Extending RISC-V Datapaths with Coarse-Grained Reconfigurable Architectures

CEI UPM
Centro de Electrónica Industrial

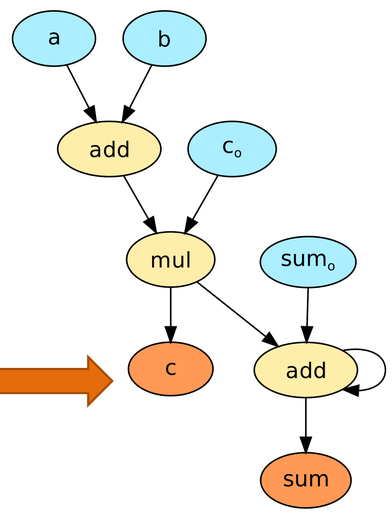
Daniel Vázquez, Alfonso Rodríguez, Andrés Otero, Eduardo de la Torre
daniel.vazquez@upm.es

RISC-V Summit Europe 2023

Coarse-Grained Reconfigurable Architectures

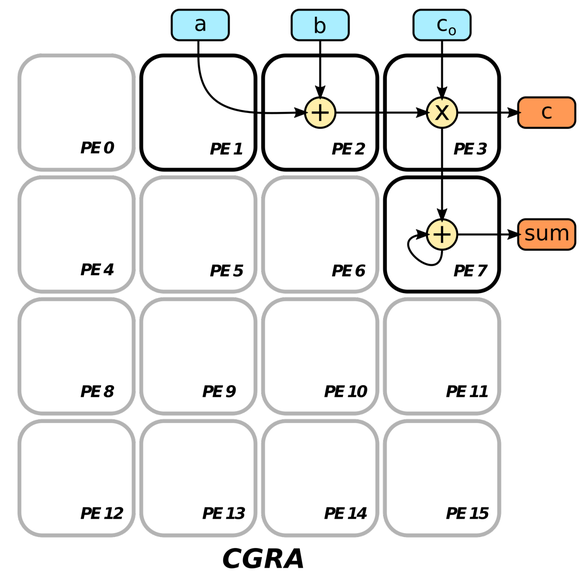
- CGRAs are domain-specific hardware accelerators that can process a wide range of applications with higher energy efficiency and lower execution times when compared to CPUs
- The application offloading consists on the generation of a data-flow graph and its mapping into the architecture

```
#define LOOP_SIZE 1000
void accumulate(int *a, int *b, int *c, int *sum)
{
    for (int i = 0; i < LOOP_SIZE; i++) {
        c[i] *= a[i] + b[i];
        *sum += c[i];
    }
}
```



CGRA specifications:

- Elastic spatially-configured CGRA
- Architecture made of equal Processing Elements with inputs in the north PEs and outputs in the east PEs
- Supported operations:
  - Add, sub, mul, shifts, AND, OR and XOR
- The functionality, the CGRA bitstream that defines the interconnections and operations performed in each PE, can be changed between executions



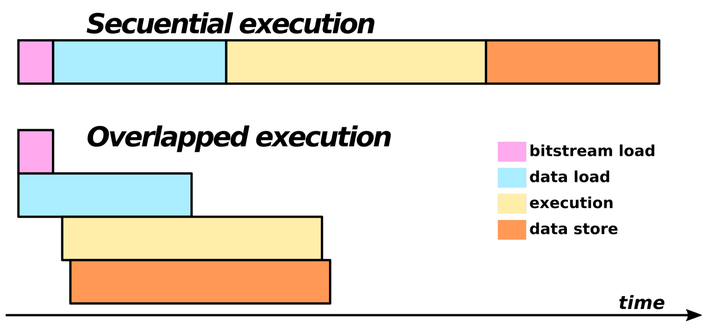
Integration into a System on Chip: Why RISC-V?

Key points to maximize the CGRA performance:

1. Reduce the **control overhead** to increase the performance when offloading consecutive kernels into the CGRA
2. Have a good **connection** with the main memory to fully exploit the hardware accelerator

2 **Use dedicated Direct Memory Access nodes for each input and output of the CGRA to feed it continuously, and also for loading the CGRA bitstream**

- Avoid a scratchpad memory: limits the vector lengths and forces a secuential execution of the applications
- Use a direct connection to main memory to load the CGRA bitstream, and load and store application data: execution can be **overlapped**



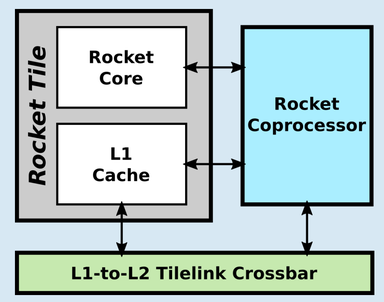
1 **Treat the CGRA as an extension of the processor datapath, using custom instructions for loading and storing the data, and setting up the CGRA bitstream (operations performed)**

Instruction	funct7 code	RS1	RS2
start	0	-	-
clear	1	-	-
bitstream_load	7	address	-
input_load	8-23	address	size and stride
output_store	24-39	address	size

This instruction space allows the control of CGRAs with size up to 16x16, a maximum vector length of 2^{16} when $xLen = 32\ bits$, and a maximum vector length of 2^{32} when $xLen = 64\ bits$

Use RocketChip [1] and Chipyard [2]:

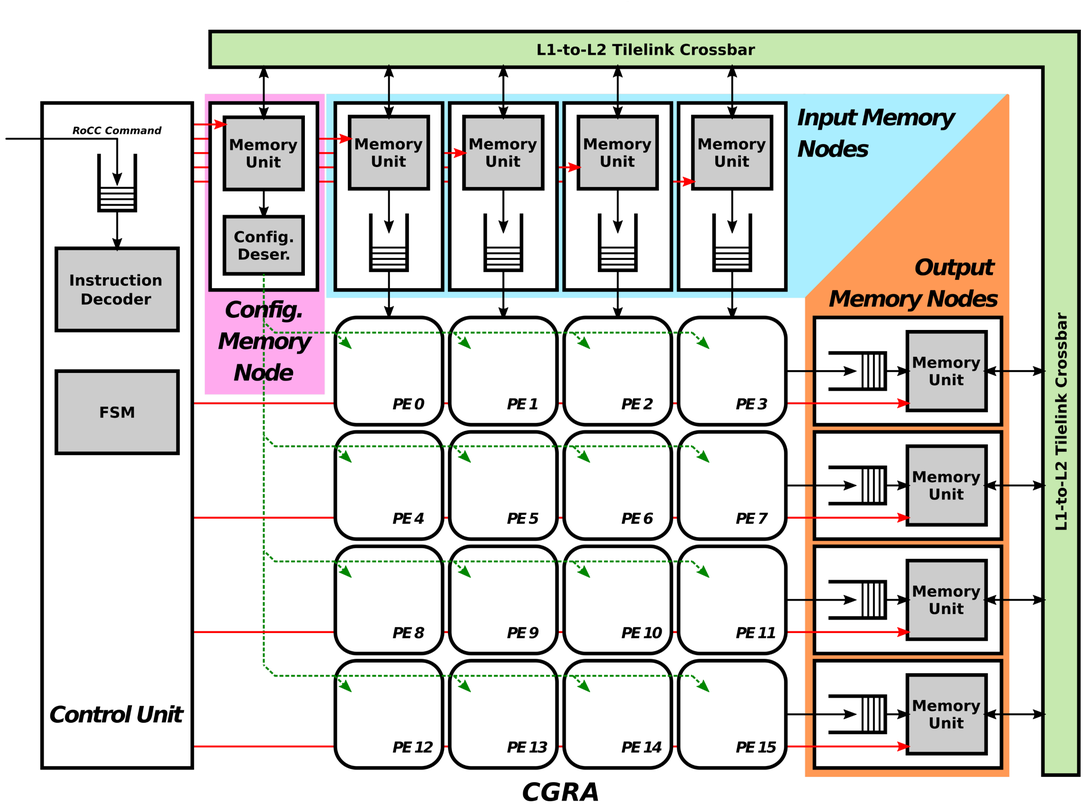
- Extend the RISC-V ISA through the RoCC interface
- Direct connection to L1 and L2 caches (high-bandwith connections), and FPU



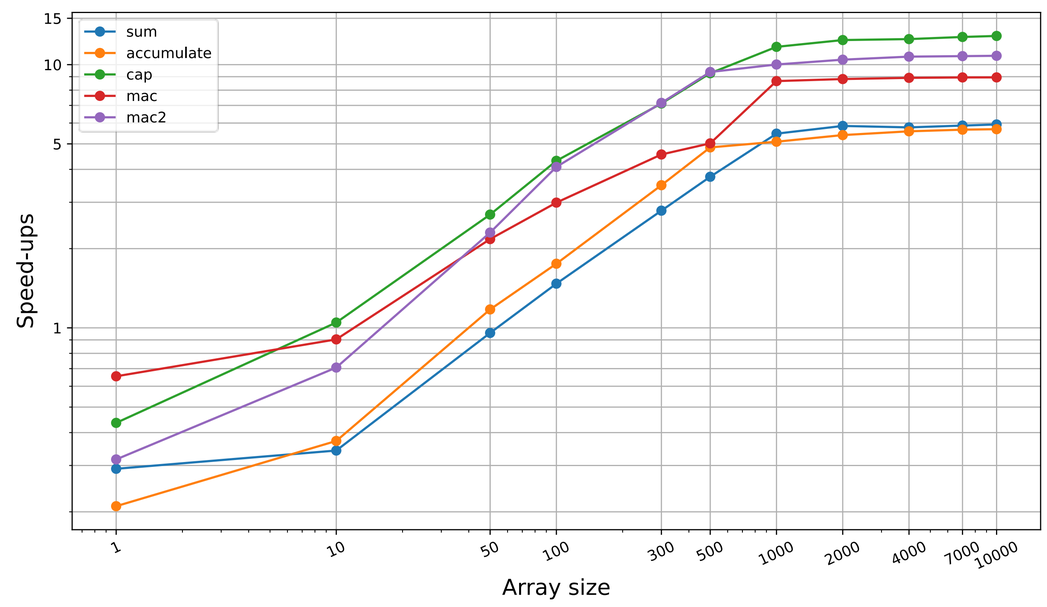
[1] Asanović et al. "The Rocket Chip Generator". Tech. rep. UCB/EECS-2016-17. EECS Department, University of California, Berkeley, Apr. 2016
[2] Alon Amid et al. "Chipyard: Integrated Design, Simulation, and Implementation Framework for Custom SoCs". In: IEEE Micro 40.4 (2020)

Proposed RISC-V coprocessor

- ISA extension to control the CGRA Accelerator. Centralized control but independent behaviour between memory nodes
- Parametrizable with the CGRA size, generating one Input Memory Node for each column and one Output Memory Node for each row. One Configuration Memory Node to fetch the CGRA bitstream
- Each memory node is connected to the L1-to-L2 Tilelink crossbar (cached) and uses burst to read and write cache data blocks of 64 bytes



Results and conclusions



- System tested with synthetic applications in a 6x6 CGRA integrated in Chipyard. Best speed-up obtained in *cap*, algorithm that uses 17 PEs and has 8 operations
- Xilinx 7000 series FPGA implementation with a 32-bit Small Rocket Core (RV32IMAC), 64 kB L1I cache, 64 kB L1D cache, 512 MB L2 cache, and a 2 GB external DDR RAM

- **SoC enhanced with spatially-distributed computing capabilities to accelerate computing-intensive sections of code**
- **By extending the RISC-V ISA the control overhead is minimized**
- **By overlapping the memory transfers and execution, speed-ups can be obtained from small array sizes**

UNIVERSIDAD POLITÉCNICA DE MADRID

This work has been funded by the MCIN/AEI /10.13039/501100011033 and the European Union NextGenerationEU/PRTR under the project COGNITION (ref. TED2021-132475B-I00)

MINISTERIO DE CIENCIA E INNOVACIÓN
UNIÓN EUROPEA Fondo Social Europeo El FSE invierte en tu futuro
AGENCIA ESTATAL DE INVESTIGACIÓN