Development and Study of One-channel Programmable Pulse Synthesizer

Valentina Rankovska[1] and Hristo Karailiev[2]

Abstract – **One-channel programmable pulse synthesizer, operating in two modes – continuous and counter mode, with and without PWM, has been designed. The design is implemented using Direct Digital Synthesis and Field-Programmable Gate Arrays. High resolution of PWM and wide range of duty cycle has been achieved.**

Keywords – **Numerically Controlled Oscillator**, **Field-Programmable Gate Arrays, Programmable Pulse Synthesizer, Pulse-Width Modulation.**

I. INTRODUCTION

Synthesizing square wave signals is widely applied – to control low power single switch invertors in real time in order to keep their optimal power [4]; in testing and calibration of microprocessor devices measuring velocity, frequency, time intervals, etc. in automobile electronics, measurements and communications, etc.; in developing building blocks of precise functional generators; to control and regulate the revolutions of DC motors, etc.

Three groups of methods for synthesizing square wave signals are known – analogue, digital, and hybrid, but during the last decade the Direct Digital Synthesis (DDS) is more and more studied and applied [1], [5], [7]. It has many advantages as: very high resolution, wide frequency range, continuous phase of the output signal, good spectral purity, etc. At the same time implementing the DDS method together with novel circuits as Field-Programmable Gate Arrays (FPGA) [3] leads to considerable possibilities as: synthesizing hardware by software means, integrating various functions of a design in one chip, implementing complex algorithms for parallel operation; increased operation speed, flexibility and reliability, low power and cost, etc.

Aim of the paper: Development and study of one-channel programmable pulse synthesizer.

Main problems:

§ Defining the features of the synthesizer: two modes – continuous pulses and definite number of pulses (counter mode), each of them with two sub-modes: with duty cycle 50% and with programmable duty cycle from 1 to 99 %;

§ Implementation of the synthesizer using FPGA;

§ Development and simulation of the project.

[1]Valentina Rankovska is with the Faculty of Electrical Engineering and Electronics at Technical University of Gabrovo, 4 H. Dimitar str., Gabrovo 5300, Bulgaria, E-mail: rankovska@tugab.bg.

[2]Hristo Karailiev is with the Faculty of Electrical Engineering and Electronics at Technical University of Gabrovo, 4 H. Dimitar str., Gabrovo 5300, Bulgaria, E-mail: hkarailiev@gmail.com.

II. ARCHITECTURE OF ONE-CHANNEL PROGRAMMABLE PULSE SYNTHESIZER

The pulse synthesizer is designed for Altera's FPGA Cyclone EP1C6Q240C8 [2]. Integrated software environment Quartus II 8.1 Web Edition [6] has been used for its developing and simulations. The design has been created with the Quartus II Block Editor using the library functions of the software.

The architecture of the synthesizer is shown in Fig. 1 and it consists of the following basic blocks:

§ **Input data block**, based mainly on the shift registers *lpm_shiftregx*. It transfers to the next blocks the input data necessary for the proper operation of the synthesizer, as: output frequency, output duty cycle and the number of output pulses.

Entering the data can be made using two types of interfaces: serial and parallel, depending on the necessary speed and hardware. Serial interface has been used at the current design, decreasing the used pins of FPGA.

The input data come from a control unit, for instance a microcontroller. The purpose of the input pins of the block is as follows: *data_clk3* – clock signal, coming from the control unit; *reg_cnt* – control signal, setting input of shift register address or input of data into the selected shift register; *data_in* – input for address or data.

§ **Numerically Controlled Oscillator (NCO) block**, defining the output frequency, based on the adder *lpm_add_sub2* and the register *lpm_dff0*. As NCO block is a part of a direct digital frequency synthesizer, its operation will not be discussed here. The input *clk* is for the clock frequency. Its operation is controlled by the input signal *mng[2]*. The output signal *qout* is used for test purposes.

§ **Pulse-width Modulation (PWM) block,** realized on the base of the digital comparator *lpm_compare5* and the multiplexer *lpm_mux12*. Choosing a sub-mode – with 50% duty cycle and with programmable duty cycle from 1% to 99 % is made by the signal *mng[0]*.

§ **Counter block**, used when the synthesizer works in counter mode (generating a definite number of pulses). It consists of the counter *lpm_counter9* and the comparator *lpm_compare3*. Two signals control its work: *pls_cnt_clr* for clearing the counter and *pls_cnt_en5*, enabling/disabling its operation.

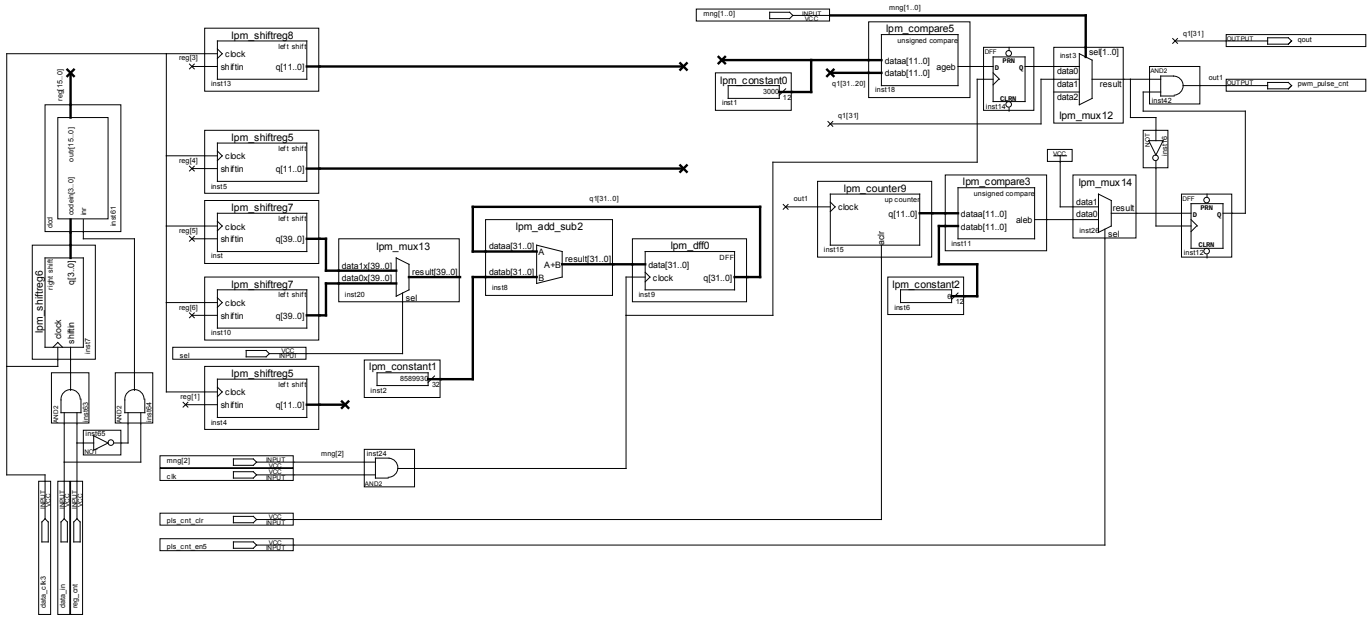

Fig. 1. Architecture of one-channel programmable pulse synthesizer

Table I shows the input constants and the corresponding blocks used in the various modes of operation of the synthesizer.

TABLE I
CONSTANTS AND CORRESPONDING BLOCKS USED IN SIMULATION AND OPERATION OF THE DESIGN

Constant	*Block used at simulation*	*Block used at programming*	*Purpose*
k_0	*lpm_constant0*	*lpm_shiftreg5 (inst5)*	Defines the duty cycle for PWM sub-modes
k_1	*lpm_constant1*	*lpm_mux13 (inst20)*	Defines the output frequency f_{NCOout}
k_2	*lpm_constant2*	*lpm_shiftreg8 (inst13)*	Defines the number of pulses in counter mode (the value must be less by 1 from the necessary number)

The output signal is *pwm_pulse_cnt.* Its frequency/ period are defined by the Eqs. (1) and (2) [1]:

$$f_{out} = f_{NCOout} = \frac{k_1 . f_{clk}}{2^n} \quad (1)$$

$$T_{out} = \frac{2^n}{k_1 . f_{clk}}, \quad (2)$$

where k_1 is the frequency control word, f_{clk} – clock frequency for NCO, n – width of the phase accumulator of NCO.

The PWM resolution is expressed by the Eq. (3):

$$\Delta\tau_I = \frac{T_{out}}{2^p}, \quad (3)$$

where p is the width of the digital comparator *lpm_compare5.* τ_I is the pulse duration, defined by the Eq. (4):

$$\tau_I = k_0 \frac{T_{out}}{2^p} \quad (4)$$

III. SIMULATION RESULTS FROM THE OPERATION OF THE ONE-CHANNEL PROGRAMMABLE PULSE SYNTHESIZER

To simulate the design it is necessary preliminary to create a file consisting of the input and output signals and the input stimuli. It is created as a table in Quartus II and includes the control signals for the four sub-modes, shown in Table II, and the two monitored output signals - *pwm_pulse_cnt* and *qout.*

The initial data for the simulations are the following: f_{clk}*=50 MHz,* f_{out}*=100 kHz/* T_{out}*=10* μs , PWM resolution – *2, 4414 ns.* Some results of the simulations are shown in Table III, where τ_{Icalc} is calculated τ_I ,τ_{Isim} is τ_I from the simulation and DC_{sim} is the duty cycle from the simulation. Figs. 2, 3, 4 and 5 show waveforms illustrating the operation of the synthesizer in the various modes – continuous and counter, with and without PWM, and for various values of the constant k_0, setting the duty cycle. Only 56 logic elements from about 6000 and 12 pins of FPGA has been used, which allows building in the design into more complex devices and systems with large number of features.

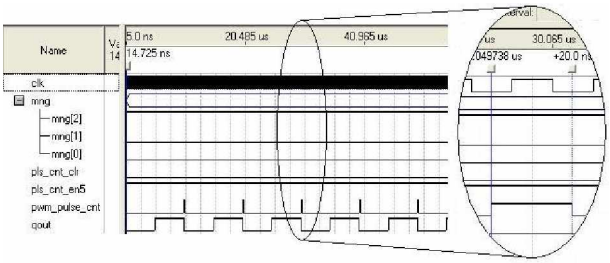

Fig. 2. Output signal in continuous mode with PWM at k_0=10 (duty cycle=0,2%)

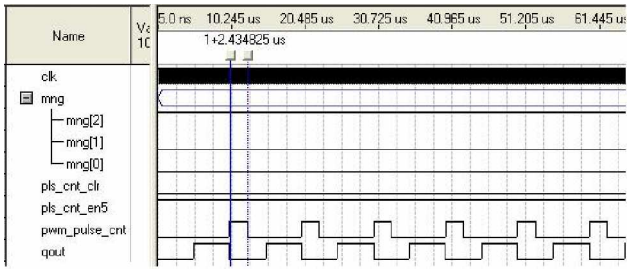

Fig. 3. Output signal in continuous mode with PWM at k_0=1000 (duty cycle =24,4%)

TABLE II
SETUPS IN THE .WVF FILE FOR THE VARIOUS MODES OF OPERATION

Control signals	*Continuous mode*		*Counter mode*	
	without PWM	*with PWM*	*without PWM*	*with PWM*
mng[0]	1	0	1	0
mng[2]	1	1	1	1
pls_cnt_en5	1	1	0	0
pls_cnt_clr	-	-	0	0

IV. CONCLUSION

One-channel programmable pulse synthesizer, operating in two basic modes - continuous and counter mode, with and without PWM, has been designed and simulated. These modes allow increasing the range of the possible applications of the synthesizer. The used novel direct digital synthesis determines very high PWM resolution – 2,4414 ns. The duty factor varies in wide range – from 1% to 99%. The simulations prove the proper operation of the circuit. The implementation is based on cheap FPGA with general purpose in which very small percent of elements and I/O pins has been used. This fact allows using the synthesizer as a building block of more complex devices and systems, implemented in one FPGA circuit.

REFERENCES

[1] A Technical Tutorial on Digital Signal Synthesis, Analog Devices, Inc., 1999.
[2] Cyclone Device Handbook, vol. 1, Altera Corp., 2005.

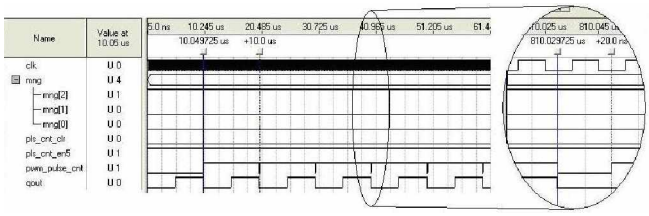

Fig. 4. Output signal in continuous mode with PWM at k_0=4090 (duty cycle =99,8%)

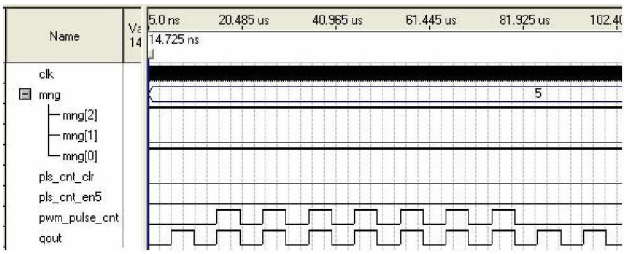

Fig. 5. Output signal in counter mode without PWM at k_2=6

TABLE III
PULSE DURATION AND DUTY CYCLE IN PWM SUB-MODES AT F_{CLK}=50 MHz, F_{OUT}=100 KHZ/ T_{OUT}=10 μs

k_0	τ_{Icalc}, μs	τ_{Isim}, μs	$DF_{sim.}$, %
5	logical 0	logical 0	0
10	0,02	0,02	0,2
20	0,04	0,04	0,4
30	0,07	0,06	0,6
40	0,09	0,1	1
50	0,12	0,12	1,2
100	0,24	0,24	2,4
500	1,22	1,22	12,2
700	1,7	1,7	17
1000	2,44	2,44	24,4
2000	4,88	4,88	48,8
3000	7,32	7,31	73,1
4000	9,76	9,75	97,5
4090	9,99	9,98	99,8
4095	logical 1	logical 1	100

[3] Hristov, M. et al. Development Systems for Microelectronics. Sofia, TU – Sofia Publishing, 2004 г. (in Bulgarian)
[4] Karailiev, H., H. Hinov, V. Rankovska. Modeling and Investigation of Inductive Cummulation Inverter. ICEST2005 Conference Proceedings, Nish, Serbia and Montenegro, 2005, pp. 237-240.
[5] Karailiev, H., V. Rankovska, "DDS Method for Generating a Frequency Grid at Systems for Test Control and Automated Regulation", ICEST 2006, Conference Proceedings, pp. 300-303, Sofia, Bulgaria, 2006.

[6] Quartus II Version 5.0 Handbook. Vol. 1: Design & Synthesis, Altera Corp., 2005.

[7] Tierney, Joseph, Charles M. Rader, Bernard Gold. A digital Frequency Synthesizer. IEEE Transactions on Audio and Electroacoustics, Vol. au-19, No. 1, March 1971, pp. 48-57