Journal of Electronic Testing (2022) 38:463–464
https://doi.org/10.1007/s10836-022-06032-9

Editorial

JETTA Volume 38, Number 5, October 2022

Vishwani D. Agrawal[1]

Received: 23 October 2022 / Accepted: 23 October 2022 / Published online: 2 November 2022

I am pleased to announce the *2021 JETTA-TTTC Best Paper Award*. The award selection committee consisted of Xiaoqing Wen (Chair), Nicola Nicolici, Jennifer Dworak, Jiun-Lang Huang, and Erik Larsson. We congratulate the winning authors, Thiago Copetti, Guilherme Cardoso Medeiros, Mottaqiallah Taouil, Said Hamdioui, Letícia Bolzani Poehls, and Tiago Balen. Their winning paper, "Evaluation of Single Event Upset Susceptibility of FinFET-based SRAMs with Weak Resistive Defects," appeared in *JETTA*, Volume 37, Number 3, pages 383-394, June 2021.

Yervant Zorian, TTTC President, announced the award in a plenary session of the International Test Conference at Anaheim, CA, on September 29, 2022.

This issue has seven articles including one letter. The topics discussed are software test and verification, printed circuit board testing, hardware security, self-healing from single event effects, and approximate circuits.

In the first paper, the authors address software test. They use machine learning to partition test cases into clusters based upon similarity. An optimization procedure removes redundant tests and prepares an execution order for tests, resulting in increased test efficiency. The contributors of this research are Varun and Karthika from Vels Institute of Science, Technology and Advanced Studies, Chennai, India.

Next, we have two papers on printed circuit board (PCB) testing. The first of these, second paper in this issue, is a review on defects in PCBs. The authors, Udaya Sankar, Gayathri Lakshmi, and Siva Sankar from SRM University, AP, India, illustrate large number of defects found in PCB assemblies.

Continuing with the theme of PCB defects, the third paper focuses on connector failures. The cause of these is fretting or wear due to relative motion of connecting parts. Such failures result in loss of signal integrity. Contributors of this work are Doranga and Zhou from Lamar University, Beaumont, Texas, USA, and Poudel from Appalachian State University, Boone, NC, USA.

Then, we have two papers on hardware security. The first of these, fourth paper of this issue, presents a design for a physical unclonable function (PUF). The state of a PUF is hardware dependent and derived from a random phenomenon. As is well known PUFs are used to provide security in hardware. Lai from Relatch, Inc., Hsinchu, Taiwan, and Huang and Lee from National Cheng Kung University, Tainan, Taiwan present a static random-access memory (SRAM) based PUF. In the past, SRAM PUFs have used stable bit cells whose power-up state is always same every time the memory is powered up. This work proposes the use of unstable bits along with the stable bits and shows the benefits.

The fifth paper focuses on logic locking. With logic locking implemented on a chip, a pirated device stays unusable unless the secret locking key is programmed into it. There are continuous efforts to make the key secure. However, keeping the key secret is an open question. This paper presents a novel key recovery technique to break secure logic locking. Authors are Zhong, Jain and Guin from Auburn University, Auburn, Alabama, USA, Rahman and Asadizanjani from University of Florida, Gainesville, Florida, USA, and Xie from Villanova University, Villanova, Pennsylvania, USA.

The sixth paper describes the design of a correcting circuit for errors from single event upsets (SEU). An SEU is a flipped logic value due to radiation external to the circuit, which can be a serious concern in space applications. The authors call their method a self-healing design based on an intrinsic evolutionary approach. They use genetic algorithms (GA) but employ heuristic for convergence in fewer iterations. Contributors of this work are Deepanjali and Mahammad Sk from Indian Institute of Information Technology, Design and Manufacturing, Kancheepuram, Chennai, India.

✉ Vishwani D. Agrawal
agrawvd@auburn.edu

[1] Auburn University, 36849 Auburn, AL, USA

Seventh paper is a *JETTA Letter* contributed by J. Rao from Vignan's Institute of Engineering for Women, Visakhapatnam, Andhra Pradesh India, T. Rao from GITAM University, Visakhapatnam, Andhra Pradesh, India, Ramya from Government Polytechnic for Women, Srikakulam, Andhra Pradesh, India, and Ajaykumar and Trinadh from Sir C. R. Reddy College of Engineering, Eluru, West Godavari District, Andhra Pradesh, India. They present an approximate multiplier design using the Karatsuba algorithm (see following paragraphs). In this algorithm, the two multiplicands are split into four integers, each with fewer digits. The final product is then obtained through multiplications and additions of those integers. Savings in circuit area, power and delay come from the reduced multiplication complexity due to shorter operands. In the technique presented in the paper the operands are rounded off prior to multiplying. The results show improved performance.

There is an interesting story about the Karatsuba multiplication algorithm. You must have heard about the Russian mathematician Andrey Kolmogorov. In 1960, at a seminar he presented his conjecture on integer multiplication complexity being proportional to the square of the number of digits in multiplicands. Soon afterwards Anatoly Karatsuba, a student attending the seminar, brought a faster algorithm to Kolmogorov, who at once wrote it up and sent for publication in the name of Karatsuba. The paper appeared in 1962 and it was only after receiving the author's reprint that Karatsuba learnt about the publication of his invention.

Multiplication of integers consists of multiplications of digits and several additions. In terms of complexity multiplications dominate and additions are neglected. Thus, we arrive at Kolmogorov's conjecture. Karatsuba's algorithm splits the two multiplicands into four integers, each with half as many digits. It then uses three multiplications among these integers to produce the result. Multiplicand splitting was known to Charles Babbage who designed the difference engine almost a century earlier. However, his split multiplicands required four products. It is Karatsuba's three products that are responsible for the reduced complexity. Also, recursive splitting of multiplicands and repeated applications of the three-product procedure produce a neat sub-quadratic complexity algorithm.