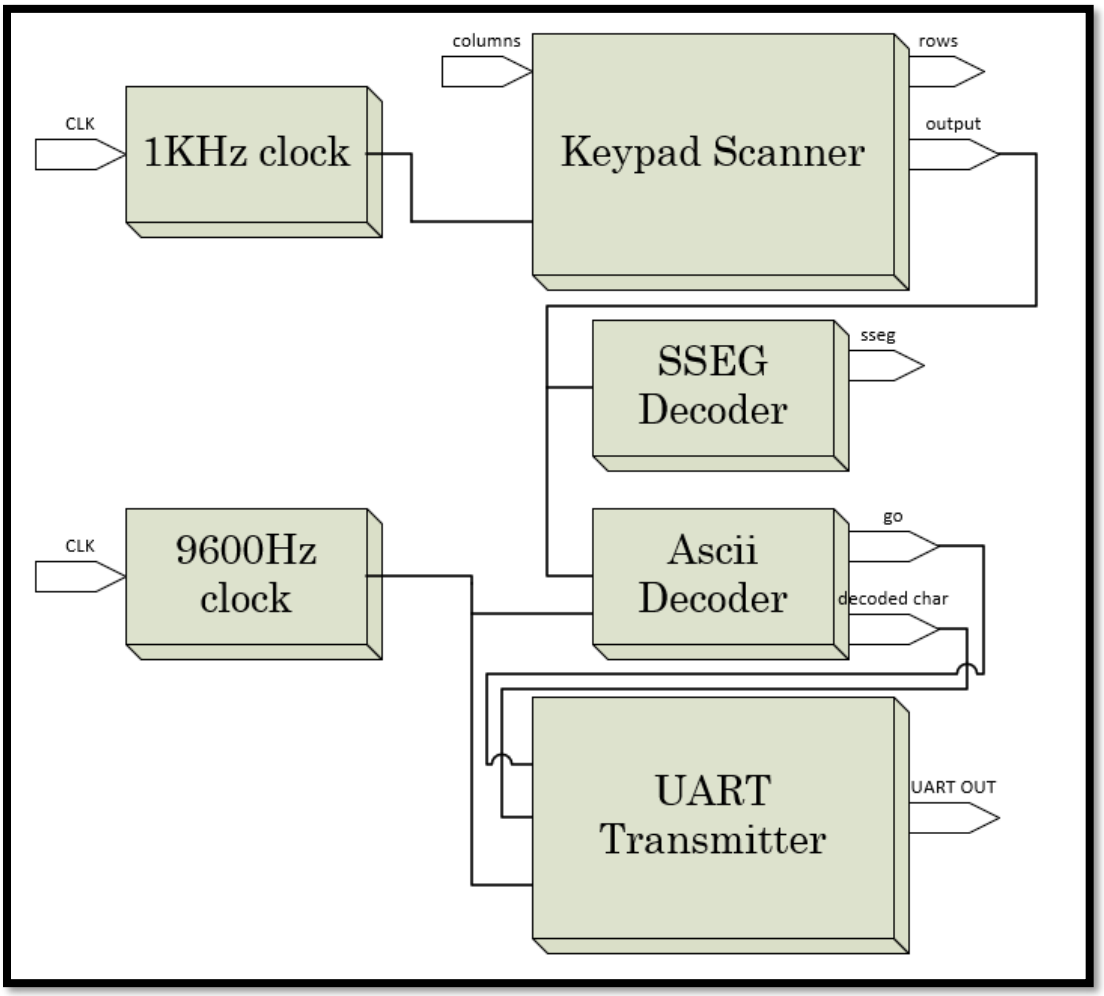
Zak Rowland
Digital Systems Design I
Lab 7 – Keypad

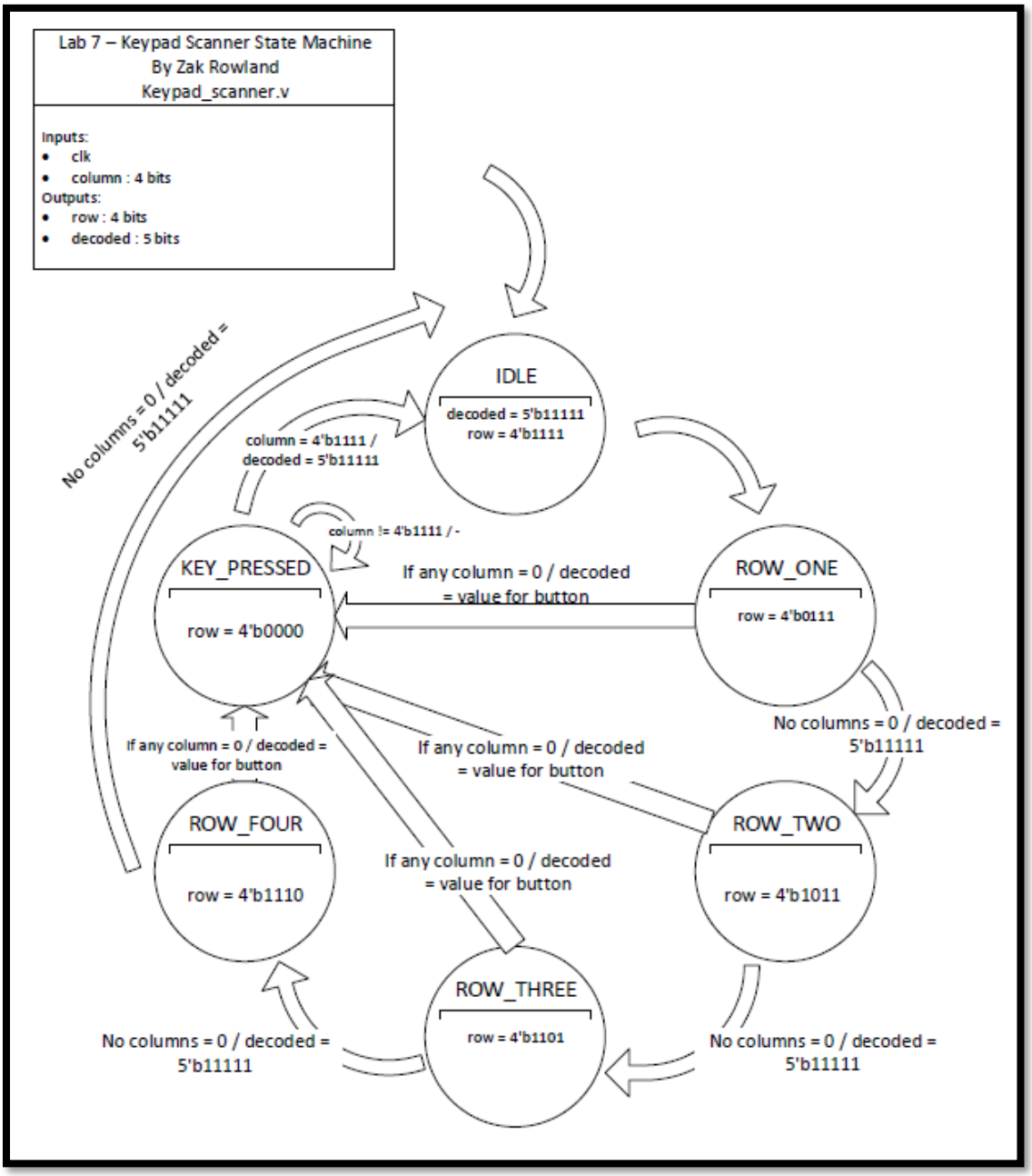
I chose the columns to be inputs and rows to be outputs. My solution consisted of 6 modules:

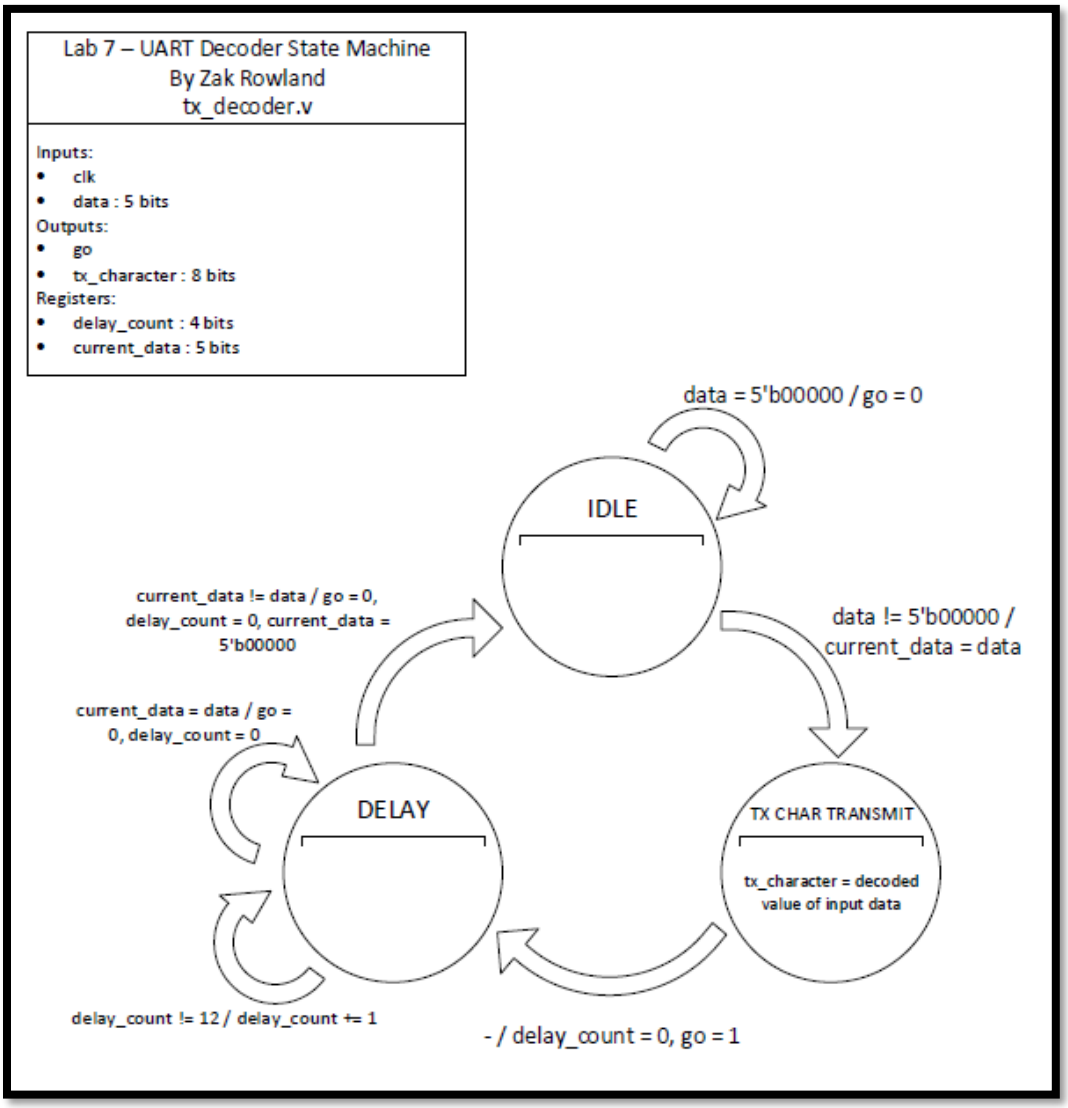
1. A 48MHz to 1KHz clock divider
2. A 48MHz to 9600Hz clock divider
3. The keypad scanner
4. A decoder for the seven segment display
5. A decoder for ascii outputs to PuTTY
6. A UART transmitter

The 1KHz clock was used to run the keypad scanner. The scanner uses a state machine to go through powering each row and check each column for button presses. If I button is pressed, the output is set to the correct value for that button. This output is fed to the decoders for the seven segment display and ascii output. Then the state machine moves to a "key pressed" state. In this state, every row is powered and if any column still sees a button press the hardware will loop in this state. This is how I implemented my button lockout. With this method no matter what key is pressed, no other key affects it, including those from the same column/row. Once the button is released, the state machine starts over and continues scanning. The ascii decoder also controls the UART output by allowing only one character to be send for a button press. It does this by making a copy of the input data in an internal register, and checking if they are the same after a character is output. The UART transmitter receives the decoded data from the ascii decoder and outputs it serially through Header B, Pin 37. The only issue with my approach is that my decoders use sequential logic instead of combinational. I learned it will still work, but it is just bad practice because it won't always work as expected.

Pin on Keypad	Row/Column?	owlBoard pin	FPGA pin	Set as
1	Column	17	80	input
2	Column	15	78	input
3	Column	13	74	input
4	Column	11	55	input
5	Row	9	50	output
6	Row	7	47	output
7	Row	5	45	output
8	Row	3	43	output


columns
rows
CLK
1KHz clock
Keypad Scanner
output
SSEG
Decoder
sseg
CLK
9600Hz
clock
Ascii
Decoder
go
decoded char
UART
Transmitter
UART OUT


Lab 7 – Keypad Scanner State Machine
By Zak Rowland
Keypad_scanner.v
Inputs:
clk
column : 4 bits
Outputs:
row : 4 bits
decoded : 5 bits
IDLE
decoded = 5'b11111
row = 4'b1111
column = 4'b1111 /
decoded = 5'b11111
No columns = 0 / decoded =
5'b11111
column != 4'b1111 / -
KEY_PRESSED
row = 4'b0000
If any column = 0 / decoded
= value for button
ROW_ONE
row = 4'b0111
No columns = 0 / decoded =
5'b11111
If any column = 0 / decoded =
value for button
If any column = 0 / decoded
= value for button
ROW_FOUR
row = 4'b1110
If any column = 0 / decoded
= value for button
ROW_TWO
row = 4'b1011
ROW_THREE
row = 4'b1101
No columns = 0 / decoded =
5'b11111
No columns = 0 / decoded =
5'b11111


Lab 7 – UART Decoder State Machine
By Zak Rowland
tx_decoder.v
Inputs:
clk
data : 5 bits
Outputs:
go
tx_character : 8 bits
Registers:
delay_count : 4 bits
current_data : 5 bits
data = 5'b00000 / go = 0
IDLE
current_data != data / go = 0, delay_count = 0, current_data = 5'b00000
data != 5'b00000 / current_data = data
current_data = data / go = 0, delay_count = 0
DELAY
TX CHAR TRANSMIT
tx_character = decoded value of input data
delay_count != 12 / delay_count += 1
- / delay_count = 0, go = 1

Zak Rowland
Digital Systems Design I
Lab 7 – Keypad