SCIENTIFIC OASIS

Decision Making: Applications in Management and Engineering

ISSN: 2560-6018, eISSN: 2620-0104


# TWO EFFICIENT APPROXIMATE UNSIGNED MULTIPLIERS BASED ON MULTI-LEVEL SEGMENTATION

**[1]Dr S Bhavani**

HOD, ECE- Sri Shakthi institute of engineering and technology, Coimbatore, India, Orcid ID - https://orcid.org/0000-0002-3463-0014

**[2]Vignesh kumar R**

M.E- VLSI final Year, Sri Shakthi institute of engineering and technology, Coimbatore, India

**[3]Ms.R.Devadharshini**

Assistant professor, RVS College Of Engineering And Technology, Coimbatore, India

**Abstract:** Approximate multipliers are very relevant aspects in the modern era of technology. Technological implementation leads to the development of the efficiency of multipliers so that the accuracy rate of output can be improved. In this context, facilities for mitigating errors along with less power consumption are the key benefit of multipliers. Regarding this fact, the significance of multipliers having advanced adders for data processing has been highlighted in this topic, apart from. The evaluation of Hardware Description Language (HDL) has been included which represents a brief history of upgradation of HDL.

## Introduction

Parallel multipliers are a significant part of high-speed systems that need high efficiency. Central Processing Units (CPU), Digital Signal processors (DSP) and many more multi-media applications are considered systems of high speeds. Practical designs are implemented in high-speed systems to mitigate complexity during signal progression. The multiplication process follows three steps for delivering output. Initially, the partial products are provided as input, allowing the sum of these products. The summing process is continued to create two rows that will be included by means. Consumption of power along with the size of the circuit is a vital part of the Parallel multiplier because these factors are the most effective part of employed gates that are the potentials of architecture.

The process of power consumption and delaying of performance includes parallel multipliers and enhancement of speed along with less power consumption of the entire system can assure improvement of performance in this stage. Approximate Unsigned Multipliers are such effective application that impacts on enhancing the performance of high-speed systems. From this perspective, the effectiveness of Approximate Unsigned Multipliers in diverse segmentation is going to be discussed in this chapter. The purpose of this study is to evaluate the benefits of the configuration of error recovery that plays an active role in multi-segmentation.

**Significance of Study**

Multipliers refer to the core circuit of arithmetic in high-speed applications such as DSP. approximate multiplier is used in DSP applications because it enables this application at a high level along with effective features of potential circuital path and less consumption of power. Advanced and high-tech approximate adder is included with this multiplier and it leads to the accumulation of product quickly and efficiently. In addition to this, accuracy levels are increased through using the gates, named 'either OR' and approximately adder is also beneficial for recovering error (Jothin *et al.,* 2020). This high performance is largely relevant in the context of the modern era because the present world relies on advanced digital support and the upgradation of technologies plays a crucial role in providing the high-level performance of the approximate undesigned multiplayer. Considering this fact, this study focuses on the efficiency of approximate multipliers that are involved in multiple segmentations of the modern technological era.

Approximate computing is one of the leading trends that has become popular for its unique design in enhancing energy efficiency to support digital systems. Perfect computing accuracy is not required for applications like data mining, and recognition of multimedia because these applications become familiar with a tolerance of error (Chippa *et al.*, 2013). The result of the DSP application is considered as the interception of human consideration. In this context, the role of approximate circuits is significant for mitigating the possibility of delaying, and power-consuming in digital systems and achieving the potential level of energy efficiency. In addition to that, the advanced adder that is used in approximate multipliers facilitates features of the parallel data process and it is completed by cutting the chain of carry propagation.

Adder of approximate multipliers can delay the critical path and it is shorter than a conventional full adder like a one-bit adder. The study of approximate unsigned multipliers is exclusive because the adder that is used in these multipliers detects the error signal by computing the sum. This feature ensures error reduction at the completion stage of the multipliers (Nojehdeh and Altun, 2020). A simple tree is designed for the approximate adders and it is used to accumulate partial products. The signals of error impact on generating better accuracy of this application. From this perspective, it needs to be said that energy efficiency along with the accuracy of data enables the approximate multipliers to provide high effectiveness to the high-speed systems, DSP application.

## Methodology

This research is done by analysing the secondary data source. It indicates that the popular articles, and journals on Approximate Unsigned Multipliers help to complete the research. These journal, articles are found in Google Scholar which is one of the reliable sources of data collection. Secondary data collection offers a wide range of data and information that is used for quick understanding. Based on the collected data the researcher finds out the key insights that help to understand the multiple segmentation of Approximate Unsigned Multipliers (Barbareschi*et al.,* 2021). The use of secondary data facilitates connecting the diverse aspects of this particular topic so that one can get an effective statement on this topic. It becomes beneficial for gathering sustainable knowledge on this topic and this helps to deliver a particular view of this topic to the readers. Analysing secondary data leads to the evaluation of this topic which helps to focus on the accuracy of this topic.

## Findings

Hardware Description Language (HDL) refers to a specialised language of computers and it plays an active role in describing the structure and behaviour of digital circuits of logic that is included in electronic circuits. HDL is similar to the programming language and is considered a textual description that consists of statements control structure and expression (Yousefifeshki *et al*., 2023). HDL is included in Electronic Design Automation (EDA) for complex circuits. In addition to that Verilog is one kind of HDL and it is used to design the hardware. Verilog provides facilities to the users for digital designing Register Transfer Level, Behaviour Labels, Switch labels and Gates labels. The Behavioural Level denotes a system of concurrent algorithms that is called behavioural. It presents the sequence of algorithms that consists of instructions of execution one algorithm after the other. Tasks, functions along blocks are the key elements of this level.

Register-Transfer Level is designed with the features of the circuit by data transfer and operations between the allotted register. The use of an explicit clock provides benefits to the design of RTL so that it contains accurate possibilities of timing (Singh and Chandel, 2020). It allows the operations that are scheduled, to happen at exact times. Based on characteristics of the logical level, the system is evaluated by timing properties along with logical variables. All signals are included as discrete signals that have definite logical values such as ‘X’, ‘Z’, ‘0’, and ‘1’. The usable operation is described by logical primitives like NOT, OR, AND. modelling of the Gate level is not an appropriate idea for the logical design level. The code of the gate level is made by using the synthesis tools and it allows the simulation of the gate level along with the backend.

Verilog was introduced as the modelling language of proprietary hardware by Gateway Design Automation in the year 1984. Initially, Verilog could not be identified as a standardized language and from 1984 to 1990 modification of language was initiated that helped to come out the revised and advanced version of language (Kong *et al*., 2023). The use of the Verilog simulator began in 1985 and it was extended to year 1987. In the initial stage, Gateway sold the

Verilog simulator implication and an extension version of Verilog was introduced as Verilog-XL. This extension version included the functioning of the XL algorithm that followed the effective process of simulation at the Gate level. Cadence Design System which was popular for a thin process simulator of film, decided to take over the automation system of Gateway (Ejtahedand Timarchi, 2022). it enabled Cadence to become responsible for the ownership of the Verilog language. At that time, this company marketed Verilog as a simulator as well as a language. At that time pressure of standardisation led to the transformation to VHDL. Open Verilog International (OVI) was organised by Cadence and it provided documentation of the Hardware Description Language of Verilog in 1991.

OVI played a large role in improving the Language Reference Manual (LRM)that clarified things and created specifications for the language. In 1994, IEEE 1364 was formed as a working group and had the purpose of transforming OVI LRM to the standard of IEEE (Ha and Lee, 2017). It was impactful because Verilog was turned into IEEE format in 1995. Apart from this, the Verilog Compiled Simulator was popular in the market and this compiler was defined as the opposite of an interpreter. It needs to be said that despite having enough time to compile, the execution speed of the simulator was fast. Verilog 2001 with advanced features was introduced and it was able to fix the problems of Verilog 1995.

The tools of Xilinx ISE are designed for entering several processes such as diagrams of state machines, graphical schematics, Verilog, and VHDL. The project navigator interface includes sub windows that are effective for accessing tools for the implementation of the design. The top left of the project navigator interface includes start, files, libraries panel and design and it enables users to access potential source files (Jain *et al*., 2017). The Start window facilitates quick access for opening a project file along with accessing documentation, reference material and tutorials. Error console and warning panels are displayed at the bottom of the interface in the project navigator that is seen in Figure 1. The multi-document Interface is placed at the right of the project navigator and it is denoted as a workspace that allows users to view text files, design reports, and simulation waveforms. Each window is facilitated with a resize and having undocked, each window can be moved.

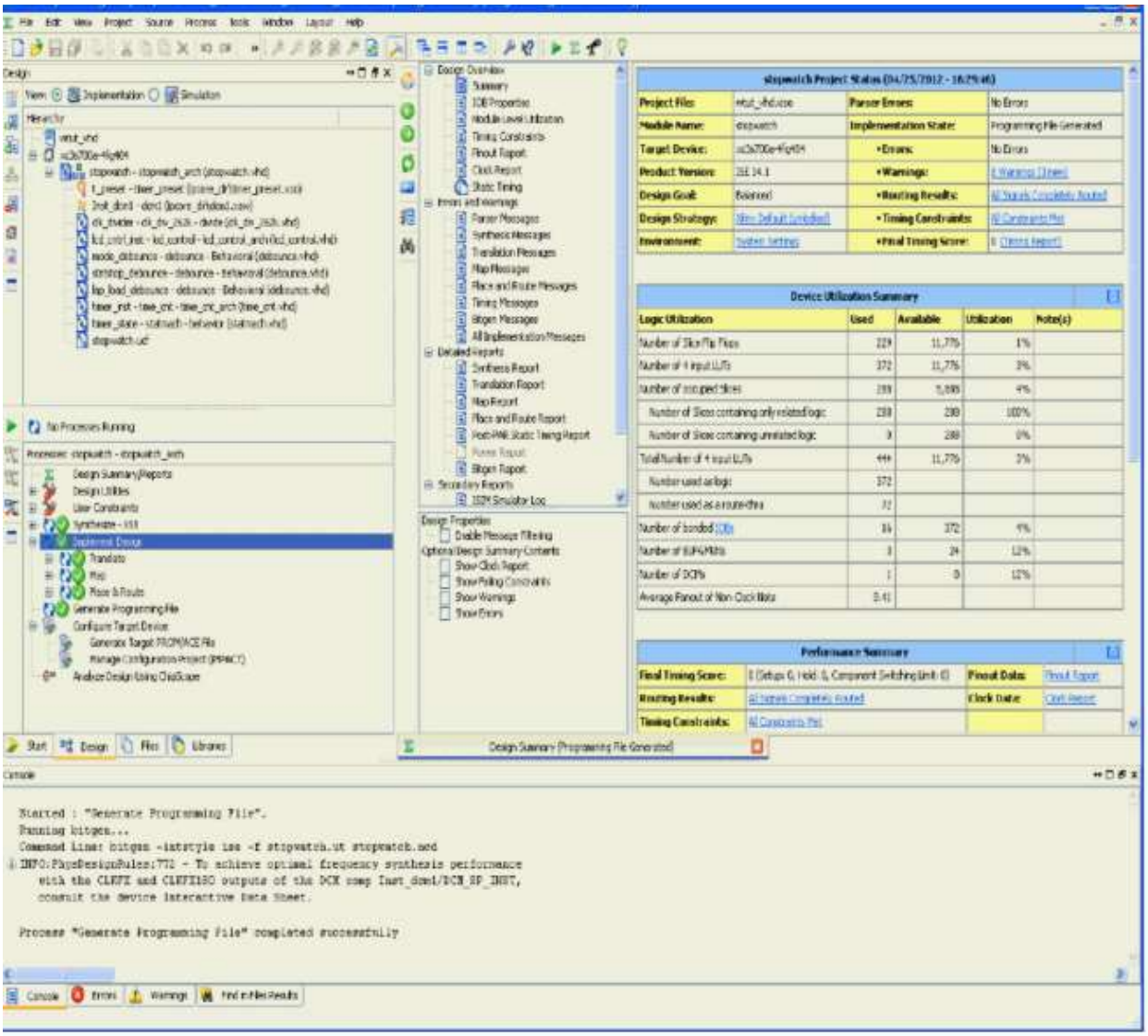

**Figure 1: Project Navigator Interface**

(Source: Accessed from Dissertation)

The design panel provides accessibility in the Hierarchy pane, View pane and Processes Pane. the radio button of the View pane helps the users to display the modules of the source that are aligned with the design view of the simulation, placed in the Hierarchy pane. The simulation phase needs to be selected by the users from the down list in case of selecting simulation (Ahmadinejad *et al.,* 2019). The hierarchy pane shows the user document, name of the project, and target device along the the source files for designing. the view pane that is shown at the top of the design panel enables the users to check the source files that are included in the design view. The icon of the Hierche pane expresses the type of file such as HDL file, core file, schematic or text file. The files having an ineffective level of hierarchy are denoted by the plus symbol (+) and by clicking on this symbol users can be allowed to expand the hierarchy. Apart

from that, having double-clicked on the file name, the file can be opened up for editing. The functions that are necessary for defining, analysing and running design can be run by the process pane. Tools for editing design, analysing, and viewing are opened in Workspace. These tools include schematic editor, ISE text editor, report viewers and constraint editor. analyser of timing and many more.

ModelSim can be defined as simulation and verification tools for Verilog, VHDL and system verilog. The multiple languages are designed in the simulation environment of HDL (Christianto *et al.,* 2022). ModelSim is used by following independent ways and aligning with XilinxISEand AlteraQuartus. It is noticed that use of automatic scripts and Graphical User Interface (GUI) are used in the performance of simulating. ModelSim has become advanced by modifying its features. In this context, diverse editions of ModelSim such as ModelSim XE, ModelSim PE, and ModelSim SE are available in the market. ModelSim PE is considered the basic level simulator for students and hobbyists while ModelSim SE includes the capability of advanced debugging along with the high level of performance. The purpose of ModelSim SE is to design multimillion gates. It works efficiently in Linux and Microsoft Windows. ModelSim XE is the advanced version of Xilinx Edition and is designed to align with Xilinx ISE. ModelSim XE is efficient for testing the programmes of HDL that are written for a series of XilinxVirtex and physical hardware is not needed for this purpose.

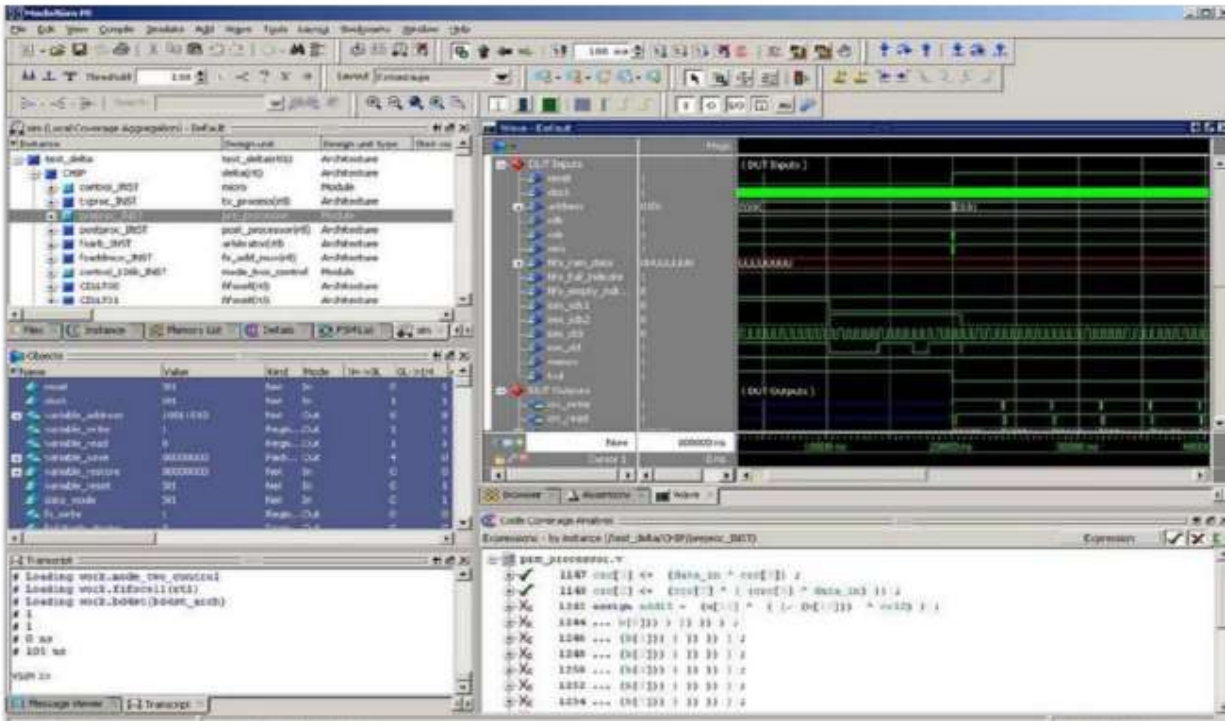

**Figure 2: ModelSim Environment**

(Source: Accessed from Dissertation)

ModelSim has a future use with Simulink or MATLAB and it is active for linking ModelSim. This linking refers to the co-simulation interface that is bidirectional and used to create a bridge between ModelSim and Simulink. Numerical simulation tools are provided by MATLAB while ModelSim offers effective tools for verifying the implementation of hardware along with timing features for the design (Jiang *et al*., 2016). ModelSim consists of the process of selecting a design that is defective by intelligently debugging the environment of engineering. debug environment of ModelSim shows the designed data for analysing and debugging all languages. From this perspective, it can be said that ModelSim has diverse features that make it proactive in this technological era. The core feature is to unify the diverse languages for easy and effective performances along with quick time for debugging. It is such an effective tool that supports to system Verilog, Verilog for handling verification for sophisticated design. Coverage of advanced code is another feature of ModelSim and analysing tools are highly active in coverage quickly. Facilities of debug environment are used from context of debug of post-sim along with interactive session. ModelSim includes a powerful waveform that impacts on easy analysis of bugs and differences.

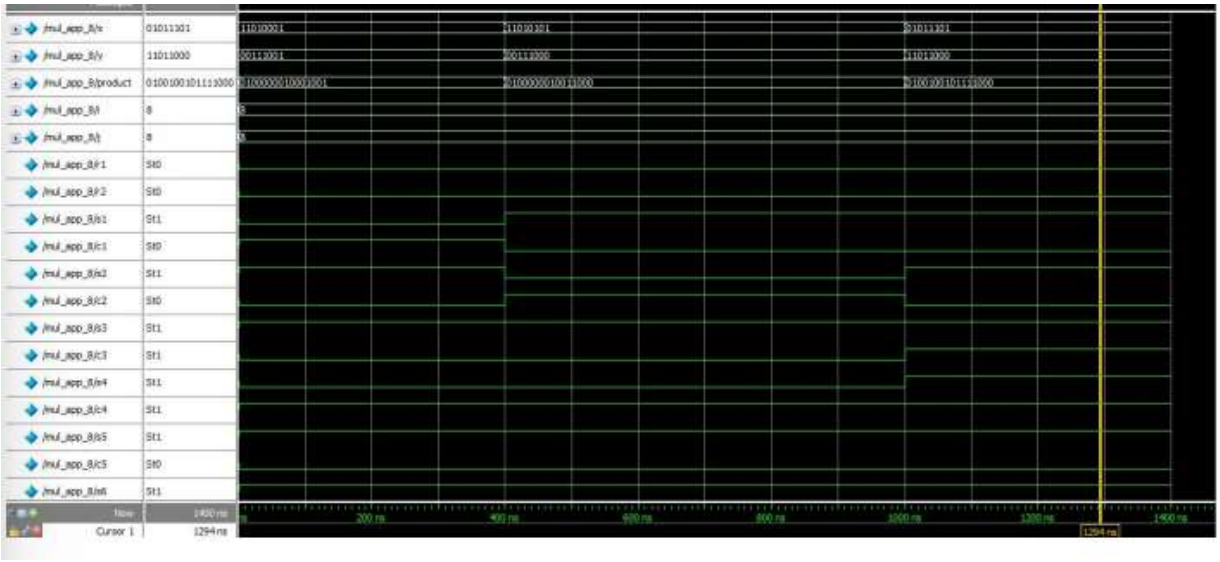

**Figure 3: Simulation result of the proposed multiplier**

(Source: Accessed from Dissertation)

From figure 3, it is noticed that X and Y mean multiplier input and multiplicand and the result of multipliers is PRODUCT. Based on it, R1 and R2 refer to the output of the OR tree indicating that for signifying lower weight the OR tree is used while the n:2 compressor is used for middle weight. AC6G 4:2 compressor indicates higher accuracy and it is used in higher weight of significance.

## Conclusion

From the above context, it can be said the design of 8*8 indicates the lower power that is highly efficient. AC6G compressor plates a significant role in achieving higher accuracy. Reduction of power consumption is the ultimate motto of approximate multipliers and it provides accuracy with less power consumption. It is noticed that approximate composers with high efficiency achieve lower distortion of power having a strong focus on providing accurate results. From this context, approximate computing has become popular in this modern era and it facilitates the various algorithms of signal processing such as multimedia, machine learning, and Digital Signal Processing. From the discussion, it is identified that applications of error-tolerant are included with approximate multipliers and it has developed day to day for functioning better. Adders and multipliers along with these applications become involved in processing digital signals and images. Moreover, it can be concluded that the impact of approximate multipliers plays a remarkable role in providing accurate output by consuming less power.

## References

Ahmadinejad, M., Moaiyeri, M.H. and Sabetzadeh, F., 2019. Energy and area efficient imprecise compressors for approximate multiplication at nanoscale. *AEU-International Journal of Electronics and Communications, 110*, p.152859.

Barbareschi, M., Barone, S. and Mazzocca, N., 2021. Advancing synthesis of decision tree-based multiple classifier systems: an approximate computing case study. *Knowledge and Information Systems, 63*, pp.1577-1596.

Chippa, V.K., Chakradhar, S.T., Roy, K. and Raghunathan, A., 2013, May. Analysis and characterization of inherent application resilience for approximate computing. In Proceedings of the 50th Annual Design Automation Conference (pp. 1-9).

Christianto, H., Lee, G.K.K., Jair, Z.W., Kasim, H. and Rajan, D., 2022. Smart interpretable model (SIM) enabling subject matter experts in rule generation. Expert Systems with Applications, 207, p.117945.

Ejtahed, S.A.H. and Timarchi, S., 2022. Efficient approximate multiplier based on a new 1-gate approximate compressor. *Circuits, Systems, and Signal Processing*, pp.1-20.

Ha, M. and Lee, S., 2017. Multipliers with approximate 4–2 compressors and error recovery modules. *IEEE Embedded Systems Letters, 10*(1), pp.6-9.

Jain, G., Jain, M. and Gupta, G., 2017, September. Design of radix-4, 16, 32 approx booth multiplier using error tolerant application. In 2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO) (pp. 314-320). IEEE.

Jiang, H., Liu, C., Maheshwari, N., Lombardi, F. and Han, J., 2016, July. A comparative evaluation of approximate multipliers. In 2016 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH) (pp. 191-196). IEEE.

Jothin, R., Mohamed, M.P. and Vasanthanayaki, C., 2020. High performance compact energy efficient error tolerant adders and multipliers for 16-bit image processing applications. *Microprocessors and Microsystems, 78*, p.103237.

Kong, L., Niu, Q. and Yang, P., 2023. Design and Implementation of UART Based on Verilog HDL. *Highlights in Science, Engineering and Technology*, 38, pp.949-955.

Nojehdeh, M.E. and Altun, M., 2020. Systematic synthesis of approximate adders and multipliers with accurate error calculations. *Integration, 70*, pp.99-107.

Singh, D. and Chandel, R., 2020. Register-Transfer-Level Design for Application-Specific Integrated Circuits. *Nanoscale VLSI: Devices, Circuits and Applications*, pp.295-319.

Yousefifeshki, F., Li, H. and Khomh, F., 2023. Studying the challenges of developing hardware description language programs. *Information and Software Technology, 159*, p.107196.