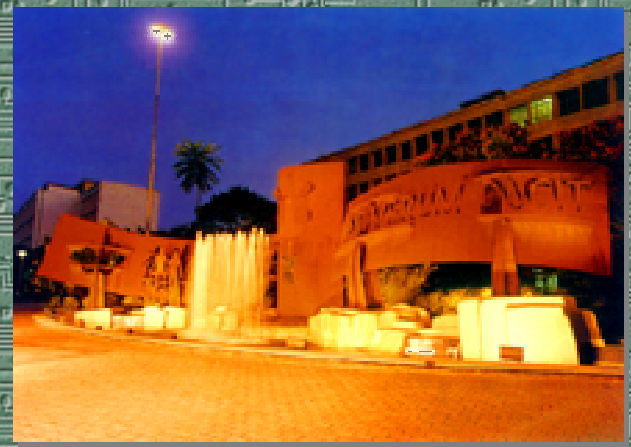
Remote and Partial Reconfiguration of FPGAs: Tools and Trends

Daniel Mesquita, Fernando Moraes, José palma, Leandro Moller, Ney Calazans



Laboratoire de Informatique, de Robotique et de Microéletronique de Montpellier (LIRMM – France)

Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS - Brazil)

Summary

- **Introduction**
- **State of art**
  - History and trends
  - Related work
- **Virtex internal organization**
  - Architecture overview
  - Addressing elements
- **Tools for partial and remote reconfiguration**
- **Conclusions**

mesquita@lirmm.fr

Summary

- **Introduction**
- **State of art**
  - History and trends
  - Related work
- **Virtex internal organization**
  - Architecture overview
  - Addressing elements
- **Tools for partial and remote reconfiguration**
- **Conclusions**

mesquita@lirmm.fr

Introduction

- Reconfigurable computing has been growing in the past two decades
- Sometimes there is confusion about the concept of dynamic reconfiguration
- We work with dynamic reconfigurable of-the-shelf devices, particularly Virtex family
- One more step towards hardware virtualization

mesquita@lirmm.fr

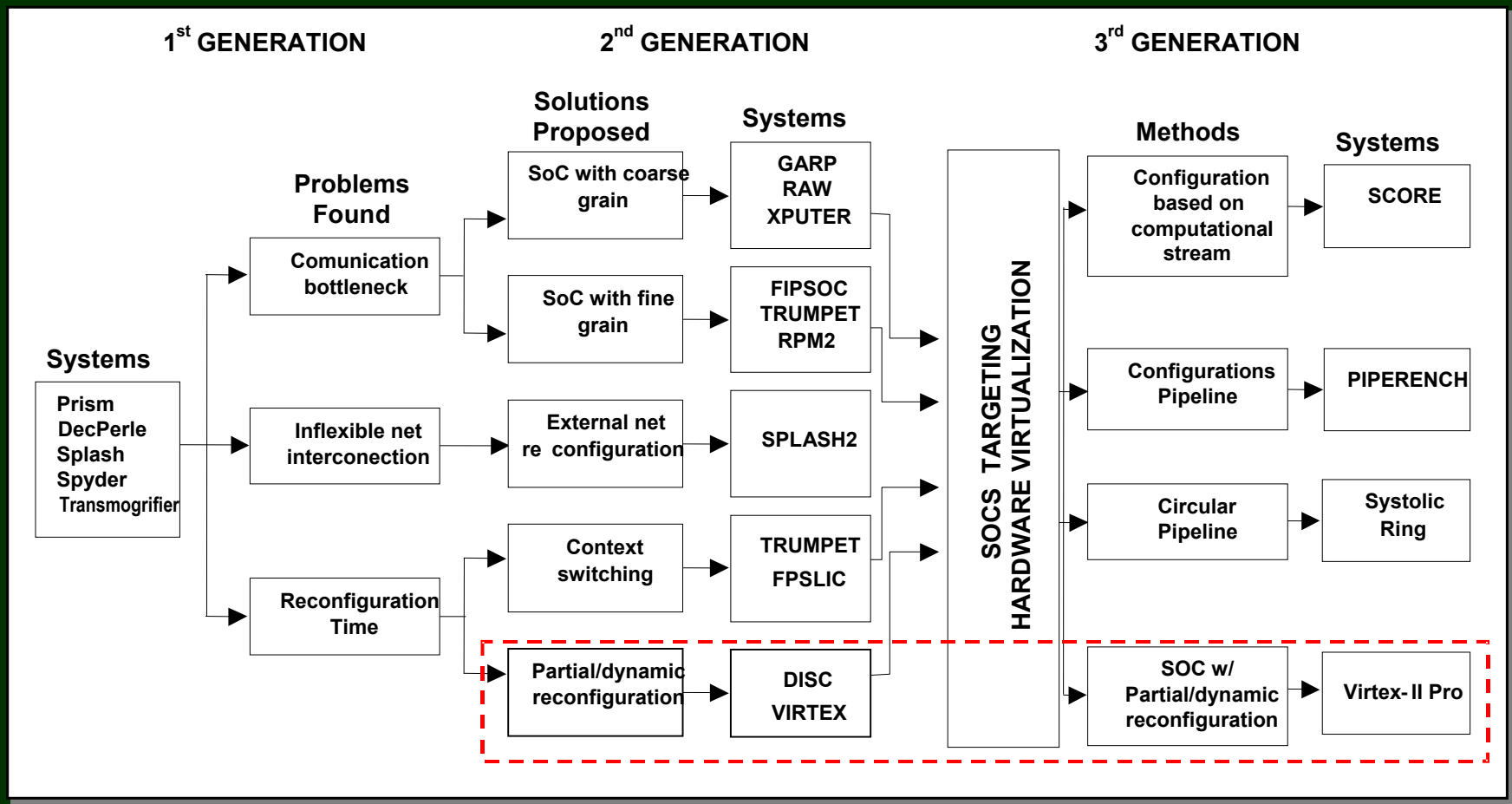
Summary

- **Introduction**
- **State of art**
  - History and trends
  - Related work
- **Virtex internal organization**
  - Architecture overview
  - Addressing elements
- **Tools for partial and remote reconfiguration**
- **Conclusions**

mesquita@lirmm.fr

State of art

- **Reconfigurable computing evolution**



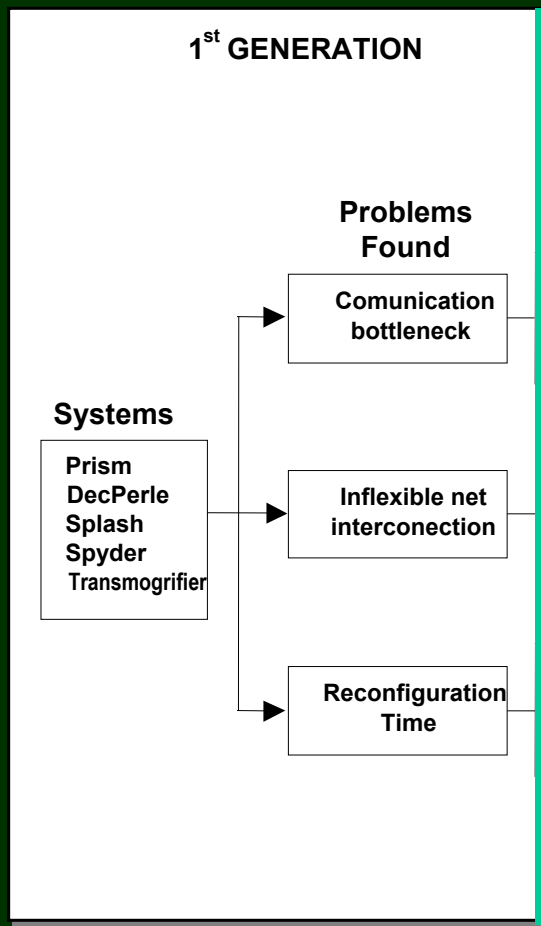
mesquita@lirmm.fr

State of art

- **1st Generation**
  - Goal: to increase performance of algorithms (e.g. Cryptography) over GPPs

State of art

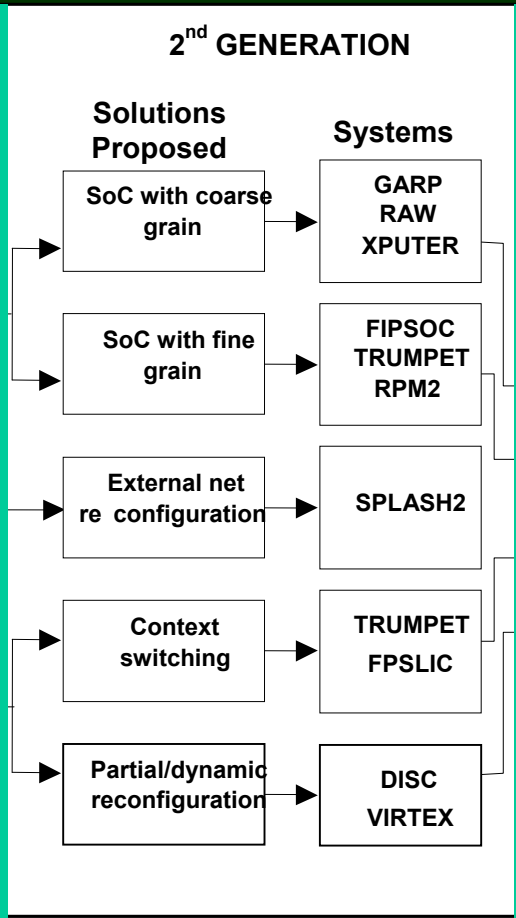
- **1st Generation**
  - Main problems:



mesquita@lirmm.fr

State of art

- **2st Generation**
  - Goal: to fix the problems from the first generation



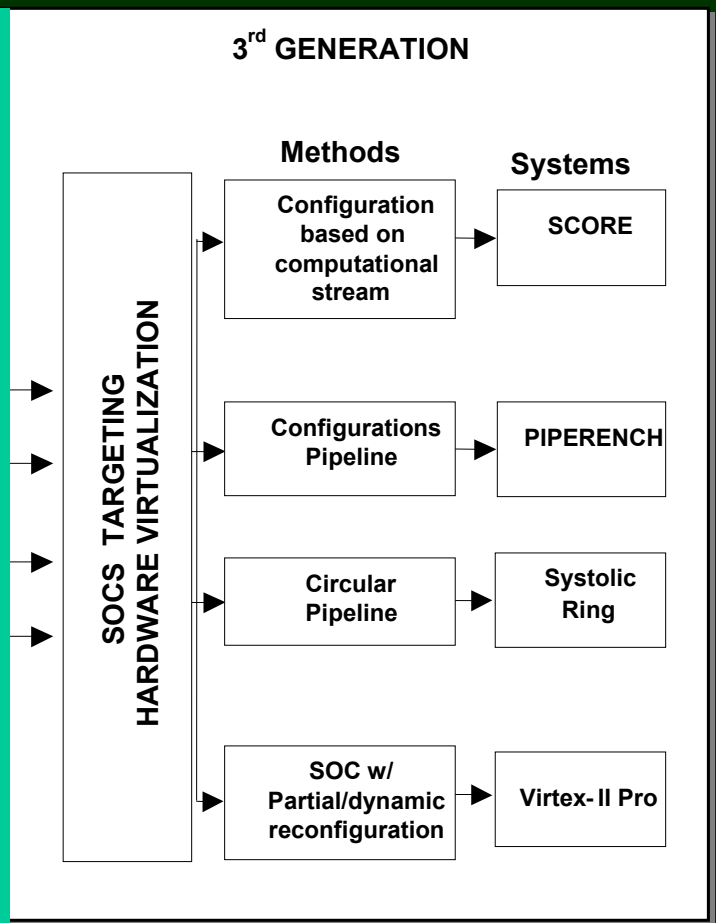
mesquita@lirmm.fr

State of art

- **3st Generation**
  - Goal: to make possible the hardware virtualization trhough dynamic reconfiguration

Trends:

SoCs, Coarse-grain architectures



mesquita@lirmm.fr

Summary

- **Introduction**
- **State of art**
  - History and trends
  - Related work
- **Virtex internal organization**
  - Architecture overview
  - Addressing elements
- **Tools for partial and remote reconfiguration**
- **Conclusions**

mesquita@lirmm.fr

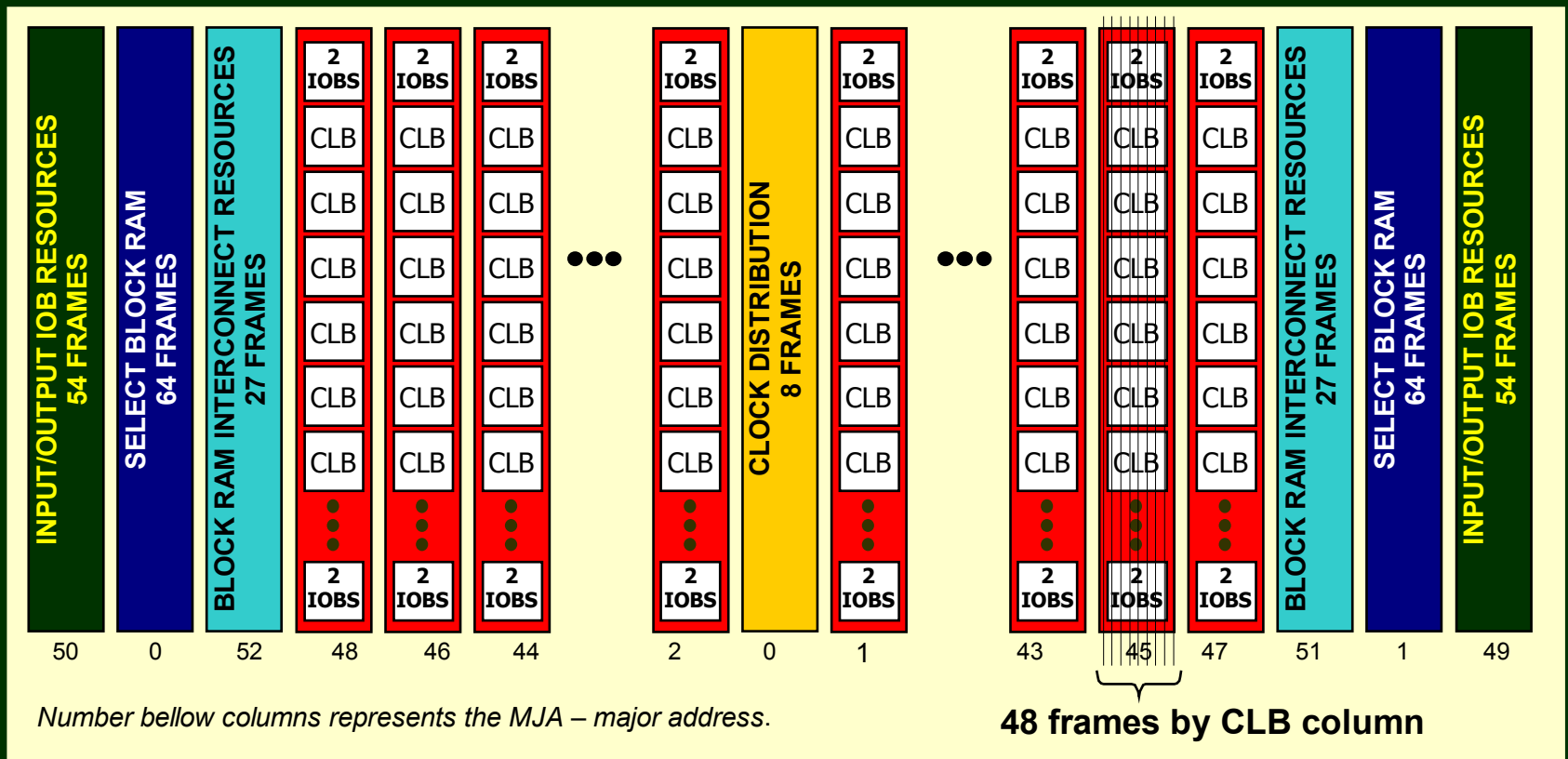
Virtex Organization

- **Architecture Overview**
  - Atomic reconfigurable unit: frame
  - Regular internal structure (composed by CLBs)
  - Allow relocation and defragmentation
  - It is partial or fully reconfigurable device

mesquita@lirmm.fr

Virtex Organization

- **Architecture Overview**
  - Abstraction of virtex internal organization



Virtex Organization

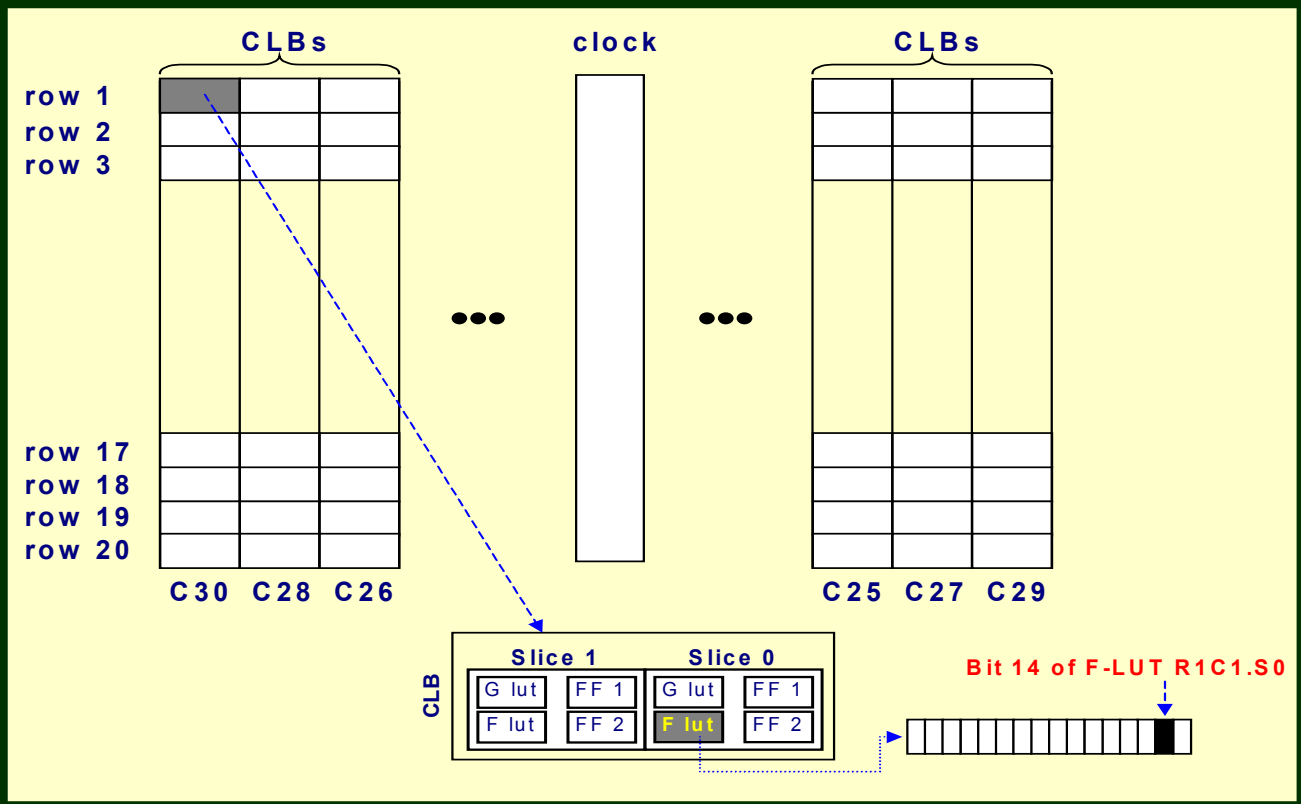
- **Addressing elements**
  - Equations

$\text{if}\left(\text{CLB}_{\text{col}} \le \frac{\textit{Chip}_{\textit{col}}}{2}\right)\ \text{then}\quad \text{MJA} = \textit{Chip}_{\textit{col}} - (\text{CLB}_{\text{col}} * 2) + 2\quad \text{else}\quad \text{MJA} = (\text{CLB}_{\text{col}} * 2) - \textit{Chip}_{\textit{col}} - 1$
$MNA = lut\_bit + wd - slice * (2*lut\_bit + 17)$
$fm\_bit\_idx = 3 + 18*CLB_{ROW} - FG + RW*32$
$fm\_st\_wd = FL * (8 + (MJA-1)*48 + MNA) + RW*FL$
$fm\_wd = abs(fm\_bit\_idx / 32)$
$fm\_wd\_bit\_idx = 31 + 32*fm\_wd - fm\_bit\_idx$

mesquita@lirmm.fr

Virtex Organization

- **Addressing elements**
  - Abstraction



Summary

- **Introduction**
- **State of art**
  - History and trends
  - Related work
- **Virtex internal organization**
  - Architecture overview
  - Addressing elements
- **Tools for partial and remote reconfiguration**
- **Conclusions**

mesquita@lirmm.fr

Tools for partial and remote reconfiguration

- **Circuit customization tool**
  - Function:
    - Generate a graphical interface to customize circuit parameters
    - Parameters stored in LUTs. The circuit can be reconfigured local or remotely
  - Benefits:
    - FPGA architecture is hidden from designer
    - Eliminates the need of external devices and/or the associated control logic to set parameters at run time
  - Remarks:
    - There are three « players » related with this process:
      - Software developer
      - Circuit designer
      - Circuit user

mesquita@lirmm.fr

Tools for partial and remote reconfiguration

- **Circuit customization tool**
  - **Software developer**
    - implements the software layer, using JBITS classes, hiding the FPGA architecture details *(applet)*
    - this applet is the same for all circuits being customized
  - **Circuit designer**
    - uses HTML tags to pass commands and parameters to the applet to customize his circuit

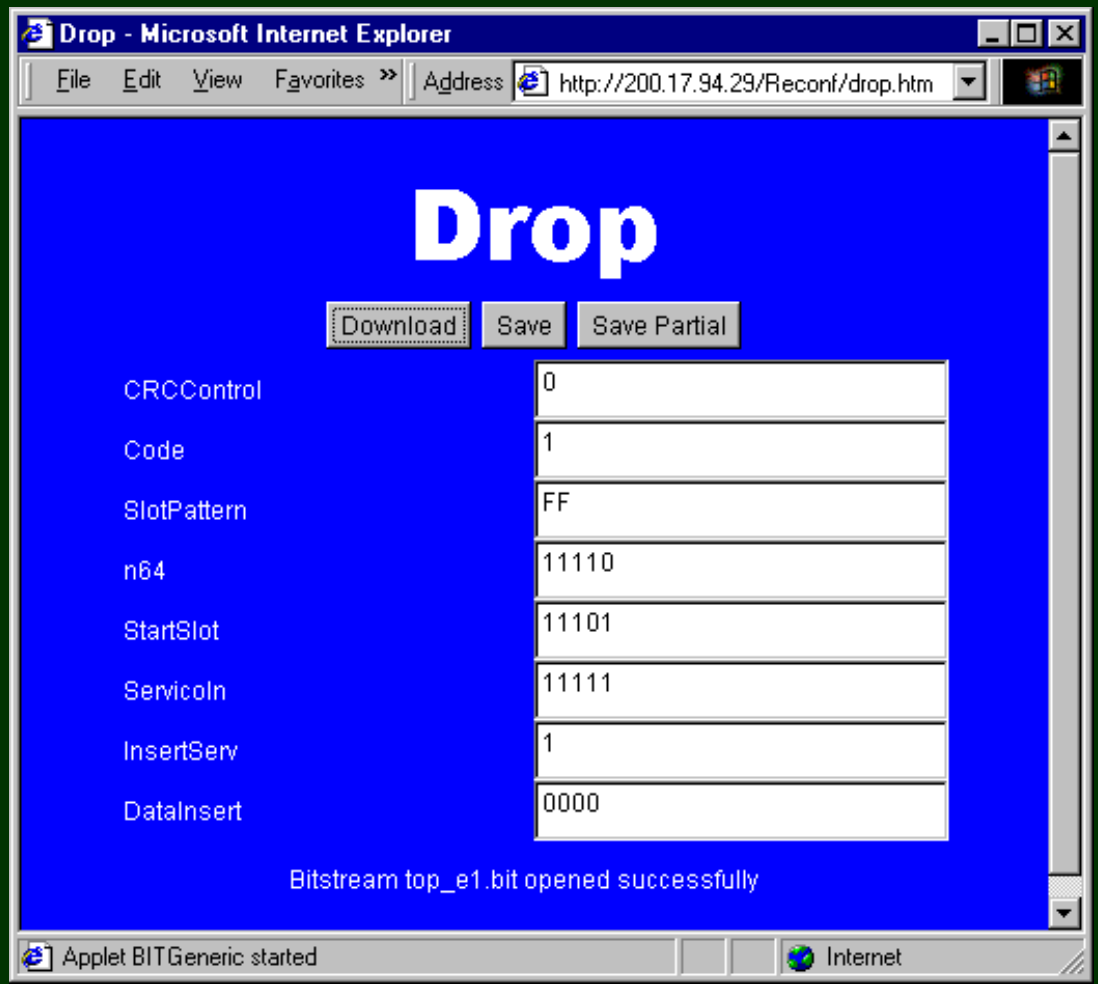
```
<APPLET code="BITGeneric.class" width=400 height=300>
<PARAM name="nbsignals" value="8">
<PARAM name="path" value="top_e1.bit">
<PARAM name="ip" value="200.17.94.29">
<PARAM name="port" value="5000">
<PARAM name="1[1]" value="CRCControl ,bin,32,37,G,0,0,0">
…
<PARAM name="1[8]" value="DataInsert ,hex,28,37,G,0,0,15">
</APPLET>
```

    - designer must indicate the physical position of the memory blocks containing parameters

mesquita@lirmm.fr

Tools for partial and remote reconfiguration

- **Circuit customization tool**
  - **Circuit user**
  - receives the bitstream and the HTML description
  - in the “reconfiguration page” the values of the signals can be modified, saved and partially downloaded into the device



mesquita@lirmm.fr

Tools for partial and remote reconfiguration

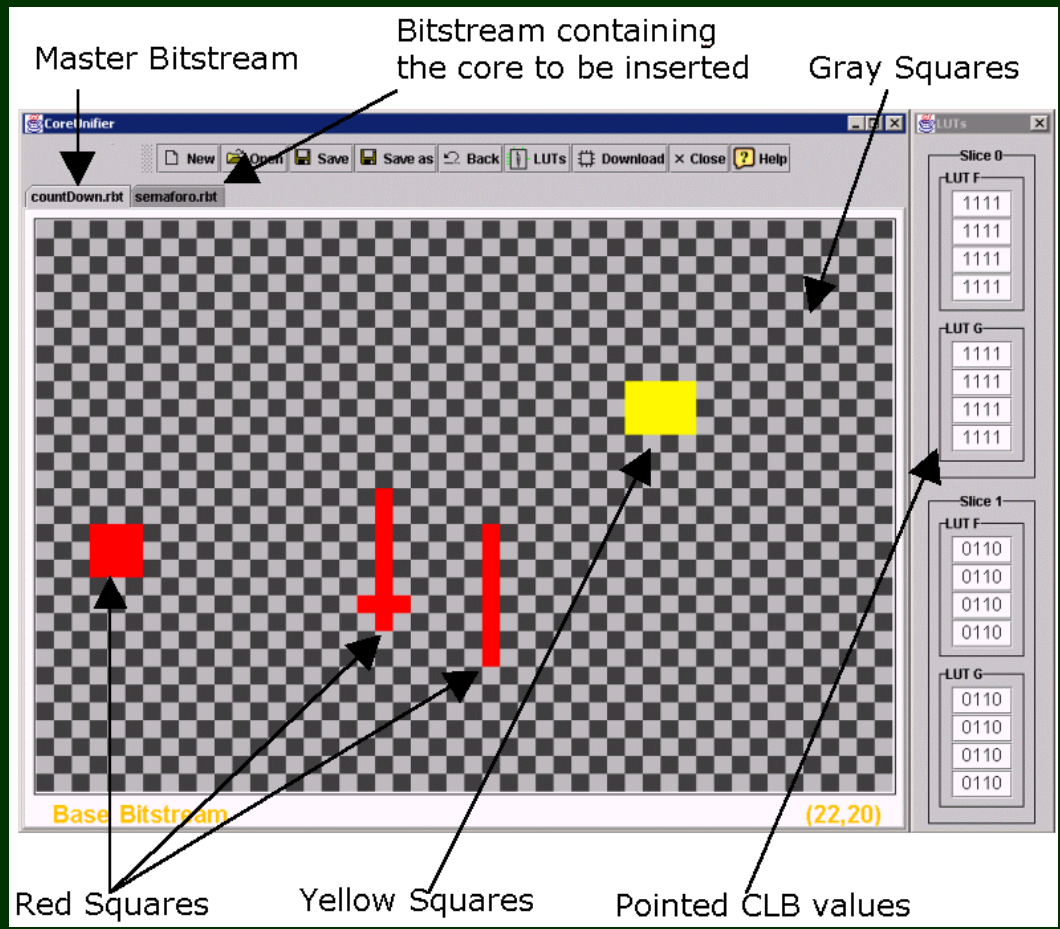
- **Core unifier tool**
  - Function:
    - Insert / remove partial bitstreams (hard cores) at runtime
  - Benefits:
    - Makes possible the hardware virtualization

mesquita@lirmm.fr

Tools for partial and remote reconfiguration

- **Core unifier tool**

Graphical interface

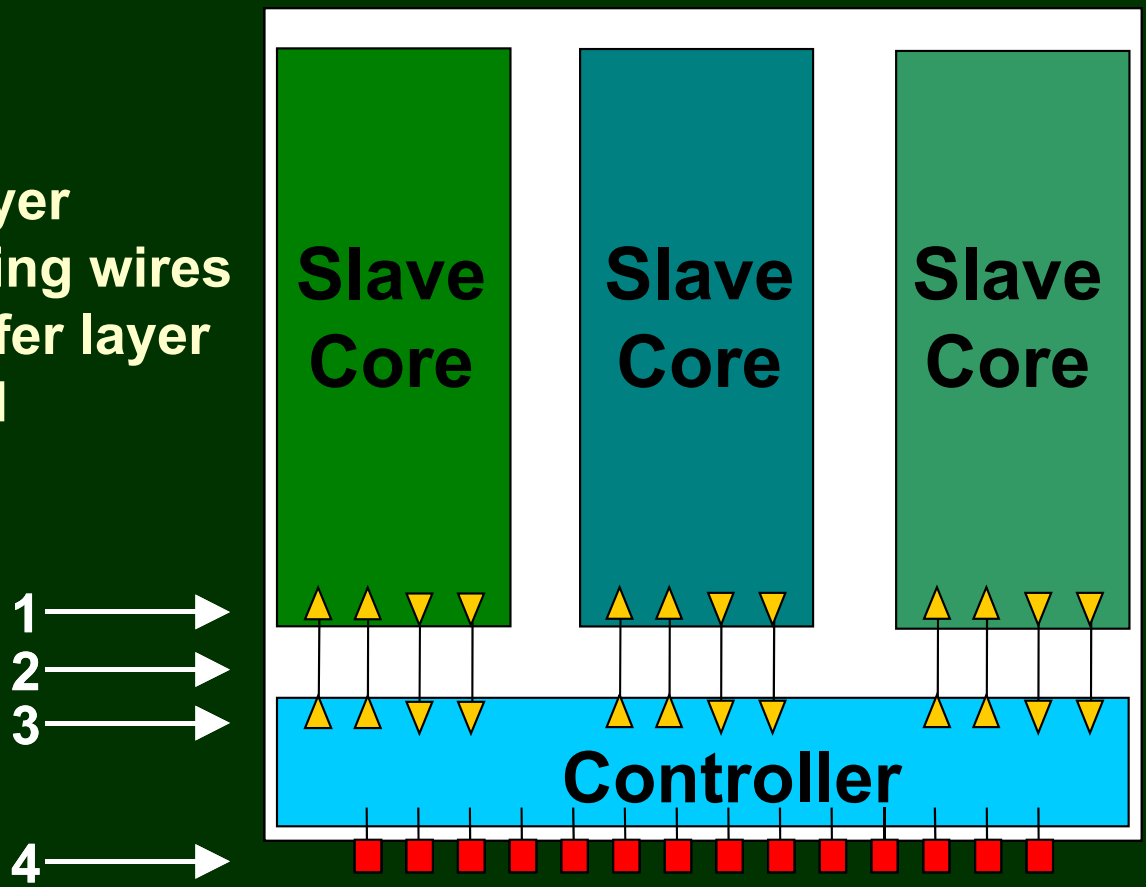


mesquita@lirmm.fr

Tools for partial and remote reconfiguration

- **Core unifier tool**

1. Core buffer layer
2. Common routing wires
3. Controller buffer layer
4. External world connection

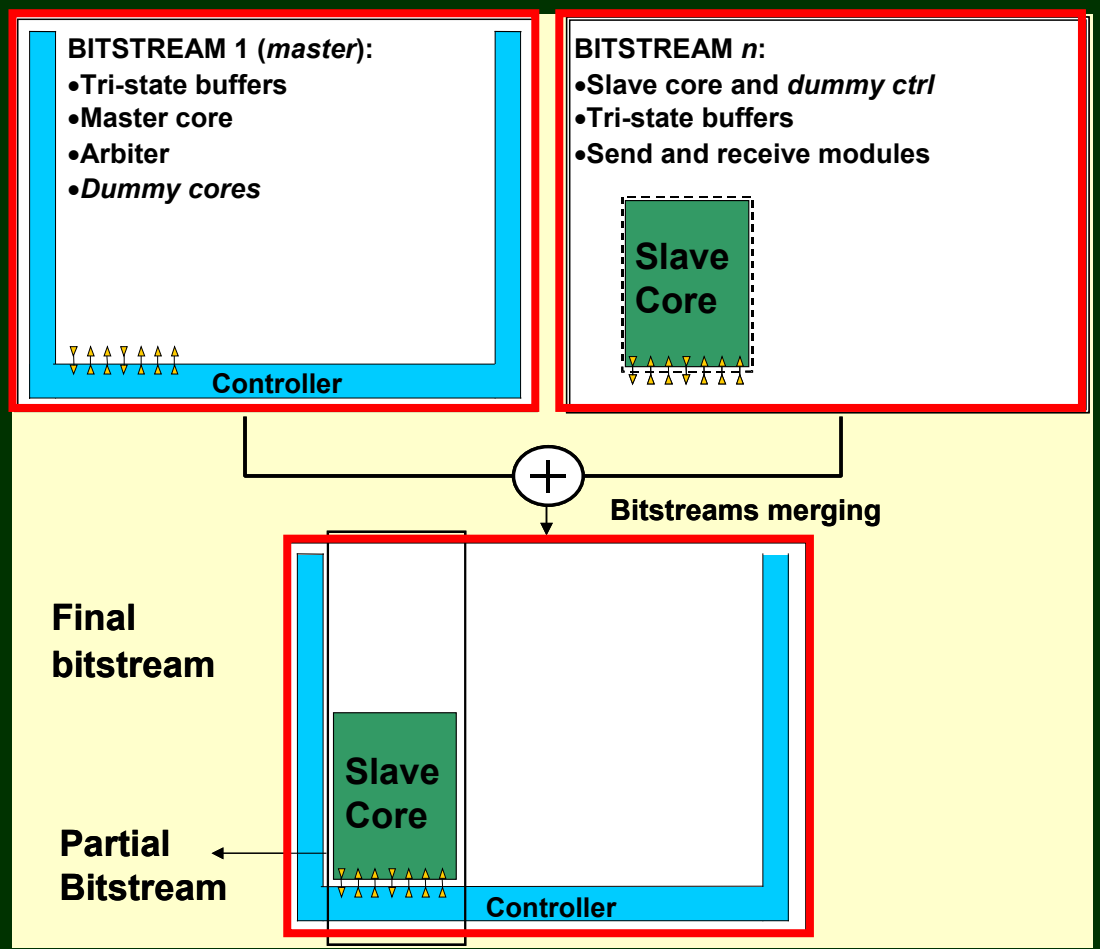


mesquita@lirmm.fr

Tools for partial and remote reconfiguration

- **Core unifier tool**

1. A complete master bitstream is opened
2. One or more bitstreams containing slave cores to be inserted into the master bitstream are opened
3. The user selects the area corresponding to one core, and all components inside this area (routing and CLBs) are inserted into the master bitstream



mesquita@lirmm.fr

Summary

- **Introduction**
- **State of art**
  - History and trends
  - Related work
- **Virtex internal organization**
  - Architecture overview
  - Addressing elements
- **Tools for partial and remote reconfiguration**
- **Conclusions**

mesquita@lirmm.fr

Conclusion

- **Contributions**
  1. State of art review, indicating trends
  2. Tool-set for remote, partial and dynamic reconfiguration
     - Remote reconfiguration is enabled
     - Parameter reconfiguration can be used to fix/modify a circuit
     - Virtual hardware is feasible with of-the-shelf FPGAs

- **Future works**
  1. To extend the bus structure, to analyze other arbitration schemes
  2. To develop CAD for the manual steps mentioned

mesquita@lirmm.fr

Conclusion

- «Final» conclusion ☺

The *core unifier* tool can be integrated with co-design tools. Currently, the hardware cores of a SOC require a programmable device having enough area to implement all cores. Another possibility is the generation of several small hardware cores by the co-design tool, with a scheduler to download these cores on-demand into the FPGA device. This can be seen as a "dynamic co-design", a new concept not yet explored.

mesquita@lirmm.fr