Multimedia Tools and Applications (2019) 78:32679–32693
https://doi.org/10.1007/s11042-018-7038-6

High speed and low area decision feed-back equalizer with novel memory less distributed arithmetic filter

Grande NagaJyothi[1] · Sriadibhatla Sridevi[1]

Received: 15 April 2018 / Revised: 29 November 2018 / Accepted: 7 December 2018 /
Published online: 2 January 2019

Abstract

In this paper an efficient implementation of decision feed back equalizer (DFE) is carried out using novel memory less distributed arithmetic (NMLDA) filter. In wireless transmission systems, DFEs are used to mitigate the inter-symbol interference (ISI). The ISI is occurred due to multi-path propagation of the transmitted signal. High data rate systems demand higher order filters in DFE architectures which increase complexity in hardware design. In our proposed NMLDA design, we have used multiplexers and enhanced compressor adders in place of memory unit and conventional adders. The proposed design occupies lower area and gives higher throughput, when compared to MAC based filter and all other memory based DA filter architectures. By using proposed NMLDA based DFE, the ISI errors in transmission signal, will be minimized and the performance of the transmission system will be enhanced. We have synthesized the NMLDA of 32-tap, 16-tap, 8-tap and 4-tap filters and implemented them on FPGA device. The proposed design has nearly 70% less number of logical elements than OBC DA and 50% less than MDA and offers better throughput than the existing designs when implemented on Altera Cyclone III EP3C55F484C6.

Keywords Distributed arithmetic · Decision feedback equalizer · Non-linear equalizer · FIR · Feed forward (FF) filter · Feed back (FB) filter · Linear equalizer · Inter symbol interference (ISI) · Quantizer

1 Introduction

In telecommunication, a signal transmitted undergoes distortions due to multi-path propagation and band limited channels which causes the inter symbol interference (ISI) [15]. ISI causes severe effect in wireless channel and makes the signal communication less reliable.

✉ Grande NagaJyothi
grande.nagajyothi@vit.ac.in

Sriadibhatla Sridevi
sridevi@vit.ac.in

[1] Department of Micro and Nano-electronics, Vellore Institute of Technology, Vellore, 632014, India

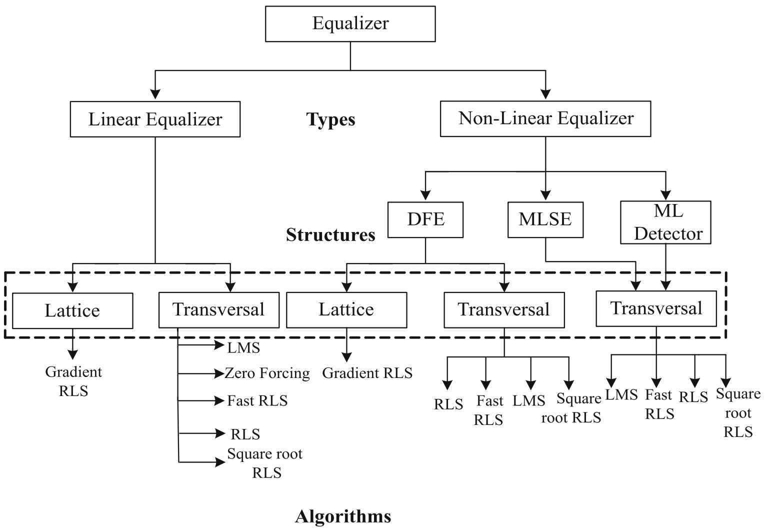

Fig. 1 Classification of equalizers

Equalizer is a device that attempts to nullify the distortion occurred by a transmitted signal through a channel. Equalizers [16] are placed at the receiver side to combat the ISI and to recover the transmitted signal. The classification of equalizers is detailed in Fig. 1. They are broadly classified as Linear equalizers and non-linear equalizers. Linear equalizers can eliminate the ISI, but enhances other noises which leads to poor signal performance. When the channel distortion is too severe and cannot be mitigated by linear equalizers, non-linear equalizers are used. The performance of non-linear equalizers is more effective to nullify the channel impairment. Decision feedback equalizer [2, 5, 16] is the non-linear equalizer which is effectively used as channel equalizer. DFE gives better signal to noise ratio when compared with linear equalizer by removing ISI and it exhibits less noise. The basic architecture of DFE is shown in Fig. 2. It consists of a decision device (quantizer), a feedforward (FF) filter, and a feedback(FB) filter.

The FF filter receives and equalizes the data with the transfer function of anti causal part of the channel and cancels pre-cursor ISI. The noise enhanced in DFE is significantly reduced. The FB filter suppresses the post-cursor ISI. The coefficients for the FF and FB filter should be carefully chosen to operate the DFE with zero errors. Untill the quantizer propagates zero value, the DFE channel will operate efficiently with less noise.

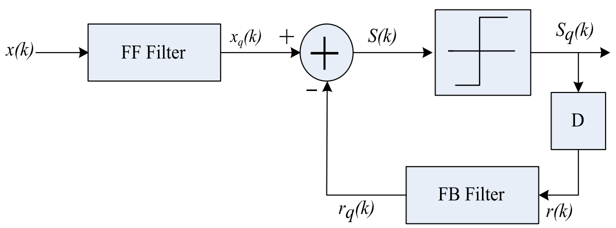

Fig. 2 Block diagram of decision feed-back equalizer

When data rate of the transmission system increases, the output of the DFE causes more symbols to overlap. To decrease this, we have to increase the order of FF and FB filter in DFE. Generally DFEs are designed using multiply accumulate units (MAC). When the filter order increases, the number of multipliers required by MAC unit also increases, which makes the hardware architecture more complex and the implementation of the DFE will become a challenging task. To reduce the area of DFE, MAC units are replaced with multiplier less architectures.

Many multiplier-less architectures have been reported in the literature [1, 4, 7, 9, 13, 20, 21], some of them have been proved efficient in certain conditions. Distributed Arithmetic (DA) architecture is one of the baseline efficient multiplier less architectures which pre-computes and stores the partial inner product of two vectors in a ROM/LUT. Generally the speed of MAC based multipliers depends on the length of the vector, but the speed of DA based architecture depends on the bit-length of the input vector. Hence DA based architectures (DAA) are faster than MAC based ones. DSP blocks such as FIR, IIR, DCT, FFT and adaptive filters can be implemented using DAA. The size of the memory in DA based FIR filter raises exponentially when the filter order is increased. To address this issue, many DAA [3, 12] have been proposed with less memory usage. In ref. [22], a modified DA (MDA) architecture was explained. In this, the LUT size is halved by replacing adder with adder/substractor. Further the LUT size of DA is reduced by introducing offset binary code (OBC) concept. DAA was constructed without the usage of memory. In [6], the memory less architecture was developed by replacing multiplexers and adders instead of LUT/ROM. Several applications like hearing-aids, software defined radio, channel equalizers utilize the DAA. The works in [8, 19] proposed DAA for efficient base band processing in software receiver application. The authors in [17, 18] have proposed block floating point (BFP) approach for adaptive DFE. The performance cost of BFP is more when observed with the fixed point approach. Basic DA based DFE and OBC DA based DFE architectures are described in [10, 11, 14]. These architectures occupy higher area due to usage of ROM/LUT.

In this paper, we propose a novel memory less Distributed Arithmetic (NMLDA) filter and developed DFE by using the proposed NMLDA, which occupies less area and offers high speed when compared to the existing DFE architectures.

The remnant of the paper is organized as follows. Section 2 comprises of the mathematical calculations of the DA. The proposed NMLDA architecture is described in Section 3. Section 4 emphasizes on the implementation of DFE with NMLDA architecture. We provide the synthesis results of existing and proposed design in Section 5. Finally conclusion encompasses in Section 6.

2 Background of distributed arithmetic

The bit serial multiplication operation of DA can be performed with in single direct step. Let us consider x_k and d_k to be the input and fixed filter coefficient vectors with K number of filter input words. Multiplication of vectors can be written as

$$y = \sum_{k=1}^{K} d_k x_k \tag{1}$$

Where x_k is written as signed $2's$ complement binary number and $|x_k| < 1$, then x_k can be expressed as:

$$x_k = -b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n} \tag{2}$$

Where $x_k = b_{k0}, b_{k1}, ...b_{k(N-1)}$ and b_{kn} has (0, 1) values. On substituting x_k in y and rearranging the summation order, we get finally

$$y = \sum_{k=1}^{K} d_k(-b_{k0}) + \sum_{n=1}^{N} \left[\sum_{k=1}^{K} d_k b_{kn} \right] 2^{-n} \tag{3}$$

Equation (3) provides us DA form. The modify form of above equation is:

$$y = \sum_{l=1}^{L-1} \left[2^{-l} C_l - C_0 \right] \tag{4}$$

Where

$$C_l = \sum_{k=0}^{K-1} (d_k b_{kn}) \tag{5}$$

C_l is the pre-computed partial inner product value which is stored in the memory. It has 2^k possible combination values and contains $2 * 2^k$ size of LUT. The LUT size will be reduced from $(2 * 2^k)$ to 2^k by using MDA design. Further the LUT size is reduced to 2^{k-1} by having OBC concept in DA architecture. When filter order increases, the LUT size of DA increases and hence memory less DA architecture (MLDA) is developed to reduce the area occupancy. The memory units are replaced with multiplexers in MLDA architecture. In the paper, we further extended MLDA design to decrease the area when filter order increases.

3 Proposed novel memory-less DA filter architecture

Our proposed Novel memory-less DA (NMLDA) filter architecture is explained in this section. It consists of 2:1 multiplexers instead of memory elements and the adders of memory less DAA [6] are replaced with enhanced 4:2 compressor as shown in Fig. 3. It consists of serial in parallel out shift register (SIPOSR), four 2:1 multiplexers, enhanced 4:2 compressor adders and shift accumulators. The input data x_k is fed to the SIPOSR and the outputs from shift registers will act as selection lines for four 2:1 multiplexers. One of the inputs of the 2:1 multiplexer is the filter coefficient and the other input is logic '0'. If the selection input line is high, then filter coefficient will present at output else the output of multiplexer is zero. The output from MUXs are A_1, A_2, A_3, A_4 which are given to the enhanced 4:2 compressor adder to get final output.

The enhanced 4:2 compressor adder is designed with dual mode logic (DML) and is shown in Fig. 4. DML logic consists of XOR/XNOR module and MUX module. The XOR/XNOR module is developed with CMOS logic and MUX module is developed using transmission logic gate (TG). By using DML realization, we can achieve better results in area and speed. The outputs from MUXs A_1, A_2, A_3 and A_4 are given to enhanced 4:2 compressor adder. A_5 (C_{in}) is the fifth input to the compressor which is the C_{out} of the

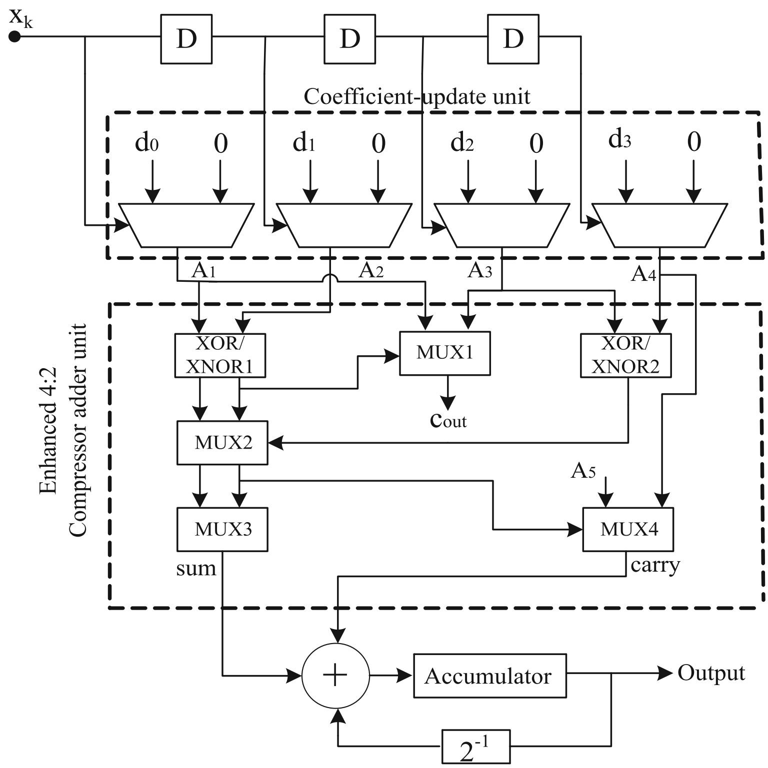

Fig. 3 Block diagram of 4-tap NMLDA filter

previous stage compressor. The four inputs A_1, A_2, A_3, A_4 and sum output will have same weights. The A_1 and A_2 inputs are fed to XOR/XNOR1 module and A_3 and A_4 are fed to XOR/XNOR2 module. The outputs from XOR/XNOR modules are given to the MUX1 module with A_4 as a selection line. The outputs from MUX1 module are given as inputs to the MUX2 module with A_5 as selection line to generate sum. To achieve carry, A_4 and A_5 are given to MUX4 with selection line as one of the output of MUX2. By using DML logic, the 4:2 compressor adder provides the outputs as follows:

$$\begin{aligned} Sum = & (((\overline{(A_3 \oplus A_4)} * (A_1 \oplus A_2)) + \bar{A}_5 * (A_3 \oplus A_4) * (\overline{(A_1 \oplus A_2)}) \\ & + \overline{(((\overline{(A_3 \oplus A_4)} * (A_1 \oplus A_2)) + ((A_3 \oplus A_4) * \overline{(A_1 \oplus A_2)}))} * A_5 \end{aligned} \tag{6}$$

$$Carry = ((\overline{(A_1 \oplus A_2 \oplus A_3 \oplus A_4)} * A_4) + (A_1 \oplus A_2 \oplus A_3 \oplus A_4) * A_5 \tag{7}$$

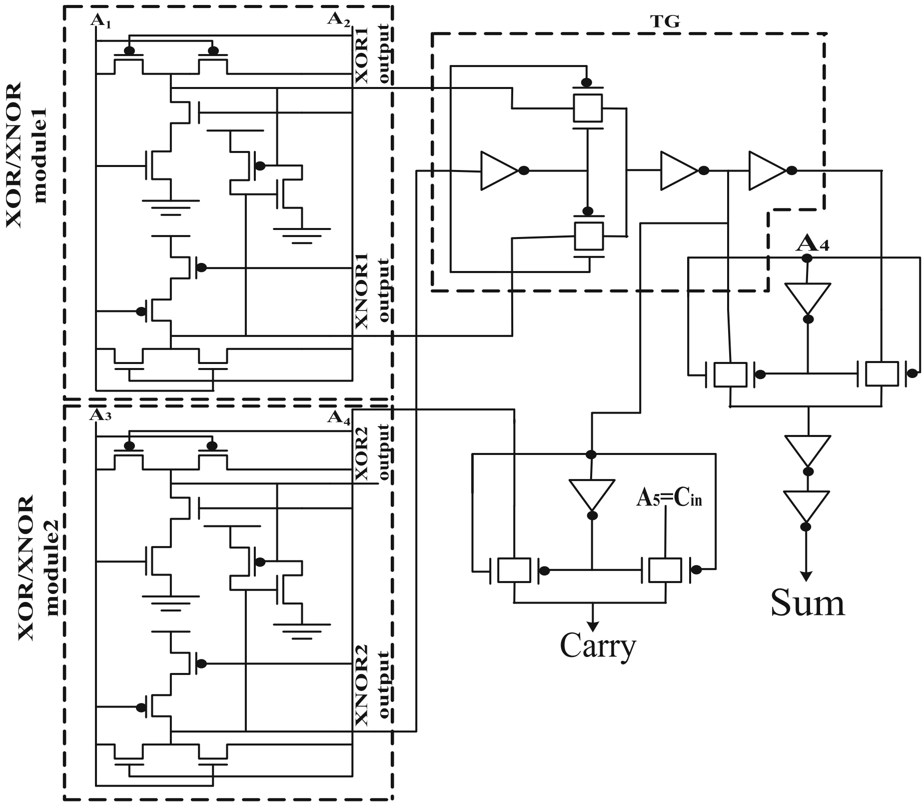

Fig. 4 Block diagram of enhanced compressor adder with DML logic

$$C_{out} = \overline{(A_1 \oplus A_2)} * A_1 + (A_1 \oplus A_2) * A_3 \tag{8}$$

The outputs sum and carry are given to shift and accumulator unit to get final result.

4 Decision feedback equalizer with proposed NMLDA

Let us consider, decision feed back equalizer shown in Fig. 2, with input signal $x(k)$, where $k \in Z$ with N_f number of FF filter coefficients and feedback output decision $r(k)$ with N_b number of FB filter coefficients. The output generated decision S_{qk} for DFE is given as follows:

$$S_{qk} = Q[S(k)] \tag{9}$$

where Q[.] represents the quantization operation.

$$S(k) = x(\hat{k}) - r(\hat{k}) \tag{10}$$

$$r(k) = S_q(k-1) \tag{11}$$

$$\hat{x(k)} = \sum_{i=0}^{N_f-1} d_i x(k-i) = d^T x(k) \tag{12}$$

$$\hat{r(k)} = \sum_{j=0}^{N_b-1} b_j r(k-j) = b^T r(k) \tag{13}$$

where

$$d^T = [d_0, d_1, d_2, ...d_{N_f-1}] \tag{14}$$

$$b^T = [b_0, b_1, b_2, ...b_{N_b-1}] \tag{15}$$

The coefficient of FF filter is d^T and the coefficient of FB filter is b^T. N_b and N_f are number of FB and FF filter coefficients respectively. The $2'$s complementary form of $x(k)$ and $r(k)$ with W word length can be expressed as

$$x(k-i) = \sum_{w=1}^{W-1} x_{i,W-1-i} 2^{-i} - x_{i,W-1} \tag{16}$$

$$r(k-j) = \sum_{w=1}^{W-1} r_{j,W-1-j} 2^{-j} - r_{j,W-1} \tag{17}$$

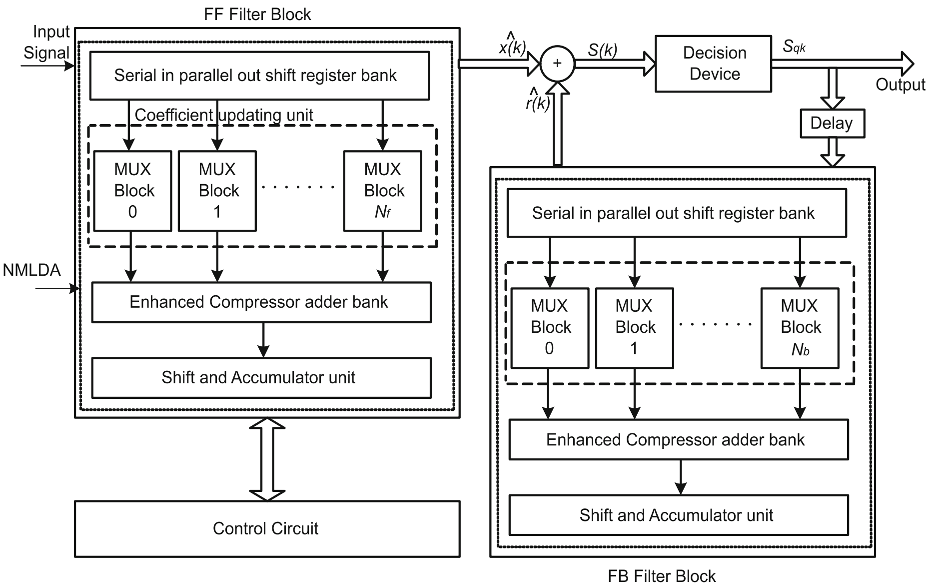

Fig. 5 Decision feedback equalizer with NMLDA

Table 1 Performance comparison of no. of logical elements for existing and proposed NMLDA architectures implemented on cyclone III EP3C55F484C6 FPGA device

Taps	MAC	MDA	OBC DA	Memory less DA	Proposed NMLDA
32-tap	3312	2482	3376	1420	1309
16-tap	1775	1247	1708	480	401
8-tap	1007	646	891	282	178
4-tap	420	337	481	159	108

On substituting Eqs. (16) and (17) in Eqs. (12) and (13) respectively and again substituting in Eq. (10) finally we get:

$$\begin{aligned} S(k) = & \left[\sum_{i=0}^{N_f-1} -d_i x_{i,W-1} + \sum_{w=1}^{W-1}\sum_{i=0}^{N_f-1} x_{i,W-1-i}2^{-i}\right] \\ & - \left[\sum_{j=0}^{N_b-1} -b_j r_{j,W-1} + \sum_{w=1}^{W-1}\sum_{j=0}^{N_b-1} r_{j,W-1-j}2^{-j}\right] \end{aligned} \tag{18}$$

DFE architecture with proposed NMLDA is shown in Fig. 5. DFE consists of FF filter block, control circuit, FB filter block and decision device. Both FF and FB filter blocks are designed by using proposed NMLDA architecture. It consists of serial in parallel out shift register bank, a block of multiplexers, enhanced compressor adder bank and shift-accumulator block. In FF filter block, the input from the SIPO register bank are given as a selection line to the multiplexer and outputs from multiplexers are passed to the enhanced 4:2 compressor adder, from there it is given to the shift-accumulator to compute the output $x(\hat{k})$. Similar operation will be performed in FB filter block and the output $r(\hat{k})$ is achieved. The difference of the outputs of FF and FB filter blocks is $S(k)$ which is given to the decision device. The output $S(k)$ is checked by the decision device whether the signal lies with in

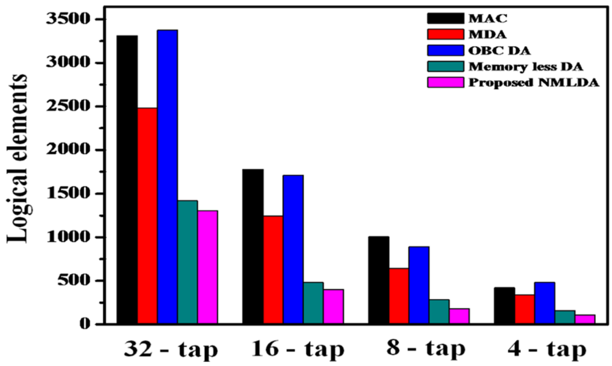

Fig. 6 Logical elements for proposed and existing DA architectures

Table 2 Performance comparison of f_{max} (MHZ) for existing and proposed NMLDA architectures implemented on cyclone III EP3C55F484C6 FPGA device

Taps	MAC	MDA	OBC DA	Memory less DA	Proposed NMLDA
32-tap	7.21	9.06	8.7	30	41.6
16-tap	14.38	17.44	16.64	47	78
8-tap	28.13	32.77	27.71	68	93.6
4-tap	56.47	60.66	46.12	72	103

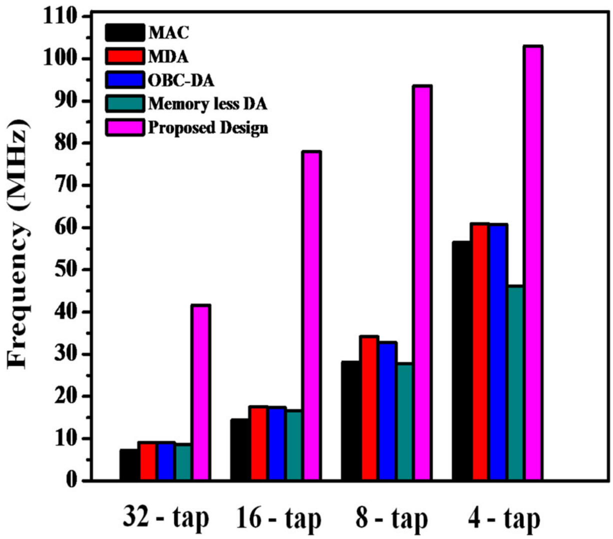

Fig. 7 Frequency response for proposed and existing DA architectures

Table 3 Performance comparison of static power consumption (mW) for existing and proposed NMLDA architectures implemented on cyclone III EP3C55F484C6 FPGA device

Taps	MAC	MDA	OBC DA	Memory less DA	Proposed NMLDA
32-tap	51.80	94.14	94.19	94.11	94.11
16-tap	51.74	94.07	94.09	94.01	94.04
8-tap	51.71	94.07	94.05	94.02	94.02
4-tap	51.70	94.02	94.02	94.01	94.01

the range of signal or not and quantizes the signal according to the modulated scheme. The process will be continued untill DFE gets zero error.

5 Results and discussion

We have validated the proposed design explained so for by performing simulation. We simulated 4,8,16 and 32 taps of NMLDA filter architecture and implemented them on FPGA. The number of logical elements, static power consumption and maximum sampling frequency are obtained by using Altera cyclone III . The results of proposed architecture are compared with MAC based filter, the OBC DA based filter, MDA filter and memory-less DA based filter and are tabulated.

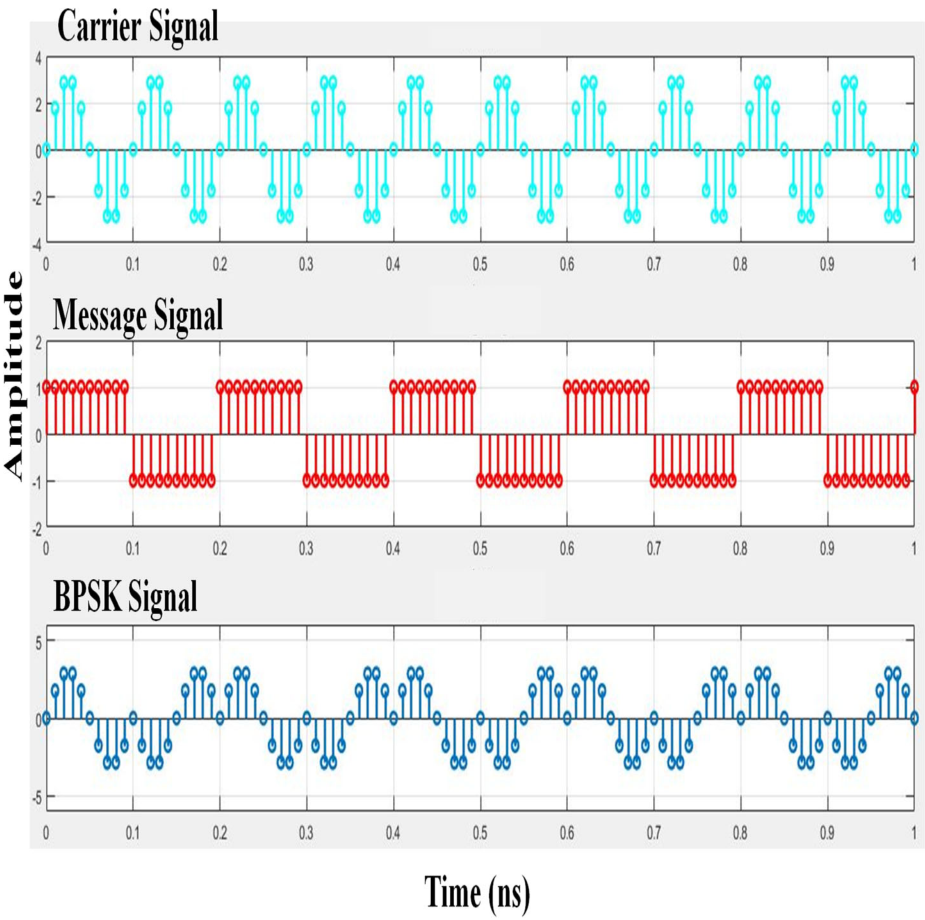

Fig. 8 Waveforms of proposed DFE based channel equalizer system using BPSK

From the results shown in Table 1, we analyzed that the number of logical elements in proposed architecture is very less when compared to MAC architecture and other memory based DA architectures. In proposed 4-tap FF filter architecture,the OBC technique and MAC design have same number of logical elements but when the filter order increases, there is large variation in their logical elements. Also the number of logical elements for proposed design is less when compared to the memory less architecture [6] and is shown in Fig. 6. It is observed from Table 2 that the maximum frequency of the proposed NMLDA is very high when compared with all other architectures and observed in Fig. 7. From Table 3, we observe that static power consumption of NMLDA is almost same as the with the memory based DA architectures.

Critical path delay is the longest delay path between any two registers in a system. The critical path delay of MAC based FB filter is $C_{MAC} = T_{mul} + NT_{add}$, where T_{mul}= Computation time of multiplier, T_{add} = Computation time of adder. Where 'N' is the positive number. Critical path delay for MAC and MDA based architectures are almost same for lower order filters. The critical path delay of MDA scheme is C_{MDA}= T_{memory} + NT_{add} + T_{mux}. The critical path delay of OBC DA based design is $C_{OBC} = T_{OBC} + T_{mux} + (N + 1)T_{add}$. The critical path delay for OBC based DA is higher when compared with all other architectures, because EXOR gates are additionally added. The critical path delay for the memory-less DA (MLDA) filter design is given as C_{MLDA}=$T_{mux} + T_{add}$. The critical path delay for the proposed NMLDA design is given as C_{NMLDA}=$T_{mux} + T_{compressoradder}$. As the critical path delay for the proposed design is less it exhibits high throughput.

The proposed DFE architecture with NMLDA has been tested by channel equalizer system with binary phase shift keying (BPSK) and frequency shift keying (FSK) modulated

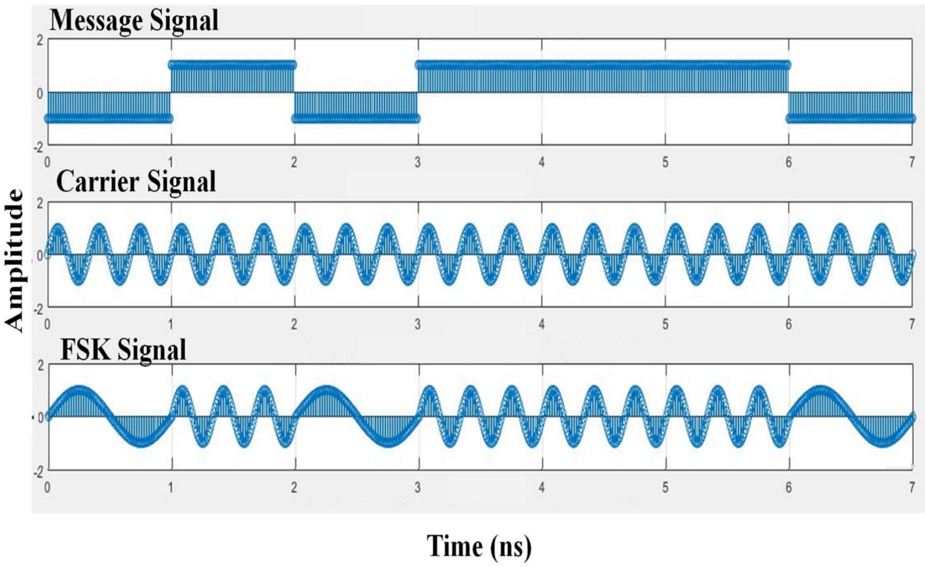

Fig. 9 Waveforms of proposed DFE based channel equalizer system using FSK

techniques. Figures 8 and 9 depict the output responses of proposed DFE based channel equalizer with BPSK and FSK signals respectively. Let us consider that a set of channel impulse response signals are modulated with carrier signal using BPSK and FSK techniques and are transmitted. At receiving side of the channel equalizer, the received signal is generated with noise and ISI type of errors. For removing noise and ISI errors in the generated signal, the signals are passed to the DFE. In DFE, the FF filter block removes the pre-cursor anti-causal part of ISI and the rest of the noise and errors are removed in the FB filter block. The DFE channel will operate untill the decision device of the DFE propagates zero value. Finally, the original signal output is obtained with out errors in DFE. The original noise signal and filtered ISI free signal for BPSK and FSK are shown in Figs. 10 and 11 respectively.

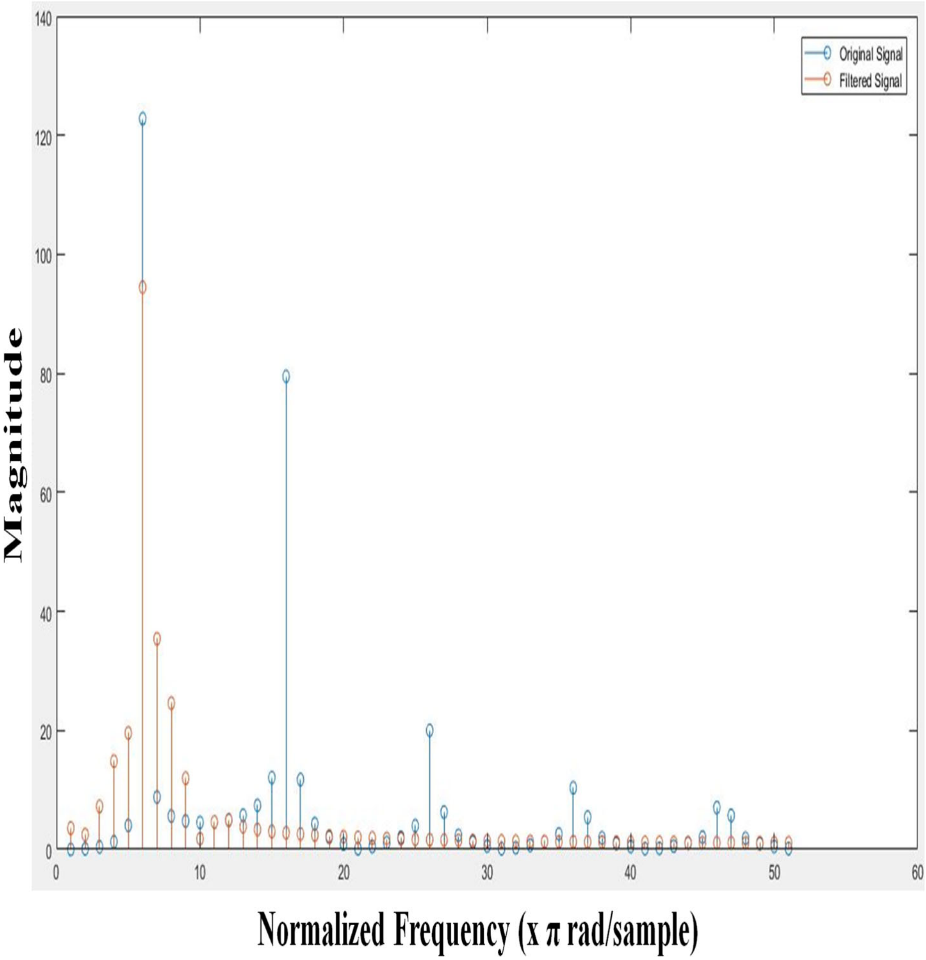

Fig. 10 Original noise signal and filtered ISI free signal of BPSK

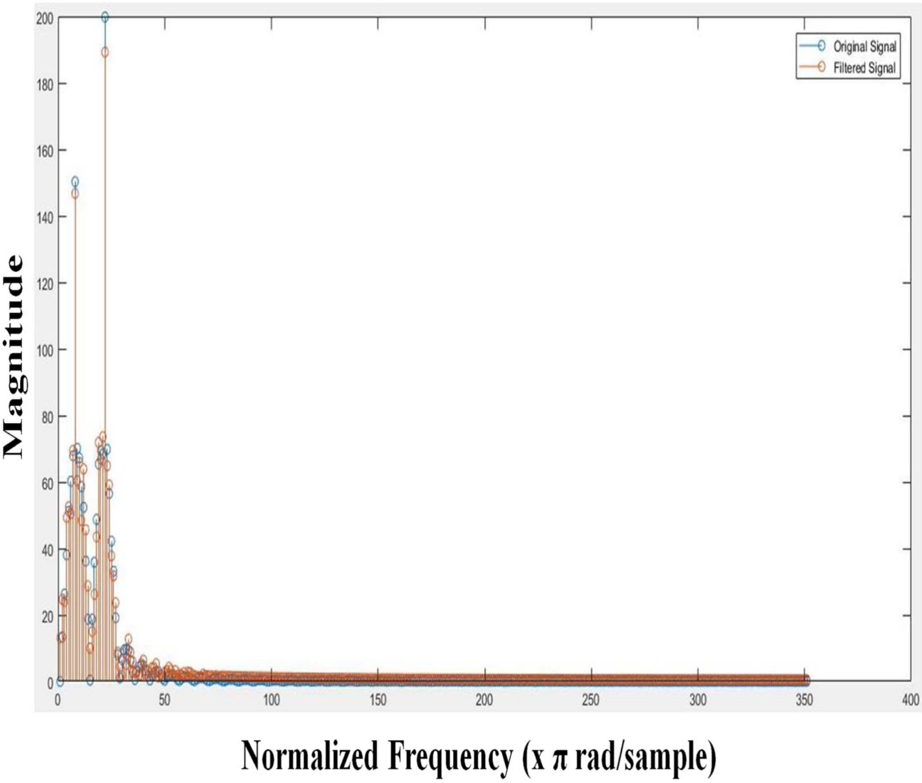

Fig. 11 Original noise signal and filtered ISI free signal of FSK

6 Conclusion

In this paper, we proposed NMLDA based DFE. By using NMLDA architecture the number of logical elements used is reduced when compared with MAC architecture and memory based DA architectures. By using novel enhanced 4:2 compressor adder in the proposed architecture, the number of adders used is reduced. So the critical path of the design is reduced, which causes increase in speed. By using NMLDA based DFE, the ISI errors in the transmitted signal can be minimized and the original signal that is transmitted can be obtained without errors. The proposed architecture becomes extremely helpful for high speed data rate modems.

References

1. Burleson WP, Scharf LL (1991) A vlsi design methodology for distributed arithmetic. Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology 2(4):235–252
2. Callender C, Theodoridis S, Cowan C (1994) Adaptive non-linear equalisation of digital communications channels. Signal Process 40(2-3):325–333

3. Chen HC, Guo JI, Jen CW, Chang TS (2005) Distributed arithmetic realisation of cyclic convolution and its dft application. IEE Proceedings-Circuits Devices and Systems 152(6):615–629
4. Croisier A, Esteban D, Levilion M, Riso V (1973) Digital filter for pcm encoded signals. US Patent 3,777,130
5. Douillard C, Jézéquel M, Berrou C, Electronique D, Picart A, Didier P, Glavieux A (1995) Iterative correction of intersymbol interference: turbo-equalization. Trans Emerg Telecommun Technol 6(5):507–511
6. Ghamkhari SF, MBGG (2012) A low-power, low-area architecture design for distributed arithmetic (da) unit. In: 20th Iranian conference on electrical engineering, vol 1. IEEE, pp i–i
7. Grande NJ, Sridevi S (2017) Asic implementation of shared lut based distributed arithmetic in fir filter. In: 2017 international conference on microelectronic devices, circuits and systems (ICMDCS). IEEE, pp 1–4
8. Hurskainen H, Raasakka J, Ahonen T, Nurmi J (2009) Multicore software-defined radio architecture for gnss receiver signal processing. EURASIP Journal on Embedded Systems 2009(1):543–720
9. Jyothi GN, Sriadibhatla S (2019) Asic implementation of low power, area efficient adaptive fir filter using pipelined da. In: Microelectronics, electromagnetics and telecommunications. Springer, pp 385–394
10. Khan MT, Ahamed SR (2016) Low cost implementation of concurrent decision feedback equalizer using distributed arithmetic. In: 2016 1st India international conference on information processing (IICIP). IEEE, pp 1–5
11. Khan MT, Ahamed SR, Chatterjee A (2016) Efficient implementation of concurrent lookahead decision feedback equalizer using offset binary coding. In: 2016 20th international symposium on VLSI design and test (VDAT). IEEE, pp 1–6
12. Meher PK (2006) Hardware-efficient systolization of da-based calculation of finite digital convolution. IEEE Trans Circuits Syst Express Briefs 53(8):707–711
13. NagaJyothi G, SriDevi S (2017) Distributed arithmetic architectures for fir filters-a comparative review. In: 2017 international conference on wireless communications, signal processing and networking (wiSPNET). IEEE, pp 2684–2690
14. Prakash MS, Shaik R, Koorapati S (2016) An efficient distributed arithmetic-based realization of the decision feedback equalizer. Circuits Systems Signal Process 35(2):603–618
15. Proakis JG (2001) Digital signal processing: principles algorithms and applications. Pearson Education India
16. Qureshi SU (1985) Adaptive equalization. Proc IEEE 73(9):1349–1387
17. Shaik R, Chakraborty M (2006) An efficient realization of the decision feedback equalizer using block floating point arithmetic. In: IEEE Asia Pacific conference on circuits and systems, 2006. APCCAS 2006. IEEE, pp 1047–1050
18. Shaik R, Chakraborty M (2013) A block floating point treatment to finite precision realization of the adaptive decision feedback equalizer. Signal Process 93(5):1162–1171
19. Waelchli G, Baracchi-Frei M, Botteron C, Farine PA (2010) Distributed arithmetic for efficient base-band processing in real-time gnss software receivers. J Electr Comput Eng 2010:2
20. White S (1986) High-speed distributed-arithmetic realization of a second-order normal-form digital filter. IEEE Transactions on Circuits and Systems 33(10):1036–1038
21. White SA (1989) Applications of distributed arithmetic to digital signal processing: a tutorial review. IEEE ASSP Mag 6(3):4–19
22. Yoo H, Anderson DV (2005) Hardware-efficient distributed arithmetic architecture for high-order digital filters. In: IEEE international conference on acoustics, speech, and signal processing, proceedings, vol 5. (ICASSP'05), pp v–125

Grande NagaJyothi received her B.Tech degree in Electronic and Electrical Engineering from Jawaharlal Nehru Technology University, Andhra Pradesh, India, in 2006, and her M.Tech degree from VIT University, Vellore, India, in 2008. She worked as assistant professor in Electronics and Communication Engineering Department of Sri Sai Institute of Technology and Science, Kadapa, Andhra Pradesh, from September 2008 to 2014. Currently, she is working towards her Ph.D degree in VLSI System Design at VIT University, Vellore, India. Her research interest includes Low power IC design and VLSI DSP.

Sriadibhatla Sridevi received the B.Tech and M.Tech degrees from Jawaharlal Nehru Technological University Andhra Pradesh, India in 1999 and 2005 respectively and Ph.D degree from Andhra University in 2014. In 2005, she joined in the department of Electronics and Communication Engineering in Sri Prakash College of Engineering, Tune, and Andhra Pradesh as Assistant professor and later superannuated as Associate professor .Presently she is working as Associate professor in VIT University; Vellore. S. SriDevi is a life member in ISOI and a member in IEEE and IEEE CAS. Her research interests include VLSI DSP and Low power digital IC design.