A Real-time, Spiking SoC for Edge RF Inference

Jon Cowart, Zephan Enciso, Mark Horeni, and Clemens Schæfer

2023-05-14

Contents

1 Overview

We present a **real-time**, **spiking** SoC for **edge radio frequency (RF) inference** in 22 nm CMOS.

1.1 System Architecture

Figure 1 depicts the primary components of the system, which are:

1. **4x4 phased array**, which is mixed down off-chip.

2. **"Spikifier"** analog circuit, which integrates the mixed RF signal during the negative phase of a 10 MHz clock and produces an output spike if the integrated signal surpassed a threshold.

3. **Data buffers** (FIFO) for synchronization.

4. Two layers of **processing elements (PEs)**: 16 layer 1 PEs, which feed 8 layer 2 PEs.

5. **RISC-V processor**, responsible for controlling PEs via a single multicast reset signal and measuring the firing rate of the layer 2 PEs to produce a classification.
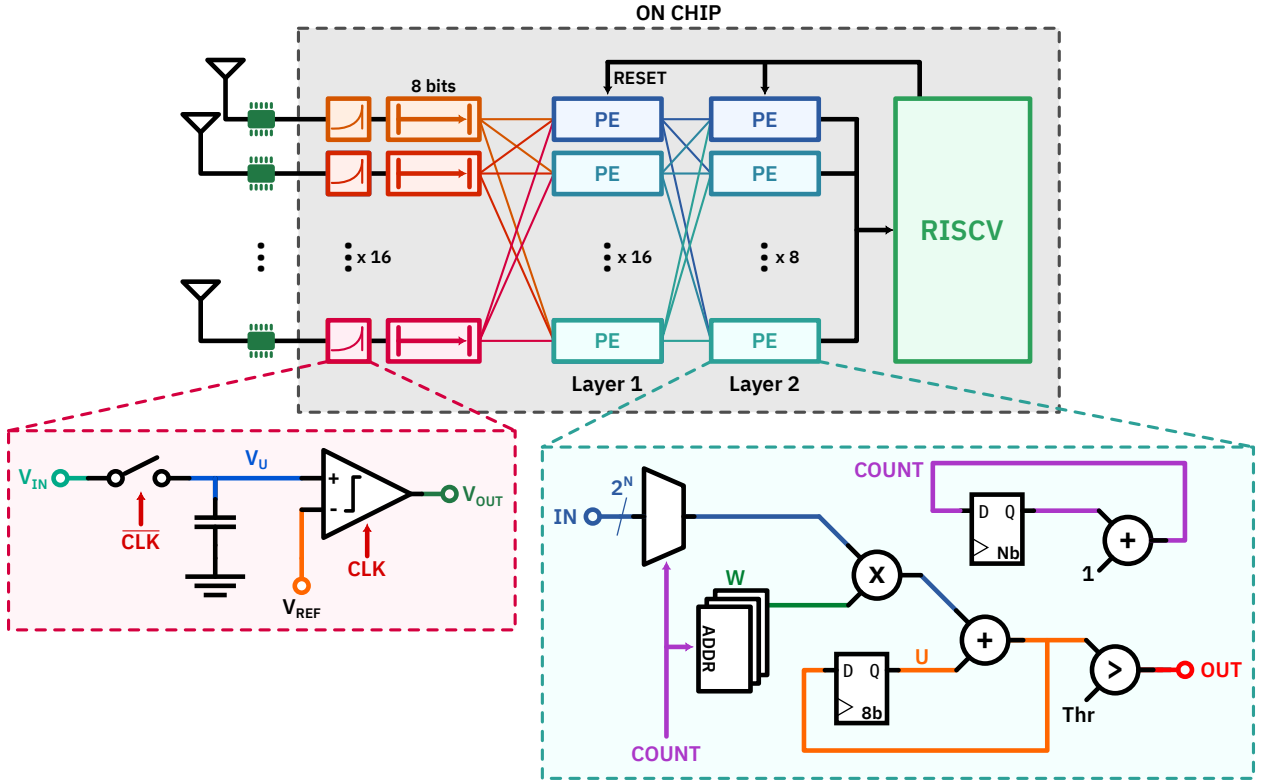

Figure 1: System architecture. Everything within the grey rectangle is on-chip. The red call-out depicts the spikifier circuit, while the teal call-out depicts the PE architecture.

1.2 Timing and Dataflow

The goal of this project is to perform **real-time** inference, which necessitates that the data consumption rate equals the data ingest rate. The system is fully pipelined (see Figure 2), and the longest single processing step is the layer 1 PEs at 16 clock cycles (the operation of the PE is described in more depth in Section 3.1). Therefore, the core clock must run at 16 times the rate of the sample clock.

The spikifiers integrate the mixed RF signal during the negative phase of a 10 MHz clock and produce an output at the positive edge. FIFOs between the spikifiers and the layer 1 PEs provide synchronization

between the 10 MHz sample clock and the 160 MHz core clock. The layer 2 PEs only require 8 clock cycles to complete their computation, after which the RISC-V processor requires 12 cycles to update its rate counts.
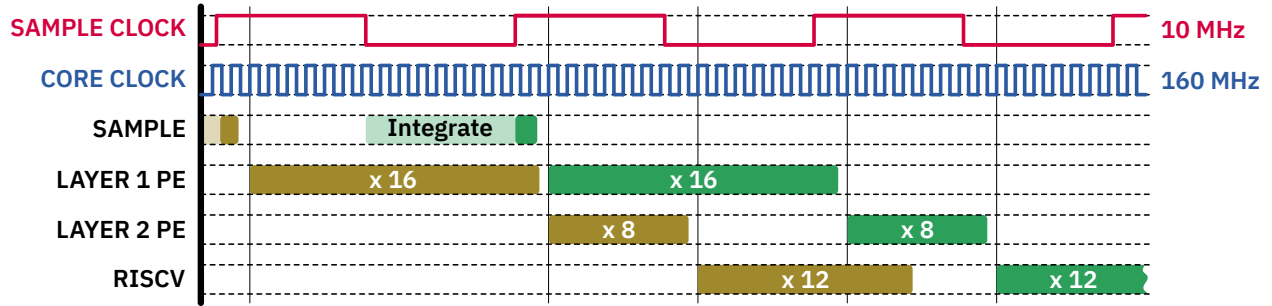

Figure 2: Pipelined system timing.
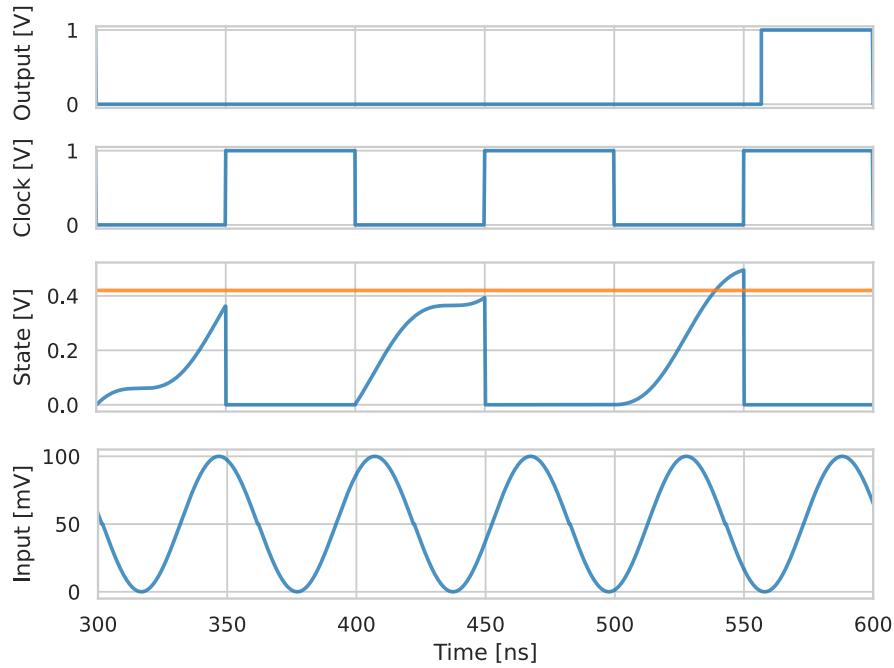
2 Analog Behavioral Model

2.1 Operation

Figure 3:

2.2 Noise Modeling

3 Digital Accelerator

3.1 Processing Element Architecture

Each PE is very simple to maximize density and energy efficiency (see Fig. 1).

3.2 Physical Implementation

4 RISC-V Control

5 Division of Labor

Group Member	Task
Jon	RTL and Validation
Zephan	System Architecture and Analog Behavioral Model
Mark	System Architecture and RISC-V computation kernel
Clemens	Synthesis and Implementation