DESIGN AND IMPLEMENTATION OF MULTI-MODULATION AND DEMODULATION SYSTEM BASED ON SOFTWARE DEFINED RADIO (SDR) TECHNOLOGY [+]

Tahseen Flaih Hasan [*]

Abstract :

In this paper, the development of multi-modulation and demodulation baseband modulator and demodulator based on Software Defined Radio (SDR) are presented. The fundamentals of digital linear modulations are presented as algorithms to develop symbol of SDR signal. The multi-modulation baseband modulator (MMBM) and demodulator (MMBD) are designed using digital signal processing (DSP) algorithms, and then implemented into Xilinx Virtex-4 FPGA. Comparing the real-time and simulation results shows that the timings are equivalent, and the sign and magnitude changes are significant. Furthermore, at least 17% FPGA utilizations have been achieved for the configurable MMBM and MMBD.

تصميم وتنفيذ منظومة تحميل وفك التحميل المتعدد بتقنية النظام الراديوي المبرمج

تحسين فليح حسن

المستخلص:

يقدم هذا البحث تطوير جديد للتحميل وفك التحميل المتعدد بالاعتماد على تقنية النظام الراديوي المبرمج.. تفاصيل التحميل الرقمي الخطي تم توضيحه بشكل لوغارتمي لتطوير التحميل وفك التحميل للانظمة الراديوية المبرمجة باستخدام معالجات الاشارة الرقمية. نفذ التطوير المقترح في البورد نوع (FPGA) . تم مقارنة نتائج المحاكات والزمن الحقيقي وتبين ان النتائج متقاربة جدا مما يعزز الوثوق بالنظام.اثبتت النتائج حصول اختزال في عناصر بورد التنفيذ بحدود 17% بالمقارنة مع التصاميم الحالية.

Introduction:

Software radio (SR) is a set of digital signal processing (DSP) primitives, a meta level system for combining the DSP primitives into communications systems functions (transmitter, channel model, receiver) and a set of target processor on which SR is hosted for real-time communications [1]. An ideal SR receiver begins processing of incoming signal right after the antenna. Similarly, an ideal SR transmitter is located just before antenna. However, it is not practical due to limitations of bandwidth and dynamic range constrained by ADC and DAC. Therefore, a more practical version of SR called software defined radio

[+]**Received on 18/4/2012 , Accepted on 1/11/2012 .**
[*]**Assit Lech. / Technology, Institute of technology**

(SDR) has emerged that performs bit-stream, baseband, and IF processing with ADC and DAC as closed to RF conversion as possible [2].

The SDR is simply defined as "radio in which some or the entire physical layer functions are software defined" by the SDR [3]. This implies that the architecture is flexible such that SDR may be configured in real time to adapt itself to various wireless standards and waveforms, frequency bands, bandwidths, and modes of operation [4]. In commercial communications systems, the adoption of SDR concept in base-stations and mobile terminals is more prevalent, because the SDR features of multi-mode multi-standard functionality, programmability, and upgradeability offer an attractive solution to the problems of high cost, high power consumption, and large size suffered by the Velcro approach which consists of multiple radios employing multiple chipsets and platforms to support various applications [4]. Moreover, the flexibility of SDR also supports various quality-of-service requirements dealing with numerous data, voice and multimedia applications during mode switching between different air-interface standards [□].

Design Flow:

The flow of design and implementation can be divided into 2 parts namely software and hardware, as shown in Figure 1.The software part can be viewed as a composition of processes run by each software tool. Figure 2 illustrates this concept in a detailed development flow of design and implementation.

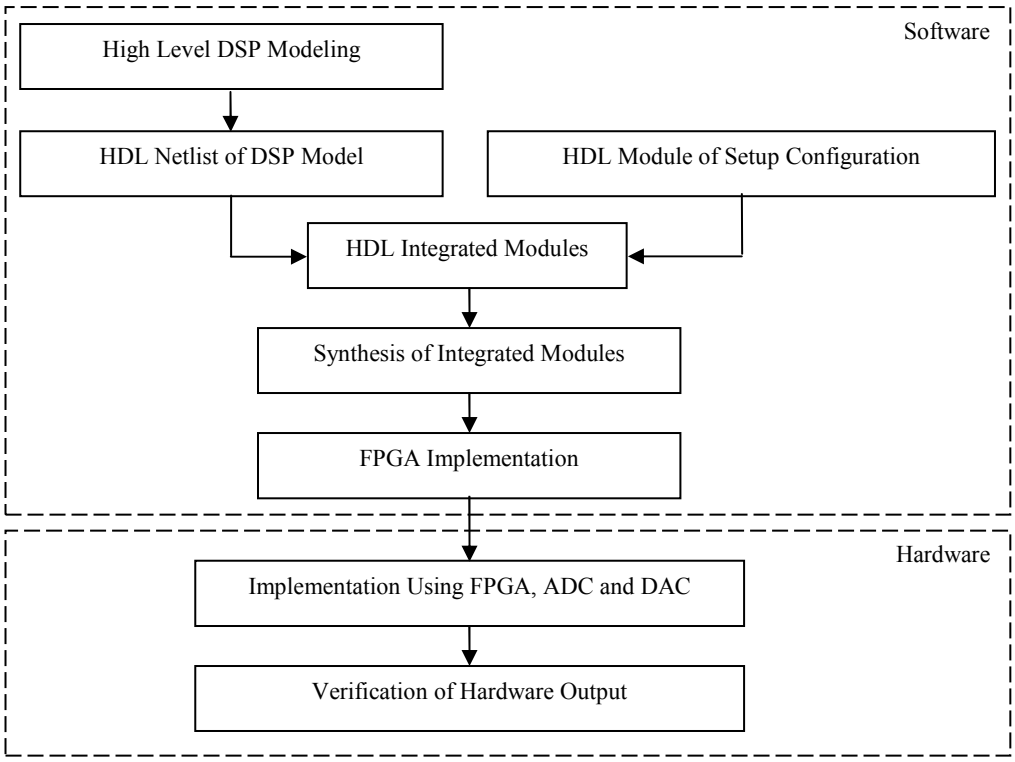

Figure 1: General Flow of Proposed Design system

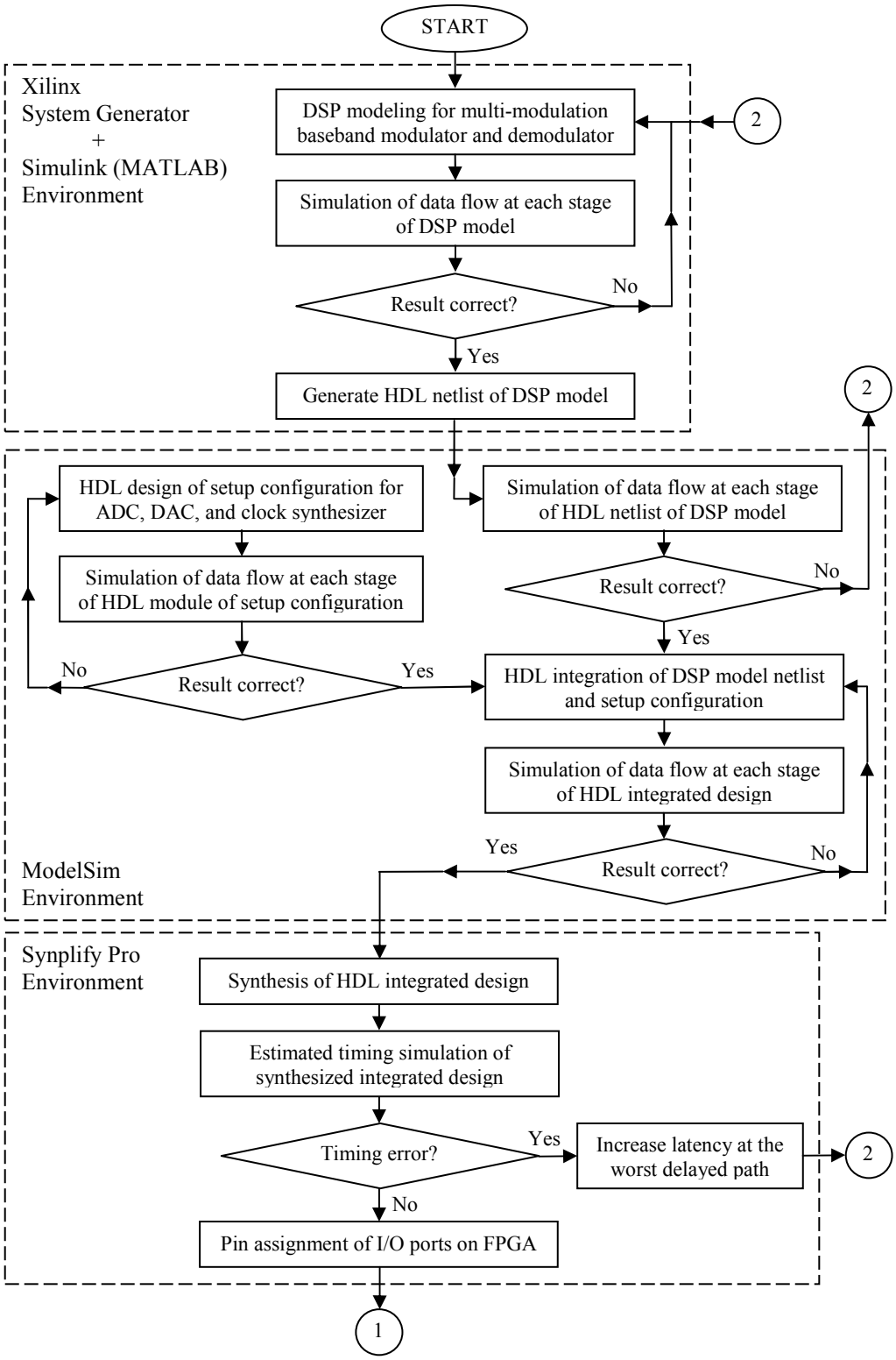

Figure 2: Development Flow of Design and Implementation.

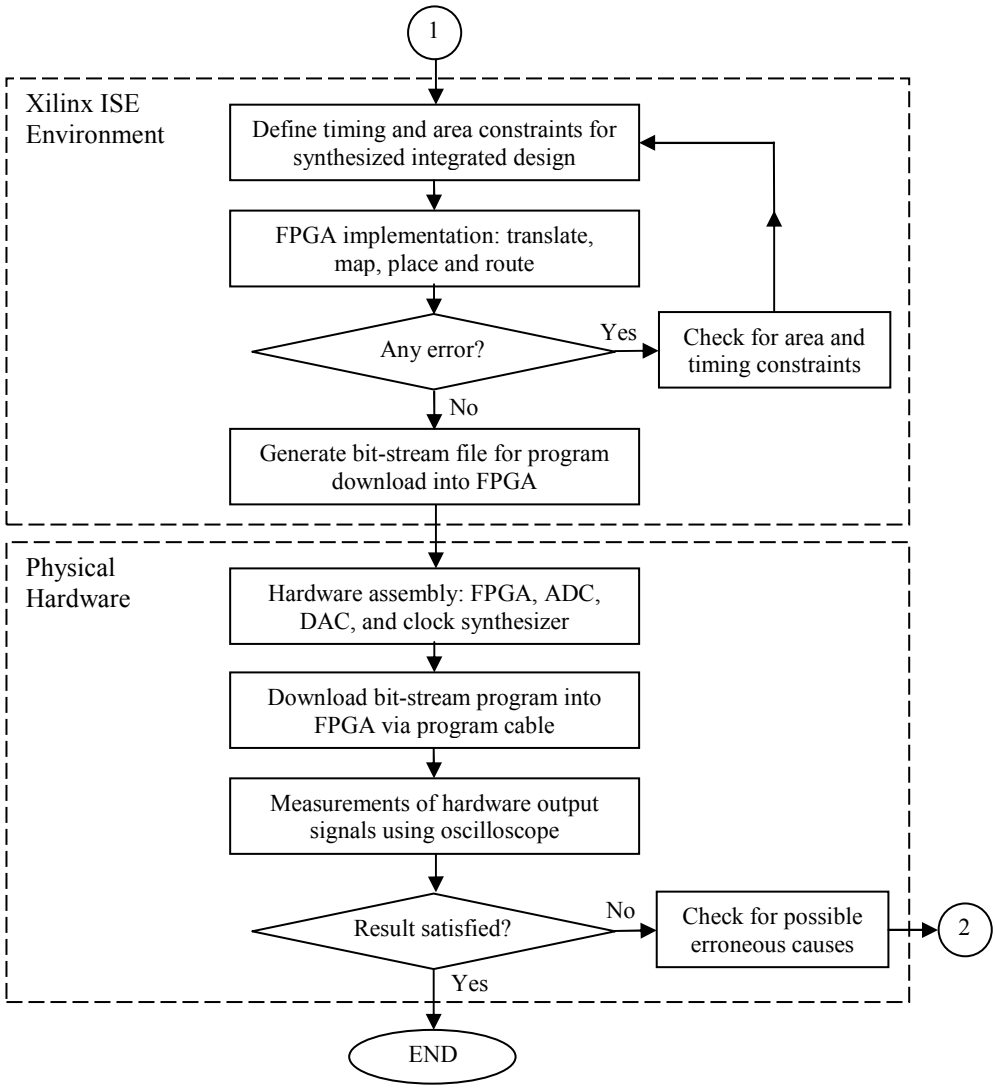

Figure 2: Continue

1- DSP Modeling

The first stage in the general flow of design and implementation is modeling the multi-modulation baseband modulator and demodulator (MMBMD) of SDR using digital signal processing (DSP) algorithms in high level graphical user interface (GUI). In this work, the high level GUI is Simulink environment contained in MATLAB software with add-in of Xilinx System Generator software. All DSP models contained in the MMBMD are built by connecting the blocks provided by libraries of Simulink Blockset and Xilinx Blockset in this GUI as shown in Figure 3.

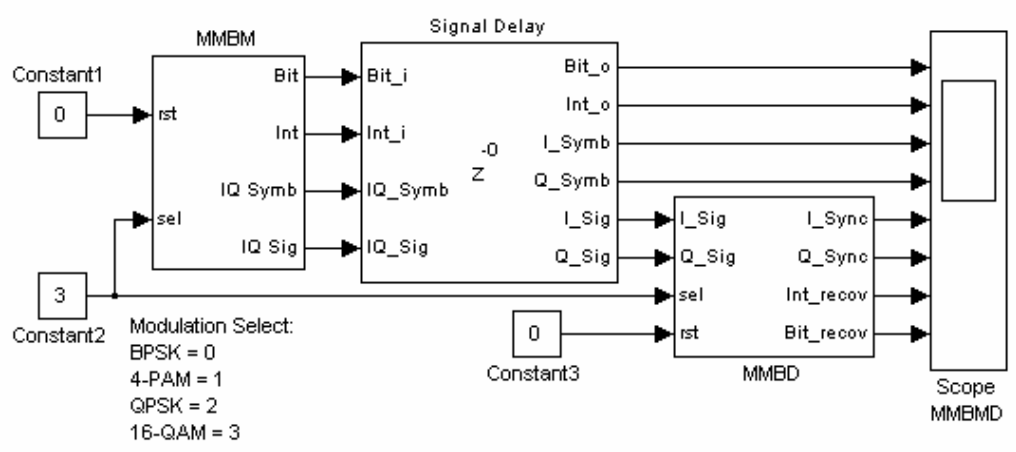

Figure 3: DSP design of MMBMD in System Generator

As illustrated in Figure 3. above, transmitter, receiver and noiseless (ideal) low-pass equivalent channel in the ideal radio communication system are represented by top level MMBM, top level MMBD and Signal Delay block respectively. The in-phase (I) and quadrature (Q) baseband modulated signals are delayed by the Signal Delay block before going to the MMBD as the effect of propagation through the low-pass equivalent channel. In addition the Signal Delay block also adds delays on input message bit, symbol integer and IQ symbols for better observation of comparisons between original and recovered data that will be displayed in Scope MMBMD block as shown in Figure 4

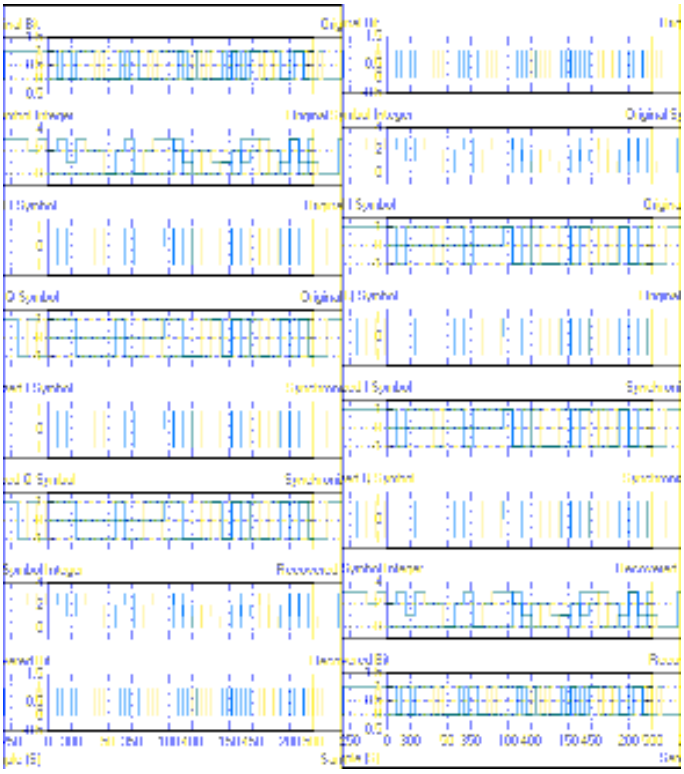

Figure 4 : Simulation output of MMBMD for QPSK

2-Multi-Modulation Baseband Modulator (MMBM)

The DSP model of top level MMBM in System generator environment is illustrated in Figure It comprises of main subsystems of Bit Generator, Symbol Mapper and Interpolation Filter. The input ports rst and sel are incoming from the Constant1 and 2 blocks respectively. The input/output signal of MMBM is shown in Figure 6.

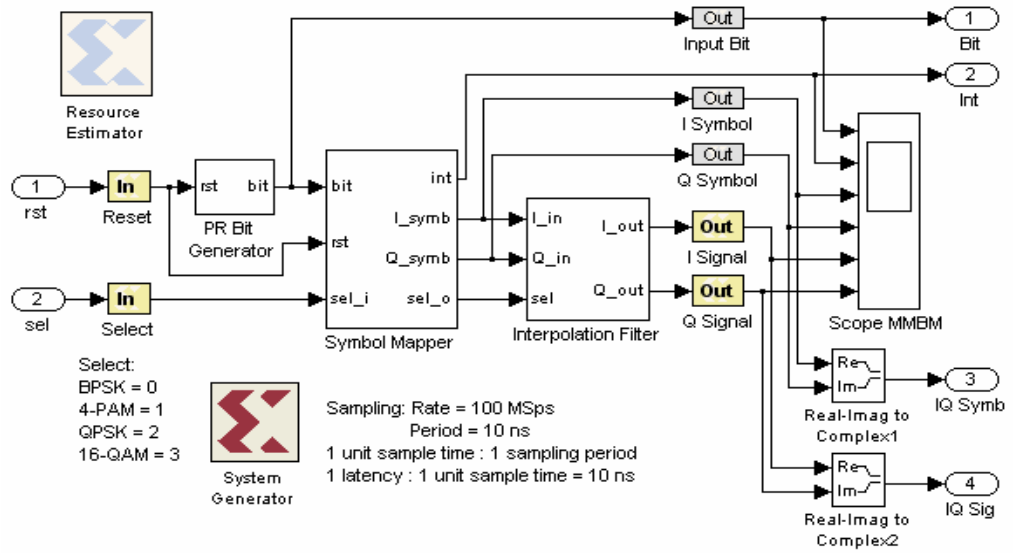

Figure 5: DSP design of MMBM

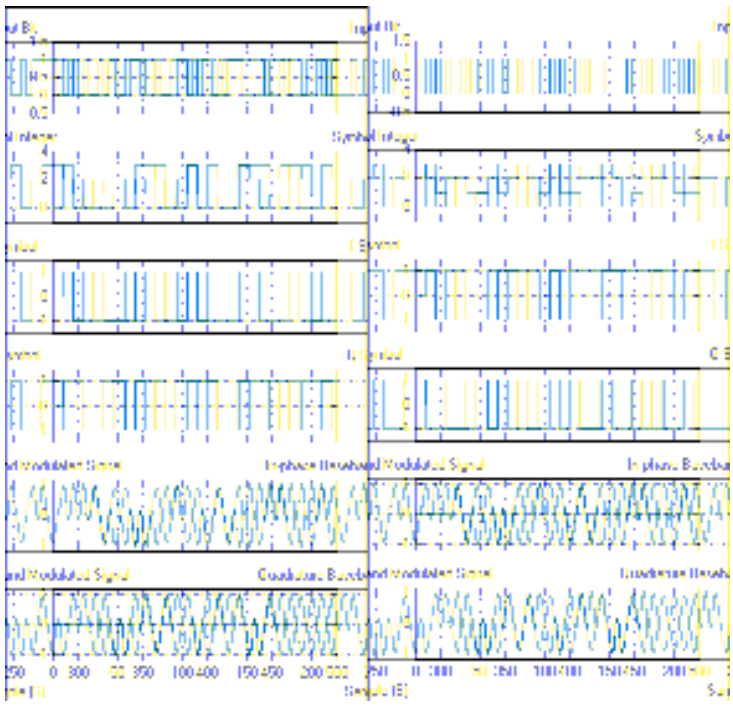

Figure 6: Simulation output of MMBM for QPSK

3-Multi-Modulation Baseband Demodulator (MMBD)

Figure 7 illustrates DSP model of top level MMBD in Sys Gen/Simulink environment. It comprises of main subsystems of RRC Filter, Timing Recovery, Symbol Demapper and Pulse Shaping. The input ports I_Sig, Q_Sig, sel and rst are incoming from the outputs I_Sig

and Q_Sig of the Signal Delay block, and Constant2 and 3 blocks respectively. The input/output signal of MMBD is illustrated in Figure 8 for QPSK scheme.

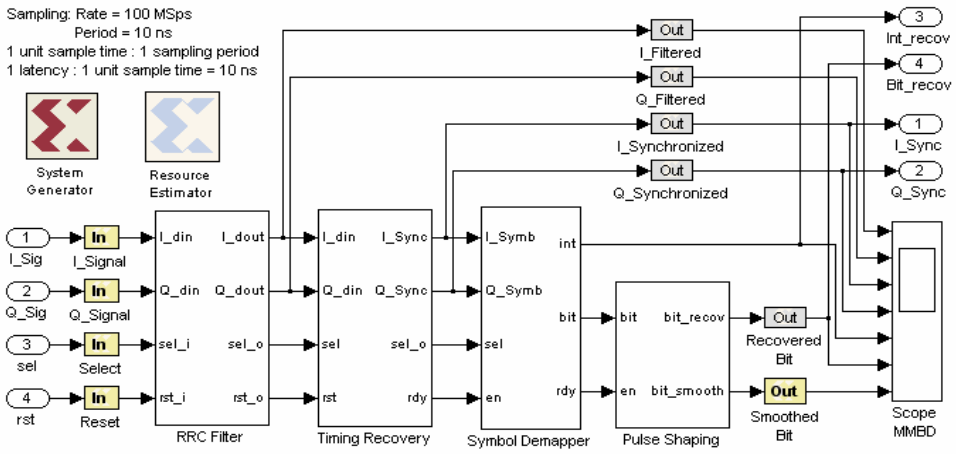

Figure 7: DSP Model of MMBD

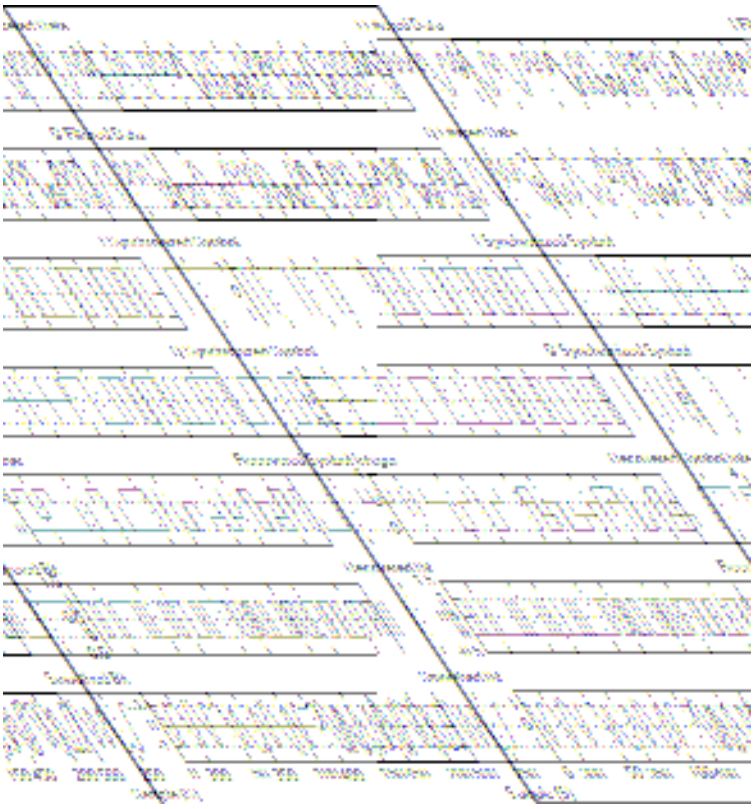

Figure 8: Simulation output of MMBD for QPSK

HDL Integrated Modules of DSP Models :

Considering real-time implementation of the DSP models of MMBM and MMBD integrated with setup configuration using FPGA, ADC and DAC in P240 should be

configured first prior to the running of DSP models of MMBM and MMBD. This can avoid instabilities of ADC and DAC that can produce undesired outputs during process of configuring ADC and DAC. Thus, the main clock enable (*ce*) of DSP model is disabled during the transfer of SPI codes to ADC and DAC in P240. Controlling the main *ce* would be easier rather than clock enable clear (*ce_clr*) which requires additional logics to adjust sampling phase $\varnothing_s$ of all the multi-sampled data.

Synthesis of HDL Integrated Modules:

Although Xilinx ISE has its own synthesis tool called XST (Xilinx Synthesis Technology), but it can only synthesize the HDL netlist generated from Sys Gen. Thus, Synplify Pro is used to perform logic synthesis for the HDL integrated modules of MMBM and MMBD (with setup configuration), in 2 stages [7]

a) *Logic compilation* and *optimization*:

The HDL integrated module is compiled to Xilinx FPGA structural elements and then optimized to be as small as possible to improve circuit performance.

b) *Technology mapping***:**

The optimized integrated module is mapped to Xilinx FPGA logic components using architectural-specific techniques.

FPGA Implementation :

The synthesis output files required by Xilinx ISE software are in EDIF and user constraints file (UCF) formats, representing the optimized netlist of synthesized integrated module, and timing constraints and FPGA pin assignment, respectively. To implement the synthesized integrated module of MMBM or MMBD into Xilinx Virtex-4 FPGA, Xilinx ISE [8], [9], [10] performs steps:

a) *Translate*:

The netlist file of synthesized integrated module in EDIF format is converted to native generic database (NGD) file containing logical descriptions of hierarchical components and Xilinx primitives for the synthesized integrated module, by using NGDBuild program.

b) *Map*:

The NGD file is going through logical design rule check (DRC) and then mapped to slices and I/O cells in Virtex-4 FPGA to create native circuit description (NCD) file. The area constraints can be sized properly using PACE (Pinout Area Constraints Editor); then re-run from *Translate* to *Map*.

c) *Place and Route* (*PAR*):

The mapped NCD file is placed and routed into Virtex-4 FPGA based on timing constraints using Timing Analysis tools, to form fully routed NCD file.

d) *Bit Generation* and *Program Download*:

The fully routed NCD file is converted to configuration bitstream file (.bit) using BitGen program, and then downloaded into Virtex-4 FPGA via JTAG cable using iMPACT program.

Before performing the last step of *Bit Generation* and *Program Download*, the results of *Translate*, *Map* and *PAR* processes should indicate zero (0) number of errors in order to ensure the validity of implementing the synthesized integrated module of MMBM or MMBD into the Virtex-4 FPGA.

Hardware Implementation and Verification Using FPGA, ADC and DAC:

The integrated modules of MMBM and MMBD (with setup configuration) are implemented separately using 2 sets of Xilinx Virtex-4 MB Board and P240 Analog Module as illustrated in Figure 9. After completely downloading program of the integrated modules of MMBM or MMBD from PC into Xilinx Virtex-4 FPGA through JTAG cable, the setup configuration module runs first [11], [12], [13].

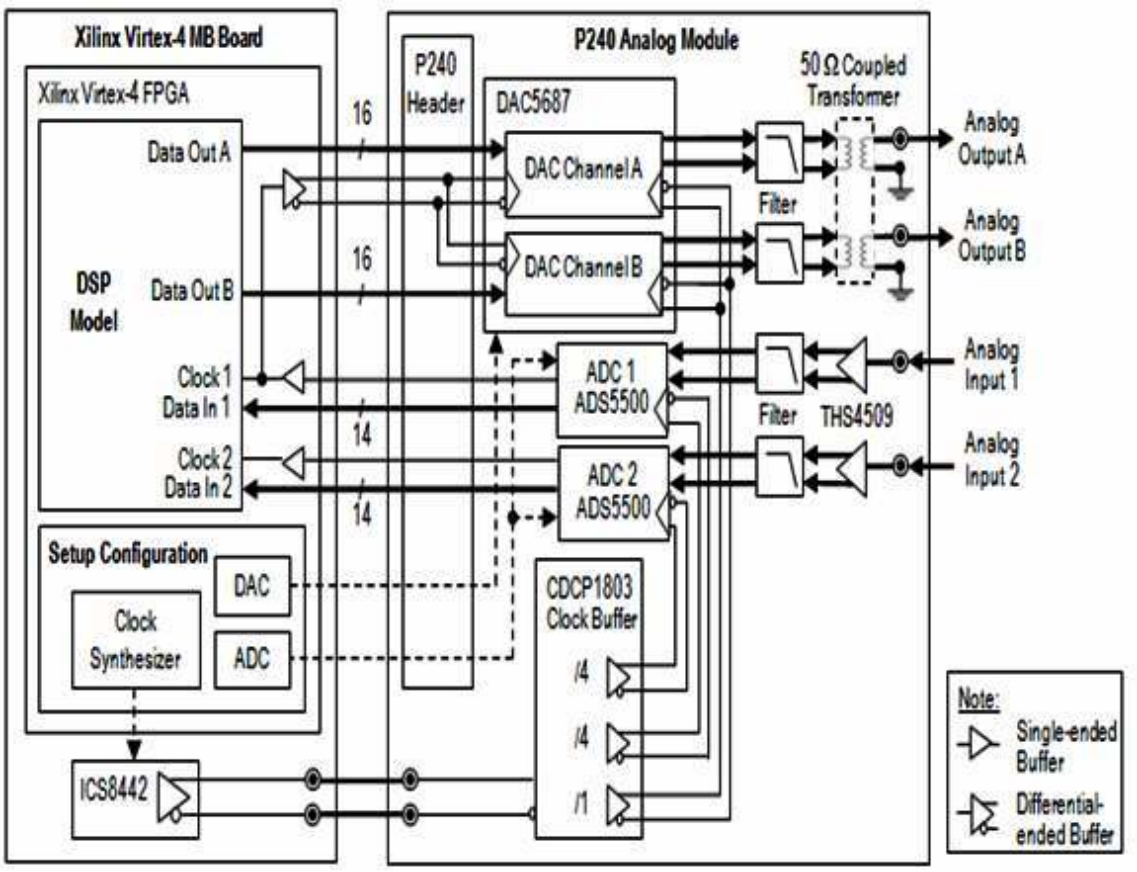

Figure 9: Hardware Implementation of FPGA and P240

After completing setup configuration module, the DSP models of MMBM or MMBD starts running with 100 MSps sample rate. By setting manually the input *Select* using FPGA on-board red DIP switch, the DSP models of MMBM or MMBD will perform baseband modulation or demodulation respectively, for either BPSK, 4-PAM, QPSK, or 16-QAM.

The analog output signals are connected to oscilloscope to display the real-time results for verifying system performances of the integrated modules of MMBM and MMBD through testing and measurements, as illustrated in Figure 10. To capture initial real-time results of configurable MMBM, the input Reset of DSP Model of MMBM is activated by pressing push button on the first Virtex-4 MB board. Whereas the input *Reset* of DSP Model of MMBD is activated first and hold until the input *Reset* of DSP Model of MMBM is finished activating, by pressing push buttons on the second and first Virtex-4 MB Board respectively, to capture initial real-time results of configurable MMBD.

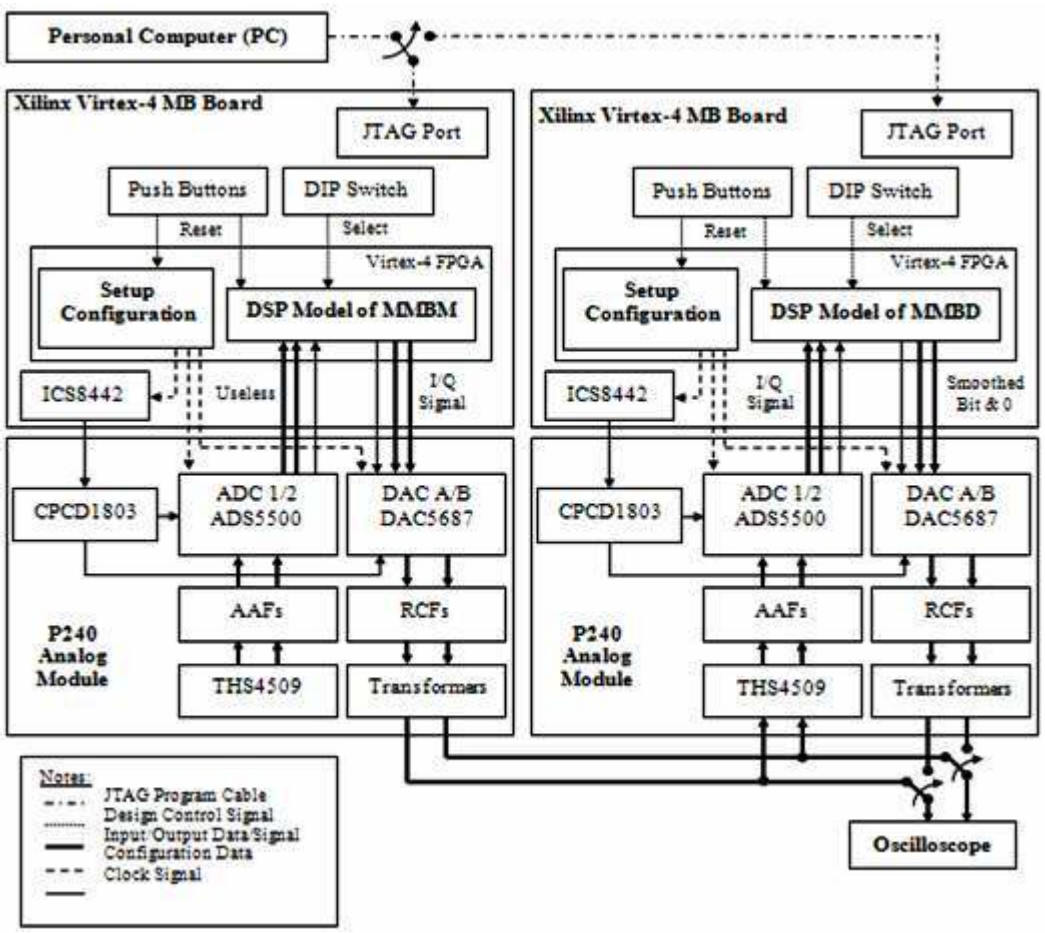

Figure 10: Hardware Assembly for Testing and Measurement

The real-time results of configurable MMBM and MMBD are shown and compared with the respective simulation results to verify functionalities of the configurable MMBM and MMBD implemented using FPGA as show in Figure 11. Finally, the efficiencies of configurable MMBM and MMBD are evaluated in terms of FPGA elements, as compared to the single- and grouped-modulation baseband modulator (BM) and demodulator (BD) respectively.

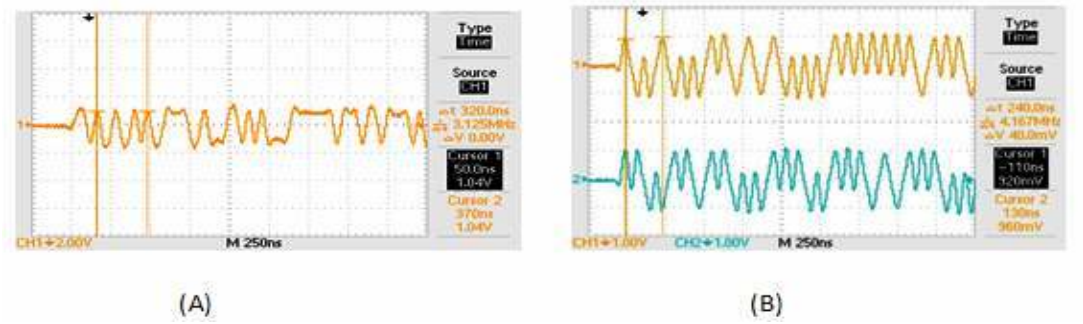

Figure 11: Real-time Results of (A): MMBD and (B): MMBM for QPSK

1-Synthesis Results in Synplify Pro Environment

Timing characteristics is an important issue that will affect the performance of FPGA implementation. The required path delay (estimated period) for Xilinx FPGA elements should

be less than the requested (constrained) clock period. Thus, timing slack (= requested period − estimated period) should be positive value; otherwise the integrated modules of MMBM and MMBD have to be reworked. The clock signals used for the ADC/DAC SPI process and DSP model of MMBM or MMBD are namely CLK_100 and LIO_CLKIN_1 respectively, and both are constrained to 100 MHz. From the estimated timing report of synthesis as shown in Table 1, the positive slack values meet the timing requirements for both of the integrated modules of MMBM and MMBD.

Table 1: Estimated Timing Report of Synthesis

Integrated Module	Clock Constraint	Requested		Estimated		Slack (ns)
		Frequency (MHz)	Period (ns)	Frequency (MHz)	Period (ns)	
MMBM	CLK_100	100.0	10.000	189.□	□276	4.724
	LIO_CLKIN_1	100.0	10.000	222.7	4.490	□□10
MMBD	CLK_100	100.0	10.000	189.4	□280	4.720
	LIO_CLKIN_1	100.0	10.000	110.4	9.0□4	0.946

2-Reports Generated in Xilinx ISE Environment

Besides ensuring no error resulted from the *Translate*, *Map* and *PAR* processes run in Xilinx ISE environment, the performances of FPGA implementations of the integrated modules of MMBM and MMBD have to be analyzed in terms of device utilization, timing constraints and power consumption.

1. FPGA Device Utilization

After the *Map* process, the accurate FPGA utilizations of the integrated modules of MMBM and MMBD were summarized as shown in Tables 2 and 3 respectively. Note that each FPGA slice consists of 2 sets of 4-input LUTs and D-FFs. The related logic refers to packing the slice logic (LUT or D-FF) that shares common inputs into a single CLB for best timing performance, where each CLB contains 4 (or 2 pairs of) interconnected slices.

Table 2: Device Utilization Summary of MMBM Module

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	505	30,720	1%	
Number of 4 input LUTs	335	30,720	1%	
Logic Distribution				
Number of occupied Slices	332	15,360	2%	
Number of Slices containing only related logic	314	332	94%	
Number of Slices containing unrelated logic	18	332	5%	
Total Number of 4 input LUTs	433	30,720	1%	
Number used as logic	335			
Number used as Shift registers	98			
Number of bonded IOBs	63	448	14%	
Number of BUFG/BUFGCTRLs	4	32	12%	
Number used as BUFGs	4			
Number used as BUFGCTRLs	0			
Number of FIFO16/RAMB16s	4	192	2%	
Number used as FIFO16s	0			
Number used as RAMB16s	4			
Number of DSP48s	4	192	2%	
Total equivalent gate count for design	275,160			
Additional JTAG gate count for IOBs	3,024			

Table 3: Device Utilization Summary MMBD Module

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	2,076	30,720	6%	
Number of 4 input LUTs	2,286	30,720	7%	
Logic Distribution				
Number of occupied Slices	2,068	15,360	13%	
Number of Slices containing only related logic	2,068	2,068	100%	
Number of Slices containing unrelated logic	0	2,068	0%	
Total Number of 4 input LUTs	2,938	30,720	9%	
Number used as logic	2,286			
Number used as a route-thru	5			
Number used as Shift registers	647			
Number of bonded IOBs	91	448	20%	
Number of BUFG/BUFGCTRLs	4	32	12%	
Number used as BUFGs	4			
Number used as BUFGCTRLs	0			
Number of FIFO16/RAMB16s	7	192	3%	
Number used as FIFO16s	0			
Number used as RAMB16s	7			
Number of DSP48s	40	192	20%	
Total equivalent gate count for design	537,987			
Additional JTAG gate count for IOBs	4,368			

Table 4: FPGA Utilization

FPGA Element	Existing Results			Proposed Results	Difference: (2011) vs.		
	[14]	[15]	[16]		2006	2008	2009
Slices	444	243	□09	366	−78	+123	−143
FFs	739	208	24□	281	−4□8	+73	+36
LUTs	339	400	821	□06	+167	+106	−31□
BRAMs	4	0	0	3	−1	+3	+3
EMs	6	3	1	2	−4	−1	+1

The FPGA utilization of the mapped partitions of setup configuration module and DSP model, and the mapped integrated module for configurable MMBM is listed in Tables 4, along with the estimated FPGA elements of the respective DSP model. For the mapped setup configuration module, the slices and LUTs for the configurable MMBM is lower than that for the configurable MMBD due to the exclusion of a pair of 14-bit data inputs in the MMBM; however the FFs for the former is higher because 18 slices were mapped to be containing unrelated logic. For DSP models of MMBM and MMBD, the RAM16s, DSP48s, BUFGs and IOBs are equal for both of the mapped and estimated cases; however the slices and LUTs for the mapped case have been reduced by nearly 40 % from the estimated case. The FFs for the mapped DSP model of MMBM has been increased by about 6 % from the estimated case due to the mapping effect of 18 slices containing unrelated logic in the setup configuration module; however the FFs for the mapped DSP model of MMBD has been reduced by about 29 % from the estimated case because all slices were mapped to be containing related logic. Note that 4 BUFGs are used to remove jitters of the clock signals connecting to the ADC and DAC SPI processes, the ADC/DAC data bypass, and the DSP model. The RAMB16s and DSP48s are FPGA elements used to implement BRAMs and EMs respectively.

2.Power Estimation Report

The Virtex-4 FPGA power consumptions of the integrated modules of MMBM and MMBD were estimated by using Analyze Power (XPower) program under the *PAR* process, as listed in Table 6. The computation of power estimation is divided into 3 parts i.e. core design of the integrated modules of MMBM or MMBD, auxiliary (JTAG and clock circuitry), and output bank, with standard voltages V_{CCINT} = 1.2 V, V_{CCAUX} = 2.□V, and V_{CCO} = 2.□V respectively. The total currents for the auxiliary and output bank parts are equal for both of the integrated modules of MMBM and MMBD, thus resulting in equal powers of 407.10 mW and 2□6□ mW respectively. However, the total current and power for core design of the integrated MMBD module are 119.13 mA and 142.9□6 mW more than that of the integrated MMBM module respectively, because the integrated MMBD module utilized more FPGA resources than the integrated MMBM module. Which mean that, the receiver path always consume more power due to use additional circuit like timing recovery?

Table 6: Estimated Power Report for Integrated Modules

Part	Voltage (V)	Current (mA)		Power (mW)		
		MMBM	MMBD	MMBM	MMBD	Difference
Core design	1.2	204.2□	323.38	24□100	388.0□6	142.9□6
Auxiliary	2.□	162.84	162.84	407.100	407.100	0.000
Output bank	2.□	10.26	10.26	2□6□0	2□6□0	0.000
Total	-	-	-	677.8□0	820.806	142.9□6

Conclusion:

The multi-modulation baseband modulator (MMBM) and demodulator (MMBD) of software defined ratio (SDR) have been developed A new technique has been used to reduce hardware size (FPGA area) as illustrated in Table 4 for implementing the configurable MMBM and MMBD, by integrating the common features shared by two or more single-modulation structures of SDR baseband modem functions. The new technique is incorporated with superposition principle of digital linear modulation, polyphase interpolator architecture with optimization, and switching network.

As compared to group-modulation baseband modulator (BM) and demodulator (BD) which are modified from the recent researches of SDR modems, the unmapped FPGA utilizations of configurable-modulation BM and BD have been reduced by □0.0% and 31.7% in slices, □0.9% and □2.□% in FFs, □0.9% and 17.3% in LUTs, 66.7% and 70.0% in BRAMs, and 66.7% and □0.0% in EMs, respectively. Therefore, validity of the proposed technique is confirmed. Besides, after implementation of the integrated modules of MMBM and MMBD, the mapped FPGA utilizations are lower than the unmapped FPGA utilizations of DSP models of MMBM and MMBD respectively in terms of slices, FFs and LUTs. Therefore, the proposed configurable MMBM and MMBD are really efficient.

References:

1- Mitola, J, III. Software Radios: Survey, Critical Evaluation and Future Directions. *IEEE Aerospace and Electronic Systems Magazine*, 8(4), pp. 2□-36, 1993

2-Mitola, J. The Software Radio Architecture. *IEEE Communications Magazine*, 33(□), pp. 26-38, 199□

3-Software Defined Radio Forum, Inc. Introduction to SDR: "What is SDR?" [Online]. [Accessed 8th September 2009]. Available from World Wide Web: http://www.sdrforum.org/pages/aboutSdrTech/whatIsSdr.asp

4-Rouphael, T. J RF and Digital Signal Processing for Software-Defined Radio: A Multi-Standard Multi-Mode Approach. MA: Newness, Elsevier, 2009

□-Aghvami, A. H., Le, T. H., and Olaziregi, N. Mode Switching and QoS Issues in Software Radio. *IEEE Personal Communications*, 8(□), pp. 38-44, 2001

6-Xilinx, Inc. *System Generator for DSP User Guide*, Release 9.2.01, 2007

7-Xilinx, Inc. Xilinx ISE 9.2i Software Manuals: Constraints Guide, and Development System Reference Guide, 2007

8-Synplicity, Inc. Synplicity FPGA Synthesis Reference Manual, 2007

9-Avnet, Inc.P240 Analog Module User Guide, Rev. 1.0 [Online]. [Access 17th March 2009]. Available from World Wide Web: http://www.files.em.avnet.com/files/177/p240_analog-ug.pdf

10-Avnet Memec, Inc. Virtex-4 MB Development Board User's Guide, Ver. 3.0 [Online]. [Access 17th March 2009]. Available from World Wide Web:http://www.files.em.avnet.com/files/177/v4mb_user_guide_3_0.pdf

11-Xilinx, Inc. *XtremeDSP for Virtex-4 FPGAs User Guide*, version 2.7 [Online]. [Access 20th May 2009]. Available from World Wide Web: http://www.xilinx.com/support/documentation/user_guides/ug073.pdf

12-Xilinx, Inc. *Virtex-4 FPGA User Guide*, version 2.6 [Online]. [Access 18th May 2009]. Available from World Wide Web: http://www.xilinx.com/support/documentation/user_guides/ug070.pdf

13-Xilinx, Inc. *Virtex-4 FPGA Data Sheet: DC and Switching Characteristics*, version 3.7 [Online]. [Access □th June 2010]. Available from World Wide Web: http://www.xilinx.com/support/documentation/data_sheets/ds302.pdf

14- Dick, C., Egg, B., and Harris, F. Architecture and Simulation of Timing Synchronization Circuits for the FPGA Implementation of Narrowband Waveforms, in the *Proceedings of Software Defined Radio Technical Conference and Product Exposition*, 13-16 November 2006, Orlando (USA), pp. 1-6, 2006

1□- Zicari, P., Sciagura, E., Perri, S., and Corsonello, P. A Programmable Carrier Phase Independent Symbol Timing Recovery Circuit for QPSK/OQPSK Signals. *Microprocessors and Microsystems*, 32(8), pp. 437-446, 2008

16- Tachwali, Y., Barnes, W. J., and Refai, H. Configurable symbol synchronizers for software-defined radio applications. *Journal of Network and Computer Applications*, 32(3), pp. 607-61□, 2009