

Combined Matched Filter and Arbitrary Interpolator for Symbol Timing Synchronization in SDR Receivers

Awan, Mehmood-Ur-Rehman; Koch, Peter

Published in:

Proceedings of the 13th IEEE International Symposium on Design and Diagnostics of Electronic Circuits and Systems, April 14-16, 2010 Vienna, Austria

DOI (link to publication from Publisher):

[10.1109/DDECS.2010.5491797](https://doi.org/10.1109/DDECS.2010.5491797)

Publication date:

2010

Document Version

Accepted author manuscript, peer reviewed version

[Link to publication from Aalborg University](#)

Citation for published version (APA):

Awan, M.-U.-R., & Koch, P. (2010). Combined Matched Filter and Arbitrary Interpolator for Symbol Timing Synchronization in SDR Receivers. In *Proceedings of the 13th IEEE International Symposium on Design and Diagnostics of Electronic Circuits and Systems, April 14-16, 2010 Vienna, Austria* (pp. 153). IEEE Press. <https://doi.org/10.1109/DDECS.2010.5491797>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy

If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Combined Matched Filter and Arbitrary Interpolator for Symbol Timing Synchronization in SDR Receivers

Mehmood-ur-Rehman Awan, Peter Koch

Center for Software Defined Radio

Department of Electronic Systems, Aalborg University, Denmark

Email: (mura, pk)@es.aau.dk

Abstract—This paper describes a low complexity multi-rate synchronizer that makes use of a polyphase filter bank to simultaneously perform matched-filtering and arbitrary interpolation for symbol timing synchronization in a sampled-data receiver. Arbitrary Interpolation between available sample points is achieved by selecting the appropriate filter in the bank having the polyphase partitioned matched filter, which provides the optimal sampling time. Two different structures are considered which are modified to perform combined arbitrary resampling. The computational complexity is analyzed to have the resource optimal solution. Simulation results are analyzed and their resource utilization for Virtex-5 FPGA implementation is presented.

I. INTRODUCTION

Software Defined Radios (SDR) are highly configurable hardware platforms that provide the technology for realizing the rapidly expanding future generation of digital wireless communication infrastructure. Many sophisticated signal processing tasks are performed in a SDR, including advanced compression algorithms, power control, channel estimation, equalization, forward error control and protocol management. In recent years, the FPGA technology has undergone revolutionary changes which has enabled implementation of SDR systems ever further. The gate densities and clock speeds of recent FPGA generations provide the communication system architect with a highly configurable logic fabric that can be used for realizing sophisticated real-time signal processing functions [1]. Amongst the more complex tasks performed in a high data rate wireless system is synchronization. A large amount of time is spent executing this task, and normally significant amount of hardware and software in a SDR is dedicated to synchronization [2]. Physical layer synchronization of symbol timing is required when samples of the received signal are misaligned with the data symbols generated by the transmitter. Synchronization can be done by the use of polyphase filter banks which allow greater flexibility and efficiency in the receiver by computing only those multiplications necessary for matched filtering while simultaneously interpolating to achieve a sample point sufficiently close to the optimum [3].

The development of the polyphase filterbank and its application to perform the interpolations required for symbol

timing synchronization is presented in [4]. The authors of [3] extended the work presented in [4] by adding an additional control loop for carrier synchronization after matched filtering. In our work we combine the polyphase structures for performing the arbitrary interpolation and the matched filtering to have a single structure for symbol timing synchronization.

II. SYMBOL TIMING RECOVERY

There exist two standard DSP approaches to obtain timing recovery in modern QAM receivers [5]. The first approach uses a polyphase interpolator to calculate the samples at the desired locations from the offset samples provided by the free running ADC. These position corrected samples are processed in the receiver matched filter whose output, through a detector, forms a timing error signal to guide the interpolating filter re-sampling process [5]. The second approach folds the

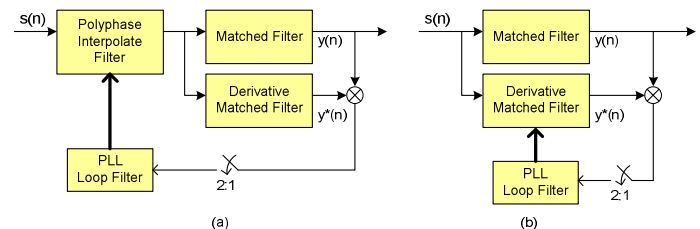


Fig. 1. Maximum Likelihood (ML) control of (a) Polyphase Interpolating Filter and of (b) Polyphase Matched Filter [5].

interpolation process into a polyphase matched filter. The separate paths of this polyphase filter represent a collection of filters matched to different time offsets between input sample positions and the output sample peak correlation value position. The timing recovery process simply has to determine which filter that matches the unknown time offset between input and output samples. Either process uses a phase locked loop (PLL) to direct the pointer to the appropriate phase leg of the polyphase filter [5].

Figure 1 presents the structure of the two timing recovery schemes based on the Maximum Likelihood (ML) error term formed from two matched filters. Fig. 1a shows the control of the polyphase interpolator while the Fig. 1b shows the

control of the matched filters. The loop in Fig. 1a exhibits a larger transport delay through the cascade of two filters than does the loop in Fig. 1b through the delay of a single filter. Due to the additional delay, the polyphase interpolating filter must have a slower (or lower bandwidth) loop filter than does the polyphase matched filter [5].

III. SYSTEM DESIGN

As a case study we will now investigate an SDR-based design of a BGAN (Broadband Global Area Network) satellite receiver as shown in Figure 2. The ADC data sampled at 57.85 MHz is down-sampled to 723.125 kHz by a cascaded structure of 16:1 and 5:1 re-samplers. It is further required to match the received signal with the transmitted pulse shape using matched filtering along with the timing recovery, thus delivering the output at 151.2 k-symbols/sec. Now the two different approaches for timing recovery, mentioned earlier are employed and compared in order to achieve an optimal solution, in terms of computational complexity.

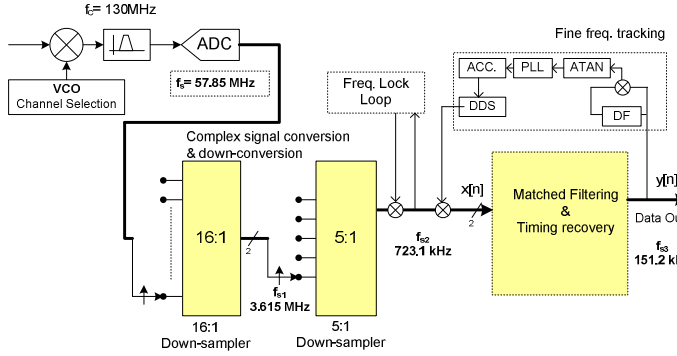


Fig. 2. SDR design for a BGAN satellite receiver. Data is sampled at 57.85 MHz and down-sampled to 723.125 kHz by a cascaded structure of 16:1 and 5:1 re-samplers followed by arbitrary interpolator and match filter for symbol timing recovery.

In the first approach having separate interpolation and match filter, the timing error signal guides the interpolation filter re-sampling operation to achieve the timing synchronization. According to the system requirements, a sample rate of 723.125 kHz is converted to 151.2 kHz. The rate conversion factor is M (number of polyphase partitions or phases) times $723.125/151.2 = 4.782$, which therefore requires an arbitrary interpolator. The structural diagram for the first approach i.e., separate polyphase interpolating filter is shown in Figure 1a, replaced with arbitrary interpolator is illustrated in Figure 3.

Figure 3 shows a cascaded structure of arbitrary re-sampler [6] and matched filter. Arbitrary re-sampler is based on linear interpolation to a position between available output points in an M -path interpolator. Interpolation is achieved by polyphase lowpass filter, and the polyphase derivative filter is used for local slope correction. The polyphase lowpass and the polyphase derivative filters' coefficient pointer is addressed by the integer part of the phase accumulator. The phase accumulator is a modulo- M (number of polyphase

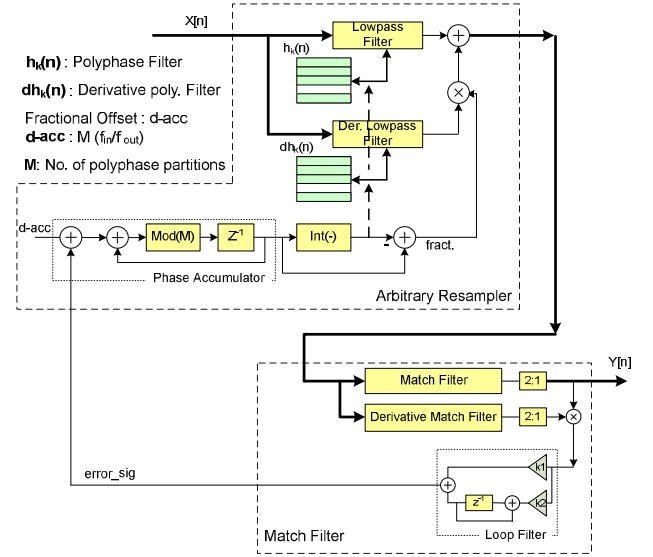


Fig. 3. Cascaded structure of arbitrary re-sampler and matched filter with ML control of polyphase interpolating filter.

partitions or phases) addition of the fractional offset factor d_{acc} and the error signal $error_sig$ generated by the loop filter. The fractional part of the phase accumulator is multiplied with the derivative filter's output and added to the lowpass filter's output for the slope correction.

In this case, the matched filter operates at twice the symbol rate i.e., $2 \times 151.2 \text{ kHz} = 302.4 \text{ kHz}$, but the loop filter still operates at 1 sample per symbol. So the required fractional offset factor d_{acc} for the arbitrary re-sampler comes out to be M times $723.125/(2 \times 151.2) = M \times 2.39$. The polyphase interpolator filter is designed at upsampled frequency $16 \times 723.125 \text{ kHz}$ (having 16 phases or subfilters) with transition band of 100-200 kHz and 60 dB side lobe attenuation. The resulting 320 coefficients are partitioned into $M=16$ polyphase sub-filters (phases) each having 20 coefficients. The fractional offset factor d_{acc} then becomes 38.26. The matched filter is designed with a roll-off factor of 0.25, filter length of ± 4 symbols, and operating at 2 samples per symbol, which results in 33 coefficients.

A computational complexity analysis is required and in this work we define the complexity in terms of multiplications per second, similar to the definition applied in [7]:

$$R_m = N \times f_s \quad (1)$$

R_m is multiplications/sec, N is the number of non-zero coefficients, and f_s is the input sampling frequency. In multirate filters, R_m can be reduced by a sampling rate conversion factor. For polyphase decimator:

$$\begin{aligned} R_{mDEC} &= (N \times f_{si})/M \\ R_{mDEC} &= N \times f_{so} \end{aligned} \quad (2)$$

where $f_{si}/M = f_{so}$, f_{si} is sample rate at input, and f_{so} is sample rate at output of a polyphase filter.
For polyphase interpolator:

$$\begin{aligned} Rm_{INT} &= (N \times f_{so})/L \\ Rm_{INT} &= N \times f_{si} \\ (\text{where } f_{so}/L &= f_{si}) \end{aligned} \quad (3)$$

The computational complexity for the arbitrary resampler and the matched filter turns out to be $2 \times (96.77) \times 10^6$ and $2 \times (9.98) \times 10^6$ multiplications per second (M-Mult/sec), respectively. The factor 2 accounts for the complex operations.

In the second approach, the interpolation process is folded into the polyphase matched filter which is realized by filter bank index selection, and therefore the separate interpolating filter following the matched filter is not required. Different loop control structures for this kind of design are possible. An M-stage polyphase filter bank with input data sampled at approximately N samples/symbol can be used in a loop that operates at (a) MN samples/symbol, (b) N samples/symbol, or (c) 1 sample/symbol [4]. It has been investigated from simulations that these loop architectures work only if the input data rate is some integer multiple of the desired output data rate, but not for the case involving arbitrary re-sampling operation. So it requires some modifications to embed the arbitrary re-sampling process.

In order to have the arbitrary re-sampling process embedded in the combined interpolation and match filtering operation, the cascaded approach presented in Figure 3 is modified by the following steps and the modified design is presented in Figure 4.

- 1) Replace the arbitrary polyphase interpolator low-pass (and polyphase derivative) filters with polyphase matched (and polyphase derivative matched) filters.
- 2) Discard the previous matched and derivative matched filters.
- 3) Reconnect the timing error and loop blocks to the polyphase matched and the polyphase derivative matched filters as they were connected before to the matched and derivative matched filters.

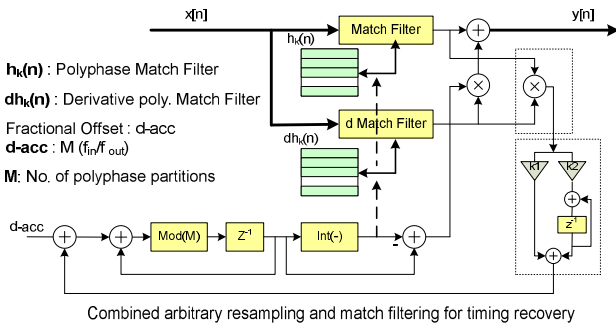


Fig. 4. Combined structure of arbitrary re-sampler and matched filter with ML control of polyphase matched filter.

To our knowledge, this modified design has not been published previously, and therefore we consider the concept as a new and innovative idea. The matched filter is designed with the following specifications; roll-off factor of 0.25 with filter length of ± 4 symbols, and the loop operates at 1 sample per symbol, which results in 624 coefficients. Upsample L and downsample M factors are 16 and 76.52, respectively. The corresponding computational complexity becomes $2 \times (94.35)$ M-Mult/sec. Based on the complexity analysis for the two designs (interpolating filter design and the modified polyphase matched filter design) as shown in Table I, it can be concluded that our modified polyphase matched filter design leads to a reduced computational complexity by more than 10%. Besides that in the first design, loop exhibits a larger transport delay through the cascade of two filters than does the modified design's loop through the delay of a single filter.

Computational Complexity (M-Mult/sec)	
Separate Arbitrary interpolation and Matched filtering	(Arbitrary Resampler) 2×96.77 (Matched Filter) 2×9.98
Combined Arbitrary interpolation and Matched filtering	2×94.35

TABLE I

IV. SIMULATIONS

Closed-loop simulations are performed for the modified polyphase matched filter design in order to demonstrate the features of polyphase filterbank for performing combined arbitrary interpolation and matched filtering for timing synchronization. It uses ML timing error detector which can be easily incorporated into the polyphase filter bank [4]. Simulated input test signal for the BGAN receiver is a 4-QAM signal having one desired channel with no noise and interference. The matched filter is designed at an up-sampled frequency of 32×723.125 kHz (having 32 phases or subfilters) with a roll-off factor of 0.25 and filter length of ± 4 symbols. The resulting 1248 coefficients are partitioned into $M=32$ polyphase sub-filters (phases) each having 39 coefficients. The fractional offset factor d_{acc} becomes 153.042. The proportional K_p and integral K_i gain values are set to 1.6617 and 0.0033 respectively. Data samples at 1 sample/symbol are processed by the polyphase matched filter and the polyphase derivative matched filter filterbanks. The product of the two filterbank outputs form the timing error which is updated once per symbol. The timing error signal is filtered by a proportional-plus-integrator loop filter which is required for a second-order loop to track out the symbol clock frequency offset [8]. The loop filter output together with fractional offset is used to control the increment in the modulo-32 counter (NCO) which becomes constant when the loop has achieved lock.

Figure 5 shows the simulation results as eye and constellation diagrams, both plotted for 30 to 4000 and 3000 to 4000 symbols. It is seen that the loop filter converges the signal to

the desired constellation. Figure 6 demonstrates the loop filter response and the incrementation of the timing accumulator. The loop filter's response converges and the NCO control (timing accumulator) settles to a constant steady-state value (153 in this case).

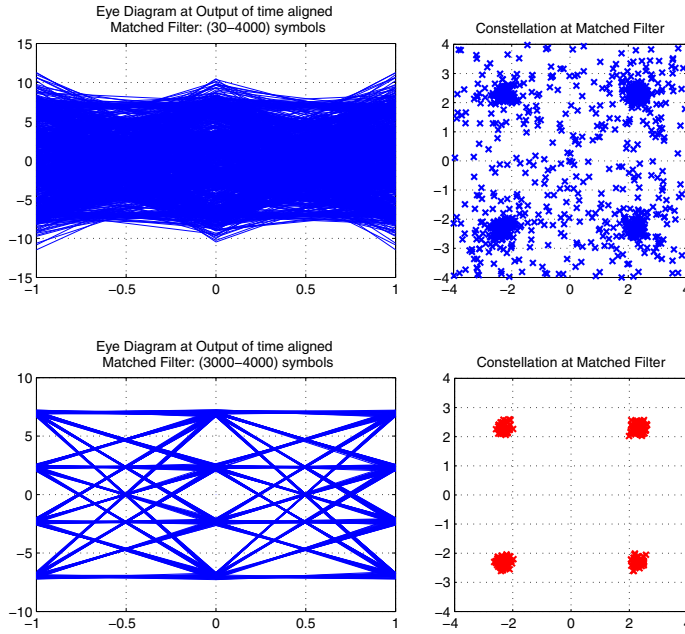


Fig. 5. Eye and Constellation diagram for 30 to 4000 and 3000 to 4000 symbols. The plots for 30 to 4000 symbols give a messy picture due to non-aligned symbols before the synchronization has achieved.

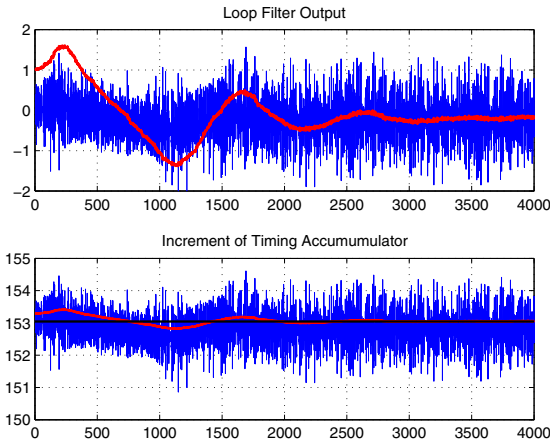


Fig. 6. Loop filter output and timing increment of Accumulator

V. FPGA IMPLEMENTATION

The design is implemented on a Virtex5-vsx50tff1136-3 FPGA. Block RAMs are used for storing filter coefficients and DPRAM (Dual Port RAM) is used for loading of the input data samples as a delayed line required for filtering operation. The device utilization summary is tabulated in Table II. The design can operate at a maximum frequency of 380.625 MHz.

Device Utilization (Virtex5-vsx50tff1136-3)

Number of Slice Registers	520 out of 32640 (1%)
Number of Slice LUTs	533 out of 32640 (1%)
Number of Block RAM/FIFO	3 out of 132 (2%)
Number of DSP48Es	6 out of 288 (2%)

TABLE II

VI. CONCLUSIONS

In this paper a combined matched filter and arbitray interpolator for symbol timing synchronization in SDR receivers was presented. Two different structures were considered for this task, which were next applied for SDR implementation of a BGAN satellite receiver. A structure with ML control of polyphase matched filter was modified to embed arbitrary interpolation, which in terms of computational complexity is more efficient than the cascaded structure of arbitrary interpolator and matched filter. We find a 10% reduction. Simulations for the modified structure with the loop operating at 1 sample/symbol were presented to illustrate the response of loop filter and timing accumulator. The loop filter's convergence and constant timing accumulator reliably indicate the achievement of symbol timing synchronization. The structure was finally implemented on a Virtex-5 FPGA having device utilization of 1% and maximum operating clock speed of 380MHz.

ACKNOWLEDGMENT

The research described in this paper is carried out in Center for Software Defined Radio at Aalborg University together with the industrial partner Thrane & Thrane A/S, Denmark. A special thanks goes to Jan Harding Thomsen and Bo Dyssegard from Thrane & Thrane A/S for their valuable discussions and suggestions throughout the system design.

REFERENCES

- [1] Chris Dick, fred harris, Michael Rice, "Synchronization in Software Radios - Carrier and Timing Recovery Using FPGAs", in *IEEE Symposium on Field-Programmable Custom Computing Machines*, 2000.
- [2] H. Meyr, M. Moeneclaey and S. A. Fechtal, "Digital Communication Receivers", John Wiley & Sons Inc., New York, 1998.
- [3] Joseph Gaeddert. et. al., "Multi-rate Synchronization of Digital Receivers in Software-Defined Radios" in *Proceeding of the SDR 07 Technical Conference and Product Exposition*, 2007.
- [4] F. J. Harris and M. Rice, "Multirate Digital Filters for Symbol Timing Synchronization in Software Defined Radios", *IEEE Journal on Selected Areas of Communications*, vol. 19, no. 12, December 2001.
- [5] Chris Dick, Benjamin Egg, fred harris, "Architecture and Simulation of Timing Synchronization Circuits for the FPGA Implementation of Narrowband Waveforms" in *Proceeding of the SDR 06 Technical Conference and Product Exposition*, 2006.
- [6] Fredric J. Harris, *Multirate Signal Processing for Communication Systems*, Prentice Hall, 2006.
- [7] Ljiljana Milic, *Multirate Filtering for Digital Signal Processing: MatLab Applications*, Information Science Reference (December 26, 2008) ISBN-13: 978-1605661780.
- [8] F. M. Gardner, "Phaselock Techniques", New York: Wiley, 1979.