

# 深圳大学实验报告

课程名称： 硬件描述语言及数字系统设计

实验项目名称： 实验二 流水灯设计

学院： 医学院

专业： 生物医学工程

指导教师： 但果、董磊

报告人： 陈焕鑫 学号： 2016222042 班级： 生工 2 班

实验时间： 2018.10.10

实验报告提交时间： 2018.10.24

教务部制

## 一、实验平台：

安装了 ISE 软件的 PC 计算机  
硬件描述语言及数字系统设计实验平台  
JTAG 下载线

## 二、实验目的：

当你完成整个项目之后，你将会学会一下内容：  
时序电路的描述方法（学习时注意与组合电路的区别）  
如何将时钟进行分频  
顶层模块调用底层模块

## 三、实验内容：

用 VHDL 代码描述一个分频电路（系统时钟为 50MHz）  
基于分频电路子模块，用 VHDL 代码编写一个 4 位流水灯程序  
编写测试激励（仿真代码），对编译好的 4 位流水灯程序进行仿真调试  
编写引脚约束文件，用 ISE 软件生成 .bit 文件，下载到硬件描述语言及数字系统设计实验平台进行板级验证

## 四、实验原理：

流水灯的工作过程就是四个 LED 发光二极管（LD7~LD4）依次点亮、熄灭，形成流水状。具体来说，就是先让最左边的一个 LED 点亮，等待一小段时间后熄灭，再让第二个 LED 点亮，等待一小段时间后熄灭，如此类推，当最边的 LED 熄灭后，再点亮最左边的第一个 LED，形成循环，现象如流水状，如图 4-1 所示。

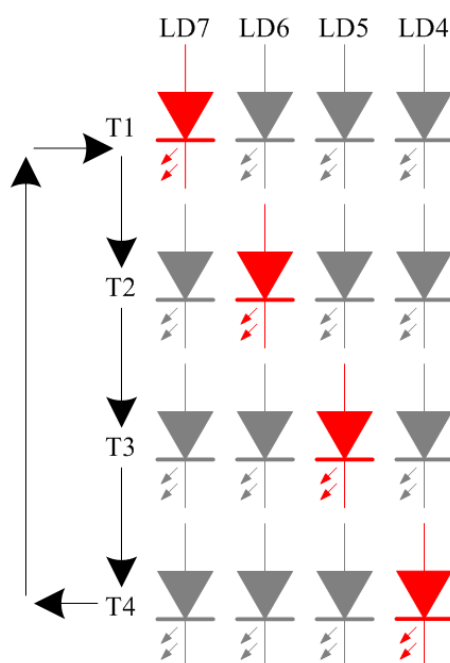


图 4-1 流水灯工作过程

需要说明的是，硬件描述语言及数字系统设计实验平台的系统时钟的频率是 50MHz，流水灯是呀需要的时钟频率一般都比较低，如果流水灯 1s 移动一位，那么久需要 1Hz 的时钟，因此就需要对高频率的时钟进行分频，产生低频率的时钟。

分频是指将一个高频信号的频率降低为原来的 1/N，就叫 N 分频。如吧 50MHz 的信号 2 分频得到 25MHz 的信号，5 分频得到 10MHz 的信号，10 分频得到 5MHz 的信号。

图 4-2 所示的是将一个周期为 T 的高频率时钟进行分频，产生周期为 2,mT 的低频率时钟，也就是进行 2m 分频。

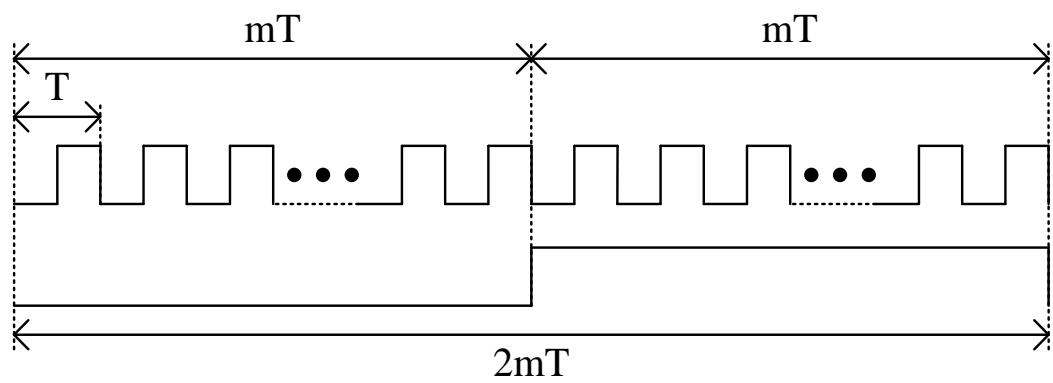


图 4-2 2m 分频波形图

## 五、实验方法步骤及 VHDL 代码:

根据分频原理进行代码设计，设计一个产生 1Hz 的时钟信号。

具体的 VHDL 代码如下所示:

```
--模块名称: clk_gen_1hz
--摘要提示:
--当前版本: 1.0.0
--模块作者:
--完成日期: 20xx 年 xx 月 xx 日
--内容提要:
--需要注意:

--取代版本:
--模块作者:
--完成日期:
--修改内容:
--修改文件:

library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

-----ENTITY DECLARATION-----
```

```

entity clk_gen_1hz is
    generic(
        CNT_MAX : integer := 50000000;
        CNT_HALF: integer := 25000000
    );

    port(
        rst_n_i : in std_logic;
        clk_i   : in std_logic;
        clk_o   : out std_logic
    );
end clk_gen_1hz;

-----
-----ARCHITECTURE STRUCTURAL-----
architecture rtl of clk_gen_1hz is

    signal s_cnt : integer range 0 to CNT_MAX := 0;

begin

    process(clk_i, rst_n_i, s_cnt)
    begin
        if(rst_n_i = '0')then
            s_cnt <= 0;
            clk_o <= '0';
        elsif rising_edge(clk_i) then
            if (s_cnt = CNT_MAX) then
                s_cnt <= 0;
                clk_o <= '0';
            elsif (s_cnt = CNT_HALF) then
                s_cnt <= s_cnt + 1;
                clk_o <= '1';
            else
                s_cnt <= s_cnt + 1;
            end if;
        end if;
    end process;

end rtl;

```

计数器模块，每接收到一次时钟信号就计数一次，计数器在“00”到“11”之间反复循环。

-----  
--模块名称: counter

--摘要提示:

--当前版本: 1.0.0

--模块作者:

--完成日期: 20xx 年 xx 月 xx 日

--内容提要:

--需要注意:

-----  
--取代版本:

--模块作者:

--完成日期:

--修改内容:

--修改文件:

-----  
**library** ieee;

**use** ieee.std\_logic\_1164.all;

**use** ieee.std\_logic\_arith.all;

**use** ieee.std\_logic\_unsigned.all;

-----ENTITY DECLARATION-----

**entity** counter **is**

**port**(

        clk      : **in**      std\_logic;

        rst\_n    : **in**      std\_logic;

        c\_out    : **out** std\_logic\_vector(1 **downto** 0)

    );

**end** counter;

-----ARCHITECTURE STRUCTURAL-----

**architecture** rtl **of** counter **is**

**signal** s\_cnt : std\_logic\_vector(1 **downto** 0);

**begin**

**process**(rst\_n, clk)

**begin**

**if**(rst\_n = '0')**then**

            s\_cnt <= "00";

**elsif** rising\_edge(clk)**then**

            s\_cnt <= s\_cnt + "01";

**end if**;

        c\_out <= s\_cnt;

```
    end process;
end rtl;
```

选择器模块,可以根据计数器输出不同的结果来采取不同的输出控制 LED 的不同状态。

```
--模块名称: mux4
--摘要提示:
--当前版本: 1.0.0
--模块作者:
--完成日期: 20xx 年 xx 月 xx 日
--内容提要:
--需要注意:
```

```
--取代版本:
--模块作者:
--完成日期:
--修改内容:
--修改文件:
```

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
```

```
-----ENTITY DECLARATION-----
entity mux4 is
    port(
        sel : in    std_logic_vector(1 downto 0);
        y   : out   std_logic_vector(3 downto 0)
    );
end mux4;
```

```
-----ARCHITECTURE STRUCTURAL-----
architecture rtl of mux4 is

begin
    process(sel)
    begin
        case sel is
            when "00" => y <= "1110";
            when "01" => y <= "1101";
            when "10" => y <= "1011";
```

```

        when "11" => y <= "0111";
        when others => y <= "1111";
    end case;
end process;

end rtl;

```

该文件通过编写程序调用上面编写的时钟模块，计数器模块和选择器模块，将这些模块例化来实现 LED 流水灯的功能。

---

```
--模块名称: water_led
```

```
--摘要提示:
```

```
--当前版本: 1.0.0
```

```
--模块作者:
```

```
--完成日期: 20xx 年 xx 月 xx 日
```

```
--内容提要:
```

```
--需要注意:
```

---

```
--取代版本:
```

```
--模块作者:
```

```
--完成日期:
```

```
--修改内容:
```

```
--修改文件:
```

---

```
library ieee;
```

```
use ieee.std_logic_1164.all;
```

```
use ieee.std_logic_arith.all;
```

```
use ieee.std_logic_unsigned.all;
```

---

```
-----ENTITY DECLARATION-----
```

```
entity water_led is
```

```
    port(
```

```
        clk_50mhz_i : in std_logic;
```

```
        rst_n_i      : in std_logic;
```

```
        water_led_o : out std_logic_vector(3 downto 0)
```

```
    );
```

```
end water_led;
```

---



---

```
-----ARCHITECTURE STRUCTURAL-----
```

```
architecture structural of water_led is
```

```

component clk_gen_1hz is

    port(
        rst_n_i : in  std_logic;
        clk_i    : in  std_logic;
        clk_o    : out std_logic
    );
end component;

-----

component counter is

    port(
        clk      : in    std_logic;
        rst_n    : in    std_logic;
        c_out    : out std_logic_vector(1 downto 0)
    );
end component;

-----

component mux4 is

    port(
        sel : in    std_logic_vector(1 downto 0);
        y   : out std_logic_vector(3 downto 0)
    );
end component;

-----

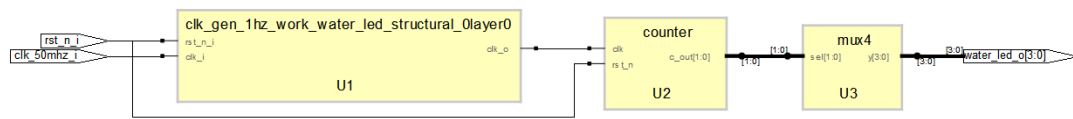
signal mid_1hz_o : std_logic;
signal sel_signal_o : std_logic_vector(1 downto 0);
begin
    U1 : clk_gen_1hz port map(rst_n_i, clk_50mhz_i, mid_1hz_o);
    U2 : counter      port map(mid_1hz_o, rst_n_i, sel_signal_o);
    U3 : mux4         port map(sel_signal_o, water_led_o);
end structural;

```

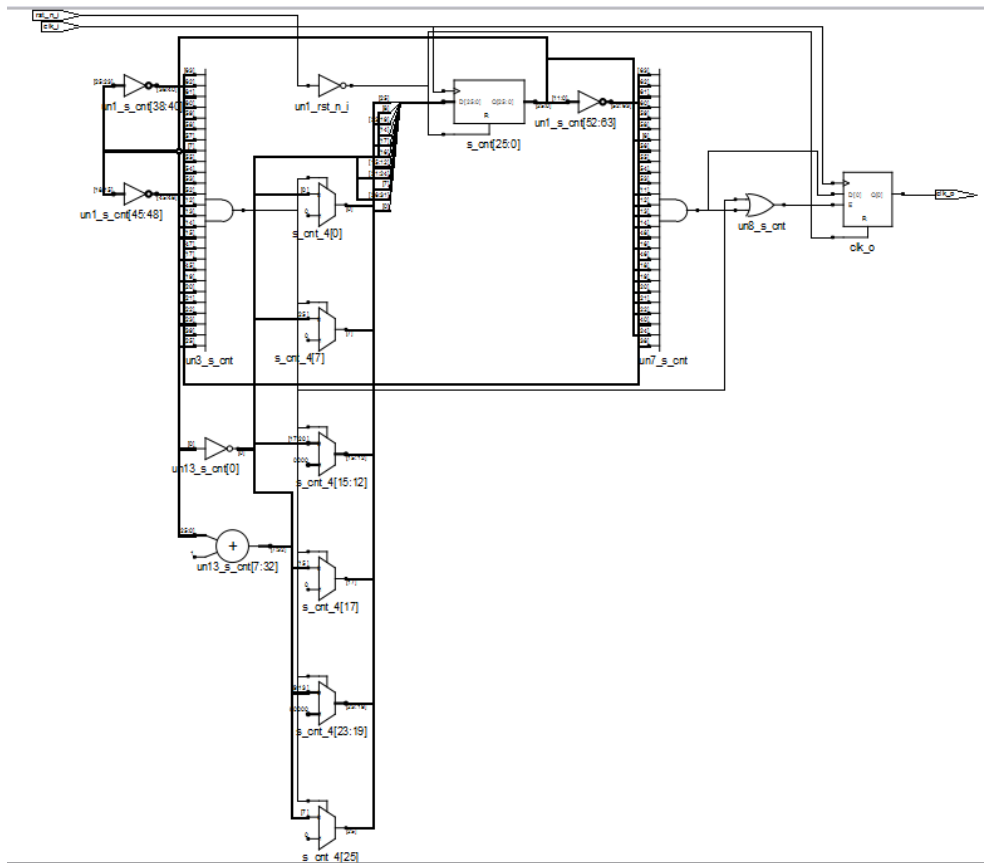


## 六、综合、仿真结果及分析：

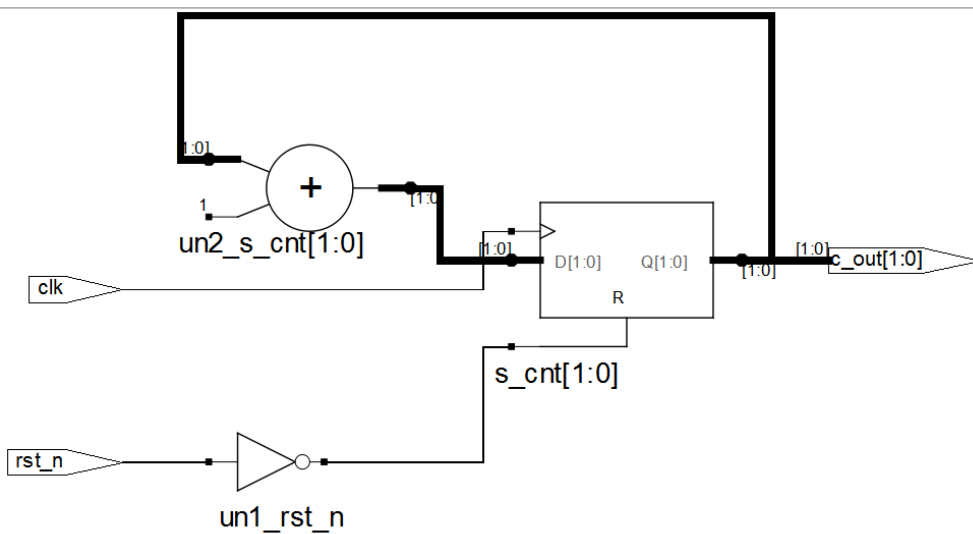
### 1、RTL 级电路综合：



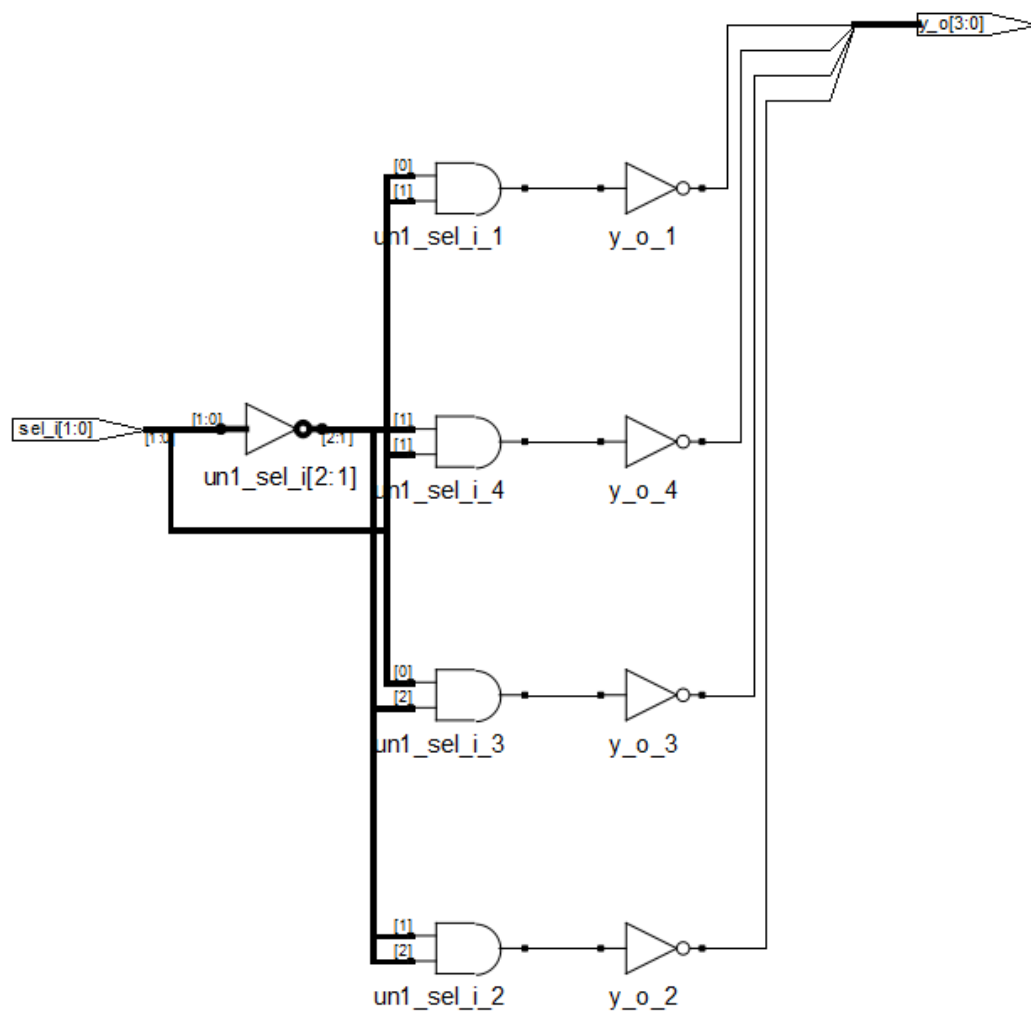
1Hz 时钟信号发生器电路图：



Counter 计数器电路图：



Mux4 选择器电路图：



## 2、电路仿真

测试激励 Test Bench 代码如下：

```
--模块名称: water_led_tb
--摘要提示: 测试激励
--当前版本: 1.0.0
--模块作者:
--完成日期: 2018 年 9 月 19 日
--内容提要:
--需要注意:

--取代版本:
--模块作者:
--完成日期:
--修改内容:
--修改文件:
```

```

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY water_led_tb IS
END water_led_tb;

ARCHITECTURE behavior OF water_led_tb IS
    -- Component Declaration for the Unit Under Test (UUT)
    COMPONENT water_led
    PORT(
        clk_50mhz_i : IN std_logic;
        rst_n_i : IN std_logic;
        water_led_o : OUT std_logic_vector(3 downto 0)
    );
    END COMPONENT;

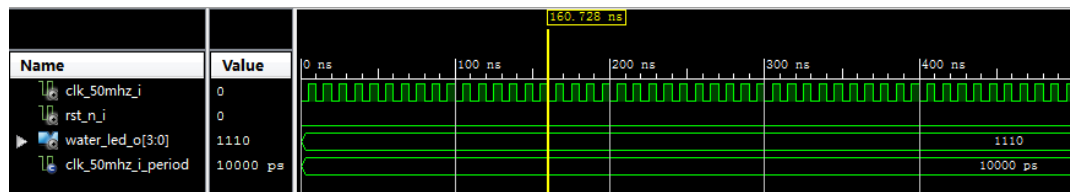
    --Inputs
    signal clk_50mhz_i : std_logic := '0';
    signal rst_n_i : std_logic := '0';
    --Outputs
    signal water_led_o : std_logic_vector(3 downto 0);
    -- Clock period definitions
    constant clk_50mhz_i_period : time := 10 ns;
BEGIN
    uut: water_led PORT MAP (
        clk_50mhz_i => clk_50mhz_i,
        rst_n_i => rst_n_i,
        water_led_o => water_led_o
    );
    clk_50mhz_i_process :process
    begin
        clk_50mhz_i <= '0';
        wait for clk_50mhz_i_period/2;
        clk_50mhz_i <= '1';
        wait for clk_50mhz_i_period/2;
    end process;
    stim_proc: process
    begin
        wait for 100 ns;
        wait for clk_50mhz_i_period*10;
        rst_n_i <= '0';
        wait for 500 ns;
        rst_n_i <= '1';
        wait;
    end process;

```

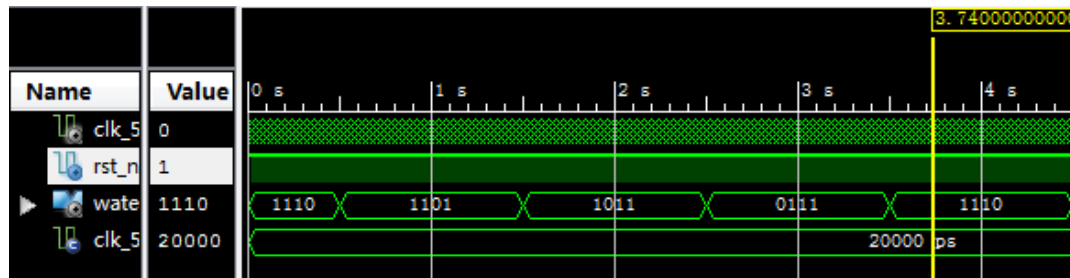
END ;

仿真结果如下图所示：

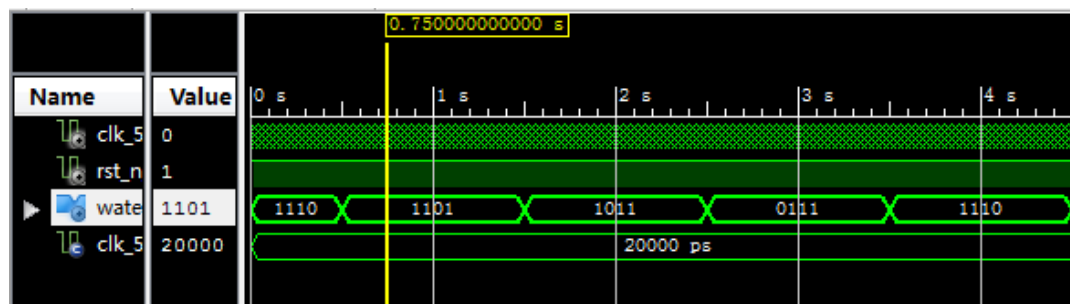
rst 为 0 时，只有第一个 LED 灯被电路



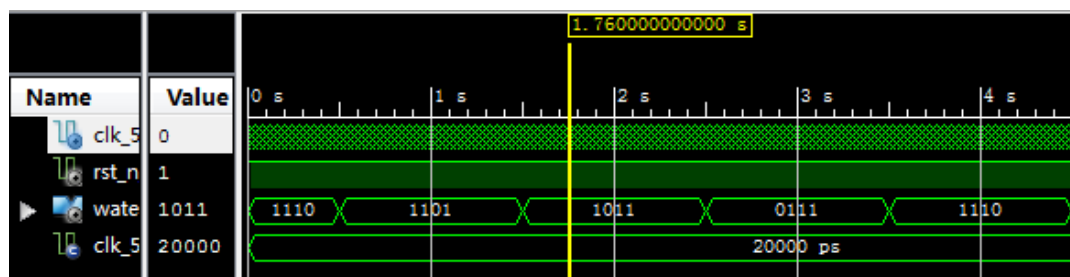
rst 为 1，选择器输出 1110:



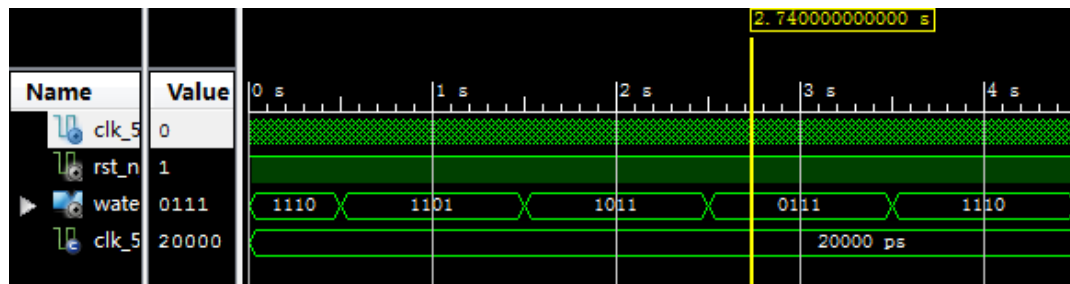
rst 为 1，选择器输出 1101:



rst 为 1，选择器输出 1011:



rst 为 1，选择器输出 0111:



## 七、FPGA 板级验证及结果分析

1、在工程中加入引脚约束文件，代码如下

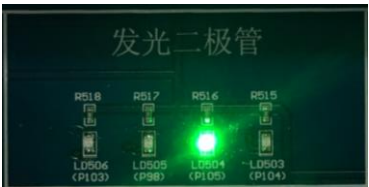
```
##-----  
##模块名称: water_led.ucf  
##摘要提示:  
##当前版本: 1.0.0  
##模块作者:  
##完成日期: 20xx 年 xx 月 xx 日  
##内容提要:  
##需要注意:  
##-----  
##取代版本:  
##模块作者:  
##完成日期:  
##修改内容:  
##修改文件:  
##-----  
Net "clk_50mhz_i"      LOC=P129;    #系统时钟, 50MHz  
Net "rst_n_i"          LOC=P85;     #复位按键, BUT0  
  
Net "water_led_o<3>"  LOC=P103;  
Net "water_led_o<2>"  LOC=P98;  
Net "water_led_o<1>"  LOC=P105;  
Net "water_led_o<0>"  LOC=P104;
```

综合之后生成 bit 文件烧进试验箱中观察到如下图所示的现象:

rst 为 1, 选择器输出 1110:



rst 为 1, 选择器输出 1101:



rst 为 1, 选择器输出 1011:



rst 为 1，选择器输出 0111:



当按下复位按键时，也就是 rst 为 0 时，现象与仿真结果一致，输出为 1110:



### 实验总结:

通过这次实验的学习,我学会了利用 FPGA 芯片的 50MHz 的时钟来产生自己需要的时钟信号,即掌握了分频的方法。同时,我还学会了编写计数器和选择器的模块,能够将各种模块例化并使用它们,即顶层模块调用底层模块。最终,实现了使用 50MHz 的时钟信号来产生 1Hz 的时钟信号,控制 LED 灯以 1Hz 的频率改变状态。

指导教师批阅意见：

成绩评定：

指导教师签字：

年 月 日

备注：

- 注： 1、报告内的项目或内容设置，可根据实际情况加以调整和补充。  
2、教师批改学生实验报告时间应在学生提交实验报告时间后 10 日内。