CHAPTER 2

FPGA BASED CONTROLLERS IN POWER CONVERTER SYSTEMS

2.1 OVERVIEW

The last two decades of advances in microcontrollers, Digital Signal Processors (DSPs), and Field Programmable Gate Arrays (FPGA) have opened up tremendous possibilities for enhancing the performance, applicability, and economy of power electronic systems and drives. The benefits of programmable digital logic like, flexible reprogramming during design and development, the possibility to include the value-added functions such as sophisticated user interface, ability to implement multi-input/multi-output control strategies, easy compatibility and the reconfiguration codes as background processes offered real commercial value to practical power electronic products.

2.2 REVIEW OF DIGITAL CONTROLLERS

There are different factors in realization of practical digital controllers for power electronic systems. Major practical issues complicating the realization of a high performance digital controller include, selection of control processor, determination of sampling rate, interfacing between the controller and the power circuit, hardware design, firmware design and software realization of the control algorithms. They are not trivial task, but they need very careful design with practical perspectives. Software
implementation plays a key role in designing a practical controller. It must be analyzed from a theoretical point of view with practical constraints. The digital control technology can effectively contribute to power savings through adaptive gate-drive timing for power semiconductors and multi-mode operation to maintain high efficiency over a wide range of operating conditions.

Digital control technology has the potential to offer a number of performance enhancements in power electronic applications. Some of them are as follows;

- Digital components are less susceptible to aging and environmental variations.
- They are less sensitive to noise.
- The programmable digital control systems have the flexibility of changing the controller without hardware alteration.
- Digital control technology opens the possibility of implementing more advanced control concepts which are impractical with standard analog circuits.

Kazuki Sugahara et al (2009) proposed a typical digital power electronics control system as shown in Figure 2.1. The selection of control device and the specification of A/D converter were the important factors to design the digital control system. Some specifications for the A/D converter have been given which are desirable for high performance digital control for power electronics system. They are;
- Sampling rate ($\geq 100$KHz, 1MHz or more desirable)
- Number of channel ($\geq 2$ ch, up to 8ch or more)
- Conversion Time ($\leq 1\mu s$)
- Input Voltage Range ($\geq 5$V pp, up to 20V pp)

![Figure 2.1](image_url)

**Figure 2.1** Block diagram of typical digital control for power electronics system

There are different kinds of digital controllers. The first generation digital controllers is the microcontroller. Sen (1990) stated that, the microprocessors, microcontrollers, and microcomputers have the tremendous impact on power electronics since early 1980s. Different from analog controllers, they can enable the implementation of sophisticated and complex control techniques with computer programs in a much easier way. In the 1980s, the single-chip microcontrollers (such as the Intel16-byte 8096), the 32-byte microprocessors such as the Motorola 68020, Intel 80386, Zilog’s 2800 and microcomputers had already provided the abilities to perform the dedicated and flexible jobs. Moreover, they enabled the implementation of modern control theories (such as vector control, siding-mode control, model-
reference-adaptive control (MRAC), fuzzy control, and state and parameter estimation for high performance drives, as pointed out by Bose (2001).

The Digital Signal Processor (DSP) is a specialized microprocessor, which has the special characteristics like, faster program execution due to its Harvard architecture that permits the overlap of instruction fetch and execution of consecutive instructions, usage of dedicated hardware multiplier and barrel shifter which permits the functions in one instruction cycle time and its suitability for extremely complex math-intensive tasks. Because of these characteristics, DSPs are very common in power converters control since it exploits their mathematical oriented resources. Many arithmetic operations are provided in a DSP to meet the demand of complex algorithms.

Due to the sequential operations and shared resources like memory buses, DSPs are not very common in high switching frequency applications or applications that require massively parallel calculations. Data loss may occur during the transfer of data and the additional cost is needed to solve these problems. Min (1994) pointed out that, if the multiple-loop schemes of the motor controller are to be realized by a DSP, most of the computation resource will be devoted to the inter current-loop and PWM gating signal generations and so only very few computation resources will be left for the other control loops and this will adversely affect the whole control system.

The tendency to use concurrent hardware for the control purpose, results in a custom hardware solution of implementing the digital control scheme in a FPGA instead of DSP. The continuous and simultaneous execution of all the internal logic elements of FPGA and also all the control procedures allows the usage of high-speed demanding algorithm for power electronics system control.
The ability of carrying out parallel processing by means of hardware mode enables a system which operates at high speed with good precision. In comparison with the Application Specific Integrated Circuits (ASICs), whose high-speed hard-wired logic which enhances the computation capability and thus relieve the DSP load factor, the FPGA supports system reconfigurability and thereby readily meet the requirements of the industrial drives which are characterized by rapid evolution and diversified applications. In recent years, the researchers are keeping more attention in FPGAs due to its shorter design cycle, lower cost, higher density and high calculation speed. Fratta et al (2004) discussed the new digital control properties of FPGA-based techniques on the basis of a comparative analysis in terms of performances and immunity to PWM environment. All the possible sampled control of DSP techniques were analyzed and compared with FPGA based technique.

The clear technical advantages of digital control combined with the tremendous growth of the processing power of FPGA device at ever decreasing cost resulted in widespread adoption of digital control technology in power electronic applications. As pointed out by Eric Monmasson et al (2011), the Simplicity and programmability of FPGA make it the most favorable choice for prototyping digital systems. FPGAs are increasingly becoming popular, as implementation platforms on which the control algorithms can be implemented by programming reconfigurable hardware logic resources of the device.

2.2.1 Challenges in implementing Digital Controllers

The implementation of digital controllers for power electronics applications is somewhat challenging one. Even though some advantages can be obtained with digital control systems, there are some issues that should be carefully considered. Some of the issues are;
- Limited Analog to Digital Conversion (ADC) resolution.
- Limited digital PWM resolution.
- Inherent sampling time delay and limit cycling.

The resolution is limited mainly by the ADC and the PWM. However, the ADC resolution is becoming a less important problem, it can be improved by the windowed ADC technique proposed by Peterchev et al (2003) and the PWM resolution needs to be higher than the ADC resolution for avoiding limit cycling as pointed out by Peterchev and Sanders (2003).

The microprocessor based control has achieved the significant acceptance and it is applied in relatively low bandwidth applications like outer loop controllers in motor drives and as supervisory or sequencing controllers in high-end computer systems with complicated power distribution architectures. Relatively high-bandwidth digital control of power supplies seemed to be economically and technically challenging as quoted by Steven (2003). Dragam Maksimovic et al (2004) reviewed some technical challenges of implementing digital control for high frequency Switched Mode Power Supply (SMPS) and showed the impact of digital technology scaling and integration in implementing a simple and practical high performance digital controllers. Steven (2004) has discussed some challenges and opportunities in making commercially successful power electronic products that incorporate digital control.

### 2.3 BASICS OF FPGA

FPGAs belong to a wide family of programmable logic components. An FPGA is defined as a matrix of Configurable Logic Blocks (CLBs), linked to each other by an interconnection network, which is entirely reprogrammable. The memory cells control the logic blocks as well as the
connections so that the component can fulfill the required application specifications. The size of an FPGA is usually characterized by the number of CLBs on it. In general, an FPGA is simply a storage element or memory. Depending on memory technology used during chip manufacturing, it can be One Time Programmable (OTP) or reprogrammed over and over again. The “programmable” term in FPGA indicates an ability to program a function into the chip after silicon fabrication is complete. This customization is made possible by the programming technology, which is a method that can cause a change in the behavior of the pre-fabricated chip after fabrication, in the “field,” where system users create designs.

The configurability nature of FPGA depends on the design technology used during fabrication. Several configurable technologies exist. Among them, only those that are reprogrammable (Flash, EPROM, SRAM) are of interest since they allow the same flexibility as that of a microprocessor. As pointed out by Trimberger (1993), the Static Random Access Memory (SRAM) based FPGA technology is the most widespread one. In SRAM based FPGA device, once a value is loaded, it will not change unless the value itself is altered or power is turned off. It consists of cells made up of multi transistors to drive the output control transistor. Depending on the storage information, the output transistor will be either “ON” or “OFF”.

2.3.1 History of Programmable Logic

In order to know the way FPGAs developed and the reason of their appearing in programmable logics, it is good to see them in context of other related semiconductor technologies. Ian Kuon et al (2007), presented an approximated timeline of semiconductor technology as shown in Figure 2.2.
First Programmable Logic Devices (PLDs) came in the year 1970 in the form of PROMs and were rather simple. Significantly more complex versions became available only in the late 1970s. In order to distinguish them from their less sophisticated ancestors, these new devices are referred to as Complex PLD (CPLD). Subsequently it became common practice to refer the original and less complex versions as simple-PLDs (SPLDs). In 1984 Altera Corporation came up with a CPLD, which is based on a combination of Complementary Metal–Oxide–Semiconductor (CMOS) and Erasable Programmable Read Only Memory (EPROM) technology. With CMOS, Altera is able to achieve the high functional density and complexity with less power consumption. The Application-Specific Integrated Circuit (ASIC) technology has been introduced in 1980’s. It is a programmable integrated circuit meant for a specific and customized application instead for general purpose. There are four main classes of ASICs. They can be classified
according to the complexity as Gate arrays, Structured ASICs, Standard cell devices and Full-custom chips.

In early 1980s, there existed a gap in the digital IC product lines exists. On one side, there were programmable devices like SPLDs and CPLDs, which were highly configurable and had fast design with reduced modification times, but could not support large or complex functions. At the other end of the spectrum were ASICs, which could support extremely large and complex functions, but were very expensive and time consuming to design. In order to address this gap, Xilinx developed a new class of ICs called FPGA, made available in the market in the year 1984. According to Carter et al (1986) the first FPGAs were based on CMOS and used SRAM cells for configuration purposes. The earliest FPGA devices were based on the concept of a programmable logic block, which comprised a 3-input lookup table (LUT), a register that could act as a flip-flop or a latch and a multiplexer.

2.3.2 Generic Architecture of FPGA

The generic architecture of an SRAM-based FPGA is presented by Wolf (2004) as shown in Fig. 2.3. This generic architecture is composed of a matrix of CLBs, which consist of a cluster of logic cells (2–16, depending on the type of FPGA). This matrix of CLBs core is bordered by a ring of configurable input/output blocks (IOBs), whose number can go upto 1,200 user IOBs. Finally, all these resources communicate among themselves through a programmable interconnection network.
With this novel architecture, FPGAs successfully bridged the gap between PLDs and ASICs. On one hand, they were highly configurable and had the fast design and reduced modification times associated with PLDs, and on the other hand they can be used to implement large and complex functions that had previously been the exclusive domain of ASICs.

### 2.4 RECENT TRENDS IN FPGA

More recently, a trend for a coarse-grained architecture has been observed, with the introduction of some dedicated blocks such as block Random Access Memory (RAM) and DSP accelerator units (hardwired multipliers with corresponding accumulators, high-speed clock management circuitry, serial transceivers). Calderon et al (2005) have used the embedded hard processor cores such as PowerPC or Advanced Reduced Instruction Set Computing (RISC) Machine, also called “ARM” processor.
In 2010, Xilinx Inc. introduced the first All Programmable System on a Chip branded Zynq-7000 that fused features of an ARM high-end microcontroller (hard-core implementations of a 32-bit processor, memory, and I/O) to make FPGAs easier for embedded designers to use. By incorporating the ARM processor-based platform into a 28 nm FPGA family, the extensible processing platform enables system architects and embedded software developers to apply a combination of serial and parallel processing to their embedded system designs.

Improvements in FPGA technology are continuing vigorously. Today’s high-density FPGAs are based on a 40-nm silicon process and contain an order of magnitude more logic than the FPGAs available at the initial stage. 32 and 22 nm silicon process technologies have already been demonstrated to be feasible and as FPGAs migrate to these improved technologies, their logic density and performance will continue to increase. Moreover, an interesting feature in the field of control applications is the recent integration of an ADC in the fusion component from Actel. Thus, the original architecture based on a CLB matrix is now enriched by efficient blocks [DSP, memories, processor, Digital Clock Manager (DCM), and ADC], making an FPGA, a true System-on-Chip (SoC) solution.

2.5 ROLE OF HARDWARE DESCRIPTION LANGUAGES IN FPGA

FPGAs are frequently used to implement complex functions. Because of the recent advances in Very Large Scale Integration (VLSI) and also due to the development of appropriate design tools and methods, which were initially reserved for the world of the ASIC. As stated by Ashenden (1995) and Palnitkar (1996), these tools are mostly based on Hardware
Description Languages (HDLs) such as Very High-Speed Integrated Circuits HDL (VHDL) or Verilog. The existence of IEEE standards has spread the use of HDLs and has allowed the creation and development of high-performance computer-aided design (CAD) tools in the field of microelectronics. Thus, the designer can take advantage of HDLs to build own circuits by using hierarchical and modular approach defined at different levels of abstraction using the top–down design methodology.

The hierarchic flow of the top–down design method and its HDL model environment is presented in Figure 2.4. The design flow is partitioned into the following four steps.

- System level, where the specifications of the circuit are given.
- Behavior level, which consists of the algorithmic description of the circuit.
- Register Transfer Level (RTL), where the circuit is described in terms of its components.
- Physical level, where the circuit is physically described by taking the target hardware characteristics into account.

At each level of abstraction, the future integrated circuit is described in HDL, such as behavioral VHDL or synthesized VHDL. This last description gives an exact representation of the operators and variables of the final circuit. In order to simulate and validate the digital circuit’s functionality, various test benches are formulated and executed.
The Integrated Software Environment (ISE) design Suite is the central Electronic Design Automation (EDA) product family of Xilinx. The ISE Design Suite features include design entry and synthesis supporting Verilog or VHDL, place-and-route (PAR), completed verification and debug using Chip-Scope Pro tools, and creation of the bit files that are used to configure the chip. Xilinx ISE is one of the major synthesizing tools used in digital designs with low cost mode. Synthesis is the process of
converting a high-level description of the design into an optimized gate-level representation, given a standard cell library and certain design constraints. A standard cell library can have simple cells, such as basic logic gates like and, or, and nor, or macro cells, such as adders, mux, and special flip flops. A standard cell library is also known as the technology library. A gate-level netlist is a description of the circuit in terms of gates and connections between them. Synthesis tools ensure that the gate-level netlist meets timing, area, and power specifications. The gate-level netlist is input to an Automatic Place and Route tool, which creates a layout. The layout is verified and then fabricated on a chip. Some major applications used in Xilinx are image compression, image transformation, secure communication, testing application etc. Xilinx tool has numerous beneficial and features in synthesis process. Some of them are specifying the different source types, performing the synthesis, simulation, post simulation process, implementation details of device, generating the program file for design implementation etc.

Architecture of any very large scale integration (VLSI) design, the Xilinx plays an imperative role to achieve the hardware efficiency of the design and functionality verification of the design. The majority of designers working with Xilinx FPGAs use the primary design tools ISE Project Navigator and PlanAhead software. In this research work, Xilinx ISE 13.2 version is used for the synthesis and implementation process.

2.6.1 Xilinx ISE Design Suite 13.2 Version

Xilinx ISE 13.2 is the latest software tool released by Xilinx in the year 2011 for synthesizing and implementation of HDL designs, which enables the developer to synthesize the designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer. It provides support for the
28nm 7 series families including the recently arrived Virtex-7 VX485T device. The Spartan families of FPGA are fully supported by this edition.

2.6.2 Cool Stuffs of Xilinx ISE Design Suite

The latest ISE software release has the enhancements to the Plan-Ahead design and analysis tool, providing partial reconfiguration support for Virtex-7 and Kintex-7 FPGAs, and front-to-back, integrated project management environment for improved productivity in designs targeting Spartan-6 FPGAs, Virtex-6 FPGAs, their defense grade counterparts, and all three 7 Series families including initial support for the low-cost Artix-7 family. provides designers the tools they need to facilitate global-team design, rapid feedback on key design considerations, best practices for low-power optimization using the X-Power Estimator tool (XPE), and dynamic power reduction through intelligent clock-gating – all of which is accessible via the Plan-Ahead tool. The Plan-Ahead tool has evolved from a world class I/O pin planner and Floor-Planner to a comprehensive development environment that accelerates time to production with unique integrated front-to-back environment that includes design analysis at each phase of the design cycle – RTL development, IP integration, verification, synthesis, place and route. The end result is rapid convergence on power consumption, resource utilization, and performance with fewer time-consuming design iterations. Updates to X-Power Estimator (XPE) tool enable designers to make power-consumption predictions with a high-level of accuracy.

2.6.3 Special features of Xilinx ISE 13.2 Project Navigator

- Embedded Development Kit (EDK) Integration improvements provide support for multiple Executable and Linkable Format (ELF) files and automatic detection of ELF files referenced by EDK designs.
• Ability to select an Evaluation Development Board (EDB) in the new Project Wizard, new Project dialog box, or design properties dialog box.

• Ability to export hardware design before running implementation

• Compare Projects feature includes additional categories and layout improvements.

• “SmartXplorer” features support for power-dedicated and custom strategies.

• CORE Generator has the ability to update a core to the latest version as well as the ability to check all core versions.

• Ability to create a new System Generator source in Project Navigator.


2.6.4 System Design Using Xilinx ISE Design Suite

Initially, the design is implemented using HDL language. By using the check syntax option in Xilinx tool, any syntax error of HDL programming can be identified. Then synthesis and implementation processes for the design are taken place using the appropriate options in Xilinx process tab. Any errors in synthesis or implementation process may be corrected and the design can be reconstructed. The bit-file is generated using the programming file generation option and finally the FPGA is configured with the design. The configuration is done by downloading the bit-file to the FPGA processor by
using the software, “Digilent Adept”. The steps involved in the design process are illustrated in Figure 2.5

![Diagram of design process steps]

**Figure 2.5 Design process steps**

2.7 **FPGAs IN POWER ELECTRONICS AND DRIVE APPLICATIONS**

The ceaseless interest during the last 30 years for the digital control of power electronics and drive applications is mainly due to the great advantages that this technology offers compared to the analog one. The flexibility, reduced computational time, and possibility to implement complex design in real time, are some of the key features of the digital control of power electronic drives. The microprocessor based PWM control has played a major role in this digital revolution. But the introduction of DSP offered an alternate solution. However, even though the scale of integration rate and execution time of the DSPs have gone up, the bandwidth available is less than
that with a simple analog controller. The main reasons are the necessity to add additional components like ADCs, DACs and zero-order hold circuits to interface with the processor with its analog environment. The computational time also gets increased. In addition to these, the quantization error that is inherent to the digital implementation has to be considered. A remedial method is to adopt a digital system of quasi-control nature, which combines the advantages of both analog and digital controllers. This leads to a third category of controllers, namely, analog controllers by digital means.

An FPGA is a good candidate for this new category of controllers. Any design implemented in the FPGA can be easily modified as per the requirement. The main difference from DSP-based solution, is that FPGAs allow concurrent operation (simultaneous execution of all control procedures), which result in high computational speed and makes the implementation of complex control methods feasible. The FPGA based implementation of PWM methods offer high-level integration density, fast computation, better reliability and good control performances with reconfigurable nature.

2.7.1 State-of-the-art FPGA Controllers Review

Fast progress of very large scale integration (VLSI) technology and Electronic Design Automation (EDA) techniques in recent years has created an opportunity for the development of complex and compact high-performance controllers for industrial electronic systems as mentioned by Cirstea et al (2002). The modern EDA tools to create, simulate, and verify a design without resorting to hardware, can quickly evaluate complex systems and ideas with very high confidence operation of the final product. Speed performance of new components and flexibility of all programmable solutions offered many opportunities in the field of digital implementation for industrial control systems. Specific hardware technologies such as FPGAs can be
considered as an appropriate solution in order to boost the performance of controllers. Indeed, these generic components combine low-cost development, use of convenient software tools and higher significant integration density as mentioned by Brown (1996).

FPGA technology is now adopted by an increasing number of designers in various fields of application such as wired and wireless telecommunications, image and signal processing. Recently, other application fields are in growing demand, such as medical equipment, robotics, automotive and space and aircraft embedded control systems. According to Lu, (2004); Velusamy (2005); and Johnson (2004), for these embedded applications, reduction of the power consumption, thermal management and packaging, reliability, and protection against solar radiation are of prime importance. Finally, industrial electrical control systems are also of great interest because of the ever-increasing level of expected performance while at the same time reducing the cost of the control systems as pointed out by Kiel (2002). Nasri Sulaiman et al (2009) reviewed the state of the art of FPGA with the focus on FPGA-based systems. The advantage of using FPGA with intelligent systems had been highlighted with an example of FPGA based neural network system and a survey for FPGA-based control systems design with different applications also presented.

2.7.2 Unique features of FPGA Based Digital Control

With the introduction of FPGAs into Digital control platforms, it is possible to lower the development cost, to meet the urging security issues, to have true parallel processing and reconfigurability, which are very attractive from the industrial product point of view. Some of these important issues are summarized below.
- Cost reduction: The cost can be reduced by using a specific FPGA architecture depending on the requirement of the application. In turn, it reduces the development time. Also, by integrating as many components as possible in a single chip, including the possibility to integrate analog parts (system-on-a-Chip (SoC)).

- Product Security: This is one of the very important issues for the industry and is directly related to the revenue generated by the product. It is of great anxiety to protect the product Intellectual Property (IP). The FPGAs provide different levels of protection to meet the necessities of IP.

- Reconfiguration: An FPGA-based controller can be adapted in runtime to the needs of the plant by dynamic reconfiguration.

- Parallel processing: The control execution time of an FPGA can be considerably reduced by designing a dedicated parallel architecture, which makes it possible to reach the level of performance of an analog circuit without downsides.

### 2.8 BENEFITS OF FPGA BASED POWER CONVERTER CONTROL

The most significant criteria to be considered for the design of a power converter control system is the choice of hardware technology to match the requirements of the algorithm to implement. The two main hardware solutions in implementing a controller are DSPs and FPGAs. According to the nature of algorithm to implement, the designer can choose the best option by considering the complexity and timing constraints of the algorithm. Since the FPGA has a high level of parallelism of the algorithm and minimum execution time, it is a better choice.
To outline the benefits of using FPGAs in power converter control, Eric Manmasson and Cristea (2007) have made a comparative study on the ability of the FPGA based controllers to execute quasi-instantaneously their tasks, as shown in Figure 2.6. The study is based on a typical example related to the current control of PWM inverters. In this figure, $T_{\text{ADC}}$ represents the Analog-to-Digital signal time conversion, $T_c$ is the execution time of the control algorithm, and $T_s$ is the sampling period that is usually taken equal to the switching period of the power converter or its half period.

Figure 2.6(a) corresponds to the case using a general purpose microcontroller. In this case, the main limiting factor is the computing power of this component. Sampling period is fixed according to this limit, leading to one and a half switching period of delay. This reduces the bandwidth of the closed-loop system and, in some cases, may destabilize the controlled process.

**Figure 2.6** Timing distributions (a) General-purpose microcontroller (b) DSP controller and (c) FPGA-based controller
Figure 2.6(b) corresponds to the case of a DSP controller implementation. This case gives much better results than the former one. By carefully designing the system, the delay can be reduced to a half switching period, which greatly improves the dynamic performance. Direct control of the power converters can be achieved, but expected results are of less quality than those obtained via an analog controller.

Figure 2.6(c) corresponds to the case with FPGA-based controller. Due to the potential parallelisms of the control algorithm, FPGAs can take only a fraction of the switching period to execute in real time a full complex algorithm. However, by using optimization techniques and pipelining, the designer can easily build a balanced architecture, which respects the area limitation and preserves the rapidity of execution of the control algorithm. Therefore, the actual computing time $T_c$ with FPGA-based controllers is far below the one reached with a programmed solution.

Such instant reactions make the FPGA-based controllers very close in their behaviors to their analog counterparts. They preserve their advantages (minimum calculation delay, higher bandwidth) without their drawbacks (parameters drifting, poor level of integration). Hence, this quasi-analog property could be sufficient to promote this technology for implementing efficient power converter control systems.

2.9 FPGA BASED PWM GENERATION: A REVIEW

Several attempts have been made in developing digital architecture for generating PWM signals. Dancy (2000) and Rahim (2004) presented the different concept and functional component usage viz. high frequency counter based, counter based and cascaded counter based. Fabrizio et al (2006) have described a methodology that allows an easy implementation of IP-Cores focusing only on their functionalities rather than their interfaces and their
integration. It has been implemented in classical Xilinx design flow using EDK and ISE. Jacques et al (2008) have implemented Large-Integer hardware multiplier using the vendor synthesis/place and route software tool. The design solutions are multiplier circuits based on embedded arithmetic blocks built in the Xilinx Virtex-4(V4) family of FPGA, which reduce the delay. Agnihotri (2010) has implemented a combined Pulse Width Modulation (PWM)-Pulse Frequency Modulation (PFM) technique in a Field Programmable Gate Array (FPGA) to control DC-DC converters. The efficiency of the converter based on the proposed technique is compared to the efficiency of processor-based implementation. The results show that the efficiency of the converter with the FPGA-based technique is higher than the efficiency of the converter with processor based technique. Christian et al (2011) has described the procedure to designing the digital application using the powerful Xilinx Design Language (XDL), with plenty of practical examples and used cases.

2.10 PROPOSED FPGA BASED PWM IMPLEMENTATIONS

A generalized block diagram for the implementation of different PWM strategies is shown in the Figure 2.7. The digital architecture is designed for each PWM technique using Very High-speed Hardware Description Language (VHDL) as in appendix A2.1 and the functional simulation are carried out by using the software tool Modelsim 6.3 as in appendix A1.1. The functionally simulated architecture is synthesized using the tool, Xilinx 13.2i. Here, the high-level language description of the design is converted to a Register Transfer Level (RTL) implementation. The Timing, Area Utilization and power analysis are done with this tool. Finally, the bit-file for the designed architecture is generated.
Figure 2.7 General block diagram for the proposed PWM implementations

The bit-file is downloaded to the FPGA device (Spartan6 - XC6SLX45) as in appendix A3.1, using the programming tool. Then the device is configured with the designed architecture. The FPGA based PWM architecture is interfaced with the prototype of VSI as in appendix A3.2, with the appropriate load. Various outputs across the load are observed using the Digital Storage Oscilloscope (DSO) for different operating specifications and its performances are compared with the conventional SPWM method.
2.11 SUMMARY

Digital control systems revolutionized the industrial world by virtually eliminating the drawbacks of typical analog systems which allows the diffusion of sophisticated control techniques. In the field of power converters, they were fundamental for the evolution from DC drives to high performance AC drives with modern and energy efficient techniques. Nowadays, almost all power converters include high power applications contain microcontrollers, DSP or FPGA or the combination of both. In power systems based on custom chips, a digital controller is usually embedded in the form of hybrid ASIC. In near future, the FPGA or ASIC based drive control will be employed wherever the power or energy has to be controlled.