The Basic Linear Algebra Subprograms (BLAS) are a set of low-level subroutines that perform common linear algebra operations. BLIS is a software framework for instantiating high-performance BLAS-like dense linear algebra libraries. BLIS[1] was chosen over GoTOBLAS, ATLAS, etc. due to its portable micro-kernel architecture and active user-base.

**BLIS features**

- ISO C99 code with flexible BSD license.
- Support for BLAS API calling conventions.
- Competitive performance [2].
- Multi-core friendly.
- Multi-layer API and code identifying and isolating a key set of computational kernels.
- Modularity and extensiveness.
- Portability (x86, x64, TIC66x, PowerPC, etc.) that doesn’t impede high performance [3].
- Foundation for mixed precision (experimental).

**Level-3 BLAS using BLIS on Myriad**

**BLIS Level 3 BLAS**

1. **CROSSMEM general memory management.
2. **TRSM** transient memory copies.
3. **GEMM/TIPS** temporary storage of right hand side.

**Mapping of matrix blocks on CMX (SGEMM)**

**BLIS Level 3 micro-kernels**

- GEMM: general matrix-matrix multiplication.
- TRSM: triangular matrix solving.
- TIPS: temporary storage.

**BLIS Execution on Leon (C99)**

- Execute on Leon (C99)
- Execute on SHAVEs (C99)
- SHAVE assembler
- CMX memory management
- Port

**Mapping of matrix blocks on CMX (SGEMM)**

- A
- B
- C
- Inner kernels/ block-kernels
- Micro-kernels

**Level-3 BLAS on Myriad**

**GEMM and TRSM operations**

- Routine: Operation
- GEMM: \( C := \alpha \cdot (A \cdot B) + C \) (Cmn X X' X'Cn)
- TRSM: \( C := \alpha \cdot (A' \cdot C) / C \) (nnm X n X'T

**Memory focused optimizations**

- Double/triple buffering of arguments.
- Buffers shared by all SHAVEs.
- Data passing using pointer arithmetic.
- Overlapped DMA accesses.

**High performance on Leon (C99)**

- Execute on Leon (C99)
- Execute on SHAVEs (C99)
- Kernels in SHAVE assembler
- CMX memory management

**Myriad architecture highlights [4]**

- 65nm ultra-low power architecture (≤ 0.35W/180MHz) with 11 power islands.
- Hardware support for SIMD, matrix transpose, spare data, sqrt@fp16, predicated execution...
- Heterogeneous SoC: 1 Leon3@fp64 + 8 Shaves@fp32.
- 32KB LRAM, 1MB CMX, 16/64MB DDR, DMAs.
- Power efficiency of 1TOPS/(W (max 8-bit equivalent).

**Myriad 2 architecture highlights [4]**

- 28nm ultra-low power (≤ 0.5W/600MHz) with 17 power islands.
- Extended hardware support over Myriad 1: clock-gating, hard-wired configurable accelerators for imaging and vision, etc.
- Heterogeneous SoC: 2 Leon4@fp64 + 12 Shaves@fp32.
- 256+32KB LRAM, 2MB CMX, DDR3 support, DMAs.
- Power efficiency of 2TOPS/W (max 16-bit equivalent).

**SGEMM and STRSM performance**

- **SGEMM vs #cores and matrix width:**
- **STRSM vs #cores and matrix width:**

**Level-3 BLAS on Multi-core Media-Processor SoC**

Myriad architecture prioritises power-efficient operation and area efficiency. In order to guarantee sustained high performance and minimise power the proprietary SHAVE (Streaming Hybrid Architecture Vector Engine) processor was developed. Data and instructions reside in a shared Connection Matrix (CMX) memory block shared by all Shave processors. Data is moved between peripherals, processors and memory via a bank of software-controlled DMA engines.