IMEC TRAINING

# FE PROGRAM

imec

## FE PROGRAM

Digital design flow: overview

Logic synthesis

LEC

ATPG

Power analysis

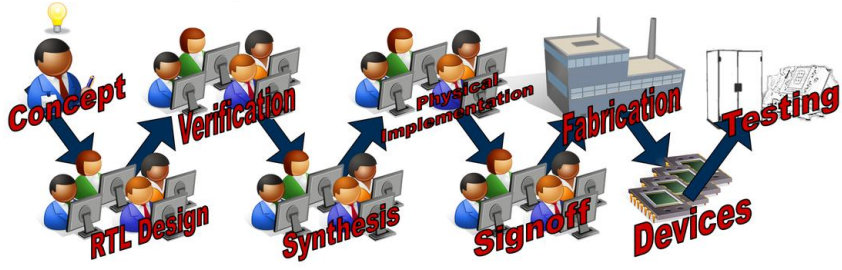
IMEC TRAINING

# DIGITAL DESIGN FLOW

**OVERVIEW**

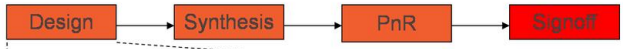
## DESIGN FLOW - OVERVIEW

A general IC design flow:



## DESIGN FLOW - OVERVIEW

The digital standard cell IC design flow:

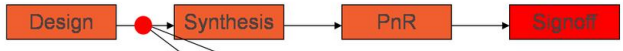



1) Creation of Spec and verification plan
2) High level modelling, Architectural exploration and HW-SW co-design
3) RTL Coding and Verification
   a) Linting
   b) Development of a regression suite
   c) Low Power Design
4) Low Power Verification
5) System Level Verification
6) Bug hunting / advanced verification techniques



## DESIGN FLOW - OVERVIEW

The digital standard cell IC design flow:




- The design phase results in some code.
- Usually written in a special coding style called Register Transfer Level (RTL) Code
- Typically this is in either VHDL or Verilog.
- Other languages such as SystemC, and SystemVerilog can also be used.
  - Check your synthesis tools support

```
GenCountAndCapture: process (clk2_muxed,ARST2N)
begin
  if ARST2N = '0' then
    Cntr       <= (others =>'0');
    CntrValue  <= (others =>'0');
  elsif rising_edge(clk2_muxed) then
    Cntr <= Cntr + 1 ; -- auto wrap
    if RisEdge_async_interrupt = '1' then
      CntrValue  <= std_logic_vector(Cntr);
      Cntr       <= (others =>'0');
    end if;
  end if;
end process GenCountAndCapture;
```

## DESIGN FLOW - OVERVIEW

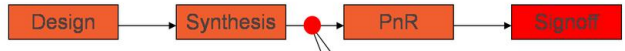
The digital standard cell IC design flow:


1) Reading HDL (VHDL or Verilog) code
2) Synthesis into a netlist in terms of generic cells
   a) Such as and, or, not, sequential elements
3) Mapping into logic cells from the library
4) Optimization of the design to meet constraints (Timing, Area and Power)
   a) Datapath optimisations can modify the implementation of operators to improve performance
5) Physical Synthesis (Or Topographical Synthesis)
   a) Requiring early floorplanning
6) Low Power Techniques (Both Basic and Advanced)
7) Compressed Scan / UDSM Test



## DESIGN FLOW - OVERVIEW

The digital standard cell IC design flow:




- The Synthesis phase results in a Verilog netlist
- This just specifics connectivity between instances from your process library
  - E.g. HDNAN2DL, HDDFF2
- The functionality of the circuit is contained in the library cells
- Not particularly readable

```
module ExtraOnClk2_CntrWidth20_N12_M1_F1_DW01_inc_0 ( A, SUM );
  input [19:0] A;
  output [19:0] SUM;

  wire   [19:2] carry;

  HA1X1 U1_1_18 ( .A(A[18]), .B(carry[18]), .CO(carry[19]), .S(SUM[18]) );
  HA1X1 U1_1_17 ( .A(A[17]), .B(carry[17]), .CO(carry[18]), .S(SUM[17]) );
  HA1X1 U1_1_16 ( .A(A[16]), .B(carry[16]), .CO(carry[17]), .S(SUM[16]) );
  HA1X1 U1_1_15 ( .A(A[15]), .B(carry[15]), .CO(carry[16]), .S(SUM[15]) );
  HA1X1 U1_1_14 ( .A(A[14]), .B(carry[14]), .CO(carry[15]), .S(SUM[14]) );
  HA1X1 U1_1_13 ( .A(A[13]), .B(carry[13]), .CO(carry[14]), .S(SUM[13]) );
  HA1X1 U1_1_12 ( .A(A[12]), .B(carry[12]), .CO(carry[13]), .S(SUM[12]) );
  HA1X1 U1_1_11 ( .A(A[11]), .B(carry[11]), .CO(carry[12]), .S(SUM[11]) );
  HA1X1 U1_1_10 ( .A(A[10]), .B(carry[10]), .CO(carry[11]), .S(SUM[10]) );
  HA1X1 U1_1_9 ( .A(A[9]), .B(carry[9]), .CO(carry[10]), .S(SUM[9]) );
  HA1X1 U1_1_8 ( .A(A[8]), .B(carry[8]), .CO(carry[9]), .S(SUM[8]) );
  HA1X1 U1_1_7 ( .A(A[7]), .B(carry[7]), .CO(carry[8]), .S(SUM[7]) );
  HA1X1 U1_1_6 ( .A(A[6]), .B(carry[6]), .CO(carry[7]), .S(SUM[6]) );
  HA1X1 U1_1_5 ( .A(A[5]), .B(carry[5]), .CO(carry[6]), .S(SUM[5]) );
  HA1X1 U1_1_4 ( .A(A[4]), .B(carry[4]), .CO(carry[5]), .S(SUM[4]) );
  HA1X1 U1_1_3 ( .A(A[3]), .B(carry[3]), .CO(carry[4]), .S(SUM[3]) );
  HA1X1 U1_1_2 ( .A(A[2]), .B(carry[2]), .CO(carry[3]), .S(SUM[2]) );
  HA1X1 U1_1_1 ( .A(A[1]), .B(A[0]), .CO(carry[2]), .S(SUM[1]) );
  INVX1 U1 ( .I(A[0]), .O(SUM[0]) );
  XOR2CX1 U2 ( .I1(carry[19]), .I2(A[19]), .O(SUM[19]) );
endmodule
```

## DESIGN FLOW - OVERVIEW

The digital standard cell IC design flow:

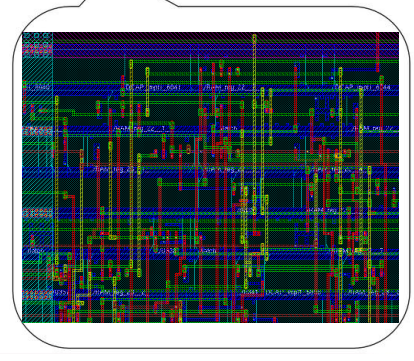

1) Floorplanning – FP init, Macro placement, Power planning, Hierarchy…
2) Placement of cells
   a) and Post Place Timing Optimisation
3) Clock Tree Synthesis (CTS)
   a) and Post CTS Timing Optimisation
4) Routing of nets
   a) and Post Route Timing Optimisation
5) Clock Tree Optimisation (CTO)
6) Design finishing – Core fillers / metal fill / metal slotting , etc…
   a) Wire and Via Optimisation
7) Full custom layout finishing (If required.)
8) Multi-Mode-Multi-Corner timing analysis and Opt throughout the flow!



## DESIGN FLOW - OVERVIEW

The digital standard cell IC design flow:


- The Place and Route phase results in the creation of a full layout of your design
- The design is exported in GDSII format for production and Layout based signoff checks.
- The design is also saved as:
  - Verilog gate level netlist
  - Standard Delay Format (SDF) file
  - Layout parasitics
- These are required for a range of other signoff checks.



## DESIGN FLOW - OVERVIEW

The digital standard cell IC design flow:


1) DRC / LVS / ERC
2) Rail Analysis (IR-Drop / EM analysis)
   a) Dynamic/Static analyses at Gate/Transistor levels
3) Static Timing Analysis
   a) With crosstalk delay and crosstalk noise analyses
4) Dynamic simulation
   a) Low Power Simulation
5) Formal equivalence checking
6) Power analysis (Peak, average and time based)
7) Transistor level simulation
8) Low Power Verification (Structural / Functional)

 DIGITAL DESIGN FLOW: OVERVIEW BY JORGO TSOUHLARAKIS 11