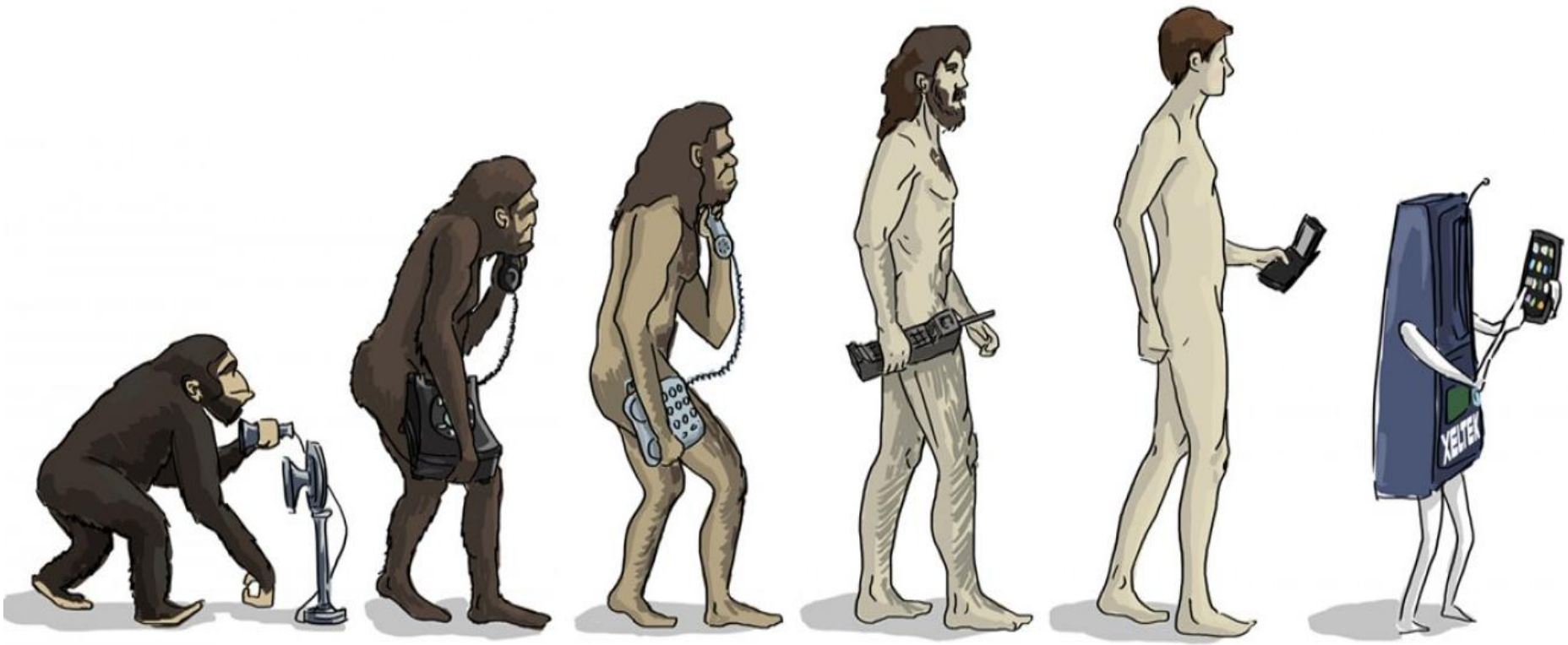
Mobile Phone evolution



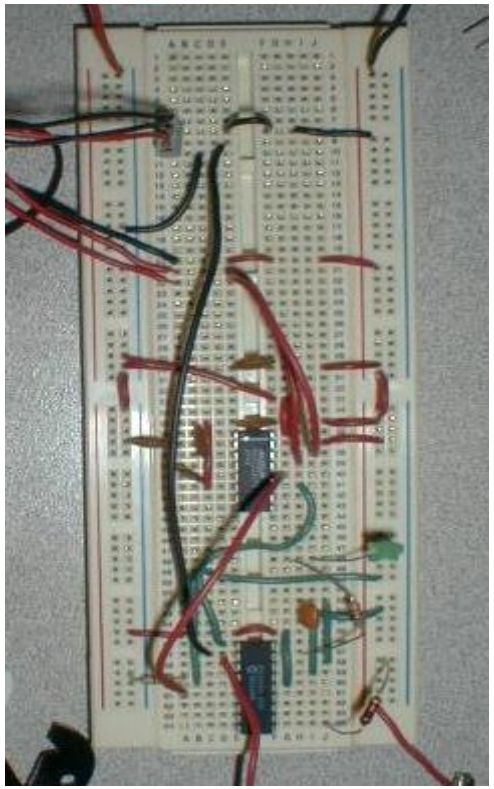
Quotes from Martin Cooper

- Well, we knew that someday everybody would have a [cell] phone
- Phones have gotten so complicated, so hard to use, that you wonder if this is designed for real people or for engineers.

UVM for verification similar?

Origins of “testbench”

Signal Generator + Supply etc.

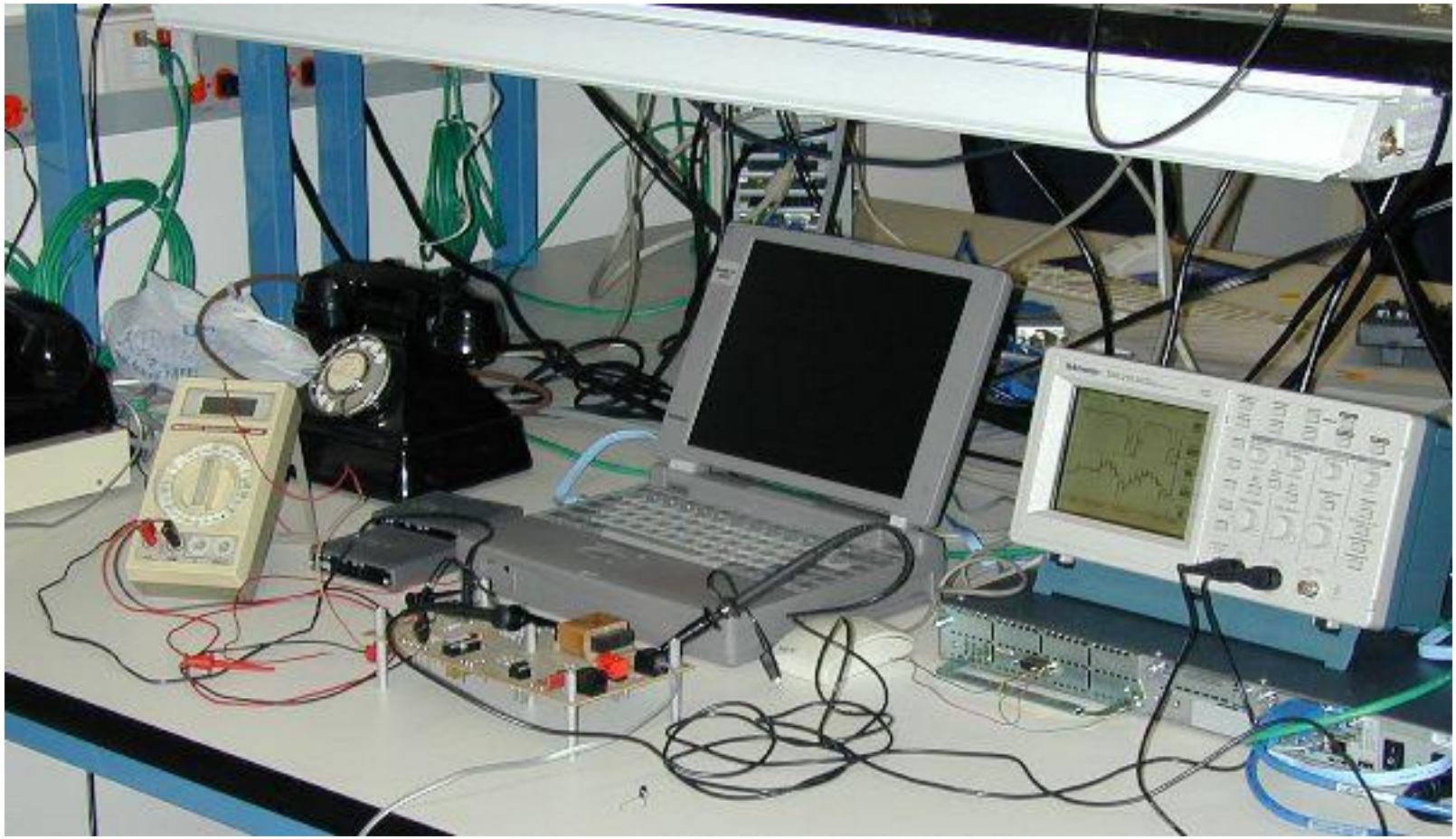


Design in Bread-board

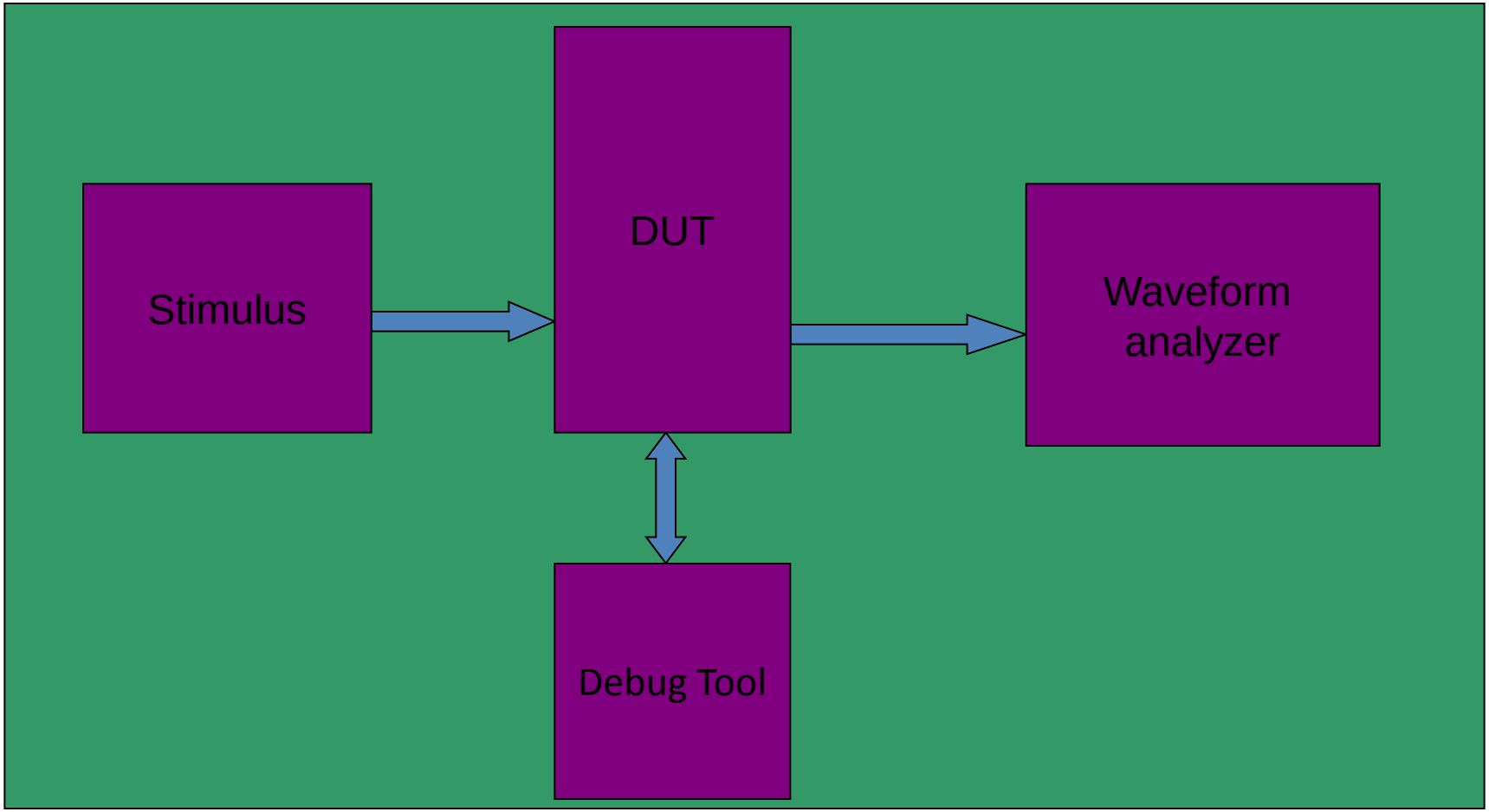
Oscilloscope

Multimeter

"Testbench"



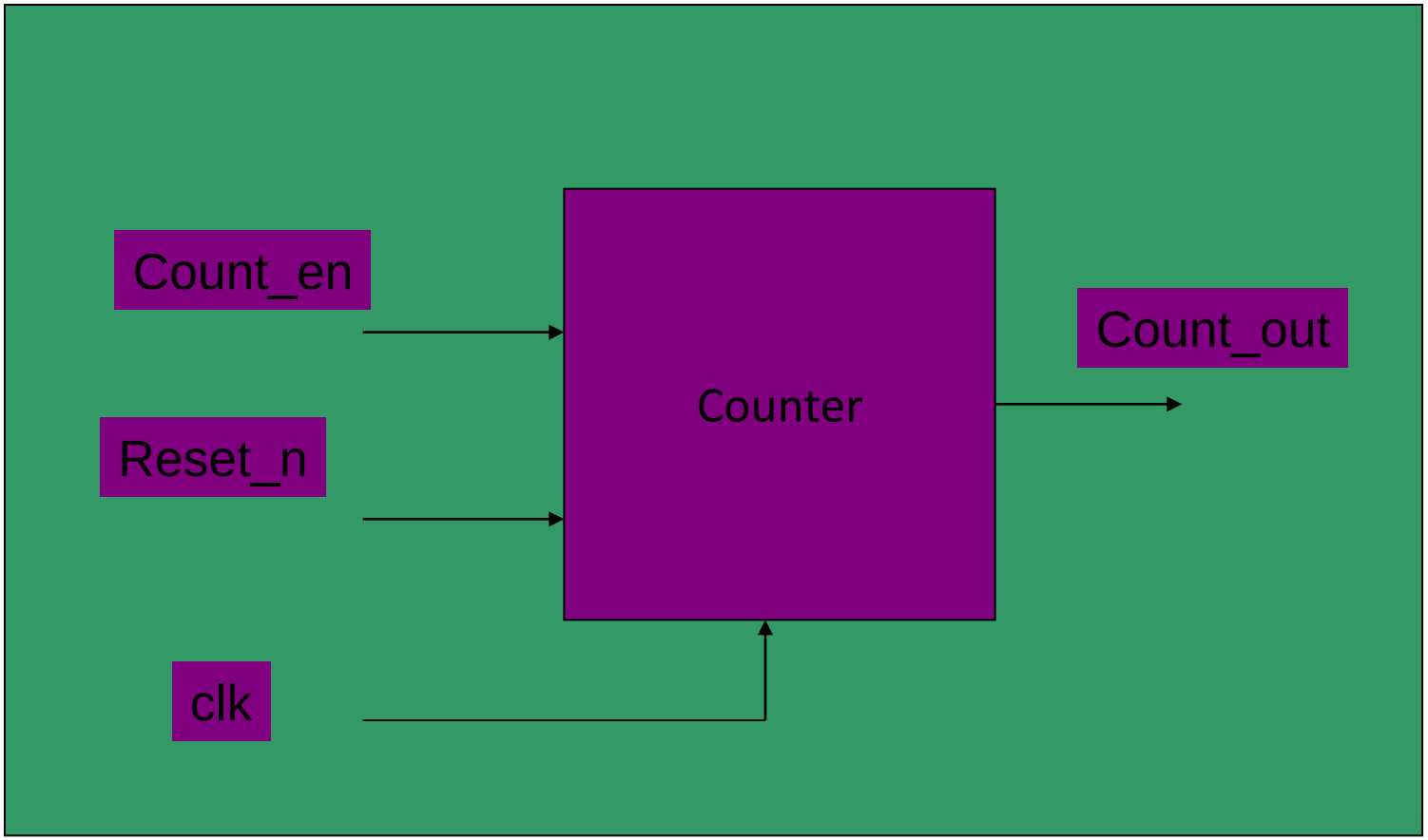
Testbench in Simulation world



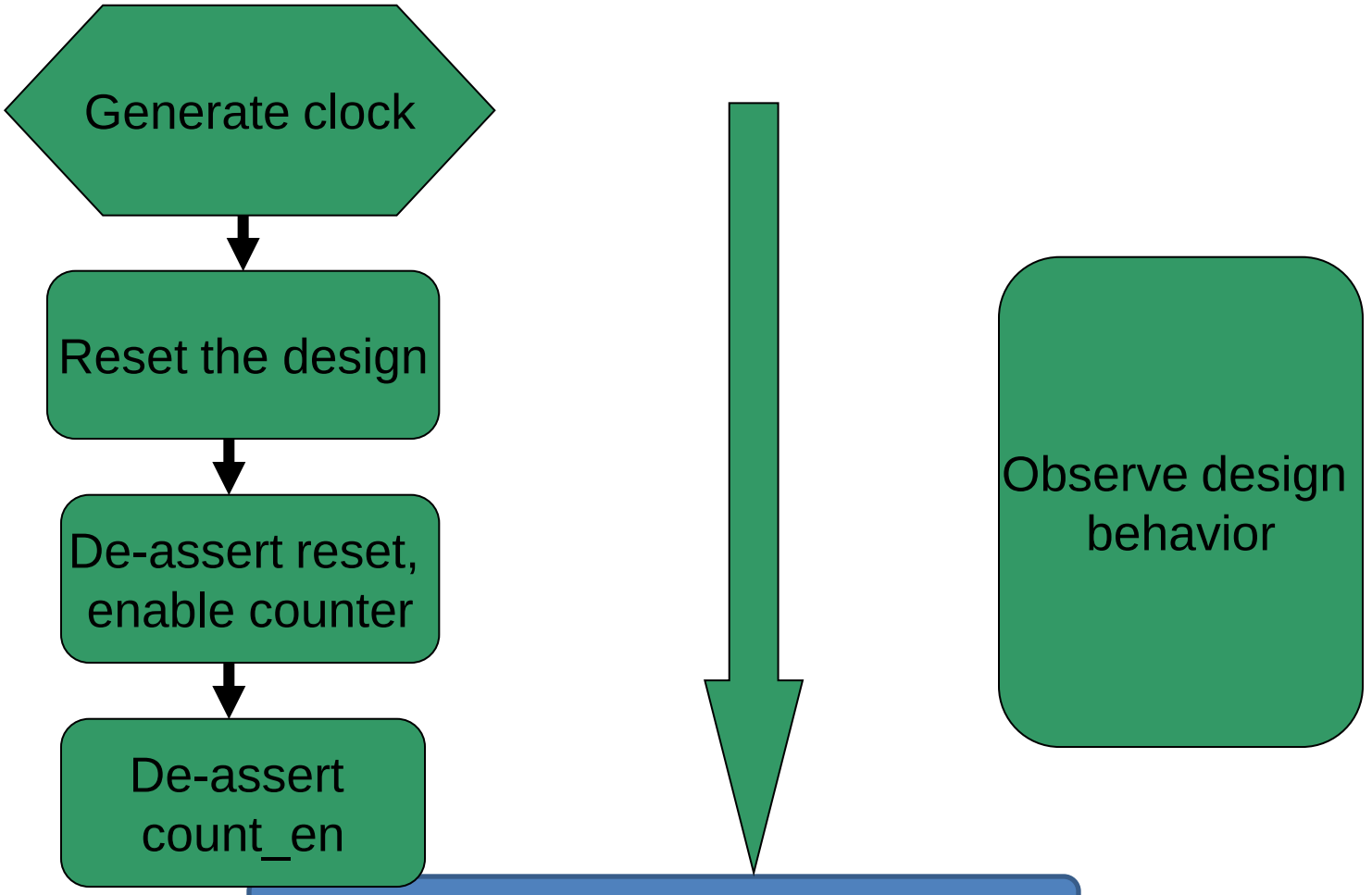
Types of Testbenches

- No single-size fits all!
- Depends on many things:
  - Complexity of the design
  - Is the design new or a derivative?
  - Application domain
  - Availability of tools, technologies
  - Skill set of team members

Counter DUT



Simplified Verification flow for the counter



Linear Testbench

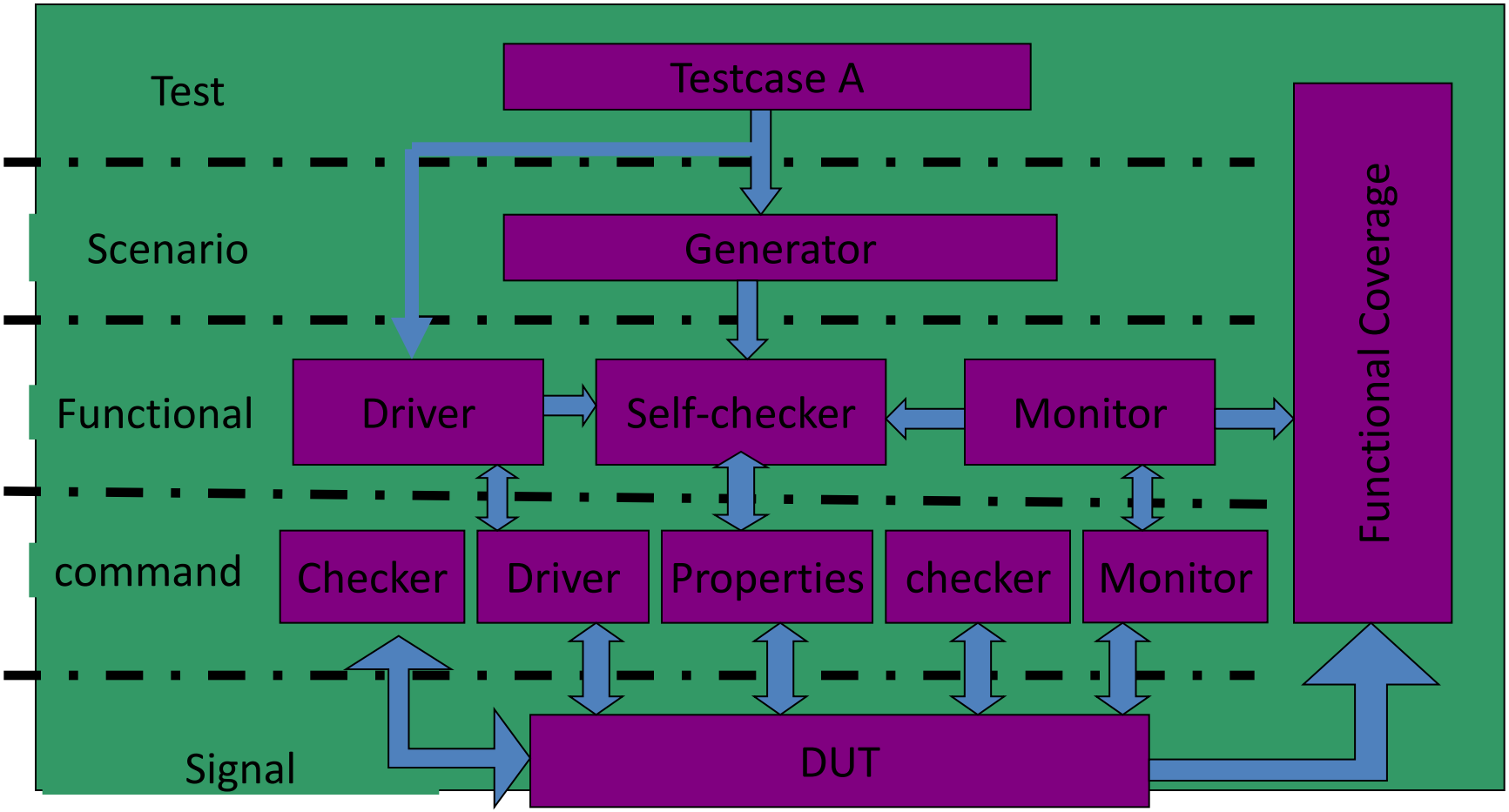
- First step into a proper Testbench
- Typically written in a HDL – Verilog/VHDL
- A simple, linear sequence of events over a period of several clocks
- Stimulus describes what happens in a series of assignments that are spread over time

```
initial begin : stim
  #10 reset = 1'b1;
  #100 reset = 1'b0;
  #10 inp_1 = 16'haabb;
        inp_2 = 4'b0010;
  #10 if (out_1 != inp_1 && inp_2) $display ("Error");
.....
end : stim
```

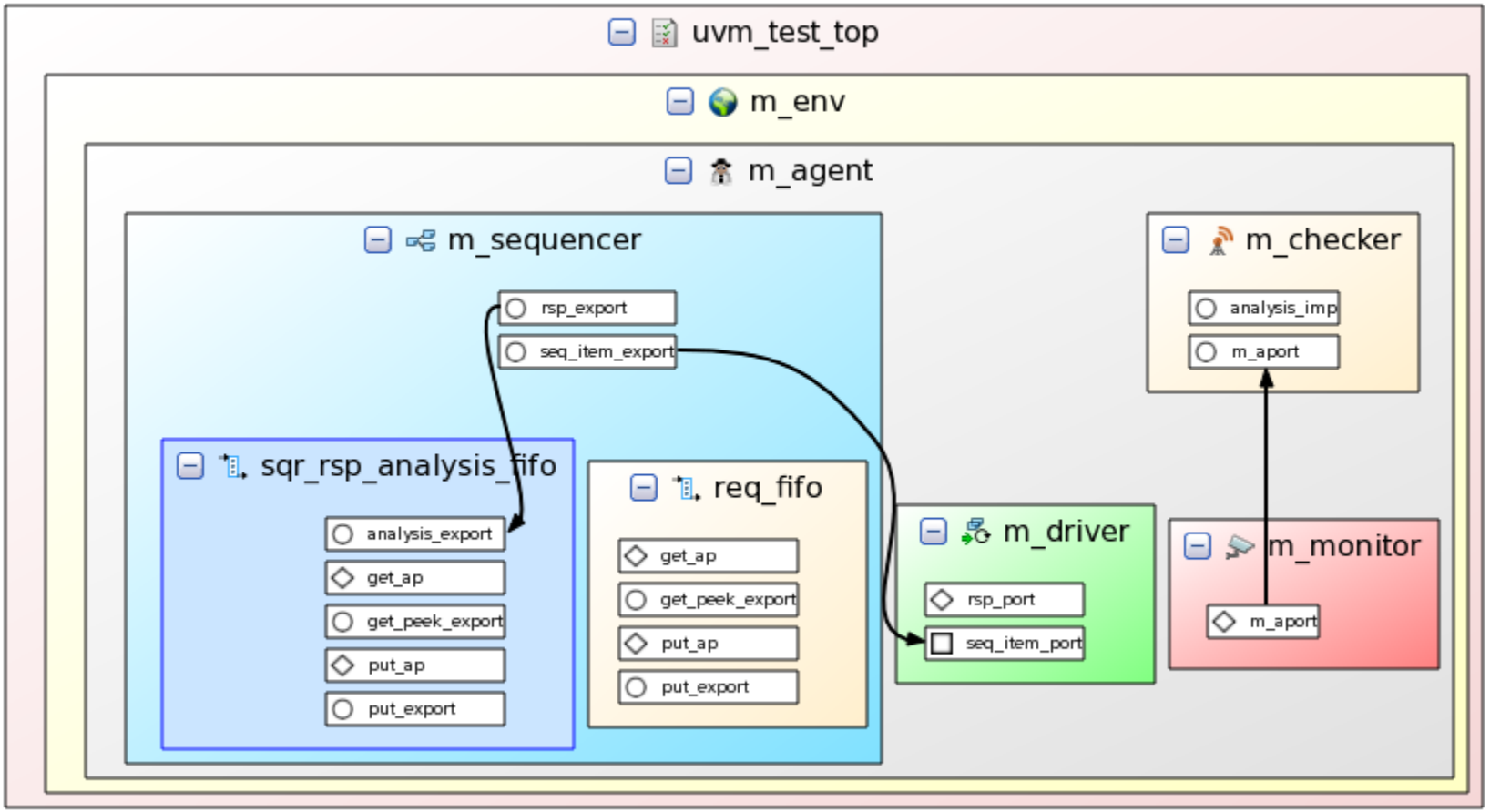
Procedural Testbench – S2P example

```
task send_ser_pkt (logic [31:0] data);
  ser_sop <= 1'b1;
  ser_eop <= 1'b0;
  ser_abort <= 1'b0;
  ser_data <= data[0];
  @ (posedge clk);
  ser_sop <= 1'b0;
  for (int i=0; i <= 30; i=i+1) begin : for_0
    ser_data <= data[i];
    @ (posedge clk);
  end : for_0
     ser_data <= data[31];
  ser_eop <= 1'b1;
  @ (posedge clk);
  ser_eop <= 1'b0;
endtask
```

Modern Verification Environment



Typical UVM architecture



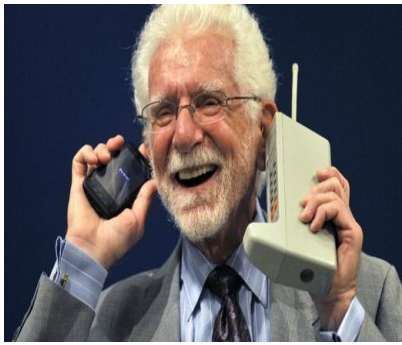
UVM journey so far

- UVM has come a long way
- Great framework
- Good block level stimulus features
- Well documented, proven to work
- Summary of decades of BKMs – Best Known Methods
- Continues to expand

Near future

- New markets, users for UVM
  - FPGA teams
  - RTL designers
  - Debug engineers
  - Assertion Developers, Formal Verification (counter examples)
  - Waveform-2-Stimulus tools

Phone - Who is it for?



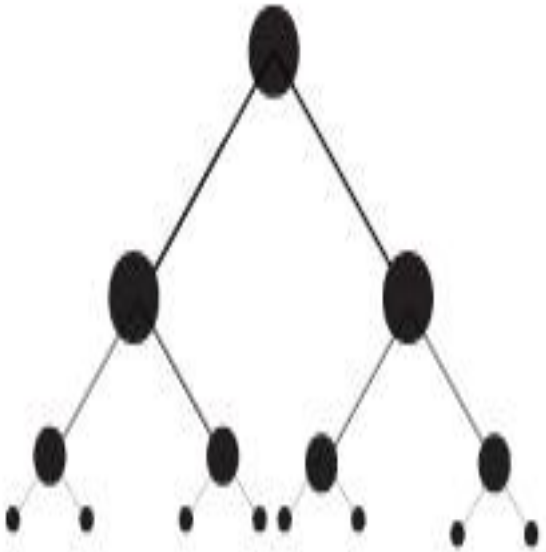
- Phones have gotten so complicated, so **hard to use**, that you wonder if this is designed for real people or for engineers.

Is UVM going the same route?

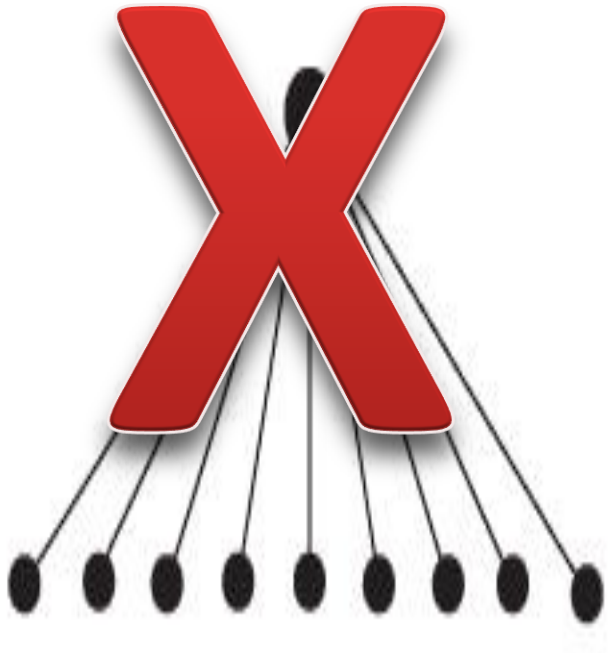
- UVM – **Universal Verification** Methodology
- Slowly becoming "only" for OOP savy, SW friendly Verification engineers
- Need to make it easy for RTL, FPGA, automation etc.

UVM is Complex, but..

COMPLEX



COMPLICATED



- Need not be complicated

CVC's Go2UVM Package

- Our goals:
  - Provide simple framework for writing high quality simulation traces
  - Not to deviate from UVM approach
  - Introduce UVM to Verilog users
  - Make them future ready for more powerful UVM
- Simple package on top of standard UVM, released in open-source via http://www.go2uvm.org

Go2UVM_pkg - Verilog vs. UVM stimulus

vlog_image.v (~\srini\proj\UVM\...Go2UVM

File Edit Tools Syntax Buffers Window Help

```
task reset ();
  $display( "Start of reset");
  rst_n <= 1'b0;
  repeat(10) @ (posedge clk);
  rst_n <= 1'b1;
  @ (posedge clk);
  $display( "End of reset");
endtask : reset

task drive();
   int i;
```

```
import uvm_pkg::*;
`include "uvm_macros.svh"

import cvc_go2uvm_pkg::*;

class go2uvm_count_test extends go2uvm_base_test;
  virtual count_if vif;

  task reset ();
    `uvm_info(log_id, "Start of reset", UVM_MEDIUM)
    vif.cb.rst_n <= 1'b0;
    repeat(10) @ (vif.cb);
    vif.cb.rst_n <= 1'b1;
    @ (vif.cb);
    `uvm_info(log_id, "End of reset", UVM_MEDIUM)
  endtask : reset

  task drive();
```

Go2UVM_pkg - Verilog vs. UVM stimulus

vlog_image.v (~\srini\pro...ex_updn_count

File Edit Tools Syntax Buffers Window Help

```
    @ (posedge clk);
    $display( "End of reset");
endtask : reset

task drive();
    int i;
    $display( "Start of Test");
    @ (posedge clk);
    load <= 1'b1;

    repeat(2) @ (posedge clk);
    load <= 1'b1;
    data <= 8'h78;

    repeat(2) @ (posedge clk);
    load <= 1'b1;
    cen <= 1'b1;
    up_dn <= 1'b1;

    repeat(3) @ (posedge clk);
    load <= 1'b1;
    cen <= 1'b1;
    up_dn <= 1'b0;
```

image.sv (~\srini\pro...updn_counter\tb) - GVIM

e Edit Tools Syntax Buffers Window Help

```
    @ (vif.cb);
    `uvm_info(log_id, "End of reset", UVM_MEDIUM)
endtask : reset

task drive();
    int i;
    `uvm_info(log_id, "Start of Test", UVM_MEDIUM)
    @ (vif.cb);
    vif.cb.load <= 1'b1;

    repeat(2) @ (vif.cb);
    vif.cb.load <= 1'b0;
    vif.cb.data <= 8'h78;

    repeat(2) @ (vif.cb);
    vif.cb.load <= 1'b1;
    vif.cb.cen <= 1'b1;
    vif.cb.up_dn <= 1'b1;

    repeat(3) @ (vif.cb);
    vif.cb.load <= 1'b1;
    vif.cb.cen <= 1'b1;
    vif.cb.up_dn <= 1'b0;
```

Go2UVM_pkg - finishing touch..

File Edit Tools Syntax Buffers Window Help

```
class go2uvm_count_test extends go2uvm_base_test;
  // Code masked
endclass : go2uvm_count_test

//Module
module go2uvm_count;
  logic clk;

   // Instantiate the test class
    go2uvm_count_test go2uvm_count_test_0;

    //Instantiation of Interface
    count_if count_if_0 (.clk(clk));

    // DUT Instantiation

  initial
    begin : stim
      go2uvm_count_test_0 = new ();
      go2uvm_count_test_0.vif = count_if_0;
      run_test ();
    end : stim

endmodule : go2uvm_count
```

CVC's role in the Verification industry

- Drove DVCon India 2014 from the front
- World-wide recognized Verification experts
  - Co-authors of 4 assertion books (Ajeetha & Srini)
- Coded VMM base classes in Synopsys
- UVMWorld.org contributions to UVM-Ref-Flow
- Delivered several training on UVM, OVM, VMM, SV, SVA, PSL
- Worked in evangelizing ABV, VMM, UVM across the world through several EDA companies
- Presented at various forums:
  - DVCon, SNUG-India, Taiwan, China, DVM (India) etc.
- Assisted standards development:
  - PSL, OVL, SVA, SV, UVM, eWG, VHDL, SV-AMS

How do access Go2UVM Package?

- Contact us via www.go2uvm.org
- Limited customer availability