Liu, C.-C. et al. (Eds.) (2014). Proceedings of the 22nd International Conference on Computers in Education. Japan: Asia-Pacific Society for Computers in Education

A Study on Teaching Debugging Strategies for Digital Circuit

Hong-Fa Ho [a*] & Duen-Huang Huang[b]
[a]*Department of Electrical Engineering, National Taiwan Normal University, Taiwan (R.O.C.)*
[b]*Department of Industrial Education, National Taiwan Normal University, Taiwan (R.O.C.)*
*jackho@ntnu.edu.tw

Abstract: It is essential for educators to identify the student's reading ability when learning debugging strategies. Forty-five senior high school students (average 17.6 years old, *SD*=0.35) participated in this study. The participants were divided into three groups to examine the effect of three teaching materials including 1) normal reasoning, 2) a debugging process model, and 3) a debugging process model and three debugging strategies, namely, forward strategy, backward strategy, and CLOCK-based strategy. While the participants debugging a digital circuit question, the eye tracker recorded the eye movement data for analysis. The results indicated that participants receiving a debugging process model and three debugging strategies improved their debugging skills significantly by: 1) increased percentage of the participants finding bugs correctly, 2) increased percentage of the participants using debugging strategies correctly, and 3) reduced time spent on debugging in average. The findings may help educators and researchers understand the benefit of debugging strategies on digital circuits to design better teaching materials, methods, and learning strategies.

Keywords: Digital logic circuit, timing diagram, debugging strategy, reading process, eye movement data

1. Introduction

Digital circuit, especially digital integrated circuit is one important electronic product. Engineers design and test digital circuit functions using software development tools such as tools for FPGA, ASIC, and CPLD. Debugging is a term to describe the engineers identifying system errors by observing a simulation software or logic analyzer. Since debugging is inevitable in system development process and required repetitive checks, one of the key capabilities of electronics engineers is to read digital circuits and timing diagrams efficiently and strategically. However, there is a lack of complete textbooks introducing debugging strategies on digital and integrated circuits. The authors identified three consistent debugging strategies based on most practices electronics theories including: 1) forward (from inputs to outputs), 2) backward (from outputs to inputs), and 3) CLOCK-based. By using eye tracker, the purpose of this study is to understand students' debugging processes and the influence of three strategies on students' debugging capability.

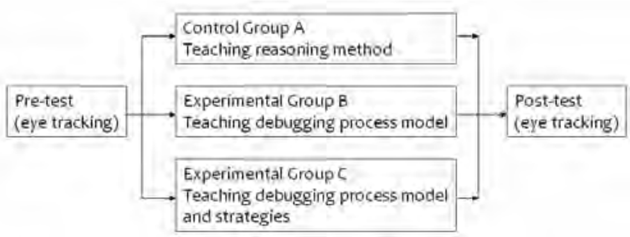
2. Literature review

Many research findings have identified the importance in understanding domain knowledge (Dochy, Segers, & Buehl, 1999), reading technology (Voss & Silfies, 1996), and reading strategies (Cottrell & McNamara, 2002) for general scientific knowledge. However, reading strategies on digital circuits and timing diagrams are rarely discussed and limited training are provided for engineers.

Araki, Furukawa, and Cheng (1991) proposed a theory of debugging process model, which is a repeat procedure including initial hypothesis, hypothesis modification, hypothesis selection, hypothesis verification, and inspection process on correct bug finding. Initial hypothesis refers to an aspect of error report. If there is no error report, the engineers need to find out where the error is in order to continue the initial hypothesis. Xu and Rajlich (2004) explained the cognitive process of knowledge, comprehension, application, analysis, synthesis, and evaluation in program debugging. The eye reading scan path recorded by eye trackers as objective evidence can examine the initial hypothesis on correct bug finding related to Bloom's six cognitive categories.

3. Research framework and process

Teaching materials and test materials were designed based on the debugging process model and three types of debugging strategies: forward strategy, backward strategy, and CLOCK based strategy (four subtypes of CLOCK triggering including high-level, low-level, positive-edge, and negative-edge). Before the experiment, the study provided readily available prior knowledge resources based on the needs of learners to avoid the sample bias on various prior knowledge and memory among participants. The detail research framework is shown in Figure 1; the participants followed the experimental procedures by pre-test, course modification, and post-test. The complete interactions were recorded in the computer eye tracker for research analysis.



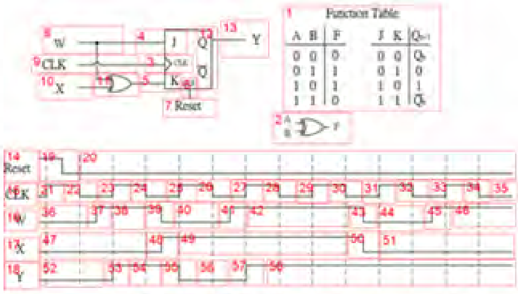


Figure 1. Research framework and process. Figure 2. Defined ROIs on the digital logic circuit.

To explore the different debugging strategies taught by the same instructor on the effect of debugging capabilities among the participants, following research method was conducted.

4. Method

4.1 Participants and Materials

Participants were volunteered from HsinChu Kuang-Fu high school; all participants had experience, and gained credit points in digital circuits. A total number of 45 senior students (average 17.6 years old, *SD*=0.35) received parental consent to participate in this study. The participants were divided into groups based on the original classes they were attending: N_A =15, N_B =14, N_C =16. The participants had normal or corrected eyesight.

Test Materials: each participant undertook four eye tracking experiments including two pre-test and two post-test questions on digital logic circuit (J-K flip-flop), timing diagram (with one bug), and function table. Three teaching materials included: 1) normal reasoning, 2) a debugging process model, and 3) a debugging process model and three debugging strategies named forward, backward, and CLOCK-based strategy. CLOCK-based strategy has four subtypes relating to digital circuits CLOCK trigger models including high-level, low-level, positive-edge, and negative-edge triggering.

4.2 Design, Procedure, Instrument, and Data Analysis

This study is a 3x9 factorial quasi-experimental design comprised of three major experimental steps including pre-test, course modification (by the same instructor), and post-test. Independent variables included teaching materials (A: normal reasoning, B: debugging process model, C: debugging process model and debugging strategies), and Region of interest (ROI in short)(nine ROIs shown in Figure 2). Dependent variables included 1) the number of bugs found correctly, 2) the number of times participants used debugging strategies been taught, 3) average time-spent in finding bugs correctly, and 4) eye movement variables (including scan paths and LCS of scan paths). The procedure is shown in Figure 1. EyeNTNU-180 eye tracker presented with a collection of experimental stimuli and record information on the participant's eye movements.

The average percentages of finding correct bugs were calculated based on participants' answers. Because the large eye movement data output from the eye tracker, the manual analysis became difficult to determine if the scan path contain equivalent eye movement data when the participant utilizing debugging strategies. The longest common subsequence (LCS) algorithm (Bergroth, Hakonen, & Raita, 2000) was then used to determine the scan path if a participant used the debugging strategies.

5. Results

Results of the analysis are shown in Table 1.

Table 1: Results on participants correctly debug digital circuit questions.

		Control group A (N_A=15)	Experimental group B (N_B =14)	Experimental group C (N_C =16)
Pre-test	# of Found bugs	16.67%	13.33%	16.67%
	Used debugging strategies	3.33%	3.33%	3.33%
	Average time-spent	236sec	277sec	244sec
Post-test	# of Found bugs	13.33%	26.67%	36.67%
	Used debugging strategies	3.33%	10.00%	60.00%
	Average time-spent	242sec	182sec	113sec

6. Findings and Implications

The results showed participants undertaking classes of debugging process model and debugging strategies had significant improvement in debugging ability and performance by: 1) increased percentage of the participants finding bugs correctly, 2) increased percentage of the participants using debugging strategies correctly, and 3) reduced time spent on debugging in average. The proven effectiveness of the debugging strategies will support and further to develop teaching software in electronic strategies.

Acknowledgements

This research is partially supported by the “Aim for the Top University Project” of National Taiwan Normal University (NTNU), sponsored by the Ministry of Education, Taiwan, R.O.C. and the “International Research-Intensive Center of Excellence Program” of NTNU and National Science Council, Taiwan, R.O.C. under Grant no. NSC 103-2911-I-003-301.

References

Araki, K., Furukawa, Z., & Cheng, J. (1991). A general framework for debugging. *Software, IEEE, 8*(3), 14-20.

Bergroth, L., Hakonen, H., & Raita, T. (2000). *A survey of longest common subsequence algorithms.* Paper presented at the String Processing and Information Retrieval, 2000. SPIRE 2000. Proceedings. Seventh International Symposium on, A Curuna.

Cottrell, K., & McNamara, D. S. (2002). *Cognitive precursors to science comprehension.* Paper presented at the Proceedings of the Twenty-fourth Annual Meeting of the Cognitive Science Society, Fairfax Virginia, USA.

Dochy, F., Segers, M., & Buehl, M. M. (1999). The relation between assessment practices and outcomes of studies: The case of research on prior knowledge. *Review of educational research, 69*(2), 145-186.

Voss, J. F., & Silfies, L. N. (1996). Learning from history text: The interaction of knowledge and comprehension skill with text structure. *Cognition and Instruction, 14*(1), 45-68.

Xu, S., & Rajlich, V. (2004). *Cognitive process during program debugging.* Paper presented at the Proceedings of the Third IEEE International Conference on Cognitive Informatics