CHAPTER 6

CONCLUSION

6.1 INTRODUCTION

This chapter summarizes the thesis and indicates possible directions of future work. The main purpose of the reengineering process in the VLSI and real time design is to provide a measure of the quality and reliability of a modern system. It is essential for improving the performance of the migrated system. Several approaches are available in the literature for improving the performance of new systems by reducing the interference, the amount of time required and to increase the fault coverage. The problems suggested and solutions available from the literature are carefully considered and the issues are identified for the proposed work. For further improvements in the performance of reengineering, new approaches are proposed experiments are carried out and the results are compared with the existing approaches to verify the effectiveness of these methods.

6.2 CONTRIBUTION OF LEGACY SYSTEM EVOLUTION

The objective of this work was to evaluate the legacy system. The goal was to convert the legacy CPP programs to Java programs without any errors. Models proposed in the research work aimed at the conversion of the legacy system to the modern system without any logical error or execution error in the migrated system. In this present research work, three models for legacy system evolution namely Semantic Role Abstract Syntax Graph
(SRASG), Automatic Specification Evaluator (ASE) and Reengineering Legacy to Modern (RL2M) System were proposed. It was observed that the time consumed by the RL2M is comparatively less when compare with the other two systems. The number of error free code conversion is higher in RL2M with improvement of up to 43.47% compared to SRASG and ASE Model. However the average execution time increased in RL2M by 3.8% which is not significant compared to the accuracy in the code translation. The additional execution time in RL2M is due to the introduction of the One Time Checker (OTC) where additional tokenization with pattern matching is involved.

The techniques proposed overcome many of the issues faced in the tokenization based code conversion by proposing automated techniques which is capable of analysis of problem areas in conversion and overcoming the same. The focus of this work shows that the proposed techniques can be used for large interrelated modules.

6.3 DIRECTIONS FOR FUTURE WORK

As an extension of this thesis work, additional research can be carried out in continuation the proposed research work. This work can be extended as follows:

The future work proposes the extension of OTC and RL2M where the exclusive functions in legacy system are to be mapped and converted to the equivalent target system then to work as an executable target system itself without bugs. The main advantage of this proposed work is the OTC can be integrated with any reengineering process and it is virtual to end user with respect to the application. It may be the extension of the transformation of highly tedious legacy systems with minimum time requirements. The process
is much more efficient and give considerable improvement when the modules grouped together as a single unit in the target compiler itself.

This work proposes the converted Java code can be used for a VLSI application. Since a typical VLSI application involves many critical operations, Java can be used for running the VLSI application and this will act as a target system. The implementation of this application can be carried out as the future work.