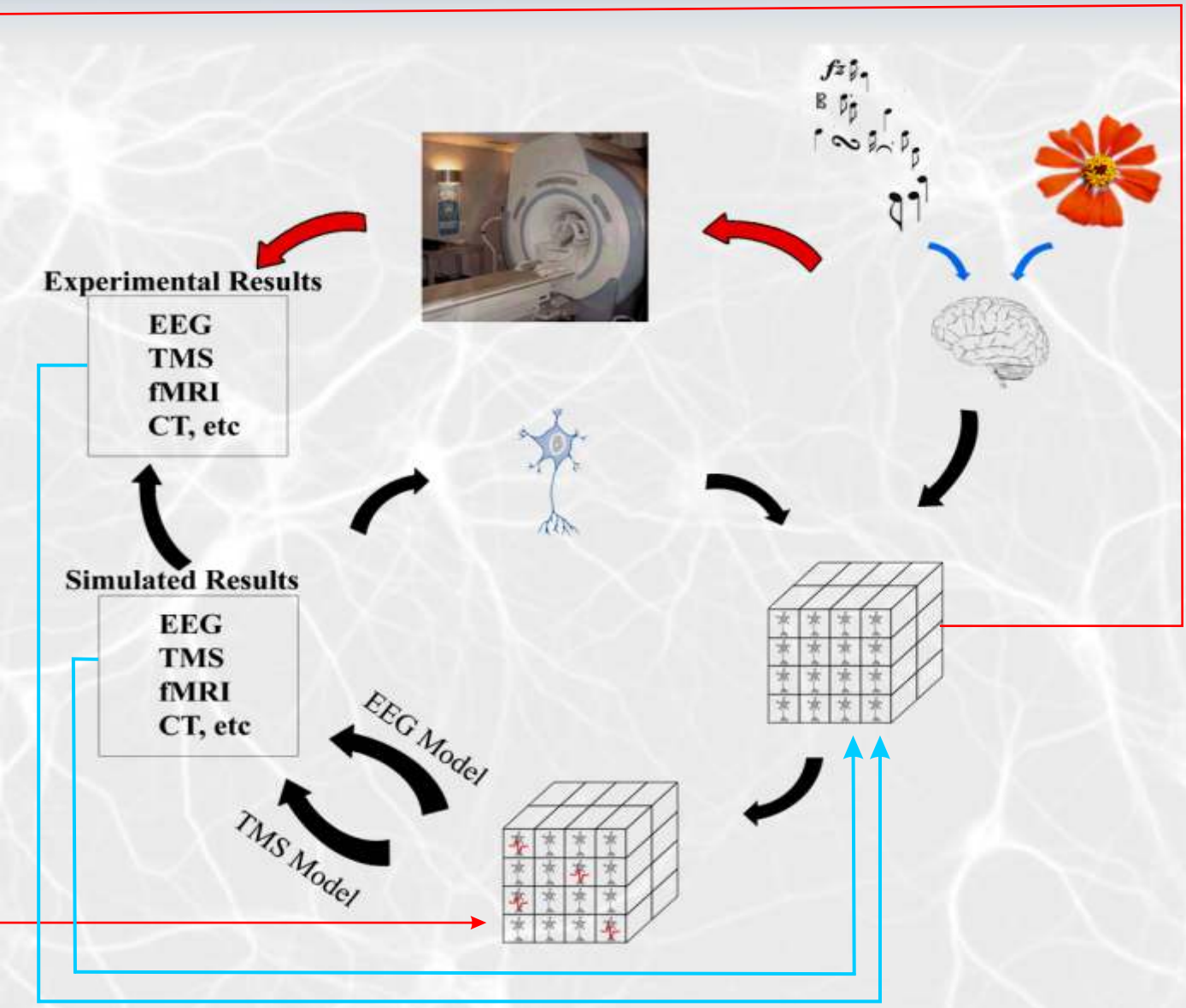
PRADHĪKALANĀ



WARFT

FUTURE GENERATION SUPER-COMPUTING CLUSTER : Traces of Multiple Applications run concurrently within every Node

MEMORY IN PROCESSOR SUPER-COMPUTER ON A CHIP (MIP SCOC)

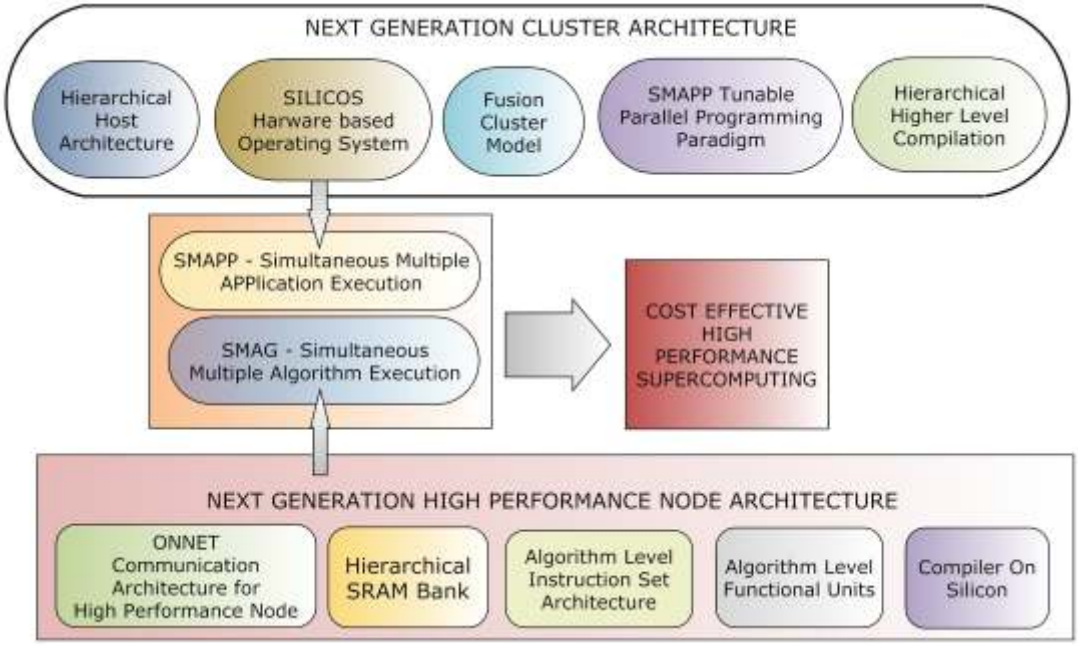
The immense computational demand imposed by the THE MMINi-DASS PROJECT, has given rise to the novel supercomputer design paradigm known as the MIP SCOC. The MIP approach attempts a very fine grain physical and logical integration of memory and logic by incorporating the memory within the logic. In the MIP SCOC architecture, memory is physically and logically integrated with the functional units of the processor. This bit-level integration of processing logic and memory has led to a tremendous increase in functionality of a single MIP SCOC node. The MIP SCOC architecture includes powerful ALFU (Algorithm Level Functional units) like chain matrix adders, multipliers, sorters, multiple operand adders and graph theoretic units like Depth-First-Search, Breadth-First-Search. This introduces a higher level of abstraction through the algorithm-level instructions (ALISA). A single ALISA is equivalent to multiple parallel VLIW. The MIP SCOC architecture includes an on-chip compiler (Compiler-On-Silicon) to generate the required instructions to feed the ALFUs of the MIP node. The Primary COS (PCOS) partitions the incoming problem according to the algorithms involved. Each SCOS generates the instructions corresponding to that column. A distributed control design is employed specific to ALFU population type (forming different heterogeneous cores) enabling parallel operation of a very large number of ALFUs.
The MIP SCOC paradigm helps Simultaneous Multiple Application (SMAPP) execution in a cluster where traces of different applications run concurrently within every node unlike the conventional approaches. The SMAPP concept besides being flexible enough for cost sharing across multiple users, will provide the requisite performance for individual applications by utilizing the heterogeneous cores efficiently across the different application traces.

HIERARCHICAL HOST ARCHITECTURE - FOR SMAPP EXECUTION

Execution of multiple applications, exchanging data among them and scheduling cannot be handled by a single host. Thus functionalities of host are distributed across the hierarchical host planes in a hybrid pyramid structure(refer back-cover page). The core OS functionalities are implemented on a hardware platform in SILICOS, the rest are formulated as software libraries residing in primary and secondary hosts. A host processor, essentially an ASIC, effectively implements OS concepts at the MIP SCOC cluster. Primary host handles the complexity involved in simultaneous multiple applications execution, and exchange of information and control. Secondary host manages scheduling, memory, pre-processing of multiple application and the I/O operations. The SILICOS by using ALFUs processors incorporate the functionalities of the cluster operating system at SMAPP level. The feed rate and the load balancing of multiple applications by host system should match with the high computational strength of hundreds of thousands of MIP nodes.

SILICON OPERATING SYSTEM

A software OS may not be proficient enough to exploit the power of the underlying MIP based nodes and match its computational speed. To simultaneously load balance multiple applications' workload when mapped across the nodes of a cluster, a hardware based operating system SILICOS, is required for handling the complexities associated with parallel mapping and data tracking of the huge amount of data associated with the different applications. However, this mapping complexity is tackled by MIP SCOC paradigm unlike the ALU based conventional cores. In this scenario, the reliability of the operating system, memory management, process scheduling, I/O handling and interrupt handling is of paramount importance particularly when dealing with million node clusters. SILICOS plays a major role in tackling the mapping complexities and sequencing the terabytes of data resulting from the simultaneous execution of several applications. Pre-processing at the cluster aids efficient execution by reducing the compiling complexity, leaving only low-level scheduling for the node.



SMAPP TUNABLE PARALLEL PROGRAMMING PARADIGM

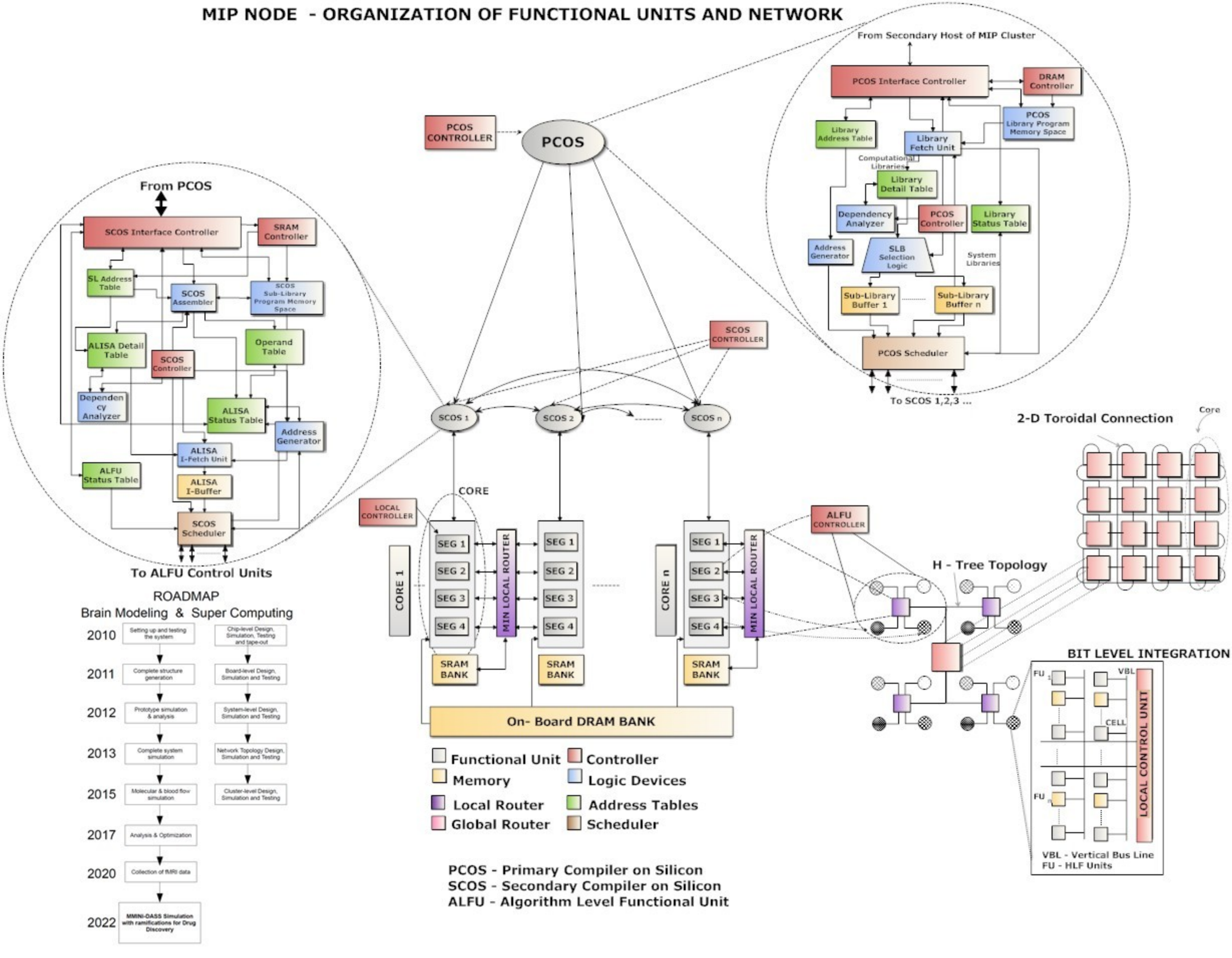
Parallel mapping of multiple applications across very large cluster will be a burden on the SILICOS and require a programming paradigm exploiting hardware level parallelism called PPL. This is very well supported by MIP SCOC paradigm. The Parallel Programming Language (PPL) model handles both data parallelism and communication model efficiently unlike current data parallel programming languages and explicit communication models where the user is completely responsible for the creation and management of processes and their interactions with one another. The PPL model is a simple, object-based and portable so that it is easy to represent, understand, and modify complex logic efficiently. The PPL constructs are capable of exploiting the high level of parallelism inherently present in the application matching the underlying MIP architecture (ISA of the node architecture).

HIERARCHICAL HIGHER LEVEL COMPILATION

Although there are schedulers to provide instructions to every part of the node, usage of ALISA burdens them. Backward-compatibility, instructions dependency and scheduling of the instructions to large number of functional units need to tackled by the compiler. This requires a higher level compiler-scheduler at the host level. Compilation process required by multiple parallel ALISAs, is a complex and demanding process and requires effective compiler to have a considerable impact on the node's performance.

Implicit detection of parallelism across hundreds of thousands of instructions in massive applications, greatly adds to the delay associated with the compiler. Besides, across a very large instruction set , this scheme is difficult to detect. Issues such as backward-compatibility, runtime optimization and dependency are added problems. Implementing the dependency analysis and scheduling done by the compiler, in hardware, will drastically reduce the compilation process delay. This calls for a design of a hardware based compiler on silicon (COS). While higher level compiler jobs such as Lexical and Semantic analysis are done by software based compiler, the Computation node is relieved of this burden and functions such as resource allocation and instruction generation are taken care by COS.

MIP NODE - ORGANIZATION OF FUNCTIONAL UNITS AND NETWORK



ROADMAP

Brain Modeling & Super Computing

2010	Setting up and testing the system	Chip-level Design, Simulation, Testing and tape-out
2011	Complete structure generation	Board-level Design, Simulation and Testing
2012	Prototype simulation & analysis	System-level Design, Simulation and Testing
2013	Complete system simulation	Network Topology Design, Simulation and Testing
2015	Molecular & blood flow simulation	Cluster-level Design, Simulation and Testing
2017	Analysis & Optimization	
2020	Collection of fMRI data	
2022	MMINI-DASS Simulation with ramifications for Drug Discovery	

NODE ARCHITECTURE : ALFU, ALISA, ONNET AND MIP CELLS

The MIP SCOC architecture exploits the opportunities afforded by the use of a single common underlying implementation technology to integrate physically and logically the separate functions of computation logic and storage. This way, a global uniform computing resource medium can be realized through replication of the modular units (MIP Cells).The repetitive use of these units in an orderly fashion would result in the design of Higher Level Functional Units capable of performing computations at the Algorithm Level.

ALFU, ALISA AND MIP CELLS

Conventional ALU based scalar / vector floating point functional units require complex compilation to break-down and execution of large applications. Instead, basic higher-level matrix, graph theoretic, vector and scalar units are utilized in MIP, which enable faster execution of the instructions. These ALFUs are unique and based on specific MIP cells where bit level integration of memory and processor is achieved. ALISA removes control dependencies, reduces the overall number of memory cycles and extracts the performance of the underlying ALFUs by exploiting hardware level parallelism. The dynamic power consumption of ALFUs when compared to ALUs is considerably less due to the decreased instruction count.

ONNET

Apart from a network processor connecting several nodes, communication network within the node having homogeneously structured heterogeneous cores made up of hundreds of ALFUs is required to transfer data between processing tiles and memory units. The On-Node NETwork architecture, adopts a zero layer approach unlike the existing Network on Chip (NoC) architectures as performance is affected by the network latency. To satisfy the random traffic pattern inside the node distributed network, tiles dedicated connection to the functional units are used. A hierarchical Multi-stage Interconnection Network switching structure efficiently distributes data thereby increasing memory and I / O bandwidth and thus the node's processing power unlike the cross bar switches in conventional NOCs.

CLOCKING SCHEME

To initiate and sustain execution of every instruction at the functional unit, and communication by ONNET, a low power clocking network is required. Since power increases substantially when the clock improves, power efficient clocking design scheme is evolved. The architecture is partitioned so that clock is delivered without any signal degradation, concentrating on optimal buffer placement over the entire clock distribution network. A unified clock synchronous cum asynchronous scheme drives the hierarchical clock that runs different levels of node architecture.

CONTROL STRUCTURE

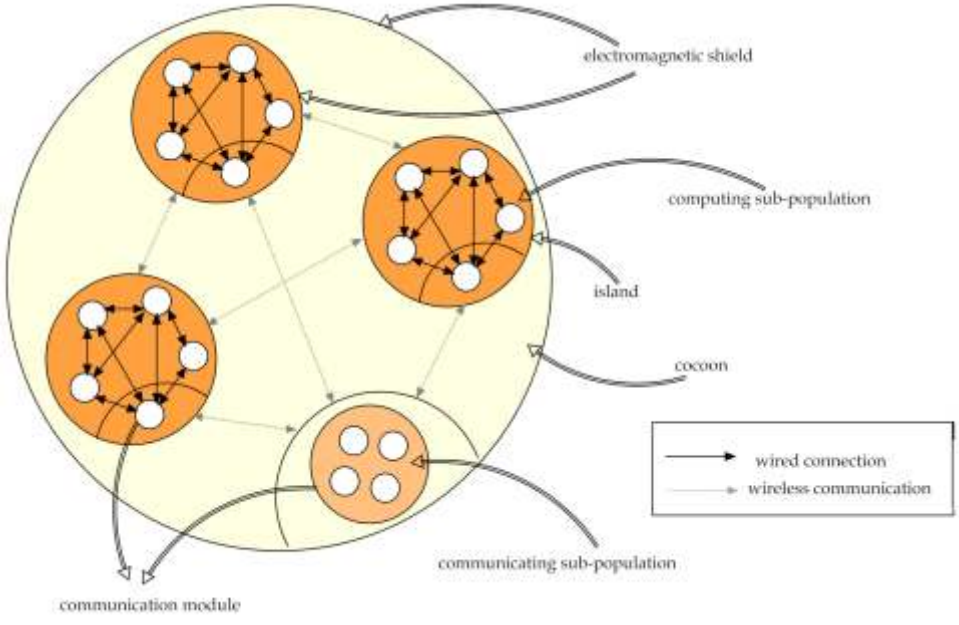
Distributed Control design is an area of paramount importance. Due to the presence of myriad number of Functional Units , a Multiple level Control design needs to be formulated. The presence of Memory-in-Logic Cells , calls for an approach for integrating the Control along the data path. Another major feature of the design adopted is the underlying principle behind the design, Design for Testability. An iterative array based testing scheme for Controller-Datapaths reduces the overall time complexity and large volumes of test data.

BRAIN MODELING AND SUPERCOMPUTING - A SYMBIOTIC RELATIONSHIP

Supercomputing is required for biologically accurate brain modeling and simulation, which in turn, constantly provides fresh challenges for high performance architecture development. Whole brain interconnectivity prediction requires powerful and specialized architectures. The MMINi-DASS and MIP project thus work in close coordination and reinforce each other.

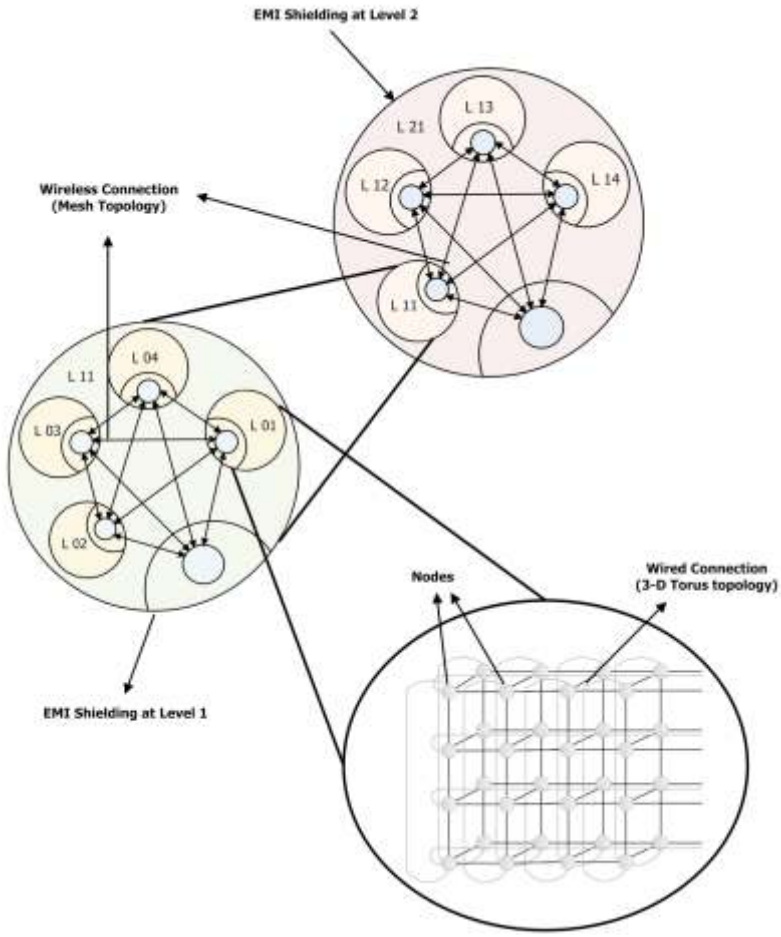
FUSION CLUSTER

While increase in power of individual nodes has been rapid, wired communication networks still have basic drawbacks. Wireless communication can be used to overcome the disadvantages of high power consumption and lack of scalability and reconfigurability of wired networks. To work around problems of insufficient bandwidth and excessive latency, a hybrid network called "Fusion Cluster" is developed consisting of "islands" and wireless links across them. Establishing a balance between number of nodes under wired and wireless communication would lead to a cluster which is both scalable and has better performance to power ratio. The construction of the cluster is achieved by creating an island or Basic Block Cluster (BBC) using a group of nodes under WIred Network Connection (WINC) which follows certain topology and a set of Tx and Rx communication antennas. These islands communicate with other islands through WIreLess network Connection (WILC). Such a group of island are EMI shielded leading to the formation of a cocoon, which helps in reuse of the same frequencies again thus catering to the high bandwidth requirement.



ARCHITECTURE OF FUSION CLUSTER

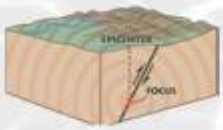
Each cocoon is EMI shielded such that antennas which communicate across cocoons are not covered whereas those antennas within a cocoon are covered such that communication within do not suffer interference from outside. The 3D torus topology is used for the WINC architecture and many-to-many broadcast connectivity with WILC architecture. 3D Torus is employed due to its intricate structure which brings down its diameter and at the same time also offers highest number of connectivity between nodes. The interconnection of WILC and WINC nodes vividly describes the attributes of varied networks and need for their disciplined application at specific layers. The performance of the cluster depends on effectively harnessing the nodes capability and catering to its requirements.



FUSION CLUSTER MODEL

SMAPP

Primary Host Plane



SMAPP

SMAPP

Secondary Host Plane

MIP Node Plane

WAran Research FoundaTion

46-B Mahadevan Street, West Mambalam,
Chennai - 600033
Ph:91 044 24899766
E-Mail : waran@warftindia.org