Digital Integrated Circuits and Systems

Andrej Trost

Lab 6: Digital filter with HLS (2)

Tool: Vitis 2025.1
Task: Describe in C++ N-tap digital filter, simulate and synthesize with directives.

UNIVERSITY OF LJUBLJANA
FE
Faculty of Electrical Engineering

Synthesize N-tap filter

- filter performance with different directives
  - Unroll delay loop
  - Unroll summation loop
  - Pipeline complete function

circuit	clock [MHz]	latency cycles	interval cycles	LUT / FF / DSP

- Is it possible to synthesize N-tap filter with parameter N?
  - explore synthesis and directives...

Finite Impulse Response (FIR) filter

input

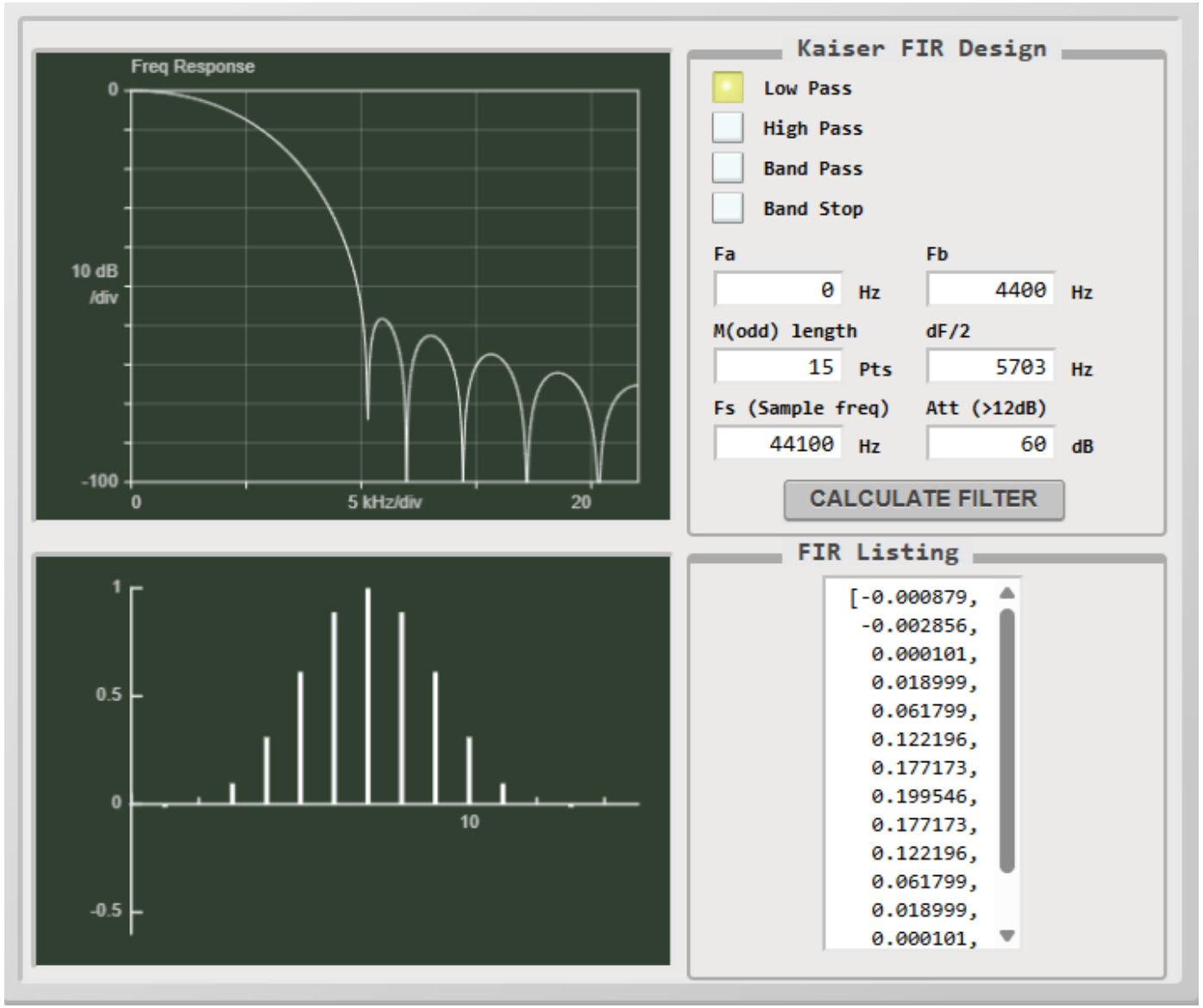
delayed inputs

$$y[n] = b_0 x[n] + b_1 x[n-1] + \cdots + b_N x[n-N]$$

$$= \sum_{i=0}^{N} b_i \cdot x[n-i],$$

coefficients

1. compute coefficients
2. design FIR filter model
3. simulate
4. synthesize and get performance



https://www.arc.id.au/FilterDesign.html

Design 7-tap FIR filter

- Modify 7-tap averaging filter to FIR filter with coefficients

```
DT c[7] = {0.0041, 0.0766, 0.245, 0.346, 0.245, 0.0766, 0.0041};
```

- Verify C++ model with unit pulse
  - input sequence: 0.99, 0, 0, 0, 0, 0, 0, 0, ... produces output sequence with coefficients
- Test with the provided testbench

```
int main() {
 DT x[] = {0.99, 0, 0, 0, 0, 0, 0, 0};

 for (int i = 0; i < 8; i++) {
   DT y = filt(x[i]);
   std::cout << "x=" << (float)x[i]
   << " y=" << (float)y << "\n";
 }

 return 0;
}
```

FIR filter synthesis

- Synthesize the FIR circuit with UNROLL and PIPELINE directives
- Modify data type for coefficients and sum
  - check filter operation with the provided test bench
  - save synthesis performance data for different configurations
  - use different coefficients set, for example:

```
0.0508,-0.1602,-0.1172,0.457,-0.1172,-0.1602,0.0508
```

circuit	clock [MHz]	latency cycles	interval cycles	LUT / FF / DSP