

# OmpSs porting: SMP and FPGA hints

## 1.1 Execution without acceleration: Timing

2) Time: 0.1484 s

```
> ^C
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.148336
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.148476
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.148370
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.148704
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.148374
ubuntu@zynq-bsc:~/lab4-acc$ █
```

3) Time: 0.7445 s

```
ubuntu@zynq-bsc:~/lab4-acc$ 
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.744756
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.744882
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.745072
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.744766
ubuntu@zynq-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.744927
ubuntu@zynq-bsc:~/lab4-acc$ █
```

## 2.1 OmpSs@FPGA porting

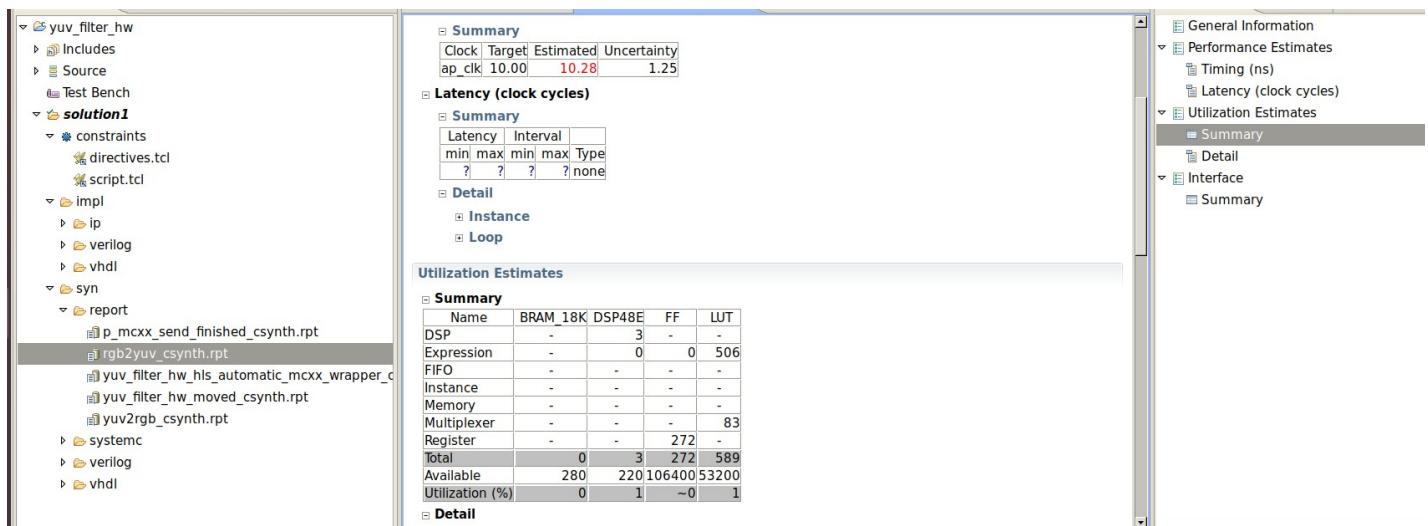
He probado de hacer lo anterior como prueba y hay una mejora en el tiempo de ejecución para los dos casos

Time 20 images: 0.08126 s SpeedUp = 1.8262

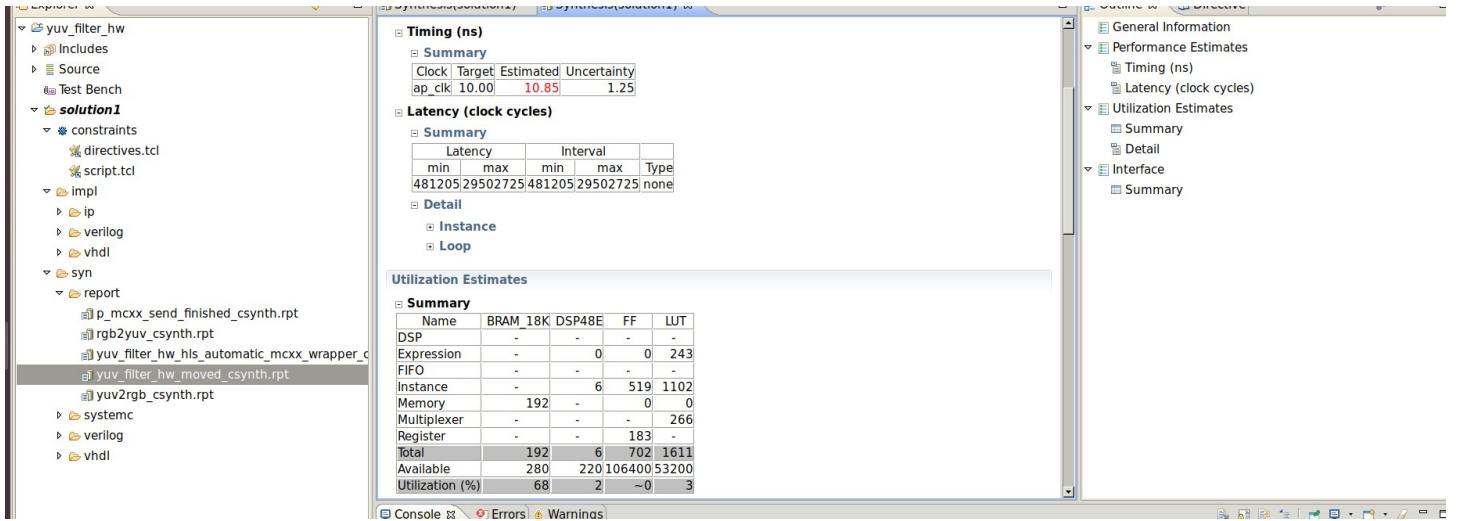
Time 100 images: 0.3964 s SpeedUp = 1.8782

```
extrae.xml Makefile smp_yuv_filter test_data yuv_filter.c yuv_filter.h
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.396690
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.396269
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.396154
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.395855
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 100
Test passed!
  Execution time (secs): 0.400536
ubuntu@zynd-bsc:~/lab4-acc$
ubuntu@zynd-bsc:~/lab4-acc$
ubuntu@zynd-bsc:~/lab4-acc$
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.081532
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.081280
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.080904
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.081072
ubuntu@zynd-bsc:~/lab4-acc$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
  Execution time (secs): 0.081420
ubuntu@zynd-bsc:~/lab4-acc$ 
```

## 2.2 HLS Compilation and Latency Information

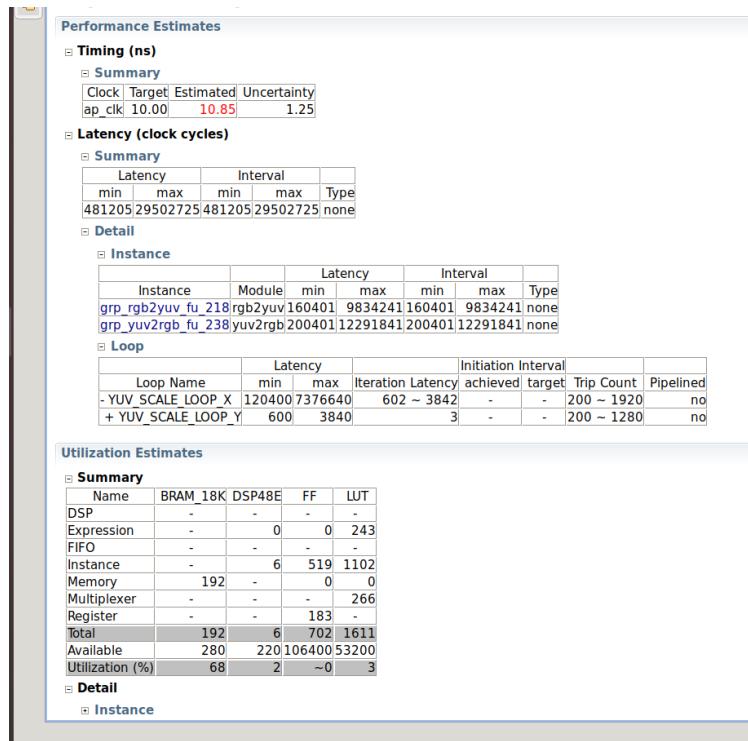


Después de descomentar el TRIPCOUNT vemos que sale información sobre la latencia

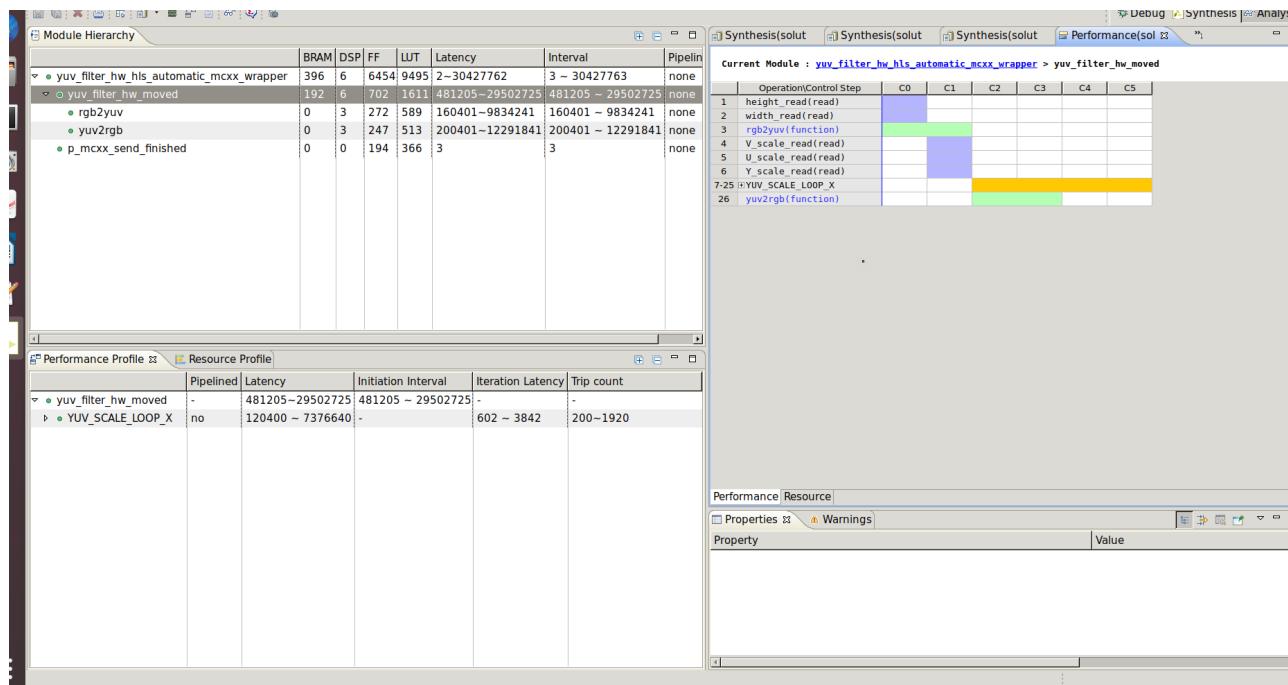


## 2.3 Loop Pipeling

### 1) Sin pipelined



2)

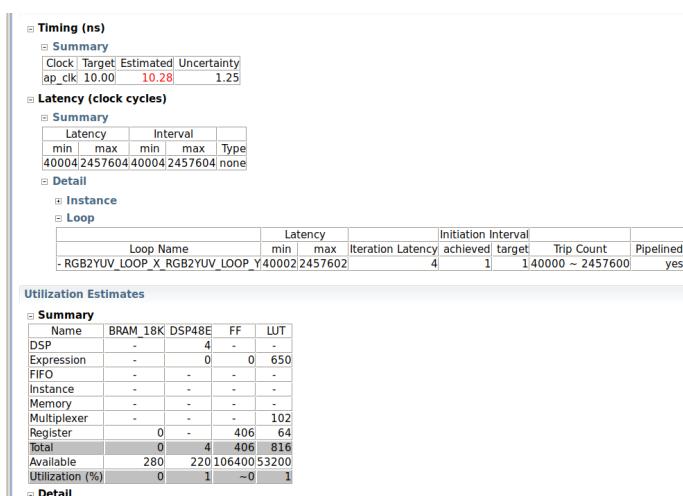


3) y 4)

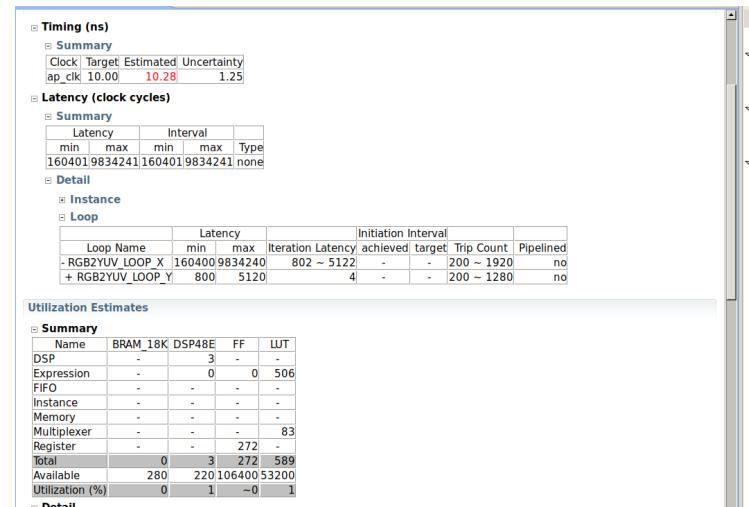
He añadido al código la directiva `#pragma HLS PIPELINE` dentro de los bucles más internos de las tres funciones siguientes

rgb2yuv

PIPELINED YES



PIPELINED NO



## yuv\_filter\_hw

PIPELINED YES

**Timing (ns)**

- Summary**

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	10.85	1.25
- Latency (clock cycles)**
  - Summary**

Latency	Interval			
min	max	min	max	Type
min	max	Type		
Type				
- Detail**
  - Instance**

Instance	Module	min	max	min	max	Type
grp_rgb2yuv_fu_233	rgb2yuv	40004	2457604	40004	2457604	none
grp_yuv2rgb_fu_253	yuv2rgb	40005	2457605	40005	2457605	none
  - Loop**

Loop Name	Latency	Initiation Interval					
min	max	Iteration Latency	achieved	target	Trip Count	Pipelined	
- YUV_SCALE_LOOP_X	YUV_SCALE_LOOP_Y	40002	2457602	4	1	140000 ~ 2457600	yes

**Utilization Estimates**

- Summary**

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	1	-	-
Expression	-	0	0	321
FIFO	-	-	-	-
Instance	0	8	830	1535
Memory	192	-	0	0
Multiplexer	-	-	-	294
Register	0	-	319	32
<b>Total</b>	192	9	1149	2182
Available	280	220	106400	53200
Utilization (%)	68	4	1	4
- Detail**
  - Instance**

PIPELINED NO

**Performance Estimates**

**Timing (ns)**

- Summary**

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	10.85	1.25
- Latency (clock cycles)**
  - Summary**

Latency	Interval			
min	max	min	max	Type
min	max	Type		
Type				
- Detail**
  - Instance**

Instance	Module	min	max	min	max	Type
grp_rgb2yuv_fu_233	rgb2yuv	160401	9834241	160401	9834241	none
grp_yuv2rgb_fu_253	yuv2rgb	200401	12291841	200401	12291841	none
  - Loop**

Loop Name	Latency	Initiation Interval						
min	max	Iteration Latency	achieved	target	Trip Count	Pipelined		
- YUV_SCALE_LOOP_X	YUV_SCALE_LOOP_Y	120400	7376640	602	~ 3842	-	200 ~ 1920	no
+ YUV_SCALE_LOOP_Y	600	3840	3	-	-	200 ~ 1280	no	

**Utilization Estimates**

- Summary**

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	-	-	-
Expression	-	0	0	243
FIFO	-	-	-	-
Instance	-	6	519	1102
Memory	192	-	0	0
Multiplexer	-	-	-	266
Register	-	-	183	-
<b>Total</b>	192	6	702	1611
Available	280	220	106400	53200

## yuv2rgb

PIPELINED YES

**Performance Estimates**

**Timing (ns)**

- Summary**

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	10.85	1.25
- Latency (clock cycles)**
  - Summary**

Latency	Interval			
min	max	min	max	Type
min	max	Type		
Type				
- Detail**
  - Instance**
    - Loop**

Loop Name	Latency	Initiation Interval					
min	max	Iteration Latency	achieved	target	Trip Count	Pipelined	
- YUV2RGB_LOOP_X	YUV2RGB_LOOP_Y	40003	2457603	5	1	140000 ~ 2457600	yes

**Utilization Estimates**

- Summary**

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	4	-	-
Expression	-	0	0	553
FIFO	-	-	-	-
Instance	-	-	-	-
Memory	-	-	-	-
Multiplexer	-	-	-	102
Register	0	-	424	64
<b>Total</b>	0	4	424	719
Available	280	220	106400	53200
Utilization (%)	0	1	~0	1

PIPELINED NO

**Performance Estimates**

**Timing (ns)**

- Summary**

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	10.85	1.25
- Latency (clock cycles)**
  - Summary**

Latency	Interval			
min	max	min	max	Type
min	max	Type		
Type				
- Detail**
  - Instance**
    - Loop**

Loop Name	Latency	Initiation Interval					
min	max	Iteration Latency	achieved	target	Trip Count	Pipelined	
- YUV2RGB_LOOP_X	200401	12291840	1002	~ 6402	-	200 ~ 1920	no
+ YUV2RGB_LOOP_Y	1000	6400	5	-	-	200 ~ 1280	no

**Utilization Estimates**

- Summary**

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	3	-	-
Expression	-	0	0	427
FIFO	-	-	-	-
Instance	-	-	-	-
Memory	-	-	-	-
Multiplexer	-	-	-	86
Register	-	-	247	-
<b>Total</b>	0	3	247	513
Available	280	220	106400	53200
Utilization (%)	0	1	~0	~0

Observamos que para las tres funciones la latencia baja haciendo pipelined. No obstante, requiere que se utilice más recursos hardware como observamos en el Summary. Esto es debido a que, para que se ejecute una instrucción sin que la otra haya finalizado, debe haber más control para que no utilicen los mismos recursos en el mismo ciclo. Además estamos ejecutando más etapas por ciclo, eso requiere de más uso del HW.

## e y f)

La primera imagen es con PIPELINED, como observamos y hemos observado anteriormente, hay un aumento de uso en los recursos HW, si es verdad que bajamos la latencia, pero aumenta el tiempo de uso. Y 2s de diferencia es mucho tiempo considero yo.

```
INFO [Design]: [1200] Entering YUV_FILTER_HW at Tue May 17 10:00:20
Finished synthesis of 'yuv_filter_hw'
DSP48E      9 used |   220 available -  4.09% utilization
BRAM_18K    398 used |   280 available - 142.14% utilization
LUT        14380 used |  53200 available - 27.03% utilization
FF         9020 used | 106400 available -  8.48% utilization
Step 'HLS' finished. 78s elapsed
Step 'design' is disabled
Step 'synthesis' is disabled
Step 'implementation' is disabled
Step 'bitstream' is disabled
Step 'boot' is disabled
Hardware automatic generation finished. 91s elapsed
FPGA Linking performed in 91.506 seconds
Whole process took 92.15 seconds to complete
```

```
INFO [Design]: [1200] Entering YUV_FILTER_HW at Tue May 17 10:00:20
Finished synthesis of 'yuv_filter_hw'
DSP48E      6 used |   220 available -  2.73% utilization
BRAM_18K    398 used |   280 available - 142.14% utilization
LUT        13809 used |  53200 available - 25.96% utilization
FF         8573 used | 106400 available -  8.06% utilization
Step 'HLS' finished. 77s elapsed
Step 'design' is disabled
Step 'synthesis' is disabled
Step 'implementation' is disabled
Step 'bitstream' is disabled
Step 'boot' is disabled
Hardware automatic generation finished. 89s elapsed
FPGA Linking performed in 89.748 seconds
Whole process took 90.24 seconds to complete
```

## 2.4 Loop Fusion

Fusion without pipelined

**Performance Estimates**

- Timing (ns)**
  - Summary**

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	10.28	1.25
- Latency (clock cycles)**
  - Summary**

Latency	Interval			
min	max	min	max	Type
1401	13441	1401	13441	none
  - Detail**
    - Instance**
    - Loop**

Loop Name	Latency	Initiation Interval					
	min	max	Iteration Latency	achieved	target	Trip Count	Pipelined
- YUV_SCALE_LOOP_XY	1400	13440	7	-	-	200 ~ 1920	no

```
INFO [Design]: [1200] Entering YUV_FILTER_HW at Tue May 17 10:00:20
Finished synthesis of 'yuv_filter_hw'
DSP48E      8 used |   220 available -  3.64% utilization
BRAM_18K    206 used |   280 available - 73.57% utilization
LUT        12926 used |  53200 available - 24.3% utilization
FF         8181 used | 106400 available -  7.69% utilization
Step 'HLS' finished. 75s elapsed
Step 'design' is disabled
Step 'synthesis' is disabled
Step 'implementation' is disabled
Step 'bitstream' is disabled
Step 'boot' is disabled
Hardware automatic generation finished. 88s elapsed
FPGA Linking performed in 88.712 seconds
Whole process took 89.16 seconds to complete
```

**Utilization Estimates**

- Summary**

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	8	-	-
Expression	-	0	0	675
FIFO	-	-	-	-
Instance	-	-	-	-
Memory	-	-	-	-
Multiplexer	-	-	-	-
Register	-	-	310	-
Total	0	8	310	728
Available	280	220	106400	53200
Utilization (%)	0	3	-0	1

## Fusion with pipelined

**Timing (ns)**

- Summary**

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	10.28	1.25

**Latency (clock cycles)**

- Summary**

Latency	Interval	min	max	min	max	Type
2071927	2071927	none				

**Detail**

- Instance**
- Loop**

Loop Name	Latency	Initiation Interval	Iteration Latency	Achieved	Target	Trip Count	Pipelined
-YUV_SCALE_LOOP_XY	2051925	7	1	1200 ~ 1920		yes	

**Utilization Estimates**

- Summary**

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	8	-	-
Expression	-	0	0	691
FIFO	-	-	-	-
Instance	-	-	-	-
Memory	-	-	-	-
Multiplexer	-	-	-	48
Register	0	-	425	64
<b>Total</b>	0	8	425	803
Available	280	220	106400	53200
Utilization (%)	0	3	~0	1

```
Finished synthesis of 'yuv_filter_hw'
DSP48E      8 used |   220 available -  3.64% utilization
BRAM_18K    206 used |   280 available - 73.57% utilization
LUT        13001 used |  53200 available - 24.44% utilization
FF         8296 used | 106400 available -  7.8% utilization
Step 'HLS' finished. 76s elapsed
Step 'design' is disabled
Step 'synthesis' is disabled
Step 'implementation' is disabled
Step 'bitstream' is disabled
Step 'boot' is disabled
Hardware automatic generation finished. 89s elapsed
FPGA Linking performed in 89.478 seconds
Whole process took 89.94 seconds to complete
```

## 2.5 Execution with acceleration: Timing

Respecto a la ejecución con aceleración obtenemos:

Time 20 images: 0.08215 s

Time 100 images: 0.36650 s

Sin Acceleración

Time 20 images: 0.1484 s

Time 100 images: 0.7445 s

SpeedUp 20 images: 1.8054

SpeedUp 100 images: 2.0314

```
ubuntu@zyng-bsc:~/lab4$ load_bitstream yuv_filter
yuv_filter -yuv_filter.bln
ubuntu@zyng-bsc:~/lab4$ load_bitstream yuv_filter.bin
ubuntu@zyng-bsc:~/lab4$ ls
test_data.yuv yuv_filter.bln
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 20
Test passed!
Execution time (secs): 0.081372
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 20
Test passed!
Execution time (secs): 0.082894
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 20
Test passed!
Execution time (secs): 0.083516
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 20
Test passed!
Execution time (secs): 0.082168
ubuntu@zyng-bsc:~/lab4$ 
ubuntu@zyng-bsc:~/lab4$ 
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 100
Test passed!
Execution time (secs): 0.367620
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 100
Test passed!
Execution time (secs): 0.368945
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 100
Test passed!
Execution time (secs): 0.369265
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 100
Test passed!
Execution time (secs): 0.365993
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./yuv_filter 100
Test passed!
Execution time (secs): 0.366566
ubuntu@zyng-bsc:~/lab4$ 
```

**Nota:** Con Loop Fusion he querido probar de compilar con make smp-zedboard y he ejecutado el smp\_yuv\_filter en la zedboard y para 20 images he obtenido mejor Timing. Y haciendo lo anterior, siendo el mismo código hay mejora respecto a la ejecución sin aceleración pero no tanta como esta

```
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
Execution time (secs): 0.060194
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
Execution time (secs): 0.060182
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
Execution time (secs): 0.060214
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
Execution time (secs): 0.060168
ubuntu@zyng-bsc:~/lab4$ NX_ARGS="--smp-workers=1" ./smp_yuv_filter 20
Test passed!
Execution time (secs): 0.060199
ubuntu@zyng-bsc:~/lab4$ 
```

Time 20 images: 0.06019 s

SpeedUp20 respecto a sin aceleración  
2.4555