Tuesday Lab 3:30-6:15 pm

Lab 6

ECE-238L-003

NIKKI BROWN and Florentino Barrientos

2018

Grey Code **Design Code**

----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 03/16/2018 02:19:26 PM

-- Design Name:

-- Module Name: grayCode - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity grayCode is

Port ( Clock\_System : in STD\_LOGIC;

Reset : in STD\_LOGIC;

outputCount : out STD\_LOGIC\_VECTOR (1 downto 0));

end grayCode;

architecture Behavioral of grayCode is

component Frequency\_Divider is

Port ( Clock\_System: in STD\_LOGIC;

Clock\_1Hz: out STD\_LOGIC);

end component;

signal presentState : STD\_LOGIC\_VECTOR (1 downto 0) := "00";

signal nextState : STD\_LOGIC\_VECTOR (1 downto 0);

signal sigClock\_1Hz : STD\_LOGIC := '0';

begin

process (sigClock\_1Hz, reset)

begin

if reset = '1' then

presentState <= "00";

elsif rising\_edge(sigClock\_1Hz) then

presentState <= nextState;

end if;

end process;

outputCount <= presentState;

nextState <= "01" when presentState = "00" else

"11" when presentState = "01" else

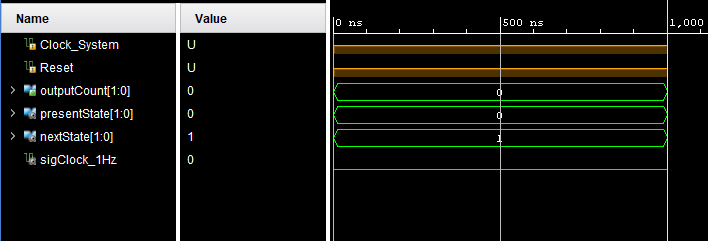
"10" when presentState = "11" else

"00";

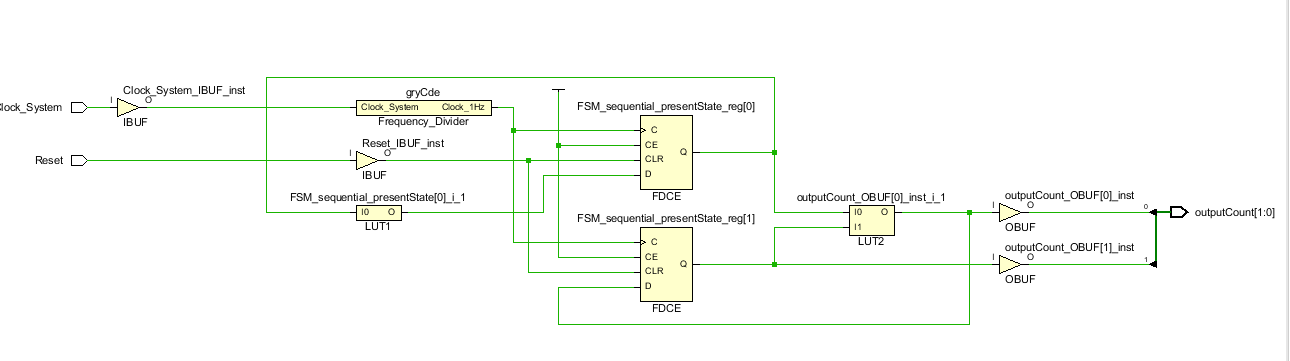
gryCde: Frequency\_Divider port map(Clock\_System => Clock\_System, Clock\_1HZ => sigClock\_1Hz);

end Behavioral;

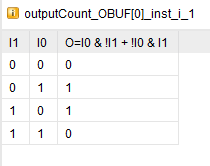
Grey Code **Waveform**



Grey Code Schematic



Grey Code Truth Tables



Grey Code Constraints

set\_property IOSTANDARD LVCMOS33 [get\_ports {outputCount[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {outputCount[0]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports Clock\_System]

set\_property IOSTANDARD LVCMOS33 [get\_ports Reset]

set\_property PACKAGE\_PIN E3 [get\_ports Clock\_System]

set\_property PACKAGE\_PIN J15 [get\_ports Reset]

set\_property PACKAGE\_PIN H17 [get\_ports {outputCount[0]}]

set\_property PACKAGE\_PIN K15 [get\_ports {outputCount[1]}]**Robot Eye Design Code**

----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 03/19/2018 03:52:46 PM

-- Design Name:

-- Module Name: RoboEye - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

-- Uncomment the following library declaration if using

-- arithmetic functions with Signed or Unsigned values

--use IEEE.NUMERIC\_STD.ALL;

-- Uncomment the following library declaration if instantiating

-- any Xilinx leaf cells in this code.

--library UNISIM;

--use UNISIM.VComponents.all;

entity RoboEye is

Port ( Clock\_System\_RoboEye : in STD\_LOGIC;

Reset : in STD\_LOGIC;

Cathode\_7SD : out STD\_LOGIC\_VECTOR (7 downto 0);

Anode\_7SD : out STD\_LOGIC\_VECTOR (7 downto 0));

end RoboEye;

architecture Behavioral of RoboEye is

component Frequency\_Divider is

Port ( Clock\_System: in STD\_LOGIC;

Clock\_1Hz: out STD\_LOGIC);

end component;

signal presentStateRobo : STD\_LOGIC\_VECTOR (4 downto 0) := "00001";

signal nextStateRobo : STD\_LOGIC\_VECTOR (4 downto 0);

signal slowClock\_1Hz : STD\_LOGIC := '0';

begin

process (Clock\_System\_RoboEye, reset)

begin

if reset = '1' then

presentStateRobo <= "00001";

elsif rising\_edge(slowClock\_1Hz) then

presentStateRobo <= nextStateRobo;

end if;

end process;

Cathode\_7SD <= "10111111";

Anode\_7SD <= "1111" & NOT presentStateRobo (3 downto 0);

nextStateRobo <= "00010" when presentStateRobo = "00001" else

"00100" when presentStateRobo = "00010" else

"01000" when presentStateRobo = "00100" else

"10100" when presentStateRobo = "01000" else

"10010" when presentStateRobo = "10100" else

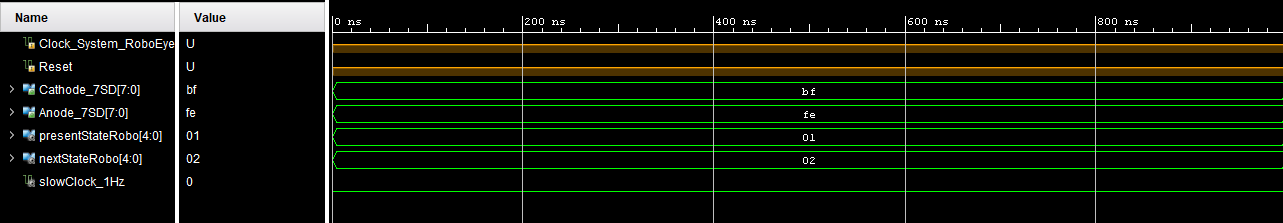
"00001";

--Anode\_7SD <= "111" & presentStateRobo;

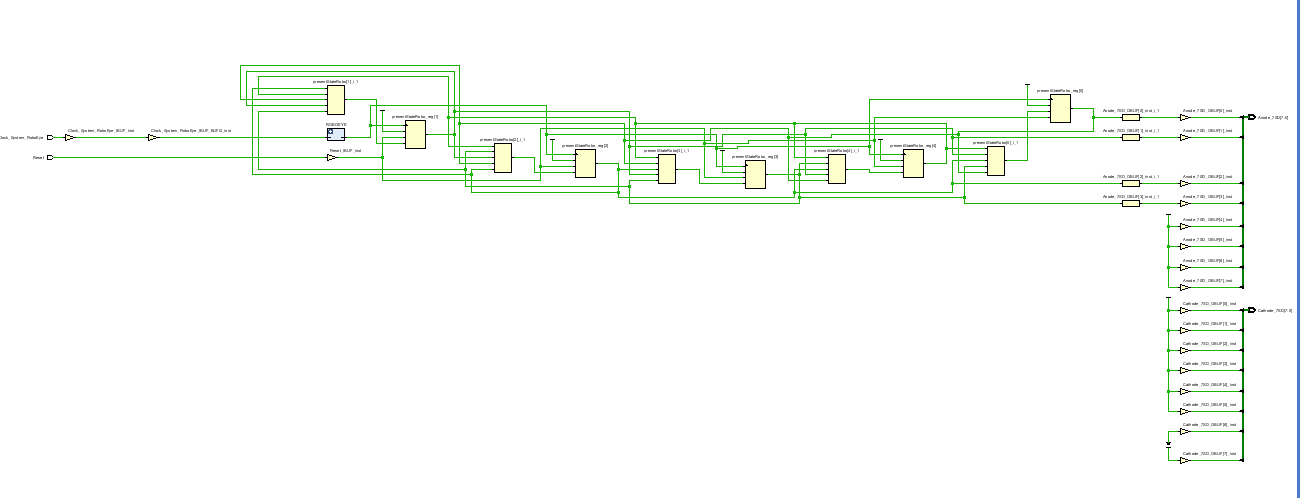
ROBOEYE: Frequency\_Divider port map(Clock\_System => Clock\_System\_RoboEye, Clock\_1HZ => slowClock\_1Hz);

end Behavioral;

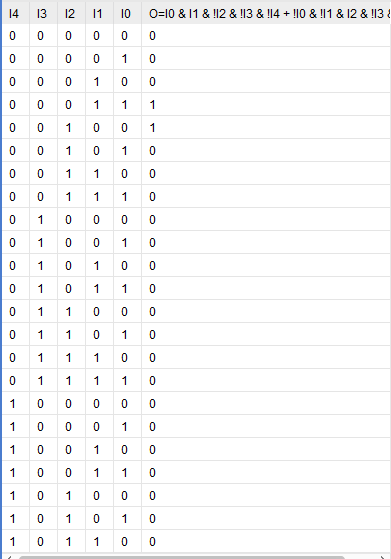
**Robot Eye Waveform**



Robot Eye Schematic



Robot Eye Truth Tables



**Robot Eye Constraints**

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[7]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[6]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[5]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[4]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[3]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[2]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[0]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[7]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[6]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[5]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[4]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[3]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[2]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[0]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports Clock\_System\_RoboEye]

set\_property IOSTANDARD LVCMOS33 [get\_ports Reset]

set\_property PACKAGE\_PIN T10 [get\_ports {Cathode\_7SD[0]}]

set\_property PACKAGE\_PIN R10 [get\_ports {Cathode\_7SD[1]}]

set\_property PACKAGE\_PIN K16 [get\_ports {Cathode\_7SD[2]}]

set\_property PACKAGE\_PIN K13 [get\_ports {Cathode\_7SD[3]}]

set\_property PACKAGE\_PIN P15 [get\_ports {Cathode\_7SD[4]}]

set\_property PACKAGE\_PIN T11 [get\_ports {Cathode\_7SD[5]}]

set\_property PACKAGE\_PIN L18 [get\_ports {Cathode\_7SD[6]}]

set\_property PACKAGE\_PIN H15 [get\_ports {Cathode\_7SD[7]}]

set\_property PACKAGE\_PIN U13 [get\_ports {Anode\_7SD[7]}]

set\_property PACKAGE\_PIN K2 [get\_ports {Anode\_7SD[6]}]

set\_property PACKAGE\_PIN T14 [get\_ports {Anode\_7SD[5]}]

set\_property PACKAGE\_PIN P14 [get\_ports {Anode\_7SD[4]}]

set\_property PACKAGE\_PIN J14 [get\_ports {Anode\_7SD[3]}]

set\_property PACKAGE\_PIN T9 [get\_ports {Anode\_7SD[2]}]

set\_property PACKAGE\_PIN J18 [get\_ports {Anode\_7SD[1]}]

set\_property PACKAGE\_PIN J17 [get\_ports {Anode\_7SD[0]}]

set\_property PACKAGE\_PIN J15 [get\_ports Reset]

set\_property PACKAGE\_PIN E3 [get\_ports Clock\_System\_RoboEye]**Smiley Face Design Code**

----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 03/20/2018 02:00:59 PM

-- Design Name:

-- Module Name: SmilyFace - Behavioral

-- Project Name:

-- Target Devices:

-- Tool Versions:

-- Description:

--

-- Dependencies:

--

-- Revision:

-- Revision 0.01 - File Created

-- Additional Comments:

--

----------------------------------------------------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

entity SmilyFace is

Port ( Clock\_System\_Smile : in STD\_LOGIC;

resetSmile : in STD\_LOGIC;

Cathode\_7SD : out STD\_LOGIC\_VECTOR (7 downto 0);

Anode\_7SD : out STD\_LOGIC\_VECTOR (7 downto 0));

end SmilyFace;

architecture Behavioral of SmilyFace is

component Frequency\_Divider is

Port ( Clock\_System: in STD\_LOGIC;

Clock\_1Hz: out STD\_LOGIC);

end component;

signal presentStateSmile : STD\_LOGIC\_VECTOR (5 downto 0) := "000000";

signal nextStateSmile : STD\_LOGIC\_VECTOR (5 downto 0);

signal slowClock\_1Hz : STD\_LOGIC := '0';

begin

process (Clock\_System\_Smile, resetSmile)

begin

if resetSmile = '1' then

presentStateSmile <= NOT "000000";

elsif rising\_edge(slowClock\_1Hz) then

presentStateSmile <= nextStateSmile;

end if;

end process;

Cathode\_7SD <= "11111101";

Anode\_7SD <= "11" & presentStateSmile(5 downto 0);

nextStateSmile <= NOT "000110" when presentStateSmile = NOT "000000" else

NOT "001111" when presentStateSmile = NOT "000110" else

NOT "000110" when presentStateSmile = NOT "001111" else

NOT "000110" when presentStateSmile = NOT "000110" else

NOT "001111" when presentStateSmile = NOT "000110" else

NOT "001111" when presentStateSmile = NOT "001111" else

NOT "001111" when presentStateSmile = NOT "001111" else

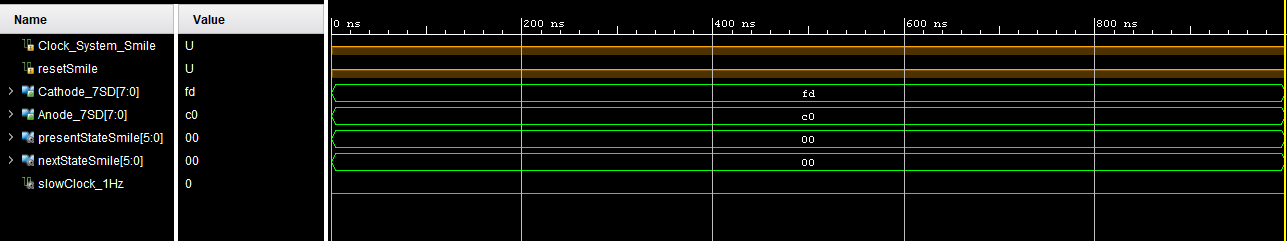
NOT "111111";

--Anode\_7SD <= "11" & presentStateSmile;

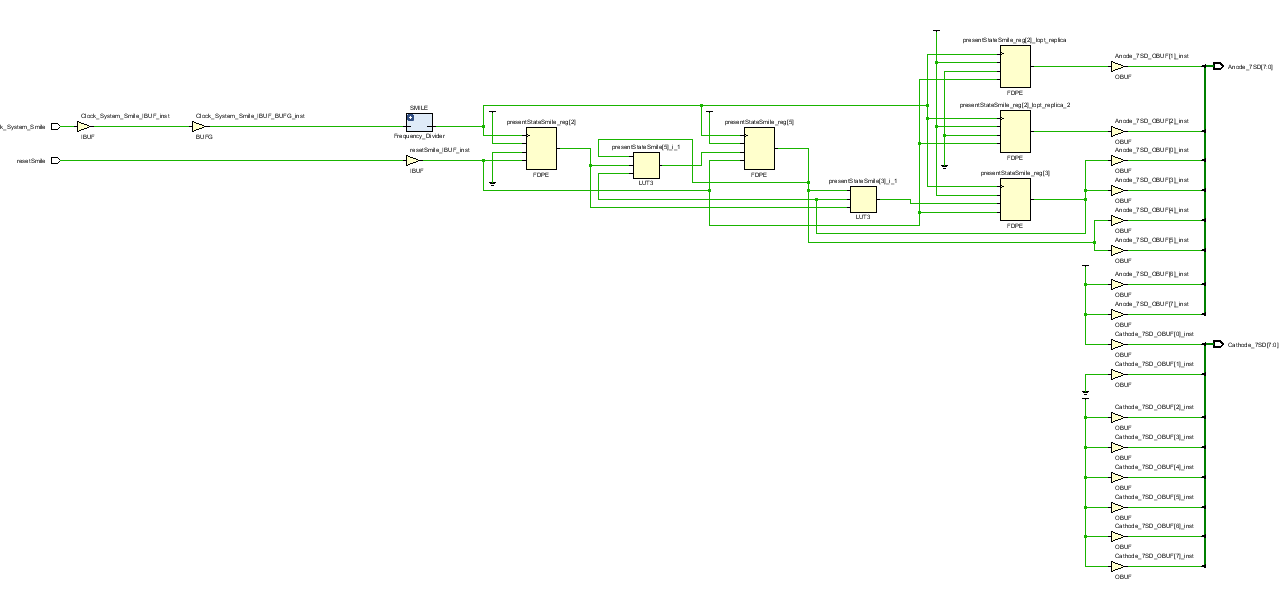
SMILE: Frequency\_Divider port map(Clock\_System => Clock\_System\_Smile, Clock\_1HZ => slowClock\_1Hz);

end Behavioral;

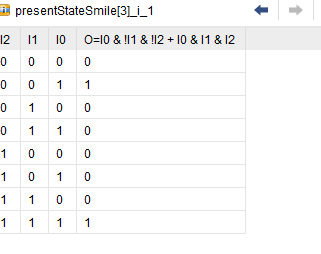
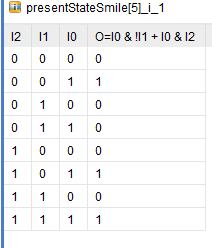
**Smiley Face Waveform**



Smiley Face Schematic



Smiley Face Truth Tables



**Smiley Face Constraints**

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[7]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[6]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[5]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[4]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[3]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[2]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Anode\_7SD[0]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[7]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[6]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[5]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[4]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[3]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[2]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[1]}]

set\_property IOSTANDARD LVCMOS33 [get\_ports {Cathode\_7SD[0]}]

set\_property PACKAGE\_PIN E3 [get\_ports Clock\_System\_Smile]

set\_property IOSTANDARD LVCMOS33 [get\_ports Clock\_System\_Smile]

set\_property PACKAGE\_PIN J15 [get\_ports resetSmile]

set\_property IOSTANDARD LVCMOS33 [get\_ports resetSmile]

set\_property PACKAGE\_PIN U13 [get\_ports {Anode\_7SD[7]}]

set\_property PACKAGE\_PIN K2 [get\_ports {Anode\_7SD[6]}]

set\_property PACKAGE\_PIN T14 [get\_ports {Anode\_7SD[5]}]

set\_property PACKAGE\_PIN P14 [get\_ports {Anode\_7SD[4]}]

set\_property PACKAGE\_PIN J14 [get\_ports {Anode\_7SD[3]}]

set\_property PACKAGE\_PIN T9 [get\_ports {Anode\_7SD[2]}]

set\_property PACKAGE\_PIN J18 [get\_ports {Anode\_7SD[1]}]

set\_property PACKAGE\_PIN J17 [get\_ports {Anode\_7SD[0]}]

set\_property PACKAGE\_PIN T10 [get\_ports {Cathode\_7SD[7]}]

set\_property PACKAGE\_PIN R10 [get\_ports {Cathode\_7SD[6]}]

set\_property PACKAGE\_PIN K16 [get\_ports {Cathode\_7SD[5]}]

set\_property PACKAGE\_PIN K13 [get\_ports {Cathode\_7SD[4]}]

set\_property PACKAGE\_PIN P15 [get\_ports {Cathode\_7SD[3]}]

set\_property PACKAGE\_PIN T11 [get\_ports {Cathode\_7SD[2]}]

set\_property PACKAGE\_PIN L18 [get\_ports {Cathode\_7SD[1]}]

set\_property PACKAGE\_PIN H15 [get\_ports {Cathode\_7SD[0]}]

Summary

The lab scheduled for this week aimed to further improve our skills in module building from components while also introducing us to state machines. The first state machine we built was the grey code which uses a counter to count in grey code. This result was displayed into the LEDS on the Artix 7 board where each LED represented a grey code value. Using similar logic used in the grey code state machine, the following lab prompted us to create a “robot eye” where a LED would shift between one of the Cathode displays. This processed differ by manipulating the last bits of the selected anodes in order to shift the LED from left to right. Switch cases were used to simulate this event and a reset input was used to restart the process. The last state machine that was assigned was to create a “smiley face display” where eyes would first appear in the cathode display then its mouth and later a blank screen. Like the Robot eye, we manipulated the last bits of the given anodes to simulate this effect. A reset switch was also implemented to reset the process. These programs ran against a timer where the usage of duplicate cases was used for the events to be witnessed in real time easily.

Problems encountered dealt with the bit manipulation and the usage of active low displays. It is easy to mess up the display if one is careless about what they are inputting. One must also keep caution in carefully analyzing the switch cases so that the process can run like it should. Also one should be wary of the base case of the switch statement so the process can reset correctly also.