Software Defined Modem
A commercial platform for wireless handsets

Charles F Sturman

VP Marketing

June 22nd ~ 24th Brussels

charles.stuman@cognovo.com
www.cognovo.com

Agenda

- **SDM – Separating hardware from software**
  - Why now
  - Multimode modem processing requirements

- **A whole system approach**
  - Optimising processor resources for the task in hand
  - Low power design and scalability
  - Performance benchmarks achieved

- **Software development methodology**
  - Task scheduling and mapping
 - System design, verification and optimisation
  - Bottom up vs Top down

- **Modem Compute Engine**
  - A commercially available platform for low-power handheld wireless

Why Now ?

- **A changing market**
  - Rapid evolution in applications and services
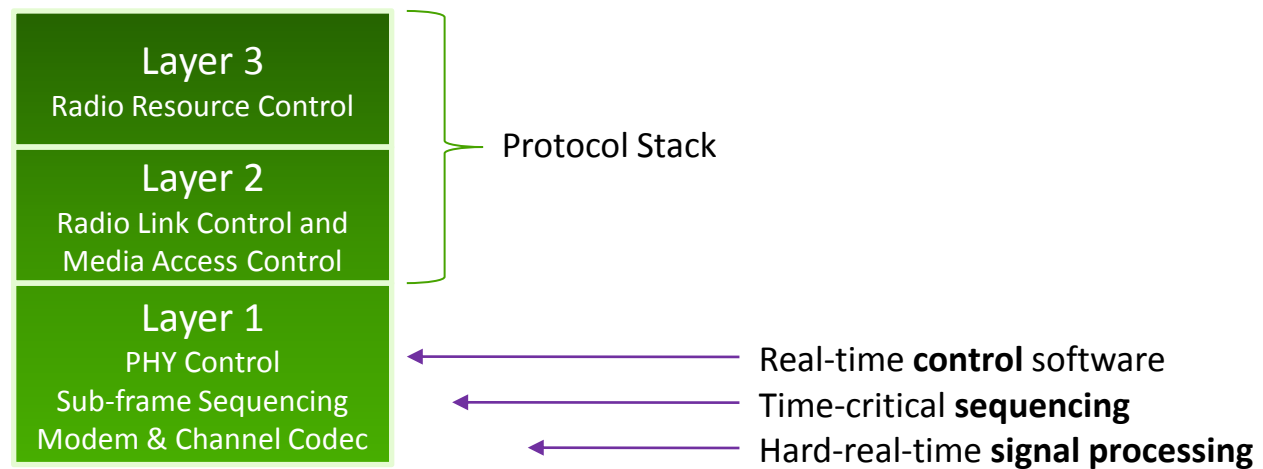 - Many products need wireless : Smartphones, Tablets, Gaming, Home entertainment

- **Rapidly increasing modem complexity**
  - Driven by applications and services
  - Increasing time to market and cost
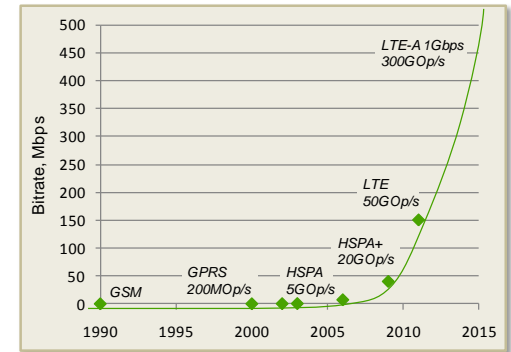
- **SDM delivers**
  - Faster : Time to market and market response
  - Flexibility : One chip = many products / many standards
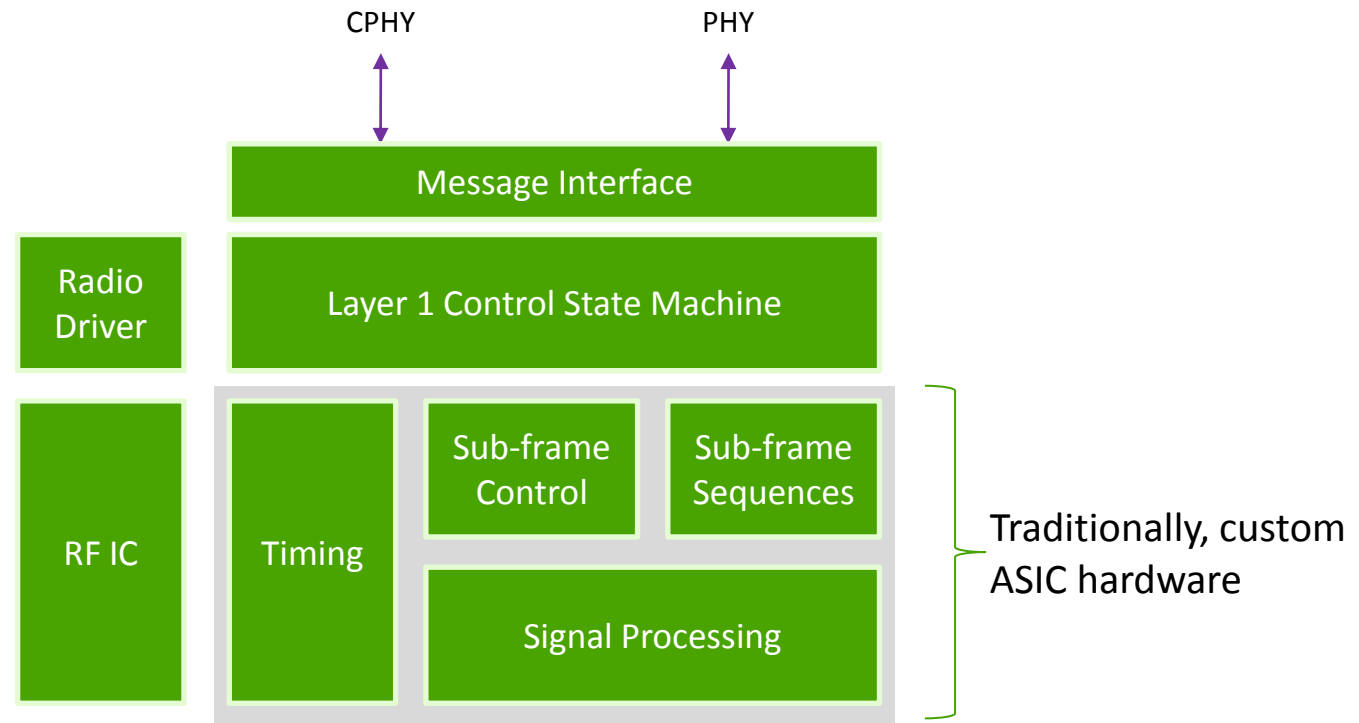 - Smaller : Smaller die size = Lower cost

Modem Requirements

Layer 3 Radio Resource Control	
Layer 2 Radio Link Control and Media Access Control	} Protocol Stack
Layer 1 PHY Control Sub-frame Sequencing Modem & Channel Codec	← Real-time **control** software ← Time-critical **sequencing** ← Hard-real-time **signal processing**

- Layer 1 combines real-time control with time critical signal processing

Modem Requirements

CPHY   PHY

Message Interface

Radio Driver

Layer 1 Control State Machine

RF IC

Timing

Sub-frame Control

Sub-frame Sequences

Signal Processing
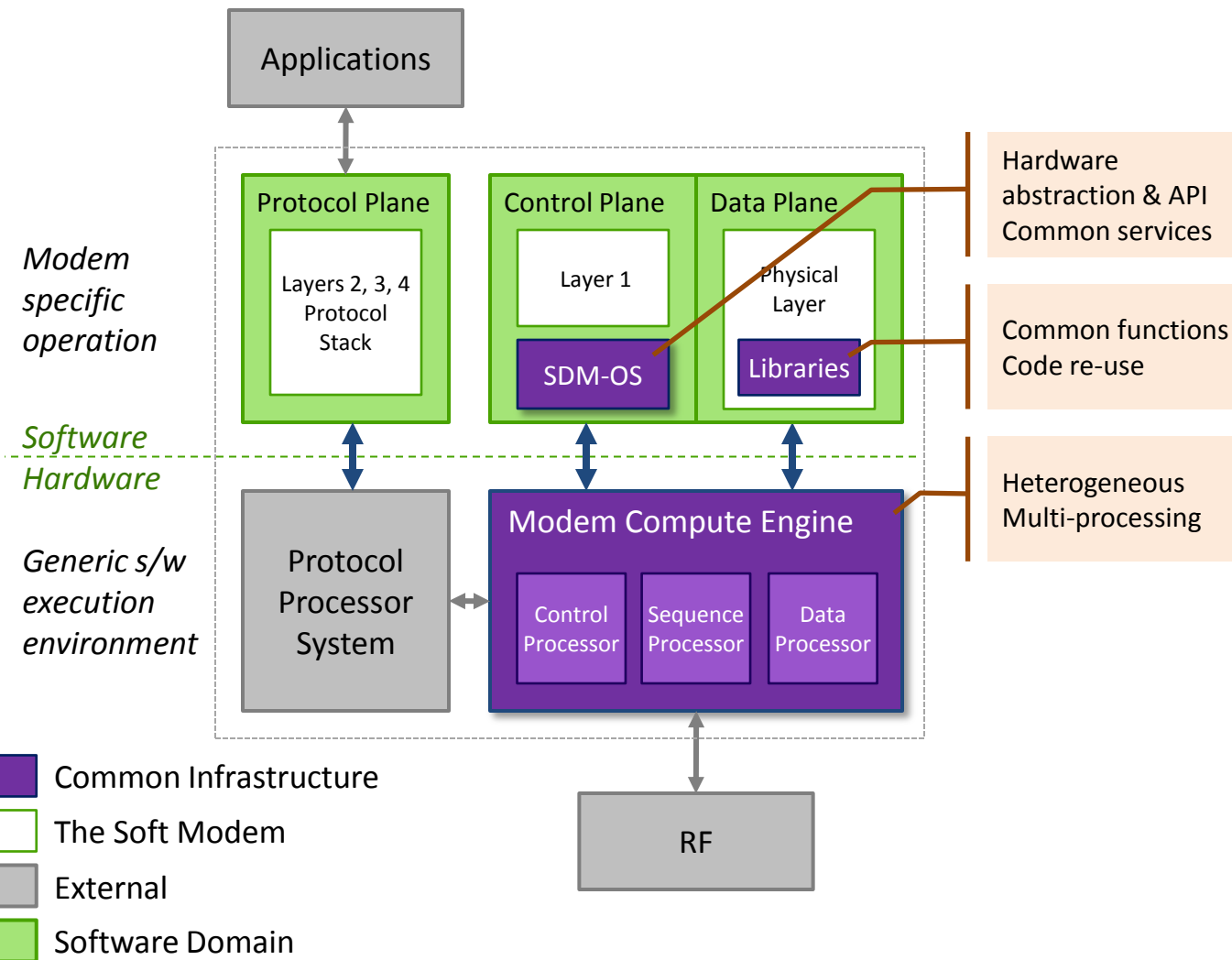
Traditionally, custom ASIC hardware

- Layer 1 combines real-time control with time critical signal processing

- Deterministic operation has traditionally demanded hardware

- SDM enables the entire modem to be expressed in software

Modem Requirements
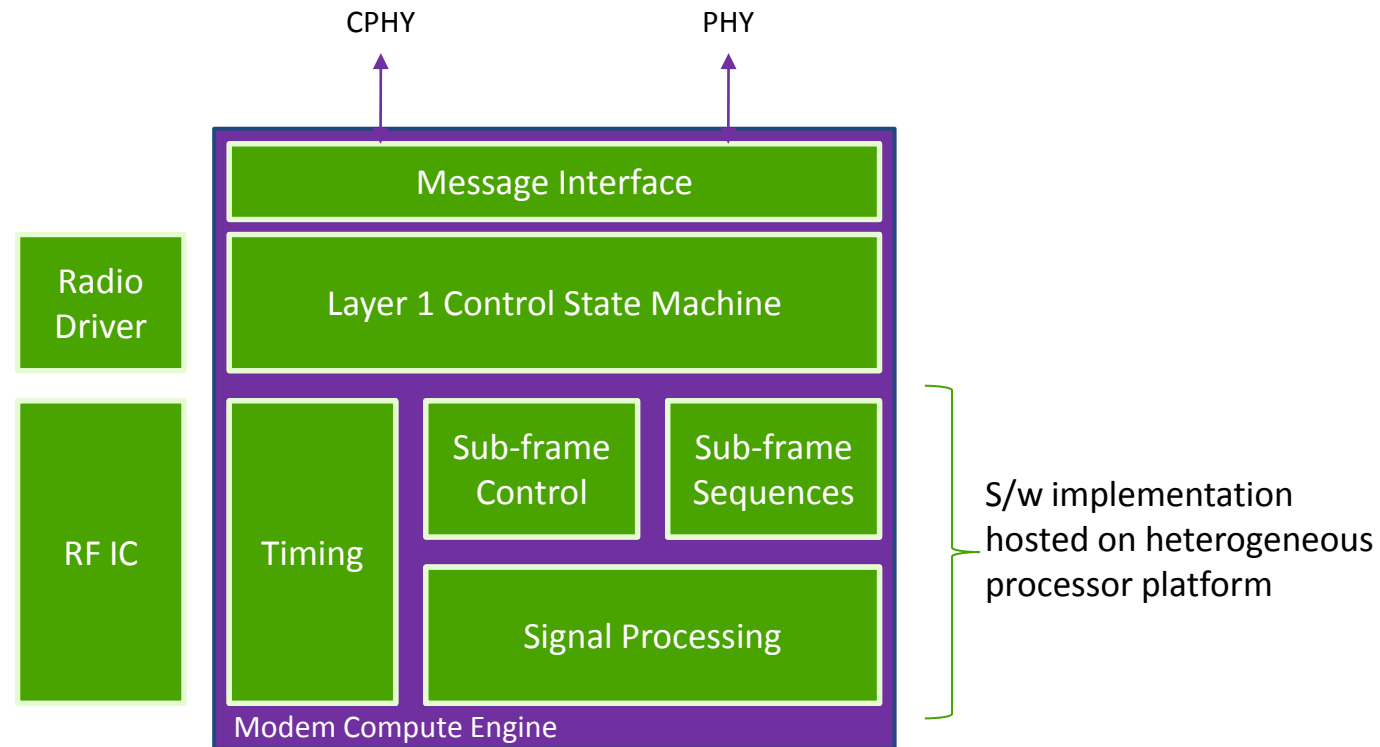
- Layer 1 combines real-time control with time critical signal processing

- Deterministic operation has traditionally demanded hardware

- SDM enables the entire modem to be expressed in software

A System Solution for SDM – Layer 1 Partitioning

Applications

Modem specific operation

Protocol Plane
Layers 2, 3, 4 Protocol Stack

Control Plane
Layer 1
SDM-OS

Data Plane
Physical Layer
Libraries

Hardware abstraction & API Common services

Common functions Code re-use

Software
Hardware

Generic s/w execution environment

Protocol Processor System

Modem Compute Engine
Control Processor
Sequence Processor
Data Processor

Heterogeneous Multi-processing

RF

Common Infrastructure

The Soft Modem

External

Software Domain

Layer 1 Partitioning
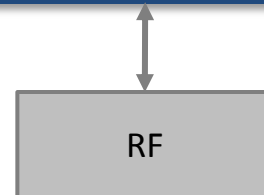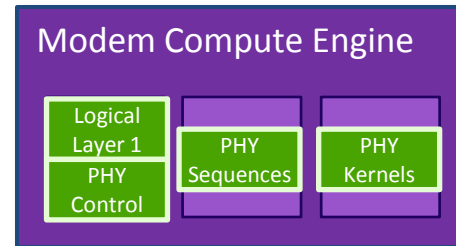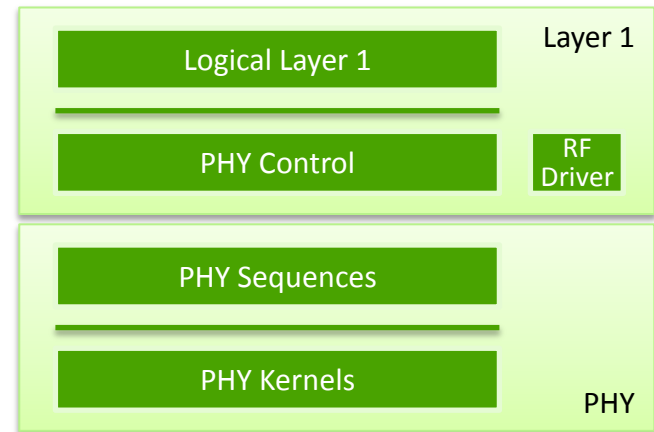
Layer 1 **Control**

- GP processor support for Layer 1 control

PHY Task **Sequencing**

- Real-time deterministic task sequencing driven by on-air timing
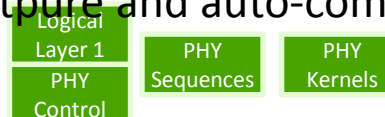
Hard-real-time **Signal Processing**

- PPA efficient wireless algorithmic processing
- No modem-specific hardware processing

Layer 1 System Design

- Existing Layer 1 design & implementation must be supported
  - Seamless integration with existing protocol stacks

- Operational control, management & service provision
  - Largely specific to SDM platform : Auto generated code avoids creating extra engineering workload & human error

- Sub-frame sequencing
  - Extension of present PHY modelling carried out with e.g. Simulink or Co-Ware : Graphical catpure and auto-compilation of flow control

  | Logical Layer 1 PHY Control | PHY Sequences | PHY Kernels |

- Uplink / Downlink PHY algorithms
  - Compilation of existing fixed-point C-code, with efficient profiling and optimisation tools. Must be able to re-validate against reference models

Introducing Cognovo's Modem Compute Engine (MCE)

Real-time L1 / PHY computing system

- PHY Data Plane
  - Multi-core Vector Signal Processing
    - Designed-for-Wireless ISA
  - Instruction & Data parallelism
  - \>> 130 GOp/s @500 MHz
  - 11-way VLIW
  - 32-way complex vector data-path

- Layer 1 Control Plane
  - ARM CPU hosts SDM OS and Layer 1
  - Mode independent timing / power ctl
 - Multi-standard Turbo engine
  - RF interface

- SDM Sequencer
  - Precise scheduling of kernels
  - Autonomous operation removes overhead from L1 CPU
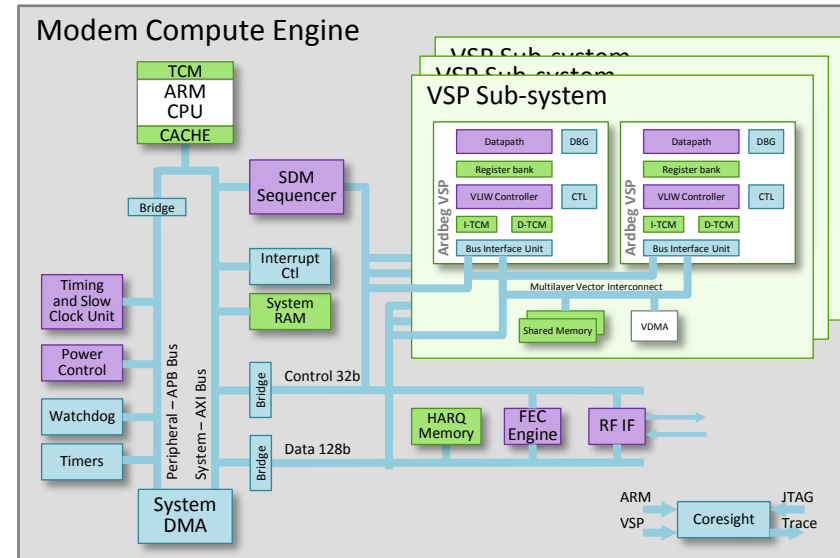
Dual VSP Core (MCE120)

- Cat 4 LTE capable (150Mb DL / 50Mb UL)
  - Simultaneous DL and UL: 62% loaded at 400MHz
- 7.5 mm^2 and < 100mw in 32nm

- Similar power consumption to existing baseband silicon but with smaller area

MCE120 Performance Benchmarks @ 40nm LP

MCE clock frequency vs on-air bit-rate
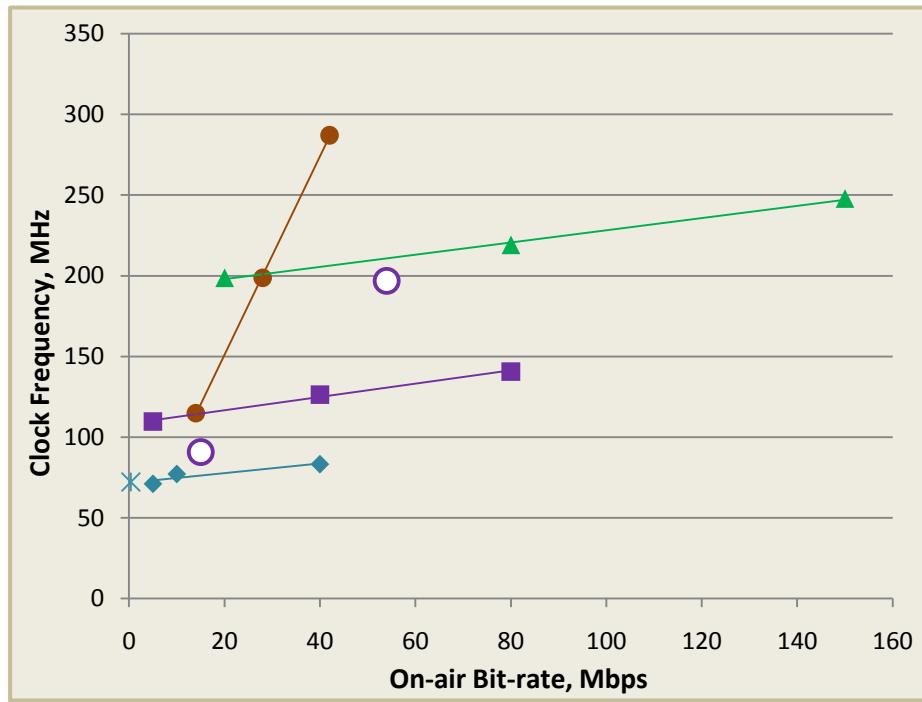
R8 LTE-FDD Configurations
- ◆ 5MHz Channel Cat 2 : 5-40 Mbps
- ■ 10 MHz Channel Cat 3 : 5-80 Mbps
- ▲ 20 MHz Channel Cat 4 : 20-150 Mbps

R5 to R8 3G HSPA Configurations
- ✕ R5 W-CDMA : 384 Kbps
- ● R6 HSPA : 14 Mbps
- ● R7 HSPA+ MIMO : 28 Mbps
- ● R8 HSPA+ Dual Carrier : 42 Mbps

Connectivity
- ○ WiFi 802.11g : 54 Mbps
- ○ DVB-T (HD) : 15 Mbps

- **Combined DL / UL estimates for 3GPP Rel5 to Rel8**
  - HSPA evolution and LTE : Clock-rate scaling from 75MHz to 300MHz

- **Clock rate and power can also scale with signal conditions**
  - High SNR (light signal processing) to low SNR (heavy signal processing)
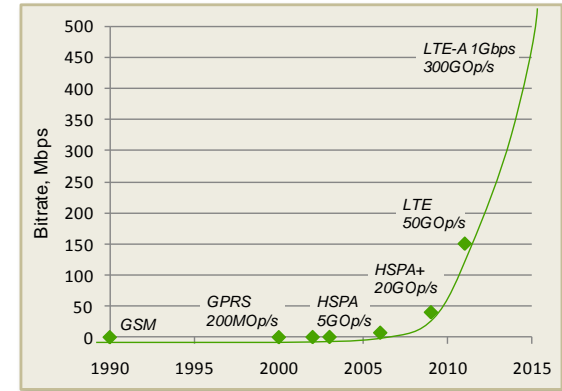
Power & Performance Scalability

- ■ **The same architecture will scale up or down**
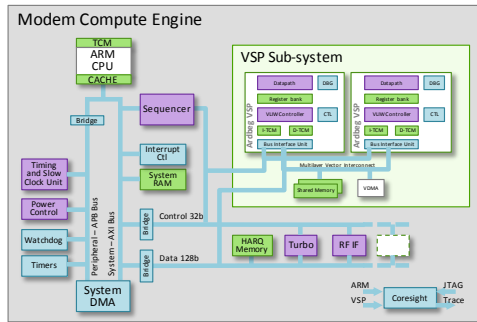
Higher clock rate

More memory

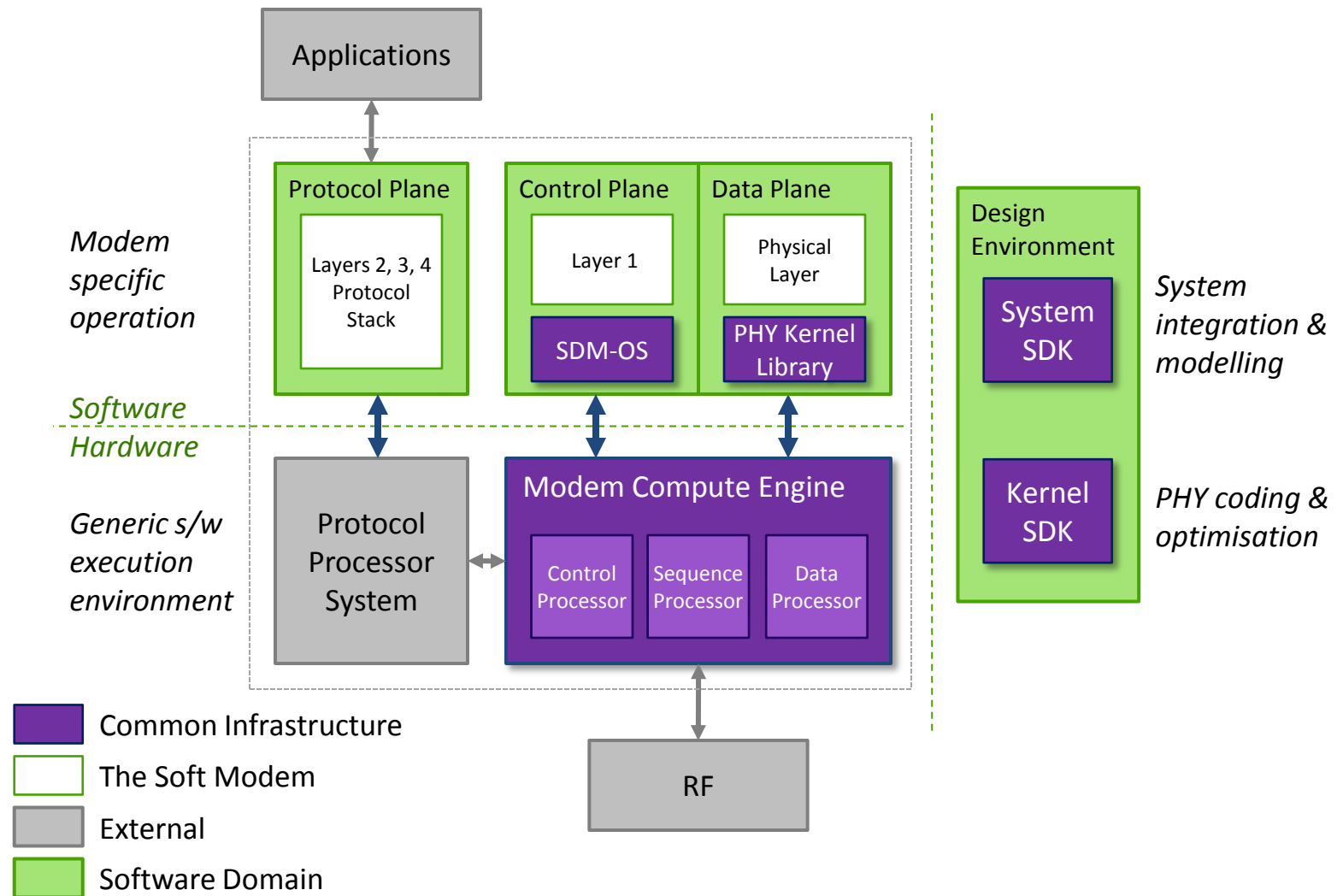More processing resources:
Multi-core / Multi-way

Lower clock rate

Less memory

Less processing resources:
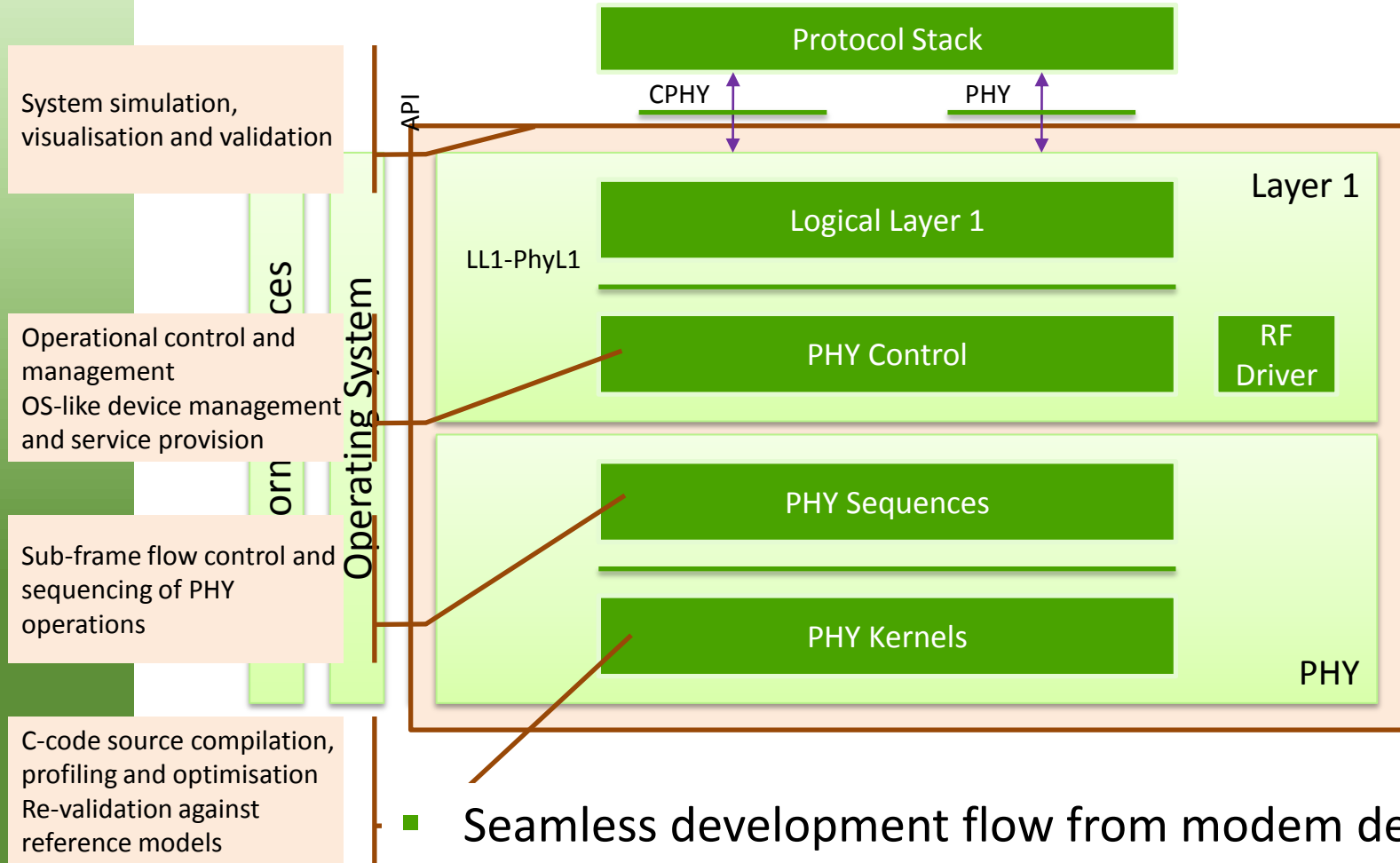Multi-core / Multi-way

- ■ **Modem requirements drive Power Performance and Area**
- ■ **PPA drives silicon implementation and process choice**

A System Solution for SDM – Development Partitioning

Applications

Protocol Plane
- Layers 2, 3, 4 Protocol Stack

Control Plane
- Layer 1
- SDM-OS

Data Plane
- Physical Layer
- PHY Kernel Library

Design Environment
- System SDK
- Kernel SDK

Modem specific operation

Software
Hardware

Generic s/w execution environment

Protocol Processor System

Modem Compute Engine
- Control Processor
- Sequence Processor
- Data Processor

System integration & modelling

PHY coding & optimisation

RF

Legend:
- Common Infrastructure
- The Soft Modem
- External
- Software Domain

Development Partitioning

System simulation, visualisation and validation

Operational control and management
OS-like device management and service provision

Sub-frame flow control and sequencing of PHY operations

C-code source compilation, profiling and optimisation
Re-validation against reference models

Operating System

API

Protocol Stack

CPHY   PHY

Layer 1

LL1-PhyL1

Logical Layer 1

PHY Control   RF Driver

PHY Sequences

PHY Kernels
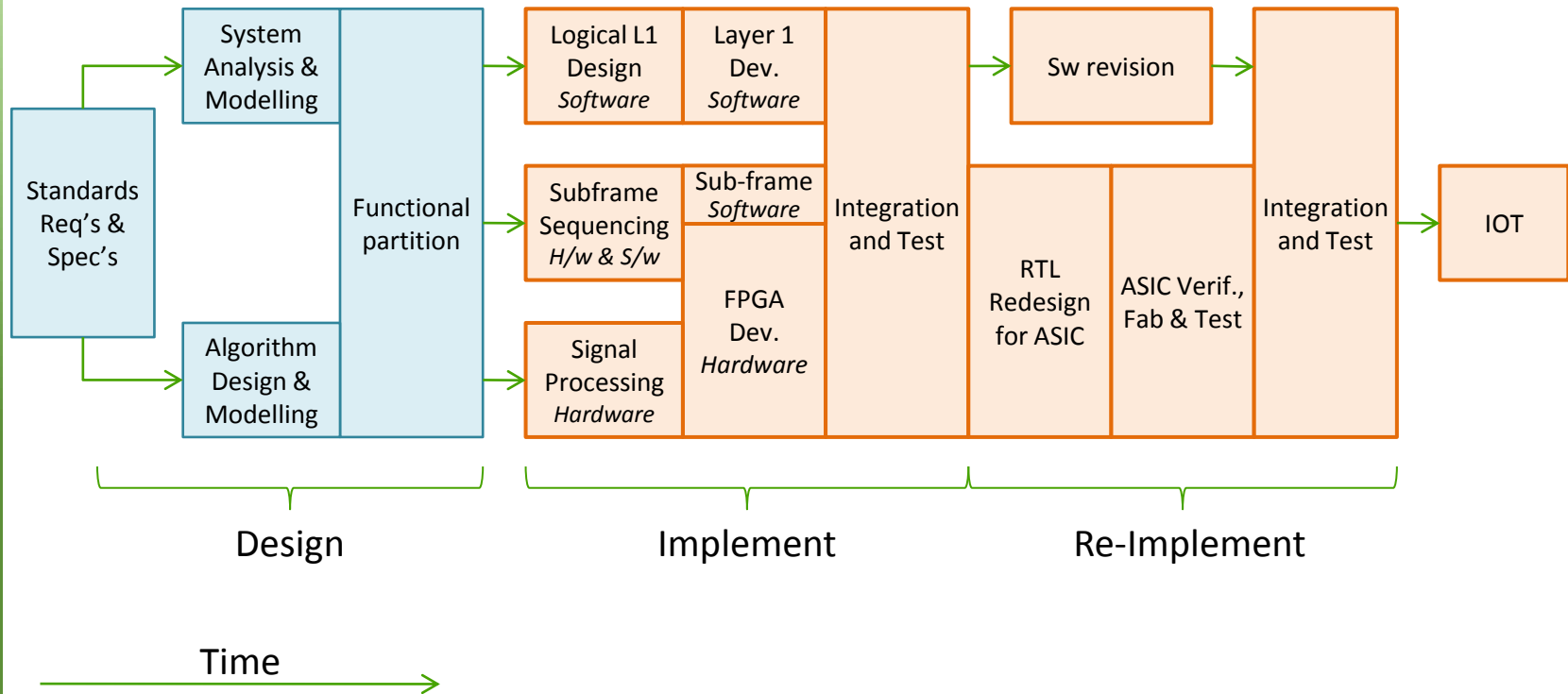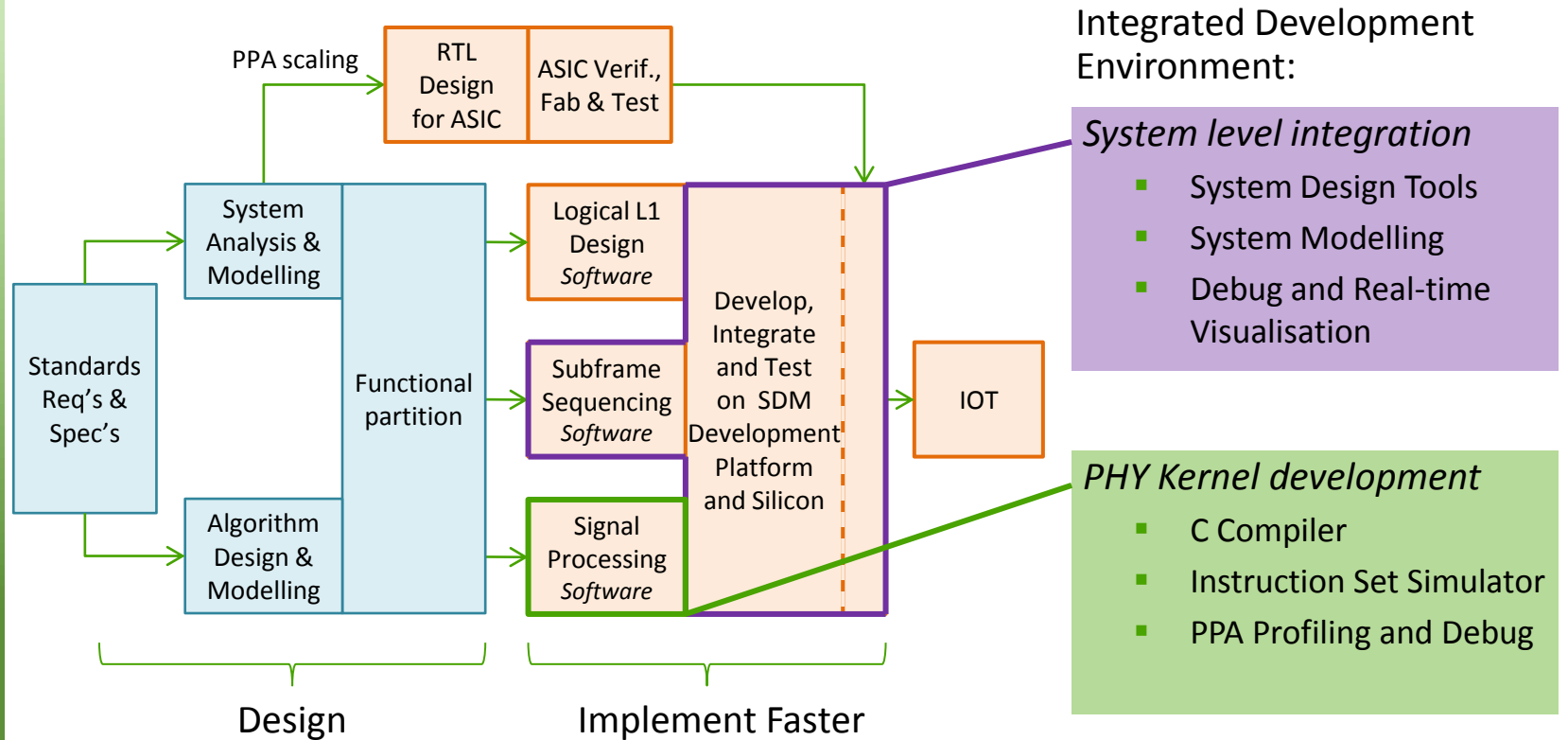
PHY

- Seamless development flow from modem design through implementation and validation is key

Traditional Modem Design Flow

SDM Design Flow


Integrated Development Environment:

System level integration
- System Design Tools
- System Modelling
- Debug and Real-time Visualisation

PHY Kernel development
- C Compiler
- Instruction Set Simulator
- PPA Profiling and Debug

Diagram labels:
- PPA scaling
- RTL Design for ASIC
- ASIC Verif., Fab & Test
- System Analysis & Modelling
- Standards Req's & Spec's
- Functional partition
- Algorithm Design & Modelling
- Logical L1 Design *Software*
- Subframe Sequencing *Software*
- Signal Processing *Software*
- Develop, Integrate and Test on SDM Development Platform and Silicon
- IOT
- Design
- Implement Faster

- Silicon development is outside modem critical path
- Software-only flow allows Develop, Integrate and Test in single step
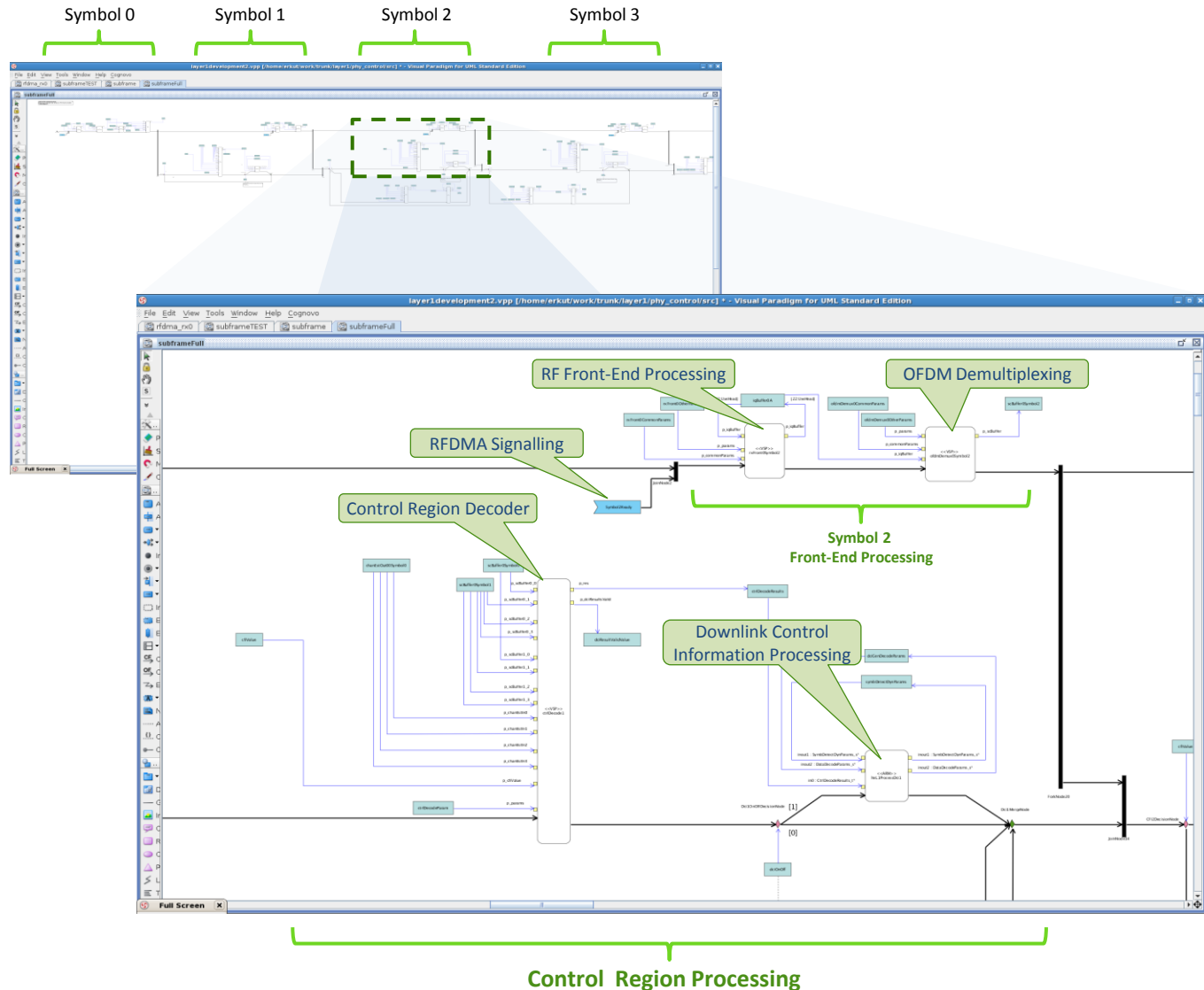
LTE Downlink Control Processing Example
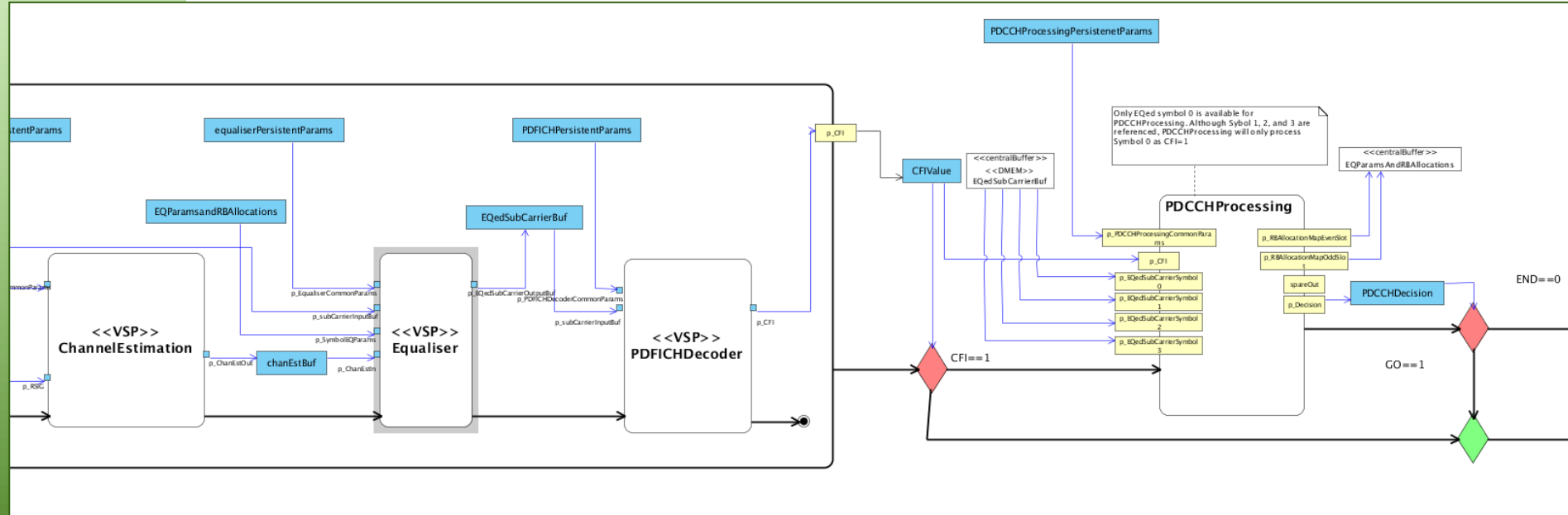
- UML constructs for control and data flow
- Parameterised PHY Kernels
- Co-ordinated by SDM-OS and executed in real-time by SDM Sequencer

PHY System Design and Capture

Visualisation Tool – LTE Example
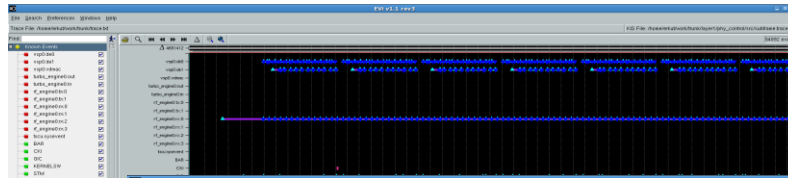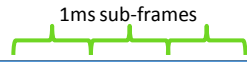
Real Time PHY System Trace

10ms LTE Frame Real-time Trace

1ms sub-frames

1ms LTE Sub-frame Real-time Trace

1ms Sub-frame
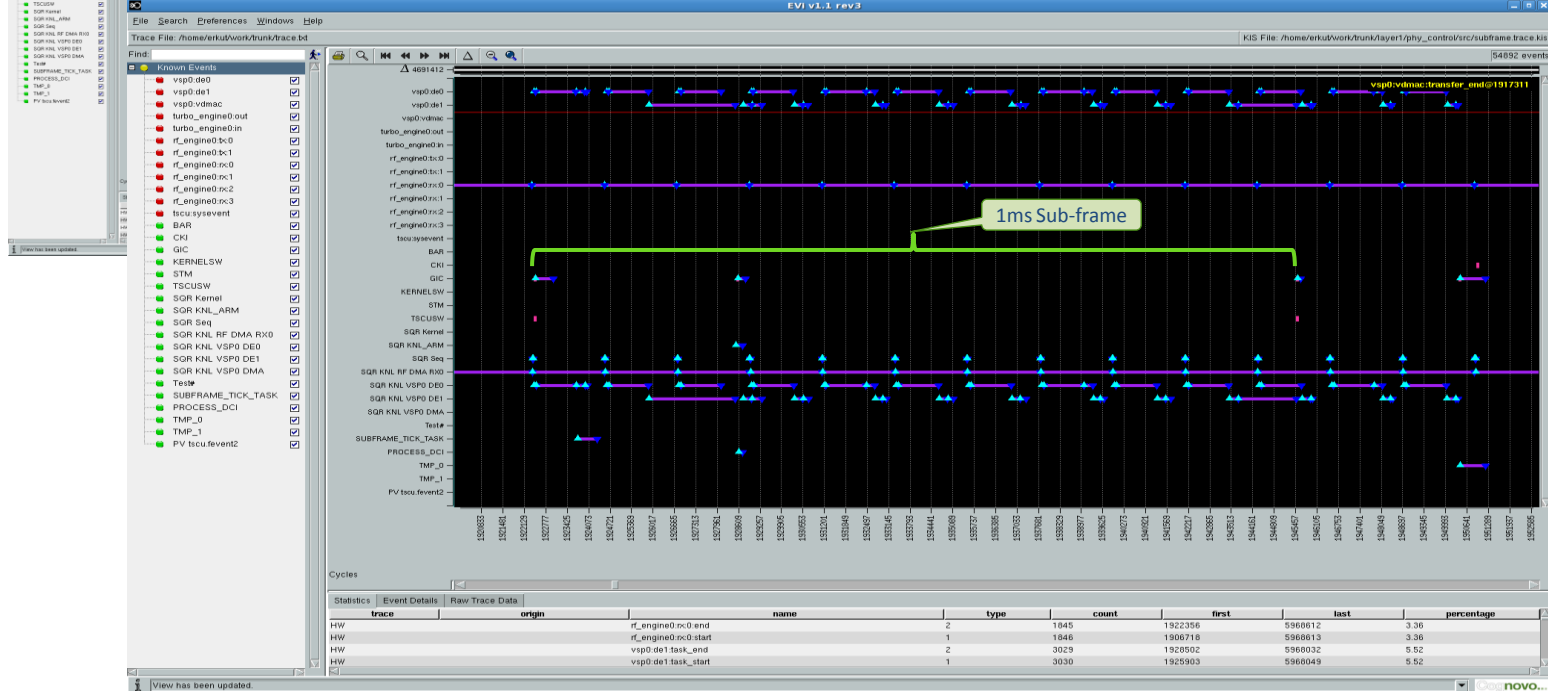
Conclusion

- **Strong demand for a new approach to wireless product development**
  - Faster : Time to market
  - Flexibility : Single solution for multiple products, markets and standards
  - Smaller : Cost of development and cost of product

- **However, there must be no penalty**
  - Existing products must be met or bettered : Power / Cost / Size
  - Legacy investment maintained : e.g. Protocol stack software / multi-mode
  - Seamless design flow

- **SDM can deliver on all of these factors**
  - Heterogeneous task optimised processing
  - Highly parallel architecture for scalability
  - Whole System approach to power management , programming and debug
  - Control / Real-time scheduling / Data processing