An Alternative Fracturable Multiply & Shift Hard-Block for FPGA

Presented by

Jean-Philippe Legault FCS University of New Brunswick

Collaborators

Dr. Kenneth Kent FCS University of New Brunswick

M.Sc Panagiotis Patros FCS University of New Brunswick

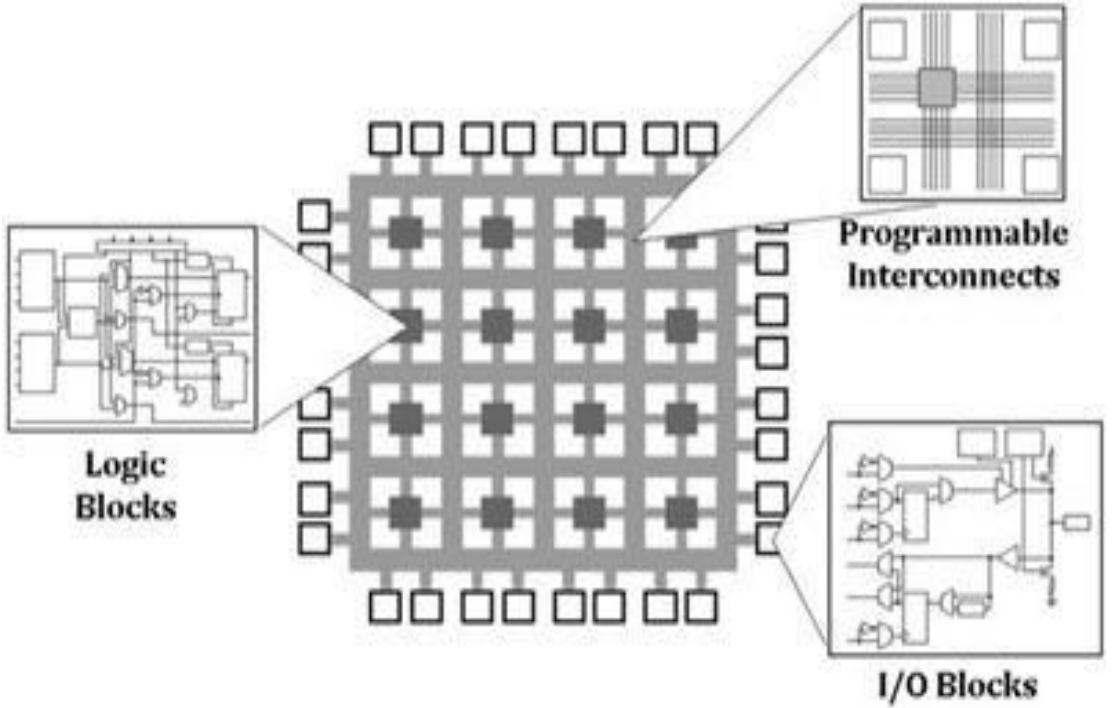
Aaron Graham FCS University of New Brunswick

On October 14, 2017

FPGA

Reconfigurable hardware used to implement digital circuit.

They offer a great alternative for low power devices like in the IOT field or for unmanned aerial imaging systems

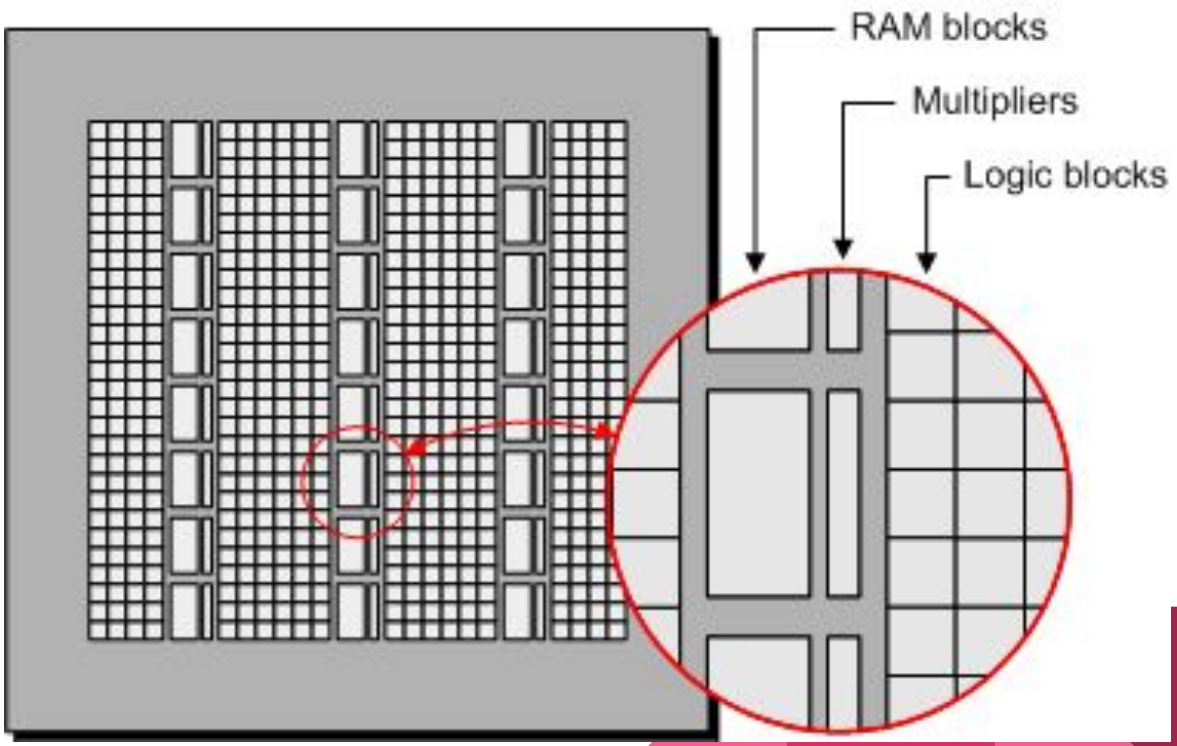


FPGA Hard-Blocks

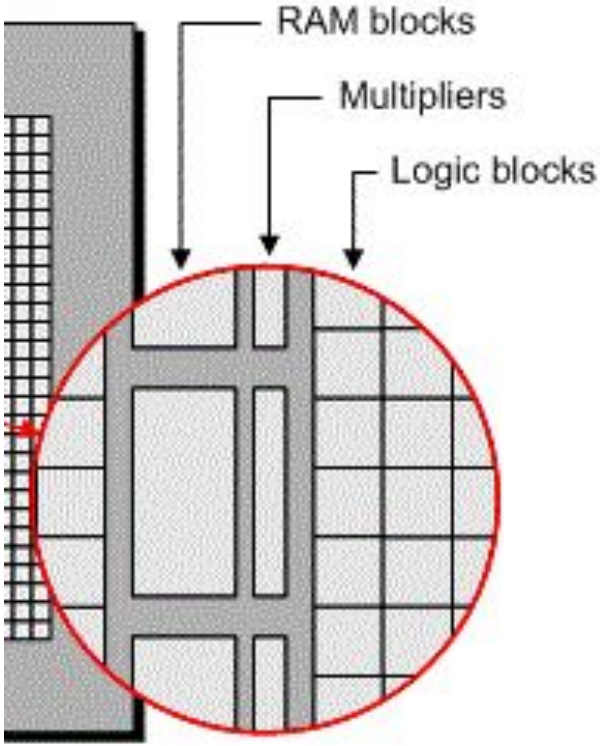
Hard-Blocks are static components embedded in the fabric of the FPGA

Improve throughput of slow operations:

- Multiplication
- Addition



Proposed Multiplier and Shift Hard-Block

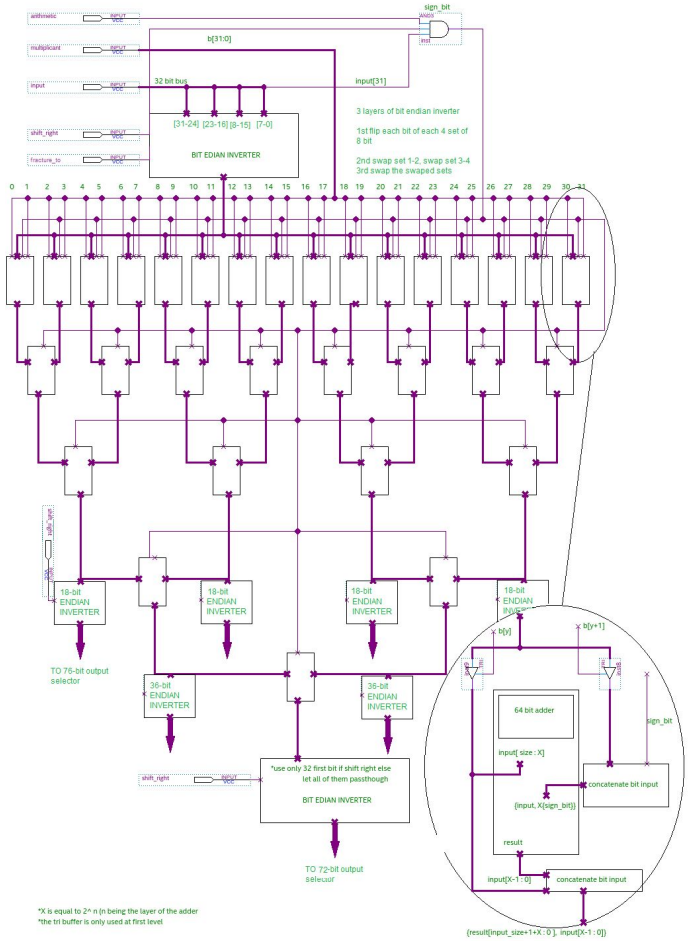


Offers:

- Better reusability than the current multiplier Hard-Block.
- Higher bit manipulation throughput using current fabric layout

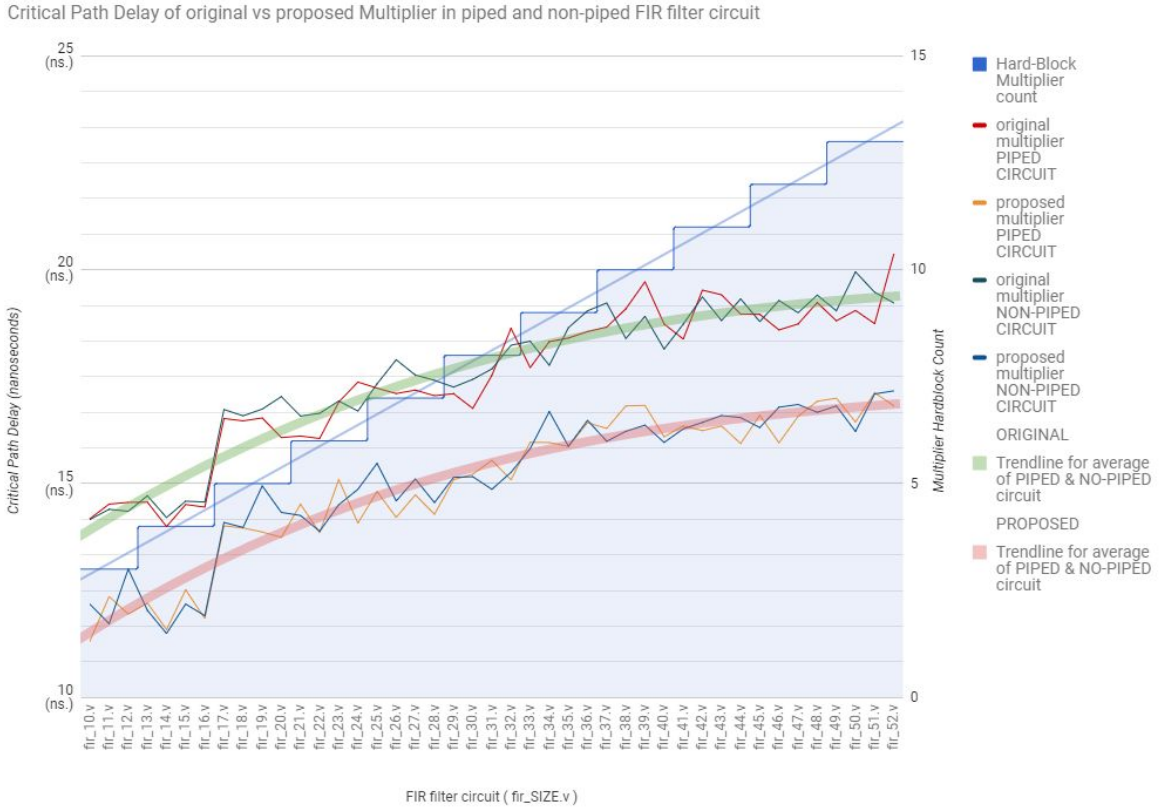
Block Design Proposal

- Higher throughput when fractured
- Allows right arithmetic bit-shifting (padding MSB with 1’s)
- Adder built using multiplexers from transmission gates:
  - shorter delay
  - lower power usage
- Direct replacement of current design

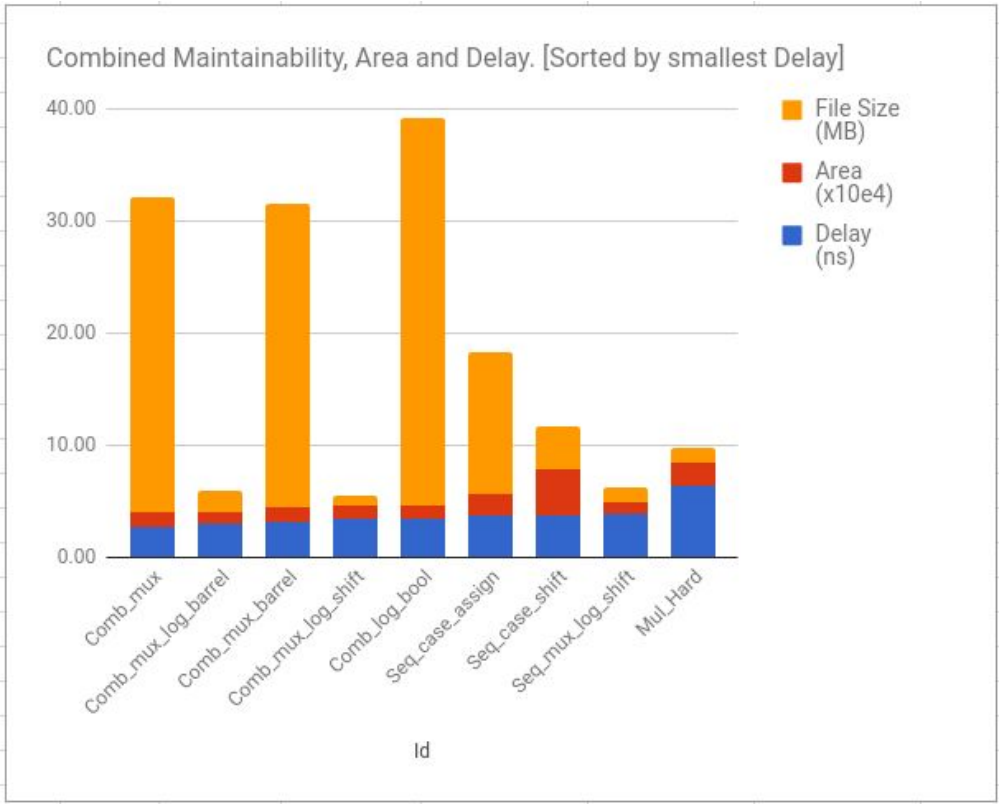


Multiplier Performance Results

Gain averaging 14% in speed for multiplication



Soft-Logic Bit-Shifter Performance Baseline



10 different Verilog standalone implementations

- Critical path delay (ns)

Best result is 383 Mhz.

Performance of Shifting

In standalone module:

582 Mhz. for proposed hard-block.

vs

383 Mhz. for comb_mux.

This is a speed increase of 51%

Conclusion

Emerging fields like IoT and UAV imaging would benefit the proposed FPGA multiplier due to its speed.

The designed multiplier and shift block also allows for a greater reusability, leading into smaller placement in FPGA mapping.

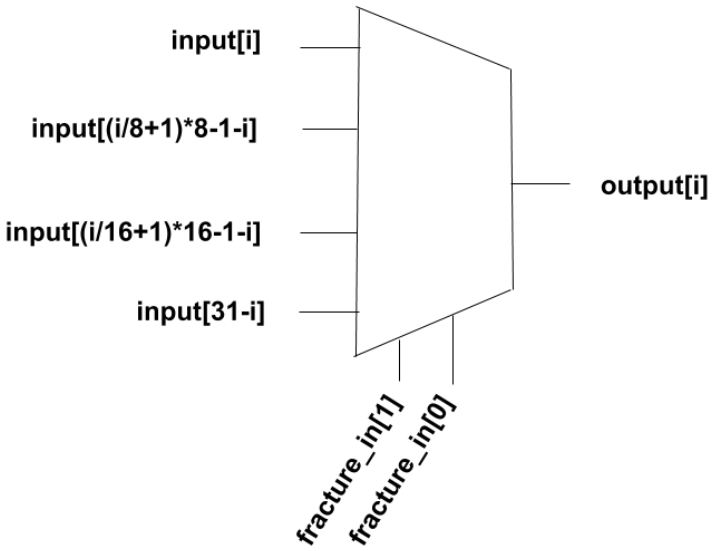
The proposed hard-block offers 51% speed increase over soft-logic for shifting.

Direct substitute for shifting and multiplication.

Design allows for floating point algebra in a single hard-block when backed by neighboring RAM and a latch.

Fracturing Input & Endian Inversion

	S1 S0			
I	00	01	10	11
0	0	7	15	31
1	1	6	14	30
2	2	5	13	29
...	...	...	...	...
7	7	0	8	24
8	8	15	7	23
...	...	...	...	...
15	15	8	0	16
16	16	24	31	15
...	...	...	...	...
24	24	9	25	8
25	25	31	24	7
...	...	...	...	...
30	30	26	17	1
31	31	25	16	0



A 4:1 multiplexer at each binary input and output select the bit based on

- fracturation level
- shifted right.

* fracturable 32-bit input binary endian inversion

Logarithmic Add & Shift

I.E: 1001 1101 X 1001 (MxN)

(1)? 1001 1101 **((N[0])? M<<0: 0)**
+ (0)? 1 0011 1010 **((N[1])? M<<1: 0)**

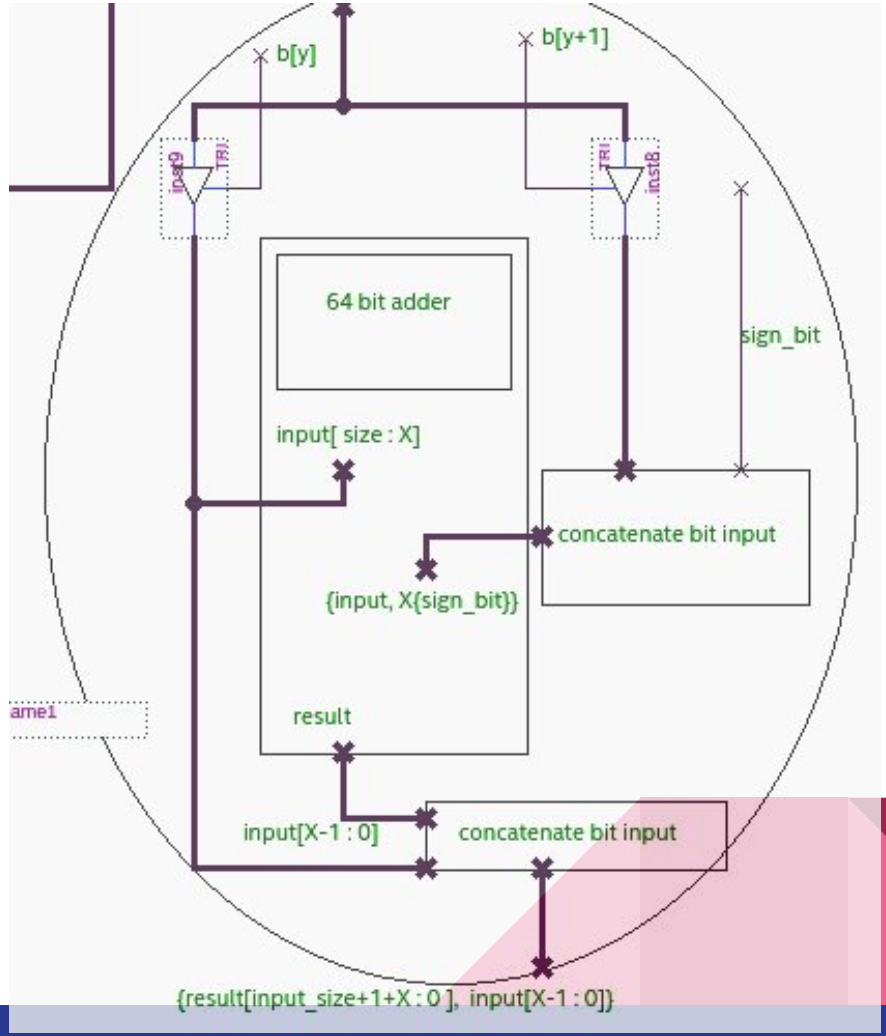
A

(0)? 1001 1100 **((N[2])? M<<0: 0)**
+ (1)? 1 0011 1010 **((N[3])? M<<1: 0)**

B

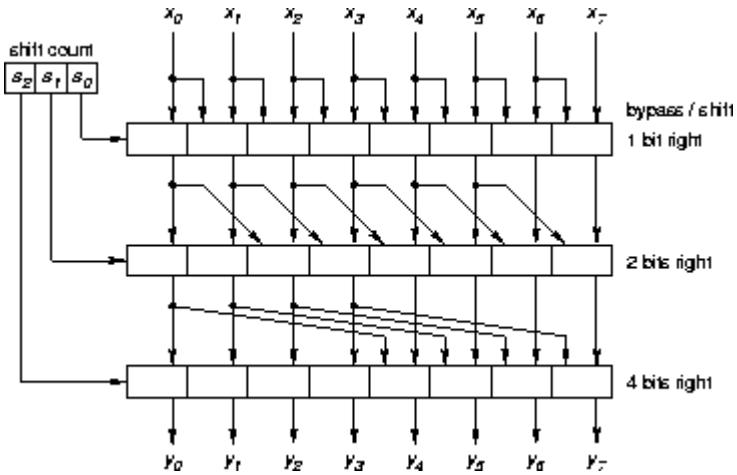
A
+ B<<2

Output



Soft-Logic Bit-Shifter Performance Baseline

The Logarithmic shifter shifts the bits by a power of two at each multiplexer layer.



Delay is a function of Log_2(input bit-width)