International Journal of Power Control Signal and Computation (IJPCSC)
Vol. 4 No. 2 April- June -2012
©gopalaxjournals,singapore
ISSN:0976-268X
Available at : http://ijcns.com

DESIGN OF INTELLIGENT PID CONTROLLER BASED ON PARTICLE SWARM OPTIMIZATION IN FPGA

S.Karthikeyan[1] Dr.P.Rameshbabu[2] ,Dr.B.Justus Robi[3]
[1]S.Karthikeyan, Research scholar JNTUK., Department of ECE, KVCET,Chennai
[2]Dr.P.Rameshbabu, Associate Professor Pondicherry University, Pondicherry
[3]Dr.B.Justus RobiProfessor,M.G.R.University Chennai

***Abstract*— This paper explains a method for the design of Intelligent PID controller based on Very large scale integrated circuits (VLSI). In PID controller parameters are tunned with particle swarm optimization (PSO) algorithm. The error is identified and the PSO algorithm control the system with many iteration of different parameters.PID controller parameters such as gain, delay, lag and various time are optimized to solve the problems. A new evaluation function including the system adjusting time, rise time, over-shoot, and system error is defined. The circuits were tested and implemented as control devices intelligent PID controller on VLSI devices for laboratory plant. The algorithm applied to the PID controller that minimizes the iterations and rapid control action reduces time. The systems time delay and performance improved .The response of a system adopts the optimization technique. Finally the intelligent PID controller implemented in FPGA**

***Keywords*— VLSI, FPGA, intelligent PID, PSO, lag, system error, over-shoot**

I. INTRODUCTION

There are basically two methods for implementing control system based on digital technology. The first method is using software which implies a memory-processor interaction. The memory holds the application program while the processor fetches, decodes, and executes the program instructions..ProgrammableLogicControllers (PLCs) microcontrollers, microprocessors, DigitalSignal Processors (DSPs), and general purpose computers are tools for software implementation. The second method is based on hardware. Here we are using Very log integrated circuits (VLSI)VLSI devices are configurable ICs and used to implement logic functions. They ensure ease of design, lower development costs, more product revenue, and the opportunity to speed products to market. At the same time they are superior to software based controllers as they are more compact, power-efficient, while adding high speed capabilities [1]. VLSI devices are high-density Programmable Logic Devices (PLDs) with, usually; more than a million gates and 100s of I/O ports. Most of the manufacturers of the VLSI devices provide software to program these devices. Hardware Description Languages such as VHDL and Verilog, which were initially used to describe and simulate digital logic circuitry, are currently being used to implement complex algorithmic logic and then loaded into the VLSIdevices.Aiming at the problem that it is difficult to confirm the parameters of the PID controller and the parameters can not be changed once identified, an intelligent PID control method is proposed. According to thesize of the system error,this algorithm controls the system with different subsections of different parameters, by using the particle swarm optimization (PSO) to optimize the parameters of the PID controller to solve the problems, such as lag, time-variety and nonlinearity. A new evaluation function including the system adjusting time, rise time, over-shoot, and system error is defined. A group of parameters of the intelligent PID controller that minimize the evaluation function is calculated rapidly by searching in the given controller parameters area. Numerical simulations show that the controller can easily be applied to the first-order and second-order systems with time delay and its performance is also better than the classic PID optimized by PSO

II. PID CONTROLLER

AnOne of the most powerful but complex controller mode operations combines the proportional, integral, and derivative modes with a control loop feedback mechanism widely used in industrial control system. A PID controller corrects the error between a measured process variable and a desired set point. It calculates the difference between the two and then outputs a corrective action thorough the feedback.A proportional-integral-derivative controller is generic control loop feedback mechanism widely used in industrial control systems. The controller attempts to minimize the error by adjusting the control inputs. The controller can provide control action designed for specific process requirements. The Problem is

International Journal of Power Control Signal and Computation (IJPCSC)
Vol. 4 No. 2 April- June -2012
©gopalaxjournals,singapore
ISSN:0976-268X
Available at : http://ijcns.com

difficultto confirm the parameters of the PID controller.A group of parameters of the PID

controller that minimize the evaluation function is calculated rapidly by searching in the given controller parameters area**.**

The analytical equation is:

$$u(t) = K_P e(t) + \frac{1}{(K_I)} \int e(t)dt + K_D \frac{de(t)}{dt} + \text{PI}(0)$$

---- (1)

Where,
Kp = proportional gain
KD = derivative gain
e(t) = error in % of full scale range
K_I = integral gain
PI(0) = value of integral term at t=0

There are various tuning algorithm available in the literature.

III. PID CONTROLLER TUNING

A typical closed loop system using a PID controller is shown in Fig.2 The control system usually requires units to interface it to the environment. For instance, a converter to PWM (Pulse-Width Modulation) may be needed when controlling DC motors. The digital PID controller can be described by the following difference equation:

$$u(k) = u(k-1) + \propto 0e(k) + \propto 1e(k-1) + \propto 2e(k-2) \quad .(2)$$

Where the coefficients ∝0, ∝1, and ∝2 are evaluated by the expressions:$\propto 0 = Kc\left(1 + \frac{Td}{Ts}\right)$,

$$\propto 1 = -Kc\left(1 + 2\frac{Td}{Ts} - \left(\frac{Ts}{Ti}\right)\right),$$

$$\propto 2 = -Kc\frac{Td}{Ts} \quad \text{.....} \quad \text{.....(3)}$$

The Kc, Ti and Td, are PID parameters for tuning, and Ts is the sampling period in seconds. There are several methods for evaluating the PID parameters, generally called PID tuning methods [2].When controlling time-invariant processes, the PID parameters can be constants and evaluated off-line, so, the PID architecture may use fixed values for the ∝0, ∝1, and ∝2 coefficients. Otherwise, for time-variant processes there is a need to update those parameters; in this case the PID architecture has Kc, Ti and Td as parameters that can be automatically updated during runtime by auto-tuning algorithms. A complete implementation of the PID controller with auto-tuning requires a component responsible for the auto tuning algorithm, whose complexity largely depends on the auto-tuning algorithm used.

The auto-tuning feature is required in most control systems for mobile robotics due to the changes that may occur in the environment and/or system. Those modifications usually need the retuning of parameters to still have a stable control system with acceptable performance criteria's. In general, it could be useful that a controller implementation accommodates both type of numerical representation: fixed- and floating-point. In VLSI implementations architecture might be preferred. However, the evaluation of the number of bits for integer and fractional parts of each operand in the system is a very time consuming procedure. In this paper we propose a methodology for design and implementation of i-PID controllers in VLSI with exploitation of the number of bits for fixed-point-representations[7].

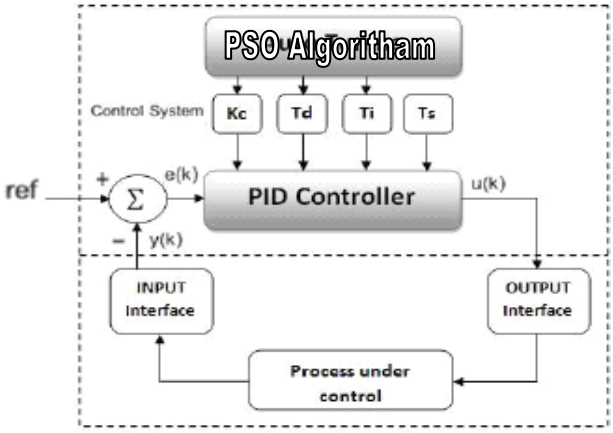


Fig.1. Typical control loops with a PID controller

Figure.1. Type control loops with a PID controller

IV.INTELLIGENT PID CONTROLLER WITH PSO

An innovative design method of PID controller based on particle swarm optimization (PSO) algorithm and FPGA is introduced. At first, with the Simulink of MATLAB, the optimized parameters of PID controller are obtained based on PSO in the closed-loop control system. Then under FPGA's controller based on FPGA and PSO algorithm is verified, and it has some advantages such as design flexible, self-tuning on line, high reliability, short cycle of technical development, high running speed and so on. This intelligent PID controller based on FPGA and PSO algorithm can be applied in high control precision digital control system with high speed responsibility and high stability.

International Journal of Power Control Signal and Computation (IJPCSC)
Vol. 4 No. 2 April- June -2012
©gopalaxjournals,singapore
ISSN:0976-268X
Available at : http://ijcns.com

V. VLSI IMPLEMENTATION

The various implementation schemes available in the literature are mainly focused on hardware efficient i-PID algorithms. The values of Kc, Ti and Td are incorporated as fixed assigned values. The PSO algorithm proposed will tune to various changes in parameters and then automatically control the set point. The implementation part demonstrated in the paper is for a given set of values of Kc, Ti and Td.. The auto tuning algorithm may be developed as per actual real time example. System Generator implements the design by considering the correct hardware platform and also takes care of the synchronization and interfacing problems.

A separate test bench application for hardware (VLSI) verification is also not required. The co-simulation block can be used with the same Simulink test bench apparatusesthat were used to test the original System Generator model. Along - disadvantages also, that are associated with the presented co simulation methodology/tools using automatic bit stream generation. With every release of System Generator, the top level output files change.

VI. PSO ALGORITHAM

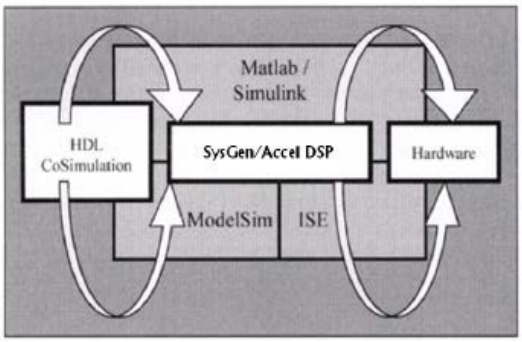


Fig 2 HW-SW Co-simulation using System Generator

The algorithm shown in Fig 3 will be the basis of a SoPC model for implementation of Digital adaptive controllers.

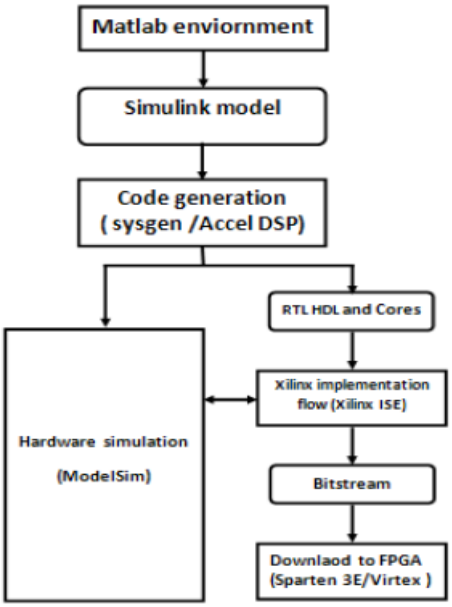
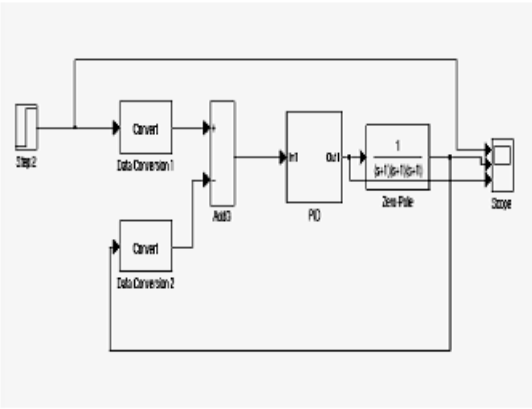


Fig 3 Design methodology [3]

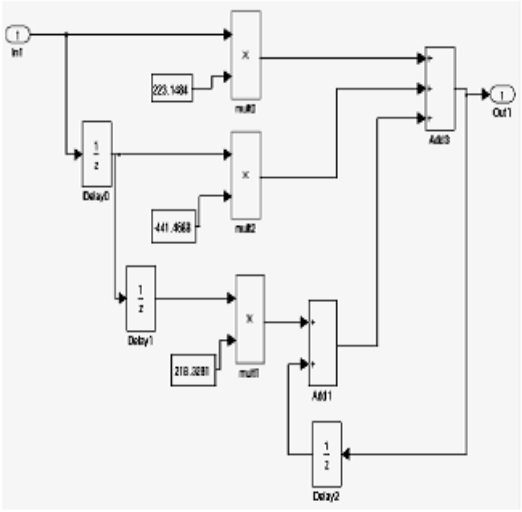
VII. DESIGN EXAMPLE

Here we have implemented multiplier based PID controller as suggested by Joao Lima, Ricardo Menott et all by using Xilinx SysGen tool box. As seen in the Fig.7.2 the floating-point values of the $\propto 0$, $\propto 1$, and $\propto 2$ coefficients are: 223.1538,-441.4616and218.3344, respectively & transfer function is G(s) = 1/(s+1)3. In the examples with fixed-point representations we use for $\propto 0$, $\propto 1$, and $\propto 2$ {9, 7},{10, 6} and {9, 6}, respectively [4]. Those representations allow representing those values as 223.1484, -441.4688 and 218.3281.18 bits are used for integer part and 7 bits are used for fractional part for the uniform fixed point representation all operands ($\propto 0$, $\propto 1$, and $\propto 2$ included) and operators. This implementation achieves none errors when compared to the floating-point implementation using the same values for $\propto 0$, $\propto 1$, and $\propto 2$ for the reference signal presented. When compared to the original floating-point values, although stable, the representation has a mean relative error of 7.9%. Note that reduction of bit-widths more than 20.9% makes the system unstable. Based on this, we can conclude that in this kind of digital systems error metrics (relative, absolute, etc.) may play a secondary role since they may not have the importance as in digital filter design. Here, we are firstly concerned with stability and then with precision [4].

International Journal of Power Control Signal and Computation (IJPCSC)
Vol. 4 No. 2 April- June -2012
©gopalaxjournals,singapore
ISSN:0976-268X
Available at : http://ijcns.com



(a)



(b)

Fig.4. Multiplier Based PID Controller

VIII. RESULTS

Here in fig.5 are included the simulation results of VLSI based tunable PID controller with multiplier. As depicted in results it provides better settling time.

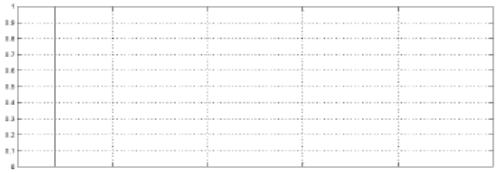


Fig.5 (a) Reference Input

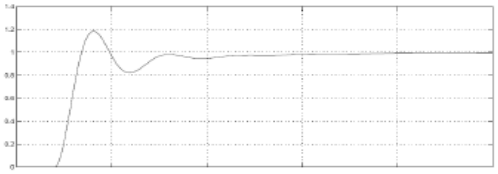


Fig.5 (b) Closed loop response

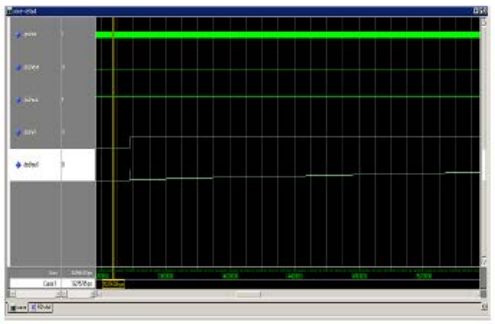


Fig.5 (c) Test bench timing diagram of PID controller

IXHARDWARE SETUP

Real time physical setup of the VLSI kit, ADC, DAC circuits, V/I converter with VLSI kit and host computer and physical setup of the three tank system which is configured for single tank process are shown in Fig6.1 to Fig 6.2

Figure 6.1: VLSI devices with interfacing circuits and host computer

International Journal of Power Control Signal and Computation (IJPCSC)
Vol. 4 No. 2 April- June -2012
©gopalaxjournals,singapore
ISSN:0976-268X
Available at : http://ijcns.com

Figure 6.2 : Synthesized circuit of Real to Floating point converter

X. CONCLUSION

Today's high-speed and high-density VLSI provide viable design alternatives to ASIC and microprocessor-based implementations. Several building modules for implementing PID controllers on these VLSI are constructed in this work. Implementing PID controllers on
VLSI devises features speed, accuracy, power, compactness, and cost improvement over other digital implementation techniques. In a future we plan to implement fuzzy logic controllers on VLSI devices.

REFERENCES

1. Wei Zhao, ByungHwa Kim, Amy C. Larson, and Richard M. Voyles., 'FPGA Implementation of Closed loop control System for Small Scale Robot', In Proceedings., 12th International conference on advanced robotics-ICAR05, 2005, pp.70-77.
2. Daijin Kim. "An implementation of Fuzzy Logic Controller on the Reconfigurable FPGA system", IEEE transactions on Industrial Electronics, Vol.47, No.3, 2000.
3. M.K.Gupta, "Implementation of Digital PID Controller on a single FPGA chip using VHDL", IEEE transactions on Industrial Electronics,
4. Y.F.Chang, M.Moallm and W.Way, "Efficient PID Controller Using FPGA", IEEE transactions on Industrial Electronics, Vol.35, pp. 119-125.
5. L.Samet, N.Masmoudi, M.W.Kharrat and L.Kamoun, "A Digital PID Controller for Real time and Multi Loop Control: a comparative study", in proceedings of 1998 IEEE international Conference on Electronics Circuits and Systems,Vol.1, 1998, pp. 291-296.
6. Akihiro Ozeki, HidejiFujikawa, "A Design Method of Self tuning controller for unknown time delay system", IEEE transactions on Industrial Electronics, 1992, pp.1218-1223.
7. Paul Leisher, Christopher Meyers.Dr.T.Stewart and Dr.G.Dempsey, "FPGA Implementation of a PID controller with DC Motor Application", Bradley University, a project report, 2002.
8. IEEE-754, a binary floating point arithmetic standard and IEEE 1076, a standard of VHDL from www.ieeexplore.ieee.org, 2002.
9. J.Bhasker, 'A VHDL primer', Pearson Education, India, 2005.
10. Stephen Brown, 'Digital logic design', Tata Mechgraw Hill publications, India, 2002.
11. Design tips for HDL implementation of Arithmetic functions from Xilinx tutorials www.xilinx.com