Contribution ID: **163** Type: **Poster**

Verification Methodology for OBELIX, the monolithic active pixel sensor for the proposed upgrade of the Belle II vertex detector at SuperKEKB

Thursday 3 October 2024 17:20 (20 minutes)

This contribution presents the methodology and verification strategy for OBELIX, the monolithic active pixel CMOS sensor designed for the proposed upgrade of the Belle II vertex detector. Leveraging a dual verification approach with cocotb and UVM, we ensure the integrity of OBELIX's digital logic. This methodology addresses the complexities of ASIC design, which includes an 896x464 pixel matrix with 7 bits Time-over-Threshold (ToT) resolution per pixel and implements a trigger logic with two stages of memory. Through comparative analysis, we enhance error detection during the design phase, thereby bolstering confidence in OBELIX's functionality.

Summary (500 words)

In the realm of ASIC design, the burgeoning complexity necessitates a robust methodology to streamline the development process. Our proposed methodology aims to identify discrepancies early on, minimizing post-production iterations and ensuring seamless integration into the experimental environment.
The Optimized Belle II monolithic pixel sensor (OBELIX) stands as the premier candidate to replace the existing sensor in the VTX layers for a potential Belle II detector upgrade at Japan's SuperKEKB accelerator within three years. OBELIX architecture has been detailed in [1], this contribution, instead, depicts our strategy for establishing a verification environment for the initial prototype (OBELIX-1) and leveraging its inherent reusability to validate its successor, OBELIX-2, slated for deployment in the experiment.
The digital logic to be validated contains: the implementation of a subset of the RD53B serial input protocol, a synchronization block alongside a command parser unit that allows access to the 105 monitor and configuration registers, a low-latency serial output, a trigger unit with two stages of memory (a pre-trigger FIFO to store up to 128 hits per trigger and a triggerable memory block), one serial output connection with 8b10b encoding, and an 896x464 pixel matrix. Each pixel has a 7-bit register for both the leading and trailing edges of the discriminator output.
Our verification schema employs two independent approaches to ensure the digital logic of the ASIC behaves as intended: Firstly, a co-routine-based cosimulation testbench (cocotb), developed in Python, ensures the functionality of each block from the ground up. This approach, developed in tandem with block development, offers comprehensive validation. Secondly, we employed the Universal Verification Methodology (UVM), an IEEE standard implemented in SystemVerilog, to orchestrate stimuli from the top of the design. Implemented in a design-agnostic manner, this methodology commences with a verification plan outlining OBELIX features to be scrutinized. Access to inner blocks of the design is facilitated from the top module, preserving the original structure and focusing on blocks directly influencing the chip control and configuration path.
In addition to ensuring the quality of individual feature implementations, our methodology allows for the definition of system-level behavior, spanning from configuration to data collection and transmission processes, aiming to reproduce the dynamic processes expected during operation.
By adopting a dual verification approach, we foster a comparative analysis of results, enhancing error detection during the design phase. Code and functional coverage metrics are utilized to track the verification procedure's completeness.

[1] H. Pham, et al., Design of the OBELIX monolithic CMOS3 Pixel Sensor for an upgrade of the Belle II vertex detector (2023). Submitted to Journal of Instrumentation as proceedings to TWEPP 2023.

Authors: FEDERICI, Luca (IPHC); Dr FEDERICI, Luca (CNRS-IPHC Strasbourg)

Presenters: FEDERICI, Luca (IPHC); Dr FEDERICI, Luca (CNRS-IPHC Strasbourg)

Session Classification: Thursday posters session

Track Classification: Programmable Logic, Design and Verification Tools and Methods