Dr Ben Black – Systems Engineer

National Instruments

ben.black@ni.com

Agenda

- Trends in Automotive Electronics
- Flexible HIL Solutions
- High Speed Deterministic Data Transfer
- Distributed HIL
- Discontinuous Simulation Solvers

The “Good”

The “Bad”

The “Ugly”

Global Automotive Industry Trends

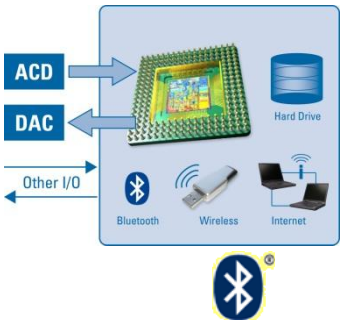
- Consumer electronics technologies in vehicles
- Alternative Energy
- Cost Reduction
- Reduced Emissions
- Increased Safety
- Differentiation through Features
- Global Design and Manufacturing

The “Ugly”

Automobiles Then and Now…

Mechanics and hydraulics	⟷	Electromechanics
3 ECUs	⟷	15 to 80 ECUs
AM/FM radio	⟷	Telematics (Infotainment)
Relay-control units	⟷	Power-control units
CAN	⟷	CAN, LIN, FlexRay, ...

…and Software defines the Functionality



Engine control unit

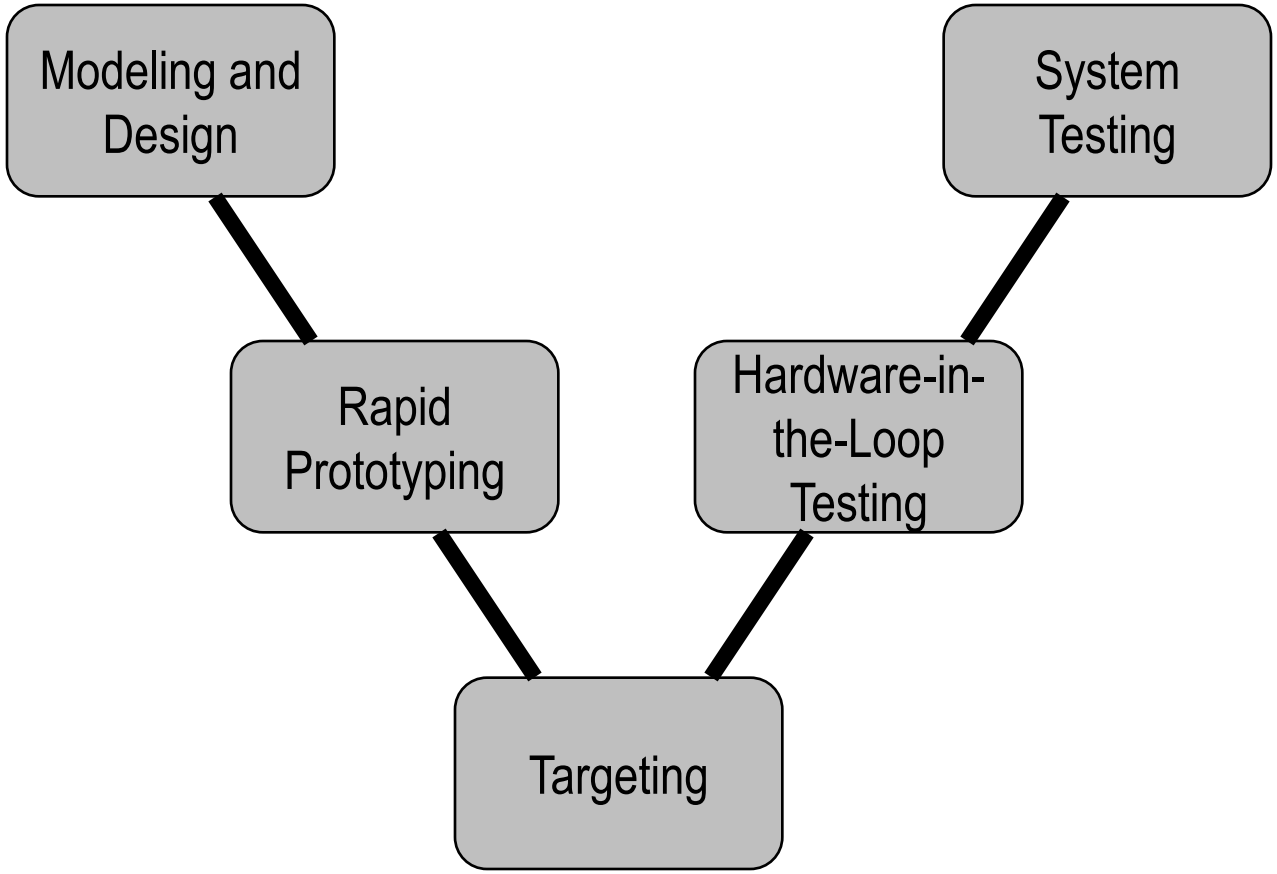
Software-Based Hardware Designs

"For the next 10 years an increase of 10%-15% of software in the share of costs of a vehicle is forecasted every year"

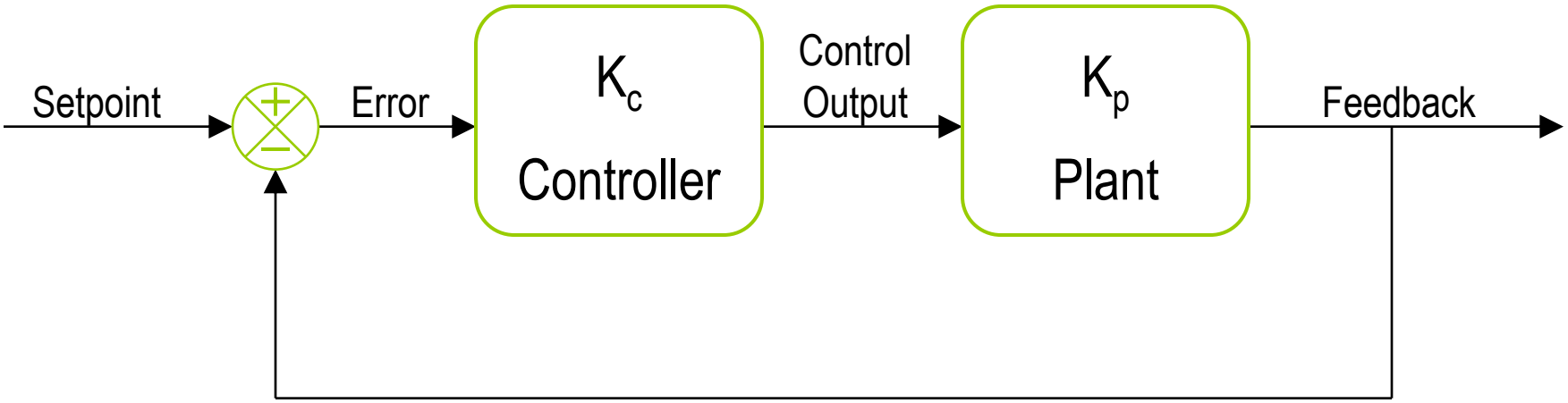
- McKinsey&Company Study

- **Benefits**
  - **Rapid Advancements**
  - **New and Improved Functionality**
  - **Lower Cost**
- **Challenges**
  - **More functions to test**
  - **More measurements to make**
  - **Unique functions to test**

Control Design Process

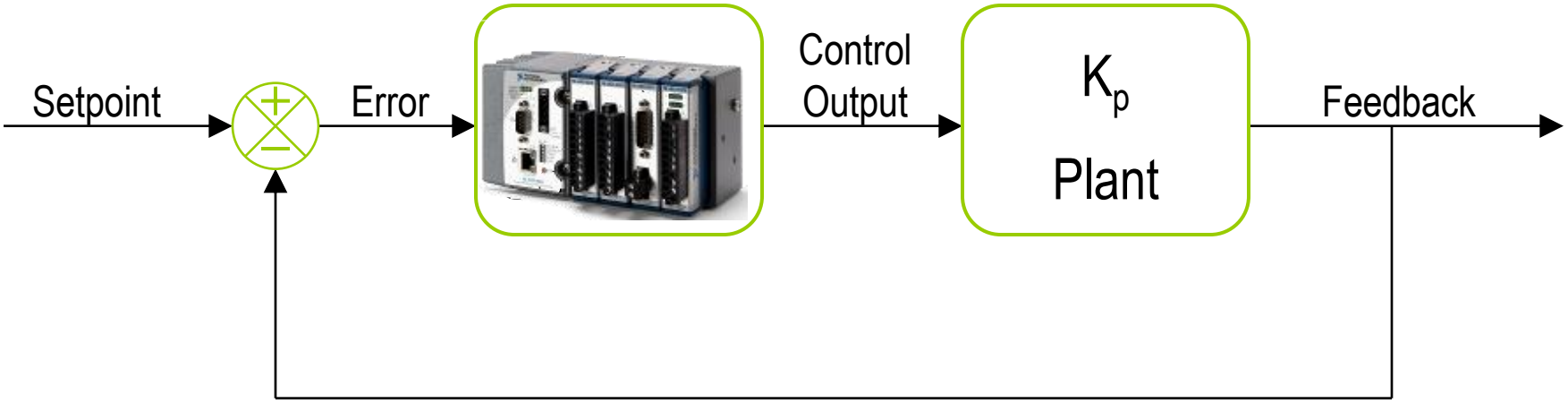


Modeling and Design



Modeling and Design Produce Controller and Plant Models

Rapid Control Prototyping

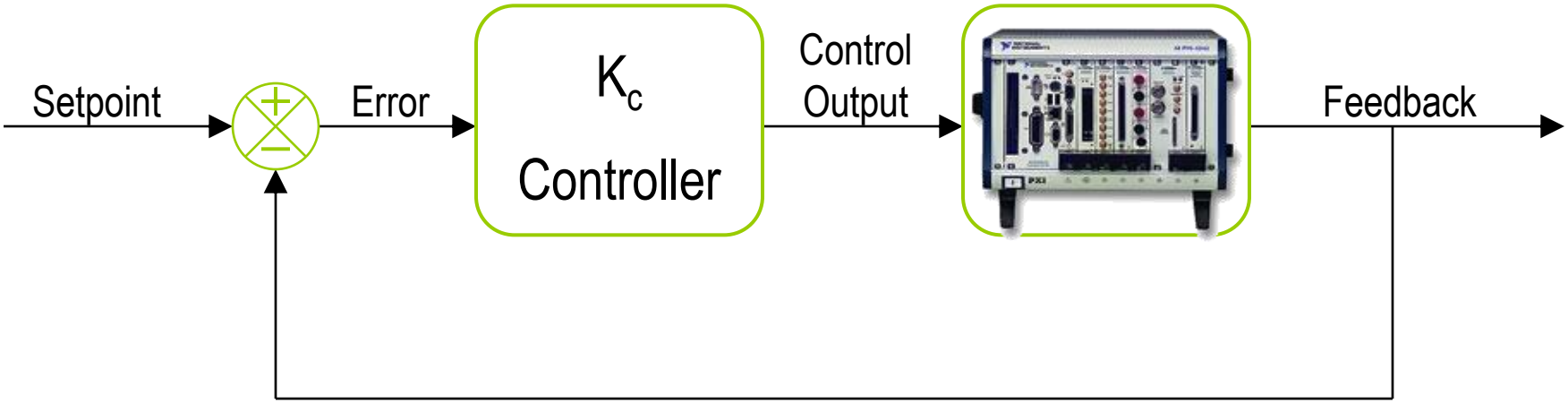


Creating a Functional Prototype of the Controller

Rapid Control Prototyping Example

Drivven: "We prototyped a full-authority engine control system … in just 3 man-months. In past projects, it took us at least 2 man-years and over $500,000 to develop similar ECU systems."

Hardware-in-the-Loop Simulation



Testing Production Controller with Simulated Plant

What is HIL?

- The use of real-time I/O hardware to simulate the dynamic behavior of a device that interfaces to the unit under test.
  - Dynamic – stimulus reacts to the response of the UUT (closed-loop)
  - Static – stimulus ignores the response of the UUT (open-loop)
- The simulator may use programming languages, state charts, modeling languages or other methods to describe the input/output behavior (dynamics) of the device

What is HIL?

- Types of test
  - Functional
  - Parametric
  - Validation (V&V)
  - Durability (HALT/HAST)
  - End-of-line
  - …
- Methods in test
  - Temperature/power variation
  - Salt/sand spray
  - EM radiation
  - Loading/resistance
  - HIL Simulation
  - …

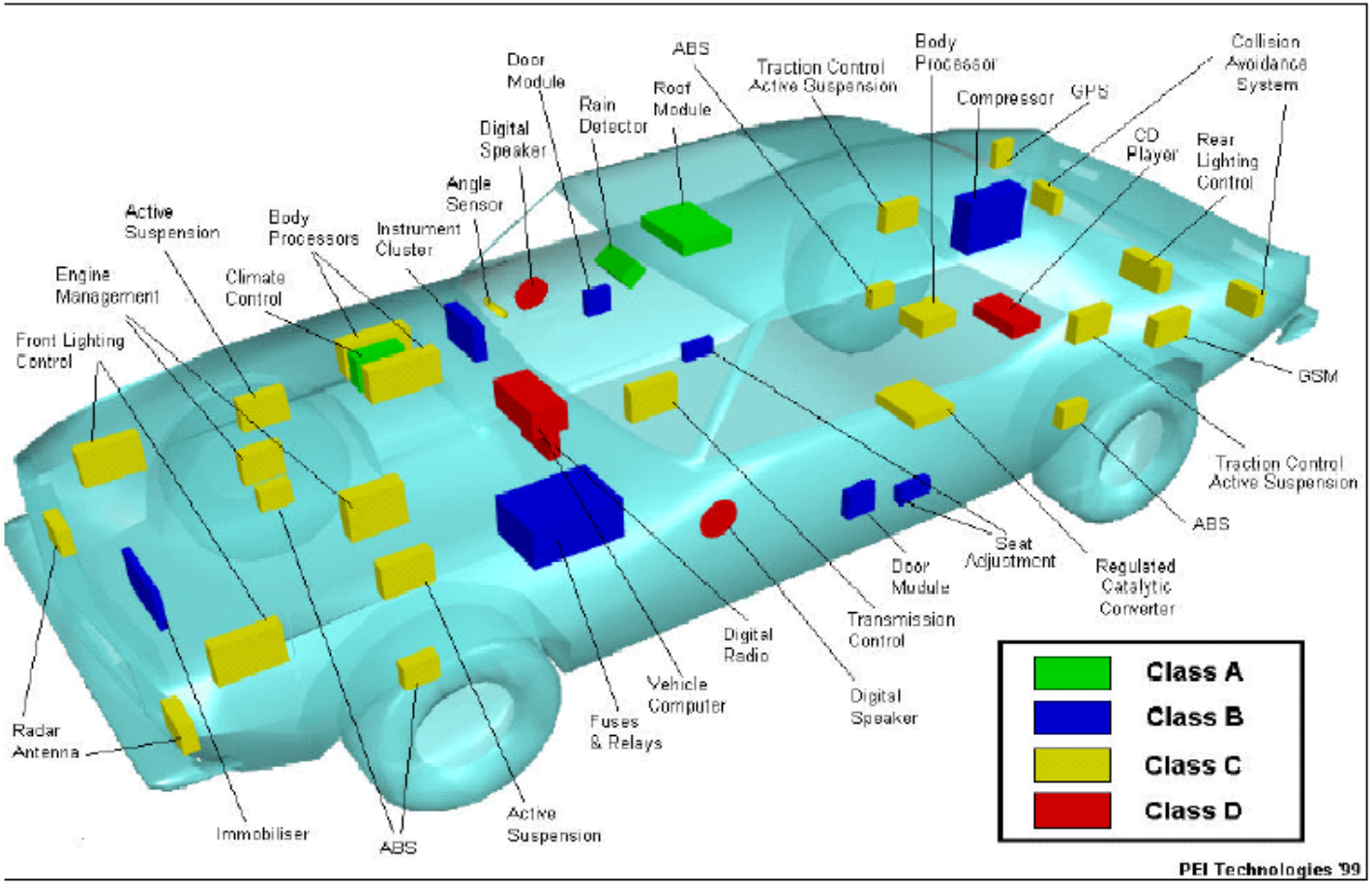
The “Good”

The “Bad”

The “Ugly”

Example…the automobile

3 ECUs → 15 to 70 ECUs in 10 years



Automotive Electronics vs. CO2 Consumption

The effect...

- Electronic causes 5% of a cars CO2 Emissions
- State of the art Infotainment System:
  - 4-6 Ampere ≈ 0,1 Liter gasoline ≈ 2 Gramms CO2
- Innovation vs. CO2 Reduction

The “Good”

The “Bad”

The “Ugly”

HIL for ECU Test

Challenges –

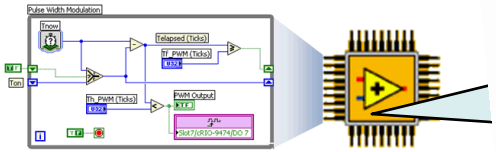
- Modularity: No method of drag-and-drop ECU hardware architecture
- Flexibility: Difficult to add or swap ECUs in a current test configuration
- Wiring/Cabling: Direct I/O wiring makes re-wiring tedious and time-consuming
- Cost: Significant loss of “up-time”

Flexible HIL Solutions

- I/O with Deterministic Data Transfer
- Integrated Signal Conditioning
- High Resolution Measurements (up to 24 bits)
- Flexible and Modular ECU/HIL Testing Environment
- Distributed Simulation

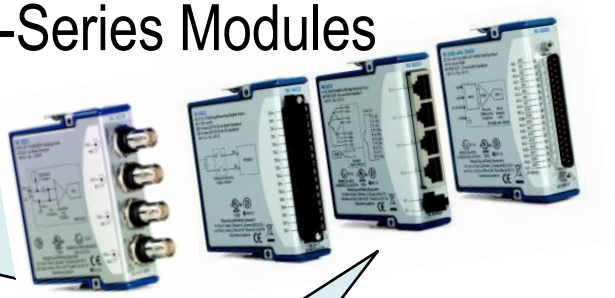
FPGA and Reconfigurable IO

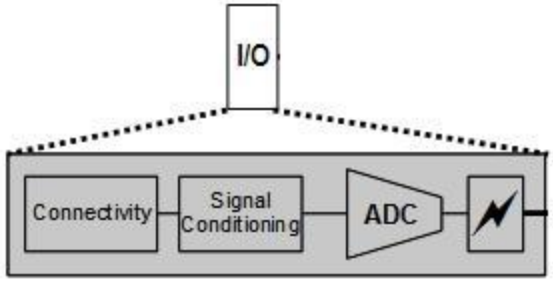
FPGA on cRIO Backplane



8-Slot cRIO

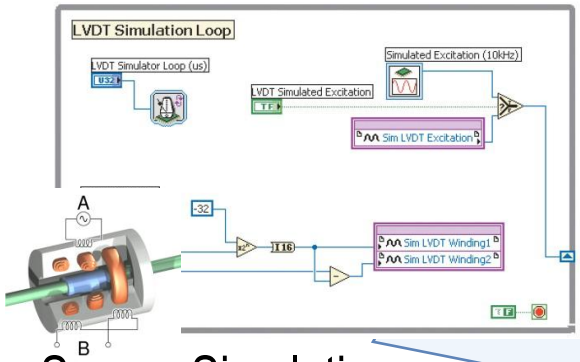
C-Series Modules



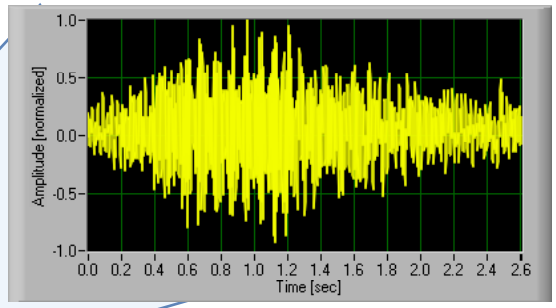


ADC and Integrated Signal Conditioning

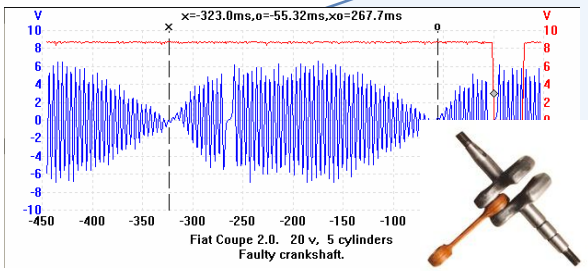
FPGA and Reconfigurable IO



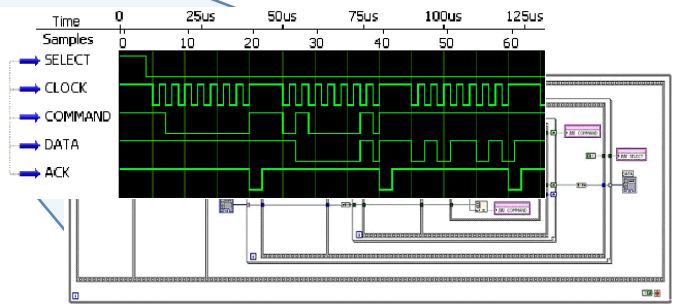
Sensor Simulation (LVDT)



Knock Signal Generation

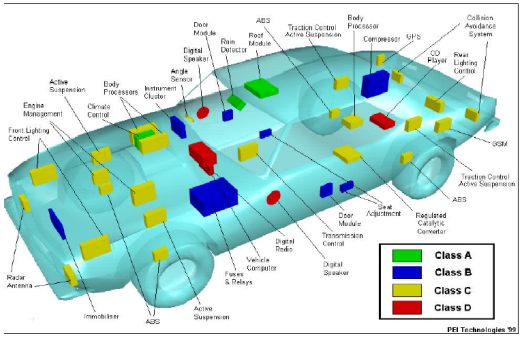


Crankshaft Simulation



Custom Serial Protocols

HIL for ECU Test



Up to 100 ECUs for a single solution

I/O Points

AI, AO, DI, DO, CAN

Real-Time Processors

FPGA in HIL ECU Test

Bring the I/O Nodes to the ECUs

ECU

NI FPGA Backplane with
I/O Modules

FPGA in HIL ECU Test

Rack-Mount Controller

ECU with I/O Module

LabVIEW Real-Time

PXI

Real-Time Processor

Desktop or Industrial PC

High Speed Deterministic Data Transfer

- Master/Slave Architecture
- Expandable I/O
- Optimized for Single-Point Industrial Data Transfer
- Predictable Timing and Precise Synchronization
- Masters Use Off-the-Shelf Ethernet Interface
- Continuous Data Flow Through Multiple Slaves
- High Bandwidth Efficiency

High Speed Deterministic Data Transfer

NI Masters

PXI

NI 8353 Rack-Mount RT

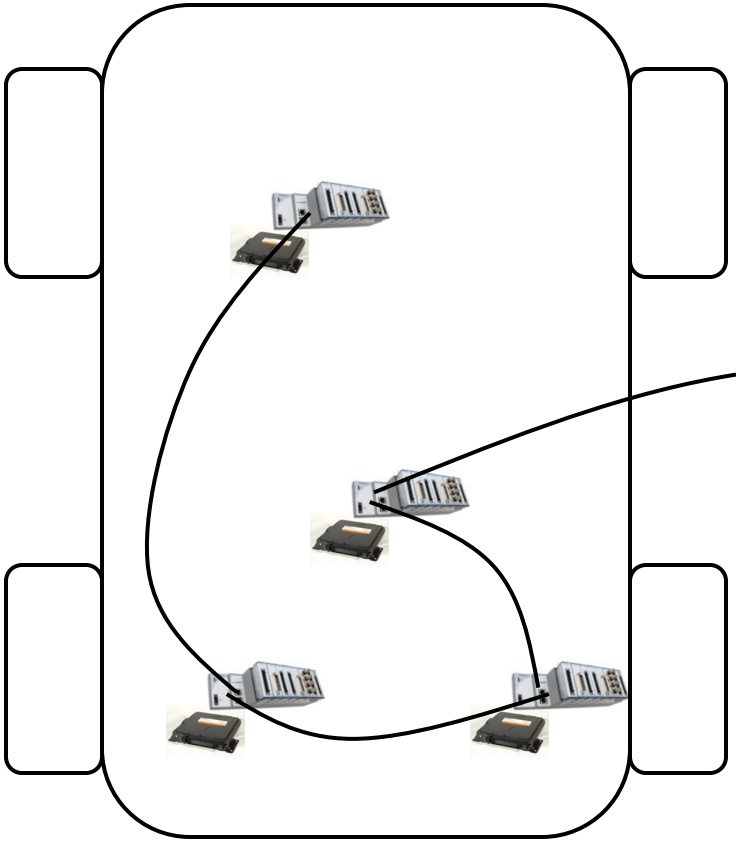
NI Slaves

8-Slot cRIO

Smart Camera

Future

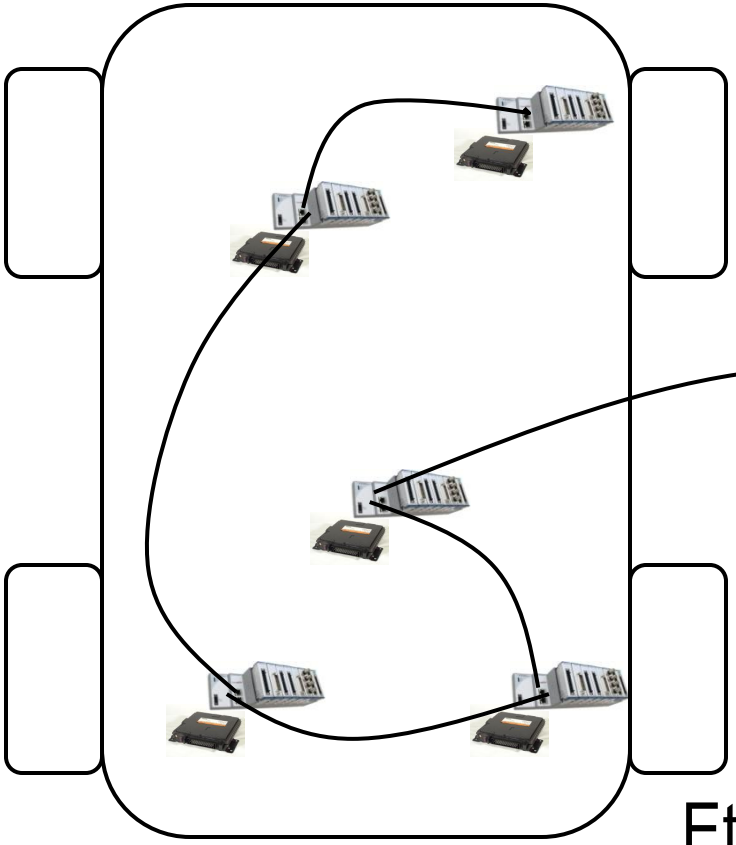
Flexible HIL Solutions



Need to add another ECU?

Real-Time Processor (Master)

Flexible HIL Solutions

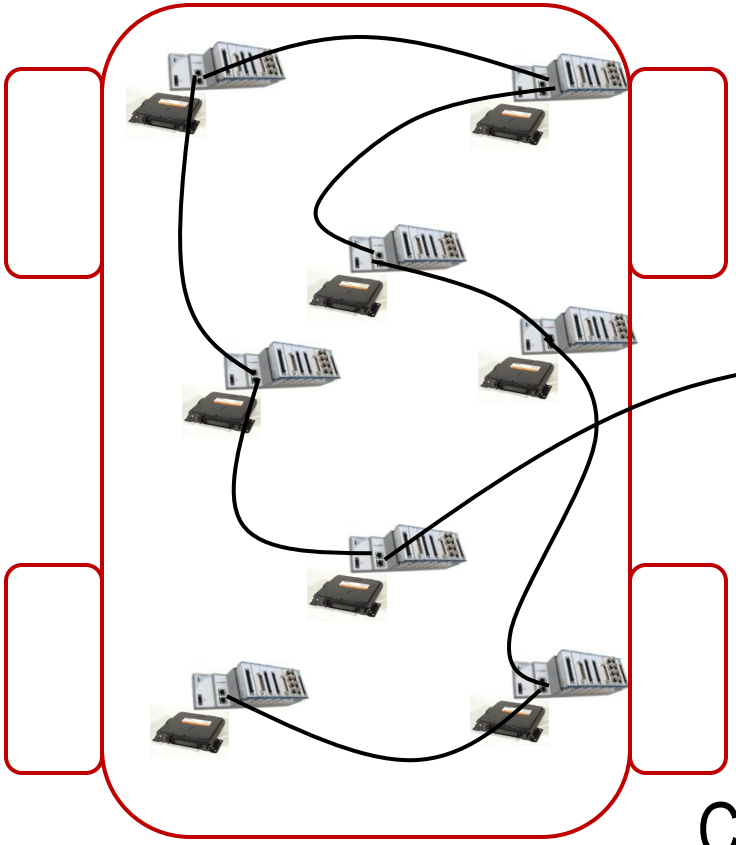


Add another Slave to the chain

Real-Time Processor (Master)

Ethernet cables make re-wiring simple

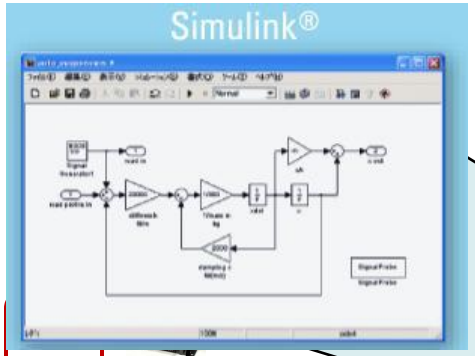
Flexible HIL Solutions

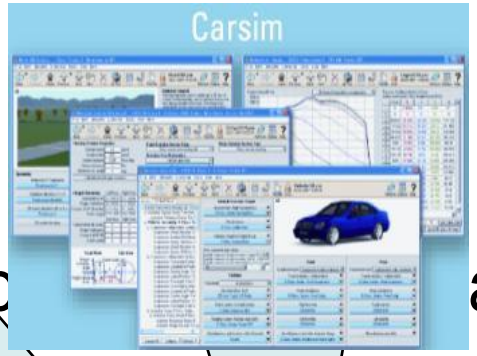


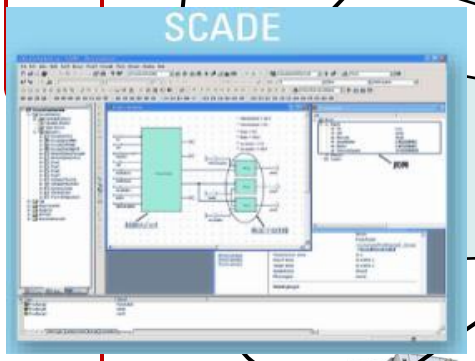
Need to test a different car?

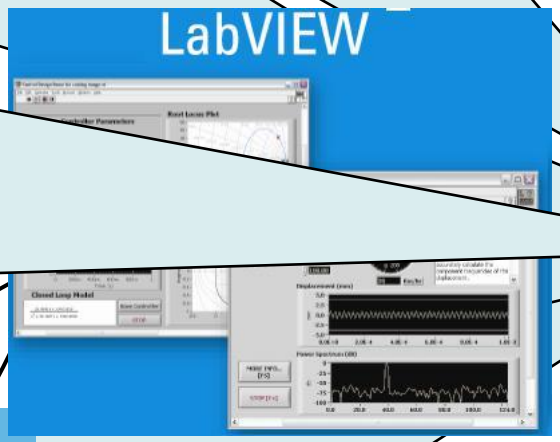
Real-Time Processor (Master)

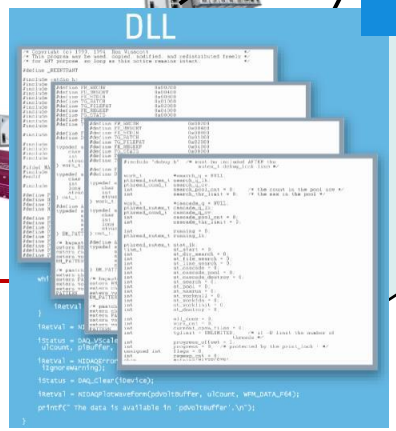
Change out ECUs as needed with pre-assembled ECUs and I/O modules







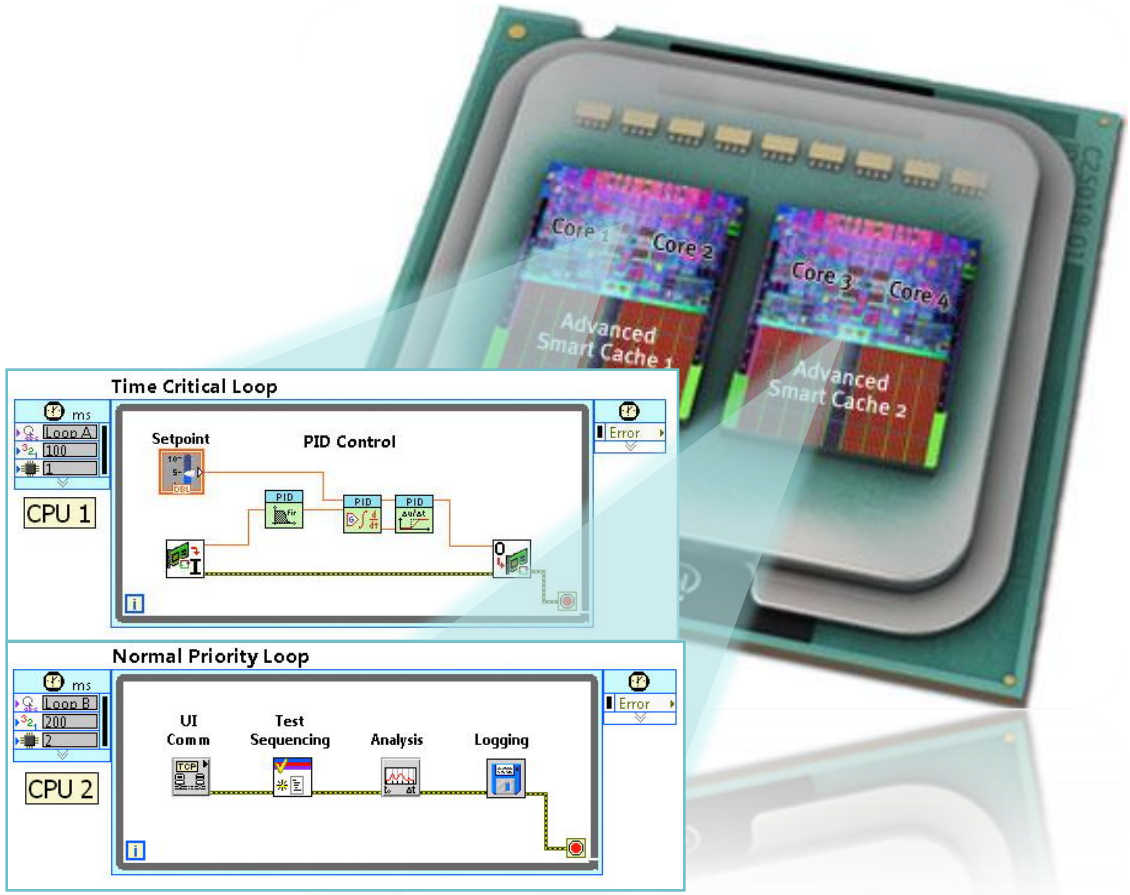




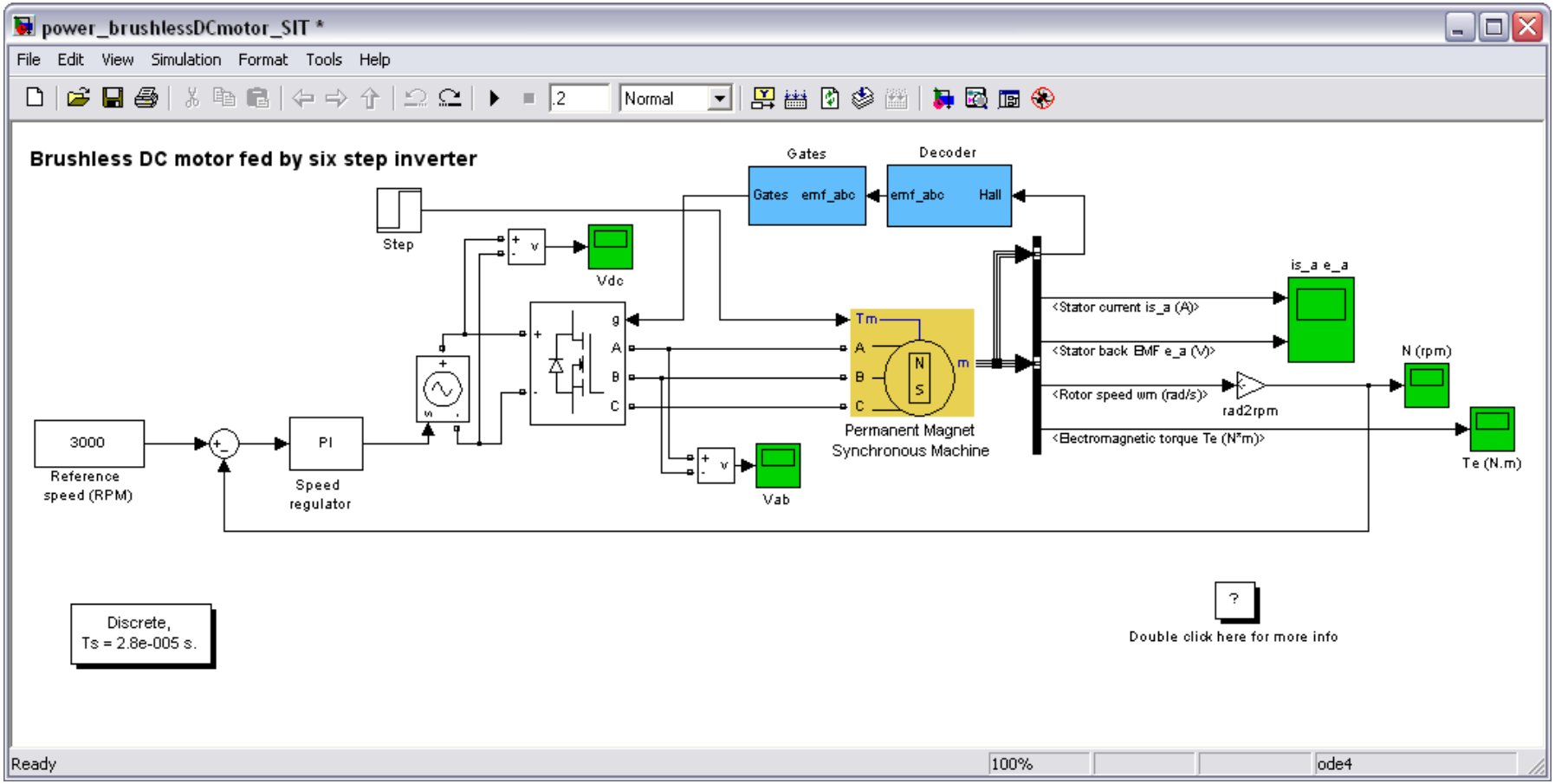
Real-Time Processor (Master)

Use same RT Processor, just switch ECU software models

Distributed Simulation



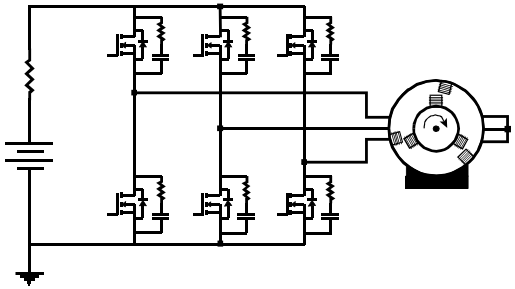
Discontinuous Simulation Solvers



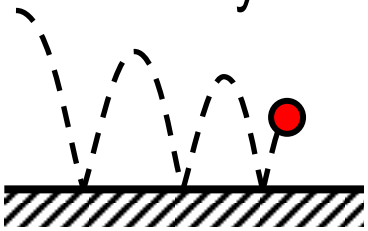
Summary of Computer Simulation

	What	Why	How
Off-line	Plant (dynamic system) Controller	Design prototype controller Investigate behavior	Variable step : for precision Fixed-step : for speed
Real-Time	Plant (dynamic system)	Validate prototype controller Field diagnostic tool	Fixed step - HIL

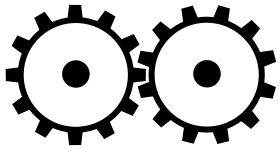
Dynamic Systems ⟷ Discontinuous Systems



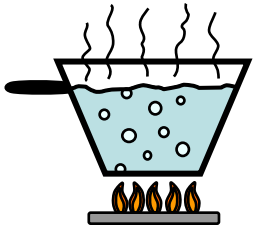
Electrical Systems



Physical Systems



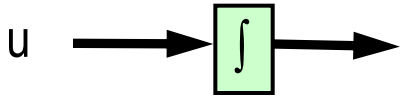
Mechanical Systems



Chemical Systems

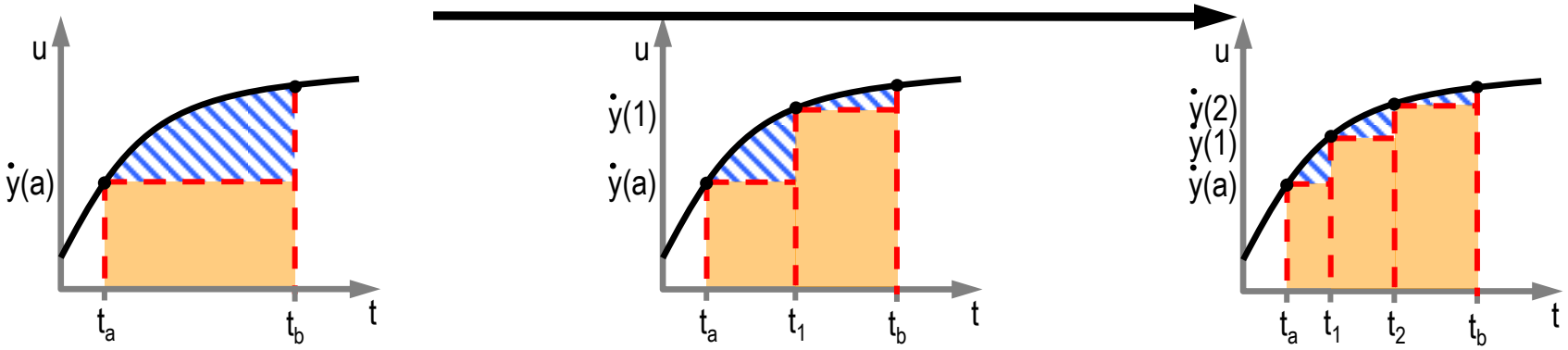
Simulation of Dynamic System

Dynamic system ⟷ **Differential equations**



$\dot{y}(t) = u(t)$

Variable step approximation Solved through error control

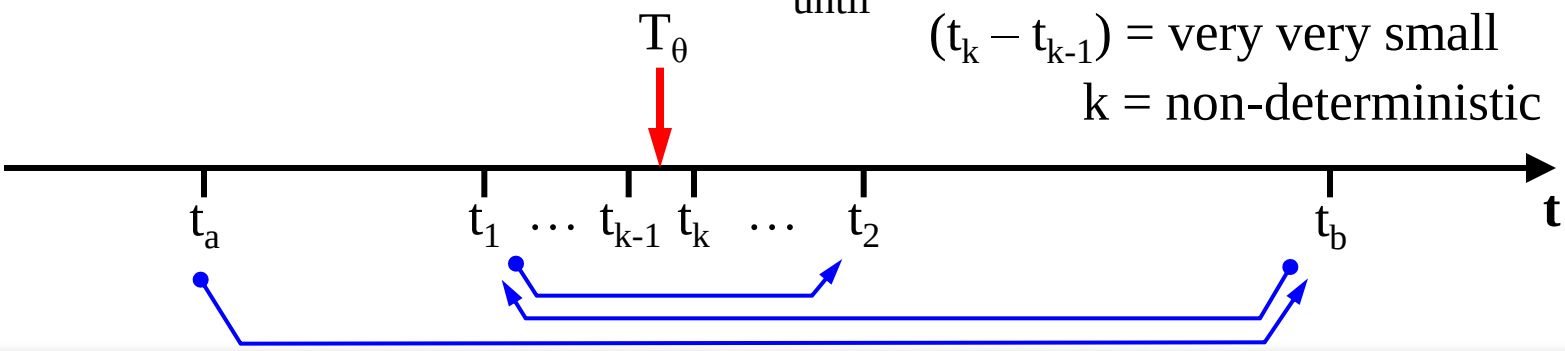


In the presence of a discontinuity :

Iteratively locate the discontinuity until

$(t_k - t_{k-1})$ = very very small

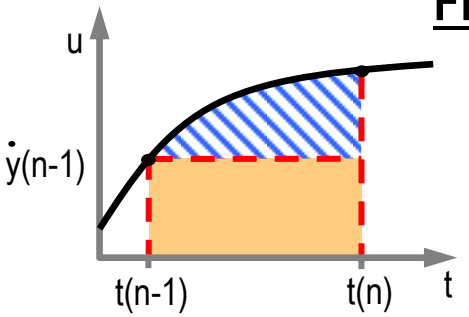
k = non-deterministic



Simulation of Dynamic System

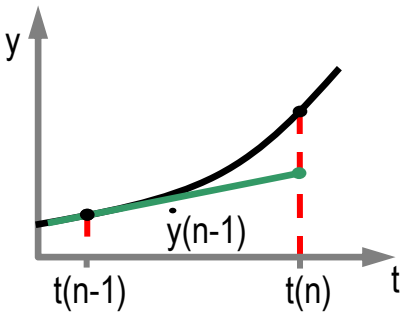
Fixed-step approximation

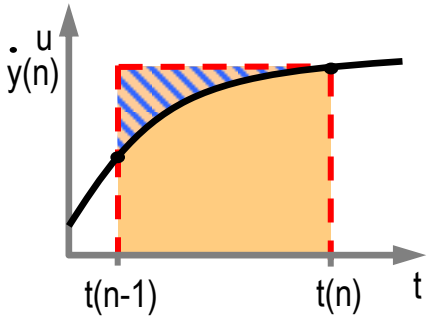
1 step methods



Forward Euler

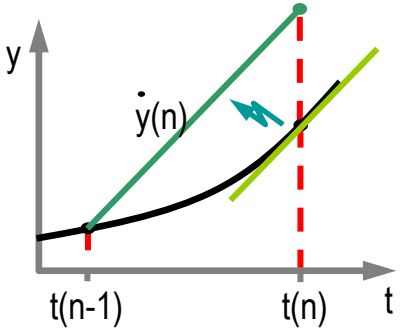
$y(n) = y(n-1)+T\dot{y}(n-1)$

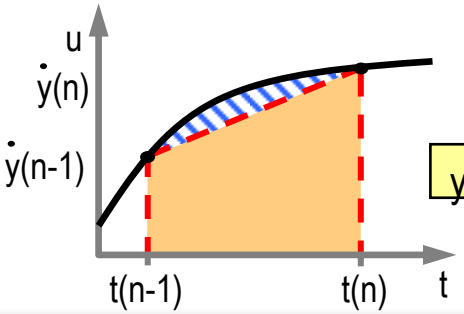




Backward Euler

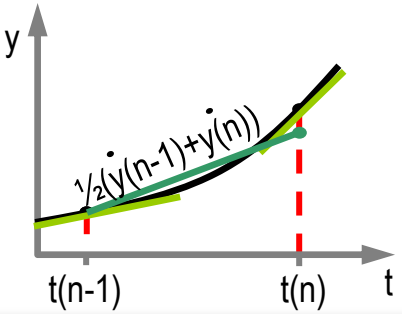
$y(n) = y(n-1)+T\dot{y}(n)$





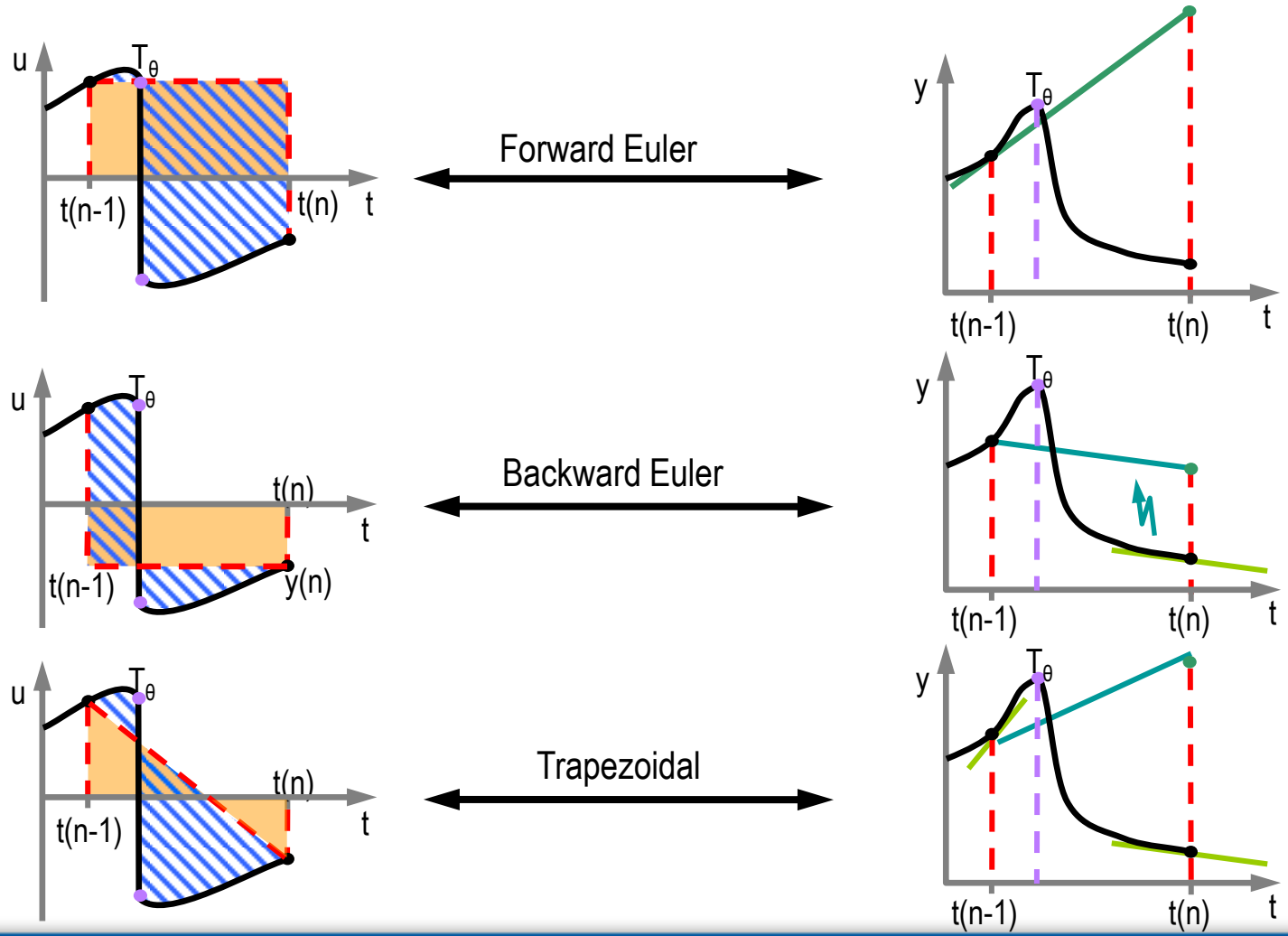
Trapezoidal

$y(n) = y(n-1)+\frac{1}{2}T(\dot{y}(n)+\dot{y}(n-1))$



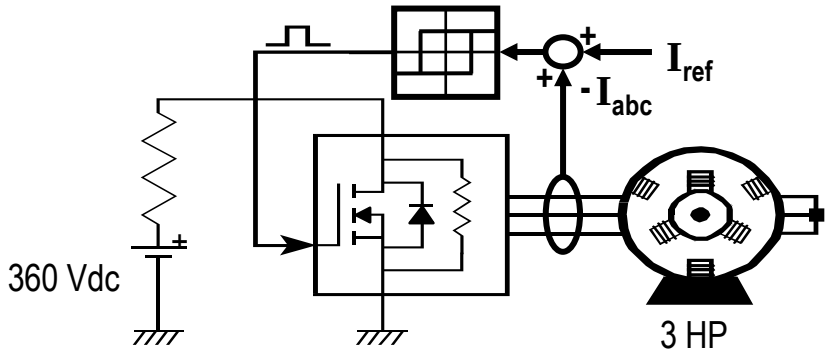
Simulation of Dynamic System

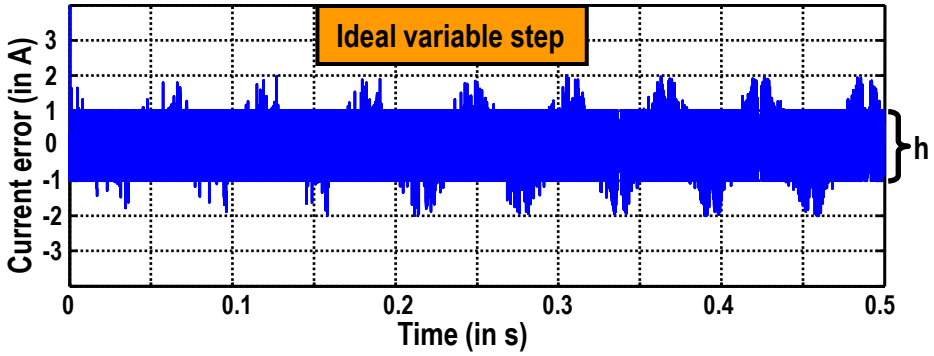
In the presence of a discontinuity :

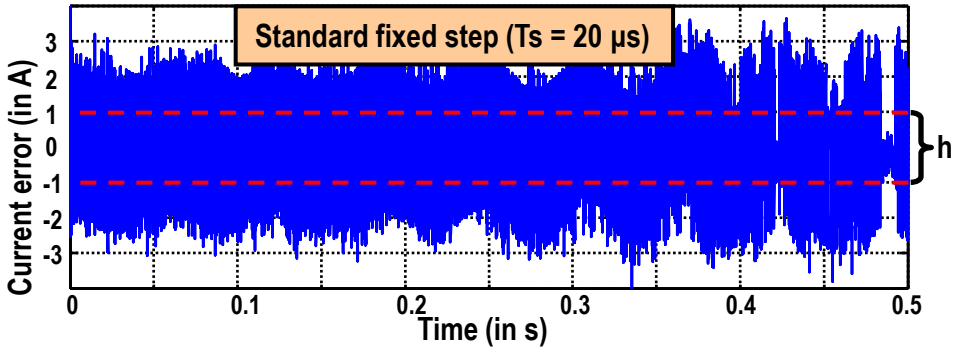


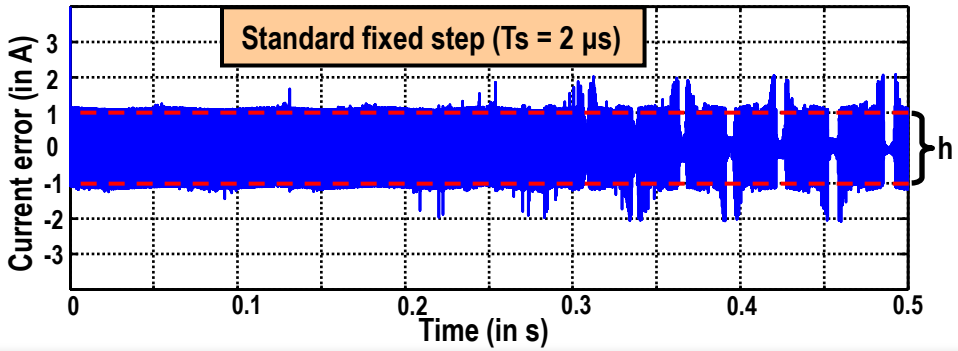
Simulation of Dynamic System with Discontinuities

Impact of discontinuity errors : Electrical system example



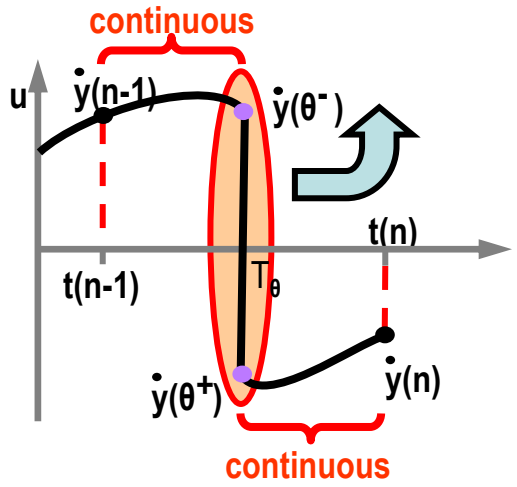






Fixed step simulation of discontinuous systems

Discontinuous system ⟷ Piecewise continuous system



Boundary : special calculation

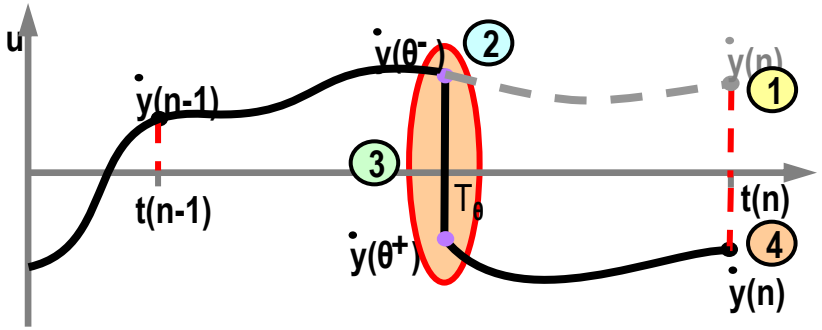
Problems :

- Where is the boundary ?

$T_\theta = ?$

- What happens at the boundary ?

In fixed step :



1 – Calculate y(n) (discontinuity undetected).

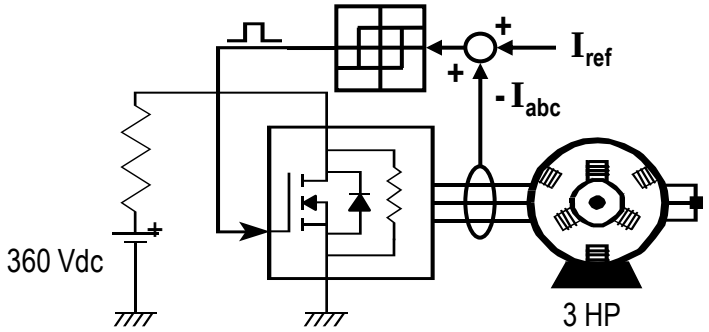
2 – Detect the discontinuity and determine T_θ.

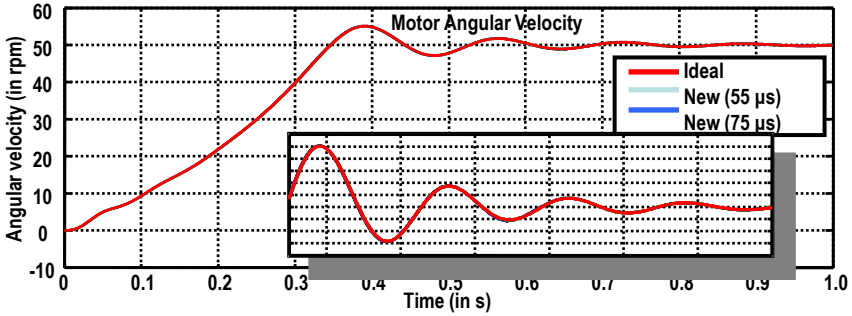
3 – Process the boundary : special calculation.

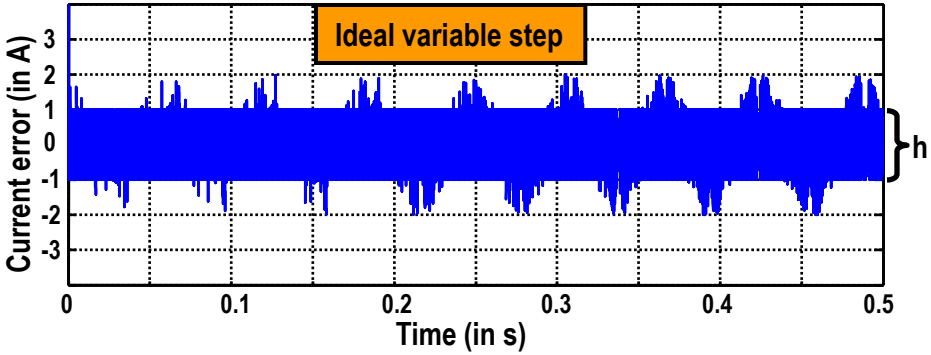
4 – Recalculate y(n).

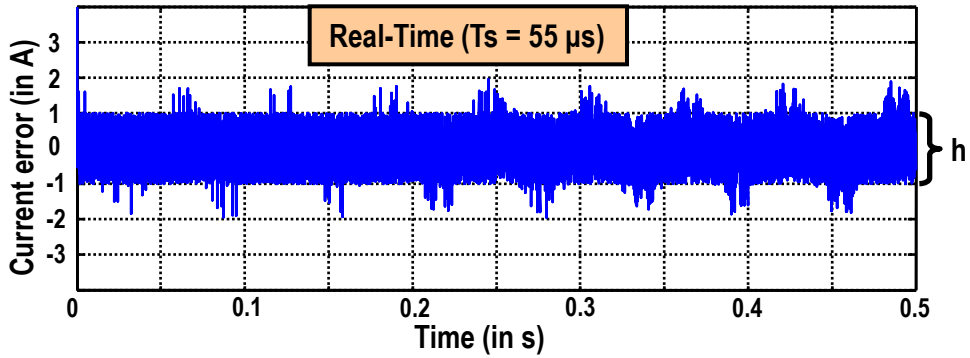
Real-Time Simulation of Power Electronics Circuits

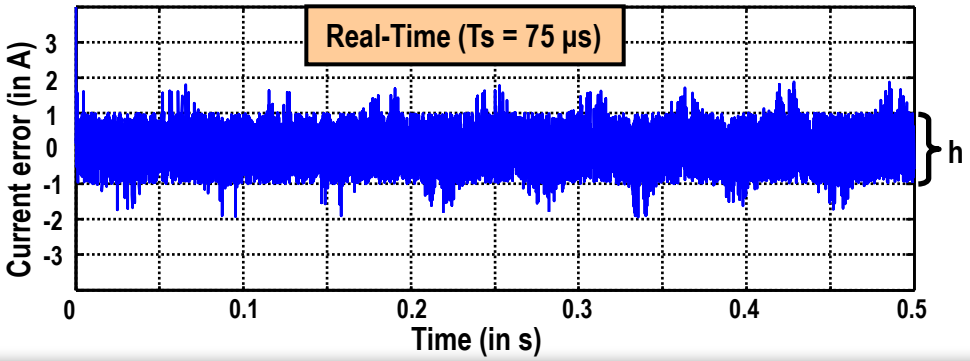
Electric Drive Test Bench











The “Good” Approach

- ✓ Flexible solutions
  - Customize software with LabVIEW
  - Customize hardware with FPGA
  - Integrate I/O nodes quickly and easily
  - Distribute the simulation
- ✓ Improved Solver