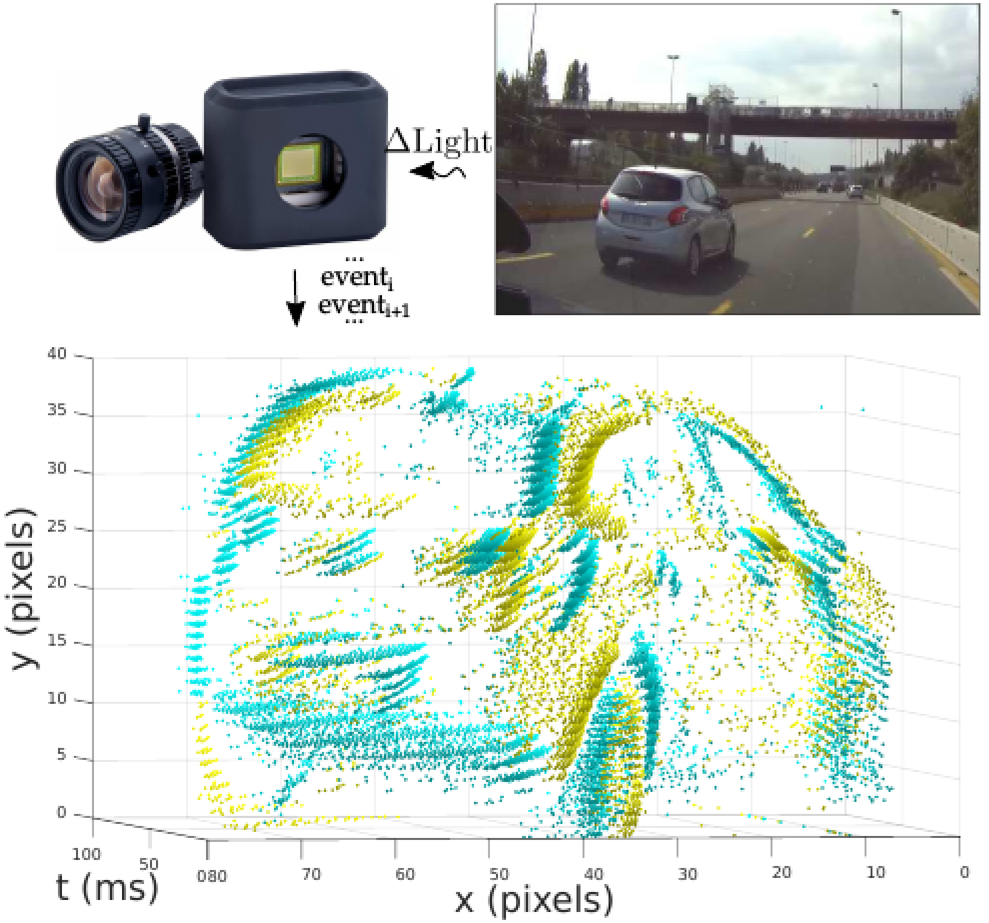
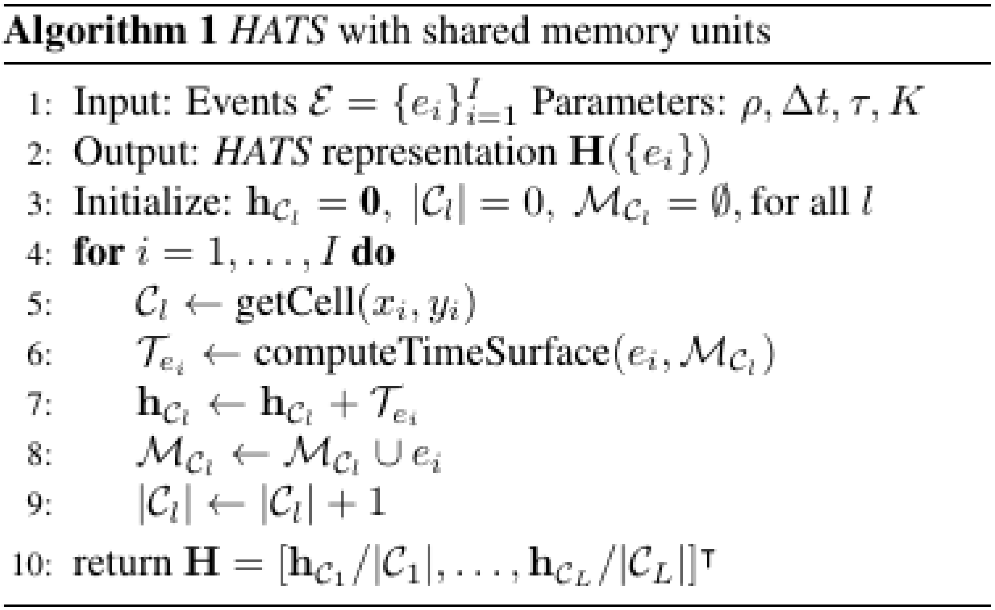
HATS: Histograms of Averaged Time Surfaces for Robust Event-based Object Classification



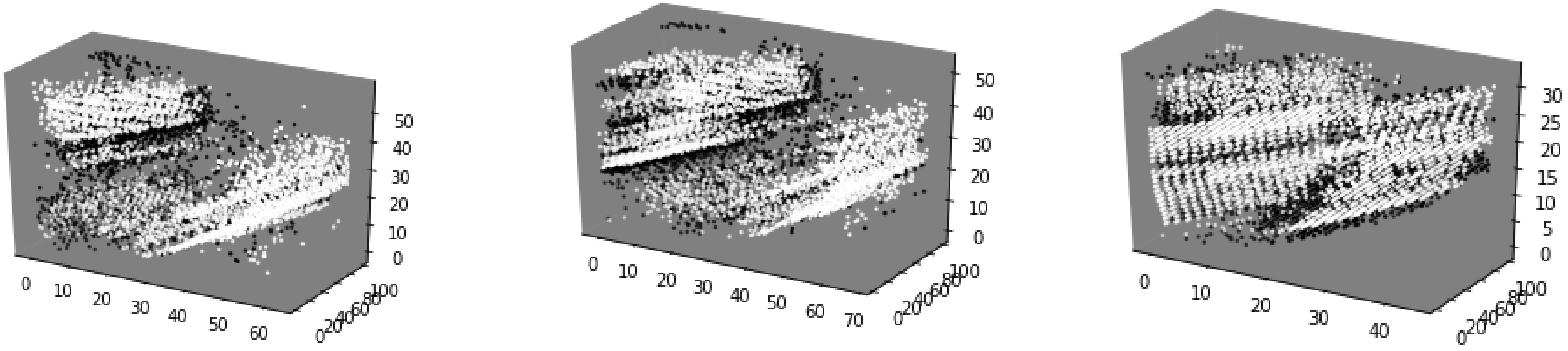
Software Implementation

- The HATS paper introduces new concept of Local Memory Time Surfaces and there is no implementation available. It makes a histogram from these time surfaces and then further uses a SVM classifier to classify the images.
- 

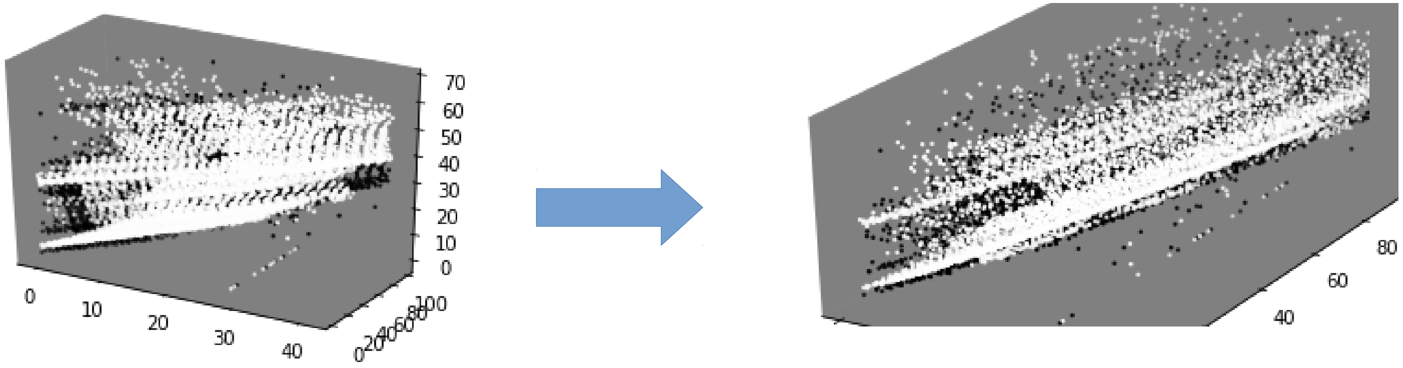
Algorithm 1 *HATS* with shared memory units

```
Input: Events $\mathcal{E} = \{e_i\}_{i=1}^{I}$ Parameters: $\rho, \Delta t, \tau, K$
Output: HATS representation $\mathbf{H}(\{e_i\})$
Initialize: $\mathrm{h}_{\mathcal{C}_l} = \mathbf{0},\ |\mathcal{C}_l| = 0,\ \mathcal{M}_{\mathcal{C}_l} = \emptyset$, for all $l$
for $i = 1, \ldots, I$ do
    $\mathcal{C}_l \leftarrow \mathrm{getCell}(x_i, y_i)$
    $\mathcal{T}_{e_i} \leftarrow \mathrm{computeTimeSurface}(e_i, \mathcal{M}_{\mathcal{C}_l})$
    $\mathrm{h}_{\mathcal{C}_l} \leftarrow \mathrm{h}_{\mathcal{C}_l} + \mathcal{T}_{e_i}$
    $\mathcal{M}_{\mathcal{C}_l} \leftarrow \mathcal{M}_{\mathcal{C}_l} \cup e_i$
    $|\mathcal{C}_l| \leftarrow |\mathcal{C}_l| + 1$
return $\mathbf{H} = [\mathrm{h}_{\mathcal{C}_1}/|\mathcal{C}_1|, \ldots, \mathrm{h}_{\mathcal{C}_L}/|\mathcal{C}_L|]^\intercal$
```

N-CARS Dataset



Real Worl Challenging images from the N-Cars Dataset. (Shown for Events>5000).
The below image shows the occlusion produced by fast-moving cars.



Expanded in Time

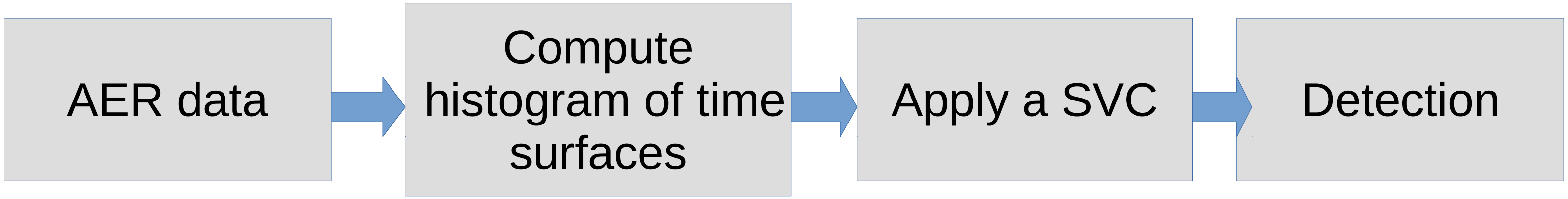
N-CARS Dataset

- Real Dataset for CARS detection acquired from ATIS camera.
- Different poses and illumination of Cars with real -world noise.
- 100ms of video for each sample.
- 7940 car training samples and 7482 bckg samples.
- 4396 car test samples and 4291 bckg test samples

SoC design Flow (Zedboard)

- Vivado HLS C code+testing, synthesis and analysis, optimisation for Hats IP production
- Vivado Design – block design, synthesis and implementation
- Linux SDK for ARM processor

HATS hardware algorithm for inference



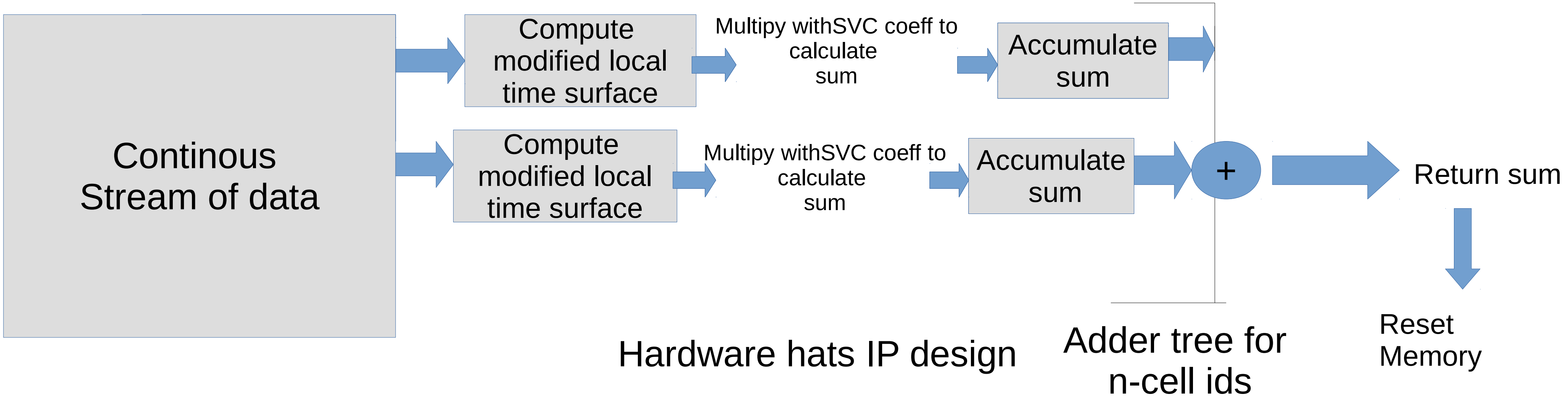
AER data

Compute histogram of time surfaces

Apply a SVC

Detection

- For each cell a histogram is calculated.
- Computation of hisogram requires memory cell for storing past events.
- Memory extensive procedure for storing histogram of time surface



Continous Stream of data

Compute modified local time surface

Multipy withSVC coeff to calculate sum

Accumulate sum

Compute modified local time surface

Multipy withSVC coeff to calculate sum

Accumulate sum

+

Return sum

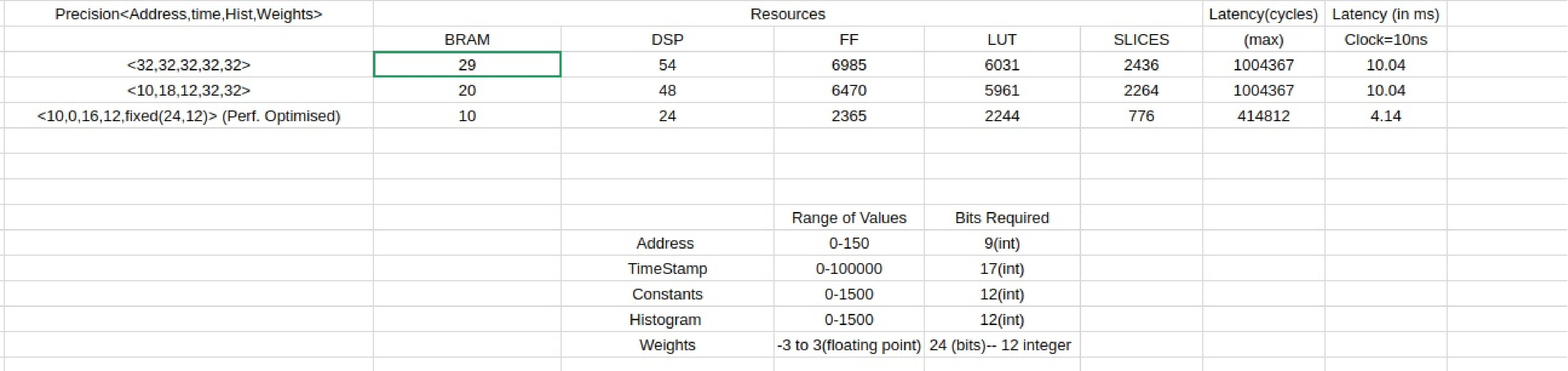
Reset Memory

Hardware hats IP design

Adder tree for n-cell ids

FPGA Resource Utilisation

HATS IP (Average resource utilisation/ip)



Precision<Address,time,Hist,Weights>	Resources					Latency(cycles)	Latency (in ms)
	BRAM	DSP	FF	LUT	SLICES	(max)	Clock=10ns
<32,32,32,32,32>	29	54	6985	6031	2436	1004367	10.04
<10,18,12,32,32>	20	48	6470	5961	2264	1004367	10.04
<10,0,16,12,fixed(24,12)> (Perf. Optimised)	10	24	2365	2244	776	414812	4.14

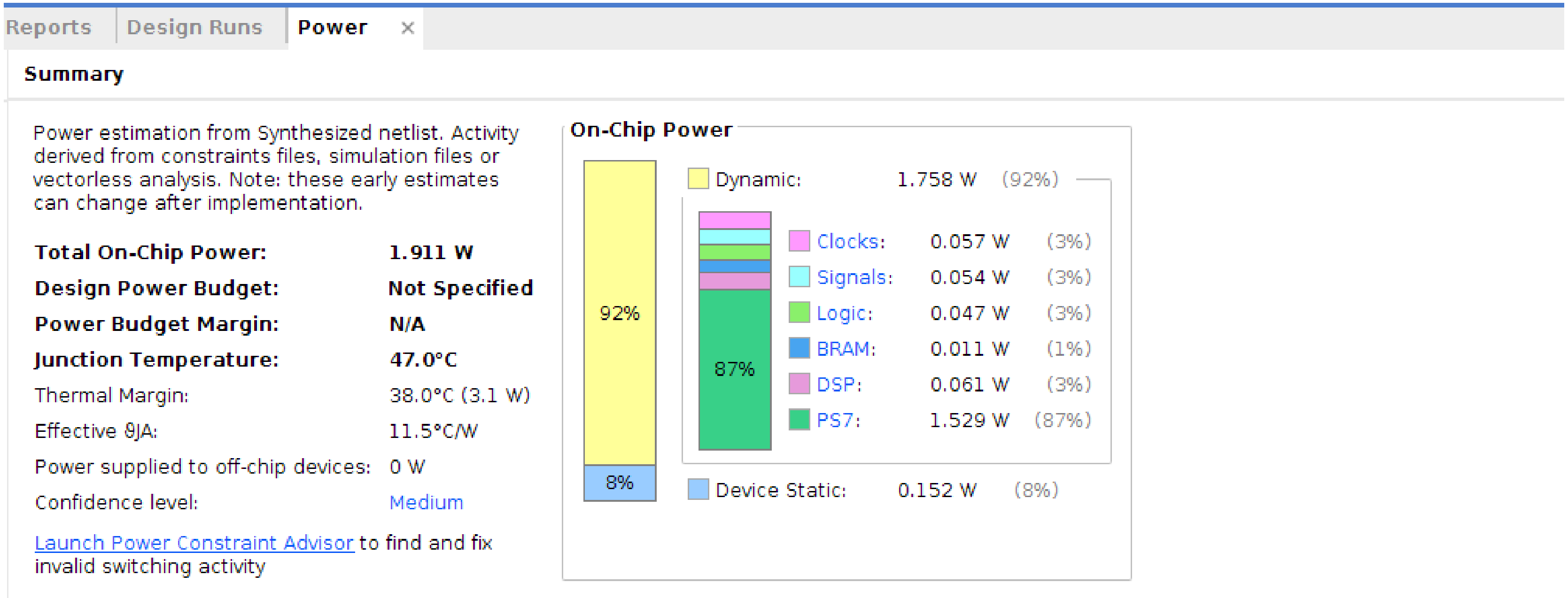
	Range of Values	Bits Required
Address	0-150	9(int)
TimeStamp	0-100000	17(int)
Constants	0-1500	12(int)
Histogram	0-1500	12(int)
Weights	-3 to 3(floating point)	24 (bits)-- 12 integer

8 Hats Accelerators used with different SVM weights

Performance Comparison

Hardware Latency : 4.14ms/event
Inference Power Estimation : 1.911 W (Logic Power – 0.382 W)
Avg. Throughput : 750*8/4.14 Kevnts/s or 1.45 Mevnts/s

Software Parallelized Latency (in Python) : 187 ms/event



Summary

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

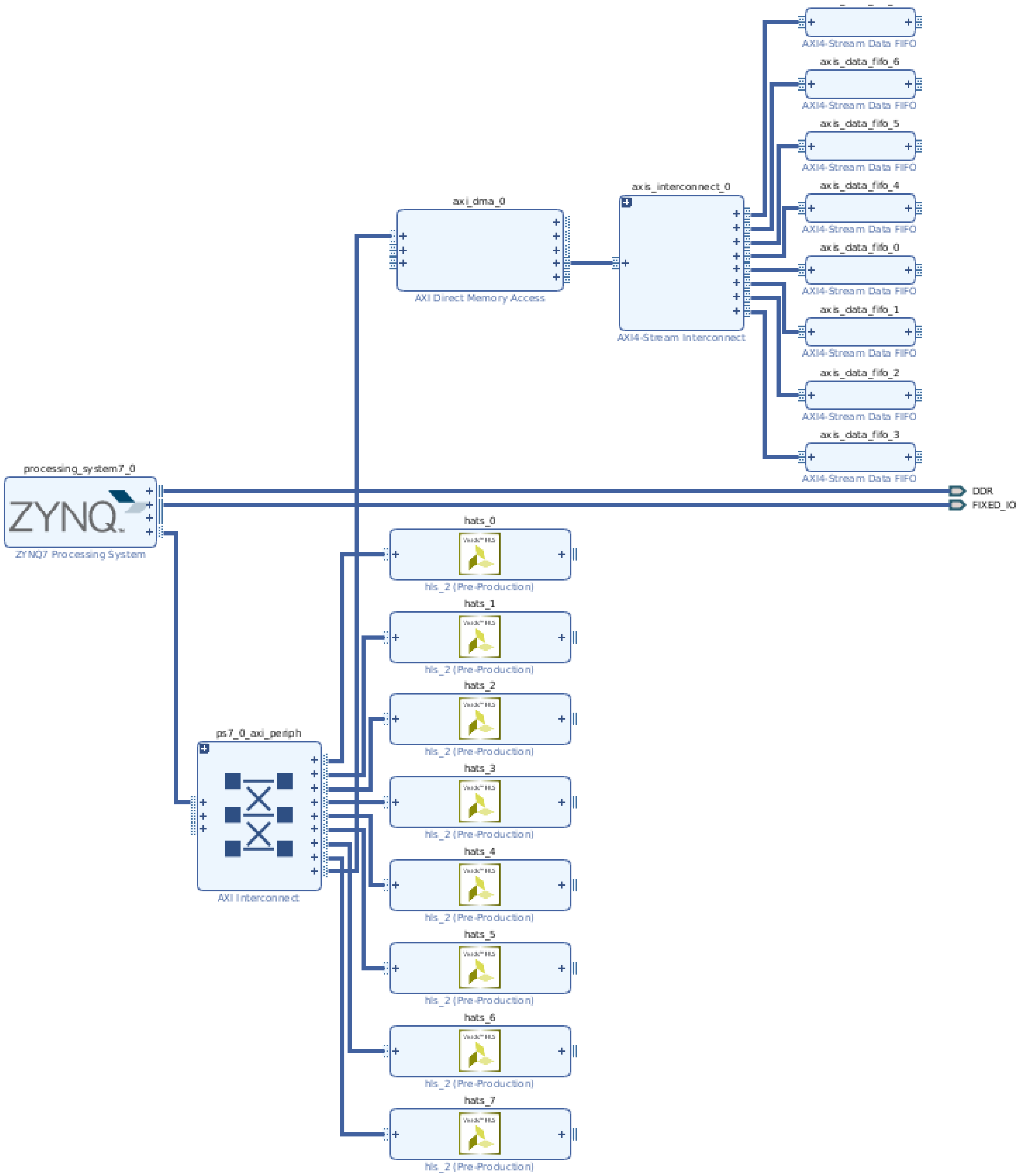
Total On-Chip Power:	**1.911 W**
Design Power Budget:	**Not Specified**
Power Budget Margin:	**N/A**
Junction Temperature:	**47.0°C**
Thermal Margin:	38.0°C (3.1 W)
Effective θJA:	11.5°C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Medium

Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power

🟨 Dynamic:	1.758 W	(92%)
🟪 Clocks:	0.057 W	(3%)
🟦 Signals:	0.054 W	(3%)
🟩 Logic:	0.047 W	(3%)
🟦 BRAM:	0.011 W	(1%)
🟪 DSP:	0.061 W	(3%)
🟩 PS7:	1.529 W	(87%)
🟦 Device Static:	0.152 W	(8%)

Block Design


AXI4-Stream Data FIFO
axis_data_fifo_6
AXI4-Stream Data FIFO
axis_data_fifo_5
AXI4-Stream Data FIFO
axis_data_fifo_4
AXI4-Stream Data FIFO
axis_data_fifo_0
AXI4-Stream Data FIFO
axis_data_fifo_1
AXI4-Stream Data FIFO
axis_data_fifo_2
AXI4-Stream Data FIFO
axis_data_fifo_3
AXI4-Stream Data FIFO
axis_interconnect_0
AXI4-Stream Interconnect
axi_dma_0
AXI Direct Memory Access
processing_system7_0
ZYNQ
ZYNQ7 Processing System
DDR
FIXED_IO
ps7_0_axi_periph
AXI Interconnect
hats_0
hls_2 (Pre-Production)
hats_1
hls_2 (Pre-Production)
hats_2
hls_2 (Pre-Production)
hats_3
hls_2 (Pre-Production)
hats_4
hls_2 (Pre-Production)
hats_5
hls_2 (Pre-Production)
hats_6
hls_2 (Pre-Production)
hats_7
hls_2 (Pre-Production)