MIPSfpga: Using a Commercial MIPS Soft-Core in Computer Architecture Education

Introduction

MIPSfpga Project:

- MicroAptiv soft-core processor (Imagination Technologies).
- Lots of learning materials. Students learn about computer architecture and hw-sw codesign on a commercial MIPS.
- Students can use the same hardware tool, an FPGA, for both digital design and computer architecture.
- Bridges the gap between existing curricula (use of toy MIPS processors) and industrial-level work (real MIPS processor).

MIPSfpga is made up of 3 packages:

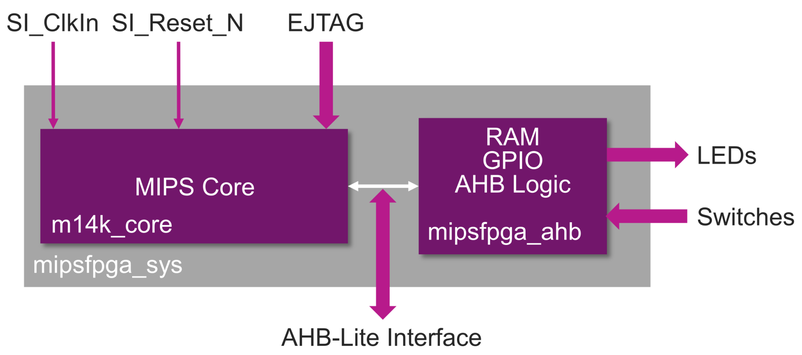
- **1st package:** MIPSfpga Getting Started Guide (GSG)
- **2nd package:** MIPSfpga Labs
- **3rd package:** MIPSfpga SoC

1st package: MIPSfpga GSG

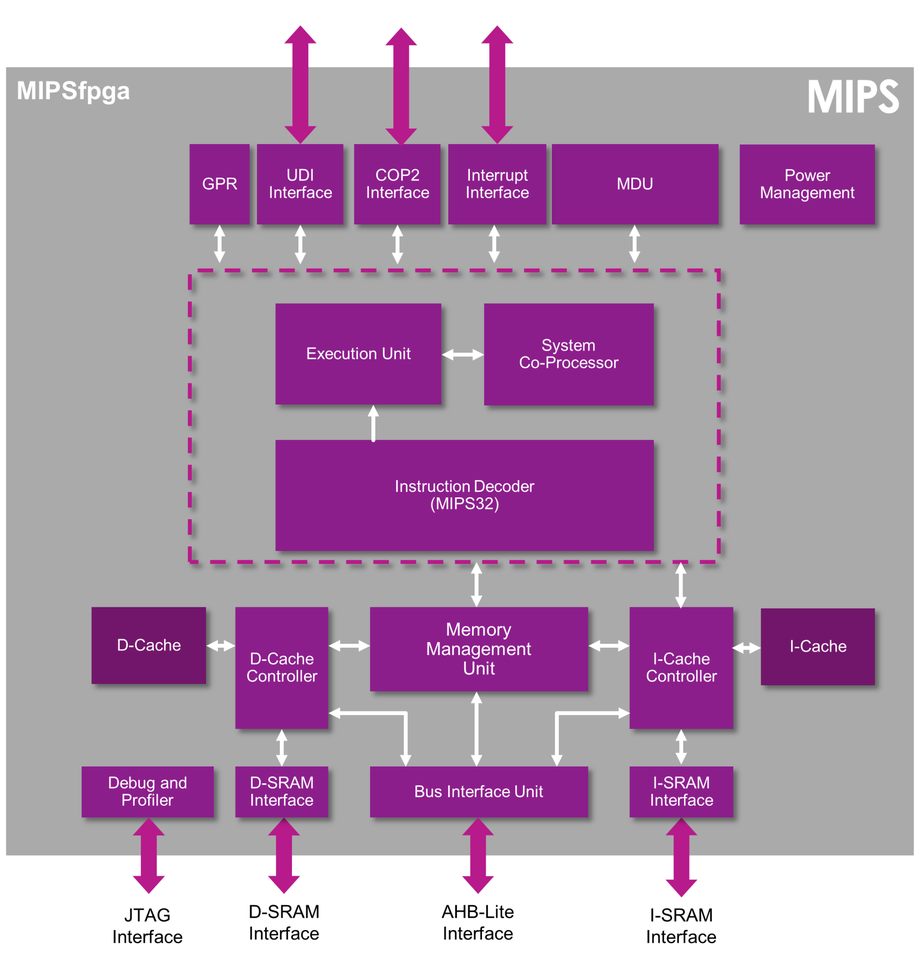
GSG Package includes:

- Unobfuscated microAptiv UP soft-core processor in Verilog.
- Thorough getting started guide (200 pages document).
- Installers for the programming and debugging tools.
- Set of scripts and examples.

The MIPSfpga System:



The microAptiv core:



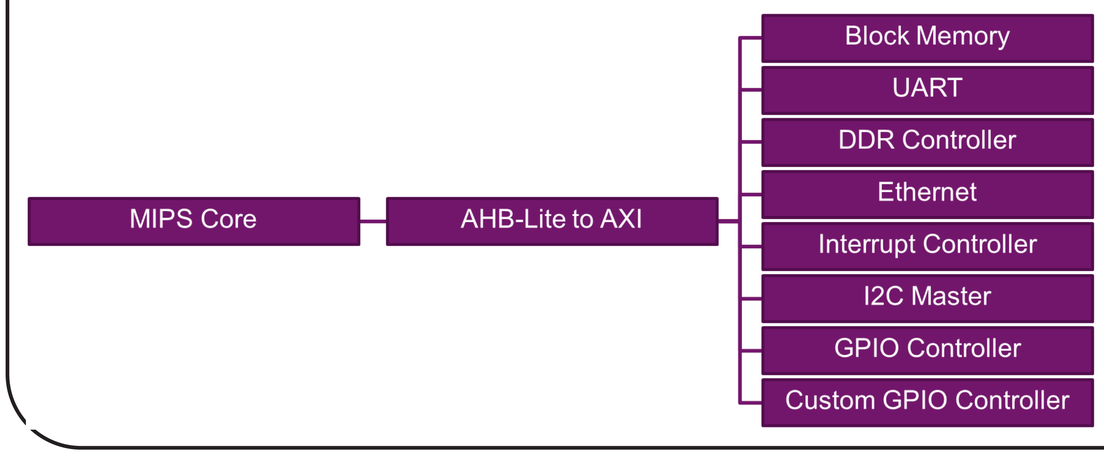
2nd package: MIPSfpga Labs

Lab	Description
1	Create a Project in Vivado or Quartus-II
2	Learn how to compile, debug and run C programs
3	Learn MIPS Assembly Programming system
4	More C Programming Practice (optional)
5	Expand the system to add 7-segment displays
6	Expand the system to add a counter
7	Expand the system to add buzzer
8	Expand the system to add SPI-Light Sensor
9	Expand the system to add a SPI-LCD
10	Interact with peripherals using interrupts
11	Build a DMA engine to drive peripherals interactions
12	Build a Data Encryption Standard encryption engine
13	Learn how to use the Performance Counters
14	Execution of an ADD and other arithmetic instructions
15	Execution of an AND and other logic instructions
16	Execution of a LW and other related instructions
17	Execution of a BEQ and other related instructions
18	Learn how the Hazard Unit is implemented
19	Learn how to use the CorExtend interface
20	Introduction to the caches available in MIPSfpga
21	Analyse the D$ and implement new configurations
22	Cache Controller: Analyse a cache hit and miss
23	Cache Controller: Analyse D$ management policies
24	Cache Controller: Analyze the Store and Fill Buffers
25	Implement an Instruction Scratchpad RAM

3rd package: MIPSfpga SoC

MIPSfpga SoC extends the original MIPSfpga system for:

- Building an SoC system using the microAptiv core.
- Loading a Linux operating system on the new SoC.



A Practical Experience

Along the second semester of year 2016/2017 (Feb-2017 to June-2017), we are using MIPSfpga v2.0 as part of an undergraduate course on *Computer Architecture & Integrated Systems*, which is part of the fourth year of the *Telecommunications Engineering* degree offered at University Complutense of Madrid.

The course includes 23 1,5-hour lectures interleaved with 13 2-hour lab sessions. The course includes a first module on microarchitecture, where the MIPS ISA and the single/multi-cycle processors are reviewed, and the pipelined processor is explained in detail. Module 2 reviews the Input/Output system and analyzes the cache hierarchy and the AMBA specification. Finally, a third module introduces System on Chip and Embedded System design. Thus, MIPSfpga v2.0 is perfectly suited for this course.