# R&D for X-Gold SDRxx platform

**Ulrich Ramacher**

**Head of System Engineering**

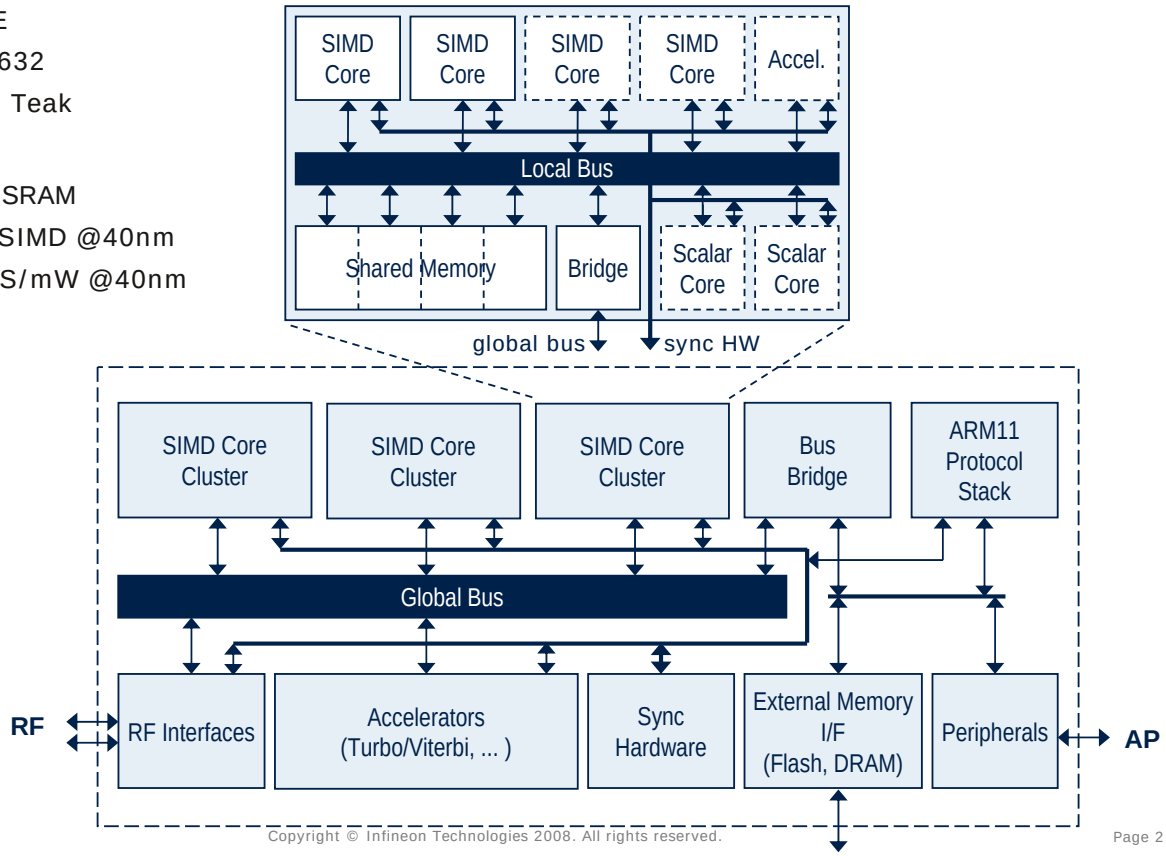
**COM SDR**

Prof. Dr. Ramacher
COM SDR SE
MPSOC 2008

Never stop thinking

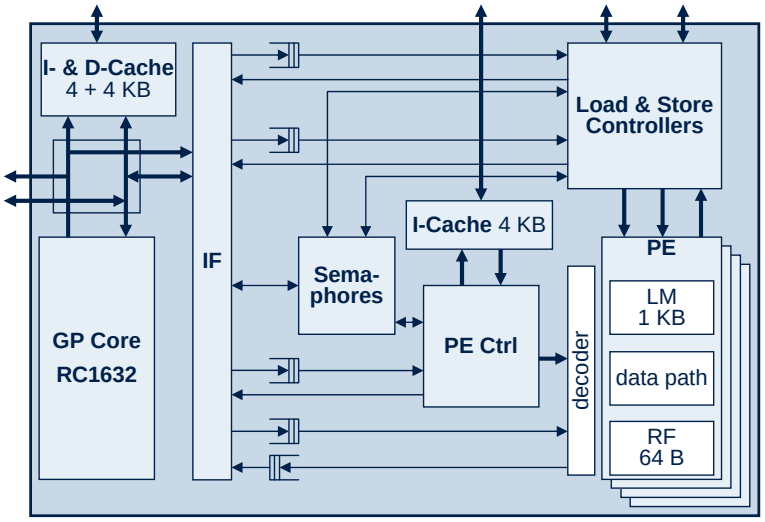
## X-GOLD™SDR2x Block Diagram

12x4 PE
21 RC1632
ARM11, Teak
~25mg
3.8 MB SRAM
12mW/SIMD @40nm
104MIPS/mW @40nm



# Decoupling of instructions and data
# Instruction set (scalar, vector)
# Size of SIMD core for 2G, 3G, LTE, 802.11b/g/n, DVB-H

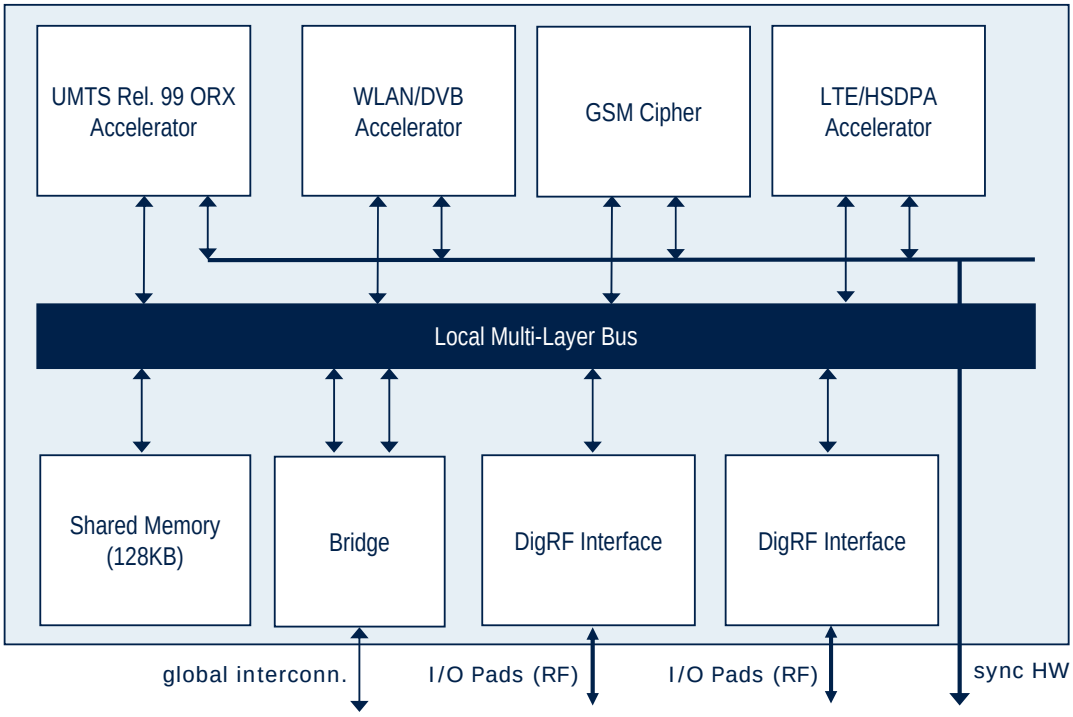
- 4 Proc. Elements + Control (300 MHz)
- Long Instruction Word
  - Computation Slot (DSP Pipe & IU)
  - Memory Slot
  - Communication Slot
- Local Data Memories with DMA Controllers
- Asynchronous Coupling of Controllers (Basic Patent)



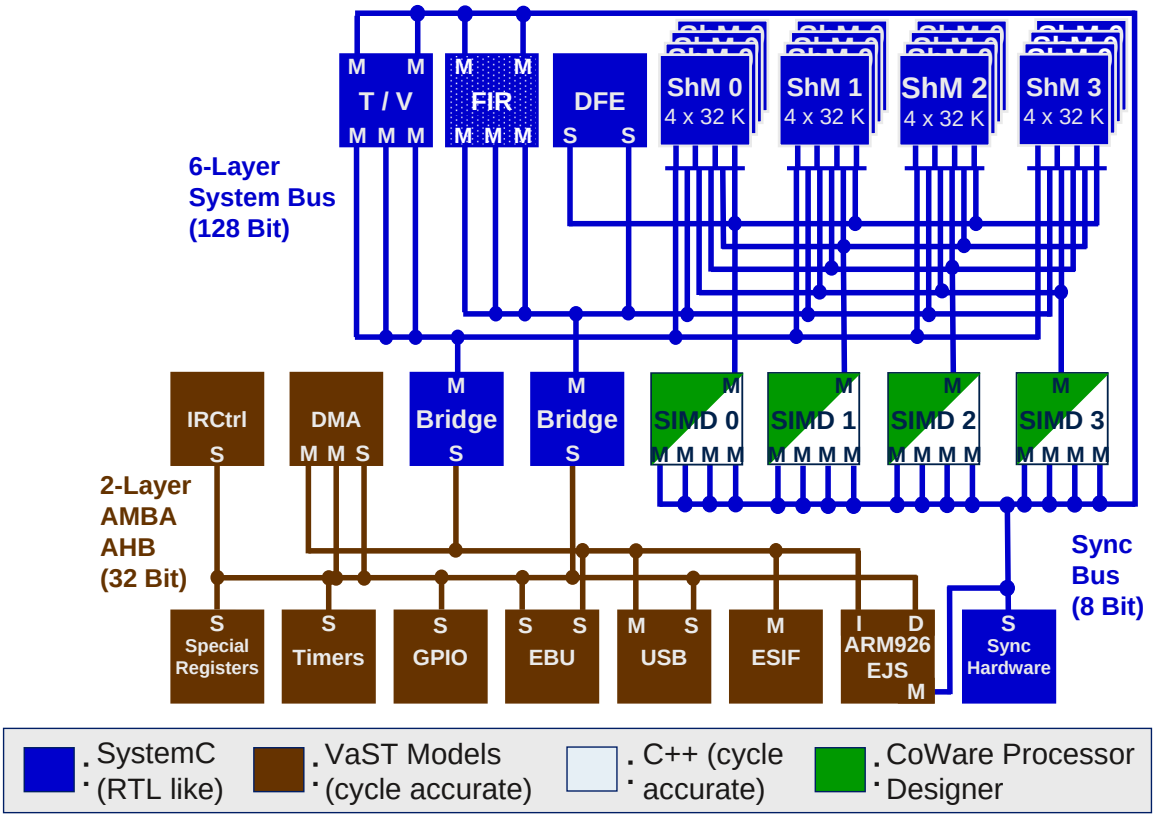

| | |
|---|---|
| GP core RC1632: | 230 instructions |
| Local memory access: | 42 load/store instructions |
| DSP Pipe: | 14 instructions (mul, div, absdif, special) |
| Integer Unit: | 82 instructions (ALU, shift, bit-level, special) |
| Inter-PE Communication: | 21 instructions (vector element permutations) |




# Accelerator Cluster Block Diagram



# MuSIC Virtual Prototype




# Software Design & Verification Process

**Specification of radio standard (e.g. 3GPP Docs for WCDMA)**

Paper

**build functional system model**
**based on inherent functional partitioning**

**Functional verification**
Concept verification
Bit error rates
Algorithm performance
Capture Timing

**Executable model of the radio standard as C-Program**
**(alter'tively: CoCentric/COSSAP, SPW, Simulink)** ← **C functions for signal processing**

Windows

**mapping to MuSIC architecture**
**(partitioning & scheduling, SIMD & VLIW by assembly)**

**software profiling & mapping**
HW & SW partitioning
Synchronization
Performance profiling
Timing estimate

**Multi-threaded C program of the radio standard using ILTOS API** ← **C & PE assembly functions for signal processing**

Windows or SystemC & Single SIMD VP

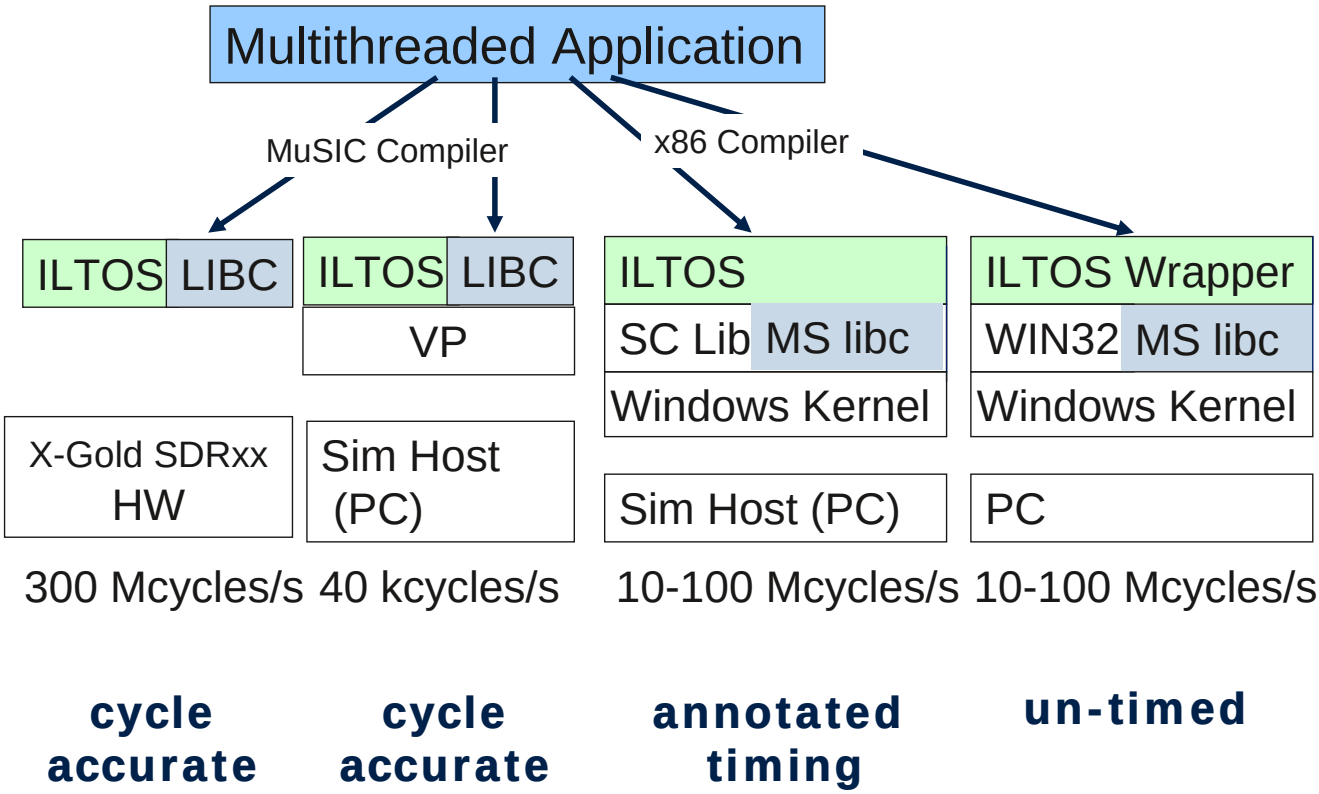
**re-compilation & re-linking for MuSIC**

**refinement of mapping, real-time certification**

**Multi-threaded C program with PE assembly functions of the radio standard using ILTOS on X-Gold SDRxx**

Virtual prototype on Windows | Silicon

## Simulation Abstraction Levels







## Challenges

BB:

1. modelling the costs of mapping

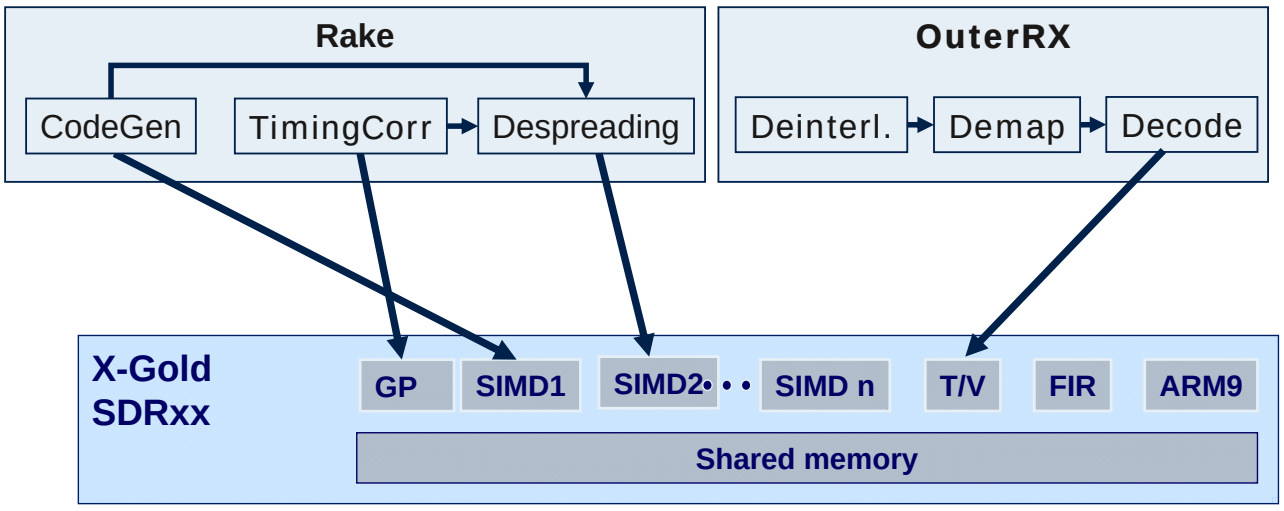
## Standards / Handovers

- Cellular Modem
  - 3G
    - UMTS/WCDMA Rel 6
    - HSDPA, Category 8
    - HSUPA, Category 6
    - LTE*
  - 2G
    - CSD
    - SAIC
    - E-GPRS
      - Class 1-12 and 30-33
    - GPRS
      - Class 1-12 and 30-33
    - DTM class
    - DTM Class 1-12 and 30-33
- GMR-3G
- GMR-3G (2)
  - monitoring for 2G (1), WCDMA/HSPA (5), LTE (5)
- GSM/GPRS/EDGE 1
  - monitoring for WCDMA/HSPA (5) LTE (5), GMR-3G (2)
- WCDMA/HSPA (8/10)
  - monitoring for LTE (0), 2G (0), GMR-3G (2)
- LTE (12)
  - monitoring for 2G (0), WCDMA/HSPA (0), GMR-3G (2)




## Partitioning & Scheduling

- **Knowledge from standard:**
  - Data & Control dependencies
- **Placement of threads**
  - Uses Virtual Prototype
- **Re-distribution or re-definition of threads**
  - Uses Virtual Prototype



# Weight of Programming Steps

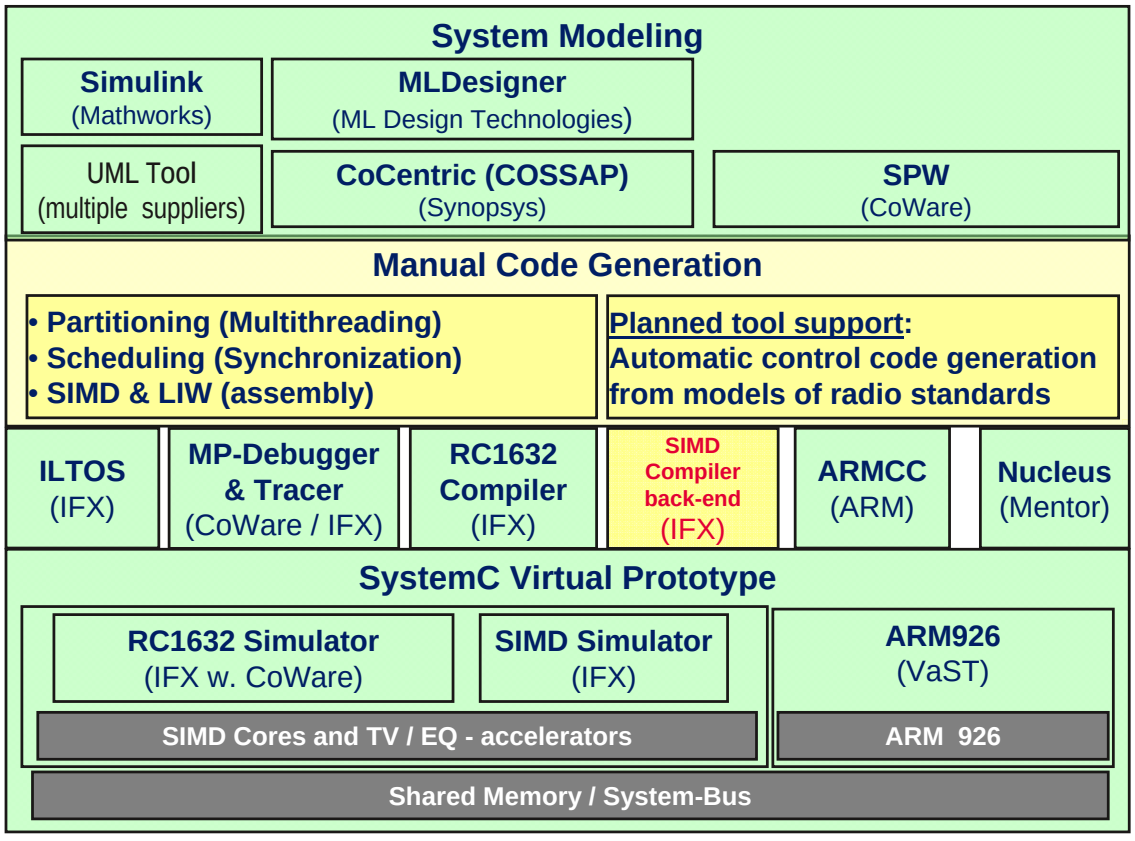
Basis for the implementation is a fix-point C-reference

- Implementation steps
  - Determine mapping of data to SIMD array 6%
  - Implement SIMD-asm function (LIW-program) 8%
  - Derive schedule for load/store units 3%
  - Optimize load/store/computation overlap 5%
  - Define threads by profiling on single SIMD core 3%
  - Partitioning and scheduling of multiple threads 20%
  - Debugging 55%

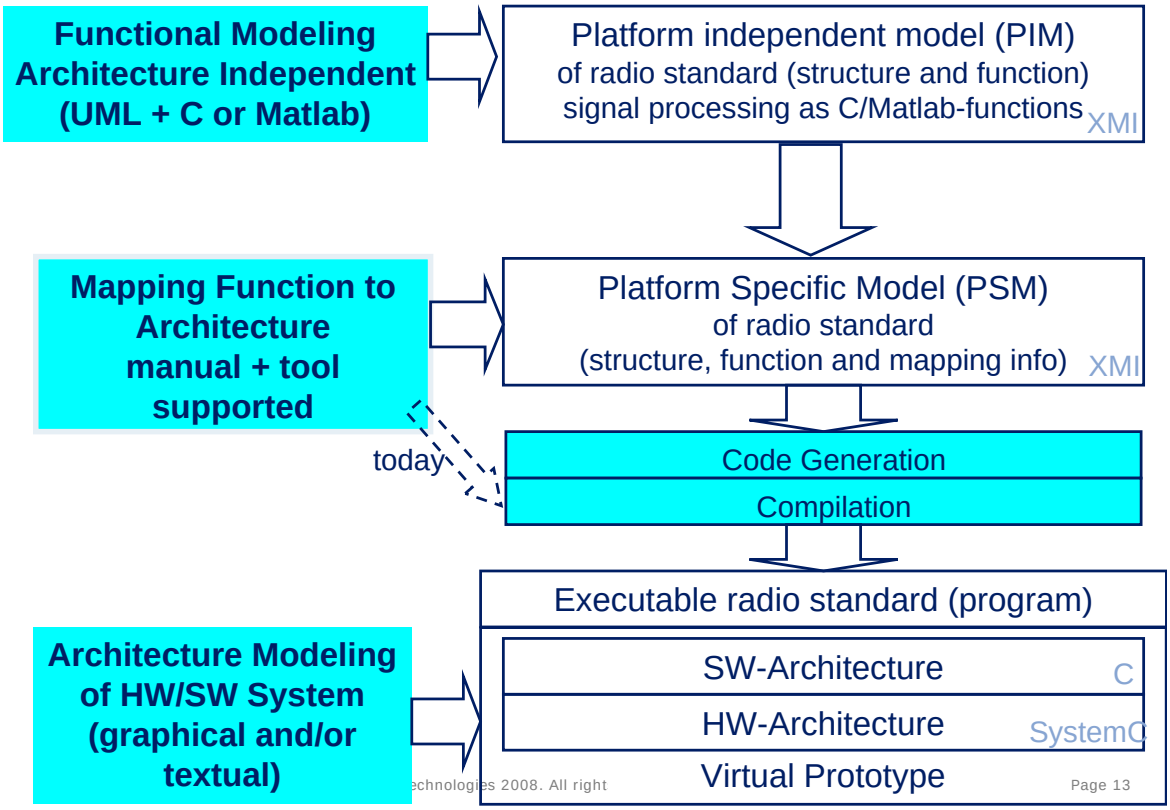
- Effort for SIMD assembly implementation depends on complexity
  - Minimum: 3 days per function (simple functions like FIR)
  - Maximum: < 20 days (complex functions like FFT)
  - This includes the implementation itself and testing/debugging




# Tools/Code Generation



# Target: Model Based Code Generation








# Challenges

BB:

1. modelling the costs of mapping
2. ASIP/RA for Outer Receiver,
3. code generation
4. compiler back-end

PS:

1. unified stack architecture
2. MAC MP architecture

RF:

1. ASIP/RA for digital front-end
2. broadband vs. tunable/configurable

# System Design Complexity