Step By Step Learning Robei

III．Encoder and decoder

Robei LLC

1. Objective

Through the design of simple encoder and decoder to realize data transformation. Common encoder used in digital design including Gray encoder, BCD encoder, 8-3 line encoder and 16-4 encoder. In this lab, we will use 8-3 line encoder and 3-8 line decoder as example to study the encoder and decoder.

2. Requirement

Priority encoder can compress multiple binary inputs into less outputs algorithm and often used in handling priority interrupt request. 8-3 line encoder compresses 8 bit input to 3 bit output, 8 bit inputs line can only have one bit in high voltage each time. Truth table of 8-3 line encoder shows in table1. 3-8 line decoder is the inverse of 8-3 line encoder.

x[7]	x[6]	x[5]	x[4]	x[3]	x[2]	x[1]	x[0]		y[2]	y[1]	y[0]
0	0	0	0	0	0	0	1		0	0	0
0	0	0	0	0	0	1	0		0	0	1
0	0	0	0	0	1	0	0		0	1	0
0	0	0	0	1	0	0	0		0	1	1
0	0	0	1	0	0	0	0		1	0	0
0	0	1	0	0	0	0	0		1	0	1
0	1	0	0	0	0	0	0		1	1	0
1	0	0	0	0	0	0	0		1	1	1

表 1. 8-3 编码器真值表

3. Procedure

3.1 Encoder Design

1) Create a new module with name “encoder” with 2 input ports and one output port. Following fig.1 to modify the corresponding port properties.

Name	Inout	DataType	Datasize	Function
x	input	wire	7:0	input data
en	input	wire	1	enable
y	output	reg	2:0	output

Fig. 1. Ports' properties

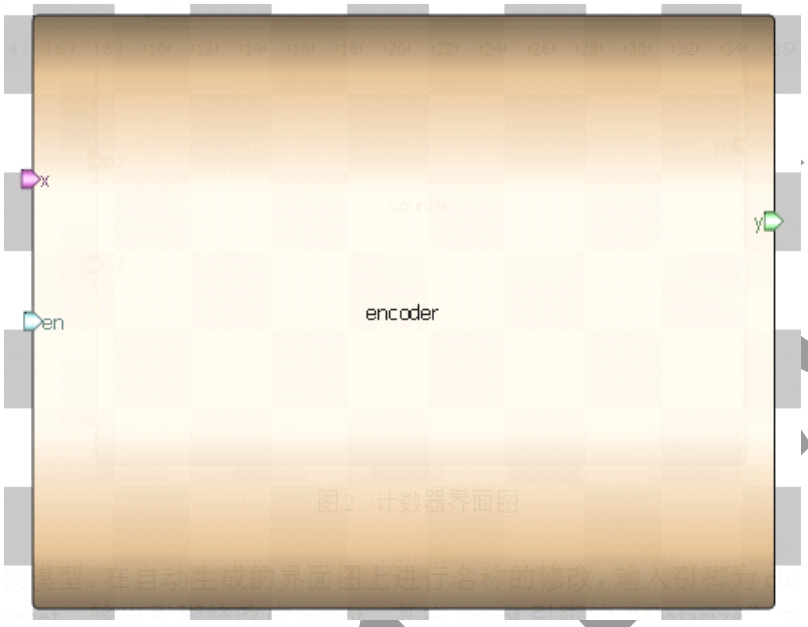

Fig. 2. Encoder interface

2) Add algorithm code. Click "Code" tab under design, and type in the following code.

encoder

```
16 always @ (en or x) begin
17   if (en) begin
18     case (x)   //学习case语句的写法
19       8'b00000001 : y = 3'b000;
20       8'b00000010 : y = 3'b001;
21       8'b00000100 : y = 3'b010;
22       8'b00001000 : y = 3'b011;
23       8'b00010000 : y = 3'b100;
24       8'b00100000 : y = 3'b101;
25       8'b01000000 : y = 3'b110;
26       8'b10000000 : y = 3'b111;
27      default : y= 3'b000;//别忘了设置default的值
28    endcase
29   end
30 end
```

Graph Code

Fig. 3. Add code algorithm

```
always @ (en or x) begin      //multiple sensitive signals in always
   if (en) begin
      case (x)       //learn how to write case in Verilog
      8'b00000001 : y = 3'b000;
      8'b00000010 : y = 3'b001;
      8'b00000100 : y = 3'b010;
      8'b00001000 : y = 3'b011;
      8'b00010000 : y = 3'b100;
      8'b00100000 : y = 3'b101;
      8'b01000000 : y = 3'b110;
      8'b10000000 : y = 3'b111;
     default : y= 3'b000;    //don't forgot default value
   endcase
 end
end
```

3) Save. Press icon  on Toolbar, or click on drop-down menu "Saveas" in menu "File", save this model in an empty folder. Note: this empty folder can not have white space in path, and can only use English characters.

4) Run. Press on icon  or click on drop-down menu "Run" in menu "Build".

   In this process, Robei will generate complete Verilog code and check. If there is any error, it will show error message in Output window. Please follow the instruction to correct any errors. After that, congratulation, you have complete the counter design.

3.2 Decoder design

1) Create a new design with name "decoder" and type as module. Set input ports to 2 and output ports to 1. Please follow fig.4 to set corresponding port's properties.

Name	Inout	DataType	Datasize	Function
din	input	wire	2:0	input data
en	input	wire	1	enable
out	output	wire	7:0	output

Fig. 4. Decoder ports

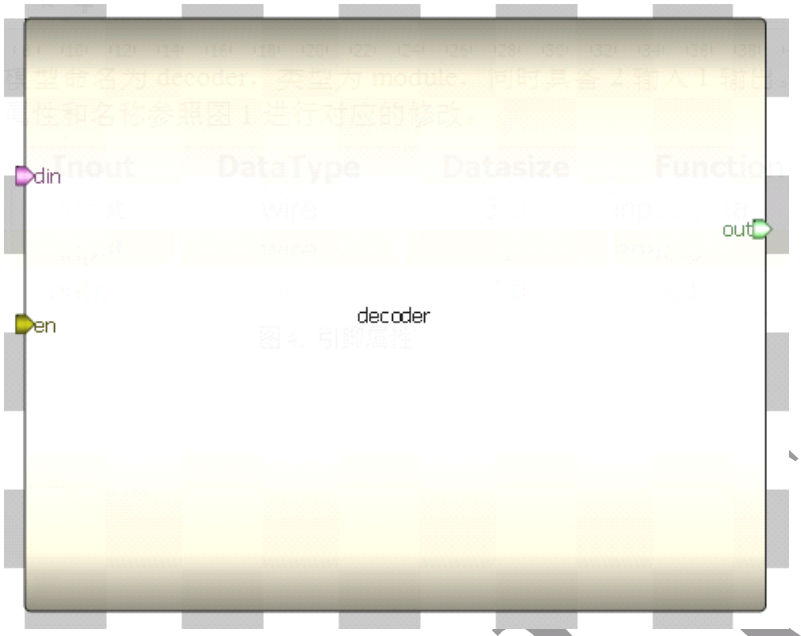

Fig. 5. Decoder interface

2) Add algorithm code. Click "Code" tab under design, and type in the following code.

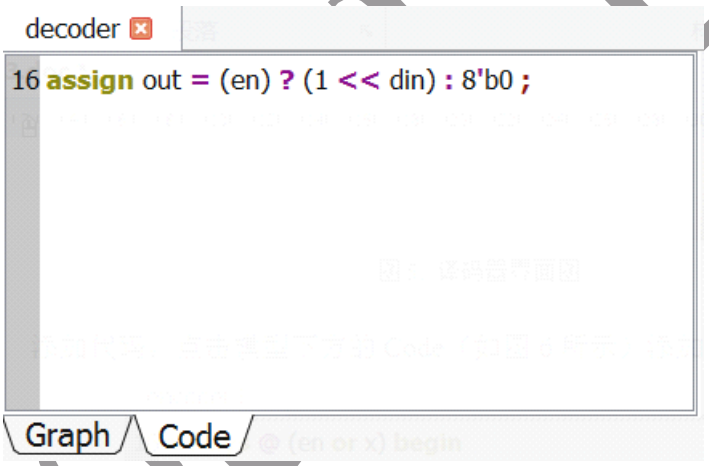

Fig. 6. Code algorithm for decoder

assign out = (en) ? (1 << din) : 8'b0 ; //if en is 1, shift right 1 with din times, otherwise out is 0. Learn how to use "?: " in Verilog

3) Save. Click on icon  , save module to the same folder as encoder.

4) Run. Click on icon  to run error check. Any errors will be reported in Output window.

3.2 Testbench design

1) Create a new design. Press on icon  in toolbar, in the pop up dialog, fill in blank with the following parameters in fig. 7.

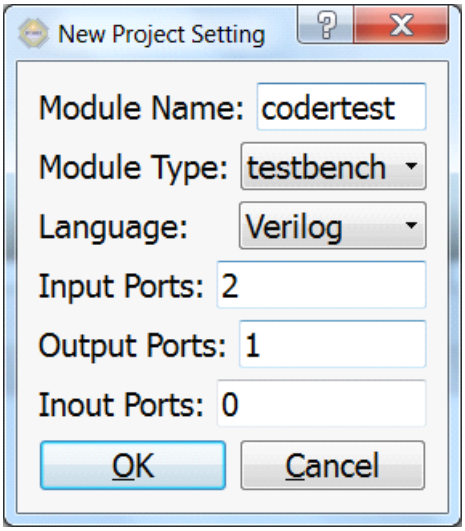

Fig. 7. Create testbench

2) Modify color of ports. Select each port, modify color from property editor. Please make sure the properties of each port are same as shown in fig.8.

Name	Inout	DataType	Datasize	Function
data	input	wire	2:0	signal data
en	input	wire	1	enable
out	output	wire	2:0	output data

Fig. 8 Ports properties

3) Save as testbench. Click on icon  in Toolbar, save testbench to the same location as “encoder” and “decoder” .

4) Add model. Under category “Current” in Toolbox, there will be two models showing up. Add each model on the testbench design, and connect them as fig.9.

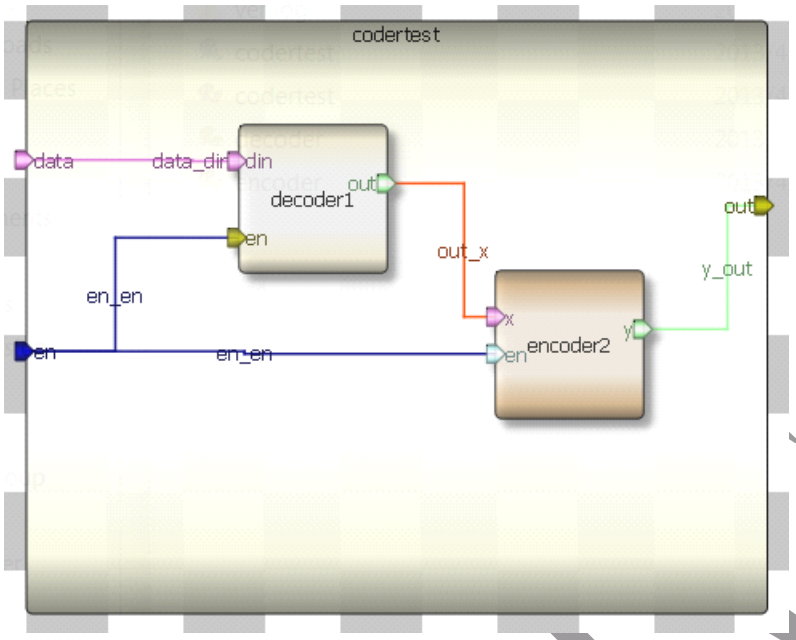

Fig. 9. Add model

5) Code stimulate. Press “Code” tab under testbech, type in the following code as stimulate for simulation. Remember to use $finish to end this simulation.

```
initial begin
   data=3'b000;
   en=0;
   #1 data=3'b000;
        en=1;
   #1 data=3'b001;
   #1 data=3'b010;
   #1 data=3'b011;
   #1 data=3'b100;
   #1 data=3'b101;
   #1 data=3'b110;
   #1 data=3'b111;
   #1 data=3'b000;
   #1 data=3'b001;
   #1 $finish;
end
```

codertest

```
15 initial begin
16   data=3'b000;
17   en=0;
18   #1 data=3'b000;
19      en=1;
20   #1 data=3'b001;
21   #1 data=3'b010;
22   #1 data=3'b011;
23   #1 data=3'b100;
24   #1 data=3'b101;
25   #1 data=3'b110;
26   #1 data=3'b111;
27   #1 data=3'b000;
28   #1 data=3'b001;
29   #1 $finish;
30 end
```

Graph Code

Fig. 10. Stimulate code

6) Run simulation and check the waveform. Click on  , check output information, if there is no error, then click on  . The Waveform window will show up. Click on signals listed in Workspace, add to wave window. Press  on toolbar of this window, it will zoom full of the wave. Analyze the waveform result and compare with your lab requirement.

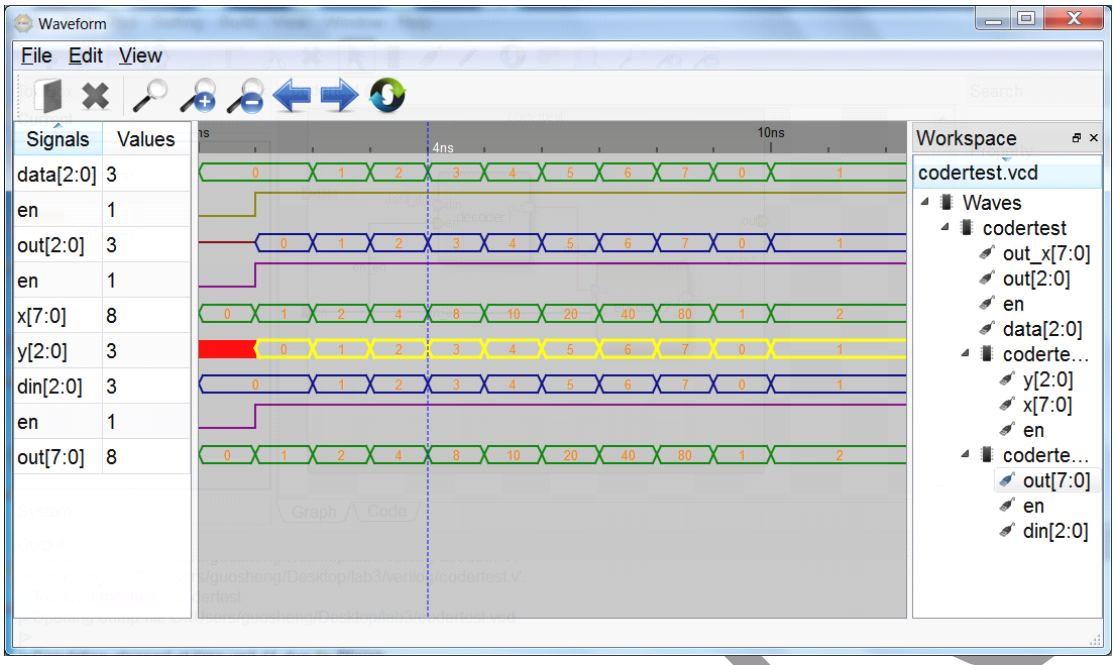

Fig. 11. Check waveform

4. Questions

1) Use Robei to design a BCD encoder and test it.
2) Use Robei to design 16-4 line encoder and 4-16 line decoder, and create testbench to verify your design.
3) Use Robei to design a Gray encoder and test it.