TU WIEN

Design Refinement of Embedded Analog/Mixed-Signal Systems

… and how to support it*

Univ.Prof. Dr. habil. Christoph Grimm

Chair Embedded Systems

Vienna University of Technology

Institut für Computertechnik

ICT

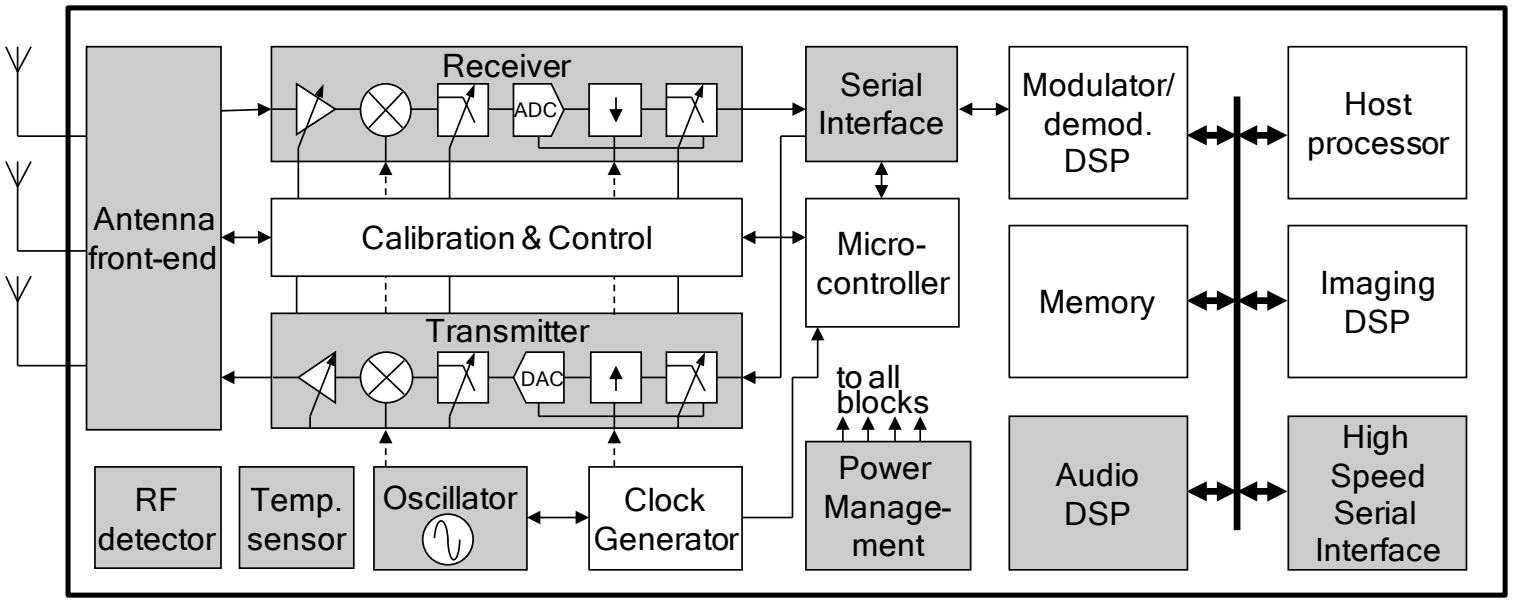
Institute of Computer Technology

*partially supported by

EC FP 6 „ANDRES“, No. IST-5-033511

BMBF/edacentrum „SAMS“, No. 01 M 3070

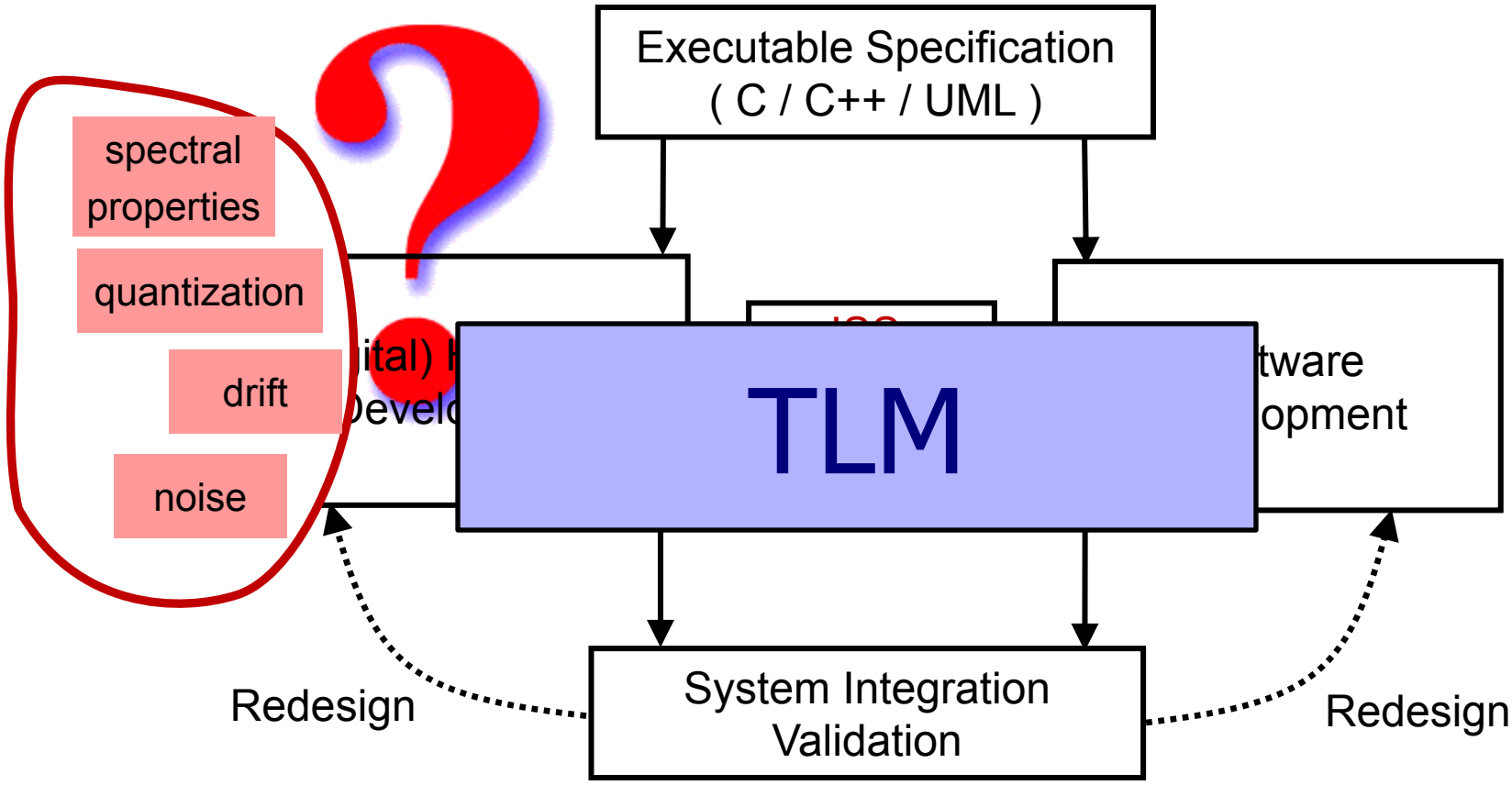
Beyond Moore!



[Grimm/Barnasconi/Vachoux/Einwich: An Introduction to Modeling Embedded Analog/Mixed-Signal Systems using SystemC AMS Extensions. OSCI, June 2008]

- Tightly interwoven SW/DSP and analog/mixed-signal functionality
  = **E**mbedded **A**nalog/**M**ixed-**S**ignal System (E-AMS)

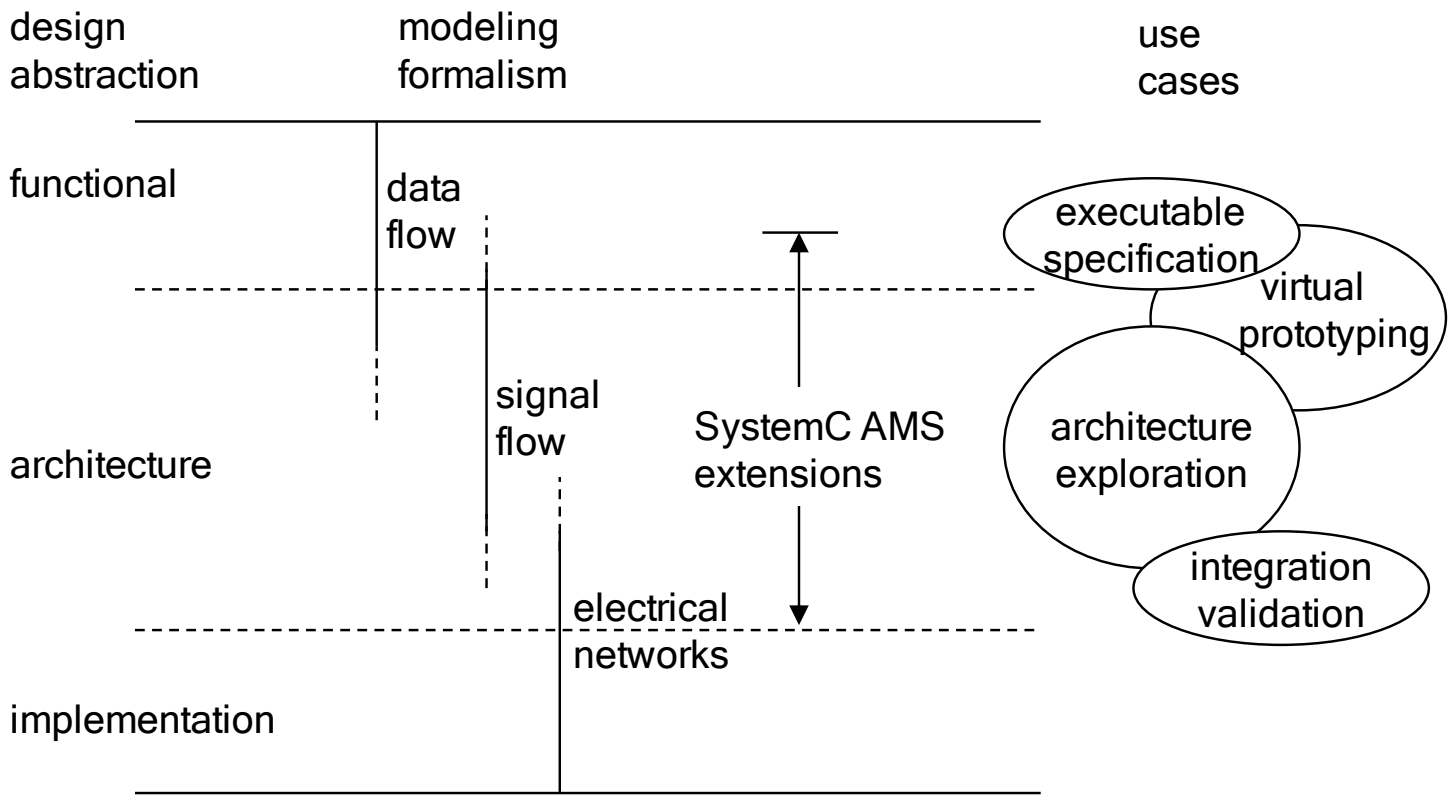
Concurrent Engineering with ISS, VP and TLM



Design refinement of E-AMS systems

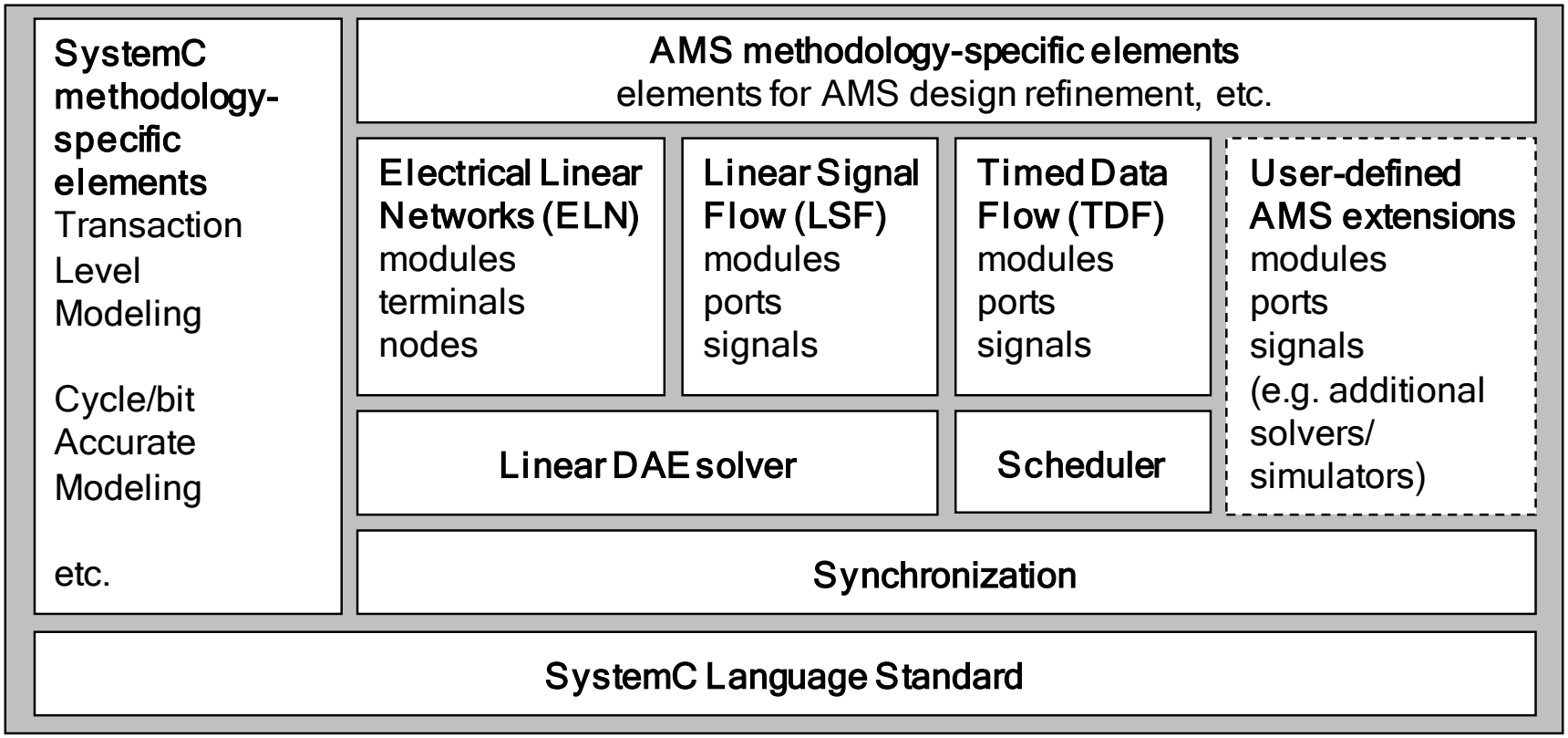
1. **SystemC AMS extensions**
2. Design refinement of E-AMS
3. Conclusion & Outlook

Between functional model and implementation



[Grimm/Barnasconi/Vachoux/Einwich: An Introduction to Modeling Embedded Analog/Mixed-Signal Systems using SystemC AMS Extensions. OSCI, June 2008]

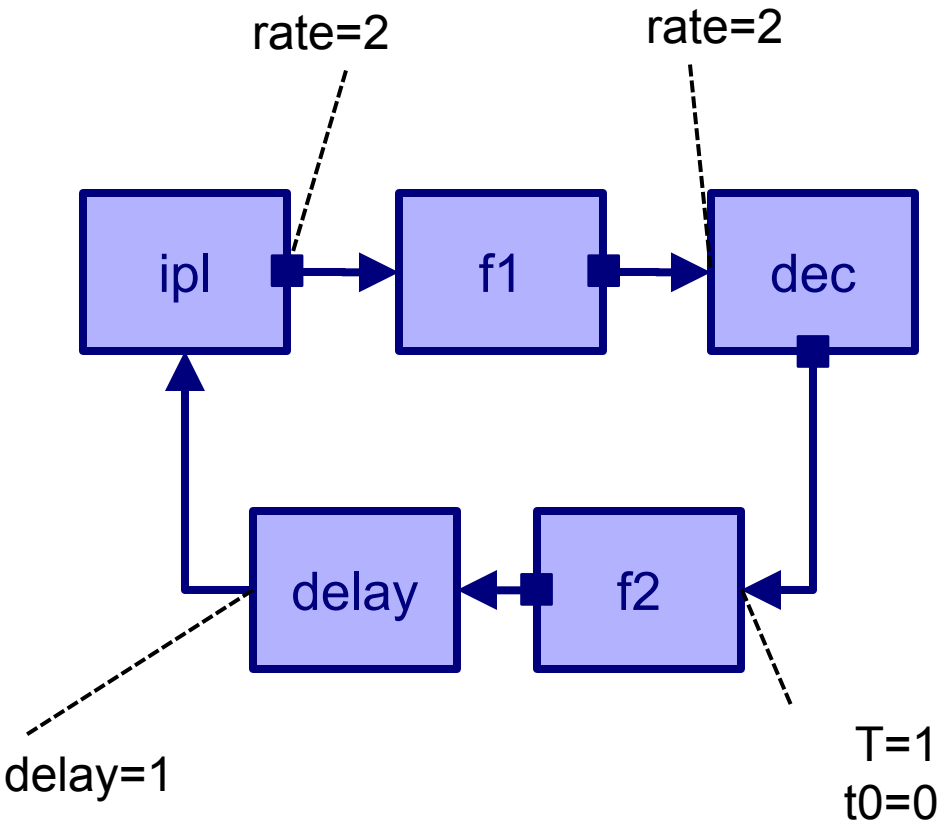
SystemC AMS extensions

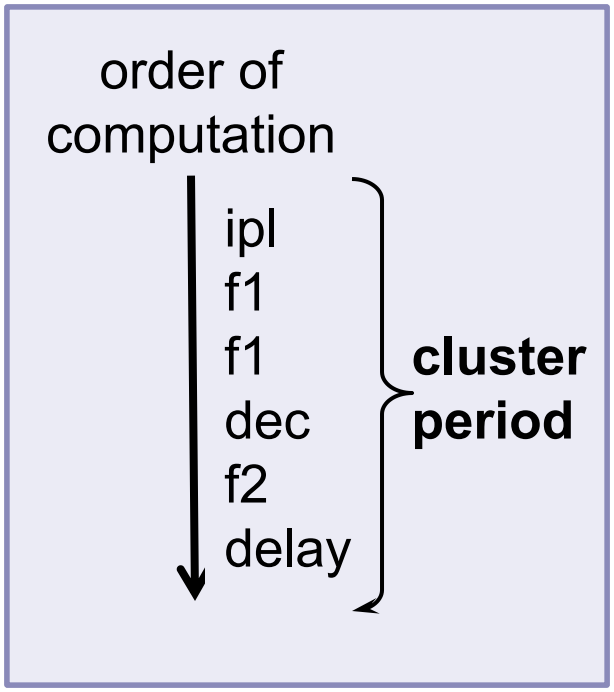


[Grimm/Barnasconi/Vachoux/Einwich: An Introduction to Modeling Embedded Analog/Mixed-Signal Systems using SystemC AMS Extensions. OSCI, June 2008]

Timed Data Flow in SystemC AMS extensions

„cluster“ := set of connected TDF modules





Timed Data Flow Example: Serializer

TDF Module:

primitive module!

```
SCA_TDF_MODULE(par2ser)
{
  sca_tdf::sca_in<sc_bv<8> > in;
  sca_tdf::sca_out<bool>     out;
```

Attributes specify

timed semantics

```
  void set_attributes()
  { out.set_rate(8);
    out.set_delay(1);
    out.set_timestep(1, SC_MS);}
```

processing()

describes

computation

```
  void processing()
  {
    for (int i=7; i >= 0 ; i-- )
      out.write(in.get_bit(i), i);
  }
  SCA_CTOR(par2ser);
}
```

Interfacing Timed Data Flow and SystemC (DE)

Converter ports towards discrete event domain

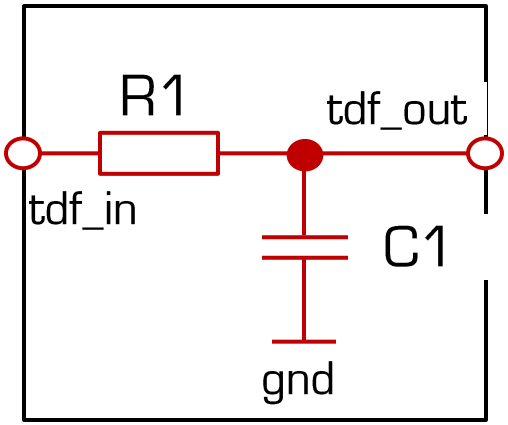
```
sca_tdf::sc_in < <type> >
sca_tdf::sc_out < <type> >
```

Note: Time in MR – TDF may run ahead DE time!

```
sc_time sca_get_time()
```

Linear Electrical Networks, Converters

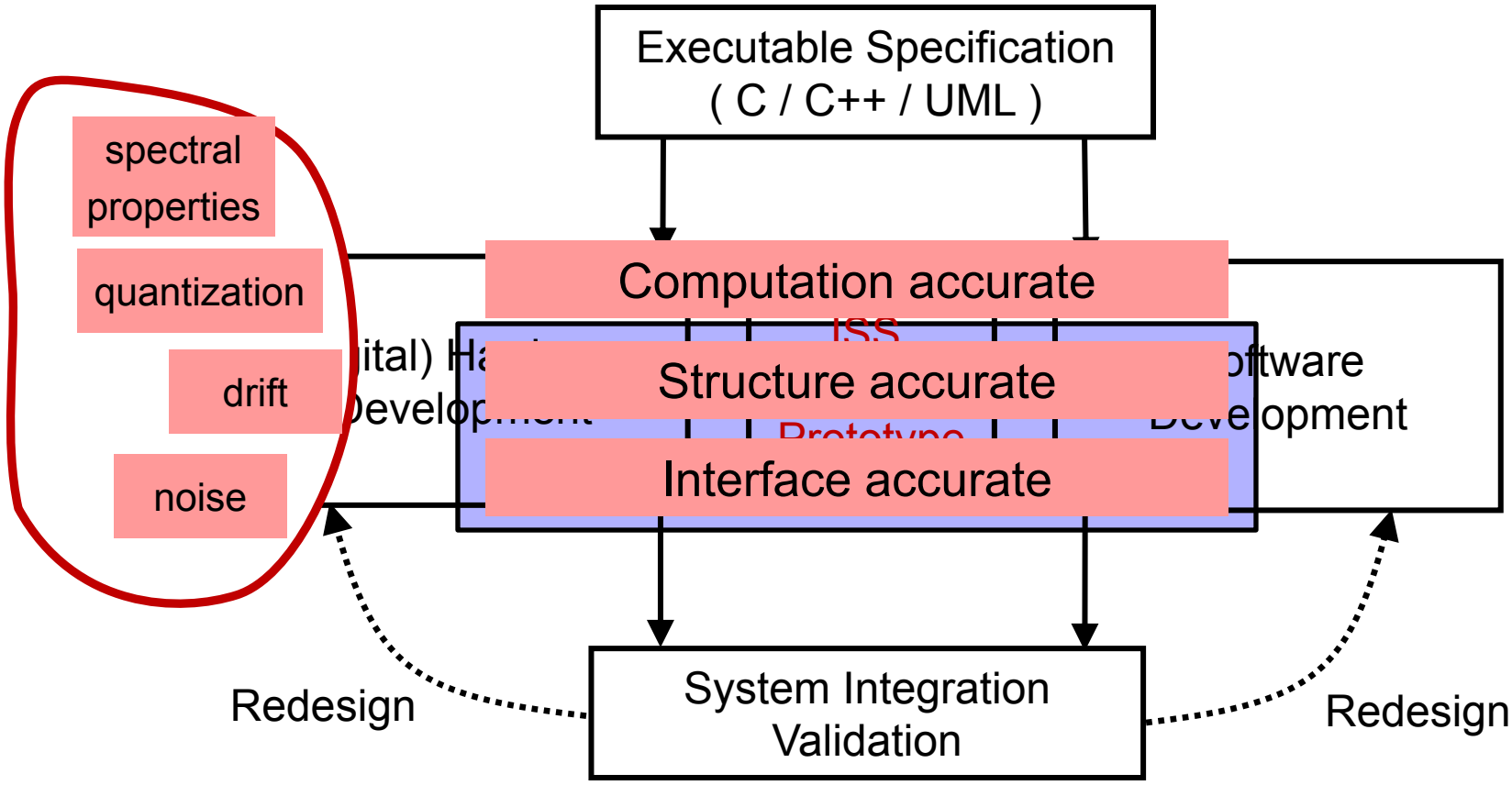
```
SC_MODULE(lp_filter_eln)
{
  sca_tdf::sca_in<double>  in;
  sca_tdf::sca_out<double> out;
  sca_eln::sca_node in_node, out_node; // nodes
  sca_eln::sca_node_ref gnd;           // reference
  sca_eln::sca_r     *r1;              // resistor
  sca_eln::sca_c     *c1;              // capacitor
  sca_eln::sca_tdf2v *v_in;            // converter TDF->U
  sca_eln::sca_v2tdf *v_out;           // converter U->TDF
  SC_CTOR(lp_filter_eln) {
    v_in = new sca_eln::sca_tdf2v("v_in", 1.0);   // scale factor 1.0
      v_in->ctrl(in); v_in->p(in_node); v_in->n(gnd);
    r1 = new sca_eln::sca_r("r1", 10e3);          // 10kOhm resistor
      r1->p(in); r1->n(out_node);
    c1 = new sca_eln::sca_c("c1", 100e-6);        // 100uF capacitor
      c1->p(out_node); c1->n(gnd);
    v_out = new sca_eln::sca_v2tdf("v_out", 1.0); // scale factor 1.0
      v_out1->p(out_node); v_out1->n(gnd); v_out->ctrl(out);
  }
};
```



Design refinement of E-AMS systems

1. SystemC AMS extensions
2. **Design refinement of E-AMS**
3. Conclusion & Outlook

Refinement of E-AMS



Computation accurate model

Quickly coded model – must me available ASAP

- Adds non-ideal effects to executable, *functional* spec
- AMS extensions: non-ideal behavior can be written directly in C-code!

```
void processing()  // Mixer refined with distortions and noise
{
  double rf = in1.read();
  double lo = in2.read();
  double rf_dist = (alpha – gamma  * rf * rf ) * rf;
  double mix_dist = rf_dist * lo;
  if_out.write( mix_dist + my_noise() );
}
```

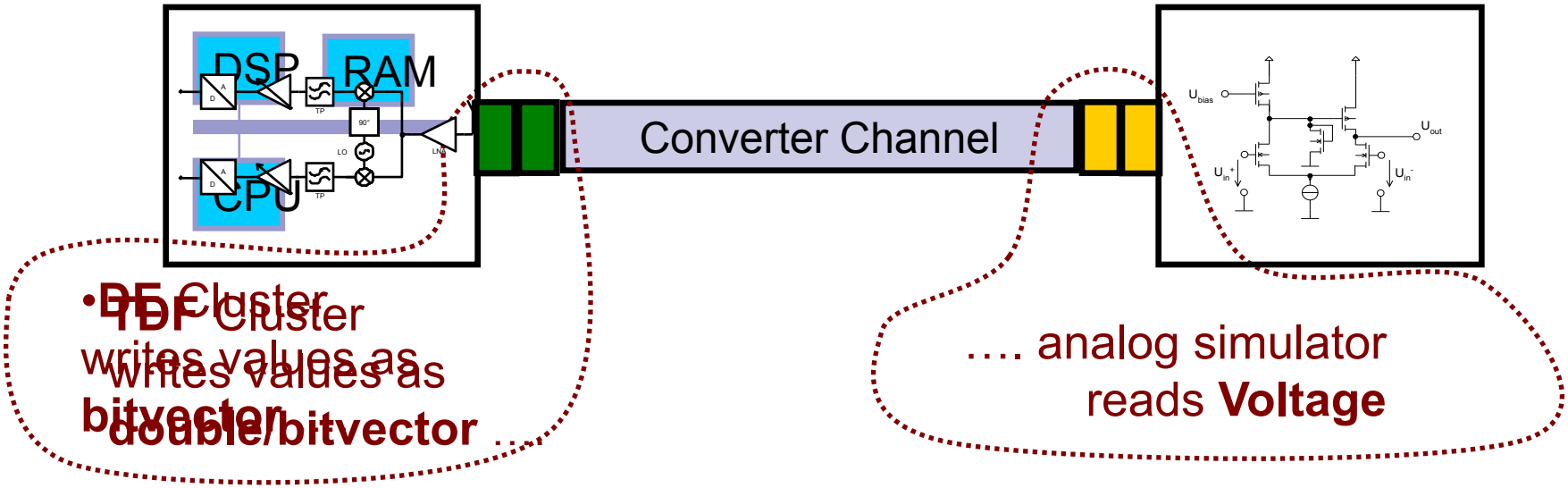
Structure accurate, re-partitioned model

- Accurate partitioning of functional blocks to analog, digital, SW. Structures/Methods like in implementation
- Adapt MoC
  - Analog: Signal flow, Network
  - Digital HW: TDF, DE (or TLM)
- Quick „top-down“ change by „static polymorphism“

```
SCA_MoC_MODULE(par2ser)
{
  sca_MoC::sca_in<sc_bv<8> > in;
  sca_MoC::sca_out<bool>     out;
  …
  SCA_CTOR(par2ser)
}
```

Structure accurate, re-partitioned model

- Easy integration of „buttom up“ available blocks by converter channels (= „analog transactors“)
- Avoids development of „wrapper“



Interface accurate models

- Interface accurate models used for verification of system integration
- All ports accurately as in implementation
  - Enable and clock signals

Modeling issue

- Clock signals or events determine activation of TDF cluster
- Adapter classes can translate between different MoC and protocols
- E.g. from TDF via DE to TLM to protocol

Open issue

- EFFICIENT coupling between TDF and TLM

Design refinement of E-AMS systems

1. SystemC AMS extensions
2. Design refinement of E-AMS
3. **Conclusion & Outlook**

Conclusion

- SystemC AMS extensions provide appropriate means for modeling E-AMS at architecture level
  - Standardization ongoing
  - Converter channels and adapter classes complement TLM transactors – ongoing work ➔ bDREAMS

- Design refinement modeling strategy integrates architecture properties successively into executable spec
  - Quickly available first models
    => Early feedback on feasibility or potential issues
  - Immediate analysis/verification after changing/adding property
    => Optimization / debugging more efficient

Outlook

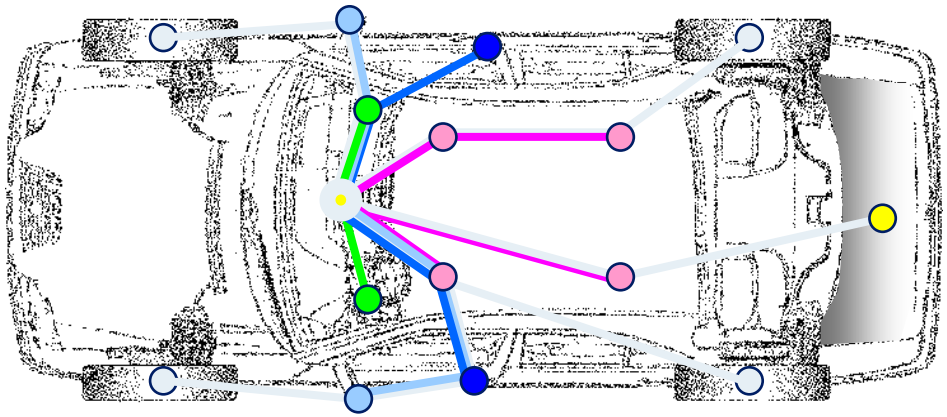
First (simple) examples seem to work …

… but industrial application needs additional effort!

MEDEA+
System Innovation on Silicon

bmvit
Bundesministerium für Verkehr, Innovation und Technologie

bDREAMS



References

- www.systemc.org
  (OSCI members)
- www.systemc–ams.org

  (For information from former SystemC-AMS SG, provides some information for the public)
- *Ch. Grimm, M. Barnasconi, A. Vachoux, K. Einwich*: An Introduction to Modeling Embedded Analog/Mixed-Signal Systems using SystemC AMS Extensions. OSCI, June 2008
- *Ch. Grimm:* Modeling and Refinement of Mixed Signal Systems with SystemC. In: *SystemC: Methodologies and Applications*. Kluwer Academic Publisher (KAP), June 2003.