數位電路設計 (Digital Circuit Design)
Lab2：組合電路之 HDL 模組撰寫與測試
(Writing and Testing the HDL Modules of Combinational Circuits)

1. 目標 (Goal)

在這次 Lab 中，我們希望同學們可以熟悉組合電路的設計原理，以 **gate-level modeling**、**dataflow modeling**、**behavioral modeling** 等不同方式撰寫其 HDL 電路模組，並撰寫測試模組。我們以二元加法器及二元編碼十進位數字加減法器為設計實例，分別模擬後，繳交模組檔案與波形圖。

The main purpose of this Lab Unit is to be familiar with the design of Combinational Circuits. Please write their HDL circuit modules by gate-level modeling, dataflow modeling, and behavioral modeling, and write the testbench for these circuit modules. We take binary adder and BCD (Binary-Coded Decimal) adder-subtractor for practice. After simulation, upload the files of the modules and the waveforms of the simulation results.

2. 撰寫 HDL 電路模組與測試模組 (Design of the HDL Circuit Modules and Testbench)

A. 二元加法器之設計 (Design of Binary Adder)

相關設計可參考教科書§4.5與§4.12小節。(Refer to Section 4.5 and 4.12 of textbook for the design of binary adder.)

(a) **半加器(Half Adder)**：設計一個具有延遲的半減器(half adder, HA)，有兩個輸入變數及兩個輸出變數。輸入變數為被加數 ***a*** 與加數 ***b***；輸出變數為和值 ***sum*** 與進位輸出 ***cout***。

Design a half adder (HA) with propagation delay which has 2 input variables and 2 output variables. The input variables are the augend (被加數) bit ***a*** and the addend (加數) bit ***b***, and the output variables produce the sum bit ***sum*** and the carry-out bit ***cout***.

i. 請以 **gate-level modeling** 方式撰寫其 HDL 電路模組，其中 AND 與 OR 邏輯閘之延遲設為 2 個單位時間，XOR 與 XNOR 邏輯閘之延遲設為 4 個單位時間，NOT 邏輯閘之延遲則忽略之。模組名稱與 port list 請訂為 Lab2_half_add (input a, b, output sum, cout)，檔案則命名為 Lab2_half_add.v。

Please write the Verilog circuit module in **gate-level modeling**. Assume that the propagation delay of AND and OR gates is 2 time units, the propagation delay of XOR and XNOR gates is 4 time units, and the propagation delay of NOT gates is ignored. The circuit module and port list should be named as Lab2_half_add (input a, b, output sum, cout), and its file should be named as Lab2_half_add.v.

ii. 請撰寫一測試模組來**完整測試**此半加器電路模組；請將此測試模組命名為 t_Lab2_half_add，檔案則命名為 t_Lab2_half_add.v。

Please write a testbench to test the half-adder circuit module thoroughly. The testbench module should be named as t_Lab2_half_add, and its file should be named as t_Lab2_half_add.v.

(b) **全加器(Full Adder)**：設計一個具有延遲的全加器(full adder, FA)，有三個輸入變數及兩個輸出變數。輸入變數為被加數 ***a***、加數 ***b***、及進位輸入 ***cin***；輸出變數為和值 ***sum*** 與進位輸出 ***cout***。

Design a full adder (FA) with propagation delay which has 3 input variables and 2 output variables. The input variables are the augend (被加數) bit ***a***, the addend (加數) bit ***b***, and the carry-in bit ***cin***, and the output variables produce the sum bit ***sum*** and the carry-out bit ***cout***.

i. 請利用**(a)**中的 gate-level 半加器來建構全加器。電路中 AND 與 OR 邏輯閘之延遲設為 2 個單位時間，XOR 與 XNOR 邏輯閘之延遲設為 4 個單位時間，NOT 邏輯閘之延遲則忽略之。請撰寫出全加器之 HDL 電路模組，模組名稱與 port list 請訂為 Lab2_full_add (input a, b, cin, output sum, cout)，檔案則命名為 Lab2_full_add.v。
Please use the gate-level half-adder designed in **(a)** to construct the full-adder. Assume that the propagation delay of AND and OR gates is 2 time units, the propagation delay of XOR and XNOR gates is 4 time units, and the propagation delay of NOT gates is ignored. Please write the HDL circuit module of the full-subtractor. The circuit module and port list should be named as Lab2_full_add (input a, b, cin, output sum, cout), and its file should be named as Lab2_full_add.v.

ii. 請撰寫一測試模組來**完整測試**此全加器電路模組；請將此測試模組命名為 t_Lab2_full_add，檔案則命名為 t_Lab2_full_add.v。
Please write a testbench to test the full-adder circuit module thoroughly. The testbench module should be named as t_Lab2_full_add, and its file should be named as t_Lab2_full_add.v.

(c) **四位元漣波進位加法器(4-bit Ripple Carry Adder, RCA)**：設計一個四位元漣波進位加法器，以產生兩個四位元二進位數字相加之結果。其輸入變數為四位元二進位被加數 ***A***、四位元二進位加數 ***B***、及進位輸入 ***cin***，輸出變數為四位元二進位和值 ***S*** 與進位輸出 ***cout***。
Design a 4-bit binary adder which may produce the arithmetic sum of two 4-bit binary numbers. The input variables are the 4-bit binary augend (被加數) ***A*** , the 4-bit binary addend (加數) ***B*****,** and the carry-in ***cin***, and the output variables produce the 4-bit sum ***S*** and the carry-out ***cout***.

i. 利用**(b)**中之全加器來建構此 4-bit Ripple Carry Adder (RCA)。請撰寫出此 RCA 之 HDL 電路模組，模組名稱與 port list 請訂為 Lab2_4_bit_RCA (input [3:0] A, B, input cin, output [3:0] S, output cout)，檔案命名為 Lab2_4_bit_RCA.v。
Please use the full adder designed in **(b)** to construct the 4-bit Ripple Carry Adder (RCA). Please write the HDL circuit module of the RCA. The circuit module and port list should be named as Lab2_4_bit_RCA (input [3:0] A, B, input cin, output [3:0] S, output cout), and its file should be named as Lab2_4_bit_RCA.v.

ii. 請撰寫此 RCA 之測試模組，至少以附圖中八組測資測試之。請將此測試模組命名為 t_Lab2_4_bit_add，檔案則命名為 t_Lab2_ 4_bit_add.v。
Please write the testbench of the RCA in which at least eight test data showed in the following figure should be included. The testbench module should be named as t_Lab2_4_bit_add, and its file should be named as t_Lab2_4_bit_add.v.

A	B	cin
0000	0000	1
1111	1111	1
0011	0110	1
0101	1001	0
0111	1010	1
1101	0010	0
1111	0111	0
1110	0101	1

(d) **四位元前看進位加法器(4-bit Carry Lookahead Adder, CLA)**：設計一個四位元前看進位加法器，以產生兩個四位元二進位數字相加之結果。其輸入變數為四位元二進

位被加數 ***A***、四位元二進位加數 ***B***、及進位輸入 ***cin***，輸出變數為四位元二進位和值 ***S*** 與進位輸出 ***cout***。

Design a 4-bit Carry Lookahead Adder, which may produce the arithmetic sum of two 4-bit binary numbers. The input variables are the 4-bit binary augend (被加數) ***A*** , the 4-bit binary addend (加數) ***B*, and the carry-in *cin***, and the output variables produce the 4-bit sum ***S*** and the carry-out ***cout***.

i. 請以 **gate-level modeling** 方式撰寫此 CLA 之 HDL 電路模組。AND 與 OR 邏輯閘之延遲設為 2 個單位時間，XOR 與 XNOR 邏輯閘之延遲設為 4 個單位時間，NOT 邏輯閘之延遲則忽略之。模組名稱與 port list 請訂為 Lab2_4_bit_CLA_gate (input [3:0] A, B, input cin, output [3:0] S, output cout)，檔案則命名為 Lab2_4_bit_CLA_gate.v。

   Please write the HDL circuit module of the CLA in **gate-level modeling**. Assume that the propagation delay of AND and OR gates is 2 time units, the propagation delay of XOR and XNOR gates is 4 time units, and the propagation delay of NOT gates is ignored. The circuit module and port list should be named as Lab2_4_bit_CLA_gate (input [3:0] A, B, input cin, output [3:0] S, output cout), and its file should be named as Lab2_4_bit_CLA_gate.v.

ii. 請以 **dataflow modeling** (**assign**)方式撰寫此 CLA 之 HDL 電路模組。無需考慮電路延遲。模組名稱與 port list 請訂為 Lab2_4_bit_CLA_df (input [3:0] A, B, input cin, output [3:0] S, output cout) ，檔案則命名為 Lab2_4_bit_CLA_df.v。

   Please write the HDL circuit module of the CLA in **dataflow modeling** (**assign**) without considering the propagation delay. The circuit module and port list should be named as Lab2_4_bit_CLA_df (input [3:0] A, B, input cin, output [3:0] S, output cout), and its file should be named as Lab2_4_bit_CLA_df.v.

iii. 請以 **behavioral modeling** (**always**)方式撰寫此 CLA 之 HDL 電路模組。無需考慮電路延遲。模組名稱與 port list 請訂為 Lab2_4_bit_CLA_beh (input [3:0] A, B, input cin, output [3:0] S, output cout)，檔案則命名為 Lab2_4_bit_CLA_beh.v。

   Please write the HDL circuit module of the CLA in **behavioral modeling** (**always**) without considering the propagation delay. The circuit module and port list should be named as Lab2_4_bit_CLA_beh (input [3:0] A, B, input cin, output [3:0] S, output cout), and its file should be named as Lab2_4_bit_CLA_beh.v.

iv. 請以 2A(c) ii 所撰寫之測試模組(t_Lab2_4_bit_add)測試此 CLA 之三個不同的電路模組。

   Simulate the three different circuit modules of this 4-bit CLA by the same testbench of 2A(c) ii (t_Lab2_4_bit_add).

B. 二元編碼十進位數字加減法器之設計 (Design of BCD adder-subtractor)

此部分之電路設計**無需考慮傳遞延遲**(propagation delay)。

No need to consider propagation delay for the circuit design of Part B.

(a) **二元編碼十進位數字之九補數轉換器(BCD 9’s Complementer)**：設計一個二元編碼十進位數字之九補數轉換器，有一個輸入變數及一個輸出變數。輸入變數為待轉換之二元編碼十進位數字 ***BCD***；輸出變數為其轉換後之九補數的二元編碼十進位數字 ***BCD_9c***。此轉換器之真值表如下所示：

Design a BCD 9’s complementer. This circuit has one input variable and one output variable. The input variable ***BCD*** is a binary-coded decimal digit, and the output variable ***BCD_9c*** is the corresponding 9's complement of the input digit. The truth table of the converter is given below:

BCD	BCD_9c
0 0 0 0	1 0 0 1
0 0 0 1	1 0 0 0
0 0 1 0	0 1 1 1
0 0 1 1	0 1 1 0
0 1 0 0	0 1 0 1
0 1 0 1	0 1 0 0
0 1 1 0	0 0 1 1
0 1 1 1	0 0 1 0
1 0 0 0	0 0 0 1
1 0 0 1	0 0 0 0
1 0 1 0～1 1 1 1	x x x x

i. 請以 **data flow modeling (assign)**方式撰寫其 Verilog 電路模組。利用 don't care conditions 化簡此電路。模組名稱與 port list 請訂為 Lab2_BCD_9c_df (input [3:0] BCD, output [3:0] BCD_9c)，檔案則命名為 Lab2_BCD_9c_df.v。
Please write the Verilog circuit module in **dataflow modeling**. Make use of the don't care conditions to simplify the circuit of this application. The circuit module and port list should be named as Lab2_BCD_9c_df (input [3:0] BCD, output [3:0] BCD_9c), and its file should be named as Lab2_BCD_9c_df.v.

ii. 請以 **behavioral modeling (always)**方式撰寫其 Verilog 電路模組，模組名稱與 port list 請訂為 Lab2_BCD_9c_beh (input [3:0] BCD, output [3:0] BCD_9c)，檔案則命名為 Lab2_BCD_9c_beh.v。
Please write the Verilog circuit module in **behavioral modeling**. The circuit module and port list should be named as Lab2_BCD_9c_beh (input [3:0] BCD, output [3:0] BCD_9c), and its file should be named as Lab2_BCD_9c_beh.v.

iii. 請撰寫一測試模組來**完整測試**此 BCD 九補數轉換器的兩個不同的電路模組，請將此測試模組命名為 t_Lab2_BCD_9c，檔案則命名為 t_Lab2_BCD_9c.v。
Please write a testbench to test the two different BCD 9's complementer circuit modules thoroughly. The testbench module should be named as t_Lab2_BCD_9c, and its file should be named as t_Lab2_BCD_9c.v.

(b) **一位數 BCD 加法器 (1-digit BCD Adder)**：設計一個一位數的 BCD 加法器，有三個輸入變數以及兩個輸出變數。輸入變數為一位數(四位元) 的 BCD 被加數 ***BCD_X***、一位數(四位元) 的 BCD 加數 ***BCD_Y***、及一位元之進位輸入 ***cin***；輸出變數為一位數(四位元) 的 BCD 和值 ***BCD_S*** 及一位元進位輸出 ***cout***。**(BCD 的加法相關設計可參考教科書§4.6 小節。)**
Design a 1-digit BCD adder. The circuit has three input variables and two output variables. The input variables are a 1-digit (4-bit) BCD augend ***BCD_X***, a 1-digit (4-bit) BCD addend ***BCD_Y*** and a carry-in bit ***cin***. The output variables are a 1-digit (4-bit) BCD sum ***BCD_S*** and a carry out bit ***cout***. **(Refer to Section 4.6 of textbook for the design of BCD addition.)**

i. 請以 **data flow modeling** (**assign**)方式撰寫其 Verilog 電路模組。模組名稱與 port list 請訂為 Lab2_BCD_1digit_add_df (input [3:0] BCD_X, BCD_Y, input cin, output [3:0] BCD_S, output cout)，檔案則命名為 Lab2_BCD_1digit_add_df.v。
Please write the Verilog circuit module in **data flow modeling** (**assign**). The circuit module and port list should be named as Lab2_BCD_1digit_add_df (input [3:0] BCD_X, BCD_Y, input cin, output [3:0] BCD_S, output cout), and its file should be named as Lab2_BCD_1digit_add_df.v.

ii. 請以 **behavioral modeling** (**always**)方式撰寫其 Verilog 電路模組，模組名稱與 port list 請訂為 Lab2_BCD_1digit_add_beh (input [3:0] BCD_X, BCD_Y, input cin, output [3:0] BCD_S, output cout)，檔案則命名為 Lab2_BCD_1digit_add_beh.v。
Please write the Verilog circuit module in **behavioral modeling**. The circuit module and port list should be named as Lab2_BCD_1digit_add_beh (input [3:0] BCD_X, BCD_Y, input cin, output [3:0] BCD_S, output cout), and its file should be named as Lab2_BCD_1digit_add_beh.v.

iii. 請撰寫一個測試模組來測試此一位數 BCD 加法器的兩個不同的電路模組並至少以附圖中八組測資測試。請將此測試模組命名為 t_Lab2_BCD_1digit_add，檔案則命名為 t_Lab2_BCD_1digit_add.v。Please write a testbench to test the two different 1-digit BCD Adder circuit modules which at least eight test data showed in the following figure should be included. The testbench module should be named as t_Lab2_BCD_1digit_add, and its file should be named as t_Lab2_BCD_1digit_add.v.

BCD_X	**BCD_Y**	cin
0000	0000	1
1001	1001	1
0011	0111	0
0101	1000	0
0011	0110	1
0110	0001	1
1001	0110	0
0101	0011	0

(c) **三位數 BCD 加法器 (3-digit BCD Adder)**：設計一個三位數的 BCD 加法器，有三個輸入變數以及兩個輸出變數。輸入變數為三位數(十二位元)的 BCD 被加數 ***BCD_3X***、三位數(十二位元)的 BCD 加數 ***BCD_3Y***、以及一位元之進位輸入 ***cin***；輸出變數為三位數(十二位元)的 BCD 和值 ***BCD_3S*** 及一位元進位輸出 ***cout***。變數 ***BCD_3X***、***BCD_3Y*** 及 ***BCD_3S*** 中[11:8]代表百位數，[7:4]代表十位數，[3:0]代表個位數。
Design a 3-digit BCD Adder. There are 3 input variables and two output variables. The input variables are a 3-digit (12-bit) BCD augend ***BCD_3X***, a 3-digit (12-bit) BCD addend ***BCD_3Y***, and a 1-bit carry input ***cin***. The output variables are a 3-digit (12-bit) BCD sum ***BCD_3S*** and a 1-bit carry out ***cout.*** For the 12-bit BCD variables ***BCD_3X***, ***BCD_3Y***, and ***BCD_3S***, [11: 8] represents the hundreds digit, [7:4] represents the tens digit, and [3:0] represents the unit digit.

i. 請撰寫其 Verilog 電路模組，利用(b) ii 設計之一位數 BCD 加法器建構之。模組名稱與 port list 請訂為 Lab2_BCD_3digit_add (input [11:0] BCD_X, BCD_Y, input cin, output [11:0] BCD_S, output cout)，檔案則命名為 Lab2_BCD_3digit_add.v。
Please write the Verilog circuit module. Use the 1-digit BCD Adder composed in (b) ii to build this 3-digit BCD Adder. The circuit module and port list should be named as Lab2_BCD_3digit_add (input [11:0] BCD_X, BCD_Y, input cin, output

[11:0] BCD_S, output cout), and its file should be named as Lab2_BCD_3digit_add.v.

ii. 請撰寫一個測試模組來測試此電路模組並至少以附圖中八組測資測試。請將此測試模組命名為 t_Lab2_BCD_3digit_add，檔案則命名為 t_Lab2_BCD_3digit_add.v。
Please write a testbench to test the circuit module, in which at least eight test data showed in the following figure should be included. The testbench module should be named as t_Lab2_BCD_3digit_add, and its file should be named as t_Lab2_BCD_3digit_add.v.

BCD_3X	***BCD_3Y***	***cin***
0000_0000_0000 (000)	0000_0000_0000 (000)	1
1001_1001_1001 (999)	1001_1001_1001 (999)	1
0110_1000_0010 (682)	1000_0011_0101 (835)	0
0100_0101_0001 (451)	0000_0110_1001 (069)	0
0011_1000_0111 (387)	0110_0001_0110 (616)	1
0111_0110_0101 (765)	1001_0100_0011 (943)	0
0101_1000_0101 (585)	0101_0101_0110 (556)	0
1001_0100_1000 (948)	0000_0101_0001 (051)	1

(d) **三位數 BCD 加減法器 (3-digit BCD Adder-Subtractor)**：設計一個三位數的 BCD 加減法器。作減法運算時，請轉為十補數的加法進行運算，亦即將被減數加上減數的十補數。此模組有三個輸入變數以及兩個輸出變數。輸入變數為兩個三位數(十二位元) BCD 運算元 ***BCD_3X***、***BCD_3Y***，以及表示進行加或減法的控制輸入 ***mode***。輸出變數為三位數(十二位元) BCD 運算結果 ***BCD_3R*** 與一位元輸出 ***kout***。變數 ***BCD_3X***、***BCD_3Y*** 及 ***BCD_3R*** 中，[11:8] 代表百位數、[7:4] 代表十位數、[3:0] 代表個位數。當 ***mode*** = 0 時，執行加法運算：輸入 ***BCD_3X*** 為被加數、***BCD_3Y*** 為加數；輸出 ***BCD_3R*** 為和值，***kout*** 為進位輸出。當輸入 ***mode*** = 1 時，則執行減法運算：輸入 ***BCD_3X*** 為被減數、***BCD_3Y*** 為減數；輸出 ***BCD_3R*** 為差值之絕對值大小，***kout*** 為 0 代表差值為正數，***kout*** 為 1 代表差值為負數。 **(BCD 的減法相關設計可參考教科書 § 1.5 小節中的 "Subtraction with Complements" 內容。)**
Design a 3-digit BCD adder-subtractor. Perform **subtraction** by ***10's complement addition***, i.e., add the minuend with the 10's complement of the subtrahend. This circuit has 3 input variables and 2 output variable. The input variables are two 3-digit (12-bit) BCD operands ***BCD_3X*** and ***BCD_3Y***, and a control signal ***mode*** indicating either addition or subtraction should be performed. The output variables are a 3-digit (12-bit) BCD result ***BCD_3R***, and a 1-bit output variable ***kout***. For the 12-bit BCD variables ***BCD_3X***, ***BCD_3Y***, and ***BCD_3R***, [11: 8] represents the hundreds digit, [7:4] represents the tens digit, and [3:0] represents the unit digit. When ***mode*** = 0, perform BCD addition: input ***BCD_3X*** is the augend and ***BCD_3Y*** is the addend; output ***BCD_3R*** is the sum and ***kout*** is the carry out. When ***mode*** = 1, perform BCD subtraction: input ***BCD_3X*** is the minuend and ***BCD_3Y*** is the subtrahend; output ***BCD_3R*** is the magnitude, i.e., the absolute value, of the difference, and ***kout*** is 0 if the difference is positive, otherwise, it is 1. **(Refer to the content of "Subtraction with Complements" in Section 1.5 of textbook for the design of BCD subtraction).**

i. 請撰寫其 Verilog 電路模組。利用(a) ii 設計之 BCD 九補數轉換器及(c)設計之三位數 BCD 加法器建構之。模組名稱與 port list 請訂為 Lab2_BCD_3digit_add_sub (input [11:0] BCD_X, BCD_Y, input mode, output

[11:0] BCD_R, output kout)，檔案則命名為 Lab2_BCD_3digit_add_sub.v。
Please write the Verilog circuit module. Use the BCD 9's Complementer composed in (a) ii and the 3-digit BCD Adder composed in (c) to build this 3-digit BCD Adder-Subtractor. The circuit module and port list should be named as Lab2_BCD_3digit_add_sub (input [11:0] BCD_X, BCD_Y, input mode, output [11:0] BCD_R, output kout), and its file should be named as Lab2_BCD_3digit_add_sub.v.

ii. 請撰寫一個測試模組來測試此電路模組，並至少以下圖中的八組測資測試。請將此測試模組命名為 t_Lab2_BCD_3digit_add_sub，檔案則命名為 t_Lab2_BCD_3digit_add_sub.v。

Please write a testbench to test the circuit module with at least eight test data showed in the following table. The testbench module should be named as t_Lab2_BCD_3digit_add_sub, and its file should be named as t_Lab2_BCD_3digit_add_sub.v.

BCD_3X	***BCD_3Y***	***mode***
0000_0000_0000 (000)	0000_0000_0000 (000)	0
1001_1001_1001 (999)	1001_1001_1001 (999)	0
0101_0100_1000 (548)	0100_0101_1001 (459)	0
1001_1001_1001 (999)	1001_1001_1001 (999)	1
0101_0110_1001 (569)	0101_0110_1000 (568)	1
0001_0000_1000 (108)	0000_0101_0001 (051)	1
0011_1000_0111 (387)	0110_0001_0110 (616)	1
0111_0110_0101 (765)	1001_0100_0011 (943)	1

C.注意事項：(Notes)

- 請用 **Icarus Verilog** 作為編譯器，以 **vvp** 執行，並以 **GTKWave** 觀察波形圖。
  Please compile your Verilog code by **Icarus Verilog**, execute the compiled code by **vvp**, and then observe the waveform by **GTKWave**.
- 請務必依照上述各項目之規定命名模組及檔案。
  Be sure to name the modules and files as described above.
- 禁止抄襲，違者(抄襲者與被抄襲者)以 0 分計算。
  Any assignment work by fraud will get a zero point.
- 助教會使用不同的測試模組來驗證同學的電路模組正確性。
  After you hand in your code, TA will use similar TestBench modules with different test data to verify the correctness of your design of the Verilog circuit modules.

3. 作業及 HDL 模組繳交(Hand in)

A. 作業報告繳交：pdf 檔，命名為 Lab2_學號

包含下列項目：配分含隱藏測資之測試及 Demo 狀況

(1) 畫出半加器之邏輯電路圖，並附上 2A(a)ii (半加器) 之模擬結果波形圖，並說明波形圖是否正確及所需之延遲時間。(5%)

(2) 畫出以半加器建構全加器之電路方塊圖，附上 2A(b)ii (全加器) 之模擬結果波形圖，並說明波形圖是否正確及所需之延遲時間。(5%)

(3) 畫出以全加器建構四位元漣波進位加法器之電路方塊圖，附上 2A(c)ii (4-bit RCA) 之模擬結果波形圖，並說明波形圖是否正確及所需之延遲時間。(5%)

(4) 列出四位元前看進位加法器之相關布林代數式(如：Pi、Gi、Ci、Si 等)，附上 2A(d)iv (4-bit CLA) 之模擬結果波形圖，說明三個不同電路模組之波型圖是否

正確，以及 gate-level modeling 電路模組所需的延遲時間。(20%)

(5) 詳述 BCD 九補數轉換器(BCD 9's Complementer)之電路設計，推導出各輸出變數的最簡 sum-of-products 布林代數式，並附上 2B(a)iii 之模擬結果，並說明是否正確。(15%)

(6) 詳述一位數 BCD 加法器之電路設計，並劃出電路方塊圖。附上 2B(b)iii 之模擬結果，並說明是否正確。(15%)

(7) 詳述三位數 BCD 加法器之電路設計，並劃出電路方塊圖。附上 2B(c)ii 之模擬結果，並說明是否正確。(5%)

(8) 詳述三位數 BCD 加減法器之工作程序及電路設計，並劃出電路方塊圖。附上 2B(d)ii 之模擬結果，並說明是否正確。(20%)

(9) 心得與感想、及遭遇到的問題或困難 (10%)

Hand in a pdf file, named Lab2_StudentID , including the following items: (Scores include answering problems in demo and the testing of extra data.)

(1) Draw the logic diagram of the half adder, attach the waveforms of the simulation results for this module tested in 2A(a)ii, describe the propagation delay of the circuit, and determine whether the waveforms are correct or not. (5%)

(2) Draw the block diagram of the full adder by using half adder as basic blocks, attach the waveform of the simulation results tested in 2A(b)ii, describe the propagation delay of the circuit, and explain whether the waveforms are correct or not. (5%)

(3) Draw the block diagram of the 4-bit ripple-carry adder (RCA) by using full adders as basic blocks, attach the waveform of the simulation results tested in 2A(c)ii, describe the propagation delay of the circuit, and explain whether the waveforms are correct or not. (5%)

(4) Derive the Boolean expressions of P_i, G_i, C_i, and S_i for the 4-bit carry lookahead adder (CLA). Attach the waveforms of the simulation results for the three different modules of the 4-bit CLA tested in 2A(d)iv, describe the propagation delay of the module written in gate-level modeling, and explain whether the waveforms of the three modules are correct or not. (20%)

(5) Describe the design process of the BCD 9's Complementer, including the Karnaugh maps and the derived simplified Boolean expressions in sum-of-products form. Attach the waveforms of the simulation results for the two different modules tested in 2B(a)iii, and explain whether the waveforms are correct or not. (15%)

(6) Describe the design process of the 1-digit BCD Adder and draw the block diagram of the circuit. Attach the waveforms of the simulation results for the two different modules tested in 2B(b)iii, and explain whether the waveforms are correct or not. (15%)

(7) Describe the design process of the 3-digit BCD Adder and draw the block diagram of the circuit. Attach the waveforms of the simulation results for the module tested in 2B(c)ii, and explain whether the waveforms are correct or not. (5%)

(8) Describe the design process of the 3-digit BCD Adder-Subtractor and draw the block diagram of the circuit. Attach the waveforms of the simulation results for the module tested in 2B(d)ii, and explain whether the waveforms are correct or not. (20%)

(9) Describe what you have learned from this lab unit and discuss the problems or difficulties encountered in this lab. (10%)

B. Verilog modules 檔案繳交：19 files

Hand in the following Verilog modules: 19 files

Lab2_half_add.v、t_Lab2_half_add.v、
Lab2_full_add.v、t_Lab2_full_add.v、
Lab2_4_bit_RCA.v、
Lab2_4_bit_CLA_gate.v、Lab2_4_bit_CLA_df.v、Lab2_4_bit_CLA_beh.v、

t_Lab2_4_bit_add.v、
Lab2_BCD_9c_df.v、Lab2_BCD_9c_beh.v、t_Lab2_BCD_9c.v、
Lab2_BCD_1digit_add_df.v、Lab2_BCD_1digit_add_beh.v、
t_Lab2_BCD_1digit_add.v、
Lab2_BCD_3digit_add.v、t_Lab2_BCD_3digit_add.v、
Lab2_BCD_3digit_add_sub.v、t_Lab2_BCD_3digit_add_sub.v

4. DEADLINE

- 本實驗單元為一人一組，作業請上傳至 E3 平台。
  This lab unit is one student per group. Please upload your Lab Report (pdf file) and the corresponding HDL code (.v files) onto e-Campus platform.

- 請將上述作業報告及 Verilog 電路模組與測試模組檔案(.v)全部壓縮成一個 **zip 檔**(禁止上傳其他檔案格式)，並以「**Lab2_學號**」的方式命名，如：「Lab2_111550001」。**壓縮檔案時，請選取好要求繳交的檔案執行壓縮，不要對整個資料夾壓縮。**
  Please compress the pdf file of lab report and the Verilog circuit modules and testbench described above all into one **zip** file (other format is not accepted), and name the zip file as “Lab2_StudentID” , for example, “Lab2_111550001”. **When compressing files, please select the requested files for compression, and do not compress the entire directory.**

- 作業繳交截止日期為 **2023/5/8 (一) 23:59**。**不接受逾期繳交**。
  The deadline for handing in lab report and Verilog files is **May 8, 2023 (Monday) 23:59.** **No late hand-in is allowed.**

- **Demo** 時間暫定為 **2023/5/10 (三) 1:00pm~9:00pm**、**2023/5/11 (四) 6:00pm~9:00pm**，之後會再發公告通知大家上網填寫 Demo 時間表。未繳交作業者，將無法參加 Demo；雖有繳交作業但未 Demo 者，亦不予計分。
  The **demo time** is arranged at **May 10 (Wed.) 1:00PM~9:00PM and May 11 (Thur.) 6:00PM~9:00PM** tentatively, and we will send an announcement to inform you to fill the Demo schedule online. Those who have not hand-in their lab reports and Verilog files will not be able to demo their work, and those who have not demo their lab units will get zero score.

- **程式碼請勿抄襲別人或讓別人抄襲，經查證後此次 lab 總分一律以 0 分計算。**
  Any assignment work by fraud will get a zero point