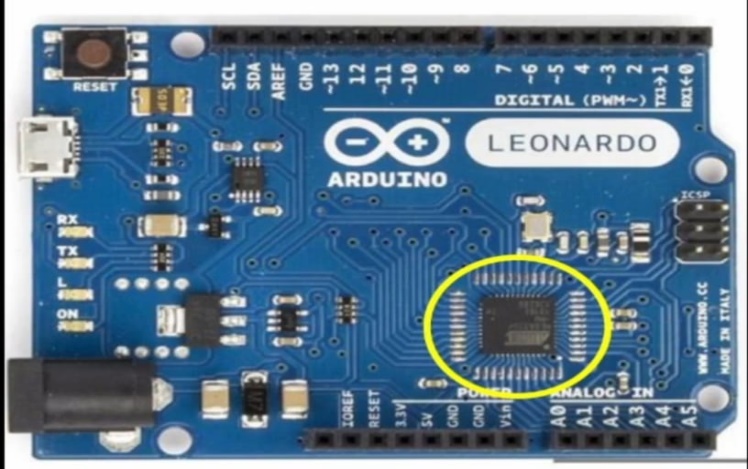
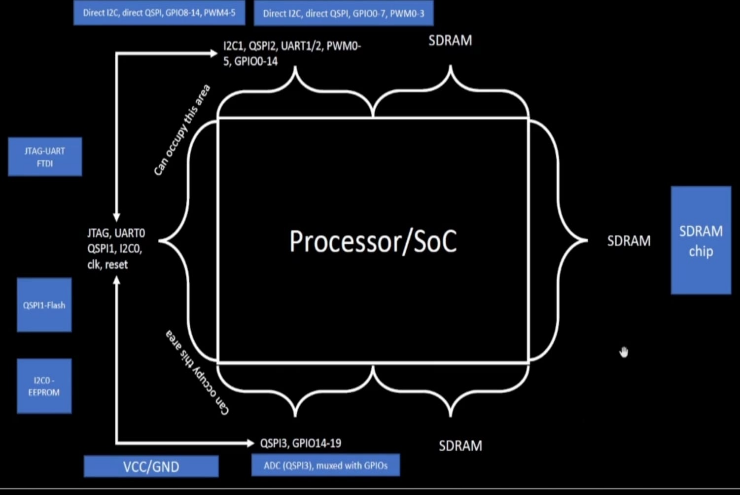
**DIGITAL VLSI SOC DESIGN AND PLANNING**

**THEORY:**



* Took the Arduino board as an example, to demonstrate the SoC /processor.



* The interfaces around the processor are JTAP to program the processor, I2C-EEPROM,QSPI1-Flash,SDRAM,UART and GPIOs.

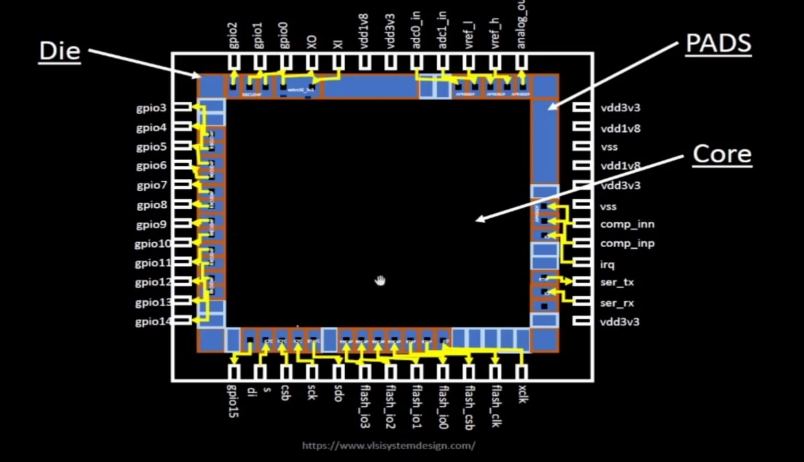
**View of package**

Packages given with the name of QFN-48 – Quad Flat No Leads(there are different other packages).

Chip is sitting at the centre of package.

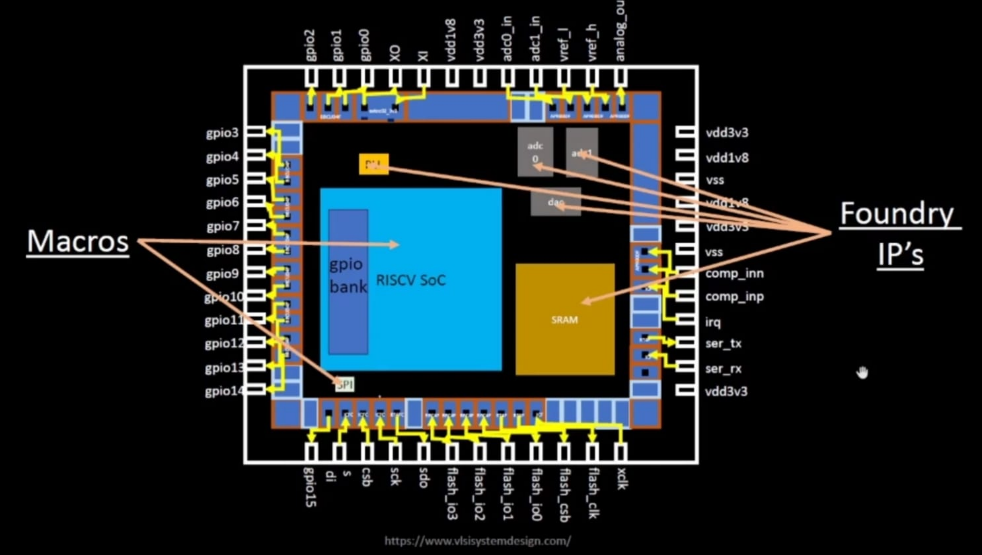
Chip is connected to pins using wirebound

**What is inside chip?**



* Chip has various components, one of the important component is PADS.
* Pads are something, through which you can send signal inside the chip.
* Core of the chip, is where all the digital logic sits.
* Die is the area of the chip. It is manufactured on a silicon wafer.

**Example of RISC V SoC:**

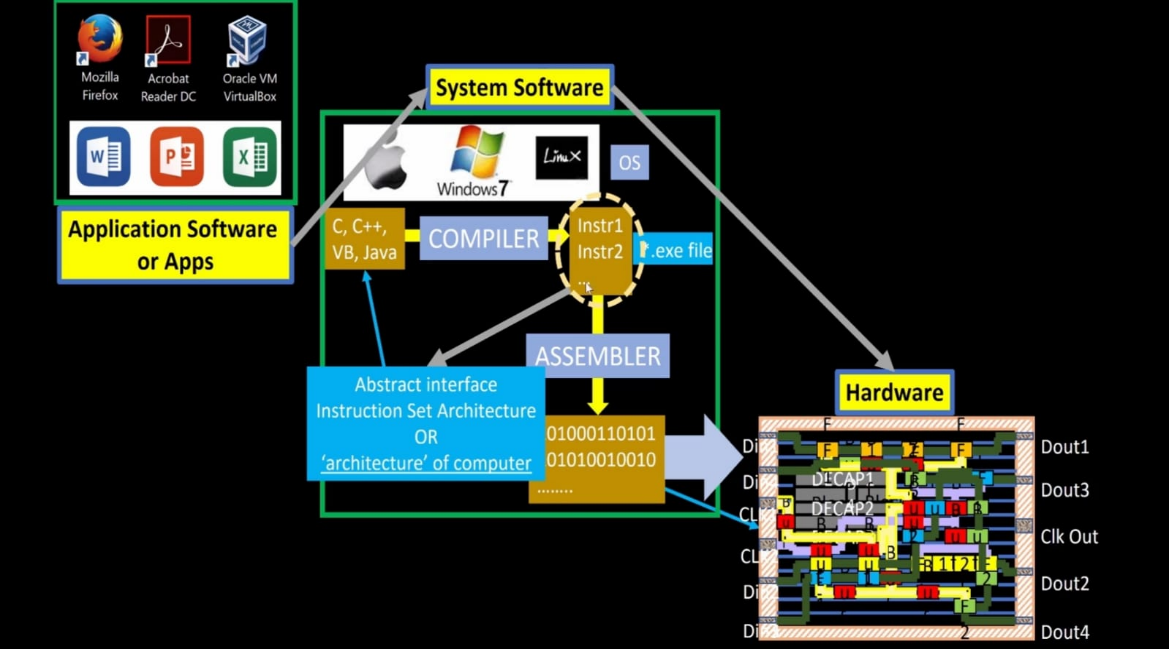


* Typical core area consists of SoC, SPI,PLL,SRAM,ADC,DAC etc,
* PLL,ADC,DAC,SRAM are called foundry IP’s.
* SoC, SPI are called macros.

**RISC V Instruction set architecture(ISA):**

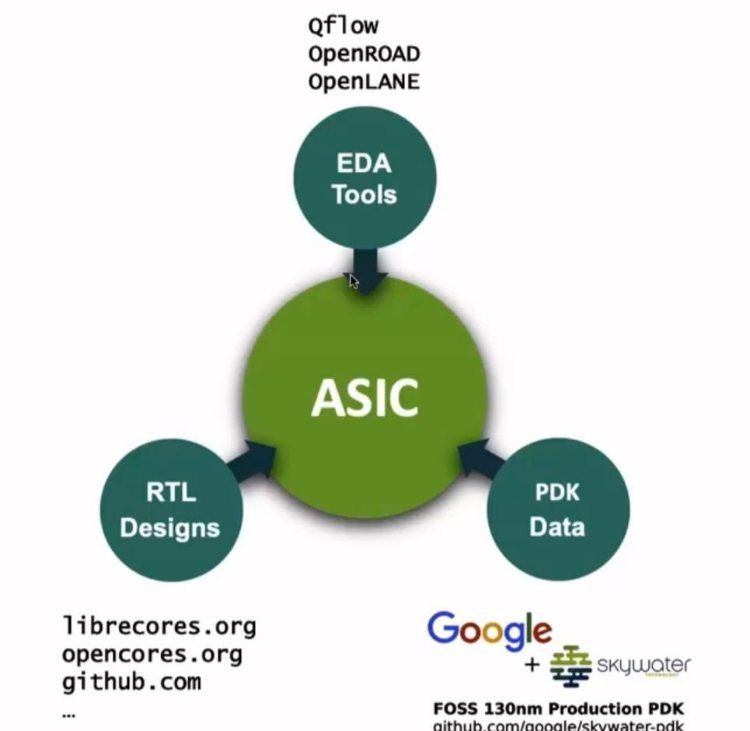
* Implement RISC V specification using RTL(picorv32 cpu core).
* Synthesized netlist of RTL.
* Physical design implementation of netlist.
* RTL to layout is the standard RTL2GDSII flow.

**How do apps run on hardware?**



* Application software enters into system software, system software consists of OS, Compiler, Assembler.
* This OS will take this apps and converts it into assembly level and then to binary or machine level language.
* Compiler takes C,C++,java (high level code) and convert it into set of instructions(.exe).
* Assembler takes this instructions and convert it into binaries (0,1s).
* These instruction set here are called as RISC V architecture.
* Interface used to reach from instructions to machine level is hardware description language(hdl).

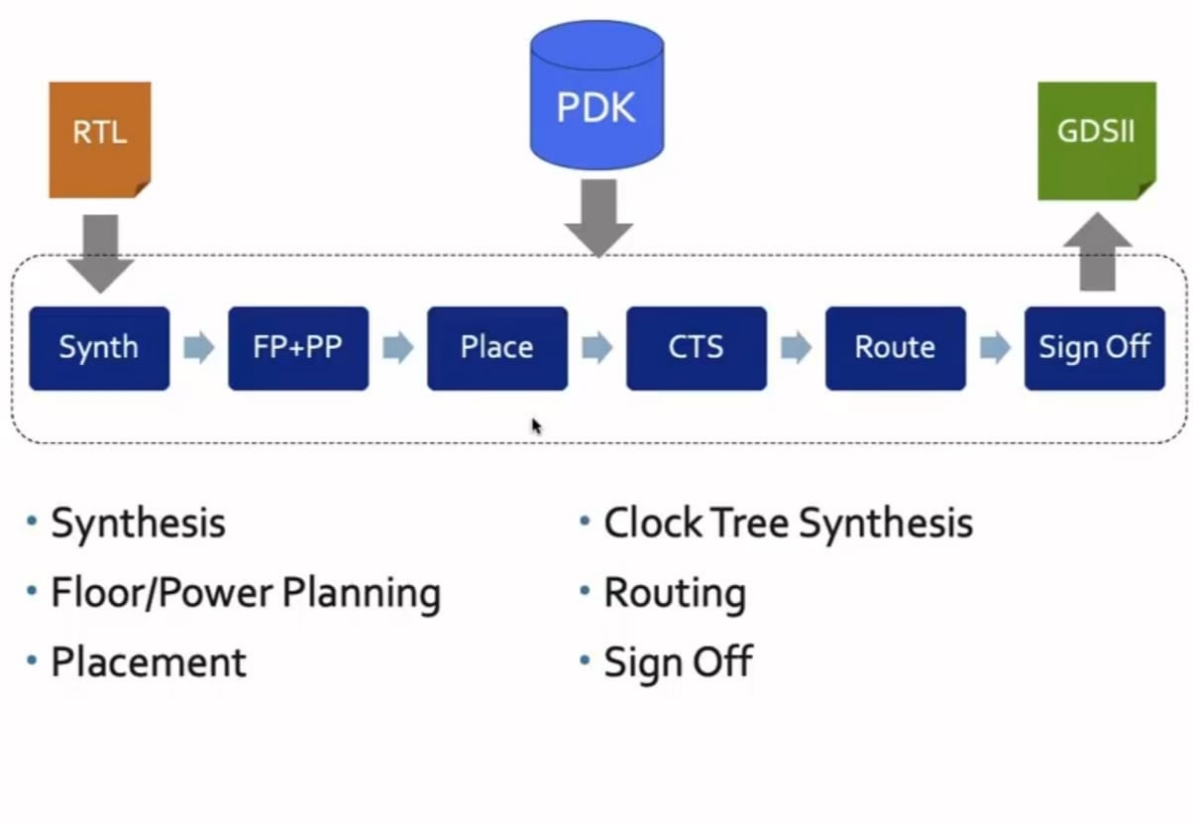
**SoC Design Using OpenLANE**



* We need RTL IP’s, EDA Tools, PDK Data to design the ASIC.
* Open source EDA Tools, RTL designs, PDK data are used.
* PDK = the interface between the fab and the designers.
* PDK contains, process design rules(DRC,LVS,PEX), device model, digital standard cell libraries, I/O libraries.

**ASIC Design Flow:**

* ASIC flow objective: RTL to GDSII( Also called automated PnR and/or physical implementation.

**Simplified RTL to GDSII Flow:**

1.Synthesis:converts RTL to a circuit out of components from the standard cell library(scl).

* Standard cells-have regular layout.

2.Floor/power planning: plan the silicon area and to create robust power distribution to power the network.

* + Chip floor planning: partion the chip die between different system building blocks and place the I/O pads.
  + Macro floor planning:dimensions, pin locations, rows definition.
  + Power planning: chip is powered by multiple vdd and gnd pins. They are connected through vertical and horizontal metal straps.

3.placement : place the cells on the floorplan rows, aligned with the sites.

* 2 steps: global(not legalled) and detailed placements(legalled).

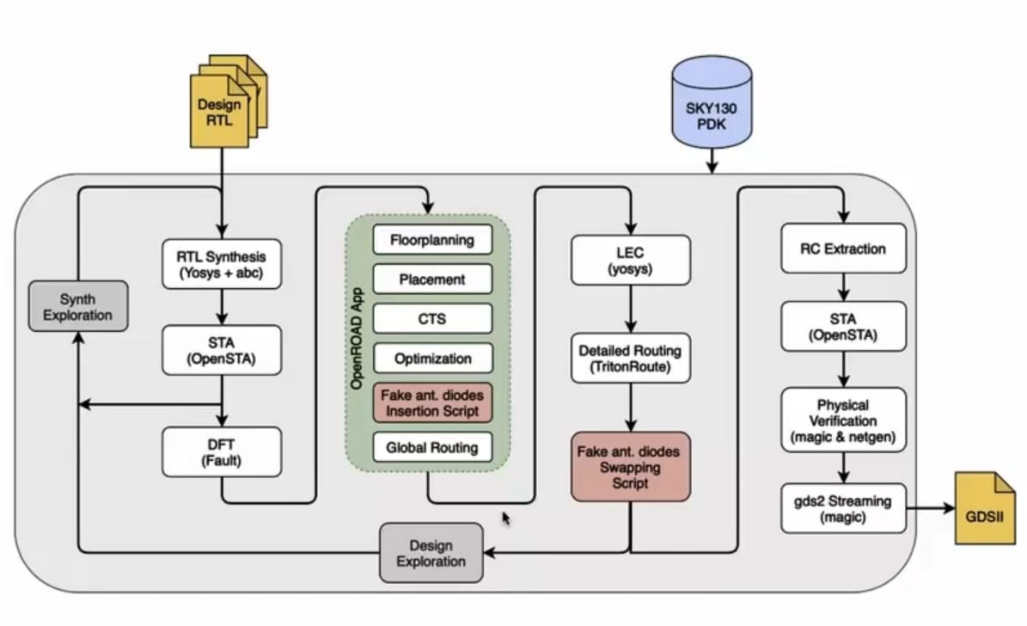
4. clock tree synthesis : create a clock distribution network to deliver the clock to all cells with minimum skew(0 is hard to achieve)

* Usually a tree- H,x,pi tree

5.routing: implement the interconnect using the available metal layers.

6.signoff :

* Physical verifications: design rules checking (DRC) and layout vs schematic(LVS).
* Timing verifications: static timing analysis(STA).

**OpenLANE ASIC flow:**

* Produce a clean GDSII with no human intervention (no-human-in-the-loop).
* Tuned for skywater130nm open PDK.
* Can be used to harden macros and chips.
* OpenLANE is based on several open source projects – openroad,fault,yosys,abc,qflow,magic .
* DFT to scan insersion(JTAG)
* Physical implementation also called automated PnR(Place and route)
* LEC(logic equvilence check): everytime the netlist is modified, verification must be performed.
* Dealing with antenna rules violations:

1. Bridging attaches a higher layer intermediary

2. Add antenna diode cell to leak away charges.

* Physical verification: DRC(magic) and LVS(magic and netgen).