

THE METHOD AND CIRCUIT FOR TESTING A CELLULAR NEURAL NETWORK WITH 16 CELLS

BY

NICOLAE PATACHE^{1,*}, VICTOR-ANDREI MAIORESCU¹
and LIVIU GORAȘ^{1,2}

¹ “Gheorghe Asachi” Technical University of Iași
Faculty of Electronics, Telecommunications and Information Technology,
²Institute of Computer Science,
Romanian Academy, Iași Branch, Romania

Received: February 3, 2015

Accepted for publication: February 17, 2015

Abstract. In this paper we present a method and a circuit for testing a 1D Cellular Neural Network with 16 cells. The main purposes of the circuit are to provide analog and digital stimuli for the testing circuit and to collect the analog response from the circuit.

Key words: Cellular Neural Networks; FPGA testing; Nexys3.

1. Introduction

The testing circuit is organized around a digital block which has two basic functions: applying stimuli on the circuit inputs (both for analog and digital control inputs) and collecting the results from the circuits (analog outputs). For extensive testing a high flexibility is needed which is achieved by offering the possibility to modify the stimuli parameters, the amplitude values, the wave-form and the timing.

To design the testing device for CNN we consider the following CNN features:

- a) charging (presetting) the initial state of the CNN cells;

*Corresponding author: *e-mail*: patachen@etti.tuiasi.ro

- b) discharging (reading) the final state of the CNN cells;
- c) maintaining the CNN in a waiting state;
- d) coupling the CNN in a free state;
- e) starting/stopping on command the transconductors or maintaining them on.

Beginning from the CNN functionality description (Patache *et al.*, 2014), on the control logic of the network we identified:

Digital control signals:

1° INIT signal: on the negative edge, the selection circuit of the CNN cells is reset by selecting the first cells for the charging of the state;

2° clkCNN signal: on the negative edge, it selects the next cell or it moves on to the next phase of the CNN function;

3° POWER_UP signal: it controls the function of the transconductors that are internal to the CNN circuit; active level is given by 0 logic

4° FREEZE signal: it controls the actual function of the CNN; when the signal is in 1 logic, the outputs of the transconductors are disconnected from the capacitors; when the signal is in 0 logic, the outputs of the transconductors are connected to the capacitors.

The functioning states:

i) IDLE state: the circuit maintains the CNN state; the storage of the state is created analog and is limited in time by the discharging of the capacitors due to the leakage currents; in this state we can configure interconnections between the network cell, by selecting for the whole functioning cycle a predefined topology;

ii) POWER UP state: it is the state in which all the transconductors of the network are on; this is obtained when the POWER_UP input signal goes in the active state, 0 logic;

iii) POWER DOWN state: it is the state in which all the transconductors of the network are off; this is obtained when the POWER_UP input signal is in the inactive state, 1 logic;

iv) LOADx state: it is the state in which a cell is selected and the state voltage of the cell is modified (charging);

v) SAVEx state: it is the state in which a cell is selected and the state voltage is observed at the output (reading);

vi) the DYNAMIC state corresponds to the coupling of the trans-conductor's outputs on the capacitances and the linear function of CNN until it reaches nonlinearity.

A number of actions that determine transition between states:

a) negative edge on INIT: resets the selection mechanism of the cell and forces CNN into LOAD0 state;

b) negative edge on CLK: selects the next cell (LOADx state);

c) negative edge on CLK: CNN goes into a waiting state;

d) negative edge on POWER_UP: all the transconductors are on;

- e) rising edge on FREEZE: the outputs of the transconductors are connected to the capacitances, the network begins to evolve;
- f) negative edge on FREEZE: the outputs of the transconductors are disconnected from the capacitances;
- g) negative edge on CLK: selects the next cell (SAVE_x state);
- h) negative edge on CLK: forces CNN in IDLE state.

2. Implementation of the Circuit

Corresponding to the two basic functions, applying stimuli and collecting results, in the structure of the testing circuit we will find two functional blocks responsible for generating stimuli and collecting results respectively, coordinated by a control block. The user interface is created with a block whose role is to get the waveform, the commands and to plot the results.

Implementation will be a test circuit storing digital waveforms, digital memory areas dedicated to the analog signals needed and will use A/D, D/A converters respectively, digitally controlled. These A/D and D/A converters will have in turn their own control with a synchronized operation. The internal structure of the test circuit is shown in 0.

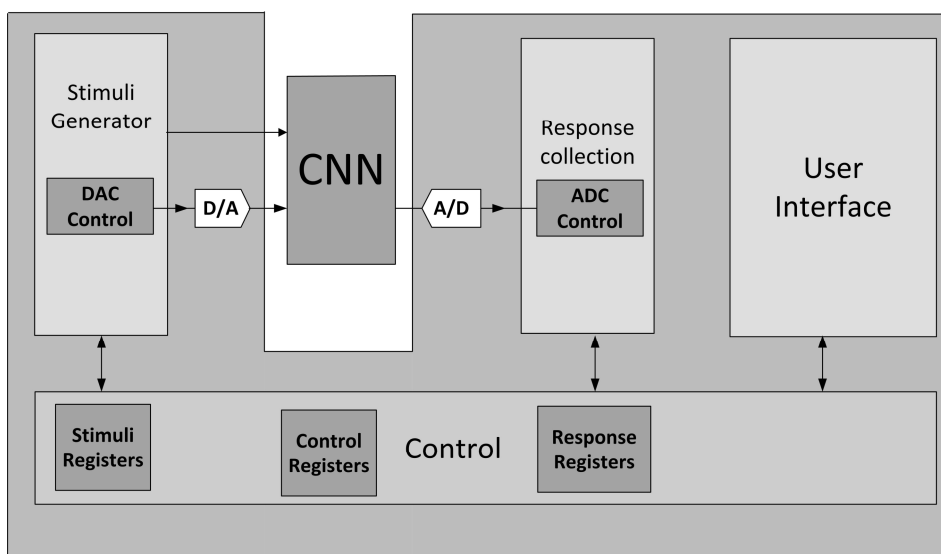


Fig. 1 – Internal structure of the test circuit and connections to the CNN circuit.

This structure of the test circuit is a flexible one, allowing the implementation of digital logic with the use of peripheral FPGA development board, to achieve user interface and the use of an ADC, DAC circuit respectively, connected as peripherals.

To test the control circuit a basic operation was defined via a simple

user interface with the resources of the test board: the inputs are made using switches (switch - SW), buttons (push button-PB), visual indicators (LED), numerical display with 7-segment cells (7SEG).

Control of the operation of the circuit will be done with a push-button called START and a LED will be ON for the whole duration of testing. By pushing the START button a full testing sequence will be triggered and cannot be stopped until it normally completes.

A complete sequence of operation includes:

- a) charging the initial state from the memory of the testing circuit to the cells (LOADx states);
- b) a waiting state after charging (RUN1 state);
- c) an activation command for the POWER_UP signal followed by a waiting time (RUN2 state);
- d) an operational state with controlled time (RUN3 state);
- e) a waiting state (RUN4 state);
- f) a deactivation command for the POWER_UP signal followed by a waiting time (RUN5 state);
- g) saving the final state of the cell from the test circuit's memory (SAVEx states).

To control the operating time (RUN3 state) and the initial state, the user will have the possibility to modify these values using two buttons (INC/DEC) and the possibility to display the values in hexadecimal code on four 7-segment BCD cells. The values of the initial and final states can also be shown in hexadecimal code on the same display.

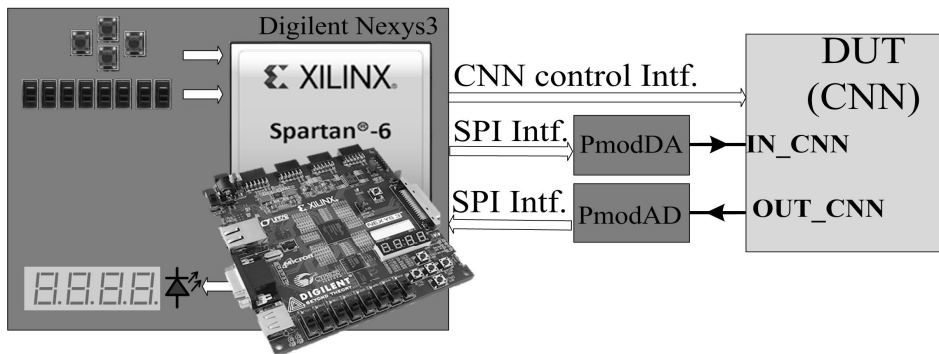


Fig. 2 – The interconnections of the components on the test circuit.

In Fig. 2 the implementation of the test circuit and its functions are shown. On this level the implementation is hardware for the control components, stimuli generation, response signal acquisition and user interface control.

The block diagram shown in Fig. 3 presents the way the signals are interconnected to the cabled structure. They can be identified as: CNN circuit,

PmodDA and PmodAD modules, FPGA circuit, integrated programmer, the CNN circuit control signals from the digital interface, the PmodAD and PmodDA modules control signals from the digital interface, the control signals of the display and peripherals integrated on the development board: oscillator at 100 MHz, 4 buttons, 8 switches, 1 LED, 1 display with four 7-segment BCD cells.

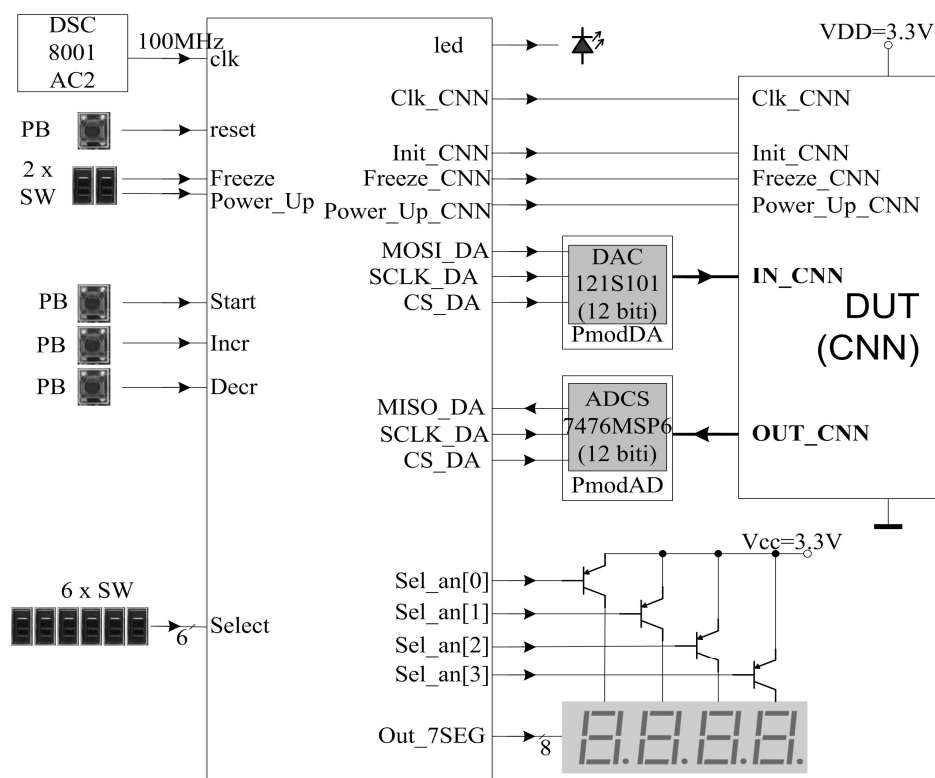


Fig. 3 – The test circuit diagram.

The system will operate using a clock signal of 100 MHz available on the development board, of which, by dividing, we obtain all the necessary clock signals.

To create stimuli we use 16 registers (one for every CNN cell) named: REG_IN_xx, the values of which, in the LOADx states, are applied, after the D/A conversion, to the CNN circuit as current cell state. The values for these registers will be ideally controlled individually by an increasing/decreasing circuit and will be shown on those 4 display cells. Through the computer interface the states of these registers will be ideally written and read respectively.

To save the response we use another 16 registers (one for every CNN

cell) named: REG_OUT_xx, of which, in the SAVEx state, the resulting values are saved after the A/D conversion, to a current cell state. The values of these registers will be shown on those four display cells. Through the computer interface the state of these registers will be ideally read.

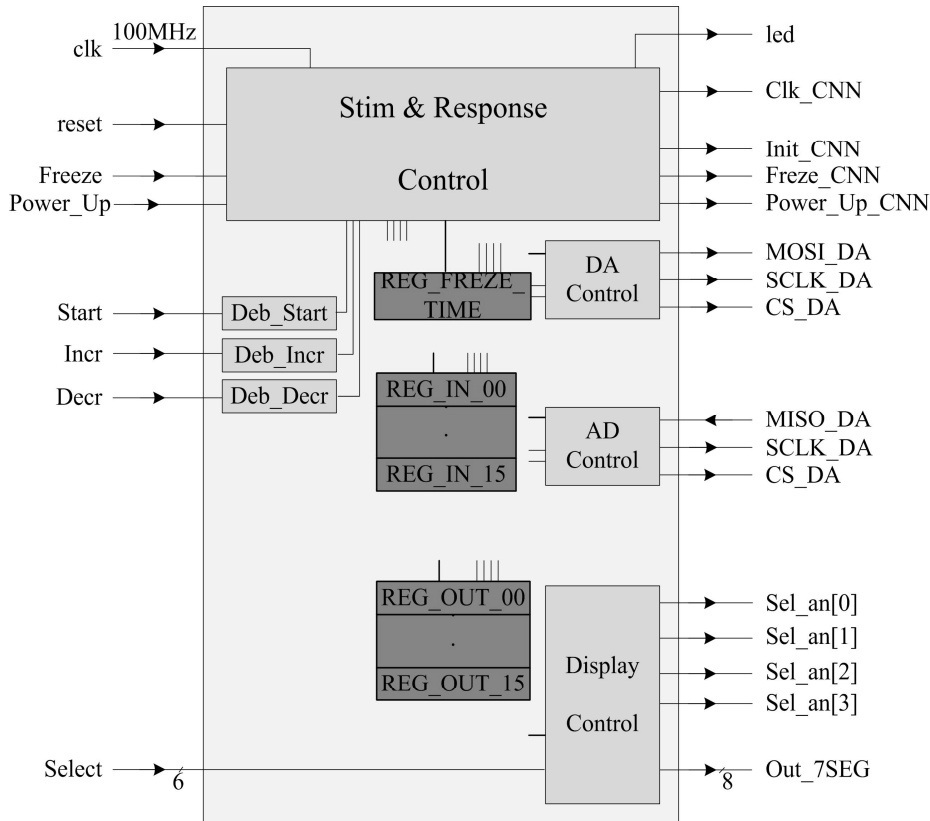


Fig. 4 – The components of the test circuit.

To control the operation of the CNN circuit, from the test circuit, control signals are generated. RUN3 state will have a duration that can be controlled by an user interface respectively by a computer interface. For this purpose a register called REG_FREEZE is used, which is similar to those described previously, with the possibility to increase/decrease, read/write and display. This register will save the duration of the RUN3 state, expressed as an integer number of periods of the clock signal for the block "Stim&Response Control".

In total there are 16 + 16 + 1 registers, but at a moment in time only a single register can be selected to increase/decrease and display. The command inputs for increasing/decreasing are passed through a suppression circuit and all the registers are multiplexed, but considering the internal logic of selection, operation is done only with the selected register. The registers for storing the

response have an increasing/decreasing function deactivated, allowing only display.

In 0 the block structure for the resulting test circuit after the detailed principal block diagram is shown.

3. Simulation and Measurement

For simulation it is necessary to put into the same testbench the test circuit with the model for a CNN circuit. In the testbench we connect the test circuit, the behavioral models for the A/D, D/A circuits respectively and the model for the digital interface of the CNN circuit, as shown in 0.

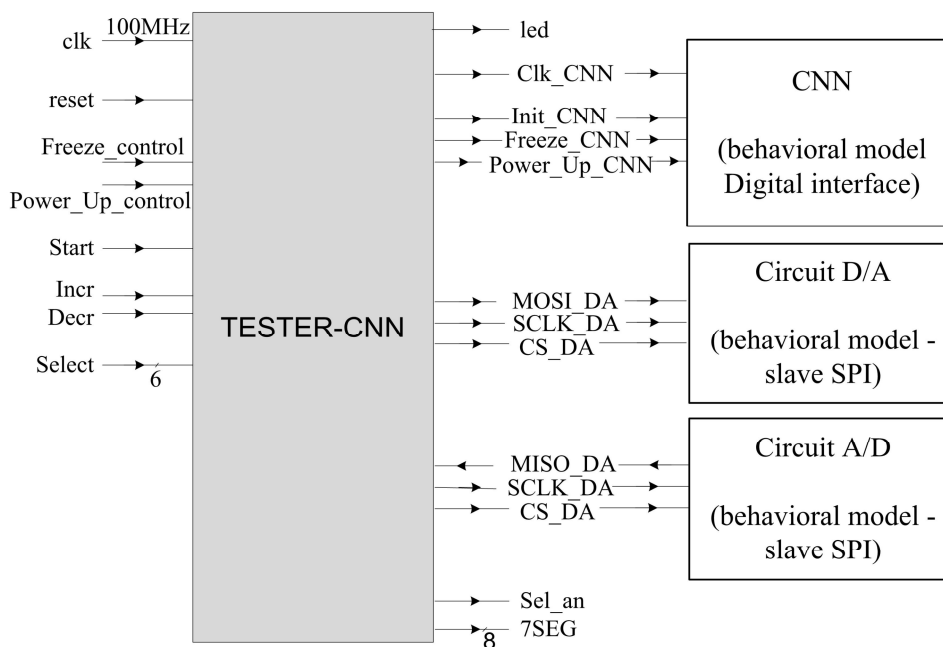


Fig. 5 – The test bench for simulation.

In 0 the signals obtained for the simulation of an entire test cycle of the CNN circuit are shown. Testing is triggered by the activation signal "start" and indicated by the value of the signal "led".

In the figure we can observe: the clock signal for the CNN, that is obtained by dividing the clock signal of the FPGA, the control signals of the CNN: INIT, POWER_UP and FREEZE, SPI transmission for the D/A converter at the cells' charging. The SPI transmission for the A/D converter at the cells' reading, FSM states, number of the cells.

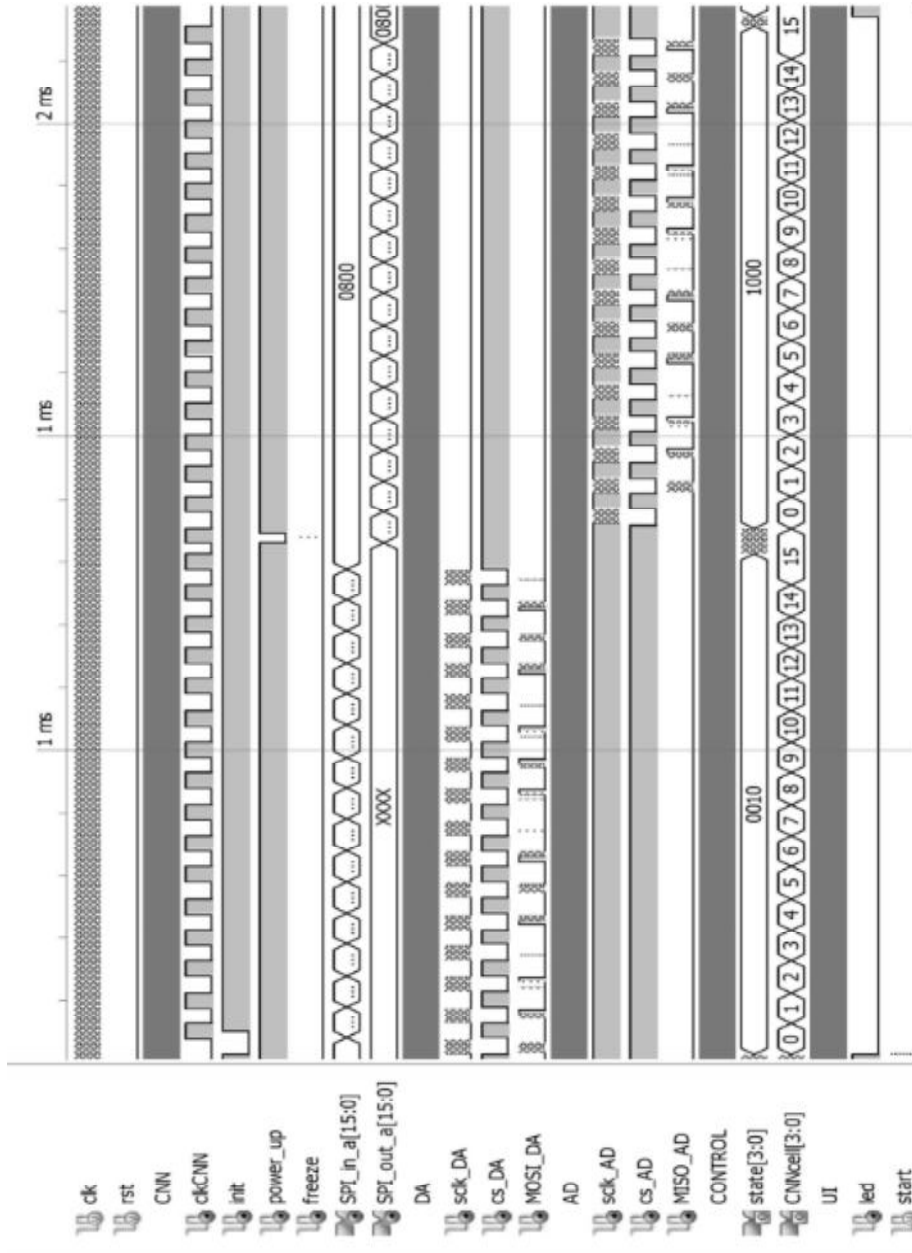


Fig. 6 – Simulation of the operation of the test circuit.

The detailed circuit control signals CNN (CLKCNN, INIT, POWER_UP and FREEZE) and the following internal states of the FSM are shown in 0.

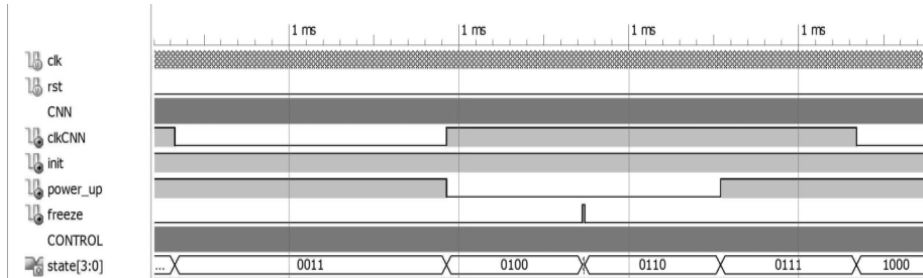


Fig. 7 – The simulation of the operation of the test circuit (RUNx detail).

The sequence of states for the "state" signal on 4 bits contains:

- a) RUN1 (0011) state - wait after the last cell is charged;
- b) Run2 (0100) state - power up the transconductors; POWER-UP signal in 0 logic;
- c) RUN3(0101) state – network evolution – the FREEZE signal is in 1 logic for the duration of a controlled time (in Fig. 7 it is the minimum value);
- d) RUN4 (0110) state – wait after the network evolution – the FREEZE signal in 0 logic;
- e) RUN5 (0111) state – power off the transconductors – the POWER_UP in 1 logic.

The results for the generated command signals for CNN are shown in 0.

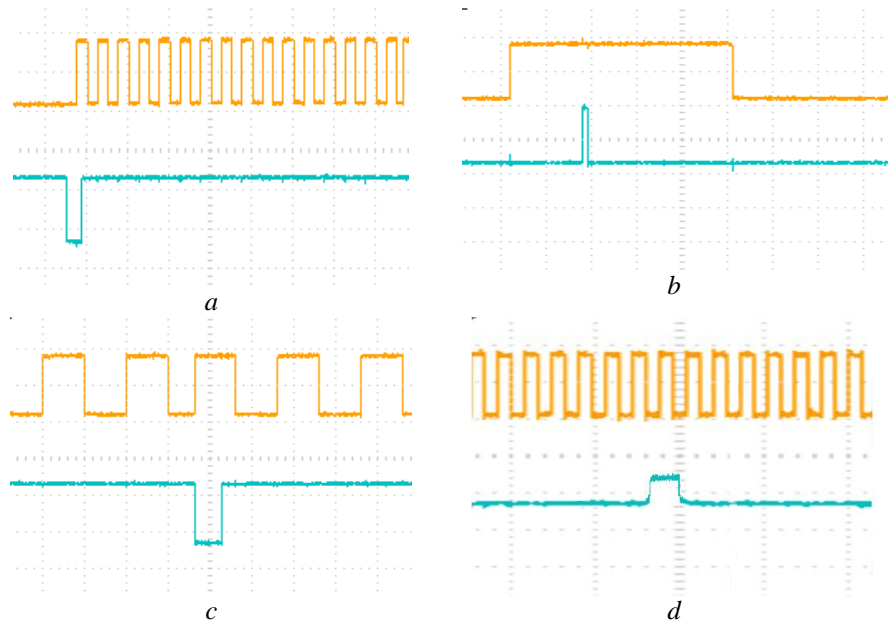


Fig. 8 – Generating INIT (a); FREEZE (b) POWER_UP (c) spatial impulse applied on an 8 cell (d).

The INIT signal, shown in 0 *a*, active on 0 logic, selects the first cell for charging the state and it must be active before the first negative edge of the CLK.

The FREEZE signal, shown in 0 *b*, active on 1 logic, can have a duration that can be modified by writing in the REG_FREEZE register.

The POWER_UP signal, shown in 0 *c*, active on 0 logic, powers on the transconductors and activates after writing the last cell or can be forced into active state by a switch on the development board. This signal is used to minimize consumption, the transconductors will not consume unless they are active.

The charging signal of the network is generated by writing those 16 input registers, REG_IN_xx. The value of a register is transferred to the D/A converter, that converts it to an analog signal and it is then applied to a cell.

Thus we can generate a pulse on the eighth cell with the value of 1.8 V (150mV with respect of the analog ground), like 0 *d*, charging the REG_IN_08 register with a binary value which is transformed by a 12-bit D/A converter to 1.8V. The rest of the registers will be charged with the binary value which will be transformed by a 12-bit D/A converter to 1.65 V (analog ground).

5. Conclusions

In this paper we have presented a way to generate a digital controlled signal for a 1D Cellular Neural Network with 16 cells and a charging signal for a network using a D/A converter with the use of a development board with FPGA,. Using an A/D converter, the network response is taken and stored in the registers created in FPGA.

REFERENCES

- Chu P., *FPGA Prototyping by Verilog Examples*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2008.
- Lai K., Leong P., *Implementation of Time-Multiplexed CNN Building Block Cell*. Proc. MicroNeuro, 1996.
- Muthuramalingam A. *et al.*, *Neural Network Implementation Using FPGA: Issues and Application*. Internat. J. of Information Technol., **4** (2008).
- Patache N. *et al.*, *CMOS Implementation of Spatio- Temporal Filters with Cellular Neural Network*. Bul. Inst. Politehnic, Iaşi, **LX (LXIV)**, 4, s. Electrot., Energ., Electron., 59-70 (2014).
- Rozkovec M. *et al.*, *Application Dependent FPGA Testing Method, Digital System Design: Architectures, Methods and Tools*. EUROMICRO Conf., 2010.
- * * Nexys3 Board Reference Manual.
- * * Adept I/O Expansion Reference Design.
- * * ADCS7476 datasheet.
- * * DAC121S101 datasheet.

METODĂ ȘI CIRCUIT PENTRU TESTAREA UNEI REȚELE NEURALE
CELULARE CU 16 CELULE

(Rezumat)

Se propune o modalitate de generare, cu ajutorul unui FPGA, a semnalelor digitale de control ale rețelei și a semnalului de încărcare a rețelei cu ajutorul unui convertor D/A. Printr-un convertor A/D se preia răspunsul rețelei și se stochează în registre realizate în FPGA. S-a realizat o simulare a circuitului de test și apoi măsurători ale semnalelor generate.

