Fixed-Point DSP Implementation of FM Demodulation/Decoding

H. Sundaresan(1), N. Kehtarnavaz(1), B. Simsek(2)
(1) DSP Lab, Dept. of Elec. Eng., Texas A&M University
(2) Texas Instruments Inc., Houston

Abstract – This paper addresses the fixed-point DSP implementation of FM demodulation and decoding for the digital radio application. The demodulation module involves division and arctan for which there is no dedicated hardware on a fixed-point DSP. Division is normally achieved either by using the Newton-Raphson method or by using conditional subtraction instruction. It is normally more efficient to use a table lookup approach. However, in a table lookup, if the denominator value is very small, the size of the table lookup will be very large for high accuracy. In this paper, we present a new method, which utilizes the in-phase representation of the denominator overcoming the table lookup problem. We have shown that this method reduces the instruction cycle time by 40% for the demodulation module. In addition, in the decoding module, we have devised an efficient implementation of pilot frequency computation, which reduces the instruction cycle time by 30%. Example implementation timings on the TI TMS320C6201 DSP are reported.

I. Introduction

In FM demodulation and decoding , the baseband multiplexed (bbmux) signal generated using Matlab is of the form,

$$[\text{left}(t) + \text{right}(t)] + Ap\times \text{Sin}(Wp\times t) + [\text{left}(t) - \text{right}(t)]\times \text{Sin}(2\times Wp\times t)$$

where,	Ap	=	pilot signal amplitude
	Wp	=	pilot angular frequency(2×pi×Fp)
	Fp	=	pilot frequency = 19 KHz

The baseband signal is modulated using a FM carrier, with the sampling frequency Fs = 44.1 × 32 KHz where the carrier frequency Fc = Fs/4. A complex baseband FM signal is generated from the conventional FM signal[1] as shown in Figure 1. The total mixing operation produces a real (In-phase) and an imaginary (Quadrature-phase) baseband component. This complex representation of FM signal has a major advantage in the design of FM demodulator, which involves division and arctan operations. In the place of the Newton-Raphson method or conditional subtraction instruction for division, a lookup table has been used in this paper which utilizes the In-phase representation of the denominator, overcoming the table lookup problem. This way is proved to be more efficient involving lesser number of instruction cycles to perform division operation.

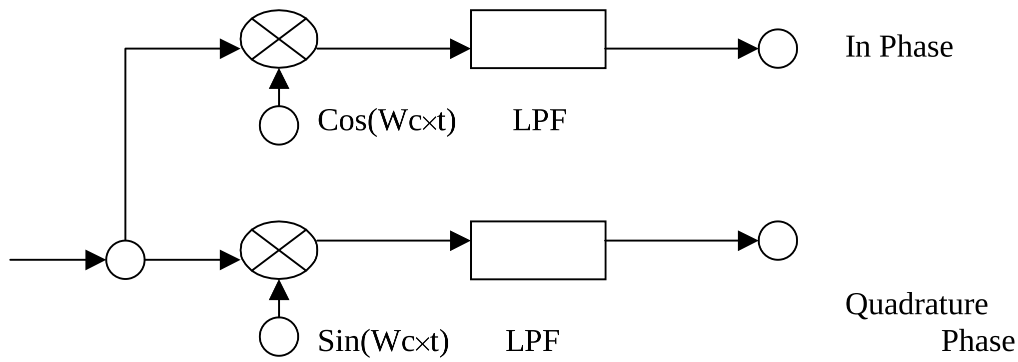


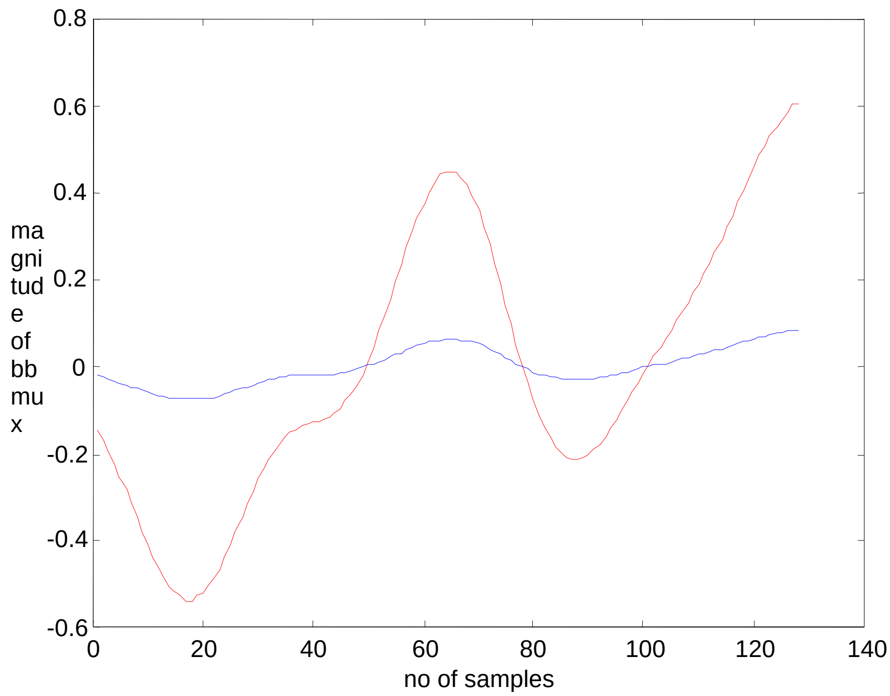
Figure 1

FM decoding is implemented using the non-coherent scheme[2]. Since this method requires the sampling rate to be 152KHz (for synchronizing the sampling rate to the phase of the incoming pilot frequency), the output of the demodulated signal has to be appropriately down sampled. This issue is not dealt with in this paper. The input for decoding is generated separately using Matlab. The decoding algorithm involves two modules: (1) a DFT snapshot (128 points) implementation at the pilot signal's frequency in order to determine the pilot magnitude and phase, and (2) implementation of polyphase interpolator filter in order to interpolate the value of the incoming signal at the instances when the pilot signal is at 45, 135, 225 or 315 [2]. An efficient method of calculating the DFT snapshot for 128 points with reduced number of multiplication and additions is also presented here.

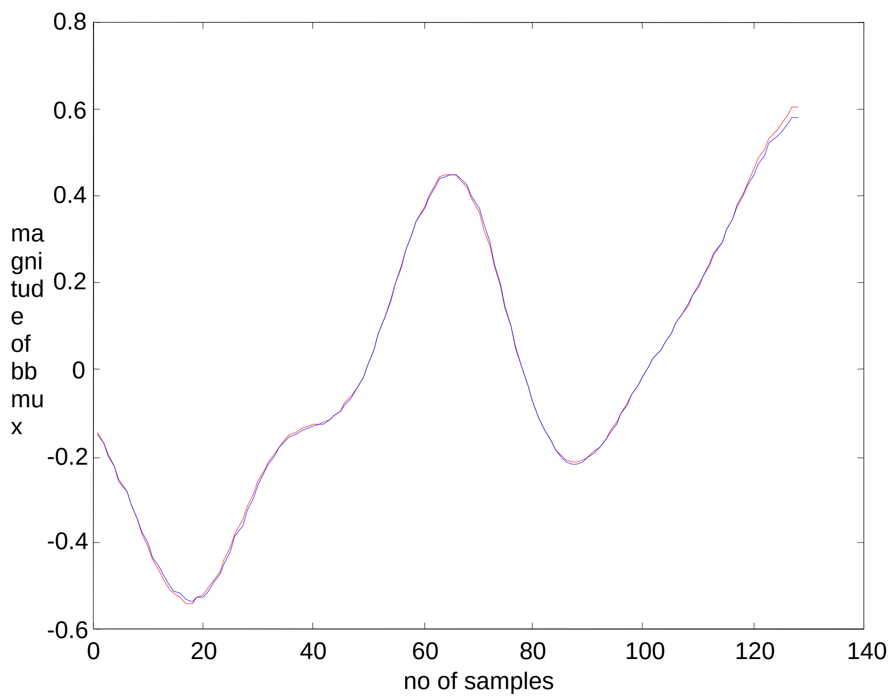
II. Division Operation

In the fixed point DSPs, normally there are no dedicated instructions to perform division or trigonometric functions like arctan. In the demodulation module, division of Imaginary (IM) by Real (RL) and arctan of IM/RL operations are encountered. One of the methods commonly used for division is the Newton - Raphson method to get the full accuracy after the use of a reciprocal instruction. One other method is by using the conditional subtraction. The third method and the fastest of all, is the use of lookup table for 1/RL and then multiply by IM. But the disadvantage of the lookup table method is that, if the value of RL is very small, then the size of the lookup table has to be very large for better accuracy in which case the memory usage is not very efficient.

This problem is overcome here by using an approach which utilizes the fact that RL is in In phase representation. Since RL is cosine(x), it always ranges between [-1,1] for any value of x. By adding two to the RL part, the range is shifted to [1,3]. The addition of two solves the problem of having a lookup table for very small values. The lookup table for the reciprocal of RL ranges from [1,1/3]. Addition of two to RL, results in a reduction of amplitude of the demodulated signal by a factor K almost uniformly as shown in Figure 2a. Thus, by multiplying with this amplification factor at the output, the demodulated signal is obtained with a reasonable SNR, as shown in Figure 2b. This method reduces the number of cycles considerably and thereby increasing the speed.



(a)



(b)

Figure 2 Plots of atan(IM/RL) and atan(IM/(RL+2)) (a) before multiplication with K, (b) after multiplication with K.

III. N Point DFT Snapshot

FM decoding is implemented using the non-coherent scheme. Decoding using this approach requires a DFT snapshot for 128 points at the pilot signal's frequency in order to determine the pilot signal's magnitude and phase. The advantage of choosing sampling rate to be 152KHz is that it is eight times the pilot frequency. Thus, for a period of 2Π, the pilot frequency computation needs to be performed at 45, 90, 135, 180, 225, 270, 315 and 360 degrees only[2]. One way of doing this is to have a cosine and a sine table each with eight entries of the values of cosine and sine at those eight phases and multiplying them with the input for every eight points in a loop for 128 points [2]. This method referred to as method A, for every eight sample points, uses 8 multiplication and 8 additions.

The second method addressed as method B, which utilizes the same principle but a different implementation, is more efficient in terms of speed. For every eight sample points, this method uses no multiplication and only 6 additions. This method makes use of the fact that cosine and sine values at 0, 90, 180, 270 and 360 are either zero or one. As a result, at those points the multiplication are avoided and only additions are performed. Those data which need to be multiplied with cosine and sine values at 45, 135, 225 and 315, are separately added inside the loop and multiplied with 0.707 outside the loop. Note that necessary scaling has to be performed to avoid overflow during addition.

IV. Results

FM demodulation and decoding algorithms were implemented on the Texas Instruments TMS320C6201 DSP and were analyzed based on their speed, memory, SNR and channel separation. The code was written in C and then in linear assembly for better optimization.

In the demodulation module, two different methods of division were implemented separately and the number of clock cycles per sample was determined using the break point feature of Code Composer Studio to compare the speed of each method. The results are as follows:

Division method used	**no of cycles/sample**
Demodulation with lookup table approximation	39
Demodulation with conditional subtraction	87
Division by Newton-Raphson method	46

As can be seen from this table, the lookup table approach is 40% more efficient than the conditional subtraction method in terms of speed. Since in fixed-point DSPs in general and in TMS320C6201 in particular, there is no reciprocal instruction, the Newton

Raphson method could not be implemented with full Q format accuray. Division by Newton Raphson for three points of accuracy was implemented to compare the timings for division. SNR was calculated for the lookup table approximation method using the expression

$$20\times\log10(\mathrm{var}(\mathrm{signal})/(\mathrm{var}(\mathrm{signal} - \mathrm{approximation}))$$

where the signal is the demodulated output obtained in the ideal case. The SNR for the lookup approximation method was 58.68.

In the decoding module, two different methods of DFT snapshot were implemented and timed to determine the speed. The results are as follows:

DFT method used	**no of cycles/sample**
DFT snapshot method A	346
DFT snapshot method B	83

As can be seen from this table, the reduction in number of cycles for DFT snapshot resulted in a 30% reduction in the timings for the entire decoding module, a significant improvement over the original approach.

V. Conclusion and Future Work

This work has addressed a new approach to perform division using a lookup table with an approximation and implementation of a faster DFT snapshot, for the fixed-point DSP implementation of FM demodulation and decoding for the digital radio application. This method is shown to be effective with respect to speed. The data representations were in Q-15. When implemented in Q-30 representation, the SNR is expected to be further increased. Some modules of the code could be hand coded which would further reduce the number of cycles and increase the speed.

References:

[1] James Michael Shima, FM demodulation using a digital radio and digital signal processing, Thesis, May 1995.

[2] Mark E. Utter, A new approach to decode an FM stereo signal using a DSP, Texas Instruments.

[3] N. Kehtarnavaz and B. Simsek, C6X-Based Digital Signal Processing, Prentice-Hall, April 2000.

[4] Texas Instruments, TMS320C6X CPU and Instruction Set Reference Guide.

[5] Texas Instruments, TMS320C6X Programmer’s Guide.

[6] Texas Instruments, TMS320C6000 Code Composer Studio Tutorial.

[7] Texas Instruments, TMS320C6X Optimizing C Compiler User’s Guide.

[8] The Math Works Inc., MATLAB