БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ • BULGARIAN ACADEMY OF SCIENCES

ПРОБЛЕМИ НА ТЕХНИЧЕСКАТА КИБЕРНЕТИКА И РОБОТИКАТА, **61**
PROBLEMS OF ENGINEERING CYBERNETICS AND ROBOTICS, **61**

София • 2009 • Sofia

A Digital Filter with an Embedded Processor

Vladimir Ivanov, Nikolay Ivanov

Institute of Information Technologies, 1113 Sofia
E-mail: ivanovnikolay@hotmail.com

Introduction

For different reasons it may be announced, that the Digital Filters (DF) passed away from the area of fantasy and ceased to be a subject of study and use by a narrow circle of specialists only. In the cases when they are created in the core of FPGA devices, their efficiency becomes repeatedly higher than the one, achieved with special processors or ASIC.

All this gives the ground for reconsidering the whole ideological basis of the tools for DSP algorithms design and realization with the help of modern, contemporary element base. In this connection, special attention should be paid to embedded processors. The embedded processors are a product of human activity that can be changed, modified and similar to a biological virus, they leave a trace in the mind of everyone trying to work, examine or use it.

Thanks to their ability to accept different dimensions and structures, as well as their possibility to be inferred in different physical wafers, the embedded processors effectively dominate over conventional and specialized microprocessors.

In this way the embedded processors materialize the appearance of a new congruent point among the interests of users and manufactures and claim for the status of basic building blocks for all present and future high technological control devices and information systems and devices.

To attract the attention of designers to the wider use of the embedded processors, the purpose of this article is to demonstrate the possibilities for practical use of small embedded processors for digital filtration of signals in real time.

Basics of digital filtration theory

One of the most common operations in DSP is filtration. It is done with the help of digital filters, whose impulse characteristics can be finite (FIR filters) or infinite (IIR filters). Until the present moment one of the major difficulties for the practical realization of all types of digital filters was connected with the absence of fast long word digital multipliers.

Recently, after the emergence of new generations of families of FPGA devices [1, 2], this problem has lost its significance, which opened new possibilities for the practical implementation of the digital filters in various scientific and applied fields.

Regardless of the type of the filter, very often the obtaining of a certain characteristic can be achieved by the appropriate connection of filters of order 1 and 2. Therefore in the early stages of the design process of different devices, containing digital filters with complex characteristics, their realization must be accomplished on the basis of several built-in microprocessors, even only for the purposes of modeling.

The classical form of a description of the digital filter [3] is as follows:

$$y(n) = \sum_{i=1}^{N} b_i x(n-i) + \sum_{i=1}^{N} a_i y(n-i), \tag{1}$$

where $y(n)$ and $x(n)$ are the input and output sequences of the filter, a_i and b_i are coefficients of the filter, N is its order.

If the coefficients a_i and $b_i \neq 0$, the type of the digital filters is IIR. When all the coefficients $a_i = 0$, then the filter is of FIR type. The order of the filter, together with the values of the weighting coefficients a_i and b_i determine its impulse response.

This form of presentation gives information only about the quantity of the arithmetic operations for its physical realization. Regarding the frequency characteristics of the digital filter and its stability, the answer can be obtained from its representation as a rational polynomial in z region.

The structure of a first order digital filter is shown as an example in Fig. 1.

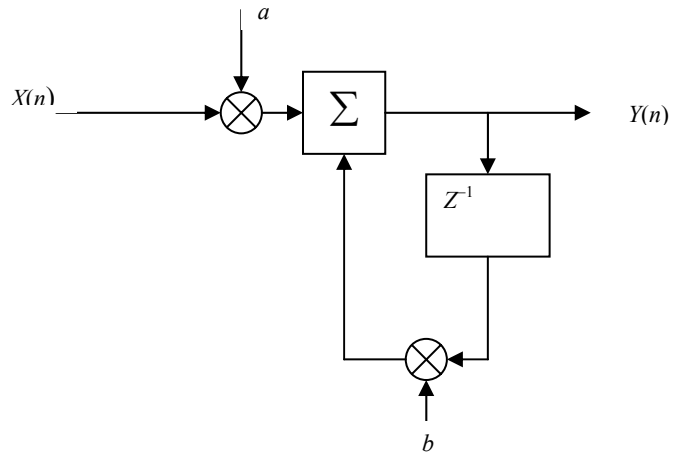

Fig. 1

The difference equation describing the relation between the input entered and the output result of the filter is of the form:

$$y(n) = ax(n) + e^{-\gamma T} y(n-1) \tag{2}$$

or

$$y(n) = ax(n) + by(n-1), \tag{3}$$

where $b = e^{-\gamma T}$.

In z domain the equation of this filter looks like:

$$H(z) = \frac{a}{1 - z^{-1} e^{-\gamma T}}, \tag{4}$$

where T is the sampling period, γ is the cut off frequency of the filter in rad/s.

In order to satisfy the conditions for a single gain of the filter [4], its coefficients can be written as $a = 1 - b$. Thus, the equation describing the filter type becomes:

$$y(n) = (1 - b)x(n) + by(n-1). \tag{5}$$

After certain algebraic conversions this equation, can be reduced to the form:

$$y(n) = y(n-1) + (x(n) - y(n-1))(1-b). \tag{6}$$

In the cases when multiplayer $1- b$ can be represented as a negative power of 2, this equation turns out to be extremely convenient for implementing with the help of conventional microprocessor [5], under assembly language programming.

If we take into account the features of modern, contemporary reprogrammable FPGA devices, the hardware realization of equation (6) becomes more attractive and easy.

Practical realization

As an example, that illustrates the practical implementation of this approach, a programmable synthesized processor PicoBlaze is selected. The main reason for this choice is based on the fact that up to the now it holds the palm of undisputed leader with the widest range of application [6] among all embedded processors known. PicoBlaze is developed by K e n C h a p m a n [7], it does not require any licence rights and with equal success it can be built in past, present and future generations of FPGA Xilinx [8]. PicoBlaze is distributed free in the form of a VHDL file.

If disregarding the specific value of the term $1 - b$ from the equation, (6) it can be seen that the implementation requires only a summation operation and some number shifts to the right. For the case when $1 - b = 1/8$, the assembler code of the program, realizing this first order filter looks like:

```
; KCPSM3 Program - Digital LP filter for Spartan-3 Starter Board.
; ------------------------------------------------------------------------------
Port definitions
            DSIN        $00             ; Address of input data port
            DSOUT       $07             ; Address of output data port
; Initialize the system
Imm:        EINT        ;interrupt  enable
            JUMP        Imm
; Interrupt service routine (ISR)
            ORG         $1FC
ISR:        IN          s0, 0     ;
            SUB         s0, s5    ;
            AND         s0, s0    ;
            ADDC s0, 4    ;
            SRA         s0        ;
            SR0         s0        ;
            SR0         s0        ;
            ADD         s5, s0    ;
            OUT         s5, 7     ;
            RETI        enable
; Interrupt vector
            ORG         $3FF
            JUMP        ISR
;
```

The normalized impulse, frequency and transfer characteristics of this filter are shown on Figs. 2, 3 and 4.

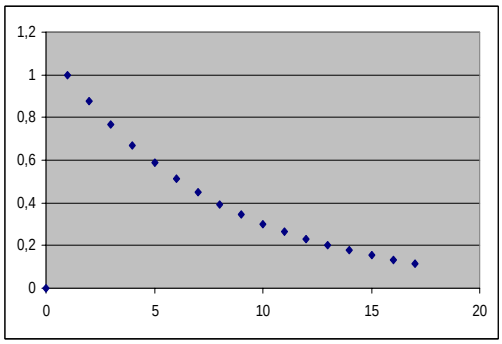

Fig. 2

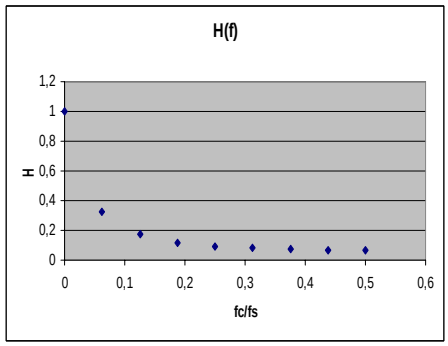

Fig. 3

To check the features of the filter thus constructed, it was modeled in pBlazeIDE environment. For this purpose, an input was entered in the form of a single step function.

$x(n)$ => 00,00,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF.

The response of the filter to this input is:

$y(n)$ => 00,00,20,3C,54,69,7C,8C,9A,A7,B2,BC,C4,CB,D2,D8,DD,E1,E5,E8,EB,EE,F0, F2,F4,F5,F6,F7,F8,F9,FA,FB,FC, FC, FC, FC.

The normalized function describing the signal at the filter output is shown in Fig. 5.

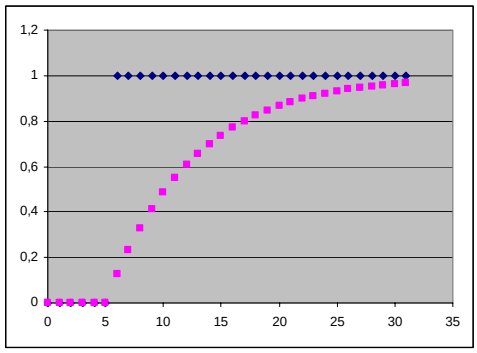

Fig. 4

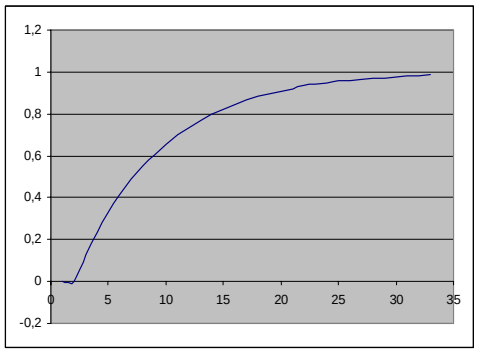

Fig. 5

The comparison between Figs. 4 and 5 shows good matching results.

Conclusion

Considered from the viewpoint of its practical implementation, this program is not large. Together with the code, processing the interruption, it is executed for 25 tacts.

Thus, its implementation in Spartan3 kit of (with a clock frequency of 50 MHz), enables the real-time processing of signals with a bandwidth range up to 1 MHz. If it is implemented in Virtex FPGA device, the 4 Mhz band with signals can be developed. Such productivity is very appropriate in the early stages of development of sophisticated equipment and advanced control and management systems, when the designers encounter the need to test different types of TSF – multi-cascade structures with decimation in time and frequency, Viner and Kalman filters, robust and adaptive filters in conditions close to the real ones.

Moreover, the ability of the embedded processors to meet more completely the requirements of each individual case, as well as the coordination of their cores with the fine internal architecture and characteristics of FPGA elements, are the main reasons for their considering as more profitable and promising, both from positions of present and future amendments, and also in terms of their rapid release on the market.

References

1. Бондаренко, М., В. Гриценко. Реализация фильтров с бесконечной импульсной характеристикой на FPGA.
2. Шипулин, С. Н., В. Ю. Храпов. Особенности проектирования цифровых схем на ПЛИС. – Chip News, 1996, No 5, p. 40.
3. Рабинер, Л., Б. Гоулд. Теория и применение цифровой обработки сигналов. М., Мир, 1978.
4. Оппенгейм, А. В., Р. В. Шафер. Цифровая обработка сигналов. М., Мир, 1979.
5. Raynos, A. 8-bit microcontroller implements digital low pass filter EDN 1/24/2008.

6. What Customers and Partners are Saying about Xilinx Embedded Processor Solutions. – Embedded Magazine, Issue 1, March 2005.
7. C h a p m a n, K. PicoBlaze 8-Bit Microcontroller for Virtex Devices XAPP213, 2002.
8. PicoBlaze 8-bit Embedded Microcontroller. User Guide, for Spartan-3, Virtex-II, andVirtex-II Pro FPGAs UG129, 2004.

Цифровый фильтр на встроенном процессоре

Владимир Иванов, Николай Иванов

Институт информационных технологий, 1113 София

(Р е з ю м е)

В работе рассматривается один подход к реализации цифрового фильтра на основе встроенного процессора PicoBlaze. Вычислены переходные и частотные характеристики фильтра. В работе приведен ассемблерский код фильтра.