ALPHA DATA

Deploying AI-based Spacecraft Telemetry Anomaly Detection on an Adaptive System-on-Chip Solution

EDHPC

13th-17th October 2025

Andrew McCormick
Alpha Data

Adewali Adetomi
Alpha Data

Ken O'Neill
AMD

alpha-data.com

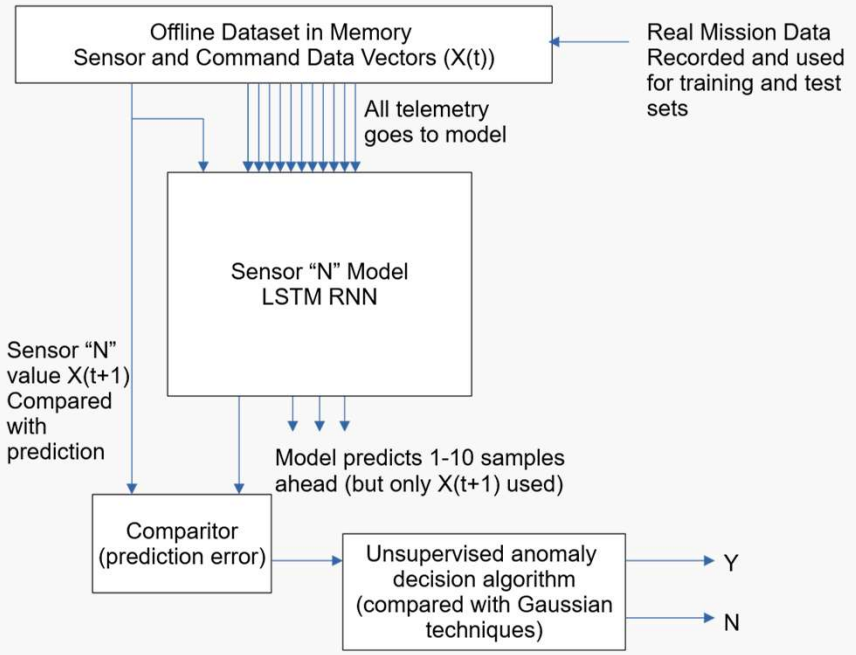
Overview

Capture and analysis of telemetry data from the tens, hundreds or even thousands of on-board sensors can give a good indication of satellite health and allow operators to act to mitigate the effect of imminent failure. Automating this process and operating in real-time will improve the operator's options and allow more timely action.

- Anomaly Detection and LSTM implementation
- Architecture of an Adaptive SoC for Space Flight Missions
- Memory Mapping of an AI Algorithm
- Simulation and Hardware Results
- Conclusions

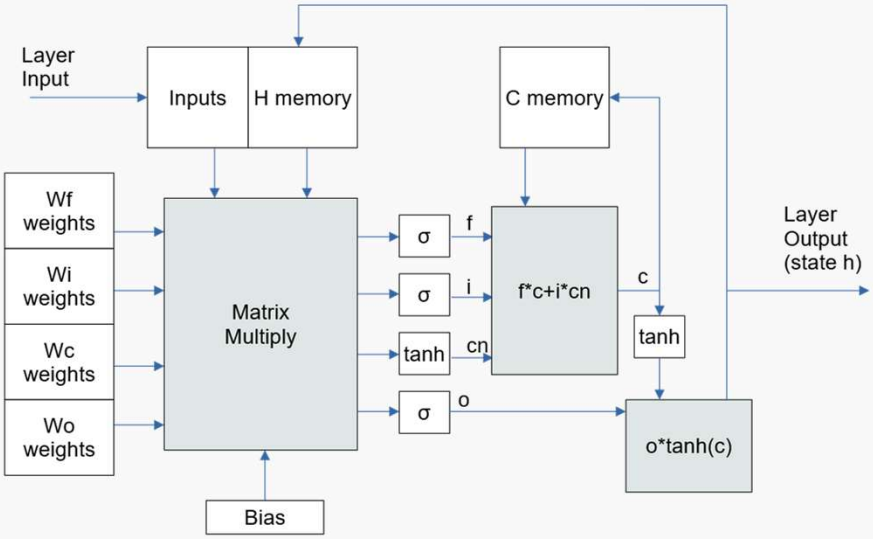
Anomaly Detection and LSTM Implementation

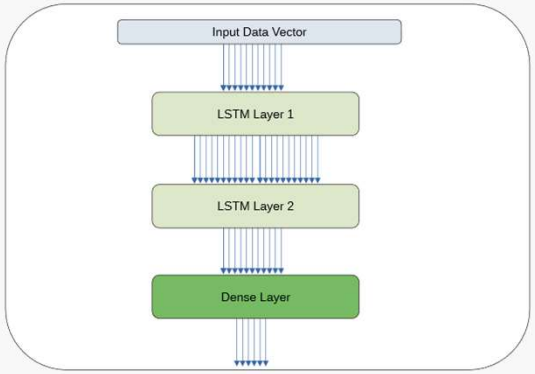
- Reference anomaly detection data comes from research based on real mission data
- Models are trained on normal data to predict future behaviour.
- Anomaly detected when model fails to track real sensor behaviour
- Avoids problem of limited training data for anomalous behaviour
- Anomaly can be detected with statistical comparison algorithm, or even a simple threshold.



Anomaly Detection and LSTM Implementation

- LSTM Model is built of 2 LSTM layers, shown here and a final dense output layer (just Matrix Multiply)
- LSTM Layer is recurrent with internal feedback of outputs and a forgetting factor.
- Core processing in layer is large matrix multiply, combining new inputs and current output state
- Other processing in layer is less than 1% of the computational load.

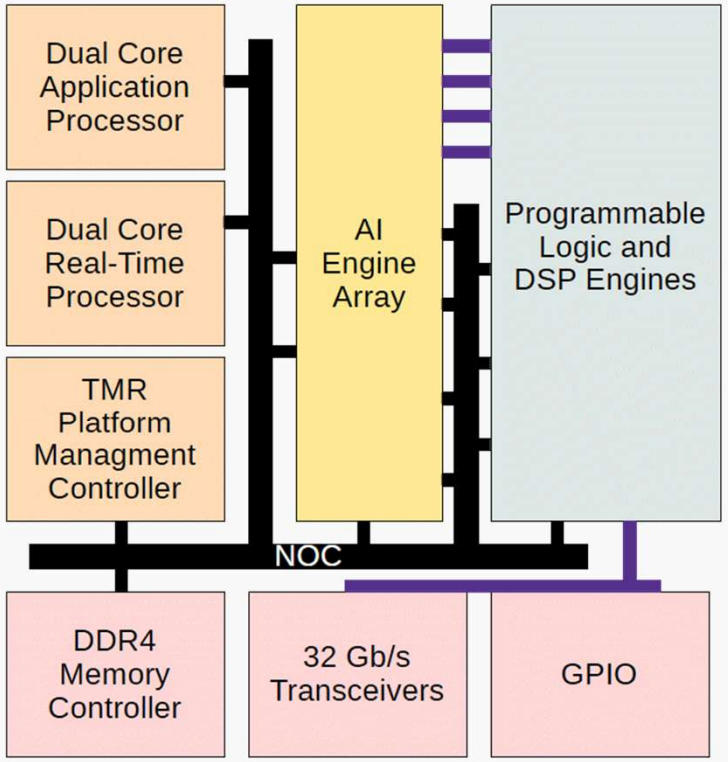




Architecture of an Adaptive SoC for Space Flight

Architecture of the Versal Edge Adaptive SoC

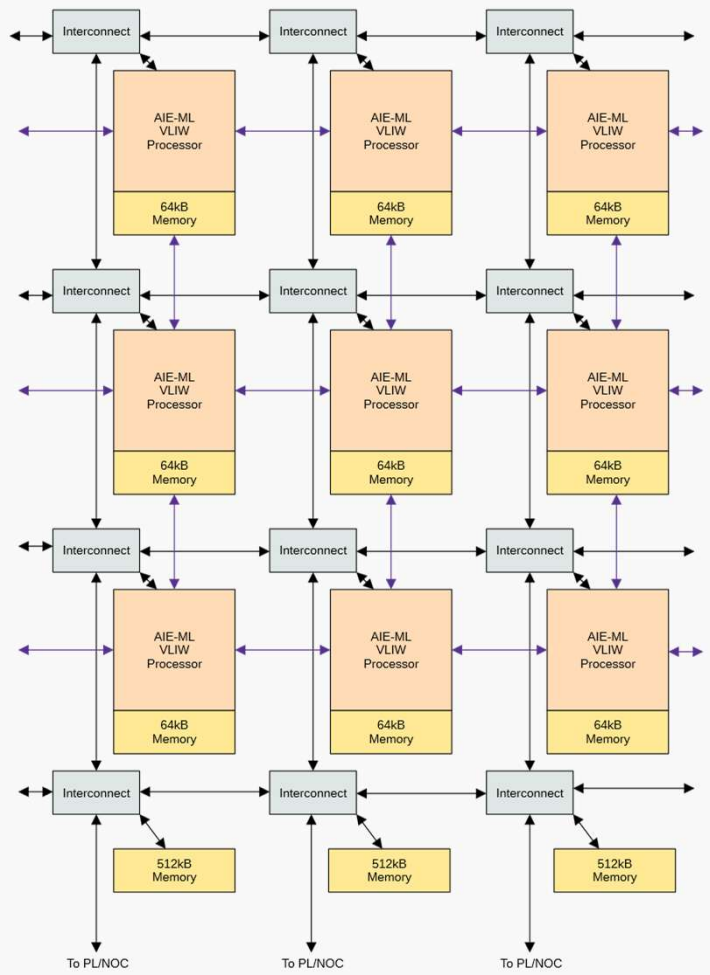
- Scalar Engines
  - Dual Core Application Processor for high end application
  - Dual Core Real-Time Processor for system level management
  - Triple-Mode Redundant microcontrollers for platform management, configuration control, system monitoring and even scrubbing.
- Adaptable Engines
  - Programmable Logic Array for user custom logic
- Intelligent Engines
  - AIE-ML Array for AI/ML and high performance signal processing applications
- Network on-chip
  - High speed packet switched backbone to connect up logic, IP, processors and memory
- High-performance IO
  - Hardened DDR4 Memory Controllers
  - Very high performance serial transceivers
  - Flexible General Purpose IO



Architecture of an Adaptive SoC for Space Flight

Architecture of the Versal AIE-ML Array

- The AIE-ML is an array of VLIW processing Tile
- Each Tile can achieve 73.6 GMACs performance with the BFLOAT16 data type
- Each Tile has a local 64kB memory store
- Each tile can access the memory of 3 neighbouring tiles a full rate
- Interconnect blocks are used to move data further around the array.
- Interconnect streams can transfer data to/from the PL or the NoC into and out of AIE-ML Tiles
- Interconnect streams can transfer data between Tiles that are not neighbours
- Interconnect can transfer data between Shared Memory 512kB blocks and AIE-ML tiles
- Interconnect can transfer data between Shared Memory 512kB blocks and the PL or NoC interfaces.



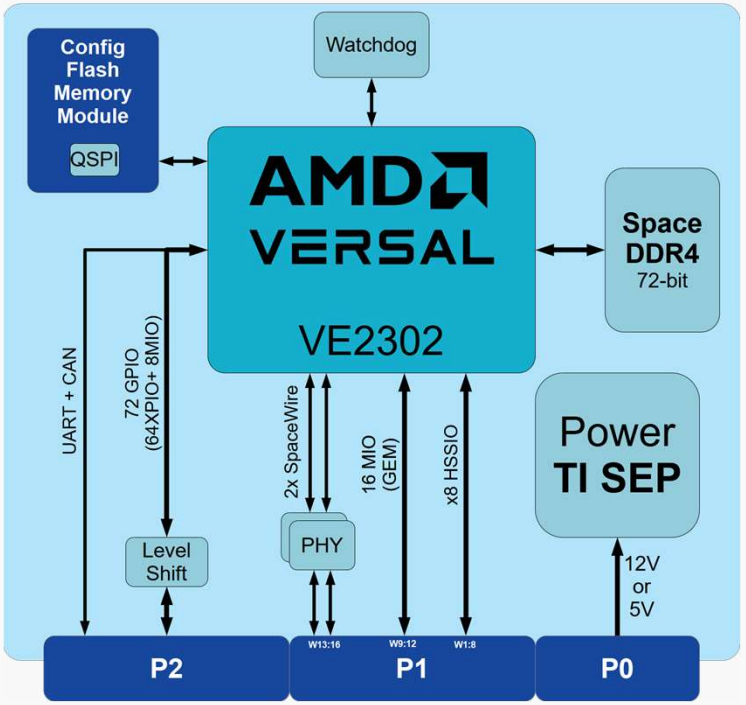
Architecture of an Adaptive SoC for Space Flight

Architecture of a 3U Space VPX reconfigurable flight computer

The ADM-VB630 provides a development and reference platform for designs that aim to use the XQR Grade AMD Versal Edge VE2302 for Space.

One key feature of the board relevant to this presentation is the 8GB Space DDR4 memory bank that will be required to hold off-chip model data and other application code and data.

Data input from sensors to the processor can be achieved via GPIO, HSSIO, SpaceWire or CAN bus interfaces running across the VPX backplane from the various sensors across the platform.



Architecture of an Adaptive SoC for Space Flight

Mapping the application onto the hardware

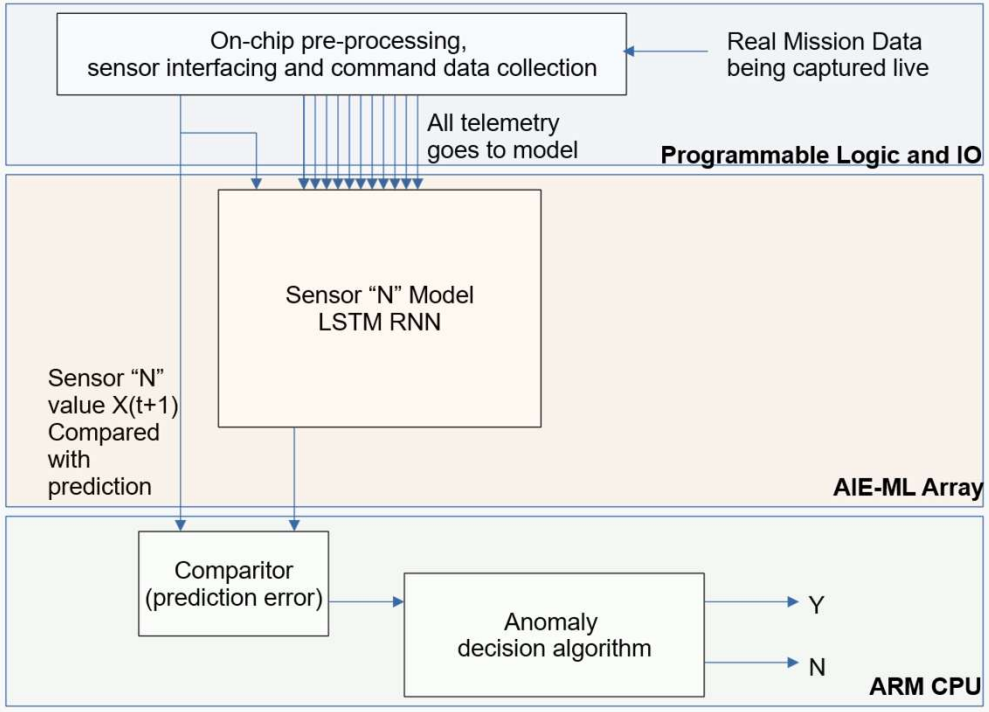
The Anomaly Detection Application is a good fit for the Versal Architecture

The programmable logic and IO can capture and pre-process the sensor data coming in from the backplane connections and pass it to the model in real time.

This can be passed to the AIE-ML array to run the LSTM RNN Models of the sensors.

The output predictions can be compared with the telemetry data in the ARM sub-system, and an anomaly diagnosis made.

The processing board can store its model weights and processing code in the off-chip DDR4 memory and load these into on-chip memory caches as required.



Memory mapping of an AI Algorithm

AI algorithms are typically memory bound, especially if a low latency response is required

- Basic implementation stores all Model Weights in 64kB AIE-ML attached memory
- Requires 2 AIE-ML blocks per LSTM layer
- Single model fits into 5 AIE-ML tiles
- Can increase performance through parallelization – 6 units will use most tiles and 88% peak performance
- Can increase peak performance and lower latency by parallelizing model further.
- Peak performance up to 97% of available resource
- But very limited number of models

Resource	VE2302	Single Model	6 x Par Models	Max model	3x Max model
AIE-ML Tiles	34	5	30	11	33
AIE-ML Memory	2.1MB	169kB	1.0MB	169kB	169kB
Shared Memory Buffers (512kB)	17	0	0	0	0
Shared Memory Bits	8.7MB	0	0	0	0
GMIO Inputs	4x6	5	(30)	(11)	(33)
PLIO Inputs	6x12	1	6	1	3
PLIO Outputs	8x28	1	6	1	3
Peak GMACs (BFLOAT16)	2502	368	2208	810	2430
Latency (µs)		0.6	0.6	0.27	0.27
# Models		1	6	1	3

Memory mapping of an AI Algorithm

The number of models that can run concurrently is limited by memory

- External DDR4 on the board can support a very high number of models
- Performance will be limited by the memory bandwidth of 21GB/s
- Result is that increasing parallelization does not improve throughput and will increase latency
- Sequential scaling up using one thread but switching between many models will achieve the same rate, but will also increase latency in proportion to the number of concurrent models.
- Can use less than 1% of the Array performance.

Resource	VE2302	AIE-ML Memory	DDR4	Par DDR4
AIE-ML Tiles	34	5	3	33
AIE-ML Memory	2.1MB	169kB	1.6kB	35kB
Shared Memory Buffers (512kB)	17	0	0	0
Shared Memory Bits	8.7MB	0	0	0
GMIO Inputs	4x6	5	3	(33)
PLIO Inputs	6x12	1	1	11
PLIO Outputs	8x28	1	1	11
Peak GMACs (BFLOAT16)	2502	368	10.5	10.5
Latency (µs)		0.6	8.1	89.1
# Models		1	1	11

Memory mapping of an AI Algorithm

Shared Memory Tiles within the AIE-ML Array Can provide a Cache

- AIE-ML Array has a number of 512kB Shared Memory Buffers (17 on VE2302)
- These can provide a Level 2 Cache between the DDR4 and the Tile Processing
- Allow 30GB/s of data to be processed per Shared Memory Tile (15 BFLOAT16 GMACs per Tile)
- This allows multiple models to be mapped into a single tile, to allow a TDM split of the processing
- This can also be parallelized within the limit of the number of memory cores available and the number of AIE-ML Tiles
- Allows a practical number of models to run (40) with a reasonable device utilization (10% of peak performance)

Resource	AIE-ML Memory	SharedMem	TDM	Par TDM
AIE-ML Tiles	5	3	3	24
AIE-ML Memory	169kB	1.6kB	1.6kB	13kB
Shared Memory Buffers (512kB)	0	2	2	16
Shared Memory Bits	0	168kB	840kB	6.7MB
GMIO Inputs	5	3	3	(24)
PLIO Inputs	1	1	1	8
PLIO Outputs	1	1	1	8
Peak GMACs (BFLOAT16)	368	30	30	240
Latency (µs)	0.6	2.8	14	14
# Models	1	1	5	40

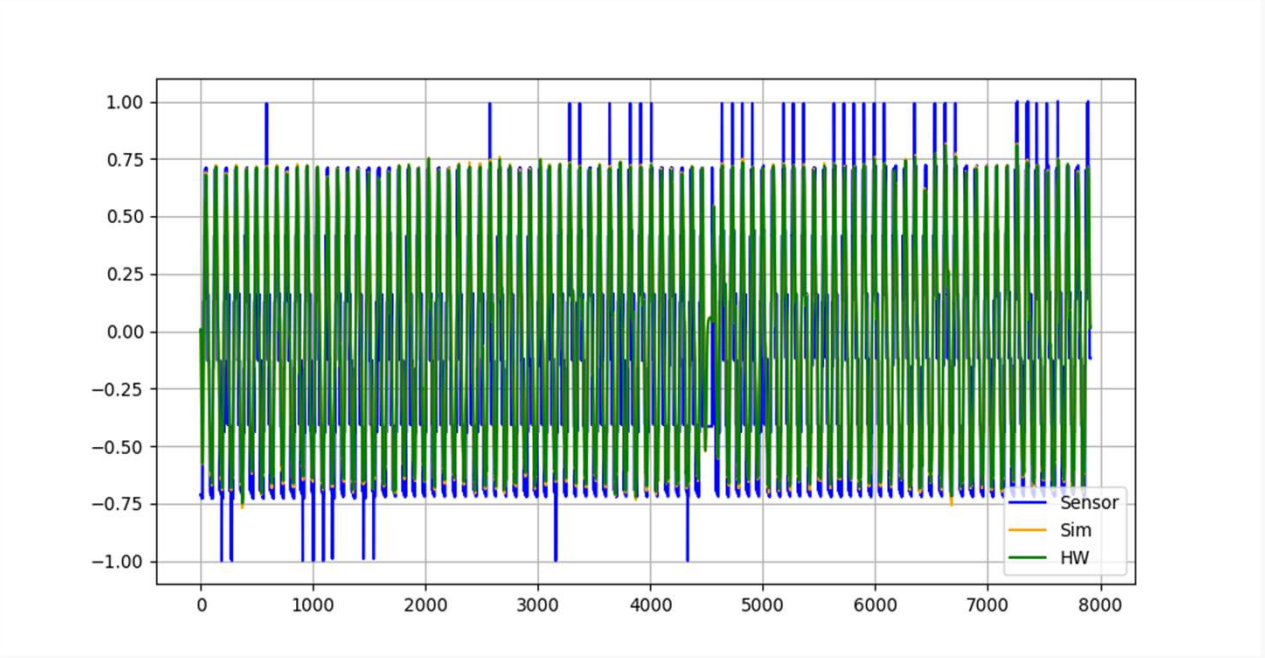
Memory mapping of an AI Algorithm

Are our models the correct size?

- The choice of model size came from reference paper and popular models at the time
- Reference paper suggested smaller models might be sufficient
- A simple pruning approach was used to reduce the models from 80 neuron layers to 32 neuron layers. With the output layer reduced to 1 neuron
- Evaluation of results showed numerical differences in the prediction results, but not enough to change the anomaly detection result
- Pruning reduces the memory footprint significantly allowing much higher throughput, or potentially a higher number of models to be supported.
- Additional Data Science and Training effort is highly recommended to find optimal model sizes

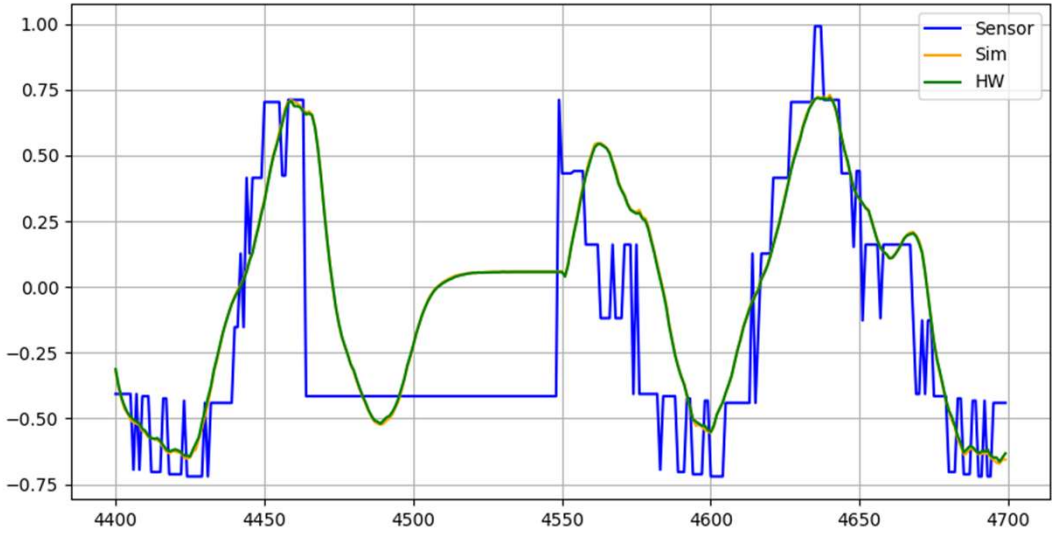
Resource	VE2302	Ref Model	Pruned	Par Pruned
AIE-ML Tiles	34	5	2	32
AIE-ML Memory	2.1MB	169kB	24kB	384kB
Shared Memory Buffers (512kB)	17	0	0	0
Shared Memory Bits	8.7MB	0	0	0
GMIO Inputs	4x6	5	2	(32)
PLIO Inputs	6x12	1	1	16
PLIO Outputs	8x28	1	1	16
Peak GMACs (BFLOAT16)	2502	368	110	1760
Latency (µs)		0.6	0.17	0.85
# Models		1	1	80

Hardware Results



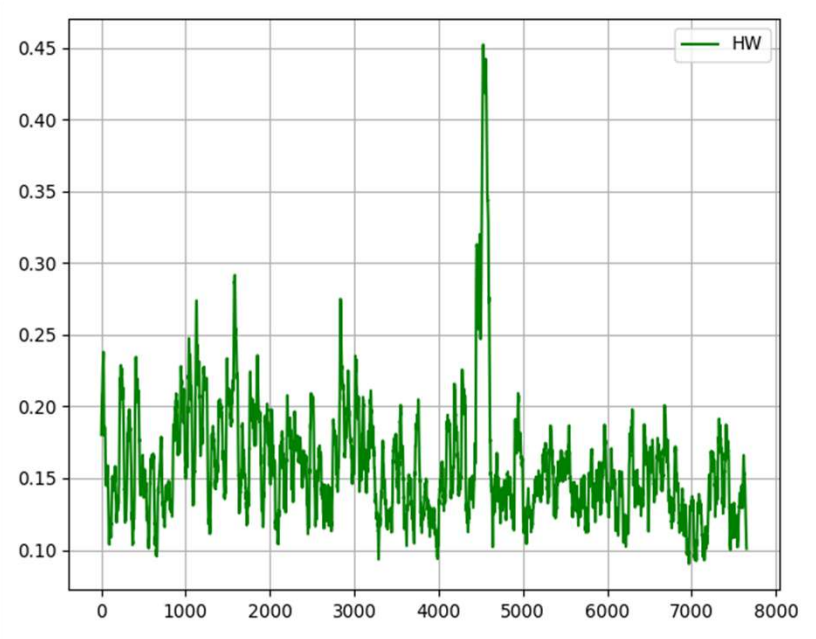
- Designs have been ported to the AMD VE2302 device running on the ADM-VB630 Platform
- A single sensor channel is plotted above comparing the simulation, the hardware run and the actual sensor values.
- The anomaly in the data can be seen around 4500 samples in.

Hardware Results



- Zooming in shows that the models can track the sensor signal under normal conditions
- But diverge when the anomaly occurs

Hardware Results



- Plotting the low pass filtered mean square difference between the model prediction and the actual next sample shows a clear peak when the anomaly occurs.

CONCLUSIONS

01

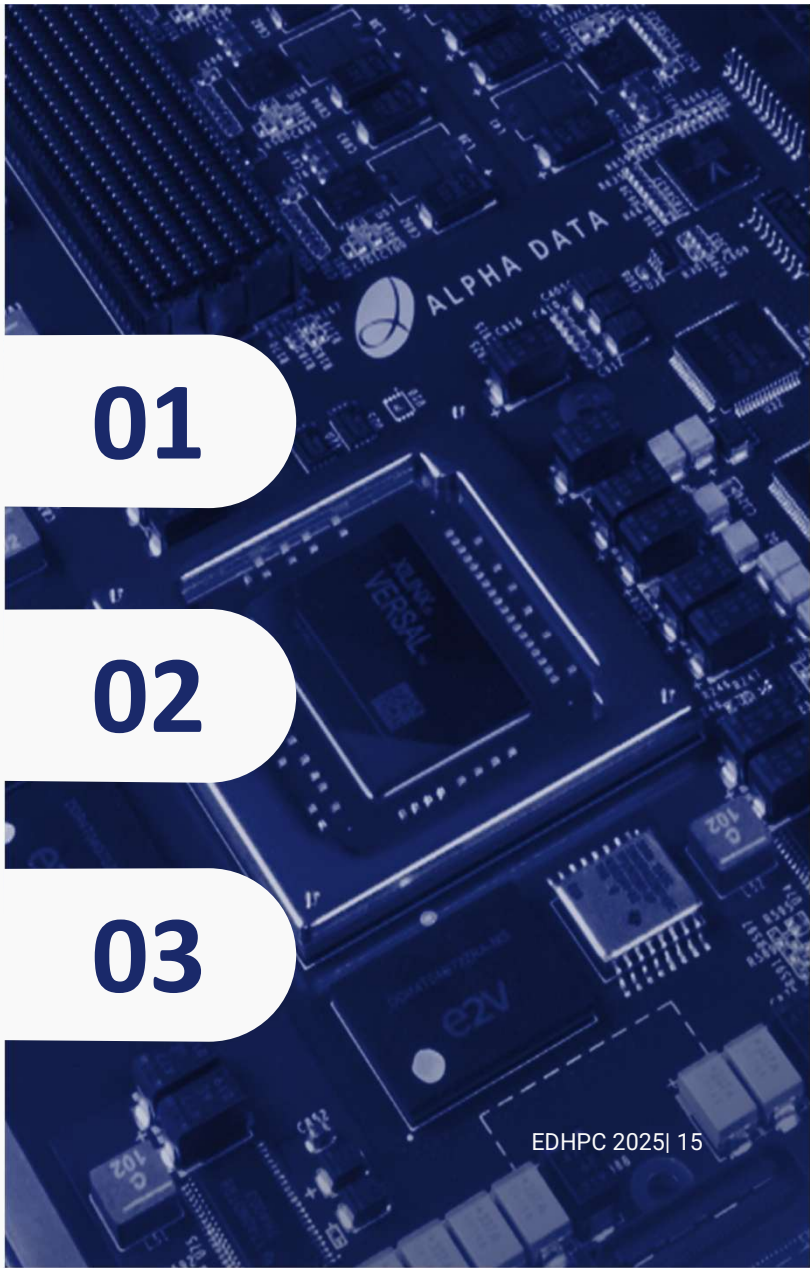
Adaptive SoC devices, such as the AMD Versal AI Edge parts provide a practical platform for deploying Machine Learning algorithms, such as LSTM-RNN models, useful in Satellite Sensor Anomaly Detection

02

The memory footprint of the models is critical in low-latency real time operation. This impacts performance with practical implementations requiring model parameter storage in level 1 or level 2 cache, and not off-chip

03

The Versal AI Edge device's Shared Memory provides a practical trade off supporting an acceptable number of models at a capable rate. However, thoroughly assessing model size and looking at options such as pruning may reveal much higher performance solutions.



ALPHA DATA