Gentry, Dane
Gorsuch, Byron

CpE 200L Final Project Report – Slot Machine

Goal – Our goal is to implement a slot machine game on the DE2 board through our Quartus circuit schematic which consists of three random number generators, each with a clock, as well as a comparator. Numerous LED's should light up as a result of a win if all three of the random numbers generated are equivalent, which is determined by the comparator.

Roles of each group member taken in the project:

Byron Gorsuch:

Brainstorm of potential projects
Quartus circuit schematic design
Generation of simulation waveform
Obtained DE2 board for implementation
Final project report
Final powerpoint presentation

Dane Gentry:

Brainstorm of potential projects
Obtained DE2 board for implementation
Pin assignments
Implementation of Quartus circuit schematic onto DE2 board
Final project report
Final powerpoint presentation

Background Theory:

Random Number - A random number is a number taken from a set of numbers, whose value could not have been previously predicted.

Slot Machine - Slot machine is a common gambling machine found in casinos. They are generally composed of three or more reels with different values or images on them which appear to spin and stop at a random moment, and if the values match or some other win criteria is met, the machine will award some money. In actuality, the random values are chosen when the button is pressed, and the machine knows the outcome before the fanfare is over. The sounds, lights, and spinning reels of the machine are meant to entice the user to stay and play.

Counter Based Number Generator (CBNG) - A CBNG circuit consists of a 4 bit counter with a 50 MHz CLK input which acts as the random number generator. Upon activation of the 50 MHz clock input, the counter rapidly races through all possible state transitions from 0000 to 1111 all within a 320 ns time period. The counter counts up, reaches the last state (S15), reverts to (S0), starts over again, and continues to repeat indefinitely until the CLK signal is removed. A 4-bit D-type FF is used to load and latch a random state of the counter at any given time via the D-FF CLK input controlled by an external catch push button. The value stored in the latch is then output to an indication circuit, in this case a binary set of LED's.

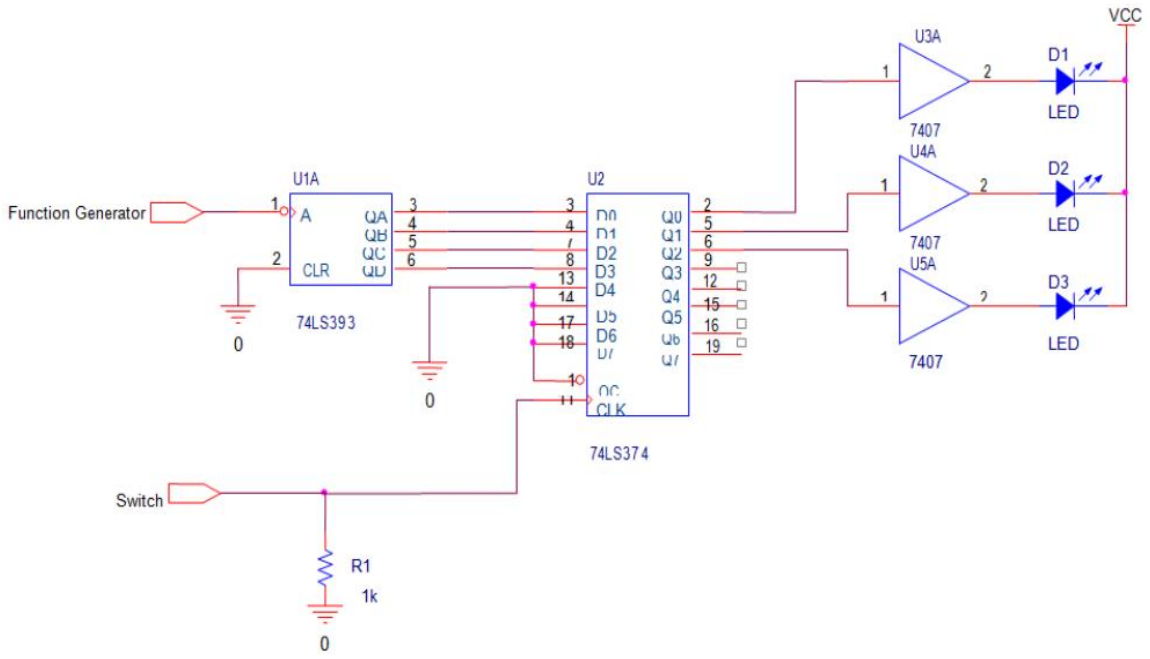


Figure 1. Simple random number generator

Clocks - DE2 board features hardware including a 50 MHz as well as 27 MHz oscillator hardware for clock sources, or in other words, the DE2 board includes two oscillators that produce 27 MHz and 50 MHz clock signals. In addition, the DE2 board also includes a SMA connector which can be used to connect an external clock source to the board. The schematic of the clock circuitry is shown in Figure 4.8, and the associated pin assignments appear in Table 4.5.

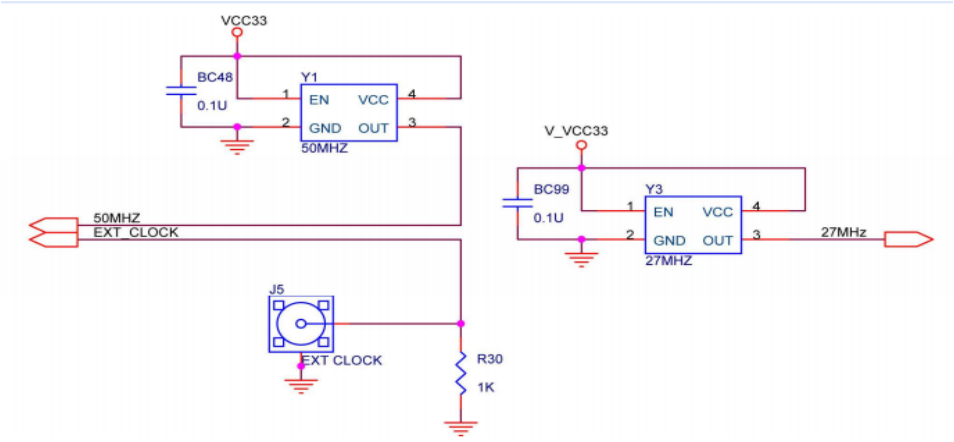


Figure 4.8. Schematic diagram of the clock circuit.

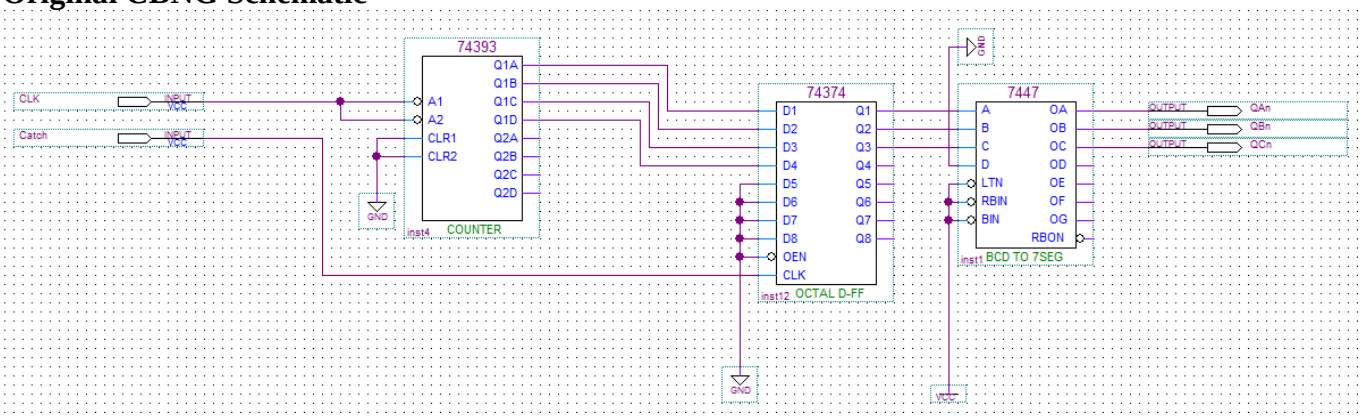
Signal Name	FPGA Pin No.	Description
CLOCK_27	PIN_D13	27 MHz clock input
CLOCK_50	PIN_N2	50 MHz clock input
EXT_CLOCK	PIN_P26	External (SMA) clock input

Table 4.5. Pin assignments for the clock inputs.

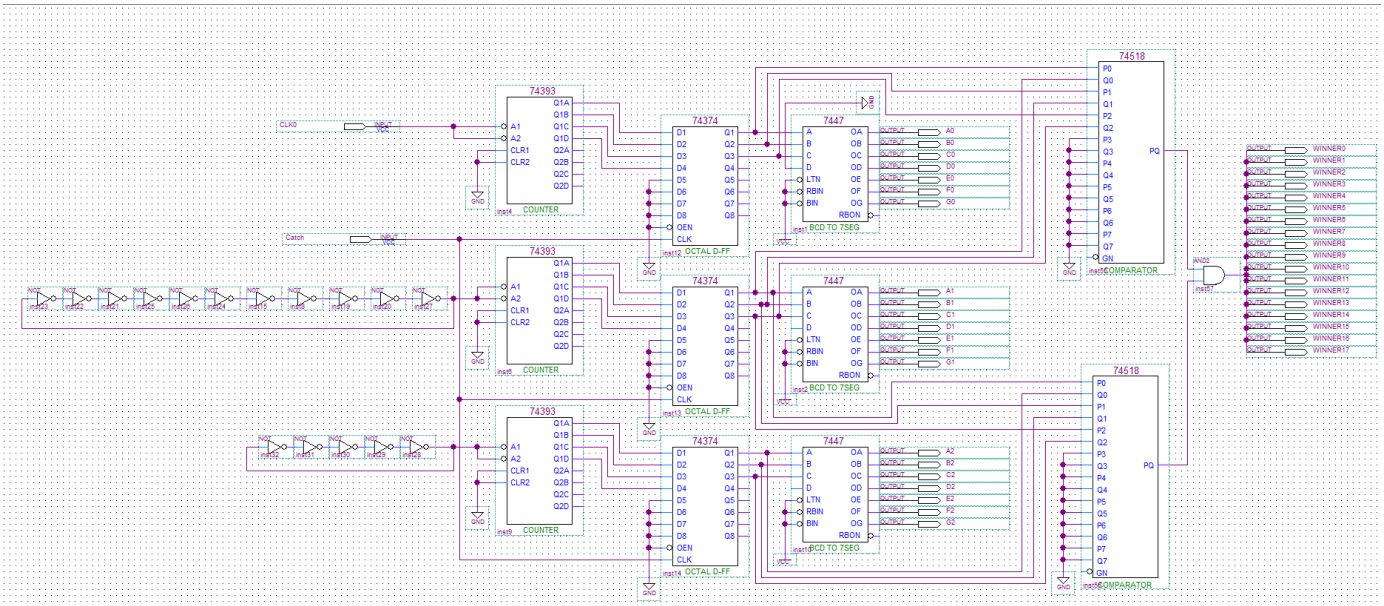
Gentry, Dane
Gorsuch, Byron

Quartus Circuit Schematics

Original CBNG Schematic

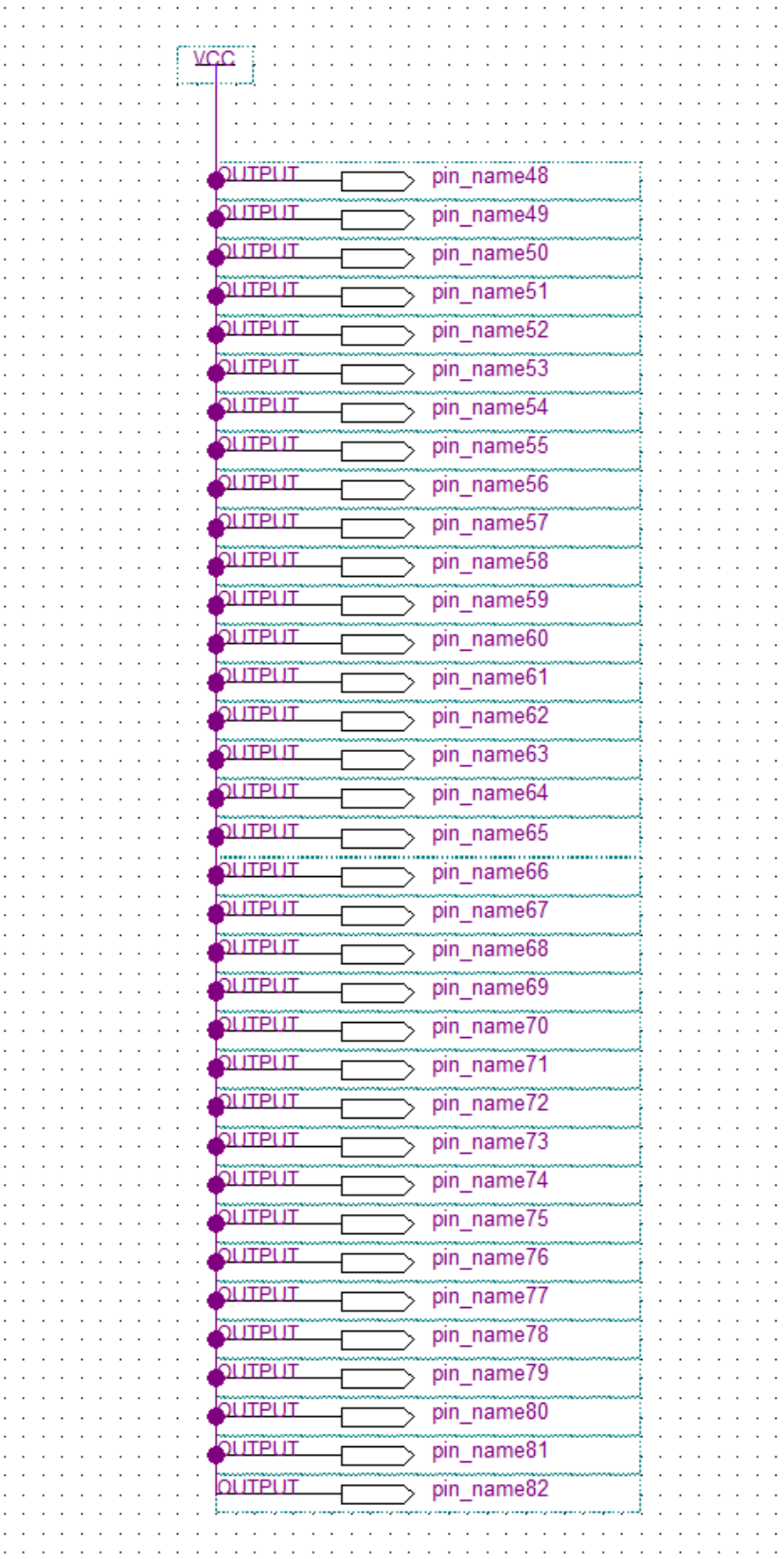


Final Schematic



Gentry, Dane
Gorsuch, Byron

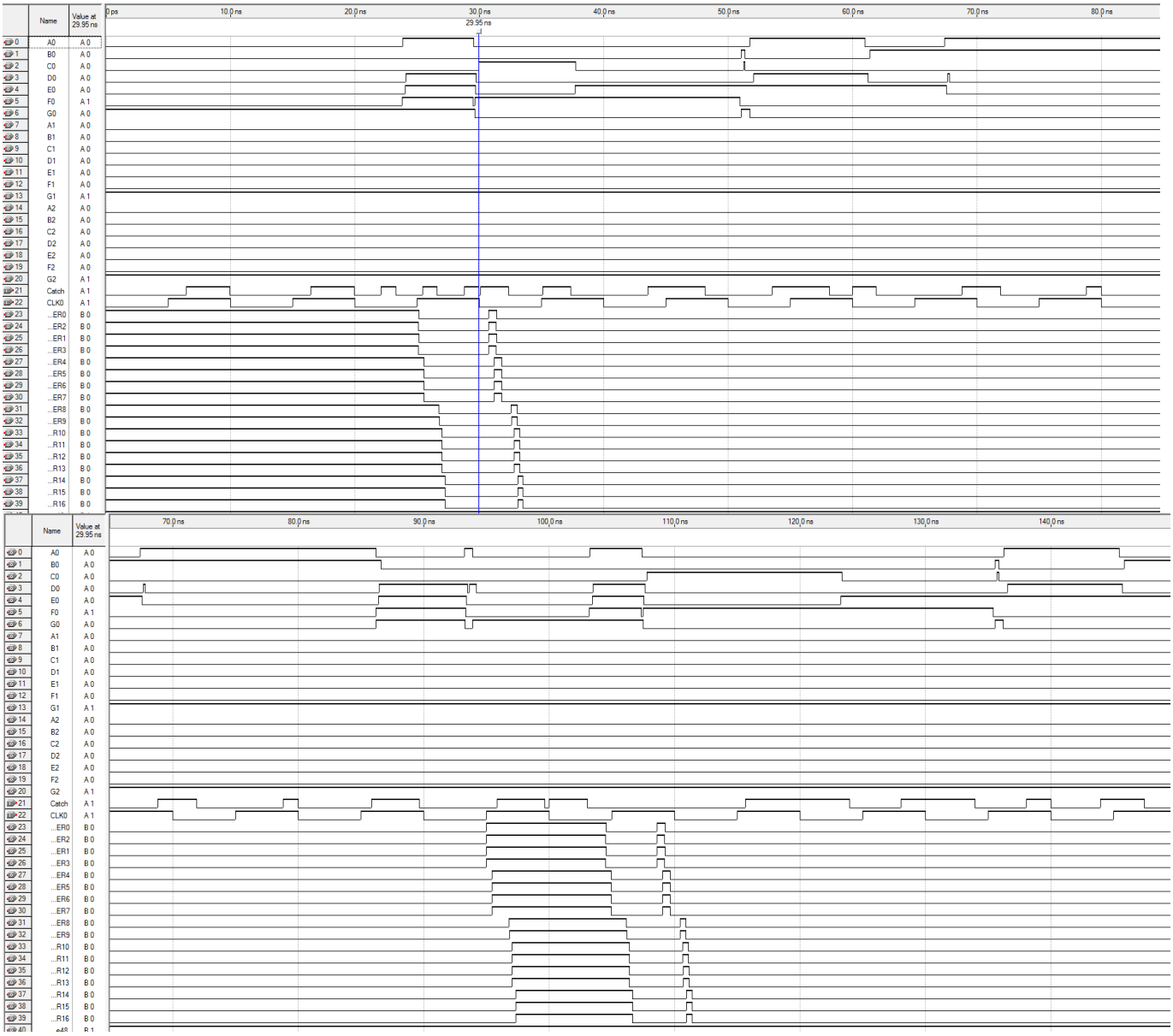
Circuit to black out unused 7-Segment Displays



Gentry, Dane
Gorsuch, Byron

Circuit Operation: The circuit is composed of three Counter Based Number Generators (CBNG's). Each counter is run by a different clock to ensure that the values change at different rates. There is a single catch input that, when activated, will enable each CBNG's D flip-flop which will, in turn, allow whatever value is in each counter to be held in memory and output onto three seven-segment displays, one for each CBNG. The circuit also consists of a comparator to compare the three numbers generated and activate "win" if all three numbers are equal.

Simulation Data



Gentry, Dane
Gorsuch, Byron

Interesting information

Slot machines in casinos work just like ours. The moment the button on the machine is pressed the game will choose one random number for each reel and stop the reel at that appointed position. What we see with the spinning reels and lights is just for show.

We used CBNGs rather than a more pseudorandom number generator such as a linear feedback shift register (LFSR) because the pseudorandom number generator LFSR process is cyclic, a pseudorandom number generator will work as long as the pattern does not repeat too quickly. This is remedied by making the sample space sufficiently large.

With enough observation it is possible, though unlikely, that a person could win at our slot machine every time. This is because we do not use true random number generators in our implementation. We used counter based random number generators which means that each reel in our slot machine changes at a constant rate. It would be possible for a persistent individual to measure the speed at which each reel changes and then press the button exactly when a winning value is expected to appear. The numbers shift so fast, however, that it is probably beyond human reflexes to achieve this.

Conclusions

Since each “reel” has 8 possible values and any set of 3 matching numbers is a winner, the probability of winning our slot machine is 1/64 (assuming true randomness). Therefore, there is a 1.56% chance to win.

There were a few problems encountered trying to implement our schematic. Our biggest issue was the 27 MHz clock did not work. Assigning pin_D13, the pin for the 27 MHz clock according to the pin table, to the input of one of the CBNGs caused the output to remain at 0. We tested our design on multiple DE2 boards to ensure it was not a hardware error, and we could not figure out a reason for the problem.

Using the function generator as the DE2 external clock did not work as expected either. When a probe was used to connect the function generator to the DE2 board the clock would not cycle on its own. We had to manually attach and detach the probe in order to have the counter of that input run. This was an issue because manually running a clock for a slot machine would be inefficient.

We solved both problems by using a cascade of inverters to form ring oscillators, as we learned about in lab 3, as clocks. This use of ring oscillators, however, caused problems in simulation as can be seen in the waveform files above. In Quartus, putting inverters in series does not appear to work as a clock input in simulation. The schematic does work, however, when implemented onto the DE2 board, so nothing was changed in the schematic.