LAB VENUE:
DIGITAL ELECTRONICS LAB @ E4-03-07

LAB SCHEDULE:

Week and Date	Activity	Session
Week 1: 11 Aug 2025	No Scheduled Lab	Mon **A.M.**
Week 1: 11 Aug 2025	No Scheduled Lab	Mon **P.M.**
Week 1: 15 Aug 2025	No Scheduled Lab	Fri **A.M.**
Week 2: 18 Aug 2025	No Scheduled Lab	Mon **A.M.**
Week 2: 18 Aug 2025	No Scheduled Lab	Mon **P.M.**
Week 2: 22 Aug 2025	**Lab 1:** Getting Started	Fri **A.M.**
Week 3: 25 Aug 2025	**Lab 1:** Getting Started	Mon **A.M.**
Week 3: 25 Aug 2025	**Lab 1:** Getting Started	Mon **P.M.**
Week 3: 29 Aug 2025	**Lab 2:** Combinational Circuits	Fri **A.M.**
Week 4: 1 Sep 2025	**Lab 2:** Combinational Circuits	Mon **A.M.**
Week 4: 1 Sep 2025	**Lab 2:** Combinational Circuits	Mon **P.M.**
Week 4: 5 Sep 2025	No Scheduled Lab	Fri **A.M.**
Week 5: 8 Sep 2025	No Scheduled Lab	Mon **A.M.**
Week 5: 8 Sep 2025	No Scheduled Lab	Mon **P.M.**
Week 5: 12 Sep 2025	**Lab 3:** Sequential Circuits	Fri **A.M.**
Week 6: 15 Sep 2025	**Lab 3:** Sequential Circuits	Mon **A.M.**
Week 6: 15 Sep 2025	**Lab 3:** Sequential Circuits	Mon **P.M.**
Week 6: 19 Sep 2025	No Scheduled Lab	Fri **A.M.**
Recess Week	No Scheduled Lab	Mon **A.M.**
	No Scheduled Lab	Mon **P.M.**
	No Scheduled Lab	Fri **A.M.**
Week 7: 29 Sep 2025	**P1:** Introduction to BASYS3 I/O Devices	Mon **A.M.**
Week 7: 29 Sep 2025	**P1:** Introduction to BASYS3 I/O Devices	Mon **P.M.**
Week 7: 3 Oct 2025	**P1:** Introduction to BASYS3 I/O Devices	Fri **A.M.**
Week 8: 6 Oct 2025	**P2:** Basic Tasks Evaluation	Mon **A.M.**
Week 8: 6 Oct 2025	**P2:** Basic Tasks Evaluation	Mon **P.M.**
Week 8: 10 Oct 2025	**P2:** Basic Tasks Evaluation	Fri **A.M.**
Week 9: 13 Oct 2025	**P3:** Project Progress Evaluation	Mon **A.M.**
Week 9: 13 Oct 2025	**P3:** Project Progress Evaluation	Mon **P.M.**
Week 9: 17 Oct 2025	**P3:** Project Progress Evaluation	Fri **A.M.**
Week 10: 20 Oct 2025	No Scheduled Lab	Mon **A.M.**
Week 10: 20 Oct 2025	No Scheduled Lab	Mon **P.M.**
Week 10: 24 Oct 2025	**P4:** Verilog Evaluation	Fri **A.M.**
Week 11: 27 Oct 2025	**P4:** Verilog Evaluation	Mon **A.M.**
Week 11: 27 Oct 2025	**P4:** Verilog Evaluation	Mon **P.M.**
Week 11: 31 Oct 2025	No Scheduled Lab	Fri **A.M.**
Week 12: 3 Nov 2025	No Scheduled Lab	Mon **A.M.**
Week 12: 3 Nov 2025	No Scheduled Lab	Mon **P.M.**
Week 12: 7 Nov 2025	No Scheduled Lab	Fri **A.M.**
Week 13	Project Assessment	-

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:

- When switch **A** turns on, only **LED1** lights up.
- When switch **B** turns on, only **LED2** lights up.
- When both switches **A** and **B** turn on, **LED1**, **LED2**, and **LED3** light up.

UNDERSTANDING | TASK 1

Complete the truth table for the simple boolean design task:

INPUT		OUTPUT			MINTERM
A	B	LED1	LED2	LED3	
0	0				$\bar{A}\bar{B}$
0	1				$\bar{A}B$
1	0				$A\bar{B}$
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**:

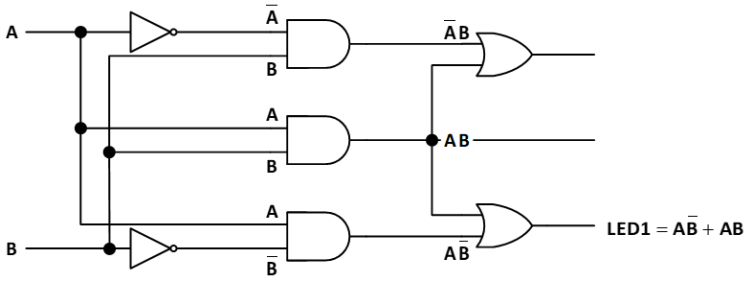
$$\mathbf{LED1} = A\bar{B} + AB$$

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



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

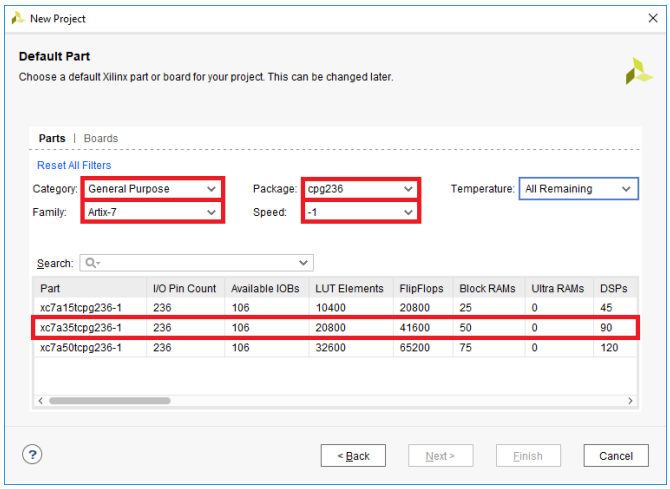
Project Type: Select **RTL Project**, and uncheck "**Do not specify sources at this time**"

Add sources:

- **Create File.** File type is Verilog. Example: simple_boolean
- **Target language:** Verilog. **Simulator language:** Mixed

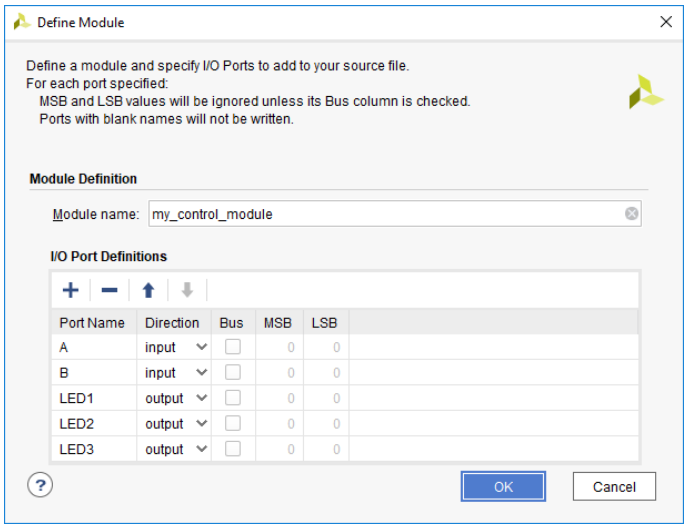
Add constraints (optional): Click on next without any changes

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

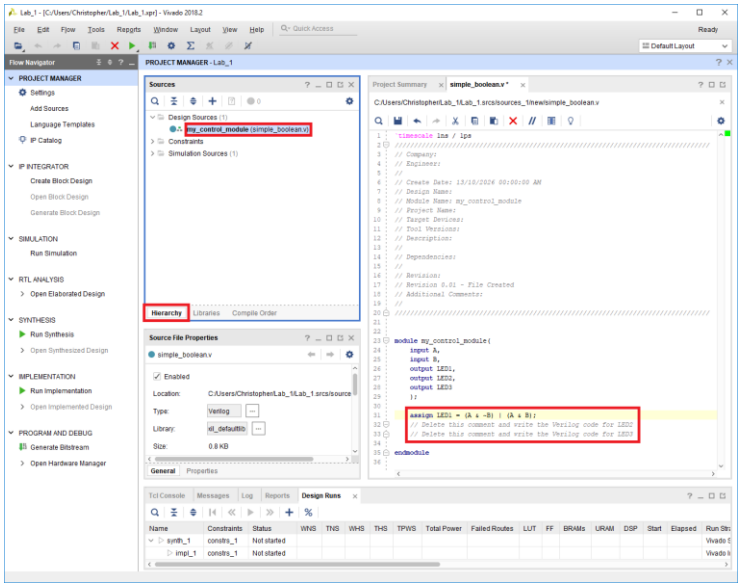
Open the module that has been created by double clicking on it in the Sources window

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	$\mathbf{A + B}$	|
AND	$\mathbf{AB}$	&
NOT	$\mathbf{\bar{A}}$	~
XOR	$\mathbf{A \oplus B}$	^



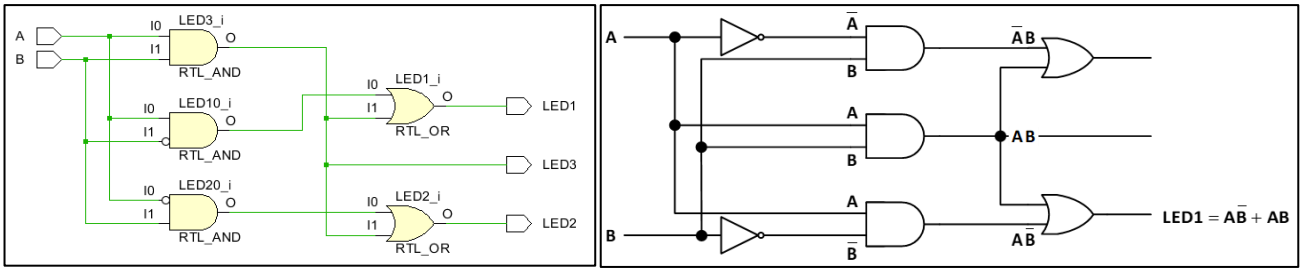
The **assign** statement causes the left hand side of the expression to be updated *every* time there is a change on the right hand side of the expression. It is therefore called a *continuous assignment* statement, describing combinational logic whereby the output on the left is a function of current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

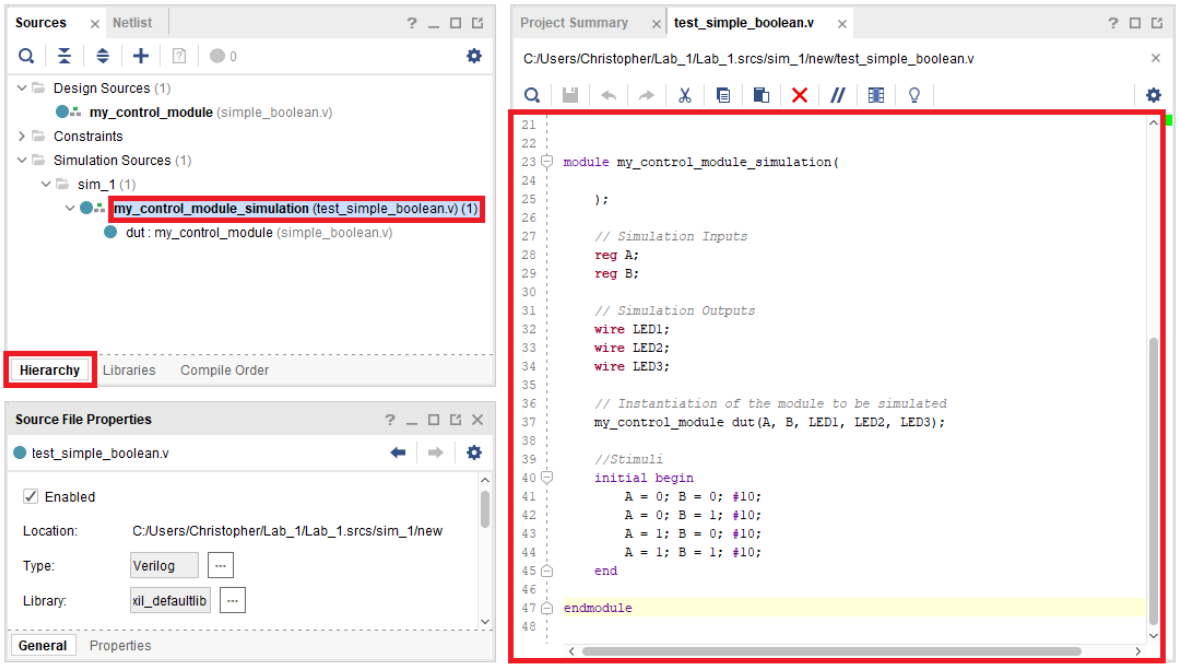


Testbench and Behavioural Simulation

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

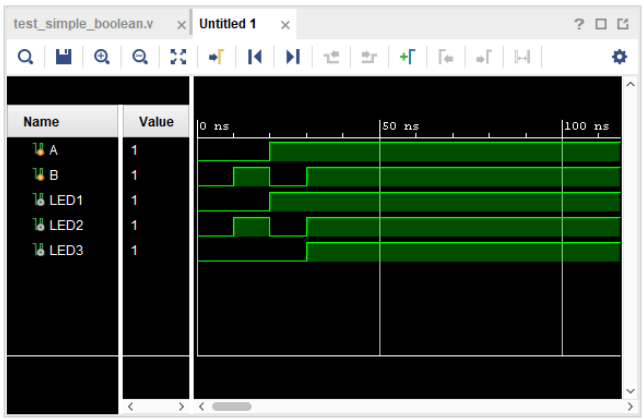
From the Sources window, open the simulation file. Then, within the simulation module, provide the following codes and save them, with the final screenshot looking similar to the image shown below:



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.



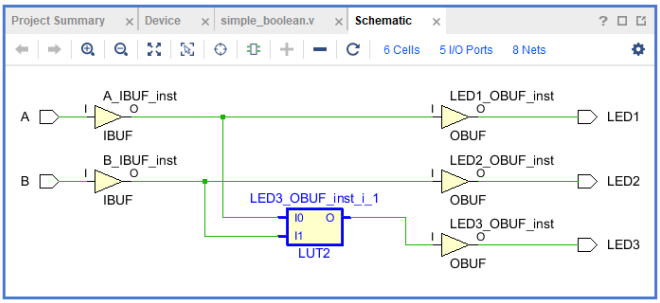
Synthesis

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.

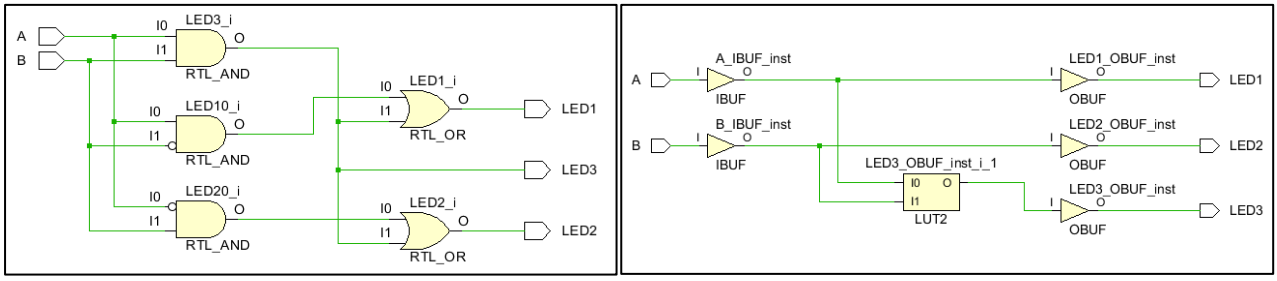
After the synthesis has been successfully completed, in the **Flow Navigator** window, under **SYNTHESIS**, expand **Open Synthesized Design**, and select **Schematics**. The schematic of the synthesised design will be generated and this synthesised circuit is an optimised version of the RTL schematic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

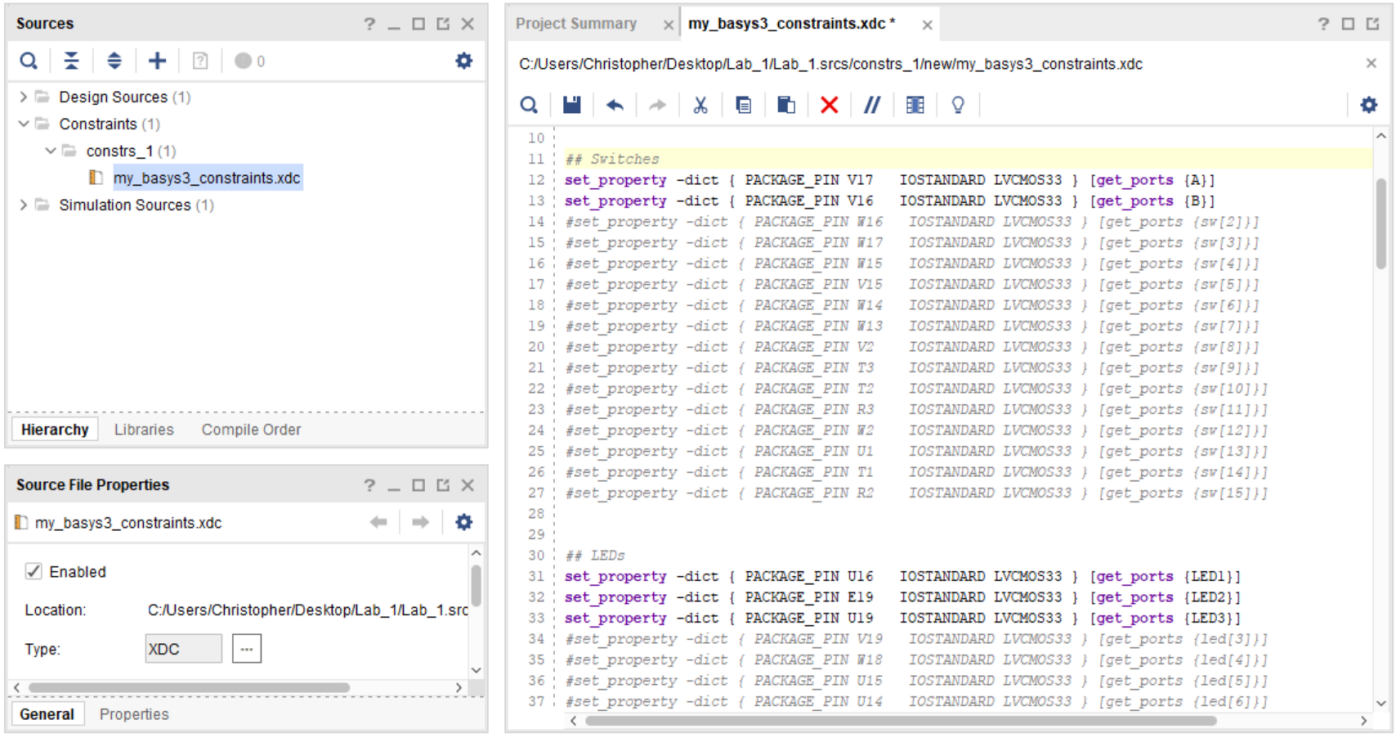


Design Constraints

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.

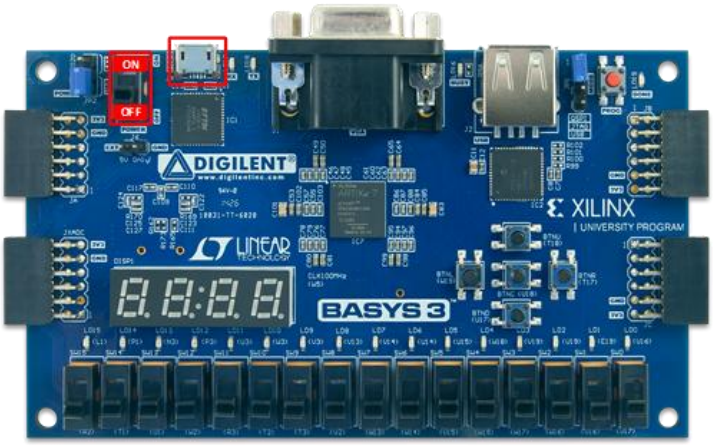
Before using your Basys 3 development board, and to prevent potential damage to it, take note of the following recommendations and warnings to extend the longevity of the device:

⚠ 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

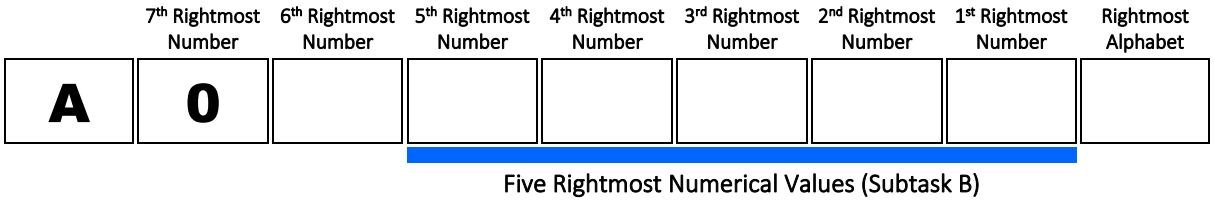
- **Create a new Vivado project from scratch. Do not reuse your existing project or design**

- 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:



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):

Rightmost Alphabet	A	B	E	H	J	L	M	N	R	U	W	X	Y
Required 7-Segments Character													

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:

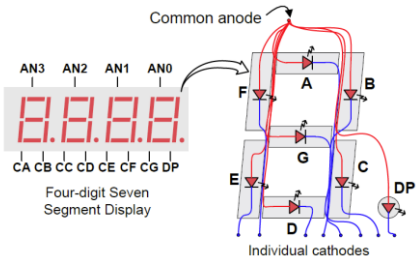
1st Rightmost Numerical Value	Anode AN3	Anode AN2	Anode AN1	Anode AN0
0 , 1	**ON**	OFF	OFF	**ON**
2 , 3	**ON**	OFF	**ON**	OFF
4 , 5	**ON**	**ON**	OFF	OFF
6 , 7	OFF	**ON**	**ON**	OFF
8 , 9	OFF	**ON**	OFF	**ON**

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)

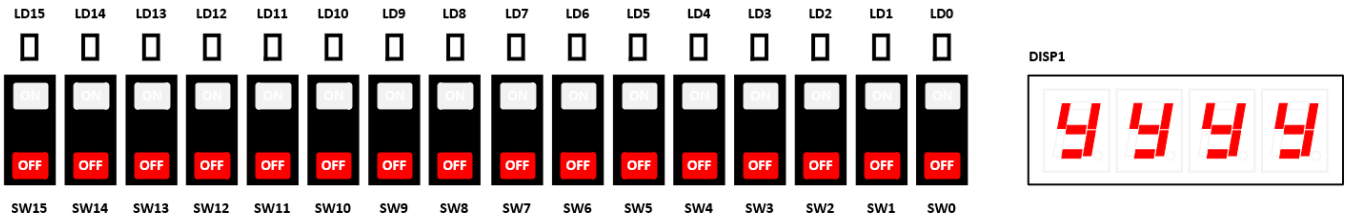
In your constraint file, it is compulsory to put constraints to the 8 segments (7 segments + decimal point) of the seven-segment display, and to the 4 anodes of the seven-segment display. **Segments or anodes that need to be OFF must explicitly be assigned a value of ‘1’ in your Verilog codes.** For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

EXAMPLE:

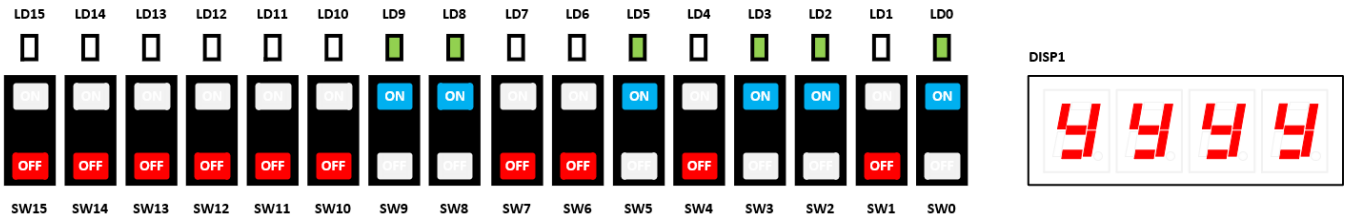
If your student matriculation number is A0159089Y, then:

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**

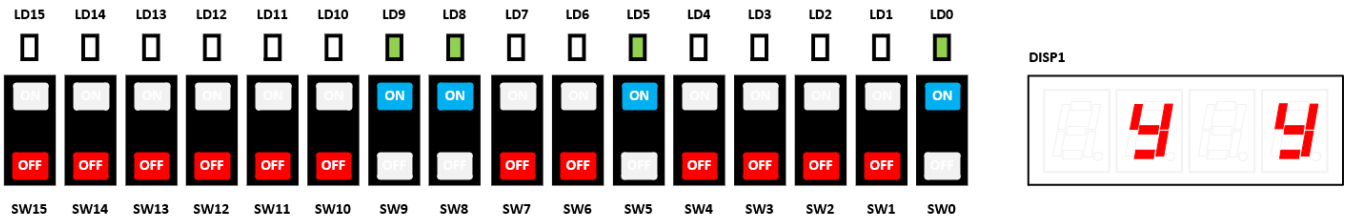
INITIALISATION



WRONG PASSWORD



CORRECT PASSWORD



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.