SafeTI Traffic Injector Enhancement for Effective Interference Testing in Critical Real-Time Systems

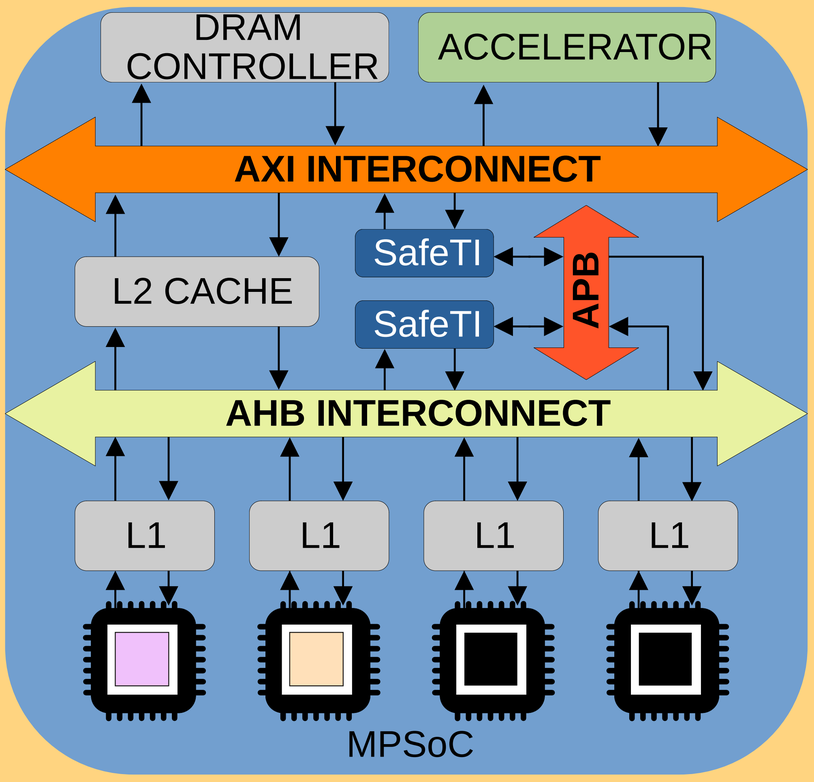
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

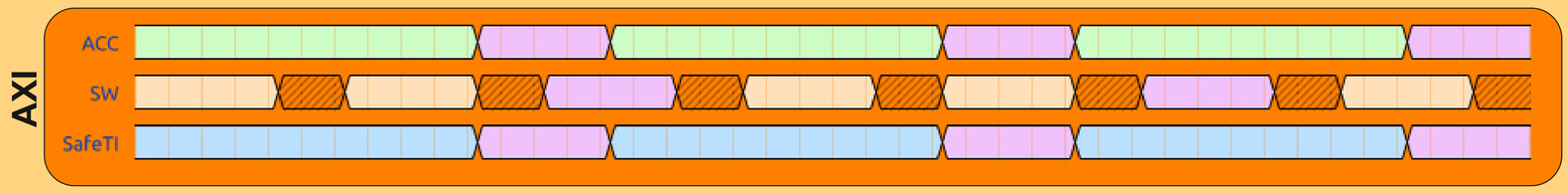
Francisco Fuentes
francisco.fuentes@bsc.es

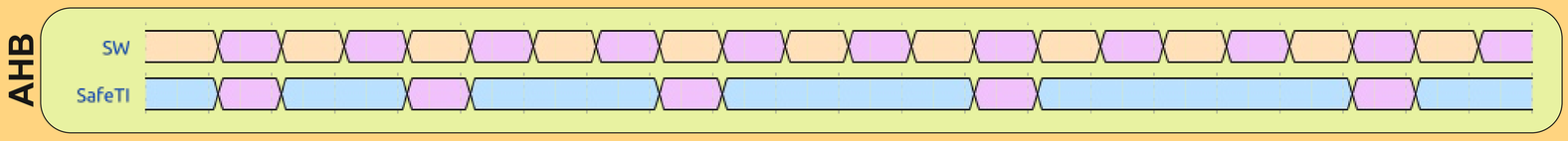
Universitat Autònoma de Barcelona
Escola d'Enginyeria

Jaume Abella, Sergi Alcaide, Raimon Casanova

Introduction & Motivation



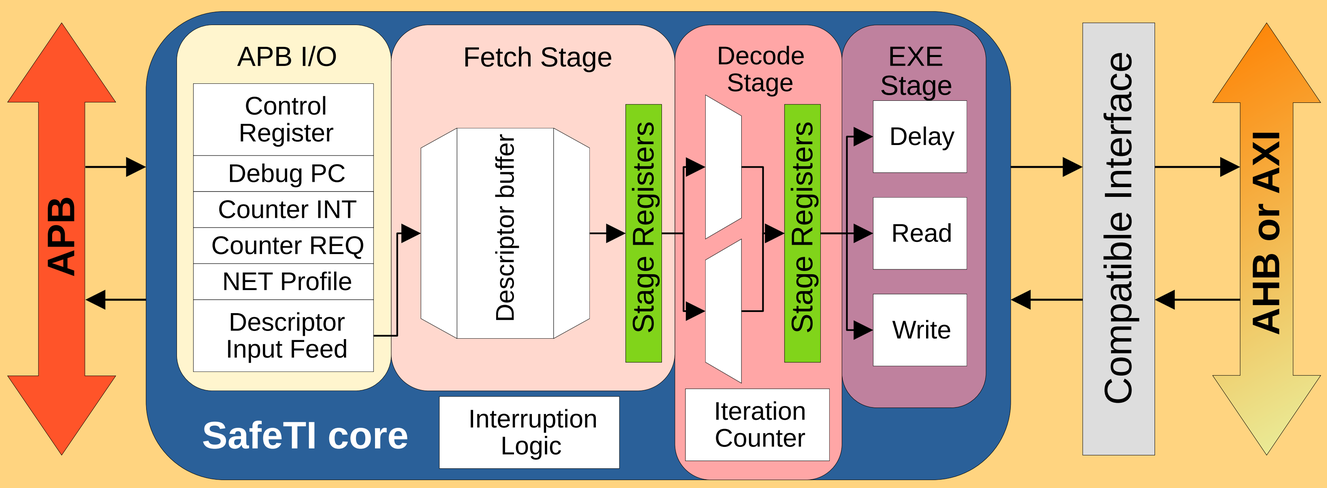




CHALLENGE: Software-only solutions are unadequate for full validation of **safety-critical high-performance MPSoC platform interconnects** at HW level.

SOLUTION: SafeTI, a programmable, flexible and integrable traffic injector, allowing extended traffic generation capabilities at any platform level.

SafeTI Features



- Read, Write and Delay descriptors for building the desired traffic injection pattern, each with unique programmable parameters such as initial address injection, access size, etc.
- Dynamic descriptor word length for optimized descriptor buffer usage.

- Three-stage pipeline topology for increased frequency operation and traffic generation throughput at interface.
- Flexible modular interface to target interconnects based on any other protocol given a compatible interface.
- Full control, configuration and programming through a single APB Subordinate integrated within SafeTI core.
- Available interruption and request counters, and debug registers.
- APB interruption options upon traffic pattern completion, descriptor completion or/and network error.
- Automatic disable options upon interruption or interconnect error, and operation option for looping traffic pattern generation.
- Traffic pattern descriptors loaded from a single address location, where they are moved to the descriptor buffer in order of execution.

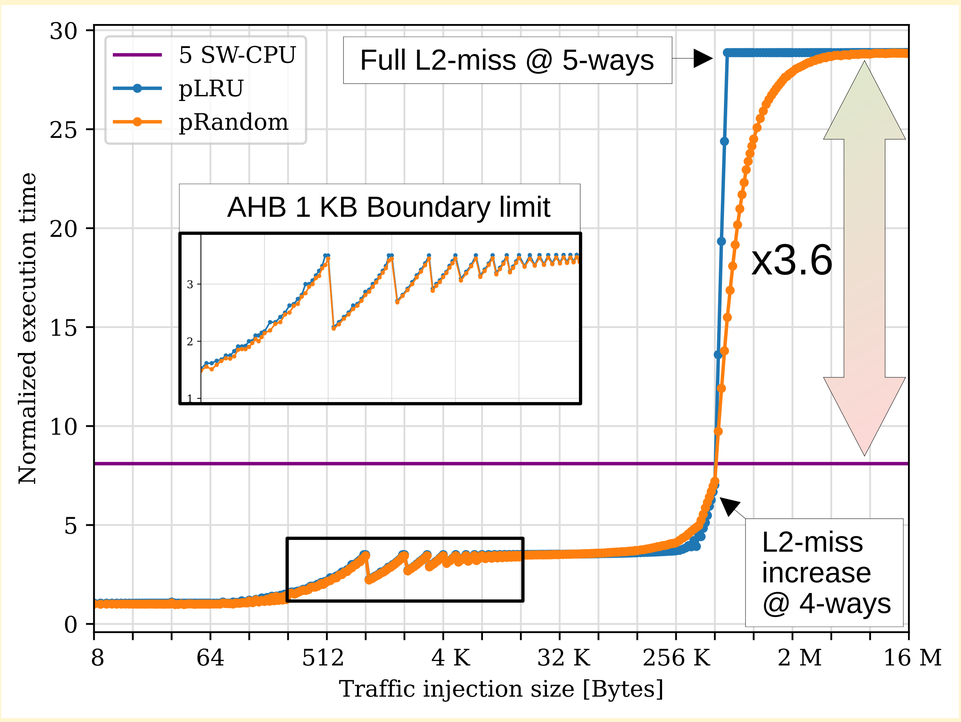
Possible Applications & Initial Results

Opportunities for SoC validation:

- **Interconnect components**
- **Cache coherency**
- **High-contention stress tests**

Other applications, traffic injection for aliveness testing or contention sensing (e.g., "ping"-like), traffic injection to counteract attacks for security purposes, programmable partial or full cache flushing, etc.

Case study: L2 cache characterization using SafeTI traffic injection from the AHB interconnect and a single CPU µBenchmark



- Clear identification of cache specifications.
- Capable of studying different replacement policies and their caching advantages.
- Limited to the targeted interconnect protocol, allowing to study specific behaviors.
- Contention capacity over x3.6 times the achieved by 5 SW contender cores.

Acknowledgments & Open Source

This work has been partially supported by the Spanish Ministry of Science and Innovation under grant PID2019-107255GB-C21 funded by MCIN/AEI/10.13039/501100011033.

A public version of the SafeTI source design files in VHDL and drivers is available at its gitlab repository, reachable using this QR code.