



## קורס:

### הכשרת מהנדסי וריפיקציה (250 שעות)

תעשיית השבבים הגלובלית צומחת בשנים האחרונות וחברות בינלאומיות רבות מגדילות את ההשקעה בתכנון, פיתוח וייצור שבבים. הביקוש למהנדסי שבבים גובר אך ישנו מחסור של אלפי עובדים בשוק הישראלי והתחרות מצד החברות על מועמדים עולה. הכשרה זו נועדה להקנות למשתתפים ידע וכלים הכרחיים לעבודה בתחום. ההכשרה תערך בשיתוף חברת **Cadence** ותכלול שימוש בכלים המובילים של החברה.

## למי מתאים הקורס?

בוגרי תארים בהנדסה, מדעי המחשב, מדעים מדויקים וטכנולוגיים. הנדסאים של כל התחומים, בוגרי י"ג + י"ד באלקטרוניקה או ניסיון צבאי רלוונטי.

## תנאי קבלה:

- מבחן מקצועי וראיון אישי עם יועצת לימודים.
- תואר רלוונטי.
- שליטה בשפה אנגלית.

SQLabs מבית SQLink הינה המכללה הטכנולוגית המובילה למקצועות הפיתוח, אבטחת מידע, מקצועות הנתונים והמידע ועוד.

אנו מציעים מגוון רחב של קורסים בטכנולוגיות ובנושאים המתקדמים והדרושים ביותר בתעשייה אשר מועברים על ידי מיטב המרצים בשוק.

צוות ההדרכה שלנו מורכב ממרצים אשר עובדים בתעשיית ההיטק ומביאים את ניסיונם העשיר בתחום לקורס.

## למה ללמוד אצלנו?

- ✓ סגל מרצים מקצועי אשר מגיע מתעשיית ההיטק ומביא את ניסיונו העשיר לכיתה.
- ✓ תיק עבודות מפורט המותאם לדרישות השוק.
- ✓ קורס אנגלית עסקית - בהכשרה שלנו תקבלו את כל הידע שתצטרכו בשפה האנגלית.
- ✓ קורס הכנה לראיונות עבודה הכולל סדנת קורות חיים וכתובת פרופיל לינקדאין.
- ✓ יחס אישי וסביבה תומכת לכל תלמיד, החל מתהליך הרישום, סביבת המחשוב, חומרי העזר ועוד.
- ✓ הקורסים מוצעים במחירים אטרקטיביים המאפשרים לכל אחד ואחד לשרג את היכולות המקצועיות שלו וזאת מבלי להתפשר על איכות ההדרכה.
- ✓ לימודים במודל היברידי – הקורס מתבצע בזום + פרונטלי לבחירתכם! תוכלו לצפות בלייב גם בשיעורים הפרונטליים.

## תוכנית הלימודים:

### מכינה

#### מבוא למערכות ספרתיות

- אלגברת מיתוג בסיסית.
- פונקציות מיתוג.
- מינימיזציה של פונקציות מיתוג.
- מודל ספרתי ושערים לוגיים.
- יחידות זיכרון.
- מערכות עקיבה. Moore/Mealy – (Sequential systems)
- תכנון מערכות עקיבה.

#### מבוא ל-VLSI

- התפתחות טכנולוגיות פיתוח השבבים, חוק מור, שלבים בפיתוח שבבים
- בחינת השבב כמכלול – המערכות שבו, רכיבי המערכות
- ות, תתי מערכות.
- תכנון השבב - הגדרות, דרישות, ארכיטקטורה (Top-down, bottom-up).
- עקרונות פעולה של טרנזיסטורי CMOS, בניית שערים לוגיים בעזרת טרנזיסטורים

- תהליך יצור השבב (Fabrication)
- LAYOUT + design rules
- מאפיינים חשמליים של טרנזיסטורים, שיקולי תכנון – הספק, תדר, שטח, טכנולוגיה, מחיר.
- מתודולוגיות תזמון, ניתוח זמנים ושיקולי תזמון
- מתודולוגיות שעון

### **Introduction & Basics of Chip design flow**

- Introduction to VLSI & technology
- ASIC basic concepts and usage models
- Gates, memories, analog and mixed signal circuits

### **Unix**

- Unix basics
- Shell commands
- Edit tools (gvim / nedit / xemacs)

### **Verilog basics - including labs**

- Introduction to RTL language
- Module definition, instantiation and parameters
- Data types
- Data flow - continuous & non continuous assignments
- Procedural blocks (process / initial)
- Flow control statements (if-else, case, loops, assign, @, #, \$)
- Operators
- State machines
- Simulation concepts & basics (display, files, functions & tasks, PLI)

### **Introduction to verification and basic concepts**

- What is Design Verification
- Verification Flow:
  - Product Development Process
  - Design verification process
  - Verification structure & concepts
  - Simulation flow

- Verification approaches – Black Box, White Box, Gray Box
- Verification techniques
  - Functional & code coverage
  - Assertions
  - Formal verification
  - Gate level
  - Emulation technologies
- Verification abstraction levels
  - Block level
  - Chip level
  - System level
- High level verification languages and methodologies background

### **Verification tools introduction and usage**

- High level overview on verification tools
- Xcelium usage and commands including Simvision (Indago)
- vManager/ IMC – Coverage and usage

### **SystemVerilog for Verification - including labs**

- Module 1: Course Overview (Verilog vs System Verilog)
- Module 2: Concepts and Guidelines
- Module 3: Syntax and Data Types
- Module 4: Procedural Blocks and Statements
- Module 5: Programming Statements
- Module 6: Operators and Operation Rules
- Module 7: Arrays, Structures, and Packages
- Module 8: Interfaces and Ports
- Module 9: Verification Constructs
- Module 10: Basic Object Oriented Programming
- Module 11: Advanced Object Oriented Programming
- Module 12: Scoreboards
- Module 13: Constrained Random Generation
- Module 14: Functional Coverage
- Module 15: Threads and Interprocess Communication

## **Essential System-Verilog for UVM (including labs)**

- Basic Classes and Randomization
- Overview of DUT
- Static Properties and Methods
- Inheritance and Polymorphism
- Aggregate Classes
- Components
- Connection to a DUT
- Building a Verification Component

## **UVM - including labs**

- Module 1: Course Overview
- Module 2: First Look At SystemVerilog UVM (UVM Evolution, OOP,...)
- Module 3: Transactions and Sequences
- Module 4: Sequencers, Drivers, and TLM
- Module 5: Monitors and Agents
- Module 6: Coverage Collectors
- Module 7: Scoreboards and Environments
- Module 8: Configuration and Factory
- Module 9: Tests and Virtual Sequences
- Module 10: Sequences for Complex Stimulus
- Module 11: UVM Register Layer Overview
- Module 12: UVM Summary

## **Formal verification essentials including SVA**

## **Final project (3 Weeks)**