DELTA OPERATOR-BASED DISCRETE SYSTEMS FOR FIXED-POINT DSP IMPLEMENTATIONS

Fernando Pinhabel Marafão Sigmar Maurer Deckmann Amauri Lopes

State University of Campinas – UNICAMP
School of Electrical and Computer Engineering – FEEC
P.O. Box 6101 – 13081-970 – Campinas – SP – Brazil
fmarafao@dsce.fee.unicamp.br

Abstract **– This paper presents a brief overview of usual discrete design procedures and discusses the application of digital signal processing on power electronics and power systems. An improved solution for digital processing using the unusual Delta-Transform instead of the conventional Z-Transform is also proposed. Special attention is deserved to the cases of high sampling frequency and fixed-point Digital Signal Processors (DSP) implementations. Since it is very frequent to use some kind of filter in the mentioned applications, the discussion has been directed to the implementation of practical digital filters. Simulations and experimental results validate the implementation of band-pass and band-stop filters using the delta operator in a 16 bits DSP.**

KEYWORDS

Delta operator, digital filters, fixed-point, finite word length, DSP, band-pass.

I. INTRODUCTION

Traditionally, designers working on power systems, power electronics or industrial applications are quite familiar with continuous system modeling. Thus, the *Laplace* Transform (s-Domain) has been the most useful tool when developing and analyzing a plant, controller or even a desired filter. However, new systems have been designed mostly using discrete systems, as a consequence of the advances on digital signal processing and the effective cost reduction of the modern Digital Signal Processors (DSP).

Essentially, there are two foremost methods to map from continuous to discrete-time domain, the impulse invariance and the bilinear transformations [3,6]. The latter is probably the most useful since it allows the representation of any sort of filter or system.

However, independently of the chosen transformation, when the sampling frequency is much higher than the Nyquist definition (e.g. greater than 50 times), stability problems may arise on the resulting discrete system [2]. Such problems can become significant in some practical applications, especially in fixed-point or finite word length DSP implementations.

Since the conventional and most practical tools in digital systems modeling are based on the Z-Transform and its shift operator [3,6], this paper presents a brief discussion of the problems related to discrete designing using z-Domain, specially when high sampling frequencies and quantization effects are present [5,6]. After that, discusses the utilization of the so-called *Delta-Transform* to solve these problems (-Domain) [1,2]. Such method is based on difference or derivative operator's theory and was firstly explored and presented by Middleton and Goodwin [1,2] and Feuer and Goodwin [4]. The Delta-Transform has been successfully applied in control theory, circuits and systems areas, particularly related to digital filtering [8] and estimation for digital control [9]. However, besides this success, it is quite uncommon to find any literature in power electronics [7,10] or power systems [11] applications, as recently mentioned in [7].

Such theory can be applied to any discrete system [4], but since it is frequent to use some kind of filter (low-pass, high-pass band-pass, band-stop) in power electronics or power systems modeling and controlling, this paper explores the results on implementing digital filters based on Z-Transform and the clearly enhanced results obtained when applying the Delta-Transform technique in fixed-point DSPs under high sampling frequencies.

Simulations of band-pass and band-stop filters validate the methodology of designing and implementing digital filters based on the delta operator and experimental results using a 16bits DSP development board confirm the expectations.

II. TRANSFORMATION FROM CONTINUOUS TO DISCRETE SYSTEMS

The canonical form of a linear continuous system is depicted in (1) and represents the transfer function between input X(s) and output Y(s) Laplace Transforms. This function is expressed by numerator (N) and denominator (D) polynomials in the *Laplace* Domain (s-Domain):

$$H(s) = \frac{Y(s)}{X(s)} = \frac{N_n s^n + \cdots + N_1 s + N_0}{D_n s^n + \cdots + D_1 s + D_0} . \qquad (1)$$

From the stability point of view, it is known that continuous systems are asymptotically stable if and only if their poles are located in the left side of the s-Domain plane (continuous stability region). For frequency response analysis, "s" should be substituted by " jw ".

Considering that continuous systems can always be represented by discrete models and any continuous signal can be represented by sequences of numbers (since basic sampling and bandwidth rules are ensured, e.g. sampling theorem) [3,6], the two central questions on implementing discrete systems are: firstly, defining a discretization method for the corresponding continuous system and secondly, manipulating an input sequence $\{x[k]\}$ in order to obtain a desired output sequence $\{y[k]\}$, by means of discrete-time operators and difference equations. The use of these operators is very important when manipulating or analyzing discrete systems exactly because it defines the calculations and the input samples necessary to obtain each output value.

Next sections discuss the use of two different discrete-time methods, the Z-Transform and its associated Shift Operator (q) and the Delta-Transform and its associated Delta-Operator (δ). Moreover, their relation with the continuous systems and how the Delta-Form can be used to optimize digital filters will also be discussed.

III. Z-TRANSFORM AND SHIFT OPERATOR

Expression (2) defines the Z-Transform to any sequence $\{x[k]\}$ and (3) represents the canonical form of a discrete-time system H(z), obtained using any of the referred discretization processes:

$$X(z) = Z\{x[k]\} = \sum_{k=0}^{\infty} x[k] z^{-k} . \tag{2}$$

$$H(z) = \frac{Y[z]}{X[z]} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}} . \tag{3}$$

From (3) it is possible to deduce the relation between the input X(z) and output Y(z) of the Z-Transformed system. Then, the inverse Z-transformation of such relation yields the n^{th} order linear difference equation:

$$a_0 y[k] = b_0 x[k] + b_1 x[k-1] + b_2 x[k-2] + \cdots + b_n x[k-n]$$
$$- a_1 y[k-1] - a_2 y[k-2] - \cdots - a_n y[k-n] . \tag{4}$$

Comparing expressions (3) and (4), one observes that the multiplication factor z^{-1} in the z-Domain corresponds to a shift in the corresponding sequence, that is:

$$z^{-1} X(z) \leftrightarrow x[k-1] . \tag{5}$$

Then, it is usual to define a *shift operator* in a backward (causal) sense as in (6) and to associate it to the factor z^{-1} in the z-Domain.

$$q^{-1}(x[k]) = x[k-1] . \tag{6}$$

Using the above shift operator, expression (4) can be rewritten as

$$a_0 y[k] = b_0 x[k] + b_1 . q^{-1}(x[k]) + \cdots + b_n . q^{-n}(x[k])$$
$$- a_1 . q^{-1}(y[k]) - \cdots - a_n . q^{-n}(y[k]) . \tag{7}$$

Although the "q" operator and the variable "z" have distinct meanings, some authors usually formulate discrete systems substituting the "q" by "z" [3,4].

Expressions (3), (4) and (7) also represent the classical formulation of Infinite Impulse Response (IIR) systems, which lead to Finite Impulse Response (FIR) systems when $[a_i ,\ i \neq 0]$ are all zero. It is important to point out that a_0 is equal to 1 when no coefficient scale or normalization is necessary.

Similar to continuous systems, the complex variable "z" should be converted into e^{jwT_s} in order to obtain the frequency response of the discrete z-Domain systems. This term with $0 \le \omega \le 2\pi$, defines the unitary circle (UC) centered on the origin of the z-Domain plane, as the domain for the frequency response function. The UC is also the boundary region for the poles of stable discrete systems, since such systems must have their poles inside this circle.

As previously mentioned, the smaller the sample period is, the more critical is the stability of a system, because the poles and zeros are progressively shifted towards the position (1,0) of the z-plane [2,11]. Therefore, high sampling frequency clusters the relatively low frequency poles and zeros together, increasing the sensibility of the coefficients of the system.

Such clustering effect is especially severe in practical implementations of low frequency filters using finite word length or fixed-point DSPs. In such implementations a simple truncation, normalization or coefficient quantization, can take the system away from the desired characteristics and even take the system to complete unstable operating conditions, even for small variations in the poles position.

Section (V) will briefly discuss some traditional solutions to these problems, but in some conditions the discrete design is almost impracticable. Simulation and experimental results will illustrate such detrimental effects.

Conversely, the application of the Delta Operator and its associated transform seem to be a simple and direct methodology to avoid such problems on discrete-time systems. It makes possible the pre-optimization of the system's coefficients using a single parameter, improving the numerical representation and rounding sensibility of the digital design, as will be summarized and illustrated in next sections.

IV. DELTA-TRANSFORM AND DELTA OPERATOR

As presented in [2,4], the forward (anti-causal) delta operator is based on the following differentiation:

$$\delta(x[k]) = \frac{x[k+1] - x[k]}{\Delta} , \tag{8}$$

and can be related with the forward shift operator as $q = \delta . \Delta + 1$, where Δ could be theoretically associated to the sample time, but practically, it is a free optimization parameter when designing the digital coefficients. Note that as Δ goes to zero, δ represents the continuous differentiation operator *d/dt* or the *Laplace* variable "s".

Considering that the same relation is valid for the complex variables ($z = \gamma . \Delta + 1$), where γ represents the *Delta* or *γ-Domain*, and based on (2) it is possible to define the *Delta-Transform* as following:

$$D\{x[k]\} = \Delta \{X(z)\}\big|_{z=\gamma . \Delta + 1} =$$
$$X_d(\gamma) = \Delta \sum_{k=0}^{\infty} x[k] . (1+\gamma \Delta)^{-k} . \tag{9}$$

Likewise (3), the canonical form of discrete δ-Domain systems is expressed by:

$$H(\gamma) = \frac{Y[\gamma]}{X[\gamma]} = \frac{b_0 + b_1 \gamma^{-1} + b_2 \gamma^{-2} + \cdots + b_n \gamma^{-n}}{a_0 + a_1 \gamma^{-1} + a_2 \gamma^{-2} + \cdots + a_n \gamma^{-n}} \tag{10}$$

and from the inverse delta transformation, it can be represented by the linear difference equation:

$$a_0 y[k] = b_0 x[k] + b_1 . d^{\ 1}(x[k]) + \cdots + b_n . d^{\ n}(x[k])$$
$$a_1 . d^{\ 1}(y[k]) \quad \cdots \quad a_n . d^{\ n}(y[k]) \quad . \qquad (11)$$

Thus, if the forward delta operator represents a discrete derivative function, the inverse causal (backward) delta operator $^{-1}$, for a sequence $\{x[k]\}$, can be defined using $q = d. \ +1$ and the first order *Euler* integration method as following:

$$\begin{aligned} & d^{\ 1}(x[k]) = o[k] \\ & o[k] = \quad x[k \quad 1] + o[k \quad 1] \end{aligned} \quad , \qquad (12)$$

In the same way that the time shift operator "q" should not be interpreted as the z-Transform complex variable "z", " " is the equivalent complex variable to the -Domain and should not be replaced by " ".

Using (12), one can evaluate the discrete time difference equation (11) by means of input and output sequence's samples in a different way that in (4). To illustrate this possibility, a second order delta difference equation is presented in the next section.

Considering the relation between the discrete z and domains as expressed in (13), it is possible to show that for the frequency response and stability analysis, the stability region for the -Domain is limited by a circle of radius (1/), centered on (-1/), as detailed in [2,4]. On the limit, considering very small sampling periods (high sample frequencies), such stability region converges to a circle with a very large radius on the left side of delta domain, exactly as in the s-Domain.

This is quite different from the z-Domain stability region. Thus, (13) and (14) are quite important when analyzing the relations between the discrete variables by means of the forward and backward equations, respectively.

$$d = \frac{q \quad 1}{} \qquad \text{or} \qquad g = \frac{z \quad 1}{} . \qquad (13)$$

$$d^{\ 1} = \frac{q^{\ 1}}{1 \quad q^{\ 1}} \qquad \text{or} \qquad g^{\ 1} = \frac{z^{\ 1}}{1 \quad z^{\ 1}} . \qquad (14)$$

The importance of converting to z domain and vice-versa [2] is based on the possibility of manipulating discrete systems in order to obtain better results using -Domain concepts, or even the possibility of analyzing designed -Domain systems, by means of the usual z-Domain analytical and computational tools.

Besides the connection between these two discrete domains, a -Domain system could even be designed directly from a continuous time original system. This is possible, e.g., using a modified bilinear transformation, as described in [2].

V. FIXED-POINT OR FINITE WORD LENGTH IMPLEMENTATIONS

Considering the problems related to fixed-point and finite word length implementations [5,6] and the availability of high level floating point DSPs, microprocessors or compilers, one could ask *why not use just these high level technologies*? But, although there are several types of modern DSP using floating-point and high-precision arithmetic, the hardware and software complexity in their assembly represent additional costs and a considerable increasing in their chip or code dimensions when compared to the conventional fixed-point structures. Thus, next sections will review the problems and usual solutions about fixed-point implementations and the possible advantages related to -Domain designs.

A. Quantization Effects and Usual Solutions

The most relevant problems with fixed-point or limited precision floating point implementations are basically related to the quantization of coefficients and internal variables (result registers). To limit these constants and variables with a determined bit size, one usually uses some procedure to avoid overflows (internal saturation, rounding, normalizations and truncations) and make a specific designed discrete system feasible.

Traditionally, there are several different methods to solve or at least to reduce these problems [5,6]. The most significant are based on:

- breaking nth order systems into a composition of first and second order sections;
- choosing the better direct or transposed form to the discrete system realization (DFI,DFII,DFIt,DFIIt);
- selecting the appropriate cascade or parallel structure to implement high order systems.

Nevertheless, if high sampling rates are required, even using such solutions it is quite difficult or sometimes impossible to realize some precise z-Domain discrete systems, with limited bit size. Then, sophisticated solutions based on multi-dimension coefficients optimization are required.

In general, when none of these solutions are possible or efficient enough, one usually develops an equivalent FIR system to represent the desired digital system. The drawback on doing that is the significant increasing on the number of coefficients (order) in the corresponding digital system.

Assuming second order sections of IIR discrete systems, the following section illustrates how to implement a -Domain realization, capable of optimizing the coefficients in order to achieve a stable discrete system even if reduced bit size is used.

B. g-Domain Realization

Although the numerator (b) and denominator (a) coefficients could be calculated directly from the continuous system, Table I demonstrates how they can be achieved from the discrete z-Domain coefficients [8].

Then, the delta coefficients can be optimized as a function of aiming to ensure lower quantization or roundoff noise effects or also to define the maximum internal DSP variables size of the digital filter. This is achieved because the spread of the delta coefficients is controlled by .

Table I. Relationships between the shift and delta coefficients on a second order section.

$_0 =$	b_0	$_0 =$	a_0
$_1 =$	$\dfrac{2b_0 + b_1}{}$	$_1 =$	$\dfrac{2a_0 + a_1}{}$
$_2 =$	$\dfrac{b_0 + b_1 + b_2}{2}$	$_2 =$	$\dfrac{a_0 + a_1 + a_2}{2}$

Conversely, the spread of the shift coefficients is fixed and dependent of the sampling frequency and according to previous discussion, when finite precision and high sampling frequencies are used, these values could lead the filter to the unstable region on the discrete z-Domain.

From (11), a second order digital IIR filter can be implemented using the delta operator as follows:

$$\begin{aligned} a_0 y[k] = b_0 x[k] + b_1 .d^{\ 1}(x[k]) + b_2 .d^{\ 2}(x[k]) \\ a_1 .d^{\ 1}(y[k]) \quad a_2 .d^{\ 2}(y[k]) \end{aligned} . \quad (15)$$

Using Transposed Direct Form II (DFIIt) realization, as depicted in Fig. 01 (not considering the S and Q blocks) and the causal delta operator as defined in (12), the delta filter could be implemented using the intermediate input (i_1 and i_2) and output (o_1 and o_2) variables, according to the following steps:

Rearranging expression (15) arises:

$$\begin{aligned} a_0 y[k] = b_0 x[k] + d^{\ 1}(b_1 x[k] \quad a_1 y[k]) \\ + d^{\ 2}(b_2 x[k] \quad a_2 y[k]) \end{aligned} . \quad (16)$$

If, $i_2[k] = (b_2 x[k] \quad a_2 y[k])$ and $o_2[k] = d^{\ 1}(i_2[k])$, then:

$i_1[k] = (b_1 x[k] \quad a_1 y[k] + o_2[k])$ and $o_1[k] = d^{\ 1}(i_1[k])$.

This demonstrates that it is necessary to calculate o_2 first and then o_1, as follows:

$$o_2[k] = o_2[k \quad 1] + \ .(b_2 x[k \quad 1] \quad a_2 y[k \quad 1]),$$

$$o_1[k] = o_1[k \quad 1] + \ .(b_1 x[k \quad 1] \quad a_1 y[k \quad 1] + o_2[k \quad 1]).$$

Thus, the output of a second order delta digital filter can be represented by (17).

$$y[k] = (b_0 x[k] + o_1[k]) / a_0 . \quad (17)$$

In order to improve and turn simpler the delta filter implementation, it is rather important that and $_0$ be a power of two, in such a way that multiplications and divisions by these values can be implemented as simple right/left shifts in the DSP's arithmetic.

Note that the delta form filter realization represents a slight increase in the computational complexity (number of multiplications and additions), when compared to the shift form (4). However, it is much faster than usual solutions using FIR filters.

The "Q" block in Fig.01 represents all the quantization effects due to coefficients truncation, rounding of the multiplication and summation result registers, etc, while the

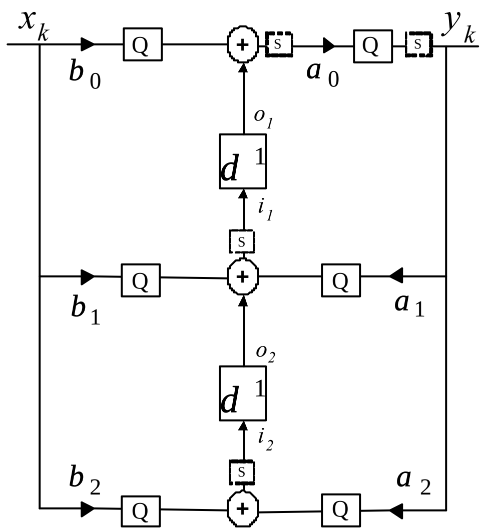


Fig. 01 – Transposed Direct Form II Digital Filter Structure.

"S" block represents additional saturation operation, implemented e.g., in anti-windup routines.

Next sections present simulation and experimental results of usual digital filters implementation using delta operators based on Direct Form II realization and fixed-point DSPs.

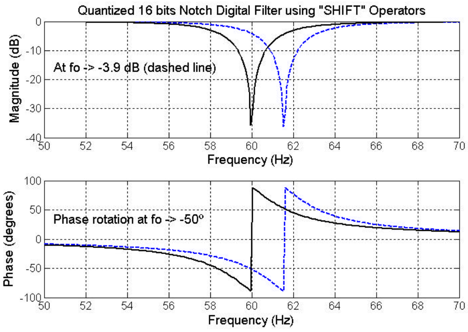
VI. SIMULATION OF NARROWBAND DIGITAL FILTERS

There are several different applications requiring some kind of digital filter in power electronics or power systems areas, such as active power filtering, motor controlling, ups (uninterruptible power supplies), power quality monitoring and analysis, etc. Some of these applications demand switching frequency ripple elimination, harmonics elimination, fundamental wave identification, feedback signals filtering, sinusoidal control references generation, etc.

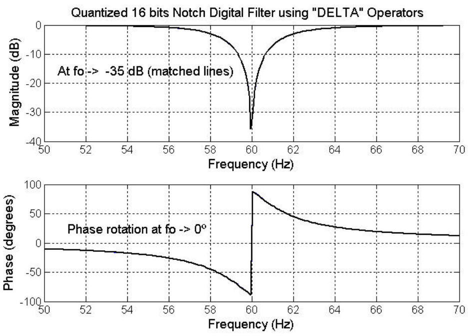
Probably, the most useful structures are band-pass, band-stop and low-pass digital filters. However, the problems discussed before have significant effects especially in narrowband filters. Thus, the following simulations consider the hard task on implementing second order digital band-stop (notch) and band-pass filters, with stopping or passing-band set to 2Hz and central frequency (fo) at 60Hz. The sampling frequency is set to 12kHz.

DSP Matlab toolbox was used in order to emulate the filter behavior when implemented in a fixed-point digital processor. Such functions allow simulating almost the exact fixed-point arithmetic, which means, coefficients normalization, truncation and roundoff, according a defined DSP bit size and also the length of internal variables (multiplications and summations result registers). The definitions were based on the ADMC401 16 bits DSP, from Analog Devices.

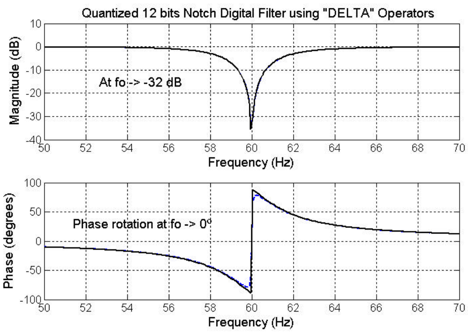
Fig.02 shows that the frequency response of the 16 bits quantized notch filter, implemented using shift operators (Fig.02a), is unstable or inefficient (dashed line) when compared to its continuous reference (solid line). The attenuation should be around 35dB instead of 3.9dB, while the phase shift is almost -50º and should be zero.



(a) - quantized "16 bits" filter using *shift* operator.



(b) - quantized "16 bits" filter using *delta* operator.



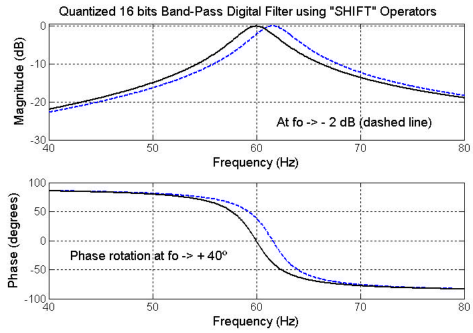
(c) - quantized "12 bits" filter using *delta* operator.

Fig. 02 – Band-stop digital filter: frequency response.

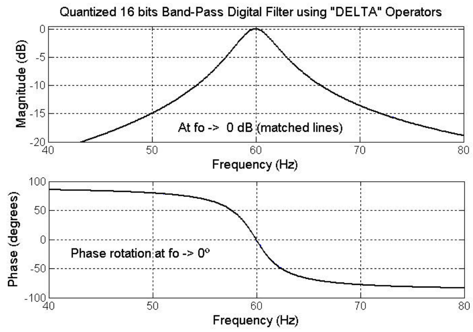
On the other hand, the quantized filter using delta operators (Fig.02b) exactly matches the frequency response of the continuous filter. Indeed, even reducing the bit size to 12 bits, the filter presents attenuation of 32dB, as shown in Fig.02c.

The ideal frequency response of a digital band-pass filter, with the mentioned features, is depicted in Fig.03a (solid line). Its 16 bits quantized band-pass filter, using shift operators, is unstable or inefficient as shown by the dashed line. However, the same 16 bits filter implemented in delta form has exactly the same response of the ideal filter, as shown in Fig.03b.

In all the simulations, the parameter was adjusted to (1/32) in order to optimize the filter responses and modify the spread of the delta coefficients.



(a) - quantized "16 bits" filter using *shift* operator.



(b) - quantized "16 bits" filter using *delta* operator.

Fig. 03 – Band-pass digital filter: frequency response.

VII. EXPERIMENTAL RESULTS USING A FIXED-POINT DSP

In order to validate the previous simulations, next figures present experimental results of such filters using a 16 bits fixed-point DSP (ADMC401).

An interesting application of notch filters in power system's quality analysis is the band-pass "algorithm" present in [12] and depicted in Fig.04. It should be responsible for the fundamental voltage waveform identification, but if implemented in a fixed-point DSP, using shift operators, it could be completely inefficient.

Fig.05 presents the algorithm input ($v_{in}[k]$ - quasi-square wave), output ($v_1[k]$ – sinusoidal wave) and residual (harmonic) signal. As foreseen in Fig.02a, the second order notch filter does not effectively attenuate the 60Hz (which means that the harmonic signal contains a significant part of fundamental wave), besides rotating its phase angle. The consequence is that the band-pass algorithm output (sinusoid) is out of phase and attenuated, relative to the theoretical fundamental component of the quasi-square wave.

A digital band-pass "filter" with the same features of the simulation in Fig.03 was also implemented using shift and delta operators. Fig. 06 shows that the second order shift filter realization is worthless, since the attenuation and phase are absolutely out of the designed points. Besides, it is a very hard task to implement a higher order band-pass digital filter without major instabilities.

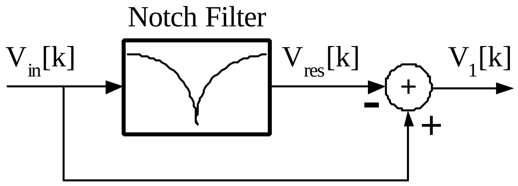


Fig. 04 – Band-pass algorithm based on notch filter.

On the other hand, Fig. 07 shows a fourth order band-pass digital filter implementation in delta form (cascade structure). Since the delta realization and the correct choice of parameter (1/16 in this case) improve the quantization sensibility, all truncation, rounding and saturation effects will not affect significantly the filter response.

Thus, using the same square wave input, it is possible to evaluate accurately its fundamental 60Hz wave, with correct amplitude and phase angle. By simple subtraction, one could even extract the residual or harmonic signal, to be used in several power systems or power electronics application, such as active power filtering [10] or THD estimation [12].

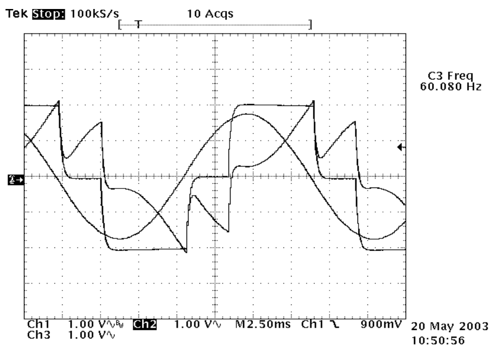


Fig. 05 – 16 bits band-pass "algorithm" input, output and residual signal, using *shift* operator.

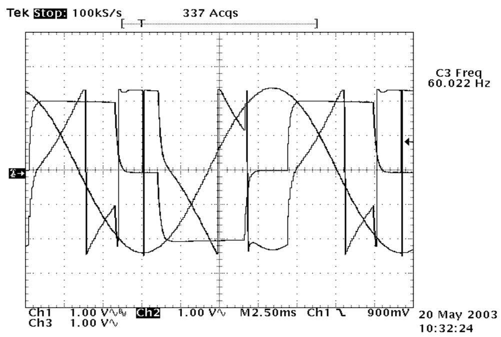


Fig. 06 – 16 bits band-pass digital filter using *shift* operator.

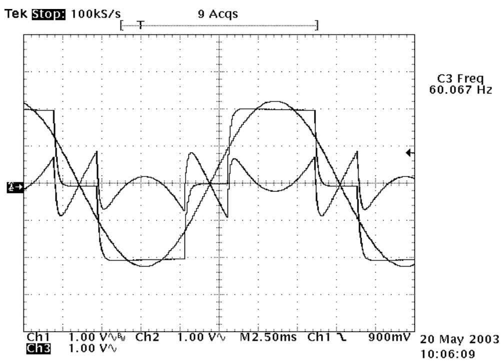


Fig. 07 – 16 bits band-pass digital filter using *delta* operator.

Finally, experimental results using a fixed-point 16 bits DSP have demonstrated that if delta form is used, even hard tasks such as very narrowband filters, can be implemented yielding good results. On the other hand, if shift form is applied, the results are absolutely unacceptable.

VIII. CONCLUSIONS

This paper discusses the problems on implementing digital systems when high sampling frequencies and fixed-point DSP are required. The unusual Delta *Operator* was summarized and used in order to optimize the discrete coefficients in such a way to improve the system's sensibility to quantization effects. Simulations results have shown that the *delta* filter performance is significantly superior to the conventional *shift* realization.

ACKNOWLEDGMENTS

Support for this research comes from FAPESP, CAPES and Analog Devices. The authors thank Mr. M. Newman for the e-mail discussion about delta operator filters.

REFERENCES

[1] R.H. Middleton, G.C. Goodwin, "Improved finite word length characteristics in digital control using delta operators, *IEEE Transaction on Automatic Control,* Vol. 31, No. 11, November 1986, pp.1015-1021.

[2] R.H. Middleton, G.C. Goodwin, *Digital Control and Estimation*, Prentice Hall Inc, 1990.

[3] K.J. Aström, B. Wittenmark, *Computer-Controlled Systems: Theory and Design*, Prentice Hall Inc., 1997.

[4] A. Feuer, G.C. Goodwin, *Sampling in Digital Signal Processing and Control*, Systems & Control: Foundations & Applications, Birkhäuser, 1996.

[5] T.W. Parks, C.S. Burrus, *Digital Filter Design*, John Wiley & Sons Inc, 1987.

[6] A.V. Oppenheim, R.W. Schafer, J.R. Buck, *Discrete-Time Signal Processing*, Prentice Hall Inc, 1999.

[7] M. J. Newman, D. G. Holmes, "Delta Operator Digital Filters for High Performance Inverter Applications", *IEEE Transaction on Power Electronics,* Vol. 18, No. 1, January 2003, pp. 447-454.

[8] J. Kauraniemi, T.I. Laakso, I. Hartimo, S.J. Ovaska, "Delta Operator Realizations of Direct-Form IIR Filters", *IEEE Transaction on Circuits and Systems-II: Analog and Digital Signal Processing,* Vol. 45, No. 1, January 1998, pp. 41-52.

[9] L.A.S. Ribeiro, C.B. Jacobina, A.M.N. Lima, "Estimação da Velocidade da Máquina Assíncrona em Acionamentos Estáticos ParteII: Estimação com Modelo corrente-Tensão", *SOBRAEP Journal*, Vol. 04, No.1, September 1999, pp.60-66, (*portuguese*).

[10] M. J. Newman, D.N. Zmood, D. G. Holmes, "Stationary Frame Harmonic Reference Generation for Active Filter System", *IEEE Transaction on Industry Applications,* Vol. 38, No. 6, November/December 2002, pp. 1591-1599.

[11] A. Khodabakhshian, V.J. Gosbell, F. Coowar, "Discretization of Power System Transfer Function", *IEEE Transaction on Power Systems,* Vol. 9, No. 1, February 1994, pp. 255-261.

[12] S.M. Deckmann and F.P. Marafão, "Time based decompositions of Voltage, Current and Power Functions," in *Proc. 2000 IEEE International Conference on Harmonics and Quality of Power*, pp. 289-294.