

ECE448

# Cascaded Integrator/Comb

---

Midterm for Tuesday

The ECE448 Team

3/16/2010

## Introduction

Your task is to describe in VHDL, debug, and implement a DSP circuit called Cascaded Integrator/Comb (CIC).

The CIC has two main sections: the comb section and the integrator section. Both sections consist of  $N$  stages each. The exact structure of each stage, and the way of connecting all stages together are presented below in the form of a block diagram.

## Description

The CIC is controlled by a one-cycle enable signal, `samp_i`, that is active every fourth clock cycle. This enable signal should be generated by the test bench. The output will change every clock cycle, so the corresponding output status signal, `samp_o`, should be fixed at '1'.

As you can see in the block diagram, the combs and integrators can be easily placed in their respective entities and can be replicated using a for-generate statement. The stage number should be passed as a generic into each unit. The output size for stage  $i$  is given by

$$\text{comb output size} = 16 + i$$

$$\text{integrator output size} = 16 + i - 2$$

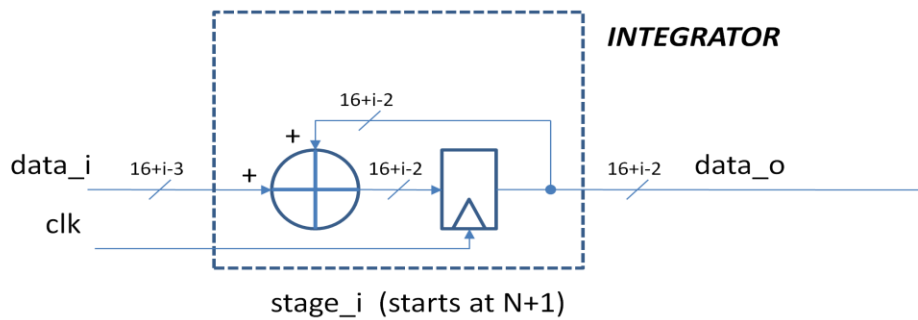
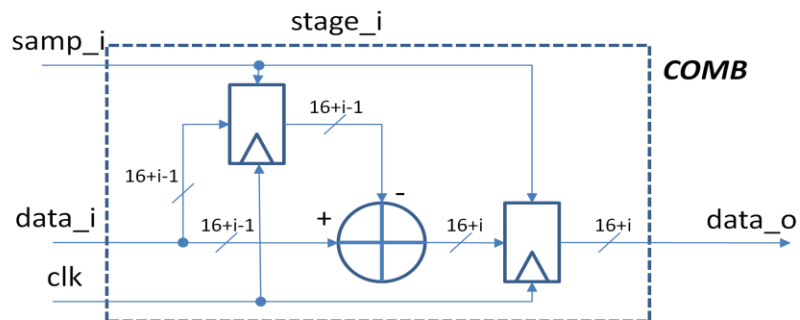
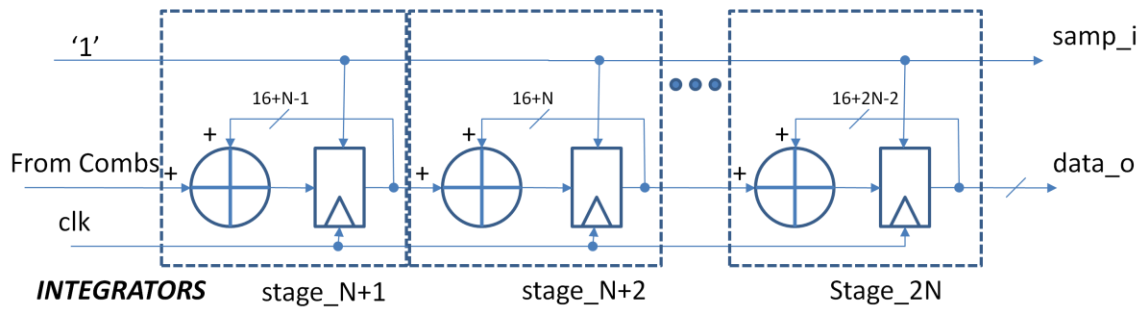
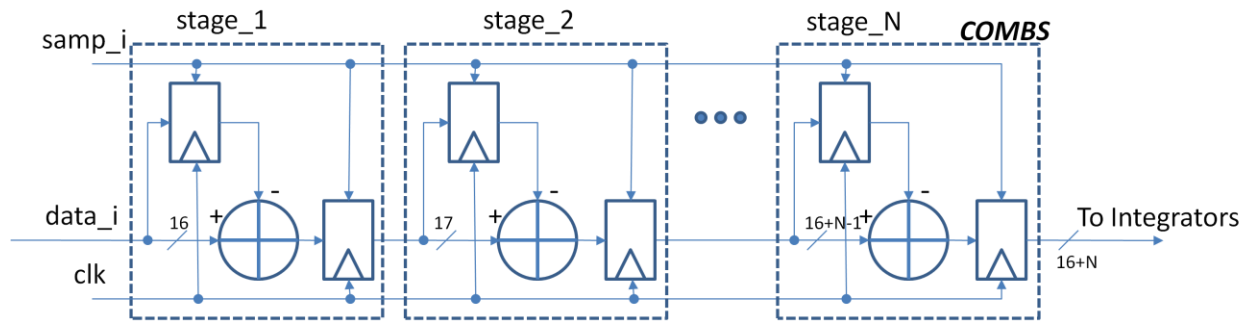
The signal used in the top level to connect the units should be an array of

```
std_logic_vector(16+2N-1 downto 0)
```

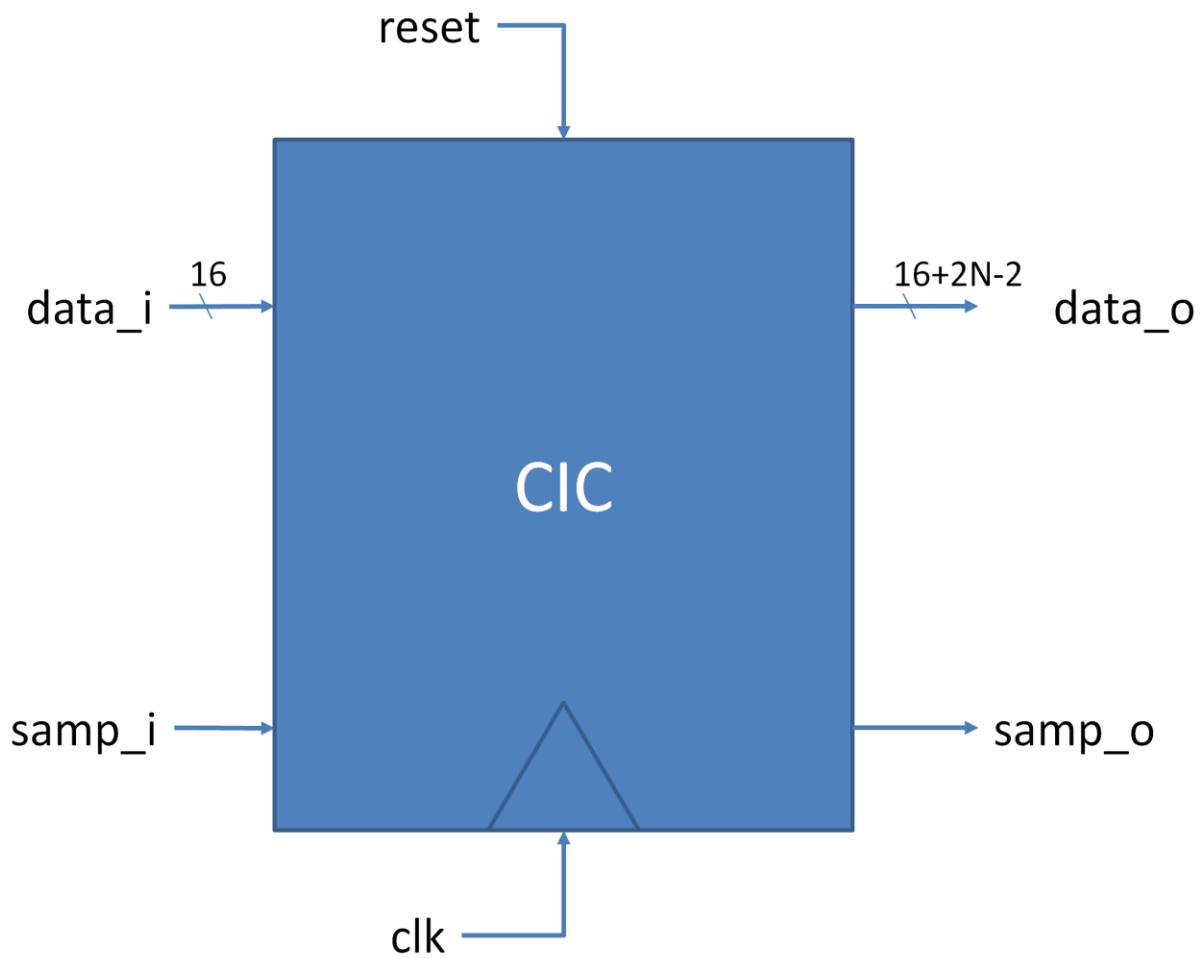
since this will be the largest output size, and consequently, the size of `data_o`.

All signals should be treated as signed, 2's complement numbers.

## Block diagram



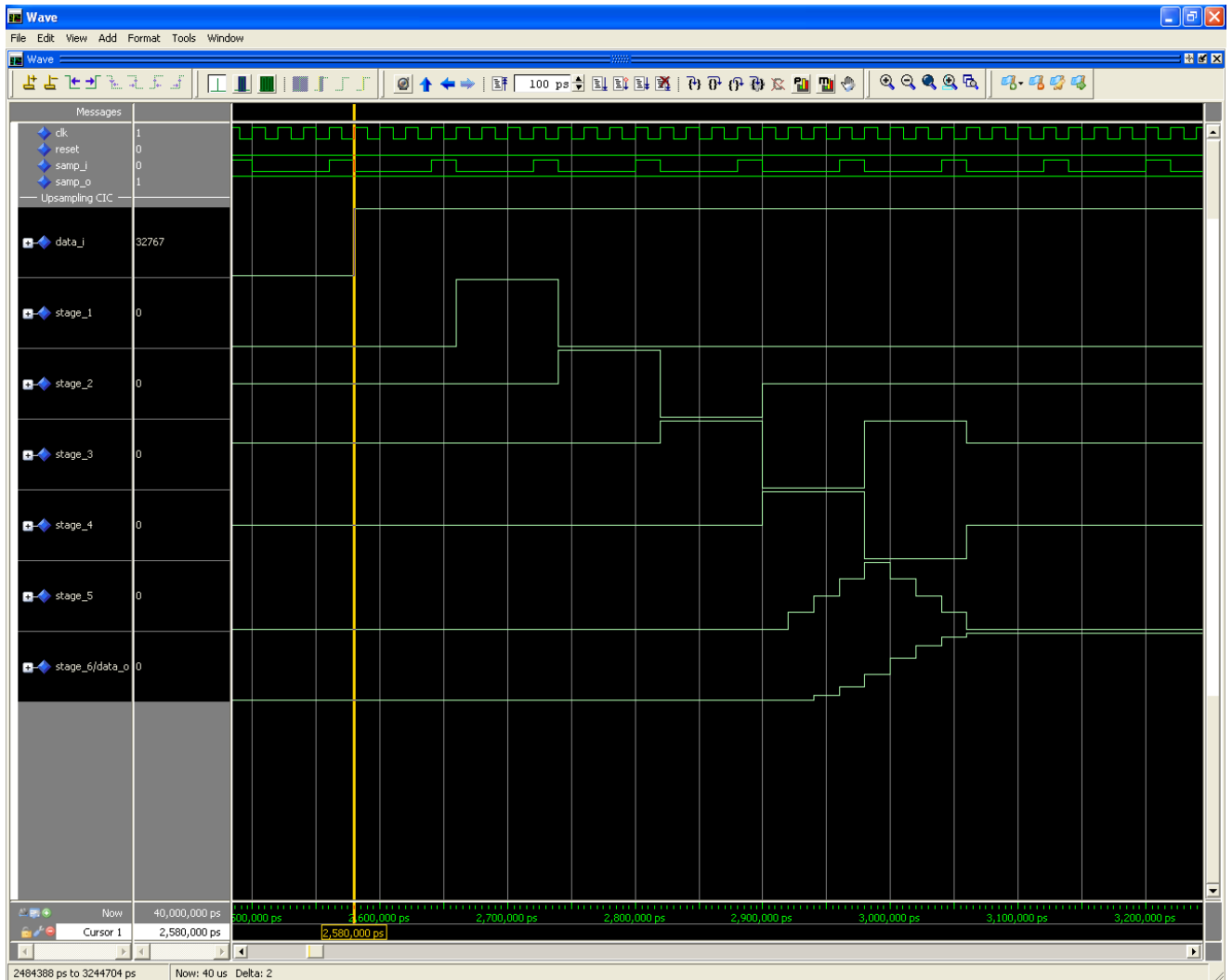
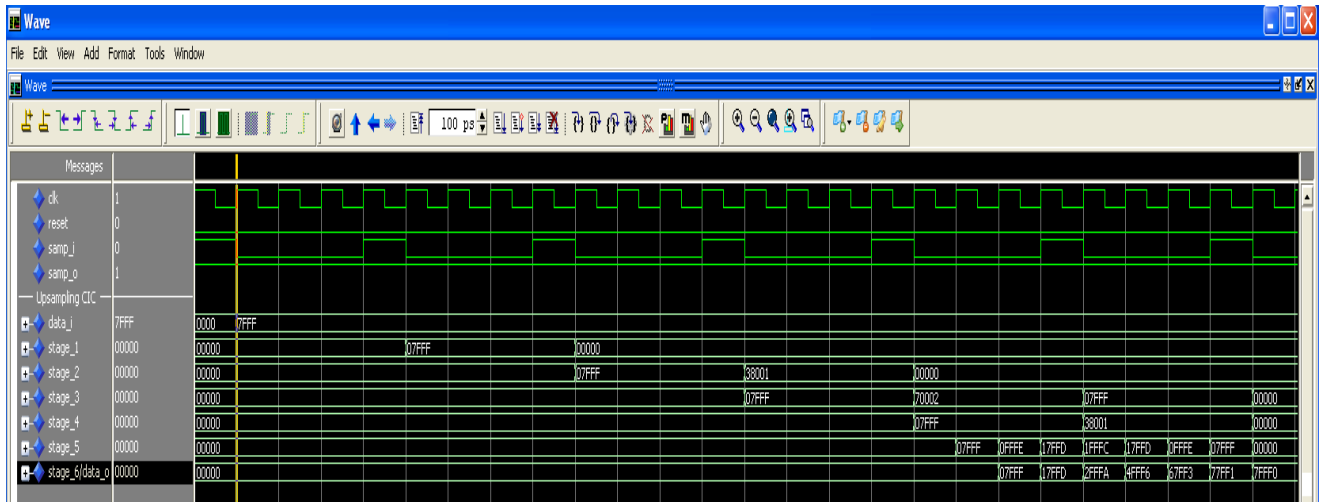
## Interface



Port name	Mode	Bits	Description
clk	in	1	50 MHz clock with 50% duty cycle
reset	in	1	Active high synchronous reset
data_i	in	16	Input data stream
samp_i	in	1	Enable signal signifying input data is ready
data_o	out	$16+2N-2$	Output data stream
samp_o	out	1	Status signal signifying output data is ready

# Waveforms

These are timing waveforms generated by the testbench for the case of N=3, and data\_i set to 0x7FFF



## Test Plan

We will be testing the CIC by checking its step response. In your test bench, you should do the following:

- 1) Generate your 50 MHz clock
- 2) Input x"0000" for 4 consecutive samples (samples are signified by samp\_i going high for one clock cycle)
- 3) Input x"7FFF" for 32 samples
- 4) Stop the simulation

Your waveform should show all internal signals. Your input and output signals (data\_i and data\_o, respectively) should be shown as an analog signal (signed, 2's complement, height=100) to aid in testing.

## Tasks

1. Write VHDL synthesizable code for the CIC filter (both integrator and combs).
2. Write a testbench verifying the operation of your circuit.
3. Perform a functional simulation of your circuit. Debug your code.
4. Synthesize your circuit using Synplify Pro or Xilinx XST. Save the RTL Schematic.
5. Implement your circuit using Xilinx ISE.
6. Perform post-synthesis simulations of your circuit using Active-HDL or ModelSim.
7. Based on the circuit block diagram and the Implementation reports, determine the most critical path in your circuit and its length.

## Deliverables

1. VHDL code of your entire circuit fulfilling the requirements specified in the Design Requirements section above.
2. VHDL code of your testbench.
3. RTL schematic of your circuit.
4. Functional simulation waveforms demonstrating the correct operation of your circuit.
5. Description of the critical path in your circuit.
6. Timing simulation waveforms demonstrating the critical path.
7. FPGA resource utilization.
8. Minimum clock period of your circuit.