B. TECH. PROJECT REPORT

On

Internship at Mentor Graphics

BY

Kunal Sikri

DISCIPLINE OF COMPUTER SCIENCE ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

NOVEMBER 2018

Internship at Mentor Graphics

A PROJECT REPORT

Submitted in partial fulfillment of the
requirements for the award of the degrees

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE ENGINEERING

Submitted by:

Kunal Sikri

Guided by:

Dr.Abhishek Srivastava ,Associate Professor

INDIAN INSTITUTE OF TECHNOLOGY INDORE

November,2018

CANDIDATE'S DECLARATION

I hereby declare that the project entitled "Internship at Mentor Graphics" submitted in partial fulfillment for the award of the degree of Bachelor of Technology in 'Computer Science Engineering',completed under the supervision of Dr.Abhishek Srivastava, Associate Professor, Computer Science Engineering,IIT Indore is an authentic work.
Further, I declare that I have not submitted this work for the award of any other degree . elsewhere .

Signature and name of the student(s) with date

_____

CERTIFICATE by BTP Guide(s)

It is certified that the above statement made by the students is correct to the best of my/our knowledge.

Signature of BTP Guide(s) with dates and their designation

PREFACE

This report on “Internship at Mentor Graphics" is prepared under the guidance of Dr.Abhishek Srivastava.

Through this report I have tried to give a detailed overview of my work at Mentor Graphics.I was part of the DVT Questa Optimisation Team.Most of the work was related to making test cases and automated test suites and verification.I have also made few scripts for automation required as per requirement and also added my option in their tool for extracting design related information from top level info to module specific information.

Kunal Sikri
B.Tech. IV Year
Discipline of Computer Science Engineering
IIT Indore

ACKNOWLEDGEMENTS

We wish to thank Dr.Abhishek Srivastava for his kind support and valuable guidance.

It is their help and support, due to which we became able to complete the design and technical report.

Without their support this report would not have been possible.

Kunal Sikri
B.Tech. IV Year
Discipline of Computer Science Engineering
IIT Indore

ABSTRACT

I had a great learning experience during my 6 months internship period at Mentor Graphics, Noida.I was part of the DVT Questa Optimisation team.Most of my work was related to testing and verification.I made several test suites for various kinds of optimisations that my team did. Some of them were in SystemVerilog and some mixture of SystemVerilog and VHDL.I made some scripts in python required for debugging purpose,testing,automation,extraction of module specific information from log files.

In my last assignment,I added an option in their tool for extracting design related information from top level info to module specific information.Option was added at both phases(optimization and simulation) that generate different kind of reports and then dump them in a directory, and integrated it with my script which can use it to make a database from all reports and extract module specific information required upon querying using script.

TABLE OF CONTENTS

WEEKLY REPORT

WEEK - 1:

23/05/2018 - 25/05/2018 -> Studied basic concepts of verilog and SystemVerilog

WEEK - 2:

28/05/2018 -> Continued studying

29/05/2018 - 01/06/2018 -> Started working with Sarthak Sir on optimization using module inling

Made various test cases based on :

01)Different data type

02)Multiple dimensional array

03)Mix pack unpacked array

04)Negative test case - array with different length,

05)Negative test case - array with different dimension

06)Multiple module instance using generate and array instance

07)Made a small python script that searches for a pattern 'Inlining' and 'optimizing module' and returns the count and module name as required .

WEEK - 3:

04/06/2018 -> day off (outstation)

05/06/2018 - 07/06/2018 -> Started working with Ashish Sir on optimization across verilog-systemVerilog boundary.Made various test cases based on :

01)Different data type

02)Different port direction

03)Using Struct,enum,multidimensional array

04)Using mix pack unpacked array

05)Using parameterized module

06)Using dynamic array

07)Using associative array(both integer and string index)

08)Negative test case-different number of array elements

09)Negative test case-different dimension of array (mix pack unpacked)

10)Negative test case-different dimension of array (unpacked)

11)Using queue

08/06/2018 -> Started working with Sarthak sir .Making python scripts for some automation

Scripts tasks include:

1.Find only module names, unique and sorted

2.Take 2 files of module names, new and ref. Find names which are in new and not in ref file.

3.Prepends +acc=p+<module_name> and name the output as acc.f.

4.Call perl run with binary search on acc.f

WEEK - 4:

11/06/2018 - 12/06/2018 -> Made last script for single possibility as well as multiple possibilities and currently working on testing it .It's working logically.

13/06/2018 - 15/06/2018 -> Analysing different test cases for number of modules in list for n=3,4,5,6 Exhaustive testing for each case , checking the output as well as tracing the path followed to reach the solution in order to verify proper working. Found That algo takes more number of runs in some cases which can be optimized and also found some bugs. Removed all bugs and moved on to optimising part . So we thought of changing the algorithm , i.e Running the binary search for full list but if number of modules in the list fall below some specific n(which can be set by the user from command prompt)Then run more optimised and efficient algorithm.

WEEK - 5:

18/06/2018-20/06/2018 -> Started working with Naresh and Sujeet Sir on coalesced optimisation.
Made various test cases in Bata format :

01.For 4 expressions
02.For 6 expressions
03.Using various operators
04.Multidimensional array (packed)
05.Using constant in expression
06.Passing value to an instance using concatenation
07.Using large array range
08.Giving value to all index of array, to some parts of array
09.Multiple range of indices of array specified with different operators
10.Using little endian or big endian format array specification
11.Corner test cases (like only upper/lower indices specified)
12.Using different operators in a same expression

21/06/2018 -> Took build path from Sujeet Sir and tried running the test cases but coalesced optimisation was not happening . So i switched back on implementing the more optimised version of the last script. After Implementing tested exhaustively all test cases for n=3,4,5 and then analyzed the performance in all cases and also verified the solution in all cases.

22/06/2018 -> Made some changes to all coalesced test cases and then first ran it with earlier build to generate reference files and then ran with modified build to compare with it.

Found test cases which were failing optimisation as desired.

WEEK - 6:

25/06/2018-27/06/2018 -> Final Script combining everything into one script that just takes two vopt logs.Tested and made detailed documentation/readme for that Script.

28/06/2018-29/06/2018 -> Started working with Sharad Sir . Made test cases for continuous assign opt..in which always block having some hanging logic are transformed to assign.

WEEK - 7:

02/07/2018-05/07/2018 -> Started working with Vinay Sir on expanding Structures optimization.
Made several test cases based on:
01)Three levels of structure hierarchy
02)Different data types in structures
03)Used enum in a structure
04)Packed and unpacked structures
05)Used String
06)Used simpler as well as complex expressions
07)Used different port directions and ranges
08)Used mixed packed unpacked array
09)Used array of structure object in another structure
10)Select Equivalent vector

11)Assigning value to complete structures(packed/ unpacked) in one go and Then verifying tree created in post_opt.

06/07/2018 -> Started working with Sharad Sir on a script.It aimed at dumping optimization related data to be dumped at one place under one directory and then query to extract module specific information.

WEEK - 8:

09/07/2018-13/07/2018 -> Made several scripts for single or multiple .v/.sv file taking as input.And made another set of script which can be used for a design.

1)First set of script that can be run on single/multiple .v/.sv files

Generate all reports in ./Reports/ directory and a summary file that has info where post_opt files are and also containing info related To queries made related to specific module for viewing later in time.There is a script query.py which can be called later in time if u want To again query module related info without compiling and running again .It will automatically extract info from reports and append the Queries made in summary.txt which can be referred later.+ point - Script is quite general and one can easily add new Options and change name of files and directories.It automatically searches for previous post_opt and

pre_opt files and place them in a temporary directory when u are running script and restore them later.

2)Second set of Script can be run on designs.You have to run the script with command to run design , it will automatically extract vlog,vopt,vsim commands from log files created most recently and then make a new makefile named ks_makefile with appended options for vopt and vsim .
This makefile is run by another script to generate reports in ./Reports/ Directory and then you can query module related information which will be displayed with colors(headings and contents properly distinguishable) on terminal as well as in summary.txt.Moreover you can query later in time using another script without running makefile again.It automatically extracts info from Reports

WEEK - 9:

16/07/2018 -> outstation

17/07/2018-20/07/2018 -> Worked with Sarthak and made test cases relating to Math functions

WEEK - 10:

23/07/2018-24/07/2018 -> P4force(version control system) setup

25/07/2018 -> outstation

26/07/2018-27/07/2018 -> Started working with Abhinav .. Studying from LRM concept Of classes,Interfaces,interface class,abstract class etc..

WEEK - 11:

30/07/2018-31/07/2018 -> Continued studying and based upon that made several practice test cases to understand the concepts..Also studied clocking block from LRM and made test case for that to understand it well.

01/08/2018-03/08/2018 -> Started making test cases for mixed design simulation based on the knowledge and constructs known till then

WEEK - 12:

06/08/2018-09/08/2018 -> Studied VHDL in detail and made more complex mixed design test cases with the purpose of finding out the type and number of constructs and functionality supported across mixed design boundary.
Studied bind construct and used in test cases.Made some test cases based upon sharing of User defined data types defined either in SV or Vhdl (using -mixedsvvh).
Studied ch-9 of questa manual and examples of regression suit to understand more about mixed designs and based upon that knowledge made a few complex mixed design test cases.

10/08/2018 -> Started working with Naveen on testing of new topics added in sv 1800-2012 LRM..Topics were :

1)Remove restrictions on NBA assignments to class members

2)Parameterized Data Types : Parameterized data types are implemented through the use of type definitions in parameterized classes

3)Interface classes : A set of classes may be created that can be viewed as all having a common set of behaviors. Such a common set of behaviors may be created using interface classes.

4)Typed constructor calls : A typed constructor call shall create and initialize a new object of the specified type.

5)Parameterized Tasks and Functions : The way to implement parameterized subroutines is through the use of static methods in parameterized classes.

WEEK - 13:

13/08/2018-15/08/2018 -> sick leave

17/08/2018

16/08/2018 -> Made few test cases and reported some bugs

Some of the bugs were:

1) Incorrect failure while compilation when using typedef within nested classes where the outer class is parametrized(QSIM-14400)

2) Not able to resolve signals in clocking block coming from vhdl entity instantiated in sv (QSIM-51634)

3) Issue while making polymorphic assignments using interface class(QSIM-51635)

4) Issue : when using parameterized structure inside parameterized class having an opaque super class (QSIM-51658)

WEEK - 14:

20/08/2018-24/08/2018 -> Made complex designs based upon topics newly added in sv 1800-2012 LRM..

WEEK - 15:

27/08/2018-28/08/2018 -> Created test cases involving typedefs of dynamic array/ queue/associative array.

1.Used these typedefs outside classes.

2.Tried all variations like nested classes/opaque superclass with above typedefs

3.Also tried it with parameterized virtual interfaces

4.Made test cases where parameterized function returning packed/unpacked/associative array and queue declared using typedef .

Submitted an excel sheet dictating what's in each test case

30/08/2018-31/08/2018 -> Started working with Sarthak on clock detection related optimization and sscanf function.

WEEK - 16:

03/09/2018-06/08/2018	->	Made test cases for sscanf functions and some test cases related to clock detection and propagation .
07/09/2018	->	Went to Indore for test

WEEK - 17:

10/09/2018-14/09/2018	->	Made complex test cases for clock detection optimization and ran build on perf test suite having 180 designs

WEEK - 18:

17/09/2018-21/09/2018	->	Out of 180,72 designs failed,extracted errors from all failed designs into a text file (size = 2MB) ,and did exhaustive analysis of errors.

WEEK - 19:

24/09/2018	->	Made a summary of the most common patterns (related to clocks)with their approximate frequencies.
25/09/2018-28/09/2018	->	Outstation (Indore) for placement test

WEEK - 20:

01/10/2018-05/10/2018	->	Worked on another kind of errors for clock detection and propagation problem and did exhaustive analysis of errors from failed designs

WEEK - 21:

08/10/2018	->	Continued last analysis
09/10/2018-10/10/2018	->	Summarising the results derived from analysis of all three kinds of errors in clock detection and propagation and clocked processes present in inhouse perf design suite.
11/10/2018	->	Making Presentation
12/10/2018	->	on leave(because presentation on skype scheduled)

WEEK - 22:

15/10/2018	->	started working with Sharad for adding an option in tool for report generation.
16/10/2018-19/10/2018	->	on leave(went home)

WEEK - 23:

22/10/2018-23/10/2018	->	working on src code for adding option in vopt
24/10/2018-26/10/2018	->	on leave (went indore for placements)

WEEK - 24:

29/10/2018-30/10/2018	->	on leave (went indore for placements)
31/10/2018	->	new office inauguration
01/11/2018	->	Diwali party
02/11/2018	->	Holiday (shifting to new office)

WEEK - 25:

05/11/2018-06/11/2018	->	Continued work on adding option in tool
07/11/2018-08/11/2018	->	Diwali Holiday
09/11/2018	->	Completed adding option in vopt and tested it and integrated with script i made earlier for querying and extracting info.

WEEK - 26:

12/11/2018-13/11/2018	->	Worked on adding option in simulator for report generation
14/11/2018	->	Last day at office.Completed work on tool with all testing and integration with script.Uploaded the changes and made a documentation for the same.

Assignment - 1

Module Inlining Optimization

Made various test cases based on :

01)Different data type

02)Multiple dimensional array

03)Mix pack unpacked array

04)Negative test case - array with different length

05)Negative test case - array with different dimension

06)Multiple module instance using generate and array instance

07)Arrays of same length but different range specification

Assignment - 2

Made a python Script that takes log file as input and return the count of modules inlined under each module along with names of inlined module under each parent module

Assignment - 3

Module Inlining Optimization across verilog - systemVerilog boundary

01)Different data type

02)Different port direction

03)Using Struct,enum,multidimensional array

04)Using mix pack unpacked array

05)Using parameterized module

06)Using dynamic array

07)Using associative array(both integer and string index)

08)Negative test case-different number of array elements

09)Negative test case-different dimension of array(mix pack unpacked)

10)Negative test case-different dimension of array(unpacked)

11)Using queue

Assignment - 4

Made few(around 5-6) python scripts for automation . The main aim was to find out the smallest set of modules which were leading to test failure because of their inability to concur with module inlining optimization.

Main concern was the performance of algorithm . I developed the algorithm and then exhaustively tested for various cases. Founding was that algorithm did pretty well for large number of modules but not for small n. So I developed a more efficient algorithm for small n and then connected both. The result was that we the system follows algorithm 1 until it reaches a threshold value and then changes to algorithm 2 . The threshold value can be set by user and there was one more option to directly start with the spitting rather than on initial list if we know it is not the smallest set

Assignment - 5

RTL Transformation (Coalesced Optimization)

Example:

reg a[0:2],b[0:2],c[0:2];

Assign a[1]=b[1] & c[1]

Assign a[0]=b[0] & c[0] Transformed to Assign a=b&c

Assign a[2]=b[2] & c[2]

Made test cases :-

01.For 4 expressions

02.For 6 expressions

03.Using various operators

04.Multidimensional array (packed)

05.Using constant in expression

06.Passing value to an instance using concatenation

07.Using large array range

08.Giving value to all index of array, to some parts of array

09.Multiple range of indices of array specified with different operators

10.Using little endian or big endian format array specification

11.Corner test cases (like only upper/lower indices specified)

12.Using different operators in a same expression

Assignment - 6

Expand Decl Structures

Made test cases :-

01)Three levels of structure hierarchy

02)Different data types in structures

03)Used enum in a structure

04)Packed and unpacked structures

05)Used String

06)Used simpler as well as complex expressions

07)Used different port directions and ranges

08)Used mixed packed unpacked array

09)Used array of structure object in another structure

10)Select Equivalent vector

11)Assigning value to complete structures(packed/unpacked) in one go and Then verifying tree created in post_optimization phase.

Assignment - 7

Made a python script that aimed at dumping Optimisation related data to be dumped at one place under one Directory and a database is also created which can then query to extract module specific information.Whatever that you do during query session gets stored in a file (suppose in file abc.txt) so that it is available for later viewing without the need to run program again.

Moreover you can run query program any time and the information collected will be appended to abc.txt.The script can be run on designs with makefiles or on simpler designs with just one or two files.

Assignment - 8

Made test cases for checking the optimisations related to various math functions like cos,sin,sinh,tanh,ln,log base 10,exponential,pow,abs,floor,ceil,sqrt,asin,acos,atan

Assignment - 9

Studied concepts from Language Reference Manual and made several test cases related to topics as defined below in order to verify whether they are being implemented properly in accordance with the LRM. The topics are :

01)Class
02)Nested Class
03)Interface
04)Interface Class
05)Abstract Class
06)Clocking Block
07)Inheritance and virtual functions

Assignment - 10

01)Studied VHDL in detail and made more complex mixed design test cases with the purpose of finding out the type and no of and constructs functionality supported across mixed design boundary.
02)Studied bind construct and used it in mixed design test cases
03)Made some test cases based upon sharing of User defined data types defined either in SV or Vhdl.

Assignment - 11

Made test cases for testing of new topics added in SystemVerilog 1800-2012 Language Reference Manual . The topics were :

01)Removing restrictions on Non blocking assignments to class members.

02)Parameterised Data types

03)Interface Classes

04)Typed Constructor Calls

05)Parameterised tasks and Functions

01)Created various tests involving typedefs of dynamic array/queue/Associative array.

02)Used these typedefs outside classes.

03)Tried all variations like nested classes/opaque superclass with Above typedefs.

04)Also tried it with parameterized interfaces

05)Made test cases where parameterised function returning packed/unpacked/ associative array and queue declared using typedef.

Bugs!!

I was able to find 4 bugs in the tool which are as follows:

1)Incorrect failure while compilation when using typedef within nested classes where the outer class is parametrized.

2)Not able to resolve signals in clocking block coming from VHDL entity instantiated in SV.

3)Issue while making polymorphic assignments using interface class.

4)Issue when using parameterised structure inside parameterised class having an opaque super class.

Assignment - 12

Analysis of Clock Detection and Propagation and Clocked Processes in perf design suite

I made some test cases related to clock detection and propagation problem ,but main task was to analyse the errors after running build on in house perf design suite.

Out of 180 designs ,72 designs failed, extracted errors from all failed designs and did exhaustive analysis of errors in order to find common pattern and categorise them along with their frequencies.

Assignment - 13

Adding an option in tool at both optimisation and simulation phase for generating reports of several kind and then querying to extract module specific information from reports and log files.

I added my option in tool at both phases that generate different kind of reports and then dump them in a directory, and integrated it with my script which can use it to make a database from all reports and extract module specific information required upon querying using script,and the information is also dumped in a file and can be retrieved anytime later for viewing.