CS M152A Project 2 Report

Melody Chen

May 3, 2020

1 Introduction and Requirement

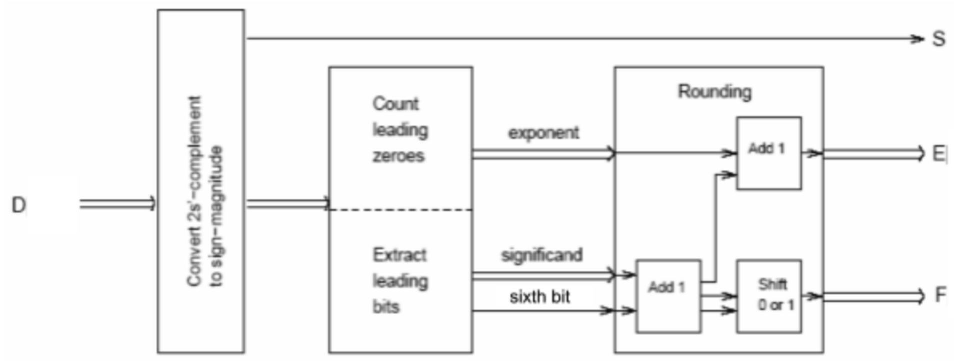
The focus of this lab is for students to learn how to use the Xilinx ISE software to design and test a floating point converter. To implement the floating point converter, we build a combinational circuit that that receives a 13-bit linear encoding of an analog signal and outputs a compounded 9-bit Floating Point(FP) Representation. For the output, we use a simplified Floating Point Representation consisting of 1-Bit Sign Representation(S), a 3-Bit Exponent(E), and a 5-Bit Significand(F). The value represented by our simplified FP can be calculated by:

$$V = (-1)^S * F * 2^E$$

Part of the focus of this lab is to take care of rounding for cases we cannot accurately represent in our FP format and to ensure that we have correct FP representation for all edge cases. More details about rounding will be explained in later parts of the report. After implementing the floating point conversion module, we design a test bench and simulate the behaviors of varying inputs to show the accuracy of our module.

2 Design Description

For the design and implementation of the floating point converter circuit, I followed the overall block diagram provided in the project description, shown below:



Overall Block Diagram of Circuit Design

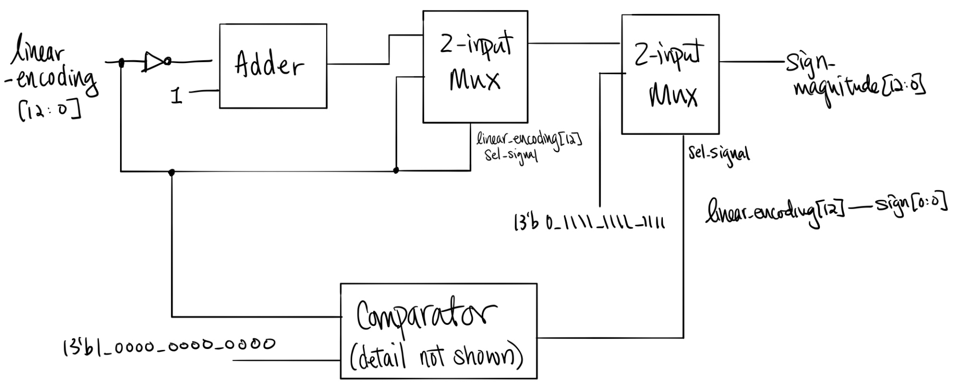
I designed 3 sub-modules, `convertToSignMagnitude`, `countLeadingZeroExtractBits`, and `round`, corresponding to each sub-block in the diagram above. My top module `FPCVT` uses the three sub-modules to convert the `13'b` linear encoding into our desired compounded `9'b` FP representation. `FPCVT` takes in 1 input `[12:0] D` that is the 13 bit linear encoding and outputs `S` which represents sign of FP, `[2:0] E`, which represents the exponent, and `[4:0] F` which represents the significand.

a. **Sub-Module 1:** `convertToSignMagnitude`

The purpose of this module is to covert the 13-bit 2's complement input to sign-magnitude representation. In sign-magnitude representation, positive numbers are unchanged, and negative number are shown as their positive counter-part.

To implement this in Verilog, I designed my module to take in one input `[12:0] linear_encoding` and output `sign` and `[12:0] sign_magnitude`. I used if statements in an `always` block to check if our input is a positive or negative number and to check if we have edge case where input is the most negative number. I dealt with the most negative number by setting `sign_magnitude` to the most positive number we can represent in 13'b.

A hand drawn schematic of my `convertToSignMagnitude` with different components is shown below. The overall idea and inputs/outputs of the comparator is shown below, detailed schematic is omitted for clarity.



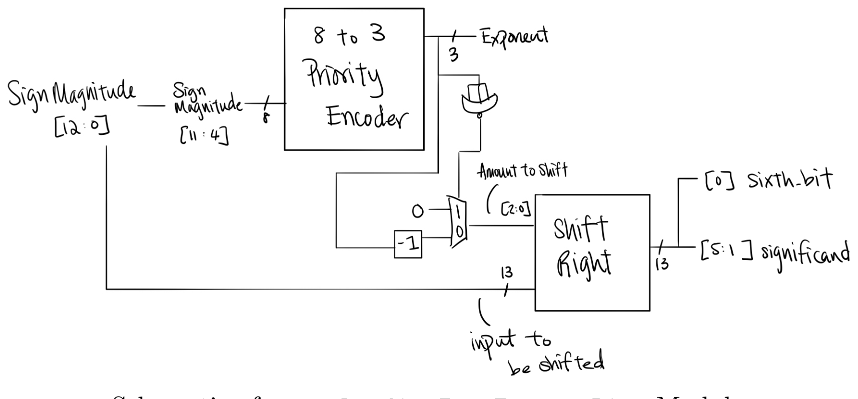
Schematic of `convertToSignMagnitude` Module

This module will output final output `S` in our compounded 9-bit FP representation, and provide input necessary for our next module `countLeadingZeroExtractBits`.

b. **Sub-Module 2:** `countLeadingZeroExtractBits`
The purpose of this module is to perform the basic linear FP conversion. The exponent is computed based on counting the number of leading zeroes. The significand is the 5 bits after the leading zeroes and sixth bit is the bit after the significand.

To implement this in Verilog, I designed my module to take in `[12:0] signMagnitude` computed from `convertToSignMagnitude` and my module outputs `[2:0] exponent`, `[4:0] significand`, and `sixth_bit`. To extract the exponent, I used the idea of a priority encoder. To implement the priority encoder in Verilog, I used an `always` block with if statements testing in which bit the first 1 appears. Depending on where the leading zeroes end, I extract the `significand` and the `sixth_bit`.

A hand drawn schematic of `countLeadingZeroExtractBits` with different components, such as Priority Encoder, Shift Right, and subtractor, is shown below. The Shift Right operator is used in the diagram below to extract the outputs, but it is not actually used in the Verilog code as `significand` can be directly extracted with the corresponding index from `signMagnitude` in each if statement in the `always` block.



Schematic of `countLeadingZeroExtractBits` Module

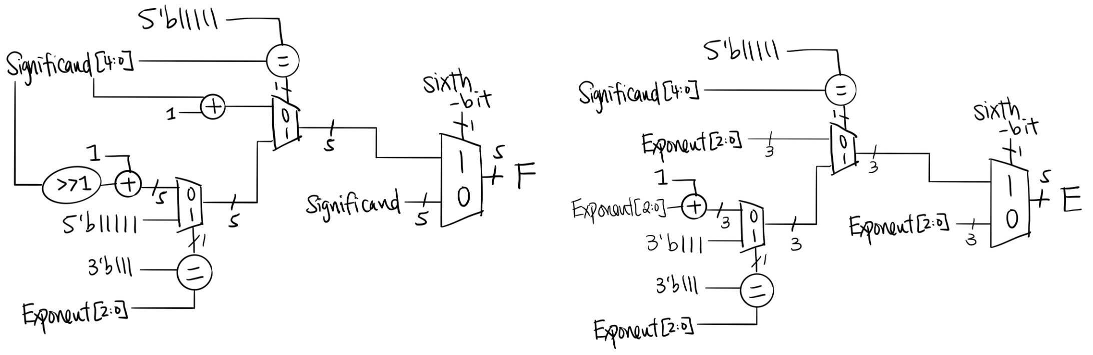
This module will output 3 outputs `exponent`, `significand`, and `sixth_bit` used as inputs to the third sub-module `round`.

c. **Sub-Module 3:** `round`

The third sub-module performs rounding of the Floating Point Representation. The rounding depends on the `sixth_bit` outputted by `countLeadingZeroExtractBits` sub-module. If `sixth_bit = 1`, we increment `significand` by 1. If `significand` overflows with this addition, we increase `exponent` by 1 and shift `significand` to the right by 1.

To implement this sub-module in Verilog, I designed my module to take in 3 inputs `[2:0] exponent`, `[4:0] significand`, and `sixth_bit`, and outputs `[2:0] E` and `[4:0] F`. I used an `always` block sensitive to all change, and used if statements within `always` block to check for how to proceed with calculating `E` and `F` depending on values of the three inputs. The if statements allow me to detect and take care of edge cases where either the `exponent` or `significand` will overflow if 1 is added.

A hand drawn schematic of `round` is shown below, which uses components such as comparators, shift right, adders, and mux. Mux is used to represent the if statements in our Verilog code.



Schematic of `round` Module

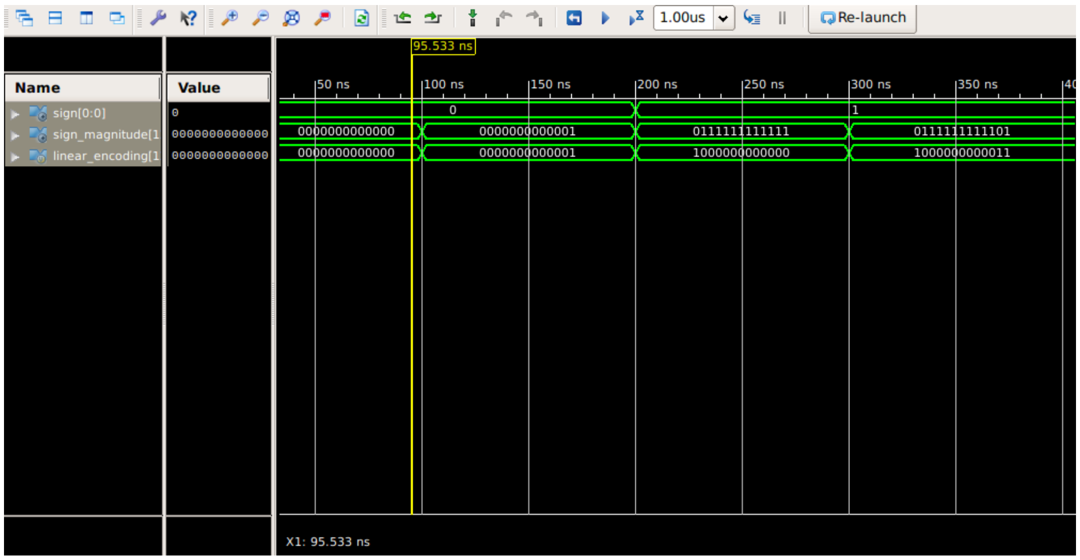
The two outputs of this sub-module `E` and `F` corresponds to final outputs of our floating point conversion circuits. `E` represents the exponent while `F` represents the significand.

3 Simulation Documentation

To test whether the floating point converter we implemented outputs the correct conversion, I first simulated the behavior of each sub-module and then finally simulated behavior of our combined top module.

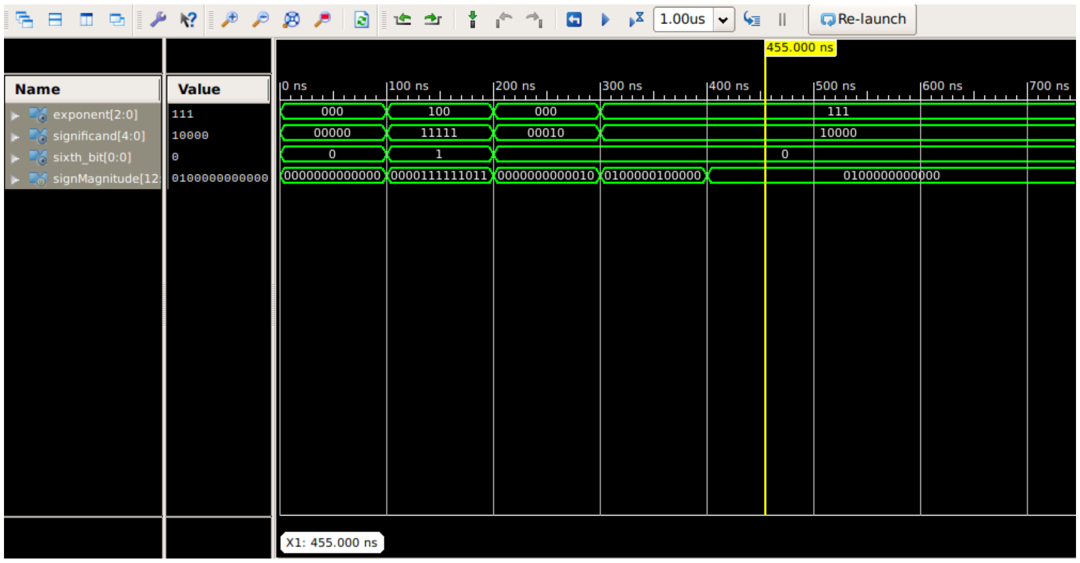
a. **Sub-Module 1:** `convertToSignMagnitude`

For the `convertToSignMagnitude` module, in my test bench I tested different values of `linear_encoding [12:0]` and observed how they change the value of output `sign` and sign_magnitude. For example, when `linear_encoding [12:0] = 0`, according to our implementation, the output sign should be 0 and the output sign_magnitude should remain unchanged as 0 is a positive number. As shown in the first 100ns of the simulation waveform, when `linear_encoding [12:0] = 0`, sign = 0 and sign_magnitude = 0. At 200ns, I tested the edge case where `linear_encoding [12:0] = 1_0000_0000_000`, the most negative number we can represent in 13 bits. The output we're looking for is the most positive number we can represent `13'b 0_1111_1111_1111`. As shown in the simulation diagram below, indeed sign = 1 and `sign_magnitude = 13'b0_1111_1111_1111`. At 300ns, we test yet another large magnitude negative number, and see that the output is correctly represented by value of sign and sign_magnitude.



Simulation Waveform for `convertToSignMagnitude`

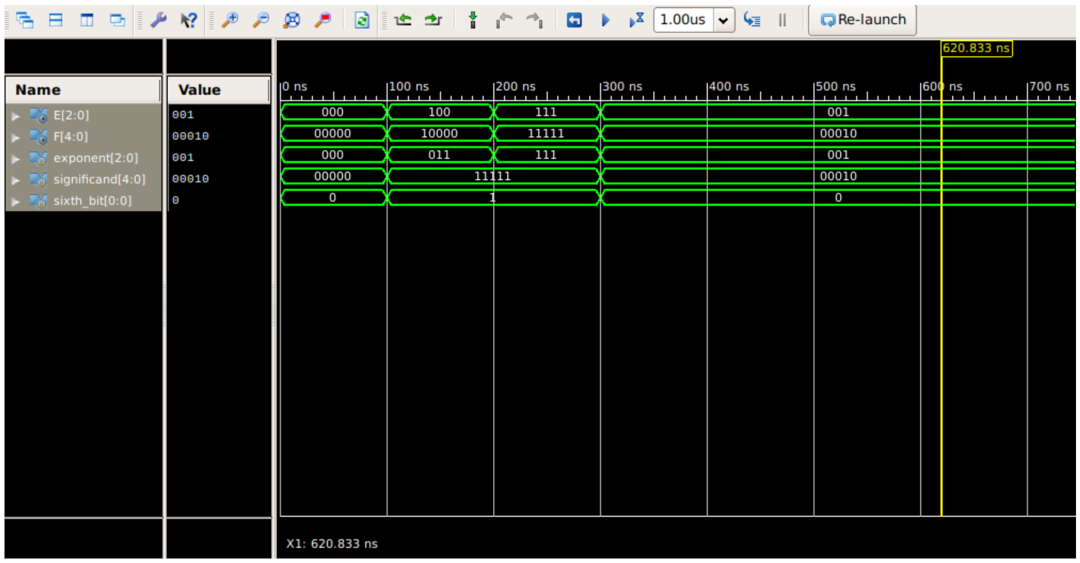
b. **Sub-Module 2:** `countLeadingZeroExtractBits`
For this sub-module, my test bench uses different value of the `signMagnitude` input and observes how the three outputs `significand`, `sixth_bit`, and `exponent` changes.



Simulation Waveform for `countLeadingZeroExtractBits`

From the waveform diagram above, we can see that at 0ns, `signMagnitude = 0`, so all of our outputs should be zero, which matches the output above. At 100ns, we set `signMagnitude = 13'b0_0000_1111_1101`. The number of leading zeroes in this case is 5, so `exponent` should have value of 3. The 5 bits after leading zeroes are `5'b11111`. The bit after the 5 bits is `1`. This matches the output shown above. At 300ns, I tested whether our implementation can correctly extract the highest value `3'b111` when `signMagnitude = 13'b0_1000_0010_0000`. At 300ns, `exponent = 111` and all the other output is correct as well, showing that our sub-module can correctly extract `significand`, `sixth_bit`, and `exponent`.

c. **Sub-Module 3:** `round`
For this sub-module, my test bench uses different value of the 3 inputs `significand`, `sixth_bit`, and `exponent` and observes how the two outputs `E`, and `F` changes.

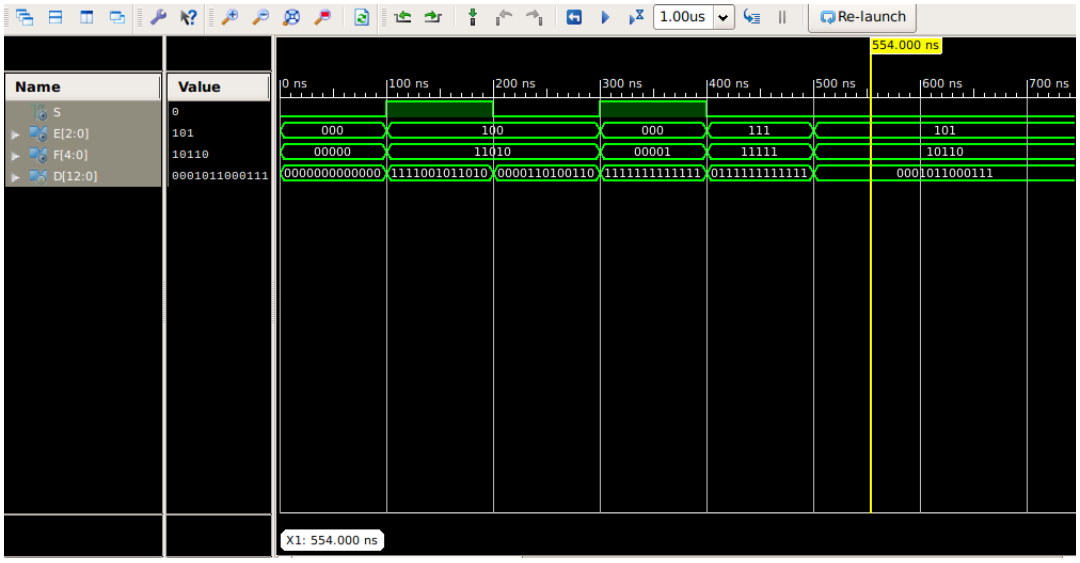


Simulation Waveform for `round`

From the waveform diagram above, we can see that at 0ns, all our inputs equal 0, showing that there is no need for rounding, so all of our outputs should be zero, which is true. At 100ns, we test for the edge case where we need to round up and `significand` will overflow, so `exponent` needs to be incremented. From diagram above, we clearly observe output `E` to be one higher than input `exponent` and output `F` is incremented by 1 and correctly shifted to the right by 1. At 200ns, we test for the next edge case where both `significand` and `exponent` overflows. In this case, we expect output `E` and `F` to be the largest it can be. Indeed, `E` and `F` are all 1s. At 300ns, we test the case where there is no need to round up as `sixth_bit = 0`. `E` and `F` retains value of `significand` and `exponent` as shown in the diagram above.

d. **Top Module:** `FPCVT`
To test to see if my top module `FPCVT` which connects the above 3 sub-modules correctly converts a linear encoding into our 9-bit compound FP representation, I used different values for input `D` and observes the change in output `S, E, D`.

The wave form from my test bench is shown below:



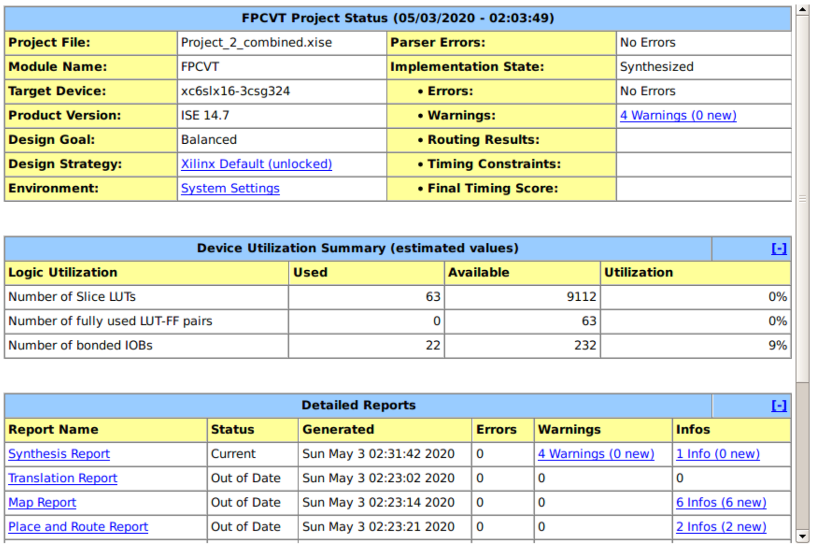
Simulation Waveform for `FPCVT`

At 100ns, the input `D = 13’b1_1110_0101_1010` Since it is negative number, we see that `S = 1`. Once converted to 2’s complement `13’b0_0001_1010_0110`, we find that it has 4 leading zeroes and there is no need to round. The value of `E` should then be `3’b100`, which matches what is shown in the diagram. Since there is no need to round `F` should be `11010`, which correctly matches output in diagram. At 300ns, we test the edge case where `D` is all 1s. We correctly observe `S = 1` and `E = 0`, and `F = 1`. At 400ns, we test the edge case where `D` is the most positive number. We then correctly observe `S = 0`, and both `E` and `D` are all 1s.

Some bugs I found during simulation include forgetting to increment 1 for certain roundings. I was able to fix that by trying to come up with the correct output without help of the module but with my own logic, and found the bug.

Synthesis and Implementation Report included at end of document.

4 ISE Design Overview Summary Report



FPCVT Project Status (05/03/2020 - 02:03:49)			
Project File:	Project_2_combined.xise	**Parser Errors:**	No Errors
Module Name:	FPCVT	**Implementation State:**	Synthesized
Target Device:	xc6slx16-3csg324	**• Errors:**	No Errors
Product Version:	ISE 14.7	**• Warnings:**	4 Warnings (0 new)
Design Goal:	Balanced	**• Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	**• Timing Constraints:**	
Environment:	System Settings	**• Final Timing Score:**	

Device Utilization Summary (estimated values) [-]

Logic Utilization	Used	Available	Utilization
Number of Slice LUTs	63	9112	0%
Number of fully used LUT-FF pairs	0	63	0%
Number of bonded IOBs	22	232	9%

Detailed Reports [-]

Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sun May 3 02:31:42 2020	0	4 Warnings (0 new)	1 Info (0 new)
Translation Report	Out of Date	Sun May 3 02:23:02 2020	0	0	0
Map Report	Out of Date	Sun May 3 02:23:14 2020	0	0	6 Infos (6 new)
Place and Route Report	Out of Date	Sun May 3 02:23:21 2020	0	0	2 Infos (2 new)

Design Summary

My ISE Design Overview Summary Report is shown above. As you can see my implementation has no errors. There are a few warnings, but they are taken care of by my condition checking, so those cases of overflow will never occur. The resources I used can also be observed from the summary report above.

5 Conclusion

In this project, I designed and implemented the floating point converter that converts a `13’b` linear encoding of an analog signal into a compounded 9-but FP Representation. I used a lot of the knowledge learned during lecture in previous weeks to formulate my design and implement the code in Verilog. I split the task into 3 sub-modules and combined them together. Each sub-module has a smaller task so it is easier for me to design smaller individual task and then combine them together.

Some difficulties I encounter include deciding between when to use `wire` and `register` which often gave me bugs when I tried to synthesized. I was able to deal with this problem by looking at slides provided by TA previously about coding combinational circuits in Verilog. I also had difficulty understanding the project description but with the help and explanation of the TA, I was able to grasp full understanding.

6 Reports

6.1 Synthesis Report

```
Release 14.7 - xst P.20131013 (lin64)
Copyright (c) 1995-2013 Xilinx, Inc.  All rights reserved.
-->
Parameter TMPDIR set to xst/projnav.tmp


Total REAL time to Xst completion: 0.00 secs
Total CPU time to Xst completion: 0.10 secs

-->
Parameter xsthdpdir set to xst


Total REAL time to Xst completion: 0.00 secs
Total CPU time to Xst completion: 0.10 secs

-->
Reading design: FPCVT.prj

TABLE OF CONTENTS
  1) Synthesis Options Summary
  2) HDL Parsing
  3) HDL Elaboration
  4) HDL Synthesis
       4.1) HDL Synthesis Report
  5) Advanced HDL Synthesis
       5.1) Advanced HDL Synthesis Report
  6) Low Level Synthesis
  7) Partition Report
  8) Design Summary
       8.1) Primitive and Black Box Usage
       8.2) Device utilization summary
       8.3) Partition Resource Summary
       8.4) Timing Report
            8.4.1) Clock Information
            8.4.2) Asynchronous Control Signals Information
            8.4.3) Timing Summary
            8.4.4) Timing Details
            8.4.5) Cross Clock Domains Report


=========================================================================
*                      Synthesis Options Summary                        *
=========================================================================
---- Source Parameters
Input File Name                    : "FPCVT.prj"
Ignore Synthesis Constraint File   : NO

---- Target Parameters
Output File Name                   : "FPCVT"
Output Format                      : NGC
Target Device                      : xc6slx16-3-csg324

---- Source Options
Top Module Name                    : FPCVT
```

```
Automatic FSM Extraction           : YES
FSM Encoding Algorithm             : Auto
Safe Implementation                : No
FSM Style                          : LUT
RAM Extraction                     : Yes
RAM Style                          : Auto
ROM Extraction                     : Yes
Shift Register Extraction          : YES
ROM Style                          : Auto
Resource Sharing                   : YES
Asynchronous To Synchronous        : NO
Shift Register Minimum Size        : 2
Use DSP Block                      : Auto
Automatic Register Balancing       : No

---- Target Options
LUT Combining                      : Auto
Reduce Control Sets                : Auto
Add IO Buffers                     : YES
Global Maximum Fanout              : 100000
Add Generic Clock Buffer(BUFG)     : 16
Register Duplication               : YES
Optimize Instantiated Primitives   : NO
Use Clock Enable                   : Auto
Use Synchronous Set                : Auto
Use Synchronous Reset              : Auto
Pack IO Registers into IOBs        : Auto
Equivalent register Removal        : YES

---- General Options
Optimization Goal                  : Speed
Optimization Effort                : 1
Power Reduction                    : NO
Keep Hierarchy                     : No
Netlist Hierarchy                  : As_Optimized
RTL Output                         : Yes
Global Optimization                : AllClockNets
Read Cores                         : YES
Write Timing Constraints           : NO
Cross Clock Analysis               : NO
Hierarchy Separator                : /
Bus Delimiter                      : <>
Case Specifier                     : Maintain
Slice Utilization Ratio            : 100
BRAM Utilization Ratio             : 100
DSP48 Utilization Ratio            : 100
Auto BRAM Packing                  : NO
Slice Utilization Ratio Delta      : 5

=========================================================================


=========================================================================
*                          HDL Parsing                                  *
=========================================================================
Analyzing Verilog file "/home/melody/152a/Project_2_combined/FPCVT.v" into library work
Parsing module <convertToSignMagnitude>.
Parsing module <countLeadingZeroExtractBits>.
Parsing module <round>.
Parsing module <FPCVT>.

=========================================================================
*                          HDL Elaboration                              *
=========================================================================

Elaborating module <FPCVT>.

Elaborating module <convertToSignMagnitude>.
WARNING:HDLCompiler:413 - "/home/melody/152a/Project_2_combined/FPCVT.v" Line 32: Result of 32-bit expression is tr

Elaborating module <countLeadingZeroExtractBits>.

Elaborating module <round>.
WARNING:HDLCompiler:413 - "/home/melody/152a/Project_2_combined/FPCVT.v" Line 111: Result of 6-bit expression is tr
WARNING:HDLCompiler:413 - "/home/melody/152a/Project_2_combined/FPCVT.v" Line 117: Result of 4-bit expression is tr

=========================================================================
*                           HDL Synthesis                               *
=========================================================================

Synthesizing Unit <FPCVT>.
    Related source file is "/home/melody/152a/Project_2_combined/FPCVT.v".
    Summary:
no macro.
Unit <FPCVT> synthesized.

Synthesizing Unit <convertToSignMagnitude>.
    Related source file is "/home/melody/152a/Project_2_combined/FPCVT.v".
    Found 32-bit adder for signal <n0010> created at line 32.
    Summary:
inferred   1 Adder/Subtractor(s).
inferred   2 Multiplexer(s).
Unit <convertToSignMagnitude> synthesized.

Synthesizing Unit <countLeadingZeroExtractBits>.
    Related source file is "/home/melody/152a/Project_2_combined/FPCVT.v".
WARNING:Xst:647 - Input <signMagnitude<12:12>> is never used. This port will be preserved and left unconnected if i
    Summary:
inferred  20 Multiplexer(s).
Unit <countLeadingZeroExtractBits> synthesized.

Synthesizing Unit <round>.
    Related source file is "/home/melody/152a/Project_2_combined/FPCVT.v".
    Found 5-bit adder for signal <significand[4]_GND_4_o_add_2_OUT> created at line 111.
    Found 3-bit adder for signal <exponent[2]_GND_4_o_add_4_OUT> created at line 117.
    Found 5-bit adder for signal <n0027> created at line 118.
    Summary:
inferred   2 Adder/Subtractor(s).
inferred   8 Multiplexer(s).
Unit <round> synthesized.

=========================================================================
HDL Synthesis Report

Macro Statistics
# Adders/Subtractors                                   : 3
 3-bit adder                                           : 1
 32-bit adder                                          : 1
 5-bit adder                                           : 1
# Multiplexers                                         : 30
 1-bit 2-to-1 multiplexer                              : 7
 13-bit 2-to-1 multiplexer                             : 2
 3-bit 2-to-1 multiplexer                              : 9
 5-bit 2-to-1 multiplexer                              : 12

=========================================================================
INFO:Xst:1767 - HDL ADVISOR - Resource sharing has identified that some arithmetic operations in this design can sh

=========================================================================
*                       Advanced HDL Synthesis                          *
=========================================================================


=========================================================================
Advanced HDL Synthesis Report

Macro Statistics
# Adders/Subtractors                                   : 3
 13-bit adder                                          : 1
 3-bit adder                                           : 1
 5-bit adder                                           : 1
# Multiplexers                                         : 30
 1-bit 2-to-1 multiplexer                              : 7
 13-bit 2-to-1 multiplexer                             : 2
 3-bit 2-to-1 multiplexer                              : 9
 5-bit 2-to-1 multiplexer                              : 12

=========================================================================

=========================================================================
*                         Low Level Synthesis                           *
=========================================================================

Optimizing unit <FPCVT> ...

Optimizing unit <convertToSignMagnitude> ...

Optimizing unit <round> ...

Mapping all equations...
Building and optimizing final netlist ...
Found area constraint ratio of 100 (+ 5) on block FPCVT, actual ratio is 1.

Final Macro Processing ...

=========================================================================
Final Register Report

Found no macro
=========================================================================

=========================================================================
*                           Partition Report                            *
=========================================================================

Partition Implementation Status
-------------------------------
```

```
  No Partitions were found in this design.

-------------------------------

=========================================================================
*                            Design Summary                             *
=========================================================================

Top Level Output File Name         : FPCVT.ngc

Primitive and Black Box Usage:
------------------------------
# BELS                             : 88
#      GND                         : 1
#      INV                         : 11
#      LUT1                        : 1
#      LUT2                        : 2
#      LUT4                        : 6
#      LUT5                        : 6
#      LUT6                        : 37
#      MUXCY                       : 11
#      VCC                         : 1
#      XORCY                       : 12
# IO Buffers                       : 22
#      IBUF                        : 13
#      OBUF                        : 9

Device utilization summary:
---------------------------

Selected Device : 6slx16csg324-3


Slice Logic Utilization:
 Number of Slice LUTs:                   63  out of   9112     0%
    Number used as Logic:                63  out of   9112     0%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:     63
   Number with an unused Flip Flop:      63  out of     63   100%
   Number with an unused LUT:             0  out of     63     0%
   Number of fully used LUT-FF pairs:     0  out of     63     0%
   Number of unique control sets:         0

IO Utilization:
 Number of IOs:                          22
 Number of bonded IOBs:                  22  out of    232     9%

Specific Feature Utilization:

---------------------------
Partition Resource Summary:
---------------------------

  No Partitions were found in this design.

---------------------------
```

```
=========================================================================
Timing Report

NOTE: THESE TIMING NUMBERS ARE ONLY A SYNTHESIS ESTIMATE.
      FOR ACCURATE TIMING INFORMATION PLEASE REFER TO THE TRACE REPORT
      GENERATED AFTER PLACE-and-ROUTE.

Clock Information:
------------------
No clock signals found in this design

Asynchronous Control Signals Information:
----------------------------------------
No asynchronous control signals found in this design

Timing Summary:
---------------
Speed Grade: -3

   Minimum period: No path found
   Minimum input arrival time before clock: No path found
   Maximum output required time after clock: No path found
   Maximum combinational path delay: 13.908ns

Timing Details:
---------------
All values displayed in nanoseconds (ns)

=========================================================================
Timing constraint: Default path analysis
  Total number of paths / destination ports: 24324 / 9
-------------------------------------------------------------------------
Delay:               13.908ns (Levels of Logic = 10)
  Source:            D<12> (PAD)
  Destination:       E<1> (PAD)

  Data Path: D<12> to E<1>
                                Gate     Net
    Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     IBUF:I->O            14   1.222   1.062  D_12_IBUF (D_12_IBUF)
     LUT6:I4->O           12   0.203   1.253  mod1/linear_encoding[12]_PWR_2_o_equal_1_o<0>1 (mod1/linear_encoding[
     LUT6:I1->O           18   0.203   1.394  mod1/Mmux_sign_magnitude121 (sign_magnitude<8>)
     LUT6:I1->O            1   0.203   0.944  mod2/Mmux_significand82 (mod2/Mmux_significand81)
     LUT6:I0->O            1   0.203   0.808  mod2/Mmux_significand85_SW0 (N14)
     LUT6:I3->O            6   0.205   0.745  mod2/Mmux_significand85 (significand<2>)
     LUT2:I1->O            2   0.205   0.721  mod3/Mmux_F5321 (mod3/Mmux_F532)
     LUT6:I4->O            2   0.203   0.981  mod3/Mmux_E221 (mod3/Mmux_E22)
     LUT6:I0->O            1   0.203   0.579  mod3/Mmux_E21 (E_1_OBUF)
     OBUF:I->O                 2.571          E_1_OBUF (E<1>)
    ----------------------------------------
    Total                     13.908ns (5.421ns logic, 8.487ns route)
                                       (39.0% logic, 61.0% route)

=========================================================================

Cross Clock Domains Report:
--------------------------
```

```
=========================================================================


Total REAL time to Xst completion: 6.00 secs
Total CPU time to Xst completion: 4.95 secs

-->


Total memory usage is 386792 kilobytes

Number of errors   :    0 (   0 filtered)
Number of warnings :    4 (   0 filtered)
Number of infos    :    1 (   0 filtered)
```

6.2 Implementation Report

```
Release 14.7 Map P.20131013 (lin64)
Xilinx Mapping Report File for Design 'FPCVT'

Design Information
------------------
Command Line   : map -intstyle ise -p xc6slx16-csg324-3 -w -logic_opt off -ol
high -t 1 -xt 0 -register_duplication off -r 4 -global_opt off -mt off -ir off
-pr off -lc off -power off -o FPCVT_map.ncd FPCVT.ngd FPCVT.pcf
Target Device  : xc6slx16
Target Package : csg324
Target Speed   : -3
Mapper Version : spartan6 -- $Revision: 1.55 $
Mapped Date    : Sun May  3 09:00:04 2020

Design Summary
--------------
Number of errors:      0
Number of warnings:    0
Slice Logic Utilization:
  Number of Slice Registers:                     0 out of  18,224    0%
  Number of Slice LUTs:                         60 out of   9,112    1%
    Number used as logic:                       59 out of   9,112    1%
      Number using O6 output only:              45
      Number using O5 output only:              11
      Number using O5 and O6:                    3
      Number used as ROM:                        0
    Number used as Memory:                       0 out of   2,176    0%
    Number used exclusively as route-thrus:      1
      Number with same-slice register load:      0
      Number with same-slice carry load:         1
      Number with other load:                    0

Slice Logic Distribution:
  Number of occupied Slices:                    21 out of   2,278    1%
  Number of MUXCYs used:                        12 out of   4,556    1%
  Number of LUT Flip Flop pairs used:           60
    Number with an unused Flip Flop:            60 out of      60  100%
    Number with an unused LUT:                   0 out of      60    0%
    Number of fully used LUT-FF pairs:           0 out of      60    0%
    Number of slice register sites lost
      to control set restrictions:               0 out of  18,224    0%

  A LUT Flip Flop pair for this architecture represents one LUT paired with
  one Flip Flop within a slice.  A control set is a unique combination of
  clock, reset, set, and enable signals for a registered element.
  The Slice Logic Distribution report is not meaningful if the design is
  over-mapped for a non-slice resource or if Placement fails.

IO Utilization:
  Number of bonded IOBs:                        22 out of     232    9%

Specific Feature Utilization:
  Number of RAMB16BWERs:                         0 out of      32    0%
  Number of RAMB8BWERs:                          0 out of      64    0%
  Number of BUFIO2/BUFIO2_2CLKs:                 0 out of      32    0%
  Number of BUFIO2FB/BUFIO2FB_2CLKs:             0 out of      32    0%
  Number of BUFG/BUFGMUXs:                       0 out of      16    0%
  Number of DCM/DCM_CLKGENs:                 0 out of       4    0%
  Number of ILOGIC2/ISERDES2s:               0 out of     248    0%
  Number of IODELAY2/IODRP2/IODRP2_MCBs:     0 out of     248    0%
  Number of OLOGIC2/OSERDES2s:               0 out of     248    0%
  Number of BSCANs:                          0 out of       4    0%
  Number of BUFHs:                           0 out of     128    0%
  Number of BUFPLLs:                         0 out of       8    0%
  Number of BUFPLL_MCBs:                     0 out of       4    0%
  Number of DSP48A1s:                        0 out of      32    0%
  Number of ICAPs:                           0 out of       1    0%
  Number of MCBs:                            0 out of       2    0%
  Number of PCILOGICSEs:                     0 out of       2    0%
  Number of PLL_ADVs:                        0 out of       2    0%
  Number of PMVs:                            0 out of       1    0%
  Number of STARTUPs:                        0 out of       1    0%
  Number of SUSPEND_SYNCs:                   0 out of       1    0%

Average Fanout of Non-Clock Nets:            3.78

Peak Memory Usage:  670 MB
Total REAL time to MAP completion:  8 secs
Total CPU time to MAP completion:   6 secs

Table of Contents
-----------------
Section 1 - Errors
Section 2 - Warnings
Section 3 - Informational
Section 4 - Removed Logic Summary
Section 5 - Removed Logic
Section 6 - IOB Properties
Section 7 - RPMs
Section 8 - Guide Report
Section 9 - Area Group and Partition Summary
Section 10 - Timing Report
Section 11 - Configuration String Information
Section 12 - Control Set Information
Section 13 - Utilization by Hierarchy

Section 1 - Errors
------------------

Section 2 - Warnings
--------------------

Section 3 - Informational
-------------------------
INFO:MapLib:562 - No environment variables are currently set.
INFO:LIT:244 - All of the single ended outputs in this design are using slew
   rate limited output drivers. The delay on speed critical single ended outputs
   can be dramatically reduced by designating them as fast outputs.
INFO:Pack:1716 - Initializing temperature to 85.000 Celsius. (default - Range:
   0.000 to 85.000 Celsius)
INFO:Pack:1720 - Initializing voltage to 1.140 Volts. (default - Range: 1.140 to
   1.260 Volts)
INFO:Map:215 - The Interim Design Summary has been generated in the MAP Report
   (.mrp).
INFO:Pack:1650 - Map created a placed design.
```

```
Section 4 - Removed Logic Summary
---------------------------------
   2 block(s) optimized away

Section 5 - Removed Logic
-------------------------

Optimized Block(s):
TYPE  BLOCK
GND  XST_GND
VCC  XST_VCC

To enable printing of redundant blocks removed and signals merged, set the
detailed map report option and rerun map.

Section 6 - IOB Properties
--------------------------

+----------------------------------------------------------------------------------------------------------------
| IOB Name                             | Type             | Direction | IO Standard          | Diff  | Drive    | Sle
|                                      |                  |           |                      | Term  | Strength | Rat
+----------------------------------------------------------------------------------------------------------------
| D<0>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<1>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<2>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<3>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<4>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<5>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<6>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<7>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<8>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<9>                                 | IOB              | INPUT     | LVCMOS25             |       |          |
| D<10>                                | IOB              | INPUT     | LVCMOS25             |       |          |
| D<11>                                | IOB              | INPUT     | LVCMOS25             |       |          |
| D<12>                                | IOB              | INPUT     | LVCMOS25             |       |          |
| E<0>                                 | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
| E<1>                                 | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
| E<2>                                 | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
| F<0>                                 | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
| F<1>                                 | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
| F<2>                                 | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
| F<3>                                 | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
| F<4>                                 | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
| S                                    | IOB              | OUTPUT    | LVCMOS25             |       | 12       | SLO
+----------------------------------------------------------------------------------------------------------------

Section 7 - RPMs
----------------

Section 8 - Guide Report
------------------------
Guide not run on this design.

Section 9 - Area Group and Partition Summary
--------------------------------------------

Partition Implementation Status
-------------------------------
```

```
  No Partitions were found in this design.

-------------------------------

Area Group Information
----------------------

  No area groups were found in this design.

----------------------

Section 10 - Timing Report
--------------------------
A logic-level (pre-route) timing report can be generated by using Xilinx static
timing analysis tools, Timing Analyzer (GUI) or TRCE (command line), with the
mapped NCD and PCF files. Please note that this timing report will be generated
using estimated delay information. For accurate numbers, please generate a
timing report with the post Place and Route NCD file.

For more information about the Timing Analyzer, consult the Xilinx Timing
Analyzer Reference Manual; for more information about TRCE, consult the Xilinx
Command Line Tools User Guide "TRACE" chapter.

Section 11 - Configuration String Details
-----------------------------------------
Use the "-detail" map option to print out Configuration Strings

Section 12 - Control Set Information
------------------------------------
Use the "-detail" map option to print out Control Set Information.

Section 13 - Utilization by Hierarchy
-------------------------------------
Use the "-detail" map option to print out the Utilization by Hierarchy section.
```