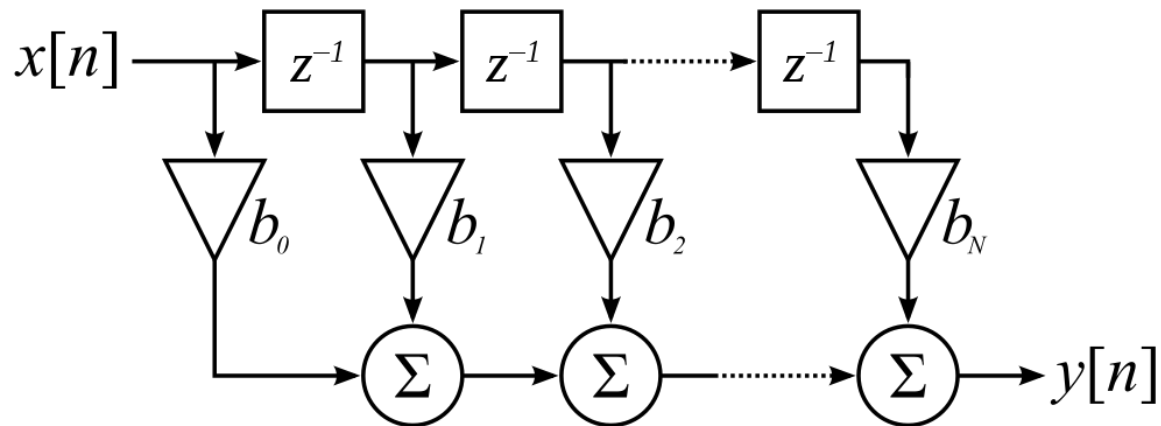


# FIR Filter SIMD Instructions

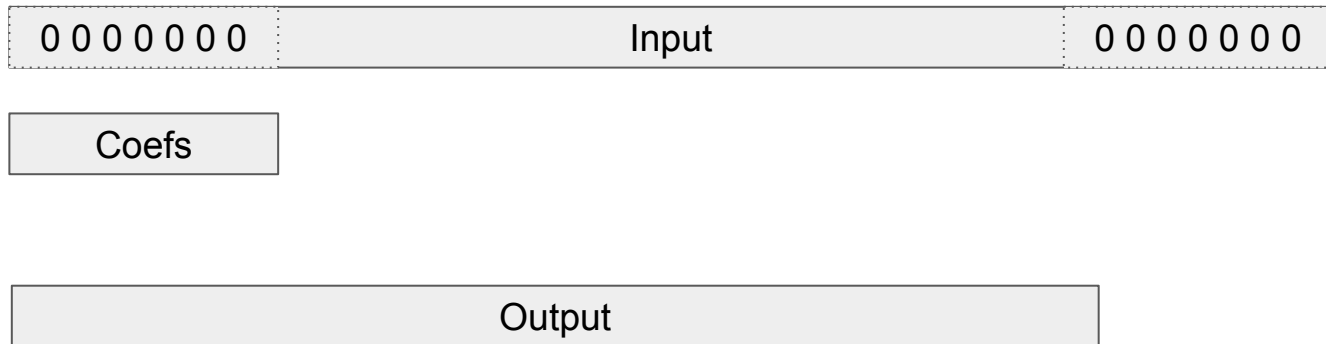
# Finite Impulse Response

Simple 1D convolution

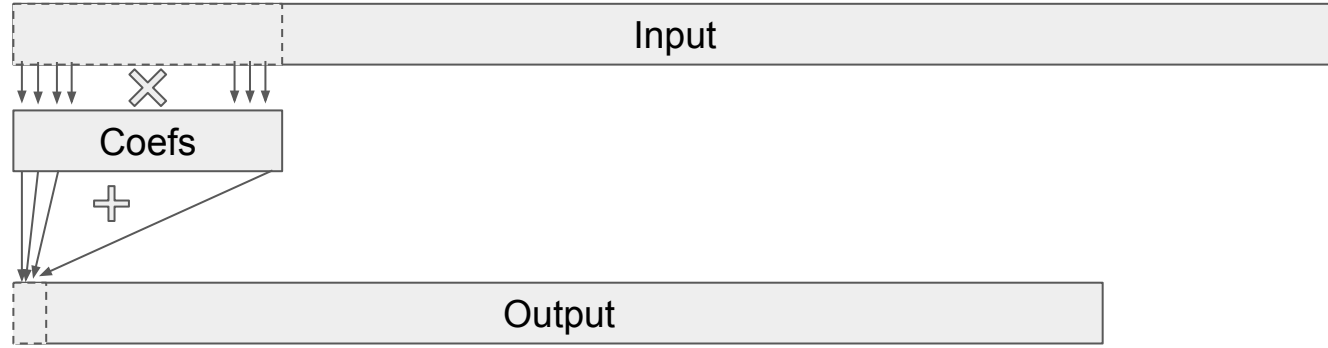


# Simple FIR filter in C++

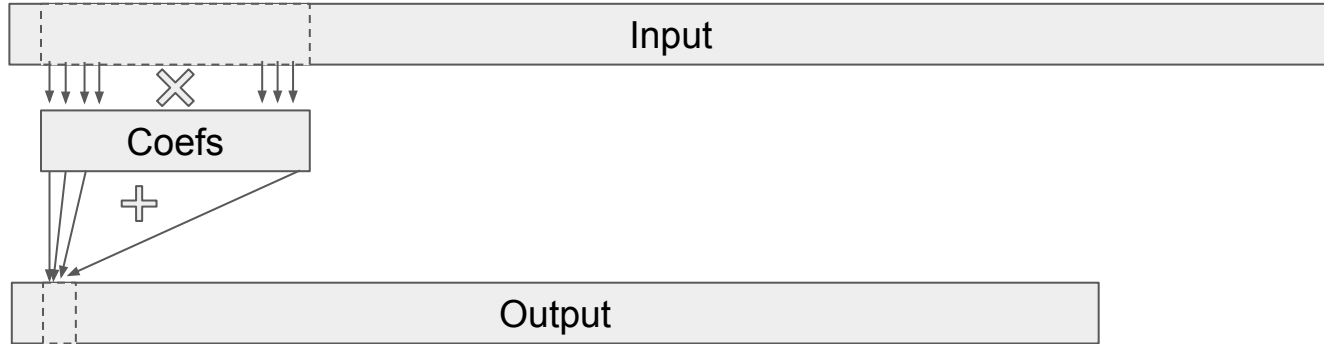
- 1 array of input data, padded with 0
- 1 array of coefficients
- 1 array for output data



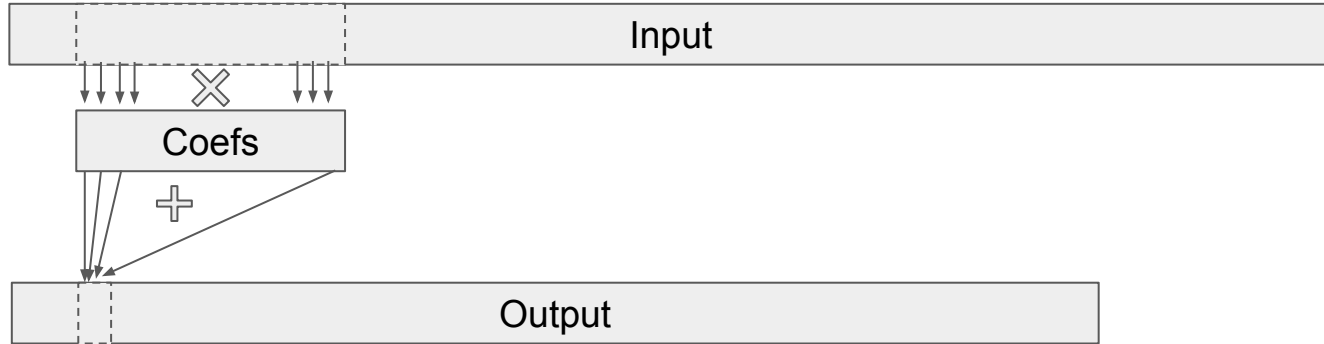
# Simple FIR filter in C++



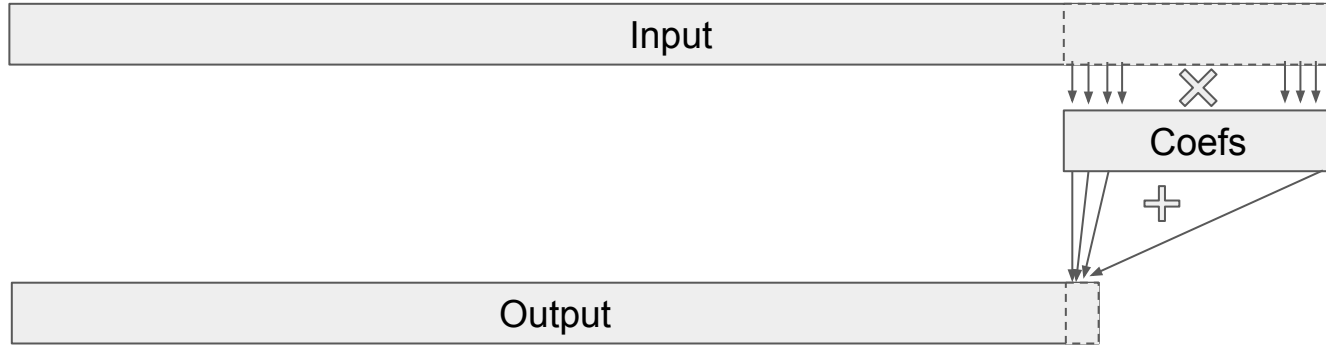
# Simple FIR filter in C++



# Simple FIR filter in C++

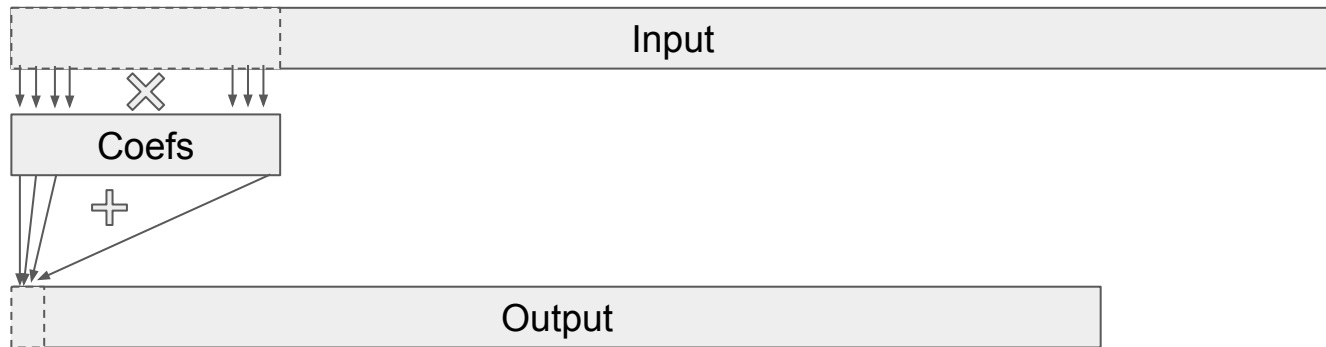


# Simple FIR filter in C++



# Simple FIR filter in C++

- Loop through (size of input - size of filter)
  - For each coefficient
    - Multiply input and accumulate
  - Store in output





# Loop unrolling

Unroll coefficient loop (inner loop) by 4:

- Manually duplicate the loop body
- Jump loop index by 4



# SIMD instructions

- Include `<arm_neon.h>`
- Add compiler flag: `-mfpu=neon`

Replace your unrolled loop body by NEON instructions:

1. Declare SIMD registers: Use 128-bits SIMD vectors
  - a. float 32-bit x 4
2. Initialize output SIMD vector with 0
3. Inside the loop:
  - a. Load input data into SIMD vector
  - b. Load coefficients into SIMD vector
  - c. Multiply-accumulate into output SIMD vector
4. Store output SIMD vector into local array
5. Add the 4 values together then store in output array