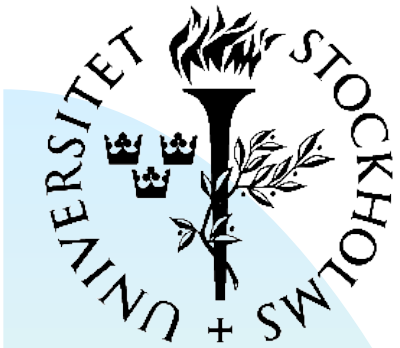


Introduction to Programmable Logic

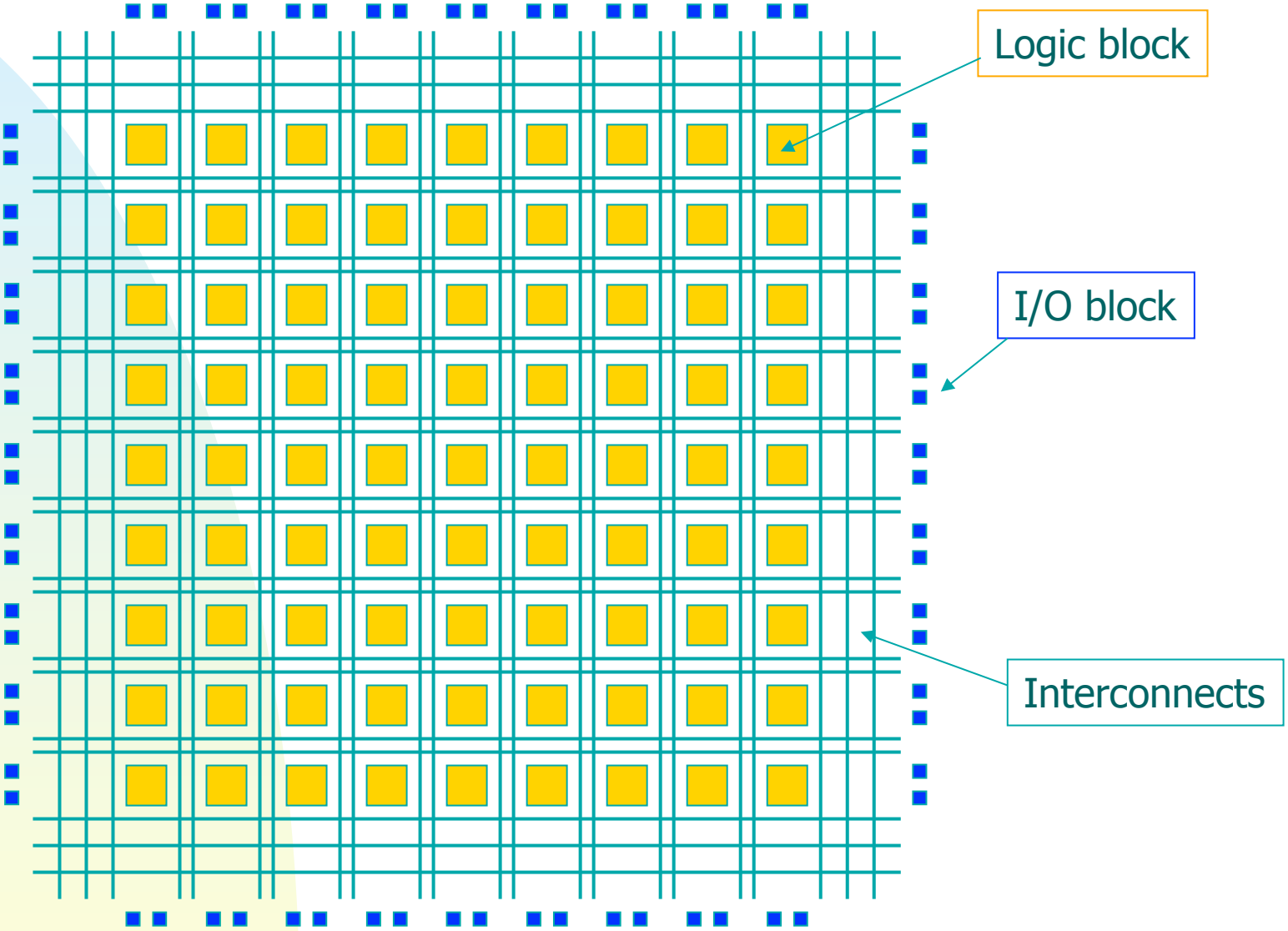
Nordic detector school

FPGA introduction

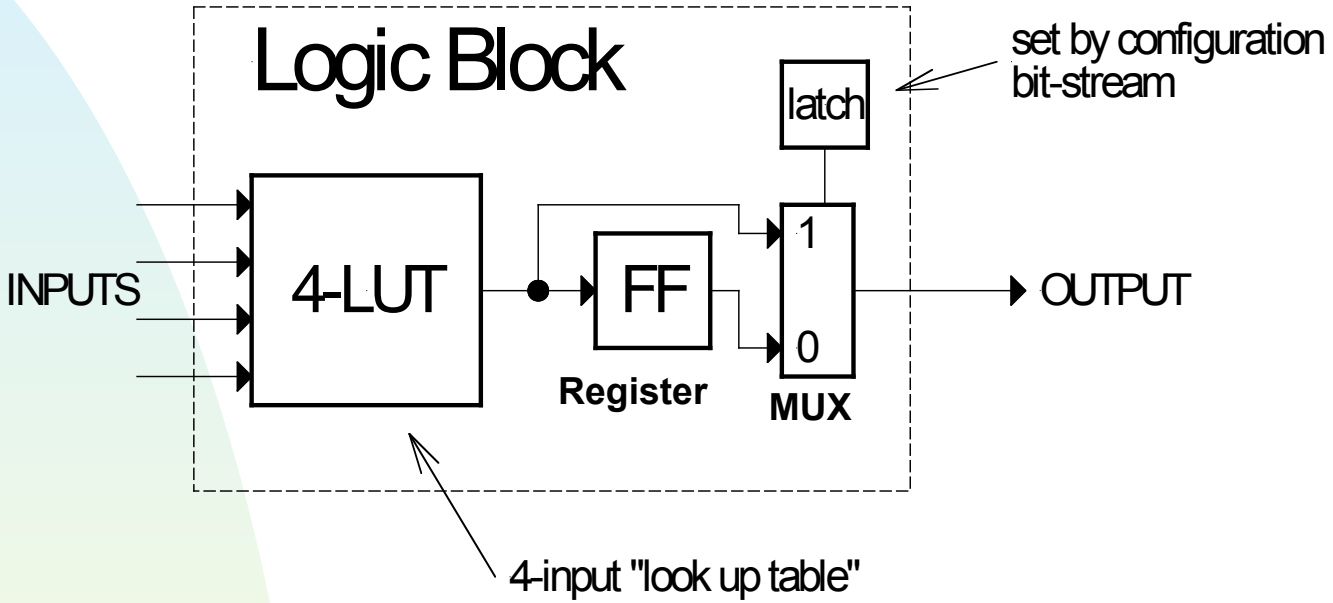
Introduction to VHDL

Implementation design flow

FPGA Structure



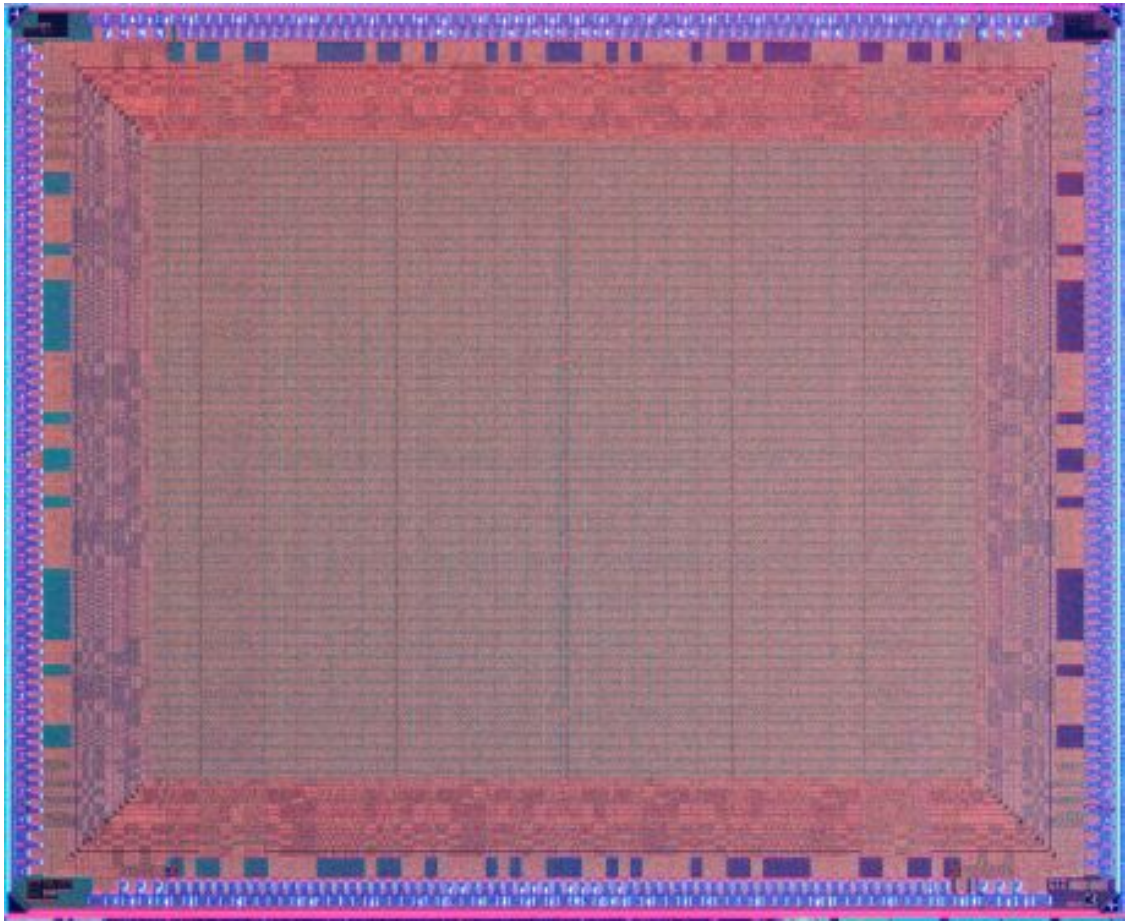
FPGA Logic Block (idealized)



- 4-input *look up table (LUT)*
  - implements combinatorial logic
- Register
  - optionally stores output of LUT

Field-Programmable Gate Arrays

- Xilinx Spartan-3 die image. Note the regularity



Other FPGA features:

- Dedicated block memories
  - Dual-port static RAM
- Digital clock management/synthesis
- Dedicated multipliers
  - Important for digital signal processing
- "Hard" CPU cores
  - Xilinx: PowerPC, ARM 9
  - Altera: Nios (Stratix II)
- Multi-Gbit transceivers

For this course:

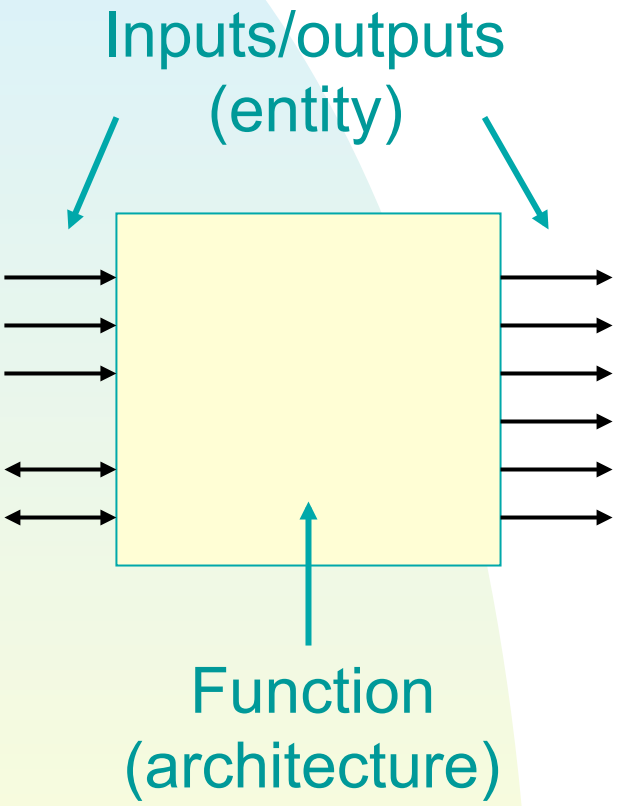
- Device: Xilinx FPGA (Spartan 3E)
- Design language: VHDL
- Simulation tool: Xilinx iSim
- Implementation: Xilinx ISE

- These are specific choices, but….
  - Overall contents are similar to other environments
  - Not very difficult to transfer skills

VHDL introduction

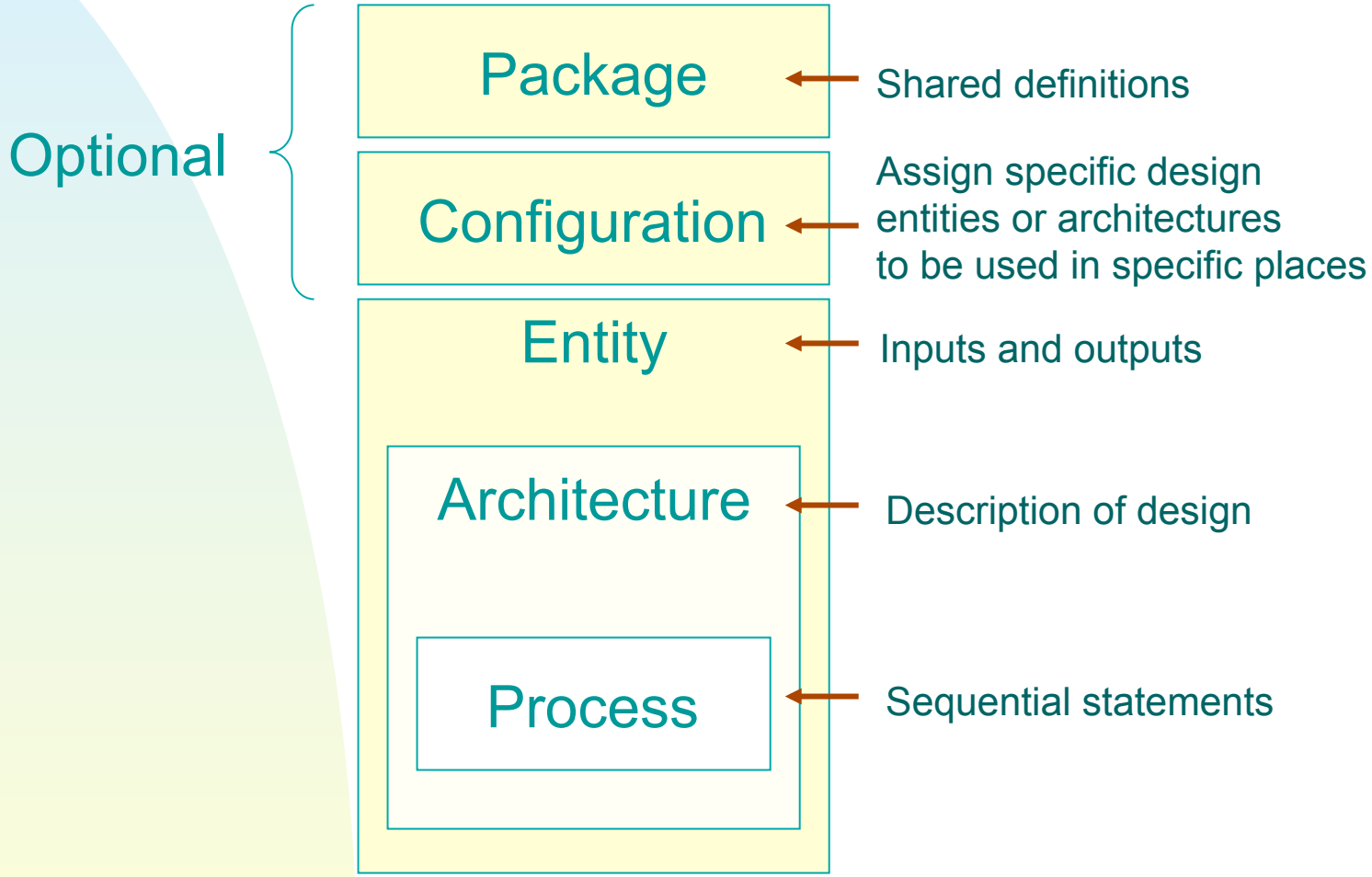
- **V**HSIC **H**ardware **D**escription **L**anguage
  - (VHSIC: Very high speed integrated circuit)
- Both concurrent and sequential operations
- International standard (1987, 1993, 2002)
  - Pure language definition
- Large standard, with multiple ways to code the same behavior

Design unit (library unit)



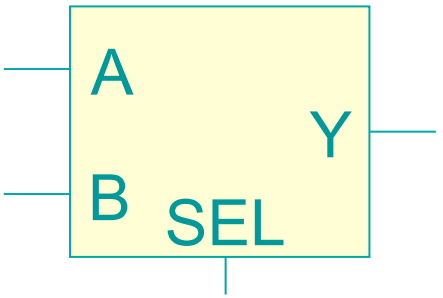
- In most basic form, defines
  - Inputs/outputs (entity)
  - Function of unit (architecture)
- Can be instantiated and connected together in higher level design units.

VHDL design unit



Multiplexer entity

```
library ieee;
use ieee.std_logic_1164.all;

entity Mux is
port(       a:    in std_logic;
            b:    in std_logic;
          sel:    in std_logic;
            y:    out std_logic
);
end Mux;
```

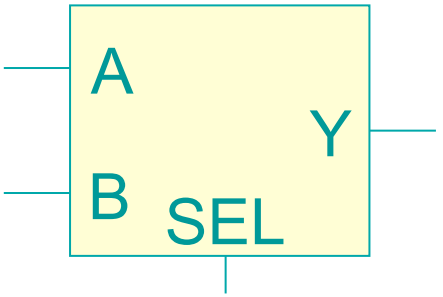


no semicolon after last port declaration!

Port name Direction Type

An architecture

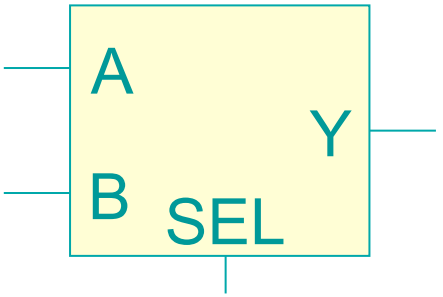
(many ways to do this)



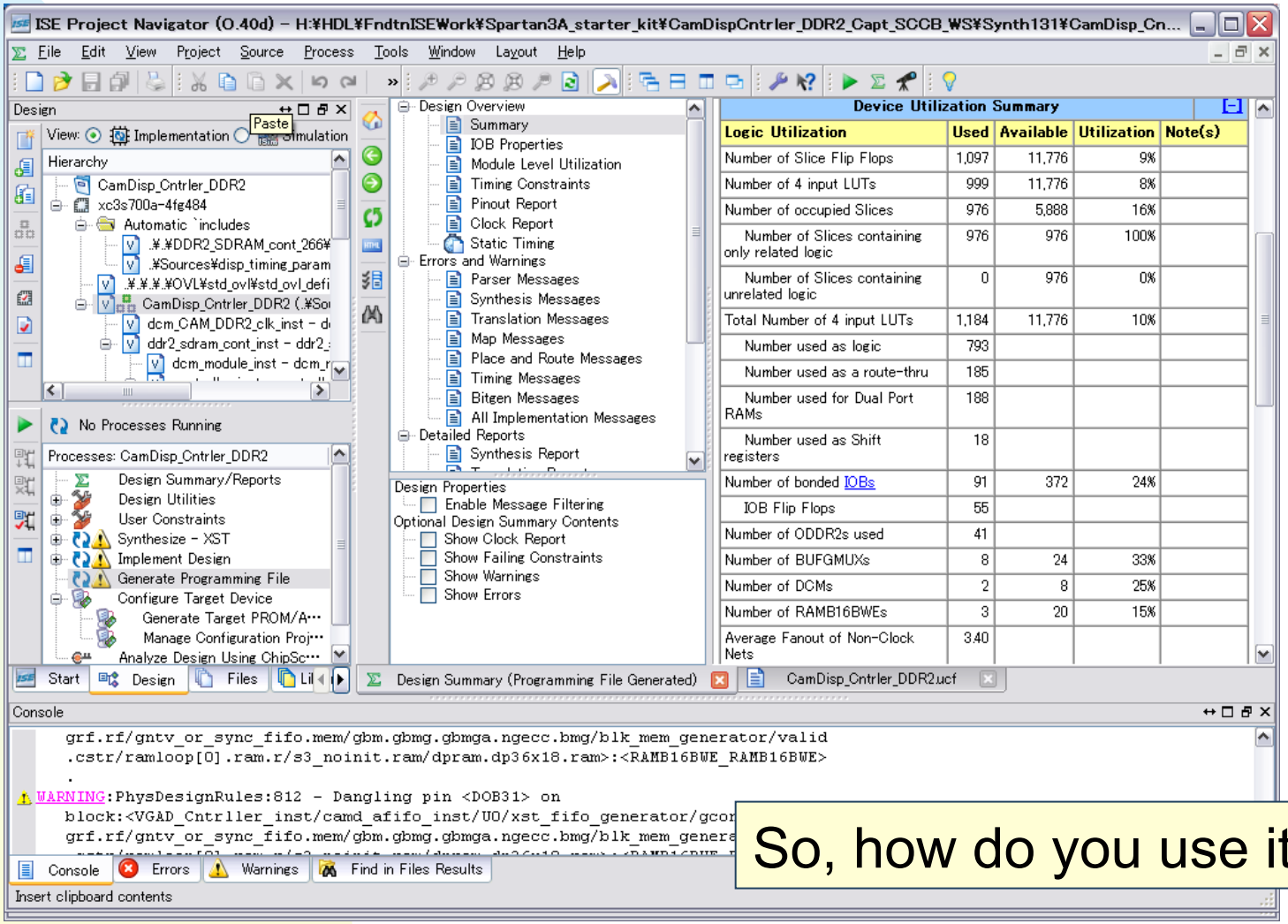

```
architecture arch3 of Mux is

begin
      y <= a when sel='0' else b;

end arch3;
```

Complete design unit:

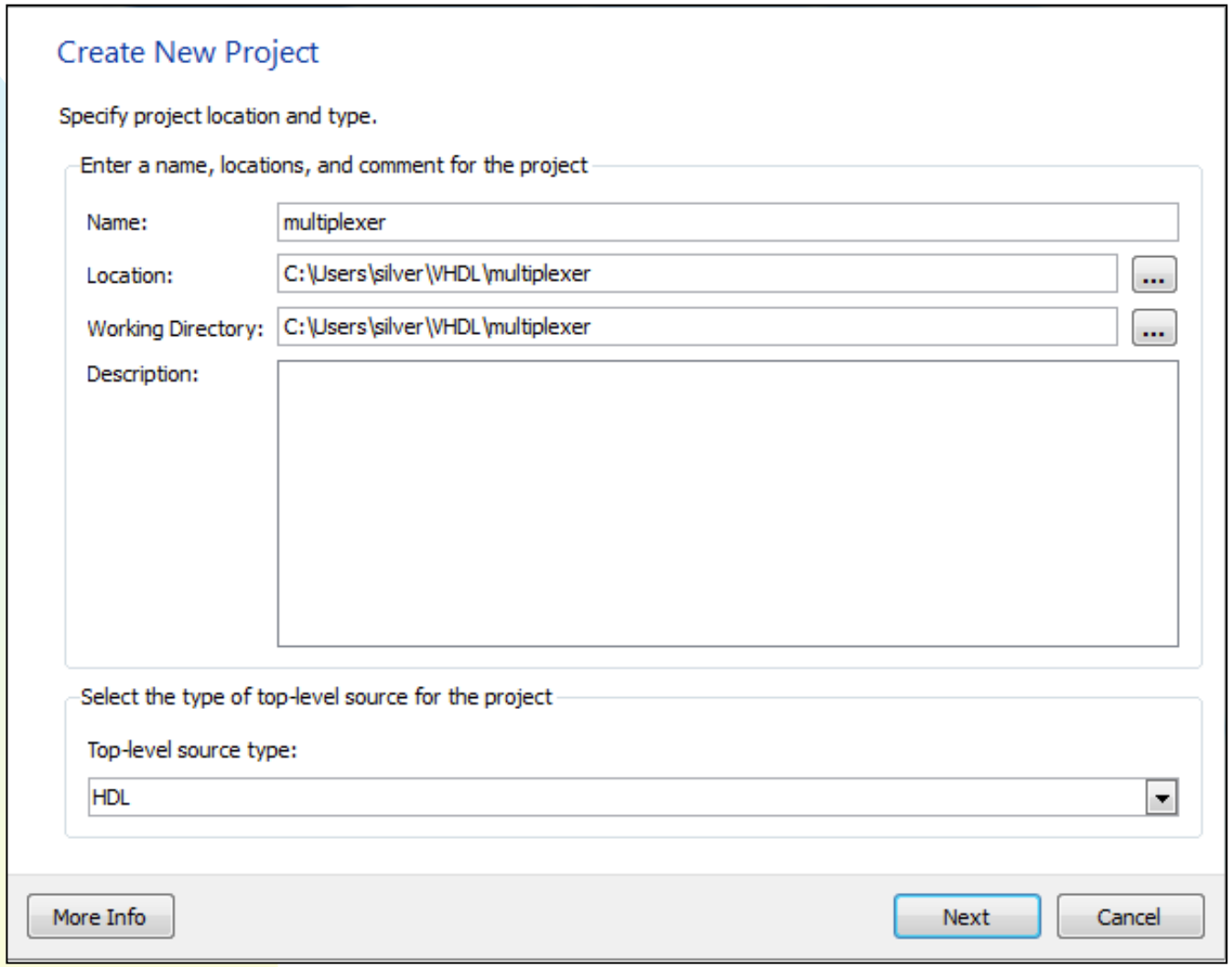
```
--
-- A 2 input multiplexer circuit
--
library ieee;
use ieee.std_logic_1164.all;

entity Mux is
port(   a:      in std_logic;
        b:      in std_logic;
        sel:    in std_logic;
        y:      out std_logic
);
end Mux;

architecture arch2 of Mux is
begin
        y <= a when sel='0' else b;

end arch2;
```



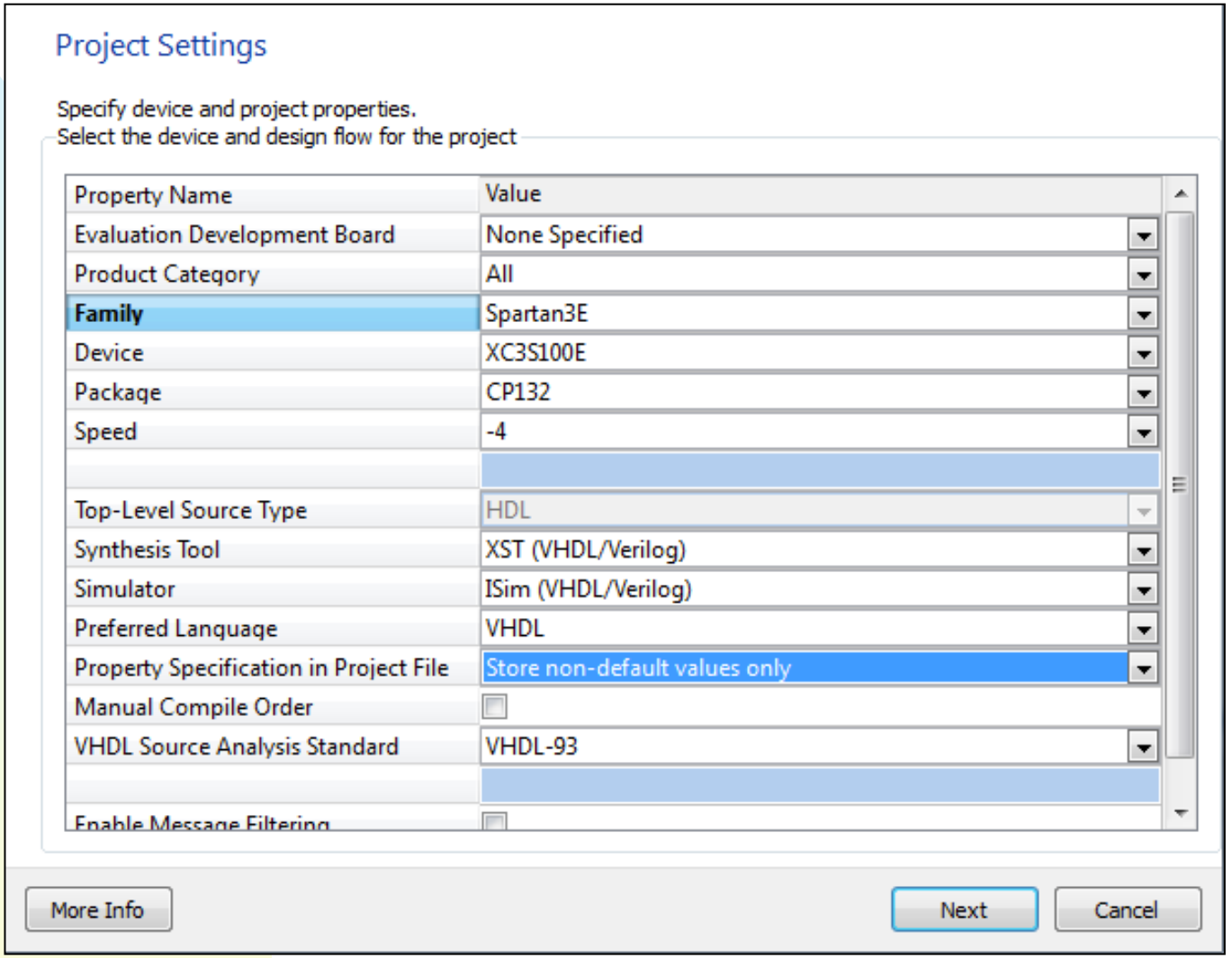
Xilinx design environment (ISE)



Create a new project

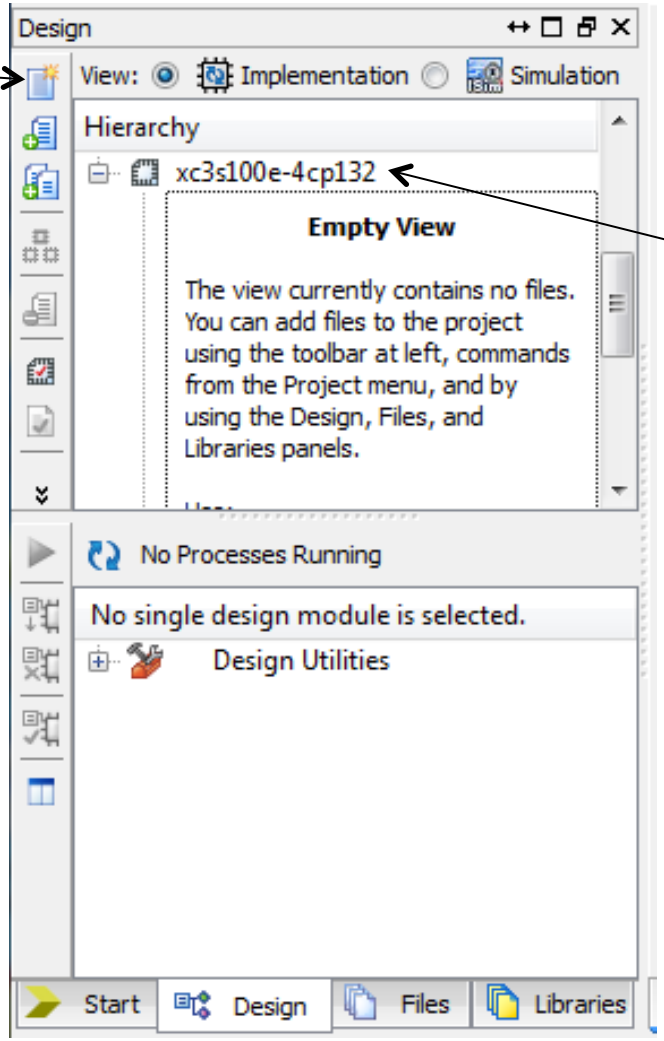


Select device, design flow



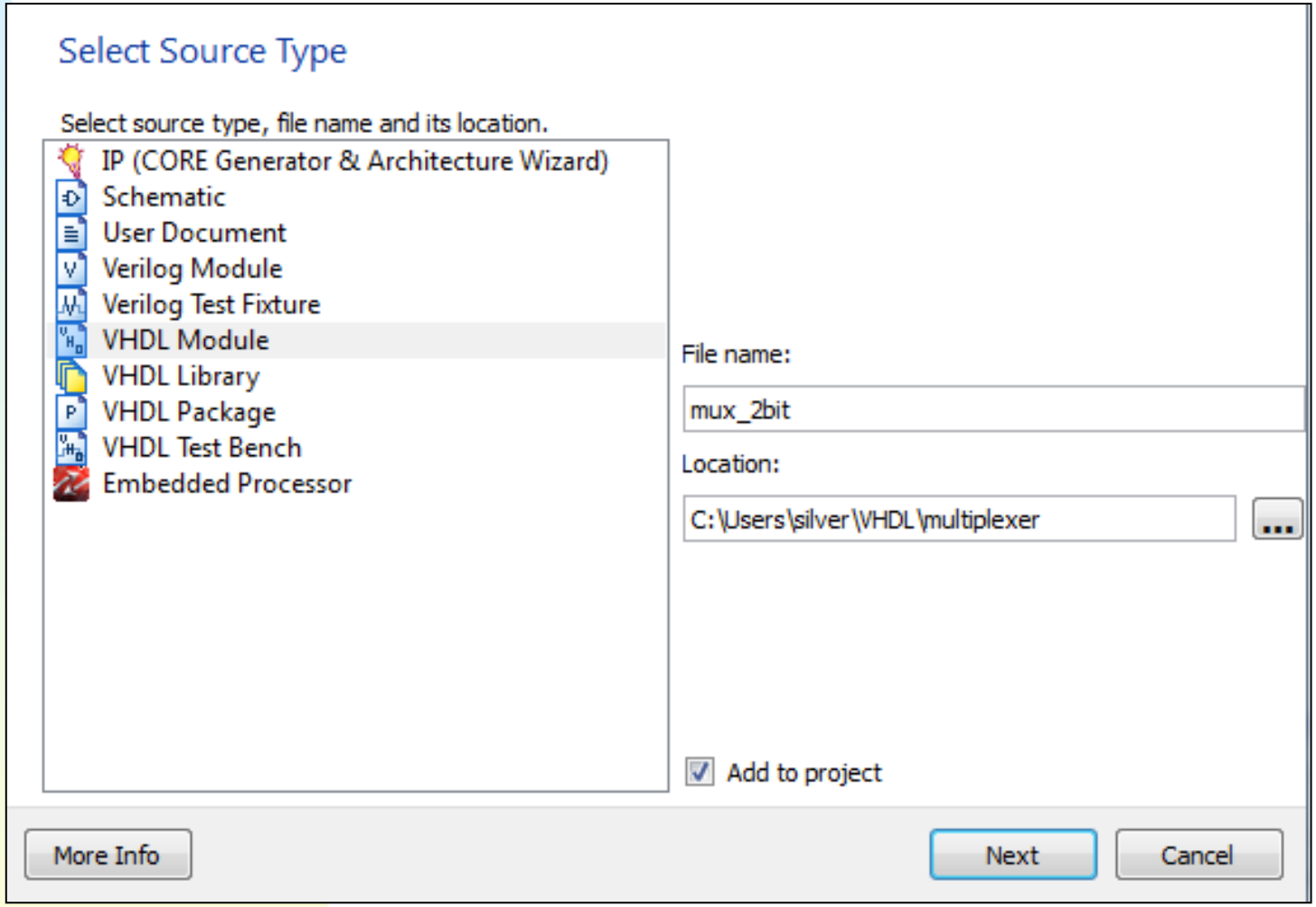
Create new source

Click here

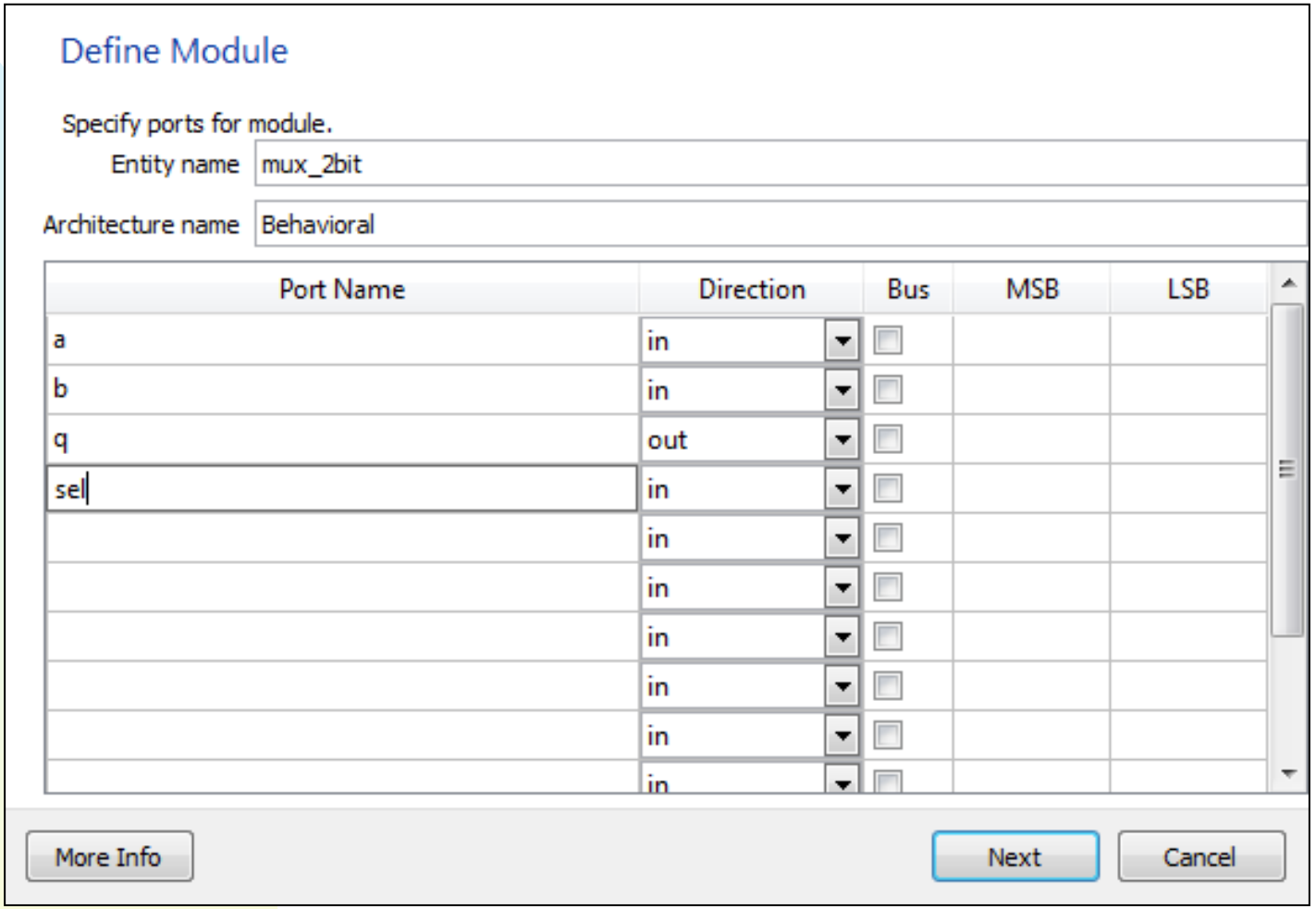


Or right-click here for menu

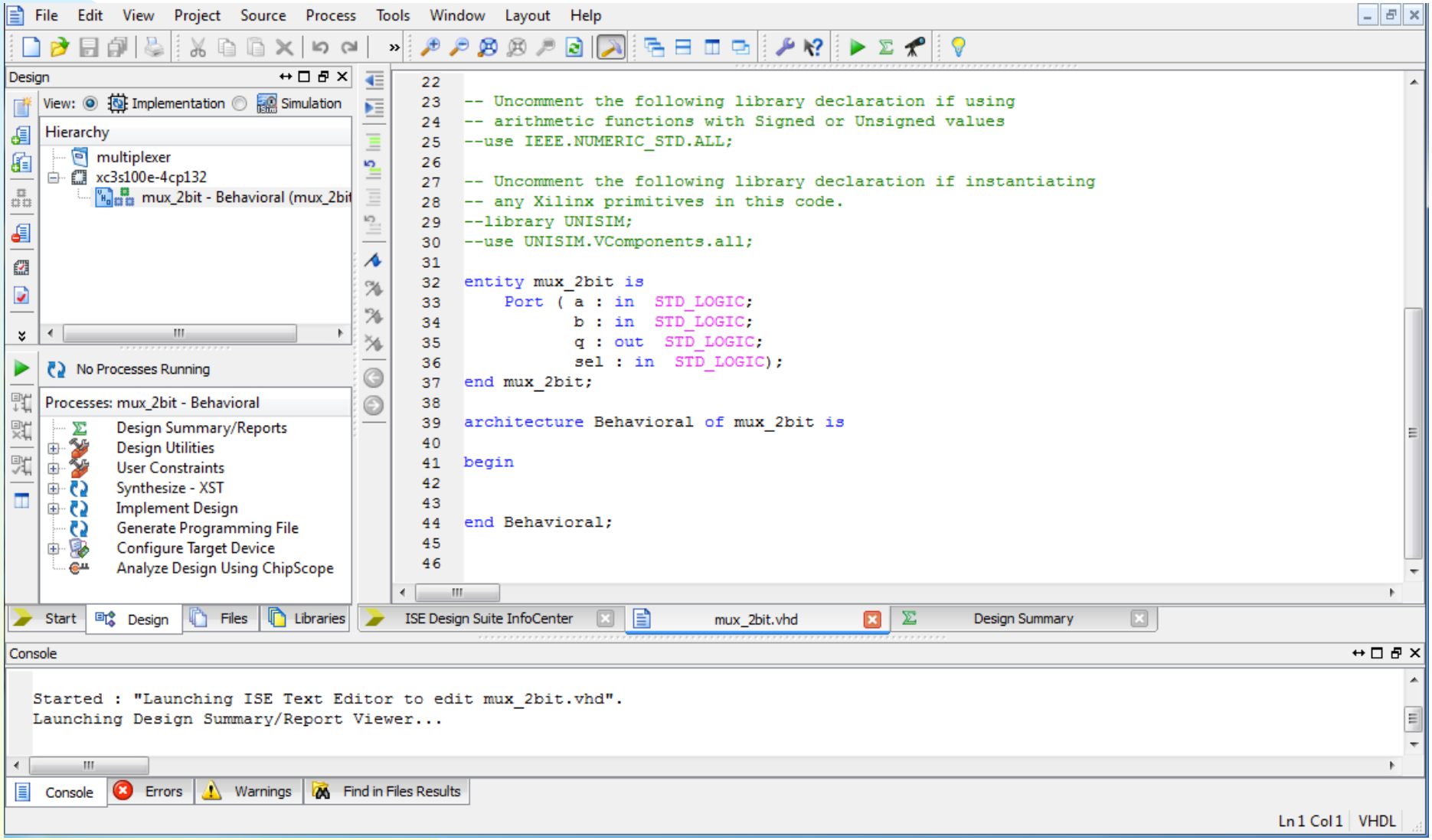
New source options



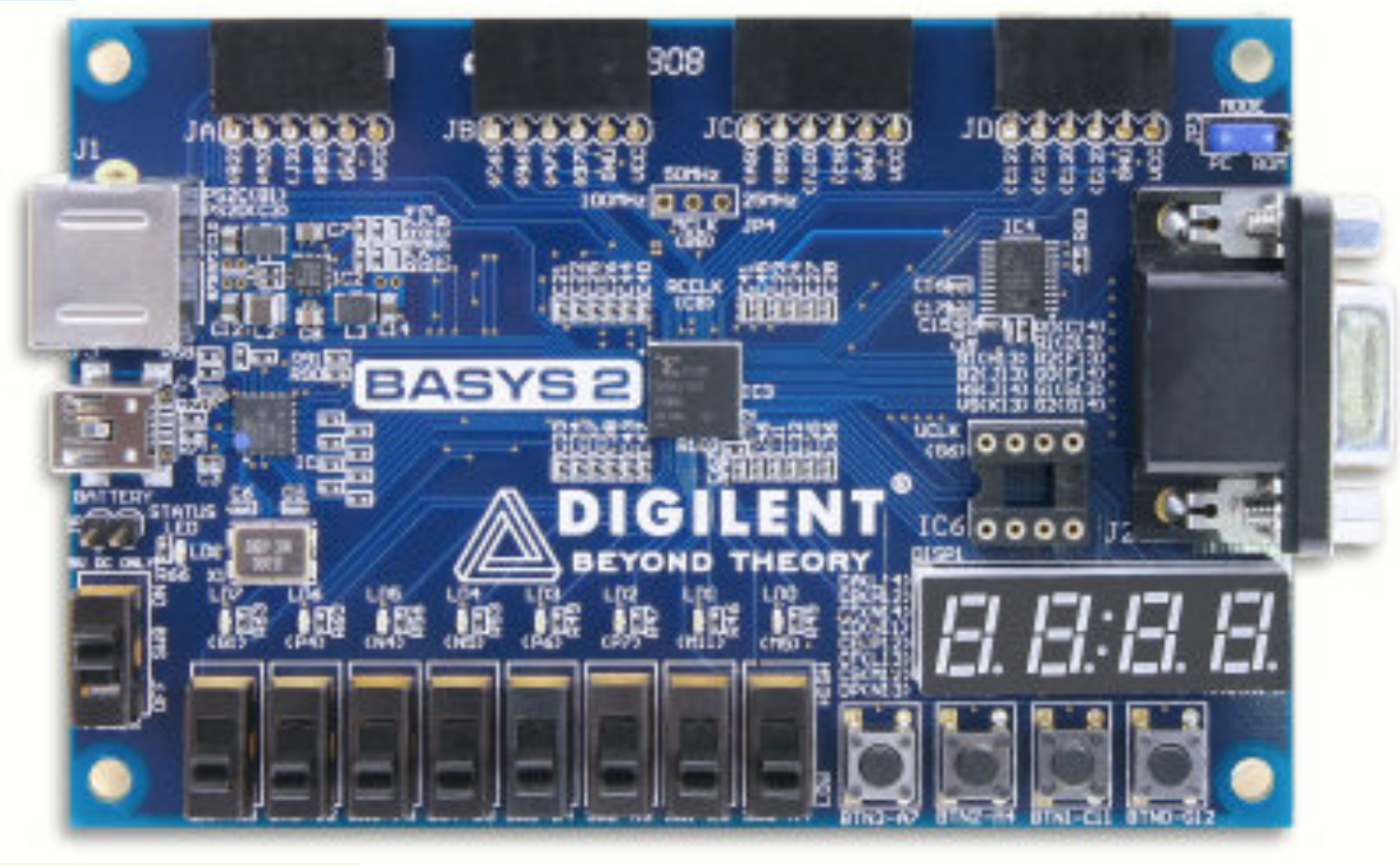
Initial definitions



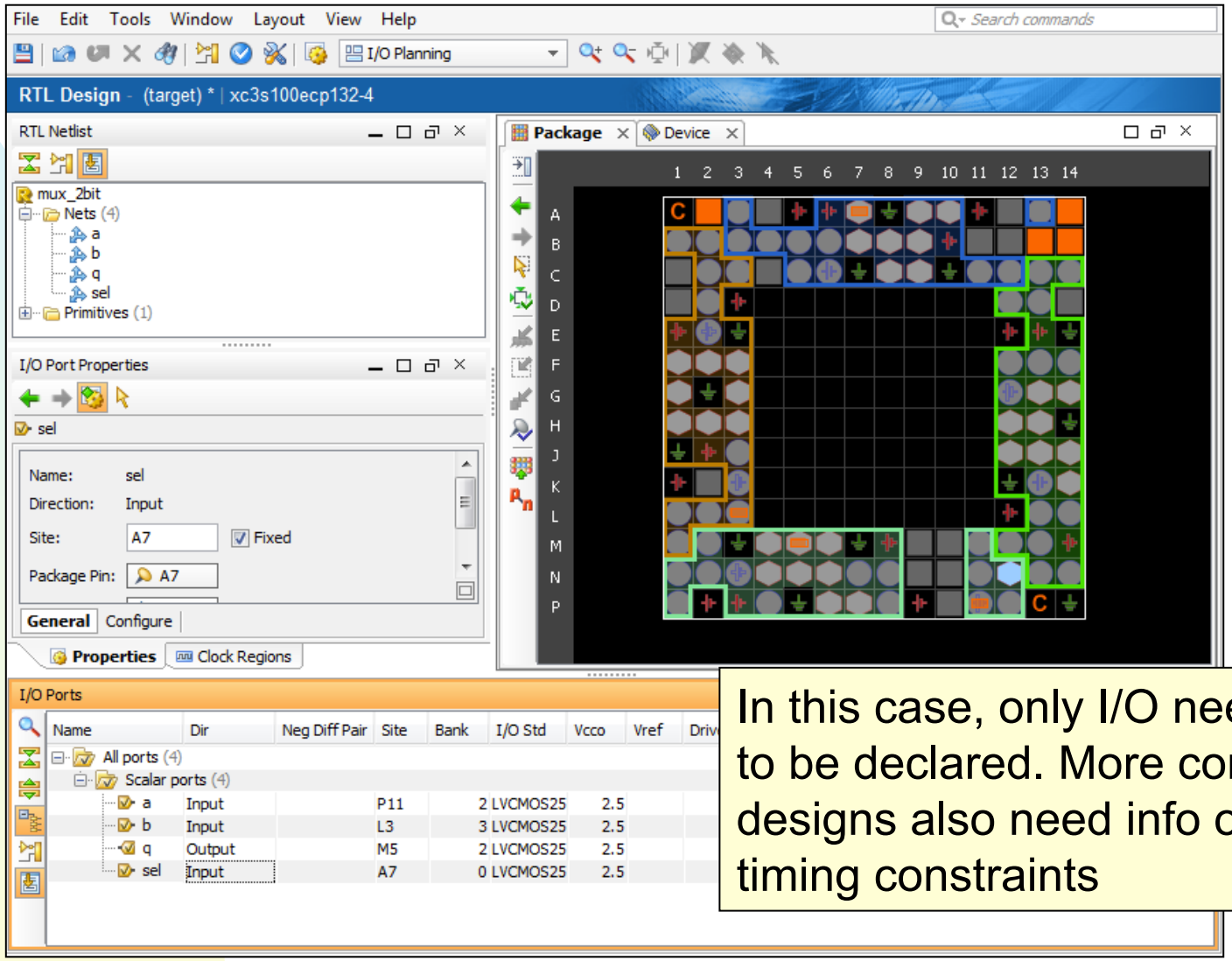
Edit sources



Target to your hardware



Define constraints

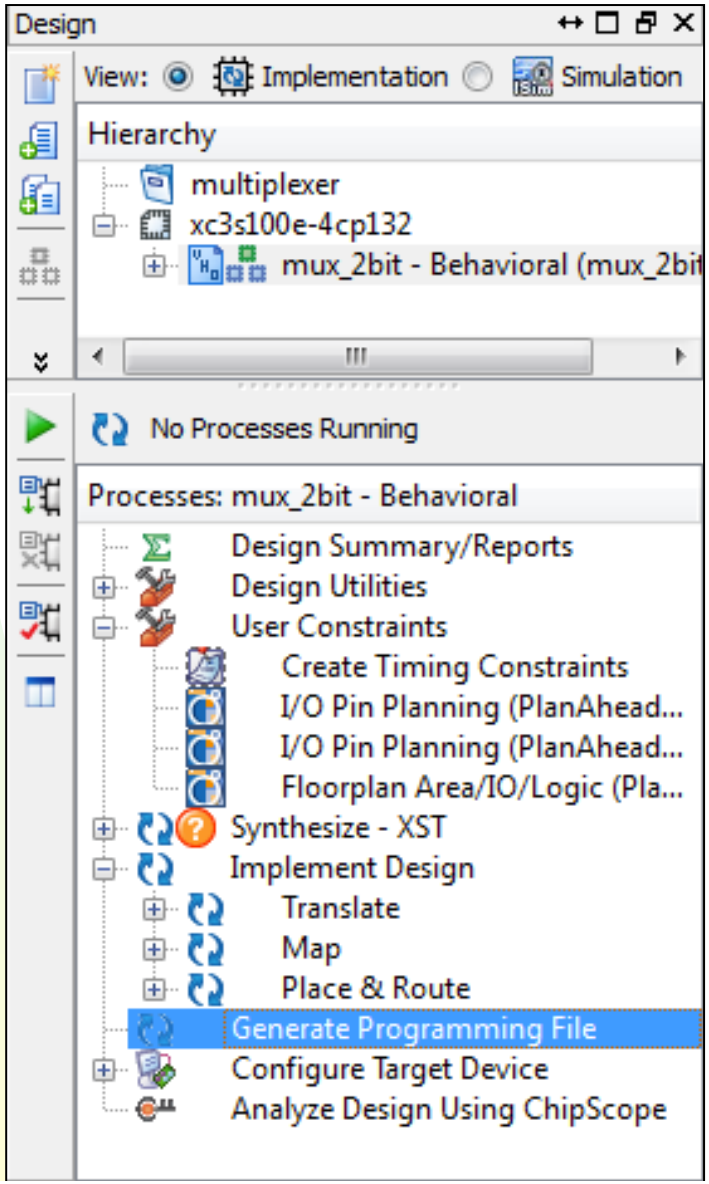


In this case, only I/O needs to be declared. More complex designs also need info on timing constraints

User Constraint File (UCF)

```
1
2  # PlanAhead Generated physical constraints
3
4  NET "a" LOC = P11;
5  NET "b" LOC = L3;
6  NET "q" LOC = M5;
7  NET "sel" LOC = A7;
8
```

Implement the design



ISE follows the design flow up to the selected point.

Configure the FPGA