* Class itinerary: Basic circuit -> Half adder -> Full Adder -> Multiplexer -> Decoder -> Flip-Flops -> Registers -> Finite State Machine (FSM).
* **FPGA:** Field-programmable gate array is an integrated circuit (IC) that can be programmed in the field after manufacture. FPGAs are similar in principle to, but have vastly wider potential application than, programmable read-only memory (PROM) chips.
* **VHDL:** Very high speed integrated circuits hardware description language. Other description languages: Verilog, Ruby… VHDL can be used to create a hardware (file to configurate a FPGA, for example). Can also be simulated.
* **Testbench:** defines the external settings for the circuit (inputs and expected outputs).
* **Entity:** Input and output pins. The circuit can have only one entity.
* **Architecture:** Circuit’s functionality. The circuit can have many architectures.
* **Complete circuit:** The circuit is composed of libraries, entity and architectures.
* **Quartus How to:** File -> New -> VHDL File -> OK -> Start Compilation -> File -> New -> University Program VWF -> OK -> Insert Node or Bus -> Node Finder -> List -> >> -> OK -> Edit -> Value -> Overwrite Clock -> Period 10.0, 20.0, 40.0 -> OK -> Run Functional Simulation.
  + The **hierarchy** is created when compiling the code.
  + To give **arbitrary values (Waveform)**, use your mouse on the input waves to select a specific space. Then use *Arbitrary Values* button to add an arbitrary value.
  + USE THE SAME NAME FOR THE VHDL FILE AND ITS ENTITY! (few hours lost here).
* **MUX 2x1:** If input s = 0, output m = input x. If input s = 1, output m = input y.
  + VHDL **structural**: m <= (not s and x) or (s and y);
  + VHDL **behavioral** (when/else): m <= x when s = ‘0’ else y;
* **Decoder:** Converts one representation of data into another, like analog to digital, or 4 bits into 7 bits to represent 4 bits in hexadecimal in a display.
* **Transistor:** Switch associated with electric current.
* **Flip-flops, latches and registers:** Made of usually 2 transistors. Stores one bit of information, one transistor for each state, when one is on the other is off.
  + **Latch** is asynchronous ie. the outputs can change as soon as the inputs do.
  + Whereas a **flip-flop** is synchronous ie. the change of outputs is determined by the clock. Can be used to build a **register**, which is capable of storing more than just one bit (8, 16, 32…).
* **Process:** Defines a sequence of commands to be executed in behavioral VHDL. The only place you can use *if*, *then* and *else*. It’s invoked when ANY! of the signals in its sensitivity list (“parameters”) changes.
* **Finite-State Machine (FSM):** Defined by a list of its finite states, its initial state and the conditions for each transition. Can be in exactly one of a finite number of states at any given time. May change its state (transition) in response to external inputs. Examples: Vending machines, elevators, combination locks. In a Moore FSM the output depends only on its current state.