Notes on Verilog

Lexical elements

- **escaped identifiers**: begins with \, ends with white space; treated the same as a nonescaped identifier
- **attributes**: directive

```
(* full_case, parallel_case *)
case (foo)
```

- port connection: ordered or named

Data types

- **nets**: wires; don't store value (except for **trireg**);
  - value is determined by the value of its drivers (e.g. contasgn or gate output)
  - default initial value is `z` (**trireg** will default to `x`, with the given strength)
  - **supply0**, **supply1**:
  - **tri**, **triand**, **trior**, **trio**, **tri1**:
  - **uwire**, **wire**:
  - **wand**, **wor**:
- **variables**: **reg**, **time**, **integer**, **real**, **realtime**
  - **reg**, **time**, **integer**: default is `x`
- **vectors**: multi-bit **net** or **reg**
  - bit-select, part-select, etc. possible
- **strengths**: can be specified for nets
  - **charge strength**: only for **trireg** nets (**large**, **medium**, **large**)
  - **drive strength**: used only when the net is driven by contasgn; e.g. `wire (strength0, strength1) x = rhs;`
- **value resolution on multiple drivers**:

Memories

- memory: one-dimensional array with elements of type **reg** (e.g. `reg mema [1:10], reg [3:0] memb [3:0]`)

Parameters

- **parameter**:
- **localparam**:
- **specparam**: parameters for timing/delay values

```
specify
  specparam tRise_clk_q = 150, tFall_clk_q = 200;
  specparam tRise_control = 40, tFall_Control = 50;
endspecify
```

Expressions

- **concatenation, replication** (`{}, {4{...}}`)
- **equality**
  - **logical equality** (==): if any bit contains `z` or `x`, the result will be `1'bx`
  - **case equality** (===): `z` and +x+ will be literally compared; result is always 0 or 1

Verilog scheduling semantics

- **type of events**
  - **update event**: value change in net or variable or named event trigger
  - **evaluation event**: processes are sensitive to update event
    - when an update event is executed, all processes sensitive that event are evaluated in arbitrary order
- **scheduling an event**: putting an event on the queue
- **event queues**
  - **active event**: occur at the current simulation time and can be processed in any order
  - **inactive event**: occur at the current simulation time but processed after all active events are processed (for `#0` delays, callback procesures from `vpi_register_cb(cbReadWriteSynch)`)
  - **nonblocking assign update event**: evaluated during some previous simulation time but assigned
  - **monitor event**: after all active/inactive/nonblocking
  - **future event**: occur at some future simulation time
    - **future inactive event**
    - **future nonblocking assignment update event**

Multi-driver

- same wire driven from multiple processes

Latches vs FFs

- **level-sensitive latch**

```
always @(g or d)
  if (g) q <= d;
```

- **edge-triggered FF**

```
always @(posedge clk)
  q <= d;
```