Pavel SILHAVY[1], Ondrej KRAJSA[1], Petr SYSEL[1], Martin KOUTNY[1]

Dept. of Telecommunications, Brno University of Technology, Purkynova 118, 612 00 Brno, Czech Republic (1)

Overlapped filtered multitone modulation and its optimization on VLIW DSP

Abstract. *In recent years, the multicarrier modulations (MCM) have been applied to wire and wireless transmission systems. MCM enable a near-perfect utilization of the provided frequency band on metallic cables and they also effectively eliminate multipath propagation in terrestrial transmission. They are mainly the implementations called Discrete MultiTone (DMT) and Orthogonal Frequency Division Multiplexing (OFDM). These approaches have also disadvantages, which have led to seeking new approaches to the implementation of the MCM modulation scheme. One of them is the Filtered MultiTone Modulation (FMT). In the article, the channel equalization in overlapped and non-overlapped FMT modulation will be compared. The optimal implementation of FMT on the VLIW DSP will be introduced in the second part.Optimized implementation of filter bank with the help of parallel processing is described, and also frame writing/reading and its synchronization are introduced.*

Streszczenie. *W ostatnich latach, w modulacjach multicarrier (MCM) zostały zastosowane systemy transmisji przewodowej i bezprzewodowej.MCM umożliwiają niemal perfekcyjne wykorzystanie pasma częstotliwości przewidziane w kablach a również skutecznie wyeliminować wielodrogowość w transmisji naziemnej. Są to przede wszystkim implementacje o nazwie Discrete wielotonowy (DMT) i Orthogonal Frequency Division Multiplexing (OFDM). Metody te mają także wady, które doprowadziły do poszukiwania nowych sposobów podejścia do wdrożenia systemu modulacji MCM. Jednym z nich jest filtrowane wielotonowe modulacji (FMT). W drugiej części artykułu została opisana optymalna implementacja modulacji FMT. Wykorzystano bank filtrów z równoległym przetwarzaniem, oraz odpowiednie ramki zapisu/odczytu oraz synchronizacji. (Wielotonowa modulacja pokrywająca się MCM i jej optymalizacja)*

Keywords: Multi-Carrier Modulation (MCM), Filtered MultiTone modulation (FMT), Half-overlap subchannel Filtered MultiTone Modulation, Decision Feedback equalizer (DFE), Very Long Instruction Word (VLIW) , Implementation, Digital Signal Processor (DSP).
Słowa kluczowe: modulacja MCM, wielotonowa modulacja FMT.

Introduction

The DMT and OFDM modulations are utilized in data transmission over metallic cables in ADSL, VDSL and PLC systems, but also in terrestrial transmission in the WLAN and WiMAX technologies, and in several other applications.

Last but not least, they are used for digital radio and television broadcasting (DVB-T)[1]. These modulations enable a near-perfect utilization of the frequency band provided in metallic cables with spectrally-shaped transmission characteristics, or an efficient suppression of the influence of multipath propagation on terrestrial transmission. Their disadvantages, which are given by the method of the implementation, include the suboptimal utilization of closely-spectrally-shaped channels, channels with narrowband disturbance, and the suboptimal realization of duplex transmission. The realization of FDM duplex transmission is suboptimal, because the frequency band in the transient area between downstream and upstream channels is not used optimally. The above disadvantages are given by the method of realizing the multicarrier modulation (MCM), where the individual symbols have been generated using FFT and inserted one by one. During the symbol duration the amplitudes and the phases of individual carriers are constant and defined by the QAM constellation diagram of the dimension used. Between individual symbols, the amplitudes and the phases of all the carriers have been changed and thus the spectrum of each carrier is the sinc function, with the parameters given by the square window of each symbol, the attenuation of the first side-lobe being -13dB. Thus each carrier affects a number of neighbouring sub-bands and, on the contrary, the narrowband disturbance affects the data transmission on a number of neighbouring carriers, which are outside the disturbance frequency bandwidth. Furthermore, if the frequency-division duplex is used, additional filtering is necessary so that the individual transmission directions do not affect each other. Paradoxically, it is these filters that have the dominant influence on the appearance of inter-symbol interference. The frequency bandwidth between individual directions is utilized suboptimally, which can be seen from the bit-loading of each ADSL modem.

For these reasons, alternative approaches to the implementation have recently been investigated to improve the above mentioned disadvantages. More recently, a filter bank implementation approach to MCM has frequently been discussed. This method is called Filtered MultiTone modulation (FMT) [2].

Filtered MultiTone Modulation

The FMT modulation represents the modulation technique using a bank of filters to divide the frequency band. The complex values obtained from QAM modulation are, similar to the DMT modulation, the inputs of the modulation system. The number of bits assigned to individual carriers is, the same as with the DMT modulation, determined during the initialization accorcing to the disturbance level and attenuation of the transmission channel. Furthermore, as will be given later, an effective implementation can be realized with the help of the FFT algorithm and thus the increased computation complexity is acceptable. The main disadvantage is the many times higher delay given by additional filtering. This modulation is commonly realized as non-overlapped FMT, where the individual sub-bands do not overlap each other. The suboptimal utilization of the frequency band between individual sub-bands is another disadvantage. Nevertheless, the overlapped FMT modulation, introduced in [3], can be realized, where, similar to the DMT modulation, the individual subchannels are mutually overlapped and thus the bandwidth is utilized effectively. However, the overlap is in the half-subchannel band only.

Principle of Filtered MultiTone modulation and demodulation

The basic structure of FMT modulator, introduced in [2], realizes the critical sampled filter bank. Each complex input X_i^k from the QAM modulators of individual carriers is filtered by FIR filter $h_i(n)$ of the order of $2\gamma N$, which is derived from the low-band prototype filter:

$$x\left(n\right)=\frac{1}{\sqrt{2N}}\sum_{k=-\infty}^{\infty}\sum_{i=0}^{2N-1}X_i^k h_i\left(n-2kN\right), \tag{1}$$

$$\text{where } X_i^k \begin{cases} X_{2N-i}^k=\left(X_i^k\right)^* & \text{for } i=1,..,2N-1 \\ 0 & \text{for } i=0,N \end{cases}$$

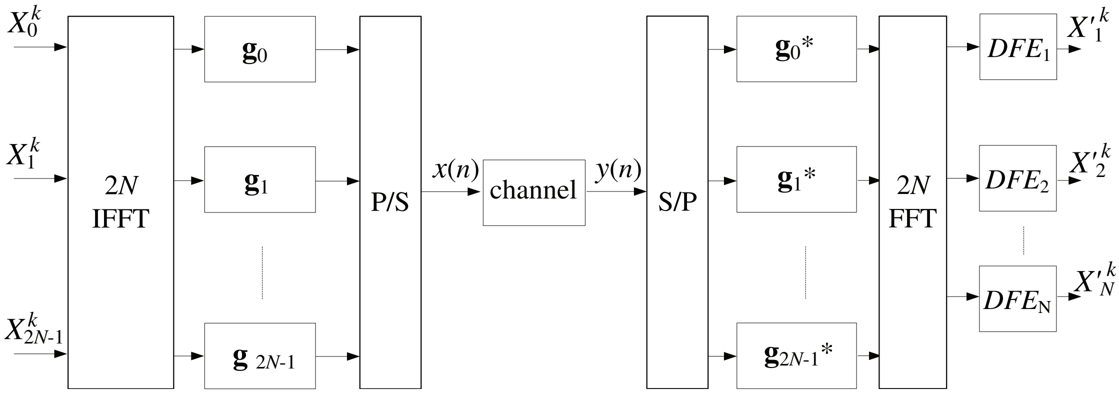

Fig. 1. The effective realization of the FMT modulator and demodulator.

This implementation is also of high computation complexity. Fig. 1 shows an effective realization of the FMT modulator and demodulator, introduced in [4]. This implementation uses the FFT transform, each output of which has been filtered by the so-called polyphase filter of the order g. The coefficients of the polyphase filters $\mathbf{g}_i = [g_i(0), ..., g_i(\gamma-1)]$ can be obtained from the prototype filter $h(n)$ of the order of $2\gamma N$:

(2) $$g_i(l) = h(2lN + i), \text{where } l = 0..\gamma - 1$$

The prototype filter can be designed by sampling the frequency characteristic (3) and applying the optimal window. Useful windows enabling the design of orthogonal filter banks are, for example, the Blackman window, the Blackmanharris window, the Hamming window, the Hann window, the Nuttall window, and some others.

(3) $$\left|H(e^{j\pi fT})\right| = \begin{cases} 1 & \text{for } -\frac{1}{2T} \leq f \leq -\frac{1}{2T} \\ 0 & \text{otherwise} \end{cases}$$

As mentioned above, the FMT modulation can be implemented as overlapped or non-overlapped.

Fig. 2 shows a comparison of DMT, non-overlapped and overlapped FMT modulations. It can be seen from the figure that, due to the overlap, DMT and overlapped FMT make optimal use of the whole frequency band. In the case of non-overlapped FMT, the area between carriers has been unused. Moreover, the wider subchannels enable achieving a higher attenuation for lower polyphase fitter orders g in the case of overlapped FMT. The subchannels are separated perfectly in the case of non-overlapped FMT, thus a disturbance or an inhomogeneity in one subchannel does not affect the neighboring carriers.

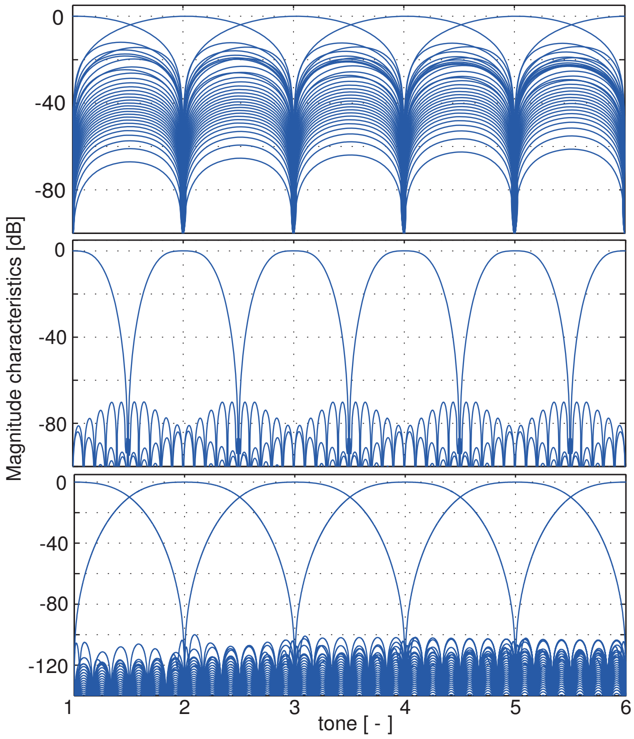

Fig. 2. Example of the comparison of the spectra of DMT, non-overlapped FMT with $\gamma = 10$ - Blackman window and overlapped FMT with $\gamma = 6$ - Nuttall window.

Equalization in FMT modulation

The principle of FMT signal demodulation can be seen in the right-hand part of figure 1. In the case of non-overlapped FMT, the individual carriers are completely separated, thus no inter-carrier interference (ICI) occurs. In the case of overlapped FMT, the suppression of ICI has been ensured thanks to the orthogonality of the modulation bank, analogously to the DMT modulation.

The inter-symbol interference occurs even in the case of ideal channel, which is given by the FMT modulation principle. To put it more exactly, the spectral shaping of individual channels by additional filtering leads to the occurrence of ISI interferences, which have to be eliminated. Equalization to minimize ISI interference can be performed individually for each subchannel in the frequency domain, because each carrier creates a separated complex channel.

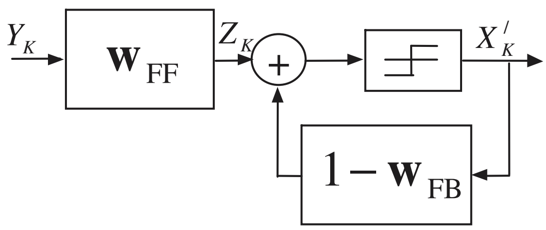

Fig. 3. Decision Feedback equalizer structure.

In figure 1 the complex values of DFE (Decision Feedback equalizer) equalizers are drawn for the 1^{st} to N^{th} channels. The DFE equalizer structure is shown in figure 3.

The DFE equalizer contains two digital FIR (Finite Impulse Response) filters and a decision circuit. The feedforward filter (FF) with the coefficients $\mathbf{w}_{\mathrm{FF}}$ and of the order of M is to shorten the channel impulse response to the feedback filter (FB) length R. The feedforward filter is designed to set the first coefficient of the shortened impulse response of channel to unity. With the help of the feedback filter (FB) with the coefficients $1-\mathbf{w}_{\mathrm{FB}}$ and of the order of R we subtract the rest of the shortened channel impulse response. The derivation of MMSE DFE equalizer solution was published several times, e.g. [5]. In this section, the useful results of derivation will be recapitulated. The coefficients of the equalizer filters for the given delay Δ can be determined using the equation: (4)

$$\mathbf{w}_{\mathrm{FB}} = \frac{\mathbf{R}_{\mathrm{X|Y}}^{-1}\mathbf{f}_0}{\mathbf{f}_0^{\mathrm{T}}\mathbf{R}_{\mathrm{X|Y}}^{-1}\mathbf{f}_0}, \quad \mathbf{w}_{\mathrm{FF}} = \left(\mathbf{H}^{\mathbf{H}} \cdot \mathbf{H}\right)^{-1}\mathbf{H}^{\mathbf{H}}\mathbf{C}^{\mathbf{H}}\mathbf{w}_{\mathrm{FB}}$$

where matrix $\mathbf{R}_{\mathrm{X|Y}}$ is:

(5) $$\mathbf{R}_{\mathrm{X|Y}} = \mathbf{I} - \mathbf{G}\mathbf{H}\left(\mathbf{H}^{\mathbf{H}} \cdot \mathbf{H}\right)^{-1}\mathbf{H}^{\mathbf{H}}\mathbf{G}^{\mathbf{H}}$$

matrix $\mathbf{H}$ is the convolution matrix of the channel from which we chose the shortened part in terms of delay parameter Δ with the help of matrix $\mathbf{G}$:

(6) $$\mathbf{G} = \mathrm{diag}(\overbrace{\underbrace{0,...,0}_{\Delta},\underbrace{1,...,1}_{R},0,...,0}^{k+M-1\ \text{samples}})$$

and $\mathbf{f}_0$ is the column vector of length R with element one in the first position:

(7) $$\mathbf{f}_0 = [1, 0_1, \ldots, 0_{R-1}]$$

To compare equalization in non-overlapped and overlapped FMT modulations we realized a series of simulations of various FMT systems with different filter orders γ and design windows. The comparative parameter of the achieved minimizations was the summation of the MSE of all the carriers used:

(8) $$MSE = \sum_{i=1}^{N}\left(\mathbf{f}_0^{\mathrm{T}}\mathbf{R}_{\mathrm{X|Y}}^{-1}\mathbf{f}_0\right)$$

Figures 4 and 5 give examples of the simulation results. In figure 4 the comparison is on the ideal channel and in figure 5 it is on the ANSI 13 channel. In both examples the systems had $N = 32$ carriers.

Tests on different channels proved that in the case of overlapped FMT modulation a better MSE minimization ant thus a better channel equalization can be achieved than in the case of non-overlapped FMT. The difference in the finite achieved minimizations is ordinal. In the case of a lower equalizer filter order a smaller minimization is achieved for overlapped FMT too in comparison with non-overlapped FMT. Overlapped modulation appears to be a better choice due to not only the optimal frequency spectrum utilization, but also due to the equalizer complexity and the final minimization value achieved.

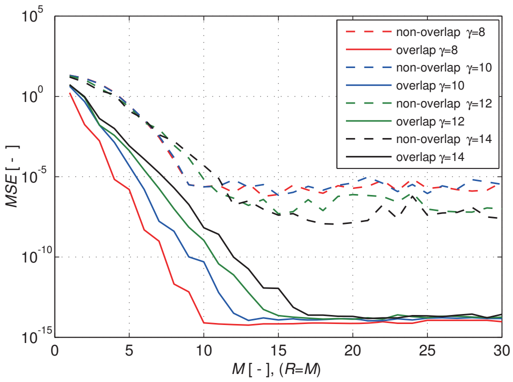

Fig. 4. Comparison of achieved MSE of non-overlapped and overlapped FMT modulations of systems with different filter orders γ and Nutall windows on ideal channel.

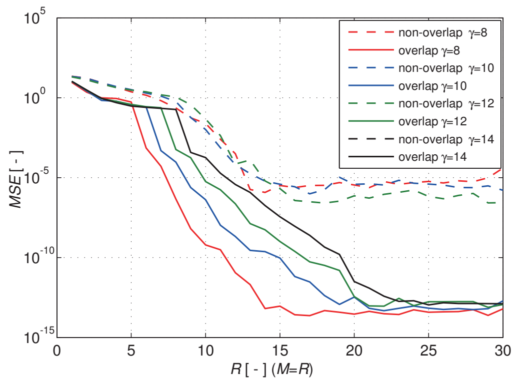

Fig. 5. Comparison of achieved MSE of non-overlapped and overlapped FMT modulation of systems with different filter orders γ and Blackman windows on ANSI 13 channel.

Optimal implementation of FMT on VLIW DSP

Compilers designed for digital signal processors differ from the ANSI-C or C++ standard in a few details, which in the ultimate result have a considerable effect on the speed and stability of algorithm implementation. The basic difference lies in that the defined data types are fully adapted to the architecture of digital signal processor. The number of data bits and the format of storing numbers in a given code (mostly the two's complement) correspond to the actual storage of numbers in digital signal processor registers. When optimizing the source code it is convenient either to enter the instructions of digital signal processor assembler directly into the C-language source code or to use the intrinsic functions, which are assembled as a single instruction. In this way, the critical parts of source code that the assembler is not able to analyze correctly can be optimized. In parallel processing, the given algorithm can be realized simultaneously for several values of the input signal. Using parallel processing will greatly increase the speed of algorithm processing. Implementation of the system is realized using the TMS320C6713DSK development kit with the TMS320C6713 digital signal processor. The AIC32 audio codec takes care about data transmission to the channel. The system distinguishes between odd and even sequences in the data field and on this basis distributes data into the right or the left

Algorithm .1: $\text{S\_DEC}(*buf, len, window, size)$

$begin$
for $i \leftarrow 0$ **to** len

do
  if $*buff \rightarrow rp > window$
    then return $(true)$
  $*buff \rightarrow rp++;$
  comment: read pointer incrementation
  if $*buff \rightarrow rp == *buff \rightarrow ep$
    then $*buff \rightarrow rp = *buff \rightarrow buff$
  comment: end of cyclic buffer testing

return $(false)$
comment: it is not a sync buffer
end

Example 1. Filter bank for FMT in C

```
void bank_filt( const int x[restrict],
                const int h[restrict],
                      short y[restrict],
                    //frame length
                    int M,
                    //overlap factor
                      int g){
  int i;

  for (i=0 ; i < M*g ; i=i+2)
  {
      //sample in i-th position
      *(y+i%M) += _mpy( h[i], x[i]);
      //sample in i-th + 1 position
      *(y+i%M+1) += _mpyh( h[i], x[i]);
  }
}
```

channel. This advantage was used in the realization of the test implementation. The first channel was thus used to ensure synchronization while the second channel was used for the actual data transfer. The processing of transmitted data can be divided into several stages, which can be implemented separately. First, blocks of QAM modulation symbols are generated from the input bit sequence. The constellation diagram is selected for each carrier while establishing a connection, using the signal-to-noise ratio. A Recommended training sequence is used when establishing the connection. Symbols are assigned to bit sequences using a look-up table so that the transmitting power is the same for all constellation diagrams. The next step on the transmitter side is to implement the IFFT, i.e. the conversion from the frequency domain to the time domain. The DSPLIB library, optimized for the TMSC6713 processor, was used in the implementation, namely the DSP_ifft16x16 function, which works with 16-bit input values of the float type and a 16-bit field twiddle-factor, also of the float type. The function return value is then also float. Next, the filtering of the frame obtained by IFFT and by a filter bank is implemented. We can choose from several approaches to implement the filter bank. The first is filtering that corresponds exactly to Fig.1, i.e. filtering each i^{th} sample by a filter with coefficients corresponding to the polyphase filter in the i^{th} branch. This approach could be optimized by using the functions of the DSPLIB library, but this feature brings a number of constraints on the length of each filter, and on data alignment[6]. Furthermore, it would be necessary to keep in memory a large number of signal samples. The second approach is filtering according to Fig.6 and example of realization in C is in example 1. The prototype filter is stored in the memory in one variable such that the sequence number of samples corresponds to the sequence number of data on each carrier.

The input sample pointer `x` and the pointer to the field of prototype filter coefficients `h` can be declared by the key word `const` since in the course of calculation the input sample value and the values of individual filter coefficients will not change. The output sample value and the values of state-space variables will, on the contrary, change during calculation and thus they cannot be declared by the key word `const`. It is obvious from the algorithm structure that the individual input arguments represent mutually independent data structures, which will be stored in separate memory locations. In that case it is of advantage to use the key word `restrict`, which informs the compiler about the memory-independence of the variables. In case the output sample was entered into the same memory field as the input samples (in-place processing), there would evidently be a dependence relation between the in and out pointers and the restrict keyword could not be used in declaring the two arguments. Comparison of optimized and non-optimized implementations are in tables 1 and 2. For the purposes of the ADC, the data has to be amplified and converted to integer data type. This data type is supported by the ADC and DAC converters. Also sorting and inserting of synchronization took place while converting these data to values appropriate for the transfer. The synchronization consisted in inserting a synchronization pulse always at the beginning of each multiframe. The multiframe contains a predefined number of frames. This value is 69 frames in ITU G.922, where the first frame of the transmission chain is used to synchronize the remaining 68 frames. Incoming frames are stored in a circular buffer. This buffer can hold up to several multiframes, but from the capacity point of view, this value is set to 5. Two independent pointers, read and write, take care of the read and write operations from/to the cyclic buffer. Thus no overwriting of data that have not been read should occur. After writing the data into the receiving buffer, the starting pulse of multiframe is identified. This is guaranteed by algorithm 1.

By calling the s_dec function the pointer to a defined structure buf is inserted into the buffer. This structure contains pointers to all the necessary data units in the implementation of reading. The current reading position is tested after the calling. If this position corresponds to a predefined level of the synchronization pulse decision window, the algorithm evaluates that it is a synchronization pulse. If the current value of the test element does not match the level of the decision window, the algorithm increments the read pointer and continues testing. If the read pointer indicates the end of the cyclic buffer, the value of starting field is assigned to pointer.

The length of the test field is always equal to the length of the defined frame. Thus no loop algorithm should occur here. In the event that within the given frame no synchronization pulse was found, it was one of the transmission frames. In the future this testing will be seen to by calculating the number of received frames and thus computational resources will be saved. Subsequently the received frames are filtered, just like in the transmitter, and the FFT is performed.

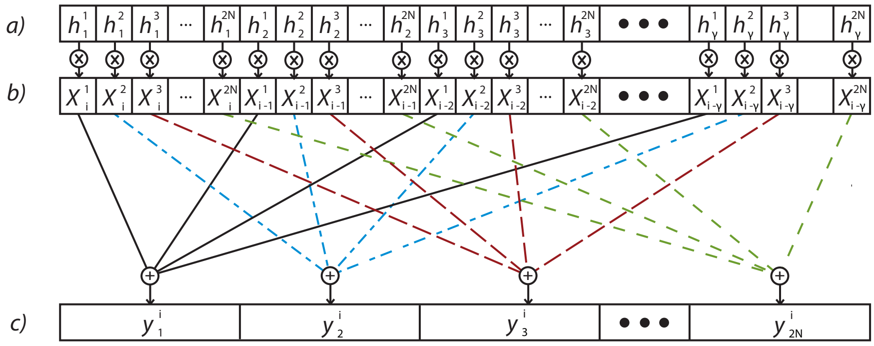

Fig. 6. Optimized algorithm for implementing FIR filter bank

FFT	DMT	FMT $\gamma = 8$	FMT $\gamma = 16$	D-FMT $\gamma = 8$	D-FMT $\gamma = 14$
512	18432	38912	55296	8392704	14684160
1024	40960	81920	114688	33562624	58728448
4096	196608	360448	491520	536903680	939556864
8192	425984	753664	1015808	2147549184	3758161920

Table 1. Without optimization

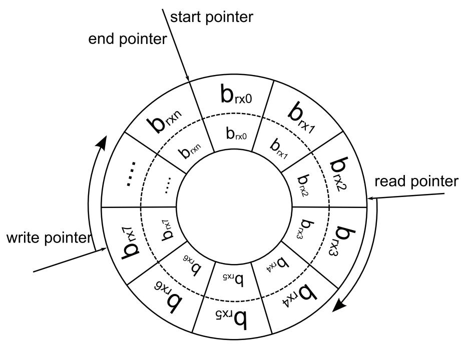

Fig. 7. Optimized algorithm for implementing FIR filter bank

The process in the receiver is exactly the opposite of the process in the transmitter.

FFT	DMT	FMT $\gamma = 8$	FMT $\gamma = 16$
512	3190	5247	6271
1024	6262	10367	12415
4096	34947	51340	59532
8192	69788	102565	118949

Table 2. Optimized implementation

Conclusions

In the article, non-overlapped FMT and overlapped FMT modulation systems were presented. We mentioned the structure of the modulator and demodulator, and the possibilities of an equalization implementation based on the DFE equalizer bank. Results of the simulation of complexity and achieved equalization results for both approaches were presented. Test results on different channels proved that in the case of overlapped FMT modulation a better MSE minimization ant thus a better channel equalization can be achieved than in the case of non-overlapped FMT. An effective implementation of this modulation using the FFT algorithm was compared with a direct implementation and a proposed optimized implementation. The implementation effectiveness was compared with the number of multiply and accumulate (MAC) instructions and cycles that are necessary for calculating one output frame. An optimized implementation of FMT modulation on DSP was also shown. This solution consists of two parts, namely the implementation of filter bank and the implementation of frame generation, writing, reading, and synchronization. Both of these parts were described with examples of solution in the C code. The proposed design was implemented and tested on a TMS320C6713 development starter kit.

Acknowledgements

This work was supported in part by Grant Agency of Czech Republic GACR project No. 102/09/1846, by MSM 021630513 research program, and by Grant Agency of Brno University of Technology No. FEKT-S-10-16.[**?**]

REFERENCES

[1] Stukavec, R.; Kratochvil, T. Simulation and Measurement of the Transmission Distortions of the Digital Television DVB-T/H Part 1: Modulator for Digital Terrestrial Television. *Radioengineering*, 2010, ro?. 19, ?. 2, s. 338-346. ISSN: 1210- 2512.

[2] Cherubini G., Eelefteriou E., Olcer S., Cioffi .M.. Filter bank modulation techniques for VHDSL. *IEEE Communication Magazine*, May 2000, pp.98 - 104.

[3] Silhavy, P. Half-overlap subchannel Filtered MultiTone Modulation with the small delay. In *The Seventh International Conference on Networking ICN 2008*. pp.474-478, Cancun, Mexico, IARIA, LCN 2007941923, April 2008. ISBN: 978-0-7695-3106-9.

[4] Berenguer, I., Wassel, J. I., FMT modulation: receiver filter bank definition for the derivation of an efficient implementation, *IEEE 7th International OFDM Workshop*, Hamburg, Germany, Sep. 2002.

[5] Sayed, Ali H. *Fundamentals of Adaptive Filtering*, Wiley, NJ, 2003 ISBN 0-471-46126-1.

[6] Sysel, P.; Krajsa, O. Optimization of FIR filter implementation for FMT on VLIW DSP. In *Proceedings of the 4th International Conference on Circuits, Systems and Signals (CSS'10)*. 1. Corfu: WSEAS Press, 2010. s. 169-173. ISBN: 978-960-474-208- 0.

Authors: Ph.D. Pavel Silhavy, M.Sc. Ondrej Krajsa, Ph.D. Petr Sysel, M.Sc. Martin Koutny, Dept. of Telecommunications, Brno University of Technology, Facultu of Electrical Engineering and Communication, Brno University of Technology,ul. Purkynova 118, 612 00 Brno, Czech Republic, email: silhavy@feec.vutbr.cz, krajsao@feec.vutbr.cz, sysel@feec.vutbr.cz, koutnym@feec.vutbr.cz