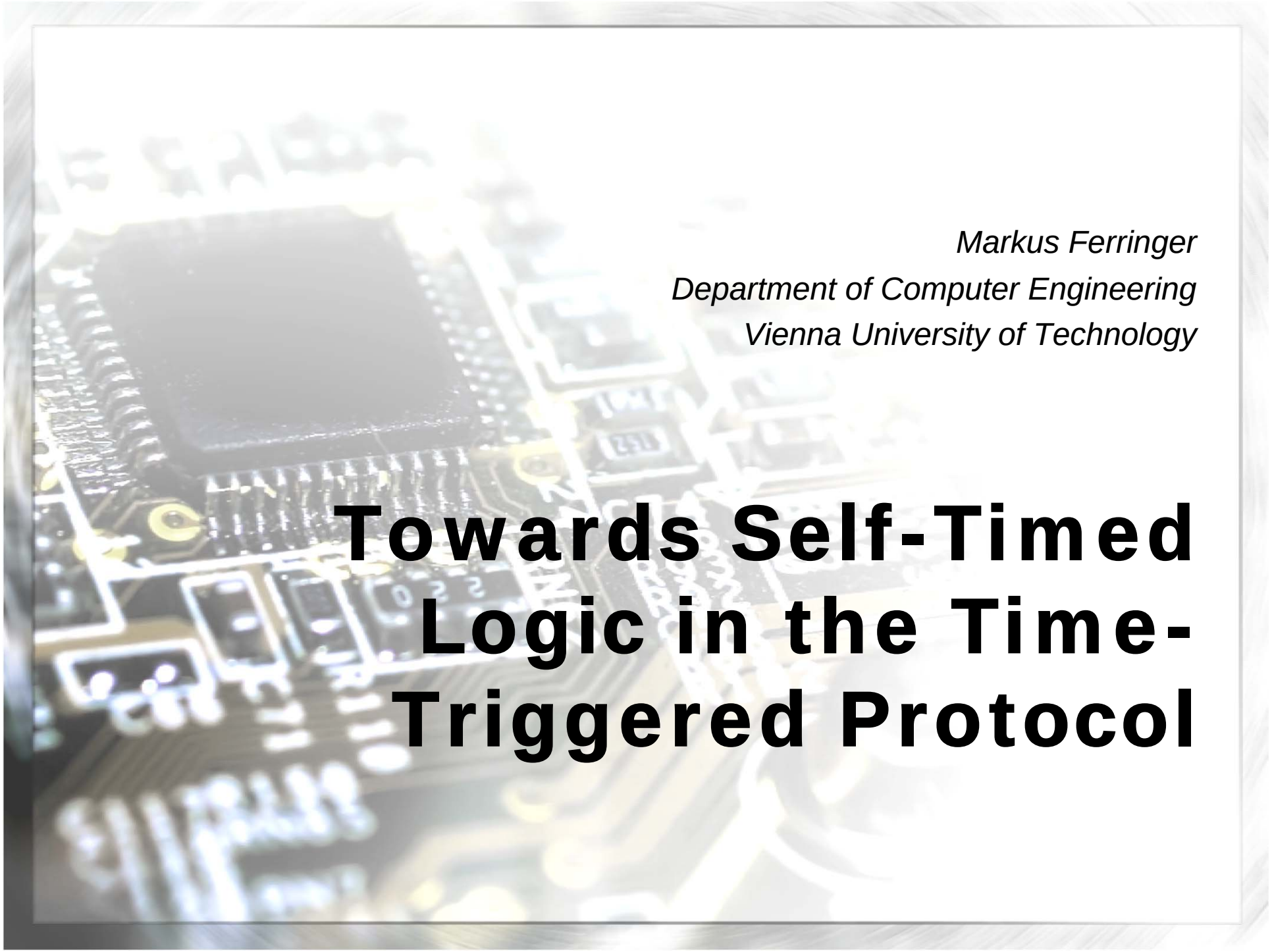


*Markus Ferringer*
*Department of Computer Engineering*
*Vienna University of Technology*

# Towards Self-Timed Logic in the Time-Triggered Protocol

# Overview

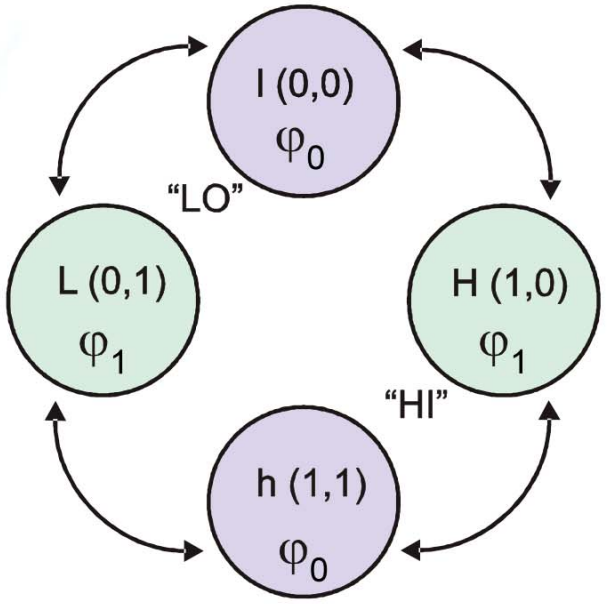
- Introduction
  - Project ARTS
  - Asynchronous design
  - Time-Triggered Protocol
  - System setup and architecture
- Time-Reference Generation
  - Design alternatives
  - Comparison
- Experimental results
  - Jitter and precision
- Summary

# Project ARTS

- Asynchronous Logic in Real-Time Systems
- Investigation of temporal predictability of asynchronous logic (QDI designs)
  - Data-dependent vs. random jitter
  - Changing operating conditions
- Case study: Asynchronous TTP controller
  - Temperature / supply voltage influence delays
  - Dynamic adaption to varying execution speeds
  - Challenge: Accurate notion of time

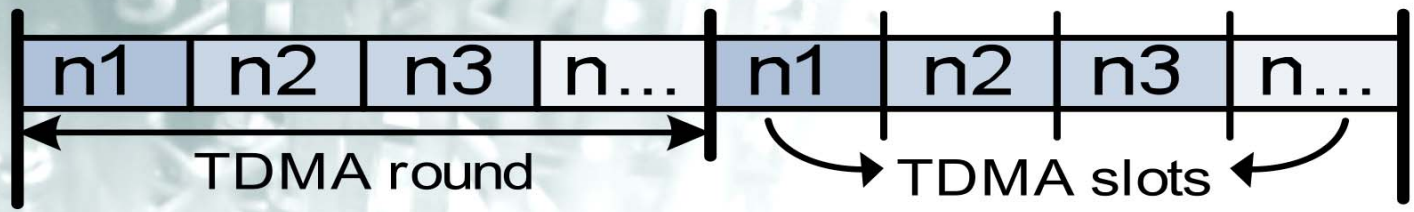
# Asynchronous Design

- Level-Encoded Dual-Rail (LEDR)
  - Phased Logic: Two code sets
  - Exactly one rail changes (2-phase protocol)
  - Simple completion detection
- (Quasi-)Delay Insensitive
  - Signal delays unconstrained
  - Timing assumptions hidden inside basic building blocks



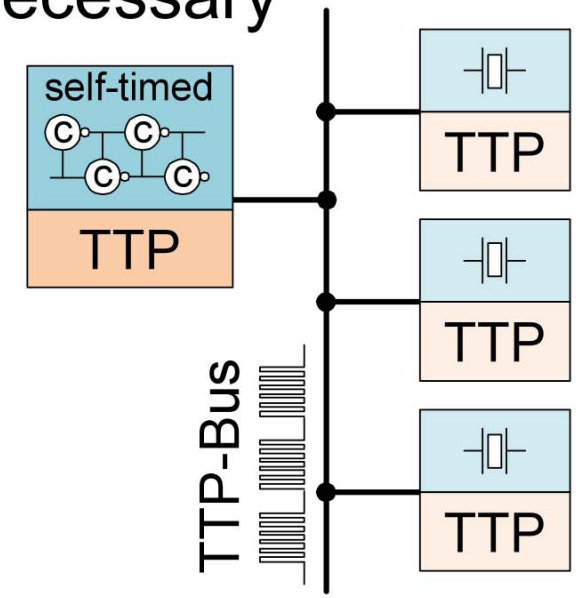
# The Time-Triggered Protocol

- Highly reliable communication protocol for hard real-time systems
  - Fault-tolerance, consistency, dependability
- Exploits a-priori system knowledge
  - Static message schedule
  - Time Division Multiple Access (TDMA)
  - Membership, global time, consistency, ...

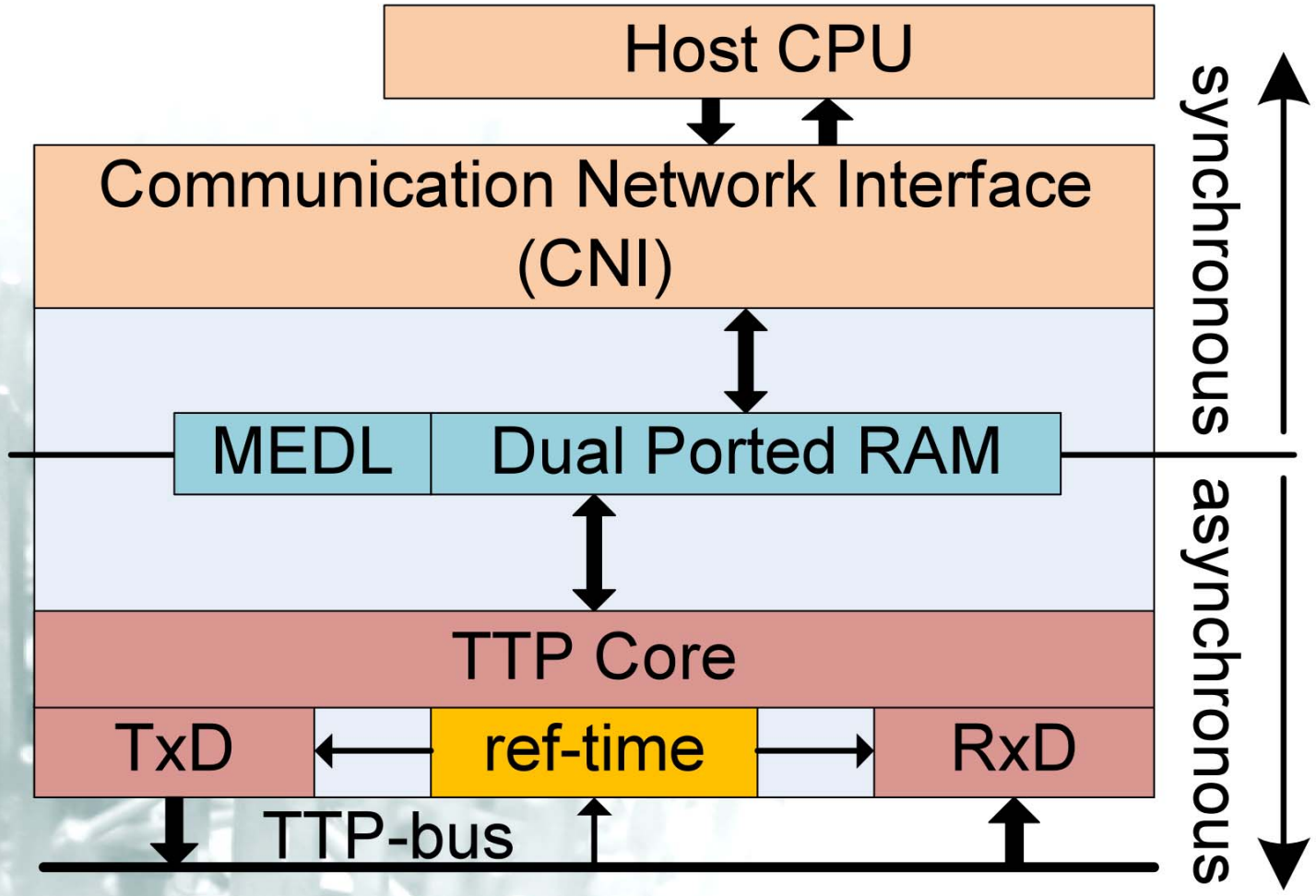


# System Setup

- One asynchronous TTP controller embedded in synchronous system
- Notion of time
  - Time reference derived from bit stream
  - Continuous resynchronization necessary
  - Known bit rate on bus
- Manchester coding
  - Data clock encoded
  - At least 1 transition per bit

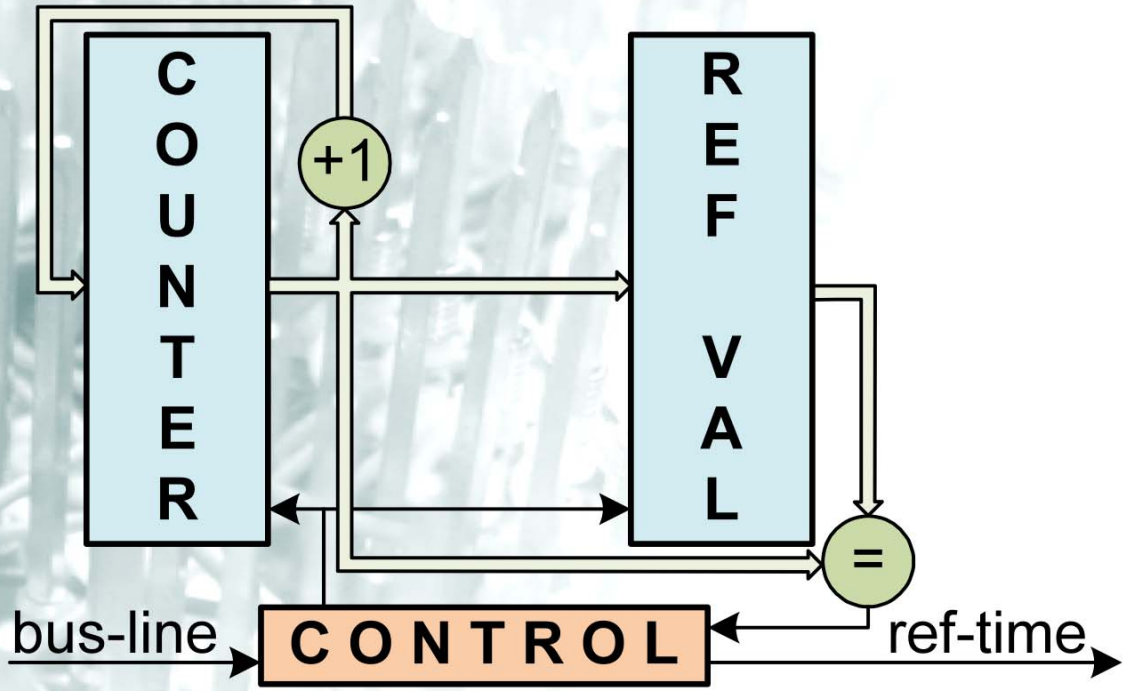


# System Setup



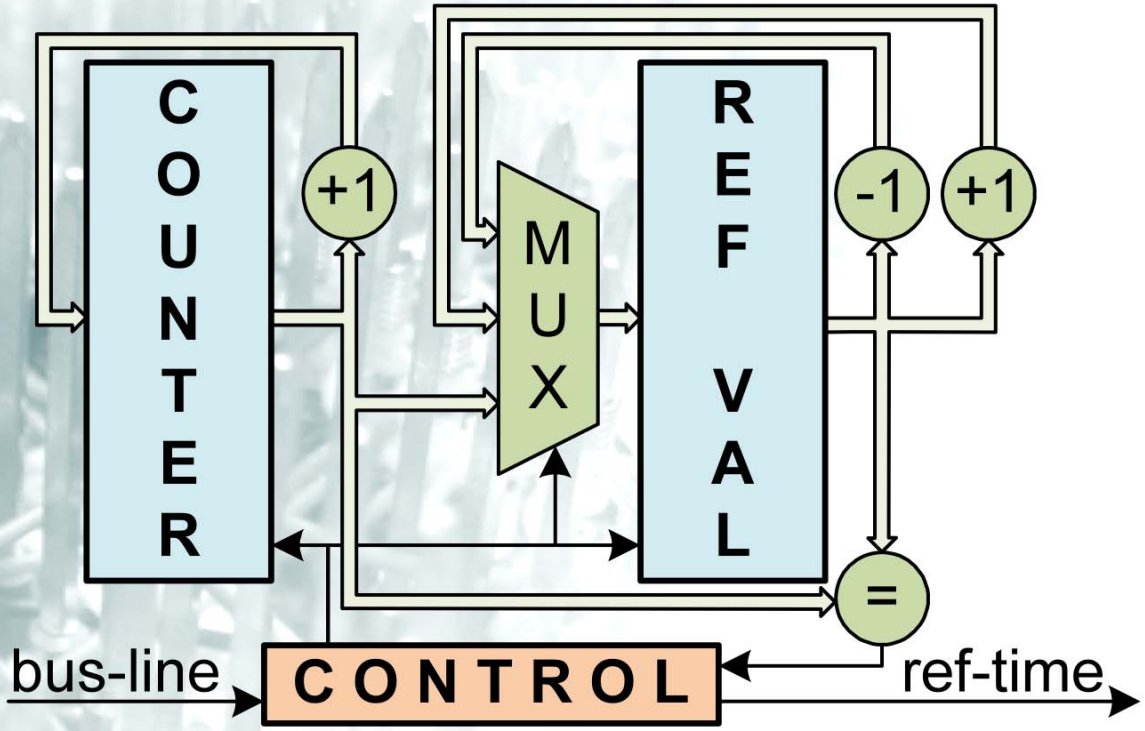
# Time Reference - Basic

- Free-running, asynchronous counter
- Reproduce duration by counting to ref-val
- Continuous adjustment of ref-val one per frame



# Time Reference - Advanced

- Additional rate correction per received bit
- Better accuracy: Less quantization error

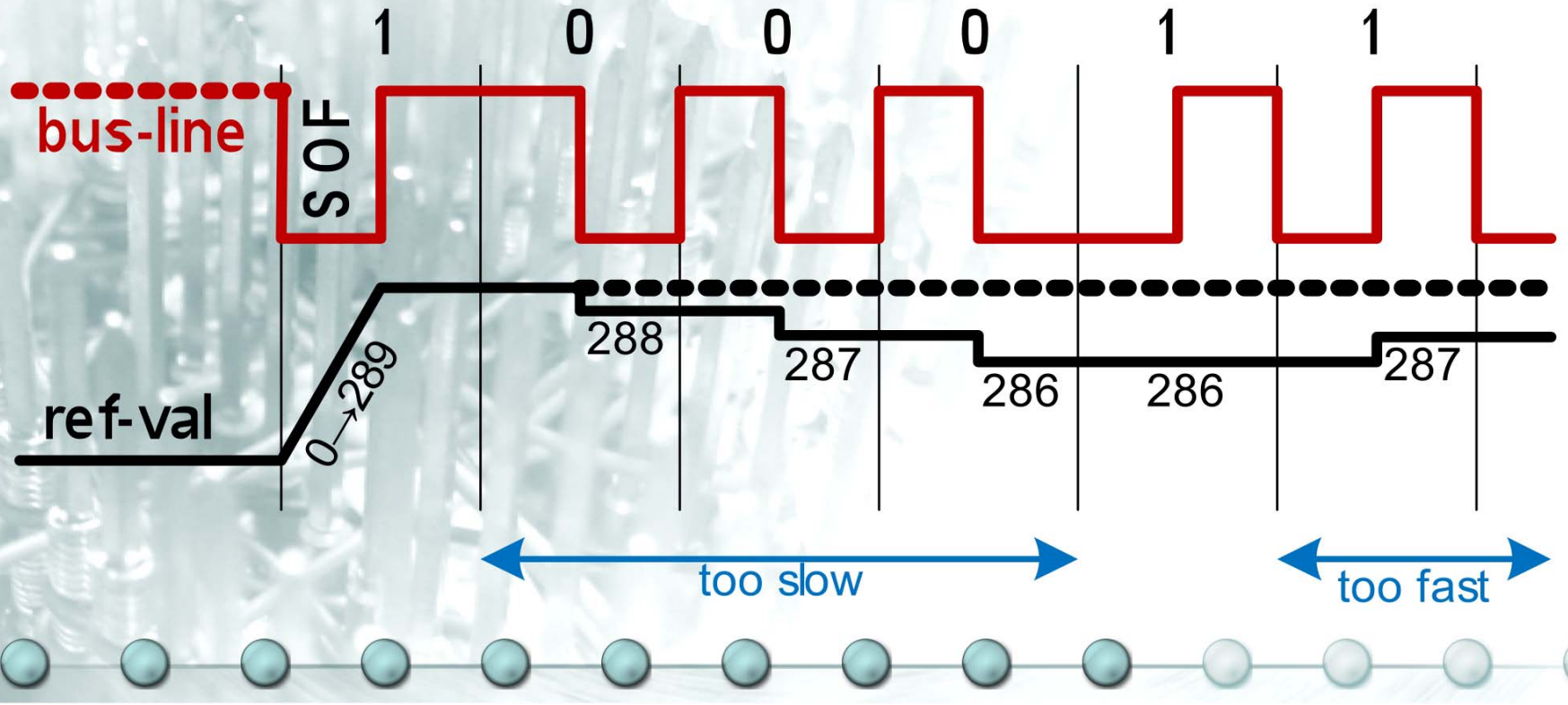


# Time Reference - LFSR

- Incrementer/Decrementer replaced by LFSR
  - Changing shift direction equals “counting down”
- Area and performance efficient
  - Only one XOR gate for a full period 15-bit LFSR
  - Propagation delay: One gate equivalent
  - Monotonic order of counting states not necessary for our purposes

# Time Reference - Example

- Basic (dashed) vs. advanced counter design
- Early transition: Decrease ref-val (too slow)
- Late transition: Increase ref-val (too fast)



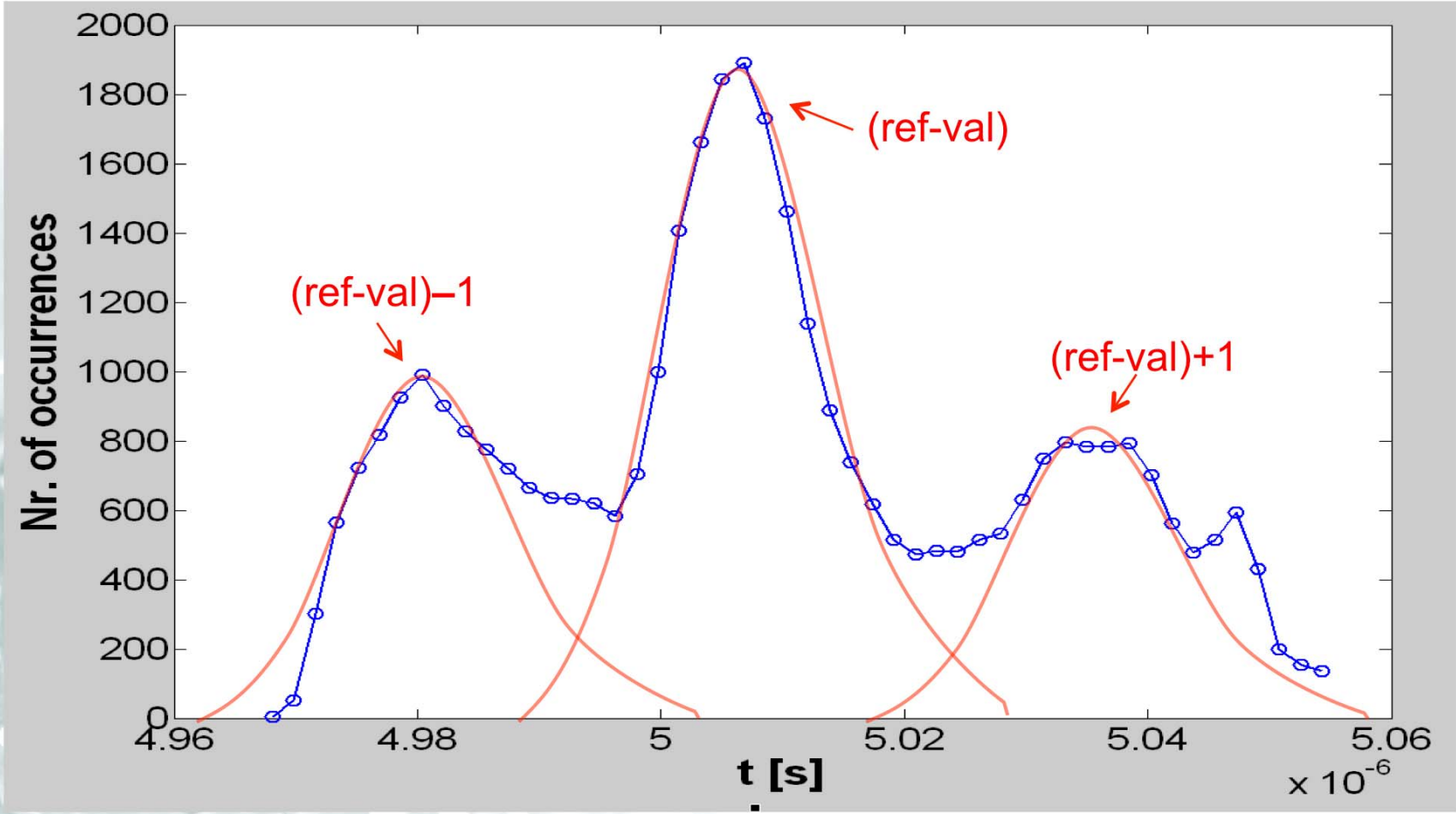
# Time Reference - Comparison

| | Counter Advanced | | LFSR | |
|---|---|---|---|---|
| Gates | 381 | (100%) | 230 | ( 60%) |
| Registers | 52 | (100%) | 42 | ( 80%) |
| Performance | 25ns | (100%) | 18ns | ( 72%) |
| Logic Depth | 7 | (100%) | 6 | ( 85%) |
| Freq. Deviation | 758ps | (100%) | 404ps | ( 53%) |

- LFSR compared to Counter Advanced
  - Area efficient
  - High performance, lower logic depth
  - Low complexity => low frequency jitter

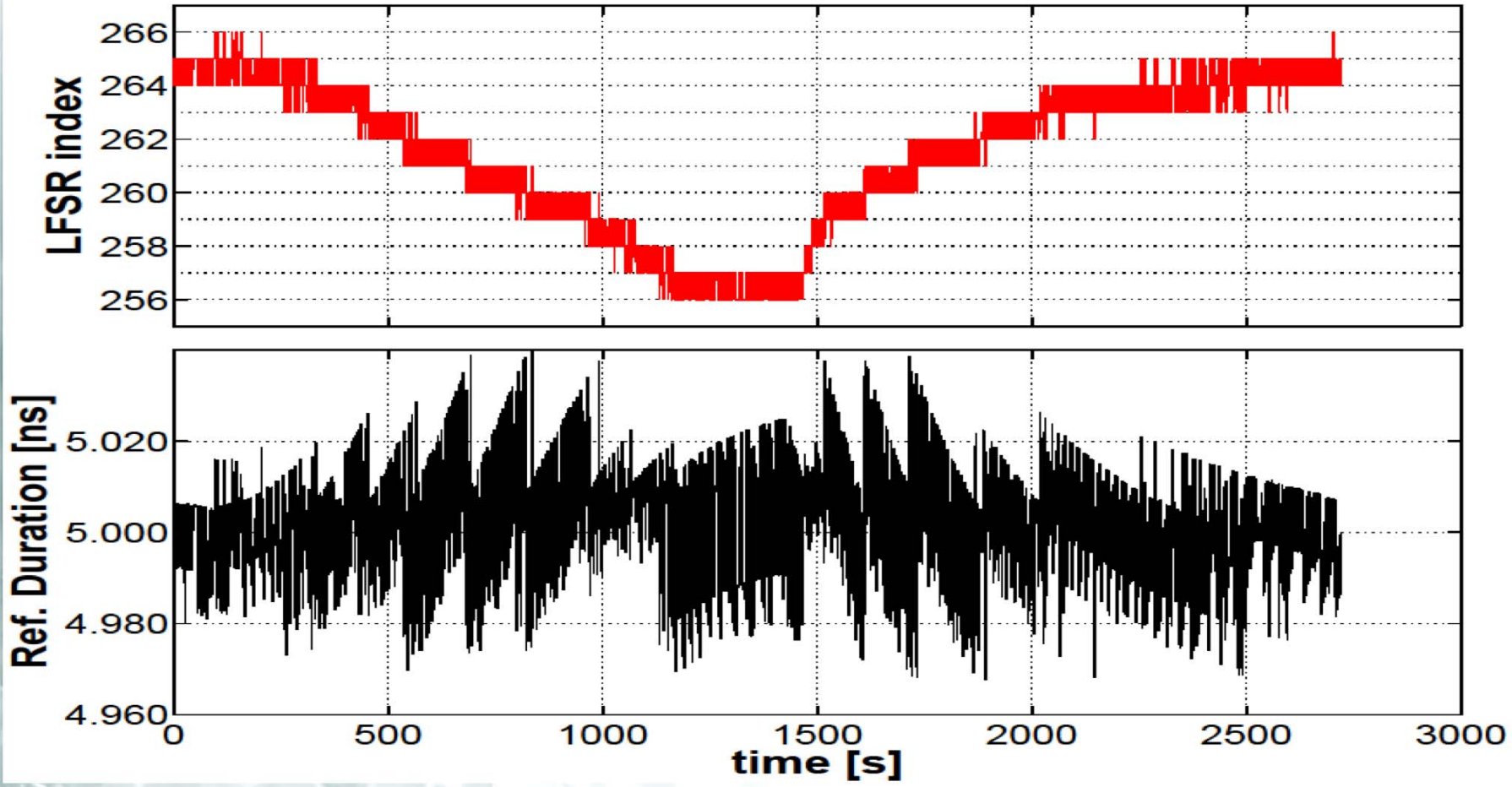
# Time Reference - Jitter

## Counter: Jitter characteristic



# Time Reference - Measurement

## LFSR: Temperature Tests (25 to 83 to 25°C)



# Summary

- Time-Reference Generation
  - Basic counter (measure SOF)
  - Additional rate correction per bit
  - Replace counters with LFSR
- LFSR: Area and performance efficient
- Precision
  - Jitter, quantization error
- Automatic adaption to changing conditions
  - Temperature, Voltage