DigitLabwork#1

Decoder-Encoder-Comparator in Verilog codes

[1] Training goals

[2] Preparations

[3] Lab-task instruction

[4] Observations

= =

[1] Training goals

Through the lab-work, class members are expected to get acquainted with the following matters.

a)) the functions of decoders, encoders and comparators;

b)) the use of Verilog for logic circuit description and simulation, including

** circuit module building up in Verilog,

** setting up of the test data;

c)) the interactive commands required in operating the Verilog-code development system:

** compilation for a syntax error-free Verilog description and test data set;

** simulation of the Verilog-coded circuit module;

[** synthesis of the Verilog-coded circuit module].

[2] Preparation

Every class member should get oneself prepared by studying following materials prior to attending the lab-work sessions.

a)) Refer to "Digital Circuits 123"-section[1] and, better yet, datasheets from IC manufacturers such that functional/electrical/timing specifications regarding the circuit components targeted in the lab-work are understood;

b)) Refer to "Digital Circuits 123"-section [4], where examples of Verilog-coding of the circuit components targeted in the lab-work are offered;

c)) Refer to "Digital Circuits 123"-section [4] so as to know about how test data for circuit simulation should be arranged;

d)) Refer to "Development Context of Verilog Code" for operational details in running the developing system.

[3] Lab-task instruction

In digitlabwork#1, 3 tasks are assigned to every class members as given below.

[TASK1]

(1) Write a Verilog codes for a 2to4 decoder in compliance with the requirements given below.

(2) Put the codes under simulation; observe and interpret the waveforms of output signals

a)) coding specifications:

** write 3 modules of 2to4 decoder in gate-level, dataflow-level and behavior-level descriptions, respectively.

** the 3 modules will share the same 2-bit input, and with respective 4-bit outputs.

** write Verilog codes for generating test data sequence as requested.

b)) circuit specifications:

** I/O layout: 2 inputs D_1D_0,

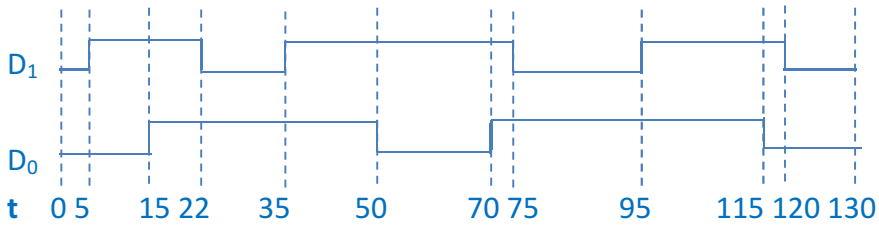
4 outputs $Y_3Y_2Y_1Y_0$;

** Functional:

D_1	D_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

** Timing: no delay(s) to be considered in this task

c)) testing data:



the testing stops at time-unit 130

[TASK2]

(1) Write a Verilog codes for a 2to4 decoder in compliance with the requirements given below.

(2) Put the codes under simulation; observe and interpret the waveforms of output signals

a)) coding specifications:

** write 3 modules of 4-bit comparator in gate-level, dataflow-level and behavior-level descriptions, respectively.

** the 3 modules will share the same 2-bit input, and with respective 3 1-bit output of comparison.

** write Verilog codes for generating test data sequence as requested.

b)) circuit specifications:

** I/O layout: 2 inputs $A_3A_2A_1A_0$, $B_3B_2B_1B_0$

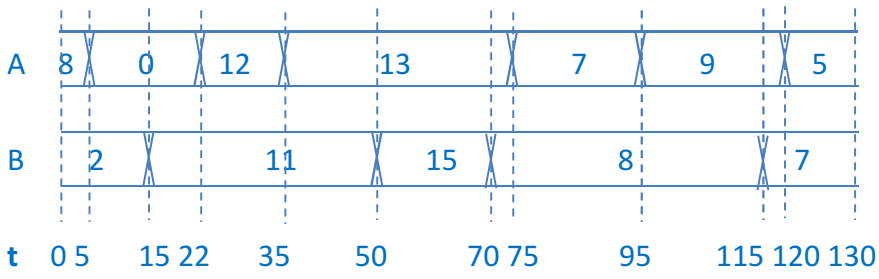
3 outputs A=B, A>B, A<B;

** Functional: [unsigned comparison]

A	B	A=B	A>B	A<B
A=B		1	0	0
A>B		0	1	0
A<B		0	0	1

** Timing: no delay(s) to be considered in this task

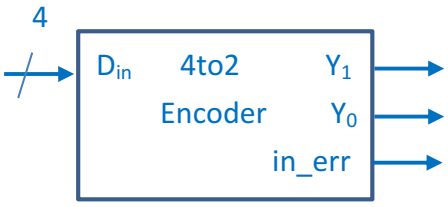
c)) testing data:



the testing stops at time-unit 130

[TASK3]

(1) Write a Verilog codes for a 4to2 encoder in compliance with the requirements given below.



(2) Put the codes under simulation; observe and interpret the waveforms of output signals

a)) coding specifications:

** write 3 modules of 4to2 encoder in gate-level, dataflow-level and behavior-level descriptions, respectively.

** the 3 modules will share the same 4-bit input, and with respective 2-bit encoded and 1-bit in_err output.

** illegal-input prevention mechanism should be included in the 3 modules (may use 4bit comparators or design of your own).

** write Verilog codes for generating test data sequence as requested.

b)) circuit specifications:

** I/O layout: 4-bit input $D_3D_2D_1D_0$,
2-bit output Y_1Y_0;

** Functional:

D_3	D_2	D_1	D_0	Y_1	Y_0	In_err
0	0	0	1	0	0	0
0	0	1	0	0	1	
0	1	0	0	1	0	
1	0	0	0	1	1	
otherwise				XX		1

** Timing: no delay(s) to be considered in this task

c)) testing data:

the testing stops at time-unit 130

[TASK4]

Choose one from any of the circuits given below.

a)) a 3to8 decoder with an additional input $\overline{EN}$ and low-active output.

b)) a 8bit comparator with an additional input EN and low-active output.

c)) an 8to3 encoder with an additional input EN and $\overline{in\_err}$ output.

(1) Write Verilog codes for the circuit you choose.

** It is advised that a block diagram with IO layout and an associated functional truth table be established prior to coding.

(2) Put the codes under simulation; observe and interpret the waveforms of output signals

a)) coding specifications:

** gate-level description only

b)) circuit specifications:

** I/O layout: as required by the circuit you choose;

** Functional: list the table of the circuit you choose;

** Timing: no delay(s) to be considered in this task

c)) testing data:

** specifying the test data sequence of your own.

[4] Observations/Evaluation

a)) Difficulties encountered while working on the assigned tasks may more or less reflect one's deficiency in one or more aspects of the following:

** experiences on the circuit structures and operations of the three basic modules which are building blocks deployed in all digital systems;

** experiences on the mastery of Verilog and the tools of the developing environment;

** readiness for the lab-task at hand.

Inexperience won't be a problem, as it fades away eventually when one pours in time and dedication; whereas the third would present a serious issue. The so called "poor-luck-in-the-laboratory" would persist, by which one will always be haunted as long as the unsound attitude continues.

b)) It's expected that, after DigitLabwork#1, one should feel confident, or at least comfortable, in dealing with circuit structures and operations of decoders, encoders, and comparators using Verilog. Hopefully this is the case of

everyone in the class.