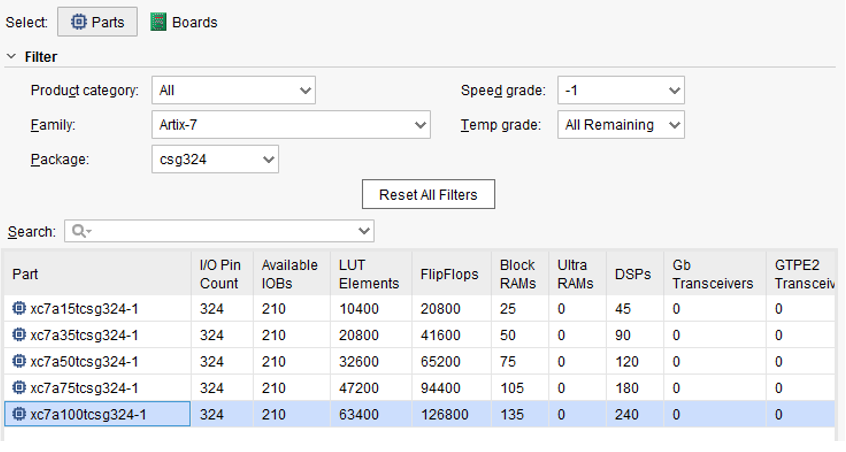
Referat 4: Microblaze

Digori Gheorghe 331CC

Neciu Laurentiu-Florin 331CC

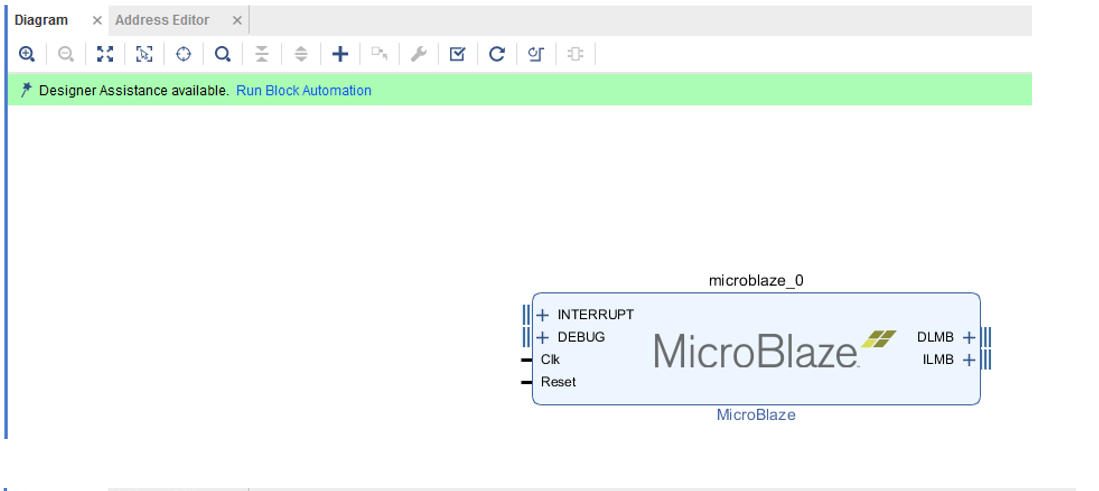
Scopul: Implementarea unui program scris in Limbajul C pe o placa FPGA folosing modulul Microblaze.

Initial creem un proiect care prezinta specifiicatiile din poza 1:



Imediat dupa selectam create block design – se deschide o fereastra “diagram”. In cadrul acelei ferestre se apasa butunul “+”si se selecteaza Microblaze.

Va apărea componenta în diagramă. Ulterior, selectăm “Run block automation”. Setam memoria locala la 32KB, Debug la “Debug & UART” și Axi port la “Enabled”, după care apasăm pe OK.Acum, se poat vizualiza toată schema.



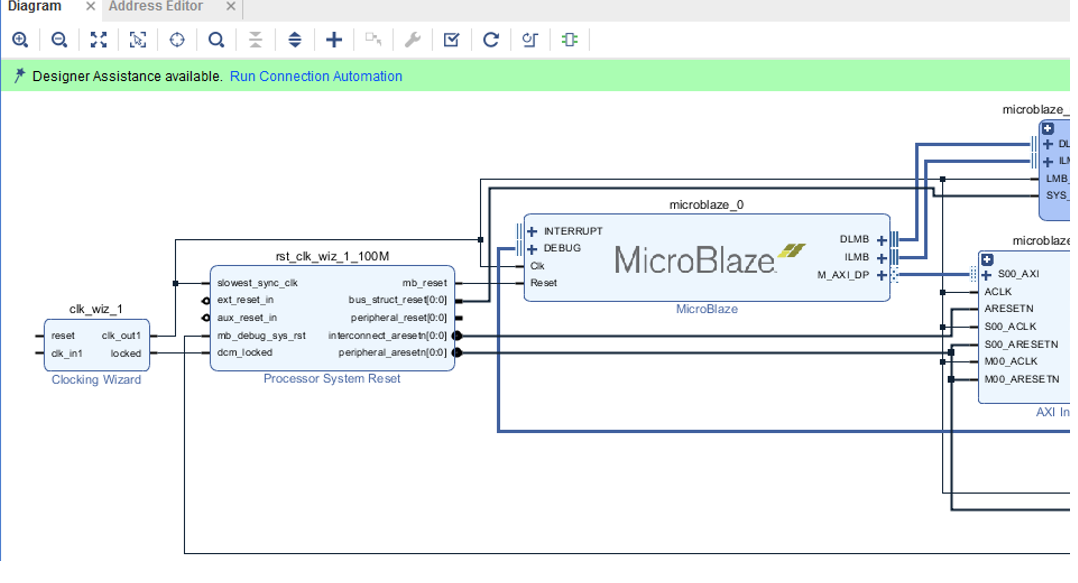
Putem adauga anumite secțiuni folosind butoanele “+” din diagramă.

Se executa click drepata pe spațiul gol al

diagramei și dăm “regenerate layout”.

Facem dublu click în diagramă pe componenta

clocking wizard. În “input clock information” la primary -> source, se schimbă în Single ended non clock pin.



Putem adauga anumite secțiuni folosind butoanele “+” din diagramă.

Se executa click drepata pe spațiul gol al diagramei și dăm “regenerate layout”.

Dublu click în cadrul diagramei pe component clocking wizard. În “input clock information” la primary -> source, schmbam în Single ended non clock pin.

Rulăm “Run clock automation”. Se bifeaza CLK\_in1 și apasăm OK.

Apăsăm din nou pe run connection Automation, și setăm reset-ul pe active\_high și apăsăm ok. Setăm ext\_reset\_in pe active\_low și apăsăm ok.

Salvăm.

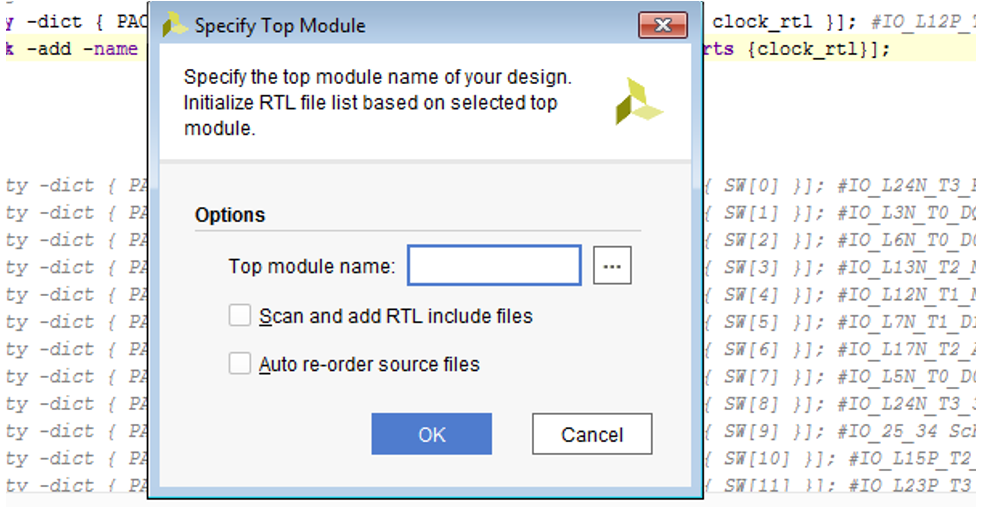
Se descarca fișierul stimuli.xdc din folderul Lab\_microblaze de pe owncloud-ul de pe cn.curs.pub.ro și îl adaugam în proiect.

Sources > Constraints > add source > add or create constraints > add file > alegem .xdc > finish

Deschidem fișierul și decomentăm primele doua linii de la “Clock”. Se inlocuieste “CLK100MHZ” cu “clock\_rtl”. Mai decomentăm: Switch > J15 > reset\_rtl Button > N17 > reset\_rtl\_0

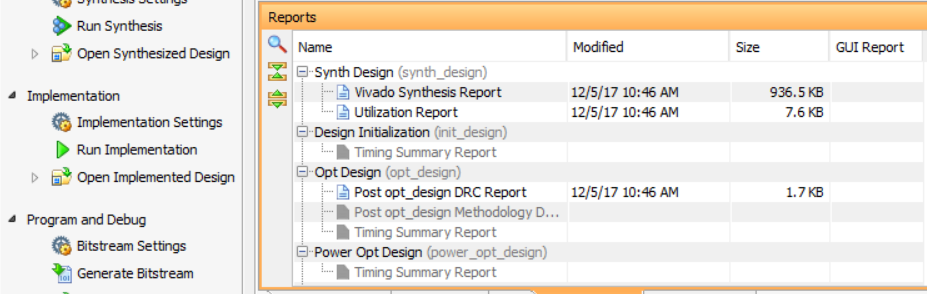
Salvăm.

Rulăm “Run Synthesis”, și introducem numele modulului Top (design\_1). După ce termină se selecteaza view reports.



Dupa, ne ducem la Synthesis > Open Elaborated Design.

Mergem în colțul din sus-drepata, Default Layout > IO Planning > FPGA Intrăm în TCL Console și verificăm dacă toate porturile reset au opțiunea Fixed bifată.

După, se selecteaza “Generate bitstream”. Selectăm “Open Implemented design” și apăsăm pe Yes. 

Din sources, selectăm design\_1, Facem click dreapta si alegem “Creade HDL wrapper”. Selectăm design\_1, click dreapta și generate output products.

File > Export > Export hardware (și includem bitsteam) > Ok.

Va da o eroare. Pentru remediere avem în owncloud un fișier care conține comenzile necesare pentru a remedia eroarea.

“set PROJECT\_NAME "lab1"

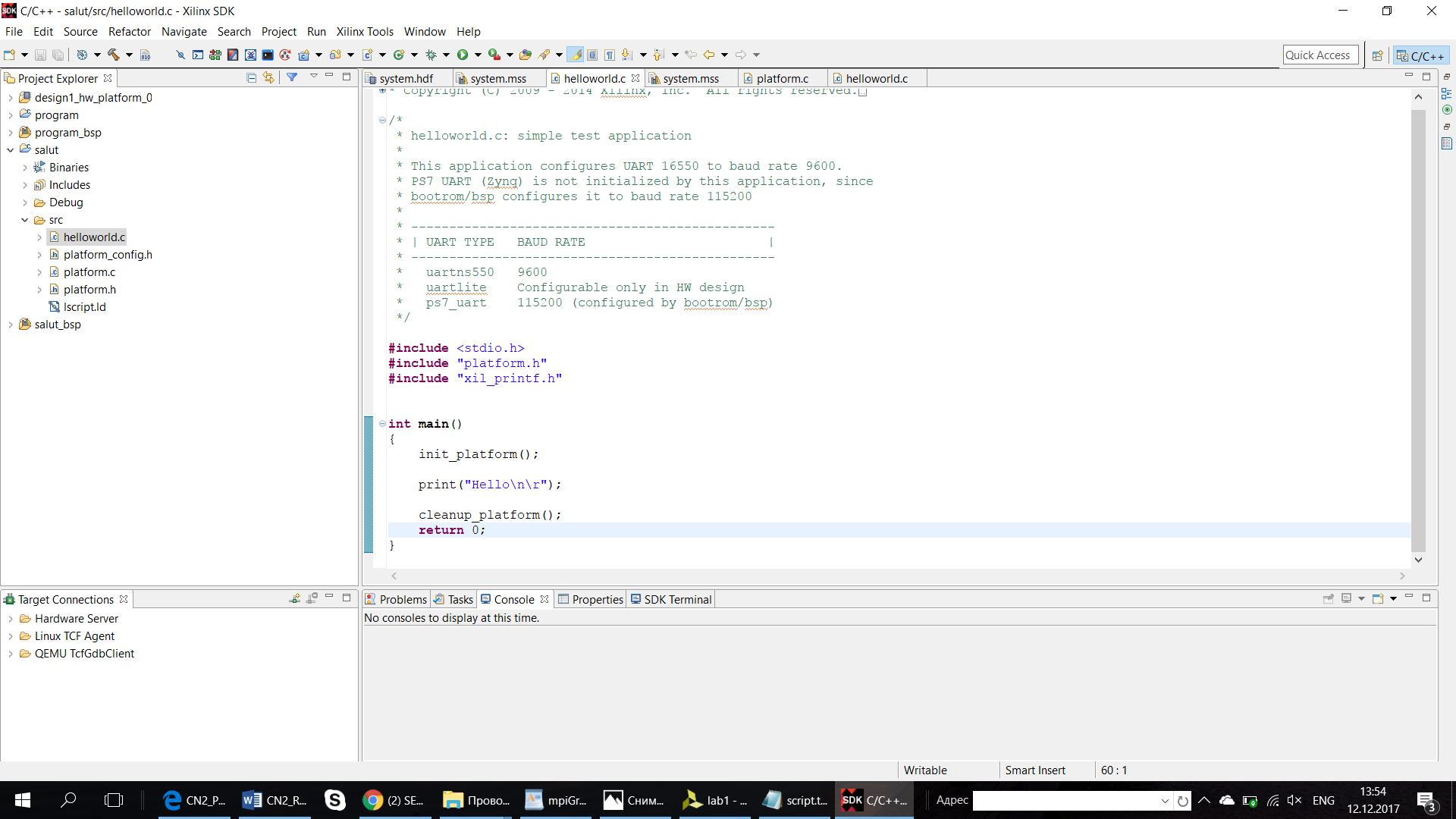
set TOPLEVEL\_NAME "design1"

write\_hwdef -force -file F:/Anul\ 3\ laboratore\ si\ ursuri/CN2/lab1/lab1.runs/$TOPLEVEL\_NAME.hwdef

write\_sysdef -force -hwdef F:/Anul\ 3\ laboratore\ si\cursuri/CN2/lab1/lab1.runs/synth\_1/$TOPLEVEL\_NAME.hwdef -bitfile F:/Anul\ 3\ laboratore\ si\ cursuri/CN2/lab1/lab1.runs/impl\_1/$TOPLEVEL\_NAME.bit

“

File > Launch SDK. File > New > Application Project > helloworld > next & finish. Helloworld > src > helloworld.c Run > Run configurations > Xilnix C++ app GDB. New > Bsfile > select design1.bit Application > browse PR name (helloworld) STDIO Connection > Connect to stdio > PORT: JTAG (UART) Run



Pentru a genera mesajul “Helloworld” pe ecran, se tine apasat butonul N17.