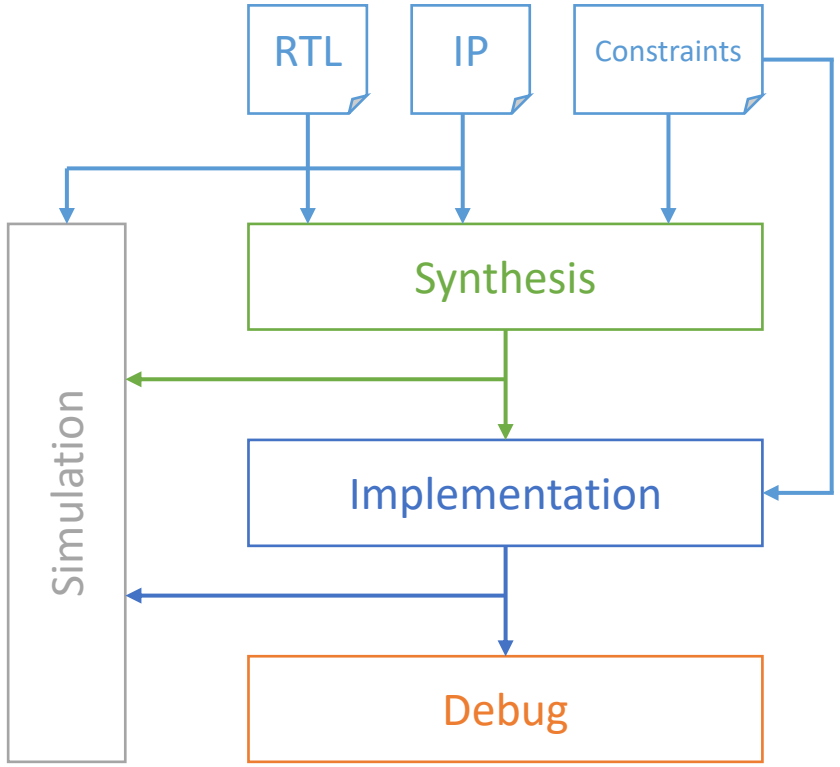
Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
Fondazione ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

FPGA Design Flow

Andrea Triossi – University of Padova – INFN Padova

Design Flow

- RTL source files
- Intellectual Property (IP)
- Constraint files
- Simulation
- Synthesis
- Implementation
- Debug



Synthesis

- Transforming an RTL design into a gate-level representation
- Start from
  - HDL languages (VHDL, Verilog, mixed languages…)
    - Coding Techniques
  - RTL attributes
    - Mapping style of certain part of design, registers and nets preservation or controlling design hierarchy
  - Constraint files (optional)
    - Timing constraints
- The outcome is a technology mapped netlist

Implementation

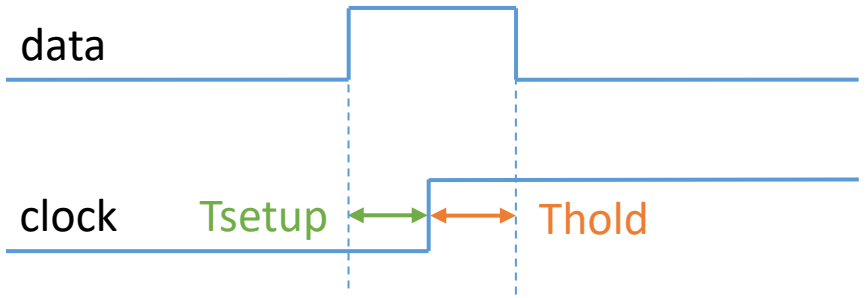
- Optimization: optimizes logical design to fit into target device and to reduce power demands
- Place: places the design onto the device and performs fanout replication to improve timing
- Post-place optimization: optimizes power, logic and placement using estimated timing
- Route: Routes the design onto the target device
- Post-route optimization: optimizes logic, placement and routing using actual routed delays
- Write bitstream: generate a bitstream that contains the device configuration

Implementation

- Design constraints
  - Physical
    - Pin location
    - Absolute/relative location of cells (Block RAM, DSP, LUT, FF…)
  - Timing
  - Power
    - Voltage and current settings
    - Switching rate

Timing violations

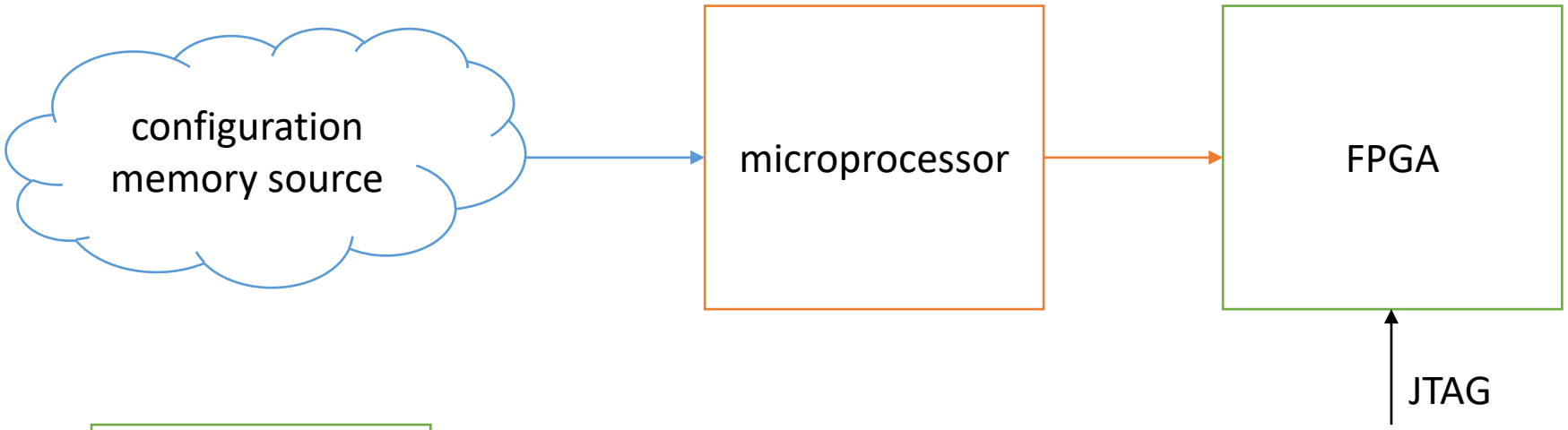
- Setup time is the minimum amount of time a synchronous data input should be held steady before the clock event so that the data input is reliably sampled by the clock event
- Hold time is the minimum amount of time a synchronous data input should be held steady after the clock event so that the data input is reliably sampled by the clock event
- Tsetup + Thold is an indication of the quality of the technology

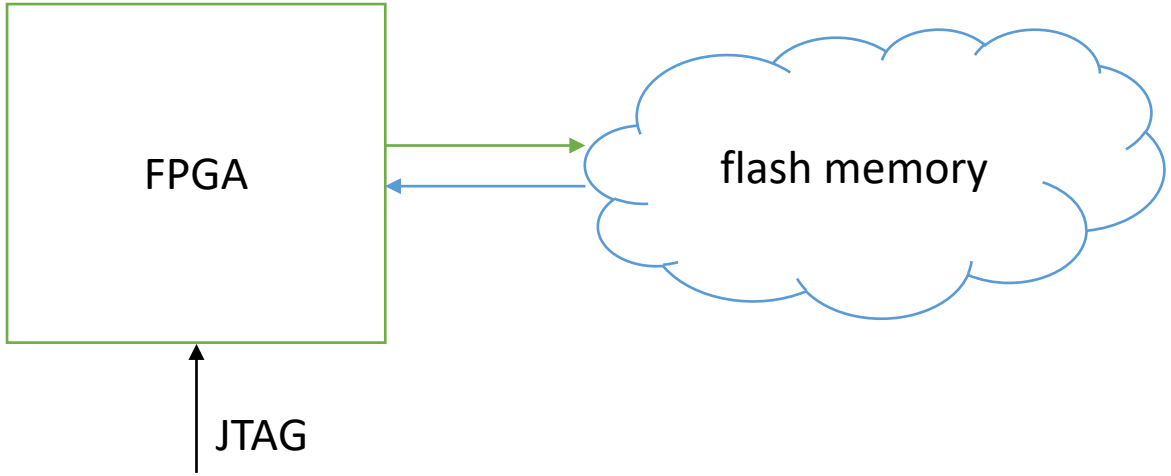


FPGA configuration

- SRAM is a volatile memory -> data is lost at power off
- At each power on a SRAM based FPGA needs to be configured
- External source for configuration
  - Loaded by FPGA from a non-volatile memory
  - Loaded to FPGA by a processor/controller
- Connections available
  - Serial for minimizing pins number
  - Parallel for performance
  - JTAG port any time available

FPGA configuration





When the flash is programmed by JTAG through the FPGA is called indirect programming

FPGA configuration

- Dynamic reconfiguration is intended for changing configuration of functional blocks, like CMT, XADC or fast transceiver, while they are operational
  - Dedicated port on the functional block that allow read and write on configuration memory
- MultiBoot and fallback allows to switch dynamically between images during reprogramming
  - The MultiBoot image is first loaded at power up from an upper address space
  - If this image fails configuration, the device automatically triggers a fall-back to the golden image stored at the bottom address
- Partial reconfiguration is an even more flexible FPGA reconfiguration
  - Only a region of the FPGA is reconfigured while the other part is operational
  - Time multiplex hardware dynamically

Intellectual property

- Modules that can be added to the design
  - Xilinx IP
  - Third-party IP
  - Designs packaged IP
  - High-Level Synthesis (HLS) IP
- IP are usually customizable
- Out-of-context synthesis
  - Speed up run time avoiding re-synthesizing
  - Simulation even without behavioral HDL

Simulation

- Emulating real design behavior in a software environment
- Verify code syntax and confirm the code behaves as expected
- RTL simulation is not architecture specific
- Timing simulation
- Simulation flow
  - Creating a testbench
  - Generating a netlist (only for post-synthesis or post-implementation)
  - Running simulation

Testbench

- In order to simulate a circuit, we need a series of stimuli and possibly an automatic way of checking results
- Testbenches are not synthesizable -> the full VHDL can be used
- A Testbench is an entity without ports, it must contain everything needed

Debug

- In-system debugging of the implemented design
  + Timing accurate
  + Actual system environment
  - Less design visibility than simulation
  - Longer iteration time
  - Debugging flow
    - Identifying the signals to probe and how to probe
    - Insertion in the design of the debug IP (at RTL or netlist level)
    - Design implementation with debug IP
    - Analysis of the output of the debug IP
- In-system high-speed I/O debugging