Spyqe: A design framework for asynchronous neuromorphic design from Python to layout

Z. Shen, R. Jordans and F. Corradi

Dep. of Electrical Engineering, Eindhoven University of Technology, the Netherlands

In neuromorphic computing, sparsity plays a crucial role by enabling event-driven computation. While synchronous designs still dominate dense computation scenarios, asynchronous approaches have been increasingly explored for processing event-based data, such as speech signals and Dynamic Vision Sensor (DVS)-based vision applications. In these contexts, Spiking Neural Networks (SNNs), when implemented with asynchronous circuits, eliminate the need for continuously toggling clock signals and exploit both spatial and temporal sparsity through handshake-based communication protocols.

However, although SNNs are typically trained using high-level Python descriptions, there is still a lack of an automated framework specifically targeting asynchronous circuit implementation. Moreover, the sequential execution semantics in Python differ fundamentally from the communication-driven concurrency model used in asynchronous design, making the translation process highly nontrivial. Existing Communicating Sequential Process (CSP)-based languages, such as Communicating Hardware Processes (CHP) [1] and Balsa [2], describe systems from a signal communication perspective, which differs significantly from the sequential dataflow representation commonly used in Python and therefore often requires substantial manual translation effort. To address this challenge, we propose a framework supported by a pre-designed library of Quasi-Delay-Insensitive (QDI) components. The proposed framework enables an automated translation flow from high-level Python-based descriptions of Spiking Neural Networks (SNNs) to asynchronous GDS layouts. It is compatible across different PDKs, allowing designers, especially those without extensive expertise in asynchronous design methodologies, to effectively exploit the event-driven nature of spike-based computation in circuit design.

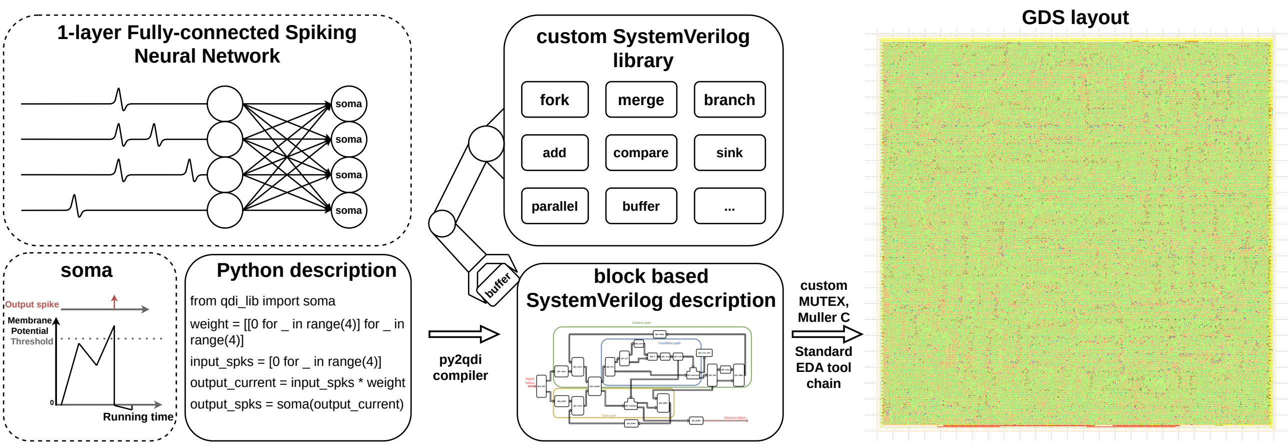

Figure 1: Abstract flow for a 4x4 fully-connected layer from Python to GDS layout with Sqyqe framework

[1] S. Ataei et al., IEEE Des. Test, 38(2), 27–37, 2021.

[2] D. Edwards and A. Bardsley, Comput. J., 45(1), 12–18, 2002.