CHAPTER 3

METHODOLOGY

3.1 Analysis of the Conventional High Speed 8-bits x 8-bits Wallace Tree Multiplier

The design analysis starts with the analysis of the elementary algorithm for multiplication by Wallace Tree multiplier. Figure 3.1 shows the algorithm for 8-bits x 8-bits multiplication performs by Wallace Tree multiplier. There are five stages to go through, to complete the multiplication process. Each stage used half adders and full adders that are denoted by the red circle for the 1 bit half adder and the blue circle for the 1-bit full adder.

Firstly, we have to reduce the partial products using half adders and full adders that are combined to build a carry-save adder (CSA) until there were just two rows of partial products left. Next, we add the remaining two rows by using a fast carry-propagate adder. For this project, ripple-carry adder (RCA) is used, to get the final product of the two operands multiplication.

Secondly, the schematic of the conventional 8-bits x 8-bits high speed Wallace Tree multiplier is design by referring to the algorithm. Figure 3.2 shows the block diagram for the conventional high speed 8-bits x 8-bits Wallace Tree multiplier.

After that, the Verilog source code for the conventional Wallace Tree multiplier is designed. We have to design the source code for the sub-elements of the design first, that are the source code for D flip-flop, a 1-bit half adder, and a 1-bit full adder. All these sub-element, have to be simulate and check their functionality before we combine them to built the conventional Wallace Tree multiplier. Half adders and full adders are used as a combination to form a carry-save adder (CSA). D flip-flops are used at the overall input and output of the design for the purpose of timing analysis in the next step.

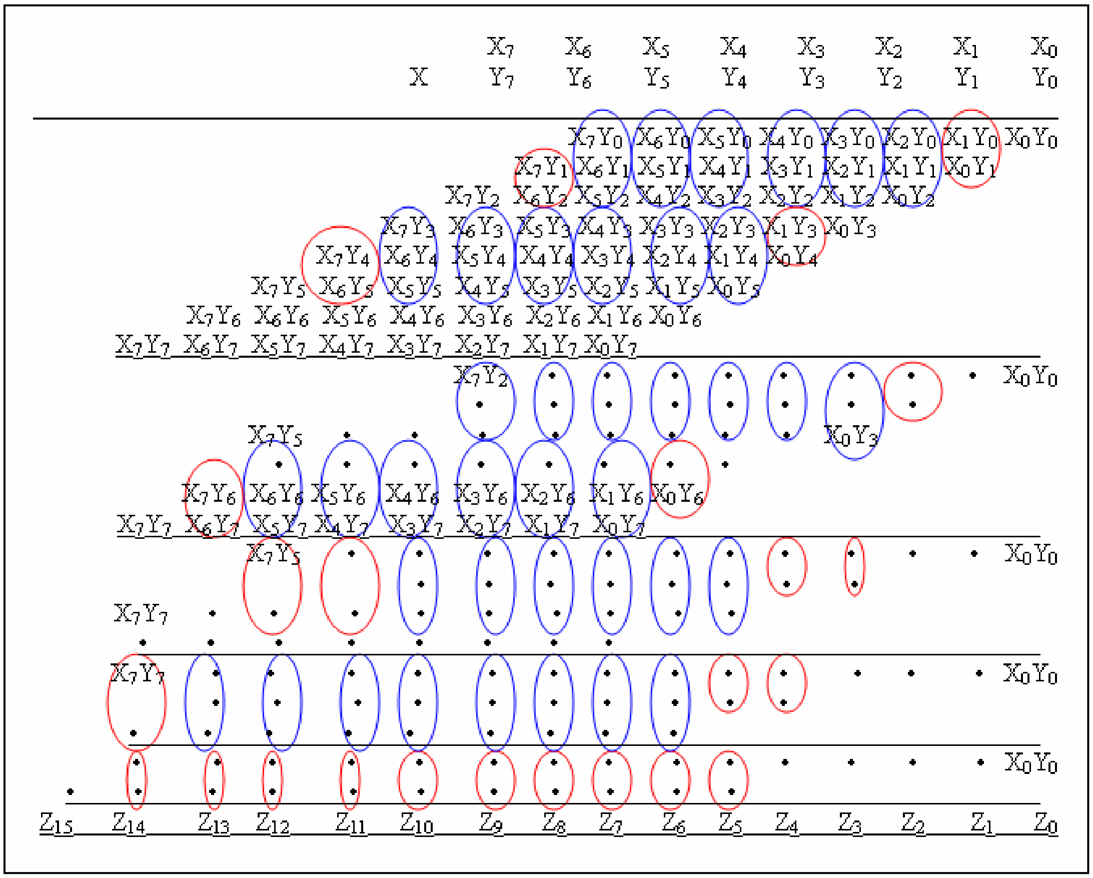

X_7 X_6 X_5 X_4 X_3 X_2 X_1 X_0
X Y_7 Y_6 Y_5 Y_4 Y_3 Y_2 Y_1 Y_0
X_7Y_0 X_6Y_0 X_5Y_0 X_4Y_0 X_3Y_0 X_2Y_0 X_1Y_0 X_0Y_0
X_7Y_1 X_6Y_1 X_5Y_1 X_4Y_1 X_3Y_1 X_2Y_1 X_1Y_1 X_0Y_1
X_7Y_2 X_6Y_2 X_5Y_2 X_4Y_2 X_3Y_2 X_2Y_2 X_1Y_2 X_0Y_2
X_7Y_3 X_6Y_3 X_5Y_3 X_4Y_3 X_3Y_3 X_2Y_3 X_1Y_3 X_0Y_3
X_7Y_4 X_6Y_4 X_5Y_4 X_4Y_4 X_3Y_4 X_2Y_4 X_1Y_4 X_0Y_4
X_7Y_5 X_6Y_5 X_5Y_5 X_4Y_5 X_3Y_5 X_2Y_5 X_1Y_5 X_0Y_5
X_7Y_6 X_6Y_6 X_5Y_6 X_4Y_6 X_3Y_6 X_2Y_6 X_1Y_6 X_0Y_6
X_7Y_7 X_6Y_7 X_5Y_7 X_4Y_7 X_3Y_7 X_2Y_7 X_1Y_7 X_0Y_7
X_7Y_2
X_0Y_0
X_0Y_3
X_7Y_5
X_7Y_6 X_6Y_6 X_5Y_6 X_4Y_6 X_3Y_6 X_2Y_6 X_1Y_6 X_0Y_6
X_7Y_7 X_6Y_7 X_5Y_7 X_4Y_7 X_3Y_7 X_2Y_7 X_1Y_7 X_0Y_7
X_7Y_5
X_0Y_0
X_7Y_7
X_7Y_7
X_0Y_0
X_0Y_0
Z_{15} Z_{14} Z_{13} Z_{12} Z_{11} Z_{10} Z_9 Z_8 Z_7 Z_6 Z_5 Z_4 Z_3 Z_2 Z_1 Z_0

Figure 3.1: The algorithm for 8-bits x 8-bits multiplication performs by Wallace Tree multiplier

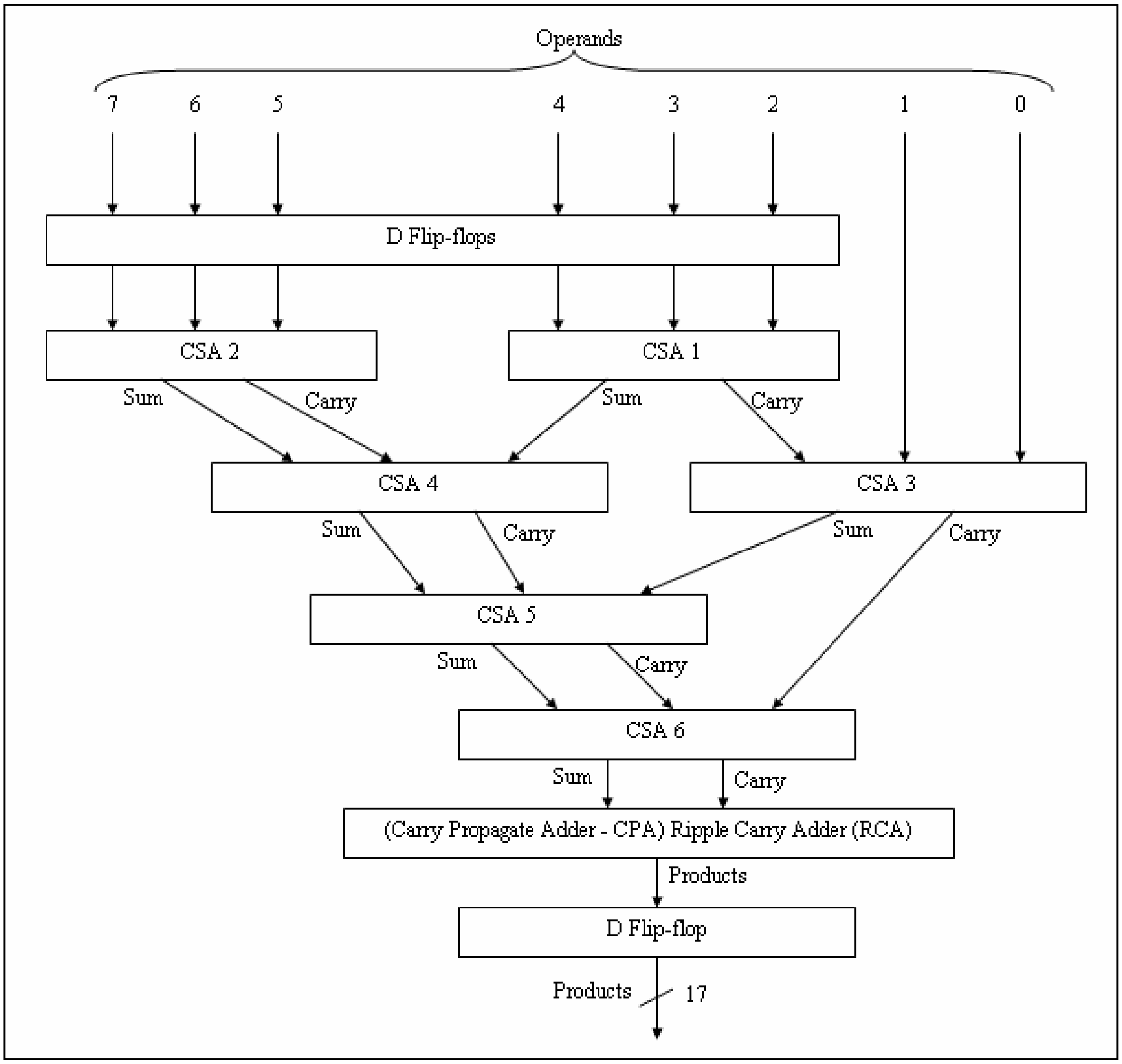


Figure 3.2: The block diagram for the conventional high speed 8-bits x 8-bits Wallace Tree multiplier

Then, when the Verilog source code of the multiplier has been design, we must simulate and check its functionality. If it is functioning correctly, we could proceed to the next step, which is to determine the maximum speed and time that the multiplier takes to complete a single multiplication process. It could be done by using the timing analyzer in the Altera Quartus II software.

3.2 Analysis of the Pipelining High Speed 8-bits x 8-bits Wallace Tree Multiplier

There are some methods that could be used to increase the speed of a multiplier such as reduce the number of partial products, used fast adder for fast addition of the partial products and by doing pipelining.

For the purpose of making the Wallace Tree multiplier faster than it usually be as in the conventional, pipelining method is choose for this project. Pipelining is a very well known technique for accelerating the execution of successive identical problems. In the pipelining method, the sub-elements circuits of the Wallace Tree multiplier are partition into a few small segments to make it operate independently on consecutive sets of operands. The executions of several successive operations overlap, and the rate at which results are produced is considerably higher than the conventional design [3]. To allow pipelining, D flip-flops, which are called pipelined stages, are added between adjacent stages in the multiplier circuit, so that when a stage works on one set of operands, the preceding stage can work on the next set of operands [3]. In this project, there are four stages of D flip-flop used to increase the Wallace Tree multiplier speed. Figure 3.3 shows the block diagram for the pipelining high speed 8-bits x 8-bits Wallace Tree multiplier.

Next, the Verilog source code for the pipelining Wallace Tree multiplier is designed. Same as in conventional design, we have to design the source code for the sub-elements of the design first, that are the source code for D flip-flop, a 1-bit pipelining half adder, and a 1-bit pipelining full adder. Then, all these sub-element have to be simulated and check their functionality before we combine them to build the pipelining Wallace Tree multiplier.

After completed the Verilog source code of the multiplier, we repeat the same step as in the conventional design that are simulation and check its functionality. If it is functioning correctly, we could proceed to the next step, which is to determine the maximum speed and time that the multiplier takes to complete a single multiplication process. It could be done by using the timing analyzer tools.

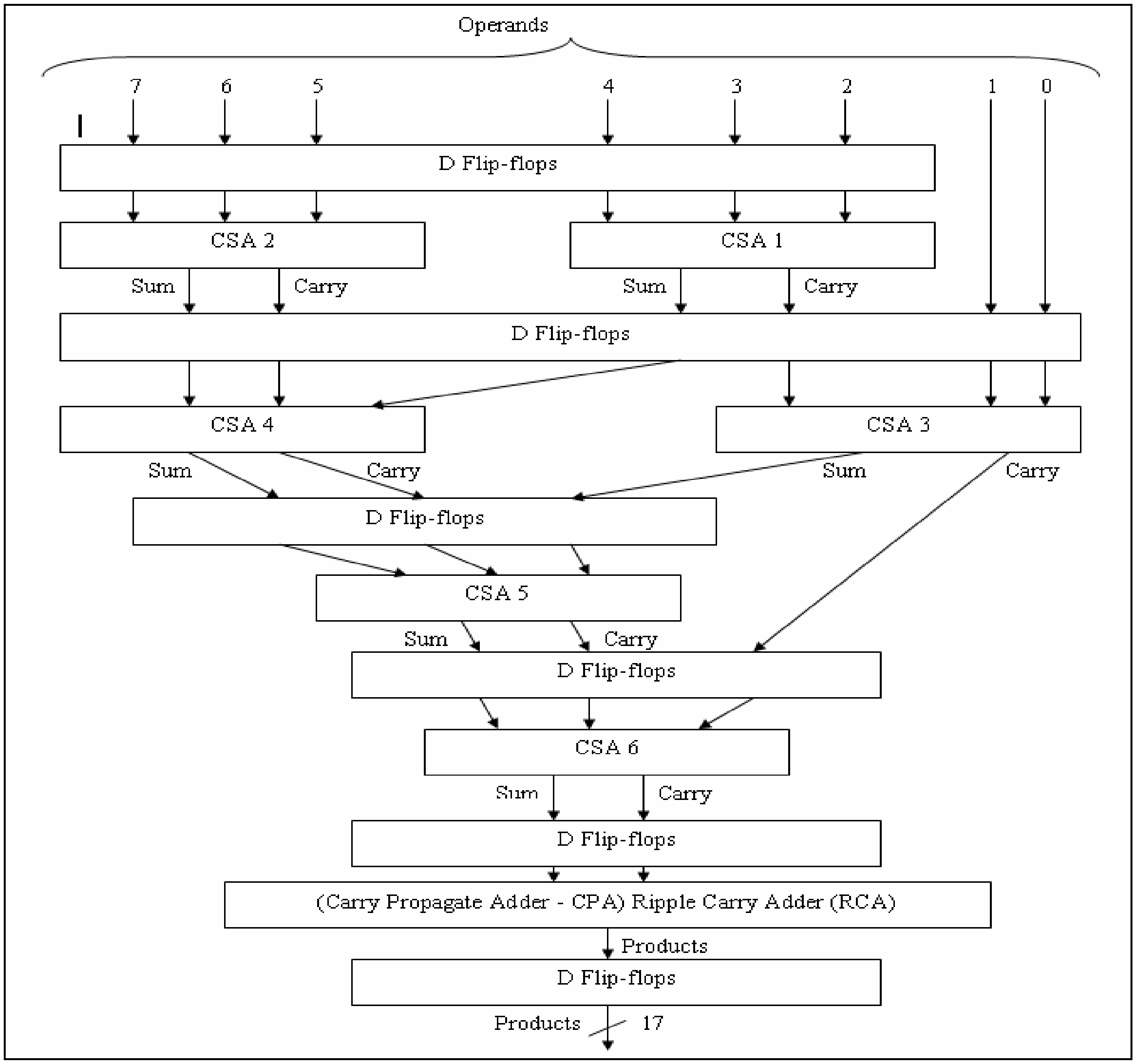


Figure 3.3: The block diagram for the high speed 8-bits x 8-bits Wallace Tree multiplier using 4-levels of pipelining

Finally, when we had completed both of the multiplier design, conventional and pipelining high-speed 8-bits x 8-bits Wallace Tree multiplier, we could compare their maximum speed and time that both of the multipliers takes to complete a single multiplication process. This will be the result of this project to show that Wallace Tree multiplier is one of the fastest multiplier.

3.3 Quartus II Software

The Quartus II system includes full support for all of the popular methods of entering a description of the desired circuit into a Computer Aided Design (CAD) system. CAD software makes it easy to implement a desired logic circuit by using a programmable logic device, such as a field-programmable gate array (FPGA) chip. A typical FPGA CAD flow is illustrated in Figure 3.4.

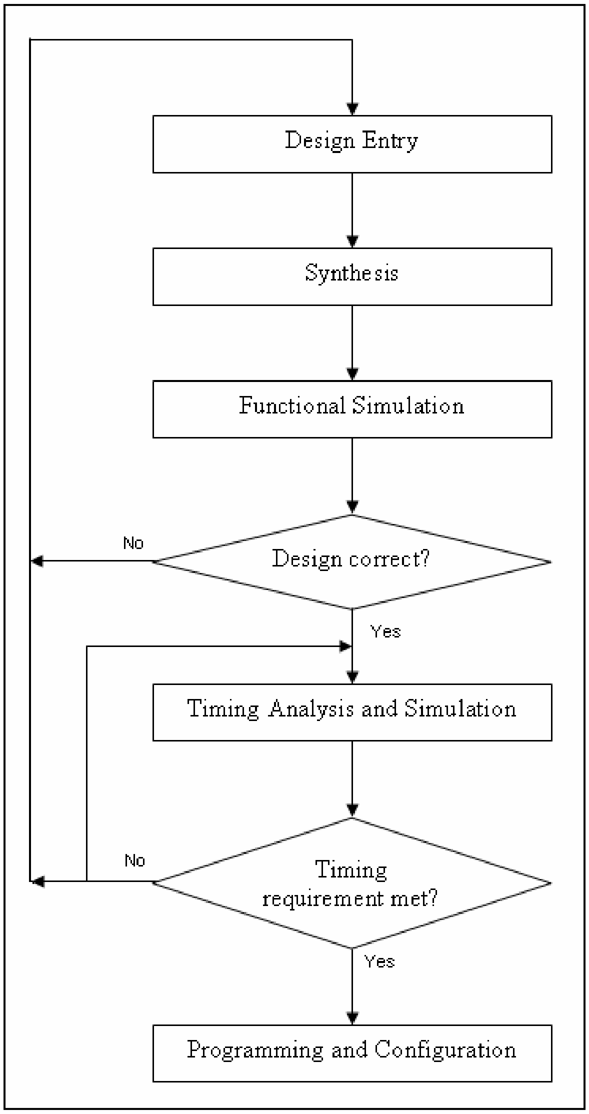


Figure 3.4: Typical CAD flow

The CAD flow involves the following steps:

- **Design Entry** – the desired circuit is specified either by means of a schematic diagram, or by using a hardware description language, such as Verilog or VHDL. For this project, the design entry used is the Verilog HDL.

- **Synthesis –** the entered design is synthesized into a circuit that consists of the logic elements (LEs) provided in the FPGA chip.

- **Functional Simulation –** the synthesized circuit is tested to verify its functional correctness; this simulation does not take into account any timing issues.

- **Timing Analysis –** propagation delays along the various paths in the fitted circuit are analyzed to provide an indication of the expected performance of the circuit.

- **Timing Simulation** – the fitted circuit is tested to verify both its functional correctness and timing.

- **Programming and Configuration –** the designed circuit is implemented in a physical FPGA chip by programming the configuration switches that configure the LEs and establish the required wiring connections.

3.4 Project Methodology

Phase 1: Research and data collecting plus complete the project proposal.

- Research through books.
- Research through internet.
- Research through journals.

Phase 2: Designing the schematic for the multiplier.

- Identify circuit characteristics for the Wallace Tree multiplier.
- Design the schematic for the conventional high speed Wallace Tree Multiplier.

Phase 3: Designing the source code for the multiplier.

- Using the Altera Quartus II 5.0 Web Edition Full to design the source code for the conventional and pipelining high speed Wallace Tree multiplier using Verilog, simulate and check whether the expected result is archive or not.
- Apply the design on the FPGA.

Phase 4: Writing the Final Year Project (FYP) report.

- Start doing some report about the project.
- Complete the FYP report draft.
- Complete the final presentation slides.

Phase 5: Complete the FYP report and make final presentation of the project.

- Submit the FYP report draft.
- Final FYP presentation.
- Do correction on the FYP report and submit.

Figure 3.5 shows the flow chart of the project progress.

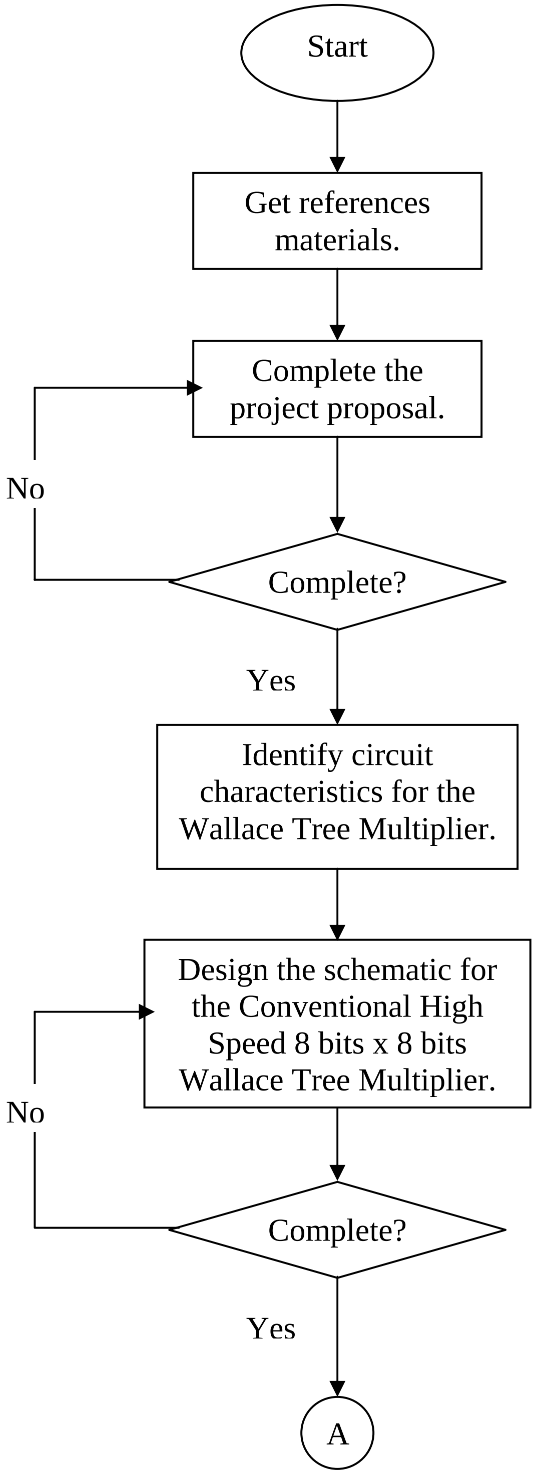
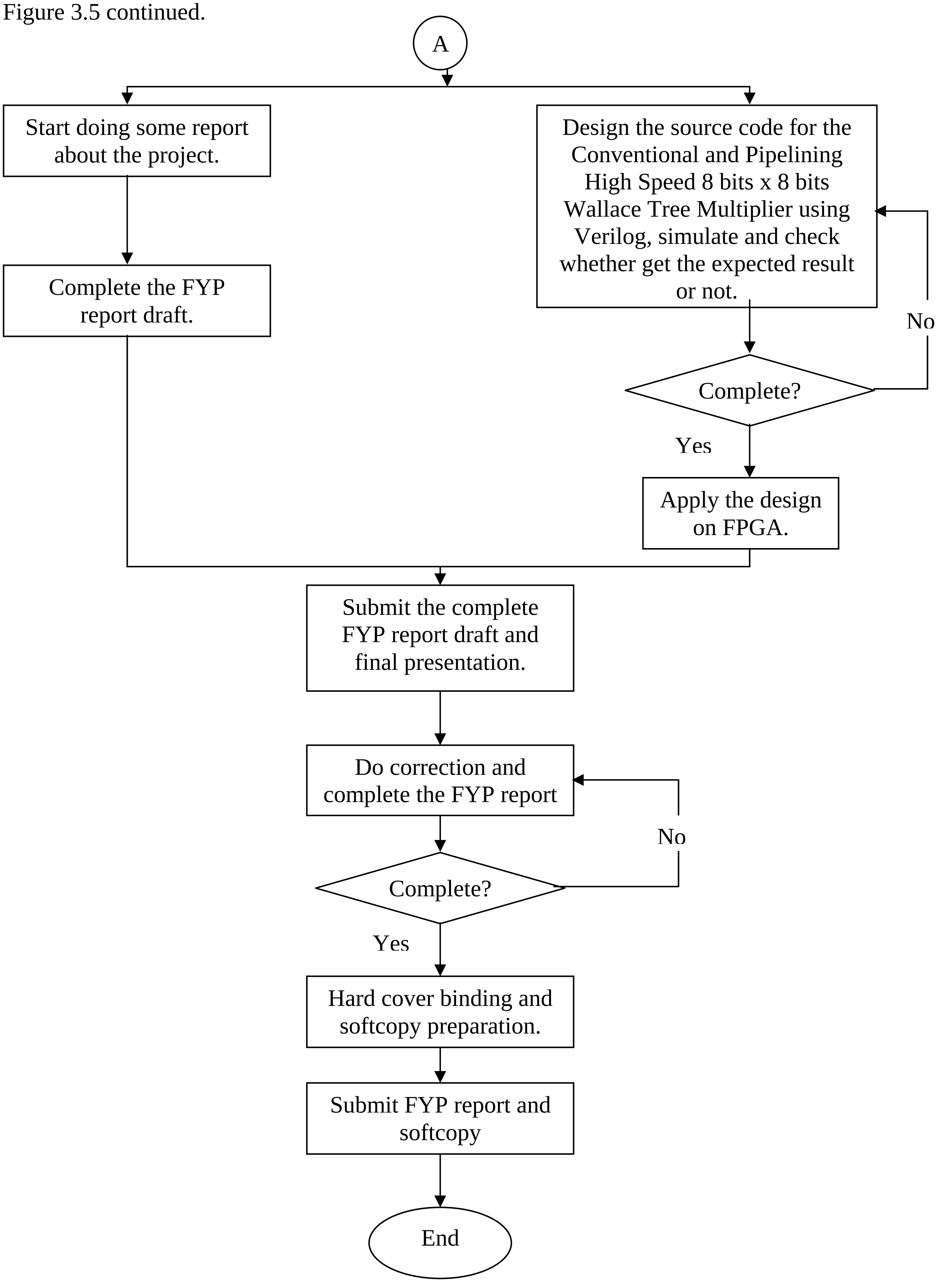

Start
Get references materials.
Complete the project proposal.
No
Complete?
Yes
Identify circuit characteristics for the Wallace Tree Multiplier.
Design the schematic for the Conventional High Speed 8 bits x 8 bits Wallace Tree Multiplier.
No
Complete?
Yes
A

Figure 3.5: Flow chart of the project progress

Figure 3.5 continued.


A
Start doing some report about the project.
Design the source code for the Conventional and Pipelining High Speed 8 bits x 8 bits Wallace Tree Multiplier using Verilog, simulate and check whether get the expected result or not.
Complete the FYP report draft.
No
Complete?
Yes
Apply the design on FPGA.
Submit the complete FYP report draft and final presentation.
Do correction and complete the FYP report
No
Complete?
Yes
Hard cover binding and softcopy preparation.
Submit FYP report and softcopy
End