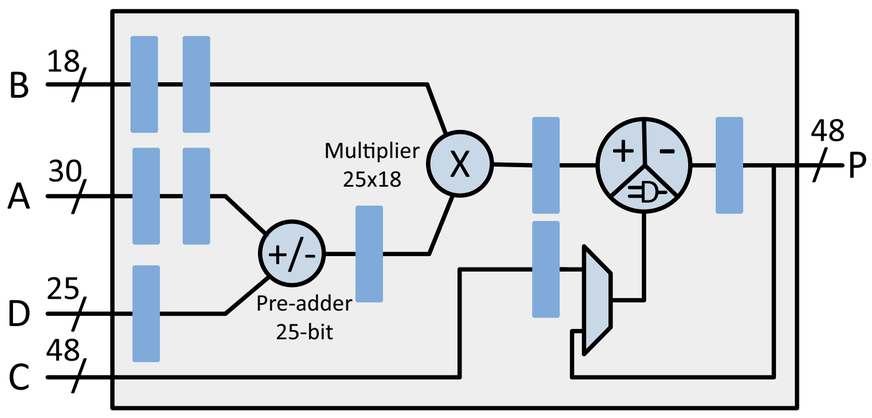
Evaluating the Efficiency of DSP Block Synthesis Inference From Flow Graphs

Bajaj Ronak, Suhaib A Fahmy

NANYANG TECHNOLOGICAL UNIVERSITY

Introduction and Motivation

- Embedded DSP Blocks in modern FPGAs highly capable
  - ✓ Can be dynamically configured to execute many functions
  - ✓ Support cascading of multiple DSP Blocks
  - ✓ Contain pipeline registers for high throughput
  - ✓ Can run up to 600 MHz
- Increased design automation complexity

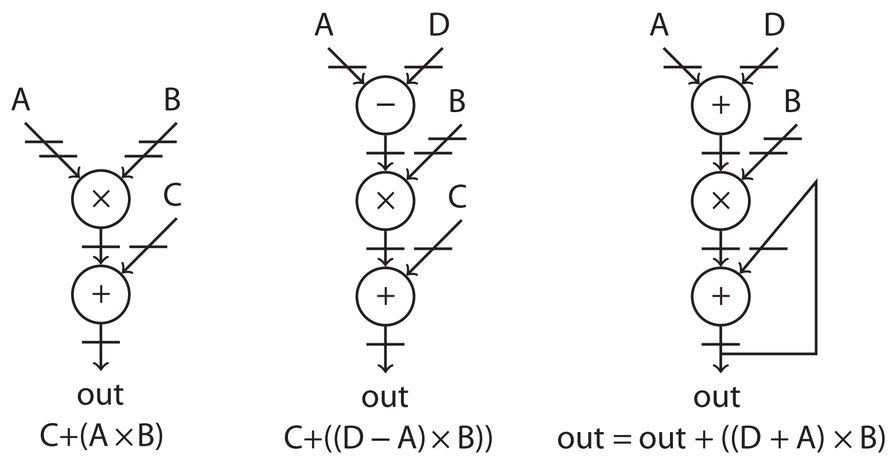


Basic Structure of DSP Block

- Vendor tools can fail to find the most efficient mapping, especially where these blocks are used in multiple configurations.

Comparison Approach

- DSP Blocks can be configured in multiple ways
- Prepared a database of 29 different DSP Block configurations
- We investigated the disparity between automatic inference of DSP Blocks from dataflow graphs, and direct instantiation of DSP configuration templates



Dataflow graphs of templates

Each circuit is implemented in three different ways:

1. As combinational logic (Comb)
2. As pipelined RTL mirroring structure of a DSP Block (Pipe)
3. By direct instantiation of DSP Blocks with appropriate configurations (Direct)

- As a first step, we implemented a circuit, mirroring DSP Block [((A+D)×B)+C]

	DSPs	LUTs	Regs	Max Freq (MHz)
Comb	1	235	99	518
Pipe	1	0	0	560
Direct	1	0	0	560

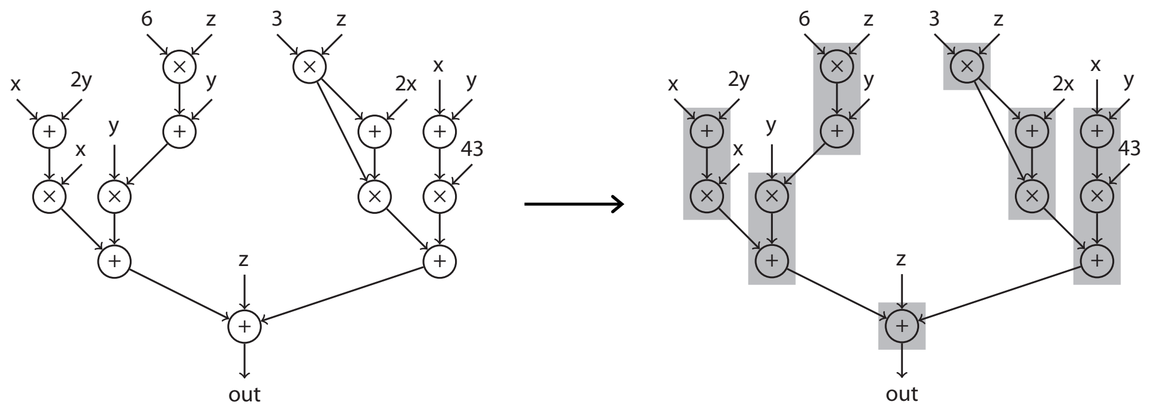
Verilog code for Pipe implementation

```
always @(posedge clk)
begin
    pipe_reg_A1 <= A;     pipe_reg_B1 <= B;
    pipe_reg_D  <= D;
    pipe_reg_B2 <= pipe_reg_B1;
    pipe_reg_AD <= pipe_reg_D + pipe_reg_A1[24:0];
    pipe_reg_C  <= C;
    pipe_reg_M  <= pipe_reg_AD * pipe_reg_B2;
    pipe_reg_P  <= pipe_reg_M + pipe_reg_C;
end
assign P = pipe_reg_P;
```

Verilog code for Direct implementation

```
always @(posedge clk)
begin
    pipeline_reg_C_L1 <= C;
    pipeline_reg_C_L2 <= pipeline_reg_C_L1;
end
DSP48E1 #(…, .AREG(1), .BREG(2), .CREG(1), .INMODEREG(1),
          .A_INPUT("DIRECT"), .B_INPUT("DIRECT")…)
dsp48e1inst (…, .ALUMODE(4'b0000), .CARRYINSEL(3'b000),
             .INMODE(5'b00101), .OPMODE(7'b0110101),…);
```

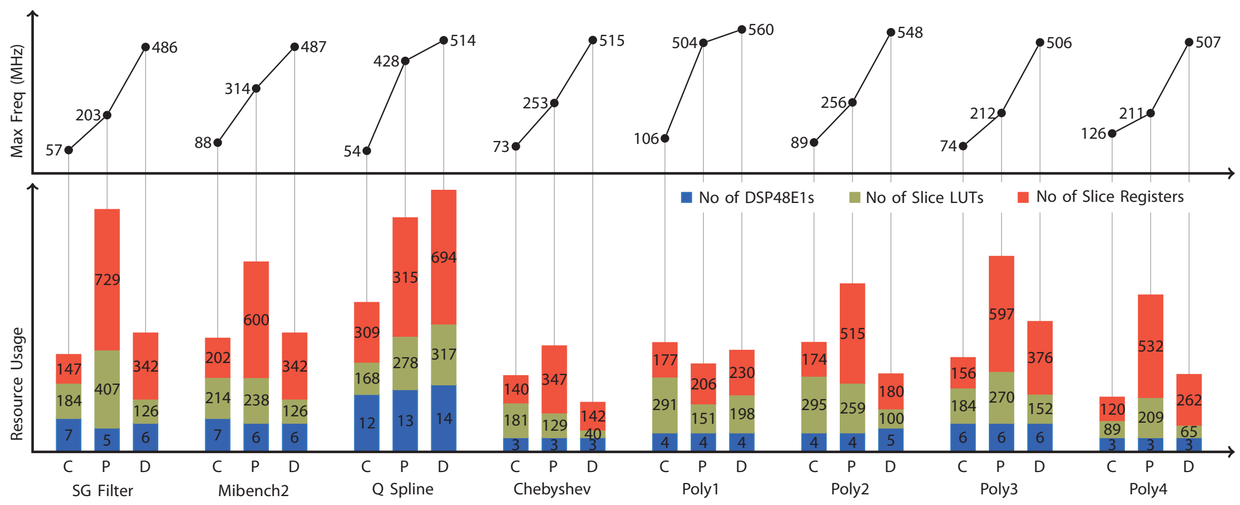
- When similar implementations are done for larger circuits, we observe that DSP Blocks are not always inferred efficiently
- We implemented a variety of algorithms using all three methods
- For the Direct method, segmentation is done so that number of segments is minimised



Dataflow graph of mibench2 filter before segmentation

Dataflow graph of mibench2 filter after segmentation

Results



Direct implementation Improvements over…

	Comb	Pipe
Max Freq	4–9.5x	1.1–2.4x
LUTs	0.8–1.03x	0.7–1.3x

Future Work

- Automate the mapping from RTL and High-Level descriptions
- Investigate resource-constrained mapping using dynamic reconfigurability of DSP Blocks

email: {ronak1, sfahmy}@ntu.edu.sg