conduction mode. It was then extended to current and speed loops, replacing the conventional proportional-integral controllers. The control algorithms were developed in detail, and verified by PC-SIMNON (developed by Lund Institute of Technology, Sweden) digital simulation. Significant performance improvement was achieved over conventional control methods.

Efficiency optimization of an indirect vector controlled induction motor drive was next considered. An accurate loss model of the converter induction machine system was first developed. Steady-state fundamental and harmonics loss characteristics, besides the dynamics of the machine were analyzed and incorporated in the model, resulting in a new synchronous frame dynamic $D^e$-$Q^e$ equivalent circuit. The converter system has been modeled accurately for conduction and switching losses. The lossy models were then used in the validation of the fuzzy logic based on-line efficiency optimization control. At steady-state, the fuzzy controller adaptively changes the excitation current on the basis of measured input power, until the maximum efficiency point is reached. The pulsating
torque, due to flux reduction, has been compensated by an ingenious feedforward scheme. During transients, rated flux is established, to get the best transient response. After a comprehensive simulation study, an experimental 5 hp drive system was tested, with the proposed controller implemented on a Texas Instrument TMS320C25 digital signal processor, and the theoretical development was fully validated.

Finally, fuzzy logic was applied in combination with model-reference adaptive control (MRAC) technique to slip gain tuning of an indirect vector controlled induction motor drive. The MRAC methods based on reactive power and D-axis voltage were combined through a weighting factor, generated by a fuzzy controller, that ensures the use of the best method for any point in the torque-speed plane. A second fuzzy controller tunes the slip gain based on combined detuning error and its slope. The drive performance was extensively investigated through simulations and experiments. The results confirmed the validity of the proposed method.
# TABLE OF CONTENTS

1. INTRODUCTION
   1.1 Artificial Intelligence and Adjustable Speed Drives
   1.2 Principles of Fuzzy Logic Control
   1.3 Outline of Dissertation

2. FUZZY CONTROL OF A DC DRIVE SYSTEM
   2.1 Introduction
   2.2 Fuzzy Controlled DC Drive System
      2.2.1 Fuzzy Linearization of Converter Characteristics
      2.2.2 Fuzzy Control of Current and Speed Loops
   2.3 Simulation Study

3. LOSS MODELING OF CONVERTER INDUCTION MACHINE SYSTEMS
   3.1 Introduction
   3.2 Dynamic Lossy Model for the Induction Machine
      3.2.1 Induction Machine Losses
      3.2.2 Temperature and Saturation Effects
3.2.3 Per-phase Harmonic Equivalent Circuit........................................ 59
3.2.4 Synchronous Frame Lossy Equivalent Circuit.................................... 62
3.2.5 Lossy Model Parameter Computation............................................... 68
3.3 Loss Modeling of Converter System................................................... 73
  3.3.1 Loss Modeling of Diode Rectifier.............................................. 73
  3.3.2 Loss Modeling of PWM Inverter.................................................. 75
3.4 Model Validation.............................................................................. 85

4. FUZZY EFFICIENCY OPTIMIZATION CONTROL......................................... 95
  4.1 Introduction..................................................................................... 95
  4.2 Fuzzy Efficiency Optimization of a Vector Control Drive.................... 98
    4.2.1 The Efficiency Optimization Controller..................................... 102
    4.2.2 Feedforward Pulsating Torque Compensation.............................. 108
    4.2.3 Transition to Optimum Transient Response Mode.......................... 112
  4.3 Simulation Program Development.................................................... 114
  4.4 Simulation Study............................................................................ 122
  4.5 Hardware Circuit Design............................................................... 127
    4.5.1 Current Controller................................................................. 129
    4.5.2 Monitoring and Protection...................................................... 129
    4.5.3 Speed Measurement and Control Interface.................................. 132
    4.5.4 DC Link Voltage and Current Sensing Interface.......................... 134
4.6 Real Time Software Design .................................................. 134
  4.6.1 Basic Vector Control Functions ...................................... 137
  4.6.2 Speed Computation ..................................................... 138
  4.6.3 Efficiency Optimization controller .................................. 138
4.7 Experimental Study ......................................................... 140

5. FUZZY LOGIC BASED SLIP GAIN TUNING ............................. 152
  5.1 Introduction ............................................................... 152
  5.2 Fuzzy Tuning Controller ................................................ 158
    5.2.1 Reactive Power and D-axis Voltage Regulators ............... 160
    5.2.2 Derivation of Combined Error Signal .......................... 162
    5.2.3 Design of the Fuzzy Tuning Controller ....................... 164
  5.3 Simulation Study .......................................................... 170
  5.4 Hardware Design .......................................................... 172
  5.5 Real Time Software Design Issues .................................... 177
  5.6 Experimental Study ........................................................ 181

6. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH .... 187

REFERENCES ........................................................................... 193

vii
APPENDICES ...................................................................................................................... 200

A. SIMNON Simulation Programs Listing ..................................................................... 201
   A.1 DC Drive Programs .................................................................................................. 201
   A.2 Loss Modeling and Efficiency Optimization Programs ....................................... 209
   A.3 Slip Gain Tuning Programs .................................................................................... 222

B. TMS320C25 Assembly Programs Listing ................................................................. 228
   B.1 Efficiency Optimization and Vector Control Programs ..................................... 228
   B.2 Slip Gain Tuning Assembly Programs ................................................................ 260

VITAE .................................................................................................................................. 285
# LIST OF TABLES

| 2.1 | Rule base for $\Delta \alpha$ compensation | 25 |
| 2.2 | Rule base for current and speed controllers | 32 |
| 2.3 | Parameters of DC drive system | 35 |
| 2.4 | Performance comparison of fuzzy and PI controlled drive system | 45 |
| 3.1 | Power circuit parameters of the AC drive system | 88 |
| 4.1 | Rule base for the fuzzy efficiency controller | 107 |
| 4.2 | Induction machine parameters for efficiency optimization studies | 123 |
| 5.1 | Rule base for weighting factor ($K_r$) calculation | 168 |
| 5.2 | Rule base for increment of slip gain ($\Delta K_s$) | 171 |
| 5.3 | Parameters for the induction servomotor | 171 |
LIST OF FIGURES

1.1 Representation of temperature using crisp sets and fuzzy sets .................... 4
1.2 Basic operations involving fuzzy sets ........................................................ 6
1.3 Fuzzy composition method by SUP-MIN principle ....................................... 9
1.4 Basic structure of a fuzzy controlled system ............................................. 10
2.1 Four quadrant phase-controlled converter dc drive ....................................... 16
2.2 Converter voltage and current waveforms .................................................. 18
2.3 Fuzzy controlled dc drive system .............................................................. 20
2.4 Theoretical V_d-I_a (pu) phase plot without compensation ......................... 23
2.5 Membership functions for Δα compensation .............................................. 26
2.6 Membership functions of speed and current controllers ............................. 31
2.7 Open loop V_d step (α =70°) response without compensation ..................... 36
2.8 Phase plot V_d-I_a with Δα compensation ................................................. 38
2.9 Open loop V_d step response with Δα compensation ................................. 39
2.10 Current loop response of the fuzzy controller .......................................... 40
2.11 Current loop response of the PI controller .............................................. 41
2.12 Fuzzy control system response to a ω_r step and T_L step ........................... 42
2.13 PI control system response to a ω_r step and T_L step ................................ 43
2.14 Fuzzy control system response to a ω_r step with a new inertia of
    four times the original value ........................................................................... 46
2.15 PI control system response to a ω_1^* step with a new inertia of

four times the original value

3.1 Per-phase equivalent circuit

3.2 Standard D-Q equivalent circuits in synchronous frame

3.3 Converter-machine system for variable speed drive

3.4 Per-phase harmonic equivalent circuit of the induction motor

3.5 D^4-Q^4 equivalent circuit with core loss resistance

3.6 D^6-Q^6 equivalent circuit with core loss resistance

3.7 Lossy D^6-Q^6 equivalent circuits in synchronously rotating reference frame.

3.8 Diode rectifier equivalent circuit

3.9 Rectifier voltage and current waves

3.10 Transistor inverter phase leg with conduction loss equivalent circuit

3.11 Pulse width modulation waves for phase leg A

3.12 Typical turn-on and turn-off switching waves for transistor Q

3.13 Q^4-axis circuit showing loops for state equations derivation

3.14 Steady state performance

3.15 Rotor flux responses at constant speed

3.16 Torque responses of the drive at rated flux

4.1 Simulation of the input power minimization process

4.2 Indirect vector controlled induction motor drive incorporating the efficiency optimization controller

4.3 Principle of efficiency optimization control with rotor flux programming
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4</td>
<td>Efficiency optimization control block diagram</td>
<td>103</td>
</tr>
<tr>
<td>4.5</td>
<td>Membership functions for the fuzzy efficiency control</td>
<td>106</td>
</tr>
<tr>
<td>4.6</td>
<td>Principles of feedforward torque compensation</td>
<td>109</td>
</tr>
<tr>
<td>4.7</td>
<td>Feedforward pulsating torque compensator block diagram</td>
<td>111</td>
</tr>
<tr>
<td>4.8</td>
<td>Transition between efficiency optimization and transient response optimization modes</td>
<td>113</td>
</tr>
<tr>
<td>4.9</td>
<td>SIMNON simulation block diagram showing I/O variables</td>
<td>115</td>
</tr>
<tr>
<td>4.10</td>
<td>Fuzzy efficiency optimization flowchart</td>
<td>116</td>
</tr>
<tr>
<td>4.11</td>
<td>Evaluation of degree of membership</td>
<td>118</td>
</tr>
<tr>
<td>4.12</td>
<td>Feedforward torque compensation flowchart</td>
<td>121</td>
</tr>
<tr>
<td>4.13</td>
<td>Time domain simulated optimum efficiency search curves</td>
<td>124</td>
</tr>
<tr>
<td>4.14</td>
<td>Simulated efficiency curves</td>
<td>125</td>
</tr>
<tr>
<td>4.15</td>
<td>Drive performance in time domain with sudden increase of load torque</td>
<td>126</td>
</tr>
<tr>
<td>4.16</td>
<td>Block diagram for vector control system hardware</td>
<td>128</td>
</tr>
<tr>
<td>4.17</td>
<td>Hysteresis band current controller</td>
<td>130</td>
</tr>
<tr>
<td>4.18</td>
<td>Monitoring and protection circuit diagram</td>
<td>131</td>
</tr>
<tr>
<td>4.19</td>
<td>Speed sensing and control interface</td>
<td>133</td>
</tr>
<tr>
<td>4.20</td>
<td>DC link voltage and current sensing</td>
<td>135</td>
</tr>
<tr>
<td>4.21</td>
<td>Software structure of the drive system including the efficiency</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td>optimization controller</td>
<td></td>
</tr>
<tr>
<td>4.22</td>
<td>Speed computation flowchart</td>
<td>139</td>
</tr>
</tbody>
</table>
4.23 AC supply phase voltage and current waveforms at $\omega_r=0.75$ pu and $T_L=0.50$ pu .............................................................. 142

4.24 Machine input currents at $\omega_r=0.75$ pu and $T_L=0.50$ pu .............................................................. 142

4.25 Experimental search curves at $\omega_r=0.25$ pu and $T_L=0.10$ pu, with pulsating torque compensation .............................................................. 143

4.26 Experimental search curves $\omega_r=0.50$ pu and $T_L=0.30$ pu .............................................................. 145

4.27 Search curves at $\omega_r=0.25$ pu and $T_L=0.10$ pu without pulsating torque compensation .............................................................. 147

4.28 Drive performance in time domain with sudden changes in command speed .............................................................. 148

4.29 Experimental efficiency curves .............................................................. 150

5.1 Indirect vector controlled induction motor drive with open loop torque and flux control .............................................................. 153

5.2 Steady-state detuning effects on torque and rotor flux .............................................................. 155

5.3 Transient response for developed torque and rotor flux .............................................................. 156

5.4 Indirect vector-controlled induction motor drive showing the proposed fuzzy tuning controller .............................................................. 159

5.5 Fuzzy logic based MRAC tuning control block diagram .............................................................. 163

5.6 Normalized control loop error vs. normalized slip gain curves ($\omega_r=0.50$ pu) .............................................................. 165

5.7 Membership functions for FLC-1 .............................................................. 167

5.8 Membership functions for FLC-2 .............................................................. 169

xiii
5.10 Performance of tuning controller to a 100% increase in rotor resistance... 174
5.11 Voltage sensing interface........................................................................ 176
5.12 Real time software flowchart................................................................. 178
5.13 Phase-shift compensation phasor diagram ............................................ 180
5.14 Saturation profile for $L_\alpha$ .................................................................. 182
5.15 Experimental tuning performance at $T_L=0.1$ pu and $K_{le}/K_{le}(0)=2$ .... 183
5.16 Experimental tuning performance at $T_L=0.5$ pu and $K_{le}/K_{le}(0)=5$ .... 183
5.17 Speed response at no load for a square-wave torque command ($i_{qs}$) .... 186
CHAPTER 1

INTRODUCTION

1.1 Artificial Intelligence and Adjustable Speed Drives

Adjustable speed drive technology has evolved enormously over the past 30 years. This evolution has been made possible because of technological advances in a number of related fields, such as power semiconductor devices, converter topologies and control techniques, microprocessors and digital signal processors (DSP). More recently, the application of intelligent control technologies, such expert systems, fuzzy logic and artificial neural networks, is advancing the frontier of this technology [19].

Among the artificial intelligence (AI) techniques, fuzzy logic is perhaps the most successful one, if judged from the standpoint of the number of practical applications [10]. Compared to other AI control techniques, i.e., expert systems and neural networks, fuzzy logic controllers numerically process structured knowledge that is embedded in only a few rules; expert systems symbolically process structured knowledge defined by a larger number of rules, whereas neural networks numerically process unstructured knowledge, and heavily depend on off-line learning techniques.

Fuzzy logic was introduced by Zadeh [1] in 1965, and can be viewed as a generalization of conventional Boolean logic. Its main purpose is the mathematical representation of imprecision and non-statistical uncertainty. The key concept in this
theory is that of a fuzzy set. A fuzzy set possesses a distinct feature of allowing partial membership, with degrees of membership varying from 0 (non-member) to 1 (full member). In contrast, in a "crisp" or Boolean set, a given element is a member of the set or it is not a member at all.

A fuzzy logic controller for a given process is capable of embedding in the control strategy, the qualitative knowledge and experience that an operator or engineer has about the process. Typically, the design of a fuzzy logic controller starts with a linguistic description of the control strategy, in the form of IF-THEN rules, that relate some process states (usually error and change in error), to the appropriate control action. The next step consists in making a quantitative interpretation of the linguistic variables, in the form of fuzzy sets. This is the most difficult step in the design process, and usually requires a significant number of iterations. The design process is completed by selecting an inference method, that essentially defines how the set of control rules, i.e., the rule base, is evaluated to derive the control action for a particular process state.

Fuzzy logic control does not require an accurate model of the plant, and therefore, it suits well to a process for which a formal model is unknown or ill-defined. It also works well for complex non-linear multi-dimensional systems, systems with parameter variation problem, or where the sensor signals are not precise.

Fuzzy Logic applications to power electronics and drives are almost entirely new, and this work is one of the first attempts to systematically apply fuzzy logic to the control of DC and AC drives. Prior to 1991, Li and Lau [5] applied fuzzy logic to a microprocessor-based servomotor controller, assuming a linear power amplifier. They also
compared system performance under PID control and model-reference adaptive control (MRAC) to that obtained with fuzzy control, and demonstrated the superiority of the latter. Da Silva et al [6] developed a fuzzy adaptive controller for a four quadrant power converter. The control system possesses a three-level hierarchical structure capable of identifying the best control law for any load condition, such that robust operation is achieved even under sudden changes in load.

Fuzzy logic shows enormous potential for advancing power electronics technology. Its ability to incorporate qualitative knowledge and to deal with imprecise information makes it very attractive to power electronics systems, where non-linearity is a common feature, and a precise model is difficult to obtain. In such cases, fuzzy logic can be applied to derive a linguistic model of the system, that describes its operational behavior. This fuzzy model can then be used in the design of a fuzzy controller for the system, just as a mathematical model is helpful in the design of a conventional controller.

1.2 Principles of Fuzzy Logic Control

It is appropriate to review here some basic concepts of fuzzy logic and fuzzy control. Although numerous fuzzy inference methods have been proposed in the literature [11]-[12], this discussion is based on Zadeh's [1]-[2] and Mamdani's [3] approaches.

A fuzzy set is characterized by a membership function, that associates each element in the set to a number in the real interval [0,1], that represents its grade of membership in the set. In contrast, a Boolean or "crisp" set is defined by a characteristic function, that maps all members to logic level 1 and the non-members to 0. Fig. 1.1
difference for the case of a hypothetical temperature control system. In Fig. 1.1(a), a crisp classification is provided, such that, the temperature value $T = 67 \, ^\circ\text{F}$ is a member of the HOT set only. In contrast, in Fig. 1.1(b) temperature is considered a fuzzy variable, and $T = 67 \, ^\circ\text{F}$ is a partial member of both MILD and HOT fuzzy sets. Because language is the primary vehicle for conveying knowledge, fuzzy variables are usually referred to as linguistic variables, and the fuzzy sets are viewed as the mathematical representation of their linguistic values (e.g., COLD, MILD, and HOT in Fig. 1.1). The numerical interval, that is relevant for the description of a fuzzy variable, is commonly named Universe of Discourse (in Fig. 1.1, the real interval $[20,90]$).

Fig. 1.1 Representation of temperature using:

(a) Crisp sets.  (b) Fuzzy sets.
A few operations of Boolean set theory are also valid in fuzzy set theory, and are described below. Let $\mu_A(x)$ denote the degree of membership of a given element $x$ in the fuzzy set $A$.

**Union:** Given two fuzzy sets $A$ and $B$, defined on a universe of discourse $X$, the union $(A \cup B)$ is also a fuzzy set of $X$, with membership function given as

$$\mu_{A \cup B}(x) = \max[\mu_A(x), \mu_B(x)]$$  \hspace{1cm} (1.1)

where $x$ is any element of $X$.

**Intersection:** The intersection of two fuzzy sets $A$ and $B$ of the universe of discourse $X$, denoted by $A \cap B$, has the membership function given by

$$\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)]$$  \hspace{1cm} (1.2)

**Compliment:** The compliment of a given set $A$ of the universe of discourse $X$ is denoted by $\neg A$, and has the membership function

$$\mu_{\neg A} = 1 - \mu_A(x)$$  \hspace{1cm} (1.3)

Fig. 1.2 illustrates these three basic properties. For fuzzy logic control, a few more concepts, such as fuzzy implication (fuzzy rules) and fuzzy composition (fuzzy inference) are important. A fuzzy rule typically has an IF-THEN format as follows:

$$\text{IF ( x is A AND y is B ) THEN (z is C)}$$

where $x$, $y$ and $z$ are fuzzy variables and $A$, $B$ and $C$ are fuzzy sets in the universes of discourse $X$, $Y$ and $Z$, respectively. If the conditions expressed in the antecedent (IF portion) are met, then the action(s) specified in the consequent (THEN portion) are
Fig. 1.2 Basic operations involving fuzzy sets.

(a) Original fuzzy sets defined on X.

(b) Union. (c) Intersection. (d) Negation.
taken. In order to design a fuzzy controller, a fuzzy rule base consisting of several rules must be constructed. For example, consider a hypothetical fuzzy speed control system for a DC motor, where the speed error (E) and change in error (CE) are used to determine changes in the control signal (DU), that in this case is the command armature current $I_a^*$. A part of the rule base would be:

Rule 1: \[ \text{IF E is Zero AND CE is Zero THEN DU is Zero}; \]
Rule 2: \[ \text{IF E is Zero AND CE is Negative Small THEN DU is Negative Small}; \]
Rule 3: \[ \text{IF E is Positive Small AND CE is Negative Small THEN DU is Zero}; \]

Here, error (E), change in error (CE) and change in control (DU) are considered fuzzy variables, with possible values given by fuzzy sets such as Positive Small, Negative Small, and so on. As illustrated in Fig. 1.1, a given numerical value can be a member of more than one fuzzy set. This means that, for a particular input pair of values (E and CE), more than one rule could be activated or "fired". Therefore, there must be a way to combine the individual control actions of the fired rule, such that a single, meaningful action is taken. In fuzzy logic terms, the composition operation is the mechanism by which such task can be performed. Although several composition principles have been proposed in the literature, the most common one is the SUP-MIN (SUPremum-MINimum) composition. In a simplistic way, given a rule base, it is possible to construct a n-dimensional fuzzy relation $R$ (consider it as a function of $n$ variables). The simplest case is a single input ($x$), single output ($u$) system, resulting in a two-dimensional fuzzy relation, represented by a membership function $\mu_R(x,u)$. For this case, the composition operation can be expressed as:
\[ \mu_b(u) = \text{Sup}_x \left[ \text{Min} (\mu_A(x), \mu_R(x,u)) \right] \]  

(1.4)

where \( A \) is the known fuzzy set for the input \( x \) and \( B \) is the inferred fuzzy set for the output \( u \). In practice, the fuzzy relation \( R \) is seldom evaluated explicitly; instead, the SUP-MIN composition is applied to one rule at a time, and the individual control actions combined using the union operation. Fig. 1.3 illustrates the fuzzy composition by SUP-MIN principle for the two stated rules. Note that the output membership function of each rule is given by MIN operator whereas the combined fuzzy output is given by the SUP operator. This will be evident by the numerical example in Chapter 2.

The general structure of a fuzzy control system is given in Fig. 1.4. The control signal \( U \) is inferred from the two state variables error (\( e \)) and change in error (\( ce \)) ( \( \text{de/dt} \) for the sampling interval). The \( e \) and \( ce \) are per unit (pu) signals derived from the actual \( E \) and \( CE \) signals by dividing with the respective gain factors as shown. In a strict sense, the fuzzy controller is designed to process fuzzy quantities only. Therefore, all crisp input values must be converted to fuzzy sets before being used. This process is called fuzzification operation, and can be performed by considering the crisp input values as "singletons" (fuzzy sets that have membership value of 1 for the given input value, and 0 for all other points in the universe of discourse). In Fig. 1.3, the given input values \( E' \) and \( CE' \) were "converted" to fuzzy sets by this process, before being compared to the other fuzzy sets. In a similar way, there is a need for converting the output of the fuzzy controller (a fuzzy set) to a crisp value required by the plant. This is called defuzzification operation, and can be performed by a number of methods of which the center-of-gravity (also known as centroid) and height methods are common. The centroid defuzzification
Fig. 1.3 Fuzzy composition method by SUP-MIN principle.
Fig. 1.4 Basic structure of a fuzzy controlled system.
method selects the output crisp value corresponding to the center of gravity of the output membership function, which is given by the expression

\[ U_o = \frac{\int u \mu(u) \, du}{\int \mu(u) \, du} \]  
(1.5)

In the height method, the centroid of each output membership function associated with every rule is first evaluated. The final output is then calculated as the average of the individual centroids, weighted by their heights (degree of membership) as

\[ U_o = \frac{\sum_{i=1}^{n} u_i \mu(u_i)}{\sum_{i=1}^{n} \mu(u_i)} \]  
(1.6)

Finally, the data base provides the operational definitions of the fuzzy sets used in the control rules, fuzzification and defuzzification operations.

Fuzzy logic controllers have been successfully applied to a number of different processes, in many cases yielding improved performance, compared to conventional control techniques. A qualitative explanation for this superior performance will now be discussed.

For a discrete PI controller, the change in control \( \Delta u(k) \) can be expressed as

\[ \Delta u(k) = K_p c e(k) + K_i \Delta t \, e(k) \]  
(1.7)

where \( K_p \) and \( K_i \) are the proportional and integral gains, respectively, \( e(k) \) is the control error, \( c e(k) \) (= \( e(k) - e(k-1) \)) is the change in error, and \( \Delta t \) is the sampling interval. For a specific value of error \( e(k) \), the control increment \( \Delta u(k) \) varies linearly with \( c e(k) \). In other
words, the mapping produced by a PI controller results in a plane in the three-dimensional space \((ce, e, \Delta u)\). On the other hand, for a fuzzy controller, any type of mapping is theoretically possible. Again, for a given value of error \(e(k)\), the relationship between \(\Delta u(K)\) and \(ce(k)\) is not confined to a straight line, resulting in practice, in a variable \(K_p\) gain. Of course, in order to realize the potential benefits of this highly flexible structure, a much more elaborated design process is required. In response to this challenge, significant research effort has been devoted to the development of a theory for fuzzy dynamic systems, addressing problems such as fuzzy state variables, controllability and stability [12].

1.3 Outline of the Dissertation

The application of fuzzy logic to the control of both DC and AC drive systems is discussed in this work. Phase-controlled converter DC drives have been used in applications that usually require fast response, such as steel mills. Under normal conditions, the converter operates in a continuous conduction mode, in which the converter transfer characteristics are linear, and fast transient response for the current loop is obtained. However, at light load conditions, the converter tends to operate in a discontinuous conduction mode, that results in non-linear converter transfer characteristics and sluggish response of the current loop. Ultimately, the overall system performance is adversely affected. Furthermore, in many applications load disturbance is quite common, as well as parameter variation in the drive system. Consequently, fuzzy logic control can be applied with potential advantages over classical control techniques.
In Chapter 2, a speed control system that uses a phase-controlled bridge converter and a separately excited DC machine was investigated. Fuzzy control was used to linearize the transfer characteristics of the converter in discontinuous conduction mode occurring at high speed and light loads. The fuzzy control was then extended to the current and speed control loops, replacing the conventional proportional-integral (PI) control method. The compensation and control algorithms were developed in detail and verified by digital simulation using PC-SIMNON.

Vector control drives are becoming the industry standard for high performance applications [19]. Under vector control, an AC machine drive dynamic imitates that of a separately excited DC machine drive, with the advantages of ruggedness and higher torque to inertia ratio.

Most drive systems operate at rated flux, to ensure fast transient responses. However, at light loads, rated flux operation results in unnecessarily high core losses, thus impairing the efficiency of the drive. To address this problem, fuzzy logic was applied to the efficiency optimization of a speed control system, using an indirect vector controlled induction motor drive. In order to validate the new control scheme, as well as to predict system performance, an accurate loss model of the converter induction machine system was first developed, as discussed in Chapter 3. The model properly represents the loss phenomena as well as dynamic behavior of both machine and converter. After validation of the lossy models, they were used by the fuzzy logic based on-line efficiency optimization control, presented in Chapter 4. During efficiency optimization, the rotor flux is typically reduced, causing undesirable low frequency torque pulsations. To address this
problem, an innovative feedforward torque compensator is also described in Chapter 4. The drive system with the proposed efficiency optimization controller was simulated with the lossy models of converter and machine, and its performance was thoroughly investigated. An experimental 5 hp drive system, with the proposed controller implemented on a Texas Instruments TMS320C25 digital signal processor, was tested in the laboratory to validate the theoretical development.

Indirect vector control induction motor drives require the knowledge of rotor time constant, to generate the slip frequency used in the synthesis of the unit vectors. Fast dynamic response is only achieved when the slip gain used in the control matches the actual machine rotor time constant. Unfortunately, machine parameters are temperature and saturation dependent, and drive performance tends to deteriorate. Chapter 5 discusses the application of fuzzy logic to the slip gain tuning of an indirect vector controlled induction motor drive, using the standard model-reference adaptive control (MRAC) technique. The MRAC methods based on reactive power and synchronous frame D-axis voltage are combined together with a weighting factor that is generated by a fuzzy controller. The weighting factor ensures the dominant use of reactive power method in low speed high torque regions whereas the D-axis voltage method is dominant in high speed low torque regions. A second fuzzy controller tunes the slip gain based on the combined detuning error and its slope, so as to ensure fast convergence at any operating point in the torque-speed plane. Drive performance was extensively investigated, initially through PC-SIMNON simulation, and then using an experimental drive system.
CHAPTER 2

FUZZY CONTROL OF A DC DRIVE SYSTEM

2.1 Introduction

Direct current (dc) drives have been the traditional industry option in applications that require adjustable speed, mainly due to the ease of control and fast speed response. Although in the recent years alternate current (ac) drives have gained increased popularity, and will certainly dominate the adjustable speed drive market, dc drives still constitute the majority of industrial drive systems.

The application of fuzzy logic to the control of a dc servomotor has been discussed in the literature [5]. The authors described the system performance in the presence of sudden changes in inertia, under fuzzy control as well as under PID and MRAC control, and showed the superiority of fuzzy control. In their study, however, a linear power amplifier was assumed, and consequently, the problems associated with the non-linearities of actual power converters were not addressed.

Among the dc drive systems, the phase-controlled bridge converter drive shown in Fig. 2.1 is the most popular. It is shown here in a four-quadrant speed control mode, with inner current control loop, that provides fast transient response and also limits the armature current. Traditionally, proportional-integral (PI) control is used in both speed and current loops. The firing angle $\alpha$ is obtained from the current PI controller output $V$, 

15
Fig. 2.1 Four quadrant phase-controlled converter DC drive.
by cosine wave crossing method, such that the converter output voltage $V_d$ is proportional to $V_s$, i.e., a linear transfer characteristic is obtained for the converter system. The converter, however, may operate in either continuous or discontinuous conduction modes, as indicated by the current and voltage waveforms of Fig. 2.2. At high load torque, the difference between applied voltage $V_d$ and the counter emf $V_c$ is large enough such that the conduction will be continuous. However, at light load torques, this difference becomes small and the conduction tends to be discontinuous. Under discontinuous conduction, the converter characteristics is no longer linear, and the current loop response tends to deteriorate. This in turn results in a sluggish speed response, rendering the system inadequate for high performance applications. Among the number of methods suggested to linearize the converter transfer characteristics at discontinuous conduction mode, the look-up table method suggested by Ohmae et al. [15] appears to be very attractive. In this method, an auxiliary compensating $\Delta \alpha$ angle is generated as a function of main $\alpha$ angle and armature current $I_a$, and then added with $\alpha$ to generate the actual firing angle. The two-dimensional relation of $\Delta \alpha$ can be pre-computed for each $X/R$ parameter and stored in the form of a look-up table for microcomputer implementation. If the parameter $X/R$ variation is considered, and the compensating angle is needed with good accuracy, then the look-up tables memory tends to be very large.

The following sections discuss the application of fuzzy logic to a DC drive system. Initially, a fuzzy controller was designed to generated a $\Delta \alpha$ compensating angle, that when added to the original firing angle $\alpha$, produces a linear characteristic for the converter system, even under discontinuous conduction mode. Next, fuzzy control was
Fig. 2.2 Converter voltage and current waveforms [15].

(a) Discontinuous conduction.

(b) Continuous conduction.
extended to speed and current control loops, replacing the conventional PI control method. The compensation and control algorithms were developed in detail and verified by digital simulation of a drive system. The results were originally published at the IEEE IAS Annual Meeting, 1991 [14].

2.2 Fuzzy Controlled DC Drive System

The speed control system under consideration is shown in Fig. 2.3. The power circuit consists of a phase-controlled bridge converter that drives a separately excited DC motor. For simplicity, the converter is used in motoring mode only with fixed field excitation. Fuzzy controllers were used in both the speed control loop and in the inner current control loop. The current loop output $V_s'$ is added with the estimated feed-forward counter emf signal $\hat{V}_c$, to generate the control signal $V_s$, which then generates the firing angle $\alpha$, by cosine wave crossing method. The feed-forward addition of counter emf gives faster loop response. The fuzzy compensator is also indicated in the figure.

As mentioned in the introduction, the converter may operate in either continuous or discontinuous conduction mode. Complete mathematical analysis of converter performance characteristics has been made for both operating modes [21], and the relevant results will be repeated here. When current is flowing into the armature, the loop equation is given by

$$L \frac{di_a}{dt} + Ri_a = V_m \sin \omega t - V_c$$

(2.1)

where $V_c = $ machine counter emf, $V_m = $ peak supply line voltage, $i_a = $
Fig. 2.3 Fuzzy controlled DC drive system.
instantaneous armature current, and \( R \) and \( L \) are, respectively, the resistance and inductance of the armature plus ac source and connections. In continuous conduction mode, the average armature current \( I_a \) and converter output voltage \( V_d \), in normalized form can be given as follows

\[
I_a(\text{pu}) = \frac{I_a}{3 V_m/\pi X} = \frac{X}{R} \left[ \cos \alpha - \frac{\pi V_c}{3 V_m} \right]
\]  

(2.2)

\[
V_d(\text{pu}) = \frac{V_d}{V_m} = \frac{3}{\pi} \cos \alpha
\]  

(2.3)

where \( X \) is the armature reactance \((\omega L)\), and \( \alpha \) is the converter firing angle. The peak line voltage \((V_m)\) can essentially be considered a constant, and therefore, \( V_d \) can be controlled linearly by \( V_s \) with cosine wave crossing technique, as indicated in Fig. 2.3.

In discontinuous conduction mode, the following armature circuit equations are valid

\[
I_a(\text{pu}) = \frac{I_a}{3 V_m/(\pi X)} = \frac{X}{R} \left[ \cos \left( \frac{\pi}{3} + \alpha \right) - \cos \left( \frac{\pi}{3} + \alpha + \theta_1 \right) - \frac{V_c}{V_m} \theta_1 \right]
\]  

(2.4)

\[
V_d(\text{pu}) = \frac{V_d}{V_m} = \frac{3}{\pi} \left[ \cos \left( \frac{\pi}{3} + \alpha \right) - \cos \left( \frac{\pi}{3} + \alpha + \theta_1 \right) - \frac{V_c}{V_m} \theta_1 \right] + \frac{V_c}{V_m}
\]  

(2.5)
\[ \frac{V_c}{V_m} = \frac{\sqrt{1 + \frac{(X/R)^2}{R}}}{1 - \exp\left(-\frac{X}{R\theta_1}\right)} \left\{ \sin\left(\frac{\pi}{3} + \alpha + \theta_1 - \phi\right) - \sin\left(\frac{\pi}{3} + \alpha - \phi\right) \exp\left(-\frac{R\theta_1}{X}\right) \right\} \]  

(2.6)

where \( \theta_1 \) is the conduction angle of current pulse \((0 < \theta_1 < \pi/3)\), as indicated in Fig.2.2(a), and \( \phi = \tan^{-1}(X/R) \). For a fixed \( X/R \) parameter, the equations 2.2 - 2.6 were plotted in Fig. 2.4 for different \( \alpha \) angles, which also indicates the boundary between continuous and discontinuous conduction modes. For example, at \( \alpha = 80^\circ \), the conduction is continuous at the point A. As the machine counter emf is increased, the \( V_d \) (pu) remains constant at decreasing \( I_a \) (pu) until point B, when the conduction becomes discontinuous. Further increase of counter emf will cause increase of \( V_d \) (pu) until it reaches the point at which \( I_a(pu) = 0 \), if the machine is not loaded. The nonlinear \( V_d(pu) - I_a(pu) \) relation adversely affects the gain characteristics of the current control loop. If, for example, the loop gain is made optimum at continuous conduction mode, the lower gain at discontinuous conduction will make the loop response sluggish. On the other hand, if the gain is optimized for discontinuous mode at a certain operating point, the loop will tend to be unstable at continuous conduction. This problem can be overcome through Fuzzy \( \Delta \alpha \) compensation. For instance, at operating point C in Fig. 2.4, if the proper \( \Delta \alpha \) angle is added to \( \alpha_0 = 80^\circ \), the compensated \( V_d \) moves to point D, that lies in the extension of the horizontal characteristics of the continuous conduction mode. As a consequence, the converter gain becomes the same as in the continuous conduction mode, and proper tuning of the current controller is now possible.
Fig. 2.4 Theoretical $V_d$-$I_a$ (pu) phase plot without compensation.
2.2.1 Fuzzy Linearization of Converter Characteristics

It was mentioned in Chapter 1 that fuzzy control is well-suited in a non-linear system, especially where parameter variation problem exists. Therefore, we applied fuzzy method of $\Delta \alpha$ angle compensation, in order to linearize the converter transfer characteristics in discontinuous conduction mode. The special feature in fuzzy control is that the $\Delta \alpha$ angle is expressed as a fuzzy relation of the variables $I_8$ and $\alpha$ angle. The set of rules for fuzzy compensation is given in matrix form in Table 2.1, where all the symbols are defined in the usual fuzzy logic terminology. A typical rule has the following structure:

**IF** $I_8$ is small negative (NS) AND $\alpha$ is small positive (PS)

**THEN** $\Delta \alpha$ is small negative (NS)

The rule base was developed by heuristic from the viewpoint of practical system operation. The current $I_8$ was represented in normalized form. Fig. 2.5 shows the membership function plots of the variables $\alpha$, $I_8(pu)$ and $\Delta \alpha$. The universes of discourse of the variables cover the whole discontinuous conduction region. The sensitivity of a variable determines the number of fuzzy sets required to describe it. The universe of discourse of $\alpha$ is described by five fuzzy sets, whereas $I_8(pu)$ and $\Delta \alpha$ are described by nine and eleven sets, respectively. The linguistic terms used for the sets are for convenience only and must be interpreted in a "context free" grammar, since their conventional meaning does not correspond to the sign and numerical values of the variables. In Fig. 2.5, 50% overlap was provided for the neighboring fuzzy sets. Therefore, at any given point of the universe of discourse, no more than two fuzzy sets
Table 2.1 Rule base for $\Delta \alpha$ compensation.

<table>
<thead>
<tr>
<th>$I_a$</th>
<th>$\alpha$</th>
<th>NB</th>
<th>NS</th>
<th>Z</th>
<th>PS</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>NVB</td>
<td>NVB</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
</tr>
<tr>
<td>NB</td>
<td>NVB</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
</tr>
<tr>
<td>NM</td>
<td>NVB</td>
<td>NS</td>
<td>NVS</td>
<td>NVS</td>
<td>NVS</td>
<td>NVS</td>
</tr>
<tr>
<td>NS</td>
<td>NVB</td>
<td>NM</td>
<td>NS</td>
<td>NS</td>
<td>NS</td>
<td>NS</td>
</tr>
<tr>
<td>Z</td>
<td>NVB</td>
<td>NB</td>
<td>NM</td>
<td>NM</td>
<td>NS</td>
<td></td>
</tr>
<tr>
<td>PS</td>
<td>NVB</td>
<td>NVB</td>
<td>NB</td>
<td>NM</td>
<td>NM</td>
<td>NM</td>
</tr>
<tr>
<td>PM</td>
<td>NVB</td>
<td>NVB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
</tr>
<tr>
<td>PB</td>
<td>NVB</td>
<td>NVB</td>
<td>NVB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
</tr>
<tr>
<td>PVB</td>
<td>NVB</td>
<td>NVB</td>
<td>NVB</td>
<td>NVB</td>
<td>NB</td>
<td>NB</td>
</tr>
</tbody>
</table>

25
Fig. 2.5 Membership functions for $\Delta \alpha$ compensation.

a) Firing angle ($\alpha$)  
b) Armature current ($I_a$)  
c) Compensating angle ($\Delta \alpha$)
have non-zero degree of membership. This choice of fuzzy partitioning along with the SUP-MIN composition method resulted in a simplification of the fuzzy linearization algorithm. It is evident that, for any input data of $I_a(pu)$ and $\alpha$, at the most four rules will be valid in the entire rule base given in Table 2.1. The algorithm for fuzzy linearization is described below, along with a numerical example, included for clarity.

1. Sample the DC current $I_a$ and firing angle $\alpha$ from the current control loop. Convert $I_a$ to $I_a(pu)$.

   [Let's assume: $I_a = 5.6$ A, $\alpha = 55^\circ$, $I_{base} = 98.5$ A (see Table 2.1)]

   $I_a(pu) = 5.6/98.5 = 0.057$

2. Calculate the interval indices $I$ and $J$ (that identify the interval number in the fuzzy sets) for $\alpha$ and $I_a(pu)$, respectively as follows:

   $I = \text{INT } ((\alpha + 10)/20)$

   $J = \text{INT } ((I_a(pu) + 0.01)/0.01)$

   [$I = 3$, $J = 6$]

3. Calculate the degree of membership of $\alpha$ and $I_a(pu)$ for the left most fuzzy subset using $I$ and $J$, respectively as follows:

   $\mu_Z(\alpha) = (20 I + 10 - \alpha) / 20$

   $\mu_{PS}(I_a(pu)) = (0.01 I - I_a(pu))/0.01$

   [$\mu_Z(55^\circ) = 0.75$, $\mu_{PS}(0.057) = 0.3$]

4. Evaluate degree of membership for other sets by complement relation

   $\mu_{PS}(\alpha) = 1 - \mu_Z(\alpha)$

   $\mu_{PM}(I_a(pu)) = 1 - \mu_{PS}(I_a(pu))$
[ \mu_{ps}(55^\circ) = 0.25, \mu_{pm}(0.057) = 0.7 ]

5. Identify the four valid rules in Table 2.1 (stored as look-up table) and calculate the degree of membership \( \mu_{R_i} \) contributed by each rule \( R_i \) \( [ i = 1, 2, 3, 4 ] \), using MIN operator

\[ \mu_{R_1} = \text{Min} \{ \mu_Z(\alpha), \mu_{ps}(I_a(pu)) \} = \text{Min} \{0.75, 0.3\} = 0.3 \]

\[ \mu_{R_2} = \text{Min} \{ \mu_{ps}(\alpha), \mu_{ps}(I_a(pu)) \} = \text{Min} \{0.25, 0.3\} = 0.25 \]

\[ \mu_{R_3} = \text{Min} \{ \mu_Z(\alpha), \mu_{pm}(I_a(pu)) \} = \text{Min} \{0.75, 0.7\} = 0.7 \]

\[ \mu_{R_4} = \text{Min} \{ \mu_{ps}(\alpha), \mu_{pm}(I_a(pu)) \} = \text{Min} \{0.25, 0.7\} = 0.25 \]

6. Retrieve the amount of correction \( \Delta\alpha_i \), \( i = 1, 2, 3, 4 \) corresponding to each rule, from Table 2.1

\( \Delta\alpha_1 = (\alpha = Z, I_a(pu) = PS) \rightarrow \Delta\alpha_1 = NB = 3^\circ \)

\( \Delta\alpha_2 = (\alpha = PS, I_a(pu) = PS) \rightarrow \Delta\alpha_2 = NM = 6^\circ \)

\( \Delta\alpha_3 = (\alpha = Z, I_a(pu) = PM) \rightarrow \Delta\alpha_3 = NB = 3^\circ \)

\( \Delta\alpha_4 = (\alpha = PS, I_a(pu) = PM) \rightarrow \Delta\alpha_4 = NB = 3^\circ \)

7. Calculate the crisp value of \( \Delta\alpha \) by height defuzzification method as follows:

\[ \Delta\alpha = (0.30*3 + 0.25*6 + 0.70*3 + 0.25*3) / (0.30 + 0.25 + 0.70 + 0.25) = 3.5 \]

The strength of fuzzy compensation is that the number of rules required to express the fuzzy relation is fairly small and memory requirement is low compared to large look-up table needed in conventional method.
2.2.2 Fuzzy Control of Current and Speed Loops

In addition to converter linearization, fuzzy logic control was applied to the speed and current loops as well, replacing the conventional PI controllers. The objective was to explore the control robustness in the presence of parameter variation and load disturbance effect. However, both loops must satisfy the requirements of fast transient response with minimum overshoot. With converter linearization, both speed and current loops have essentially first order characteristics. Therefore, intuitively the same fuzzy control strategy should be valid for both.

The selected fuzzy controller structure possesses two input variables, namely error (E) and change in error (CE), and one output variable, the change in control setting (ΔU). Such controller can be viewed as a generalization of the conventional PI controller, where the effective gains $K_p$ and $K_i$ are dependent on the input state (E,CE). The input variables considered in the fuzzy rule base are defined as:

\[
E(k) = R(k) - C(k)
\]

\[
CE(k) = E(k) - E(k-1)
\]

where $R(k)$ = reference signal, $C(k)$ = output signal and $k$ = sampling interval. The control strategy is described linguistically by rules of the following format:

\[
\text{IF } E(k) \text{ is X AND CE(k) is Y THEN } \Delta U(k) \text{ is Z}
\]

where $\Delta U(k)$ is the change in the control setting, X, Y and Z are the fuzzy sets defined in the universe of discourse of E, CE and $\Delta U$, respectively. The variables can be expressed as per unit quantities as follows:

\[
e(\text{pu}) = E(k)/GE
\]
ce(pu) = CE(k)/GCE

δu(pu) = ΔU(k)/GU

where GE, GCE and GU are the respective gain factors of the controller. The gain factors are normally different for speed and current control loops. The representation of the variables in terms of per unit values permits flexibility in the design and tuning of the controller. Fig. 2.6 shows the membership functions of e(pu), ce(pu) and δu(pu) variables. Note that the fuzzy sets for each variable have asymmetrical shape causing more crowding near the origin. This permits precision control near the steady state operating point, without unduly increasing the number of sets. However, a finer partitioning for δu(pu) was necessary considering the sensitivity of this variable. As fuzzy controller design is based on intuition and experience, instead of the system model, the following considerations were given in the beginning:

1. If both e(pu) and ce(pu) are zero, then maintain the present control setting U(k) (i.e., δu(pu) = 0);

2. If e(pu) is not zero, but is approaching this value at satisfactory rate, then maintain the present control setting U(k);

3. If e(pu) is growing, then change in the control signal δu(pu) is not zero and its value depends on the magnitude and sign of e(pu) and ce(pu) signals [7].

Table 2.2 gives the rule base matrix for current and speed controllers. A close look at the rule base indicates that the auxiliary diagonal consists of Z (zero) elements which conform to the second consideration as given above. Note that the value assigned to δu(pu) depends on the distance from the auxiliary diagonal. For instance, if e=PS and
Fig. 2.6 Membership functions of speed and current controllers.

a) Error  b) Change in error  
c) Change in control
ce=NB, the system is approaching the steady state point \((e=Z, ce=Z)\) too fast, and \(\delta u\) is made NM to prevent a large overshoot. On the other hand, if \(e=PS\) but \(ce=NS\), the system is converging to the steady state point at the desired rate, and no change in control is required, i.e., \(\delta u=Z\). The control strategy is dominantly affected by changes in the rule base, and these constitute the primary means for controller tuning.

Table 2.2 Rule base for current and speed controllers.

<table>
<thead>
<tr>
<th>ce</th>
<th>NB</th>
<th>NM</th>
<th>NS</th>
<th>Z</th>
<th>PS</th>
<th>PM</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB</td>
<td>NVB</td>
<td>NVB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
</tr>
<tr>
<td>NM</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
</tr>
<tr>
<td>NS</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
</tr>
<tr>
<td>Z</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
</tr>
<tr>
<td>PS</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
</tr>
<tr>
<td>PM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PM</td>
<td>PM</td>
<td>PB</td>
</tr>
<tr>
<td>PB</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PM</td>
<td>PVB</td>
<td>PVB</td>
</tr>
</tbody>
</table>
A finer tuning was obtained by iterating the parameters $e_1, e_2, \ldots, c_1, c_2, \ldots, u_1, u_2, \ldots$ in Fig. 2.6, until the desired controller performance was obtained. The control procedure is essentially identical to that of $\Delta \alpha$ compensation scheme. The steps for speed control can be summarized as follows:

1. Sample $\omega_r^*$ and $\omega_r$.

2. Compute error (E), change in error (CE) and their per unit values as follows:
   \[
   E(k) = \omega_r^*(k) - \omega_r(k)
   \]
   \[
   CE(k) = E(k) - E(k-1)
   \]
   \[
   e(\text{pu}) = E(k)/GE
   \]
   \[
   ce(\text{pu}) = CE(k)/GCE
   \]

3. Identify the interval indices $I$ and $J$ (1, 2, etc. shown in Fig. 2.6) for $e(\text{pu})$ and $ce(\text{pu})$, respectively, by comparison method.

4. Compute the degree of membership of $e(\text{pu})$ and $ce(\text{pu})$ for the relevant fuzzy sets.

5. Identify the four valid rules in Table 2.2 and calculate the degree of membership $\mu_{R_i}$ using MIN operator.

6. Retrieve $\delta u_i$ for each rule from Table 2.2.

7. Calculate the resultant crispy value of $\delta u(\text{pu})$ by height defuzzification method.

8. Compute the next control signal as
   \[
   U(k) = U(k-1) + GU \ast \delta u(\text{pu})
   \]

The control for the current loop is the same as above, except here the gain factors GE, GCE and GU are different.
2.3 Simulation Study

In order to validate the control strategies discussed above, digital simulation studies were made using PC-SIMNON language. The phase-controlled rectifier and DC machine SIMNON models were initially developed and tested. Next, the fuzzy $\Delta \alpha$ compensator was constructed, along with the current and speed fuzzy controllers, and integrate to the other sub-systems. All SIMNON routines are listed in the appendix. Although SIMNON is well suited for linear and non-linear systems simulation, it does not support branch instructions, what complicates the programming of fuzzy controllers. A complete description of the simulation methodology will be postponed to Chapter 4.

Table 2.3 shows the parameters of the drive system used in the simulation study. The speed and current loops of the drive were also designed and simulated with PI control, in order to compare performance with the respective fuzzy control loop. The compensation and feedback control algorithms were iterated until best simulation results were obtained.

In the beginning, the performance of the fuzzy compensation scheme was tested keeping both the speed and current loops open. Fig. 2.7 shows the voltage and current responses without compensation. Initially, $V_s$ was set such that $\alpha = 70^\circ$, and then the drive simulation was enabled. With inertia load, the machine speed builds up freely with the developed torque. Initially, the conduction is continuous and $V_d = V_d^* = KV_s$. As speed builds up, the higher counter emf forces the converter to enter into discontinuous conduction, and $V_d$ rises above the reference value $V_d^*$ as shown. The experiment was then repeated with fuzzy $\Delta \alpha$ compensation enabled, for different values of $\alpha$ angle.
Table 2.3 Parameters of DC drive system.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC Motor:</td>
<td></td>
</tr>
<tr>
<td>Voltage</td>
<td>110 V</td>
</tr>
<tr>
<td>HP</td>
<td>2.5</td>
</tr>
<tr>
<td>RPM</td>
<td>1800</td>
</tr>
<tr>
<td>Resistance</td>
<td>0.6 Ω</td>
</tr>
<tr>
<td>Inductance</td>
<td>8 mH</td>
</tr>
<tr>
<td>Moment of inertia</td>
<td>0.0465 Kg.m²</td>
</tr>
<tr>
<td>Damping</td>
<td>0.004 N.m.sec/rad</td>
</tr>
<tr>
<td>Back EMF constant</td>
<td>0.55 V.sec/rad</td>
</tr>
<tr>
<td>Current</td>
<td>20 A (rated value)</td>
</tr>
<tr>
<td>Line voltage to transformer</td>
<td>90 V</td>
</tr>
<tr>
<td>Load torque</td>
<td>$2.78 \times 10^{-4}$ N.m.sec²/rad</td>
</tr>
</tbody>
</table>

$T_L = K_L \omega_r^2$ (load torque)
Fig. 2.7 Open loop $V_d$ step ($\alpha = 70^\circ$) response without compensation.

a) Reference (1) and actual (2) dc voltages.

b) Armature current (dc value).
The corresponding $V_d(\text{pu}) - I_a(\text{pu})$ phase plots are given in Fig. 2.8. It is evident that with fuzzy $\Delta \alpha$ compensation, $V_d(\text{pu})$ at a certain angle essentially remains constant for the whole region of $I_a(\text{pu})$, i.e., $V_d$ is essentially equal to $V_d^\ast$ in both continuous and discontinuous regions. Fig. 2.9 shows the corresponding time domain plot of converter DC voltage ($v_d$) and armature current ($i_a$) waves with $\Delta \alpha$ compensation for $\alpha = 70^\circ$, where time starts at 0.46 sec.

The current control loop was then tested with the fuzzy controller with the speed loop remaining open, but the speed was locked to a fixed value by assuming a very large inertia, so as to establish the discontinuous conduction mode. Fig. 2.10(a) shows the current loop response with $\Delta \alpha$ compensation which can be compared with Fig. 2.10(b) that gives response without compensation. In both cases, actual current waveform is shown. The boost of transient response due to converter linearization is evident. The responses of the current loop with PI control, with and without $\Delta \alpha$ compensation, were obtained and shown in Fig. 2.11 for comparison purpose. The response improvement in Fig. 2.10 for either condition indicates the superiority of fuzzy control.

Next, the speed loop was closed, and transient response was tested with both fuzzy current and speed control at linearized converter condition. Fig. 2.12 shows the speed and current response that covers both continuous and discontinuous regions. The figure also shows the effect of 40% step load torque applied at 0.8 sec. Fig. 2.13 shows the
Fig. 2.8 Phase plot $V_d$-$I_a$ with $\Delta \alpha$ compensation.
Fig. 2.9 Open loop $V_d$ step response with $\Delta \alpha$ compensation.

a) Armature current (actual value).

b) Converter output voltage (actual value).
Fig. 2.10 Current loop response of the fuzzy controller.

a) With $\Delta \alpha$ compensation.

b) Without $\Delta \alpha$ compensation.
Fig. 2.11 Current loop response of the PI controller.

a) With $\Delta \alpha$ compensation.

b) Without $\Delta \alpha$ compensation.
Fig. 2.12 Fuzzy control system response to a $\omega_r^*$ step and $T_L$ step.

a) Speed. b) Armature currents.
Fig. 2.13 PI control system response to a $\omega_r^*$ step and $T_L$ step.

a) Speed. b) Armature currents.
corresponding system response under PI control in both loops. Table 2.4 summarizes the response improvement under fuzzy control.

Finally, the drive system was tested with $\omega_r^*$ step at the same condition as before but with four times the effective inertia load. Fig. 2.14 shows the response with fuzzy control, and Fig. 2.15 gives the response with PI control, for comparison. Although the major portion of the rise time occurs with the current loop saturated, some improvement in rise time and overshoot under fuzzy control is evident.

In conclusion, the simulation study demonstrated the successful application of fuzzy logic to a phase-controlled converter DC drive system. The proposed fuzzy $\Delta \alpha$ compensation scheme effectively linearizes converter characteristics at discontinuous conduction mode, and is simpler than the conventional look-up table method which requires a large memory. Fuzzy logic was also applied to the design of current and speed controllers, and the performance was compared with that of a conventional PI controlled system. The simulation study clearly indicated the superior performance of fuzzy control. The reason for superior performance of fuzzy controlled system is that basically it is adaptive in nature, and the controller is able to realize different control law for each input state (E and CE). The response of PI controlled system, on the other hand, is sensitive to model change that occurs with parameter variation. The main drawback in fuzzy control is the lack of a well defined design methodology, that results in a time consuming heuristic tuning process. Furthermore, a high speed microprocessor is needed to implement the control in real time. However, the control law is simpler to implement than expert system and neural network techniques.
Table 2.4 Performance comparison of fuzzy and PI controlled drive system.

<table>
<thead>
<tr>
<th></th>
<th>Fuzzy</th>
<th>PI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rise time (s)</td>
<td>0.410</td>
<td>0.425</td>
</tr>
<tr>
<td>Overshoot (rpm)</td>
<td>13</td>
<td>16</td>
</tr>
<tr>
<td>Speed drop with $T_L$ (rpm)</td>
<td>13</td>
<td>25</td>
</tr>
<tr>
<td>Recovery time with $T_L$ (s)</td>
<td>0.09</td>
<td>0.20</td>
</tr>
</tbody>
</table>
Fig. 2.14  Fuzzy control system response to a \( \omega_r^* \) step with a new inertia of four times the original value.

a) Speed.       b) Armature currents.
Fig. 2.15 PI control system response to a $\omega_r^*$ step with a new inertia of four times the original value.

a) Speed.  b) Armature currents.
CHAPTER 3

LOSS MODELING OF CONVERTER INDUCTION MACHINE SYSTEMS

3.1 Introduction

Precise and reliable loss models for induction motor and converter systems are very important for performance prediction of variable speed drives. In particular, for the efficiency optimization study discussed in Chapter 4, it was realized that adequate converter and induction machine loss models were not available in the literature. In spite of the numerous studies that have been reported on the subject, in most cases the study focus on some particular aspects of interest and tend to neglect the overall picture, mainly because of the extreme complexity of the loss phenomena in induction machines. Traditionally, the machine electrical losses have been studied by using some variation of the per-phase equivalent circuit of Fig. 3.1, because the losses become primarily important at steady state condition. On the other hand, for dynamic studies, the synchronously rotating frame $D^*Q^*$ model of Fig. 3.2 is normally used, but in this case the losses are not properly represented. Even under sinusoidal supply, a precise evaluation of machine losses is not straightforward, but it becomes much more complex when the machine is fed by a inverter, due to the wide harmonic spectrum present in the impressed voltage. Nevertheless, in order to predict the performance of the fuzzy logic based efficiency
Fig. 3.1 Per-phase equivalent circuit.

Fig. 3.2 Standard D^e -Q^e equivalent circuits in synchronous frame.
   a) Q^e - axis circuit   b) D^e - axis circuit.
optimization control discussed in Chapter 4, it was necessary to derive a machine model capable of representing both the dynamic behavior and the loss characteristics of the induction machine, as well as a loss model for the converter system.

Loss modeling of induction motor has received wide attention in the literature over a number of years. Neglecting the effects of space harmonics, the machine parameters become dependent of time harmonic frequencies impressed by the inverter. Klingshirn and Jordan [30] applied superposition principle to calculate harmonic currents and the corresponding losses with per-phase equivalent circuit, where the rotor resistance and leakage inductance for each harmonic were corrected for deep bar (skin) effect. Kawagishi et al. [31] were able to verify experimentally the frequency dependency of parameter, by using a high frequency power supply, and validate some of the theoretical predictions. Honsinger [25] systematically studied the losses for a six-step inverter-fed machine and propose harmonic per-phase equivalent circuit. Since both core loss and stray load loss are basically due to hysteresis and eddy current effects, he proposed representation of stray loss by frequency dependent resistance in parallel with leakage inductance in the equivalent circuit. More recently, Udayagiri and Lipo [27] proposed a new simulation model that incorporates core loss but neglects the skin effect and leakage flux induced core loss, thereby underestimating the total loss.

In high frequency PWM inverter system, both conduction loss and switching loss are important in the loss model. The switching loss were discussed analytically by McMurray [32,] where the effects of both turn-on and turn-off snubbers were considered. Jovanovich et al. [33] made experimental evaluation of switching characteristics and
losses for a number of power devices with different base drives and load conditions. Ikeda et al. [34] have proposed loss modeling of PWM voltage-fed inverter and discussed the effect of carrier frequency on inverter losses. Circuit simulation programs, such as PSPICE that embed the detailed model of devices can give realistic lossy converter simulation. However, the drawbacks are that the losses remain somewhat transparent and can not be easily partitioned between the conduction and switching losses. Besides, such programs are not convenient for drive system simulation.

In the next sections, a unified loss model for the variable frequency drive system of Fig. 3.3 is discussed. The machine electrical losses, such as stator and rotor copper loss, core loss and stray loss, were considered for both fundamental and harmonic frequencies. Also considered were the skin effect on rotor resistance, temperature effect on both stator and rotor resistances, magnetizing inductance saturation, and friction and windage loss. All the above features were incorporated in a synchronous frame dynamic $D^o-Q^o$ equivalent circuit. The converter system, consisting of a diode rectifier and PWM transistor inverter, was modeled accurately for conduction and switching losses. Validity of the models, in both steady-state and transient conditions, was verified by simulations. These results were first presented at the 1992 international conference on industrial electronics, control and instrumentation (IECON'92), [42].

3.2 Dynamic Lossy Model for the Induction Machine

The derivation of the dynamic lossy model is presented here. A discussion on machine losses is followed by the construction of a suitable harmonic equivalent circuit,
Fig. 3.3 Converter-machine system for variable speed drive.
and finally, by the derivation of the complete model through superposition principle.

3.2.1 Induction Machine Losses

Machine electrical and mechanical losses are examined here both qualitatively and quantitatively. The analysis follows a conventional partition of losses, encountered in the literature.

Copper Losses

Copper or Ohmic losses are caused by the non-ideal characteristics of the conductors employed in the stator and rotor windings, and have been modeled by

$$P_{cu} = m R I^2$$  \hfill (3.1)

where \( m \) is the number of phases, \( R \) is the winding resistance and \( I \) the rms current. Proper evaluation of these losses requires the consideration of temperature effect on winding resistance, as well as its frequency dependency, commonly referred to as skin effect.

The skin effect has been widely discussed in the literature. In inverter-fed machine, the skin effect due to fundamental slip frequency can be ignored, but for the harmonic frequencies the rotor appears almost stationary, and therefore, practically all the stator harmonic currents flow in the rotor creating dominant skin effect. The rotor resistance at harmonic frequency \( f_n \) is given approximately by [29]
\[ R_{rn} = R_{rdc} (1 + c_1 d f_n^{0.5}) \]  

where \( R_{rdc} \) is the dc resistance, \( d \) = bar depth, and \( c_1 \) is a constant that takes into account the bar material and shape. With a number of harmonic frequencies, the superposition principle can be applied approximately by assuming that machine parameters for all harmonic frequencies are identical to those computed at carrier frequency. For a PWM inverter with sinusoidal PWM or hysteresis-band current control, the carrier frequency and the near sidebands are most dominant.

### Core Losses

Core losses consist of hysteresis and eddy-current losses, created by the time varying flux in the machine laminations. The eddy current loss may be expressed as

\[ P_e = k_e f^2 \psi_m^2 \]  

where \( \psi_m \) is the maximum mutual flux linkage, \( f \) the supply frequency and \( k_e \) a coefficient whose value depends on machine geometry and size, lamination thickness and resistivity of the iron. The hysteresis loss has been empirically modeled as

\[ P_h = k_h f \psi_m^2 \]  

where the coefficient \( k_h \) is mainly influenced by the magnetic properties of the iron and machine size. The stator core loss \( P_{cs} \) due to fundamental frequency mutual flux \( \psi_m \) can be expressed by

\[ P_{cs} = k_h f \psi_m^2 + k_e f^2 \psi_m^2 \]  

Under normal operation, the fundamental rotor frequency is a small fraction of the stator
frequency and the corresponding rotor core loss $P_{cr}$ is given by

$$P_{cr} = k_h s f \psi_m^2 + k_e (s f)^2 \psi_m^2$$  \hspace{1cm} (3.6)$$

where $f$ has been substituted for $s f$, $s$ being the per unit slip. Equations 3.5 and 3.6 can be added and rearranged to get the total fundamental core loss as

$$P_c = P_{cs} + P_{cr} = \left[ k_h \left( \frac{1+s}{f} \right) + k_e (1 + s^2) \right] f^2 \psi_m^2$$  \hspace{1cm} (3.7)$$

As the mutual or air-gap flux linkage $\psi_m$ is related to air-gap voltage $V_m$ by

$$\psi_m = \sqrt{\frac{k_e}{f}} \frac{V_m}{f}$$  \hspace{1cm} (3.8)$$

(3.7) can be rewritten as

$$P_c = k_c \left[ k_h \left( \frac{1+s}{f} \right) + k_e (1 + s^2) \right] V_m^2$$  \hspace{1cm} (3.9)$$

The equivalent core loss resistance $R_m$ can then be derived as

$$R_m = \frac{1}{k_c \left[ k_h \left( \frac{1+s}{f} \right) + k_e (1 + s^2) \right]}$$  \hspace{1cm} (3.10)$$

Assuming that the core losses due to mutual harmonic flux are governed by the same principles that control the fundamental losses, the coefficients $k_h$ and $k_e$ remain the same at harmonic frequency, and since harmonic slip $s_n \approx 1$, the equivalent core loss resistance $R_{mn}$ at frequency $f_n$ can be obtained from (3.10) as

$$R_{mn} = \frac{0.5}{k_c \left[ k_h + k_e \right]}$$  \hspace{1cm} (3.11)$$
Stray Losses

The stray losses by definition are the excess of the total losses actually occurring in a motor at a given load current, over the sum of the calculated copper losses, the no-load core loss, and the friction and windage loss, according to Alger et al. [35]. They used empirical equations to evaluate each individual loss component, what requires the knowledge of machine dimensions, type of core material, lamination thickness, winding geometry, etc. In this work, however, instead of evaluating stray loss individually, we treated them as a whole. The fundamental idea is that, the stray loss is essentially due to eddy current and hysteresis losses, induced by various types of leakage fluxes in the laminations and other structural parts of the machine. Therefore, the stray loss can be modeled in a way similar to that used for core loss modeling. The stator per phase stray loss at harmonic frequency $f_n$ can be given as

$$P_{sln} = k_{sln} \left[ \frac{k_h}{f_n} + k_e \right] V_{sln}^2$$  \hspace{1cm} (3.12)

where $V_{sln}$ is the voltage across the stator leakage inductance and $k_{sln}$ the stray loss constant. Therefore, the stray loss can be represented by an equivalent resistance $R_{sln}$ in parallel with the leakage inductance as

$$R_{sln} = \frac{1}{k_{sln} \left[ \frac{k_h}{f_n} + k_e \right]}$$  \hspace{1cm} (3.13)

A similar expression was derived for rotor harmonic stray loss. The stray loss due to fundamental current is essentially concentrated in the stator, and an equation similar to (3.13) could also be used. However, the fundamental stray loss was represented by a
resistance in series with the stator leakage reactance $X_{ls}$, for reasons that will become clear later. The fundamental voltage drop $V_{sll}$ across the leakage reactance $X_{ls}$ can be expressed as $(2\pi f L_{ls} I_{s1})$, where $I_{s1}$ is the fundamental stator current. It can be substituted into (3.12) to derive fundamental per-phase stray loss $P_{sll}$ as

$$P_{sll} = k_{sll} [k_h f + k_e f^2] I_{s1}^2 = R_{sll} I_{s1}^2$$  \hspace{1cm} (3.14)

where $R_{sll}$ is the equivalent series resistance. From this expression, $R_{sll}$ is given as

$$R_{sll} = k_{sll} [k_h f + k_e f^2]$$  \hspace{1cm} (3.15)

Friction and Windage Losses

The friction and windage losses are essentially a function of motor speed $\omega_r$ and does not depend on the type of power supply. It can be expressed as

$$P_{fw} = k_{fw} \omega_r^3$$  \hspace{1cm} (3.16)

3.2.2 Temperature and Saturation Effects

The preceding discussion on machine losses did not include the effects of temperature on winding resistances, nor the saturation effects on machine inductances. These effects will now be considered.
Temperature Effects

Both stator and rotor resistances increase with temperature. The stator temperature can be monitored and approximate correction factor can be applied, but there is no easy way to measure or estimate the rotor temperature. Precise prediction of temperature in each part of the machine requires detailed dynamic thermal model that depends on machine geometry, material characteristics, cooling effects, etc, and is extremely difficult to estimate. The machine transient thermal response can be given approximately by a first order model, where the temperature rise $\Delta T$ is expressed as

$$\Delta T = \frac{P_h}{\theta (1 + \tau s)}$$

where $P_h$ is the total machine loss, $\theta$ is the steady-state thermal resistance, and $\tau$ is the thermal time constant. The $\theta$ and $\tau$ parameters can be estimated approximately by experimentation. Both rotor and stator resistances were corrected for temperature effects by using the well known formula

$$R_{T2} = R_{T1} (1 + \alpha_{T1} (T_2 - T_1))$$

where $\alpha_{T1} =$ temperature coefficient (usually at $T_1=25 ^\circ C$), and $\Delta T = (T_2 - T_1)$. The temperature corrected resistances were then used to calculate fundamental and harmonic copper losses. For harmonic rotor losses, the skin effect was superimposed on the temperature effect.
Saturation Effects

Although saturation is strictly present in both leakage and magnetizing inductances, it was ignored in the former, and represented in the magnetizing inductance \( L_m \) by a piece-wise linear function of magnetizing current \( I_m \) as

\[
L_m = \begin{cases} 
L_{mo} & \text{if } I_m \leq I_{mo} \\
L_{mo} - m(I_m - I_{mo}) & \text{if } I_m > I_{mo}
\end{cases}
\]

(3.19)

where \( L_{mo} \) is the unsaturated inductance and \( I_m \) is the magnetizing current at the start of saturation. The saturation coefficient \( m \) was selected to best fit the actual saturation curve of the machine.

3.2.3 Per-phase Harmonic Equivalent Circuit

The effects of time harmonics have been traditionally investigated by solving the per-phase equivalent circuit [25] shown in Fig. 3.4(a), where the harmonic stray losses are represented by shunt resistances \( R_{\text{ssn}} \) and \( R_{\text{tsn}} \). For each harmonic component, the circuit is solved, and superposition principle is applied to get the overall harmonic effect [25]. In this way, the frequency dependence of machine parameters can be taken into account precisely. The following simplifying assumptions can be made at this point:

- For sinusoidal PWM or hysteresis-band current-controlled inverter, only the carrier frequency is considered for computation of frequency dependent parameters, and the resulting circuit can be used to compute the effect of all the harmonics with little loss of precision.
- The harmonic frequencies are sufficiently high such that the harmonic slip \( s_n \) is essentially one.
Fig. 3.4 Per phase harmonic equivalent circuit of the induction motor.

a) Generic circuit for harmonic of order n.

b) "Series" equivalent form of circuit (a).

c) Modified "shunt" equivalent form of circuit (b).
By applying these assumptions, Fig 3.4(a) was converted to the series equivalent form of Fig. 3.4(b). The barred parameters are simply the series equivalents of the corresponding original parameters. For example, the series equivalent stator stray loss resistance can be expressed as

\[
\bar{R}_{ssn} = \frac{R_{ssn} X_{ssn}^2}{R_{ssn}^2 + (X_{ssn} + X_{tsn})^2} = \frac{R_{ssn} X_{ssn}^2}{R_{ssn}^2 + X_{tsn}^2}
\]

(3.20)
since the secondary leakage reactance \(X_{ssn}\) is very small. The harmonic rotor resistance \(R_m\) is shown split into fundamental rotor resistance \(R_r\) and \((R_m - R_r)\). Similarly, \(R_{ssn}\) is shown as the sum of the fundamental frequency stray loss resistance \(R_{sl1}\) and \((R_{ssn} - R_{sl1})\). The harmonic core loss resistance \(R_{mn}\) was substituted in Fig. 3.4(b) by a series combination of the fundamental core loss resistance \(R_m\) and a modified secondary magnetizing reactance \(X_{lmm}'\), so as to ensure constancy of harmonic core loss \(P_{cln}\). From Fig. 3.4(a), the harmonic core loss \(P_{cln}\) (neglecting small \(X_{lmm}\)) is given as

\[
P_{cln} = \frac{3 V_{mn}^2}{R_{mn}}
\]

(3.21)

where \(V_{mn}\) is the rms harmonic airgap voltage. From Fig. 3.4(b), \(P_{cln}\) is given by

\[
P_{cln} = \frac{3 V_{mn}^2 R_m}{R_m^2 + X_{lmm}^{'2}}
\]

(3.22)

In order to keep the harmonic core loss invariant, the two equations must yield the same result. By equating the two expressions, the modified secondary magnetizing reactance was derived as follows:

\[
X_{lmm}' = \sqrt{R_{mn} R_m - R_m^2}
\]

(3.23)
The circuit of Fig. 3.4(b) was next transformed into the modified "shunt" form of Fig. 3.4(c). The final values of the harmonic stray loss resistances $R_{ssn}'$ and $R_{rnn}'$ were obtained by equating the corresponding resistive terms in Figs. 3.4(b) and 3.4(c). Neglecting the small $X_{rnn}'$ the following expression was derived:

$$\frac{R_{rnn}' X_{brn}}{R_{rnn}'^2 + X_{brn}^2} = R_{rle}$$  \hspace{1cm} (3.24)

where $R_{rle} = R_{rnn} + (R_m - R_r)$. Solving for $R_{rnn}'$,

$$R_{rnn}' = \frac{X_{brn}^2 \pm \sqrt{(X_{brn}^2)^2 - 4 X_{brn}^2 \left(\frac{1}{R_{rle}}\right)}}{2}$$  \hspace{1cm} (3.25)

For most practical drives, $R_{rnn} > X_{brn}$ and therefore, the plus sign is considered in the above equation. With a similar procedure, the expression for $R_{ssn}'$ was also derived. In practice, the value of $R_{m1}$ is very small compared to $R_{ssn}$. Therefore, $R_{ssn}'$ could be taken equal to $R_{ssn}$. Note that $R_{rnn}'$ represents not only the rotor harmonic stray loss, but also the additional harmonic copper loss due to skin effect.

### 3.2.4 Synchronous Frame Lossy Equivalent Circuits

The per-phase equivalent circuit derived in Fig. 3.4(c) is only valid for steady-state operation, and cannot be used for dynamic performance study. Usually, synchronously rotating frame $D^o-Q^o$ equivalent circuits are used for dynamic study. The standard $D^o-Q^o$ equivalent circuits cannot directly incorporate a core loss resistor in parallel with magnetizing inductance because DC current (equivalent fundamental frequency current)
will not flow through it. In this section, it is described how these circuits were modified to incorporate the core loss resistor, and then how the harmonic equivalent circuits were superimposed to them to derive the unified lossy equivalent circuits.

**D^s-Q^s Equivalent Circuits with Core Loss Resistor**

The stationary frame D^s-Q^s equivalent circuits can easily incorporate core loss resistance in parallel with magnetizing inductance, as shown in Fig. 3.5. With this modifications the following equations can be readily written:

\[
R_s \left( i_{qs} \right) + L_{ls} \frac{d}{dt} \left( i_{qs} \right) + R_m \left( i_{qm} \right) = \left( v_{qs} \right) \quad (3.26)
\]

\[
-\omega_e \begin{pmatrix} \Psi_{dr}^s \\ \Psi_{qr}^s \end{pmatrix} + R_r \left( i_{qr} \right) + L_{dr} \frac{d}{dt} \left( i_{dr} \right) + L_{m} \frac{d}{dt} \left( i_{dm} \right) = \left( v_{qr}^s \right) \quad (3.27)
\]

Synchronous frame quantities are related to their stationary frame counterparts by a transformation matrix \( T \), defined as

\[
T = \begin{bmatrix} \cos \omega_e t & \sin \omega_e t \\ -\sin \omega_e t & \cos \omega_e t \end{bmatrix} \quad (3.29)
\]

where \( \omega_e \) is the synchronous frequency, \( \cos \omega_e t \) and \( \sin \omega_e t \) are the unit vectors. For the stator currents, the relationship can be expressed as
Fig. 3.5 $D'\cdot Q'$ equivalent circuit with core loss resistance.
\[
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} = T
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix}
\] (3.30)

Substitution of (3.30) and similar relations for the remaining variables into (3.26) results in

\[
R_s T
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} + L_b \frac{d}{dt}
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} + R_m T
\begin{pmatrix}
i_{qrn}^s \\
i_{drm}^s
\end{pmatrix} = T
\begin{pmatrix}
v_{qs}^s \\
v_{ds}^s
\end{pmatrix}
\] (3.31)

Pre-multiplying (3.31) by the inverse transformation \( T^{-1} \) yields

\[
R_s
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} + L_b T^{-1} \frac{d}{dt}
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} + R_m
\begin{pmatrix}
i_{qrn}^s \\
i_{drm}^s
\end{pmatrix} = T
\begin{pmatrix}
v_{qs}^s \\
v_{ds}^s
\end{pmatrix}
\] (3.32)

As \( T \) is a function of time, the product rule must be used to evaluate the derivative, and the second term can be rewritten as

\[
L_b T^{-1} \frac{d}{dt}
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} = L_b \left\{ T^{-1} \frac{dT}{dt}
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} + T^{-1} T \frac{d}{dt}
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} \right\}
\]

\[
= L_b \left\{ \begin{pmatrix} 0 & \omega_e \\ -\omega_e & 0 \end{pmatrix}
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} + \frac{d}{dt}
\begin{pmatrix}
i_{qs}^s \\
i_{ds}^s
\end{pmatrix} \right\}
\] (3.33)

Substitution of the last form of (3.33) into (3.32) produces
where $\psi_{dls} = L_{ls} i_{ds}$ and $\psi_{qls} = L_{ls} i_{qs}$ are the stator leakage fluxes, and the new terms $\omega_e \psi_{dls}$ and $-\omega_e \psi_{qls}$ represent the speed voltages due to rotation of the reference axes. By applying the same methodology to (3.27) and (3.28), their synchronous frame counterparts can be obtained as

$$\omega_e \left( \psi_{dm} + L_m \frac{d}{dt} (i_{qm}) - R_m (i_{qrm}) \right) = (0)$$

$$\left( \omega_e - \omega_p \right) \left( \psi_{dr} + R_r (i_{qr}) + L_{lr} \frac{d}{dt} (i_{dr}) \right) + L_m \frac{d}{dt} (i_{qm}) = (v_{qr})$$

Equation 3.34 through 3.36 can be represented by the equivalent circuits of Fig. 3.6, where $\psi_{dm} = L_m i_{dm}$ and $\psi_{qm} = L_m i_{qm}$ are the air-gap fluxes, $\psi_{dr} = \psi_{dm} + L_{lr} i_{dr}$ and $\psi_{qr} = \psi_{qm} + L_n i_{qr}$ are the rotor fluxes.

**D$^e$-Q$^e$ Equivalent Circuits with Core Loss and Harmonic Loss**

Although the harmonic equivalent circuit of Fig. 3.4(c) is a per phase stationary frame circuit, it can be used in the synchronously rotating frame as well, with little loss of precision, because the axes rotation will have the effect of adding (or subtracting) the fundamental frequency to the particular harmonic frequency. Consequently, under the assumption $f_p >> f$, the effect of axes rotation is negligible. The circuit of Fig. 3.4(c) was therefore, superimposed on the synchronously rotating frame D$^e$-Q$^e$ equivalent circuits with core loss resistance of Fig. 3.6, resulting in the complete lossy dynamic D$^e$-Q$^e$...
Fig. 3.6 $D^e$-$Q^e$ equivalent circuit with core loss resistance.
equivalent circuits shown in Fig. 3.7. The model represents true physical behavior of the machine, and can be used for evaluation of fundamental and harmonic losses, as well as for dynamic studies. Since at steady-state conditions all fundamental variables appear as DC quantities, the fundamental current will not flow through $R'_{ssn}$ and $R'_{rsn}$, i.e., the harmonic stray loss shunt terms have no effect on fundamental losses. Similarly, the presence of the secondary magnetizing inductance $L'_{lm}$ has no effect on fundamental core loss. For harmonic losses, the circuits of Fig. 3.7 give almost the same result as that of Fig. 3.4(c). This follows from the fact that the harmonic fluxes are very small, and therefore, the corresponding harmonic voltages present in the dependent voltage sources are negligible, i.e., the terms such as $\omega_c \psi_{dm}$ essentially represent fundamental counter emf. During transient operations, some amount of fundamental current will flow through the shunt stray loss resistances, and this is theoretically coherent, because a change in fundamental leakage flux induces extra stray losses both in the stator and the rotor. During the transient, the effective rotor resistance seen by the fundamental current increases and the effective inductance decreases, and this is also consistent with the theory, because the skin effect is present during sinusoidal transients.

3.2.5 Lossy Model Parameter Computation

In order to use the equivalent circuits of Fig. 3.7, it was necessary to evaluate the parameters for the particular machine. The fundamental frequency parameters can be easily obtained from standard tests, but harmonic frequency parameters are much more difficult to obtain experimentally [28]. On the other hand, accurate theoretical prediction
3.7 Lossy \( D^e - Q^e \) equivalent circuits in synchronously rotating reference frame.

a) \( Q^e \)-axis circuit  

b) \( D^e \)-axis circuit
of harmonic frequency parameters also constitutes a formidable task. In this work, the standard fundamental frequency parameters were taken from literature [39], and the remaining parameters were estimated from practical considerations [31], as discussed below.

The fundamental stray load loss $P_{sll}$ at rated conditions amount to a few percent (typically 1% to 3%) of the rated machine power $P_0$ [35]. Let $\varepsilon_{sll}$ be the selected (or measured) percentage for a particular machine with a rated current $I_r$. The stray loss resistance $R_{sll}$, at base frequency $f_b$, is given by

$$R_{sll} = \frac{\varepsilon_{sll} P_0}{3 I_r^2} \tag{3.37}$$

At any other fundamental frequency $f$, the fundamental stray loss resistance of (3.15) can be rewritten as a function of $R_{sll}$

$$R_{sll} = R_{sll} \left( \frac{f}{f_b} \right) \left( \frac{1 + \gamma_1 f}{1 + \gamma_1 f_b} \right) \tag{3.38}$$

where $\gamma_1 = K_c / k_h$.

Although the core loss resistance $R_m$ at base 60 Hz frequency can be easily computed from standard machine tests, the estimation of eddy current coefficient $k_e$ and hysteresis loss coefficient $k_h$ requires the breakdown of total core loss $P_c$ into its eddy current $P_e$ and hysteresis $P_h$ components. Usually, the eddy current loss $P_e$ amounts to 20% - 50% of total core loss. Let $\varepsilon_{ec}$ be the assumed $P_e/P_c$ ratio. The corresponding relation between eddy current and hysteresis losses becomes
Substituting (3.3) and (3.4) into (3.39) leads to

\[ k_e = \frac{k_h e_{ec}}{f (1 - e_{ec})} \]  \hspace{1cm} (3.40)

At no load, the per unit slip \( s \) is practically zero, and (3.10) can be rewritten as

\[ \frac{k_h}{f} + k_e = \frac{(2\pi)^2}{R_m} \]  \hspace{1cm} (3.41)

where \((2\pi)^2\) has been substituted for \( k_c \). Finally, solving (3.40) and (3.41) results in

\[ k_e = \frac{(2\pi)^2 e_{ec}}{R_m} \]  \hspace{1cm} (3.42)

\[ k_h = \frac{(2\pi)^2 (1 - e_{ec}) f}{R_m} \]  \hspace{1cm} (3.43)

The harmonic frequency stray loss resistances will now be discussed. Kawagishi et al. [31] have indicated that the efficiency of an induction machine fed by a 5 KHz sinusoidal PWM inverter is nearly 99\% of that attainable under a 60 Hz AC supply. Their findings were also confirmed by Bausch et al. [59] for a much larger machine, operating from a hysteresis band PWM inverter. It can be assumed, therefore, that the harmonic stray losses \( P_{sh} \) of an induction motor fed by a 5 KHz PWM inverter is a percentage \( e_{sh} \) (\( \approx 1\% \)) of the rated output power \( P_o \). From Fig. 3.4(a) it is clear that, essentially all harmonic voltage will appear across the leakage reactances \( X_{lsn} \) and \( X_{lmn} \), since they are much larger than the resistive terms \( R_s \) and \( R_r \). Neglecting the small
secondary leakage reactance terms \(X_{ssn}\) and \(X_{rnn}\), the total harmonic stray loss resistance \(R_{sh}\) can be obtained from

\[
R_{sh} = \frac{3 V_h^2}{\epsilon_{sh} P_o} \tag{3.44}
\]

where \(V_h\) is total rms harmonic phase voltage. For a given PWM inverter, \(V_h\) can be obtained from simulation, or computed from the harmonic spectrum, if known. Distribution of \(R_{sh}\) into rotor and stator components is accomplished by assuming a partition proportional to the respective leakage inductances as

\[
R_{rsb} = R_{sh} \frac{L_{ts}}{L_{ts} + L_{ls}} \tag{3.45}
\]

\[
R_{rsb} = R_{sh} \frac{L_{ts}}{L_{ts} + L_{ls}} \tag{3.46}
\]

The actual harmonic stray loss resistances \(R_{ssn}\) and \(R_{rnn}\), at any other carrier frequency \(f_c\) are obtained from (3.13) as

\[
R_{ssn} = R_{ssb} \left( \frac{f_c}{f_{cb}} \right) \left( \frac{1 + \gamma_h f_{cb}}{1 + \gamma_h f_c} \right) \tag{3.47}
\]

\[
R_{rnn} = R_{rsb} \left( \frac{f_c}{f_{cb}} \right) \left( \frac{1 + \gamma_h f_{cb}}{1 + \gamma_h f_c} \right) \tag{3.48}
\]

where \(\gamma_h = k_{eh}/k_{hh}\) is assumed to be identical to \(\gamma_1\), and \(f_{cb}\) is the base carrier frequency (5 KHz, in this discussion).

The secondary leakage inductances \(L_{ssn}\) and \(L_{rnn}\), due to eddy current fluxes, are small compared to the conventional leakage inductances [25]. In the simulation study
discussed later, they were taken as 5% of the corresponding leakage inductances. Their presence in the equivalent circuit is mainly to ensure consistency of the dynamic behavior of the model. In fact, without them the equivalent circuit would become purely resistive at infinite frequency, whereas the actual induction motor becomes purely inductive.

3.3 Loss Modeling of Converter System

While for a diode rectifier only the conduction loss need to be considered, for a PWM inverter both switching loss and conduction loss are relevant.

3.3.1 Loss Modeling of Diode Rectifier

The three-phase diode rectifier with capacitor filter of Fig. 3.3 conducts discontinuously, and it can be shown that only two diodes (one in the upper group and one in the lower group) conduct at any instant. Fig. 3.8 shows the Thevenin equivalent circuit, where \( V_r \) is the ideal rectifier voltage profile, and the DC link inductance is \( L_d = 2 L_{sl} \), where \( L_{sl} \) is the per-phase source leakage inductance. The typical DC link voltages and current waves are shown in Fig. 3.9. The diodes conduct a current pulse \( i_d \) when \( V_r > (V_d + V_r) \), but conduction ceases sometime after \( V_r < (V_d + V_r) \), due to \( L_d \) effect. The two conducting diodes can be modeled by an off-set voltage \( V_r \) in series with a nonlinear resistance \( R \) as indicated. From the conduction characteristics of a particular diode, the voltage drop equation can be derived by using the software TABLE-CURVE [36]. The software takes a set of \((x,y)\) coordinate pairs as input and calculates the equation of the best fitting curve as given by
Fig. 3.8 Diode rectifier equivalent circuit.

Fig. 3.9 Rectifier voltage and current waves
\[ v_{dd} = v_{do} + K i_d^m \]  

(3.49)

where \( v_{do} \) is the offset voltage, and \( m \) the resistive drop exponent. The instantaneous conduction loss \( P_{ild} \) for the diode bridge can then be given as

\[ P_{ild} = 2 v_{dd} i_d \]  

(3.50)

Similarly, the instantaneous rectifier input power \( P_{iir} \) and output power \( P_{ior} \) can be expressed as

\[ P_{iir} = V_r i_d \]  

(3.51)

\[ P_{ior} = V_d i_d \]  

(3.52)

In the simulation program these expressions are integrated and averaged over a 60° interval to get the corresponding average values, required for efficiency computations.

### 3.3.2 Loss Modeling of PWM Inverter

For a PWM inverter, both conduction and switching losses should be considered. The conduction loss, in turn, is distributed between transistors and feedback diodes.

**Conduction Loss**

A typical inverter phase leg, with feedback diodes and polarized snubbers, is shown in Fig. 3.10(a). The conduction loss equivalent circuit is shown in Fig. 3.10(b), where each conduction drop has been modeled by a resistance in series with an offset voltage. Again, from the transistor saturation characteristics, the following linear voltage drop equation was derived with the help of TABLE-CURVE as shown in (3.53).
Fig. 3.10 (a) Transistor inverter phase leg.

(b) Conduction loss equivalent circuit.
\[ v_{ad} = v_{a0} + R_s i_c \] (3.53)

For the feedback diodes, the voltage drop eqn. (3.48) is valid for the particular devices.

It is important to observe that, for the positive half cycle of phase current \( i_a \), only transistor \( Q_1 \) or feedback diode \( D_4 \) conducts, whereas for the negative half cycle, \( Q_4 \) or \( D_1 \) conducts. Let \( G_A \) be the logic signal from the PWM control, such that, if \( G_A = 1 \) the upper transistor \( Q_1 \) conducts and conversely, if \( G_A = 0 \), \( Q_4 \) conducts. The leg voltage with respect to "ground", \( v_{ag} \), can be derived from the equivalent circuit as follows

\[
\begin{align*}
    v_{ag} &= V_d - v_{dda} \quad \text{if } i_a > 0 \text{ and } G_A = 1 \\
    &= -v_{dda} \quad \text{if } i_a > 0 \text{ and } G_A = 0 \\
    &= V_d + v_{dda} \quad \text{if } i_a < 0 \text{ and } G_A = 1 \\
    &= v_{dda} \quad \text{if } i_a < 0 \text{ and } G_A = 0
\end{align*}
\] (3.54)

Conversely, the instantaneous conduction loss in phase leg A can be expressed as

\[
\begin{align*}
    P_{cla} &= (V_d - v_{ag}) i_a \quad \text{if } G_A = 1 \\
    &= -v_{ag} i_a \quad \text{if } G_A = 0
\end{align*}
\] (3.55)

Similar expressions can be written for the other two phase legs, and the total instantaneous conduction loss \( P_{cli} \) obtained as the sum of the three components.

**Switching Losses**

Proper computation of switching losses requires careful analysis of inverter operation, as illustrated in Fig. 3.11 by the PWM waves for phase leg a. Assume, for example, that \( i_a \) is positive and \( Q_1 \) is initially ON, such that current flows through it.
Fig. 3.11 Pulse width modulation waves for phase leg A.
At instant 1, \(Q_1\) is turned OFF and \(Q_4\) is turned ON, the energy stored in \(C_{s1}\) is dissipated across the series resistor, and \(Q_1\) goes through a turn-off switching loss. However, as the current is now actually flowing through \(D_4\), no turn-on loss occurs in \(Q_4\). Similarly, at instant 2, when \(Q_1\) is again turned ON, \(C_{s1}\) discharges and at the same time the device has a turn-on loss, but no loss occurs in \(Q_4\), since it was not carrying current. Typical turn-on and turn-off switching waves are shown in Fig. 3.12. The energy loss during switchings can be related to the area of the product curve \(v_{CE} i_c\) (not shown). Summarizing, at every period of the carrier wave, the switching losses per phase leg are:

- Two discharges of capacitor snubbers;
- One turn-on loss;
- One turn-off loss.

**Shunt Snubber Loss**

After the qualitative discussion above, loss expressions will be derived. The energy loss associated with one snubber capacitor \(C_s\) discharge is

\[
W_1 = \frac{1}{2} C_s V_d^2
\]  
\[(3.56)\]

The total snubber energy loss of the inverter, in one fundamental period \(T\), can be represented as

\[
W_t = 3N_s W_1 = \frac{3}{2} N_s C_s V_d^2
\]  
\[(3.57)\]

where \(N_s\) is the number of switchings (on-and-off) of a phase leg in one fundamental period \(T\). Consequently, the average power loss due to snubber discharge is
Fig. 3.12 Typical turn-on and turn-off switching waves for transistor Q₁.
This loss can be represented by an equivalent shunt resistance $R_s$ across the DC bus as

$$R_s = \frac{2}{3fN_sC_s}$$

(3.59)

where $f = 1/T$ is the fundamental frequency. For a sinusoidal PWM, the quantity $N_s$ is a constant, whereas for a hysteresis band PWM it can be easily counted.

**Turn-off Switching Loss**

Turn-off and turn-on switching losses have been discussed and mathematically analyzed in detail by Me Murray [32], for a DC chopper. The same formulation was extended here for the case of inverter design.

The selection of snubber capacitor typically takes into account a number of factors, such as the maximum $dV/dt$ that the transistor can withstand, the allowable voltage overshoot during turn-off, and the turn-off switching losses. However, only the loss aspect was considered here. It can be shown [32] [37] that, for minimum total switching loss (snubber loss + turn-off loss), the value of snubber capacitor $C_s$ is usually small, such that, during turn-off the collector-emitter voltage rise time $t_{rv}$ is smaller than the collector current fall time $t_f$, as indicated in Fig. 3.12. Besides the assumption of a small $C_s$, another one is made in the following analysis [32]:

"During turn-off, the current fall is a linear time function completely determined
by the transistor characteristics, while the voltage rise is determined by the resulting action of the shunt snubber".

By applying this simplifying assumption, the transistor action during a turn-off can be expressed as

\[ i_C = I_L \left(1 - \frac{t}{t_f}\right) \quad ; \quad 0 \leq t \leq t_f \quad (3.60) \]

where \( I_L \) is the load current, \( i_C \) is the transistor collector current, and \( t_f \) the current fall time, as indicated in Fig. 3.12. Similarly, the transistor collector-emitter voltage \( v_{CE} \) is governed by the snubber action as

\[ v_{CE} = \frac{1}{C_s} \int_0^t (I_L - i_c) \, dt \quad (3.61) \]

Substituting (3.60) into (3.61) and solving the integral leads to

\[ v_{CE} = \frac{I_L t_f^2}{2 C_s t_f} \quad ; \quad 0 \leq t < t_r \quad (3.62) \]

The \( v_{CE} \) rise time \( t_r \) can be obtained from (3.62) by applying the end condition \( v_{CE} = V_d \) :

\[ t_r = \sqrt{\frac{2 C_s V_d t_f}{I_L}} \quad (3.63) \]

Neglecting the small voltage ringing, the transistor power dissipation can be expressed as:

\[ P_{dl} = \frac{I_L^2}{2 C_s} \left(1 - \frac{T}{t_f}\right) \frac{t_r^2}{t_f} \quad ; \quad 0 \leq t < t_r \quad (3.64) \]

\[ P_{d2} = V_d I_L \left(1 - \frac{t}{t_f}\right) \quad ; \quad t_r \leq t < t_f \quad (3.65) \]

Correspondingly, the transistor energy loss in one turn-off is
Equation 3.63 to 3.66 indicate that the transistor turn-off loss decreases as the snubber capacitance is increased, at the expenses of augmented snubber loss. For an inverter, the load current $I_L$ is nearly sinusoidal, and a precise computation of switching losses would require the solution of the preceding equations at every switching. By using the absolute value of the half-cycle average $I_{av}$ of the load current in place of $I_L$, a good estimate of switching loss can be produced, with significant reduction in computation time. As a result, the average transistor turn-off power loss for all three phase legs $P_{toff}$ can be expressed as

$$P_{toff} = 3K_{off}V_d(N_s/2)I_{av}f$$

where $N_s/2$ is the number of lossy turn-offs per converter leg in one cycle of fundamental frequency $f$, and the turn-off parameter $K_{off}$ is given by

$$K_{off} = \frac{t_f}{2}\left(1 - \frac{4}{3} \frac{t_{rv}}{t_f} + \frac{1}{2} \left(\frac{t_{rv}}{t_f}\right)^2\right)$$

From (3.67) it can be seen that the transistor turn-off loss can be represented by an equivalent DC link shunt resistance as

$$R_{toff} = \frac{V_d}{3K_{off}(N_s/2)I_{av}f}$$
Turn-on Switching Loss

For a transistor inverter, no snubber inductance $L$ as indicated in Fig. 3.10 is normally used. The stray inductance due to the wiring between the DC link capacitor and the transistor acts as a parasitic inductance for turn-on snubber. The operation of the parasitic series snubber is dual to that of the shunt snubber, and by using a procedure similar to that used for turn-off loss computation, the transistor turn-on power loss can be given as [32]

$$P_{ton} = 3K_{on}V_{d}(N_s/2)I_{av}f = \frac{V_d^2}{R_{ton}}$$

(3.70)

where the turn-on constant $K_{on}$ is defined as

$$K_{on} = \frac{t_{rv}}{2} \left( 1 - \frac{4}{3} \frac{t_r}{t_{rv}} + \frac{1}{2} \left( \frac{t_r}{t_{rv}} \right)^2 \right)$$

(3.71)

where $t_r$ is the collector current rise time and $t_{rv} = t_r + t_{tr}$ is the $v_{CE}$ voltage fall time. Again, $t_r$ is given by

$$t_r = \sqrt{\frac{2L_t_{rv}I_{av}}{V_d}}$$

(3.72)

From eqn. 3.70, the transistor turn-on loss can be represented by an equivalent DC link shunt resistance as

$$R_{ton} = \frac{V_d}{3K_{on}(N_s/2)I_{av}f}$$

(3.73)
Equations 3.70 to 3.72 indicate that the turn-on loss decreases as the inductance L is increased.

### 3.4 Model Validation

Both the converter and machine models, as discussed before, were simulated in detail, using PC-SIMNON, for a 10 HP drive with indirect vector control, with all the routines included in the Appendix. The diode rectifier routine is a direct translation of the mathematical model of Section 3.3.1. The inverter model uses hysteresis-band current control (HBPWM), and some implementation details will be now discussed. For phase leg A, a logic state variable A is defined such that, if A=1 the upper transistor (Q₁) conducts, whereas if A=0, the lower transistor is turned on. The new state of the variable (NA) is defined as

\[
NA = \text{If } I_{ac} > HB \text{ Then } 1 \text{ Else If } I_{ac} < -HB \text{ Then } 0 \text{ Else } A
\]

where \( I_{ac} \) is the current error (\( I_s^* - I_s \)) and HB is the amplitude of the hysteresis band. The phase A voltage with respect to "ground" (\( v_{ag} \)) is obtained from (3.54). The actual phase to neutral voltage (\( v_{an} \)), is given by [17]:

\[
v_{an} = \frac{2}{3} v_{ag} - \frac{1}{3} v_{bg} - \frac{1}{3} v_{cg}
\]

(3.74)

The number of switchings per cycle \( N_s \) is computed by counting the transitions in the logic state variable A. The carrier frequency \( f_c \) is directly obtained by the product of \( N_s \) and the fundamental frequency \( f \). These variables are then used in the computation of converter and machine frequency dependent parameters. Converter conduction and switching loss computation closely follows the mathematical formulation presented in
Section 3.3.

The equivalent circuits of Fig. 3.7 were used in the derivation of machine state equations. For convenience, the $Q_e$-axis circuit is reproduced in Fig. 3.13, indicating the selected state variables. For each axis, five state variables were selected, in this case the currents through the inductances $L_{ls}, L_{qsn}, L_{m}, L_{lr}$ and $L_{rsn}$. The current through $L_{lm}$ ($i_{qrm}$) does not constitute an independent state variable, since the following nodal equation must hold:

$$i_{qsl} = -i_{qsr} + i_{qrm} + i_{qml} - i_{qrl} - i_{qrr} \quad (3.75)$$

The above equation can also be written in terms of current derivatives:

$$\frac{di_{qsl}}{dt} = -\frac{di_{qsr}}{dt} + \frac{di_{qrm}}{dt} + \frac{di_{qml}}{dt} - \frac{di_{qrl}}{dt} - \frac{di_{qrr}}{dt} \quad (3.76)$$

In order to obtain a valid SIMNON model, the system must be described in a state equation format. To avoid unnecessary coupling among the state variables derivatives, the loop equations were carefully selected, as shown in Fig. 3.13, such that the inductance $L_{ls}$ is present in every loop. For example, the loop through $L_{qsn}$ yields:

$$\frac{di_{qsr}}{dt} = \frac{1}{L_{qsn}'} \left[ L_{s} \frac{di_{qsl}}{dt} - R_{snn} \cdot i_{qsr} \right] \quad (3.77)$$

Substitution of (3.77) along with the other loop equations into (3.76) produced the state equation for $i_{qsl}$. The remaining state equations were readily obtained by further substitution of the right side of $di_{qsl}/dt$ into the corresponding loop equations.

Both steady-state and transient conditions of the system were considered in the validation process of the models. Table 3.1 shows the complete power circuit parameters of the drive. The steady-state system performance was initially investigated, for various
Fig. 3.13 $Q^*$-axis circuit showing loops for state equations derivation.
Table 3.1 Power circuit parameters of the AC drive system.

**Machine:** 10 hp 230 / 460 V 27/13.5 A  
1755 rpm 60 Hz Class B 1.15 SF

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_s$</td>
<td>0.2264 Ω</td>
</tr>
<tr>
<td>$L_s$</td>
<td>0.00155 H</td>
</tr>
<tr>
<td>$R_{sh}$</td>
<td>0.0341 Ω</td>
</tr>
<tr>
<td>$L_{sh}$</td>
<td>0.0032 H</td>
</tr>
</tbody>
</table>

**D'-Q' equivalent circuit parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_r$</td>
<td>0.1256 Ω</td>
</tr>
<tr>
<td>$L_u$</td>
<td>0.00193 H</td>
</tr>
<tr>
<td>$R_{rsn}$</td>
<td>146.52 Ω</td>
</tr>
<tr>
<td>$L_{rsn}$</td>
<td>7.75 $10^{-5}$ H</td>
</tr>
</tbody>
</table>

**Diode:** POWEREX CD 411230 Dual diode module  
30 A / 1200 V

**Drop equation parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_{d0}$</td>
<td>0.8 V</td>
</tr>
<tr>
<td>K</td>
<td>0.052</td>
</tr>
<tr>
<td>m</td>
<td>0.585</td>
</tr>
</tbody>
</table>

**Transistor:** POWEREX KS 524503 Single Darlington transistor module, 30 A / 600 V

**Drop equation parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_{io}$</td>
<td>0.7 V</td>
</tr>
<tr>
<td>$R_t$</td>
<td>0.020</td>
</tr>
</tbody>
</table>
load torque and speed conditions, and the results are shown in Fig.3.14. From the machine loss profiles of Fig. 3.14(a) it can be seen that, for given speed, the total loss increases with torque, primarily due to increased fundamental copper and stray load losses. For a constant load torque, the losses increase with speed, mainly because of additional core loss and friction and windage losses. Fig. 3.14(b) shows the corresponding total converter loss for the same load torque and speed conditions. It can be seen that the converter loss is more affected by an increase in load torque at constant speed, rather than by increase of machine speed at constant load torque. This can be explained as follows:

With rated flux, the machine current is essentially a function of load torque and is practically independent of speed. Therefore, the inverter losses, that basically depend on machine current, is dominantly influenced by load torque. On the other hand, the diode rectifier loss is a function of DC link current, that increases with converter output power. Therefore, the rectifier loss is influenced by both speed and torque of the machine. Fig. 3.14(c) shows the total system efficiency at various torque and speed conditions. It can be noticed that the point of maximum efficiency moves toward the rated torque, as the speed increases, since the machine is designed to yield maximum efficiency at rated torque and speed. The upper speed used in the study is less than the base value of 1800 rpm, because the DC link voltage is not sufficiently high to enforce pure PWM operation near base speed, what is required for correct operation of the vector control system.

The transient performance will now be discussed. Fig. 3.15(b) and (c) show the flux responses to the magnetizing command current $i_{ds^*}$ step of Fig. 3.15(a). The response
Fig. 3.14 Steady state performance.
   a) Machine loss at various torques and speeds.
   b) Converter loss at various torques and speeds.
   c) System efficiency at various torque and speed.
Fig. 3.15 Rotor flux responses at constant speed.
   a) Command current step.
   b) Standard $D^s$-$Q^s$ model response.
   c) Lossy $D^s$-$Q^s$ model response.
of the standard Dₚ - Qₚ model of Fig. 3.2 is initially shown in Fig. 3.15(b), whereas that of the lossy model of Fig. 3.7 is presented in Fig. 3.15(c). It can be seen that the response of the proposed model has a smaller rise time than that of the standard model. This is caused by the following factors:

- There is an increase in rotor resistance due to temperature effect, included in the proposed model, but neglected in the standard model.

- The shunt branches that represent core loss and rotor stray loss have a transient effect of increasing the effective rotor resistance while decreasing the equivalent rotor leakage inductance. Both factors combine to reduce the effective rotor time constant \( \tau_r = L_r/R_r \). While the amount of reduction in \( \tau_r \) is dependent on the proper estimation of machine parameters, it is consistent from a theoretical point of view.

Another aspect that can be observed is that, the steady-state value of rotor flux is somewhat smaller for the proposed model. This reflects the saturation effect on the magnetizing inductance, included in the proposed model, but ignored in the standard model.

Finally, the transient torque response of an indirect vector-controlled drive was investigated at constant speed, as indicated in Fig. 3.16, to study possible effect for Dₚ-Qₚ equivalent circuits modification. Fig. 3.16(a) shows the step in the torque component of current \( (i_{qs}) \) at the rated flux condition. Fig. 3.16(b) shows the corresponding torque response for lossless converter and ideal Dₚ-Qₚ machine model with slip gain parameter tuned with the nominal machine parameters. The observed rise time is essentially due to
Fig. 3.16 Torque responses of the drive at rated flux (speed = 900 RPM).
   a) Command current step.
   b) Ideal and lossless converter-machine model (slip gain parameters are nominal machine parameters).
   c) Lossy converter-machine model (slip gain parameters are nominal machine parameters).
   d) Lossy converter-machine model (slip gain parameters track with machine parameters).
intrinsic delay of hysteresis-band current controller. Fig. 3.16(c) shows the response for lossy converter machine model, with nominal machine parameters used in the slip gain, whereas in Fig. 3.16(d) the same model was used, except for the slip gain, that is tuned for the actual machine parameters at the particular operating condition. The responses for all the three conditions are practically identical, what reflects a small mismatch between actual and nominal machine parameters used in the slip gain computation, for this particular operating condition.

In conclusion, the loss modeling of converter induction machine system proposed here takes into consideration the relevant losses occurring in a practical drive system. The simulation study has demonstrated the coherent system behavior at both steady-state and dynamic conditions. The actual accuracy of the models is dependent on the precision with which the machine and converter parameters can be obtained. However, the topic of parameter measurement or estimation is beyond the scope of this work.
CHAPTER 4

FUZZY EFFICIENCY OPTIMIZATION CONTROL

4.1 Introduction

Efficiency improvement in variable frequency drives is getting a lot of attention in the recent years. Higher efficiency is important not only from the viewpoints of energy saving and its obvious financial payoff, but also from the broad perspective of environmental pollution control. In fact, as the use of variable speed drives continues to increase in areas traditionally dominated by constant speed drives, motivated by the quest for increased productivity, the financial and environmental payoffs reach new importance.

The efficiency of a drive system is a complex function of the type of machine used, converter topology, type of power semiconductor switches and the selected PWM algorithm. In addition, the control system has profound effect on the drive efficiency. A drive system normally operating at rated flux gives the best transient response. However, at light loads, rated flux operation causes excessive core loss, compared to copper loss, thus impairing the efficiency of the drive. Since drives operate at light load most of the time, optimum efficiency can be obtained by flux control.

A number of methods for efficiency improvement through flux control have been proposed in the literature, since its principle was first introduced by Nola [38]. They can be classified into three basic types:
1) The simple pre-computed flux program as a function of torque, that is widely used for light load efficiency improvement. The scheme can be improved by generating the flux program at discrete speeds, to take the frequency dependency into consideration. This method, however, yields only a partial improvement in the system efficiency.

2) The real time computation of losses and corresponding selection of flux level that results in minimum losses is quite elegant. However, as the loss computation is based on a machine model, parameter variations caused by temperature and saturation effects tend to yield sub-optimal efficiency operation [60].

3) The on-line efficiency optimization control [39]-[41] on the basis of search, in which the flux is reduced in steps until the measured input power settles down to the lowest value. The control does not require the knowledge of machine parameters, is completely insensitive to parameter changes, and the algorithm is applicable universally to any arbitrary machine.

Induction motors are by far the most used among all electric motors, and are responsible for the consumption of a large fraction of the total electric energy produced. It is, therefore, natural that they have been the primary focus of the efficiency improvement studies. Part of this effort has been dedicated to vector controlled induction motor drives, since they possess inherent decoupling of flux and torque control loops, making them well suited for efficiency optimization control. In fact, Kirschen [40] investigated such system, using on line search method. Fig. 4.1 shows a typical power minimization process for a partial load condition. As the command flux is decreased, the
Fig. 4.1 Simulation of the input power minimization process [40].
input power initially decreases and then settles to the minimum value. There is, however, an undesirable side effect: as the flux is reduced, the developed torque is also affected, as demonstrated by the torque pulses of Fig. 4.1(c). If the load inertia is small, the low frequency torque pulsation will translate in speed fluctuation, that can eventually lead to mechanical resonance.

In this chapter, a fuzzy logic based on-line efficiency optimization control for an indirect vector controlled drive system is discussed. Fast convergence has been achieved by using adaptive step size of the excitation current. The low frequency pulsating torque generated by the efficiency controller has been suppressed by a feed-forward compensation algorithm. Both simulation and experimental studies were performed to validate the theoretical development. A paper on this subject [62] has been accept for presentation at the 1993 international conference on industrial electronics, control, and instrumentation, IECON' 93.

4.2 Fuzzy Efficiency Optimization of a Vector Control Drive

An indirect vector controlled induction motor drive incorporating the proposed efficiency optimization controller is shown in Fig. 4.2. It consists of a diode rectifier and a hysteresis band PWM transistor inverter. All the control functions indicated by the dashed outline are implemented in real time by a single digital signal processor (TMS320C25 from Texas Instrument, Inc.). The feedback speed control loop generates the active or torque current command \( i_{qs}^* \), as indicated. The vector rotator receives the torque and excitation current commands \( i_{qs}^* \) and \( i_{ds}^* \), respectively, from the two positions.
Fig. 4.2 Indirect vector controlled induction motor drive incorporating the efficiency optimization controller.
of a switch: the transient position (1), where the excitation current is established to the rated value \( i_{dr} \) and the speed loop feeds the torque current; and the steady-state position (2), where the excitation and torque currents are generated by the fuzzy efficiency controller and feedforward torque compensator which will be explained later. The fuzzy controller becomes effective at steady-state condition, i.e., when the speed loop error \( \Delta \omega \), approaches zero. Note that the DC link power \( P_d \), instead of input power, has been considered for the fuzzy controller since both follow symmetrical profiles. This can be demonstrated as follows:

- Minimization of \( P_d \) translates into minimization of DC link current \( i_d \), since the DC link voltage \( v_d \) is essentially constant;
- Rectifier loss are proportional to \( i_d \), (Eqn. 3.50);
- Minimum \( i_d \) yields minimum rectifier loss, and consequently, minimum input power.

The principle of efficiency optimization control with rotor flux programming at a steady-state torque and speed condition is explained in Fig. 4.3. The rotor flux is decreased by reducing the magnetizing current, which ultimately results in a corresponding increase in the torque current, normally by action of the speed controller, such that the developed steady-state torque remains constant. As the flux is reduced, the iron loss decreases with the attendant increase of copper loss. However, the total system loss (converter and machine) decreases, resulting in a reduction of DC link power. The search is continued until the system settles down at the minimum input power point A, as indicated. Any excursion beyond the point A will force the controller to return to the
Fig. 4.3 Principle of efficiency optimization control with rotor flux programming.
4.2.1 The Efficiency Optimization Controller

The fuzzy efficiency controller operation is explained in Fig. 4.4. The DC link power is sampled and compared with the previous value to determine the increment $\Delta P_d$. In addition, the last excitation current decrement $L \Delta i_d$ is reviewed. On these basis, the decrement step of $\Delta i_d^*$ is generated from fuzzy rules through fuzzy inference and defuzzification [14], as indicated. The adjustable gains $P_b$ and $I_b$, generated by the scaling factors computation block, convert the input variable and control variable, respectively, to per unit values, such that a single fuzzy rule base can be used for any torque and speed condition. The input gain $P_b$ as a function of machine speed $\omega_r$ can be given as

$$P_b = a \omega_r + b \quad (4.1)$$

The output gain $I_b$ is computed from the machine speed and an approximate estimate of machine torque $\hat{T}_e$ as

$$I_b = c_1 \omega_r - c_2 \hat{T}_e + c_3 \quad (4.2)$$

where

$$\hat{T}_e = K_f i_{ds} i_{qs}^* \quad (4.3)$$

The appropriate coefficients $a$, $b$, $c_1$, $c_2$, and $c_3$ were derived from simulation studies as follows:

1. Using simplified models, the approximated optimum magnetizing current $i_{ds0}$ was determined for selected operating points in the torque-speed plane;
Fig. 4.4 Efficiency optimization control block diagram.
. Least-square estimation was used to obtain a linear representation of \( i_{ds0} \) as a function of machine torque \( T_e \) and speed \( \omega_r \).

. The output gain expression (4.2) was defined as one third of the distance between rated magnetizing current \( i_{dsr} \) and approximate optimum \( i_{ds0} \).

. A new set of simulations was performed, for a load torque of 0.1 pu and different speeds. The magnetizing current was reduced by a step size \( I_b \) given by (4.2), and the corresponding reduction in input power \( \Delta P \) was obtained.

. The input power expression (4.1) was defined as the linear representation of \( \Delta P^{-1} \) as a function of speed, derived by least square estimation.

A few words on the importance of the input and output gains are appropriate here. In the absence of input and output gains, the efficiency optimization controller would react equally to a specific value of \( \Delta P_d \) resulting from a past action \( \Delta i_{ds^*}(k-1) \), irrespective of operating speed. Since the optimal efficiency point A (Fig. 4.3) is speed dependant, the control action could easily be either too conservative, resulting in slow convergence, or excessive, yielding an overshoot in the search process, with possible adverse impact on system stability. As both input and output gains are functions of speed, this problem does not arise. Eqn. 4.2 also incorporates the "a priori" knowledge that the optimum value of \( i_{ds^*} \) is a function of torque as well as machine speed. In this way, for different speed and torque conditions, the same \( \Delta i_{ds^*}(pu) \) will result in different \( \Delta i_{ds^*} \), ensuring a fast convergence. One additional advantage of per unit basis operation is that the same fuzzy controller can be applied to any arbitrary machine, by simply changing the coefficients of input and output gains.
The membership functions for the fuzzy efficiency controller are shown in Fig. 4.5. Due to the use of input and output gains, the universe of discourse for all variables are normalized in the [-1,1] interval. It was verified that, while the control variable $\Delta i_{ds}^*$ required 7 fuzzy sets to provide good control sensitivity, the past control action $L\Delta i_{ds}^*$, i.e., $\Delta i_{ds}^*(k-1)$, needed only 2 fuzzy sets, since the main information conveyed is the sign. The small overlap of the positive (P) and negative (N) membership functions is required to ensure proper operation of the height defuzzification method, i.e., to prevent indeterminate result in case $L\Delta i_{ds}^*$ approaches zero.

The rule base for fuzzy control is given in Table 4.1. An example fuzzy rule can be given as:

**IF the power increment ($\Delta P_d$) is negative medium (NM) and the last $\Delta i_{ds}^*$ ($L\Delta i_{ds}$) is negative (N), THEN the new excitation increment ($\Delta i_{ds}^*$) is negative medium (NM).**

The basic idea is that, if the last control action indicated a decrease of DC link power, proceed searching in the same direction, and the control magnitude should be somewhat proportional to the measured DC link power change. In case the last control action resulted in an increase of $P_d$ ($\Delta P_d > 0$), the search direction is reversed, and the $\Delta i_{ds}^*$ step size is reduced to attenuate oscillations in the search process.

As the optimum point varies with speed and load conditions, and is also affected by changes in machine parameters, the fuzzy logic search controller constitutes a natural choice for this problem. Initially, a linguistic description of the control strategy could be easily obtained. Fuzzy logic then provided the appropriate mathematical framework for the derivation of the actual controller. The fine tuning process was, nevertheless,
Fig. 4.5 Membership functions for the fuzzy efficiency control.
(a) Change in DC link power ($\Delta P_d$ (pu)).
(b) Last change in magnetizing current ($L\Delta i_{ds}$ (pu)).
(c) Magnetizing current increment ($\Delta i_{ds}$ (pu)).
Table 4.1 Rule base for the fuzzy efficiency controller.

<table>
<thead>
<tr>
<th>$L\Delta i_{ds}(pu)$</th>
<th>$\Delta P_d(pu)$</th>
<th>N</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>PB</td>
<td>PB</td>
<td>PM</td>
<td>NM</td>
</tr>
<tr>
<td>PM</td>
<td>PM</td>
<td>PS</td>
<td>NS</td>
</tr>
<tr>
<td>PS</td>
<td>PS</td>
<td>PS</td>
<td>NS</td>
</tr>
<tr>
<td>ZE</td>
<td>ZE</td>
<td>ZE</td>
<td></td>
</tr>
<tr>
<td>NS</td>
<td>NS</td>
<td>NS</td>
<td>PS</td>
</tr>
<tr>
<td>NM</td>
<td>NM</td>
<td>NM</td>
<td>PM</td>
</tr>
<tr>
<td>NB</td>
<td>NB</td>
<td></td>
<td>PB</td>
</tr>
</tbody>
</table>
somewhat time consuming.

4.2.2 Feed-forward Pulsating Torque Compensation

As the excitation current is reduced in adaptive steps by the fuzzy controller, the rotor flux $\psi_r$ decreases exponentially [17], which is given by

$$\frac{d}{dt} \psi_{dr} = \frac{L_m i_{ds} - \psi_r}{\tau_r} \tag{4.4}$$

where $\tau_r (= L_r/R_r)$ is the rotor time constant and $L_m$ the magnetizing inductance. The decrease of flux causes loss of torque, which normally is compensated slowly by the speed control loop. Such pulsating torque at low frequency is very undesirable because it causes speed ripple and may create mechanical resonance. To prevent these problems, a feed-forward pulsating torque compensator is proposed here.

Under correct field orientation control, the developed torque is given by

$$T_e = K_i i_{qs} \psi_r \tag{4.5}$$

For an invariant torque, the torque current $i_{qs}$ should be controlled to vary inversely with the rotor flux $\psi_r$. A practical implementation of this concept is illustrated in Fig. 4.6. A compensating signal $\Delta i_{qs}(t)$ is added to the original $i_{qs}$ to counteract the decrease in flux $\Delta \psi_r(t)$, where $t \in [0,T]$ and $T$ is the sampling period for efficiency optimization control. Let $i_{qs}(0)$ and $\psi_{dr}(0)$ be the initial values for $i_{qs}$ and $\psi_{dr}$, respectively, for the k-th step change of $i_{ds}$ (k=2 in Fig. 4.6.) For a perfect compensation, the developed torque must
Fig. 4.6 Principles of feedforward torque compensation.
remain constant, and the following equality holds

\[
[\psi_r(0) + \Delta \psi_r(t)][i_{qs}(0) + \Delta i_{qs}(t)] = \psi_r(0) i_{qs}(0)
\] (4.6)

Solving for \(\Delta i_{qs}(t)\) yields

\[
\Delta i_{qs}(t) = \frac{-\Delta \psi_r(t) i_{qs}(0)}{\psi_r(0) + \Delta \psi_r(t)}
\] (4.7)

where \(\Delta \psi_r(t)\) is governed by eqn. 4.4, with \(\Delta i_{ds}\) substituted for \(i_{ds}\). To implement such compensation, eqn. 4.7 is adapted to produce \(\Delta i_{qs}^*(t)\), using flux estimate \(\hat{\psi}_r\), and command \(i_{qs}^*\) in place of actual signals. A good approximate solution for \(\Delta i_{qs}^*(t)\) can be obtained by replacing the denominator of (4.7) by its estimated steady-state value \(\hat{\psi}_r(T) = L_m i_{ds}^*(T)\). In this case, the compensation can be implemented in two steps as shown in Fig. 4.7. First, the value for the compensating current step is computed by discrete eqn. 4.8.

\[
\Delta i_{qs}^*(k) = \frac{\psi_r(k-1) - \psi_r(k)}{\psi_r(k)} i_{qs}^*(k-1)
\] (4.8)

Next, the current step is processed through a first order low pass filter of rotor time constant, and then added to the previous compensating steps, as also illustrated in Fig. 4.6. This current is added to the original speed loop generated current \(i_{qs}^*\), so that, at any instant the product \(i_{qs} \psi_r\) remains essentially constant. It may be noted, however, that the actual rotor time constant of the machine may vary, giving somewhat imperfect pulsating torque compensation.

To ensure adequate compensation, magnetizing inductance saturation effects were taken into account using a piece-wise linear model, similar to that developed for the induction machine model. The magnetizing inductance \(L_m\) is estimated on line from the
Fig. 4.7 Feedforward pulsating torque compensator block diagram.
where $L_{m0}$ is the non-saturated value of $L_m$, $i_{ds0}$ is the linear region breakpoint, and $m_d$ is a saturation coefficient. All parameters are obtainable from a standard no-load test.

### 4.2.3 Transition to Optimum Transient Response Mode

It may be noted from the preceding discussion that efficiency optimization control is only effective at steady-state condition. A disadvantage of this control mode is that the transient response becomes sluggish, because the weaker flux reduces the maximum torque capability, as indicated by eqn. 4.5. For any change in load torque or speed command, fast transient response capability of the drive can be restored by establishing the rated flux. Fig. 4.8 shows the criteria for transition between efficiency optimization (Mode 2) and transient response optimization (Mode 1). The system starts in Mode 1 and then switches to Mode 2 when the steady-state criteria is met. In the event of a load disturbance or a change in set speed, the system switches back to Mode 1 by establishing the rated magnetizing current.

At steady-state, current must be kept within the thermal limits of the machine and inverter, that is usually the rated machine current. During a transition from Mode 2 to Mode 1, this limit can be temporarily increased, (typically by 50% for 60 secs.) such that more torque is available to meet the acceleration / deceleration demand. This constraint is given by the relation

$$L_m = L_{m0}, \quad \text{if } i_{ds}^* \leq i_{ds0}$$

$$L_m = L_{m0} - m_d (i_{ds}^* - i_{ds0}), \quad \text{if } i_{ds}^* > i_{ds0}$$

(4.9)

command $i_{ds}^*$ as
Fig. 4.8 Transition between efficiency optimization and transient response optimization modes.
\[ \sqrt{i_{ds}^*}^2 + i_{qs}^*^2 < I_{\text{lim}} \]

where \( I_{\text{lim}} = 1.5 I_{sr} \), and \( I_{sr} \) is the rated stator current.

Notice that, as the rotor flux does not change instantly, the maximum torque capability following a transition might not be enough to overcome the torque demand, even with the relaxed current limit. In such scenario, there is a possibility that the drive will stall, and preventive measures should be taken. One simple solution consists in imposing a minimum value for \( i_{ds}^* \), during Mode 2, that would keep the maximum torque capability at a safe level, at the expenses of suboptimum efficiency operation.

### 4.3 Simulation Program Development

In chapter 3, SIMNON routines for converter and induction motor lossy models were developed and integrated to a speed control system, using indirect vector control. In this section the Fuzzy Efficiency Optimization Routine (FEOPT) and Feed Forward Torque Compensation Routine (FFTC) development will be discussed. The simulation block diagram of Fig. 4.9 shows the various routines used in the study, and the associated input/output variables for every subsystem. Discrete subsystems are identified by a D whereas continuous subsystems are marked with a C.

The flowchart of Fig. 4.10 shows the structure of the FEOPT routine. For convenience, the determination of the operating mode (Mode 1 or Mode 2 of Fig. 4.8) is included in the FEOPT routine. First, the program checks the previous operating mode and verifies if any transition is required. If after the tests the system is found to be in the
Fig. 4.9 SIMNON simulation block diagram showing I/O variables.
Fig. 4.10 Fuzzy efficiency optimization flowchart.
efficiency optimization (EO) mode, then the program proceeds to the tasks actually related to the efficiency optimization process, as indicated in the flowchart. In order to initiate the efficiency optimization process, following a Mode 1 to Mode 2 transition the first $\Delta i_{ds}$ is set to -1 pu. The actual value in Amperes is determined by the output scaling factor $I_b$. From the second iteration on, $\Delta i_{ds}$ value is truly derived from the fuzzy controller. After the optimum point is reached, a minimum (non-zero) value is imposed to $\Delta i_{ds}$, to keep the search process active, and ensure true optimum operation in case of parameter variation or small change in load condition.

The algorithm for fuzzy control implementation has already being described in detail in Section 2.2.1, for the case of fuzzy $\Delta \alpha$ compensation. However, its implementation using SIMNON language is somewhat complicated, mainly because SIMNON does not support branch instructions. Furthermore, the order of program execution is not completely controlled by the user, since during compilation the program sorts the instructions to optimize for execution time.

The methodology used in the actual derivation of fuzzy efficiency optimization SIMNON routine is explained here with the help of Fig. 4.11.

1. Compute input and output scaling factors, $P_b$ and $I_b$, respectively, and the change in input power, $\Delta P_d$ (pu);

2. Make a preliminary computation of degree of membership for all fuzzy sets. For instance, the degree of membership of $\Delta P_d$ (k) in the fuzzy set PM, shown in Fig. 4.11 is given by:
Fig. 4.11 Evaluation of degree of membership.
3. Define the interval index \( J \) that identifies which fuzzy sets possess non-zero degree of membership. SIMNON supports the conditional assignment of value to a variable, but does not allow a continuation line. This problem can be solved by defining some auxiliary variables \((J_1, J_2, \ldots)\) as indicated below:

\[
J = \begin{cases} 
\text{If } \Delta P_d < -P_1 \text{ Then 1} \\
\text{Else If } \Delta P_d < -P_2 \text{ Then 2} \\
\text{Else } J_1
\end{cases}
\]

\[
J_1 = \begin{cases} 
\text{If } \Delta P_d < -P_3 \text{ Then 3} \\
\text{Else If } \Delta P_d < 0 \text{ Then 4} \\
\text{Else } J_2
\end{cases}
\]

...

4. Evaluate the degree of membership for the relevant fuzzy sets:

\[
\mu_{p_1} = \begin{cases} 
\text{If } J < 3 \text{ Then } \mu_{NB} \\
\text{Else If } J < 4 \text{ Then } \mu_{NM} \\
\text{Else... Else } \mu_{PB}
\end{cases}
\]

\[
\mu_{p_2} = 1 - \mu_{p_1}
\]

For the situation illustrated in Fig. 4.11, \( \mu_{p_1} = \mu_{PS} \) and \( \mu_{p_2} = \mu_{PM} \).

5. Apply similar procedure to compute degrees of membership \( \mu_{l_1} \) and \( \mu_{l_2} \) for the second input variable, \( L\Delta i_{ds}^{*}(pu) \).

6. Evaluate the antecedent of the relevant rules, using MIN operator. Typically, four rules are fired. The first rule is illustrated here:

\[
\mu_{RA} = \text{MIN} (\mu_{p_1}, \mu_{l_1})
\]

7. Retrieve the contribution of each fired rule, from the rule base:

\[
\Delta I_a = \begin{cases} 
\text{If } J < 3 \text{ Then } I_1 \\
\text{Else If } J < 4 \text{ Then } I_2 \\
\text{Else...}
\end{cases}
\]

Due to the symmetry of this particular rule base, \( \Delta I_b = -\Delta I_a \).

8. Calculate the new \( \Delta i_{ds}^{*} \) by height defuzzification method.
The feed forward torque compensation (FFTC) routine operation is explained by the flowchart of Fig. 4.12. Its principle has already being discussed in section 4.2.2, and therefore, only some relevant aspects will be discussed here. During the search process (EO mode), the test for maximum stator current is based on the steady-state limit $I_{lim1}$, that corresponds to the rated motor current. If it is detected that the new control action $\Delta i_{ds}^*$ would result in a violation of such limit, the new magnetizing current $i_{ds}^*(k)$ is reset to its previous value $i_{ds}^*(k-1)$, and similarly, no correction is made in $i_{qs}^*(k)$. If the program detects that the system has switched to a TO mode (Mode 1), rated $i_{ds}^*$ is reestablished and the accumulated compensating signal $\Sigma \Delta i_{qs}^*(k)$ is reset to zero. The actual compensating signal $\Sigma \Delta i_{qs}$ will approach zero following the first order filter dynamics. This action ensures that no sudden change in machine torque will take place, following transition in operating mode. Furthermore, as the original torque current command $i_{qs}^*$ from the speed controller immediately affects $i_{qs}^*$, proper control action to any disturbance is ensured. For the TO mode, the stator current limit is set to 50% higher than its steady-state value, as mentioned before. Besides the functions related to torque compensation, the routine also computes rotor flux estimate, performs slip frequency calculation and is also responsible for approximate machine torque estimation, required by the FEOPT routine.
Fig. 4.12 Feedforward torque compensation simulation flowchart.
4.4 Simulations Study

After the development and tests of the SIMNON routines, a detailed performance evaluation study was conducted, using the machine parameters given in Table 4.2. Fig. 4.13 shows the time domain optimum efficiency search curves at speed $\omega_r = 0.25 \text{ pu}$ and load torque $T_L = 0.1 \text{ pu}$. The optimum flux level is achieved in only four steps of $i_{ds}^*$. The search algorithm, however, imposes a minimum non-zero step size of $\Delta i_{ds}^*$ when $\Delta P_d$ approaches zero, in order to ensure true optimum operation for parameter variation or small change in load condition. The effectiveness of the pulsating torque compensation is evident from the torque profile of Fig. 4.13(d) and speed response of Fig. 4.13(e). Fig. 4.14 gives efficiency curves at different operating points on torque-speed plane, with and without efficiency optimization control. As the class B induction machine was designed for 230 V, 60 Hz operation, the highest speed was limited to 0.75 pu (1350 rpm), because the DC link voltage is not high enough to enforce current regulation at higher speeds, under rated flux conditions. As expected, maximum efficiency gains occur at light load, where the rated flux core losses are very dominant. Fig. 4.15 illustrates system performance during operating mode transitions. The system is initially at steady-state and the E.O. mode is activated. At $t=3$ sec. the system has reached the optimum efficiency point, when the load torque is suddenly increased from 0.1 pu to 0.5 pu, causing a speed droop (Fig. 4.15(d)), that in turn forces a transition to TO mode (Mode 1). The rated $i_{ds}^*$ is readily reestablished, and the speed controller increased $i_{qs}^*$, bringing the speed back to its command value in less than 0.3 sec. Once the system reaches a steady-state condition, it switches back to EO mode (Mode 2) and a new search is initiated. Due to
Table 4.2 Induction machine parameters for efficiency optimization studies.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power (hp)</td>
<td>5</td>
</tr>
<tr>
<td>Voltage (V)</td>
<td>230/460</td>
</tr>
<tr>
<td>Frequency (Hz)</td>
<td>1710</td>
</tr>
<tr>
<td>Current (A)</td>
<td>13.4/6.7</td>
</tr>
<tr>
<td>Poles</td>
<td>4</td>
</tr>
<tr>
<td>Class</td>
<td>B</td>
</tr>
</tbody>
</table>

**D^P-O^e equivalent circuit parameters:**

- $R_s = 0.406 \, \Omega$
- $L_{ls} = 0.00213 \, H$
- $L_m = 0.0494 \, H$
- $R_T = 0.478 \, \Omega$
- $L_T = 0.00213 \, H$
Fig. 4.13  Time domain simulated optimum efficiency search curves at $\omega_r=0.25$ pu and $T_L=0.1$ pu. (a) Magnetizing current ($i_{ds}^*$). (b) DC link power ($P_d$). (c) Torque current ($i_{qs}^*$). (d) Machine torque ($T_e$). (e) Speed ($\omega_r$).
Fig. 4.14 Simulated efficiency curves.
Fig. 4.15 Drive performance in time domain with sudden increase of load torque (transition to transient response optimization mode).
higher load torque the output gain $I_b$ (in FEOPT) is reduced, what results in a smaller first step for the new search, as indicated in Fig. 4.15(a).

4.5 Hardware Circuit Design

The vector control system used in the experimental study was derived from an AC200 servo drive system, from General Electric Company. The modular servo system originally had the following components:

- Servo System Rack.
  
The rack is constructed with two receptacles to receive the plug in servo module and the power supply module. The rack has integrated cooling fan and DC link capacitor bank.

- Power Supply.
  
This module contains a power diode rectifier, rated at 230 V AC input, 325 V DC output. It also contains a low voltage, 80 KHz, auxiliary power supply for the control circuitry of the servo module. The control circuitry for dynamic breaking is also included, that automatically connects an external resistor across the DC bus, whenever the DC bus voltage reaches 385 Volts. In this way, the energy recovered from the deceleration of the drive is properly dumped, and dangerous DC voltage levels are prevented.

- Servo Module.
  
This module comes with a three-phase power transistor inverter and associated base drive and control circuit boards. The system originally used a sinusoidal
PWM current control, and had all the vector control functions implemented in dedicated hardware. As the system control board was missing, a new control board was designed.

The structure of the vector control system hardware is presented in Fig. 4.16. Its major component is a DSP-16 control board, from Ariel Corporation, based on Texas Instruments TMS320C25 DSP. The board is internally mounted on an IBM 286 compatible host computer, that serves as a development and debugging environment, as well as user interface during normal operation. The control board comes with two D/A channels, used to pass command currents \(i_a^*, i_b^*\) to the current controller, and two A/D channels, required for DC link voltage and current sensing. The speed interface is directly connected to the TMS320C25 data, address and control buses, and operates as a parallel peripheral of the DSP. The system is designed with basic protection and monitoring functions.

![Fig. 4.16 Block diagram for vector control system hardware.](image-url)
4.5.1 Current Controller

A hysteresis band current controller (HBPWM) was selected, because of its simplicity and robustness. Fig. 4.17 shows the circuit diagram of the proposed controller. Only two D/A channels are available; consequently, the reference current for phase c, $i_c^*$, is obtained by addition of $i_a^*$ and $i_b^*$. After some very small filtering and scaling, the actual machine current $i_a$ is subtracted from the command value $i_a^*$. The error is next compared to the hysteresis band to determine the state of phase leg A switches. Notice that, by using isolating diodes the hysteresis band for the three phases are controlled by a single trimpot. Identical circuits are used for phases b and c, and therefore, not shown here in detail.

4.5.2 Monitoring and Protection

Fig. 4.18 shows the circuit diagram for monitoring and protection. The system comes with built in inverter temperature sensor and under voltage protection, that are combined into a single trip signal (TRIP). Over current (OC) protection is obtained by extracting the positive and negative current profiles from the feedback current signals and comparing with pre-set limits. A small delay is introduced to prevent noise related false tripping. An emergence stop (E/S) push button is included for added safety. At power on, the delay circuits ensure a disable state for the drive circuit, and this must be cleared by the RESET push button. With all the protection circuitry enabled, a RUN command must be issued from the control software to effectively enable the base drive (STOP/RUN signal). LED lamps are associated with every trip/control signal for monitoring purpose.
Fig. 4.17 Hysteresis band current controller.
Fig. 4.18 Monitoring and protection circuit diagram.
4.5.3 Speed Measurement and Control Interface

Precise speed sensing is very important for proper operation of vector control systems. In fact, an error of 1% in speed measurement can result in severe degradation of drive performance.

The M/T method discussed in this section is characterized by very high precision in all speed range. Its implementation is shown in the circuit diagram of Fig. 4.19. Basically, it consists in counting pulses from an encoder (ENC) and from a high frequency clock (CLK). Speed information is obtained in software, by dividing the encoder counter increment ($\Delta M$) by the clock counter increment ($\Delta T$), with proper scaling, what will be explained later. Two 8-bit binary counters with output register (74ALS590) are used in cascade, for both ENC and CLK counters. The counter state is passed to the output register only at the positive edge of the register clock (RCK). By using the ENC pulse as the RCK signal of both counters, the contents of the output registers precisely reflects the actual speed.

A 3 to 8 decoder (74ALS138) is used to properly interface the counters to the DSP. Although the 74ALS590 output registers are directly connected to the DSP data bus, they are normally in a high impedance state, until the output enable pin (G) is asserted low by the 74ALS138 decoder. The output register content is then latched into the DSP.

The direction of rotation (DIR) and FAULT signals are similarly passed to the DSP through an octal bus transceiver (74ALS245). Finally, the START/STOP and counter clear CCLR signals are sent from the DSP to the control board through a dual 4-bit D flip-flop.
Fig. 4.19  Speed sensing and control interface.
4.5.4 DC Link Voltage and Current Sensing Interface

Adequate DC link voltage $V_d$ and current $I_d$ sensing is crucial for DC link power $P_d$ computation, and consequently, for the correct operation of the efficiency controller.

DC link voltage sensing is shown in Fig. 4.20(a). A potential divider is initially used to bring the DC bus voltage to an adequate level and is next fed to an isolation amplifier (AM227), that ensures complete isolation between power and signal circuits. Its output is directly connected to the A/D input.

DC current sensing is performed by a high precision hall effect sensor (LEM LF 50-P), as indicated in Fig. 4.20(b). The sensor is placed between rectifier output and the DC capacitor bank. This location ensures that the PWM current harmonics generated by the inverter operation are filtered out by the DC link capacitor, resulting in an $I_d$ waveform consisting of 6 pulses per period of 60 Hz supply. The LEM module works as a current transformer with a 1000:1 ratio, and with a 50Ω load resistor, a 50 mV/A gain is obtained. To ensure a better utilization of the A/D dynamic range, a low pass filter with a 3.81 gain is used. The reduction in DC link current bandwidth has no detrimental impact on $P_d$ computation, since only its average value is relevant. In fact, as $V_d$ is essentially constant, the average of $P_d$ closely follows $I_d$ average.

4.6 Real Time Software Design

The complete drive control system, including the fuzzy efficiency controller and pulsating torque compensator, was implemented in assembly language in a TMS320C25 DSP based control board from Ariel Corporation. Figure 4.21 shows the software structure
Fig. 4.20 DC link voltage and current sensing.
Fig. 4.21 Software structure of the drive system including the efficiency optimization controller
of the entire control system. Both the A/D and D/A converters operation are interrupt driven, and the maximum sampling frequency of 50 KHz per channel was selected, to benefit from the embedded 20 KHz bandwidth input and output filters.

The timer interrupt (TINT) was set to 200 μs, that is the sampling time of the vector control routines. The remaining routines are executed at lower sampling frequency, and controlled by software timers.

4.6.1 Basic Vector Control Functions

As previously indicated in the block diagram of Fig. 4.2, the implementation of the indirect vector control requires the generation of unit vectors \( \sin \theta_e \) and \( \cos \theta_e \) from the speed signal \( \omega_r \) and slip frequency \( \omega_{sl} \). The value of \( \theta_e \) is obtained by discrete integration of \( \omega_e \), using double precision to ensure accurate computation at low speeds. A sine table with a resolution of 5.689 bits per degree was initially constructed and stored in program memory. The \( \sin \theta_e \) and \( \cos \theta_e \) values are obtained by simply scaling down the \( \theta_e \) value to the interval \([-1024, 1024]\) bits and then retrieving the proper value from the sine table (table look-up method). The unit vectors are next used to calculate the stationary reference currents \( i_{q*} \) and \( i_{d*} \), from which the phase current references \( i_a^* \) and \( i_b^* \) are obtained. Phase C reference \( i_c^* \) is obtained from \( i_a^* \) and \( i_b^* \) by a summing operational amplifier (Fig. 4.17).
4.6.2 Speed Computation

Speed computation will be discussed in details here. The basics of the M/T method have already being explained in the discussion of the speed interface hardware. The same principles are used here, as indicated by the speed computation flowchart of Fig. 4.22. The usual sampling time of 1 ms can be extended up to 5 ms, in case of very low speed. If no change occurs in the content of the encoder pulse counter (M) within the 5 ms extended period, the algorithm assumes $\omega_r = 0$. Under normal operation $M_{new} > M_{old}$ and the speed computation proceeds as indicated. For operation in both directions, the direction signal (DIR) must be probed, and the speed polarity corrected accordingly. Robust operation of the algorithm is achieved by imposing a pre-defined limit ($\Delta\omega_{max}$) to the speed change ($\Delta\omega_r = \omega_{new} - \omega_{old}$). This protects the system against eventual malfunction of the hardware interface or encoder failure. Furthermore, by selecting $\Delta\omega_{max}$ slightly larger than the maximum possible speed change under normal operation, no delay is introduced in the system, as would occur if a low pass filter were used.

4.6.3 Efficiency Optimization Controller

The real time fuzzy efficiency controller uses the same algorithm described in the simulation study. Consequently, only some implementation aspects will be discussed here.

The algorithm basically derives the next control action ($\Delta i_{ds}(k)$) from the measured change in input power ($\Delta P_d(k)$) and the last control action ($L\Delta i_{ds}(k)$). Initially, the interval index for each input variable is determined, thus defining the membership
Fig. 4.22 Speed computation flowchart.
functions with non-zero degree of membership, as previously indicated in Fig. 4.11, for the case of $\Delta P_d$.

Because the DSP does not have a division instruction, whenever possible a division operation is replaced by a multiplication. This approach is used in the computation of the degree of membership. For the particular value of $\Delta P_d(k)$ shown in Fig. 4.11, we have

$$\mu_{PM}(\Delta P_d) = (\Delta P_d(k) - P_3) (P_2 - P_3)^{-1}$$  \hspace{2cm} (4.11)

As $P_2$ and $P_3$ are constants, $(P_2 - P_3)^{-1}$ is pre-computed and stored in data memory.

The rule base evaluation requires the retrieval of the control signal associated with every fired rule. This task is readily accomplished by the use of interval indices and a look up table containing the rule base. Let $J$ be the index for $\Delta P_d$, $I$ the index for $L\Delta i_{ds}$, $M$ and $N$ the number of fuzzy sets for $\Delta P_d$ and $L\Delta i_{ds}$, respectively. The relative position (POS) of the first fired rule within the look up table is given by

$$POS = (J - 1) M + I$$  \hspace{2cm} (4.12)

The second rule is located at $POS+1$, the third one at $POS+N$, and the last one at $POS+N+1$. Once the rule base evaluation is performed, the new control action is computed by the height defuzzification method.

4.7 Experimental Study

After careful software and hardware systems integration and debugging, the drive system was tested with a standard 5 hp NEMA class B induction machine, rated at 1710
rpm, 230 V, whose parameters were given on Table 4.2.

Typical AC supply phase voltage and current waveforms are shown in Fig. 4.23 for phase A, when the drive system was operating under rated flux, at a speed of 1350 rpm (0.75 pu) and a load torque of 7.68 lbf ft (0.5 pu). As discussed in Chapter 3, the diode rectifier conducts discontinuously, and therefore, significant amounts of harmonic currents are injected into the AC supply. The small unbalance in line voltages produces current pulses of varying amplitude, as shown. Some distortion in supply voltage waveform is also observable, generated by the harmonic current flow through the AC source impedance.

The machine input currents are shown in Fig. 4.24, when the drive was operating under rated flux, at 1350 rpm (0.75 pu) and delivering a load torque of 10.75 lbf ft (0.70 pu). The intrinsic operations of the hysteresis band PWM insures that the actual currents closely follow the sinusoidal command currents.

The operation of the fuzzy efficiency controller is illustrated in Fig. 4.25. The drive system was initially in a steady-state condition, at a speed $\omega_r = 450$ rpm (0.25 pu) and load torque $T_L = 1.54$ lbf ft (0.10 pu). At $t = 0$ the efficiency controller was enabled, and the magnetizing current command $i_{ds}^*$ was adaptively reduced on the basis of measured DC link power as shown in Fig. 4.25(a). The optimum operating point was achieved in only six steps. After that, the search algorithm remained active, with small $\Delta i_{ds}^*$ steps, to ensure proper tracking in case the optimum point shifts due to parameter variation or slow changes in load torque. The operation of the pulsating torque compensation scheme is illustrated in Fig. 4.25(b). The bottom trace shows the
Fig. 4.23 AC supply phase voltage and current waveforms at $\omega_r=0.75$ pu and $T_L=0.50$ pu. Scales: 50 V/div., 10 A/div., 2 msec./div.

Fig. 4.24 Machine input currents at $\omega_r=0.75$ pu and $T_L=0.70$ pu. Scales: 10 A/div., 2 msec./div.
Fig. 4.25 Experimental search curve at $\omega_r=0.25$ pu and $T_e=0.10$ pu.
(a) Top: $i_{ds}^*$ (3.33 A/div.); Bottom: $P_d$ (58.5 W/div.).
(b) Top: $i_{qs}^*$ (1.67 A/div.); Bottom: $\Sigma\Delta i_{qs}^*$ (1.67 A/div.).
(c) Top: $i_{ds}^*$ (3.33 A/div.); Bottom: $\omega_r - \omega_r^*$ (61 rpm/div.).
Time scale: 2 sec./div.
compensating signal, $\Sigma \Delta i_{qs}^*$, whereas the upper trace represents the final torque current command, $i_{qs}^*$. It can be seen that both signals follow similar profiles, demonstrating that the output of the speed controller ($i_{qs}^*$) remains essentially constant. The effectiveness of the torque compensation can also be observed in Fig. 4.25(c), since the drive speed (bottom trace) remains practically constant.

Another search curves are shown in Fig. 4.26, for the case of $\omega_r = 900$ rpm (0.5 pu) and $T_L = 4.61$ lbf ft (0.3 pu). As expected, the optimum $i_{ds}^*$ is higher than that of Fig. 4.25(a), due to increased load torque. Accordingly, the observed reduction in DC link power (lower trace) is smaller, confirming that less efficiency gain is attainable as the load torque increases.

The associated torque current command, pulsating torque compensating signal and speed signal have similar profiles to those shown in Fig. 4.25 (b) and (c), and are not included here.

Some small fluctuations in DC link power are observable at any load condition, even at rated flux. These are caused by several factors, such as actual fluctuations in the load (dynamometer); the small unbalance in practical motors that creates negative sequence torques; and also some eventual small speed oscillations due to closed loop system dynamics. To reduce these fluctuations and ensure proper operation of the efficiency controller, a low pass filter of time constant $\tau = 0.5$ sec. was introduced in the DC link power measurement routine, and the sampling interval for efficiency optimization increased to 2 sec.

To further demonstrate the effectiveness of the torque compensation scheme, the
Fig. 4.26 Experimental search curve at $\omega = 0.50$ pu and $T_L = 0.30$ pu.
Top: $i_d$ (3.33 A/div.); Bottom: $P_d$ (58.5 W/div.)
Time scale: 2 sec/div.
same operating point of Fig. 4.25 is repeated in Fig. 4.27, but with the pulsating torque compensation disabled. The flux reduction and the attendant reduction in developed torque results in larger speed drops, as shown in Fig. 4.27(a). Of course, the speed controller reacts to restore the speed to its set value. However, as shown in Fig. 4.27(b) (bottom trace), the increase in torque current command $i_{qs}^*$ is slow, and does not prevent the speed to drop. In some applications, speed fluctuations are not permissible, and in all cases they tend to create fluctuations in DC link power, that adversely affects the operation of the efficiency controller.

In practical drive system, the operating point can vary, either by changes in load torque or in command speed. Fig. 4.28 demonstrates the ability of the proposed control strategy to cope with sudden changes in operating condition. The drive was initially operating in a steady-state mode, at $\omega_0 = 450$ rpm (0.25 pu) and $T_L = 1.54$ lbf ft (0.10 pu), and the efficiency optimization controller brought $i_{ds}^*$ (top trace) to its optimum value. At $t=16$ sec. the command speed $\omega_r^*$ was suddenly changed to 900 rpm, forcing the system to transition to a dynamic mode, where the rated $i_{ds}^*$ was readily reestablished. At $t = 19$ sec. the system had already reached a new steady-state mode, as indicated by the speed signal (bottom trace in Fig. 4.28(a)). A new search for the optimum efficiency point was initiated, and when it was completed, another transient was imposed by setting $\omega_r^* = 450$ rpm once more. In any case the drive speed response was quite fast, demonstrating the adequacy of the proposed method for applications where fast transient response must be maintained. Fig. 4.28(b) shows the torque current command $i_{qs}^*$ (lower trace) for an identical $\omega_r^*$ profile, at same load condition. During the steady-state modes,
Fig. 4.27  Search curves at $\omega_r=0.25$ pu and $T_L=0.10$ pu, without pulsating torque compensation.
(a) Top: $i_{ds}$ (3.33 A/div.); Bottom: $\omega_r-\omega_r^*$, (61 rpm/div.)
(b) Top: $i_{ds}$ (3.33 A/div.); Bottom: $i_{qs}$ (3.33 A/div.).
Time scale: 2 sec./div.
Fig. 4.28 Drive performance in time domain with sudden changes in command speed.
(a) Top: $i_{ds}$ (3.33 A/div.); Bottom: $\omega$ (305 rpm/div.)
(b) Top: $i_{ds}$ (3.33 A/div.); Bottom: $i_{qs}$ (3.33 A/div.).
Time scale: 5 sec./div.
changes are mainly caused by the pulsating torque compensator, whereas in the
transient modes they are solely governed by the speed controller, as indicated by the
spikes in $i_{qs}$ waveform.

The drive system was further tested at several operating points of the torque-speed
plane, both at rated and optimum flux conditions, to assess the efficiency improvements.
During experimentation, it was realized that the available power measurement equipment
did not have adequate bandwidth to accurately measure the system input power. On the
other hand, the DC link power measurement scheme operating at 50 KHz possesses very
good precision. It was then clear that the system input power could be better estimated
by computing the diode rectifier loss, and then adding to the measured DC link power.
However, no specifications on the power diodes were available, and consequently, the
diode volt-ampere (V-I) characteristic had to be obtained experimentally. The
experimental data was fed to the software TABLE-CURVE [36] to obtain the diode
voltage drop equation, similar to (3.49). At every operating point, the measured DC link
power and voltage were used to compute the average DC link current ($I_{d(ave)}$), that in turn
was substituted into (3.50) to obtain the rectifier loss.

The AC-200 system has a discharge resistor continuously connected across the DC
link capacitor, such that no dangerous voltage levels remain after a few minutes the
system has being shut down. However, this design feature creates extra power loss during
normal operation. To make the results more general, this extra loss is deducted from the
measured DC link power, prior to the diode rectifier loss and efficiency computations.

Finally, the overall system efficiency was determined, and resulting efficiency
curves are shown in Fig. 4.29. As mentioned earlier in this Chapter, the DC link power and the AC supply power follow similar profiles. Consequently, the efficiency curves of Fig. 4.29 reflect true optimum system operation. The efficiency gains are expressive, particularly at light load torques, but somewhat smaller than the predicted simulated efficiency gains. A number of factors can be enumerated to explain such differences. The most significant ones are:

- The experimental hysteresis-band size is larger than the one used in simulations, due to lock-out time effect. This translates in lower order harmonic voltages and consequently, extra harmonic losses in the machine;

![Graph of experimental efficiency curves](image)

Fig. 4.29 Experimental efficiency curves.
The 5 hp class B motor is not perfectly symmetrical, i.e., the flux distribution in the air-gap has a negative sequence component, that in turn creates a second harmonic torque. As a consequence, the resultant torque is smaller than that of a perfectly symmetrical machine, and this leads to lower efficiencies.

Irrespective of the practical limitations of both machine and converter systems, the proposed fuzzy controller is capable of achieving the maximum possible efficiency for a given operating condition.

The technique is currently being extended to an electric vehicle drive, as part of a technology transition project for Delco-Remy company. The drive system under investigation is somewhat similar to that shown in Fig. 4.2, except that the indirect vector control operates in a torque control mode, and the inverter is fed by a battery bank. Fuzzy efficiency control is engaged when the vehicle is traveling at constant speed, i.e., when the drive system is in a steady-state mode. The system switches back to a dynamic mode, if the torque command from the vehicle driver \( i_{qs}^* \) is modified, or if it is detected that a speed variation has occurred. In such cases, rated flux is promptly restored to ensure maximum torque capability. The torque compensation scheme prevents torque pulsations during efficiency optimization, that could lead to reduction in riding comfort.
CHAPTER 5

FUZZY LOGIC BASED SLIP GAIN TUNING

5.1 Introduction

Indirect vector control is by far the most popular type of control for high performance induction motor drives. It offers the independent torque and flux control characteristics of a separately excited dc machine drive, with the advantages of lower machine cost, higher torque to inertia ratio and virtually maintenance free operation. One limitation of this control is its dependency on machine parameter, namely rotor resistance ($R_r$) and rotor inductance ($L_r$). It is the standard practice to use the nominal equivalent circuit parameters of the machine to compute the slip gain. For an unknown machine, the drive system can be self-commissioned, based on initial automated identification of parameters. However, machine parameters may vary widely in operating condition, due to changes in machine temperature and saturation, resulting in detuned operation.

Under detuned operation, there is a cross-coupling between flux and torque control loops, that adversely impacts their transient responses as well as results in incorrect steady state values of torque and flux. These effects will be illustrated with the help of the torque control system shown in Fig. 5.1. In this system, the flux command is kept constant at its rated value, while the torque command can be set to any value between
Fig. 5.1 Indirect vector-controlled induction motor drive with open loop torque and flux control.
zero and the maximum torque. A fixed saturation factor is assumed, i.e., machine inductance are considered constant. The steady-state characteristics of developed torque and rotor flux are shown in Fig. 5.2, for three different values of slip gain estimate \( \hat{K}_s \). When the slip gain estimate \( \hat{K}_s \) precisely matches the actual value \( K_{so} \), a linear torque transfer function is obtained, as shown by curve 2 of Fig. 5.2(a). Furthermore, the rotor flux \( \psi_r \) is unaffected by changes in torque, as indicated by curve 2 of Fig. 5.2(b). For the case \( \hat{K}_s = 0.5 K_{so} \) (curve 1), as the torque command is increased, the insufficient slip frequency causes the magnetizing current to increase. The associated overflux forces the machine into saturation and increases the core losses, that may result in overheating of the machine. On the other hand, for \( \hat{K}_s = 2 K_{so} \), the opposite situation occurs. The machine is demagnetized as the torque command increases, resulting in a reduction of the maximum torque capability, that in turn might cause the drive to stall at high load torques. It is also clear the loss of linearity of the input/output torque characteristic for \( \hat{K}_s \neq K_{so} \).

For the case a closed loop torque control system, the torque error due to detuned operation would be corrected by action of the torque controller, such that at steady-state the torque demand would be met. However, this would require a higher stator current, that in turn would lead to extra copper losses and consequently, to a derating of the machine.

The transient effects of detuning are illustrated in Fig. 5.3. In Fig. 5.3(a) the slip gain was set to twice the correct value. The cross-coupling effect is quite evident, from higher order dynamics for torque and flux transients. The oscillatory torque response renders the system unsuitable for torque control applications. On the other hand, Fig.
Fig. 5.2 Steady state detuning effects.
(a) Torque characteristics.
(b) Flux characteristics.
Fig. 5.3 Transient response for developed torque and rotor flux.
(a) Detuned condition ($K_s/K_{so} = 2$.)
(b) Tuned condition.
5.3(b) shows the ideal transient response for torque, as well as the perfectly decoupled flux response, for the case of correct slip gain.

Various on-line slip gain tuning techniques have been reported in the literature. Slip gain adaption based on the difference between ideal machine reactive power and actual reactive power was first reported by Garces [45]. Pseudo-random binary signal injection in one axis and then tuning on the basis of the induced signal in the other axis is described by Gabriel et al. [46]. Moreira et al [47] proposed a digital deadbeat controller, that derives the corrective term $\Delta K_s$ for the slip gain on the basis of an approximate machine inverse model and rotor flux estimates. The method, however, requires elaborated pre-computation of experimental saturation functions for air-gap flux and magnetizing inductance estimation. Zai and Lipo [50] used an extended Kalman filter to estimate the inverse rotor time constant on line, that makes use of the wide-band harmonic voltages, intrinsic to power inverters, as the random test signal. Systematic application of model-reference adaptive control (MRAC) technique based on reference torque model (ideal torque at tuned condition) and actual estimated torque for tuning the slip gain was reported in [48]. The MRAC tuning method was extended by Rowan et al. [49] to include voltage control method, which also evaluated the MRAC based torque ($T_e$), reactive power ($Q$), D-axis voltage ($v_{ds}$), Q-axis voltage ($v_{qs}$) and voltage magnitude ($v_5$) control methods, at fixed speed (60 Hz), and compared their performances. The study analyzed the advantages and limitations of all the methods, and concluded that none of them can solve the tuning problem satisfactorily in all the regions of torque-speed plane. Nevertheless, it was noticed from their study that the reactive power
model possesses good convergence characteristics, except at low torques. On the other hand, the D-axis voltage model exhibits excellent sensitivity to detuning, but its implementation at low speeds is complicated by the small signal to noise ratio, inherent to PWM inverters operating at small modulation factors. Furthermore, its dependency on stator resistance tends to reduce the method's precision at high load torques. Both model characteristics are discussed in detail in section 5.2. These complementary features suggests the creation of a new hybrid method, capable of performing slip gain tuning in practically the entire torque-speed plane. Such method must be able to select the most adequate model for each particular operating point, such that convergence to correct slip gain is achieved. The application of fuzzy logic in the derivation of this new method is discussed in the following sections. The results constitute the subject of a paper [63], to be presented at the 1993 international conference on industrial electronics, control and instrumentation, IECON'93.

5.2 Fuzzy Tuning Controller

Fig. 5.4 shows the simplified block diagram of the proposed tuning controller, integrated with the indirect vector controlled drive system. The close loop speed controller generates the \( i_{qs}^* \) command which then multiplies with the slip gain \( K_s \) to generate the slip frequency command \( \omega_{sl}^* \). The MRAC system generates the reference models (\( A^* \)) with the help of command currents and frequency, compares with the estimated outputs of actual models (A), and then updates the slip gain \( K_s \) through the fuzzy tuning controller.
Fig. 5.4 Indirect vector-controlled induction motor drive showing the proposed fuzzy tuning controller.
The proposed method is actually based on two models, reactive power \( (Q) \) and D-axis voltage \( (v_{ds}) \), and consequently, \( A \) and \( A^* \) are vector variables.

### 5.2.1 Reactive Power and D-axis Voltage Regulators

The derivation of reference models for reactive power \( (Q^*) \) and D-axis voltage \( (v_{ds^*}) \) control was based on the synchronous frame D-Q model of the induction machine, shown in Fig. 3.3. The stator voltage equations are given by

\[ v_{qs} = R_s i_{qs} + \frac{d}{dt} \psi_{qs} + \omega_e \psi_{ds} \quad (5.1) \]

\[ v_{ds} = R_s i_{ds} + \frac{d}{dt} \psi_{ds} - \omega_e \psi_{qs} \quad (5.2) \]

Where the stator fluxes can be expressed as

\[ \psi_{qs} = L_s i_{qs} + L_m i_{qr} \quad (5.3) \]

\[ \psi_{ds} = L_s i_{ds} + L_m i_{dr} \quad (5.4) \]

Under steady-state field orientation the following conditions apply [17]

\[ \psi_{qr} = L_r i_{qr} + L_m i_{qs} = 0 \quad (5.5) \]

\[ i_{dr} = 0 \quad (5.6) \]

From equation (5.5), \( i_{qr} \) can be expressed as
Substitution of (5.6) and (5.7) into stator flux equations 5.3 and 5.4 leads to

\[ i_{q} = -\frac{L_{m}}{L_{r}} i_{qs} \quad (5.7) \]

Substitution of (5.6) and (5.7) into stator flux equations 5.3 and 5.4 leads to

\[ \psi_{qs} = (L_{s} - \frac{L_{m}^{2}}{L_{r}}) i_{qs} = L_{o} i_{qs} \quad (5.8) \]

\[ \psi_{ds} = L_{s} i_{ds} \quad (5.9) \]

Finally, substitution of (5.8) and (5.9), and the steady-state conditions of equation 5.10, into (5.1) and (5.2), results in the field oriented stator voltage equations 5.11 and 5.12.

\[ \frac{d}{dt} \psi_{qs} = \frac{d}{dt} \psi_{ds} = 0 \quad (5.10) \]

\[ v_{qs} = R_{s} i_{qs} + \omega_{e} L_{s} i_{ds} \quad (5.11) \]

\[ v_{ds} = R_{s} i_{ds} - \omega_{e} L_{o} i_{qs} \quad (5.12) \]

The steady-state reactive power of the machine [49] is expressed by

\[ Q = v_{qs} i_{ds} - v_{ds} i_{qs} \quad (5.13) \]

Substituting \( v_{qs} \) and \( v_{ds} \) from (5.11) and (5.12) into equation 5.13 leads to

\[ Q = \omega_{e} (L_{s} i_{ds}^{2} + L_{o} i_{qs}^{2}) \quad (5.14) \]

The reference quantities \( Q^{*} \) and \( v_{ds}^{*} \) are readily obtained from the field oriented equations (5.12) and (5.14) respectively as
\[ v_{ds}^* = \hat{R}_d i_{ds}^* - \omega_e \hat{L}_d i_{qs}^* \]  

(5.15)

\[ Q^* = \omega_e (L_d i_{ds}^{*2} + L_o i_{qs}^{*2}) \]  

(5.16)

Estimates of \( L_d, L_o \) and \( R_s \) are required, but no rotor parameter is involved. The feedback variables \( Q \) is computed from (5.13), and actual \( v_{ds} \) is given by

\[ v_{ds} = v_{qs}^* \sin \theta_e + v_{ds}^* \cos \theta_e \]  

(5.17)

where \( \cos \theta_e \) and \( \sin \theta_e \) are the unit vectors available in the vector controller.

### 5.2.2 Derivation of Combined Error Signal

Fig. 5.5 shows the details of the proposed control system. The scheme depends on reference model computation of reactive power \( (Q^*) \) and \( D - \) axis voltage \( (v_{ds}^*) \) as discussed above. The reference models are compared with the respective actual estimates of the quantities, and the corresponding loop error is divided by a base value to convert to per unit variables. Handling the variables in pu form is convenient for fuzzy controller. Besides, the same controller can be easily extended to other drive systems. The base values for the respective variables are essentially the same as reference variables, i.e., \( Q_b = Q^* + \varepsilon_1 \) and \( V_b = |\omega_e L_o i_{qs}^*| + \varepsilon_2 \), where the parameters epsilon (a small positive constants) have been added to avoid indeterminate computation, in case the reference model approaches zero at critical conditions. Note that the reverse polarity of \( Q \) is due to opposite behavior of \( v_{ds} \) and \( Q \) with respect to \( \dot{\Psi}_s \), which is explained later. The first fuzzy logic controller (FLC-1) generates the weighting factor \( K_p \), which permits
Fig. 5.5 Fuzzy logic based MRAC tuning control block diagram.
appropriate distribution of $Q$ control and $v_{ds}$ control on the $i_q$s-$\omega_e$, i.e., torque-speed plane. The combined error signal is defined as

$$E = \Delta Q K_f + \Delta v_{ds} (1 - K_f)$$  \hspace{1cm} (5.18)

5.2.3 Design of the Fuzzy Tuning Controller

The second fuzzy controller (FLC-2) generates the corrective incremental slip gain $\Delta K_s$, as shown in Fig. 5.5. As the error signal is derived from two different models, both its steady-state and dynamic characteristics vary with the operating point. Under such conditions, the use of a fuzzy controller, that inherently implements distinct control laws at different operating points, ensures fast convergence at practically any point in the torque-speed plane.

In ideally tuned condition, the signals $\Delta Q$ and $\Delta v_{ds}$, and correspondingly the $E$ signal, will be zero and the slip gain $K_s$ will be set to the correct value $K_{so}$. If the system is detuned, for example due to change of rotor resistance, the actual $Q$ and $v_{ds}$ variables will deviate from the respective reference variables. The resulting error will alter the $K_s$ value until the system becomes tuned, i.e., $E = 0$. In order to derive the knowledge required for the design of the fuzzy controllers, the static characteristics of both $Q$ and $v_{ds}$ loops were initially investigated through simulation. To this end, the speed control system of Fig. 5.4 was used, but with tuning algorithm disabled. The slip gain $K_s$ was intentionally varied from 0.5 to 2 times the correct value $K_{so}$. The speed control loop ensured any steady-state operating point on torque-speed plane, even with detuned condition. Fig. 5.6 shows the normalized error relation with normalized slip gain $(\dot{K}_s/K_{so})$
Fig. 5.6 Normalized control loop error vs. normalized slip gain curves ($\omega_r=0.5$ pu).

(a) Reactive power error.
(b) D-axis voltage error.
(c) Combined error.
at speed $\omega_s = 0.5$ pu, for different load torque conditions. Note that, at ideally tuned condition $K_s/K_{so} = 1$, and correspondingly $\Delta Q (\text{pu}) = \Delta v_{ds}(\text{pu}) = E(\text{pu}) = 0$. If the system is detuned, for example with lower $K_s$ (i.e., higher rotor resistance), then $Q > Q^*$, and the controller will raise $K_s$ until the tuning point is reached. Note that the loop error tends to be very small at low torque, but increases with a higher load torque. Fig. 5.6(b) shows similar curves for the voltage control loop. The voltage loop error tends to be very large at detuned condition, except at $T_L = 1.0$ pu and $K_s/K_{so} > 1$. These characteristics indicate that the parameter $K_s$ should be large at high torque region, but small at low torque region.

Consider now the parameter variation problem in (5.15) and (5.16). If the machine parameters deviate from those used in the reference model, the slip gain will be tuned to an incorrect value. The inductance $L_s$ will decrease by saturation caused by air-gap flux, but its value will remain unchanged if rated flux is maintained. On the other hand, the inductance $L_a$ will vary by saturation due to stator current, i.e., is load dependant. The stator resistance in (5.16) will vary with stator temperature, but its effect can be negligible at high speed. The parameter variation effect, along with the characteristics in Fig. 5.6(a) and (b), indicate that the $v_{ds}$ control method is better at high speed, low torque region, whereas $Q$ control method is superior at low speed high torque region.

Fig. 5.6(c) shows the combined error $E$ (pu) as a function of $K_s/K_{so}$, with optimum $K_s$ distribution on torque-speed plane, obtained through FLC1. Fig. 5.7 gives the membership functions for frequency (i.e., speed), the current $i_{qs}$ and the weighting factor.
Fig. 5.7 Membership functions for FLC1.
(a) Speed ($\omega_e$)
(b) Torque current ($i_{qs^*}$).
(c) Weighting factor ($K_f$).
$K_r$, and Table 5.1 gives the corresponding rule base for fuzzy controller FLC-1. An
example rule can be stated as

**IF speed ($\omega_e$) is low (L) and torque ($i_{qs}^*$) is high (H) THEN weighting factor ($K_r$) is high (H).**

In fuzzy control, an imprecise information can be useful. For the controller FLC-1, even
if the system is detuned, $i_{qs}^*$ can still be used as a measure of torque, at approximately
rated flux condition. As mentioned before, the control strategy built into Rule Base I
ensures high sensitivity to detuning for any given operating condition.

Fig. 5.8 gives the membership functions for error (E), change in error (CE) and
slip gain increment ($\Delta K_s$) for the controller FLC-II, where input and output gains were
used to normalize the variables in the [-1,1] interval.

Table 5.1 Rule base for weighting factor ($K_r$) calculation.

<table>
<thead>
<tr>
<th>$i_{qs}^*$</th>
<th>$\omega_e$</th>
<th>H</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td></td>
<td>H</td>
<td>M</td>
</tr>
<tr>
<td>L</td>
<td></td>
<td>L</td>
<td>M</td>
</tr>
</tbody>
</table>
Fig. 5.8 Membership functions for FLC-2.
(a) Error (E).
(b) Change in error (CE).
(c) Increment in slip gain ($\Delta K_s$)
The asymmetry of the functions gives better resolution in control, as the respective variable approaches zero. The control loop was designed to provide overdamped $K_v$ response, and therefore, only $CE$ magnitude was considered, resulting in simplification of the rule base, as shown on Table 5.2. Broadly speaking, the control strategy, embedded into rule base 2, consists in selecting a $\Delta K_v$ increment that forces the combined error $E$ to approach zero, at a desired rate $CE$. The final rule base is a result of a fine tuning process, that used intensive simulation of the drive system, under various operating points in the torque-speed plane.

5.3 Simulation Study

Once the control algorithm was formulated in detail, the complete drive system with the proposed controller, as shown in Figs. 5.4 and 5.5, was simulated in SIMNON. The system has an outer speed control loop and a hysteresis-band PWM (HBPWM) in the inner loop. Simulation studies were conducted using 5 hp induction servomotor, originally used in the General Electric AC-200 vector control system, whose parameters are given on Table 5.3.

In order to keep the simulation time small, ideal (lossless) models were used for the rectifier and inverter subsystems, and a standard synchronous frame D-Q model was selected for the machine. The fuzzy controller was implemented using the same methodology described for the fuzzy efficiency controller, but here a 1 ms sampling time was selected. All SIMNON routines are given in Appendix A.3.
Table 5.2 Rule base for increment of slip gain ($\Delta K_s$)

<table>
<thead>
<tr>
<th>E</th>
<th>NB</th>
<th>NM</th>
<th>NS</th>
<th>ZE</th>
<th>PS</th>
<th>PM</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>ZE</td>
<td>PVS</td>
<td>PM</td>
<td>PS</td>
</tr>
<tr>
<td>M</td>
<td>NB</td>
<td>NM</td>
<td>ZE</td>
<td>ZE</td>
<td>PVS</td>
<td>PM</td>
<td>PS</td>
</tr>
<tr>
<td>S</td>
<td>NB</td>
<td>NM</td>
<td>NVS</td>
<td>ZE</td>
<td>PVS</td>
<td>PB</td>
<td>PM</td>
</tr>
<tr>
<td>Z</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>ZE</td>
<td>PVS</td>
<td>PB</td>
<td>PB</td>
</tr>
</tbody>
</table>

Table 5.3 Parameters for the induction servomotor.

5 hp  180 V  15 A
4 poles  2200 rpm

$D^e-Q^e$ equivalent circuit parameters

$R_s=0.177 \ \Omega \quad R_f=0.318 \ \Omega \quad R_m=121.4 \ \Omega$

$L_{ls}=0.00205 \ \text{H} \quad L_{lr}=0.00130 \ \text{H} \quad L_m=0.0328 \ \text{H}$
As the system dynamics varies with the slip gain, performance studies were initially carried out to fine tune the membership functions as well as the rule bases of both controllers, until optimum performance was obtained in the tuning process. Fig. 5.9 shows the normalized slip gain \( \frac{K_s}{K_{so}} \) and error \( E(\text{pu}) \) versus time, at \( \omega_r = 1100 \text{ rpm} \) (0.5 pu) for different load torques. The slip gain was intentionally set to an incorrect initial value, and the system was brought to a steady state condition at \( t=0 \), when the tuning algorithm was enabled. In Fig. 5.9(a), the load torque \( (T_L) \) is small, and therefore, the voltage control is dominant. On the other hand, in Fig. 5.9(c) the high \( T_L \) results in the dominance of reactive power control. In Fig. 5.9(b) both controls contribute equally for \( E(\text{pu}) \) computation. Evidently, the fuzzy controller is capable of bringing the system to a tuned condition in approximately 2 sec., in all cases. Similar convergence profiles were observed for other speed conditions, but are not shown here. Fig. 5.10 shows the tuning controller response when the drive is delivering rated torque at \( \omega_r = 0.5 \text{ pu} \), and the rotor resistance is suddenly increased by 100% from the initially tuned condition. Actual tuning time will be much shorter, because detuning range is hardly so large. Since large thermal time constant of the machine causes slow changes of rotor resistance, this tuning time delay is perfectly acceptable.

5.4 Hardware Design

The 5 hp vector control system described in chapter 4 was adapted for the slip gain tuning experimental study, but used with the 5 hp induction servomotor, whose parameters were given on Table 5.3. The major change concerns the machine voltage
Fig. 5.9 System tuning performance at $\omega_r = 0.5$ pu and different torques.
Fig. 5.10 Performance of tuning controller to a 100% increase in rotor resistance.
sensing interface, shown in Fig. 5.11. Its principal component is a 5 KHz bandwidth isolation amplifier (AD 204) from Analog Devices, Inc., that provides complete separation of power and control circuits. The line voltages $v_{ac}$ and $v_{cb}$ are derived by using the AD204 built in operational amplifier in a differential configuration. The large attenuation factor (80:1) of the differential amplifier brings the line voltage signal within the device's input voltage range of $\pm 5$ V. The secondary $v_{ac}$ and $v_{cb}$ signals are next used to compute the stationary frame $D^s - Q^s$ voltages $v_{ds}^s$ and $v_{qs}^s$, as shown in the following development.

By definition, the $D^s - Q^s$ voltages are obtained from the abc voltages as

$$v_{qs}^s = \frac{2}{3} v_{an} - \frac{1}{3} v_{bn} - \frac{1}{3} v_{cn}$$  \hspace{2cm} (5.19)$$

$$v_{ds}^s = -\frac{v_{bn}}{3} + \frac{v_{cn}}{3}$$  \hspace{2cm} (5.20)$$

Adding and subtracting $v_{cn}$ to (5.19) and rearranging, eqn. 5.21 is obtained

$$v_{qs}^s = (2v_{ac} + v_{cb})/3$$  \hspace{2cm} (5.21)$$

Rearranging (5.20) directly yields eqn. 5.22

$$v_{ds}^s = v_{cb}/3$$  \hspace{2cm} (5.22)$$

By using (5.21) and (5.22) in place of (5.19) and (5.20), the neutral line was not required, and only two isolation amplifiers were needed. Furthermore, by performing the abc to $D^s - Q^s$ transformation in hardware, only two A/D channels were needed. Low pass filters were used both at the power and signal sides, to attenuate the PWM related harmonics, as well as to prevent aliasing in the sampling process.
Fig. 5.11 Voltage sensing interface.
The sensing of machine current was not required, since the intrinsic operation of the HBPWM ensures that the actual currents closely track the command values. Therefore, the command currents were used also in the computation of actual $v_{ds}$ and $Q$ quantities.

### 5.5 Real Time Software Design Issues

The structure of the real time software related to the slip gain tuning is shown in Fig. 5.12. As the vector control routines are the same as those for the efficiency optimization study, they are not shown in the figure.

Proper operation of the tuning controller requires accurate measurement of the machine fundamental voltage. This task is complicated by the variable switching frequency, intrinsic to hysteresis band PWM control operation. This fact led to a conservative design of the analog low pass filters, required by the limited sampling frequency of the DSP. For the particular experimental system used, a cut-off frequency of approximately 1 KHz was selected, that has a negligible impact on the amplitude of the fundamental component, but introduces some amount of phase lag. Without any correction, the phase lag would translate into incorrect values for the synchronous frame voltages, particularly for the $v_{ds}$ signal, that usually has small amplitude. This is turn would cause improper operation of the tuning controller.

A simple steady-state compensation scheme was derived for this problem. For a first order analog low-pass filter of time constant $\tau$, the transfer function is given by
Start

Read $v_{ds}^s$ and $v_{qs}^s$

Perform vector rotation and phase-shift compensation to get $v_{ds}$ and $v_{qs}$

Extract fundamental $v_{ds}$ and $v_{qs}$ components through filtering.

Compute $Q^*, Q, \Delta Q, v_{ds}^*,$ and $\Delta v_d$.

Calculate $K_F, E$ and $CE$.

Compute $\Delta K_S$

Update $K_S$

Return

Fig. 5.12 Real time software flowchart.
\[ G(s) = \frac{1}{1 + \tau s} \]  

The phase shift \( \theta_{ps} \) introduced at frequency \( s = j\omega_e \) is determined by

\[ \theta_{ps} = \tan^{-1}(\omega_e \tau) \]  

For small \( \theta_{ps} \), \( \tan \theta_{ps} \approx \theta_{ps} \), and therefore, \( \theta_{ps} \approx \omega_e \tau \). For the case of higher order filters, the same principle can be applied, by computing the equivalent time constant \( \tau_{eq} (= \Sigma \tau_i) \), as long as the overall phase shift remains small. The mechanism of phase shift compensation is illustrated on Fig. 5.13, where only the \( v_{qs} \) component of stator voltage is considered, for simplicity. As indicated, the measured \( \hat{v}_{qs} \) lags the correct value by \( \theta_{ps} \). By subtracting the estimated \( \theta_{ps} \) from the original angle \( \theta_e \), the \( q^e \)-axis is also shifted backwards, and correct phase relationship between \( \hat{v}_{qs} \) and \( q^e \)-axis is restored.

The \( v_{ds} \) and \( v_{qs} \) equations then become

\[ v_{ds} = v_{qs}^e \sin(\theta_e - \theta_{ps}) + v_{ds}^e \cos(\theta_e - \theta_{ps}) \]  

\[ v_{qs} = v_{qs}^e \cos(\theta_e - \theta_{ps}) - v_{ds}^e \cos(\theta_e - \theta_{ps}) \]  

The parameter variation problem in the reference models given in (5.15) and (5.16) is important, since incorrect reference models would try to establish wrong slip gain \( K_s \). The parameter \( L_e \) is susceptible to magnetic saturation, as mentioned before. Even at constant flux command \( (i_{ds}^* = i_{ds}) \), use of incorrect slip gain would result in overfluxing.
Fig. 5.13 Phase-shift compensation phasor diagram.
(for $k_s < k_{so}$) or underfluxing (for $k_s > k_{so}$), causing variation on $L_s$. However, as long as the tuning algorithm possess sufficiently high sensitivity to detuning (as in the proposed method), the error polarity for a large detuning will be coherent, and the controller will drive the estimate $k_s$ towards $k_{so}$, and consequently, force machine flux towards its correct value. This translates into a self correcting effect for $L_s$, i.e., it will automatically approach its rated value, used in the reference model. It was verified during initial experimentation that the inductance $L_s$ exhibited significant variation with load torque. To enhance the performance of the controller, this saturation effect was modeled by a quadratic function of the stator current command $I_s^*$, obtained by least square estimation of the measured saturation profile, as shown in Fig. 5.14. The stator resistance $R_s$ essentially varies linearly with stator temperature, and can be corrected by measurement or estimation of stator temperature. However, no compensation has been provided for this parameter.

5.6 Experimental Study

Experimental performance of the fuzzy tuning controller was thoroughly investigated at various torques and speeds. Prior to the actual tests, the correct slip gain value $k_{so}$ was determined by off-line tests. The controller performance was primarily evaluated by intentionally initializing the slip gain estimate ($k_s$) to an incorrect value, activating the tuning controller, and then observing the time domain $k_s$ and Error (E) responses. Fig. 5.15 shows the normalized slip gain ($k_s/k_{so}$) and error (E) responses for $T_L = 0.1$ pu, when $k_s$ is initialized to twice the correct value $k_{so}$, and the tuning algorithm
Fig. 5.14 Saturation profile for $L_\sigma$. 
Fig. 5.15 Experimental tuning performance at $T_L=0.1$ pu and $K_r/K_{zo}(0)=2$.
(a) $\omega_r=0.25$ pu.  
(b) $\omega_r=0.50$ pu.
Top trace: $K_r/K_{zo}$, 0.84 /div.
Bottom trace: Error, 0.2 /div.
Time: 1 sec./div.
is enabled at $t=0$. In Fig. 5.15(a), $\omega_r = 0.25$ pu, whereas in Fig. 5.15(b) $\omega_r = 0.5$ pu. Fig. 5.16 shows similar responses for $T_L = 0.5$ pu, when $\hat{K}_s$ is initialized to half $K_{so}$, and converges to a steady-state value in about 2 sec., with an steady-state error smaller than 9.4\% in both cases. Further investigation of proper tuning is shown by the no load speed response to a square wave torque command, present in Fig. 5.17. The waveform of Fig. 5.17(a) reflects a tuned condition ($\hat{K}_s = \hat{K}_{so}$), and it differs slightly from the ideal triangular shape, due to the dragging effect of bearing friction and windage. Fig. 5.17(b) shows similar response to a detuned condition ($\hat{K}_s = 0.5\hat{K}_{so}$), where the higher order torque dynamics results in speed response degradation, manifested by waveform distortion and amplitude reduction.

In conclusion, a fuzzy logic based tuning method was proposed, that combines the features of MRAC reactive power and d-axis voltage formulations. Fuzzy logic is initially used to determine the dominant method for a given speed and load torque, to ensure a high sensitivity to detuning for the entire torque-speed plane. Next, a second fuzzy controller was used to derive the correcting term $\Delta K_s$ from the combined error and error change, that results in a fast convergence under various operating conditions. Both simulation and experimental studies were carried out, demonstrating the effectiveness of the method. The algorithm was able to converge to a good estimate of $K_s$, with only a small dependency on load torque. However, as the performance of indirect vector control drive systems is insensitive to small slip gain errors [55], this torque dependency does not constitute a practical problem for the application of the proposed technique to actual drive systems.
Fig. 5.16  Experimental tuning performance at $T_L = 0.5$ pu and $K_r/K_{so}(0) = 0.5$.
(a) $\omega_r = 0.25$ pu.  (b) $\omega_r = 0.50$ pu.
Top trace: $K_r/K_{so}$, 0.42 /div.
Bottom trace: Error, 0.1 /div.
Time: 0.5 sec./div.
Fig. 5.17 Speed response at no load for a square-wave torque command ($i_{qs}^*$).
(a) $K_s/K_{so}=1.0$.  
(b) $K_s/K_{so}=0.5$.
Top trace: $i_{qs}$, 5 A/div.
Bottom trace: $\omega_r$, 286 rpm/div.
Time: 100 msec/div.
CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

The systematic application of fuzzy logic in the control of both DC and AC drives has been discussed in this dissertation. Several control problems have been addressed, using a methodology that typically involved an initial system analysis, followed by control design and validation through digital simulation. Experimental studies were also conducted in most cases, with results that confirm the feasibility of the proposed schemes.

Fuzzy logic control was initially applied to a phase-controlled converter DC drive system that uses a separately excited DC machine. The contributions for this control system are summarized below:

. Study of converter transfer characteristics at continuous and discontinuous conduction modes;
. Development of a fuzzy logic compensation algorithm for the converter system, capable of linearizing converter transfer characteristics during discontinuous conduction mode;
. Design of fuzzy logic controllers for current and speed loops;
. Development of a methodology for simulation of fuzzy controllers in PC-
SIMNON, used in the simulation of fuzzy controllers for both DC and AC drives;

- SIMNON programs design for phase-controlled converter and DC machine;
- Comparative simulation study of fuzzy controlled DC drives and those with conventional PI controls.

The application of fuzzy $\Delta \alpha$ compensation significantly improves the speed of current loop response, particularly when combined with fuzzy current control. Use of fuzzy speed and current controllers in place of conventional PI controllers resulted in a more robust system.

The remaining research effort was devoted to indirect vector controlled induction motor drives. The on-line efficiency optimization via flux control was first considered. In order to validate the control strategy and predict the system performance, a detailed loss model of the converter induction machine system was derived.

The achievements on this subject were:

- Analysis and modeling of induction machine copper losses, core losses and stray losses, under PWM inverter supply;
- Inclusion of skin effect on rotor resistance, temperature effects on winding resistances and modeling of saturation for magnetizing inductance;
- Development of a novel synchronous frame $D^e$-$Q^e$ lossy equivalent circuit for the induction machine, capable of representing both the loss phenomenon and its dynamic behavior;
- Modeling of conduction and switching losses for a converter system, consisting of diode bridge rectifier and PWM transistor inverter;
Implementation and testing of machine and converter models in PC-SIMNON, for a 10 hp drive system.

A consistent performance was observed both at steady-state and dynamic conditions. Actual accuracy of lossy models is dependent on the precision with which the machine and converter parameters can be obtained. Currently, one disadvantage of the lossy models is the long simulation time, when using PC-SIMNON. This is especially true when a long term trend is important, such as efficiency optimization studies. Of course, this problem can be minimized by using more powerful computers.

The efficiency optimization study resulted in the following developments:

- Design of a fuzzy logic controller that adaptively adjusts the magnetizing current on the basis of measured DC link power, until the optimum efficiency is achieved;
- Introduction of a feedforward torque compensator to suppress the low frequency pulsating torque, due to changes in flux;
- Development of a criterion for transition between efficiency optimization and transient optimization modes;
- Comprehensive simulation study of a 5 hp drive system with proposed efficiency and torque compensation controls;
- Construction of hysteresis band current controller, protection and monitoring circuits, DC link voltage and current sensing interfaces;
- Design and testing of a speed sensing interface for the TMS320C25 DSP;
- Development of assembly language software for the vector control system;
- Design of assembly language programs for efficiency optimization, torque...
compensation and transition controls;

. Experimental evaluation of steady-state and transient performances of proposed control schemes.

The experimental performance correlates well with that predicted by simulation. The efficiency gains are somewhat smaller than expected, because some practical aspects are difficult to model. For example, actual induction machines have some phase unbalance, that creates torque fluctuations and adversely impacts machine efficiency. Irrespective of these problems, the proposed controller ensures true optimum efficiency operation. The method can be incorporated into an existing vector drive system, with a minimum of extra hardware. In most cases, the added software can be easily implemented by the existing processor, resulting in a very cost effective efficiency optimized drive.

The proposed fuzzy efficiency controller is applicable to a number of practical drive systems. In fact, it is currently being considered for an electrical vehicle drive, as part of a technology transfer project to Delco-Remy company.

Finally, fuzzy logic application to the slip gain tuning of an indirect vector-controlled induction motor drive was considered. The salient contributions of the study are:

. An initial fuzzy controller was designed, that combines two MRAC models, D-axis voltage and reactive power, to generate the error signal. This ensures good control sensitivity in the entire torque-speed plane;

. A second fuzzy controller was designed to tune the slip gain based on combined error signal and its slope, resulting in fast convergence at any load condition;
Simulation programs were developed for the fuzzy tuning controllers and vector control drive, followed by a comprehensive simulation study;

A hardware voltage interface for \( v_{ds} \) and \( v_{qs} \) sensing was designed and built;

Assembly language programs were developed for the fuzzy tuning controllers, and experimental validation carried out, for a 5 hp drive system.

The test results correlate well with the theory and simulation performance. All the tuning controls were executed by the same TMS320C25 DSP, used for the basic vector control function. Therefore, the extra cost involved in the fuzzy slip gain tuning controller implementation is reasonably low.

Further research topics to extend the current work would comprehend:

- Use of adjustable DC link voltage topologies to minimize machine harmonic losses;
- Extend the fuzzy tuning control to flux weakening region, along with proper saturation modeling of machine inductances;
- Integration of slip gain tuning technique to efficiency optimization control, to ensure proper pulsating torque compensation as well as fast torque response;
- Investigate the potential use of neural networks to the implementation of fuzzy controllers as well as for better representation of saturation effects.

The fuzzy logic control of drive systems has the potential of enhancing system performance, as well as to reduce sensitivity to disturbances and parameter variation effects. Of course, other control techniques such as Model-Reference Adaptive Control (MRAC), variable structure control (VSC) can also be applied to get robust system
performance. Fuzzy control, however, has the advantages of not requiring a mathematical model of the plant and delivering a chatter free system response.

When compared to other AI techniques, fuzzy control shares some common features with both expert systems and neural networks. While expert systems are ideal for structured knowledge representation and symbolic processing, they possess poor numerical capabilities and are not adequate for real time control of fast power electronic systems. On the other side, neural networks are very efficient in processing unstructured knowledge. Typically, a neural network can be trained to implement a desired mapping, such as a non-linear controller, provided that an extensive numerical input/output data is available to train the network. Although the automated training makes it simpler to tune than a heuristic based fuzzy controller, the training process is not always convergent, and in many cases not very precise. Furthermore, neural networks lack the ability to incorporate qualitative knowledge.

Recently, fuzzy neural networks have been considered. Essentially, neural network techniques are applied in the implementation of parts of a fuzzy controller, such as in the representation of more elaborate membership functions. It can be viewed as an attempt to combine the strengths of both technologies, such as the qualitative knowledge representation of fuzzy logic, with the numerical capabilities of neural networks.

In conclusion, it is expected that, as the research on fuzzy control and other emerging technologies evolves, a formal methodology for fuzzy logic control design and analysis will be developed, along with new CAD tools. Then, the full benefit of this new technology will be realized.
REFERENCES


APPENDICES
A.1 DC Drive Programs

```
" --------------- ---------------
| DC MOTOR       |
" --------------- ---------------

CONTINUOUS SYSTEM DCM

INPUT Vt
OUTPUT W1 Ea X Ia IAPU
TIME T
STATE W Ia Ila
DER DW Dla Dlla

"-------------------- Define state variables.
DW=(-B/J)*W+(Te-TL)/J  "Speed.
Dla=(-Ra/La)*Ia+(Vt-Ea)/La  "Armature current.
Dlla=Ia  "Integral of Ia, for average computation.
Te=Kt*Ia  "Electromagnetic torque.
TL=KL*W*W  "Load torque.
Ea=K1*W  "Counter emf.
Eapu=Ea/Vm

"-------------------- Auxiliary variables for OUTPUT.
W1=W  "Used in FCC.
X=Ila  "Used in RCFR.
Ial=Ia
Iapu=Ia/53.73

"-------------------- Parameters.
Kt:0.55  "Torque constant.
K1:0.55  "Flux constant.
La:0.008  "Armature inductance.
Ra:0.6  "Armature resistance.
J:0.0465  "Inertia.
B:0.004  "Viscous damping.
Vm:169.7  "Base value.
KL:2.78E-04  "Load constant.
END

" --------------- ---------------
| FUZZY DELTA ALPHA COMPENSATION |
" --------------- ---------------

DISCRETE SYSTEM ACOMP
STATE A D  "A, D Store the previous values of alpha
NEW NA ND  " and delta alpha , respectively.
INPUT ALPHA Ia lar
```
OUTPUT ALP1 ALP2 IDN  "Final values of alpha.

TIME T
TSAMP Z
Z = T + DT

ALFA=(ALPHA*180)/3.1416  "Alpha in degrees.
IDN=IF Ia>0 THEN Ia/Base ELSE 0  "PU current.
J=INT((ALFA+10)/20)  "Interval indices.
I=INT((IDN+0.01)/0.01)

" ------------------- Get degrees of membership and perform rule base evaluation.
MA1=(20*J+10-ALFA)/20
MID1=(0.01*I-IDN)/0.01
MRA=MIN(MID1,MA1)
MRB=MIN(MID1,(1-MA1))
MRC=MIN((1-MID1),MA1)
MRD=MIN((1-MID1),(1-MA1))

" ------------------- Compute value of delta alpha.
IJ=I+J  "Retrieve contribution of first rule.
DAA=IF 1<2 THEN C1 ELSE IF 1<3 THEN C2 ELSE IF 1<4 THEN C3 ELSE T1
T1=IF 1=5 THEN C4 ELSE IF 1<6 THEN C5 ELSE IF 1=7 THEN C6 ELSE T2
T2=IF 1<8 THEN C7 ELSE IF 1<9 THEN C8 ELSE IF 1<10 THEN C9 ELSE 0

" C1=IF J<3 THEN 30 ELSE IF J<5 THEN 27 ELSE 25
C2=IF J<2 THEN 5 ELSE IF J<3 THEN 16 ELSE 17
C3=IF J<2 THEN 0 ELSE IF J<3 THEN 10 ELSE 12
C4=IF J<2 THEN 0 ELSE IF J<3 THEN 6 ELSE 9
C5=IF J<2 THEN 0 ELSE IF J<3 THEN 2 ELSE 7
C6=IF J<3 THEN 0 ELSE IF J<4 THEN 4 ELSE 6
C7=IF J<3 THEN 0 ELSE IF J<4 THEN 2 ELSE 4
C8=IF J<4 THEN 0 ELSE 2
C9=IF J<4 THEN 0 ELSE 1

" L=J+1  "Retrieve contribution of second rule.
DAB=IF 1<2 THEN D1 ELSE IF 1<3 THEN D2 ELSE IF 1<4 THEN D3 ELSE T6
T6=IF 1=5 THEN D4 ELSE IF 1<6 THEN D5 ELSE IF 1<7 THEN D6 ELSE T7
T7=IF 1<8 THEN D8 ELSE IF 1<9 THEN D9 ELSE T10

" D1=IF L<3 THEN 30 ELSE IF L<5 THEN 27 ELSE 25
D2=IF L<3 THEN 16 ELSE 17
D3=IF L<3 THEN 10 ELSE 12
D4=IF L<3 THEN 6 ELSE 9
D5=IF L<3 THEN 2 ELSE 7
D6=IF L<3 THEN 0 ELSE IF L<4 THEN 4 ELSE 6
D7=IF L<3 THEN 0 ELSE IF L<4 THEN 2 ELSE 4
D8=IF L<4 THEN 0 ELSE 2
D9=IF L<4 THEN 0 ELSE 1

" M=J+1  "Retrieve contribution of third rule.
DAC=IF M<3 THEN C2 ELSE IF M<4 THEN C3 ELSE IF M<5 THEN C4 ELSE T9
T9=IF M<6 THEN C5 ELSE IF M<7 THEN C6 ELSE IF M<8 THEN C7 ELSE T10
T10=IF M<9 THEN C8 ELSE IF M<10 THEN C9 ELSE 0
DAD=IF M<3 THEN D2 ELSE IF M<4 THEN D3 ELSE T11
T11=IF M<5 THEN D4 ELSE IF M<6 THEN D5 ELSE T12
T12=IF M<7 THEN D6 ELSE IF M<8 THEN D7 ELSE T13
T13=IF M<9 THEN D8 ELSE IF M<10 THEN D9 ELSE 0

" Test of overflow and underflow.
OV=IF J=9 THEN 0 ELSE IF J>9 THEN ELSE 1
UF=IF IJ<2 THEN 0 ELSE 1

202
COND=OV*UF*FLAG
SUM=MRA+MRB+MRC+MRD
PROD=DAA*MRA+DAB*MRB+DAC*MRC+DAD*MRD

"------------------------ Define final value of alpha depending on sign of E=IAR-IA"
ND=DALFA
DK=PROD/SUM
E=IAR-IA "Current loop error.
DALFA=IF COND<1 THEN 0 ELSE IF E>0 THEN CD ELSE CO2
CD=IF DK<D THEN DK ELSE D
CO2=IF DK>D THEN DK ELSE D
ALP1=IF T>0.0028 THEN AU ELSE ALPHA
AU=ALPHA+(DALFA*3.1416)/180 "Alpha in rad.
NA=ALP1
ALP2=A

" ----------------------- Parameters."
DT:2.778E-03 "Sampling interval.
Ibase:53.73 "Base armature current.
FLAG:1 "FLAG=1: compensation on; =0 compensation off.
A:1.5708 "Initial condition for ALP2.
END

" "

" Fuzzy Speed Controller"
" Discrete System FSC2"

"Note: seven fuzzy sets are used for state variables E and CE, while eleven are used for DU."

INPUT W
OUTPUT IAR
STATE X U
NEW NX NU
TIME T
TSAMP TS
TS=T+AT

" "

" Compute error FE, change in error FCE."
FE=0.10472*WR-W
FCE=FE-X " X = E(k-1)
E=FE/GEs "Get the PU values E and CE.
C=FCE/GCs

" "

" Compute preliminary values of degree of membership."
A1=MIN(1,(E-Es1)/(1-Es1))
A2=(E-Es2)/(Es1-Es2)
A3=E/Es2
A5=-E/Es2
A6=(E-Es2)/(Es1-Es2)
A7=(E-Es1)/(1-Es1)
B1=MIN(1,(C-Cs1)/(1-Cs1))
B2=(C-Cs2)/(Cs1-Cs2)
B3=C/Cs2
B5=-C/Cs2
B6=(C-Cs2)/(Cs1-Cs2)
B7=(C-Cs2)/(1-Cs1)

203
Get interval indices \( J \) and \( I \), for \( E \) and \( CE \), respectively.

\[
\begin{align*}
J &= \text{IF } E < ES_1 \text{ THEN } 1 \text{ ELSE IF } E < ES_2 \text{ THEN } 2 \text{ ELSE IF } E < 0 \text{ THEN } 3 \text{ ELSE } T1 \\
T1 &= \text{IF } E < ES_2 \text{ THEN } 4 \text{ ELSE IF } E < ES_1 \text{ THEN } 5 \text{ ELSE } 6 \\
I &= \text{IF } C < CS_1 \text{ THEN } 1 \text{ ELSE IF } C < CS_2 \text{ THEN } 2 \text{ ELSE IF } C < 0 \text{ THEN } 3 \text{ ELSE } T2 \\
T2 &= \text{IF } C < CS_2 \text{ THEN } 4 \text{ ELSE IF } C < CS_1 \text{ THEN } 5 \text{ ELSE } 6
\end{align*}
\]

Compute degree of membership for relevant fuzzy sets.

\[
\begin{align*}
ME_1 &= \text{IF } J < 2 \text{ THEN } \min(1, A_7) \text{ ELSE IF } J < 3 \text{ THEN } A_6 \text{ ELSE } T3 \\
T3 &= \text{IF } J < 4 \text{ THEN } A_5 \text{ ELSE IF } J < 5 \text{ THEN } 1 - A_3 \text{ ELSE IF } J < 6 \text{ THEN } 1 - A_2 \text{ ELSE } \min(1, 1 - A_1) \\
ME_2 &= 1 - ME_1 \\
MC_1 &= \text{IF } I < 2 \text{ THEN } \min(1, B_7) \text{ ELSE IF } I < 3 \text{ THEN } B_6 \text{ ELSE IF } I < 4 \text{ THEN } B_5 \text{ ELSE } T4 \\
T4 &= \text{IF } I < 5 \text{ THEN } 1 - B_3 \text{ ELSE IF } I < 6 \text{ THEN } 1 - B_2 \text{ ELSE } \min(1, 1 - B_1) \\
MC_2 &= 1 - MC_1
\end{align*}
\]

Perform rule base evaluation.

\[
\begin{align*}
M_{RA} &= \min(ME_1, MC_1) \\
M_{RB} &= \min(ME_2, MC_1) \\
M_{RC} &= \min(ME_1, MC_2) \\
M_{RD} &= \min(ME_2, MC_2)
\end{align*}
\]

Retrieve the control signal for each fired rule.

First rule.

\[
\begin{align*}
D_1 &= \text{IF } J < 2 \text{ THEN } C_1 \text{ ELSE IF } J < 3 \text{ THEN } C_2 \text{ ELSE IF } J < 4 \text{ THEN } C_3 \text{ ELSE } T5 \\
T5 &= \text{IF } J < 5 \text{ THEN } C_4 \text{ ELSE IF } J < 6 \text{ THEN } C_5 \text{ ELSE } C_6 \\
L &= J + 1 \\
D_2 &= \text{IF } L < 3 \text{ THEN } C_2 \text{ ELSE IF } L < 4 \text{ THEN } C_3 \text{ ELSE IF } L < 5 \text{ THEN } C_4 \text{ ELSE } T6 \\
T6 &= \text{IF } L < 6 \text{ THEN } C_5 \text{ ELSE IF } L < 7 \text{ THEN } C_6 \text{ ELSE } C_7
\end{align*}
\]

In the first column of the rule base.

\[
\begin{align*}
C_1 &= \text{IF } I < 2 \text{ THEN } -1 \text{ ELSE IF } I < 5 \text{ THEN } -U_1 \text{ ELSE IF } I < 6 \text{ THEN } -U_2 \text{ ELSE } -U_3 \\
C_2 &= \text{IF } I < 2 \text{ THEN } -1 \text{ ELSE IF } I < 3 \text{ THEN } -U_1 \text{ ELSE IF } I < 5 \text{ THEN } -U_2 \text{ ELSE } T6A \\
T6A &= \text{IF } I < 6 \text{ THEN } -U_3 \text{ ELSE } 0 \\
C_3 &= \text{IF } I < 2 \text{ THEN } -U_1 \text{ ELSE IF } I < 3 \text{ THEN } -U_2 \text{ ELSE IF } I < 5 \text{ THEN } -U_3 \text{ ELSE } T7 \\
T7 &= \text{IF } I < 6 \text{ THEN } 0 \text{ ELSE } U_3 \\
C_4 &= \text{IF } I < 2 \text{ THEN } -U_1 \text{ ELSE IF } I < 3 \text{ THEN } -U_2 \text{ ELSE IF } I < 4 \text{ THEN } -U_3 \text{ ELSE } T8 \\
T8 &= \text{IF } I < 5 \text{ THEN } 0 \text{ ELSE IF } I < 6 \text{ THEN } U_3 \text{ ELSE } U_2 \\
C_5 &= \text{IF } I < 2 \text{ THEN } -U_2 \text{ ELSE IF } I < 3 \text{ THEN } -U_3 \text{ ELSE IF } I < 4 \text{ THEN } 0 \text{ ELSE } T9 \\
T9 &= \text{IF } I < 6 \text{ THEN } U_3 \text{ ELSE } U_2 \\
C_6 &= \text{IF } I < 2 \text{ THEN } -U_3 \text{ ELSE IF } I < 3 \text{ THEN } 0 \text{ ELSE IF } I < 4 \text{ THEN } U_3 \text{ ELSE } U_2 \\
C_7 &= \text{IF } I < 2 \text{ THEN } 0 \text{ ELSE IF } I < 3 \text{ THEN } U_3 \text{ ELSE IF } I < 4 \text{ THEN } U_2 \text{ ELSE } U_1
\end{align*}
\]

Access third and fourth rules.

\[
\begin{align*}
D_{21} &= \text{IF } J < 2 \text{ THEN } V_1 \text{ ELSE IF } J < 3 \text{ THEN } V_2 \text{ ELSE IF } J < 4 \text{ THEN } V_3 \text{ ELSE } T10 \\
T10 &= \text{IF } I < 5 \text{ THEN } V_4 \text{ ELSE IF } J < 6 \text{ THEN } V_5 \text{ ELSE } V_6
\end{align*}
\]

\[
\begin{align*}
D_{22} &= \text{IF } L < 3 \text{ THEN } V_2 \text{ ELSE IF } L < 4 \text{ THEN } V_3 \text{ ELSE IF } L < 5 \text{ THEN } V_4 \text{ ELSE } T11 \\
T11 &= \text{IF } L < 6 \text{ THEN } V_5 \text{ ELSE IF } L < 7 \text{ THEN } V_6 \text{ ELSE } V_7
\end{align*}
\]

In the \( i \)-th column of the rule base.

\[
\begin{align*}
V_1 &= \text{IF } M < 5 \text{ THEN } -U_1 \text{ ELSE IF } M < 6 \text{ THEN } -U_2 \text{ ELSE IF } M < 7 \text{ THEN } -U_3 \text{ ELSE } 0 \\
V_2 &= \text{IF } M < 2 \text{ THEN } -1 \text{ ELSE IF } M < 3 \text{ THEN } -U_1 \text{ ELSE } T12 \\
T12 &= \text{IF } M < 5 \text{ THEN } -U_2 \text{ ELSE IF } M < 6 \text{ THEN } -U_3 \text{ ELSE } T12B \\
T12B &= \text{IF } M < 7 \text{ THEN } 0 \text{ ELSE } U_3 \\
V_3 &= \text{IF } M < 3 \text{ THEN } -U_2 \text{ ELSE IF } M < 5 \text{ THEN } -U_3 \text{ ELSE IF } M < 6 \text{ THEN } 0 \text{ ELSE } T13 \\
T13 &= \text{IF } M < 7 \text{ THEN } U_3 \text{ ELSE } U_2 \\
V_4 &= \text{IF } M < 3 \text{ THEN } -U_2 \text{ ELSE IF } M < 4 \text{ THEN } -U_3 \text{ ELSE IF } M < 5 \text{ THEN } 0 \text{ ELSE } T14 \\
T14 &= \text{IF } M < 6 \text{ THEN } U_3 \text{ ELSE IF } M < 7 \text{ THEN } U_2 \text{ ELSE } U_1 \\
V_5 &= \text{IF } M < 3 \text{ THEN } -U_3 \text{ ELSE IF } M < 4 \text{ THEN } 0 \text{ ELSE IF } M < 6 \text{ THEN } U_3 \text{ ELSE } U_2 \\
V_6 &= \text{IF } M < 3 \text{ THEN } 0 \text{ ELSE IF } M < 4 \text{ THEN } U_3 \text{ ELSE IF } M < 7 \text{ THEN } U_2 \text{ ELSE } 1 \\
V_7 &= \text{IF } M < 3 \text{ THEN } U_3 \text{ ELSE IF } M < 4 \text{ THEN } U_2 \text{ ELSE IF } M < 7 \text{ THEN } U_1 \text{ ELSE } 1
\end{align*}
\]

Defuzzification using height method.
SUM=MRA+MRB+MRC+MRD
PROD=D11*MRA+D12*MRB+D21*MRC+D22*MRD
DU=PROD/SUM

" Compute the next control signal.
UK=U+DU*GUs
IAR=IF UK<UMIN THEN UMIN ELSE MIN(UK,UMAX)
NU=IAR

Wr=IF T<0.8 THEN WR1 ELSE WR2
WR1:1000
WR2:1000

" Parameters for the fuzzy controller.
Cs1:0.5
Cs2:0.2
Es1:0.5
Es2:0.2
U1:0.5
U2:0.125
U3:0.05
U4:0.01
UMAX:25
Umin:0
GES:20
Gcs:0.9
GUs:12.5
AT:2.778E-03
END

" FUZZY CURRENT CONTROLLER
" DISCRETE SYSTEM FCC
INPUT Iar Ila Ea
OUTPUT ALPHA1 ALPHA2 Ia Vt
STATE X U I A
NEW NX NU NI NA
TIME T
TSAMP TS
TS=T+AT

NI=Ila
Ia=(Ila-1)/AT

" Compute average Ia.

FE=Iar-Ia
FCE=FE-X
NX=FE
E=FE/GEc
C=FCE/GCc

" PU values.

" Compute preliminary degrees of membership.
A1=MIN(1,(E-Ec1)/(1-Ec1))
A2=(E-Ec2)/(Ec1-Ec2)
A3=E/Ec2
A5=-E/Ec2
A6=(-(E-Ec2)/(Ec1-Ec2)
A7=(-(E-Ec1)/(1-Ec1)
B1 = \text{MIN}(1, (C - Cc1)/(C - Cc1))
B2 = (C - Cc2)/(Cc1 - Cc2)
B3 = C/Cc2
B4 = -C/Cc2
B5 = -(C - Cc2)/(Cc1 - Cc2)
B6 = -(C - Cc1)/(C - Cc1)

Get interval indices J and H, for E and C, respectively.
J = \text{IF } E < Ec1 \text{ THEN } 1 \text{ ELSE IF } E < Ec2 \text{ THEN } 2 \text{ ELSE IF } E < 0 \text{ THEN } 3 \text{ ELSE } T1
T1 = \text{IF } E < Ec2 \text{ THEN } 4 \text{ ELSE IF } E < Ec1 \text{ THEN } 5 \text{ ELSE } 6
H = \text{IF } C < Cc1 \text{ THEN } 1 \text{ ELSE IF } C < Cc2 \text{ THEN } 2 \text{ ELSE IF } C < 0 \text{ THEN } 3 \text{ ELSE } T2
T2 = \text{IF } C < Cc2 \text{ THEN } 4 \text{ ELSE IF } C < Cc1 \text{ THEN } 5 \text{ ELSE } 6

---Compute degree of membership for relevant fuzzy sets.
ME1 = \text{IF } J < 2 \text{ THEN } \text{MIN}(1, A7) \text{ ELSE IF } J > 0 \text{ THEN } A6 \text{ ELSE } T3
T3 = \text{IF } J < 4 \text{ THEN } A5 \text{ ELSE IF } J > 3 \text{ THEN } 1 - A3 \text{ ELSE } T3B
T3B = \text{IF } J < 6 \text{ THEN } 1 - A2 \text{ ELSE MIN}(1, 1 - A1)
ME2 = 1 - ME1
MC1 = \text{IF } H < 2 \text{ THEN } \text{MIN}(1, B7) \text{ ELSE IF } H > 3 \text{ THEN } B6 \text{ ELSE } T4
T4 = \text{IF } H < 4 \text{ THEN } B5 \text{ ELSE IF } H < 3 \text{ THEN } 1 - B3 \text{ ELSE } T4B
T4B = \text{IF } H < 6 \text{ THEN } 1 - B2 \text{ ELSE MIN}(1, 1 - B1)
MC2 = 1 - MC1

---Perform rule base evaluation.
MRA = \text{MIN}(ME1, MC1)
MRB = \text{MIN}(ME2, MC2)
MRC = \text{MIN}(ME1, MC2)
MRD = \text{MIN}(ME2, MC2)

---Retrieve the control signal for each fired rule.
C1 = IF H < 2 THEN -1 ELSE IF H < 3 THEN -U1 ELSE IF H < 6 THEN -U2 ELSE -U3
C2 = IF H < 2 THEN -1 ELSE IF H < 3 THEN -U1 ELSE IF H < 5 THEN -U2 ELSE T6A
T6A = IF H < 6 THEN -U3 ELSE 0
C3 = IF H < 2 THEN -U1 ELSE IF H < 3 THEN -U2 ELSE IF H < 5 THEN -U3 ELSE T7
T7 = IF H < 6 THEN 0 ELSE U3
C4 = IF H < 2 THEN -U1 ELSE IF H < 3 THEN -U2 ELSE IF H < 4 THEN -U3 ELSE T8
T8 = IF H < 5 THEN 0 ELSE IF H < 6 THEN U3 ELSE U2
C5 = IF H < 2 THEN -U2 ELSE IF H < 3 THEN -U3 ELSE IF H < 4 THEN 0 ELSE T9
T9 = IF H < 6 THEN U3 ELSE U2
C6 = IF H < 2 THEN -U3 ELSE IF H < 3 THEN 0 ELSE IF H < 4 THEN U3 ELSE U2
C7 = IF H < 2 THEN 0 ELSE IF H < 3 THEN U3 ELSE IF H < 4 THEN U2 ELSE U1
C8 = IF H < 2 THEN -U1 ELSE IF H < 3 THEN -U2 ELSE -U3
C9 = IF H < 2 THEN -U2 ELSE IF H < 3 THEN -U3 ELSE -U2
C10 = IF H < 2 THEN -U3 ELSE -U1

---Access third and fourth rules.
D21 = IF J < 2 THEN Vi ELSE IF J = 3 THEN V2 ELSE IF J = 4 THEN V3 ELSE T10
T10 = IF J < 5 THEN V4 ELSE IF J = 6 THEN V5 ELSE V6
D22 = IF L < 3 THEN V2 ELSE IF L < 4 THEN V3 ELSE IF L = 5 THEN V4 ELSE T11
T11 = IF L < 6 THEN V5 ELSE IF L < 7 THEN V6 ELSE V7

---Vi also represents the i-th column of the rule base.
V1 = IF M < 5 THEN -U1 ELSE IF M < 6 THEN -U2 ELSE IF M < 7 THEN -U3 ELSE 0
V2 = IF M < 2 THEN -1 ELSE IF M < 3 THEN -U1 ELSE T12
T12 = IF M < 5 THEN -U2 ELSE IF M < 6 THEN -U3 ELSE IF M < 7 THEN 0 ELSE U3
V3=IF M<3 THEN -U2 ELSE IF M<5 THEN -U3 ELSE IF M<6 THEN 0 ELSE T13
T13=IF M<7 THEN U3 ELSE U2
V4=IF M<3 THEN -U2 ELSE IF M<4 THEN -U3 ELSE IF M<5 THEN 0 ELSE T14
T14=IF M<6 THEN U3 ELSE U2 ELSE U1
V5=IF M<3 THEN -U3 ELSE IF M<4 THEN 0 ELSE IF M<6 THEN U2 ELSE U2
V6=IF M<3 THEN 0 ELSE IF M<4 THEN U3 ELSE IF M<7 THEN U2 ELSE 1
V7=IF M<3 THEN U3 ELSE IF M<4 THEN U2 ELSE IF M<7 THEN U1 ELSE 1

" " ------------------ Defuzzification using height method.
SUM=MRA+MRB+MRC+MRD
PROD=DI1*MRA+DI2*MRB+D21*MRC+D22*MRD
DU=PROD/SUM

" " Compute the next control signal.
UK=DU*GUc+U
UKE=UK+Ea
Vt=IF UKE<Umin THEN Umin ELSE MIN(UKE,Umax)
NU=UK+Vt-UKE

" " Calculate firing angle by cosine crossing method.
ALPHAI=ARCCOS(Vt/(1.35*VL))
NA=ALPHAI
ALPHA2=A

" " Parameters for the fuzzy controller.
UMAX:110
Umin:-81
AT:2.778E-03
VL:90
Cc1:0.5
Cc2:0.2
Ec1:0.5
Ec2:0.2
U1:0.5
U2:0.125
U3:0.05
GEc:25
GCc:5
GUc:40

" " Initial values
A:1.5708
END

" " PHASE-CONTROLLED RECTIFIER
" CONTINUOUS SYSTEM RCFR
" INPUT ALPHA1 ALPHA2 Ia Ea
OUTPUT Vt IVT VTPU
STATE IV
USED TO GENERATE Vdc IN CPI
DER DIV
TIME T
TSH=T-N*2.778E-03
N=INT((T-1E-10)/2.778E-03)

" " DEFINE BASIC VOLTAGES PROFILES.
Vab=Vmax*SIN(1.0472+W*TSH)
Vcb = Vmax * SIN(2.0944 + W * TSH)
Vca = Vmax * SIN(3.1416 + W * TSH)

" --------------- FIRING ANGLE REFLECTED IN THE INTERVAL 0 - 60 DEG.
UA = IF ALPHAI < 1.0472 THEN ALPHAI ELSE ALPHAI - 1.0472
FT = UA / W
Void = IF ALPHAI < 1.0472 THEN Vcb ELSE Vca
Vnew = IF ALPHAI < 1.0472 THEN Vab ELSE Vcb
VTI = IF TSH < FT THEN Void ELSE Vnew
VT = IF VT > Ea THEN VT ELSE CONI
CONI = IF la > 1E-03 THEN VT ELSE Ea

" --------------- EVALUATE INTEGRAL OF VT
Vtpu = Vt / Em
DIV = Vtpu
IVT = IV

" --------------- Parameters.
W = 376.99
Vmax = 127.28

END

" ------------------------ DISCRETE PI CURRENT CONTROLLER
" Note: INCORPORATES ANTI-WINDUP.
DISCRETE SYSTEM CPI
INPUT Iar Ila II IVT Ea "Ii is the actual current and Ila is its integral.
OUTPUT ALPHAI ALPHA2 la Vt VDC
STATE X I Vd A " A = ALPHAI(k-1)
NEW NX NI NVd NA
TIME T
TSAMP TS
TS = T + AT

" --------------- Computation of Idc and Vdc (pu).
NI = IIa
la = (Ila - l) / AT
NVd = IVT
Vdc = (IVT - Vd) / AT

" --------------- COMPUTE ERROR E, CONTROL SIGNAL U
E = Iar - la
V = Kpi * E + X + Ea
U = IF V < Umin THEN Umin ELSE IF V > UMAX THEN UMAX ELSE V
NX = X + Kii * E * AT
Vt = U
ALPHA1 = ARCCOS(Vt / (1.35 * VL)) " Firing angle.
NA = ALPHA1
ALPHA2 = A " Alpha(k-1)

" --------------- PARAMETERS
UMAX = 110 "Max. and min. control output.
Umin = -81
Kpi = 1.2 "PI gains.
Kii = 80
AT = 2.778E-03 "Sampling interval.
VL = 90 "Line voltage.

" --------------- INITIAL VALUES.
" DISCRETE PI SPEED CONTROLLER
" NOTE: Using anti-windup.
DISCRETE SYSTEM SPI
INPUT W
OUTPUT Iar
STATE X
NEW NX
TIME T
TSAMP TS
TS=T+ADT

" COMPUTE ERROR E AND CONTROL SIGNAL U.
E=0.10472*Wr-W
V=Kps*E+X
U=IF V<0 THEN 0 ELSE IF V>UMAX THEN UMAX ELSE V
NX=X+Kis*E*ADT+U-V "WITH ANTI WINDUP
Iar=U
Wr=IF T<0.8 THEN Wr1 ELSE Wr2 "Reference current.

" PARAMETERS.
Kis:28 "PI gains.
Kps:2
ADT:2.778E-03 "Sampling interval.
Wr1:1000
Wr2:1000
END

A.2 Loss Modeling and Efficiency Optimization Programs

" AC POWER SUPPLY
" CONTINUOUS SYSTEM AC
OUTPUT vas vbs vcs we
TIME t

" Generate 3-phase sinusoidal voltages
vas=Vm*sin(wt)
vbs=Vm*sin(wt-2.0944)
vcs=Vm*sin(wt-4.1888)
w=we*t
we=377
Vm:187.79
END
"LOSSY MODEL OF A THREE-PHASE DIODE BRIDGE RECTIFIER

CONTINUOUS SYSTEM RCFL
INPUT Vas Vbs Vcs I2
OUTPUT Vd Pild Pior Piir
STATE Vc I1
DER DVc DII
TIME t

"GENERATION OF IDEAL RECTIFIER VOLTAGE Vr
Vp = MAX(Vas, MAX(Vbs, Vcs)) "Positive profile,
Vn = MIN(Vas, MIN(Vbs, Vcs)) "Negative profile
Vr = Vp-Vn

"EQUATIONS FOR CURRENT AND VOLTAGES
Vdd = Vd0+Cd*ABS(Il)*Ed "Diode voltage drop
DVc = (11-I2)/C "Capacitor voltage
DII = IF Il<0 AND AUX<0 THEN 0 ELSE AUX "Current through L
AUX = (Vr-2*Vdd-Vc)/L "Voltage across L
Vd = Vc

"INSTANTANEOUS POWERS AND LOSSES
Pild = 2*Vdd*abs(Il) "Instantaneous condution losses in diodes
Pior = Vd*I1 "Instantaneous power output of rectifier
Piir = Vr*I1 "Instantaneous power input of rectifier

"INITIAL VALUES
Vc:314
I1:1.57

"PARAMETERS FOR POWEREX CD 411230 (30 A/1200-1600 V)
C:0.0048 "DC link capacitor
L:0.001 "DC link equivalent of AC source inductance
Cd:0.032 "Volt-ampere coefficient of diode characteristics
Ed:0.585 "Exponent of diode characteristics
Vd0:0.8 "Forward voltage drop of diode
END

"LOSSY MODEL OF A THREE-PHASE HYSTERESIS
BAND PWM INVERTER

"DISCRETE SYSTEM INVL
INPUT lac lbc lcc la1 lb1 lc1 Vd We AT
OUTPUT Va Vb Vc I2 Naux Piinv
STATE A B C sl s Sla lav "Inverter Switching State
NEW NA NB NC Ns1 Ns NSla NSlav "A=1 -> T1 on, T4 off
STATE la lb lc
NEW Nla Nlb Nlc
TIME T "A=0 -> T4 on, T1 off
TSAMP Ts
Ts = T + AT

"DEFINE AUXILIAR STATE EQUATIONS FOR CURRENT
Nla = la1
Nlb = lb1
Nlc = lc1

210
" DEFINITION OF ERROR = REF-ACTUAL
Iae = Iac-ia
Ibe = Ibc-Ib
Ice = Icc-Ic

" LOGIC TO GENERATE PWM BY HB COMPARISON
NA = IF Iae>HB THEN 1 ELSE IF IAE<-HB THEN 0 ELSE A
NB = IF Ibe>HB THEN 1 ELSE IF IBE<-HB THEN 0 ELSE B
NC = IF Ice>HB THEN 1 ELSE IF ICE<-HB THEN 0 ELSE C

" GENERATION OF VOLTAGES (WITH RESPECT TO GROUND)
Vag = IF Ia>O THEN C1a ELSE C2a
C1a = IF NA>O THEN Vd-TdropA ELSE -DdropA
C2a = IF NA>O THEN Vd+DdropA ELSE TdropA
TdropA = ABS(ia)*Rt + V0
DdropA = Vd0 + Cd*ABS(ia)*Ed

Vbg = IF Ib>0 THEN C1b ELSE C2b
C1b = IF NB>O THEN Vd-TdropB ELSE -DdropB
C2b = IF NB>O THEN Vd+DdropB ELSE TdropB
TdropB = ABS(ib)*Rt + V0
DdropB = Vd0 + Cd*ABS(ib)*Ed

Vcg = IF Ic>0 THEN C1c ELSE C2c
C1c = IF NC>O THEN Vd-TdropC ELSE -DdropC
C2c = IF NC>O THEN Vd+DdropC ELSE TdropC
TdropC = ABS(ic)*Rt + V0
DdropC = Vd0 + Cd*ABS(ic)*Ed

" GENERATION OF PHASE VOLTAGES
Va = (2*Vag-Vbg-Vcg)/3
Vb = (2*Vbg-Vag-Vcg)/3
Vc = (2*Vcg-Vbg-Vag)/3

" CONDUCTION LOSSES COMPUTATION
Pea = IF NA THEN (Vd-Vag)*la ELSE -Vag*la
Pcb = IF NB THEN (Vd-Vbg)*lb ELSE -Vbg*ib
Pcc = IF NC THEN (Vd-Vcg)*lc ELSE -Vcg*lc
Pici = Pca+Pcb+Pcc

" CONTABILIZATION OF SWITCHING CYCLES/PERIOD
FOR UPPER TR OF LEG A. (Naux)
RES = SIGN(Iac) "LOGICAL SIGNAL (+1 OR -1)
DRES = DELAY(RES,AT) "DELAYED LOGICAL SIGNAL
PROD = RES*DRES "PULSE FOR RESET (-1 AT RESET INSTANT)
Nsl = IF PROD=0 THEN (IF A<NA OR A>NA THEN s1+1 ELSE s1) ELSE 0
Ns = IF PROD<0 THEN 2*s1 else s
Naux = s

" COMPUTE AVERAGE VALUE OF ABS(ia)
fs = We/π2
NSla = IF PROD>0 THEN Sla+ ABS(ia) ELSE 0 "Integral over half period.
Nlav = IF PROD>0 THEN lav ELSE (2*fs*AT*Sla) "Average over half cycle.

" COMPUTATION OF TRs TURN-ON SWITCHING LOSSES
Nif = (Naux/2)*lav*fs
Kon = 1.5*ton*(1-1.333*(t1/Ton)+0.5*(t1/Ton)+(t1/Ton))
Rton = Vd/(Nif*Kon+alpha)
Pton = Vd*Vd/Rton "Turn on losses.
Iton = Vd/Rton "Current component due to Pton

" COMPUTATION OF TRs TURN-OFF SWITCHING LOSSES
Koff = \( 1.5 \times t_{off} \times (1 - 1.333 \times (t_{1C} / t_{off}) + 0.5 \times (t_{1C} / t_{off}) \times (t_{1C} / t_{off})) \)
Rtoff = \( V_d / (N_i^* \times K_{off} + \alpha) \)
Ptoff = \( V_d \times V_d / R_{ttoff} \)
l_{toff} = \( V_d / R_{ttoff} \) "Current component due to Ptoff"

"COMPUTATION OF SNUBBER LOSSES"
Rsn = \( 2 / (3 \times N_{aux} \times f_s \times C_s + \alpha) \)
Ps = \( V_d \times V_d / R_{sn} \)
l_{sn} = \( V_d / R_{sn} \) "Current component due to Ps"

"COMPUTATION OF DC LINK CURRENT AND INV. OUTPUT POWER"
l2 = \( N_A \times l_a + N_B \times l_b + N_C \times l_c + (l_{on} + l_{toff} + l_{sn}) \)
P_{linv} = \( P_{c_{lin}} + P_{ton} + P_{toff} + P_{sn} \) "Total inverter losses."

"INITIAL VALUES"
A: 1 "Base drive states."
B: 1
c: 0
s: 1
s: 1
lav: 10

"PARAMETERS"
Data for POWEREX single darlington TR ks524503 (30 A/600 V)
toff: 0.8e-06 "Turn-off time of TR (tf)"
ton: 0.52e-06 "Turn-on time of TR (tr)"
Rt: 0.01 "Conduction resistance of TR"
Vt0: 0.7 "Saturated coletor-emitter of TR"
Vd0: 0.7 "Forward voltage drop of freewheeling diode"
Cd: 0.051 "Coefficient of VxA characteristic of FWD"
Ed: 0.671 "Exponent of VxA charac. of FWD"

Data related to snubbers
Cs: 0.0068E-06 "Snubber capacitor"
Ls: 0.2e-06 "Snubber inductance"
l1L: 0.11e-06
t1C: 0.382e-06

Other parameters
alpha: 1e-07
pi2 = 8 * atan(1)
HB: 1.5 "Histeresis band"
END

* INDUCTION MOTOR LOSSY D-Q MODEL IN SYNCHRONOUS FRAME *

* FEATURES INCLUDED IN THIS MODEL
  1) TEMPERATURE EFFECTS ON BOTH ROTOR AND STATOR RESISTANCES;
  2) SATURATION EFFECTS ON MAGNETIZING IMPEDANCE (Lm);
  3) SKIN EFFECT ON ROTOR RESISTANCE (HARMONIC FREQUENCY);
  4) STRAY LOSSES (BOTH FUND. AND HARMONICS) ARE INCLUDED;
  5) FRICTION AND WINDAGE LOSSES ARE CONSIDERED;
  6) CORE LOSSES ARE SEPARATELY COMPUTED FOR FUND. AND HARMONIC FREQUENCIES AND INTEGRATED IN THE MODEL.
* CONTINUOUS SYSTEM IMS3
INPUT va vb vc we Naux SINWT COSWT
OUTPUT ia ib ic wr Lm Rr po Pin

212
"-----DEFINE SOME USEFUL CONSTANTS AND VARIABLES

pi = atan(1)*4
p2 = pi*2
pb2s = p2*p2
sq2 = sqrt(2)
sqr3 = sqrt(3)

tl = k1*wr*wr
spd = wr*60/p2
we = p2*fc
f = we/p2
fca = max(1450,2500)
wc = p2*fc
dfc = (fca-fc)/talt

"-----COMPUTE MACHINE FLUXES

fqls = Lls*iqsl
fdls = Lls*idsl
fqm = Lm*iqm
fdm = Lm*idm
fqr = fqm + Llr*iqrl
fdr = fdm + Llr*idrl

"-----DEFINE CURRENTS AND AUXILIAR VARIABLES

iqs = iqsl + iqsr
iqr = -(iqrl + iqrr)
iqm = iqs + iqr - iqm

F1q = -vqs + Rl*iqs + we*fdls
F2q = (RbL2*iqsr*Leq2+RbL3*iqrr+we*fdm+we*fdr*Leq4+f2qc)/Leq1
F2qc = (RbL5+RbL6)*iqr-RbL4*iqrr
LF2q = Lls*F2q

ids = idsl + idsr
idr = -(idrl + idrr)

F1d = -vds + Rl*ids - we*fqls
F2d = (RbL2*idsr*Leq2+RbL3*idrr+we*fqm*Leq3+wsl*fqr*Leq4+f2dc)/Leq1
F2dc = (RbL5+RbL6)*idr-RbL4*idrr
LF2d = Lls*F2d

"-----STATE EQUATIONS FOR MACHINE DYNAMICS

diqsl = F2q

diqsr = -(F2q*leq3+Leq4+Rr*iqr)
diqm = (F2q*leq3+Leq4+Rr*iqr)
didsl = F2d

disdslr = -(F2d*leq3+Leq4+Rr*idr)
disd = -(F2d*leq3+Leq4+Rr*idr)

dw = (te-tl-b*Wr*Kfw+Wr*Wr)*pole/(j*2)"
"---VOLTAGE TRANSFORMATIONS

\[ v_{qss} = v_a \]
\[ v_{dss} = (v_c-v_b)\sqrt{3} \]
\[ V_{qs} = v_{qss}\cos \omega t - v_{dss}\sin \omega t \]
\[ V_{ds} = v_{qss}\sin \omega t + v_{dss}\cos \omega t \]

"---COMPUTATION OF DEVELOPED TORQUE

\[ t_e = -(f_{dr}\cdot i_{qr}-f_{qr}\cdot i_{dr})\cdot 0.75\cdot p \]

"---CURRENT TRANSFORMATIONS

\[ i_{qss} = \cos \omega t \cdot i_{qs} + \sin \omega t \cdot i_{ds} \]
\[ i_{dss} = -\sin \omega t \cdot i_{qs} + \cos \omega t \cdot i_{ds} \]
\[ i_a = i_{qss} \]
\[ i_b = -(0.5\cdot i_{qss}) - (0.5\cdot \sqrt{3} \cdot i_{dss}) \]
\[ i_c = -(0.5\cdot i_{qss}) + (0.5\cdot \sqrt{3} \cdot i_{dss}) \]

"---CALCULATE THE RMS CURRENT THROUGH Lm

\[ i_m = \sqrt{(i_{qm}\cdot i_{qm} + i_{dm}\cdot i_{dm})/2} \]

"---COMPUTE TIME DEPENDENT MODEL PARAMETERS

\[ r_s = r_{so}(1+A_1\cdot D_{temp}) \]
\[ r_r = r_{ro}(1+A_2\cdot D_{temp}) \]
\[ L_m = \text{if } i_m < i_{m0} \text{ then } L_{m0} \text{ else } L_{m0} - m \cdot (i_m - i_{m0}) \]
\[ L_r = L_m + L_{ir} \]

\[ w_{sl} = \omega - w_e \]
\[ s = w_{sl}/w_e \]
\[ R_m = p_b2s*f/((1+abs(s))\cdot K_h + (1+s_s)\cdot K_e\cdot f) \]
\[ R_{mc} = p_b2s*f_c / (2\cdot (K_h + K_e\cdot f_c)) \]
\[ R_{temp} = R_m \cdot (R_{mc} - R_m) \]

\[ L_{lm} = \text{if } R_{temp} > 0.18 \text{ then } \sqrt{R_m \cdot (R_{mc} - R_m)}/w_c \text{ else } 10^{-06} \]

\[ R_{sll} = R_{sb}\cdot f/(f/b)\cdot (1+gm\cdot f)/(1+gm\cdot fb) \]
\[ C_{fr} = (f_c/fcb)/(1+gm\cdot fcb)/(1+gm\cdot fc) \]
\[ R_{ssn} = R_{sb}\cdot C_{fr} \]
\[ R_{rc} = R_s\cdot K_r \cdot \sqrt{f_c} \]

\[ X_{lrc} = L_{lr}\cdot w_c \]
\[ X_{lrc2} = X_{lrc}\cdot X_{lrc} \]
\[ R_{rsn} = C_{fr}\cdot R_{rsb} \]
\[ R_{rle} = R_{rsnb} + (R_{rc} - R_r) \]
\[ X_{bR} = X_{lrc}/2 \]
\[ X_{bR2} = X_{lrc}/2 \]

\[ \text{SIG} = \text{if } X_{lrc} < R_{rsn} \text{ then } +1 \text{ else } -1 \]
\[ DLT = \max((X_{bR}\cdot X_{bR}) - 4\cdot X_{lrc2}, 0) \]
\[ R_{rsnb} = \text{if } DLT > 0 \text{ then } (X_{bR}\cdot \text{SIG}\cdot \sqrt{DLT})/2 \text{ else } X_{lrc}/2 \]

\[ \text{po} = t_{lw} \]
\[ \text{pin} = v_a\cdot i_a + v_b\cdot i_b + v_c\cdot i_c \]
\[ \text{Pl} = \text{pin-po} \]

"---ESTIMATE TEMPERATURE RISE

\[ D_{temp} = P_{fl} \cdot \text{theta} \]

"---DEFINE AUXILIAR CONSTANTS

\[ L_{eq4} = 1/(L_{rsn} + 1/L_r) \]

214
\[
\begin{align*}
\text{Leq3} &= \text{Leq4} + 1/\text{Lm} \\
\text{Leq2} &= \text{Leq3} + 1/\text{Llm} \\
\text{Leq1} &= (\text{Leq2} + 1/\text{Lssn})*\text{Li} + 1 \\
\text{Rbl2} &= \text{Rssn}/\text{Lssn} \\
\text{Rbl3} &= \text{Rm}/\text{Lm} \\
\text{Rbl4} &= \text{Rrsnp}/\text{Lrsn} \\
\text{Rbl5} &= \text{Rr}/\text{Llr} \\
\text{Rbl6} &= \text{Rr}/\text{Lrsn} \\
\text{Rl} &= \text{Rs} + \text{Rll} \\
\end{align*}
\]

"-----CONSTANT MACHINE PARAMETERS"

CLASS B 5 hp machine.

\text{rs0}=0.370 \quad \text{"Resistance values at 25C, 60 Hz} \\
\text{rt0}=0.436 \quad \text{"" " " dc} \\
\text{Rsh}=0.1384 \quad \text{"Base value of series fund. stray loss stator res.} \\
\text{Rsb}=164.46 \quad \text{"Base value of stator harm. stray loss res. (at fcb=5KHz)} \\
\text{Rrsb}=164.46 \quad \text{"Base value of rotor harm. stray loss res. (at fcb)} \\
\theta = 0.06319 \quad \text{"Thermal resistance} \\
\text{A1}=0.00385 \quad \text{"Temperature coefficient for copper 100} \\
\text{A2}=0.00389 \quad \text{"" " for aluminum 60} \\
\text{kh}=4.3811 \quad \text{"Hysteresis coefficient} \\
\text{ke}=0.0313 \quad \text{"Eddy current coefficient} \\
\text{gm}=0.00238 \quad \text{"Skin effect coef. for 3/8" deep bar} \\
\text{Kr}=0.1371 \quad \text{"Base frequency (1800 rpm)} \\
\text{fb}=60 \quad \text{"Linear value of magnetizing inductance} \\
\text{lm0}=0.06277 \quad \text{"Linear region breakpoint of Lm} \\
\text{im0}=3.4 \quad \text{"Saturation coefficient} \\
\text{m}=0.00366 \quad \text{"Leakage inductances} \\
\text{Lls}=0.00213 \quad \text{"Machine inertia} \\
\text{Llr}=0.00213 \quad \text{"Viscous damping coefficient} \\
\text{Lssn}=0.05*\text{Li} \quad \text{"Load constant (Rated = 0.3862e-03)} \\
\text{Lrsn}=0.05*\text{Llr} \quad \text{"Friction and windage coefficient} \\
\text{J}=17.5E-03 \quad \text{"Time constant for filters} \\
\text{b}=4.05e-04 \quad \text{"Base carrier frequency} \\
\text{pole}=4 \quad \text{"1800 rpm --> 377 rad/s.} \\
\text{kl} = \text{factor} \times 0.5862e-03 \quad \text{"Load constant (Rated = 0.3862e-03)} \\
factor = \text{if } t<3 \text{ then 0.4 else 2 !} \\
\text{Kfw}=2.248e-06 \quad \text{"Friction and windage coefficient} \\
\text{Ta1}=0.0004 \quad \text{"Time constant for filters} \\
\text{Ta2}=0.010 \quad \text{"Base carrier frequency} \\
\text{fcb}=5000 \quad \text{"1800 rpm --> 377 rad/s.} \\
\text{.jd}=9.786 \quad \text{"1800 rpm --> 377 rad/s.} \\
\text{idm}=9.7 \quad \text{"1800 rpm --> 377 rad/s.} \\
\text{iqs}=1.6 \quad \text{"1800 rpm --> 377 rad/s.} \\
\text{iqr}=1.6 \quad \text{"1800 rpm --> 377 rad/s.} \\
\text{w}=188.5 \quad \text{"1800 rpm --> 377 rad/s.} \\
\text{fc}=5000 \quad \text{"1800 rpm --> 377 rad/s.} \\
\text{END}
DISCRETE SPEED PI CONTROLLER

NOTES:
1) ANTI-WINDUP IS USED, AS WELL AS SATURATION FUNCTION TO LIMIT VALUE OF CONTROL VARIABLE iqec;
2) GAINS ARE SELECTED FOR A 5 HP MACHINE;

DISCRETE SYSTEM WCTRL
INPUT Wr
OUTPUT E iqecp
STATE X
NEW NX
TIME T
TSAMP TS
TS=T+AT

COMPUTE SPEED ERROR E AND INITIAL CONTROL SIGNAL V
E = 0.1047197*Wref-Wr
V = Kp*E+X
NX = X+Ki*E*AT+kwup*(iqecp-V)

COMPUTE FINAL CONTROL SIGNAL iqec
iqecp = IF ABS(V)<IQM THEN V ELSE SIGN(V)*IQM

PARAMETERS
Wref:2200
IQM:26.7
Kp:1.25
Ki:25
Kwup:0.02
AT:1E-03

INITIAL VALUES
X:5
END

FUZZY EFFICIENCY OPTIMIZATION ROUTINE

DISCRETE SYSTEM FEOPT
INPUT Dw, Wr, Pi, DT, T, lest
OUTPUT DIDEC RT, FSSA
STATE Po, DPi, CR, DC, Did, FSS
NEW NPo, NPi, NCR, NDC, NDid, NFSS
TIME T
TSAMP TS
TS=T+DT

CHECK FOR STEADY STATE (SS) AND DEFINE FLAGS
E1 = DELAY(Dw, T, W)
E2 = DELAY(E1, T, W)
SUM = ABS(Dw)+ABS(E1)+ABS(E2)
FSSS = IF FSS THEN TOEX ELSE TOST
TOEX = IF SUM<TOLEX THEN 1 ELSE 0
TOST = IF SUM<TOLST THEN 1 ELSE 0
FSSA = FSS
NCR = IF FSS<1 THEN 0 ELSE IF RT THEN CR+1 ELSE CR
NDC = IF DC < 0.99 THEN COUNT ELSE DC-1
RT = IF DC < 0.99 THEN 1 ELSE 0

Maximum iqec (50% higher Is).
Proportional gain
Integral gain
Anti-windup coefficient.
Speed sampling time.
"---DEFINE BASE VALUE FOR Dids (GID), IMPOSING Min. AND Max.
GIDA = C1*Wr/Wrated + C2*Ti/Trated + C3
"PRELIMINARY VALUE.
GID = IF ABS(GIDA) < TOL4 THEN SIGN(GIDA)*TOL4 ELSE GIDC
GIDC = IF ABS(GIDA) > TOL5 THEN SIGN(GIDA)*TOL5 ELSE GIDC
"---COMPUTE POWER GAIN (1/Pbase) AND SCALED DELTA Pi (DPi)
GPbA = -.05*Wr/Wrated + 0.0625
GPb = MIN(GPbmax, MAX(GPbmin, GPbA))

NPo = IF RT THEN Pi ELSE Po
NDPi = IF RT THEN (Pi-Po)*GPb ELSE DPi

"---EVALUATION OF Cid (CHANGE IN Dids)
"(A FUZZY RELATION IS USED TO EVALUATE Cid)
"---PRELIMINARY COMPUTATION OF MEMBERSHIP VALUES
A1 = MIN(l,-(NDPi-P2)/(P1-P2))
A2 = (-NDPi/P3)/(P2-P3)
A3 = (-NDPi/P3)
A5 = NDPi/P3
A6 = (NDPi-P3)/(P2-P3)
A7 = MIN(l, (NDPi-P3)/(P1-P2))

"---GET INTERVAL INDEX FOR DPi
J = IF NDPi< -1 THEN 1 ELSE IF NDPi< -2 THEN 2 ELSE J1
J1 = IF NDPi< -3 THEN 3 ELSE IF NDPi< 0 THEN 4 ELSE J2
J2 = IF NDPi< -5 THEN 5 ELSE IF NDPi< -2 THEN 6 ELSE J3
J3 = IF NDPi < 1 THEN 7 ELSE 8

"---EVALUATION OF DEGREES OF MEMBERSHIP FOR DPi
MP1 = IF J<3 THEN A1 ELSE IF J<4 THEN A2 ELSE IF J<5 THEN A3 ELSE T1
T1 = IF J<6 THEN (1-A5) ELSE IF J<7 THEN (1-A6) ELSE (1-A7)
MP2 = 1-MP1

"---COMPUTES DEGREE OF MEMBERSHIP FOR LAST Dids
M11 = IF Dids<-L3 THEN 0 ELSE MIN(l,((Dids+L3)/(L2+L3)))
M12 = IF Dids>L3 THEN 0 ELSE MIN(l,((-Dids+L3)/(L2+L3)))

"---PERFORM RULE BASE EVALUATION
MRA = MIN(MP1,M11)
MRB = MIN(MP1,M12)
MRC = MIN(MP2,M11)
MRD = MIN(MP2,M12)

"---GET CONTROL SIGNAL FOR EACH RELEVANT FUZZY RULE
DIA = IF J<3 THEN 11 ELSE IF J<4 THEN 12 ELSE IF J<5 THEN 13 ELSE T2
T2 = IF J<6 THEN 0.0 ELSE -13
DID = -DIA
"FOR THIS SPECIFIC RB
DIC = IF J<3 THEN 12 ELSE IF J<4 THEN 13 ELSE IF J<5 THEN 0.0 ELSE T3
T3 = IF J<7 THEN -13 ELSE -12
DID = -DIC
"FOR THIS SPECIFIC RB

"---EVALUATE Cid USING HEIGHT DEFUZZIFICATION METHOD
FV = (MRA*DIA+MRB*DIB+MRC*DIC+MRD*DID)/(MRA+MRB+MRC+MRD)

Cid = IF NCR<1 THEN 0 ELSE IF NCR<2 THEN -1.0 ELSE FV

"---CALCULATION OF NEW Dids
NDids = IF RT THEN Cid ELSE Dids
DAUX = NDIDS*GID
DIDEC = IF ABS(DAUX)<TOL2 THEN SIGN(DAUX)*TOL2+FSS*RT ELSE DAUX

"---PARAMETERS
*DT:5E-06
*"SAMPLING INTERVAL OF THE DISCRETE SYSTEMS

217
TW:1E-03  "PERIOD FOR TEST OF SS CONDITION
TOLST:1  "TOL. TO START STEADY STATE COND. (IN RAD/S)
TOLEX:3  "TOL. TO EXIT STEADY STATE COND.
C1:1.083  "COEFFICIENTS FOR GID COMUTATION
C2:3.047
C3:1.496
GPbmax:0.05  "MAX AND MIN VALUES FOR POWER GAIN
GPbmin:0.0125
Trated:20.83  "RATED MOTOR TORQUE
Wrated:188.5  "RATED Wr IN MECH. RAD/S
TOL2:0.1  "MIN. CHANGE FOR Dids (A)
TOL4:0.5  "MIN. VALUE FOR GID (AMPS)
TOL5:1.5  "MAX. VALUE FOR GID (AMPS)
COUNT=0.5/DT  "DOWN COUNTER INITIAL VALUE (FOR Teff=0.5 S)

"-----PARAMETERS FOR FUZZY SETS
P1:1  "CONSTANTS USED IN DEFINITION OF FUZZY SETS FOR DPi
P2:0.5
P3:0.3
I1:1.0  "CONSTANTS USED IN DEFINITION OF FUZZY SETS FOR Dids
I2:0.7
I3:0.40
L1:1.0  "CONSTANT USED IN DEFINITION OF FUZZY SETS FOR LDids
L2:0.1
L3:0.001

"----- INITIAL VALUES
DC:100000  "DOWN COUNTER
END

*-------------------------------------------------------------
* " FEED FORWARD TORQUE COMPENSATION ROUTINE
*-------------------------------------------------------------
DISCRETE SYSTEM FFTC

INPUT iqecp Dids RT FSS AT
OUTPUT Wsl ideca iqec TLest
STATE idec fdr e iqst iqsc fdr
NEW Nidec Nfdr Niqst Niqsc Nfdr
TIME T
TSAMP TS
TS=T+AT

"-----DEFINES idec, Lm AND STEADY-STATE ESTIMATED D-AXIS FLUX
aidec = idec+Dids
Lm = Lm0-Mbs2*(aidec-idec0)
Nfdr = if RT then lm*aidec else fdr

"-----COMPUTE diqs(DELTA iqst), TEST FOR ISmax
" AND COMPUTE iqst (STAIR-CASE iqst)
diqsa = FSS*(fdr-e-Nfdr)*iqec/Nfdr
aiqec = iqec+iqst+diqsa
IP2 = aidec*aidec+aiqec*aiqec  "SQUARE OF NEW STATOR CURRENT VECTOR
diqs = if IP2 < IMAX then diqsa else 0  "SQUARE OF NEW STATOR CURRENT VECTOR
iqst = if RT then iqst+diqs else iqst  "Accumulated diqs.
"-----COMPUTE iqsc (compensating signal) and iqec (iqs*)
Niqsc = iqsc + AT*(iqsc-iqsc) / T
Nidec = if FSS then idec else if IP2<IMAX and RT then aidec else idec
ideca = Nidec "Ids*(k).
iqec = min((iqec+ipqsc),sqrt(IMAX-idec•idec)) "lqs*(k).

"-----EVALUATE SLIP FREQUENCY Wsl, USING ESTIMATED FLUX
Tarl=Lr/Rr
fdr = if FSS then fdre else fdrr
Nfdr = fdr + AT*(fdr-fdr) / Talr
Wsl=(Lm*Rr*iqec)/(Lr*fdr)

"-----ESTIMATE MACHINE TORQUE
TLtest = Kt*iqec*fdr

"-----PARAMETERS
IMAX = if FSS then IMAX1 else IMAX2
Lr=Lm+Llr
Llr=0.002 13
Lr:0.478
Lm=0.06277
Mbs=2.588e-03
Mbs:2.88e-03
idec=4.8 "
idecc=9.786
idec=9.786
fdr=0.4833
fdr=0.4833
fdre=0.4833
fdre=0.4833
IMAX1:360
IMAX2:808
Kt=2.655 "Torque constant.

"-----INITIAL CONDITIONS
fdr=0.4833
fdr=0.4833
fdr=0.4833
fdr=0.4833
END

" "REFERENCE CURRENT GENERATOR AND VECTOR ROTATOR

" CONTINUOUS SYSTEM VECROT
INPUT iqec idec Wr Wsl
OUTPUT iac ibc icc We coswt sinwt
STATE TH
DER DTH
" TIME T
""-----GENERATE UNIT VECTORS
We=(pole/2)*Wr+Wsl
Db=We
ten=MODX(th,p2)
COSWT=COS(thec)
SINWT=SIN(thec)

""-----SYNCHRONOUS TO STATIONARY D-Q REF. FRAME
iqsc=iqec*COSWT+idec*SINWT
idsr=idec*COSWT-iqec*SINWT

""-----STATIONARY TO ABC REF FRAME
iac=iqsc
ibc=(idsr*SQR(3)+iqsc)*0.5
icc=(iac+ibc)

219
**DEFINE CONSTANTS**

\[ p2 = 8 \cdot \text{ATAN}(1) \]

pole 4

END

**COMPUTATION OF AVERAGE LOSSES, INPUT AND OUTPUT POWERS AND EFFICIENCY**

CONTINUOUS SYSTEM ECALC

INPUT Pinv Pild Pin Po Pior Piir

OUTPUT Piav

STATE X1 X2 X3 X4 X5 X6

DER DX1 DX2 DX3 DX4 DX5 DX6

TIME T

**COMPUTE AVERAGE (FILTERED) VALUES OF LOSSES AND POWERS**

\[ DX1 = X2 \]

\[ DX2 = A21 \cdot X1 + A22 \cdot X2 + Piir \]

Pair = \(-A21 \cdot X1\)

\[ DX3 = X4 \]

\[ DX4 = A21 \cdot X3 + A22 \cdot X4 + Pior \]

Paor = \(-A21 \cdot X3\)

\[ DX5 = X6 \]

\[ DX6 = A21 \cdot X5 + A22 \cdot X6 + Pin \]

Paim = \(-A21 \cdot X5\)

Piav = Paor

Nsys = 100 \cdot Po/Pair

Ndc = 100 \cdot Po/Paor

Nim = 100 \cdot Po/Paim

**PARAMETERS AND INITIAL VALUES.**

Tal2: 40e-03

A21 = \(-1/(Tal2 * Tal2)\)

A22 = \(-2/Tal2\)

X1: 2.0368

X3: 2.0368

X5: 2.0368

END

**CONNECTING SYSTEM FOR VECTOR CONTROL**

CONNECTING SYSTEM CIVCS

\[ \text{VAS}[RCFL] = \text{VAS}[AC] \]

\[ \text{VBS}[RCFL] = \text{VBS}[AC] \]

\[ \text{VCS}[RCFL] = \text{VCS}[AC] \]

\[ \text{VA}[IM3] = \text{VA}[INVL] \]

\[ \text{VB}[IM3] = \text{VB}[INVL] \]

\[ \text{VC}[IM3] = \text{VC}[INVL] \]

\[ \text{Vd}[INVL] = \text{Vd}[RCFL] \]

\[ \text{IA}[INVL] = \text{IA}[IM3] \]

\[ \text{IB}[INVL] = \text{IB}[IM3] \]

\[ \text{IC}[INVL] = \text{IC}[IM3] \]
IAC(INVL)=IAC[VECROT]
IBC(INVL)=IBC[VECROT]
ICC(INVL)=ICC[VECROT]
IQECP(FFTC)=IQECP[WCTRL]
IQEC(VECROT)=IQEC[FFTC]
IDEC(VECROT)=IDEC[FFTC]
I2(RCFL)=I2(INVL)
DIDS(FFTC)=DIDS[FEOPT]

We(IMS3)=We[VECROT]
Wr[WCTRL]=Wr[IMS3]
Ws[WCTRL]=Ws[FFTC]
Wr[VECROT]=Wr[IMS3]
We(INVL)=We[VECROT]
Wr[FEOPT]=Wr[IMS3]
DWr[FEOPT]=DWr[WCTRL]

Nsaux[IMS3]=Nsaux[INVL]
RT(FFTC)=RT[FEOPT]
FSS(FFTC)=FSS[A[FEOPT]
TLess[FEOPT]=TLess[FFTC]

Pi[FEOPT]=Piav[ECALC]
Pftr[ECALC]=Pftr[RCFL]
Pinf[ECALC]=Pinf[RCFL]
Pin[ECALC]=Pin[IMS3]
P0[ECALC]=P0[IMS3]
PInv[ECALC]=PInv[INVL]
PInv[ECALC]=PInv[RCFL]

SINWT(IMS3)=SINWT[VECROT]
COSWT(IMS3)=COSWT[VECROT]

AT(INVL)=1E-06
AT(FFTC)=1E-06
DT(FEOPT)=1E-06
END

" " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " " 

SIMU -0.30 4 0.5E-06 / fig9a 0.005
### A.3 Slip Gain Tuning Programs

```
!---------------------------------------------
" IDEAL HYSTERESIS BAND PWM INVERTER
!---------------------------------------------
DISCRETE SYSTEM INV
INPUT ia ib ic iae ibc icc
OUTPUT va vb vc
STATE A B C
NEW NA NB NC
TIME T
TSAMP TS
TS=T+AT

"......GET CURRENT ERRORS
iae=iac-ia
ibe=ibc-ib
icc=icc-ic

"......CALCULATE NEW STATES OF THE SWITCHES
NA=IF iae>L THEN 1 ELSE IF iae<-L THEN 0 ELSE A
NB=IF ibe>L THEN 1 ELSE IF ibe<-L THEN 0 ELSE B
NC=IF ice>L THEN 1 ELSE IF ice<-L THEN 0 ELSE C
Vd3=Vd/3
va=(2*NA-NB-NC)*Vd3
vb=(2*NB-NA-NC)*Vd3
vc=(2*NC-NB-NA)*Vd3

"-------- PARAMETERS AND INITIAL STATES.
AT:2E-06 "Sampling interval.
L:1.2 "Hysteresys band.
Vd:310 "Assumed dc link voltage
A:1
B:1
C:0
END
```

```
" SYNCHRONOUS REFERENCE FRAME D-Q MODEL OF
" AN INDUCTION MACHINE IN TERMS OF CURRENT
"  CONTINUOUS SYSTEM machine
INPUT va vb vc we coswt sinwt
OUTPUT ia ib ic wr Vds Vqs
STATE iqqs ids iqr idr w x
DER diqs dids diqr didr dw dx
TIME t
```
pi = stan(1)*4
p2 = pi^2
sqr3 = sqrt(3)
spd = w/pole/pi^60

"----- STATE EQUATIONS
diqs = (lr*(vqs-rs1*iqs)-fse*ids+lm*r*iqr-lrm*w*idr)*k1
dids = (fse*iqs+lr*(vds-rs1*ids)+lrm*w*iqr+lm*r*iqr)*k1
diqr = (lm*(rs1*iqs-vqs)+w*lam*ids-iqr*r*ls+fre*ids)*k1
didr = (-w*lam*iqs+lm*(ids*rs1-vds)-fre*iqr-r*ls+ids)*k1
dw = (te-tl-b*w^2/pole)*j*pole/2

tl=kll*wr*wr
wr=w^2/pole

" SPEED IN MECH. RAD/S

"----- VOLTAGE TRANSFORMATIONS
vqss = va
vdss = (v-cb)*sqrt3
vqs = vqss*coswt-vdss*sinwt
vds = vqss*sinwt+vdss*coswt
delw = we-w
fse = lsr*we-delw*lmm
fre = we*lmm-delw*lsr
te = 0.75*pole*lm*(idr*iqs-iqr*ids)

" Stationary frame.
" Synchronous frame.
" Slip frequency.
" Auxiliary expressions.
" Developed torque.

"----- NEW VARIABLES BEING INTRODUCED
DX = (Te - X)^TAL
Tefpu = X*TB

fdr = Lr*idr+Lm*ids
fqr = Lr*iqr+Lm*ids
frpu = sqrt(fdr*fdr+fqr*fqr)/frb

" Per-unit filtered torque.
" D and Q axis rotor flux.
" Per-unit rotor flux.

"----- GET MACHINE PARAMETERS
ls = (xm+xs)/wb
lr = (xm+xr)/wb
lm = xm/wb
lrn = lr*lm
lmm = lm*lr
lsr = ls*lr
lmm = lm*lm
k1 = lsr-lmm

" Stationary D-Q frame.
" ABC frame.

"----- CURRENT TRANSFORMATIONS
iqss = iqs*coswt+ids*sinwt
ids = ids*coswt-iqs*sinwt
ia = iqs
ib = -(0.5*iqss)-(0.5*sqr3*ids)
ic = -(0.5*iqss)+(0.5*sqr3*ids)

"----- PARAMETERS: FOR 5 HP IM OF AC 200 SYSTEM
wb:377
rs1:0.177
rr:0.099
xs:0.8068
xr:0.5127
xm:12.8105
j:6.58E-03
b:6.58E-04
pole:4

" Rated frequency (rad/sec.)
" Stator and rotor resistances.
" Stator and rotor reactances.
" Magnetizing inductance.
" Machine inertia
" Viscous damping coefficient
At rated speed and torque, \( k_{ll} = 3.2027 \times 10^{-4} \)

\( k_{factor} \): Initial value at rated flux.

\( \text{lcfactor} \): Filter time constant.

\( \text{lds}: 9.216 \)

\( \text{TAL}: 30 \times 10^{-6} \)

\( \text{frb}: 0.3132 \)

\( \text{ Tb}: 17 \)

END

"FUZZY SLIP GAIN TUNER"

"NOTE: USE E AND CE AS INPUT VARIABLES"

"DISCRETE SYSTEM FSOT"

INPUT DQ DVd Wr Iqec

OUTPUT Kslip

"Kslip = slip gain"

STATE Ks Err X1 X2
NEW NKs NErr NX1 NX2
TIME T
TSAMP TS
TS = T + AT

"FUZZY CONTROL 1 - Kf COMPUTATION"

\( MH_{Wr} = \min(1, \text{ABS}(W_{rWb})) \)

\( ML_{Wr} = 1 - MH_{Wr} \)

\( I_{qecpu} = \frac{I_{qec}}{I_{qerc\text{rated}}} \)

\( MH_{Iq} = \min(1, \text{ABS}(I_{qecpu})) \)

\( ML_{Iq} = 1 - MH_{Iq} \)

"PERFORM RULE BASE EVALUATION - FUZZY CONTROLLER 1"

\( MR_{AI} = \min(MH_{Wr}, MH_{Iq}) \)

\( MR_{BI} = \min(ML_{Wr}, MH_{Iq}) \)

\( MR_{CI} = \min(MH_{Wr}, ML_{Iq}) \)

\( MR_{DI} = \min(ML_{Wr}, ML_{Iq}) \)

"COMPUTE VALUE OF Kf, USING HEIGHT DEFUZZ. METHOD"

\( \text{PROD} = MR_{AI} \times K_{f1} + MR_{BI} \times K_{f2} + MR_{CI} \times K_{f3} + MR_{DI} \times K_{f4} \)

\( \text{SUM} = MR_{AI} + MR_{BI} + MR_{CI} + MR_{DI} \)

\( K_f = \frac{\text{PROD}}{\text{SUM}} \)

"COMPUTE Error AS A COMBINATION OF DQ AND DVd"

\( \text{NErr} = -1 \times (K_{f} \times D_{Q} + (1 - K_{f}) \times D_{Vd}) \)

\( \text{AUX} = \frac{\text{NErr}}{G_E} \)  

\( E = \text{IF} \ \text{AUX} < e_1 \ \text{THEN} \ e_1 \ \text{ELSE} \ \text{IF} \ \text{AUX} > e_6 \ \text{THEN} \ e_6 \ \text{ELSE} \ \text{AUX} \)  

"COMPUTE CHANGE IN ERROR (ABSOLUTE VALUE)"

\( \text{DE} = (\text{NErr} - \text{Err}) \times G_C \)

\( \text{NX1} = \text{X1} + (D_{E} - X_{1}) \times A_{TAL} \)

\( \text{NX2} = \text{X2} + (X_{1} - X_{2}) \times A_{TAL} \)

\( \text{CE} = \min(c_3, \text{ABS}(X2)) \)

"DEFINE INTERVAL INDEX (I) AND D.M. FOR CE"

\( I = \text{IF} \ \text{CE} < c_1 \ \text{THEN} \ 3 \ \text{ELSE} \ \text{IF} \ \text{CE} < c_2 \ \text{THEN} \ 2 \ \text{ELSE} \ 1 \)

\( c_1 = \text{FUNC}(4, I) \)  

\( c_1 = \text{FUNC}(5, I) \)  

\( \text{MC1} = (c_1 - \text{CE}) / (c_1 - c_1) \)

\( \text{MC2} = 1 - \text{MC1} \)
GET INTERVAL INDEX FOR E

- GET INTERVAL INDEX FOR E

J = IF E < e2 THEN 1 ELSE IF E < e3 THEN 2 ELSE J1
J1 = IF E < 0 THEN 3 ELSE IF E < e4 THEN 4 ELSE J2
J2 = IF E < e5 THEN 5 ELSE 6

DEFINE M.F. FOR E

ei = FUNC(2,J)
MEL = Func(3,J)

MEL = (ei - E) / (ei - ei)
ME2 = 1 - MEL

PERFORM RULE BASE EVALUATION - FUZZY CONTROL 2

MRA = MIN(MC2,ME1)
MRB = MIN(MC2,ME2)
MRC = MIN(MC1,ME1)
MRD = MIN(MC1,ME2)

GET Delta Ks VALUES FROM FUNCTION

II = (1-1) • 7 + J

DKA = Func(IJJ)
DKB = Func(IJJ+1)
DKC = Func(IJJ+7)
DKD = Func(IJJ+8)

EVALUATE DKs USING HEIGHT DEFUZZIFICATION METHOD

SUM = MRA+MRB+MRC+MRD
DKs = (MRA•DKA+MRB•DKB+MRC•DKC+MRD•DKD)/SUM

GET NEW VALUE FOR DELTA Ks

NKs = IF T < 0 THEN Ks ELSE Ks + DKs•GKs

PARAMETERS FOR 5 HP MACHINE

Ls: 36.12E-03
Lsig: 3.45E-03
Rs: 0.177
Wb: 230.383

ACCOUNTED 2200 RPM

PARAMETERS FOR FUZZY CONTROLLER

e1: -0.6
e2: -0.3
e3: -0.06
e4: 0.1
e5: 0.5
e6: 1.0
c1: 0.2
c2: 0.6
c3: 1

KF1: 0.8
KF2: 0.9
KF3: 0.98

OGE: 0.5
GC: 200
GKs: 2.5e-03

INITIAL VALUES

Ks: 0.608
"* COMPUTATION OF FUNDAMENTAL DELTA Q, DELTA Vd
* CONTINUOUS SYSTEM ECOMP
INPUT  We  Vds  Vqs  Idec  Iqec
OUTPUT  DQE  DVdE
STATE  X1  X2  X3
DER  DX1  DX2  DX3
TIME  T

"------COMPUTE FILTERED VALUES
DX1 = (Vds - X1) / TAL  "X1 = Vds filtered
DX2 = (Vqs - X2) / TAL  "X2 = Vqs filtered
DX3 = (X1 - X3) / TAL  "X3 = 2nd order filtered Vds

QEC = We * (Ls * Idec * Idec + Lsig * Iqec * Iqec)  "Reference from command signals
Qb = 0.5 + QEC  "Base value

" ------- COMPUTE DELTA REACTIVE POWER
DQE = (QEC - QE) / Qb  "Delta Q

" ------- COMPUTE DELTA Vd
VdEC = rs * Idec - We * Lsig * Iqec  "Command Vds
Vb = 0.1 + ABS(We) * Lsig * ABS(Iqec)  "Base Vds
DVdE = -(VdEC - X3) / Vb  "Delta Vds.

"------PARAMETERS
Tal: 1E-03
Ls: 34.877E-03
Lsig: 3.2975E-03
Rs: 0.177

CONNECTING SYSTEM FOR SLIP GAIN TUNING
CONNECTING SYSTEM CMACH
VA[MACHINE] = VA[INV]
VB[MACHINE] = VB[INV]
VC[MACHINE] = VC[INV]
Vds[ecalc] = Vds[MACHINE]
Vqs[ecalc] = Vqs[MACHINE]
Ia[INV] = Ia[MACHINE]
Ib[INV] = Ib[MACHINE]
Ic[INV] = Ic[MACHINE]
Iac[INV] = Iac[VECROT]
Ibc[INV] = Ibc[VECROT]
Icc[INV] = Icc[VECROT]
Iqec[VECROT] = Iqecp[WCTRL]
Iqec[FSGT] = Iqecp[WCTRL]
Idec[VECROT] = Idec[WCTRL]
Iqec[ECALC] = Iqecp[WCTRL]
IDEC[ECALC]=IDEC[WCTRL]

= 
We[MACHINE]=We[VECROT]
We[ECALC]=We[VECROT]
Wr[WCTRL]=Wr[VECROT]
Wr[VECROT]=Wr[MACHINE]
Wr[FSGT] = Wr[MACHINE]
Ws[VECROT]=Ws[WCTRL]

= 
SINWT[MACHINE]=SINWT[VECROT]
COSWT[MACHINE]=COSWT[VECROT]

DQ[FSGT] = DQ[ECALC]
DVd[FSGT] = DVd[ECALC]
Ks[WCTRL] = Ks[FSGT]

END

---------------------------------------------
MACRO FOR SLIP GAIN TUNING

MACRO MMACH
SYST INV MACHINE WCTRL VECROT ECOMP FSgt CMACH
STORE iqecpu Tefpu frpu spd
PLOT fct

error 0.0001
IMPORT MYFUNC < MYFUNC /2
par AT[INV]:-5*E-06
split 3 2

INIT Ks:0.304
par wref:1100

par kfactor:4.0
par GC:500
par GKS:0.3e-03

AREA 1 1
axes h -0.5 2 v 0 2
TEXT 'Ks/Kso'
SIMU -0.50 2 /fig1B 0.001 //MYFUNC

area 1 1
ashow 0 2 iqecpu
TEXT 'iqs*(pu)' 
area 2 1
ashow 0 2 tefpu
text 'Te (pu)'
area 3 1
ashow 0 2 frpu
text 'Fr (pu)'

END
APPENDIX B

TMS320C25 ASSEMBLY PROGRAMS LISTING

B.1 Efficiency Optimization and Vector Control Programs

; INDIRECT VECTOR CONTROL WITH FUZZY LOGIC
; EFFICIENCY OPTIMIZATION PROGRAM

GILBERTO C. D. SOUSA
UNIVERSITY OF TENNESSEE, KNOXVILLE.

; THIS PROGRAM INCLUDES:
; 1 - VECTOR CONTROL FUNCTIONS
; 2 - SPEED COMPUTATION ROUTINE
; 3 - PI SPEED CONTROL
; 4 - ADC ISR FOR TWO CHANNELS
; 5 - DAC ISR
; 6 - DC LINK POWER CALCULATION
; 7 - SLIP GAIN COMPUTATION FOR VARIABLE FLUX
; 8 - FEED-FORWARD TORQUE COMPENSATION
; 9 - FUZZY EFFICIENCY OPTIMIZATION CONTROLLER
; 10 - TRANSITION CONTROL

; NOTE: USING DATA FROM 5 hp CLASS B MACHINE.

; I/O PORT ADDRESS

MCNT: EQU 0 ; Encoder pulse count.
TCNT: EQU 1 ; Clock pulse count.
INSTU: EQU 2 ; Inverter status READ.
INCOM: EQU 2 ; Inverter command WRITE.
DATA: EQU 8 ; Buffer data port.
ADDR: EQU 9 ; Buffer address port.
READHOST: EQU 10 ; Host read port.
REFRESH: EQU 11 ; Enable buffer refresh.
DAC: EQU 13 ; DAC latch.
HOSTSTAT: EQU 14 ; Host status.
WRITEHOST: EQU 15 ; Host write port.

; CONSTANTS

PRDC: EQU 2000 ; Main sampling time (=0.2 msec).
Idrated: EQU 12826 ; Rated Ids = 9.786 A (= 12826).
SLIPGAIN: EQU 810 ; Rated slip gain.
NUMKSLC: EQU 179 ; Numerator of Sg (=LmRr/Lr). 1 PU = 179
DT: EQU 4000 ; Delta t for theta calculation.
TBL90: EQU 512 ; 90 degree offset.
SSAMPC: EQU 5 ; Speed sampling time counter. (=1msec.)
STMAXC: EQU 25 ;Max. speed sampling time. (5 ms)
DELWRCC: EQU 32767-100 ;Max speed variation in 1 ms.

; ----------- MEMORY MAPPED REGISTERS (ALREADY DEFINED IN RESMON)
;DRR: EQU 0 ;A/D input data register
;TIM: EQU 2 ;Timer.
;PRD: EQU 3 ;Period register.
;IMR: EQU 4 ;Interrupt mask register.
;GREG: EQU 5 ;Global memory allocation register.

; DEFINE CONSTANTS FOR PI CONTROLLER
K1C: EQU 3929 ;3935 ;For kp=0.4 and ki=0.8: K1C=3935
K2C: EQU -3920 ;K2C=-3927
IQMV: EQU 322 ;|lq*| limit / 64 = 322

; DEFINE CONSTANTS FOR FIRST ORDER FILTER (FOF)
; FOR SLIP GAIN COMPUTATION AND FFTC.
; CURRENTLY USING Tr = 0.10778 s.
A1C: EQU 32465
B1C: EQU 504
C1C: EQU 19570
D1C: EQU 151

; DEFINE CONSTANTS FOR FOF OF lqs*, for FFTC only!
A2C: EQU 32442
B2C: EQU 503
C2C: EQU 21127
D2C: EQU 163

; DEFINE CONSTANTS FOR FOF OF Pdc.
A3C: EQU 32702
B3C: EQU 307
C3C: EQU 6983
D3C: EQU 32

; CONSTANTS FOR FFTC ROUTINE.
Idc0: EQU 6302 ;Linear region breakpoint.
Mbs2c: EQU 21200 ;Coefficient for Lm computation (1.216e-03).
Lm0c: EQU 20568 ;Linear value of Lm (0.06277 H).
ldmaxc: EQU 17038 ;ldmax = 15729 (12A).
ldminc: EQU 18350 ;ldmin = 3932 (3A).
IMAX: EQU 18830 ;Square of max. stator cur. (Idmax = 18.95^2).

; CONSTANTS FOR TRANSITION CONTROL.
TOLex: EQU 300 ;Limit to exit eff. opt. mode.
TOLst: EQU 100 ;Limit to start eff. opt. mode.

; DEFINE CONSTANTS FOR FEOP ROUTINE.
I1: EQU 32767 ;Definition of fuzzy sets for DId (Delta)
I2: EQU 22938 ; I1*
I3: EQU 13107
L3: EQU 328 ;Def. of fuzzy sets for LDId (Last DId).
P1: EQU 2154 ;Def. of fuzzy sets for DP (Delta Pi).
P2: EQU 1077
P3: EQU 646

DPimaxc: EQU 32767-100 ;Used to clamp (Pi(k)-Pi(k-1) to 100
L2limc: EQU 32767-3277 ;Clamp LDId to 3277.
CGIC: EQU 2955 ;C1, C2 and C3 coef. for GID comp.
CG2C: EQU -31827
CG3C: EQU 1961
G1C: EQU -3438 ; Coefficients for GPb computation.
G2C: EQU 5408
GIDMAXC: EQU 32767-1966 ; Const. for max and min GID values.
GIDMNC: EQU 16580 ; == (32768+393)\( \frac{1}{2} \), min GID = 393.
GPBMAXC: EQU 32767-6760 ; Const. for max and min GPb comp.
GPBMNC: EQU 18356 ; == (32768+3943)\( \frac{1}{2} \), min GPb=3943.
Dldminc: EQU 131 ; Min. value for Dlde (==0.1 A).
RP12C: EQU 7606 ; \frac{1}{(P1-P2)}.
RP23C: EQU 19007 ; \frac{1}{(P2-P3)}.
RP3C: EQU 12681 ; \frac{1}{P3}.
RL23C: EQU 9089 ; \frac{1}{(L2+L3)}.

; -----------------------------
DEFINE DATA RAM
; -----------------------------
; --------- PAGE 0
RR0 061H
SSV0: DS 1
SSV1: DS 1
SSTEMP: DS 1

; --------- PAGE 6
ORGG 00H ; On-board data RAM (Block B1).
OUTBFA: DS 1 ; Channel A output buffer.
OUTBFB: DS 1 ; Channel B output buffer.
COMMAND: DS 1 ; Host command word.
COMMAND2: DS 1 ; Value passed to TMS25.
DISP: DS 1 ; Value passed to HOST.
HOSTBUF: DS 1 ; Buffer host status.
DUMMY: DS 1 ; Dummy register.
Vd: DS 1 ; DC link voltage.
Id: DS 1 ; DC link current.

;--------- MAIN INTERRUPT (0.2 ms) ROUTINE.
THETAL: DS I
THETAH: DS 1
We: DS 1
Kslip: DS 1
SIN: DS 1
COS: DS 1
Iqs: DS 1
Ids: DS 1
Ias: DS 1
Ibs: DS 1
RT34: DS 1
TEMPO: DS 1

;--------- DATA BUFFER STORAGE VARIABLES.
BUFAL: DS 1
BUFAH: DS 1
COL: DS 1
LOW: DS 1
BFADATA: DS 1
BFBDATA: DS 1
TEMP2: DS 1
SACCH: DS 1
SACC: DS 1
ZERO: DS 1
; FEEDBACK SPEED COMPUTATION ROUTINE.

Wrst: DS 1
STP: DS 1
SSAMP: DS 1
MNEW: DS 1
MOLD: DS 1
TNEW: DS 1
TOLD: DS 1
Wr1: DS 1
WROLD: DS 1
DELM: DS 1
DELT: DS 1
DElwrc: DS 1

; DEFINE VARIABLES FOR PI CONTROLLER

IQH: DS 1
IQL: DS 1
ILIM: DS 1 ; Limit for test of iqs*max (32767-IQMAX).
EK: DS 1 ; Speed error e(k).
EK1: DS 1 ; e(k-1).
WREF: DS 1 ; Speed reference.
KWREF: DS 1 ; Constant for RPM-->decimal conversion.
KBR: DS 1 ; " " decimal -->RPM conversion.
IQMAX: DS 1 ; IQMAX location.
K1: DS 1 ; Modified PI gains.
K2: DS 1

; VARIABLES FOR HOST/DSP COMMUNICATION.

INCOMDATA: DS 1 ; Main routine step.
MSTEP: DS 1 ; Buffer storage var. 1 (Ch. A).
VAR1: DS 1 ; Buffer storage var. 2 (Ch. B).
VAR2: DS 1

; VARIABLES FOR TRANSITION CONTROL.

FSS: DS 1 ; Flag for steady state (1=ss, 0=dyn.mode).
CSS: DS 1 ; Counter for state transition.

; VARIABLES FOR NEW ADC.

PHIGH: DS 1 ; Sum of Vd(k)*idc(k)
PLOW: DS 1
TREG: DS 1
Idc: DS 1
SPdH: DS 1
SPdL: DS 1

; VARIABLES FOR Pdc COMPUTATION

SWT0: DS 1 ; S/W timer 0: variable Ts, for Buffer storage
SWT1: DS 1 ; S/W timer 1: 1 ms for PI speed and FFTC filter
SWT2: DS 1 ; S/W timer 2: 50 ms for Pdc ave comp.
SWT3: DS 1 ; S/W timer 3: 0.5 s for FEOPT, FFTC
Pdc: DS 1 ; Pdc average over 50 ms
Pdc1: DS 1 ; Instantaneous Pdc(k)
TEMP: DS 1 ; Used in ADCGIL
PTEMP: DS 1 ; Used in computation of Pdc

; VARIABLES FOR SLIP GAIN COMPUTATION (VARIABLE FLUX)

Idef: DS 1 ; Filtered value of Ide (=flux/Lm)
xks: DS 1 ; State variable for filter 1
A1: DS 1 ; Constants for filter 1
; VARIABLES FOR FFTC
Dlde: DS 1 ; Delta Ide
Alde: DS 1 ; Auxiliary Ide
Ide: DS 1 ; Isds*
Lm: DS 1 ; Magnetizing inductance
Mbs2: DS 1 ; Coefficient for Lm computation
Nfde: DS 1 ; New steady state estimate of rotor flux
Fdre: DS 1 ; Current value of flux estimate
Dlqe: DS 1 ; Delta lqe
lqep: DS 1 ; lqs*, from PI speed controller
Alqe: DS 1 ; Actual lqs*, including compensation
lqe: DS 1 ; Auxiliary variable (=lqe)
SDlqe: DS 1 ; Sum of Delta lqs. Staircase shape
SLdfe: DS 1 ; Output of 1st order filter (filtered SDlqe)
lmax: DS 1 ; (32767-lmax)
lmin: DS 1 ; (32768-lmin)/2
IP2: DS 1 ; Ia**2
Lm0: DS 1 ; Linear value of Lm
xkf: DS 1 ; State variable for 1st order filter FOFT
TLest: DS 1 ; Torque estimate

; VARIABLES FOR FEEDBACK SPEED SIGNAL
A2: DS 1 ; Coefficients for filter FOFW
B2: DS 1
C2: DS 1
D2: DS 1
XWK: DS 1 ; State variable
Wr: DS 1 ; Filtered speed signal
XKL: DS 1 ; 16 lsb of xkw (Speed fb)
XKLS: DS 1 ; xks (Slip gain)
XKLT: DS 1 ; xkf (Torque comp.)

; TEST VARIABLES
SWeh: DS 1 ; Average computation of Wr
SWel: DS 1
lqave: DS 1
SPDH2: DS 1
SPDL2: DS 1

; VARIABLES FOR Pdc FILTER
Pdc1: DS 1 ; Original Pdc
Pd16: DS 1 ; Pd ave. over 8 samples
Poffset: DS 1
A3: DS 1 ; Coefficients for filter FOFP
B3: DS 1
C3: DS 1
D3: DS 1
xpk: DS 1 ; State variable
XPL: DS 1 ; 16 lsb of xpk (Pdc filter)

; DEFINE VARIABLES AND ADJUSTABLE CONSTANTS FOR FEOP T.
; PAGE 7.

ORG 30h ; Use data page 7 for FEOPT variables.
TLest2: DS 1 ; Load torque estimate, dp 7.
Wr2: DS 1 ; Speed, dp7.
DPI: DS 1 ; DPI = Pi(k)-Pi(k-1).

232
CG1: DS 1 ; Coef. for GID computation
CG2: DS 1
CG3: DS 1
G1: DS 1 ; Coef. for GPb computation
G2: DS 1
GID: DS 1 ; Scale factor for Dlde
GIDMAX: DS 1 ; Impose max. value to GID (32767-1966)
GIDMIN: DS 1 ; Impose min. value to GID (32767+655/2)
GPb: DS 1 ; Scale factor for DPi
GPbMAX: DS 1 ; Impose max. value to GPb (32767-17564)
GPbMIN: DS 1 ; Impose min. value to GPb (32767+4378)
Dldmin: DS 1 ; Dldmin is the min. value for Dlde
JF: DS 1 ; Interval index for DPi
MP1: DS 1 ; Degree of membership for DPi (Left)
MP2: DS 1 ; Same (right)
M11: DS 1 ; Degree of membership for last Dlds (Pos.)
M12: DS 1 ; Same (Neg.)
MRA: DS 1 ; Truth value for rule A
MRB: DS 1 ; B
MRC: DS 1 ; C
MRD: DS 1 ; D
DIA: DS 1 ; Control signal due to rule A
DIB: DS 1 ; B
DIC: DS 1 ; C
DID: DS 1 ; D
Dlds: DS 1 ; Per-unit value of delta Ids
DMONE: DS 1 ; Degree of membership value of one
RP12: DS 1 ; 1/(P1-P2)
RP23: DS 1 ; 1/(P2-P3)
RP3: DS 1 ; 1/P3
RL23: DS 1 ; 1/(L2+L3)
Pik: DS 1 ; Pdc(k)
Pik1: DS 1 ; Pdc(k-1)
DPimax: DS 1
L2lim: DS 1
SUMMR: DS 1 ; Sum of truth values of fired rules
SUMPROD: DS 1 ; Sum of products (sum of MRi*Di, i=A,B,C,D)
TEMPi: DS 1 ; Temporary variable

; Reset and interrupt vectors
;----------------------------------------------------------------------------
; ORG 0
;RES: B INIT ;Reset vector: currently in RESMON
; ORG 4
;INT1: B AINT ;Channel A service interrupt routine
; ORG 6
;INT2: B BINT ;Channel B service interrupt routine
; ORG 24
;INTT: B TINT ;MAIN INTERRUPT (TIMER)
; ORG 26
;RCV: B ININT ;ADC service interrupt routine

;----------------------------------------------------------------------------
; INITIALIZE PROCESSOR
;----------------------------------------------------------------------------
; ORG 1024 ;WITH RESMON ORG MUST BE 1024
;INIT: LDPK 0

233
LALK PRDC
SACL PRD ;SET TIMER PERIOD
SOVM
SSXM ;Set sign extension mode.
SPM 0 ;No shift in product register output
FORT 0 ;Configure for 16 bit word serial data
SXF ;Set output channel flag
LACK 30 ;Enable RINT(16), AINT(2), BINT(4), TIMER(8)
SACL IMR ;Set interrupt mask register
LDPK 6
LACK 1
SACL INCOMDATA
OUT INCOMDATA, INCOM ;Clear M and T counters

; -------------- CONSTANT INITIALIZATION.
ZAC
SACL SSAMP
LALK DELWRCC
SACL DELWRC

; -------------- Initialization of task handler and speed computation
LDPK 6
IN MOLD, MCNT ;Initialize MOLD=MCOUNT
IN TOLD, TCNT ;" TOLD=TCOUNT
ZAC
SACL Wrl
SACL Wr
SACL Wref
SACL TEst
SACL WROLD
SACL OUTBFA
SACL OUTBFB
SACL STP
LALK SLPGAIN
SACL Kslip
LALK 28377
SACL RT34

; -------------- BUFFER INITIALIZATION
LACK 3
SACL BUFAH
LALK 0FE00H
SACL BUFAL
ZAC
SACL VAR1 ;Selection of variable to be stored
SACL VAR2
SACL ZERO
LACK 1
SACL ONE

; -------------- FILTER 1 INIT: SLIP GAIN COMPUTATION UNDER VAR. FLUX
LALK A1C ;Initialize coefficients
SACL A1
LALK B1C
SACL B1
LALK C1C
SACL C1
LALK D1C
SACL D1
LALK A2C ;Initialize coefficients for Wr filter
SACL A2
LALK  B2C
SACL  B2
LALK  C2C
SACL  C2
LALK  D2C
SACL  D2

LALK  A3C ;Initialize coefficients for Pdc filter
SACL  A3
LALK  B3C
SACL  B3
LALK  C3C
SACL  C3
LALK  D3C
SACL  D3

LALK  lderated ;Initialize ldef (always to rated value)
SACL  ldef
LALK  20133 ;Initialize xks to rated value (due to lder)
SACL  xks
LALK  NUMKSLC ;Initialize numerator of slip gain fraction
SACL  NUMKSL

; ---------- INITIALIZE VARIABLES FOR PI CONTROLLER AND SPEED FILTER
ZAC
SACL  WR1 ;Original feedback speed
SACL  WR ;Filtered feedback speed
SACL  XWK
SACL  WREF
SACL  lqep
SACL  lqe
SACL  IQH
SACL  IQL
SACL  EK1
LALK  K1C
SACL  K1
LALK  K2C
SACL  K2
LALK  IQMV
SACL  IQMAX ;Store iqmv into IQMAX
LALK  32767
SUB  IQMAX ;
SACL  ILIM ;ILIM = 32767 * IQMAX

; ---------- INITIALIZE VARIABLES FOR Pdc COMPUTATION
LALK  5 ;Multiple of 0.2ms (50 --> 10 ms)
SACL  SWT0 ;Counter for Buffer storage (variable)
LALK  5
SACL  SWT1 ;Counter for PI speed ctrl and FOF for FFTC
LALK  1000
SACL  SWT2 ;Counter for Pdc ave. computation
LALK  1000 ; 1000bits or 2 sec.
SACL  SWT3 ;Counter for FEOPT AND FFTC
ZAC
SACL  FSS ;Initially FSS=0 (dynamic mode)
SACL  CSS ;Counter for transition control
SACL  EOM ;Eff. Opt. Mode (user defined)
SACL  Pd16 ;8 samples average
LARK  AR5,7 ;Counter for Pd16 average computation

; ---------- INITIALIZE VARIABLES AND CONSTANTS FOR HOST/DSP COMM.
LALK  29320

235
; ---------- INITIALIZE VARIABLES AND CONSTANTS FOR FFTC
LALK Mbs2c
SACL Mbs2
LALK Idmaxc ;Current limits: Idmax = 12 A
SACL Idmax ;Dldmax=32767-ldmax
LALK Idminc ; Idmin = 3 A
SACL Idmin ;Dldmin=(32768-ldmin)/2
LALK Idrated ;Rated Ide = 9.216 A for 5 hp m/c.
SACL Ide
SACL Alde ;Alde is also initialized to lderated
LALK 6335 ;Rated flux is 0.4833 Wb (=6335)
SACL Nfdre
SACL fdre ;Both fldre and Nfdre are init. to rated values.
LALK 16340 ;Rated Lm (49.38 mH or 16340 bits)
SACL Lm
SACL Lm0c
SACL Lm0

ZAC ;Initialize to zero the remaining variables
SACL SDIqe
SACL SDIqf
SACL Dlde ;Delta Ide
SACL Dlqe ;Delta lqe
SACL xkf ;State variable for FOFT
SACL TLest ;Torque estimate
SACL DISP
SACL XKL ;Lsb of fof, speed, slip gain, torque comp.
SACL XKLS
SACL XKLT

; ---------- INITIALIZE VARIABLES AND ADJUSTABLE CONST. FOR FEOPT
LDPK 7
LALK CG1C ;Coef. for GID computation
SACL CG1
LALK CG2C
SACL CG2
LALK CG3C
SACL CG3
LALK GIDMAXC ;Max and min values for GID
SACL GIDMAX
LALK GIDMINC
SACL GIDMIN
LALK G1C ;Coef. for GPb computation
SACL G1
LALK G2C
SACL G2
LALK GPbMAXC ;Max and min values for GPb
SACL GPbMAX
LALK GPbMINC
SACL GPbMIN
LALK Dldminc ;Min value for Did
SACL Dldmin

236
LALK 1000
SACL DMONE ; Value for degree of membership one
LALK DPimacx ; Value used in the DPi max computation
SACL DPimac
LALK L2limc ; Used for max LDlds computation
SACL L2lim
ZAC
SACL GID

; Initialize constants for evaluation of degree of memb.
LALK RP12C
SACL RP12
LALK RP23C
SACL RP23
LALK RP3C
SACL RP3
LALK RL23C
SACL RL23

; BACKGROUND LOOP - HOST COMMUNICATION AND IDLE TIME

MAIN: EQU $
EINT
LDPK 6 ; Write selected variable to HOST
OUT DISP, WRITEHOST ; VAR --> WRITEHOST (PA15)

; HOST/DSP16 COMMUNICATION ROUTINE
NCLUD HOSTDSP.ASM

; MAIN INTERRUPT ROUTINE

; CONTEXT SAVE
AR7 IS THE STACK POINTER (AR7=124!)
TINT: LARP AR7
LARK AR7,124
MAR ".
SST1 ".
SST ".
EINT
SACH ".
SACL ".
SPH ".
SPL ".
MPYK 1
SPL ".
LDPK 6

; PERFORM FEEDBACK SPEED COMPUTATION
CALL SPEED
LAC Wr1
SACL Wr

; PI SPEED CONTROL
FIRST ORDER FILTERS FOR SLIP GAIN AND FFTC
TRANSITION CONTROL ROUTINE
TS1MS: LDPK 6 ;1 ms sampling time routines
     LAC SWT1 ;Check for the sampling time
     SUBK 1
     SAACL SWT1 ;Decrement counter SWT1
     BGZ TS1RET
     LACK 5 ;5<-->1ms
     SAACL SWT1 ;Reload counter if zero
     CALL PI ;PI speed controller
     CALL FOFL ;Compute estimate of "flux" (Idef).
     CALL FOFT ;Get compensated iqe, for FFTC
     CALL FOFW ;Filter for Iqs* (FFTC use only).
     CALL TRCTL ;Execute transition control routine
     CALL FOFP ;Compute filtered Pdc.
     CALL AVE16 ;Average variables for monitoring only.
TS1RET: EQU $ 

; PERFORM BASIC VECTOR CONTROL FUNCTIONS
CALL VECROT ;Original program: slip calc. + v.r

; OUTPUT VARIABLES (FOR DAC)
SPM 1 LT ias ;Compensates for DAC gain mismatches
MPYK 203 PAC
ADDH ias SACH ias LAC ias
ADDK 90 SAACL OUTBFA ;Channel A of DAC
LT lbs ;Compensates for DAC gain mismatches
MPYK 128 PAC
ADDH lbs SACH lbs LAC lbs
ADDK 120 SAACL OUTBF B ;Channel B of DAC

; COMPUTATION OF Pdc AVERAGE
LDPK 6 LAC SWT2 ;Check for the sampling time of Pdc comput.
SUBK 1 SAACL SWT2 ;Decrement counter SWT2
BGZ PDCTRL LALK 1000 ;SWT2 = 200 ms (1000 x 0.2 ms)
SAACL SWT2 ;Reload counter if zero and
CALL PDCAVE ;Compute Pdc ave.
PDCTRL: EQU $ 

; FUZZY EFFICIENCY OPTIMIZATION AND
; FEED-FORWARD TORQUE COMPENSATION
LAC FSS ;Test operating mode
If FSS = 1, go to steady state mode

; Dyn. mode. Make SDlqe = 0

; ... and bypass FFTC and FEOPT

BGZ SSMD ; If FSS = 1, go to steady state mode
ZAC ; Dyn. mode. Make SDlqe = 0
SACL SDlqe
B FRET

SSMD:
LAC SWT3 ; Check for the sampling time of FFTC AND FEOPT
SUBK 1
SACL SWT3 ; Decrement counter SWT3
BGZ FRET
LALK 10000 ; TIMER = 2 S (10000*0.2ms)
SACL SWT3 ; Reload counter if zero and
CALL FEOPT ; Calculate Dlde by fuzzy logic
CALL FFTC ; Compute value of Dlqe

FRET: EQU $

;--------------------------------------------------------
; BUFFER STORAGE
;--------------------------------------------------------
; SELECT THE VARIABLE TO BE STORED IN CHANNEL A

SPM 1
LDPK 6
LAC VAR1
SUBK 1 ; Case VAR1 = 1
BNZ VA2
LAC lde
SACL DISP
B VAEND

VA2: SUBK 1 ; Case VAR1 = 2
BNZ VA3
LAC lqe,1
SACL DISP
B VAEND

VA3: SUBK 1 ; Case VAR1 = 3
BNZ VBSTART ; Wrong value for VAR1; check VAR2
LAC Pd16 ; Pd16 is the 1.6 s average of Pdc
SACL DISP

VAEND: SACL BFADATA ; Store selected variable into CH. A

; SELECT THE VARIABLE TO BE STORED IN CHANNEL B

VBSTART:
LDPK 6
LAC VAR2 ; Case VAR2 = 1
SUBK 1
BNZ VB2
LAC Pdc
SUB Poffset
SACL DISP.5
LAC DISP
B VBEND

VB2: SUBK 1 ; Case VAR2 = 2
BNZ VB3
LAC SDlqf,1 ; Torque compensating signal.
SACL DISP
B VBEND

VB3: SUBK 1 ; Case VAR2 = 3
BNZ VAREND ; Value of VAR2 is not valid; ignore it.

;--------------------------------------------------------
; BUFFER STORAGE
;--------------------------------------------------------
; SELECT THE VARIABLE TO BE STORED IN CHANNEL A

SPM 1
LDPK 6
LAC VAR1
SUBK 1 ; Case VAR1 = 1
BNZ VA2
LAC lde
SACL DISP
B VAEND

VA2: SUBK 1 ; Case VAR1 = 2
BNZ VA3
LAC lqe,1
SACL DISP
B VAEND

VA3: SUBK 1 ; Case VAR1 = 3
BNZ VBSTART ; Wrong value for VAR1; check VAR2
LAC Pd16 ; Pd16 is the 1.6 s average of Pdc
SACL DISP

VAEND: SACL BFADATA ; Store selected variable into CH. A

; SELECT THE VARIABLE TO BE STORED IN CHANNEL B

VBSTART:
LDPK 6
LAC VAR2 ; Case VAR2 = 1
SUBK 1
BNZ VB2
LAC Pdc
SUB Poffset
SACL DISP.5
LAC DISP
B VBEND

VB2: SUBK 1 ; Case VAR2 = 2
BNZ VB3
LAC SDlqf,1 ; Torque compensating signal.
SACL DISP
B VBEND

VB3: SUBK 1 ; Case VAR2 = 3
BNZ VAREND ; Value of VAR2 is not valid; ignore it.

;--------------------------------------------------------
; BUFFER STORAGE
;--------------------------------------------------------
; SELECT THE VARIABLE TO BE STORED IN CHANNEL A

SPM 1
LDPK 6
LAC VAR1
SUBK 1 ; Case VAR1 = 1
BNZ VA2
LAC lde
SACL DISP
B VAEND

VA2: SUBK 1 ; Case VAR1 = 2
BNZ VA3
LAC lqe,1
SACL DISP
B VAEND

VA3: SUBK 1 ; Case VAR1 = 3
BNZ VBSTART ; Wrong value for VAR1; check VAR2
LAC Pd16 ; Pd16 is the 1.6 s average of Pdc
SACL DISP

VAEND: SACL BFADATA ; Store selected variable into CH. A

; SELECT THE VARIABLE TO BE STORED IN CHANNEL B

VBSTART:
LDPK 6
LAC VAR2 ; Case VAR2 = 1
SUBK 1
BNZ VB2
LAC Pdc
SUB Poffset
SACL DISP.5
LAC DISP
B VBEND

VB2: SUBK 1 ; Case VAR2 = 2
BNZ VB3
LAC SDlqf,1 ; Torque compensating signal.
SACL DISP
B VBEND

VB3: SUBK 1 ; Case VAR2 = 3
BNZ VAREND ; Value of VAR2 is not valid; ignore it.
LAC Wr1,1 ;Speed.
SACL DISP
VBEND: SACL BFBDATA

; BUFFER STORAGE SUBROUTINE

VAREND:
LPC 6  ;Check for the sampling time of Buffer Storage.
LAC SWTO
SUBK 1  ;Decrement counter SWTO
SAACL SWTO
BGZ BUFRET  ;By-pass if not required
LALK 100  ;SWTO = 10 ms (variable) 50 x 0.2ms = 10ms
SAACL SWTO
CALL BUFSTR  ;Perform buffer storage
BUFRET: EQU $5

; CONTEXT RESTORE
LARP AR7
MAR *+  
MAR *+  
LT *+  
MPYK 1  
LT *+  
MAR *+  
LPH *+  
ZALS *+  
ADDH *+  
DINT
LDPK 0  
SST1 SSTEMP
BIT SSTEMP,0bh
BBNZ PPPP
LDPK 6
LST *+  
EINT  
RET
PPPP: LDPK 6
LST *+  
EINT  
RET

; LIST OF INCLUDED SUBROUTINES

NCLUD ADC.ASM ;ADC interrupt service subroutine.
NCLUD BUFSTO.ASM ;Buffer storage subroutine
NCLUD DAC.ASM ;DAC interrupt service subroutine.
NCLUD PLASM ;PI speed controller.
NCLUD SPEED.ASM ;Feedback speed computation.
NCLUD PDCOMP.ASM ;Average DC link power computation.
NCLUD FFTC.ASM ;Feedforward pulsating torque compensation.
NCLUD FEOPT.ASM ;Fuzzy efficiency optimization controller.
NCLUD FOFG.ASM ;First order filter for slip gain computation
NCLUD FOFT.ASM ;First order filter for Iqe compensation, FFTC
NCLUD FOFW.ASM ;First order filter for Iqs* (Use in FFTC only)
NCLUD FOFP.ASM ;First order filter for Pdc.
NCLUD TRNCTRL.ASM ;Transition control.
NCLUD SIN.ASM ;Table for sin/cos theta computation.
NCLUD VR.ASM ;Vector rotation.
NCLUD AVE16.ASM ;Average Pdc computation over 1.024 s.

240
ADC INTERRUPT SERVICE ROUTINE

AR6 is used here to store DRR content

ININT: EQU $  ; If channel B jump to CHB

---------- CONTEXT SAVE AND INPUT DRR CONTENT
SST1  SSV1  ; SST1 and SST instr. always load into page 0!
SST   SSV0
LDPK  0
LAR   6,DRR  ; Note that DRR is in DATA PAGE 0
LDPK  6  ; Remaining of ISR are in DATA PAGE 6
SACH  SACCH  ; Save ACC
SACL  SACCL
SPM   0
SPH   PHIGH  ; Save P register
SPL   PLOW
MPYK  1  ; Save T register (PR=TR)
SPL   TREG

---
Compute \( \text{SPd} = \text{sum of } \text{Vd}(k) \times \text{Idc}(k) \)

LT   Vd
MPY  Idc
PAC
SACH  TEMP,1  ; Now this LS of 1 is for product mode only!
ZALH  SPdH
ADDS SPdL
ADD   TEMP
SACL  SPdL  ; SPdL and SPdH are in DATA PAGE 6
SACH  SPdH

---------- CONTEXT RESTORE
CHEXT: ZALS SACCL  ; Restore ACC
ADDH SACCH
SPM  0
LT   PLOW  ; Restore low P register
MPYK 1  ; (TR) --> PRL
LT   TREG  ; Restore T reg.
LPH   PHIGH  ; Restore High P reg.
LDPK  0
LSTI  SSV1  ; Restore status regs.
LST   SSV0

-------------------- END OF PROGRAM --------------------

241
EINT
RET

;Enable interrupts

; AVERAGE COMPUTATION
; AVERAGE OF SELECTED VARIABLES OVER 1024 SAMPLES

AVE16:
   LARP  5 ;Select AR5, down counter of 16
   BANZ NZERO1 ;AR5=AR5-1, and test if AR5>0
   LALK 1024
   SACL TEMPO
   LAR AR5,TEMP0 ;Case AR5 = 0; Reinitialize to 1024
   ZALH SWeH ; and get average of We
   ADDS SWeL
   SACH lqave,6 ;lqave, average over 1024 samples (1.024 s)
   ZALH SpdH2 ; Also get average of Pdc
   ADDS SpdL2
   SACH Pd16,6 ;Pd16, average over 1024 samples (1.024 s)
   ZAC ;Reset summation.
   SACL SWeH
   SACL SWeL
   SACL SPdH2
   SACL SPdL2
   SACH lqave,6
   SACH SWeH
   SACH SWeL
   SACH TEMPO
   ANDK 1FEH
   SACL LOW
   SACL COL
   LALK 14062 ;Convert Pdc1 into Watts
   SACL PTEMP
   LT Pdc1
   MPY PTEMP
   PAC
   SACH PTEMP ;New Pdc in WATTS.
   ZALH SpdH2 ;Uses double precision to compute average.
   ADDS SpdL2
   ADD PTEMP ;Add Pdc1(WATTS) to summation, Spd (H,L).
   SACL SpdL2
   SACL SpdH2
   RET

; BUFFER STORAGE SUBROUTINE

; DATA STORAGE CH-A

BUFSTR: ZALS BUFAH
   ADDH BUFAH
   ADDK 2
   SACH BUFAH
   SACL BUFAH
   SACL TEMPO,7
   ANDK 1FEH
   SACL LOW
   SACL TEMPO
   ANDK 1FFH
   SACL COL
   OUT ONE,11
IN    TEMPO,8
OUT   LOW,ADDR
OUT   COL,ADDR
OUT   BFADATA,DATA ;White line in GRAPH
OUT   ZERO,11

; ---------------- DATA STORAGE CH-B
LALK   1FFH
SUB    COL
BGZ    JMPFUL
LALK   0FE00H
SACL   BUFAL
LACK   3
SACL   BUFAH
JMPFUL: LAC    LOW
ADDK   1
SACL    LOW
OUT   ONE,11
IN    TEMPO,8
OUT   LOW,ADDR
OUT   COL,ADDR
OUT   BFBDAT A,DATA ;Red line in GRAPH
OUT   ZERO,11
RET

; OUTPUT INTERRUPT SERVICE ROUTINE
; ------------------------------
; ------------------ Channel A
AINT:  EQU    $SSV0
LDPK   6
OUT    OUTBFA, DAC ;Load DAC LATCH
NOP
SXF    ;Set output channel flag
LDPK   0
LST    SSV0
EINT   ;Enable interrupts
RET    ;Return from interrupt
;
; ------------------ Channel B
BINT:  EQU    $SSV0
LDPK   6
OUT    OUTBFB, DAC
NOP
RXF    ;Reset output channel flag
LDPK   0
LST    SSV0
EINT   ;Enable interrupts
RET    ;Return from interrupt
;
; FUZZY EFFICIENCY OPTIMIZATION ROUTINE - FEOPT
; ---------------------------------------------
; START FEOPT
FEOPT: SOVM   ;Use overflow mode throughout FEOPT
SPM    1

243
; ----------- Move some data from dp6 to dp7
LDPK 6
LAR AR1,Wr
LAR AR2,Pdc
LAR AR3,TLest
LDPK 7
SAR AR1,Wr2
SAR AR2,Pik
SAR AR3,TLest2

; ----------- Compute GID, the scale factor for Dld5
LT TLest2
ZALH CG3
MPY CG2
LTA CG1
MPY Wr2
APAC

; ----------- Perform test for max |GID|
ADDH GIDMAX
SUBH GIDMAX
ADDH GIDMAX
B GIDEND
GIDEND: SACH Old

; --------- Compute scale factor for DPl (GPb)
ZALH 02 ;ACCH = 02, DP = 7
LT 01
MPY Wr2
APAC ;ACCH = 01 *Wr + 02

; ----------- Test for max and min GPb
ADDH GPbmax ;Max
SUBH GPbmax
SUBH GPbmin ;Min
SUBH GPbmin
ADDH GPbmin
ADDH GPbmin
SACH GPb ;Final value for GPb

; ----------- Compute DPl (Pdc(k)-Pdc(k-1))*GPb
ZALH Pik
SUBH Pik1 ;ACCH = Pik-Pik(k-1)
ADDH DPimax ;Clamp |DPi| to DPimax
SUBH DPimax
SUBH DPimax ;DPimax = 32767 - max DPl
ADDH DPimax
SACH TEMPi,7 ;Now TEMPi = (Pik-Pik(k-1)) 2**7
LT GPb
MPY TEMPi ;P=(Pik-Pik(k-1)) 2**7 GPb
PAC

244
SACH DPi ; Final value for DPi
DMOV Pik ; Pi(k) --> Pi(k-1)

; ----------- Compute Degree of Membership for DPi
LAC DPi
BGEZ DPiGEZ
ADLK P3
SAACL TEMPi
BLZ NP1
LT RP3
LACK 4
SAACL JP
B DPdone

NP1: LAC DPi
ADLK P2
BLZ NP2
LT RP23
SAACL TEMPi
LACK 3
SAACL JP
B DPdone

NP2: LAC DPi
ADLK P1
LT RP12
SAACL TEMPi
LACK 2
SAACL JP
B DPdone

DPiGEZ: SBLK P2
BLZ POS1
SAACL TEMPi
LACK 7
SAACL JP
LT RP12
B DPdone

POS1: LAC DPi
SBLK P3
BLZ POS2
SAACL TEMPi
LACK 6
SAACL JP
LT RP12
B DPdone

POS2: LACK 5
SAACL JP
LT RP3
LAC DPi
SAACL TEMPi

DPdone: MPY TEMPi
PAC
SACH MP2,2
LAC DMOONE
SUB MP2
SAACL MP1

; ----------- Compute DM for LDlds (=Dld, up to this point)
ZALH Dlds ;Limit |Dlds| to L2
ADDH L2lim
SUBH L2lim
SUBH L2lim
ADDH L2lim
SACH Dlds ;Now |Dlds|\leq L2
LAC Dlds ;Get DM for Positive MF
ADLK L3
SACL TEMPi
LT RL23
MPY TEMPi
PAC
SACH MII NEGMF ;MII is the Positive MF of LDlds
ZAC
SACL MIL ;MII = 0, since DIpi < -L3

NEGMF: ZAC ;Get DM for Negative MF
SUB Dlds ;ACC = -Dlds
ADLK L3
SACL TEMPi ;ACC = -Dlds+L3
MPY TEMPi ;T = RL23, unchanged
PAC
SACH MII2
BEZE IDONE ;If MII2 > 0, the value is valid
ZAC
SACL MI2 ;MI2 = 0, since computed DM<0

IDONE: EQU $

; ----------- Perform rule base evaluation (Min operator)
LAC MP1 ;MRA = MIN(MPI, MII)
SACL MRA ;Assume MRA = MP1
SUB MII ;ACC = MP1-MII
BGEZ NXT1 ;Assumption is correct if ACC<0
LAC MII ;No: MP1>MII. Make MRA=MII
SACL MRA

NXT1: LAC MP1 ;MRB = MIN(MPI, MII2)
SACL MRB ;Assume MRB = MP1
SUB MII2 ;ACC = MP1-MII2
BGEZ NXT2 ;Assumption is correct if ACC<0
LAC MII2 ;No: MP1>MII2. Make MRB=MII2
SACL MRB

NXT2: LAC MP2 ;MRC = MIN(MPI, MII)
SACL MRC ;Assume MRC = MP2
SUB MII ;ACC = MP2-MII
BGEZ NXT3 ;Assumption is correct if ACC<0
LAC MII ;No: MP2>MII. Make MRC=MII
SACL MRC

NXT3: LAC MP2 ;MRD = MIN(MPI, MII2)
SACL MRD ;Assume MRD = MP2
SUB MII2 ;ACC = MP2-MII2
BGEZ NXT4 ;Assumption is correct if ACC<0
LAC MII2 ;No: MP2>MII2. Make MRD=MII2
SACL MRD

NXT4: EQU $
; ----------- Rule A
LAC JP
SUBK 2 ;ACC = JP-2 (table offset)
ADLK EFTBL ;Points to Eff. Optimization Table
TBLR DIA ;Read DIA from table
ADDK 1 ;Change pointer to read DIC
TBLR DIC ;ACC = JP-1

; ----------- Compute remaining control signals (DIB, DID)
LALK DMONE
SUB DIA
SAACL DIB ;DIB = 1 - DIA, for this RB
LALK DMONE
SUB DIC
SAACL DID ;DID = 1 - DIC, for this RB

; ----------- DEFUZZIFICATION BY HEIGHT METHOD
; ----------- Get sum of truth value of fired rules (MRA+MRB+MRC+MRD)
LAC MRA
ADD MRB
ADD MRC
ADD MRD
SAACL SUMMR ;SUMMR = MRA+MRB+MRC+MRD

; ----------- Get sum of products: MRi*Di, i=A,B,C,D
LT MRA
MPY DIA ;ACC = MRA*DIA
LTP MRB
MPY DIB
LTA MRC
MPY DIC
LTA MRD
MPY DID
APAC
SACH SUMPROD ;SUMPROD=MRA*DIA+....+MRD*DID

; ----------- Divide SUMPROD BY SUMMR do get Dlds (Fractional division)
; Here, the sign of the result is defined by SUMPROD only
; since SUMMR is by def. always positive
ZALH SUMPROD
ABS ;Make numerator positive
RFTK 14
SUBC SUMMR ;Result is in low ACC
SAACL Dlds

; LAC SUMPROD
BGEZ DIVDONE ;Done if sign is positive
ZAC
SUB Dlds
SAACL Dlds ;Negate quotient if negative

; ----------- GET VALUE OF Dlds IN AMPS
DIVDONE: LT GID
MPY Dlds
PAC
SACH TEMPi ;TEMPi = Dlds

; ----------- Impose minimum value for Dlds to keep search active
ABS :ACCH = |Dlds|
SUBH Dldmin
BGEZ FEND

247
LAC TEMPI ;Case $|Dlde| < Dldmin. Make $Dlde = Dldmin$
BGEZ POSlde ;Case $Dlde < 0.$
ZAC $Dldmin$
SUB $Dldmin$
SACL TEMPI ;$Dlde = -Dldmin$
LALK L3
SACL Dlds ;Prevents $|Dlds| < L3$
FEND

POSlde: LAC Dldmin ;Case $Dlde >= 0$
SACL TEMPI ;$Dlde = +Dldmin$
LALK L3
SACL Dlds ;Prevents $|Dlds| < L3$
FEND: ROVM
LAC TEMPI
LDPK 6
SACL Dlde ;Store new $Dlde$ into dp 6
RET

; -----------------------------------------------
; DEFINE THE TABLE FOR DM's FOR $Dlds$
EFTBL: DW 11, 12, 13, 0, -13, -12, -12

; FEED FORWARD TORQUE COMPENSATOR

; -------- NEW VALUE FOR $I_{de} = A_{ide}$ (Preliminary value)
FFTC: LDPK 6 ;Select DP 6
SPM 1 ;For product shift correction
LAC Ide
ADD Dlde ;$Dlde$ from FEOPT
SACL Aide ;Preliminary value of new Ide

; ----------- CHECK FOR MAX AND MIN $I_{de}$ (SAFETY REASONS)
SOVM ;Use saturation features for test purpose
ZALH Aide ;$Id_{max}=15729$ (== 12 A), $Id_{min}=3932$ (== 3 A)
ADDH Idmax ;$Id_{max} = 32767$-max $Id$
SUBH Idmax ;End of MAX test
SUBH Idmin ;Test for MIN $Id_{de}$
SUBH Idmin ;Subtract and add twice since
ADDH Idmin ;... $Id_{min} = 0.5*(32768+Id_{min})$
ADDH Idmin ;End of MIN test
SACH Aide ;$Aide$ now is between $Id_{min}$ and $Id_{max}$
SUBH ide ;Corrects $Dlde$ in case $Aide$ violates limits
SACH Dlde ;$Dlde = Aide$-$Id_{de}$

; ----------- COMPUTE ESTIMATE OF $Lm$
LAC Aide
ADLK -ldecO ;$ldecO$ is the linear region breakpoint
BGZ GTRL ;
LALK Lm0c ;$Case$ $Ide < ldecO$, linear region
SACL Lm ;$Lm = Lm0$
B ENDL
GTRL:
SACL TEMP ;$Case$ $Ide >= ldecO$, saturation region
ZALH Lm0 ;$ACCH = Lm0$
LT Mbs2
MPY TEMP ;$TEMP = Ide-ldecO$
SPAC ;$ACC = Lm0-Mbs2*(Ide-ldecO)$
SACH Lm

248
; -------- ESTIMATE NEW VALUE OF ROTOR FLUX
ENDL: LT Lm
MPY Alde
PAC ;ACC = Lm*Alde
ABS ;Flux has to be always positive (Safeguard for div.)
SACH Nfdre ;New steady state value of flux

; -------- COMPUTE Dlqe - STEP SIZE FOR Iqe
IMPLEMENTATION: Dlqe = 2 * ((fdre/2)/Nfdre - 1/2)) * Iqe
; --------DIVIDE NUMERATOR (NUMKSL=Rr/Lr) BY Idef TO GET Kaip
ZALH fdre ;Numerator is placed in High ACC
RC ;0 -> C , Carry bit in STl, before Rotation
ROR ;Divide numerator by 2 (fdre/2)

; -------- START FRACTIONAL DIVISION ROUTINE (Positive numbers only)
RPTK 14
SUBC Nfdre ;ACC = (fdre/2) / Nfdre
ADLK -16384 ;ACC = (fdre/2) / Nfdre - 1/2
SACL TEMP,1 ;TEMP = 2 * ((fdre/2) / Nfdre - 1/2 ), 1 LSHIFT = x 2
LT TEMP
MPY Iqave
PAC
SACH Dlqe

; -------- GET NEW Iqe AND TEST FOR MAX STATOR CURRENT
ADDH Iqe ;ACC=Iqe+Dlqe
SACH Alqe ;Get Alqe**2
SQRAR AIDe ;Get Alde**2; ACC=Alq**2
ZAC ;ACC = Alde**2 + Alq**2 = Is**2
APAC ;ACC = Alde**2 + Alq**2 = Is**2
SACH IP2 ;IP2 = Is**2
LAC IP2
ADLK -MAX ;IMAX is the square of max. stator current
BGZ GTRIS ;Branch if Is**2 < IMAX
LAC SDlqe ;Update SDlqe, sum of Dlqe
SACL SDlqe
DMOV Alde ;Alde --> Ide
DMOV Nfdre ;Nfdre --> fdre for next iteration
B ENDIS

GTRIS: ZAC ;Case Is**2 > IMAX, neglect changes
SACL Dlqe
ENDIS: SPM 0 ;Restore for compatibility with old routines
RET

;---------------------------------------------------------------
; FIRST ORDER FILTER FOR Pdc SIGNAL
;---------------------------------------------------------------
; TAL = 500 ms
; Tsamp = 1 ms
; *****************************************************
; * y(k) = c x(k) + d u(k)   *
; * x(k+1) = a x(k) + b u(k) *
; *****************************************************
; *****************************************************
; -------- COMPUTE Pdc - Filtered version of Pdcl
FOFP: LDPK 6
SLIP GAIN COMPUTATION, INCLUDING FIRST ORDER FILTER

Tr = 0.1073s (Rotor time constant)
Tsamp = 1 ms

**y(k) = c x(k) + d u(k)**

**x(k+1) = a x(k) + b u(k)**

---

FOFI: SPM
ZAC
LT xks
MPY Cl
LTA Nfdre
MPY D1
APAC ldef
SACH ldef
ZAC
MPY B1
LTA xks
MPY Al
APAC

**y(k) = c x(k) + d u(k)**

**x(k+1) = a x(k) + b u(k)**

; ---DIVIDE NUMERATOR (NUMKSL=Rr/Lr) BY ldef TO GET Kalip
LAC ldef
ABS ;Get abs of ldef to avoid error (ldef<0 always)
SACL ldef ;
ZALH NUMKSL ;Numerator is placed in High ACC

; --- START DIVISION ROUTINE (Positive numbers only)
RPTK 14
SUBC ldef
SACL Kalip
RET

---
FIRST ORDER FILTER FOR FFTC · COMPUTATION OF SDiqf

Tr = 0.1819s (Rotor time constant)
Tsamp = 1 ms

\[ y(k) = c x(k) + d u(k) \]
\[ x(k+l) = a x(k) + b u(k) \]

\[ y(k) \rightarrow Iqave \quad u(k) \rightarrow SDiqf \]
\[ a \rightarrow A1 \quad b = B1 \]
\[ c \rightarrow C1 \quad d \rightarrow D1 \]
\[ x(k) \rightarrow x(k+1) \rightarrow x(k+2) \]

\[ y \rightarrow Iqave \quad u \rightarrow SDiqf \]
\[ a \rightarrow A1 \quad b = B1 \]
\[ c \rightarrow C1 \quad d \rightarrow D1 \]
\[ x(k) \rightarrow x(k+1) \rightarrow x(k+2) \]

\[ \text{GET COMPENSATED Iq} \]
\[ \text{ADD SDiqf} \]
\[ \text{LAC Iqep} \]
\[ \text{ADD SDiqf} \]
\[ \text{SACL Iq} \]

\[ \text{COMPUTE TORQUE ESTIMATE, TTest} \]
\[ \text{LT Iqave} \]
\[ \text{MPC fdre} \]
\[ \text{PAC} \]
\[ \text{SACL TTest} \]

\[ \text{FIRST ORDER FILTER FOR Iqs* (Used in FFTC only)} \]

TAL = 0.1 sec
Tsamp = 1 ms

\[ y(k) = c x(k) + d u(k) \]
\[ x(k+1) = a x(k) + b u(k) \]

\[ y \rightarrow Iqave \quad u \rightarrow SDiqf \]
\[ a \rightarrow A2 \quad b \rightarrow B2 \]
\[ c \rightarrow C2 \quad d \rightarrow D2 \]
\[ x(k) \rightarrow x(k+1) \rightarrow x(k+2) \]

\[ \text{COMPUTE Iqave, filtered version of Iqs*} \]
FOFW: LDPK 6
SPM 1
ZAC
LT  xwk
MPY  C2  ;ACC = c x(k) , T = u(k)
LTA  lqe
MPY  D2  ;ACC = c x(k) + d u(k)
APAC  lqave  ;Correct result before storing ( * 2 )
ZAC
MPY  B2  ;T = u(k) has not changed
LTA  xwk  ;ACC = b u(k) , T = x(k)
MPY  A2  ;ACC = a x(k) + b u(k)
APAC
ADDS  XKL  ;Add 16 lsb from last iteration
SACH  xwk  ;Actually x(k+1) ---> x(k) for next iteration
ANDK  65280  ;Get upper half of ACCL
SACL  XKL  ;Save 16 lsb for next iteration
RET

HOST COMMUNICATION ROUTINE

LAC  MSTEP
BZ  SBF0  ;MSTEP=0
SUBK  1
BZ  SBF1  ;MSTEP=1
SUBK  1
BZ  SBF2  ;MSTEP=2
ZAC  ;MSTEP value is incorrect, make MSTEP=0
SACL  MSTEP

CASE MSTEP=0 : READ FIRST VALUE FROM HOST (Code)
SBF0:  IN  HOSTBUF, HOSTSTAT;Read host status
DINT
BIT  HOSTBUF, 15  ;Test host status
BBZ  MAIN  ;Loop if no data is available
EINT
IN  COMMAND, READHOST  ;Read host 1st data (COMMAND)
CALL  CASEA  ;Single command functions
B  MAIN

CASE MSTEP=1 : READ SECOND VALUE FROM HOST (Par. value)
SBF1:  IN  HOSTBUF, HOSTSTAT  ;Read host status
DINT
BIT  HOSTBUF, 15  ;Test host status
BBZ  MAIN  ;Loop if no data is available
EINT
IN  COMMAND2, READHOST  ;Read host 2nd data (COMMAND2)
LACK  2  ;Make MSTEP=2
SACL  MSTEP
B  MAIN

CASE MSTEP=2 : FUNCTIONS THAT REQUIRE 2 COMMANDS
; IDENTIFY AND STORE INFORMATION IN PROPER PARAM/VAR.
SBF2:  ZAC  ;Set MSTEP=0
SACL  MSTEP

S0 THROUGH S6 TEST FOR EACH PARTICULAR COMMAND(INSTRUCTION)
S0:  LAC  COMMAND
SUBK  2  ;COMMAND=2? (Wref input)
BNZ  S1  ;NO: Test next condition
SPM  2
ZAC 252
LT  COMMAND2 ; YES: Read Wref (in rpm)
MPY  KWREF ; and convert it to software value
APAC
SACH  WREF ; Now Wref is correct software value
SPM  0
B  CMDEND

S1:  LAC  COMMAND
    SUBK  6
    BNZ  S2 ; COMMAND=6 (Def. Buffer A var.)
    LAC  COMMAND2
    SACL  VAR1 ; VAR1 is the code for 1st var.
    B  CMDEND

S2:  LAC  COMMAND
    SUBK  7 ; COMMAND=7 (Def. Buffer B var.)
    BNZ  S3
    LAC  COMMAND2
    SACL  VAR2 ; VAR2 is the code for 2nd var.
    B  CMDEND

S3:  LAC  COMMAND
    SUBK  8
    BNZ  S4 ; COMMAND=8 (Input Poffset)
    LAC  COMMAND2
    SACL  Poffset ; Offset value for Pdc.
    B  CMDEND

S4:  LAC  COMMAND
    SUBK  9 ; COMMAND=9 (Input K1 of PI
    BNZ  S5 ; speed controller)
    LAC  COMMAND2
    SACL  K1
    LDPK  6
    B  CMDEND

S5:  LAC  COMMAND
    SUBK  10 ; COMMAND=10 (Input K2 of PI
    BNZ  S6 ; speed controller)
    LAC  COMMAND2
    SACL  K2
    B  CMDEND

S6:  LAC  COMMAND ; DEBUG FUNCTION
    SUBK  11 ; COMMAND=11 (Input Ide ids*)
    BNZ  CMDEND ; set S/W timer 3 to 2500 and
    LAC  COMMAND2 ; reset buffer storage
    SACL  Ide
    LALK  10000
    SACL  SWT3
    B  CMDEND

CMDEND:  ZAC ; If the instruction was acknowledged
    SACL  MSTEP ; set MSTEP=0 and loop again
    B  MAIN

; ----------- RESET BUFFER STORAGE SUBROUTINE
BUFSTO:  DINT
    ZAC
    SACL  BUFAH
    SACL  BUFAL
    SACL  COL

253
SAACL LOW
EINT
RET

; ----------- CASE A: Single instruction routines

CASEA: LAC COMMAND ;COMMAND=0 :Stop system
    BNZ CMD1
    LACK 1
    SAACL INCOMDATA
    OUT INCOMDATA,INCOM
    B AEND

CMD1: LAC COMMAND ;COMMAND=1 : Set rated flux
    SUBK 1
    BNZ CMD2
    LACK 3
    SAACL INCOMDATA
    OUT INCOMDATA,INCOM
    B AEND

    SUBK 3
    BNZ CMD3
    LACK 1
    SAACL EOM
    CALL BUFSTO
    LALK 10000
    SAACL SWT3
    B AEND

CMD3: LAC COMMAND ;COMMAND=4: Enter dynamic mode
    SUBK 4
    BNZ CMD4
    ZAC
    SAACL EOM
    B AEND

CMD4: LAC COMMAND ;COMMAND=5: Start buffer storage
    SUBK 5
    BNZ CMD5
    CALL BUFSTO
    B AEND

CMD5: LALK 1 ;Case of 2 instr. commands
    SAACL MSTEP
    RET

AEND: ZAC MSTEP
    SAACL MSTEP
    RET

; ROUTINE FOR Pdc (average) COMPUTATION

; NOTES: 1) AVERAGE IS COMPUTED EVERY 200 ms INTERVAL;
; 2) SPd=(SPH,SPdL) IS THE SUM OF Vd(k)*Idc(k) OVER 200 ms
; (TRAPEZOIDAL RULE OF INTEGRATION), FROM ADC ISR
; 3) THE AVERAGE IS COMPUTED BY MAKING
; SUM OF Vd(k)*Idc(k) FOR k = 1,10000) * (1/10000)
; 4) TO INCREASE RESOLUTION, THE SUM IS FIRST MULTIPLIED BY 2^3
; AND THEN DIVIDED BY 2**16 (TAKING THE ACCH)
5) In order to have correct scale, the result is multiplied by 26844 (= 1/1.2207). Overall effect is:

\[(1/10000) = 2^{**3} / (2^{**16} * 1.2207)\]

PDCAVE:

<table>
<thead>
<tr>
<th>LDPK</th>
<th>SPM</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDPK</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>ZALH</td>
<td>SPdH</td>
<td></td>
</tr>
<tr>
<td>ADDS</td>
<td>SPdL</td>
<td></td>
</tr>
<tr>
<td>ABS</td>
<td>Pdc1,3</td>
<td></td>
</tr>
<tr>
<td>SACH</td>
<td>PddH</td>
<td></td>
</tr>
<tr>
<td>SACL</td>
<td>SPdH</td>
<td></td>
</tr>
<tr>
<td>SACL</td>
<td>SPdL</td>
<td></td>
</tr>
</tbody>
</table>

LDPK 6 ;Store full SPd into ACC
ZALH SPdH ;Only positive Pdc are possible with diode rect.
ADDS SPdL ;This value is actually 1.2207 * (true Pdc)
ABS ZALH Pdc1,3 ;Reset SPd for new 50 ms interval
SACH SACL SPdH
SACL SACL SPdL

; Compute true Pdc

LALK 17619 ;Constant to convert Pdc to its correct scaled value (k/2 = 17619)
SACL PTEMP
LT Pdc1
MPY PTEMP
PAC
SACH Pdc1,1 ;Left shift 1 to get proper scale
RET

; PI SPEED CONTROLLER FOR VECTOR CONTROL SYSTEM

PI:

LDPK 6 ;USE PAGE 6
SPM 1
LAC WREF ;ACC = WREF
SUB WR ;ACC = WREF-WR
SACL EK ;e(k) = WREF-WR
ZALH IQL ;ACC = iqs*(k-1)
ADDS IQL
LT EK1 ;T = e(k-1)
MPY K2 ;P = K2*e(k-1)
LTD EK ;e(k-1) -> e(k)-e(k-1), iqs*(k-1)+K2*e(k-1) -> ACC
MPY K1
APAC ;ACC = iqs*(k-1)+K2*e(k-1)+K1*e(k) = v
SACH IQL
SACL IQL

; Test for max and min iqs* (u = f(V))

SOVM ;Set overflow mode
ADDH ILIM ;Test for max iqs*
SUBH ILIM
SUBH ILIM ;Test for minimum iqs*
ADDH ILIM
SACH Iqep,6 ;Iqep (multi. by 64 = 2**6 for better precision)
ROVM ;To prevent error in other routines
RET

; SPEED FEEDBACK SUBROUTINE

; NOTE: Modified for 5 hp Class B machine.

SPEED: LDPK 6
SPM 0
LAC SSAMP ;Increment iteration counter SSAMP
ADDK 1

255
SACL SSAMP
LAC STP ;STP = 0 --> Read M, T counts
BZ STP0
SUBK 1
BZ STPI ;STP = 1 --> Compute new Wr
ZAC
SACL STP ;If STP not 0 or 1, reset to 0

STP0:
LAC SSAMP ; STEP 0 : Read M (P.G.) and T (Clock)
SBLK SSAMPC ; =1msec
BLZ SFEXIT
IN MNEW,MCNT ;Input M, T counters content
IN TNEW,TCNT ;Read MCNT again for security reasons
LAC TEMP0
SUB MNEW
BNZ SFEXIT ;If TEMP0 not = MNEW, neglect readings
LAC MNEW ; New data has been latched?
SUB MOLD
BZ STP02 ; No: Check if max Ts has elapsed.

; -------- Case MNEW not= MOLD
LAC SSAMP ; Greater than max sampling time?
SBLK STMAXC
BGZ STP01 ; Yes: speed is close to 0. Branch to STP01

; LAC MNEW ; calc. delta M
SUB MOLD
SACL DELM,6 ; dM=32*(MNEW-MOLD ) FOR CLASS B
LALK 18641 ; USE NEW PULSE ENCODER
SACL TEMP0 ; FACTOR = 0.5689
LT TEMP0 ; Overall: 1/7.0313 = 1/4 * 0.5689
MPY DELM
PAC
SACH DELM,1
LAC TNEW
SUB TOLD
SACL DELT ;DELT = TNEW - TOLD
LACK 1 ; Next iteration execute STPI
SACL STP
B SFEXIT

STP01: ZAC ; SSAMP = 0, Wr = 0
SACL SSAMP
SACL DELM
DMOV MNEW ; MOLD <= MNEW
DMOV TNEW ; TOLD <= TNEW
DMOV Wr1
B SFEXIT

STP02: LAC SSAMP
SBLK STMAXC
BLZ SFEXIT
LACK STMAXC ; SSAMP = Stmax, Wr = 0
SACL SSAMP
ZAC
SACL DELM ; Originally was Wr
B SFEXIT

STP1: EQU $ ; Compute new speed value

256
TRANSITION CONTROL SUBROUTINE

TRCTL: LAC FSS
BGZ FSS1 ; FSS=1 -> Steady state mode.
FSS0: LAC EK ; EK = Wref-Wr
ABS
SBLK TOLst
BLEZ CSST1
RET ; If EK > TOLst, stay in Dyn. mode

CSST1: LAC CSS
ADDK 1
SACL CSS
SUBK 3 ; CSS-3 >= 0 ?
BGEZ SS
RET ; No: stay in dynamic mode

SS: LACK 1 ; Yes: ss mode has been detected.
AND EOM ; AND with EOM flag, to allow operator control
SACL FSS ; FSS = 1 indicates ss mode
BZ RTN ; If FSS = 0, it is because EOM=0!

---------- Initialize Pik1 and Dids for the first iteration
LDPK 6
LALK 10000
SACL SWT3
LALK  1400
ADD    Pdc
LDPK  7
SACL    Pikl ;Initialize Pikl to Pdc+1400, such that DPi=NB
LALK -32767
SACL    Dlids ;Initialize LDlids to NB

RTN:    ZAC
SACL    CSS ;Reset counter
RET

;----------- Case FSS = 1. Check if a dyn. mode has come
FSS1: LAC    EOM ;Check if operator has terminated eff. opt. mode
  BZ    DYN ;If EOM = 0, exit eff. opt mode, regardless of EK
  LAC    EK
  ABS
  SBLK    TOLex
  BGZ    CSST2 ;EK > TOLex ?
  RET ;No: stay in ss mode
CSST2: LAC    CSS ;Yes: Make further tests.
  ADDK    1 ;Increment counter, for dynamic test
  SACL    CSS
  SUBK    3 ;CSS >= 3 ?
  BGEZ    DYN ;No: stay in ss mode
DYN:    ZAC ;Yes: Make transition to dynamic mode
  SACL    FSS ;FSS = 0 indicates ss mode
  SACL    CSS ;Reset counter
  LALK    lderated ;Restore rated flux
  SACL    Ie
  LALK    6335 ;Rated flux for 5 hp class B m/c.
  SACL    fdre ;Reset flux to rated value
  RET

;-----------------------------------------
; PERFORM BASIC VECTOR CONTROL FUNCTIONS
;-----------------------------------------

VECROT: SPM    0
LDPK 6
ZALH    Wr ;We computation. We = Wr + Kslip*Iqe
LT    Kslip ;As PM=0, net effect is Wr+(Kslip/2)*Iqe
MPY    Iqe
APAC
APAC
SACH    We

;------- Calc. angle theta
BP1:    ROVM ;Reset overflow mode
ZALH    THETAH
ADDS    THETAL
LT    We
MPYK    DT
APAC
SACL    THETAL
SACH    THETAH

;---------------- Get SIN(theta), COS(theta) from look-up table
SSXMS ;Set sign extension
LT    THETAH

258
MPYK 2048
PAC
SACH TEMP0
LAC TEMP0
BGEZ POSIN
ADLK 1024; If negative, add 180 degrees
ANDK 1023

; NESIN: SOVM
ADLK SINTBL
TBLR SIN
ADLK TBL90
TBLR COS
LAC SIN
NEG
SACL SIN
LAC COS
NEG
SACL COS
B SINEND

POIN: SOVM ; Set overflow mode
ADLK SINTBL
TBLR SIN
ADLK TBL90
TBLR COS
SINEND: EQU $5

; ----------- Synchronous to stationary conversion
; lqs = lqe * COS + lde * SIN
; lds = lqe * SIN + lde * COS
ZAC
LT SIN
MPY lde
LTA COS
MPY lqe
APAC
SACH lqs, 1
ZAC
MPY lde
LTA SIN
MPY lqe
SPAC
SACH lds, 1

; ----------- 2 TO 3 Phase conversion
LAC lqs
SAACL las
ZAC
SUB lqs, 14
LT lds
MPY RT34
SPAC
SACH lbs, 1
LAC lbs
RET ; ----------- End of vector rotation
B.2 Slip Gain Tuning Assembly Programs.

---

FUZZY LOGIC BASED SLIP GAIN TUNING
FOR VECTOR CONTROL AC200 SYSTEM

---

GILBERTO C. D. SOUSA
UNIVERSITY OF TENNESSEE, KNOXVILLE.

THIS VERSION INCLUDES:
1 - VECTOR CONTROL FUNCTIONS
2 - SPEED COMPUTATION ROUTINE
3 - PI SPEED CONTROL
4 - ADC ISR FOR TWO CHANNELS
5 - DAC ISR
6 - REVERSE VECTOR ROTATION FOR Vds AND Vqs
7 - FILTER FOR Vds AND Vqs
8 - DELTA Q, DELTA Vd COMPUTATION
9 - RULE BASE I - Kf COMPUTATION
10 - RULE BASE II - DELTA Ks COMPUTATION

---

I/O PORT ADDRESS
---

MCNT: EQU 0 ; pulse cnt. read
TCNT: EQU 1 ; clock cnt. read
INSTU: EQU 2 ; inverter status READ
INCOM: EQU 2 ; inverter command WRITE

DATA: EQU 8 ; Buffer data port
ADDR: EQU 9 ; Buffer address port
READHOST: EQU 10 ; Host read port
REFRESH: EQU 11 ; Enable buffer refresh
DAC: EQU 13 ; DAC latch
HOSTSTAT: EQU 14 ; Host status
WRITEHOST: EQU 15 ; Host write port

---

CONSTANTS
---

PRDC: EQU 2000 ; MAIN SAMPLING TIME=2msec
Inderated: EQU 12080 ; RATED Ids = 9.216 A (= 12080)
SLIPGAIN: EQU 244 ; UNDER RATED Ids, SLIPGAIN==244
DT: EQU 4000 ; FOR THETA CALC.
TBL90: EQU 512 ; 90 DEGREE
SSAMPC: EQU 5 ; SPEED SAMPLING=1msec
STMAXC: EQU 10 ; 1s=3msec MAX. SPEED SAMP. TIME
DELWRCC: EQU 32767-100 ; Max speed VARIATION IN 1 MS

---

MEMORY MAPPED REGISTERS (ALREADY DEF. IN RESMON)

; DRR: EQU 0 ; A/D Input data register
; TIM: EQU 2 ; TIMER
; PRD: EQU 3 ; PERIOD REG.
; IMR: EQU 4 ; Interrupt mask register
; GREG: EQU 5 ; Global memory allocation register

---

DEFINE CONSTANTS FOR PI CONTROLLER

K1C: EQU 19637 ; For kp = 2 and ki = 0.2: K1C=19637
K2C: EQU -19635 ; K2C=-19635
IQMV: EQU 391 ; Iqeq limit / 64 == 391
Maximum value of Iqeq=19.11 A == 25047

---

260
; DEFINE CONSTANTS FOR FIRST ORDER FILTER: Vds, Vqs.
A1C: EQU 29647
B1C: EQU 3121
C1C: EQU 31208
D1C: EQU 1560

; DEFINE CONSTANTS FOR FOF OF FEEDBACK SPEED
A2C: EQU 31357
B2C: EQU 2048
C2C: EQU 22090
D2C: EQU 706

; DEFINE CONSTANTS FOR FIRST ORDER FILTER: CE
CURRENTLY USING TAL = 30 ms, Ts = 1 ms.
A3C: EQU 31694
B3C: EQU 1074
C3C: EQU 32231
D3C: EQU 537

; CONSTANTS FOR RULE BASE II, DKs COMPUTATION.
e1: EQU -19660 ;Membership functions for Error
e2: EQU -9830
e3: EQU -1966
e4: EQU 3277
e5: EQU 16384
e6: EQU 32767
c1: EQU 6553 ;Membership function for CE.
c2: EQU 19660
c3: EQU 32767
dk1: EQU -32767 ;Rule base for DKs.
dk2: EQU -19660
dk3: EQU -13107
dk4: EQU -1311
dk5: EQU 4915
dk6: EQU 9830
dk7: EQU 19660
dk8: EQU 32767

; DEFINE DATA RAM
; PAGE 0 SSV0 and SSVI must be in PAGE 0!
ORG 061H
SSV0: DS 1
SSVI: DS 1
SSTEMP: DS 1

; PAGE 6
ORG 00h ;Start of on-board data RAM (PAGE6)
OUTBFA: DS 1 ;Channel A output buffer
OUTFBF: DS 1 ;Channel B output buffer
COMMAND: DS 1 ;Host command word
COMMAND2: DS 1 ;Value passed to TMSC25
DISP: DS 1 ;Value passed to HOST
HOSTBUF: DS 1 ;Buffer host status

; MAIN INT.
THETAL: DS 1
THETAH: DS 1
We: DS 1
Kslip: DS 1
SIN: DS 1
COS: DS 1
Iqs: DS 1
Ids: DS 1
las: DS 1
lbs: DS 1
RT34: DS 1

;--------- DATA SAVE
BUFAL: DS 1
BUFAH: DS 1
COL: DS 1
LOW: DS 1
BFADATA: DS 1
BFBDATA: DS 1
TEMP2: DS 1
SACCH: DS 1
SACCL: DS 1
ZERO: DS 1
ONE: DS 1

;--------- SPEED FEEDBACK
STP: DS 1
SSAMP: DS 1
MNEW: DS 1
MOLD: DS 1
TNEW: DS 1
TOLD: DS 1
Wrl: DS 1
WORLD: DS 1
DELM: DS 1
DELT: DS 1
DELRD: DS 1

;--------- VECTOR CONTROL VARIABLES
Ide: DS 1
Iqe: DS 1
PHIGH: DS 1
PLOW: DS 1
TREG: DS 1

;--------- DEFINE VARIABLES FOR PI CONTROLLER
IQH: DS 1
IQL: DS 1
ILIM: DS 1
; Limit for test of iqs*max (32767*IQMAX)
EK: DS 1
; Speed error e(k)
EK1: DS 1
; e(k-1)
WRF: DS 1
; Speed reference
KWF: DS 1
; Constant for RPM-->decimal conversion
KBR: DS 1
; " decimal --> RPM conversion
IQMAX: DS 1
; IQMAX location
K1: DS 1
; PI gains (derived from, but not equal, kp,ki).
K2: DS 1

;--------- VARIABLES FOR HOST/DSP COMMUNICATION
INCOMDATA: DS 1
; MAIN ROUT. STEP
MSTEP: DS 1
; Buffer storage var. 1 (Ch. A)
VAR1: DS 1
VAR2: DS 1
; Buffer storage var. 2 (Ch. B)
(TIMERS FOR REAL TIME SCHEDULER)

SWTO: DS 1 ;SW timer 0: variable Ts, for Buffer storage
SWTI: DS 1 ;SW timer 1: 1 ms for PI speed and SGT.
SWT2: DS 1 ;Test of speed response

(VARIABLES FOR FEEDBACK SPEED SIGNAL)

A2: DS 1 ;Coefficients for filter FOFW
B2: DS 1
C2: DS 1
D2: DS 1
XWK: DS 1 ;State variable
Wr: DS 1 ;Filtered speed signal
XKL: DS 1 ;16 lsb of xkw (Speed fb)

(VARIABLES FOR SLIP GAIN ROUTINES)

(AVR.ASM: INVERSE VECTOR ROTATION)

SVdH: DS 1
SVdL: DS 1
SVqH: DS 1
SVqL: DS 1
AV: DS 1 ;Coefficients for temp. compensation
BV: DS 1 ;of Vds and Vqs sensing.
RT123: DS 1 ;1 / sqrt(3).
Talv: DS 1 ;Tal for phase shift correction.
FSG: DS 1 ;Flag for slip gain process.
Vds: DS 1 ;Stationary frame D-Q voltages.
Vqs: DS 1
Vde: DS 1 ;Synchronous frame D-Q voltages
Vqe: DS 1
TEMP0: DS 1 ;Auxiliary variable
SIND: DS 1 ;Unit vector with phase shift compensation.
COSD: DS 1

(FILTER.ASM, VDE AND VQE FILTER)

xkd1: DS 1 ;Vde state variable for 1st filter.
xd2: DS 1 ;Vde state variable for 2nd filter.
Vde1: DS 1 ;Output of first Vde filter.
Vdef: DS 1 ;Final Vde, after 2nd filter.
xq: DS 1 ;Vqe state variable.
Vqef: DS 1 ;Filtered Vqe.
xd1: DS 1 ;16 lsb of xkd1
xd2: DS 1 ;16 lsb of xkd2
xq: DS 1 ;16 lsb of xq
A1: DS 1 ;Constants for filter Vde, Vqe filters
B1: DS 1
C1: DS 1
D1: DS 1

(DELTA.CASM, DELTA Q AND DELTA Vd COMP.)

QE: DS 1 ;Actual Q.
QEC: DS 1 ;Command Q.
Qb: DS 1 ;Base Q (= Kq + QEC).
Vde: DS 1 ;Command Vd.
Vqec: DS 1
Vb: DS 1 ;Base Vd.
TEMP1: DS 1 ;Auxiliary variable.
KQ: DS 1 ;Epsilon for Qb.
KV: DS 1 ;Epsilon for Vd.

263
Lsb8: DS 1 ; Stator inductance (Lls+Lm)/4
Lsig: DS 1 ; Combined inductance with sat. (appr. Lls+Llr)
Lsigr: DS 1 ; Rated Lsig.
Rs: DS 1 ; Stator resistance.
NUM: DS 1 ; Auxiliary variable.
Asb2: DS 1
Cs: DS 1

; ----- Page 6, test variables
Iqave: DS 1 ; Average Iqe
S1qH: DS 1
S1qL: DS 1
Weave: DS 1 ; Average We
SWeH: DS 1
SWeL: DS 1
FLAG: DS 1 ; Speed response to square wave comm.
Vs: DS 1 ; Stator voltage magnitude.

; -------------- RULE BASE 1 - Kf COMPUTATION, E AND CE COMPUTATION

DVb4: DS 1 ; Delta Vd/4.
DEMAX: DS 1 ; = 32767 - 1024, Max. DE.
Wmax: DS 1 ; Upper bound for We (32767 - Wb)
Imax: DS 1 ; Upper bound for Iqe (32767 - Iqmax)
Wea: DS 1 ; Wea is the auxiliary We (bounded)
Iqea: DS 1 ; Iqea is the auxiliary Iqe (bounded)

RWb: DS 1 ; 1/Wb, scaled
MWH: DS 1 ; Degree of membership HIGH for Wea.
MWL: DS 1 ; Degree of membership LOW for Wea.
DMONE: DS 1 ; Degree of membership one.

Rlqr: DS 1 ; 1/lqr, scaled.
MIH: DS 1 ; Degree of membership HIGH for Iqea.
MIL: DS 1 ; Degree of membership LOW for Iqea.

MRA: DS 1 ; = MIN(MWH,MIH)
MB: DS 1 ; = MIN(MWL,MIH)
MRC: DS 1 ; = MIN(MWH,MIL)
MRD: DS 1 ; = MIN(MWL,MIL)

Kfl: DS 1 ; Membership functions for Kf.
Kf2: DS 1
Kf3: DS 1

SUMPROD: DS 1 ; Sum of product: MRi*Kfn
SUMMR: DS 1 ; Sum of D.M.'s: MRA+...+MRD
Kf: DS 1 ; Weight factor between Q and Vd errors.
DQE: DS 1 ; Delta Q (error from Q model)
Err: DS 1 ; Combined error.
elmin: DS 1 ; = 32767 + e1
EkC_H: DS 1 ; Clamped Error at time k
EkC_1H: DS 1
EkC_L: DS 1
EkC_1L: DS 1 ; " " from time k-1
GC: DS 1 ; Input gain for change in error.
DE: DS 1 ; Original delta error.
A3: DS 1 ; Coefficients for CE filters.
B3: DS 1

264
C3: DS 1
D3: DS 1

xkc1: DS 1 ; State variables for CE filter.
xkc2: DS 1
xcl1: DS 1 ; 16 lsb of xkc1
xcl2: DS 1 ; 16 lsb of xkc2
CE: DS 1 ; Filtered change in error, bounded by c3.
TEMP3: DS 1 ; Auxiliary variable for p. 7.

; 1/(e2-e1), scaled.
RE2: DS 1
RE32: DS 1
RE3b2: DS 1
RE4: DS 1
RE54: DS 1
RE55: DS 1
RC1: DS 1
RC21: DS 1
RC32: DS 1
Kmax: DS 1 ; 32767 - MaxKs (520) = 32247.
Kmin: DS 1 ; (32767 + MinKs(130))/12 = 16449
DKA: DS 1 ; Rule A value.
DKB: DS 1
DKC: DS 1
DKD: DS 1

ME1: DS 1 ; Error membership value (left).
ME2: DS 1
MC1: DS 1 ; Change in error memb. value (left).
MC2: DS 1
IJ: DS 1 ; Offset for rule base look-up table.
IC: DS 1 ; Row index.
JE: DS 1 ; Column index.
DKs: DS 1 ; Delta Ks.
GKs: DS 1 ; Auxiliary variable.
KsH: DS 1 ; High byte for slip gain.
KsL: DS 1 ; Low byte for slip gain.

; RESET AND INTERRUPT VECTORS
ORG 0
;RESET: B INIT ; Reset vector currently in RESMON
ORG 4
INT1: B AINT ; Channel A service interrupt routine
ORG 6
INT2: B BINT ; Channel B service interrupt routine
ORG 24
INTT: B TINT ; MAIN INTERRUPT (TIMER)
ORG 26
RCV: B ININT ; ADC service interrupt routine

; INITIALIZE PROCESSOR
ORG 1024 ; WITH RESMON ORG MUST BE 1024
INIT: LDPK 0
LALK PRDC
SAACL PRD ;SET TIMER PERIOD

SOVM
SSXM ;set sign extension mode
SPM 0 ;No shift in product register output
FORT 0 ;Configure for 16 bit word serial data
SXF ;Set output channel flag
LACK 30 ;Enable RINT(16), AINT(2), BINT(4), TIMER(8)
SAACL IMR ;Set interrupt mask register
LDPK 6
LACK 1
SAACL INCOMDATA
OUT INCOMDATA, INCOM ;Clear M and T counters

; ******* CONSTANT INITIALIZATION
ZAC
SAACL SSAMP
LALK DELWRCC
SAACL DELWRC
LALK 5
SAACL SWT1
LACK 50
SAACL SWT0
LALK 300
SAACL SWT2
ZAC
SAACL FLAG

; ******* Initialization of task handler and speed computation
LDPK 6
IN MOLD, MCNT ;Initialize MOLD=MCOUNT
IN TOLD, TCNT ; " TOLD=TCOUNT
ZAC
SAACL DLWR0 ;Case of constant acceleration.
SAACL Wr1
SAACL Wr
SAACL Wref
SAACL WROLD
SAACL OUTBFA
SAACL OUTBFB
SAACL STP
LALK SLIPGAIN
SAACL Kslip
LALK 28377
SAACL RT34

; ******* BUFFER INITIALIZATION
LACK 3
SAACL BUFAH
LALK 0FE00H
SAACL BUFAL
ZAC
SAACL VAR1 ;Selection of variable to be stored
SAACL VAR2
SAACL ZERO
LACK 1
SAACL ONE

; ******* FILTER 1 INIT.: SLIP GAIN COMPUTATION UNDER VAR. FLUX
LALK A1C ;Initialize coefficients
SAACL A1
LALK B1C
; Initialize coefficients for Wr filter
SACL A2C
SACL A2
SACL B2C
SACL B2
SACL C2C
SACL C2
SACL D2C
SACL D2
SACL Iderated
; Initialize Ide (always to rated value)
SACL Ide

; INITIALIZE VARIABLES FOR PI CONTROLLER AND SPEED FILTER
ZAC
SACL WR1
SACL WR
SACL XWK
SACL WREF
SACL Iqe
SACL IQH
SACL IQL
SACL EK1
SACL K1C
SACL K1
SACL K2C
SACL K2
SACL IQMV
SACL IQMAX
SUB IQMAX
SACL ILIM

; INITIALIZE VARIABLES AND CONSTANTS FOR HOST/DSP COMM.
LALK 29320
SACL KWREF
SACL KBR

; INITIALIZE VARS AND CONSTANTS FOR SLIP GAIN PROJECT
; DELTAC ROUTINE (p. 6)
LALK 3224
SACL AV
SACL BVs
LALK 18919
SACL RTI23
LALK 17731
SACL Lsb8
LALK 13548
SACL Lsigr
SACL Lsig
LALK 725
SACL Rs
LALK 4
SACL KQ
SACL KV
; Coefficients for Lsig saturation effect.

; FILTER ROUTINE (Vds AND Vqs FILTERS)

; Flag for slip gain process.

; Now using isolation amp. circuit.

; Measured time delay = 72.2 Micro sec.

; Stator voltage magnitude (Vs(0)).

; VARIABLES AND CONSTANTS FOR RULE BASE I

; Uses page 7

; Wmax = 32767 - Wb(-15748).

; Imax = 32767 - Iqrated (=25047).

; DMONE = degree of membership 1.

; RWb = 1/Wb, scaled.

; RLqr = 1/lqrated.

; Kf1 = 0.8 (26214)

; Kf2 = 0.9 (29490)

; Kf3 = 0.98 (32112)

; elmin = -32768 - elmin (-0.6°32768)

; For GC = 400, 18431.

; GC = remainder of 400/256.

; DEMAX = 32767 - max DE (= 1024).

; Initialize coefficients for CE filter

; State var. for CE filter.

; 16 slb of xkc1, xkc2.

; VARIABLES AND CONSTANTS FOR RULE BASE II.

268
LALK 10900 ;REmn = 1 / (em - en)
SACL RE21
LALK 13625
SACL RE32
LALK 27250
SACL RE3b2
LALK 32700
SACL RE4
LALK 8175
SACL RE54
LALK 6540
SACL RE65

LALK 16350 ;RCmn = 1 / (cm - cn).
SACL RC1
LALK 8175
SACL RC21
LALK 8175
SACL RC32

LALK 32247
SACL Ksmax ;Ksmax = 32767 - maxKs.
LALK 16449
SACL Ksmin ;Ksmin = (32767+Ksmin)/2.

LALK 12452 ;Original gain = 82.
SACL GKs ;Output gain.

LALK SLIPGAIN ;Initialize Ksh with rated slip gain.
SACL KsH
ZAC
SACL KsL

; ----------------------------------------
; BACKGROUND LOOP - HOST COMMUNICATION AND IDLE TIME
; ----------------------------------------
MAIN: EQU $
EINT
LDPK 6 ;Write selected variable to HOST
OUT DISP,WRITEHOST ;VAR --> WRITEHOST (PA15)

; ----------------------------------------
; HOST/DSP16 COMMUNICATION ROUTINE
; ----------------------------------------
NCLUD HOSTDSP.ASM

; ----------------------------------------
; MAIN INTERRUPT ROUTINE
; ----------------------------------------

;------------------ CONTEXT SAVE
; AR7 IS THE STACK POINTER (AR7=124)
TINT: LARP AR7
LARK AR7,124
MAR *-
SST1 *-
SST *-
EINT
SACH *-
SACL *-
SPH *-

269
PERFORM SLIP GAIN TUNING ROUTINES WITH $T_s = 0.2$ ms.

CALL IVR ; Reverse vector rotation ($V_{de}$, $V_{qe}$ comp.)
CALL FILTER ; $V_{de}$ and $V_{qe}$ filters.

PERFORM FEEDBACK SPEED COMPUTATION

CALL SPEED
LAC Wr1
SACL Wr

PI SPEED CONTROL

**TS 1MS:**
PERFORM FEEDBACK SPEED COMPUTATION
CALL SPEED
LAC Wr1
SACL Wr

**TS Sims:**
LDPK 6 ; 1 ms sampling time routines
LAC SWT1 ; Check for the sampling time
SUBK 1
SACL SWT1 ; Decrement counter SWT1
BGZ TS1RET
LACK 5 ; S<->1 ms
SACL SWT1 ; Reload counter if zero
CALL PI ; PI speed controller

PERFORM SLIP GAIN TUNING ROUTINES WITH $T_s = 1$ ms.

LAC FSG
BZ TS1RET ; Slip gain tuning is active if FSG = 1.
CALL DELTAC ; Delta Q and delta Vd computation.
CALL RBI ; KF, E and CE computation.
CALL RBlI ; DKs, Ks computation.
TS1RET: EQU $5$

PERFORM BASIC VECTOR CONTROL FUNCTIONS

CALL VECROT ; Original program: slip calc. + v.r

OUTPUT VARIABLES (FOR DAC)

SPM 1 ; Compensates for DAC gain mismatches
LT las
MPYK 203
PAC
ADDH las
SACH las
LAC las
ADDK 64 ; Offset compensation for DAC A
SACL OUTBFA ; Channel A of DAC
LT lbs ; Compensates for DAC gain mismatches
MPYK 139
PAC
ADDH lbs
SACH lbs
LAC lbs
ADDK 115 ;Offset compensation for DAC B
SACL OUTBF
CALL AVE16 ;1.024 sec average for Vdef, Vqef (monitoring).

;---------------------------------------------
; BUFFER STORAGE
;---------------------------------------------
; ---------- SELECT THE VARIABLE TO BE STORED IN CHANNEL A
SPM 1 ;*******************************
LAC VAR1 ;*VAR1 = 1 STORE: Ekch *
LDPK 6 *
SUBK 1 ;Case VAR1 = 1 *
BNZ VA2 *
LDPK 7 *
LAC Ekch
NEG B VAEND

VA2: SUBK 1 ;Case VAR1 = 2
BNZ VA3
LAC Iqe
NEG B VAEND

VA3: SUBK 1 ;Case VAR1 = 3
BNZ VBSTART ;Wrong value for VAR1; check VAR2
Wref

VAEND: LDPK 6
SACL DISP
SACL BFADATA ;Store selected variable into CH. A

;---------- SELECT THE VARIABLE TO BE STORED IN CHANNEL B
;
VBSTART:LDPK 6 ;*******************************
LAC VAR2 ;*VAR2 = 1 STORE: Kslip *
SUBK 1 ;Case VAR2 = 1 *
BNZ VB2 *
LAC Kslip,S
NEG B VBEND

VB2: SUBK 1 ;Case VAR2 = 2
BNZ VB3
LDPK 7
LAC CE
B VBEND

VB3: SUBK 1 ;Case VAR2 = 3
BNZ VAREND ;Value of VAR2 is not valid; ignore it.
LDPK 6
LAC Iqe

VBEND: LDPK 6
SACL BFBDATA
SACL DISP

;---------------------------------------------
; BUFFER STORAGE SUBROUTINE
;---------------------------------------------
; Check for the sampling time of Buffer Storage.

; Decrement counter SWTO

; By-pass if not required

; SWTO = 10 ms (variable) 50 x 0.2ms = 10ms

; Reload counter if zero and

; Perform buffer storage

LIST OF INCLUDED SUBROUTINES

; ADC INTERRUPT SERVICE ROUTINE
NCLUD ADCSG.ASM

; BUFFER STORAGE SUBROUTINE
NCLUD BUFSTO.ASM

; OUTPUT INTERRUPT SERVICE ROUTINE
NCLUD DAC.ASM

; PI SPEED CONTROLLER
NCLUD PICOMP.ASM

; SPEED FEEDBACK SUBROUTINE
NCLUD SPEED.ASM

; First order filter for speed, FOFW
NCLUD FOFW.ASM

; TABLE FOR SIN/COS COMPUTATION
NCLUD SIN.ASM

; BASIC VECTOR CONTROL FUNCTIONS
NCLUD VR.ASM

; Reverse vector rotation.
NCLUD IVR.ASM

; Vde and Vqe filters.
NCLUD FILTER.ASM

; Delta Q and delta Vd computation.
NCLUD DELTAC.ASM

; Kf, E and CE computation.
NCLUD RBL.ASM

; DKs, Ks computation.
NCLUD RBII.ASM

; 16 ms average for Vdef and Vqef.
NCLUD AVE16.ASM

; Stator voltage amplitude computation.
NCLUD VSMOD.ASM

; --------------------------END OF PROGRAM ------------------------
Computation of DELTA Q and DELTA Vd

DELTAC: SPM
LDPK 6
SOVM

Compute QE, actual Q from measurements
LT Vqef
MPY Ide
LTP Vdef ;ACC = Vqef*Ide; TR = Vdef
MPY Iqe
SPAC
SACH QE

Compute QEC, command Q from reference model
SQR A Ide
LTP Ls/8 ;TR = Ls/8, ACCH = Ide^2
SACH TEMP0 ;TEMP0 = Ide^2
MPY TEMP0
PAC
SACH TEMP0,3 ;TEMP0 = 8*Ls/8*Ide^2
SQR A Iqe
LTP Lsig ;TR = Lsig, ACCH = Iqe^2
SACH TEMPI ;TEMPI = Iqe^2
MPY TEMPI
PAC
ADDH TEMP0 ;ACCH = Ls*Ide^2+Lsig*Iqe^2
SACH TEMPI
LT We
MPY TEMPI
PAC
SACH QEC ;QEC = We*(Ls*Ide^2+Lsig*Iqe^2)
ADDH QK ;Qb = QEC + QK ;KQ = constant
SACH Qb

Compute DELTA Q, by division
LAC QEC
SUB QE
SACL NUM ;NUM = QEC-QE

Clamp |QEC - QE| to Qb
LALK 32767
SUB Qb
SACL TEMPI
ZALH NUM
ADDH TEMPI
SUBH TEMPI
SUBH TEMPI
ADDH TEMPI

Divide NUM BY Qb to get DQE (Fractional division)
Here, the sign of the result is given by NUM only,
since Qb is always positive
DST1: ZALH NUM ;Make numerator positive
ABS
RPTK 14
SUBC Qb ;Result is in low ACC
SACL TEMPI
LAC NUM
BGEZ DONE1 ;Done if sign is positive
LAC TEMPI
NEG
SACL TEMPI ;Negate quotient if NUM is negative

DONE1: LAC TEMPI
LDPK 7
SACL DQE ;DQE is now in p. 7!
LDPK 6

;----------- Introduce saturation effect on Lsig.
; xsr = A ls^4 + B ls^2 + C.
; Lsig = Lsigr + xsr*Lsigr

SQR A ide
ZAC ;PR = ide^2.
SQR A lqe ;ACCH = ide^2, PR = lqe^2.
APAC
SACH TEMP0 ;ls2 = ide^2 + lqe^2 = TEMP0.
SQR A TEMP0
PAC
SACH TEMPI ;ls4 = ls^4 = TEMP1.

ZALH Cs
MPY Bsb2 ;TR = ls2 has not changed.
APAC ;PR = (Bs/2) ls^2.
LTA TEMPI ;ACCH = B*ls^2 + C.
MPY Asb2 ;TR = ls4.
APAC ;PR = (As/2) ls^4.
LTA Lsigr ;ACCH = A ls^4 + B ls^2 + C = xsr = TEMP2.
SACH TEMP2
ZALH Lsigr ;TR = Lsigr.
MPY TEMP2
APAC ;ACCH = Lsigr + xsr*Lsigr = Lsig.
SACH Lsig ;Saturation compensated Lsig.

;----------- Perform computation of Vdec, commanded Vd
LT We
MPY Lsig ;TR = We.
LTP lqe ;TR = lqe, ACC = We*Lsig
SACH TEMP0 ;TEMP0 = We*Lsig
MPY TEMP0
LTP ide ;ACCH = We*Lsig*lqe = TEMP0; TR = ide
SACH TEMP0
MPY Rs
PAC ;ACCH = Rs*ide
SUBH TEMP0
SACH Vdec ;Vdec = Rs*ide - TEMP0
LAC TEMP0
ABS
ADD KV ;KV is a constant
SACL Vb,2 ;Vb = 4*(KV + ABS(We*Lsig*lqe))

;----------- Compute DELTA Vd, by division
LAC Vdef ;Sign reversal is intentional!
SACL NUM ;NUM = Vdef - Vdec
LALK 32767
SUB Vb ;Clamp [NUM] to 4*Vb
SACL TEMPI ; to ensure proper fractional division.
ZALH NUM
ADDH TEMPI

274
SUBH TEMP1
SUBH TEMP1
ADDH TEMP1
SACH NUM

;---------- Divide NUM BY Vb(x4) to get DVb4 (Fractional division)
; Here, the sign of the result is given by NUM only,
; since Vb is always positive
ZALH NUM
ABS
RPTK 14
SUBC Vb
SAACL TEMP1
LAC NUM
BGEZ DONE2
LAC TEMP1
NEG
SAACL TEMP1

DONE2:
LACL TEMP1
LDPK 7
SAACL DVb4
LDPK 6
RET

---------------------------
FILTER FOR Vde AND Vqe
---------------------------
fcf = 53.05 Hz (TAL = 3 ms), fsamp = 5 KHz
y(k) = c x(k) + d u(k)
x(k+1) = a x(k) + b u(k)

; Two cascade first order filters are used for Vde
; One filter is used for Vqe
; Discrete version of ECALC (first part of)

FILTER: SPM 1
LT xkd2
MPY C1
LTP Vde
MPY D1
APAC ;ACC = c x(k), T = u(k)
SACH Vdel ;Vde after first filter
MPY B1 ;T = u(k) has not changed
LTP xkd2 ;ACC = b u(k), T = x(k)
MPY A1
APAC ;ACC = a x(k) + b u(k)
ADDS xdl1 ;Add 16 lsb from last iteration
SACH xkd1 ;Actually x(k+1) -> x(k) for next iteration
ANDK 65280 ;Get upper half of ACCL
SAACL xdl1 ;Save 16 lsb for next iteration.

;---------- Get filtered version of Vde
LT xkd2
MPY C1
LTP Vdel
MPY D1
APAC ;ACC = c x(k), T = u(k)
SACH Vdef
MPY B1 ;T = u(k) has not changed
LTP xkd2 ;ACC = b u(k), T = x(k)
; Get filtered Vqe (1st order only)
LT xkq
MPY C1
LTP Vqe
MPY D1
APAC Vqef
SACH Vqef
MPY B1
LTP xkq
MPY A1
APAC xql
SACH xkq
ANDK 65280
SACL xkq
RET

; REVERSE VECTOR ROTATOR FOR Vds AND Vqs COMPUTATION
; Correct phase-shift due to analog filters
IVR: SPM 0

; Angle from forward vector rotator
ZALH THETAH
ADDS THETAL
LT We
MPY Talv
SPAC TEMP0
SACH TEMP0

; Correcting angle is proportional to We, when phase shift is small.
; Current Talv = 9145 bits (457.3 Ms)

; Compute shift-compensated unit vectors
SSXM
LT TEMP0
MPYK 2048
PAC
SACH TEMP0
LAC TEMP0
BOEZ POSIND
ADLK 1024
ANDK 1023

; If negative, add 180 degrees
; Extract the first seven bits only.

; Add address offset
ADLK SINTBL
TBLR SIND
ADLK TBL90
TBLR COSD
LAC SIND
NEG
SACL SIND
LAC COSD

; Reverse sign of vectors
POSIND: SOVM ;Case of positive angle
ADLK SINTBL
TBLR SIND
ADLK TBL90
TBLR COSD

SINED: EQU $

; Make ADC offset corrections and temp. compensation
; for Vds and Vqs sensing.
SPM 1 ;From this point on, use PM = 1.
LAC Vqs
ADLK 164
SACL TEMP1 ;Offset compensation.
LAC Vds
SUBK 40
SACL TEMP2
LT AV ;Temperature effect compensation.
MPY We
PAC
ADDH BVs
SACH TEMP0 ;TEMP0 = Coefficient of correction for temp. effect on Vds and Vqs.
LT TEMP0 ;TR = temp. coeff.
MPY TEMP1
PAC
ADDH TEMP1
SACH TEMP1 ;TEMP1 = Vqs*(1 + coeff)
MPY TEMP2 ;TR = temp. coeff.
PAC
ADDH TEMP2
SACH TEMP2 ;TEMP2 = Vds*(1 + coeff)

; Stationary to synchronous frame conversion
; Vqe = Vqs*COS-Vds*SIN
; Vde = Vqs*SIN+Vds*COS
LT COSD
MPY TEMP1
LTP SIND ;ACC = Vqs*COSD
MPY TEMP2
SPAC ;ACC = Vqs*COSD - Vds*SIND = Vqe
SACH Vqe,l ;Uses a new scale for voltage
MPY TEMP1 ;TR = SIND
LTP COSD ;ACC = Vqs*SIND; TR = COSD
MPY TEMP2
APAC ;ACC = Vqs*SIND + Vds*COSD
SACH Vde,l ;Using new scale for voltage.
RET

; RULE BASE I - Kf COMPUTATION
; ALSO E (Error) AND CE (Change in Error) COMPUTATION
; NOTE: VARIABLES ARE DEFINED IN P. 7
RBI: SPM 1
LDPK 6

277
LAR AR1,We  ;Copy We and Iqe from p. 6 into p. 7.
LAR AR2,Iqe
LDPK 7
SAR AR1,Wea
SAR AR2,Iqea

SOVM ;Use overflow mode to clamp We, Iqe
ZALH Wea  ;Only module of We is relevant.
ABS Wmax  ;Clamp We to Wb = 2200 rpm.
ADDDH Wmax
SUBH Wmax
SACH Wea  ;Wea is the auxiliary We
ZALH Iqea  ;Only module of Iqe is relevant.
ABS Imax  ;Clamp Iqe to Iqrated = 19.2 A.
ADDDH Imax
SUBH Imax
SACH Iqea  ;Iqea is the auxiliary Iqe

; Get degree of membership for Wea and Iqea
LT Wea
MPY RWb  ;RWb is 1/Wb, scaled.
PAC ;ACC = MWH, D.M. HIGH for Wea.
SACH MWH
LAC DMONE  ;ACC = 1 - MWH = MWL
SUB MWH
SACL MWL

LT Iqea
MPY RIqr  ;RIqr is 1/Iqr, scaled.
PAC ;ACC = MIH, D.M. of right MF for Iqea
SACH MIH
LAC DMONE  ;ACC = 1 - MIH = MIL
SUB MIH
SACL MIL

; Perform rule base evaluation (Min operator)
LAC MWH  ;MRA = MIN(MWH,MIH)
SACL MRA  ;Assume MRA = MWH
SUB MIH  ;ACC = MP1-MIH
BLEZ NXT1  ;Assumption is correct if ACC<=0
LAC MIH  ;No: MP1>MIH. Make MRA=MIH
SACL MRA

NXT1: LAC MWL  ;MRB = MIN(MWL,MIH)
SACL MRB  ;Assume MRB = MWL
SUB MIH  ;ACC = MWL-MIH
BLEZ NXT2  ;Assumption is correct if ACC<=0
LAC MIH  ;No: MWL>MIH. Make MRB=MIH
SACL MRB

NXT2: LAC MWH  ;MRC = MIN(MWH,MIL)
SACL MRC  ;Assume MRC = MWH
SUB MIL  ;ACC = MWH-MIL
BLEZ NXT3  ;Assumption is correct if ACC<=0
LAC MIL  ;No: MWH>MIL. Make MRC=MIL
SACL MRC

NXT3: LAC MWL  ;MRD = MIN(MWL,MIL)
SACL MRD  ;Assume MRD = MWL
SUB MIL  ;ACC = MWL-MIL
BLEZ NXT4  ;Assumption is correct if ACC<=0
LAC MIL  ;No: MWL>MIL. Make MRD=MIL

278
DEFUZZIFICATION BY HEIGHT METHOD

Get sum of truth value of fired rules (MRA+MRB+MRC+MRD)

LAC MRA
ADD MRB
ADD MRC
ADD MRD
SACL SUMMR ;SUMMR = MRA+MRB+MRC+MRD

Get sum of products: MRi*Kfn, i=A,B,C,D, n=1,2,3

LT MRA
MPY Kf2
LTP MRB ;ACC = MRA*Kf2, TR = MRB
MPY Kf3
LTA MRC
MPY Kf1
LTA MRD
MPY Kf2
APAC
SACH SUMPROD ;SUMPROD=MRA*Kf2+....+MRD*Kf2

Divide SUMPROD BY SUMMR do get Kf (Fractional division)

Here, the sign of the result is defined by SUMPROD only

since SUMMR is by def. always positive

ZALH SUMPROD
ABS
RPTK 14
SUBC SUMMR ;Result is in low ACC
SACL Kf
LAC SUMPROD
BGEZ DIVDONE ;Done if sign is positive
ZAC
SUB Kf
SACL Kf ;Negate quotient if negative

DIVDONE:EQU $ 

Compute Error (E) and Change in Error (CE).

SPM 0 ;PM = 0 introduces GE = 0.5 automatically!
LALK -32767
ADD Kf
SACL TEMP3,2 ;TEMP3 = -4*(1 - Kf). As (1 - Kf)<0.2
; no overflow occurs.
LT Kf
MPY DQE
ZAC
LT3 TEMP3 ;ACC = -0.5*Kf*DQE, TR = -4*(1 - Kf).
MPY DVB4 ;DVB4 = DVde / 4.
APAC
SACH Err ;Err = -0.5*(Kf*DQ + 4*(1-Kf)*DVB4)
SPM 1 ;Restores PM to its normal value.

Now impose lower (el) bound to Err

(Upper bound e6 has been enforced by OVM = 1)

SUBH elmin ;elmin = -32767 + e1.
ADDH elmin
SACH EkcH ;Ekc = Err, if Err > e1.
SACL EkcL ;Stores in 32 bit to improve precision.

Compute Change in Error
SUBS Ekc_H
SUBH Ekc_IH
; Get delta error in extended precision.
ADDH DEMAX
; Use saturation features to clamp DE to 1024
SUBH DEMAX
; DEMAX = 32767 - 1024 = 32686
SUBH DEMAX
; Define DE = (Ekc - Ekc_l) * 2^S.
DMOV EkcH
; Ekc --> Ekc_l.
DMOV EkcL
; Ekc --> Ekc_l.

; ----------- Pass DE through filters to get CE (final change in error)
LT xkcl
MPY C3
LTP DE
MPY D3
APAC TEMP3
SACH xkcl
MPY B3
; T = u(k) has not changed
LTP xke1
MPY A3
APAC;
ADDx xcll
SACH TEMP3
MPY B3
; T = u(k) has not changed
LTP xkcl
MPY A3
ADDx xcll
SACH xkcl
ANDK 65280
SACL xcll
; Save upper 16 bits for next iteration.

; ----------- Second filter for DE
LT xkc2
MPY C3
LTP TEMP3
MPY D3
APAC;
ADDx xcl2
SACH CE,S
MPY B3
; T = u(k) has not changed
LTP xkc2
MPY A3
APAC;
ADDx xcl2
SACH xkc2
ANDK 65280
SACL xcl2
; Save upper 16 bits for next iteration.
RET

; RULE BASE II - DELTA Ks COMPUTATION

; Use overflow mode throughout RBII
SOVM 1
SPM 1
LDPK 7
; Uses page 7 throughout.

; Compute Degree of Membership for ERROR (E)
LAC EkcH
BGEZ EGEZ
; Case E<0
SBLK e3
SACL TEMPi,1
; Check if it is in the interval (e3,0).
BLZ NE1
; Here TEMPi = 2 * (E - e3) to compensate for RE3b2.
LT RE3b2
; Case e3<E<0
LACK 3
; Interval index J = 3
SAACL JE
B Edone

NE1: LAC Ekh ;Compare E with e2
  SBLK e2
  BLZ NE2 ;Case e2 < E < e3
  LT RE32 ;Interval index J = 2.
  SAACL TEMPi
  LACK 2
  SAACL JE
  B Edone

NE2: LAC Ekh
  SBLK e1
  LT RE21 ;Case e1 < E < e2
  SAACL TEMPi
  LACK 1 ;Interval index J = 1
  SAACL JE
  B Edone

EGEZ: SBLK e5 ;—— Case E > 0
  BLZ POS1 ;Test if e5 < E < e6
  SAACL TEMPi ;Case e5 < E < e6
  LACK 6
  SAACL JE ;Interval index J = 6
  LT RE65
  B Edone

POS1: LAC Ekh
  SBLK e4
  BLZ POS2 ;Test if e4 < E < e5
  SAACL TEMPi ;Yes
  LACK 5
  SAACL JE ;Interval index J = 5
  LT RE54
  B Edone

POS2: LACK 4 ;Case 0 < E < e4
  SAACL JE ;Interval index J = 4
  LT RE4
  LAC Ekh
  SAACL TEMPi

Edone: MPY TEMPi
  PAC ;ACC = ME2, D.M. of right MF for E
  SACH ME2
  LAC DMONE
  SUB ME2 ;ACC = 1 - ME2 = ME1
  SAACL ME1

; --------- Compute DM for Change in Error (CE).
  LAC CE
  SBLK ce2 ;CE > 0 by definition.
  BLZ PCE1 ;Test if ce2 < CE < ce3
  SAACL TEMPi ;Case e5 < E < e6
  LACK 1
  SAACL IC ;Interval index I = 1.
  LT RC32
  B CEdone

PCE1: LAC CE
  SBLK ce1

281
; Test if ce1 < CE < ce2
SACL TEMPi ; Yes
LACK 2
SACL IC ; Interval index I = 2
LT RC21
B CEdone

PCE2: LACK 3 ; Case 0 < CE < ce1
SACL IC ; Interval index I = 3
LT RC1
LAC CE
SACL TEMPi

CEdone: MPY TEMPi
PAC ; ACC = MC2, D.M. of right MF for CE.
SACH MC2
LAC DMONE
SUB MC2 ; ACC = 1 - MC2 = MC1
SACL MC1

; Perform rule base evaluation (Min operator)
LAC ME1 ; MRA = MIN(ME1,MC2)
SACL MRA ; Assume MRA = ME1
SUB MC2 ; ACC = ME1-MC2
BLEZ NXK1 ; Assumption is correct if ACC<=0
LAC MC2 ; No: ME1>MC2. Make MRA=MC2
SACL MRA

NXK1: LAC ME2 ; MRB = MIN(ME2,MC2)
SACL MRB ; Assume MRB = ME2
SUB MC2 ; ACC = ME2-MC2
BLEZ NXK2 ; Assumption is correct if ACC<=0
LAC MC2 ; No: ME2>MC2. Make MRB=MC2
SACL MRB

NXK2: LAC ME1 ; MRC = MIN(ME1,MC1)
SACL MRC ; Assume MRC = ME1
SUB MC1 ; ACC = ME1-MC1
BLEZ NXK3 ; Assumption is correct if ACC<=0
LAC MC1 ; No: ME1>MC1. Make MRC=MC1
SACL MRC

NXK3: LAC ME2 ; MRD = MCN(ME2,MC1)
SACL MRD ; Assume MRD = ME2
SUB MC1 ; ACC = ME2-MC1
BLEZ NXK4 ; Assumption is correct if ACC<=0
LAC MC1 ; No: ME2>MC1. Make MRD=MC1
SACL MRD

NXK4: EQU $ ; Get control signal for each of the 4 fired rules

; Compute offset for look-up table. (IJ)
LAC IC,3 ; ACC = 8*IC
SUB IC ; ACC = 7*IC
ADD JE
SUBK 8 ; ACC = 7*IC+JE-8 = IJ
SACL IJ

ADLK SGTBL ; Points to slip gain table
TBLR DKA ; Read DKA from table
ADDK 1 ; Change pointer to read DKB
TBLR DKB ; ACC = IJ+1

282
ADDK  6 ; Change pointer to read DKC
TBLR  DKC ; ACC = IJ+7
ADDK  1 ; Change pointer to read DKD
TBLR  DKD ; ACC = IJ+8

;-----------------------------------------------------------------------
; ------- DEFUZZIFICATION BY HEIGHT METHOD
; ------- Get sum of truth value of fired rules (MRA+MRB+MRC+MRD)
LAC   MRA
ADD   MRB
ADD   MRC
ADD   MRD
SACL  SUMMR ; SUMMR = MRA+MRB+MRC+MRD

;------- Get sum of products: MRi*DKi, i=A,B,C,D
LT    MRA
MPY   DKA
LTP   MRB ; ACC = MRA*DKA
MPY   DKB
LTA   MRC
MPY   DKC
LTA   MRD
MPY   DKD
APAC  SUMPROD ; SUMPROD = MRA*DKA+....+MRD*DKD

;------- Divide SUMPROD BY SUMMR do get DKs (Fractional division)
; Here, the sign of the result is defined by SUMPROD only
; since SUMMR is by def. always positive.
ZALH  SUMPROD
ABS   ; Make numerator positive
RPTK  14
SUBC  SUMMR ; Result is in low ACC
SACL  DKs

; LAC  SUMPROD
BGEZ  DDONE2 ; Done if sign is positive
ZAC   DKs
SACL  DKs ; Negate quotient if negative

;-----------------------------------------------------------------------
; ------- GET NEW VALUE OF Ks (SLIP GAIN)
; DDONE2:
LDPK  6
LAC   SWT2 ; Introduce a delay after FSG is set,
BZ    PROC
SUBK  1 ; to store rated condition into buffer.
SACL  SWT2 ; Decrement counter SWT2
BGZ   QUIT

PROC:  LDPK  7
LT    GKs
MPY   DKs
PAC
SACH  TEMPi
ZALH  KsH ; Use double precision to compute Ks.
ADD   KsL
ADD   TEMPi ; Add to low ACC since GKs = 0.64 bit.

283
; Imposed min and maximum values for Ks, for safety reasons.
ADDH Ksmax ; Ksmax = 32767 - maxKs (maxKs = 2Kso = 520)
SUBH Ksmax
SUBH Ksmin ; Use saturation features to enforce min Ks.
SUBH Ksmin ; Actually Ksmin = (32768+minKs)/2
ADDH Ksmin ; Currently, minKs = 0.5 Kso = 130.
ADDH Ksmin

SACH KsH
SACL KsL
LDPK 6
SACH KsH
QUIT: RET

; DEFINE THE TABLE FOR OM's FOR Delta Ks
SGTBL: DW dk2, dk3, 0, 0, dk5, dk7, dk8
DW dk1, dk2, dk4, 0, dk5, dk8, dk7
DW dk1, dk1, dk2, 0, dk5, dk8, dk8
Gilberto Costa Drumond Sousa was born in Faria Lemos, MG, Brazil on October 5, 1958. He attended city elementary and middle schools and went to the Federal Technical School of Espirito Santo, Vitoria, ES, finishing his High School studies in 1976. He received a B.S. degree in Electrical Engineering from the Federal University of Espirito Santo, (UFES) in 1981. He joined the faculty of UFES as an Auxiliary Professor in 1982. Then he went to the Federal University of Santa Catarina for his Master studies and received a M.Sc degree in Electrical Engineering in 1986. He returned to UFES as an Assistant Professor and from 1986 to 1989 taught several undergraduate courses at the Electrical Engineering Department. In 1989 he received a four-year scholarship from the National Research Council and joined the University of Tennessee, Knoxville, for his doctoral studies. He received his Ph.D. degree in Electrical Engineering in December 1993.

The author has been involved in research areas including Power Systems and Industrial Controls, and more recently, Fuzzy Logic applications to Power Electronics and Drives. He has published several papers on the areas listed above.