ELEC 3500 Alarm Clock Report

Thomas Shih

March 31, 2020

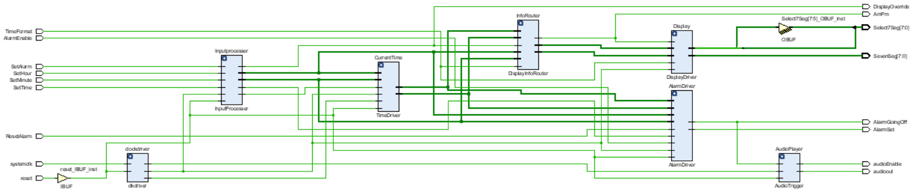

Figure 1: Overall Flow of Design

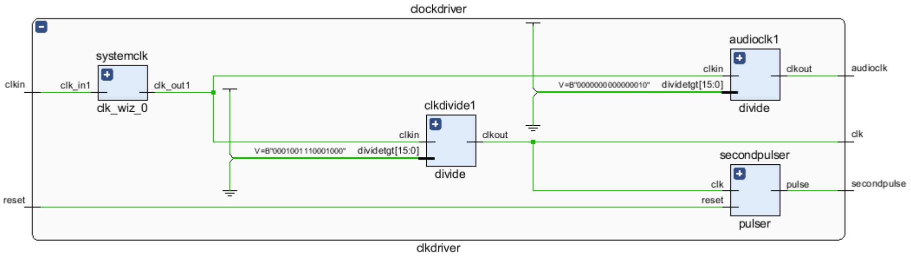

Figure 2: ClockDriver

0.1 Overall Design Philosphy

The alarm clock code was designed to be modular such that additional functionality can be passed in. This required more resources to separate the information processing portion of the code from the routing portion of the code. The alarm clock consists of five main processing units; which are critical to the functionality of the alarm clock, and two auxiliary processing units; which is placed to enable additional functionality.

0.1.1 ClockDriver

The ClockDriver module is the module that takes in the system clock (100MHz) and converts it into a more usable system clock (1kHz). The clock driver also outputs a enable pulse every one second for 1 ms. Additional functionality if provided in forms of a separate Audio Clock for music control, this audio clock frequency can be set to any integer division of the system clock. Additional clocks or pulses, if needed, can be wired out of this module easily as the clocks and pulses are instantiated with prebuilt modules using parameter instantiation functionality.

0.1.2 InputProcesser

The InputProcesser module takes the input from the onboard buttons, denounces the serial inputs, and then outputs synchronous data parallel out to the desired target, weather it be to change the current time, or to change the

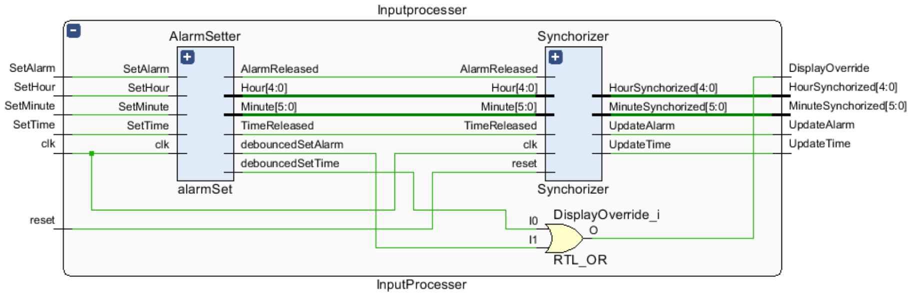

Figure 3: InputProcesser

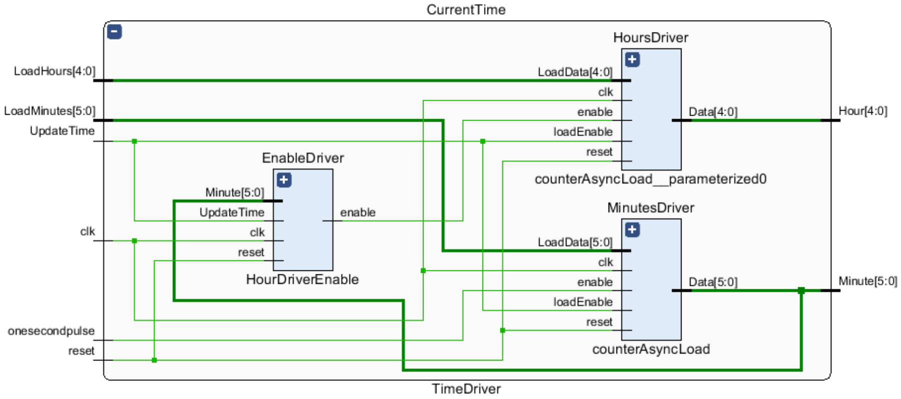

Figure 4: TimeDriver

alarm time. Additional function is provided in such a way that additional input targets can also be added through the use of preexisting parameterized modules.

0.1.3 TimeDriver

The TimeDriver keeps track of time in 24 hour format. There is functionality to load in a new time.

0.1.4 DisplayDriver

The DisplayDriver takes any hour and minute input, performs processing and multiplexing, and displays it on four seven segment LEDs. An additional seven segment display is used to display Am Pm.

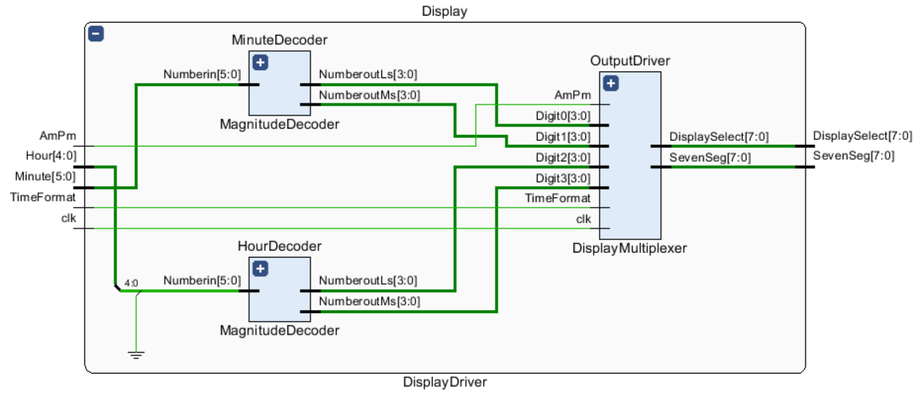

Figure 5: DisplayDriver

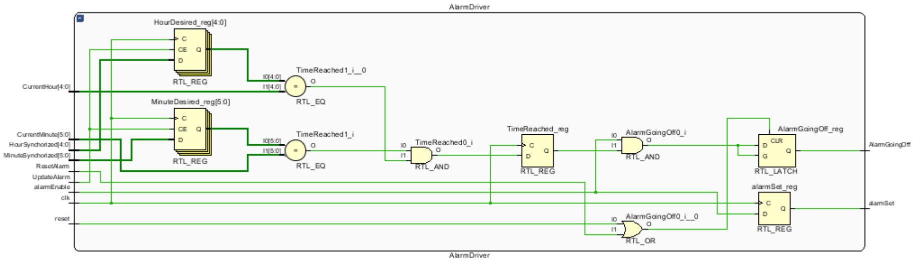

Figure 6: AlarmDriver

0.1.5 AlarmDriver

The AlarmDriver compares the current time output by the TimeDriver, and the set AlarmTime, should they match, the AlarmDriver will output a pulse indicate the time has reached. The Alarm Time can be passed in, as well as reset.

0.1.6 AudioPlayer

The AudioPlayer takes any hexadecimal data file, and creates a ROM, using each byte of the hexadecimal file as a PDM coefficient to set the density of pulses. The audio can be played without any human noticeable distortion, fidelity compared to mp3 audio is the same, if not superior.

0.1.7 InfoRouter

The InfoRouter takes in the time from InputProcesser, and also the CurrentTime, and routes the correct/desired information to the time driver.

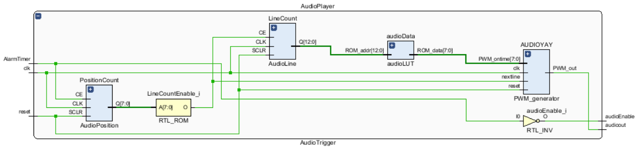

Figure 7: AudioPlayer Beta

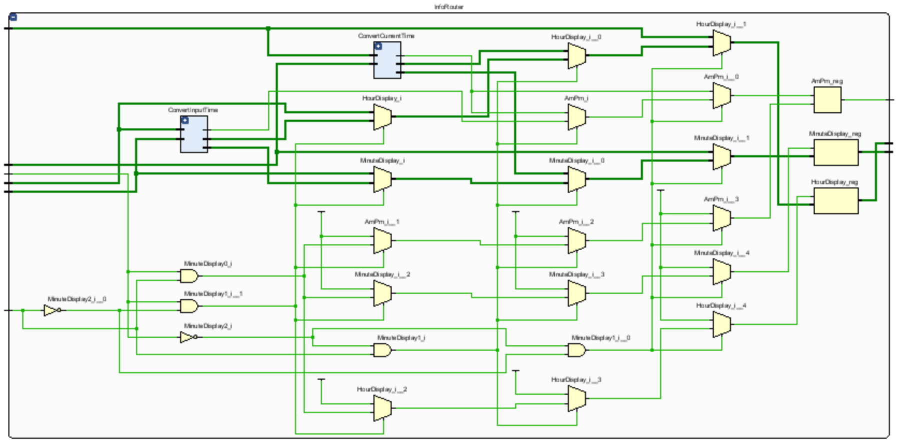

Figure 8: InfoRouter

0.2 Conclusion

An alarm clock was built and the clock is very pretty. The audio is very nice.