

DSP Acceleration using nnMAX™

Cheng Cheng Wang, Senior VP
Flex Logix Technologies, Inc.

cheng@flex-logix.com

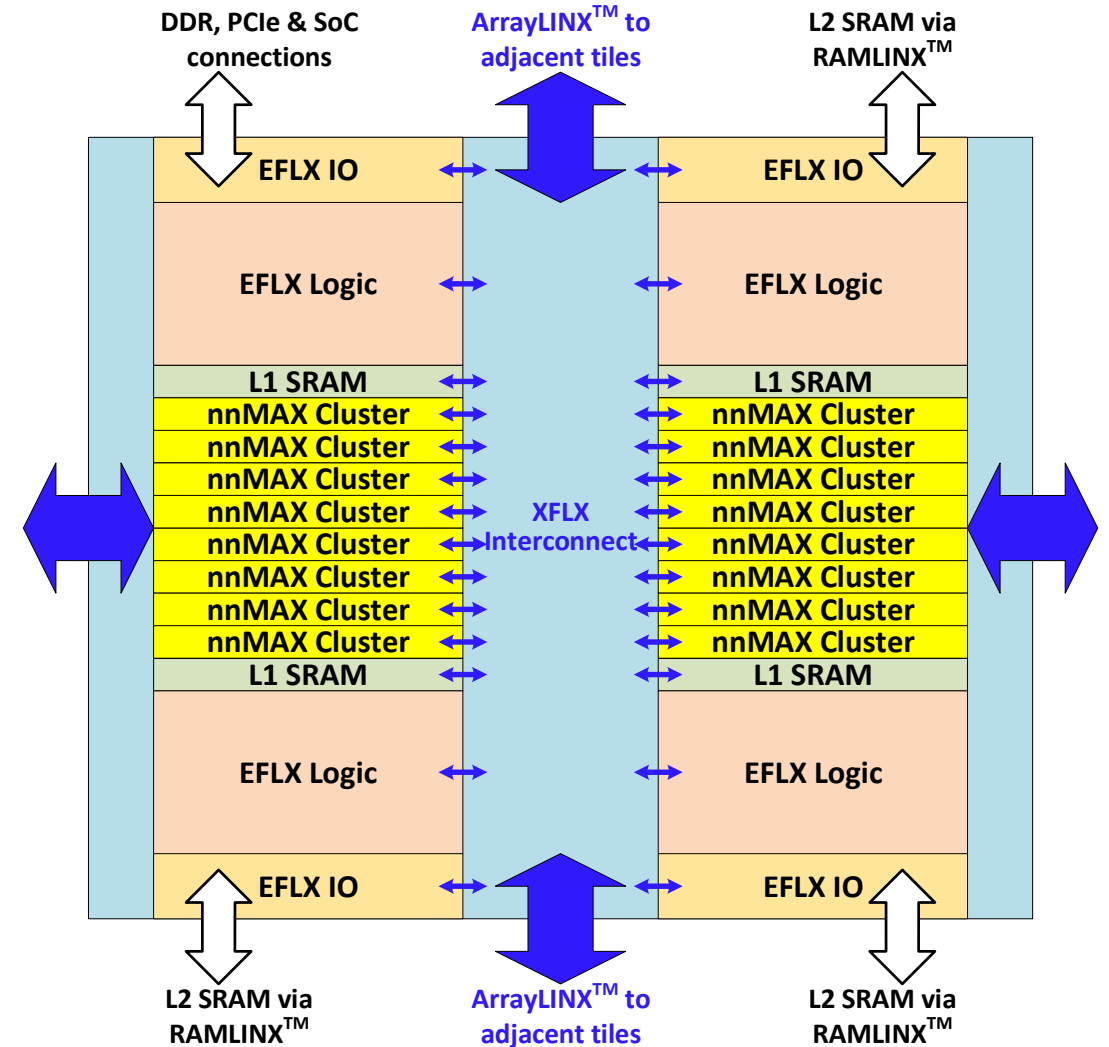
Linley Fall Processor Conference
April 7th 2020, Santa Clara, CA

| Overview

- nnMAX is silicon IP and software developed for AI Inference acceleration
- Customers have asked if we can use the MACs for DSP
- nnMAX is very good at FIR filters: faster than FPGAs at much lower cost
- We have customers now planning to use nnMAX for DSP

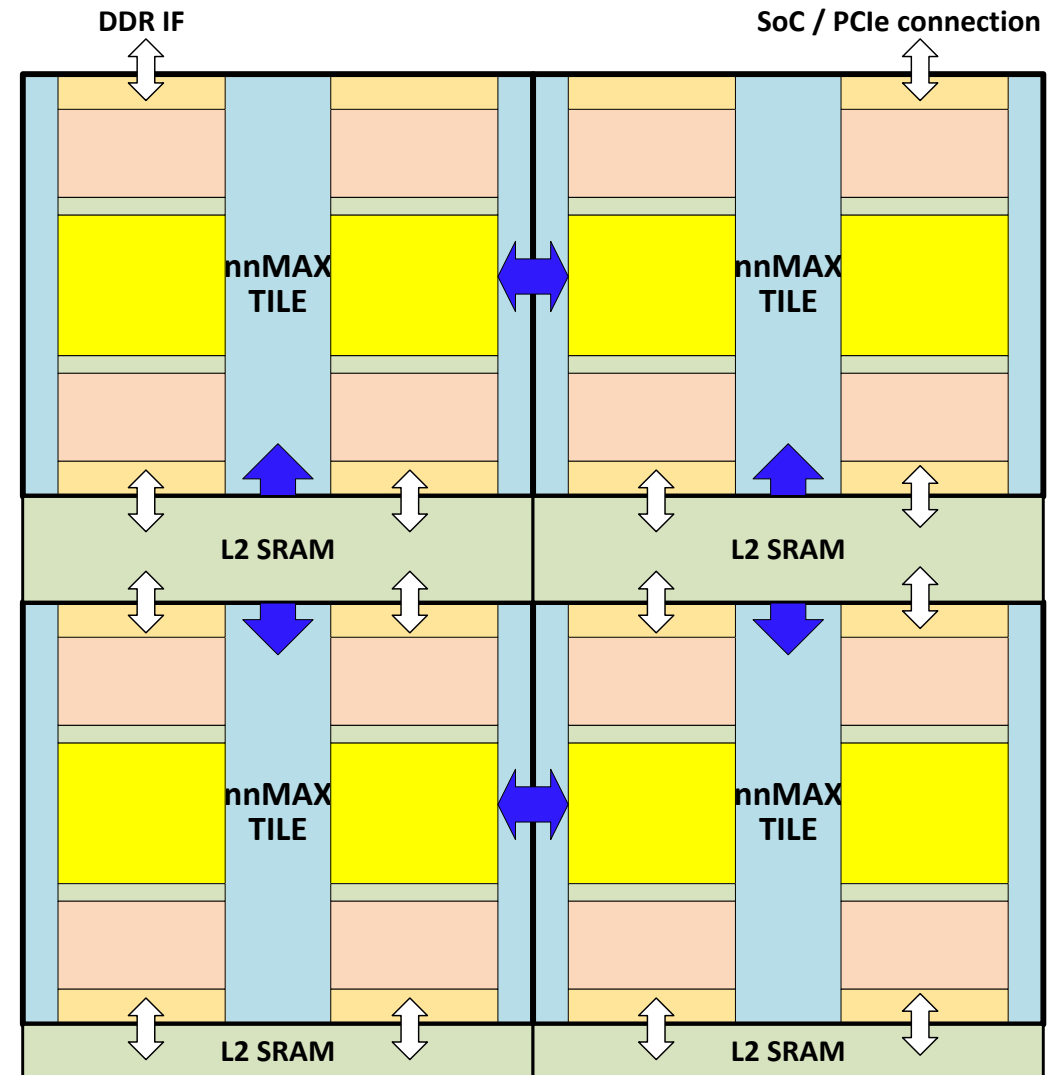
nnMAX™ 1K Tile for TSMC 16/12nm developed for AI Inference

- 4.5 mm² in TSMC16FFC
- 1024 configurable MACs @ 933MHz
 - INT8x8, INT16x8 at full throughput
 - BFloat16x16, INT16x16 at half throughput
 - Support mixed precision (INT8, INT16, BF16)
- Winograd acceleration for INT8
 - 2.25x performance gain for applicable layers
 - Automatically invoked by nnMAX Compiler
- Programmed by TensorFlow Lite/ONNX: multiple models can run simultaneously

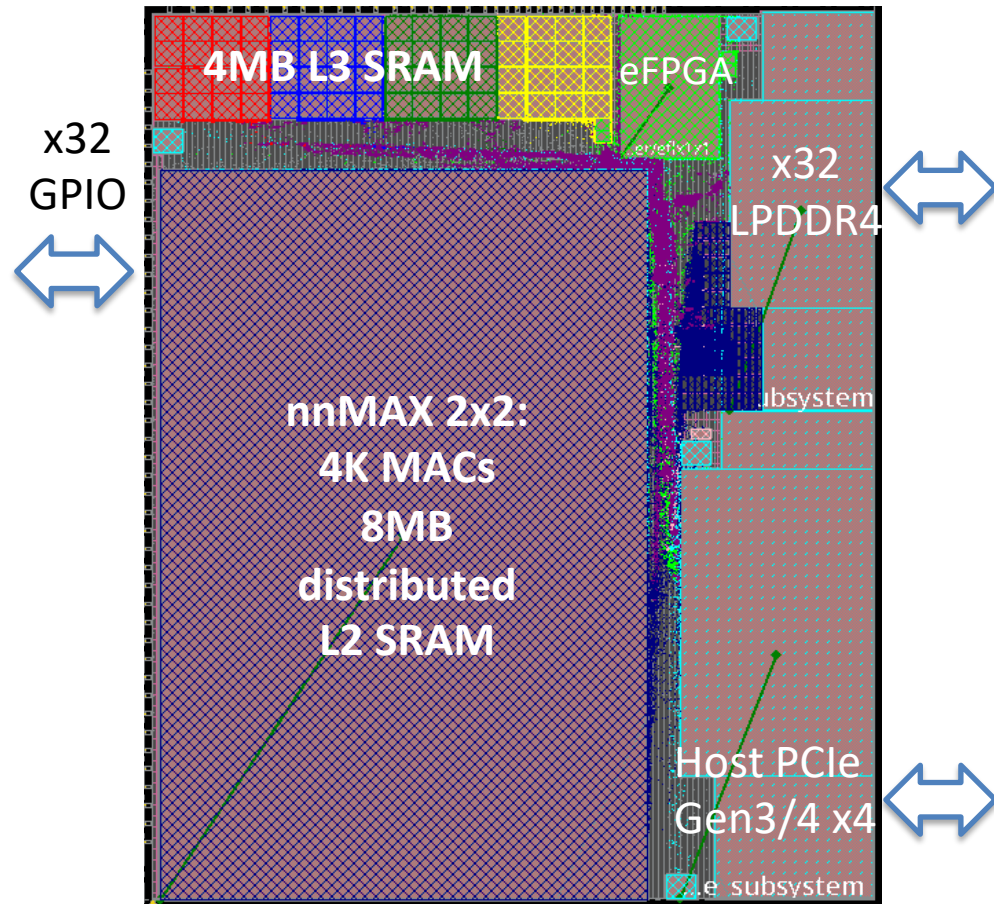


nnMAX tiles are Arrayed to provide more compute capacity

- ArrayLinx interconnect (blue) is a top level interconnect mesh between all tiles
 - This is used in our eFPGA and is silicon proven
- 2MB L2 SRAM attached to every Tile
- 2x2 array shown here; we have already fabricated 7x7 eFPGA arrays using ArrayLinx
- Linearly scalable: An NxN array has $\sim N^2$ the performance of a single tile



nnMAX is the foundation of the InferX™ X1 AI Inference Co-Processor



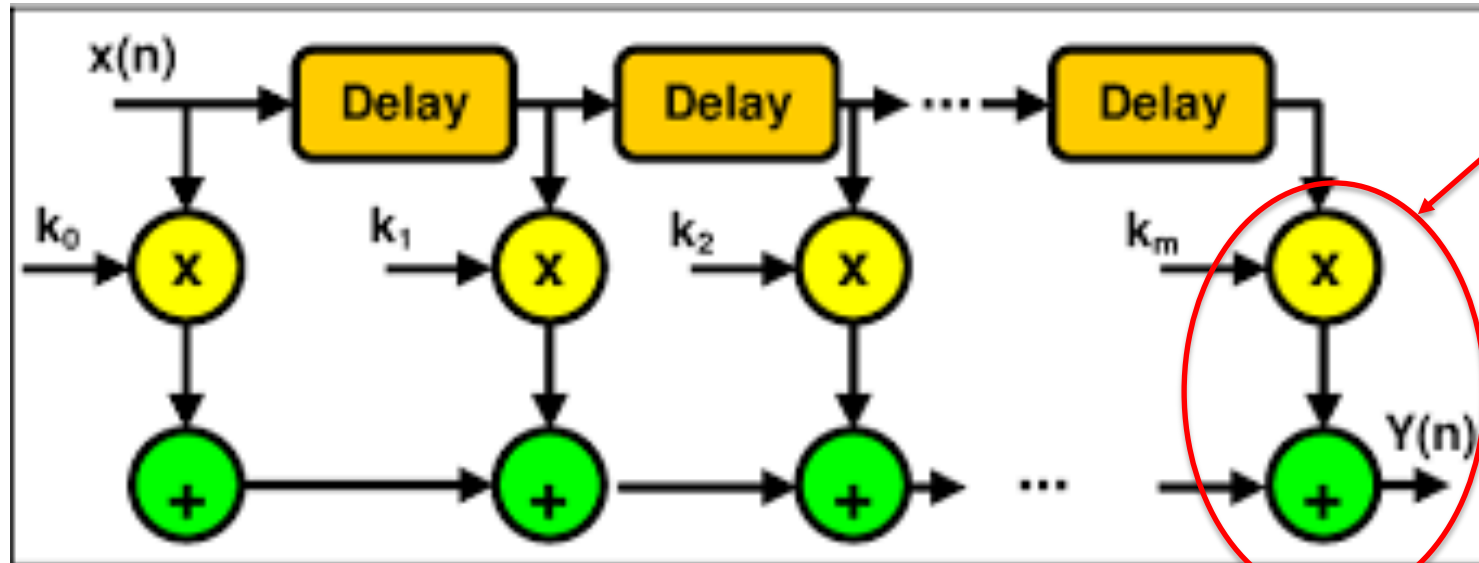
- 54mm² TSMC 16FFC
- 933MHz Operation
- Available as Chip & PCIe Board
- Samples Q3
- Partners: TSMC, GUC, Synopsys, Arteris, Analog Bits, Cadence, Mentor
- **THURSDAY 1110AM Talk:**
we benchmark X1 for Real-World Edge Inference Applications and compare to what customers use now

| Many customers are using expensive FPGAs for DSP

- Testers, 5G, Base Stations, Radar, Imaging, ...
 - High sample rate; large numbers of taps
- Using large, expensive FPGAs or expensive high end DSPs
 - As one customer says they buy the FPGA just for the MACs: they don't use the rest
- Many customers have asked us if nnMAX/X1 can do signal processing and are engaged giving us applications to model

FIR Filter: typically INT16 Real or Complex

Incoming data arrives at Z Megasamples/second



MACs need to run at the sample rate of Z

Outgoing data sent at Z Megasamples/second

Figure 1. Logical Structure of an FIR filter.

$X(n)$ is the incoming signal arriving at Z megasamples/second

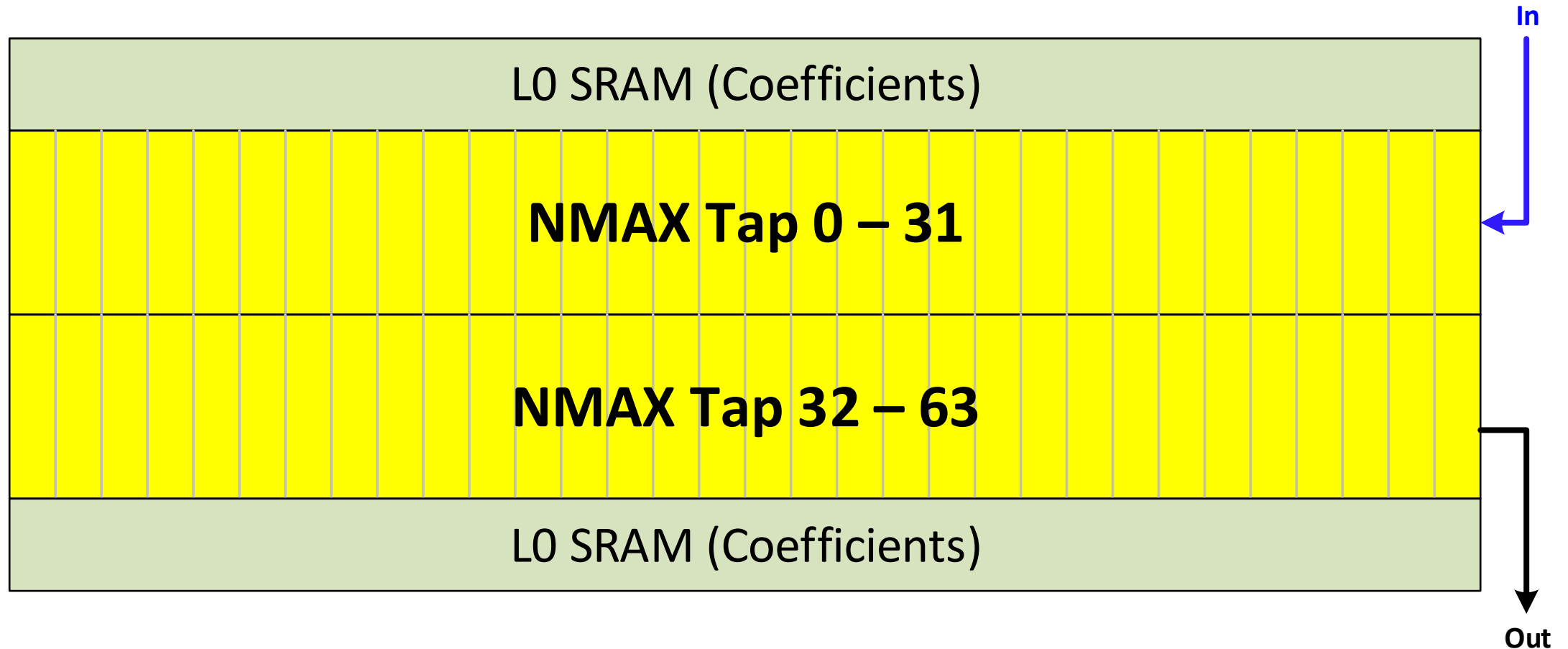
$K(m)$ is the tap or coefficient value

The number of taps can range from dozens to thousands

$Y(n)$ is the outgoing signal sent out at Z megasamples/second

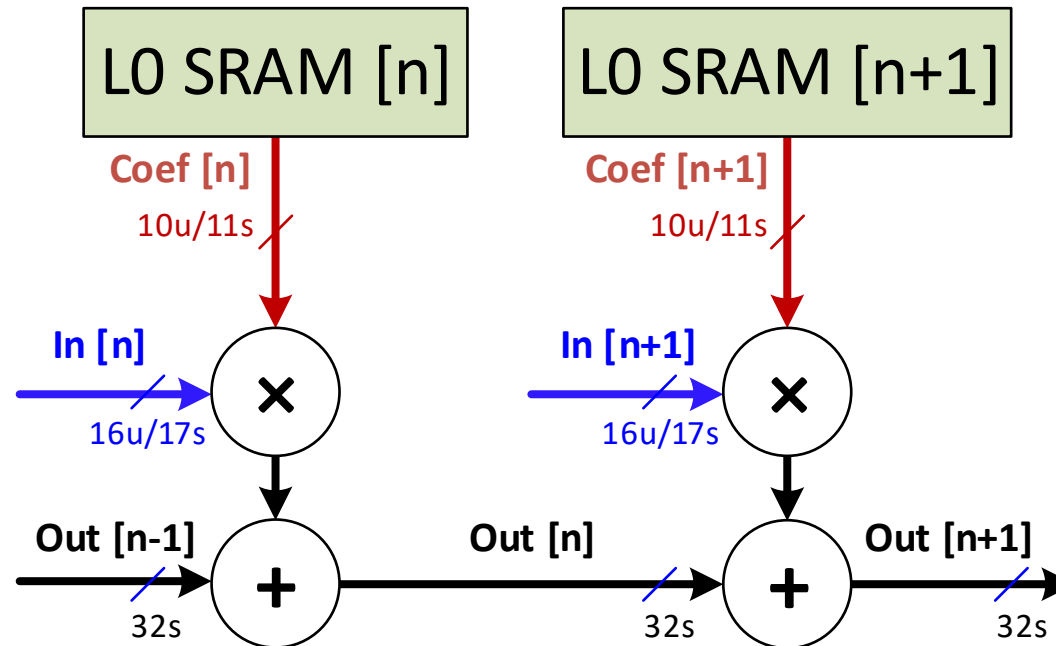
nnMAX cluster basic structure

- Each NMAX cluster can perform a 64 tap filter



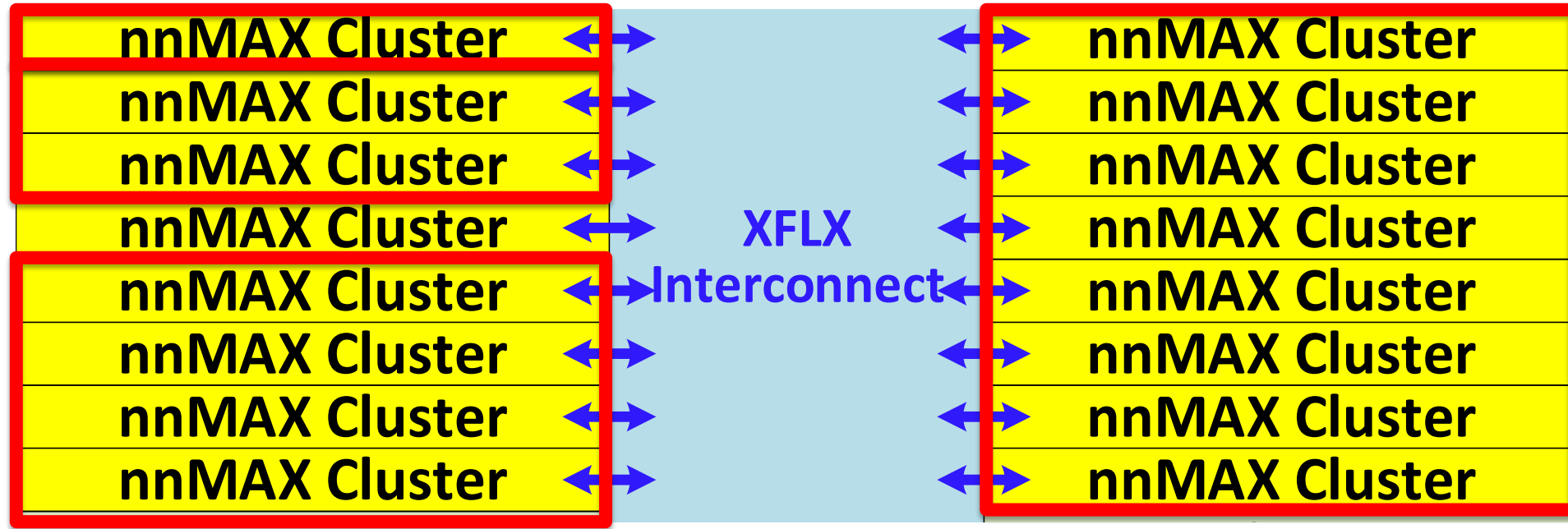
nnMAX has native precision of 16b x 10b INT, expandable to FP

- 10b of filter resolution is sufficient for most signal processing applications



- But 2 nnMAX coefficients can be combined to achieve even higher resolution:
 - 16b x 16b = 32b MAC (32 MAC per cluster)
 - BF16 x BF16 = FP24 MAC (32 MAC per cluster)

Chain nnMAX Clusters in nnMAX Tile for Longer FIR Filters



- Minimum FIR filter size is one cluster
- Maximum is all clusters in array (N×N tiles)
- Clusters can be linked across tiles for 1000's or 10,000+ taps

| Re-Configuring nnMAX

- nnMAX array can be reconfigured in ~ 2 μ seconds from one FIR configuration to any other using configuration files stored in the local DRAM
 - Coefficients are loaded into SRAM in the nnMAX clusters (part of the configuration file)

Two options to Map FIR Filters to nnMAX Clusters (Real INT16/BF16)

MegaSamples per second	nnMAX Cluster	nnMAX 1K Tile	nnMAX Array (2x2 tiles)
1,000 MS/s	16 Taps	256 Taps	1024 Taps
500 MS/s	32 Taps	512 Taps	2048 Taps

Trade-off between throughput and # of taps

Complex INT16/BF16 runs at $\frac{1}{2}$ the sample rate with $\frac{1}{4}$ of the taps shown above

Real INT16x10 has 2x the throughput (Taps*SampleRate)

Based on 1GHz clock rate

| nnMAX runs FIR filters of any # of taps faster & cheaper than Ultrascale

- 16-bit FIR, 21-taps, sample period = 1
 - Virtex UltraScale (20/16nm) Fmax = 633MHz
 - Virtex UltraScale+ (16/14nm) Fmax = 800MHz
 - Performance for >21 taps is likely 50%
 - An FPGA with 2000 MACs for 2000 Taps is 100's of mm² and 100's of \$\$
- nnMAX (16nm) runs at 800MHz/933MHz worst case conditions
 - An nnMAX 2x2 array can run 1000 Taps at the same rate as an Ultrascale 21-tap FIR
 - An nnMAX 2x2 array with 8MB SRAM is just 26mm²!!

| nnMAX is faster and cheaper than TI high end DSP

	Ti's Fast DSP: C6678	nnMAX 1K Tile
Cost	\$120 (1K quantity)	6.5mm ² in 16nm
FIR execution time: INT16, 16 taps, 128 samples	260nsec	128nsec <i>2x faster</i>
# Simultaneous FIRs	8	16
Scalable	?	Yes

| GigaOPS/sec (INT16x16) Compared to New CEVA XC16 DSP IP

	CEVA XC16	nnMAX 1K Tile
Process Node	7nm	16nm
Frequency	1.8GHz	~1GHz
Common FIR Operations	1600 GOPS/sec	2000 GOPs/sec <i>1.2x faster in a less expensive node</i>

| Roadmap for nnMAX for DSP Applications

- FIR filters are our focus for first applications
 - We are working with a major customer
 - We can generate the programs for initial customers for them
 - We expect to develop a DSP Compiler to take MATLAB output and map onto nnMAX
- nnMAX will be ported next to GF12LPP and TSMC N7/N6
 - nnMAX 1.1 will double the throughput of FIR filters
 - We are evaluating changes for very very fast FFT, much faster than Ultrascale
 - we can share performance estimates under NDA

| Conclusion

- nnMAX IP, great for AI Inference, is also higher throughput/\$ and throughput/watt for key DSP functions
- If you are interested, join our Breakout Room at Noon or email me: cheng@flex-logix.com