Journal of ELECTRICAL ENGINEERING, VOL. 60, NO. 5, 2009, 246–254

PLD MODELING OF ALL–DIGITAL DLL

Tomislav Švedek — Tomislav Matić — Marijan Herceg *

An all-digital delay-locked loop (DLL) suitable for implementation in the programmable logic device (PLD) is presented in this paper. Analog parts of the conventional DLL are realized by digital circuitry. Digital-controlled delay line (DCDL) is made of programmable digital-controlled delay elements (DCDE) based on the binary-weighted multiplex LCELL structures. Problems encountered in PLD implementation of the DLL are emphasized and discussed. Simulation and measuring results of the proposed DLL realized by ALTERA device EPM7128SLI10, are presented.

K e y w o r d s: all-digital DLL, PLD implementation, digital-controlled delay line

1 INTRODUCTION

In many areas, the Delay-Locked Loop (DLL) can be an alternative to the traditional frequency generation technique - the Phase-Locked Loop (PLL) [1]. The DLLs are used in the circuits where precise clock delays need to be defined [2], in the local oscillators for the PCS applications [3], in the frequency synthesizers of the UWB applications [4], frequency multipliers [5], clock and data recovery (CDR) applications and the clock generators in microprocessors, DSP, multi-core SoCs, etc. In some circuits, the DLL is preferred because of its better stability, smaller jitter accumulation, and shorter locking time, as compared to the PLL. The implementations of the DLL can be categorized by its delay elements which can be continuously variable or analog [4], discretely-ordered/continuous or analog-digital [2, 5], or all-digital [6].

The all-digital DLL was realized by the Altera development board (UP10), using logic functions available in the programmable logic device (PLD) EPM7128SLI10. Although the PLD, neither by its structure nor by its simulation tools, do not support such high-frequency delay-based circuits, the inability to access the appropriate VLSI technology forced us to consider this approach. The project was the part of the teaching process in the course "RF communications circuits", and its only purpose was to introduce the students to the DLL basics and its use as a frequency multiplier. The paper is organized as follows: Section II describes the basic structure of DLL and gives the current state of the art overview of the delay cells architectures. Section III describes a proposed all-digital DLL architecture and problems in its PLD implementation. Section IV compares simulation and experimental results, and Section V gives some conclusions.

2 DLL BASICS

A typical analog DLL consists of five basic elements (Fig. 1): a delay line (DL) with n-delay elements (DE), an edge combiner (EC), a phase discriminator (PD), and a charge pump (CP) followed by a loop filter (LF).

The performances of the DLL are defined by its lock range, lock time and jitter characteristics. DLL *lock time* is the time interval needed to attain a stable locking state (from the initial state to the f_{REF} lock). It depends on the speed of the PD, the time needed for charging and discharging current in the CP, and the delay introduced by the loop filter. DLL lock range directly defines the DLL operating frequency range ($f_{OUTmax} - f_{OUTmin}$) which is based on the minimal and maximal delay achieved by the DL. The DLL time jitter is a time period variation of the generated output frequency f_{OUT}. For the higher reference frequency, the output period is smaller and the influence of the time jitter increases.

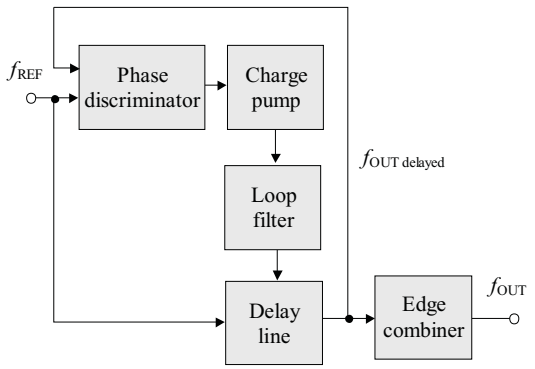

Fig. 1. Block diagram of an analog DLL

Input signal with the reference frequency f_{REF} propagates through the delay line - DL, where each of its delay elements - DE causes a unit phase shift (or a unit time delay) of this signal. The unit delay duration of each DE can be adjusted and controlled by the loop filter voltage (V_C). The phases of the delayed output signal $f_{REFdelayed}$ and the reference signal f_{REF} are compared within the PD. According to the phase difference, the CP increases or decreases the loop filter voltage V_C, which controls the time delay duration of each DE. A stable locking state

* Department of Communications, Faculty of Electrical Engineering, J. J. Strossmayer University of Osijek, Kneza Trpimira 2b, Osijek, Croatia, tomislav.svedek@etfos.hr

ISSN 1335-3632

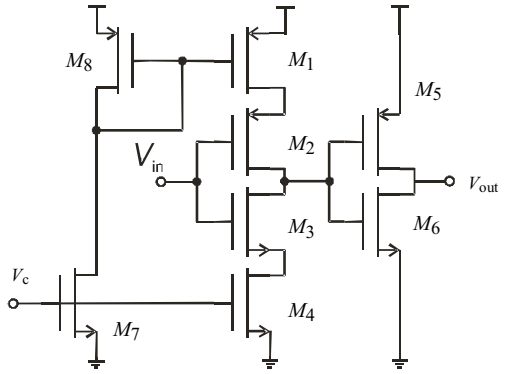

Fig. 2. Current starved delay element [4]

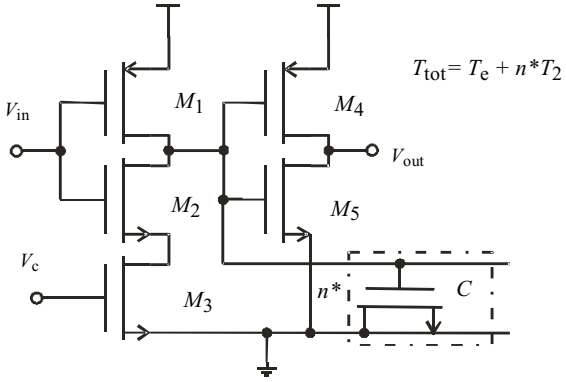

Fig. 3. Analog-digital delay element with load capacitor scaling

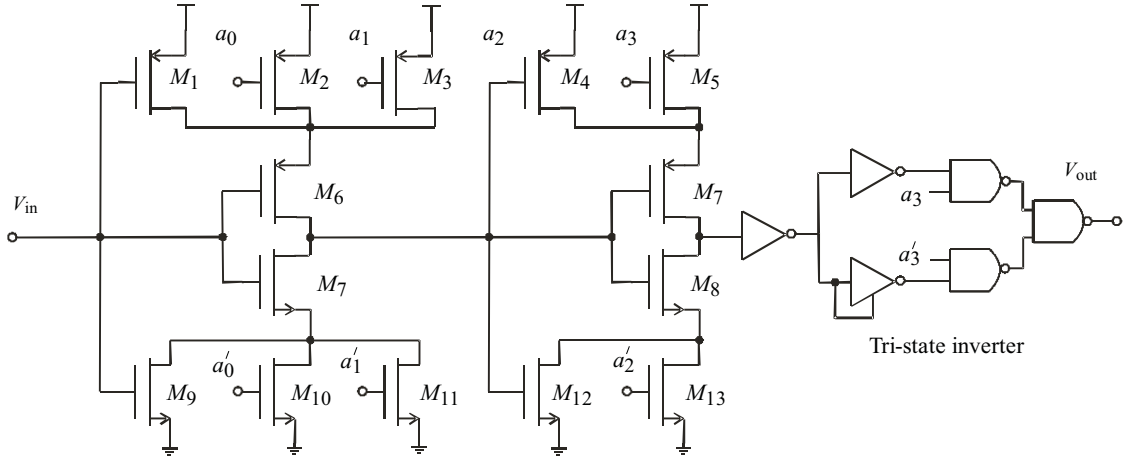

Fig. 4. Analog-digital delay element with dynamic binary-weighted paralleling of the triode region transistors of the current starving inverters. Delay generation: $a_0 = 20\,\text{ps}$, $a_1 = 20\,\text{ps}$, $a_2 = 80\,\text{ps}$, $a_3 = 160\,\text{ps}$,

is achieved when the phases of the input reference signal and delayed reference signal are matched. A pulse train, formed at the DLL output by the edge combiner - EC circuit, has frequency that is n-times higher than the frequency of the input reference signal $f_{OUT} = n\, f_{REF}$.

2.1 Overview of Delay Cells Architecture

The main issue of the DLL design is the creation of the delay element with controllable delay. There are plenty examples of the DLL delay element architectures in present literature; they can be continuously variable or *analog*, discretely-ordered/continuous or *analog-digital*, or *all-digital*. In all architectures the basic analog delay element is the current starved delay cell. One of its version, presented in [4] by T. C. Lee and K. J. Hsiao, and in [7] by Zhuang, Du and Kwasniewski, is shown in Fig. 2. The delay element is a single-ended current starved inverter, consisting of M_2 and M_3 in series with M_1 and M_4 operating in the triode region. The equivalent resistance of M_1 and M_4, controlled by analog voltage V_C, determines the delay of the circuit, *ie* falling and rising times at its output. An additional inverter comprising M_5 and M_6 serves as an output buffer for higher frequency operation.

Two examples of the discretely-ordered/continuous or *analog-digital* delay elements are also based on the current starved inverter. The first is the analog delay element with non-dynamic discretely scaled load capacitor of the buffer, by Jansson *et al* [2], and the second is the analog delay element with dynamic binary-weighted paralleling of the triode region transistors, by Wang at all in [5].

The structure of the non-dynamic delay element, used in the fine interpolator [2], is shown in Fig. 3. It consists of current starving inverter followed by the output buffer. The basic delay T_e of the current starving inverter is controlled dynamically by the analog control voltage V_C, and the very small delay differences between delay elements are made by scaling the load capacitors of the inverter. The exact delay difference between two delay elements ($n\,T_2$) is realized by having a specific number n of unit capacitance loads in the delay elements. In this application, the unit capacitance loads are defined by the design (layout) of the delay element, and cannot be changed during the work (non-dynamic).

An analog delay element with dynamic discretely changed delays, used in fine delay line [5], is shown in Fig. 4. It consists of two current starving inverters, and the variable delay cell (VDC) composed of the three

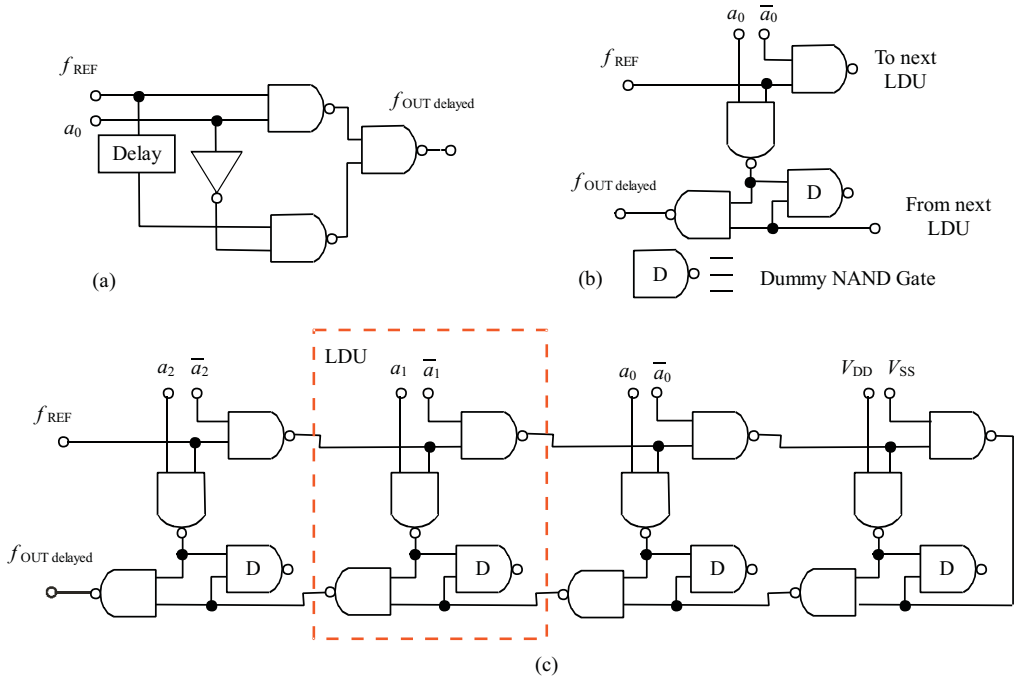

Fig. 5. All-digital delay element, b) lattice delay unit - LDU, c) lattice delay line - LDL

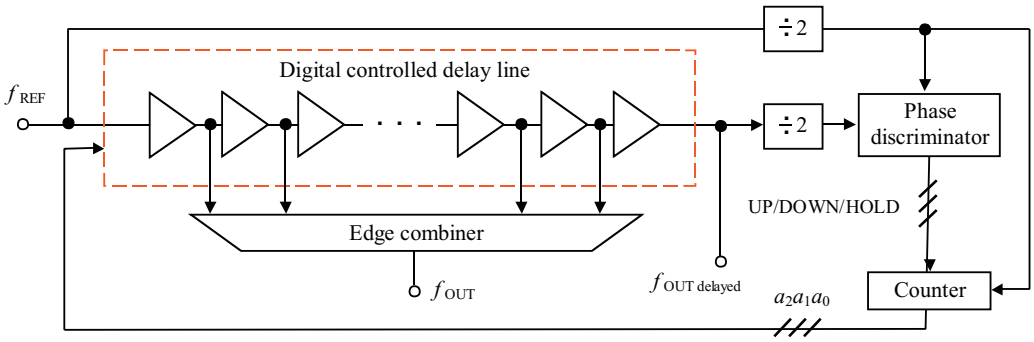

Fig. 6. All-digital DLL

NAND gates, three inverters and one tri-state inverter. Four binary-weighted bits, a_1 (LSB) to a_2 (MSB), control the delay of the fine delay line. Those bits are generated in the phase discriminator circuit. The a_1 and a_2 control paralleling of the triode region transistors of the first inverter, a_3 controls the same action for the second inverter, and a4 controls the delay path of VDC. The reported tuning range is 320 ps, and minimum timing resolution is 20 ps. This analog delay element performs discrete, but dynamical change of the delay; see Table in Fig. 4.

One of the many all-digital delay elements is presented in [6] by R.J. Yang and S.I. Liu. It is conventional digital controlled delay unit with two different delay paths controlled by a multiplexer (Fig. 5a). To expand the operating frequency range, the number of cascaded delay units is increased. The maximum operating frequency is restricted by the large intrinsic delay. The proposed lattice delay unit (LDU) is shown in Fig. 5b) and the lattice delay line (LDL) constructed by cascading the LDUs is shown in Fig. 5c). The binary-weighted control bits are converted into the thermometer codes (T0. T1, T2). Both the intrinsic delay and the delay step in an LDL are the delay of two NAND gates. As the operating frequency increases, the number of activated delay units is reduced and the power consumption remains the same. Several dummy NAND gates are used to match the loading effect ant the numbers of fan-in and fan-out. Even this, all-digital solution, uses the current starving inverters as the fine-tuning delay units, which are incorporated in front the LDL (not shown in Fig 5.)

3 DIGITAL DLL ARCHITECTURE AND PROBLEMS IN ITS PLD IMPLEMENTATION

Basic idea behind this project is the design of an all-digital DLL. That means that for its implementation only digital elements and the building blocks available in the PLD functions library have to be used, and that all analog parts from Fig.1 should be converted into their digital counterparts. The complete structure of an all-digital DLL is shown Fig. 6.

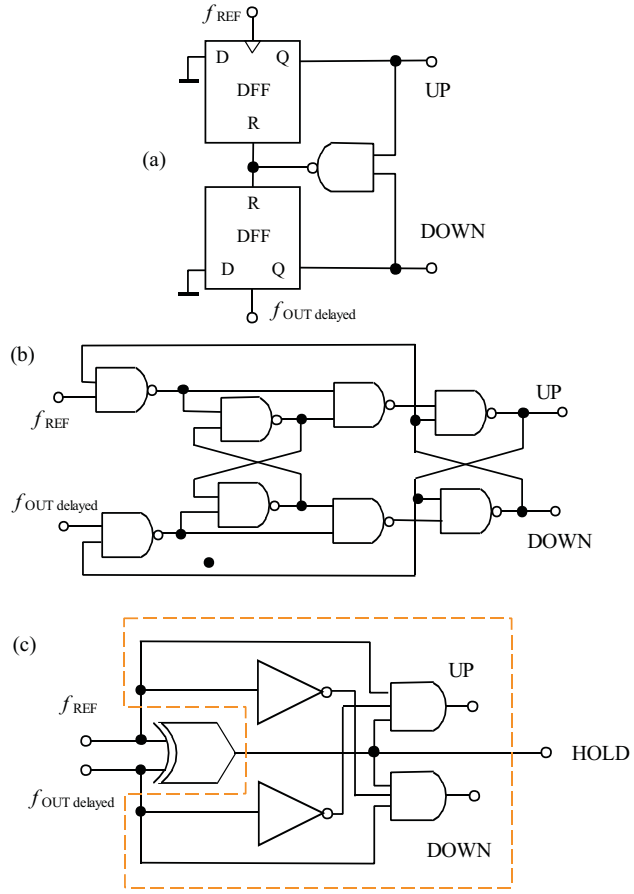

Fig. 7. Phase discriminators (PD)

3.1 Phase Discriminator and M-bit Binary Up/Down/Hold Counter

Control part of the analog DLL consists of the phase discriminator, charge pump and loop filter (Fig.1). Outputs of the delay line and the reference signal are fed into the phase discriminator. When the phases of these two signals are mismatched, charge pump increases or decreases the charge injection, depending on the phase lag of the reference or delayed signal. The loop filter smooth-out this signal, creating the low frequency control voltage V_C that is used further on for the delay change of the delay element - DE.

There are, in general, two types of phase discriminators, analog and digital [8]. When the analog types are built from the analog circuits that have been applied for analog computation, the digital types are based on digital circuits, such as the logic gates: AND, OR, EX-OR, D-FF, *etc.* Digital types of phase discriminators operate on the binary signals exclusively, and are usually combined with a m-bit binary UP/DOWN/HOLD counter that replaces the analog charge pump and the loop filter of the analog DLL. When the delay of the DE has to be increased/decreased (phase mismatch), the PD generates the UP/DOWN signal, which increments/decrements m-bit counter. When the phases are matched, the delay of DE should stay unchanged, and the PD generates the HOLD signal, freezing the m-bit counter. So, the PD and m-bit UP/DOWN/HOLD counter form the control bits $[a_2^{m-1} \ldots a_1, a_0]$, where $a_i \in \{0, 1\}$, by which the delay of the DE is adapted in the conformity with the input reference frequency.

Phase discriminators, shown in Fig. 7, are digital circuits and they require square waves for both reference and output signals. Fig. 7a shows a latched type of phase discriminator, often referred to as a phase/frequency detector. The UP and DOWN signals are defined here as active-low signals. In the case of zero phase error, both UP and DOWN signals are permanently HIGH. If the signal f_{OUT} delayed lags the reference signal f_{REF}, the UP output generates pulses with a duty-cycle ratio proportional to the phase error. In the opposite case, DOWN output is pulsed.

The phase discriminator in Fig. 7b is a combination of the two edge-triggered D-FFs. It is phase- and frequency-sensitive PD. The output of the upper FF (output UP) is set HIGH by the falling edge of the signal f_{REF}, and it is set low by the falling edge of the signal f_{OUT} delayed.

The simplest of three devices is EX-OR gate with HOLD output (Fig. 7c), which is modified with two added UP and DOWN outputs. If the signals f_{REF} and $f_{OUTdelayed}$ are exactly in phase, the output HOLD is set to zero and $f_{OUTdelayed}$ will not change the phase. If the signals f_{REF} and $f_{OUTdelayed}$ are shifted in the phase, the output UP or DOWN will change the phase of the $f_{OUTdelayed}$, according to the difference in the signals phase. This modified EX-OR phase discriminator has been adopted for the realization, because of the implementation simplicity and the existing of the UP/DOWN/HOLD outputs. The counter output *vs* phase error, and the PD transfer characteristic are shown in Fig. 8.

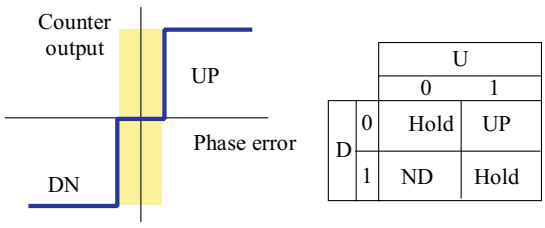

Fig. 8. Counter output *vs* phase error and the PD transfer characteristic

3.2 Digital Controlled Delay Line

In all digital realization of the DLL circuit the first and the main problem is the design of the programmable, digitally controllable delay elements-DE. In the analog design, time delays of an inverter or buffer cell are usually taken as the unit delay T_0.

However, PLD manufacturers do not recommend the inverter or the buffer cells as the time delay elements.

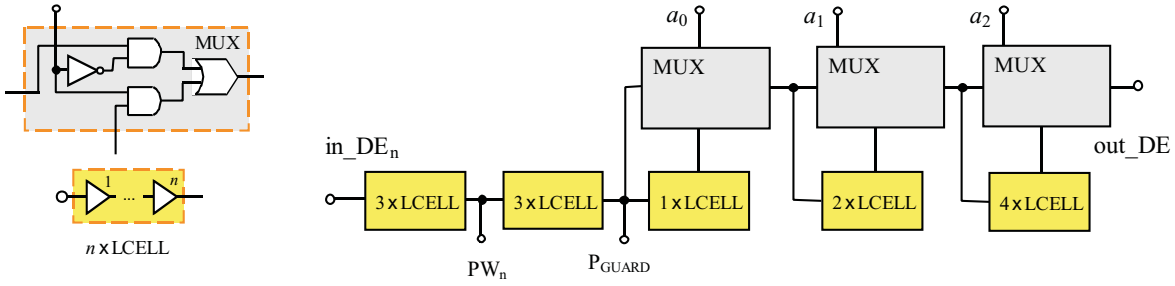

Fig. 9. Digital-Controlled Delay Element (DCDE) for $m = 3$

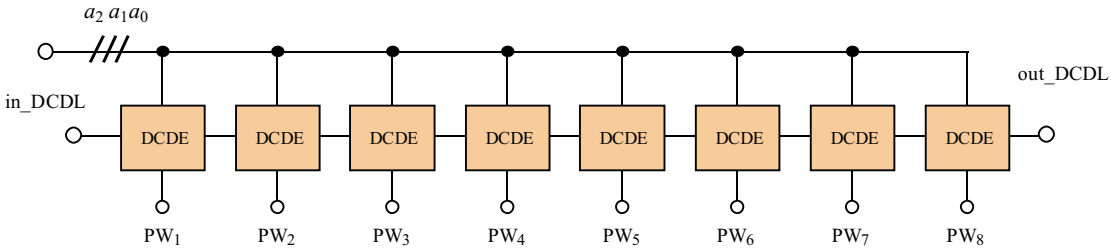

Fig. 10. Digital-Controlled Delay Line (DCDL) for $n = 8$

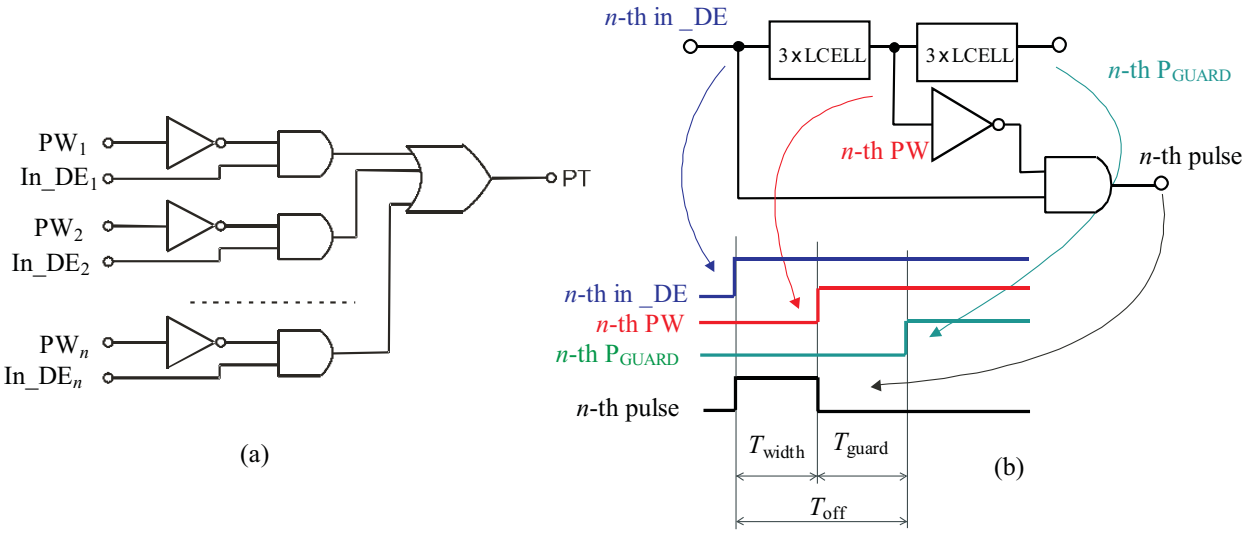

Fig. 11. Edge combiner a) (EC) and pulse forming process in EC b)

Instead, they recommend a LCELL element (LCELL delay) and an EXP cell, which, however, have very poorly defined delays, that vary with the power supply and local routing in the PLD. We have carried out a simulation and measurement of delay times for LCELL and EXP cells (Table I, rows a and b) and have also built a DL with D-flip-flops as a delay element (Table I, row c). For this DLL implementation, we have chosen the LCELL as a delay element. Because the PLD simulation tools use only the worst-case delay, the measured delays ($T_{0M} = 4.4$ ns) are much shorter than the simulated ones ($T_{0S} = 8$ ns). That is one of the reasons for a relatively large discrepancy (more than 50 %) between the simulated and measured results of the DLL output frequencies. The measured frequency is almost twice the simulated.

Table 1. Simulation and measurement results of delay lines - DL realized by a) LCELL, b) EXP and c) D-flip-flop cell

	Type of the delay cell	Delay (ns) T_{0S}	Delay (ns) T_{0M}
(a)	IN — LCELL LCELL — OUT	8.0	4.4
(b)	IN — EXP EXP — OUT	5.0	2.7
(c)	V_{CC}; IN, D, R, S, OUT	9	5

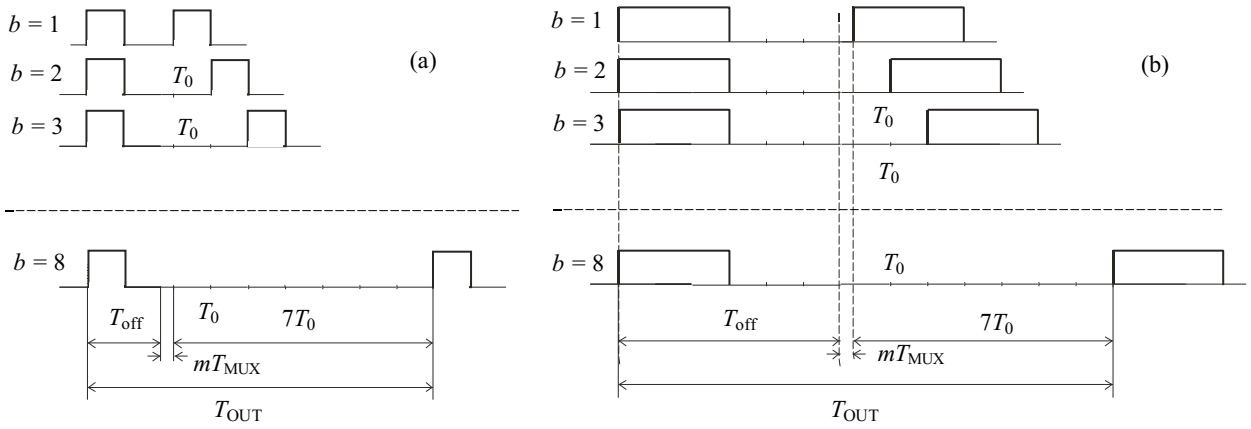

Fig. 12. T_{OUT} with: (a) – $T_{off} = 2T_0$, (b) – $T_{off} = 6T_0$

3.3 Digital Controlled Delay Element And Maximal Output Frequency

The Digital-Controlled Delay Element (DCDE) is based on the MUX structure (Fig. 9 illustrates the case of $m = 3$). According to the combination of the$m$ control bits $[a_2^{m-1} \dots a_1, a_0]$ generated in the counter of the phase discriminator, multiplexers serially combine binary-weighted LCELL delays, resulting with the following delay increments

$$T_{\mathrm{OUT}}(b) = (b-1)T_0 + mT_{\mathrm{MUX}} \tag{1}$$

where m is the DE resolution, $b \in \{1, 2, .2^m\}$ is the weighting factor formed by the control bits combination $[a_2^{m-1} \dots a_1, a_0]$, T_0 is the unit delay of the LCELL and T_{MUX} is the delay of the each MUX circuit. The Digital-Controlled Delay Line (DCDL) consists of n serially connected DCDE (Figure 10 illustrates the case of $n = 8$). So, the maximal multiplication factor can be equal to $f_{OUTMAX} = n\, f_{REF}$.

Because of the short delay of the LCELL and the long rise time (the output rise time of LCELL is longer than the single delay of LCELL), the edge combiner - EC circuit (Fig. 11 a) cannot produce the measurable output pulse at the edge of the delayed signal. To cope with this problem, the width of the EC output pulse Twidth has been expanded to $3T_0$, and the guard time T_{guard} of the same duration was added behind the EC output pulse. Those six LCELLs are added in front of the MUX-ed binary weighted LCELLs, which is together with pulse forming process shown in Fig. 11b).

Consequently, (1) should be expanded by this offset time T_{off}

$$T_{\mathrm{OUT}}(b) = T_{\mathrm{off}} + (b-1)\,T_0 + mT_{\mathrm{MUX}} \tag{2}$$

where $T_{off} = T_{width} + T_{gurad} = 6T_0$ is now the minimum pulse width (the highest output frequency). Taking $T_{MUX} \approx 0$, the output DLL frequencies $f_{OUTmax0}$ and $f_{OUTmin0}$ (where index "0" stands for $T_{off0} = 2T_0$ - ideal case with shorter rise time) can be defined as

$$f_{OUT\,\max 0}|_{b=1} = \frac{1}{2T_0 + mT_{MUX}} \approx \frac{1}{2T_0} \tag{3}$$

$$f_{OUT\,\min 0}|_{b=2^m} = \frac{1}{2T_0 + (2^m - 1)T_0 + mT_{MUX}} \approx \frac{1}{(2^m+1)T_0} = \frac{2}{2^m+1} f_{OUT\,\max 0} t \tag{4}$$

From Fig. 11(a) can be seen that T_{off} has impact on the controllable part of the delay T_{OUT}, and according to (3) and (4) to f_{OUTmax} and f_{OUTmin} od the DLL. The longer T_{off}, the smaller controllable part and lower output frequencies.

For the 3-bit resolution $(m = 3), T_0 = 8$ ns V(EPM7 128SLI10), $T_{off} = 6T_0$ and T_{MUX} set to 0, simulated delay values are shown in Table II. As can be seen, they can be digital-controlled in the range of 48 to 104 ns, in 8 ns increments. MAX and MIN output frequencies are decreased, so (3) and (4) became

$$f_{OUT\,\max} = \frac{1}{6T_0} = \frac{f_{OUT\,\max 0}}{3} \tag{5}$$

$$f_{OUT\,\min} = \frac{2^m+1}{2^m+5} f_{OUT\,\min 0} = \frac{2}{2^m+5} f_{OUT\,\max 0}. \tag{6}$$

Table 2. Simulated delay values for $M = 3$

$a_2a_1a_0$	Delay T_{delay}	Frequency T_{delay}	Simulated delay (s)
000	$6T_0$	$1/6\,T_0$	48
001	$7T_0$	$1/7\,T_0$	56
010	$8T_0$	$1/8\,T_0$	64
011	$9T_0$	$1/9\,T_0$	72
100	$10T_0$	$1/10\,T_0$	80
101	$11T_0$	$1/11\,T_0$	88
110	$12T_0$	$1/12\,T_0$	96
111	$13T_0$	$1/13\,T_0$	104

The consequence of the introduction of the offset delay $T_{off} = 6T_0$ is pushing the MAX and MIN output frequencies to lower values, and the MAX/MIN frequency ratio is correspondingly smaller. That means the lowering of DLL operating frequency range $B_{OUT} = f_{OUTmax} - f_{OUTmin}$

$$B_{OUT} \approx \frac{2^m - 1}{3(2^m + 5)} f_{OUT\,\max 0} \tag{7}$$

Figure 13. shows the narrowing of the B_{OUT} with the increase of the offset time - T_{off}

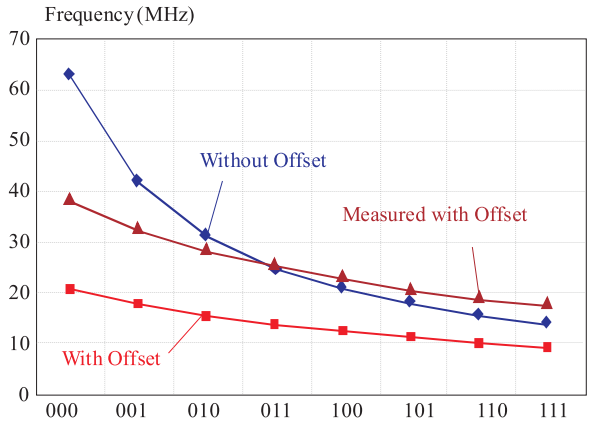

Fig. 13. The narrowing of the B_{OUT} with the increase of the T_{off} time

For the defined m, T_{off}, T_0 and T_{MUX} , the output DLL frequencies f_{OUTmax} and f_{OUTmin} are constant. Input DLL frequencies f_{INmax} and f_{INmin} should be chosen according to the output frequencies and the number of the feedback stage n, as

$$f_{IN\,\max} = \frac{1}{n} f_{OUT\,\max} = \frac{1}{n} \frac{1}{T_{off} + mT_{MUX}} \tag{8}$$

$$f_{IN\min} = \frac{1}{n} f_{OUT\,\min} = \frac{1}{n} \frac{1}{T_{off} + (2^m - 1)T_0 + mT_{MUX}} \tag{9}$$

If DLL operating frequency range B_{OUT} is constant, the input DLL referent frequency range $B_{IN} = f_{INmax} - f_{INmin}$ is narrower for the increased n (see Fig. 14. and Table III.). In Fig.14. we can for $f_{IN} = f_{REF} =$ constant, find the corresponding output frequencies multiplied by factors $n = 7, 8,$ and 9. The DLL output frequency depends on the unit delay T_0 of LCELL, that significantly differs in simulation (8 ns) and in measurement (4,4 ns). Without the introduction of the T_{off}, simulated MAX output frequency would be equal to 125 MHz (for input of 15,625 MHz), MIN output frequency would be equal to 15,625 MHz (for input of 1,953 MHz), and the MAX/MIN frequency ratio will be increased from 2,1 to 8.

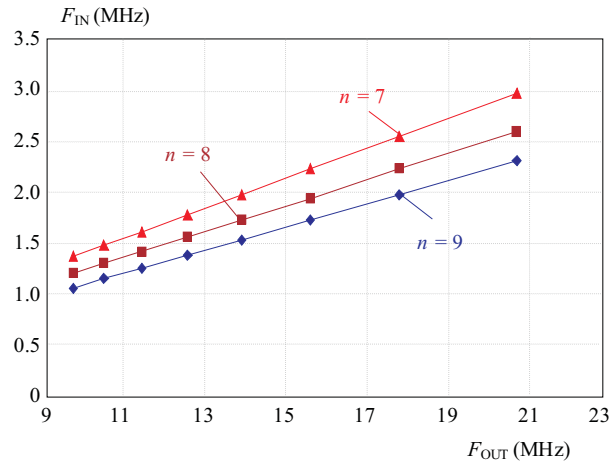

Fig. 14. f_{IN} *vs* f_{OUT} for different number of delay cells - n.

Table 3. Simulated f_{IN} values for $n = 7, 8, 9$.

f_{OUT} (MHz)	f_{IN} $(n = 9)$	f_{IN} $(n = 8)$	f_{IN} $(n = 7)$
9.62	1.07	1.20	1.37
10.42	1.16	1.30	1.49
11.36	1.26	1.42	1.62
12.50	1.39	1.56	1.79
13.89	1.54	1.74	1.98
15.63	1.74	1.95	2.23
17.86	1.98	2.23	2.55
20.83	2.31	2.60	2.98

4 EXPERIMENTAL RESULTS

The proposed digital DLL circuit was implemented by the MAX+plus II software and tested on the Altera development board (UP10), using EPM7128SLI10 device. Because of the PLD resources restriction, DLL was realized by only eight delay elements (multiplication factor $n = 8$), and with 3-bit resolution $(m = 3)$. Fig. 15 shows the measured input and output signals of the DLL, where the upper trace of the oscillogram shows the input reference signal set to 2,37 MHz. The lower trace displays a pulse train of eight time higher frequency, which is, according to Table II., equal to 18,94 MHz.

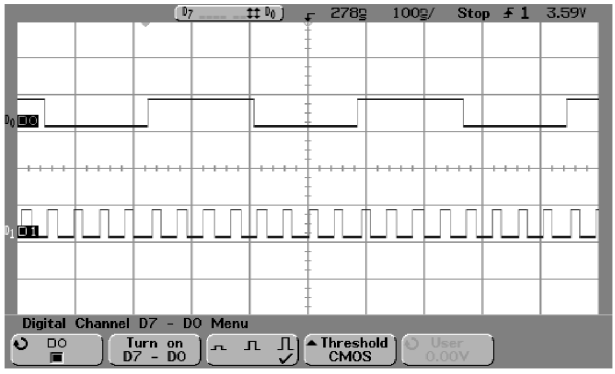

Fig. 15. Reference clock and pulse train

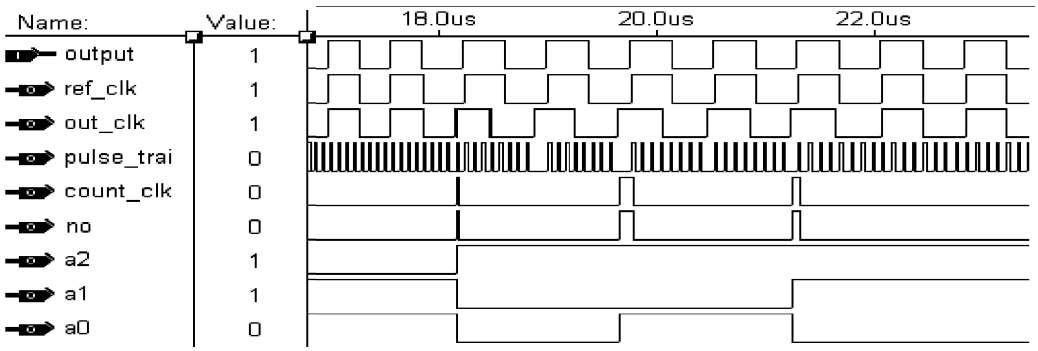

Fig. 16. Reference clock and pulse train

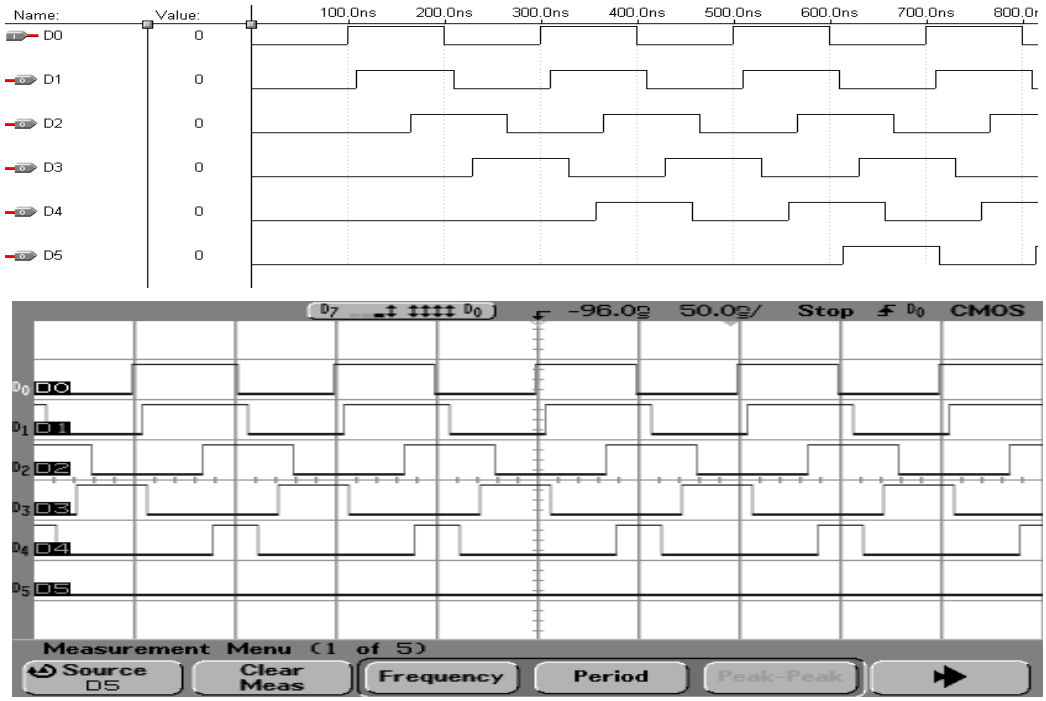

Fig. 17. (a) — Simulated, and (b) — measured waveforms of input and delayed output 10 MHz signals showing the DCR lowering

Table 4. Simulated and measured Dll input and output frequencies for $M = 3$

Input clock frequency		Output clock frequency	
Simulated	Measured	Simulated	Measured
(MHz)	(MHz)	(MHz)	(MHz)
2.60	4.73	20.83	37.87
2.23	4.06	17.86	32.46
1.95	3.55	15.63	28.41
1.73	3.16	13.89	25.25
1.56	2.84	12.50	22.72
1.42	2.58	11.36	20.66
1.30	2.37	10.41	18.94
1.20	2.19	9.61	17.48

Table 4 shows simulated and measured values of the output frequencies and appropriate input frequencies. Apart from the worst-case delay simulation, another reason for the simulation vs. measurement discrepancy is fact that the Max+Plus II software does not support back annotation, *ie* the influence of routing and placement on simulation.

The delay locking process is shown in Fig. 16. The locking process is finished when the sum of periods of 8 pulses becomes equal to the period of the input reference clock. The whole process is finished for the three periods of the referent signal.

Due to the different rise and fall times of the Altera EPM7128SLI10 LCELLs, the DCDL created by these cells does not retain the duty cycle ratio (DCR) of the reference input clock signal. That is, although the input signal has the DCR equal to 50%, the delayed output signal has the DCR lower than 50%. Such a situation does not appear in the simulation but only in the measurement. Figure 17 shows the simulated (a) and measured (b) waveforms of the input signal and delayed output signals for different number of serially connected LCELLs, at the f_{REF} equal to 10 MHz. Probe D0 stands for the

reference input signal, while probes D1 to D5 stand for the LCELL series of $n = 1, 8, 16, 32$ and 64 LCELLs, respectively. The same measurement but for different input frequencies is also shown in Table5. It can be seen that for the higher frequencies and for the higher number of LCELLs, the DCR of the output signal gets lower.

Table 5. Duty cycle ratio (DCR) *vs* input reference frequency and number of LCELLs (N)

	Input signal frequency — f_{REF}					
N	100 Hz	1 kHz	10 kHz	100 kHz	1 MHz	10 MHz
1	50	50	50	50	50	50
8	50	50	50	50	49.50	42.50
16	50	50	49.95	49.90	48.50	35.00
32	50	50	49.75	49.70	47.00	20.00
64	50	49.8	49.60	49.40	43.50	0

Values are in %

From Table V, the empirical relationship of the DCR deterioration *vs* input reference frequency and the number of the serially connected LCELLs can be obtained as $\mathrm{DCR} = (0.5 - N\,\Delta T\,f_{REF})\,100$ (%), where N stands for the number of serially connected LCELLs, and ΔT is the difference between rise and fall time of the LCELL. The measured value for ΔT is 0.94 ns. The chosen phase comparator is very sensitive to duty-cycle. To cope with this problem, the reference input signal and delayed output signal are fed to T-FF (dividing by 2) before they are compared in the phase discriminator (see Fig. 6).

5 CONCLUSION

This paper describes the practical implementation and the problems facing the implementation of an all-digital-DLL in Altera device EPM7128SLI10 PLD. The proposed circuit has been simulated by the MAX+Plus II simulation tools. Despite the implementation problems, simulation results clearly show the basic operation of an all-digital-DLL: frequency multiplication, delay programming, locking process, what was the main goal of this educational student project.

Resources of the used PLD limit the achieved frequency range and the resolution of the realized DLL. One LCELL consumes one of 128 macro cells available in the Altera EPM7128SLI10 and the design with a 3-bit resolution DLL requires 104 LCELL elements. It can be expected that using of the higher capacity PLD and more precise control of the placement and routing can improve the resolution and give a wider the frequency range of proposed DLL circuit.

Acknowledgement

This paper was supported by the Ministry of Science, Education and Sports in the Republic of Croatia under the project 165-0361630-3049.

References

[1] BEST, R. E.: Phase-Locked Loops: Theory, Design and Applications, McGraw-Hill, New York, 1998.

[2] JANSSON, J.-P.—MANTYNIEMI, AS.—KOSTAMOVAARA, J.: A CMOS Time-to-Digital Converter With Better Than 10 ps Single-Shot Precision, IEEE Journal of Solid-State Circuits **41** No. 6 (June 2006).

[3] CHIEN, G—GRAY, P. R.: A 900-MHz local oscillator using a DLL-based frequency multiplier technique for PCS applications, IEEE Journal of Solid-State Circuits **35** No. 12 (December 2000).

[4] LEE, T.-C.—HSIAO, K.-J.: The Design and Analysis of a DLL-Based Frequency Synthesizer for UWB Application, IEEE Journal of Solid-State Circuits **41** No. 6 (June 2006).

[5] WANG, Y.-J.—KAO, S.-K.—LIU, S.-T.: A 40-550 MHz harmonic-free all-digital delay-locked loop using a variable SAR algoritam, IEEE Journal of Solid-State Circuits **42** No. 6 (June 2006).

[6] YANG, R.-Y.—LIU, S.-I.: A 40-550 MHz harmonic-free all-digital delay-locked loop using a variable SAR algorithm, IEEE Journal of Solid-State Circuits **41** No. 2 (February 2007).

[7] ZHUANG, J.—DU, Q.—KWASNIEWSKI, T.: A -107dBc, 10kHz Carrier Offset 2-GHz DLL-Based Frequency Synthesizer, in Proc. IEEE Custom Integrated Circuits Conf. Sep. 2003, 301-304.

[8] BEST, R. E.: PHASE-LOCKED LOOPS Theory, Design, and Applications, McGraw-Hill, New York 1984.

[9] MATIC, T.—SVEDEK, T.—HERCEG, M.: Problems in PLD Implementation of All-digital DLL, 2008 3rd International Symposium on Communications, Control and Signal Processing (ISCCSP 2008), St. Julians, Malta.

Received 26 February 2009

Tomislav Švedek received BSc, MSc and PhD degrees form Faculty of Electrical Engineering Zagreb, University of Zagreb, Croatia. He joined Institute for Electronics, Telecommunications and Automation (IETA) RIZ Zagreb, in 1975 as development engineer, PTT Project buerau, Zagreb in 1980 as project leader, Electrotechnical Institute of Rade Konar, Department for design of Integrated Circuits in 1986 as ASIC supervisor, and Faculty of Electrical Engineering, University of Osijek in 1993, as Full professor for Electronic circuits and Microelectronics. His scientific interests are design and testability of ASICs, microelectronic RF circuits, and digital modulation methods.

Tomislav Matic was born in Osijek, Croatia, in 1978. He received BS degree from Faculty of Electrical Engineering in Osijek on 2002. Currently he is assistant at Department of Communications at Faculty of Electrical Engineering in Osijek, where he is working towards PhD degree in Communications and his research interests are A/D conversion, sigma-delta A/D converters and FPGA programming for communications applications.

Marijan Herceg was born in Osijek, Croatia, in 1978. He received the BS degree in electrical engineering from Faculty of Electrical Engineering in Osijek, Croatia in 2002. He is currently working towards PhD degree in communication and informatics at the Faculty of Electrical Engineering in Osijek, Croatia and his research interest are advanced modulation techniques like power efficient modulations for ultra-wide band (UWB) and FPGA programming for communications applications.