

Equalizers of the FUL DFE(N,M) Family on the NeuroMatrix NM6403 Processor

V.G. Osipov

RC "Module", Box: 166, Moscow, 125190, Russia

Phone +7(095)152-9335, Fax: +7(095)152-4661,

E-mail: vosip@module.ru WWW: http://www.module.ru

ABSTRACT

A large number of different devices aimed for communication link amplitude-frequency characteristics grading are studied and created at present. Such devices are called Equalizers. Below you can find an example of such a device known as FUL DFE(2,4) according to the adopted classification and used in cellular telephony as one of the variant for the GSM standard. Also its operating modes are considered. Then NM6403 processor calculation algorithms realizations will be considered and peculiarities connected with generation of similar structures of higher order if applied to NM6403 will be obtained.

INTRODUCTION

This article considers the application of NM6403 [5] processor supporting matrix operations as of the adaptive equalizer. Here all the Equalizer operating modes are considered. You can find an example of generating the matrix corresponding to FUL DFE(2,4) for NM6403 and principals of analogous matrixes forming for equalizers of the higher order used in wire and wireless modems.

1. BRIEF DESCRIPTION OF STRUCTURES OF THE FUL DFE(N,M) TYPE

The device corresponding to the FUL DFE(N,M) classification is a circuit consisting of two transversal filters showed in Fig. 1.1 - direct and reverse. The direct filter consists of "N" triggers and the reverse one - of "M" triggers. To describe the direct filter M1 coefficients are needed and to describe the reverse filter - M2 coefficients. Here the following expressions take place:

$$M1=N+1, M2=M$$

1.1

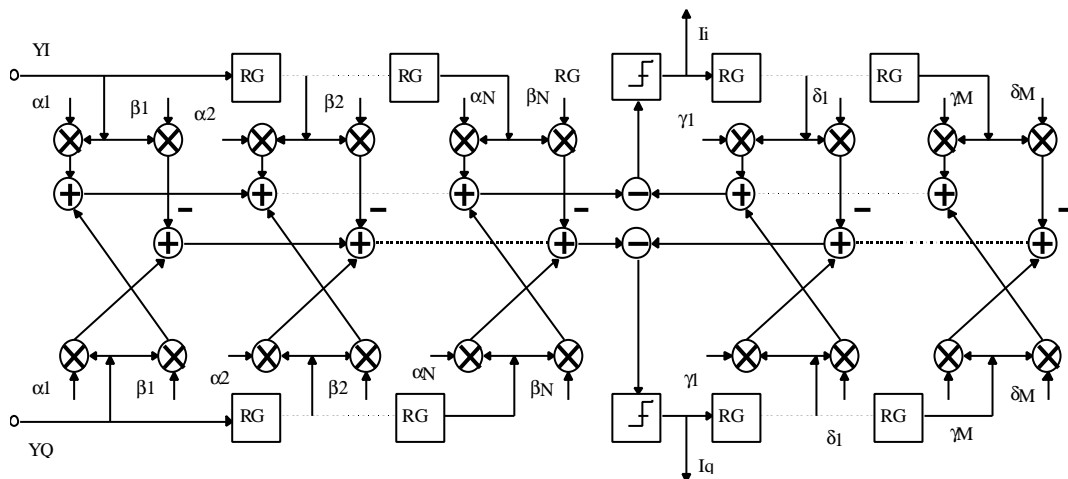


Fig 1.1

Where RG is a P-bit register, YI, YQ are inputs of the 1-st and the 2-nd links, Iq, Ii are symbol outputs, $\alpha_1+i\beta_1, \dots, \alpha_{N+i\beta_N}$, $\gamma_1+i\delta_1, \dots, \gamma_{M+i\delta_M}$ are coefficients of the direct and reverse transversal filters correspondingly.

The direct and reverse filters are included into the summator whose output is given to the detector. The signal from the detector is given to the reverse filter input and thus closing the feedback. In the simplest case detector is a threshold device forming the digital signal levels according to the input analog signal sign. Let's designate the analog signal at the detector input as "I1" and the digital signal at the output as "I2". We'll designate the equalizer input signal with the help of the variable "o", and filters coefficients - with variables α for direct filter and β for reverse filter. Equation 1.2. explains the scheme outlined above:

$$I1 = \sum_{j=1}^{M1} a_j x_k - \sum_{j=1}^{M2} b_j I2_{k-d-j} \quad 1.2$$

Where index "d" defines the delay relatively to the k-th count of the input signal. Here the inequality 1.3. takes place for the index "d"

$$0 < d < M1 + M2 \quad 1.3$$

2. EQUALIZER OPERATING MODES

The structure given above can operate in several modes: training 2.1; direct filtering 2.2 and adaptive filtering 2.3. In practice it is sensible to combine these operating modes. The following combinations are interesting: first 2.1, then 2.3 or 2.1 and then 2.2. These combinations can form sequences and be combined in any order. Input signal direct filtering is made with constant coefficients of direct and reverse filters and with closed loop of feedback in the detector chain.

2.1 To train the equalizer the feedback in the detector chain is opened in the structure described above and known prototype sequence is given to the input of the reverse transversal filter. The same sequence is received in the communication link and goes to the equalizer input, here the inequality 1.3. must be satisfied. Coefficients of the two filters of equalizer change with every new input signal count according to the selected adaptive algorithm [3]. We'll consider only the "Steepest descent" method. Common "Steepest descant" method is represented by the following iterative process: $\tilde{N}_k = C_{k-1} - \Delta \tilde{A}_k$, $\tilde{A}_k = -\varepsilon_k X_{k,t}^*$ (2.2.1) where error signal is: $\varepsilon_k = I_k - C_{k-1,t} X_k$ (2.2.2) $\tilde{N}_k = (\tilde{n}_1, \tilde{c}_2 \dots \tilde{c}_N, \tilde{b}_1, \tilde{b}_2 \dots \tilde{b}_M)$ generalized filters coefficients $X_{k,t} = (x_k, \dots, x_{k+N}, I_{k-1}, \dots, I_{k-M})$ generalized signal in filters \tilde{a} , $X_{k,t}^*$ complexly conjugated with $X_{k,t}$. Equalizer training requires to apply prototype sequences. For example, in the GSM standard there are eight such sequences with the twenty six bit length. The result of the training process will be slightly changing filters coefficients and if the condition (1.3) is satisfied, then the k-th symbol position with its delay "d" will be accurately stated. As the "Steepest descent" method gives the convergence from one hundred iterations and the input signal necessary for training has the length equal to the prototype, it is stored and cycled together with the prototype one for the necessary number of times to define the generalized coefficients.

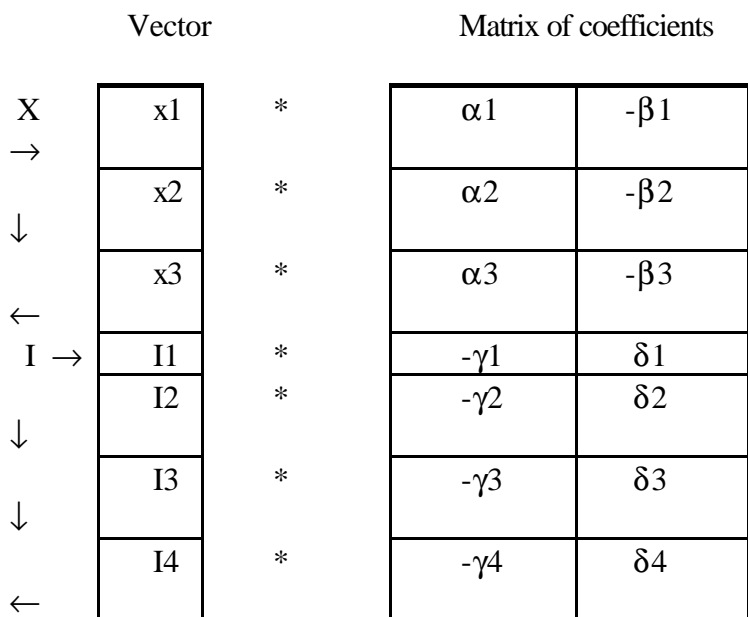
2.2 Direct filtering is shown in Fig. 1.1, where the filters coefficients are constant.

2.3 Adaptive filtering takes place with filters coefficients changing according to the "Tracking Algorithm" method [3]. In this case the feedback in the detector chain is not opened. Since in this case the equalizer input signal serves as prototype, the number of errors referred to the received bits should be about two percent. It is

possible to mind this value in two ways: the essence of the first way is in coincidence of the received and training sequences and of the second one - in a small correction in the filters coefficients for several tens of iterations. If the error flow of errors and received bits does not accrue too fast and if its value is limited by the condition given above then the equalizer is itself fitted with the data flow to the changes of the communication link existence environment decreasing the number of bits incorrectly received due to frequency-domain distortions. One should also take into account the noise in the link which distorts the received bits of the signal. This disadvantage can be corrected by a more complex detector detecting several symbols. We'll not consider such a detector here. However this circumstance adds consideration of a new parameter which allows to combine the equalizer operating modes more effectively. When the noise in the equalizer is big, then it expedient first to train the equalizer and then to use the simple filtering mode in case of slowly changing communication link or adaptive filtering for the link changing during one time slot. In case the communication link is good, it is possible to train the device using one cycle of training sequence with following direct filtering.

3. PECULIARITIES OF NM6403 USAGE

Peculiarities of NM6403 usage are connected with fast execution of matrix operations, such as matrix multiplication by vector. Example of the FULDFE(2,4) equalizer will illustrate obtaining of the algorithm for NM6403 given below using such a matrix multiplication. Equalizer requires to multiply the input complex vector of the received signal with the number of elements - 14 by the matrix sized 14x2. Let's see how this matrix is organized. In NM6403 the software splitting of the principal matrix sized 64x64 into matrix of smaller size takes place and it is possible to work with vectors whose elements dimensionality is more or equal to two but is less or equal to 64. That's why it is convenient to the use the structure given below in Fig 3.1 where all elements "O" and "Y" are 8-bit and the elements "I", "Q" have the minimal possible word length equal to two bits. Fourteen-element vector with the word length indicated above fully occupies a 64-bit word in the processor memory. Minimal word length of elements of matrix column is selected according to the condition that sums of all products of the vector components by their corresponding numbers recorded in the column elements, are placed in the column element without its overfilling. The columns can be divided as 24+24 plus 16-bit remainder for eight-bit matrix coefficients. But since matrix is filled row-by-row it is more convenient to present it as two columns for 32-bit each.



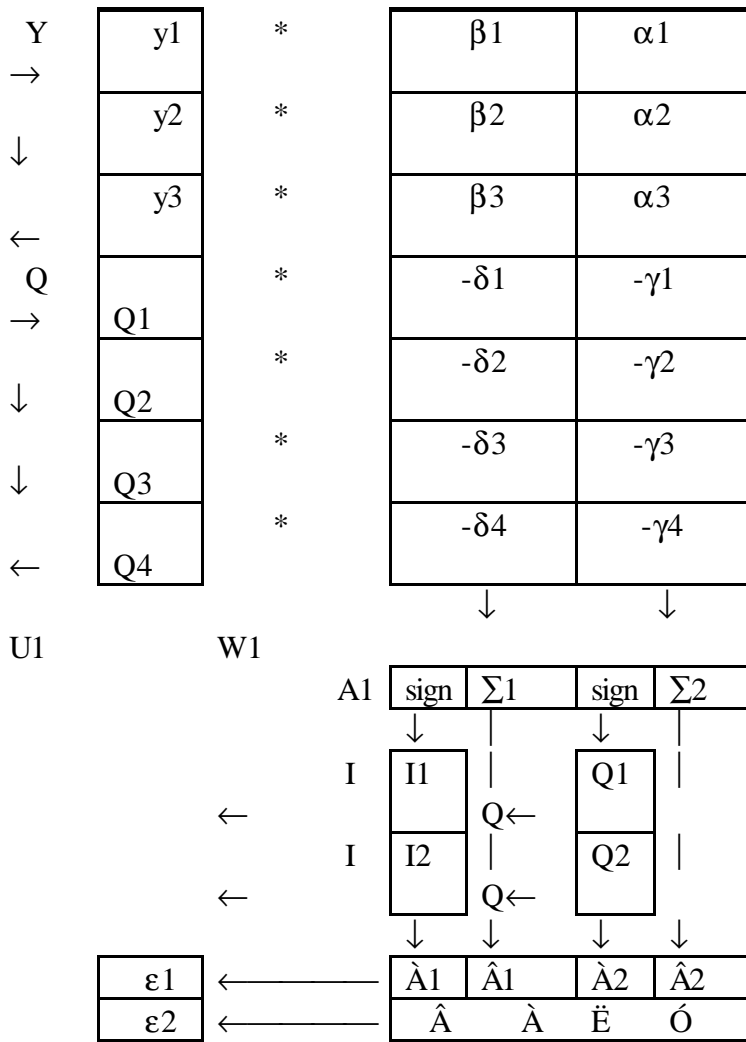


Fig 3.1

Fig. 3.1 shows registers I1,I2,Q1,Q2, $\epsilon_1,\epsilon_2,U1$, matrix of coefficients W1, vector accumulator A1 and vector ALU. Arrows in the Figure showing cyclic rearrangement of the vector $X_{k,t}$ elements explain the U1 register filling. Registers I1 and Q1 store the signs of the first and the second results of summation. Registers I2 and Q2 store the prototype training sequence values. Depending on the equalizer operating mode, the contents of the first or the second pair of registers goes to the ALU input. The same data take part in the U1 filling. Registers ϵ_1 and ϵ_2 store the intermediate calculation result necessary to correct the coefficients of W1 matrix. It is necessary in the 2.2. and 2.3. modes. Let's consider the 2.1 mode. In this mode the values of W1 matrix do not change and only U1 vector components are updated. In 2.2. mode the A1 accumulator contents is subtracted from the value of the training sequence prototype signal in the processor vector ALU. The result will be ϵ_1 and ϵ_2 whose sign will be used to form the U1 vector. W1 matrix is filled row-by row with updated coefficients according to the equation 2.2.1, where Δ is a constant ≈ 0.007 , providing the algorithm convergence. Multiplication by this constant is realized with the shift by seven binary bits. The process continues until the correction of the W1 coefficients is equal to zero. In the 2.3. mode the signs of numbers contained in A1 are used for vector U1 forming instead of the prototype sequence values. The scheme illustrating all the described states is given in Fig. 3.2.

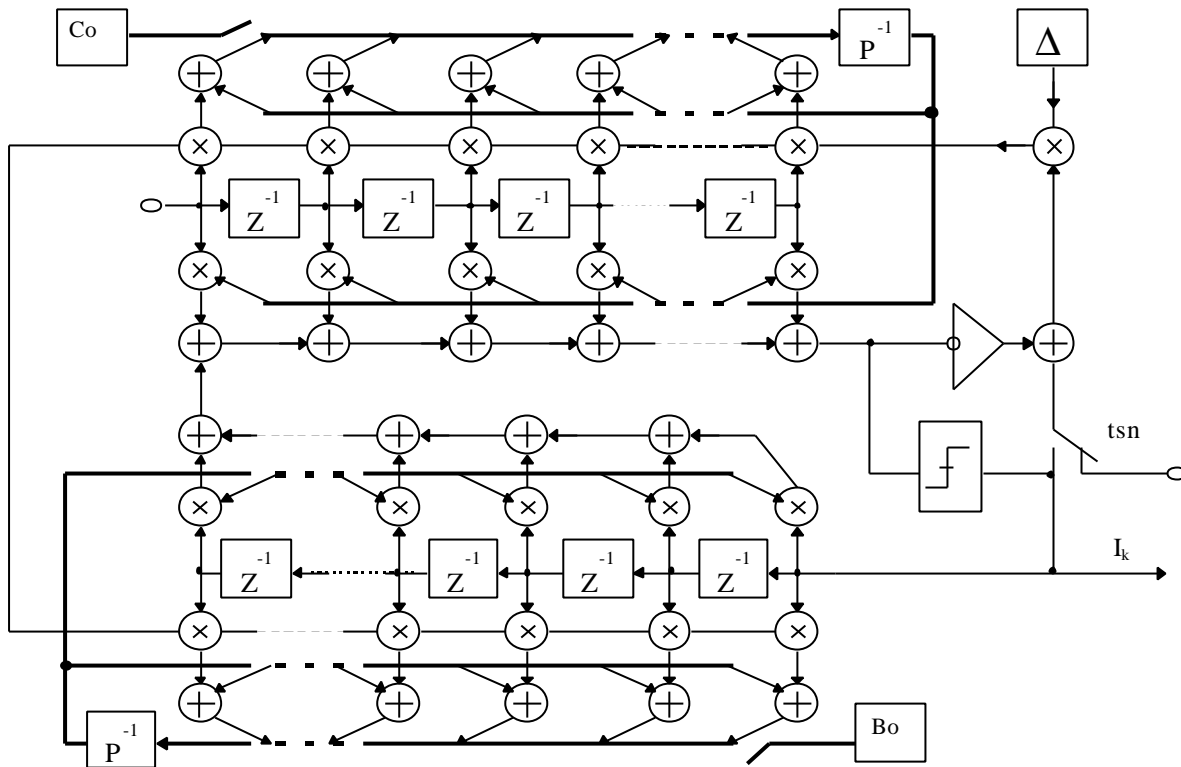


Fig 3.2

I_k - input sequence of symbols, t_{sn} - training sequence where $Z^{-1} = T$ delay connected with the symbols frequency and convolution factor β with the following relationships:

$$2F_{\max} = (1 + \beta) / T, \quad \beta \in (0, 1)$$

3.1

Below you can find graphs of average squared error and error bits of received signal in the link with the dispersion 3.4

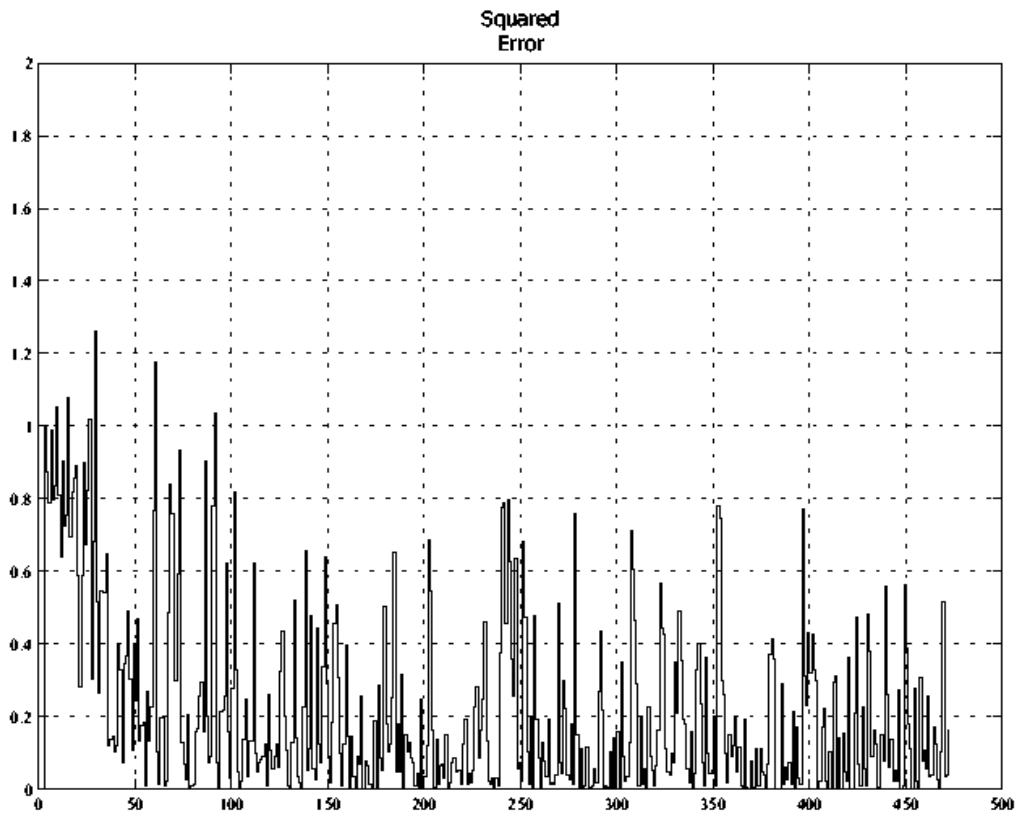


Fig 4.1

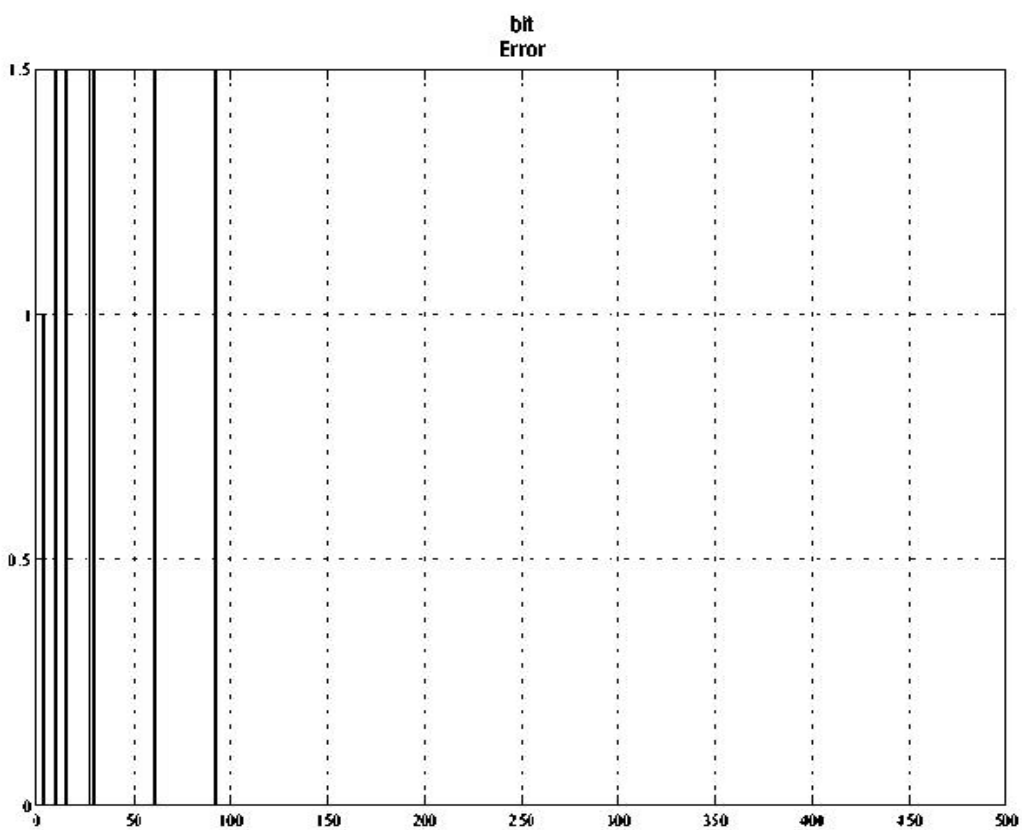


Fig 4.2

4. RESULTS OF MODELING

the algorithm given above has been simulated in the "MATLAB" system. The results are the following. Firstly, convergence takes place within one hundred iterations (Fig. 4.2.) for the links dispersion range from 1 to 3,5. Secondly, limitation of the input signal word length slows down the iteration process and limits the range of allowed dispersions suitable for correct work of such an Equalizer. However, it is possible to use the input signal lowering to increase the equalizer order without lowering the filters coefficients word length [4] in the above described structure. It allows to work with link dispersions which are larger by several times under the conditions of slowly changing link dispersion or under the condition of cellular telephony on the typical time of one time slot [1]. On the previous page you can see the graph of squared error (Fig.4.1) in the tests FUL DFE(2,4) under the condition of dispersion equal to 3,4 and noise which power is -6decibel. Under such conditions convergence takes place after eighteen iterations, and 40 iterations will be enough for the dispersion 2,5.

CONCLUSION

NM6403 spends one cycle per one iteration in the 2.1. mode. It takes it two clock rates per one iteration in the 2.2 mode. About one thousand two hundred cycles [2] are needed for training the FUL DFE(2,4) structure. It corresponds to the training sequence length for the GSM standard approximately equal to 26 **maxwell** at the processor operating clock rate of 40 MHz. Practical application of complex equalizers FUL DFE(5,4) and FUL DFE(4,6), realized at NM6403 with the filters coefficients word length not less than eight bits and input signal word length equal to four bits. These structures are boundary in the number of coefficients in the direct and reverse equalizer filters in the sense of maximal approach of N to M. Other realizations can be found from integer-number relationship 4.0, where "à" is the input signal word length.

$$a(N+1)+2M \leq 32 \quad 4.0$$

These devices operate with complex signals. Equalizers for real signal at NM6403 are obtained according to the expression 4.1 analogous to 4.0.

$$a(N+1)+2M \leq 64 \quad 4.1$$

It is interesting to underline that the transfer from the equalizer of one dimensionality to the equalizer of another dimensionality is performed by a suitable division of the NM6403 matrix, which is software realized.

INFERENCE

NM6403 allows to create highly effective flow filters of different order capable of outputting the result at the processor clock rate $\frac{1}{N}$ of the input clock rate.

BIBLIOGRAPHY

1. 1982 IEEE Anderson, et. ,reference 6-6.
2. Adaptive Equalization for TDMA Digital Mobile Radio, Jon G. Proakis, Fellow, IEEE.
3. Least Square Estimation with Applications to Digital Signal Processing, A. Giordano, M. Hsu.
4. IEEE transaction on VEHICULAR TECHNOLOGY /may 1991 volume 1 2.
5. NM6403 "Architectural Overview". <http://www.module.vympel.msk.ru>