Modeling and Simulation of FPGA based direct Torque Control of Induction Motor Drive.

Amol A. Kalage[1], Vasant M. Jape[2], Sarika V. Tade[3], Manik M. Hapse[4] and Ravindra G. Dabhade[5]
[1] Govt. College of Engineering/Department of Electrical Engineering, Amravati, India.
Email: amolkalage@rediffmail.com
[2] Govt. College of Engineering/Department of Electrical Engineering, Amravati, India.
[3] S.N.D. College of Engineering/Department of Electrical Engineering, Yeola, India.
[4] P.D.V.V.P. College of Engineering/Department of Electrical Engineering, Ahmednagar, India.
[5] S.N.D. College of Engineering/Department of Electrical Engineering, Yeola, India.
Email :{ vasantmjape@yahoo.com , sarika.tade@rediffmail.com, hapsemanik@rediffmail.com,
rgdabhade@rediffmail.com }

***Abstract*—Modern AC drives require a fast digital realization of many mathematical operations concerning control and estimator's algorithms, which are time consuming. Therefore developing of custom-built digital interfaces as well as digital data processing blocks and sometimes even integration of ADC converters into single integrated circuit is necessary. Due to the fact that developing an ASIC chip is expensive and laborious, the FPGA based solution should be used on the design stage of the algorithm. In this paper the application of FPGA in Direct Torque control induction motor drive is presented**

***Index Terms*— Induction motor, DTC, FPGA.**

I. Introduction

The rapid development of very large scale integration (VLSI) technology in recent years created an opportunity for the development of complex and compact high performance controllers efficient control of the induction motor drive which is a complex system having components of different nature: power converters built from linear and nonlinear electric elements, electrical machines and control system containing linear and nonlinear elements. Effective modeling and simulation of such systems require software tool that can handle all these functions in an integrated environment

Now a days FPGA are replacing DSP in many applications due to some advantages In an FPGA multiple operations can be executed in parallel so that algorithm can run much faster which are required by control system. And if the algorithm is to be implemented on prototyping platform it is desirable to have simulation software that can represent exactly the hardware used. The available simulation software for electronic circuits or dynamic circuits can be classified into two main categories: circuit simulation programs such as EMTP and equation solver program such as MATLAB/Simulink. These programs are not designed specifically for power electronic systems; therefore it is necessary to develop a model to fulfill the needs. Which can give good results but it requires considerable efforts.

In this paper a method for modeling and simulation of Induction motor drives using MATLAB/Simulink is presented. As field programmable gate array (FPGA) is used to run the algorithm, a software Xilinx system generator, a toolbox of MATLAB/Simulink can be used. It will simulate the hardware as well as generate the VHDL code needed for the implementation in FPGA. It can automatically convert the model into VHDL code. In .this paper the application of FPGA for the direct torque control of Induction motor drives is described and simulation results are presented

II. DESCRIPTION OF POWER SYSTEM BLOCKSET

Matlab/Simulink is a systems simulator and unable to direct simulate electrical circuits so to simulate electrical circuits power system blocksets are used which is having libraries of electrical blocks and analysis tools which are used to convert electrical circuits into Simulink diagrams. The electrical blocks are electrical models such as electrical machines, current and voltage sources, and different electric elements, power electronic switches, connectors, and sensors for measurement purpose. When the simulation starts Simulink use the Pm Blockset and transfers the electrical circuit into a state–space representation with the initial conditions of state variables. The actual simulation starts after this initial conversion, This allows the use of a wide variety of fixed step and variable step algorithms available in Simulink. As variable time step algorithms are faster than fixed time step method because the number of steps are less so these algorithms are used for small- and medium-size systems, And for large systems containing a more number of states and/or power switches, a fixed time step algorithm is used. A Simulink scopes can be used to display the Simulation results or these results can be sent to workspace during the simulation. The variety of MATLAB functions and toolboxes are present for processing and plotting of waveforms from stored data.

III. DESCRIPTION OF XILINX SYSTEM GENERATOR

Matlab/Simulink consists of Xilinx System Generator Blockset, which is developed by Xilinx. Due to development of Xilinx System Generator Blockset the use of high level programming tool for FPGA users become very easy. The Xilinx System Generator Blockset consists of a library of FPGA blocks, required for buildings of model. It also consists a VHDL code generator which automatically generates VHDL code from the created models. The Xilinx System Generator Blockset can work only with Xilinx FPGAs. For implementation of VHDL code in an FPGA an additional conversions are required. Initially the generated VHDL codes are transformed into a standard schematic circuit and then the schematic circuit is transformed by an implementation tool into a configuration file that can be downloaded to the FPGA.

The main advantage of using the Xilinx System Generator Blockset to simulate the control algorithm is the possibility to generate code that can be used to program an FPGA directly from the simulation model. Furthermore, the implemented code is guaranteed to perform exactly as in the simulation because the Xilinx System Generator Blockset represent exactly the FPGA hardware. This reduces the implementation time because the algorithm needs to be modeled and simulated only once.

The Xilinx System Generator Blockset library contains many blocks that are classified as follows:

1. Basic elements: Contains simple signal processing functions (delay, multiplexers, registers, …..).
2. Math: Contains common mathematical functions (add, multiplication, relational, negate, . . .).
3. MATLAB I/O: Contains blocks to connect the SG section to the Simulink model and determine the quantization error created by the use of fixed-point values.
4. Memory: Contains memory elements found in FPGA (RAM, ROM, FIFO, . . .)
5. DSP: Contains DSP like math functions (FIR, FFT, . .)

Many of these blocks are replicas of standard Simulink blocks. So it is easy to build a model the same way than with standard Simulink blocks. To connect the SG algorithm to the remainder of the Simulink model, two specials blocks are used. “Gateway In” transforms an incoming floating point signal to a fixed-point one with a fixed time step. “Gateway Out” is used to return a value to the normal Simulink environment. These two blocks represent the boundaries of the SG model. Everything inside them can be implemented. A special block (Xilinx System Generator) is used to generate the implementation VHDL code from the model. The conversion is automatic. The only thing needed to be specified is the FPGA model that will be used. The generated code can be then used to incorporate the control algorithm in an FPGA.

IV. MODELING AND SIMULATION USING MATLAB/SIMULINK.

The formation of a Simulink model for the power circuit and the control algorithm requires few steps which can be separated in two main parts.

1. Design and testing of the system in MATLAB environment.
2. Generation and implementation of the VHDL code in the FPGA.

The detailed step are shown in the flow diagram in Fig. 1.The first step is to determine the parameters of the power electronic section and use them to create a model using the PSB blocks. Also the hardware section of the prototyping board like the ADC can be simulated with a mathematical model. In this way, the control algorithm will be simulated as an exact replica of the real system. The next step is the building of the control algorithm with the use of only Xilinx System Generator blocks. It is then connected to the simulated power system by “Gateway In” and “Gateway Out” blocks. Fig. 2 shows the interconnections between the control and the power circuit. The next step is the simulation of the complete system in Simulink and the VHDL code generation which is done with the “Xilinx System Generator” block. The process is totally automatic except the choice of the FPGA model required. Normally, the FPGA contains some other functions in addition to the algorithm. Therefore the algorithm must be connected to the remainder of the VHDL code. This code is created only once and comes with the prototyping platform. A standard interface or a VHDL software can be used to join all these codes. The last step is the synthesis and implementation of the VHDL code. This is done with Xilinx software which requires a few simple steps. After this, a configuration file is obtained permitting programming the FPGA. If some modifications of the system are required, only some steps need to be done again so the process will be much faster. The Matlab/Simulink model is shown in Fig. 3.

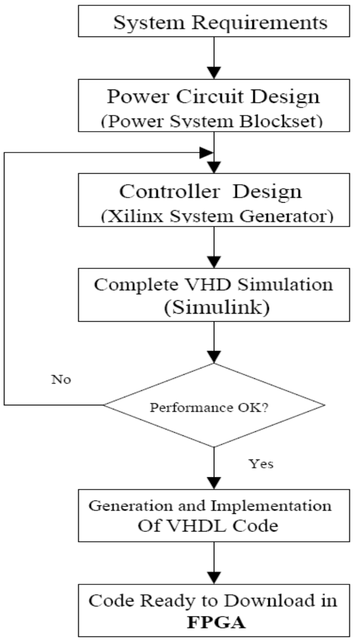


Fig. 1. Induction Motor drive controller design and implementation process.

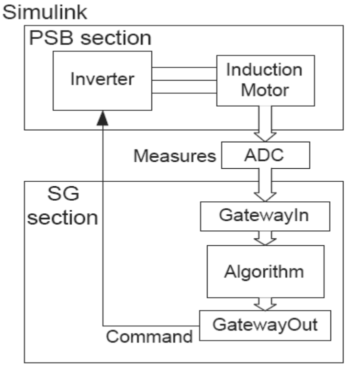


Fig. 2. Interconnection between power and control circuit.

V. A DTC CONTROLLER FOR A INDUCTION MOTOR DRIVE.

A Direct Torque control algorithm of Induction Motor Drive has been modeled and implemented on an FPGA-based prototyping platform. This algorithm has been chosen because it has wide range of application and offers a good level of complexity. The formation of algorithm model is very simple because almost all the blocks required are present in the Xilinx System Generator library. A simulation of the original algorithm side by side with the Xilinx System Generator is done to check and compare the performance

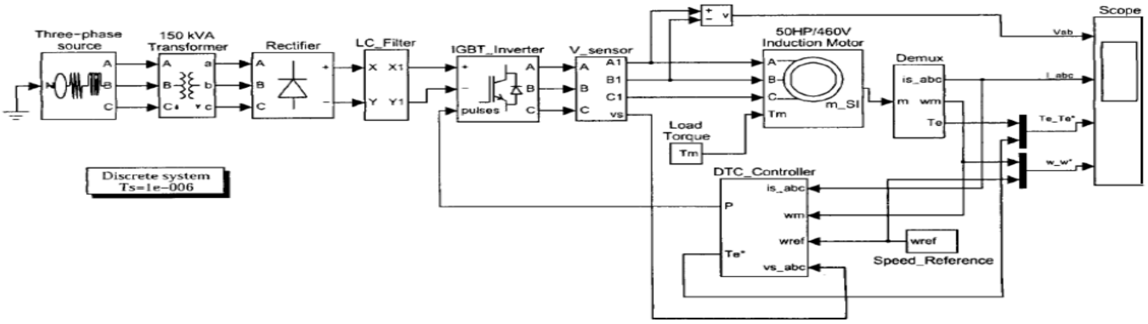
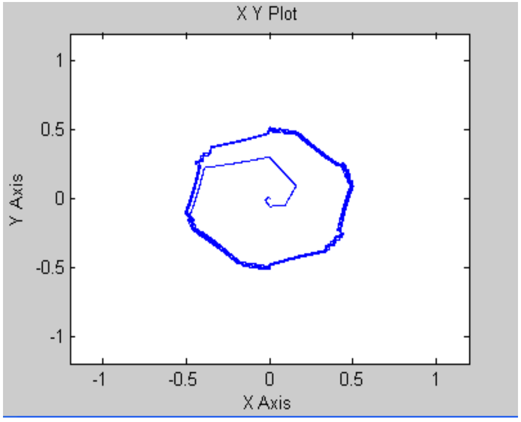


Fig 3. – Simulink diagram representing DTC Induction Motor Drive.

SIMULATION RESULTS



Locus of the stator flux

VI. CONCLUSION

A new approach to simulate complete Direct Torque control of Induction Motor drive systems using Matlab/Simulink (Power System Blockset, Xilinx System Generator) has been presented. This approach gives many advantages when the control algorithm is to be implemented for an FPGA-based system. As the modeling of the algorithm has to require doing only once it saves a lot of time also the simulation done using SG blocks represents exactly the hardware used to run the algorithm. This reduces

the time to implement the algorithm because there are no modifications in the hardware. The implementation of the DTC algorithm has proved that the use of the SG in combination with the PSB is powerful and easy. Implementation time was reduced and the control algorithm was working as expected on the hardware platform on the first try.

VII. REFERENCES

1 Hoang Le-Huy..'Modeling and Simulation of Electrical Drives using MATLAB/Simulink and Power System Blockset'. IEcon'01: The 27th Annual Conference of the IEEE Industrial Electronics Society.

2 Czeslaw T. Kowalski, Jacek Lis, Teresa Orlowska-Kowalska,. 'FPGA Implementation of DTC Control Method for the Induction Motor Drive'. EUROCON 2007 : The International Conference on "Computer as a Tool" Warsaw, Sept. 9-12,2007

3. Cristian Lascu, Ion Boldea, 'A Modified Direct Torque Control for Induction Motor Sensorless Drive'. IEEE Transactions on industry applications, Vol. 36, No. 1 Jan/Feb 2000.

4. Eric Monmasson and Marcian N. Cirstea, 'FPGA Design Methodology for Industrial Control Systems—A Review' IEEE Transactions on industrial Electronics, Vol. 54, August 2007.