Embedded Deep Learning in Space : Artificial Intelligence with Qormino®

Abstract

Artificial Intelligence (AI) algorithms are known to be highly demanding in terms of computing resources. Thanks to the increase of computational power of the latest processing devices, **AI is also becoming popular for the Space industry** for various applications such as **On-board data processing** for observation satellites, **automated guidance of Spacecrafts**, **On-board decision for collision prevention**, **Communication** satellites, **Fusion of data sources** for better predictability, …

Until recently, Space industry was facing the challenge to get access to state-of-the-art processing components that would comply with Space requirements, i.e. high reliability, robustness, and radiation tolerance.

Led by the Grenoble University Space Centre (CSUG), the QlevEr Sat project **leverages the high computing capabilities of Qormino QLS1046-Space** radiation tolerant processing modules to run AI algorithms on-board, together with the **high resolution of the images** taken by the **Emerald sensor**.

This white paper first presents the **general performances and functionality of the QLS1046-Space processor**. Then, the **main results from those benchmarking activities** are given, to **demonstrate the feasibility to use QLS1046-Space to run embedded AI in Space**.

Introduction

Artificial Intelligence (AI) algorithms are known to be highly demanding in terms of computing resources. Thanks to the increase of computational power of the latest processing devices, AI is becoming popular for ground applications. AI now competes with traditional data processing in a number of applications, such as face recognition, autonomous driving, or robots.

The Space industry can also benefit from AI in various applications:

- On-board data processing for early warnings to situations,
- Observation and meteorological satellites, where on-board processing allows to send only relevant and pre-processed data to the ground, reducing downlink bandwidth requirements,
- AI can improve performance in automated guidance of Spacecrafts in critical maneuvers such as docking or landing,
- On-board decision allows better collision prevention thanks to early reaction, and offers possibilities of self-health monitoring and ultimately autonomous self-reconfiguration,
- Communication satellites can benefit from smart data routing and optimized antenna pointing based on actual traffic and weather conditions to increase data rate and minimize power consumption,
- Fusion of data sources from various kind of sensors, allowing to see what is not visible to the “human eye”, including on-board analysis of large data sets in deep Space and Science missions.

Until recently, despite this wide range of new possibilities, Space industry was facing the challenge to get access to state-of-the-art processing components that would comply with Space requirements, i.e. high reliability, robustness, and radiation tolerance.

Led by the Grenoble University Space Centre (CSUG), the QlevEr Sat is developing a nanosatellite using artificial intelligence algorithms to observe the Earth and meet social challenges such as observation of illegal deforestation, monitoring of CO2 emissions or evaluation of damages after a natural disaster.

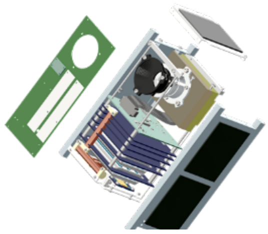

Figure 1 : QlevEr Sat Nanosatellite

This smart satellite will embed an Emerald 16MP image sensor and a Qormino® QLS1046-Space processing module, both new radiation-tolerant and Space-qualified components from Teledyne e2v. The project leverages the high computing capabilities of QLS1046-Space to be able to run the AI algorithms on-board, together with the high resolution of the images taken by the Emerald sensor.

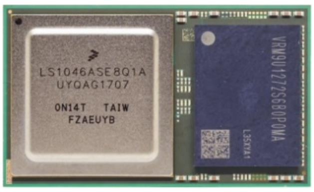

Figure 3: Qormino® QLS1046-4GB

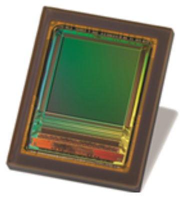

Figure 2: EMERALD Sensor

In the frame of this project, a part of the feasibility study aimed at verifying the computing capability of the Qormino QLS1046-Space for AI algorithms. This white paper first presents the general performance and functionality of QLS1046-Space. Then, main results obtained in those benchmarking activities are given, demonstrating the feasibility to use QLS1046-Space to run AI in Space.

I. General performance and functionality of Qormino® QLS1046-Space

Qormino is a line of processing modules from Teledyne e2v dedicated to Space and High-reliability applications. Those modules combine GHz-class multicore processors, with high speed DDR4 memories, in compact 44 x 26mm dimensions. They come in a 0.8mm BGA package, and are designed to respond to SWaP (Size, Weight and Power) constraints. With built-in DDR4 bus layout and "building-block" approach, design is facilitated while guaranteeing a high performance.

QLS1046-Space is the Qormino version dedicated to Space. It embeds a Quad-Core Arm® Cortex®-A72 Microprocessor running up to 1.8GHz, with ECC-protected L1 and L2 cache memories for reliable behaviour. It features a rich set of peripherals, including integrated packet processing acceleration, high speed serial links supporting 10 Gb Ethernet, PCIe® Gen3, SATA 3.0 and USB, as well as a number of general purpose interfaces such as SPI, I²C, and UART. The current version integrates 4GB of DDR4 with transfer speed up to 2.4GT/s, and a version with 8GB is also targeted.

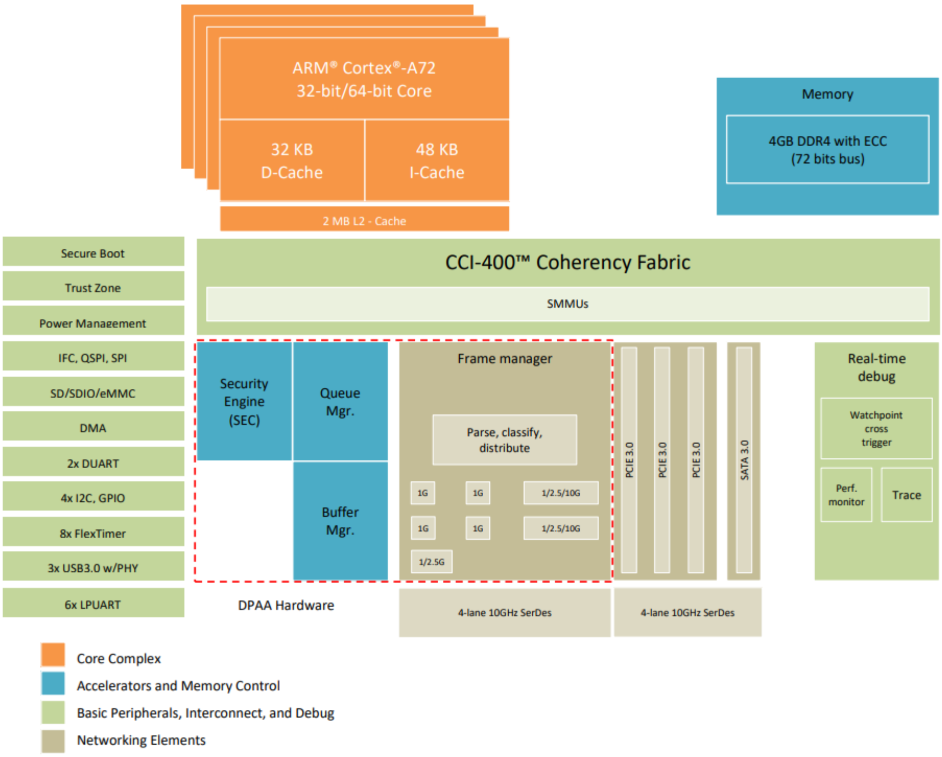

Figure 4: Architecture of QLS1046-4GB-Space

Apart from the pure performance aspect, the reason for selecting this device is that it is Space-compliant. Both the processor and the memory are radiation tolerant:

- SEL free up to more than 60MeV.cm²/mg
- Known SEU/SEFI cross-sections up to more than 60MeV.cm²/mg
- TID: 100krad (Si)

In addition, QLS1046-Space and its components are qualified, manufactured, and screened following NASA or ECSS standards.

II. Benchmark & Results

Benchmarking activities were performed to verify in practice the computing capability of QLS1046-Space to run AI algorithms for Space applications. The focus is mainly on AI for image processing, since the QlevEr Sat project targets earth observation use cases. In this study, only neural networks with deep learning have been tested. Classical machine learning usually requires less computing resources, thus it would be expected to get even better results in machine learning.

In this study, the performances of QLS1046-Space were evaluated on three different axes:

1) The pure computing performances were evaluated in terms of GFLOPS (Giga Floating Point Operations Per Second), since this is the typical way of evaluating the computing performance of a device in AI applications.
2) An inference benchmark was performed to verify the capability of the device to execute neural networks. Several classical neutral network architectures have been tested.
3) Training performance was briefly assessed, to evaluate the possibility of applying learning or fine-tuning on QLS1046-Space.

Benchmark setup

The performance assessment was realized with a QLS1046-Space development kit, which has a number of available interfaces. The operating system used throughout the benchmark was Linux (Ubuntu 18.04). The QSL1046-Space device inside the development kit had 4GB of integrated DDR4 memory. The version with 8GB of DDR4 memory would have been more efficient to execute AI, but it was not available at the time of the testing. In addition, the processor was running at 1.6GHz, instead of 1.8GHz maximum frequency. This means that the results presented in this white paper are somewhat limited by the amount of DDR4 memory available and the running frequency of the processor.

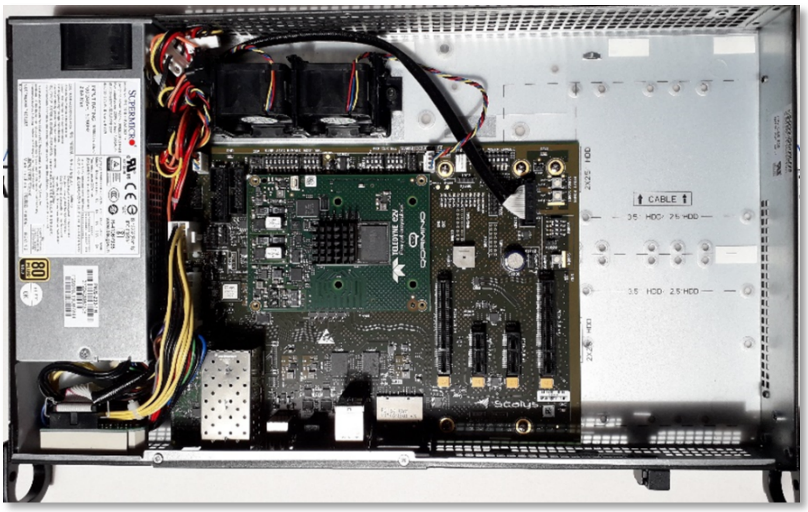

Figure 5 : QLS1046-Space Development Kit

In some of the following benchmark results, a regular computer was used as a basis for comparison to rate QLS1046-Space performance. This computer had an Intel® Core™ i7-9750H processor running at 2.6GHz and 32GB of DDR4. It was running Linux. It is considered as a good computer to perform AI, which is why it is a convenient reference in the following.

Benchmark results

Performances of QLS1046-Space were evaluated on the following three axes:

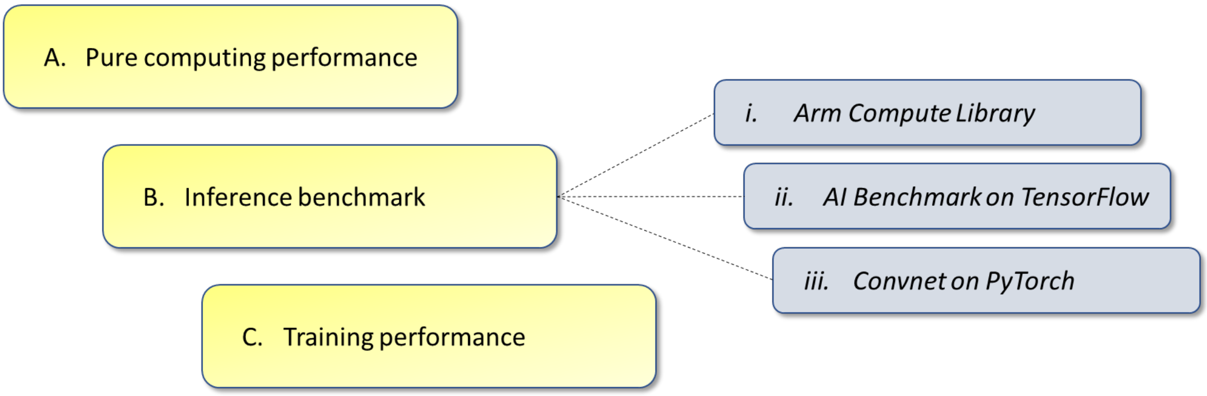

Figure 6: Benchmarks

A. Pure computing performance

For the pure computing performance evaluation, the benchmark [1] was used, which consists in a small and simple test software. In the results, the performance of QLS1046-Space is compared to that of the computer with the Intel® Core™ i7-9750H processor. It should be noticed that the execution of the software does not take advantage the hardware accelerators of the processors. This explains in particular why the GFLOPS numbers obtained here are lower that what can be found in the literature for those processors. Figure 7 presents the pure results in GFLOPS to compare both targets. Figure 8 compares power efficiency since this is a key topic in Space applications.

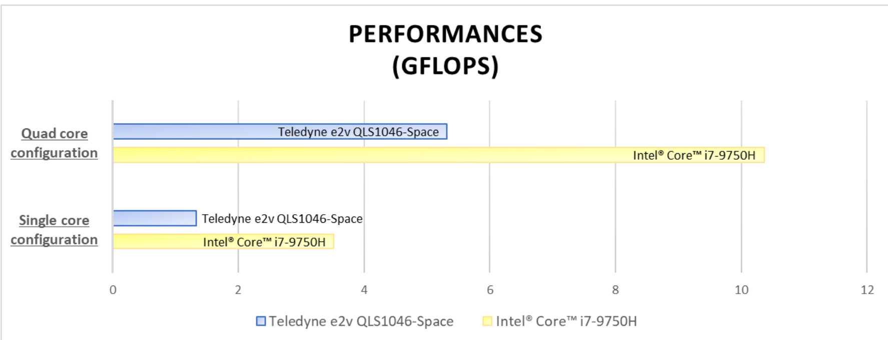

Figure 7: Summary of the computing performance comparison.

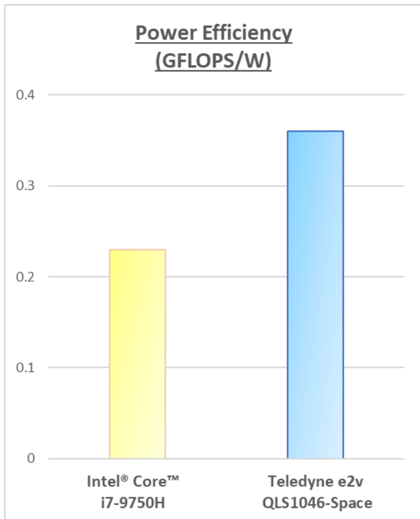

Figure 8: Power efficiency in quad core operation.

Calculated from thermal power characteristics of both devices, 45W @100°C for the i7 (Table 5-2 of [2]), 14.6W @105°C for QLS1046-Space (Table 8 of [3]).

It is observed that the gap between the two devices depends on the number of cores used, and with higher number of cores the difference in performance reduces. Those results highlight that QLS1046-

Space offers about half of the computing capabilities of the i7 in the quad-core configuration, which is known to be a good processor to perform AI on ground. Hence, QLS1046-Space offers a fair amount of computing performance to perform AI in Space. In addition, QLS1046-Space exhibits higher power efficiency making it well suited for Space systems.

B. Deep learning inference benchmark

In this benchmark, tests are performed to evaluate the performance of QLS1046-Space in inference, meaning when the device uses a neural network to process an image. Only classical neural networks are tested in the study, first with Arm Compute Library [5], then AI-Benchmark [4] on TensorFlow [7], and ConvNet [6] on PyTorch [8]. It should be noticed that the two most popular libraries used for IA are TensorFlow and PyTorch proposed by Google and Facebook respectively, with both libraries supported by Arm [9]. Those networks are pre-trained to identify objects in pictures and are widely used in the existing object classifiers such as r-cnn, fast-rcnn, faster r-cnn [10] or CenterNet [11]. However, TensorFlow and PyTorch libraries are evolving very quickly, and this is the reason for evaluating first the performance with Arm Compute Library, which is considered more stable.

i. Arm Compute Library

In this benchmark, Arm Compute Library [5] is used to run different classical neural networks. The results obtained on QLS1046-Space are shown in the Table 1:

Network	Execution time [ms]		Number of operations for an inference [MFLOP]	Computing performances [GFLOPS]	
	Single core	Quad core		Single core	Quad core
Alexnet	153	74	727	5	10
Googlenet	286	109	1500	5	14
Inception v3	848	314	6000	7	19
Inception v4	1870	655	13000	7	20
Mobilenet	118	44	570	5	13
Resnet50	501	206	4000	8	19
Squeezenet	145	64	360	2	6
Vgg16	1090	418	16000	15	38
Yolov3	6540	2500	66000	10	26

Table 1: Performance of QLS1046-Space with Arm compute library.

Those results confirm that it is possible to perform on-board image classification using QLS1046-Space, with this kind of common classifiers, and with reasonable execution time. Those results are especially interesting considering that Arm compute library is one of the major frameworks for AI.

ii. AI-Benchmark

AI-Benchmark [4] instantiates backbones in the TensorFlow format, which are very common neural networks originally created for image classification. The results of the benchmark for different neural networks are given in the Table 2:

Backbone	Picture size	Execution time [ms]	Variability [ms]	Description
VGG16[9]	224x224	1320	7	Network trained on ImageNet [12] to classify 1000 objects.
VGG19[9]	512x512	13562	144	Network trained on ImageNet [12] to classify 1000 objects.
ResNet-V2-50	346x346	868	5	Classifier based on residual neural network [13]
ResNet-V2-152	256x256	1538	18	Classifier based on residual neural network

Table 2: AI-Benchmark results on QLS1046-Space.

The results show that QLS1046-Space allows to perform an on-board image classification with classical neural networks in about 1s. This implies an optimized memory management with the use of FP16 type, and with picture size suitable with the memory available of 4GB. It is noticed that VGG19[9] is around 10 times longer to execute than other tests, which may be due to cache memories configuration and DDR4 size limitation.

Based on the results, QLS1046-Space obtains a score of 103. Neural networks are known to require large amounts of memory, hence the performance obtained here is limited by the DDR4 size of 4GB on the tested version. Much higher ranking is expected with an 8GB version.

iii. Convnet

In this benchmark, ConvNet [6] on PyTorch is tested on QLS1046-Space. Pytorch tends to be used more and more often over TensorFlow. PyTorch was originally more complex to use but was more flexible. From PyTorch version 1.8, an important reduction in complexity is expected to benefit to QLS1046-Space. It should also be noticed that PyTorch is now can handle tools such as SLURM [14] on pytorch-lightning [15]. Convnet benchmark results on PyTorch are given in the Table 3:

Network	Execution time [ms]	
	QLS1046-Space @1.6GHz	**Intel® Core™ i7-9750H @2.6GHz**
Alexnet	187	1.72
VGG11	764	4.28
ResNet50	578	7.29
Squeezenet1_0	328	2.28
Densenet121	1283	17.93
Mobilenet_v2	2337	6.38
Shufflenet	1278	8.49
Unet	1263	4.98

Table 3: Convnet results on QLS1046-Space.

The benchmark shows that the i7 is performing much faster than QLS1046-Space, which is limited again by the size of memory available. Despite the gap in performance, it is still considered that the performance level offered by QLS1046-Space is acceptable to implement on-board AI processing.

C. Deep learning training performance

Training performance using QLS1046-Space was quickly tested on Convnet with TensorFlow. It was not extensively tested since most up-to-date backpropagation [16] benchmarks require at least 8GB of RAM memory. Table 4 shows the comparison of the training time for one sample on ResNet50 between QLS1046-Space and the Intel® i7.

Network	Training time for one sample [ms]	
	On QLS1046-Space	On Intel® Core™ i7-9750H
ResNet50	3782	20

Table 4: Comparison of training performance.

This result clearly shows the penalty of the lack of RAM memory on the current version of QLS1046-Space for training on traditional image classifiers. It should be noticed that a complete training usually requires hundreds of samples. This result has to be mitigated due to the fact that image classifiers are known to be highly demanding in computing resources. Since it will be time-consuming to perform a complete training on QLS1046-Space, an alternative that can be considered is to perform fine-tuning [17] on-board.

Training small convolutional neural networks for simple detection use cases seems feasible with QLS1046-Space, as well as deep learning for processing time-series or 1-D signals. In terms of training capabilities on images, QLS1046-Space would be more efficient in classical machine learning, but those models are more complex to build.

III. Discussion

QLS1046-Space offers a decent amount a computing capability allowing to run deep learning AI for image processing in Space. The device is not as powerful as tailored-made solutions that are available for AI inference in ground applications, but it is the most powerful Space-qualified CPU available on the market. In terms of pure computing capabilities, it offers performance in the same order of magnitude as an Intel® Core™ i7-9750H. From the AI performance point of view, the main drawback of the current version is the 4GB memory, which requires an optimized memory management to run AI for image processing. On next versions with 8GB DDR4 memory or more, AI performance would be significantly increased, and would reduce the burden of optimized memory management.

Performance obtained in the previous benchmarks was evaluated with classical deep neural networks without taking advantage of the specific QLS1046-Space architecture. Different AI topologies are more optimized to run on embedded targets, which would bring a better efficiency of the AI running on QLS1046-Space. Apart from AI computing performance, the study shows that QLS1046-Space exhibits good power efficiency making it well suited for Space systems where electrical power is limited and power dissipation is an issue. From the electronic architecture point of view, it might be relevant to add an FPGA as a companion-chip for QLS1046-Space, in which case the FPGA could take care efficiently of the pre-processing, and QLS1046-Space would then perform the heavy work.

In this study, the primary focus was on deep learning AI for image processing, which is considered one of the most demanding application in terms of computing resources. For instance, processing of 1-D time series is much less demanding that image processing. Hence, the outcome of the study is that QLS1046-Space would also be suitable for other AI applications in Space, such as heterogeneous data analytics, on-board decision making, and ground to Space and satellite to satellite communications. QLS1046-Space would also be able to address traditional machine learning applications.

References

[1] S. Das, shaswata56/BenchUtil. 2020.

[2] Datasheet of the Intel® Core™ i7-9750H processor.

[3] Datasheet of QLS1046-4GB.

[4] ≪ AI-Benchmark ≫. http://ai-benchmark.com/ranking_deeplearning_detailed.html

[5] ARM-software/ComputeLibrary. Arm Software, 2020

[6] M. Mingfei, mingfeima/convnet-benchmark-py, 2020

[7] ≪ TensorFlow ≫. https://www.tensorflow.org/?hl=fr

[8] ≪ PyTorch ≫. https://www.pytorch.org

[9] A. Ltd, ≪ AI and Machine Learning | Developer material ≫, Arm Developer. https://developer.arm.com/solutions/machine-learning-on-arm/developer-material

[10] R. Gandhi, ≪ R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms ≫, Medium, jul. 09, 2018. https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yoloobject-detection-algorithms-36d53571365e

[11] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, et Q. Tian, ≪ CenterNet: Keypoint Triplets for Object Detection ≫, ArXiv190408189 Cs, apr. 2019, http://arxiv.org/abs/1904.08189.

[12] P. Gnv, ≪ ImageNet Challenge: Advancement in deep learning and computer vision ≫, Medium, sept. 08, 2020. https://medium.com/@prudhvi.gnv/imagenet-challenge-advancementin-deep-learning-and-computer-vision-124fd33cb948

[13] S. Bhattacharyya, ≪ Understand and Implement ResNet-50 with TensorFlow 2.0 ≫, Medium, sept. 28, 2020. https://towardsdatascience.com/understand-and-implement-resnet-50-withtensorflow-2-0-1190b9b52691

[14] ≪ Slurm Workload Manager - Generic Resource (GRES) Scheduling ≫. https://slurm.schedmd.com/gres.html, dec. 16, 2020

[15] ≪ Computing cluster (SLURM) — PyTorch-Lightning 0.7.3 documentation ≫. https://pytorch-lightning.readthedocs.io/en/0.7.3/slurm.html dec. 16, 2020

[16] https://en.wikipedia.org/wiki/Backpropagation

[17] ≪ A Comprehensive guide to Fine-tuning Deep Learning Models in Keras (Part I) | Felix Yu ≫. https://flyyufelix.github.io/2016/10/03/fine-tuning-in-keras-part1.html dec. 16, 2020