RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Register File Generation

Christian Leber
<leber@uni-hd.de>

Computer Architecture Group
http://cag.uni-hd.de
University of Heidelberg

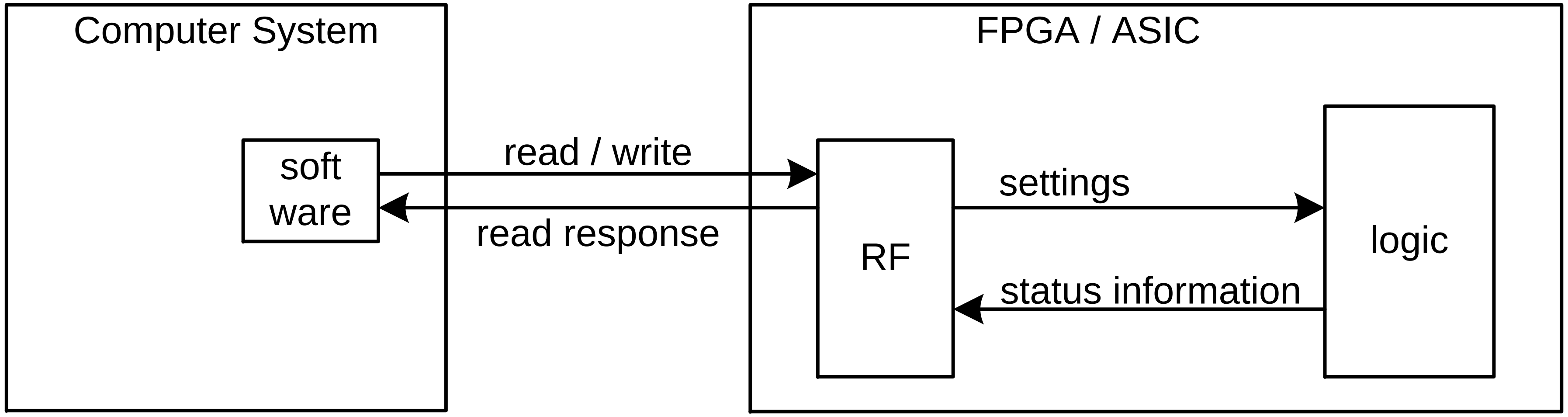
RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Overview

- Problem
- Other solutions
- Language Specification
- Example

Problem

- Hardware has to be controlled and status information has to be read



RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Problem

- RFs can contain different elements:
  - Classical CSR registers
  - Counters
  - RAMs
- RFs are not complex, but way too big to write them manually

RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

State of the „Art“

- Denali Blueprint
    - Was used previously at the CAG
    - SystemRDL
- Duolog Socrates Bitwise
    - IP-XACT (industry standard)
- Semifore CSRCompiler
    - IP-XACT (industry standard)
    - did not answer questions to sales
- Vregs
    - Extracts input from HTML tables, that is no expressive enough

RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

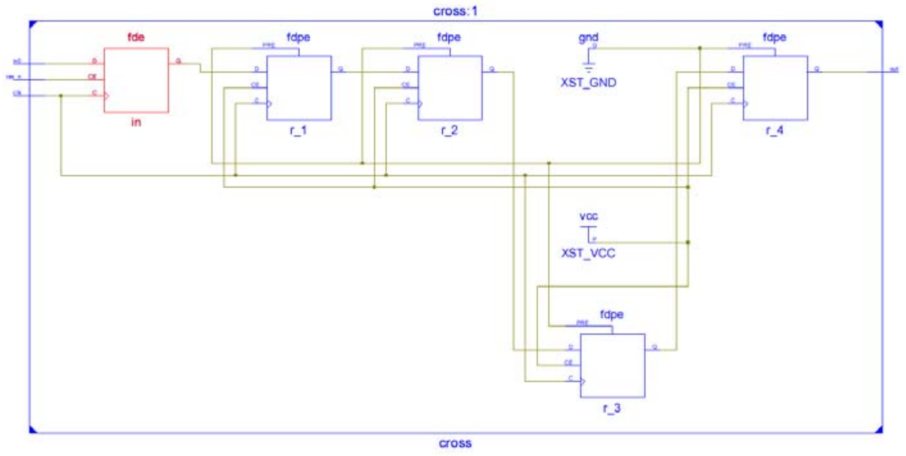
A new tool?

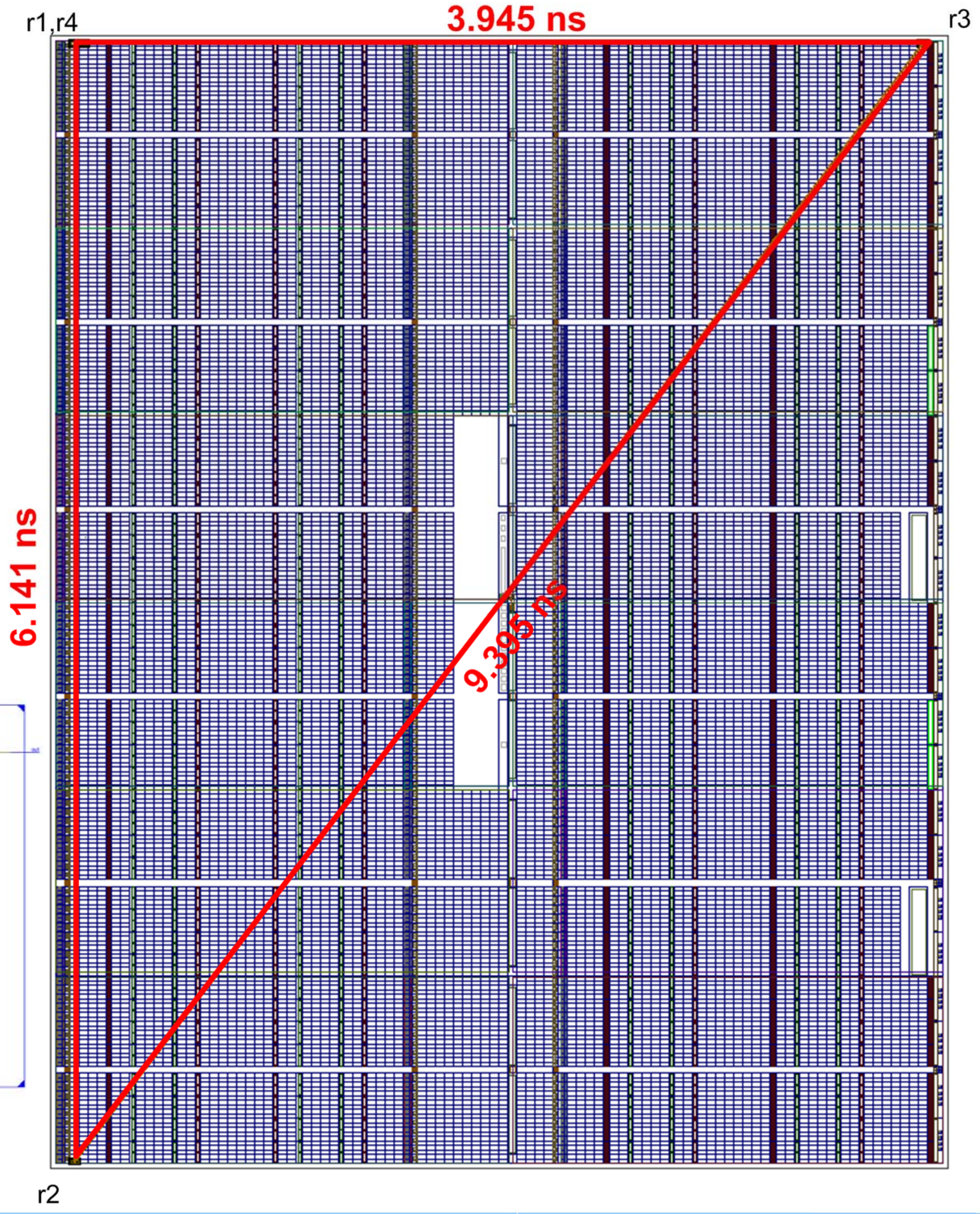
- Denali is expensive and has problems
- Clear hierarchical designs
- A new tools allows also to implement new features easily without battling with „vendor extensions“ and „plugin mechanisms“
- Name of the tool: RFS (Register File System)
- IP-XACT is obnoxious (by using “spirit” too often) and comes in different versions; we were interested in UVM support for RFs, but the Cadence package supported only IP-XACT 1.4, but we wanted a 1.5 feature.

  -
    ```
    <spirit:register>
      <spirit:name>rxtx0</spirit:name>
      <spirit:description>Data receive/transmit register 0</spirit:description>
      <spirit:addressOffset>0x0</spirit:addressOffset>
      <spirit:size>32</spirit:size>
      <spirit:access>read-write</spirit:access>
      <spirit:reset>
              <spirit:value>0x00000000</spirit:value>
              <spirit:mask>0xffffffff</spirit:mask>
      </spirit:reset>
    </spirit:register>
    ```

RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

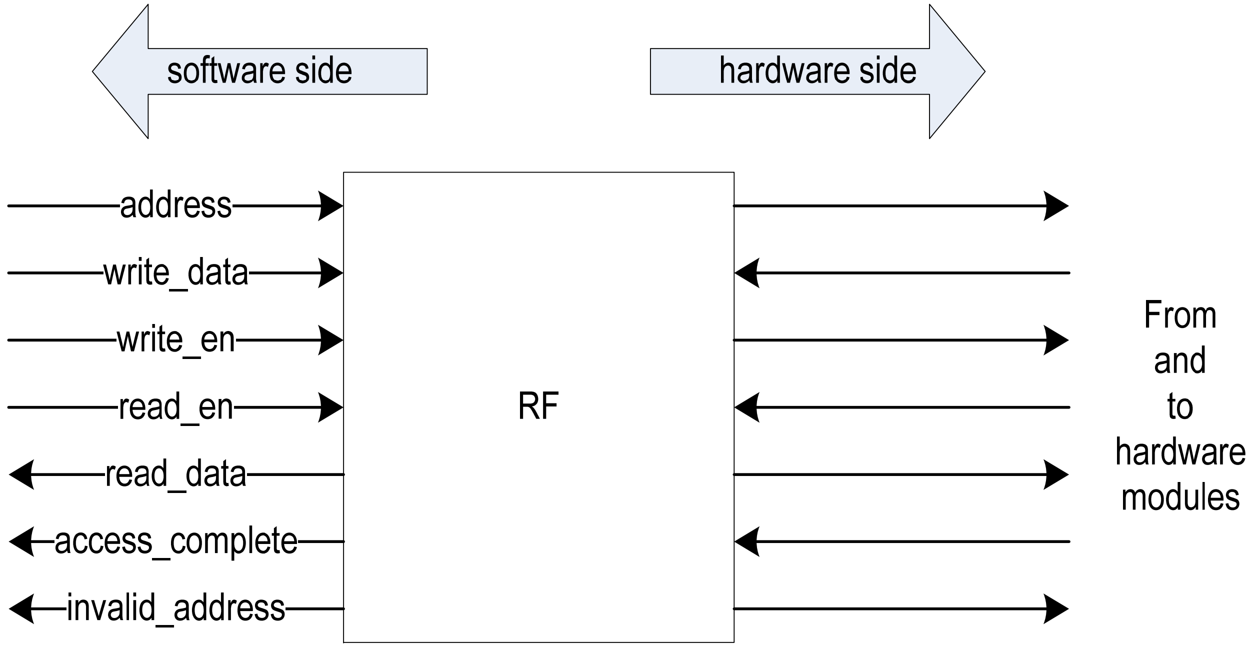
- Middle sized FPGA: Virtex6 240K (XC6VLX240T)
- Single RF can not work at useful frequencies

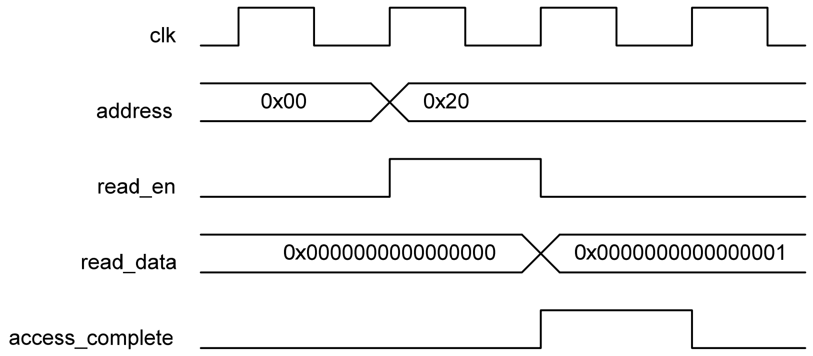


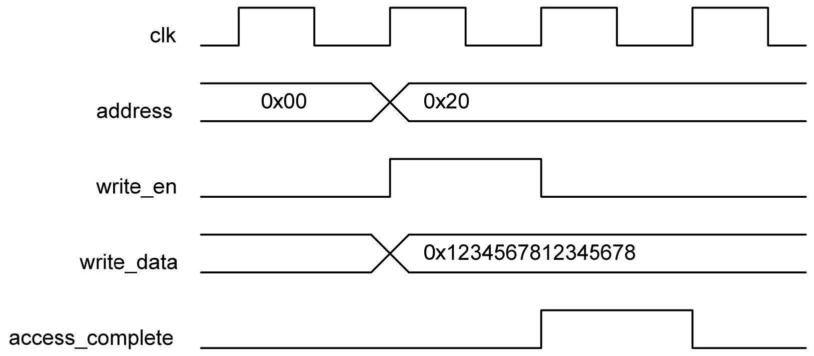


RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Termonology

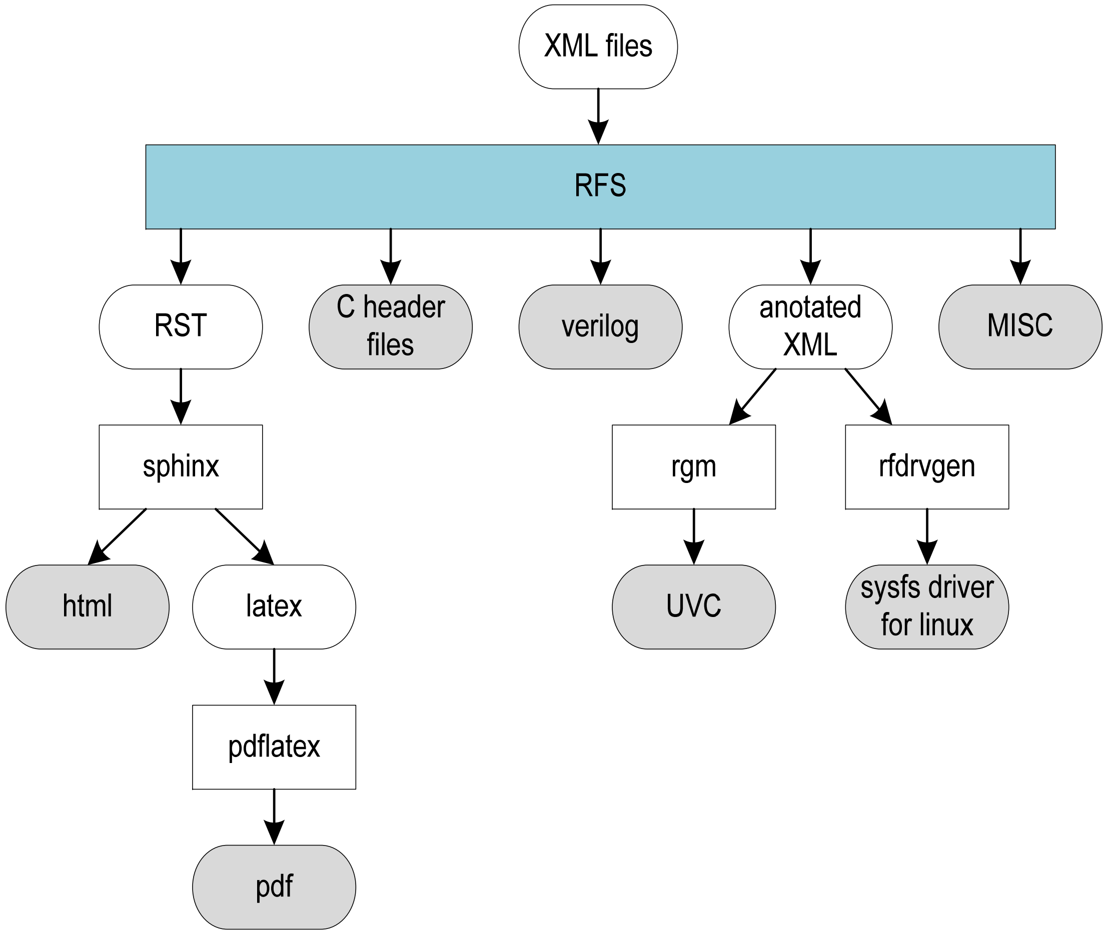






RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

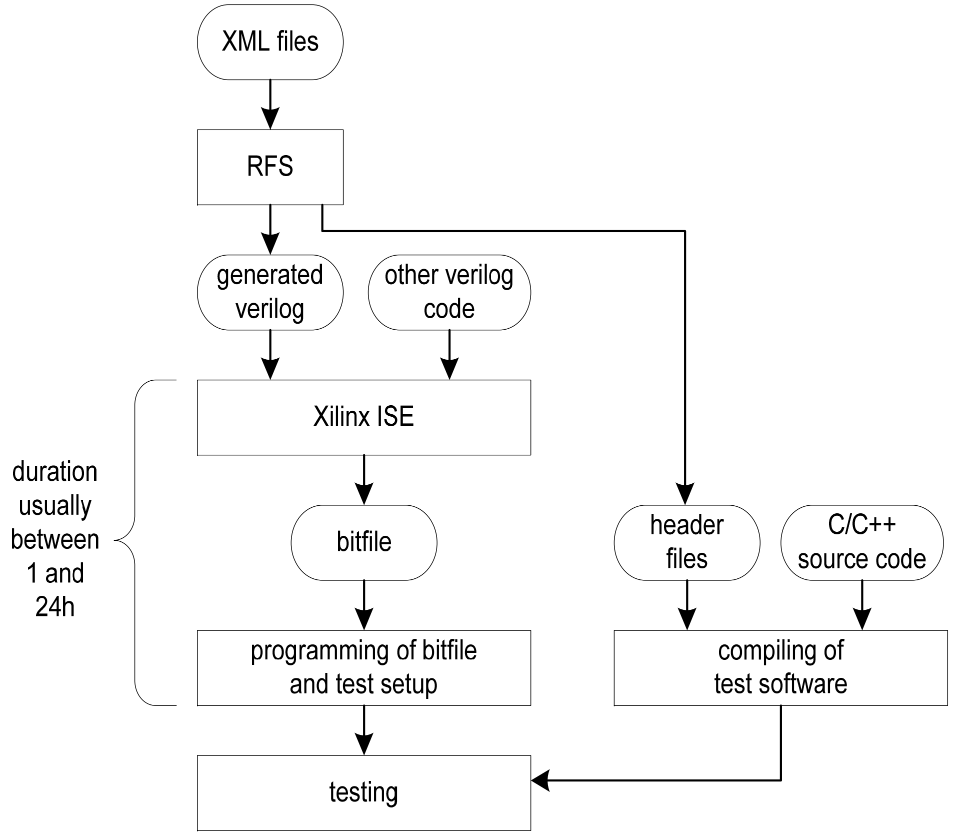
Flow



RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Flow

- Possibilities to ensure consistency between HW and SW
- RF version
  => you will forget to update it
- Complete XML in RF
  => a bit wasteful
  => probably no ROM in the ASIC
- $seconds
  store the generation time of the RF in the RF and in a C header file



RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Specification

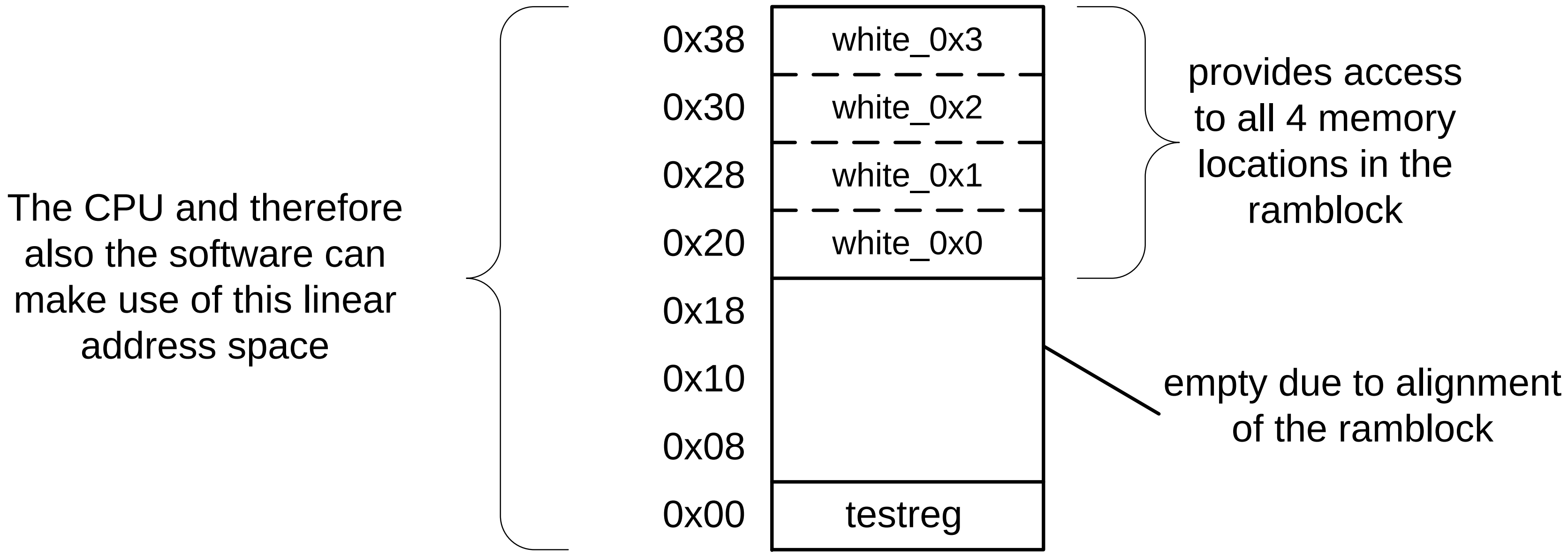
project.xml

```
<?xml version="1.0"?>
<vendorsettings><somesetting/></vendorsettings>
<regfile>
        <doc name="test" project="test"  version="0.1“ author="test"  copyright="nobody"/>
        <wrapper value="htax">
                <rrinst name="top“ file="top.xml" />
        </wrapper>
</regfile>
```

top.xml

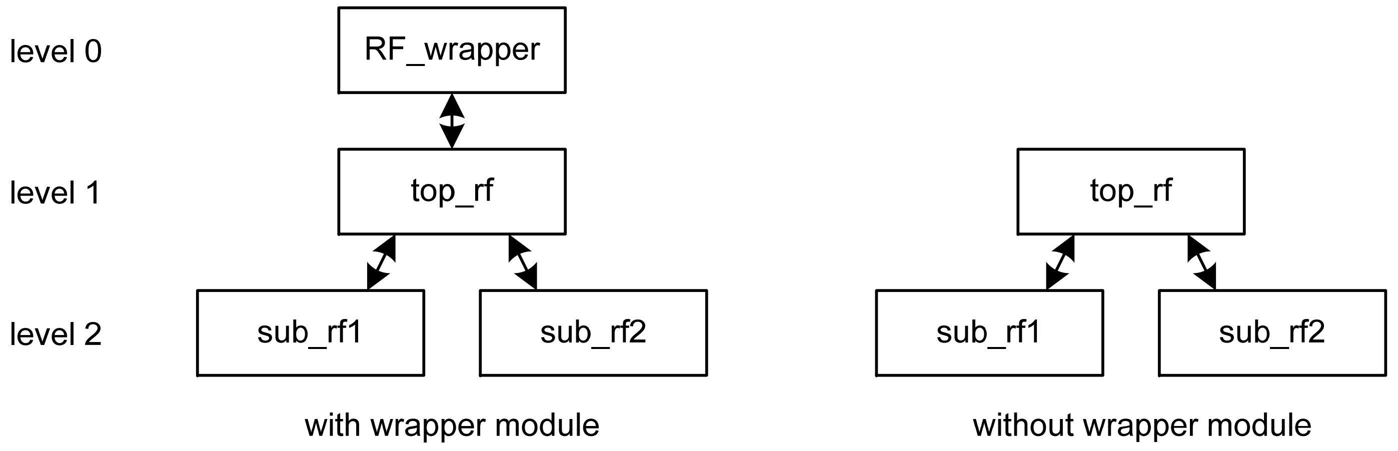
```
<?xml version="1.0"?>
<regroot>
        <reg64 name="testreg">
                <hwreg width="64" sw="rw" hw="rw" reset="$zero" />
        </reg64>
        <ramblock name="white" addrsize="2" ramwidth="64" sw="wo" hw="ro"/>
</regroot>
```

Specification


0x38
white_0x3
0x30
white_0x2
0x28
white_0x1
0x20
white_0x0
0x18
0x10
0x08
0x00
testreg
provides access to all 4 memory locations in the ramblock
empty due to alignment of the ramblock
The CPU and therefore also the software can make use of this linear address space

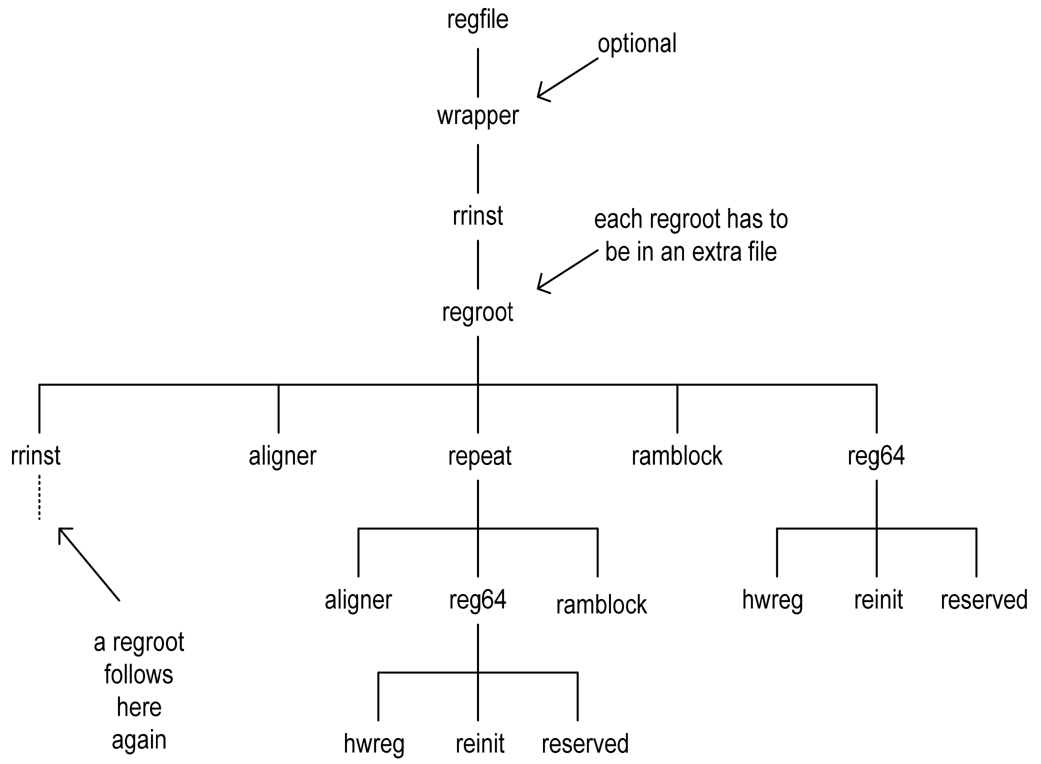
RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Specification



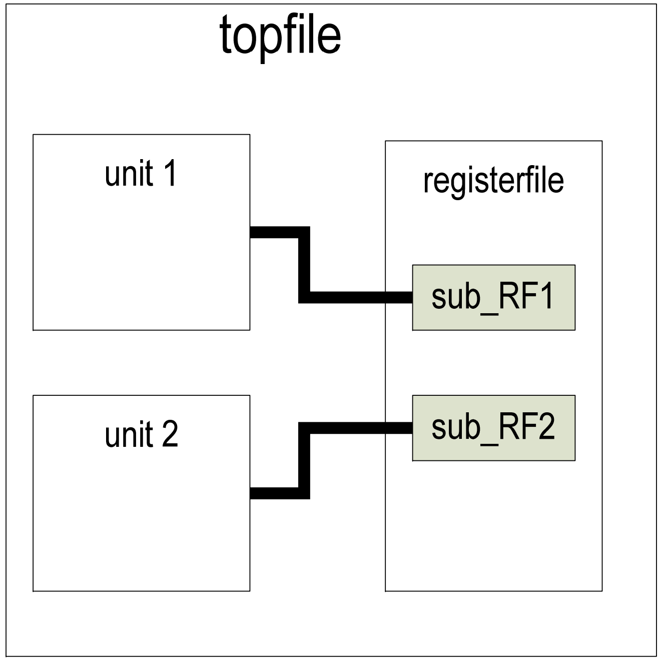
RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Specification

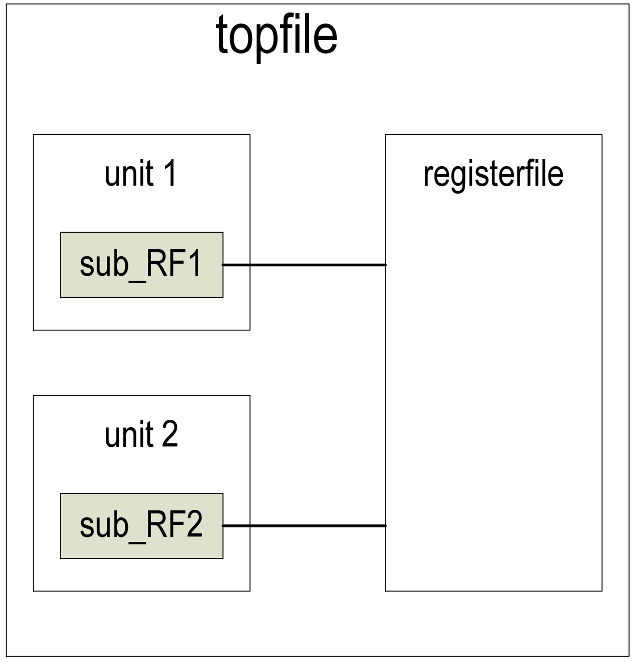


RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

External regroot



without external regroots



with external regroots

Thick lines symbolize big amounts of wiring

RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

sysfs

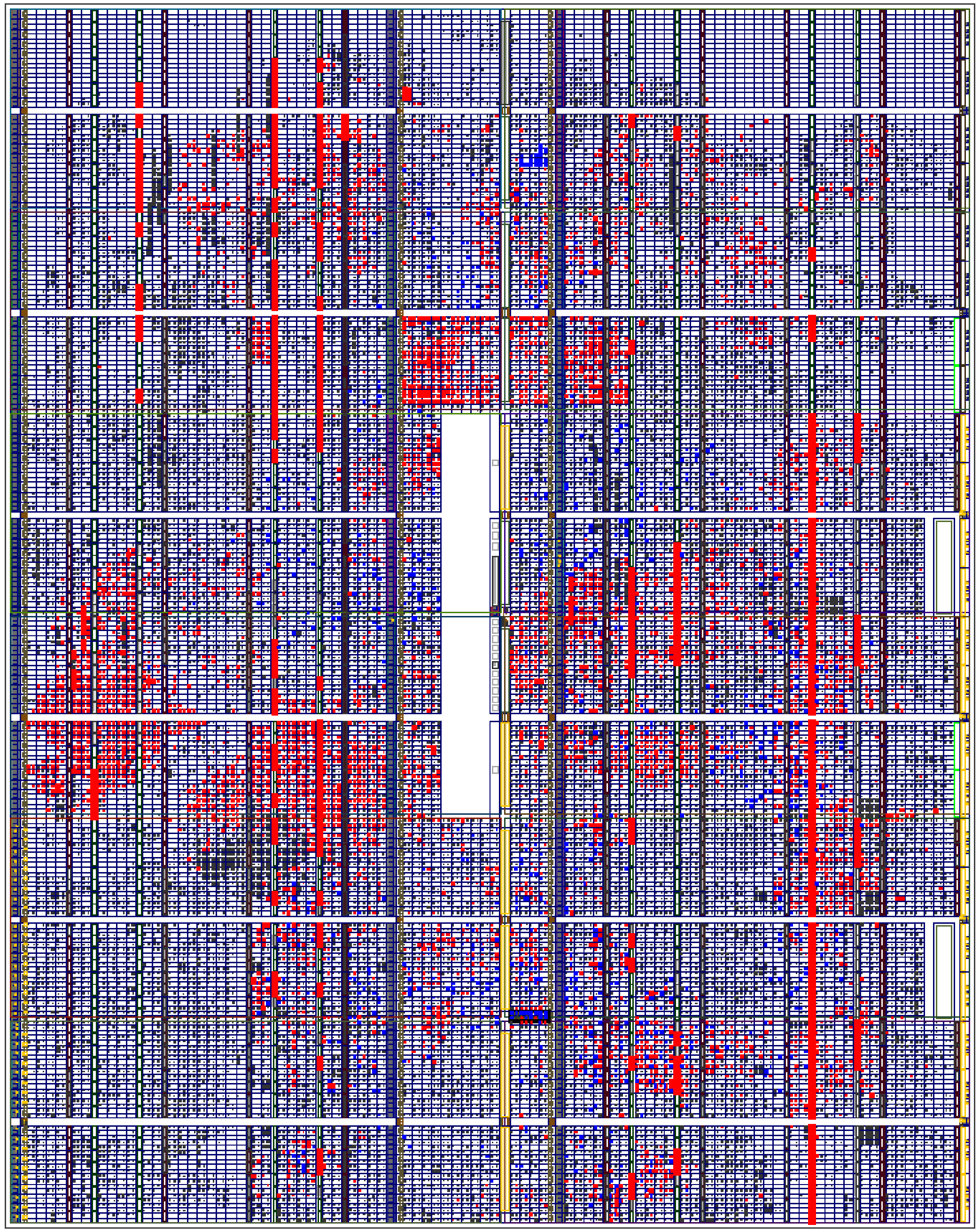
- Virtual linux filesystem
- Code auto generated

```
root@dl165b:/sys/devices/pci0000:20/0000:20:06.0/net_rf# cat tcp_rf_target_mac
addr: cd7e7a211b00; valid: 1;
root@dl165b:/sys/devices/pci0000:20/0000:20:06.0/net_rf# echo 0xAABBCCDDEE > tcp_rf_target_mac
root@dl165b:/sys/devices/pci0000:20/0000:20:06.0/net_rf# cat tcp_rf_target_mac
addr: aabbccddee; valid: 0;
```

Example Project

- EXTOLL R2 design
- stats_num_regroot=52
  stats_num_reg64=502
  stats_num_reg64_hwreg=1313
  stats_num_reg64_hwreg_counter=257
  stats_num_reg64_hwregbits=18385
  stats_num_ramblocks=56
- Output:
  30 Verilog files
  22273 lines of code (617 kB)

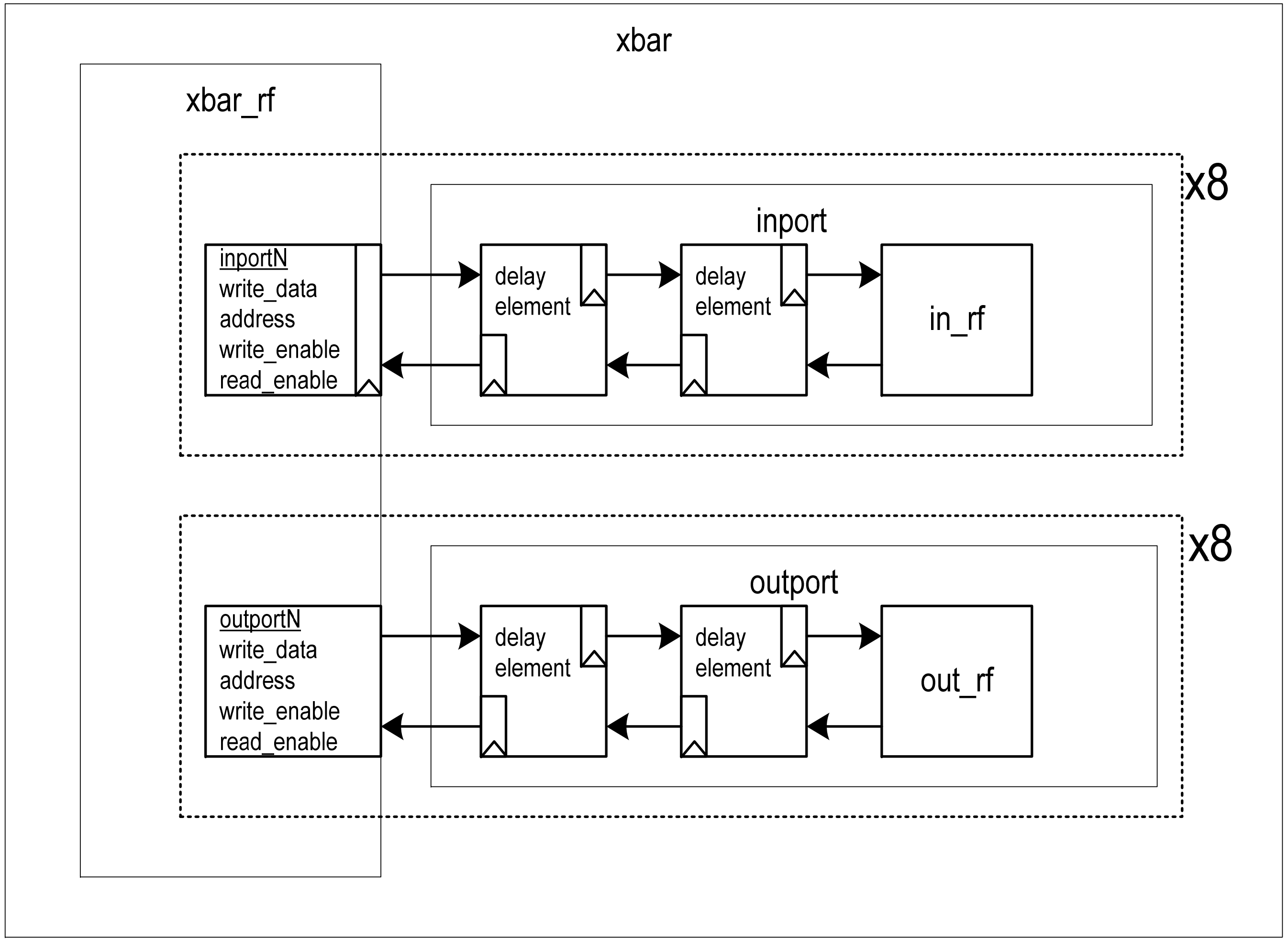
RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG



Red: RFs
Blue: delay elements to improve timing

RUPRECHT-KARLS-
UNIVERSITÄT
HEIDELBERG

Example Project



Last Slide

Thanks for your attention

Questions?