ABSTRACT

LUO, YUKUN. FPGA Based High Bandwidth Motor Emulator for Interior Permanent Magnet Machine Utilizing SiC Power Converter. (Under the direction of Iqbal Husain).

A high bandwidth ($> 20\text{ kHz}$) motor emulator for IPM machine, utilizing FPGA based high fidelity motor model and hybrid model predictive controller (MPC), voltage source power converter, and fast-switching SiC devices, is presented in this dissertation. The bandwidth achieved far exceeds that of existing Motor Emulator (ME) solutions that can only emulate fundamental current and only few orders of harmonic content. The developed ME uses an MPC based control strategy with a unique gate stitching strategy that synchronizes the inverter switching state with the ME switching state for accurate representation of the emulated motor currents in the physical inverter hardware output. The MPC-gate stitching hybrid algorithm avoids the need for excessively high switching frequency of the ME converter. The developed high bandwidth ME can emulate up to the switching ripple current of the Inverter Under Test (IUT) of which the current slope can change up to 6 times within one switching period when using SVM. The FPGA based fast iterating online motor model is another key component which along with the high performance ME current regulation algorithm can accurately emulate the motor current. The high bandwidth also allows the use of a small line inductor which reduces the size and cost of the ME system. The FPGA implementation and control strategy are given in the dissertation. Simulation and experiment results are provided to verify the high bandwidth current emulating capability.

To demonstrate the superiority and flexibility of the motor emulator, advanced motor control techniques are also investigated in this dissertation. Moreover, the details of the motor modeling and FPGA system design methodology are discussed in corresponding chapter.

© Copyright 2021 by Yukun Luo

All Rights Reserved

FPGA Based High Bandwidth Motor Emulator for Interior Permanent Magnet Machine
Utilizing SiC Power Converter

by
Yukun Luo

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Electrical Engineering

Raleigh, North Carolina
2021

APPROVED BY:

Wensong Yu

Srdjan Lukic

Jun Liu

Iqbal Husain
Chair of Advisory Committee

DEDICATION

To my parents

Jianwu Luo

Lin Zhou

BIOGRAPHY

Yukun Luo received the B.S. degree in applied physics from the Huazhong University of Science and Technology, Wuhan, China, in 2015 and the M.S. degree in electrical engineering from the North Carolina State University, Raleigh, NC, USA, in 2017. He is currently working toward the Ph.D. degree with the FREEDM Systems Center, North Carolina State University, Raleigh, NC, USA. His research interests include design of high-performance embedded systems for power electronic systems and electric motor control.

ACKNOWLEDGEMENTS

First of all, I would like to sincerely thank my advisor Dr. Iqbal Husain for his support and guidance. I feel extremely fortunate to have the opportunity to pursue my Ph.D. degree under Dr. Iqbal Husain's supervision. His professional knowledge and wisdom enlightened me and taught me what true academic research means. I feel very proud to be one of his students.

I would also like to acknowledge Dr. Wensong Yu, whose technical insights have been critical in my research. I am very thankful for Dr. Srdjan Lukic and Dr. Jun Liu for their service on my doctoral committee, and their guidance and suggestion on my research.

I would like to thank all my friends in FREEDM systems center: Siyuan Chen, Dakai Wang, Dr. Hao Tu, Siye Cen, Dr. Xinyu Liang, Fei Teng, Luowei Wen, Dr. Hui Yu, Dr. Chi Zhang, Dr. Hao Feng, Dr. Li Yang, Dhrubo Rahman and all the others. You make my Ph.D. life much easier than it would have been without you. In particular, I would like to thank Dr. M A Awal for all the good discussions and collaborations. It is lucky to have you both as a friend and a colleague.

Last but not the least, I would like to thank my parents Jianwu Luo and Lin Zhou, my girlfriend Min Zhang, and my cat Doupi for their unconditional support and love, especially during the difficult time of COVID-19.

TABLE OF CONTENTS

List of Tables	vii
List of Figures	viii
Chapter 1 INTRODUCTION	1
1.1 ME Definition	3
1.2 Motivations	4
1.3 Proposed System Overview	6
Chapter 2 HIGH FIDELITY ELECTRIC MOTOR MODELING	10
2.1 Ideal IPM Model	10
2.2 Non-ideal IPM Model	12
2.3 Real-time Implementation	16
Chapter 3 MOTOR DRIVE CONTROL WITH HIGH FIDELITY MOTOR MODEL . .	21
3.1 Typical Motor Drive Current Controller Structure	21
3.2 Low Speed Optimization: 6th Harmonic Current Suppression	24
3.3 High Speed Optimization: Rotor Position Compensation	26
3.4 PM Flux Parameter Mismatch	27
Chapter 4 HIGH PERFORMANCE ME CURRENT CONTROLLER & GATE STITCH- ING MODULATION SCHEME	29
4.1 ME Current Controller Design	29
4.2 Controller Performance Analysis	35
4.3 Gate Stitching Modulation Scheme	37
4.4 Simulation & Experiment	39
4.4.1 Simulation Result	40
4.4.2 Experimental Result	41
Chapter 5 FPGA DESIGN METHODOLOGY FOR HIGH PERFORMANCE POWER ELECTRONICS APPLICATION	47
5.1 FPGA Platform & Design Toolchain	49
5.2 Design Methodology	51
5.3 Detailed Implementation	57
5.4 Verification & Performance Analysis	60
Chapter 6 SUMMARY & FUTURE WORK	62
6.1 Summary	62
6.2 Future Work	64
References	65

Chapter 7	FPGA CONTROLLER PCB DESIGN TIPS	70
Chapter 8	VARIABLES	73
Chapter 9	ACRONYMS	75

LIST OF TABLES

Table 2.1	Back-EMF FFT result of General Motor IPM machine	13
Table 4.1	ME Simulation and Experiment Parameters	33
Table 5.1	Comparison between different type of FPGA implementation	50
Table 5.2	Comparison of internal AXI interface	51
Table 5.3	Comparison of resource utilization and latency for online motor model block.	52
Table 8.1	A summary of common meteorological variables and their abbrevia- tions in alphabetical order.	73
Table 9.1	A summary of acronyms used in alphabetical order.	75

LIST OF FIGURES

Figure 1.1	Average historical CO2 emission values and adopted CO2 standards for new passenger cars in the EU. All CO2 values refer to New European Driving Cycle (NEDC) measurements (1)	2
Figure 1.2	Variable power factor back-to-back power hardware-in-the-loop testing method for traction inverter. (a) Vector diagram showing the principle. (b) Experimental result of 100 <i>kVar</i> . (c) Topology for the back-to-back testing setup.	3
Figure 1.3	Drivetrain system for electric vehicle in a power hardware-in-the-loop configuration with a motor drive and subsystem to be emulated.	5
Figure 1.4	ME architecture with FPGA implementation overview.	7
Figure 1.5	Electrical topology for ME.	8
Figure 2.1	The measured per phase back-EMF data and its spectrum.	12
Figure 2.2	The digitally reconstructed per phase back-EMF.	13
Figure 2.3	Digitally reconstructed permanent magnet flux in dq synchronous reference frame. Note: the angle range of 0 to 2π is scaled to 12-bit resolution (0 to 4095).	14
Figure 2.4	<i>B</i> of NdFeB magnet according to PM temperature.	15
Figure 2.5	2D look-up tables for inductance according to motor current (a) L_d variation against i_d and i_q . (b) L_q variation against i_d and i_q	15
Figure 2.6	Proposed resistor based PWM voltage edge sensing circuit.	17
Figure 2.7	Proposed diode based PWM voltage edge sensing circuit.	18
Figure 2.8	Comparison between the motor model from PLECS and the motor model implemented in FPGA. (a) Result from ideal motor model implemented in FPGA. (b) Result from native ideal motor model from PLECS.	19
Figure 2.9	Resolver emulation experimental results with zoomed view to indicate the delay.	20
Figure 2.10	The decoded phase angle by the RDC IC on IUT controller.	20
Figure 3.1	PI-based current controller for IPM motor with feedforward decoupling terms.	22
Figure 3.2	Bode plot for the open-loop response with different K_p value.	23
Figure 3.3	Fundamental frequency 100 Hz, i_d^* step change from -100 A to -200 A, i_q^* step change from 250 A to 500 A, (a) K_p with apparent inductance, (b) K_p with incremental inductance.	24
Figure 3.4	Bode plot for the open-loop response with and without the resonant controller component.	25
Figure 3.5	Fundamental frequency 25 Hz, i_d^* holds steady at -10 A, i_q^* hold steady at 25 A: (a) Without resonant controller; (b) with resonant controller.	26

Figure 3.6	Fundamental frequency 1000 Hz, i_d^* step changes from -400 A to -600 A, i_q^* step changes from 75 A to 50 A: (a) Without phase angle compensation; (b) With phase angle compensation.	27
Figure 3.7	Fundamental frequency 1000 Hz, i_d^* step changes from -400 A to -600 A, i_q^* step changes from 75 A to 50 A: (a) With 10% PM flux mismatch; (b) Without PM flux mismatch.	28
Figure 4.1	Equivalent circuit of the system for MPC design.	30
Figure 4.2	ME current controller and modulation architecture.	31
Figure 4.3	Sequential diagram for hybrid MPC implementation.	32
Figure 4.4	Equivalent system model for small signal analysis.	36
Figure 4.5	Compensated open-loop response $G_{OL}(s)$ of the MPC for different parasitic resistance R_l	37
Figure 4.6	Disturbance rejection response $Y_d(s)$ of the MPC for different parasitic resistance R_l	38
Figure 4.7	Disturbance rejection response $Y_{dr} = i_x(s)/v_x(s)$ of the MPC with and without the IUT pulse voltage feedforward.	39
Figure 4.8	Gate stitching modulation scheme.	40
Figure 4.9	Simulation results with 3-phase current and reference under multiple transients and online enable/disable of gate stitching strategy. (a) During 0 to 0.005 sec the fundamental current is controlled to be zero. Step changes in i_d , i_q and motor speed reflected at 0.01 sec. (b) Magnified view during transients of phase current and enable & disable of gate stitching.	41
Figure 4.10	Scope waveform of all four operating point. Current: 10 A/div for (a) and (c); 20 A/div for (b) and (d), voltage: 200 V/div, time: 1 ms/div. CH1, CH3, CH4: 3-phase current, CH5: V_{DS} of ME, CH6: V_{DS} of IUT. (a) 15 A, 200 Hz (3000 rpm). (b) 35 A, 200 Hz (3000 rpm). (c) 15 A, 400 Hz (6000 rpm). (d) 35 A, 400 Hz (6000 rpm).	42
Figure 4.11	Comparison between current reference and current feedback with FFT analysis. (a) 15 A, 200 Hz (3000 rpm). (b) 35 A, 200 Hz (3000 rpm). (c) 15 A, 400 Hz (6000 rpm). (d) 35 A, 400 Hz (6000 rpm).	43
Figure 4.12	ME step change response.	45
Figure 4.13	Within Δt of 253 μs , there are 36 pulses in the V_{DS} of ME SiC device. The active time of the gate stitching is labelled in the waveform. No spike in phase current at the edge of IUT V_{DS} . The time division is 50 μs	46
Figure 5.1	Workflow of proposed FPGA design methodology for rapid prototyping in power electronics.	48
Figure 5.2	Example code snippet with HLS pragma for resource utilization optimization (2).	49
Figure 5.3	Co-simulation setup in Simulink environment, with the source for different blocks labelled.	52

Figure 5.4	The configuration menu in System Generator for DSP can be accessed by double-click the red Xilinx icon in Fig. 5.3. The target device, system constraints, and export preference can be defined in this menu.	53
Figure 5.5	ZYNQ-7000 design block diagram for ME.	54
Figure 5.6	Subsystem sequence and runtime.	55
Figure 5.7	ZYNQ-7000 resource utilization report.	56
Figure 5.8	Placed and routed design in ZYNQ-7000 SoC chip.	58
Figure 7.1	The traces of the four differential signals for Ethernet RJ45 connector with impedance matching to minimize propagation delay among the channels.	70
Figure 7.2	The local ground planes to sandwich the high speed Ethernet signal traces.	71
Figure 7.3	The current sensor layout with appropriate isolation barrier.	71

CHAPTER

1

INTRODUCTION

Nowadays, due to the worsened air pollution and more extreme global climate change, countries around the world are pushing to phase out pure internal combustion engine (ICE) vehicles. China, France, Germany, UK and California state in US have all announced national or state level plans to strictly limit the CO₂ emission by passenger vehicle or even ban ICE vehicle new car sales and manufacturing within the coming decades. The Figure 1.1 presents the adopted CO₂ standards for new passenger cars in the EU (1). This kind of high level policy change means great opportunities and challenges for automobile manufacturers around the globe in this electrification era. However, the performance bottle neck of the chemical battery means battery electric vehicle (BEV) still cannot replace ICE vehicle under all circumstances. Therefore, there are many other electrification solutions like HEV (Hybrid EV), PHEV (Plug-in Hybrid EV) and HFCV (Hydrogen Fuel Cell Vehicle). Among the most popular category (HEV/PHEV), depending on the position and power rating of electric motor, there are 5 sub-categories. P0: The electric motor is connected with the internal combustion engine through a belt, on the front end accessory drive (15-30 kW, 48-300 V). P1: The electric motor is connected directly with the crankshaft between the internal combustion engine and the transmission (15-30 kW, 48-300 V). P2: Electric motor located between engine and transmission with additional clutch to enable pure

Figure 1.1: Average historical CO₂ emission values and adopted CO₂ standards for new passenger cars in the EU. All CO₂ values refer to New European Driving Cycle (NEDC) measurements (1)

electric drive mode (15-80 kW, 48-300 V). P3: Electric motor located after the output of transmission (15-50 kW, 48-300 V). P4: Electric motor integrated into secondary axle which is not driven by the engine (15-100 kW, 48-300 V). As shown above, with so many technical vehicle electrification solutions, the electric motor's role in the vehicle drivetrain system is more and more crucial.

In the development process of electric drivetrain for hybrid and electric vehicles, hardware-in-the-loop (HIL) test, R-L load test, and dyno test bench are widely used for design validation and testing. However, all of the above testing methods have their respective shortcomings. The signal level HIL testbed take the gate signals from the motor controller as inputs and then generate phase current and resolver angle signal as outputs. The motor and inverter power stage is simulated by the model in the HIL equipment. Although this method is very flexible, the complicated non-ideal and parasitic characteristics of the motor drive power stage are not taken into consideration. The R-L load test with a physical inverter and power supply can introduce actual power flow in the system. However, high power and variable power factor test require large power supply and different combinations of resistor

Figure 1.2: Variable power factor back-to-back power hardware-in-the-loop testing method for traction inverter. (a) Vector diagram showing the principle. (b) Experimental result of 100 $kVar$. (c) Topology for the back-to-back testing setup.

and inductor. A variable power factor back-to-back inverter test method has been proposed earlier, as depicted in Figure 1.2, but the method doesn't go far enough to accurately emulate the motor currents in an EV powertrain where the non-linearities of the IUT and motor are not taken in account (3). Although the dyno test bench is the best set-up to evaluate the real operation condition of motor drive system, there are still some drawbacks. First, for a dyno set-up, all the drivetrain components should be physically built to set the testbed. Secondly, the the speed dynamics of the dyno set-up maybe limited by the high inertia of the dyno.

1.1 ME Definition

The power hardware-in-the-loop (PHIL) setup with increasing adoption in power and energy area (4) would be suitable for motor emulator (ME) application. Photovoltaic (PV)

inverter system emulators using PHIL setup have been reported in (5; 6). While in (7), a battery emulator utilizing PHIL setup is presented. ME which combines the advantages of signal level HIL testbed and full-scale dyno setup is an efficient, cost-effective, and rapid testing method for evaluating a motor drive system. The ME should be capable of simulating the terminal electrical characteristics of an actual motor. Since the motor model is embedded in the software, ME can emulate any type of motor, such as IM (8; 9), SRM (10), axial flux SPM (11) and PMSM (12; 13; 14), under any operating condition in a more controlled manner. A megawatt-scale ME has been reported in (15). However, the available ME reported in the literature so far are of limited current control bandwidth below 1.5 *kHz* which does not replicate the actual motor currents along with their high frequency components (8; 9; 10; 11; 12; 13; 14; 16; 17; 18; 19). Since the majority of the EV traction inverters are running at a switching frequency between 5 *kHz* and 10 *kHz*, low bandwidth ME could only emulate fundamental frequency current. The inability to emulate harmonic current introduced by device switching and electric machine nonlinear parameters means these low bandwidth MEs can only be used for inverter validation, rather than testing powertrain control algorithms and motor design. The bandwidth limitation of conventional ME mainly come from the relatively low switching frequency of IGBT devices (around 20 *kHz*) and an inaccurate motor model due to slow iteration of differential equations that describe the electrical and mechanical subsystems of electric motor. The need for a higher bandwidth ME is even more critical now as SiC traction inverters with switching frequency exceeding 20 *kHz* and electric machine speeds exceeding 18,000 *rpm* are being introduced by the automotive manufacturers.

A complete electric vehicle drivetrain system is shown in Figure 1.3 which consists of a motor drive inverter, a motor and transmission system, and the vehicle load model. We propose a power hardware-in-the-loop (PHIL) system which can emulate the real world load for an electric vehicle motor drive inverter. The load profile of the inverter is determined by the electric motor and vehicle model.

1.2 Motivations

With the availability of high frequency and high power rated SiC type WBG devices and powerful FPGA control platform, fast switching and high bandwidth power electronics converters such as inverter (20) and rectifier (21) have been reported. Therefore, it is possible to replace the IGBT in traditional ME with fast switching SiC MOSFETs. However, the real challenge of

Figure 1.3: Drivetrain system for electric vehicle in a power hardware-in-the-loop configuration with a motor drive and subsystem to be emulated.

a high bandwidth ME is that the high frequency switching ripple current of a three-phase VSI motor drive contains abundant harmonics content (22). Thus, even with the relatively fast switching SiC MOSFETs, the traditional control method would not necessarily yield good results, especially considering the trend of increasing motor drive switching frequency and motor fundamental frequency. In this dissertation, we report a customized hybrid model predictive control method and a novel modulation scheme called gate stitching implemented using FPGA and SiC converter to advance the ME technology. This integrated control and modulation strategy improves the system control bandwidth without the need for excessively high switching frequency from the ME power stage. It is worth noting that the gate stitching concept is first introduced in our previous work (23), but it was not tested experimentally.

A high bandwidth ME concept has been proposed earlier in (24), but the use of a DSP based controller constrained the iteration speed of the motor model at high speeds which limited its accuracy. Additionally, the low switching frequency for the IUT and large line inductance used pose limitations for its use in a real world application. The topology and control scheme have also not been revealed in (24).

In this dissertation, a faster switching SiC based high control bandwidth and rapid FPGA based motor model iteration ME system is presented. The gate stitching modulation strategy, proposed in this dissertation, to synchronize the switching states of the IUT and the ME to avoid current spikes in the system is the key to the high control bandwidth. The contributions of this dissertation are as follows: (1) The ME system topology, MPC based current controller, and discrete-time model formulations are presented in detail; (2) a new gate stitching modulation strategy has been proposed and experimentally verified; (3) the performance of the MPC controller have been evaluated for ME current regulation both in simulation and experiment; (4) FPGA design methodology for high performance power electronics application with short development time has been illustrated.

1.3 Proposed System Overview

The detailed system architecture of the proposed emulator separated between the IUT and the ME is shown in Figure 1.4. Ideally, the ME should operate like a VSI in the current control mode. The current command for the ME is generated by the embedded fast iterating motor model, which takes the PWM terminal voltage of IUT as input. Also, the electric torque output from the motor model combined with the vehicle powertrain mechanical model can be used to calculate the speed and angle position of the motor. The position information is then transmitted to the IUT controller for FOC control via resolver emulation circuit. One of the key criteria for a well designed ME testbed is that the ME setup should be non-intrusive, which means the connection between IUT and ME should be limited to three phase power connection and position sensor interface. In this case, the position sensor for the IUT is the resolver. Although, the ME presented in (25) uses PLL to sense the phase angle, it is not suitable for traction motor drive application. It is worth noting that since both the IUT and the ME are trying to control the current in the system, to ensure stability the control bandwidth of one subsystem (ME) should be much higher than the other subsystem (IUT) (26). Although, both IUT and ME are controlling the current in the system, the current commands from IUT ($i_{d,IUT}^*$ and $i_{q,IUT}^*$) are only the average values. While for the ME, the current commands ($i_{d,ME}^*$ and $i_{q,ME}^*$) are generated by the motor model in (1), they would contain harmonics components caused by the switching of IUT.

As depicted in Figure 1.4, the ME testbed contain three major parts.

Figure 1.4: ME architecture with FPGA implementation overview.

Power Stage & Sensor Solution

The coupling network between the IUT and ME can be L or LCL; a comparison between the two type of networks appear in (27). In this dissertation, the L interface is selected for the system due to the fact that LCL network require more complicated controller structure which could reduce the system emulation bandwidth. Also, the LCL resonance would deteriorate system stability. The inductor value in the dissertation is set to be small to achieve high system dynamic thanks to the high bandwidth current controller. It is worth noting that the inductance of the target motor should only be larger or equal to the actual line inductance, shown as L_l in Figure 1.4. Several different power converter topologies are available to implement the power stage of ME. Two-level six-switch converter (12; 13), modular multiphase multilevel converter with special modulation scheme (14; 28), interleaving converter with special inductor design (16; 29), or even modular-multilevel converter (MMC) (8) are among the options. In this dissertation, the conventional two-level VSI topology, shown in Figure 1.5, is used for the ME implementation and validation of the novel gate stitching strategy which can synchronize the switching states of IUT and

Figure 1.5: Electrical topology for ME.

ME. The two isolated DC source for IUT and ME provide the opportunity to emulate the high frequency ripple current when compared to the shared single DC source solution. The zero sequence current in the shared DC bus topology (30; 31) could compromise the ripple current tracking accuracy. Also two separate DC source enable the adjustment of ME DC bus voltage to achieve larger operation range for target motor (24). It can be seen later section VI that even with the most basic topology, the novel gate stitching strategy, together with the model predictive control, can ensure a good current tracking capability.

To ensure high emulation fidelity, the sensors' performance for the ME is crucial. The IUT PWM voltage sensing is done in two steps. The first is to sense the DC bus voltage of IUT, which is of relatively slow dynamics, using conventional resistor divider based DC voltage sensing solution. The bandwidth and delay of the DC bus voltage sensing is 100 kHz and 3.7 μs respectively. The second step is to sense the logic level of PWM voltage, which has high dynamics and requires dedicated comparator based circuit, shown in Fig. ???. The isolation between the comparator output and the FPGA pins is done by the optical fiber. The FPGA can sample the logic level of PWM voltage at 200 MHz, while the total delay is 64 ns. On the other hand, the current sensor for the ME is decisive for the current emulation error. In this application, the ACS732 series automotive grade galvanically isolated current sensor IC from Allegro is selected. The particular IC provides 1 MHz bandwidth and only 200 ns of propagation delay. Although the shunt based current sensor could provide theoretically higher bandwidth, but the additional circuits needed for conditioning and isolation would deteriorate the performance with respect to both bandwidth and propagation delay, and hence, complicate the design.

Re-configurable Real-time Motor Model

The re-configurable real-time motor model is the key component to the proposed emulation platform. The model can be of different kind of motor, but in this dissertation we are focusing on the IPM machine model, which will be elaborated in Chapter 2. The motor model will use the PWM voltage from IUT as input, and then generate the current reference output to the current controller of ME. To ensure modelling accuracy, the iteration speed of the motor model should be very fast. Therefore, powerful ZYNQ-7000 SoC (System-on-Chip) platform from Xilinx, which contains FPGA and ARM processor, is selected for the application. In this dissertation, 1 MHz iteration frequency for a high fidelity motor model is achieved.

ME High-Performance Current Controller

For ME system the control objective is to track the current reference generated by the embedded electric motor model with minimum error. Since the current reference contains switching harmonic content caused by the 10 kHz switching of three-phase VSI, within one switching period the applied switching states can change 6 times resulting in variable frequency and amplitude ripple current. Dead-beat controller (14), PI controller (19) and optimal controller (32) are reported to be used in the PHIL application. However, the performance of those controllers are not ideal for the high frequency ripple current control requirement. Model predictive control (MPC) is selected for this application due to the high current tracking accuracy and fast dynamic response requirement. The high computation burden of the MPC is relieved by the simple coupling network (pure L) and the powerful and abundant FPGA computation resource. One of the main constraint of the system is the switching frequency of ME. Even with advanced wide band-gap SiC power devices, the switching frequency cannot be arbitrarily high. Novel PWM techniques are proposed in (33; 34; 35) with different optimization goals for a motor drive. Especially in (35), since the topology of the proposed ME is similar to an open-end winding motor drive. In this paper, a new gate stitching modulation technique which synchronize the switching states of IUT and ME is proposed to enhance the current tracking accuracy during the switching transient of IUT and optimize device switching frequency. The synchronization of the gate stitching strategy is essentially utilizing the zero voltage vector in open-end winding motor drive system to suppress uncontrolled current spike during the transient of $S_{x,IUT}$. The in depth illustration of the current controller design and modulation scheme is covered in section IV.

CHAPTER

2

HIGH FIDELITY ELECTRIC MOTOR MODELING

2.1 Ideal IPM Model

The ideal Interior Permanent Magnet (IPM) machine model in dq synchronous reference frame can be described with electrical and mechanical equations in (2.1) and (2.2).

$$\left. \begin{aligned} v_{qs} &= r_s i_q + \frac{d}{dt} \lambda_{qs} + \omega_e \lambda_{ds}, \\ v_{ds} &= r_s i_d + \frac{d}{dt} \lambda_{ds} - \omega_e \lambda_{qs}, \\ \lambda_{qs} &= L_q i_q, \\ \lambda_{ds} &= L_d i_d + \lambda_{pm} \end{aligned} \right\} \quad (2.1)$$

$$\left. \begin{aligned} T_e &= \frac{3}{2} \cdot \frac{P}{2} [\lambda_{PM} \cdot i_q + (L_d - L_q) i_d i_q], \\ T_e &= T_L + \omega_m B + J \cdot \frac{d\omega_m}{dt}, \\ \theta_e &= \int_t \omega_e dt, \\ \omega_e &= \omega_m \cdot \frac{P}{2} \end{aligned} \right\} \quad (2.2)$$

Here, λ , ω_m , ω_e , B , P , T_L and J represent flux linkage, rotor mechanical speed, rotor electrical speed, viscosity coefficient, pole number, load torque and rotor moment of inertia, respectively. The inputs for the model is q-axis and d-axis voltage (v_{qs} and v_{ds}), which can be obtained by passing the three-phase voltage, applied at the terminal of the motor, through abc-dq transformation. The outputs for the electrical subsystem is q-axis and d-axis current (i_q and i_d). While the outputs for the mechanical subsystem is the electrical torque (T_e) and rotor speed/position (ω_m and θ_m). It is worth mentioning that for digital real-time implementation of the above motor model, (2.1) and (2.2) should be discretized and rearranged to express current and rotor speed/position explicitly. Here, the forward Euler discretization method is used, the new equations ready for digital implementation is shown in (2.3) and (2.4).

$$\left. \begin{aligned} \lambda_{qs,k+1} &= \lambda_{qs,k} + (v_{qs,k} - r_s i_{q,k} - \omega_{r,k} \lambda_{ds,k}) \cdot T_s, \\ \lambda_{ds,k+1} &= \lambda_{ds,k} + (v_{ds,k} - r_s i_{d,k} + \omega_{r,k} \lambda_{qs,k}) \cdot T_s, \\ i_{q,k+1} &= \frac{\lambda_{qs,k+1}}{L_q}, \\ i_{d,k+1} &= \frac{\lambda_{ds,k+1} - \lambda_{PM}}{L_d} \end{aligned} \right\} \quad (2.3)$$

$$\left. \begin{aligned} T_{e,k} &= \frac{3}{2} \cdot \frac{P}{2} [\lambda_{PM} \cdot i_{q,k} + (L_d - L_q) i_{d,k} i_{q,k}], \\ \omega_{m,k+1} &= \omega_{m,k} + \frac{T_{e,k} - T_L - \omega_m B}{J} \cdot T_s, \\ \theta_{e,k+1} &= \theta_{e,k} + \omega_{e,k} \cdot T_s, \\ \omega_e &= \omega_m \cdot \frac{P}{2} \end{aligned} \right\} \quad (2.4)$$

With (2.3) and (2.4), current in dq reference frame is obtained. If abc stationary reference frame current is needed, additional abc-dq transformation is required. Even though the motor model in (2.3) and (2.4) do not include any non-linearity, cross coupling effect,

spacial harmonics and temperature induced parameter variation, it is sufficient to test the non-linearity of the power stage caused by the switching of IUT. In other word, if the input to the model is the PWM voltage rather than the average voltage, the motor current calculated by the model would contain switching harmonic. Therefore the fidelity of the model is determined by the iteration speed of (2.3) and (2.4) and the edge caption accuracy of the IUT PWM voltage. In this work, we have achieved 1 *MHz* iteration speed of the motor model and designed an accurate PWM voltage detection circuit. How to incorporate the non-linearity, cross coupling, spacial harmonics and temperature effect (36) into the motor model will be covered in the next section.

Figure 2.1: The measured per phase back-EMF data and its spectrum.

2.2 Non-ideal IPM Model

Normally in a real IPM machine, the sources of non-linearity are as follows: spatial harmonics of back-EMF, saturation of back-EMF due to temperature and high fundamental

Figure 2.2: The digitally reconstructed per phase back-EMF.

Table 2.1: Back-EMF FFT result of General Motor IPM machine

Harmonics Order	Amplitude
Fundamental	5.82 V
5th	0.277 V
7th	0.1682 V
11th	0.3057 V
13th	0.08087 V
17th	0.06285 V
19th	0.05037 V

frequency, and saturation, cross-coupling, and spatial harmonics of d-axis and q-axis inductance. In this work, the non-ideal IPM machine model is constructed using the measured experimental data of a 100 kw General Motors IPM motor in the lab.

To model the spatial harmonics of back-EMF, first the open circuit test of the motor should be conducted to measure the motor terminal voltage when the motor is spinning at constant speed. This way, the per phase back-EMF data can be obtained. Next, with the measured back-EMF data, FFT (Fast Fourier Transformation) can be conducted in MATLAB to extract the harmonics information. Thus, the back-EMF could be reconstructed digitally in the FPGA. The measured back-EMF data and FFT result are shown in Figure 2.1 and Table 2.1. While the digitally reconstructed back-EMF is presented in Figure 2.2. It is worth noting that the IPM machine is modeled in dq synchronous reference frame. And back-EMF voltage need to be converted into permanent magnet flux by dividing the electrical angular speed. Therefore, there is an additional step to convert the three-phase back-EMF data into

Figure 2.3: Digitally reconstructed permanent magnet flux in dq synchronous reference frame. Note: the angle range of 0 to 2π is scaled to 12-bit resolution (0 to 4095).

PM flux data in dq synchronous reference frame. This conversion can be conducted offline in MATLAB. The result is shown in Figure 2.3. Above is how to model the spatial harmonics effect of the back-EMF. For implementation in FPGA, two look-up tables (LUT) containing the PM flux data for each axis are established. The size of the LUTs is 4096 to match the 12-bit resolution of the rotor position defined by the resolver position sensor.

The next step is to model the saturation of permanent magnet flux due to high temperature and high speed. In general, the magnetic flux density B of magnet would decrease as temperature rises. In Figure 2.4, the curve of magnetic flux density against temperature for NdFeB magnet is presented. As we can see, the variation of B according to temperature is quite slow, and close to linear. Therefore, this effect can be easily modeled with a coefficient. Moreover, under high fundamental frequency, the PM flux could also saturate. And this saturation effect could be modeled with an coefficient, too. All in all, in FPGA, one coefficient could model both saturation effects at high PM temperature and/or high fundamental frequency.

The final step is to model the non-linearities of d-axis and q-axis inductance, which

Figure 2.4: B of NdFeB magnet according to PM temperature.

includes saturation, cross coupling, and spatial harmonics. As can be seen, the relationship between inductance and all those variables are very complicated. And it is difficult to come up with analytical expressions of inductance in this non-ideal model, let alone compute them online. Therefore, the most efficient way to model the motor inductance is to establish multi-dimensional LUTs to obtain inductance directly from magnetic flux λ . The inductance data in the LUTs can be derived from the magnetic flux data under different operation points. Those magnetic flux data may come from experimental measurement or FEA (Finite Element Analysis) simulation.

Figure 2.5: 2D look-up tables for inductance according to motor current (a) L_d variation against i_d and i_q . (b) L_q variation against i_d and i_q .

The dimension of the LUTs are determined by the number of non-linear effects to consider. If all the non-linearity, like saturation, cross coupling, spatial harmonics, and temperature effect, are to be included, then the LUTs would be four dimensional. Furthermore, two sets of LUTs are needed. One set for apparent inductance (λ/i) in d-axis and q-axis. Another set for incremental inductance ($d\lambda/di$) in d-axis and q-axis. Since different inductance is needed in different places. For instance, the inductance output from LUTs to be use in motor model (2.3) should be apparent. While in the PI current controller design, the incremental inductance should be used to calculate the proportional parameters.

In this work, the dimension of the inductance LUTs is two, since the original magnetic flux data provided by General Motors only contains the variation of magnetic flux against dq current. After derivation, interpolation, and extrapolation, the inductance LUTs are presented as in Figure 2.5. The size of the 2D LUTs are 89x89. The step of the current in the LUTs is 8 A. This way the LUTs can cover the operation of the motor up to 720 A, which matches the current rating of the machine. For FPGA implementation, the reciprocal of the inductance, rather than the actual inductance, is stored in the LUTs to optimize the execution time of the motor model. The expression of the fully modeled non-ideal IPM motor model is presented in (2.5), where T_s , ω_e , and $K(\omega_e, \tau)$ stand for sampling period, electric angular speed, and coefficient of PM flux saturation due to high fundamental frequency and temperature, respectively.

$$\left. \begin{aligned} \lambda_{ds,k+1} &= \lambda_{ds,k} + T_s \cdot (v_{ds,k+1} - r_s i_{d,k} + \omega_e \lambda_{qs,k}), \\ \lambda_{qs,k+1} &= \lambda_{qs,k} + T_s \cdot (v_{qs,k+1} - r_s i_{q,k} + \omega_e \lambda_{ds,k}), \\ i_{q,k+1} &= \frac{\lambda_{qs,k+1} - k(\omega_e, \tau) \cdot \lambda_{pmh,q,k}(\theta_e)}{L_{q,k,app}}, \\ i_{d,k+1} &= \frac{\lambda_{ds,k+1} - k(\omega_e, \tau) \cdot \lambda_{pmh,d,k}(\theta_e)}{L_{d,k,app}}, \\ \frac{1}{L_{q,k,app}} &= LUT \{[|i_{d,k}|], [|i_{q,k}|]\}, \\ \frac{1}{L_{d,k,app}} &= LUT \{[|i_{d,k}|], [|i_{q,k}|]\} \end{aligned} \right\} \quad (2.5)$$

2.3 Real-time Implementation

As mentioned before, to achieve high fidelity, the motor model should be iterate at very high speed. Therefore, high performance FPGA digital control platform is selected for this

Figure 2.6: Proposed resistor based PWM voltage edge sensing circuit.

work. The Xilinx ZYNQ-7000 SoM board used in this application integrates the software programmability of dual-core ARM Cortex-A9 processors with the hardware programmability of FPGA. This solution provides better performance and simplicity over the DSP plus CPLD structure. The simulation results of the FPGA implemented motor model are shown in Figure 2.8. The detailed FPGA based system implementation and FPGA design methodology are covered in Chapter 5.

Now, with the computation issue of (2.3) and (2.4) solved, we need to find a way to accurately sense the edges of the PWM voltage output of IUT. The designed circuit for fast and accurate PWM voltage sensing is shown in Figure 2.6. The circuit consists of a voltage divider resistor network and a comparator, the output of the comparator represent the logic level of the PWM voltage. Each phase needs a set of detection circuit. A few design tips. First, the higher the comparator supply voltage (V_{cc}), the better the noise immunity. However, the comparator with high supply voltage (V_{cc}) usually has long propagation delay. So to achieve balanced performance, a comparator with 12 V supply voltage and 4 ns propagation delay is selected. Second, between the output of the comparator and the FPGA pin, there should be some kind of isolation. Therefore, a small print circuit board containing the PWM detection circuit is fabricated and mounted close to IUT. The transmission of the detected voltage logic level to the FPGA is done through optic fiber. The propagation

Figure 2.7: Proposed diode based PWM voltage edge sensing circuit.

delay introduced by the optic fiber link is 60 ns. With total propagation delay of 64 ns, the performance is sufficient for ME application. With the logic level of the PWM voltage accurately sensed, the only missing information is the DC bus voltage of the IUT. Since the dynamic of the DC bus voltage is usually quite slow, a regular DC voltage sensing circuit and analog-to-digital conversion (ADC) should be adequate. However, the drawback of the proposed PWM voltage sensing circuit is quite obvious. With certain voltage divider resistor value and comparator reference voltage, the detectable range of the DC bus voltage is limited. If the DC bus voltage of the IUT changes a lot, then this circuit is not effective. In Figure 2.7, a diode based PWM edge sensing circuit, which has wide detectable DC bus voltage, was proposed and tested. However, due to the inevitable parasitic of the diode, the circuit generate a lot of noise and could not capture the PWM voltage edge accurately. Fortunately, for this application, the DC bus voltage of the IUT is varying in a very small range. Thus, the first resistor based circuit is selected.

$$\left. \begin{aligned} RR &= V_p \times \sin(\omega t), \\ SS13 &= V_s \times \sin(\omega t) \times \sin(\theta_e), \\ SS24 &= V_s \times \sin(\omega t) \times \cos(\theta_e) \end{aligned} \right\} \quad (2.6)$$

One of the key criteria for a well designed ME testbed is that the ME setup should be

Figure 2.8: Comparison between the motor model from PLECS and the motor model implemented in FPGA. (a) Result from ideal motor model implemented in FPGA. (b) Result from native ideal motor model from PLECS.

non-intrusive, which means the connection between IUT and ME should be limited to three phase power connection and position sensor interface. In this case, the position sensor for the IUT is the resolver. Therefore, a resolver emulation circuit is required to interact with the Resolver-to-Digital Converter (RDC) IC on the control board of IUT. The resolver contains three sets of winding, one rotating primary winding and two set of secondary winding perpendicular to each other. To interact with the RDC chip, one channel of ADC is required to sense the excitation signal RR (usually a 10 kHz sinusoidal wave) and send the information to FPGA. A computation core inside FPGA is needed to encode the excitation signal with the phase angle (calculated by the motor model (2.3) and (2.4)) and generate the amplitude modulated SS13 and SS24 signal, the FPGA computation core design is covered in Chapter 5. The relationship between excitation signal (RR) and feedback signal (SS13 and SS24) are described in (2.6). Finally, two channels of DAC are needed to convert the SS13 and SS24 into analog signal for the RDC IC to decode. It is worth pointing out that the

Figure 2.9: Resolver emulation experimental results with zoomed view to indicate the delay.

whole resolver emulation operation is also running at 1 MHz. The total delay of this signal chain is $3 \mu\text{s}$, which means the feedback signal (SS13, SS24) has 10.8 degree phase shift compared to the 10 kHz excitation signal RR. This delay is well within the phase shift limit of the RDC IC. The experiment results to demonstrate the resolver emulation circuit low latency are presented in Figure 2.9. While in Figure 2.10, the phase angle decoded by the RDC IC from IUT controller is presented, the phase angle is very stable with no oscillation observed.

Figure 2.10: The decoded phase angle by the RDC IC on IUT controller.

CHAPTER

3

MOTOR DRIVE CONTROL WITH HIGH FIDELITY MOTOR MODEL

With the high fidelity non-ideal motor model mentioned in last chapter, more advanced motor controller could be designed and validated in simulation environment without the presence of physical machine. In this chapter, we address the issues of conventional motor drive current controller at low speed and high speed region and examine the impact of parameters mismatch on the controller performance.

3.1 Typical Motor Drive Current Controller Structure

The typical PI-based current controller for IPM motor is depicted in Figure 3.1. The feed-forward decoupling terms are very crucial to the performance of current regulator during dynamic transient. Without the decoupling terms, according to the complex vector frequency response function in (37), the system close loop gain is not longer unity when the frequency deviates away from the synchronous frequency. It is worth noting there is an equivalent alternative to this feedforward decoupling approach. Instead of adding

Figure 3.1: PI-based current controller for IPM motor with feedforward decoupling terms.

decoupling terms to the output of PI controller, they can be added at the input of the integral controller. This method is known as the complex vector PI controller (37). The merit of the complex vector PI current controller is less dependence on system parameters compared with the feedforward method. However, in this work, for simplicity, the more straightforward feedforward structure is selected.

With an ideal motor model, the parameters for the PI controller are usually fixed. However, in practice, the motor parameters are highly variable. To achieve good current regulation performance, the controller parameters should be adaptive according to the operating points.

$$K_p^{dq} = \frac{L_{inc}^{dq}(i_d, i_q)}{\tau_{cc}}; \quad K_i^{dq} = \frac{r_s}{\tau_{cc}}; \quad \tau_{cc} = 333\mu s \quad (3.1)$$

$$G_{OL}(s) = \frac{0.5V_{dc}F_{zoh}(s)e^{-sT_s}}{sL_{dq} + r_s} \left[K_p^{dq} + \frac{T_s K_i e^{-sT_s}}{1 - e^{-sT_s}} \right]; \quad F_{zoh}(s) = (1 - e^{-sT_s})/(sT_s); \quad (3.2)$$

The controller parameters can be obtained from motor inductance as in Eq. 3.1, which can be found in the inductance LUTs in Chapter 2. For the K_p calculation, incremental inductance should be used, since the proportional part should address the dynamics of

Figure 3.2: Bode plot for the open-loop response with different K_p value.

the motor current (di/dt). While for the calculation of the feedforward terms, apparent inductance should be used to compute the flux values ($\lambda = L \cdot i$). The impact of using apparent inductance in controller K_p parameters design is that the K_p value would become larger than ideal. The overestimated K_p results in significantly higher gain at higher frequency and reduced phase margin. This causes more pronounced oscillation during step change transient. In Figure 3.2, the open-loop response $G_{OL}(s)$ in Eq. 3.2 of the system with different K_p values are presented, with maximum of 400% mismatch between apparent and incremental inductance during transient. It can be observed that with K_p calculated from apparent inductance, plant is not fully cancelled, resulting in gain increase at high frequency without phase margin improvement.

To further validate the point, PLECS simulation with two sets of K_p values are carried out, as shown in Figure 3.3. With the apparent inductance, the steady state performance is similar but during transient, the ringing and overshoot are quite unsatisfactory. With incremental inductance, the settling time and overshoot improve by 76% and 67%, respectively. Although, this controller structure is demonstrated to be very effective even under high dynamic

Figure 3.3: Fundamental frequency 100 Hz, i_d^* step change from -100 A to -200 A, i_q^* step change from 250 A to 500 A, (a) K_p with apparent inductance, (b) K_p with incremental inductance.

transient. Special treatments are still needed for optimization at very low or high motor speed.

3.2 Low Speed Optimization: 6th Harmonic Current Suppression

Due to the inclusion of permanent magnet flux spatial harmonics and saturation in the IPM motor model, lower order harmonic current, especially the 6th order, gets included in the motor current. The harmonics in the motor current cause additional torque ripple and NVH issues. Since at high speeds the torque ripple cannot be filtered out by the inertia of the mechanical subsystem, it is important to address this issue in the low speed operating region up to few tens of Hz (38). Resonant controller can be added in parallel with the PI controller to suppress the 6th order harmonic current caused by the permanent flux spatial harmonic of interior permanent magnet machine. It is worth noting that for digital implementation, the resonant controller should be modified as in Eq. 3.3. This modification is equivalent to adding a low pass filter to the ideal resonant controller to reduce the infinite peak gain and spread out the spectrum. The open-loop transfer function of the system with the resonant controller could be written as in Eq. 3.4. $G_C^R(s)$ is the continuous representation of $G_C^R(z)$. The comparison of the open-loop response with and without resonant controller is presented in Figure 3.4. It can be seen that, with the addition of resonant controller, the

Figure 3.4: Bode plot for the open-loop response with and without the resonant controller component.

gain at resonant frequency is significantly improved with price of slightly dropped gain at lower frequency.

$$G_R(s) = \frac{K_r \omega_B s}{s^2 + \omega_B s + \omega_r^2} \quad (3.3)$$

$$G_{OL}^R(s) = \frac{0.5 V_{dc} F_{zoh}(s) e^{-sT_s}}{s L_{dq} + R_s} \left[K_p^{dq} + \frac{T_s K_i e^{-sT_s}}{1 - e^{-sT_s}} + G_C^R(s) \right]; G_C^R(z) = \frac{T_s \omega_B K_r z(z-1)}{((z-1)T_s)^2 + \omega_B z(z-1)T_s + (z\omega_r)^2} \quad (3.4)$$

To further prove the point, PLECS simulation with and without the resonant controller are carried out, as shown in Figure 3.5. Under low fundamental frequency of 25 Hz and very low motor current, the sixth order harmonic current is quite dominant in the dq -reference frame. This mimic the operation of electric at low speed with little acceleration. With the

Figure 3.5: Fundamental frequency 25 Hz, i_d^* holds steady at -10 A, i_q^* hold steady at 25 A: (a) Without resonant controller; (b) with resonant controller.

addition of resonant controller, the sixth harmonic is suppressed significantly. The current waveform THD decrease from 10.9% to 7.4% with the help of resonant component of the controller. The sixth harmonic content of i_d reduces from 3.3 A to 0.16 A with the addition of resonant controller. At higher fundamental frequency and higher motor current operating region, the resonant part of the controller can be disabled, since the effect of 6th order harmonic current would be less noticeable.

3.3 High Speed Optimization: Rotor Position Compensation

For most of the AC motor drives, the current control is implemented digitally in DSP or MCU. Due to the sequential nature of the processor, digital execution delay is inevitable. The delay between sampling and duty cycle application is one sampling time-period T_s . The PWM modulation would introduce an additional half of T_s delay, and therefore, a total of $1.5 T_s$ delay exists in digitally controlled PWM motor drive system. There are methods to compensate for this delay on the controller side.

$$K_{comp} = K(\omega_e, T_s) \cdot e^{j(1.5T_s\omega_e)}; K(\omega_e, T_s) = \frac{2}{T_s\omega_e} \sin\left(\frac{T_s\omega_e}{2}\right); \quad (3.5)$$

However, this delay could also affect the $dq - abc$ transformation, which is usually ignored at lower speed. Conventionally, the sampling of the phase angle is at the very beginning of the control period, adjacent to the A/D conversion for current/voltage. By the

Figure 3.6: Fundamental frequency 1000 Hz, i_d^* step changes from -400 A to -600 A, i_q^* step changes from 75 A to 50 A: (a) Without phase angle compensation; (b) With phase angle compensation.

time the controller output is applied to the power stage at the beginning of next control cycle, the reference frame has rotated an additional angle of $1.5 \cdot T_s \cdot \omega_e$. The angle mismatch would become significant under a high fundamental frequency. The compensation for the phase angle can be done as in Eq. 3.5 (39) where this term is applied to the voltage vector in $\alpha\beta$ reference frame. The simulation results with and without this compensation technique are compared in Figure 3.6. With the phase angle compensation, the operation is stabilized under a high fundamental frequency.

3.4 PM Flux Parameter Mismatch

As illustrated in Chapter 2, the permanent magnet flux could saturate at high fundamental frequency and high temperature. The saturation caused by high fundamental frequency can be easily accounted for in the current controller design, since the motor speed can be measured accurately via the resolver position sensor. However, the saturation due to temperature is difficult to predict, due to the inability to directly measure the permanent magnet temperature. Therefore, the effect of the permanent magnet flux mismatch on current controller performance is analyzed.

In Figure 3.7, the current control performance with and without permanent magnet flux mismatch is compared. The fundamental frequency is set to be 1000 Hz. i_d^* step changes from -400 A to -600 A, while i_q^* step changes from 75 A to 50 A. In Figure 3.7(a), there is 10% mismatch between the permanent magnet flux used in the current controller and

Figure 3.7: Fundamental frequency 1000 Hz, i_d^* step changes from -400 A to -600 A, i_q^* step changes from 75 A to 50 A: (a) With 10% PM flux mismatch; (b) Without PM flux mismatch.

motor model. In Figure 3.7(b), there is not parameter mismatch. As can be seen, with 10% mismatch, the controller's dynamic performance is deteriorated, reflected in a 125% longer settling time when compared with the case without any mismatch. Although, in real EV propulsion application, the current reference would not have this kind of large step change. An accurate permanent magnet temperature estimation algorithm is definitely appreciated when it comes to improving current controller dynamic performance at high temperature.

CHAPTER

4

HIGH PERFORMANCE ME CURRENT CONTROLLER & GATE STITCHING MODULATION SCHEME

4.1 ME Current Controller Design

The first step of designing a model predictive controller is to establish the system model, as shown in Figure 4.1. The voltage equations of the ME system in the stationary frame are as follows

$$\left. \begin{aligned} 2(V_{ab,IUT} - V_{ab,ME}) + (V_{bc,IUT} - V_{bc,ME}) &= 3L_l \cdot \frac{di_a}{dt} \\ 2(V_{bc,ME} - V_{bc,IUT}) + (V_{ab,ME} - V_{ab,IUT}) &= 3L_l \cdot \frac{di_c}{dt} \\ \frac{di_a}{dt} + \frac{di_b}{dt} + \frac{di_c}{dt} &= 0 \end{aligned} \right\} \quad (4.1)$$

where L_l stands for line inductance, and i_a , i_b and i_c represent three phase currents, respectively.

Figure 4.1: Equivalent circuit of the system for MPC design.

Since the ME has much higher switching frequency than the IUT, one can first assume that the switching state of IUT remains unchanged within one control period of ME $T_{c,ME}$. Therefore, the phase to phase voltage for ME and IUT could be re-written as follows

$$\left. \begin{aligned} V_{xy,IUT} &= (S_{x,IUT} - S_{y,IUT}) \cdot V_{DC}^{IUT} \\ V_{xy,ME} &= (d_{x,ME} - d_{y,ME}) \cdot V_{DC}^{ME} \\ x, y &\in \{a, b, c\} \end{aligned} \right\} \quad (4.2)$$

Here S (equals to 0 or 1) is the switching state for the top switch of one phase leg of IUT and d (ranges from 0 to 1) represent the duty cycle for the top switch of one particular phase leg of the ME over one control period $T_{c,ME}$. The effect of IUT switching transient will be discussed in section 4.2.

Another assumption to simplify the analysis is that the DC bus voltage for IUT and ME are set to be identical and isolated, although different bus voltages as well as a common DC-link can be used with design modifications (24). The adopted approach simplifies the design and ensures that there is no zero sequence current in the system. With these assumptions, the discrete-time model for the system can be written as follows and given in equations (4.3) through (4.5).

Figure 4.2: ME current controller and modulation architecture.

$$\begin{aligned}
 i_{a,k+1} = & i_{a,k} + T_{c,ME} \cdot \\
 & \left\{ \frac{(V_{DC}^{IUT} \cdot (2S_{a,IUT} - S_{b,IUT} - S_{c,IUT}))}{3L_l} \right. \\
 & \left. - \frac{(V_{DC}^{ME} \cdot (2d_{a,ME}^k - d_{b,ME}^k - d_{c,ME}^k))}{3L_l} \right\}
 \end{aligned} \tag{4.3}$$

$$\begin{aligned}
 i_{c,k+1} = & i_{c,k} + T_{c,ME} \cdot \\
 & \left\{ \frac{(V_{DC}^{ME} \cdot (d_{a,ME}^k + d_{b,ME}^k - 2d_{c,ME}^k))}{3L_l} \right. \\
 & \left. - \frac{(V_{DC}^{IUT} \cdot (S_{a,IUT} + S_{b,IUT} - 2S_{c,IUT}))}{3L_l} \right\}
 \end{aligned} \tag{4.4}$$

$$i_{a,k+1} + i_{b,k+1} + i_{c,k+1} = 0 \tag{4.5}$$

The high level controller structure is shown in Figure 4.2. The class of model predictive controllers (MPC) can be divided into two groups (40): Finite control set MPC (FCS-MPC) and continuous control set MPC (CCS-MPC). CCS-MPC requires a modulator which maintains a constant switching frequency. FCS-MPC is conceptually easy to understand, but

Figure 4.3: Sequential diagram for hybrid MPC implementation.

with the increase of prediction horizon N the computation burden goes up exponentially. Also the switching frequency would vary a lot when using FCS-MPC. Since the switching frequency of the ME power stage is the main constraint in the ME implementation, CCS-MPC is preferred because the modulator can limit the maximum switching frequency. The prediction horizon is set to be one ($N = 1$). Based on (4.3) to (4.5), the prediction equation for the ME system can be derives as

$$\left. \begin{aligned}
 d_{ab,ME}^k &= d_{a,ME}^k - d_{b,ME}^k \\
 &= (S_{a,IUT} - S_{b,IUT}) - \frac{[2(i_{a,k+1}^* - i_{a,k,fb}) + (i_{c,k+1}^* - i_{c,k,fb})] \cdot L_l}{T_{c,ME} \cdot V_{DC}} \\
 d_{bc,ME}^k &= d_{b,ME}^k - d_{c,ME}^k \\
 &= (S_{b,IUT} - S_{c,IUT}) - \frac{[2(i_{c,k+1}^* - i_{c,k,fb}) + (i_{a,k+1}^* - i_{a,k,fb})] \cdot L_l}{T_{c,ME} \cdot V_{DC}} \\
 \min[d_{a,ME}^k, d_{b,ME}^k, d_{c,ME}^k] &= 0
 \end{aligned} \right\} \quad (4.6)$$

Table 4.1: ME Simulation and Experiment Parameters

Parameters	Value
IUT DC Link Voltage ($V_{DC, IUT}$)	300 V
ME DC Link Voltage ($V_{DC, ME}$)	300 V
IUT Switching Frequency ($f_{sw, IUT}$)	10 kHz
ME Controller Frequency ($f_{c, ME}$)	400 kHz
Line Inductance (L_l)	0.15 mH
Line Inductance in MPC (L_{l0})	0.1 mH
Target Motor Inductance (L_d)	0.4 mH
Target Motor Inductance (L_q)	0.6 mH
Target Motor Resistance (R_s)	0.1 Ω
Target Motor PM Flux (λ_{pm})	0.1 Wb
Target Motor Pole Number (P)	8
Dead-band Time IUT ($t_{DB, IUT}$)	2000 ns
Dead-band Time ME ($t_{DB, ME}$)	200 ns
ME integral controller gain (K_i)	0.0023

where $i_{a,k+1}^*$ and $i_{c,k+1}^*$ are the phase current references for the $(k+1)$ instant. The stationary frame current reference can be obtained by passing i_d and i_q from (??) through power variant $dq - abc$ transformation.

From (4.6), the required duty cycles for the power devices are calculated using current feedback ($i_{x,k,fb}$), future step current reference ($i_{x,k+1}^*$) and switching states of IUT ($S_{x,IUT}$) based on the system model described in (4.3) to (4.5). It is worth mentioning that only the differential duty cycle are derived in (4.6). The common mode duty cycle of three phases can be set according to different constrains and optimization goals. In this research, the smallest of the $d_{a,ME}^k$, $d_{b,ME}^k$ and $d_{c,ME}^k$ is set to be zero to avoid voltage saturation and minimize switching frequency of the ME.

$$i_{a,k+1}^{est} = i_{a,k,fb} + T_{c,ME} \cdot \left\{ \frac{(V_{DC}^{IUT} \cdot (2S_{a,IUT} - S_{b,IUT} - S_{c,IUT}))}{3L_l} - \frac{(V_{DC}^{ME} \cdot (2d_{a,ME}^k - d_{b,ME}^k - d_{c,ME}^k))}{3L_l} \right\} \quad (4.7)$$

$$i_{c,k+1}^{est} = i_{c,k,fb} + T_{c,ME} \cdot \left\{ \frac{(V_{DC}^{ME} \cdot (d_{b,ME}^k - 2d_{c,ME}^k + d_{c,ME}^k))}{3L_l} - \frac{(V_{DC}^{IUT} \cdot (S_{b,IUT} - 2S_{c,IUT} + S_{a,IUT}))}{3L_l} \right\} \quad (4.8)$$

The sequential diagram for implemented MPC controller is shown in Figure 4.3. At the start of k^{th} control period, the duty cycle calculated within the previous $(k-1)^{th}$ control period is applied to the ME power stage and the ADC for phase current sensors starts converting. Then, when the ADC results are ready, the computation of MPC starts. The computation should be finished by the end of the k^{th} cycle; and the result will be applied to the power stage at the very beginning of the $k+1^{th}$ cycle. Figure 4.3 shows that there is a one $T_{c,ME}$ delay in the real MPC implementation. To compensate for the delay, the estimation of phase current at $(k+1)^{th}$ instant ($i_{x,k+1}^{est}$) is required during the k^{th} control cycle prior to the computation of MPC. The estimation is done by (4.7) and (4.8).

The estimated current value is then used to replace the current feedback in prediction equation (4.6). Therefore, the final form of the prediction equation can be written as given in (4.9). The calculated duty cycle in (4.9) will be passed through a traditional modulator consisting of a carrier wave and comparator to generate the switching pattern.

$$\left. \begin{aligned} d_{ab,ME}^k &= d_{a,ME}^k - d_{b,ME}^k \\ &= (S_{a,IUT} - S_{b,IUT}) \cdot \frac{V_{DC}^{IUT}}{V_{DC}^{ME}} \\ &\quad - \frac{[2(i_{a,k+1}^* - i_{a,k,fb}) + (i_{c,k+1}^* - i_{c,k,fb})] \cdot L_l}{T_{c,ME} \cdot V_{DC}^{ME}} \\ d_{bc,ME}^k &= d_{b,ME}^k - d_{c,ME}^k \\ &= (S_{b,IUT} - S_{c,IUT}) \cdot \frac{V_{DC}^{IUT}}{V_{DC}^{ME}} \\ &\quad - \frac{[2(i_{c,k+1}^* - i_{c,k,fb}) + (i_{a,k+1}^* - i_{a,k,fb})] \cdot L_l}{T_{c,ME} \cdot V_{DC}^{ME}} \\ &\quad \min[d_{a,ME}^k, d_{b,ME}^k, d_{c,ME}^k] = 0 \end{aligned} \right\} \quad (4.9)$$

One of the drawbacks of MPC is its reliance on the model accuracy. In the real implementation for the ME system, the non-linearity of the line inductors and the deadband time of the converter, which are not included in (4.9), would introduce steady state error in the phase current. To solve the issue, integral (I) controllers are added in parallel with the MPC controller (41). Since the integral controllers are operating in dq-reference frame, corresponding transformations are also needed. Moreover, the line inductance value used in MPC is set to be 33% smaller than the actual line inductance (L_l). This way, the MPC can have better noise immunity, at the price of bandwidth (41). Moreover, the reduction of inductance at high current due to non-linearity can be taken into account.

4.2 Controller Performance Analysis

To evaluate the stability margins of the controller, next, a small-signal model is developed. For simplicity, we consider $V_{DC}^{IUT} = V_{DC}^{ME} = V_{DC}$. Using (4.3), (4.4), and (4.5), the current dynamics in each phase can be derived as

$$i_{x,k+1} = i_{x,k} + \frac{V_{DC} T_{c,ME}}{L_l} (-d_{x,ME}^k + d_{CM,ME}^k + S_{x,IUT} - S_{CM,IUT}), \quad (4.10)$$

where, $d_{CM,ME}^k$ and $S_{CM,IUT}$ denote the common-mode (CM) duty ratios for the ME and the IUT, respectively. For the small signal signal analysis, we exclude the CM and the controller dynamics can be derived as

$$d_{x,k+1} = -d_{x,k} + 2S_{x,IUT} - \frac{L_l}{V_{DC} T_{c,ME}} (i_{x,k+1}^* - i_{x,k+1}) \quad (4.11)$$

The equivalent system model is shown in Figure 4.4, where the plant model includes the parasitic resistance R_l of the inductor. The MPC is designed ignoring the R_l ; additional unmodelled dynamics may arise due to nonlinearity of the circuit components and/or sensing and signal conditioning circuit. To account for such unmodelled dynamics, an integral compensation with gain K_i is added. A zero-order-hold (ZOH) is used to model the modulator. With a slight misuse of notation, the sampling and the ZOH are taken together as $F_{zoh}(s) = (1 - e^{-sT_s})/(sT_s)$, where $T_s = T_{c,ME}$. Any variation in the DC bus voltage of the IUT and/or any other external disturbance is represented by v_d . To differentiate between the model used by the MPC and the actual plant, the line physical inductor and the model

used by the MPC are denoted as L_{l0} and L_l , respectively. A continuous-time equivalent of the system below the Nyquist frequency $f_s/2 = 1/(2T_s)$ can be obtained using $z = e^{sT_s}$. The stability margin of the current control loop can be assessed using the open-loop response

$$G_{OL}(s) = \frac{F_{zoh}(s)e^{-sT_s}}{sL_{l0} + R_l} \left[\frac{(L_l/T_s)}{1 + e^{-sT_s}} + \frac{K_i V_{dc} e^{-sT_s}}{1 - e^{-sT_s}} \right] \quad (4.12)$$

The disturbance rejection capability can be assessed using the admittance $Y_d(s)$ given as

$$Y_d(s) = \frac{i_x(s)}{v_d(s)} = 1/[(sL_l + R_l)(1 + G_{OL}(s))] \quad (4.13)$$

Figure 4.4: Equivalent system model for small signal analysis.

To illustrate the design process, we consider the ME system with parameters listed in Table 4.1. The open-loop response and the admittance for two different parasitic resistance values of $R_l = 20m\Omega$ and $R_l = 100m\Omega$ are shown in Figure 4.5 and Figure 4.6, respectively. Here, $L_l = 0.67L_{l0}$ is used to achieve better noise immunity at the cost of bandwidth (41); the integral gain is set as $K_i = 0.0023$. The current controller bandwidth of $\approx 20kHz$ remain unaffected for the variation in R_l . Considering that the current controller bandwidth of the IUT (switching at $10kHz$) is designed to be less than $2kHz$, the stability of the whole emulation testbed is therefore ensured by the large difference in ME and IUT current control bandwidths (26). Moreover, high disturbance rejection capability of $> 25dB$ is achieved over the entire frequency range with $> 50dB$ attenuation up to hundreds of Hz .

Figure 4.5: Compensated open-loop response $G_{OL}(s)$ of the MPC for different parasitic resistance R_l .

The pulse-voltage sensing described in 4.3 is used to implement the feedforward of IUT pulse voltage v_x , which is essential for accurate current tracking. Figure 4.7 shows the comparison of frequency responses of $Y_{dr}(s) = i_x(s)/v_x(s)$ with and without the pulse voltage feedforward. Evidently, the feedforward greatly augments the disturbance rejection capability. During the transient of IUT switching states $S_{x,IUT}$, the IUT pulse voltage feedforward would become inaccurate, deteriorating the controller disturbance rejection performance, as indicated in Figure 4.7. Gate stitching modulation strategy is therefore proposed to solve this issue.

4.3 Gate Stitching Modulation Scheme

As mentioned previously, one of the assumptions during the system analysis is that the switching state of IUT ($S_{x,IUT}$) remains the same within one control period of the ME ($T_{c,ME}$). Even though the $T_{c,ME}$ is much smaller than the switching period of IUT ($T_{sw,IUT}$), there is bound to be a $T_{c,ME}$ within which the $S_{x,IUT}$ are not constant. The ME controller will not be able to react within the same and the next control period when these instances occur. Thus, there will be a large current spike during the transient of $S_{x,IUT}$. To solve the issue, a gate stitching modulation strategy to synchronize the switching state of IUT and ME during the

Figure 4.6: Disturbance rejection response $Y_d(s)$ of the MPC for different parasitic resistance R_l .

transient of $S_{x,IUT}$ has been developed. The logic of the gate stitching modulation is shown in Figure 4.8. The principle of the gate stitching modulator is to force the switching state of ME ($S_{x,ME}$) to be identical to that of the IUT whenever it detects an edge in $S_{x,IUT}$. The current slope in all three phases will be minimized when the switching states in IUT and ME are controlled to be identical, as long as $V_{DC}^{ME} \leq 2V_{DC}^{IUT}$. The gate stitching modulation strategy suppresses the phase current spikes during the transient of $S_{x,IUT}$. The active time of the gate stitching is twice of $T_{c,ME}$, shown by the pulse width of *GateStitching_En* signal in Figure 4.8, since the transient of $S_{x,IUT}$ will affect two control periods. During the time when gate stitching is active, the phase current slope is minimized, which would differ from what should have been if IUT drives an actual motor. This could introduce some error in phase current. However, the active time for the gate stitching is very small compared to the switching period of IUT ($T_{sw,IUT}$) and the high performance current controller can compensate for the small error in a timely manner. Later in the Section. VI, it can be seen that there is not error in the fundamental phase current, meaning the modulation index of IUT is not corrupted by the gate stitching strategy.

In Figure 4.8, $S_{x,IUT}$ represent the switching state of corresponding phase leg of IUT, which can be easily obtained by measuring the logic level of IUT phase PWM voltage. The dx, ME, fb signals in Figure 4.8 represent the duty cycle to be fed back to the MPC block ($d_{x,ME}^{k-1}$ in Figure 4.2). These signals should also be overridden with the corresponding $S_{x,IUT}$

Figure 4.7: Disturbance rejection response $Y_{dr} = i_x(s)/v_x(s)$ of the MPC with and without the IUT pulse voltage feedforward.

since during the transient of $S_{x,IUT}$, the actual gate pulses of ME are not dictated by the hybrid MPC controller due to gate stitching. The *GateStitching_En* signal is used to select the source for $S_{x,ME}$. During the transient of $S_{x,IUT}$, the $S_{x,ME}$ are set to be identical to $S_{x,IUT}$. Otherwise, the $S_{x,ME}$ is generated by passing the duty cycle $d_{x,ME}$ ($d_{x,ME}^k$ in Figure 4.2) from hybrid MPC through a traditional 200 kHz up-down carrier PWM block. It is worth noting that gate stitching could create very narrow pulses. Therefore, special digital circuit consisting of logic gates and monoflops are designed to restrict the minimal pulse width generated in the system. In this work, the minimum pulse width is limited to 400 ns. All the components and function blocks needed to achieve the gate stitching logic in Figure 4.8 and restrict minimal pulse width can be easily and efficiently implemented in FPGA, while on a DSP based platform some additional logic ICs are required to achieve the same functionality.

4.4 Simulation & Experiment

To verify the performance of proposed hybrid model predictive controller and gate stitching modulation scheme, simulations with multiple transients are performed. While the

Figure 4.8: Gate stitching modulation scheme.

experiments are carried out on a ME prototype with detailed analysis on current tracking performance. The experiment setup consists of a SiC based ME prototype and IGBT based IUT.

4.4.1 Simulation Result

Simulation results are presented in this section to validate the proposed high bandwidth current control and gate stitching modulation scheme for ME. The simulation platform used is PLECS from Plexim. The simulation parameters are listed in Table. I. In order to represent the real user case, the IUT is operated in closed loop current control mode. From 0 sec to 0.005 sec, the fundamental current are controlled to be zero to highlight the ripple current. At 0.0025 sec, the gate stitching is enabled resulting in improved tracking accuracy of the ripple current, with magnified view shown in top subplot of Figure 4.9(b). At 0.01 sec, the d -axis current step changes from 15 A to 0 A, the q -axis current step changes from 0 A to 35 A and the motor speed step changes from 3000 rpm (200 Hz) to 6000 rpm (400 Hz). The magnified view during the transient is presented in the middle subplot of Figure 4.9(b), in which the ability of the system to track the high frequency ripple current is clearly demonstrated. At 0.0144 sec, the gate stitching is disabled, resulting in large current spike in the system, shown in the bottom subplot of Figure 4.9(b). The top and bottom subplots in Figure 4.9(b) demonstrate that gate stitching strategy can indeed improve the ripple current tracking accuracy. In Figure 4.9(a), the top subplot shows three phase current and reference generated by the motor model, the bottom subplot indicates how many times the device has switched. The slope of the Gate Pulse Count curve in Figure 4.9(a) during

Figure 4.9: Simulation results with 3-phase current and reference under multiple transients and online enable/disable of gate stitching strategy. (a) During 0 to 0.005 sec the fundamental current is controlled to be zero. Step changes in i_d , i_q and motor speed reflected at 0.01 sec. (b) Magnified view during transients of phase current and enable & disable of gate stitching.

the time when gate stitching is enabled is smaller than that when gate stitching is disabled, demonstrating the gate stitching can also help reducing the switching frequency of ME. The average absolute current error from 0.005 sec to 0.01 sec is 0.70 A with phase current amplitude of 15 A. While during 0.01 sec to 0.015 sec, the average absolute current error is 0.69 A for 35 A phase current amplitude. According to the results, the ME system can follow the current reference from the motor model pretty closely. Also the average and peak current error is irrelevant to the magnitude of the phase current. Even under simultaneously multiple transients of d -axis current, q -axis current and fundamental frequency at 0.01 sec, the current tracking capability is still quite good. The overall average switching frequency of the ME with gate stitching enabled is kept at a relatively low 110 kHz without large variation and imbalance between different phases.

4.4.2 Experimental Result

A ME prototype is built with 1.2 kV, 50 A SiC MOSFET 6-pack full bridge module from Wolfspeed (CCS050M12CM2) to validate the proposed high bandwidth controller design and gate stitching strategy for the ME system. The ME prototype power stage is rated at 50

Figure 4.10: Scope waveform of all four operating point. Current: 10 A/div for (a) and (c); 20 A/div for (b) and (d), voltage: 200 V/div, time: 1 ms/div. CH1, CH3, CH4: 3-phase current, CH5: V_{DS} of ME, CH6: V_{DS} of IUT. (a) 15 A, 200 Hz (3000 rpm). (b) 35 A, 200 Hz (3000 rpm). (c) 15 A, 400 Hz (6000 rpm). (d) 35 A, 400 Hz (6000 rpm).

kVA. The SiC device can sustain high switching frequency under high power operation (42). The ME testbed developed is scalable and can be designed for higher power levels by using higher current rated SiC devices. If large enough devices are not available in a single package, multiple smaller rated devices such as the one used in this presented prototype can be put in parallel (interleaved or non-interleaved) to boost the power level. In either solution, the proposed hybrid MPC based controller and gate stitching strategy would still be applicable. The IUT is a 50 kW IGBT based inverter switching at 10 kHz. The line inductors are in-house designed, utilizing high flux material CH571060 from CWS ByteMark with $\mu = 60$, to have good linearity within the current rating of the ME. The core loss of the inductor under 130 kHz switching frequency is estimated to be 17 W with 10 A peak-to-peak ripple current and 50 A RMS current rating. The inductor design and loss analysis methodology are presented in our previous work (43). The DC source for ME is high voltage battery pack with nominal voltage of 300 V, while the IUT uses a commercial DC power supply.

The experimental parameters are the same as in the simulation as given in Table. I. To demonstrate the high bandwidth current control capability of ME system, test waveform with several fundamental cycles are captured to highlight the tracking capability of the

Figure 4.11: Comparison between current reference and current feedback with FFT analysis. (a) 15 A, 200 Hz (3000 rpm). (b) 35 A, 200 Hz (3000 rpm). (c) 15 A, 400 Hz (6000 rpm). (d) 35 A, 400 Hz (6000 rpm).

high frequency ripple currents. This high bandwidth capability clearly indicates that slower dynamic changes such as those in a drive cycle can be easily tracked. For the first experiment, the fundamental frequency is set to 200 Hz, which is equivalent to a motor speed of 3000 rpm. Meanwhile, the average phase current is controlled to be 15 A and 35 A to validate the newly proposed control and modulation scheme. For the second set of experiments, the fundamental frequency is doubled to 400 Hz (6000 rpm). While the phase current is still controlled to be 15 A and 35 A. The scope screenshots for low current (15 A) and high current (35 A) operation for different fundamental frequencies are shown in Figure 4.10. The two metrics utilized for evaluating the current tracking accuracy are the average absolute current error over one fundamental cycle and the FFT results for the current reference and

current feedback. It is important to log the current reference from the motor model and current feedback simultaneously for an accurate analysis. The integrated logic analyzer (ILA) within Xilinx Zynq-7000 SoC is suitable for this application. The current feedback and current reference are sampled and logged at 1 MHz frequency.

In Figure 4.11 (a) and (b), the current reference and feedback of two operating points at 200 Hz (3000 rpm) are plotted and analyzed. The average absolute phase current error during the time periods of Figure 4.11 (a) and (b) for 15 A operation is 0.97 A, while that at 35 A is 0.96 A. In Figure 4.11 (c) and (d), the current reference and feedback of two operating points at 400 Hz (6000 rpm) are plotted and analyzed. The average absolute phase current error during the time period of Figure 4.11 (c) and (d) for 15 A operation is 1.11 A, while that at 35 A is 1.25 A. Therefore, it is safe to say that even under high phase current operation, the current tracking error would still be maintained at a relatively low level. The current tracking error increases with the fundamental frequency which is expected as reported in (24), but the error at higher speeds is still reasonable. Also, the phase current error is not proportional to the amplitude of the phase current, but is mainly determined by the DC bus voltage, the line inductance and the control frequency. The relationship between current error and the system parameters are as follows

$$i_{err} \propto \frac{V_{DC} \cdot T_{c,ME}}{L_l} \quad (4.14)$$

To further evaluate the current controller tracking performance, one can take a look at the FFT results for phase current reference and feedback in Figure 4.11. According to the FFT results, the current feedback matches current reference very well up to twice the switching frequency of IUT (20 kHz). The matching of the fundamental current confirms that the gate stitching strategy does not reduce the modulation index of IUT. However, the matching of the harmonic order is important since this is related to the motor torque ripple order and the inverter experiences the exact conditions as with a real motor. To further demonstrate the high bandwidth of the ME, a step response test is carried out as shown in Figure 4.12. The current reference for the ME step change from 15 A to -15 A in *d*-axis. The response is very fast with a transient of only 50 μ s, which is consistent with the estimated 20 kHz bandwidth from the analysis in Section IV A. Moreover, in this paper, the utilization of ultra-fast model predictive control makes it possible to have a large target motor inductance (L_{dq}) to line inductance (L_l) ratio, which could widen the range of the target motor inductance. It is worth noting that the observed switching frequency of ME device ranges between 120 kHz to 140 kHz during the aforementioned tests, with the highest

Figure 4.12: ME step change response.

switching frequency observed at 25 A, 200 Hz, as shown in Figure 4.13. The dashed pink lines in Figure 4.13 indicate the intervals where gate stitching is activated to force the switching states of ME to be identical to that of IUT, eliminating almost completely any spike in phase current. The start of the intervals are triggered by the edges of IUT V_{DS} . The length of the intervals is twice that of $T_{c,ME}$ ($5 \mu s$). The theoretical maximum switching frequency for ME is limited to 200 kHz by the triangle carrier wave PWM module used to convert the duty cycle output of hybrid MPC to gate signals. The causes for the reduction and variation of the ME device switching frequency are the gate stitching and the optimization of common mode duty cycle by the hybrid MPC.

The nominal DC bus voltage for IUT and ME are the same in this paper, although the battery pack for ME experience around 15% voltage fluctuation during the experiment. The

Figure 4.13: Within Δt of $253 \mu s$, there are 36 pulses in the V_{DS} of ME SiC device. The active time of the gate stitching is labelled in the waveform. No spike in phase current at the edge of IUT V_{DS} . The time division is $50 \mu s$.

difference in DC bus voltages is taken into account by the current controller and the relative small voltage difference would not deteriorate the effect of gate stitching in suppressing current spike. For emulating certain target motors with high inductance and/or high speed rating for the full operating range, the required DC bus voltage of ME may need to be double or more to that of IUT as mentioned in (24). In our proposed ME system, the large DC bus voltage difference could corrupt the effect of the gate stitching strategy in reducing current spike during the transient of $S_{x,IUT}$ when using two-level VSI topology for ME. The high control frequency of our proposed ME addresses the issue to certain extent. Another solution to circumvent the issue, if necessary, is to use more complex alternative converter topology such as the multi-level inverter where the proposed hybrid MPC and gate stitching strategy are still applicable. Such a topology will help achieve a large V_{DC}^{ME} to V_{DC}^{IUT} ratio while maintaining high ripple current tracking accuracy.

The proposed hybrid model predictive controller along with the novel gate stitching modulation strategy achieves $20 kHz$ current control bandwidth. The high frequency ripple current caused by the $10 kHz$ switching of IUT can be accurately emulated in the system. Also, the DC-link voltage (V_{DC}) to line inductance (L_l) ratio in this paper is much higher than anything reported previously which means a smaller value inductor can be used for the developed ME system.

CHAPTER

5

FPGA DESIGN METHODOLOGY FOR HIGH PERFORMANCE POWER ELECTRONICS APPLICATION

The adoption rate of wide-bandgap devices, such as Silicon Carbide (SiC), is steadily increasing in power electronics with the advances in manufacturing processes. The faster switching capability of wide-bandgap devices than that of Si devices provides an opportunity to improve control bandwidth and poses challenges in device protection. An FPGA-based short-circuit protection circuit for the 10-kV SiC MOSFET is proposed in (44). To fully utilize the high switching frequency capability to boost the control performance, the FPGA-based high-performance digital control platform provides a better choice since the increased switching frequency poses a limitation on the allowable latency for the digital system, including the sampling and control algorithm computation delay. In (21), high bandwidth VIENNA rectifier with over 1 *MHz* switching frequency was achieved using an FPGA based control platform.

FPGA-based system development using hardware description language (HDL) is a

Figure 5.1: Workflow of proposed FPGA design methodology for rapid prototyping in power electronics.

rigorous and cumbersome process. Fortunately, FPGA manufacturers have introduced high-level design tools, such as high-level synthesis (HLS) (45), and block diagram based design toolbox, such as *System Generator for DSP* from Xilinx and *DSP Builder* from Altera, to help expedite the development process. A critical step to accelerate the development process is to efficiently incorporate these high-level design tools into the existing design workflow in power electronics. Examples of previous work on the FPGA controller development process include predictive control in FPGA design using HLS (46), resource streaming design technique in motor drive applications (47), and block diagram based FPGA design (48). In (49), detailed design and implementation of an impedance identification algorithm block for resonant inverter using HLS were presented. In this work, the digital control platform chosen is the SoC type (ARM+FPGA) in order to bring more flexibility in the system design and deliver high bandwidth, low latency power electronic controllers. This paper presents the design methodology using a toolchain involving multiple software applications while illustrating the merit of this type of platform. Two examples are provided with different implementation strategies. Compared to other FPGA implementation papers (50; 51) which focus on only specific function block or design tool, this paper provides a more systematic high-level perspective and methodology for designing digital systems requiring high control bandwidth, high fidelity modeling, and low latency for rapid prototyping of power electronics applications.

```

void apint_arith(dinA_t  inA, dinB_t  inB,
                dout1_t *out1
                ) {

    dout2_t temp;
    #pragma HLS RESOURCE variable=temp core=AddSub_DSP

    temp = inB + inA;
    *out1 = temp;

}

```

Figure 5.2: Example code snippet with HLS pragma for resource utilization optimization (2).

The proposed rapid prototyping approach not only maximizes the FPGA resource utilization, but also can be naturally integrated into the existing design workflow of power electronics applications. Moreover, this paper divulges how to assign various power electronics digital control tasks, like ADC, reference frame transformation, control algorithm, PWM generation, and communication, to different parts (ARM core or FPGA) of the SoC platform according to system-level priority requirement, latency requirement, and implementation complexity.

5.1 FPGA Platform & Design Toolchain

There are quite a few ways to use FPGA in real-time control systems. In Table. I, four types of FPGA digital system architecture are listed and compared. Type 1, denoted as pure FPGA architecture, is the bare FPGA structure that does not provide the peripherals and requires low-level design knowledge. Type 2 solution combines the abundant peripherals of DSP with the powerful FPGA(52); however, building a fast and reliable communication link between DSP and FPGA could be pretty challenging. Furthermore, the two-chip solution complicates the controller board design. Type 3 is the FPGA-embedded general-purpose processor system-on-chip (SoC) platform, which typically includes many peripherals, like CAN, USB, Ethernet and UART, and analog-to-digital converter. These peripherals can make the platform more flexible and save FPGA's logic resource for computational use compared with Type 1 architecture. Third-party FPGA solution is denoted as Type 4. Companies, such

Table 5.1: Comparison between different type of FPGA implementation

	Type 1	Type 2	Type 3	Type 4
Architecture	Pure FPGA	FPGA+DSP	SoC	Third party solution
Toolchain	Native	Native	Native	Third party software
Merit	<ul style="list-style-type: none"> 1. Simple one chip solution. 2. Highly customizable. 	<ul style="list-style-type: none"> 1. Highly customizable. 2. Abundant peripherals. 	<ul style="list-style-type: none"> 1. Simple one chip solution. 2. Highly customizable. 3. Abundant peripherals. 	<ul style="list-style-type: none"> 1. High level graphic development tool is easy to learn. 2. No need for low level FPGA design knowledge.
Demerit	<ul style="list-style-type: none"> 1. The lack of peripherals. 2. Require knowledge on low level FPGA design. 	<ul style="list-style-type: none"> 1. Hardware design is complicated. 2. Chip-to-chip communication needed. 3. Require knowledge on low level FPGA design. 	<ul style="list-style-type: none"> 1. Require knowledge on low level FPGA design. 2. Logic resource may be limited. 	<ul style="list-style-type: none"> 1. Hardware is not customizable. 2. Performance of the FPGA may be limited by the third party development software.

as *National Instruments*, provide their hardware consisting of an FPGA chip from Xilinx or Altera and their own high-level block diagram-based development software such as LabVIEW FPGA. This solution provides an easier learning curve for FPGA programming but has several limitations. First, this platform's flexibility is limited since the hardware cannot be customized to achieve specific functionality. Second, the FPGA performance might not be fully utilized since the third-party software's code optimization is usually subpar compared to that of the native development tools.

In this work, we opt for the Type 3 implementation using the ZYNQ-7000 family from Xilinx for efficient development and effective resource utilization. The ZYNQ-7000 System-on-Chip (SoC) integrates a dual-core ARM Cortex-A9 based processing system (PS) and 28 nm Xilinx programmable logic (PL) in a single device. This solution provides better

Table 5.2: Comparison of internal AXI interface

Name	AXI4-Lite	AXI4	AXI4-Stream
Implementation Complexity	Low	Medium	High
Address Format	Memory-mapped	Memory-mapped	No address
Throughput	Low	Medium	High
Overhead	Low	Medium	High
Target Application	Mainly for IP core control purpose	For bulk data transmission	For image and video processing application

communication performance between processor and FPGA and simplicity of controller board design over Type 2 solution using separate DSP and FPGA chips. A PicoZed system on module (SoM) board (XC7Z015-1SBG485) from Avnet is used for design simplicity. It is worth noting that the selected board is not the most powerful option from the PicoZed family. The selected option provides 3.3 V I/O voltage level whereas more powerful and faster devices can only support 1.8 V or 2.5 V I/O voltage levels. I/O pins with lower voltage levels have lower noise immunity. In power electronics applications, especially with fast switching SiC devices, signal noise is a big concern, and hence, the selection is made for better noise immunity.

The development toolchain used for ZYNQ-7000 SoC in this work includes Xilinx Vivado, Xilinx software development kit (SDK), Vivado High-Level Synthesis (HLS), and System Generator for DSP. The Vivado is for the system-level synthesis and implementation of the FPGA hardware design. The Xilinx SDK is used to program the ARM core (PS) within the ZYNQ-7000 SoC. The Vivado HLS is used for designing the FPGA IP core using C/C++. Last but not least, System Generator for DSP is a block diagram-based MATLAB Simulink toolbox into which HDL and HLS designs can be imported. It should be pointed out that other FPGA manufacturers have similar toolchains for the same purpose.

5.2 Design Methodology

The workflow of the proposed FPGA design methodology for rapid prototyping digitally controlled power electronics system is shown in Figure 5.1. The design methodology em-

Figure 5.3: Co-simulation setup in Simulink environment, with the source for different blocks labelled.

phasizes the concurrent design of the power stage and digital controller, which enables high-level system performance and optimum resource utilization. The power stage's basic specifications include topology, power device performance, passive component values, and sensor performance. These hardware specifications could also affect the digital control system's design, such as the controller structure, controller parameters, and control iteration speed. Using software like PLECS and Simulink, the power stage can be accurately represented by the circuit diagram. Simultaneously, the function blocks like C-Script and S-Function could simulate the C code-based digital controller's behavior. This way, the preliminary system specifications can be established.

After the preliminary round of design iteration, we should have a functional prototype

Table 5.3: Comparison of resource utilization and latency for online motor model block.

	W/ code style adaption and pipelining	W/O code style adaption	W/O pipelining
Latency	29 CLK	41 CLK	33 CLK
DSP	38%	61%	38%
BRAM	17%	17%	12%
Flip-flop	5%	8%	4%
LUT	2%	4%	2%

Figure 5.4: The configuration menu in System Generator for DSP can be accessed by double-click the red Xilinx icon in Fig. 5.3. The target device, system constraints, and export preference can be defined in this menu.

C code for the digital controller. The next question is whether to implement the control algorithm in PS or PL cores. The PS CPU is versatile and easy to program using high-level languages. However, the PS's sequential nature means the absolute performance (algorithm execution time) for a specific task is no match to the PL. Although there are HLS tools to simplify FPGA's design process, it is still more complicated than the PS design. Also, the hardware resource of FPGA would limit how many algorithms can be executed concurrently. Resource utilization and clock speed are trade-offs. If the resource utilization is very high, there is less placing and routing flexibility, resulting in slower clock speed, with techniques like loop unrolling, loop merge, loop flattening, pipelining, and resource sharing, FPGA clock speed, and latency can be improved. Even though the latency may increase slightly, the total execution time could be reduced with faster clock speed. With the help of tools like HLS, those optimization techniques can be easily applied using corresponding directives

Figure 5.5: ZYNQ-7000 design block diagram for ME.

(2). Whereas using the conventional FPGA design method, the exploration of the design space might not be sufficiently rigorous. In sections IV and V, two examples of the detailed decision-making between PL and PS are presented.

For implementation in PS, this prototype code can be directly applied with minor modification. However, if the algorithm is to be implemented in PL, HLS would be a good choice. The Xilinx HLS tool synthesizes a C function into an IP block to integrate into a hardware system. It is tightly integrated with the rest of the Xilinx design tools and provides comprehensive language support, and features for creating the optimal implementation for a C algorithm (2). The design flow of HLS contains the following stages:

1. Compile, execute, and debug the C algorithm.
2. Synthesize the C algorithm into an RTL implementation, optionally using user optimization directives.
3. Generate comprehensive reports and analyze the design.
4. Verify the RTL implementation using a push button flow.
5. Package the RTL implementation into a selection of IP formats.

In this work, to better tailor this design flow for power electronics applications, stage 4 and 5 are replaced by System Generator for DSP toolbox. In stage 1, the adaption of code style is

Figure 5.6: Subsystem sequence and runtime.

needed for better results. For example, avoiding too many arithmetic operations within one line of code could improve resource utilization, as discussed in Chapter 3 of (2). After the code style optimization, the next step is to add directives to define device constraints, and resource utilization preference, described in stage 2, such as setting the clock speed and using DSP core to implement a multiplier (2). In Figure 5.2, an example code snippet with *pragma* directive to select DSP core for summation operation is presented. More information regarding how to apply directives and pragma can be found in Chapter 1 of (2). Moreover, different interfaces, like AXI4 and RAM, can be added to HLS design using corresponding directives. In stage 3, timing, latency, and resource utilization estimation are provided after C synthesis in a few minutes. This feature significantly accelerates the performance optimization process of HLS design.

Another critical step in the workflow is the functional verification of the HLS design. In (49), a C-based test bench was preferred for fast verification of the design in HLS, referred to as stage 4. However, for power electronics application, System Generator for DSP toolbox is preferred for verification purpose in this paper. System Generator for DSP is a design tool in the Vivado Design Suite that enables the MathWorks model-based Simulink design environment for FPGA design. Previous experience with Xilinx FPGA devices or RTL design methodologies is not required when using System Generator. Designs are captured in the

Resource	Utilization	Available	Utilization %
LUT	7558	46200	16.36
LUTRAM	797	14400	5.53
FF	10792	92400	11.68
BRAM	62	95	65.26
DSP	89	160	55.62
IO	77	150	51.33
BUFG	2	32	6.25

Figure 5.7: ZYNQ-7000 resource utilization report.

Simulink modeling environment using a Xilinx-specific block set. Downstream FPGA steps, including RTL synthesis and implementation, are automatically performed to produce an FPGA programming bitstream (53). Design in M-code, HDL code, and C/C++ code (HLS) can be imported to the System Generator environment for simulation and verification using MCode, Black Box, and Vivado HLS block, respectively, described in Lab 2 of (53). As mentioned before, the power stage is modeled in the PLECS environment. With the PLECS blockset, the PLECS model can be imported to the Simulink environment. Therefore, the co-simulation between digital control algorithm, modeled in System Generator, and power stage, modeled in PLECS blockset, can be easily carried out in Simulink. The screenshot of the Simulink co-simulation setup is shown in Figure 5.3. In Figure 5.3, the blocks with white background are native Simulink blocks, enclosed in the red rectangle. The blocks with yellow background, enclosed in orange rectangle, are from PLECS blockset, in which power stage can be modeled. The blocks with Xilinx icon, enclosed in purple rectangle, are from System Generator for DSP toolbox. It is worth noting that the gateway blocks in blue rectangle and the configuration menu icon in the green rectangle are also from System Generator toolbox. The gateway blocks are used to define the boundary of HDL design. While in the configuration menu, the target device, system constraints, and export preference can be established. The detail of this menu is shown in Figure 5.4. Admittedly, the C-based simulation in HLS is faster than the co-simulation in Simulink. However, with the existing power stage model in PLECS, the co-simulation method can be integrated into the traditional design workflow of power electronics more naturally and is still faster than

the conventional RTL-based design verification (49). Furthermore, with the fully functional prototype C code, a high number of design iterations are not expected in the co-simulation stage.

With the successful verification, the System Generator for DSP can package the design as an intellectual property (IP) core and export it to Vivado for system-level RTL synthesis and bitstream generation. The System Generator export setting can be accessed in the menu by double-clicking the Xilinx logo in Figure 5.3. In the menu presented in Figure 5.4, one can set the target device and clock speed and the IP export directory. More details regarding this menu setting can be found in Lab 3 of (53). Once the setting is finalized, the IP can be generated at the designated directory by hitting the “Generate” button. To use the IP in Vivado, the IP directory should be added to the IP catalog. With all the necessary IPs added to Vivado, system-level design can be synthesized and implemented, followed by the bitstream generation.

Then the bitstream can be exported to Xilinx SDK. Once the hardware design is transferred to the SDK environment, the board support package (BSP) with all the peripheral driver application programming interface (API) is generated. Finally, the application software for the ME system can be developed. The Xilinx SDK comes with great example designs for almost all the APIs. Since the Xilinx ZYNQ-7000 SoC platform selected for the examples of this paper contains a dual-core ARM processor (PS) and the FPGA (PL), there is more flexibility in the digital system design. The PS system development is quite similar to that of a DSP or microprocessor and beyond the scope of this paper. However, the PS and PL’s interconnection is essential for system performance and should not be ignored. The ZYNQ-7000 device provides a powerful internal bus called AXI4 (Advanced eXtensible Interface 4) to transfer data between the PL and the PS. There are three variations of the AXI interface for different application purposes. The comparison of the three variations is presented in Table. 5.2. The detailed decision-makings for each function block and AXI interface are illustrated in the examples provided in the next section.

5.3 Detailed Implementation

As shown in Figure 1.4, the computation cores for ME contains high fidelity motor model iterating at 1 *MHz*, a hybrid MPC current controller iterating at 400 *kHz*, and resolver emulator iterating at 1 *MHz*. To achieved satisfactory run-time performance, those three tasks should be executed in parallel. Implementing all three computation cores in PL ensures

Figure 5.8: Placed and routed design in ZYNQ-7000 SoC chip.

parallelism and eliminated the data transmission requirement between PS and PL for control purposes. Therefore, the entire digital system for ME is implemented in PL, except the Ethernet communication. The Ethernet is mainly for data logging and monitoring purposes. The Ethernet communication between ZYNQ-7000 SoC and PC happens every 5 *ms*, and the PS handles the Ethernet communication. Transferring data from the controller and motor model (located in PL) to the PS is the key to communication. For the ME application, although a small amount of data is generated every 400 *kHz* or 1 *MHz*, the single-shot AXI4-Lite interface is too slow to finish the transmission within the control period; a higher performance interface is needed for the ME application. As pointed out in Table. 5.2, the

higher performance interface has a large overhead. A block memory in PL is utilized as a buffer to bulk the data up for easier high-performance interface implementation to address the issue. AXI4 interface is eventually selected for this application. AXI4-Stream is overkill, and the PL implementation is more complicated. The AXI4 transaction will initiate every 5 *ms* to move the data in the buffer to the on-board memory for the PS. Then the Ethernet user datagram protocol (UDP) is initiated to send the data to the PC. This way, the data logging is running entirely in parallel with the control algorithm. The overall implementation block diagram for ME is presented in Figure 5.5. ADC interface and PWM block are also assigned to the PL part of the ZYNQ-7000 SoC.

Based on the iteration speed, the digital system for ME can be divided into three subsystems. The first one iterates at 1 *MHz*, which contains the real-time motor model and resolver emulator. The second one iterates at 400 *kHz* system, consisting of MPC, ADC, and PWM generation. The third one is the 200 *Hz* system, which is for the communication between the SoC and the PC laptop. Those three subsystems are running in parallel with each other. The sequence of operation for the 1 *MHz* and 400 *kHz* subsystems is presented in Figure 5.6. The motor model update is concurrent with the resolver emulation. The ADC for the resolver emulation has significant latency; therefore, it is done in parallel with the resolver computation and the DAC. This way, the 1 *MHz* iteration speed is achieved at the price of slightly longer latency. As for the 400 *kHz* subsystem, at the very start of the 2.5 μ s control period, the duty cycle calculated during the last control period is loaded into the PWM generation block. Also, the ADC for the phase current and DC bus voltage starts at the beginning of the 2.5 μ s control period. When the conversion is finished, then the MPC and integral (I) controller update are initiated. The newly calculated duty cycle is applied at the start of the next control cycle.

For PL computation core design, the workflow in Figure 5.1 is followed. One critical decision to make is whether to use floating-point or fixed-point arithmetic. The main advantage of floating-point data type is the ease of implementation, and the PL of ZYNQ-7000 SoC also supports floating-point data types. However, since both the real-time motor model and MPC require lots of computation and low latency, the full floating-point implementation would exceed the limit of on-chip resources and timing requirement. Thus in this application, the fixed-point data type is used. Xilinx provides a fixed point data type called *ap_fixed* in the high-level synthesis (HLS) software (54), which makes it easy to use fixed-point data types in the design. As mentioned above, the coding style adaption and pipelining are also crucial to optimizing latency and resource utilization. In Table. 5.3, the comparison of resource utilization and latency of the online motor model block with and

without code style adaption and pipelining is shown. It can be seen that the code style adaption significantly reduces the resource and clock cycle required for the computation core, while the pipelining improves the latency at the cost of slightly more resource utilization. Once the design in HLS is finished, it can be verified using the workflow shown in Figure 5.1.

For the sequential logic blocks, the PWM generation block consists of several counters and logic gates, designed using library blocks from the System Generator for DSP toolbox in Simulink. However, the ADC interface requires careful design. There are five physical quantities in the ME application that need to be sampled at 400 *kHz* rate. Three-phase current and two DC bus voltages, one for ME, another one for IUT. To reduce the latency, four external ADC ICs are utilized. Three of them are MAX1304, one for each phase current. The other ADC IC is ADS7865, for two DC bus voltage sensing. Since the timing requirement is much higher than the APF, each ADC IC can have a dedicated 12-bit parallel data bus and control bus to execute the conversion concurrently. This way, the conversion for all five quantities can be finished within 1.25 μs .

5.4 Verification & Performance Analysis

The implemented hardware design with routing is presented in Figure 5.8. The orange dots represent the PS core in use, while the blue dots represent the utilized PL resource. The green lines is the connection between the logic resource. The clock speed for PL (200 *MHz*) and PS (666.7 *MHz*) is kept the same as in the APF application. The motor model and resolver emulator are iterating at 1 *MHz*. Simultaneously, the current sensing and current controller are running at 400 *kHz*. The latency result shown in Figure 5.6 for each task is highly satisfactory, which is achieved since all the intensive computation is executed in PL, and the pipelining technique improves the clock speed. The results also demonstrate the effectiveness of the proposed methodology. If all the computation cores were to be implemented in PS, the additional time it takes to transmit the data between PL and PS could make the whole execution time longer than the control period, let alone the slower computation of PS. As indicated in the figure, the most time-consuming task is the ADC for current sensing. According to the timing analysis, the system's slowest path still has 0.17 *ns* slack, and there is no hold violation. If the traditional FPGA design method were to be used, this high timing requirement would be difficult to meet. For resource utilization, the I/O pins and DSP core are the most utilized resource at 51.33% and 55.62%, respectively. This is

mainly due to the various peripheral circuits like ADC and DAC needed for sensing and resolver emulation and intense computation of the model predictive control and motor model. It is worth mentioning that the utilization of BRAM is relatively high at 65.26%. However, this is mainly consumed by the Integrated Logic Analyzer (ILA) for debugging purposes. Once development is finished, the BRAM space could be freed up for other use, such as integrating the more accurate and complicated finite element analysis (FEA) based motor model. All other resource utilization rates are relatively low, as shown in Figure 5.7.

CHAPTER

6

SUMMARY & FUTURE WORK

6.1 Summary

This dissertation presents a high bandwidth ME prototype for AC machines which incorporates a unique gate stitching modulation strategy to accurately track the high frequency motor ripple current. The motor emulator system consists of a SiC based power converter with an inductive coupling, a reconfigurable, real-time model for an IPM machine implemented in FPGA iterating at 1 *MHz*, and a hybrid CCS-MPC current controller which can accurately track the emulated motor current including its ripple contents.

In Chapter 1, the background and state-of-the-art ME technology are introduced. The key motivation of this work is to advance the emulation bandwidth power hardware-in-the-loop technology utilizing fast switching WBG device and high-performance FPGA digital control platform. An overview of our proposed high bandwidth ME system is illustrated as well.

To fully utilize the available emulation bandwidth, high fidelity IPM machine model, containing non-linearity like inductance saturation, cross-coupling, and spatial harmonic effects, is discussed in Chapter 2 for real-time implementation in the FPGA platform. Fast

PWM voltage sensing circuit and resolver position sensor emulation circuit are developed so that the motor emulator can interface with IUT in a non-intrusive manner. The comparison results shows a great match between the FPGA based motor model and PLECS native motor model.

More advanced motor control techniques are investigated in Chapter 3 to demonstrate the superiority and flexibility of motor emulator and high fidelity motor model. At low fundamental frequency operation, the 6th order harmonic current contributed by the spatial harmonic of PM flux is quite dominant. Resonant controller can be added to alleviate this issue. At very high speed fundamental frequency operation, the error of rotor position at the beginning and the end of same control period would increase and cause larger error in the voltage applied to the motor. This issue can be solved with appropriate compensation terms. Moreover the impact of motor parameter, like PM flux, variation can also be studied. With the high bandwidth ME system and high-fidelity IPM motor model, the comparison and analysis of these techniques can be done in simulation and emulation, eliminate the need for dyno setup.

In Chapter 4, the detailed control and modulation strategy for the ME power converter are presented. The developed gate stitching modulation scheme for the ME power converter synchronize the switching states of ME and IUT to avoid current spikes in the system. The gate stitching strategy, when combined with the MPC, enables high control bandwidth without the need for excessively high switching frequency or complex topology. The high bandwidth of the current controller enables a high DC bus voltage to line inductance ratio that allows the use of a small line inductor. In addition, the high target motor inductance to line inductance ratio enables a wider target motor parameters' range for the motor emulator.

Chapter 5 presents an FPGA design methodology for rapid prototyping in power electronics applications. The proposed method utilizes the HLS tool and Simulink code generation blockset to minimize the need for HDL programming during the system design and expedite the verification process while not compromising the FPGA platform's performance. High FPGA clock speed (200 MHz) and low control latency are achieved with reasonable resource consumption. The methodology applies to FPGA-based digital systems available from different manufacturers for various power electronics applications. Experimental results of the motor emulator demonstrate the effectiveness of the proposed methodology based on the two key performance indicators of clock speed and resource utilization for an FPGA-based embedded system.

6.2 Future Work

There are a few aspects where this work can be improved or extended.

1. It has been illustrated in the dissertation that the isolated DC power supply are needed for IUT and ME to achieve switching ripple current tracking capability. Moreover, for EV motor drive, the DC bus voltage is varying depending on the state-of-charge of the battery pack. Traditionally, to achieve these functionality, two sets of active front end are required, one for ME, another for IUT. The isolation is done on the grid side with low frequency transformer. Apparently, this solution is very expensive and bulky. A dual port voltage variable isolated DC power supply would be a efficient solution.
2. The control bandwidth of the IUT is ever increasing, achieved by adopting either faster switching device, like SiC and GaN, or more complex inverter topology, like interleaving converter or multi-level converter. Different topology for ME power converter can be explored to achieve even higher emulation bandwidth. Interleaving topology is a very promising candidate. The increased effective switching frequency can boost the control bandwidth. Moreover, the device stress can be alleviated, especially for high power application.
3. Gate stitching modulation scheme, which synchronizes switching states of ME and IUT during IUT switching transient, is proposed to decouple the switching effect of IUT and ME without the need for large inductance or complicated inductor design. However, if other topologies for ME are selected, gate stitching modulation strategy should be adapted accordingly.
4. For PM motors, like IPM and SPM, the back-EMF is solely determined by the PM flux. However, for induction motor, the flux contains the dynamic of d-axis current (dq synchronous frame), as described in (6.1). This characteristic make the induction motor harder to be modeled in real-time with high accuracy when compared with PM motors. An efficient implementation for induction motor modeling could be a new direction for further research.

$$\lambda_{dr}^e = \frac{L_m i_{ds}^e}{1 + p \frac{L_r}{R_r}} \quad (6.1)$$

REFERENCES

- [1] “CO2 emission standards for cars and vans: Council agrees its position,” in *European Council: Brussels*, 2018.
- [2] “Vivado Design Suite User Guide: High-Level Synthesis,” 2018. [Online]. Available: https://www.xilinx.com/support/documentation/sw/_manuals/xilinx2018_3/ug902-vivado-high-level-synthesis.pdf
- [3] Y. Luo, L. Yang, J. Korhonen, W. Yu, and I. Husain, “A variable power factor high power testbed for traction inverter using back-to-back connection,” in *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2018, pp. 4591–4598.
- [4] C. S. Edrington, M. Steurer, J. Langston, T. El-Mezyani, and K. Schoder, “Role of power hardware in the loop in modeling and simulation for experimentation in power and energy systems,” *Proceedings of the IEEE*, vol. 103, no. 12, pp. 2401–2409, 2015.
- [5] W. Cao, Y. Ma, J. Wang, L. Yang, J. Wang, F. Wang, and L. M. Tolbert, “Two-stage pv inverter system emulator in converter based power grid emulation system,” in *2013 IEEE Energy Conversion Congress and Exposition*, 2013, pp. 4518–4525.
- [6] H. Alatrash, A. Mensah, E. Mark, G. Haddad, and J. Enslin, “Generator emulation controls for photovoltaic inverters,” *IEEE Transactions on Smart Grid*, vol. 3, no. 2, pp. 996–1011, 2012.
- [7] O. König, C. Hametner, G. Prochart, and S. Jakubek, “Battery emulation for power-hil using local model networks and robust impedance control,” *IEEE Transactions on Industrial Electronics*, vol. 61, no. 2, pp. 943–955, 2014.
- [8] K. Saito and H. Akagi, “A power hardware-in-the-loop-simulation (p-hils) system using two modular multilevel dsc converters for a synchronous-motor drive,” in *2017 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2017, pp. 2692–2699.
- [9] H. J. Slater, D. J. Atkinson, and A. G. Jack, “Real-time emulation for power equipment development. ii. the virtual machine,” *IEE Proceedings - Electric Power Applications*, vol. 145, no. 3, pp. 153–158, 1998.
- [10] F. E. Fleming and C. S. Edrington, “Real-time emulation of switched reluctance machines via magnetic equivalent circuits,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 6, pp. 3366–3376, 2016.
- [11] Y. Boukadida, A. Masmoudi, G. M. Casolino, F. Marignetti, A. Andreoli, and M. Albanesi, “Automotive propulsion drive emulation using two cascaded inverters,” in *2018 Thirteenth International Conference on Ecological Vehicles and Renewable Energies (EVER)*, 2018, pp. 1–7.

- [12] K. S. Amitkumar, R. S. Kaarthik, and P. Pillay, "A versatile power-hardware-in-the-loop-based emulator for rapid testing of transportation electric drives," *IEEE Transactions on Transportation Electrification*, vol. 4, no. 4, pp. 901–911, 2018.
- [13] O. Vodyakho, M. Steurer, C. S. Edrington, and F. Fleming, "An induction machine emulator for high-power applications utilizing advanced simulation tools with graphical user interfaces," *IEEE Transactions on Energy Conversion*, vol. 27, no. 1, pp. 160–172, 2012.
- [14] A. Schmitt, J. Richter, M. Gommeringer, T. Wersal, and M. Braun, "A novel 100 kw power hardware-in-the-loop emulation test bench for permanent magnet synchronous machines with nonlinear magnetics," pp. 1–6, 2016.
- [15] M. Steurer, C. S. Edrington, M. Sloderbeck, W. Ren, and J. Langston, "A megawatt-scale power hardware-in-the-loop simulation setup for motor drives," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 4, pp. 1254–1260, 2010.
- [16] S. Uebener and J. Bocke, "Application of an e-machine emulator for power converter tests in the development of electric drives," *European Electric Vehicle Congress (EEVC)*, 2012.
- [17] T. Iguchi, "Development of motor emulator provided with hil simulator for simulation of hev power control unit current during vehicle operation," in *SAE Technical Paper*. SAE International, 04 2016. [Online]. Available: <https://doi.org/10.4271/2016-01-1218>
- [18] I. U. W. A. Perterson, (EM Power, "FPGA-Based Implementation of a Predictive Current Controller for Power Converters," *US Patent 8,587,322 B2*, 2013.
- [19] S. Grubic, B. Amlang, W. Schumacher, and A. Wenzel, "A high-performance electronic hardware-in-the-loop drive-load simulation using a linear inverter (linverter)," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 4, pp. 1208–1216, 2010.
- [20] H. Yang, R. Yang, W. Hu, and Z. Huang, "FPGA-Based Sensorless Speed Control of PMSM Using Enhanced Performance Controller Based on the Reduced-Order EKF," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, pp. 1–1, 2019.
- [21] M. Hartmann, S. D. Round, H. Ertl, and J. W. Kolar, "Digital Current Controller for a 1 MHz, 10 kW Three-Phase VIENNA Rectifier," *IEEE Transactions on Power Electronics*, vol. 24, no. 11, pp. 2496–2508, 2009.
- [22] D. Jiang and F. Wang, "Current-ripple prediction for three-phase pwm converters," *IEEE Transactions on Industry Applications*, vol. 50, no. 1, pp. 531–538, 2014.
- [23] Y. Luo, M. A. Awal, L. Yang, W. Yu, and I. Husain, "Fpga based high bandwidth motor emulator for interior permanent machine utilizing sic power converter," in *2019 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2019, pp. 1–8.

- [24] Y. Lee, Y. Kwon, and S. Sul, "Dc-link voltage design of high-bandwidth motor emulator for interior permanent-magnet synchronous motors," in *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2018, pp. 4453–4459.
- [25] J. Wang, Y. Ma, L. Yang, L. M. Tolbert, and F. Wang, "Power converter-based three-phase induction motor load emulator," in *2013 Twenty-Eighth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2013, pp. 3270–3274.
- [26] L. Zhu, D. Jiang, R. Qu, L. M. Tolbert, and Q. Li, "Design of power hardware-in-the-loop simulations for integrated starter-generator systems," *IEEE Transactions on Transportation Electrification*, vol. 5, no. 1, pp. 80–92, 2019.
- [27] S. Lentijo, S. D'Arco, and A. Monti, "Comparing the dynamic performances of power hardware-in-the-loop interfaces," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 4, pp. 1195–1207, 2010.
- [28] A. Schmitt, M. Gommeringer, C. Rollbuhler, P. Pomnitz, and M. Braun, "A novel modulation scheme for a modular multiphase multilevel converter in a power hardware-in-the-loop emulation system," in *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*, 2015, pp. 001 276–001 281.
- [29] C. Nemec, O. Lehmann, M. Heintze, and J. Roth-Stielow, "Optimal inductor setup for a power-hardware-in-the-loop machine emulator," in *2013 IEEE 10th International Conference on Power Electronics and Drive Systems (PEDS)*, 2013, pp. 364–369.
- [30] H. Zhan, Z. Zhu, and M. Odavic, "Analysis and suppression of zero sequence circulating current in open winding pmsm drives with common dc bus," *IEEE Transactions on Industry Applications*, vol. 53, no. 4, pp. 3609–3620, 2017.
- [31] J. Korhonen, J. Honkanen, T. J. Kärkkäinen, J. Nerg, and P. Silventoinen, "Modulation and control methods to reduce zero sequence current in open-end winding motors," in *2017 IEEE International Electric Machines and Drives Conference (IEMDC)*, 2017, pp. 1–6.
- [32] Y. Srinivasa Rao and M. C. Chandorkar, "Real-time electrical load emulator using optimal feedback control technique," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 4, pp. 1217–1225, 2010.
- [33] M. Antivachis, J. A. Anderson, D. Bortis, and J. W. Kolar, "Analysis of a synergetically controlled two-stage three-phase dc/ac buck-boost converter," *CPSS Transactions on Power Electronics and Applications*, vol. 5, no. 1, pp. 34–53, 2020.
- [34] M. Antivachis, D. Bortis, A. Avila, and J. W. Kolar, "New optimal common-mode modulation for three-phase inverters with dc-link referenced output filter," *CPSS Transactions on Power Electronics and Applications*, vol. 2, no. 4, pp. 331–340, 2017.

- [35] Q. An, J. Liu, Z. Peng, L. Sun, and L. Sun, "Dual-space vector control of open-end winding permanent magnet synchronous motor drive fed by dual inverter," *IEEE Transactions on Power Electronics*, vol. 31, no. 12, pp. 8329–8342, 2016.
- [36] S. Li, D. Han, and B. Sarlioglu, "Modeling of interior permanent magnet machine considering saturation, cross coupling, spatial harmonics, and temperature effects," *IEEE Transactions on Transportation Electrification*, vol. 3, no. 3, pp. 682–693, 2017.
- [37] F. Briz, M. W. Degner, and R. D. Lorenz, "Analysis and design of current regulators using complex vectors," *IEEE Transactions on Industry Applications*, vol. 36, no. 3, pp. 817–825, 2000.
- [38] H. Cho, Y. Kwon, and S. Sul, "Optimal current trajectory control of ipmsm for minimized torque ripple," in *2019 IEEE Transportation Electrification Conference and Expo (ITEC)*, 2019, pp. 1–6.
- [39] *Control of Electric Machine Drive Systems*. Sul, Seung-Ki, 2010, ch. 7, pp. 324–353.
- [40] P. Cortes, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodriguez, "Predictive control in power electronics and drives," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 12, pp. 4312–4324, 2008.
- [41] M. A. Awal, W. Yu, and I. Husain, "Passivity-based predictive-resonant current control for resonance damping in lcl-equipped vscls," *IEEE Transactions on Industry Applications*, vol. 56, no. 2, pp. 1702–1713, 2020.
- [42] L. Yang, Y. Luo, M. A. Awal, W. Yu, and I. Husain, "Application of high performance fpga to boost bandwidth of sic shunt active power filter," in *2019 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2019, pp. 6454–6461.
- [43] D. Rahman, A. J. Morgan, Y. Xu, R. Gao, W. Yu, D. C. Hopkins, and I. Husain, "Design methodology for a planarized high power density ev/hev traction drive using sic power modules," in *2016 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2016, pp. 1–7.
- [44] S. Ji, M. Laitinen, X. Huang, J. Sun, W. Giewont, F. Wang, and L. M. Tolbert, "Short-circuit characterization and protection of 10-kv sic mosfet," *IEEE Transactions on Power Electronics*, vol. 34, no. 2, pp. 1755–1764, 2019.
- [45] G. Martin and G. Smith, "High-level synthesis: Past, present, and future," *IEEE Design Test of Computers*, vol. 26, no. 4, pp. 18–25, 2009.
- [46] S. Lucia, D. Navarro, Lucía, P. Zometa, and R. Findeisen, "Optimized FPGA Implementation of Model Predictive Control for Embedded Systems Using High-Level Synthesis Tool," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 1, pp. 137–145, 2018.

- [47] S. Wendel, A. Dietz, and R. Kennel, "FPGA based finite-set model predictive current control for small PMSM drives with efficient resource streaming," in *2017 IEEE International Symposium on Predictive Control of Electrical Drives and Power Electronics (PRECEDE)*, 2017, pp. 66–71.
- [48] E. Guerrero, E. Guzmán, J. Linares, A. Martínez, and G. Guerrero, "Fpga-based active disturbance rejection velocity control for a parallel dc/dc buck converter-dc motor system," *IET Power Electronics*, vol. 13, no. 2, pp. 356–367, 2020.
- [49] D. Navarro, Lucía, L. A. Barragán, I. Urriza, and Jiménez, "High-level synthesis for accelerating the fpga implementation of computationally demanding control algorithms for power converters," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 3, pp. 1371–1379, 2013.
- [50] O. Gulbudak and E. Santi, "Fpga-based model predictive controller for direct matrix converter," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 7, pp. 4560–4570, 2016.
- [51] R. Wiśniewski, G. Bazydło, P. Szcześniak, and M. Wojnakowski, "Petri net-based specification of cyber-physical systems oriented to control direct matrix converters with space vector modulation," *IEEE Access*, vol. 7, pp. 23 407–23 420, 2019.
- [52] M. Hamouda, H. F. Blanchette, K. Al-Haddad, and F. Fnaiech, "An efficient dsp-fpga-based real-time implementation method of svm algorithms for an indirect matrix converter," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 11, pp. 5024–5031, 2011.
- [53] "Vivado Design Suite Tutorial: Model-Based DSP Design Using System Generator," 2020. [Online]. Available: https://www.xilinx.com/support/documentation/sw_manuals/xilinx2020_1/ug948-vivado-sysgen-tutorial.pdf
- [54] A. Finnerty and H. Ratigner, "Reduce Power and Cost by Converting from Floating Point to Fixed Point," 2020. [Online]. Available: https://www.xilinx.com/support/documentation/white_papers/wp491-floating-to-fixed-point.pdf

CHAPTER

7

FPGA CONTROLLER PCB DESIGN TIPS

The design of the FPGA control board is very critical to the system performance. On the FPGA carrier board, we need to integrate high speed signals, for Ethernet, and high power signals, for current and voltage sensors. There are four pairs of differential signals for the Ethernet RJ45 connector. Those signal traces for high speed Ethernet (100 Mbps or 1 Gbps) requires accurate impedance matching, otherwise the delay mismatch between different

Figure 7.1: The traces of the four differential signals for Ethernet RJ45 connector with impedance matching to minimize propagation delay among the channels.

Figure 7.2: The local ground planes to sandwich the high speed Ethernet signal traces.

channels could degrade the link speed. A good layout example with impedance matching is presented in Figure 7.1. In the first version of the board, the impedance matching is not perfect, resulting in de-rated link speed from 1 Gbps to 100 Mbps. Moreover, local ground planes should be added in the neighboring layers to shield the high speed signals, as shown in Figure 7.2. These local ground planes should be connected to the main ground plane in multiple points to avoid potential variation.

Figure 7.3: The current sensor layout with appropriate isolation barrier.

The current sensor used for the ME application is IC type (ACS732 from Allegro), and the main current trace should be design on the board. The current sensor IC can be treated as

isolation barriers. On the low voltage side are the power supply and sensor output signal. On the high voltage side are the current traces, connected to the switching node. The signals on two sides should not overlap (similar to the gate driver isolation barrier design), as illustrated in Figure 7.3. Otherwise the isolation could get corrupted, introducing noise to the system. In the worst case, the power supply to the FPGA could get disturbed. It is worth noting that the high power current trace should be assigned to a layer furthest away from the low voltage signals. Another trick to assure isolation is to manually adjust the clearance between different layers of copper. For instance, if the high voltage side trace is assigned to the bottom layer, then the clearance of the bottom layer should be set to be larger than the clearance between the low voltage signal layers.

Lastly, for application in even worse EMI environment, a metal dog cage can be added above the FPGA chip as a shielding cage. This cage is electrically connected to the main ground layer of the PCB. Furthermore, an input filter inductor can be added for the main 12 V power input of the board to filter out the external disturbance from the bench power supply.

CHAPTER

8

VARIABLES

A summary of all variables is documented in Table 8.1.

Table 8.1: A summary of common meteorological variables and their abbreviations in alphabetical order.

Variable	Abbreviation
Angular momentum	J
Control period of motor emulator	$T_{c,ME}$
D-axis stator voltage	v_{ds}
D-axis inductance	L_d
D-axis stator flux	λ_{ds}
D-axis current	i_{ds}
DC bus voltage	V_{DC}
Dead-band time of IUT	$t_{Db,IUT}$
Dead-band time of ME	$t_{Db,ME}$
Duty cycle	d
Electric angular speed	ω_e
Electric angle	θ_e

Electric torque	T_e
Line inductance	L_l
Load torque	T_L
Mechanical angular speed	ω_m
Permanent magnet flux	λ_{PM}
Pole number	P
Q-axis stator voltage	v_{qs}
Q-axis inductance	L_q
Q-axis stator flux	λ_{qs}
Q-axis current	i_{qs}
Switching state	S
Sampling time	T_s

CHAPTER

9

ACRONYMS

A summary of all acronyms is documented in Table 9.1.

Table 9.1: A summary of acronyms used in alphabetical order.

Acronym	Abbreviation
Analog-to-digital conversion	ADC
Digital-to-analog conversion	DAC
Electric vehicle	EV
Field programmable gate array	FPGA
Field-oriented control	FOC
High level synthesis	HLS
Hardware-in-the-loop	HIL
Inverter under test	IUT
Interior permanent magnet	IPM
Induction machine	IM
Model predictive control	MPC
Motor emulator	ME
Pulse width modulation	PWM

Programmable logic	PL
Processing system	PS
Resolver-to-digital converter	RDC
Space vector modulation	SVM
Synchronous reluctance machine	SRM
Continuous control set	CCS
Finite control set	FCS
Register-transfer level	RTL