Lecture 18 – Emulation & Prototyping

Sources:
Steven Smith

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

Lecture 18: 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 Series 7 FPGAs
  • Zynq-7000
  • ZedBoard
Design Validation

- Verification accounts for more than 70% of SoC design effort
  - (Formal) verification vs. (functional) validation
- 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 & debugability
- 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 18: 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 Series 7 FPGAs
  - Zynq-7000
  - ZedBoard
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

![PAL Diagram]
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
    - RAM blocks
    - Multipliers/DSP blocks
- Configurable I/O blocks (IOBs)
- Programmable logic fabric
  - Configurable logic blocks (CLBs)
    - Lookup tables (LUTs) & Flip-flops (FFs)
  - Programmable interconnect
    - Switch matrix

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

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

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
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


---

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.
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

<table>
<thead>
<tr>
<th>Major FPGA Device Families</th>
</tr>
</thead>
</table>

**Xilinx**

<table>
<thead>
<tr>
<th>Technology</th>
<th>Low-end</th>
<th>Mid-range</th>
<th>High-Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>120/150 nm</td>
<td>Spartan 2</td>
<td>Virtex-4</td>
<td></td>
</tr>
<tr>
<td>90 nm</td>
<td>Spartan 3</td>
<td>Virtex-5</td>
<td></td>
</tr>
<tr>
<td>65 nm</td>
<td>Spartan 6</td>
<td>Virtex-6</td>
<td></td>
</tr>
<tr>
<td>40/45 nm</td>
<td>Artix-7</td>
<td>Kintex-7</td>
<td>Kintex UltraScale</td>
</tr>
<tr>
<td>28 nm</td>
<td>Artix-7</td>
<td>Kintex-7</td>
<td>Kintex UltraScale</td>
</tr>
<tr>
<td>20/16 nm</td>
<td>Kintex UltraScale</td>
<td>Virtex UltraScale</td>
<td></td>
</tr>
</tbody>
</table>

**Altera**

<table>
<thead>
<tr>
<th>Technology</th>
<th>Low-end</th>
<th>Mid-range</th>
<th>High-Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>130 nm</td>
<td>Cyclone</td>
<td></td>
<td>Stratix</td>
</tr>
<tr>
<td>90 nm</td>
<td>Cyclone II</td>
<td></td>
<td>Stratix II</td>
</tr>
<tr>
<td>65 nm</td>
<td>Cyclone III</td>
<td></td>
<td>Stratix III</td>
</tr>
<tr>
<td>40 nm</td>
<td>Cyclone IV</td>
<td></td>
<td>Stratix IV</td>
</tr>
<tr>
<td>28 nm</td>
<td>Cyclone V</td>
<td></td>
<td>Stratix V</td>
</tr>
<tr>
<td>20/14 nm</td>
<td>Arria 10</td>
<td></td>
<td>Stratix 10</td>
</tr>
</tbody>
</table>
Xilinx 7 Series

- **Unified architecture (28 nm, 1.0V)**
  - Scalable column-based architecture
    - Using common building blocks
      - CLBs, DSPs, IOBs, Transceivers, PCIe, ADCs, Clock management tiles (CMTs)
    - Low-power features
  - High-end Virtex family
    - 3D multi-die stacked silicon interconnect (SSI)
    - up to 8 Mbytes RAM
    - up to 1200 user I/O pins
    - up to 2M logic cells & 2.4M FFs
- Mid-range Kintex
- Low-end Artix

![Artix-7 FPGA](image1)
![Virtex-7 FPGA](image2)
![Virtex-7 FPGA](image3)

Xilinx 7 Series CLBs

- **2 slices per CLB**
  - 6-input LUT
    - RAM/SR in SLICEM
    - Logic only in SLICEL
  - 4 FF/Ls
  - 4 FFs
    - Only if all FFs
  - Carry chain
  - Wide muxes

![CLBs Diagram](image4)
Xilinx 7 Series Block RAMs

- **36kB (or 2x 18kB) block RAMs (BRAMs)**
  - Synchronous RAM
  - Dual-ported (no contention avoidance!)
    - 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
    - Two adjacent BRAMs cascadable to 64Kx1 without any logic overhead
  - Built-in error correction
    - 64-bit ECC (correct single, detect two bit)
  - FIFO mode

Source: Xilinx, "7 Series FPGA Memory Resources User Guide"

Xilinx 7 Series DSP48E1 Slices

- **Support 96-bit Multiply-Accumulate (MACC) operation**
  - 25-bit pre-adder, 25x18 signed multiplier, 48-bit ALU
    - Plus 17-bit shifter, pattern detector
    - Cascade paths for wide functions
  - Pipelined, SIMD operation (12/24 bit)

Source: Xilinx, "7 Series FPGA DSP48E1 Slice User Guide"
**Xilinx 7 Series Clocking**

- **Divide into clock regions**
  - ½ width x 50 CLBs each
    - 10 BRAMs, 20 DSPs high
    - 50 IOBs, 4 T/x, PCIe
    - Buffers/repeaters

- **Clock management tiles (CMTs)**
  - Adjacent to I/O columns
    - 1 per region, 2 columns/device
  - Center clock spine
    - 1 horizontal clock row/region

Source: Xilinx, "7 Series FPGA Clocking Resources User Guide"

**Xilinx 7 Series I/O**

- **SelectIO Input/Output Blocks (IOBs)**
  - 2 columns per device, 50 IOBs per bank
  - Two distinct IOB types
    - High range: supports standards up to 3.3V
    - High performance: supports I/O standards up to 1.8V

- **High-Speed Serial I/O Transceivers**
  - Transceivers in quads, 4 per block
  - Different types, multiple standards
    - GTP/GTX/GTH/GTZ (3.75/12.5/13.1/28 Gbps)

- **PCI Express (PCIe) Blocks**
  - Build on GTX serial I/O transceivers
    - Compliant to PCIe protocol Gen1/Gen2/Gen3 (2.5/5/8 Gbps)

- **Analog-to-Digital Converter Blocks (XADCs)**
  - Two 12-bit, 1 MspS ADCs plus on-chip sensors per XADC
Lecture 18: 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 Series 7 FPGAs
    - Zynq-7000
    - ZedBoard

Xilinx Zynq-7000

- Extensible Processing Platform (EPP) / All-Programmable (AP) SoC
  - FPGA fabric with embedded ARM processor on single die
    - Programmable logic (PL)
    - Processing system (PS)
  - PL based on Xilinx Series 7 FPGA technology
    - Artix-7 or Kintex-7 fabric (28nm TSMC HPL process)
    - Multi-standard I/O, gigabit tranceivers, analog-to-digital converters (ADCs)
  - PS based on dual-core Cortex-A9
    - ARM Cortex-A9 MPcore (ARMv7 ISA, up to 1GHz)
    - L1/L2 caches, on-chip SRAM, ext. DRAM interfaces, peripherals

  SoC prototyping platform
  - Industry-standard ecosystems
    - Tool support (Xilinx/HLS +ARM/Linux)
Zynq-7000 Block Diagram

EE382V: SoC Design, Lecture 18 © 2014 A. Gerstlauer


Zynq-7000 Processing System (PS)

- Application Processing Unit (APU)
  - Dual-core Cortex-A9
    - NEON SIMD/FP unit
    - 512kB L2 cache
  - Snoop Control Unit (SCU)
    - L1 cache coherency
  - On-Chip Memory (OCM)
    - Dual-ported 256kB SRAM
- External memory interfaces
  - DDR3/DDR2
    - ECC memory controller
  - Config & legacy
    - Quad-SPI, NAND/NOR flash
- Peripherals
  - Standard I/O for PS or PL
    - GPIO, 2x USB, 2x CAN, 2x I2C, 2x UART, 2x Ethernet, 2x SPI
  - System peripherals controlled by/local to PS
    - Clock (PLL), debug access port (DAP), DMA ctrl., interrupt ctrl. (GIC), timers


© 2014 A. Gerstlauer
Zynq-7000 Programmable Logic (PL)

- PL-PS Interfaces
  - Accelerator Coherence Port (ACP)
    - Coherent access to caches
  - General-purpose (GP) AXI ports
    - 2x master, 2x slave
    - Connect to central crossbar
  - High-performance (HP) AXI ports
    - 4x master, FIFO buffered
    - Direct memory access (DMA)
- System interfaces
  - 16 shared interrupts to GIC, 4 private interrupts to cores (2x nIRQ/nFIQ)
  - Debug


Zynq-7000 Logic Fabric

- Xilinx Series-7 FPGA fabric
  - Embedded block RAM (BRAM) & DSP slices
  - Clock management tiles (CMT), I/O
  - PCI-Express & A/D interfaces

Zynq-7000 Clocks

- **Clock generation module**
  - Generate clocks for CPU, DDR, I/O and PL
    - Phase-locked loops (PLLs)
    - PS_CLK is external 30-60 MHz reference clock
  - Four general-purpose clocks for PL
    - FCLK_CLK0, FCLK_CLK1, FCLK_CLK2, FCLK_CLK3

Zynq-7000 Memory Map

- 4GB of addressable memory
Zynq-7000 Device Family

- Currently 7 different Z-devices
  - Low-end (7010, 7015, 7020), mid-range (7030, 7035, 7045, 7100)
  - Different packaging options (I/O)
  - Different speed grades (max. frequency)

<table>
<thead>
<tr>
<th></th>
<th>Z-7010</th>
<th>Z-7015</th>
<th>Z-7020</th>
<th>Z-7030</th>
<th>Z-7045</th>
<th>Z-7100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor Core</td>
<td>Dual ARM® Cortex™-A9 MPCore™ with CoreSight™</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Processor Extensions</td>
<td>NEDON™ &amp; Single / Double Precision Floating Point for each processor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cache</td>
<td>L1: 32kB Instruction/32kB Data per processor, L2: 512kB shared</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>On-Chip Memory</td>
<td>256 KB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Memory Interfaces</td>
<td>DDR3, DDR3L, DDR2, LPDDR2, 2x Quad-SPI, NAND, NOR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peripherals</td>
<td>2x USB 2.0 (OTG), 2x Tri-mode Gigabit Ethernet, 2x SD/SDIO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Programmable Logic</td>
<td>Artix-7</td>
<td>Artix-7</td>
<td>Artix-7</td>
<td>Kintex-7</td>
<td>Kintex-7</td>
<td>Kintex-7</td>
</tr>
<tr>
<td>Logic Cells (Gates equiv.)</td>
<td>28K Cells (~430k)</td>
<td>74K Cells (~1.1M)</td>
<td>65K Cells (~1.3M)</td>
<td>125K Cells (~1.9M)</td>
<td>350K Cells (~5.2M)</td>
<td>444K Cells (~6.8M)</td>
</tr>
<tr>
<td>Block RAMs @ 36kB</td>
<td>60 (240kB)</td>
<td>95 (380kB)</td>
<td>140 (560kB)</td>
<td>265 (1,060kB)</td>
<td>545 (2,180kB)</td>
<td>755 (3,020kB)</td>
</tr>
<tr>
<td>DSP Slices</td>
<td>80</td>
<td>160</td>
<td>220</td>
<td>400</td>
<td>900</td>
<td>2,020</td>
</tr>
<tr>
<td>Analog/Mixed-Signal</td>
<td>- Gen2 x4</td>
<td>- Gen2 x4</td>
<td>- Gen2 x4</td>
<td>- Gen2 x8</td>
<td>- Gen2 x8</td>
<td></td>
</tr>
<tr>
<td>Processor System</td>
<td>up to 100</td>
<td>up to 150</td>
<td>up to 200</td>
<td>up to 100</td>
<td>up to 250</td>
<td>up to 250</td>
</tr>
<tr>
<td>I/O</td>
<td>up to 130</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>High-Range 3.3V</td>
<td>up to 100</td>
<td>up to 150</td>
<td>up to 200</td>
<td>up to 100</td>
<td>up to 250</td>
<td>up to 250</td>
</tr>
<tr>
<td>High-Performance 1.8V</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>up to 150</td>
<td>up to 150</td>
<td>up to 150</td>
</tr>
<tr>
<td>Gigabit Transceivers</td>
<td>- 4 (8.25 Gb/s)</td>
<td>- 4 (8.25 Gb/s)</td>
<td>- 4 (12.5 Gb/s)</td>
<td>up to 4 (12.5 Gb/s)</td>
<td>up to 16 (12.5 Gb/s)</td>
<td>up to 16 (10.3Gb/s)</td>
</tr>
</tbody>
</table>

Source: Xilinx, "Zynq-7000 All Programmable SoC Overview"

ZedBoard

- Zynq-7000 based prototyping board
  - Z-7020, -1 grade
    - 667MHz max. ARM clock
    - 150 MHz bus clock
  - On-board DRAM
    - 512MB DDR3
  - MIO interfaces
    - LED & switch GPIO
    - Ethernet, USB, UART
    - SD card
  - PL peripherals
    - Audio, video, display
    - Peripheral modules (Pmods)
    - FPGA mezzanine cards (FMC)
  - System control
    - Reset, clock, debug
ZedBoard (Front)

- Power supply
- 2x 2Gbit Micron DDR3
- HDMI Transmitter (Analog Devices ADV7511)
- XC7Z020-CLG484
- OLED

Source: Taewon Suh, Korea University

ZedBoard (Back)

- 4-bit SPI (Quad-SPI)
- serial NOR flash
- Memory Serial (Analog Devices ADAU1763)
- 33.33MHz clock for PS
- SD (Secure Digital) Card
- 100MHz clock for PL

Source: Taewon Suh, Korea University
ZedBoard Multiplexed I/O (MIO)

- LED/Buttons
- USB UART
- USB OTG (Host)
- Gigabit Ethernet
- Pmod

ZedBoard PS I/O

- Boot configuration jumpers JP11-JP7 (MIO[0:2])
ZedBoard PL I/O

More Zynq/ZedBoard Information


- **Xilinx presentations**

- **COM509, “Computer Systems”, Korea University**

- **Avnet® X-fest 2012 presentations**

- **Xilinx 7 Series FPGA tutorials**
  [https://arco.esi.uclm.es/public/doc/tutoriales/Xilinx/FPGA_Architecture/7-Series/](https://arco.esi.uclm.es/public/doc/tutoriales/Xilinx/FPGA_Architecture/7-Series/)
Lecture 18: 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