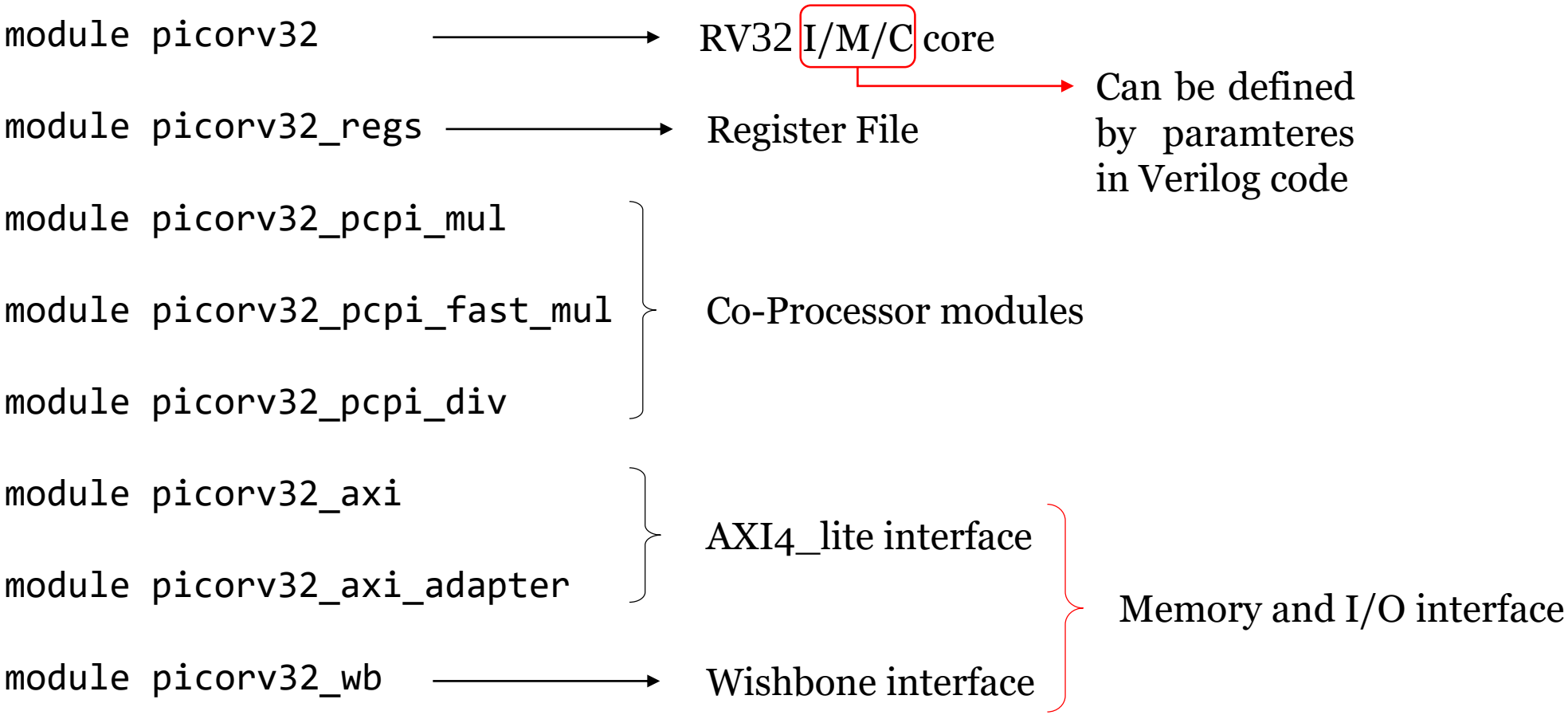
Approximation-aware Partitioning for Periodic Tasks on an Approximate-Exact MPSoC

by: Stefan Huemer

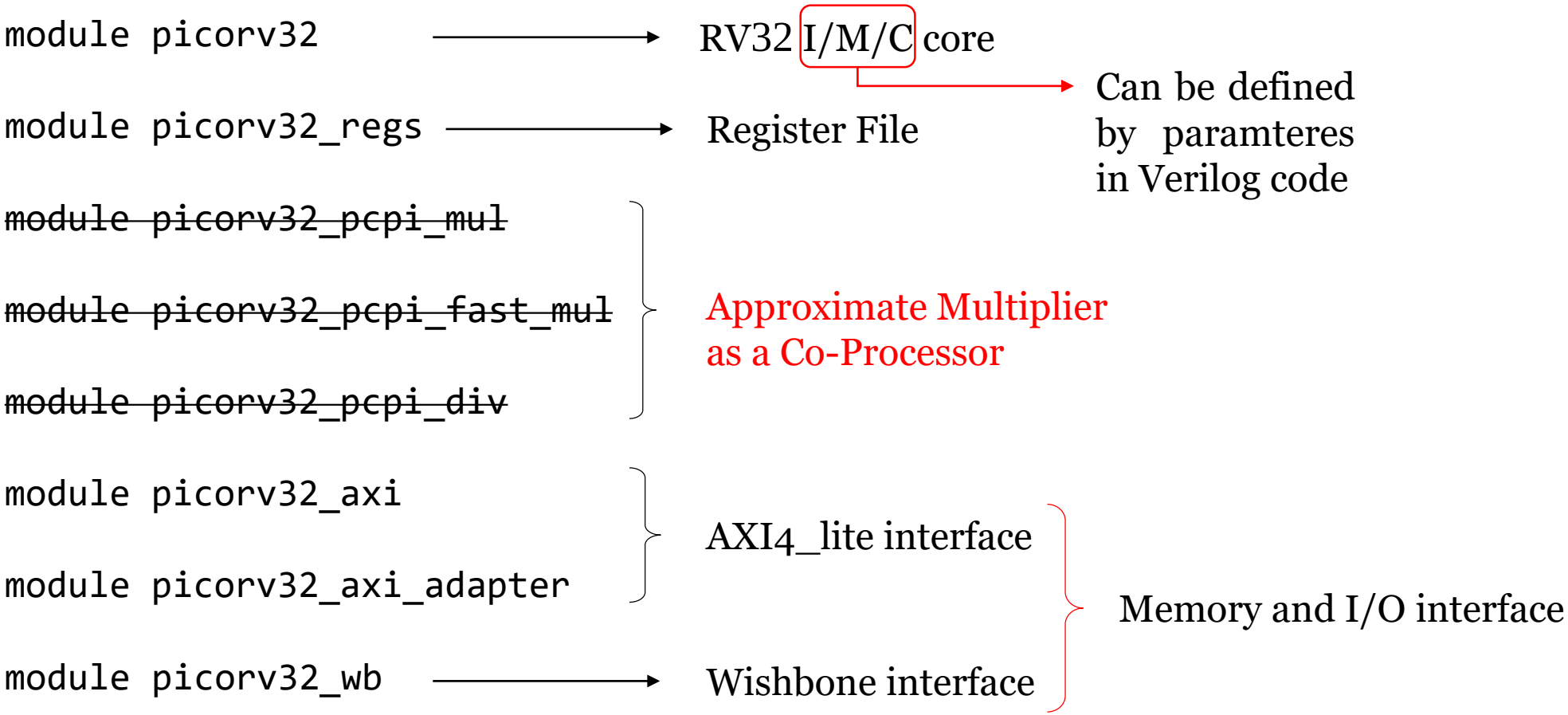
A review on the RISC-V core used in the project:

- Modules that connect to each other using the AXI-Light interface.
- Using these modules, bigger components can easily be constructed, for example a Node, consisting of a CPU module among others.

PicoRV32 structure and modules:



PiXoRV32 structure and modules:

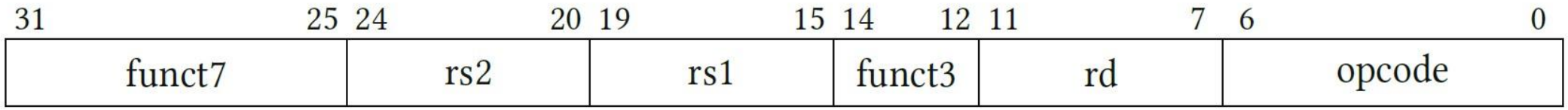


New R-Type instruction defined in RISC-V ISA:

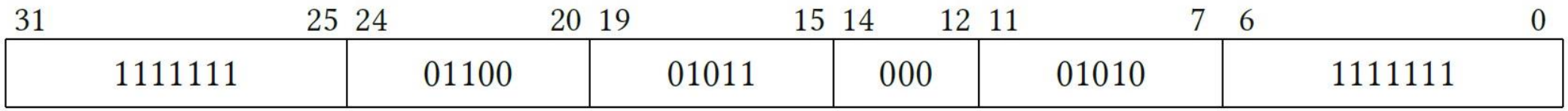
- Hard coded word with the value 0xF EC5857F
- R-Type instruction with an opcode not used by original ISA

- C code for approximate multiplication function:

```
__attribute__((noinline))
int amul( int rd , int rs1, int rs2)
{
asm__volatile__(".word␣0xFEC5857F\n");
asm__volatile__("addi␣%0,␣x10,␣0":"=r"(r));
return rd;
}
```



R-Type instruction



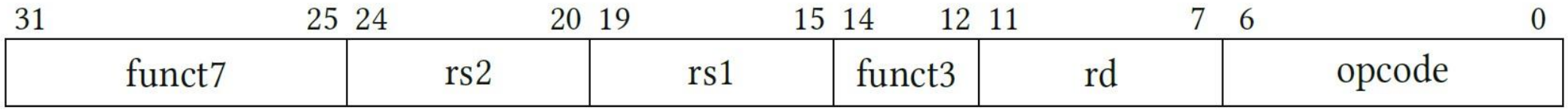
R-Type instruction with encoded bits

New R-Type instruction defined in RISC-V ISA:

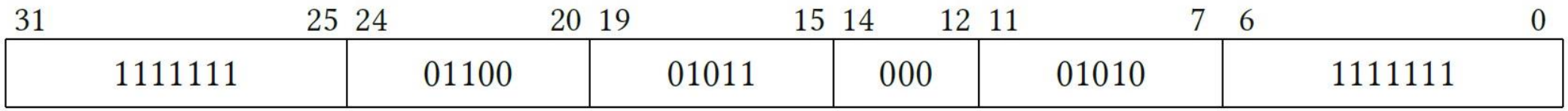
- Hard coded word with the value 0xF EC5857F
- R-Type instruction with an opcode not used by original ISA

- C code for approximate multiplication function:

```
__attribute__((noinline))
int amul( int rd , int rs1, int rs2)
{
asm__volatile__(".word␣0xFEC5857F\n");
asm__volatile__("addi␣%0,␣x10,␣0":"=r"(r));
return rd;
}
```

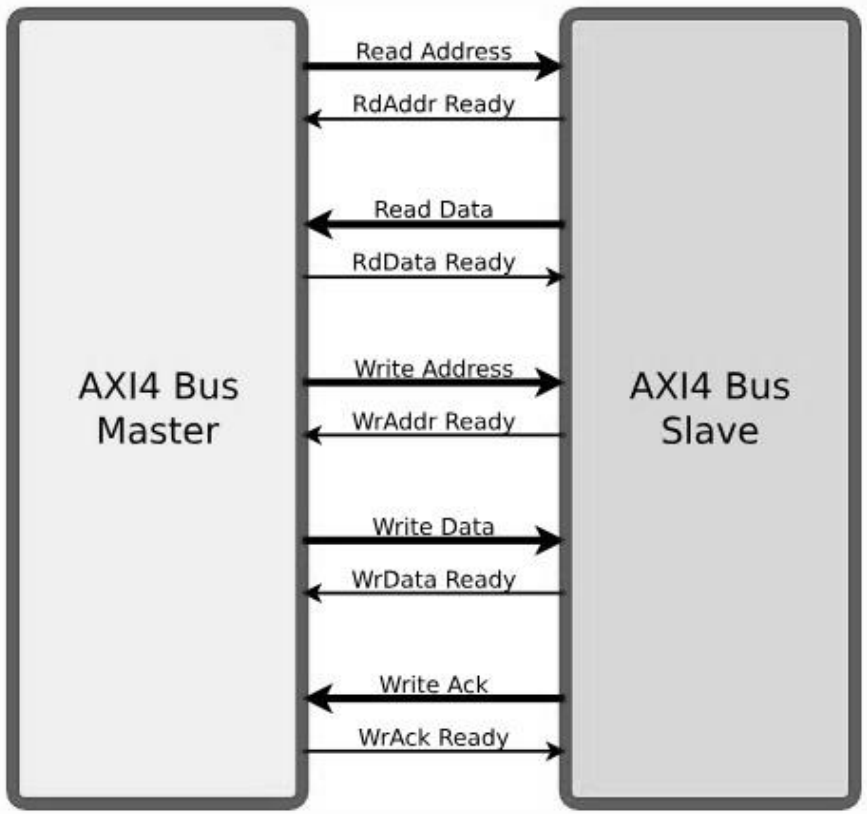


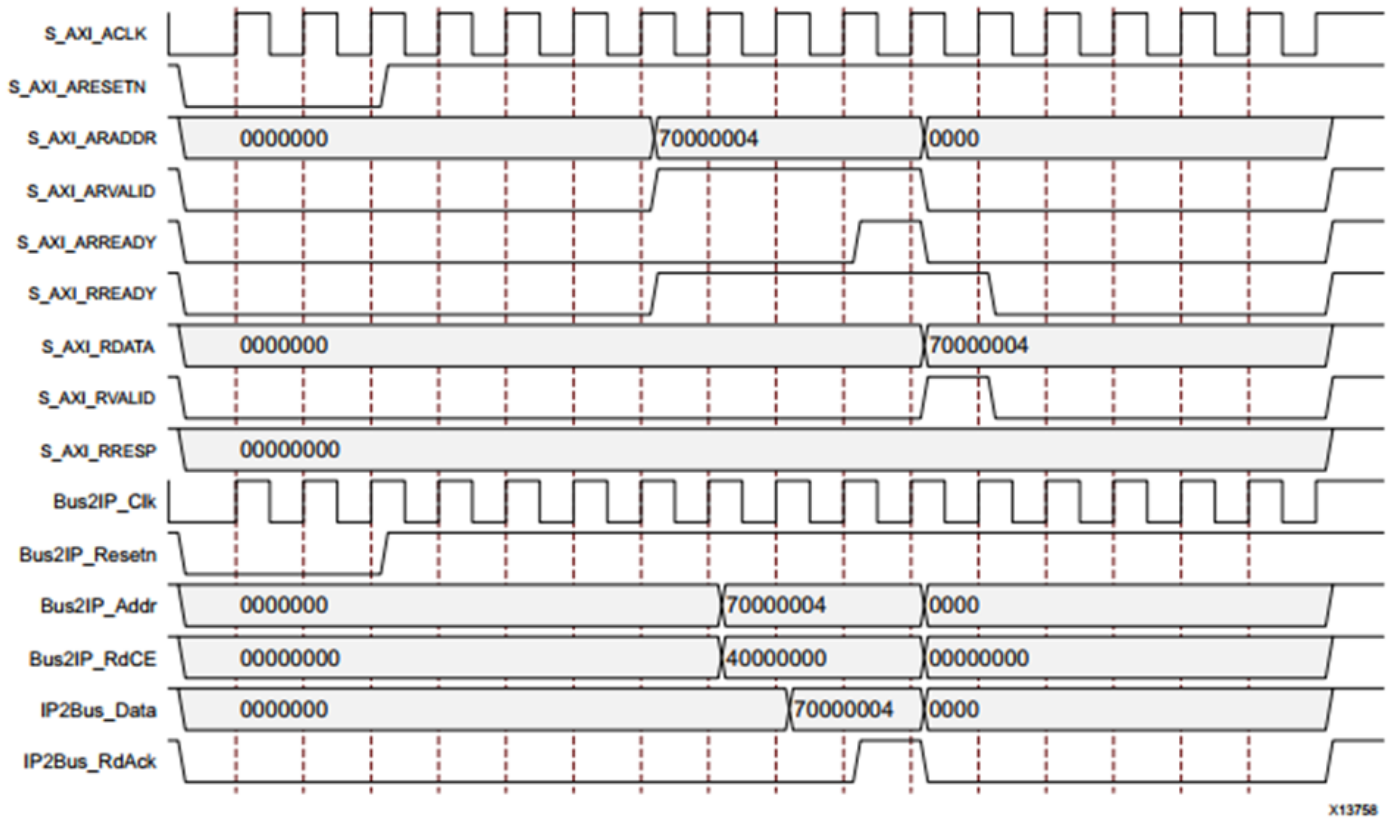
R-Type instruction



R-Type instruction with encoded bits

- NoC and PicoRV32 connection: AXI4-Lite





Implementation and Simulations:

- Software: C++ application GNU / Verilator
- Synthetize tool: Xilinx Vivado (FPGA)
- Memory Packer: BRAM on FPGA