Barcelona Supercomputing Center
Centro Nacional de Supercomputación

esa

European Workshop on On-Board Data Processing
14-17 June 2021 | Online Event
OBDP 2021

OBPMark (On-Board Processing Benchmarks) – Open Source Computational Performance Benchmarks for Space Applications

David Steenari[1], Leonidas Kosmidis[2,3], Ivan Rodriguez-Ferrandez[3,2], Alvaro Jover-Alvarez[3,2], Kyra Förster[1]

[1] ESA/ESTEC
[2] Barcelona Supercomputing Center (BSC)
[3] Universitat Politècnica de Catalunya (UPC)

OBPMark Overview

- OBPMark ("On-Board Processing Benchmarks") is set a computational performance benchmarks for space applications

- It is currently under development by ESA and BSC/UPC
  - A private beta version currently available – **beta repository now open**
  - Early usage in a number of ESA activities

- Funding through the GPU4S activity (see previous presentation)
  - Builds on several lessons-learned from **GPU4S Bench**
  - Next phase of activity will focus on the finalization of OBPMark

- OBPMark and GPU4S Bench are provided together
  - GPU4S Bench focuses on benchmarking of parellelised processing building blocks
  - OBPMark focuses on **application level benchmarks for space**
    - OBPMark applications reuse optimized GPU4S Bench building blocks
    - Porting GPU4S algorithms can be the starting point for new architecture ports

OBPMark (On-Board Processing Benchmarks) – Open Source Computational Performance Benchmarks for Space Applications

BACKGROUND AND MOTIVATION

Background

- Traditionally, processing in space has been dominated by single core CPUs and DSPs; FPGAs and dedicated ASICs

- But as of late, several new device classes have been introduced for on-board applications, driven by influx of COTS components and need for flexible processing:

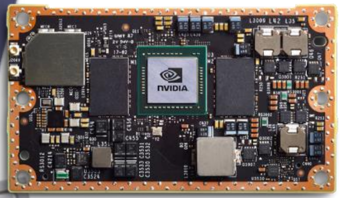
Multicore processors (GR740, LS1046); MPSoCs (ZynqUS+, NG-ULTRA, Cyclone-V); array processors (HPDP); many core DSPs (RC64, MPPA); GPUs (NVIDIA, AMD); VPUs (Myriad)

→ See also presentation tomorrow (16th of June) at 13:00:
"Survey of High-Performance Processors and FPGAs for On-Board Processing and Machine Learning Applications"

- Different device classes increase the difficulty for easily and accurately comparing computational performance in key on-board processing applications

- Lack of openly available benchmarks for space applications identified during GPU4S study (ESA GSP activity)

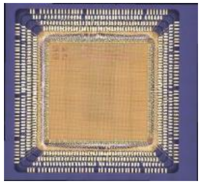
GR740 (Gaisler)

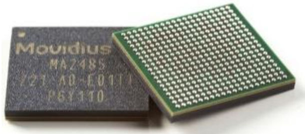


TX2 (NVIDIA)

ZUS+ (Xilinx)

Kalray (MPPA)


HPDP (ISD)



Myriad 2 (Intel)

Benchmarking Challenges

- Difficult to determine which are the most performant and efficient processing systems, due to:
  - Lack of benchmarking data in all device categories (CPUs, GPUs, VPU, FPGAs, etc.)
  - Lack of standardized tests (standard only targeting CPUs or OpenCL compliant)
  - Operations/s (FLOPS, GMACs) do not give the whole picture. Neither do synthetic benchmarks

- Commercial benchmarks mostly cover application-specific benchmarks or dependent on software implementation for single/multicore CPUs
  - E.g. Dhrystone, CoreMark, SPEC, PARSEC (multi-threaded programs), NAS (parallel)
  - ADASMark and MLPerf etc

- Commercial benchmarks are not fully usable for space:
  - Performance per power metric not included
  - Tasks not representative of on-board processing (hyperspectral, radar, etc.)
  - FPGAs not considered, high level of parallelization not considered
  - Not easily portable to all devices (FPGAs, HPDP, RC64, Kalray, etc.) or heterogeneous systems

- GPU4S Bench → step in right direction (open, reproducible) covers processing building blocks
- **Need for a set of application benchmarks that cover "most" typical OBP and is openly available**

OBPMark (On-Board Processing Benchmarks) – Open Source Computational Performance Benchmarks for Space Applications

OBPMARK DESCRIPTION

OBPMark Objectives

- To promote a standard set of **application-level benchmarks**, as to enable a method of comparing end-user performance of different devices and systems – such as both RHBD and COTS processors, FPGAs and ASICs

- To better understand **limitations of different types of devices and systems** and to quickly decide on division task in hardware and software for implementations in heterogeneous systems

- To **identify new (COTS) components and architectures** that have a exceptional performance for specific processing applications

- To allow ESA to quickly **provide recommendations for processing systems in future missions**, through identifying key parameters together with the project teams

- Benchmark standard on-board processing functions, so that implementers will have the possibility for reusing the **invested work in real-world use cases**

OBPMark Requirements Overview

- Coverage:
  - …shall cover common OBP applications: image processing, compression, radar processing, encryption, common building blocks (for radar, radiometry, SDR, etc) and machine learning
  - …shall allow to add future benchmarks (through version update).

- Comparable:
  - …ensure identical output among different implementations
  - …shall provide comparable results for: overall performance, performance / power, absolute power.
  - …shall provide all necessary configuration parameters and test data, including golden reference output.

- Portable:
  - …shall be written in standard C.
  - …shall support standard parallelization schemes: OpenMP, OpenCL and CUDA (and new ones in future releases)
  - …shall be possible to port to FPGA implementations*.

- Openness:
  - …shall be openly available (open-source license, open repository)
  - …shall be open for community response/feedback
  - …shall be open for community contributions (porting etc.)

OBPMark Components

- OBPMark has the following components:
    1. A Technical Note (TN) defining the benchmark algorithms and parameters, and defines a result reporting template
    2. Reference input and output data for verification
    3. Reference implementations in standard C-code, and standard parallelisation schemes: OpenMP, OpenCL and CUDA
    4. A database of reported test results

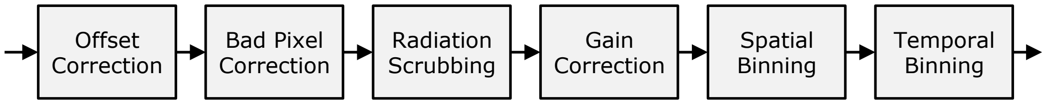
OBPMark Benchmark Lists for V1.0

ID	Benchmark Name	Sub ID	Sub-Benchmark Name	Status
#1	Image Processing	#1.1	Image Calibration and Correction	Done
		#1.2	Radar Image Processing	TBD. GPU4S CCN2
#2	Standard Compression Algorithms	#2.1	CCSDS 121.0 Data Compression	Done
		#2.2	CCSDS 122.0 Image Compression	Done
		#2.3	CCSDS 123.0 Hyperspectral Image Compression	On-going
#3	Standard Encryption Algorithms	#4.1	AES128	TBD. GPU4S CCN2
#4	Common Processing Building Blocks	#5.1	FIR Filter	Done
		#5.2	FFT Processing	Done
		#5.3	Convolutional Filter	Done
		#5.4	Matrix Multiplication	Done
#5	Machine Learning Inference	#6.1	Object detection	TBD. GPU4S CCN2
		#6.2	Cloud screening	TBD. GPU4S CCN2

Benchmark #1: Image Processing

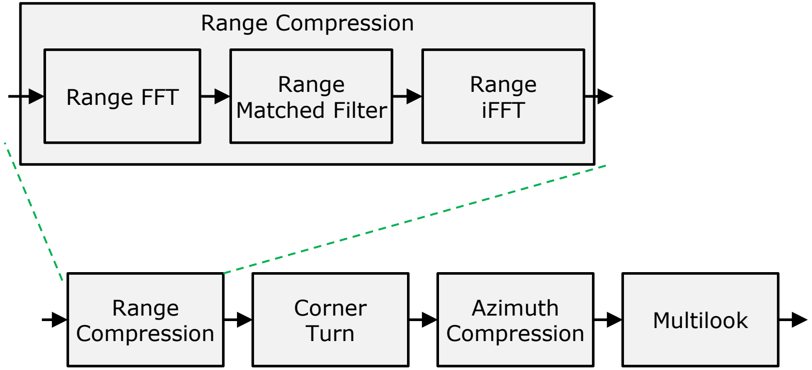
- On-board image processing is a typical on-board application for Science and EO

- Two sub-benchmarks defined
  #1.1 Image Calibration and Correction
  #1.2 Radar Image Processing



- #1.1 is based on typical processing performed in ESA science optical payloads with long exposure times

- #1.2 is based on the range-Doppler algorithm
  - Reconstruction of radar images on-board can enable additional on-board processing applications



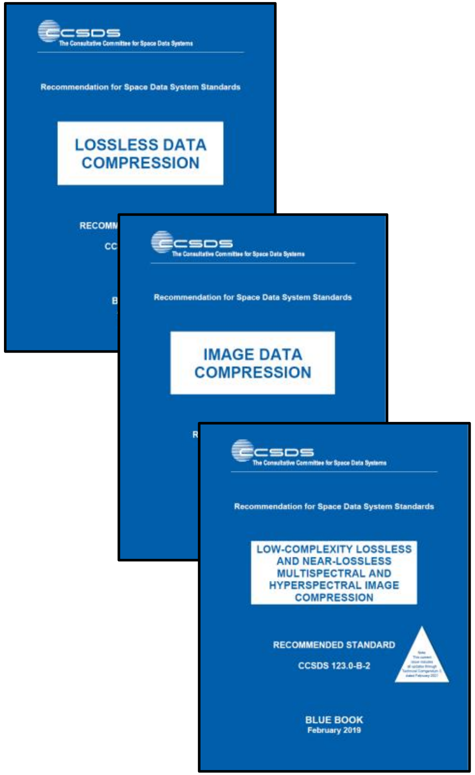
Benchmark #2: Standard Compression

- CCSDS compression algorithms are widely used on ESA (and NASA) missions
- Currently three standards available: CCSDS 121.0-B-3 Lossless Data Compression; CCSDS 122.0-B-2 Image Data Compression; CCSDS 123.0-B-2 Low-Complexity Lossless and Near-Lossless Multispectral and Hyperspectral Image Compression

- The throughput of each of the algorithms depends on multiple compression settings
  - In particular the CCSDS 123 has many dozens of parameters
  - Performance and throughput of two implementations cannot be compared fairly unless the same parameters and data are used

- Three sub-benchmarks defined:
  - #2.1 CCSDS 121.0 Data Compression
  - #2.2 CCSDS 122.0 Image Compression
  - #2.3 CCSDS 123.0 Hyperspectral Compression

- Existing compressor implementations can be benchmarked using the provided input and output verification data; and configuring the specified compressor parameter settings
- New implementations can be reused for future payload applications



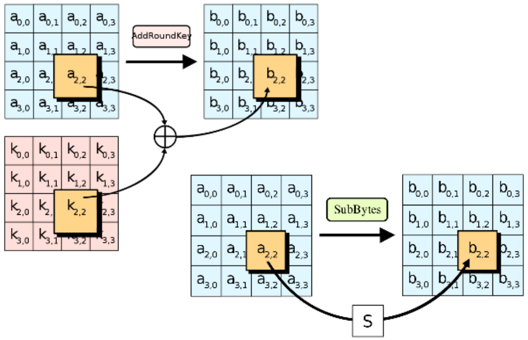
Benchmark #3: Standard Encryption

- On-board encryption used for sensitive data, in particular in commercial applications

- Currently one encryption benchmark defined, choice based on existing space standard:

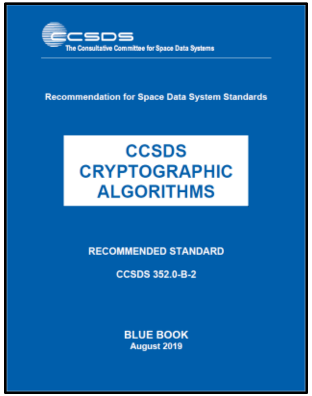
  CCSDS 352.0-B-2, CCSDS Cryptographic Algorithms

- … which uses the well-known AES encryption standard



- Benchmark #3.1 defines guidelines for benchmarking of AES and provides input and output data for verification
- Several key-lengths defined: 128-bit, 192-bit, 256-bit
- Performance parameters include: samples/s and samples/s/W

- Implementations can be reused in on-board applications


CCSDS
The Consultative Committee for Space Data Systems

Recommendation for Space Data System Standards

CCSDS CRYPTOGRAPHIC ALGORITHMS

RECOMMENDED STANDARD

CCSDS 352.0-B-2

BLUE BOOK
August 2019

Benchmark #4: Building Blocks

- To increase coverage of application cases to several signal processing applications that were so far too complex to include in their entirety (SDR processing, sounders, radiometry) – it was decided to include a number of reoccurring processing building blocks

- Four sub-benchmarks specified:
  - #4.1 – FIR filter
  - #4.2 – FFT
  - #4.3 – Convolution Kernel
  - #4.4 – Matrix Multiplication

- Based on implementations provided in the GPU4S Bench
  - In fact, mostly identical to GPU4S implementation

- Well-known algorithms with extensive usage for other benchmarking fields
  - e.g. matrix multiplication is often used in non-space related benchmarks for parallel processors
  - also sometimes used to measure errors in processors/GPUs during radiation testing

Benchmark #5: Machine Learning

- Survey performed of openly available annotated / labeled training data sets for machine learning applications
  - Survey also included possible standard CNN architectures that could be applied

- Two sub-benchmarks tentatively defined:
  - #5.1 Object detection
  - #5.2 Cloud screening

- Object detection tentatively to target ship (or airplane) detection
- Cloud screening a common application, lately using machine learning (see e.g. Φsat-1 and CHIME)

- Approach
  - To use standard models (e.g. SSD MobileNetv2 for object detection)
  - Provide pre-quantized models in standard formats (TF, TFLite, etc.) for different data types (INT8, INT16, FP16, FP32)
  - Training data openly available (if re-training is required to support specific platforms)

- Implementation work to be covered in CCN2 of GPU4S – workshop planed to finalise benchmarks' definition
- In parallel: on-going ESA (GSTP) activities on ML benchmarking on several devices – see next presentation:
  "Machine Learning Application Benchmark for satellite on-board data processing", Max Ghiglione, ADS
- On-going effort to coordinate and harmonize ML performance benchmarks

Possible Future Benchmark Additions

- OBPMark is designed to allow expansion with additional benchmarks
- Possible future extensions include (pending availability and community feedback):
  1. Video encoding / compression
     - Video compression is currently being studied in ESA activities, i.e. evaluation of JPEG and H.264/H.265 standards
     - Possible future standardization of recommendation of video encoding in space
  2. Additional compressors – e.g. SAR compressor currently under standardization
  3. Additional ML/AI benchmarks
     - Expand scope outside of Convolutional Neural Networks (CNNs), with other methods as they evolve (e.g. RNNs, SNNs, etc.)
  4. AOCS/GNC processing and/or visual-based navigation
  5. Full SDR processing chain, e.g. DVB-S2 encoding/decoding

OBPMark Availability

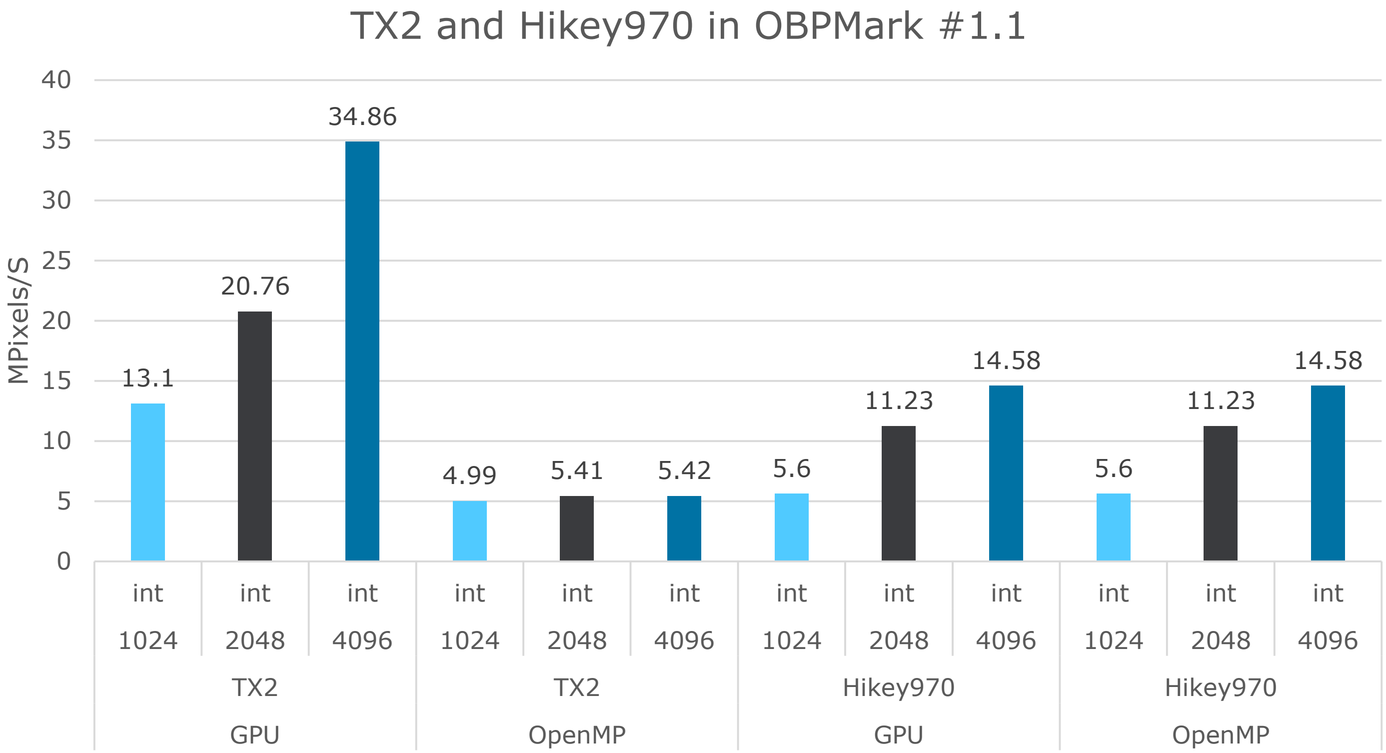
- Both OBPMark and GPU4S Bench are available on the public git repositories
  - See OBPMark.org for more information
  - Currently in **beta mode**, seeking community feedback from early adopters

- Benchmarks already available:
  - #1.1 – Image Calibration and Correction
  - #2.1 – CCSDS 121.0 Data Compression
  - #2.2 – CCSDS 122.0 Image Compression
  - #4.1 – FIR filter
  - #4.2 – FFT
  - #4.3 – Convolution Kernel
  - #4.4 – Matrix Multiplication

- Additional benchmarks will be added during 2021, throughout progress of next phase
- Full version to be released by end of 2021
- **Want to start benchmarking? Get in contact OBPMark@esa.int**

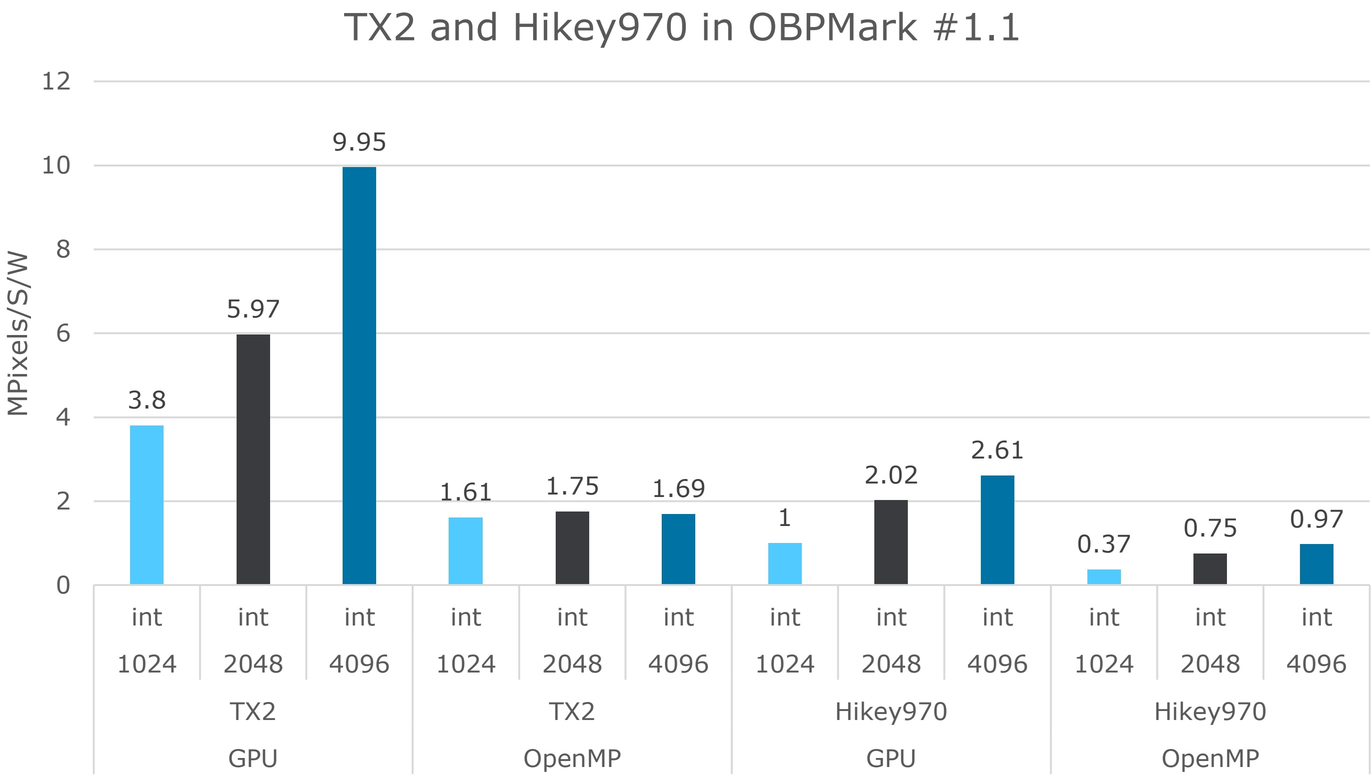
- **Next slides – some selected benchmark results from BSC/UPC (targeting GPU SoCs)**

Benchmark #1.1 Image Calibration and Correction - Performance / Throughput



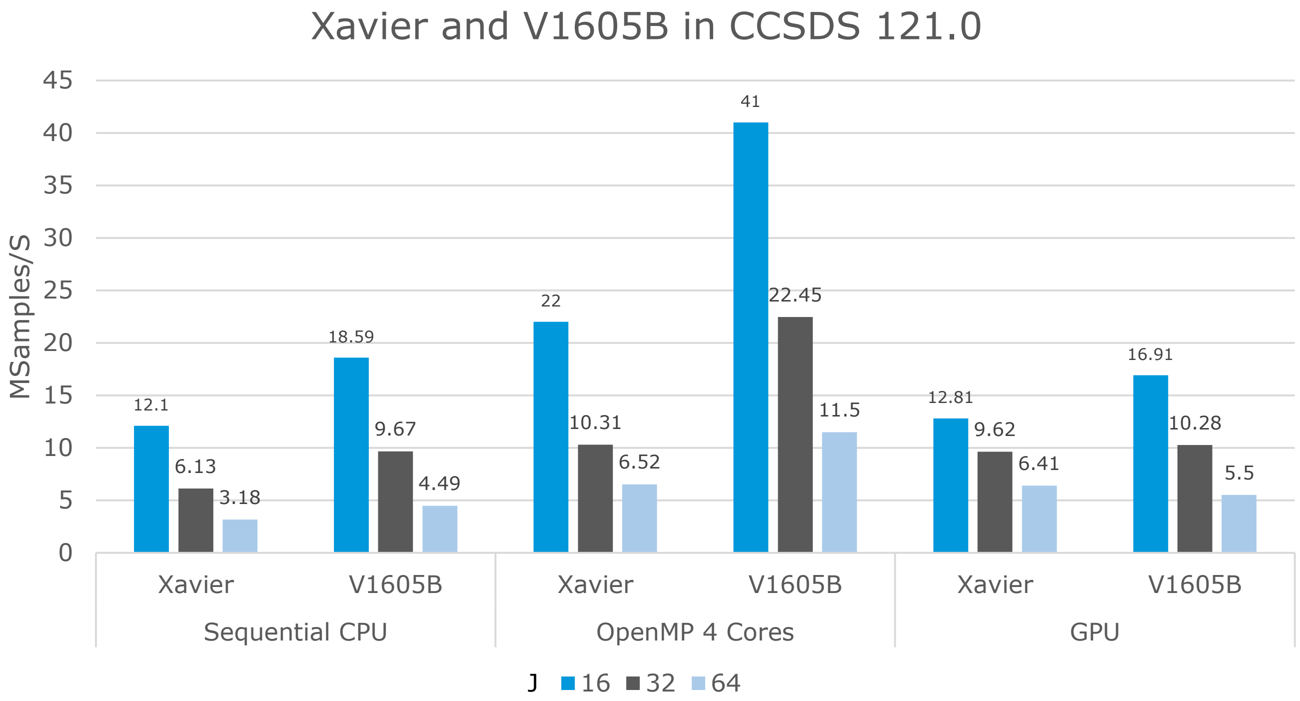
- Benchmark details:
  - Random images
  - TX2: CUDA
    - 15W TDP mode
  - Hikey970: OpenCL
    - 15W Mode
  - OpenMP Cores: 4
  - Tested with different image sizes

Benchmark #1.1 Image Calibration and Correction – Energy Efficiency



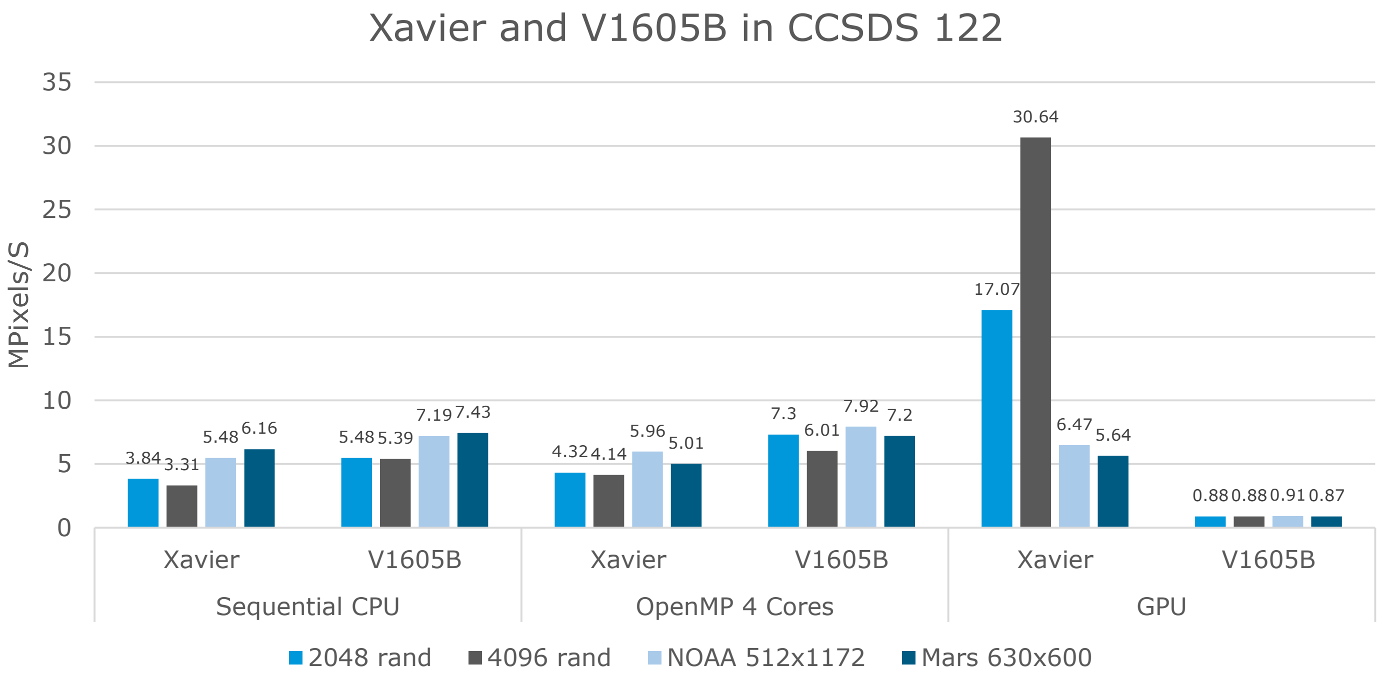
- Benchmark details:
  - Random images
  - TX2: CUDA
    - 15W TDP mode
    - Energy efficiency computed with measured power consumption
  - Hikey970: OpenCL
    - 15W Mode
    - Energy efficiency computed with measured power consumption (only OpenCL)
  - OpenMP Cores: 4
  - Tested with different image sizes

Benchmark #2.1 CCSDS 121.0 Data Compression – Performance / Throughput



- Benchmark details:
  - Xavier: CUDA
    - 15W TDP Mode
  - V1605B: OpenCL
    - ~15W Mode
  - OpenMP Cores: 4
  - Tested with 16MB random input divided in 1024 steps with 256 sample intervals for different Block Sizes J (16, 32 and 64)

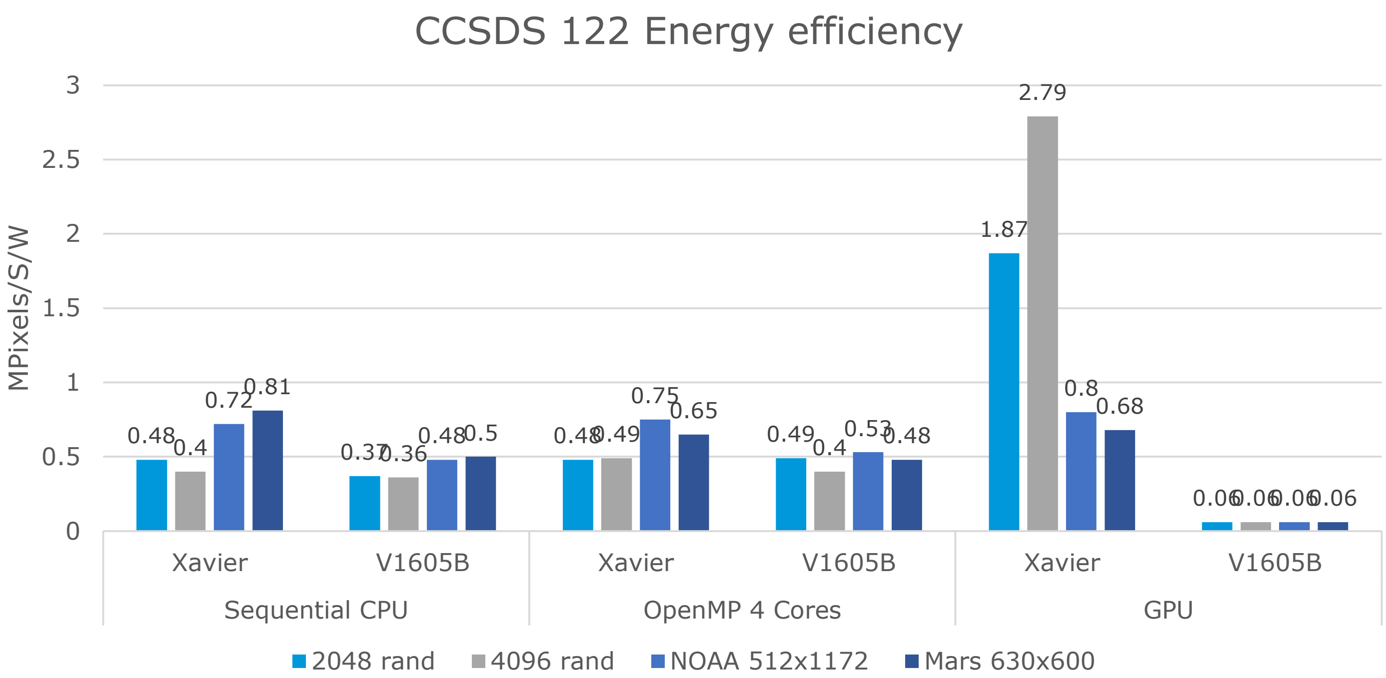
Benchmark #2.2 CCSDS 122.0 Image Compression – Performance / Throughput



- Benchmark details:
  - Real and Random images
  - Xavier: CUDA
    - 15W TDP mode
  - V1605B: OpenCL
    - ~15W Mode
  - OpenMP Cores: 4
  - Tested with different image sizes

- Note V1605B GPU results are **NOT** representative (on-going issue with driver)

Benchmark #2.2 CCSDS 122.0 Image Compression – Energy Efficiency



- Benchmark details:
  - Real and Random images
  - Xavier: CUDA
    - 15W TDP mode
    - Energy efficiency computed with measured power consumption
  - V1605B: OpenCL
    - ~15W Mode
    - TDP used
  - OpenMP Cores: 4
  - Tested with different image sizes
- Note V1605B GPU results are **NOT** representative (on-going issue with driver)

OBPMark Summary

- A **lack of openly available performance general and reusable benchmarks for space** applications has been identified
- Currently benchmarks are **application-specific and (often) closed source**.

- The number of **different devices used for processing** on-board spacecraft is increasing, making accurate comparison of computational performances difficult

- A new set of Open Source Computational Performance Benchmarks for Space Applications is proposed: **"OBPMark (On-Board Processing Benchmarks)"**
- Provided as **complementary with GPU4S Bench**

- **Beta version** is available on the repository – community feedback is welcome
- Interested implementers are **encouraged to get in contact**

OBPMark (On-Board Processing Benchmarks) – Open Source Computational Performance Benchmarks for Space Applications

THANK YOU FOR YOUR ATTENTION!

Contact: OBPMark@esa.int

Author: David.Steenari@esa.int