We are IntechOpen, the world’s leading publisher of Open Access books
Built by scientists, for scientists

5,000
Open access books available

125,000
International authors and editors

140M
Downloads

154
Countries delivered to

TOP 1%
Our authors are among the most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™
Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected. For more information visit www.intechopen.com
PID Controller Using FPGA Technology

Abdesselem Trimeche, Anis Sakly, Abdelatif Mtibaa and Mohamed Benrejeb
Ecole Nationale d'Ingénieurs de Monastir-Département de Génie Electrique,
Laboratoire d’Electronique et de microélectronique - Université de Monastir,
Unité de recherche LARA Automatique de l’Ecole Nationale d’Ingénieurs de Tunis,
Tunisia

1. Introduction

Since the Years 60, the law of Moore predicts that the complexity in terms of built-in circuit transistors doubles every two years, remain verified. The programmable FPGA circuits (Field Programmable Gate Array) didn't escape to this law. Since the first FPGA, developed like an evolution natural of the CPLD (Complex Programmable Logic Devices), these circuits didn't stop winning in complexity and integrated henceforth until one billion of transistors for the most recent generations. This increase of the integration level resulted in a similar growth of the power of calculation of these circuits. The FPGAs have been used then to make the fast samples of ASICs (Application Specific Integrated Circuits) and find since some years their place in many domains of applications. However, the order of the processes industrial requires more and more elements of powerful calculations. This type of order is in the same way in perpetual evolution with the development of the numeric circuits of calculation. Thus, the PID controllers represent the majority of the controllers used in the industrial systems control. Of this fact, it will be necessary to digitalize the PID algorithm. The modern digital control systems require more and more strong and fastest calculation components. This type of elements becomes yet indispensable with the utilization of some new control algorithms like the fuzzy control, the adaptive control, the sliding mode control… [1]. Although the PID controllers are the oldest they represent the most used controllers in the industrial control systems.

2. Discrete PID equation

The PID algorithm consists of three basic modes, the Proportional mode, the Integral and the Derivative modes. When utilizing this algorithm it is necessary to decide which modes are to be used (P, I or D) and then specify the parameters (or settings) for each mode used. Generally, three basic algorithms are used P, PI or PID.

The implementation of PID controllers using microprocessors and DSP chips is old and well known [2] [3], whereas very little works can be found in the literature on how to implement PID controllers using FPGAs [4].

Field Programmable Gate Arrays (FPGA) have become an alternative solution for the realization of digital control systems, previously dominated by the general purpose microprocessor systems.
In our work we introduce a simple method for implementing PID controllers. Some other contributions focused on proposing algorithms for tuning the coefficients of PID controllers using FPGAs while the controller itself is still implemented in software.

In [5] the authors describe the architecture of a data acquisition system for a gamma ray imaging camera.

In the past two years, Spartan II and III FPGA families from Xilinx have been successfully utilized in a variety of applications which include inverters [6][7], communications [8][9], embedded processors [10], and image processing [11].

In our work, data acquisition for the PID controller, which is implemented using Xilinx Spartan-3 Starter Kit Board, is based on 8-bit serial A/D converters extensible from Digilent board AO1. Similar converters are utilized in [12] to implement an adaptable strain gage conditioner using FPGAs.

The application of a PID controller in a feedback control system is shown in fig.1, where ref is the set point signal, y is the feedback signal, e is the error signal, and u is the control input.

The simplest form of the PID control algorithm is given by:

$$u(t) = k_p \left( e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right)$$

According to the study done in [13] the digitized PID equation is brought back to:

$$u_k = u_{k-1} + b_0 \cdot e_k + b_1 \cdot e(k - 1) + b_2 \cdot e(k - 2)$$

Where the coefficients $b_0$, $b_1$, and $b_2$ are evaluated by the expressions:

$$b_0 = k_p \cdot \left( 1 + \frac{T_d}{T} \right);$$

$$b_1 = k_p \cdot \left( -1 + \frac{T}{T_i} - 2 \frac{T_d}{T} \right);$$

$$b_2 = k_p \cdot \frac{T_d}{T}$$

The Kp, Ti and Td, are PID parameters for tuning, and T is the sampling period in seconds.
3. Digital PID architecture

To improve the speed and minimize the cost while offering clearly good performances, the adopted architecture used includes essentially three combinational logic multiplier, one substractor three adders and three registers. The fig 2 gives the adopted architecture. Indeed, this architecture requires the availability of all calculation operators in each phase.

![Fig. 2. PID Architecture.](image)

3.1 Conversion blocks presentation

The AIO1 board is a peripheral board designed to work with Digilent’s family of system boards. The AIO1 contains analog-to-digital and digital-to-analog converters from Analog Devices, two dual op amps, a variety of analog signal I/O connectors, and a solderless breadboard. All analog components use an on-board 5VDC voltage source. All unused I/O signals are passed through the AIO1 board so that it can be used between a system board and other peripheral boards.

The AIO1 uses an 8-bit, 200Ks analog-to-digital converter (the AD7823), and an 8-bit, 1 MHz digital-to-analog converter (the AD7303), both from Analog Devices. The AD8534 op amps (also from Analog Devices) can drive 250mA outputs rail-to-rail with a 3 MHz bandwidth, so many useful devices can be driven directly.

Fig 3 describes all the components of this block.

3.2 Analog input interface

FPGAs are well suited for serial Analog to Digital (A/D) converters. This is mainly because serial interface consumes less communication lines while the FPGA is fast enough to accommodate the high speed serial data. The AD7823 is a high speed, low power, 8-bit A/D converter. The part contains a 4 μs typical successive approximation A/D converter and a high speed serial interface that interfaces easily to FPGAs. The A/D interface adapter (ADIA) is implemented within the FPGA (Figure 5). Inside the FPGA, this adapter facilitates parallel data acquisition. Sampling is initiated at the rising edge of a clock applied at the line sample. The timing diagram of the communication protocol is illustrated in figure 4. The whole conversion and acquisition period is 5.4 μs allowing sampling up to a rate of 185 Kilo Sample per second. This rate is more than sufficient for most PID control applications.
The output coding of the AD7823 is straight binary. The designed code transitions occur at successive integer LSB values (i.e., 1 LSB, 2 LSBs, etc.). The LSB size is $= V_{REF}/256$. 

www.intechopen.com
3.3 Analog output interface

The AD7303 is a dual, 8-bit voltage output Digital to Analog (D/A) converter. This device uses a versatile 3-wire serial interface that operates at a clock up to 30 MHz. The serial input register is 16 bits wide; 8 bits act as data bits for the D/A converter, and the remaining 8 bits make up a control register. It is interfaced to an FPGA as illustrated in Figure 6. The D/A interface adapter (DAIA), which is implemented within the FPGA, facilitates parallel data input for the dual D/A converters. The timing diagram of the communication protocol is illustrated in figure 7. The transmission period of a sample is 680 ns allowing D/A conversion at an excellent rate of 1.47 MHZ.

Any DAC output voltage can ideally be expressed as:

\[ V_{OUT} = 2 \times V_{REF} \times \left( \frac{N}{256} \right) \]

where: N is the decimal equivalent of the binary input code. An N range from 0 to 255xVREF is the voltage applied to the external REF pin when the external reference is selected and is VDD/2 if the internal reference is used.

3.4 Implementation results

The proposed based PID controller is implemented using the Xilinx Inc FPGA technology and can be used as a general purpose controller for different applications. The simulation results obtained with the generated VHDL, in this work, the ModelSim® simulator was used. The circuits for the PID controllers have been obtained by logic synthesis and place
and route using Xilinx ISE 7.1i, from the VHDL representation generated by the static
analyzer. We use a Xilinx Spartan-3 xc3s200-ft256-4 FPGA. The results presented herein are
estimations directly obtained from Xilinx ISE 7.1i.

Fig. 8. Design properties.

Table 1 shows the minimum number of multiplications, additions and registers required for
the PID controller without conversions block.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplication</td>
<td>3</td>
</tr>
<tr>
<td>Addition</td>
<td>3</td>
</tr>
<tr>
<td>subtraction</td>
<td>1</td>
</tr>
<tr>
<td>register</td>
<td>3</td>
</tr>
<tr>
<td>total</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 1. Arithmetic Number for PID controller.

The PID controller block, into a complete control system consisting of analog and digital
I/O, is illustrated in figure 9.

Fig. 9. PID Controller Block.

The simulation results adapted to this block is shown in figure 10.
Fig. 10. Simulation diagram of PID controller block.

The synthesis of PID controller block using a Xilinx Spartan-3 xc3s200-f256 -4 FPGA gives the following results.

![Simulation diagram of PID controller block.](image)

### Device Utilization Summary

<table>
<thead>
<tr>
<th>Logic Utilization</th>
<th>Used</th>
<th>Available</th>
<th>Utilization</th>
<th>Note(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slave Flip Flop</td>
<td>128</td>
<td>3,840</td>
<td>5%</td>
<td></td>
</tr>
<tr>
<td>Number of 4 input LUTs</td>
<td>536</td>
<td>3,840</td>
<td>14%</td>
<td></td>
</tr>
</tbody>
</table>

### Logic Distributions

- Number of occupied Slice: 300
- Number of Slices containing only related logic: 368
- Number of Slices containing unrelated logic: 0

- Total Number of input LUTs: 650
- Number used as logic: 536
- Number used as a route thru: 111

- Number of bonded I/Os: 24
- Number of I/Os: 1

Fig. 11. Devices utilizations summary for the PID controller Implementation.

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.

4. Application of PID controller using FPGA technology to command a system of second order

4.1 System presentation

The figure 12 describe the system of second order.

www.intechopen.com
The values of all the components are:

- \( R = 12 \, k\Omega \)
- \( R_1 = 15 \, k\Omega \)
- \( R_2 = 10 \, k\Omega \)
- \( C_1 = 6,8 \, nF \)
- \( C_2 = 22 \, nF \)

The Transfer function of the system is given by the following equation

\[
G(p) = \frac{v_o(p)}{v_i(p)} = \frac{k_i}{1 + \frac{2\pi \omega_0}{m} p + \left(\frac{1}{m}\right)^2 p^2}
\]  \hspace{1cm} (4)

With:

- \( K_s = \frac{R_2}{R_1} = 0,67 \): static gain
- \( \omega_0 = \frac{1}{R_1 \sqrt{C_1 \cdot C_2}} = 6,81 \cdot 10^3 \, rad/s \)
- \( m = \sqrt{\frac{C_2}{C_1}} = 1,80 > 1 \): Amortization factor

While permuting the positions of \( C_1 \) and \( C_2 \), the amortization factor becomes:

\[
m = \sqrt{\frac{C_1}{C_2}} = 0,56 < 1
\]

### 4.2 Experimental result

At the time of the order of the system two cases appear; when the amortization is \( m > 1 \) and \( m < 1 \).

#### 4.2.1 Experimental results for an amortization \( m > 1 \)

**4.2.1.1 System answer results in open buckle**

For an order (ref) of the order of 2V applied to the system in BO one gets the answer \( y(t) \) presented on the following figure.
Fig. 13. System Answer in open buckle ($m > 1$).

According to this answer, one verifies that the tension of exit stabilizes without oscillations ($m>1$) nearly to the value $1.4V$, that corresponds more or less to the theoretical value:

$$V_2 = K_s \cdot V_c = 0.67 \cdot 2 = 1.34V.$$

4.2.1.2 System answer results with P regulator

For an order (ref) of the order of $2V$ applied to the system ordered by a Proportional regulator (P) with $KP = 2$, one gets the answer $y(t)$ presented on the following Figure

Fig. 14. Answer of the system ordered by proportional regulator P.
One notices that this answer presents a static mistake of the order of 40%. Theoretically this mistake is given by:

\[ \frac{1}{1+K_p K_i} \cdot 100\% = 42\%. \]

### 4.2.1.3 System answer results with PI regulator

For an order (ref) applied of the order of 2V to the system ordered by a PI regulator with \( KP=2, KI=0.5 \), one gets the answer \( y(t) \) presented in the following figure:

![Fig. 15. Answer of the system ordered by PI regulator \((m > 1)\).](www.intechopen.com)

One notices the annulment of the static mistake well thanks to the introduction of the I action.

### 4.2.2 Experimental results for an amortization \( m<1 \)

#### 4.2.2.1 System answer results in open buckle

For an order (ref) of the order of 2V applied to the system in BO one gets the answer \( y(t) \) presented on the following figure

![www.intechopen.com](www.intechopen.com)
According to this answer, one verifies that the tension of exit stabilizes with oscillations (m < 1) nearly to the value 1.4V.

4.2.2.2 System answer results with P regulator

For an order (ref) of the order of 2V applied to the system ordered by a Proportional regulator (P) with KP = 2, one gets the answer y (t) presented on the following Figure

**Fig. 16. Answer of the system in open buckle (m < 1).**

**Fig. 17. Answer of the system ordered by proportional regulator P.**
One notices that this answer presents a static mistake of the order of 40%. Theoretically this mistake is given by:

\[ \frac{1}{1 + K_p K_v} \cdot 100\% = 42\%. \]

### 4.2.2.3 System answer results with PD regulator

For an order (ref) of the order of 2V applied to the system ordered by a PD regulator with \( K_P = 2, K_D = 1 \) one gets the answer \( y(t) \) presented on the following Figure.

![Figure 18. Answer of the system ordered by proportional regulator P D.](www.intechopen.com)

One notices that the answer gotten present less oscillations that the one with the regulating \( P \) thanks to the Derivative action.

### 4.2.2.4 System answer results with PI regulator

For an order (ref) of the order of 2V applied to the system ordered by a PI regulator with \( K_P = 2, K_I = 0.5 \) one gets the answer \( y(t) \) presented on the following Figure.
one notices the annulment of the static mistake well thanks to the introduction of the I action

4.2.2.5 System answer results with PID regulator

For an order (ref) of the order of 2V applied to the system ordered by a PI D regulator with KP = 2, KD=1 and KI=0, 5 one gets the answer y(t) presented on the following Figure
One notices that with the addition of the Derivative action, one has a light reduction of the oscillations in relation to the answer gotten by a regulating PI.

4.3 Tentative evaluation of sampling period
While following the evolution of the order $u(t)$ one could estimate the value of the sampling period experimentally ($T$) as it’s indicated in the following figure:

![Fig. 21. Tentative evaluation of the sampling period ($T$).](image)

According to the figure 21, one estimates the value of the sampling period ($T$) that is the order of 6.7μs.

5. Conclusion
A digital PID controller implemented in FPGA technology is a configurable controller in terms of latency, resolution, and parallelism.
The speed or execution or latency of the controller can be precisely controlled with the amount of reuse of arithmetic elements such as the speed of execution of FPGA based PID controller can be less than 100 ns if desired for high throughput requirements. Implementing PID controllers on FPGAs features speed, accuracy, power, compactness, and cost improvement over other digital implementation techniques. In a future fork we plan to investigate implementation of fuzzy logic controllers on FPGAs. Also we plan to explore embedded soft processors, such as MicroBlaze, and study some applications in which design partitioning between software and hardware provides better implementations.

6. References


[4] Mohamed Abdelati, the Islamic University of Gaza, Gaza, Palestine:" FPGA-Based PID Controller Implementation".


Since the foundation and up to the current state-of-the-art in control engineering, the problems of PID control steadily attract great attention of numerous researchers and remain inexhaustible source of new ideas for process of control system design and industrial applications. PID control effectiveness is usually caused by the nature of dynamical processes, conditioned that the majority of the industrial dynamical processes are well described by simple dynamic model of the first or second order. The efficacy of PID controllers vastly falls in case of complicated dynamics, nonlinearities, and varying parameters of the plant. This gives a pulse to further researches in the field of PID control. Consequently, the problems of advanced PID control system design methodologies, rules of adaptive PID control, self-tuning procedures, and particularly robustness and transient performance for nonlinear systems, still remain as the areas of the lively interests for many scientists and researchers at the present time. The recent research results presented in this book provide new ideas for improved performance of PID control applications.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:
