ECE382M.20: System-on-Chip (SoC) Design

Lecture 14 – Emulation & Prototyping

Sources:
Steven Smith

Andreas Gerstlauer
Electrical and Computer Engineering
University of Texas at Austin
gerstl@ece.utexas.edu

The University of Texas at Austin
Electrical and Computer Engineering
Cockrell School of Engineering

Lecture 14: Outline

- **Emulation and prototyping**
  - Design validation
  - Field Programmable Gate Arrays (FPGAs)
- **Programmable Logic Devices**
  - History and types
  - FPGA technology
  - FPGAs for production, emulation & prototyping
- **Prototyping Board**
  - Xilinx UltraScale FPGA family
  - Zynq UltraScale+ MPSoC
  - Ultra96 board

Design Validation

- **Verification & validation (debug) account for more than 70% of SoC design effort**
  - (Formal) verification vs. (functional) validation
    - Correctness (wrt specification) vs. performance (wrt purpose/requirements)
    - "Did we build the thing right?" vs. "Did we build the right thing?"
  - Validation of implementation properties requires execution
- **Complex SoCs are impractical to simulate at the whole-system level**
  - Simulation more tractable at the block level
  - SoCs depend upon complex interactions between SW and among disparate HW elements.
  - Execution of application SW usually required

➢ **Emulation and prototyping is on the order of 50 to 10,000 time faster than host-based simulation**

Validation Approaches

- **Simulation (aka Virtual Prototyping)**
  - Execute model of design on host machine
    - Co-simulation between different models (e.g. SystemC+HDL)
  - Very good observability & debugabality
- **Emulation**
  - Execute model of design in (reconfigurable) hardware
    - Can potentially simulate logical time in hardware-accelerated form
  - Integrate extensive debugging & tracing capabilities
- **(Physical) Prototyping**
  - Synthesize RTL directly into (reconfigurable) hardware
    - Cycle-accurate execution at speed of prototyping hardware
  - Limited observability & debugging

Field Programmable Gate Arrays (FPGAs)

- **Pre-manufactured yet reconfigurable logic**
  - Emulation and prototyping platform for ASIC designs
    - Validation and verification before costly ASIC spin
    - Limits in size and speed
  - In production as system component
    - Flexibility of static or dynamic reconfiguration via download of bitstream
    - Between hardware and software, cost vs. benefit analysis
- ➢ **Implement logic via memories**
  - Lookup tables (LUTs)
    - Arbitrary boolean functions as table in memory
  - Configurable Logic Blocks (CLBs)
    - Combine LUTs with flip-flops and latches to realize sequential logic
  - Switch matrices (programmable interconnect)
    - Connect array of CLBs via multiplexers configured by internal registers

Lecture 14: Outline

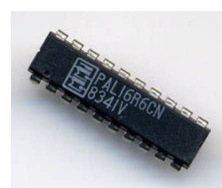
- ✓ **Emulation and prototyping**
  - ✓ Design validation
  - ✓ Field Programmable Gate Arrays (FPGAs)
- **Programmable Logic Devices**
  - History and types
  - FPGA technology
  - FPGAs for production, emulation & prototyping
- **Prototyping Board**
  - Xilinx UltraScale FPGA family
  - Zynq UltraScale+ MPSoC
  - Ultra96 board

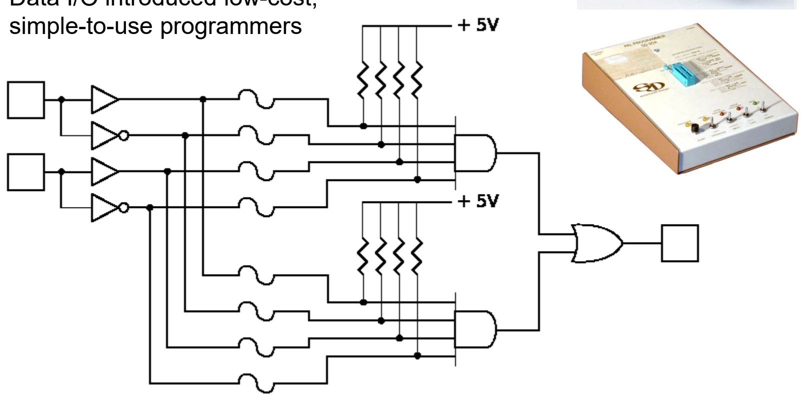
Early Programmable Logic Devices

- **Programmable Read-Only Memory (PROM) devices (1956)**
  - Programmed to realize arbitrary combinational functions
    - – Combinational inputs wire to PROM address bits
    - – Combinational outputs driven by PROM data bits
- **Mask-programmable gate arrays (MPGA) were introduced by Motorola in 1969**
  - Similar "Programmable Logic Array" (PLA) by TI in 1970
  - Customized during fabrication by the device vendor
    - – High non-recurring engineering (NRE) charge and long lead times
- **In 1971, General Electric combined PROM technology with gate array structures**
  - First field programmable logic device
    - – Customized by end user
    - – Low NRE costs and fast time-to-market
  - Experimental only – never released

Programmable Array Logic (PAL)

- **Monolithic Memories Inc. (MMI), based on GE ideas (1978)**
  - Programmable AND and OR planes
    - – Each junction in the PAL is a fuse
    - – Simpler and faster than earlier PLAs
  - Simple design flow and tools (PALASM)
    - – Data I/O introduced low-cost, simple-to-use programmers



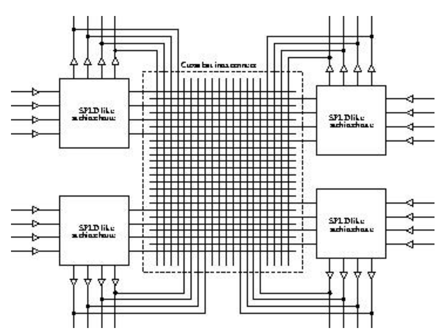


Field Programmable Logic Devices

- **Altera (formed in 1983), introduced the reprogrammable Electrically Programmable Logic Device (EPLD) in 1984**
- **Lattice Semiconductor introduced Generic Array Logic (GAL) devices in 1985**
  - Basically a reprogrammable PAL
- **Complex Programmable Logic Device (CPLD) technology emerged in the mid 1980s, first released by Altera**
  - A number of Simple PLDs (PAL-like structures + FF)
  - With programmable interconnect
- **Xilinx (founded in 1984), introduced the first Field Programmable Gate Array (FPGA) in 1985, the XC2064**
  - Contained 64 complex logic blocks (CLBs), each with two 3-input look-up tables (LUTs)

Complex Programmable Logic Device (CPLD)

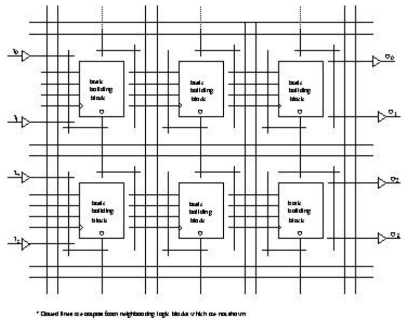
- **Typically combine coarse-grained SPLD structures with a programmable crossbar interconnect**
  - Don't scale well because of the crossbar interconnect
  - Only limited support for multi-level logic
  - Compared to FPGAs
    - Higher gate density
    - Less interconnect density
    - Better timing uniformity
    - Generally faster in equivalent device technology



- **Non-volatile technology for programming**
  - Memory (reprogrammable)
  - Fuse/anti-fuse (one-time programmable – OTP)

Field Programmable Gate Arrays (FPGAs)

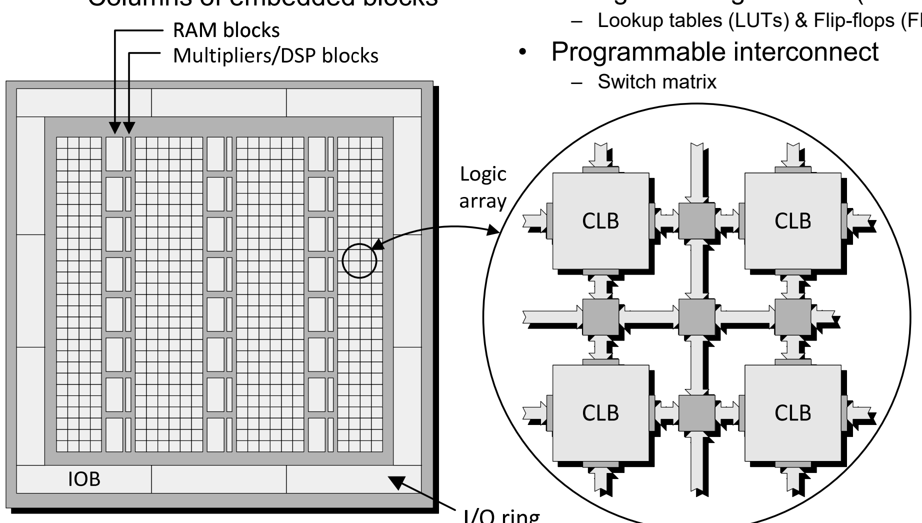
- **Two-dimensional array of customizable logic blocks combined with an interconnect array**
  - Logic blocks based on look-up tables (LUTs) or any other functionally complete behavior
    - – Each logic block must offer functional completeness
  - Interconnect based on flexible wire segments
    - – Interspersed switches for greater interconnect flexibility than CPLDs



- **Combines the advantages of MPGA and (S)PLD**
  - Comparatively lower gate density with much more complex programmable interconnect capabilities than CPLDs

Modern FPGA Architecture

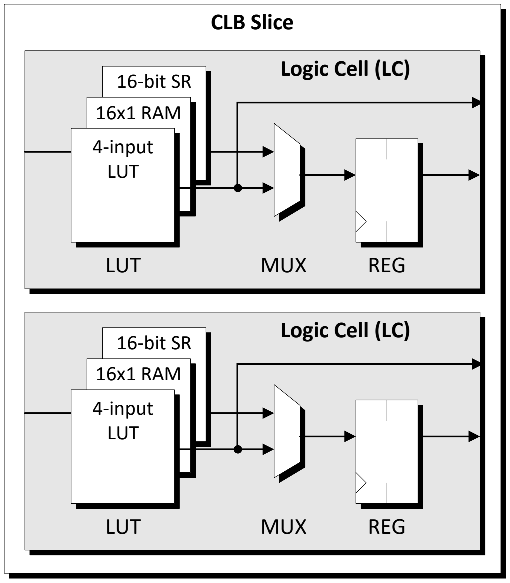
- **Hardcoded logic**
  - Columns of embedded blocks
- **Programmable logic fabric**
  - Configurable logic blocks (CLBs)
    - – Lookup tables (LUTs) & Flip-flops (FFs)
  - Programmable interconnect
    - – Switch matrix



- **Configurable I/O blocks (IOBs)**

Source: Clive Maxfield, "The Design Warrior's Guide to FPGAs Devices, Tools, and Flows," ISBN 0750676043, © 2004 Mentor Graphics Corp. (mentor.com)

Configurable Logic Blocks (CLBs)



- **Each CLB has one or more Slices**
- **Each Slice has one or more Logic Cells (LCs)**
  - 1 Flip-flop (FF) or latch
  - 1 Lookup Table (LUT)
    - Stores truth table for combinational logic
    - Some LUTs can be used as distributed RAM/ROM or shift registers
  - Carry look-ahead (CLA) logic
  - Dedicated muxes

Source: Clive Maxfield, "The Design Warrior's Guide to FPGAs Devices, Tools, and Flows," ISBN 0750676043, © 2004 Mentor Graphics Corp. (mentor.com)

FPGA Programmability

- **Field programmable capabilities derive from switches**
  - Devices based on fuses (bi-polar) or anti-fuses (CMOS) are one-time programmable (OTP)
  - Devices based on memory are reprogrammable
- **Non-volatile memory-based devices support instant-on functionality (as do OTP devices) and don't require external memory to store device configuration information.**
  - Flash, EPROM, or EEPROM
- **SRAM-based devices offer faster configuration, but require an external non-volatile memory to store configuration information**
  - Requires device "boot"

(Partial) Dynamic Reconfiguration (PDR)

- **Introduced with Xilinx XC6000 in the mid 1990s**
  - Continues to operate while portions are reconfigured
  - Comparatively fine-grained reconfiguration
    - Newer devices, beginning with the Virtex-2 Pro, have more coarse addressability at the bank or slice level
  - Xilinx support for PDR has been sporadic and tentative
    - Repeatedly, announced tool support only to later retract
    - Currently supported and for the first time the tools actually help
- **Altera has not yet developed devices capable of PDR**
  - There are rumors that they may

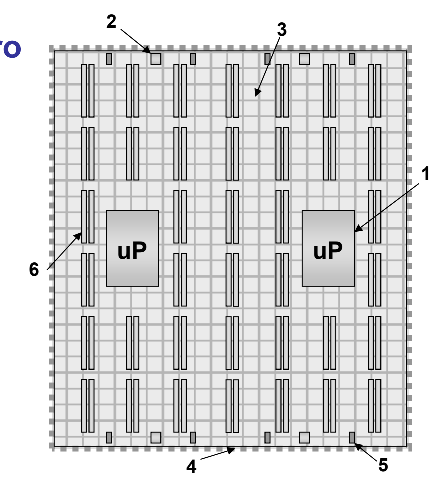
➢ **Many interesting applications**
  - Work around size limitations (module swapping)
  - Self-modifying, dynamic instruction set architectures
  - Dynamically instantiate HW accelerators in SoCs

ECE382M.20: SoC Design, Lecture 14 © S. Smith 15

Embedded Processor Cores

- **Pioneered in Xilinx Virtex-II Pro**
  - Up to 4 PowerPC cores
    - Hard macros
  - Throughout Virtex family
    - Virtex 2 through 6
    - Switch to ARM with 7 Series

➢ **HW/SW co-design**
  - Native SW performance
    - As opposed to emulated soft cores in FPGA fabric



1. PowerPC block
2. RocketIO Multi-Gigabit Transceivers
3. CLB and Configurable Logic
4. SelectIO-Ultra
5. Digital Clock Managers
6. Multipliers and Block SelectRAM

ECE382M.20: SoC Design, Lecture 14 © 2021 A. Gerstlauer 16

FPGA vs. ASIC for Production Use

- **Much shorter design time**
  - ~Less than a year versus 2-3 years for an ASIC
- **Cost**
  - No NRE vs. $M development cost for an ASIC
  - Much higher unit costs than those for ASICs
  - ➢ Depends on anticipated volume: NRE + (RE * Volume)
- **Performance gap**
  - Power consumption: ~7 times dynamic power*
  - Area consumption: ~18 times the area*
- **IP Protection**
  - Exposure during fabrication vs. in the field

➢ **FPGAs are the fastest growing semiconductor segment**
  - From 10% to approximately 25% in recent years
  - Dramatic decline in ASIC design starts: 11,000->1,500, ’97-’02

* Kuon, I.; Rose, J. (2006). Measuring the gap between FPGAs and ASICs. Intl. Symposium on FPGAs, 2006.

FPGAs for Emulation & Prototyping

- **Unmatched execution performance**
- **Cost effective, especially if FPGA evaluation boards are used as an ad hoc emulator**
  - Commercial system can be quite expensive, but are still cheaper than an extra ASIC spin
- **Robust verification possible**
  - Application software may be used in verification process, where it is typically impractical for simulation
- **Reduces design risk for ASICs**
  - Facilitates the fastest path to the market for complex SoC design

Emulation & Prototyping Challenges (1)

- **Size restrictions require partitioning**
  - Most "interesting" designs will require multiple FPGAs
  - Quality of partitioning determines emulation performance
    - Tool support is vendor-specific and not always particularly effective
    - Often the difference between 10 MHz and 400 MHz system clock rates
    - Manual intervention often necessary, costly and time consuming
    - Interface signals among FPGAs may be insufficient for optimal partitions
- **HDL targeting ASIC doesn't always map easily into FPGAs**
  - Clock and initialization logic
  - Memory technology and I/O interfaces may differ
    - E.g., implementation uses flash but emulation only has DRAM
  - Bus models and their implementation may differ
    - Generally no tri-state signals internal to FPGAs
  - Debug, controllability and visibility additions
  - ➢ Develop HDL with both FPGA and ASIC in mind

Emulation & Prototyping Challenges (2)

- **Co-verification / co-emulation**
  - Third-party IP may not be available in suitable (HDL) form
  - Interface FPGAs to simulator or C/C++ model running on a general-purpose host
    - Always ends up being gating factor on performance, severely constraining achievable emulation speeds
    - Discrete HW instantiation of third-party IP may require custom interface and models
  - Differences in software processor architectures
    - e.g., FPGA's internal PowerPC hard core instead of target ARM
- **Emulation speed may be limited by I/O bottlenecks**
  - Data collection, Stimuli
- **Partitioning and bit stream generation is time consuming**
  - Recompilation may take hours (or worse)

Emulation & Prototyping Challenges (3)

- **In-circuit / in-environment emulation**
  - Interaction with the environment or with other systems
  - If emulated speed is less than the target operational speed, need to consider the impact on real-time operation
    - Network interfaces can often be scaled to retain effective equivalence with real-time operation
    - E.g., Use 10 Mbps Ethernet on emulator running at 1/10 the rate of the target operational speed which is intended to work with 100 Mbps networks

➢ **In the end, executing an approximate model of target SoC**
  - Important to bear this fact in mind when interpreting results
    - Still need to do extensive verification through simulation of those blocks known to be different between the emulated system and the target design.
    - Same is true for interfaces between blocks and clock and reset logic.

Lecture 14: Outline

✓ **Emulation and prototyping**
  - ✓ Design validation
  - ✓ Field Programmable Gate Arrays (FPGAs)

✓ **Programmable Logic Devices**
  - ✓ History and types
  - ✓ FPGA technology
  - ✓ FPGAs for production, emulation & prototyping

- **Prototyping Board**
  - Xilinx UltraScale FPGA family
  - Zynq UltraScale+ MPSoC
  - Ultra96 board

Major FPGA Device Vendors

- **Xilinx and Altera are market leaders in SRAM-based FPGAs**
  - Combined controlling >80% of FPGA and CPLD market
    - Xilinx ~50%, Altera ~35%
  - Also offer non-volatile and OTP devices
- **Actel (Microsemi) offers anti-fuse and flash-based devices**
  - Igloo and Igloo Nano devices have very low power and sophisticated sleep mode options
    - Finally a programmable logic solution suitable for battery-powered applications
- **Lattice Semiconductors offers SRAM-based devices with integrated configuration flash**

Major FPGA Device Families

- **Xilinx**

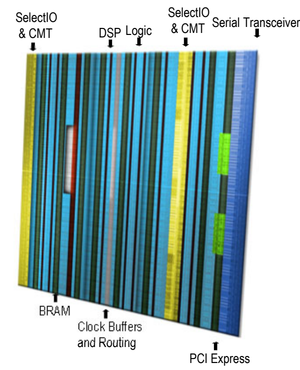
Technology	Low-end	Mid-range	High-Performance
120/150 nm			Virtex-II
90 nm	Spartan 3		Virtex-4
65 nm			Virtex-5
40/45 nm	Spartan 6		Virtex-6
28 nm	Artix-7	Kintex-7	Virtex-7
20 nm		Kintex UltraScale	Virtex UltraScale
16 nm	Artix UltraScale+	Kintex UltraScale+	Virtex UltraScale+

- **Altera**

Technology	Low-end	Mid-range	High-Performance
130 nm	Cyclone		Stratix
90 nm	Cyclone II		Stratix II
65 nm	Cyclone III	Arria I	Stratix III
40 nm	Cyclone IV	Arria II	Stratix IV
28 nm	Cyclone V	Arria V	Stratix V
20/14 nm		Arria 10	Stratix 10

Xilinx UltraScale Series

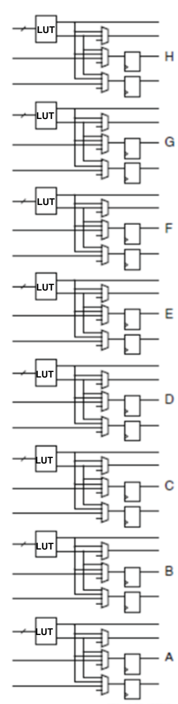
- **Unified architecture (28/16nm FinFET+)**
  - Scalable column-based architecture
    - Using common building blocks
      - » CLBs, DSPs, IOBs, Tranceivers, PCIe, ADCs, Clock management tiles (CMTs)
    - Low-power features
    - 3D multi-die stacked silicon interconnect (SSI)
  - High-/mid-/low-range families
    - Virtex, Kintex, Artix
    - Number of CLBs, DSPs, BRAMs, etc.
  - UltraScale+ (16nm) vs. UltraScale (28nm)
    - Power management options
    - Adds high-density UltraRAM with more flexibility
    - Virtex UltraScale+ available with 3D-stacked HBM options
      - » Connected via silicon interposer



Source: Xilinx, "UltraScale Architecture and Product Data Sheet: Overview"

Xilinx UltraScale Series CLBs

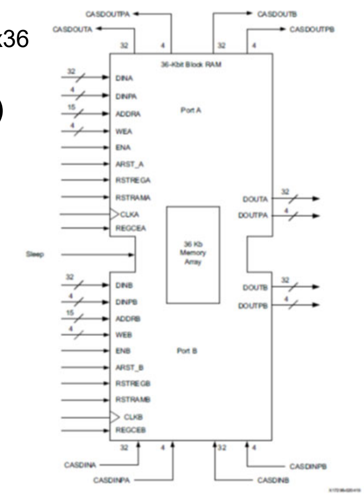
- **1 slice per CLB**
  - Eight 6-input LUTs
    - Configurable as two 5-input LUTs
  - Sixteen FFs
    - Configurable as latches
  - Carry logic
  - Wide muxes
- **Two types of slices**
  - SLICEL (logic)
    - Regular LUT
  - SLICEM (memory)
    - LUTs configurable as 64-bit distributed memory
    - Or as shift registers



Source: Xilinx, "UltraScale Architecture Configurable Logic Block: User guide"

Xilinx UltraScale Series BRAMs

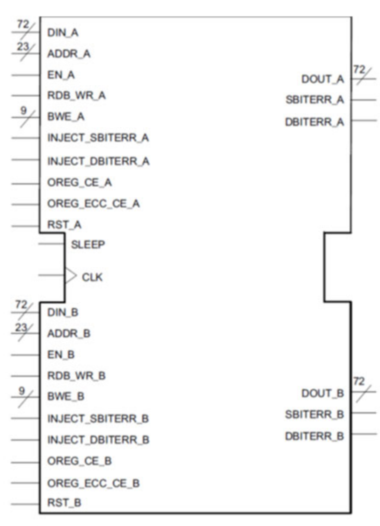
- **36kB (or 2x 18kB) block RAMs (BRAMs)**
  - Synchronous RAM
  - Dual-ported (no contention avoidance!)
    - Two read and two write ports
    - 36kB: 32Kx1, 16Kx2, 8Kx4, 4Kx9, 2Kx18, 1Kx36
    - 18kB: 16Kx1, 8Kx2, 4Kx4, 2Kx9, 1Kx18
  - Single-port mode ("Simple Dual-Port")
    - 512x36 (18kB) or 512x72 (36kB)
  - Can be cascaded
    - Adjacent BRAMs cascadable without any logic overhead
  - Built-in error correction
    - 64-bit ECC (correct single, detect two bit)
  - FIFO mode
  - Individual write enables (for CPUs)
  - Sleep mode



Source: Xilinx, "UltraScale Architecture Memory Resources: User Guide"

Xilinx UltraScale Series UltraRAMs

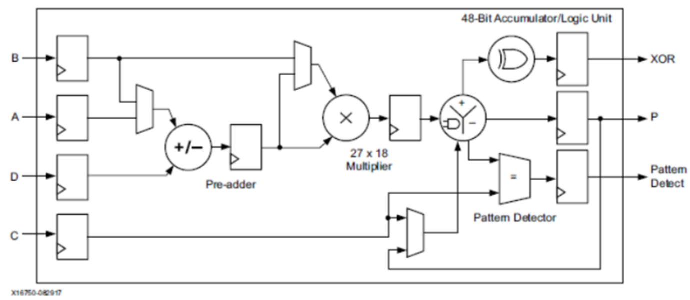
- **288kB ultra RAMs (UltraRAMs)**
  - High-density synchronous RAM
  - Dual-ported
    - 4K x 72 configuration
    - No support for configurable widths
  - Can be cascaded
    - Adjacent UltraRAMs in a column cascadable with little logic overhead
  - Built-in ECC
    - Single-bit correction
    - Double-bit detection
  - Sleep mode
    - Automatic sleep mode



Source: Xilinx, "UltraScale Architecture Memory Resources: User Guide"

Xilinx UltraScale Series DSP48E2 Slices

- **Support 96-bit Multiply-Accumulate (MACC) operation**
  - 27-bit pre-adder, 27x18 signed multiplier, 48-bit ALU
    - Plus 17-bit shifter, pattern detector
    - Cascade paths for wide functions
  - Pipelined, SIMD operation (12/24 bit)
  - Two DSP slices form a DSP tile (same height as one BRAM)



Source: Xilinx, "UltraScale Architecture DSP Slice: User Guide"

Xilinx UltraScale Series I/O

- **SelectIO Input/Output Blocks (IOBs)**
  - Flexible I/O standards
    - High range: supports standards up to 3.3V
    - High performance: supports I/O standards up to 1.8V
- **High-Speed Serial I/O Transceivers**
  - Different types, multiple standards
    - GTH/GTY, GTM (0.5 - 32 Gbps, 58 Gbps)
  - Dedicated PCIe blocks
    - Gen3/Gen4 (8/16 GT/s)
- **Networking Blocks**
  - 150Gbps Interlaken, 100Gbps Ethernet
- **RF Data Converters (RF-ADCs and RF-DACs)**
  - Only available in Zynq UltraScale+ RFSoC devices

Lecture 14: Outline

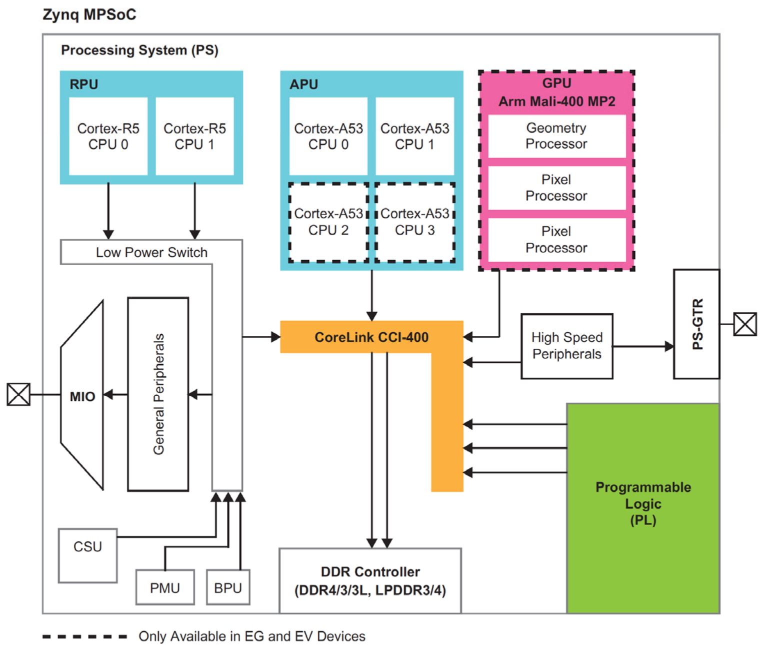
- ✓ **Emulation and prototyping**
  - ✓ Design validation
  - ✓ Field Programmable Gate Arrays (FPGAs)
- ✓ **Programmable Logic Devices**
  - ✓ History and types
  - ✓ FPGA technology
  - ✓ FPGAs for production, emulation & prototyping
- **Prototyping Board**
  - ✓ Xilinx UltraScale FPGA family
  - Zynq UltraScale+ MPSoC
  - Ultra96 board

Xilinx Zynq UltraScale+ MPSoC

- **Heterogeneous Multi-Processor Platform**
  - FPGA fabric with embedded ARM subsystem
    - Programmable logic (PL)
    - Processing system (PS)
  - PL based on Xilinx UltraScale+ FPGA technology
    - UltraScale+ fabric (16nm FinFET+)
    - Multi-standard I/O, serial I/O tranceivers, PCIe, networking, RF
  - PS based on dual- or quad-core core 64-bit ARM platform
    - Application Processing Unit (APU): 2x/4x Cortex-A53 (ARMv8 ISA), L1/L2
    - Real-Time Processing Unit (RPU): 2x Cortex-R5F (ARMv7), L1
    - Graphics Processing Unit (GPU): Mali-400 MP2
    - Video Coding Unit (VCU): HEVC, AVC (H.264/H.265)
    - On-chip SRAM, ext. DRAM controller, peripherals (USB, UART, etc.)

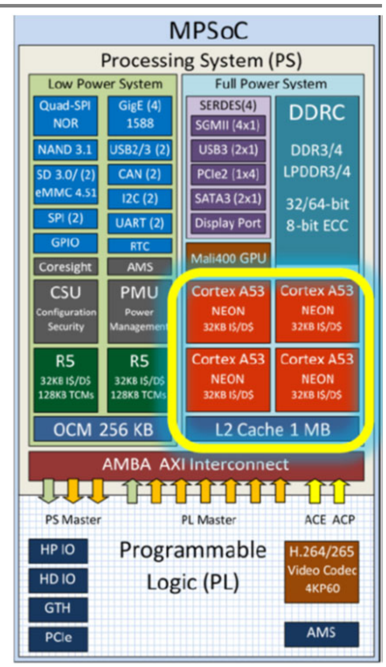
➢ **SoC prototyping**
  - Industry-standard ecosystems
    - Tool support (Xilinx/HLS +ARM/Linux)

Zynq UltraScale+ MPSoC Block Diagram



Application Processing Unit (APU)

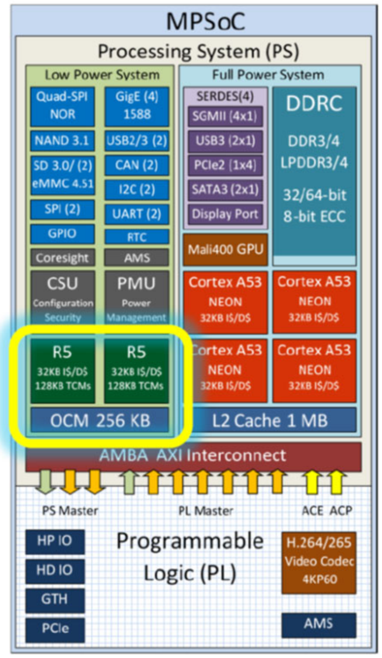
- **Dual- or quad-core ARM Cortex-A53**
  - 64-bit ARMv8 architecture
  - 32kB private L1 caches
  - NEON SIMD/FP unit
  - Up to 1.5GHz
- **1MB shared L2 cache**
- **Full-power domain**
  - Per-core power gating
  - L2 power gating



Source: V. Boppana *et al.*, "UltraScale+ MPSoC and FPGA Families," *HotChips 27*, 2015

Real-Time Processing Unit (RPU)

- **Dual-core ARM Cortex-R5F**
  - 32-bit ARMv7 architecture
  - 32kB private L1 caches
  - 128kB tightly-coupled memories (TCMs)
  - Single-/double-precision FPU
  - Supports lock-step operation
  - Up to 600MHz
- **256kB on-chip memory (OCM)**
- **Low-power domain**
  - Full-power domain powered off

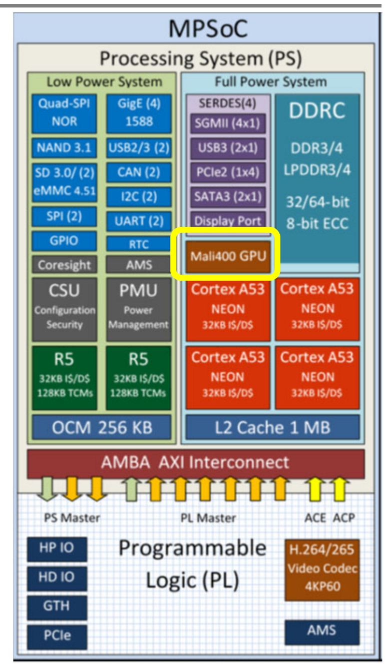


Source: V. Boppana *et al.*, "UltraScale+ MPSoC and FPGA Families," *HotChips 27*, 2015

ECE382M.20: SoC Design, Lecture 14 © 2021 A. Gerstlauer 35

Graphic & Video Units (GPU/VCU)

- **ARM Mail-400 MP2 Graphics Processing Unit (GPU)**
  - 1 geometry and 2 pixel processors
  - 64kB L2 cache
  - 1080p graphics
  - Up to 667MHz
- **Video Codec Unit (VCU)**
  - Hardware accelerated video encoding/decoding
  - H.265 (HEVC) 8Kx4K @ 15 fps, 4Kx2k @ 60fps
- **DisplayPort**
  - Up to 4Kx2K resolution (30 fps)
  - Up to 8-ch. 24-bit audio @ 192kHz

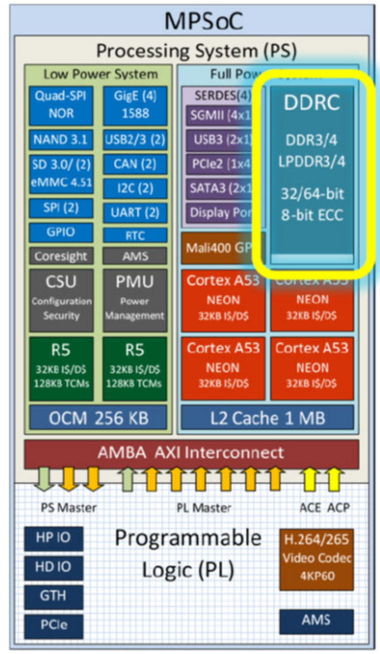


Source: V. Boppana *et al.*, "UltraScale+ MPSoC and FPGA Families," *HotChips 27*, 2015

ECE382M.20: SoC Design, Lecture 14 © 2021 A. Gerstlauer 36

Memory and Peripherals

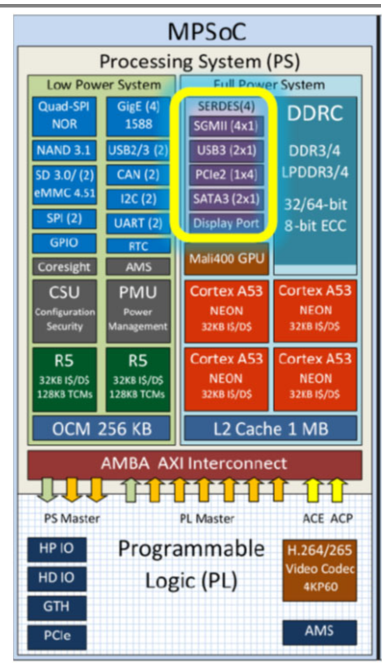
- **DDR Controller (DDRC)**
  - Six ports
  - DDR3/4 & LPDDR3/4
  - 32/64-bit DDR with ECC
  - QoS support
  - Memory protection, partitioning
  - Up to 2400Mbps



Source: V. Boppana *et al.*, "UltraScale+ MPSoC and FPGA Families," *HotChips 27*, 2015

Memory and Peripherals

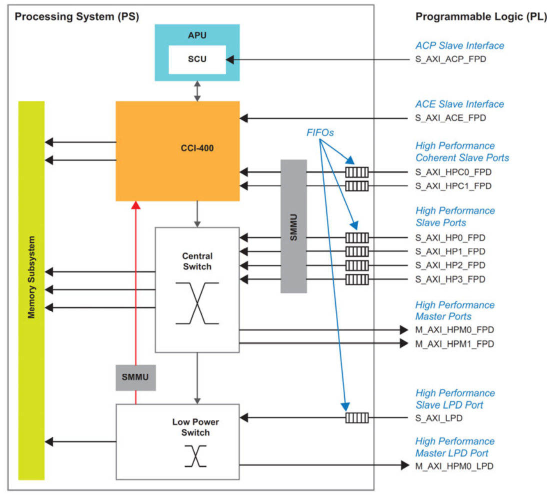
- **DDR Controller (DDRC)**
  - Six ports
  - DDR3/4 & LPDDR3/4
  - 32/64-bit DDR with ECC
  - QoS support
  - Memory protection, partitioning
  - Up to 2400Mbps
- **High-speed peripherals**
  - 2x USB3, 2x SATA, PCIe, DP
- **Low-speed peripherals**
  - GPIO, 2x CAN, 2x UART, 2x I^{2}C, 2x USB2/3, SDC, SPI, Ethernet



Source: V. Boppana *et al.*, "UltraScale+ MPSoC and FPGA Families," *HotChips 27*, 2015

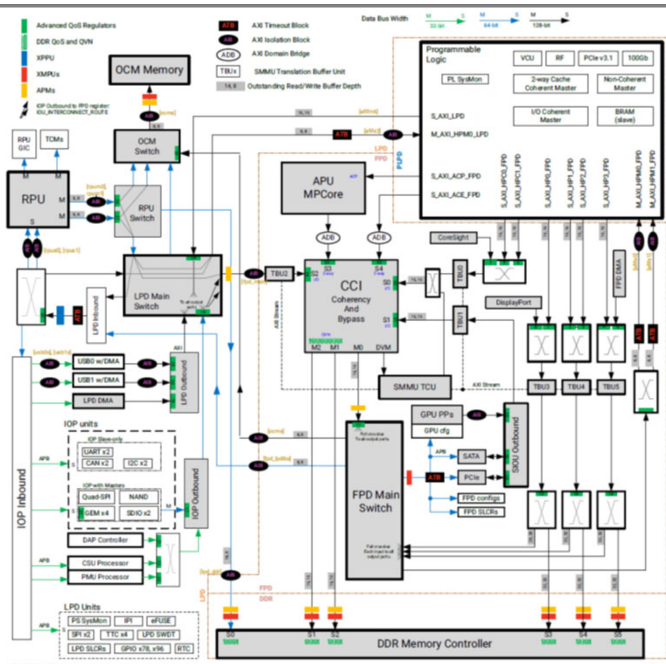
Programmable Logic (PL)

- **PL-PS Interfaces**
  - Accelerator Coherence Port (ACP)
    - One-way I/O coherency with APU caches
  - AXI Coherency Extensions (ACE)
    - Full coherency
  - PL master ports
    - Central crossbar and direct memory access (DMA)
    - 2x I/O coherent via CCI
    - 4x virtualization via SMMU
    - FIFO buffered
  - PL slave ports
    - 2x memory-mapped slave interfaces
  - Low-power domain interfaces
    - 1x master, 1x slave



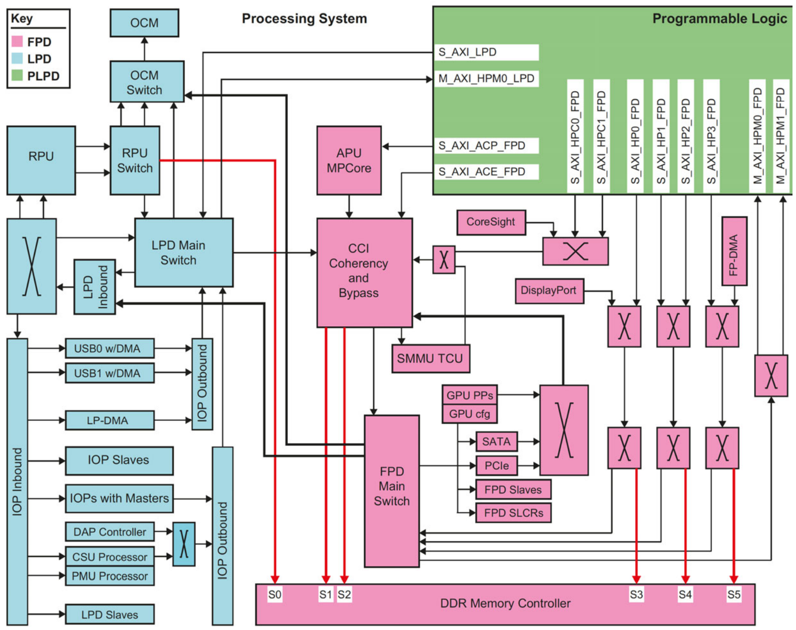
Source: V. Boppana *et al.*, "UltraScale+ MPSoC and FPGA Families," *HotChips 27*, 2015

System Interconnect

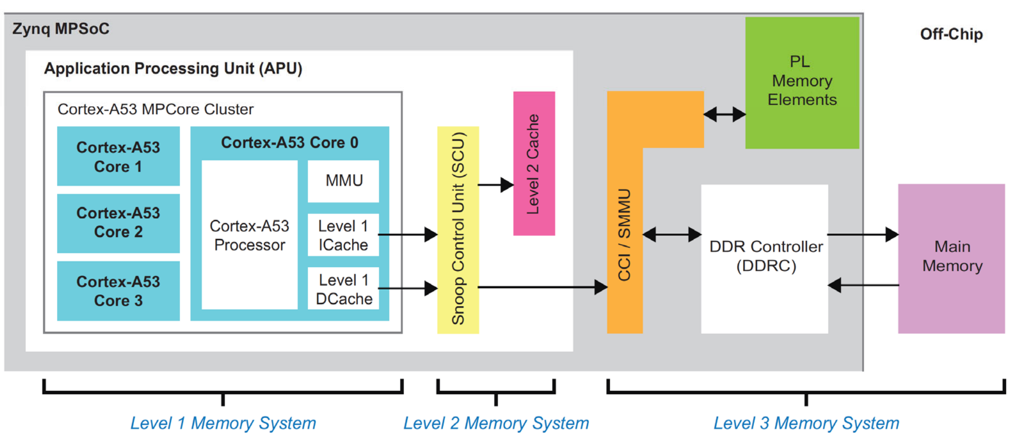


Zynq UltraScale+ MPSoC Technical Reference Manual, *Figure 15-1:* **PS Interconnect**

Power Domains

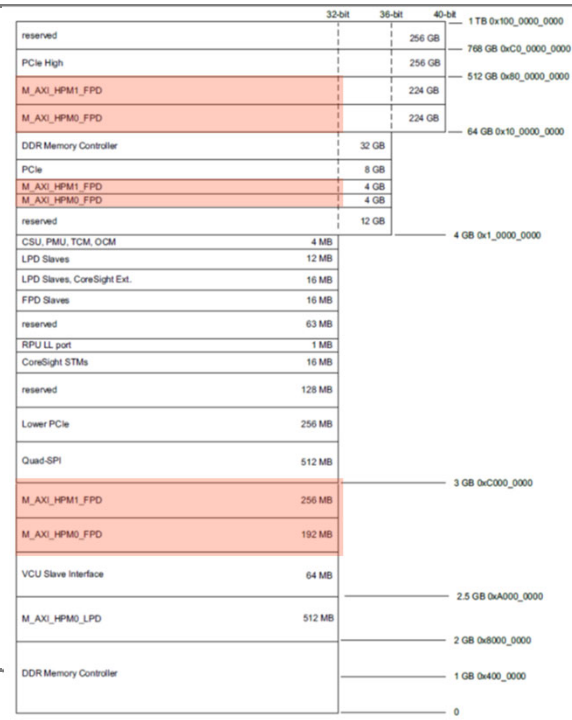


Memory Hierarchy



Memory Map

- **1TB of addressable space**



Zynq UltraScale+ Technical Reference Manual, *Figure 10-1:* **Global System Address Map**

Zynq UltraScale+ MPSoC Device Family

- **Three classes of MPSoC devices**
  - Low-end CG (dual-core APU)
  - High-end EG (quad-core APU + GPU)
  - Multimedia & vision EV (quad-core APU, GPU, VCU)
- **RFSoC devices**
  - DR (quad-core APU + RF-ADCs/RF-DACs)

	MPSoC			RFSoC
	CG Devices	**EG Devices**	**EV Devices**	**DR Devices**
APU	Dual-core Arm Cortex-A53	Quad-core Arm Cortex-A53	Quad-core Arm Cortex-A53	Quad-core Arm Cortex-A53
RPU	Dual-core Arm Cortex-R5F	Dual-core Arm Cortex-R5F	Dual-core Arm Cortex-R5F	Dual-core Arm Cortex-R5F
GPU	–	Mali-400MP2	Mali-400MP2	–
VCU	–	–	H.264/H.265	–

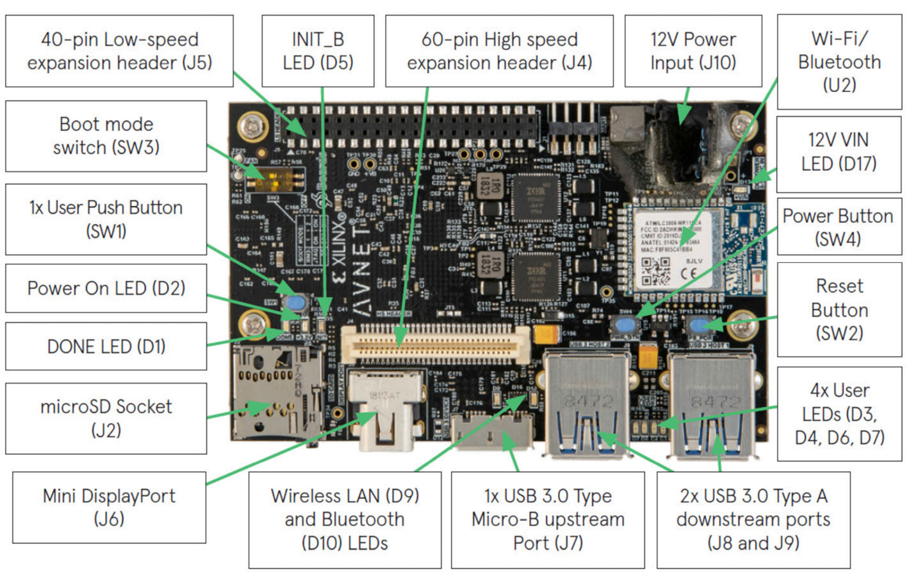
Source: Xilinx, "UltraScale Architecture and Product Data Sheet: Overview"

Zynq UltraScale+ MPSoC EG Devices

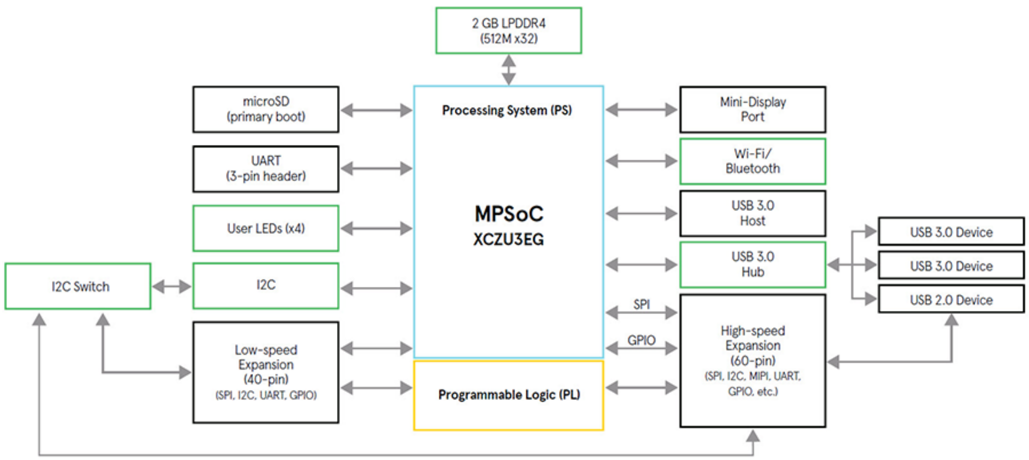
	ZU1EG	ZU2EG	ZU3EG	ZU4EG	ZU5EG	ZU6EG	ZU7EG	ZU9EG	ZU11EG	ZU15EG	ZU17EG	ZU19EG
Application Processing Unit	Quad-core Arm Cortex-A53 MPCore with CoreSight; NEON & Single/Double Precision Floating Point; 32KB/32KB L1 Cache, 1MB L2 Cache											
Real-Time Processing Unit	Dual-core Arm Cortex-R5F with CoreSight; Single/Double Precision Floating Point; 32KB/32KB L1 Cache, and TCM											
Embedded and External Memory	256KB On-Chip Memory w/ECC; External DDR4; DDR3; DDR3L; LPDDR4; LPDDR3; External Quad-SPI; NAND; eMMC											
General Connectivity	214 PS I/O; UART; CAN; USB 2.0; I2C; SPI; 32b GPIO; Real Time Clock; WatchDog Timers; Triple Timer Counters											
High-Speed Connectivity	4 PS-GTR; PCIe Gen1/2; Serial ATA 3.1; DisplayPort 1.2a; USB 3.0; SGMII											
Graphic Processing Unit	Arm Mali-400 MP2; 64KB L2 Cache											
System Logic Cells	81,900	103,320	154,350	192,150	256,200	469,446	504,000	599,550	653,100	746,550	926,194	1,143,450
CLB Flip-Flops	74,880	94,464	141,120	175,680	234,240	429,208	460,800	548,160	597,120	682,560	846,806	1,045,440
CLB LUTs	37,440	47,232	70,560	87,840	117,120	214,604	230,400	274,080	298,560	341,280	423,403	522,720
Distributed RAM (Mb)	1.0	1.2	1.8	2.6	3.5	6.9	6.2	8.8	9.1	11.3	8.0	9.8
Block RAM Blocks	108	150	216	128	144	714	312	912	600	744	796	984
Block RAM (Mb)	3.8	5.3	7.6	4.5	5.1	25.1	11.0	32.1	21.1	26.2	28.0	34.6
UltraRAM Blocks	0	0	0	48	64	0	96	0	80	112	102	128
UltraRAM (Mb)	0	0	0	13.5	18.0	0	27.0	0	22.5	31.5	28.7	36.0
DSP Slices	216	240	360	728	1,248	1,973	1,728	2,520	2,928	3,528	1,590	1,968
CMTs	3	3	3	4	4	4	8	4	8	4	11	11
Max. HP I/O[1]	156	156	156	156	156	208	416	208	416	208	572	572
Max. HD I/O[2]	24	96	96	96	96	120	48	120	96	120	96	96
System Monitor	1	2	2	2	2	2	2	2	2	2	2	2
GTH Transceiver 16.3Gb/s[3]	0	0	0	16	16	24	24	24	32	24	44	44
GTY Transceivers 32.75Gb/s	0	0	0	0	0	0	0	0	16	0	28	28
Transceiver Fractional PLLs	0	0	0	8	8	12	12	12	24	12	36	36
PCIE4 (PCIe Gen3 x16)	0	0	0	2	2	0	2	0	4	0	4	5
150G Interlaken	0	0	0	0	0	0	0	0	1	0	2	4
100G Ethernet w/ RS-FEC	0	0	0	0	0	0	0	0	2	0	2	4

Source: Xilinx, “UltraScale Architecture and Product Data Sheet: Overview”

Ultra96v2 Board



Ultra96v2 Block Diagram



ECE382M.20:SoC Design, Lecture 0 © 2021 A. Gerstlauer 47

Lecture 14: Summary

- **Programmable logic devices emerged in the 1970s and have advanced steadily since**
  - CPLDs and FPGAs have fundamentally different structures and, typically, different applications.
  - The emergence of very low-power devices has opened up potential applications in battery-powered applications, previously a complete non-starter
- **FPGAs are the fastest growing segment of the semiconductor market.**
  - All but very high volume consumer applications are likely better served by FPGAs than ASICs.
- **Emulation and prototyping offers an effective and powerful means of reducing design risks, development time and costs for ASIC designs**

ECE382M.20: SoC Design, Lecture 14 © S. Smith 48