ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

Low Power VLSI Architectures for Digital PID Controller Applications

Abrar Ul Haq[1], Dr. Sreerama Reddy G. M[2]

VTU Research Scholar, Assistant Professor, Dept. of Electrical & Electronics Engineering, Presidency University, Bangalore, India

Professor & HOD, Dept. of ECE, CBIT, Kolar, India

ABSTRACT: Embedded systems are playing an increasingly important role in control engineering. Despite their popularity, embedded systems are generally subject to resource constraints and it is therefore difficult to build complex control systems on embedded platforms. Traditionally, the design and implementation of control systems are often separated, which causes the development of embedded control systems to be highly time-consuming and cost. In this paper we present the novel digital proportional–integral–derivative (PID) controller architecture, based on the adder and multipliers which consumes lower power. Nowadays embedded control applications requires low power and fast acting PID controllers with a closed loop performance using less resources, resulting in cost reduction. This paper briefs about the significance of datapath components in achieving the low power constraint. Here the datapath architectural optimizations are done at the gate level implementation for the processors used in PID applications. Arithmetic units of the PID controller were designed with low power datapath components to reduce the power consumption while still improving the design performance. Standard Top Down FPGA design methodology was utilized for gate level implementations with XILINX Virtex-7 Family device. The proposed optimized datapath architectures has resulted in 16% less dynamic power with same performance and area. From this paper we have demonstrated that the proposed concept can be applied to designs to achieve the power optimizations.

KEYWORDS: PID, FPGA, Low power VLSI, Adder, Multiplier, Data path.

I.INTRODUCTION TO PID CONTROLLER

The PID is by far the most commonly used feedback controller due to its simple structure and robust performance. An important feature of this controller is that it does not require a precise analytical model of the system that is being controlled, which makes it very attractive for a large class of dynamic systems.

In embedded control applications, such as in small-scale mobile robot, the control-loop-cycle is very tight and the power budget is very limited. A low sample rate leads to poor and degraded control-performance. And high power consumption shortens the battery lifetime. To cope with these two severe and antagonistic constraints, the need for both a high-speed and low-power PID structure is of utmost importance. A proportional–integral–derivative controller (PID controller) is a generic control loop feedback mechanism (controller) widely used in industrial control systems – a PID is the most commonly used feedback controller. A PID controller calculates an "error" value as the difference between a measured process variable and a desired set point. The controller attempts to minimize the error by adjusting the process control inputs.PID controller for a generic system is as shown in figure 1.

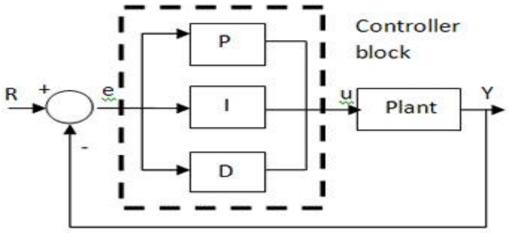


Figure 1: PID control for general system

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

A PID is widely used in feedback control of industrial processes on the market in 1939 and has remained the most widely used controller in process control until today. Thus, the PID controller can be understood as a controller that takes the present, the past, and the future of the error into consideration. After digital implementation was introduced, a certain change of the structure of the control system was proposed and has been adopted in many applications. But that change does not influence the essential part of the analysis and design of PID controllers. A proportional– integral–derivative controller (PID controller) is a method of the control loop feedback. This method is composing of three controllers [1]:

1. Proportional controller (PC)
2. Integral controller (IC)
3. Derivative controller (DC)

A. Role of a Proportional Controller (PC)

The role of a proportional depends on the present error, I on the accumulation of past error and D on prediction of future error. The weighted sum of these three actions is used to adjust Proportional control is a simple and widely used method of control for many kinds of systems. In a proportional controller, steady state error tends to depend inversely upon the proportional gain (i.e.: if the gain is made larger the error goes down). The proportional Response can be adjusted by multiplying the error by a constant Kp, called the proportional gain. The proportional term is given by:

P = KP. error (t) ……………… (a))

A high proportional gain results in a large change in the output for a given change in the error. If the proportional gain is very high, the system can become unstable. In contrast, a small gain results in a small output response to a large input error. If the proportional gain is very low, the control action may be too small when responding to system disturbances. Consequently, a proportional controller (Kp) will have the effect of reducing the rise time and will reduce, but never eliminate, the steady-state error.

B. Role of an Integral Controller (IC)

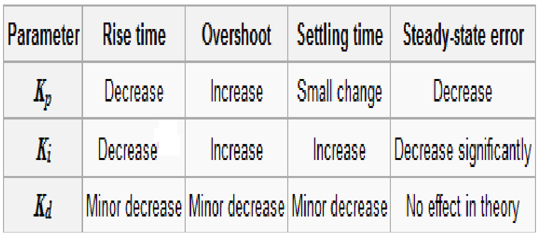
An Integral controller (IC) is proportional to both the magnitude of the error and the duration of the error. The integral in a PID controller is the sum of the instantaneous error over time and gives the accumulated offset that should have been corrected previously. Consequently, an integral control (Ki) will have the effect of eliminating the steady-state error, but it may make the transient response worse. The integral term is given by: I=$\int_0^t error\ (t).dt$

C. Role of a Derivative Controller (DC)

The derivative of the process error is calculated by determining the slope of the error over time and multiplying this rate of change by the derivative gain Kd. The derivative term slows the rate of change of the controller output. A derivative control (Kd) will have the effect of increasing the stability of the system, reducing the overshoot, and improving the transient response. The derivative term is given by:

$$D = \frac{derror(t)}{dt}$$

Effects of each of controllers Kp, Kd, and Ki on a closed-loop system are summarized in the table shown below in table 1.



Parameter	Rise time	Overshoot	Settling time	Steady-state error
K_p	Decrease	Increase	Small change	Decrease
K_i	Decrease	Increase	Increase	Decrease significantly
K_d	Minor decrease	Minor decrease	Minor decrease	No effect in theory

Table 1: In PID controller

The PID controller calculation (algorithm) involves three separate constant parameters, and is accordingly sometimes called three-term control: the proportional, the integral and derivative values, denoted P, I, and D. Heuristically, these values can be interpreted in terms of time: P depends on the present error, I on the accumulation of past errors, and D is a prediction of future errors, based on current rate of change. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve or the power supply of a heating element.

In the absence of knowledge of the underlying process, a PID controller is the best controller. By tuning the three parameters in the PID controller algorithm, the controller can provide control action designed for specific process requirements. The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the set point and the degree of system oscillation. Note that the use of the PID algorithm for control does not guarantee optimal control of the system or system stability.

In the digital version of PID controller the proportional value is same as the Analog PID controller, the integral value becomes the sum and the derivative becomes the difference. The continuous time signal e(t) is sampled in fixed intervals .An A/D converter interfaces the input of the continuous time signal and D/A converter interfaces the output of the sampled fixed time intervals.

Implementing PID control algorithms on optimized and dedicated hardware provides lower cost, higher performance, and power benefits over a standard off the shelf, or even optimized microprocessor/DSP.

The remaining section of the paper is organized as follows. Section 2 describes continuous time PID controller ,section3 deals about the discrete PID controller ,proposed work is described in section 4,section 5 describes the results and observations .section 6 deals about the conclusion. At the last references are listed.

II. CONTINUOUS TIME PID CONTROL

Analog controllers provide continuous processing of the signal, and they can be used for large-bandwidth systems. They also provide almost infinite resolution, thus ensuring precise control. Analog controllers have been around for a long time, and there is a great deal of literature, practical experience, and design methods available. Analog controllers are implemented using one or more operational amplifiers and a number of components (e.g., resistors or capacitors).

A standard equation of PID controller is given below.

$$U(t)= \ k\,[e(t)+\frac{1}{Ti}\int_0^t e\,(\tau)d\tau+T_d\,\frac{de(t)}{dt}]\ldots\ldots\ldots\ldots\ldots(1)$$

Where the error e(t), the difference between command and plant output, is the controller input, and the control variable u (t) is the controller output. The 3 parameters are K (the proportional gain), Ti (integral time), and Td (derivative time). Figure 2 represents the block diagram for the implementation of continuous time PID controller.

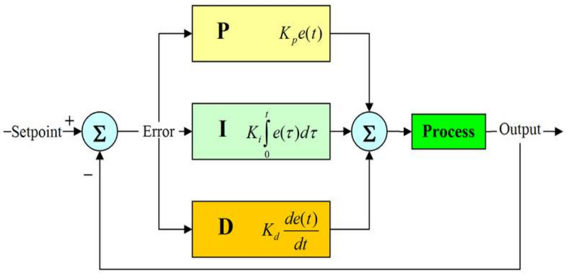


Fig1: Continuous time PID controller block diagram.

Performing Laplace transform on equation 1, we get

$$G(s) = K(1+\frac{1}{sTi}+sT_d)\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

The PID controller can be realized in the other format called as parallel format of PID controller.

$$U\ (t)= K_p e\ (t)+K_i \int_0^t e\,(\tau)d\tau+K_d\,\frac{de(t)}{dt}\ldots.(3)$$

And its Laplace transform is given as follows

$$G(s) = K_p+\frac{Ki}{s}+sK_d.......$$

The parameters can be converted from one to another form as given below.

$$k_{p=}\mathrm{K},\ K_I=\frac{K}{Ti},\ k_{d=}\mathrm{k}\,T_d$$

A. Implementation Issues in Analog Controllers

Analog Controllers Are Implemented Using One Or More Operational Amplifiers And A Number Of Components (E.G., Resistors Or Capacitors). If They Are Integrated On A Single Chip (Without Using External Components), Their Performance Is Limited By The CMOS Component Specifications, I.E., Resistors Of Some Hundreds Of Kilo-Ohms And Capacitors Of Some Tens Of Pico-Farads. Furthermore, Analog Controllers Suffer From Component Aging, Component Process Variations, And Temperature Drifts.

Analog controllers are also limited to simple control algorithms from classical control theory like P controllers, PID controllers, or lead-lag compensators. The number of components (such as capacitors, resistors, or operational amplifiers) increases with the order of the Analog controller. In the niche applications which require high-bandwidth and low noise performance, such as laser diode controllers, the analog PID controllers are used. The analog solutions require analog to digital converter, Which will increase the system cost & error. The analog architectures lack design Scalability & Portbility challenges. The analog architectures have to be designed from scratch every time, since they cannot be reused. The analog architectures are not suitable for higher order control systems. Consequently, both its reliability and accuracy decrease. Hence in order to overcome these issues discrete time PID controllers' are implemented.

III. DISCRETE-TIME PID CONTROLLER

The current day electronic controllers are implemented in digital domain based on either microcontrollers or FPGAs. In industry, the most modern PID controllers are implemented in programmable logic controllers (PLCs) or as a panel-mounted digital controller.

Digital controllers sample the signal at discrete time intervals, which limits the bandwidth that can be handled by the controllers. Digital controllers can be programmed, which renders them flexible and easy to upgrade. The implementation of a low order digital controller is as reliable as the implementation of a high-order digital controller. The discrete form of PID controller is as shown

$$U\ (z) =[\ K_p+\frac{K}{1-z1}+K_d(1-z1)]E(z).......(4)$$

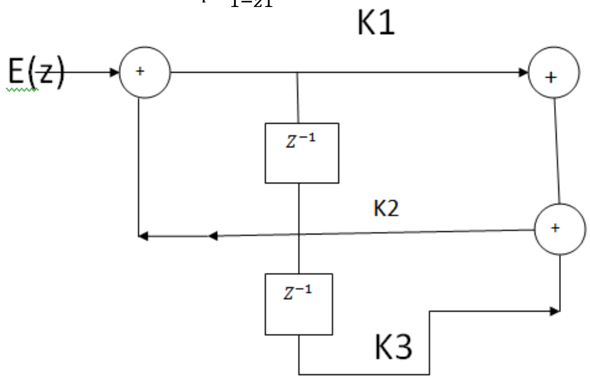


Fig.2 Implementation of Discrete time PID control

Figure 2 indicates the implementation of the discrete PID controller. It is a closed loop system which consists of the both combinational & sequential components. The combinational components include Adders & Multipliers (K1. K2 &

K3). The sequential component includes Delay Elements (Flip Flops). The multipliers are used to adjust the gain of the control loop.

The various state of the art low power minimizations are Clock Gating, Functional Gating, Multi Voltage, Power Gating, Multi Vt, Back Biasing, Different Dielectric Materials etc. The primary limitations of existing systems are:

- The existing architectures are optimized for performance corner. But the PID applications require low frequency of operations.
- The Voltage Scaling & multi voltage designs are used to reduce power consumption, which will increase the delay & complexity of the verification time.

The stringent requirement of the power cannot be met by the current day solutions.

IV. PROPOSED WORK

Generally for digital signal multiplication and addition on FPGA digital signal processor used which is based on conventional multiplier and adders which occupies more area on FPGA and produce undesired delay which affect PID time response, even consuming more power. Hence here by choosing particular adder and the multiplier the power requirement can be reduced when compared with the exiting architecture of discrete PID controller, i.e here choosing of adder and multiplier will decides the amount of power consumption of the design.

In the state of the art deep sub-micron technology nodes, interconnect has a significant impact on design constraints such as area, performance & power. The interconnect delay & power will be larger than the logic delay & power consumed. The primary focus is reducing the interconnect impact in the proposed architectures. We can achieve this by proposing complex gates which will occupy minimal chip area & hence results in power reduction.

The leakage power consumed is directly proportional to the transistor stack length. Since the transistor ON resistance will increase & leakage current will reduce. In the proposed complex cells, the transistor stack length is increased due to which the leakage power is reduced.

Thus the proposed datapath architectures have lesser number of gate counts in the critical path and requires lesser number of complex cells to implement required functionality with parallel output paths. This results in reduced area and minimum interconnects between the gates. Hence the low power consumption and this improve the efficiency of the system with no or minimum trade-offs.

The different datapath components such as half adder, full adder, ripple carry adder & multiplier are implemented based upon concept of approximate arithmetic. In the proposed approximate based Full adder architecture, the half adder the carry path is approximated to simpler gates.

Full Adder: In the proposed approximate based Full Adder architecture, the carry path of the full adder is approximated to one AND gate instead two AND gates. This will result in both area & power reduction.

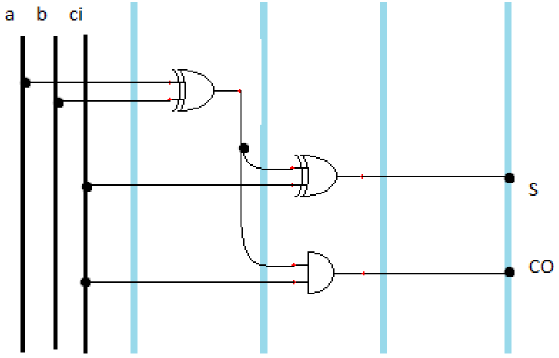


Fig 3: Full adder

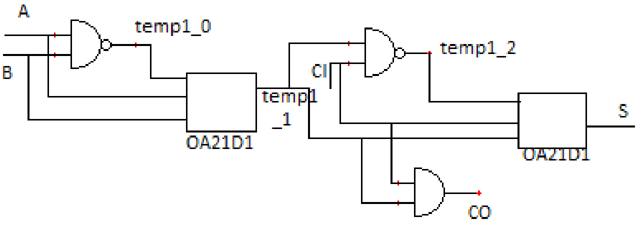


Fig 4: Full adder with OA21D1 gate

Ripple Carry Adder: To optimize area & power, based upon the approximate full adders both 8-bit & 9-bit ripple carry adders are implemented.

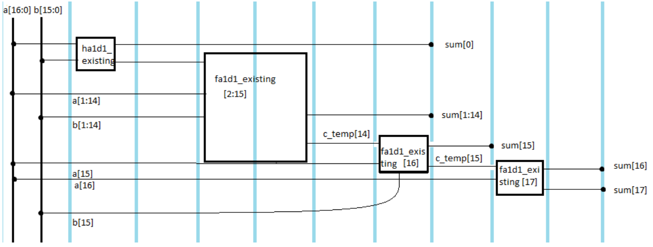


Fig 5: Ripple carry adder

Compressor: In the proposed approximate based Compressor architecture, the carry path of the full adder is approximated to AND gate instead of complex AO22 gate, which will result in both area & power reduction.

Multiplier: Booth encoded Wallace tree architecture based multiplier is implemented. The proposed approximate ripple carry adders & compressor based multiplier is implemented. The proposed approximate arithmetic concept is applied to the datapath logic architecture shown in figure 6. It consists of adder & multiplier as primary datapath components.

Top-Down Design Methodology is used in this research work. The Verilog Hardware Description Language is used to architecture, functionally verified using Modelsim & implemented using XILINX ISE. Power analysis is performed on the Post placed & routed netlist to get accurate results close to silicon values.

V. RESULTS AND OBSERVATIONS

The digital PID controller has been designed with low power architectures to achieve minimal power consumption. Critical datapath components were identified in the digital PID controller and low power datapath optimizations were applied to datapath blocks rather than the entire system. The proposed architectures were implemented using VLSI Frontend based on Top-Down design methodology. The Design was modelled at Register Transfer level using Verilog, Functionally verified using Modelsim & mapped to Xilinx Virtex 7 device using Synthesis tool. The Xilinx XPOWER tool was used to perform power analysis.

The digital PID controller with conventional and proposed architectures was modelled using Verilog HDL and their functionality was verified using model-sim simulator's waveform editor. Designs were synthesized with standard FPGA methodology using Xilinx ISE compiler by targeting to Virtex7 device. Proposed architecture was applied to all possible parts of the digital PID architecture and the importance of datapath architectural optimizations are observed. The results of the conventional and proposed architectures are tabulated in Table I.

Synthesis of both existing & proposed designs are implemented using the Xilinx tool. The designs are mapped to Virtex 7 device by performing post place, map and route. The gate level simulation has been carried out. This particular design is optimized for power.

DOI:10.15662/IJAREEIE.2015.0503130

Parameter	Existing	Proposed	%change
Area (BELs)	1422	1422	0
Delay (nano seconds)	16.607	16.607	0
Dp (milli watt)	32.36	26.99	16.59

Table I: Benchmarked Digital PID Controller results

Note: Dp is Dynamic Power

Table II gives the typical constraints used during synthesis. It is for demonstration purpose only. The proposed concepts can be proven for any FPGA Family of devices, any speed grade & any process technology.

Family	Virtex7
Part	xc7v285t
Package	ffg1157
Grade	Commercial
Process	Typical
Speed Grade	-3

Table II: FPGA Synthesis constraints

Table I gives the results of the digital PID controller with conventional and proposed architectures. It shows that the proposed architecture has outperformed the conventional architectures in all the design parameters. As mentioned in the previous sections that the proposed architecture was built with prime intention to reduce the leakage power and the Table I demonstrates the exact outcome of the proposed architecture. It has reduced 16% of dynamic power than the corresponding existing architecture. The adder required in this set-up was of only 8-bit and hence only eight full adders cells were optimized. For larger bit-width PID controller, the number of adders will be more and the optimizations will also be higher.

The proposed approximate arithmetic concept was applied to the full adders & multiplier of the PID controller architecture. The proposed concept of dynamic power reduction importance hold good (higher than other parameters) even at this level and this proves that the proposed technique behave similarly at any hierarchical level of the design cycle. The results of the Table I show that the datapath optimizations impacts significantly for all design parameters. The proposed architecture has reduced 16% of dynamic power of the PID controller. Results from the Table suggest that the datapath architectural optimizations are unique. Hence it is a generalized solution which can be applied to any hierarchical level in the design cycle and the results will be intact for any bit width.

VI. CONCLUSION

The data path for adder and multiplier are designed for the discrete PID controller which consumes less power is proposed. The design is synthesized and simulated using the Xilinx tool, and the obtained results are compared with the existing design for area, power and timings. To improve the speed and minimize the cost while offering the good performance, the proposed architecture involves adder and multipliers which will consumes less amount of power when compared with that of previous architectures.

In this paper, a low power datapath architecture for PID controller applications has been illustrated using industry standard FPGA design methodology. Reduced power datapath architectural optimizations were proposed for PID controller running with lower technological node cells where leakage power is one of the primary design constraint. The proposed architecture has reduced 16% of dynamic power. Further analysis of the proposed architectures suggests that the proposed datapath architectural optimizations are unique & it is a generalised solution which can be applied to any hierarchical level in the design cycle and the results will be intact for any bit width.

VII. FUTURE WORK

Further optimizations can be explored at the transistor level designs by implementing the low power design methodologies effectively. There is scope to further optimize power by using proposed architectures. Further power reduction is possible by applying the transistor level optimizations techniques.

REFERENCES

[1] K. Warwick and D. Reeds, Industrial Digital Control Systems. London, U.K.: Peter Peregrinus, 1988.
[2] D.Wilson, ―16-bit DSP servo control with MC68HC16Z1,‖ Microprocessors Microsyst., vol. 18, no. 2, pp. 109–117, 1994.
[3] K. Astrom and H. Steingrimsson, ―Implementation of a PID controller on a DSP, Texas Instruments Inc. Digital Signal Process. Products, 1990.
[4] G. Norris, ―Inside the 777,Flight Int., pp. 23–26, Apr. 1994.
[5] G. Franklin, J. Powell, and A. Emami-Naeini, Feedback Control of Dynamic Systems. Reading, MA: Addison-Wesley, 1994.
[6] J. B. M. Anderson, Optimal Control: Linear Quadratic Methods. Englewood Cliffs, NJ: Prentice-Hall, 1990.
[7] R. Sanchez- Peña and M. Sznaier, Robust Systems Theory and Applications. New York: Wiley, 1998.
[8] J.-J. Slotine and W. Li, Applied Nonlinear Control. Englewood Clifs, NJ: Prentice-Hall, 1991.
[9] D. A. White and D. A. Sofge, Handbook of Intelligent Control Neural, Fuzzy, and Adaptive Approaches. NewYork: Van Nostrand Reinhold, 1992.
[10] H. Hanselmann, ―Implementation of digital controllers—A survey,‖ Automatica, vol. 23, no. 1, pp. 7–32, 1987.
[11] A. Oppenheim and R. Schafer, Discrete-Time Signal Processing. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[12] ChandraMohan U, “High Speed Squarer”, Proceedings of the 8th VLSI Design and Test Workshops, VDAT, August 2004.