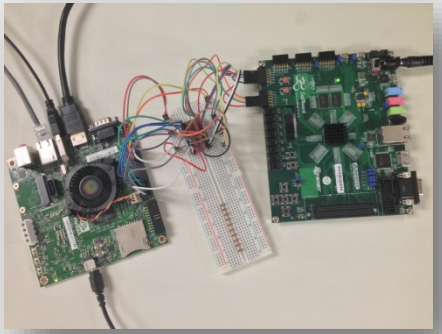
GPU and FPGA Based Architecture Design for Real-time Signal Classification

THE UNIVERSITY OF ARIZONA



Nilangshu Bidyanta, Ali Akoglu, Garrett Vanhoy,

Mohammed A. Hirzallah, Tamal Bose

(Department of Electrical and Computer Engineering)

Bo Ryu

(EpiSci, Inc.)

Modulation Classification

- Spectral Correlation Density (SCD)
  - Signals that can be classified: BPSK, QPSK, 8PSK, 16QAM, GMSK, CPFSK, AM, FM, OFDM.
  - Relatively immune to noise
- Challenge:
  - Computational complexity

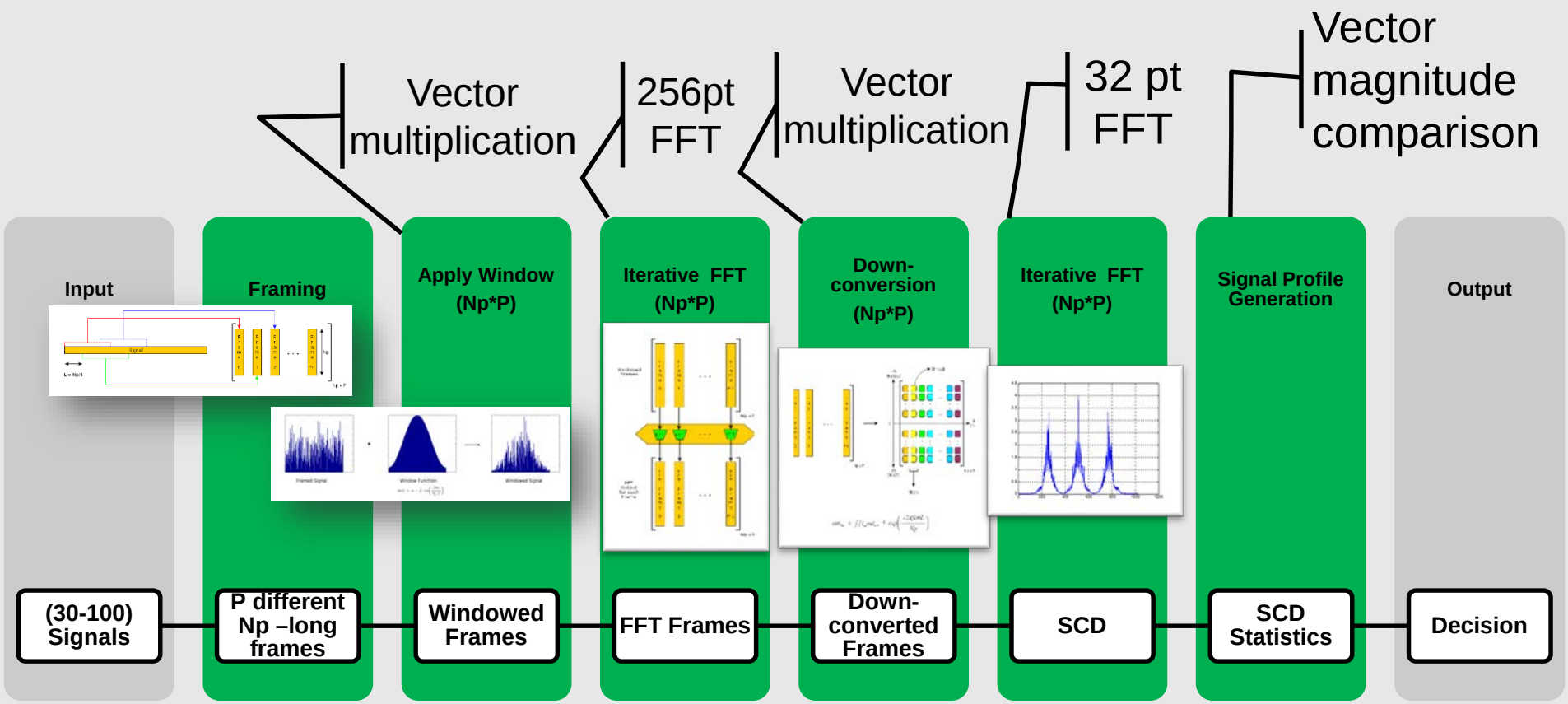
Goals

- Design a parallel computing platform for SCD analysis based on GPU and FPGA compute engines

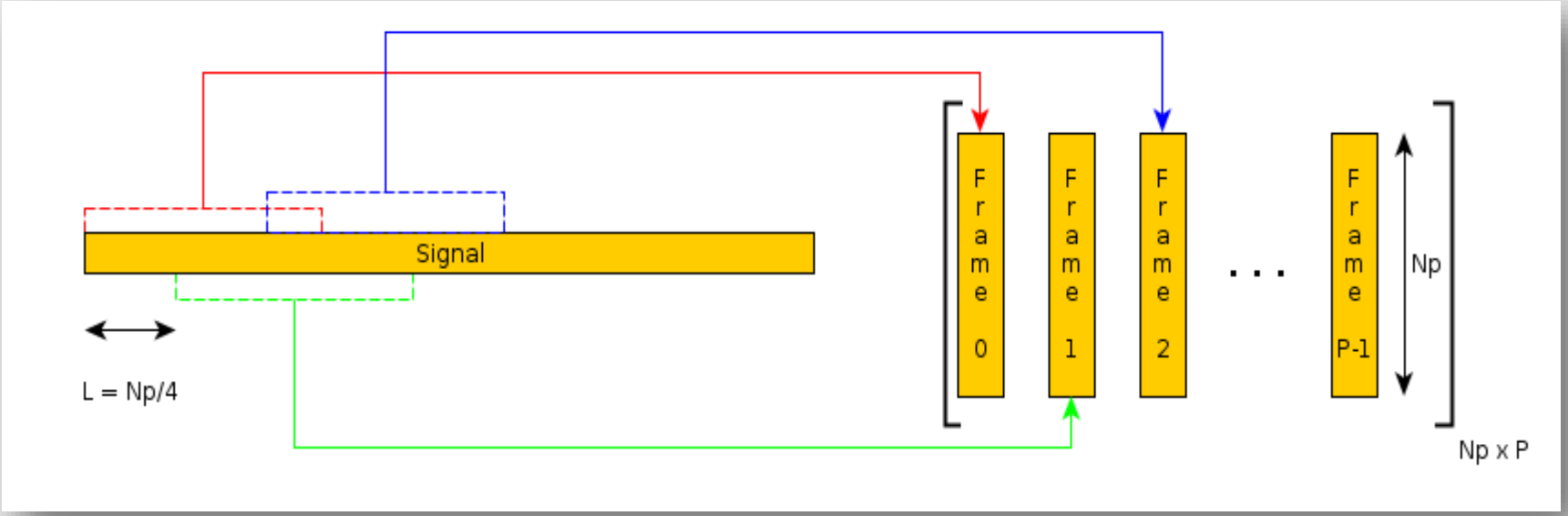
- Demonstrate the feasibility of such an architecture
  - Classification throughput
  - Power consumption

Process Chain

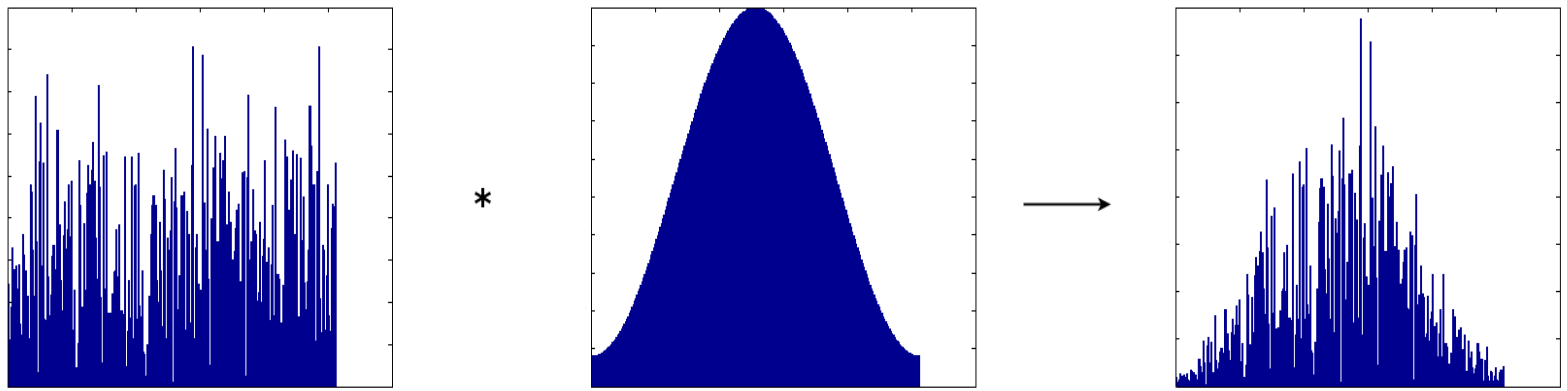
THE UNIVERSITY OF ARIZONA®



- Step-1: Framing the signal
  - signal split into P parts (32) of length Np (256)
    - each part overlaps with its predecessor
    - offset between two consecutive parts L = Np/4
  - Frames arranged column-wise - Np x P matrix

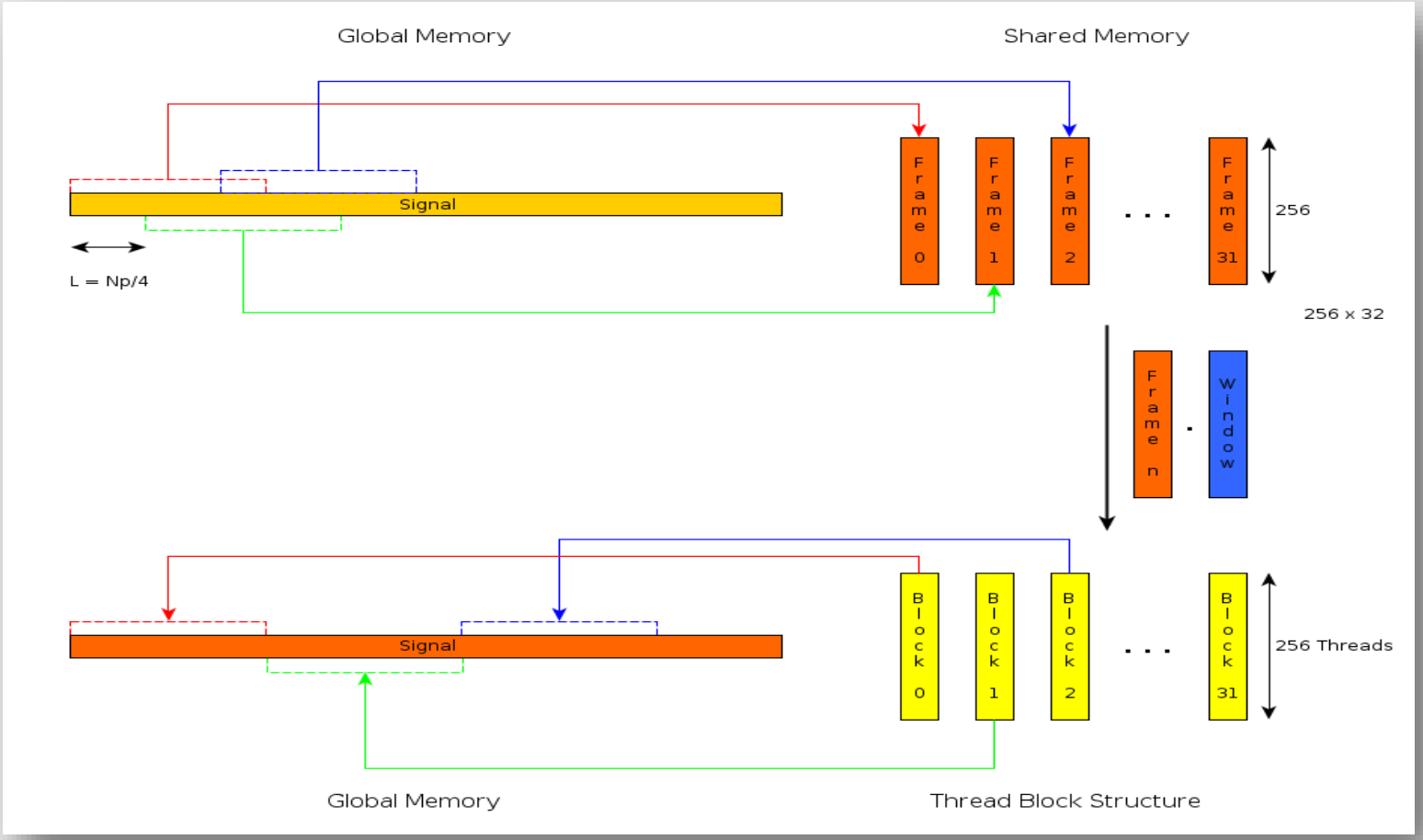


- Step-2: Hamming Window
  - steep cut-offs at both ends of the frame (step-1) introduce high frequency components
  - "raised cosine" function of length Np

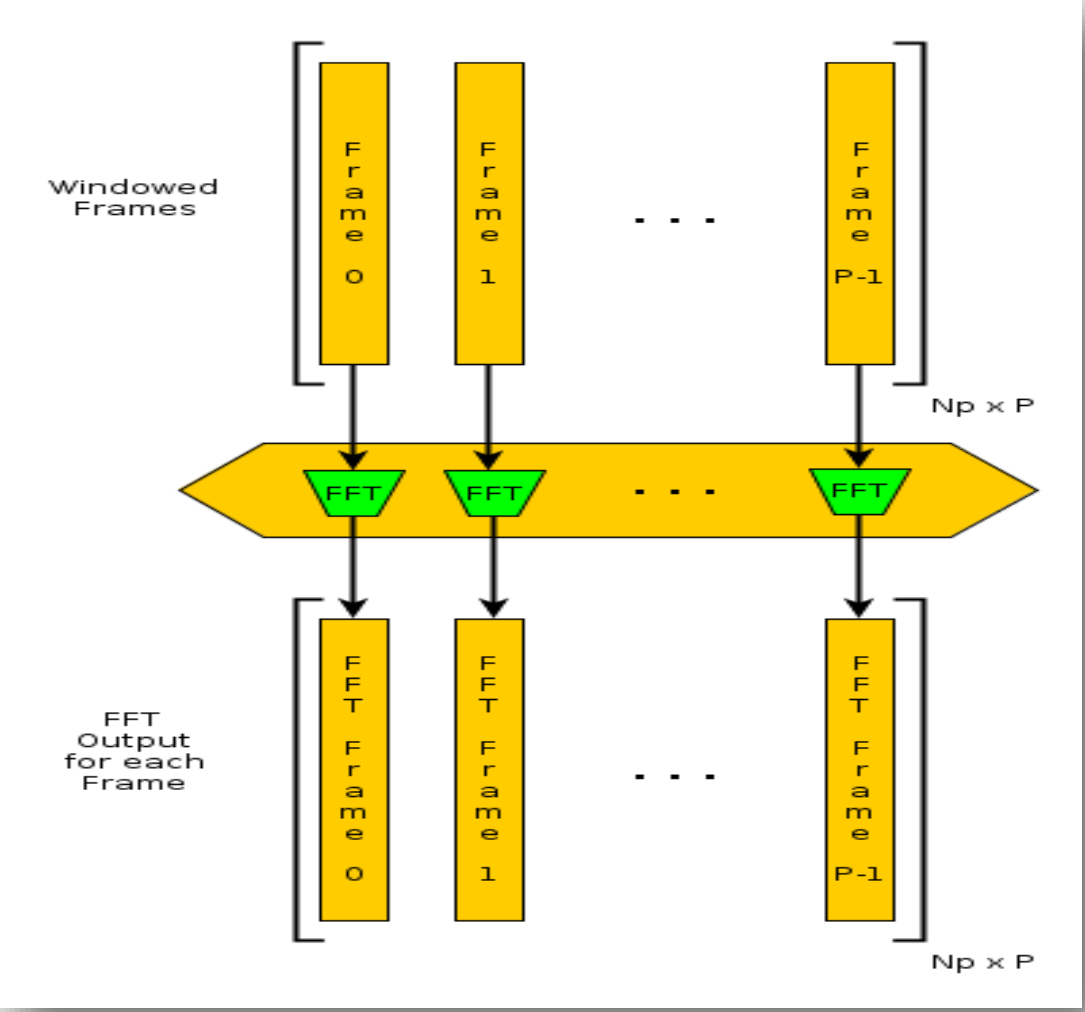


$$\omega(n) = \alpha - \beta \cos\left(\frac{2\pi n}{N_p - 1}\right)$$

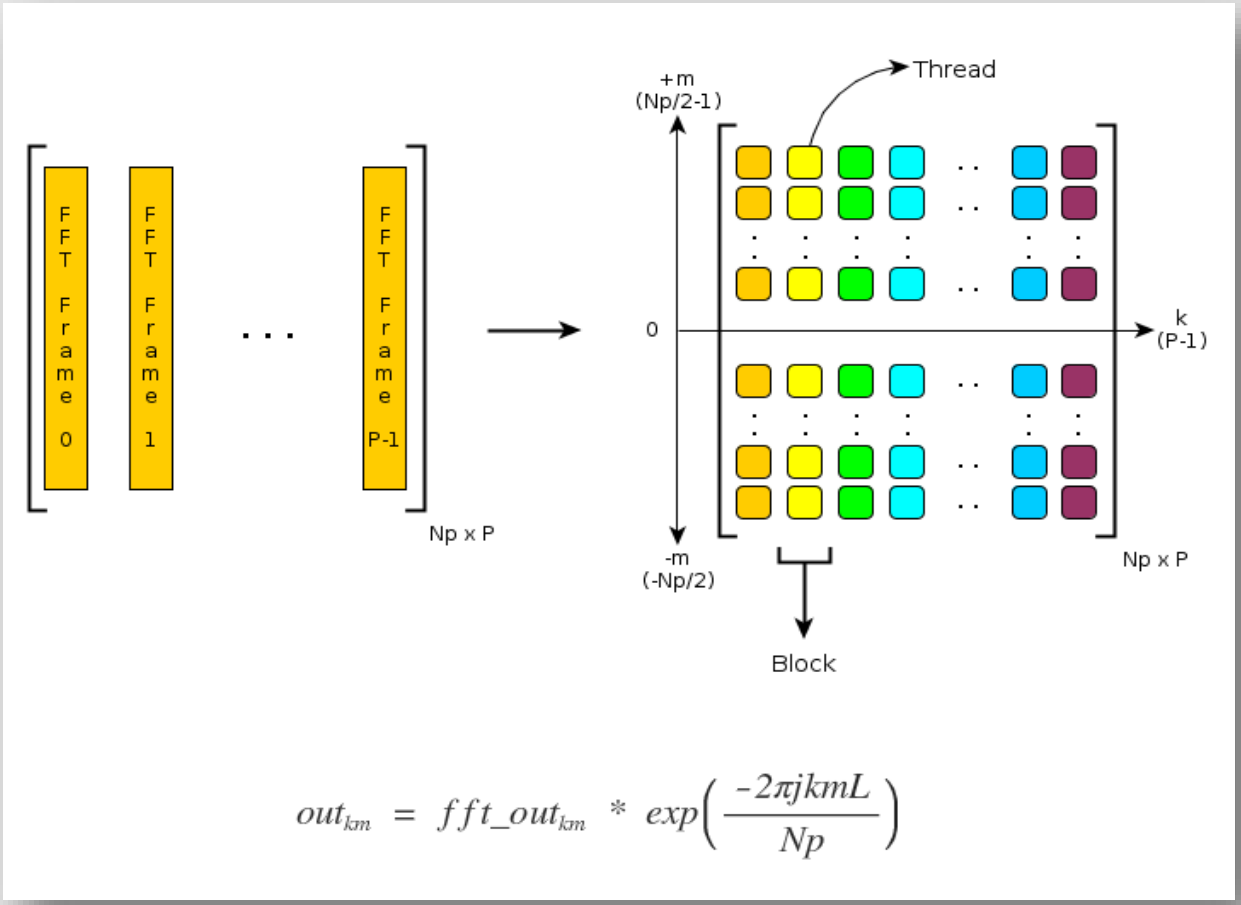
- Step-2: Hamming Window
- Framing and Windowing (Kernel 1)



- Step-3: Iterative FFT (Kernel 2)
  - FFT applied in parallel to windowed frames



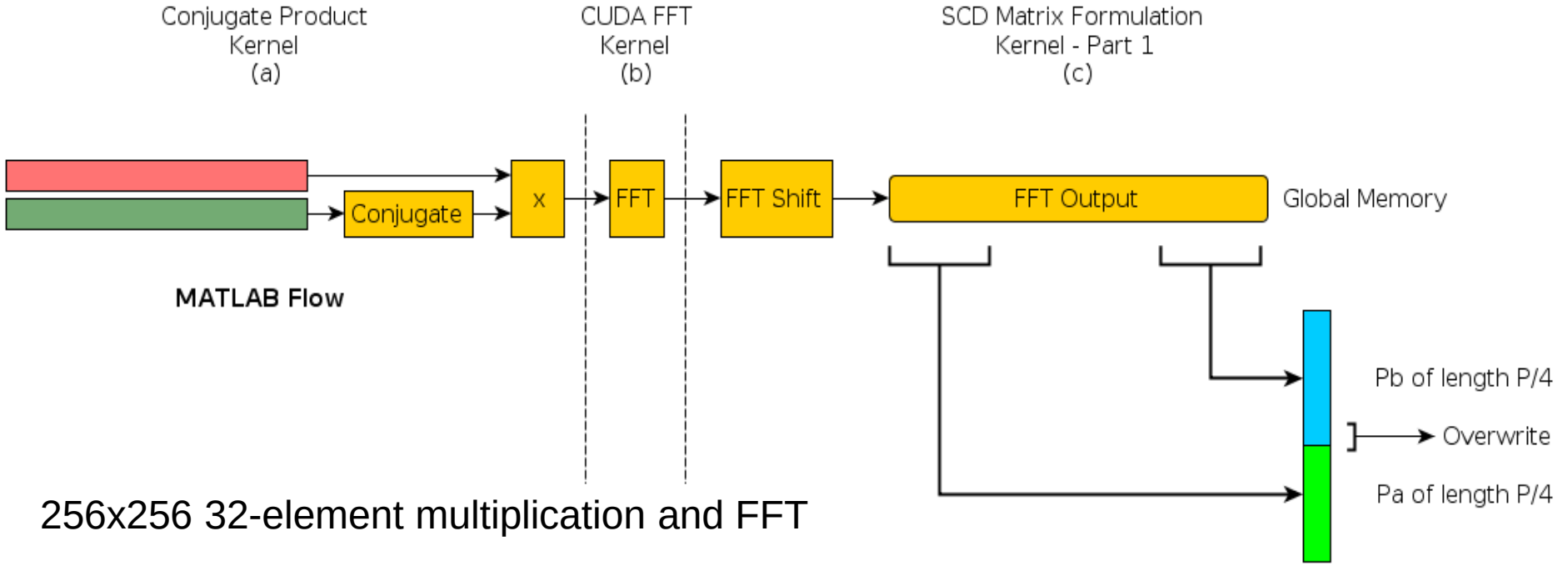
- Step-4a: Down Conversion (Kernel 3)
  - Apply the window to multiple frames
  - Each element is handled by a thread.



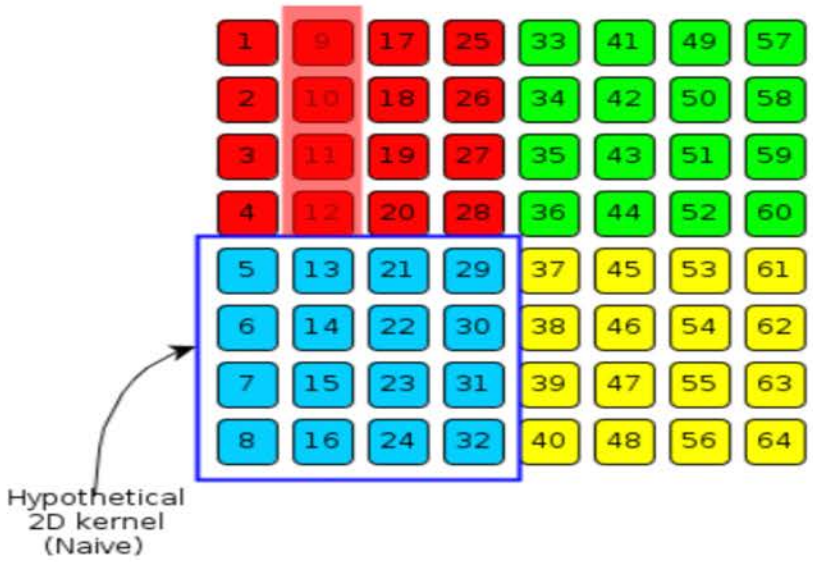
$$out_{km} = fft\_out_{km} * exp\left(\frac{-2\pi jkmL}{Np}\right)$$

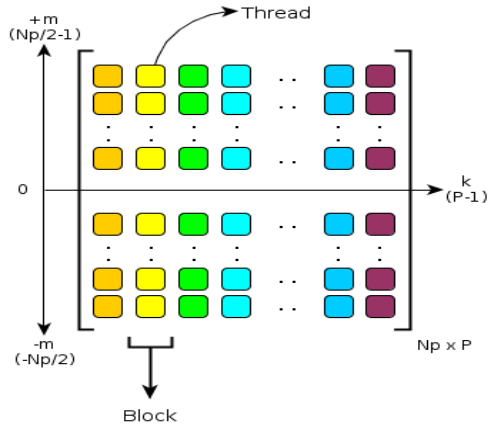
Spectral Correlation Density

THE UNIVERSITY OF ARIZONA®

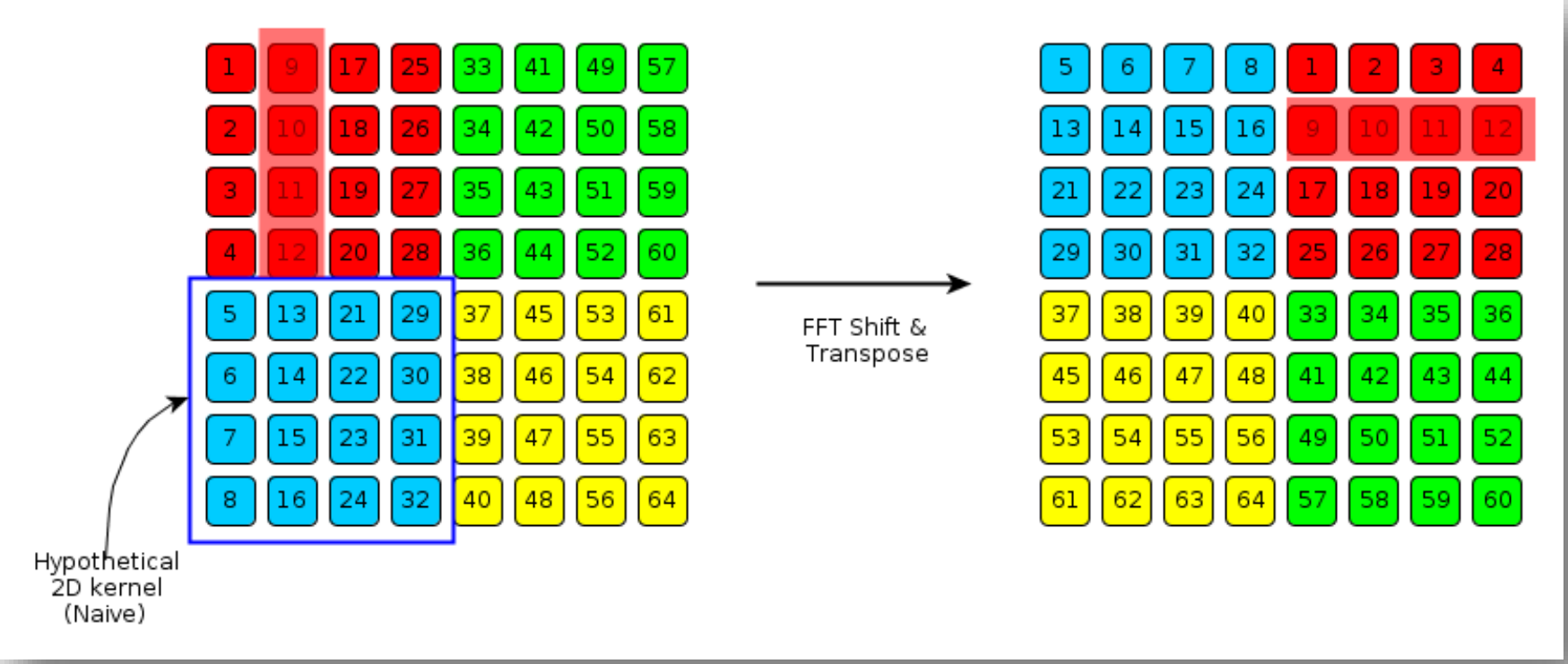


256x256 32-element multiplication and FFT

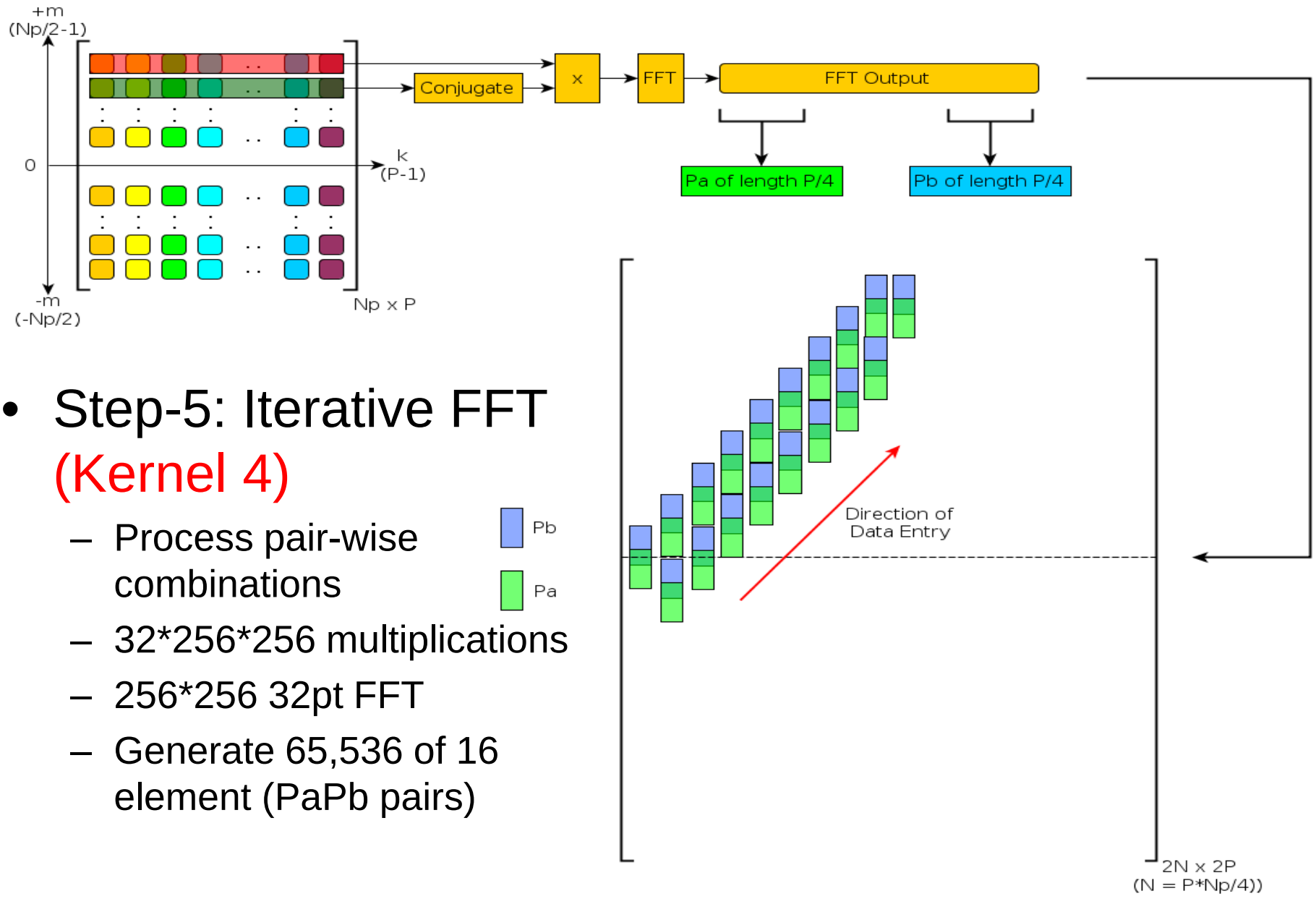




- Step-4b: FFT Shift and Transpose (Kernel 3)
  - Matrix is arranged in column major order
  - Setting up for coalesced memory access for FFT
  - Color coding shows how the data move before and after the transformation takes place.



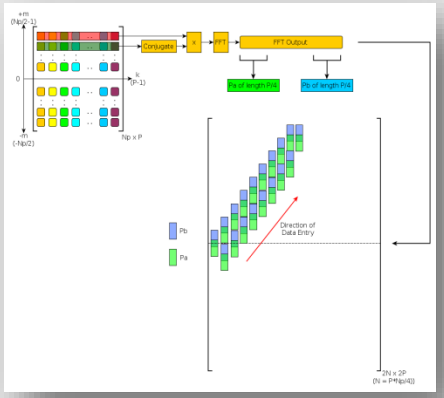
Spectral Correlation Density



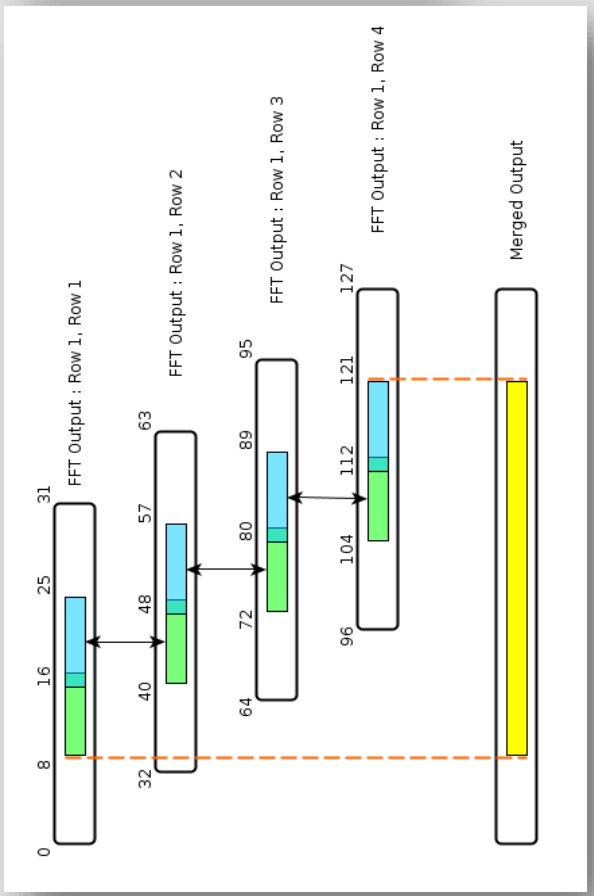
- Step-5: Iterative FFT (Kernel 4)
  - Process pair-wise combinations
  - 32*256*256 multiplications
  - 256*256 32pt FFT
  - Generate 65,536 of 16 element (PaPb pairs)

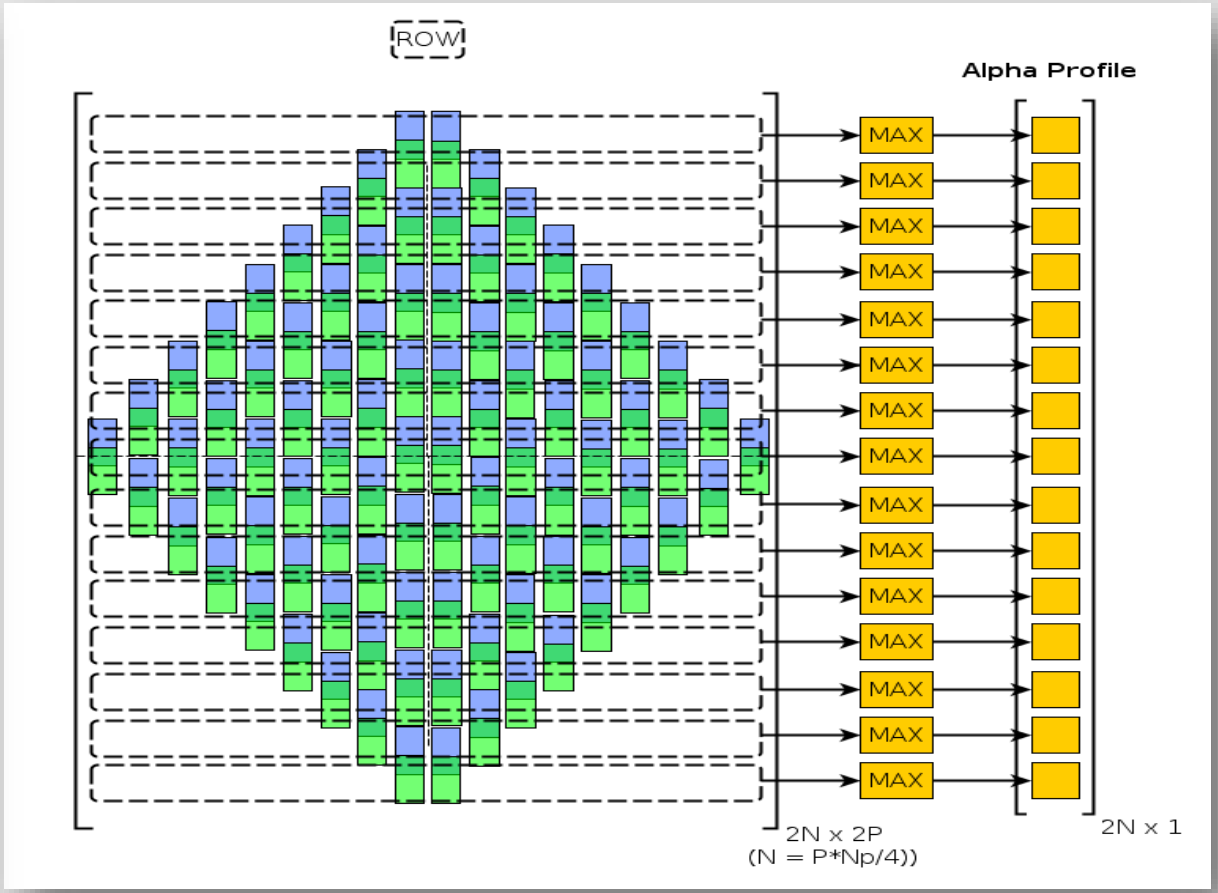
Spectral Correlation Density

THE UNIVERSITY OF ARIZONA®



- Step-6: Signal Profile Generation (Kernel 5)





Profiling of the entire process

THE UNIVERSITY OF ARIZONA®

Computation Efficiency Stages	GPU	Characteristic
Kernel 1. Framing & Windowing	100.00%	Compute Intensive (suitable for GPU)
Kernel 2. FFT (set 1)	100.00%	
Kernel 3. DC + FFT Shift + Transpose	100.00%	
Kernel 4. Iterative FFT	100.00%	
Kernel 5.1 Partial signal profile	76.32%	Data movement intensive (suitable for FPGA)
Kernel 5.2 Merge partial signal profiles	58.33%	
Kernel 5.3 Update main signal profile	98.19%	

Computation efficiency is the thread utilization per multiprocessor on the GPU

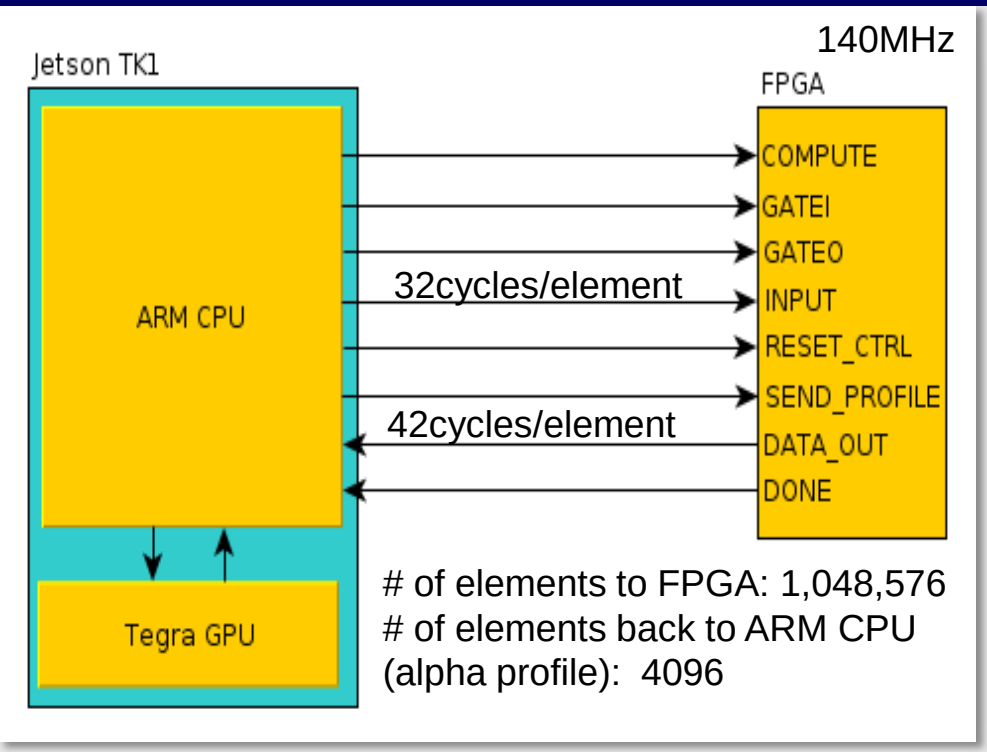
GPU Results

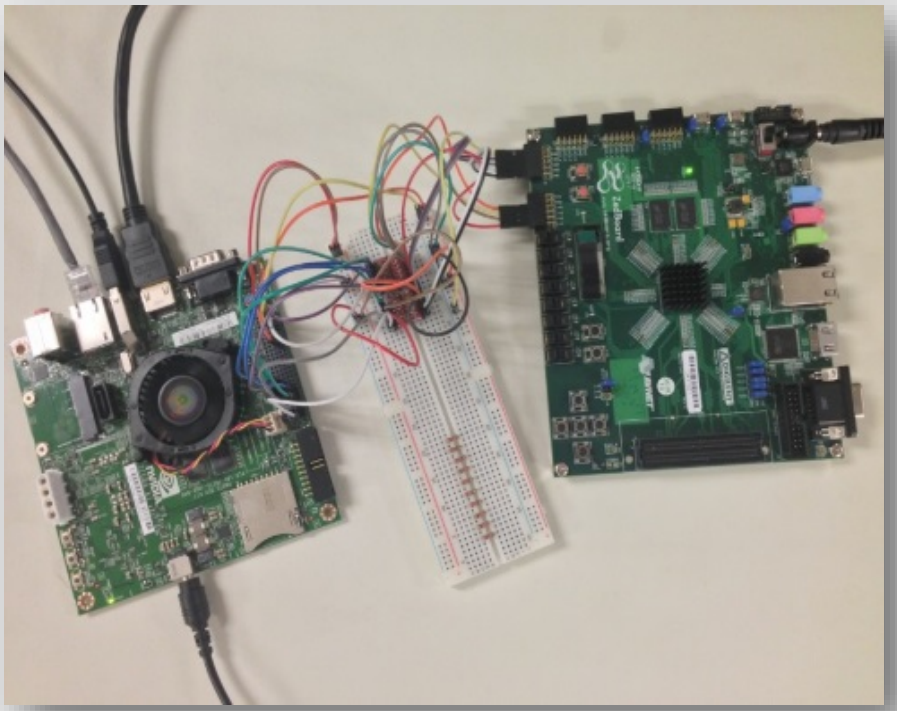
- Mapped the entire SCD process onto two types of GPUs
- K20
  - consumes ~51W, costs ~$3000
- Tegra K1
  - consumes ~3.5W, costs ~$200

GPU	Matlab on Intel I5 (2.3GHz, 8GB RAM)	CUDA on K20 (706MHz, 2496 cores)	CUDA on Tegra K1 (850MHz, 192 cores)
Execution Time/Signal (ms) (includes data transfer)	3502.29	8.96	111.19
Speedup		390X	31X
Throughput (Signals/sec)		111.61	9.01

- Results validated against the Matlab implementation
  min. error of 0.0041% and a maximum error of 0.0051%.
- Execution time is based on 4096 points digital signal.
- Input parameters (window size - 256 and number of parts of the signal - 32)

Bit-serial (Demo version)

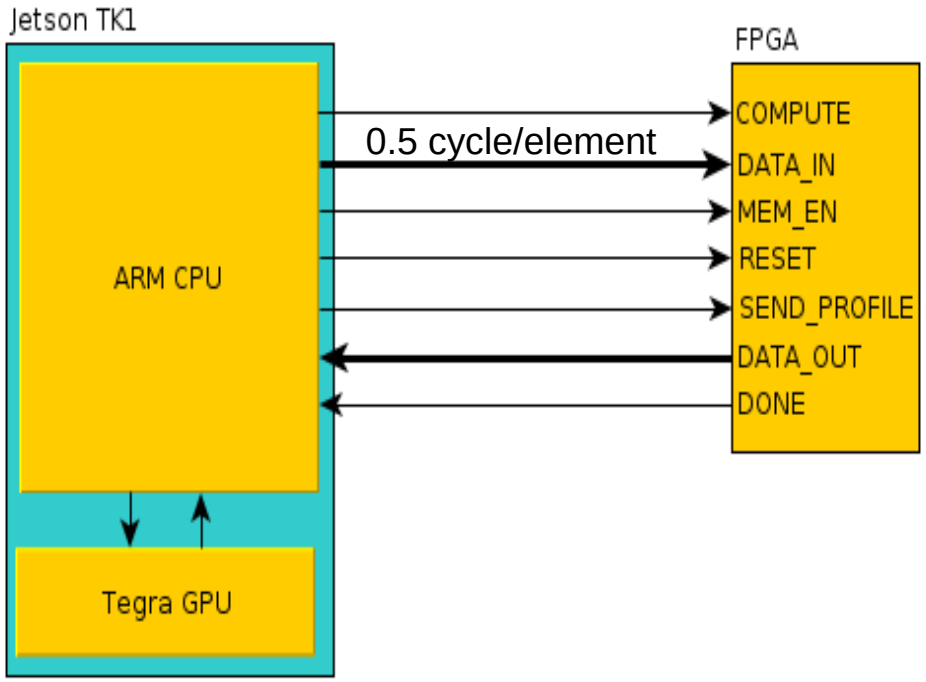


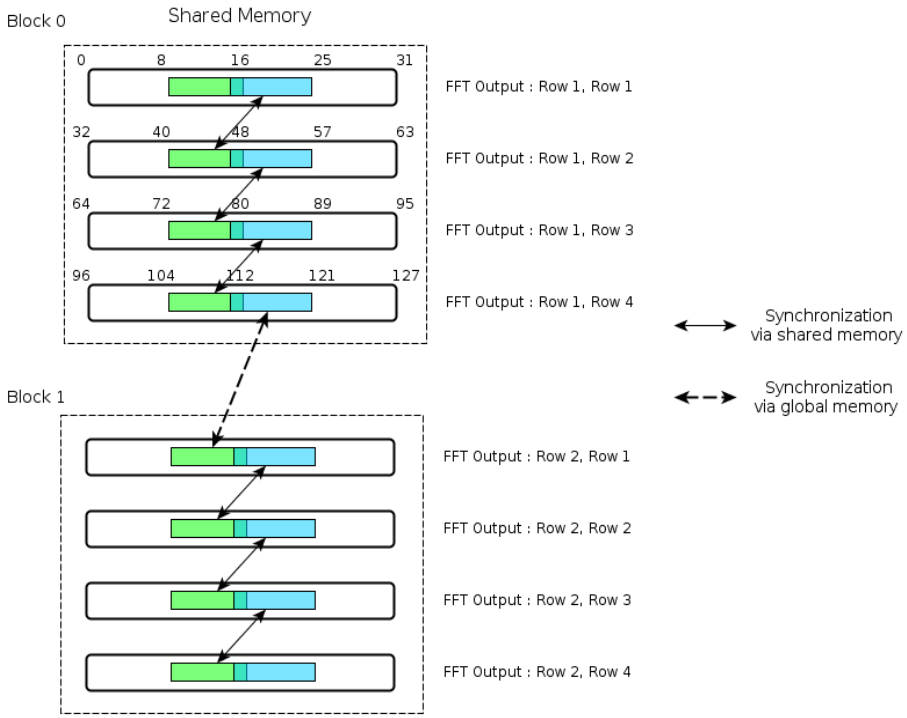


- FPGA Time: 278.35ms (including data transfer from and to FPGA)

Parallel Data Transfer – FPGA Compute Cycles

THE UNIVERSITY OF ARIZONA®





- 484 pins available on the FPGA
- 64-bit bus to transfer 2 elements per cycle
  - require 101 pins.
- 5 compute cycles for n-lanes

Tegra K1 (ms)	FPGA (ms) 64 bit data bus			
	1-way	16-way	32-way	64-way
6.59	37.53	6.17	5.00	4.42

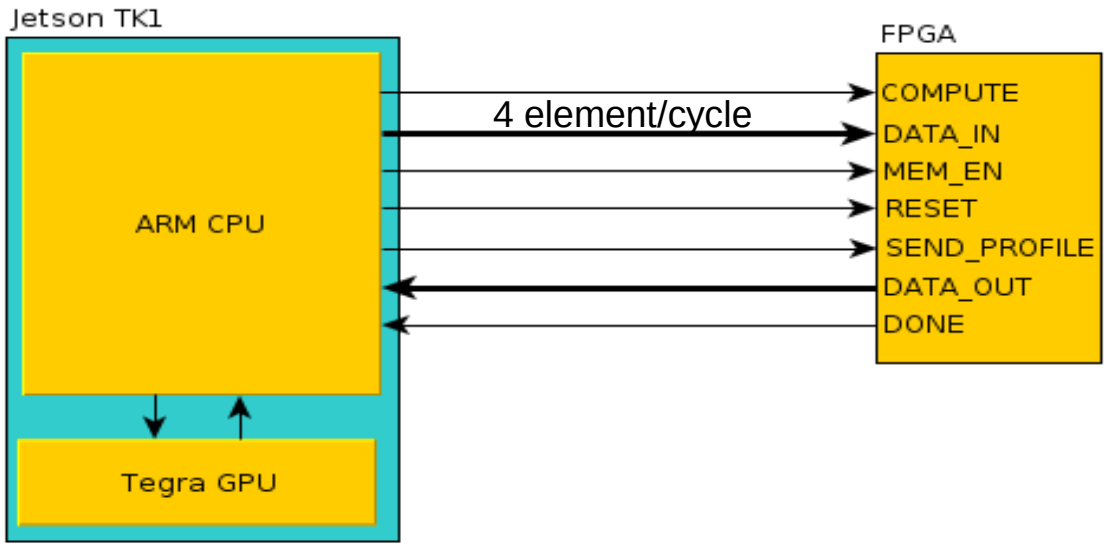
Number of elements vs N-way

Computation times for Kernels 5.1, 5.2 and 5.3 (in ms)					
Tegra TK1	*Bit Serial*	*Simple Parallel*	*16-way Parallel*	*16-way Parallel*	*64-way Parallel*
			1 Element	*2 Elements*	*4 Elements*
6.586 ms	278.352 ms	45.027 ms	9.918 ms	6.173 ms	1.609 ms
	0.02x	0.15x	0.66x	1.07x	4.09x

- 4 elements per cycle (each 32 pins)
  - 165 bit input data bus
- Combined with 64-way

Coupling Tegra K1 with FPGA

THE UNIVERSITY OF ARIZONA



Computation Efficiency		
Stages	GPU	Characteristic
Kernel 1. Framing & Windowing	100.00%	Compute Intensive (GPU)
Kernel 2. FFT (set 1)	100.00%	
Kernel 3. DC + FFT Shift + Transpose	100.00%	
Kernel 4. Iterative FFT	100.00%	
Kernel 5.1 Partial local signal profile	76.32%	Data movement intensive (FPGA)
Kernel 5.2 Merge partial local signal profile	58.33%	
Kernel 5.3 Update main signal profile	98.19%	

Throughput (signals per second)

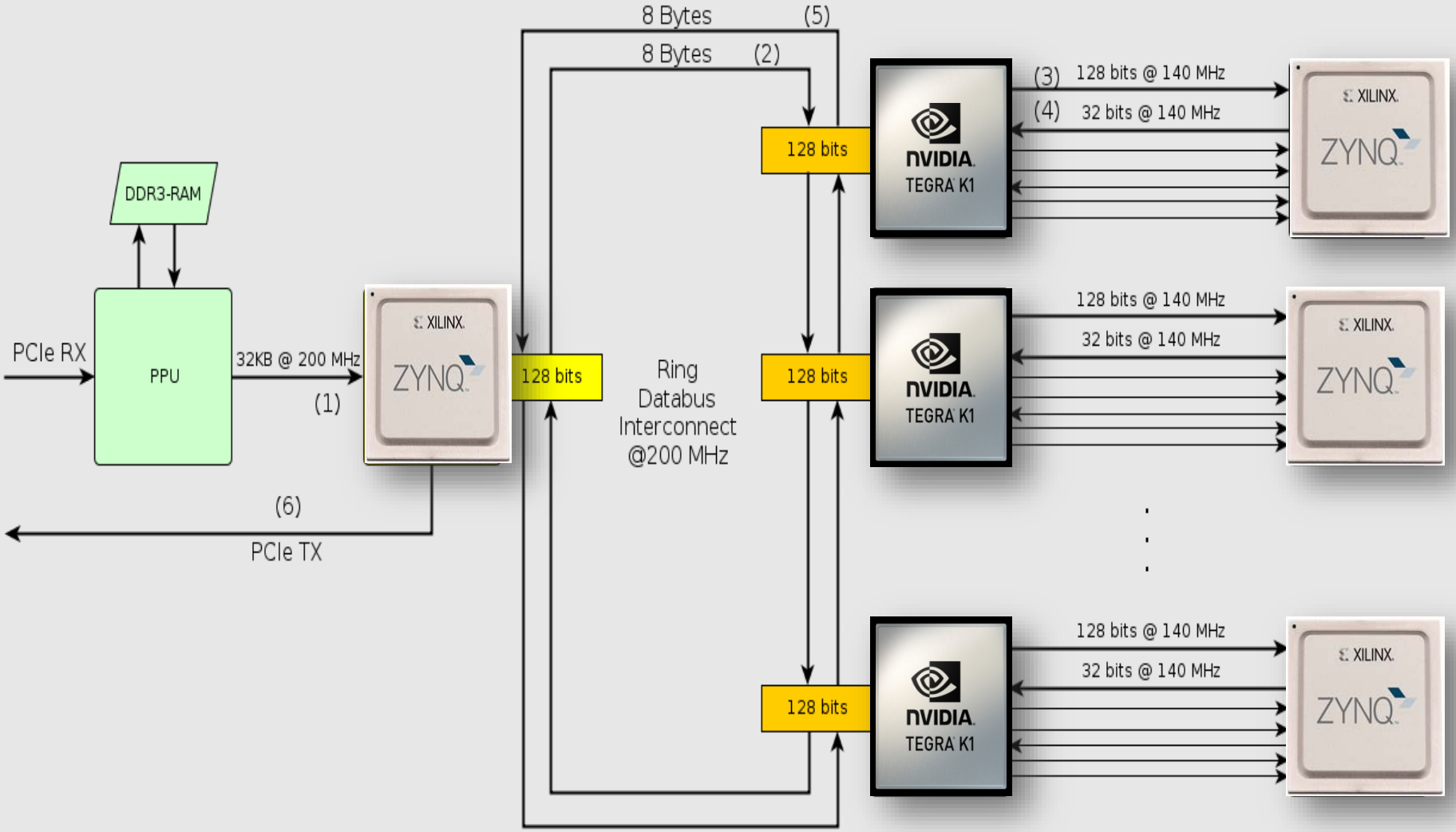
GPU-Only	GPU-FPGA				
Tegra-K1	Bit Serial	Simple Parallel	16-way 1 Element	16-way 2 Elements	64-way 4 Elements
9	3	10	16	17	19

- Tegra K1 (Kernels 1-4) and FPGA (Kernel 5)
  - Total execution time: 50.95ms
  - Throughput 19 signals/second

Performance Comparison	Serial	GPU-Only		Hybrid
Platform	Intel I5	K20	Tegra K1	FPGA + Tegra K1
Total Time – SCD (per signal)	3502.28	8.98	111.61	50.95
Speed-up over MATLAB	--	390x	31x	68x
Throughput (Signals/second)	<1	111	9	19
Power Consumption		51W	3.5W	~5W

- Serial
  - Matlab on Intel I5 (2.3GHz, 8GB RAM)
- GPU-Only:
  - CUDA on K20 GPU (706MHz, 2496 Cores, $3,000)
  - CUDA on Tegra GPU (850 MHz, 192 Cores, $192)
- Hybrid: Tegra TK1 GPU and Zynq FPGA (Zedboard, $319)

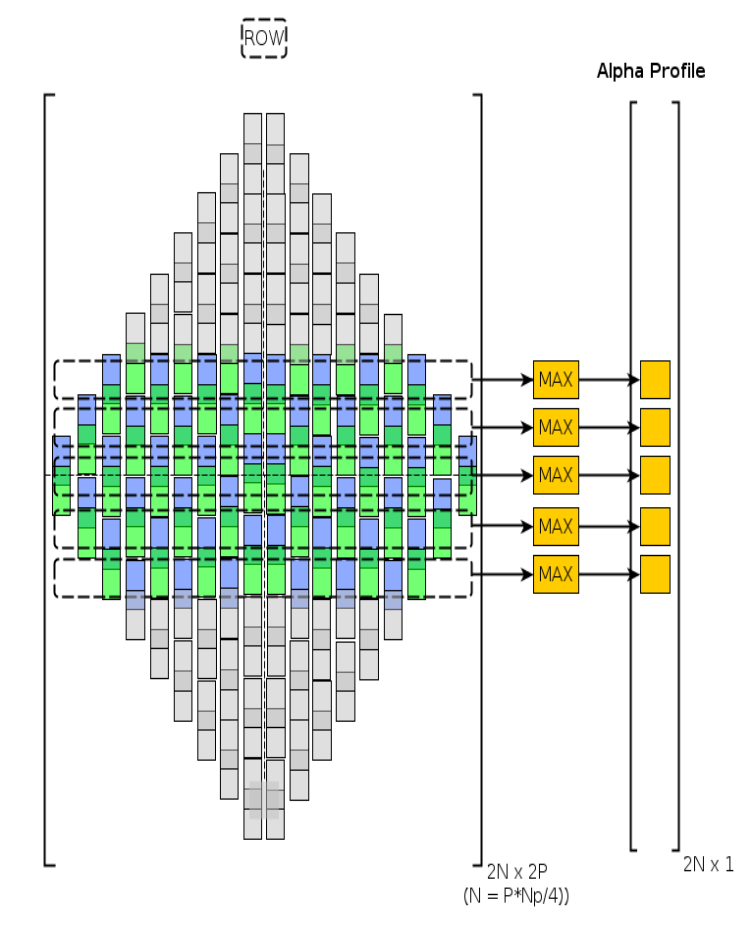
Multi-Lane Architecture



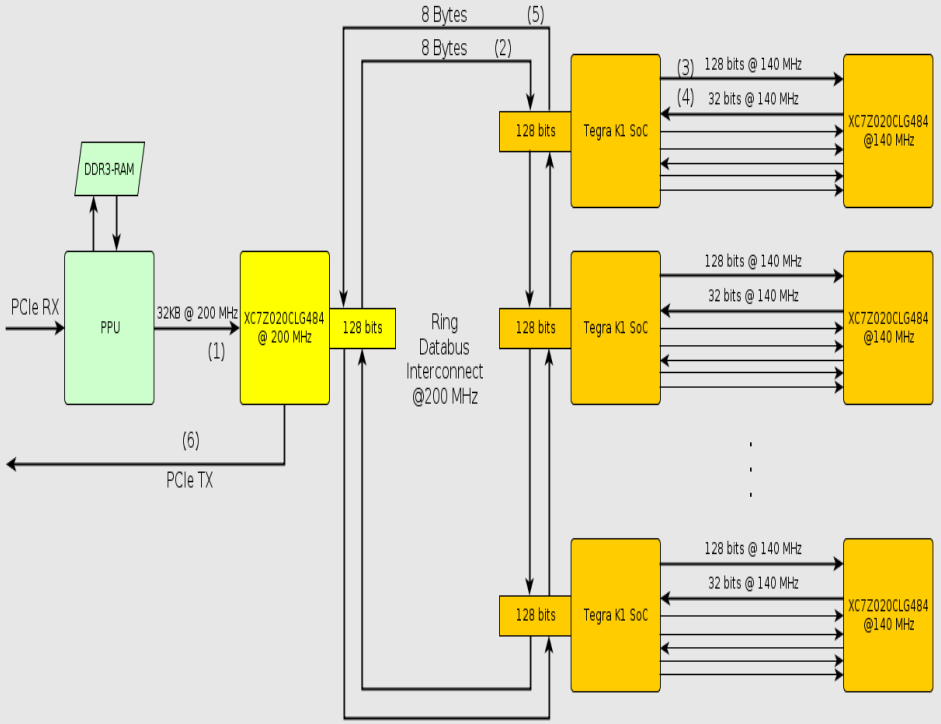
- Partition the data parallel workload among lanes
  - N-signals , Up to 8 is natural, Minimum 2 lanes

Future Parallelization Efforts

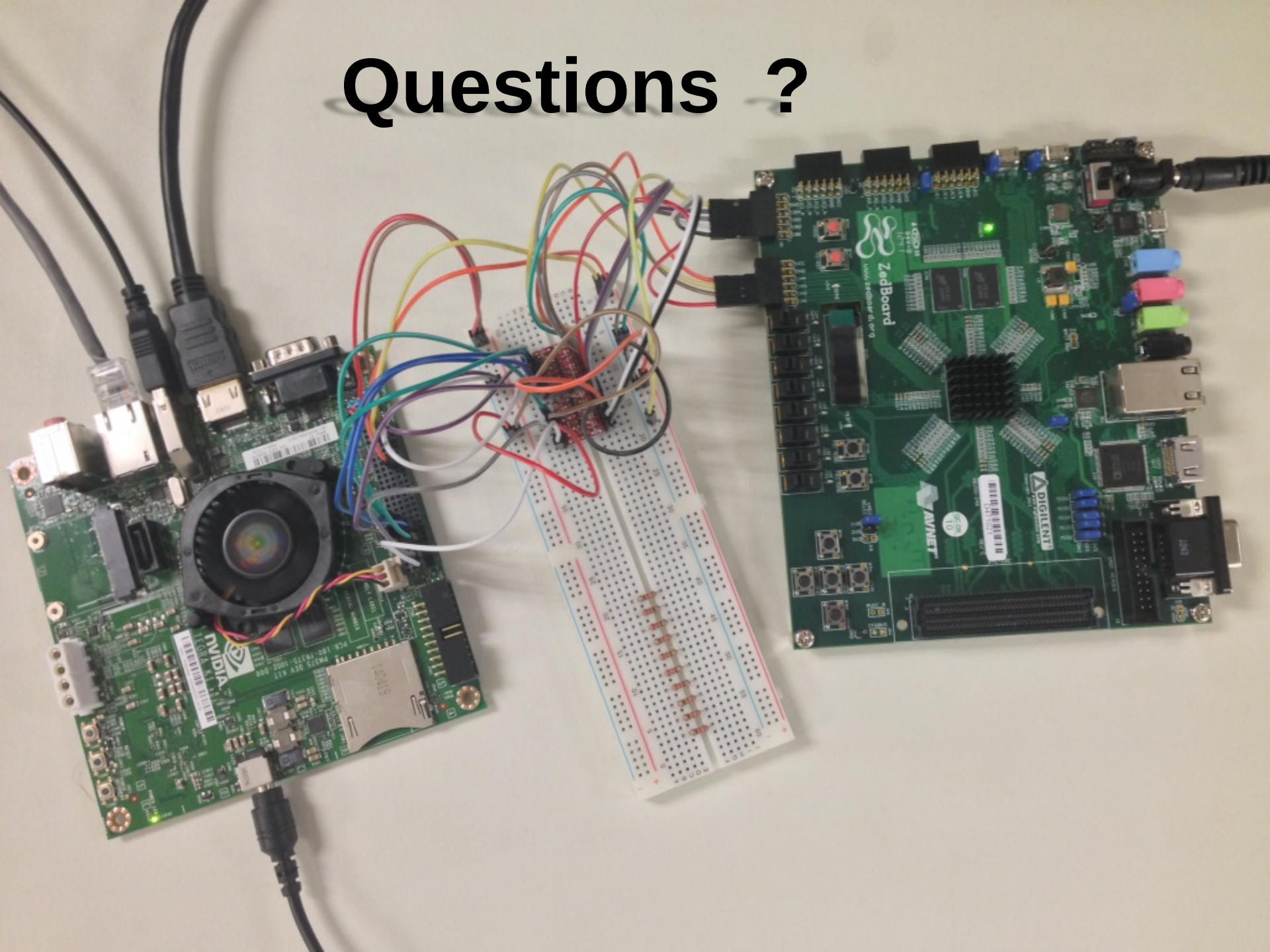
- Within the signal



- Across signals
  - Deeply pipelining
  - Burst send/burst receive mode overlapping with computation

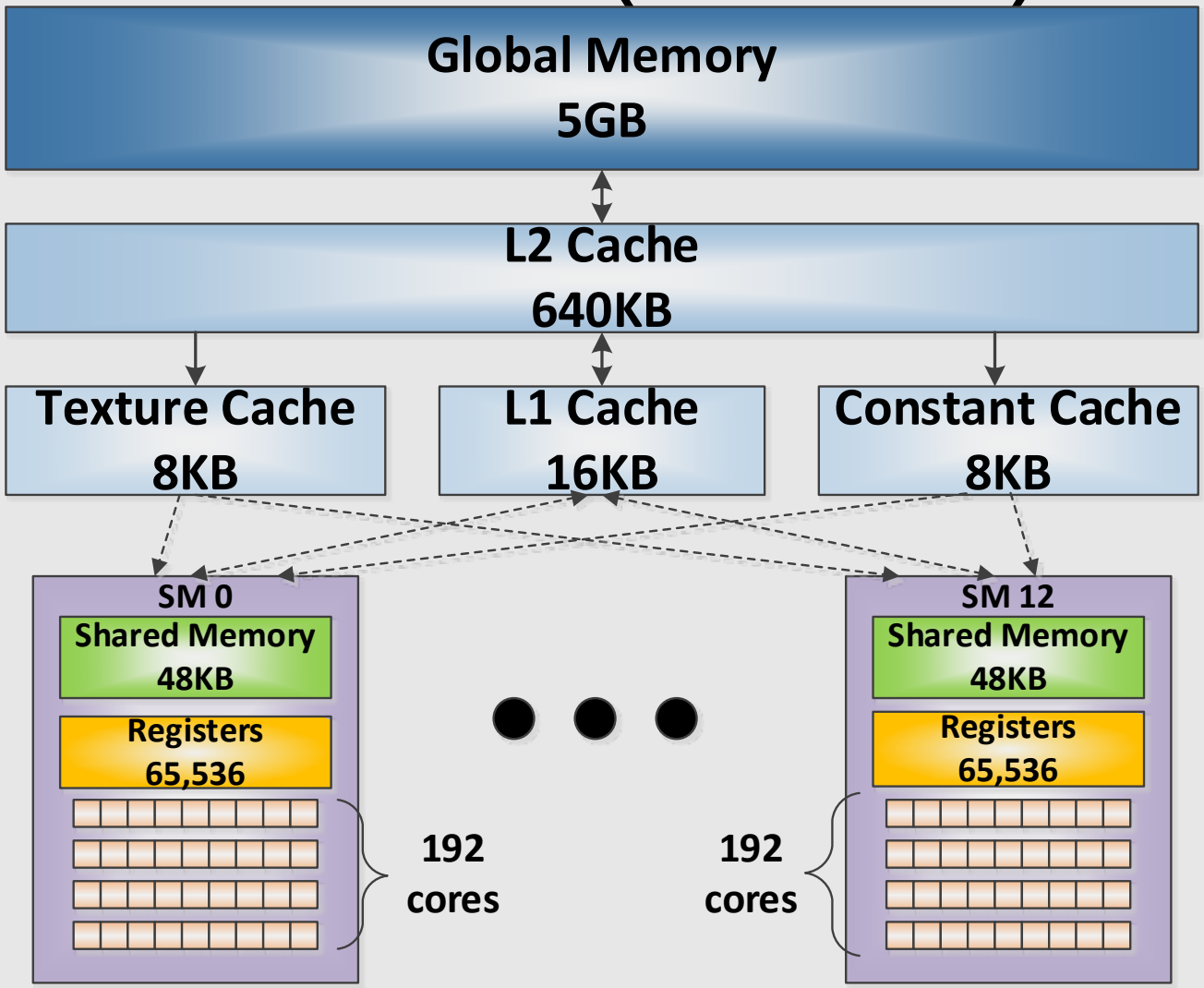


Questions ?

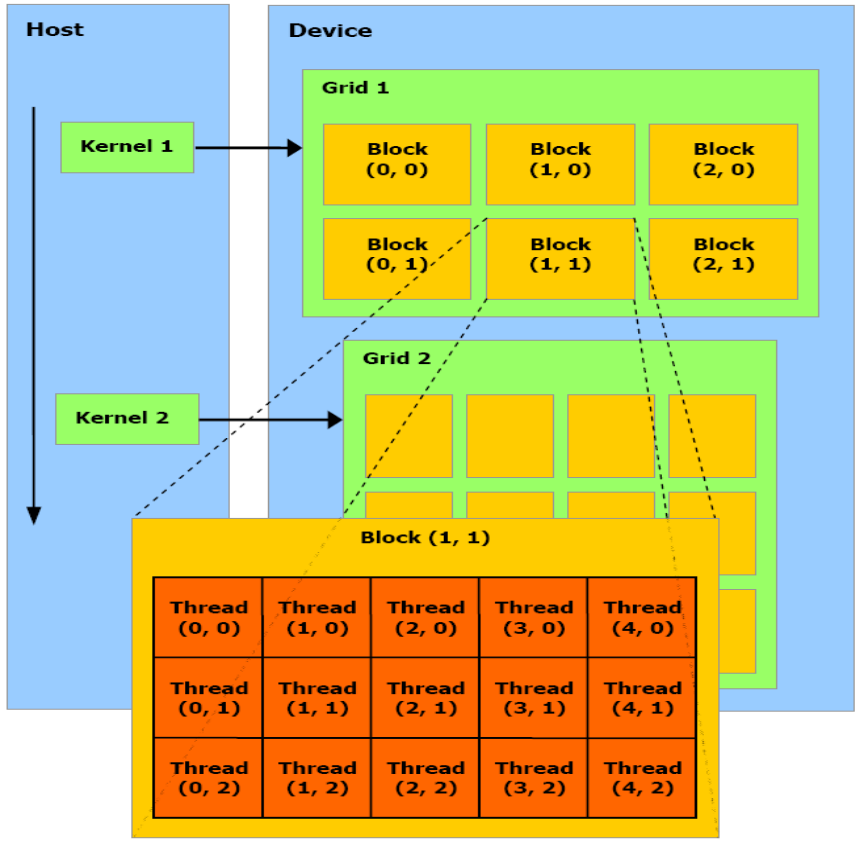


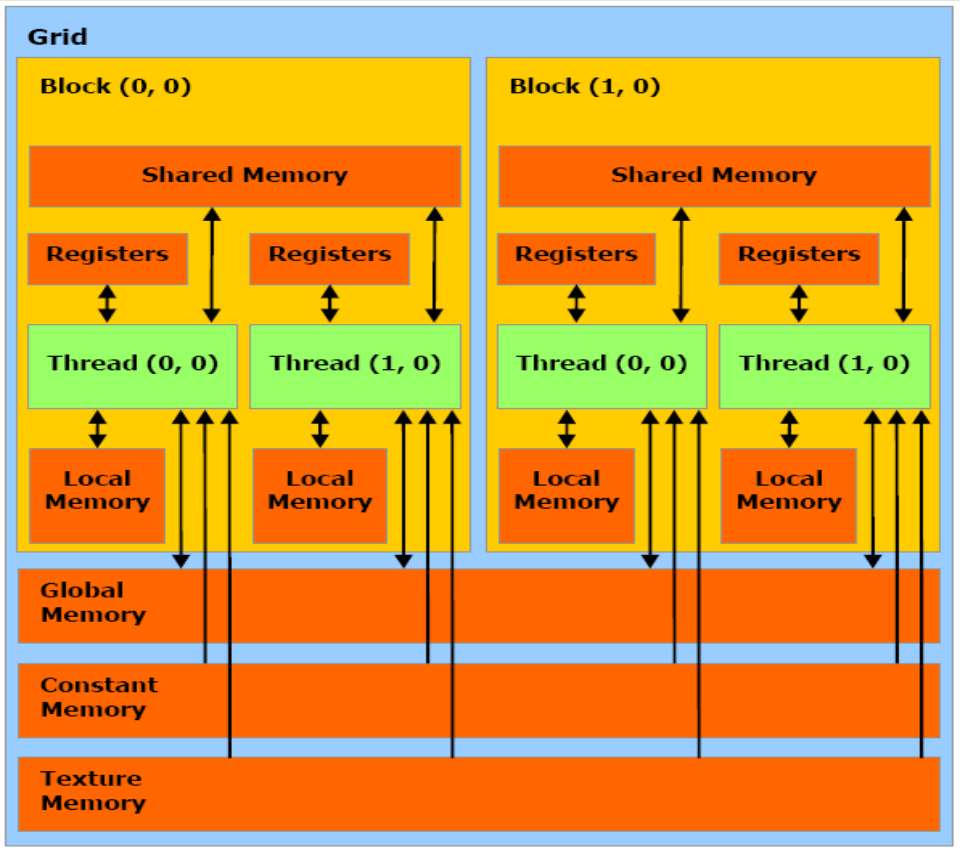
BACKUP

K20 GPU (706MHz)



Up to 2048
Threads/SM

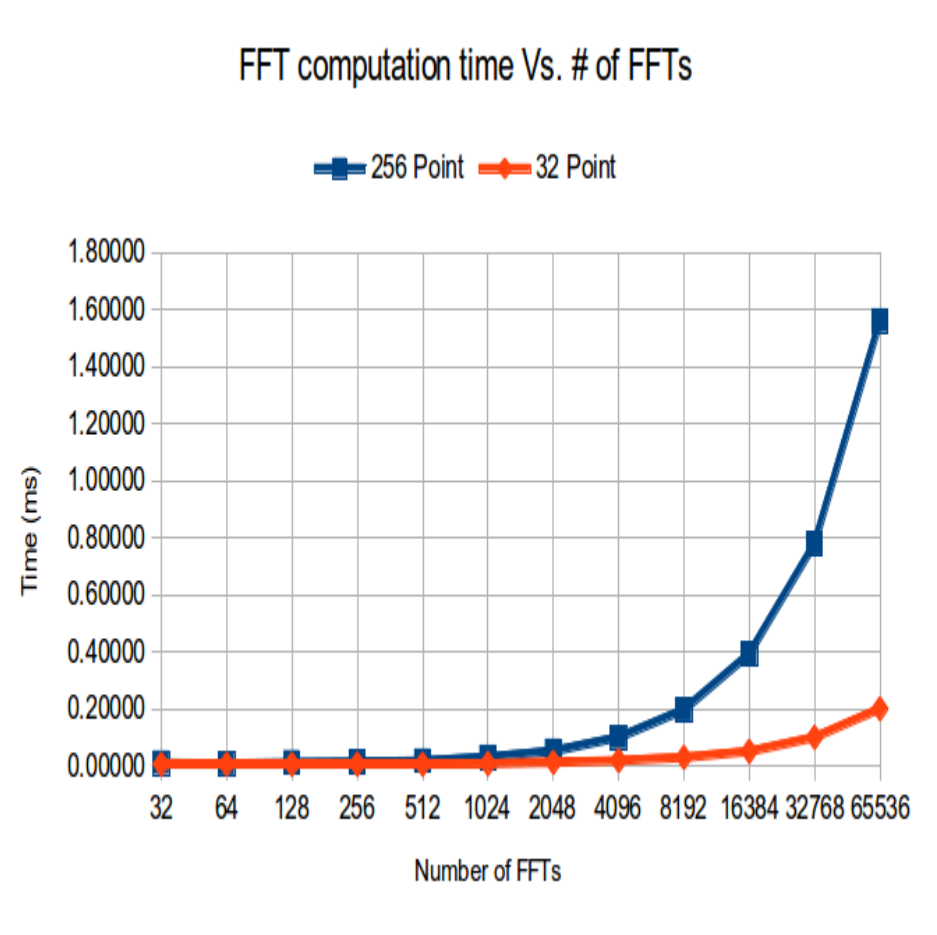




- A thread is associated with each data element
- Hardware handles thread management, not OS
- 32 threads within a block work collectively

FFT Analysis on GPU

	K20 – 256 Points	K20 – 32 points
# FFTs	Time (ms)	Time (ms)
32	0.01152	0.00746
64	0.00909	0.00749
128	0.01299	0.00717
256	0.01485	0.00774
512	0.01888	0.00742
1024	0.03216	0.00954
2048	0.05335	0.01392
4096	0.10128	0.02026
8192	0.19789	0.03110
16384	0.39363	0.05219
32768	0.78004	0.10144
65536	1.56023	0.20167



GPU executes 2048 – 256 points FFT or 16384 – 32 points FFT in parallel

FFT Analysis

FFT execution time for the workload defined in the SCD flow during Iterative FFT Stage (256*256=65536 32-point FFTS)			
	Virtex 7 (XC7VX690T)	**K20 GPU**	**Tegra K1 GPU**
12 Channel (Parallel) FFT (32 point)	0.002014286	0.0504175	0.0504175
Max FFTs (resource limit) (32 point)	216	16384	2048
65536 32 point FFTs in batches	304	4	32
Total time (ms)	0.612342857	0.20167	1.61336
Precision supported	32 bit magnitude, 32 bit phase	32 bit magnitude, 32 bit phase	32 bit magnitude, 32 bit phase
Type	Fixed point	Floating point	Floating point

- Workload size makes K20 GPU a better option
- Tegra K1 behind Virtex 7!

Resource Analysis on FPGA

FPGA Summary : Virtex 7 (XC7VX690T-2FFG1761C)				
	LUT	FF	18K BRAM	DSP
	433200	866400	2940	3600
Resources used per 32 point FFT	2559	3094	8	12
12 Channel (Parallel) FFT (32 point)	16950	23741	50	192
FFT Caps (# of 12 Channel FFTs)	25	36	58	18
FFT Caps (# of parallel FFTs)	300	432	696	**216**
Resources used per 32 bit complex multiplication	263	720	0	12
Multiplication Caps (# of parallel multiplications)	1647	1203		**300**

- 216 parallel 32-point FFTs or 300 complex multiplications
- N-bit multiplication:
  - n^2 LUTs
  - large multiplier longer delay if LUT-based
    - more flip flops, multiply-accumulate pipeline

Multiplication Analysis

Multiplication execution time for the workload defined in the SCD flow during Iterative FFT Stage for Conjugate Product Calculations (256*256*32= 2097152 pairwise multiplications)			
	Virtex 7 (XC7VX690T)	**K20 GPU**	**Tegra K1 GPU**
32-bit complex multiplication	0.000092857	0.0000004206	0.0000057373
Max. parallel multiplciations	**300**	26624	2048
Multiplication in batches	6991	79	1024
Total time (ms)	0.649164286	0.00003322	0.005875

- Workload size makes K20 GPU and Tegra K1 a better option

Multiplication and FFT Analysis

32x32 Conjugate Multiplication followed by 32-point FFT execution time during Iterative FFT Stage (256*256*32= 2097152 pairwise multiplications and 256*256=65536 FFTs)		
	Virtex 7 (XC7VX690T)	**Tegra K1 GPU + Zynq FPGA**
Multiply + FFT	0.00211	0.0504
Max. parallel operations	9	
Batches	7282	
Total time (ms)	15.34	1.619

- Each 32x32 multiplication coupled with 1 32-point FFT
  - Maximum of 288 multiplications and 9 FFTs concurrently
- Configuring FPGA for multiplication followed by reconfiguring the entire device for the FFT is not feasible
- Virtex7 @ 140MHz (Signal profile on Zynq FPGA @140MHz)
  - Even at 500MHz, Tegra K1+FPGA is a better choice

Cost Analysis

FFT execution time for the workload defined in the SCD flow during Iterative FFT Stage (256*256=65536 32-point FFTS)			
	Virtex 7 (XC7VX690T)	K20 GPU	Tegra K1 GPU
Cost ($)	7,836	3021	198
Peak Power (W)	5.68	225	5

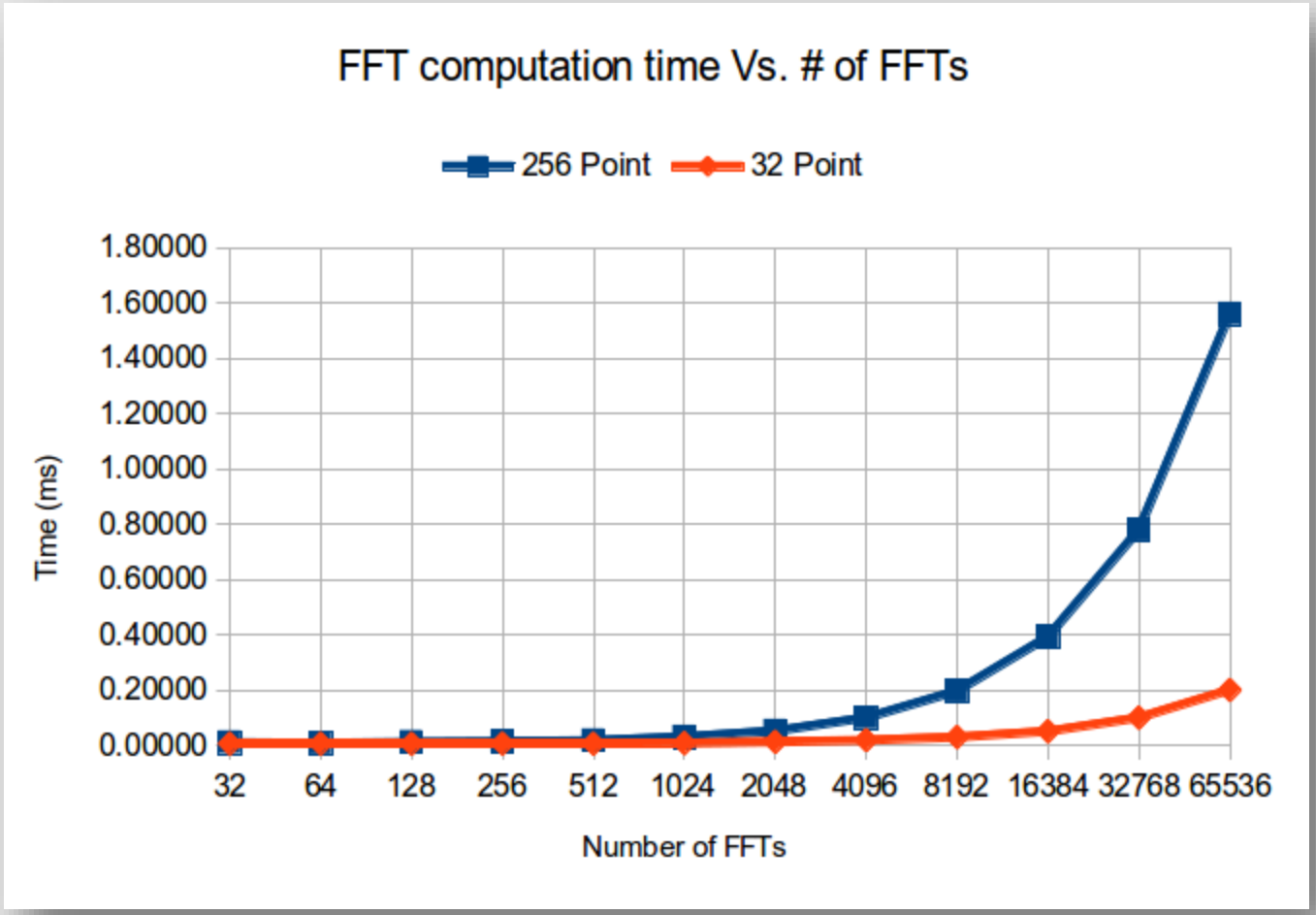
GPU Results

- Mapped the entire SCD process onto two types of GPUs
- K20
  - consumes ~51W, costs ~$3000
- Tegra K1
  - consumes ~3.5W, costs ~$200
  - **Not enough to achieve 30 signals/second with a single GPU**

GPU	Matlab on Intel I5 (2.3GHz, 8GB RAM)	CUDA on K20 (706MHz, 5GB)	CUDA on Tegra K1 (Mobile Platform)
Execution Time/Signal (ms) (includes data transfer)	3502.29	8.96	111.19
Speedup		390X	31X
Throughput (Signals/sec)	<1	111.61	9.01

- Results validated against the Matlab implementation
  min. error of 0.0041% and a maximum error of 0.0051%.
- Execution time is based on 4096 points digital signal.
- Input parameters (window size - 256 and number of parts of the signal - 32)

FFT Analysis on GPU



GPU executes 2048 – 256 points FFT or 8192 – 32 points FFT in parallel

FFT Analysis on FPGA

	Virtex 7 (XC7VX690T)	GPU - K20
32-point FFT	3.73 microsec	6.19 microsec
# of concurrent 32 point FFTs	98	256
Iterations	669	256
Total time	~2.5ms	1.58ms
Precision supported	32 bit Real, 24 bit Imaginary	32 bit Real, 32 bit Imaginary

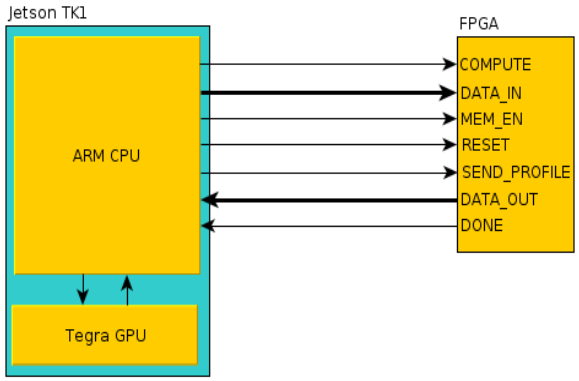
SCD Stages

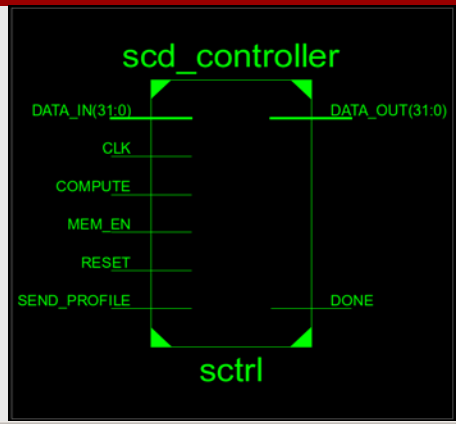
THE UNIVERSITY OF ARIZONA®

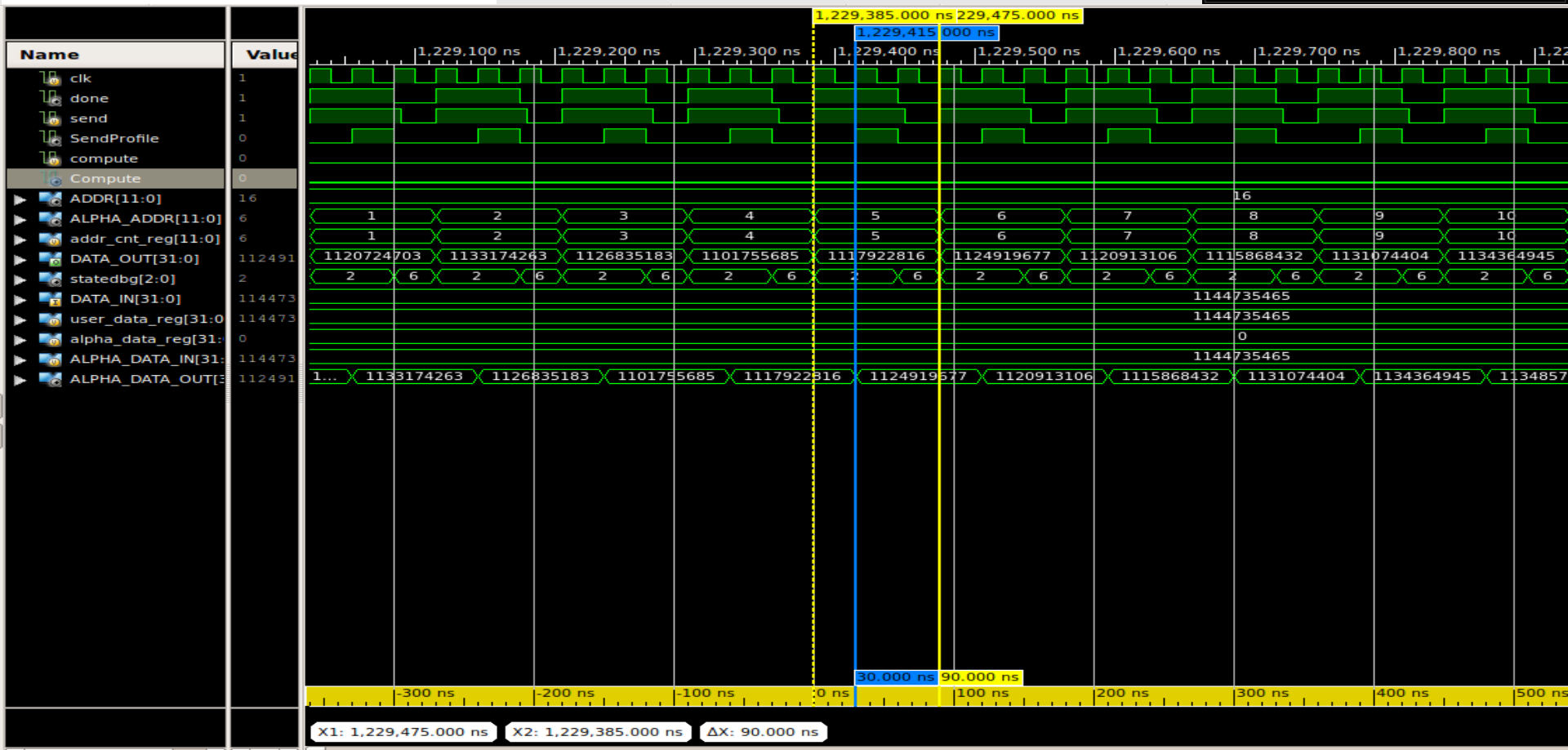
(ms)	Serial	GPU-Only		Hybrid
Kernel	MATLAB	K20	Tegra K1	FPGA + Tegra K1
Framing & Windowing	0.531	0.006	0.259	0.259
FFT (set 1)	0.965	0.012	0.130	0.130
FFTShift + DC + Transpose	1.321	0.005	0.125	0.125
Conjugate Products	3100.00	0.882	0.047	0.047
FFT (set 2)		1.544	0.058	0.068
Compute partial local alpha profile		1.157	0.075	0.076
Merge partial local alpha profile		0.453	0.008	0.006
Update main alpha profile		0.630	0.018	
Time – SCD (per iteration)	--	4.294	0.217	0.197
Total Time – (per signal 256 iterations)	3102.286	8.98	111.61	50.95
Speed-up over MATLAB	--			

Parallel Data Transfer – FPGA to CPU

THE UNIVERSITY OF ARIZONA®







FFT Analysis on FPGA

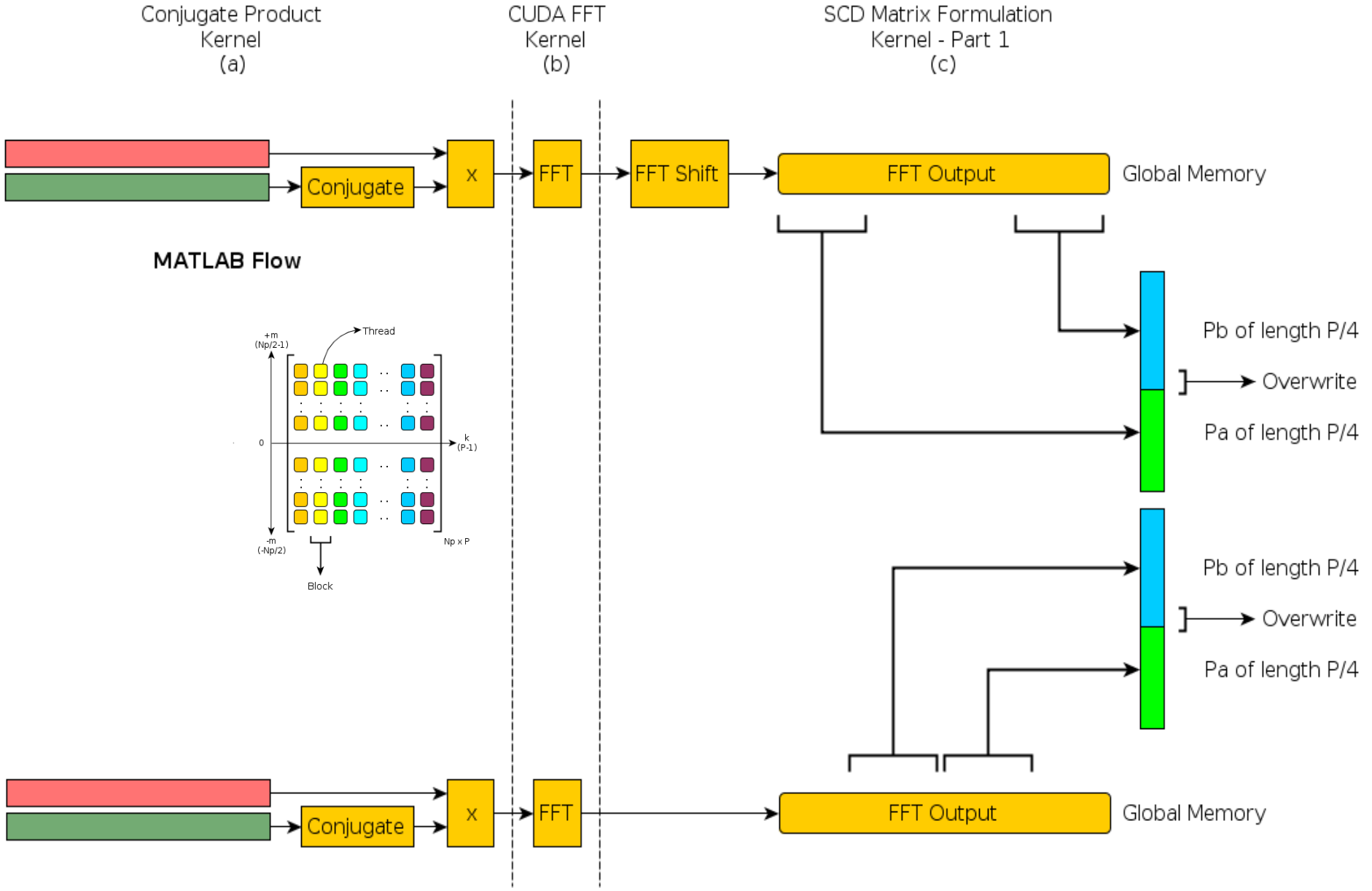
THE UNIVERSITY OF ARIZONA®

FPGA Summary : Virtex 7 (XC7VX690T-2FFG1761C)				
Available Resources	LUT	FF	18K BRAM	DSP
	433200	866400	2940	3600
Resources required per 32 point FFT	4420	5683	8	12
FFT Caps	98	152	367	300

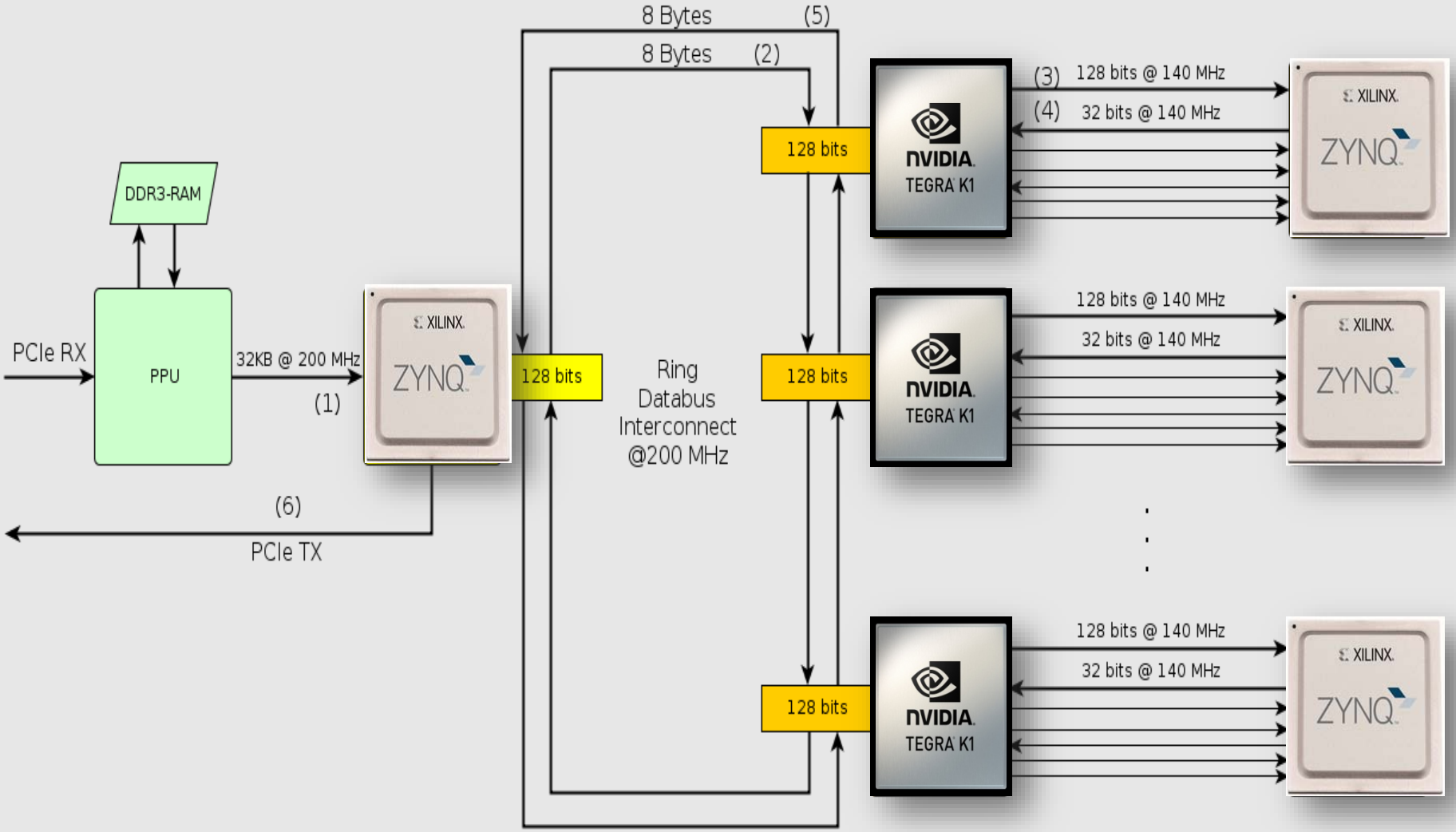
Platform	Virtex 7 (XC7VX690T)	GPU - K20
Cost ($)	7,836	3,021
Peak Power (W)	5.68	225
# of concurrent 32 point FFTs	98	2048
Precision supported	32 bit Real, 24 bit Imaginary	32 bit Real, 32 bit Imaginary

Spectral Correlation Density

THE UNIVERSITY OF ARIZONA®



Multi-Lane Architecture



- Partition the data parallel workload among lanes
  - N-signals , Up to 8 is natural, Minimum 2 lanes

- **Zynq7000:**
- Number of External IOBs 200
- Number of RAMB36E1s 140
- Number of Slices 13300
- Number of Slice Registers 106400
- Number of Slice LUTS 53200