

# Modules

Martin Mihle Nygaard ([martimn@ifi.uio.no](mailto:martimn@ifi.uio.no))

- a) I first implemented `bin2ssd` as an entity, which worked fine, but after feedback from my first attempt I've now changed it to a function wrapped in a package. See `bin2ssd_pck.vhd` for implementation, and `tb_bin2ssd.vhd` for a working test bench<sup>1</sup>.
- b) I chose to alternate the displays at at least 50 Hz. I then need to calculate the number of  $100 \times 10^6$  Hz cycles fit in 50 Hz, which is given by

$$\frac{100 \times 10^6 \text{ Hz}}{50 \text{ Hz}} = 2 \times 10^6.$$

The appropriate amount of bits is given by  $\lfloor \log_2 2 \times 10^6 \rfloor = 20$ , I round down get a faster (rather than slower) counter. More precisely, this gives a refresh rate of  $\approx 95$  Hz.

See `seg7ctrl.vhd` for implementation, and `tb_seg7ctrl.vhd` for a working test bench.

Setting either inputs `d0` or `d1` to 0000 will clear the displays.

A possible schematic of the entity `seg7ctrl` is shown in figure 1, at least as I imagine it.

- c) See `seg7ctrl_arch.vhd`, `self_test_unit.vhd`, `self_test_system.vhd` for implementations, with a test bench in `tb_self_test_system.vhd`.

First, I misunderstood this as “light up display for one clock cycle once per second”, which wouldn't possible be visible to the human eye; and I was somewhat confused on how to factor the code and components, too. Hopefully it's up to spec now, after some guidance from helpful lab supervisors.

- d) It's alive! See `constraints.xdc` and `self_test_system.bit` for constraints and bit file, respectively. I was a bit unsure how to include the timing and utilization reports (and had no supervisor on hand), but I've included `timing.rpx` and `utilization_report.txt`, which Vivado allowed me to export. I've also included a video as proof of a functional, programmed board.

I read the “hidden” message as: *WELL donE YoU ArE good.*

---

<sup>1</sup>Unless Canvas screws up the filenames again ...

