SOC BASED SYSTEM DESIGN FOR HEP APPLICATIONS

By

Dr. Swagata Mandal

Assistant Professor

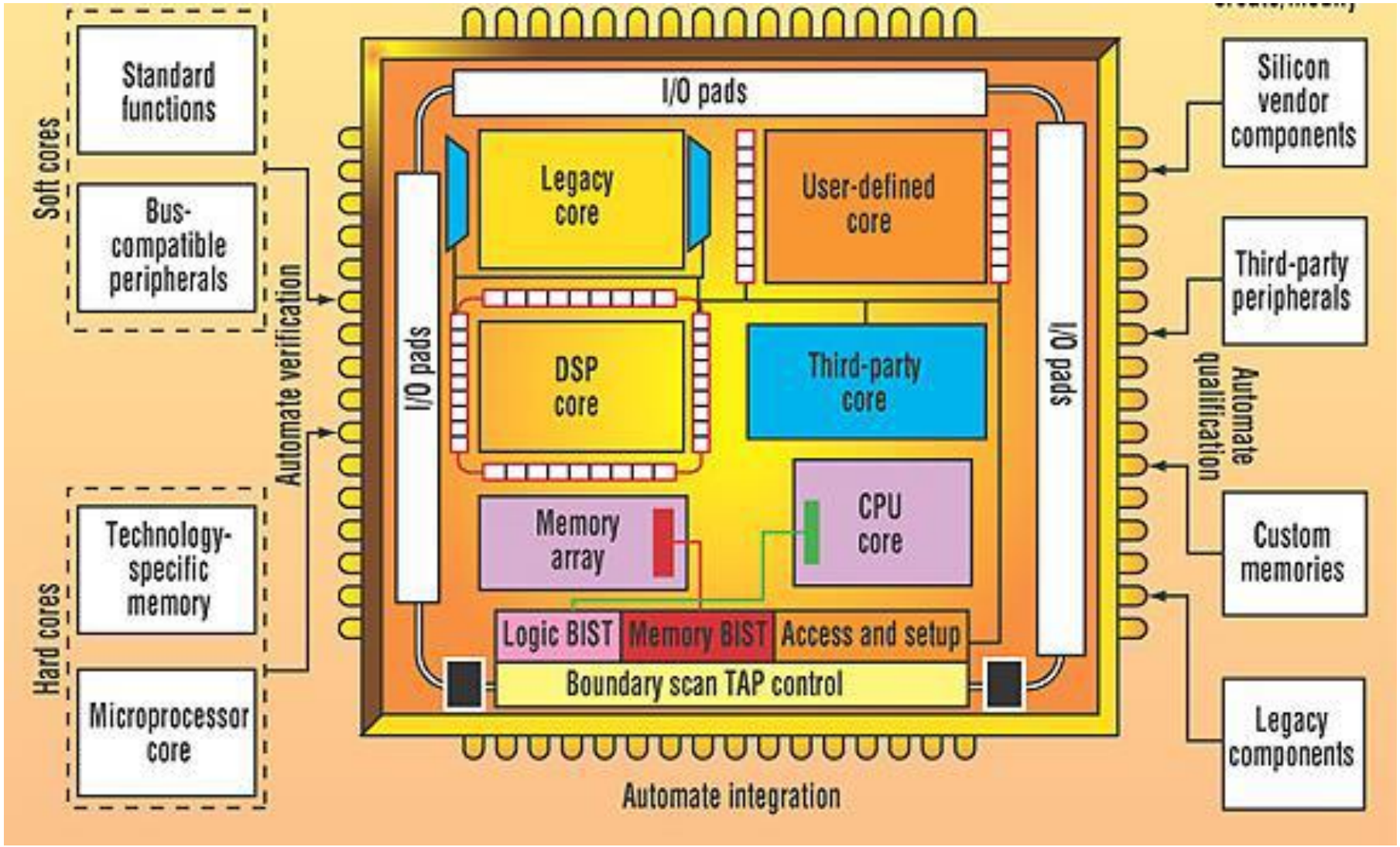
Department of Electronics and Communication

Jalpaiguri Government Engineering College

INTRODUCTION

- A complex integrated circuit (IC) that integrates major functional elements into a single chip.
- In addition to IC, SoC consists of software and interconnection structure for integration.
- May contain digital, analog, mixed-signal functions in a single chip substrate
- The components of SoC include CPU, GPU, Memory, I/O devices, etc.
- Compared to a multi-chip architecture, an SoC with equivalent functionality will have increased performance and reduced power consumption as well as a smaller semiconductor die area.
- Due to their low power consumption, SoCs are common in the mobile electronics market and finds a wide use in the arena of embedded systems.

BASIC ARCHITECTURE OF SOC



INTERNAL ARCHITECTURE OF SOC

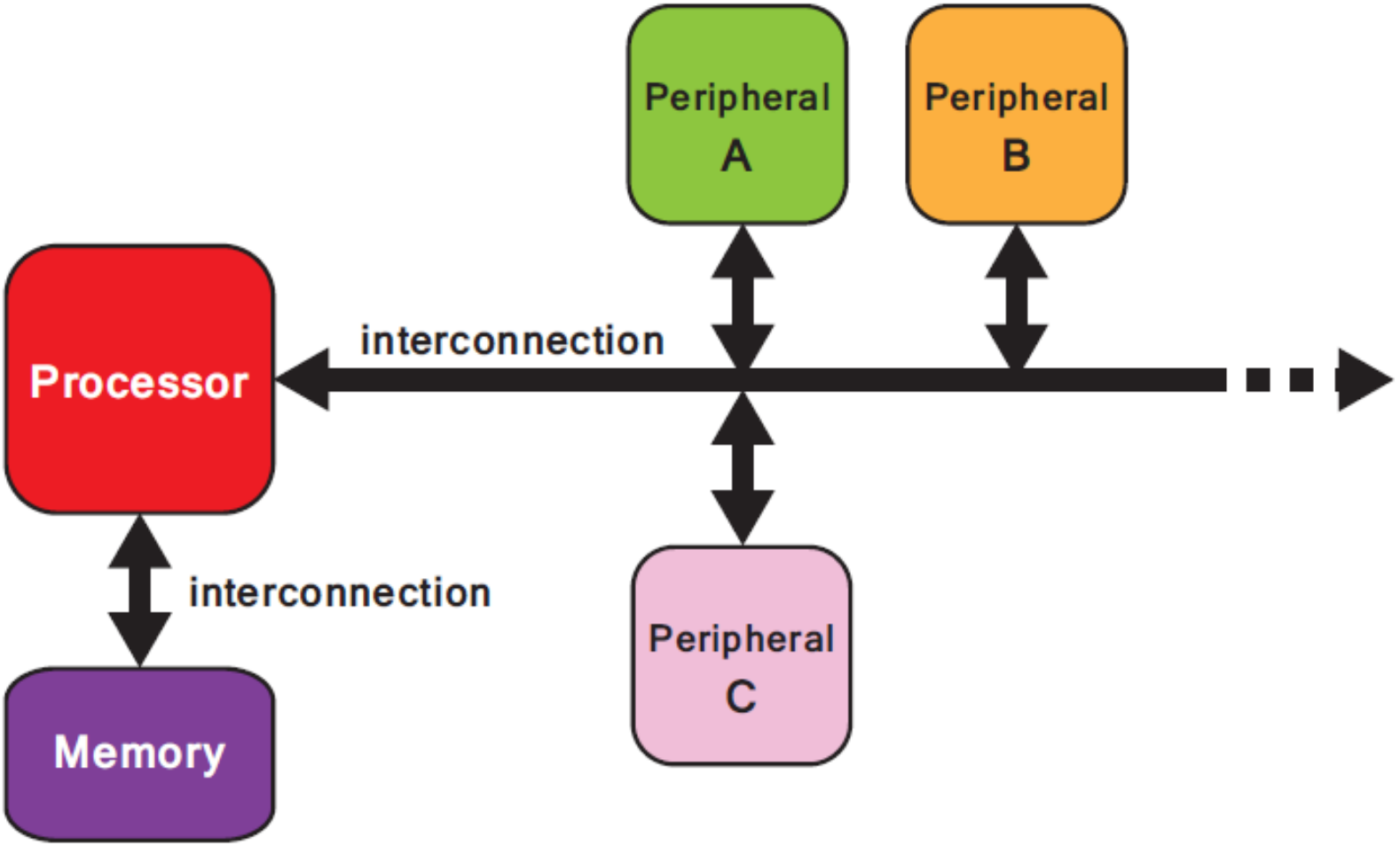
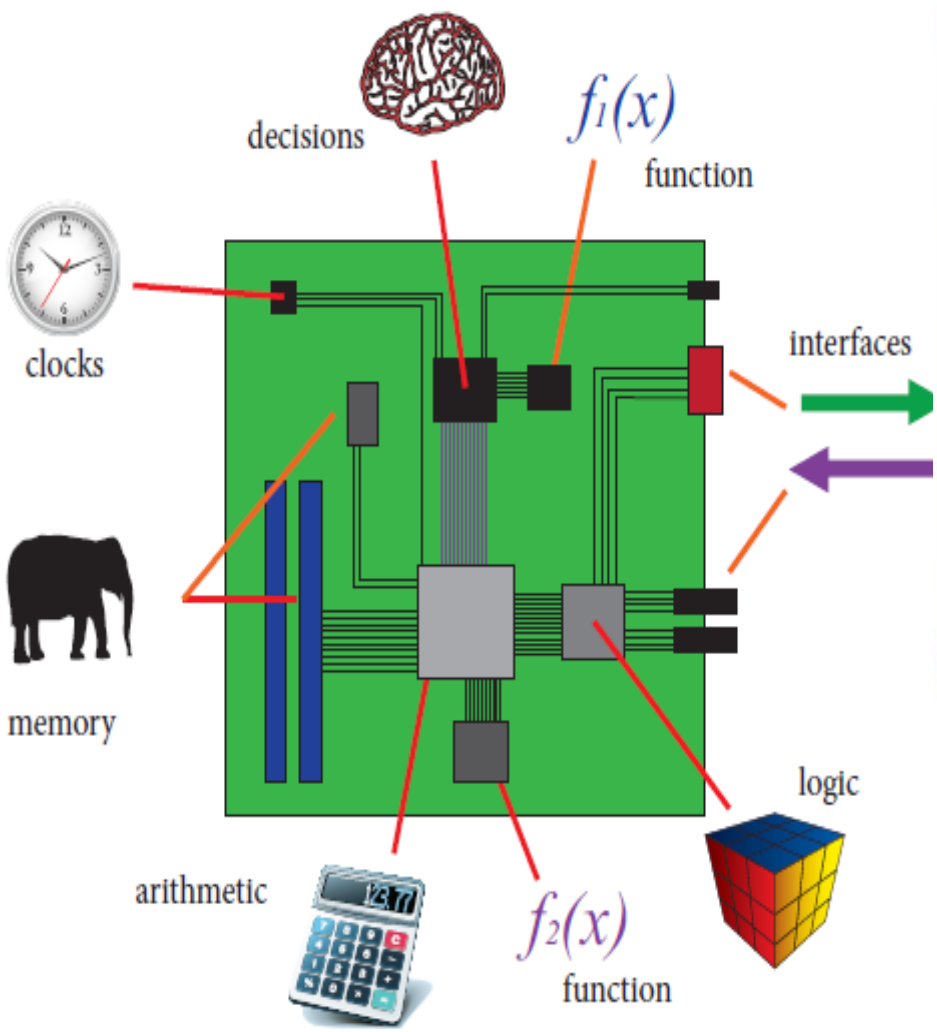
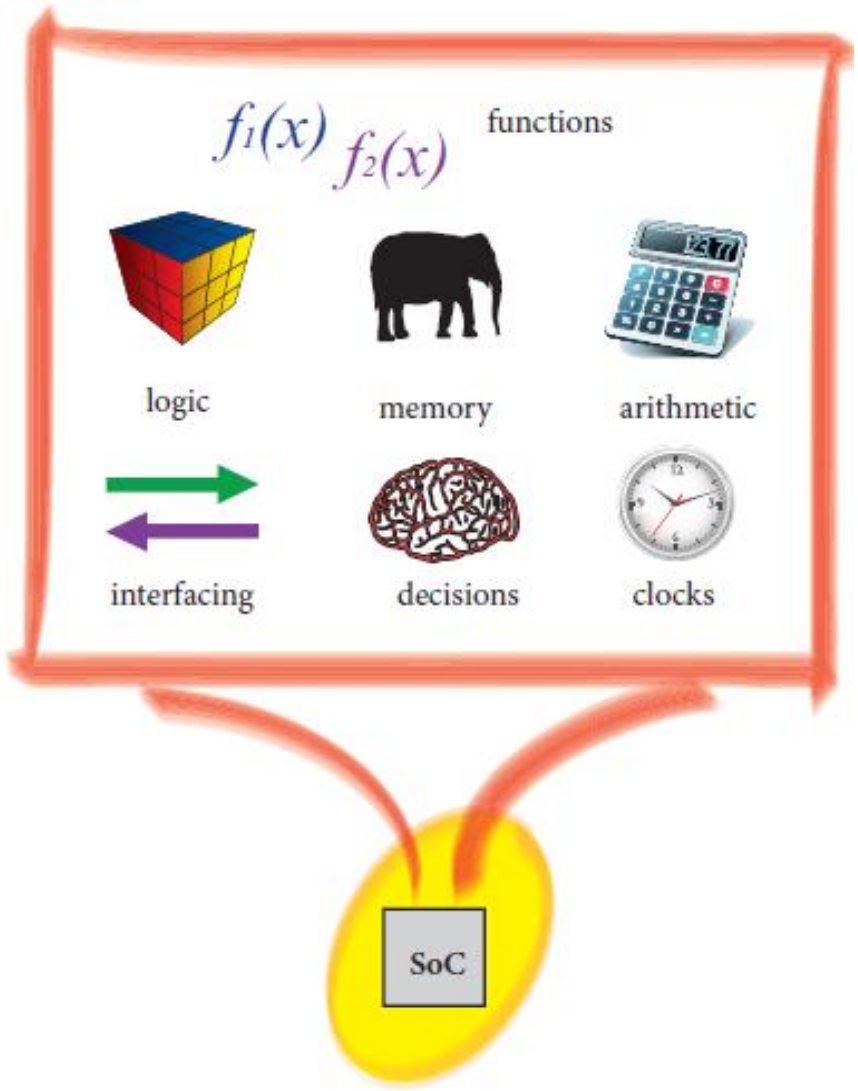


Fig: Hardware system architecture of an embedded SoC

SOB VS SOC





System on Board

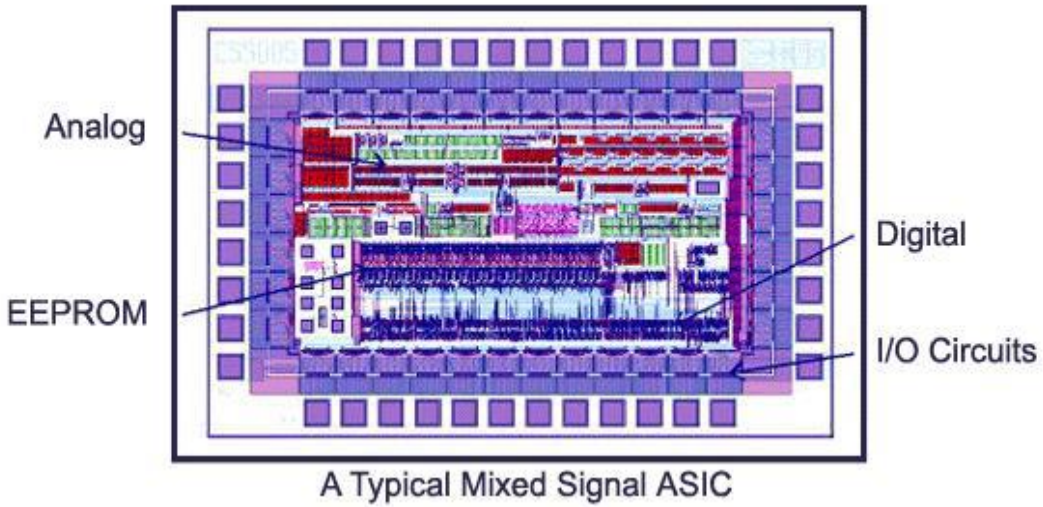
System on Chip

Advantages & Disadvantages

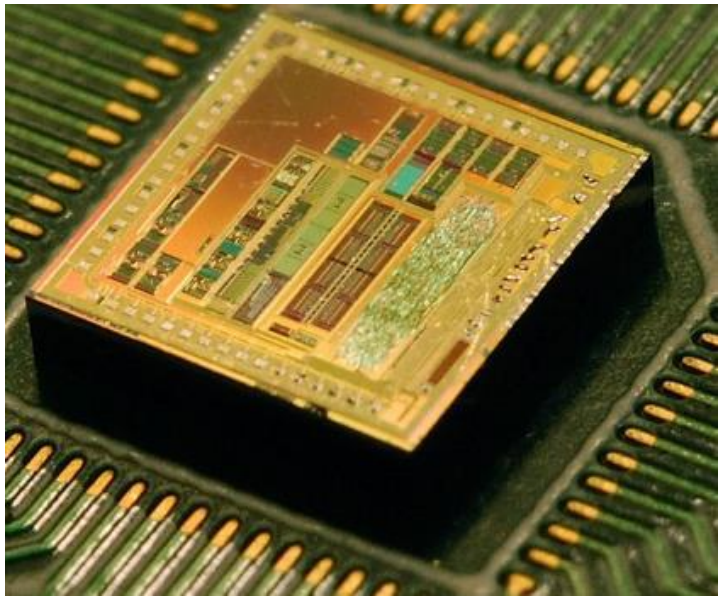
- **Advantages of SoC**
  - Low power.
  - Low cost.
  - High reliability.
  - Small form factor.
  - High integration levels.
  - Fast operation.
  - Greater design.
  - Small size.
- **Disadvantages of SoC**
  - Fabrication cost.
  - Increased complexity.
  - Time to market demands.
  - More verification.

ASIC TO SOC

- **ASICs:** Application Specific ICs are close to SoC designed to perform a specific function for embedded and other applications.
- ASIC vendors supply libraries for each technology they provide. Mostly, these libraries contain pre designed/verified logic circuits.
- SOC is an IC designed by combining multiple stand alone VLSI designs to provide a functional IC for an application. It composes of pre designed models of complex functions e.g. cores (IP block, virtual components, etc.) that serve various embedded applications.



A Typical Mixed Signal ASIC



ZYNQ SOC

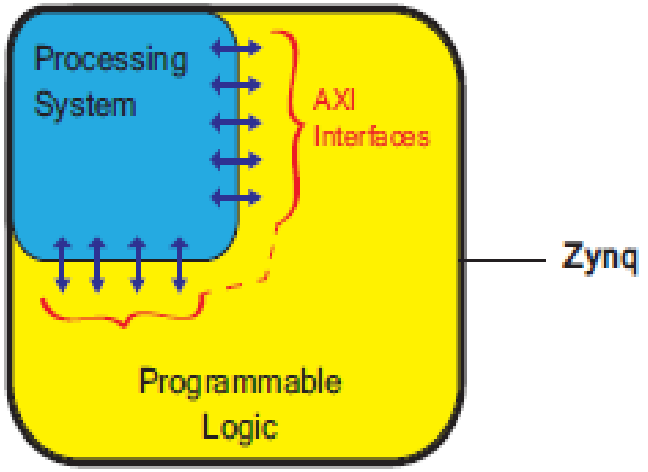


Fig: A simplified model of the Zynq architecture

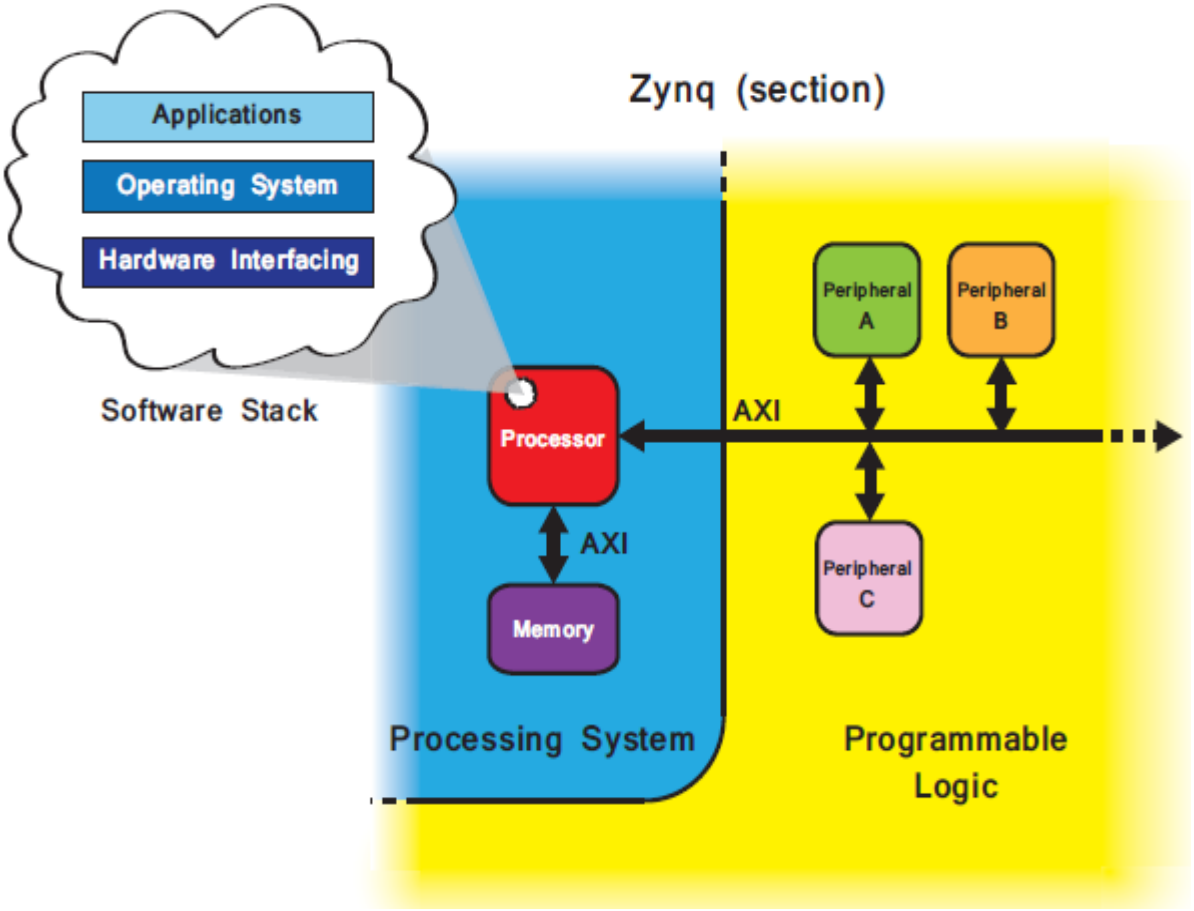
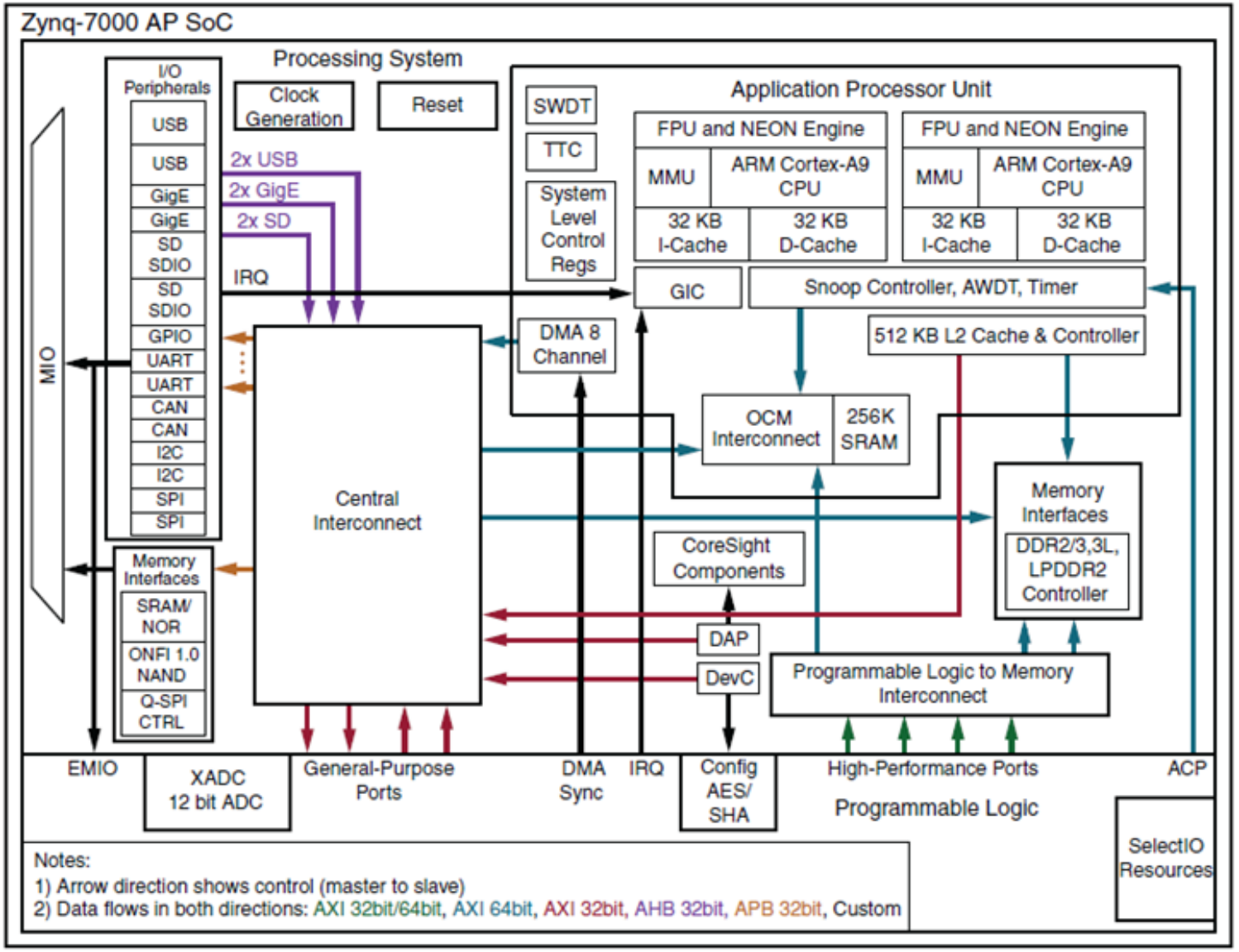


Fig: Relationship of the software system, hardware system, and Zynq architecture

ZYNQ-7000 SOC BLOCK DIAGRAM



System Design for HEP

- Important part of HEP experiments are the detectors, Data acquisition system and trigger & control unit.
- Variety of particle detectors have been developed over the years to measure the energy, direction, spin, charge, etc. of particles.
- High speed electronics are used within detectors to capture the particle data and send this data to external computer systems for post-processing.
- SoC can be a good solution for development various component of data readout and trigger unit.
- Due to presence of FPGA based processing logic and processing system on the same platform it can be useful for data processing in space and time constrained applications.

Architecture Vulnerabilities

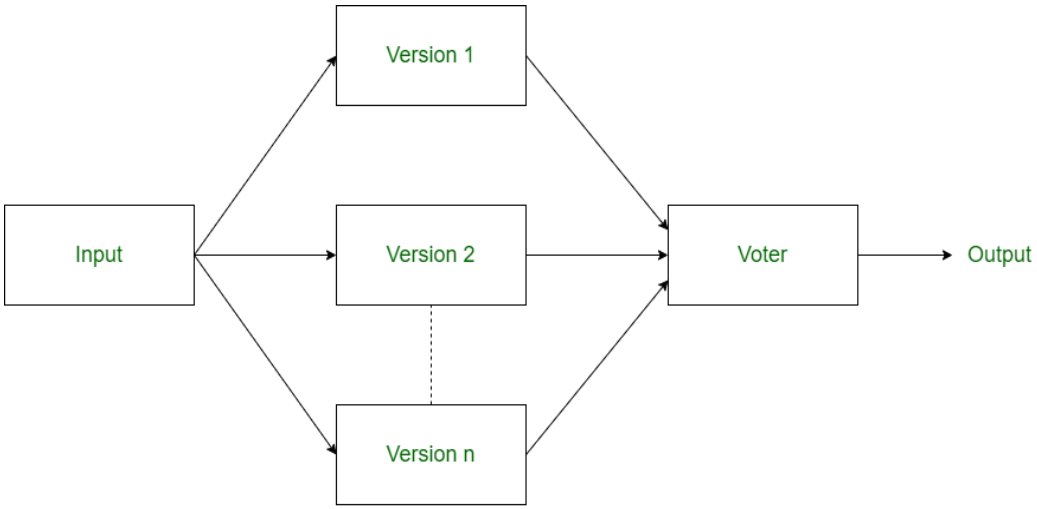
- An intense radiation field is generated from the high energy particle collisions within HEP experiments.
- The actual radiation environment depends heavily on the experiment itself and on the location within the experiment.
- Radiated charge particles can incur faults in various location of SoC:
  - ❑ Memory Unit (Both flash and Block memory)
  - ❑Configuration memory of PL unit
  - ❑PS unit
  - ❑PS & PL interface
- Nature of fault:
  - ❑Temporary or soft error
    - ➢ Transient error
    - ➢ Single Bit Upset
    - ➢ Multi-bit Upset
  - ❑Permanent or hard error

ERROR MITIGATIONS IN PL UNIT

- Hardware Redundancy
  - ✓ TMR, CED, state machine encoding, special purpose placement and routing, design diversity redundancy, reduced precision redundancy, duplication with compare
- Configuration Scrubbing
  - ✓ Used in configuration memory of PL unit
- Error Correction Coding:
  - ✓ BCH, Hamming, Product code for both SBU & MBU
- Flip-Flop Mitigation:
  - ✓ the state of the flip-flops can be protected from SEUs by exploiting redundancy
- System-Level Mitigation:
  - ✓ Use of watch-dog timers, radiation-hardened system monitors, and system checkpointing.

ERROR MITIGATIONS IN PS UNIT

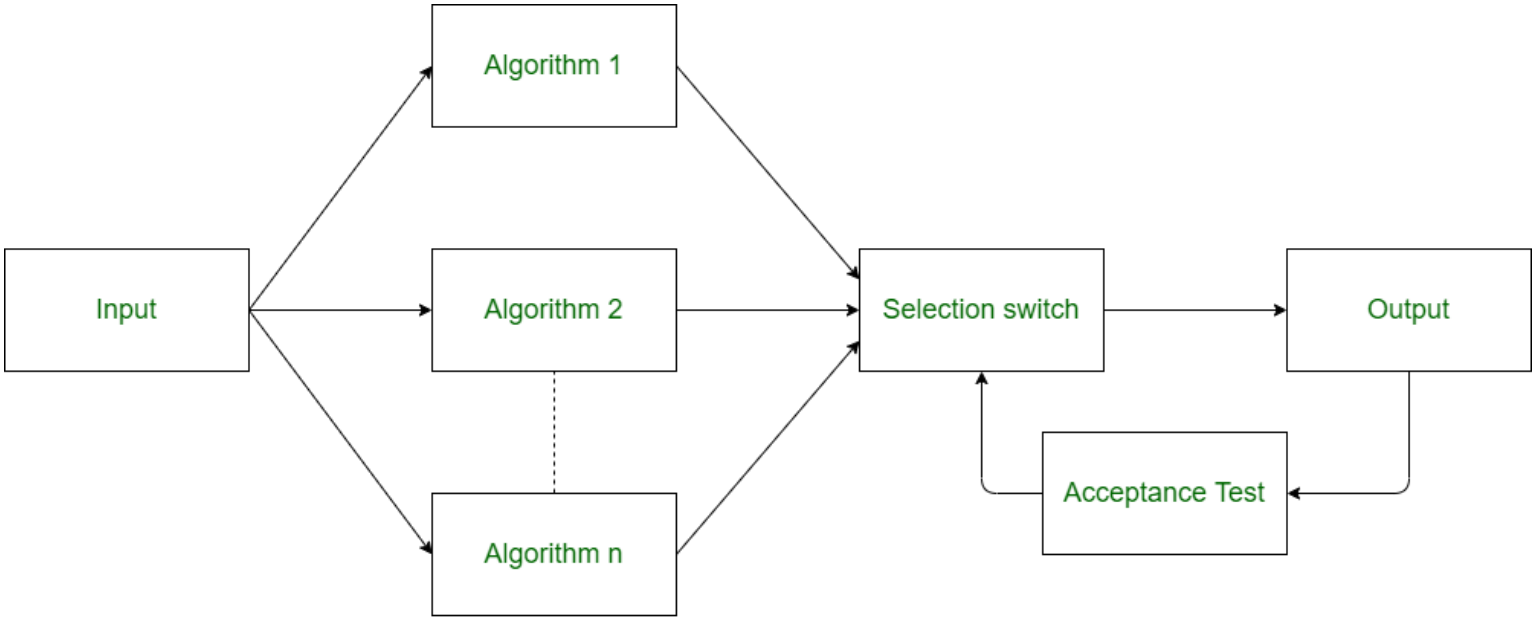
- N-version Programming: In N-version programming, N versions of software are developed by N individuals or groups of developers.
- N-version programming is just like TMR in hardware fault-tolerance technique.
- In N-version programming, all the redundant copies are run concurrently and result obtained is different from each processing.
- The idea of n-version programming is basically to get the all errors during development only.



N-version Programming

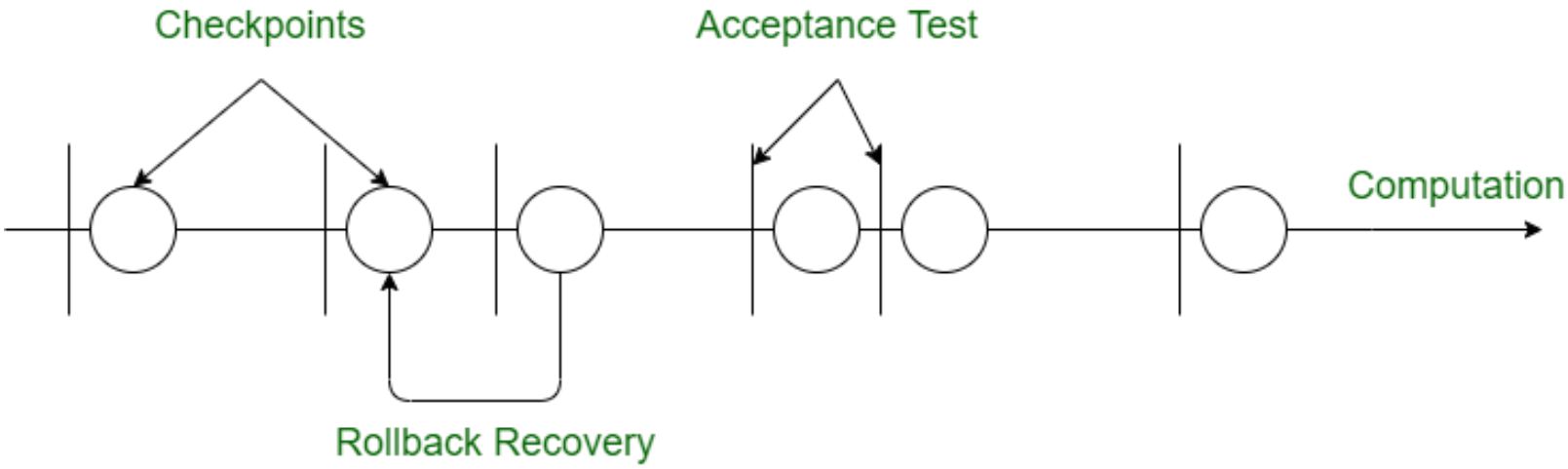
RECOVERY BLOCKS

- Recovery blocks technique is also like the n-version programming but in recovery blocks technique, redundant copies are generated using different algorithms only.
- In recovery block, all the redundant copies are not run concurrently and these copies are run one by one.
- Recovery block technique can only be used where the task deadlines are more than task computation time.



Check-pointing and Rollback Recovery

- This technique is different from above two techniques of software fault-tolerance.
- In this technique, system is tested each time when we perform some computation.
- This techniques is basically useful when there is processor failure or data corruption.



THANK YOU