Ultra Low Power Switched Current Finite Impulse Response Filter Banks Realized in CMOS 0.18 μm technology

Rafał Długosz[1,2,*]

1- University of Neuchâtel, Institute of Microtechnology
Rue A.-L. Breguet 2, CH-2000, Neuchâtel, Switzerland
2- University of Alberta, Department of Electrical and Computer Engineering
W2-079 ECERF Building, Edmonton, T6G 2V4, Canada
[*] fellow of the Marie Curie European Union Outgoing International Fellowship
rdlugosz@ualberta.ca

ABSTRACT

Ultra low power circuits are in high demand in many applications especially in wireless sensor networks (WSN), where energy is scavenged from environment. WSN systems contain various blocks, such as: sensors, filters, analog-to-digital converters, very often a simple processor and the RF front end. This paper is devoted to ultra low power finite impulse response (FIR) filters and filter banks implemented in a switched current technique. In this paper new SI FIR filter structures and filter banks have been proposed that operate in a current mode where no operational amplifiers are used. This enables ultra low power operation that is on the level of several μW. Proposed filters work under threshold level, for supply voltage varying in the range between 0.5 – 0.7 V. The simulated attenuation is limited to about 45 dB, what is due to different nonidealities, but this is usually sufficient in WSN applications. SI technique features many interesting mechanisms that simplify realization of analog filter banks. Samples in SI filters are copied from delay line to the filter coefficients. As a result, many sets of different filter coefficients can be connected to a single delay line without speed limitation, what is the problem in case of SC FIR filters. Ultra low power operation is possible also due to a special structure of the clock generator that consists only from switches and NOT gates.

Keywords: Switched Current FIR filter banks, WSN applications, discrete wavelet transform

1. INTRODUCTION

Filtration is one of the common tasks in electronic systems. Various filtration techniques have been described worldwide. Filters can be roughly classified into several groups. They are used, for example, in filtration of continuous or discrete time signals, in processing of analog or digital (quantized) signals. In discrete time domain, filters that are the most often used filters are finite impulse response (FIR) filters. These filters have many important advantages, such as: possibility to obtain the linear phase response and absolute stability. Although FIR filters usually are related to discrete time signals, there were also reported FIR filters with delay line operating in continuous time domain [4], where delay elements are realized as all-pass biquadrate active filters. FIR filters usually are implemented in DSP or FPGA systems for applications requiring high selectivity, but in many applications small dynamic range is sufficient, but other parameters, like for example power dissipation and high speed become the most important criterions. In such cases analog FIR filters are often the best solution due to very low power dissipation and simple structure. There exist different classes of analog FIR filters, but switched capacitor (SC) or switched current (SI) circuits are the most common [1 – 8].

This paper focuses on SI FIR filters and on SI FIR filters banks. Several structures of these filters have been earlier proposed by author in [9]. Taking into account other existing solutions [1, 2, 3], filters proposed in [9] feature smaller power dissipation, what is an important parameter in wireless sensor network (WSN) applications (power dissipation often is below 1 μW). Such power dissipation level is possible due to elimination of operational amplifiers (the main power consumers) and by implementation of ultra low power clock generators.

The general block diagram of the FIR filter is shown in Fig. 1. This circuit consists of a delay line (delay elements T), coefficients (h_i) and the summing circuit. The filter order N is equal to the number of filter taps minus one. When input

signal $x(t)$ is passing the filter, particular signal samples x_i are storing in the delay line. After multiplication by coefficients h_i they are summing at the output. The output samples y_i are calculated using the following equation:

$$y_n = \sum_{i=0}^{N} h_i x_i \tag{1}$$

In the current mode approach particular building blocks work in current mode, e.g. multiplication by coefficients is realized using current mirrors with gains that are proportional to the values of the coefficients. Summing of signals is realized in junctions, as signals in SI FIR filters are represented by currents.

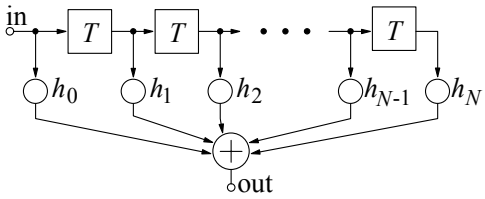

Figure 1. Block diagram of a finite impulse response (FIR) filter with three layers: delay line (delay elements T), filter coefficients (h_i), and the summing circuit

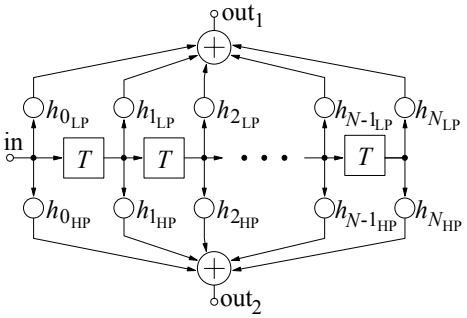

Figure 2. QMF bank of two FIR filters sharing one delay line

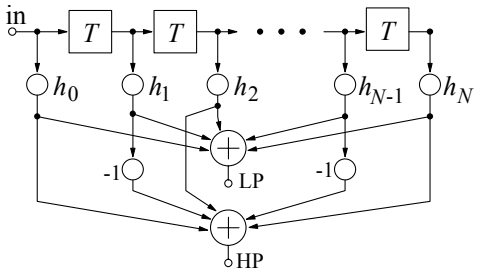

Figure 3. General idea of the QMF bank composed of filters with symmetrical transfer function

In many applications, such as: wavelet transformers, multirate and coding systems, filter banks are very common used. Filter banks are composed of at least two, but often of a bigger number of filters working in parallel and processing the same input signal. When FIR filters are used in filter banks, then one delay line is shared by all filters, what simplifies the hardware layer of the system. An example filter bank with two composing filters is shown in Fig. 2. Such structure implemented as a quadrature mirror filter (QMF) bank is a composition of two complementary filters i.e. the halfband low-pass filter with the transfer function LP(z) and the halfband high-pass filter whose transfer function HP(z) is the quadrature mirror image of LP(z) with respect to the normalized frequency equal to $\pi/2$. In practice, to get the high-pass filter one must change sign of every second coefficient of the lowpass filter's transfer and then reverse order of the coefficients. The second operation can be omitted in case of filters with symmetrical transfer function, i.e. with coefficients expressed as:

$$h_i = h_{N-i}, \text{ for: } i = 0, ..., N \quad (2)$$

The QMF bank of filters with the symmetrical transfer function can be very easy implemented in SI technique. In this case the same circuitry is used to multiply signal samples by coefficients and only one additional current mirror per filter tap is used to turn sign of the resulting product. The general idea of such filter bank is shown in Fig. 3.

The QMF bank composed of FIR filters can be used in subband coding and multiresolution analysis in a multistage discrete wavelet transform (DWT), shown schematically in Fig. 4 [10]. The calculation procedure in this structure starts with passing the input signal $x(n)$ through a lowpass filter (LP). In the nest step every second sample is being removed from the resulting signal according to the Nyquist's rule, using decimation circuit with the decimation factor of 2. The procedure is repeated on each level with sampling frequency that is two times smaller than on the previous levels. Number of levels depends on number of the input samples. When, for example, number of samples in the original signal is equal to 256, then decomposition process continues until two samples are left. In this case 8 leveles are required. Output samples from the highpass filters on each level are output DWT coefficients. Discrete wavelet transform briefly described here is very useful tool in image processing as enables reducing the image size without loosing much of the resolution, what is important from the image transferring point of view. In WSN applications such tool is paramount, as enable reduction of energy consumed per transmitted image.

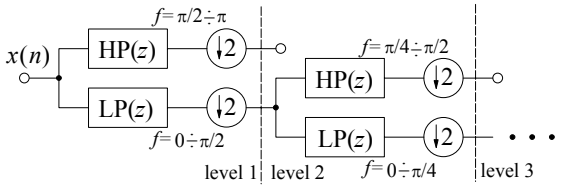

Figure 4. Wavelet transformer as binary tree-structured QMF filter bank

The paper is organized as follows. Section 2 presents existing SI FIR filters reported earlier as well as some new filter structures. SI FIR filter banks are discussed in the next section. Finally, the conclusions are drawn in section 4.

2. SI FIR FILTER STRUCTURES

Analog SI FIR filters are very efficient in realization of filter banks. In this section several SI FIR filters are presented to make the background for the SI FIR filter banks described in the next section.

The first example of SI FIR filters is the programmable delay line structure shown in Fig. 5 (a) [9]. This filter is controlled by a simple 2-phase clock generator what is the main advantage, as it significantly simplifies the circuit's structure. This filter suffers from limited accuracy, what is due to many data rewriting operations between delay elements. The resulting error depends on filter order and increases when filter length increases. As a result, application of this filter is limited to applications where relatively small number of taps is sufficient. Programmable filter coefficient used in this filter as well as in the next structures is shown in Fig. 5 (b) [9]. Advantage of this circuit is that those branches that are not used are completely turned off and do not dissipate power. The appropriate placement of configuration switches enables shortening of gates of transistors used as current sources to VSS, resulting in eliminating the leakage effect. This circuit enables also realization of negative coefficients, what enlarges area of potential applications of the filter. To implement the QMF filter bank composed of two filters with symmetrical transfer functions, as shown in Fig 3, the circuit shown in Fig. 5 (b) have been supplemented by an additional output stage, producing the second output signal. This circuit is shown in Fig 5 (c). Advantage of this solution is possibility to obtain two filters on the expense of only several additional transistors in each coefficient. This enables saving of both the power dissipation as well as chip area, which increase only moderately.

Another two SI FIR structures are shown in Fig 6 (a) and (b) [9]. These are the rotator and the circular memory structures respectively. Both structures use the same circular delay line. Advantage such a delay line is that there is no data rewriting between delay elements, as it was in the delay line structure, resulting in better filter accuracy. Each

sample remains on a fixed position, as long as after the N clock phases is replaced by a new sample. Difference between both filters lies in construction of the filter coefficients block. In rotator structure coefficients remain on fixed positions, while the rotating effect is realized in the rotator switch, which is kind of multiplexer that enables connection between each delay element and each coefficient. Disadvantage of this filter is more complicated clock generator, as half of all clock phases are used to control the rotator switch. In case of the circular memory filter rotating effect is realized by rotating of the coefficients in the structure, which after calculation of each sample change their position. Advantage of this structure is simpler structure, as each delay element have a fixed connection with a given coefficient circuit. As the rotator switch has been eliminated number of clock phases is reduced by half. Disadvantage is switching over in the digital memory. Typically this is the source of large power dissipation [1, 3], but in solutions presented in this paper the power dissipation is reduced by a special construction of the clock generator, what has been described in next section.

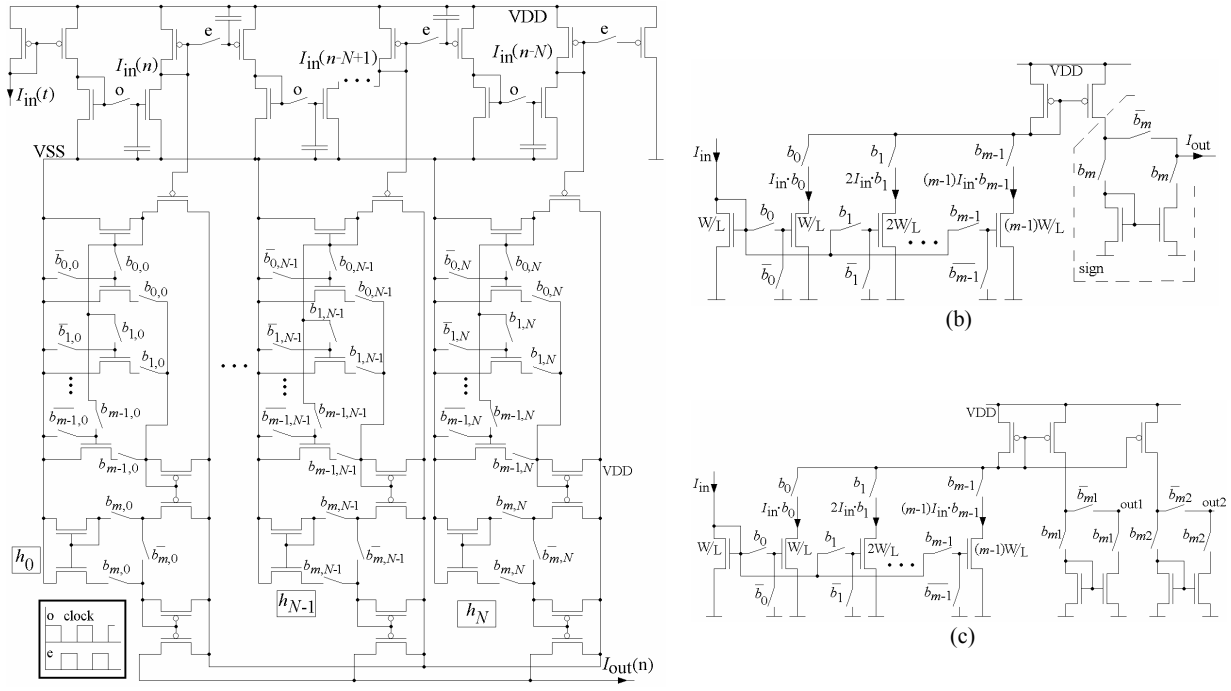

Figure 5. Programmable parallel delay line filter: (a) the general structure, (b) programmable coefficient used in a single filter mode, (c) programmable coefficient used in QMF bank of two filters with symmetrical transfer functions

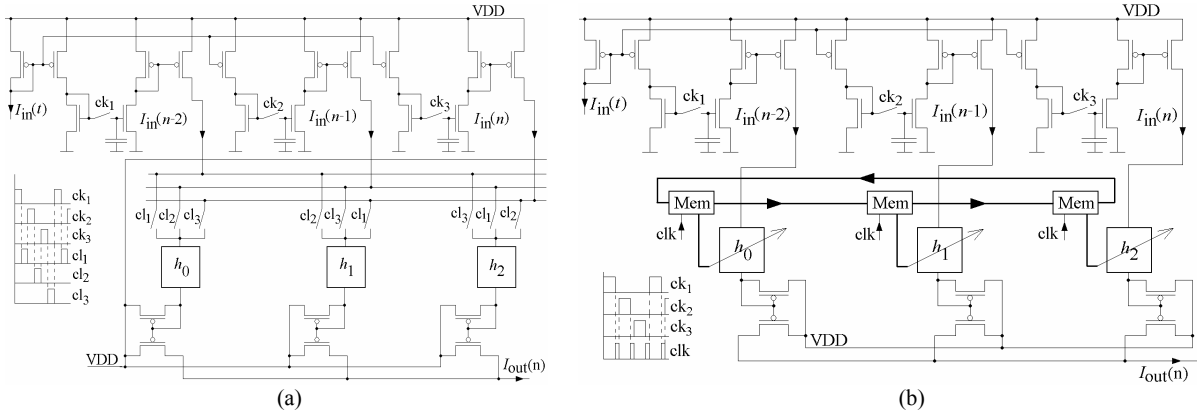

Figure 6. (a) Rotator SI FIR filter structure and (b) circular memory SI FIR filter structure

New SI FIR filter structure with reduced chip area

SI FIR filters described above are a direct implementation of the following equation:

$$y(n) = x(n)h_0 + x(n-1)h_1 + x(n-2)h_2 + \ldots + x(n-N+1)h_{N-1} + x(n-N)h_N \tag{3}$$

where: programmable filter coefficients are realized as sums of components that are products of numbers being powers of 2 and configuration bits *b*. Number of bits in each coefficient "*m*" is called the coefficient resolution. Each coefficient can be expressed as:

$$h_i = b_{i,m-1}2^{m-1} + b_{i,m-2}2^{m-2} + \ldots + b_{i,2}2^2 + b_{i,1}2 + b_{i,0} \tag{4}$$

There is also another way to realize the SI FIR filters. In this approach number of multiplications is reduced in exchange of bigger number of additions. First, let us combine equations (3) and (4). As a result, we get the following equation:

$$\begin{aligned} y(n) = {} & x(n)(b_{0,m-1}2^{m-1} + b_{0,m-2}2^{m-2} + \ldots + b_{0,2}2^2 + b_{0,1}2 + b_{0,0}) + \\ & x(n-1)(b_{1,m-1}2^{m-1} + b_{1,m-2}2^{m-2} + \ldots + b_{1,2}2^2 + b_{1,1}2 + b_{1,0}) + \\ & x(n-2)(b_{2,m-1}2^{m-1} + b_{2,m-2}2^{m-2} + \ldots + b_{2,2}2^2 + b_{2,1}2 + b_{2,0}) + \\ & \ldots \\ & x(n-N)(b_{N,m-1}2^{m-1} + b_{N,m-2}2^{m-2} + \ldots + b_{N,2}2^2 + b_{N,1}2 + b_{N,0}) \end{aligned} \tag{5}$$

that can be rewritten as:

$$\begin{aligned} y(n) = {} & 2^{m-1}\left(x(n)b_{0,m-1} + x(n-1)b_{1,m-1} + x(n-2)b_{2,m-1} + \ldots + x(n-N)b_{N,m-1}\right) + \\ & 2^{m-2}\left(x(n)b_{0,m-2} + x(n-1)b_{1,m-2} + x(n-2)b_{2,m-2} + \ldots + x(n-N)b_{N,m-2}\right) + \\ & + \ldots + \\ & 2^2\left(x(n)b_{0,2} + x(n-1)b_{1,2} + x(n-2)b_{2,2} + \ldots + x(n-N)b_{N,2}\right) + \\ & 2\left(x(n)b_{0,1} + x(n-1)b_{1,1} + x(n-2)b_{2,1} + \ldots + x(n-N)b_{N,1}\right) + \\ & \left(x(n)b_{0,0} + x(n-1)b_{1,0} + x(n-2)b_{2,0} + \ldots + x(n-N)b_{N,0}\right) \end{aligned} \tag{6}$$

In equation (6) first we calculate sums in each pair of brackets. These are sum of all input samples stored in delay line, weighted by particular bits *b*. In the next step these sums are multiplied by numbers that are powers of 2. Finally all products are summed again, producing the output sample at the filter output. Implementation of equation (6) in SI technique requires introduction of separate bits for the negative and the positive filter coefficients. As a result we get the following equation:

$$\begin{aligned} y(n) = {} & 2^{m-1}\left(\mathrm{sgn}|x(n)|(b_{0,m-1,1} - b_{0,m-1,2}) + \mathrm{sgn}|x(n-1)|(b_{1,m-1,1} - b_{1,m-1,2}) + \ldots + \mathrm{sgn}|x(n-N)|(b_{N,m-1,1} - b_{N,m-1,2})\right) + \\ & 2^{m-2}\left(\begin{array}{l} \mathrm{sgn}|x(n)|(b_{0,m-2,1} - b_{0,m-2,2}) + \\ + \mathrm{sgn}|x(n-1)|(b_{1,m-2,1} - b_{1,m-2,2}) + \ldots + \mathrm{sgn}|x(n-N)|(b_{N,m-2,1} - b_{N,m-2,2}) \end{array}\right) + \ldots + \\ & 2\left(\mathrm{sgn}|x(n)|(b_{0,1,1} - b_{0,1,2}) + \mathrm{sgn}|x(n-1)|(b_{1,1,1} - b_{1,1,2}) + \ldots + \mathrm{sgn}|x(n-N)|(b_{N,1,1} - b_{N,1,2})\right) + \\ & \left(\mathrm{sgn}|x(n)|(b_{0,0,1} - b_{0,0,2}) + \mathrm{sgn}|x(n-1)|(b_{1,0,1} - b_{1,0,2}) + \ldots + \mathrm{sgn}|x(n-N)|(b_{N,0,1} - b_{N,0,2})\right) \end{aligned} \tag{7}$$

Filter structure that is a direct implementation of equation (7) is shown in Fig. 7. This structure is an implementation of the programmable circular memory filter shown in Fig 6 (b). The control bits *b*, shown in Fig 7, rotate in the digital memory that is organized into several rings. The main advantage of this filter is minimal number of multipliers used in

the circuit. Number of multipliers is always equal to the coefficients' resolution m, and does not depend on the filter order N. As multipliers are composed of large transistors, presented filter is an area efficient solution.

One of the important design aspects here is that additional currents denoted in Fig 7 as $\underline{I}_{\text{level}}$ must be delivered. Currents that flow to particular multiplying current mirrors are sums of both positive currents flowing from PMOS type mirrors and the negative currents flowing into the NMOS type mirrors. In case when, in a given branch, sum of all negative currents will be bigger than sum of positive currents, or both sums would have very similar values, then given output NMOS type mirror will be turned off. To avoid this situation the, DC bias current (I_{level}) must by added in order to polarize the output current mirrors.

The idea of digital memory that is used in proposed filter is shown in Fig. 8. In equation (7), each pair of brackets contains $2{\cdot}m$ configuration bits. During programming these bits are initially stored in D flip-flops and then rewritten to the ring memory that is further controlled by a 2-phase clock generator ("e" and "o"). This memory is realized using gate-to-source parasitic capacitances in NOT gates. This solution is more power efficient than, for example, memory realized using only DFFs. Number of rings is equal to coefficients' resolution m, as each ring delivers also negated counterparts of given configuration bits.

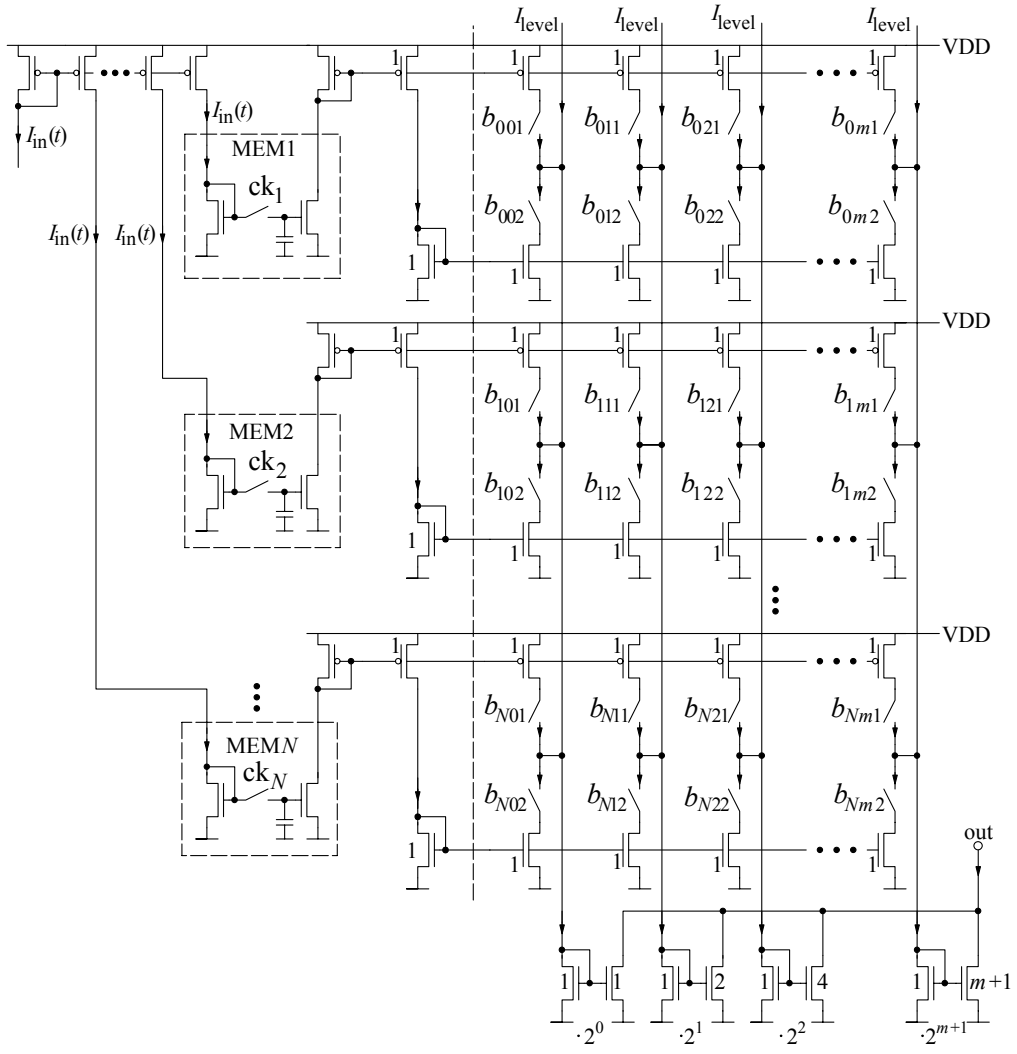

Figure 7. Proposed SI FIR circular memory filter

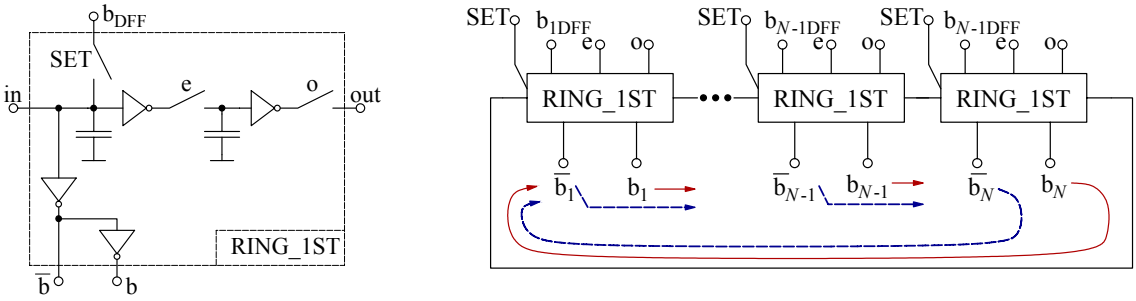

Figure 8. Ring digital memory, controlled by a 2-phases clock generator used in circular memory structure

To illustrate the principle of proposed filter let us consider the following example of 5th order filter with flat frequency response:

$$H(z) = z^0 + 5z^{-1} + 10z^{-2} + 10z^{-3} + 5z^{-4} + z^{-5} \tag{8}$$

Particular coefficients in this function in binary notation can be expressed as: {0001, 0101, 1010, 1010, 0101, 0001}. The resulting map of connections in the filter realized using structure shown in Fig 7 is shown in Fig. 9. Number of multipliers has been reduced to 4.

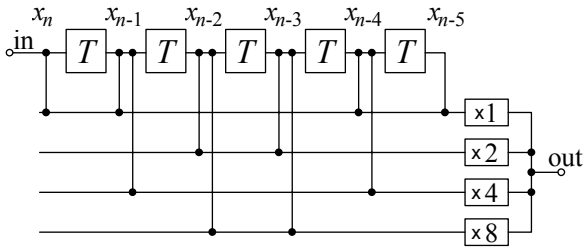

Figure 9. Structure of example 5th order filter implemented using proposed filter structure shown in Figure 7

3. DISCRETE SI FIR FILTER BANKS

Using SI FIR filters described in previous section, efficient filter banks can be implemented. The simplest case, looking from the hardware implementation point of view, is the QMF filter bank composed of filters with symmetrical transfer function described earlier. In this bank both filters share about 90% of the circuitry, without significant power increase and without speed limitation. Absolute values of the coefficients are exactly the same in both the low-pass and the high-pass filters. To change sign of every second coefficient in high-pass filter, each coefficient contains now one additional current mirror as shown in Fig. 5 (c).

In other filter banks that are composed of many filters, or QMF banks with filters with non symmetrical transfer functions, the same coefficient circuits cannot be used. In these cases the only common blocks are delay line and the clock generator. Proposed filter bank that is the universal structure is shown in Fig. 10. For better illustration this filter has been slightly simplified in Fig 10. In the full implementation the additional switches are used in particular mirrors, as is in the filter shown in Fig. 5.

In this bank particular coefficient circuits produce many copies of the signals coming from particular delay elements. In each of these circuits these copies are divided into *m* groups, each producing signals with different gain. Configuration bits have the similar meaning like it was in case of filter shown in Fig. 7 and described by equation (7). The additional 4th index indicates given filter in the bank. This index is in the range 1, …, *k*. To enable full programming possibilities, number of copies in each group is exactly equal to the number of filters, but this number can be further optimized to

minimize number of transistors in the circuit. This is worth noting that in this structure currents flow only in these branches, which are currently turned on.

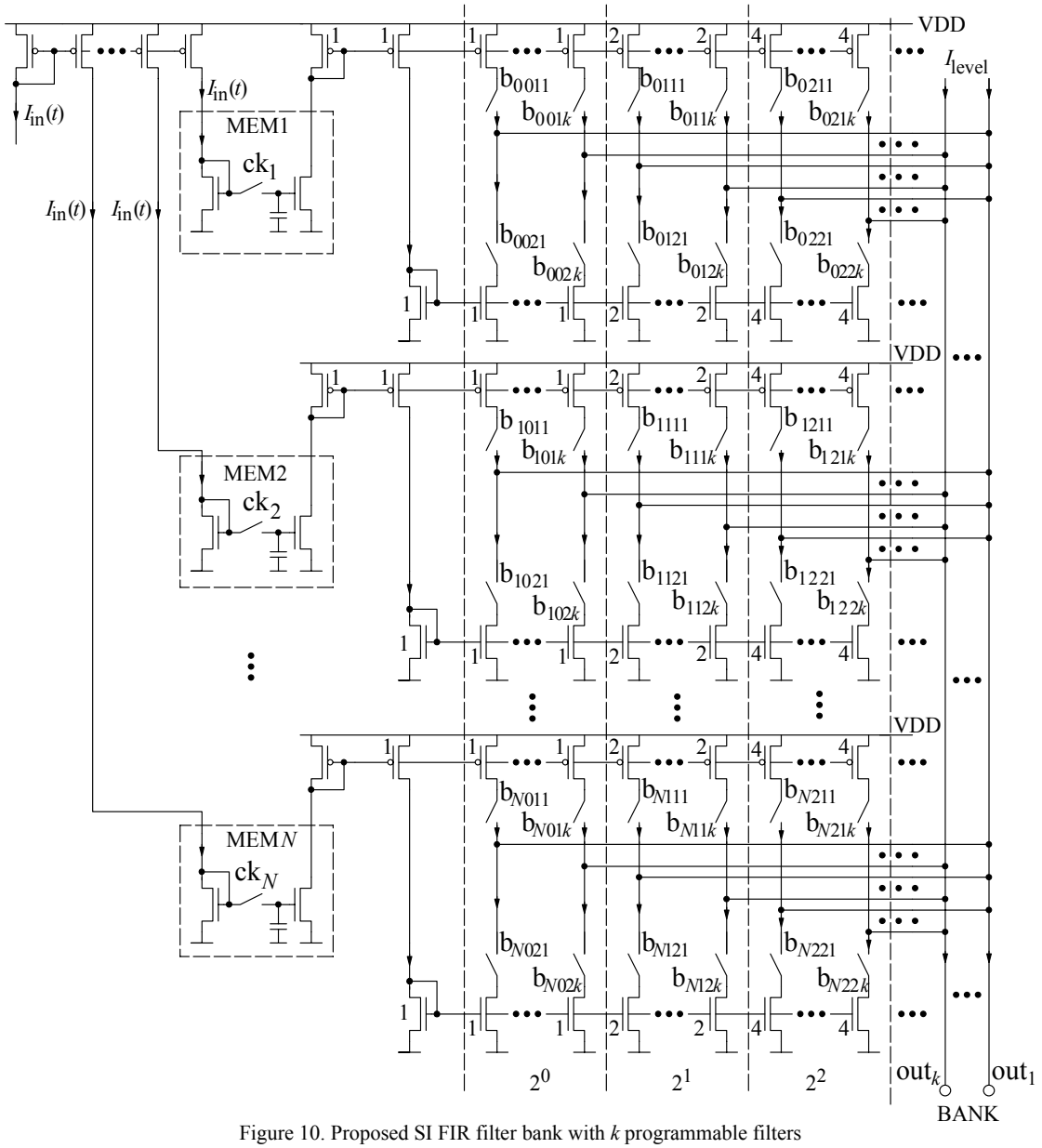

Figure 10. Proposed SI FIR filter bank with *k* programmable filters

Theoretically even large values of resolutions *m* can be implemented, but in practice this is not recommended due to several reasons. The first problem associated with large value of the *m* parameter is transistor matching. When, for example, resolution *m* is equal to 8, then maximum spread between transistors' channels' widths is 128. To ensure good transistor matching in this case, the smallest transistors should be as large as possible, but this significantly increases the total chip area. Another problem is related to the power dissipation. In case when number of large transistors in the circuit is big, then overall power dissipation also increases. The additional problem, that became especially important in case of the circular memory filters, is related to large values of parasitic capacitance in big transistors. Turning on and off of particular current mirrors (due to rotation of bits in the ring digital memory) composed of large transistors, is a time

consuming process, especially when signal currents have small values, that are below several dozen nA. This problem does not occur in case of other SI FIR filters and filter banks, where coefficients are on fixed positions whole the time. This problem can be minimized by introduction of additional mechanism that injects a small amount of charge to parasitic capacitances in those transistors that in a given time moment are turning on. Such mechanism is realized by additional switches, controlled by short impulses and additional constant reference voltage.

On the other hand, when looking from the potential applications point of view filters should enable as large value of parameter *m* as possible. Roughly speaking, higher values of *m* enable better parameters of filters, for example higher attenuations in many cases. There is the trade-off between the system level requirements and the hardware implementation possibilities. An example, very effective solutions of this problem is a serial connection of several FIR filters, what has been described in [11]. In this case using filters with small value of parameter *m*, one can obtain filters with relatively high dispersion between filter coefficients.

Table 1. Implementation of the wavelet Daubechies (Db10) transfer functions in designed filter bank

LP theoretical	LP rounded	LP binary	HP theoretical	HP rounded	HP binary	HP2 rounded
-0.0076	-1	1000001	-0.0189	-2	1000010	-2
0.0010	0	0000000	0.1331	15	0001111	17
0.0026	3	0000011	-0.3728	-41	1101001	-48
-0.0208	-2	1000010	0.4868	53	0110111	62
-0.0505	-6	1000110	-0.1988	-22	1010110	-25
0.0658	7	0000111	-0.1767	-19	1010011	-23
0.0901	10	0001010	0.1386	15	0001111	18
-0.1386	-15	1001111	0.0901	10	0001010	12
-0.1767	-19	1010011	-0.0658	-7	1000111	-8
0.1988	22	0010110	-0.0505	-6	1000110	-6
0.4868	53	0110111	0.0208	2	0000010	3
0.3728	41	0101001	0.0026	3	0000011	3
0.1331	15	0001111	0.001	0	0000000	0
0.0189	2	0000010	-0.0076	-1	1000001	-1

Proposed filter bank has been implemented in CMOS 0.18 μm technology. Value of the *m* parameter has been selected to be equal to 7. The six LSBs in each coefficient are used to adjust the absolute value of the filter coefficient, while the MSB controls sign of the coefficient. This enables programming of the filter coefficients in the range between -63 and 63 with step 1. Filter bank has been designed with order equal to 15 (16 taps). Such filter bank is suitable for different applications, for example in implementation of wavelets. As an example, let us consider the QMF bank of two filters realizing Daubechies (Db10) wavelets. Theoretical transfer functions of such filters can not be directly realized, what is due to large spread between the smallest and the largest coefficients. To make implementation of these wavelets possible in designed filter structure, the smallest coefficients that are below some chosen threshold are first zeroed. The other coefficients are rounded in such a way, to limit them to be within the range between -63 and 63 and to have the integer values. This is shown for illustration in Table 1. The LP and HP denote the lowpass and the highpass filters respectively. During rounding, theoretical values of filter coefficients that have values greater than selected threshold, are first divided by the absolute value of the smallest coefficient (0.0076 in this case), then multiplied by a factor *r* (equal to 0.834 in this case), and finally rounded to the nearest integer number. Value of the factor *r* has been selected experimentally. The final binary notation of particular coefficients is also presented in Table 1 for illustration. Frequency responses of presented transfer functions are shown in Fig. 11 (a). In case of filters with rounded coefficients attenuation in stopband is limited to about 45 dB, what is due to rounding errors. In general, difference between the theoretical attenuation and attenuation in case of filter with rounded coefficients depends on values of the rounding errors. This effect can be minimized using different approaches. To increase the resolution *m* is one of possibilities, but this is disadvantageous looking from hardware implementation point of view. This method can be treated rather as a rough method, which should be supplemented by another method that can be simple described as searching for the optimum case for a given resolution *m*. For different values of parameter *r* attenuation can vary even significantly. In Table 1, in the last column labeled HP2 there is shown another case, when exactly the same theoretical filter coefficients have been calculated using different

factor $r = 0.97$. The resulting frequency responses are shown in Fig. 11 (b) for comparison. Both presented cases fit into the resolution $m = 7$, but in the second case attenuation for many frequencies is more than 10 dB worse than in the first case. The second case is not optimal also due to another reason. Many coefficients have larger values than their counterparts in the first case. Larger coefficients mean larger power dissipation.

Power saving is also possible due to a special construction of the digital memory implemented as rings of NOT gates. In Table 1 the binary notation of particular coefficients is shown. Each memory ring contains bits from a given position in all coefficients (column). For example first ring contains all LSBs {10100101101110}. When data rotate in the digital memory, particular bits change their positions. In practice, in proposed solution, always only part of NOT gates is switched over. For example the 6^{th} ring contains the following bits {00000000001100}. Now, when these bits are moved between memory cells then always only two NOT gates in the ring are switched over (on borders between the 00…00 and the 11 blocks). The important feature here is that NOT gates dissipate power only when they are switched over. As a result, the total power dissipated in the digital memory during switching over is proportional to the number of places where bits change their values. In presented example the total number of NOT gates that are switched over after calculation of each output sample is equal to $4 \cdot (10 + 4 + 8 + 4 + 2 + 2 + 6) = 144$, though total number of NOT gates in rings is equal to 448.

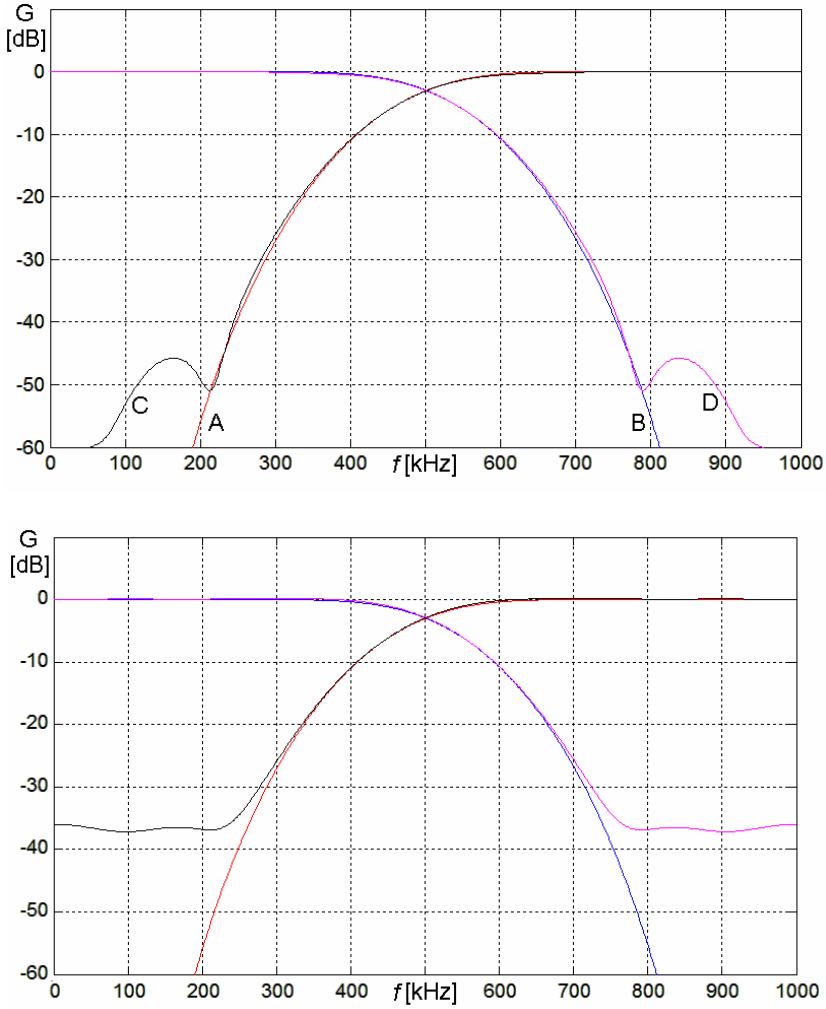

Figure 11. Frequency responses of the lowpass (B, D) and the highpass (A, C) Daubechies (Db10) filters used in QMF bank in multistage digital wavelet transform. Both theoretical (A, B) as well as realized (C, D) filters are presented.

4. CONCLUSIONS

Switched current finite impulse (SI FIR) filter banks presented in this paper can be powerful tools in many applications, like for example in image processing in realization of discrete wavelet transformer (DWT). DWT enables reducing the power dissipated during data transmission, what is paramount feature in WSN applications, where energy is rationed. The example filter bank with filters of order 15 has been designed in CMOS 0.18 μm technology. Resolution of the filter coefficients is equal to 7, what enables for example implementation of DWT, where Daubechies (Db10) transfer functions are realized. Simulated power dissipation of a single filter bank with two composing Db10 filters is equal to about 5 μW for 0.6V supply voltage and sampling frequency about 500 kHz. Low power dissipation is possible due to subthreshold operation of transistors in filter but what is more important due to new circuit solutions like for example ultra low power clock generator that dissipates only about 200 nW.

REFERENCES

1. Y. L. Cheung, A. Buchwald, A sampled-data switched-current analog 16-tap FIR filter with digitally programmable coefficients in 0.8μm CMOS, *IEEE International Solid-State Circuits Conference*, Vol. 40, Feb. 1997, pp. 54 – 55 and 129.
2. M. Helfenstein, J.E. Franca, G.S. Moschytz, Exact design of multirate switched-current FIR filters with improved phase linearity, *IEEE International Conference on Electronics, Circuits and Systems*, 1998, Vol.3, pp. 351 – 354,
3. F.A. Farag, C. Galup-Montoro, M.C. Schneider, Digitally programmable switched-current FIR filter for low-voltage applications, *IEEE Journal of Solid-State Circuits*, Vol. 35, Issue 4, April 2000, pp. 637 – 641
4. E. Burlingame, R. Spencer, An analog CMOS high-speed continuous-time FIR filter, *Proceedings of the 26'th European Solid-State Circuits Conference* (*ESSCIRC*), 2000, pp. 288 – 291
5. G. Fischer, Analog FIR filters by switched-capacitor techniques, *IEEE Transactions Circuits and Systems*, Vol. CAS-37, No. 6, 1990, pp. 808 – 814.
6. P. Gillingham, Stray-free switched-capacitor unit-delay circuit, *Electronics Letters*, 1984, 20 (7), pp. 308 – 310
7. A. Dąbrowski, *Multirate and multiphase switched-capacitor circuits*, Chapman & Hall, London, 1997
8. R. Długosz, New Architecture Of Programmable SC FIR Filter With Circular Memory, *International Conference Mixed Design of Integrated Circuits and Systems* (*MIXDES*), Poland 2005
9. R. Długosz, New Ultra Low Power Switched – Current Finite Impulse Response Filters Realized in CMOS 0.18 μm Technology, *International Conference Mixed Design of Integrated Circuits and Systems* (*MIXDES*), Poland, 2006
10. R. Długosz, P. Pawłowski A. Dąbrowski, Finite Impulse Response Filter Banks Realised using Switched Capacitor Technique, *European Conference on Circuit Theory and Design* (*ECCTD*), Cork, Ireland 2005, pp. III – 257 – 261
11. A. Dąbrowski, R. Długosz, P. Pawłowski, Integrated CMOS GSM Baseband Channel Selecting Filters Realized Using Switched Capacitor Finite Impulse Response Technique, *Microelectronics Reliability Journal*, Vol. 46, May-June 2006