

Study of Data Flow Graph of Finite Impulse Response Digital Filter

Mandeep Kaur¹, Ranjit Kaur², Damanpreet Singh³

¹Department of Electronics and Communication Engineering, Punjabi University, Patiala, India

²Department of Electronics and Communication Engineering, Punjabi University, Patiala, India

³SLIET Longowal, Punjab, India

Abstract: Processing of signals requires designing and implementation of systems called Filters. Filter design depend on its type, structure and the method to which it comes in practice. This paper discussed the various realization of Digital Finite-Duration Impulse Response (FIR) filters with its graphical representation. The purpose of using the realization and the graphical representation is to find the critical and shortest path using different algorithm. The realization can be done using FDA tool or coding in MATLAB. The graphical representation way is done in form of Data Flow Graph in term of nodes and edges.

I. Introduction

FIR Filters are one type of digital filters used in Digital Signal Processing (DSP). FIR Filters are also named as non- recursive type of filters . It can be implemented easily and efficiently. Finite Impulse Response filters are named as finite because they do not have any feedback [1, 5]. A FIR filter has no feedback and its input-output relation is given by

$$y(n) = \sum_{k=0}^M p(k)x(n - k)$$

The output of a non recursive filter is a function only of the input signal. The response of such a filter to an impulse consists of a finite sequence of $(M+1)$ samples, where M is the order of filter. Hence, the filter is known as a Finite-Duration Impulse Response (FIR) filter [5] .These FIR filters are simple to design and are guaranteed to be bounded input-bounded output (BIBO) stable. These have linear phase which is a desirable property for many applications such as audio and video processing.

In this work, FIR Filter structures such as Direct form , Transpose Direct form , Cascade form , Parallel form and the Lattice structure has been implemented in MATLAB / Simulink. Further, these structure are represented in graphical form. Graphical forms are efficient for investigating and analyzing the properties of DSP algorithms[3]. These representation can be used to map DSP algorithms to hardware implementations. These various types of graphical forms are Data Flow Graph, Signal Flow Graph, Dependence Graph. In this paper, Data flow graph are preferred.

Data flow graphs (DFG) are the powerful representation of digital filters which are a part of signal processing algorithms. DFG consists of nodes and arcs. Nodes in the DFGs represent multiplication or addition in filters. They can be represented as:

$$G = (V; E; d; w)$$

where V represents nodes such as adders or multiplier elements in digital filters, E is the set of directed edges that represent communication between the nodes, d is the computation time of the node and w is the weight in terms of delays on the edge.

II. Method

MATLAB offers DSP toolbox using which can easily design the required digital filter and can observe its characteristics and construct realization structure of the designed filter. The FIR filter structures are realized in MATLAB by following steps:

1. By FDA tool, figure 1 opens.
First specify the filter specification then click on realize model icon. After that, specify input processing block name and select the build the block using basic elements. Another way to realize the model, first specify the filter specification then from file click on export to Simulink model. This generated a block in Simulink. By double click, structure of FIR filter will appear. By using convert icon, it can change to different structures. But it is limited up to direct form and its transpose form.
2. These structure can also be realized using the coding with 'dfilt' object. By writing the coding, it is very easy to realize the different structures of FIR filters.
After realization of the structure, the DFG is constructed from the filter block diagram based on the design under consideration. DFGs can also be represented in the Node-weight matrix and Incidence Matrix. [6]. Node-weight Matrix corresponds node weights based on computation time. Incidence Matrix define the edge weights between all the nodes.



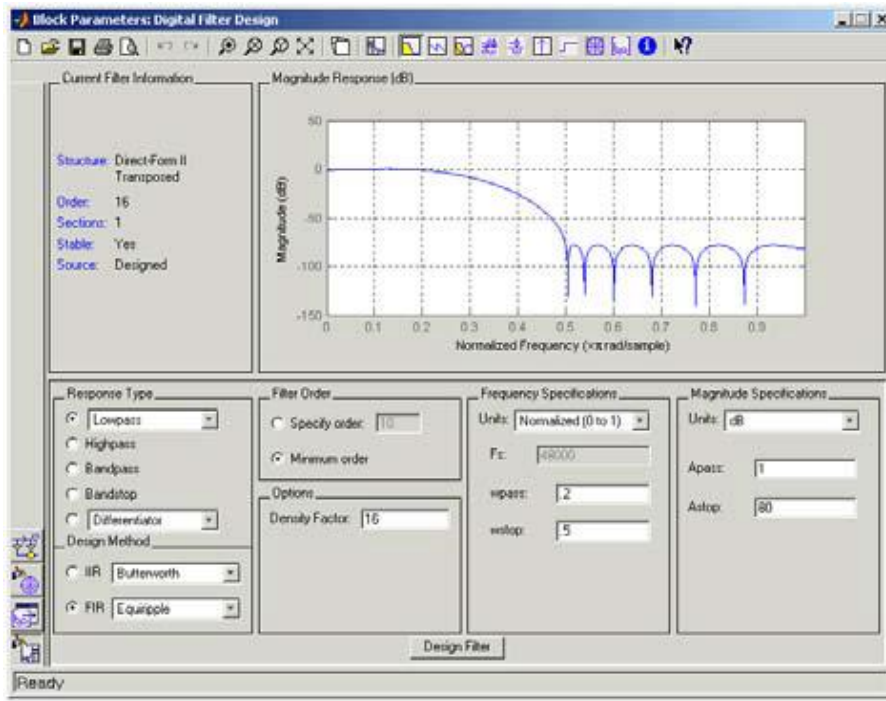


Fig 1: The Fdatool GUI for implementing Digital Filter structures.

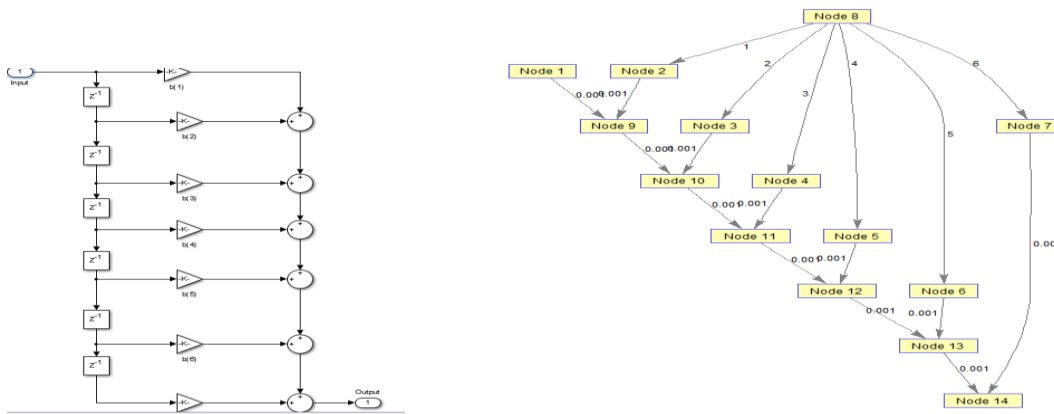


Fig 2: 6th order FIR filter block diagram and the DFG

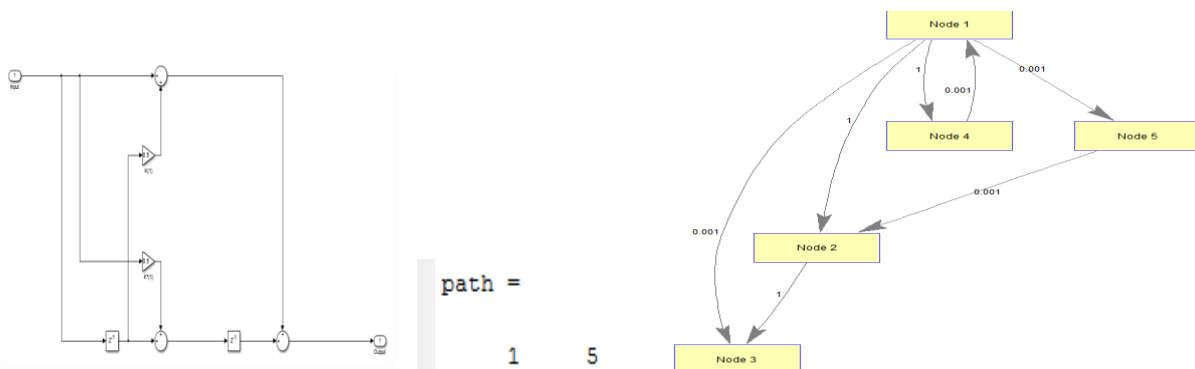


Fig3: Single-Stage Lattice Realization of FIR filter with DFG

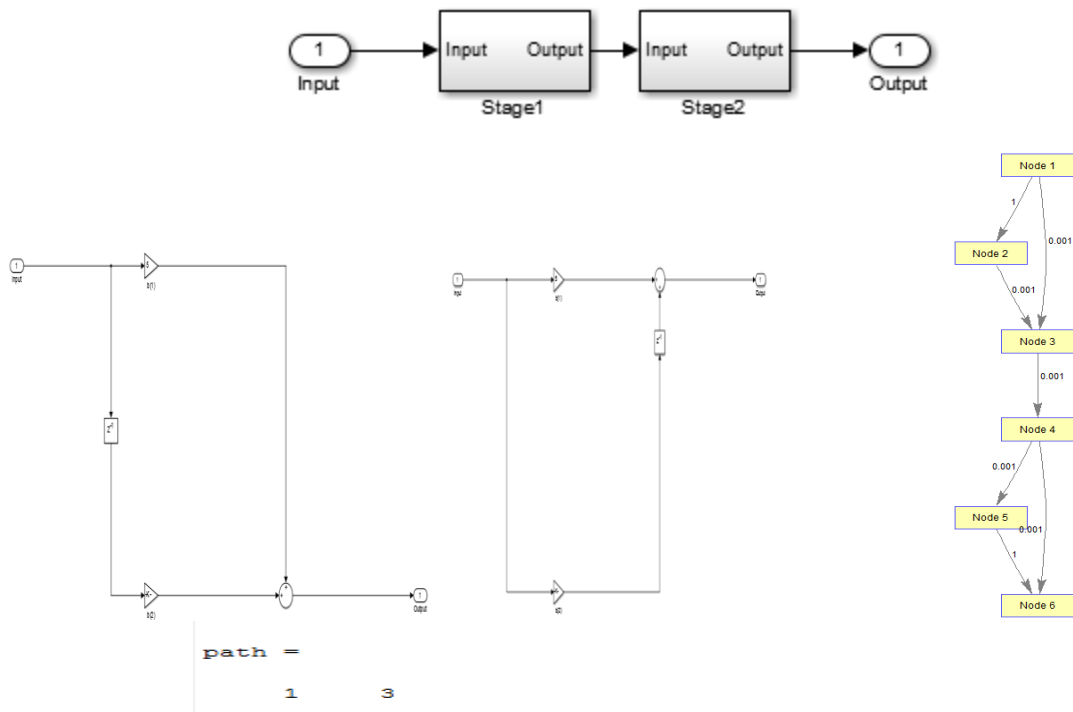


Fig 4: Cascade structure of FIR filter and its DFG

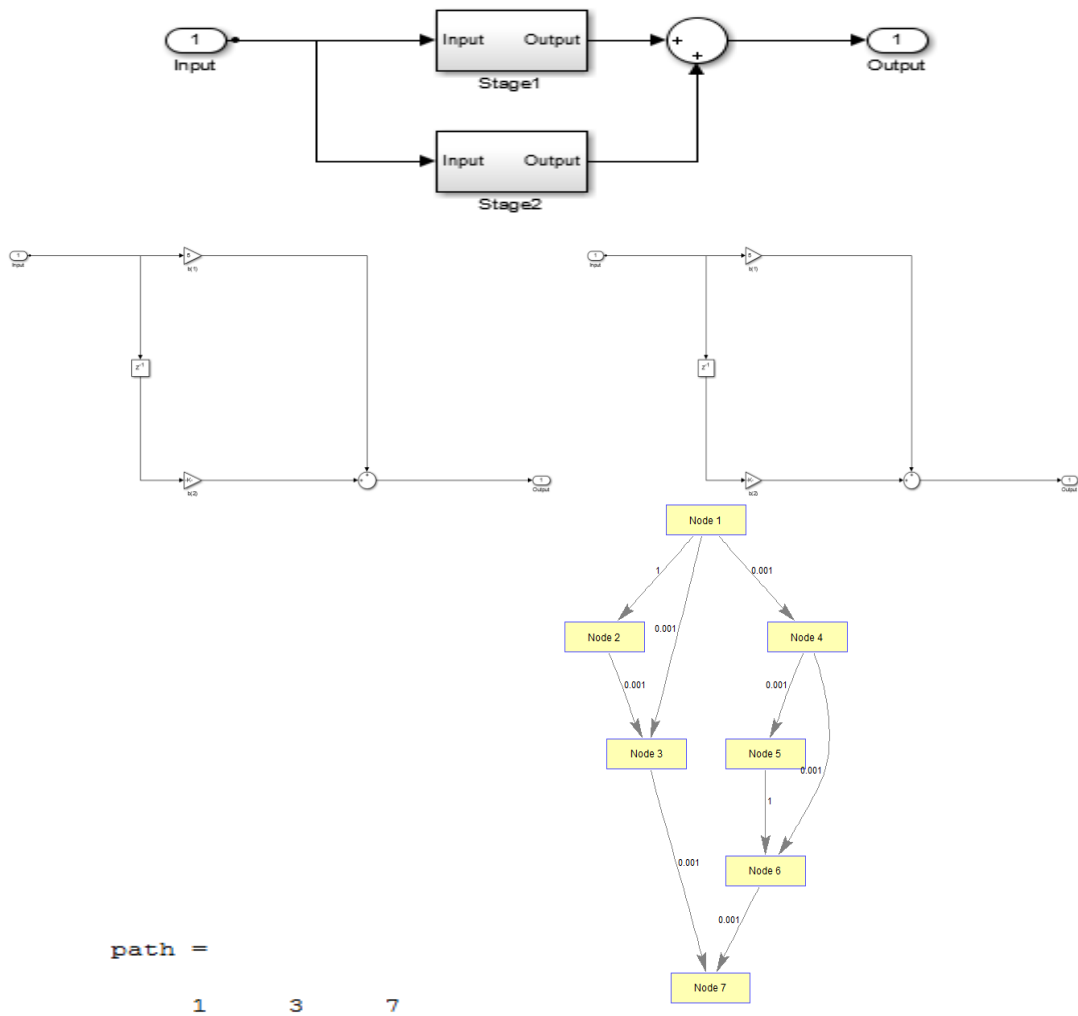


Fig 5: Parallel structure of FIR filter and its DFG representation



III. Conclusion

The paper presented the FIR digital filter realization and its graphical representation. The DFG of the FIR filters shows the connection between nodes which gives an idea to find the shortest path. The DFG representation and the realization of FIR filters will be further discussed in term of complexity, critical path and shortest path. As further continuation to this work, these DFG can be realised on various transformation techniques which can enhanced the performances of the digital filters. Various transformation techniques such as retiming, folding, unfolding, pipelining are employed for minimizing the number of functional blocks in synthesizing DSP architecture. These transformation techniques results in speed enhancement by reducing the critical path in most DSP systems.

References

- [1] Chonghua Li, "Design and Realization of FIR Digital Filters Based on MATLAB", International Conference on Anti-Counterfeiting Security and Identification in Communication (ASID), IEEE, 2010 , pp: 101 – 104.
- [2] Yu-Chi Tsao and Ken Choi, "Area-Efficient Parallel FIR Digital Filter Structures for Symmetric Convolutions Based on Fast FIR Algorithm", IEEE Transcation on VLSI Systems, Feb 2012, Vol. 20, Issue. 2, pp: 366 – 371
- [3] Yagain, D.; Vijaya, K.A., "A Novel Framework for Retiming Using Evolutionary Algorithm for HLS of Digital Filters ", IEEE Transcation on Swarm and Evolutionary Computation , Vol 20, February 2015, pp: 37–47
- [4] K. Mandeep, K. Ranjit , "Review of Retiming based Digital Filter for Low Power Consumption ", International Conference on Recent Trends in Computing, Aug 30-31 , 2016
- [5] K.K.Parhi , *VLSI Digital Signal Processing: Design and Implementation*, Wiley India , 2007
- [6] Mandeep Kaur , " Analysis of various algorithms for the low power consumption in embedded systems using different architecture.", International Journal of Electronics Engineering, 2(1), 2010, pp. 213-217

