ET SoC-1 Preliminary Datasheet

Revision 1.00

TABLE OF CONTENTS

TABLE OF CONTENTS

1 ET-SoC-1 Product Overview

The Esperanto ET-SoC-1 is a flexible, fast, and efficient inferencing engine for artificial intelligence machine learning (ML) applications. The ET-SoC-1 is composed of 1,093 general-purpose RISC-V microprocessor cores connected through a mesh-based network and to a set of industry-standard interfaces

The chip contains 1,088 in-order ET-Minion cores for ML computes, four out-of-order ET-Maxion cores that may be used for management tasks, and one simplified ET-Minion core for device management and security functions.

Esperanto's ET-Minion and ET-Maxion are full 64-bit cores based on the RISC-V architecture. Each ET-Minion core also includes a wide, fast vector unit that can perform vector and tensor (matrix-based) operations.

The device contains 140 MB of on-die SRAM distributed across the chip and configurable as cache or scratchpad RAM. Each 1 MB block of this SRAM can be configured to operate as a local L2 cache, part of a chip-wide memory-side L3 cache, or as globally accessible scratchpad memory.

A PCI Express Gen4 interface with eight lanes can implement a x4 or x8 interface to a host system, a x4 or x8 interface to a peripheral device, or a x4 host interface plus a x4 peripheral interface, delivering peak throughput up to 128 Gbps in both directions simultaneously.

The Esperanto SoC is composed of the following elements:

- 1,088 ET-Minion 64-bit RISC-V dual-threaded scalar in-order CPU cores with custom vector/tensor units
- Four ET-Maxion 64-bit RISC-V single-threaded superscalar out-of-order CPU cores
- One ET-Minion-based Service Processor
- A mesh-based network on chip (NoC) operating as the top-level interconnect
- Sixteen 16-bit LPDDR4X controllers operating at up to 4,266 megatransfers/sec (133 Gbytes/s)
- An 8-lane PCI Express 4.0 interface operating at speeds up to 25.78 Gbits/sec per lane that supports Root Complex, Endpoint, and dual-mode operation
- A bootable eMMC flash memory interface
- A USB2 OTG interface that can act as a host or device
- Several serial interfaces: SMBus/I^{2}C (2), I3C (1), SPI (1), UART (2)
- Two debug interfaces: a second device-only USB 2.0 interface and a dedicated JTAG port

2 Theory of Operation

The Esperanto ET-SoC-1 contains a large number of CPU cores, on-chip memories, off-chip memory interfaces, and various on-chip peripherals and off-chip peripheral interfaces. These elements are arranged in a multi-level hierarchy designed to deliver very high levels of sustained performance on regular, parallelizable workloads, particularly those associated with machine learning algorithms.

To understand the ET-SoC-1, it's easiest to start with the chip's fundamental element of computation—a single ET-Minion core—then build up to the full chip.

2.1 ET-SoC-1 Hierarchy

The ET-Minion core is the base computing component of the ET-SoC-1 device. Each ET-Minion core contains an eight-lane vector unit and 4 KB of dedicated data cache.

A *Neighborhood* consists of eight ET-Minion cores and 32 KB of shared instruction cache.A set of four Neighborhoods is then grouped into a Shire. Therefore, each shire has 32 ET-Minion cores. Each Shire also contains 4 MB of shared L2 cache.

A *Minion Shire* consists of four Neighborhoods, a four-bank 4 MB shared L2/L3 cache, a *mesh stop* interface to the mesh-based network on chip (NoC), and a crossbar switch that interconnects these elements of the Shire.

The ET-SoC-1 contains 34 Minion Shires for a total of 1,088 ET-Minion cores. Typically, 32 Minion Shires operate in parallel as the *Compute Array* with one other Minion Shire operating as a *Management Shire*, but this is just a matter of software convention; all Minion Shires are the same.

The *PCI Shire* contains two independent PCI Express controllers and an 8-lane PCIe physical interface. The *I/O Shire* contains a *Maxion Neighborhood* consisting of four ET-Maxion cores, another 4 MB shared L2/L3 cache, a *Service Processor* with its own ROM and 1 MB scratchpad SRAM, a hardware *Root of Trust*, and a variety of other industry-standard interfaces such as USB, I2C, SPI, and UARTs.

Figure 2-1 shows a hierarchical block diagram of the ET-SoC-1 processor. The following subsections provide more detail on each block in the diagram.

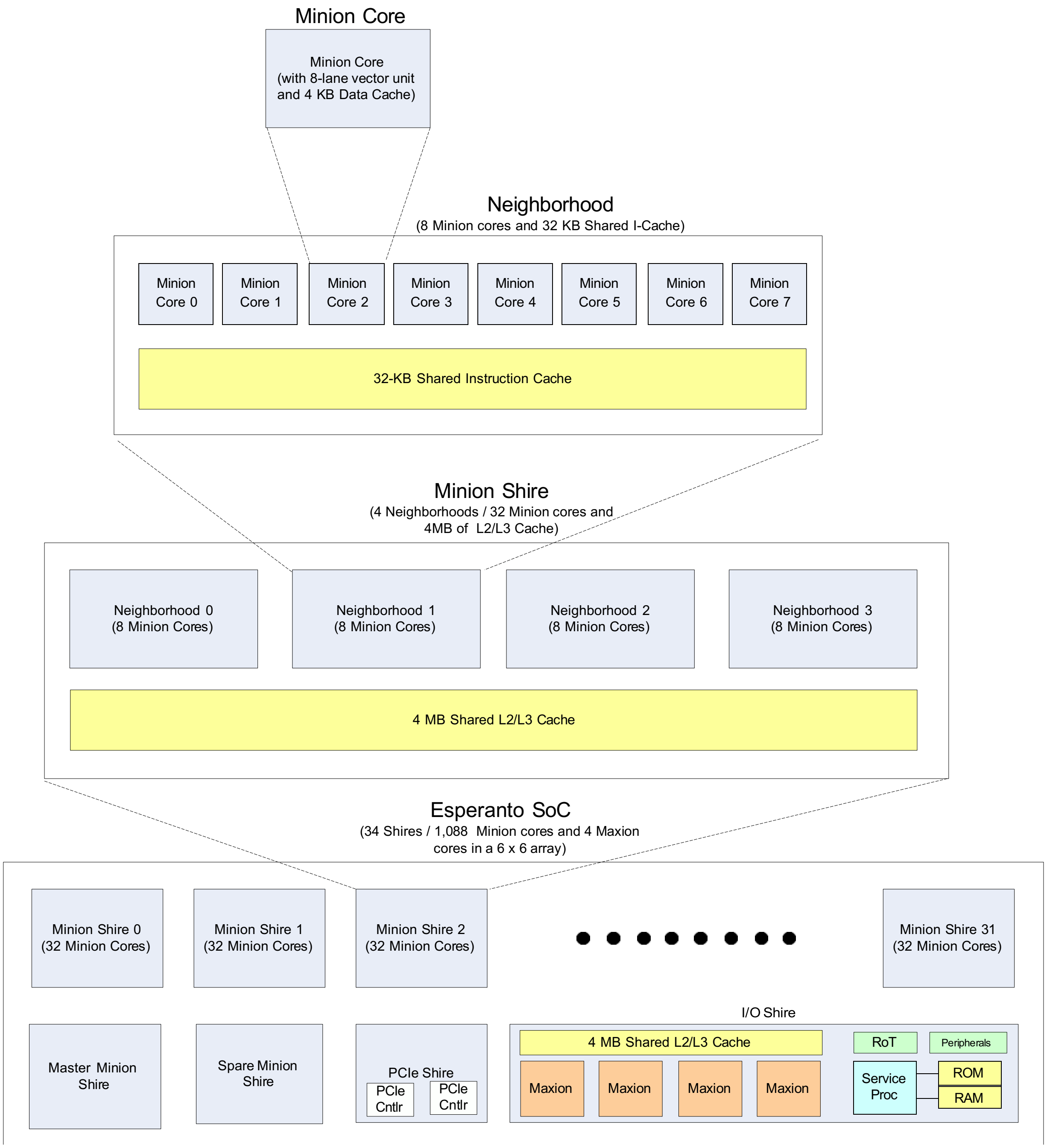
Minion Core
Minion Core (with 8-lane vector unit and 4 KB Data Cache)
Neighborhood
(8 Minion cores and 32 KB Shared I-Cache)
Minion Core 0
Minion Core 1
Minion Core 2
Minion Core 3
Minion Core 4
Minion Core 5
Minion Core 6
Minion Core 7
32-KB Shared Instruction Cache
Minion Shire
(4 Neighborhoods / 32 Minion cores and 4MB of L2/L3 Cache)
Neighborhood 0 (8 Minion Cores)
Neighborhood 1 (8 Minion Cores)
Neighborhood 2 (8 Minion Cores)
Neighborhood 3 (8 Minion Cores)
4 MB Shared L2/L3 Cache
Esperanto SoC
(34 Shires / 1,088 Minion cores and 4 Maxion cores in a 6 x 6 array)
Minion Shire 0 (32 Minion Cores)
Minion Shire 1 (32 Minion Cores)
Minion Shire 2 (32 Minion Cores)
Minion Shire 31 (32 Minion Cores)
I/O Shire
Master Minion Shire
Spare Minion Shire
PCIe Shire
PCIe Cntlr
PCIe Cntlr
4 MB Shared L2/L3 Cache
RoT
Peripherals
Maxion
Maxion
Maxion
Maxion
Service Proc
ROM
RAM

Figure 2-1 Esperanto SoC Hierarchy

2.1.1 The ET-Minion Core

The ET-Minion is a dual-threaded, in-order, single-issue, RV64IMFC RISC-V core equipped with a proprietary 8-lane SIMD vector/tensor execution unit. Each ET-Minion core has its own 4 KB private L1 data cache (D-cache) that can also be configured as a scratchpad RAM.

The vector/tensor unit implements proprietary Esperanto SIMD extensions that operate on 8-bit integer, 32-bit integer, 16-bit floating point, and 32-bit floating point values. **Figure 2-2** shows a block diagram of an Minion core.

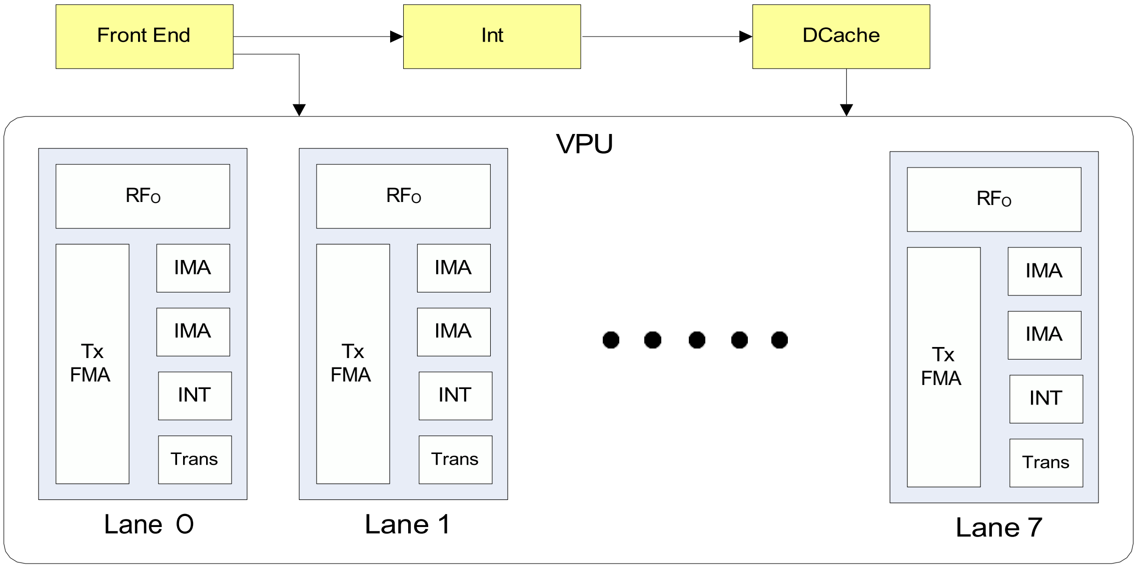

Figure 2-2 The ET-Minion Core

As shown in **Figure 2-2**, the ET-Minion is composed of a front-end (FE) unit for fetching instructions from an external, shared, Instruction cache (I-cache). Instructions are then fed to the Decoder unit (part of the FE) which transfers them either to the integer/D-cache pipeline or to the vector unit.

2.1.1.1 ET-Minion Core Features

The ET-Minion core implements the RISC-V RV64IMFC specification, supporting the basic integer instruction set as well as the integer multiply/divide (M), single-precision floating point (F), and compressed instructions (C) extensions. It also implements the following features:

- Esperanto specific instruction extensions, including:
  - Vector
  - Atomic
  - Messages
  - Tensor
  - Cache management for L2, L3, memory, and instruction cache
  - Fast sync (FLB, FCC)
- Two hardware threads (harts) per ET-Minion core.
- A 4 KB L1 data cache that can be configured as a scratchpad memory.

2.1.1.2 Vector Processing Unit

As shown in **Figure 2-2**, the Vector Processor Unit (VPU) is composed of eight identical lanes. The eight lanes operate completely synchronously, so to the programmer, the VPU appears as a single entity that performs 8 operations per cycle. Each VPU Lane contains the following elements:

- A register file (RF) containing thirty-two 256-bit wide registers per thread. The bottom 32 entries are used by thread 0 while upper 32 are used for thread 1.Therefore, each lane has 32 registers of 32 bits per register, so one register has 32 bits x 8 lanes = 256 bits.
- A Floating Point Multiply Add unit (TxFMA) capable of performing one 32-bit or two 16-bit floating point multiply-add operations
- A pair of Integer Multiply Add (IMA) units each capable of performing four 8-bit integer multiply-accumulate operations
- An Integer unit (INT) capable of performing 32-bit integer and logical operations
- A Transcendental unit capable of performing 32-bit floating point transcendental instructions (exp2, log2, reciprocal)
- A mask module holding eight 8-bit wide mask registers used for logical operations.
- Finite state machines that generate long sequences of vector operations to implement multi-cycle tensor operations, such as tensor fused multiply-add, tensor reduce, tensor quantify, and tensor store for machine learning applications.

2.1.1.3 Front-End

The Front-end (FE) block provides for communication between the pipeline and the instruction cache to which it is attached. It keeps the current program counter, loads new instructions into the I-cache, and propagates I-cache exceptions to the pipeline.

To hide latency between the FE and the memory, the FE uses a double buffer scheme. This allows the FE to issue a new request to the I-cache while the core is consuming the previously-received instructions. Each buffer is capable of storing half a cache line of instructions (256 bits).

The front-end also allows for injecting instructions through the debug hardware to inspect and modify the ET-Minion state. The program buffer uses half of one entry of the same double buffer (128 bits) where the I-cache responses are stored.

Decoder

For timing purposes, the VPU Controller is located inside the Front-End block and receives the decoded instructions from the core's front-end together with additional control signals from the integer pipeline. the Decoder is located inside the Front-end and the decoding happens one cycle before the VPU receives an instruction from the core

This instruction is then interpreted by the Decoder unit. Once the Decoder unit determines the parameters of the instruction, the instruction is forwarded to the corresponding functional unit for execution.

2.1.1.4 VPU Control Block

The VPU Control block is located inside the VPU-top block and manages all interactions between the ET-Minion core and the data cache and maintains important state information regarding the status of instructions in flight.

When the core issues a valid instruction to the VPU, the control starts executing its pipeline to sequence the control of the VPU lane units and also provide in-flight instruction state information to the core and the data cache.

The VPU Control block consists of the following elements:

- Mask
- Machine Learning
- Transcendental

- Load/Store
- Scatter/Gather

Each of these block is described in the following subsections.

Mask

The Mask extension introduces (8) mask registers, each 8 bits in size. Mask register m0 is special and is intended to affect all packed single (PS) and packed integer (PI) instructions as described in those extensions. A set of simple logical instructions operating on pairs of mask registers is provided, as well as a method to read/write all mask registers at once for context switching needs.

Machine Learning

The Machine Learning (ML) block contains a number of finite state machines that are used to perform selected operations, including:

- TensorFMA16A32. This state machine multiplies two FP16 matrices (A and B), adds the resulting FP32 matrix to an FP32 matrix C0, and writes the result into an FP32 matrix C.
- *TensorIMA8A32*. This state machine directs the TIMA unit to multiply two INT8 matrices (A and B), optionally add the resulting INT32 matrix to an INT32 matrix C0, and write the result into an INT32 matrix C.
- *TensorFMA32*. This state machine directs the TXFMA unit to multiply two FP32 matrices (A and B), add the resulting FP32 matrix to an FP32 matrix C0, and write the result into an FP32 matrix C.

In each of the operations listed above, the matrices A, B, C0, and C are stored as follows:

- The A matrix is stored in row-major-order, in consecutive cache lines in the scratchpad.
- The B matrix is stored in column-major-order, either in consecutive cache lines.
- The C0 matrix is stored in N floating-point registers (f_0 through f_{N-1}).
- The resulting C matrix is stored in N floating-point registers (f_0 through f_{N-1}).

Transcendental Support

This unit performs 32-bit floating point transcendental scalar operations, including exp2, log2, and reciprocal operations.

Load/Store Support

Load/Store handling is performed inside the VPU Control block. The ET-SoC-1 device implements load/store support by using Esperanto-specific instruction extensions.

Scatter/Gather Support

Scatter/Gather handling is performed inside the VPU Control block. The ET-SoC-1 device implements scatter/gather support using Esperanto-specific instruction-set extensions.

2.1.1.5 Data Cache

Each ET-Minion core contains a 4 KB data cache to support the operations being performed by the VPU. The data cache can be configured in one of two ways:

- 16-set, 4-way structure with 64 bytes per line.
- 4-set, 4-way structure with 64 bytes per line and 3K of the cache configured as a scratchpad.

The physical data cache memory is organized as 4 LRAM (Latch-RAM) blocks, with 128 rows and 64 bits per row. The data cache control module accepts 4-dimensional read and write requests, one per memory block.

Having the possibility to access multiple rows at a time is useful in case of a misaligned accesses. Reading the data cache blocks is always "continuous". The 4 blocks of 64-bits each would allow more "complex" accesses, but reads or writes only happen in continuous memory locations to deliver the data in the requested range (base address + data size).

For instance, if an access overflows from half the cache line stored in each LRAM row, the data cache will continue reading data from the row that stores the second half of the cache line. Similarly, if a misaligned 64-bit access overflows from one memory block, the data cache will read from the continuous block.

Consider the following examples:

- A 64-bit read (8 bytes) at byte offset 28. The first 4 bytes are read from block 3, row K, and the next 4 bytes will be read from block 0, row K+1. The other two blocks won't be accessed in this case.
- A 256-bit read (32 bytes) at byte offset 16. The first 16 bytes will be read from blocks 2 and 3, row N, and the next 16 bytes will be read from blocks 0 and 1, row N+1

Figure 2-3 shows the organization of the data cache in the default 16-set/4-way configuration.

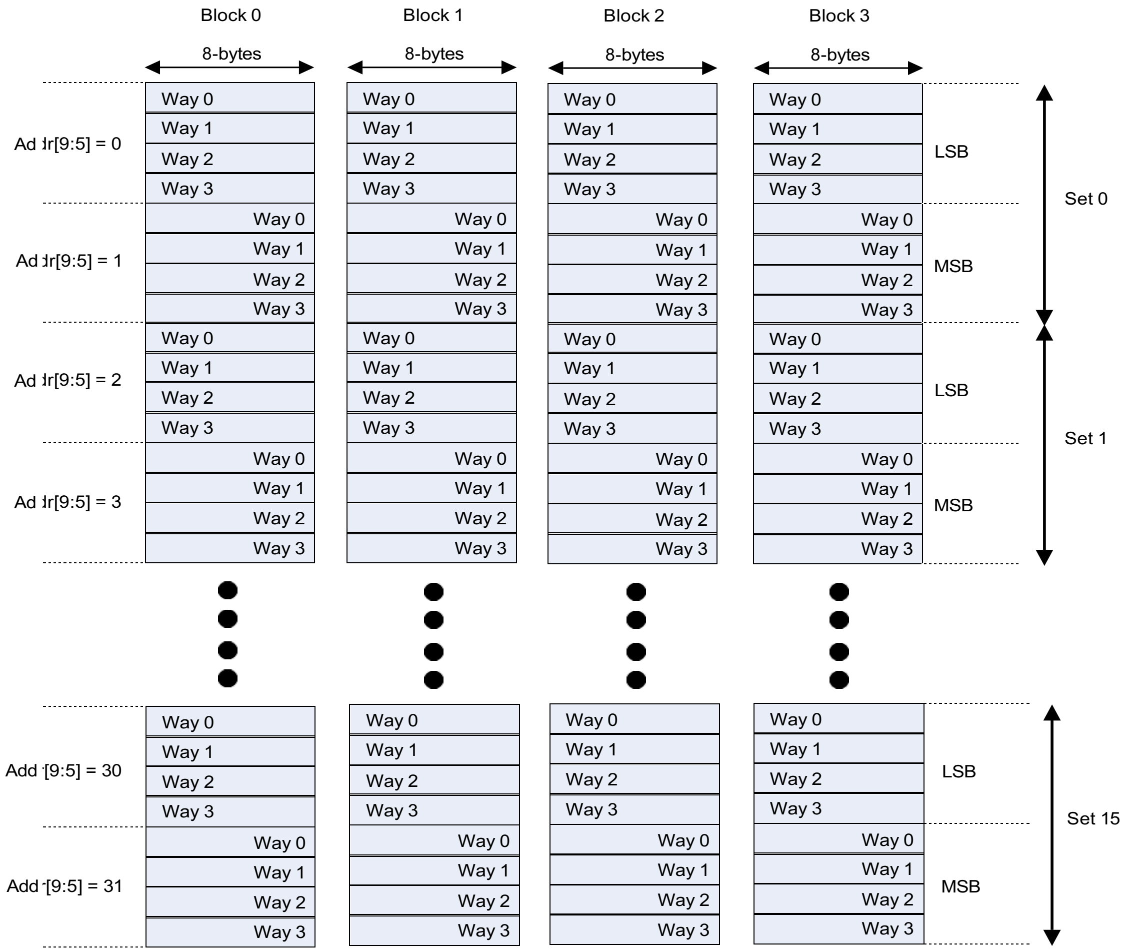

Figure 2-3 ET-Minion Data Cache Layout — Default Configuration

Selecting the Data Cache Configuration

The data cache can be configured as 4-set/4-way with 12 sets reserved for scratchpad, or as 16-set/4-way. The default configuration is 16-set/4-way.

The 4-set/4-way scratchpad mode is implemented by considering two more bits in the tag for each stored cache line, and by forcing the two MSB bits of the set to 2'b11. This way, only sets 15:12 from the default set values are used. In the default 16-set/4-way mode, these two extra bits of the tag are also stored and used. This means that for tag matching there is a single implementation.

The process used to change from 4-set/4-way mode with 12 sets reserved for scratchpad, to the 16-set/4-way mode, involves changing bit 0 in Scratchpad Control CSR register. Each time the bit is modified the process to change the data cache mode starts. During this process no new requests are allowed to enter the data cache and the core cannot initiate any new requests to other data cache modules (via CSR) until the operation is completed.

Data Cache to Shire Cache Interface

The Data Cache interfaces directly to the Shire Cache via an ET-Link interface. At the Neighborhood level the ET-Link requests from the different blocks, including both instruction and data caches, are arbitrated before reaching the Neighborhood to Shire Cache interface.

Data Cache to VPU Interface

The VPU interface to the data cache performs the following functions:

- Provides direct access to the scratchpad RAM (SCP)
- Requests information during Tensor Reduce and Tensor Store operations
- Transfers Tensor Load data from memory to the VPU buffer

2.1.2 The ET-Minion Neighborhood

Eight ET-Minions are grouped together in a structure called a Neighborhood. The eight ET-Minions in a neighborhood share some infrastructure within the block: the L0 microcache, the page table walkers[1], and the input/output buses. As shown in **Figure 2-4** below, four ET-Minions share a L0 microcache and a page table walker. The L0 microcaches are serviced from a local 32 KB shared I-cache. There are two L0 microcaches per neighborhood, each serving four ET-Minion cores (eight threads). Each L0 microcache has 16 fully-associative entries (so, on average, there is two entries per thread).

[1] The Page Table Walkers (PTW) are part of the ET-SoC-1 architecture and are included in **Figure 2-4** for completeness, but are not implemented in the first revision of silicon. They will be included in a future revision.

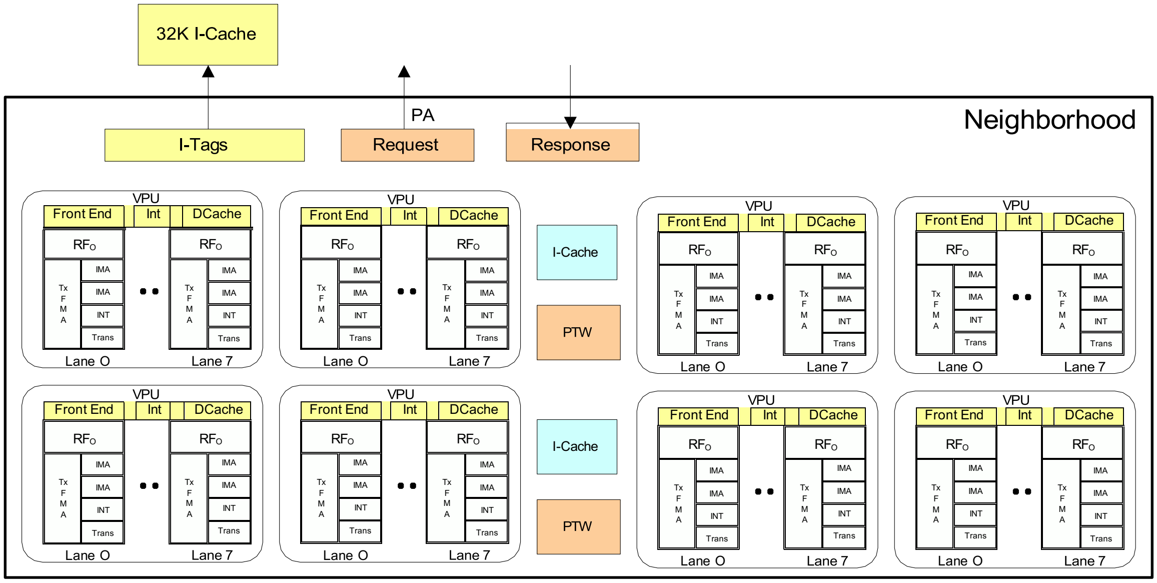

Figure 2-4 Neighborhood with Eight ET-Minion Cores and Shared Instruction Cache

2.1.2.1 ET-Minion Neighborhood Instruction Caches

The eight Minions share a single level-1 (L1) 32KB I-Cache memory. However, each group of 4 ET-Minions also share a an L0 microcache.

- L0 microcache: A lower-level 16-entry fully-associative cache. There are two instances of the L0 microcache. Each microcache is shared among four ET-Minion cores. The access of the cores is arbitrated for in the neighborhood channel.
- L1 I-cache: The main cache. The size of the L1 I-cache is 32KB (128 sets and 4 ways). The data RAMs are located out of the neighborhood and are accessed asynchronously from the L1 Icache pipeline. The two L0 microcaches arbitrate to access the L1 Icache when they miss a fetch request.

These micro I-caches are used to "approach" the memory to the Minions as it is relatively "far" (the memory is in the Shire Channel, outside of the Neighborhood). Each micro L0 microcache has capacity for 16 entries (each entry is a cache line of 512 bits).

The lower-level L0 microcache resides between the ET-Minions and the main I-cache so that the ET-Minion core maintains a low access latency.

The Esperanto platform offers a code prefetching service that software can use to preload critical kernels into the shared I-caches prior to executing them. This service is programmed through the Shire ESRs that reach the I-cache through the I-cache prefetch interface.

The I-cache can notify certain types of errors, like SBE or DBE in a cache line.

2.1.2.2 Page Table Walker

There are two page table walkers (PTW) in each Neighborhood. Each PTW is shared by four ET-Minion cores.

In the PTW, the page sizes supported are 512 GB, 1 GB, 2 MB and 4 KB. Thus, up to four page levels may be found. If no page table entry (PTE) is found in the PTW's 8-entry fully associative cache, a read request is sent to the L2 Shire cache.

The fill request returns a full 512-bit cache line. The received PTE is stored into the PTW's cache if it is a non-leaf PTE. If a valid leaf PTE is found, and permissions and privilege mode are OK, then the physical page number is returned to the requester.

2.1.2.3 Neighborhood Performance Monitoring

The Performance Monitoring Unit (PMU) is a block shared among the eight Neighborhood Minions. It includes a total of twelve 64-bit generic counters that can be used to count any of the events that the Minions and the Neighborhood can generate. Eight counters are reserved for Minion events, while the remaining four are reserved for Neighborhood events.

2.1.3 The ET-Minion Shire

A ET-Minion shire contains a total of 32 ET-Minions, as shown below.

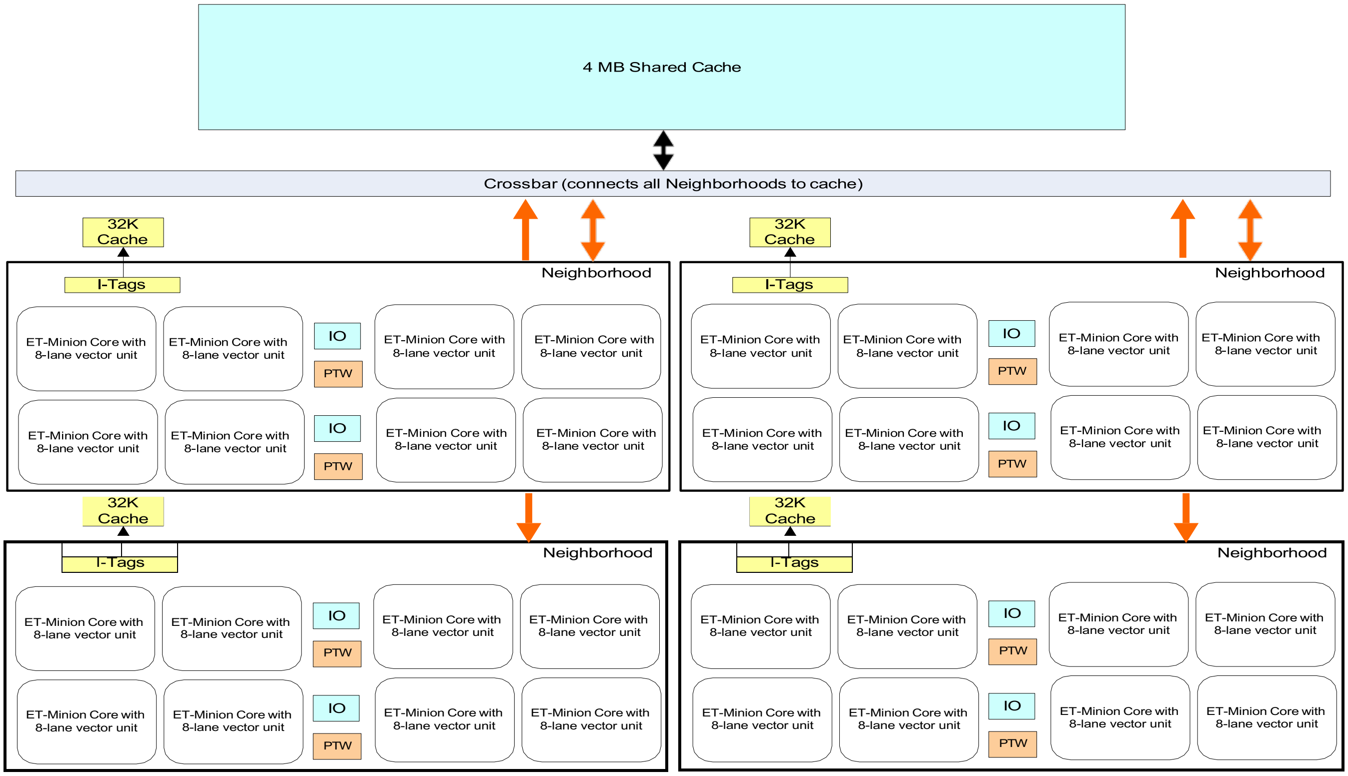
4 MB Shared Cache
Crossbar (connects all Neighborhoods to cache)
32K Cache
I-Tags
Neighborhood
ET-Minion Core with 8-lane vector unit
IO
PTW

Figure 2-5 ET-Minion Shire with Four Neighborhoods and Thirty-Two ET-Minion Cores

L2 cache misses (and uncachable accesses) leave the L2 cache and go to the NoC (Network on Chip) interface. The NoC interface allows read/write requests to get onto the top-level NoC (which is independent from the crossbars internal to the Shire) and travel to other Shires or to the memory controllers.

2.1.3.1 ET-Minion Shire Cache

The 4MB Shire Cache in the ET-Minion Shire is configured as four 1MB banks each with four sub-banks. The Shire Cache can be partitioned and each partition configured for use as L2 cache, L3 cache, or scratch-pad memory. The size of each partition is determined by values programmed into the ESRs.

The L2 partition contains cache memory that is shared by the ET-Minion CPUs within the Shire, but is not shared with other Shires, thus creating an independent L2 cache per Shire.

The L3 partition enables each Shire to implement a slice of a chip-level distributed L3 cache.

The scratch-pad partition contains memory-mapped RAM that configured for access by other agents on the chip, or through the PCI Express interface, by other agents in the system, including the system host processor or other ET-SoC-1 devices.

2.2 ET-SoC-1 Block Diagram

Figure 2-6 shows a block diagram of the ET-SoC-1 device.

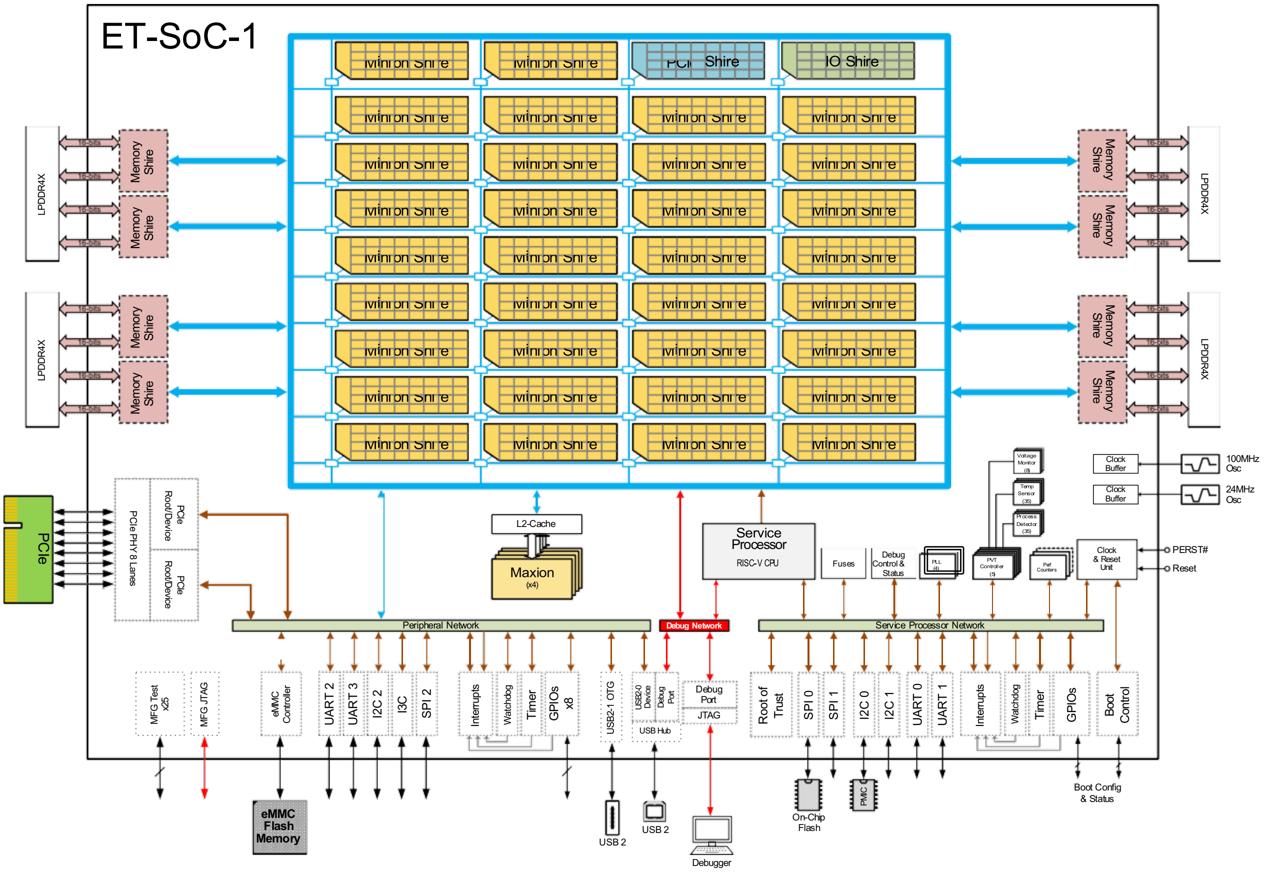

Figure 2-6 ET-SoC-1 Block Diagram

As shown in Figure 2-6, the ET-SoC-1 device contains the following main blocks:

- 34 ET-Minion Shires, with each Minion Shire consisting of 32 ET-Minion cores.
- Eight Memory Shires for communication with DDR memory.
- Peripheral network used to communicate with external devices such as PCIe.
- An ET-Minion-based Service Processor used to control the ET-SoC-1.
- Debug network used to perform debug operations through the JTAG and USB2 interfaces.
- Four ET-Maxion cores

Each of these main blocks is described in the following subsections.

2.2.1 ET-Minion Shire Array

The ET-SoC-1 contains an array of 34 ET-Minion Shires, yielding a total of 1,088 ET-Minion cores. The array is divided into ET-Minion Cores, Neighborhoods, and Shires as described in Section 2.1 “ET-SoC-1 Hierarchy”.

2.2.2 Memory Shires

The ET-SoC-1 memory subsystem contains four banks of LPDDR4X memory. Each bank is controlled by two Memory Shires for a total of eight Shires. For more information, refer to **Section 7 "Memory Shires"**.

2.2.3 Peripheral Network

The Peripheral Network in the ET-SoC-1 device contains the following interfaces for communicating with external devices.

- PCIe. This interface is controlled by two PCIe controllers as described in **Section 2.3 "PCIe Interface"**.
- UART
- I2C
- I3C
- USB-Device
- USB-OTG
- GPIO

2.2.4 Service Processor Network

The Service Processor network in the ET-SoC-1 device contains the following interfaces for communicating with external devices.

- SPI
- I2C
- UART
- GPIO
- Boot Control

2.2.5 Debug Network

The Debug network in the ET-SoC-1 device contains the following interfaces for communicating with external devices.

- JTAG
- USB2.0 device port

2.2.6 Clocking

The Clocking block in the ET-SoC-1 device contains the following clock sources.

- 24 MHz oscillator
- 100 MHz oscillator

2.2.7 Maxion Cores

The ET-SoC-1 device contains four Maxion RISC-V cores for general purpose control.

2.3 PCIe Interface

The PCIe interface contains an 8-lane PCIe Gen 4 physical layer and two dual mode controllers. The PCIe interface contains a maximum of 8 lanes which can be configured for endpoint, root complex, or a combination of modes. The three operating modes are outlined in the following subsections.

Figure 2-7 shows the Esperanto PCIe interface modes of operation. The PCIe interface will generally be factory-configured for Endpoint operation. If Root Complex operation is desired (for example, for self-hosted operation), the customer should contact Esperanto for details.

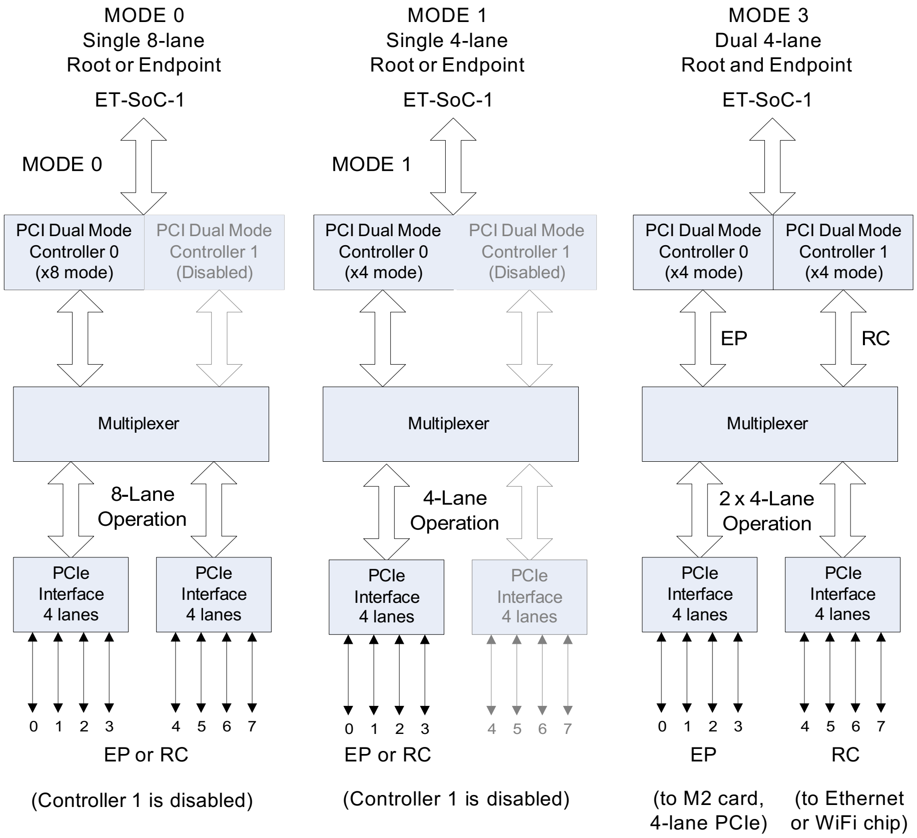

Figure 2-7 PCIe Modes of Operation

2.3.1 Mode 0

In this mode:

- PCIe dual mode controller 0 is enabled and configured as an 8-lane Endpoint or Root Complex.
- PCIe dual mode controller 1 is disabled
- Each interface auto-negotiates to match the PCIe version of the remote device (generation 4, 3, or 2) and the number of lanes connected (8, 4, 2, or 1).

2.3.2 Mode 1

In this mode:

- PCIe dual mode controller 0 is enabled and configured as a 4-lane Endpoint or Root Complex.

- PCIe dual mode controller 1 is disabled
- Each interface auto-negotiates to match the PCIe version of the remote device (generation 4, 3, or 2) and the number of lanes connected (4, 2, or 1).

2.3.3 Mode 3

In this mode:

- PCIe dual mode controller 0 is enabled and connected to PCIe interface lanes 0–3, configured as an Endpoint
- PCIe dual mode controller 1 is enabled and connected to PCIe interface lanes 4–7, configured as a Root Complex
- Each interface auto-negotiates to match the PCIe version of the remote device (4, 3, or 2) and the number of lanes connected (4, 2, or 1).

Peripherals which utilize only a single lane interface can either have the other three lanes disabled by software or the PHY will auto-detect the available lanes during negotiation.

2.4 Service Processor

The service processor (SP) consists of a single ET-Minion core located in the I/O Shire that is used to manage communication with the I/O peripherals. This core is similar to other Minion cores and has access to all resources on the SoC that any other agent can access. It also has exclusive access to some resources that no other agent can access. However, it does not include any of the other Neighborhood or Shire shared blocks or cache resources as in the ET-Minion Shire and has only one hardware thread.

2.5 I/O Subsystem

The I/O subsystem contains the aforementioned Service Processor (SP) and its ancillary components responsible for booting the ET-SoC-1. It also contains I/O devices that are accessible from ET-Maxions and the Minion Shire selected as a Master Minion Shire.

2.5.1 Service Processor I/O

The Service Processor I/O block contains peripherals that only the Service Processor has access to. The Service Processor also has access to all the resources on the ET-SoC-1. It plays a significant role in booting the SoC and provides runtime services to the entire SoC.

2.5.2 Maxion Cores

The ET-Maxion is a high-performance out-of-order RISC-V core. The ET-SoC-1 contains four identical ET-Maxion cores. These cores are part of the I/O Shire. The I/O Shire also contains the system bus and coherency hub that ties the cores together.

2.6 ET-SoC-1 Mailbox Structure

The ET-SoC-1 incorporates six instances of the secure mailbox, providing a secure communication channel between four different agents: the Service Processor (SP), Master Minions, Maxions, and the PCIe host. The mailboxes are shown in yellow and are connected to each agent as shown in **Figure 2-8**.

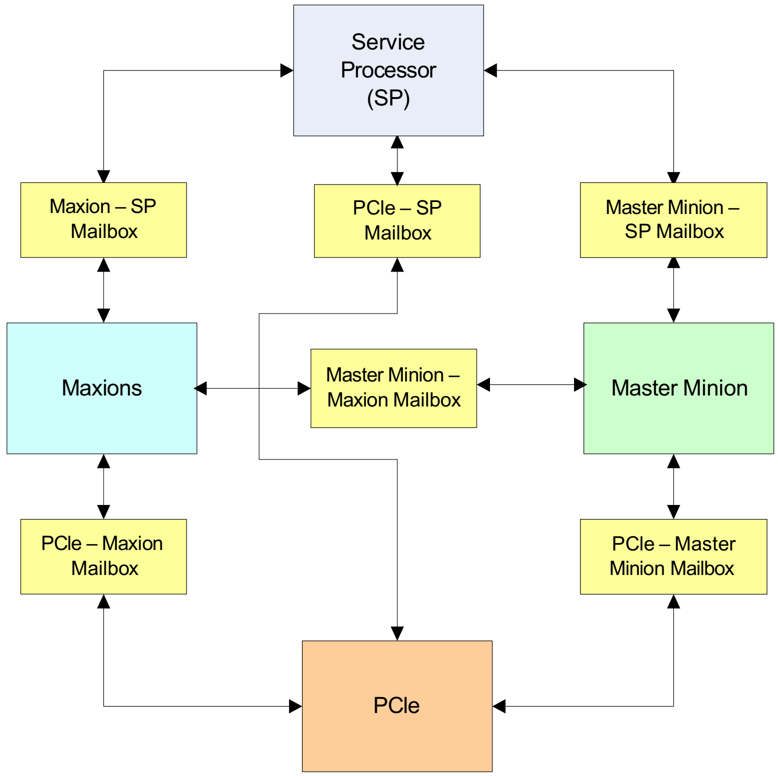

Figure 2-8 Mailbox Connections

3 ET-SoC-1 Memory Map Overview

The Esperanto ET-SoC-1 uses 40 bits of physical address which define a 1 TByte addressing space. Each of these regions is further subdivided into smaller regions that are explained below. In addition, each of these smaller regions may have different access permissions. As defined in the RISC-V specification, if a core or hardware thread within a core (known as a hart) attempts to access a region of memory where it does not have permissions based on the type of access performed, or attempts to access a reserved address, then an *Access Fault* exception is generated.

- **I/O region**: The I/O region contains the devices listed below. Note that firmware can disable ET-Minion access to this region on a per-Shire basis. Therefore, if access is disabled to a given Shire, it is disabled for all 32 ET-Minion cores in that Shire.
  - ET-Maxion cores
  - Globally accessible peripheral interfaces
  - Mailbox buffers
- **Service Processor (SP) region**: This area is private to the Service Processor. No other agent in the platform can access it. This area holds interfaces for multiple peripherals.
  - Peripheral interfaces local to the SP
  - PLLs
  - PVT sensors
  - Mappings to reach the configuration registers for LPDDR4X, PCIe and Debug components.
- **Scratchpad region:** This regions maps the union of all Shire Cache scratchpad areas. Any Minion, Maxion or the Service Processor can read and write to any of the scratchpads in the system.
- **Esperanto System Registers (ESR) region**: This region maps all the Esperanto-defined platform system registers, which control a variety of different functions.
- **PCIe region:** The PCIe region includes the following areas:
  - Window into the Root Complex: This region, if enabled through the PCIe configuration space, will allow any agent in the system, with the appropriate access permissions, to read and write to/from the Root Complex memory space.
  - PCIe Configuration Space: also located here in an aligned 256MB region.
  - PCIe Endpoint space: When ET-SoC-1 acts as PCIe Root Complex, this region allows access to the devices connected to the ET-SoC-1 root.
- **DDR region**: This region is used to map all the available DDR memory in the platform. It is further subdivided into special usage sub-regions as described later in this document.

Each type of agent capable of generating a read/write request may have different permissions to the regions defined above as described in the Physical Memory Attributes and Other Access restrictions section.

For more information on the contents of the ET-SoC-1 memory map, refer to the Memory Map chapter of the *ET-SoC-1 Programmers Reference Manual*.

4 ET-SoC-1 Mesh Network on Chip

The mesh network-on-chip (NoC) connects all the shires in the ET-SoC-1. The NoC is organized as an 8 x 6 grid that includes the following elements:

- 34 ET-Minion Shires
- 1 PCIe Shire
- 1 I/O Shire
- 8 Memory Shires

Because there are only four Memshires on each side of the chip, the corners of the grid are not occupied, and there are 44 total mesh stops.

5 Clocking and Reset

5.1 Clock Management

Figure 5-1 shows the clock management diagram for the ET-SoC-1.

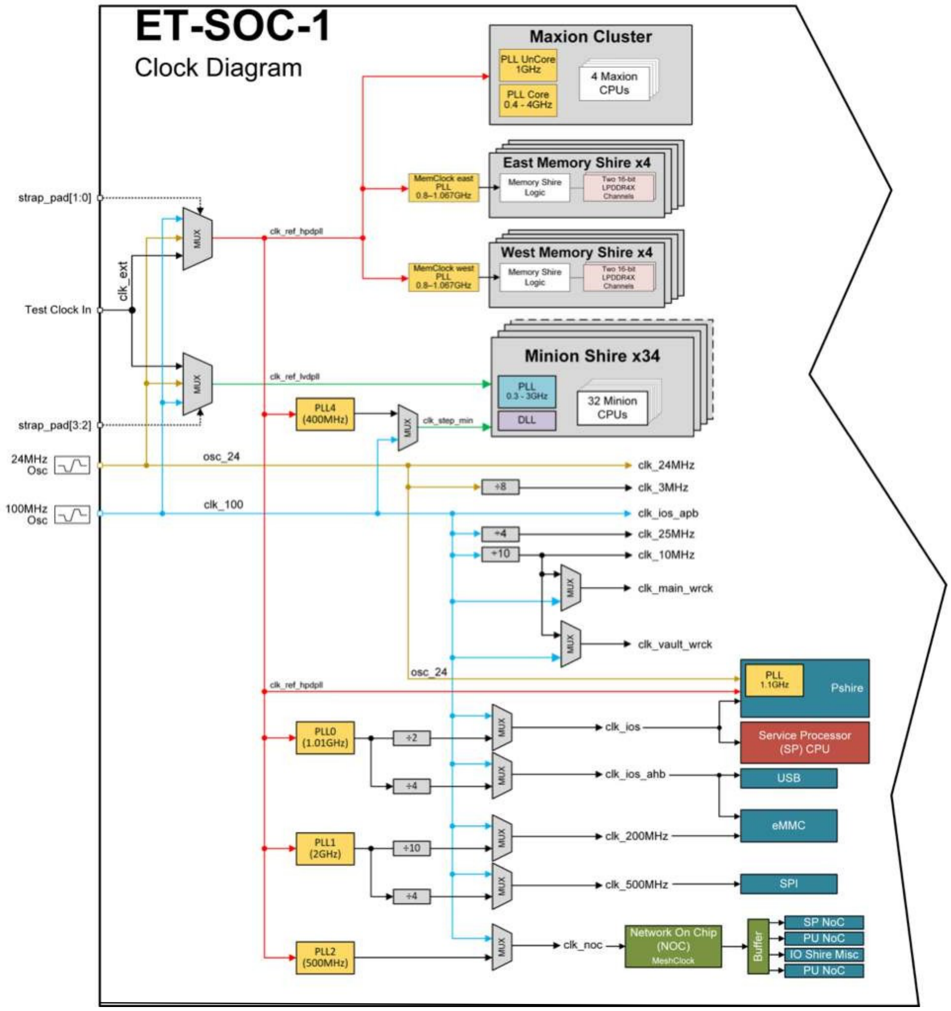

Figure 5-1 I/O Shire Clock Management Diagram

5.1.1 I/O Shire External Clock Sources

There are three external clock sources for the I/O Shire:

1. *clk_100_in*: a 100 MHz external oscillator.
2. *osc_24_in*: a 24 Mhz external clock oscillator.
3. *clk_ext_in*: This clock is for a test clock. Leave unconnected.

5.2 Reset Manager

The reset sequence consists of several stages. Each stage reaches a system state to enable the next stage operation. There are two main stages; a hardware-controlled reset stage, and a software-controlled reset stage.

5.2.1 Hardware-Controlled Reset Stage

After the main SoC system reset input RESET_n is asserted, it must remain in the asserted state (low) for at least 1 mS. Then it can be deasserted. Once the RESET_n pin is deasserted, the finite state machine inside the Reset Manager begins operation and deasserts the various resets for the rest of the ET-SoC-1.

5.2.2 Software-Controlled Reset Stage

The Service Processor (SP)is the first major block to be brought out of reset by the Reset Manager, and this begins the boot process. The SP fetches code from ROM, which starts the software boot sequence. The SP initializes components such as USB, DDR, and PCIe interfaces, etc. The boot sequence is controlled by the program stored in the ROM and flash.

6 ET-SoC-1 Interfaces

6.1 I2C/SMBus

I2C (Inter-Integrated Circuit) is a synchronous, multi-master, multi-slave, packet switched, single-ended, serial computer bus. It can be used for attaching lower-speed peripheral ICs to the ET-SoC-1 device.

SMBus is a subset of the I2C bus developed primarily for power management in PCs. The ET-SoC-1 device supports both I2C and SMBus, requiring only minimal reconfiguration either by command, or output pin use.

Devices, both masters and slaves, can bedesigned to be freely interchangeable between both buses. Both buses feature addressable slaves (although specific address allocations can vary between the two buses). SMBus mode has a maximum operating frequency of 100 KHz. I2C mode supports speedsup to 1 MHz.

6.1.1 I2C Controllers

The ET-SoC-1 contains three independent I2C controllers:

- ***spio_i2c0*** — For use by the Service Processor (SP) to communicate with an external power management controller. This interface will operate as a master.
- ***spio_i2c1***— For general purpose use by the SP only
- ***pu_i2c*** — For general purpose use by the ET-SoC-1, including ET-Maxions and the master ET-Minion

6.1.2 I2C Features

Some features of the I2C controller are as follows:

- Conforms to I2C bus specification version 6.0
- Supports SMBus mode
- Master or slave only operation, configured by the Service Processor (SP) as needed. Note that master/slave operation is not supported in dual mode.
- Speed modes:
  - Standard speed mode (up to 100 Kbps)
  - Fast speed mode (up to 400 Kbps)
  - Fast speed mode plus (up to 1 Mbps)
- 7-bit addressing
- Input spike suppression
- Clock synchronization
- Slave clock stretching
- Bus arbitration
- General call address
- Bus clear operation
- No read device ID feature
- Programmable timing parameters, including: (t_{LOW}), (t_{HIGH}), ($t_{HD;STA}$), ($t_{SU;STA}$), ($t_{HD;DAT}$), ($t_{SU;STO}$), (t_{BUF}), and (t_{SP})
- Programmable FIFO watermarks
- Single interrupt per interface

6.1.3 I2C Pins

The two external SoC pins for the SPI interface are as follows:

- **i2c_scl**
- **i2c_sda**

6.1.4 I2C Address Space

The APB bus has an 8-bit address, each I2C controller is allocated a 4KB memory space in the address map. If locations outside of the 8-bit address space are addressed, accesses wrap around to the start of the address space.

6.2 I3C

The I3C bus interface can be used for connecting sensors to the ET-SoC-1 processor. It incorporates key attributes of the traditional I2C and SPI interfaces to provide a new, unified, high-performing, very low power solution. The I3C bus supports a minimum data rate of 10 Mbps with options for higher performance high data rate modes.

Features include multi-master support, dynamic addressing, command-code compatibility, and a uniform approach for advanced power management features, such as sleep mode. The I3C bus provides synchronous and asynchronous time-stamping to improve the accuracy of applications that use signals from various sensors. It can also batch and transmit data quickly to minimize energy consumption of the host processor.

The ET-SoC-1 contains one I3C controller:

- ***pu_i3c*** — For general purpose use, including ET-Maxion cores and the master Minion Shire.

6.2.1 I3C Features

Some features of the I3C controller include:

- Single data rate mode (SDR): New I3C enhanced version of the I2C protocol supporting private messages, and adding two kinds of standard built-in messages:
  - Broadcast messages, which are sent to all I3C Slaves on the Bus
  - Direct messages, which are addressed to specific Slaves
- Speeds from 11 MHz to 12.9 MHz
- Dynamic address assignment
- I3C Slave generated requests:
  - In-Band Interrupt
  - Hot-Join request
  - Peer-to-Peer request
  - Mastership request
- I3C common command codes
- Legacy I2C device co-existence on the same bus
- Memory for retaining slave devices
- Command queue

6.2.2 I3C Pins

The I3C interface consists of two external SoC pins:

- **i3c_scl**

- **i3c_sda**

6.3 SPI

The ET-SoC-1 processor contains three SPI ports. The Serial Peripheral Interface (SPI) consist of one master device and one or more slave devices. The master device provides the SPI clock and the slave receives the SPI clock from the master.

The SPI has a 4-wire synchronous serial interface. Data communication is enabled with a low active Slave Select (SS) or Chip Select wire (CSB). Data is transmitted with a 3-wire interface consisting of wires for serial data input (MOSI), serial data output (MISO) and serial clock (SCK).

6.3.1 SPI Features

The SPI controller supports:

- Full-duplex operation, simultaneous receive and transmit
- Master SPI mode of operation only, Synopsys controller must be statically configured for one or the other
- Four wire bus -- data RX, data TX, clock and select
- Supports multi-master environment - Identifies an error condition if more than one master detected
- Buffered operation with separate transmit and receive FIFOs - the APB can read from the RX FIFO and write to the TX FIFO
- Multiple word transfer per FIFO location, in which FIFO word width and programmed transfer size can be different, providing one is a multiple of the other
- Programmable master mode clock frequencies
- Serial clock with programmable polarity
- Programmable transmission format
- FIFO levels available via DUT outputs, or through software accessible registers
- FIFO level status can be polled via software or can be interrupt driven
- Programmable interrupt generation
- Up to 16 external peripheral selects
- Scan test interface

6.3.2 SPI Controllers

The ET-SoC-1 will have three independent SPI controllers:

- ***spio_spi0*** — For use by the Service Processor (SP) to communicate with on-chip flash memory. This interface will operate as a master and will only have one slave select pin.
- ***spio_spi1*** — For general purpose use by the SP only
- ***pu_spi*** — For general purpose use by ET-SoC-1, including ET-Maxion and the master ET-Minion cores

6.3.3 SPI Pins

There are six external pins for each of the three SPI interfaces. These pin are defined as follows:

- **spi_sclk_out** — clock out pin for the SPI port.
- **spi_rxd** — Receive data pin for the SPI port.
- **spi_txd** — Transmit data pin for the SPI port.

- **spi_ss_0_n** — Active low slave select 0. Output from master.
- **spi_ss_1_n** — Active low slave select 1. Output from master.
- **spi_ss_in_n** — Active low slave select 0. Input from slaves.

6.3.4 SPI Address Space

The APB bus has an 8-bit address. Each SPI controller is allocated a 4KB memory space in the address map. If locations outside of the 8-bit address space are addressed, the accesses wrap around to the start of the address space. For the exact location of the SPI address space, refer to the *ET-SoC-1 Programmer's Reference Manual*.

6.4 UART

The ET-SoC-1 processor contains four independent UART serial ports that are used by various elements of the device as described below.

6.4.1 UART Features

Each of the four UART controllers has the following features:

- Programmable baud rate generator
- Configurable receive and transmit FIFOs, with byte, two byte or four byte APB access mechanisms
- 6, 7, or 8 data bits
- 1, 1.5, or 2 stop bits
- Odd, even, sticky, or no parity
- Parity, framing, and overflow error detection
- Line break generation and detection
- Loopback mode
- Interrupt generation
- Modem control signals: CTS, RTS, DSR, DTR, RI and DCD which are to be hard wired in RTL to their operational states, only txd and rxd to be available on pins.

6.4.2 UART Ports

The ET-SoC-1 processor contains four independent UART controllers:

- **spio_uart0** — For use by the Service Processor (SP) for console functionality
- **spio_uart1** — For general purpose use by the SP only
- **pu_uart0** — For use by the master ET-Minion core for console functionality
- **pu_uart1** — For use by the ET-Maxion cores for console functionality

6.4.3 UART Pins

Each UART interface contains the following two signals:

- **uart_rx** — UART receive
- **uart_tx** — UART transmit

6.5 GPIO

There are two types of GPIO pins in the ET-SoC-1 processor.

- PU GPIO — Peripheral unit GPIO pins (8)
- SPIO GPIO — Service Processor GPIO pins (16)

6.6 Timers

The ET-SoC-1 device contains the following timers:

- Eight generic PU timers
- Eight generic SPIO timers
- Three RISC-V timers

6.6.1 Generic Timers

The ET-SoC-1 device incorporates eight generic timers used by the Peripheral Unit (PU), and eight generic timers used by the Service Processor I/O (SPIO). The PU timers are used by the Service Processor, Maxion cores, and the Master ET-Minion core. The SPIO timers are used by the Service Processor. These timers are located at the following locations:

- The PU timers are used by the peripheral unit and resides in a 4KB space at physical address 0x00_1200_5000 - 0x00_1200_5FFF
- The SPIO timers are used by the Service Processor I/O unit and resides in a 4KB space at physical address 0x00_5202_5000 - 0x00_5202_5FFF

The timers are accessed through a single AMBA APB interface. The timers count down from a programmed value and generate an interrupt when the count reaches zero.

Each timer has an independent clock input that can be connected to pclk (also known as the system clock or the APB clock), or to an external clock source. The width of the PU and SPIO timers is fixed at 32 bits.

The initial value is loaded into the timer using the appropriate load count register (*TimerNLoadCount*). Two events can cause a timer to load the initial count from its *TimerNLoadCount* register:

1. Timer is enabled after being reset or disabled
2. Timer counts down to 0

All interrupt status registers and end-of-interrupt registers can be accessed at any time.

6.6.2 RISC V Timers

The RISC V (RV) timers are real-time counters for the ET-SoC-1 device. They provide the Service Processor, ET-Minion, and ET-Maxion processors a way to measure time at a fixed frequency that is independent of the frequency that the processors are running at. The ET-SoC-1 provides three RV timers.

- One 64-bit RV timer dedicated to the Service Processor (SP)
- One 64-bit RV timer used by all ET-Minion cores in the system
- One 64-bit RV timer used by the Maxion Neighborhood. Note that this timer is provided as part of the RISC V architecture. Its implementation is different than the timers described in this section. Refer to the RISC V documentation for more information on this timer.

6.6.2.1 RISC V Timer Registers

Each RV timer consists of two registers: *mtime* and *mtimecmp*.

The *mtime* register is a counter that increments every 100 ns (i.e. at a frequency of 10 MHz). If the counter ever overflows, it wraps around to 0 and continues counting. However, because the counter is 64 bits wide, overflow will occur sometime around 64,000 years.

The *mtimecmp* register holds a value with which the *mtime* counter is compared. When the *mtime* value is equal or greater than the *mtimecmp* value, the RV timer raises a timer interrupt. For the Service Processor RV timer, this interrupt is sent directly to the SP. For the PU RV timer, the timer interrupt is sent directly to the ET-Minion shires. The RV timer will continue to assert the interrupt request until software writes the *mtimecmp* register. The interrupt will be held even if the *mtime* counter wraps to a value that is now less than that of the *mtimecmp* register.

Software accesses the *mtime* and *mtimecmp* registers via the AMBA 3 AHB-Lite protocol with a 64 bit data bus.

6.6.2.2 RV Timer Reset Request

The RV timer is reset by the AHB interface reset signal. This signal is handled as an asynchronous active low reset. The RV timer state is reset within one or two cycles of the assertion of reset.

During an RV timer reset operation, the *mtime* register is reset to 0, while the *mtimecmp* register is reset to its maximum value (2^{64} - 1). This in turn gives the boot code plenty of time to set up the RV Timer interrupt enables and handlers before the timer can possibly raise an interrupt.

6.6.2.3 ET-Minion RV Timer Arbitration

All of the ET-Minions in the ET-SoC-1 arbitrate for use of the ET-Minion RV timer.

7 Memory Shires

The ET-SoC-1 incorporates eight Memshires, four on the east side and four on the west side. Each Memshire contains two 16-bit LPDDR4X physical interfaces sharing a single 32-bit controller. Two Memshires communicate with each 64-bit LPDDR4X memory device at a rate of 4,266 Mbps.

7.1 Memshire Features

- Each Memshire supports two 16-bit LPDDR4X channels each at a maximum data rate of 4,266 Mbps.
- Minimum capacity at full bandwidth: 4 gigabytes of LPDDR4X at 2 Gbits per channel
- Maximum 32 GByte using LPDDR4X at 16 Gbits per channel
- LPDDR4X support for global atomic operations
- Debug Support
  - Debug of incoming AXI memory traffic from NoC
  - Tracing and counting of performance signals from memory controllers
  - Support for streaming of debug messages from rest of SOC into and out of an off-chip memory circular buffer.

7.1.1 Network-on-Chip Interface

The NoC connects three ports to each MemShire that control the following:

- Memory traffic
- Global atomic responses
- ESR accesses

The NoC delivers a maximum bandwidth of 32 GByte/s, allowing burst transfer rates between the NoC and the Memshire almost twice as fast as the sustained throughput of the corresponding LPDDR4X interfaces.

7.1.1.1 NoC Memory Traffic

The NoC supplies memory requests and responses through the memory slave AXI port. This port has a 512-bit data path to support the full 64-byte cache line bandwidth.

7.1.1.2 Status Monitor (SM)

The Memshire Status Monitor is used to debug and measure performance of the memory controller subsystem by monitoring many status signals driven by the memory controller, such as:

- 200 data and 200 match signals
- No advanced filtering or state sequencing
- 2 counters
- Asynchronous boundary crossing inside the SM

There are 200 debug signals from the memory controller subsystem that can be traced. A debug mask ESR is used to select specific signals of interest.

8 Electrical Specifications

8.1 Absolute Maximum Ratings

This information will be included in a future release of this document.

8.2 Recommended Operating Conditions

This information will be included in a future release of this document.

8.3 DC Electrical Characteristics

Additional details will be included in a future release of this document.

8.4 AC Electrical Characteristics

This information will be included in a future release of this document.

9 Packaging

9.1 Package Thermal Information

This information will be included in a future release of this document.

9.2 Solder Reflow Profile

This information will be included in a future release of this document.

9.3 Moisture Sensitivity Levels

This information will be included in a future release of this document.

9.4 Restriction on Hazardous Substances (RoHS) Compliance

The ET-SoC-1 is compliant with Directive 2011/65/EU of the European Parliament and of Council of 8 June 2011 and its amendment Directive (EU) 2015/863 of 31 March 2015 on the restriction of the use of certain hazardous substances in electrical and electronics equipment (RoHS Directives).

In addition, the following regulated substances known to be in various types of electronic devices, are not intentionally added or knowingly present in the ET-SoC-1:

- Asbestos
- Azo Colorants
- Cadmium and its compounds
- Hexavalent Chromium and its compounds
- Mercury and its compounds
- Ozone Depleting Substances (CFCs, HCFCs, HBFCs, carbon tetrachloride, etc.)
- Tributyl Tin Oxide (TBTO)
- Tributyl Tin (TBT)
- Triphenyl Tin (TPT)
- Polychlorinated Biphenyls (PCBs)
- Polychlorinated Terphenyls (PCTs)
- Polychlorinated Naphthalenes (more than 3 chlorine atoms)
- Short chain Chlorinated Paraffins (SCCPs)
- Polybrominated biphenyls (PBBs)
- Polybrominated diphenylethers (PBDEs) including Deca-BDE and Radioactive Substances.

9.5 Package Outline and Dimensions

Figure 9-1 shows the package outline and dimensions for the ET-SoC-1 device.

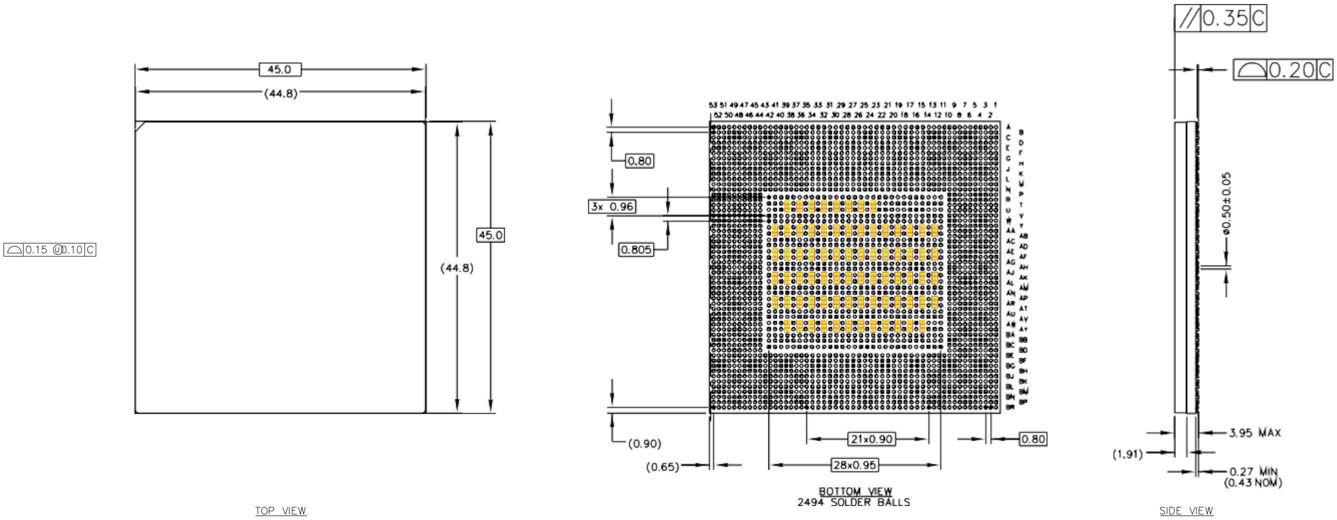

Figure 9-1 ET-SoC-1 Package Outline and Dimensions

10 Product Ordering Information

10.1 Package Marking

Figure 10-1 identifies the various markings on the ET-SoC-1 package.

Figure 10-1 ET-SoC-1 Package Markings

Table 10-1 provides additional packaging information.

Table 10-1 ET-Soc-1 Packaging Information

Marking	Type
ET-SoC-1	Product name
T7G923.00	Fab lot identifier
A0	Die revision
LOT 4	Assembly lot number
2103	Date code
ES	Engineering sample
TAIWAN	Country of assembly

Appendix A Glossary of Terms

Table 10-2 defines terms that are commonly used throughout this document.

Table 10-2 Glossary of Terms

Mnemonic	Name	Mnemonic	Name
AHB	Advanced high performance bus	ML	Machine Learning
AMBA	ARM-based bus interface standard	Neighborhood	A group of 8 ET-Minion cores
APB	Advanced peripheral bus	NoC	Network on chip
BL0	Boot level 0. Lowest and most secure boot level.	PC	Program counter
BL1	Boot level 1. Second level boot structure	PCIe	Peripheral Component Interconnect express
BL2	Boot level 2. Third level boot structure	PCIe Shire	PCIe control block
CPU	Central Processing Unit	PHY	Physical layer interface
CSR	Control and status registers	PI	Packed integer
D-cache	Data cache	PLIC	Platform level interrupt controller
DDR	Dual data rate	PLL	Phase lock loop
eMMC	Embedded multi-media card	PMU	Performance monitoring unit
ESR	Esperanto status registers	PPN	Physical page number
ET-Maxion	General purpose RISC-V core (x4)	PS	Packed single
ET-Minion Shire	A unit of 32 ET-Minion cores	PTE	Page table entry
ETLink	Proprietary Esperanto internal interconnect	PTW	Page table walker
FE	Front end	PU	Peripheral Unit
FIFO	First in first out	PVT	Process, voltage, temperature
FLB	Fast local barrier	RSA	Rivest-Shamir-Adelman cryptographic algorithm
FLEN	Floating point register length	SCP	Scratchpad RAM
FMA	Floating point multiply add	SHA	Secure hash algorithm
FSM	Finite state machine	SP	Service processor
GPIO	General purpose input/output	SPIO	Service processor I/O
I2C	Inter-integrated circuit bus	RISC-V	Reduced instruction set computer, release 5.
I3C	Inter-integrated circuit bus	Shire	A group of 4 Neighborhoods with 32 ET-Minion cores.
I-cache	Instruction cache	SIMD	Single instruction multiple data
IMA	Integer multiply-accumulate	SMBus	System management bus
INT	Integer unit of ET-Minion core	SoC	System-on-Chip
IPI	Inter-processor interrupt	SPI	Serial peripheral interface
JTAG	Joint test action group	TLB	Translation lookaside buffer

Table 10-2 Glossary of Terms (Continued)

Mnemonic	Name
L2	Level 2 cache
L3	Level 3 cache
LPDDR	Low power dual data rate
Mb	Megabits
MB	Megabytes
Memory Shire	Logic block for controlling DDR memory accesses
UART	Universal asynchronous receiver-transmitter. UART signals include: CTS: Clear to Send RTS: Request to Send DSR: Data Set Ready DTR: Data Terminal Ready RI: Ring Indicator DCD: Data Carrier Detect
USB	Universal serial bus
USB2	Universal Serial Bus version 2.0
USB-OTG	Universal Serial Bus-On The Go
VPU	Vector processing unit of ET-Minion core.