# FPGA IMPLEMENTATION OF PSM BASED FIR FILTERS WITH INTERPOLATOR AND DECIMATOR

**G.Srisatyavani[1], B.Sunith Kumar[2]**

[1] *ECE Dept, DRKIST, (India)*

[2] *Asst. Professor, ECE Dept, DRKIST, (India)*

## ABSTRACT

*In order to achieve computational efficiencies multi-rate digital signal processing is used to convert the sampling rates. . In multi-rate systems, decimation and interpolation filters are the most important building blocks. In this Paper A new architecture, namely, Programmable Shift Method(PSM) based FIR Filter design was presented in which Processing Element(PE) performs the coefficient multiplication operation with the help of a shift and add unit for reducing the complexity and have been proposed and implemented on Xilinx FPGA device. By combining this FIR Filters with Interpolator and Decimator, the FIR filter can be operated with multiple sampling rates and complexity of Interpolator and Decimator filters can also be reduced. Power dissipation is recognized as a critical parameter in modern VLSI design field.We expect that the proposed FIR filter architecture with Processing Element performing the coefficient multiplication with the help of shift and add unit can be adapted to various fields requiring multiple sampling rates such as the signal processing areas.*

***Key Words: Interpolator, Decimator, PSM, Poly Phase Interpolation and Polyphase Decimation.***

## I INTRODUCTION

Recently, there has been rapid progress in the area of multirate digital signal processing. The applications of multirate systems include subband coding of video, audio, and speech signals, fast transforms using digital filter banks, wavelet analysis of all types of signals, and many other fields [l]. In multirate systems, decimation and interpolation filters are the most important building blocks. **A** great amount of literature deals with the theory and design of decimation and interpolation filters**.** However, issues concerning the VLSI implementation scheme for multirate filters have not been investigated thoroughly**.** Since the speed of processing time and the silicon area are the crucial factors in the VLSI implementation] a scalable implementation scheme to flexibly and efficiently implement the multirate FIR filters is presented in this paper.

One of the most important applications of multirate systems is subband coding **(SBC).** Since it was introduced by Crochiere *et al.* in 1976, subband coding is so far one of the most effective coding approaches for video and audio applications. Because subband coding needs filter banks to split the input signal (see Fig. l), aside from the coding mechanism, the filter banks are the kernel of this coding scheme. In the past, various types of subband filter banks have been proposed, such as quadrature mirror filters (QMF) **,** conjugate quadrature filters (CQF) **,** and symmetric short kernel filters (SSKF) , etc. For subband coding systems,, implementation of their filter banks is the most important task. Since filter banks usually deal with large amount of data, high speed computing hardware **is** indispensable. In order to achieve both high performance and low complexity, by employing the scalable implementation scheme, we propose an efficient design technique suitable for all types of subband filter banks.

In Digital Signal Processing System sometimes it becomes necessary to convert the data to a new rate to make it easier to process or to achieve compatibility with another system. Therefore Multirate Signal Processing is used which is defined as the discrete time system that process data at more than one sampling rate to perform the desire digital operations.

In this work, in general we are going to present a systematic approach for the low-power design of a general linear time-invariant (LTI) FIR/IIR system based on the multirate approach. The direct implementation of the system transfer function ***H ( z )*** (see Fig. l) has the constraint that the speed of the processing elements must be as fast as the input data rate. It cannot compensate the speed penalty under low supply voltage. On the other hand, the multirate system will require only low-speed processing elements at half of the original clock rate to maintain the same throughput. Therefore, the processing elements can be operated at a lower supply voltage to reduce the power dissipation and the data throughput rate is not degraded by the lowered voltage. As a result, the multirate implementation can provides a direct and efficient way to compensate the speed penalty in low-power designs at the algorithmic/ architectural level.

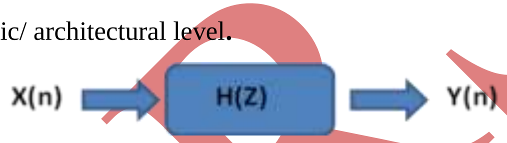


**Fig. 1 An LTI FIR/IIR system.**

Multirate filters are required when changing the sampling rate of a data path in a system. Multirate filters include both interpolation and decimation filters. Interpolation increases the sample rate by inserting zero-valued samples between the original samples, while decimation discards samples to decrease the sample rate. The FIR Compiler automatically creates interpolation and decimation filters using polyphase decomposition. Polyphase filters simplify the overall system design and also reduce the number of computations per cycle required by the hardware.

## II. DESIGN ASPECTS

### 2.1 Interpolation filters

An interpolation filter increases the output sample rate by a factor of *I* through the insertion of *I-1* zeros between input samples, a process known as zero padding. Polyphase decomposition reduces the number of operations per clock cycle by ignoring the zeros that are padded in between the original input samples. Polyphase interpolation filters provide both speed and area optimization because each polyphase filter runs at the input data rate for maximum throughput.

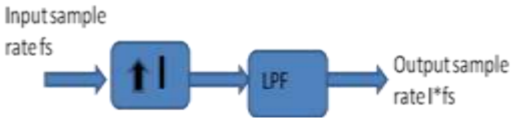


**Fig 2.(a) Polyphase interpolation block diagram**

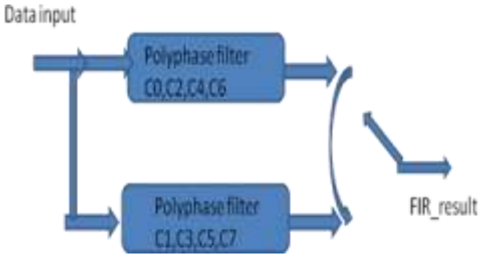


**Fig 2.(b) polyphase decomposition for interpolation filters**

### 2.2 Decimation Filters

A decimation filter decreases the output sample rate by a factor of *D* by keeping only every *D*-th input sample. Polyphase decomposition reduces the number of computations per cycle by ignoring the input data samples that are discarded during down sampling. Polyphase decimation filters provide speed optimization because each polyphase filter runs at the output data rate.

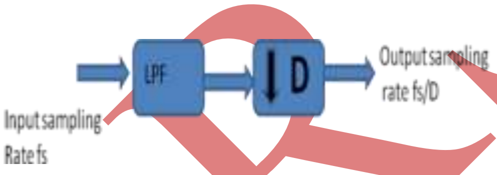


**Fig 3.(a) Polyphase decimation block diagram**

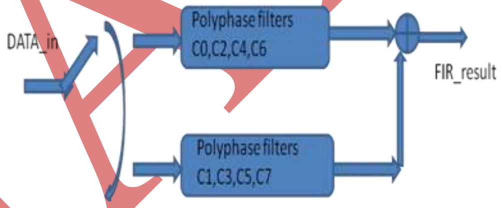


**Fig 3.(b) polyphase decomposition for decimation filters**

### 2.3 Filter design

Fir digital filters find extensive applications in mobile communication systems for applications such as channelization, channel equalization, matched filtering, and pulse shaping, due to their absolute stability and linear phase properties. The filters employed in mobile systems must be realized to consume less power and operate at high speed.

The complexity of FIR filters is dominated by the complexity of coefficient multipliers. few works addressed the problem of reducing the complexity of coefficient multipliers in reconfigurable FIR filters, hardly any work demonstrated reconfigurability in higher order filters. Moreover, we note that there is sufficient scope for more work on complexity reduction in reconfigurable filters especially for wireless communication applications where higher order filters are often required to meet the stringent adjacent channel attenuation specifications. In this paper, we use architecture that integrate reconfigurability and low complexity to realize FIR filters. The FIR filter architecture used is programmable shifts method (PSM). The design analysis of the architecture are presented.

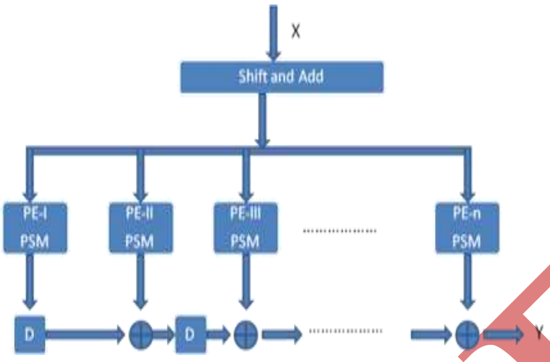


**Fig 4. Transposed direct form of FIR filter**

In this section, the architecture of the used FIR filter is presented. Our architecture is based on the transposed direct form FIR filter structure as shown in Fig. 4. The dotted portion in Fig. 4 represents the MB. In Fig. 4, PE-*i* represents the processing element corresponding to the *i*th coefficient. PE performs the coefficient multiplication operation with the help of a shift and add unit. In the architecture of PE for PSM, the PE consists of programmable shifters (PS). The FIR filter architecture can be realized in a serial way in which the same PE is used for generation of all partial products by convolving the coefficients with the input signal ($h * x[n]$) or in a parallel way, where parallel PE architectures are employed. The basic architecture of the PE is shown in Fig. 5.

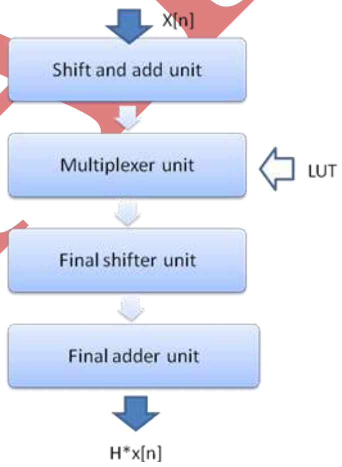


**Fig 5. Block diagram for Architecture of PE for fir filter**

### 2.4 Architecture of PSM

The PSM is based on the BCSE algorithm . The PSM architecture presented in this section incorporates reconfigurability into BCSE. The PSM has a pre-analysis part in which the filter coefficients are analyzed using the BCSE algorithm. Thus, the redundant computations (additions) are eliminated using the BCSs and the resulting coefficients in a coded format are stored in the LUT. The number of multiplexer units required can be obtained from the filter coefficients after the application of BCSE. The number of multiplexers is selected after considering the number of non-zero operands (BCSs and unpaired bits) in each of the coefficients after the application of the BCSE algorithm. The number of multiplexers will be corresponding to the number of non-zero operands for the worst-case coefficient (worst-case coefficient being defined as coefficient that has the maximum number of non-zero operands).The architecture of PE for PSM is shown in Fig. 6. The coefficient wordlength is fixed as 16 bits.

The LUT consists of two rows of 18 bits for each coefficient of the form SDDDDXXDDDDXXMMMML and DDDDXXDDDDXXDDDDXX, where “S” represents the sign bit, “DDDD” represents the shift values from $2^0$ to $2^{-15}$ and “XX” represents the input “*x*” or the BCSs obtained from the shift and add unit. In the coded format, *XX* = “01” represents “*x*,” “10” represents $x+2-1x$, “11” represents
$x + 2-2x$, and “00” represents $x + 2-1x + 2-2x$, respectively.

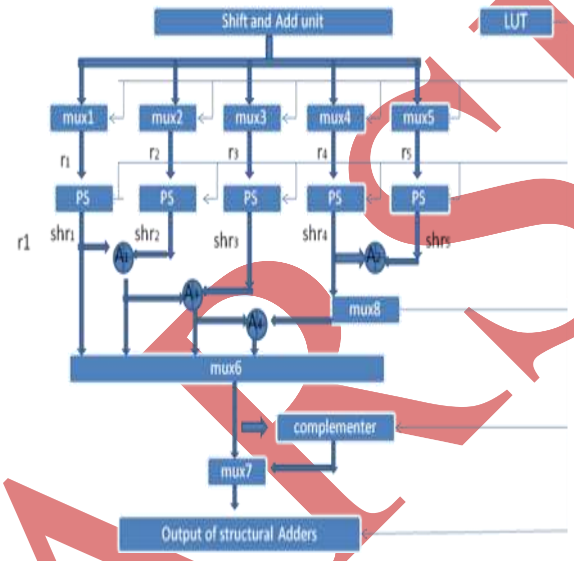


**Fig 6. Architecture of PE for PSM**

The XX values are given as select signals for Mux1 to Mux5. The values of DDDD are fed to corresponding PS. The multiplexer Mux6 and Mux8 will select the appropriate output in case the number of operands after BCSE is less than 5. The use of Mux6 and Mux8 reduces the number of adders utilized by selecting the output from the appropriate adder as all the adders in the PE are not always needed and hence consumes zero current and power. The select signals of Mux6 and Mux8 have five bits and hence 25 different control signals are possible which adds lots of flexibility to the architecture which can be employed in future if required. Mux7 is used to complement the output in case of a negative coefficient and its select signal is the sign bit “S” of the coefficient. The PSM architecture has advantage that it offers the flexibility of changing the wordlength of coefficients. The same PSM architecture designed for 16-bit coefficients is capable of operating for any coefficient wordlength less than 16 bits. This means, if the wordlength is reduced, the format of the LUT can be changed if required. The main advantage of reducing the precision is that some of the adders in the PSM architecture will be unloaded resulting in zero dynamic power. the PSM architecture is the first approach toward programmable coefficient wordlength FIR filter architecture. This means that the coefficient wordlength of the used PSM architecture can be changed dynamically without any change in hardware.

## III. LOGIC BLOCK DIAGNOSIS

### 3.1 Main Concept

An interpolation filter increases the output sample rate by a factor of *I* through the insertion of *I-1* zeros between input samples, a process known as zero padding. Polyphase decomposition reduces the number of operations per clock cycle by ignoring the zeros that are padded in between the original input samples. Polyphase interpolation filters provide both speed and area optimization because each polyphase filter runs at the input data rate for maximum throughput.

A decimation filter decreases the output sample rate by a factor of *D* by keeping only every *D*-th input sample. Polyphase decomposition reduces the number of computations per cycle by ignoring the input data samples that are discarded during down sampling. Polyphase decimation filters provide speed optimization because each polyphase filter runs at the output data rate.

The drawback of using interpolator and decimator is complexity of the circuit is increased if we use normal FIR filters.

To overcome this draw back we go for low complexity FIR filters such as FIR filter design by using PSM(programmable shift method) architecture .By doing so the overall complexity of the Interpolator and decimator circuit is redused.

### 3.2 Implementation and Experimental Results:

Multirate Signal Processing studies Digital Signal Processing systems which include sample rate conversion. This technique is used for systems with different input and output sample rates, but may also be used to implement systems with equal input and output rates.

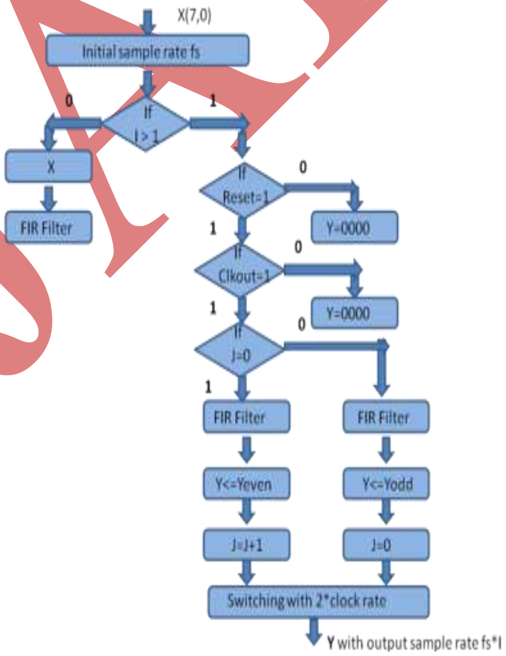


**Fig 7.ASM chart of Interpolator Top-order module**

Multirate filters are required when changing the sampling rate of a data path in a system. Multirate filters include both interpolation and decimation filters. Interpolation increases the sample rate by inserting zero-valued samples between the original samples, while decimation discards samples to decrease the sample rate. The FIR Compiler automatically creates interpolation and decimation filters using polyphase decomposition. Polyphase filters simplify the overall system design and also reduce the number of computations per cycle required by the hardware.

Figure 7 represents the ASM chart of top-order module of the Interpolator. The ASM chart describes the sequence of events as well as the timing relationship between the states of a sequential controller and the events that occur while going from one state to the next.

Interpolation module is explained in five stages.Stage1 represents the sample rate check operation i.e sampling rate at input module is compared with sampling rate of data(fs).stage2 checks the scaling factor I if I = 1 then sampling rate remains as it is and the input message X is given as input to the filter or else we check reset condition.stage3 check the reset condition of circuit which is an active low enable, if reset = 0 then the output the circuit is y=00000 or else we check for output clock enable i.e clkout which is also an active high enable, if clkout=0 then the output of the circuit is y=00000 or else we perform polyphase decomposition by checking for j=0.stage4 if j=0 filtering is done to the even coefficients and the output of the filter is y=yeven or else the filtering is done to the odd coefficients and the output of the filter is y=yodd.stage5 now we operate the output switch of the circuit with double the clock frequency so now the output of the interpolator is double the sampling rate at the input.

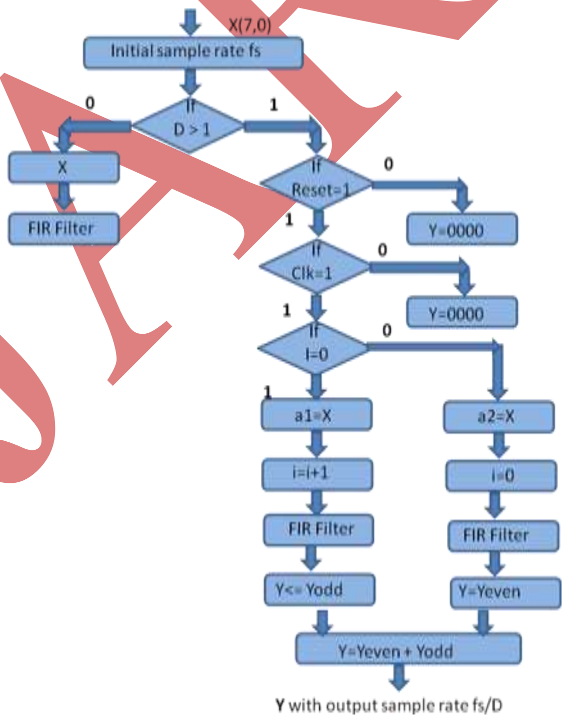


**Fig 8.ASM chart of Decimator Top-order module**

Figure 8 represents the ASM chart of top-order module of the Decimator. Decimator module is explained in five stages.Stage1 represents the sample rate check operation i.e sampling rate at output module is compared with sampling rate of data(fs).stage2 checks the scaling factor D if I = 1 then sampling rate remains as it is and the input message X is given as input to the filter or else we check reset condition.stage3 check the reset condition of circuit which is an active low enable, if reset = 0 then the output the circuit is y=00000 or else we check for clock enable i.e clk which is also an active high enable, if clk=0 then the output of the circuit is y=00000 or else we perform polyphase decomposition by checking for i=0.stage4 if i=0 then X is given as input to the firfil1 and the output of the firfil is yodd or else X is given as input to the firfil2 and the output of the firfil is yeven.stage5 hear we summate all the outputs of y i.e yeven and yodd and so now the output of the decimator is half the sampling rate at the input.

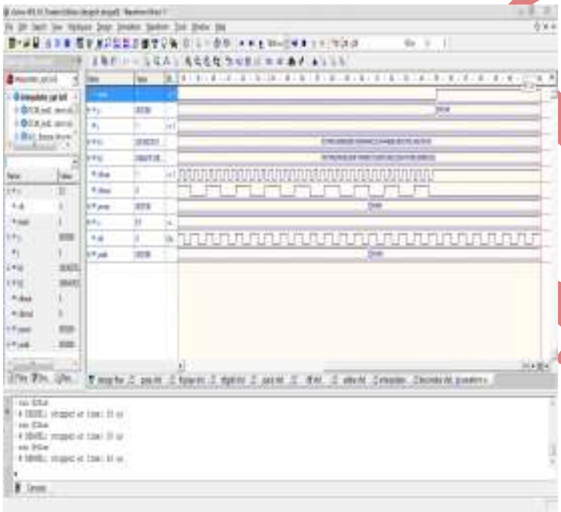
## IV. SIMULATION RESULTS

### 4.1 Interpolator

We can achieve multirate sampling by using Interpolator at input side but the drawback of this technique is complexity is increased to reduce this complexity we use low complexity FIR filters. this is reduced by designing FIR filter by using shift and add module along with programmable shifting method.

### 4.2 Decimator

The same complexity persists in the decimator as well and this is also redused by using same technique.



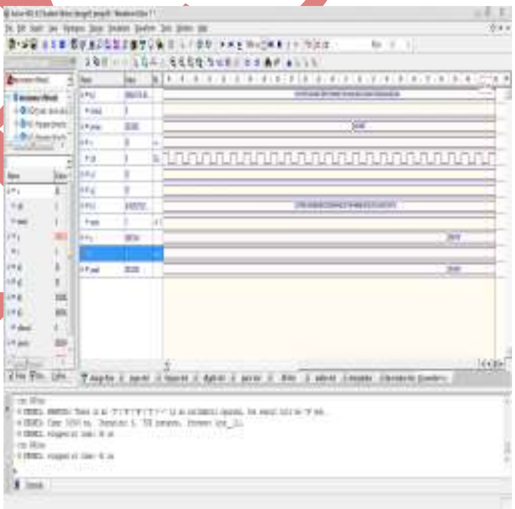


**Fig 9.Simulation results of interpolator Top-order module**

**Fig 10.Simulation results of decimator Top-order module**

## V FPGA Realization

The designed system is targeted on to Xilinx Virtex FPGA device belonging to virtex6v family with a speed grade of -1. It is observed that above 40% area for the targeted FPGA is covered for the implementation of this system. The CLB’S are connected in cascade manner to obtain the functionality for the designed system.

### 5.1 Synthesis Report

Synthesis is a process of constructing a gate level net list from a model of a circuit described in VHDL Figure Depict the synthesis report of multiple fault diagnosis modules in Xilinx ISE 14.1 Environment for VHDL source code.

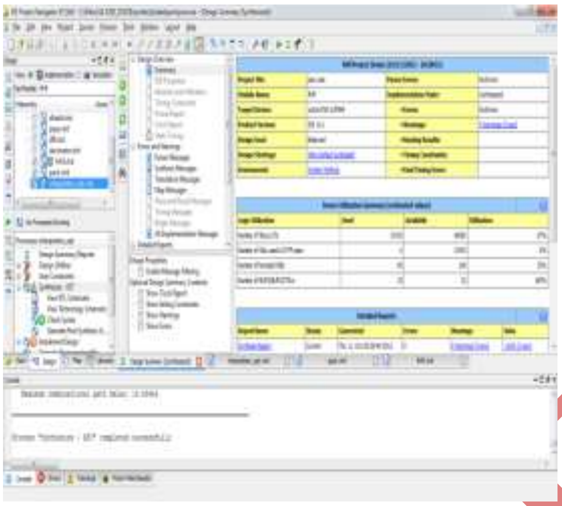


**Fig 11. Synthesis Report of Interpolator and Decimator**

The above figure shows the device summary of the multiple fault diagnosis top order module which represents the summary of the used 6301 IO Buffers, 512 flipflops ,LUTs ,buffers, inverters etc.

The synthesis result for the proposed algorithm is presented:

```
=====================================
*                    Design Summary                         *
=====================================
Top Level Output File Name        : firfil.ngc
Primitive and Black Box Usage:
------------------------------
# BELS                          : 43803
#      GND                      : 65
#      INV                       : 128
#      LUT1                     : 6
#      LUT2                     : 3202
#      LUT3                     : 12801
#      LUT4                       : 15296
#      MULT_AND                    : 448
#      MUXCY                       : 3850
#      MUXF5                       : 4160
#      XORCY                       : 3847
# FlipFlops/Latches              : 512
#      FD                        : 512
# Clock Buffers                  : 32
#      BUFGP                      : 32
# IO Buffers                     : 6301
#      IBUF                      : 4712
#      OBUF                       : 1589
```

From the result it is observed that logical counts of 43803 Basic Element Logic (BEL) is required for the realization of DST processor, and the total memory usage of 254952 kilobytes.

### 5.2 RTL View

In figure 12 shows the RTL schematic of the Interpolator and decimator RTL is an acronym for *register transfer level*. This implies that the source code VHDL/Verilog HDL describes how data is transformed as it is passed from register to register

### 5.3 Technology Schematic

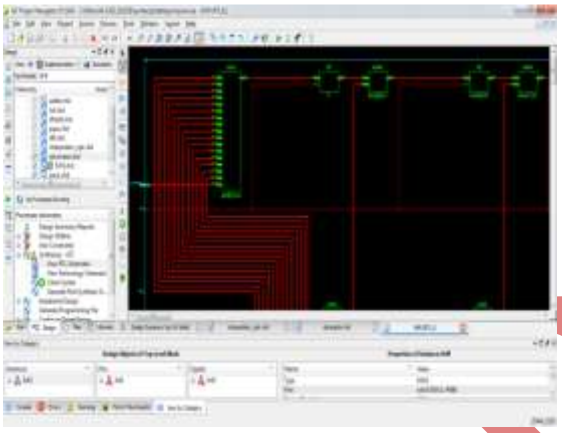


**Fig 12. RTL Schematic View of Interpolator and Decimator**

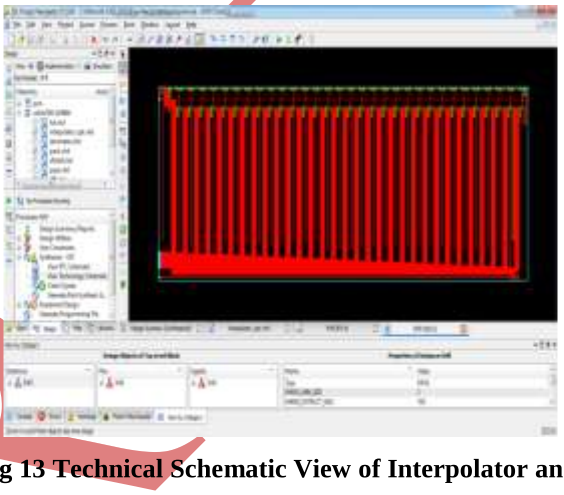


**Fig 13 Technical Schematic View of Interpolator and Decimator**

The above figure 13 shows the technical view of the Interpolator and decimator.

### 5.4 Implementation Observations

The implementation of proposed algorithm is illustrated in various pictorial views obtained during the process of realization i.e., Fig.11 Represents the Synthesis is a process of constructing a gate level net list from a model of a circuit described in VHDL. Fig.12 shows the RTL views of existing and proposed algorithms.Fig.13 shows the one of the technical schematic of targeted FPGA.

## VI . CONCLUSION & RESULTS

In this paper, we have presented how to make FIR filters applicable for multiple sampling rates with the help of interpolator and decimator. For reducing the complexity we had used the low complexity FIR filter design with the help of programmable shifting method.

The presented technique in this paper is able to reduse the complexity of decimator filter and interpolator filter circuits in DSP processing where multiple sampling rates are required.

**Advantages**

The advantages of this work is that it Reduced complexity which in turn reduces the area, Power dissipation ,gives Higher throughput rate, Higher processing speed, Fast Computation, LFSR can rapidly transmit a sequence that indicates high-precision relative time offsets and many more.

## REFERENCES

[1] R. Mahesh and A. P. Vinod, "A new common subexpression elimination algorithm for realizing low complexity higher order digital filters," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 27, no. 2, pp.217–219, Feb. 2008.

[2] R. Mahesh and A. P. Vinod, " New Reconfigurable Architectures for Implementing FIR Filters with Low Complexity" *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 2, Feb. 2010.

[3] Ms. Purvi U. Gandecha and Dr. S. A. Ladhake "Multirate signal processing approaches" IJMIE vol 2,issue 3, ISSN: 2249-0558

[4] Altera corporation design for polyphase interpolation and decimation filters *FIR Compiler MegaCore Function User Guide*.

[5] Liang-Gee Chen Po-Cheng Wu Tzi-Dar Chiueh," For multirate FIR filters and its application in efficient design of subband filter banks " 0-7803-2612-1/9$54 .00 *0* 1995 IEEE

[6] T. Saramaki, "A class of linear-phase FIR filters for decimation, interpolation, and narrow-band filtering,'' *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 32, pp. 1023-1036, Oct. 1984.