UNIVERSAL VERIFICATION METHODOLOGY (UVM) INTRODUCTION TRAINING

1 Introduction

This document outlines courses in “Universal Verification Methodology (UVM) Introduction”. The training provider (Alpinum Training, see section 3) and the trainer (Dr. Mike Bartley, see section 5) have delivered this course and similar in design verification with the main objective of ensuring participants have the skills and knowledge to be more productive on projects immediately after the course, with tips (gained from 30+ years in verification) for increasing productivity.

2 Contact person

Mike Bartley, mike@alpinumconsulting.com, mobile +44 7796 307958

3 Training provider

The training would be delivered by Dr. Mike Bartley, CEO of the commercial training provider ALPINUM TRAINING LTD (registered company number 14579405 at address 26 Queensway, London, London, England, W2 3RX).

The course materials used by Alpinum Training are based on training expertise gained by Mike Bartley since he began his teaching/training career in 1983. The specific content of this course are based on Design Verification (DV) methodology, simulation-based DV and formal verification training courses developed and delivered internally for colleagues by Mike since 1998.

In 2008 Mike developed his first public training course on Design Verification that included both HDL and Specman/OVM content. That material has been developed over the years as new DV languages and methodologies (SV, SVA and UVM) have been adopted by the industry in the subsequent years. During that time the various courses have been delivered 40+ times to 500+ students. Student feedback is captured on every delivery and improvements incorporated into the materials and delivery.

All the courses are now owned and delivered by Mike Bartley through Alpinum Training.

4 The course prerequisites, objectives, structure and content

4.1 Course prerequisites

It is assumed that this training will be delivered to an industry audience of engineers with prior experience in and good working knowledge of Verilog and SystemVerilog who are engaged in the verification of digital integrated circuits.

4.2 Course objectives

By the end of the course, participants should be able to:

1. Understand and use main UVM features to build test benches to verify IP designs only (details of the UVM features covered are detailed in section
2. Create, configure and customize reusable, scalable, and robust UVM Verification Components (UVCs)
3. Combine multiple UVCs into a complete verification environment
4. Integrate scoreboards, multichannel sequencers and Register Models

5. Efficient/effective use of DV tools for UVM (e.g. simulators, static checkers, and debug tools) to increase productivity on real projects.
6. UVM reuse to improve DV productivity
7. Give insights into how the use of AI can be used to enhance UVM productivity.
8. Ensure students have sufficient UVM knowledge, skills and practical experience for learning advanced UVM.

The instructor will attempt to tailor all the above according to the specific IP, methodologies and processes of the participants (where time and confidentiality allows).

Note that topics can be moved between the UVM introduction and advanced UVM courses if requested by the reviewers

4.3 Course delivery

4.3.1 Main course delivery format

There are 2 pre-defined options:

- Online
    - The course will be delivered through a live online delivery format (using Teams or similar if preferred). The timing can be adapted to suit delegate preferences. The following are 2 suggestions for the timing of the live online lectures:
        - 4 hours per day (2 hours in the morning and 2 hours in the afternoon) over 4 continuous days duration within a single working week (Monday to Thursday preferred, Tuesday to Friday possible)
        - 2 hours per day (2 hours in the morning or 2 hours in the afternoon) over 2 weeks for 4 continuous days duration within each of those weeks (Monday to Thursday preferred, Tuesday to Friday possible)
    - The course delegates will be expected to perform self-study work on quizzes, examples and exercises outside of the above hours such that the 1 week course should be regarded as full-time and the 2 week course should be regarded as 50% part-time. Online support will be available to help course delegates in their self-study work.
- Physical onsite[1] face-face
    - The course will be face-face in a classroom environment:
        - 4 hours per day (2 hours in the morning and 2 hours in the afternoon) over 4 continuous days duration within a single working week (Monday to Thursday preferred, Tuesday to Friday possible)
        - The course delegates will be expected to perform self-study work on quizzes, examples and exercises outside of the above hours such that the course should be regarded as full-time. Online or face-face support will be available to help course delegates in their self-study work.

“Bespoke” combinations can be discussed if required (e.g. first 1 or 2 sessions and last session face-face, rest online).

4.3.2 Proposed dates

To be discussed

[1] If all delegates are from a single company that this can be run at their offices. Otherwise public courses will be run at a convenient location and the price of the course will reflect this.

4.3.3 Additional course deliverables and formats

This section outlines additional materials provided to support students in their learnings.

Note that all students have a Moodle login for the course and their personal use. This allows the student to get feedback on their understanding and progress and allows lecturer to track the same. This enables the lecturer to intervene in the student learning if needed.

A sample of previous Moodle courses are provided for reviewers of this proposal. Login details to review the sample are provided in section 8.3

4.3.4 Online forms

The lectures use MS forms to give students a chance to for discussion and learning from other students. They also allow short breaks in the delivery as previous experience suggests it can be difficult for students to focus for 2 hours continuously during delivery. The forms allow the students to reflect on the content and their understanding, encourage discussion, allow the trainer to get feedback on student understanding and provide clarity if/when required.

4.3.5 Quizzes via Moodle

Short quizzes per lecture are available to the students via their Moodle platform immediately after each lecture. The quizzes with automated grading) to provide feedback on understanding immediately after each lecture.

4.3.6 Lecture PDF

A PDF of each lecture will be provided in advance (as past experience shows some students like to read ahead or make notes on the PDF as they go)

4.3.7 Lecture video

A video of each online lecture is provided to allow students to review each lecture at their own time and pace.

4.3.8 Examples

A database of examples of best practice is provided for using UVM for DV. This is provided in advance to companies so students can access the database using the company preferred simulators and infrastructure.

4.3.9 Exercises

A database of exercises of is provided for using UVM for DV. This is provided in advance to companies so students can access the database using the company preferred simulators and infrastructure.

The exercises are provided at different levels of difficulty (beginner, medium, advanced) and different paths through the exercises are suggested based on those levels.

All exercises have model answers provided.

4.3.10 FIFO example and exercises

An extended example and exercise is provided for a FIFO to demonstrate how a block goes from IP level verification, through subsystem and onto SoC integration verification.

The IP level verification builds on the SV examples and uses basic UVM, the subsystem uses basic UVM (with an APB transactor) and the SoC uses basic UVM (the FIFO is connected to a RiscV CPU via an APB and the CPU communicates with an external UVM test bench, the test bench includes simple examples of filling/emptying the FIFO).

The same FIFO is used for the SV, UVM introduction and advanced UVM courses so students can build on their previous knowledge and to provide students with a sense of progression.

4.3.11 AI-based Q&A bot

Automated Q&A support based on GenAI-based bots pre-trained on SV, UVM and advanced UVM, the course materials, the quizzes, the examples and the exercises are provided to give effective support 24 hours per day.

Each student has a personal login to the bot for privacy purposed but this also allows the bot to learn from the previous sessions and thus adapt to the student's style of questions and learning.

The same login can be provided for all courses taken by the student so that they continue with their personal bot.

4.4 Measuring student progress and certification

The student is encouraged to use the online Moodle quizzes and to report which exercises they have completed. As mentioned above, this allows the student to get feedback on their understanding and progress and allows lecturer to track the same. This enables the lecturer to intervene in the student learning if needed.

A final test is also provided via Moodle that tests all concepts and techniques taught on the course.

For an attendance certificate (or similar) students need to demonstrate

- All of the Moodle topic tests have been taken and a pass mark achieved.
- All lectures attended (or the recordings watched).
- Sufficient coding labs have been correctly completed.
- The final quiz has been taken and a pass mark achieved.

4.5 Post course support

It is recognised that students cannot always complete all of the exercises and labs that they would like to during the course and so continuing support is provided after the course. Specifically:

- Recordings of the lecturers are kept available for 1 month (or longer if required).
- Access to the personalised Q&A bot is kept available for 1 month (or longer if required). It is re-enabled when the student starts a subsequent course.
- Moodle is kept open for 1 month (or longer if required) including the final test.
- Support for the students to complete the labs is provided for 1 month (or longer if required).

A final report on participation etc can be shared with employer if requested and suitable permissions given.

5 Trainer Profile and previous experience in delivering similar programmes

Dr Mike Bartley studied mathematical logic to PhD level at Bristol University before joining industry in 1988 to use formal languages (Z and VDM) in the development of software. He was also involved in the development and delivery of internal and external courses in both languages. In 1994, Mike moved to the semiconductor industry (at ST Micro) as part of a team introducing constrained random techniques (using a tool from IBM Research, Haifa) and formal verification into development of a new CPU and SoC. Mike used mathematical proof (e.g. for memory coherency), formal property proofs (again using tool from IBM) and developed a new equivalency verification solution for the full chip (using a new tool from Bull, France). After the successful tapeout Mike was engaged to deploy the new flow and train engineers globally across the company (working closely with Synopsys Research and their new tool formality). Mike moved to Infineon in 1999 to establish new verification flows using Specman (which he selected over Vera) and a formal tool from Siemens (that was eventually spun out into OneSpin). This established Mike as an engineer able to quickly understand new verification techniques, how best to deploy them productively

and how best to train other semiconductor professionals (from new beginners to “veterans”) in their use. He has continued this path through 2 start-ups and his own company (Test and Verification Solutions) which was acquired by Tessolve in 2020. Mike setup a new Alpinum Consultancy company delivering consultancy and training services and the training arm was spun out in 2023.

Mike qualified as a teacher in 1984 and has worked in schools, colleges and universities, as well as developing and delivering training courses for industry and the university sector. His formal teaching background established firm pedagogic foundations which he has developed through his professional career. Mike has also continued his own personal development including studies in higher education to obtain multiple master’s degrees. He has experienced a range of face-face and fully online learning and teaching styles and delivery techniques which he uses in all his courses: quick quizzes in the classroom; post-lecture Moodle tests; labs; final tests. These are used to provide feedback on understanding and progression to participants and trainer (in case some topics need further teaching).

Mike has been using constrained random verification since 1994 (when he started to use a research tool from IBM Haifa for CPU verification at ST Microelectronics). Mike was an early adopter of Specman in 2000 at Infineon and helped Verisity to develop the eRM . He has moved to newer verification languages (e.g. SV), methodologies (OVL, OVM, SVA, UVM, PSS, etc) as and when they were released, providing internal training to colleagues as and when required, as well as through public courses since 2008. Mike has worked on a wide variety of designs from different companies since 1994 using both simulation-based and formal verification, giving him invaluable insights for how to increase productivity. with a number of organisations, both as an employee and consultant. The following gives some examples from the list of company names which can be shared: ST Micro, Infineon, ARM, Intel, NXP, AMD, Panasonic, Samsung plus a number of smaller start-ups. Mike has also run an annual “Verification Futures” Conference in the UK in since 2011 and in Austin, USA since 2003. Mike has always been active in research and keeping abreast of new developments in semiconductors and verification specifically, a list of his presentations and papers is given in section 8.2.4

5.1 Example customer testimonials

More testimonials available upon request

5.1.1 Senior Director of Verification[2], Infineon Technologies

The Design Verification (DV) course (delivered by Dr. Mike Bartley on behalf of the TechWorks Academy) met our main objective which was to upskill our graduate and junior verification engineers to have the knowledge and understanding of verification to start supporting projects in the shortest time possible whilst minimising initial training and supervision time from our senior engineers.

The course completely covered the principles and background of DV to give a good foundation for the application of current best practice in constrained random stimulus generation, automated checkers and both code and functional coverage. All of this was bought together by the practical application of these techniques to an extensive set of examples and exercise that culminated in a full SV and UVM test bench. The course also compared and contrasted simulation and formal approaches to verification, allowing for a good understanding where each may be best deployed. Additionally, the course introduced important topics that might influence DV in the future, such as the application of AI/ML.

The course covers both IP, integration and SoC verification with the latter using an SoC running C-based tests running on an embedded CPU. A particular IP block is followed through the hierarchy and this enabled students to put each level of verification into context and understand the whole process to better appreciate the complete process from verification panning to signoff, and the activities and signoff criteria used at each level.

[2] Note, Infineon have kept the name of the Director confidential. But evidence that the feedback was provided by Infineon can be given.

Course understanding was helped with the extensive set of examples and exercises mentioned above. In addition, the course uses short quizzes in class to give a quick check on student understanding of a concept. These also prompted discussion which helps the interaction in online delivery. The course also provides continuously available online tests with immediate, automated feedback on student answers to help consolidate understanding. These are delivered using the Moodle platform which also allowed Dr. Bartley to track individual student progress and give individualised feedback. Dr. Bartley also kept us fully informed on course attendance.

One of the key benefits was he flexible delivery and timing of the course which not only allowed graduate participants to continue working on their own assigned projects during the training but also some senior engineers to participate to brush up their skills and also have first-hand feedback of how the junior engineers are progressing through the course. We did not have to assign internal senior engineers to support the upskilling of these engineers.

The course was delivered by Dr. Bartley, a very senior presenter with over 30 years' experience in DV who is still very active in the industry running large verification teams; organising, attending and speaking at conferences; providing DV training; helping companies to improve verification strategies, processes and efficiency; as well as being active in research. He was able to provide valuable insights into the practicalities of verifying real designs and was very responsive to participant questions, taking time outside of the online sessions to give individual support.

After each delivery of the course, Dr. Bartley took extensive feedback from all participants which get incorporated into the next delivery. We continue to put students on this course who benefit from this continuous improvement. We have also used a variation of the course for analog/mixed-signal verification and a new course on CPU verification. We continue to be impressed by the quality of both content and delivery.

5.1.2 Pete Leonard, Electronics Design Manager, Group Engineering, Renishaw

- The Design Verification course (delivered by TechWorks Academy) met our main objective which was to upskill our graduate engineers to have the knowledge and understanding of the application of VHDL and OSVVM in verification environments, the course also covered SV and UVM. Course understanding was helped with the extensive set of examples and exercises which allowed the participants to progress from initially simple, to complex design and test benches – which included debug and fix.
- The flexible delivery and timing of the course allowed participants to continue working on their own assigned projects during the training. We did not have to assign internal senior engineers to support the upskilling of these engineers.
- The course was delivered by a very senior presenter with over 30 years' experience in Design Verification. He was able to provide valuable insights into the practicalities of verifying real designs and was very responsive to participant questions, taking time outside of the online sessions to give individual support.
- After the 1st run and full delivery of the course, the organiser took extensive feedback from all participants which has now been incorporated into the next course. The next course will start on February 7th and we have already registered 8 participants for this.

6 Proposed Price

To be discussed once delivery format is agreed

7 Recommended Maximum no. of participants:

The ideal number of students is 14 although it can be run with less students (a minimum of 6 is suggested) or more (a maximum of 18 is suggested).

8 Appendices

8.1 Appendix A: Detailed course content

- Introduction to the UVM library, methodology and Universal Verification Components (UVCs)
- UVM component classes
- Packaging and directory structures
- Overview of the structure of a UVM test bench
- Top level test bench and connecting to a DUT
- Sequence items
  - Data types, constraints
  - Data operations (copy, clone, print, etc.)
- Sequences and sequencers
- The basics of virtual sequencers
- Test and testbench classes
  - Testbench layer
  - Test and test selection
- Reports
- Overview of the simulator and UVM phases
- Creating a simple UVM environment
- Basics of configuration
  - Configuration database (uvm_config_db)
  - Configuration objects
  - How configuration works
- Introduction to the factory
  - Type overrides
- UVM macros (such as uvm_do)
- Introduction to the objection mechanism
- Interfaces
  - Adding tasks, etc
- Introduction to UVCs
- Building a simple scoreboard
- Connecting components with tlm analysis interfaces
- Adding flexibility via the command line using parameters etc
- Register models
- Basic UVM debugging features.

Commercial simulators supported

- Aldec Riviera-PRO™
- Cadence Incisive®
- Siemens EDA Questa®
- Synopsys VCS®

8.2 Appendix B: Mike Bartley short CV and biography

8.2.1 Summary

• Currently working to define and execute an organisation-wide strategy for the deployment of Artificial Intelligence
• 10 years of creating and growing a technology service company to 400 employees in Europe, APAC and USA with over $8M annual turnover and $1M profit. Acquired by Tessolve in 2020.
• 30+ years of full life-cycle experience in software and hardware development for commercial, real-time, embedded and safety-related products including multicore hardware and software products

• Responsible for defining and managing software test strategies and software roll-out processes, measurable release criteria and objective signoff of numerous complex software/hardware products
• Management of large development teams and offshore teams
• Extensive experience of programming and debugging in a variety of languages (such as assembler, C, C++, Java, SQL, Unix scripting (awk, perl, tcl, etc), VHDL, Verilog, System Verilog, UVM and e)
• Strong academic qualifications (MSc in SW Engineering; Maths PhD; MBA) and professional certificates (ISEB Practitioner Certificates in SW Test Management and Analysis; Prince2)

8.2.2 Details

Feb 20 -	Senior Vice President,Tessolve	Responsible for European P&L. global VLSI sales support, R&D & AI strategy
May 08 – Feb 20	Founder and CEO, Test and Verification Solutions Limited	Developing UK-based self-funded company delivering services in software testing and hardware verification, acquired by Tessolve
June 06 – May 08	Software Test and Hardware Verification Manager, ClearSpeed Technology Plc	Responsible for the test and verification of all ClearSpeed SW & HW products, and the outsourcing of appropriate efforts to achieve that • Introduced SW test process & outsourced many activities leading to measurable quality improvements; reduced time to market; cost savings • Managed the version control, build and automated test framework, the defect tracking system, plus the development of internal tools. • Defined software release criteria • Lead the verification of aspects of existing chip Si verification – chip worked first time. Lead strategy for development of next chip.
02 – 06	Software Test and Hardware Verification Manager, Elixent (now Panasonic)	Responsible for the test and verification of all Elixent SW & HW products • Developed an automated test framework (written using make, Perl and other scripting languages) and tests (written in C, Verilog and VHDL) for testing the software toolchain and hardware IP • Developed and implemented tools and strategies for test and verification of all Elixent IP products • Developed test and verification deliverables (written in C) to help support customer integration of Elixent products
`99 - 02	Hardware Verification Manager, Infineon Technologies	Responsible for creating a new HW verification team and processes, and for spreading excellent QA practises across Infineon • Recruited, developed and managed team of over 35 engineers verifying all Bristol-based designs and many multi-site designs • Implemented verification processes employing state-of-the-art techniques and tools such that the Bristol site became recognised as a centre of excellence for functional verification • Wrote verification IP using the e language
`94 - `99	Verification Engineer, STMicroelectronics	• Verification manager for ST40 • Automated equivalence checking for multiple projects • Verification of a 64-bit CPU including development of a significant test generation tool using C++ and use of formal mathematical methods
`93 - `94	Lecturer, Bath College	• Lecturer in computing
`90 - `93	Software Engineer, Praxis	• Testing a distributed, real-time system specified in Z and written in C • QA team-leader on a client-server petro-carbon accounts system
`88 - `90	Programmer, IPL	• C and assembler tester and programmer and formal methods expert.

`90 - `99	Lecturer (part-time), Open University	• Java; SW Testing; SW Engineering; Pascal; Formal Methods

8.2.3 Education

2024	M.Sc. Blockchain and Digital Currency (University of Nicosia)
2022	M.Sc. (with Merit) in Data Science, Technology & Innovation (Uni of Edinburgh)
2022	M.Sc. (with Merit) in International Corporate Finance (Uni of Salford)
2021	M.Sc. (with Merit) in Intelligent Systems & Robotics (De Montfort Uni)
2019	M.Sc. (with Merit) in Information Security (Uni of London)
2018	Financial Times Non-Executive Director Diploma
2008	ISEB Practitioner Certificates in SW Test Mgt & Analysis
2001	MBA (Open Uni)
1999	MBA in Technology Management (Open Uni)
1997	M.Sc. in Software Engineering (Open Uni)
1988	Ph.D. (thesis on Category Theory) (Bristol Uni)
1986	M.Sc. (with distinction) in Logic (Bristol Uni)
1984	Postgraduate Certificate in Education (Bristol Uni)
1983	1st Class Hons. Degree in Mathematics (Bristol Uni)
1980	A-levels: Maths (A**); Further Maths (A*); Physics (A)

8.2.4 Appendix C: Examples of relevant papers written by Mike Bartley

- "*Testing Autonomous Vehicles*", STeP, India, 2020
- "*Verification Techniques for Safety and Security in Autonomous Vehicle Software*", Autonomous Vehicle Test & Development Symposium, 2018, Stuttgart
- "*Formal Verification Bootcamp*", Accellera Tutorial, DVCon, USA, 2019
- "*PSS: The Promises and Pitfalls of Early Adoption"*, DVCon Europe, 2018 and DVCon, USA, 2019
- "*V&V methods and tools for testing complex autonomous system*", AESIN 2018
- "*Delivering on the promises of Portable Stimulus*", CDNLive! EMEA, 2018
- "*Avoiding Test Debt*", EuroSTAR, Cpenhagen, 2017
- "*5G Challenges & Solution*", Embedded Conference Scandinavia, 2017
- "*Cyber Security of Medical Devices*", Clinical Engineering & IT Connectivity Conference NPAG, 2017
- "*Reliable Software Development*", Device Developers Conference, 2016
- "*Crowd Testing*", Test Management Forum, 2016
- "*Hardware Security Challenges and Solutions*", CDNLive! EMEA, 2016
- "*ISO26262 compliance*", NMI ISO 26262 Practitioners Workshop, 2016
- "*IoT Security*", Smart Summit Asia, Singapore, 2016
- "*Requirements driven Verification*", CDNLive! EMEA, 2015 and ESSS India, 2016
- "*Verifying Functional, Safety and Security Requirements (for Standards Compliance)*", DVCon, USA, 2015
- "*Static Power Intent Verification of Power State Switching Expressions*", DVCon, India, 2015
- "A Framework for AMS VIP development with UVM and Verilog-AMS", DVCon Europe, 2014
- "*Using Assertions to Capture and Check Application Wide Properties*", NMI Embedded Software Security, 2013
- "*Innovations in Hardware Verification and their novel application in Software Testing*", STeP, India, 2013
- "*Doing business in India*", UKIBC, 2013
- "*Requirements Testing: Turning Compliance into Commercial Advantage*", BCS SIGiST, 2013
- "*FPGA Verification*", NMI FPGA Conference, 2013
- "*UKTI/Tata Mentoring and Partnership Program*", UKTI, 2013
- "*Virtual Platform Verification*", Verifying Virtual Prototypes, 2013
- "*Achieving ISO 26262 Compliance for Silicon*", NMI ISO 26262 Conference, 2013
- "*Are we too hard for agile?*", IP-SoC 2012
- "*Benchmarking Functional Verification*", DVCon Europe, 2012
- "*Formal Hardware Verification*", Jasper User Conference, 2011
- "*Experiences in Developing and Using UVM Verification IP*", CDNLive! EMEA, 2013
- DAC – formal poster and "Resistance is Futile"
- "*Experiences in Automating Requirements and Risk-Based Based Testing*", Testing Experience, Dec 2011
- "*Knowing When You're Done with Your Requirements*", NMI Embedded Automotive Event, Nov 2011
- "*Standardizing the "e" language*", DVCon Europe, 2011
- "*Successful adoption of OVM and what we learned along the way*", with S. Holloway, Mentor Verif Horizons, Oct 2011
- "*Results of the NMI Verification Roadmap Study*", NMI 2010

- "*Benchmarking Functional Verification*", with Mike Benjamin, Mentor Verif Horizons, Oct 2011
- "*Strategic Outsourcing Of Software Testing*", Testing & Finance 2009, Homburg, Germany, June 2009
- *"How to Build & Maintain a Successful Outsourced, Offshored Testing Partnership",* Testing Experience, March 2009
- Chair & speaker on *"Introduction to Low Power Verification" at NMI Low Power verification Conference, Feb 2009*
- *"Successful offshoring of Software Development*", British Computer Society, January 2008
- "*Improved time to market through automated software testing*", Testing Experience, December 2008
- "*How to achieve win-win relation with outsource partner*", Test 2008, Bengaluru, India. October 2008
- "*Achieving Agility in Testing through Outsourcing*", Test 2008, Delhi, India. October 2008
- "*Lies, Damned Lies and Hardware Verification*", SNUG Europe, October 2008
- "*How to Boost your Productivity through Outsourcing*", Software and Systems Quality Conference, London, Sept 2008
- "*Outsourcing workshop*", SIGiST, British Computer Society, March 2008
- "*What SW testers can learn from HW verification engineers*", SIGiST, British Computer Society, March 2008
- "*Learning Not to Fear PCI Express Compliance Using a Predictable, Metrics Based Verification Closure Methodology*", CDNLive!, Munich, April 2008.
- "*A comparison of three verification techniques: directed testing, pseudo-random testing and property checking*", Mike Bartley, Darren Galpin and Tim Blackmore. DAC 2002.
- "*Why is verification so hard to plan?*", Mike Bartley. SNUG 2002, Paris.
- "*The Art of Verification*", Mike Bartley. Synopsys Compiler, Europe, September 2001.
- "*Verifying a complex, configurable peripheral using Specman Elite*", Mike Bartley and François Cerisier. Verisity User Group Conference, Munich, 2001.
- "*Verification - it's all about confidence*", Mike Bartley. SNUG 2001, Munich.
- "*Verification* IP: Travelling the Axes of Reuse", Mike Bartley, IP 2001
- "*Verification IP: Reuse of verification knowledge and tests*", Mike Bartley and Ian Hill. IP2000, Edinburgh.
- "Use of Formality" at San Jose SNUG 1999.
- "*Case Study: Using formal verification to design the Chameleon microprocessor*", Thomas Goust, Mike Bartley, Geoff Barrett and Frederic Rocheteau, Avant! Electronics Journal (IC Design Automation), August 1998.
- "*Coverage analysis in microprocessor verification*", Mike Bartley. Open University MSc dissertation.
- "*Microprocessor design verification by two-phase evolution of variable length tests*", Jim Smith, Mike Bartley and Terry Fogarty. IEEE International Conference on Evolutionary Computing, 1997.
- "*Functional verification methodology of Chameleon Processor*", C. Berthet et al., DAC 1996.
- Chapters on SW testing and maintenance for Open University MSc course on "*Software Engineering*"
- VDM entry in "*Encyclopaedia of Software Engineering*", Jon Wiley & Sons.

8.3 Appendix C: Moodle site and login details provided to allow reviewers to review Moodle sample

Course: Design Verification Course - Sample with Video | Your School

user ID: user1

Password: user1$123

Password to view the sample video = ONLINEDVCOURSEMIKE