World Transactions on Engineering and Technology Education
Vol.2, No.2, 2003

Computer-based software for testing students in digital logic theory and design

Charles Hacker

Griffith University
Gold Coast, Australia

ABSTRACT: Computer-based education is used extensively in the undergraduate digital electronic courses lectured by the author at Griffith University, Gold Coast, Australia. In order to aid in the teaching of the required digital logic concepts, the author has previously developed a suite of digital logic tutorial and simulation modules known as *WinLogiLab*. These modules provide tutorial and simulation of the entire digital electronic curriculum of the author's undergraduate course. The final requirement of the tutorial suite was to provide for student testing. This article describes the development of a computer-based testing software module, called *LogicTester*, that presents multiple-choice, short answer and, most importantly, *digital logic circuit design* questions for unsupervised student testing. The testing software is fully interactive and requires students to input their own designed digital circuits to solve given tasks, which the input circuit is then graded for accuracy. To allow for unsupervised student testing, extensive encryption of the software data files and user results files were necessary in order to ensure no deception in the student submitted results.

INTRODUCTION

The curriculum of the introductory undergraduate digital electronic courses lectured by the author at Griffith University, Gold Coast, Australia, covers the varied topics of number systems, Boolean algebra, discrete digital logic, minimisation techniques and sequential digital logic.

The author desired to make use of computer-based education to assist in the teaching of these digital electronic topics. However, it was initially discovered by the author that no presently available user-friendly visual computer software was obtainable that would effectively achieve this teaching [1-4]. This prompted the author to develop a suite of digital logic tutorial and simulation modules, collectively known as *WinLogiLab* [1-4]. This provided easy-to-follow, visual, step-by-step tutorials that covered all topics in the undergraduate digital electronic curriculum.

A demonstration version of the current stage of the complete *WinLogiLab* tutorial suite can be downloaded from the Griffith University Website at:
http://www.gu.edu.au/school/eng/mmt/ WinLLab.html.

The existing modules of the *WinLogiLab* tutorial suite have been successfully incorporated as part of the digital electronics curriculum in a second year course. Student feedback indicated the software as being educationally useful, beneficial to their studies and that it conveyed the desired information. Student feedback also provided valuable suggested improvements [1-4].

Currently, the full digital electronic course curriculum is covered by the existing *WinLogiLab* tutorial modules. Thus, to finish off the tutorial suite capabilities, it was deemed beneficial to add the capability of providing for testing of the introduced theory for the design of digital circuits.

This article presents the development of the additional software module, called *LogicTester*. The *LogicTester* module offers, firstly, standard multiple-choice and short answer type questions. Yet the most important extra feature of this software is that it presents *digital logic circuit design* questions. For this feature, the student is required to self-design and input a digital logic circuit to perform the requested task. This circuit is then checked against the computer-generated answer for a final grading.

EXISTING DIGITAL SIMULATION PACKAGES

The designed computer-based testing software was required to be compatible with the already successful *WinLogiLab* tutorial suite, and was required to be supplied free of charge to students in the digital electronics course. These requirements made existing available teaching software packages unlikely to be acceptable.

However, before attempting the programming of the computer-based testing software, an extensive check was conducted on existing software. It was discovered that little commercial and freeware packages on digital electronic testing were available. The search revealed the following three packages had incorporated testing capabilities. These were: *Computer Aided Tutorial Systems (CATS)* [5], *Digital Technology Learning Package* [6], and *Digital Logic Tutor* [7]. Various useful aspects of these software applications were incorporated into this designed *LogicTester*, specifically the fully interactive, user friendly and visual-based aspects.

The designed *LogicTester* actually provides more than what is offered by the existing applications, which only provide multiple-choice and short answer type questions. The designed *LogicTester* presents these capabilities, but also extends this to actually request students to design and input their own digital electronic circuits on screen.

THE BENEFITS OF COMPUTER-BASED TESTING

Computer-based testing provides a valuable learning exercise for students. A successful test will verify to students that they have obtained the required knowledge in digital electronic theory and circuit design. With the ability for students to install the software on their own computer, or via university 24-hour access computing laboratories, students are able to undertake a self-test at their own convenience and at their own pace.

A further beneficial aspect, for the author's concern, is that the software can provide for unsupervised examinable competency-based mastery tests. This refers to students being able to re-take the mastery tests until a suitable grade had been obtained. The author has previously introduced paper-based mastery tests for each topic within the digital electronic course. This had been shown to result in better student course completion rates and has been stated in student evaluations as being beneficial to students. Yet despite its student benefits, it also imposed a greater teaching workload for the author. The workload increased due to the amount of marking and the time and effort required for setting and supervising tests. The unsupervised computer-based testing will thus reduce this teaching load.

The provision of competency-based testing, where students can continuously retake tests, necessitated that a different selection of questions be presented with each test. This was achieved by firstly providing a large database of questions. Thus, different questions are drawn from the database with each subsequent test. Also, many questions have sections randomly generated during the test, which ensures the uniqueness of these particular questions.

For the software to provide unsupervised examinable testing, insurances are required that no student deception has occurred. One such insurance is to ensure a student's identity by requiring a password *log in* to the software. Another is to encrypt the questions, images and answers stored in the test database. This makes it unfeasible for students to reveal or extract the questions, images or answers by simply viewing the encrypted database. There is also a possibility for deception when submitting the final results. In order to overcome this, the saved results are encrypted and the hardcopy printed results contain encrypted validation security keys.

DESIGNED SOFTWARE

The flow diagram of the *LogicTester* software structure is shown in Figure 1. The *LogicTester* software starts by requesting to either start a new User Test file, or to load an existing saved User Test file.

When selecting to start a new User Test file, a screen opens to request the student's information. An example of the *new user input* screen is given in Figure 2. The requested information desired from a new user is the student name, student number and a user password. The password is stored in the encrypted Test file and is required to be entered when re-accessing the Test file. That is, upon loading a saved User Test file, the software requests the entering of a password and will allow access only if the entered password matches the stored password.

Encrypting the User Test file ensures that the file will have no meaning when an attempt is made to view by auxiliary software. This makes it impractical for any attempt at user modification of the file in any intelligible way.

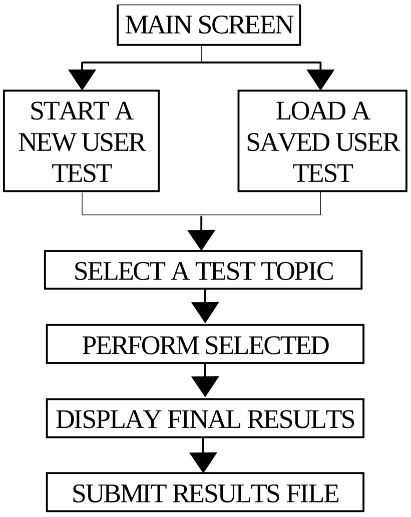
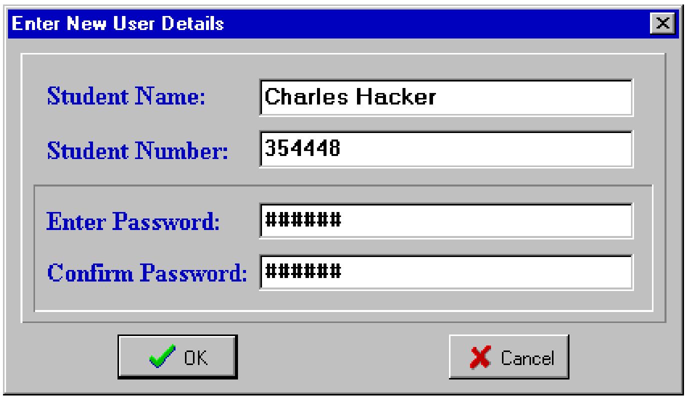


Figure 1: Flow diagram of the *LogicTester* software.


Enter New User Details

Student Name: Charles Hacker

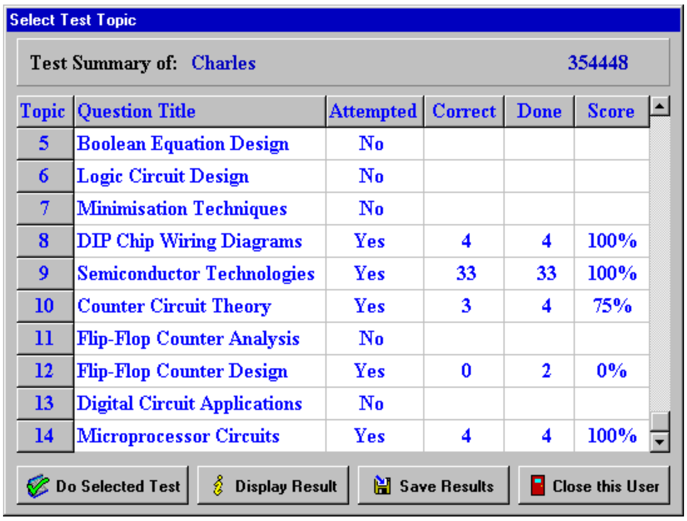
Student Number: 354448

Enter Password: ######

Confirm Password: ######

Figure 2: Input for a *new user* screen.

Once a new User Test file has been established, or a saved User Test file is loaded, a *Select Test Topic* screen is displayed, as demonstrated in Figure 3.


Select Test Topic

Test Summary of: Charles 354448

Topic	Question Title	Attempted	Correct	Done	Score
5	Boolean Equation Design	No			
6	Logic Circuit Design	No			
7	Minimisation Techniques	No			
8	DIP Chip Wiring Diagrams	Yes	4	4	100%
9	Semiconductor Technologies	Yes	33	33	100%
10	Counter Circuit Theory	Yes	3	4	75%
11	Flip-Flop Counter Analysis	No			
12	Flip-Flop Counter Design	Yes	0	2	0%
13	Digital Circuit Applications	No			
14	Microprocessor Circuits	Yes	4	4	100%

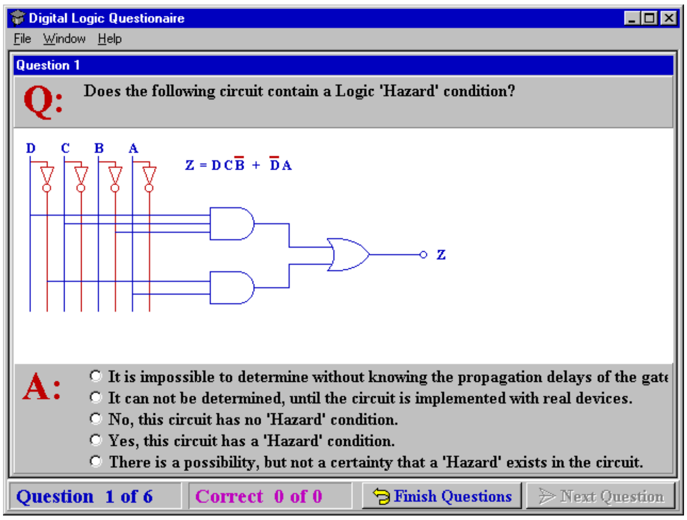
Figure 3: Test topic selection screen.

The screen in Figure 3 illustrates how the user is allowed to select various topics in digital electronics. These topics approximately follow the weekly lecture topics presented in the author's undergraduate digital electronics course.

The *select test topic* screen also displays the results of previously attempted test topics. This enables students to monitor their progress and offers them the opportunity to retake topic tests that have not met the required percentage score.

As previously stated, each test topic has a significant number of set and random generated questions stored in the test database. This allows for completely different sets of questions to be presented to the student with each subsequent retake of the topic test. Once a test topic has been selected, the program will randomly select a set of questions from the database and present each question to the student in turn.

The presented questions have a number of different formats. One of the formats is of the standard multiple-choice questionnaire. In this format, a stored question is displayed to the student and a selection of possible answers is displayed. The student is thus requested to select the most appropriate answer from the list. This is demonstrated in Figure 4.


Digital Logic Questionaire

Question 1

Q: Does the following circuit contain a Logic 'Hazard' condition?

A:
○ It is impossible to determine without knowing the propagation delays of the gate
○ It can not be determined, until the circuit is implemented with real devices.
○ No, this circuit has no 'Hazard' condition.
○ Yes, this circuit has a 'Hazard' condition.
○ There is a possibility, but not a certainty that a 'Hazard' exists in the circuit.

Question 1 of 6 | Correct 0 of 0

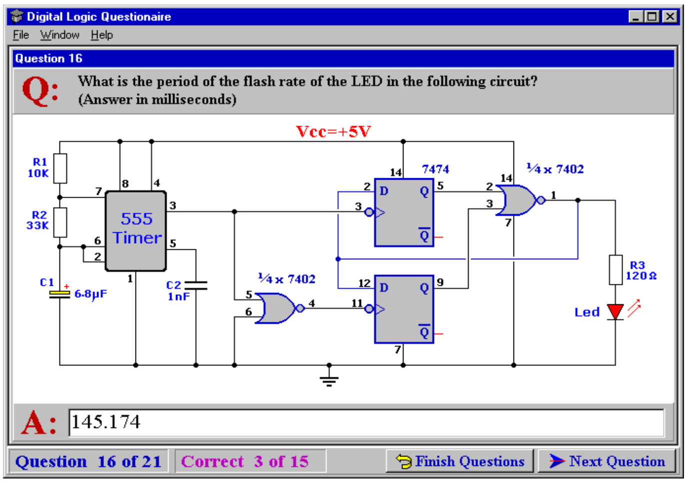
Figure 4: Example of a multiple-choice question.

As shown in Figure 5, the presented questions are usually based on a digital circuit or diagrams based on digital electronics. This circuit or diagram is also included in the test database as a stored encrypted digital image.

Another designed question format is that of the short answer type. In this case, the displayed question requests some form of typed input to answer the question, such as a short word, phrase or numerical input. An example of the short answer type of question is provided in Figure 5.

The above multiple-choice and short answer type questions are actually similar to all other available computer-based testing software. This software was designed to be much more, so the significant improvement in this designed *LogicTester* software actually tests the student’s ability to design and analyse digital circuits. This was made possible by utilising the code developed for the design of digital circuits incorporated in other *WinLogiLab* software modules.

One of the enhanced question formats allows the input of a Boolean equation. In this case, the student is required to analyse a given circuit image or other digital electronic images, and input the Boolean equation that describes the given image. An example of this question format is given in Figure 6.


Digital Logic Questionaire

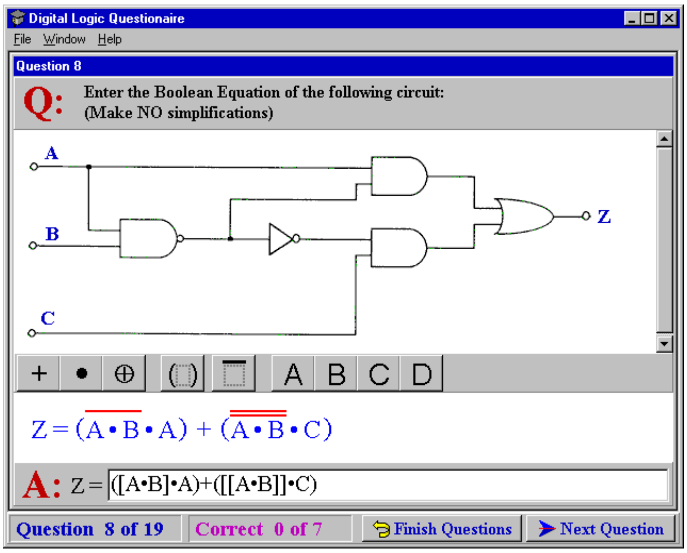
Question 16

Q: What is the period of the flash rate of the LED in the following circuit? (Answer in milliseconds)

A: 145.174

Question 16 of 21 | Correct 3 of 15

Figure 5: Example of a short answer question.


Digital Logic Questionaire

Question 8

Q: Enter the Boolean Equation of the following circuit: (Make NO simplifications)

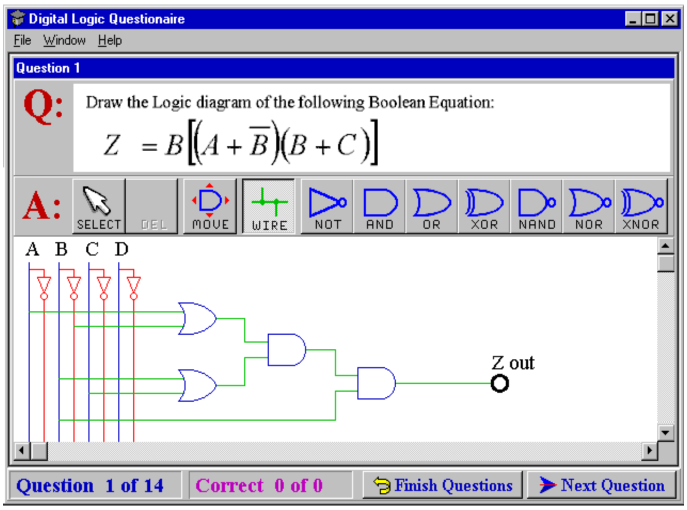
$Z = (\overline{A \bullet B} \bullet A) + (\overline{\overline{A \bullet B}} \bullet C)$

A: Z = ([A•B]•A)+([[A•B]]•C)

Question 8 of 19 | Correct 0 of 7

Figure 6: Example test to input a Boolean equation.

Another enhanced question format requires the student to enter a digital circuit. With this question format, either a Boolean equation, or design problem is presented to the student. A student is required to design a digital circuit in order to solve the given problem and input this designed circuit. The entered circuit is then examined against the internal computer generated solution. An example of this is given in Figure 7.


Digital Logic Questionaire

Question 1

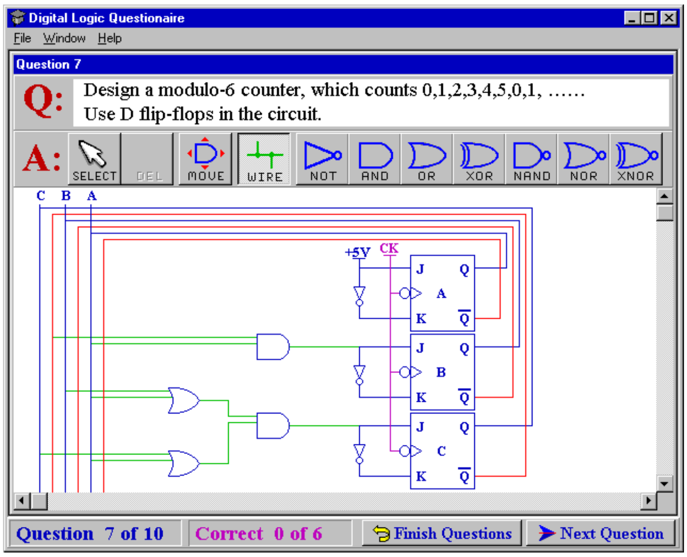
Q: Draw the Logic diagram of the following Boolean Equation:

$$Z = B\left[\left(A + \overline{B}\right)\left(B + C\right)\right]$$

Question 1 of 14 | Correct 0 of 0

Figure 7: Example test to design a digital circuit.

The final enhanced question format allows for the entering of a digital counter circuit. Similarly, in this case, a problem is presented to the student in which a digital counter is required to be designed to perform the task. The digital counter circuit is then entered on screen and subsequently compared to the internal computer generated solution. An example of this is given in Figure 8.



Q: Design a modulo-6 counter, which counts 0,1,2,3,4,5,0,1, ... Use D flip-flops in the circuit.

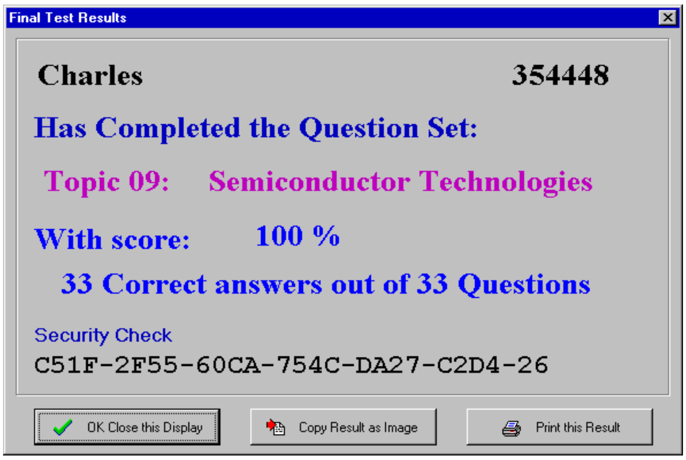
A:

Question 7 of 10

Correct 0 of 6

Figure 8: Example test to design a digital counter.

Once the user ends the selected question topic, or the user depletes the particular topic questions in the database, then a final result screen is displayed. An example of this is provided in Figure 9.



Charles 354448

Has Completed the Question Set:

Topic 09: Semiconductor Technologies

With score: 100 %

33 Correct answers out of 33 Questions

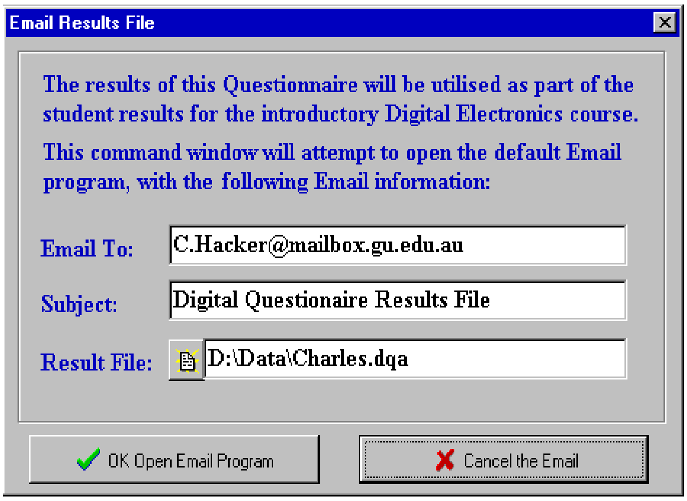
Security Check
C51F-2F55-60CA-754C-DA27-C2D4-26

Figure 9: Results of current test topic.

The result screen can be copied to an image for importing to other applications, such as word processors, or for later disk storage. The result screen can also be printed to provide a permanent hardcopy of the student results.

For the digital electronics course lectured by the author, the students are required to submit a printout of this result screen for examination purposes. To ensure that no deception has occurred by the altering of the result screen image, a security check string is provided. This check string contains a compressed and encrypted form of the student name, student number, question topic and results. This string can therefore be decoded and used to verify that the submitted result image is unmodified.

For further examination purposes, the student is required to periodically submit their entire encrypted saved Test file. This Test file can then be decrypted by the lecturer's decryption program and used to verify a student's results. The submission is performed electronically, via e-mail or can be copied and submitted on disk. An example of the e-mail submission of a Test file is given in Figure 10.



The results of this Questionnaire will be utilised as part of the student results for the introductory Digital Electronics course.

This command window will attempt to open the default Email program, with the following Email information:

Email To: C.Hacker@mailbox.gu.edu.au

Subject: Digital Questionaire Results File

Result File: D:\Data\Charles.dqa

Figure 10: E-mail submission of test results.

CONCLUSION

This paper presents the development of the software module *LogicTester*, which forms the final phase of the existing *WinLogiLab* computer-based educational software suite. The *LogicTester* module completes the software suite by providing student testing on the required digital electronic theory and circuit design concepts. With the inclusion of encryption and security checks, the software will be utilised for unsupervised computer-based competency testing. This will result in a reduced teaching workload in the author's digital electronics course.

REFERENCES

1. Hacker, C. and Sitte, R., Development of a computer program to electronically design digital logic circuits. *Proc. 9th Annual Australasian Assoc. for Engng. Educ. (AaeE97)*, Ballarat, Australia, 353-357 (1997).
2. Hacker, C. and Sitte, R., A Computer based tutorial for demonstrating the solving of digital electronic circuits. *Proc. 10th Annual Australasian Assoc. for Engng. Educ. (AaeE98)*, Gladstone, Australia, 509-519 (1998).
3. Hacker, C. and Sitte, R., Implementing the Espresso - two level logic minimiser algorithm in the MS-Windows environment. *Proc. 2nd Asia-Pacific Forum on Engng. and Technology Educ.*, Sydney, Australia, 124-127 (1999).
4. Hacker, C. and Sitte, R., A computer based teaching program for the design of digital counter circuits. *Proc. 3rd UICEE Annual Conf. on Engng. Educ.*, Hobart, Australia, 225-228 (2000).
5. Computer Aided Tutorial Systems (CATS). Glencoe (2002), http://www.sweethaven.com/glencoe/bsee
6. Delahoy, R.J., Digital Technology Learning Package. Melbourne (1995).
7. Foster, G.N., *Digital Logic Tutor I*. New York: Maxwell MacMillian Publishing (1994).