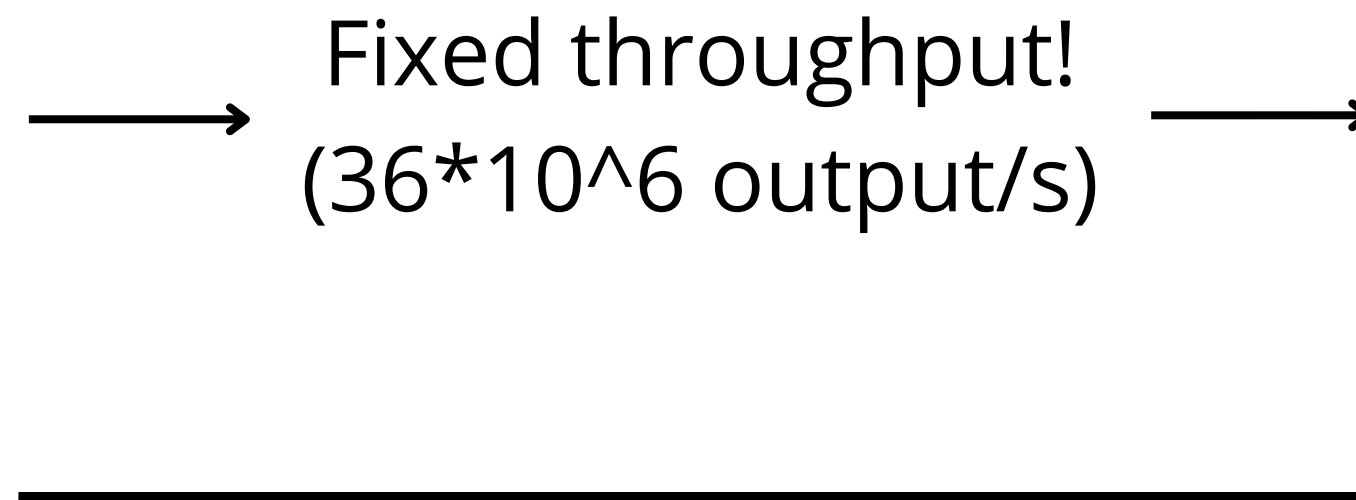# Reducing 8 taps FIR filter power consumption: Low Power Challenge

Matteo Matta, Fabio Piras

Which architecture?

- 2 MUL ?
- 4 MUL ?
- PIPELINED ?
- PARALLEL ?

→ Fixed throughput!
(36*10^6 output/s)

→ The higher the slack,
the higher the margin
to reduce power
+
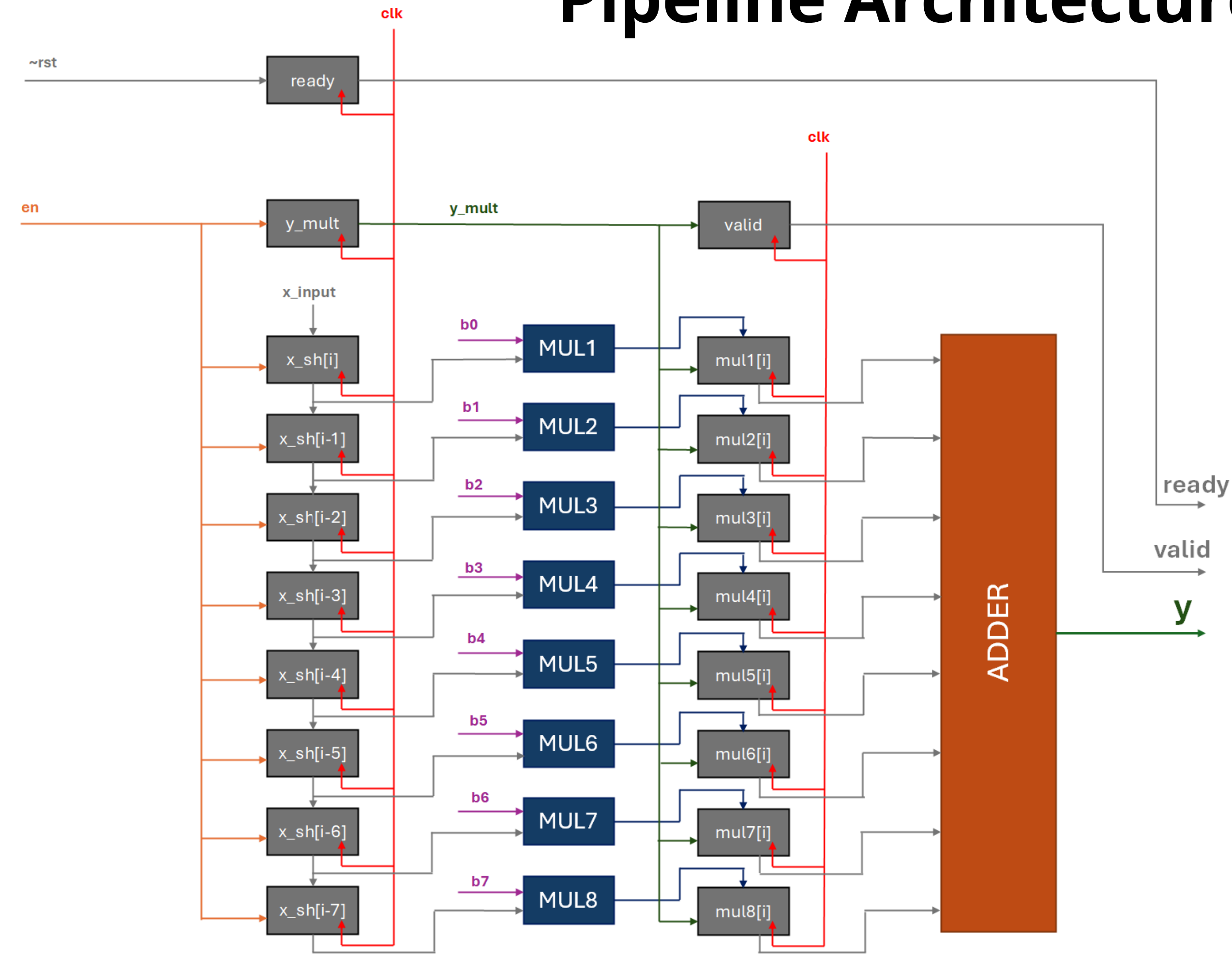lower power
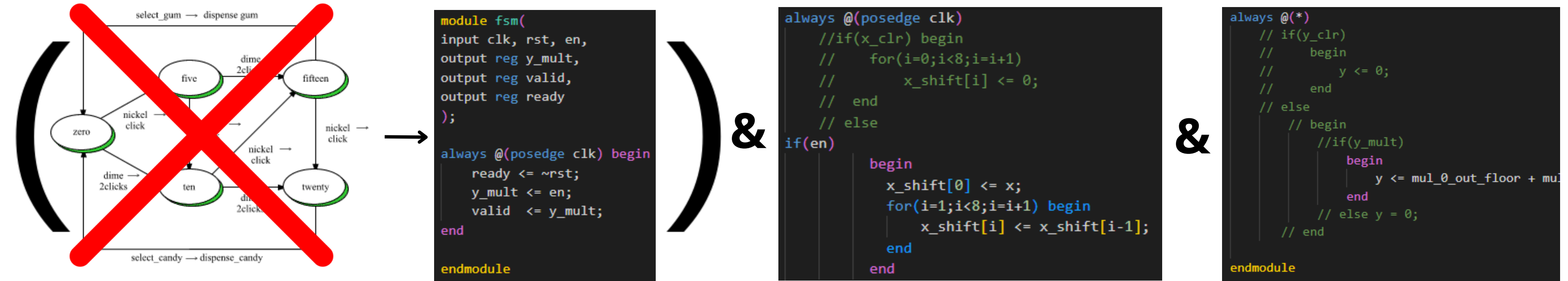consumption at
beginning

So... **PIPELINED**
or
PARALLEL

?

Which architecture?

- 2 MUL ?
- 4 MUL ?
- PIPELINED ?
- PARALLEL ?

$\longrightarrow$

Fixed throughput!
($36*10^6$ output/s)

$\longrightarrow$

The higher the slack,
the higher the margin
to reduce power
+
lower power
consumption at
beginning

So...   **PIPELINED**
or
PARALLEL   **?**   **PIPELINED !**

Pipeline Architecture

$THR = 36 \times 10^6$ output/cc

$T\_clock = 27.7$ ns

Simplifying the datapath and the control logic


```verilog
module fsm(
input clk, rst, en,
output reg y_mult,
output reg valid,
output reg ready
);

always @(posedge clk) begin
    ready <= ~rst;
    y_mult <= en;
    valid  <= y_mult;
end

endmodule
```

&

```verilog
always @(posedge clk)
//if(x_clr) begin
//    for(i=0;i<8;i=i+1)
//        x_shift[i] <= 0;
//   end
// else
if(en)

    begin
        x_shift[0] <= x;
        for(i=1;i<8;i=i+1) begin
            x_shift[i] <= x_shift[i-1];
        end
    end
```

&

```verilog
always @(*)
    // if(y_clr)
    //     begin
    //         y <= 0;
    //     end
    // else
        // begin
            //if(y_mult)
                begin
                    y <= mul_0_out_floor + mu
                end
            // else y = 0;
        // end
endmodule
```

Stimulating clock-gating by the synthesizer

```verilog
always @(posedge clk)
    if (y_mult) begin
        mul_0_out <= x_shift[0]*b0;
        mul_1_out <= x_shift[1]*b1;
        mul_2_out <= x_shift[2]*b2;
        mul_3_out <= x_shift[3]*b3;
        mul_4_out <= x_shift[4]*b4;
        mul_5_out <= x_shift[5]*b5;
        mul_6_out <= x_shift[6]*b6;
        mul_7_out <= x_shift[7]*b7;
    end
    // else begin
    //     mul_0_out <= mul_0_out;
    //     mul_1_out <= mul_1_out;
    //     mul_2_out <= mul_2_out;
    //     mul_3_out <= mul_3_out;
    //     mul_4_out <= mul_4_out;
    //     mul_5_out <= mul_5_out;
    //     mul_6_out <= mul_6_out;
    //     mul_7_out <= mul_7_out;
    // end
```

&

```verilog
integer i;
always @(posedge clk)
    if(en)
        begin
        x_shift[0] <= x;
        for(i=1;i<8;i=i+1) begin
            x_shift[i] <= x_shift[i-1];
        end
    end
```

Moving the output register
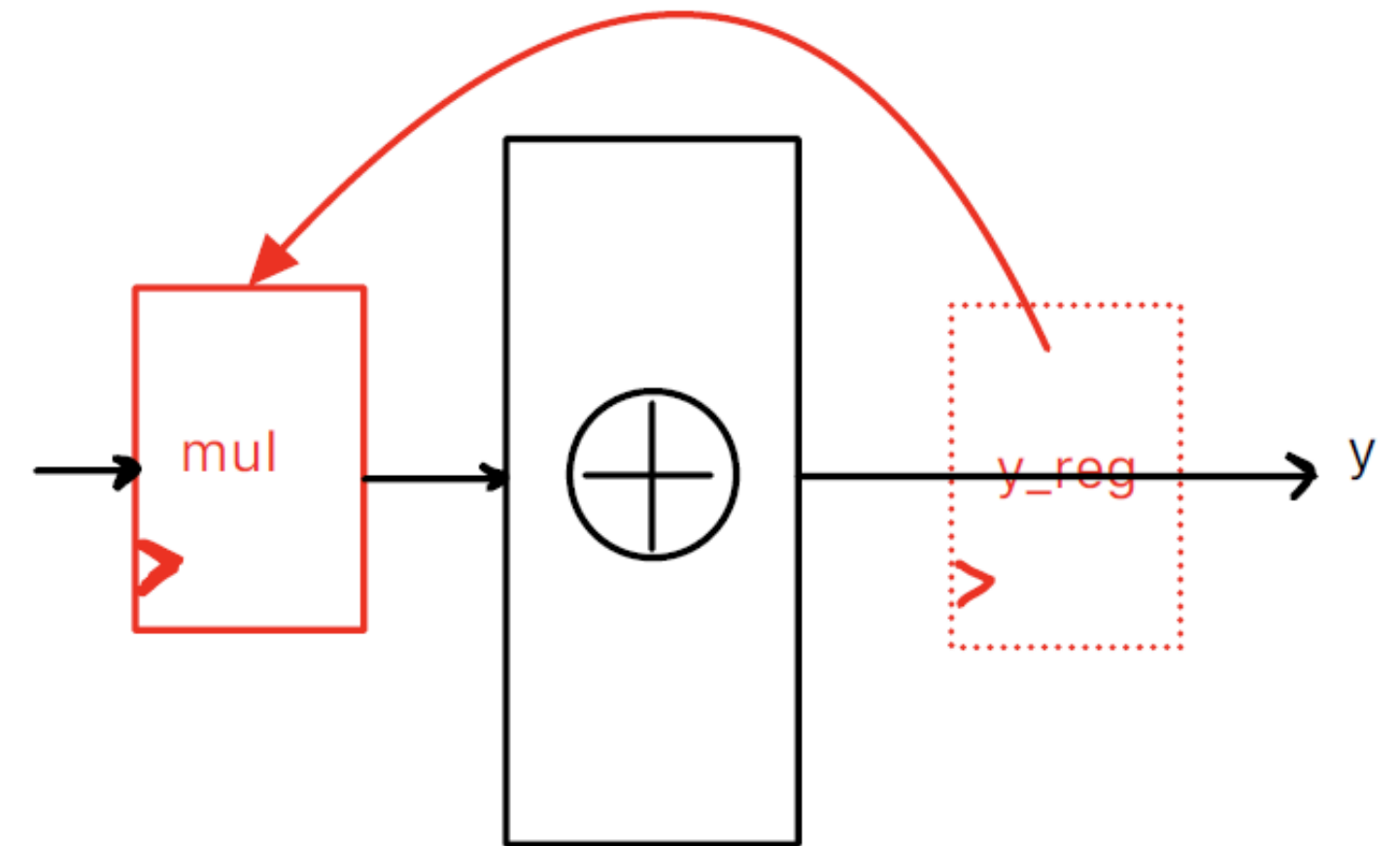
Computing the final power consumption...

gate-level switching

```
          "clk"  :   "0.500262  1805";
          "rst"  :   "0.000880  1";
          "en"   :   "0.433120  843";
```

+

enabling clock-gating

```
34 set attribute lp insert clock gating true /
```

+

setting a constraint for dynamic power

```
9 set_attribute max_dynamic_power 200 /designs/top
```

+

down-scaling the power voltage

```
27 set attribute library liberty LPMOS/v3 0 1/PVT 1 20V range/D CELLS HD LPMOS typ 1 20V 25C.lib
```

Results at each optimization stage

Starting Design · Gated Design · Final Design

Final result

Total Power ──────────────→ 663,977 uW

Energy/output ──────────────→ 18,3927 pJ