FPGA Implementation of a DB – DTFC Scheme for Induction Machines

Gabriel Alejandro Dominguez Olavarría

Department of Measurement Technology and Industrial Electrical Engineering
Division of Industrial Electrical Engineering and Automation,
Faculty of Engineering, Lund University, Sweden.

Abstract——controlling an induction machine has been a difficult task since its invention, especially at low speed operation and during highly dynamic regimes, where the required torque/speed changes constantly. Several methods have been developed in order to solve those limitations. One of the newest approaches to this problem is Deadbeat Direct Torque and Flux Control (DB-DTFC). This technique is able to reach the torque and flux references over one sampling period with a good performance both in steady and transient state.

This thesis aims to go through the whole process of simulation, implementation and validation of a DB-DTFC system.

1. INTRODUCTION

Deadbeat Direct Torque and Flux Control is a recently developed method for controlling induction motors. It combines some aspects of Field Oriented Control (FOC) and Direct Torque Control (DTC). With FOC shares that both use Space Vector Modulation (SVM) supported by Pulse Width Modulation (PWM) and reference frame transformations, therefore providing fixed switching frequency. With DTC shares that both control the torque and flux magnitude directly without intermediate current loop. However the DB-DTFC solution uses an inverse model of the induction machine to calculate the theoretical voltage vector needed to move the machine torque and stator flux to the desired values.

In order to implement this technique high sampling and computational rates are needed.

Besides, according to all the bibliography found, this technique has only been implemented in a micro controller and exclusively by the Departments of Mechanical Engineering and Electrical and Computer Engineering, University of Wisconsin, Madison.

For the last two reasons, it seems an interesting project to implement this control system on a different platform and to verify if the claimed performance is reachable.

The implementation is done into a new platform from National Instruments (NI), the CompactRIO (cRIO). This device combines an FPGA, a Real Time micro controller, a computer and a variety of external modules into a complex and flexible system that allows high sampling and computational rate, task division in different levels and remote monitoring via Ethernet.

2. IMPLEMENTATION

2.1 OBSERVERS

In order to be able to implement a DB-DTFC scheme, information about the flux linkages is required. Since it is not possible to measure these values directly, flux observers need to be implemented. A flux observer is basically a mathematical model of the induction motor that is feeded with the same inputs as the real machine. There are several ways to implement a flux linkage observer. However, the different implementations have different frequency response and parameter sensitivity.

Since this control scheme is pretended to work in a wide frequency range, the flux estimation must be accurate within the same range. In order to obtain this performance, two different flux observers were implemented. The firs observer is based on the stator current which provides a good performance at low frequencies, been sensitive almost only to the mutual inductance. The second observer is based on the stator voltage and it works better at high frequencies.

Both observers are combined into a more robust and reliable observer by placing a PI controller in between as it is shown in Figure 1. This controller provides a soft transition between the two models. The obtained observer is called a “Gopinath style observer”.

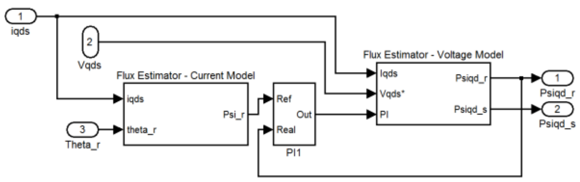

Figure 1: Gopinath style Observer

2.2 CONTROL LAW

The aim of a DB-DTFC scheme is to reach both the torque and flux references over one sampling period. In order to do this, the voltage time vectors that produce the desired change in torque must be calculated. When the torque equation is derivated in discrete time, assuming that the rotational speed is constant over one sampling period the obtained expression is a straight line in the voltage time plane. All the points on that line produce the desired change in torque but they result on different stator flux values.

Later on, the voltage time vectors that yield the stator flux reference need to be calculated. When the stator flux equation is worked in discrete time and moved into the stator flux reference frame it is possible to observe that the loci of the voltage time vectors that achieve the desired stator flux module is a circle in the voltage time space. In order to represent the change in the stator flux, the centre of the circle is shifted to the position of the flux in the previous sampling period.

The voltage time vectors that achieve the desired torque and flux reference at the same time are at the intersections between the flux circle and the torque line. But the DC link voltage limitation needs to be considered. This limitation is represented in the voltage time space as a hexagon.

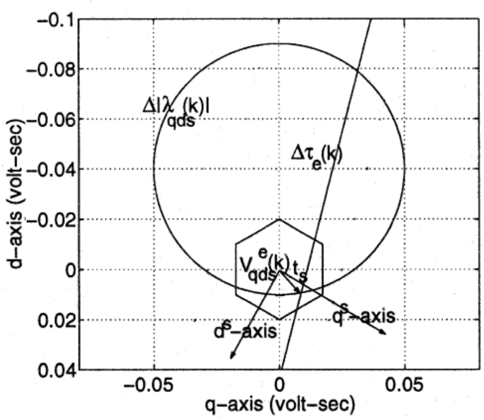

Figure 2: Voltage time vector selection

Figure 2 shows the straight line that represents the desired change in torque, the circle representing the desired change in the stator flux and the hexagon representing the DC link voltage limitation. If there is one intersection between the torque line and the flux circle that falls inside the hexagon, then both references can be reached over one sampling period.

3. RESULTS

3.1 SIMULATIONS

The first part of this thesis consists in verifying the performance of the control system. In order to do that a Simulink model is implemented, and several tests are performed.

Figure 3 shows the system response when a flux step is commanded. It can be observed that the settling time is higher than one sampling period, this is due to the DC link voltage limitation, which makes impossible to reach the reference in one sampling time. It is worth noticing that no overshoot or ripple are produced.

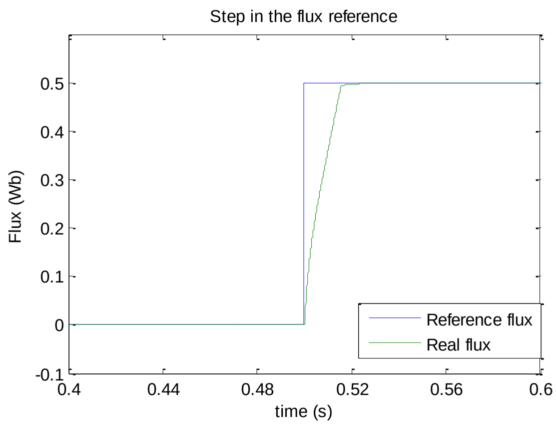

Figure 3: Step in the flux reference, Simulation

Figure 4 shows the system response when a step is applied to the torque reference. As in the previous case, the reference in not reached over one sampling period, due to the voltage limitation, and no ripple or overshot are produced.

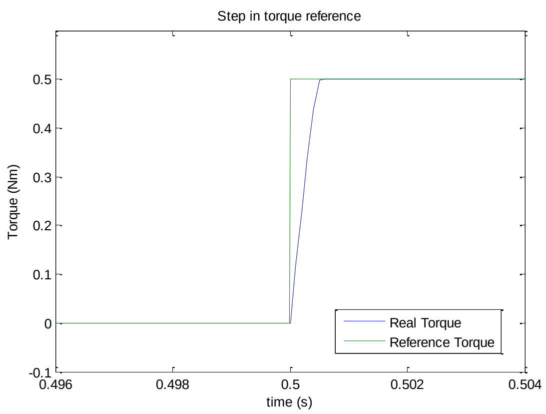

Figure 4: Step in the torque reference, Simulation

3.2 EXPERIMENTAL RESULTS

The physical implementation of the control system is done on the cRIO using LabVIEW as the programming language. As it was explained before, a multi level code is developed in order to perform all the different tasks. The critical operations like signal sampling, flux estimation and voltage vector calculations are performed at the FPGA level. Parameter calculations and rotor position monitoring are performed in the Real Time micro controller. All the plotting and data acquisition is performed at the PC level.

After the implementation is completed and validated some test are conducted in order to verify the performance of the control system.

Figure 5 shows the system response when a flux step is commanded. As it is explained in the simulation section, the reference is not reached over one sampling period due to the DC link voltage limitation. However, it may be observed that no overshoot is produced and the flux ripple is not significant.

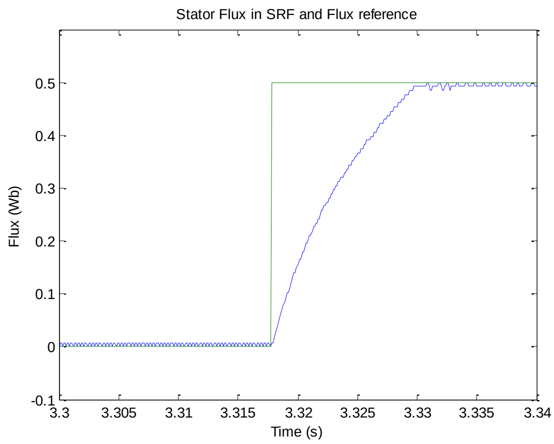

Figure 5: Step in the flux reference, Experimental

The next test consist in applying several steps in torque while the rotor is blocked (the rotational speed is kept at 0 rad/s). In Figure 6 it can be noticed that the dynamical response of the system is quite fast and the torque ripple is not significant. This satisfies the initial expectations of the control system performance.

No test is performed with a free rotor since it does not provide any useful information. When a step in torque is commanded, the rotational speed of the induction motor increases rapidly due to the low inertia of the rotor, until the induced voltage has almost the same value as the DC link voltage. At this point, there is no remaining voltage left to keep the torque at the reference level. Therefore, a way to keep the rotational speed at reasonable levels during the tests is needed.

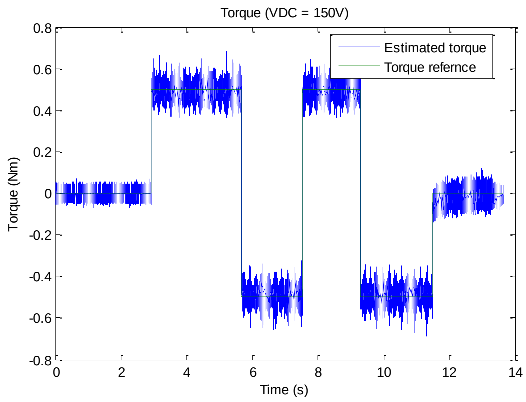

Figure 6: Step in the torque reference, Experimental blocked rotor

One way to have some control over the rotational speed of the induction motor is to use another motor as a load. A fast and simple alternative is to connect a DC motor to the test motor shaft, and feed it with a constant voltage. In this way, when a torque is commanded to the induction motor, the rotational speed of both machines will reach a constant value in steady state (depending on the applied torque).

Figure 7 shows the response of the control system when several steps in torque are applied while the induction motor is connected to a DC motor that is being supplied with a constant DC voltage.

It is worth noticing that during the second and third steps in torque the rotational speed is close to zero. This represents one of the most difficult operation points for the flux observers, but is possible to observe that the control system never lose the references and it keeps the rotational speed at zero without any problem.

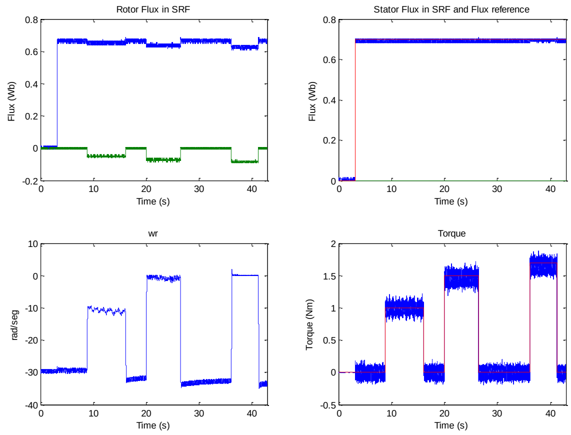

Figure 7: Results for VDC = 150V; Initial rotational speed -30 rad/s (Red: reference; Blue: measured value/d component; Green: measured value/q component)

4. CONCLUSIONS

The Deadbeat Direct Torque and Flux Control (DB-DTFC) strategy implemented relies on the stator and rotor flux linkage estimations, from which the torque is calculated. Therefore, it is crucial to have accurate and reliable observers. The overall performance of the Gopinath style flux observer implemented depends first on the accuracy obtained in the motor parameters during the characterization process, and second on the exactitude of the rotor position.

Implementing such a complex control system on an FPGA using LabVIEW is a challenging task. Several aspects should be considered, since they will determine the correct operation of the system, such as the size of the fixed point variables, the execution order, the resources required vs. the available resources in the FPGA, etc. However, at the same time it has significant advantages, like extremely precise timing, parallel execution and high sampling and computational rate.

Finally, from the experimental results presented, it can be concluded that the control system implemented is robust and provides fast response to changes in the torque and flux references, with a low ripple and nearly no overshoot.