CMOS VLSI Lab
Fall 2014

Final Project Report
Digital Thermometer ASIC

Today's Date:	12-12-14
Group Members and Work Distribution:	**John Gangemi** – Designed logic for ASIC, Created layout components, Integrated components into padframe (Floorplanning & Routing) **Alex Holst** - Padframe I/O buffering, Assisted with layout and circuit testing **Raj Patel** - Wrote Reports, Assisted with Layout , and Coordinated Meetings
No. of Hours Spent:	100
Exercise Difficulty: (Easy, Average, Hard)	Hard
Any Other Feedback:	Power routing proved to be the most difficult part of this project.

I. INTRODUCTION

In this lab we were tasked with creating a custom Application Specific Integrated Circuit (ASIC) which should be able to interface with other components to build a digital thermometer. The digital thermometer should be able to read values in both Celsius and Fahrenheit.The temperature range of the ASIC is from 0 to 32 degrees Celsius and its equivalent in Fahrenheit. Also the user should be able to choose the way that they want to display the temperature through the mode button and there should be a reset button to reset the ASIC.

II. SYSTEM DESCRIPTION

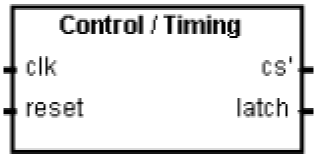
The system specifications outlined in the design document have not changed, the four variations of Mode0 and Mode1 remain the same as shown below...

Mode 1	Mode 0	Out1 (0-6) & Out2 (0-6) [*7 Segments*]	Out0 (0-7) [*Test Pins*]
0	0	Celsius-to-Fahrenheit module	Celsius-to-Fahrenheit module
0	1	SPI module	SPI module
1	0	Celsius-to-Fahrenheit module	SPI module
1	1	SPI module	Binary-to-BCD module

The ASIC was designed around the requirements for Mode0 and Mode1. All other features such as a system reset were implemented after the fact. While the original intention was to design a circuit that could function at 1 KHz it was quickly decided that a 500 Hz clock was far more convenient to control and still updated the circuit more than once per second as required.

III. LOGIC DESIGN

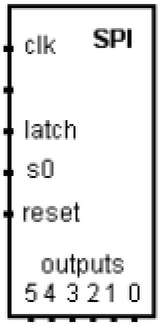
Control



Components:
.8-bit Counter (mod 169)
.1 NAND & 3 NOR gates

This logical block is fundamentally the most important block in the entire design as it provides basic functionality to the other blocks and slave devices connected to the ASIC. An 8-bit counter will assert ‘cs’ active low at clock cycle 160 (minimum conversion time of 320 ms) triggering the TMP121 to transmit data through the SPI protocol. After nine clock cycles (169) the counter will assert ‘latch’ active high and on the 170th clock cycle the counter will asynchronously reset to zero therefore setting ‘cs’ to active high and ‘latch’ to active low.

SPI

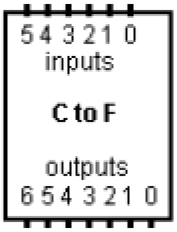


Components:
.6-bit Left Shift Register (6BLSR)
.6-bit Parallel In/ Parallel Out Shift Register (6BPIPO)
.1 NOR gate

The 6-bit Left Shift Register in this block continuously shifts in data given the presence of a clock signal. By design the 6BLSR is triggered on the rising edge and has no means of clearing its registers, as it’s not necessary to perform such event when there is a 6BPIPO used to transmit the correct celsius binary value to the combinational logic blocks that follow.

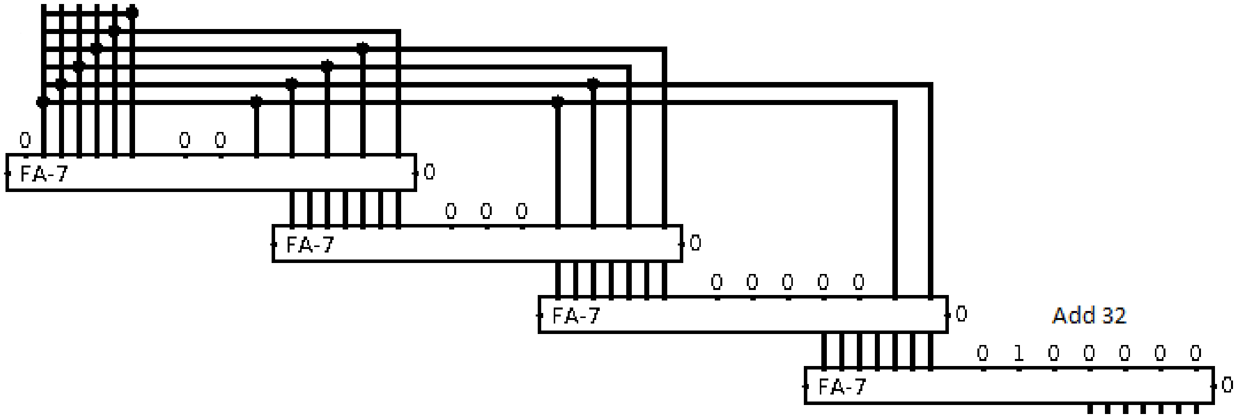
Upon ‘latch’ being active high and the clock is on the falling edge of its 169th cycle the 6BPIPO will latch data from the outputs of the 6BLSR assuming the 6BLSR has the appropriate 6 bits (12-7 from TMP121) in its registers. Therefore the 6BPIPO blocks the combinational logic blocks from calculating the intermediate values of the 6BLSR as it shifts in data.

Celsius to Fahrenheit

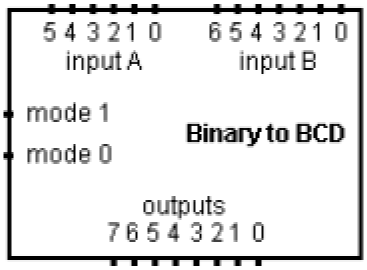


Components:
.4 x 7-bit Full Adder (7BFA)

In order to compute a fahrenheit value from the 6-bit celsius value given by the SPI block, a series of four 7-bit Full Adders are required to perform the calculation *F=C*9/5 + 32* . The calculation requires the computation of 9/5 which can be done by left shifting the 6-bit input once for the first 7BFA, twice for the second 7BFA, and four times for the third 7BFA. The algorithm leads to the equivalent mathematical equation of C*1.8125. The following logical representation is shown below…

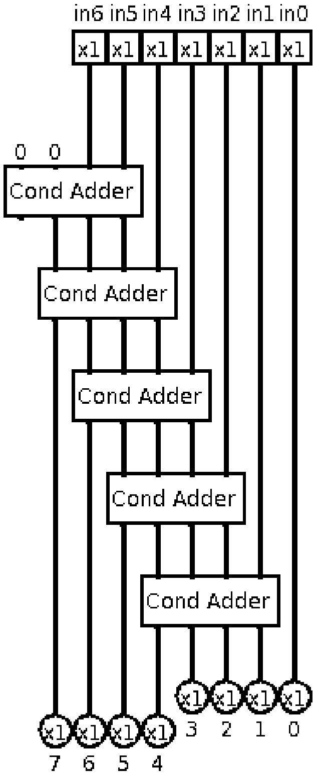


Binary to BCD



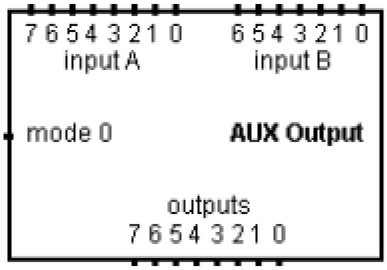
Components:
.5 x 4-bit Conditional Adder
.7 x 2:1 Multiplexer (14 to 7 Selection)

The conversion of binary to binary-coded-decimal is done through the implementation of a “Double Dabble” algorithm. This requires five 4-bit Conditional Adders in which the 4-bit binary number increased by three if it is greater than four (0100) in a working set from 0-9. The resulting output is an 8-bit number that supports the maximum two-digit requirement for the ASIC. The following image shows the structure of the “Double Dabble” algorithm…



Selection of input bits is determined by a block of seven 2:1 Multiplexers which select between SPI and Celsius-to-Fahrenheit modules, control signal for the multiplexers is determined by the ORing of Mode0 and Mode1.

AUX Output

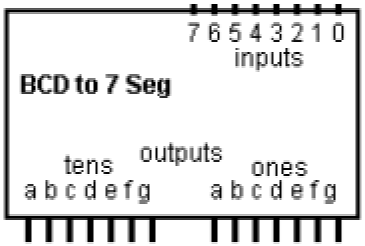


Components:

.7 x 2:1 Multiplexers (14 to 7 Selection)
.2:1 Multiplexer

Given the value of Mode0 the AUX logic block will either send the output of the Celsius-to-Fahrenheit block or the Binary-to-BCD block to the 8 I/O pins prefixed with Out0. This logic block is purely for meeting the testing requirements set forth by the design guidelines.

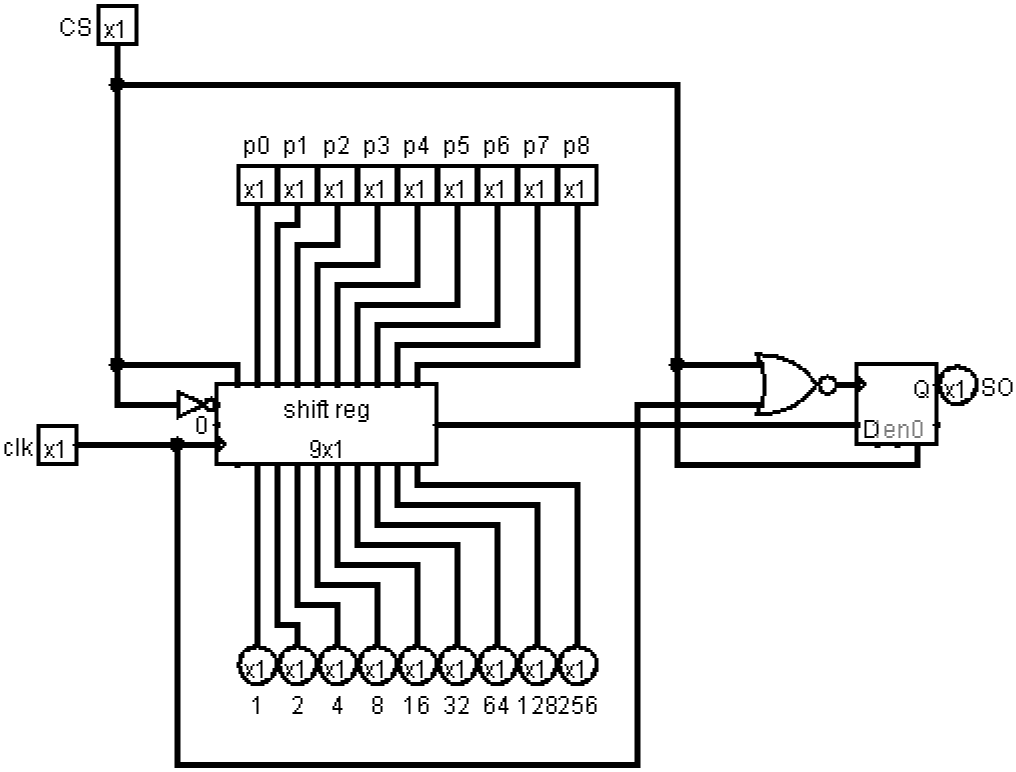
7 Segment Display Driver



Components:
.2 x 7 BCD-to-7 Segment Driver

Interfacing with 7 Segment Displays requires the translation of binary-coded-decimal values into a 7-bit value where each bit represents a “segment” of the display and is activated when the respective bit is active high.

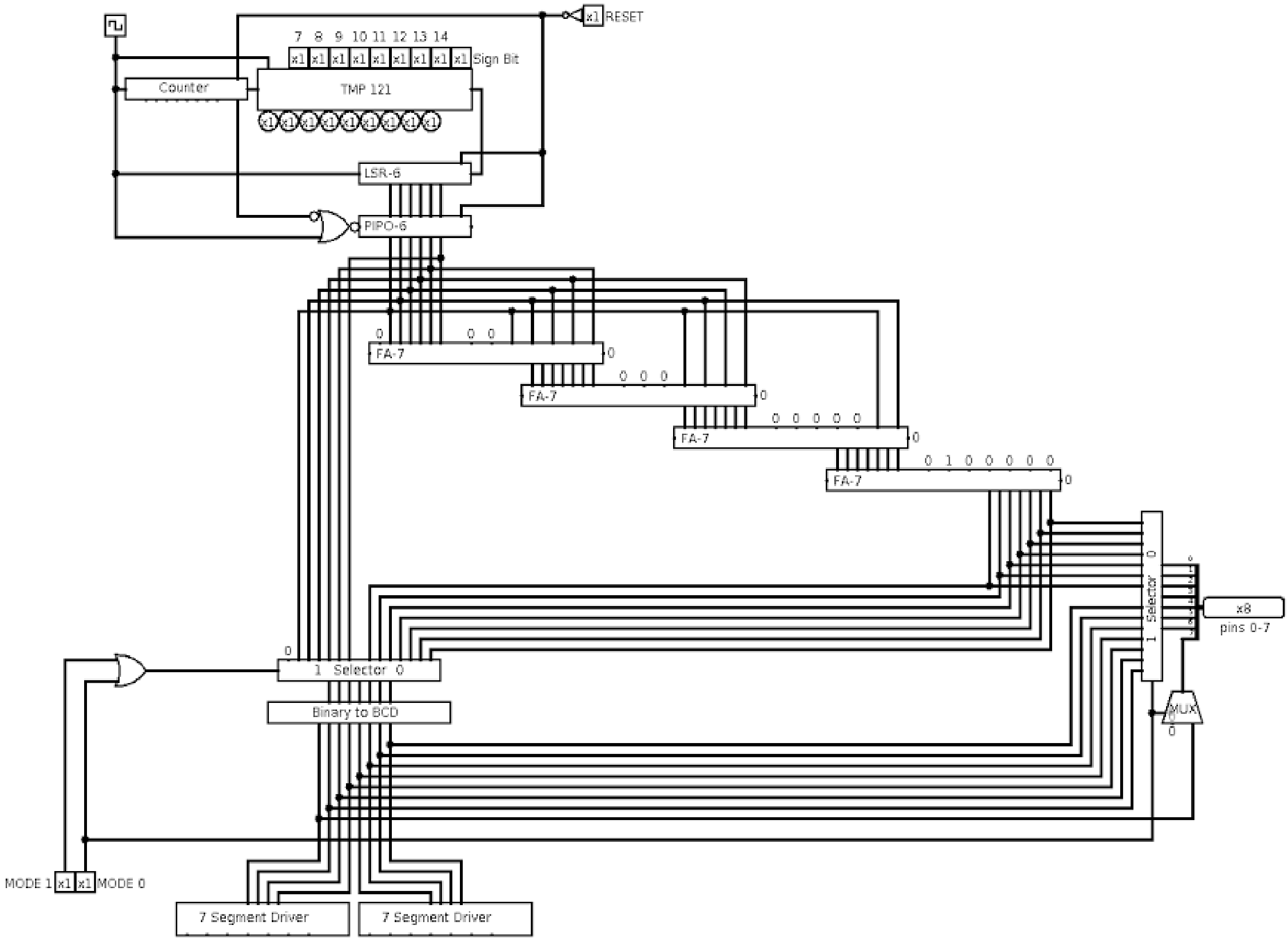
TMP 121



Description:

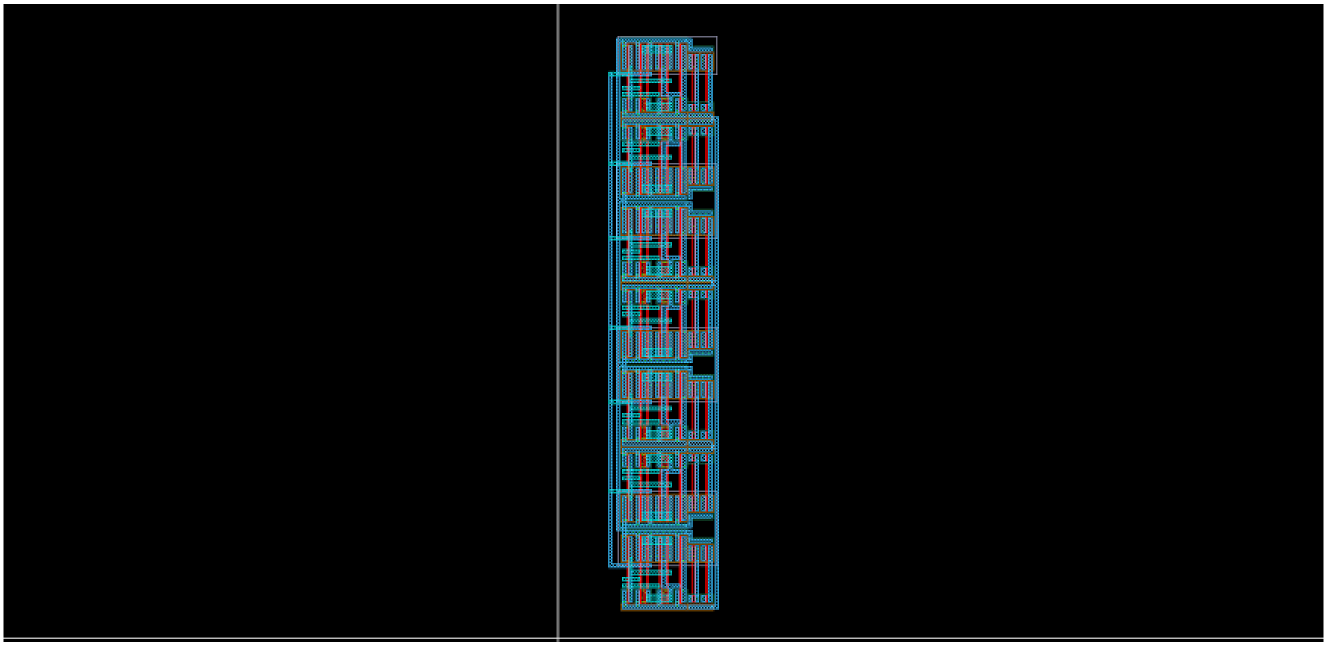
Since the design of the ASIC is completely dependent upon the functionality of the TMP121 circuit it was deemed necessary to model the temperature sensor in Logisim to sufficiently test a design before beginning work in layout. The circuit diagram shown above will shift out a single bit when 'cs' is active low and the clock is on its falling edge. A temporary storage element allowed the designer to check the signal being propagated from the TMP121 whilst stepping through clock cycles to verify a design's SPI module.

Digital Thermometer 500 Hz

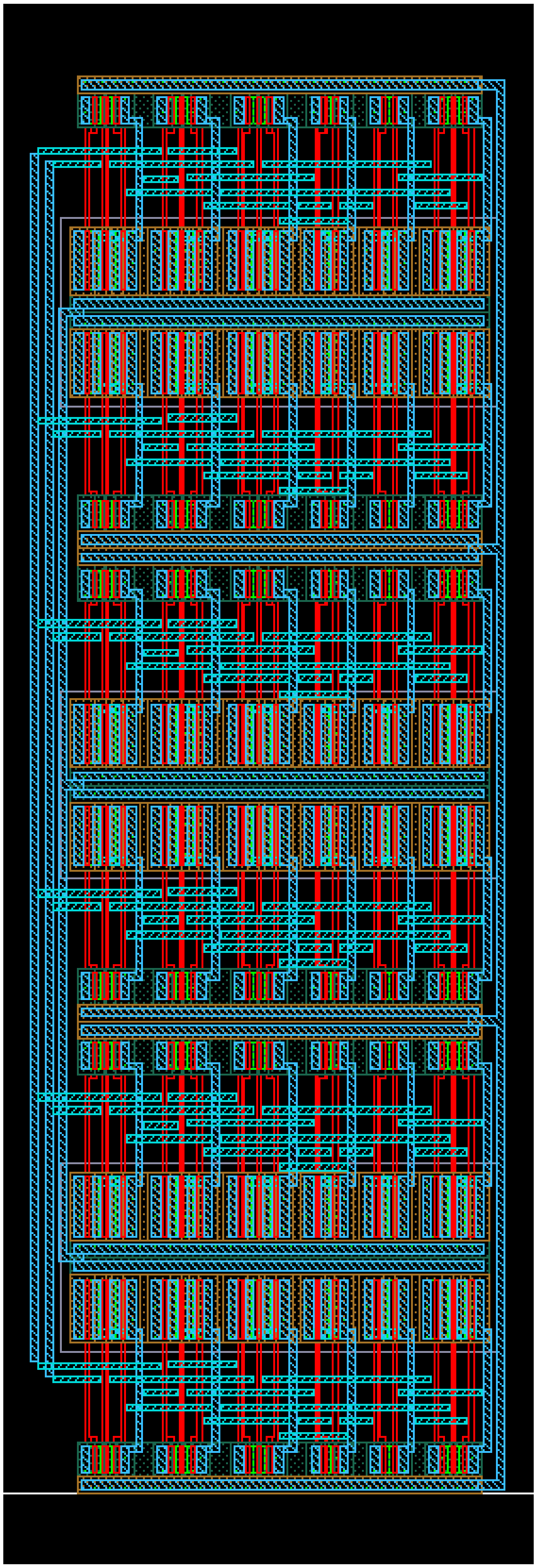


IV. LAYOUT DESIGN

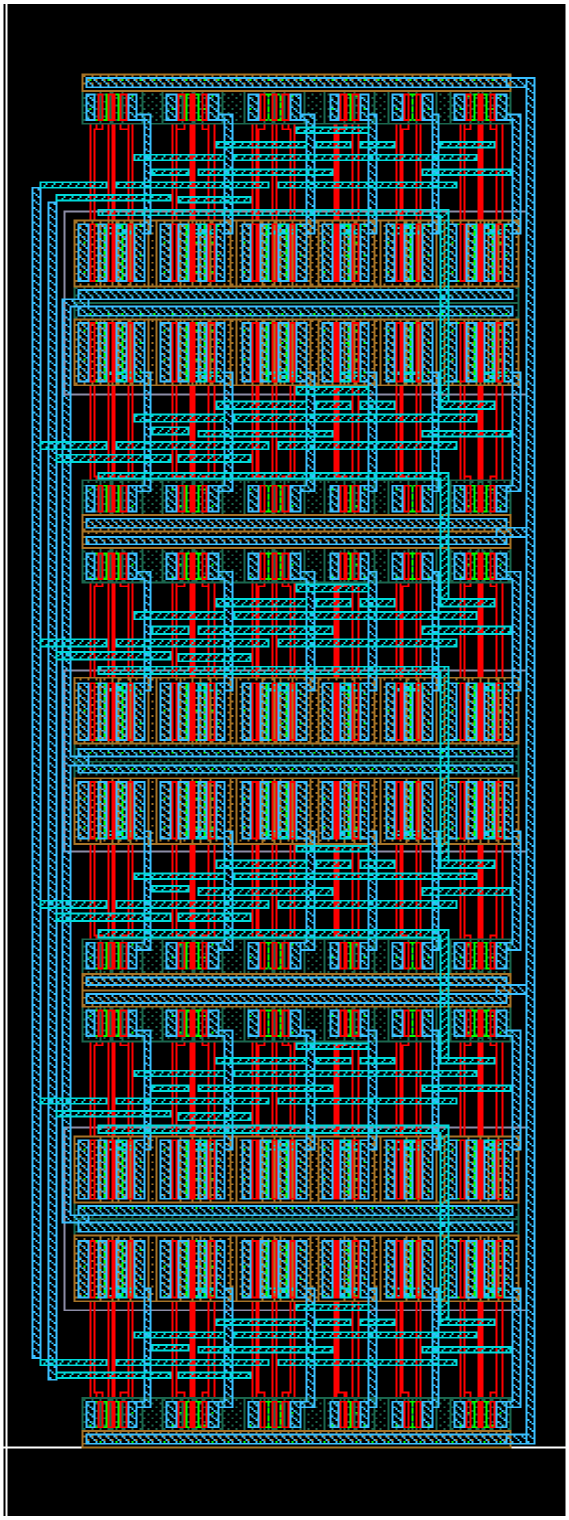
14 to 7 selection block



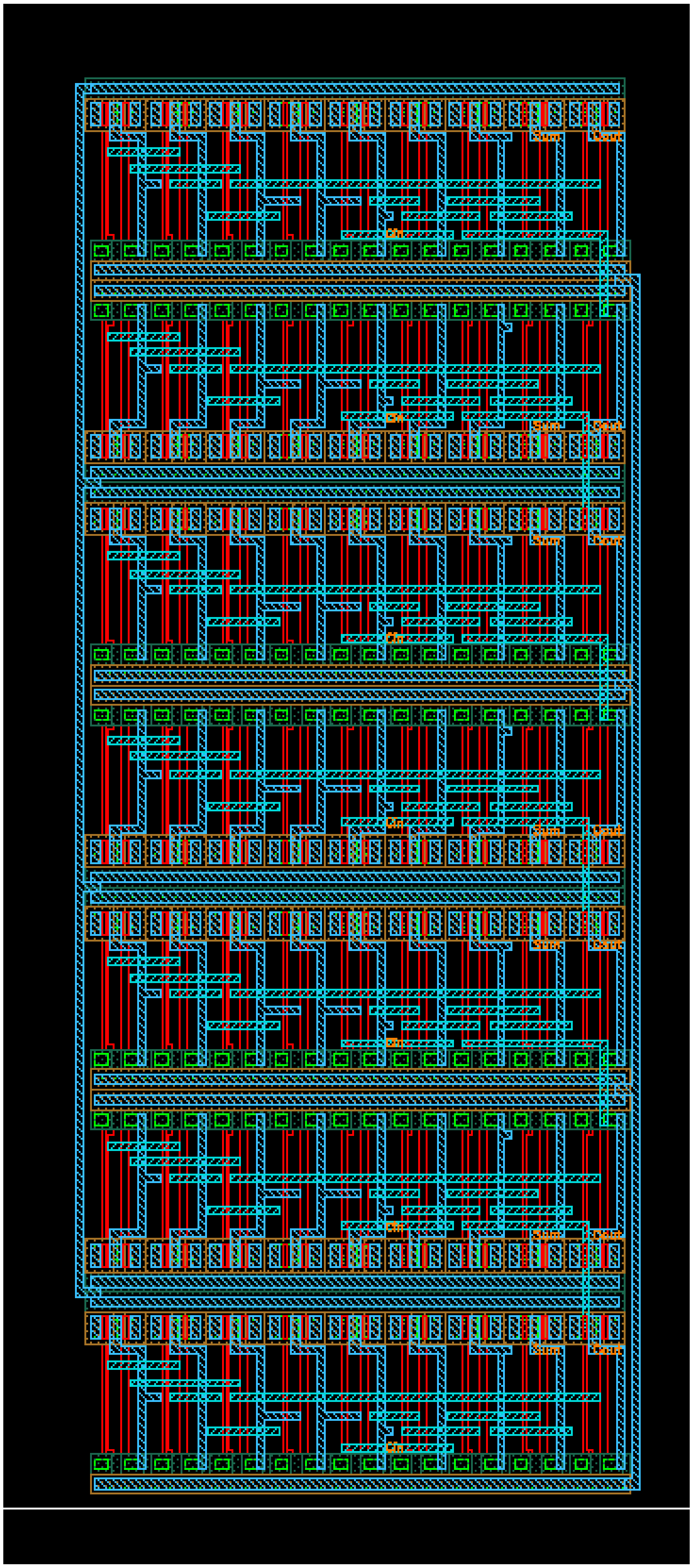
6 bit Parallel In/Parallel Out Register



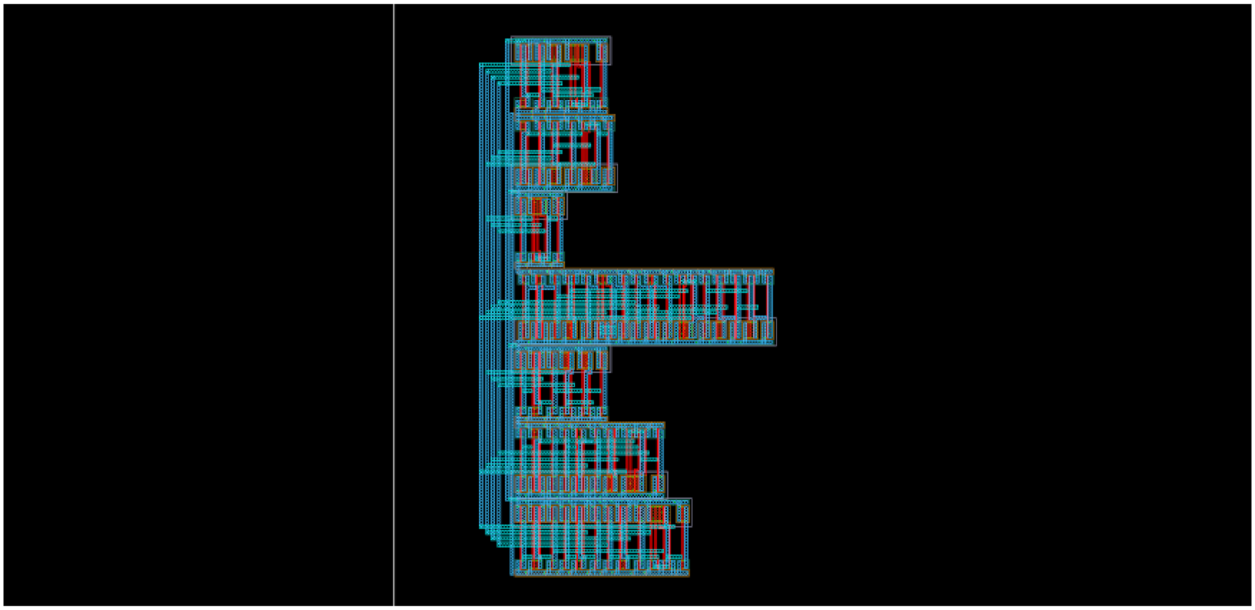
6 Bit Shift Register



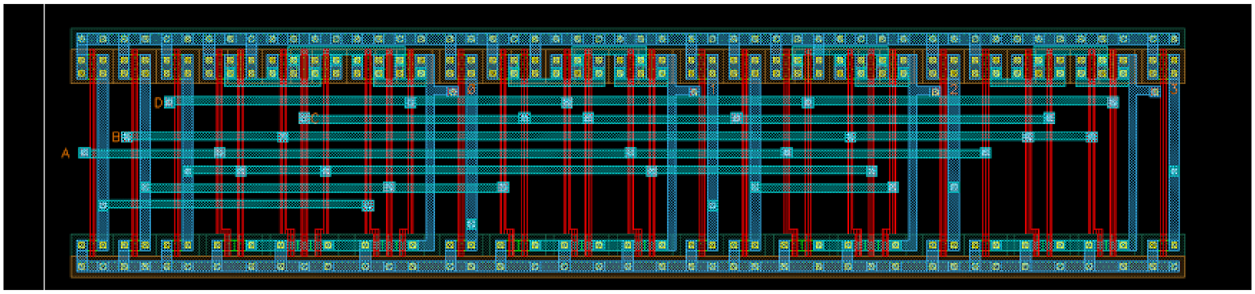
7 bit Full Adder



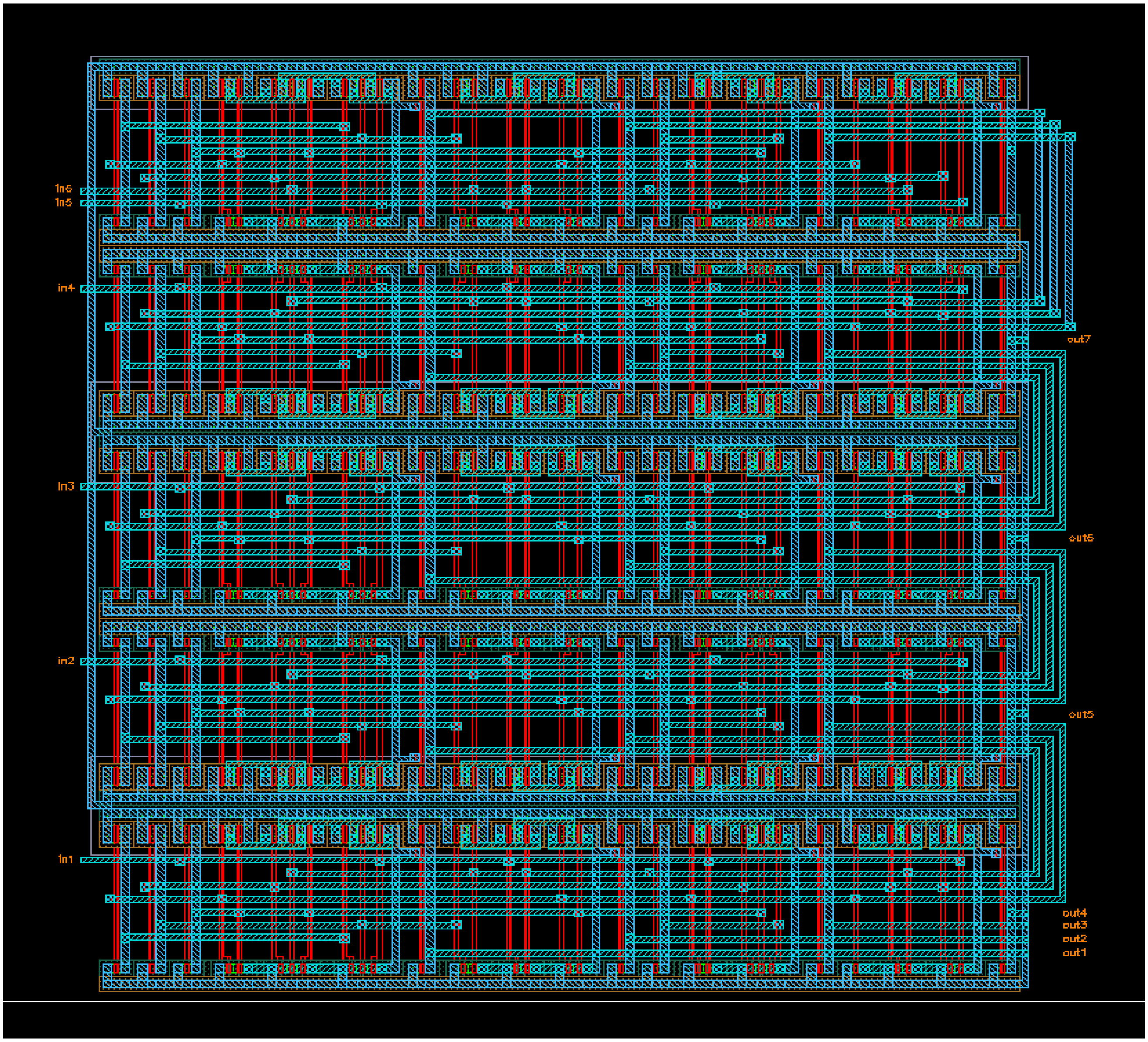
7 Segment Driver



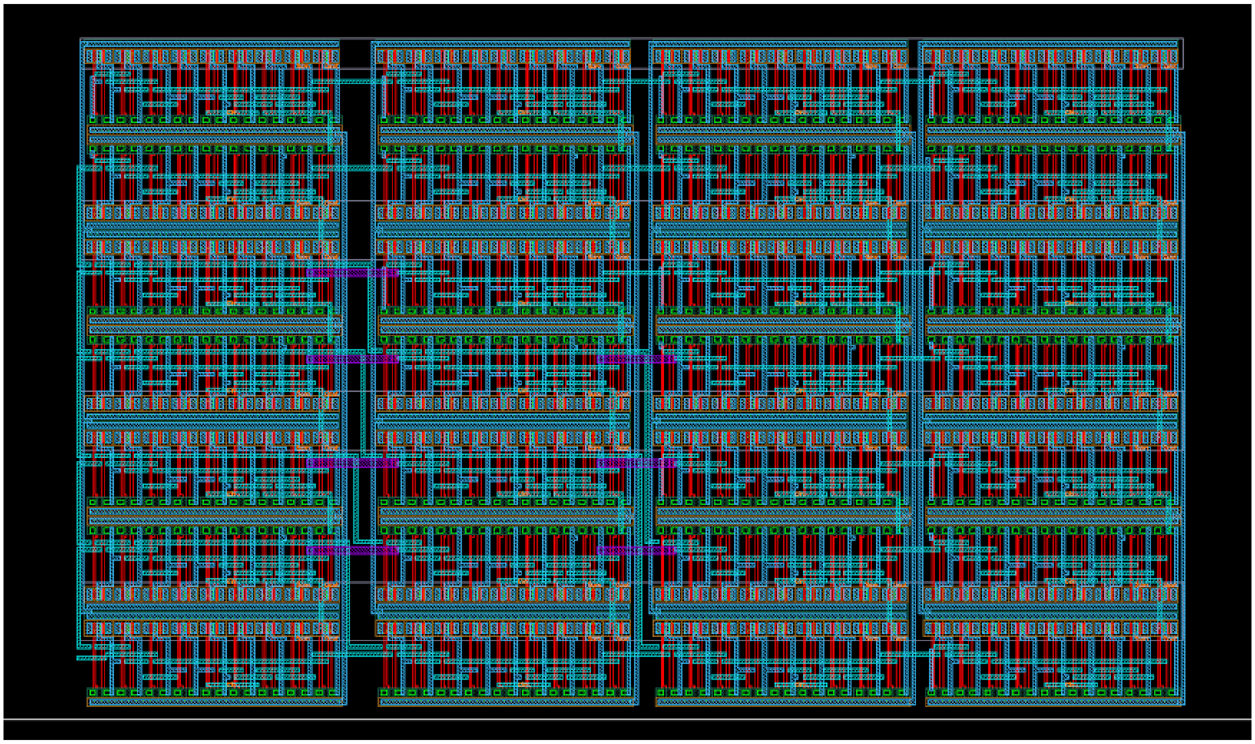
Conditional Adder



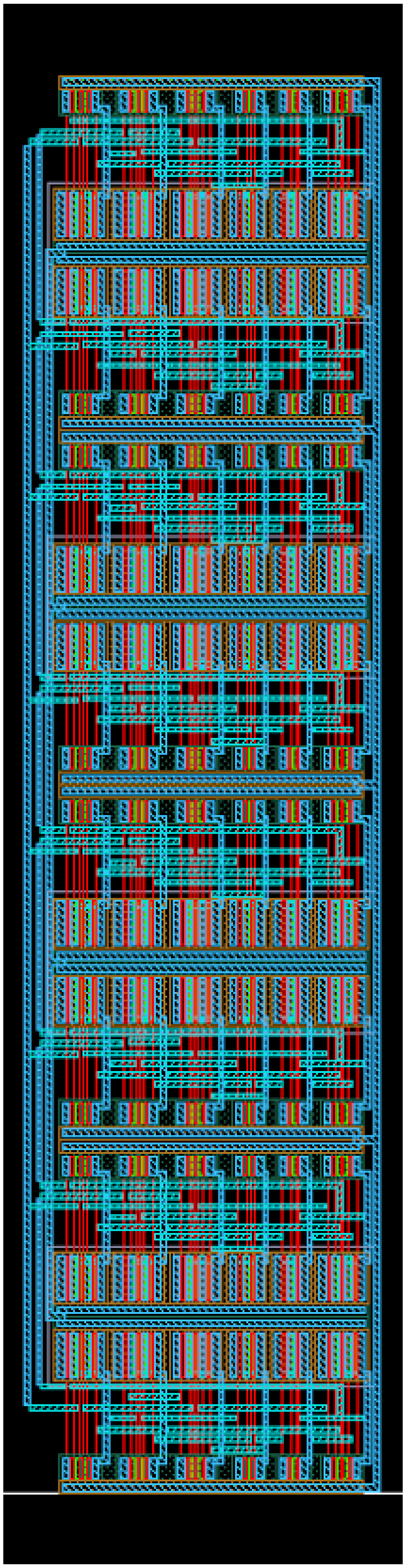
Binary to BCD



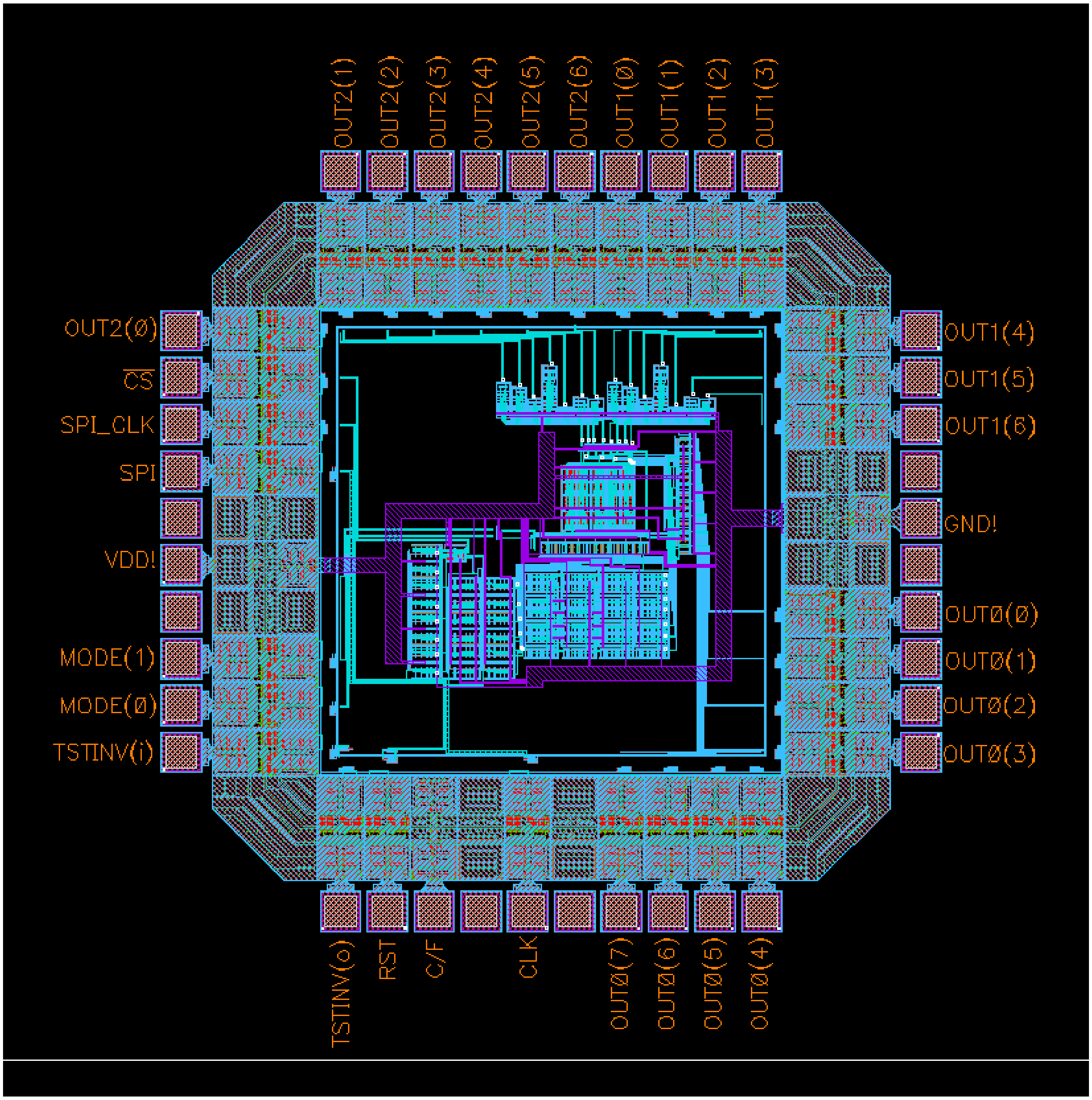
Celsius to Fahrenheit



8 bit Counter

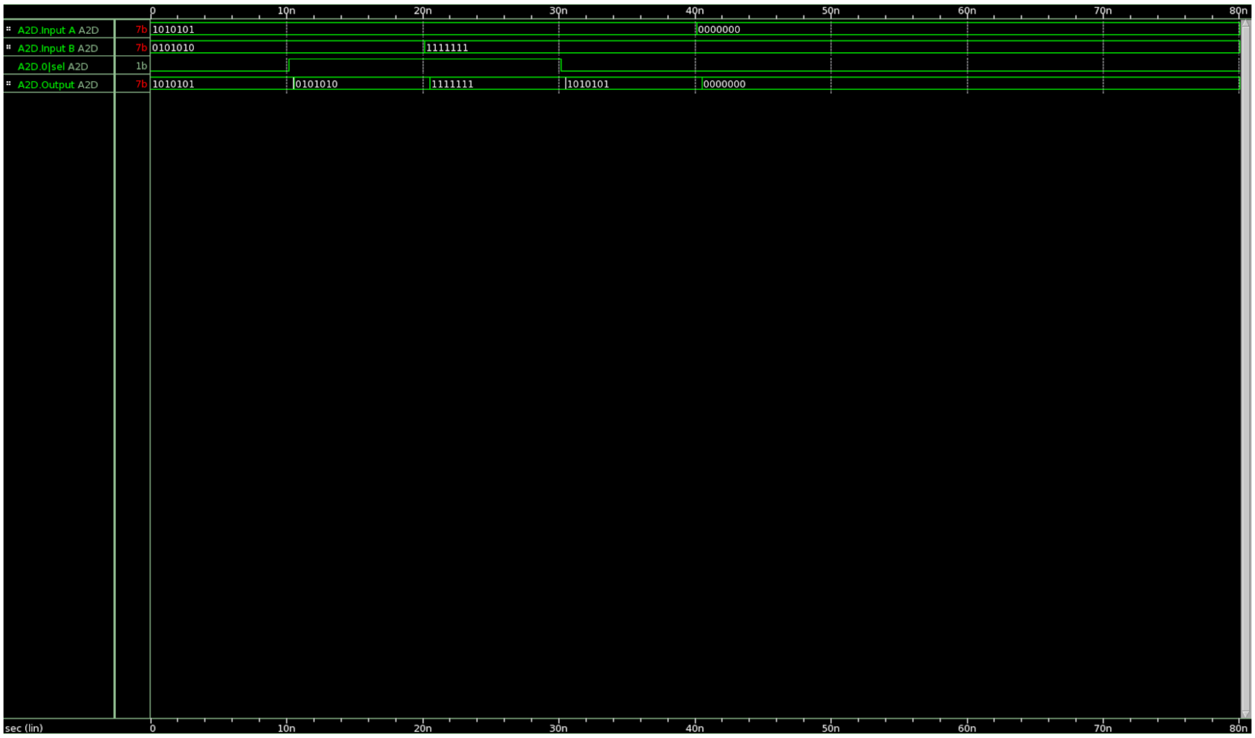


Digital Thermometer

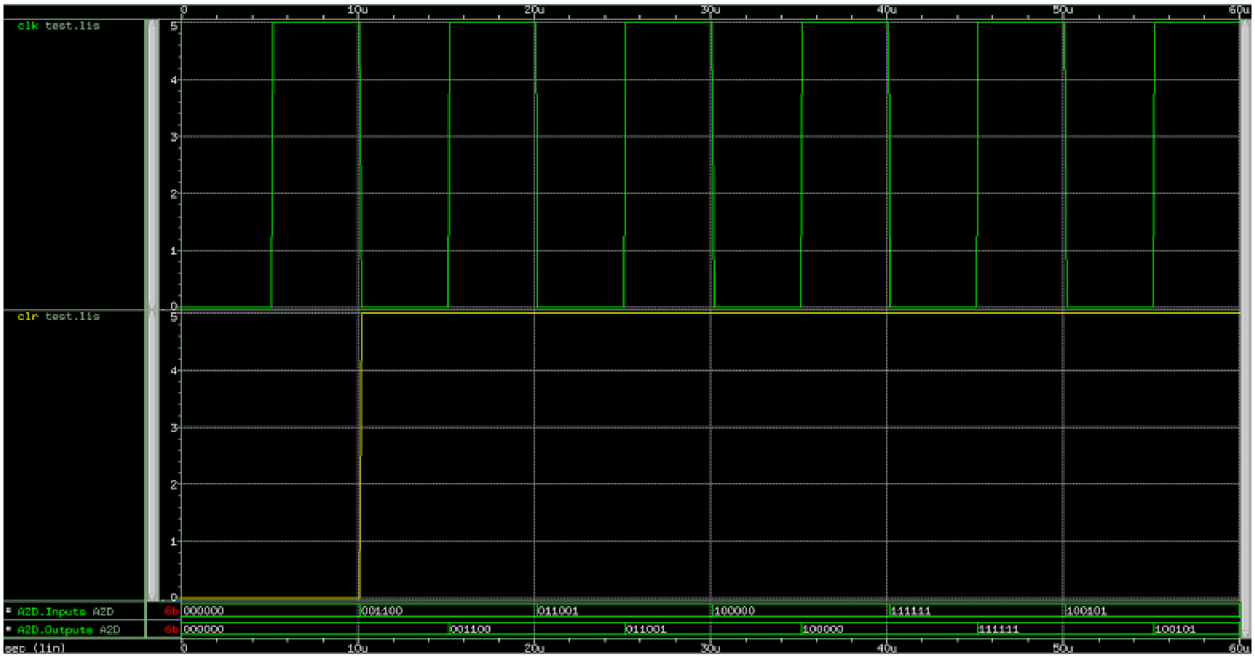


V. SIMULATIONS

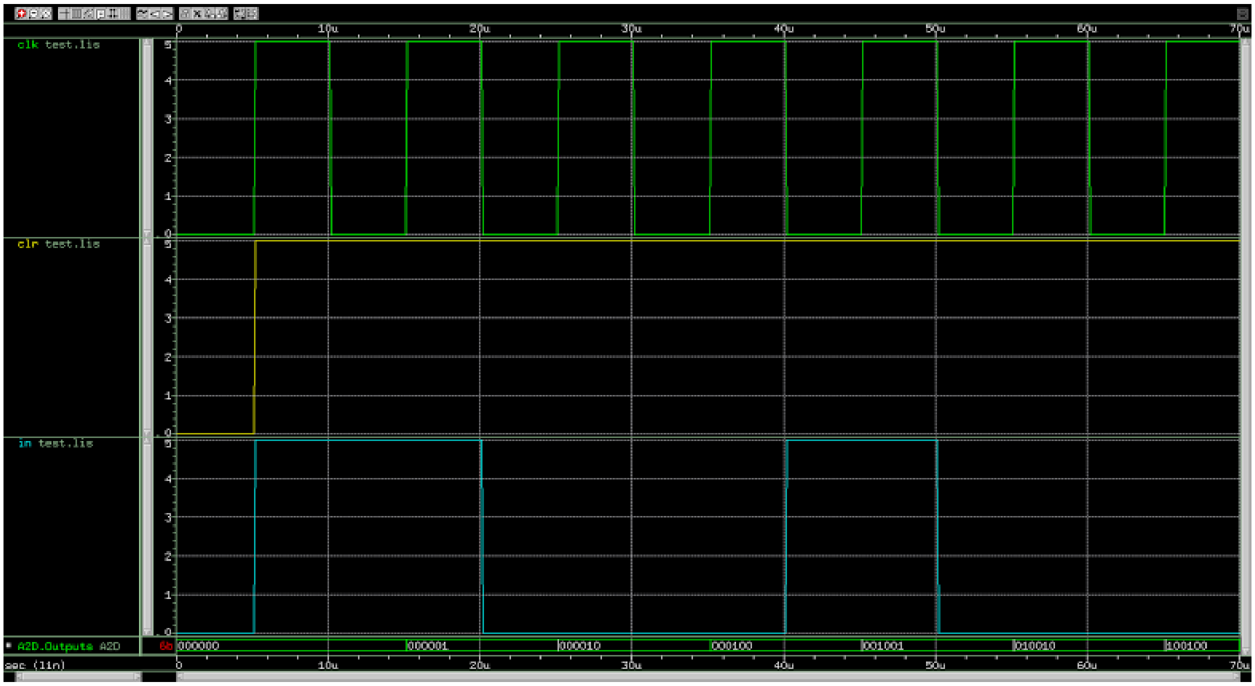
14 to 7 Selection



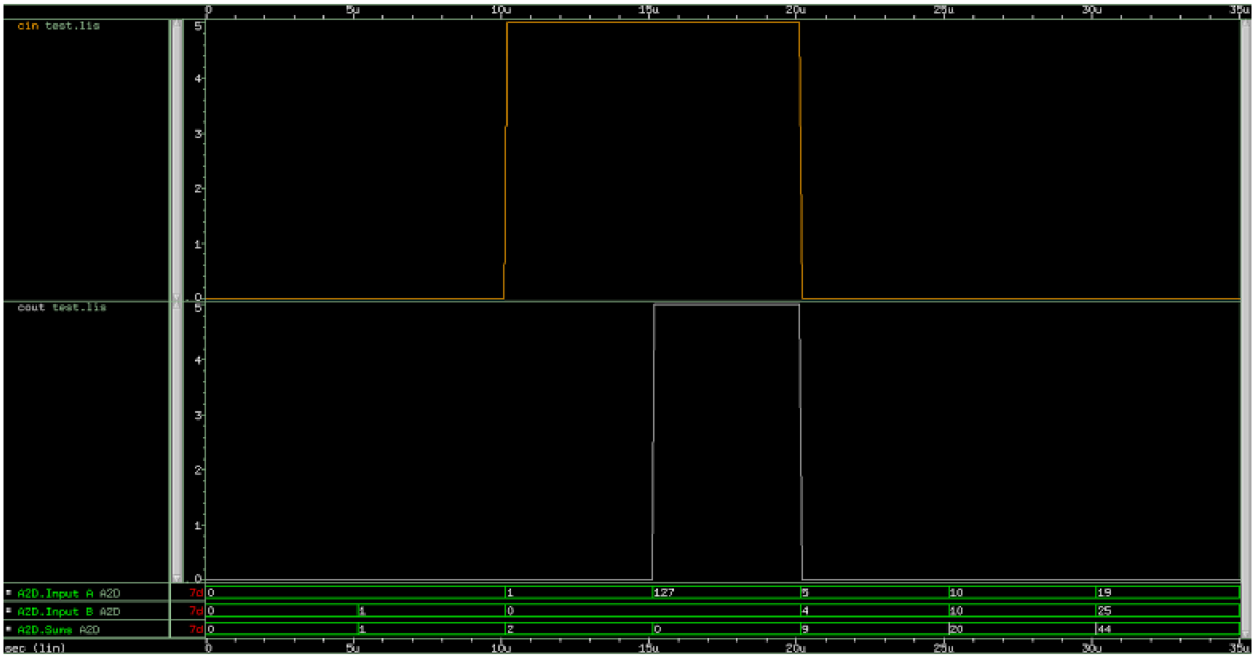
6 Bit Parallel In/Parallel Out



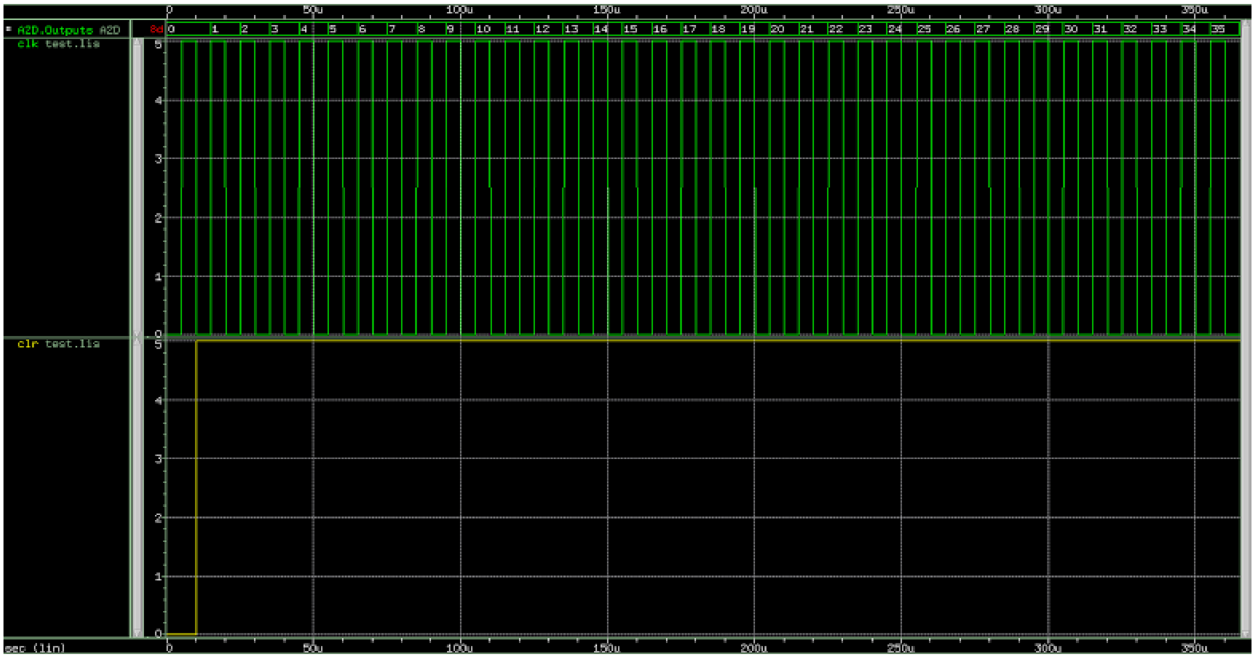
6 Bit Shift Register



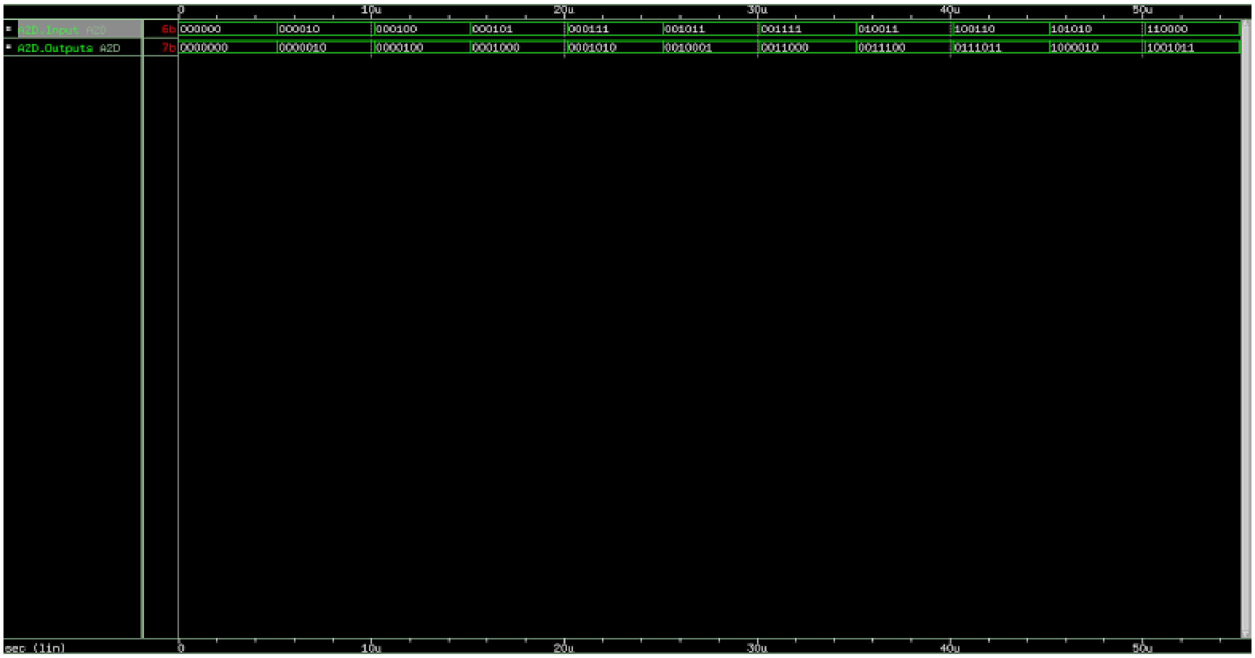
7 Bit Full Adder



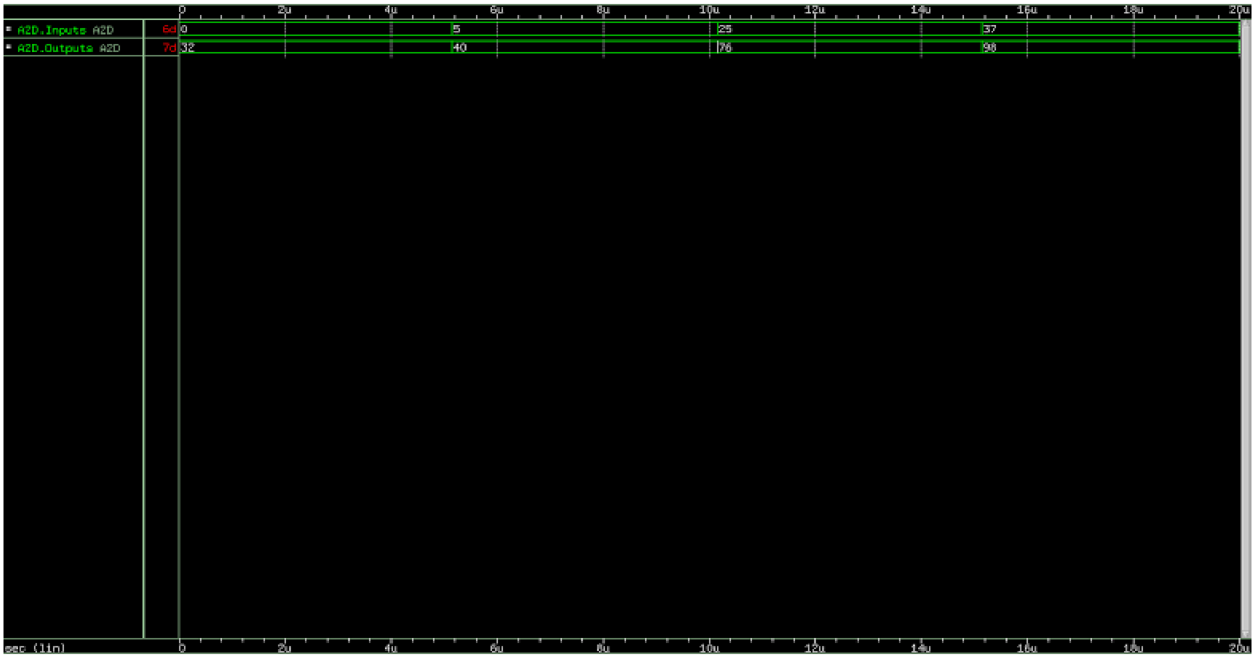
8 Bit Counter



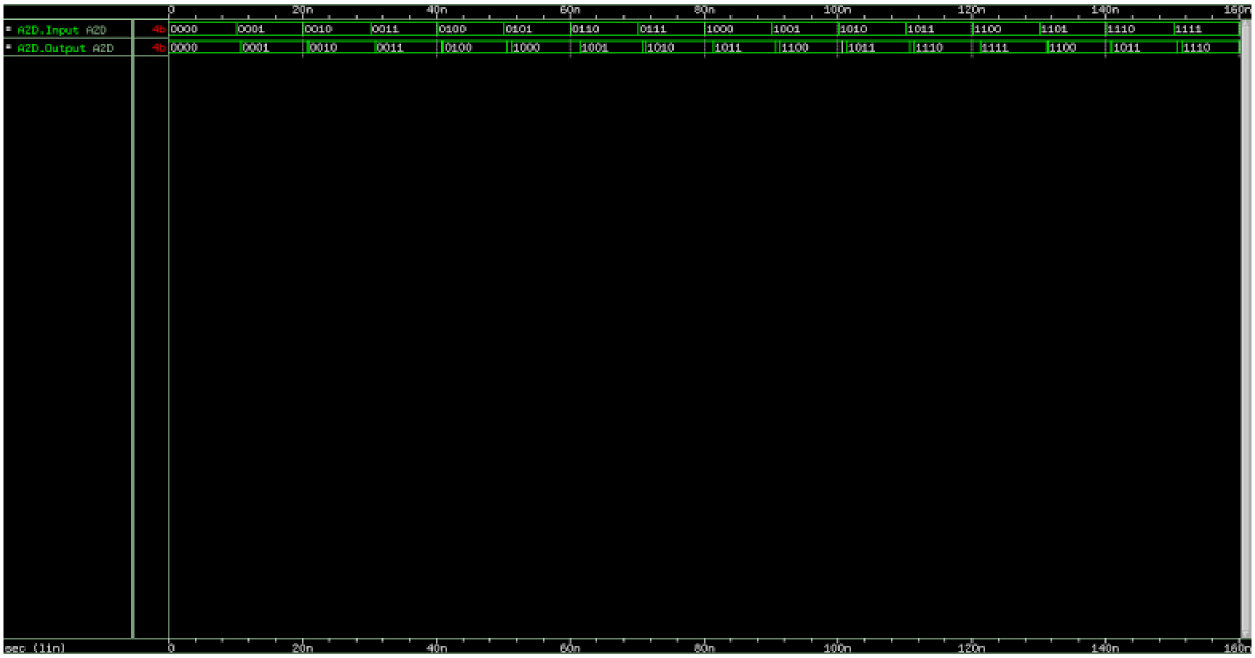
Binary to BCD



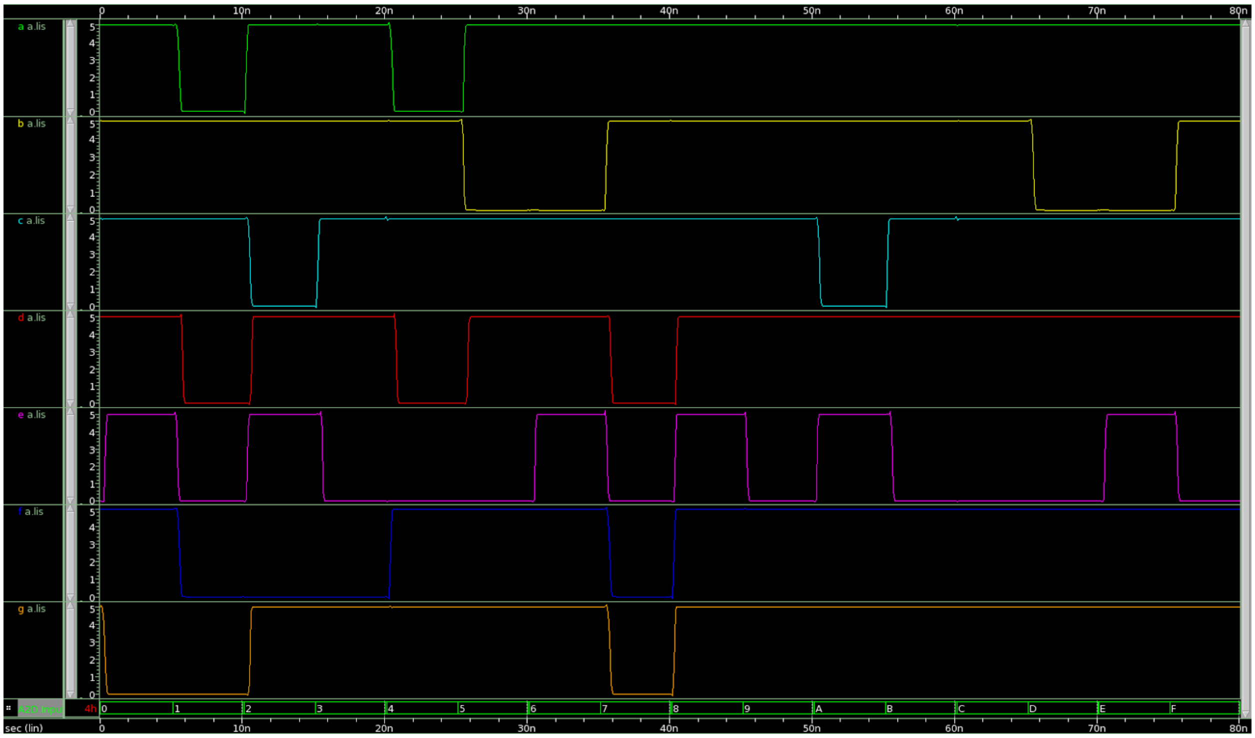
Celsius to Fahrenheit



Conditional Adder



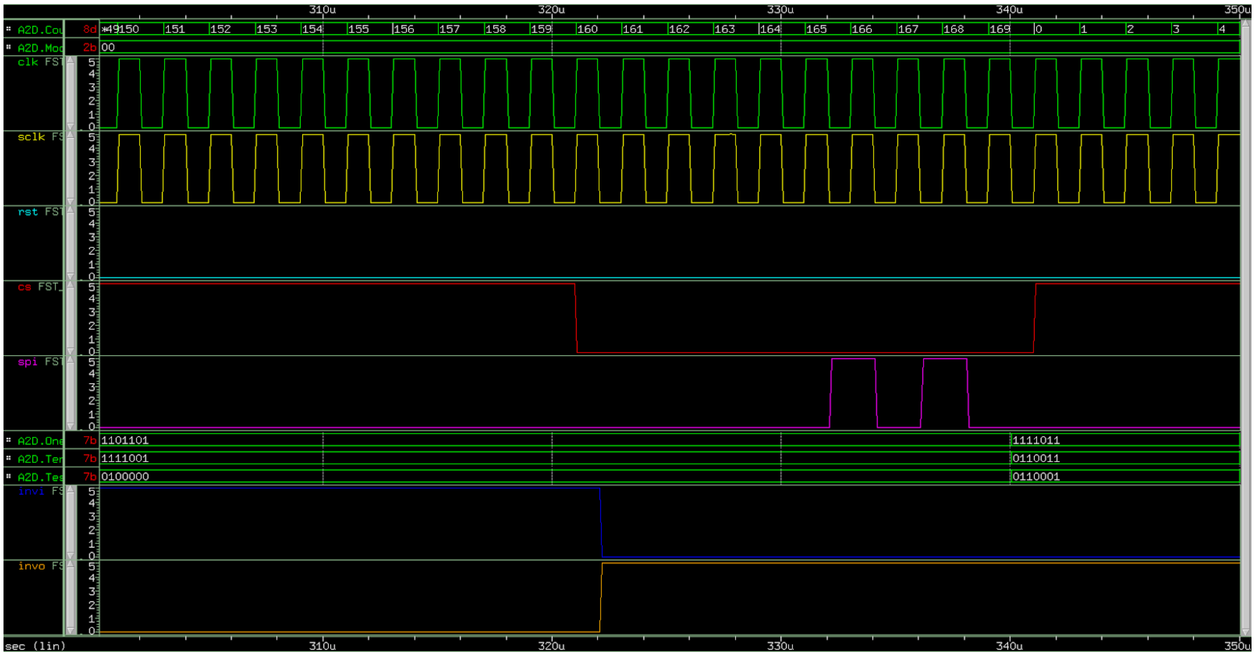
7 Segment Driver



In order to reduce the number of images in this document only a select number of images were chosen to display the appropriate testing of Mode0 and Mode1. All of the other test vectors are included in a zip file.

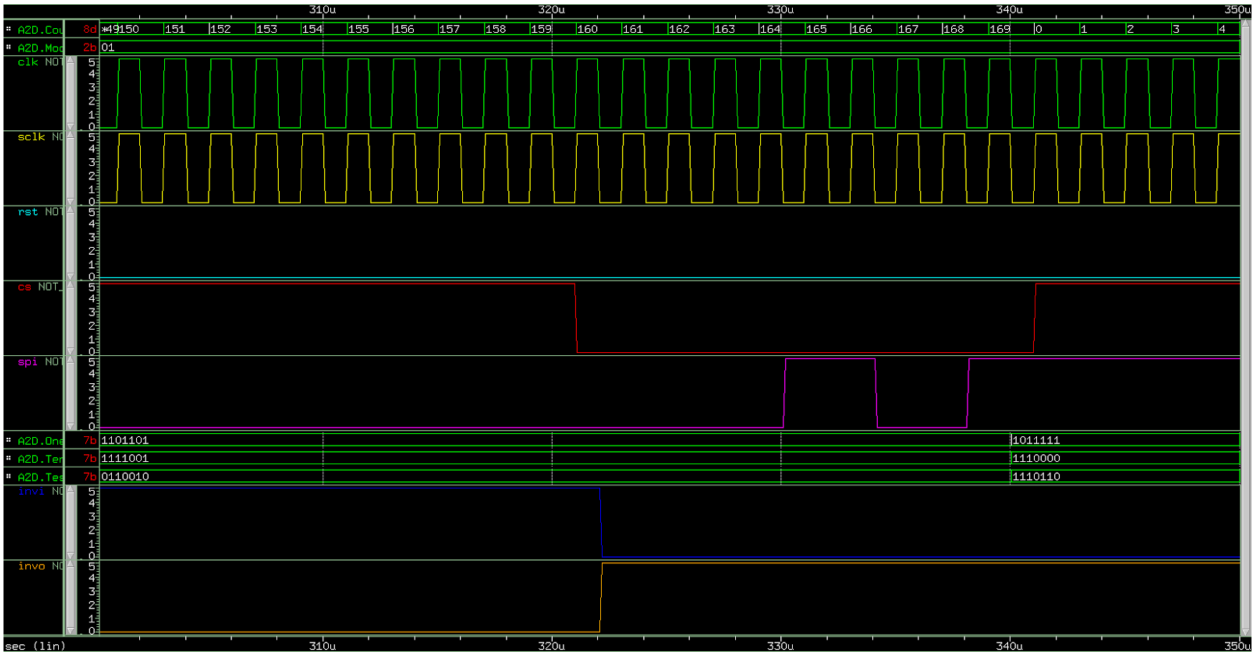
Test Vector 1 (mode0 = 0, Mode1 = 0):

Shows testing with temperature 10^{0}C



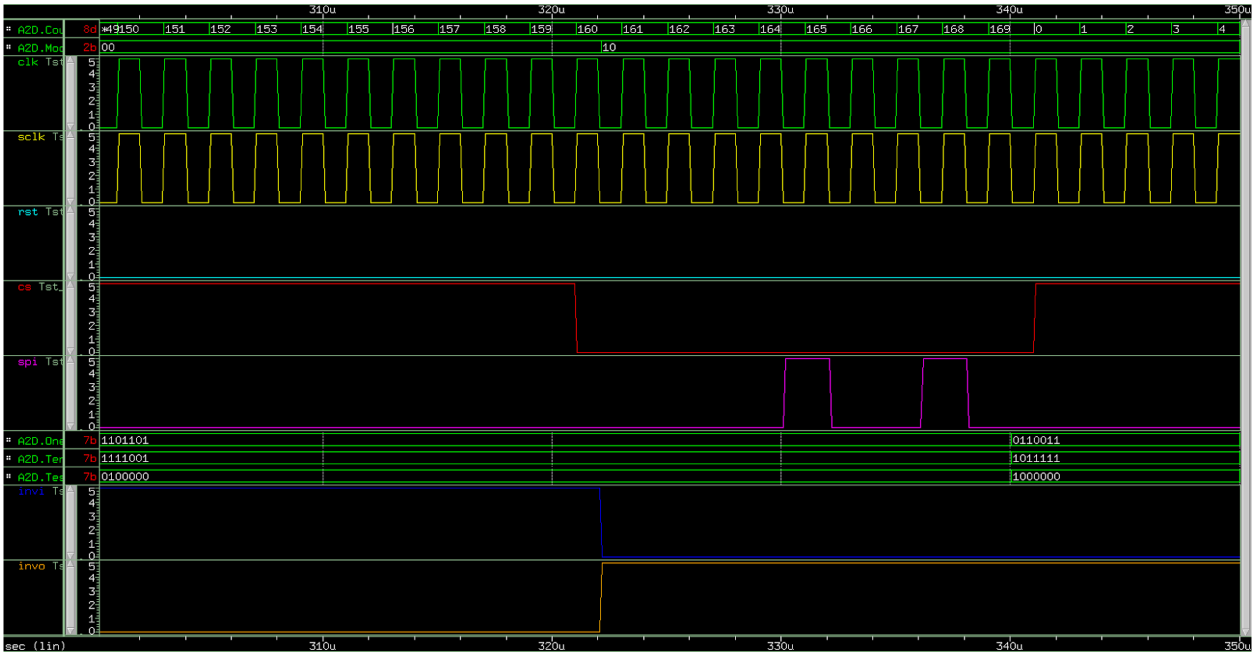
Test Vector 2 (Mode 0 = 1, Mode1 = 1):

Shows testing with temperature 15^{0}C



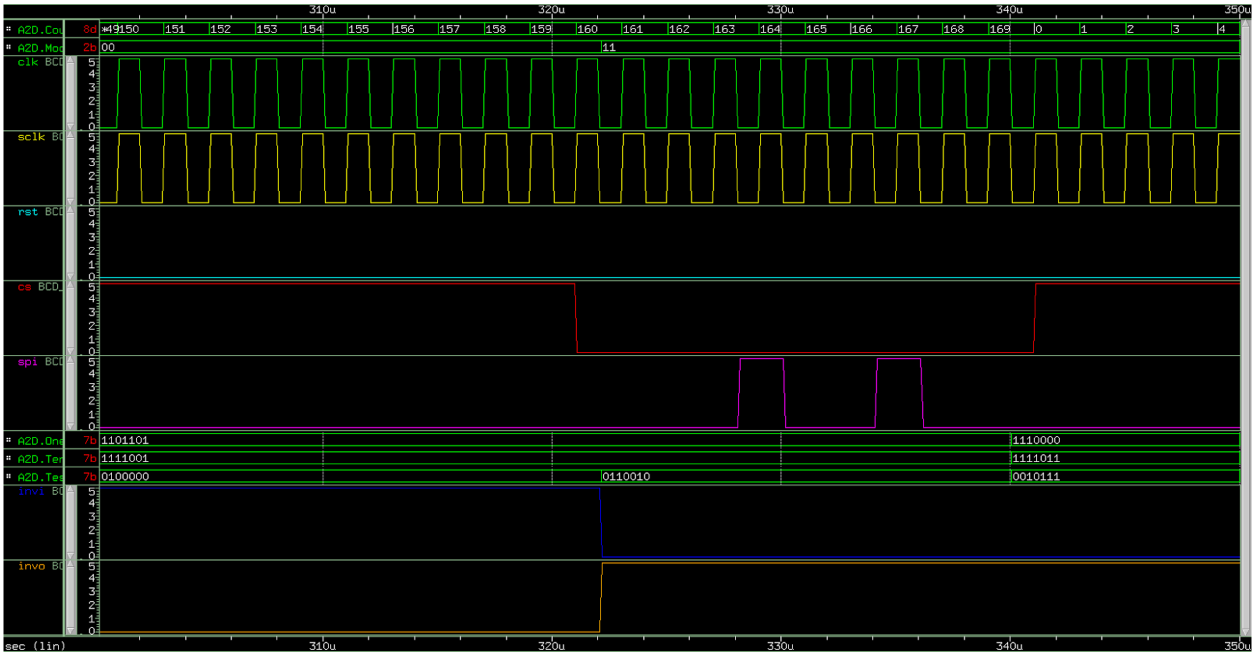
Test Vector 3 (Mode0 =0, Mode1 = 1):

Shows testing with 18^{0}C



Test Vector 4 (Mode0 = 1, Mode1 = 1):

Shows testing with 36^{0}C



VI. CONCLUSIONS

Overall this project was difficult but rewarding. The logic design was rather intuitive but layout of said logic was at times miserable given the numerous power issues encountered. Better floorplanning may have helped prevent such issues seeing as how a simple adjustment in wire size for the clock was the solution.