MSoC Self-paced 2: PP4FPGA FIR

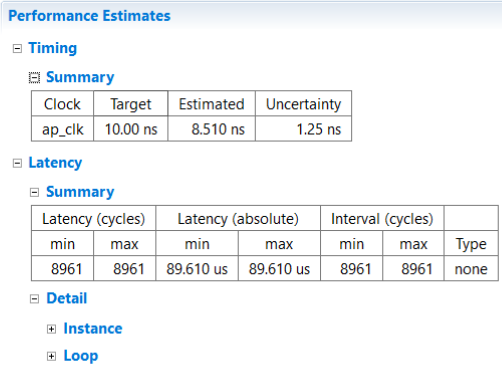
R08943011 黃文璁

- **[v] HLS C-sim/synthesis/cosim**
- **[v] System bring-up (zedboard)**
- **[v] Improvement (latency, area)**
- **[v] Github: https://github.com/b04901060/MSoC-Application-Acceleration-with-High-Level-Synthesis**

1. Introduction

本 example 實作有限脈衝響應 (finite impulse response) 濾波器，輸入訊號和對應的 tap 參數相乘後累加得到輸出。

2. C Synthesis and cosim



Performance Estimates

⊟ Timing

⊟ Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00 ns	8.510 ns	1.25 ns

⊟ Latency

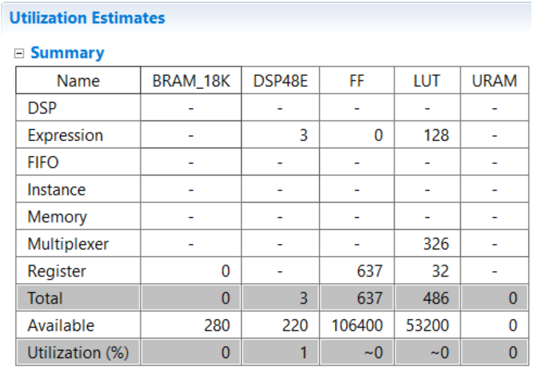
⊟ Summary

Latency (cycles)		Latency (absolute)		Interval (cycles)		
min	max	min	max	min	max	Type
8961	8961	89.610 us	89.610 us	8961	8961	none

⊟ Detail

⊞ Instance

⊞ Loop

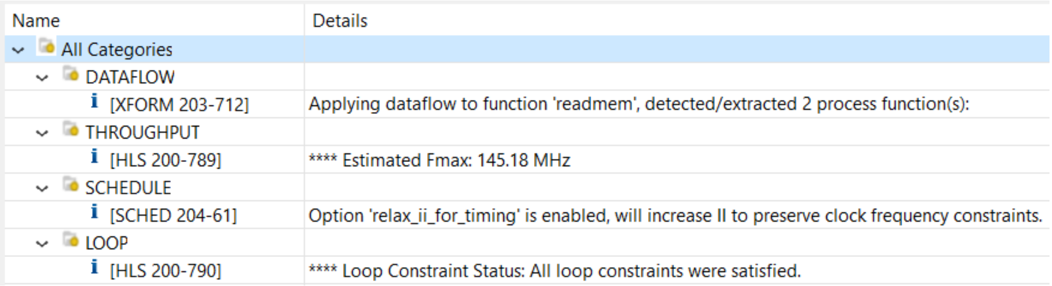


Utilization Estimates

⊟ Summary

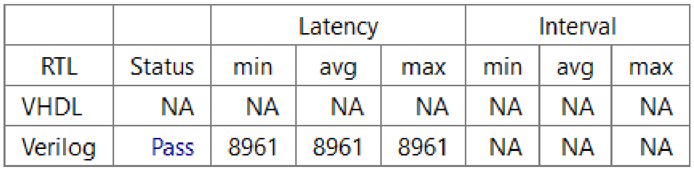
Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	3	0	128	-
FIFO	-	-	-	-	-
Instance	-	-	-	-	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	326	-
Register	0	-	637	32	-
Total	0	3	637	486	0
Available	280	220	106400	53200	0
Utilization (%)	0	1	~0	~0	0

另外可以注意到合成結果沒有 warning：



Name	Details
All Categories	
DATAFLOW	
[XFORM 203-712]	Applying dataflow to function 'readmem', detected/extracted 2 process function(s):
THROUGHPUT	
[HLS 200-789]	**** Estimated Fmax: 145.18 MHz
SCHEDULE	
[SCHED 204-61]	Option 'relax_ii_for_timing' is enabled, will increase II to preserve clock frequency constraints.
LOOP	
[HLS 200-790]	**** Loop Constraint Status: All loop constraints were satisfied.

Cosim:



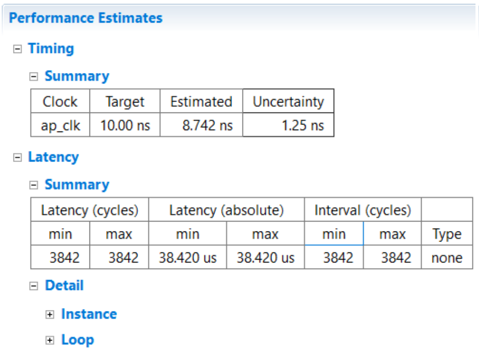
		Latency			Interval		
RTL	Status	min	avg	max	min	avg	max
VHDL	NA	NA	NA	NA	NA	NA	NA
Verilog	Pass	8961	8961	8961	NA	NA	NA

-> Verilog pass

3. Optimization

在原版中的雙層迴圈只在最內層的迴圈加上 PIPELINE pragma，我的版本中再於外圈 loop 加上 PIPELINE 指令，以達到更低的 latency，但便需要更多 FPGA resource，在 area 和 performance 間做 tradeoff。

- **Latency optimized synthesis result**



Performance Estimates

⊟ **Timing**

⊟ **Summary**

Clock	Target	Estimated	Uncertainty
ap_clk	10.00 ns	8.742 ns	1.25 ns

⊟ **Latency**

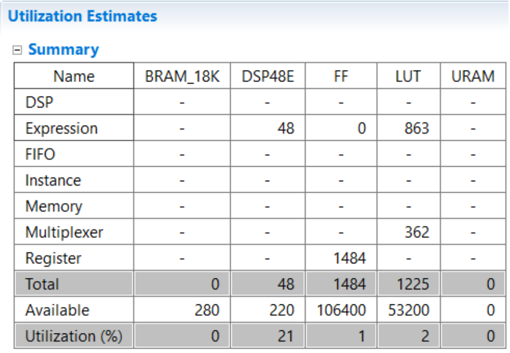
⊟ **Summary**

Latency (cycles)		Latency (absolute)		Interval (cycles)		
min	max	min	max	min	max	Type
3842	3842	38.420 us	38.420 us	3842	3842	none

⊟ **Detail**

⊞ **Instance**

⊞ **Loop**



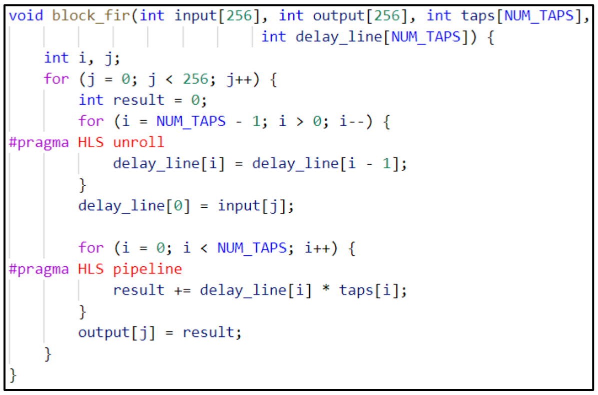
Utilization Estimates

⊟ **Summary**

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	48	0	863	-
FIFO	-	-	-	-	-
Instance	-	-	-	-	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	362	-
Register	-	-	1484	-	-
Total	0	48	1484	1225	0
Available	280	220	106400	53200	0
Utilization (%)	0	21	1	2	0

Latency:	8961	->	3842	**(-57%)**
FF:	637	->	1484	**(+133%)**
LUT:	486	->	1225	**(+152%)**

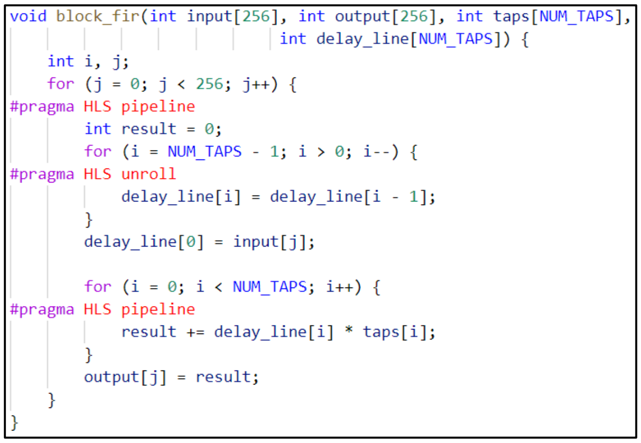
Before



```
void block_fir(int input[256], int output[256], int taps[NUM_TAPS],
                                    int delay_line[NUM_TAPS]) {
    int i, j;
    for (j = 0; j < 256; j++) {
        int result = 0;
        for (i = NUM_TAPS - 1; i > 0; i--) {
#pragma HLS unroll
            delay_line[i] = delay_line[i - 1];
        }
        delay_line[0] = input[j];

        for (i = 0; i < NUM_TAPS; i++) {
#pragma HLS pipeline
            result += delay_line[i] * taps[i];
        }
        output[j] = result;
    }
}
```

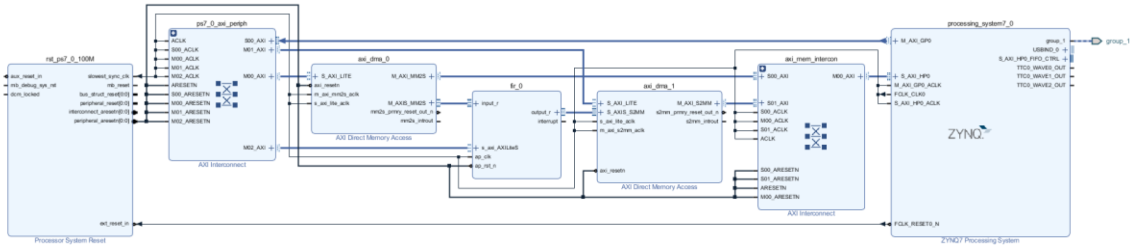
After



```
void block_fir(int input[256], int output[256], int taps[NUM_TAPS],
                                    int delay_line[NUM_TAPS]) {
    int i, j;
    for (j = 0; j < 256; j++) {
#pragma HLS pipeline
        int result = 0;
        for (i = NUM_TAPS - 1; i > 0; i--) {
#pragma HLS unroll
            delay_line[i] = delay_line[i - 1];
        }
        delay_line[0] = input[j];

        for (i = 0; i < NUM_TAPS; i++) {
#pragma HLS pipeline
            result += delay_line[i] * taps[i];
        }
        output[j] = result;
    }
}
```

4. Block Diagram



rst_ps7_0_100M
aux_reset_in
mb_debug_sys_rst
dcm_locked
slowest_sync_clk
mb_reset
bus_struct_reset[0:0]
peripheral_reset[0:0]
interconnect_aresetn[0:0]
peripheral_aresetn[0:0]
ext_reset_in
Processor System Reset
ps7_0_axi_periph
ACLK
S00_ACLK
M00_ACLK
M01_ACLK
M02_ACLK
ARESETN
S00_ARESETN
M00_ARESETN
M01_ARESETN
M02_ARESETN
S00_AXI
M01_AXI
M00_AXI
M02_AXI
AXI Interconnect
axi_dma_0
S_AXI_LITE
axi_resetn
m_axi_mm2s_aclk
s_axi_lite_aclk
M_AXI_MM2S
M_AXIS_MM2S
mm2s_prmry_reset_out_n
mm2s_introut
AXI Direct Memory Access
fir_0
input_r
output_r
interrupt
s_axi_AXILiteS
ap_clk
ap_rst_n
axi_dma_1
S_AXI_LITE
S_AXIS_S2MM
s_axi_lite_aclk
m_axi_s2mm_aclk
axi_resetn
M_AXI_S2MM
s2mm_prmry_reset_out_n
s2mm_introut
AXI Direct Memory Access
axi_mem_intercon
S00_AXI
S01_AXI
S00_ACLK
M00_ACLK
S01_ACLK
ACLK
S00_ARESETN
S01_ARESETN
ARESETN
M00_ARESETN
M00_AXI
AXI Interconnect
processing_system7_0
M_AXI_GP0
S_AXI_HP0
M_AXI_GP0_ACLK
FCLK_CLK0
S_AXI_HP0_ACLK
FCLK_RESET0_N
group_1
USBIND_0
S_AXI_HP0_FIFO_CTRL
TTC0_WAVE0_OUT
TTC0_WAVE1_OUT
TTC0_WAVE2_OUT
ZYNQ
ZYNQ7 Processing System
group_1