LET’S GET PHYSICAL: ADDING PHYSICAL DIMENSIONS TO CYBER SYSTEMS

Alberto Sangiovanni-Vincentelli
The Edgar L. and Harold H. Buttner Chair of EECS, University of California at Berkeley
Co-founder, Member of the Board, Cadence Design Systems
McKinsey’s Disruptive Technologies

Twelve potentially economically disruptive technologies

<table>
<thead>
<tr>
<th>Technology</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile Internet</td>
<td>Increasingly inexpensive and capable mobile computing devices and Internet connectivity</td>
</tr>
<tr>
<td>Automation of knowledge work</td>
<td>Intelligent software systems that can perform knowledge work tasks involving unstructured commands and subtle judgments</td>
</tr>
<tr>
<td>The Internet of Things</td>
<td>Networks of low-cost sensors and actuators for data collection, monitoring, decision making, and process optimization</td>
</tr>
<tr>
<td>Cloud technology</td>
<td>Use of computer hardware and software resources delivered over a network or the Internet, often as a service</td>
</tr>
<tr>
<td>Advanced robotics</td>
<td>Increasingly capable robots with enhanced sensors, dexterity, and intelligence used to automate tasks or augment humans</td>
</tr>
<tr>
<td>Autonomous and near-autonomous vehicles</td>
<td>Vehicles that can navigate and operate with reduced or no human intervention</td>
</tr>
<tr>
<td>Next-generation genomics</td>
<td>Fast, low-cost gene sequencing, advanced big data analytics, and synthetic biology (“writing” DNA)</td>
</tr>
<tr>
<td>Energy storage</td>
<td>Devices or systems that store energy for later use, including batteries</td>
</tr>
<tr>
<td>3D printing</td>
<td>Additive manufacturing techniques to create objects by printing layers of material based on digital models</td>
</tr>
<tr>
<td>Advanced materials</td>
<td>Materials designed to have superior characteristics (e.g., strength, weight, conductivity) or functionality</td>
</tr>
<tr>
<td>Advanced oil and gas exploration and recovery</td>
<td>Exploration and recovery techniques that make extraction of unconventional oil and gas economical</td>
</tr>
<tr>
<td>Renewable energy</td>
<td>Generation of electricity from renewable sources with reduced harmful climate impact</td>
</tr>
</tbody>
</table>
Google's robotic cars have about $150,000 in equipment including a $70,000 LIDAR (laser radar) system. The range finder mounted on the top is a Velodyne 64-beam laser. This laser allows the vehicle to generate a detailed 3D map of its environment. The car then takes these generated maps and combines them with high-resolution maps of the world, producing different types of data models that allow it to drive itself.
Computers and mobiles to disappear!

Predictions: 7 trillions devices servicing 7 billion people!
1,000 devices per person by 2025

The Immerged Human
Real-life interaction between humans and cyberspace, enabled by enriched input and output devices on and in the body and in the surrounding environment

Courtesy: J. Rabaey
Intelligent systems that gather, synthesize and apply information will change the way entire industries operate.

**Smart water**
Apply monitoring and management technologies to help optimize the availability, delivery, use, and quality of water as well as related systems including energy and chemical treatment.

**Smart traffic**
Use real-time traffic prediction and dynamic tolling to reduce congestion and its byproducts while positively influencing related systems.

**Smart energy**
Analyze customer usage and provide customized products and services that help to boost efficiency from the source through the grid to the end user.
Another One: BioCyber (?) Systems
Linking the Cyber and Biological Worlds

Examples: Brain-machine interfaces and body-area networks

Courtesy: J. Rabaey
Towards Integrated Wireless Implanted Interfaces

Moving the state-of-the-art in wireless sensing

Power budget: mWs to 1 mW
Vision 2025

- Integrated components will be approaching molecular limits and/or may cover complete walls
- Every object will be smart
- The Ensemble is the Function!
  - Function determined by availability of sensing, actuation, connectivity, computation, storage and energy
- Collaborating to present unifying experiences or to fulfill common goals

A humongous networked, distributed, adaptive, hierarchical, hybrid control problem
THE SWARM

is coming!

IRWIN ALLEN's production of "THE SWARM"
CyberPhysical Systems (CPS)

• The traditional embedded systems problem
  – Embedded system is the union of computing hardware and software immersed in a physical system it monitors and/or controls. The physical system is a given. The design problem is about the embedded system only.

• Hybrid Systems
  – Mixed discrete and continuous time systems

• The CPS problem
  – Cyber-Physical Systems (CPS): Orchestrating networked computational resources with physical systems
  – Co-design of physical system and controller
  – Computation and networking integrated with physical processes. The technical problem is managing dynamics, time, and concurrency in networked, distributed computational + physical systems.
Modeling Cyber-Physical Systems

Model

Equation-based model

Abstraction
“physical modeling”

System

Physical system (the plant)

Sensors

Actuators

Networking

Embedded systems (computation)

Courtesy: D. Broman
Modeling Cyber-Physical Systems


System

Physical system (the plant)

Model

Equation-based model

Abstraction

“physical modeling”

Concept of Time

Different models of computation

C-code

Networking

Sensors

Actuators

Embedded systems (computation)

Courtesy: D. Broman
First Major Challenge on the Cyber Side: Modeling Timing and Concurrency

Correct execution of a program in C, C#, Java, Haskell, etc. has nothing to do with how long it takes to do anything. All our computation and networking abstractions are built on this premise.

Timing of programs is not repeatable, except at very coarse granularity.

Programmers have to step outside the programming abstractions to specify timing behavior.
CPS Complexity and Heterogeneity: Further Challenges  
(Derler, Lee, ASV, Proc. Of the IEEE, 2012)

**Modeling**
- Challenge 2: Modeling Interactions of Functionality and Implementation

**Integration**
- Challenge 3: Keeping Model Components Consistent
- Challenge 4: Preventing Misconnected Model Components
- Challenge 5: Improving Scalability and Accuracy of Model Analysis Techniques

**Specification**
- Challenge 6: Capturing System Requirements
- Challenge 7: Managing Requirements
Integration Challenges: Plug and Play?

Plug and Pray!
The Design Specification/Implementation Nightmare

Specification:
P. Picasso
"Femme se coiffant"
1940

Implementation:
To discover that what it was really wanted
How Did we Cope with Complexity?

Methodologies
(Freedom from Choice)
Coping with Moore’s Law

- **CMOS**
- **Bipolar, NMOS**
- **Intel8080**
- **Intel386**
- **Pentium**
- **Pentium Pro**
- **Pentium III**
- **Pentium Pro**
- **Pentium III**
- **IA-64**

**Feature size (nanometers):**
- 1000nm
- 100nm
- 10nm
- 1nm

**Timeline:**
- 1970
- 1980
- 1990
- 2001
- 2010
- 2020
- 2030
- 2040
- 2050

- **1 million transistors**
- **100 million**
- **10 billion**
General Principles
Traditionally complexity has been managed by two basic approaches:

- Decomposition/Composition: reduce the number of items to consider by breaking the design object into semi-independent parts (*divide et impera*)
- Abstraction/Refinement: reduce the number of items by aggregating objects and by eliminating unnecessary details with respect to the goal at hand

Complexity is also managed by “construction”

- Constrain “artificially” the space (regular layout, synchronous designs)
- Start high in the abstraction layers and define a number of refinement steps that go from the initial description to the final implementation
Status Quo: There are several areas where change may be fruitful

**Planning**
- Constrain: N/A
- Optimize: Cost

**Programmin**
- Constrain: Performance
- Optimize: Cost

**Budgeting**
- Constrain: Cost
- Optimize: N/A

**Execution**

**Source Selection**
- Constrain: Performance
- Optimize: Cost

**Systems Engineering**
- Constrain: Performance
- Optimize: SWaP

**Verification & Validation**
- Constrain: Performance
- Optimize: N/A

---

**Cost-centric acquisition process provides dis-incentives to incorporation of flexibility and adaptability into system designs**

- **Design process arbitrarily decomposes system and largely ignores complexity — undesired and multi-mode interactions and emergent system behaviors**

**SWaP = Size, Weight, and Power**

**V&V = Verification & Validation**

**Undesirable interactions (thermal, vibrations, EMI)**

**Desirable interactions (data, power, forces & torques)**

**Conventional V&V techniques do not scale to highly complex or adaptable systems (i.e., those with large or infinite numbers of possible states/configurations)**

**MILD**

MIL-STD-499A (1969) systems engineering process: as employed today
The Platform Concept

(ASV, Quo Vadis, SLD? Reasoning About the Trends and Challenges of System Level Design, Proc. of the IEEE, 2007.)

• Meet-in-the-Middle Structured methodology that limits the space of exploration, yet achieves good results in limited time
• A formal mechanism for identifying the most critical hand-off points in the design chain
• A method for design re-use at all abstraction levels

An intellectual framework for the complete electronic design process!

See AUTOSAR, Intel, National Instruments, Cadence, Synopsys, UTC, GM, Magneti Marelli, ELT, Xilinx,....
Separation of Concerns: Keep the What Separated from the How (see AUTOSAR)

Development Process:
- Specification
- Analysis
- Implementation

Behavior Components:
- C-Code
- Matlab
- Dymola

Virtual Architectural Components:
- CPUs
- Buses
- Operating Systems

Behavior Platform:

Mapping

Performance Analysis

Refinement

Evaluation of Architectural and Partitioning Alternatives

IPs

C-Code

Matlab

Dymola

ECU-1

ECU-2

ECU-3

Bus
Platform-Based Design

Platform: library of resources defining an abstraction layer with interfaces that identify legal connections

- Resources do contain virtual components i.e., placeholders that will be customized in the implementation phase to meet constraints
- Very important resources are interconnections and communication protocols
The Key to Platform Based Design

- Components
- Composition rules
- Refinement rules
- Abstraction rules
Approach:
A Platform-Based Design Flow With Contracts

Abstraction

Requirement Formalization

Refinement

Rules

Contracts

Abstraction

Composition

Rules

Performance

Reliability

Safety

Application

System Specification

System Requirements

Synthesis (Optimization)

Behavioral and Non-Functional Models

Architectural Space

Platform Library

Sensors

Actuators

Networks

Processors

Controllers
Compositional Reasoning

Reliably derive global properties of systems based on local properties of components

- Contracts as Assume/Guarantee pairs
- Component properties guaranteed under a set of assumptions on the environment
- Composition valid if and only if all assumptions are satisfied

PRIOR WORK:
- Lamport ‘83
- McMillan ‘97
- Clarke ‘98
- Henzinger ‘01
- Giese ‘00
- Damm, ASV ‘08
- ASV, Damm ‘11
- Henzinger ‘08
- Misra ‘81
Refinement and Abstraction

Vertical Contracts

• So far, contracts for components at the same level of abstraction
  – We refer to this kind of composition as horizontal, and we talk about horizontal contracts

• A component could express assumptions and guarantees w.r.t. another level of abstraction
  – For instance, it may assume an execution environment with certain properties or performance
  – Likewise, it may guarantee certain patterns of activation to the execution environment
  – Contracts that span different levels of abstraction are referred to as vertical contracts
Summary

We need a design and integration platform

• To deal with heterogeneity:
  – Where we can deal with Hardware and Software
  – Where we can mix digital and analog, cyber and physical
  – Where we can assemble internal and external IPs
  – Where we can work at different levels of abstraction

• To handle the design chain

• To support integration
  – Tool integration
  – IP integration

Platform-Based Design with Contracts can be the foundation for this platform
The Problem: Unprecedented Power Requirements in Future Aircraft

Today

- Power sources/sinks
- Electric distribution
- Control system

Complex, heterogeneous system with multiple overlapping time scales

- Power sources ~1
- Loads ~100
- Peak power ~ 400kW

Future

- Power sources ~10
- Loads ~1000
- Peak power ~ 4MW

Platform-Based Design enables architecture exploration (tradeoff weight, stability, ...)

Incremental conservative design
Steady state worst-case power draw
2x overdesign results in weight penalty

Redesign

Power System Architecture
Control System Architecture
Hardware, Software, Communications

Robust design for distributed control system

Tools enable robust design of complex dynamical systems

Dynamics, control, communication latency addressed in all layers

Comms latency impacts stability

Constraints Propagation Performance Abstraction

Application Space

Implementation Space (Platform)

Level 1: Physical layer
Level 2: Data link layer

Courtesy: Hamilton Sundstram (UTC)
PBD for Buildings: Systems of Systems Co-Design

New design paradigms: Integrated multi-domain models analyzed in multiple levels

topology ↔ mechanical/electrical systems ↔ multi-scale controls ↔ sensors + networks

Building Siting

Mechanical and Electrical Systems Architecture

Control Architecture

Mechanical and Electrical Systems

Control Design

Network Design

Program Research Thrusts
• Choice of Layers
• Design Flow/Methodology
• Co-simulation Environment
• Pilot Studies

Final implementation

specifications

constraints

specifications

constraints

specifications

specifications

Courtesy UTC

No bitstuffing

Extact bit stuff.

worst-case

0

0.0002

0.0004

0.0006

0.0008

0.001

0.0012

0.0014

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Packet #

seconds
The Swarm as a Platform

Presenting a uniform API to Apps Developers (similar to trends in the Cloud)

[J. Rabaey, VLSI '11]
συνθήσεις n. 1.a. the combination of separate elements to form a coherent whole.

- Synthetic biology seeks, through understanding, to design biological systems and their components to address a host of problems that cannot be solved using naturally-occurring entities
- Enormous potential benefits to medicine, environmental remediation and renewable energy
Off-the-shelf parts?

Microbial Synthesis of Artemisinin

AcCoA \rightarrow HMG-CoA \rightarrow AcAcCoA \rightarrow Mev \rightarrow Mev-PP \rightarrow Mev-P \rightarrow IPP \rightarrow FPP

\textbf{AcCoA}: Adenylate

\textbf{HMG-CoA}: Hydroxymethylglutaryl-CoA

\textbf{AcAcCoA}: Acylated Adenylate

\textbf{Mev}: Mevalonate

\textbf{Mev-PP}: Mevalonate Phosphate

\textbf{Mev-P}: Mevalonate Phosphate

\textbf{IPP}: Isopentenyl Diphosphate

\textbf{FPP}: Dimethylallyl Diphosphate

\textbf{AMS}: 3-Phospho-D-manno-octulosonic acid

\textbf{tHMGR}: Methylenenitulopantoic Acid kinase

\textbf{HMGR}: HMG-CoA Reductase

\textbf{PMK}: Phosphomevalonate Kinase

\textbf{MPD}: Methylerythritol 4-phosphate Decarboxylase

\textbf{ispA}: Isopentenyl Diphosphate Isomerase

\textbf{ADS}: 2-Cyclohexene-1,1-diphosphate Synthase

\textbf{AMO}: 2-Cyclohexene-1,1-diphosphate Cyclase

\textbf{CPR}: Cytochrome P450 Reductase

\textbf{Artemisinin}
Platform-based Design Environment for Synthetic Biological Systems
Douglas Densmore (EECS),
J.Christopher Anderson (Bioengineering),
Alberto Sangiovanni-Vincentelli (EECS)

Clotho is a design environment for the creation of biological systems from standardized biological parts.

- Clotho is a design environment for the creation of biological systems from standardized biological parts.
- Composed of “views”, “connectors”, “interfaces” and “tools”
- Versions available at http://cidarlab.org/software-overview/.
Final Words of Wisdom
The Way Forward for CPS

- *Everything is Connected*: Society, Electronic and System Industry facing an array of complex problems from design to manufacturing involving complexity, power, reliability, re-configurability, integration....
- Complexity is growing more rapidly than ever seen
- Interactions among subsystems increasingly more difficult to predict
- Pre-existing systems put to work to provide new services
- *Need work at all levels*: Methodology, Modeling, Tools, Algorithms

- Deep collaboration among
  - Governments, industry, and research centers
  - Different Disciplines: Control, Communication, Computer Science, Electrical Engineering, Mechanical Engineering, Civil Engineering, Chemistry, Biology.....