

**实验题二：设计一个可循环左/右移，并可
进行并行输入的 8 位移位寄存器。**

姓名：王科

学号：1310583

专业：计算机科学与技术

完成日期：2016/4/9

目录

一、	实验内容	- 2 -
二、	实验原理	- 2 -
1.	设计概述	- 2 -
1.1	输入输出特性表	- 2 -
1.2	设计外部端口	- 2 -
2.	设计内部实现逻辑	- 3 -
3.	设计描述	- 3 -
三、	实验步骤	- 3 -
1.	设计 8 位的循环寄存器的端口、内部 VHDL 逻辑程序	- 3 -
2.	使用图形化方法设计原理图	- 4 -
3.	观察 FPGA 电路板，将输入输出端口进行引脚绑定	- 5 -
4.	电路连接示意图	- 5 -
5.	实验结果	- 6 -
四、	实验原理图和 vhd1 程序	- 10 -
1.	整体原理图（“move8.bdf”）	- 10 -
2.	元部件 1（“control.vhd”）	- 10 -
五、	实验总结	- 12 -
1.	实验遇到的问题	- 12 -
2.	实验感悟	- 12 -

一、 实验内容

本次实验是使用图形设计方法与 VHDL 编程相结合在 FPGA 芯片内构建独立逻辑实验，要求设计一个可循环左/右移，并可进行并行输入的 8 位移位寄存器，用单脉冲作为并行输入或移位的控制脉冲。

通过 VHDL 编程，实现双向移位寄存器，使用两个模式选择 S0、S1 端口接逻辑开关用来选择移位模式。S1、S0 为 1、1 的情况下，CLK 上升沿来时寄存器并行输入；S1、S0 为 0、1 的情况下，CLK 上升沿来时寄存器左移一位；S1、S0 为 1、0 的情况下，CLK 上升沿来时寄存器右移一位。使用单脉冲作为并行输入和移位的控制脉冲。

二、 实验原理

1. 设计概述

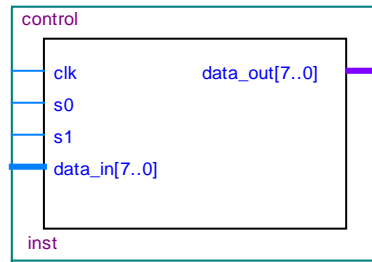
8 位移位寄存器，就是通过两个端口（s0、s1）选择是并行将输入端口的信号到输出端口，还是循环左移或右移一位输出端口的信号。其中，输入输出特性如下图：

1.1 输入输出特性表

INPUT				OUTPUT
clk	S1	S0	Data_in	Data_out
↑	0	0	xxxxxxx	Data_in
↑	0	1	xxxxxxx	Data_out 左移一位
↑	1	0	xxxxxxx	Data_out 右移一位
↑	1	1	xxxxxxx	Data_out

1.2 设计外部端口

按照设计要求，即需要 11 个输入端口和 8 个输入端口，其中输入端口包括一个脉冲信号 clk、两个模式选择的端口 s0、s1，8 个输入端口；输出端口为 8 个输出信号。其整体外部端口如下：



2. 设计内部实现逻辑

按照 74194 的设计逻辑，我使用 VHDL 语言设计了一个 8 位的循环移位寄存器元件，通过判断 s1、s0 端口的信号，若是“11”则直接将输入信号赋给输出端口；若是“01”则将输出端口的信号左移一位再赋给输出端口；若是“10”则将输出端口的信号右移一位再赋给输出端口。

3. 设计描述

使用自底向上设计方法，先使用 VHDL 语言设计一个 8 位的循环寄存器，为其生成元部件，然后设计的主体框架是一个使用 quartus 图形化设计方法设计的一个 8 位移位寄存器，并为其设置引脚并编译生成工程文件。

三、 实验步骤

1. 设计 8 位的循环寄存器的端口、内部 VHDL 逻辑程序

按照实验要求设计一个 8 位的循环寄存器的元部件，包含 11 个输入端口和 8 个输出端口，其中输入端口包括一个脉冲信号 clk、两个模式选择的端口 s0、s1，8 个输入端口；输出端口为 8 个输出信号，vhdl 语言设计端口如下：

```
PORT(
    clk : IN STD_LOGIC;
    s0: IN STD_LOGIC;
    s1: IN STD_LOGIC;

    data_in: IN STD_LOGIC_VECTOR(7 downto 0);
    data_out: BUFFER STD_LOGIC_VECTOR(7 downto 0)
);
```

在其内部实现逻辑功能代码如下：

```
ARCHITECTURE content OF control IS
BEGIN

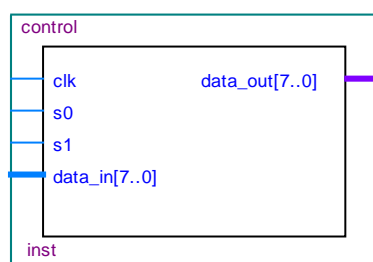
    PROCESS (clk)
    BEGIN

        IF rising_edge (clk) THEN
            -- left move
            IF (s0='0' and s1='1') THEN
                data_out <= data_out(6 downto 0) & data_out(7);
            END IF;

            -- right move
            IF (s0='1' and s1='0') THEN
                data_out <= data_out(0)&data_out(7 downto 1);
            END IF;

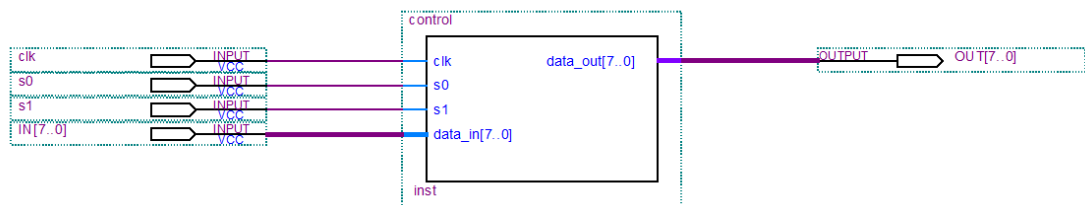
            -- 并行输入
            IF (s0='1' and s1='1') THEN
                data_out <= data_in;
            END IF;
        END IF;
    END PROCESS;
END content;
```

编译完成后为其生成. bdf 元器件符号如下：



2. 使用图形化方法设计原理图

使用 quartus 包含上述设计的元件，设计原理图如下：



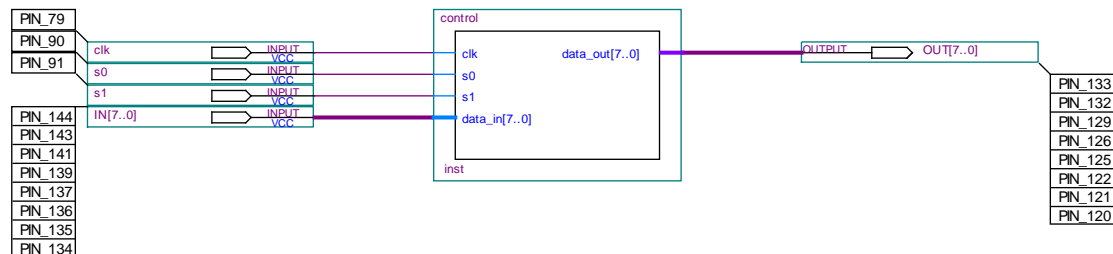
3. 观察 FPGA 电路板，将输入输出端口进行引脚绑定

根据 FPGA 电路板上空闲的剩余引脚，在这次实验中个输入输出端对应的引脚号如下表：

	Node Name	Direction	Location	I/O Bank	VrefGroup	I/O Standard	Reserved	Group	Current Strength
1	clk	Input	PIN_79	3	B3_N1	3.3-V LVTTTL (default)			24mA (default)
2	IN[7]	Input	PIN_144	2	B2_N1	3.3-V LVTTTL (default)		IN[7..0]	24mA (default)
3	IN[6]	Input	PIN_143	2	B2_N1	3.3-V LVTTTL (default)		IN[7..0]	24mA (default)
4	IN[5]	Input	PIN_141	2	B2_N1	3.3-V LVTTTL (default)		IN[7..0]	24mA (default)
5	IN[4]	Input	PIN_139	2	B2_N1	3.3-V LVTTTL (default)		IN[7..0]	24mA (default)
6	IN[3]	Input	PIN_137	2	B2_N1	3.3-V LVTTTL (default)		IN[7..0]	24mA (default)
7	IN[2]	Input	PIN_136	2	B2_N1	3.3-V LVTTTL (default)		IN[7..0]	24mA (default)
8	IN[1]	Input	PIN_135	2	B2_N1	3.3-V LVTTTL (default)		IN[7..0]	24mA (default)
9	IN[0]	Input	PIN_134	2	B2_N1	3.3-V LVTTTL (default)		IN[7..0]	24mA (default)
10	OUT[7]	Output	PIN_133	2	B2_N1	3.3-V LVTTTL (default)		OUT[7..0]	24mA (default)
11	OUT[6]	Output	PIN_132	2	B2_N1	3.3-V LVTTTL (default)		OUT[7..0]	24mA (default)
12	OUT[5]	Output	PIN_129	2	B2_N1	3.3-V LVTTTL (default)		OUT[7..0]	24mA (default)
13	OUT[4]	Output	PIN_126	2	B2_N0	3.3-V LVTTTL (default)		OUT[7..0]	24mA (default)
14	OUT[3]	Output	PIN_125	2	B2_N0	3.3-V LVTTTL (default)		OUT[7..0]	24mA (default)
15	OUT[2]	Output	PIN_122	2	B2_N0	3.3-V LVTTTL (default)		OUT[7..0]	24mA (default)
16	OUT[1]	Output	PIN_121	2	B2_N0	3.3-V LVTTTL (default)		OUT[7..0]	24mA (default)
17	OUT[0]	Output	PIN_120	2	B2_N0	3.3-V LVTTTL (default)		OUT[7..0]	24mA (default)
18	s0	Input	PIN_90	3	B3_N0	3.3-V LVTTTL (default)			24mA (default)
19	s1	Input	PIN_91	3	B3_N0	3.3-V LVTTTL (default)			24mA (default)

编译下载到 FPGA 电路板上，然后连接实验箱的电路。

编译后如下：



4. 电路连接示意图

连接 FPGA 电路板上电路：

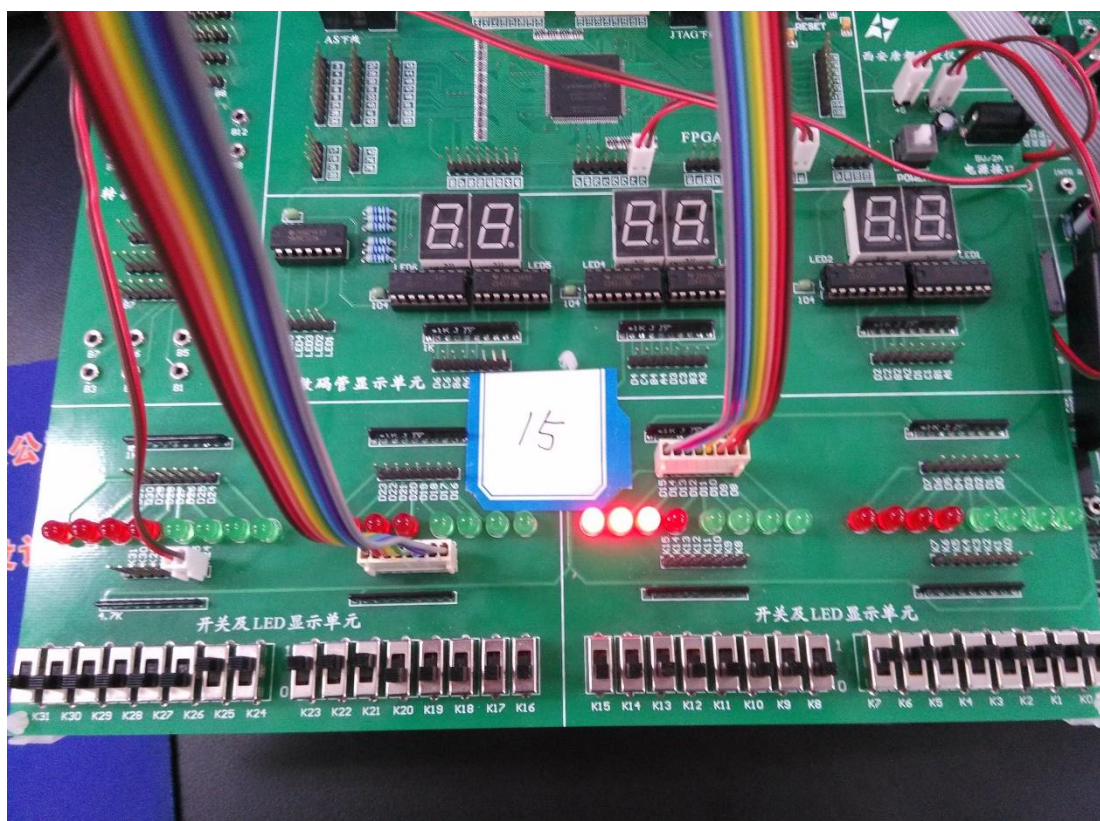
- 将 FPGA 板上的 PIN_79 接到实验箱上的脉冲信号开关 kk0 上；
- 将 FPGA 板上的 PIN_90、PIN_91 接到控制开关 k8、k9 上；
- 将 FPGA 板上的 PIN_144~PIN_134 接到控制开关 k7~k0 上；
- 将 FPGA 板上的 PIN_133~PIN_120 接到 LED 灯泡 L7~L0 上；

对应端口连接表			
说明	引脚	说明	引脚

FPGA 板上的脉冲信号 clk	PIN_79	实验箱上的脉冲信号开关	KK0
FPGA 板上的模式选择信号 s0、s1	PIN_90、PIN_91	FPGA 电路板控制开关	K8~K9
FPGA 板的输入端 (IN7~IN0)	PIN_144~PIN_134	FPGA 电路板控制开关	K7~K0
FPGA 板的输出端 (OUT7~OUT0)	PIN_133~PIN_120	FPGA 电路板的 LED 灯泡	L7~L0

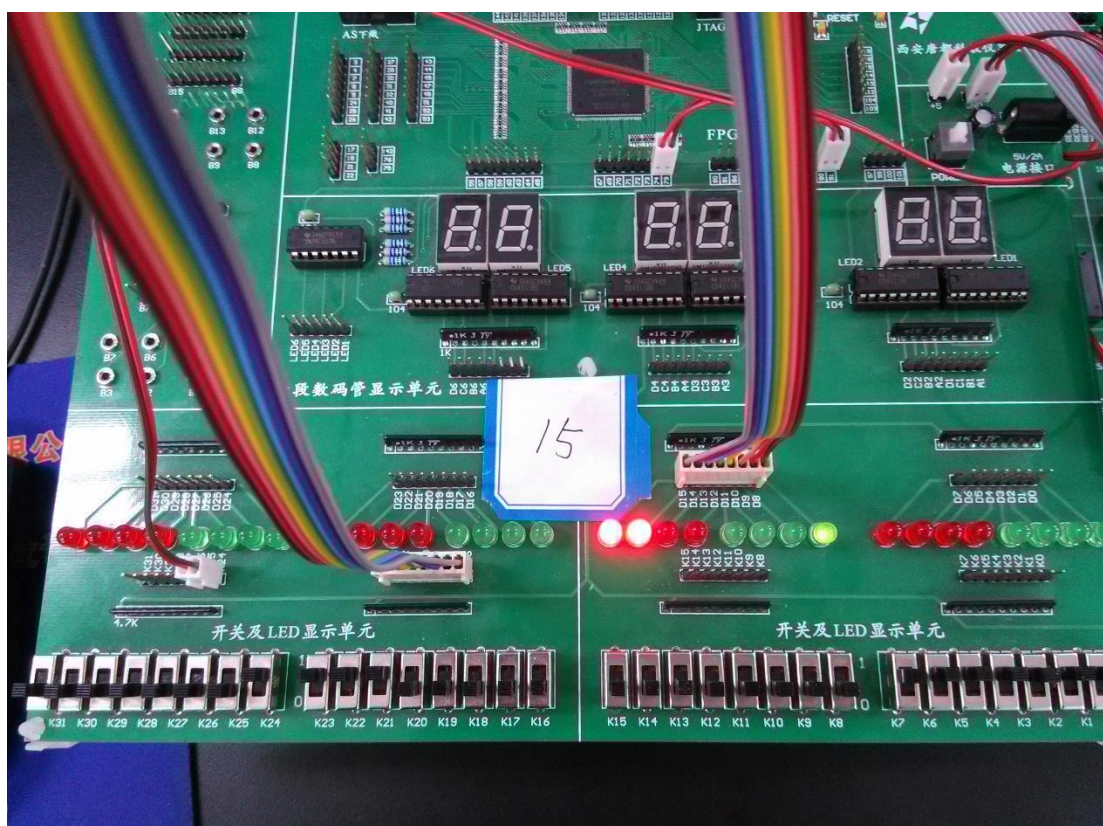
5. 实验结果

实验时将 s1、s0 值成“11”（并行输入），输入信号是“11100000”时，输出如下：

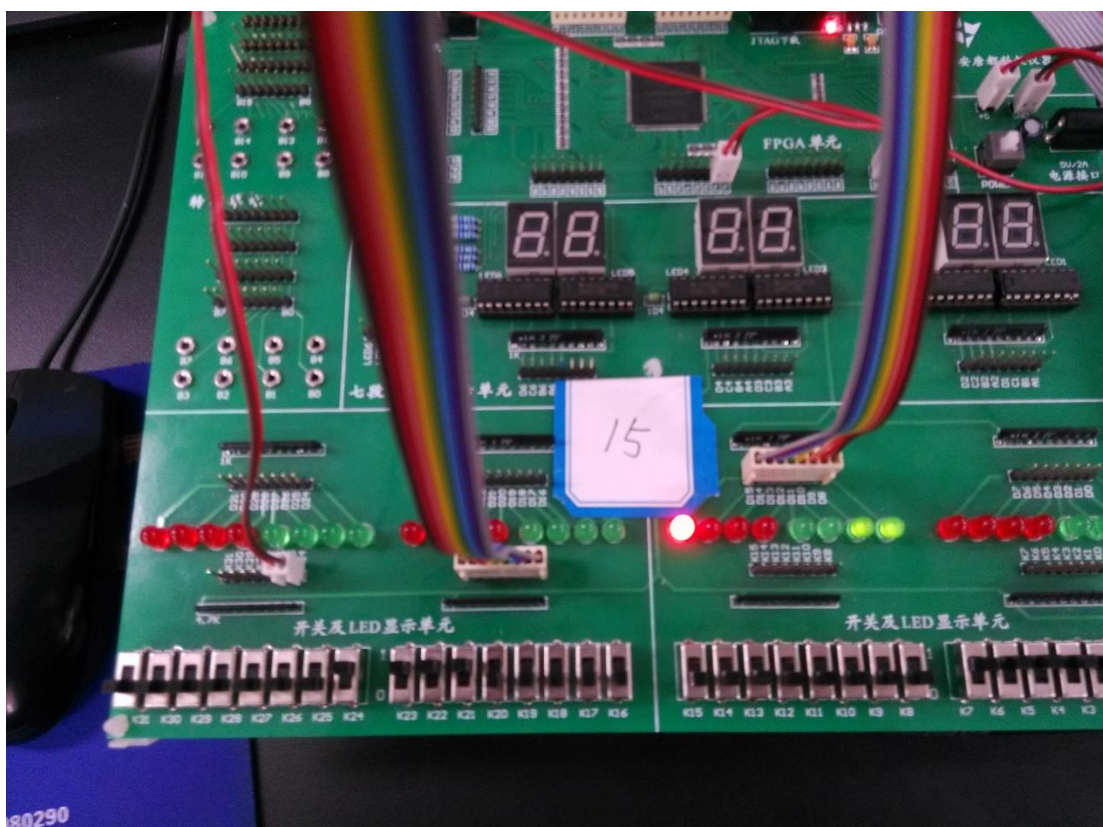


可以看到输出结果也是“11100000”，验证了并行输出。

然后将 s1、s0 值成“01”（左移），按下脉冲开关一次，其结果如下图：

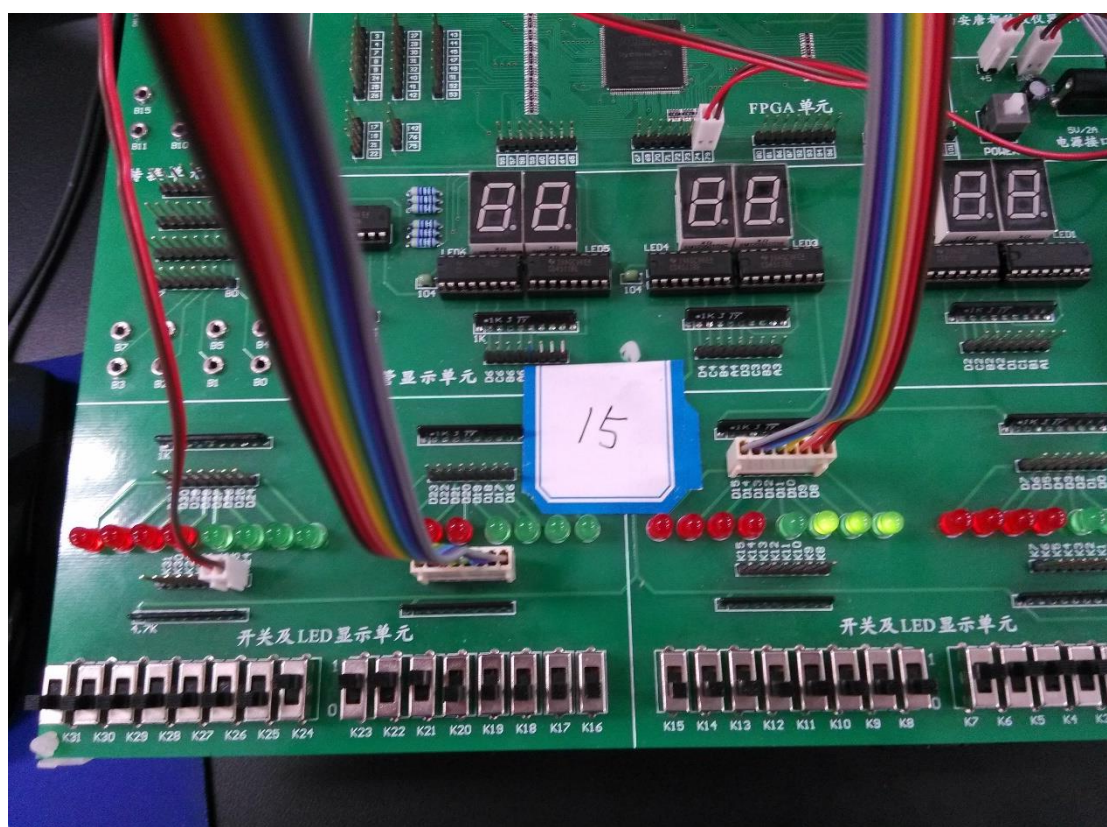


可以看到输出是“11000001”，即原来的数循环左移了一位。再次按下脉冲开关一次，其结果如下：

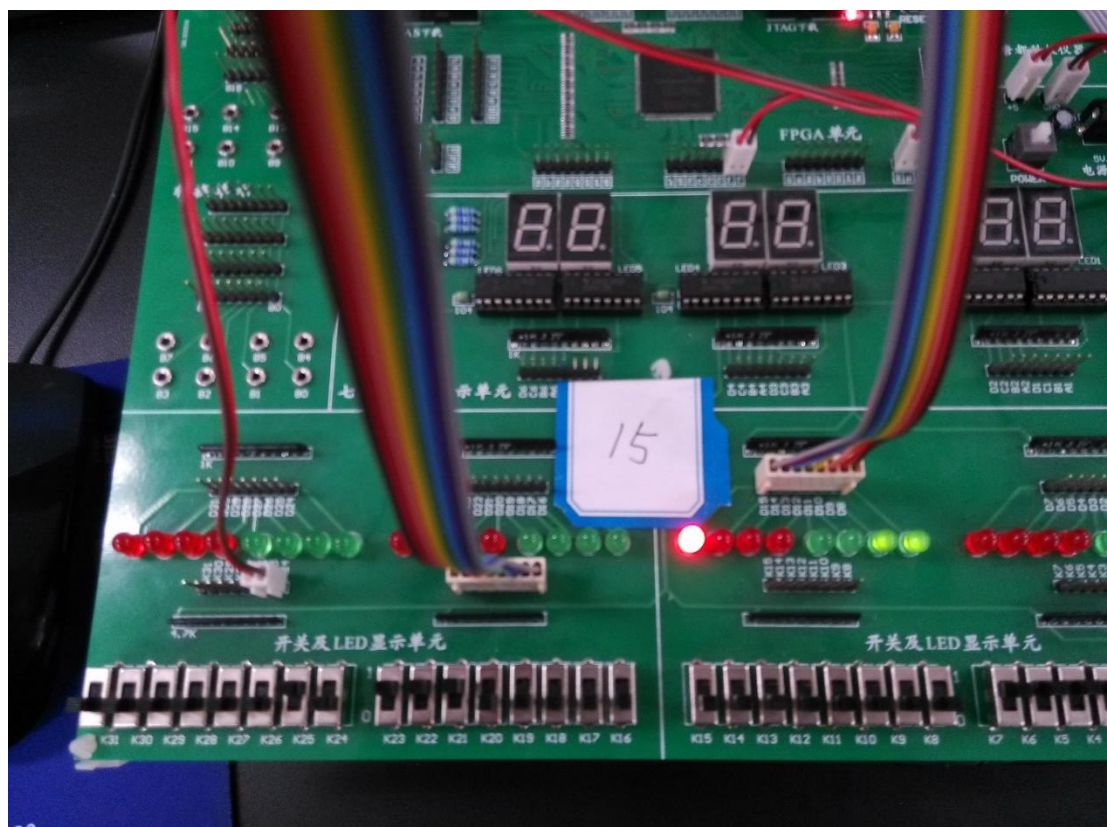


输出结果是“10000011”，可以看到再次左移了一位。再次按一次脉冲开关，左

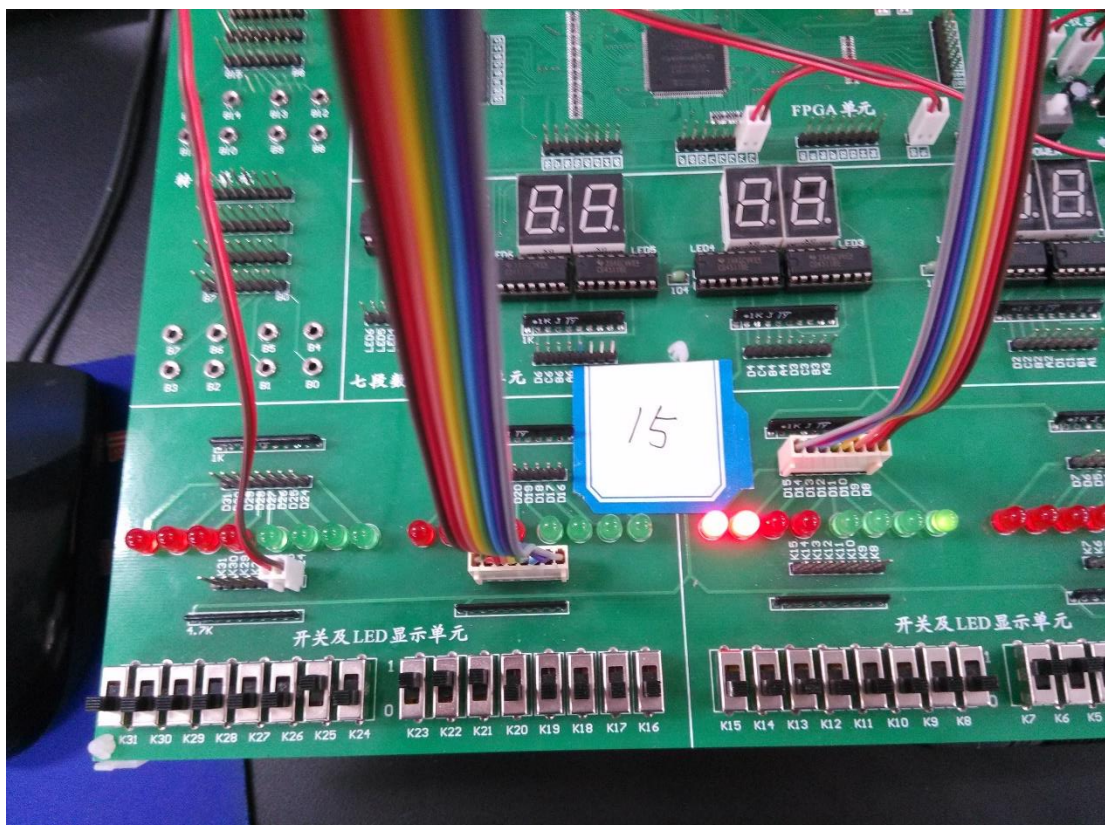
移一位如下：



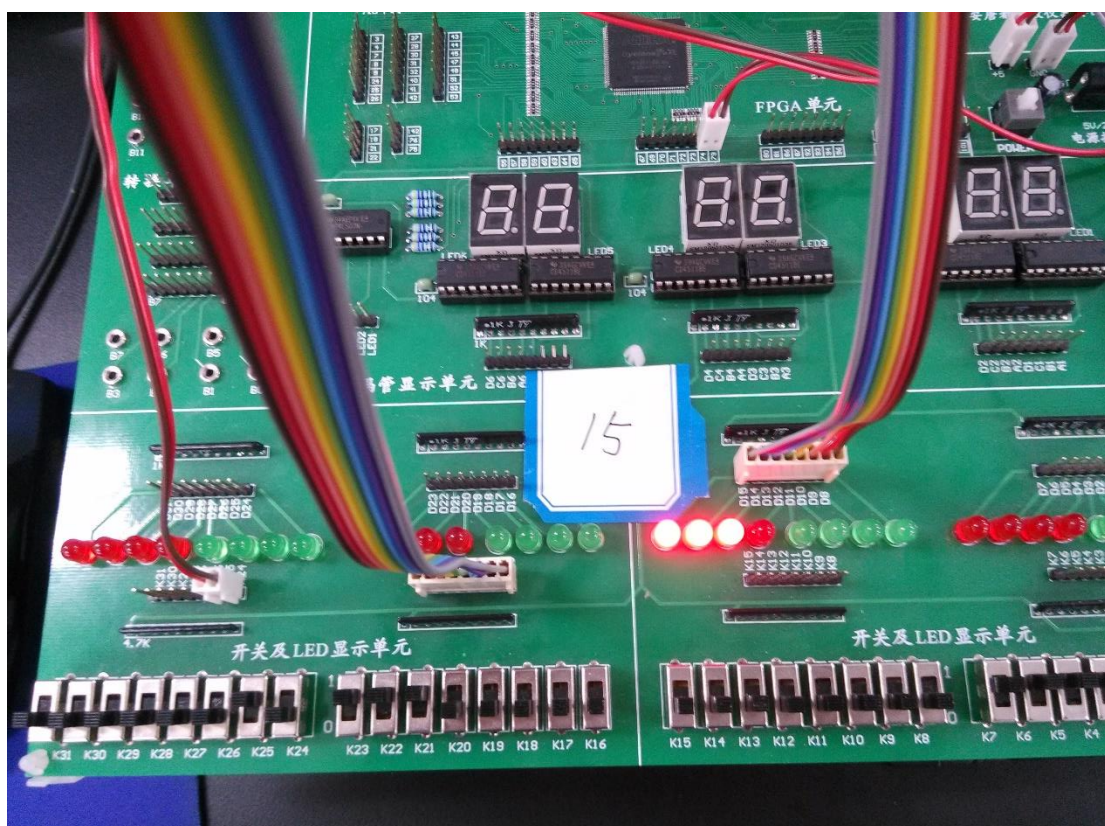
这时将 s1、s0 值成“10”（友移），按下脉冲开关一次，其结果如下图：



观察结果可知其右移了一位，再次按下脉冲开关，右移一位如下：

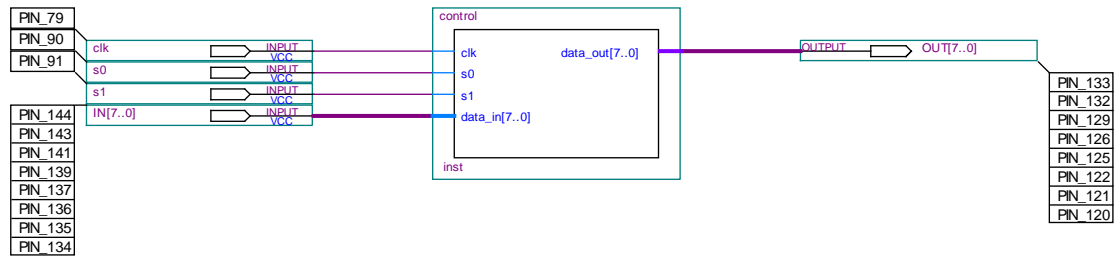


再次按下脉冲开关，右移一位如下：



四、 实验原理图和 vhd1 程序

1. 整体原理图 (“move8.bdf”)



2. 元部件 1 (“control.vhd”)

```
Library ieee;
Use ieee.std_logic_1164.all;
Use ieee.std_logic_unsigned.all;
Use ieee.std_logic_arith.all;

ENTITY control IS
PORT(
    clk : IN STD_LOGIC;
    s0: IN STD_LOGIC;
    s1: IN STD_LOGIC;

    data_in: IN STD_LOGIC_VECTOR(7 downto 0);
    data_out: BUFFER STD_LOGIC_VECTOR(7 downto 0)
);
END control;

ARCHITECTURE content OF control IS

BEGIN

    PROCESS (clk)
    BEGIN

        IF rising_edge (clk) THEN
            -- left move
            IF (s0='0' and s1='1') THEN
                data_out <= data_out(6 downto 0) & data_out(7);
            END IF;
        
```

```
-- right move
IF (s0='1' and s1='0') THEN
    data_out <= data_out(0)&data_out(7 downto 1);
END IF;

-- 并行输入
IF (s0='1' and s1='1') THEN
    data_out <= data_in;
END IF;
END IF;

END PROCESS;

END content;
```

五、 实验总结

1. 实验遇到的问题

这次使用图形设计与 VHDL 程序设计相结合的方式设计电路，逻辑上相对于上一个实验来说较复杂，主要在于设计左移和右移的逻辑功能上遇到了困难。主要遇到的问题及解决方法如下：

- 在实现左移和右移的功能时为了在程序内部使用输出端的值，不能将输出端定义成 OUT LOGIC，而是 BUFFER LOGIC，只有这样才能使用输出端口的值。
- 在做实验的过程中，一开始 LED 灯泡怎么都不亮，我反反复复检查了电路和程序也没有发现任何问题，最后发现是 FPGA 电路板不稳定的原因，更换 FPGA 电路板过后，就正常的显示出来了，在这之前我浪费了大量的时间，终于明白以后一定要对自己有自信，大胆怀疑是不是硬件出现了问题。

2. 实验感悟

这次使用图形设计和 VHDL 语言相结合的方式设计一个 8 位的循环移位寄存器使我对于这种自底向上设计电路的方法更加的熟练了一些，在遇到问题的时候不像第一次那样一无所知，从头查起，这次很顺利的知道了问题所在。总之，对于以后的设计电路打下了良好的基础。

最后感谢老师们不厌其烦的为我解答疑惑并帮助我调试 bug，我才能顺利的完成了这次实验！