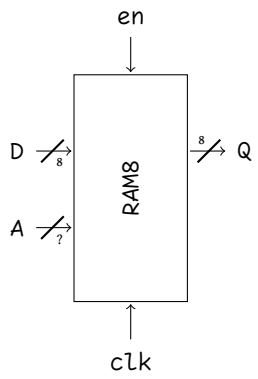
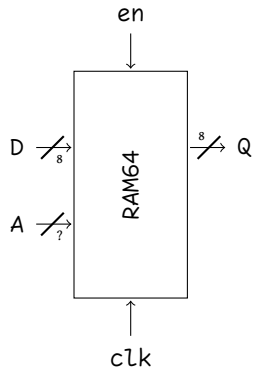
Use *Digtal* to construct the following **synchronous** gates.

1. Using only the built in `Byte` (or `Register` set to 8 bits), and whatever combinatorial gates you want, build a `RAM8`, a memory with 8 addressable bytes. The address line(s) determine which of the eight bytes are presented on `Q` and which byte will latch `D` (on a rising clock signal while `en` is asserted).



2. Using as many of your `RAM8` as you like, and whatever combinatorial gates you need, build a `RAM64`, a memory with 64 addressable bytes. The address, load and clock serve the same sort of purpose as in `RAM8`. **Note:** think about the *number* of address lines coming in to the `RAM64` and *how many* each `RAM8` needs to do its job.



Deliverables: When all the circuits are tested and working, put them into a `git` repository (no need for a `README` here) and submit them through `Gitea` at `https:cs-devel.potsdam.edu`.