

Fundamental Combinatorial Blocks

Topics

- Decoders and Encoders
- Behavioral simulation

Problems

Part I

1. *[Paper and pencil]* Design a 2:4 binary decoder, with 2 data inputs (X1 and X0), 4 data outputs (Y3...Y0) and 2 enable inputs, one active high (E1) and the other active low (E0_L). Its interface is shown in Fig. 1. Write the truth table, derive the Boolean equations and draw the logic diagram of the decoder.

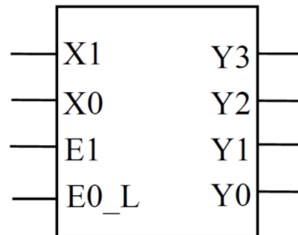


Fig. 1 - 2:4 binary decoder interface.

2. *[Quartus Prime]* Using the *Quartus Prime* software, create a new project named “DecoderDemo”, with a top-level entity with the same name as the project. Create a new file for a schematic diagram called “Dec2_4.bdf” to implement the decoder based on logic gates, accordingly to the logic diagram of the previous point. Create a symbol for the “Dec2_4” module, so that it can be used in a schematic diagram, and save it with the name “Dec2_4.bsf”.

3. *[Quartus Prime]* Create a new file for a schematic diagram called “DecoderDemo.bdf” that will act as the top-level of the project, instantiate the decoder built in the previous point and connect it to input and output ports.

4. *[Quartus Prime]* Perform the behavioural simulation of the decoder, applying input stimulus to evaluate conveniently its operation.

5. *[Paper and pencil]* Create a 4:16 binary decoder using the 2:4 binary decoder as a building block. Its interface is shown in Fig. 2 and consists of 4 data inputs (X3...X0), 16 data outputs (Y15...Y10) and 2 enable inputs, one active high (E1) and the other active low (E0_L). Draw the logic diagram of the 4:16 decoder based on 2:4 binary decoder modules.

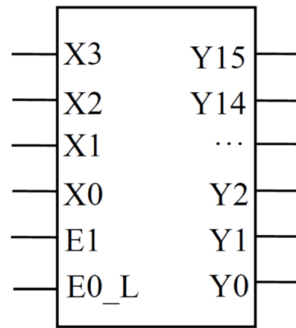


Fig. 2 - 4:16 binary decoder interface.

6. *[Quartus Prime]* Create a new file for a schematic diagram called “Dec4_16.bdf” to implement the 4:16 decoder, accordingly to the logic diagram of the previous point. Create a symbol for the “Dec4_16” module, so that it can be used in a schematic diagram, and save it with the name “Dec4_16.bsf”.

7. *[Quartus Prime]* Create a new file for a schematic diagram called “DecoderDemo2.bdf” and select it as the new top-level of the project, instantiate the 4:16 decoder built in the previous point and connect it to input and output ports.

8. *[Quartus Prime]* Perform the behavioural simulation of the 4:16 decoder, applying input stimulus to evaluate conveniently its operation.

Part II

Consider the following Boolean function not necessarily minimal:

$$f(A,B,C,D)=A'.B.C + A.D + A.C$$

1. *[Paper and pencil]* Draw the corresponding Karnaugh map and determine the first canonical form of the function $f(A,B,C,D)$.
2. *[Paper and pencil]* Create an implementation of $f(A,B,C,D)$ based on a 4:16 binary decoder and additional OR gate(s).
3. *[Quartus Prime]* Create a new file for a schematic diagram called “DecoderDemo3.bdf” and select it as the new top-level of the same project used in *Part I*, instantiate a 4:16 decoder with the additional OR gate(s), as drawn in the previous point, to implement $f(A,B,C,D)$.
4. *[Quartus Prime]* Simulate “DecoderDemo3.bdf” and check its behavior for all the possibilities of the $f(A,B,C,D)$ truth table.

Part III

1. *[Paper and pencil]* Design a priority encoder with 8-bit data inputs ($X_7 \dots X_0$) and 3-bit data outputs ($Y_2 \dots Y_0$). In addition to data inputs and outputs, the circuit must have an enable input (E_L) and an output strobe (OS_L) indicating whether the encoding is valid. Enable input (E_L) and output strobe (OS_L) must be both active low as shown in Fig. 3. An encoding is valid when both the input E_L and at least one of the data inputs (X_i) are active. Write the truth table of the priority encoder. To derive the Boolean equations of the priority encoder, use the intermediate H prioritized signals as shown in the lecture's slides, followed by a naïve encoder. Draw the logic diagram of the priority encoder based on elementary gates (AND, OR and inverters).

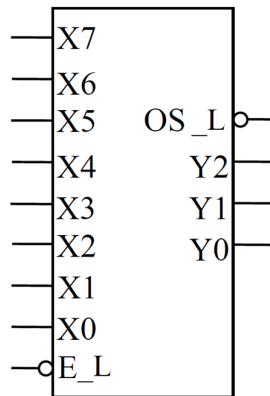


Fig. 3 - 8:3 priority encoder interface.

2. *[Quartus Prime]* Using the *Quartus Prime* software, create a new project named “EncoderDemo”, with a top-level entity with the same name as the project. Create a new file for a schematic diagram called “PEnc8_3.bdf” to implement the priority encoder based on logic gates, accordingly to the logic diagram of the previous point. Create a symbol for the “PEnc8_3” module, so that it can be used in a schematic diagram, and save it with the name “PEnc8_3.bsf”.

3. *[Quartus Prime]* Create a new file for a schematic diagram called “EncoderDemo.bdf” that will act as the top-level of the project, instantiate the priority encoder built in the previous point and connect it to input and output ports.

4. *[Quartus Prime]* Perform the behavioural simulation of the priority encoder, applying input stimulus to evaluate conveniently its operation.