

### 1. FIR 11 Baseline

- a) Latency: 19 clock cycles, Initiation Interval II: 20 clock cycles
- b) BRAM: 0, DSP: 2, FF: 733, LUT: 383

### 2. Variable Bitwidths

a)

Bitwidth (coef_t, acc_t)	Latency	Initiation Interval II	Resource Usage (BRAM, DSP, LUT, FF)
(128,128)	135 clock cycles	136 clock cycles	(2, 2, 330, 622)
(16,16)	134 clock cycles	135 clock cycles	(2, 1, 239, 244)
(5,16)	134 clock cycles	135 clock cycles	(2, 1, 239, 244)

b) Minimum bitwidth of **coef\_t** and **acc\_t**:

- i) Minimum **coef\_t** bitwidth: **5**
- ii) Minimum **acc\_t** bitwidth: **16**

### 3. Pipelining

- a) Latency: 135, Initiation Interval: 136, BRAM: 3, DSP: 2, FF: 610, LUT: 305
- b) Latency: min - 257 max - 641, Initiation Interval: min - 258 max - 642, BRAM: 2, DSP: 2, FF: 412, LUT: 279

c)

II=<value>	Estimated clock period (ns)	Latency (clock cycles)	Initiation Interval II (clock cycles)	Resource Usage (BRAM, DSP, LUT, FF)	Throughput
1	6.912	135	136	3,2,305,610	1.063 MHz
2	6.912	262	263	2,2,513,321,0	550.1 KHz
3	6.912	389	390	2,2,480,318,0	371 KHz
4	6.912	516	517	2,2,480,323,0	279.8 KHz
5	6.912	643	644	2,2,453,300,0	224.7 KHz
6	6.912	643	644	2,2,453,300,0	224.7 KHz

d) Largest sensible II value: **5**

e) Default II value: **1**

### 4. Removing Conditional Statements

a) Automatically pipelined

Condition	Latency (clock cycles)	Initiation Interval II (clock cycles)	Resource Usage (BRAM, DSP, LUT, FF)
-----------	------------------------	---------------------------------------	-------------------------------------

With Conditional	135	136	3,2,305,610
Without Conditional	134	135	2,2,316,418

b) Non-pipelined

Condition	Latency (clock cycles)	Initiation Interval II (clock cycles)	Resource Usage (BRAM, DSP, LUT, FF)
With Conditional	min - 257 max - 641	min - 258 max - 642	2,2,412,279
Without Conditional	636	637	2,2,270,345

## 5. Loop Partitioning

- a) Split the instructions in the for loop to have their own separate loop so we can optimize each instruction separately.

b)

Partitioning	Latency (clock cycles)	Initiation Interval II (clock cycles)	Resource Usage (BRAM, DSP, LUT, FF)
With Loop Partitioning	267	268	3,2,343,400
Without Loop Partitioning	135	136	3,2,305,610

- c) Loop unrolling with loop partitioning:

Latency (clock cycles): 172, Initiation Interval II (clock cycles): 173, Resource Usage (BRAM, DSP, LUT, FF): 6,8,769,1249

- d) Loop pipelining parallelizes based on the timing of instructions and loop unrolling does parallelism based on the number of operations that can be executed in parallel. They can be applied together. To conclude, pipelining does not duplicate the hardware on the board. It just uses the same resources on separate time intervals. On the other hand, unrolling uses more resources to process the loop in parallel. In part c we saw that when implementing loop unrolling there were more resources used when compared to part b results.

## 6. Memory Partitioning

- a) Latency (clock cycles): 170, Initiation Interval II (clock cycles): 171, Resource Usage (BRAM, DSP, LUT, FF): 0,8,7544,5258

Memory Partitioning	Latency (clock cycles)	Initiation Interval II (clock cycles)	Resource Usage (BRAM, DSP, LUT, FF)
---------------------	------------------------------	--	--

Complete	170	171	0,8,7544,5258
Cyclic factor = 4	173	174	8,8,823,1298
Block factor = 4	205	206	8,8,1301,1278

Complete partition performs the best

b) Block factor = 4 and without loop unrolling:

Latency (clock cycles): 268, Initiation Interval II (clock cycles): 269, Resource Usage (BRAM, DSP, LUT, FF): 280,220,53200,106400

## 7. Best Design

a) In this optimization we used variable bitwidth, complete memory partition, and loop unrolling

Latency (clock cycles): 27, Initiation Interval II (clock cycles): 28, Throughput: 5.408 Mhz

b) (BRAM, DSP, LUT, FF): 0,1, 2200, 3181