A logo with a symbol and text

Description automatically generated**WORKSHOP ON**

**CHIP DESIGN**

**At**

**NIT CALICUT**

**Bachelor of Technology**

**in**

**Electronics and Communication Engineering**

Summer Workshop

*Submitted by*

**Student Name: Ankush Verma**

**Registration Number: 200907058**

A logo with a brown logo

Description automatically generated

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

MANIPAL INSTITUTE OF TECHNOLOGY

(A Constituent institute of MAHE, Manipal)

MANIPAL – 576104, KARNATAKA, INDIA

**1. INTRODUCTION :**

OpenROAD FDP (Flows, Data, and PDKs) is a groundbreaking open-source initiative that aims to transform the landscape of chip design by enhancing accessibility and collaboration among engineers and researchers. The project centers around empowering designers to contribute to the development of chip design flows, thereby facilitating a more inclusive and open environment for innovation in the semiconductor industry.

The chip design process is traditionally complex and resource-intensive, making it challenging for smaller organizations, startups, and academic partners to participate actively in the field. Proprietary design tools and restrictive licensing agreements often impose significant barriers to entry, limiting the ability of diverse and talented individuals to explore their ideas in chip design.

OpenROAD FDP seeks to address these challenges by providing a comprehensive set of open-source tools, data, and Process Design Kits (PDKs) to the chip design community. Let's break down the key components of OpenROAD FDP:

1. Flows: Chip design involves a sequence of complex steps, referred to as design flows, that cover everything from initial concept to the final physical layout. OpenROAD FDP collaboratively develops and maintains design flows using open-source methodologies. By pooling together the expertise of various contributors, the project can create more efficient, flexible, and scalable design flows that benefit the entire community.

2. Data: The project makes relevant and useful data openly available, ranging from chip design datasets to benchmark designs. Access to such data encourages knowledge sharing and accelerates research and development efforts for chip designers. This data availability enables designers to experiment, learn from existing designs, and build upon previous work, fostering a culture of cooperation and innovation.

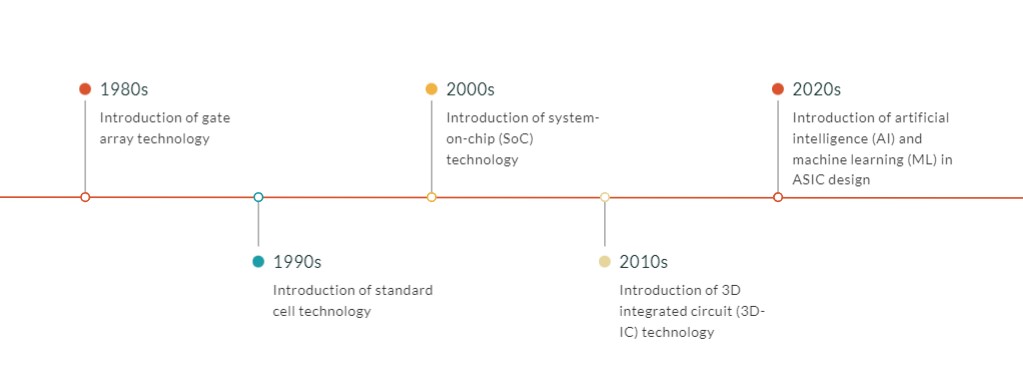
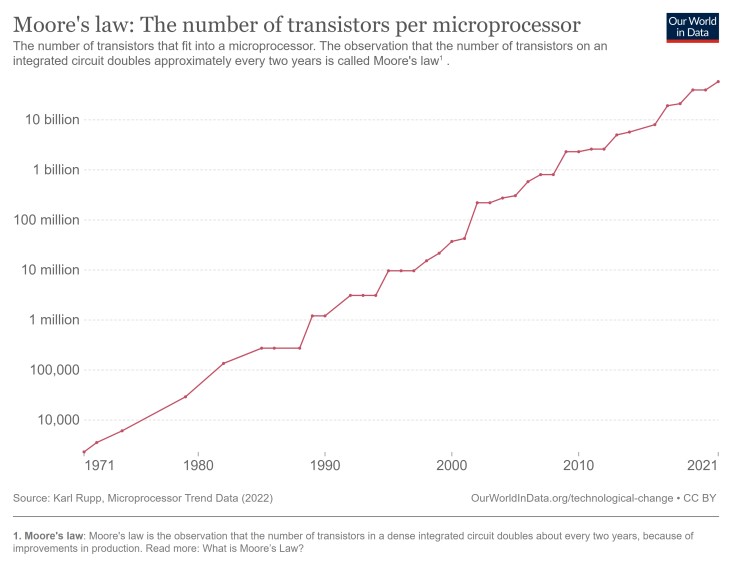
3. Process Design Kits (PDKs): PDKs are essential resources for chip designers, providing a collection of files, models, and rules that define the manufacturing process for a particular semiconductor technology. OpenROAD FDP ensures the development and distribution of open-source PDKs, making it easier for designers to adapt to different process nodes and semiconductor technologies. This accessibility enables a broader range of engineers and researchers to participate in chip design projects without facing licensing constraints.

By combining these three pillars - Flows, Data, and PDKs - OpenROAD FDP endeavors to democratize chip design and encourage the collaboration of a diverse community of engineers and researchers. By lowering the barriers to entry in chip design, the initiative empowers smaller organizations, startups, and academic partners to participate actively in the development of cutting-edge semiconductor technologies.

Through this open-source approach, OpenROAD FDP has the potential to foster innovation, drive faster advancements in chip design, and ultimately revolutionize the semiconductor industry as a whole. By facilitating broader participation and knowledge sharing, the initiative accelerates progress and promotes a more inclusive and collaborative ecosystem for chip design.

**2. OPENROAD:** **DRIVING INNOVATION FOR CHIP DESIGN**

*2.1 Exploring Chip design Innovation Trends:*



source OpenAI

* Moore’s law – 1965 – Every two years chips double in size as cost/transistor reduces.
* Driving principle for the Chip industry to produce smaller, faster and more power.

devices

* ASIC and SoC design methodology – key inflection points in 2000s
* Entire systems on chip leads to complexity EDA software
* Design flow, Database, analyses, verification, packaging
* Disparate tools from vendors, expert software, and hardware engineers

*2.2 Exploring Chip design Innovation Trends:*

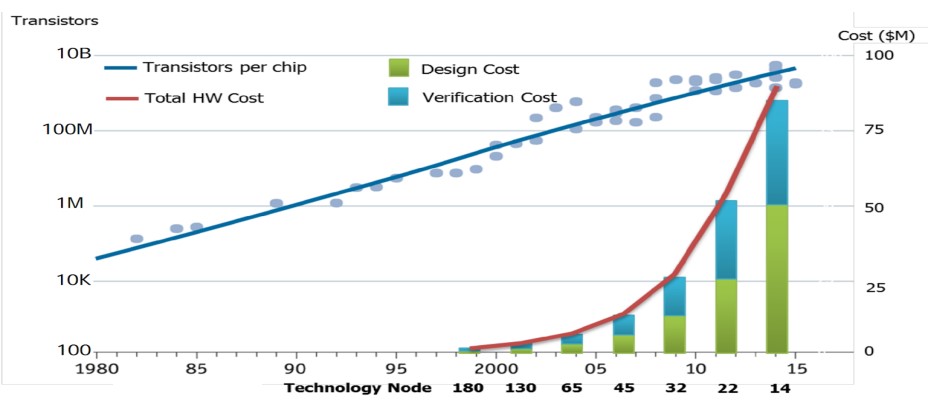
The semiconductor industry heavily relies on innovation to maintain technology leadership and stay competitive. However, there are several challenges that hinder innovation in chip design:

1. High Design Costs: The cost of designing semiconductor chips can be prohibitively high. Licensing fees for Electronic Design Automation (EDA) tools, which are essential for chip design, can be a significant barrier for smaller organizations, startups, and academic researchers.

2. Focus on Latest Technology Node: EDA tools often prioritize delivering solutions for the most advanced technology nodes. This focus on cutting-edge technology can limit the accessibility of chip design tools for those who work on less advanced nodes or have different design requirements.

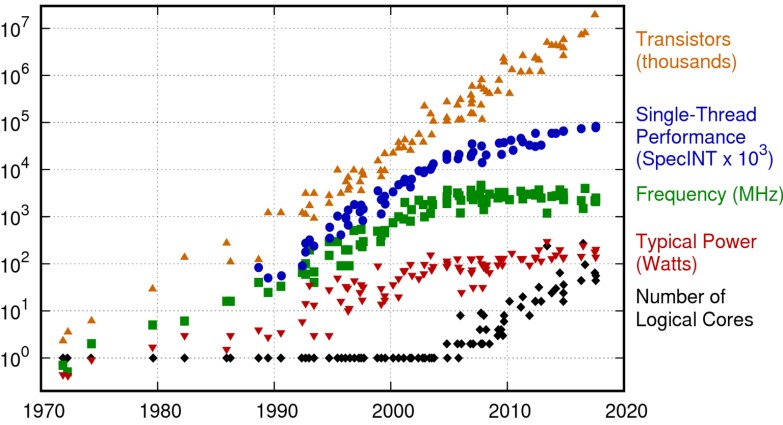
3. Untapped Computational Power: With the increasing power of cloud computing, advanced processors, and GPUs, there is tremendous potential to expedite chip design processes. However, the semiconductor industry has been slow to fully harness this computational power for innovative chip design.

Addressing these challenges is crucial to unlock the full potential of chip design innovation and to ensure that the industry can continue to evolve and advance in technology leadership.



-

18



*2.3 OpenROAD: Open-source Innovation for Chip design*

OpenROAD is an open-source initiative revolutionizing chip design by focusing on accessibility and collaboration. It aims to break down major barriers in chip design innovation:

* Low-cost and Easy-to-install: OpenROAD emphasizes affordability, making it accessible to a broader community. It offers permissive licenses, avoiding complex hardware requirements.
* Simplified Installation: Pre-built binaries and Docker support ease the installation process, reducing setup challenges for users.
* Schedule Barrier: OpenROAD aims to achieve autonomous RTL-to-GDS (Register Transfer Level to Graphic Design System) in just 24 hours, streamlining the design schedule.
* Expertise Barrier: The project aims for no-human-in-loop tapeout, automating the design process to lower the need for expert intervention.
* Cost Barrier: By being open source and supporting low-cost extensions, OpenROAD reduces the financial burden associated with traditional proprietary tools.

Key Features and Benefits:

* Radical Innovation: OpenROAD encourages out-of-the-box thinking and fosters a culture of radical innovation in chip design.
* Rapid Design and Algorithmic Exploration: The initiative facilitates swift chip design and algorithmic exploration, speeding up the development process.
* Collaborative Ecosystem: OpenROAD emphasizes collaboration over competition, fostering an ecosystem where designers and researchers can work together effectively.
* Support for Industry Standards: OpenROAD supports industry standards such as Efabless, SCS, DATC RDF, and Bazel, ensuring compatibility and interoperability.
* Cloud Computing: By harnessing the power of the cloud, OpenROAD taps into abundant computational resources, enhancing design capabilities.
* Machine Learning (ML) Integration: OpenROAD utilizes ML to explore designs with improved Power, Performance, and Area (PPA) metrics, increasing productivity by over 10 times.

Overall, OpenROAD aims to democratize chip design, empower engineers and researchers, and drive chip design innovation to new heights, ensuring the industry remains at the forefront of technology leadership.

*2.4 Recent Innovations in the OpenROAD flow for Large SoCs:*

* Intelligent Design partitioning - TritonPart

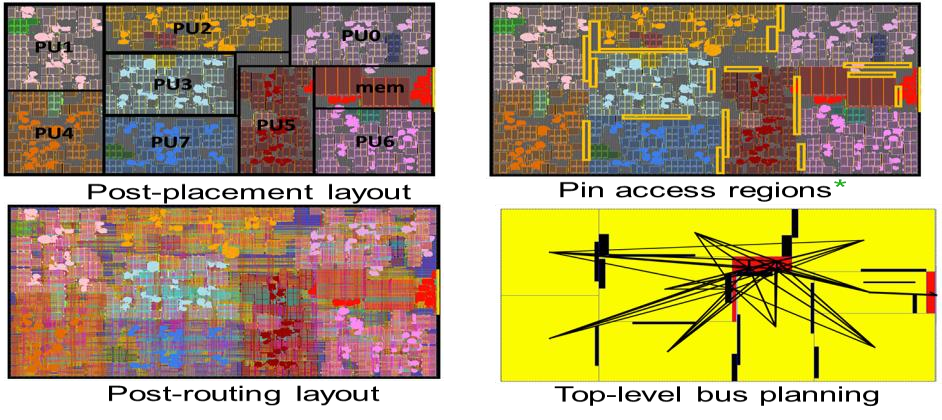
Modern partitioning (hMETIS) is constraints-dominated, does not scale for large designs, large memory and computational requirements.

Triton Part address performance and runtime for large designs

* Hierarchical macro placement –RTLMP

Automatic clustering, pin-aware for timing and routability.

* Low-power design



Multi-voltage domain, UPF

* Support for 3DIC – Future

*2.5 OpenROAD-flow-scripts – Complete RTL-to-GDS*

1. Key features – NHIL < 24 hrs

* Fully Automated flow with flexible flow control
* Incremental optimization - tight integration to Open database and timing engine
* Parasitic extraction for accurate timing signoff

2. Usability

* PDK support - public :sky130, GF180, nangate45, asap7, private: intel 22/16, GF12, tsmc65
* Enhanced GUI for visualization and analysis
* Tutorials, documentation, metrics

3. Power-Performance-Area (PPA)

* Algorithmic enhancements to logic synthesis, clock tree, timing optimization Metrics2.1 driven reporting and tracking

4. ML based Design Exploration: AutoTuner

A screenshot of a computer

Description automatically generated

*2.6 Heat Maps:*



**3. OPENROAD FLOW INSTALLATION IN WINDOWS:**

*3.1 System Requirement*

* Processor – Minimum 4 core with 2.3GHZ
* OS Name – Window 10/11 with WSL installed (Ubuntu22.04)

Memory - > = 16.0 GB

* Disk Capacity - > = 512G

*3.2 WSL Installation*

* Start - > Windows command prompt > Right click and select “Run

as administrator"

* A terminal open
* Type wsl --install for wsl installation.
* After installation, restart the machine.
* If wsl is already installed in the machine, skip the above steps and

proceed for doctor installation in next slide.

*3.3 Docker Installation*

* Go to URL: https://docs.docker.com/desktop/install/windows-install/ and download

Docker Desktop for Windows. If docker already installed skip and continue from

OpenROAD flow installation.

* Double click on the downloaded .exe file and install it.
* Restart the machine after installation.
* Open the installed docker in the machine - > continue without signing -> skip ->

settings icon at top - Resources > WSL integration > Enable integration with my

default WSL distro and also enable “UBUNTU” - > click Apply & Restart.

* Restart the machine.
* Open Ubuntu app (orange in colour) from start -> a new ubuntu terminal opens

*3.4 Enable Docker as Non-root User in Ubuntu*

● Open Ubuntu app (orange in colour) from start -> a new ubuntu terminal opens

● Enter the following two commands

○ sudo groupadd docker

○ sudo usermod -aG docker $USER

○ sudo apt-get update; sudo apt-get upgrade; sudo apt install -y

build-essential python3 python3-venv python3-pip make

● Exit Ubuntu terminal

*3.5 Verify Docker Installation in Ubuntu*

● Open Ubuntu app (orange in colour) from start -> a new ubuntu terminal opens

● Enter the following two commands

○ newgrp docker

○ docker run hello-world

● You should see the message in the terminal - Hello from Docker!

*3.6 Open Road Flow Installation*

Open Ubuntu app (orange in colour) from start -> a new ubuntu terminal opens

● Enter the following commands one by one

○ mkdir ORFS && cd ORFS

○ git clone --recursive

https://github.com/The-OpenROAD-Project/OpenROAD-flow-scripts

○ cd OpenROAD-flow-scripts

○ ./build\_openroad.sh --threads 1

**4. OPEN POWER:**

OpenPOWER is a high-performance RISC-based free

and open Instruction Set Architecture…

● With 30 year proven computing history

● Completely driven by open collaboration through the foundation

● Enabling freedom of design across all domains and industries

● Including reference design, tools, and resources to de-risk Development.

*4.1 BENEFITS OF OPEN POWER:*

● Design risk

● Cost of entry

● Partner limitations

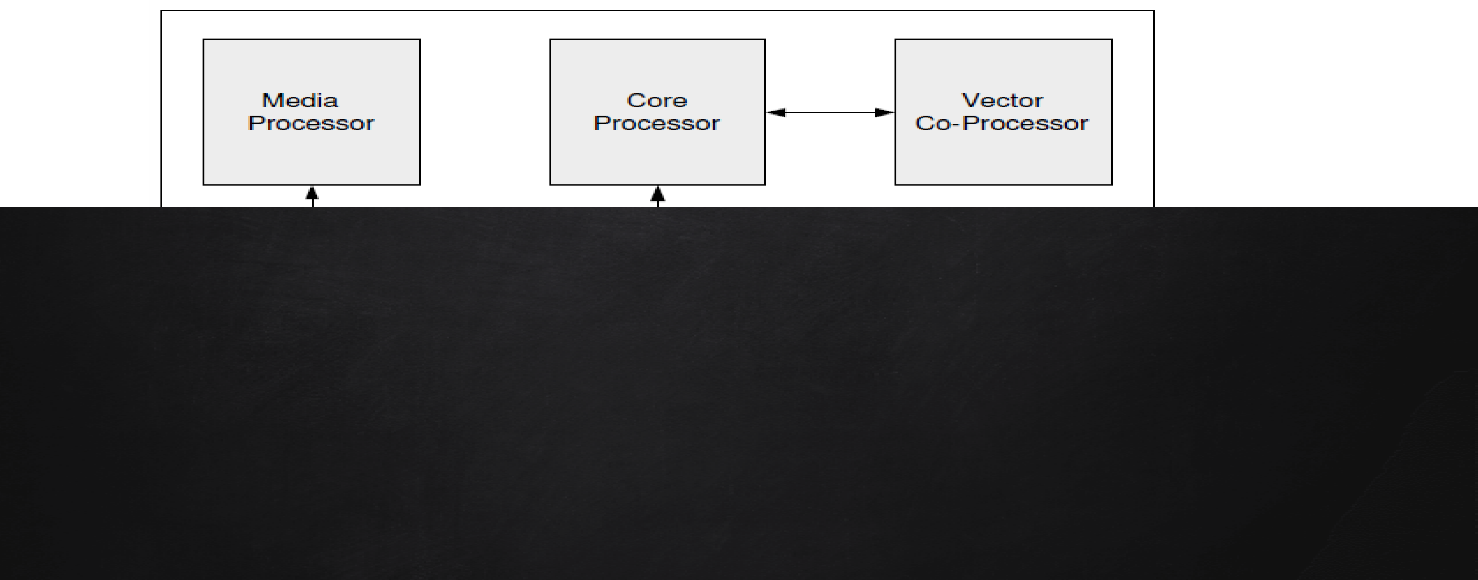
● Supply chain

**A screenshot of a computer

Description automatically generated***4.2 USES OF OPEN POWER :*

**5. SYSTEM ON CHIP (SOC):**

*A diagram of a diagram

Description automatically generated5.1 Basic system-on-chip model*

*5.2 SOC vs processors on chip*

*5.3 Basics Elements of System on Chip*

System On Chip are now very complex systems in a single chip, designing is very challenging, and the right tools are needed:

* Collections of base elements to reuse (Cores, IPs).
* Right tools to aggregate all togethers.

|  |  |  |
| --- | --- | --- |
|  | ***System on chip*** | ***Processors on chip*** |
| **processor** | **multiple, simple, heterogeneous** | **few, complex, homogeneous** |
| **cache** | **one level, small** | **2-3 levels, extensive** |
| **memory** | **embedded, on chip** | **very large, off chip** |
| **functionality** | **special purpose** | **general purpose** |
| **interconnect** | **wide, high bandwidth** | **often through cache** |
| **power, cost** | **both low** | **both high** |
| **operation** | **largely stand-alone** | **need other chips** |

1. Clocking:

System on Chip are mostly synchronous and require a clock to cadence the system.

Phase Locked Loop circuits are used to create the various clock frequencies of the system.

A system can have use multiple clocks, but special care must taken to transmit data from one clock domain to another.

2. CPU:

Brain of the system, execute sequential instructions.

Embedded systems almost all use ARM CPUs (AXY). Core are used as IPs and integrated in systems.

ARM is proprietary and you have to pay royalties to use in your system. OpenPOWER Microwatt & RISC-V are successful free alternatives.

3. Memory Mapped Interconnect:

* Buses
* Connect Master(s) to Slave(s)
* Arbiters/Muxes

4. Streaming Interconnect:

* Data Flow
* Connect Master(s) to Slave(s)
* Arbiters/Muxes

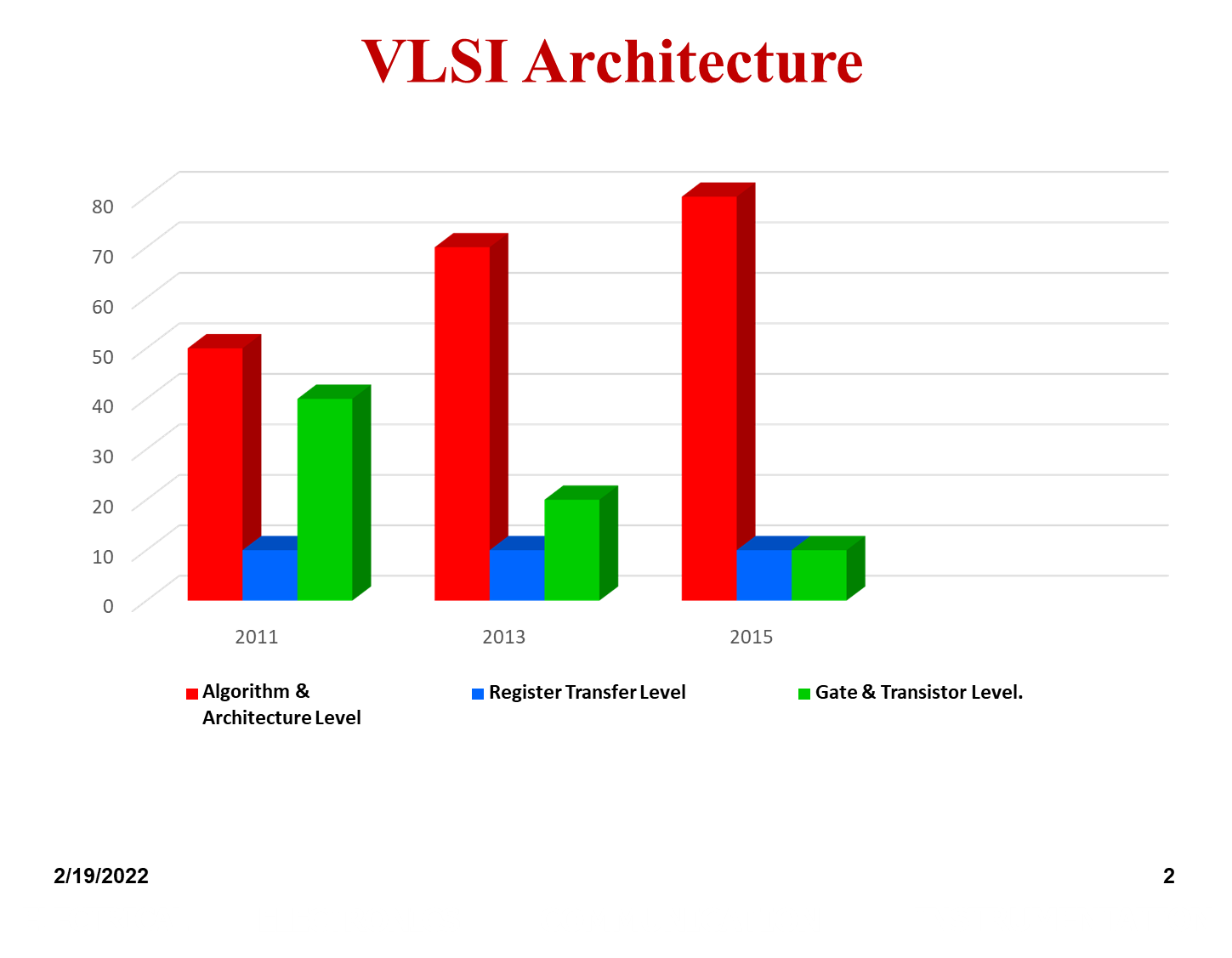
Think of it like “plumbing”:

Connect this data flow source to this data flow sink.

5. DMA:

* Avoid involving the CPU to copy data from one location to another.
* CPU is only used to start DMA and interrupted when DMA is done. It can do others tasks while DMA is in progress

**5. VLSI ARCHITECTURE:**

****

VLSI Architecture Example 1

Ex: Behavioural Model of an Adder

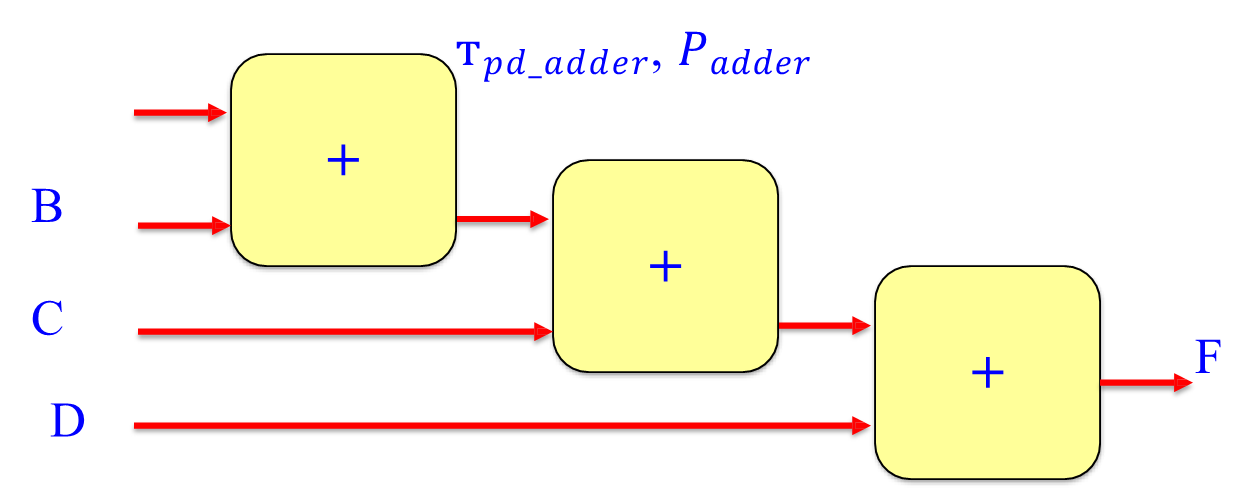
Z= A + B + C + D

*Architectural choices*

To realize digital hardware solution for a given behavioural description - creation of suitable architectural plan

OPTION1:

A



A table with text and symbols

Description automatically generated

F



Option 2:

A

B

C D

**+**



**+**

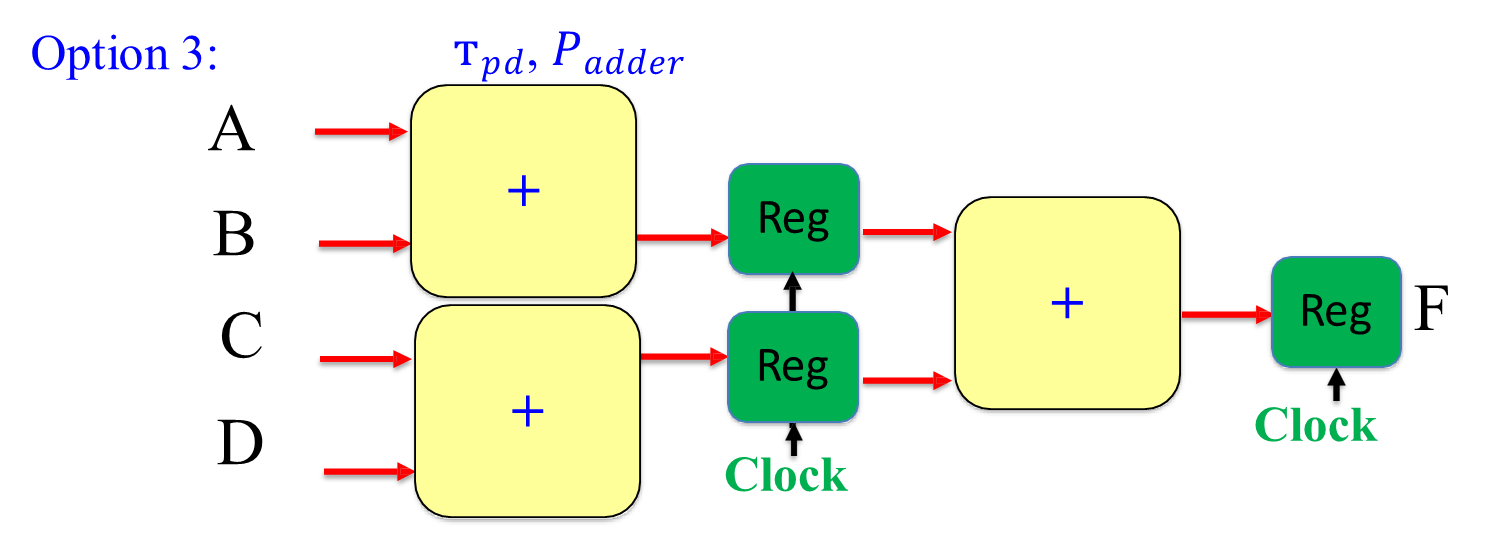


**+**

т𝑝𝑑\_𝑎𝑑𝑑𝑒𝑟, 𝑃𝑎𝑑𝑑𝑒𝑟

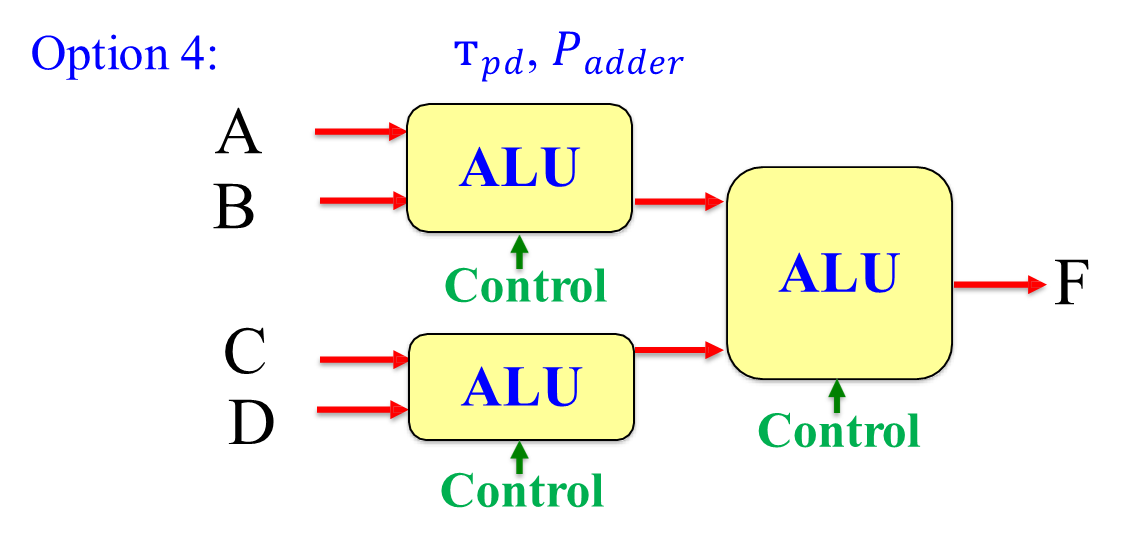
A table with text and symbols

Description automatically generated



A table with text and symbols

Description automatically generated



A table with text and symbols

Description automatically generated

**7. CONCLUSION:**

In conclusion, OpenROAD FDP is a pioneering open-source initiative that aims to revolutionize the chip design landscape by promoting accessibility and collaboration among engineers and researchers. The project's three key components - Flows, Data, and Process Design Kits (PDKs) - work together to break down the barriers that traditionally hindered smaller organizations, startups, and academic partners from actively participating in chip design.

With collaborative development and maintenance of design flows, OpenROAD FDP creates more efficient and scalable processes, benefiting the entire chip design community. By openly sharing relevant data, including datasets and benchmark designs, the initiative fosters knowledge sharing and accelerates research and development efforts, encouraging cooperation and innovation.

One of the critical aspects of OpenROAD FDP is the provision of open-source PDKs, which define the manufacturing process for different semiconductor technologies. This accessibility enables a broader range of designers to adapt to various process nodes without facing restrictive licensing constraints.

Through its inclusive and cooperative approach, OpenROAD FDP seeks to democratize chip design and inspire diverse engineers and researchers to actively engage in cutting-edge semiconductor technology development. By lowering entry barriers, the initiative empowers a more diverse community to contribute to chip design, thereby accelerating progress and driving faster advancements in the industry.

Overall, OpenROAD FDP's open-source philosophy has the potential to transform the semiconductor industry by fostering innovation and creating a more inclusive and collaborative ecosystem for chip design. With its emphasis on accessibility and knowledge sharing, this initiative paves the way for a brighter and more collaborative future in chip design.