Hindawi Publishing Corporation
International Journal of Reconfigurable Computing
Volume 2011, Article ID 342532, 10 pages
doi:10.1155/2011/342532

Research Article

A New High-Performance Digital FM Modulator and Demodulator for Software-Defined Radio and Its FPGA Implementation

Indranil Hatai and Indrajit Chakrabarti

Electronics and Electrical Communication Engineering, Indian Institute of Technology, Kharagpur 721302, India

Correspondence should be addressed to Indranil Hatai, indranil.hatai@gmail.com

Received 31 March 2011; Accepted 8 September 2011

Academic Editor: Scott Hauck

This paper deals with an FPGA implementation of a high performance FM modulator and demodulator for software defined radio (SDR) system. The individual component of proposed FM modulator and demodulator has been optimized in such a way that the overall design consists of a high-speed, area optimized and low-power features. The modulator and demodulator contain an optimized direct digital frequency synthesizer (DDFS) based on quarter-wave symmetry technique for generating the carrier frequency with spurious free dynamic range (SFDR) of more than 64 dB. The FM modulator uses pipelined version of the DDFS to support the up conversion in the digital domain. The proposed FM modulator and demodulator has been implemented and tested using XC2VP30-7ff896 FPGA as a target device and can operate at a maximum frequency of 334.5 MHz and 131 MHz involving around 1.93 K and 6.4 K equivalent gates for FM modulator and FM demodulator respectively. After applying a 10 KHz triangular wave input and by setting the system clock frequency to 100 MHz using Xpower the power has been calculated. The FM modulator consumes 107.67 mW power while FM demodulator consumes 108.67 mW power for the same input running at same data rate.

1. Introduction

In the prevalent audio broadcasting applications like private mobile radio (PMR) and digital audio broadcasting-terrestrial (DAB-T) standards, excellent clarity along with the source stability is required for the voice transmission. Frequency modulation (FM) scheme is used in most of these standards. Traditionally, FM signal generation was performed using some analog components to support the audio broadcasting standards. But difficulties arose in analog FM modulation scheme due to the use of the voltage-controlled oscillator (VCO). Using the VCO, it is very difficult to obtain a good clarity as well as source stability in FM-modulated or demodulated signal as VCO suffers from lack of linearity over the desired frequency range. Therefore, digital implementation of FM modulation scheme has evolved to replace the traditional analog counterpart. Nowadays, to get superior performance and good voice clarity in any audio broadcasting system, digital FM modulation and demodulation technique is widely used. To ensure linearity over the entire frequency range, designers choose to replace the VCO by a DDFS, sometimes referred to as a numerically controlled oscillator (NCO). Considerable research has been performed on different digital FM modulator architecture. Some of these have laid stress on reducing the distortion effects of quantization noise, which occurs due to the bit resolution at the input and output of the DDFS [1]. Some of them have discussed the area optimization and low-power consumption [2–4] as main objective. In the present work, one high-speed, low-power, and reduced-area digital FM modulator has been implemented in the FPGA device to support the audio broadcasting system in software-defined radio (SDR) system.

There exist various architectures [5–14] for implementing digital FM demodulator into a single chip, although their performance has mostly been limited by analog signal processing accuracy. The basic fundamentals behind FM demodulation are how to discriminate accurately a small

frequency deviation of the FM-modulated signal from its center frequency. PLL method is one of the popular techniques for FM demodulation. It can be easily implemented in integrated forms, but sudden departure from its linearity property of the VCO in some portions of the frequency range degrades the overall system performance. Digital PLLs provide a better possible solution to overcome some of the bottlenecks of analog PLLs [15]. Due to this, in the present FM demodulators, the digital phase-locked loop (DPLL) is mostly used for accomplishing the frequency discrimination. The DPLL tracks the variations in the received signal phase and frequency. There are also some other techniques by which the frequency can be computed from the ratio of the in-phase (I) and the quadrature (Q) components. Modern communication revolves around high-speed, high data rate transmission and reception. DPLL-based implementation of FM demodulators in DSP often does not meet such demanding requirements of a wireless communication system. An alternative solution is to implement it in FPGA due to its flexibility and modularity. A reduced-area, low-power, and high-speed linear digital FM demodulator using the DPLL technique [5, 6] has been implemented towards the development of an SDR system. Componentwise improvements have been carried out in this work to get compact architecture, a faster system clock, and achieve a less power consumption while compared with existing implementations of digital FM demodulator. In SDR application less-area and low-power consumption with high data rate support is the key concern. Targeting to the next generation SDR-based wireless communication transceiver, in this work all the basic components of DPLL-based FM demodulator are fully optimized without losing the system output behavior in comparison with the previous DPLL-based FM demodulator implementations.

The present paper is as follows. Section 2 describes the principle and architecture of FM modulator and DPLL-based FM demodulator along with the architecture of individual component of FM modulator and DPLL-based FM demodulator, and in Section 3, FPGA implementation results in terms of synthesis results, simulation results, on-chip-verified results, and comparison results are mentioned. Conclusions are summarized in Section 4.

2. Architecture of Digital FM Modulator and DPLL-Based FM Demodulator

2.1. FM Modulator. In the FM modulation technique, which is a kind of angle modulation methods, the instantaneous frequency of the carrier signal varies linearly with the baseband-modulated message signal $m(t)$ as follows:

$$\begin{aligned} S_{\mathrm{FM}}(t) &= A_c \cos[2\pi F_c t + \theta(t)] \\ &= A_c \cos\left[2\pi F_c t + 2\pi K_f \int_0^t m(n)dn\right], \end{aligned} \tag{1}$$

where A_c is the amplitude of the carrier, F_c is the carrier frequency, and K_f is the frequency deviation constant. The architecture of the FM modulator is as shown in Figure 1.

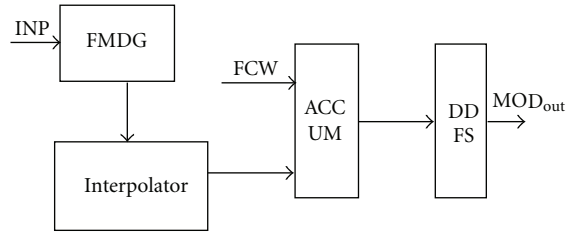

Figure 1: Block diagram of digital FM modulator.

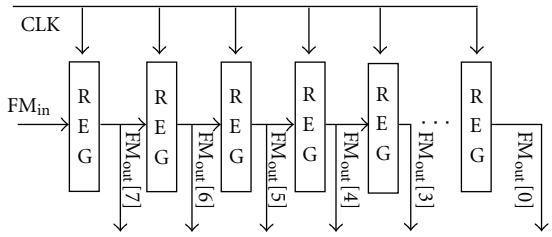

Figure 2: Block diagram of FM data generator.

The FM modulator consists of (1) an FM data generator, (2) an interpolator with interpolation factor of 32, (3) an accumulator, and (4) a DDFS block. The FCW signal has been used for generating different carrier frequency. The accumulator block adds the instantaneous frequency of the input audio signal with the selected carrier frequency. And finally DDFS block takes this frequency as an input and generates the FM-modulated signal. The architecture of the DDFS block has been discussed in the later section.

(1) FM Data Generator. Usually, FM modulation scheme is used to support audio processing in the audio broadcasting system. Generally, the audio signal is processed in the range between 44 Kbps and 320 Kbps. The FM input data is sampled at each FM symbol clock and stored in a register for further processing. The digitized input data is passed through a serial-to-parallel converter to generate the 8-bit FM input data. The architecture for FM data generator is shown in Figure 2.

(2) Interpolator. Interpolator block is used in FM modulator to get a better power level for the FM transmission. In this work, an interpolation factor of 32 has been used between two consecutive audio samples. The circuits first calculate the difference and then divide the remainder by 32. To perform the division by 32, the new FM input data is shifted by one bit before and by four bits after the subtraction. Then the output is added with the previous input data on every symbol clock. One subtractor, one adder, and some registers are required to perform the interpolation operation in hardware. The architecture for the interpolator is as shown in Figure 3.

The power spectral density of the designed FM modulator has been shown in Figure 4.

2.2. DPLL-Based FM Demodulator. Operation of the digital phase-locked loop as an important component of FM

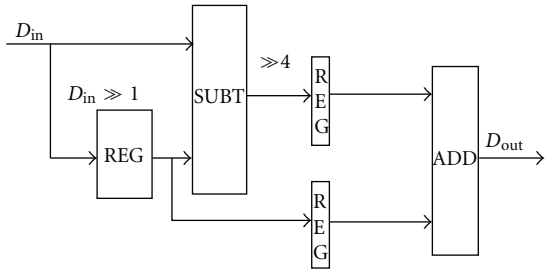

Figure 3: Block diagram of interpolator circuit.

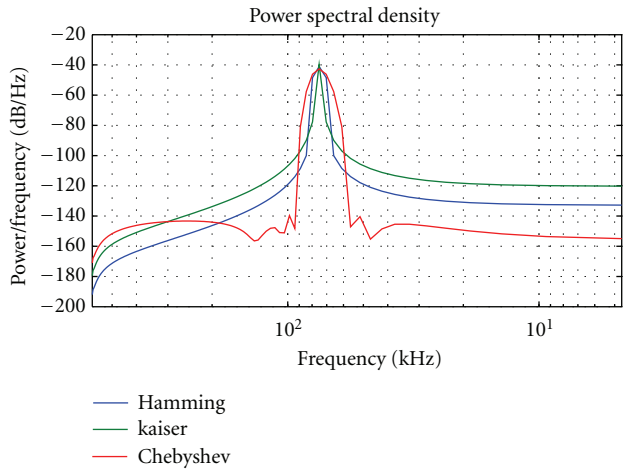

Figure 4: Power spectral density of FM modulator.

demodulator has been conceived in the early 1970s [16, 17]. The input frequency modulated signal can be expressed as follows:

$$V_i(t) = \sin(\omega_i t + \theta_i(t)). \tag{2}$$

Feedback loop mechanism of the PLL makes the DDFS to generate a sinusoidal signal $V_0(t)$ with the same frequency as that of $V_i(t)$, where

$$V_0(t) = \cos(\omega_i t + \theta_0(t)). \tag{3}$$

The output of the phase detector, which is the product of these two signals, is found using familiar trigonometric identity:

$$\begin{aligned} V_d(t) &= K_d[V_i(t) * V_0(t)], \\ V_d(t) &= K_d[\sin(\omega_i t + \theta_i(t)) * \cos(\omega_i t + \theta_0(t))] \\ &= \frac{K_d}{2}[\sin(2\omega_i t + \theta_i(t) + \theta_0(t)) + \sin(\theta_i(t) - \theta_0(t))], \end{aligned} \tag{4}$$

where K_d is the gain of the phase detector. The first term in (4) corresponds to the high-frequency component. The second term corresponds to the phase difference between $V_i(t)$ and $V_0(t)$. The phase difference, that is, $(\theta_i(t) - \theta_0(t))$ between the modulated signal and the carrier produces the desired original signal with frequency ω_i.

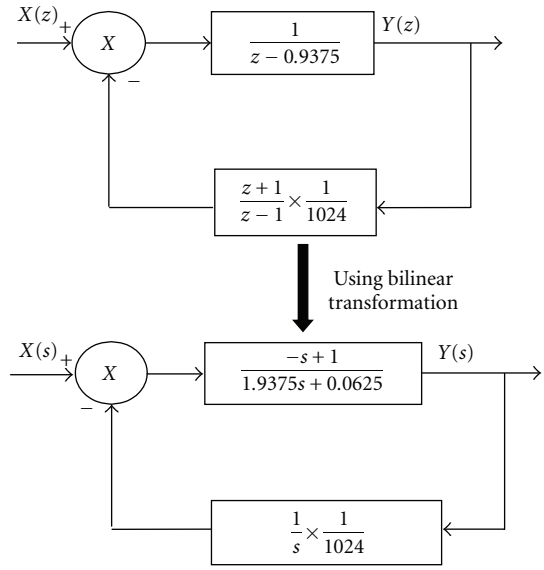

Figure 5: Block diagram of PLL system in analyzing transient response.

The single most important point to realize while designing the PLL is that it is a feedback system and, hence, it is characterized mathematically by the same equations that are applied to other more conventional feedback control systems. The mathematical model of the digital PLL system can be derived to analyze the transient and steady state responses. The block diagram of a typical digital PLL system in z domain [18] (discrete domain) and its transformation in s domain [19] (continuous time) is shown in Figure 5.

The transfer function of the system is

$$\frac{Y(s)}{X(s)} = -\frac{-s^2 + s}{1.9375s^2 + 0.06161s + 0.00089}. \tag{5}$$

The second-order DPLL system improves the performance of the loop in terms of speed and locking range as compared to the first-order DPLL system. That is why the DPLL system used here is a second-order system. The unit step response curve is obtained using MATLAB for the system shown in Figure 6. From the figure it can be seen that the system is stable with overshoots at the transient state.

The complete FM receiver consists of the basic building blocks as shown in Figure 7. The FM receiver consists of four basic parts: (1) Phase Detector (PD), (2) Loop Filter (LF), (3) Direct Digital Frequency Synthesizer (DDFS), and (4) FIR Filter.

3. Phase Detector

The phase detector is used to detect the phase error between the incoming frequency-modulated signal from the ADC and the output frequency generated from the DDFS. This operation needs one register and one multiplier module. The modified Radix-4 Booth-Encoded Wallace-tree multiplier [20–22] architecture is used instead of a signed arithmetic multiplier. This architecture has been chosen because it

Table 1: Truth table of modified booth encoder.

Y_{i+1}	Y_i	Y_{i-1}	Value	$X1\_b$	$X2\_b$	Neg	Z
0	0	0	0	1	0	0	1
0	0	1	1	0	1	0	1
0	1	0	1	0	1	0	0
0	1	1	2	1	0	0	0
1	0	0	−2	1	0	1	0
1	0	1	−1	0	1	1	0
1	1	0	−1	0	1	1	1
1	1	1	0	1	0	1	1

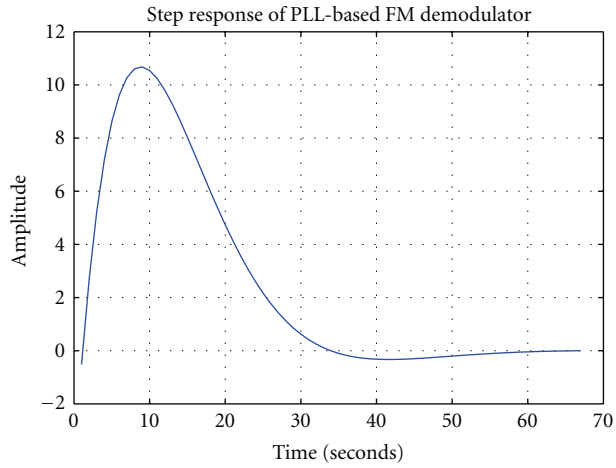

Figure 6: Unit step response for PLL system used in FM demodulator.

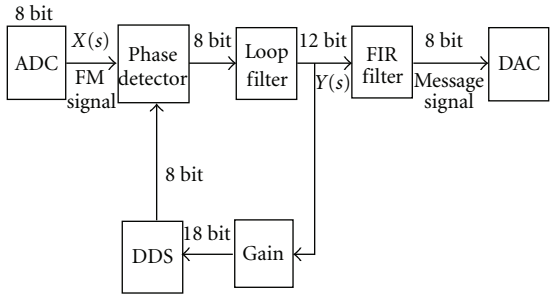

Figure 7: Block diagram of digital FM demodulator circuit.

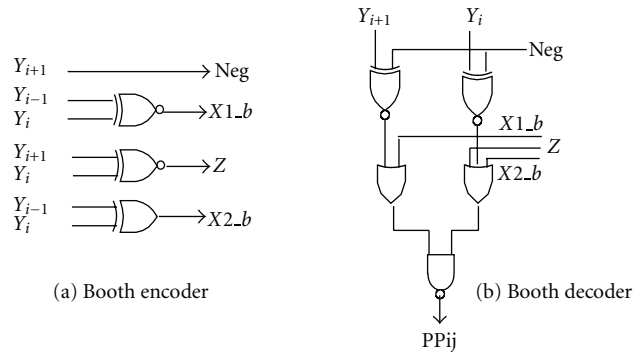

Figure 8: Booth encoder and decoder for modified booths multiplier.

reduces the number of partial products to $N/2$ for an $N * M$ bit multiplication process. Conventionally, in a radix-4 Booth multiplier, there are three basic steps to be followed: (1) generate the reduced partial product according to Booth's algorithm, (2) reduce the number of additions of the partial product, and finally (3) use a high-speed adder like carry look-ahead adder (CLA) for the last two rows of the partial product tree. For the signed multiplication operation, the sign extension scheme has been combined with Booth's algorithm which is known as the modified Booth algorithm. To multiply X by Y using radix-4-modified Booth's algorithm, the three bits of the multiplier part will be grouped and will be encoded into one of $\{-2, -1, 0, 1, 2\}$ as per Table 1.

The modified Booth encoder, which is shown in Figure 8(a), is implemented using some logic gates. The partial products are generated using the Booth decoder as shown in Figure 8(b). The block diagram of the Booth-encoded Wallace tree multiplier [21] is shown in Figure 9.

While generating the partial products from the modified booth decoder, then we followed Fadavi-Ardekani's [23] sign extension prevention. Wallace Tree Carry Save Adder structure [24] has been used for adding the Pi + 1 with Pi in a parallel fashion until the last two rows remained. The last two rows have been added using a very high-speed Carry Look-ahead Adder (CLA) to obtain the final multiplication result. The architecture as a block diagram of the designed 8×8 bit multiplier using modified Booth's algorithm is shown in Figure 9. Here the multiplicand is X and multiplier is $Y \cdot Y$ input is encoded by Booth encoder to generate the encoded signal which is used by the Booths decoder to generate the partial product term by taking X as the input. After generating all the partial products, the Wallace tree performs the addition operation in a parallel fashion. Finally CLA is used to complete the multiplication procedure of the two 8-bit numbers.

4. Loop Filter

Loop filter, which is a first-order lowpass filter, is used to remove the high-frequency components of the output of

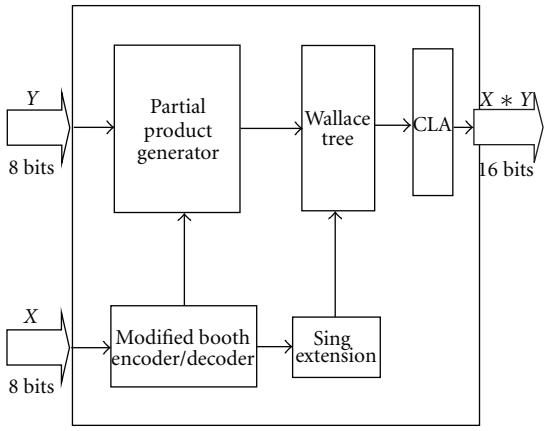

Figure 9: Block diagram of booth-encoded Wallace tree multiplier.

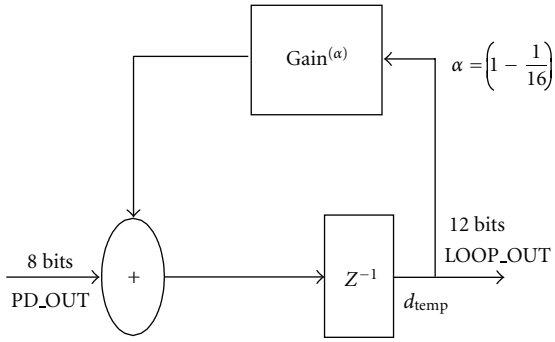

Figure 10: Block diagram of first order loop filter.

the phase detector given by (4). Figure 10 shows the block diagram of the first-order loop filter used in the DPLL-based FM demodulator system. The transfer function of the loop filter is given by

$$H(z) = \frac{1}{z - \alpha}. \tag{6}$$

Equation (6) can be implemented in hardware by the addition of the output signal from the phase detector (PD_OUT) and the register output multiplied by a coefficient $\alpha = (1 - 1/16) = 15/16 = .09375$, which is chosen to ensure the system stability. Multiplication by a factor of 1/16 has been implemented by 4-bit right shift instead of a multiplier.

5. Direct Digital Frequency Synthesizer

DDFS finds wide use as a component in modern communication system, radio detector, electronic warfare, high precision measurement system, and high precision biomedical applications. DDFS accepts arbitrary frequency as its reference frequency depending on the frequency control word and generates one or more frequencies. The DDFS architecture was first given in [25]. The arithmetic operations required to build a DDFS are a phase accumulator which generates the phase for generating the cosine waveform and a phase to amplitude converter. Various researches have been performed to design a high-performance circuit for phase-to-amplitude conversion as summarized in [26–28]. The quarter-wave symmetry ROM technique is very useful where a very low phase resolution has been used [29]. Many ROM compression techniques have been proposed, but for low-resolution bit, these techniques are not suitable as they maximize the error. This DDFS has been designed for waveform synthesis in DPLL-based FM demodulator. The ROM-based DDFS has been designed for use in the DPLL-based FM demodulator. In DPLL-based FM demodulator, the quadrature output from the DDFS is not required. Due to this fact, the ROM-based architecture (LUT) is considered to be superior to the CORDIC-based architecture [30] for the phase-to-amplitude conversion. To overcome the disadvantages of ROM-based DDFS, namely, high-power consumption and low speed, a pipelined ROM-based DDFS approach has been considered in the present work. Pipeline technique will help reduce the power consumption and also maximize the operating frequency. In the present work, the designed pipelined look-up table-based DDFS architecture is used as shown in Figure 11.

6. Fir Filter

The DDFS consists of a phase accumulator, a ROM look-up table, two 1's complementers, a pipelined register, and an XOR gate. The designed DDFS has a free running frequency of 1 MHz and requires 1024 sample values to define one cycle of a cosine signal. The DDFS generates the cosine signal waveform by addressing the cosine ROM LUT at a frequency set by an 18-bit control word. If the reference system clock (Fclk) is set to 100 MHz, then the frequency resolution will be 381.468 Hz. According to the accumulation rate in phase accumulator set by the FCW, the ROM produces the cosine waveform at that programmed frequency. In this implementation, the frequency control word (FCW) and the output bits have been chosen to be 18 bits and 8 bits, which provides spurious free dynamic range (SFDR) of 64.3 dB. As the design is pipelined, the frequency switching will suffer from 2-clock cycle latency.

MATLAB 7.4.0 version is used for the performance analysis of the designed two DDFS blocks. The floating point cosine wave generated using the MATLAB in-built function and cosine wave generated by proposed pipelined ROM-based DDFS has been analyzed. The results are shown in Figure 12. Hence the FLTPNT_COSINE is the MATLAB-generated cosine wave and FXDPOINT_COSINE is the cosine wave generated by our proposed ROM-based DDFS. The error between these two signals in the first quadrant has been shown in Figure 13 (as quarter wave symmetry property has been adopted). The minimum error is −0.0088 and maximum error is 0.0089 which is nothing but the quantization error (as 8 bits of amplitude has been considered in proposed design).

At the last stage of the receiver, a lowpass Finite Impulse Response (FIR) filter is used to perform the signal shaping. Here a 16-tap transposed FIR filter architecture [31] is used, as shown in Figure 14. This filter is essentially an averaging

Table 2: Timing, Area, and Power results for 2vp30-7ff896.

	Area				Timing (MHz)	Power (mW)
	No. of slice	No. of FF	No. of LUT	Gate count		
FM modulator	96	90	148	1,931	334.5	107.67
FM demodulator	233	240	436	6,400	131	108.67

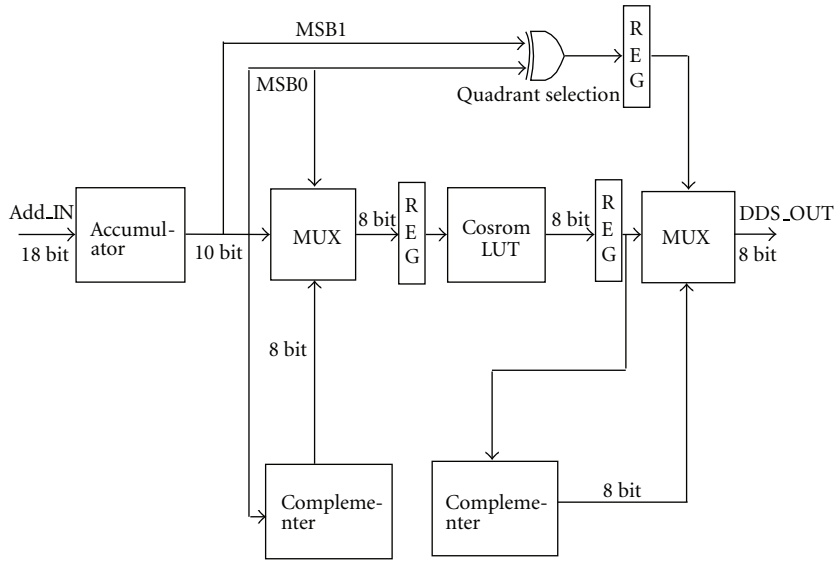

Figure 11: Block diagram of a pipelined ROM-based direct digital synthesizer.

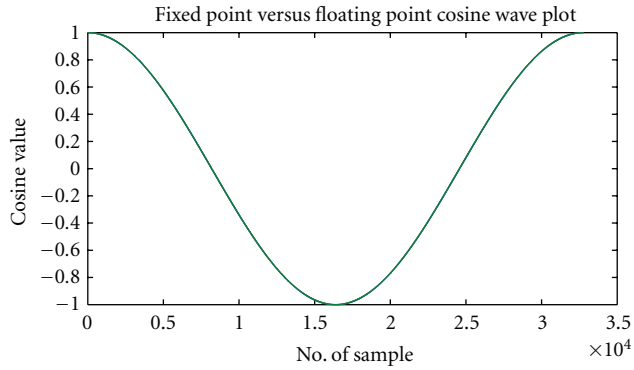

Figure 12: Plot of software-generated versus hardware-generated (using proposed ROM-based DDFS) cosine wave.

filter since its output is equal to the average value of its input values over the last n-tap samples where n is the number of taps used. As in direct form digital FIR filter the total propagation delay of the circuit increases more due to the addition of the 16 data samples, a transposed FIR filter architecture is chosen [32–34] in the present implementation. Here the coefficients are the same 1/16, and in reality 1/16 can be implemented by just 4-bit right shift operation. Hence no multiplier is required.

7. FPGA Implementation Details

7.1. Synthesis Results. The proposed FM demodulator has been described using the Verilog hardware description language and Xilinx ISE 9.2i is used for synthesis and FPGA implementation. Xilinx XCV2vp30-7FF896 device has been used as the target device for FPGA implementation, XST has been used as a synthesis tool, and XPower has been used for power calculation. The power is being calculated by simulation-based switching activities of all the signals. The synthesis results for the FM modulator and demodulator have been listed in Table 2. Table 3 listed the detailed dynamic power analysis result by applying a 100 Mbps data rate to the FM modulator and demodulator. Table 4 shows the componentwise implementation reports.

Table 3: Detailed power analysis results for 2vp30-7ff896.

	FM modulator	FM demodulator
Clock power	1.72 mW	1.72 mW
Input power	0.73 mW	0.33 mW
Logic power	0.03 mW	1.83 mW
Output power	2.81 mW	11 mW
Signal power	0.04 mW	3.23 mW
Total power	107.67 mW	108.67 mW

7.2. Simulation Results. For the post place and route simulation in FPGA, the Modelsim-Xe 6.3c Starter version from Mentor Graphics is used as a logic simulator. The modulated response of a 10 KHz triangular wave is shown in Figure 15. The demodulated response of the FM-modulated signal is

Table 4: Component wise FPGA implementation result.

Module name	Maximum frequency (MHz)	FPGA resources used			Gate count
		Slices	Slice FF	No. of LUTs	
FM data generator	1045.8	7	14	0	139
Interpolator	386	12	11	16	318
Pipelined DDFS	334.5	75	55	120	1,325
Accumulator	335.2	9	18	18	357
Phase detector	131	46	0	71	1,106
Loop filter	221	12	12	23	372
DDFS	155.6	87	25	159	1,371
FIR filter	366.7	97	180	194	3,750

Table 5: Comparison result for DDFS.

Reference	Ours (pipelined)	Xilinx core [35]	CORDIC [36]	Nicholas [37]	[38]	[39]
Output SFDR	−64.3 dBc	N/A	−90.3 dBc	−90.3 dBc	N/a	−95.2 dBc
L, W, K^*	18, 10, 8	30, 10, 14	32, 15, 14	32, 15, 14	16, N/A, N/A	32, 15, 14
Area	75 Slices	212 Slices	440 Slices	101 Slices	248 CLBs	101 Slices
Frequency	334.5 MHz	72 MHz	113 MHz	121 MHz	62.1 MHz	121 MHz
BRAMs	0	0	0	2	0	2
Device	XC2VP 30-7	N/A	XCV 300-4	XCV 300-4	XC 4020xl	XCV 300-4

* where L is the accumulator length, W is the word length of truncated phase, and K is the word length of DDFS output.

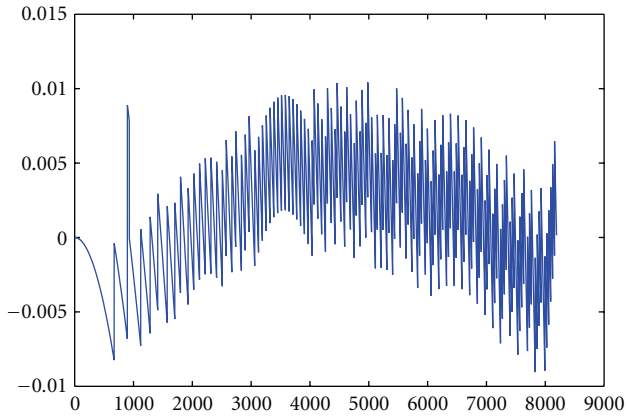

Figure 13: Error plot between software-generated and hardware-generated cosine wave.

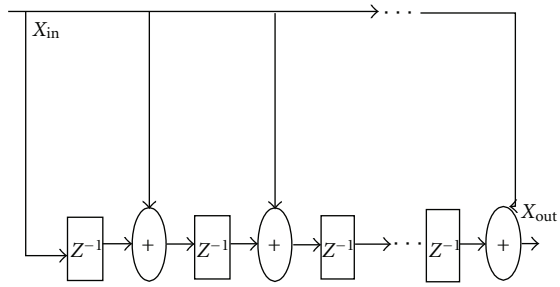

Figure 14: Transposed FIR filter block diagram.

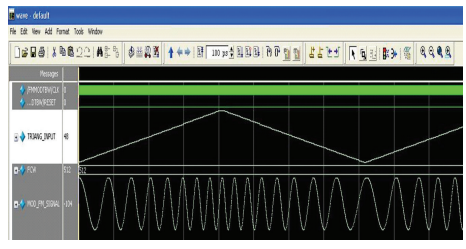

Figure 15: Simulation result of digital FM modulator.

as shown in Figure 16. Carrier frequency has been taken to be 1.5 MHz by setting the frequency control word to 512 and the input clock to 100 MHz with a modulation index of 10. In Figure 13, the signals from the top represent the input triangular wave (TRIANG_INPUT), the frequency control word (FCW) for setting the carrier frequency, and the modulated input data (FM_MOD). For the triangular wave the modulated signals from the top are the demodulated output data (FM_DEMOD), the modulated input data (FM_MOD), the FIR filter output (TRI_FIR), the loop filter output data (TRI_LOOP), the DDS output data (TRI_DDS), and finally for phase detector output (TRI_PD) shown in Figure 16. At the initial simulation phase, the demodulated output overshoots since the phase synchronization is in convergence phase and after that the system is stable.

7.3. On-Chip-Verified Results. The designed system has been implemented using the Xilinx impact tool to the Virtex-2 Pro University Board. Xilinx Chipscope-Pro 9.2i has been used for capturing the demodulating (FM_DEMOD) data for verifying the FPGA implementation result of the designed circuit. Here 2048 samples of the output have been captured after implementing the design into FPGA. The captured output results are shown in Figure 17 for triangular input. It can be concluded from inspection of these figures that

Table 6: FPGA resource usage comparison result with other FM demodulators.

Architecture using Xilinx Spartan3 3S200FT256-4	Time		Area		
	Delay (ns)	Frequency (MHz)	Slices	Slices FF	LUT
PLL (optimized) [7]	9.725	102.828	491	548	721
Sigma Delta Arch. [11]	2427 out of 3071 slices in Xilinx Virtex2 XC2V500 device				
Proposed implementation	12.948	77.3	237	244	437
	234 out of 3072 slices in Xilinx Virtex2 XC2V500 device				

Table 7: Area utilization comparison with other FM demodulator implementations [5, 6].

Without optimization/no. of gate		With optimization/no. of gate	
Proposed design	Existing architecture	Proposed design	Existing architecture
Phase detector: 389	Phase detector: 422	Phase detector: 413	Phase detector: 616
Loop filter: 202	Loop filter: 200	Loop filter: 277	Loop filter: 297
FIR filter: 3213	FIR filter: 2367	FIR filter: 3696	FIR filter: 3511
DDS/NCO: 608	DDS/NCO: 1534	DDS/NCO: 828	DDS/NCO: 1833
Total FM: 4844	Total FM: 10545	Total FM: 6052	Total FM: 14314

Table 8: Timing analysis comparison with other FM demodulator implementation [5, 6].

Without optimization		With optimization	
Proposed design	Existing architecture	Proposed design	Existing architecture
Operating frequency 149.4 MHz	Operating frequency 143.7 MHz	Operating frequency 158.1 MHz	Operating frequency 155.8 MHz

the designed circuit can effectively demodulate the FM input signal back to its original form.

7.4. Comparison Results. By optimizing the basic components of the FM demodulator, the reduction of the hardware usage and improvement in the performance has been done. Table 5 has summarized the comparison result with other ROM compression techniques while implementing a direct digital synthesis. In this context, Table 6 shows the comparison result with other existing FPGA implementations of FM demodulator [7, 11].

The proposed circuit has been synthesized using the Leonardo Spectrum 2005b.24 Level 3 from Mentor Graphics using the TSMC 350 nm (typical) as a target technology library. During the synthesis, speed has been considered as the main constraint for the designed circuit. Another FM receiver circuit has also been designed and synthesized using the Leonardo Spectrum 2004a.63 from Mentor Graphics and TSMC 350 nm (Fast) as a target technology library. From Tables 7 and 8, it is observed that the FM demodulator designed in this chapter is better in performance compared to the available DPLL-based FM demodulator [5, 6].

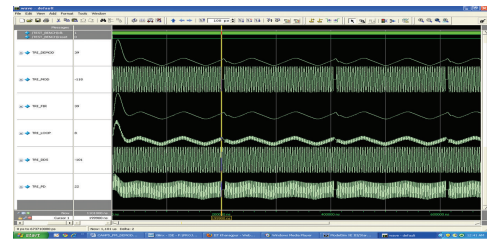

Figure 16: Simulation result of FM demodulator for Triangular wave modulated input.

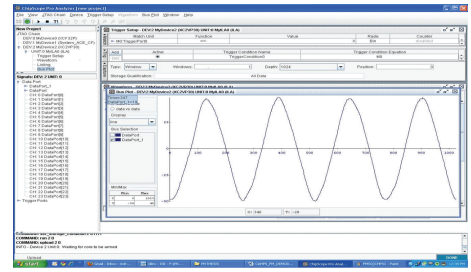

Figure 17: On-chip-verified result of the FM demodulator output for the triangular wave-modulated input.

8. Conclusions

A new high-performance digital FM modulator and a digital phase-locked loop-based FM demodulator have been proposed in this paper. The FM modulator and demodulator are designed to satisfy the constraint for the application in personal wireless communication and digital audio broadcasting. Individual componentwise optimization has made the overall design superior than other implementations. FPGA implementation of the proposed design has been carried out for quick prototyping of the digital FM modulator and demodulator chip. The simulation and synthesis result of FM modulator shows that the digital up conversion is very much possible as it can achieve maximum clock frequency of 334.5 MHz. From the on-chip-verified result it

can be clearly seen that the proposed FM demodulator can demodulate the signal back in its original form by consuming only 6.4 K equivalent gate count. The comparison results for both FPGA and ASIC implementations have shown that the proposed design is superior to the existing digital FM chips. Hence it is concluded that the designed high-performance FM modulator and demodulator can be easily fitted into the next generation software-defined radio-based handset where low power and minimum hardware utilization with the maximum clock frequency are desired features.

References

[1] S. Twelves and C. J. Kikkert, "Investigation into the effects of quantisation noise on a digital FM modulator," in *Proceedings of the IEEE TENCON Digital Signal Processing Applications (TENCON '96)*, vol. 1, pp. 317–322, November 1996.

[2] J. Lahti and M. Niemisto, "A digital frequency modulator circuit for a dual-mode cellular telephone," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '98)*, pp. 405–408, June 1998.

[3] R. Unsikartano, J. Niittyahti, and M. Renfors, "Area-optimized FPGA implementation of a digital FM modulator," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '99)*, vol. 4, pp. 360–362, July 1999.

[4] S. Twelves and C. J. Kikkert, "Performance of a digital stereo FM modulator with reduced output resolution," *IEEE Transactions on Broadcasting*, vol. 43, no. 1, pp. 104–113, 1997.

[5] J. P. M. Brito and S. Bampi, "Design of a digital FM demodulator based on a 2nd order all-digital phase-locked loop," in *Proceedings of the 20th Symposium on Integrated Circuits and System Design (SBCCI '07)*, Session: Circuits for Communications—Part 3, pp. 137–141, 2007.

[6] J. P. M. Brito and S. Bampi, "Design of a digital FM demodulator based on a 2nd-order all-digital phase-locked loop," *Analog Integrated Circuits and Signal Processing*, vol. 57, no. 1-2, pp. 97–105, 2008.

[7] http://www.opencores.org/?do=project&who=simple_fm_receiver.

[8] E. B. Ronald, *Phase Locked Loop, Theory, Design, and Applications*, McGraw-Hill, New York, NY, USA, 2003.

[9] http://www.altera.com/literature/cp/fpga-cores-for-sdr.pdf.

[10] C. Zhang, C. R. Anderson, and P. M. Athanas, "All digital FPGA based FM radio receiver," in *Proceedings of the SDR Forum Technical Conference*, 2007.

[11] F. Yu, "FPGA implementation of a fully digital FM demodulator," in *Proceedings of the 9th IEEE Singapore International Conference on Communication Systems (ICCS '04)*, pp. 446–450, September 2004.

[12] J. K. Holmes and C. R. Tegnelia, "A second-order all-digital phase-locked loop," *IEEE Transactions on Communications*, vol. 22, no. 1, pp. 62–68, 1974.

[13] B. S. Song and I. S. Lee, "Digital FM demodulator for FM, TV, and wireless," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 42, no. 12, pp. 821–825, 1995.

[14] N. Boutin, "An arctangent type wideband PM/FM demodulator with improved performance," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. 5–9, 1992.

[15] G. Pasternack and R. L. Whalin, "Analysis and synthesis of a digital phase-locked loop for FM demodulation," *Bell System Technical Journal*, vol. 47, pp. 2207–2237, 1968.

[16] C. Kelly and S. Gupta, "The digital phase-locked loop as a near-optimum FM demodulator communications," *IEEE Transactions Communications*, vol. 20, no. 3, pp. 406–411, 1972.

[17] D. R. Stephens, *Phase-Locked Loops for Wireless Communications Digital, Analog and Optical Implementations*, Kluwer Academic Publishers, 2nd edition, 1998.

[18] J. G. Proakis and D. G. Manolakis, *Digital Signal Proessing*, Prentice Hall, Upper Saddle River, NJ, USA, 4th edition, 1996.

[19] K. Ogata, *Modern Control Engineering*, Prentice Hall, Upper Saddle River, NJ, USA, 3rd edition, 1997.

[20] K.-S. Cho, J.-O. Park, J.-S. Hong, and G.-S. Choi, "54×54-bit radix-4 multiplier based on modified booth algorithm," in *Proceedings of the 13th ACM Great Lakes Symposium on VLSI*, pp. 233–236, 2003.

[21] H. L. Lin, R. C. Chang, and M. T. Chan, "Design of a novel radix-4 Booth multiplier," in *Proceedings of the IEEE Asia-Pacific Conference on Circuits and Systems (APCCAS '04)*, vol. 2, pp. 837–840, December 2004.

[22] R. Hussin, A. Y. M. Shakaff, N. Idris, Z. Sauli, R. C. Ismail, and A. Kamarudin, "An efficient modified booth multiplier architecture," in *Proceedings of the International Conference on Electronic Design (ICED '08)*, pp. 1–4, December 2008.

[23] J. Fadavi-Ardekani, "M × N booth encoded multiplier generator using optimized Wallace trees," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 1, no. 2, pp. 164–167, 1993.

[24] S. Kumar, K. Forward, and M. Palaniswami, "A fast-multiplier generator for FPGAs," in *Proceedings of the 8th International Conference on VLSI Design*, pp. 53–56, January 1995.

[25] J. Tierney, C. Rader, and B. Gold, "A digital frequency synthesizer," *IEEE Transaction on Audio and Electroacoustics*, vol. 19, no. 1, pp. 48–57, 1971.

[26] D. A. Sunderland, R. A. Strauch, S. S. Wharfield, H. T. Peterson, and C. R. Cole, "CMOS/SOS frequency synthesizer LSI circuit for spread spectrum communications," *IEEE Journal of Solid-State Circuits*, vol. 19, no. 4, pp. 497–506, 1984.

[27] H. T. Nicholas III and H. Samueli, "A 150-MHz direct digital frequency synthesizer in 1.25-μm CMOS with -90-dBc spurious performance," *IEEE Journal of Solid-State Circuits*, vol. 26, no. 12, pp. 1959–1969, 1991.

[28] K. A. Essenwanger, V. S. Reinhardt, and A. Zarowin, "Sine output DDSs a survey of the state of the art," in *Proceedings of the IEEE International Frequency Control Symposium*, pp. 370–378, May 1998.

[29] A. Bellaouar, M. S. O'brecht, A. M. Fahim, and M. I. Elmasry, "Low-power direct digital frequency synthesis for wireless communications," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 3, pp. 385–390, 2000.

[30] S. Nahm, K. Han, and W. Sung, "A CORDIC-based digital quadrature mixer: comparison with a ROM-based architecture," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '98)*, vol. 4, pp. 385–388, June 1998.

[31] A. T. Erdogan and T. Arslan, "Low power multiplication scheme for FIR filter implementation on single multiplier CMOS DSP processors," *Electronics Letters*, vol. 32, no. 21, pp. 1959–1960, 1996.

[32] M. Mehendale, S. D. Sherlekar, and G. Venkatesh, "Techniques for low power realization of FIR filters," in *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC '95)*, vol. 3, pp. 404–416, September 1995.

[33] "An introduction to digital filters," an9603, Application note, Intersil, 1999.

[34] C. Nagendra, M. J. Irwin, and R. M. Owens, "Area-time-power tradeoffs in parallel adders," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 43, no. 10, pp. 689–702, 1996.

[35] Xilinx Core Generator System, http://www.xilinx.com/products/intellectual-property/index.htm.

[36] J. Vankka, "Methods of mapping from phase to sine amplitude in direct digital synthesis," in *Proceedings of the 50th IEEE International Frequency Control Symposium*, pp. 942–950, June 1996.

[37] H. T. Nicholas and H. Samueli, "An analysis of output spectrum of direct digital frequency synthesizers in the presence of phase-accumulator truncation," in *Proceedings of the 41st Annual Frequency Control Symposium,*, pp. 495–502, 1987.

[38] F. Cardells-Tormo and J. Valls-Coquillat, "Optimized FPGA-implementation of quadrature DDS," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '02)*, pp. V/369–V/372, May 2002.

[39] M. Park, K. Kim, and J. A. Lee, "CORDIC-based direct digital frequency synthesizer: comparison with a ROM-based architecture in FPGA implementation," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 83, no. 6, pp. 1282–1285, 2000.