Synthesis of Digital Systems
Lab Lecture -1

Getting started with ZedBoard

Munish Jassi

Organisation and Laboratory Content

Desired Learning from this Lab

- Understanding of System Prototyping.
- Understanding HW/SW interfacing for SoCs.
- High Level Synthesis (HLS) and its usage for Hardware Accelerated systems.

Content of the Lab

- System prototyping on FPGA and associated software tool chain.
- Industry standard High Level Synthesis (HLS) tool chain.
- Implementation of video processing hardware System and software interface.
- Hardware Accelerator (HA) for performance improvement.

Organization

- Lab has four sections, 3 weeks for each section:
  - Lab A – Introduction to ZedBoard
  - Lab B – Introduction to HLS using Vivado_hls
  - Lab C – Video processing on ZedBoard (SW only)
  - Lab D – Hardware Accelerated video processing
- Submissions at the end of each lab according to the schedule,
  - Code submission.
  - Submission of the Questionaire.
  - Results submission (if any).

Grading

- Lab makes 50% of the SDS course.
- Final Grading:
  - Lab code submissions and reports (25% of the final grade)
    - Code submission (1/2 of points)
    - Reports (1/2 of points)
  - End semester exam (75% of the final grade):
    - 2/3 points on SDS Lectures.
    - 1/3 points on Lab-section.

Schedule

- Lab lectures:

Date	Topic
09.04.2014 (Room:2977)	Lab A – Getting started with ZedBoard
30.04.2014 (Room:2977)	Lab B – HLS using Vivado_hls
21.05.2014 (Room:2977)	Lab C – Video processing SoC, SW only implemention
11.06.2014 (Room:2977)	Lab D – Video processing SoC, HA implementation

- Submission deadlines:

Date	Topic
29.04.2014 (23:55hr)	Lab A Submissions
20.05.2014 (23:55hr)	Lab B Submissions
10.06.2014 (23:55hr)	Lab C Submissions
01.07.2014 (23:55hr)	Lab D Submissions

Lab Administration

- Presence in the Lab lectures is mandatory (Inform supervisor if not possible to attend in advance).
- Each group with 1 to 2 people.
- Work on your own schedule in Room 2902.
- Individual working directories are at this path,

  /usr/local/labs/SDS/current/<login>/
- Project material for each lab is available at,

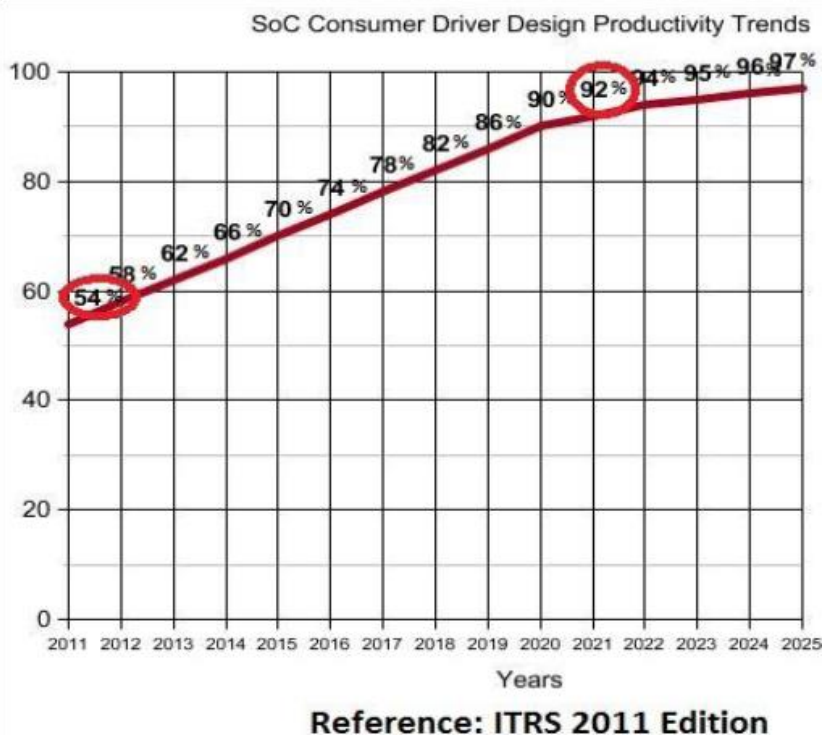
  /usr/local/labs/SDS/current/<login>/project_documents/
- Code submission must be in this path,

  /usr/local/labs/SDS/current/<login>/Lab<>
- Tutor Hours: Wednesday, 16:45-18:15, Room 2902
  - Benjamin Bordes, benjamin.bordes@tum.de

Lab-A
Getting Started with ZedBoard

Motivation

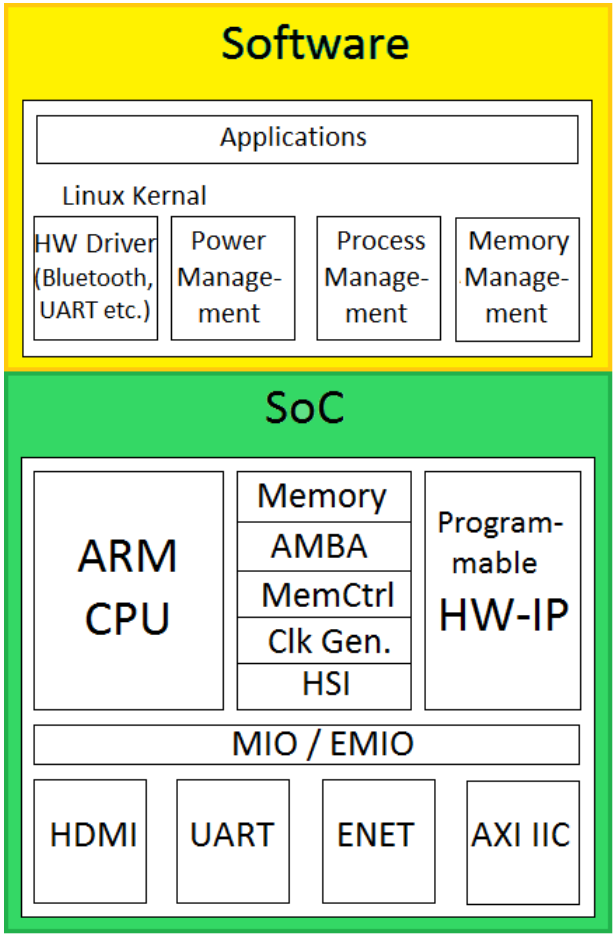
- Designer's productivity is not increasing proportional to SoC design complexity.
- IP-reuse to be more than 90% by 2020.
- Chips like Xilinx-Zynq, provides scope for integration of customized HW into SoC.



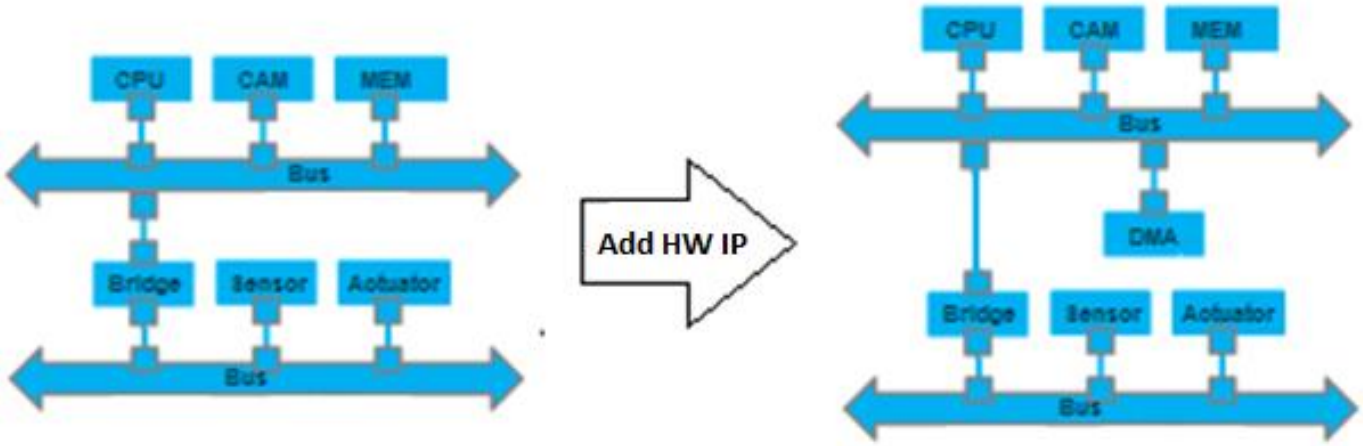
System On Chip – HW/SW Partitioning

- Main elements for hardware System,
  - CPU
  - Memories
  - Bus system
  - Peripherals

- Main elements for Software System,
  - Operating System,
    - HW drivers
    - Process Management
    - Power/Memory management
  - Applications (with/without OS)

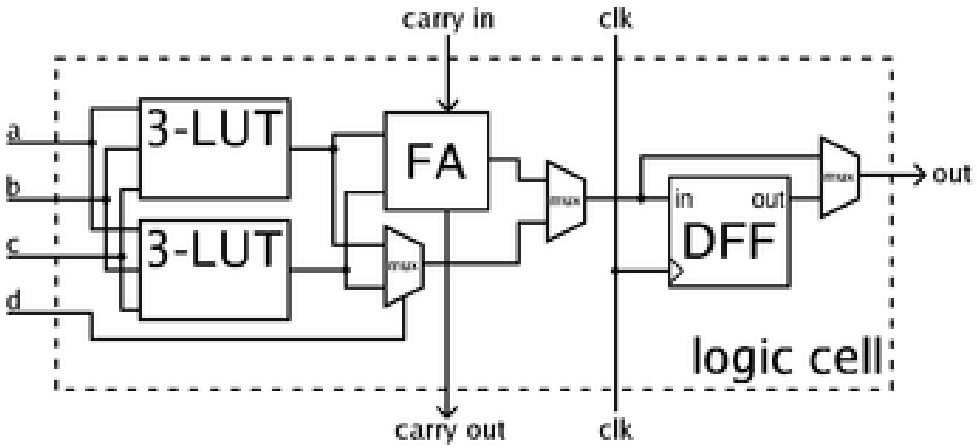


System On Chip – HW System Architecture

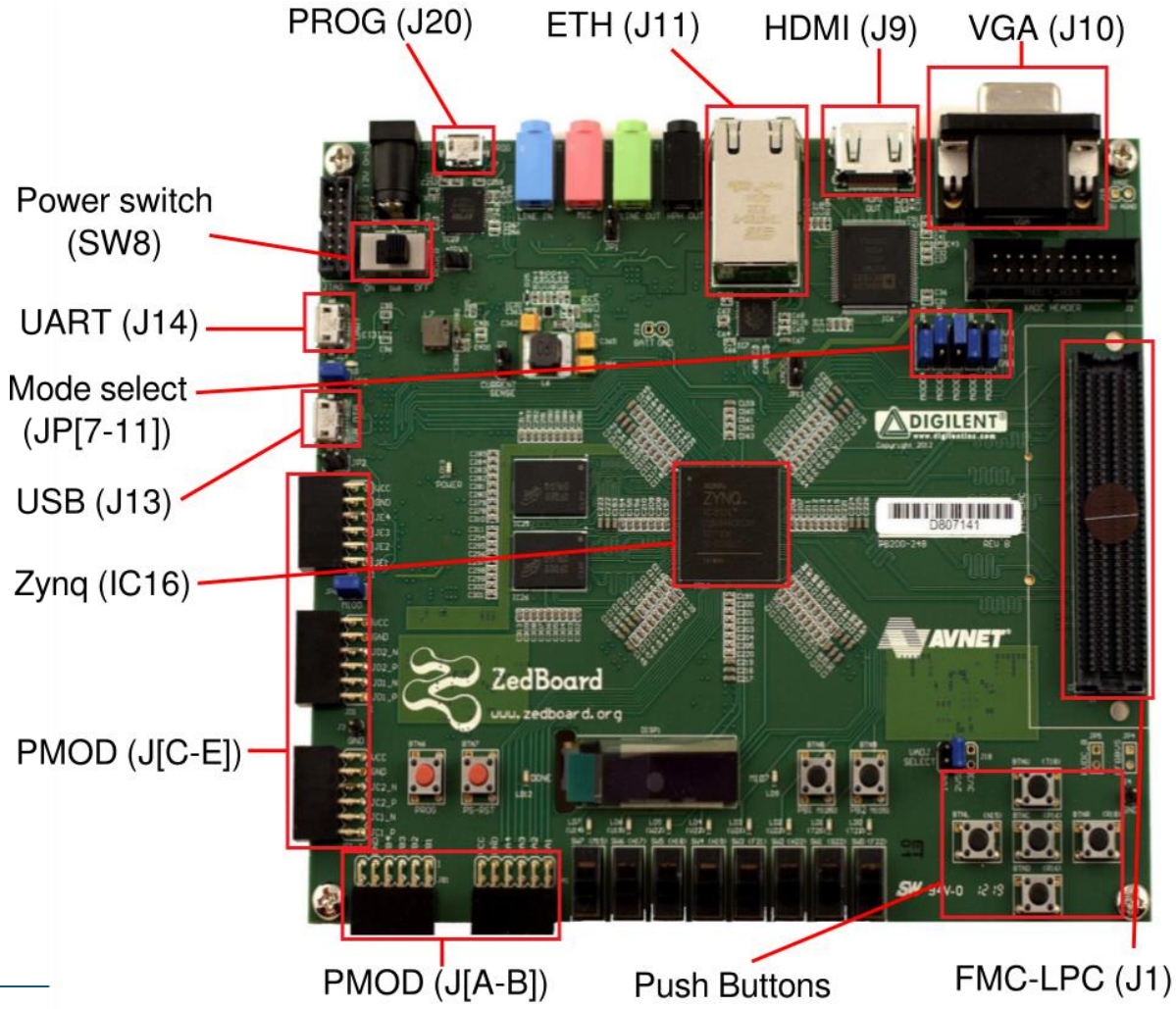


FPGA – Field Programmable Gate Array

- Reconfigurable hardware via programmable interconnections.
- LUTs are basic blocks within each logic cell.
- Data flow is controlled by programmable MUXs.
- FPGA provides more flexibility than ASIC, on cost of higher power and lower performance.



ZedBoard


PROG (J20)
ETH (J11)
HDMI (J9)
VGA (J10)
Power switch (SW8)
UART (J14)
Mode select (JP[7-11])
USB (J13)
Zynq (IC16)
PMOD (J[C-E])
PMOD (J[A-B])
Push Buttons
FMC-LPC (J1)
DIGILENT
AVNET
ZedBoard
www.zedboard.org

ZedBoard

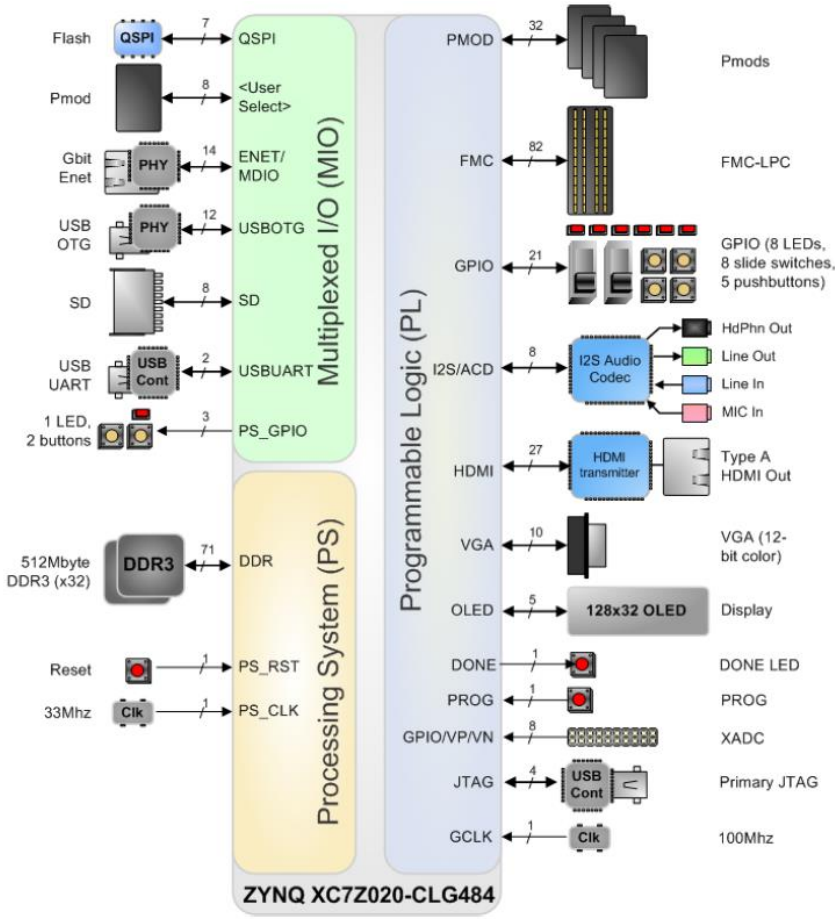
- **Zynq-7000 AP SoC XC7Z020-CLG484**
- **Memory**

  512 MB DDR3 , 256 Mb Quad-SPI Flash, SD card
- **Communication**

  USB-JTAG , Ethernet, USB OTG 2.0 and USB-UART
- **Expansion connectors**

  FMC-LPC, 5 Pmod™ headers
- **Display**

  HDMI, VGA output, OLED display
- **General Purpose I/O**



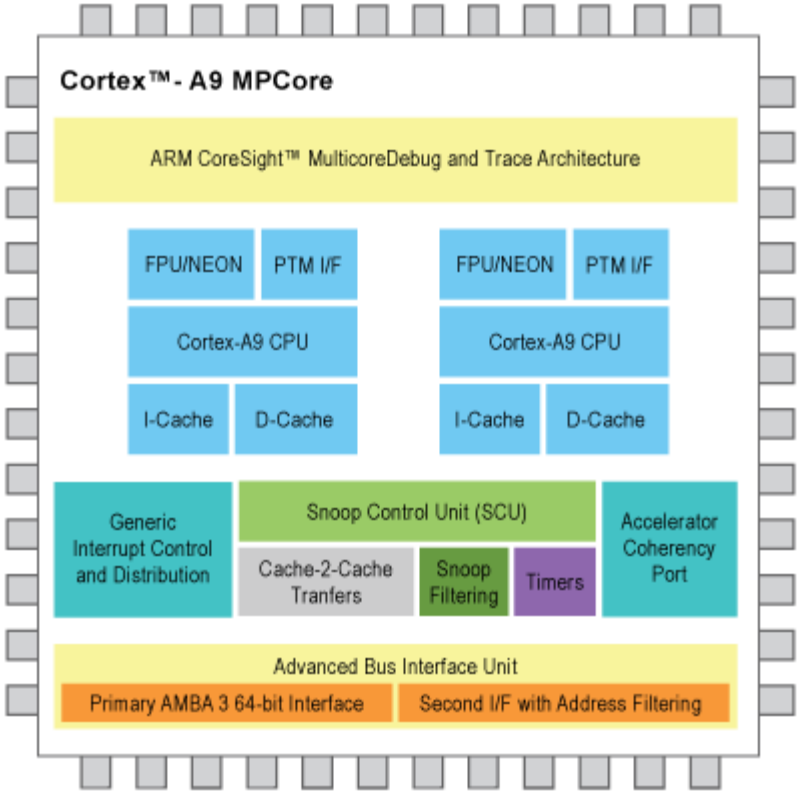
Ref.: www.zedboard.org

ZedBoard

- Clocking
  - 33.33333 MHz clock source for PS
  - 100 MHz oscillator for PL
- Display
  - HDMI output supporting 1080p60 with 16-bit
  - VGA output (12-bit resolution color)
  - 128x32 OLED display
- General Purpose I/O
  - 8 user LEDs
  - 7 push buttons
  - 8 DIP switches

ARM-Cortex A9

Architecture	ARMv7-A Cortex
Dhrystone Performance	2.50 DMIPS/MHz per core
Multicore	1-4 cores
ISA Support	• ARM • Thumb-2/Thumb • Jazelle • DSP extension • Advanced SIMD NEON unit • Floating Point Unit

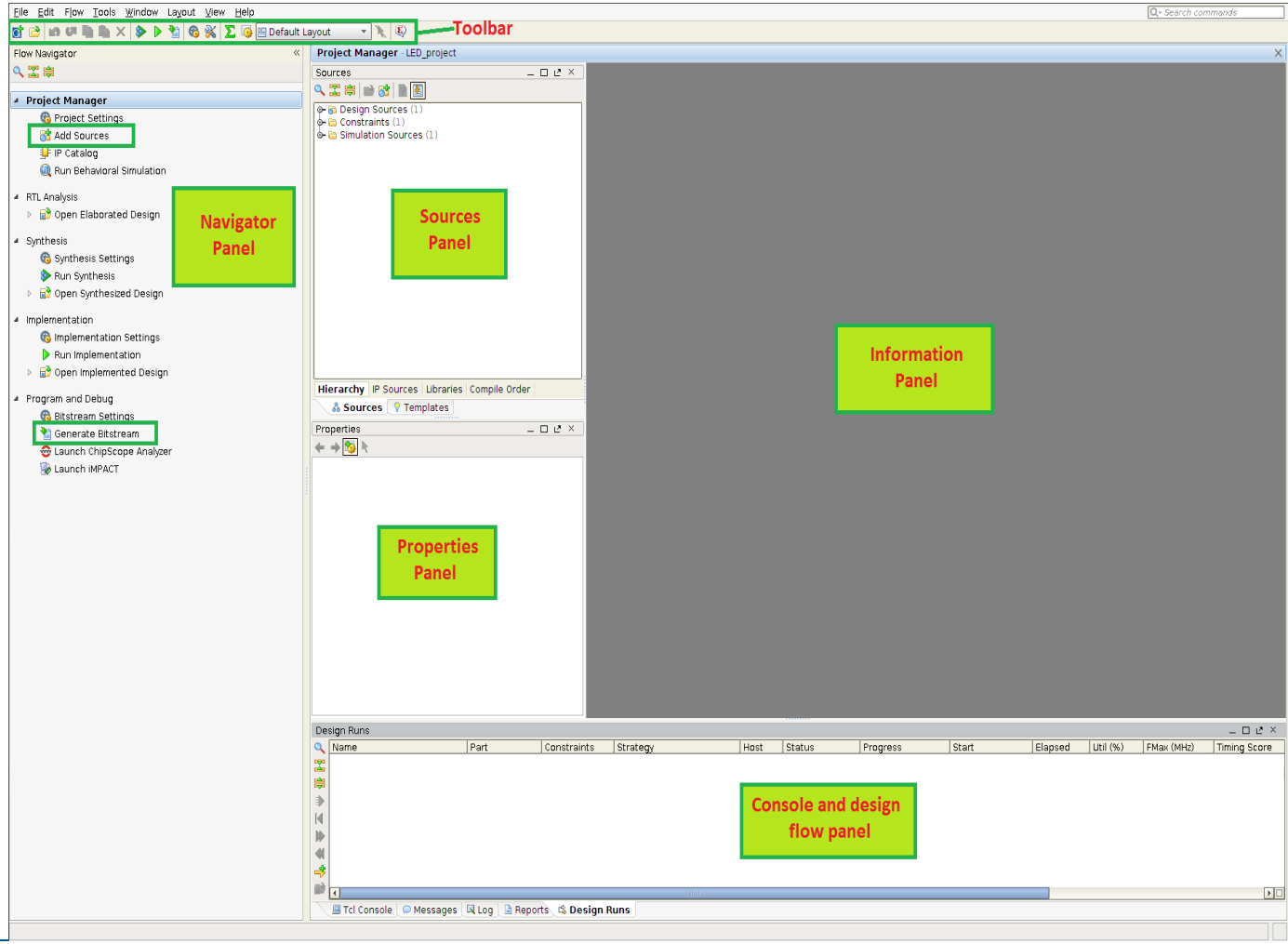


Ref.: www.altera.com

Xilinx Tool Chain
for ZedBoard

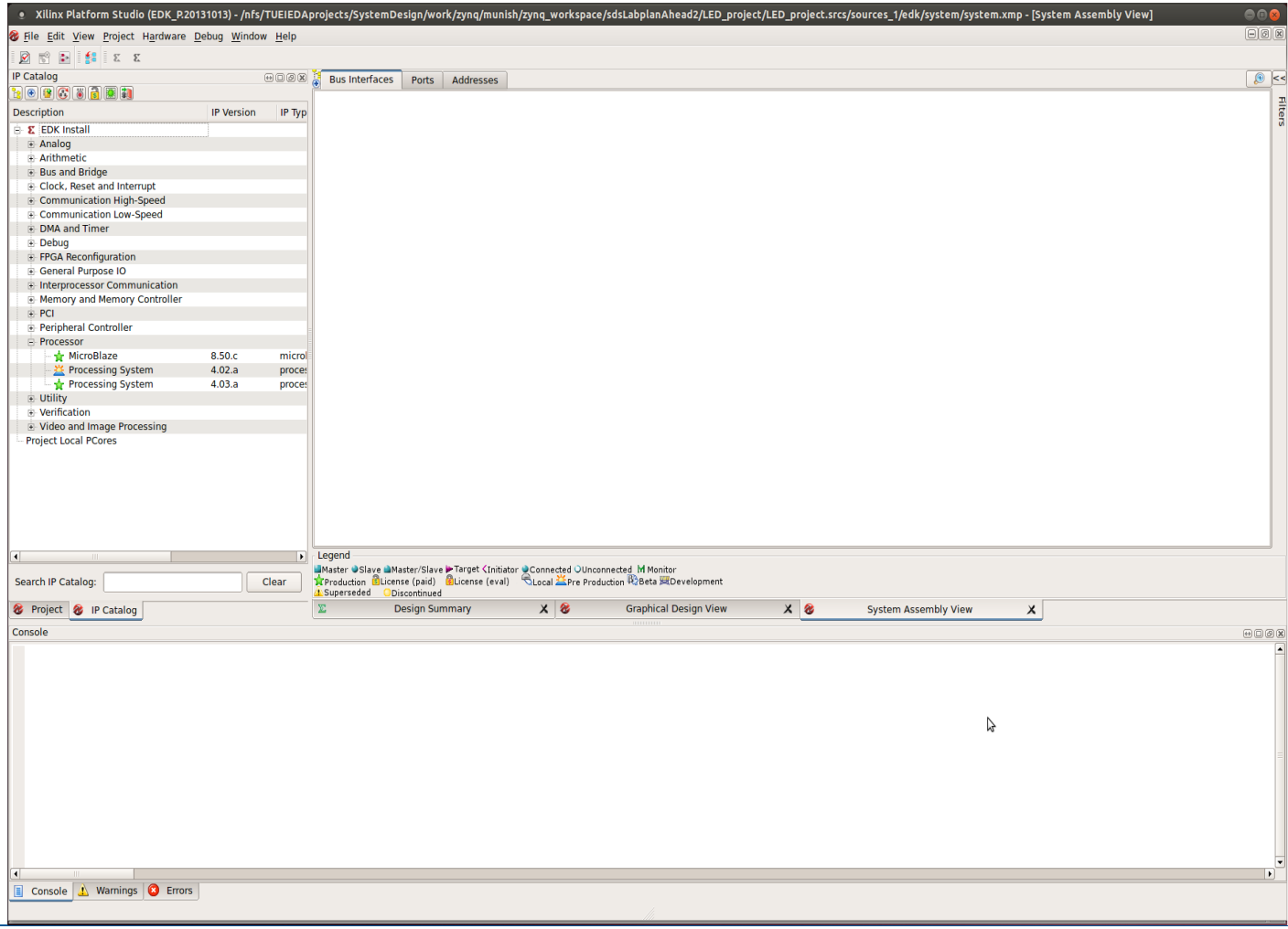
Xilinx - PlanAhead

- Eclipse based GUI.
- Central tool for ZedBoard project.
- Define the Board configurations.
- Include the project Sources and Constraints.
- Top-level Netlist generation.
- Invoking other Tools.



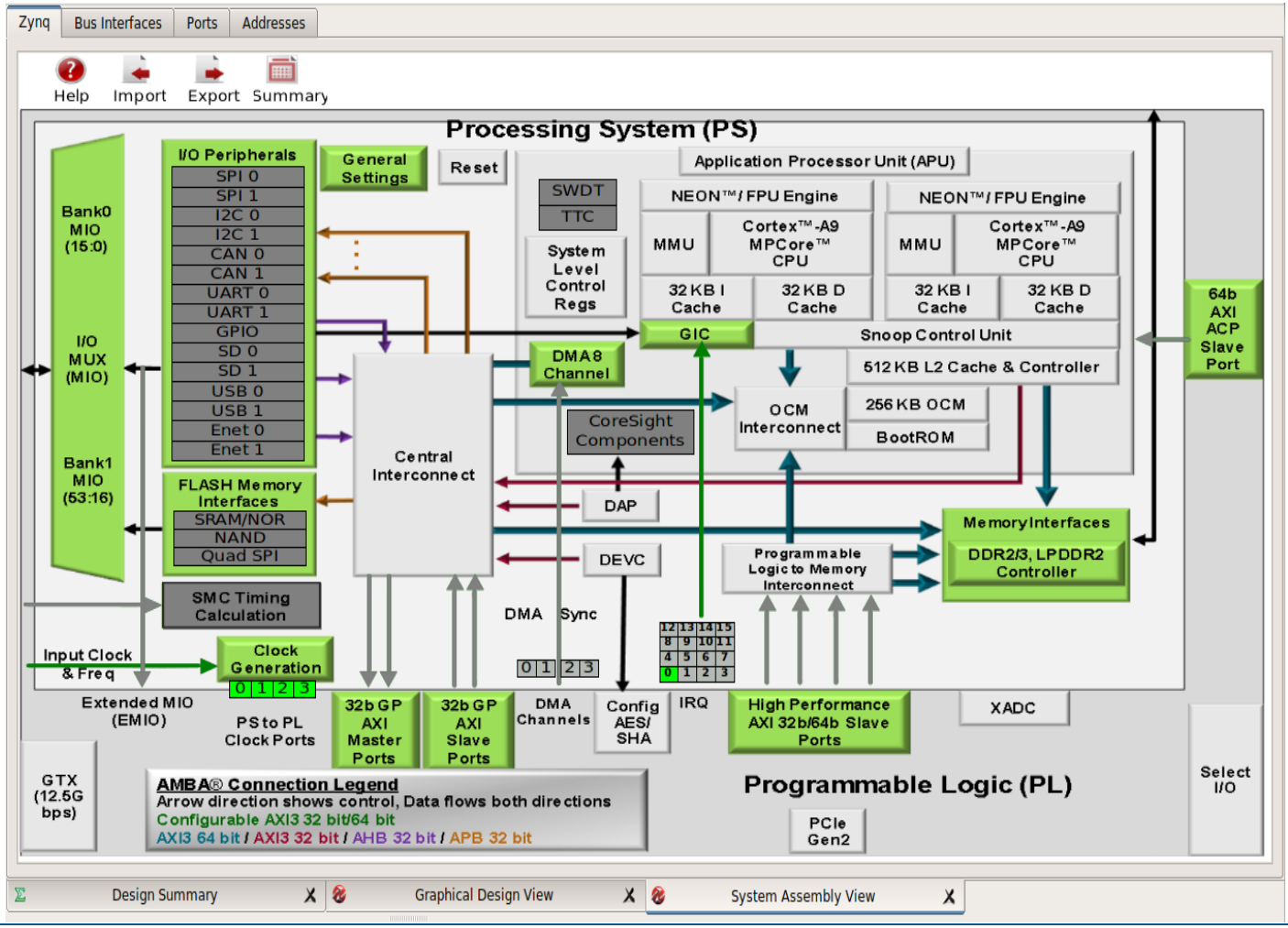
Xilinx Platform Studio – XPS

- Hardware configuration of Embedded cores.
- Integration of peripheral HW IPs.
- Bus & Port connections.
- Address mapping.
- Clocks/Memory configuration.



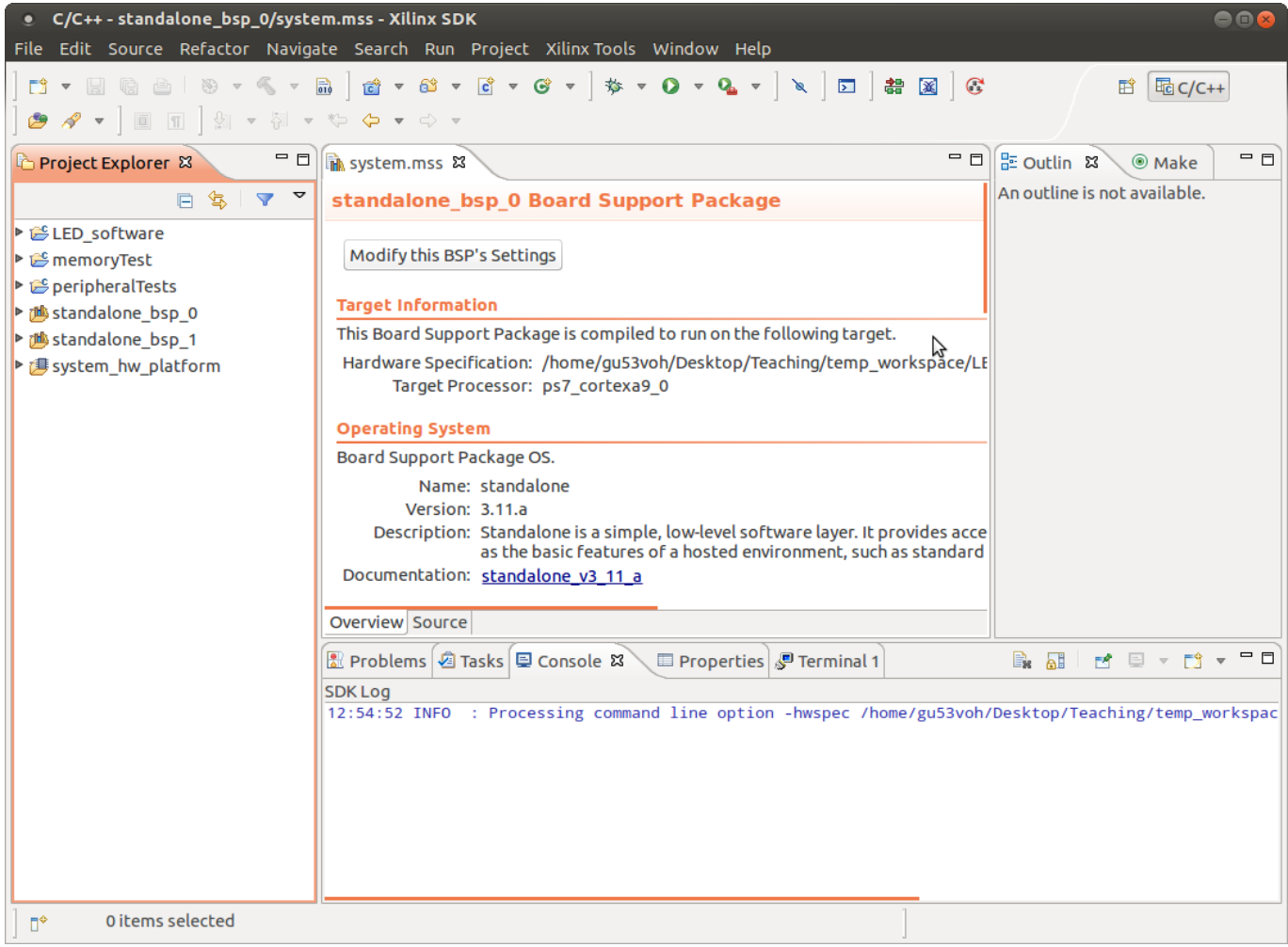
Xilinx Platform Studio – XPS

- Integration of peripheral HW IPs.
- Configuration of Processing System.
- Clocks/Memory configuration.

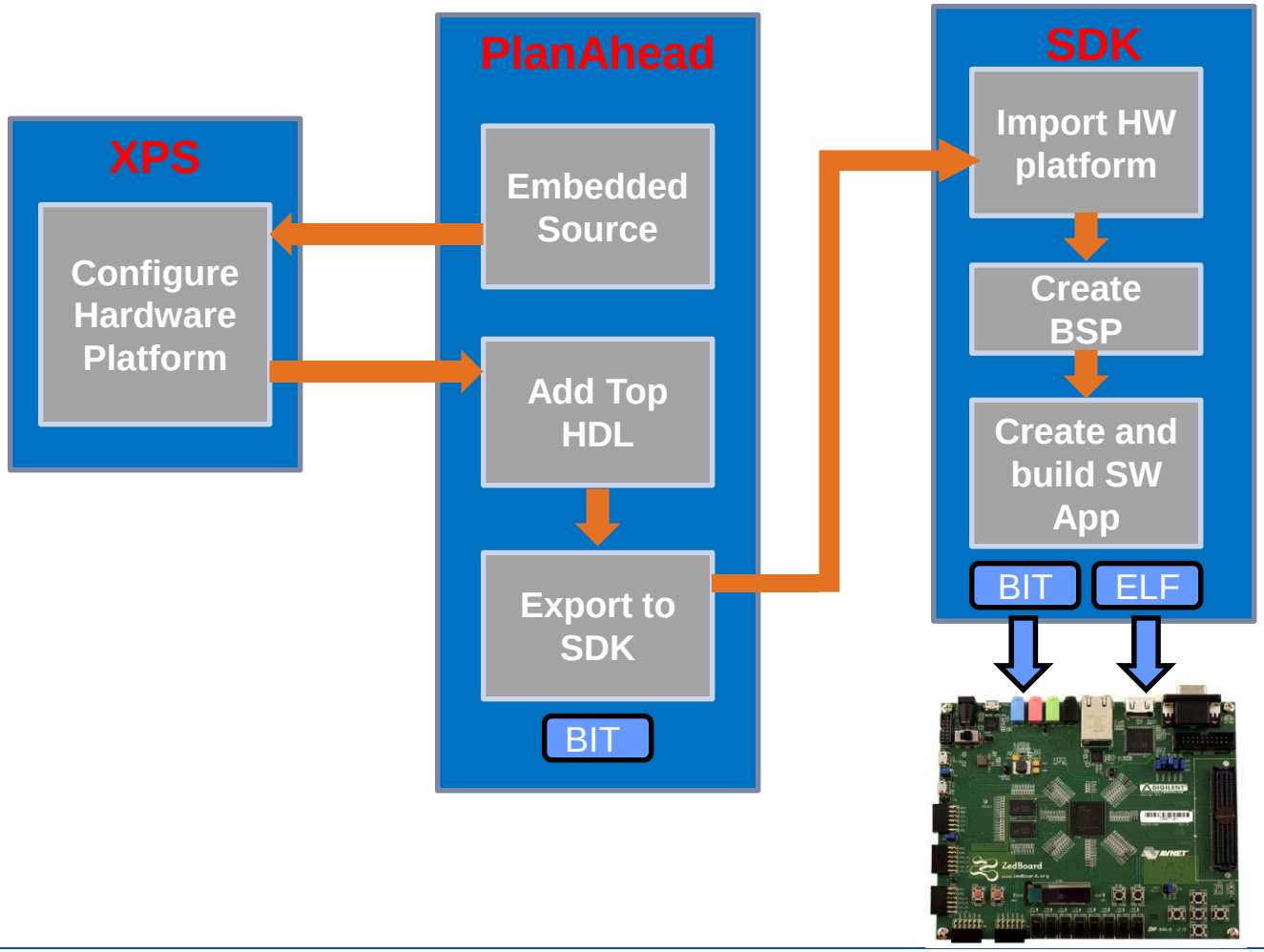


Xilinx Software Development Kit – SDK

- Eclipse type GUI.
- ARM Cross compiler to compile C/C++ application code for ARM CPU.
- Board Support Package (BSP), provide the driver header files for System peripherals.
- Program the ZedBoard from SDK.



Xilinx Tools Chain



“Hello World” Exercise
Getting familiar with Xilinx’s Tool Chain

Get started with Xilinx Tools

1. PlanAhead, XPS and SDK: 14.7 version
2. Paths used in Lab manual,
   - **<work dir>** : Working directory provided to you for lab-work. */usr/local/labs/SDS/current/<your_lrz_account>/*
   - **<project docs>** : Directory provided with initial set of materials for labs. */usr/local/labs/SDS/current/<your_lrz_account>/project_documents*
   - **<$dir >** : Path to directory ”dir”.
   - **<project name>** : Name of the active Lab project.
3. To load environmental settings, *$module load xilinx/ise/14.7*
4. To start tool, PlanAhead: *$planAhead*

ZedBoard – Hello World

- Project is already completed, and is to get familiar with Tool chain
- Copy the HelloWorld directory to <work dir> and start planAhead there.
- Load HelloWorld.ppr on planAhead.
- On *PlanAhead,* double-click *system_i – system (system.xmp)* to open XPS.
- On *PlanAhead*, select *File> Export Hardware for SDK..* and select *Launch SDK* to open SDK.
- Connect hardware and click *Program FPGA.*
- Click *Run configuration…* and set the appropriate paths for *system_stub.v* and *ps7_init.tcl* and click *Run* to execute the project.

Thank You

for your participation