

**Universitatea Tehnică “Gheorghe Asachi” din Iaşi**

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE**

**ELECTRONICĂ DIGITALĂ**

**Proiect**

**Tema: MODUL TIMER0 – v3**

Studenţi:

Gasca Valentin

Grupa : 1208B

Coordonator:

Asistent doctorand Ionica Pletea

**2022**

**Tema proiectului:**

**TIMER0 – v3**

**1. Specificaţiile proiectului:**

MODUL TIMER0 – v3

Să se implementeze în FPGA prin descriere în limbaj VHDL, utilizând programul VIVADO, modulul prezentat în figura 1 care este descris prin urmatoarele specificaţii:

a) registrul TMR0 are dimensiunea de 8 biţi

b) sursa de clock clk\_ex va fi generată de un buton extern

c) modulul „decodor BCD ” va fi implementat utilizad instructiunea VHDL „case”

d) iniţializarea registrului TMR0 de la butoanele externe se face pentru EN=’1’

e) valoarea registrului TMR0 se va afisa pe modulul display al plăcii BASYS 3

f) la depăşirea valorii maxime bitul „flag” TMRIF ia valoarea ‚1’

Implementarea proiectului va fi făcută printr-o descriere structurală



Fig. 1 schema bloc a modulului TIMER0\_v3

Fișierul bitstream rezultat în urma procesului de implementare va fi verificat utilizând placa de dezvoltare BASYS3

**2. Modulul TIMER0\_v3**

Modulul TIMER0\_v3 stocheaza o valoare care, după introducerea sa într-un convertor BCD, se va afișa pe 7 segment-display al plăcii Basys3. Fiecare tact de clock incrementează valoarea modulului, iar

**3. Metoda de implementare**

Utilizarea resurselor: circuit FPGA, limbajul VHDL, programul de sinteză Vivado

Modul de implementare : COMPORTAMENTAL

**4. Descrierea (scurtă) a sistemului de dezvoltare BASYS 3**

Placa de dezvoltare educațională BASYS 3 este o placă de dezvoltare făcută special pentru suita de sinteză VIVADO, folosind arhitectura Xilinx Artix-7. Această placă de dezvoltare este perfectă pentru educație, învățământ și pentru începători în general.

**Text

Description automatically generated5. Editarea fișierului VHDL**

**Fișierul VHDL (.vhd) este alcătuit dintr-o singură entitate numita** “**TMR0**”, cu declarația ei și arhitectura sa. Porturile entității sunt:

1. **Clock\_100Mhz care reprezintă clockul plăcii Basys3**
2. **CLK\_EX care reprezintă un clock extern**
3. **EN flagul necesar introducerii valorii predefinite în registru**
4. **RESET butonul cu ajutorul căruia setam valoare registrului pe 0**
5. **SWT reprezintă switchurile cu ajutorul cărora introducem valoarea in registru**
6. **Anode\_Activate reprezintă display-urile pe care se vor afișa numerele**
7. **LED\_OUT reprezintă codificarea a 7 segment display a plăcii Basys3.**

**Arhitectura entității are la rândul său mai multe semnale, acestea fiind:**

1. **TIMER , reprezentând modulul studiat**
2. **TMRIF, reprezentând flagul de depășire a valorii maxime**
3. **LED\_activating\_counter, reprezentând cele 4 display-uri**
4. **LED\_BCD, reprezentând numărul afișat pe display**
5. **Displayed\_number, reprezentând toate numerele ce urmează a fi afișate**
6. **Refresh\_counter, reprezentând viteza de reîmprospătare a informațiilor ce trebuie afișate**

**Pentru a funcționa programul, am avut nevoie de următoarele procese:**

**Table

Description automatically generated**

**Acest proces reprezintă decodorul BCD. Datele de intrare fiind în registrul LED\_BCD, iar rezultatul afișat se află în registrul LED\_out.**

**Text

Description automatically generated with medium confidence**

**Aceste 2 procese ( process(clock\_100Mhz,reset) și process(LED\_activating\_counter)) reprezintă metoda necesară afișării cifrelor diferite pe fiecare dintre cele 4 display-uri ale plăcii Basys3.**

**Text

Description automatically generated**

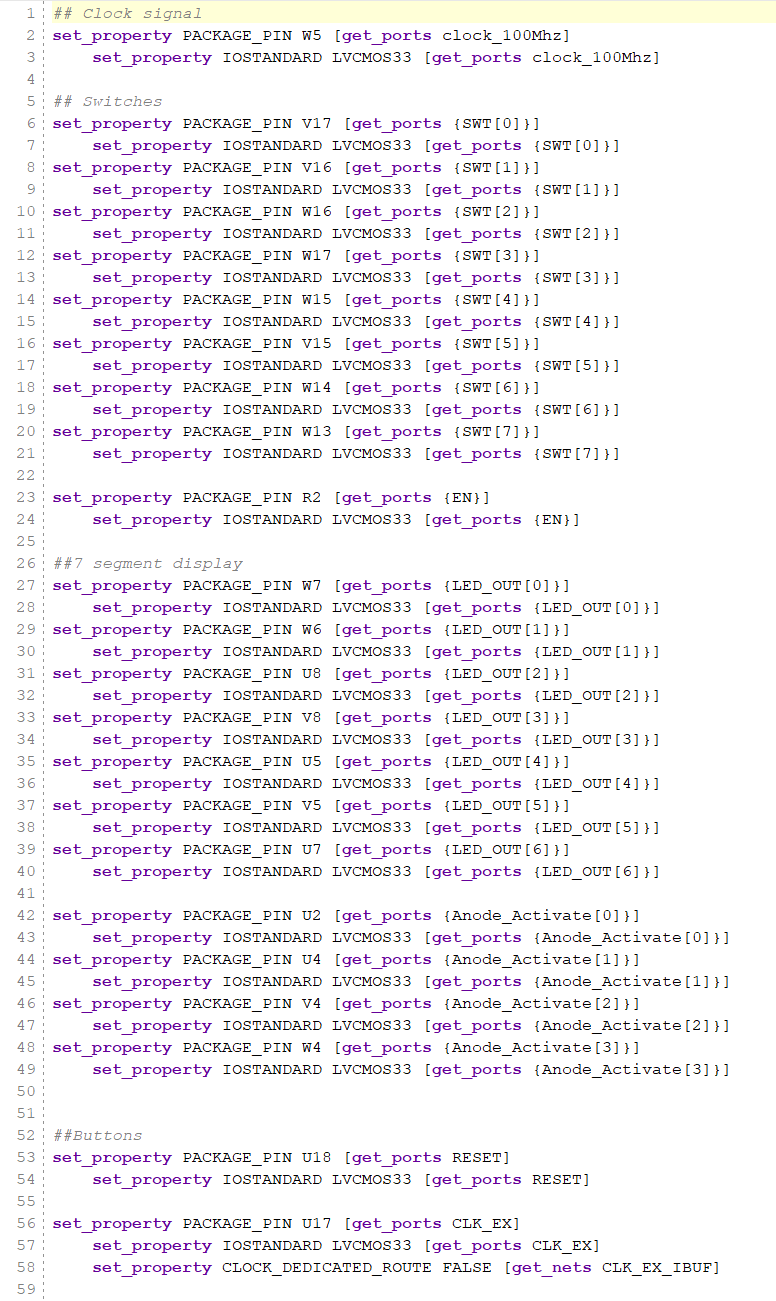
**Acest proces ne introduce valoarea corespunzătoare în registrul studiat și/sau o resetează.**

**Text

Description automatically generated**

**Acest proces ne face conversia din un număr binar pe 8 biți într-un număr BCD**

**6. Editarea fişierului de constrângeri**

****

**Fișierul de constrângeri este editat în următorul mod:**

* **PACKAGE\_PIN W5 – clockul plăcii**
* **PACKAGE\_PIN V17, V16, W16, W17, W15, V15, W14, W14 – switchurile cu ajutorul cărora introducem valoarea prestabilită în registru.**
* **PACKAGE\_PIN R2 – flagul EN necesar funcționării switchurilor**
* **PACKAGE\_PIN W7, W6, U8, V8, U5, V5, U7 – cele 7 segmente ale display-ului**
* **PACKAGE\_PIN U2, U4, V4, W4 – cele 4 displayuri ale plăcii**
* **PACKAGE\_PIN U18 - butonul de reset al programului**
* **PACKAGE\_PIN U17 – clockul extern.**

**7. Descrierea paşilor de sinteză şi testarea circuitului rezultat**

Graphical user interface, text, application, chat or text message

Description automatically generated

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**8. Concluzii**

**Implementarea acestui modul a fost o experiență utilă care ne va ajuta în viitor atunci când vom aplica pentru un job în domeniu.**

Bibliografie:

1. PIC16LF1937, datasheet, <http://ww1.microchip.com/downloads/en/DeviceDoc/41364E.pdf> (pag. 191)

2. VHDL Reference Manual, <http://www.ics.uci.edu/~jmoorkan/vhdlref/Synario%20VHDL%20Manual.pdf>

3. BASYS 3 Reference Manual, <https://reference.digilentinc.com/reference/programmable-logic/basys-3/reference-manual>

4. <http://www.microcontroller.it/tutorials/PIC/Peripheals/Timer0/timer0_3.htm>

5. <https://www.fpga4student.com/2017/09/vhdl-code-for-seven-segment-display.html>

6. <https://github.com/bilalkabas/Basys3-VHDL-Basics>

7. <https://allaboutfpga.com/bcd-to-7-segment-decoder-vhdl-code/>

8. [https://www.youtube.com/watch?v=gUiX8aMV5oE](https://www.youtube.com/watch?v=gUiX8aMV5oE&ab_channel=IBElectronicsWorld)

9. [https://www.youtube.com/watch?v=VKKGyOc4zRA&t=288s](https://www.youtube.com/watch?v=VKKGyOc4zRA&t=288s&ab_channel=LBEbooks)