

# UNIVERSITATEA TEHNICĂ "GH. ASACHI" IAȘI

Facultatea de **Electronică, Telecomunicații și Tehnologia Informației**

Domeniul: **Inginerie Electronică și Telecomunicații**

Specializarea: **Tehnici Moderne de Prelucrare a Semnalelor**

Forma de învățământ: **MASTER / ZI** Anul de studii: **1** Anul universitar: **2009-2010**

## P R O G R A M A A N A L I T I C Ă

a disciplinei: **Verificarea funcționala a circuitelor integrate digitale**

- 1. Titularul disciplinei:** Sef. Lucrari dr. ing. **Tiberiu-Dinu TEODORESCU**  
**2. Tipul disciplinei:** **DI, DS** codul: **602\_TMPS**  
**3. Structura disciplinei:**

Semestrul	Numărul de ore pe săptămână				Forma de evaluare finală	Numărul de ore pe semestru				
	C	S	L	P		C	S	L	P	Total
<b>3</b>	<b>2</b>	<b>-</b>	<b>2</b>	<b>-</b>	<b>E</b>	<b>28</b>	<b>-</b>	<b>28</b>	<b>-</b>	<b>56</b>

#### 4. Obiectivele cursului:

Disciplina de „Verificarea funcționala a circuitelor integrate digitale” urmărește introducerea unor noțiuni legate de verificarea funcționala a circuitelor integrate digitale. De asemenea, se dorește asimilarea notiunilor de proiectare a circuitelor integrate digitale pornind de la infrastructuri de verificare. Se prezintă notiuni legate de componentele unui mediu software de verificare și a modului de interacțiune al acestora, precum și implementarea acestora într-un mod reutilizabil inter-proiecte.

La sfârșitul cursului studenții vor fi capabili să proiecteze un mediu de verificare pornind de la specificațiile tehnice ale proiectului, folosind un limbaj OOP (System Verilog, Vera sau C++), se dezvoltă un plan de testare și să verifice gradul de verificare al circuitului la nivel de model funcțional prin intermediul măsurilor specifice: acoperire de cod, funcționala, de comutare.

Se va prezenta pe larg metodologia de verificare funcționala bazată pe tranzacții folosind parametrii aleatori, precum și metodologia de teste direcționate, în vederea acoperirii cazurilor de graniță.

#### 5. Concordanța între obiectivele disciplinei și planul de învățământ:

Disciplina este una de specialitate, având nevoie pentru buna desfășurare de cunoștințe de teoria sistemelor, procesare numerică de semnal, circuite integrate digitale, programare în C/C++, VLSI digital. Scopul ei este de a pune în evidență și a exemplifica practic structuri reutilizabile pentru verificarea circuitelor integrate digitale, de a dezvolta în cursul proiectelor IP-uri de verificare într-un mod similar celui prezent în proiecte mari la nivel mondial.

#### 6. Rezultatele învățării exprimate în competențe cognitive, tehnice sau profesionale

La sfârșitul acestui curs, studenții vor fi capabili să abordeze un proiect ca ingineri de verificare funcționala de hardware, capătând practic toate abilitățile dezvoltării profesionale în această direcție.

## 7. Proceduri folosite la predarea disciplinei:

- predarea cursului cu ajutorul videoproietorului
- conceptele prezentate sunt insotite de exemple ce au legatura cu tehnologiile actuale folosite de liderii de piata.
- implementarea practica a conceptelor prezentate la curs in cadrul laboratorului, practic exista o suprapunere de circa 80% intre notiunile prezentate la curs si implementarea lor practica
- metode de invatare centrate pe student: teme de casa, dezvoltarea unui exemplu minimal in cadrul cursului;
- strategii de actualizare a predării conform programului de studiu, caracteristicilor studenților, formei de învățământ și criteriilor de calitate adoptate: tema de casa constituie un prilej de a rezolva o problema individual si de a o integra cu solutiile colegilor, proiectele constand in proiectarea unor arhitecturi in grupe de 2-3 studenti, fiecare avand o tema clar definita. Aceasta strategie va consolida spiritul de echipa si va crea cultura lucrului folosind o baza de date comuna, pe baza de versiuni.

## 8. Sistemul de evaluare:

### *Evaluarea continuă:*

*Activitatea* la seminar / laborator / proiect / practică

Ponderea în nota finală: 60% - CC

Realizarea aplicatiilor constituie un prilej de a rezolva o problema individual si de a putea intelege functionarea sistemul pina la gradul de detaliere cerut.

### *Evaluarea finală:* (Se precizează: examen sau colocviu.)

Ponderea în nota finală: 40% - T

Probele: Teza scrisa: test de cunostinte cu intrebari inchise, fara documentatie sau proba orala – prezentarea unui proiect in fata colegilor: 40%

## 9. Conținutul disciplinei:

### a) Curs

#### **Cap. 1. Prezentarea principalelor componente ale unui mediu de verificare 5 ore**

- 1.1. Generatori
- 1.2 Tranzactori (Drivere, BFM)
- 1.3 Monitoare si verificatoare de protocol
- 1.4 Tabele de verificare de date (scoreboards)

#### **Cap 2. Studiu de caz pentru un mediu de verificare freeware – Teal si Truss 5 ore**

- 2.1. Prezentarea lantului de unelte implicate in mediul de verificare cu precizarea rolului functional al fiecaruia
- 2.2. Componente specifice ale Teal
- 2.3. Testarea multithreading
- 2.4. Reutilizarea componentelor SW utilizate in verificare. Truss
- 2.5. Proiectarea testarii automate in regresii cu ajutorul limbajelor de scripting

#### **Cap 3. System Verilog 10 ore**

- 3.1. Elemente de limbaj, modul, program, interfețe.
- 3.2. Mostenire și polimorfism
- 3.3. Esantionare, control stimuli, sincronizare
- 3.4. Structuri de date utilizate în System Verilog
- 3.5. Așertiuni combinate și cu stare

**Cap. 4. Metodologia testării folosind System Verilog – OVM/VMM**

**8 ore**

- 4.1. Clase de bază în OVM/VMM
- 4.2. Randomizare constransă. Stabilitate în randomizare
- 4.3. Controlul proceselor concurente
- 4.4. Mecanisme de raport
- 4.5. Studiu de caz ce implică folosirea interfețelor virtuale
- 4.6. Acoperire, grupuri de acoperire (coverage)
- 4.7. Studiu de caz ce implică folosirea tuturor claselor de bază ale OVM/VMM

Total 28 ore

**b) Aplicații**

**Sedințele de laborator (2h/sedință):**

- 1. Prezentarea mediului Teal cu exemple
- 2. Verificarea unui DUT de tip counter cu reset sincron și încărcare folosind Teal
- 3. Verificarea unui DUT de tip coadă sincronă folosind Teal și monitor implementat într-un thread separat
- 4. Verificarea unui DUT de tip coadă sincronă folosind Teal, verificator folosind model de referință – golden model și monitor implementate în thread-uri separate
- 5. Verificarea unui DUT de tip stivă asincronă folosind Teal și monitor implementat într-un thread separat
- 6. Verificarea unui DUT de tip stivă sincronă folosind Teal, verificator și monitor implementate în thread-uri separate
- 7. Studiul unor exemple de verificare prezentate la curs
- 8. Implementarea mediului pentru verificarea unui switch în System Verilog cu DUT implementat în același limbaj ca model funcțional
- 9. Rafinarea exemplului precedent prin realizarea DUT într-un mod sintetizabil
- 10. Testarea unui controller de memorie folosind System Verilog și OVM/VMM - I
- 11. Testarea unui controller de memorie folosind System Verilog și OVM/VMM - II
- 12. Testarea unui arbitru folosind System Verilog și OVM/VMM - I
- 13. Testarea unui arbitru folosind System Verilog și OVM/VMM - II
- 14. Întrebări– răspunsuri cu exemplificări

Total 28 ore

**10. Bibliografie selectivă**

- 1. David A. Patterson, John L. Hennessy, Computer Organization & Design, Morgan Kaufmann, San Francisco, California, USA, 1997, ISBN 1-55860-428-6 (cloth)-ISBN 1-55860-491-X(paper)
- 2. Keshab K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation, John Wiley & Sons, NY, USA, 1999, ISBN 0-471-24186-5
- 3. Mike Mintz, Teal and Truss documentation, resursa online: [www.trusster.com](http://www.trusster.com)

4. Janick Bergeron, Eduard Cerny, Alan Hunter, Andrew Nightingale - Verification Methodology  
Manual for System Verilog., Springer, 2006, ISBN-13: 978-0-387-25538-5

Data: 01.09.2008

Titular curs:

s.l. dr. ing. Teodorescu Tiberiu-Dinu

Titular aplicații:

s.l. dr. ing. Teodorescu Tiberiu-Dinu

**Semnături:**