Paris-Saclay University

A2 : Embedded Electronic Systems

Heterogeneous architecture for Big Data algorithms
Final presentation

Presented by:
DOU Yuhan
FORCIOLI Quentin
GHAOUI Mohamed Anis
TERRACHER Audrey

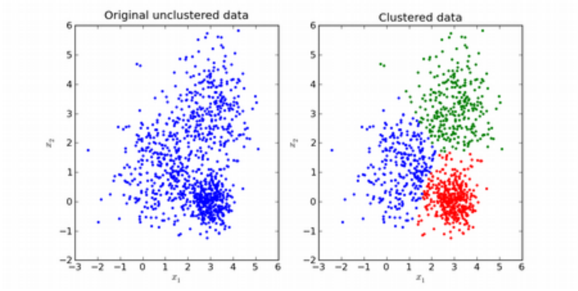
2020 February 03

TEAM

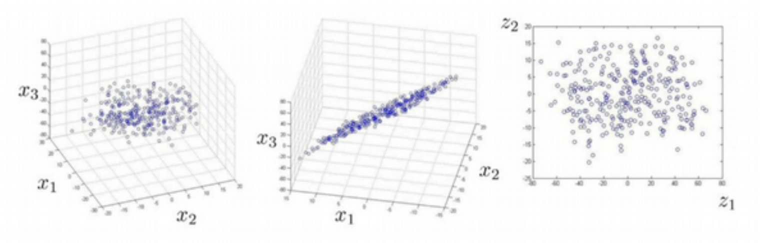
- **Software:**
  - DOU Yuhan
  - TERRACHER Audrey
- **Hardware:**
  - FORCIOLI QUENTIN
- **HLS and lead :**
  - GHAOUI Mohamed Anis

Objective : Complete a full development cycle

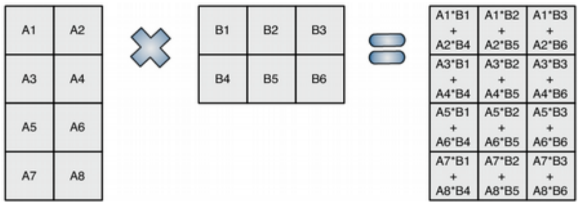
Our approach and algorithms chosen



K-Means



PCA/SVD

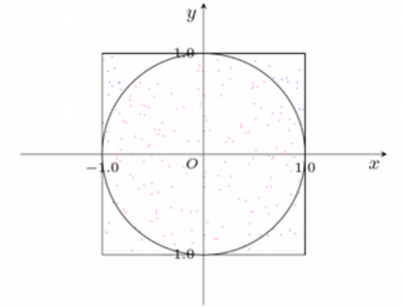


A1	A2
A3	A4
A5	A6
A7	A8

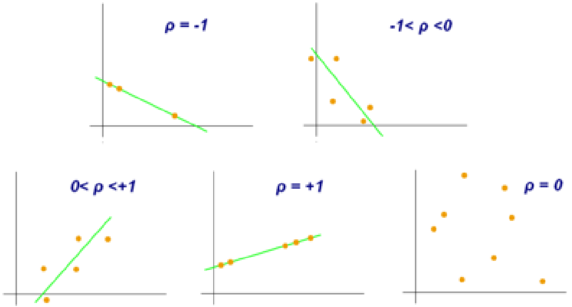
B1	B2	B3
B4	B5	B6

A1*B1 + A2*B4	A1*B2 + A2*B5	A1*B3 + A2*B6
A3*B1 + A4*B4	A3*B2 + A4*B5	A3*B3 + A4*B6
A5*B1 + A6*B4	A5*B2 + A6*B5	A5*B3 + A6*B6
A7*B1 + A8*B4	A7*B2 + A8*B5	A7*B3 + A8*B6

BMM

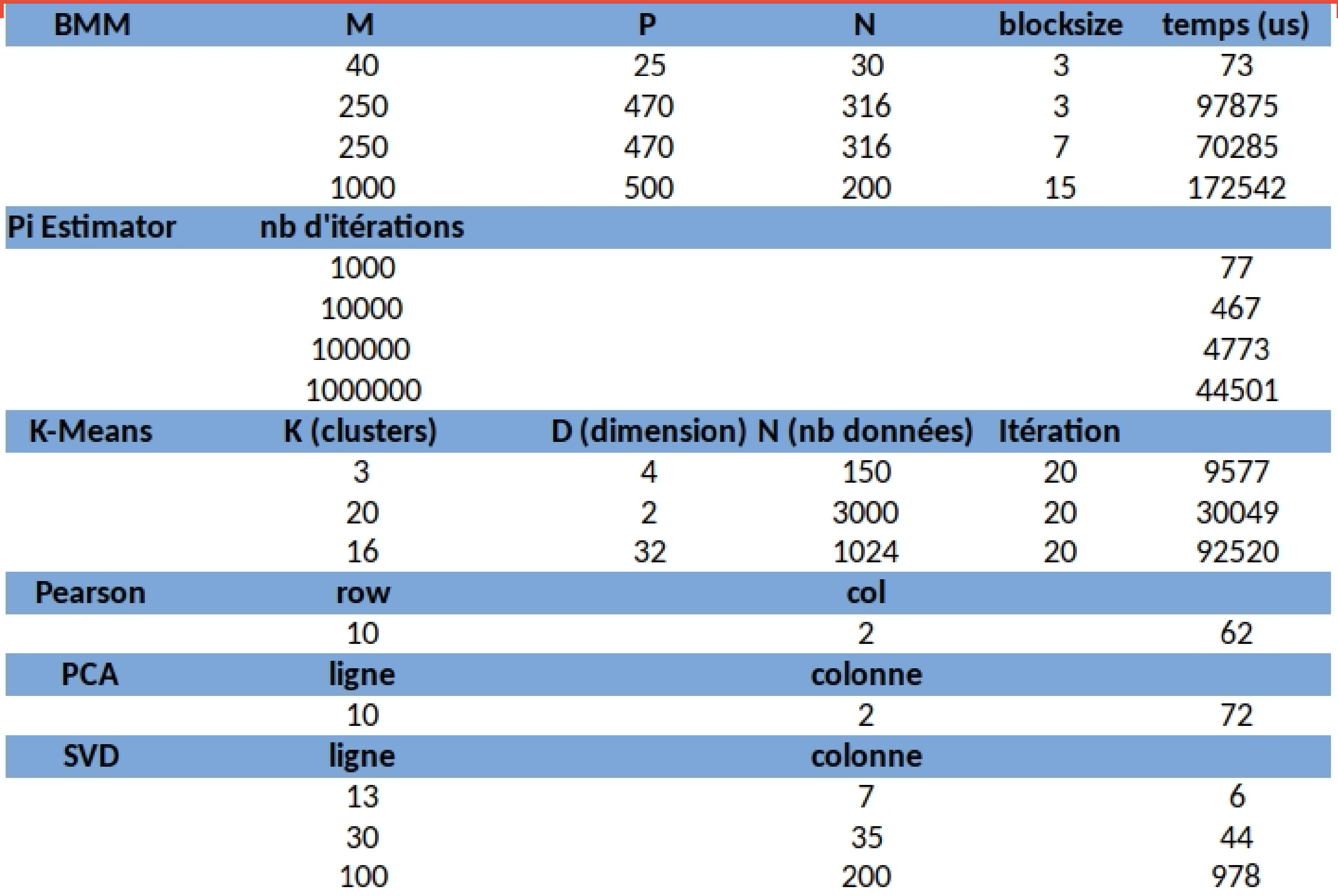


Pi Estimator



Pearson correlation

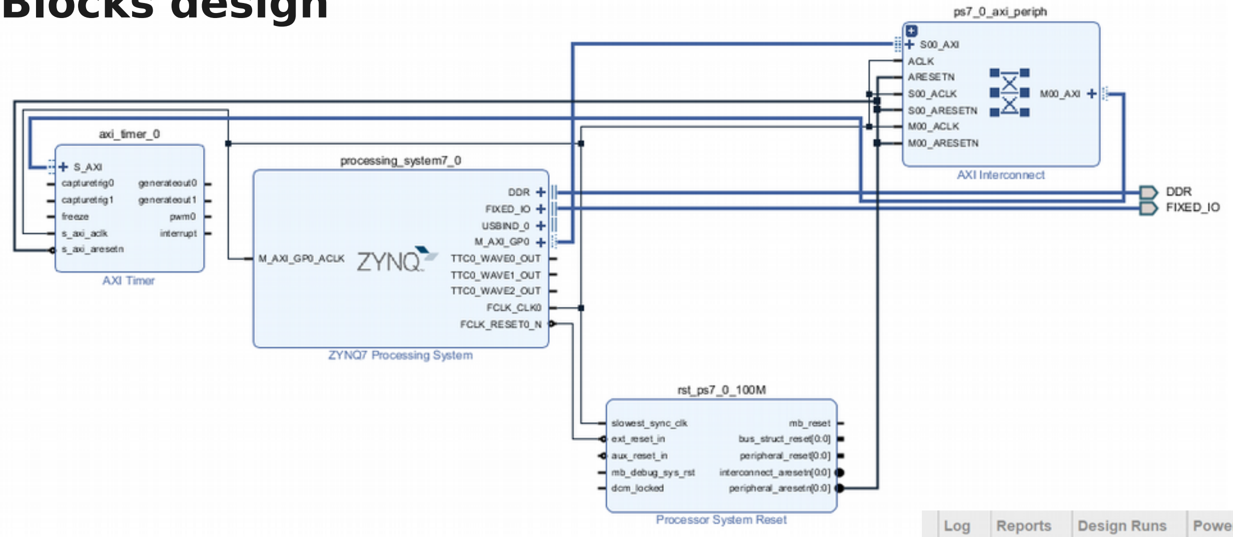
Software – PC Implementation

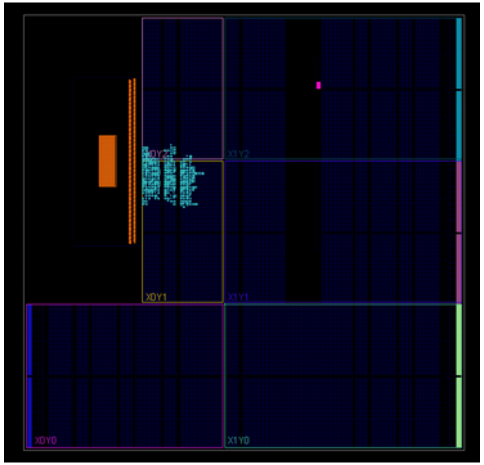


BMM	M	P	N	blocksize	temps (us)
	40	25	30	3	73
	250	470	316	3	97875
	250	470	316	7	70285
	1000	500	200	15	172542
Pi Estimator	**nb d'itérations**				
	1000				77
	10000				467
	100000				4773
	1000000				44501
K-Means	**K (clusters)**	**D (dimension)**	**N (nb données)**	**Itération**	
	3	4	150	20	9577
	20	2	3000	20	30049
	16	32	1024	20	92520
Pearson	**row**		**col**		
	10		2		62
PCA	**ligne**		**colonne**		
	10		2		72
SVD	**ligne**		**colonne**		
	13		7		6
	30		35		44
	100		200		978

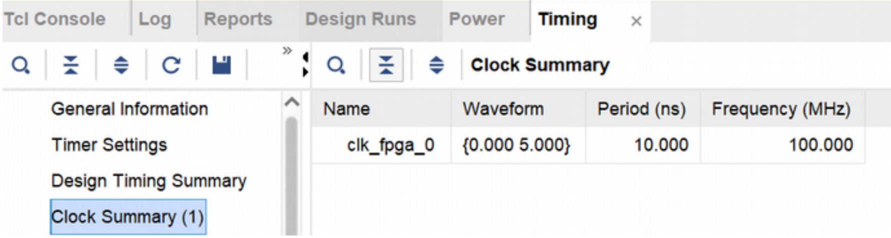
Software – ARM Implementation

Blocks design



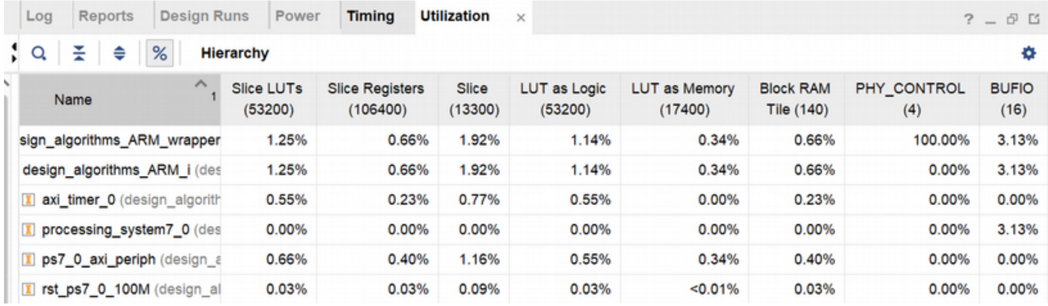


Report



Clock Summary

Name	Waveform	Period (ns)	Frequency (MHz)
clk_fpga_0	{0.000 5.000}	10.000	100.000



Hierarchy

Name	Slice LUTs (53200)	Slice Registers (106400)	Slice (13300)	LUT as Logic (53200)	LUT as Memory (17400)	Block RAM Tile (140)	PHY_CONTROL (4)	BUFIO (16)
sign_algorithms_ARM_wrapper	1.25%	0.66%	1.92%	1.14%	0.34%	0.66%	100.00%	3.13%
design_algorithms_ARM_i (des	1.25%	0.66%	1.92%	1.14%	0.34%	0.66%	0.00%	3.13%
axi_timer_0 (design_algorith	0.55%	0.23%	0.77%	0.55%	0.00%	0.23%	0.00%	0.00%
processing_system7_0 (des	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	3.13%
ps7_0_axi_periph (design_a	0.66%	0.40%	1.16%	0.55%	0.34%	0.40%	0.00%	0.00%
rst_ps7_0_100M (design_al	0.03%	0.03%	0.09%	0.03%	<0.01%	0.03%	0.00%	0.00%

Software – ARM Implementation

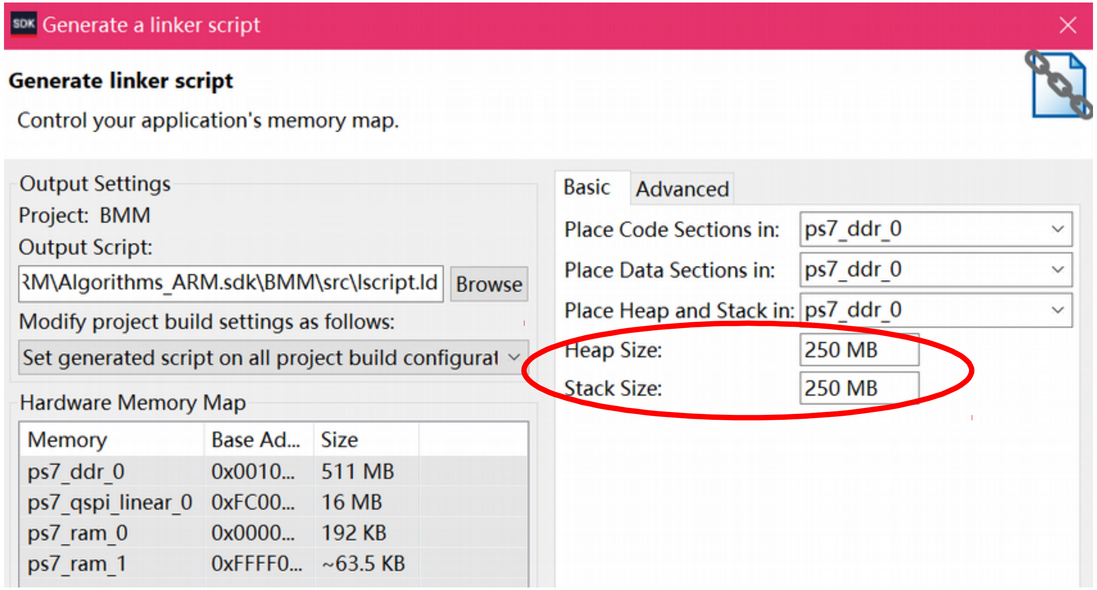
Design on Vivado SDK

1/ Tools Chain : Xilinx ARM v7 GNU Toolchain

Generator : GNU make

2/ Modification of heap and stack sizes :

1 KB (default) → 250 MB



Generate a linker script

Generate linker script

Control your application's memory map.

Output Settings

Project: BMM

Output Script:

RM\Algorithms_ARM.sdk\BMM\src\lscript.ld

Modify project build settings as follows:

Set generated script on all project build configurat

Hardware Memory Map

Memory	Base Ad...	Size
ps7_ddr_0	0x0010...	511 MB
ps7_qspi_linear_0	0xFC00...	16 MB
ps7_ram_0	0x0000...	192 KB
ps7_ram_1	0xFFFF0...	~63.5 KB

Place Code Sections in: ps7_ddr_0

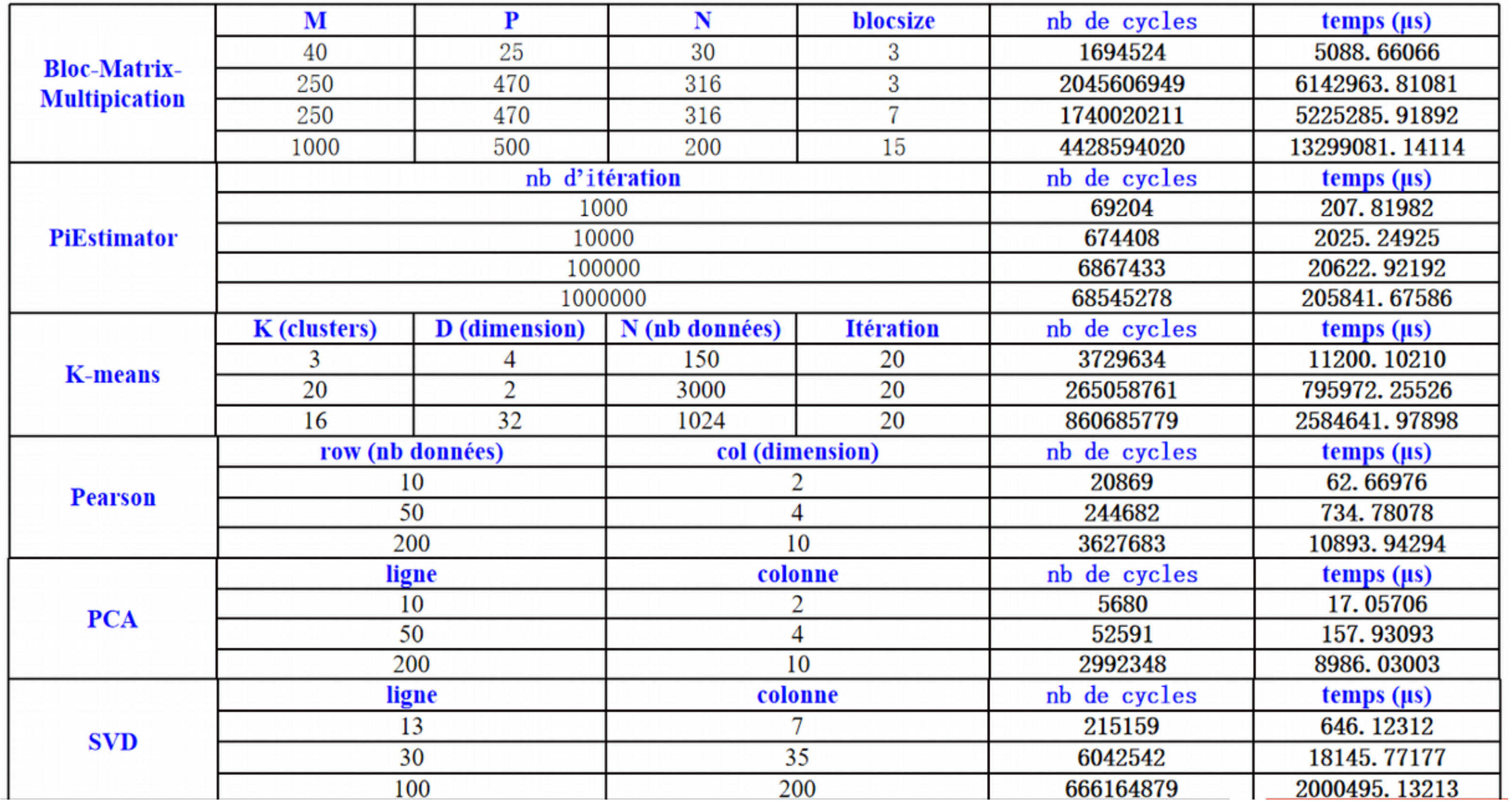
Place Data Sections in: ps7_ddr_0

Place Heap and Stack in: ps7_ddr_0

Heap Size: 250 MB

Stack Size: 250 MB

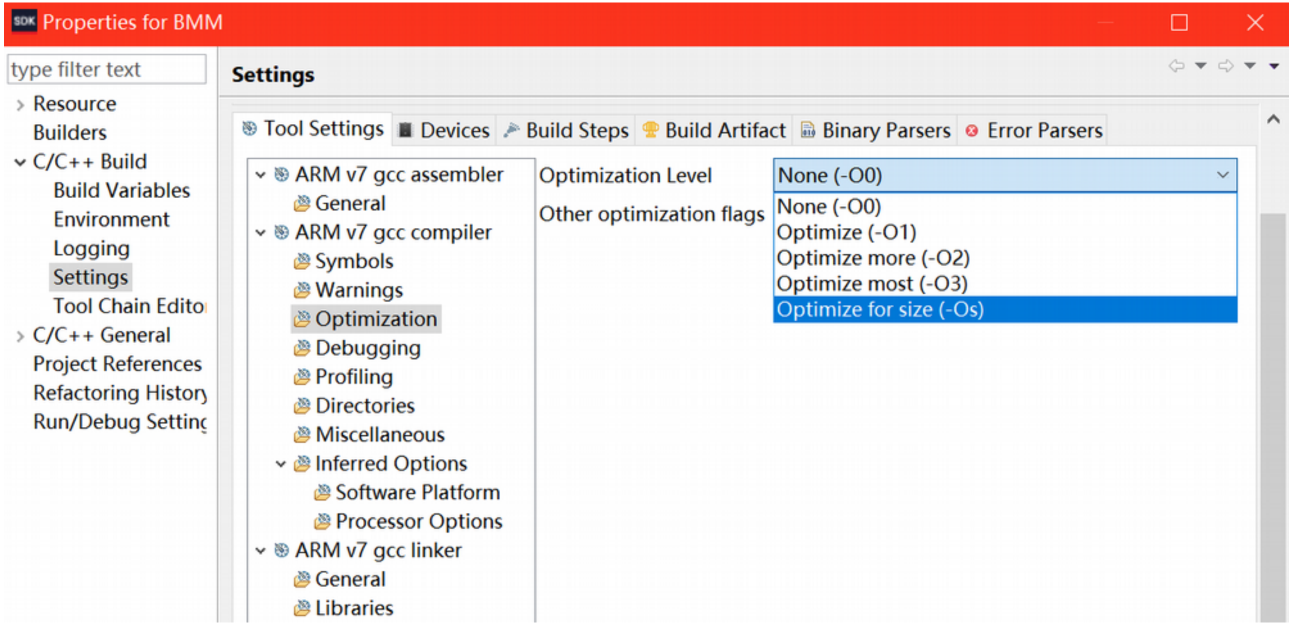
Software – ARM Implementation



	M	P	N	blocsize	nb de cycles	temps (µs)
Bloc-Matrix-Multipication	40	25	30	3	1694524	5088. 66066
	250	470	316	3	2045606949	6142963. 81081
	250	470	316	7	1740020211	5225285. 91892
	1000	500	200	15	4428594020	13299081. 14114
PiEstimator	nb d'itération				nb de cycles	temps (µs)
	1000				69204	207. 81982
	10000				674408	2025. 24925
	100000				6867433	20622. 92192
	1000000				68545278	205841. 67586
K-means	K (clusters)	D (dimension)	N (nb données)	Itération	nb de cycles	temps (µs)
	3	4	150	20	3729634	11200. 10210
	20	2	3000	20	265058761	795972. 25526
	16	32	1024	20	860685779	2584641. 97898
Pearson	row (nb données)		col (dimension)		nb de cycles	temps (µs)
	10		2		20869	62. 66976
	50		4		244682	734. 78078
	200		10		3627683	10893. 94294
PCA	ligne		colonne		nb de cycles	temps (µs)
	10		2		5680	17. 05706
	50		4		52591	157. 93093
	200		10		2992348	8986. 03003
SVD	ligne		colonne		nb de cycles	temps (µs)
	13		7		215159	646. 12312
	30		35		6042542	18145. 77177
	100		200		666164879	2000495. 13213

Software – ARM Implementation

Software optimization



Optimization Level: None (-O0)

Other optimization flags

Three main aspects :

1/ branches delays

2/ use of cache

3/ data dependencies

Optimization options for GNU :

- O0 : no optimization

- O1 : optimization on branches

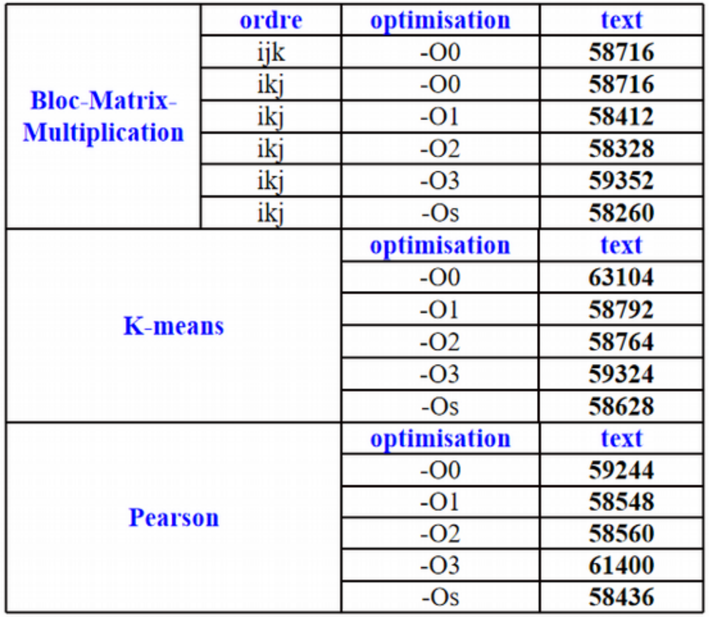
- O2 : optimization at the registers and instruction level

- O3 : optimization at the highest level (eSIMD vectorization , “inline”...)

- Os : optimization on code size

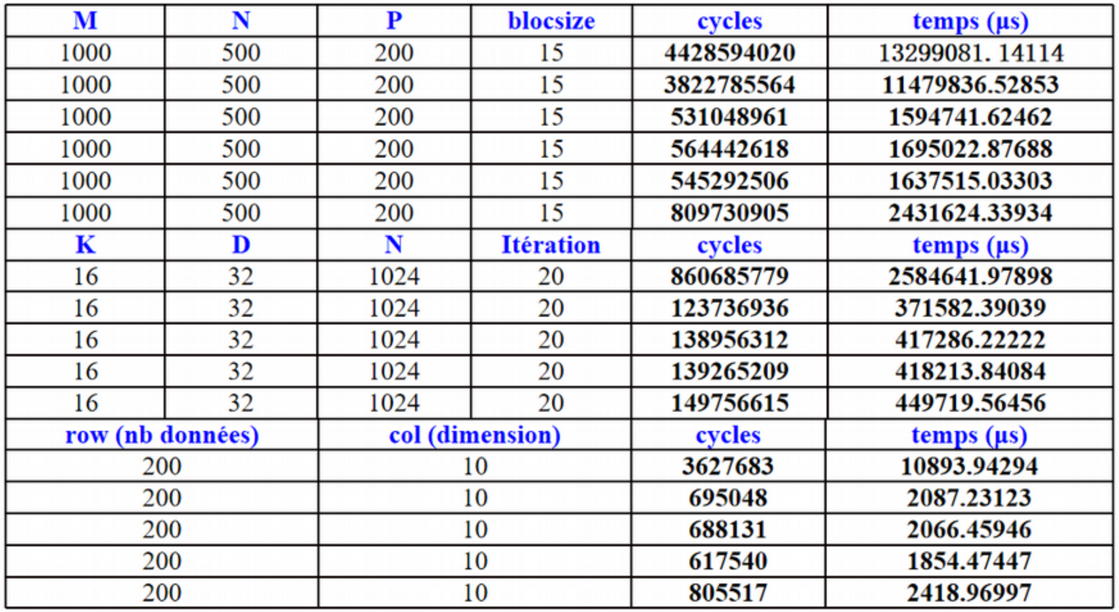
Software – ARM Implementation

Code size (segmentation “text”)



	ordre	optimisation	text
Bloc-Matrix-Multiplication	ijk	-O0	58716
	ikj	-O0	58716
	ikj	-O1	58412
	ikj	-O2	58328
	ikj	-O3	59352
	ikj	-Os	58260
K-means		optimisation	text
		-O0	63104
		-O1	58792
		-O2	58764
		-O3	59324
		-Os	58628
Pearson		optimisation	text
		-O0	59244
		-O1	58548
		-O2	58560
		-O3	61400
		-Os	58436

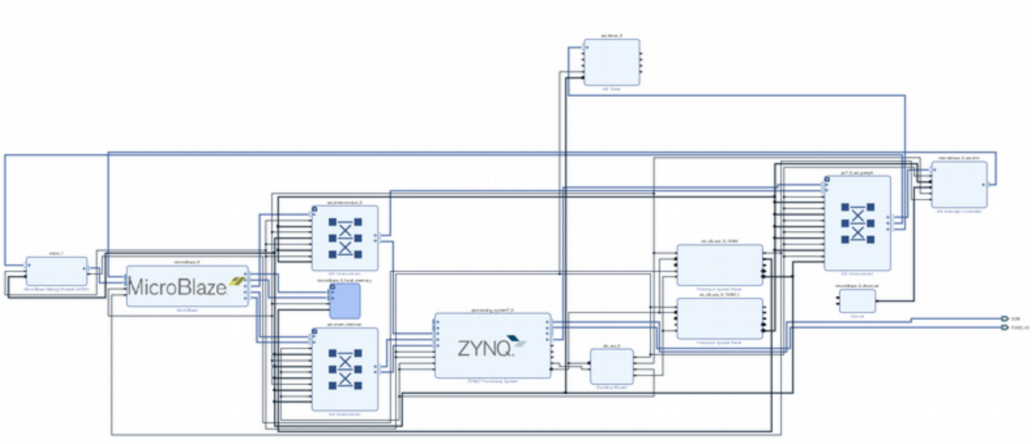
Execution times

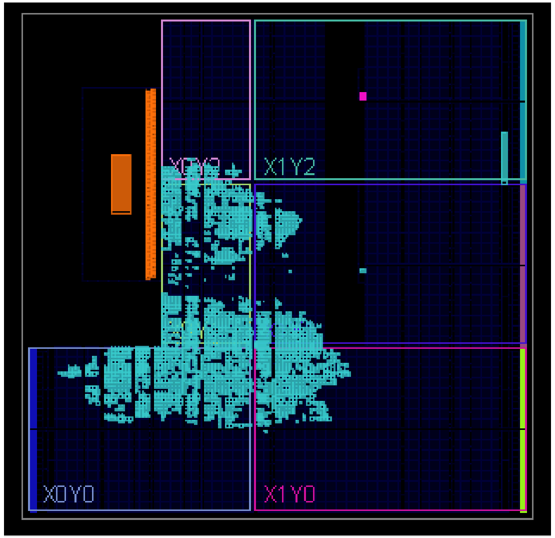


M	N	P	blocsize	cycles	temps (µs)
1000	500	200	15	4428594020	13299081. 14114
1000	500	200	15	3822785564	11479836.52853
1000	500	200	15	531048961	1594741.62462
1000	500	200	15	564442618	1695022.87688
1000	500	200	15	545292506	1637515.03303
1000	500	200	15	809730905	2431624.33934
K	D	N	Itération	cycles	temps (µs)
16	32	1024	20	860685779	2584641.97898
16	32	1024	20	123736936	371582.39039
16	32	1024	20	138956312	417286.22222
16	32	1024	20	139265209	418213.84084
16	32	1024	20	149756615	449719.56456
row (nb données)		col (dimension)		cycles	temps (µs)
200		10		3627683	10893.94294
200		10		695048	2087.23123
200		10		688131	2066.45946
200		10		617540	1854.47447
200		10		805517	2418.96997

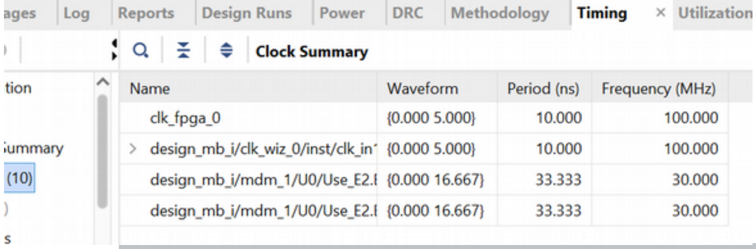
Software – Microblaze Implementation

Blocks design

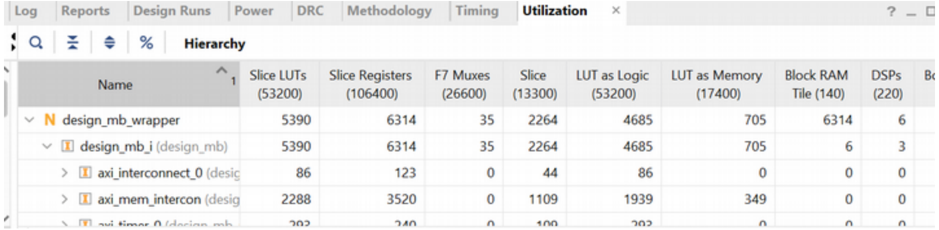




Report


ages | Log | Reports | Design Runs | Power | DRC | Methodology | Timing × | Utilization
Clock Summary
tion
ummary
(10)
)
s

Name	Waveform	Period (ns)	Frequency (MHz)
clk_fpga_0	{0.000 5.000}	10.000	100.000
> design_mb_i/clk_wiz_0/inst/clk_in1	{0.000 5.000}	10.000	100.000
design_mb_i/mdm_1/U0/Use_E2.E	{0.000 16.667}	33.333	30.000
design_mb_i/mdm_1/U0/Use_E2.E	{0.000 16.667}	33.333	30.000


Log | Reports | Design Runs | Power | DRC | Methodology | Timing | Utilization ×
Hierarchy

Name	Slice LUTs (53200)	Slice Registers (106400)	F7 Muxes (26600)	Slice (13300)	LUT as Logic (53200)	LUT as Memory (17400)	Block RAM Tile (140)	DSPs (220)	Bo
design_mb_wrapper	5390	6314	35	2264	4685	705	6314	6	
design_mb_i (design_mb)	5390	6314	35	2264	4685	705	6	3	
axi_interconnect_0 (desic	86	123	0	44	86	0	0	0	
axi_mem_intercon (desig	2288	3520	0	1109	1939	349	0	0	
axi_timer_0 (design_mb	293	240	0	100	293	0	0	0	

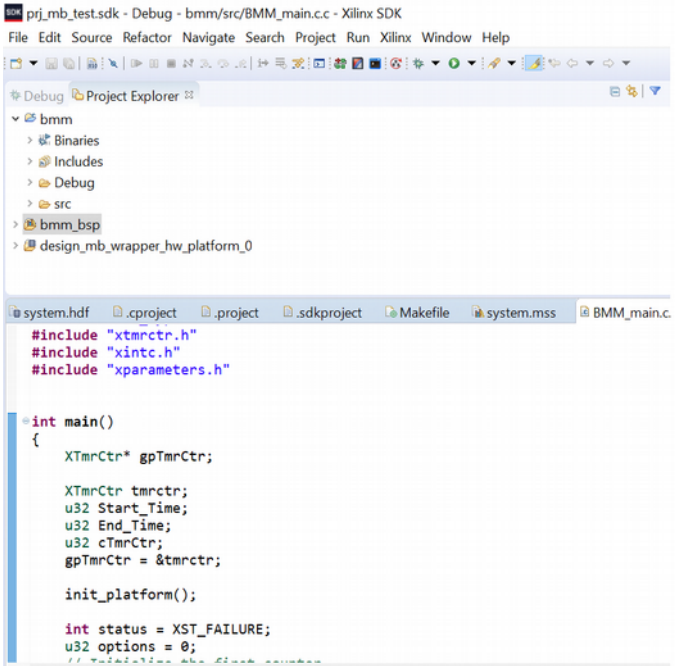
Software – Microblaze Implementation

Design on Vivado SDK

1/ Changes in libraries and functions used for timing :

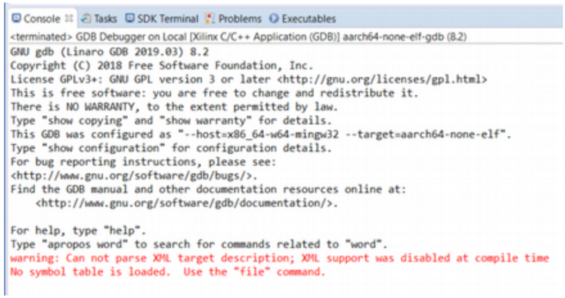
xtime_l.h for ARM Implementation

xtmrctr.h for Microblaze Implementation

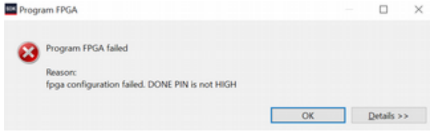

prj_mb_test.sdk - Debug - bmm/src/BMM_main.c.c - Xilinx SDK
File Edit Source Refactor Navigate Search Project Run Xilinx Window Help
Debug
Project Explorer
bmm
Binaries
Includes
Debug
src
bmm_bsp
design_mb_wrapper_hw_platform_0
system.hdf
.cproject
.project
.sdkproject
Makefile
system.mss
BMM_main.c

```
#include "xtmrctr.h"
#include "xintc.h"
#include "xparameters.h"


int main()
{
    XTmrCtr* gpTmrCtr;

    XTmrCtr tmrctr;
    u32 Start_Time;
    u32 End_Time;
    u32 cTmrCtr;
    gpTmrCtr = &tmrctr;

    init_platform();

    int status = XST_FAILURE;
    u32 options = 0;
    // Initialize the first counter
```

2/ Several bugs launching FPGA programming..

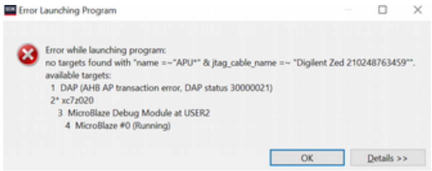


```
<terminated> GDB Debugger on Local [Xilinx C/C++ Application (GDB)] aarch64-none-elf-gdb (8.2)
GNU gdb (Linaro GDB 2019.03) 8.2
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "--host=x86_64-w64-mingw32 --target=aarch64-none-elf".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word".
warning: Can not parse XML target description; XML support was disabled at compile time
No symbol table is loaded.  Use the "file" command.
```



Program FPGA failed

Reason:
fpga configuration failed. DONE PIN is not HIGH



```
Error while launching program:
no targets found with "name =~"APU*" & jtag_cable_name =~ "Digilent Zed 210248763459"".
available targets:
  1  DAP (AHB AP transaction error, DAP status 30000021)
  2* xc7z020
    3  MicroBlaze Debug Module at USER2
      4  MicroBlaze #0 (Running)
```

High – Level Synthesis

1) Reanalyse the algorithms

2) Directive usage to infer RTL blocks

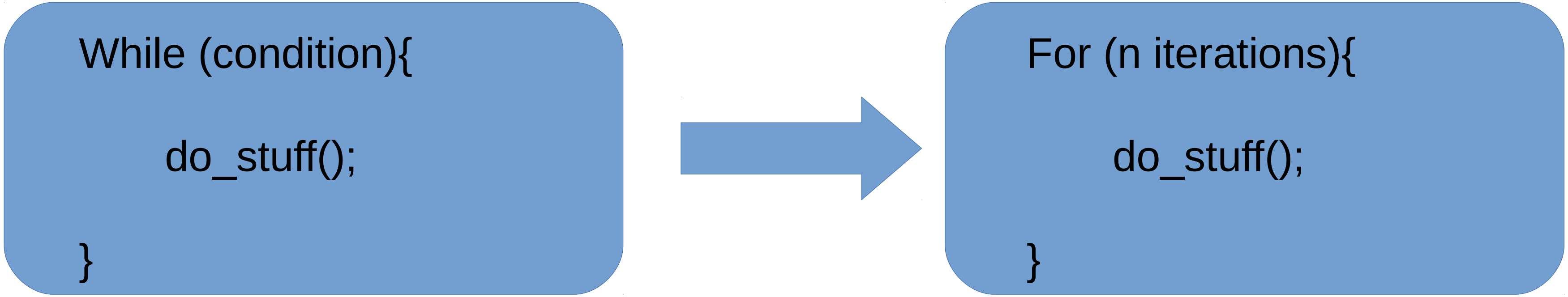
3) Implement Axi/Axilite interfaces

4) Manage and reshape memory

5) Automated material optimisations

High – Level Synthesis

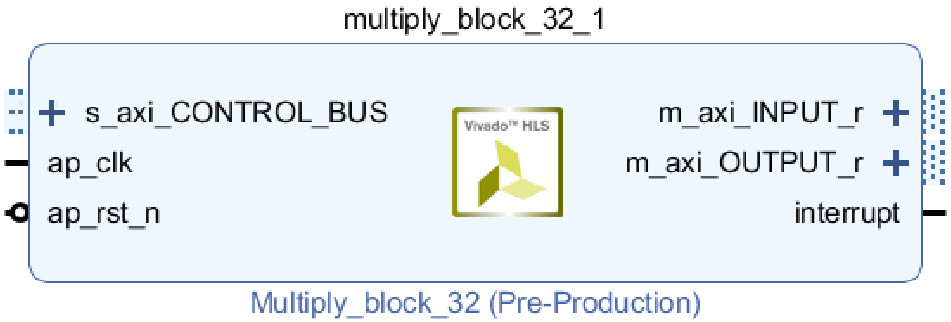
Control sequence conversion :



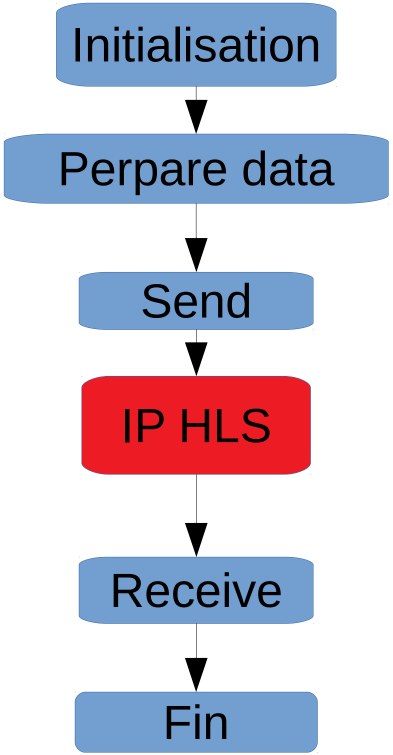
```
While (condition){

        do_stuff();

}
```

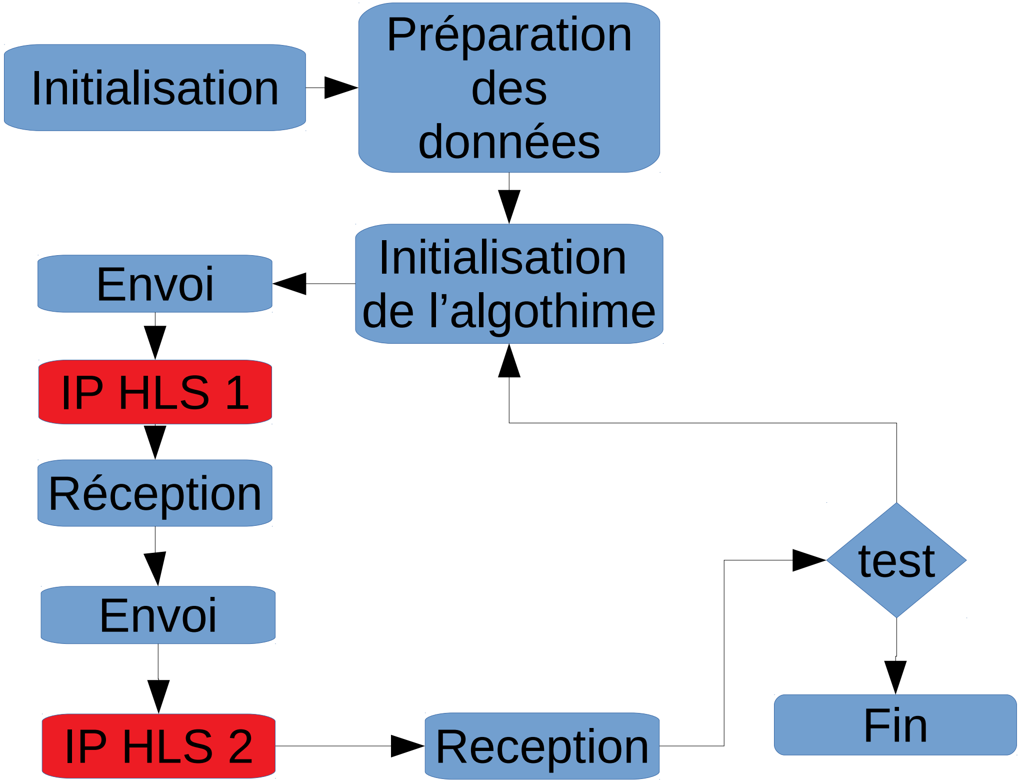
```
For (n iterations){

        do_stuff();

}
```

- **AXI/lite Interface directives :**
- Top-Level function : BUS CONTROL
- Inputs
- Outputs
- Addresses managed by the CPU



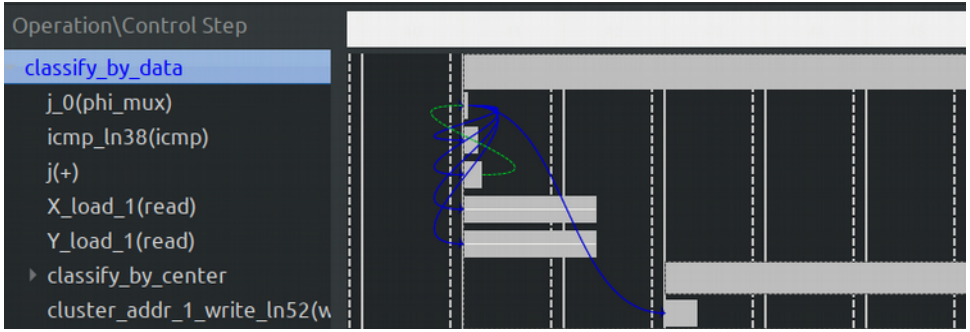
High – Level Synthesis

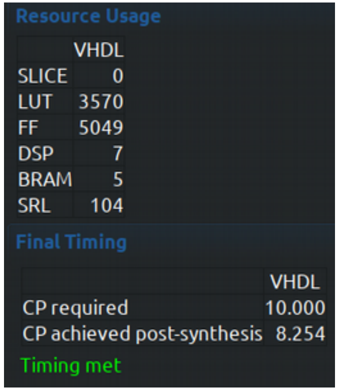




High – Level Synthesis : scheduler

- Analyse dependencies
- Minimise Initialisation interval
- Respect timing and FPGA resources limits
- If necessary : reshape the arrays



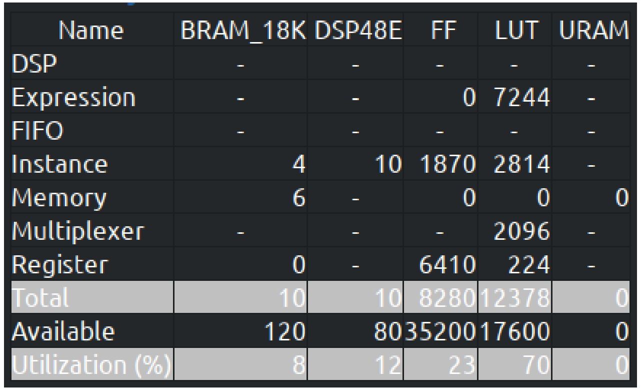

Resource Usage

	VHDL
SLICE	0
LUT	3570
FF	5049
DSP	7
BRAM	5
SRL	104

Final Timing

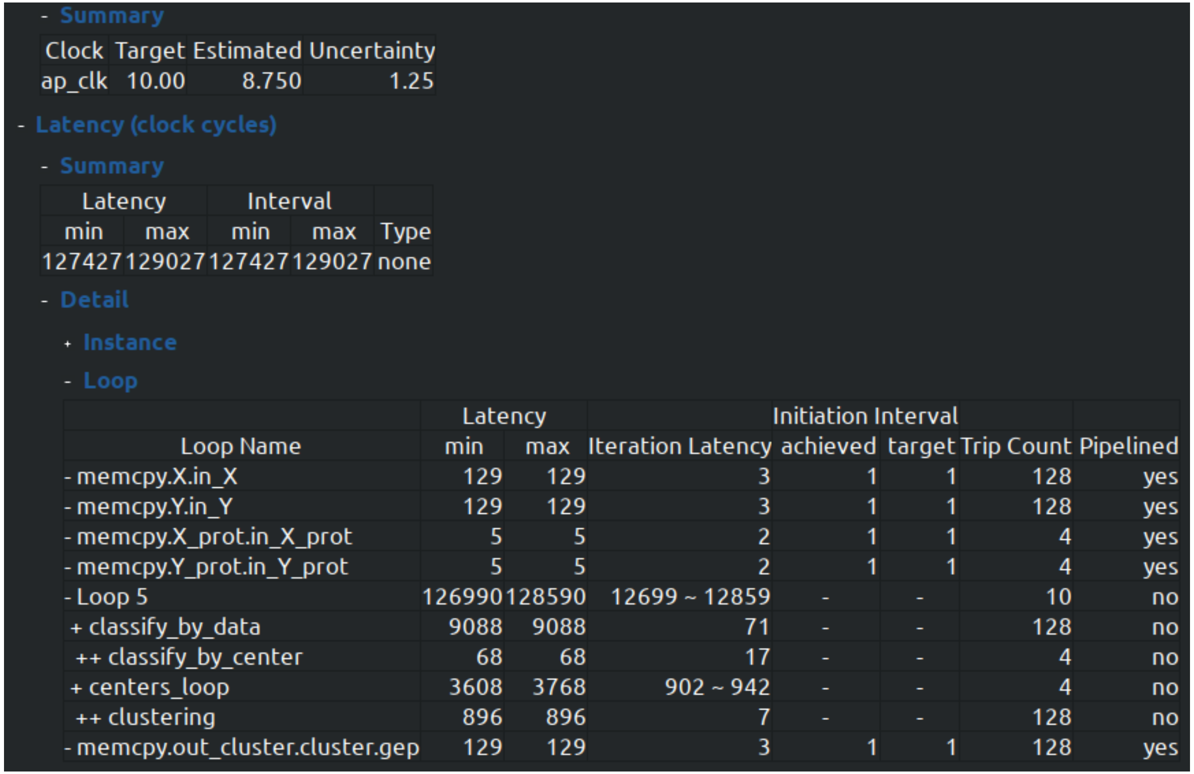
	VHDL
CP required	10.000
CP achieved post-synthesis	8.254

Timing met



Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	-	0	7244	-
FIFO	-	-	-	-	-
Instance	4	10	1870	2814	-
Memory	6	-	0	0	0
Multiplexer	-	-	-	2096	-
Register	0	-	6410	224	-
Total	10	10	8280	12378	0
Available	120	80	35200	17600	0
Utilization (%)	8	12	23	70	0

High – Level Synthesis : Synthesis



- Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	8.750	1.25

- Latency (clock cycles)

- Summary

Latency		Interval		
min	max	min	max	Type
127427	129027	127427	129027	none

- Detail

+ Instance

- Loop

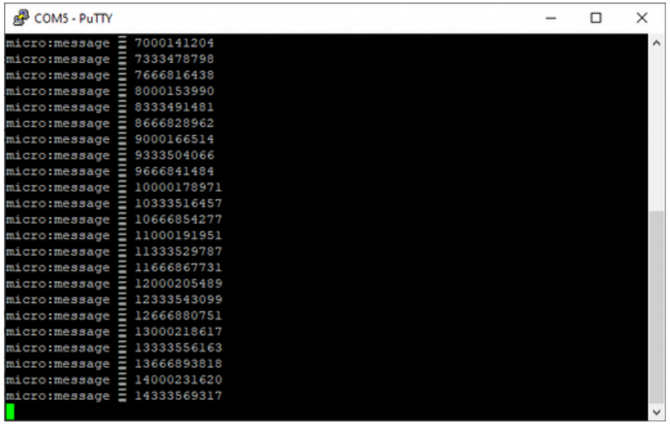
	Latency			Initiation Interval			
Loop Name	min	max	Iteration Latency	achieved	target	Trip Count	Pipelined
- memcpy.X.in_X	129	129	3	1	1	128	yes
- memcpy.Y.in_Y	129	129	3	1	1	128	yes
- memcpy.X_prot.in_X_prot	5	5	2	1	1	4	yes
- memcpy.Y_prot.in_Y_prot	5	5	2	1	1	4	yes
- Loop 5	126990	128590	12699 ~ 12859	-	-	10	no
+ classify_by_data	9088	9088	71	-	-	128	no
++ classify_by_center	68	68	17	-	-	4	no
+ centers_loop	3608	3768	902 ~ 942	-	-	4	no
++ clustering	896	896	7	-	-	128	no
- memcpy.out_cluster.cluster.gep	129	129	3	1	1	128	yes

- Not always necessary/ possible to pipeline all the loops
- HLS must respect time constraints
- Optimisation can be very time/resource costly

Architecture exploration

- First design used by the SW team
- 2nd design using BRAM
- 3rd design with BRAM and HLS IP

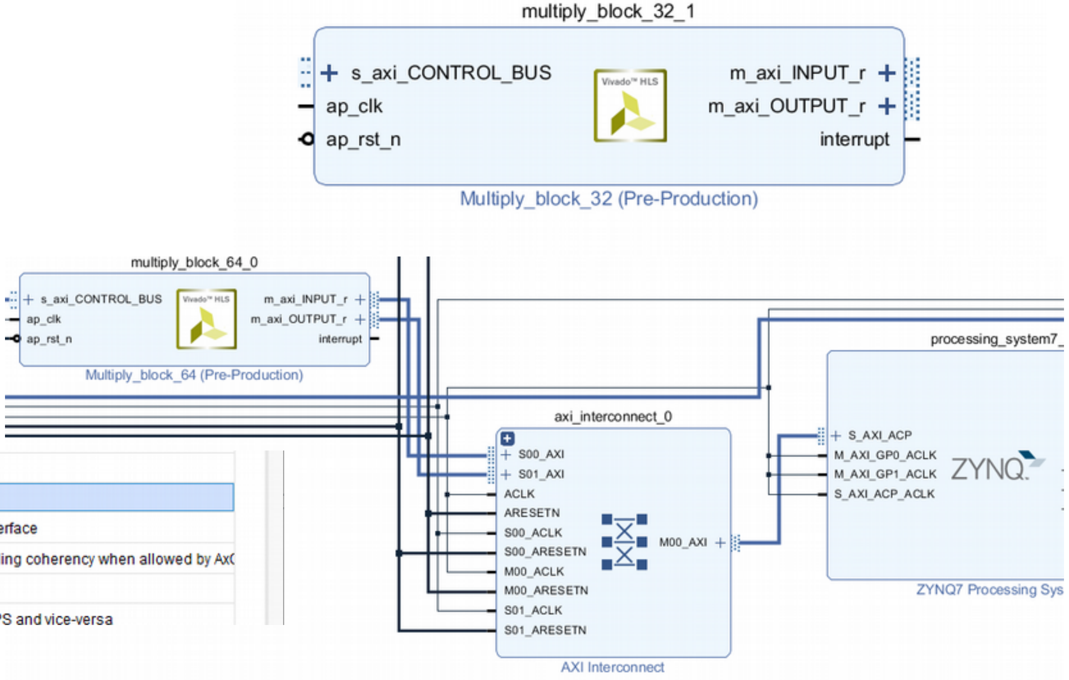
  (still in development)

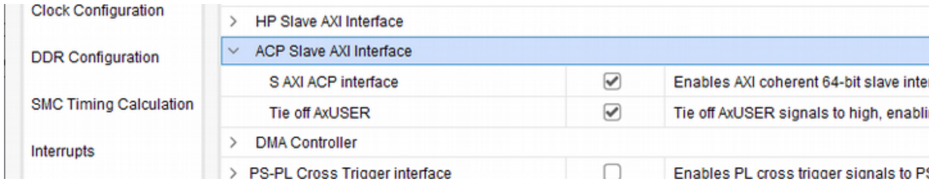


```
micro:message ≡ 7000141204
micro:message ≡ 7333478798
micro:message ≡ 7666816438
micro:message ≡ 8000153990
micro:message ≡ 8333491481
micro:message ≡ 8666828962
micro:message ≡ 9000166514
micro:message ≡ 9333504066
micro:message ≡ 9666841484
micro:message ≡ 10000178971
micro:message ≡ 10333516457
micro:message ≡ 10666854277
micro:message ≡ 11000191951
micro:message ≡ 11333529787
micro:message ≡ 11666867731
micro:message ≡ 12000205489
micro:message ≡ 12333543099
micro:message ≡ 12666880751
micro:message ≡ 13000218617
micro:message ≡ 13333556163
micro:message ≡ 13666893818
micro:message ≡ 14000231620
micro:message ≡ 14333569317
```

Interfacing Zynq to HLS IP

- Using ACP port => Cache coherency
- Having different clock domains





Clock Configuration

DDR Configuration

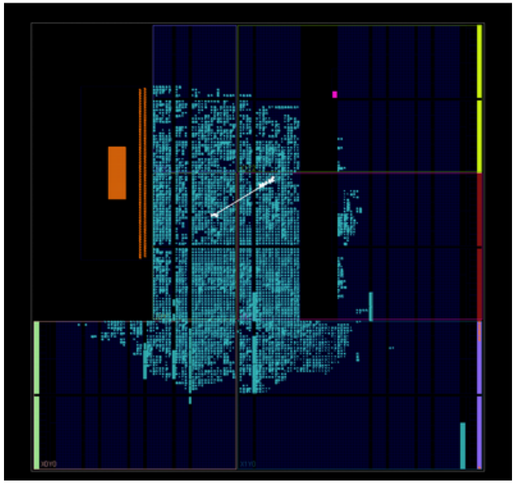
SMC Timing Calculation

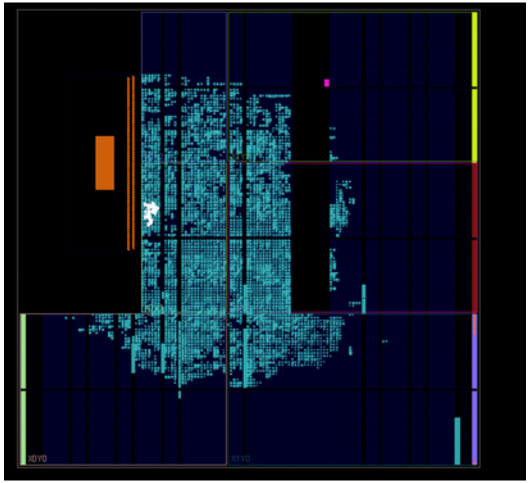
Interrupts

> HP Slave AXI Interface		
⌄ ACP Slave AXI Interface		
S AXI ACP interface	☑	Enables AXI coherent 64-bit slave interface
Tie off AxUSER	☑	Tie off AxUSER signals to high, enabling coherency when allowed by AxC
> DMA Controller		
> PS-PL Cross Trigger interface	□	Enables PL cross trigger signals to PS and vice-versa

Improving clock with timing

- Using Timing report to increase HLS clock frequency
- Using more performance-oriented placement and synthesis strategies





Performance

- First time @ 100MHZ
- Test on MUL64:
  - AXI@180MHZ
  - HLS@131MHZ
  - Acceleration : **320%**

IP	HW time (µs)	SW time (µs)	Improved clock	Improve HW times (µs)	Best acceleration
mul64	10280	26132	130MHZ	7042	**271%**
mul32	2412	3282	125 MHZ	1810	81%
pearson	32	5	115 MHZ	21	**-74%**
kmeans	1297	1527	120MHZ	1082	41%

Future improvement

- Using interruption instead of polling
- A microblaze handle every task involving IP
- Multi-microblazes multi-IPs.

Conclusion

- 6 Algorithms implemented and tested
- 3 IP conceived and implemented
- Multi-clock periods for a heterogeneous functionning
- Development method is satisfying with room for improuvement
- Set up the usage of automated development tools
- Develop multi-microblazes multi-Ips
- Maximise board usages

- **Completed a full development cycle : objective achieved**

Conclusion

Thank you for you attention