# *DESIGN OF CASCADED-INTEGRATOR-COMB FILTER FOR PSK MODEM*

**Ela Chandel**
***Department of Electronics & Communication Engineering***
***Dehradun Institute of Technology,***
***Dehradun- 248001***
***Email: elachandel83@yahoo.com***

## INTRODUCTION

The project presents the design of cascaded-integrated-comb filter devised by Hogenauer for the pulseshaping the digital data for PSK Modulator and its implementation of Field Programmable Gate Array. We have designed a three stage CIC filter with a differential delay of 1 and have completed its software simulation and its implementation on Field Programmable Gate Array. The input data is sampled at the frequency 1MHz and the CIC filter has a programmable decimator from $2^1$ to $2^6$.

## GENERAL DESCRIPTION OF CIC FILTER

### CIC Decimator

The integrator section consists of N ideal integrator stages operating at the high sampling rate fs. Each stage is implemented as a one-pole filter with a unity feedback coefficient.

The transfer function for a single integrator is

$$HI(Z) = 1 / 1 - Z^{-1} ) \qquad (1)$$

The comb section operates at the low sampling rate fs /R where R is the integer rate change factor. This section consists of comb stages with a differential delay of M samples per stage. The differential delay is a filter design parameter used to control the filter's frequency response. M is restricted to be either 1 or 2. The transfer function for a single comb stage, referenced to the high input sample rate is

$$H_C(z) = 1 - z^{-RM} \qquad (2)$$

To ensure high system clock frequencies, the CIC decimator is actually implemented using the pipelined architecture in Figure 1(b) The pipeline registers P0, P1, P2 and P3 shorten the critical path through the differentiator cascade of the basic architecture shown in Figure1(a).

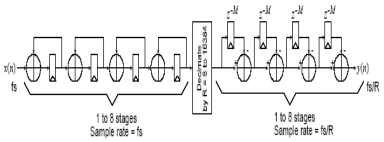



Fig 1(a) CIC Decimation Filter.

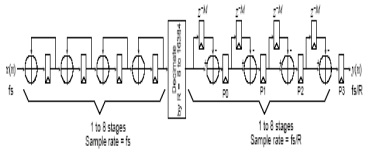



Fig1(b) Pipelined CIC decimator

The CIC decimator, shown in Fig 1(a), consists of a cascade of integrators followed by a resampling switch and a cascade of differentiators. The differential delay M in the differentiator chain may be defined by the user to be either 1 or 2.

### CIC Interpolator

Exchanging the integrator cascade with the differentiator cascade, as shown in Figure 2(a), produces a CIC interpolator. Data is presented to the filter at the rate f s / R where it is processed by the differentiators. Just like the CIC decimator, the Core implementation uses a pipelined structure as shown in Figure 2.(b)

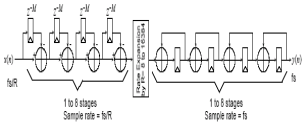



Fig 2(a) CIC Interpolator System

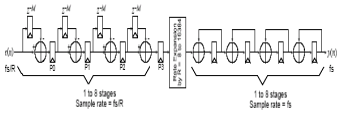



Fig 2(b) Pipelined CIC Interpolation Filter

## BIT GROWTH

For CIC decimators, the gain G at the output of the final comb section is

$$G = (RM)^N \qquad (3)$$

Assuming two's complement arithmetic, we can use this result to calculate the number of bits required for the last comb due to bit growth [2 and 3]. If Bin is the number of input bits, then the number of output bits, B(out),

$$B(out) = [N\log_2 RM + Bin] \quad (4)$$

It also turns out that Bout bits are needed for each integrator and comb stage. The input needs to be sign extended to Bout bits, but LSB's can either be truncated or rounded at the later stages.

## ARCHITECTURE AND IMPLEMENTATION

Finite word-length implementation of CIC filter requires that careful consideration be given to the effects of truncation and overflow errors in the integrator stages [5]. Since each integrator stage has a DC gain of M, the decimation factor, an additional $\log_2 M$ digits must be allocated at each integrator stage to prevent overflow errors during the M clock cycles. The scaler at the input of the filter implements the 1/M scale factors for all three integrator stages.

### The Integrator Stage

The recursive integrator is the critical path in the CIC filter and its delay will determine the maximum achievable input data sample rate[5]. In order to achieve a high sample rate, the recursive integrators are implemented using carry-save arithmetic. This reduces critical path to two full-adder delays between registers(Fig 3.2).

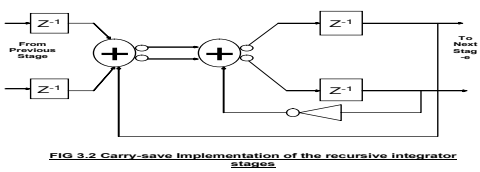



Fig .3 The integrator

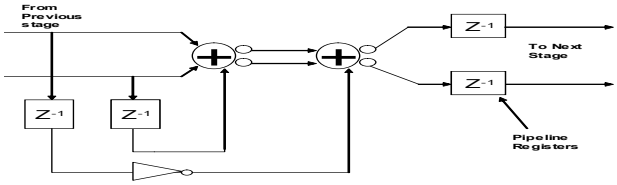



Fig 4 The Comb

### The Comb

The comb filters are also implemented using carry-save arithmetic to facilitate high speed operation(Fig4). Again, the critical path is two full adder delays for each comb section and the adders are designed with inverted sum and carry outputs[5]. Pipeline registers must be inserted between the comb filter sections to break up the long chain of adders.

### The Programmable Decimator

The programmable decimator is implemented as a programmable counter that produces selectable power-of-two output clocks( Fig 3.4)[5]. The counter is implemented using carry-save arithmetic to achieve high speed operation.

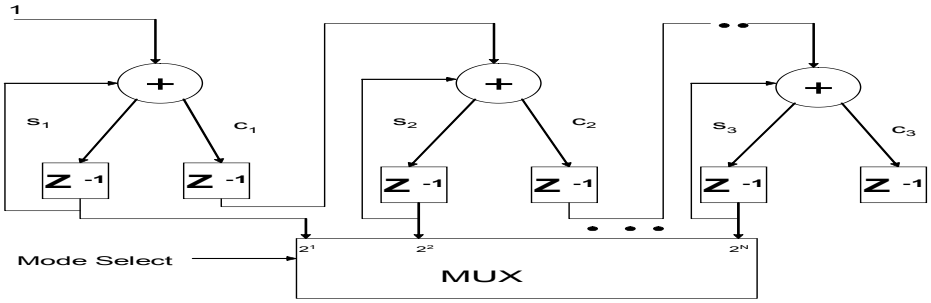



**Fig5** The Programmable Decimator

**The Scaler**

.

The shifter implements the required programmable scaling for decimation factors M from $2^1$ to $2^6$ by using a multiplexer that selects one of ten possible hardwired shifted versions of the input data(Fig 3.5).

S S …….S S X X …....X X

With S X X……X X (sum)

1 1 ……...1 1 0 0 …….0 0 (carry)

**RESULTS AND DISCUSSION**

**Frequency Response of the CIC filter using MATLAB**

***Frequency Response***

CIC filters have a lowpass frequency characteristic. The magnitude frequency response as

$$|H(f)| = \left[ \frac{\sin \pi M f}{\sin \frac{\pi f}{R}} \right]^N \quad (5)$$

The frequency response of Cascaded-Integrator-Comb filter can be seen on MATLAB. The MATLAB programme requires three parameters to be provided by the user—

1. The Number of stages of the CIC filter.
2. The Differential Delay
3. The Rate Change

***Comparison of CIC filters with Different no of stages, differential delay and Rate Change***

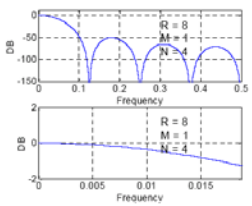



**Fig.7** CIC filter frequency response
$N = 4$, $M = 1$, $R = 7$ and $fc=1/8$

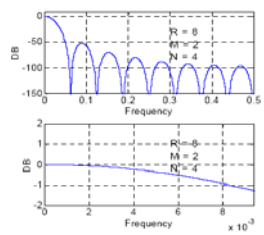
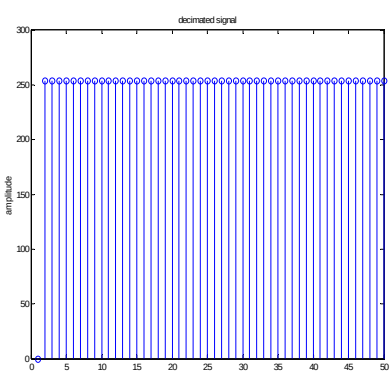



**Fig.8** CIC filter frequency response
$N = 4$, $M = 2$, $R = 7$ and $fc=1/8$

.

***Step Response of the CIC filter using MATLAB***

The step response of the CIC filter can be seen on MATLAB for different values of number of stages, differential delay and rate change [8]. The input is a step function having amplitude 127.



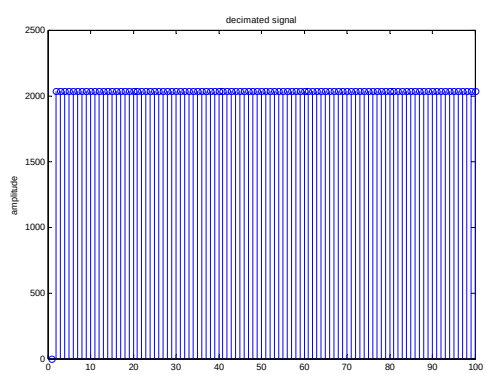




**Fig.9 a.** Step Response Of The Three Stage CIC filter for M=1, N=3,R=2
**Fig.9 b.** Step Response Of The Three Stage CIC filter for M=1, N=3,R=16

**Simulation Results**

For the single stage CIC filter having a step input of amplitude 127, the implementation is done using verilog on Xilinx ModelSim and Maxplus 9.5 for different values of selection inputs, the decimation ratio can be programmed from $2^1$ to $2^6$.

The single stage CIC filter is successfully fused on Flex 10K100E and the outputs can be verified from the simulation results. The output of the hardware realization can be seen on the oscilloscope.

**Discussion**

The successful implementation of the CIC filter is done on field-Programmable Logic Array with the help of Verilog. The carry save implementation reduces the required chip area, decreases the critical path, and reduces the power dissipation.

**APPLICATIONS**

• Channelization functions in a digital radio or MODEM
• Part of the digital up-conversion signal processing chain in a transmitter
• Any filter structure that is required to efficiently effect a large sample rate change like in narrowband extraction of the wideband sources.

**FUTURE SCOPE**

Since their inception, CIC filters have become an important building block for Digital Signal Processing systems. They have found a particular niche in digital transmitters and receivers. They are currently used in highly integrated chips from Intersil, Graychip, Analog devices, as well as other manufacturers and custom designs.

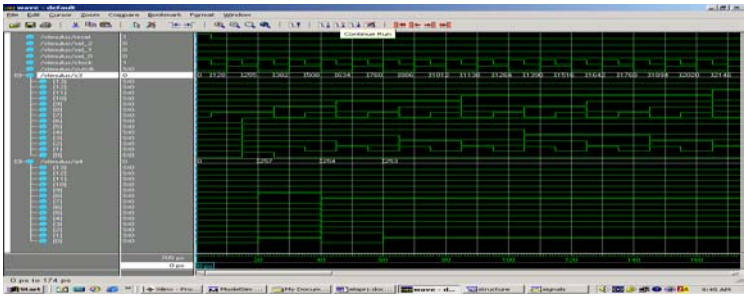


**Fig.10** Verilog Simulation result of CIC filter for N=1,M=1,R=2

**REFERENCES**


1. Alan Y. Kwentus, Member, IEEE, Zhongnong Jiang, and Alan N. Willson, Jr., Fellow, IEEE, "Application of Filter Sharpening to Cascaded-Integrator-Comb Decimation Filters", IEEE Transactions on Signal Processing**,**Vol.45, No.2,February1997.
2. E.B. Hogenauer, "An economical class of digital filters for decimation and interpolation", IEEE Transactions on Accoustics, Speech, and Signal Processing, vol. ASSP-29,pp.155-162,April 1981.
3. Matthew P. Donadio"CIC Filter Introduction", July 18,2000.
4. Samir Palnitkar, "Verilog HDL, a Guide to Digital Design and Synthesis", Mountain View, CA, Suncoft Press,pp.8-8, 1996.
5. Wayne Burleson, Konstantinides, Teresa Meng, "VLSI Signal Processing IX", IEEE Signal Processing Society, pp. 231-240,1996.
6. Harris Semiconductor chip datasheet, HSP 50214,February 1998.
7. Xilinx Inc, Virtex-II Platform FPGA Handbook, 2000.
8. Meyer-baese, Uwe, "Digital Signal Processing with FPGA", Barlin, Springer,pp. 155-178,2003.