FPGA Design and Implementation of Selectable M-PSK Modulators

Shahana K[1], K.S.Parikh[2]

[1] Satcom Experiments & Operations Division, Space Applications Centre, Delhi, India

[2] Satcom Navigation & Applications Area, Space Applications Centre, Ahmedabad, India

Abstract: **As PSK modulation techniques are power and bandwidth efficient, it is the mostly preferred and popular modulation scheme in wireless communications. This paper deals with the design and implementation of selectable M-PSK modulators viz. BPSK, QPSK, 8PSK, 16PSK and 32PSK on Spartan-3AN FPGA device using Xilinx System Generator and VHDL. The communication channel also includes channel encoding like CCITT Scrambler (V.35), Differential encoder, Convolutional encoder (K=7, rate=1/2) for error control. The prime objective is to study the bandwidth efficiency of modulated signal as the modulation index increases. Also, the ease and versatility of FPGA device is also studied in terms of flexibility to change the design parameters like data rate, carrier frequency etc before the design is implemented.**

Keywords: **QPSK, BPSK, RRC, FPGA, M-PSK, symbol rate.**

I. INTRODUCTION

Fundamental to all wireless communications is modulation, the process of superimposing the data to be transmitted on the radio carrier. Most wireless communications today are digital, and with the scarce spectrum available, the type of modulation is more critical than it has ever been. The applications of digital modulation techniques has evolved and grown in tandem with the digital communications industry. This growth, in turn, has spawned an increasing need to seek software implementation of digital modulation types on the most popular and low cost FPGA platforms. The amplitude shift keying (ASK), frequency shift keying (FSK), Phase shift keying (PSK) and quadrature amplitude modulation (QAM) are all forms of digital modulation techniques. Analog transmission of digital data like ASK is very susceptible to noise and therefore is seldom used except for very low speed telemetry circuits. FSK has less bandwidth efficiency, low data rates usage, poorer error performance than PSK or QAM and consequently is seldom used for high-performance digital radio systems. So, PSK is the commonly used digital modulation technique. The PSK waveforms have constant envelope like FSK but discontinuous phase transitions from symbol to symbol. In [1] MPSK modulator is used in quasi-optical wireless array applications, and MPSK Modulation scenarios to Compressed Image Communication in Mobile Fading Channel [2]. [3] explains the implementation of QPSK, 8PSK modulators for satellite communication using the concept of Direct Digital Frequency Synthesis (DDFS) and COordinate Rotation Digital Computer (CORDIC) algorithm Phase shift keying (PSK). The motivation of the work is to design, implement MPSK modulation schemes on a basic low cost FPGA board and measure their bandwidth efficiency.

II.   MULTIPLE PHASE SHIFT KEYING (M-PSK)

Phase shift keying is one of the most efficient digital modulation techniques. It is used for very high bit rates. In PSK, the phase of the carrier is modulated to represent Binary values. In BPSK, the carrier phase is used to switch the phase between 00and1800 by digital polar format. Hence it is also known as phase reversal keying or bi-phase modulation (BPSK).

More sophisticated forms of PSK exist. In Mary or multiple phase-shift keying (MPSK), there are more than two phases, usually four (0, +90, -90, and 180 degrees) or eight(0,+45,-45, +90, -90, +135, -135, and 180 degrees). In the case of four

phases (m = 4), the MPSK scheme is called quadrature phase-shift keying (QPSK), and each phase shift is represented by two symbols. In the case of eight phases (m = 8), the MPSK scheme is known as eight phase-shift keying (8PSK), and each phase shift is represented by three symbols. MPSK provides faster data transmission relative to the number of phase changes per unit time, in comparison to BPSK. In general, for higher transmission rates, more bandwidth is required. In MPSK modulation, the signal is hidden in the phase, therefore, there is enormous saving in the consumption of scarce bandwidth.

In BPSK, each symbol is represented by a single bit resulting in two values of θ (t), one symbol represents 0, and the other symbol represents 1. Since there are 2π radians per cycle of carrier, and as the symbols can take only two distinct values, we can choose θ (t) such that: θ1 (t) representing a one, be 0, and let θ0 (t) representing a zero, be π. Doing so, we obtain:

$S_1(t) = \sqrt{E_s} . \cos(2\pi f_c t + \pi)$,

$S_2(t) = \sqrt{E_S} . \cos(2\pi f_c t + 0)$,

Where $\sqrt{ES}$ is the peak amplitude of the modulated sinusoidal carrier, $S_1(t)$ is the BPSK signal that represents a zero, and $S_2(t)$ is the BPSK signal that represents a one. Table 1 represents the symbol mapping rule and Fig.1 the signal constellation.

Table 1: Mapping rules for BPSK

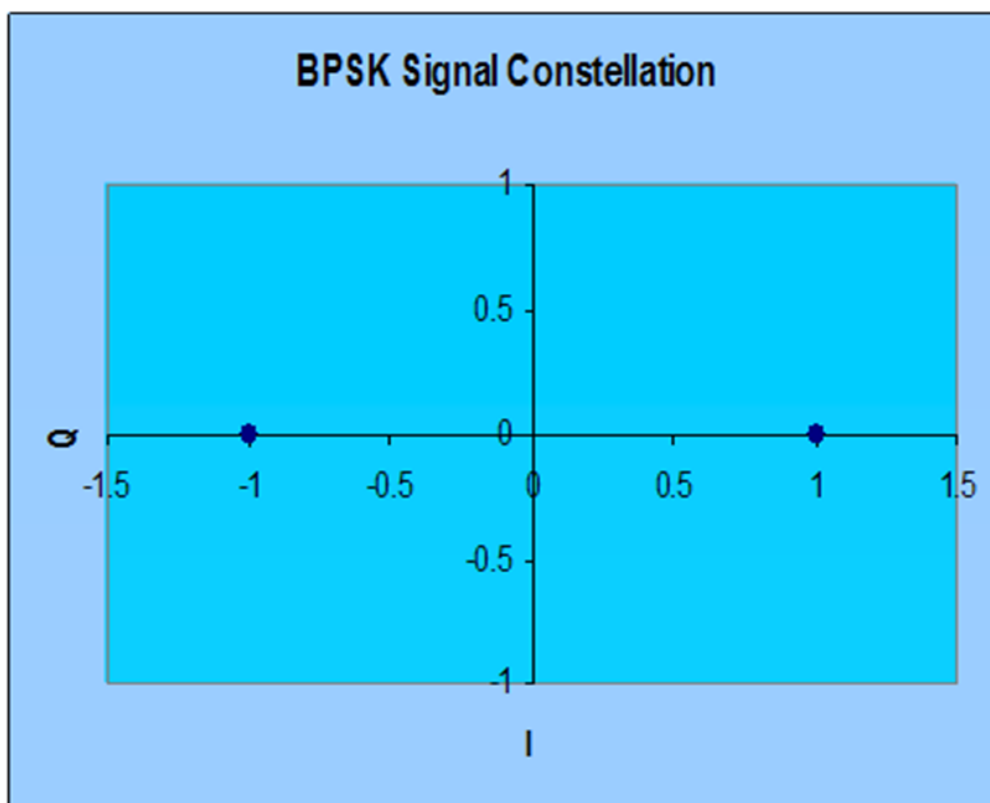| Symbol	Bits	Phase (Deg.)	I
S1	0	45	1
S2	1	135	-1

Fig.1 BPSK constellation

Digital phase modulation is not limited to the simple binary BPSK. By grouping bits and choosing the phase modulation appropriately, we obtain M-ary PSK. For M=4, the bits are grouped into pairs, called 'dibits', and the resulting signal is known as quadrature phase shift keying (QPSK).

Page | 75

(a) Symbol mapping:

In QPSK, there are four symbols, each symbol represents a dibit value, resulting in four values for θ(t), the time-varying phase modulation of digital pass band signal. Suppose we use the mapping scheme listed in Table 2 to assign phase modulation to each of the four possible symbols.

Table 2 : Mapping rules for QPSK

Symbol	Bits	Phase (Deg.)	I	Q
S1	00	45	1	1
S2	01	135	-1	1
S3	11	225	-1	-1
S4	10	315	1	-1

A constellation diagram shows the location of symbols in complex signal space. The horizontal axis indicates the real or in-phase component, equivalent to the amplitude of the cosine portion of the quadrature carrier. The vertical axis indicates the imaginary or quadrature component, equivalent to the amplitude of the sine portion of the quadrature carrier. The amplitude or instantaneous energy, of a symbol is a measure of its distance from the origin while the phase angle is a measure of its angular displacement from the positive horizontal axis. The QPSK constellation obtained from the symbol map of Table 2 is shown in Fig.2.

The four symbols are represented according to the mapping of Table 2. Draw a circle passing through the 4 points, the circle represents a locus of constant signal energy, meaning any point on this circle requires the same amount of transmitter power.
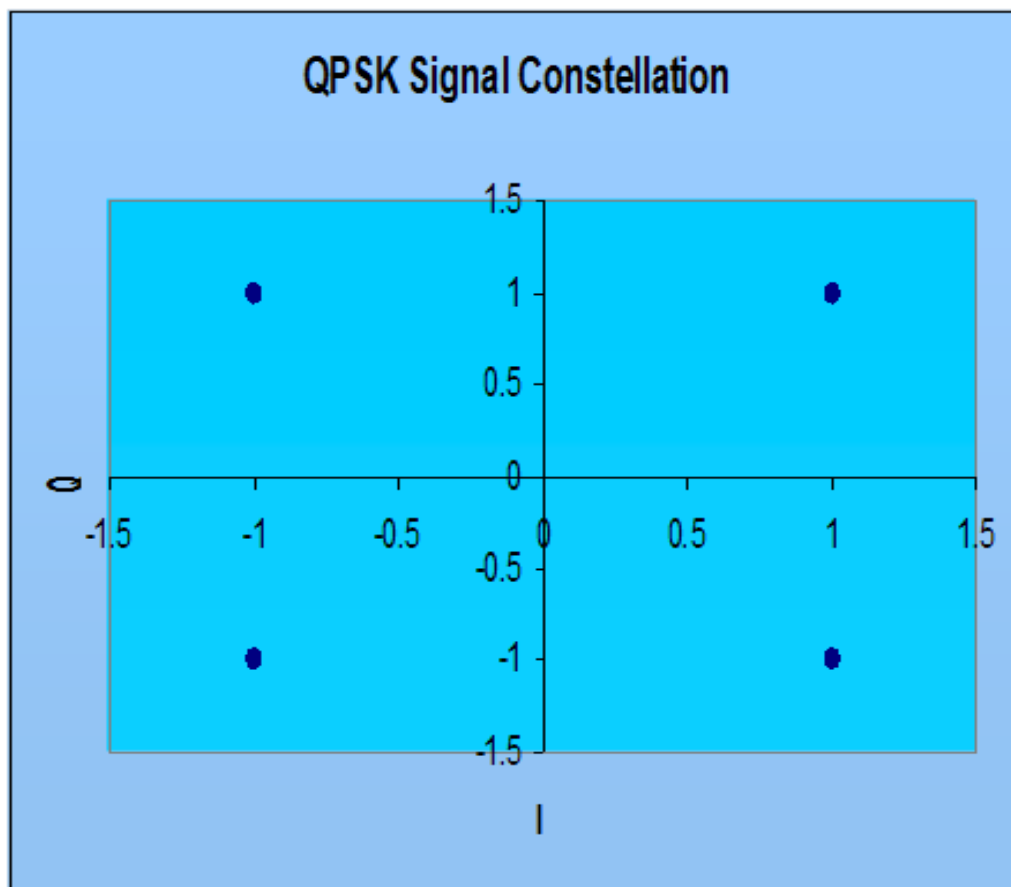

Fig.2   QPSK Constellation

With 8-PSK, three bits are encoded producing 8 different output phases. To encode eight different phases, the incoming bits are encoded in groups of three, called tribits ($2^3 = 8$). Fig. 3 shows the 8-PSK signal constellation.
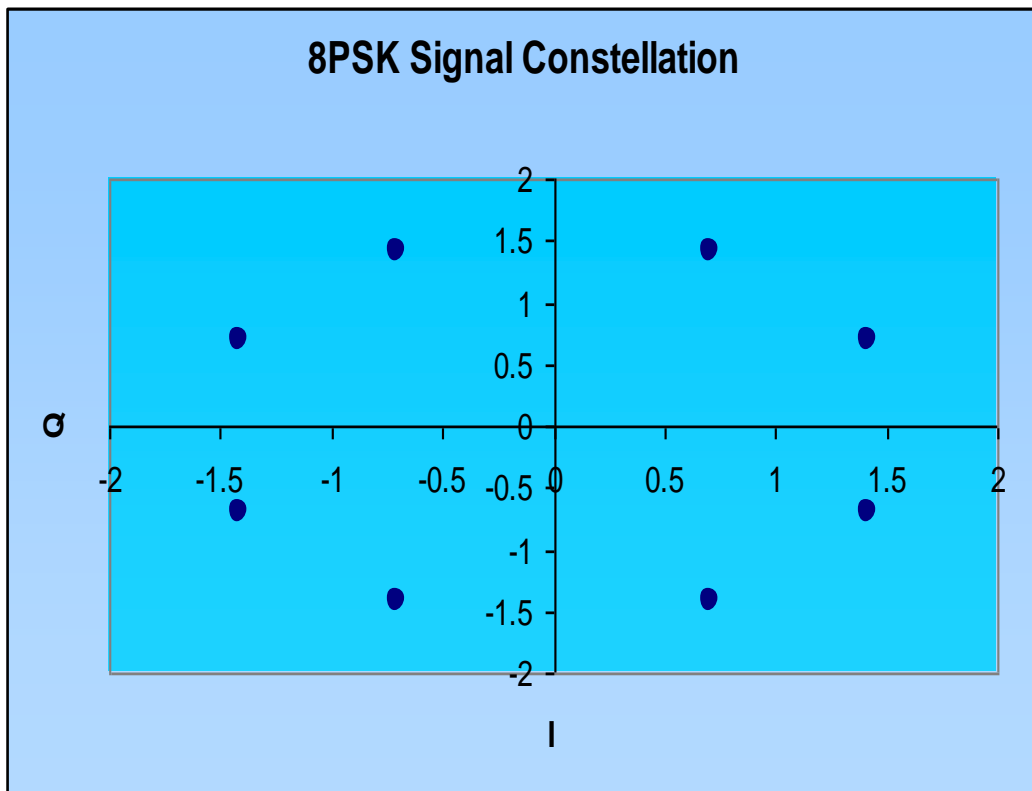

Fig.3   8PSK Constellation

Table 3: Mapping rules for 8 PSK

Symbol	Bits	Phase (Deg.)	I	Q
S1	000	22.5	1.414	0.707
S2	001	67.5	0.707	1.414
S3	011	112.5	-0.707	1.414
S4	010	157.5	-1.414	0.707
S5	110	202.5	-1.414	-0.707
S6	111	247.5	-0.707	-1.414
S7	101	292.5	0.707	-1.414
S8	100	337.5	1.414	-0.707

16-PSK is an M-ary encoding technique where four bits (quad bits) are encoded producing 16 distinct output phases . With 16-PSK, n = 4 and M = 16; therefore, the minimum bandwidth and baud equal one-fourth the bit rate (fb/4). Fig. 4 shows the 16PSK signal constellation.

32-PSK is an M-ary encoding technique where five bits are combined, producing 32 different output phases. With 32-PSK, n = 5 and M = 32; therefore, the minimum bandwidth and baud equal one-fourth the bit rate (fb/5). Fig. 5 shows the 32-PSK signal constellation.
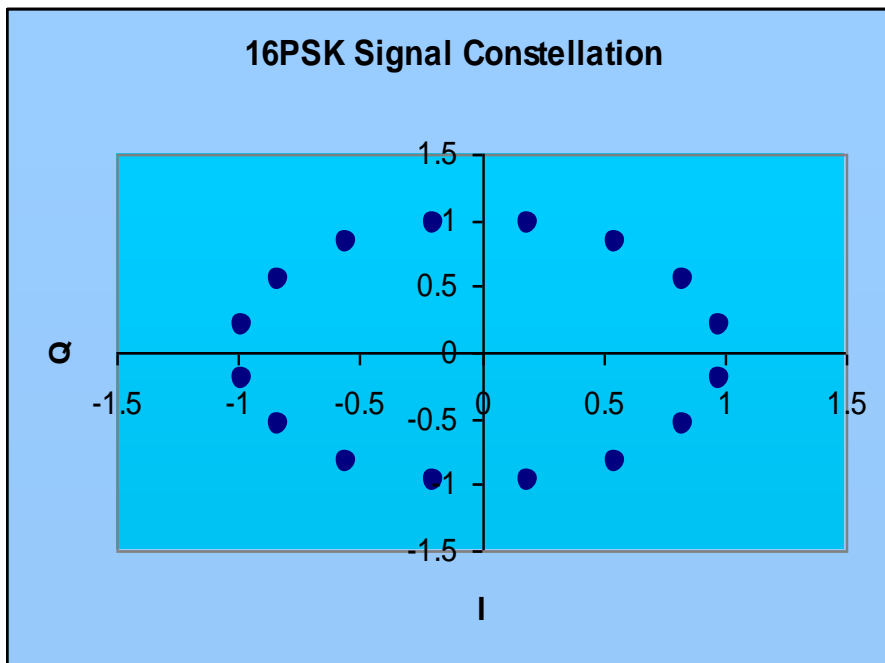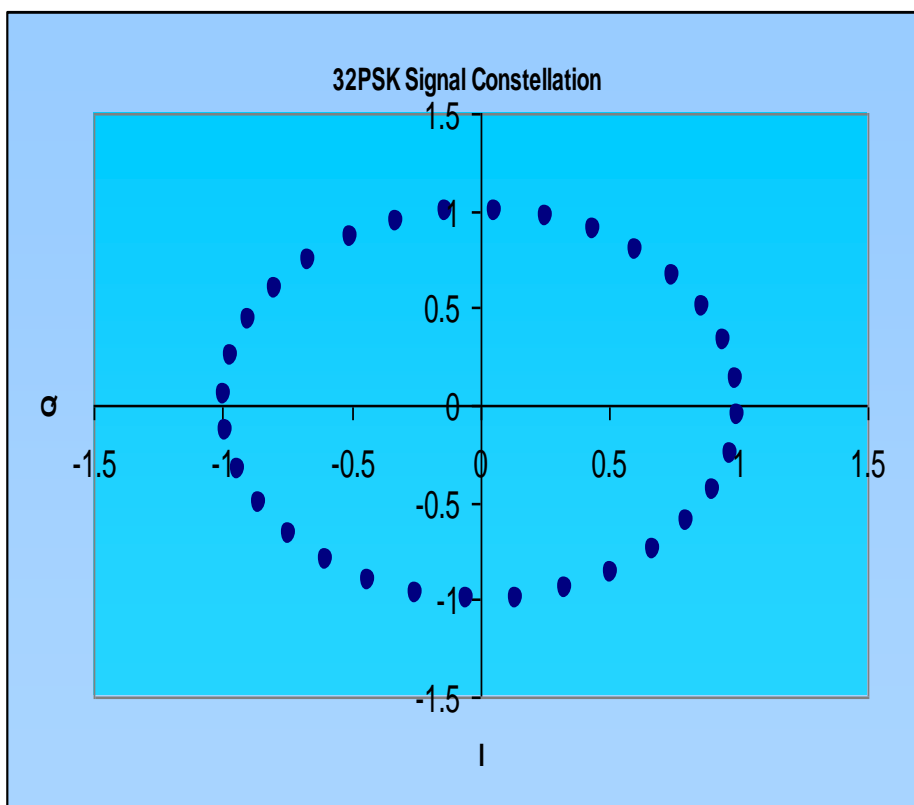
Fig.4 16PSK Constellation

Fig.5   32PSK Constellation

(b) Spectral Containment:

An important characteristic of any digital modulation scheme is the ratio of data bit rate to the transmission bandwidth. In order to control the signal bandwidth, the baseband symbols are filtered prior to modulating them onto the carrier using raised-cosine filtering for pulse shaping. An important parameter that can be chosen in raised-cosine filter design known as excess bandwidth or 'roll-off' factor controls the smoothness of the pulse shape and determines the signal bandwidth. As a rule of thumb, the bandwidth, B, of a PSK signal is:

$B = f_{sym} \cdot (1+\alpha)$; $f_{sym} = R_b /m$

Where f_{sym} is the symbol rate, R_b is the data rate, m=1,2,3,4,5 and α is the filter roll-off factor.

Smaller the filter roll-off factor, smaller will be the signal bandwidth; however, doing so yields a signal from which symbol timing recovery is difficult as the symbol timing recovery schemes rely on abrupt transitions in the baseband signal in order to achieve timing synchronization. As we increase the amount of lowpass filtering we apply to the baseband signal, the symbol transitions become so smooth that they are impossible to identify. Reasonable filter roll-off factors are from 0.25 to 0.5.

III.   ERROR CONTROL CODING

While modulation plays a key role in improving the spectral efficiency in wireless design, other aspects influencing it is the use of forward error correction (FEC) techniques which can greatly improve the BER. Error coding methods add extra bits to the signal so that errors can be detected and corrected. These extra bits add overhead to the transmitted signal, reducing the data rate, but normally an acceptable tradeoff for the single-digit dB improvement in carrier-to-noise ratio. Such coding gain is in practice in almost all wireless systems.

(a) Scrambler:

In telecommunications, a scrambler is a device that eliminates undesired long sequences of 0 or 1 by manipulating the data stream before transmitting. The manipulations are undone by a descrambler at the receiving side. It is widely used in satellite communications, radio relay systems and PSTN modems. Scrambling can be done before FEC or after the FEC, just before the modulation. A scrambler in this context has nothing to do with encrypting, as the purpose is to give the transmitted data useful carrier synchronization properties. The two main reasons of scrambling are : (1) it facilitates timing recovery at the receiver by eliminating long sequences consisting of '0' or '1'. (2) It eliminates the dependency of received signal power spectrum from the transmitted data, making it more dispersed to meet maximum power spectral density requirements (because if the power is concentrated in a narrow frequency band, it can interfere with adjacent channels due to the cross modulation and the intermodulation caused by non-linearities of the receiving tract. The CCITT V.35 scrambler consists of 16 D flip-flops preset with D2B2 as shown in Fig.6. Bit 1 and 15 forms the input to Ex-OR gate-1 while the output is in feedback loop to the shift register and I/P to Ex-OR gate-2. The second I/P of Ex-OR gate-2 is the actual data. The scrambler output is given to the differential encoder.

(b) Differential Encoding:

Differential encoding scheme is used for monosemous signal reception when using some types of modulation like phase shift keying and quadrature amplitude modulation. It ensures that data to be transmitted depends on both the current bit (or symbol) and the previous one. To demodulate BPSK signal local oscillator of the receiver need *synchronous* with that of the transmitter. This is accomplished by a carrier recovery circuit. However, a carrier can be recovered in many ways like Costas loop, Squaring loop, Gardiner carrier recovery etc.
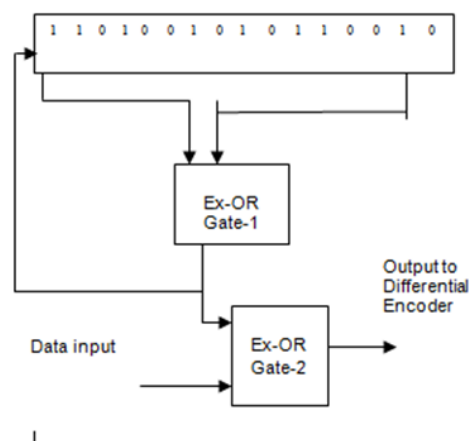

Fig. 6 Scrambler Schematic

With differential encoding, if a carrier is recovered incorrectly, the received data is inverted. Assuming that x_i is a bit intended for transmission, and y_i is a bit actually transmitted (differentially encoded, Fig. 7), if

Page | 79

$$y_i = y_{i-1} \oplus x_i, \qquad (1)$$

is transmitted, then on the decoding side

$$x_i = y_i \oplus y_{i-1}. \qquad (2)$$

can be reconstructed, where $\oplus$ indicates binary or modulo-2 addition.

The decoded data will always be correct as x_i depends only on a difference between y_i and y_{i-1} and not on their values. So, it does not matter whether the data stream is inverted or not.
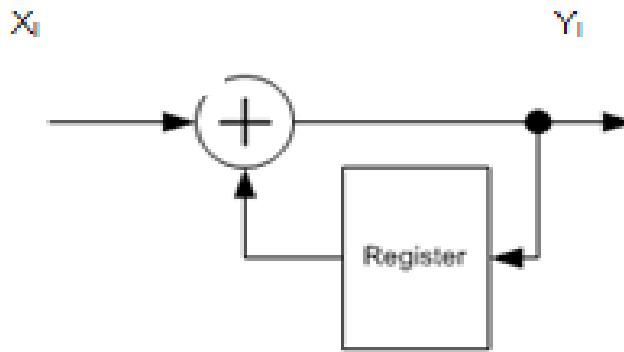

Fig. 7 Differential Encoder Schematic

(c)   Convolutional Encoding:

Communications channels are mostly affected by the additive white Gaussian noise in their environment. Data disturbances cannot be eliminated but their effect can be minimized by the use of Forward Error Correction (FEC) techniques in the transmitted data stream and decoders in the receiving system that detect and correct bits in error. Improvements in bit error rate performance can be realized for power-limited and bandwidth- limited systems typical of satellite communication channels. Using a rate ½ convolutionally encoded system, approximately 5dB of coding gain can be realized.
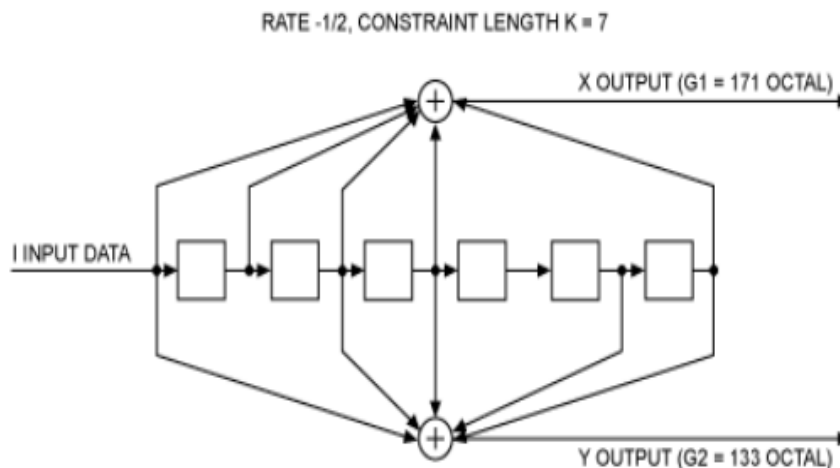

Fig. 8 Convolutional Encoder Schematic

Fig. 8 depicts the convolutional encoder typically used with an R=1/2, K=7. The term "K" defines the length or number of stages (n) in the shift register. In the drawing below there are seven stages to the shift register; therefore, K=7. The "R" term speaks of the output rate (frequency) relative to the input. In an R=1/2 system, the output rate is twice the input rate, it means two bits are output for every single bit that is input. In an R=2/3 system, three bits are output for every two bits that are input. The terms G1 and G2 defines the location of the taps on the shift register. As shown in the drawing, G1 = 171 represents the octal code for the upper connections to the shift register while G2 =133 describes the lower connections.

IV. DESIGN AND IMPLEMENTATION ON FPGA

The M-PSK modulator design is illustrated in Fig. 9(a). BPSK, QPSK, 8PSK, 16PSK modulators are implemented on a single FPGA with selection using switches. Either one, two, three or all the modulators are selectable i.e any combination of the modulators can be selected as per Table 4. 32PSK modulator was however implemented separately as it involves large look-up tables for symbol mapping (ROM table each for I & Q).
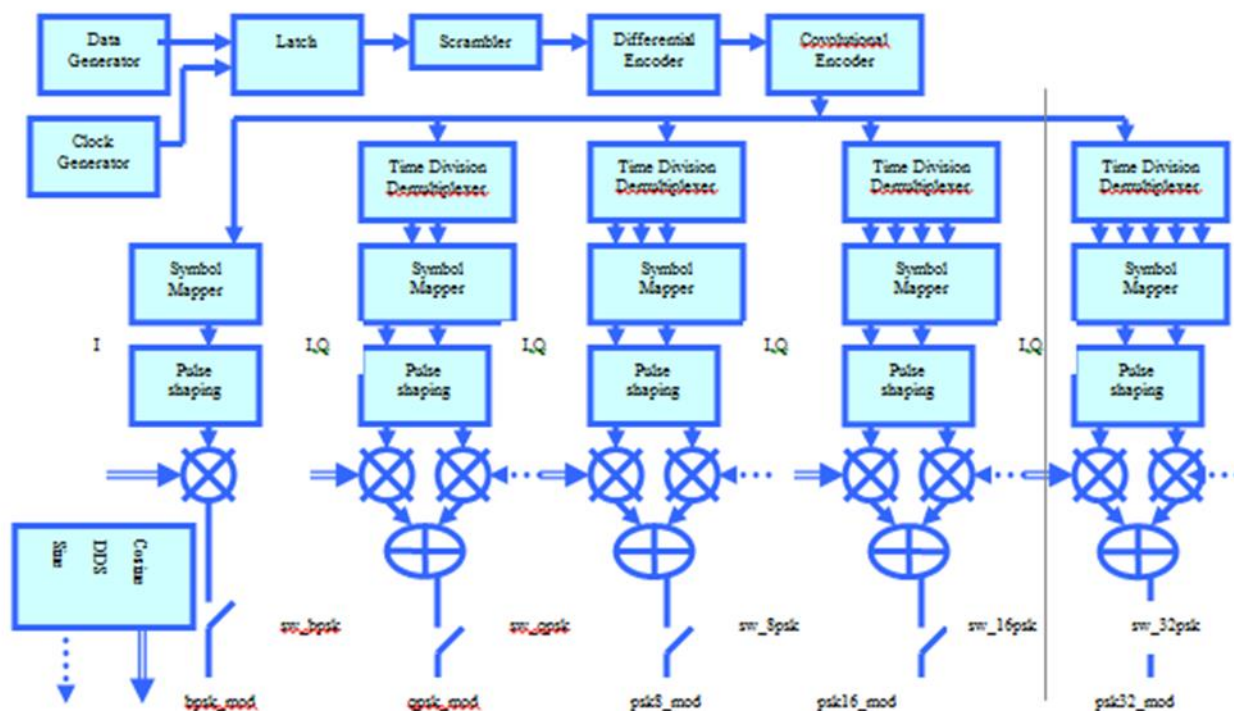

Fig.9(a)   Schematic of Selectable M-PSK Modulators

Table 4: M-PSK selection

sw_ bpsk	sw_ qpsk	sw_ 8psk	sw_ 16psk	sw_32psk	Modulator
1	0	0	0	-	bpsk
0	1	0	0	-	qpsk
0	0	1	0	-	8psk
0	0	0	1	-	16psk
1	1	1	1	-	All
-	-	-	-	1	32psk

Data generator block generates a Pseudo-Random sequence of 32 bits which is latched on the rising edge of a 50 KHz clock. The data is fed to a CCITT V.35 scrambler, differential encoder and convolutional encoder (k=7, rate= ½) for error control as shown in Fig. 9(b). The encoded data is converted from serial to parallel using a time division demultiplexer and is mapped using the respective symbol table to get the In-phase (I) and Quadrature-phase (Q) components.

The respective NRZ streams are shaped using raised cosine filters (roll off = 0.5) then multiplied with Cosine and Sine Carriers of

1MHz and added to give the modulated signal for transmission. The impulse response of the RRC filter is shown in Fig. 10. As M-PSK modulators requires m=1,2,3,4,5 bits to effect phase changes, the symbol rate also changes respectively as per the Table 5. The modulator outputs are selectable by using the slide switches available on the Spartan 3AN FPGA board.
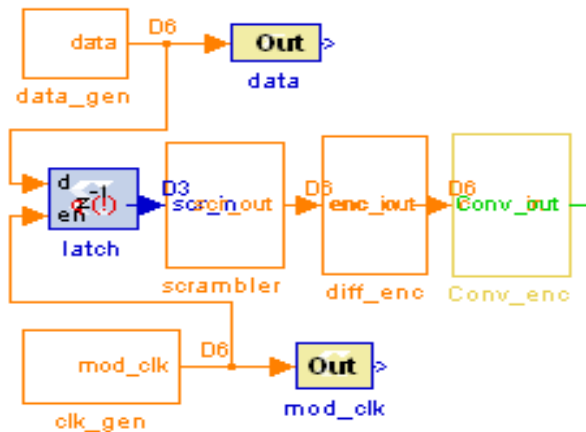
Fig.9(b)   Error correction scheme

The Convolution Encoder core is used for error correction with a constraint length of K=7, rate=1/2. The basic function of the Convolutional Encoder core is to get the incoming data into the constraint register a bit at a time, and the output bits are generated by modulo-2 addition of the required bits from the constraint register. The bits to be XOR'd are selected by the convolution codes depending on the generator polynomials G1, G2.

In a basic convolution encoder, two or three bits (depending on the encoder output rate) are transmitted over the channel for every input bit.. Convolution encoding is used to encode data prior to transmission over a channel. The received data is decoded by the classic Viterbi decoder.

Xilinx DDS Compiler is a Direct Digital synthesis block also called as NCO has been used to generate sinusoids (sine, cosine carriers) using a lookup table. A digital integrator (accumulator) generates a phase that is mapped by the lookup table into the output waveform.

The Xilinx FIR Compiler IP core implements a single rate Distributed-Arithmetic FIR filter. It accepts a stream of input data and computes filtered output with a fixed delay based on the filter configuration.

Simulation of the design flow was done using Simulink's system Generator, a system level modeling software from Xilinx [4]. This software can be used for designing and simulating DSP systems for FPGAs in a visual data flow environments such as MATLAB/ Simulink [5]. It offers flexibility to simulate the signal processing algorithms and verify their functionalities at the bit-level without translating the design into hardware. Design was simulated and verified using System Generator. The HDL Net list was generated for Spartan 3AN FPGA. The VHDL code was then synthesized, translated, mapped in ISE design Suite by modifying the constraints and pin-outs. Finally, the bit stream was generated and loaded on Spartan 3AN FPGA board to complete the implementation process.
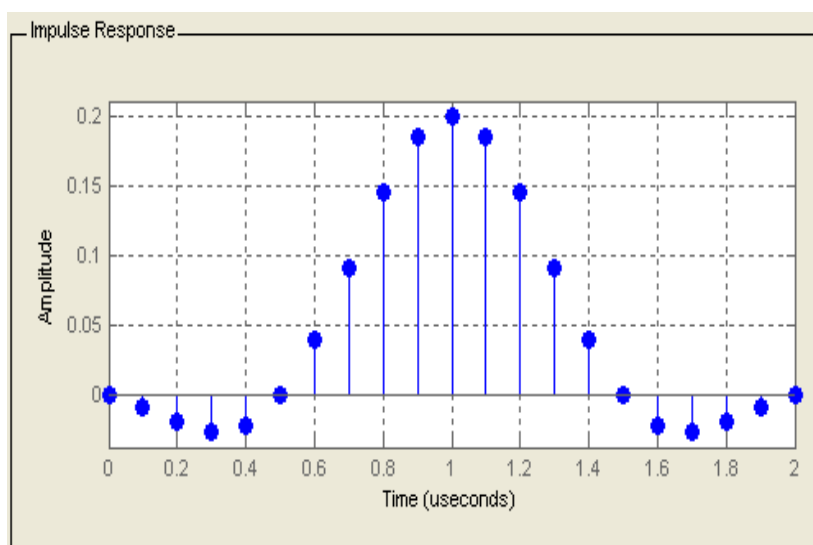

Fig.10   Impulse Response of RRC filter

V.   RESULTS AND DISCUSSION

(i) Initially M-PSK modulators were simulated using Simulink of Matlab before translating the design on FPGA. Fig. 11(a) shows the clock, data and FEC while Fig.11(b) shows the different modulator output waveforms.

(ii) The FPGA device utilization summary is as per Table 5. The top level RTL schematic and the component level RTL schematic of the implementation is shown in Fig. 12, 13.
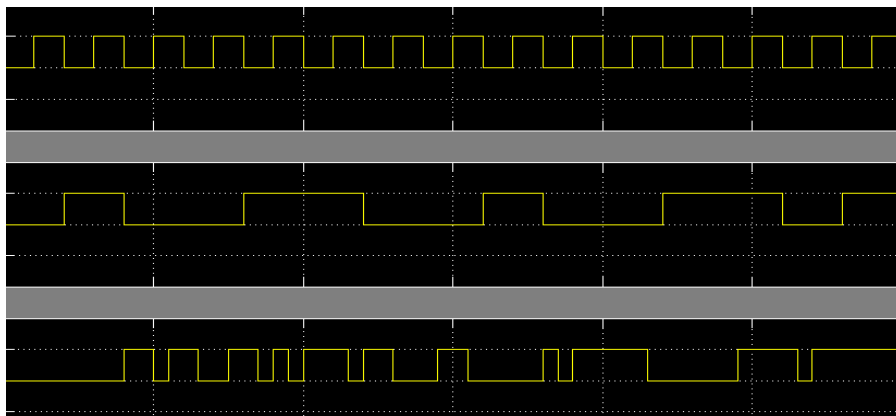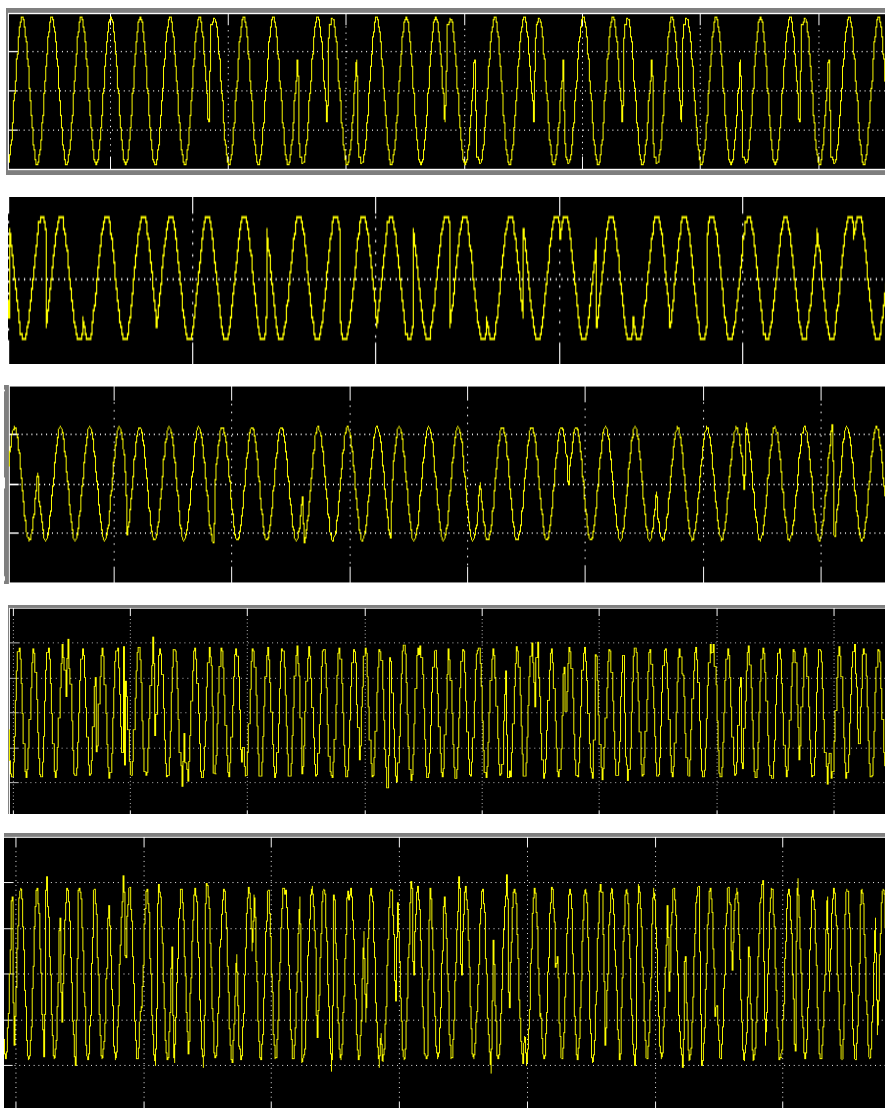

Fig. 11(a) Clock, Data and FEC

Fig. 11 (b) BPSK, QPSK, 8PSK, 16PSK, 32PSK modulator output waveforms

Table 5: Spartan 3AN FPGA device utilization summary

| Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	10,753	11,776	91%	
Number of 4 input LUTs	8,852	11,776	75%	
Number of occupied Slices	5,712	5,888	97%	
Number of Slices containing only related logic	5,712	5,712	100%	
Number of Slices containing unrelated logic	0	5,712	0%	
Total Number of 4 input LUTs	9,100	11,776	77%	
Number used as logic	8,808			
Number used as a route-thru	248			
Number used as Shift registers	44			
Number of bonded IOBs	14	372	3%	
IOB Flip Flops	1			
Number of BUFGMUXs	2	24	8%	
Number of RAMB16BWEs	6	20	30%	
Average Fanout of Non-Clock Nets	1.97			
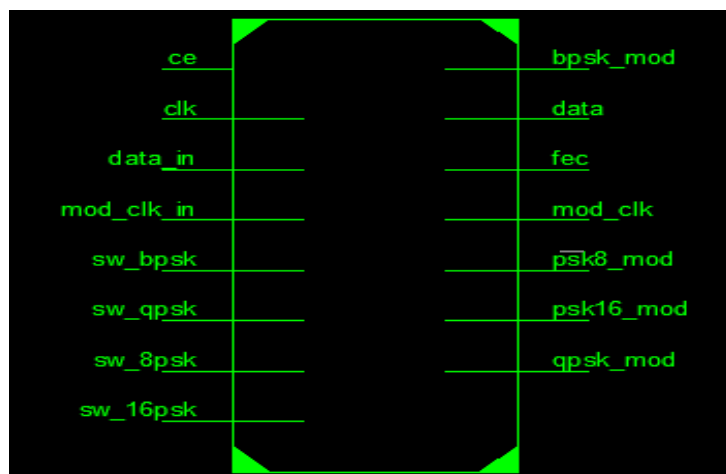
Fig. 12 Top level RTL schematic of BPSK, QPSK, 8PSK, 16PSK modulators
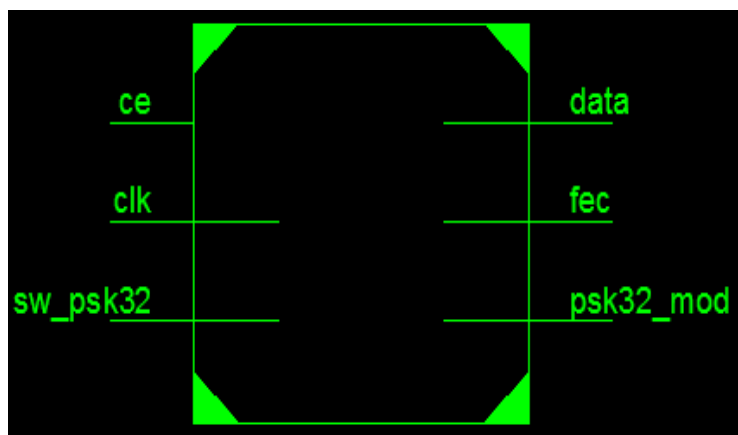

Fig.13 Top level RTL schematic of 32PSK modulator

Page | 84

(iii) Fig.14, 15 shows the VHDL test bench simulation indicating the phase changes of M-PSK modulators. The FPGA output of the M-PSK modulators measured on spectrum analyzer is as shown in Fig.16, 17, 18, 19, 20, 21. Typically all the switches were enabled to measure the output and with each switch being disabled sequentially / randomly, the corresponding modulator output is nil. Table 6 gives the summary of the measurements made and results. It can be seen that the bandwidth occupied by the signal depends on the modulation type (symbol rate), FEC and filter roll-off factor (α =0.5). For a particular data rate, as the modulation index increases the symbol rate decreases and hence the bandwidth efficiency increases. The sideband separation is also equals the symbol rate. Selection of the modulator output is also checked with the respective slide switch in ON/OFF condition. To test the versatility and flexibility of FPGA implementation, the design parameters of 32PSK modulator was modified to data rate of 500 KHz and carrier frequency of 4MHz. The modulator output reflects the modification as expected with a bandwidth of 300 KHz and sub-carrier spacing of 200 KHz as show in Fig. 19 and Table 4.
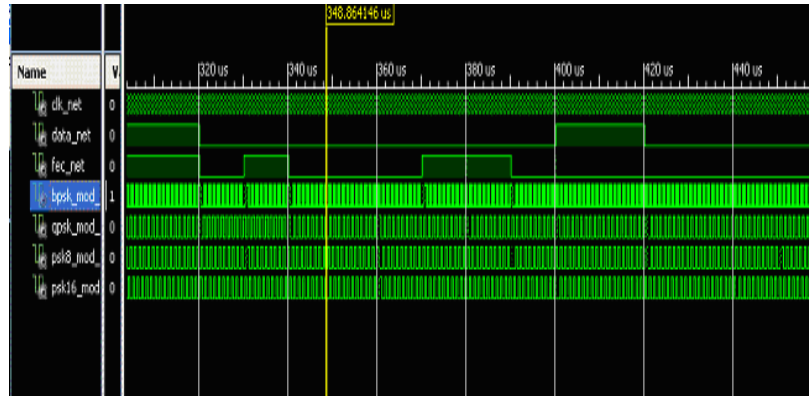

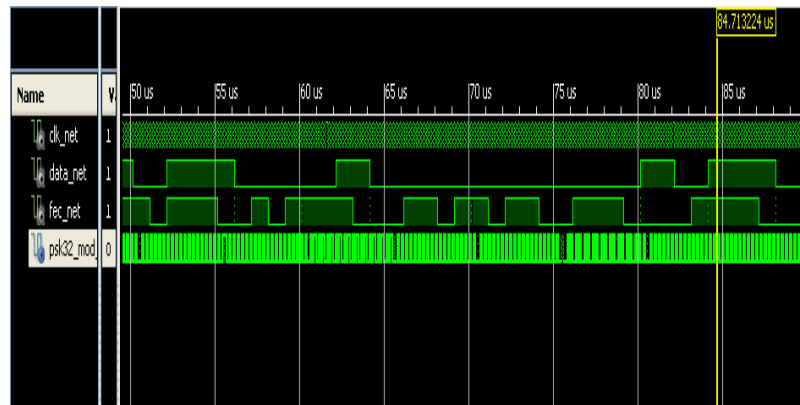**Fig.14 Xilinx simulation of BPSK, QPSk, 8PSK, 16PSK modulators**

Fig.15 Xilinx simulation of 32PSK modulator

Table 4 : Summary of the Results of M-PSK modulators

| Modulation Type	Data rate	Data rate with FEC (rate ½)	Symbol Rate $f_{sym}=R_b/m$	Calculated $B=f_{sym}(1+\alpha)$	Measured $B=f_{sym}(1+\alpha)$ at 3dB	Side band carrier separation
BPSK	50 KHz	100 KHz	100 KHz	150 KHz	153.3 KHz	100 KHz
QPSK	50 KHz	100 KHz	50 KHz	75 KHz	76.6 KHz	50 KHz
8PSK	50 KHz	100 KHz	33.3 KHz	50 KHz	50.6 KHz	33.3 KHz
16PSK	50 KHz	100 KHz	25 KHz	37.5 KHz	40 KHz	25 KHz
32PSK	50 KHz	100 KHz	20 KHz	30 KHz	30.6 KHz	20 KHz
32PSK	500 KHz	1MHz	200 KHz	300 KHz	300 KHz	200 KHz
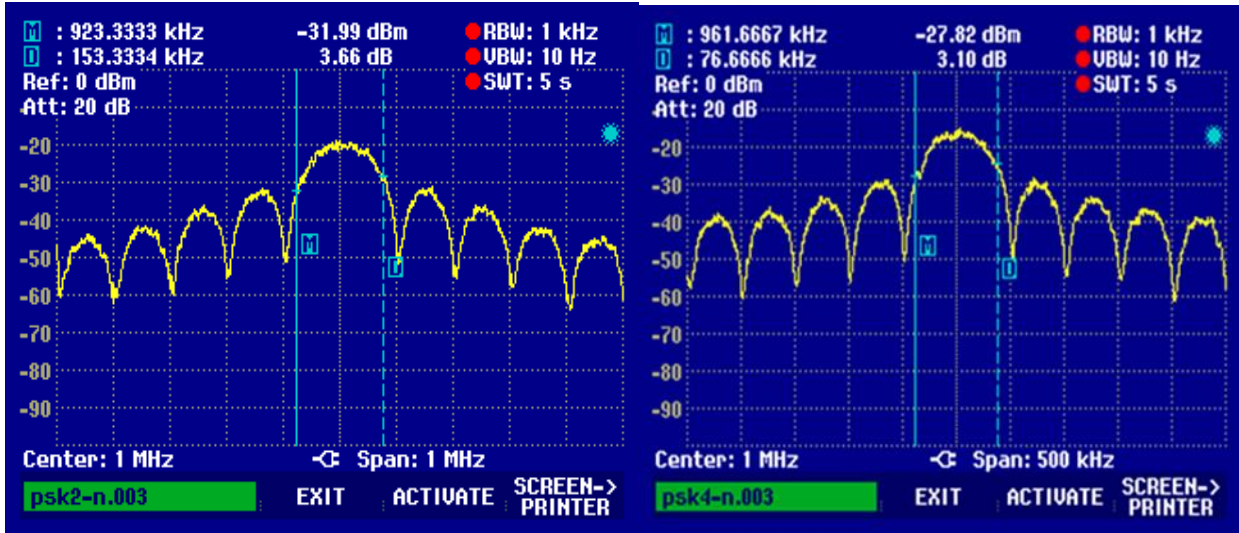
Fig.16 BPSK modulator output

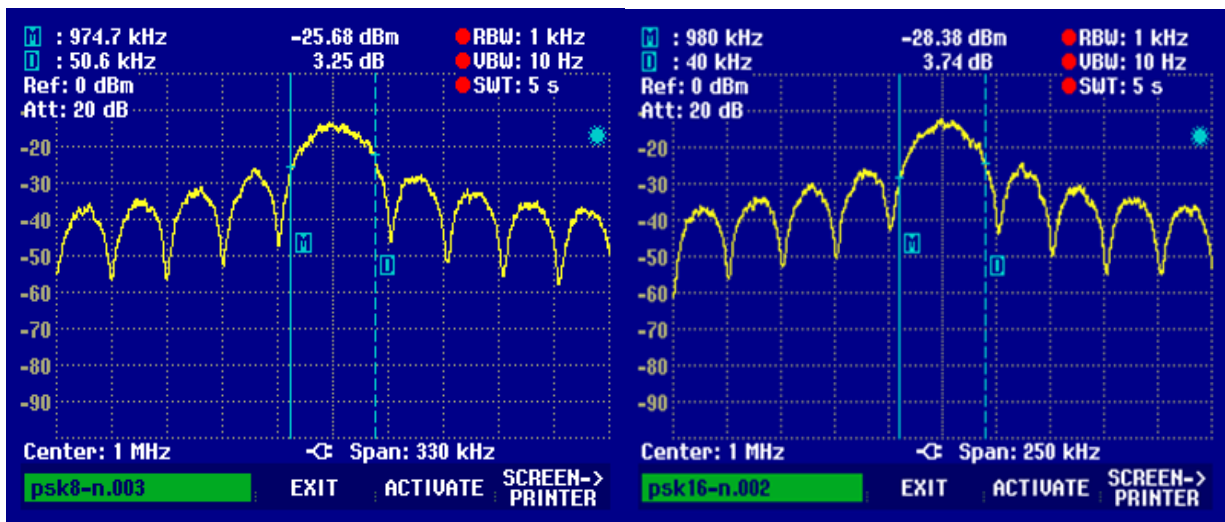**Fig.17 QPSK modulator output**

Fig.18 8PSK modulator output

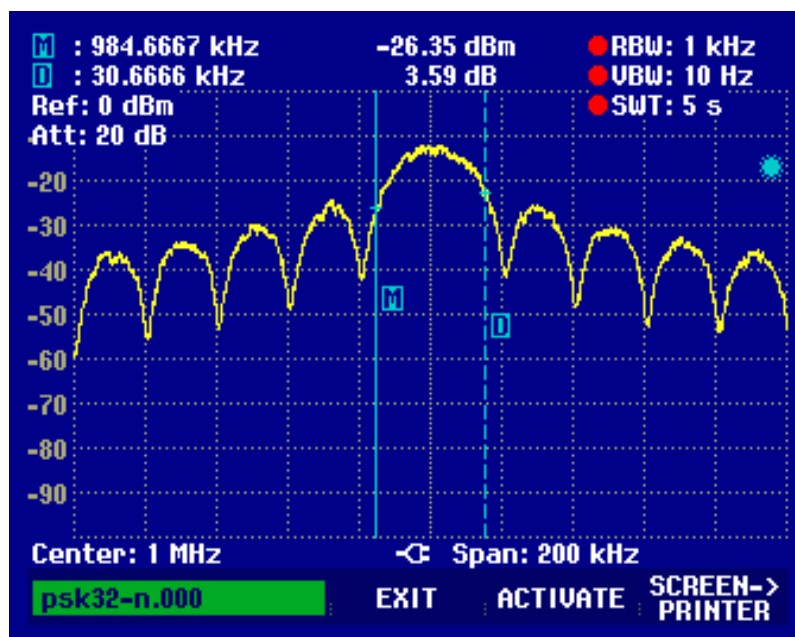**Fig.19 16PSK modulator output**
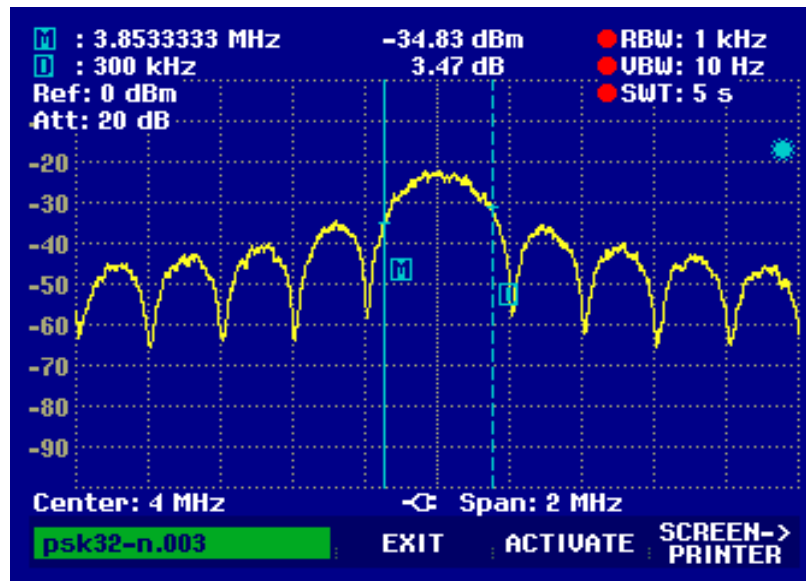

Fig.20 32PSK modulator output

Fig.21 Re-configured 32PSK modulator output

(iv) BPSK, QPSK modulators were tested with Industry made standard modem – PSM 500. The modulated ouput signal of 1MHz was up-converted to 70MHz using a mixer (MIQC88M) and given to PSM 500 demoulator section. The demodulator settings such as BPSK/QPSK modulation, 70MHz Carrier frequency, V.35 Descrambler, Differential decoder ,Viterbi decoder (rate=1/2) were enabled. The demodulator locked to the Carrier in both the cases, with a level of -13.2 dBm and E_b/N_0 of 16.2dB. This indicates that BPSK, QPSK modulators are compliant with commercial demodulators.

VI.   CONCLUSION

M-PSK modulators were successfully implemented on a single FPGA with selection and parameters like carrier frequency, symbol rate, signal bandwidth, side band carrier separation etc were measured and tabulated. FPGA platform enumerates the ease of implementation and provides flexibility in implementing the design with a change in design parameters. Also it provides a low cost, low power solution. Further work would involve design of M-PSK modulators with high data rates and M-APSK modulators for comparison of performance characteristics.

REFERENCES

[1]   Saavedra, C. E.; Vaughan, M.J., Compton R.C. "An MPSK modulator for Quasi –Optical Wireless Array Applications," IEEE MTT-S Digest, 1996.

[2]   Zhang, L.; Cao, Z., Gao, C. "Application of RS-coded MPSK Modulation scenarios to Compressed Image Communication in Mobile Fading Channel," IEEE Vehi. Tech. C 2000, pp. 1198- 1203.

[3]   Sharma, S. ; Kulkarn, S. ; Pujari, V. ; Vanitha, M. ; Lakshminarsimhan, P. "FPGA Implementation of M-PSK Modulators for Satellite Communication" Advances in Recent Technologies in Communication and Computing (ARTCom), 2010 , Page(s): 136 – 139

[4]   J. Hwang, B. Milne, N. Shirazi, J. Strommer, "System Level Tools for DSP in FPGAs," Xilinx Inc., 2001.

[5]   http://www.mathworks.co.uk/support/. (Matlab / Simulink sources).

[6]   Wenmiao Song, Qiongqiong Yao "Design and implement of QPSK modem based on FPGA" 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT),   (Volume: 9), 2010.