EFFICIENT DYNAMIC SYSTEM IMPLEMENTATION OF FPGA BASED PID CONTROL ALGORITHM FOR TEMPERATURE CONTROL SYSTEM

1M. A. Majed, 2Prof. C.S. Khandelwal,
1,2ETC, J.N.E.C. Aurangabad (India)

1amajed@rediffmail.com, 2chhaya.khandelwal@rediffmail.com,

ABSTRACT

In this paper, an efficient design scheme for implementation of the Proportional-Integral-Derivative (PID) controller using Field Programmable Gate Array (FPGA) technology is presented. It is efficient scheme in the sense dynamic power system, effective cost over other digital implementation techniques and etc. The VHDL (VHSIC Hardware Description Language; VHSIC: very-high-speed-integrated circuit) will be used for code development, simulation and synthesis. The hardware configuration with multichannel ADCS7476MSPS 12 bit ADC and 9 to 12 volt DC fan along with FPGA will be implemented. To the work, the structure of the built system is going to be designed to only include one hardware PID controller, and by switching the analog input and output, the control board could realize PID controller to fulfill a control demand. The PID controller implemented using a scheme where a Look-Up-Table (LUT) mechanism inside the FPGA is utilized. A design which is efficient in terms of power consumption and chip area, while FPGA chip can resulting in a cost reduction of the controller hardware. Spartan3 family 3s500efg320-5 FPGA development board with suitable ADC and DC fan will be used for realizing the hardware. The Xilinx Chip-Scope tool will be used to test the FPGA inside results while the logic running on FPGA. Xilinx power tool will be used for power analysis of the implemented FPGA based core.

Keywords: FPGA design, Dynamic System, PmodAD1, PID controller, VHDL, Chip-Scope analyzer.

I INTRODUCTION

The proportional-integral-derivative (PID) controller is one of the most common type of feedback controllers that are used in dynamic systems [3]. An important feature of this controller is that it does not need a precise analytical model of the system that is being controlled. This controller has being widely used in many different areas, such as manufacturing, robotics, automation, process control, aerospace, and transportation system. Implementation of PID controllers has gone through several stages of evolution from early mechanical and pneumatic design to microprocessor-based system. Recently, FPGAs have become an alternative solution for the realization of digital control systems, which were previously dominated by general purpose microprocessor systems [1], [2], [5]. FPGA technology is now considered by an increasing number of designers in various fields of applications such as wired and wireless telecommunications, image and signal processing, medical equipments, robotics, automotive, and space and aircraft embedded control systems [4]. For these embedded applications, reduction of the power consumption, thermal management and packing, reliability and protection against solar radiation are of prime importance [3]. FPGA-based controllers offer advantages such as high speed, complex functionality, low power consumption and reduction in cost. These are attractive features from the embedded system point of view [1]. Another advantage of FPGA-
based platforms is their capability to execute concurrent operations, allowing parallel architectural design of digital controllers [2],[4]. Conventional implementations of FPGA-based controllers have not focused on optimal use of hardware resources. These designs usually require a large number of multipliers and adders and don’t efficiently utilize the memory rich characteristics of FPGA’s. An FPGA chip consists of many memory blocks referred to as LUT’s and can be utilized to improve the performance of certain operations such as multiplication, while the tradeoff for speed can be tolerated. In this paper, we study the design and implementation of an efficient PID controller scheme, which is an efficient LUT design method and is very promising in the FPGA implementation of PID controller. The proposed PID controller reduces the cost of the FPGA design by enabling the chip to accommodate more logic an arithmetic functions while requiring less power consumption. In addition due to the flexibility of using LUT’s in FPGA’s, the design method can be used to implement other algorithms, such as antiwindup [4] compensation or adaptive control schemes.

In this paper, a case study is presented in which a modular FPGA-based design approach is applied to design a temperature control system. The same approach can be extended to design other embedded controllers using FPGA [5]. The complete system is implemented by dividing system functions in reconfigurable module. The organization of the paper is as follows: In section II a PID controller is considered and its implementation using its scheme is discussed. In section III an overview of the components of a general purpose PID based feedback control system is presented followed by an approach for designing the control system using FPGA technology. In section IV the implementation results on a Xilinx FPGA chip[6] and in modelSim6.2c are discussed. Comparisons are made between the proposed scheme and the design based on conventional methods. Conclusions are discussed in section V.

II FPGA HARDWARE IMPLEMENTATION PID CONTROLLER

The application of a PID controller in a feedback control system is shown in Fig. 1, where \( u_c \) or SP is the analog signal or command signal form the user, \( y \) is the feedback signal[2], \( e \) is the error signal and \( u \) is the control output in the range of 0V to +3.3V(controller output) to match the ADC.

![Fig. 1 Block Diagram of a general PID based Feedback control system](image)

The PID control action algorithm in analog controller is given by equation (1) [4],

\[
u(t) = K_p(e(t) + \frac{1}{T_i} \int_0^t e(\tau)d \tau + T_d \frac{d}{dt} e(t)) - (1)
\]
where \( K_p \) is the proportional gain, \( T_i \) is the integral time, \( T_d \) is the derivative time, \( e(t) \) is the error signal to the controller input, \( u(t) \) is the output of the controller. From a practical point of view, implementation of the above algorithm has certain limitations [2]. Firstly, actuator saturation can cause integrator windup, leading to a sluggish transient response. Secondly, the pure differentiation term amplifies noise, leading to a deterioration of the control command. Finally, the differentiation term acts on the error signal, taking the derivative of the command signal as well. This may lead to spikes in the command signal when the user changes the set point abruptly. Further it was found advantageous, not to let the derivative act on the Command signal and let only a fraction of the command signal act on the proportional part. Eq. (1) can also be represented as the form;

\[
u(t) = K_p e + K_i \int_0^t e(\tau)d \tau + K_d \frac{d}{dt} e(t) \quad - (2)
\]
where \( K_i = K_p/T_i \) and \( K_d = K_p T_d \), and are also the control Parameters to be designed.

To realize the system by FPGA, the backward difference equivalence,

\[
u(t) = K_p e + K_i \int_0^t e(\tau)d \tau + K_d \frac{d}{dt} e(t)
\]

is chosen to convert the continuous-time system into discrete-time system,

\[
u(z) = K_p + K_i \frac{T_o}{1-z^{-1}} + K_d \frac{1-z^{-1}}{T_o} \quad - (3)
\]
where \( T_s \) is the sampling time.

In the FPGA implementation of the PID controller, major effort is placed on the hardware optimization of the controller. In this regard, an area-efficient algorithm for the PID controller is proposed in this section. An area efficient controller means that it can fit in a smaller FPGA chip, resulting in cost reduction of the controller hardware[2]. In order to implement the control algorithm using FPGA, it has to be discretized as in Eq. (3).

To build a generalized PID controller [5], the controller’s parameters are going to be arbitrarily assigned according to the system’s characters. Therefore, the controller’s parameters are set in the range as Table I shows, where the corresponding sampling frequency is in the range [1Hz to 100 kHz]. To attain the performance shown in Table I, the controller’s parameters with the desired bit number shown in Table II. As regarding the other parameters, to make sure the controller’s performance has enough precision, we plan 24 bits for \( T_s \) (clk_4hz), and 12 bits for \( K_p \), \( K_i \) and \( K_d \). As Fig.1 shows, sensor analog inputs from 12-bit ADC are set to convert the analog signals of command input and process output to digital type, the ADCs are 12 bits width. Because the PID controller output is directly sent PWM, the controller output \( u \) is also limited to be 12 bits.

### Table I. The parameters Range of the Designed controller

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_s )</td>
<td>[0.00001, 1]</td>
<td>sec</td>
</tr>
<tr>
<td>( K_p )</td>
<td>[0, 2047.9375]</td>
<td></td>
</tr>
<tr>
<td>( K_i )</td>
<td>[0, 127.99609375]</td>
<td></td>
</tr>
<tr>
<td>( K_d )</td>
<td>[0, 127.99609375]</td>
<td></td>
</tr>
</tbody>
</table>

### Table II. The Bit no. of Control Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Bit no. of integer</th>
<th>Bit no. of fraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_s )</td>
<td>24</td>
<td>0</td>
</tr>
<tr>
<td>( K_p )</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>( K_i )</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>( K_d )</td>
<td>4</td>
<td>7</td>
</tr>
</tbody>
</table>

### III The Implementation of PID Controller and Application to Control System

The z-domain block diagram for digital PID controller to be implemented is shown in Fig.2 [5], where \( T_z \) is sampling time, it stands for the ADC on board. The architecture used to realize the PID controller shown in Fig. 2 is demonstrated in Fig.3 [5]. The block diagram of a general purpose PID based feedback system is shown in Fig. 1, where \( u_c \) is the command signal, \( y \) is the feedback signal, \( e \) is the error signal, and \( u \) is the control input. Fig. 4 shows the block diagram of FPGA-based temperature control system. The system consists of following components:

1. a tube with a fan, a light bulb, and a thermistor;
2. an I/O panel and push-button keys;
3. an ADC chip PMOD AD1;
4. FPGA development board consisting a Xilinx Spartan-3E FPGA.
The thermistor is used for temperature measurement inside the tube, output of thermistor i.e. voltage across thermistor, is used to calculate the temperature that is sampled by the ADC to be used in the control law. PWM generator is implemented to control the dc motor and the lamp[2]. The motor as shown in Fig. 4 and the lamp are turned either ON or OFF depending on the PID controller output u. If u is positive, the opposite will be true. In order to generate the correct PWM output. Both PWM generators run at a clock frequency of 50 MHz with a PWM period of 20 ms. The ADC that is used in this system is a 12-bit ADC with an internal reference voltage of 3.3V. The ADC Interface polls the ADC every 20 ms to receive.

**IV FPGA EXPERIMENTAL RESULTS IMPLEMENTATION IN XILINX AND MODELSIM**

PID controller is implemented using the Xilinx Inc. FPGA technology and can be used as a general purpose controller for different applications. The FPGA design flow is as follows. First, the controller was implemented by using the Xilinx ISE [6] foundation tools and simulated at the Register Transfer Level (RTL) using ModelSim 6.2c tool to verify the correctness of the design in Fig. 5. By using the Xilinx ISE foundation tools, the logic synthesis was carried out to optimize the design, and the placement and routing were carried out automatically to generate the FPGA implementation file. Finally, the generated implementation file was downloaded to the FPGA development board for testing and its Device utilization summary is shown in Table III. This PID controller is targeted to a Xilinx Spartan 3E FPGA for a 13 bit input.
It is seen from Table III that the design uses about 33% of the logic resources required by the design. Since in the control system considered, the 15 cycles of 50 MHz clock are fast enough, the tradeoff of speed to hardware resource and power saving is useful for the PID controller design. A design which is efficient in terms of power consumption and chip area, means that the FPGA chip can be used to accommodate more controllers with adequate speed and low power consumption, resulting in a cost reduction of the controller hardware. The Different wave forms for PWM to Fan using ModelSim test bench result is shown in Fig. 6, and the Chip Scope Pro analyzer waveform of the thermistor and DC Fan is shown in Fig. 7.
In this paper, a novel PID based controller was presented, for FPGA implementation. By using the memory inside FPGA has been utilized to provide efficient design for PID controllers, resulting in reduction of FPGA design cost. In addition, due to the flexibility in the FPGA, this FPGA-based PID controller can be easily extended to incorporate other algorithms, such as antiwindup protection or adaptive scheme. The FPGA implementation results shows that, the DA design requires only 33% of logic devices. Furthermore, the power consumption is reduced which is calibrated by Xilinx-X power tool as shown in fig. 8. Hence the efficient dynamic system implementation is done which uses only 5.17 mw of the power. The experimental setup is as shown in fig.9.

V CONCLUSION

Fig. 6 Output waveform of PID controller

Fig. 7 Chip Scope Analyzer output wave form

311
Fig. 8 Xilinx X Power result of PID algorithm based Temperature Control system

Fig. 8 Experimental set up of PID Temperature Control system

REFERENCES

6. www.xilinx.com