Project Documentation

TOC TOC

Table of Contents:

Intentionally left blank

§1 Filing Structure:

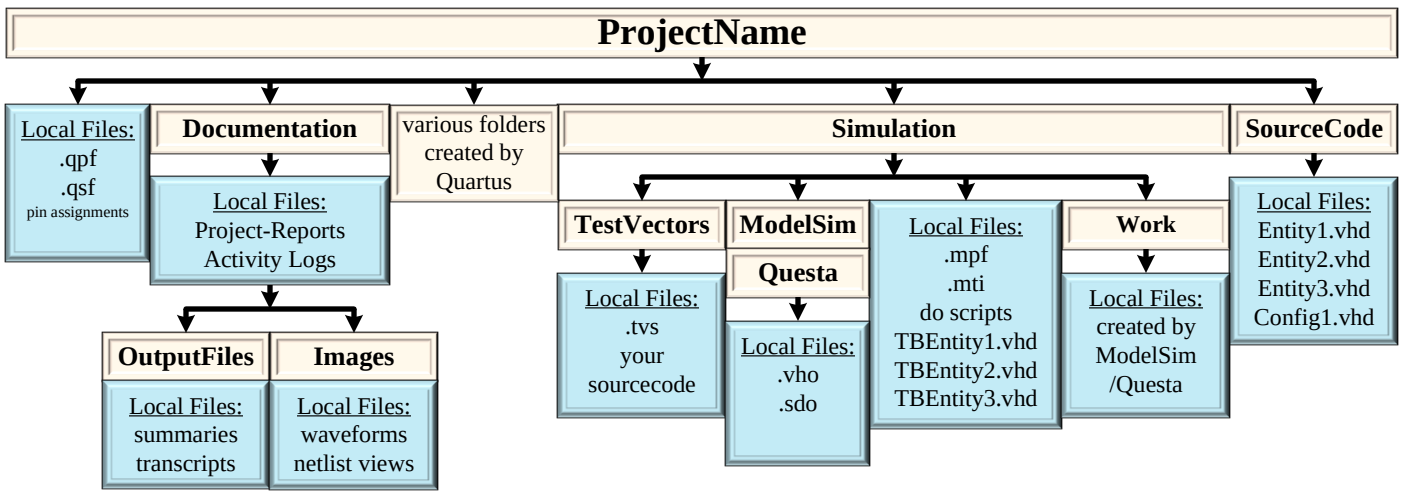
A single project will contain many files and folders, some of which will be created by Quartus & ModelSim/Questa. For all projects use the filing hierarchy described below.

Create a tree of folders as shown.
Scripts that you create or that I create on your behalf will assume that files may be located accordingly.

- Use the convention that files associated with a specific design entity are named with a prefix, using the entity name "EntityName"
  or
  a suitable abbreviation using capital letters ie. "**EN**".
- Testbenches have the same name as the entity being tested, with the prefix "**TB**".
- The root of the ModelSim/Questa project folder should contain
  - ✓ VHDL-Testbenches & Test Vectors, VHDL-Configurations,
  - ✓ ModelSim/Questa Scripts, and output Transcripts.
- The folders Simulation/ModelSim & Simulation/Questa is used by Quartus.
- The folder TestVectors is to be used as workspace for programs that generate test vector files.

Avoid directory names in the path that contain "spaces", use underscores instead.

ProjectName:	The top-level folder to be used as Quartus' project folder.
Documentation:	Used to collect files needed for creating project documentation.
SourceCode:	Used to store the synthesisable VHDL design files.
Simulation:	Used as the ModelSim project folder. (contains testbenches, test vectors, scripts)
ModelSim:	Used by Quartus to store post-fit netlists and timing files, for ModelSim.
Questa:	Used by Quartus to store post-fit netlists and timing files, for Questa.
TestVectors:	Used by you to create Test Vectors using a programming language of your choice.



§1.2 Creating a Build Archive: (to Build the Project from Source Files)

Documentation contains many human readable files, such as; project reports, .pdf sourcecode listings, summary reports, simulation transcripts, etc. In addition to these files there are the actual sources that are needed to reconstruct the final results.

Reconstructing the results requires

- Quartus project information – (.qpf, .qsf)
- ModelSim project information – (.mpf)
- All VHDL sourcecode. – (.vhd)
- All Testbench Sourcecode – (.vhd)
- All TestVector Files – (.tvs)
- All scripts – (.tcl, .do)
- Pre-complied Netlists – (.vho, .sdo)

Theses sources and their respective filing hierarchy should be preserved.
Create a bare filing structure containing only the relevant sources needed to reconstruct all result.

- Save this structure into a single archive as specified in the specific project requirements document.

To create the bare structure after a successful build.

- Do not copy the root folder as sometimes the files that control the toolchain contain absolute file references.

Remove all files created by ModelSim/Questa and Quartus, except for

- the project files (.qpf, .qsf & .mpf)
- the pre-compiled netlists.
- any output files that have been required as deliverable for the specific project.

In general output files such as summary reports and transcripts should be renamed and copied to the subdirectory “Documentation/OutputFiles” as you conduct project tasks.
Often a project requirement will be to condense and merge the information in these files to produce a single concise file. Perform this task if required as deliverable for a specific project.

Rename or merge all output files that are required as deliverable for the specific project.
Delete all other output files.

Output files that were manually renamed and moved it to a specific location, must be preserved in the build hierarchy.

§2 Recording Research Activity:

§2.1 Instructions for Activity Logs

Record your project activities in the log file. "ProjectLog-Gxx-xxxx-350-xxxx.xlsx". The excel workbook contains many worksheets. Type your log entries in the correct worksheet.

- ✓ Each group member must maintain their own activity log file.
  - Gxx-xxxx = (two-digit-Group number)-(Last 4-digits of StudentID)
  - 350-xxxx = Course Number-SFU semester code.

- ✓ Include all micro-activities of duration half-an-hour to (at most) two hours.
- ✓ Include time spent researching/studying related topics as separate activity items.

- ✓ Include time stamps for all micro-activities: the Start date/time, End date/time.
  - Ensure that **excel** recognises the cell data as valid date or time entries;
    - date format = dd-mmm-yy, time format = h:mm AM/PM
  - Do not modify the provided **excel** spreadsheet.

- ✓ Add a complete description of all micro-activities, including all relevant details.
  - Each description should contain precise specific verbs and nouns providing detailed information. Do Not use meaningless verbs such as "worked on"
  - Avoid general statements; be specific. Describing the actual activity. Do not write vague statements.

§2.2 Instructions for Engineering Notebooks

It is important that each group member maintain an engineering notebook. The notebook may be physical or you could maintain a soft notebook on a computer or tablet

- If you use a computer you should ensure that it is conveniently accessible while you are conducting your work or even simply thinking about the project.

The notebook should contain all of your thoughts related to the project.

- ✓ You should Make note of casual observations.
- ✓ You should record all casual calculations, no matter how obvious.
- ✓ You should make notes about your activities.

While in the middle of a project you will consider many facts as obvious, however one or two years in the future, these facts will be forgotten.

- Your notes should be chosen as though you were talking to yourself in the future.
- Additionally, your notes will be useful when it is time to write a report.

§3 Reports: Written Documents Submitted for Assessment

§3.1 Various Types of Reports:

There are many types of reports each serving different purposes and targeting different readers.

- Academic-style term papers, written by general undergraduate students, read by instructors/teaching assistants.
- Laboratory reports describing a particular experiment, written by science students, read by instructors/teaching assistants.
- Academic journal papers, written by university researchers, read by academic & industrial researchers.
- Business and financial reports, written by business administrators, commonly read by investors.

You may be accustomed to creating "lab reports" in some lower division engineer courses.
Please abandon that style; the report requirements for Ensc 350 are different.

A practicing professional engineer regularly produces many written documents.

- ✓ For Ensc 350 there are two final reports, a summary report and a project report.
- ✓ Both documents are to be carefully and incrementally constructed throughout the term.

Caution – DO NOT do the following.

In the past you may have become accustomed to, (***Abandon these habits***)

1) Consider the reader to be a teaching assistant or instructor.

2) Consider the purpose to be to score grade points, based on the fact that you accomplished specified tasks.

From considerations (1) & (2) a student typically proceeds by

- Collecting screenshots of lab instrument display, screenshots of S/W generated images showing synthesised circuits, simulations waveforms, SPICE schematic entries and display of simulated node voltage/currents.
- Inserting images into a document as though it were a catalog.
- Text is then added near the images containing mostly redundant (obvious) statements about the general content of the images.
- The text serves no real purpose other than to create the illusion that the catalog is actually a report.
- The document is then enclosed by
  - ✓ a beginning section that provides very basic and obvious theoretical discussions, and
  - ✓ ended with a section that makes glorious conclusions with very little factual basis and with only a vague marginal relation to the cataloged content.

§3.2 Report Outlines:

When creating written documents you should begin by organizing the document structure.

- Define the reader,
- Identify and consider the reader's concerns; items that interest the reader.
- Define the purpose of the document and list the most important information to be contained.

A outline will contain "content rectangles" which mark the position of objects that you are to place in the document. Examples;

- Written text,
- Images such as circuit diagrams, flowcharts,
- Tables of data,
- Very short segments of sourcecode, etc.

Within each content rectangle insert,

1) an appropriate title and identify the type of object.
2) a brief statement of purpose – identifying why this object is present.
3) a brief statement describing how this object logically links to the preceding objects.
4) a short bullet list of the important information to be highlighted in the object.

§3.3 A Summary Report:

Philosophy:
Imagine that your boss is the Engineering Manager who is in charge of many projects. The manager needs a summary report that concisely provides information to oversee your project.

- ✓ A summary report should not be too long, at most two pages.

Think carefully about what role the manager plays in guiding a project. Write a list of the responsibilities of a project manager in the context of a specific project.
In this case the project will be the activities for Ensc 350, DP1, DP2 and FP.
Example responsibilities: (*Please consider additional items.*)

- ✓ Budget, (Financial resources),
- ✓ Qualified personnel, (Human resources),
- ✓ Facilities scheduling and management,
- ✓ Equipment procurement, (Hardware & Software),
- ✓ Schedule of project tasks and milestones,
- ✓ etc.

Once you have determined the Manager's items of interest,

- ✓ consider the actions and decisions that a manager makes.
- ✓ create a list of all information that the manager would need and
- ✓ consider how you would write a summary report that is easy for the manager to quickly acquire the relevant information.

Your summary report should contain clear and concise feedback to a project manager, informing them of the current status of the "plan".

- If an item of interest is progressing exactly as planned; then state this concisely.
- If an item of interest is not going as planned, or new facts arise that were not considered when the original plan was formulated;
  - ✓ then identify important factors, analyse the issue thoroughly, provide a concise discussion and
  - ✓ conclude with a recommendation for resolving the issue.

Structure this summary report to document your R&D activities. As you are training to become a professional engineer it is important to write this part highlighting your "professionalism" .
You should decide how best to structure sections in this part.

§3.4 A Design Project Report:

§3.4.1 Philosophy:

A typical project will be extensive and may span many months (or even years). When a new engineer joins the project they must quickly and efficiently become familiar with the project.

Project documentation should be organized and contain complete technical information about all aspects of the project. Should an engineer working on one part of a large project need specific technical information about another part of the project; the information should be available in the project documents.
The structure and content of documentation varies considerably from one project to another. A software development project, an electronic system design, a building design are vastly different.

Think carefully about the scope of a particular Ensc 350 project.

- ✓ How would you sub-divide the project documents into top-level categories?
- ✓ How would you logically further sub-divide these categories?
- ✓ In what form is it best to document specific information?

You should view Ensc 350 Project Reports as a learning exercise for creating engineering documents. It is not the same as a university course term paper.

You should consider the reader to be a qualified professional engineer who

- ✓ wishes to quickly and efficiently begin practical work on the same project.
- ✓ Has a specific technical question about the project and wishes to quickly find the answer.

The purpose of an Ensc 350 Design Project Report IS NOT to provide evidence that you conducted the work. DO NOT simply create a document containing a catalog of images.

Your report should be well organised. You choose how to divide the report into sensible sections. All **recommendations or guidelines that follow are NOT a RECIPE** for writing the report.

It is obvious to the reader when a report has been constructed with minimal thought and effort.
*A report that fills space with source code listings, screen captures produced from various tools, accompanied by a few redundant sentences is **not** considered acceptable for upper division university students*.

§3.4.2 Design Project Report Guidelines:

Do not include excessive general theory. Assume that the reader is already educated, and thus only requires a quick reminder of less obvious theoretical details. You should however discuss theoretical aspects that are very specific to your particular project.

- ✓ Example: The ideas behind ripple adders or the carry-select principle are general theory and reasonably obvious. The particular definition of a conditional-sum adder is a theoretical detail that may require a reminder. The actual implementation of a conditional-sum adder, with recursion isolated to a carry network and implemented with 4-input LUTs is specific to a particular project.

Documenting Experimental Procedures:

The overall task of acquiring observable results should be precisely described in the document. The larger picture of how configurations, scripts and a testbench should be concisely presented.

Typically, multiple scripts will be written to facilitate this experimental process. Ensure that the document contains sufficient information for the reader to be able

- ✓ to use the existing scripts to generate results and also
- ✓ to use the existing scripts to quicky create newer scripts.

Once again, flowcharts are very useful for summarizing such information, avoid verbose paragraphs describing procedural flows.

Documenting Results and Analysis of Results:

The results and analysis of results should dominate the report, (*maybe 40% - 70% of the report*.) Throughout the iterative design process, predictions and made, results are observed and useful metrics are derived from results.

All quantities should be concisely defined and sample calculations of derived metrics should be provided. Additionally, the plethora of information should be concisely summarized in small tables, graphs or other diagrams.

Once the values of predictions, measurements & metrics have been concisely provided, analysis begins.

- The patterns and relationships must be identified.
- Anomalies must be identified.
- Refined measurements should occur to test any arising hypothesis. Only then can reasonable conclusions drawn.

Documenting Design Candidates and Internal Entities:

Describe the hierarchical structure of the design topology showing the relationship between the internal entities. The use of configurations and components in the hierarchy should be carefully described. Provide comprehensive information describing internal entities,

- A reference to the name of the VHDL source file.
- A short statement describing the functional behaviour of the internal entity.
- The VHDL interface – descriptions of the purpose of port signals.
- A sensibly drawn circuit diagram for the entity. (RTL Views may not be sufficient) (all labels must identically match your VHDL sourcecode.)

The description of design candidate should not be excessive and not scattered over many pages. Try to limit the description to at most two pages.

Documenting Testbenches and Test Vectors:

The discussion describing the design of the testbench should include;

- ✓ a diagram of the processes and signals within the architecture,
- ✓ a description of the flow for the procedure STIM: (maybe including a flowchart) and
- ✓ a description of the choice of any test vectors that you created and how they were created.

Try to keep this section concise, limited to only a few pages. Do not create paragraphs describing the procedural flow of the testbench; a well constructed flowchart with expanded annotations will provide the same information without boring the reader.

Describing Topologically Dependent Quantities:

Certain empirical items are specific to a design topology.

- ✓ Test vectors must cover estimated worst-cases.
- ✓ Predictions of circuits complexity.
- ✓ Predictions of circuit timing.

How do you present the detailed description of these items? You should consider the reader and decide whether is best to

- discuss these items individually within the discussion of the topologies or
- to collect the discussion of all topologies into one section.

The objective is clarity in presenting the information and convenience when searching for information. (consider cross-referencing)

§4 Guidelines for Content:

§4.1 Documenting VHDL Sourcecode:

A .PDF document rendered to contain all VHDL sourcecode, with coloured syntax highlighting and line numbers is called a Listing. The listing is a document for humans and as such the comments within the listing are very important.

If the complete listing of all sourcecode produces more than 3 pages as a PDF file,

- then it should be provided as a separate PDF file, but listed in the table of contents of the project report as an appendix. (provide the filename)

Else if three or less pages,

- the listing can be included within the project report, as an appendix.

Your VHDL sourcecode listing must

- contain coloured syntax highlighting, and
- omit the LIBRARY/USE clauses because they are obvious and unnecessarily waste space, except for the case of entities that use special libraries.
- contain useful comments.

When commenting your code, imagine that you were reading the text in the distant future. Consider the comments as notes to yourself so that you could efficiently continue working on the project.

Small segments of VHDL code may occur in the body of a project report. In this case the code would be included because the author has decided that showing a small segment of VHDL would be useful in describing some specific topic. A small circuit diagram is usually more useful and the VHDL segment would accompany this circuit diagram.

If you use notepad++, you can copy the selected text to the clipboard with syntax highlighting. Use the menu, Plugins->NppExport->Copy all formats to clipboard. You can also change the colours of the syntax highlighting using the menu, settings->style configurator. Then select the language VHDL (on the left) and choose your colours. I chose mine to best match the default colour of the ModelSim Editor.

In addition to the .pdf listing, the actual .vhd files must also be include as part of the documentation. The sourcecode files should be located sensibly.

- Refer to the section describing the filing hierarchy for further details.

§4.2 Documenting Quartus Output Reports: (Synthesis, Mapping and Fitting)

Using Quartus, you will synthesise and fit your design entities. Quartus will produce summary reports located in a folder called "output_files". You will need to compile the summary information as part of formal project documentation. Review these text files.

projectname.flow.rpt – description of the sequence of tools used to process the project.
projectname.map.rpt – describes actions of the synthesis and technology mapping.
projectname.fit.rpt – describes actions of the fitter performing place & route.
projectname.asm.rpt – describes actions of the assembler to create the sof.
projectname.sta.rpt – describes actions of the static timing analyser.
projectname.eda.rpt – describes actions of the EDA netlist writer.

For formal Ensc 350 project documentation, we usually include the .map.rpt and .fit.rpt files
If a report file is rendered as a pdf, your should ensure that

- a non-proportional font is used and a font size is chosen to preserve the tables. (**`courier`**)
- Do not allow the table rows to wrap around.
- There is absolutely no value in rendering the file when it cannot be read.

I found that using courier new font, size 5, to a landscape letter size page reproduces the wide table in the mapper report.

- Even though the text in the pdf file is tiny, it becomes quite legible when the page is zoomed.
- Using a larger font would not be acceptable because the line wrap-around would misalign the table make a very confusing document.

~~The end of this document contains a sample page of a report rendered to a pdf as an example~~.

Quartus also generates a short form of these reports, with the file suffix .summary.

For formal Ensc 350 project documentation the .map , .summary and .fit.summary documents from individual synthesis runs should be merged into a single text file.

✓ Place the Deliverable merged Quartus Reports in the folder Documentation/OutputFiles.

- Distinctive section boundaries should be inserted to separate the individual files using the original filename in a descriptive title.

for example,

```
/*****************************************************************************/
/* MyEntity.map.summary
/*****************************************************************************/
```

§4.3 Documenting Netlist Views:

The RTL netlist viewer and the Post-fitting netlist viewer generate useful diagrams for debugging but these images are not generally useful for project documentation. You should manually create diagrams by yourself that are specific to the needs of your project report.

The circuit images generated by software are not the same as those created by rational human. You must be very careful if you decide to use software generated images instead of drawing them yourself. It is absolutely necessary to compensate for the unnecessary absurdity arising from software generated images.

Use the following guidelines when inserting images produced by netlist viewers into a formal project report.

- For larger circuits these images can become confusing and thus you will need to pay careful attention when constructing your documents.

When capturing an image, you may either convert immediately to a bit-mapped format or you can export the image to a pdf. Bit-mapped images have issues with resolution and file size.

- Whenever possible you should try to capture images in a vector-graphic format. For now capture a bit-mapped image but make sure that the final image is readable.

Factors to consider when documenting images.

- Choose the zoom-level of the original image so that the important aspects of the image are clear and captured with the best resolution.
- Choose a size and position on the page that makes sense without excessively disturbing the original aspect ratio.
- If necessary, include a birds-eye view. *See the following examples*.
- Include annotations to efficiently direct the reader's attention to the important aspects of the image. *See the following examples*.
- Label the image with a figure number.
- Add a line of summary text to explain & identify to the reader the content of the image.
- If the viewer chooses an absurd order for the elements of the diagram, insert numbers onto the elements to compensate for the foolish ordering.
- If the viewer draws an important signal path using a convoluted path, try to annotate the diagram to compensate for the unnecessary obsurity.

A project report will be judged on your ability to sensibly choose these images.
You will score zero if it is clear that you are not making sufficient effort to graphically convey useful information about the underlying circuit.
Refer to the examples on the next few pages.

Each screen clipping from a netlist viewer should occupy either a full page or half a page so that details are not lost. (Here is an example)

Choose the page to be either landscape or portrait, whichever best matches the aspect ratio of your image.

Each image should have a brief description. When appropriate, the image should contain both the zoomed view and the window for the bird's eye view. Position and size the bird's eye window so that it doesn't obscure too much of the zoomed circuit. Choose the zoomed region wisely.

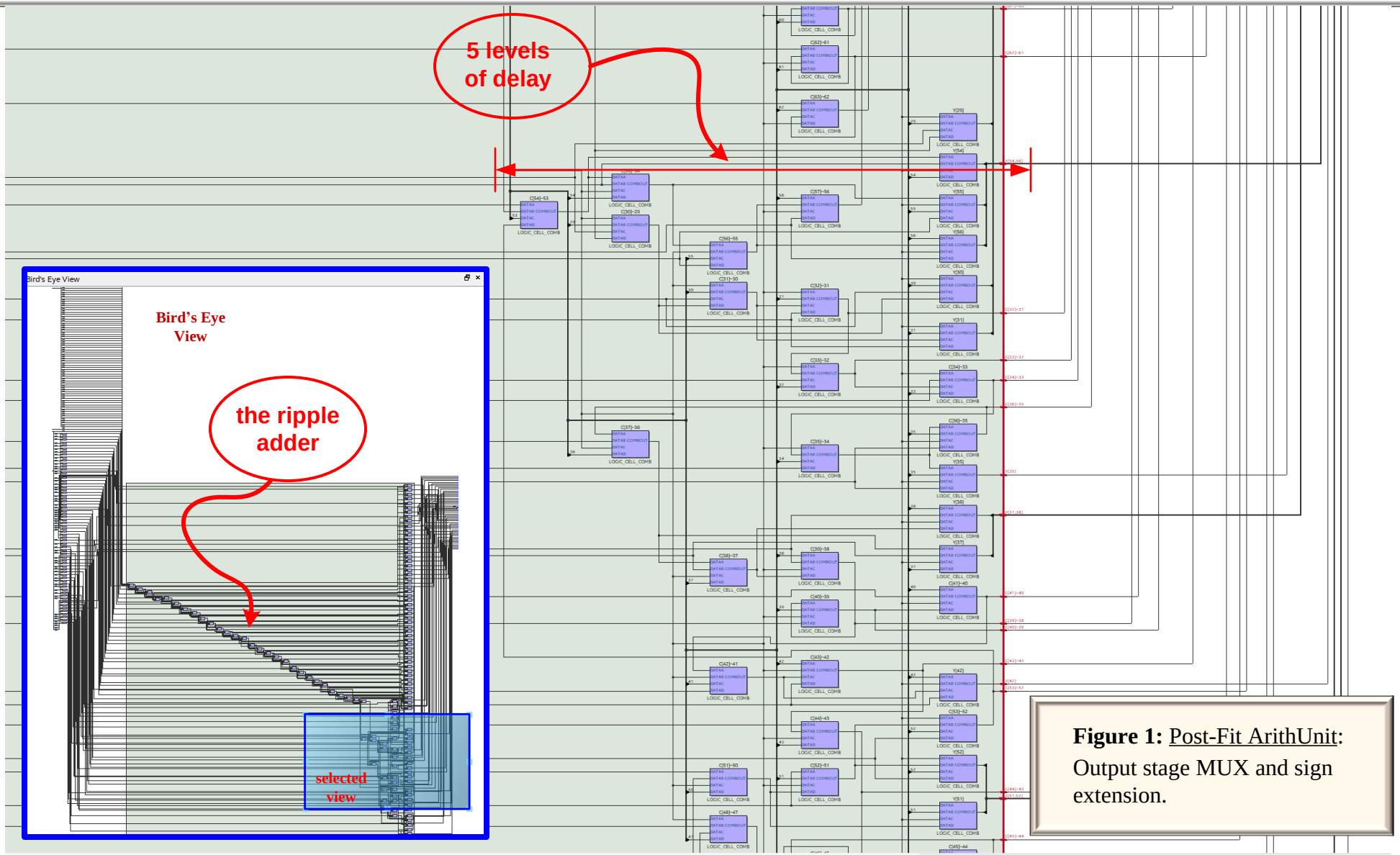


Figure 1: Post-Fit ArithUnit: Output stage MUX and sign extension.

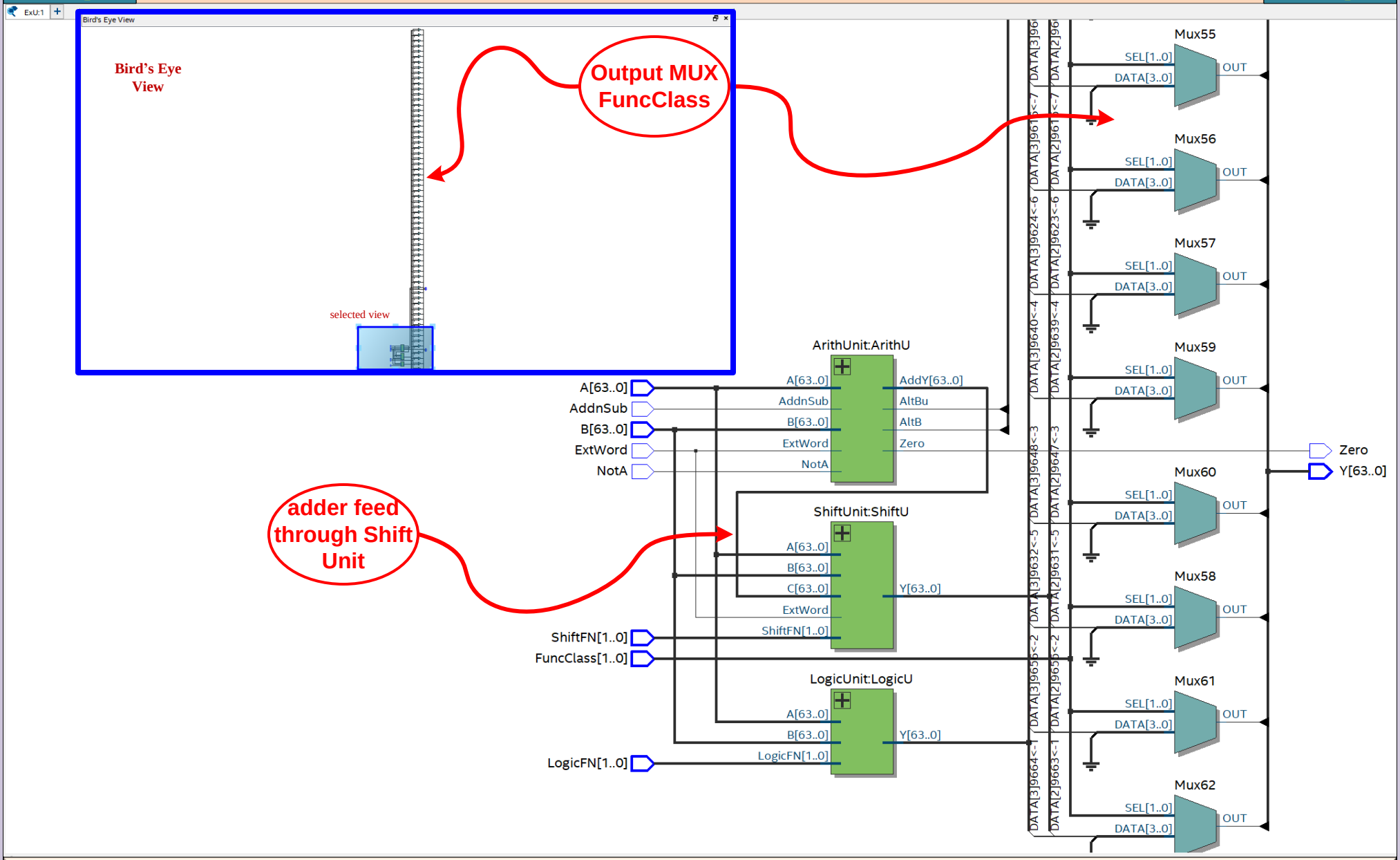


Figure 2: Post Synthesis RTL Circuit – Execution Unit:
The Execution Unit Top-Level View, showing the three main sub-circuits and the output MUX controlled by FuncClass.

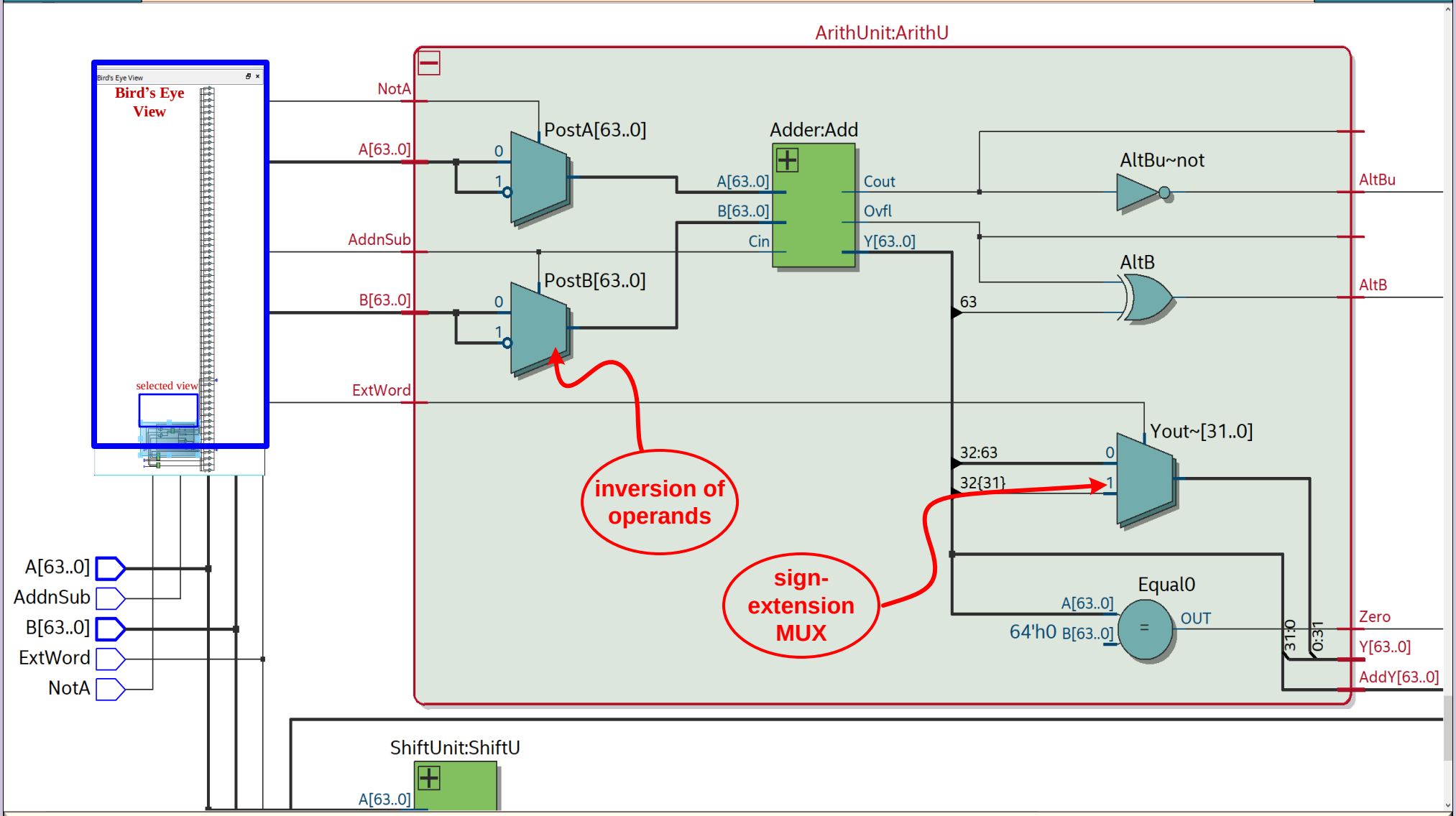


Figure 3: Post-Synthesis RTL View – Execution Unit:

The internal view of the Arithmetic Circuit showing

- the formation of comparison signals (A<B) from output Carry and Overflow, the formation of the Zero flag and also the two input stage MUXes for selectively inverting the operands.
- The outputs Cout, Ovfl and Y[63..0] are left unconnected.

§4.4 Documenting Simulation Waveforms:

You should also make screen clippings of the waveforms produced by ModelSim.
You should capture a set of waveforms from each individual simulation run.

You should use a script to easily reproduce the waveforms.
If you need to make your own script, (*or wish to modify a provided script*)

- If script for displaying waveforms is provided, run the script.
- Manually include additional signals.
- Arrange the order of signals in the wave window.
- Add dividers to group related signals and name the dividers sensibly.
- Adjust the colours, row height and display format.

You should decide by yourself how best to effectively convey useful information.

To create a script that generates the wave window:

- When the wave window is active, type <ctrl-s> and save the changes.
- You may wish to over-write a provided script.
  If you create/modify a provided script, your version must be included in the build archive.

The simulation waveforms should be sensibly included in your project report, with titles and annotations. Use two images per landscape page as shown in the example on the next page.

Each set should be accompanied with a discussion that explains why this simulation run verifies the functional behaviour and timing of the tested entity. The highlights on the images should be chosen to enhance specific statements within the discussion.

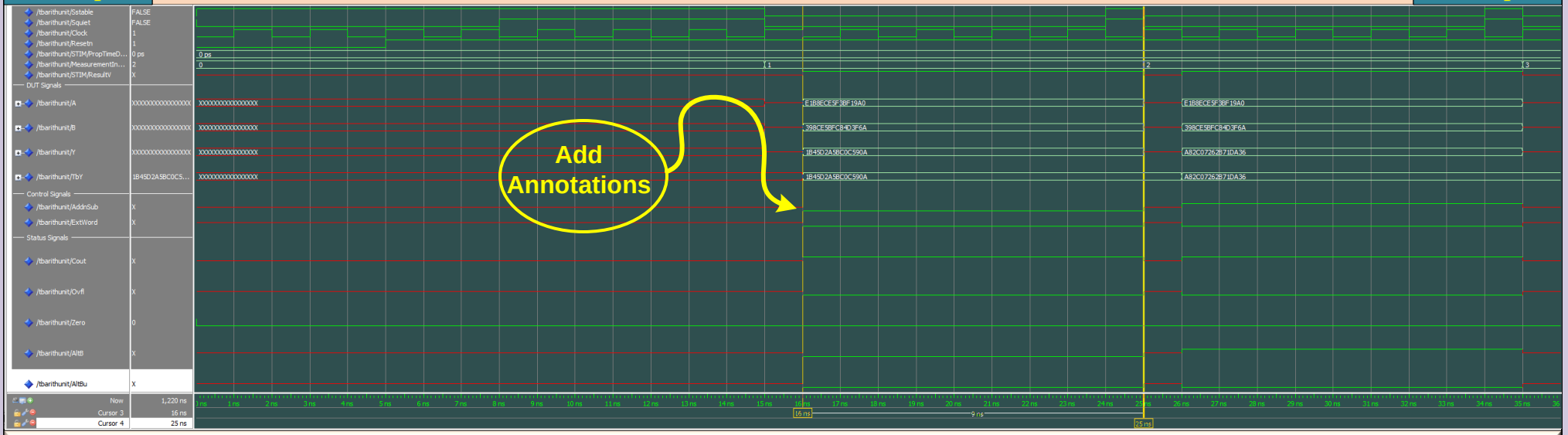
To make the acceptable images for a project report,

- Undock the wave window so that is occupies a large amount of the screen.
- Resize the window. Adjust the bottom so that the time-axis is close to the bottommost waveform.
- Type “r” - to select a zoom range for specific measurements.
- Add measurement cursors to mark significant instances of a measurement.
- Clip the image - <window+shift+S>, then
- Paste the image to a file in the Documentation folder. <ctrl+shift+alt+V>

Always check that the text for the signal groups is readable.

The following page shows examples of some captures.

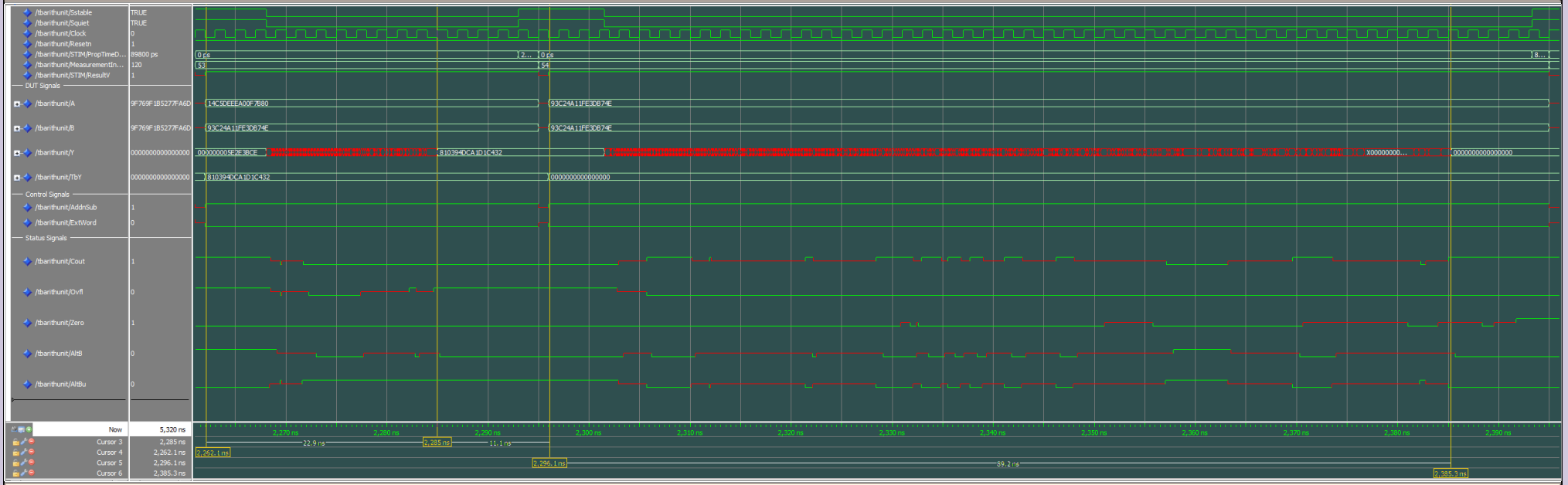
Example



Example

Functional Simulation - ArithUnit:

add some description of which measurements are being observed. Also the placement of the cursors. Add annotations to the wave.



Timing Simulation - ArithUnit:

Add a brief description. Measurement 53, t_{pd} = 22.9 ns, Measurement 54, t_{pd} = 89.2 ns