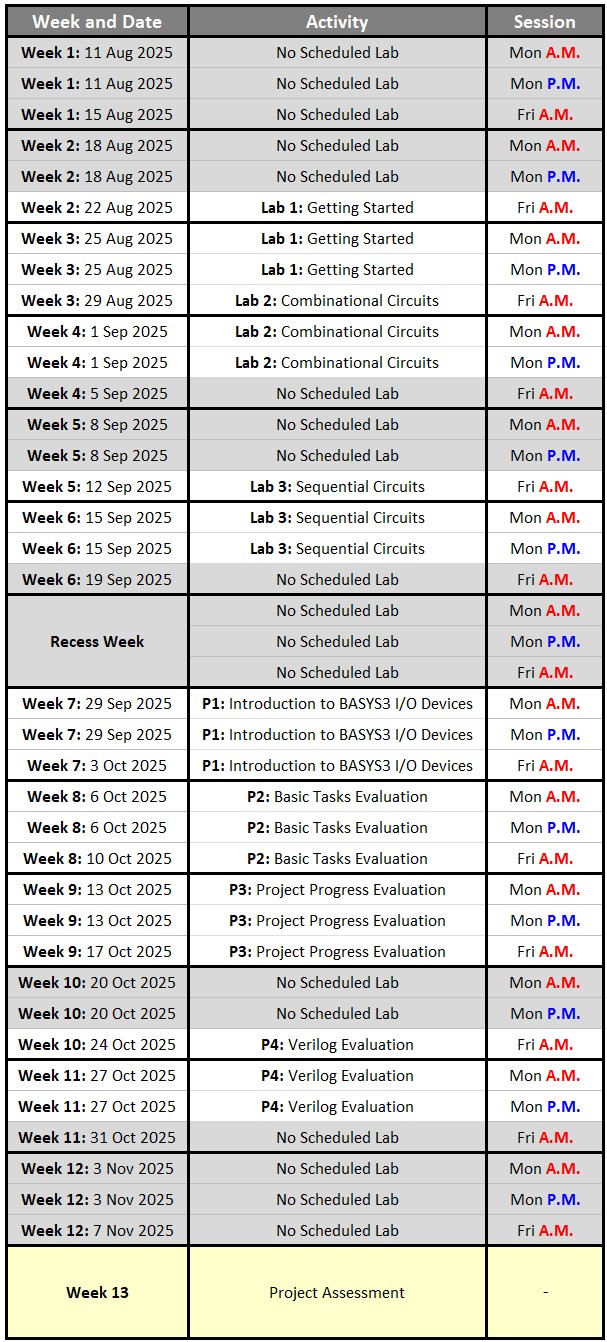
**LAB VENUE:**

**DIGITAL ELECTRONICS LAB @ E4-03-07**

**LAB SCHEDULE:**



**All the labs have graded assignments that need to be completed before the next lab.**

**Delaying any of these assignments will make all subsequent labs exponentially more difficult to manage!**

**EE2026: DIGITAL DESIGN**

**Academic Year 2025-2026, Semester 1**

**LAB 1: Quick Start Guide to Vivado 2018.2,**

**Basys 3 Development Board, and Verilog HDL**

**HARDWARE COMPATIBILITY:**

* Vivado 2018.2 allows for **Basys3** **hardware connection** only on Intel or AMD x86/x64 based processors, and using Windows. It will not allow Basys3 hardware connection on ARM based processors (Such as Apple’s M1, M2, M3 etc. processors).

* Virtualisation of windows, or non-English character (Eg. Chinese version of windows) Windows, generally have issues with the Basys3 hardware connection also.

**FOR ALL EE2026 LAB AND PROJECT SESSIONS:**

* You are **strongly** **encouraged to bring to lab, your own Windows laptop with Vivado 2018.2 already installed**. You may still use the desktop PC in lab if you do not have a **windows laptop (With x86/x64 processors)** that can be brought to lab.

* Use the **D:\MyWork** folder for your work if you are using the lab PC. You are required to **delete** all folders within the **D:\MyWork** folder before starting your lab session.

* **Delete** your work folder from the laboratory’s computers after your session is over. You are responsible to **safeguard** your confidential programs. For assessable programs, you will be penalised if two programs with similarities beyond empirical evidence are detected. The source(s) and recipient(s) of plagiarised programs are equally penalised.

**OVERVIEW:**

Using a simple Boolean design problem, an introductory approach to the Vivado software used in EE2026 will be covered. Quick instructions on downloading and installing the Vivado software on your personal computer are provided. The Vivado software is a comprehensive integrated development environment (IDE) for FPGA design flow.

In this lab:

* An introduction to very basic Verilog HDL (Hardware Description Language) is provided.
* The overall process flow of designing, synthesising, simulating and implementing a program is covered.
* Programming Digilent’s Basys 3 development board, which features an FPGA from Xilinx’s Artix-7 family, is illustrated.

**GRADED ASSIGNMENT [CANVAS SUBMISSION: FRIDAY 29th AUGUST 2025, 6:00 A.M.]:**

The graded assignment for lab 1 is required to be started during your first lab session and preferably submitted within the same day. It is thus compulsory to have the Vivado software already installed on your **English-Based Windows laptop with Intel / AMD processors** before the lab session.

Details for the assignment are available at the end of this lab manual.

**VIVADO DOWNLOAD AND INSTALLATION:**

The Vivado 2018.2 software is already installed on the computers in the Digital Electronics Lab, and are ready for immediate usage. Some quick guidelines on installing the required software for EE2026 on your personal computer is provided in this section.

**Software:**

AMD/Xilinx Vivado Design Suite - HLx Editions - 2018.2 [**Jun 18, 2018**]

The direct download link is: **https://www.xilinx.com/member/forms/download/xef-vivado.html?filename=Xilinx\_Vivado\_SDK\_2018.2\_0614\_1954.tar.gz**

Registration is required only downloads from the AMD/Xilinx website. It is not required for program installation and usage. The file size is more than 17 GB, so it may take many minutes/hours to download.

***Warning:*** *Do not use other versions of the software. Only the* ***Vivado 2018.2*** *Windows version has been tested.*

*Computer compatibility issues will occur with other versions of the software, and assessment of your project may not be possible. This will lead to loss of project marks if your project cannot be assessed.*

# Installation

During the installation phase, you will be given an option on the edition to install. The edition to be installed is:

• Vivado HL WebPACK

For subsequent customisation options, you can leave it to the default settings.

# Post-Installation

Restart your computer before using the Vivado 2018.2 software. You are recommended to **uninstall the “Xilinx Information Centre”** from the Windows control panel as it is not needed. This will prevent unnecessary pop-up messages by Xilinx from appearing.

**DESCRIPTION OF THE SIMPLE BOOLEAN DESIGN TASK**

The following task is required to be implemented on the Basys 3 development board:

* S

## UNDERSTANDING | TASK 1

Complete the truth table for the simple boolean design task:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **INPUT** | |  | **OUTPUT** |  | **MINTERM** |
| **A** | **B** | **LED1** | **LED2** | **LED3** |
| 0 | 0 | 0 | 0 | 0 | 𝐴̅𝐵̅ |
| 0 | 1 | 0 | 1 | 0 | 𝐴̅𝐵 |
| 1 | 0 | 1 | 0 | 0 | 𝐴𝐵̅ |
| 1 | 1 | 1 | 1 | 1 | 𝐴𝐵 |

# Deriving an SOP Boolean Equation for the Design Task

Given any truth table with any number of input variables, the sum-of-products (SOP) or product-of-sums (POS) form may be used to write out a Boolean equation for each output variable. Let us use the canonical SOP form for **LED1**:

𝐋𝐄𝐃𝟏=𝑨𝑩̅+𝑨𝑩

**𝐋𝐄𝐃2= 𝐴̅𝐵+𝐴𝐵**

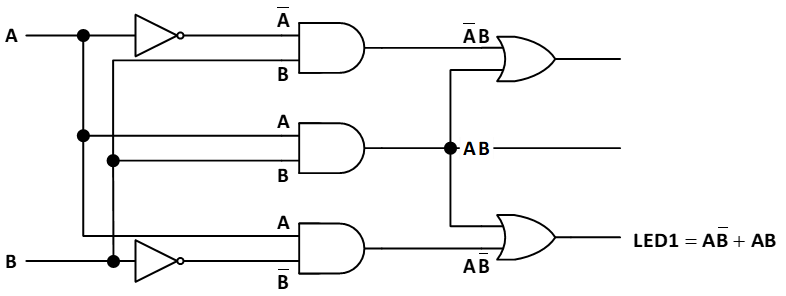
**𝐋𝐄𝐃3= 𝐴𝐵**

**UNDERSTANDING | TASK 2**

Work out the canonical SOP Boolean equations for **LED2** and **LED3**

# Illustrating Logic Expressions by Using a Schematic of Gates

The Boolean equations for **LED1**, **LED2**, and **LED3**, can be implement by using: 2 **NOT** gates, 3 **AND** gates, 2 **OR** gates

60

# Verilog Hardware Description and FPGA Implementation

Xilinx’s Vivado software is an integrated design environment that has numerous amounts of advanced features used in the industry, and among which we will be introducing the following:

* Writing and editing HDL codes for digital system designs.
* Simulation of the design’s behaviour.
* Synthesis of the codes, in order to convert the design from textual description into logic gates.
* Implementation of the design to map and route the logic to a target FPGA.
* Optimising the synthesis, implementation, and bitstream generation according to the user’s strategies. The default optimisation strategies shall be used in EE2026, as changing them is beyond the scope of introductory digital designs.
* Programming an FPGA with the optimised bitstream.

The remaining part of this lab manual will now briefly show the general steps required to go from the design task, to the FPGA implementation on the Basys 3 development board, for EE2026 purposes.

# INTRODUCTORY QUICK START GUIDE TO XILINX’S VIVADO 2018.2 SOFTWARE

During your lab session, your EE2026 graduate and lab assistants may provide you helpful hints on the usage of the Vivado 2018.2 software, beyond the most basic things that are described in this section.

## Creating a New Verilog Project in Vivado

**Start Menu:** Open the executable: Vivado 2018.2. You will need to wait multiple seconds before the program opens

**Quick Start:** Select **Create Project** and continue

**Project Name:** Enter a **Project name** and **Project Location**. Ensure that the **Project name** and complete **Project location** for your project folder does not have any spaces or special characters, and that your **Project name** does not start with a number

**Project Type:** Select **RTL Project**, and uncheck “**Do not specify sources at this time**”

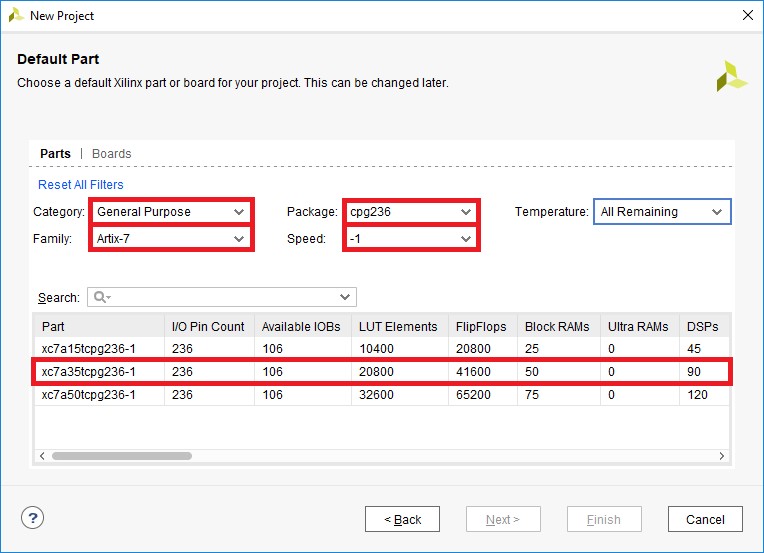
**Add sources:**

* **Create File.** File type is Verilog. Example: simple\_boolean
* **Target language:** Verilog. **Simulator language:** Mixed

**Add constraints (optional):** Click on next without any changes

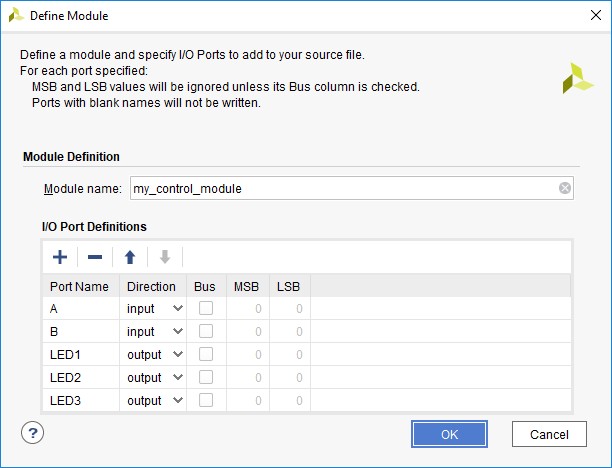
**Default Part:** Specify the FPGA chip that will be used. The Basys 3 development board uses the **xc7a35tcpg236-1** chip





**New Project Summary:** To create the project, click **Finish**

**Define Module:** A module, that is contained within the file, need top be created. Create one based on the inputs and outputs of the simple boolean design task.



## Using Vivado Text Editor to Write Verilog HDL Code

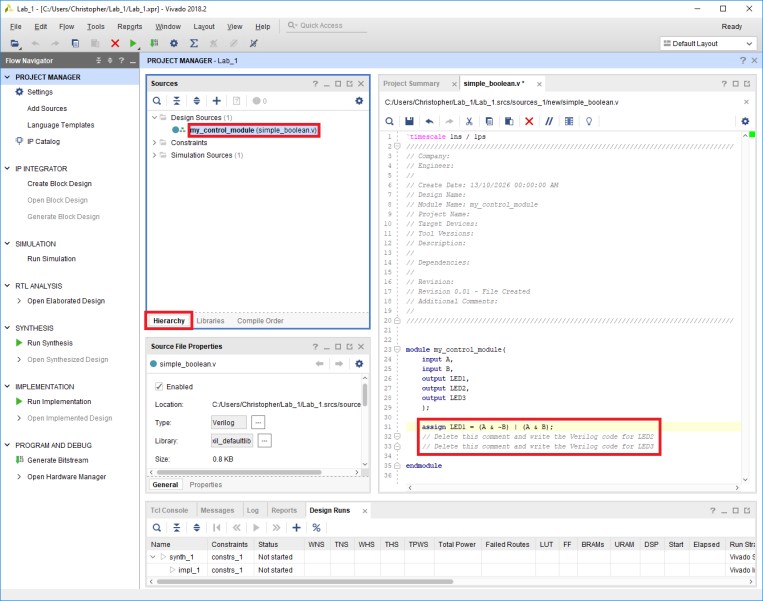
Open the module that has been created by double clicking on it in the Sources window

### UNDERSTANDING | TASK 3

Code the behaviour of the module by converting the SOP expressions for **LED1**, **LED2**, and **LED3** to the Verilog equivalent. The codes are to be inserted between the keywords **module** and **endmodule**.

Some Verilog representation of common operators are as tabulated below:

|  |  |  |
| --- | --- | --- |
| **Operators** | | **Verilog Representation** |
| OR | 𝐀+𝐁 | | |
| AND | 𝐀𝐁 | & |
| NOT | 𝐀̅ | ~ |
| XOR | 𝐀⊕𝐁 | ^ |



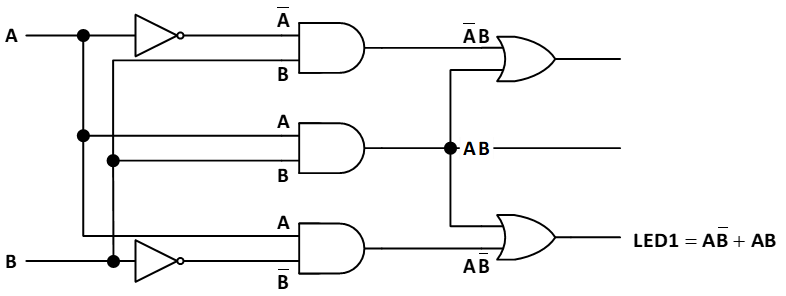
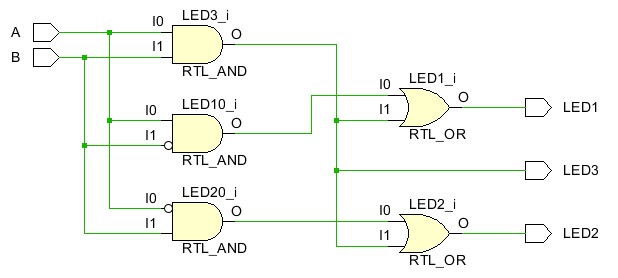
The **assign** statement causes the left hand side of the expression to be updated *every* time there is a change on the right hand side of the expression. It is therefore called a *continuous assignment* statement, describing combinational logic whereby the output on the left is a function of current inputs on the right.

The statements on line 31 till line 33 execute concurrently. This is in contrast to sequential execution of statements in a computer programming language such as C, or procedural assignment that will be taught in subsequent lab sessions.

Save your current file by clicking on **File**  **Save File**, or by pressing **Ctrl+S**. Each time a file is saved, a syntax check is carried out. After saving, perform the following: In the **Flow Navigator** window, under **RTL ANALYSIS**: **Open Elaborated Design**, select **Schematic.** The schematics window will appear, showing the Register Transfer Level (RTL) schematic of the design.

### UNDERSTANDING | TASK 4

What similarities and differences do you notice between the RTL schematic and the schematic obtained from the previous section. How do they compare to the actual schematic obtained on your computer screen?

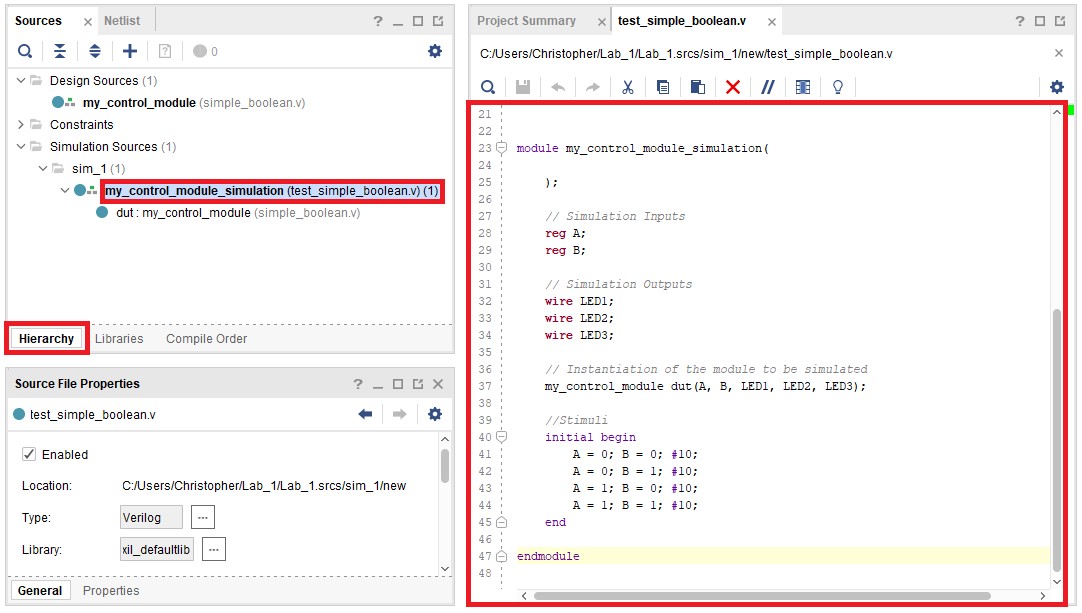


## Testbench and Behavioural Simulation

After writing the codes, there is a need to test them to check their behaviours. Inputs are applied to a module, and the outputs are checked to verify whether the module operates as intended. A testbench is an HDL module that is used to test another module. In this example, a testbench will be created to apply inputs to the module to be tested:

* From the **PROJECT MANAGER**, click on **Add Sources**, followed by **Add or create simulation sources**
* **Create File**, and provide a Verilog file name, such as **test\_simple\_boolean**
* In the subsequent **Define Module** window, provide a **Module name**, such as **my\_control\_module\_simulation**
* Do not input any **I/O Port Definitions**, and click on **OK** to finish creating the simulation module template

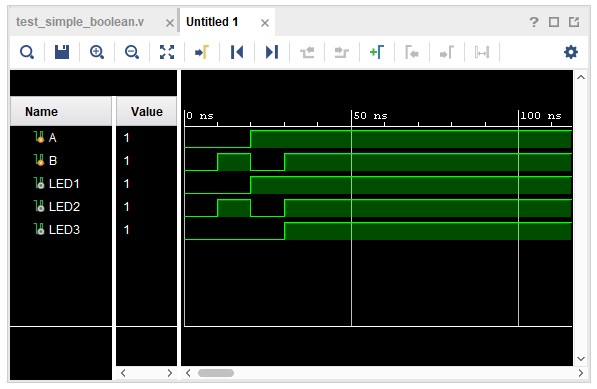
From the Sources window, open the simulation file. Then, within the simulation module, provide the following codes and save them, with the final screenshot looking similar to the image shown below:



If there are no syntax errors, in the **Flow Navigator** window, under **SIMULATION**, select **Run Simulation**, followed by **Run Behavioural Simulation** in order to create the simulation waveform window.

A noticeable waveform pattern may not be seen by default, as the time resolution used in the simulation is very small as compared to the amount of time the simulation is ran. Hence, with the simulation windows being the active window and from the menu, select **View**  **Zoom Fit**, or press **Ctrl+0**

Look at the simulation results closely. How do the waveforms show that your design is indeed working as desired? Consider trying out the various options provided in the simulation window before going back to the Workspace. Do not save the simulation window waveform, as this consumes a large amount of storage space.



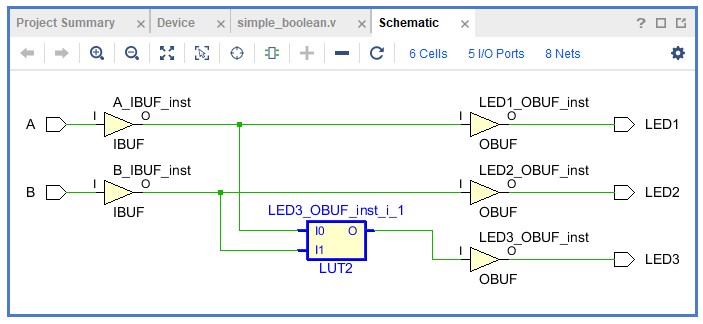
## Synthesis

Logic synthesis transforms HDL code into an optimised set of logic gates to reduce the amount of hardware, and to efficiently perform the intended function.

Right-click on your Verilog design source file and select **Set as Top**. This option is disabled if the file is already the top module, and in such a case, proceed directly to the next step. In general, when there are multiple design and simulation modules, the “Set as Top” option selects the design, or simulation, modules to be considered when performing the different stages of the project flow.

In the **Flow Navigator** window, under **SYNTHESIS**, select **Run Synthesis**. While Vivado performs synthesis, the Project Status Bar at the top right provides an indication of the ongoing progress.

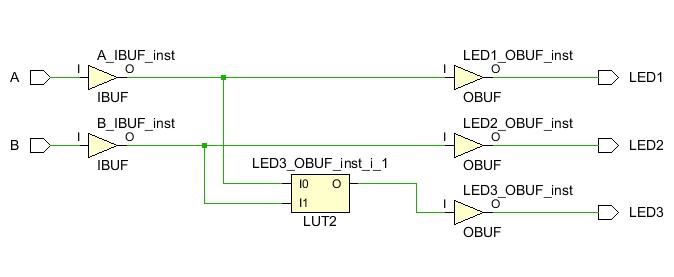
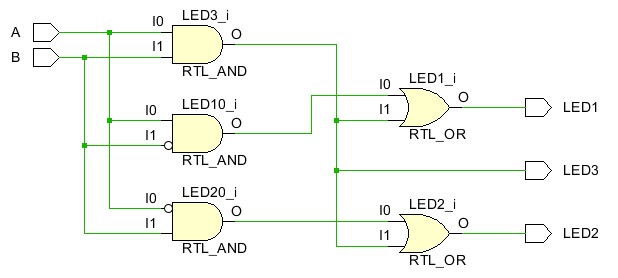
After the synthesis has been successfully completed, in the **Flow Navigator** window, under **SYNTHESIS**, expand **Open Synthesized Design**, and select **Schematics**. The schematic of the synthesised design will be generated and this synthesised circuit is an optimised version of the RTL schematic that was obtained



Click on the Look-up Table (LUT) that defines how the output LED3 behaves. The **Cell Properties** window will appear for that specific LUT. In the **Cell Properties** window for the LUT of **LED3**, open the **Truth Table** tab. Notice how for this simple example, this LUT is behaving as a simple AND gate.

### UNDERSTANDING | TASK 5

Compare the optimised and non-optimised schematics. How is this optimised circuit equivalent to the SOP equations of the simple boolean design task?



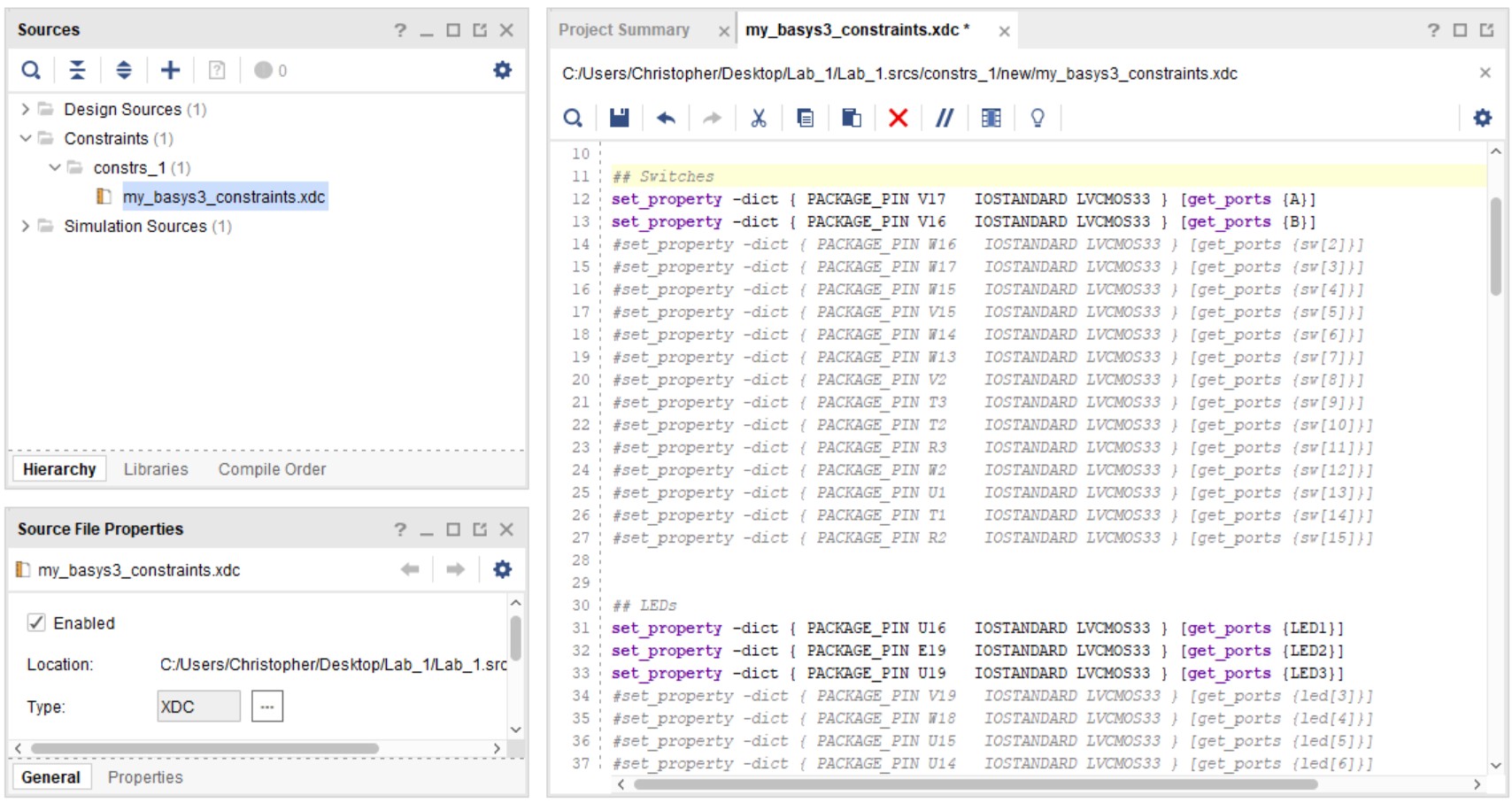
## Design Constraints

Design constraints, such as timing and physical I/O pin mapping, must be defined before doing an implementation, following which the program can be downloaded to the FPGA device. Proceed with the following sequence:

* Expand **PROJECT MANAGER** in the **Flow Navigator** panel, and **click Add Sources**.
* Select **Add or create constraints** and click **Next.**
* Click on **Create File** and give the XDC file a file name, such as **my\_basys3\_constraints**. The XDC format stands for Xilinx Design Constraints.
* Open the **my\_basys3\_constraints.xdc** file from the **Sources** window. It will be an empty .xdc file.
* A template, known as the **Basys3\_Master.xdc** is provided. Open that template using a basic text editor, such as notepad.
* Copy all the contents from that template to your **my\_basys3\_constraints.xdc**. All the lines are commented out by default.
* Link the signals (A, B, LED1, LED2, LED3) of your design, to some physical pins of the FPGA, by uncommenting relevant lines.

Input signals can be linked to switches, whereas the output signals can be linked to LEDs, on the Basys3 development board.

An example of the above steps is shown below:



## Implementation, Bitstream Generation and Program Download

The implementation phase will map the design to available physical resources on the FPGA hardware. In the **Flow Navigator** window, under **IMPLEMENTATION**, select **Run Implementation**. This will make use of the design constraint file that had been created earlier on.

After the implementation phase, there is a need to generate a file that can be downloaded to the FPGA. Such a file is called a bitstream file, and it consists of binary values 0’s and 1’s that tells the FPGA how to behave. In the **Flow Navigator** window, under **PROGRAM AND DEBUG**, select **Generate Bitstream**. A successful bitstream generation is the last step required before downloading the program to the FPGA.

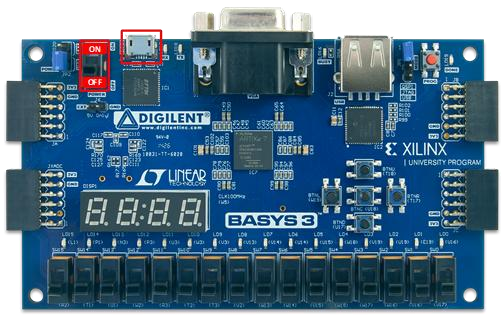
Before using your Basys 3 development board, and to prevent potential damage to it, take note of the following recommendations and warnings to extend the longevity of the device:

⚠ Make sure the Basys 3 development board is powered OFF by placing SW16 in the OFF position before connection to/removal from the USB port of the computer.

⚠ Carefully connect to the USB cable

⚠ The chips on the board are electrostatic sensitive. Avoid touching them. Handle the board by the edges to prevent damage.

⚠ Make sure the board is not in contact with any metal components, whether above or below. Do not place any liquid sources near the FPGA board.



After connection of the Basys 3 development board to the computer, turn on the power by setting SW16 in the ON position. If confirmed to be working, proceed with the following steps:

* Expand **Program and Debug** in the **Flow Navigator Panel**
* Expand **Open Hardware Manager**
* Click **Open Target**
* Select **Auto Connect.** In case connection fails, consider pressing the ‘reset’ button, or turn your device OFF for a few seconds and ON again, while ensuring that it is detected and installed on your computer. Then try **Auto Connect** again
* If successful, the **Program Device** will be enabled, and you will be able to select **xc7a35t\_0**
* By default, if the bitstream was successfully generated, the path name in the **Bitstream file** is automatically provided
* Download the .bit file to the FPGA by clicking on **Program**

### UNDERSTANDING | TASK 6

Your program will then be downloaded to the FPGA. Verify the functionality of the design by using the input devices you have assigned to A, B, and observing the output devices assigned to LED1, LED2 and LED3. Check what happens if the ‘reset’ pushbutton on the Basys 3 development board is pressed, or if power is loss for a short amount of time.

**CLOSING NOTES FOR LAB 1**

Now that you have successfully completed your FPGA design flow, one final practice task is provided to you for completion before ending the lab session. This practice task is not graded.

### FINAL UNDERSTANDING | PRACTICE TASK FOR LAB 1

* **Create a new Vivado project from scratch. Do not reuse your existing project or design**

* The same design as described for the simple boolean design task need to be implemented, with the following exception: There is an additional switch C, and if this switch C is in the OFF state, it forces all the three LEDs to be in the OFF state. If the switch C is in the ON state, the design behaves exactly as described for the simple boolean design task. The switch C is to be mapped to SW[*Your birthday month + 3*] on the Basys 3 development board

* Simulate your design, as well as implement it on the Basys 3 development board

**BEFORE STARTING ON THE GRADED ASSIGNMENT:**

**Each student has a personalised requirement based on student matriculation number. Carefully write down your number, as mistakes in the student matriculation number are not accepted:**

7th Rightmost 6th Rightmost 5th Rightmost 4th Rightmost 3rd Rightmost 2nd Rightmost 1st Rightmost Rightmost

Number Number Number Number Number Number Number Alphabet

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **0** |  |  |  |  |  |  |  |

Five Rightmost Numerical Values (Subtask B)

Students are required to carefully read all the assignment requirements, and to enhance their understanding by looking at the example.

Marks are given for using your correct student matriculation number, and grading is done only once. Re-submissions, late submissions, or updates, even if very simple or minor, are **not accepted** after the grading.

**GRADED POST-LAB ASSIGNMENT**

Complete as much as possible, in **ONE working bitstream for this whole assignment**. It is much better to have a working program with some completed functionalities, instead of submitting a program without a working bitstream (No marks given).

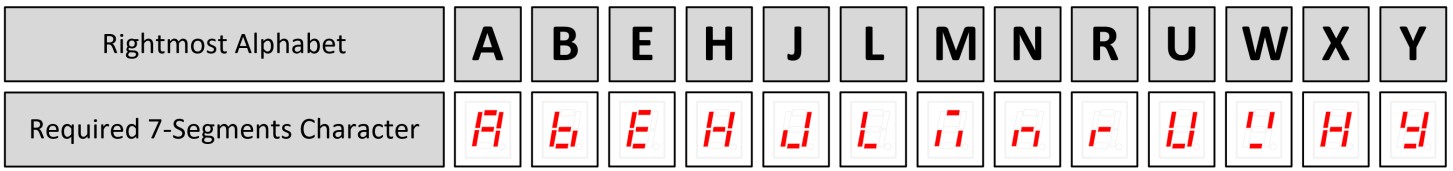
### IMPORTANT CHARACTERS

In this assignment, these are the important characters to note from your student matriculation number:

* The rightmost alphabet of your student matriculation number (Initialistion)
* The five rightmost numerical values of your student matriculation number (Subtask B) • The 1st rightmost numerical value of your student matriculation number (Subtask B)

### INITIALISATION

When the program starts, all 16 active-high switches (SW0 to SW15) are in the OFF position. All 16 active-high LEDs (LD0 to LD15) are also OFF. The seven segment displays must automatically display the **rightmost alphabet** of your student matriculation number on all the 4 (four) anodes of the 7-segment displays. The character must be displayed **exactly** as indicated (Decimal point is OFF):



### SUBTASK A

Consider the 10 (ten) switches SW0 to SW9 (Ignore SW10 to SW15). Whenever any of these 10 switches are ON, the corresponding LED LD**X**, where **X** is a number ranging from 0 to 9, must be ON. (For example, if SW0 is ON, then LD0 must be ON. If SW3, SW7 and SW9 are ON, then LD3, LD7 and LD9 must be ON.) **Do not put constraint to SW10 to SW15, and LD10 to LD15.**

### SUBTASK B

Continuing from SUBTASK A, create your personal student matriculation number password based on the **five** **rightmost numerical values** **of your student matriculation number**.

These five digits (May be less than five digits if you have duplicate numbers) will represent the switches that need to be ON, while all the other switches between SW0 to SW9 must be OFF, to be considered a correct password. Whenever the password is wrong, all the 4 (four) anodes of the 7-segment displays show the **rightmost alphabet**. (For example, if all the ten switches SW0 to SW9 are turned ON, that is considered as a wrong password)

Whenever the password entered by the user is the correct password, then instead of all the 4 (four) anodes of the 7-segment displays showing your rightmost alphabet, only specific anodes of the 7-segment displays must show that alphabet. The specific anodes on which the character should be displayed is dependent on the **1st rightmost numerical value** of your student matriculation number, as indicated in the table below:

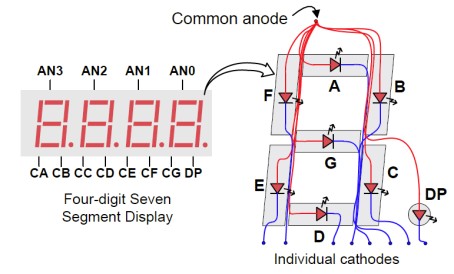
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | 1st Rightmost  Numerical Value | | **0 , 1** | | **2 , 3** | | **4 , 5** | | **6 , 7** | | **8 , 9** | | |  | | --- | | Anode AN3 | | **ON** | | **ON** | | **ON** | | OFF | | OFF | | |  | | --- | | Anode AN2 | | OFF | | OFF | | **ON** | | **ON** | | **ON** | | |  | | --- | | Anode AN1 | | OFF | | **ON** | | OFF | | **ON** | | OFF | | |  | | --- | | Anode AN0 | | **ON** | | OFF | | OFF | | OFF | | **ON** | |

***Note:*** *Marks will* ***not be given*** *if wrong switch constraints are used! If your student matriculation number password has a* ***0****, then it refers to SW****0****. If your student matriculation number password has a* ***1****, then it refers to SW****1****. If your student matriculation number password has a* ***2****, then it refers to SW****2*** *etc…Ensure you have noted down your 5 rightmost numerical values of your student matriculation number accurately for the password.*

### SUGGESTIONS

* Create a new Vivado project for this assignment, instead of continuing from your previous Vivado project.
* This assignment can be fully completed by using SOP / POS expressions.
* This is a warm-up exercise. It is not recommended to use contents not taught in this lab session.
* Simulation is not required in the submission.

### GETTING STARTED WITH THE SEVEN-SEGMENT DISPLAYS

There are 7 LED segments in each display, with an additional decimal point segment. They are respectively denoted by “seg[0]” to “seg[6]”, and “dp”, in the Basys\_Master.xdc constraint file.

There are 4 seven-segment displays on the Basys 3 development board. Each one of the displays is controlled by a common anode pin, thus resulting in a total of 4 common anodes. These active-low pins are denoted as “an[3]” to “an[0]” in the

Basys\_Master.xdc constraint file. (For more information, you can refer to the Basys 3 reference manual, pages 14 to 16)

In your constraint file, it is compulsory to put constraints to the 8 segments (7 segments + decimal point) of the seven-segment display, and to the 4 anodes of the seven-segment display. **Segments or anodes that need to be OFF must explicitly be assigned a value of ‘1’ in your Verilog codes.** For example, if the ‘dot’ segment is required to be off, assign a value of ‘1’ to it.

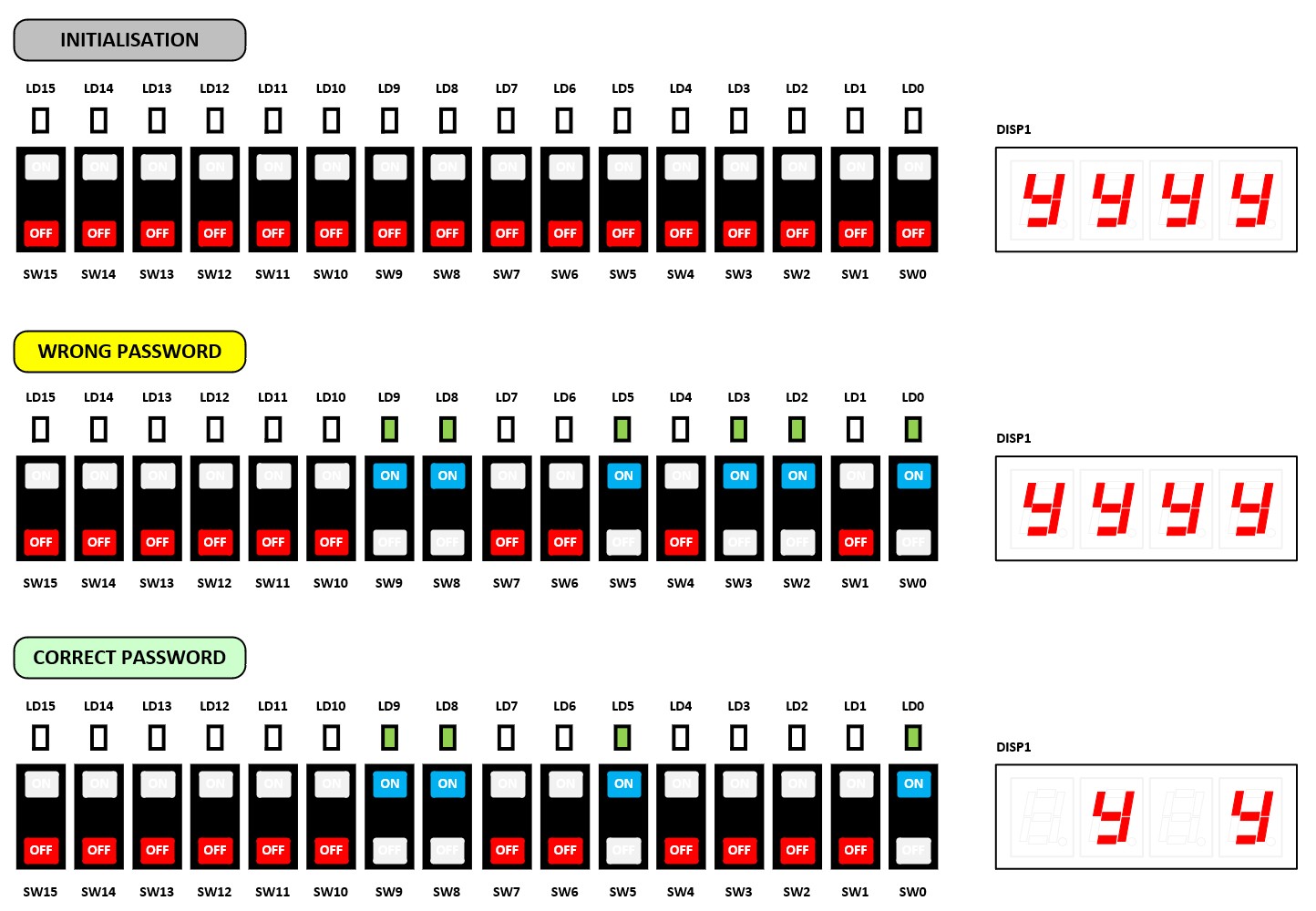
**EXAMPLE:**

If your student matriculation number is A0159089Y, then:

1st rightmost numerical value: **9** (Means only AN2 and AN0 will be ON if password is correct)

Five rightmost numerical values: **59089** (Means password is SW0, SW5, SW8, SW9)

Rightmost alphabet: **Y**



**CANVAS SUBMISSION INSTRUCTIONS**

* Ensure that your bitstream has been successfully generated and tested on your Basys 3 development board **BEFORE** archiving your Vivado workspace for CANVAS upload.

* It is compulsory to archive your project in a compressed form without any saved simulation waveforms. Follow the instructions given in the pdf: “Archive Project in Vivado 2018.02”. **The archive size should not exceed 5 MB in size for any lab assignments.** If the archive size exceeds, ensure that there is no “.sim” folder in your Vivado project folder before archiving.

* **After** following the instructions in “**Archive Project in Vivado 2018.02**”, rename your project archive as indicated in the appendix of this lab manual.

* Upload to CANVAS EE2026 -> Assignments -> Lab 1 Graded Assignment -> Lab 1 Submission (On-Time).

* Download your CANVAS archive after uploading. **Click and drag the single folder within that archive to desktop**, and then open the Vivado project (.xpr) in that **extracted folder** to see if it can be opened. You can ignore “out-of-date” warnings. **Check if you can also run your bitstream correctly.** No project files and no working bitstream is equivalent to losing all marks.

* The CANVAS upload must be completed by **Friday 29th August** **2025, 6:00 A.M. (Morning)**.Avoid planning to upload during the grace period of 2 hours. (Grace period is for last-minute cases, such as due to slow internet, wrong uploads etc.)

* The late submission folder closes 1 week after the original deadline. **Late submissions are not graded if submissions are found in the on-time folder. Do not submit in the late submission folder if you have already submitted in the on-time folder. Inform your lab instructor if you have uploaded in both the on-time folder and late submission folder.** The late submission folder will be located at: CANVAS EE2026 -> Assignments -> Lab 1 Graded Assignment -> Lab 1 Submission (Late).

* Re-submissions, late submissions, or updates, even if very simple or minor, are **not accepted** after your original submission has been graded.

* Submissions through emails, file sharing programs, or links, are **not accepted**. Submissions must be officially submitted to, and downloadable from, Canvas in the correct on-time or late folder.

**Plagiarism is penalised with a 100% penalty for all SOURCES and RECIPIENTS**

All past and future submissions, and marks, will be reviewed in greater detail, for any person found to have plagiarised

**ALL THE SUBMISSION INSTRUCTIONS LISTED ABOVE WILL AFFECT YOUR GRADES!**

\_ \_ \_ \_ \_ \_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_\_ \_ \_ \_ \_ \_

## GRADING PROCESS

* During subsequent lab sessions, our graders will be providing you updates on the grading of your submission.

* Submissions not following all the ***CANVAS SUBMISSION INSTRUCTIONS*** (listed above) will not be graded immediately, and they will instead be reviewed towards the end of the semester. **You will not be able to see your submission results during the lab sessions in such situations**.

**APPENDIX (COMPULSORY renaming just before CANVAS upload):**

It is **compulsory to rename your project archive** just before the CANVAS upload, as listed in the table below.

Do not change any other part of the naming. Simply **copy** the naming from the table below, and **paste** it while renaming your project archive. Penalties will be incurred if your submission cannot be found according to the exact naming template below.

CANVAS may automatically add a suffix (Example. “-1”, “-2”, “-3” etc.) at the end of the filename if you submit multiple files. That is acceptable and we will grade the latest file submitted within that on-time folder.

|  |  |
| --- | --- |
| **Name** | **Archive Naming (.xpr.zip and .zip accepted)** |
| AARAV RAJESH | Lab1\_Mon\_AM\_AARAV RAJESH\_165\_Archive |
| ABE FOO QIAN YIN | Lab1\_Mon\_AM\_ABE FOO QIAN YIN\_426\_Archive |
| ABHIJIT BALAJEE | Lab1\_Mon\_PM\_ABHIJIT BALAJEE\_045\_Archive |
| ABIRAMI BASKARAN | Lab1\_Mon\_PM\_ABIRAMI BASKARAN\_550\_Archive |
| ADIT BISWAS | Lab1\_Fri\_AM\_ADIT BISWAS\_789\_Archive |
| AFSHAL GULAM | Lab1\_Mon\_PM\_AFSHAL GULAM\_376\_Archive |
| AHMAD TIRMIZI BIN ADAM | Lab1\_Mon\_PM\_AHMAD TIRMIZI BIN ADAM\_523\_Archive |
| AIDAN CHIA TONG | Lab1\_Mon\_PM\_AIDAN CHIA TONG\_306\_Archive |
| ALEXANDER YAW KAI MUN | Lab1\_Mon\_PM\_ALEXANDER YAW KAI MUN\_328\_Archive |
| ALIYEV ESHGIN | Lab1\_Mon\_PM\_ALIYEV ESHGIN\_149\_Archive |
| ARIFF MUHAMMED AHSAN HUSAIN | Lab1\_Mon\_PM\_ARIFF MUHAMMED AHSAN HUSA\_407\_Archive |
| ASHLEE CHANG YEE TING | Lab1\_Fri\_AM\_ASHLEE CHANG YEE TING\_686\_Archive |
| AW SHUO JIE | Lab1\_Fri\_AM\_AW SHUO JIE\_803\_Archive |
| AYDEN VAN ETTEN | Lab1\_Mon\_PM\_AYDEN VAN ETTEN\_971\_Archive |
| BADRINATH SANDHYA | Lab1\_Fri\_AM\_BADRINATH SANDHYA\_704\_Archive |
| BAE SOOHUN | Lab1\_Mon\_AM\_BAE SOOHUN\_652\_Archive |
| BAJAJ KRISHNA | Lab1\_Mon\_PM\_BAJAJ KRISHNA\_769\_Archive |
| BENJAMIN CHEK JUN KIET | Lab1\_Mon\_AM\_BENJAMIN CHEK JUN KIET\_381\_Archive |
| BENJAMIN LOH JIAN WEI | Lab1\_Mon\_PM\_BENJAMIN LOH JIAN WEI\_455\_Archive |
| BHADRA PARV | Lab1\_Mon\_PM\_BHADRA PARV\_813\_Archive |
| BINDAWALA AARAV | Lab1\_Fri\_AM\_BINDAWALA AARAV\_779\_Archive |
| BRENDAN TEY SHI HAN | Lab1\_Mon\_AM\_BRENDAN TEY SHI HAN\_957\_Archive |
| BRIEN LIM CHONG | Lab1\_Fri\_AM\_BRIEN LIM CHONG\_162\_Archive |
| CEDRIC TAN SI YU | Lab1\_Mon\_AM\_CEDRIC TAN SI YU\_331\_Archive |
| CHAKRABORTY SHRABASTI | Lab1\_Mon\_PM\_CHAKRABORTY SHRABASTI\_685\_Archive |
| CHAN YI FENG | Lab1\_Mon\_PM\_CHAN YI FENG\_824\_Archive |
| CHANG YI HERNG | Lab1\_Mon\_AM\_CHANG YI HERNG\_612\_Archive |
| CHARNG HE | Lab1\_Mon\_AM\_CHARNG HE\_580\_Archive |
| CHEN GUANWEN | Lab1\_Fri\_AM\_CHEN GUANWEN\_978\_Archive |
| CHEN HONGYU | Lab1\_Fri\_AM\_CHEN HONGYU\_122\_Archive |
| CHEN XINGTONG | Lab1\_Mon\_PM\_CHEN XINGTONG\_961\_Archive |
| CHEN YU HSIN | Lab1\_Mon\_PM\_CHEN YU HSIN\_432\_Archive |
| CHEO ZHI XIAN MATHEU | Lab1\_Fri\_AM\_CHEO ZHI XIAN MATHEU\_851\_Archive |
| CHEONG JIAN HAO | Lab1\_Mon\_PM\_CHEONG JIAN HAO\_240\_Archive |
| CHEW EN WEI | Lab1\_Fri\_AM\_CHEW EN WEI\_658\_Archive |
| CHIA HAO JUN | Lab1\_Mon\_AM\_CHIA HAO JUN\_207\_Archive |
| CHIA LE, ISAAC | Lab1\_Mon\_AM\_CHIA LE ISAAC\_940\_Archive |
| CHNG RUI YONG SEAN | Lab1\_Mon\_PM\_CHNG RUI YONG SEAN\_435\_Archive |
| CHONG KAI JIE | Lab1\_Mon\_PM\_CHONG KAI JIE\_314\_Archive |
| CHONG WEIXUAN, CYDRIC | Lab1\_Mon\_AM\_CHONG WEIXUAN CYDRIC\_871\_Archive |
| CHOY ZHAN HONG | Lab1\_Mon\_PM\_CHOY ZHAN HONG\_444\_Archive |

|  |  |
| --- | --- |
| CHUA YONG LIANG | Lab1\_Mon\_AM\_CHUA YONG LIANG\_014\_Archive |
| CUI JIAHAO | Lab1\_Fri\_AM\_CUI JIAHAO\_085\_Archive |
| DANIEL CHUA ZHENG JIE | Lab1\_Mon\_PM\_DANIEL CHUA ZHENG JIE\_443\_Archive |
| DENG HAOFU | Lab1\_Mon\_PM\_DENG HAOFU\_338\_Archive |
| DENY HAANS HAZRIEL BIN ARMADA | Lab1\_Mon\_PM\_DENY HAANS HAZRIEL BIN AR\_153\_Archive |
| DIWAKAR VIDYA ADHAVAN | Lab1\_Mon\_PM\_DIWAKAR VIDYA ADHAVAN\_335\_Archive |
| DYLAN LIM | Lab1\_Mon\_PM\_DYLAN LIM\_821\_Archive |
| EMRY DANIEL BIN ABDUL LATHIFF | Lab1\_Mon\_AM\_EMRY DANIEL BIN ABDUL LAT\_550\_Archive |
| FOO KANG | Lab1\_Fri\_AM\_FOO KANG\_353\_Archive |
| GABRA SHUBHAN | Lab1\_Mon\_AM\_GABRA SHUBHAN\_258\_Archive |
| GABRIEL LEE JING YI | Lab1\_Mon\_PM\_GABRIEL LEE JING YI\_006\_Archive |
| GAN ANDREW HOA THIEN | Lab1\_Fri\_AM\_GAN ANDREW HOA THIEN\_281\_Archive |
| GOH ANG LEE | Lab1\_Fri\_AM\_GOH ANG LEE\_598\_Archive |
| GOH SHAO ANN | Lab1\_Fri\_AM\_GOH SHAO ANN\_219\_Archive |
| GOH SZE ANH | Lab1\_Mon\_AM\_GOH SZE ANH\_325\_Archive |
| GOH YOU YI | Lab1\_Fri\_AM\_GOH YOU YI\_326\_Archive |
| GORDON HONG JIA JIE | Lab1\_Mon\_PM\_GORDON HONG JIA JIE\_465\_Archive |
| GU MINGYOUJIA | Lab1\_Fri\_AM\_GU MINGYOUJIA\_260\_Archive |
| GUO ZICHENG | Lab1\_Mon\_AM\_GUO ZICHENG\_823\_Archive |
| HANNAH WESTERHOUT HASAN | Lab1\_Mon\_AM\_HANNAH WESTERHOUT HASAN\_413\_Archive |
| HAO YIAN | Lab1\_Mon\_PM\_HAO YIAN\_006\_Archive |
| HIEW T G | Lab1\_Fri\_AM\_HIEW T G\_306\_Archive |
| HOWIE YEO HAO YU | Lab1\_Fri\_AM\_HOWIE YEO HAO YU\_345\_Archive |
| HU LIFAN | Lab1\_Fri\_AM\_HU LIFAN\_030\_Archive |
| HU XIRAN | Lab1\_Fri\_AM\_HU XIRAN\_503\_Archive |
| HUANG HAU SHUAN | Lab1\_Mon\_AM\_HUANG HAU SHUAN\_356\_Archive |
| HUANG YUANJIN | Lab1\_Mon\_AM\_HUANG YUANJIN\_080\_Archive |
| IAN CHOO TZIE ZHENG | Lab1\_Fri\_AM\_IAN CHOO TZIE ZHENG\_747\_Archive |
| IRWAN AHMED NOOR | Lab1\_Mon\_AM\_IRWAN AHMED NOOR\_184\_Archive |
| JAIRUS LEUNG JIE RUI | Lab1\_Fri\_AM\_JAIRUS LEUNG JIE RUI\_337\_Archive |
| JANSEN KEN PEGRASIO | Lab1\_Mon\_AM\_JANSEN KEN PEGRASIO\_566\_Archive |
| JAVIER YEOH ZHI JI | Lab1\_Fri\_AM\_JAVIER YEOH ZHI JI\_430\_Archive |
| JOEL KU | Lab1\_Fri\_AM\_JOEL KU\_232\_Archive |
| JOEL LIM JUN YI | Lab1\_Fri\_AM\_JOEL LIM JUN YI\_423\_Archive |
| JOHN KENNETH LAYBA AGPAOA | Lab1\_Mon\_PM\_JOHN KENNETH LAYBA AGPAOA\_489\_Archive |
| JOSHUA TAM KA YUI | Lab1\_Fri\_AM\_JOSHUA TAM KA YUI\_315\_Archive |
| JOSHUA YEO WEE TZE | Lab1\_Fri\_AM\_JOSHUA YEO WEE TZE\_878\_Archive |
| JOVIAN JOSH | Lab1\_Mon\_AM\_JOVIAN JOSH\_585\_Archive |
| KARTHIK KATHIRESH | Lab1\_Mon\_AM\_KARTHIK KATHIRESH\_122\_Archive |
| KARTHIKEYAN VETRIVEL | Lab1\_Mon\_PM\_KARTHIKEYAN VETRIVEL\_899\_Archive |
| KENNETH WONG CUN WI | Lab1\_Fri\_AM\_KENNETH WONG CUN WI\_742\_Archive |
| KEVIN LOKE WEI YI | Lab1\_Mon\_AM\_KEVIN LOKE WEI YI\_957\_Archive |
| KHOO JUNHAO | Lab1\_Fri\_AM\_KHOO JUNHAO\_052\_Archive |
| KOH LI TIAN | Lab1\_Mon\_AM\_KOH LI TIAN\_429\_Archive |
| KOTHARI SMEET RONAK | Lab1\_Fri\_AM\_KOTHARI SMEET RONAK\_661\_Archive |
| KUAH JUN HONG, BRYAN | Lab1\_Fri\_AM\_KUAH JUN HONG BRYAN\_384\_Archive |
| LABELLE LEE | Lab1\_Mon\_AM\_LABELLE LEE\_619\_Archive |
| LAM ZHEN LEI, ETHAN | Lab1\_Fri\_AM\_LAM ZHEN LEI ETHAN\_558\_Archive |
| LAU EN XIN, GRACE | Lab1\_Mon\_PM\_LAU EN XIN GRACE\_497\_Archive |
| LEE KAH HOE, BRIAN | Lab1\_Fri\_AM\_LEE KAH HOE BRIAN\_300\_Archive |
| LEE KUAN YI | Lab1\_Mon\_PM\_LEE KUAN YI\_201\_Archive |
| LEONARD LIM TZE YANG | Lab1\_Mon\_PM\_LEONARD LIM TZE YANG\_460\_Archive |
| LI LINYING | Lab1\_Mon\_PM\_LI LINYING\_166\_Archive |

|  |  |
| --- | --- |
| LI XIAOYANG | Lab1\_Mon\_PM\_LI XIAOYANG\_853\_Archive |
| LIM KAI LER, ETHAN | Lab1\_Fri\_AM\_LIM KAI LER ETHAN\_939\_Archive |
| LIM SWEE HOW GABRIEL | Lab1\_Mon\_PM\_LIM SWEE HOW GABRIEL\_173\_Archive |
| LIM ZE EN, MATTHIAS | Lab1\_Mon\_AM\_LIM ZE EN MATTHIAS\_943\_Archive |
| LIU LEKUAN | Lab1\_Mon\_PM\_LIU LEKUAN\_703\_Archive |
| LIU YIHAN | Lab1\_Fri\_AM\_LIU YIHAN\_493\_Archive |
| LOH HAN XIANG | Lab1\_Mon\_AM\_LOH HAN XIANG\_229\_Archive |
| LOU YU | Lab1\_Fri\_AM\_LOU YU\_962\_Archive |
| LOUIS AGARA PERIN | Lab1\_Mon\_AM\_LOUIS AGARA PERIN\_320\_Archive |
| LOW ZEN WEI | Lab1\_Mon\_AM\_LOW ZEN WEI\_635\_Archive |
| LU QIANXI | Lab1\_Mon\_AM\_LU QIANXI\_033\_Archive |
| LUO HONGXUN | Lab1\_Mon\_AM\_LUO HONGXUN\_391\_Archive |
| MAHESH PURAV | Lab1\_Fri\_AM\_MAHESH PURAV\_679\_Archive |
| MANDI SIDDID UMESH | Lab1\_Mon\_AM\_MANDI SIDDID UMESH\_173\_Archive |
| MANU DAGUR | Lab1\_Mon\_PM\_MANU DAGUR\_861\_Archive |
| MAO XIAOHAN | Lab1\_Mon\_PM\_MAO XIAOHAN\_110\_Archive |
| MARK NG JIAN XIONG | Lab1\_Fri\_AM\_MARK NG JIAN XIONG\_883\_Archive |
| MATHEW SAAYUJ ION | Lab1\_Mon\_AM\_MATHEW SAAYUJ ION\_282\_Archive |
| MICHAEL SHYAM WILFRED DAVID SAMUVEL | Lab1\_Fri\_AM\_MICHAEL SHYAM WILFRED DAV\_218\_Archive |
| MO HANQI | Lab1\_Mon\_AM\_MO HANQI\_531\_Archive |
| MOHAMED FARAS S/O FARIDUL HUK | Lab1\_Mon\_AM\_MOHAMED FARAS SO FARIDUL\_806\_Archive |
| MUHAMMAD AKMAL HANIS BIN MD JOHANI | Lab1\_Fri\_AM\_MUHAMMAD AKMAL HANIS BIN \_123\_Archive |
| NAILA FAIQAH BINTE MUHAMMAD AZHAR | Lab1\_Mon\_PM\_NAILA FAIQAH BINTE MUHAMM\_406\_Archive |
| NAVANEETHAN SANJAI | Lab1\_Mon\_AM\_NAVANEETHAN SANJAI\_231\_Archive |
| NEERAJ VENGADESSAN | Lab1\_Mon\_AM\_NEERAJ VENGADESSAN\_959\_Archive |
| NI SHI YONG | Lab1\_Mon\_AM\_NI SHI YONG\_596\_Archive |
| NICHOLAS LAU HONGYI | Lab1\_Mon\_PM\_NICHOLAS LAU HONGYI\_324\_Archive |
| NOCHUR SIVANSH | Lab1\_Mon\_PM\_NOCHUR SIVANSH\_332\_Archive |
| ONG CHONG YAO | Lab1\_Mon\_PM\_ONG CHONG YAO\_901\_Archive |
| ONG XIANG KAI | Lab1\_Mon\_AM\_ONG XIANG KAI\_232\_Archive |
| PANG ANG SHENG ASHER | Lab1\_Fri\_AM\_PANG ANG SHENG ASHER\_237\_Archive |
| PRANAV JANAKIRAMAN | Lab1\_Mon\_PM\_PRANAV JANAKIRAMAN\_346\_Archive |
| PREMIL ROSHAN | Lab1\_Mon\_AM\_PREMIL ROSHAN\_805\_Archive |
| PRERANA RAVI SHANKAR | Lab1\_Fri\_AM\_PRERANA RAVI SHANKAR\_092\_Archive |
| RAJAN PRAVEEN | Lab1\_Mon\_PM\_RAJAN PRAVEEN\_146\_Archive |
| RAJARAM SUSHMIITHAA | Lab1\_Mon\_PM\_RAJARAM SUSHMIITHAA\_309\_Archive |
| RAJESH KUMAR ASWIN | Lab1\_Mon\_PM\_RAJESH KUMAR ASWIN\_713\_Archive |
| REHAAN MAHMOOD | Lab1\_Mon\_PM\_REHAAN MAHMOOD\_719\_Archive |
| RISHABH RAMPRASAD SHENOY | Lab1\_Fri\_AM\_RISHABH RAMPRASAD SHENOY\_597\_Archive |
| ROHAN H | Lab1\_Mon\_AM\_ROHAN H\_155\_Archive |
| ROSHAN ALAGAR PREMKUMAR | Lab1\_Mon\_PM\_ROSHAN ALAGAR PREMKUMAR\_245\_Archive |
| RUSSELL NG JUN HENG | Lab1\_Fri\_AM\_RUSSELL NG JUN HENG\_204\_Archive |
| RYAN KOH JUN HAO | Lab1\_Mon\_PM\_RYAN KOH JUN HAO\_601\_Archive |
| RYAN PANG ZE XI | Lab1\_Mon\_PM\_RYAN PANG ZE XI\_453\_Archive |
| RYAN TAN RONG CHANG | Lab1\_Mon\_PM\_RYAN TAN RONG CHANG\_406\_Archive |
| SAMUEL LEE WEN JIN | Lab1\_Mon\_AM\_SAMUEL LEE WEN JIN\_232\_Archive |
| SANGHI NISHCHAY | Lab1\_Mon\_PM\_SANGHI NISHCHAY\_666\_Archive |
| SARAVANAN RAJESHWARI AKSHAY PRANAV | Lab1\_Mon\_AM\_SARAVANAN RAJESHWARI AKSH\_513\_Archive |
| SEAN LEE WEI SHU | Lab1\_Mon\_PM\_SEAN LEE WEI SHU\_252\_Archive |
| SEAN TAN KAI JIE | Lab1\_Fri\_AM\_SEAN TAN KAI JIE\_767\_Archive |
| SEAN TAN LIYU | Lab1\_Fri\_AM\_SEAN TAN LIYU\_776\_Archive |
| SEOW JACKIE JAVIER | Lab1\_Mon\_PM\_SEOW JACKIE JAVIER\_092\_Archive |
| SHAH JAINAM AMIT | Lab1\_Mon\_PM\_SHAH JAINAM AMIT\_339\_Archive |

|  |  |
| --- | --- |
| SHAH KUSHAL HITESH | Lab1\_Fri\_AM\_SHAH KUSHAL HITESH\_473\_Archive |
| SHAUN TAN SHU REN | Lab1\_Mon\_PM\_SHAUN TAN SHU REN\_075\_Archive |
| SHENNON TAY | Lab1\_Fri\_AM\_SHENNON TAY\_016\_Archive |
| SHYAMAL VARNAN VENKATARAMAN | Lab1\_Mon\_AM\_SHYAMAL VARNAN VENKATARAM\_949\_Archive |
| SINGH SIDDHANT NARAYAN | Lab1\_Mon\_PM\_SINGH SIDDHANT NARAYAN\_752\_Archive |
| SRINIVASAN RANGANATHAN | Lab1\_Mon\_AM\_SRINIVASAN RANGANATHAN\_330\_Archive |
| SRIVASTAVA HARSHIT | Lab1\_Mon\_AM\_SRIVASTAVA HARSHIT\_749\_Archive |
| SUWEI SHRESTHA | Lab1\_Mon\_PM\_SUWEI SHRESTHA\_946\_Archive |
| TAN CHUN LIANG | Lab1\_Fri\_AM\_TAN CHUN LIANG\_676\_Archive |
| TAN KAI CONG | Lab1\_Fri\_AM\_TAN KAI CONG\_476\_Archive |
| TAN PANG | Lab1\_Mon\_AM\_TAN PANG\_905\_Archive |
| TAN WEI HENG | Lab1\_Mon\_AM\_TAN WEI HENG\_981\_Archive |
| TAN YAN HAO MALCOLM | Lab1\_Mon\_AM\_TAN YAN HAO MALCOLM\_363\_Archive |
| TAN YUE YANG | Lab1\_Mon\_PM\_TAN YUE YANG\_427\_Archive |
| TANPRASERTKUL PRAN | Lab1\_Fri\_AM\_TANPRASERTKUL PRAN\_173\_Archive |
| TAO ENZE | Lab1\_Mon\_PM\_TAO ENZE\_746\_Archive |
| TEO YU XIANG ALOYSIUS | Lab1\_Fri\_AM\_TEO YU XIANG ALOYSIUS\_442\_Archive |
| THEN CHIN KIAT | Lab1\_Mon\_PM\_THEN CHIN KIAT\_534\_Archive |
| THIA YANG HAN | Lab1\_Fri\_AM\_THIA YANG HAN\_615\_Archive |
| TIRODKAR OM MILIND | Lab1\_Mon\_AM\_TIRODKAR OM MILIND\_212\_Archive |
| TJHIN BRIAN | Lab1\_Fri\_AM\_TJHIN BRIAN\_808\_Archive |
| TOH EE SEN, IZEN | Lab1\_Fri\_AM\_TOH EE SEN IZEN\_600\_Archive |
| TOH YI WEI | Lab1\_Mon\_PM\_TOH YI WEI\_295\_Archive |
| TONG KAR YUE ABIGAIL | Lab1\_Mon\_AM\_TONG KAR YUE ABIGAIL\_126\_Archive |
| VANSH PURI | Lab1\_Mon\_PM\_VANSH PURI\_003\_Archive |
| VENKATESH KSHEERABTHI NATHAN | Lab1\_Mon\_AM\_VENKATESH KSHEERABTHI NAT\_380\_Archive |
| VIHAAN | Lab1\_Mon\_AM\_VIHAAN\_665\_Archive |
| VINAY VANJRE RAVI | Lab1\_Fri\_AM\_VINAY VANJRE RAVI\_336\_Archive |
| WAI YAN | Lab1\_Fri\_AM\_WAI YAN\_450\_Archive |
| WANG CHUHAO | Lab1\_Fri\_AM\_WANG CHUHAO\_566\_Archive |
| WANG JIAWEI | Lab1\_Mon\_AM\_WANG JIAWEI\_498\_Archive |
| WANG PENGJIN | Lab1\_Fri\_AM\_WANG PENGJIN\_104\_Archive |
| WANG ZAIXI | Lab1\_Mon\_PM\_WANG ZAIXI\_940\_Archive |
| WEN JUN YU | Lab1\_Mon\_PM\_WEN JUN YU\_517\_Archive |
| WENG XIAN YANG | Lab1\_Fri\_AM\_WENG XIAN YANG\_191\_Archive |
| WESLEY LOW | Lab1\_Mon\_PM\_WESLEY LOW\_090\_Archive |
| WONG EE SHAWN | Lab1\_Mon\_PM\_WONG EE SHAWN\_244\_Archive |
| WU ZI EN, ELLIOT JOHN | Lab1\_Mon\_AM\_WU ZI EN ELLIOT JOHN\_902\_Archive |
| XU HUANGHAO | Lab1\_Mon\_PM\_XU HUANGHAO\_370\_Archive |
| XU ZIHAO | Lab1\_Mon\_PM\_XU ZIHAO\_485\_Archive |
| XYLON CHAN YI CHONG | Lab1\_Fri\_AM\_XYLON CHAN YI CHONG\_514\_Archive |
| YADAV ARYAN SUNILKUMAR | Lab1\_Mon\_PM\_YADAV ARYAN SUNILKUMAR\_694\_Archive |
| YAN XIANGYU | Lab1\_Mon\_PM\_YAN XIANGYU\_416\_Archive |
| YAO XIANG | Lab1\_Mon\_PM\_YAO XIANG\_192\_Archive |
| YAP JIT EN FAITH | Lab1\_Mon\_PM\_YAP JIT EN FAITH\_368\_Archive |
| YEE JIA JIE | Lab1\_Mon\_AM\_YEE JIA JIE\_896\_Archive |
| YEO CHEN XIAN | Lab1\_Mon\_PM\_YEO CHEN XIAN\_213\_Archive |
| YEO SI ZHAO | Lab1\_Fri\_AM\_YEO SI ZHAO\_884\_Archive |
| YEO TECK IAN BRYAN | Lab1\_Fri\_AM\_YEO TECK IAN BRYAN\_832\_Archive |
| YEO YEE CHING | Lab1\_Mon\_AM\_YEO YEE CHING\_332\_Archive |
| YEOH SOO LEONG | Lab1\_Mon\_PM\_YEOH SOO LEONG\_125\_Archive |
| YEUNG KEITH | Lab1\_Mon\_AM\_YEUNG KEITH\_484\_Archive |
| YI YANG | Lab1\_Mon\_PM\_YI YANG\_914\_Archive |
| ZHANG YIZE | Lab1\_Fri\_AM\_ZHANG YIZE\_948\_Archive |
| ZHAO ZEYU | Lab1\_Mon\_AM\_ZHAO ZEYU\_093\_Archive |
| ZHENG HUIJIE | Lab1\_Fri\_AM\_ZHENG HUIJIE\_077\_Archive |
| ZHENG KAIWEN | Lab1\_Fri\_AM\_ZHENG KAIWEN\_674\_Archive |
| ZHU YICHENG | Lab1\_Fri\_AM\_ZHU YICHENG\_485\_Archive |

If you have registered for EE2026 labs late, your name may not be present in this lab manual. It will be included in subsequent lab manuals. Please use this naming template meanwhile: Lab1\_*Name as on Matriculation Card*\_Archive

**-------------------------------------------------------------------------------------**

**FINAL CHECKLIST FOR ASSIGNMENT 1:**

**Grading is done only once**. Re-grading due to not meeting the checklist below are not accepted.

|  |  |
| --- | --- |
| □ | I have used the **exact and** **correct numbers and alphabet** from my student matriculation number for the assignment. |
| □ | I have named the submitted archive **according to the given template**. |
| □ | The submitted archive is **less than 5 MB** (Delete “.sim” folder if more than 5 MB. Penalty of up to 10% apply if ≥5 MB). |
| □ | I confirm that the uploaded **submission is the correct / latest version**, and not an older one. |
| □ | I have **downloaded my submission** before the deadline and confirmed that it was properly uploaded to canvas. |
| □ | I confirm that the **grader need not “Generate bitstream”**, as the bitstream is already there during “Program Device”. |
| □ | I confirm that I have carefully verified the bitstream in my **CANVAS submission**, and it works exactly as I expected. |