INTRODUCTION TO FPGA PROGRAMMING

Lesson 10: IP Blocks

Dr. Davide Cieri[1]

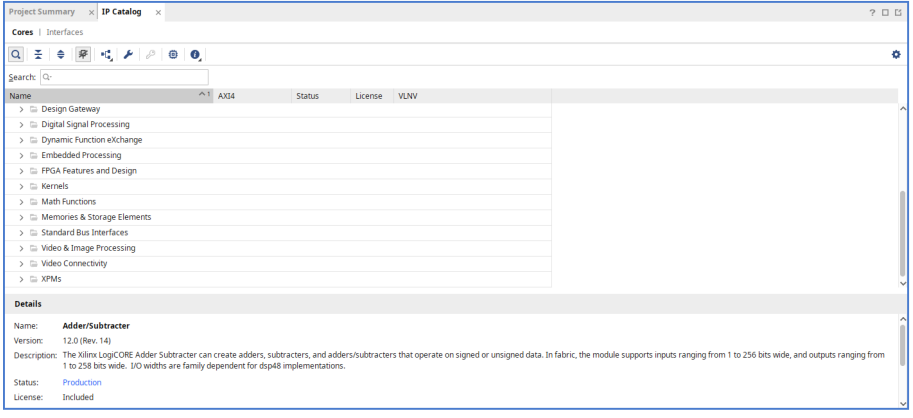
[1]Max-Planck-Institut für Physik, Munich

September 2024

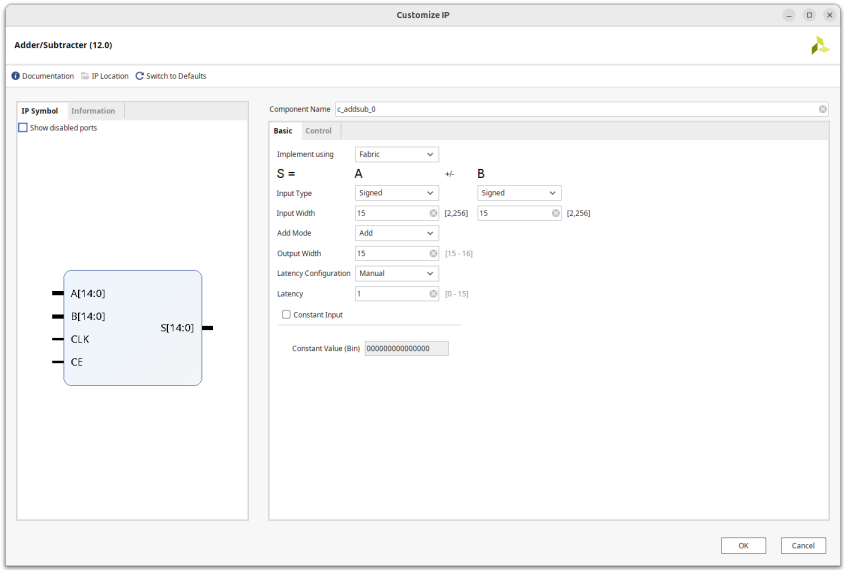
MAX-PLANCK-INSTITUT FÜR PHYSIK

WHAT ARE IP BLOCKS

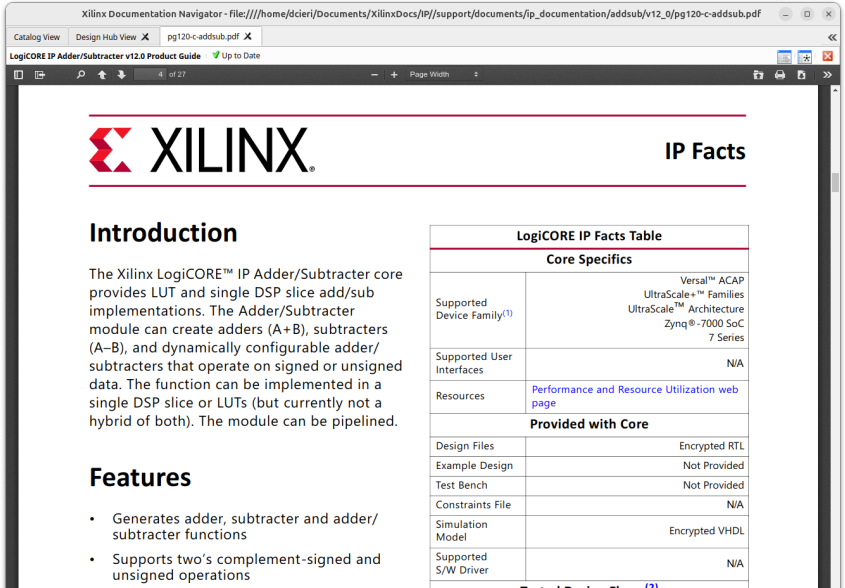
- IP (Intellectual Property) blocks or IPs are a collection of pre-built configurable blocks provided by the FPGA vendor or third-parties
- Encrypted synthesised blocks with full simulation models



IP GENERATION



IP DOCUMENTATION



XILINX®

IP Facts

Introduction

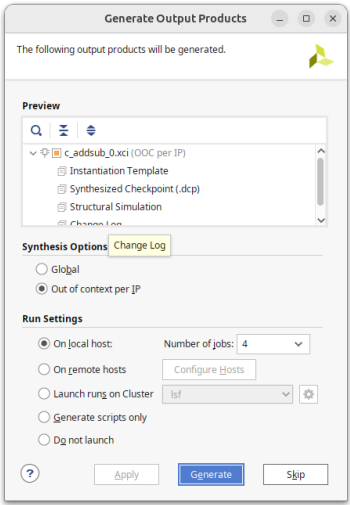
The Xilinx LogiCORE™ IP Adder/Subtracter core provides LUT and single DSP slice add/sub implementations. The Adder/Subtracter module can create adders (A+B), subtracters (A–B), and dynamically configurable adder/subtracters that operate on signed or unsigned data. The function can be implemented in a single DSP slice or LUTs (but currently not a hybrid of both). The module can be pipelined.

Features

- Generates adder, subtracter and adder/subtracter functions
- Supports two's complement-signed and unsigned operations

LogiCORE IP Facts Table	
Core Specifics	
Supported Device Family(1)	Versal™ ACAP UltraScale+™ Families UltraScale™ Architecture Zynq®-7000 SoC 7 Series
Supported User Interfaces	N/A
Resources	Performance and Resource Utilization web page
Provided with Core	
Design Files	Encrypted RTL
Example Design	Not Provided
Test Bench	Not Provided
Constraints File	N/A
Simulation Model	Encrypted VHDL
Supported S/W Driver	N/A

IP GENERATED FILES



- Generated products of the IP includes:
  - Instantiation Template to add IP to your design (next slide)
  - Synthesis Checkpoint
  - Structural Simulation, to allow simulation of IP
  - Changelog
- IP Synthesis options
  - Out-of-context: IP is synthesised as a standalone module
  - Global: IP is synthesised within your design. Any change to the design will require to resynthesise the IP as well. Not recommended.

IP INSTANTIATION IN VHDL

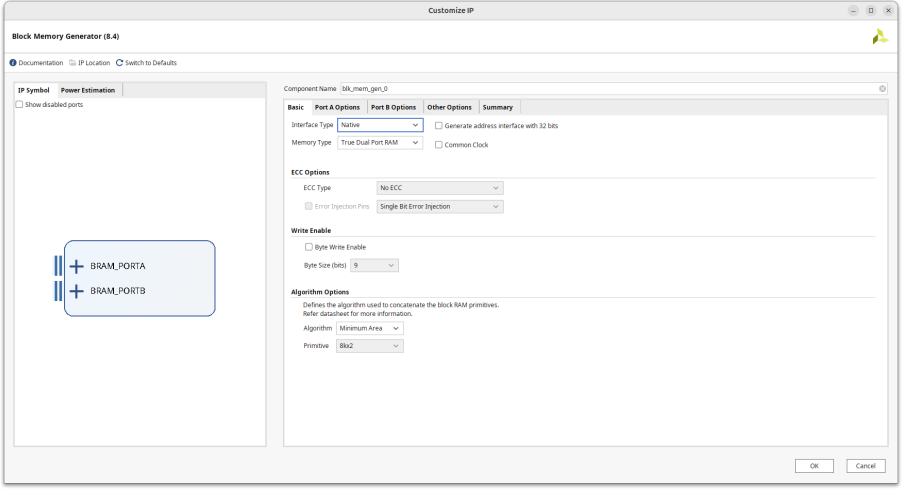
c_addsub_0.vho

/home/dcieri/Work/fpga-course-tum/labs/lab01/project_1/project_1.gen/sources_1/ip/c_addsub_0/c_addsub_0.vho

Read-only

```
-- IP Revision: 14

-- The following code must appear in the VHDL architecture header.

------------ Begin Cut here for COMPONENT Declaration ------ COMP_TAG
COMPONENT c_addsub_0
  PORT (
    A : IN STD_LOGIC_VECTOR(14 DOWNTO 0);
    B : IN STD_LOGIC_VECTOR(14 DOWNTO 0);
    CLK : IN STD_LOGIC;
    CE : IN STD_LOGIC;
    S : OUT STD_LOGIC_VECTOR(14 DOWNTO 0)
  );
END COMPONENT;
-- COMP_TAG_END ------ End COMPONENT Declaration ------------

-- The following code must appear in the VHDL architecture
-- body. Substitute your own instance name and net names.

------------ Begin Cut here for INSTANTIATION Template ----- INST_TAG
your_instance_name : c_addsub_0
  PORT MAP (
    A => A,
    B => B,
    CLK => CLK,
    CE => CE,
    S => S
  );
-- INST_TAG_END ------ End INSTANTIATION Template ---------

-- You must compile the wrapper file c_addsub_0.vhd when simulating
-- the core, c_addsub_0. When compiling the wrapper file, be sure to
```

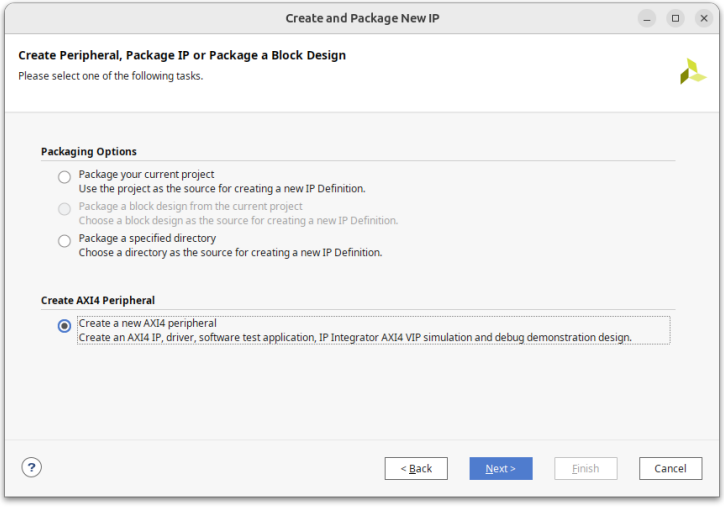
ANOTHER EXAMPLE: BLOCK RAM



USER IPS

- You can pack your design into an IP that can be later placed in a repository
- User IP can me parametrised thanks to generics/generate
- You can include documentation, instantiation templates, and constraint files (even pin locations)
- Include a test-bench or design example
- Customise the IP generation GUI

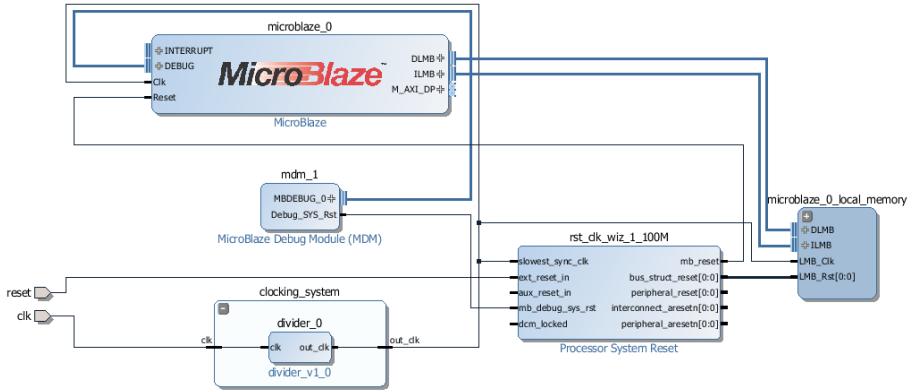
USER IP CREATION



IP INTEGRATOR

- Vivado provides a smart schematic editor to connect IPs together into a Block Design
  - Including user IPs
- Legal connections are highlighted
- Designs are validated
- Block Designs can be auto wrapped into a VHDL file
  - Wrapper can be used as top level of your design
- Alternatively, they can be instantiated as a normal component

BLOCK DESIGN EXAMPLE



LAB 16: USING IPS

The figures in these slides are taken from:

- Digital Design: Principles and Practices, Fourth Edition, John F. Wakerly, ISBN 0-13- 186389-4. ©2006, Pearson Education, Inc, Upper Saddle River, NJ. All rights reserved
- allaboutfpga.com
- nandland.com
- docs.amd.com
- https://www.symmetryelectronics.com/
- https://www.edn.com/
- Stephen A. Edwards, Columbia University, Fundamentals of Computer Systems, Spring 2012
- https://medium.com/well-red/state-machines-for-everyone-part-1-introduction-b7ac9aaf482e