Detailed Functional Description
Contributors: Cooper Krauth & Chase Livingston

## The following user interface resources are utilized by the device:

**SW0-SW15:**
All 16 switches are utilized for various purposes. SW0 is a reset switch that resets the LEDs when in Color Send mode. SW[3:1] have two purposes. Firstly, they are used to select the number of bits to send to the LED strip, and thus the number of LEDs to light up, determined as the binary value of the switches. They are also used to denote Automatic Color Cycle and Game mode (3'b110 and 3'b111 respectively). When the device is in Color Send mode, SW[15:4] denote the GRB bit patterns that will be sent to the LEDs. SW[15:12], SW[11:8], and SW[7:4] correspond to the Green, Red, and Blue channels respectively.

**JA1:J1:**
The port JA1 is used for serial data transfer to the LEDs and should be connected to the DataIn line on the LEDs.

**BTNC:**
BTNC is used at the "Go" signal for serial data transfer in Color Send mode. It is also used in Game mode to receive input from the user as the LEDs are flashing. BTNC does not have any functionality in Automatic Color Cycle mode.

**LED0:**
LED0 is used to denote a state in the SSStateMachine FSM. LED0 is assigned to be on when the "Ready2Go" signal is 1, meaning that when LED0 is on, the device is ready for serial data transfer.

**Seven-Segment Display:**
The seven-segment LED display is constantly driven as AN[3:0] = 4'b1110. This results in only the rightmost display being driven and is sufficient for the design's purposes. The seven-segment LEDs will display the current score of Game mode, starting at 0 and incrementing to 7, at which point the user has won the game.

Figure 1 shows the RTL schematic of the device. It gives a high level overview of the following modules as defined within SimpleSend, the top level module: BitCounter SSStateMachine, GRBStateMachine, NZRBitGen, ShiftRegister, and Score27Seg, all of which will be described in the following section.

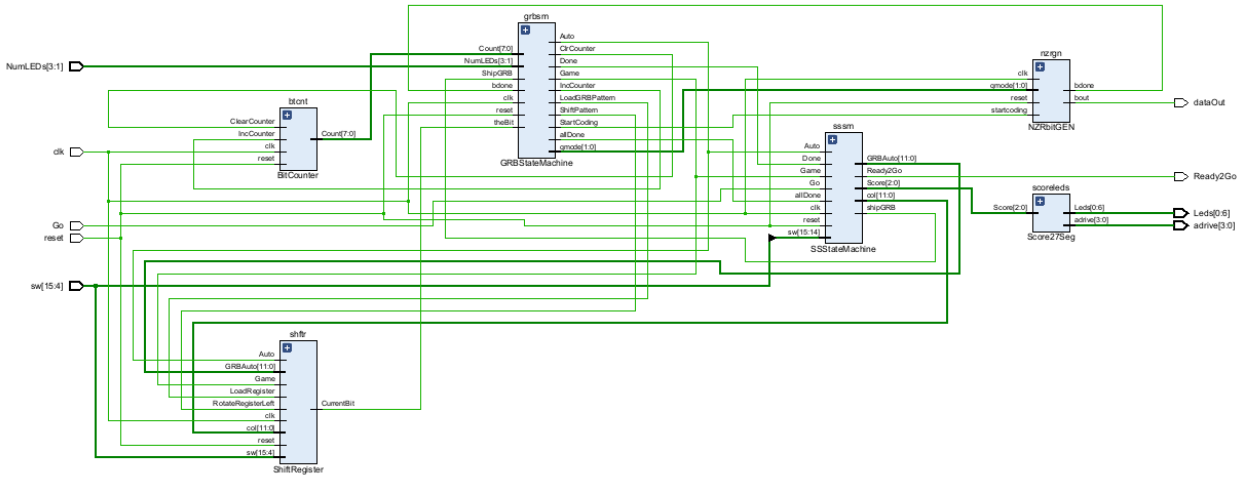


Figure 1: RTL Design Schematic

## Functionality:

### Simple Send:

The top level module, SimpleSend, receives 5 inputs: Go, sw, NumLEDs, clk, and reset. Two of its outputs, dataOut and Ready2Go, are sent to the LED strip and LED0 respectively. The outputs Leds and adrive are sent to the seven-segment display and are used to drive the rightmost display, on which the current score of the game is displayed. The sw, NumLEDs, and reset inputs come from the basys switches and are used to control the GRB values, number of bits to send, and active high reset respectively.

### SSStateMachine:

The state machine module, SSStateMachine, receives inputs from the clk, reset, and switches 15 and 14. Switches 15 and 14 are used to control the speed on the Automatic Color Cycle mode, and reset is used as an active high reset for the Color Send mode. The inputs Game and Auto are used in the state machine to determine which mode the device should be in. When the machine is in Game mode, a simple 3-bit linear feedback shift register is used to generate a pseudo-random sequence of colors to blink on the leds, which is the col output. When in Auto mode, the color generation is incremented by an internal counter. This output is GRBAuto. There are also inputs allDone and Done which come from the GRBStateMachine and are used for debouncing and determining when the device is ready to send more serial data to the LEDs. A separate

submodule is included for debouncing and edge detection of the button input specifically for the game application, in which the application of incrementing the score is slightly different. The output shipGRB is sent to the GRBStateMachine to initiate serial data transfer. Finally, the output Ready2Go is sent to LED0 as a visual indicator that the device is ready to send data, and the score of the game is sent to a decoder.

**GRBStateMachine:**
The GRBStateMachine module receives the input NumLEDs from the Basys switches and uses this to determine the number of LED modules to send to, as well as the Game and Auto modes, which are outputs. The input Count comes from a bit counter and is used to determine if all the bits specified by NumLEDs have been accounted for. This count is controlled by the outputs ClrCounter and IncCounter. There are also the outputs bdone, StartCoding, and qmode, which are used to control the generation of bits for dataout. Bdone is a clock tick as the frequency required by the LED strip, qmode specifies whether the data should be equivalent to 1, 0, or reset, and StartCoding is a signal that tells the generator when to restart the count, similar to reset. GTBStateMachine also has outputs LoadGRBPattern and ShiftPattern, which are sent to a shift register, making sure that one bit is output at a given instant.

**NZRBitGen:**
The NZRbitGEN module handles the generation of bits in the correct line code for the LED strip. The output bout is the line code generated by the module, and the output bdone indicates that generation is finished. The input startcoding resets the internal counter, and the counter resumes incrementing from 0. This internal count, in combination with qmode, generates the line code dependent on which value qmode has (i.e. 0, 1, or reset). This module also has inputs for clk and reset.

**BitCounter:**
The BitCounter module is a simple 8-bit counter that has inputs to clear or increment the counter, and its only output is the value of the count, which is used to track how many bits have been sent to the LED strip.

**ShiftRegister**
The ShiftRegister module is used to take the switch inputs and pad them with zeros to convert the 12 bit input to the 24 bits actually used in serial data transfer of GRB values. This padded 24 bit register is then rotated left every time the RotateRegisterLeft signal is received, until all the bits have been output as the CurrentBit. The register should then be parallel loaded with new GRB bits upon receiving the LoadRegister signal. This module also has Game and Auto inputs that determine which bits to swizzle when loading the register. If the input is Game, then the value of the col input is used, and if the input is Auto, then the GRBAuto input is used (switches used as input by default).

**Score27Seg:**

The Score27Seg module is a simple 3-bit decoder that takes the score of the game as input and outputs seven-segment LED values. It also outputs adrive, which constantly drives the rightmost display, as multiplexing is not needed for only one display.