Faculty of Engineering, Environment and Computing

201CDE Module Title

Assignment Brief

Module Title **Analogue and Digital Electronics2**	Individual	Cohort (Sept)	Module Code **201CDE**
Coursework Title (e.g. CWK1) **Digital Coursework**			Hand out date: **16th September 2019**
Lecturer **Dr. Arfan Ghani**			Due date and time: **25th October 2019** Moodle: 18:00:00
Estimated Time (hrs): **15** Word Limit*:	Coursework type: **Digital Design and Implementation**		% of Module Mark/ **25%**

Submission arrangement online via CUMoodle:
File types and method of recording: **Submit through Moodle as a pdf document naming the file name with your student identification number as the file name** Mark and Feedback date:
Mark and Feedback method:

Module Learning Outcomes Assessed:

1. Analyse and design synchronous sequential systems.
2. Model digital systems for simulation and synthesis using an HDL.

Task and Mark distribution:

Marks breakdown	**Max**	**Awarded**
Sequential design and simulation	25	
VHDL model and simulation	25	
Assessor's signature	Total 50	Total

This coursework involves the design and simulation of a synchronous sequential counter using a traditional approach employing small scale logic components and a modern approach using behavioural modelling with VHDL. The design specification for each student is unique and based on their student ID number.

Specification

Write down the seven digits of your student identification number below. An example is shown so that you know what to expect.

Example

1	4	2	7	5	1	4

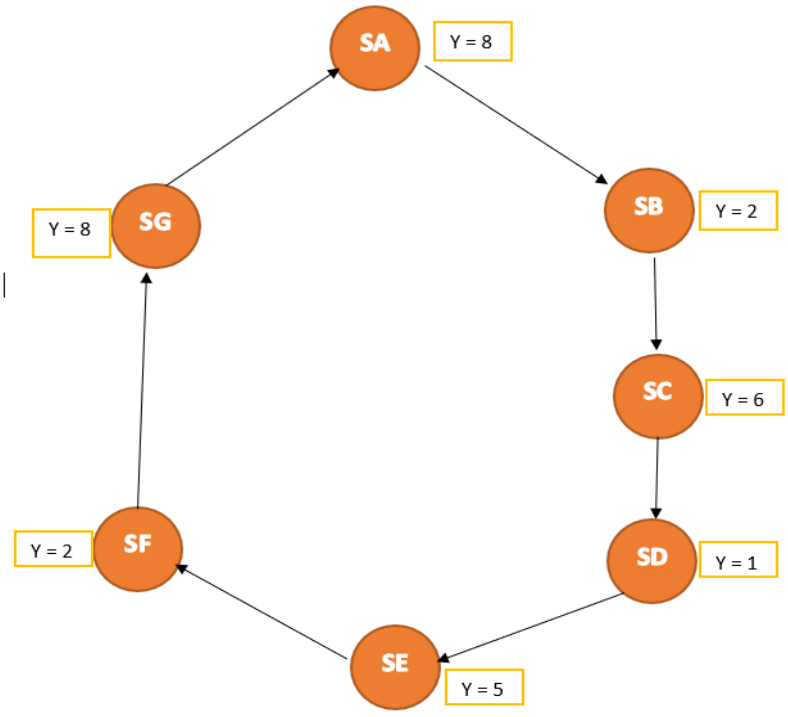
Actual SID

8	2	5	1	6	2	8

The digits in your ID number, in **the right to left order** are used to specify the sequential behaviour that a synchronous counter will execute in response to applied clock pulses. Assume that the ID digits are represented in a conventional **4-bit BCD code**. The behaviour required of the counter is that it produces a 4bit digital output that represents the **BCD** value of the successive digits of your ID number in the right to left sequence above. Once the left-hand digit is output the counter will go back to the first and the sequence repeats. Note the digits represent the outputs of the counter and not its state variables.

1. Traditional design

Draw a state diagram to represent the behaviour of the counter you are required to design. It is suggested you employ the letters of the alphabet to refer to each state and keep the outputs as decimal numbers at this stage.



STATE VALUE ASSIGNMENT
Let:

SA = 1 0 0 0

SB = 0 0 1 0

SC = 0 1 1 0

SD = 0 0 0 1

SE = 0 1 0 1

SF = 0 0 1 0

SG = 1 0 0 0

The state diagram above represents a simplified way of representing the state machine

Convert the state diagram into its associated state table.

➢

STATE TABLE

Present State	Next State	Output (Y)
SA	SB	**8**
SB	SC	**2**
SC	SD	**6**
SD	SE	**1**
SE	SF	**5**
SF	SG	**2**
SG	SA	**8**

Assume that a simple binary state variable allocation can be made and that any unused states can be treated as don't cares. Draw the corresponding transition table for the counter including flip-**flop (DFF)** inputs and BCD outputs for your designated storage device.

➢

TRANSITION TABLE

Ps2	Ps1	Ps0	Ns2	Ns1	Ns0	Y3	Y2	Y1	Y0
0	0	0	0	1	1	1	0	0	0
0	0	0	1	0	0	0	0	1	0
0	1	0	1	1	0	0	1	1	0
0	1	1	0	0	0	0	0	0	1
1	0	1	0	1	0	0	1	0	1
1	0	1	1	0	0	0	0	1	0
1	1	0	0	0	1	1	0	0	0
X	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**

*In the table above '**X**' represents a Don't-Care*

Using excitation maps determine the next state logic functions that can be used to drive your flipflop inputs.

➢

Next State Logic Functions

Groups:

Ns2 =>

Ps2Ps1 / Ps0	00	01	11	10
0	0	0	0	**1**
1	0	**1**	X	**1**

i

ii

Y = i + ii

Y = $\overline{Ps2}$Ps1Ps0 + Ps2$\overline{Ps1}$

Ns1 =>

Ps2Ps1 / Ps0	00	01	11	10
0	0	1	0	0
1	1	0	X	1

i

ii

Y = i + ii

Y = $\overline{Ps2}$Ps1$\overline{Ps0}$ + $\overline{Ps1}$ Ps0

Ns0 =>

Ps2Ps1 / Ps0	00	01	11	10
0	1	1	0	1
1	0	0	X	0

i

ii

Y = i + ii

Y = $\overline{Ps1}$ $\overline{Ps0}$ + $\overline{Ps2}$ $\overline{Ps0}$

Also draw Karnaugh maps to deduce minimal logic functions for the 4 output functions that are required to give the BCD output codes. Note that depending on your ID number not all outputs will be significant in all cases.

Output Logic Functions

Groups:

Y3 =>

Ps2Ps1 / Ps0	00	01	11	10
0	1	0	1	0
1	0	0	X	0

i

ii

Y = i + ii

Y =Ps2Ps1 $\overline{Ps0}$ + $\overline{Ps2}$ $\overline{Ps1}$ $\overline{Ps0}$

Y2 =>

Ps2Ps1 / Ps0	00	01	11	10
0	0	**1**	0	**1**
1	0	0	X	0

i

ii

Y = i + ii

Y = $\overline{Ps2}$Ps1$\overline{Ps0}$ + Ps2 $\overline{Ps1}$ $\overline{Ps0}$

Y1 =>

Ps2Ps1 / Ps0	00	01	11	10
0	**0**	**1**	0	0
1	**1**	0	X	**1**

ii

i

Y = i + ii

Y = $\overline{Ps1}$ $Ps0$ + $\overline{Ps2}$Ps1$\overline{Ps0}$

Y0 =>

Ps2Ps1 / Ps0	00	01	11	10
0	**0**	0	0	**1**
1	**0**	**1**	X	0

i

ii

Y = i + ii

Y = $Ps2$ $\overline{Ps1}$ $\overline{Ps0}$ + $\overline{Ps2}$Ps1$Ps0$

- Enter the associated schematic circuit diagram here along with commentary. The simulation results clearly show the implementation is fully compliant with the specification. (6 marks)

Schematic Circuit:

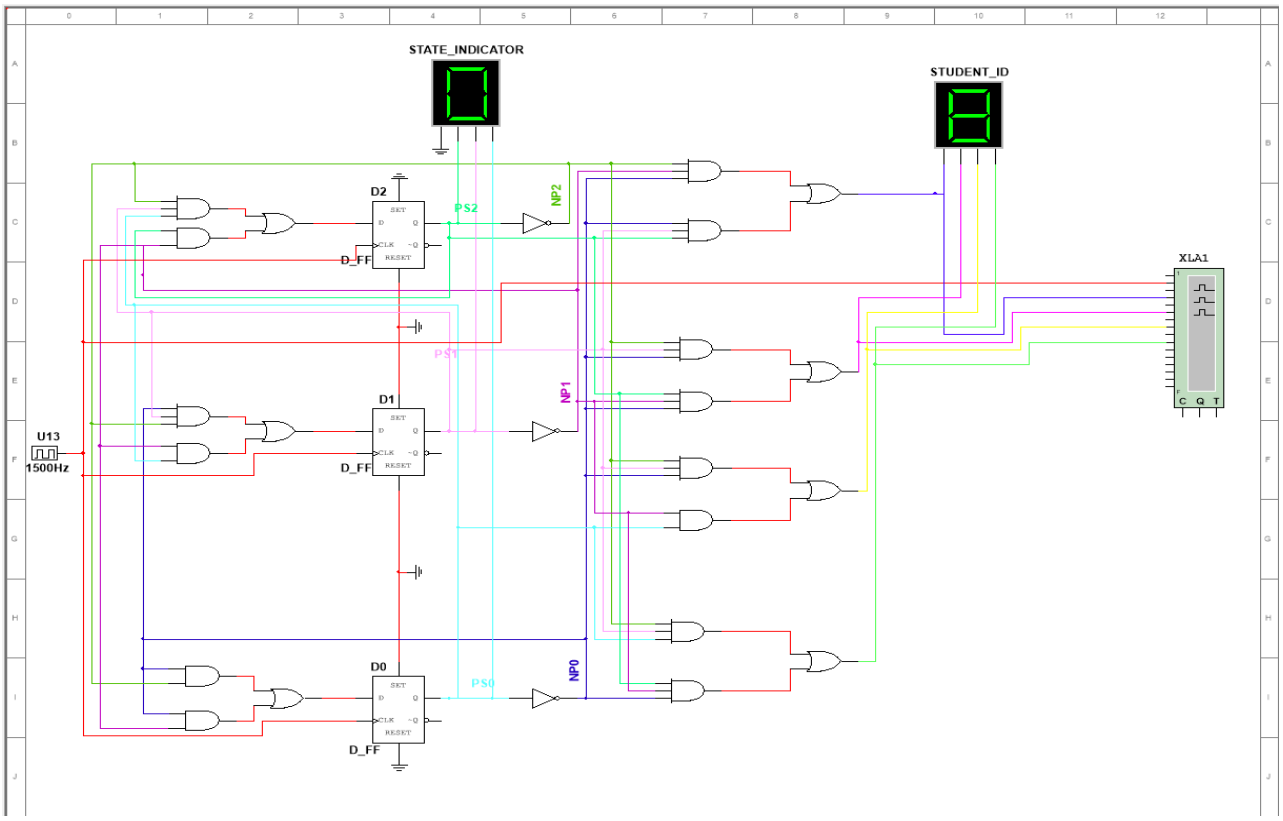
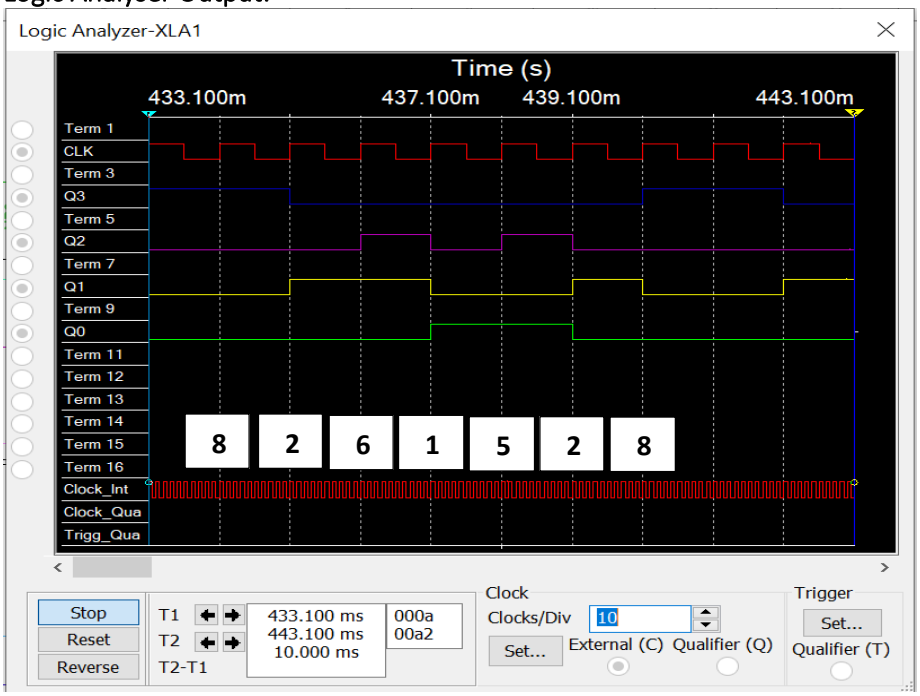


Fig 1.0 *Schematic Circuit of BCD Counter*

Logic Analyser Output:



The first waveform depicted in **red** is the Clock input

The **blue** waveform is the output for Y3

The **purple** waveform is the output for Y2

The **yellow** waveform is the output for Y1

The **green** waveform is the output for Y0

Fig 2.0 *Logic Analyzer Output of BCD Counter*

The schematic diagram shown in **fig 1.0** depicts a synchronous sequential counter which outputs my student ID number in reverse order **(8261528**). The logic behind the three D Flipflops depicts the next state logic.
The next state logic is fed into the **Data** port which helps to drive the inputs of the D Flip-flop as seen in the figure **3.0** below:

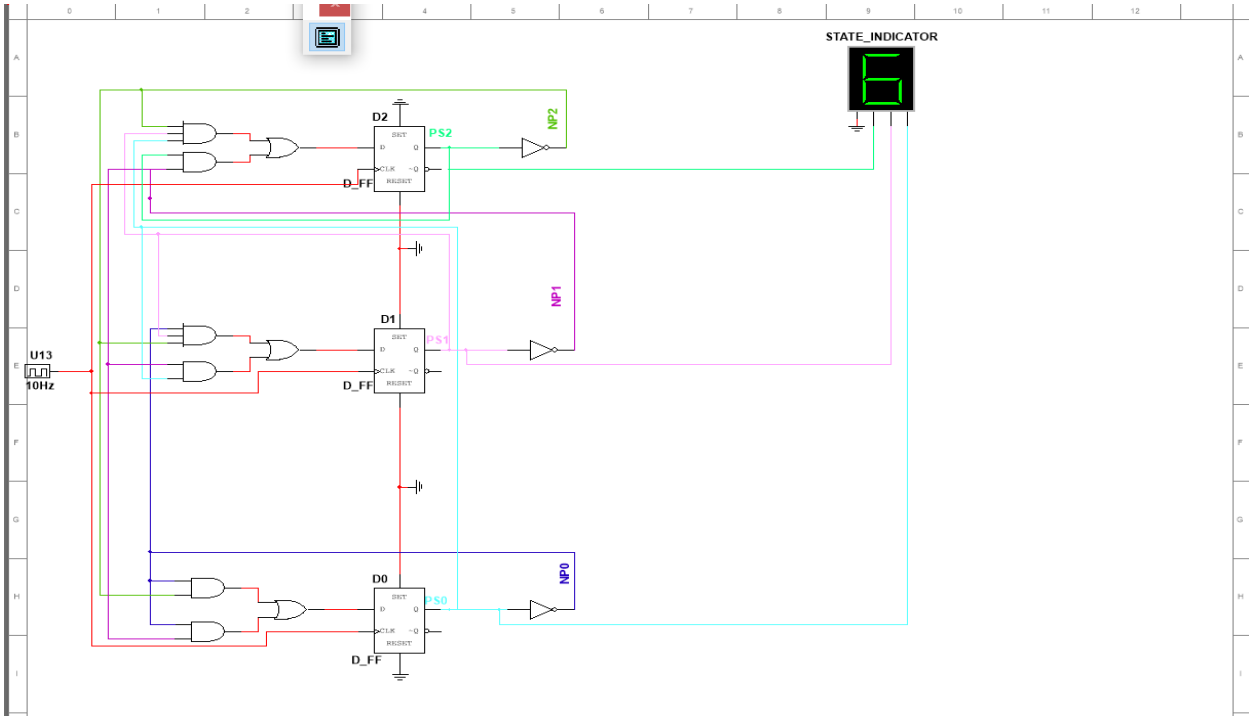


Fig 3.0 *Schematic Diagram of Next State Functions*

The minimal functions were gotten through the help of Karnaugh maps. The next state logic gives a counter input which counts from (0-6) as seen in the video implementation.

Link to video simulation of Circuit: https://youtu.be/2Iq64RGAdck

This is then fed into the D Flip Flops which is then combined with a combinational logic as seen in the schematic diagram in **Fig 1.0** to give out the required outputs.

The waveforms shown in **Fig 2.0** displays the logic analyser outputs of the combinational logic, which is Q3, Q2, Q1, Q0. At each rising edge of the clock, the current state outputs the corresponding the output. For instance, at the first rising edge, Q3 is “1”, Q2 is “0”, Q1 is “0” and Q0 is “0” which gives the decimal output of **8,** at each rising edge a corresponding output is depicted.

2. VHDL Design

For exactly the same counter specification in **Part 1**, write a VHDL entity and architecture that will model the required sequential behaviour. The VHDL code should be entered in a VHDL tool such as Xilinx Vivado to confirm that it compiles without errors.

➢ Insert your VHDL model design with proof of compilation here. (9 marks)

VHDL Model Design:

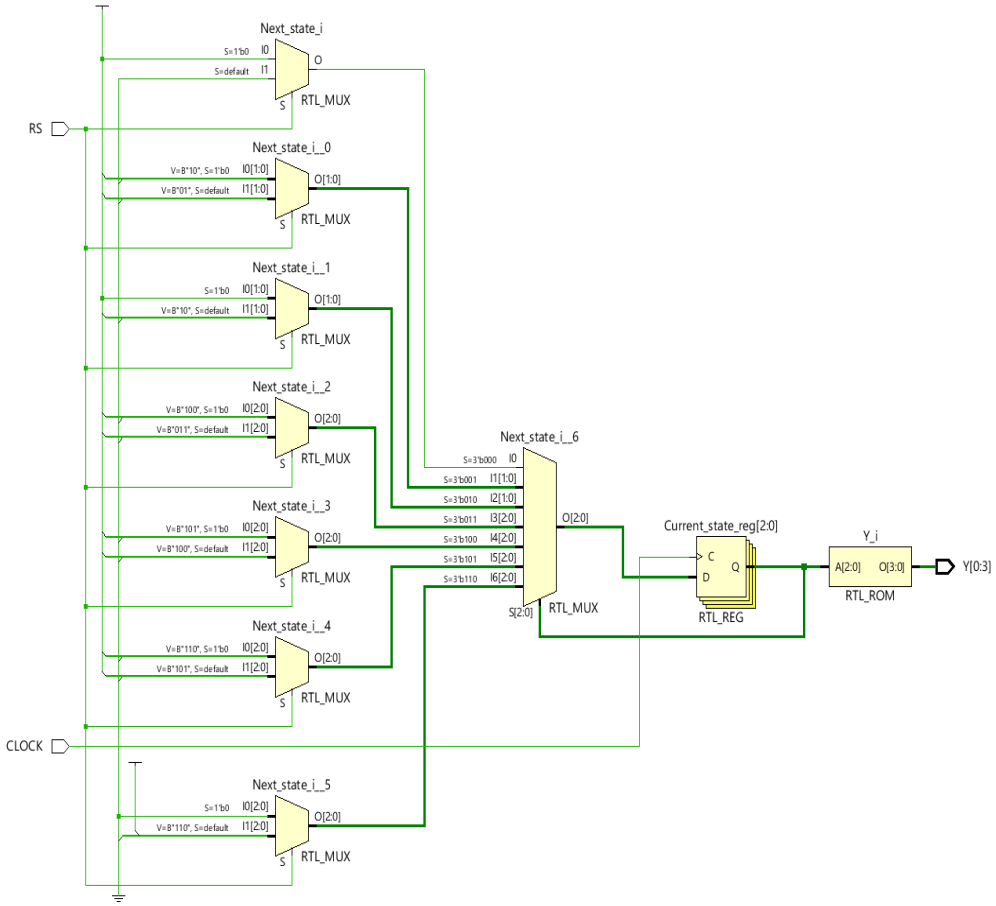


Fig 4.0 *VHDL Model Design*

Proof of Compilation:

VHDL Entity and Architecture Code:

```
1  ----------------------------------------------------------------------------------...
20 library IEEE;
21 use IEEE.STD_LOGIC_1164.ALL;
22
23 entity Student_ID is --Student_ID entity declaration
24   Port ( RS,CLOCK : IN std_logic; -- Initalising Input ports
25    Y : OUT std_logic_vector (0 to 3)); -- Initalising Output Ports
26 end Student_ID; --End of Student_ID entity declaration
27
28 architecture Behavioral of Student_ID is --Architecture of Student ID which outputs my student ID
29 type STATE_TYPE is (SA,SB,SC,SD,SE,SF,SG); -- declaring the 7 states and state type
30 signal Current_state, Next_state: STATE_TYPE; --declaring the signals
31
32 begin
33 COMBIN : process (Current_State, RS) --In this block, we are sensitive to Current_state and RS
34 begin
35 case Current_state is
36
37 when
38  SA =>
39 Y <= "1000";
40 if RS = '0' then
41 Next_state <= SB;
42 else
43 Next_state <= SA;
44 end if;
45
46 when
47  SB =>
48 Y <= "0010";
49 if RS = '0' then
50 Next_state <= SC;
51 else
52 Next_state <= SB;
53 end if;
54
```

```
55 when
56  SC =>
57 Y <= "0110";
58 if RS = '0' then
59 Next_state <= SD;
60 else
61 Next_state <= SC;
62 end if;
63
64 when
65  SD =>
66 Y <= "0001";
67 if RS = '0' then
68 Next_state <= SE;
69 else
70 Next_state <= SD;
71 end if;
72
73 when
74 SE =>
75 Y <= "0101";
76 if RS = '0' then
77 Next_state <= SF;
78 else
79 Next_state <= SE;
80 end if;
81
82 when
83  SF =>
84 Y <= "0010";
85 if RS = '0' then
86 Next_state <= SG;
87 else
88 Next_state <= SF;
89 end if;
90
```

C:/Users/44784/Documents/Analogue and Digital/Vhdl_coursework33/Vhdl_coursework33.srcs/sources_1/new/Vhdl_entity.vhd

```
77  Next_state <= SF;
78  else
79  Next_state <= SE;
80  end if;
81
82  when
83   SF =>
84  Y <= "0010";
85  if RS = '0' then
86  Next_state <= SG;
87  else
88  Next_state <= SF;
89  end if;
90
91  when
92   SG =>
93  Y <= "1000";
94  if RS = '0' then
95  Next_state <= SA;
96  else
97  Next_state <= SG;
98  end if;
99  end case;
100 end process COMBIN;
101
102 -- Process to hold synchronous elements (flip-flops)
103 SYNCH: process
104 begin
105 wait until CLOCK'event and CLOCK =
106 '1';
107 CURRENT_STATE <= NEXT_STATE;
108 end process SYNCH;
109
110 end Behavioral; --End Architecture of Student ID which outputs my student ID
111
```

Proof of VHDL Synthesis and compilation:
https://youtu.be/_YtBA05BbN4

Also write a test bench that will enable the correct function of the counter model to be simulated. Again include proof that it compiles without errors on your design tool.

- Insert your VHDL test bench here. (8 marks)

Vhdl_tb2.vhd

C:/Users/44784/Documents/Analogue and Digital/Vhdl_coursework33/Vhdl_coursework33.srcs/sim_1/new/Vhdl_tb2.vhd

```
21  use IEEE.STD_LOGIC_1164.ALL;
22
23  entity Student_IDtb2 is --Entity is Empty
24  --  Port ( );
25  end Student_IDtb2;
26
27  architecture Behavioral of Student_IDtb2 is --Architecture declaration for my student_ID testbench
28
29  component Student_ID Port( -- Defining the input and output ports  of student id component
30  Clock : IN std_logic;
31  RS : IN std_logic;
32  Y : OUT std_logic_vector);
33  end component; --Ending the component declaration
34
35  signal RS:  std_logic := '0';  --Assigning the reset signal (RS) a fixed value
36  signal Clock:  std_logic := '0';
37  signal Y:  std_logic_vector (3 downto 0);
38
39  begin
40
41  uut : Student_ID PORT MAP ( --Ports defined in the entity are being instanced
42  Clock => Clock,
43  RS => RS,
44  Y => Y);
45
46
47   stim_proc: process
48   begin
49  Clock <= '0'; wait for 10ns; -- Clock is assigned 0 for the first 10ns
50  Clock <= '1'; wait for 10ns; --Clock is assigned 1 after the next 10ns after waiting
51  end process;
52  end Behavioral;
53
54
55
56
```

➢ Finally include output from the simulator that confirms your design simulates correctly. Briefly explain why you consider the result gives this confirmation. Insert simulation results and commentary here. (8 marks)

Simulator Output:

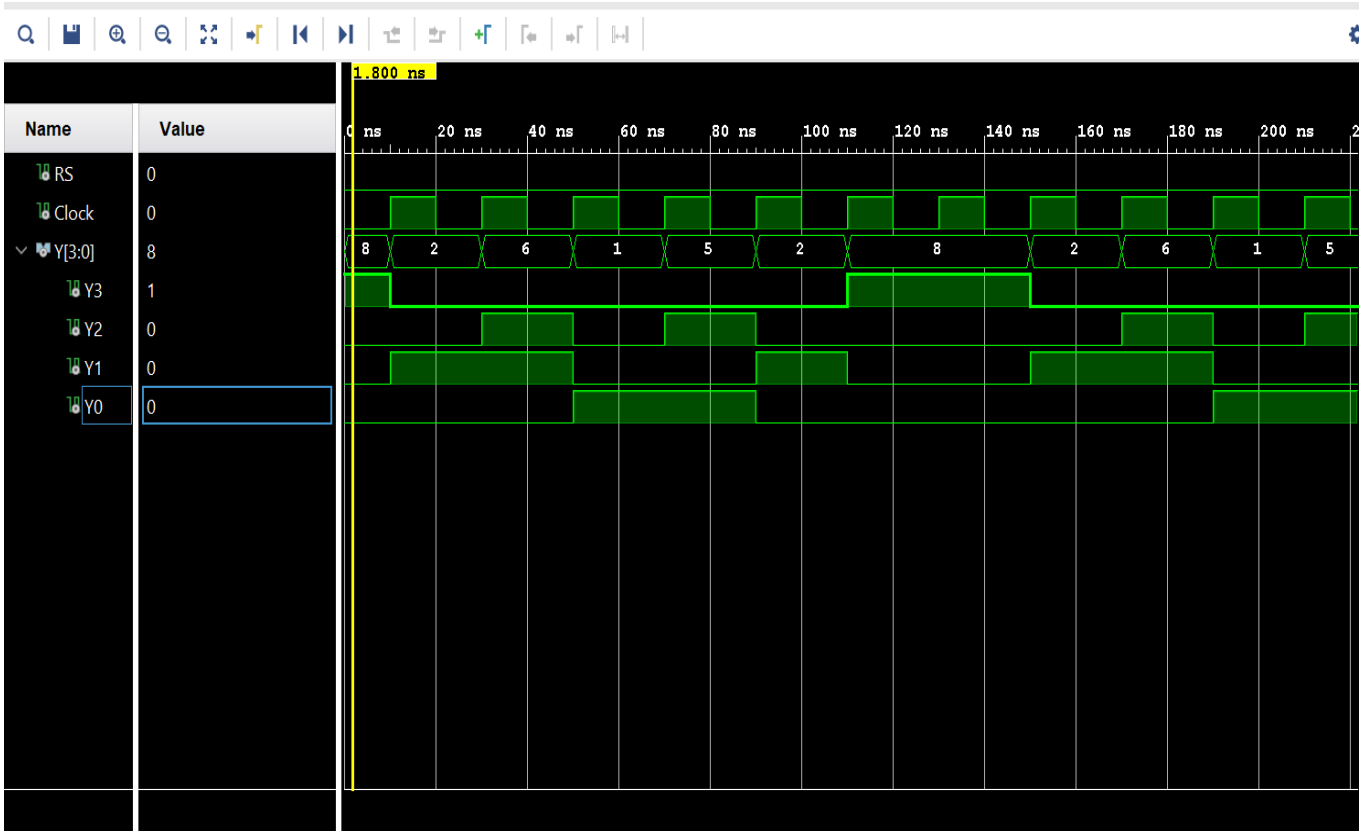


Fig 5.0: *Simulation Waveform*

Commentary:

The waveform depicted above in figure shows the output of the counter which is sensitive to the Reset (RS) and Clock signal, The Reset Signal (RS) is kept at 0, the clock alternates from 0 and 1 after 10 nanoseconds. At the first rising edge, the output is clocked, and the output is displayed (1 0 0 0) on each rising edge, the present state is assigned to the next state. Y3 is the most significant bit holding a BCD weight of 8, while Y2 holds a BCD weight of 4, Y1 holds a BCD weight of 2 and Y0 is the least significant bit holding a BCD weight of 1.

The video link below shows the full process in which the inputs are clocked in order to give the full output:

Link to Video simulation: https://youtu.be/4_Z3r6iCxck

The simulator outputs shows the expected output, as each rising edge the states change. The waveform depicted is similar to the output gotten from the multisim simulation.

Notes:

1. You are expected to use the Coventry University Harvard Referencing Style. For support and advice on this students can contact Centre for Academic Writing (CAW).
2. Please notify your registry course support team and module leader for disability support.
3. Any student requiring an extension or deferral should follow the university process as outlined here.
4. The University cannot take responsibility for any coursework lost or corrupted on disks, laptops or personal computer. Students should therefore regularly back-up any work and are advised to save it on the University system.
5. If there are technical or performance issues that prevent students submitting coursework through the online coursework submission system on the day of a coursework deadline, an appropriate extension to the coursework submission deadline will be agreed. This extension will normally be 24 hours or the next working day if the deadline falls on a Friday or over the weekend period. This will be communicated via your Module Leader.
6. You are encouraged to check the originality of your work by using the draft Turnitin links on your Moodle Web.
7. Collusion between students (where sections of your work are similar to the work submitted by other students in this or previous module cohorts) is taken extremely seriously and will be reported to the academic conduct panel. This applies to both courseworks and exam answers.
8. A marked difference between your writing style, knowledge and skill level demonstrated in class discussion, any test conditions and that demonstrated in a coursework assignment may result in you having to undertake a Viva Voce in order to prove the coursework assignment is entirely your own work.

9. If you make use of the services of a proof reader in your work you must keep your original version and make it available as a demonstration of your written efforts.
10. You must not submit work for assessment that you have already submitted (partially or in full), either for your current course or for another qualification of this university, unless this is specifically provided for in your assignment brief or specific course or module information. Where earlier work by you is citable, ie. it has already been published/submitted, you must reference it clearly. Identical pieces of work submitted concurrently will also be considered to be selfplagiarism.

Mark allocation guidelines to students (to be edited by staff per assessment)

0-39	40-49	50-59	60-69	70+	80+
Work mainly incomplete and /or weaknesses in most areas	Most elements completed; weaknesses outweigh strengths	Most elements are strong, minor weaknesses	Strengths in all elements	Most work exceeds the standard expected	All work substantially exceeds the standard expected

 Marking Rubric

GRADE	ANSWER RELEVANCE		ARGUMENT & COHERENCE		EVIDENCE		SUMMARY	
First **≥70**	Innovative response, answers the question fully, addressing the learning objectives of the assessment task. Evidence of critical analysis, synthesis and evaluation.		A clear, consistent in-depth critical and evaluative argument, displaying the ability to develop original ideas from a range of sources. Engagement with theoretical and conceptual analysis.		Wide range of appropriately supporting evidence provided, going beyond the recommended texts. Correctly referenced.		An outstanding, well-structured and appropriately referenced answer, demonstrating a high degree of understanding and critical analytic skills.	
Upper Second **60-69**	A very good attempt to address the objectives of the assessment task with an emphasis on those elements requiring critical review.		A generally clear line of critical and evaluative argument is presented. Relationships between statements and sections are easy to follow, and there is a sound, coherent structure.		A very good range of relevant sources is used in a largely consistent way as supporting evidence. There is use of some sources beyond recommended texts. Correctly referenced in the main.		The answer demonstrates a very good understanding of theories, concepts and issues, with evidence of reading beyond the recommended minimum. Well organised and clearly written.	
Lower Second **50-59**	Competently addresses objectives, but may contain errors or omissions and critical discussion of issues may be superficial or limited in places.		Some critical discussion, but the argument is not always convincing, and the work is descriptive in places, with over-reliance on the work of others.		A range of relevant sources is used, but the critical evaluation aspect is not fully presented. There is limited use of sources beyond the standard recommended materials. Referencing is not always correctly presented.		The answer demonstrates a good understanding of some relevant theories, concepts and issues, but there are some errors and irrelevant material included. The structure lacks clarity.	
Third **40-49**	Addresses most objectives of the assessment task, with some notable omissions. The structure is unclear in parts, and there is limited analysis.		The work is descriptive with minimal critical discussion and limited theoretical engagement.		A limited range of relevant sources used without appropriate presentation as supporting or conflicting evidence coupled with very limited critical analysis. Referencing has some errors.		Some understanding is demonstrated but is incomplete, and there is evidence of limited research on the topic. Poor structure and presentation, with few and/or poorly presented references.	
Fail **<40**	Some deviation from the objectives of the assessment task. May not consistently address the assignment brief. At the lower end fails to answer the question set or address the learning outcomes. There is minimal evidence of analysis or evaluation.		Descriptive with no evidence of theoretical engagement, critical discussion or theoretical engagement. At the lower end displays a minimal level of understanding.		Very limited use and application of relevant sources as supporting evidence. At the lower end demonstrates a lack of real understanding. Poor presentation of references.		Whilst some relevant material is present, the level of understanding is poor with limited evidence of wider reading. Poor structure and poor presentation, including referencing. At the lower end there is evidence of a lack of comprehension, resulting in an assignment that is well below the required standard.	
Late submission	0		0		0		0	