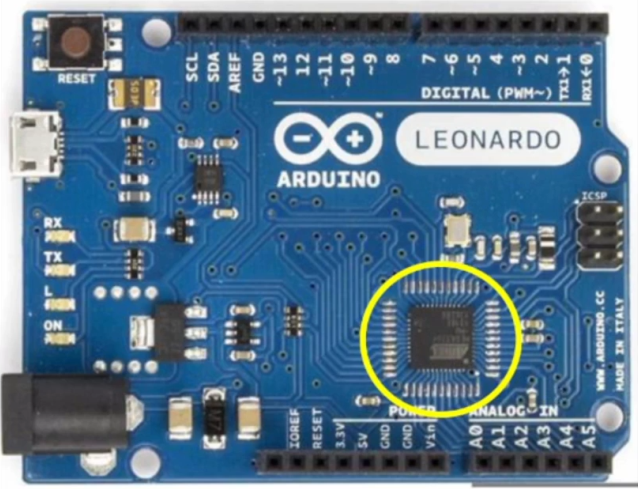
DIGITAL-VLSI-SOC-DESIGN-AND-PLANNING

Day 1:

1. **Introduction to IC package, chip, pads, core and IPs**

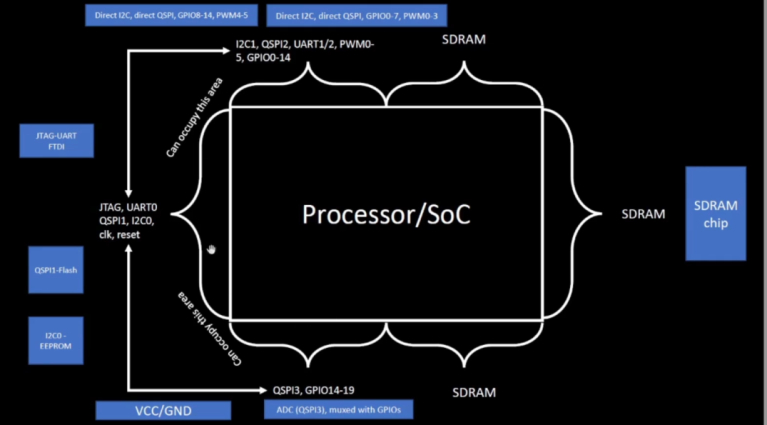
Integrated Circuits (ICs) are fundamental to modern electronic devices, from smart phones to advanced computing systems. Understanding the structure and function of ICs is to appreciate how electronic devices operate at a fundamental level. The key components of ICs include their packaging, chips, pads, core, and Intellectual Properties (IPs) etc.

Arduino microcontroller board : The highlighted section illustrates the microprocessor (chip) which is connected to other elements on the board. The design process of this chip, from an abstract level to its final fabrication, is carried out using the RTL to GDSII flow. Arduino encompasses both a programmable physical circuit board, commonly known as a microcontroller, and a software component known as the IDE (Integrated Development Environment). This software operates on a computer and is used to create and transfer code to the physical board.

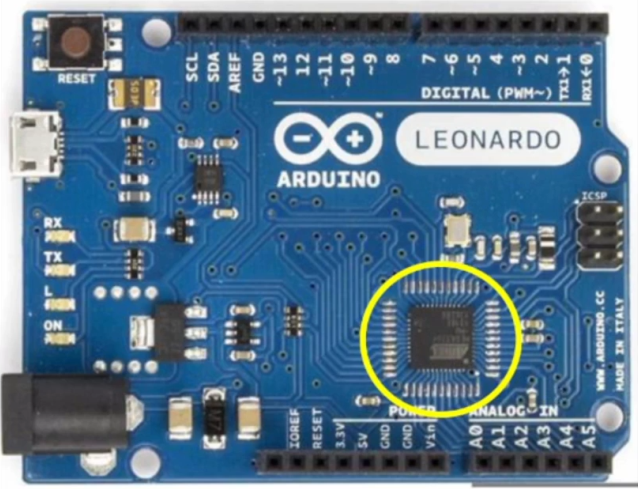
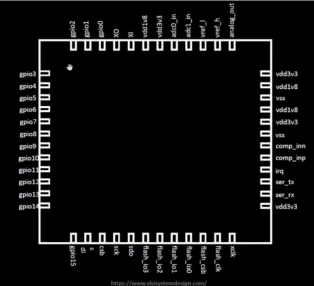
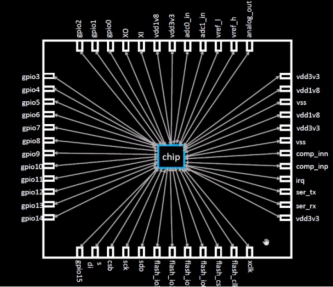


Microcontroller with QFN package

The processor / SoC is interfaced to external world through the other peripheral devices in the board namely JTAG, Flash, I2C-EEPROM, power ports, ADCs muxed with GPIOs, SDRAM etc.

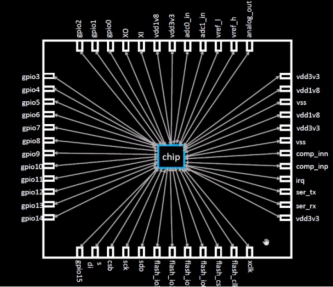
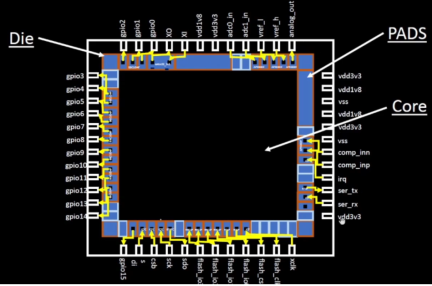
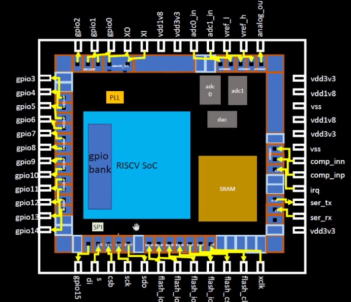


If we open up the chip, it looks like this. Here, we consider the Quad Flat No-leads (QFN) -48 package chip design. There are different types of packaging with different dimensions. The chip is placed at the centre of this package and the connections from each pin from the package to the respective pins in the chip are made using wire bonds. Through this the external signals from outside world reaches the chip.

When we open up the chip, we can visualize the various components inside namely **PADS** - through which we can transfer the signals to the chip and from the chip to the outside world, **Core**- where the entire circuit blocks /digital blocks are placed and Die- Present at the corner, it is the size of the entire chip. Typical core consists of SoC, SRAM, PLL, ADCs, DACs, etc.

PLL, ADC, DAC, SRAM are known as Foundry IPs and SoC, SPIs are basically called as Macros (digital design).

**Foundary IPs**

(Foundry IPs: Needs intelligent to build these blocks)

1. Introduction to RISC V –ISA

RISC-V is a open source instruction set architecture (ISA) that was developed at the University of California, Berkeley. It is based on established RISC principles and is provided under open source licenses. Several companies are offering or have announced RISC-V hardware, and open source operating systems with RISC-V support are available. The instruction set is supported in several popular software tool chains. The instruction set is designed to be versatile, with a fixed length of **32-bit** naturally aligned instructions, and the ability to have variable length extensions with each instruction being any number of 16-bit parcels in length. The ISA also supports address space variants of 32-bit and 64-bit. Chip is connected to the package with the help of bond wires.

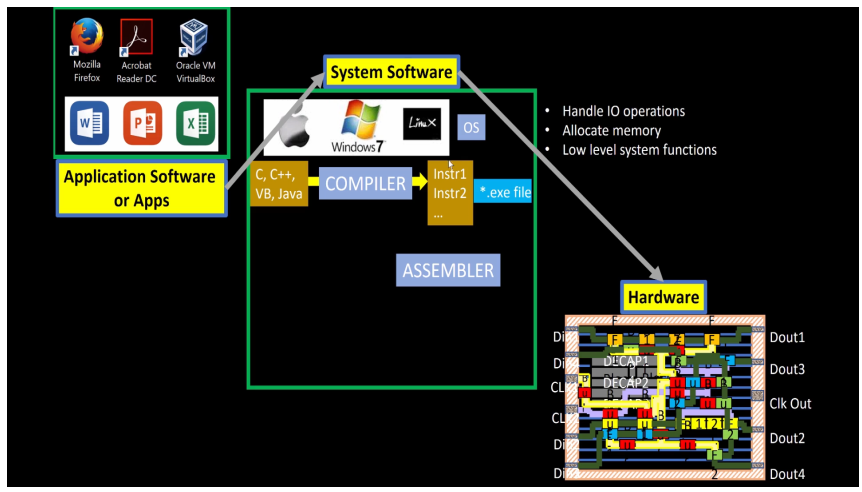
C-program -> using compiler converted to Assembly language -> Binary streams -> reaches the chip

Another interface – HDL

C-program-> RISC V specification in RTL then RTL to layout



1. Application software to Hardware



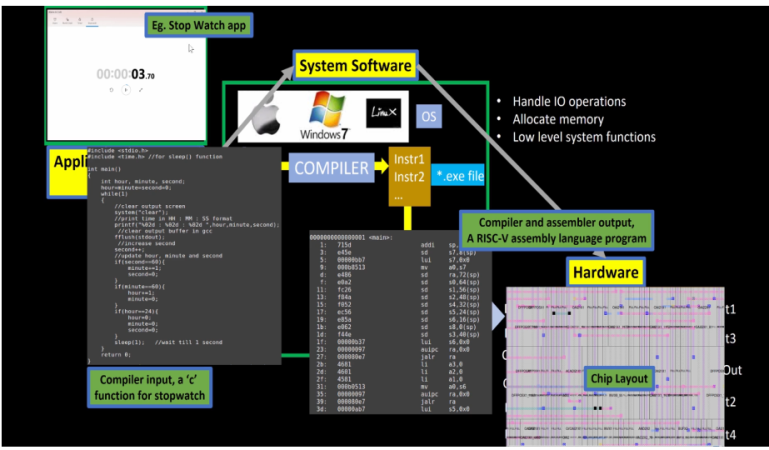
**Application Software - > System Software - > Hardware chip**

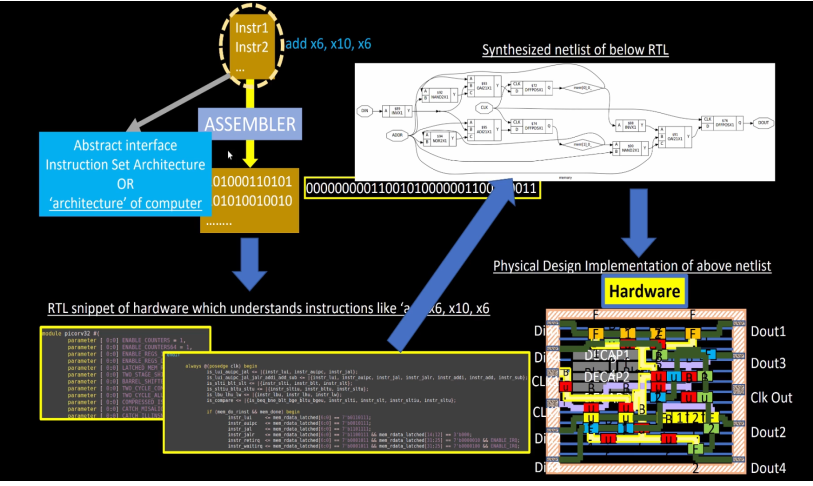
Apps enter into a block of system software and system software converts the entire program into binary language. There are some layers inside the system software - Operating System, Compiler, Assembler

**Operating system** handles input/output operations and allocate memory also it manage the low level system functions.

**Compiler** takes the output from the operating system as C,C++,Java and convert them into instructions. This instruction depends upon which hardware is used.(For eg. ARM processor then the instruction set will be ARM set)

**Assembler** takes the instructions from compiler and converts them into respective binary numbers. This binary language now sends to hardware and hardware performs output based on the function it receives and gives the output. Instruction acts as abstract interface between C-language and the hardware.





1. SoC Design using OpenLane

**Introduction to Open source Digital ASIC Design**

**What is RTL design?**

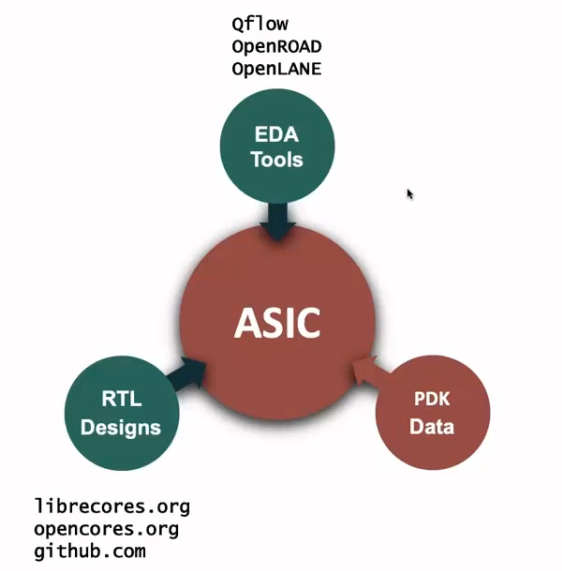
In digital circuit design, register-transfer level (RTL) is a design abstraction which models a synchronous digital circuit in terms of the flow of digital signals (data) between hardware registers, and the logical operations performed on those signals. For these designs many open sources are available.

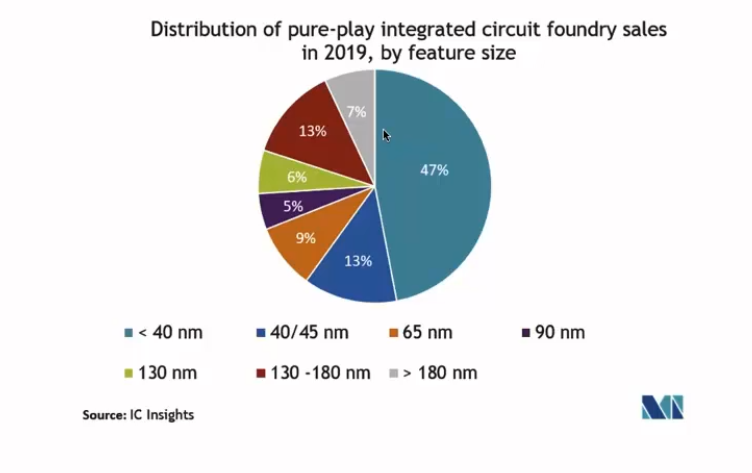
**What are EDA tools?**

The term Electronic Design Automation (EDA) refers to the tools that are used to design and verify integrated circuits (ICs), printed circuit boards (PCBs), and electronic systems. Many open sources tools are available like Qflow, OpenROAD, OpenLANE, etc...

**What is PDK?**

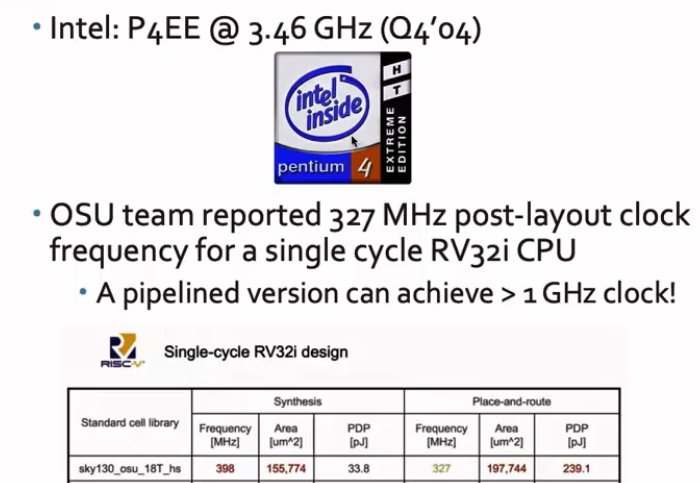
Process Design Kit (PDK) is a software tool used in the design and simulation of semiconductor manufacturing processes. It provides a comprehensive set of tools and libraries for modelling and simulating various aspects of the manufacturing process, including process flow, equipment layout, and process parameters. PDK is used by semiconductor manufacturers to design and optimize their manufacturing processes, from the early stages of process development to the final stages of production. It helps to identify bottlenecks and potential problems in the process, reduce the risk of defects and improve the overall yield and quality of the devices being manufactured.



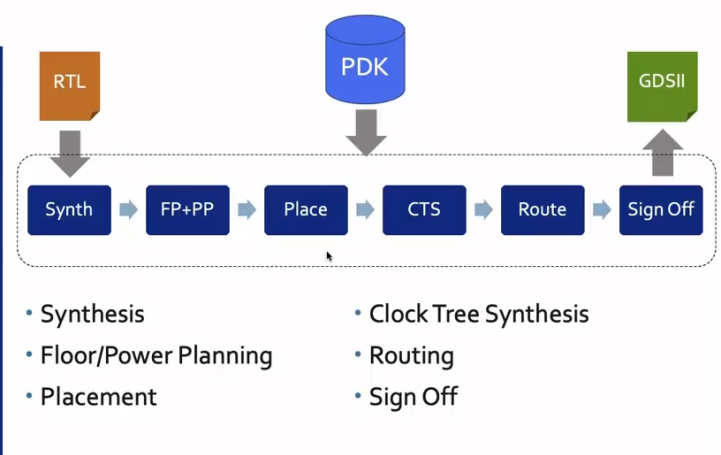


In 2020, GOOGLE release the open source PDK for FOSS 130nm production with the skywater technology. Current Technology node is in 3 nm.. But in many applications, the advance node is not required, and the cost of advanced node is also high as compared to 130 nm processors. This 130nm processor is also fast processor.

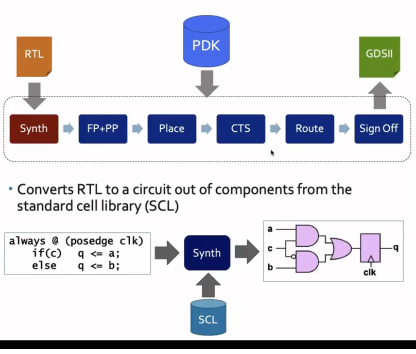
For example, Intel: P4EE @3.46 GHz(Q4'o4), sky130\_OSU (single cycle RV32i CPU) pipeline version can achieve more than 1 GHz clock.



**Simplified RTL to GDS Flow**

****

**Step 1. Synthesis:- In the synthesis, the design RTL is translated to a circuit out from the standard cell library (SCL). The resultant circuit is described in HDL - gate level netlist. The gate level net list is functionally equivalent to the RTL.**

****

**Step 2. Floor/Power Planning:-**

**Floor planning is the process of dividing the silicon area of a chip into smaller regions or blocks, and distributing the components and interconnects within those blocks. The main objective of floor planning is to optimize the layout of the components and interconnects to reduce the distance between them, minimize the number of interconnects, and improve the overall performance of the chip.**

**In floor planning, the chip is divided into smaller areas called "dies," and each die is further divided into smaller regions called "cores." The components and interconnects are then placed within these cores to optimize the layout.**

**The main objective here is that to plan silicon area and distribute the power to the whole circuit. In the chip floor planning, the partition chip die between different system building blocks and place the i/o pads. In micro floor planning, we define the dimensions, pin locations, rows. In power planning, the power network is constructed. Typically, the chip is power by multiple VDD and GND. so; total components are connected to power supply horizontally and vertically by metal strips. Here parallel structures are used to reduce the resistance. To address the electro magnetization problem, power distribution network uses upper metal layers, which are thicker than lower metal layers. Hence have less resistance.**