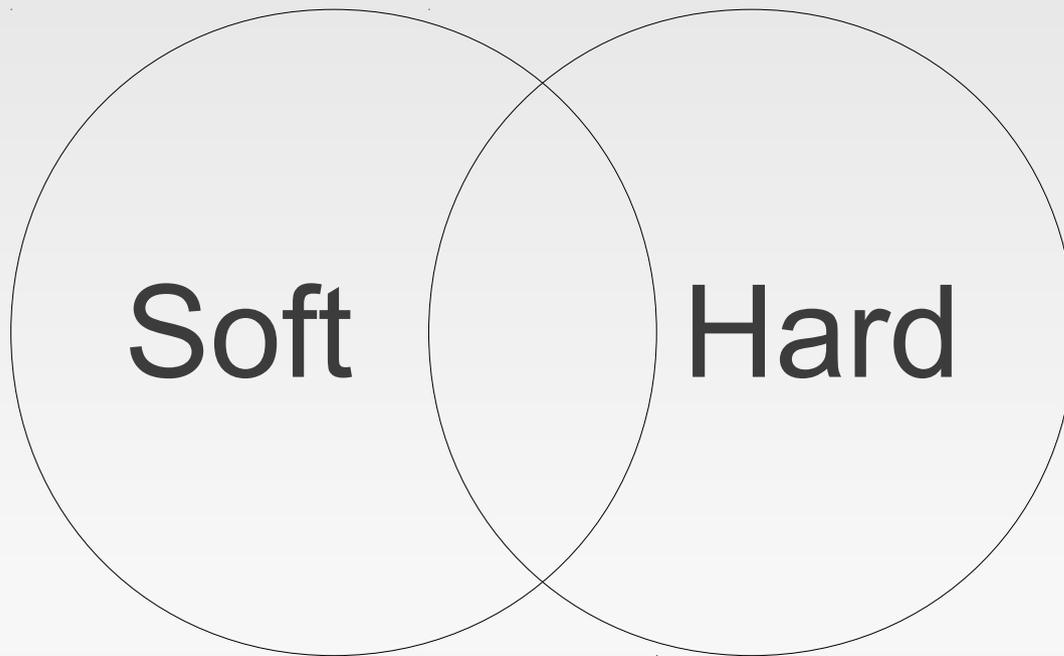


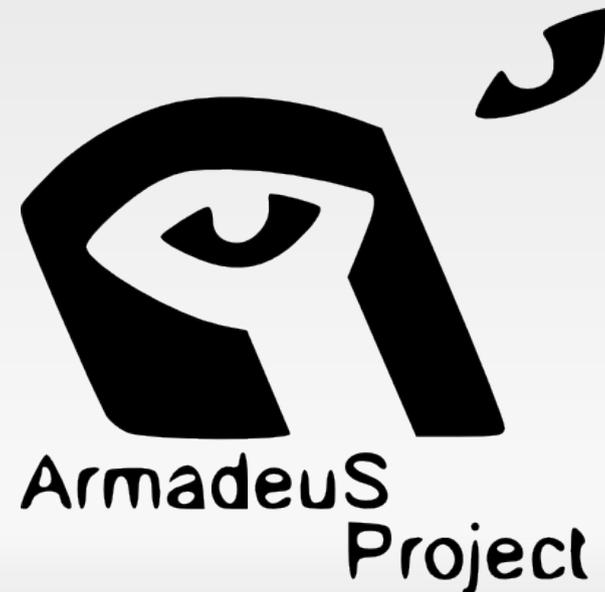
Quels outils libres pour le FPGA ?

Quand le monde du logiciel rencontre celui du matériel



Qui suis-je ?

- Membre d'ARMadeus Project
- Responsable FPGA chez ARMadeus Systems
- Mainteneur de POD (Peripherals On Demand)
- Libriste

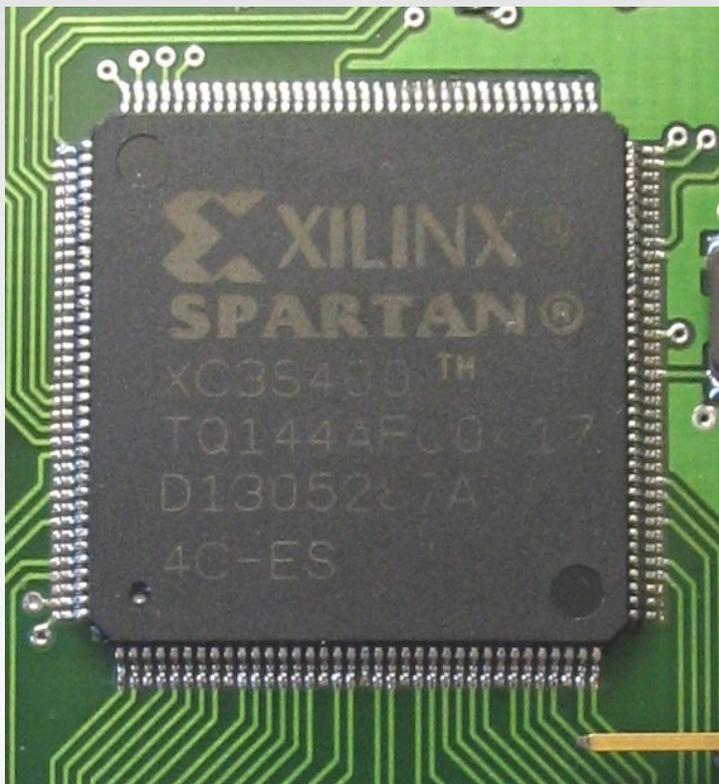


Sommaire

1. Qu'est-ce qu'un FPGA
2. La chaine de développement FPGA
3. Modélisation, simulation
 - VHDL
 - Verilog
 - SystemC
4. Spécialisation matériel
5. Génération du bitstream
6. Utilisation réelle

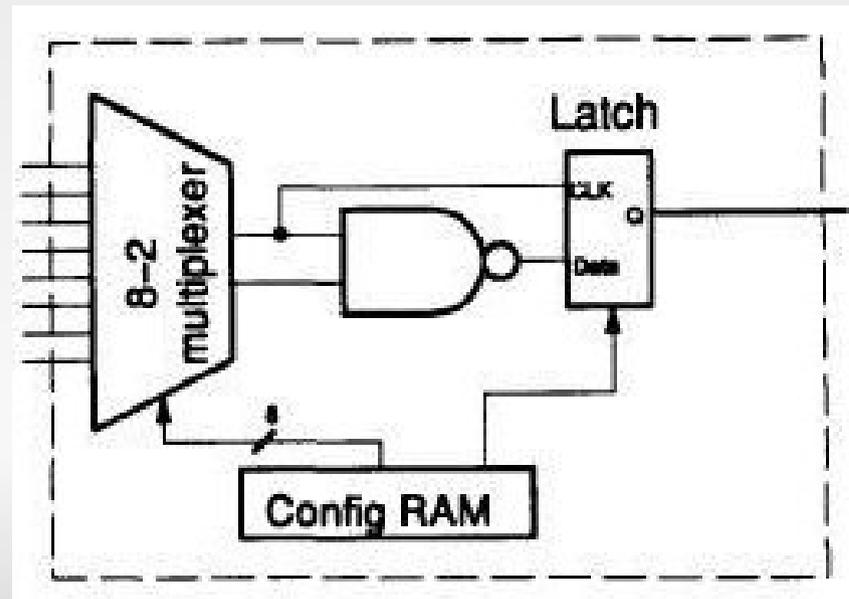
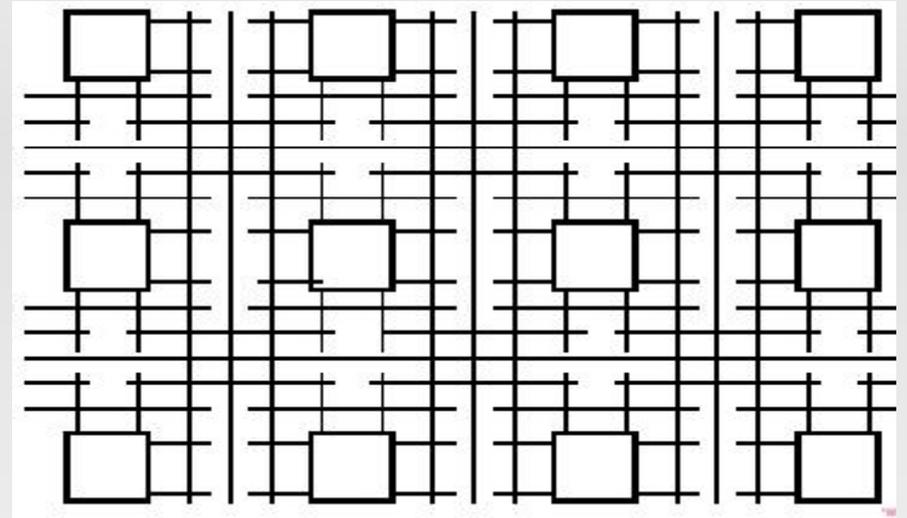
Qu'est-ce qu'un FPGA ?

- **F**ield **P**rogrammable **G**ates **A**rray
- Matrice de Champs de Portes Programmables



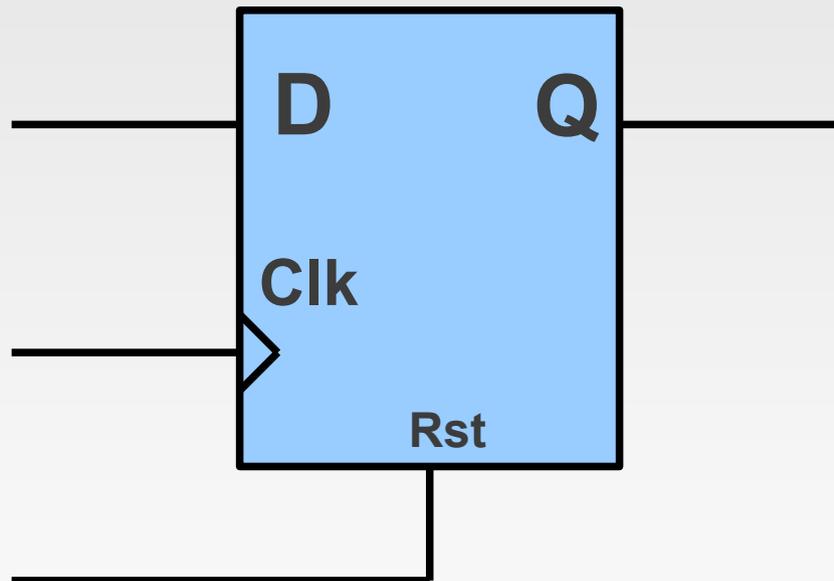
Qu'est-ce qu'un FPGA ?

- Matrice de portes logiques
- Interconnexions reconfigurables
- Différentes technologies ram, flash, ...



Une simple bascule D

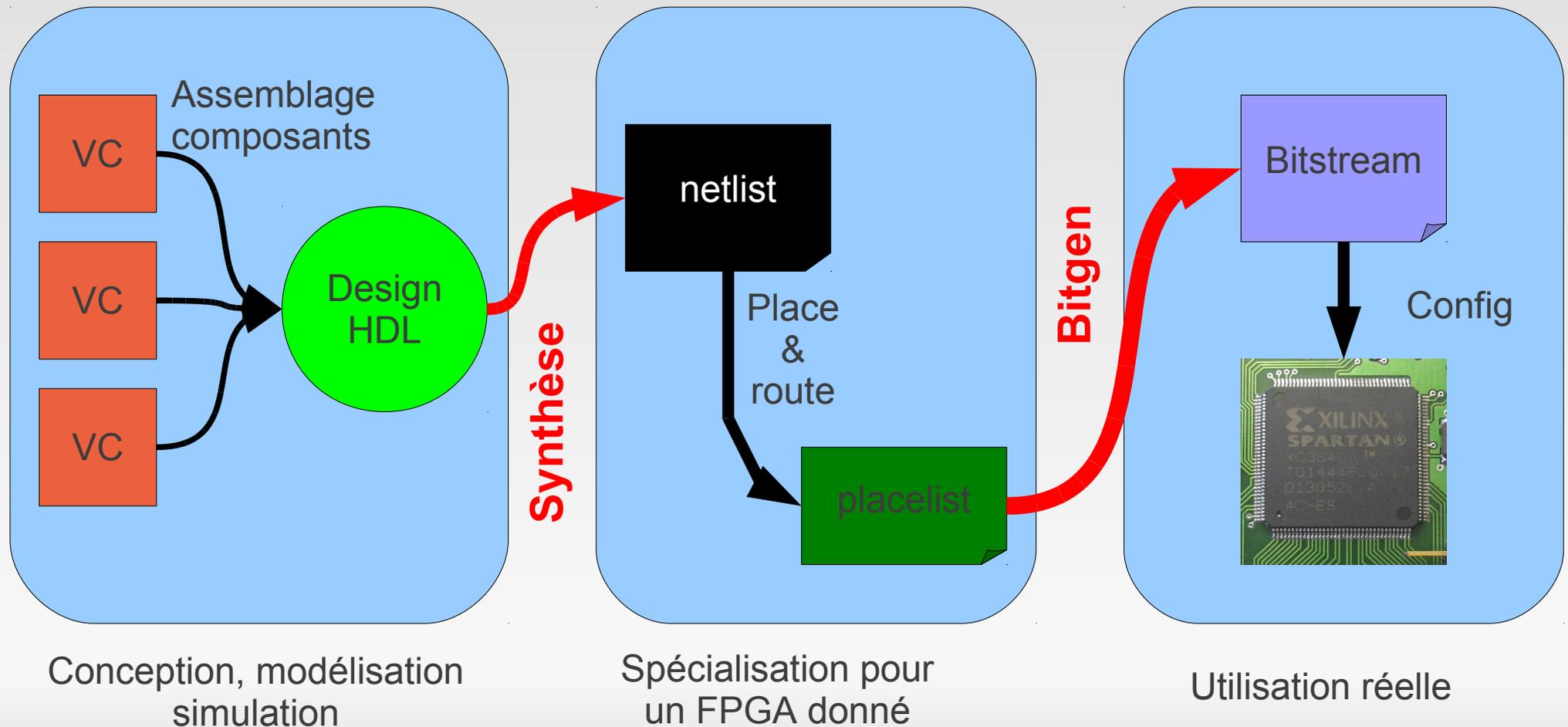
- La base d'un design synchrone.
- Recopie l'entrée D sur la sortie Q à chaque front de Clk.



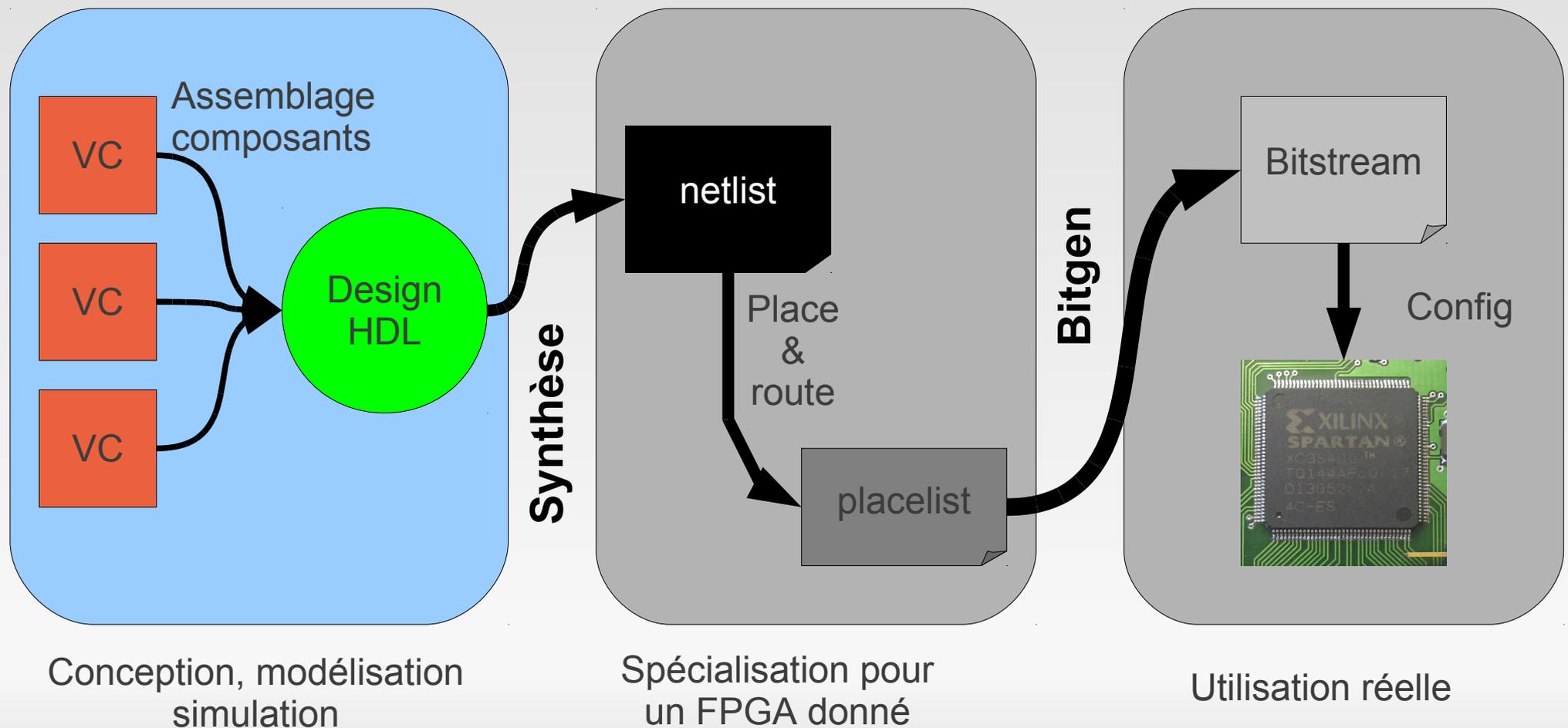
Composants Virtuels

- Aussi appelé IP (Intellectual property)
- Décrit dans un langage de modélisation
- «Synthétisable»
- Composants virtuels libres :
 - OpenCores
 - Milkymist
 - POD

Chaine de développement FPGA



Chaine de développement FPGA



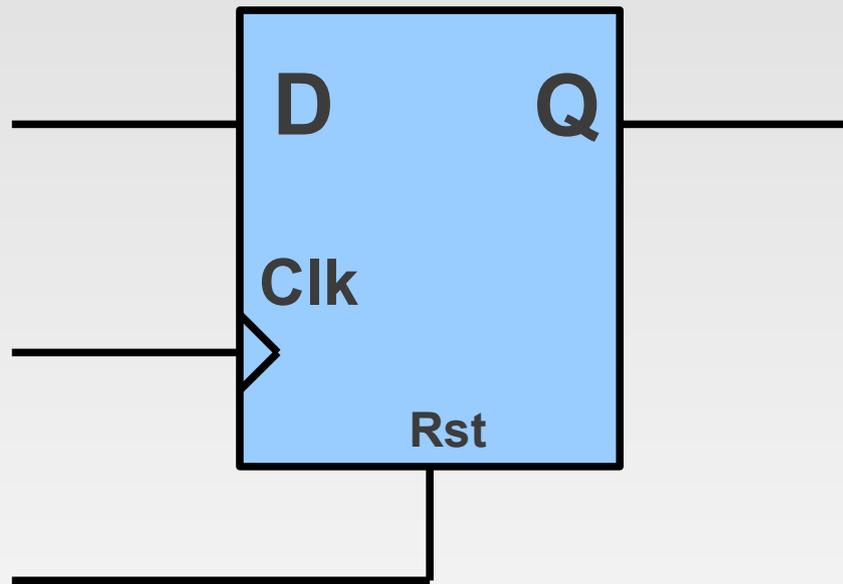
Modélisation et simulation

- Trois langages «standard»:
 - VHDL
 - Verilog
 - SystemC
- Tous simulables «librement»
- Tous les outils GNU/GPL disponibles

VHDL

- **VHSIC Hardware Description Language**
 - VHSIC : **V**ery **H**igh **S**peed **I**ntegrated **C**ircuit
- Language de description matériel pour circuit intégré à très grande vitesse...
- Commande de l'armée américaine
- Syntaxe «ada»
- Beaucoup utilisé en europe

Exemple simple



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

entity bascule_d is
    port(
        clk    : in std_logic;
        reset  : in std_logic;
        d      : in std_logic;
        q      : out std_logic
    );
end entity bascule_d;

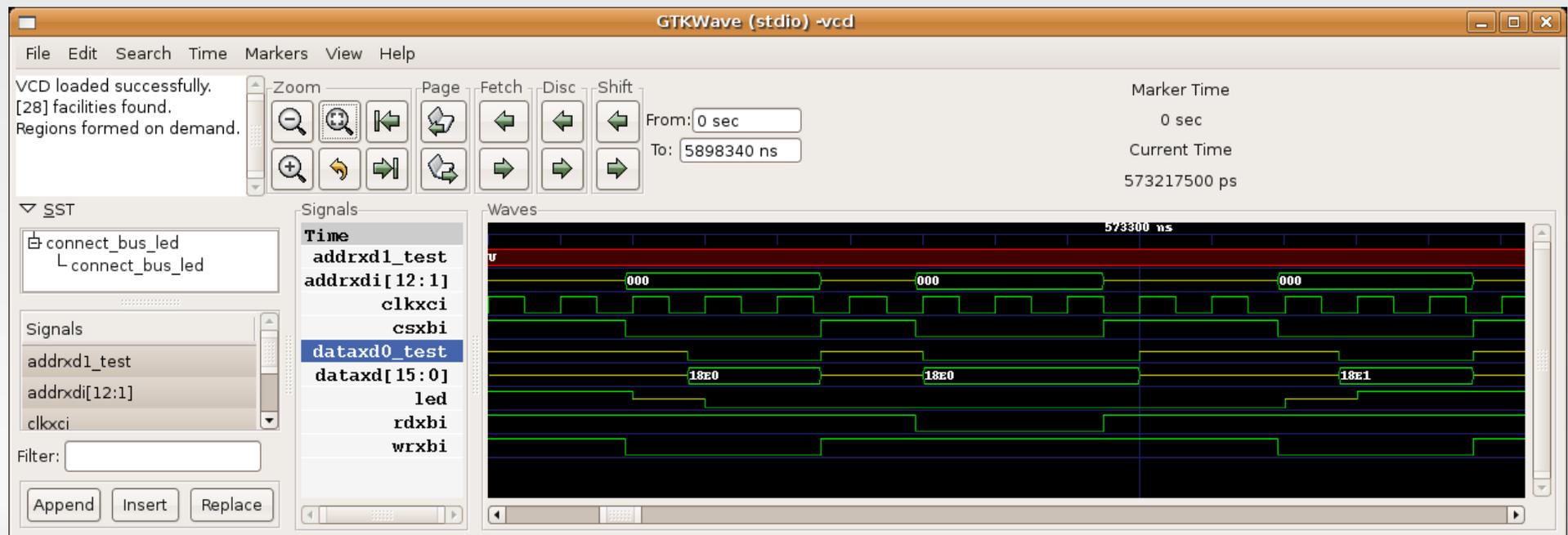
architecture comport of bascule_d is
begin
    bascule_p : process (clk, reset)
    begin
        if reset = '1' then
            q <= '0';
        elsif rising_edge(clk) then
            q <= d;
        end if;
    end process bascule_p;
end comport;
```

Outils VHDL

- **GHDL:**
 - Tristan Gingold (ada france)
 - Front-end de **gcc**
 - Couverture de code avec **gcov**
- **GTKWave:** visualiseur de signaux
 - Compatible tous langages de simulation
- **POD:** Peripherals On Demand
 - Outils Python d'assemblage de composants
 - ARMadeus Project

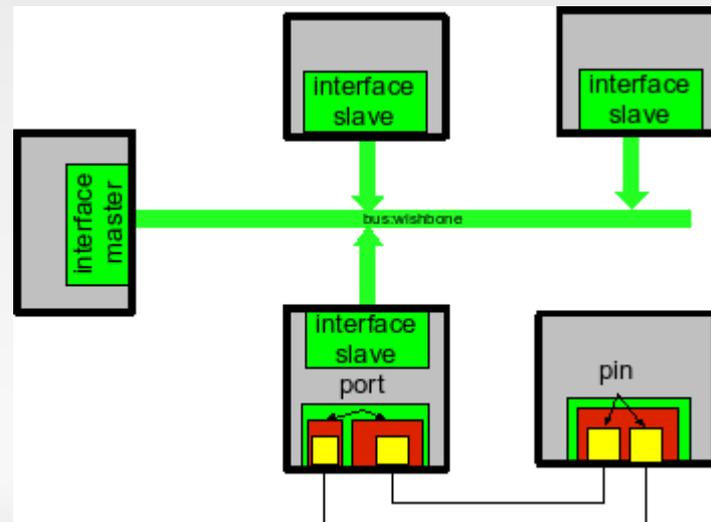
GTKWave

- Visualiseur de signaux en GTK
- Nombreux formats : vcd, ghw, vzt, lxt, ...
- Développement actif.



POD: Peripherals On Demand

- Outil Python d'assemblage de Composants virtuels
- Génère la «glue» en VHDL.
- Ligne de commande + complétion.



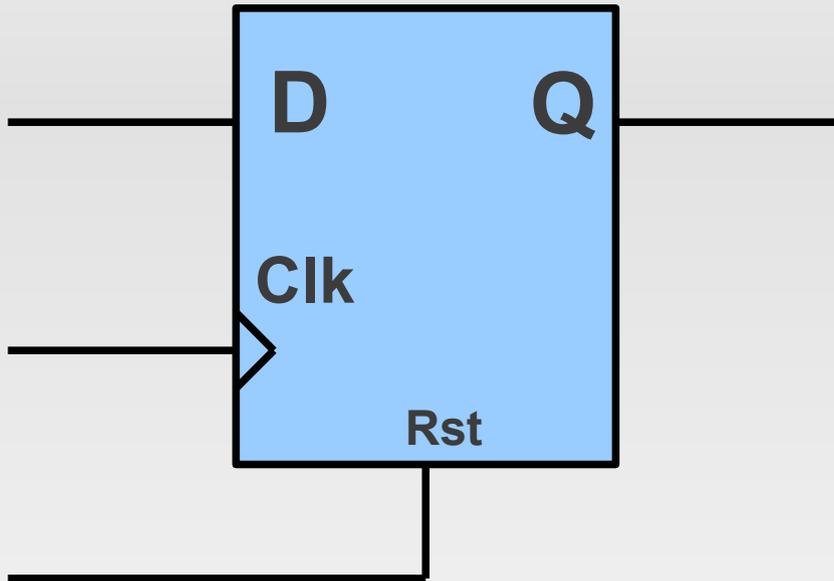
Demo



Verilog

- Développé par Cadence[©] depuis 1984
- Vaguement inspiré du C
- Standardisé par l'IEEE en 1995
- Beaucoup utilisé aux USA
- Plus concis

Exemple simple



```
module bascule_d(clk, reset, d, q);  
  input clk;  
  input d;  
  input reset;  
  output q;  
  
  reg q;  
  
  always @(posedge clk or posedge reset)  
    if (reset)  
      q <= 1'b0;  
    else  
      q <= d;  
  
endmodule
```

Outils Verilog

- **Icarus (iVerilog)**
 - Simulation, Synthèse
 - Conversion vers VHDL.
- **Verilator**
 - Convertisseur verilog → systemC, C++, ...
 - Accélérateur de simulation.
- **GTKWave**
 - Vermin, rtlbrowse
- **Dinotrace**

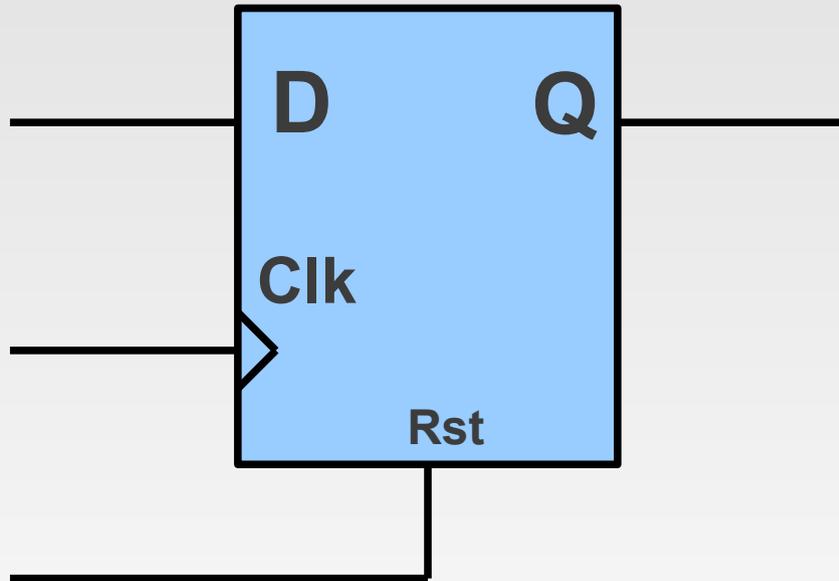
Demo



SystemC

- Initialement développé par Synopsys (scenic)
- Bibliothèque de classes et macros C++
- Normalisé par l'OSCI :
 - **Open SystemC Initiative**
- Code source disponible (login).
- Simulation comportementale (behavioral)
- Rapide.

Exemple simple



```
#include <systemc.h>

SC_MODULE(bascule_d)
{
    // Ports
    sc_in<bool> clk;
    sc_in<bool> reset;

    sc_in<bool> d;
    sc_out<bool> q;

    // Processus principal
    void bascule();

    // Constructeur
    bascule_d(sc_module_name name);

    SC_HAS_PROCESS(bascule_d);
};
```

```
#include "bascule_d.h"

bascule_d::bascule_d(sc_module_name name)
    :sc_module(name)
{
    SC_METHOD(bascule);

    sensitive << clk.pos() << reset;
}

void bascule_d::bascule()
{
    if (reset)
        q = false;
    else if (clk.event())
        q = d;
}
```

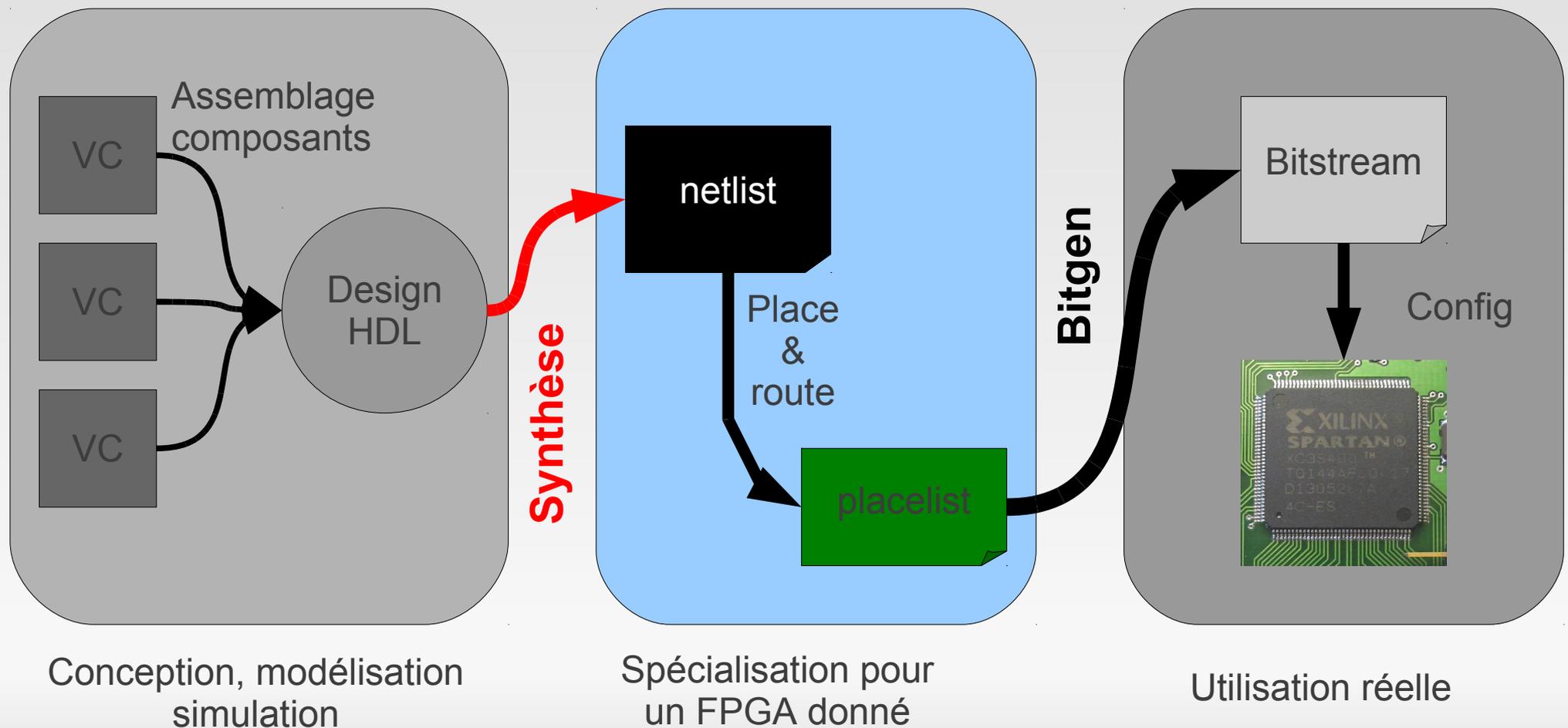
Les outils SystemC

- **systemc-2.2.0.tgz** : à télécharger sur le site de l'OSCI
- **GCC (g++)**
- **GtkWave**
- **Outils C++**

Demo



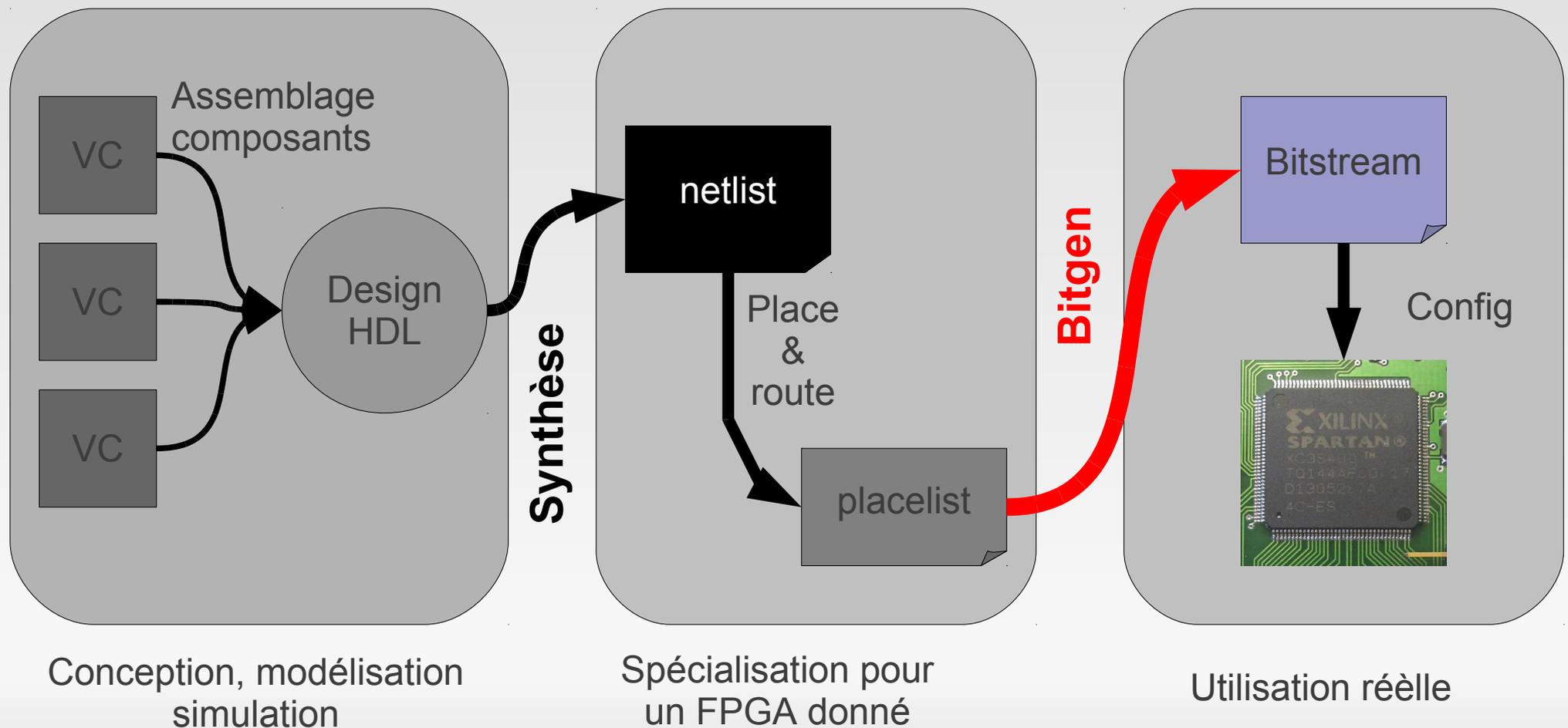
Chaine de développement FPGA



Spécialisation matériel

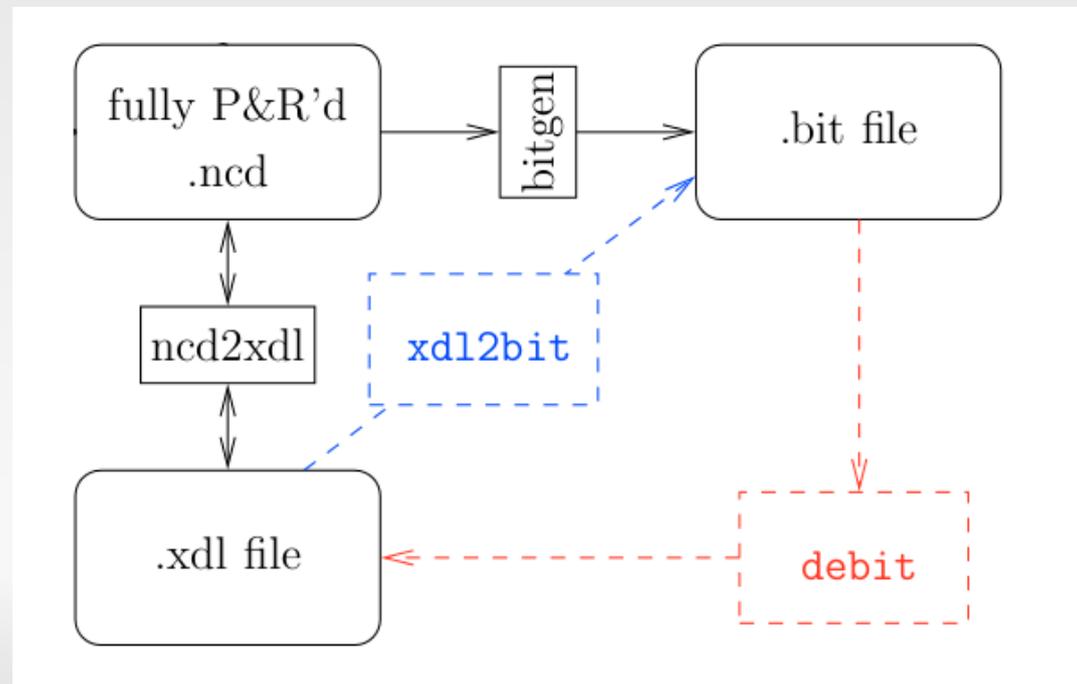
- De nombreux formats
 - Netlist:
 - NCD: Xilinx
 - EDIF: Standard
 - LLHDL: Projet libre (Sébastien Bourdeauducq)
 - Placelist:
 - XDL: Xilinx lisible
 - NCD: Xilinx
- Antares
 - Logiciel de placement routage pour Xilinx. (Sébastien Bourdeauducq)

Chaine de développement FPGA



Bitgen:Debit

- Projet Debit
 - Jean-Baptiste Note et Éric Rannaud (ENS)
 - www.ulogic.org
 - Reverse des bitstream Xilinx



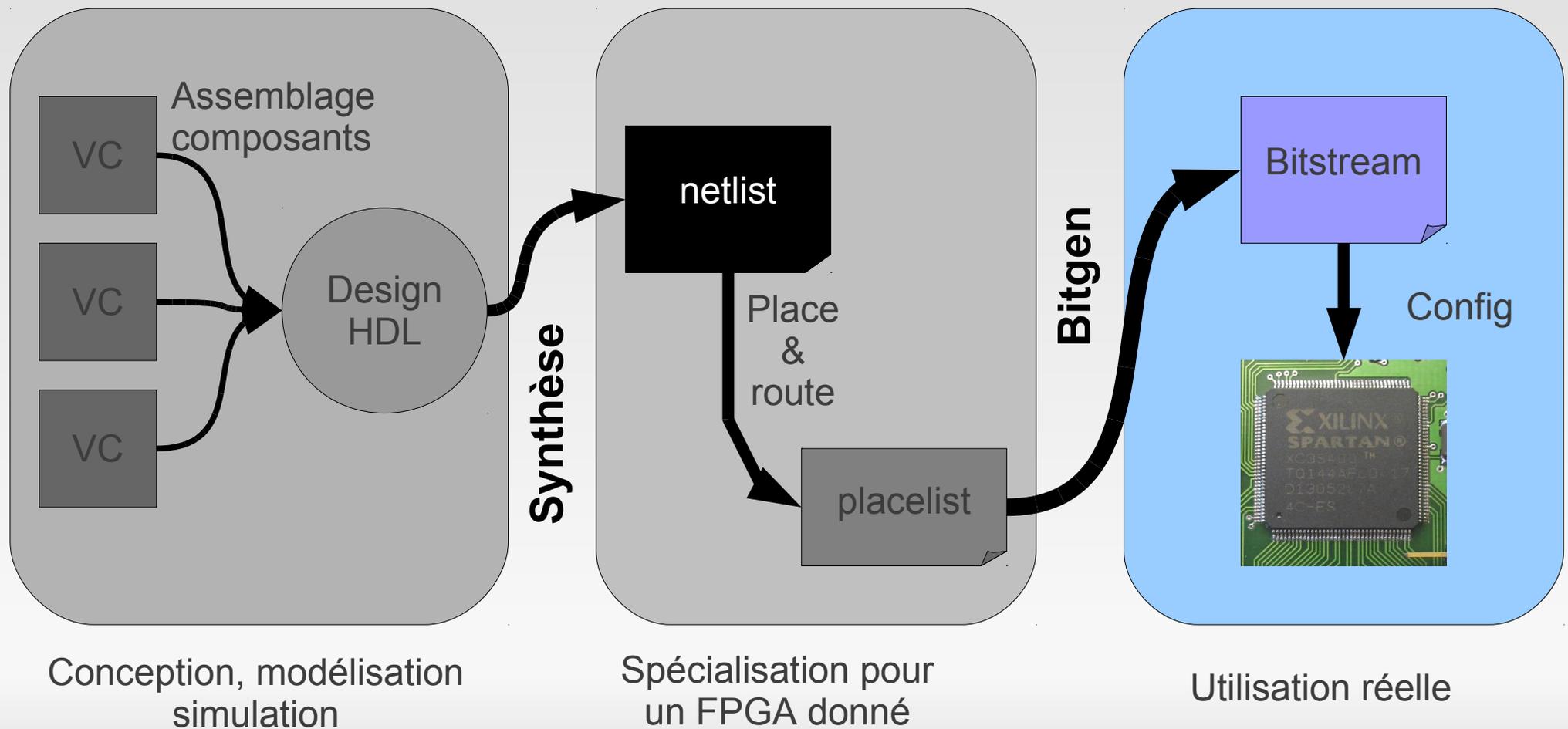
Bitgen:Debit

- Génère des bitstreams
 - Pour Virtex2
 - Pour Spartan3

Hardware config			5-DES, 201363 PIPs		10-DES, 431545 PIPs		20-DES, 883463 PIPs	
Processor	Clock-speed	RAM	debit	bitgen	debit	bitgen	debit	bitgen
Opteron	2.2GHz	2GB	0.65s	63.54s	0.91s	101.90s	1.47s	177.08s
P-IV	1.8GHz	1GB	1.14s	166.25s	1.56s	259.58s	2.29s	464.52s
PowerPC	500MHz	256MB	3.90s	N/A	6.60s	N/A	9.96s	N/A
ARM	220MHz	64MB	18.14s	N/A	23.86s	N/A	35.73s	N/A

Table 1: Speed comparison of debit and bitgen

Chaine de développement FPGA



Utilisation réelle

- Configuration du FPGA correctement documenté.
 - EEPROM
 - JTAG
 - SPI
 - Parallèle
- Oscilloscope
- Fonctionnement réel

Conclusion

- Simulation et modélisation:
 - Des projets tout à fait utilisables
- Synthèse, placement routage:
 - Des projets naissants mais prometteur
- Génération de bitstream:
 - C'est possible !
 - «from the bitstream to the netlist»

La pente est raide mais la route est droite !

Liens

- GHDL : <http://ghdl.free.fr>
- GtkWave : <http://gtkwave.sourceforge.net/>
- POD : <http://periphondemand.sourceforge.net/>
- Icarus : <http://iverilog.wikia.com/>
- Verilator, Dynotrace : <http://www.veripool.org/>
 - VHDL : github.com/sebx86/VerilatorVHDL
- SystemC : <http://www.systemc.org/>
- LLHDL : github.com/sbourdeauducq/llhdl
- Antares : github.com/sbourdeauducq/antares
- Debit : www.ulogic.org

Questions ?

Questions ?