Refrigerator Thermometer

Keeping Food at Safe Temperatures

Jacob Monks
Rafal Astifo
Nathan Gilmer
Joseph Asteefan

Introduction

- A Thermometer is an essential tool used to measure temperature.
- They mainly consist of having a sensor to measure temperature and some form of visual to display the temperature reading.
- Many applications, such as refrigeration, rely on thermometers to ensure the appropriate temperatures are maintained.
- For our application, a standard thermometer system was developed for refrigerators to help indicate hazardous temperature readings.

Methodology

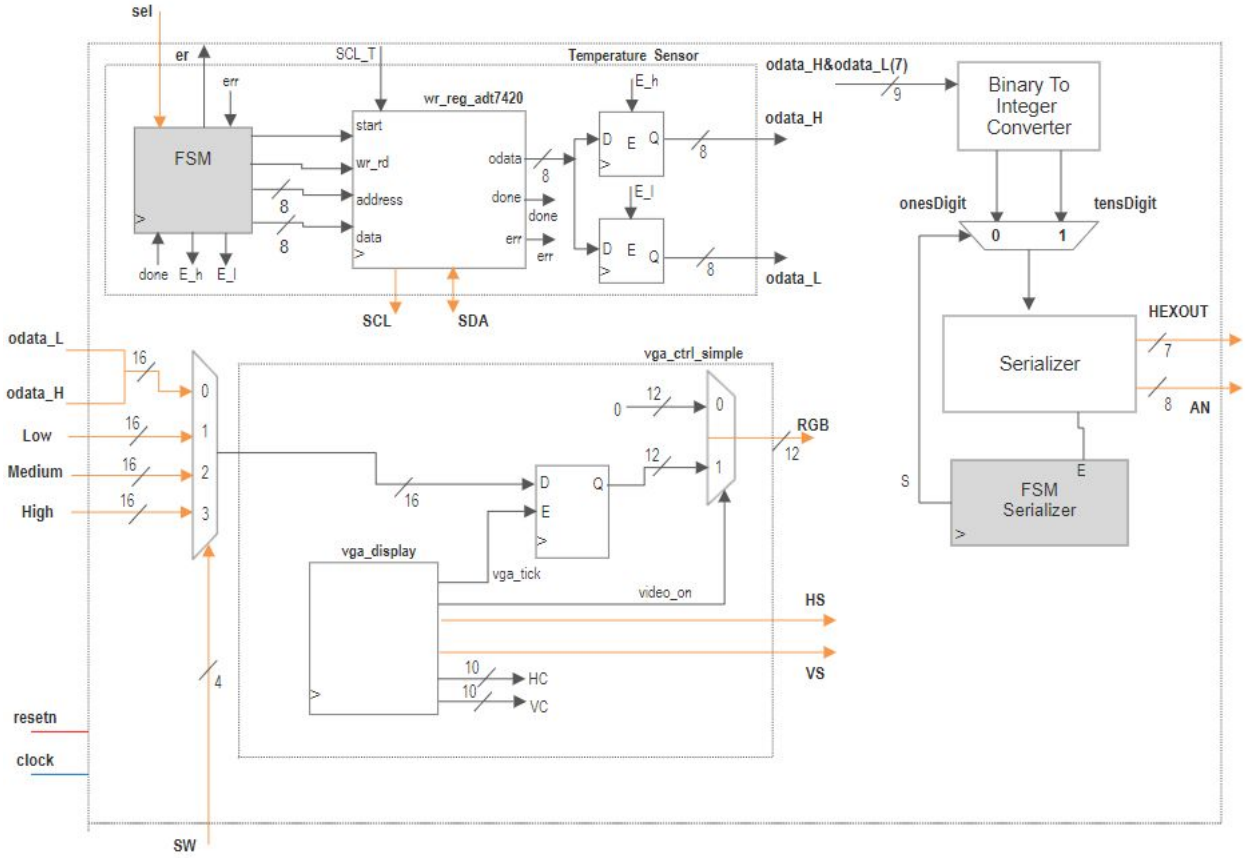
- The Nexys-A7 board readily comes preinstalled with a temperature sensor.
- The output signal readings from the sensor are to be displayed onto a monitor through the use of VGA interfacing.
- The visual is developed in an image of a horizontal bar that increases and decreases in accordance with the temperature.
- The temperature range is indicated through the use of colors:
  - Red = Overheating (hotest)
  - Blue = Freezing (coldest)

Experimental Setup

- Application was created through the use of VHDL.
- Temperature sensor, VGA controller, and Seralizer are each created having their own set of components, and then interfaced within the top file.
- The VGA display will display the output readings from the temperature sensor, and a set of switches to demonstrate each temperature range:
  - Blue (Freezing): < 28℃
  - Cyan: 28-31℃
  - Green (ideal temperature): 31-34 ℃
  - Orange: 34-37℃
  - Red (Overheating): > 37℃

Block Diagram

Inputs/Outputs	Ports
Temperature	Low, Medium, High
Temperature Data Select	SW[0-3]
Temperature Range Select	SW[4]
VGA RGB	R[0-3] G[0-3] B[0-3]
VGA HS	HS
VGA VS	VS
Temperature Sensor SCL	SCL
Temperature Sensor SDA	SDA
7 Segment Display	HEXOUT[6-0]
Serializer Out	AN[7-0]



Temperature Sensor

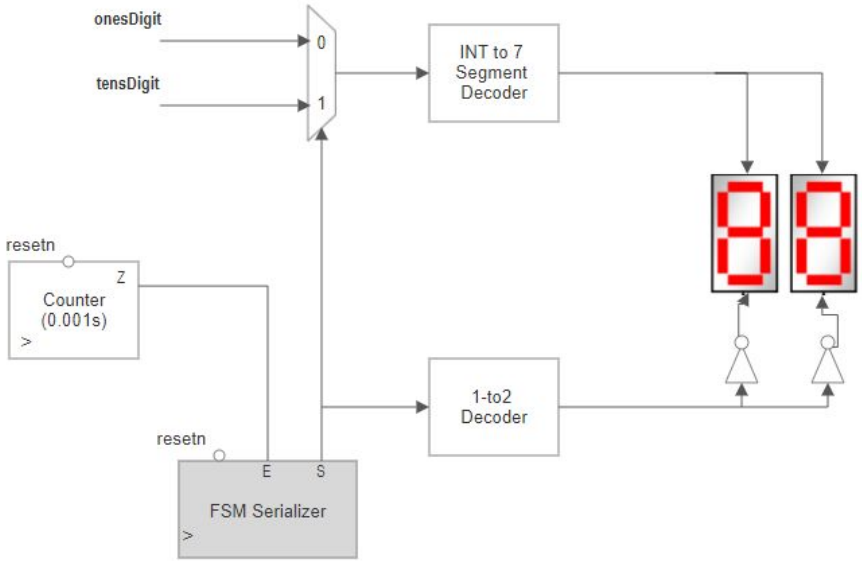
- The preexisting basic controller code for the temperature sensor was used.
- The code is set to write/read to/from the temperature value registers (8-bit):
  - Register 0x00: Temperature value MSB
  - Register 0x01: Temperature value LSB
- Resulting temperature measurement is recorded into a 16-bit, two's complement, data format.
- Data is set to refresh at 32 ms per reading.
- ADT7420 reads temperatures between -40℃ to 150℃.

VGA Display

- The SIMPLE version for the VGA controller 12-bit core code was used.
- A 12-bit color is outputted, with each color having 4 bits (RGB).
- The horizontal bar diagram is placed in the center of the display, and each color is fixated at a specified range.
- A monitor with a refresh rate of **30hz** and a resolution of **640x480** was used to display the results.
- The color scheme for the thermometer was defined to resemble a typical thermometer.
- It consists of 5 blocks: Blue, Cyan, Green, Orange and Red. Each block has a height of 96 rows and a width of 320 columns.

Seven Segment Display Serializer

- A seven segment serializer is used to display the temperature. This value is truncated to its whole number.
- Every millisecond, the serializer switches between displaying the ones digit and the tens digit.
- This is so fast that we see both digits on at the same time.
- The temperature was converted to an integer. This integer divided by 10 to find the tens digit, then the mod operator was used to find the ones digit.




```
tempInt <= conv_integer(unsigned(Output_Temp(15 downto 7)));
tens <= tempInt / 10;
ones <= tempInt mod 10;
```

Process

1. The temperature sensor (continuously) reads the temperature, converts the resulting data into Celsius and stores it into the temperature value registers (0x00, 0x01).
2. The 9 MSB outputted from the following registers are then sent to the seven segment display, while the whole 16-bit temperature is sent to the VGA controller, to display the resulting temperature.
3. The data sent to the VGA control defines the output of the display through two separate processes that executes the data through the use of if-statements:

    A. Block Height Process: This is explained in the next slide.
    B. Color/Block Placement Process: If the temperature results are within specific ranges, and if the horizontal and vertical pixels are within a specific range, then the color defined for the temperature range will appear.

Block Height Process

- Considering the VGA has 480 rows, the 5 colors or blocks are equally divided to have 96 rows each
- The red and blue blocks do not gradually increase, as for this application it is not necessary.
- The middle blocks all have a temperature range of 3℃.
- 3 / 96 = 0.03125℃ per row. This temperature is equivalent to having “0000000000000100”, we’ll say “4”, for the 16-bit temperature value.
- Since we needed to increase or decrease the bar every 4 bits, the temperature reading was subtracted by the minimum temperature value for that block, then divided by 4 to find that block’s height.
- When setting the v-count parameters, the base height of the block was subtracted by the block’s height. This showed the 0.03125℃ per row increase on the VGA display.

```
if (Output_Temp > "0000111000000000" and Output_Temp < "0000111110000000" ) then --cyan block from 28-31 deg
    tempSUB <= Output_Temp - "0000111000000000"; --subtracting temp sensor reading for block's base temp i.e. temp sensor-28
    cyan_block_h <= "00"&tempSUB(15 downto 2); --dividing number by 4 by bit shifting 2 bits right
end if;
```

Expected Results (Simulation)

SW	Temperature Range (℃)	V-count Range	H-count Range	RGB	Thermostat Output Image
0010 (L)	T < 28	384 to 480	160 to 480	Blue	Blue block
0100 (M)	31< T < 34	288 to 192	160 to 480	Green	Blue, Cyan, Green,
1000 (H)	T > 37	96 to 0	160 to 480	Red	Blue, Cyan, Green, Orange, Red

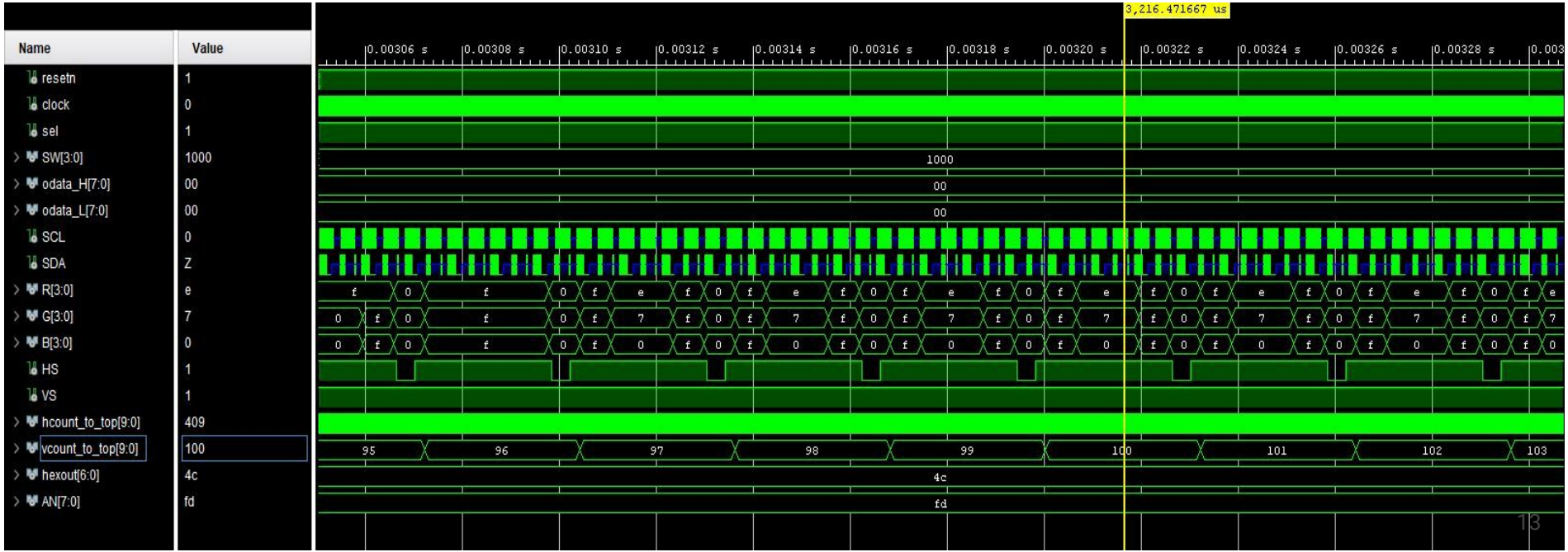
**Remaining pixels set to white

Results High Temperature (Simulation)

Red Block Range:

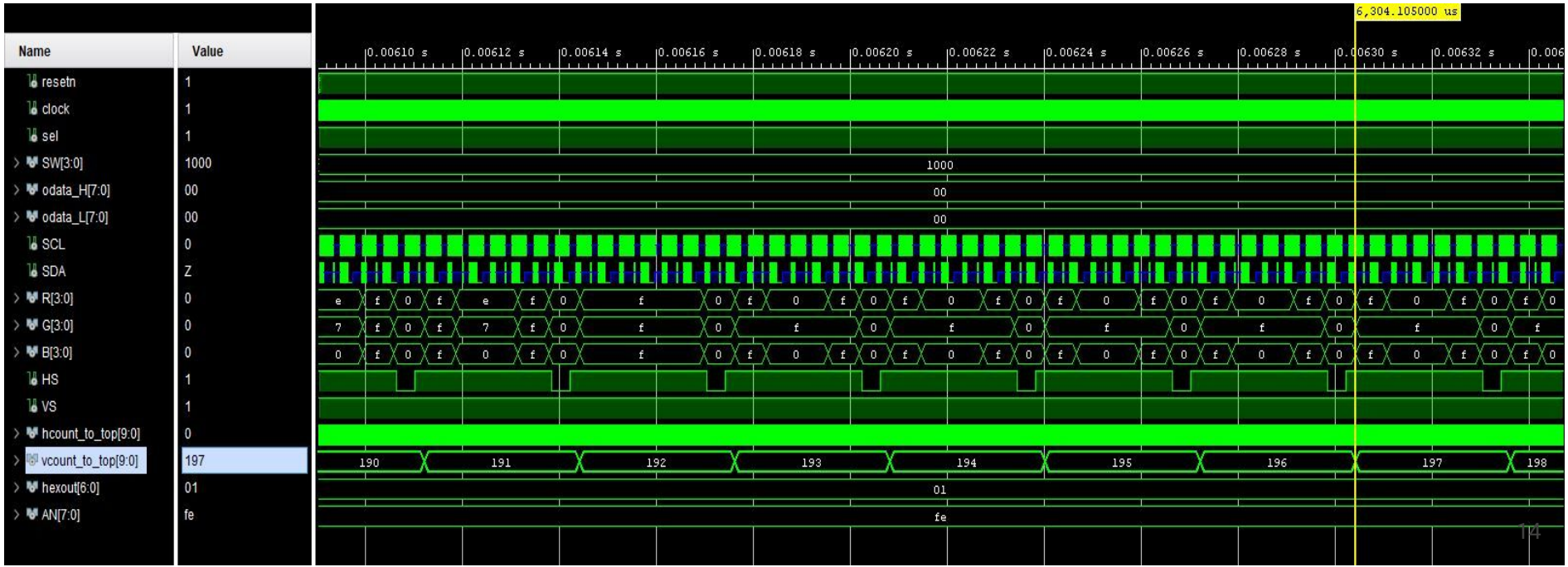
Results High Temperature (Simulation)

Orange Block Range:



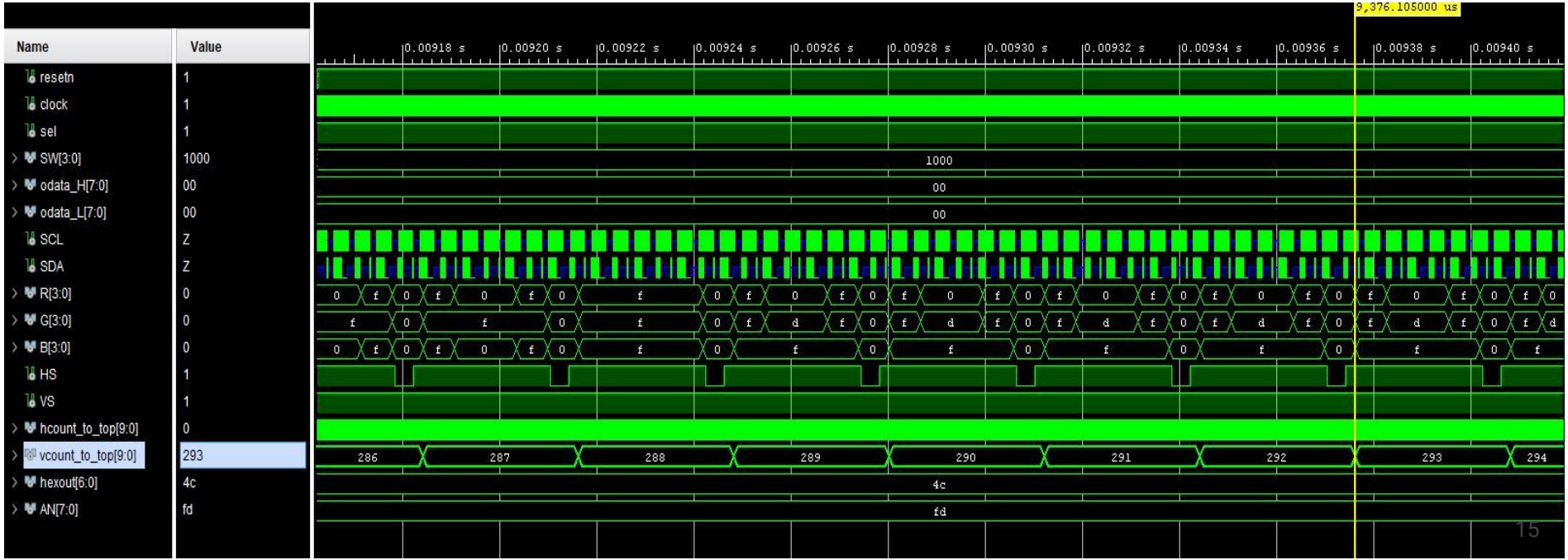
Results High Temperature (Simulation)

Green Block Range:



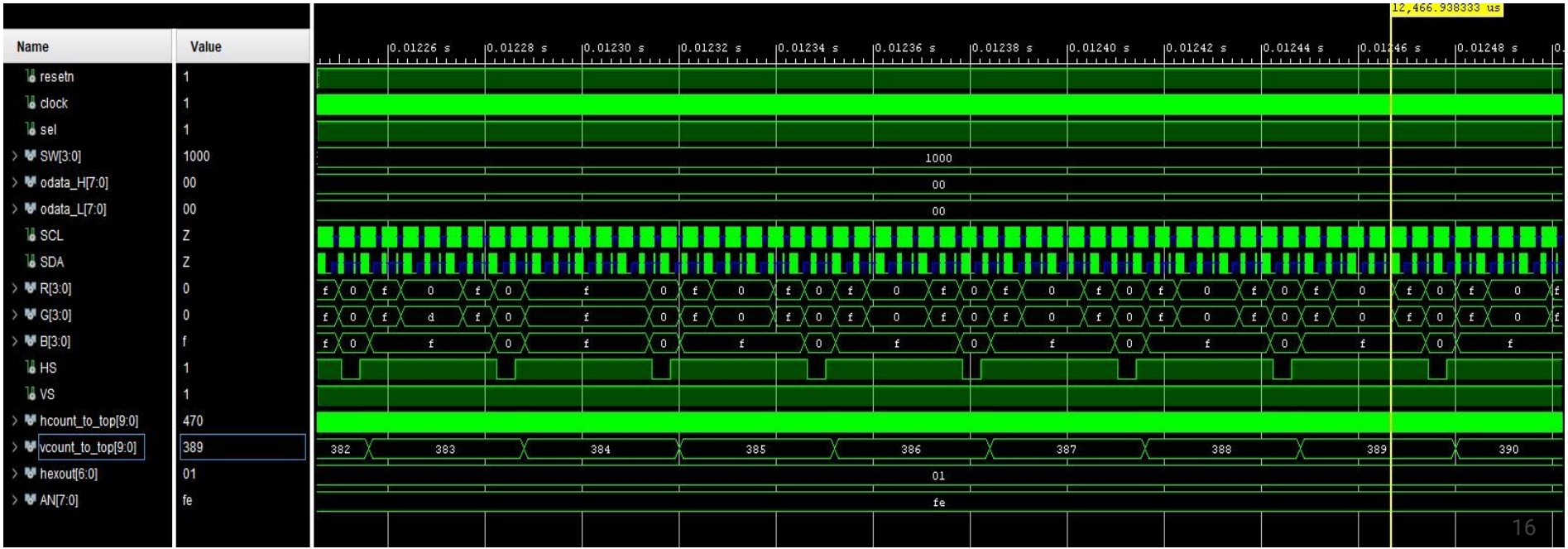
Results High Temperature (Simulation)

Cyan Block Range:



Results High Temperature (Simulation)

Blue Block Range:



Live Demonstration

Conclusion/Future Improvements

- Temperature range for this project was determined to demonstrate the overall workings of our application, and may be adjusted accordingly for future uses.
- The display of the temperature value can be modified to appear on the display rather than the board to better improve the overall thermometer.
- Showing the color density change as it increases/decreases to show the different shades for each temperature block can better demonstrate the overall temperature.
- The temperature sensor implemented with the board may be accurate but a better sensor can be used to decrease some of the fidgeting noticed from the sensor.
- Possibly including noise, or warning texts to the display to better fit the needs of the user.

References

- "Are You Storing Food Safely?" *U.S. Food and Drug Administration*. 9 February, 2021. Are You Storing Food Safely?
- Brown, Arthur. "Nexys A7 Reference Manual." *Nexys A7 Reference Manual - Digilent Reference*, Digilent, Nexys A7 Reference Manual
- "Celsius to Fahrenheit (°C to °F)." *Metric Conversion*, Wight Hall Ltd., 2020, www.metric-conversions.org/temperature/celsius-to-fahrenheit.htm
- Llamocca, Daniel. *VHDL Coding for FPGAs*, www.secs.oakland.edu/~llamocca/VHDLforFPGAs.html