A REAL TIME SIGNAL PROCESSING APPLICATION IN FPGA AND FPAA

RAHUL P R
PG Scholor, Dept. Of Electronics & Communication
TKM Institute of Technology,Kollam, Kerala,India
rahulpram@gmail.com

***Abstract*—Introduction of reconfigurable device technology provide tremendous changes in the VLSI technology for the last few decades. Recently the VLSI designers reasearches in the area of analog reconfigurable technology due to some limitations in the digital reconfigurable devices. This paper presents the importance and advantage of analog reconfigurable device and digital reconfigurable devices for real time signal processing applications. We discuss the major issues of FPGA based on the recent advancements in the field of signal processing. The comparison of digital reconfigurable device (FPGA) and analog reconfigurable device (FPAA) is based on a case study of a PID controller. Based on the simulation results of the two technologies, we discuss many design issues, such as accuracy, size and ease of implementation.**

Keywords*—*Analog Reconfigurable Device (FPAA), Digital Reconfigurable Device (FPGA), PID Controller Real Time Signal Processing Applications

I. INTRODUCTION

Recent trends in VLSI focus towards the development of mixed signal arrays, which contains Field Programmable Analog Arrays [FPAA], Field Programmable Gate Arrays [FPGA] and mixed signal interface comprised in a single platform. The single array used for processing both analog and digital signals, which increases the reliability of signal processing applications. Paul chow and P Glenn Gulak, which proposed a field programmable mixed signal array [FPMA], which contains separate FPGA and FPAA array and interconnect using data converters, such as the MADAR chip style. Richard B Wenderlich, Farhan Adil and Paul Hasler proposed a new mixed signal array as the FPAADD, which highly interleaves the digital and analog computational elements in a heterogeneous Manhattan style interconnect scheme.

Many FPGA are currently available in the market to implement digital circuits. The characteristics and many functions of FPGA are well known to designers and end users. More recently, researches has been going in the area of FPAA, which provide a low cost, low power, reusable and reconfigurable hardware technology. This new field of FPAA reprogrammable devices appears as a new challenge when dealing with real time control applications. The original idea in this paper is to analyze the analog reconfigurable device [FPAA] with digital reconfigurable device [FPGA] by using the simulation results of both for a particular signal processing application.

This paper is organized as follows, section II explains the methodology. Section III devoted to FPGA architecture. Section IV detailed the FPAA architecture. Section V presents the results and discussion.

II. METHODOLOGY

To compare the two technologies, we have chosen the ideal PID algorithm.

Ideal PID algorithm

$$V(t)= Kp.e(t)+Ki\int e(t)dt+Kd\frac{de(t)}{dt} \quad (1)$$

Here, the above algorithm was implemented in FPGA using digital components and also in FPAA using analog components. For more detailed study on the technology, the algorithm was designed its structural level. This helps to study the size and ease of implementation of the application in two technologies.

To study the technologies in detail, we implement a temperature controller using this PID algorithm. The PID controller output is verified for study the various design issues of the two technologies. The detailed diagram of the implementation shown below.

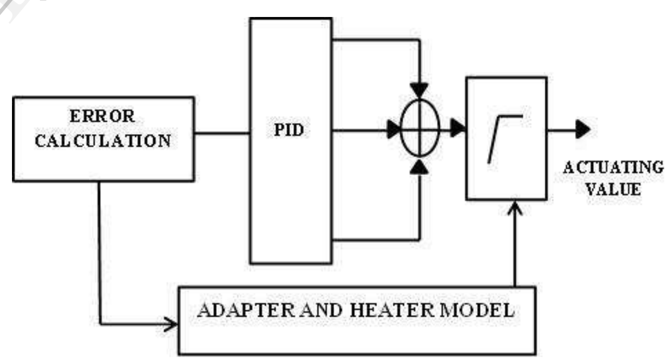

Fig. 1 Architeture Of The Temperature Controller

III. IMPLEMENTATION IN FPGA

The PID algorithm used here to implement in FPGA is the digital form of the ideal PID algorithm. This algorithm is known as position algorithm. Malfunctions in controller or communication loss and temporary loss of control can be tolerated by using this algorithm. In such cases, after control is restored the control system will act to recover and eventually reach the desired state.

By combining all the three modes of control

$$V(n) = Kp.e(n)+Ki.Ts\sum_{i=0}^{i=NT} e(i) + \frac{Kp}{Ts}[e(n)-e(n-1)] \quad (2)$$

A digital system is not a continuous system, it uses the sampled data or discrete control system. In a digital system, measurement of errors is available only at regular sampling intervals.

To overcome the disadvantage of truncation and to extend the the study for recent floating point support FPGAs, the input to the controller is given as IEEE-754 32 bit single precision formats.

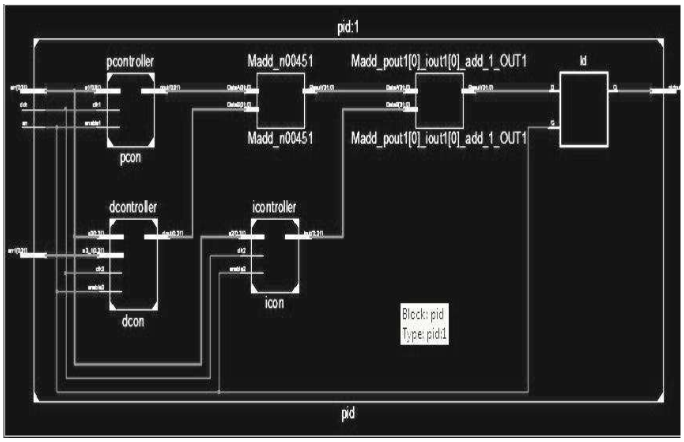

Fig.2 Block Diagram of the implemented PID algorithm

The controllers are designed using basic arithmetic modules and avoiding the use of integrators and differentiators, because, this will increase the efficiency of the PID algorithm, when it comes practical. Thus the simple arithmetic modules such as adders, multipliers, sub tractors and dividers are used for the design of each controller. The algorithm is implemented in structural level and coded using Verilog HDL. Fig.2 is the implemented PID algorithm for FPGA. Here we use Spartan- XC3S500E to implement the PID controller.

IV. IMPLEMENTATION IN FPAA

The implementation process in FPAA is using the same ideal PID analog algorithm, we can use the algorithm for realizing the analog components needed for the controllers. Analog controllers are the most popular type of controllers used in industry as date and have been used since the inception of control system. Modern FPAA uses switched capacitor circuits instead of resistors. It helps to reduce the size and area used in the FPAA device.

Operational amplifiers are the building blocks of the FPAA. The error signal entering the controller is processed by proportional, integral and derivative actions, which is used to implement the PID algorithm. The gain of these actions is given through the CAM properties, which is calculated using the PID gain equations.

There is a limited number of FPAA manufactures, among them here we use the FPAA from Anadigm. It has many advantages and reduce the complexity of analog projects, the designer can use the blocks directly after realizing the structure. The Anadigm Designer development systems provide the free software for us.

The Anadigm provide many types of FPAA chips, here we use Anadigm's AN221E04 for the implementation. For the whole implementation of the project we use two FPAA devices, one for the PID controller and other for the temperature controller as shown below.

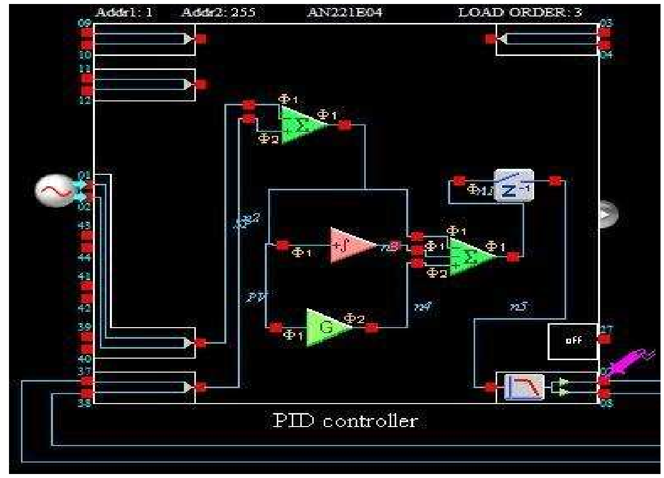

Fig.3 FPAA for PID controller

V. RESULTS AND DISCUSSION

The input for the FPGA is given as IEEE-754 32 bit single precision floating point numbers. The structural level implementation helps to study the area used in the FPGA device and ease of implementation. The whole PID controller architecture was designed using Verilog HDL for Xilinx Spartan-3E. The Behavioral simulation is performed using Modelsim XE-5.7g. The digital form of the output is shown in Fig.5.

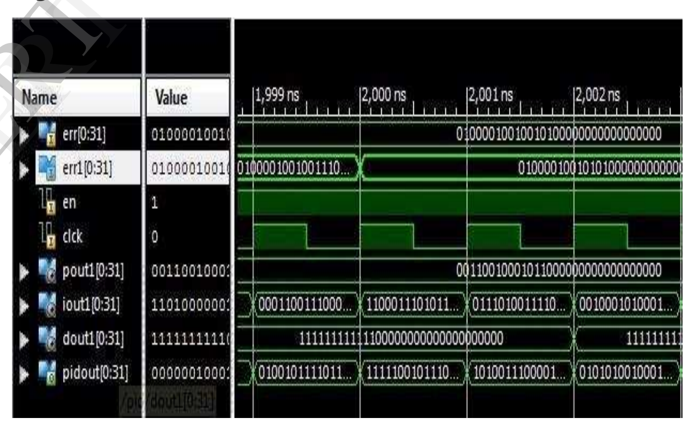

Fig.5 Simulation result of PID controller

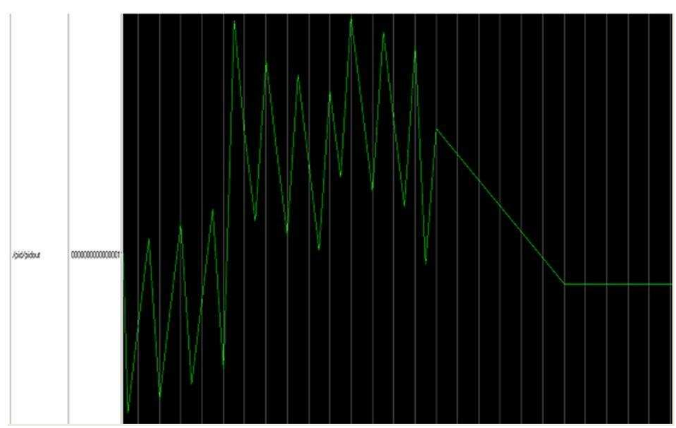

Fig.6 Analog format of the PID controller

The Fig.6 shows the simulation result of the PID controller in analog format. When the digital signals are converted to analog format, the actual values are round off to the nearest integer value and due to this, the analog signal get distorted. The result shows the problems related with FPGA. The distortions in the wave reveal the limitations of FPGA in the field of analog signal processing applications.

The simulation for the FPAA was accomplished by using Anadigm Designer2 software tool. The input to the FPAA is given as the analog format itself, here the simulation result of the derivative controller shown in the fig.7.

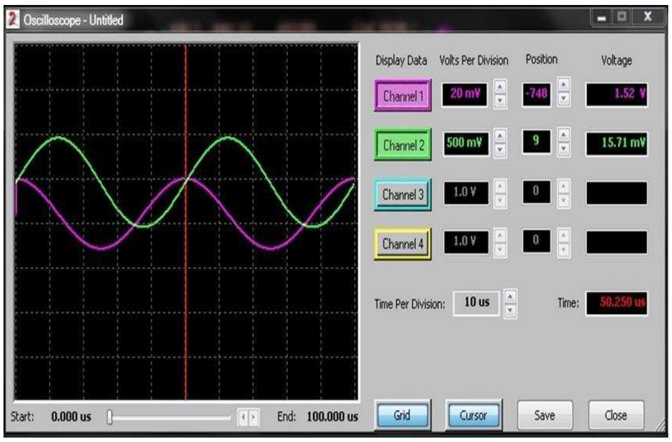

Fig.7.Simulation result of a derivative controller

Implementation procedure of the controller in FPGA is more complex than in FPAA. FPGA needs separate signal converters for the conversion of analog input signal to digital form, but in FPAA we can provide the analog input signal without using signal converters. The size and area used by the two devices is an important factor in VLSI technology. The simulation results show that in FPGA approach uses only 12% of the device capacity, but in FPAA we need two devices to implement the same.

The FPGA technology has many latest options in the device selection. More recently we have FPGA's with floating point number support. So here to compensate that we use IEEE-754 floating point representation for giving the inputs. Due to the limitations in the FPAA devices, we are not able to compare the two technologies by considering all the design aspects. For the digital signals applications FPGA is the best solution, while we using analog signals FPAA is the best alternative.

The reconfigurable technologies need an efficient development platform for the design. FPGA has many development tools, which is very efficient and has many resources for the design of complex projects. The tools available for the FPAA are limited, and here the manufacture Anadigm, provides the development tool with predesigned analog blocks. This reduces the complexity in the programming and support the designer.

When implementing the project, the power consumption for the two devices is an important factor to be considered. The estimated power consumption for FPAA device is more than FPGA. In practical FPGA need external devices for implementation, that will increase the overall power consumption for FPGA.

VI. CONCLUSIONS

Reconfigurable devices such as FPGA achieve much advancement in the last few decades. In the field of signal processing applications FPGA has a vital role but in the field of real time signal processing applications, FPGA has the limitations to process analog signals. Several design issues such as accuracy, power consumption and ease of implementation has to be taken in to account, while we choose a device for a particular application. FPAA devices are one of the alternatives for the FPGA, when we dealt with analog signal processing applications. FPAA devices provide accuracy and reduce the area used in the device for signal processing applications.

Future works will focus on the other areas of the signal processing applications and compare with the latest FPGA devices available in the market.

REFERENCES

[1] Richard B. Wunderlich, Farhan Adil, and Paul Hasler, Senior Member, IEEE "Floating Gate- Based Field Programmable Mixed-Signal Array" *IEEE Transactions On Very Large Scale Integration (VLSI) Systems*, Vol. 21, No. 8, August 2013.

[2] Gregory K. McMillan, "*Industrial Applications of PID Control*", PID Control in the third Millennium: Lessons Learned and New Approaches, Springer 2012.

[3] Li Bing, Chenhua, "The research of the PID Controller Based on FPGA", *International Conference on Electronic & Mechanical Engineering and Information Technology*, August 2011.

[4] Gregory K. McMillan, "Industrial Applications of PID Control" *IEEE Transactions on Industrial Control Systems*, August 2010.

[5] Tyson S. Hall, Christopher M. Twigg, Paul Hasler, and David V. Anderson, "Developing Large–Scale Field–Programmable Analog Arrays*", Proceedings of the 18th International Parallel and Distributed Processing Symposium*, April 2004.

[6] Tyson S. Hall, Paul Hasler, David V. Anderson, "Field Programmable Analog Arrays: A Floating gate Approach", *Proceeding of: Field-Programmable Logic and Applications, Reconfigurable Computing Is Going Mainstream, 12th International Conference*, Sept 2002.

[7] J. Rose, A. El Gamal, and A. Sangiovanni- Vincentelli, "Architecture of field-programmable gate arrays," *Proc. IEEE*, vol. 81, no. 7, pp. 1013– 1029, Jul. 2003.

[8] D. Anderson', C. Marcjan', D. Bersch'. H. Anderson', P. Hu. 0. Palusinski, D. Gettman, I. Macbeth, A. Brat, "A Field Programmable Analog Array and its Application", *IEEE Custom Integrated Circuits Conference*, 1997.

[9] Haydar kutuk , Sung-Mo (Steve) Kang, "A Field-Programmable Analog Array (FPAA) Using Switched-Capacitor Techniques", *Circuits and Systems, ISCAS'96, Connecting the World., IEEE International Symposium,* May 1996.

[10] Edward K. F. Lee', P. Glenn Gulak, "A Transconductor-Based Field-Programmable Analog Array", *IEEE International Solid-Stde Circuits Conference*, 1995.

[11] Edward K. F. Lee and P. Glenn Gulak, "A CMOS Field-Programmable Analog Array", *IEEE Journal Of Solid-State Circuits*, Vol. 26, No. 12, December, 1991.