点阵字符显示控制器设计（仿真实现）

叶剑飞、刘艳杰、苏维、张印

湖南科技大学
计算机科学与工程学院
10 网络二班

2012 年 11 月 4 日

题目要求

题目要求

- 用 16×16 点阵的发光二极管显示字符;

题目要求

- 用 16×16 点阵的发光二极管显示字符;
- 可显示字符为 0~9 的数字字符与 A~F 英文字母;

题目要求

- 用 16×16 点阵的发光二极管显示字符;
- 可显示字符为 0~9 的数字字符与 A~F 英文字母;
- 输入为四位二进制矢量。

设计思路

设计思路

- 行共阳，列共阴。用 clk 作为时钟信号，逐行扫描显示。用 current_row 变量来储存当前显示的行，枚举每行的显示内容即可显示。

设计思路

- 行共阳，列共阴。用 clk 作为时钟信号，逐行扫描显示。用 current_row 变量来储存当前显示的行，枚举每行的显示内容即可显示。
- 四个输入按钮，分别代表四位二进制的输入信号。然后用 CASE 语句，下面 16 个 WHEN，分别处理“0000”至“1111”，外加上一个“WHEN OTHERS”，即可把这几个数字枚举转化成整型。把这个整型存在信号量 decimal 中。

设计思路

- 行共阳，列共阴。用 clk 作为时钟信号，逐行扫描显示。用 current_row 变量来储存当前显示的行，枚举每行的显示内容即可显示。
- 四个输入按钮，分别代表四位二进制的输入信号。然后用 CASE 语句，下面 16 个 WHEN，分别处理“0000”至“1111”，外加上一个“WHEN OTHERS”，即可把这几个数字枚举转化成整型。把这个整型存在信号量 decimal 中。
- 同样用 CASE-WHEN 枚举 16 个整数信号量，分别处理十六个整数的显示。

设计思路

- 行共阳，列共阴。用 clk 作为时钟信号，逐行扫描显示。用 current_row 变量来储存当前显示的行，枚举每行的显示内容即可显示。
- 四个输入按钮，分别代表四位二进制的输入信号。然后用 CASE 语句，下面 16 个 WHEN，分别处理“0000”至“1111”，外加上一个“WHEN OTHERS”，即可把这几个数字枚举转化成整型。把这个整型存在信号量 decimal 中。
- 同样用 CASE-WHEN 枚举 16 个整数信号量，分别处理十六个整数的显示。

VHDL 源代码

```
--  ------------------------------------------------------------------
--  点阵字符显示控制器设计
--
--
--  叶剑飞、刘艳杰、苏维、张印
--  ------------------------------------------------------------------

LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;

ENTITY binToHexDisplay IS
  PORT (    clk : IN STD_LOGIC;
       binaryInput : IN STD_LOGIC_VECTOR ( 3
                                DOWNTO 0 );
       rows : OUT STD_LOGIC_VECTOR(0 TO 15);
       columns: OUT STD_LOGIC_VECTOR(15
                                DOWNTO 0 )
    );
END ENTITY binToHexDisplay;

ARCHITECTURE showDigits OF binToHexDisplay IS
```

```
  SIGNAL decimal : INTEGER;
  SHARED VARIABLE current_row : INTEGER := 0;
BEGIN

  PROCESS (binaryInput)
  BEGIN
    CASE binaryInput IS
      WHEN "0000" =>  decimal <= 0;
      WHEN "0001" =>  decimal <= 1;
      WHEN "0010" =>  decimal <= 2;
      WHEN "0011" =>  decimal <= 3;
      WHEN "0100" =>  decimal <= 4;
      WHEN "0101" =>  decimal <= 5;
      WHEN "0110" =>  decimal <= 6;
      WHEN "0111" =>  decimal <= 7;
      WHEN "1000" =>  decimal <= 8;
      WHEN "1001" =>  decimal <= 9;
      WHEN "1010" =>  decimal <= 10;
      WHEN "1011" =>  decimal <= 11;
      WHEN "1100" =>  decimal <= 12;
      WHEN "1101" =>  decimal <= 13;
      WHEN "1110" =>  decimal <= 14;
```

```
    WHEN "1111" =>  decimal <= 15;
    WHEN OTHERS =>  decimal <= 1000;
  END CASE;
END PROCESS;

PROCESS (decimal,clk)
BEGIN
  IF clk='1' AND clk'EVENT THEN

    IF  current_row = 15  THEN
      current_row := 0;
    ELSE
      current_row := current_row + 1;
    END IF;

    CASE current_row IS
      WHEN 0  => rows <= "0111111111111111";
      WHEN 1  => rows <= "1011111111111111";
      WHEN 2  => rows <= "1101111111111111";
      WHEN 3  => rows <= "1110111111111111";
      WHEN 4  => rows <= "1111011111111111";
      WHEN 5  => rows <= "1111101111111111";
```

```
    WHEN 6  => rows <= "1111110111111111";
    WHEN 7  => rows <= "1111111011111111";
    WHEN 8  => rows <= "1111111101111111";
    WHEN 9  => rows <= "1111111110111111";
    WHEN 10 => rows <= "1111111111011111";
    WHEN 11 => rows <= "1111111111101111";
    WHEN 12 => rows <= "1111111111110111";
    WHEN 13 => rows <= "1111111111111011";
    WHEN 14 => rows <= "1111111111111101";
    WHEN 15 => rows <= "1111111111111110";
    WHEN OTHERS => rows<="XXXXXXXXXXXXXXXX";
END CASE;

CASE decimal IS
  WHEN 0  =>
    CASE current_row IS
      WHEN 0 =>columns<="0000000000000000";
      WHEN 1 =>columns<="0000000000000000";
      WHEN 2 =>columns<="0000011111100000";
      WHEN 3 =>columns<="0000111001110000";
      WHEN 4 =>columns<="0011100000011100";
      WHEN 5 =>columns<="0111000000001110";
```

```
        WHEN 6 =>columns<="0111000000001110";
        WHEN 7 =>columns<="0111000000001110";
        WHEN 8 =>columns<="0111000000001110";
        WHEN 9 =>columns<="0111000000001110";
        WHEN 10=>columns<="0111000000001110";
        WHEN 11=>columns<="0011100000011100";
        WHEN 12=>columns<="0000111001110000";
        WHEN 13=>columns<="0000011111100000";
        WHEN 14=>columns<="0000000000000000";
        WHEN 15=>columns<="0000000000000000";
        WHEN OTHERS =>
                columns <= "XXXXXXXXXXXXXXXX";
    END CASE;
  WHEN 1  =>
    CASE current_row IS
        WHEN 0 =>columns<="0000000000000000";
        WHEN 1 =>columns<="0000000011110000";
        WHEN 2 =>columns<="0000000111110000";
        WHEN 3 =>columns<="0000001101110000";
        WHEN 4 =>columns<="0000011001110000";
        WHEN 5 =>columns<="0000000001110000";
        WHEN 6 =>columns<="0000000001110000";
```

```
            WHEN 7 =>columns<="0000000001110000";
            WHEN 8 =>columns<="0000000001110000";
            WHEN 9 =>columns<="0000000001110000";
            WHEN 10=>columns<="0000000001110000";
            WHEN 11=>columns<="0000000001110000";
            WHEN 12=>columns<="0000000001110000";
            WHEN 13=>columns<="0000000001110000";
            WHEN 14=>columns<="0000001111111100";
            WHEN 15=>columns<="0000000000000000";
            WHEN OTHERS =>
                    columns <= "XXXXXXXXXXXXXXXX";
        END CASE;
    WHEN 2  =>
        CASE current_row IS
            WHEN 0 =>columns<="0000000000000000";
            WHEN 1 =>columns<="0000000000000000";
            WHEN 2 =>columns<="0000011111110000";
            WHEN 3 =>columns<="0001110000011100";
            WHEN 4 =>columns<="0011100000001110";
            WHEN 5 =>columns<="0011100000001110";
            WHEN 6 =>columns<="0000000000011100";
            WHEN 7 =>columns<="0000000000111000";
```

```
            WHEN 8 =>columns<="0000000011100000";
            WHEN 9 =>columns<="0000001110000000";
            WHEN 10=>columns<="0000011100000000";
            WHEN 11=>columns<="0001110000000000";
            WHEN 12=>columns<="0011111111111100";
            WHEN 13=>columns<="0111111111111100";
            WHEN 14=>columns<="0000000000000000";
            WHEN 15=>columns<="0000000000000000";
            WHEN OTHERS =>
                    columns <= "XXXXXXXXXXXXXXXX";
        END CASE;
    WHEN 3  =>
        CASE current_row IS
            WHEN 0 =>columns<="0000000000000000";
            WHEN 1 =>columns<="0001111111000000";
            WHEN 2 =>columns<="0000000001110000";
            WHEN 3 =>columns<="0000000000111000";
            WHEN 4 =>columns<="0000000000001100";
            WHEN 5 =>columns<="0000000000001100";
            WHEN 6 =>columns<="0000000001110000";
            WHEN 7 =>columns<="0001111111000000";
            WHEN 8 =>columns<="0000000001110000";
```

```
            WHEN 9 =>columns<="0000000000111000";
            WHEN 10=>columns<="0000000000001100";
            WHEN 11=>columns<="0000000000001100";
            WHEN 12=>columns<="0000000000111000";
            WHEN 13=>columns<="0000000001110000";
            WHEN 14=>columns<="0001111111000000";
            WHEN 15=>columns<="0000000000000000";
            WHEN OTHERS =>
                    columns <= "XXXXXXXXXXXXXXXX";
        END CASE;
    WHEN 4   =>
        CASE current_row IS
            WHEN 0 =>columns<="0000000000000000";
            WHEN 1 =>columns<="0000000010000000";
            WHEN 2 =>columns<="0000001110000000";
            WHEN 3 =>columns<="0000110110000000";
            WHEN 4 =>columns<="0000110110000000";
            WHEN 5 =>columns<="0001100110000000";
            WHEN 6 =>columns<="0001100110000000";
            WHEN 7 =>columns<="0011000110000000";
            WHEN 8 =>columns<="0111111111111100";
            WHEN 9 =>columns<="1111111111111100";
```

```
            WHEN 10=>columns<="0000000110000000";
            WHEN 11=>columns<="0000000110000000";
            WHEN 12=>columns<="0000000110000000";
            WHEN 13=>columns<="0000000110000000";
            WHEN 14=>columns<="0000001111000000";
            WHEN 15=>columns<="0000000000000000";
            WHEN OTHERS =>
                    columns <= "XXXXXXXXXXXXXXXX";
        END CASE;
    WHEN 5   =>
        CASE current_row IS
            WHEN 0 =>columns<="0000000000000000";
            WHEN 1 =>columns<="0011111111111100";
            WHEN 2 =>columns<="0011111111111100";
            WHEN 3 =>columns<="0011100000000000";
            WHEN 4 =>columns<="0011100000000000";
            WHEN 5 =>columns<="0011100000000000";
            WHEN 6 =>columns<="0011100000000000";
            WHEN 7 =>columns<="0011100000000000";
            WHEN 8 =>columns<="0011111111000000";
            WHEN 9 =>columns<="0000000001110000";
            WHEN 10=>columns<="0000000000111000";
```

```
            WHEN 11=>columns<="0000000000011100";
            WHEN 12=>columns<="0011100000011100";
            WHEN 13=>columns<="0011100000111000";
            WHEN 14=>columns<="0001111111100000";
            WHEN 15=>columns<="0000000000000000";
            WHEN OTHERS =>
                    columns <= "XXXXXXXXXXXXXXXX";
        END CASE;
    WHEN 6   =>
        CASE current_row IS
            WHEN 0 =>columns<="0000000000000000";
            WHEN 1 =>columns<="0000111111110000";
            WHEN 2 =>columns<="0011100000011100";
            WHEN 3 =>columns<="0011100000000011";
            WHEN 4 =>columns<="0111000000000000";
            WHEN 5 =>columns<="0111000000000000";
            WHEN 6 =>columns<="1110000000000000";
            WHEN 7 =>columns<="1110000000000000";
            WHEN 8 =>columns<="1110011111100000";
            WHEN 9 =>columns<="1111111111111000";
            WHEN 10=>columns<="1111000001111100";
            WHEN 11=>columns<="1110000000011110";
```

```
            WHEN 12=>columns<="0110000000001111";
            WHEN 13=>columns<="0110000000001100";
            WHEN 14=>columns<="0011100000111000";
            WHEN 15=>columns<="0000111111110000";
            WHEN OTHERS =>
                    columns <= "XXXXXXXXXXXXXXXX";
        END CASE;
    WHEN 7   =>
        CASE current_row IS
            WHEN 0 =>columns<="0000000000000000";
            WHEN 1 =>columns<="0011111111111110";
            WHEN 2 =>columns<="0011111111111100";
            WHEN 3 =>columns<="0000000000001100";
            WHEN 4 =>columns<="0000000000011000";
            WHEN 5 =>columns<="0000000000110000";
            WHEN 6 =>columns<="0000000001110000";
            WHEN 7 =>columns<="0000000001110000";
            WHEN 8 =>columns<="0000000011100000";
            WHEN 9 =>columns<="0000000011100000";
            WHEN 10=>columns<="0000000111000000";
            WHEN 11=>columns<="0000000111000000";
            WHEN 12=>columns<="0000001110000000";
```

```
        WHEN 13=>columns<="0000001110000000";
        WHEN 14=>columns<="0000001110000000";
        WHEN 15=>columns<="0000001110000000";
        WHEN OTHERS =>
                columns <= "XXXXXXXXXXXXXXXX";
    END CASE;
WHEN 8  =>
    CASE current_row IS
        WHEN 0 =>columns<="0000111111110000";
        WHEN 1 =>columns<="0001110000111000";
        WHEN 2 =>columns<="0011100000011100";
        WHEN 3 =>columns<="0111000000001110";
        WHEN 4 =>columns<="0111000000001110";
        WHEN 5 =>columns<="0011100000011100";
        WHEN 6 =>columns<="0001110000111000";
        WHEN 7 =>columns<="0000011111100000";
        WHEN 8 =>columns<="0000011111100000";
        WHEN 9 =>columns<="0001110000111000";
        WHEN 10=>columns<="0011100000011100";
        WHEN 11=>columns<="0111000000001110";
        WHEN 12=>columns<="0111000000001110";
        WHEN 13=>columns<="0011100000011100";
```

```
        WHEN 14=>columns<="0001110000111000";
        WHEN 15=>columns<="0000111111110000";
        WHEN OTHERS =>
                columns <= "XXXXXXXXXXXXXXXX";
    END CASE;
WHEN 9  =>
    CASE current_row IS
        WHEN 0 =>columns<="0000011111100000";
        WHEN 1 =>columns<="0001111001111000";
        WHEN 2 =>columns<="0011100000011100";
        WHEN 3 =>columns<="0111000000001110";
        WHEN 4 =>columns<="1110000000000111";
        WHEN 5 =>columns<="1110000000000111";
        WHEN 6 =>columns<="0111000000001110";
        WHEN 7 =>columns<="0001111001111111";
        WHEN 8 =>columns<="0000011111000111";
        WHEN 9 =>columns<="0000000000001110";
        WHEN 10=>columns<="0000000000011100";
        WHEN 11=>columns<="0000000001110000";
        WHEN 12=>columns<="0000000111000000";
        WHEN 13=>columns<="0000011110000000";
        WHEN 14=>columns<="0011110000000000";
```

```
        WHEN 15=>columns<="0000000000000000";
        WHEN OTHERS =>
                columns <= "XXXXXXXXXXXXXXXX";
    END CASE;
  WHEN 10 =>
    CASE current_row IS
        WHEN 0 =>columns<="0000001111000000";
        WHEN 1 =>columns<="0000011001100000";
        WHEN 2 =>columns<="0000110000110000";
        WHEN 3 =>columns<="0000110000110000";
        WHEN 4 =>columns<="0001100000011000";
        WHEN 5 =>columns<="0001100000011000";
        WHEN 6 =>columns<="0001100000011000";
        WHEN 7 =>columns<="0011000000001100";
        WHEN 8 =>columns<="0011111111111100";
        WHEN 9 =>columns<="0011111111111100";
        WHEN 10=>columns<="0110000000000110";
        WHEN 11=>columns<="0110000000000110";
        WHEN 12=>columns<="0110000000000110";
        WHEN 13=>columns<="1100000000000011";
        WHEN 14=>columns<="1100000000000011";
        WHEN 15=>columns<="1100000000000011";
```

```
        WHEN OTHERS =>
                columns <= "XXXXXXXXXXXXXXXX";
    END CASE;
WHEN 11 =>
    CASE current_row IS
        WHEN 0 =>columns<="1111111111100000";
        WHEN 1 =>columns<="1100000001111000";
        WHEN 2 =>columns<="1100000000001110";
        WHEN 3 =>columns<="1100000000000111";
        WHEN 4 =>columns<="1100000000000111";
        WHEN 5 =>columns<="1100000000001110";
        WHEN 6 =>columns<="1100000001111000";
        WHEN 7 =>columns<="1111111111100000";
        WHEN 8 =>columns<="1111111111100000";
        WHEN 9 =>columns<="1100000001111000";
        WHEN 10=>columns<="1100000000001110";
        WHEN 11=>columns<="1100000000000111";
        WHEN 12=>columns<="1100000000000111";
        WHEN 13=>columns<="1100000000001110";
        WHEN 14=>columns<="1100000001111000";
        WHEN 15=>columns<="1111111111100000";
        WHEN OTHERS =>
```

```
                columns <= "XXXXXXXXXXXXXXXX";
    END CASE;
  WHEN 12 =>
    CASE current_row IS
      WHEN 0 =>columns<="0000111111110000";
      WHEN 1 =>columns<="0001110000011100";
      WHEN 2 =>columns<="0011100000001110";
      WHEN 3 =>columns<="0110000000000111";
      WHEN 4 =>columns<="0110000000000000";
      WHEN 5 =>columns<="1100000000000000";
      WHEN 6 =>columns<="1100000000000000";
      WHEN 7 =>columns<="1100000000000000";
      WHEN 8 =>columns<="1100000000000000";
      WHEN 9 =>columns<="1100000000000000";
      WHEN 10=>columns<="0110000000000000";
      WHEN 11=>columns<="0110000000000000";
      WHEN 12=>columns<="0110000000000111";
      WHEN 13=>columns<="0011100000001110";
      WHEN 14=>columns<="0001110000011100";
      WHEN 15=>columns<="0000111111110000";
      WHEN OTHERS =>
                columns <= "XXXXXXXXXXXXXXXX";
```

```
    END CASE;
  WHEN 13 =>
    CASE current_row IS
      WHEN 0 =>columns<="1111100000000000";
      WHEN 1 =>columns<="1111111110000000";
      WHEN 2 =>columns<="1100000111000000";
      WHEN 3 =>columns<="1100000001110000";
      WHEN 4 =>columns<="1100000000011100";
      WHEN 5 =>columns<="1100000000000110";
      WHEN 6 =>columns<="1100000000000011";
      WHEN 7 =>columns<="1100000000000011";
      WHEN 8 =>columns<="1100000000000011";
      WHEN 9 =>columns<="1100000000000110";
      WHEN 10=>columns<="1100000000000110";
      WHEN 11=>columns<="1100000000001100";
      WHEN 12=>columns<="1100000001111000";
      WHEN 13=>columns<="1100001111000000";
      WHEN 14=>columns<="1111111000000000";
      WHEN 15=>columns<="1111100000000000";
      WHEN OTHERS =>
              columns <= "XXXXXXXXXXXXXXXX";
    END CASE;
```

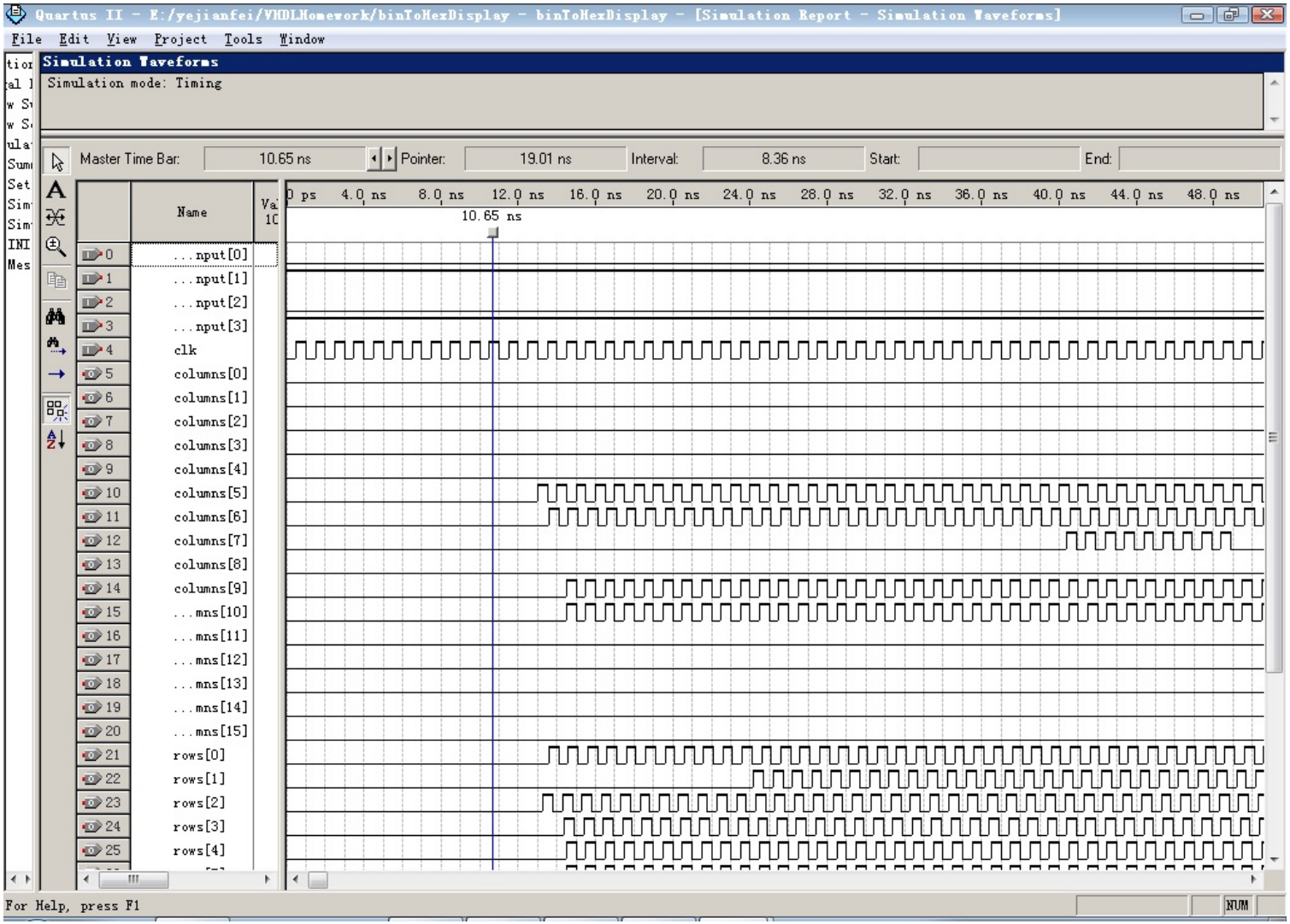
```
WHEN 14 =>
   CASE current_row IS
      WHEN 0 =>columns<="1111111111111111";
      WHEN 1 =>columns<="1111111111111111";
      WHEN 2 =>columns<="1100000000000000";
      WHEN 3 =>columns<="1100000000000000";
      WHEN 4 =>columns<="1100000000000000";
      WHEN 5 =>columns<="1100000000000000";
      WHEN 6 =>columns<="1100000000000000";
      WHEN 7 =>columns<="1111111111111111";
      WHEN 8 =>columns<="1111111111111111";
      WHEN 9 =>columns<="1100000000000000";
      WHEN 10=>columns<="1100000000000000";
      WHEN 11=>columns<="1100000000000000";
      WHEN 12=>columns<="1100000000000000";
      WHEN 13=>columns<="1100000000000000";
      WHEN 14=>columns<="1111111111111111";
      WHEN 15=>columns<="1111111111111111";
      WHEN OTHERS =>
               columns <= "XXXXXXXXXXXXXXXX";
   END CASE;
WHEN 15 =>
```

```
    CASE current_row IS
      WHEN 0 =>columns<="1111111111111111";
      WHEN 1 =>columns<="1111111111111111";
      WHEN 2 =>columns<="1100000000000000";
      WHEN 3 =>columns<="1100000000000000";
      WHEN 4 =>columns<="1100000000000000";
      WHEN 5 =>columns<="1100000000000000";
      WHEN 6 =>columns<="1100000000000000";
      WHEN 7 =>columns<="1111111111111111";
      WHEN 8 =>columns<="1111111111111111";
      WHEN 9 =>columns<="1100000000000000";
      WHEN 10=>columns<="1100000000000000";
      WHEN 11=>columns<="1100000000000000";
      WHEN 12=>columns<="1100000000000000";
      WHEN 13=>columns<="1100000000000000";
      WHEN 14=>columns<="1100000000000000";
      WHEN 15=>columns<="1100000000000000";
      WHEN OTHERS =>
              columns <= "XXXXXXXXXXXXXXXX";
    END CASE;
  WHEN OTHERS =>
    columns <= "XXXXXXXXXXXXXXXX";
```

```
        END CASE;
    END IF;
  END PROCESS;

END ARCHITECTURE showDigits;
```

仿真实现


Quartus II - E:/yejianfei/VHDLHomework/binToHexDisplay - binToHexDisplay - [Simulation Report - Simulation Waveforms]
File Edit View Project Tools Window
Simulation Waveforms
Simulation mode: Timing
Master Time Bar: 10.65 ns
Pointer: 19.01 ns
Interval: 8.36 ns
Start:
End:
Name
10.65 ns
clk
columns[0]
columns[1]
columns[2]
columns[3]
columns[4]
columns[5]
columns[6]
columns[7]
columns[8]
columns[9]
rows[0]
rows[1]
rows[2]
rows[3]
rows[4]
For Help, press F1
NUM

演示完毕！