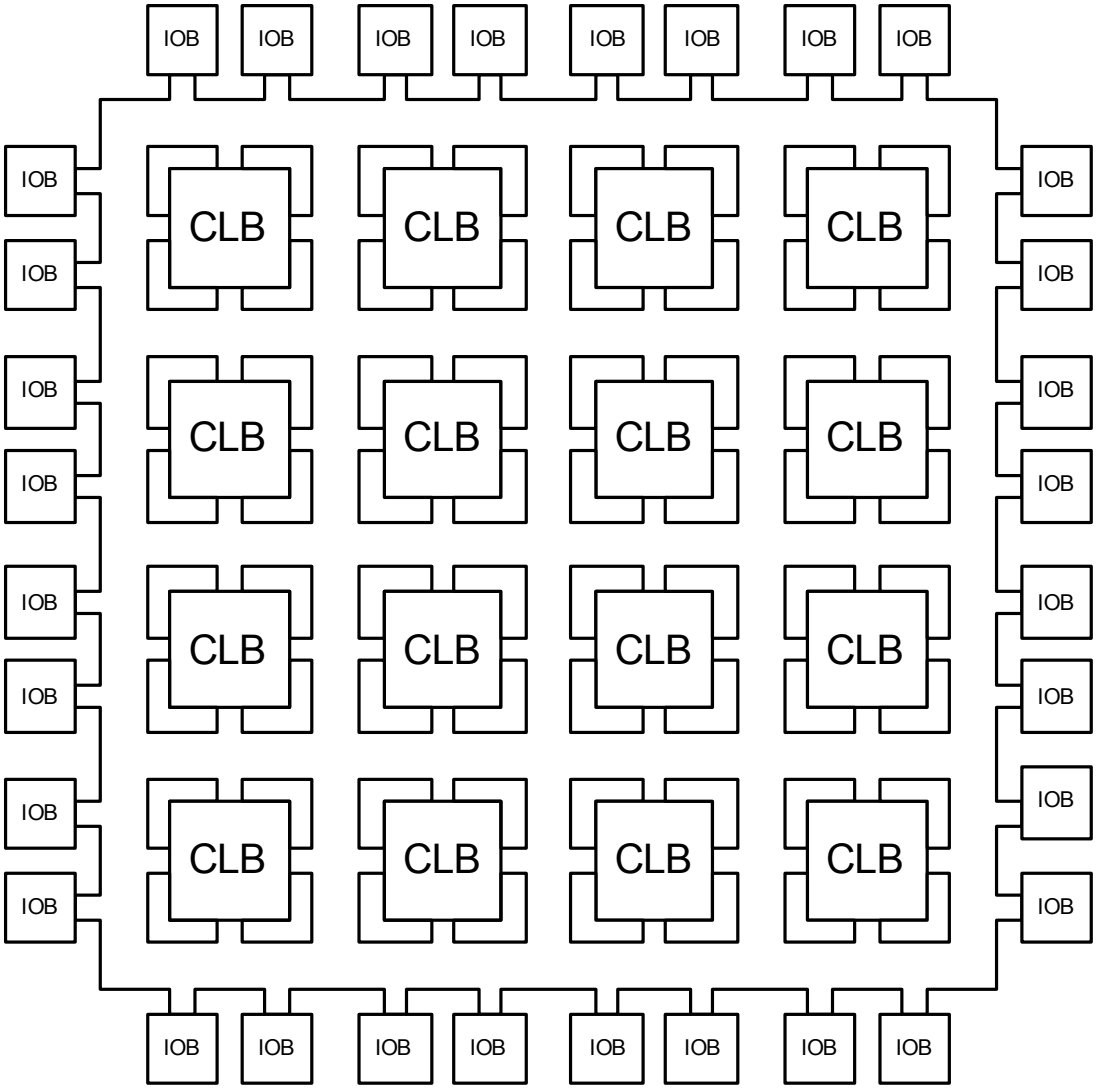
FPGA architectures for acceleration of deep learning inference

Field Programmable Gate Array (FPGA)

- Configurable Logic Block (CLB)
  - Look-up table (LUT)
  - Register
  - Logic circuit
    - Adder
    - Multiplier
    - Memory
    - Microprocessor
- Input/Output Block (IOB)
- Programmable interconnect



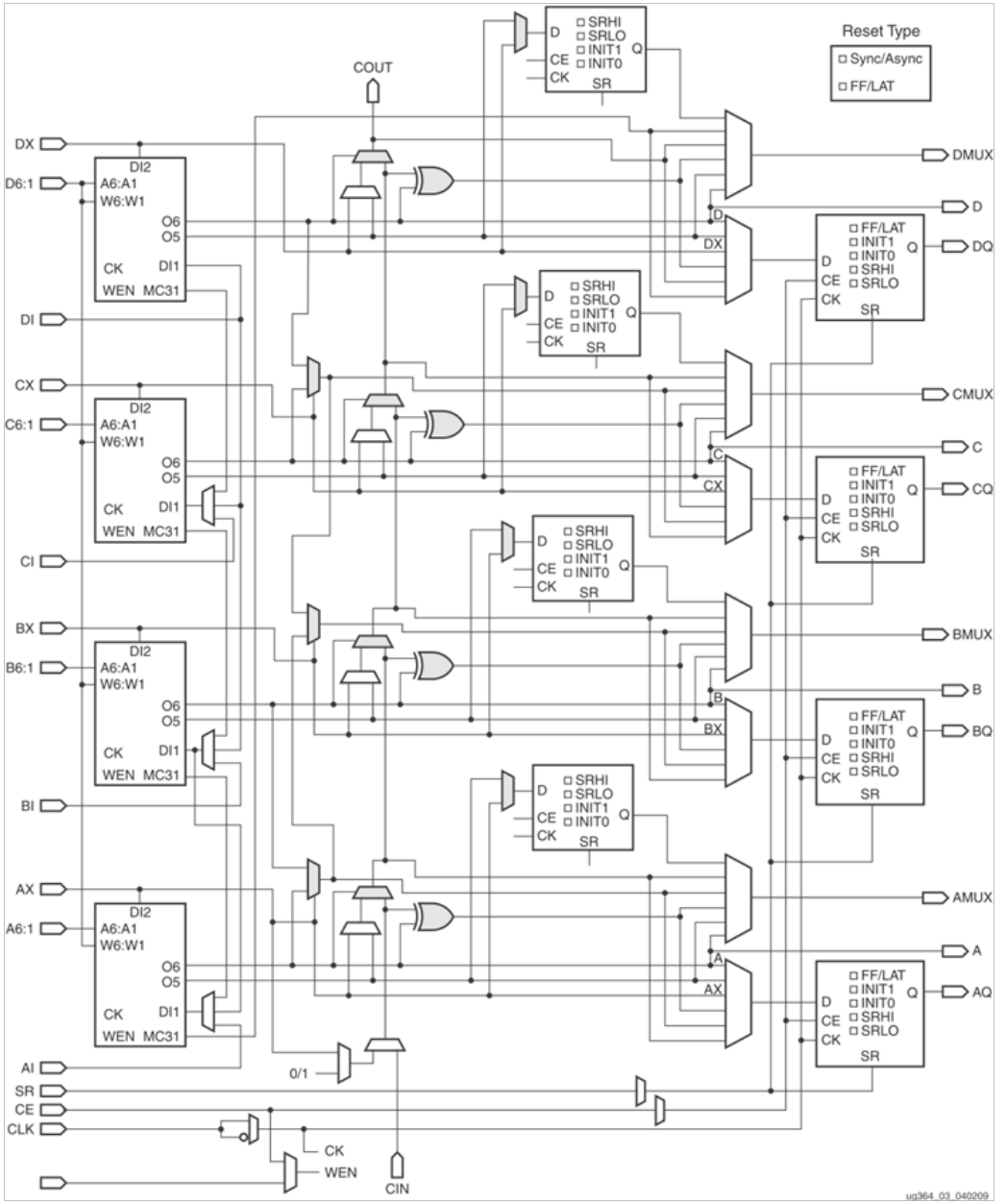
Xilinx FPGA families

- High performance
  - Virtex (1998)
    - 768-12.2K LC, 0.22µm
  - Virtex-E/EM (1999)
    - 768-32.4K LC, 0.18µm
  - Virtex-II (2000)
    - 512-93.1K LC, 0.15µm
  - Virtex-II Pro/X (2002)
    - 2.8K-111.2K LC, 0.13µm
  - Virtex-4 (2004) [LX, FX, SX]
    - 12.2K-178.1K LC, 90nm
  - Virtex-5 (2006) [LX, LXT, SXT]
    - 19.2-207.3K LC, 65nm
  - Virtex-6 (2009) [LXT, SXT, HXT]
    - 46.5K-474.2K LC, 40nm
  - Virtex-7 (2010)
    - 178.8K-1.22M LC, 28nm
  - Virtex UltraScale
    - 358K-2.53M LC, 20nm
  - Virtex UltraScale+
    - 394K-1.728M LC, 16nm
- Low cost
  - Spartan-II (2000)
    - 0.22µm
  - Spartan-IIE (2001)
    - 0.18µm
  - Spartan-3 (2003)
    - 1.5K-66.5K LC, 90nm
  - Spartan-3E (2005)
    - 1.9K-29.5K LC, 90nm
  - Spartan-6 (2009)
    - 1.5K-92.1K LC, 45nm
  - Artix-7 (2010)
    - 11.2K-220K LC, 28nm
- Mid-range
  - Kintex-7 (2010)
    - 19K-254K LC, 28nm
  - Kintex UltraScale
    - 145.4K-663K LC, 20nm
  - Kintex UltraScale+
    - 163K-523K LC, 16nm

http://www.xilinx.com

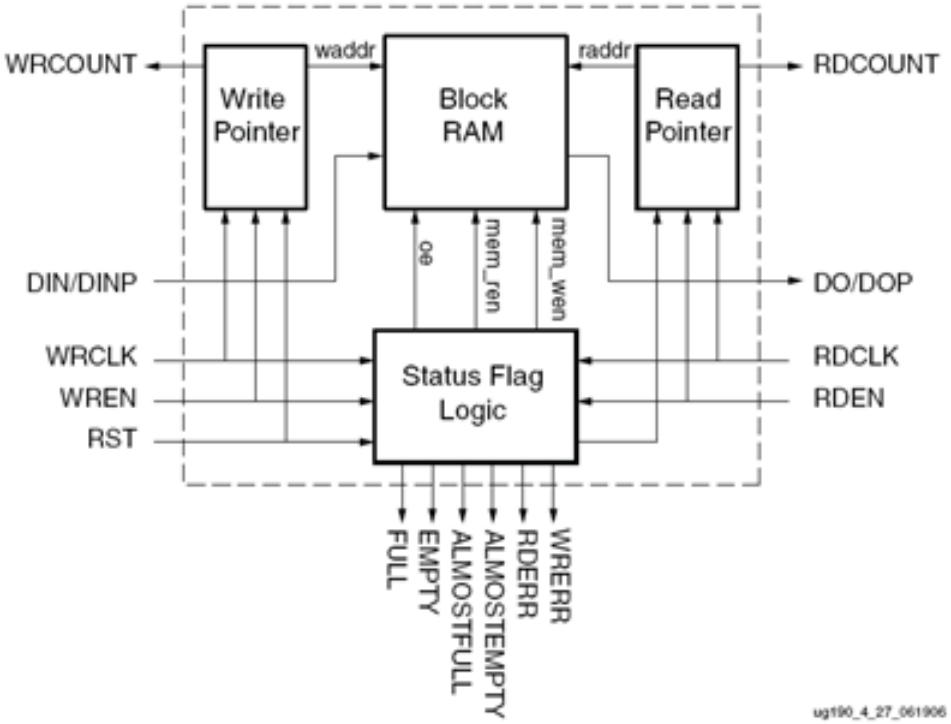
Xilinx 7 Series FPGA Logic resources

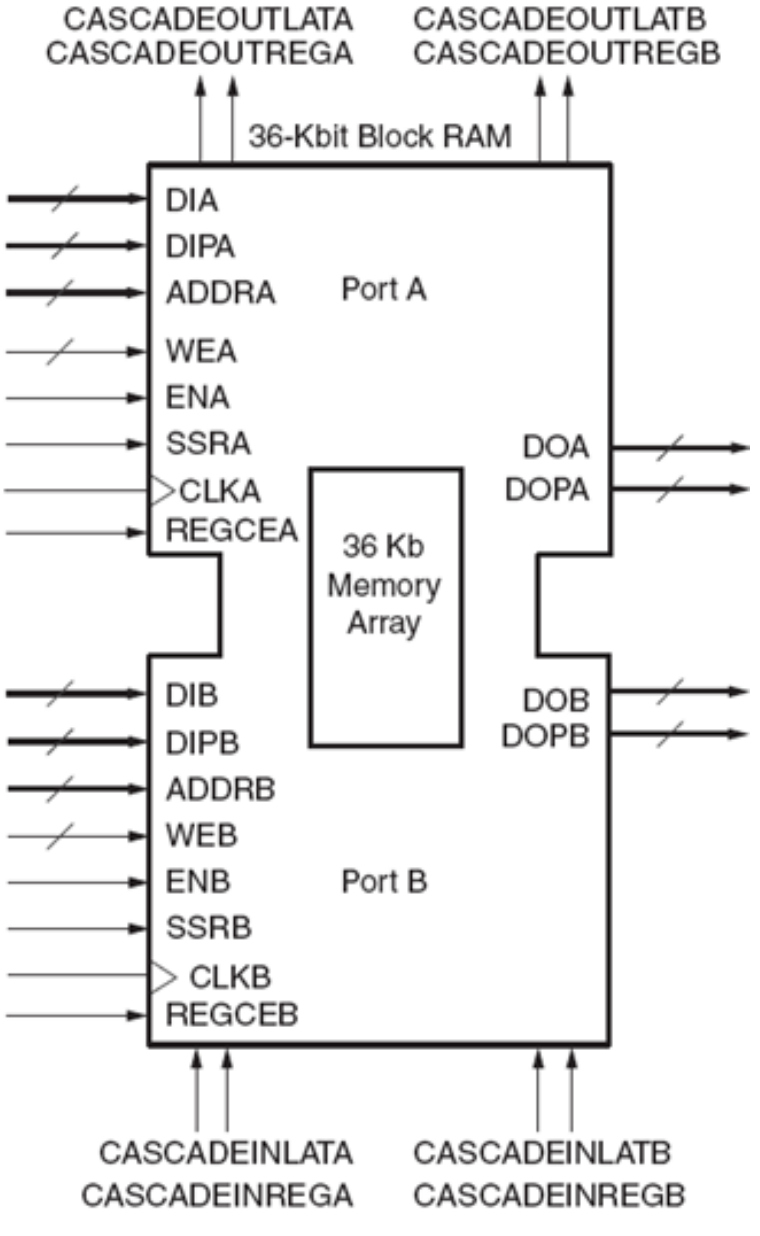
- 6 input LUT
  - 5 input 2 output LUT
  - 32bit shift register
  - 64bit RAM
- 8 register
- 4 carry logic
- 2:1 multiplexers
  - 7, 8 input LUT
  - 128, 256 bit RAM
  - 8:1, 16:1 multiplexer



Xilinx 7 Series FPGA Dedicated resources, Block RAM

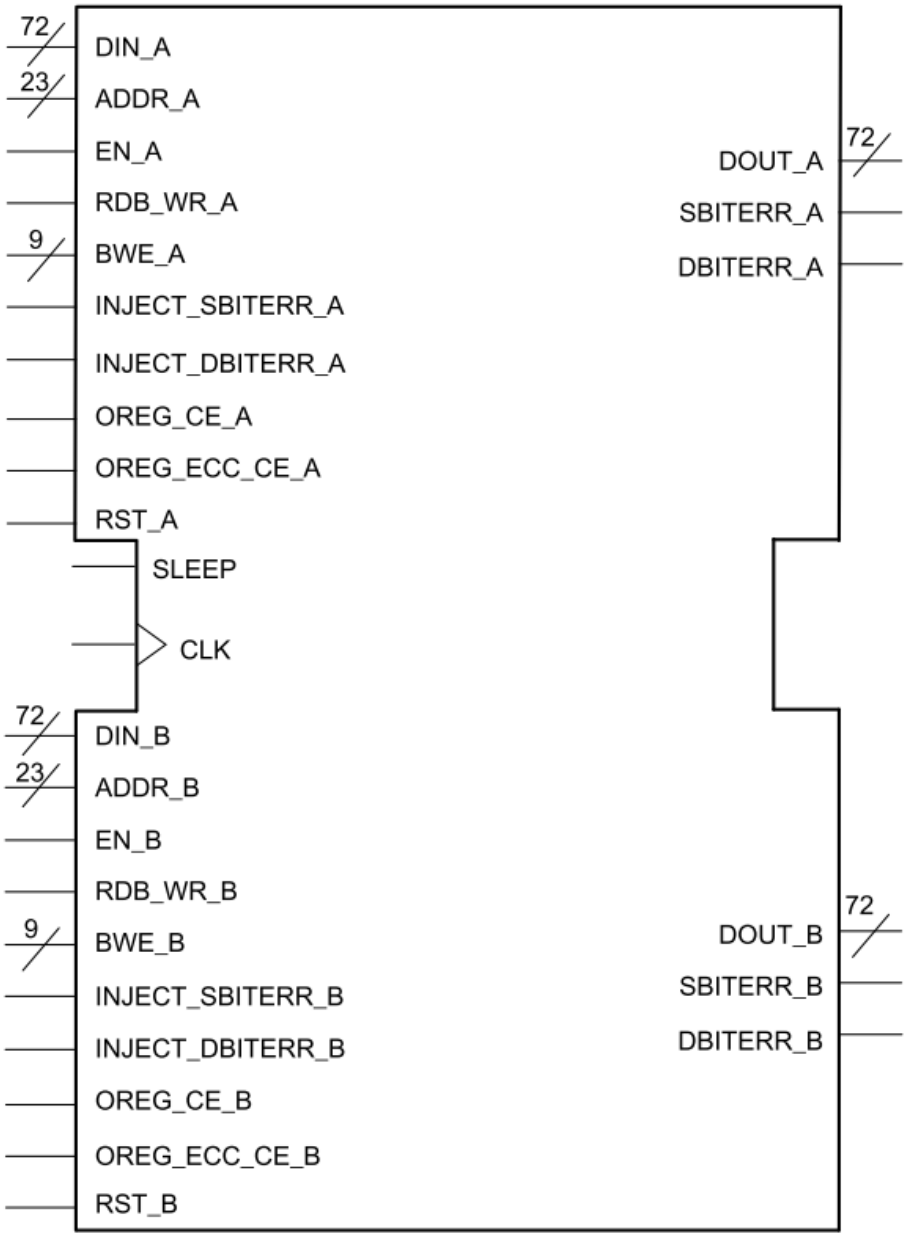
- 36kbit dual ported SRAM (32k x 1 - 512 x 72)
- Can be configured as two 18kbit independent SRAM
- Dedicated FIFO logic





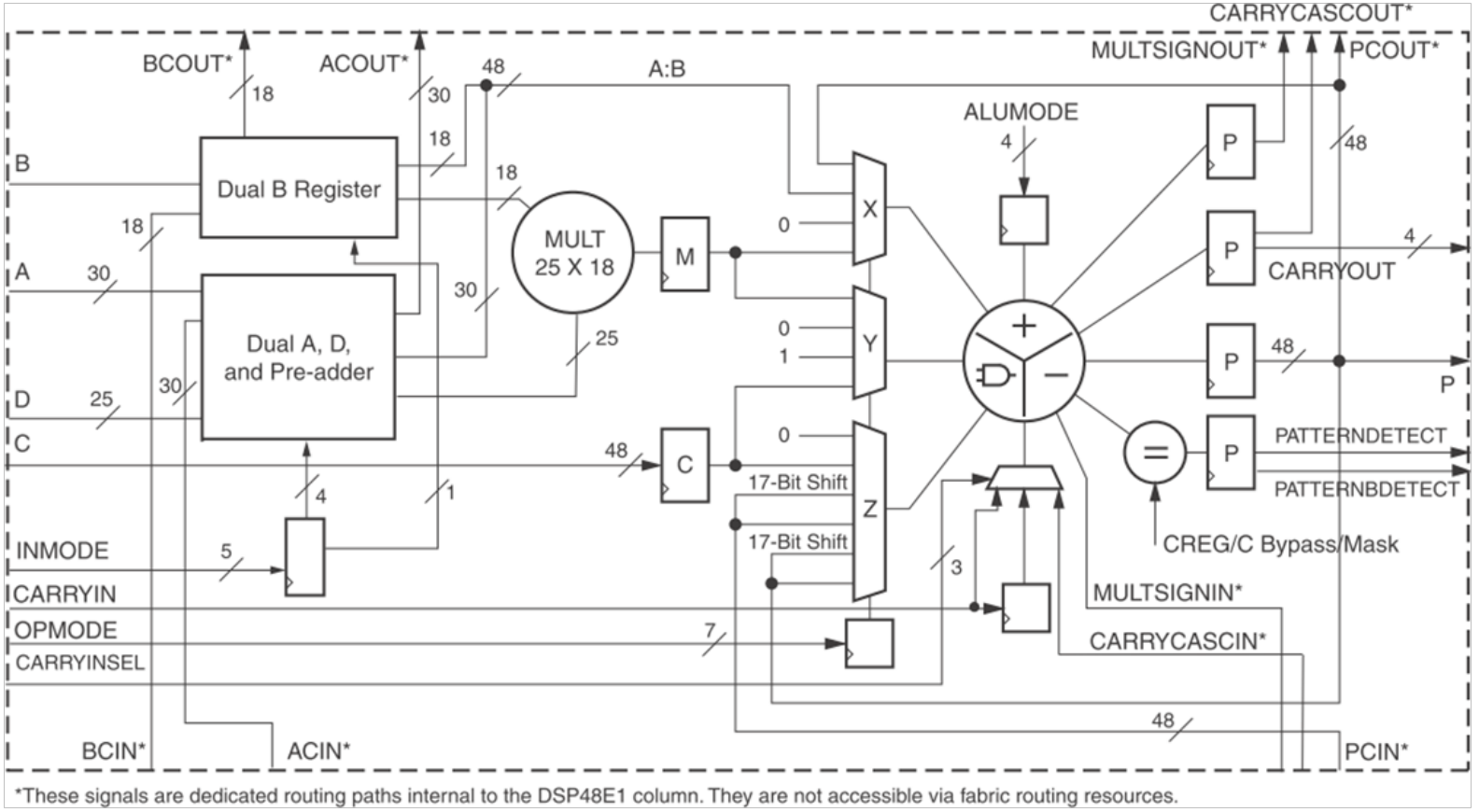
Xilinx UltraScale+ FPGA Dedicated resources, Ultra RAM

- 288kbit dual ported memory
- 4k x 72bit configuration
- Single clock
- Cascadable
- Reset to 0



Xilinx 7 Series FPGA Dedicated resources

- DSP48E1 block
  - 25x18bit signed multiplier (27x18 for Ultrascale+)
  - 48bit adder/accumulator, ALU



Virtex-7 Family

Device[1]	Logic Cells	Configurable Logic Blocks (CLBs)		DSP Slices[3]	Block RAM Blocks[4]			CMTs[5]	PCIe[6]	GTX	GTH	GTZ	XADC Blocks	Total I/O Banks[7]	Max User I/O[8]	SLRs[9]
		Slices[2]	Max Distributed RAM (Kb)		18 Kb	36 Kb	Max (Kb)									
XC7V585T	582,720	91,050	6,938	1,260	1,590	795	28,620	18	3	36	0	0	1	17	850	N/A
XC7V2000T	1,954,560	305,400	21,550	2,160	2,584	1,292	46,512	24	4	36	0	0	1	24	1,200	4
XC7VX330T	326,400	51,000	4,388	1,120	1,500	750	27,000	14	2	0	28	0	1	14	700	N/A
XC7VX415T	412,160	64,400	6,525	2,160	1,760	880	31,680	12	2	0	48	0	1	12	600	N/A
XC7VX485T	485,760	75,900	8,175	2,800	2,060	1,030	37,080	14	4	56	0	0	1	14	700	N/A
XC7VX550T	554,240	86,600	8,725	2,880	2,360	1,180	42,480	20	2	0	80	0	1	16	600	N/A
XC7VX690T	693,120	108,300	10,888	3,600	2,940	1,470	52,920	20	3	0	80	0	1	20	1,000	N/A
XC7VX980T	979,200	153,000	13,838	3,600	3,000	1,500	54,000	18	3	0	72	0	1	18	900	N/A
XC7VX1140T	1,139,200	178,000	17,700	3,360	3,760	1,880	67,680	24	4	0	96	0	1	22	1,100	4
XC7VH580T	580,480	90,700	8,850	1,680	1,880	940	33,840	12	2	0	48	8	1	12	600	2
XC7VH870T	876,160	136,900	13,275	2,520	2,820	1,410	50,760	18	3	0	72	16	1	6	300	3

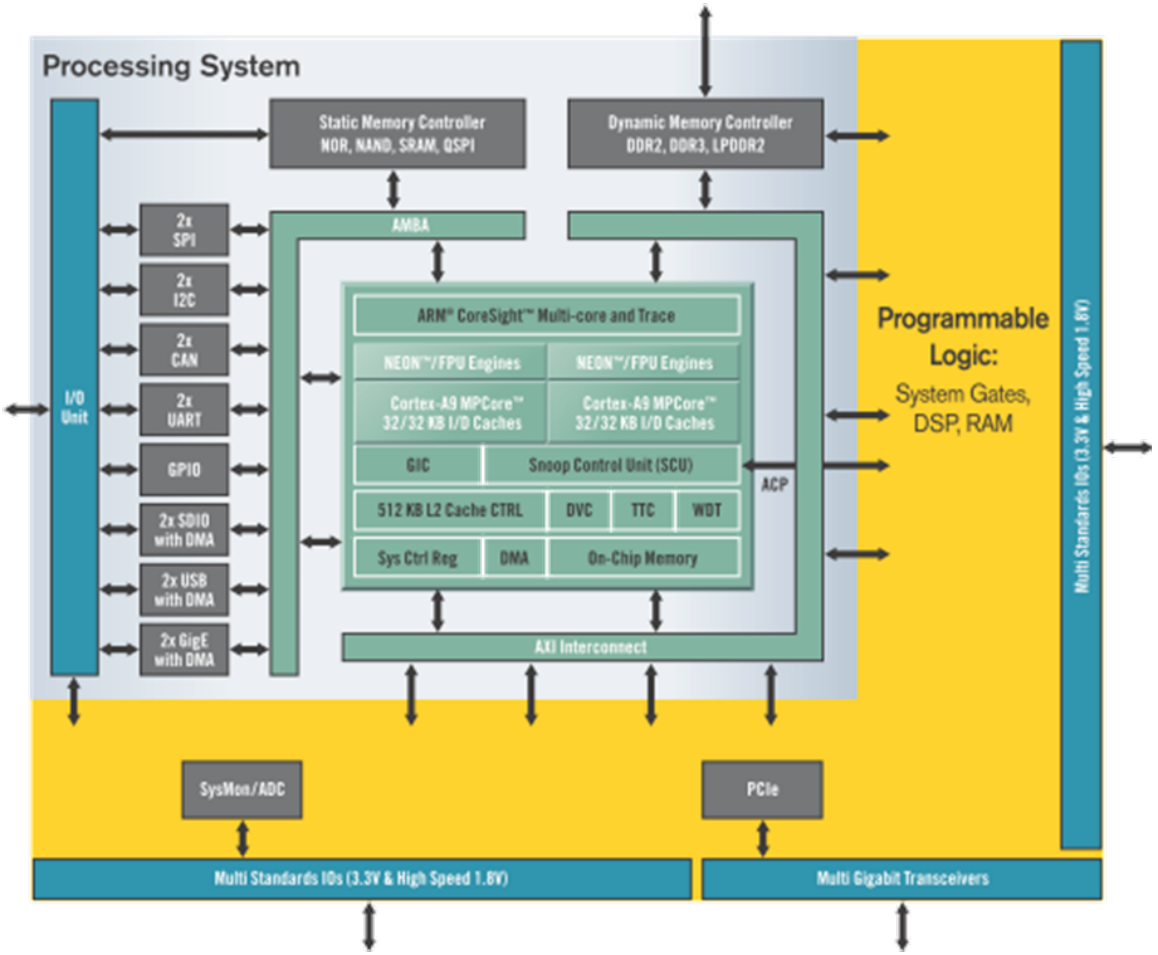
Virtex Ultrascale+

.

	VU3P	VU5P	VU7P	VU9P	VU11P	VU13P	VU31P	VU33P	VU35P	VU37P
System Logic Cells	862,050	1,313,763	1,724,100	2,586,150	2,835,000	3,780,000	961,800	961,800	1,906,800	2,851,800
CLB Flip-Flops	788,160	1,201,154	1,576,320	2,364,480	2,592,000	3,456,000	879,360	879,360	1,743,360	2,607,360
CLB LUTs	394,080	600,577	788,160	1,182,240	1,296,000	1,728,000	439,680	439,680	871,680	1,303,680
Max. Distributed RAM (Mb)	12.0	18.3	24.1	36.1	36.2	48.3	12.5	12.5	24.6	36.7
Block RAM Blocks	720	1,024	1,440	2,160	2,016	2,688	672	672	1,344	2,016
Block RAM (Mb)	25.3	36.0	50.6	75.9	70.9	94.5	23.6	23.6	47.3	70.9
UltraRAM Blocks	320	470	640	960	960	1,280	320	320	640	960
UltraRAM (Mb)	90.0	132.2	180.0	270.0	270.0	360.0	90.0	90.0	180.0	270.0
HBM DRAM (GB)	–	–	–	–	–	–	4	8	8	8
CMTs (1 MMCM and 2 PLLs)	10	20	20	30	12	16	4	4	8	12
Max. HP I/O(1)	520	832	832	832	624	832	208	208	416	624
DSP Slices	2,280	3,474	4,560	6,840	9,216	12,288	2,880	2,880	5,952	9,024
System Monitor	1	2	2	3	3	4	1	1	2	3
GTY Transceivers 32.75Gb/s(2)	40	80	80	120	96	128	32	32	64	96
Transceiver Fractional PLLs	20	40	40	60	48	64	16	16	32	48
PCIe Gen3 x16 and Gen4 x8	2	4	4	6	3	4	4	4	5	6
CCIX Ports(3)	–	–	–	–	–	–	4	4	4	4
150G Interlaken	3	4	6	9	6	8	0	0	2	4
100G Ethernet w/RS-FEC	3	4	6	9	9	12	2	2	5	8

Zynq-7000 All Programmable SoC

- Feature-rich dual-core or single-core ARM Cortex-A9 based processing system (PS)
- Programmable logic (PL)
- In a single device.



Processing System (PS)

- Dual-core ARM® CortexTM-A9 Based Application Processor Unit (APU)
  - 2.5 DMIPS/MHz per CPU
  - CPU frequency: Up to 1 GHz
  - Coherent multiprocessor support
  - ARMv7-A architecture
  - NEONTM media-processing engine
  - Single and double precision Vector Floating Point Unit (VFPU)
  - Timer and Interrupts
    - Three watchdog timers
    - One global timer
    - Two triple-timer counters
- Caches
  - 32 KB Level 1 4-way set-associative instruction and data caches (independent for each CPU)
  - 512 KB 8-way set-associative Level 2 cache (shared between the CPUs)
  - Byte-parity support
- On-Chip Memory
  - On-chip boot ROM
  - 256 KB on-chip RAM (OCM)
  - Byte-parity support
- External Memory Interfaces
  - 16-bit or 32-bit interfaces to DDR3, DDR3L, DDR2, or LPDDR2 memories
  - ECC support in 16-bit mode
  - 1GB of address space using single rank of 8-, 16-, or 32-bit-wide memories

Processing System (PS)

- Static memory interfaces
  - 8-bit SRAM data bus with up to 64 MB support
  - Parallel NOR flash support
  - ONFI1.0 NAND flash support (1-bit ECC)
  - 1-bit SPI, 2-bit SPI, 4-bit SPI (quad-SPI), or two quad-SPI (8-bit)
  - serial NOR flash
- 8-Channel DMA Controller
  - Memory-to-memory
  - Memory-to-peripheral
  - Peripheral-to-memory
  - Scatter-gather transaction support
- Interconnect
  - High-bandwidth connectivity within PS and between PS and PL
  - ARM AMBA® AXI based
- I/O Peripherals and Interfaces
  - Two 10/100/1000 tri-speed Ethernet MAC peripherals
  - Two USB 2.0 OTG peripherals, each supporting up to 12 Endpoints
  - Two full CAN 2.0B compliant CAN bus interfaces
  - Two SD/SDIO 2.0/MMC3.31 compliant controllers
  - Two full-duplex SPI ports with three peripheral chip selects
  - Two high-speed UARTs (up to 1 Mb/s)
  - Two master and slave I2C interfaces
  - GPIO with four 32-bit banks
- Up to 54 flexible multiplexed I/O (MIO) for peripheral pin assignments

Zynq-7000 Family

	Device Name	Z-7007S	Z-7012S	Z-7014S	Z-7010	Z-7015	Z-7020	Z-7030	Z-7035	Z-7045	Z-7100
	Part Number	**XC7Z007S**	**XC7Z012S**	**XC7Z014S**	**XC7Z010**	**XC7Z015**	**XC7Z020**	**XC7Z030**	**XC7Z035**	**XC7Z045**	**XC7Z100**
Programmable Logic	Xilinx 7 Series Programmable Logic Equivalent	Artix®-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Kintex®-7 FPGA	Kintex-7 FPGA	Kintex-7 FPGA	Kintex-7 FPGA
	Programmable Logic Cells	23K	55K	65K	28K	74K	85K	125K	275K	350K	444K
	Look-Up Tables (LUTs)	14,400	34,400	40,600	17,600	46,200	53,200	78,600	171,900	218,600	277,400
	Flip-Flops	28,800	68,800	81,200	35,200	92,400	106,400	157,200	343,800	437,200	554,800
	Block RAM (# 36 Kb Blocks)	1.8 Mb (50)	2.5 Mb (72)	3.8 Mb (107)	2.1 Mb (60)	3.3 Mb (95)	4.9 Mb (140)	9.3 Mb (265)	17.6 Mb (500)	19.1 Mb (545)	26.5 Mb (755)
	DSP Slices (18x25 MACCs)	66	120	170	80	160	220	400	900	900	2,020
	Peak DSP Performance (Symmetric FIR)	73 GMACs	131 GMACs	187 GMACs	100 GMACs	200 GMACs	276 GMACs	593 GMACs	1,334 GMACs	1,334 GMACs	2,622 GMACs
	PCI Express (Root Complex or Endpoint)[3]		Gen2 x4			Gen2 x4		Gen2 x4	Gen2 x8	Gen2 x8	Gen2 x8
	Analog Mixed Signal (AMS) / XADC	2x 12 bit, MSPS ADCs with up to 17 Differential Inputs									
	Security[2]	AES and SHA 256b for Boot Code and Programmable Logic Configuration, Decryption, and Authentication									

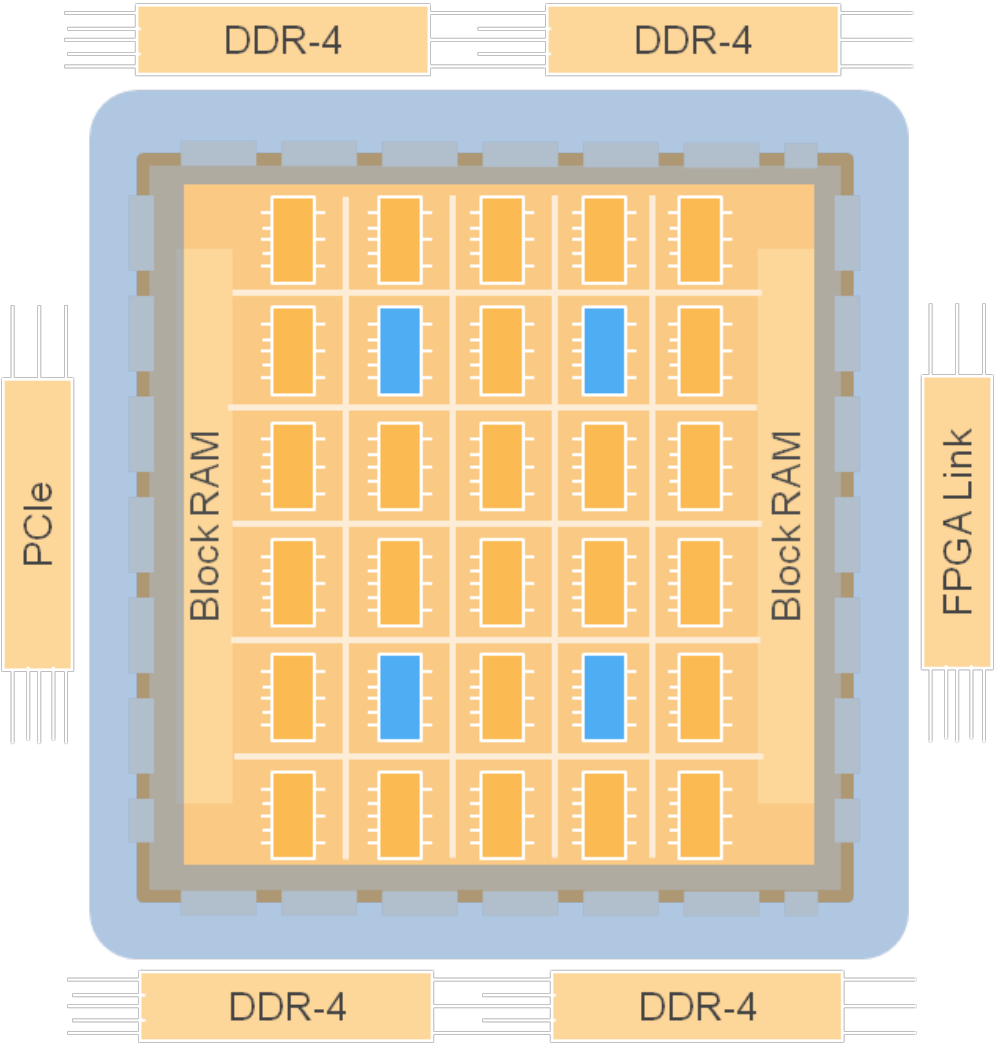
F1 FPGA Instance Types on AWS

- New EC2 FPGA instance type for accelerated computing
- Up to 8 Xilinx UltraScale+ 16nm VU9P FPGA devices in a single instance
- The **f1.16xlarge** size provides:
  - 8 FPGAs, each with over 2 million customer-accessible FPGA programmable logic cells and over 5000 programmable DSP blocks
  - Each of the 8 FPGAs has 4 DDR-4 interfaces, with each interface accessing a 16GiB, 72-bit wide, ECC-protected memory, 2400MHz operating frequency, 19.2Gbyte/s/interface, 76.8Gbyte/s

Instance Size	FPGAs	DDR-4 (GiB)	FPGA Link	FPGA Direct	vCPUs	Instance Memory (GiB)	NVMe Instance Storage (GB)	Network Bandwidth*
f1.2xlarge	1	4 x 16	-	-	8	122	1 x 480	10 Gbps Peak
f1.16xlarge	8	32 x 16	Y	Y	64	976	4 x 960	30 Gbps

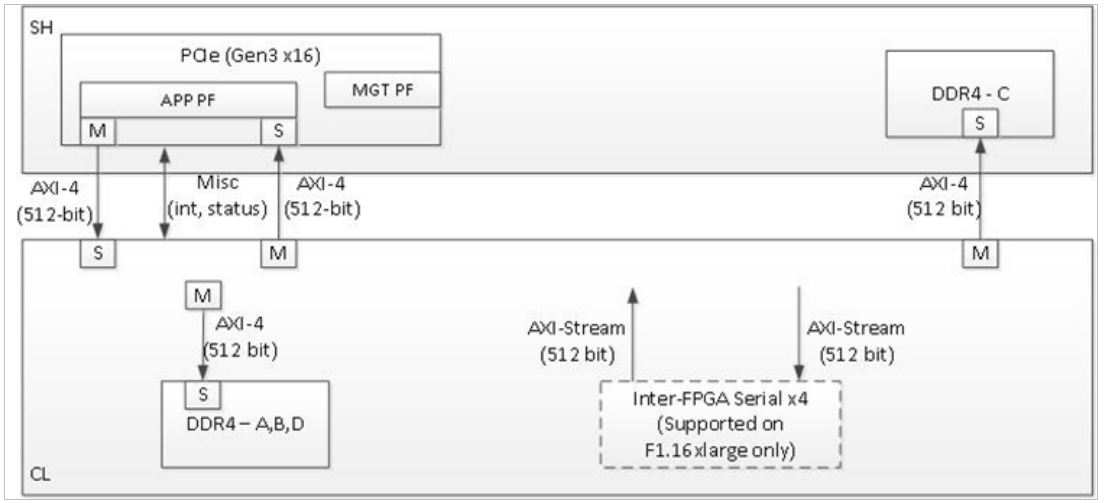
*In a placement group

Abstracting FPGA I/O

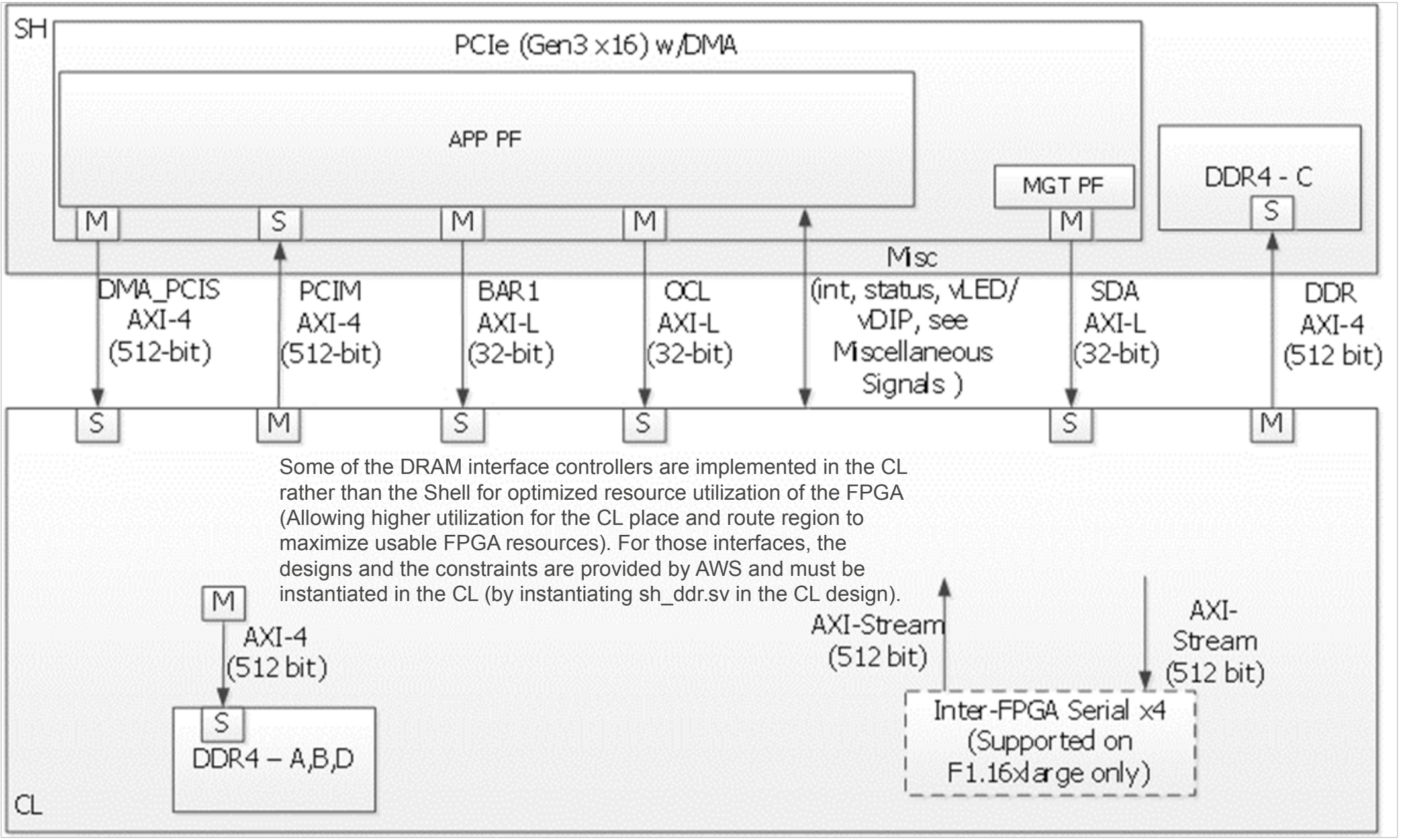


AWS FPGA Shell
FPGA I/O is provided using pre-configured, pre-tested, and secure I/O components, allowing FPGA developers to focus on their differentiating value

The FPGA Shell allows for faster coding of core acceleration functions by removing the need to develop I/O related FPGA hardware



FPGA Shell and FPGA Custom Logic

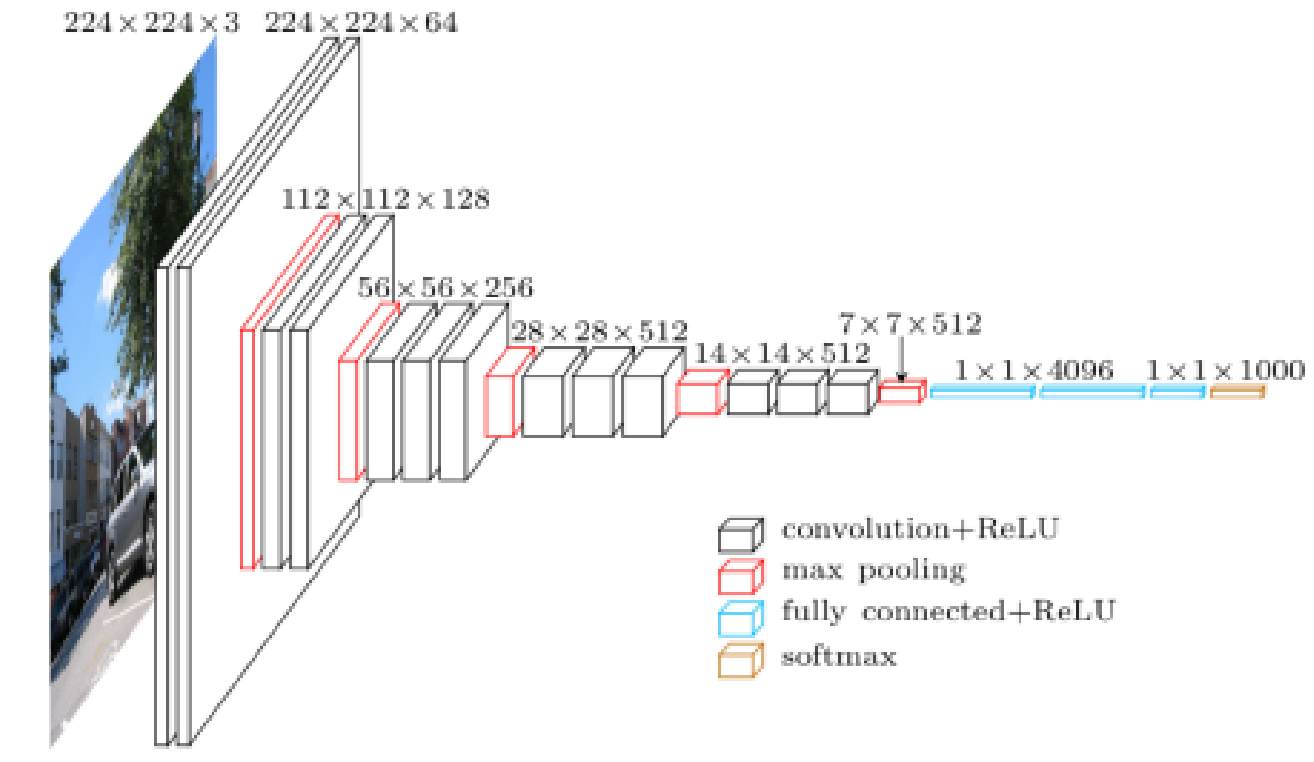


FPGA development process

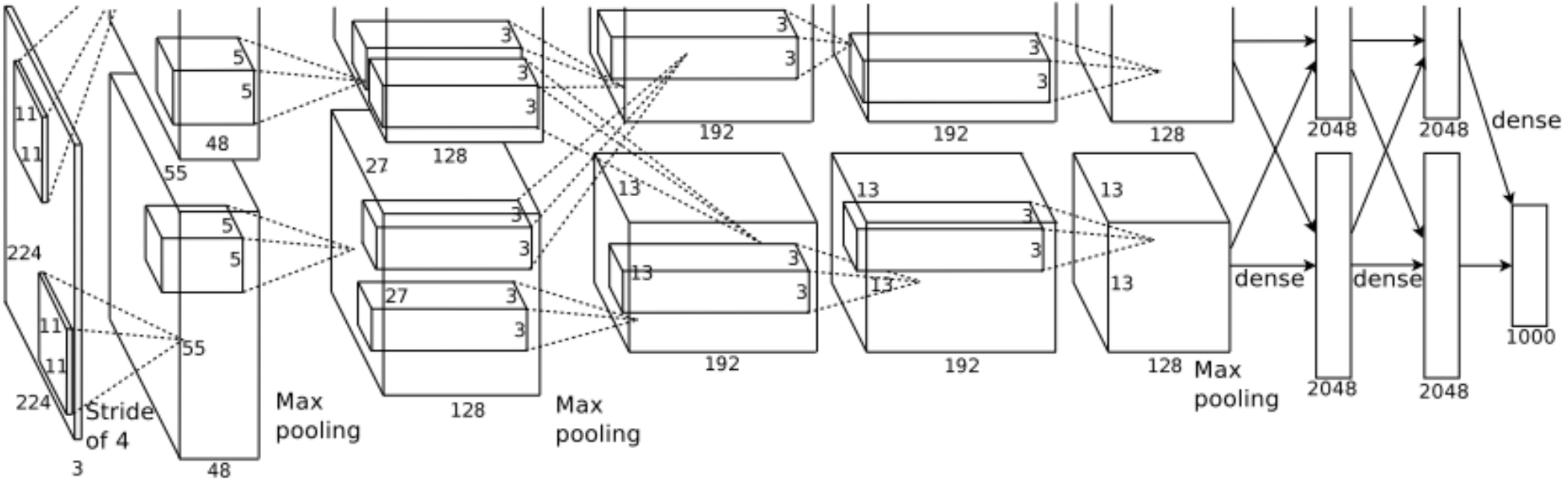
- Traditional design process
- Input language:
  - VHDL
  - Verilog
- Register Transfer Level description
- Early architectural decisions
- Development time: several months

- High-Level Synthesis
- Input language:
  - C/C++
  - OpenCL
- Pure ANSI C/C++
- Optimization directives
- Architecture exploration
- Development time: several days

VGG-16/19


224 × 224 × 3
224 × 224 × 64
112 × 112 × 128
56 × 56 × 256
28 × 28 × 512
14 × 14 × 512
7 × 7 × 512
1 × 1 × 4096
1 × 1 × 1000
convolution+ReLU
max pooling
fully connected+ReLU
softmax

AlexNet

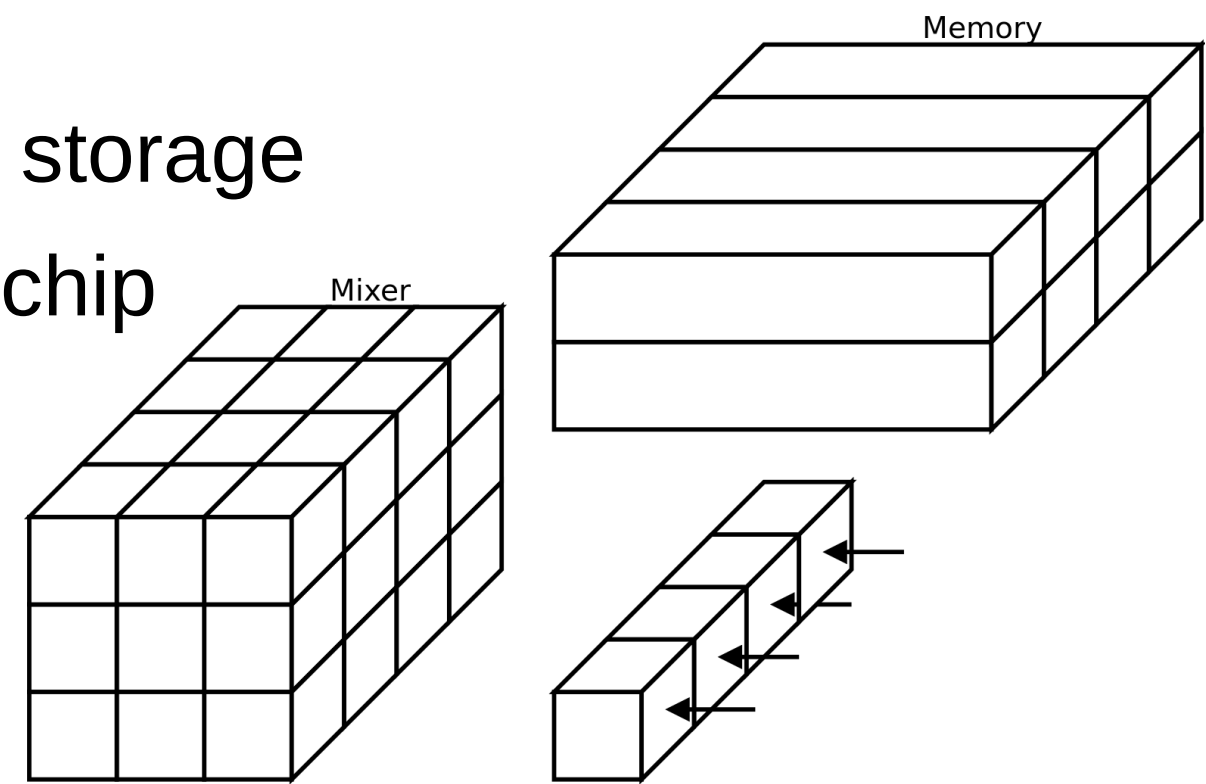

224
224
11
11
3
Stride
of 4
55
55
5
5
48
48
Max
pooling
27
27
3
3
128
128
Max
pooling
13
13
3
3
192
192
13
13
3
3
192
192
13
13
3
3
128
128
Max
pooling
dense
2048
2048
dense
2048
2048
dense
1000

Simple CNN network

- Input: gray scale 24x24x1
- Conv1: 3x3x4, output: 24x24x4
- Pool1: output: 8x8x4
- Conv2: 3x3x4, output: 8x8x4
- Conv3: 3x3x4, output: 10x10x4
- Pool2: output: 3x3x4
- Conv4: 3x3x4, output: 5x5x4
- FC1: input: 100, output 10

FPGA implementation conv layer

- Image size WxH
- N input layers
- KxL kernel
- M output layers
- (L-1)xNxW on-chip storage
- Weights stored on-chip



Conv layer
Vivado HLS C/C++ implementation

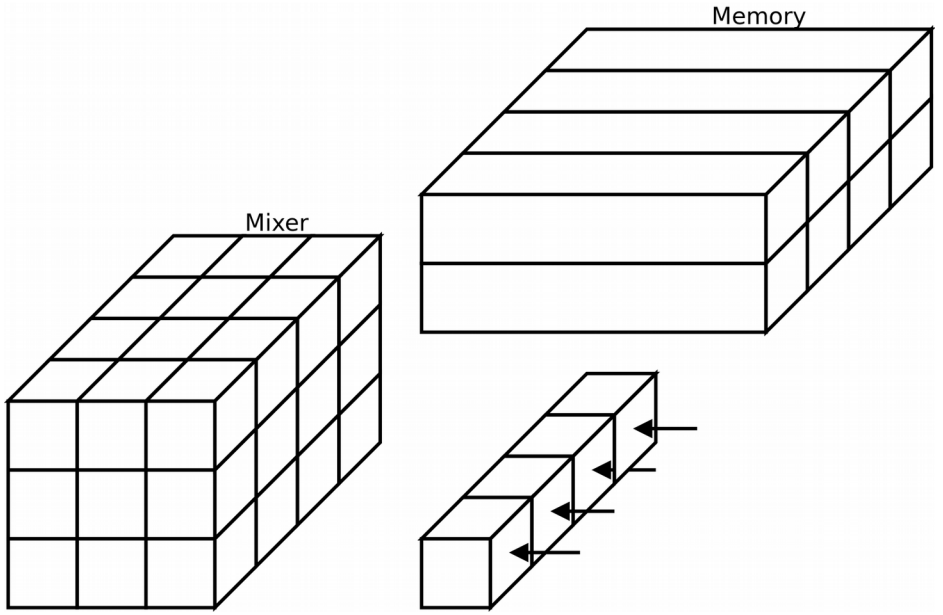
```
typedef short my_data_type;

my_data_type mem[N][L-1][W];

my_data_type mixer[N][L][K];

my_templ_type tmem[M][N][L][K];
```

- Arrays implemented as single 1D memory
- array_partition directive can be used for parallel access

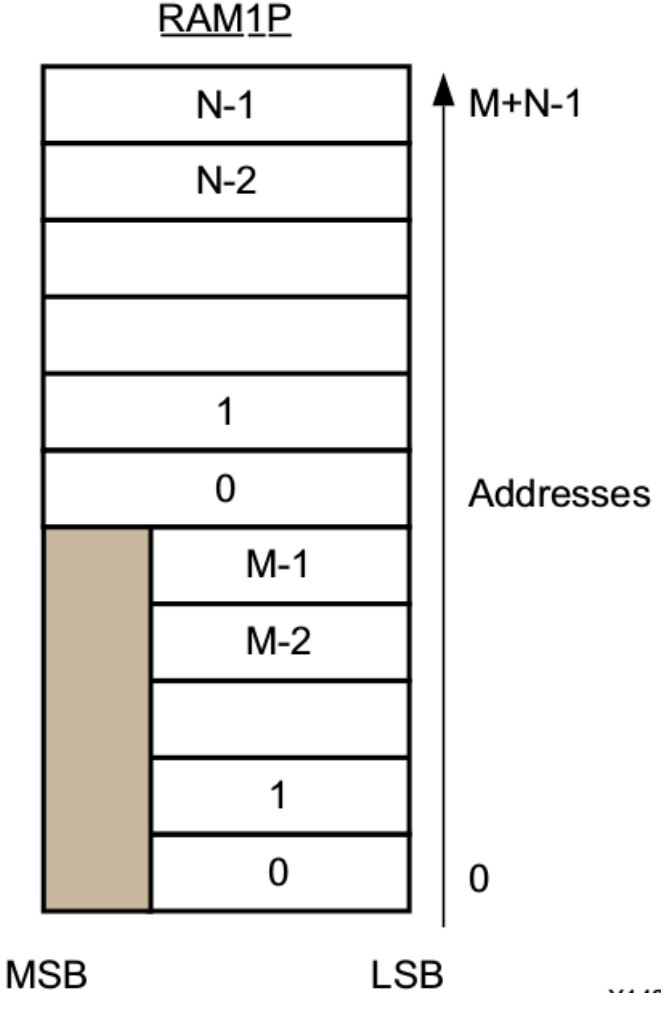


Memory and Mixer unit

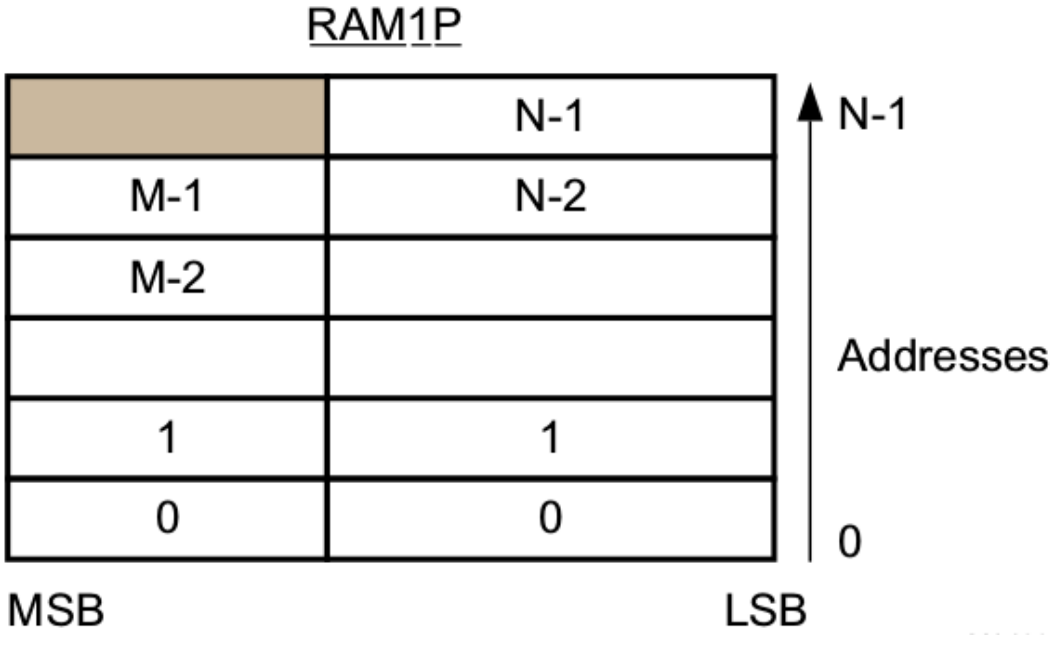
```
FOR_HEIGHT : for(j=0,wr_addr=0,rd_addr=1;j<HEIGHT;j++) {
  FOR_WIDTH : for(i=0;i<WIDTH;i++) {
    tmp_in_v=x_in.read();
    FOR_IN_LAYERS : for(m=0;m<INLAYERS;m++) {
      tmp_in[m]=tmp_in_v.data((m+1)*sizeof(my_data_type)*8-
1,m*sizeof(my_data_type)*8);
      for(k=0;k<TSIZE;k++) {
        for(l=0;l<TSIZE-1;l++) {
          mix[m][k][l]=mix[m][k][l+1];}}
      FOR_MIX : for(k=0;k<TSIZE-1;k++) {
#pragma HLS DEPENDENCE variable=mem inter false
        mix[m][k][TSIZE-1]=mem[m][k][rd_addr];
        mem[m][k][wr_addr]=mix[m][k+1][TSIZE-1];}
      mem[m][TSIZE-2][wr_addr]=mix[m][TSIZE-1][TSIZE-1];
      mix[m][TSIZE-1][TSIZE-1]=tmp_in[m];}}
```

Mapping Many Arrays into One Large Array

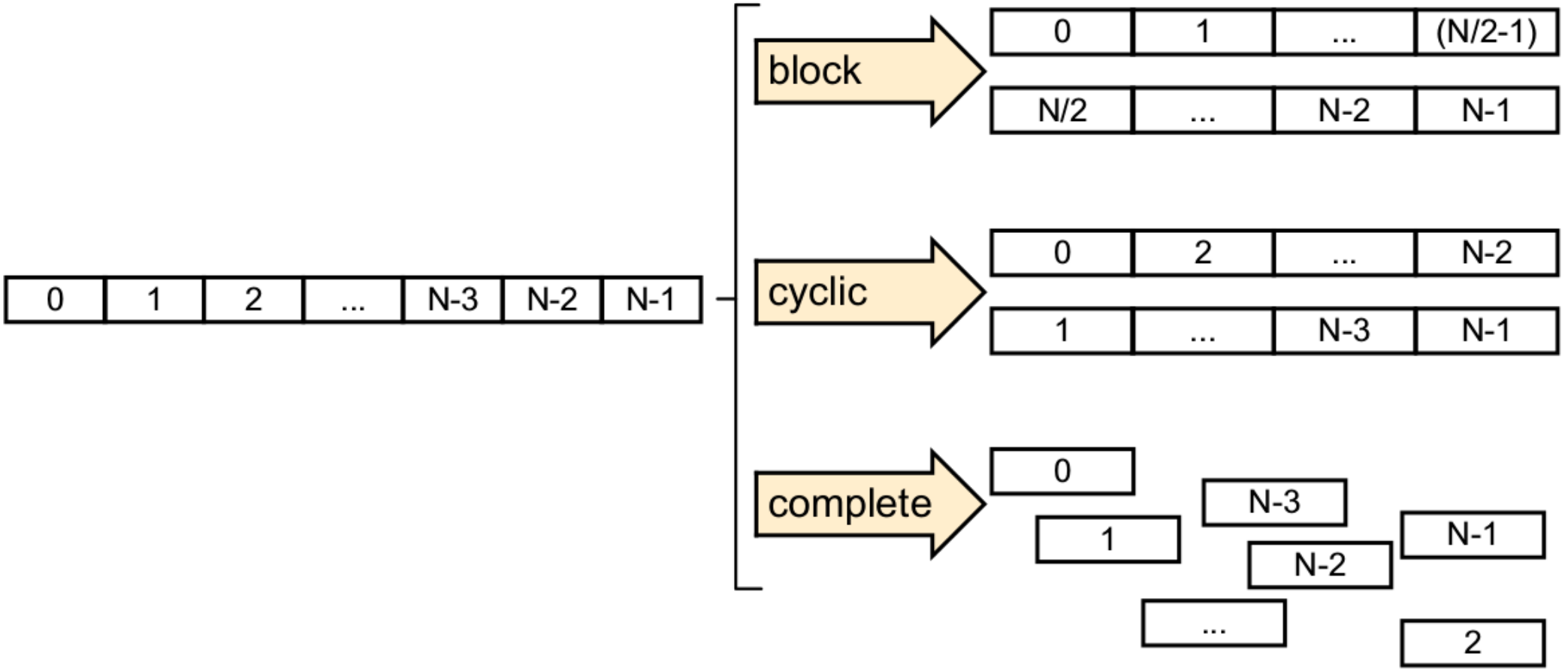
- Horizontal Array Mapping



- Vertical Array Mapping



Partitioning Arrays to Improve Pipelining



Arithmetic unit

```
for(k=0;k<TSIZE;k++) {
  for(l=0;l<TSIZE;l++) {
    FOR_OUT_LAYERS : for(n=0;n<OUTLAYERS;n++) {
      if ((k==0)&&(l==0)&&(m==0)) {
        memtmp[n]=tconst[n]+mix[m][k][l]*tmem[n][m][k][l];
      } else {
        memtmp[n]+=mix[m][k][l]*tmem[n][m][k][l];
}}}}
```

- Loops can be fully or partially unrolled
- Number of multipliers can be controlled by complier directives

Example: conv + pool

- C++ template functions

```
template<int INLAYERS, int OUTLAYERS, int TSIZE, int WIDTH, int HEIGHT> inline void conv_t( … )
```

```
template<int LAYERS, int TSIZE, int WIDTH, int HEIGHT> inline void pool_t( … )
```

- Conv layer
  - Input: 512x512x4
  - Output: 510x510x8
  - Template: 3x3
- Pooling layer
  - Input: 510x510x8
  - Output: 255x255x8
  - Size: 2x2