Modern Computer Architectures

ET4074 - 2018/2019

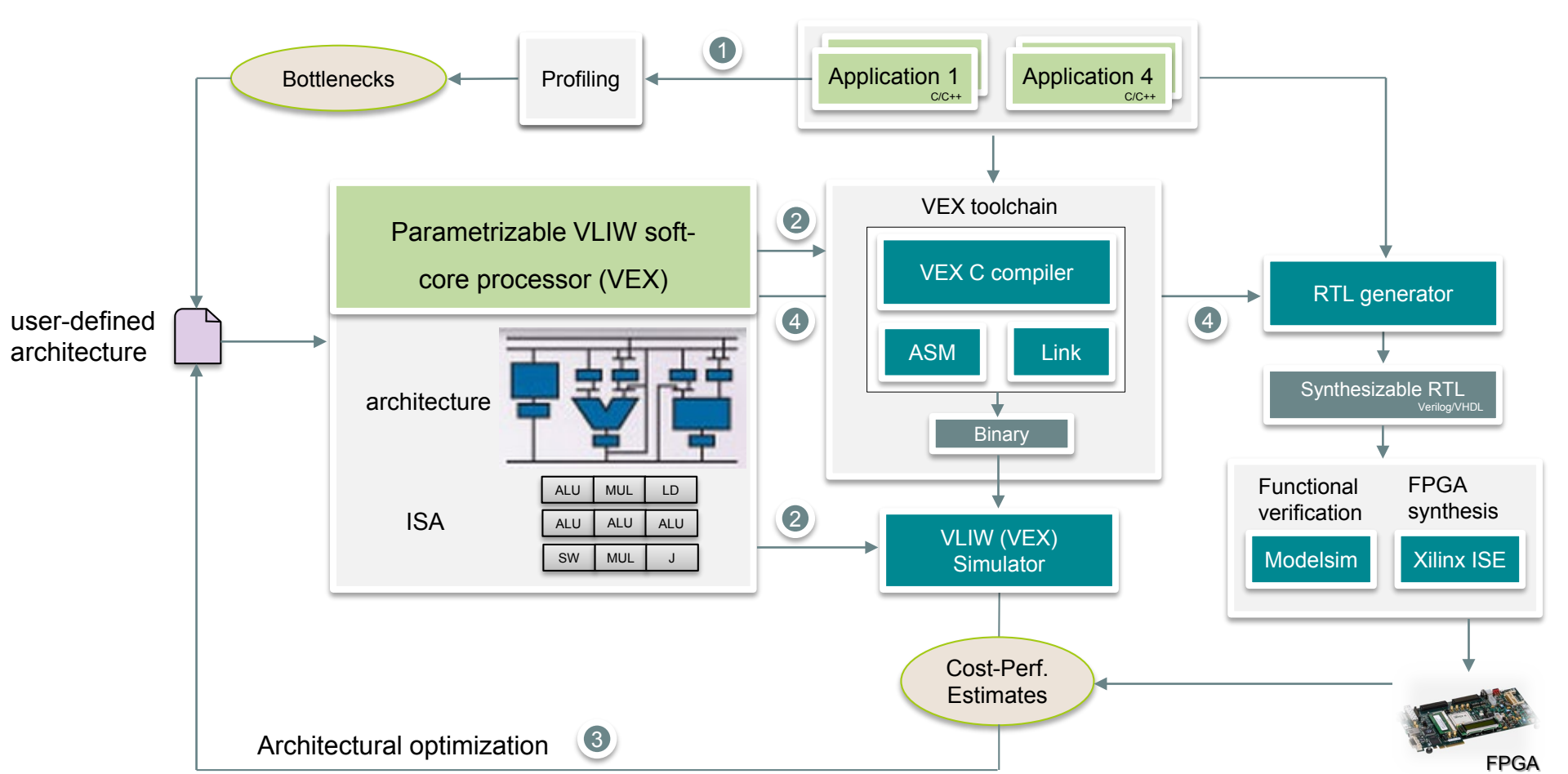
TU Delft

Lab sessions & deadlines

Nov. 27	Lab 1 Kick-off
Dec. 4	Lab question session
Dec. 11	***Assignment 1 submission deadline*** *(40% grade)* Lab 2 Kick-off
Dec. 12	Lab question session
Dec. 19	Lab question session
Jan. 9	Lab question session
Jan. 21	***Assignment 2 submission deadline*** *(60% grade)*

Lab Grade = 40% Assignment 1 + 60% Assignment 2

TU Delft



TU Delft

Assignment 1 - 1 2 3

Assignment 2 - 1 4 3

Assignment 2 details

Given:

- 4 C-coded applications (common domain) as workload for each group
- Parametrizable VLIW (VEX) processor
- FPGA synthesis and functional verification tools

Goal:

- Determine VLIW uni/multi-processor architecture, optimal for all applications in terms of performance, area utilization and energy consumption.

3 Scenarios (Designs): embedded low power
high performance environment
perfomance-area-energy compromise

Method:

- Design Space Exploration (DSE) by changing the architecture configuration parameters.

TU Delft

Architecture configuration parameters

File: *configuration.rvex*

Issue width = n

VLIW instruction: 

[cores]	# of VLIW cores
CONFIG	Issue Width and # of Mult e.g., (1010) – issue width 4, 2 multipliers (1100) – issue width 4, 2 multipliers (11) – issue width 2, 2 multipliers
ICache	Instruction cache size [Bytes]
DCache	Data cache size [Bytes]
STOPBIT	The minimum number of operations encoded in a VLIW instruction. e.g.: issue width = 4; without STOPBIT ;; ALU ALU NOP NOP ;; issue width = 4; STOPBIT = 2 ;; ALU ALU ;; one bit indicating the end of the current VLIW instruction

Restrictions:

- Only slot 1 (first slot) can execute lw; sw; prefetch operations
- Only slot n (last slot) can execute branch and jump operations
- Issue width n should be a power of two

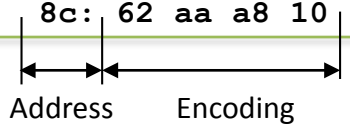
Instead of encoding 4 operations, out of which two are just mask bits (NOPs), a better way is to use a single bit at the end of the second operation indicating the end of the current instruction and thus to encode only the first 2 operations.

TU Delft

Architecture configuration parameters

STOPBIT

Instr. 1	70: 15 80 a8 00	stw
	74: 43 00 aa c0	cmpge
	78: 60 00 00 00	nop
	7c: 60 00 00 00	nop ;;
Instr. 2	80: 62 aa a8 10	add
	84: 60 00 00 00	nop
	88: 60 00 00 00	nop
	8c: 62 aa a8 10	add ;;



Issue Width = 4; Without STOPBIT

Fixed-length instruction encoding

Instr. 1	70: 15 80 a8 00	stw
	74: 43 00 aa c2	cmpge ;;
Instr. 2	78: 62 aa a8 10	add
	7c: 62 aa a8 12	add ;;
Instr. 3	80: 1a 82 66 30	sub
	84: 80 00 06 b4	limmh
	88: 60 00 00 00	nop
	8c: 22 03 db 02	call ;;

Issue Width = 4; STOPBIT = 2

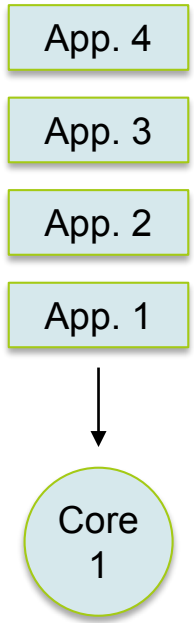
Variable-length instruction encoding

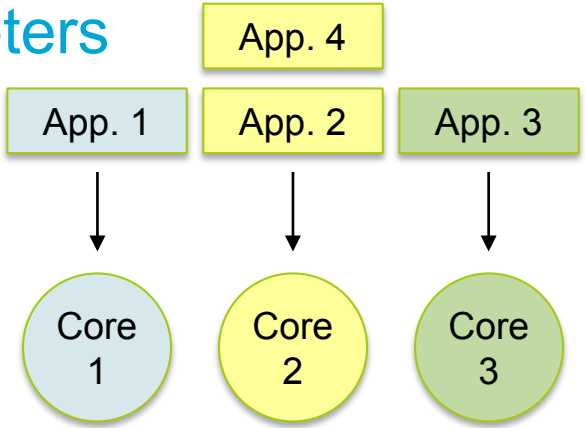
TU Delft

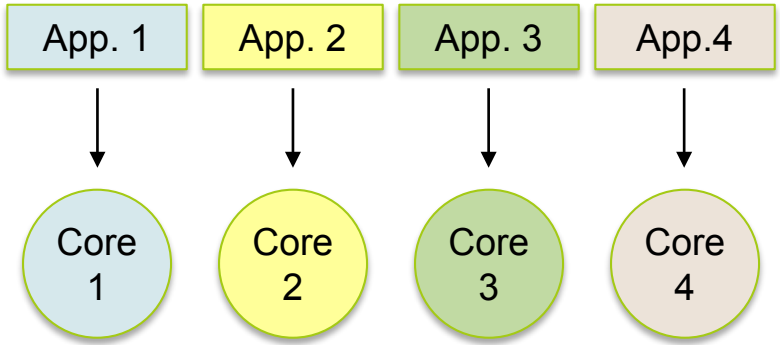
Software configuration parameters

Applications to cores mapping:

File: *config.compile; main-core*.c*







Correctness of operation

Timing errors during FPGA synthesis:

(design too complex for FPGA and cannot satisfy the targeted clock frequency)

File: *timing.txt*

```
All timing constraints were met.
```

Applications execution errors when running on the FPGA:

File: *run*-core*.log*

Performance estimation

File: *performance.txt*

```
Total cycle counts:
Run 1: 4244425 cycles
Run 2: 4244471 cycles
Run 3: 4244440 cycles
Average: 4244445 cycles
```

Useful performance statistics:

File: *run*-core*.log*

IACC	# instruction cache accesses
IMISS	# instruction cache misses
DRACC	# data cache read accesses
DRMISS	# data cache read misses
DWACC	# data cache write accesses
DWMISS	# data cache write misses

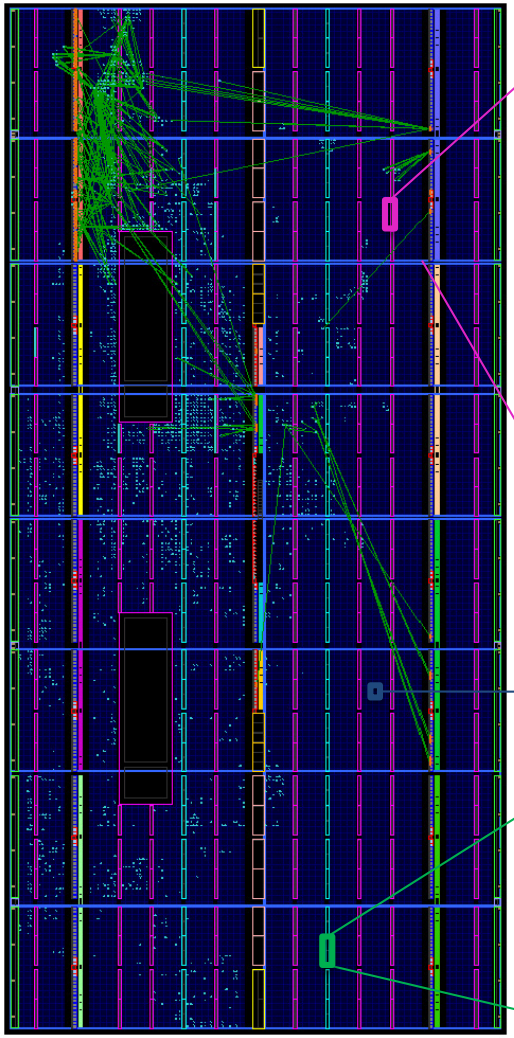
CYC	Total # of cycles
STALL	Total # of stall cycles
BUN	Total # of instructions executed
SYL	Total # of operations executed
NOP	Total # of NOPs executed

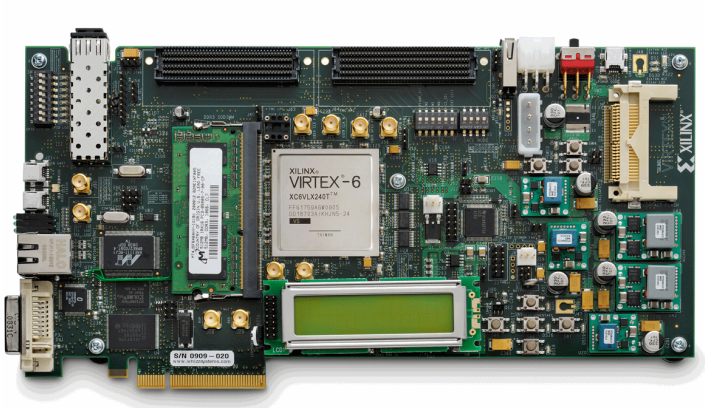
Energy estimation

File: *energy.txt*

```
Dynamic energy (idle power is subtracted from it):
Run 1: 1.67 mj
Run 2: 1.81 mj
Run 3: 1.91 mj
Average: 1.80 mJ
```

FPGA & Floorplan



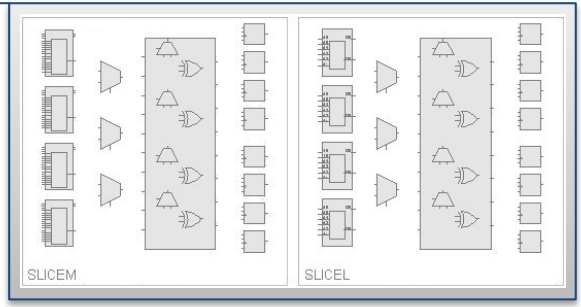


RAMB18E1

FIFO18E1

RAMB36E1

DSP48E1



One Configurable Logic Block (CLB)

Slices	2
LUTs	8
Flip-Flops	16
Arithmetic & Carry Chains	2
Distributed RAM	256
Shift Registers	128

Resources in one CLB

Area estimation

File: *area.txt*

Area estimates for individual FPGA components

A_{1_Slice}	$0.5 \cdot A_{1\_CLB}$
$A_{1\_RAMB36E1}$	$2.4 \cdot A_{1\_CLB}$
$A_{1\_RAMB18E1}$	$1.2 \cdot A_{1\_CLB}$
$A_{1\_DSP48E1}$	$0.7 \cdot A_{1\_CLB}$

Total Area for Assignment 2:

$$A \cong (\#\ \text{occupied slices}) * A_{1\_Slice} + (\#\ \text{RAMB36E1}) * A_{1\_RAMB36E1} + (\#\ \text{RAMB18E1}) * A_{1\_RAMB18E1} + (\#\ \text{DSP48E1}) * A_{1\_DSP48E1}$$

Area units: A_{1_CLB}

Report

- Max. 5-page
- Method of searching the design space.
- Justify configuration choices behind the 3 designs
  - High performance
  - Embedded low power
  - Energy-area-performance compromise
- Discuss which design of the three is best suited for the 4 applications domain and why.
- Follow requirements from course guide pdf.

Submission requirements

- Max. 5-page report pdf
- Archive generated by *make pack* command
- Submit by e-mail

- Naming convention:
  - ET4074_2018_A2_report_group#.pdf
  - ET4074_2018_A2_src_group#.tgz