**Background: Clock Domain Crossing**

Most modern systems consist of several clock domains whose components run in relative asynchrony and need to function correctly. Verifying that Clock Domain Crossing (CDC) logic works correctly is a daunting task that is complicated by the inability of digital simulators to reproduce some of the problematic analog behaviour of flip-flops at asynchronous interfaces (e.g. setup/hold time violations, metastability, prolonged clk-to-q delays, non-deterministic crossover bit values and metastable flip-flop outputs and propagation of metastability between flip-flops).

**Tool Description**

We present a first prototype of a gate-level tool that enables straight-forward and intuitive verification of multi-clock designs. The tool's underlying methodology (described in the paper "Formal Verification of Clock Domain Crossing using Gate-level Models of Metastable Flip-Flops" to be presented in the conference) relies on substituting flip-flops with specialized models and applying morphological circuit transformations to enable an input netlist to reproduce problematic CDC behaviour digitally. By modelling these effects at the gate-level the tool enables the designer to leverage the rigour of formal verification to identify CDC issues and debug them in an intuitive fashion (faults can be observed and studied using counter-example waveforms). The main advantages of this approach are:

1. It reports fewer false positives
2. It is applicable to non-stereotypical designs
3. It can determine failure consequences
4. It can demonstrate failures in signal waveforms
5. It requires no input from the designer about what CDC design patterns are used or how the interface is supposed to work

**Demo**

The tool is an early stage of development but consists of a functional Verilog parser and CDC transformation functions that can be invoked programmatically. The demo will showcase the tool and demonstrate its usage using simple sender-receiver circuits. Visitors will have the opportunity to examine the tool's inner working and discuss the advantages/limitations of the proposed verification approach with the main developer.