Component PERCUSSION

episode II

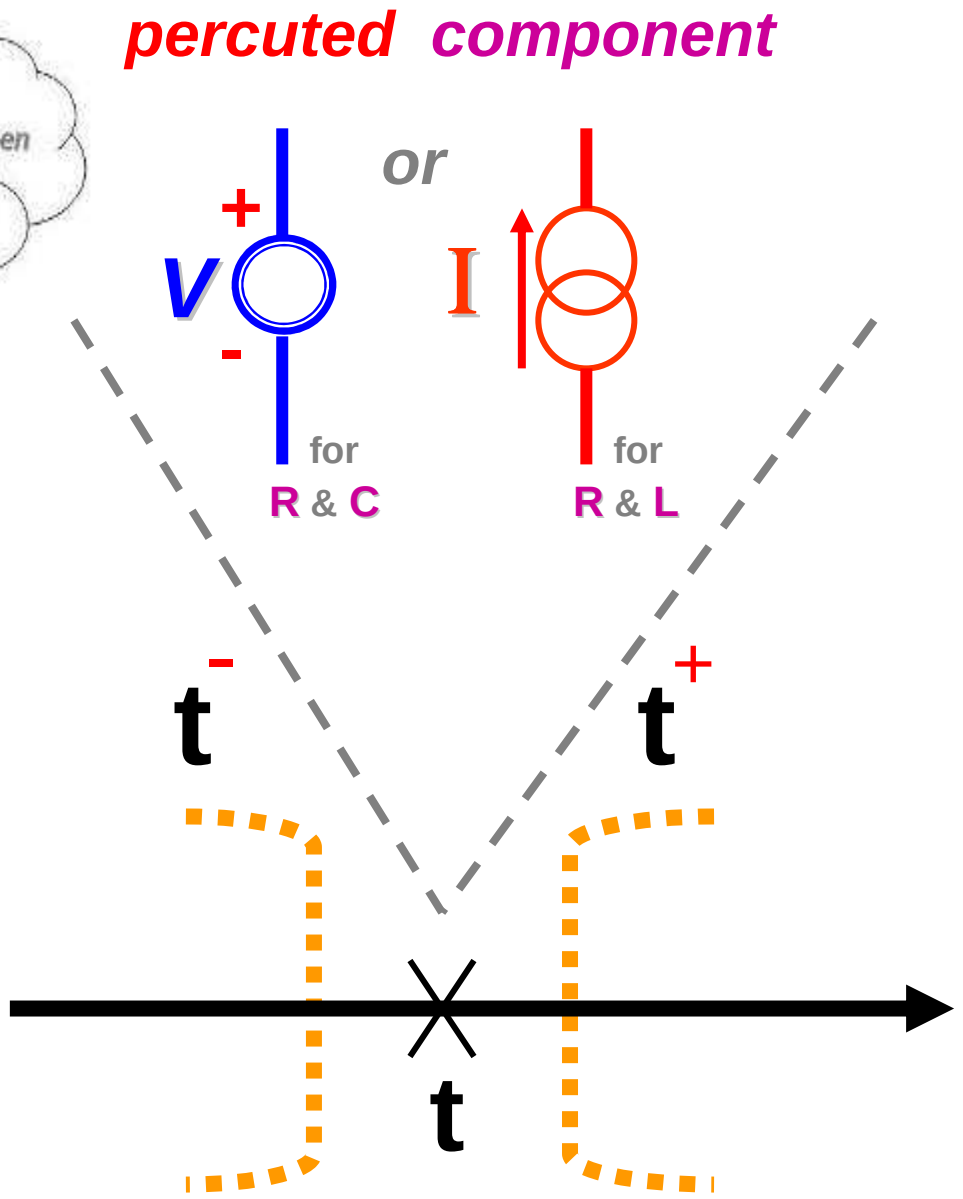
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

In the previous episode …

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Differential equations

by means of **Distributions**

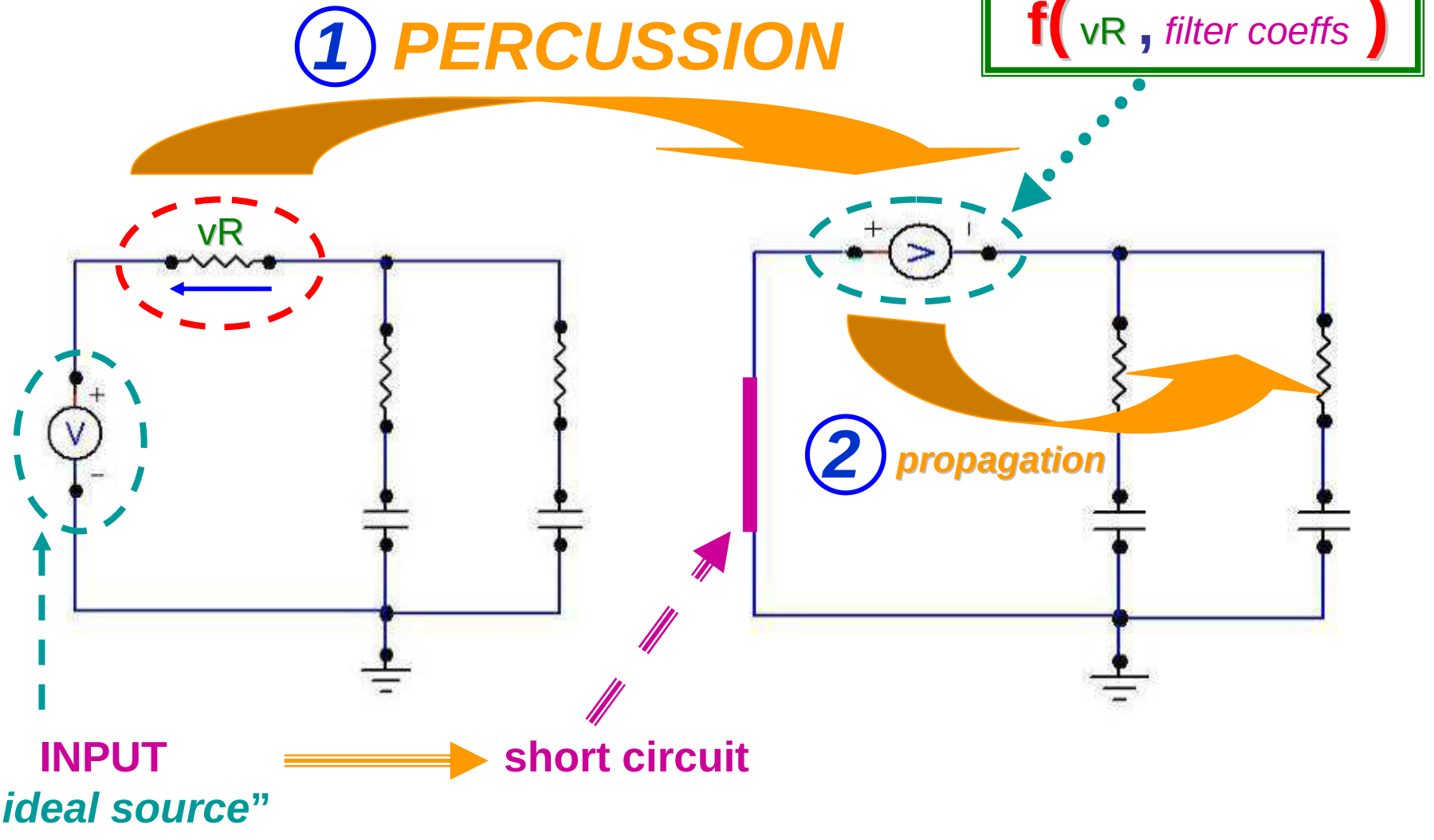


Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

2 stages mystery . . .

"ideal source"
but for a null time !!!

f(vR , *filter coeffs*)



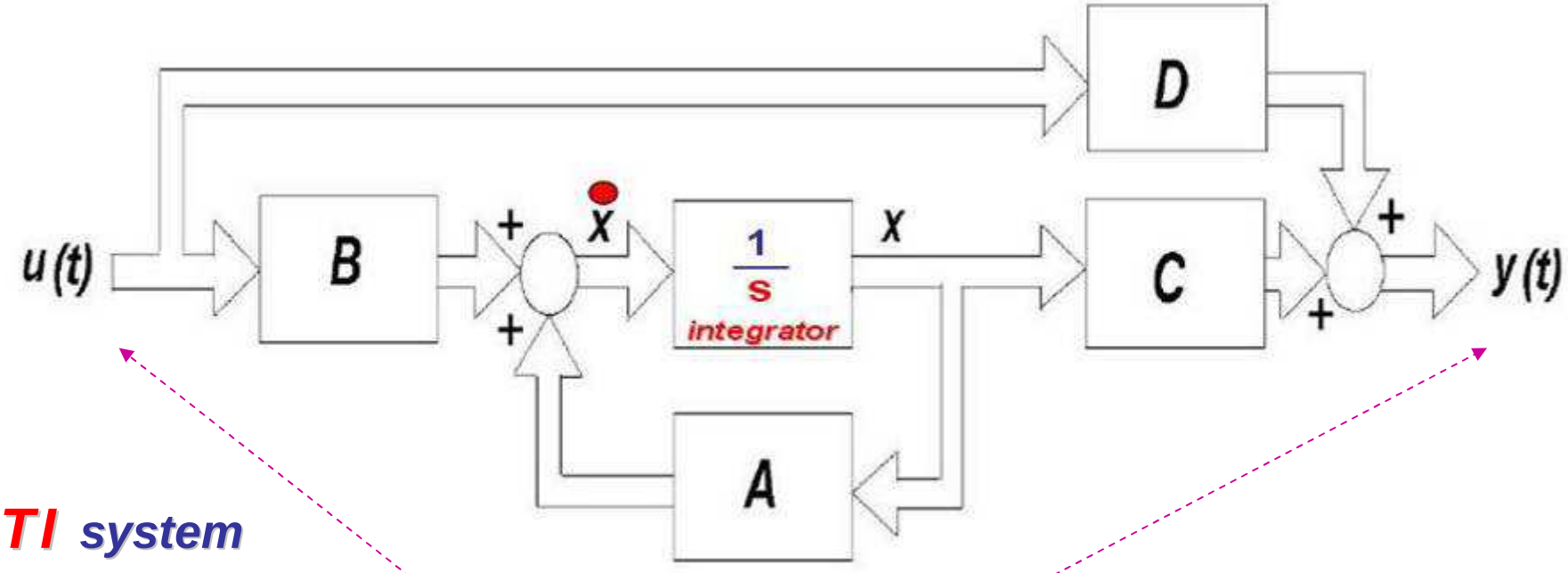
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Semi-analytical recursive algorithms for convolution calculations
W. Janke G. Blakiewicz
Department of Electronics, Technical University of Gdansk, Poland
IEE Proc. Circuits Devices Syst., Vol. 142 , No. 2, April I995

STATE MODEL

MATRICES & *VECTORS*

$$\dot{x}(t) = A\,x(t) + B\,u(t)$$
$$y(t) = C\,x(t) + D\,u(t)$$

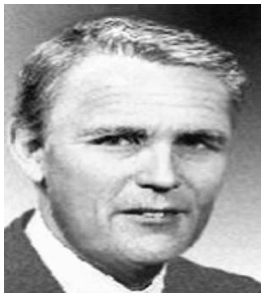


***LTI** system*

block diagram

*multiple **inputs** / multiple **outputs** (**MIMO**)*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Rudolf (Rudy) Emil Kálmán
born May 19, 1930
awarded in 2009 by
President Barack Obama with the
National Medal of Science

$$\dot{x}(t) = A\,x(t) + B\,u(t)$$
$$y(t) = C\,x(t) + D\,u(t)$$

In the 60's, based on Kálmán research, the State Model approach has been fully extended and used, for examples, on the Apollo and Polaris projects

For a given system, the ***transfer function*** output / input is unique but multiple set of **A, B, C, D** matrices can be expressed possibly corresponding to different **orders** (*and thus different number of* ***poles*** !!!)

But, the ***minimal order*** only corresponds to the **A matrices** that enforce both ***Controllable*** and ***Observable*** ***Kálmán criteria***

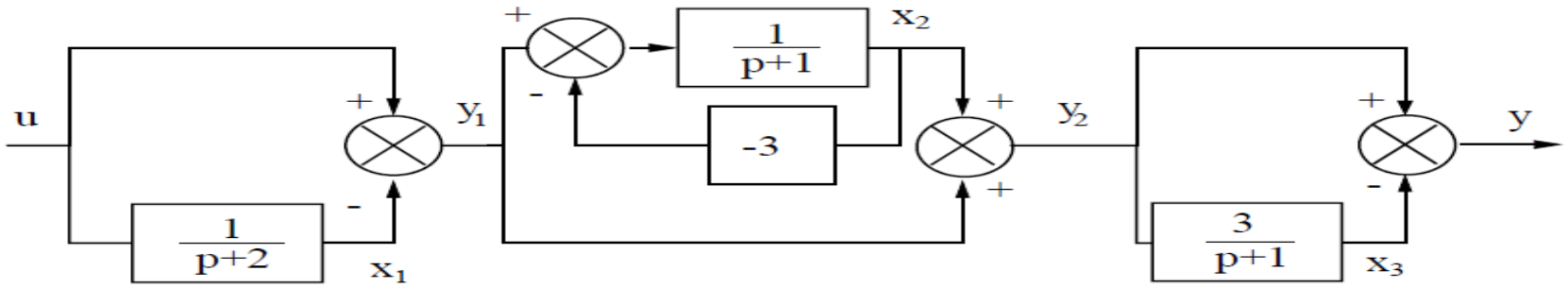
It also exists, for singular systems, a "***generalized state representation***"

$$E\,\dot{X} = A\,X + B\,U$$
$$Y = C\,X + D\,U$$

E is called the "***derivative***" matrix

for ***non-singular*** systems **E** = ***I****dentity*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



system "*external*" VIEW — the ***transfer function*** expression is **unique**

1 pole: -2

$$\frac{Y(p)}{U(p)} = \frac{p-1}{p+2}$$

pole-zero cancellations has occurred

The **output** / **input** ***differential equation***

2 poles: -2 , -1

$$\ddot{y} + 3\dot{y} + 2y = \ddot{u} - u$$

system "*internal*" VIEW — ***state equation***

matrix **A**

3 poles: -2 , -1 , 2

$$\dot{x} = \begin{bmatrix} -2 & 0 & 0 \\ -1 & 2 & 0 \\ -3 & 3 & -1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 1 \\ 3 \end{bmatrix} u$$

one of the possible "not minimal" state equation

$$\dot{x} = [-2]\,x + [1]\,u$$

1 pole: -2

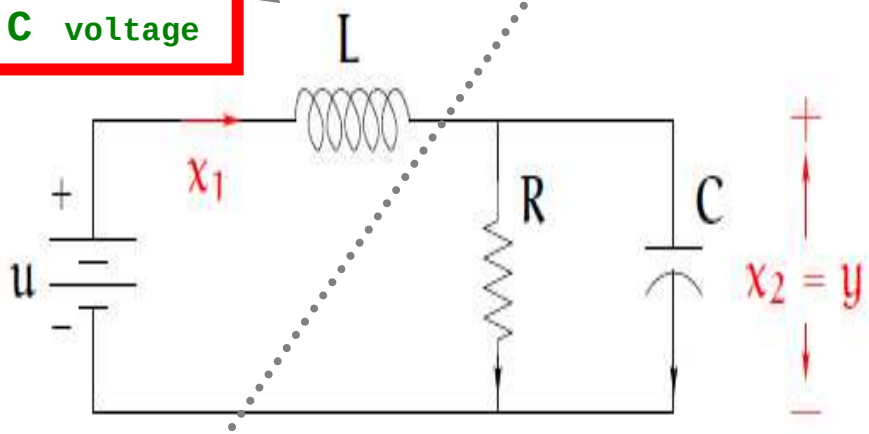
"minimal" state equation

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Equivalent STATE equations

STATEs can be:
L *current*, **C** *voltage*, LOOP *current*,
NODE *voltage*, **L** *flux*, **C** *charge*, **etc …**

L current
C voltage

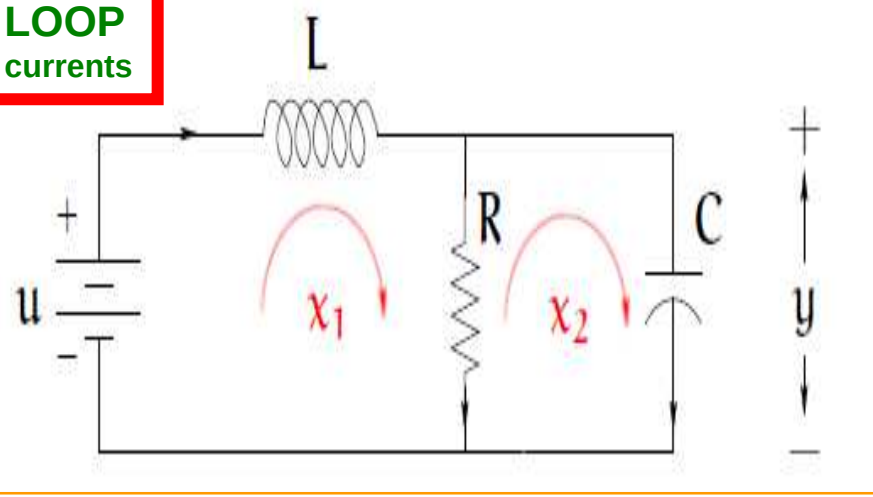


$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & -1 \\ 1 & -1 \end{bmatrix}}_{A} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{B} u$$

$$y = \underbrace{\begin{bmatrix} 0 & 1 \end{bmatrix}}_{C} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$R = 1\Omega \quad L = 1H \quad C = 1F$

LOOP currents



$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} -1 & 1 \\ -1 & 0 \end{bmatrix}}_{A} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}_{B} u$$

$$y = \underbrace{\begin{bmatrix} 1 & -1 \end{bmatrix}}_{C} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE
equivalence transformation

$$Z = T\,X \quad \Leftrightarrow \quad X = T^{-1}\,Z$$

Algebraic Equivalence

$$\dot{X} = A\,X + B\,U \qquad Y = C\,X + D\,U$$

$$\Rightarrow ?$$

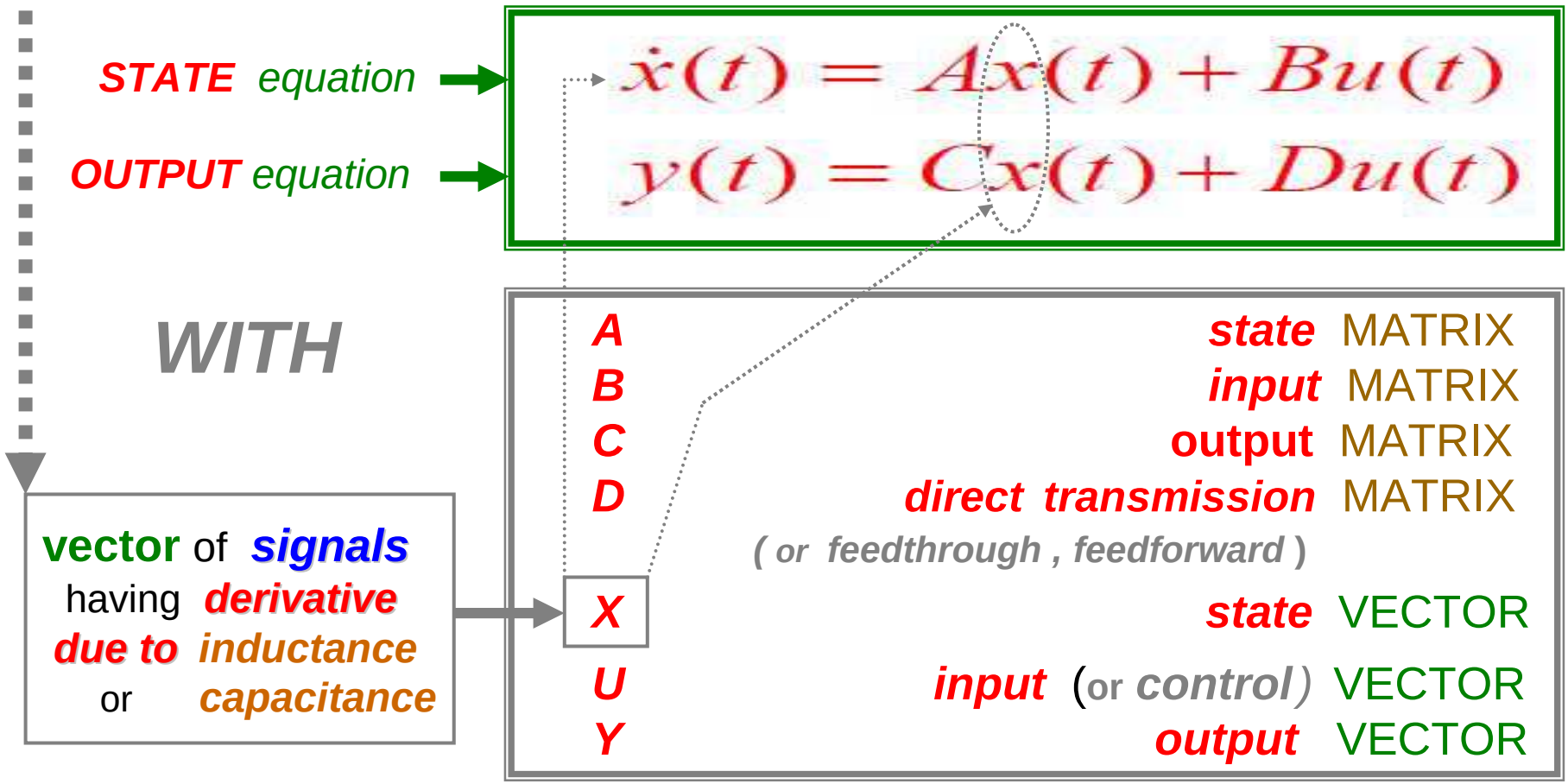
$$\dot{Z} = \tilde{A}\,Z + \tilde{B}\,U \qquad Y = \tilde{C}\,Z + \tilde{D}\,U$$

$$\tilde{A} = T\,A\,T^{-1} \qquad \tilde{B} = T\,B$$

$$\tilde{C} = C\,T^{-1} \qquad \tilde{D} = D$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE MODEL



This formalism* really highlights the *Linear Time Invariant* *terminology

It is mostly used by the "automatism" world ("control command")

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

$$\dot{x}(t) = A\,x(t) + B\,u(t)$$
$$y(t) = C\,x(t) + D\,u(t)$$

DIMENSIONS

	Variable	Dimension	Name
	s		Number of states
	i		Number of inputs
	o		Number of outputs
MATRICES	A	$s \times s$ matrix	State matrix
MATRICES	B	$s \times i$ matrix	Input matrix
MATRICES	C	$o \times s$ matrix	Output matrix
MATRICES	D	$o \times i$ matrix	Direct transmission matrix
VECTORS	$\mathbf{X}$	s vector	State vector
VECTORS	$\mathbf{U}$	i vector	Input vector
VECTORS	$\mathbf{Y}$	o vector	Output vector

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE equation

$$\dot{X} = A X + B U$$

not easy to compute (searching X involves a " ***convolution*** ")

order of matrix A is ***nearly*** related to the number of ***Inductances*** and ***Capacitances***

When ***order*** ≠ ***number energy-storing elements*** the network is called " ***degenerate network*** "

The **order** of complexity of any network equals the total number of ***energy-storing elements*** **minus** the ***number of all-capacitor*** " ***loops*** " **and** the ***number of all-inductor*** " ***cut-sets*** "

OUTPUT equation

$$Y = C X + D U$$

trivial to compute as soon as X is known

able to return any **voltage** or **current** node similarly to **Kirchhoff / Ohm**

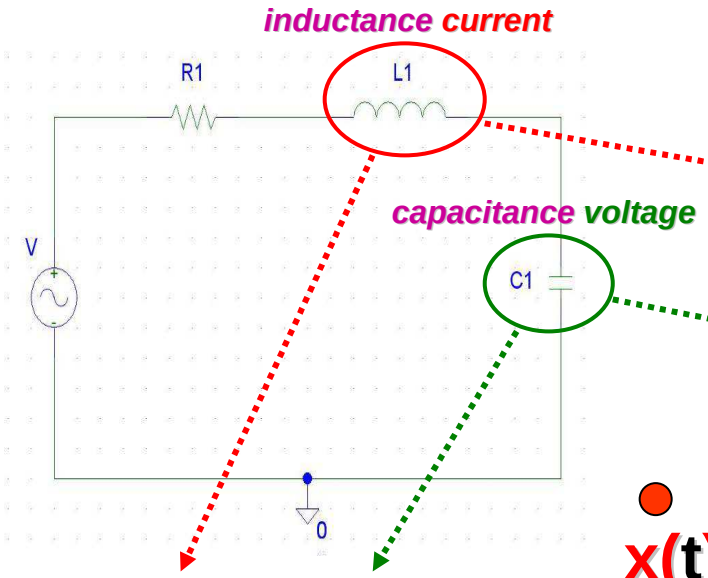
order of **vector** Y depends on the number of ***expected observed values***

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE MODEL

$$\dot{x}(t) = A\,x(t) + B\,u(t)$$

$$y(t) = C\,x(t) + D\,u(t)$$



$$L1\ \frac{d\,iL1(t)}{d\,t} = -R1\,iL1(t) - vC1(t) + u(t)$$

$$C1\ \frac{d\,vC1(t)}{d\,t} = iL1(t)$$

2 *impedance components*

state dimension 2

$$\dot{x}(t) = A\,x(t) + B\,u(t)$$

$$\begin{pmatrix} \frac{d\,iL1(t)}{d\,t} \\ \frac{d\,vC1(t)}{d\,t} \end{pmatrix} = \begin{pmatrix} -\frac{R1}{L1} & -\frac{1}{L1} \\ \frac{1}{C1} & 0 \end{pmatrix} \begin{pmatrix} iL1(t) \\ vC1(t) \end{pmatrix} + \begin{pmatrix} \frac{1}{L1} \\ 0 \end{pmatrix} u(t)$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Now, let's suppose that

for the **y(t)** *output vector*

we want to observe ***vR1(t)***

from the schematic →

$$vR1(t) = R1 \; iL1(t)$$

→

$$vR1(t) = \begin{pmatrix} R1 \\ 0 \end{pmatrix} \begin{pmatrix} iL1(t) \\ vC1(t) \end{pmatrix} + \begin{pmatrix} 0 \end{pmatrix} u(t)$$

$$y(t) \quad C \quad x(t) \quad D \quad u(t)$$

The resulting **4** matrices **A B C D** are ***intrinsically characteristic*** of our ***circuit***

$$A = \begin{pmatrix} -\frac{R1}{L1} & -\frac{1}{L1} \\ \frac{1}{C1} & 0 \end{pmatrix} \quad B = \begin{pmatrix} \frac{1}{L1} \\ 0 \end{pmatrix} \quad C = \begin{pmatrix} R1 \\ 0 \end{pmatrix} \quad D = \begin{pmatrix} 0 \end{pmatrix}$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

the ***STATE MODEL***

is giving us the opportunity to

answer to the ***frightening question*** *:*

What is the rational behind

transfer function ***denominator*** ***???***

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

We claimed that the 4 matrices A B C D are characteristic of the circuit

does that means that they are independent of the "processing domain" ???

***In other words**, are they meaningful as well, for example, to the Laplace "s" domain*

$$X(t) \rightarrow X(s)$$

$$\dot{X}(t) \rightarrow s\,X(s)$$

derivate

$$\dot{x}(t) = A\,x(t) + B\,u(t)$$

$$y(t) = C\,x(t) + D\,u(t)$$

$$s\,X(s) = A\,X(s) + B\,u(s)$$

$$Y(s) = C\,X(s) + D\,u(s)$$

$$\frac{Y(s)}{u(s)} = H(s) = C\,(s\,I - A)^{-1}\,B + D$$

$$H(s) = \frac{C\,[\,cof(s\,I - A)\,]^{T}\,B + D\,Q(s)}{Q(s)}$$

with $Q(s) = \det(s\,I - A)$ *Characteristic Polynomial of matrix A*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

This implies that the **zeros** *of this* **Characteristic Polynomial** *of matrix* **A**

$$Q(s) = \det(sI - A)$$ *are* **also** *the* **POLEs** *of* $H(s)$

and so . . . the **POLEs** *of* $H(s)$ *are from the expression of* $Q(s)$

the **Eigen values** *of* **matrix** **A** ("*valeurs propres*" *in french*)

$$A = \begin{pmatrix} -\frac{R1}{L1} & -\frac{1}{L1} \\ \frac{1}{C1} & 0 \end{pmatrix}$$

Eigen values

$$-\frac{R1}{2\,L1} - \frac{\sqrt{C1^2\,R1^2 - 4\,C1\,L1}}{2\,C1\,L1}$$

$$-\frac{R1}{2\,L1} + \frac{\sqrt{C1^2\,R1^2 - 4\,C1\,L1}}{2\,C1\,L1}$$

POLEs of the **Transfer Function** $H(s)$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

$$A = \begin{pmatrix} -\frac{R1}{L1} & -\frac{1}{L1} \\ \frac{1}{C1} & 0 \end{pmatrix}$$

Characteristic Polynomial *of matrix* ***A***

$$Q(s) = \det(sI - A)$$

$$Q(s) = s^2 + \frac{R1\,s}{L1} + \frac{1}{C1\,L1}$$

please note that such ***theoretical denominator*** *expression*

is ***literally*** $(s - p1) * (s - p2)$

and thus has an **unit** s^2 coefficient

$$= \frac{C1\,L1\,s^2 + C1\,R1\,s + 1}{C1\,L1}$$

If you remember, **this is effectively** ***coherent*** **with what we have found for the** ***"denominators"*** **when we used the** ***Kirchhoff laws*** **to compute** ***Laplace Transfer Functions***

$$\frac{vr1}{V} = \frac{C1\,R1\,s}{C1\,L1\,s^2 + C1\,R1\,s + 1}$$

$$\frac{il1}{V} = \frac{C1\,s}{C1\,L1\,s^2 + C1\,R1\,s + 1}$$

$$\frac{vc1}{V} = \frac{1}{C1\,L1\,s^2 + C1\,R1\,s + 1}$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

So, what have we learned ???

The ***polynomial*** that, *for a given circuit*,

sounds "***strangely***" reappearing on the ***denominator*** of its various ***output Transfer Functions*** is getting ***a justification*** *based on the*

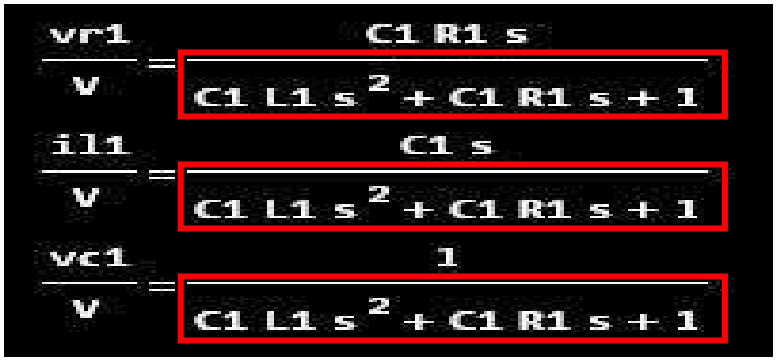
> ***Characteristic Polynomial*** *of the* ***STATE*** *matrix* ***A***

which is ***independent*** *of the* ***observed OUTPUT***

vr1

il1

vc1



So The expression "***Characteristic Polynomial***" is therefore taking its full sense since it truly ***characterizes*** the ***INTRINSIC STATE*** of the circuit

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

For the fun, let's verify that the State Model $vR1(s)$ *Transfer Function*

is effectively the same than the one found with Kirchhoff

$$A = \begin{pmatrix} -\frac{R1}{L1} & -\frac{1}{L1} \\ \frac{1}{C1} & 0 \end{pmatrix} \quad B = \begin{pmatrix} \frac{1}{L1} \\ 0 \end{pmatrix} \quad C = \begin{pmatrix} R1 & \\ 0 & \end{pmatrix} \quad D = \begin{pmatrix} 0 \end{pmatrix}$$

$$\frac{vR1(s)}{u(s)} = \frac{C\,[\,cof(s\,I - A)\,]^{T}\,B + D\,Q(s)}{Q(s)}$$

$$Q(s) = s^2 + \frac{R1}{L1}s + \frac{1}{C1\,L1}$$

$$[\,cof(s\,I - A)\,]^{T} = \begin{bmatrix} s & -\frac{1}{L1} \\ \frac{1}{C1} & \frac{L1\,s + R1}{L1} \end{bmatrix}$$

transpose of the MATRIX of *cofactors*

$$\Rightarrow \quad \frac{vR1(s)}{u(s)} = \frac{\frac{R1\,s}{L1}}{\frac{C1\,L1\,s^2 + C1\,R1\,s + 1}{C1\,L1}} = \frac{C1\,R1\,s}{C1\,L1\,s^2 + C1\,R1\,s + 1}$$

QED !!!

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE MODEL

?

$$expression(t) \Rightarrow x(t)$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

$$\dot{x} = ax + bu \Rightarrow \dot{x}(t) - ax(t) = bu(t)$$

$$e^{-at}(\dot{x}(t) - ax(t)) = e^{-at}(bu(t))$$

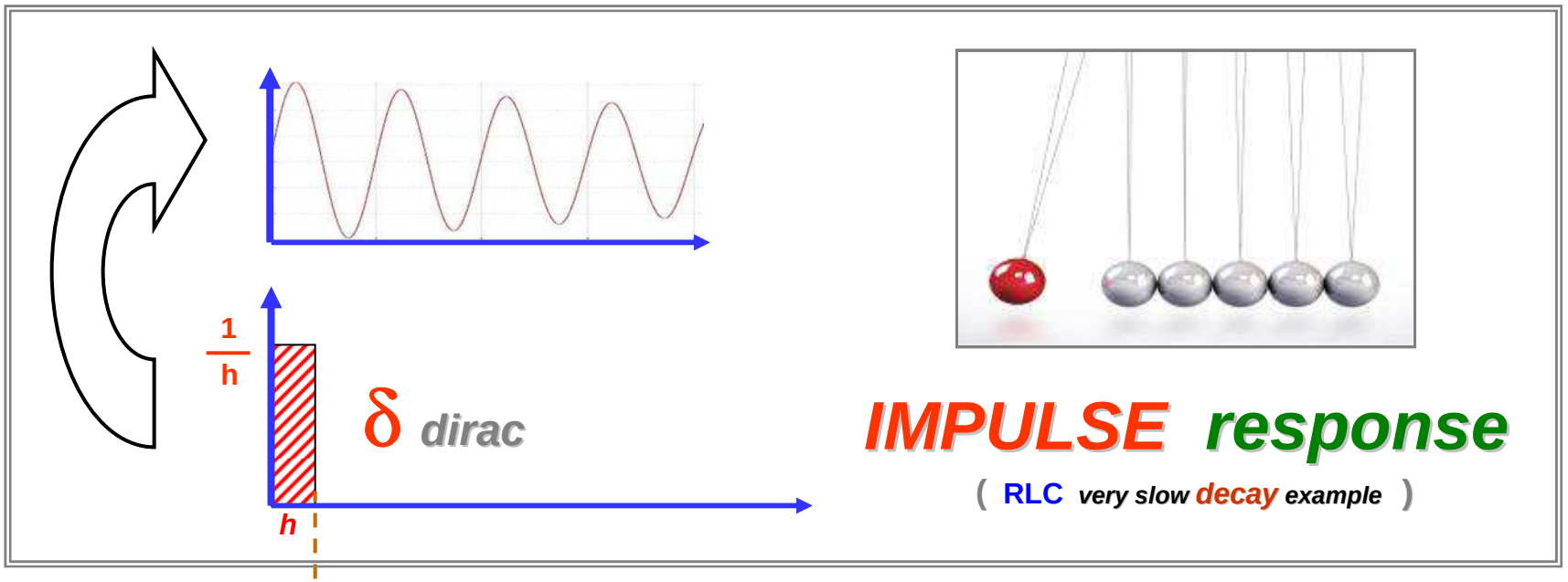
$$e^{-at}\dot{x} - e^{-at}ax = \frac{d}{dt}\left(e^{-at}x(t)\right) \Rightarrow \frac{d}{dt}\left(e^{-at}x(t)\right) = e^{-at}bu$$

$$\int_0^t \frac{d}{d\tau}\left(e^{-a\tau}x(\tau)\right)d\tau = \int_0^t e^{-a\tau}bu(\tau)\,d\tau \Rightarrow e^{-at}x(t) - x(0) = \int_0^t e^{-a\tau}bu(\tau)\,d\tau$$

$$x(t) = e^{at}x(0) + \int_0^t e^{a(t-\tau)}bu(\tau)\,d\tau$$

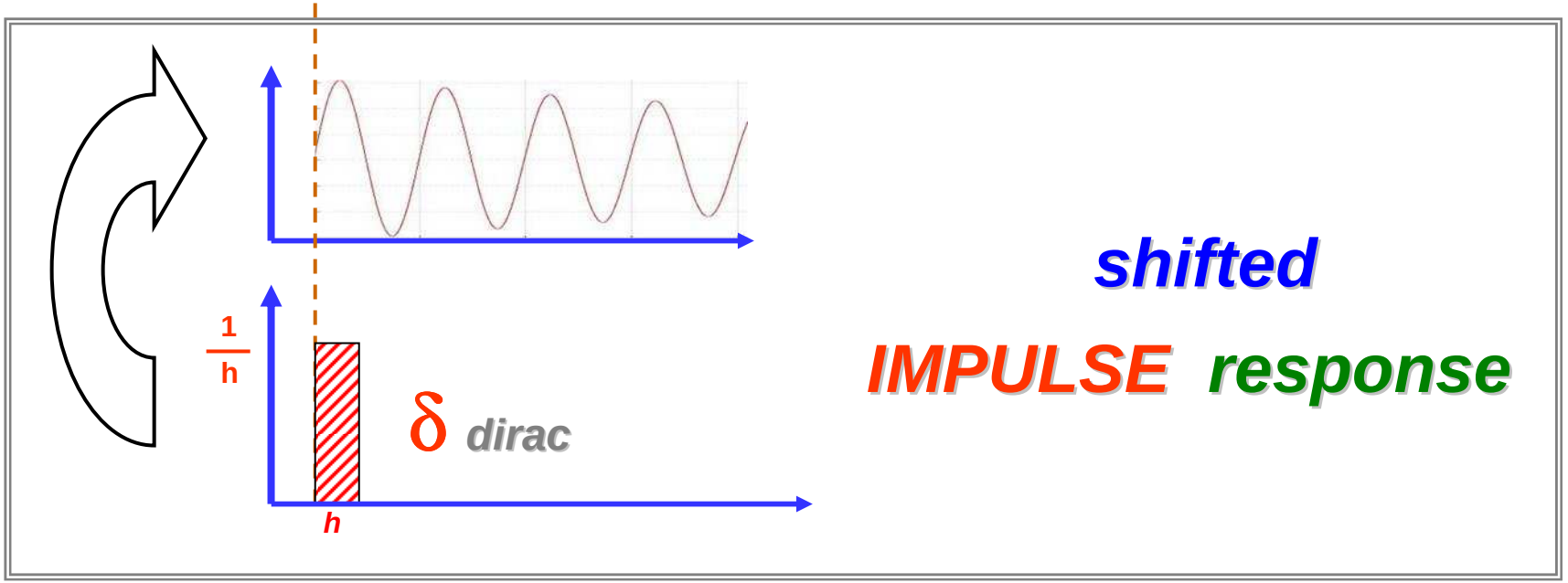
convolution **integral**

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



IMPULSE response

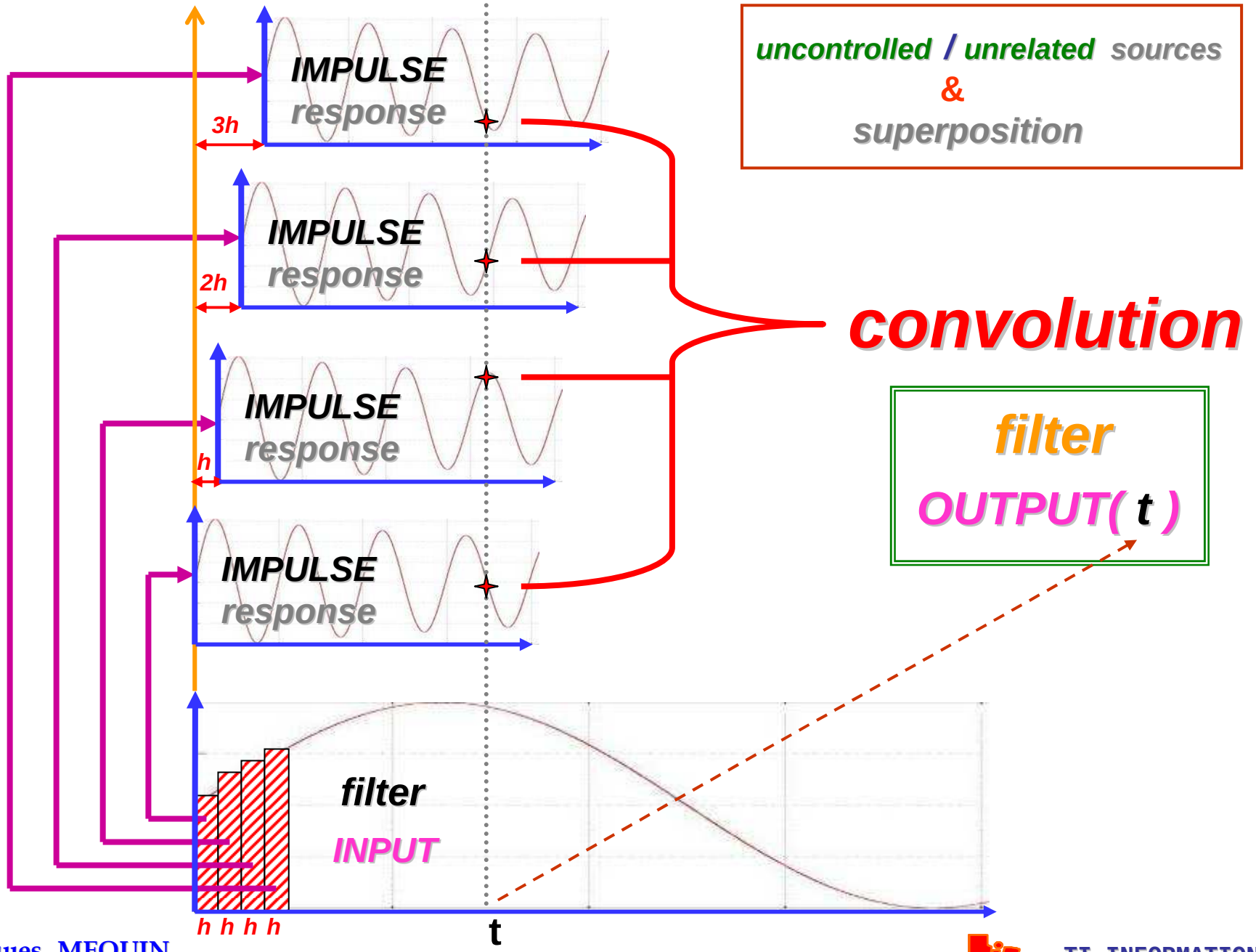
(RLC *very slow decay example*)



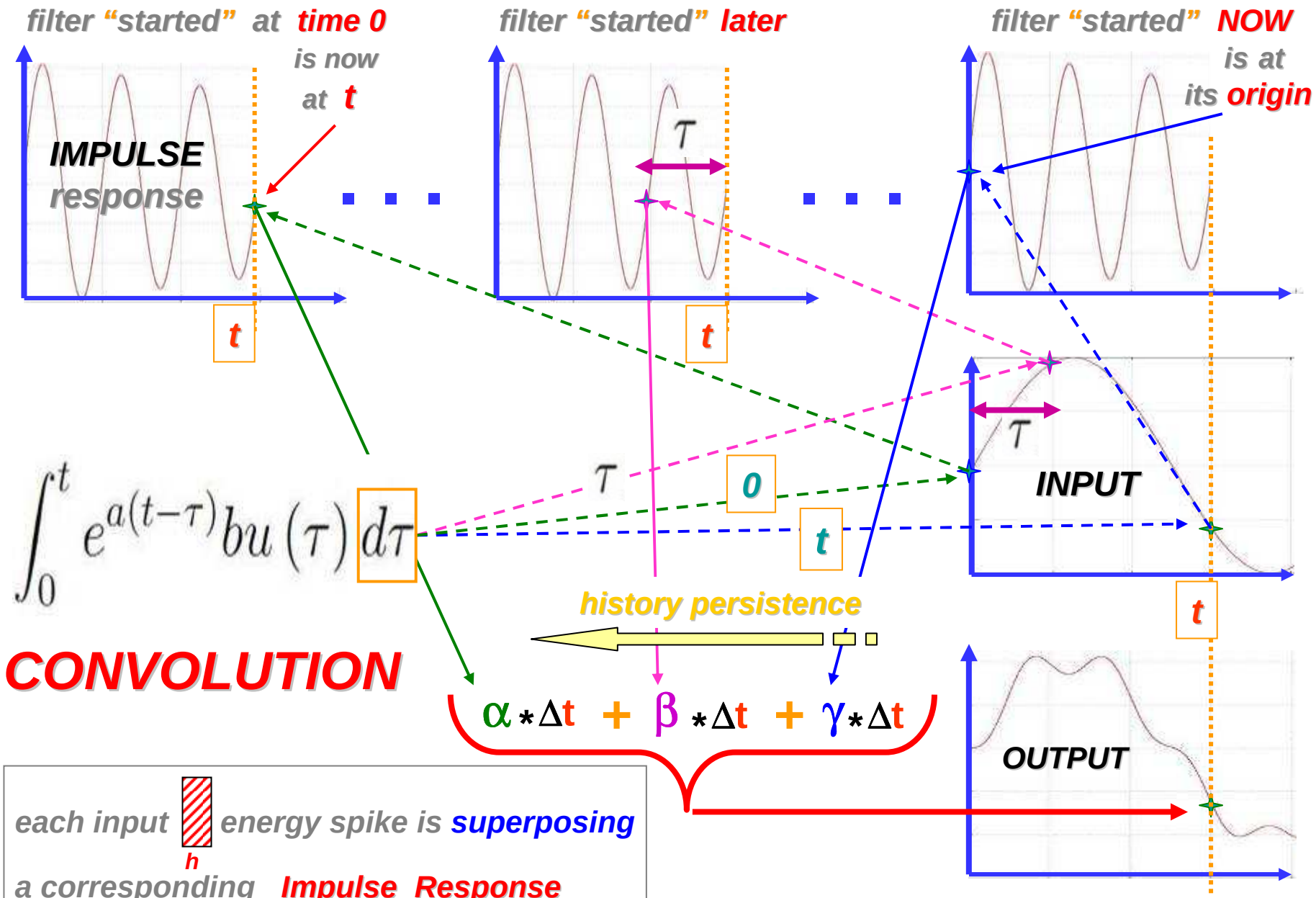
shifted

IMPULSE response

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



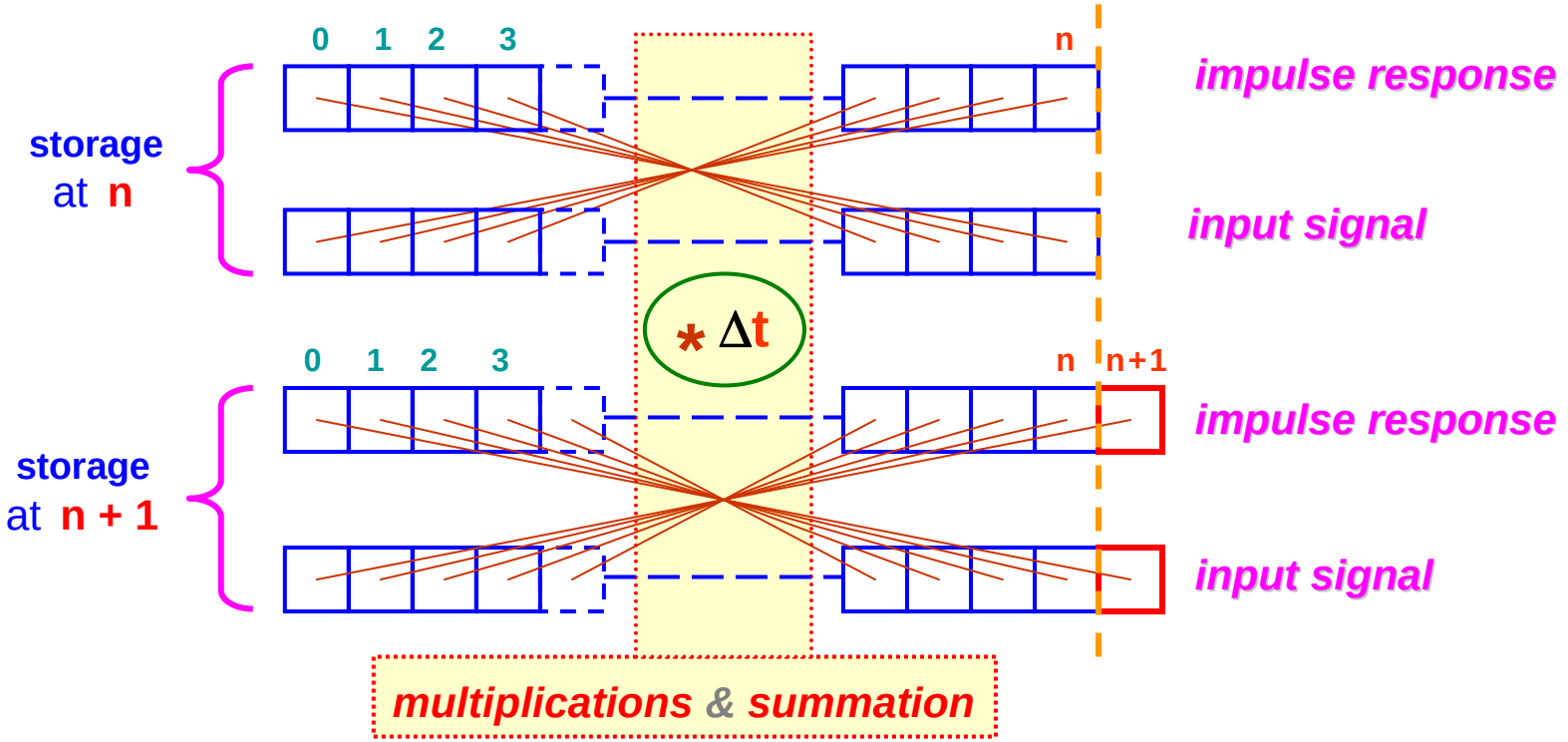
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Unfortunately,

brut force convolution numerical usage is unlikely

unknown input signal function u(t) prevents direct integration

sample based discrete numerical evaluation is very resource consuming



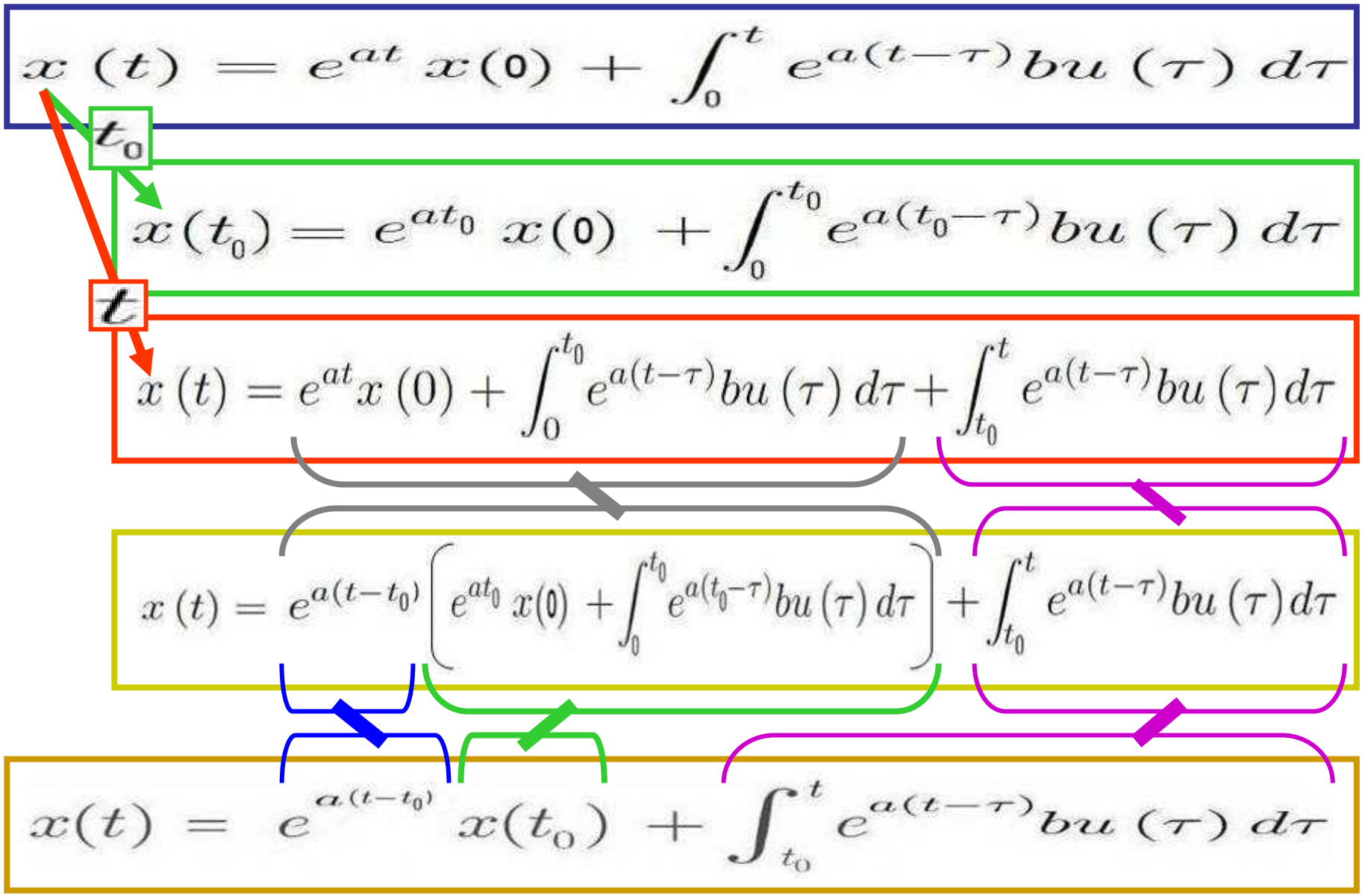
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE MODEL

☑ $expression(t) \Rightarrow x(t)$

? $x(t) \Rightarrow x(t+h)$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Recursive Convolution Formula (RCF)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE MODEL "time domain" evolution

STATE matrix **A**

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau$$

new STATE at t	u(t) = 0 "free" mode since STATE t_0	effective contribution from u(t) "forced" mode since STATE t_0

$$e^x = \sum_{k=0}^{\infty} \frac{1}{k!}x^k = 1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3 + \cdots$$

$$e^{At} = \sum_{k=0}^{\infty} \frac{t^k A^k}{k!} = I + At + \frac{t^2A^2}{2!} + \cdots + \frac{t^kA^k}{k!} + \cdots$$

The *Transition Matrix* e^{At} can be calculated

by searching *individual matrix coefficient* ***Inverse Laplace Transform***

OR ***globally*** through ***analytic or approximate matricial methods***

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE MODEL

- ✓ $expression(t) \Rightarrow x(t)$
- ✓ $x(t) \Rightarrow x(t+h)$
- ? $expression(t) \Rightarrow e^{At}$

example: calculation based on **Inverse Laplace Transform** method

$$e^{At} = InverseLaplaceTransform\left\{K^{-1}\right\} \quad \text{with} \quad K = sI - A$$

Characteristic Matrix of matrix A

$$A = \begin{bmatrix} -\frac{R1}{L1} & -\frac{1}{L1} \\ \frac{1}{C1} & 0 \end{bmatrix} \Rightarrow K = \begin{bmatrix} \frac{L1\,s+R1}{L1} & \frac{1}{L1} \\ -\frac{1}{C1} & s \end{bmatrix} \Rightarrow K^{-1} = \begin{bmatrix} \frac{C1L1\,s}{C1L1\,s^2+C1R1\,s+1} & -\frac{C1}{C1L1\,s^2+C1R1\,s+1} \\ \frac{L1}{C1L1\,s^2+C1R1\,s+1} & \frac{C1L1\,s+C1R1}{C1L1\,s^2+C1R1\,s+1} \end{bmatrix}$$

$$e^{At} = \begin{bmatrix} ILT\left(\frac{s\,C1L1}{s\,C1R1+s^2\,C1L1+1}\right) & ILT\left(-\frac{C1}{s\,C1R1+s^2\,C1L1+1}\right) \\ ILT\left(\frac{L1}{s\,C1R1+s^2\,C1L1+1}\right) & ILT\left(\frac{C1R1+s\,C1L1}{s\,C1R1+s^2\,C1L1+1}\right) \end{bmatrix} = \begin{bmatrix} ILT\left(\frac{s}{(s-p1)(s-p2)}\right) & ILT\left(-\frac{1}{(s-p1)(s-p2)L1}\right) \\ ILT\left(\frac{1}{(s-p1)(s-p2)C1}\right) & ILT\left(\frac{R1+s\,L1}{(s-p1)(s-p2)L1}\right) \end{bmatrix}$$

transformed matrix *is a symbolic "time domain"* expression of e^{At}

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

MATRIX decomposition

$$A = M \Phi M^{-1}$$

$$f(A) = f(M \Phi M^{-1})$$

$$f(A) = M f(\Phi) M^{-1}$$

similarity matrix Φ

diagonalization

$$\begin{bmatrix} \lambda_1 & & 0 \\ & \lambda_2 & \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$$

diagonal

A may not be **diagonalizable**
(**defective matrix**)

JORDAN

$$\begin{pmatrix} \lambda_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ & \lambda_1 & 0 & 0 & 0 & 0 & 0 \\ & & \lambda_1 & 1 & 0 & 0 & 0 \\ & & & \lambda_1 & 0 & 0 & 0 \\ & 0 & & & \lambda_2 & 1 & 0 \\ & & & & & \lambda_2 & 0 \\ & & & & & & \lambda_3 \end{pmatrix}$$

bi-diagonal

triangularization

Schur form

$$\begin{bmatrix} \lambda_1 & t_{12} & t_{13} \\ 0 & \lambda_2 & t_{23} \\ 0 & 0 & \lambda_3 \end{bmatrix}$$

upper-triangular

strictly upper-triangular (**nilpotent** property)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

example: e^{At} calculation based on *Jordan*

(even if this **A** *is directly diagonalizable)*

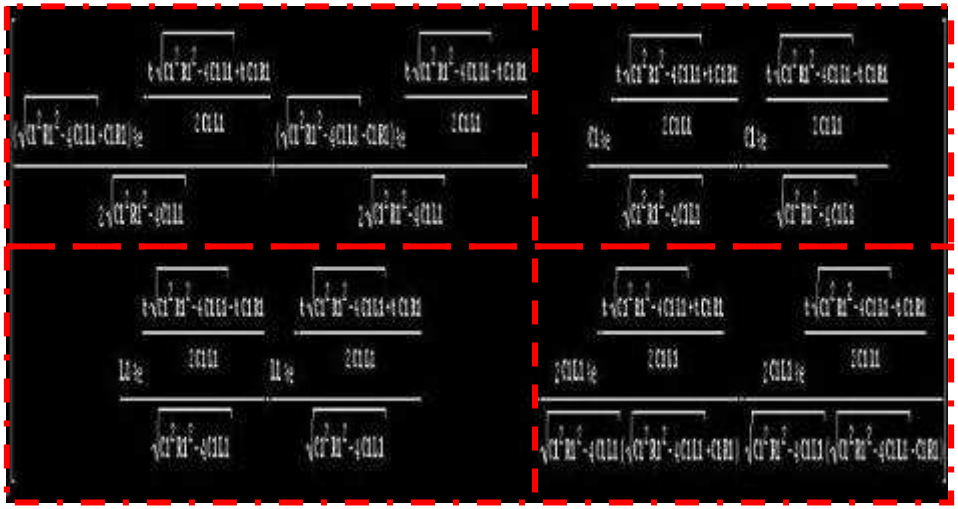
$$A = \begin{bmatrix} -\frac{R1}{L1} & -\frac{1}{L1} \\ \frac{1}{C1} & 0 \end{bmatrix} \Rightarrow \text{Jordan} = \begin{bmatrix} \frac{\sqrt{C1^2 R1^2 - 4 C1 L1} + C1 R1}{2 C1 L1} & 0 \\ 0 & \frac{\sqrt{C1^2 R1^2 - 4 C1 L1} - C1 R1}{2 C1 L1} \end{bmatrix} = \begin{bmatrix} p1 & 0 \\ 0 & p2 \end{bmatrix}$$

$$A = MJM^{-1} \quad \textit{similarity transform}$$

$$A^k = (MJM^{-1})^k = MJ^kM^{-1}$$

$$M = \begin{bmatrix} 1 & 1 \\ \frac{1}{p1\ C1} & \frac{1}{p2\ C1} \end{bmatrix}$$

$$e^{At} = \sum_{k=0}^{\infty} \frac{t^k A^k}{k!} = Me^{Jt}M^{-1} =$$



transition matrix

that could be rewritten

based on the poles *p1* & *p2*

(that are conjugates)

$$\begin{bmatrix} \frac{p2\ e^{p2\,t} - p1\ e^{p1\,t}}{p2 - p1} & \frac{e^{p2\,t} - e^{p1\,t}}{(p2 - p1)L1} \\ \frac{e^{p2\,t} - e^{p1\,t}}{(p2 - p1)C1} & \frac{p1\ e^{p2\,t} - p2\ e^{p1\,t}}{p2 - p1} \end{bmatrix}$$

even with complex poles, ***A*** *real* ➔ *transition matrix real*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Inverse Laplace Transform or Jordan

p1 & p2 conjugates

Either way …

$$e^{At} = \begin{bmatrix} \frac{p2\,e^{p2\,t} - p1\,e^{p1\,t}}{p2-p1} & \frac{e^{p2\,t} - e^{p1\,t}}{(p2-p1)L1} \\ \frac{e^{p2\,t} - e^{p1\,t}}{(p2-p1)C1} & \frac{p1\,e^{p2\,t} - p2\,e^{p1\,t}}{p2-p1} \end{bmatrix}$$

poles "distance"

let $p = r \pm v = r \pm z\,w$

r real
z 1 or i
w real > 0

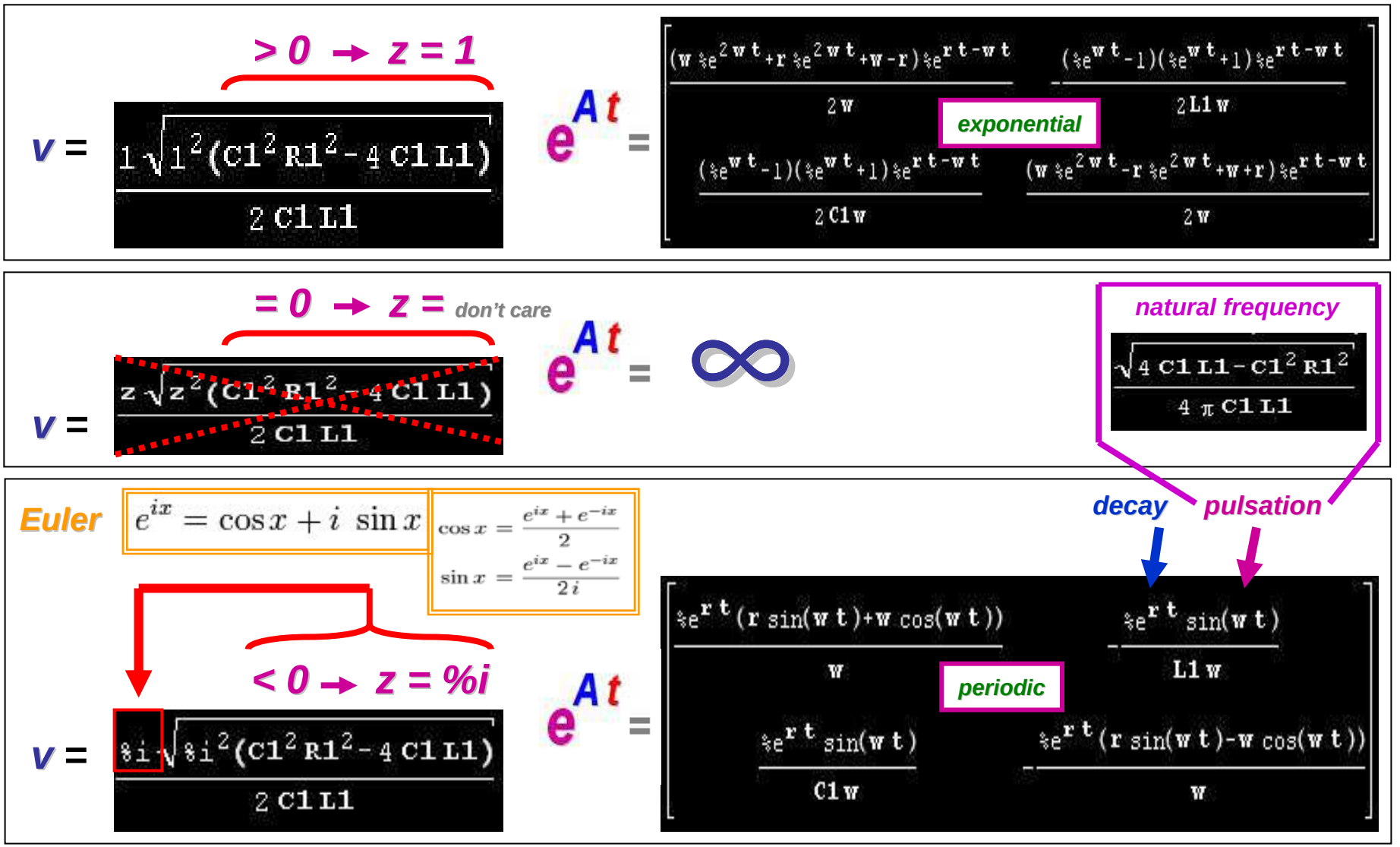
r must be negative for stability

$$r = \frac{p1 + p2}{2} = -\frac{R1}{2\,L1}$$

$$v = z\,w = z\left|f\left(z, \frac{p1 - p2}{2}\right)\right| = \frac{z\sqrt{z^2(C1^2\,R1^2 - 4\,C1\,L1)}}{2\,C1\,L1}$$

$$e^{At} = \begin{bmatrix} \frac{(v\,e^{2\,v\,t} + r\,e^{2\,v\,t} + v - r)\,e^{r\,t - v\,t}}{2\,v} & -\frac{(e^{v\,t} - 1)(e^{v\,t} + 1)\,e^{r\,t - v\,t}}{2\,L1\,v} \\ \frac{(e^{v\,t} - 1)(e^{v\,t} + 1)\,e^{r\,t - v\,t}}{2\,C1\,v} & \frac{(v\,e^{2\,v\,t} - r\,e^{2\,v\,t} + v + r)\,e^{r\,t - v\,t}}{2\,v} \end{bmatrix}$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Those expressions are clearly highlighting the ***intrinsic influence*** of the ***Transition Matrix***

on the overall ***state*** behavior ***regardless*** of the ***system input activation***

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later

Cleve Moler , Charles Van Loan

SIAM review 1978 and then 2003

	1	*Taylor series*
✓	2	*Padé approximation*
	3	*Scaling and squaring*
	4	*Chebyshev rational approximation*
	5	*General purpose ODE solver*
	6	*Single step ODE method*
	7	*Multistep ODE solver*
	8	*Cayley - Hamilton*
	9	*Lagrange interpolation*
	10	*Newton interpolation*
	11	*Vandermonde matrix*
academic	12	*Inverse Laplace transform*
	13	*Companion matrix*
	14	*Eigenvectors*
	15	*Triangular systems of eigenvectors*
academic	16	*Jordan canonical form*
not e^x *specialized*	17	*Schur decomposition*
	18	*Block diagonal matrix*
	19	*Splitting method*
	20	*Krylov space methods* (*added in 2003*)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Brook Taylor (1685 – 1731)

"Tailor *series*" order 3

$$e^{x} \approx 1 + x + \frac{x^2}{2} + \frac{x^3}{6}$$

series development

approximation

Henri Padé (1863 – 1953)

"Padé *approximants*" order 3

$$e^{z} \approx \frac{1 + \frac{1}{2}z + \frac{1}{10}z^2 + \frac{1}{120}z^3}{1 - \frac{1}{2}z + \frac{1}{10}z^2 - \frac{1}{120}z^3}$$

"*rational fraction*"

$$e^z = \cfrac{1}{1 - \cfrac{z}{1 + \cfrac{\frac{1}{2}z}{1 - \cfrac{\frac{1}{6}z}{1 + \cfrac{\frac{1}{6}z}{1 - \cfrac{\frac{1}{10}z}{1 + \cfrac{\frac{1}{10}z}{1 - + \ddots}}}}}}}$$

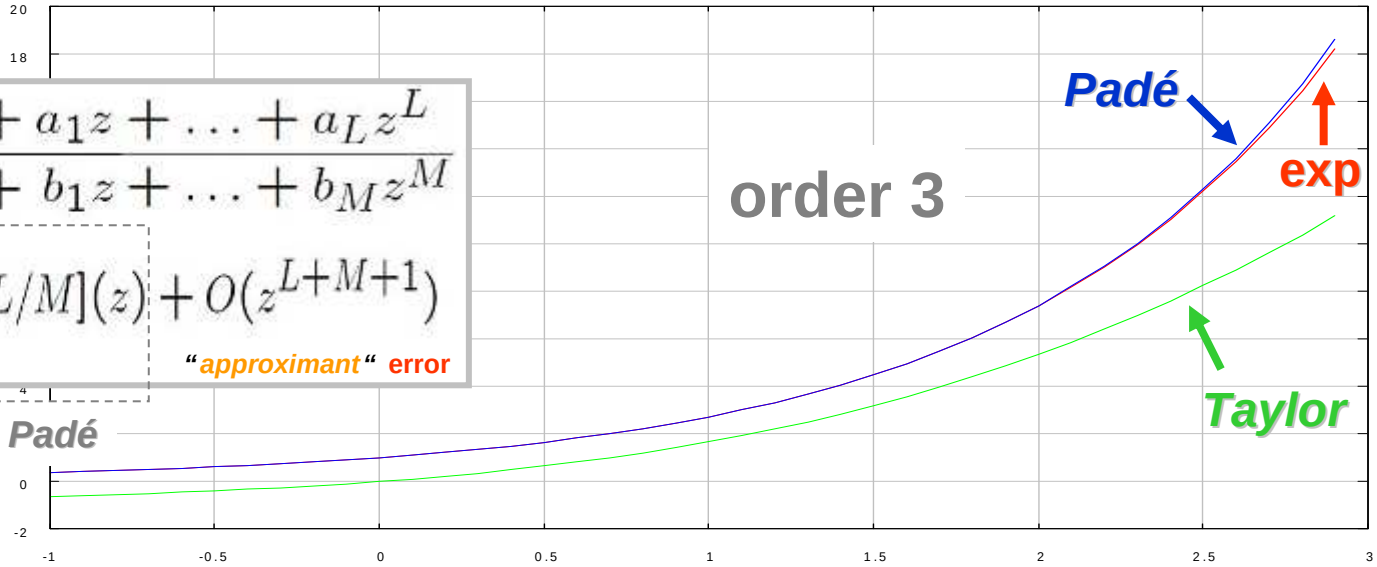
"*continued fraction*" *form*

$$[L/M](z) = \frac{a_0 + a_1 z + \ldots + a_L z^L}{b_0 + b_1 z + \ldots + b_M z^M}$$

$$f(z) = \sum_{i=0}^{\infty} c_i z^i = [L/M](z) + O(z^{L+M+1})$$

"*approximant*" error

Tailor *Padé*



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE MODEL

- ☑ $expression(t) \Rightarrow x(t)$
- ☑ $x(t) \Rightarrow x(t+h)$
- ☑ $expression(t) \Rightarrow e^{At}$
- ? unknown input $u(\tau)$ ⇒ approximation

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Semi Analytical Recursive Convolution algorithm (SARC)

$$\int_{t\text{-}h}^{t} e^{a(t-\tau)}\, u(\tau)\, d\tau$$

$$e^{at} \int_{t\text{-}h}^{t} e^{-a\tau}\, u(\tau)\, d\tau$$

$$e^{at} \int_{t\text{-}h}^{t} e^{-a\tau}\, \mathit{interpolation}(\tau)\, d\tau$$

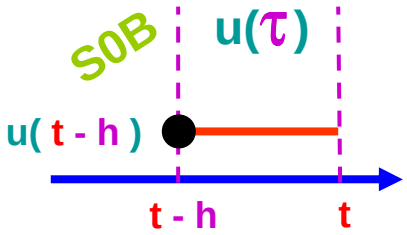
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

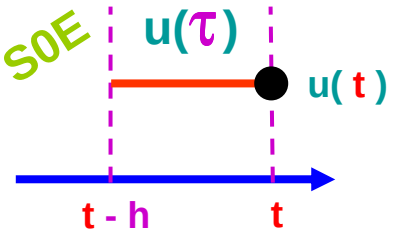
Lagrange polynomial

$$u(\tau) = \sum_{i=0}^{n} L_i(\tau)\, u_i \qquad \text{with} \qquad L_i(\tau) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{(\tau - \tau_j)}{(\tau_i - \tau_j)}$$

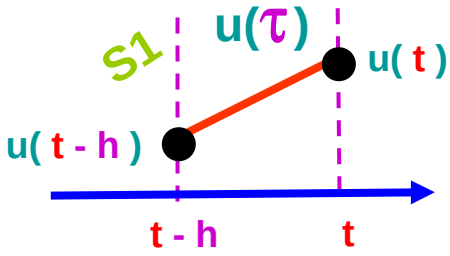
$$L_i(\tau_i) = 1$$
$$L_i(\tau_j) = 0 \ (j \neq i)$$

order n = *0* — u(τ) is approximated on the *interval* [t - h , t] by a **constant**





order n = *1* — u(τ) is approximated on the *interval* [t - h , t] by a **slope**



etc ...

interpolation **error**

$O(h^{n+1})$ — $\frac{h}{2}$ → error $\frac{}{2^{n+1}}$

$$e(\tau) = \frac{u^{(n+1)}(\xi)}{(n+1)!}\,\lambda(\tau) \qquad \xi \in [\tau_0 , \tau_n]$$

interpolation points

with $\lambda(\tau) = (\tau-\tau_0)(\tau-\tau_1)\ldots(\tau-\tau_n)$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

$$\int_{t-h}^{t} e^{a(t-\tau)}\, u(\tau)\, d\tau \;\Rightarrow\; e^{at}\int_{t-h}^{t} e^{-a\tau}\, \text{interpolation}(\tau)\, d\tau$$

order 0

$$e^{at}\int_{t-h}^{t} e^{-a\tau}\left(u_t\right) d\tau$$

$$\Phi = e^{ah} \quad \text{constant}$$

$$\frac{\Phi - 1}{a}\, u_t$$

order 1

$$e^{at}\int_{t-h}^{t} e^{-a\tau}\left(\frac{t-\tau}{h}\, u_{t-h} + \frac{\tau-(t-h)}{h}\, u_t\right) d\tau$$

$$\frac{1}{a}\left(\Phi - \frac{\Phi - 1}{ah}\right) u_{t-h} + \frac{1}{a}\left(\frac{\Phi - 1}{ah} - 1\right) u_t$$

does not depend on e^{at} **!!!** *(Markov process)*

The "shaking" due to [t-h , t] input is independent of the current oldness of the system

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

putting everything together

$$x(t) = e^{at}\, x(0) + \int_0^t e^{a(t-\tau)} bu(\tau)\, d\tau$$

$$x(t) = e^{a(t-t_0)}\, x(t_0) + \int_{t_0}^t e^{a(t-\tau)} bu(\tau)\, d\tau$$

$$x(t) = e^{ah}\, x(t-h) + \int_{t-h}^t e^{a(t-\tau)} bu(\tau)\, d\tau$$

$$x(t) = e^{ah}\, x(t-h) + b\, e^{at} \int_{t-h}^t e^{-a\tau}\, \text{interpolation}(\tau)\, d\tau$$

order 0

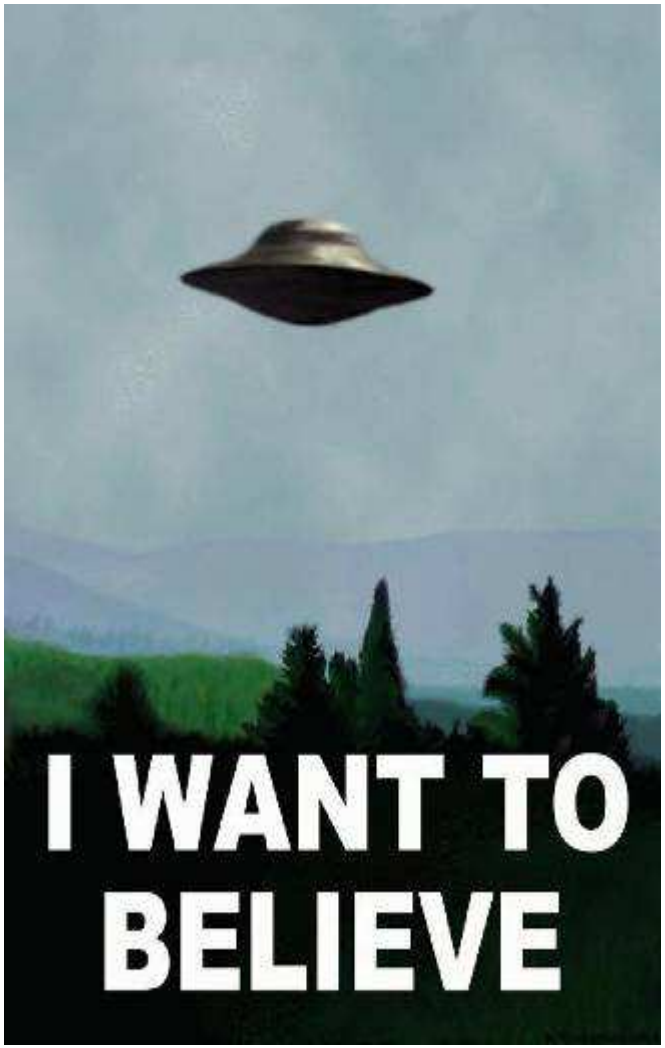
$$x_t = \Phi\, x_{t-h} + \frac{\Phi - 1}{a}\, b\, u_t$$

constant constant

$$\Phi = e^{ah}$$

amazingly trivial
compared to *Spice*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



x	⇒	X
a	⇒	A
b	⇒	B
$\frac{1}{a}$	⇒	*invert(A)*
$*$	⇒	$\bullet$ *(matricial product)*
e^{at}	⇒	e^{At} *(19 + 1 methods ...)*

$$A\,e^{At} = e^{At}\,A$$

***SARC** equations are also applicable to **STATE MODEL** matrices*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

SARC

ERROR ???

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

SARC ERROR

Can I help ???

input *interpolation*

- *order*
- *step*

$\Phi = e^{A\,h}$

choose finer **calculation** **method**

(*may be jeopardized by* **poles** *configuration*)

Euler $e^{ix} = \cos x + i\ \sin x$

complex number

floating number

IEEE STANDARD 754

	float	double	long double	
bits	32	64	80 96	128
digits	7	16	19	34
exponent	≈ ±38	≈ ±308	≈ ±4932	≈ ±4932

x87 standard

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

contribution of interpolation to SARC step ERROR

$$x(t) = e^{a(t-t_0)}\,x(t_0) + \int_{t_0}^{t} e^{a(t-\tau)}\,bu(\tau)\,d\tau$$

$$x(t) = e^{ah}\,x(t-h) + b\,e^{at}\int_{t-h}^{t} e^{-a\tau}\,\text{interpolation}(\tau)\,d\tau$$

Lagrange interpolation step error

$$\varepsilon_{\text{step interpolation}} = b\,e^{at}\,\frac{u^{(n+1)}(\xi)}{(n+1)!}\int_{t-h}^{t} e^{-a\tau}\,\lambda(\tau)\,d\tau$$

with

interpolation points

$$\lambda(\tau) = (\tau - t_0)\ \ldots\ *\ (\tau - t_n)$$

$\xi \in$ interpolation involved segments

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

order n = 0

S0B

step constant = u_{t-h}

$$\varepsilon(k)_{\substack{step \\ interpolation}} = \frac{b\, e^{a\,t}}{1!} \sup_{[t-h,t]} \left\{ \frac{d\,u(t)}{dt} \right\} \int_{t-h}^{t} e^{-a\tau} \left[\tau - (t-h) \right] d\tau$$

$$\Rightarrow \quad \varepsilon(k)_{\substack{step \\ interpolation}} = \frac{1}{a^2} \left[\Phi - ah - 1 \right] \sup_{[t-h,t]} \left\{ b \frac{d\,u(t)}{dt} \right\}$$

order n = 0

S0E

step constant = u_{t}

$$\varepsilon(k)_{\substack{step \\ interpolation}} = \frac{b\, e^{a\,t}}{1!} \sup_{[t-h,t]} \left\{ \frac{d\,u(t)}{dt} \right\} \int_{t-h}^{t} e^{-a\tau} \left[\tau - (t) \right] d\tau$$

$$\Rightarrow \quad \varepsilon(k)_{\substack{step \\ interpolation}} = \frac{1}{a^2} \left[(1-ah)\,\Phi - 1 \right] \sup_{[t-h,t]} \left\{ b \frac{d\,u(t)}{dt} \right\}$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

order n = 1 *slope between* u_{t-h} & u_t

S1

interpolation points

$$\varepsilon(k)_{\substack{step \\ interpolation}} = \frac{b\, e^{a\,t}}{2!} \sup_{[t-h,t]} \left\{ \frac{d^2 u(t)}{dt} \right\} \int_{t-h}^{t} e^{-a\tau} \left[\tau - (t-h)\right] \left[\tau - (t)\right] d\tau$$

$$\varepsilon(k)_{\substack{step \\ interpolation}} = \frac{1}{2\,a^3} \left[(2 - ah)\,\Phi - ah - 2 \right] \sup_{[t-h,t]} \left\{ b\, \frac{d^2 u(t)}{dt^2} \right\}$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

SARC cumulated interpolation ERROR

RECURSIVITY

$$\varepsilon(k)_{\substack{\text{cumulated}\\\text{interpolation}}} = \sum_{i=1}^{k} \left(\phi^{k-i} \, \varepsilon(i)_{\substack{\text{step}\\\text{interpolation}}} \right)$$

constant " *majorant* "

MAJORANT

$$\varepsilon(k)_{\substack{\text{cumulated max}\\\text{interpolation}}} < \left(\frac{1 - \phi^{k}}{1 - \phi} \right) \left| \varepsilon_{\substack{\text{max}\\\text{step}\\\text{interpolation}}} \right|$$

for large k

$t = k\,h$

$$\varepsilon(\text{only large } t)_{\substack{\text{cumulated max}\\\text{interpolation}}} < \frac{\left| \varepsilon_{\substack{\text{max}\\\text{step}\\\text{interpolation}}} \right|}{1 - \phi}$$

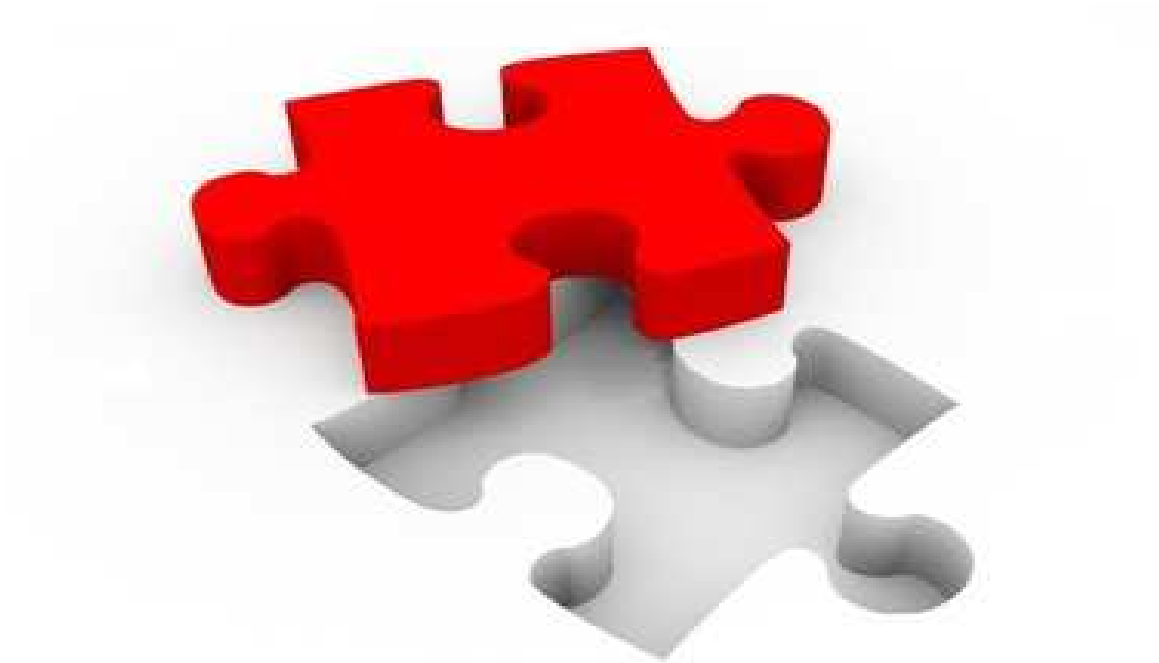
asymptotic

error majorant

(proof of absolute stability)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

estimated SARC error, in practice



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

cumulated SARC error due to **interpolation** can be **estimated** at **run time**

example, for order n = 0 with **step** constant = u_{t-h}

$\frac{\Delta u}{\Delta t}$

$$\varepsilon(k)_{\substack{step \\ interpolation}} = \frac{1}{a^2}\left(\Phi - ah - 1\right)\left(b\,\frac{u(t) - u(t-h)}{h}\right)$$

$$\varepsilon(k)_{\substack{cumulated \\ interpolation}} = \phi\,\varepsilon(k-1)_{\substack{cumulated \\ interpolation}} + \varepsilon(k)_{\substack{step \\ interpolation}}$$

$$\varepsilon(k)_{\substack{cumulated\ max \\ interpolation}} \approx \max\left(\left|\varepsilon(k-1)_{\substack{cumulated \\ interpolation}}\right|, \left|\varepsilon(k)_{\substack{cumulated \\ interpolation}}\right|\right)$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

SARC SUMMARY

$$\Phi = e^{A\,h}$$

recursive convolution

$$\Phi\, X_{t-h} + \sum_{n=0}^{order} \left(c_n\, U_{t - n*h} \right) \Rightarrow X_t \Rightarrow \left(Y \right)$$

likely to engender ***low noise***

$\left(U \right)$ multi ***inputs*** vector

multi ***outputs*** vectors

estimated U_t *(order + 1)*

recursive convolution

$$\Phi\, \varepsilon_{t-h} + \hat{c}_{order} \left\{ \frac{\Delta^{order+1} U}{h} \right\} \Rightarrow \varepsilon_t \Rightarrow \left(Y_{err} \right)$$

ε_{t-h} interpolation

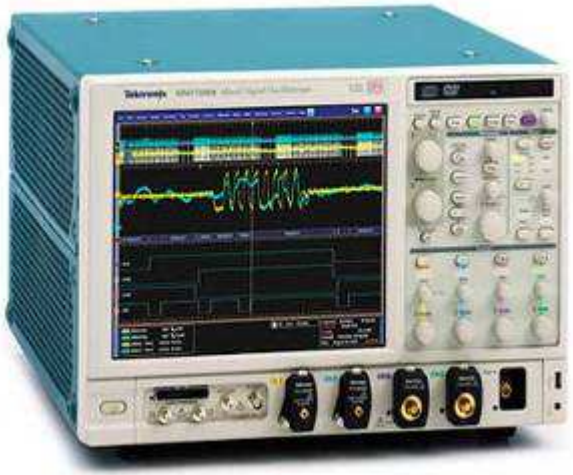
$X_{error\ t-h}$

ε_t interpolation

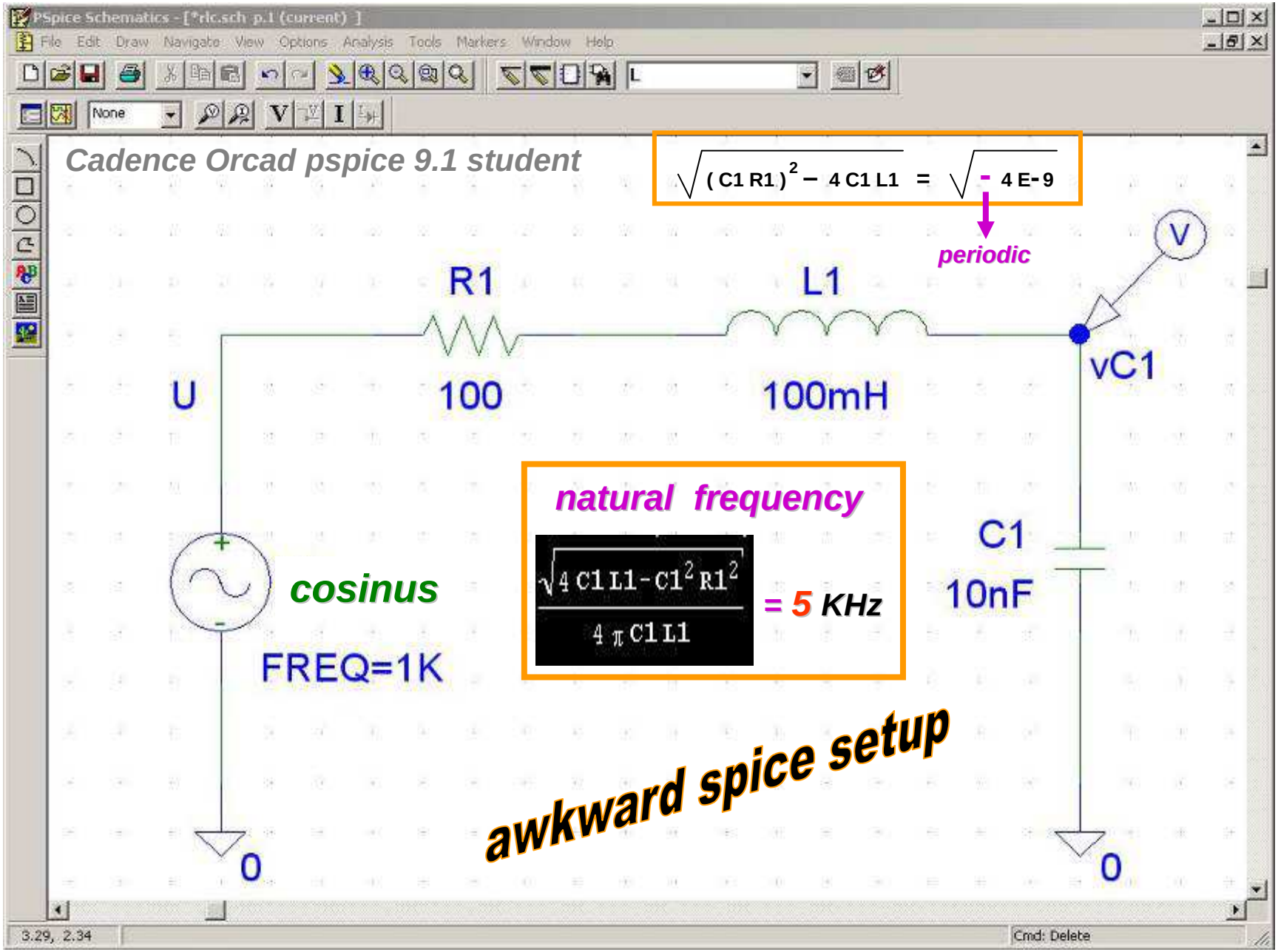
runtime *estimated* *interpolation* ***error***

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

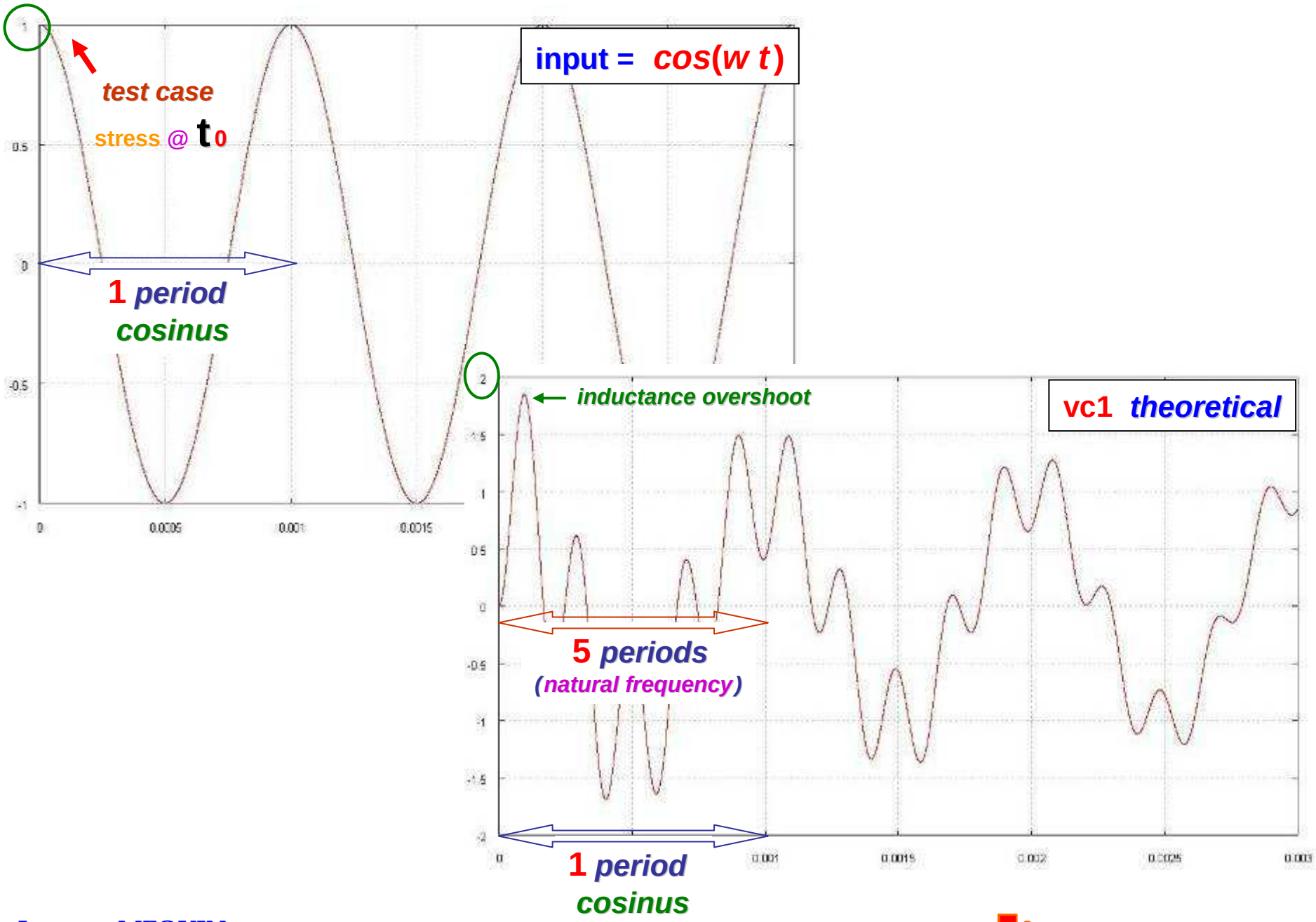
TEST



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



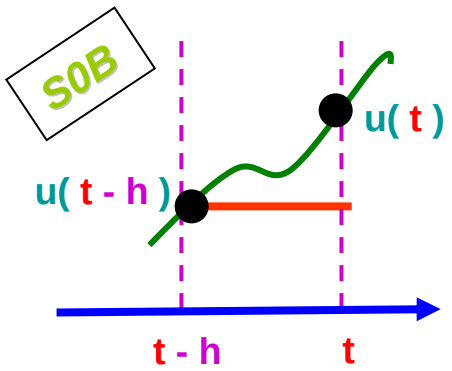
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

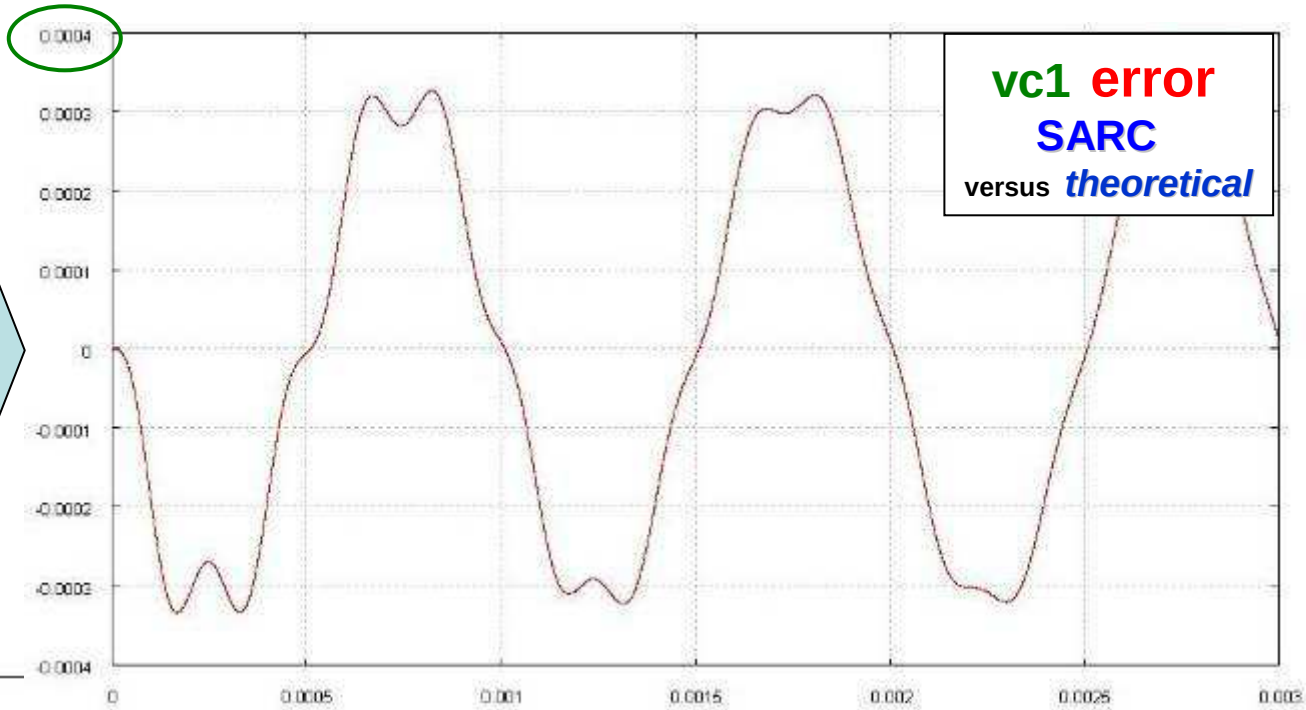


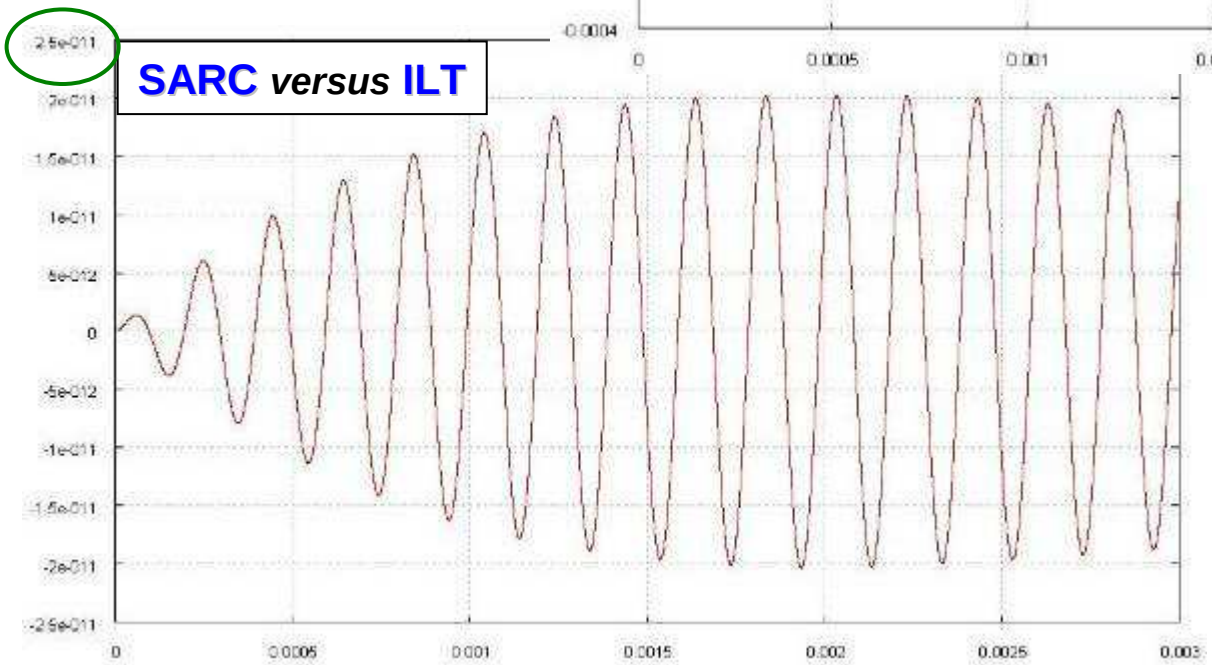
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Lagrange interpolation

order 0



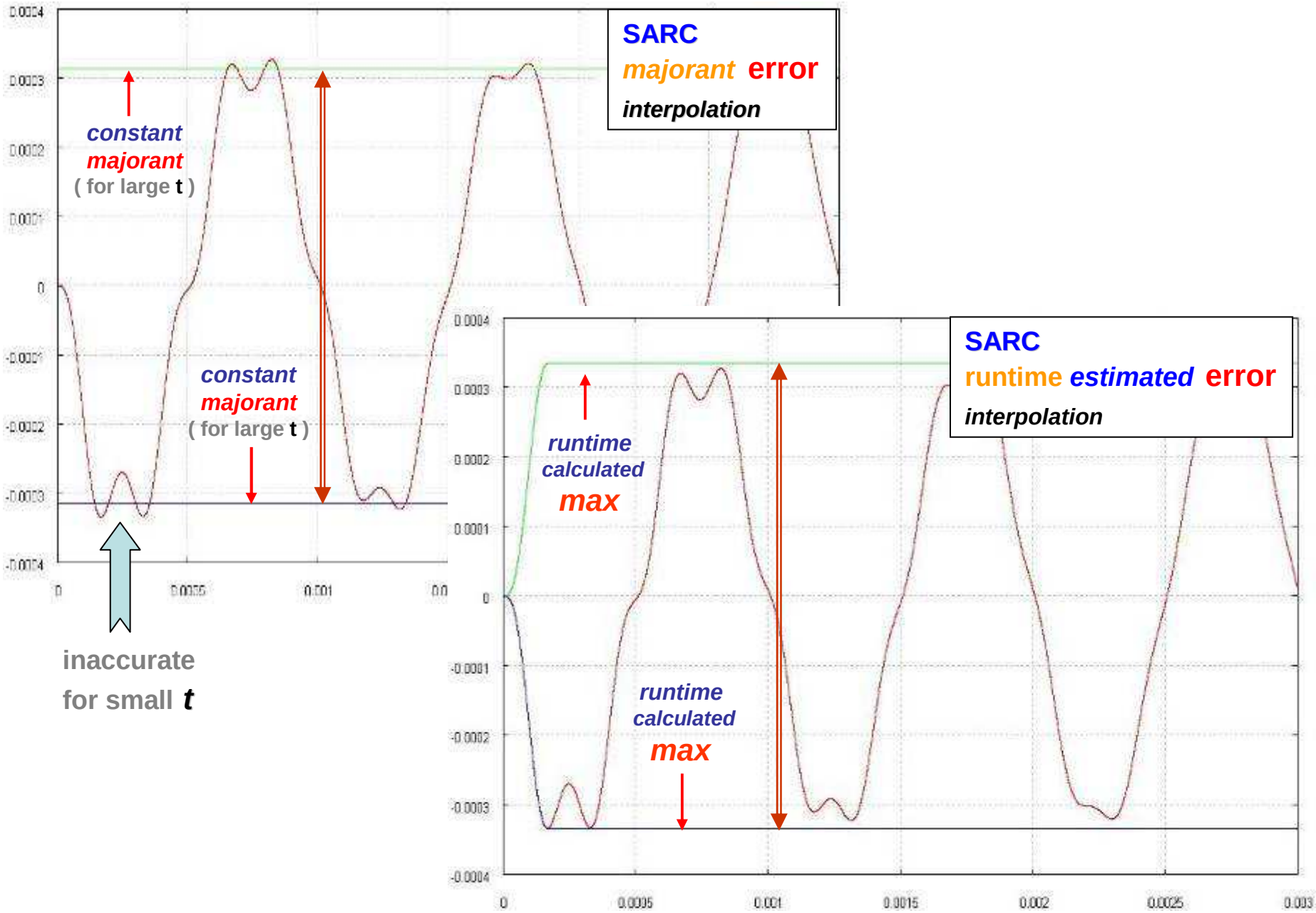




methods comparison

$\approx 10^{-11}$ *!!!*

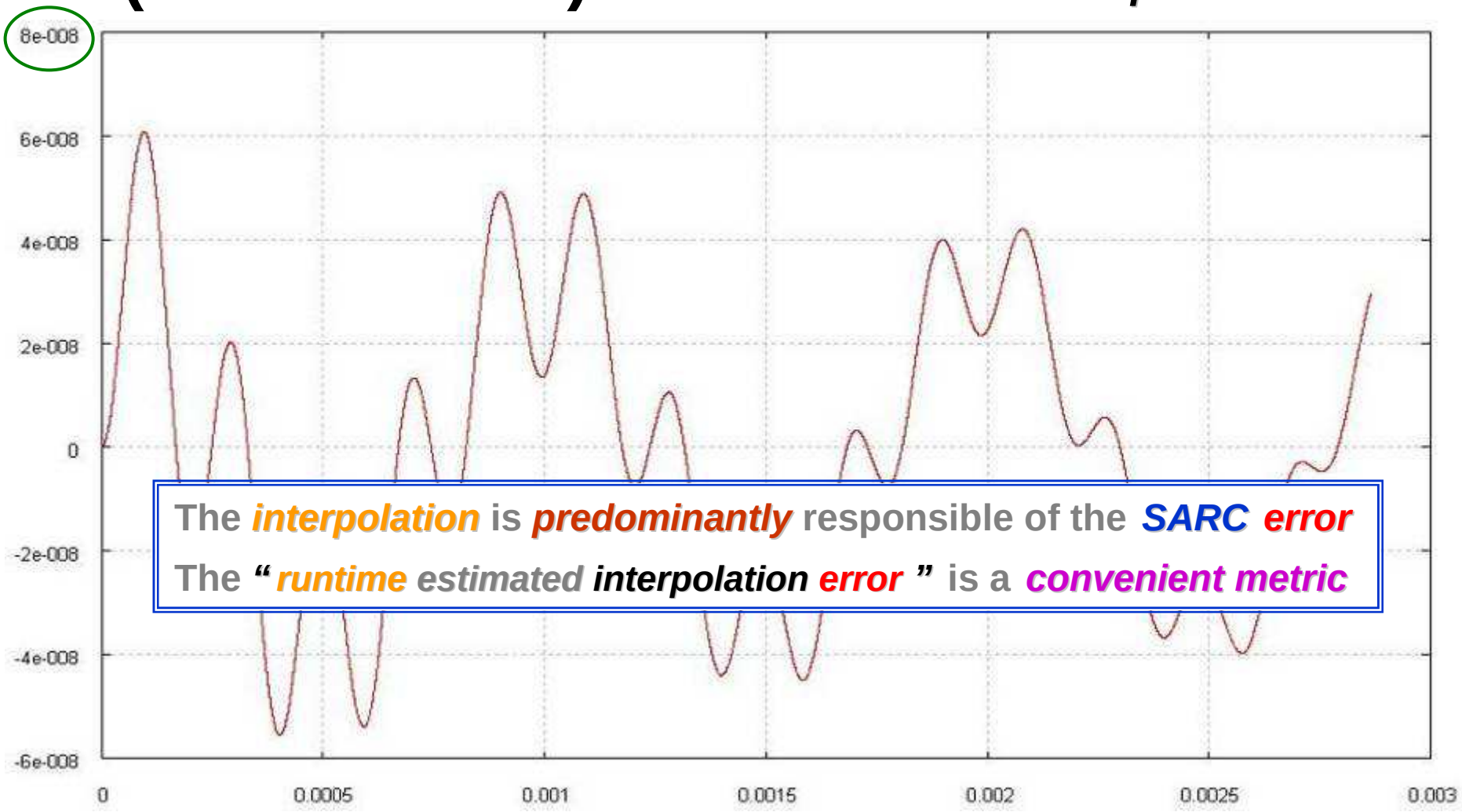
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

full SARC error

(*theoretical* - *SARC*) - *runtime estimated interpolation error*

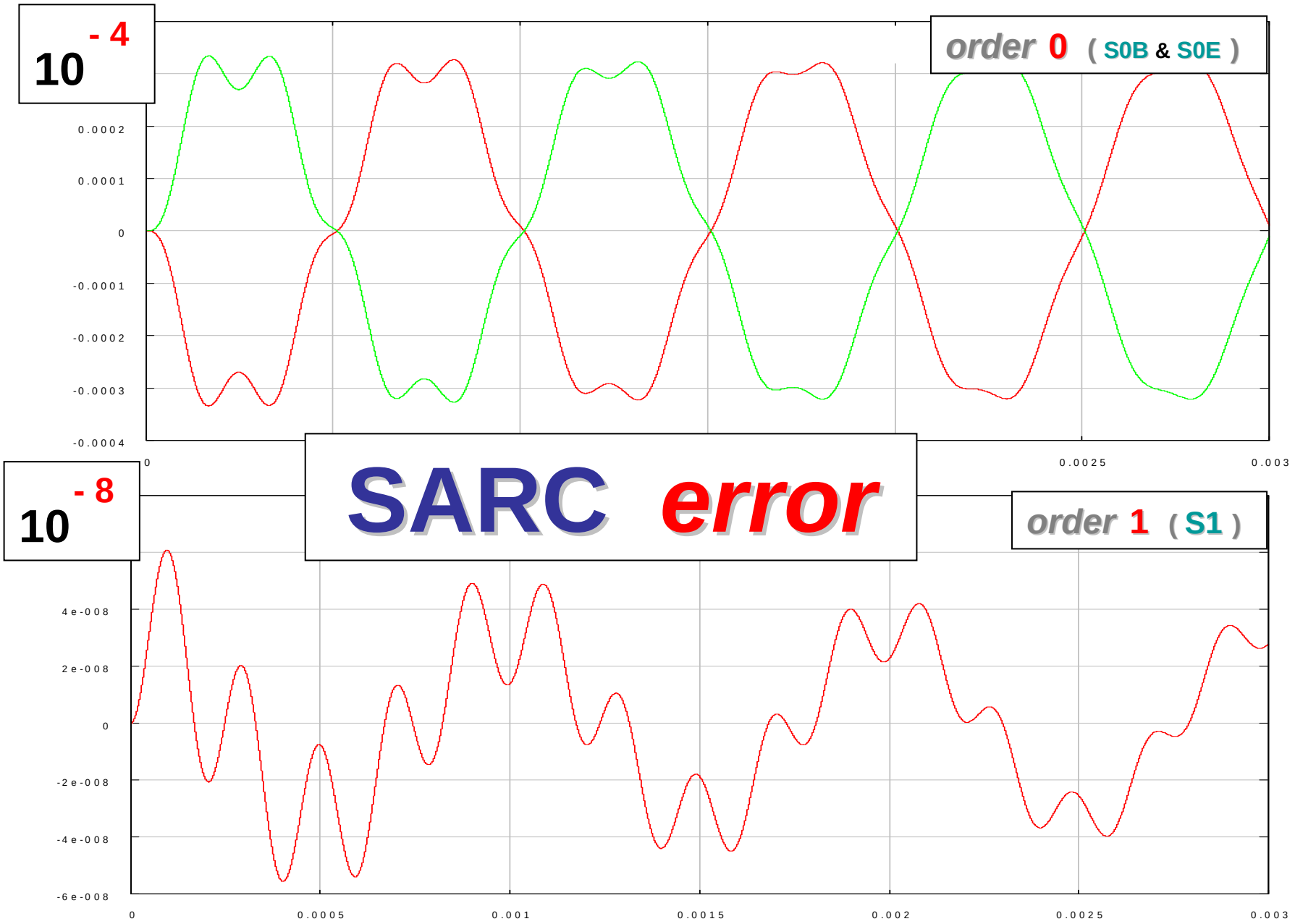


The *interpolation* is *predominantly* responsible of the *SARC error*

The “*runtime estimated interpolation error*” is a *convenient metric*

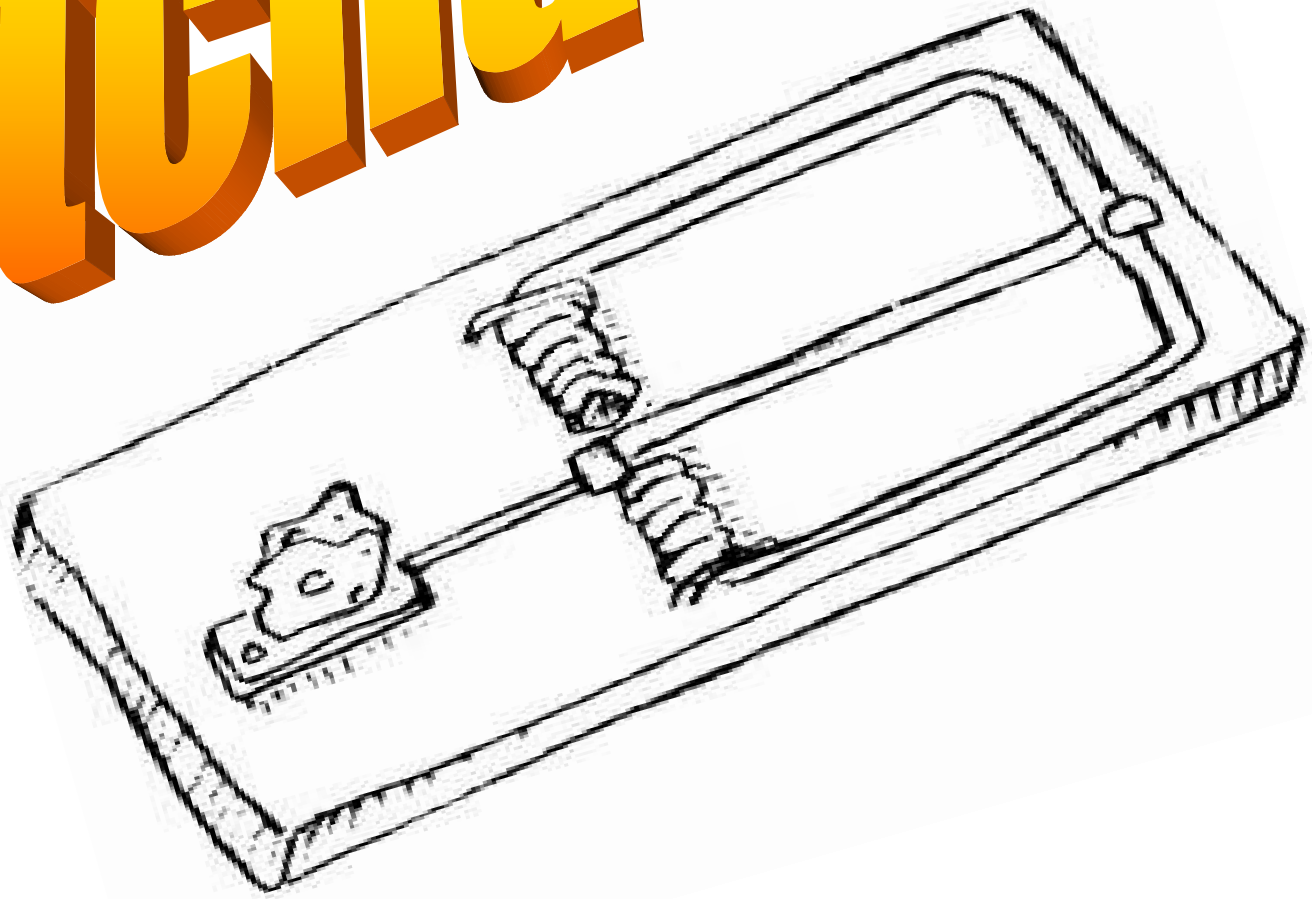
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

SARC error



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Gotcha



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

INITIALIZATION

a single **missing**

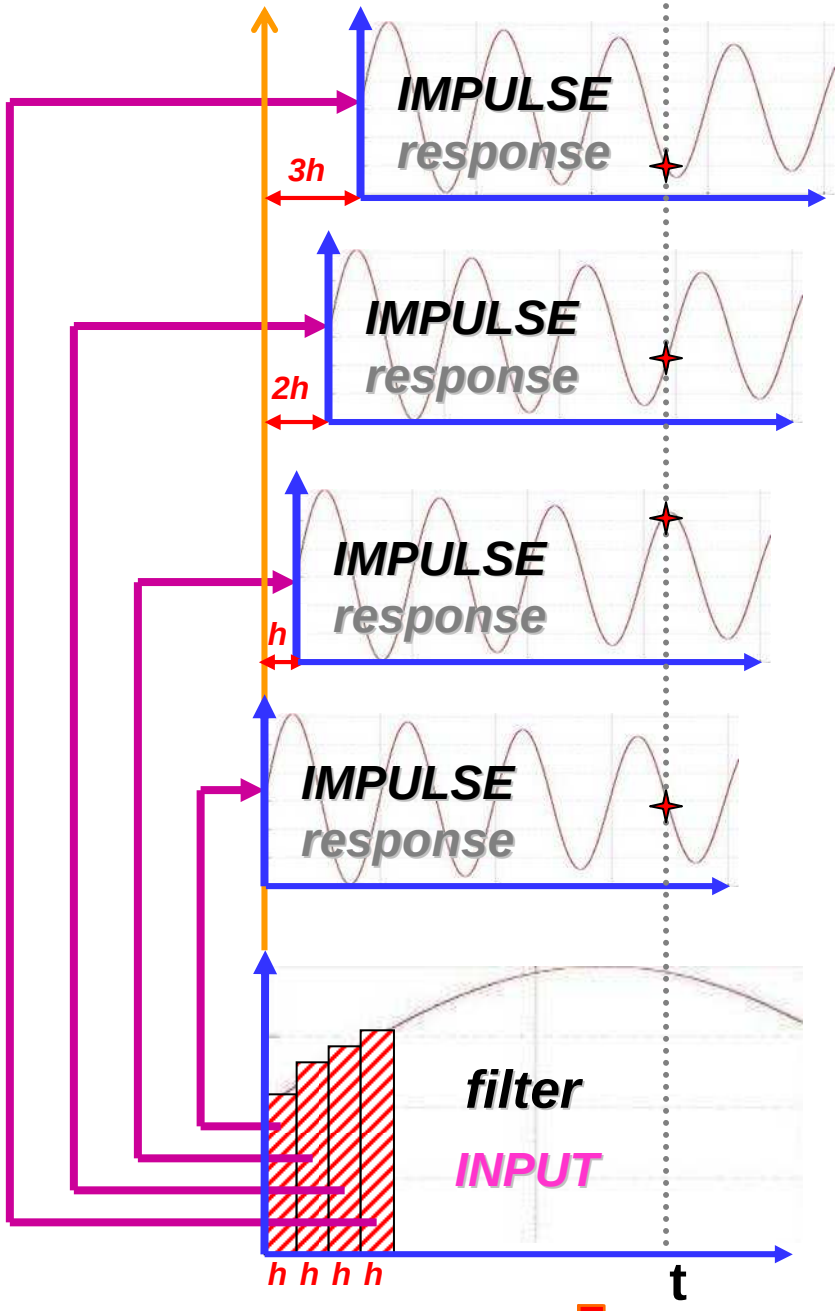
misplaced

or **aberrant**

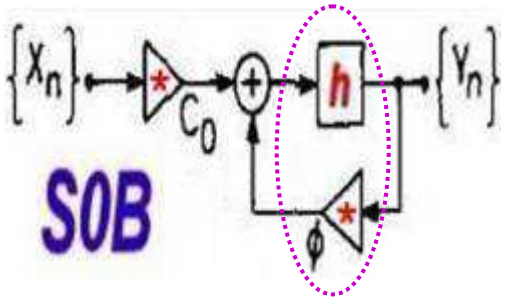
input sample response

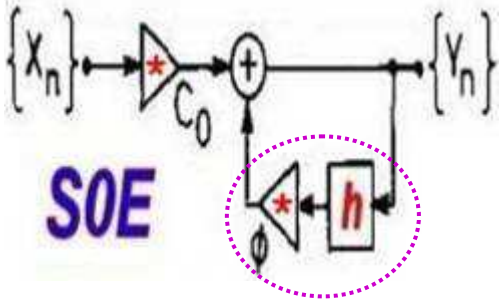
may result in a significant ***ERROR***

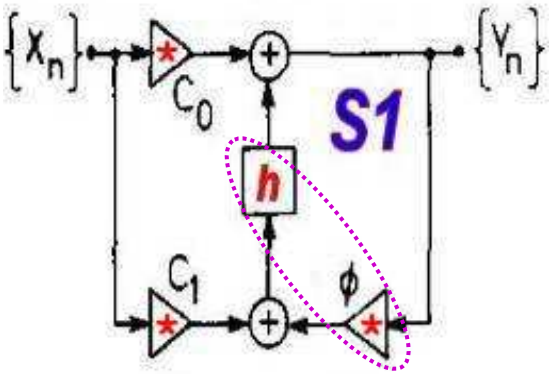
over a long interval

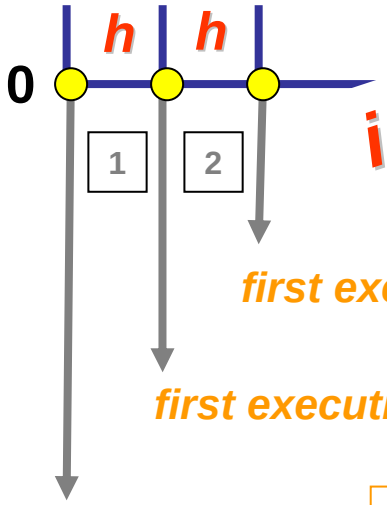


Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com









initialization

ALL U(t) samples MUST BE coherent

in particular for signal not null in the t = 0 vicinity

t

example: R L C / complex poles / slow decay / cosinus input

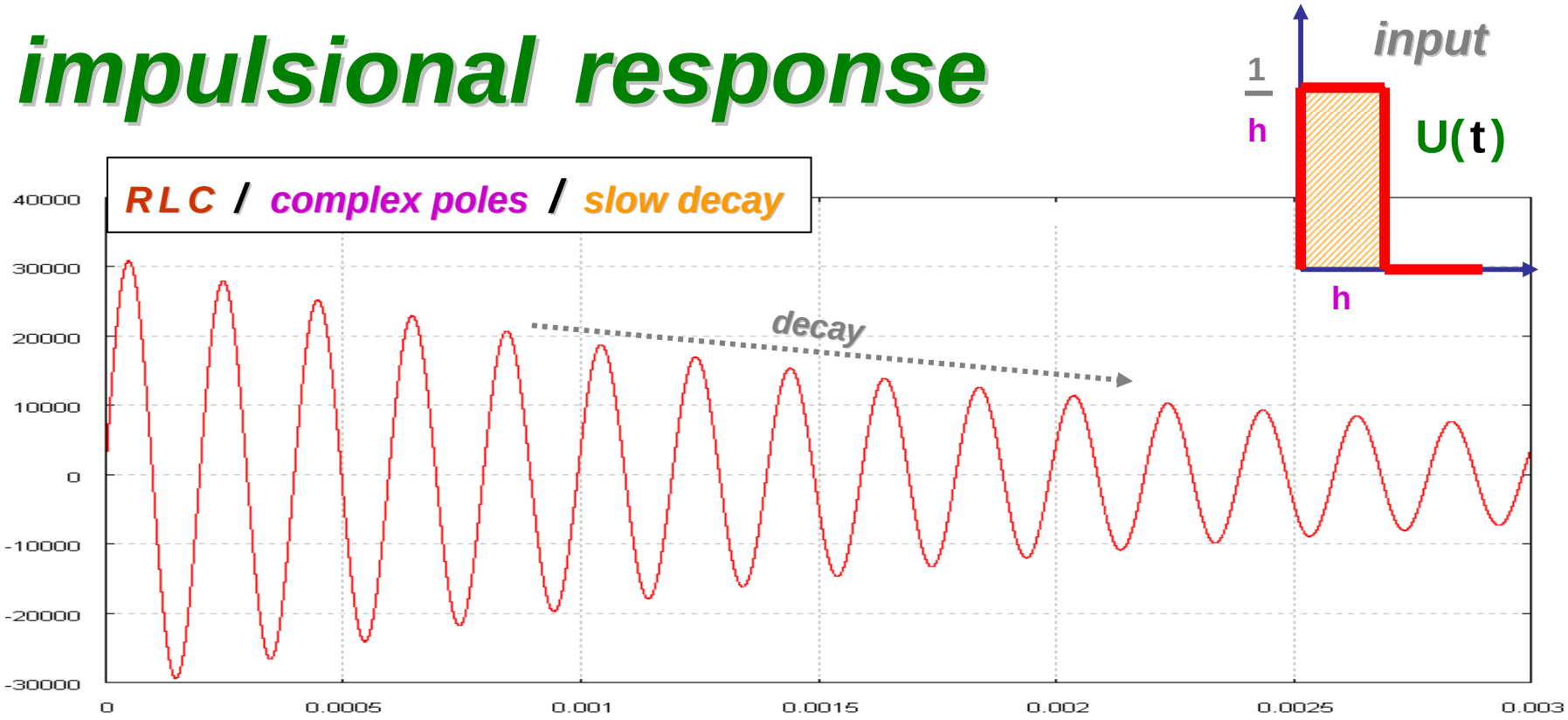
first execution — **order 1 estimated error** — (cf 2nd derivative)

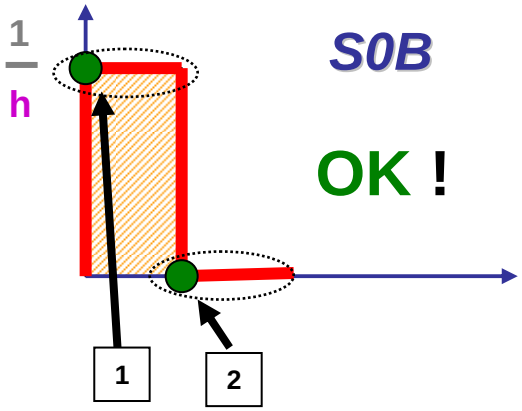
first execution — **SARC and order 0 estimated error** — (cf t - h >= 0)

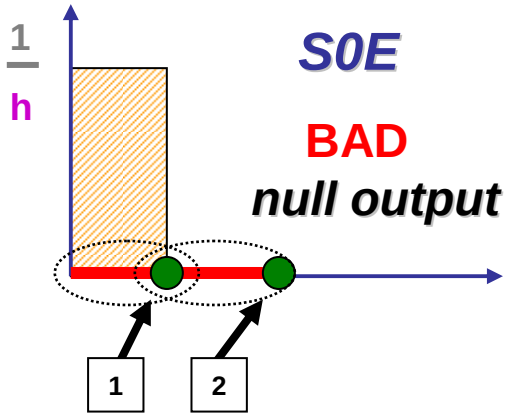
first execution — **ILT** — (and ... S0B If we neglect that S0B(t = 0) = 0 by definition)

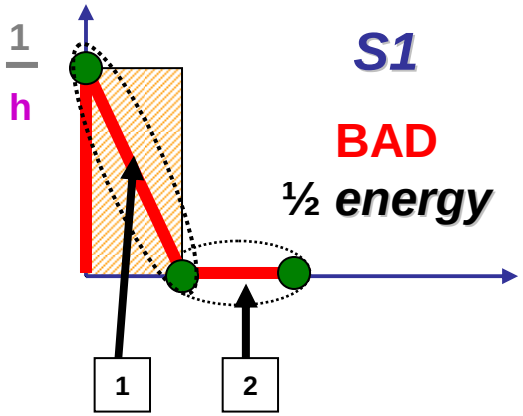
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

impulsional response

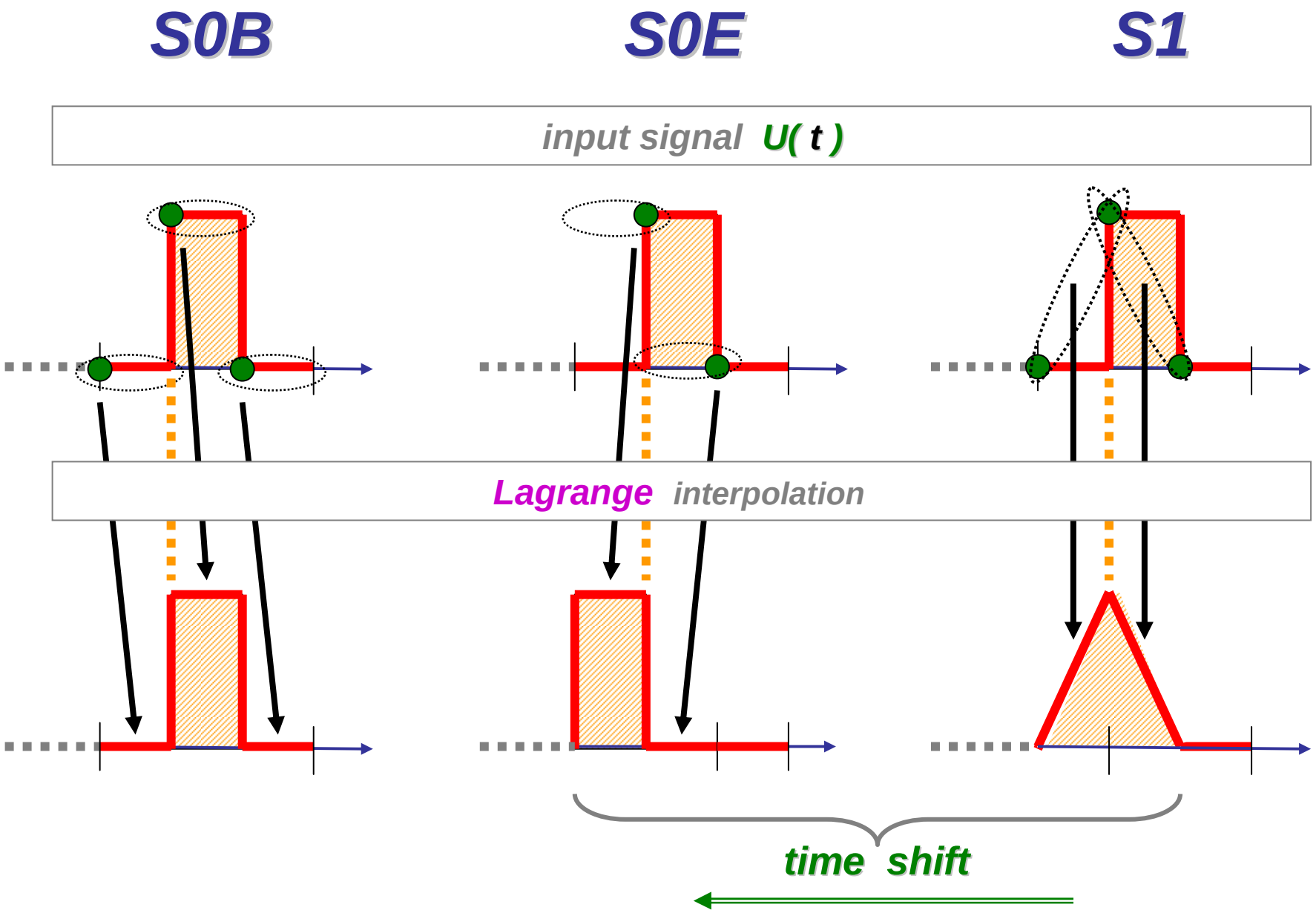








Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

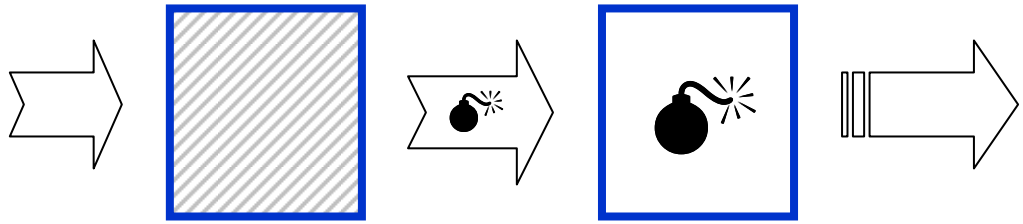
order 1 *is better than* **order 0**

BUT

general purpose simulators cascading sub-circuits

cannot tolerate initialization missing *output samples*

U(0)

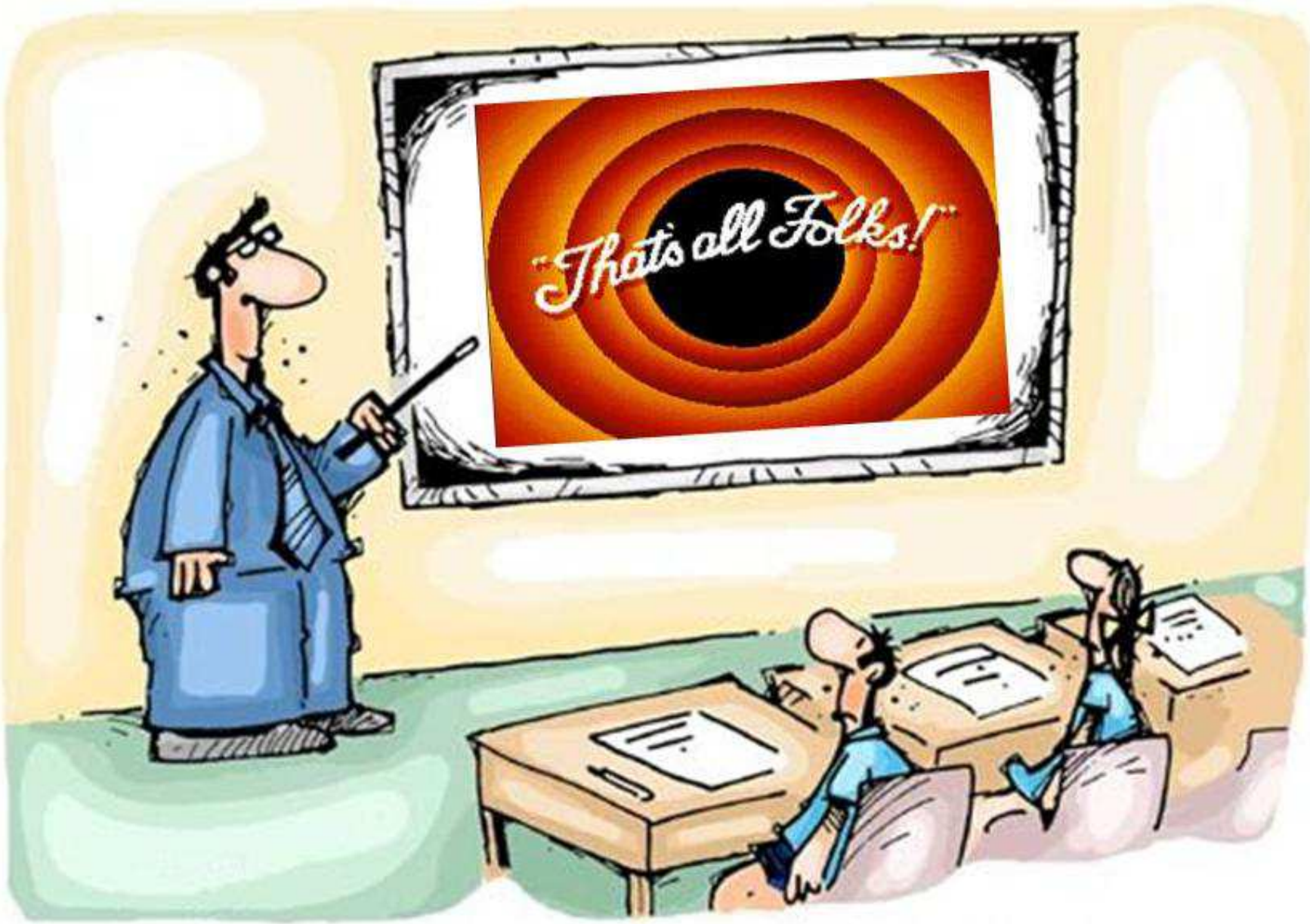


So, the

is more likely to be **S0B**

similar to ***ILT***

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Wait a minute ,

we have not talked

about

PERCUSSIONS

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

CHARGE conservation

$$Q = C \quad V$$

$$Q = C' \quad V'$$

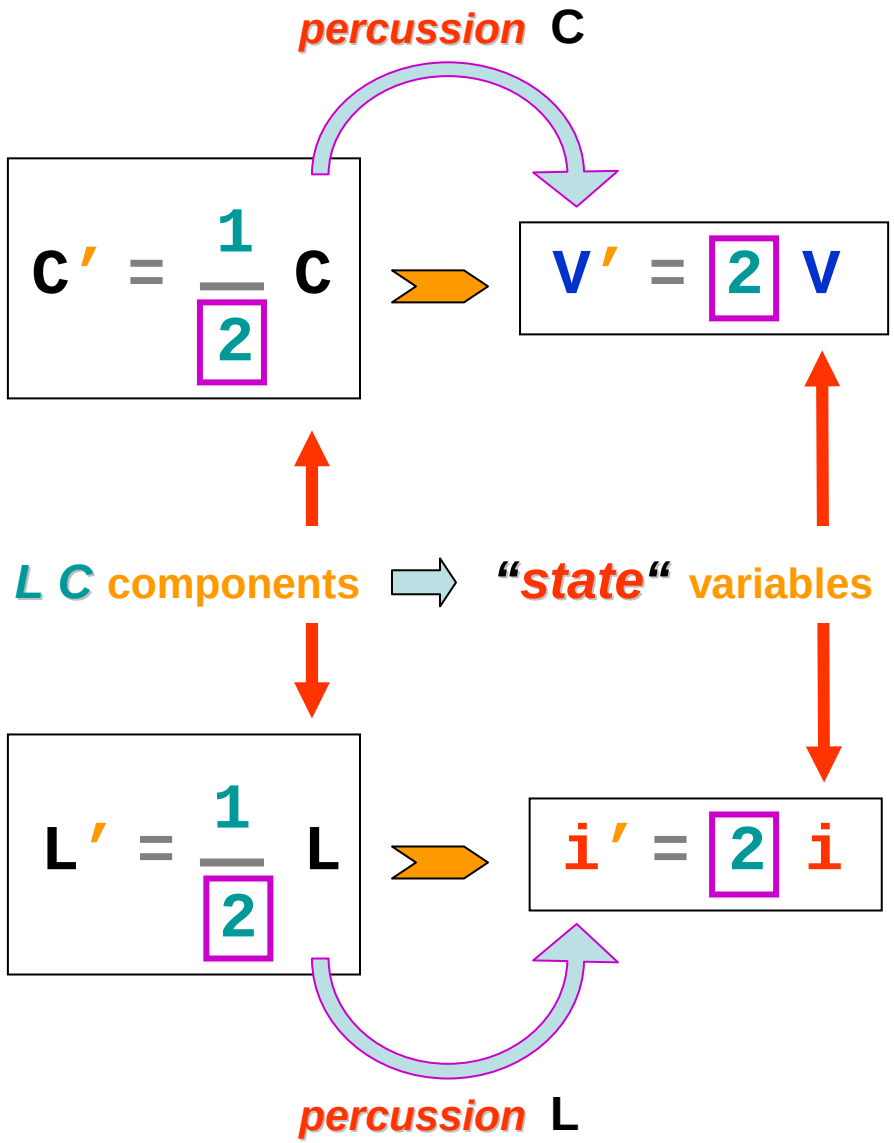
$$V' = \frac{C}{C'} V$$

FLUX conservation

$$\phi = L \quad i$$

$$\phi = L' \quad i'$$

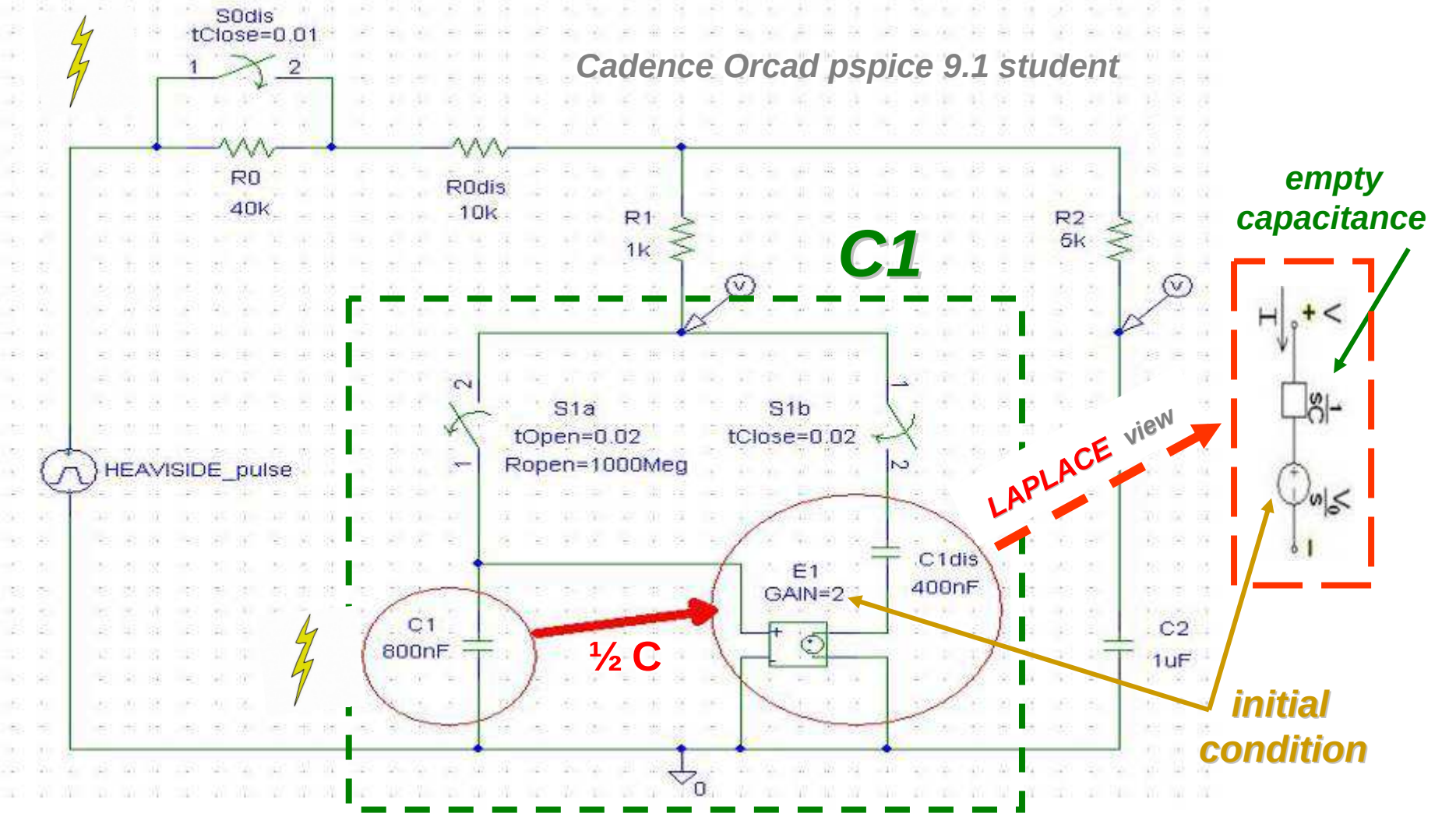
$$i' = \frac{L}{L'} i$$



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

PERCUSSION

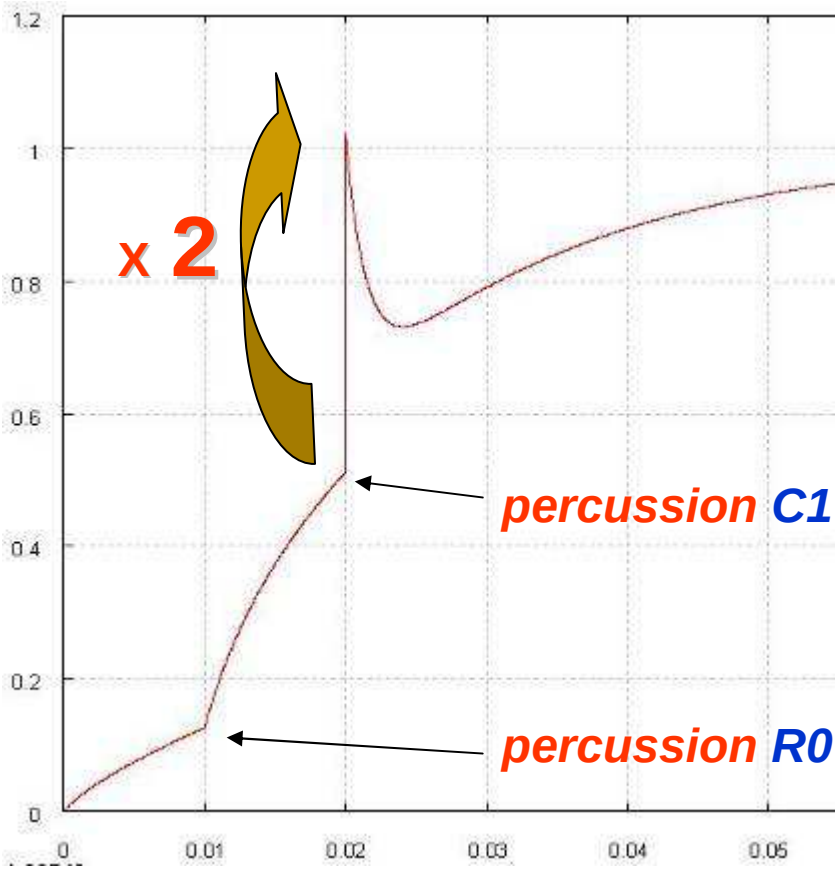
at	t =	0.01	R0	50 Kohm	==>>	10 Kohm
at	t =	0.02	C1	800 nF	==>>	400 nF

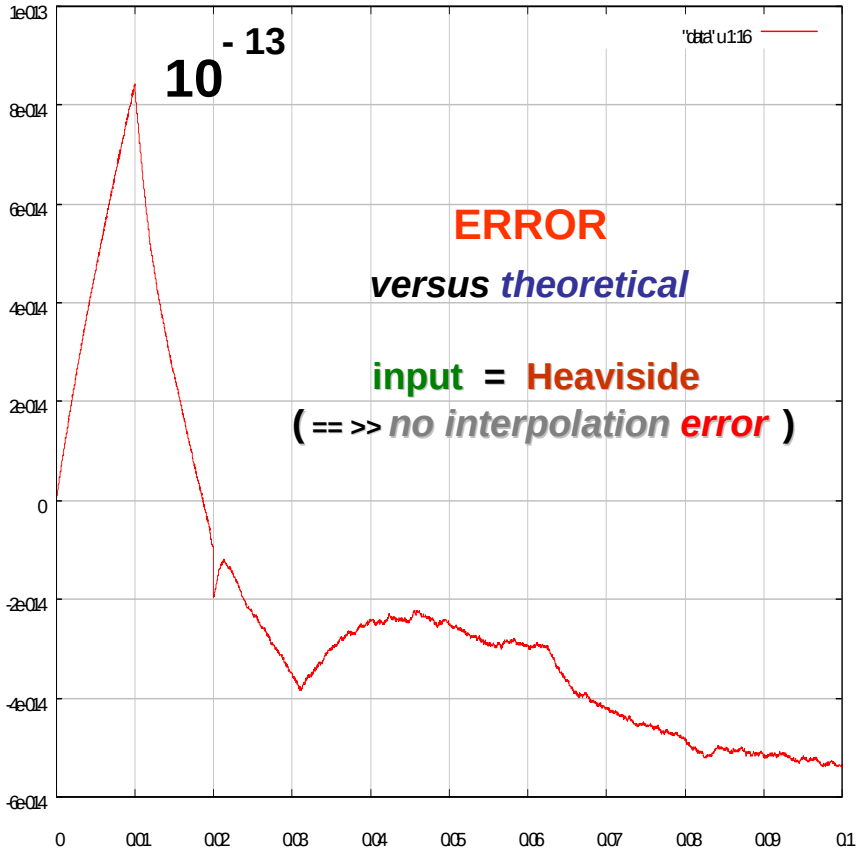


In fact, at least for network not " degenerate ", it is trivial !!!

At the correct instant, update the percurted component values (R0 or C1)

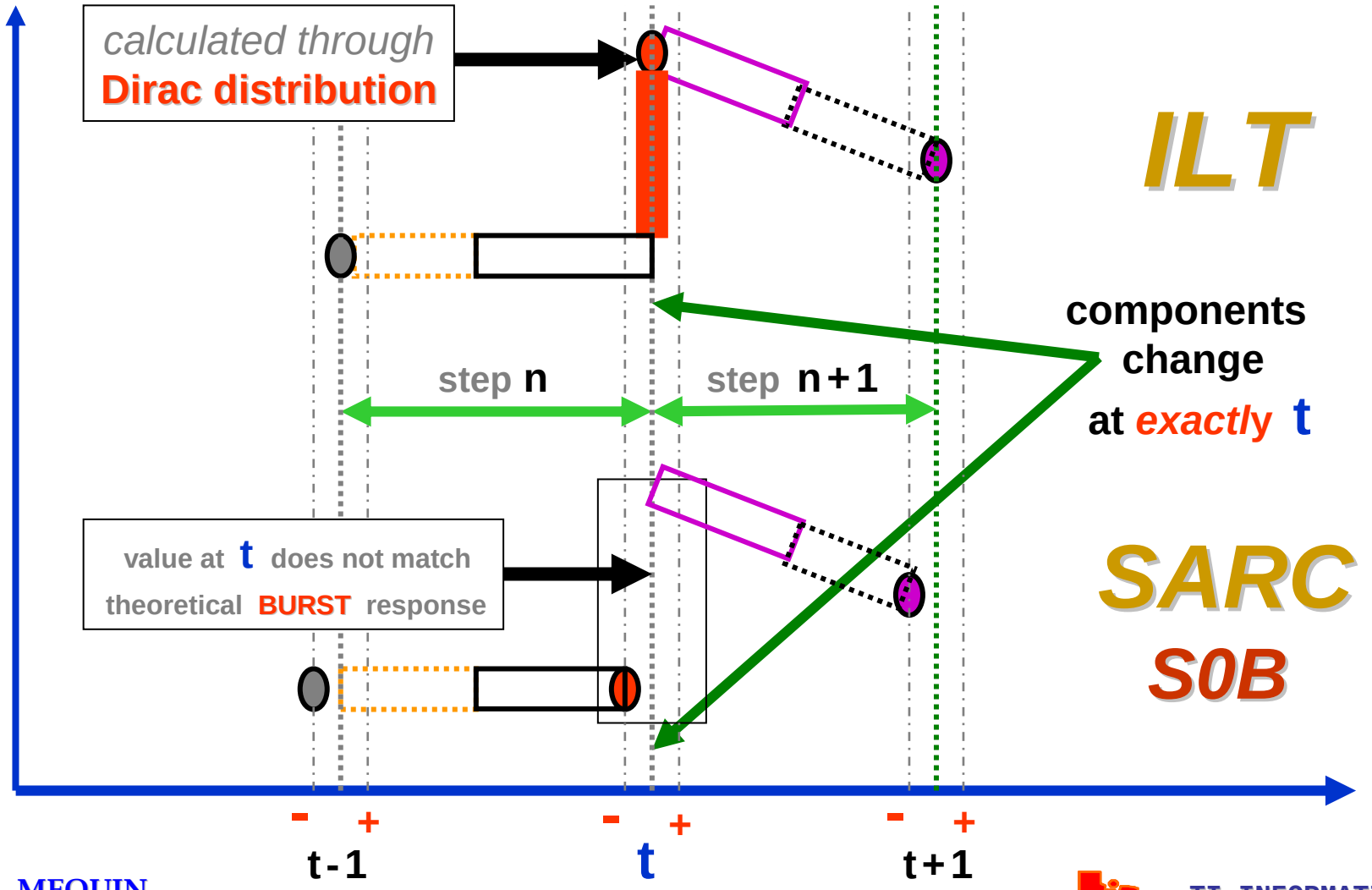
and for the capacitance C1 "double" its **STATE VARIABLE** voltage



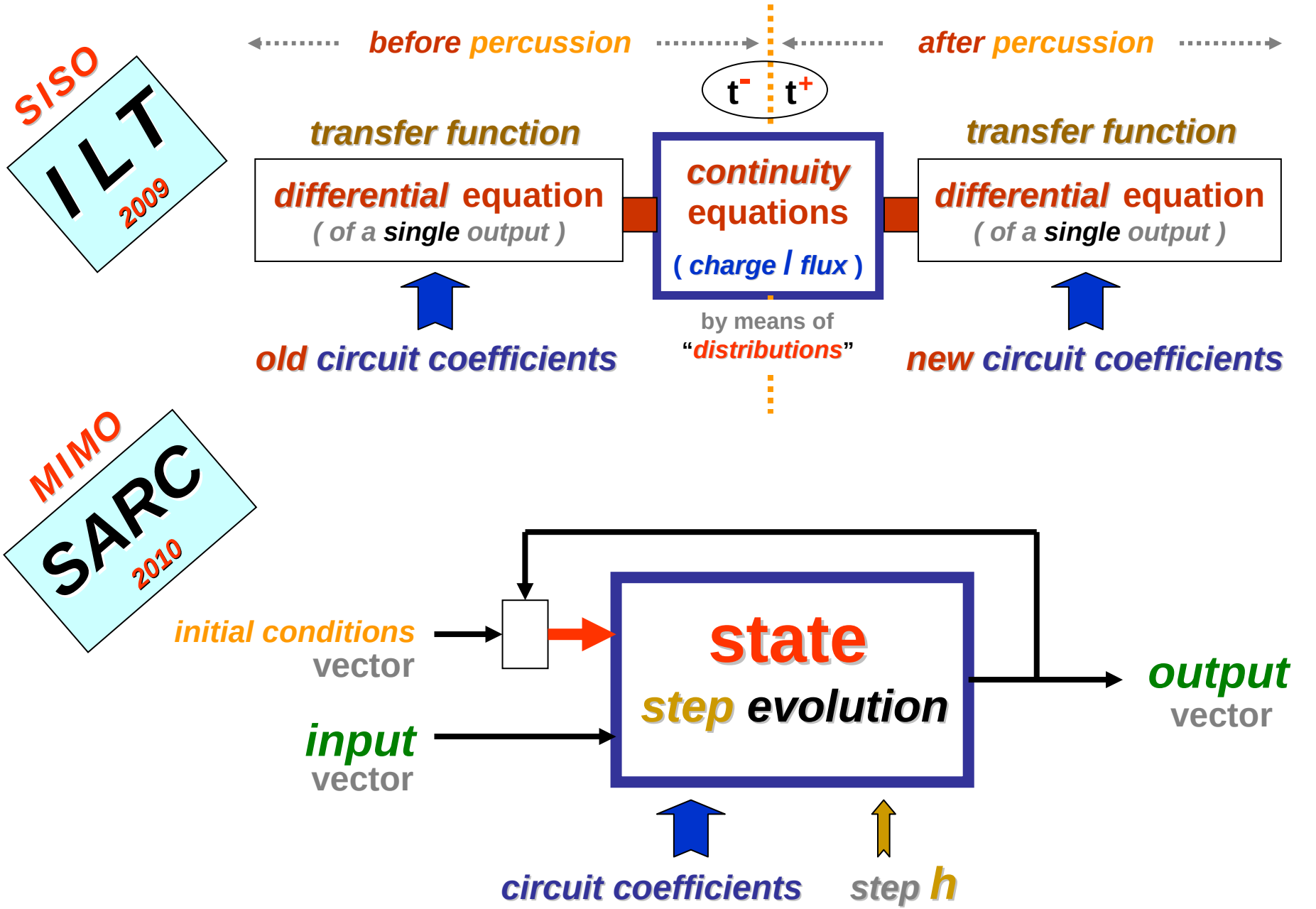


Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

SARC versus ILT percussion



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

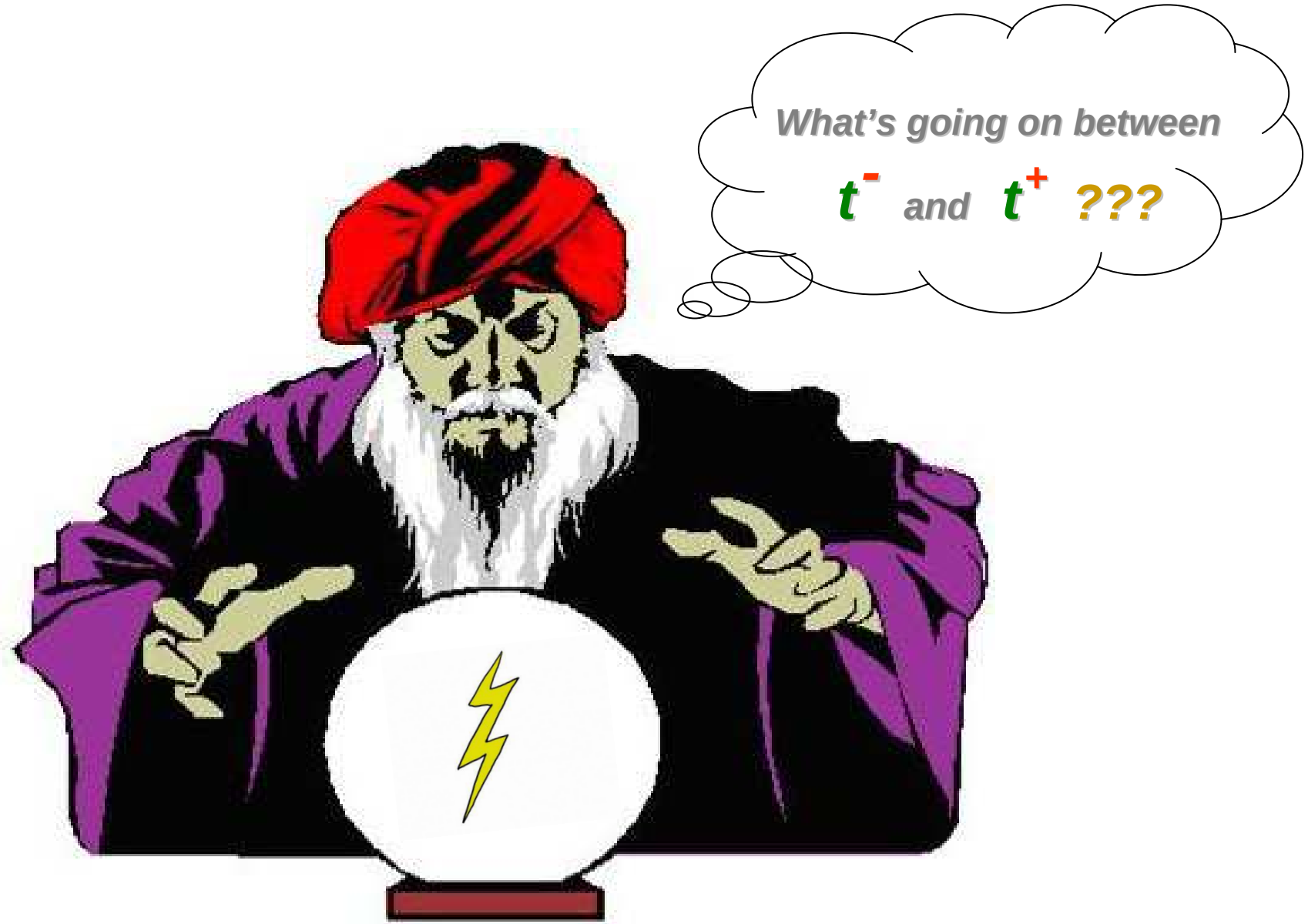
QUESTIONS ?

Component PERCUSSION

of

Linear Time Invariant analog circuit

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Digital and Analog simulations are the 2 faces of a same coin

In the ***digital world*** architects require (through ***ESL***) ***high level functional*** simulations to perform numerous ***"what if"*** and ***analysis*** in order to challenge the concept and to ***increase confidence*** about its validity

Unfortunately in the ***analog world*** architects do have much less tools and methodologies available to perform equivalent ***high level functional*** simulations

In many cases, ***SPICE*** turns to be ***the only practical approach***

This is unfortunate because ***SPICE*** is so ***low level*** that its ***runtime*** quickly becomes a bridle to the architect ambition

Less experiments, Less tested alternatives, Less confidence …

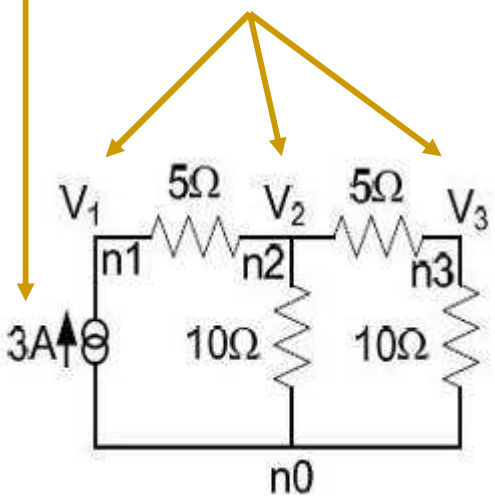
SPICE

Simulation Program with Integrated Circuit Emphasis

current source

("**Norton**" equivalent for **Voltage**)

nodes



"NODAL" equations based on KCL

(Kirchhoff Current Law $\Sigma I = 0$)

$$[G] * [V] = [I]$$

conductance

$$\begin{bmatrix} 0.2 & -0.2 & 0 \\ -0.2 & 0.5 & -0.2 \\ 0 & -0.2 & 0.3 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \\ 0 \end{bmatrix}$$

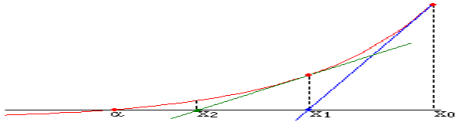
"Gaussian" elimination

to make it "**triangular**"

then substitutions ...

$$\begin{bmatrix} 0.2 & -0.2 & 0 \\ 0 & 0.3 & -0.2 \\ 0 & 0 & 0.25 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix}$$

NON-LINEAR COMPONENT behavior is approximated by doing iterations (convergence) based on Newton-Raphson method (improvement of an equation root "guess")



$$f(x) \cong f(x_0) + f'(x_0)(x - x_0)$$

==> recurrence

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

NUMERICAL INTEGRATION (for linear inductances and capacitances) is approximated according to 3 user-selectable methods : Trapezoidal , backward-Euler , Gear

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

NUMERICAL INTEGRATION (for linear inductances and capacitances)

Why several different methods ?

Depending on the "expected" waveform, one method may have an advantage over another

What makes them strong or weak ?

Accuracy How much error (Local Truncation Error) each method introduces ?

Stability Does the error accumulate (that's bad) or decrease to zero (that's good) over time ?

Each method has its own strengths and weaknesses

Trapezoidal

$$x_{n+1} = x_n + h \cdot \frac{(\dot{x}_{n+1} + \dot{x}_n)}{2}$$

Better accuracy than stability
may oscillate in some cases

Backward Euler

$$x_{n+1} = x_n + h \cdot \dot{x}_{n+1}$$

Average accuracy and stability

Gear-2

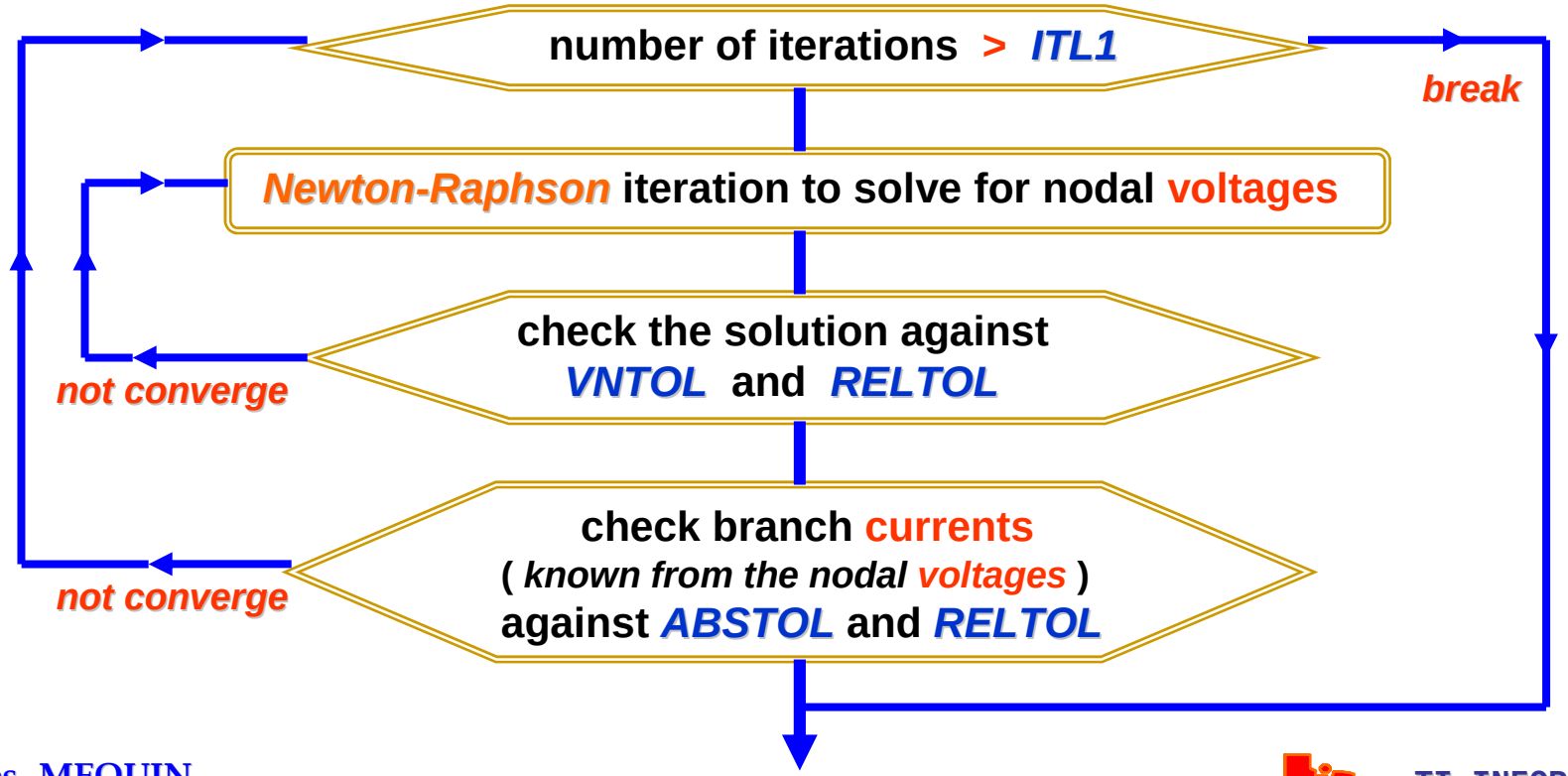
$$x_{n+1} = \frac{4}{3} \cdot x_n - \frac{1}{3} x_{n-1} + \frac{2}{3} \cdot h \cdot \dot{x}_{n+1}$$

Better stability than accuracy
Most stable method

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

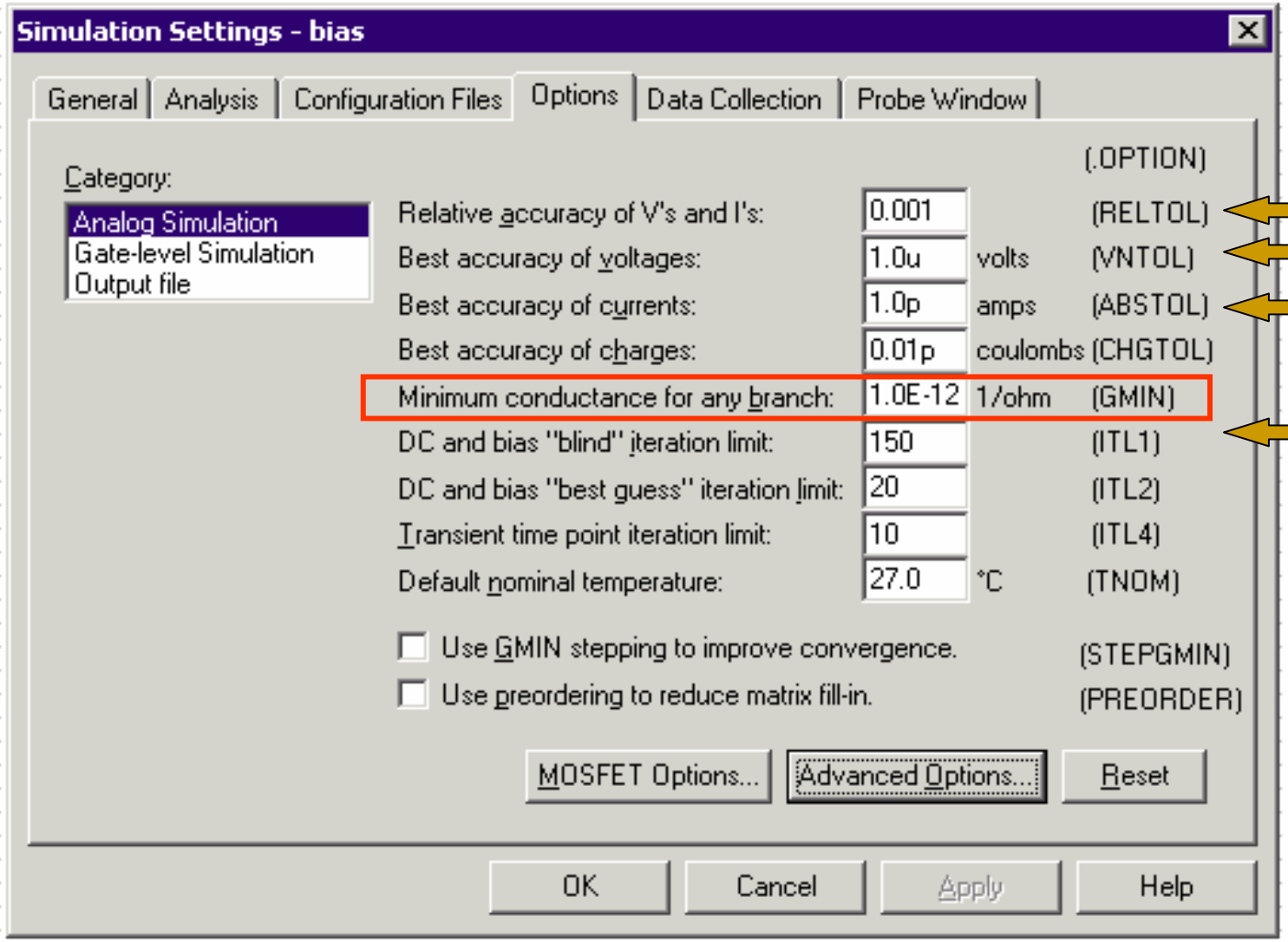
SPICE "convergence" control variables

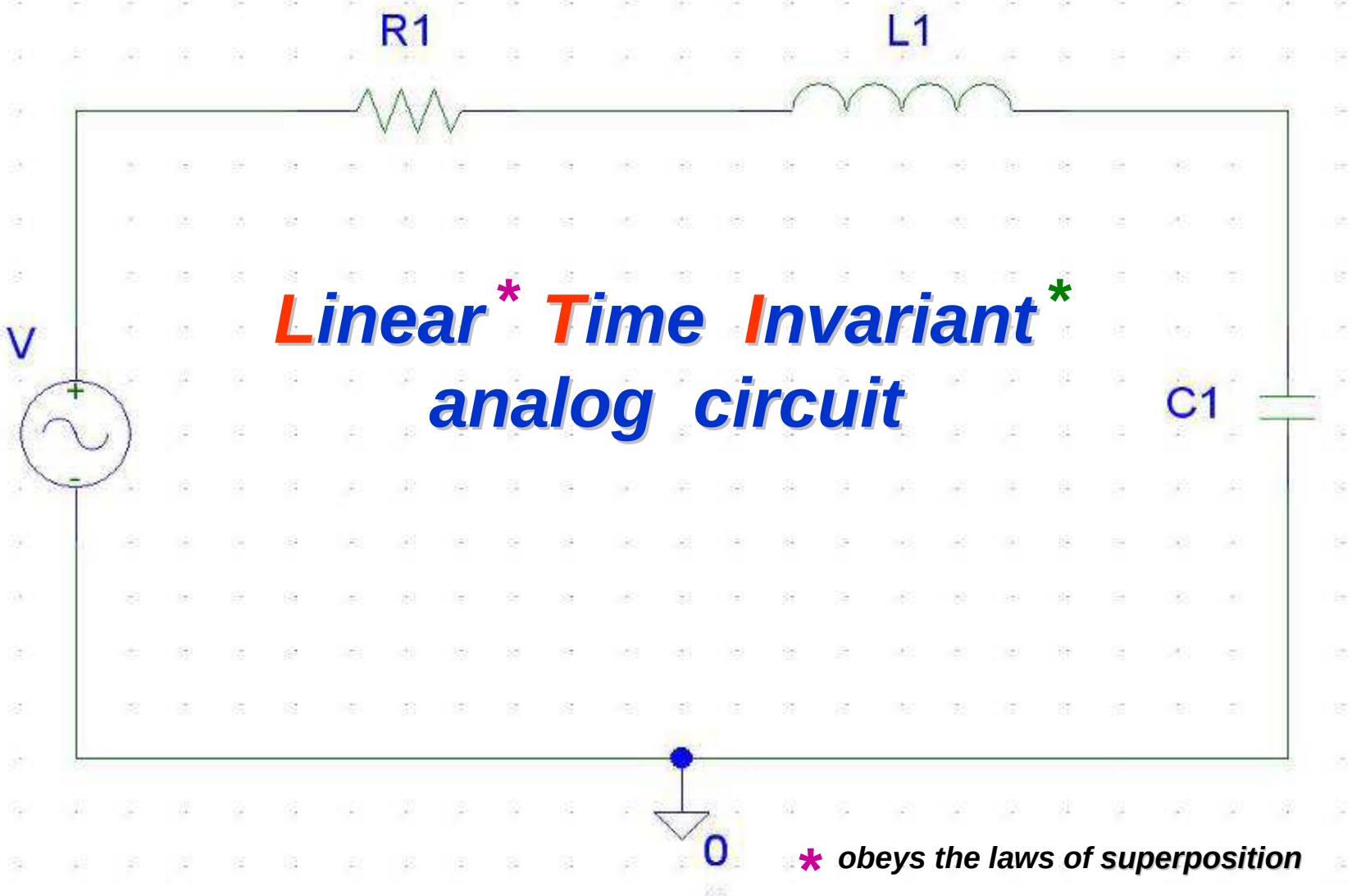
ITL1	number of iterations allowed for a bias point calculation
VNTOL	node voltage tolerance
RELTOL	fractional tolerance of voltages and currents
ABSTOL	current branch tolerance



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

SPICE


Simulation Settings - bias
General
Analysis
Configuration Files
Options
Data Collection
Probe Window
Category:
Analog Simulation
Gate-level Simulation
Output file
(.OPTION)
Relative accuracy of V's and I's: 0.001 (RELTOL)
Best accuracy of voltages: 1.0u volts (VNTOL)
Best accuracy of currents: 1.0p amps (ABSTOL)
Best accuracy of charges: 0.01p coulombs (CHGTOL)
Minimum conductance for any branch: 1.0E-12 1/ohm (GMIN)
DC and bias "blind" iteration limit: 150 (ITL1)
DC and bias "best guess" iteration limit: 20 (ITL2)
Transient time point iteration limit: 10 (ITL4)
Default nominal temperature: 27.0 °C (TNOM)
Use GMIN stepping to improve convergence. (STEPGMIN)
Use preordering to reduce matrix fill-in. (PREORDER)
MOSFET Options...
Advanced Options...
Reset
OK
Cancel
Apply
Help



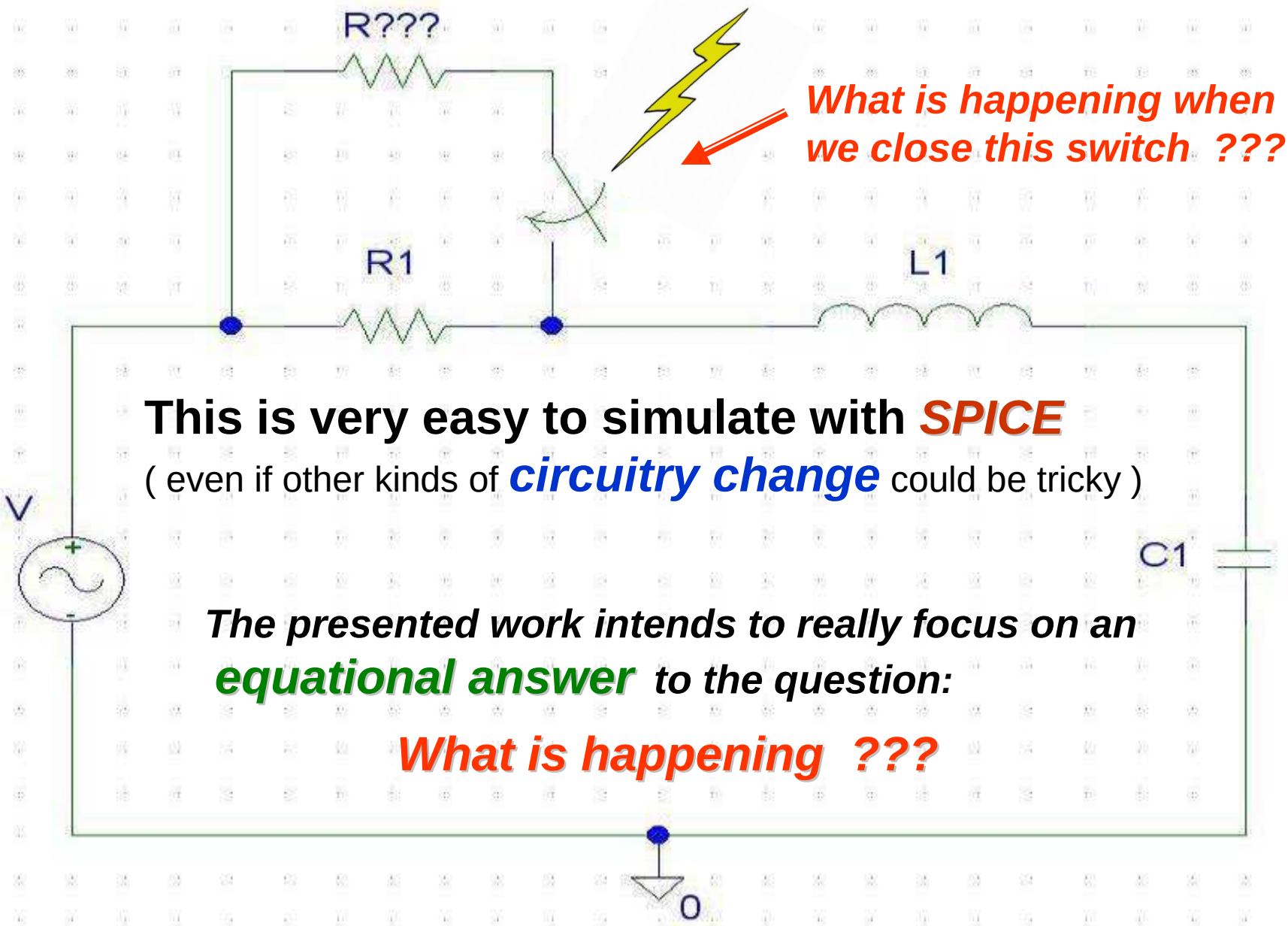
Linear Time Invariant* analog circuit*

* *obeys the laws of superposition*

* *does not change its behavior over time*

(i.e. constant intrinsic parameters)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



What is happening when we close this switch ???

This is very easy to simulate with *SPICE*

(even if other kinds of ***circuitry change*** could be tricky)

The presented work intends to really focus on an

equational answer ***to the question:***

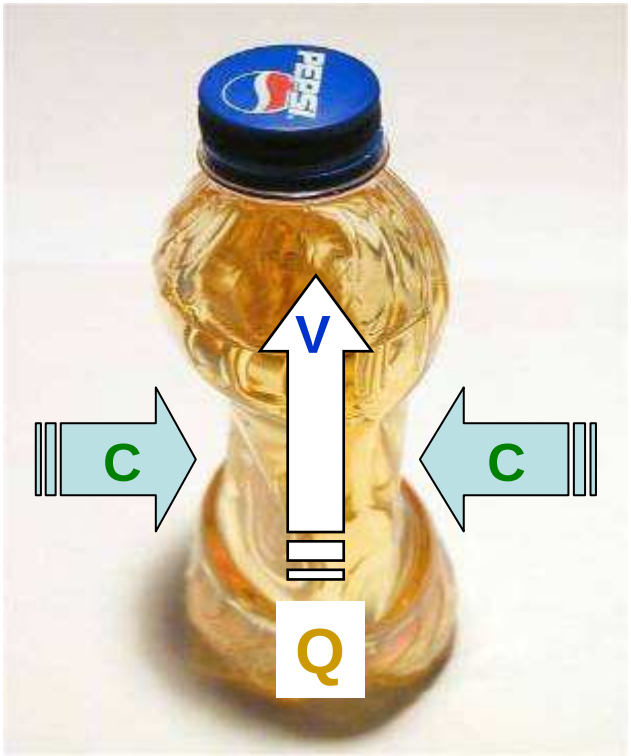
What is happening ???

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

other example:

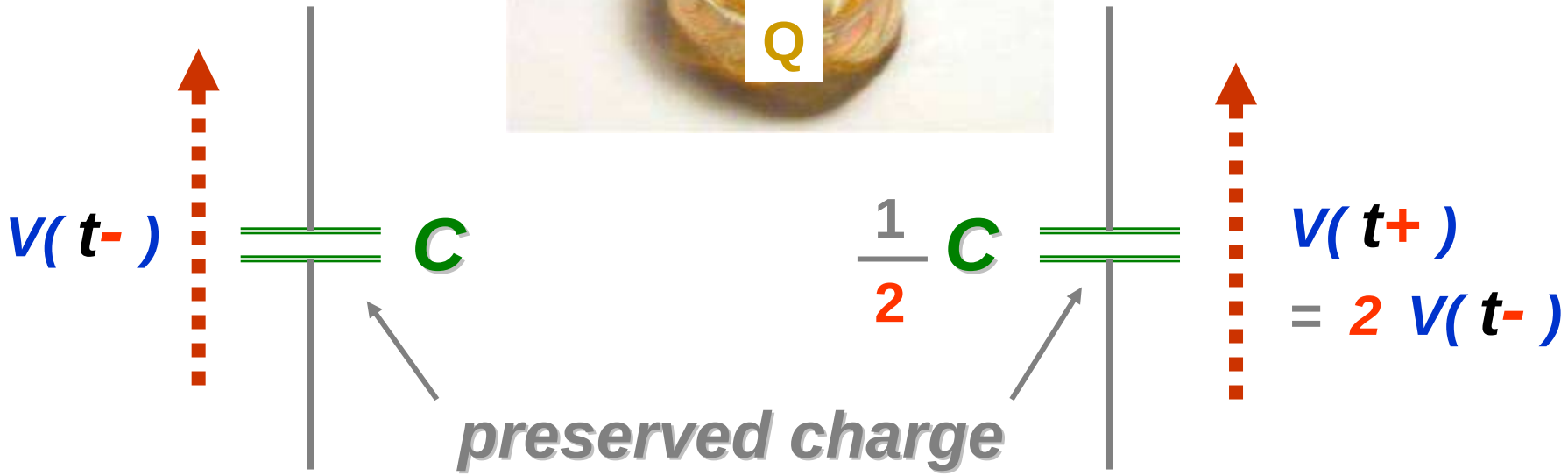
capacitance

change ?



$$Q = C\,V$$

$$V = \frac{Q}{C}$$



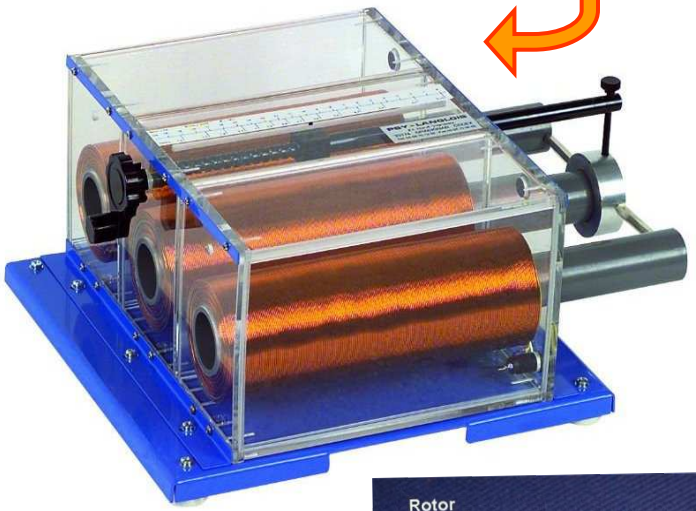
preserved charge

$Q(t-)$

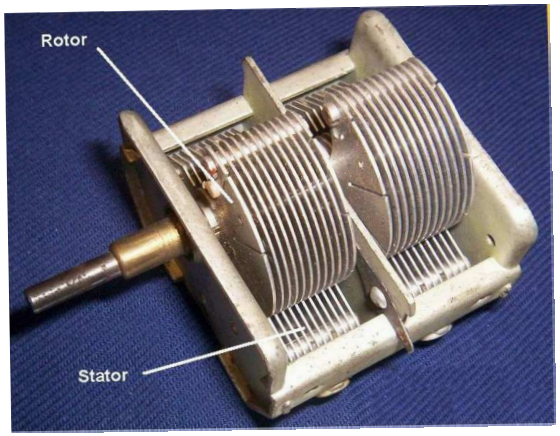
variation of length

R

variation of permeability (μ)



L



variation of surface

C

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

percussion

Definition:

1) **musical instruments that are hit:** the group of musical instruments that produce sound by being struck, including drums and cymbals, or the section of the orchestra playing such instruments

2) **act of detonating percussion cap:** the striking or detonating of a percussion cap in a firearm

3) **impact:** the impact of one object striking another, or the noise or ***shock*** ***created when two objects hit each other***

4) **medicine** **tapping of body:** examination of part of a patient's body by tapping with the fingers to assess the presence of fluid, the enlargement of organs, or the solidification of normally hollow parts

Wait a minute, is there anything **wrong** with the title of this presentation ???

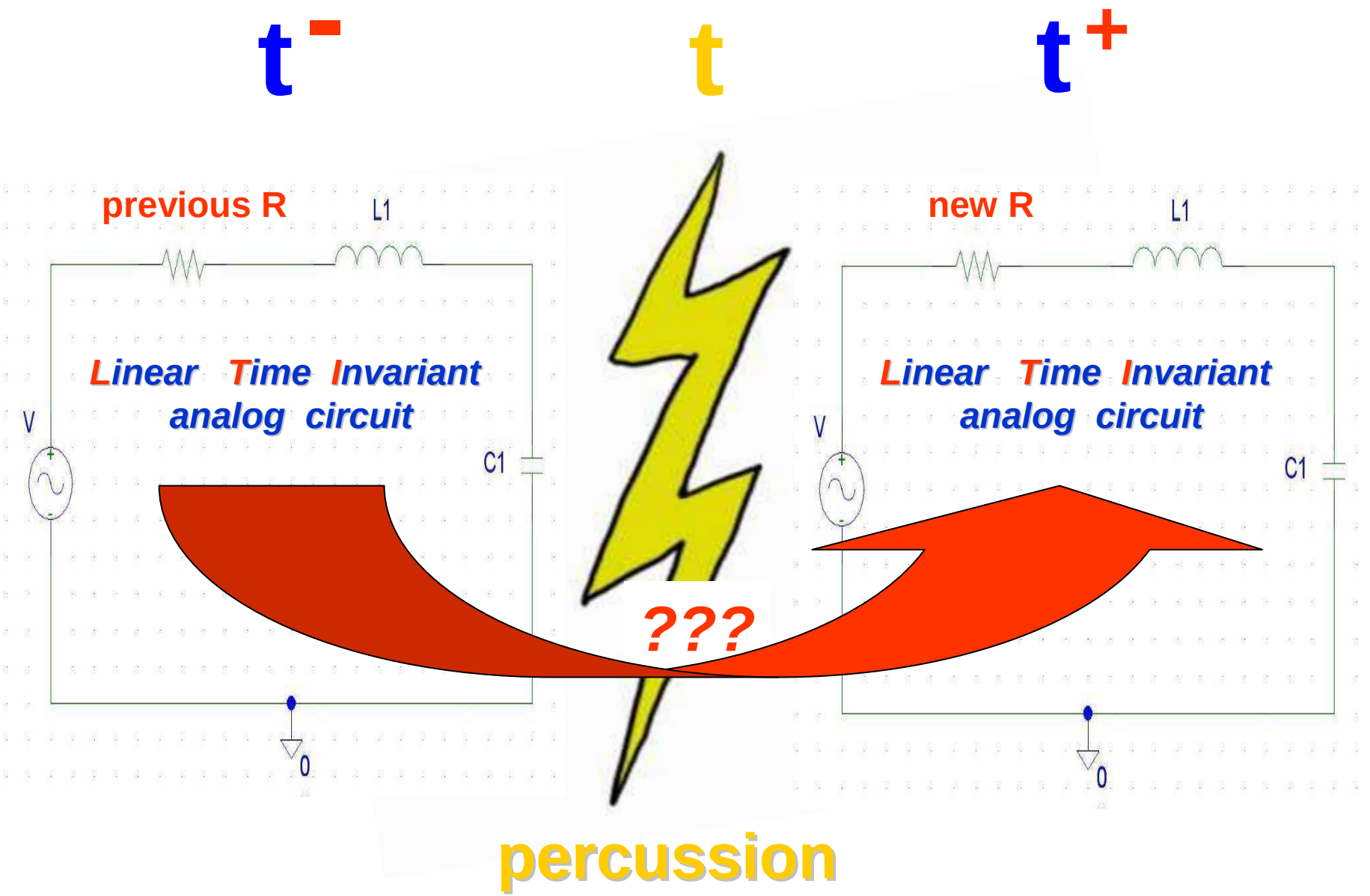
*Component PERCUSSION of Linear * Time Invariant * analog circuit*

* *obeys the laws of superposition*

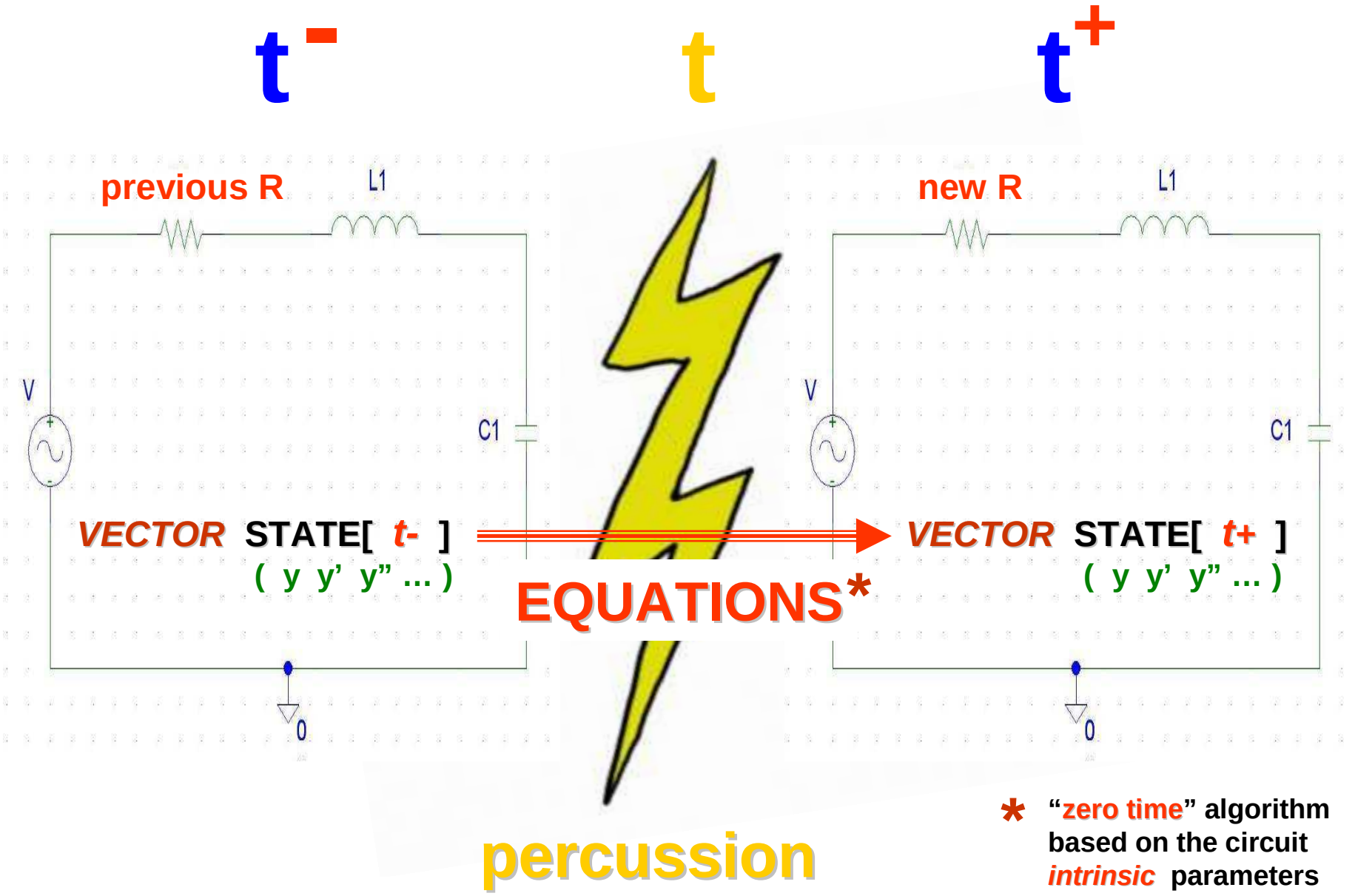
* *does not change its behavior over time*

(i.e. constant intrinsic parameters)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



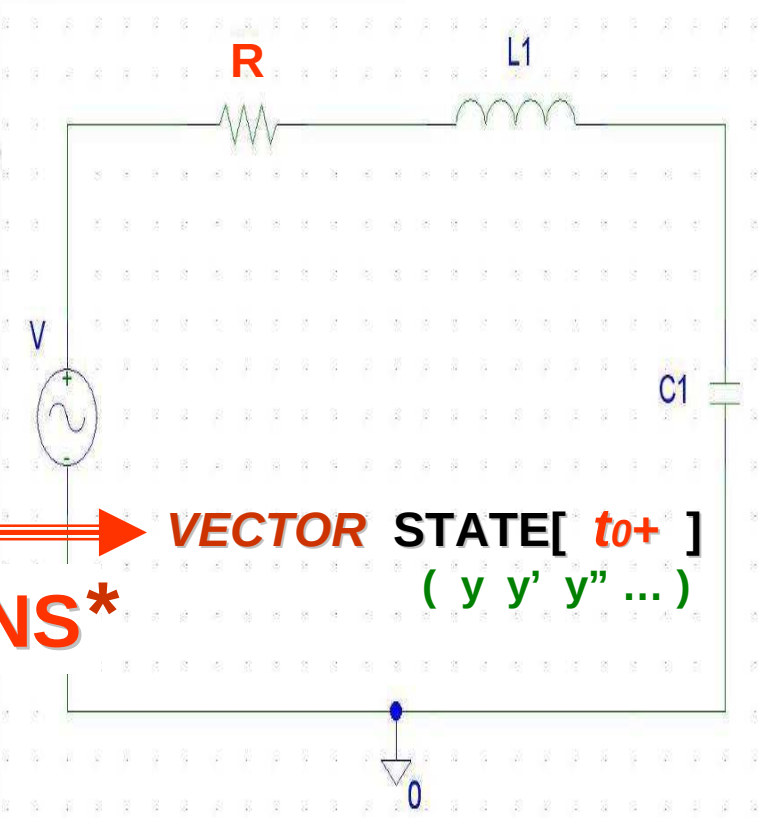
* "zero time" algorithm based on the circuit *intrinsic* parameters
not a convergence !

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Moreover,

the circuit *"initialization"*
is just a *particular case* of such
instantaneous state *change*

t_0 t_0^+



circuit
"*at rest*"

EQUATIONS*

VECTOR STATE[*to+*]
(y y' y" ...)

percussion

* "zero time" algorithm based on the circuit *intrinsic* parameters
not a convergence !

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

In summary, a ***PERCUSSION*** can be

⇨ a ***"sharp"*** **change of the** **INPUT** **(including *INITIALIZATION*)**

⇨ **an *"INSTANTANEOUS"* change of** ***R*** **,** ***L*** **or** ***C***

However,

the processing of INPUT and COMPONENT *PERCUSSIONS* being "*similar*"
but one modeling an *external shock*
and the other one an *internal structural change*

in this *Inverse Laplace Transform* *software package*

a "*sharp* " change of an INPUT is referenced as a *DISCONTINUITY*

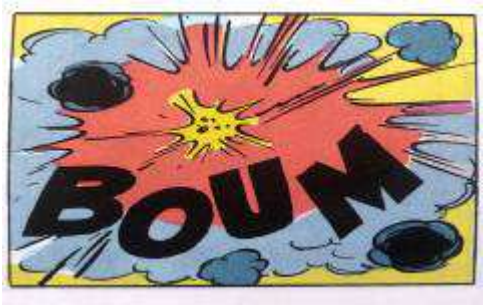
but a modification of *R*, *L* or *C* is effectively called a *PERCUSSION*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

In fact, we are playing with words

a ***discontinuity*** on the **INPUT** results in a ***percussion*** of the circuitry

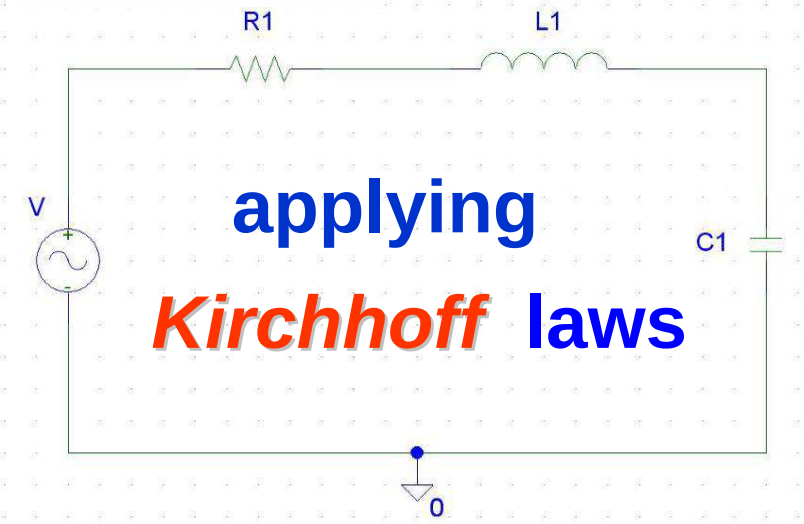
a ***percussion*** of a **COMPONENT** results in a ***discontinuity*** on the **OUTPUT**



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

MAXIMA

Symbolic
(Computer Algebra System)



applying
Kirchhoff laws

```
EQ  : [
        vr1  =   R1 * ir1          ,
        vl1  =  ZL1 * il1          ,
        vc1  =  ZC1 * ic1          ,
        ir1  =  il1                ,
        ir1  =  ic1                ,
        V    =  vr1 + vl1 + vc1
      ] ;
```

```
solve( eliminate( EQ , [ vl1, vc1, ir1, il1, ic1 ] ) , vr1 ) ;
solve( eliminate( EQ , [ vr1, vl1, vc1, ir1, ic1 ] ) , il1 ) ;
solve( eliminate( EQ , [ vr1, vl1, ir1, il1, ic1 ] ) , vc1 ) ;
```

$$[\ vr1 = \frac{R1\ V}{ZL1 + ZC1 + R1}\ ]$$

$$[\ il1 = \frac{V}{ZL1 + ZC1 + R1}\ ]$$

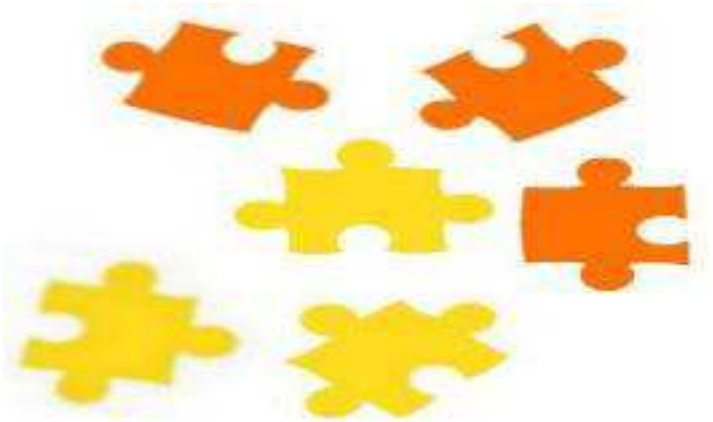
$$[\ vc1 = \frac{V\ ZC1}{ZL1 + ZC1 + R1}\ ]$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

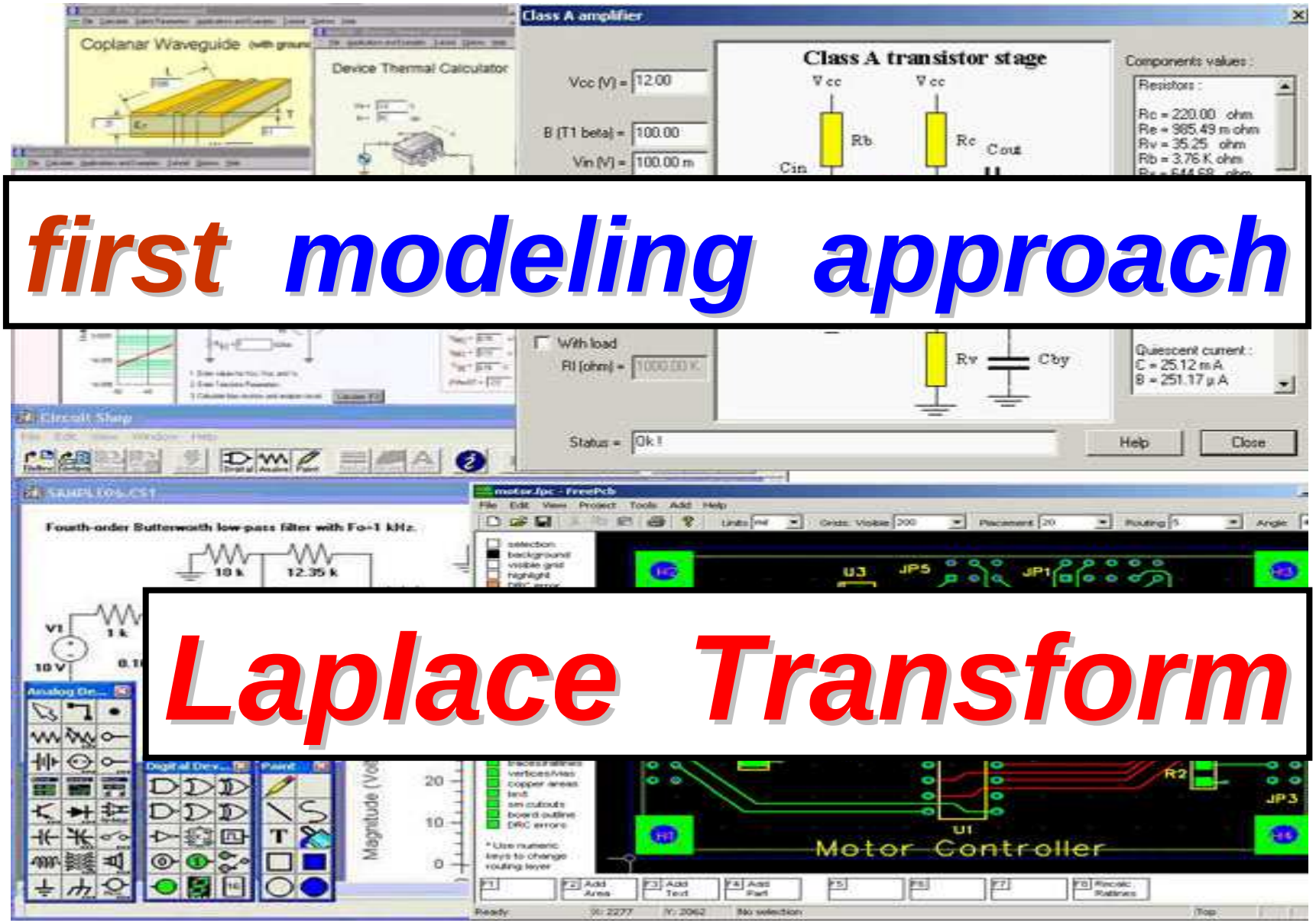
UNFORTUNATLY,

Multiple approaches exist ***to model*** *such kind of filters :*

- ***Laplace Transform***
- ***State Model***
- ***Differential equations***
- ***Differential equations*** by means of ***Distributions***



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



first modeling approach

Laplace Transform

LAPLACE transform

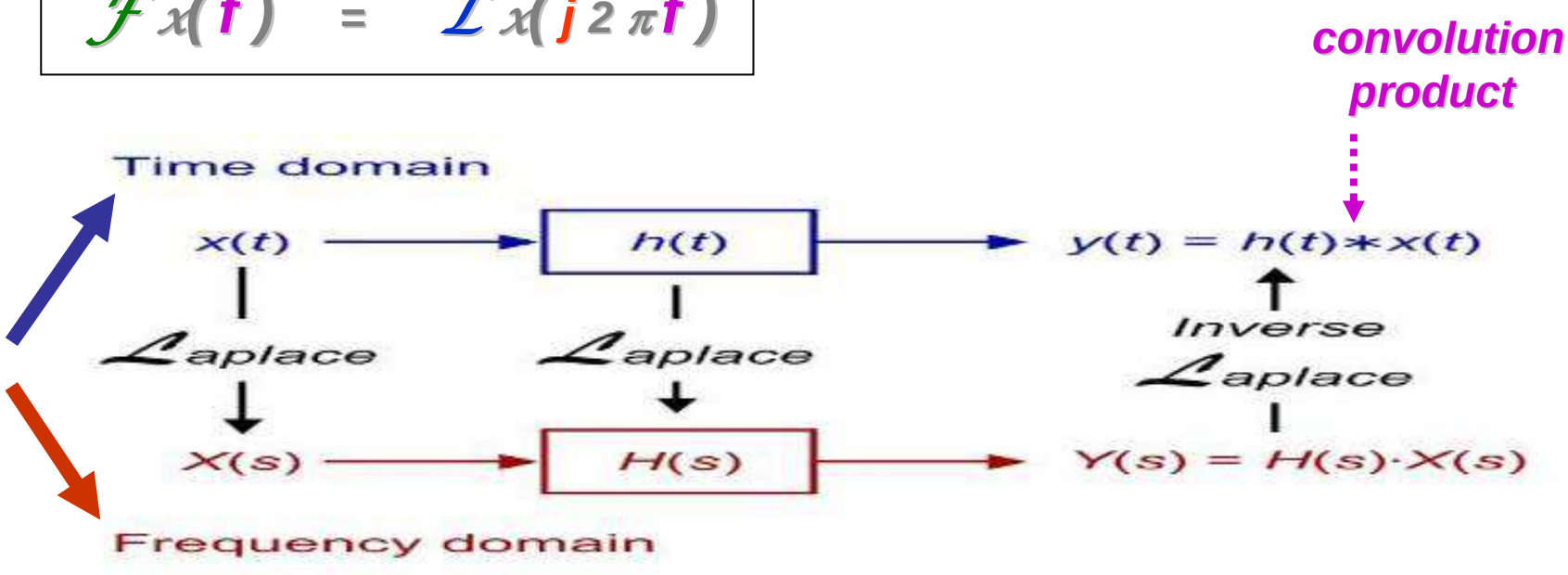
$$L\{f(t)\} = F(s) = \int_0^{\infty} f(t)\, e^{-st}\, dt$$

$$s = \alpha + j\omega$$

1749 - 1827

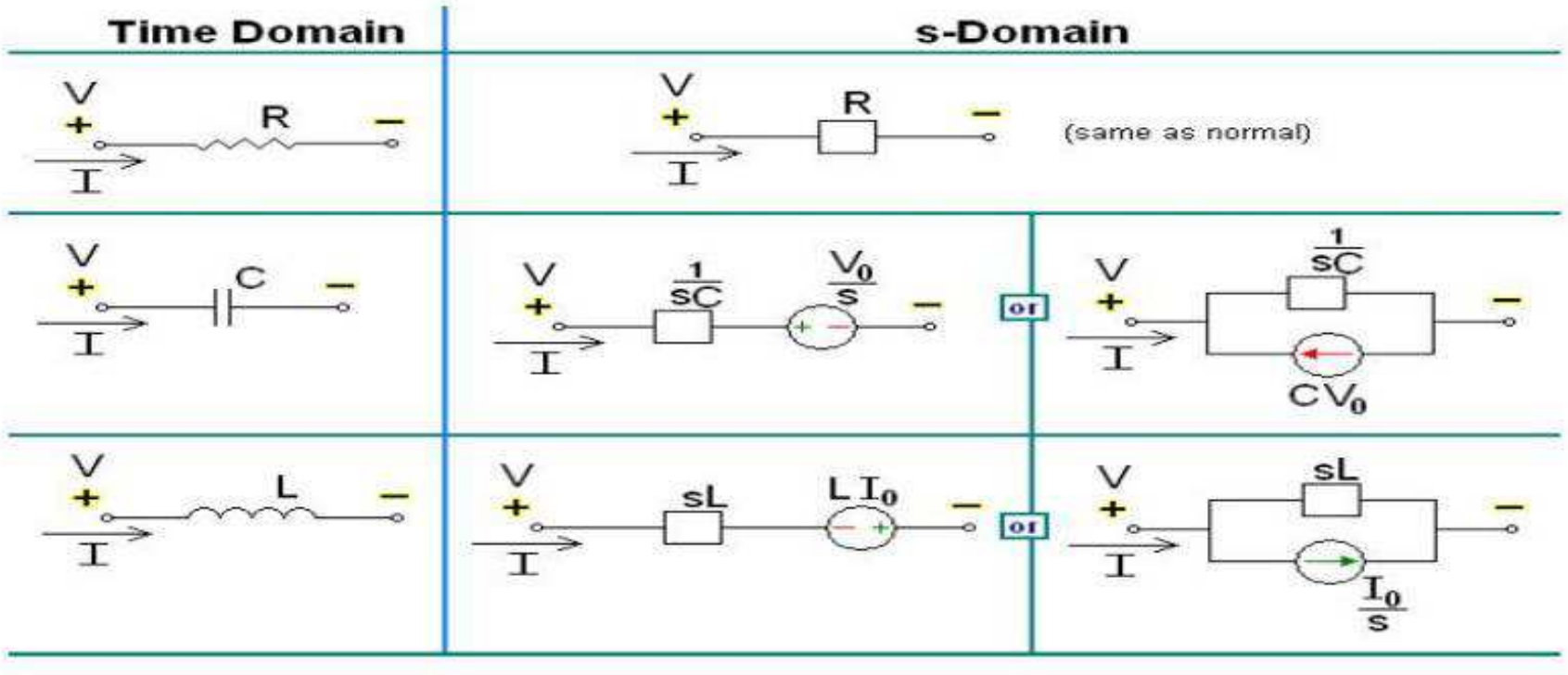
for causal / energy bounded signals

Fourier	=	Laplace
$\mathcal{F}\, x(f)$	=	$\mathcal{L}\, x(j\,2\,\pi f)$



Laplace

"s" (or "p") transform



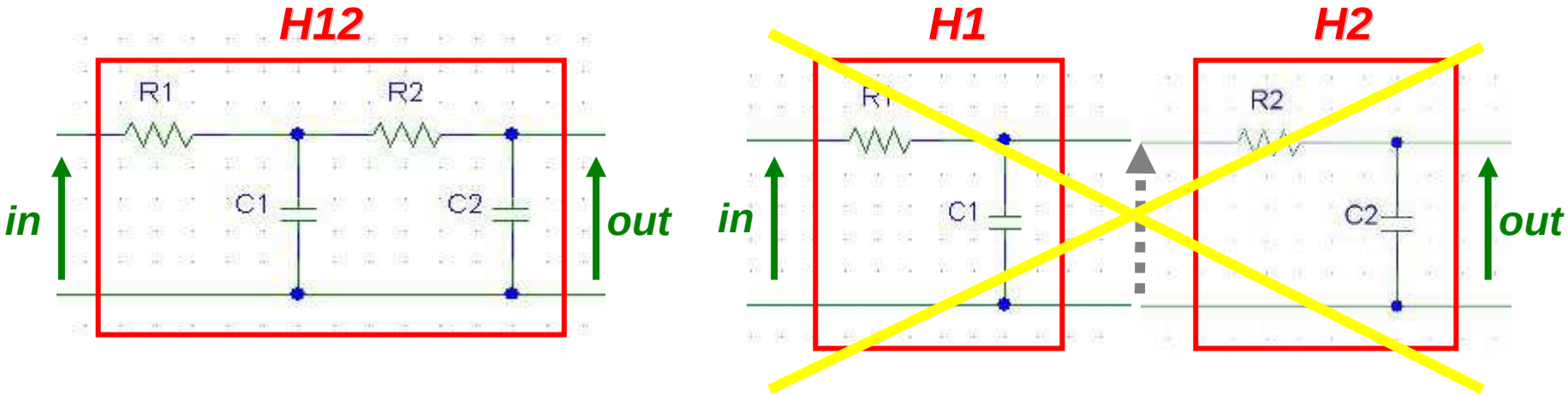
terminology

resistance	R
conductance	G

impedance	Z(s)	immittance
admittance	Y(s)	

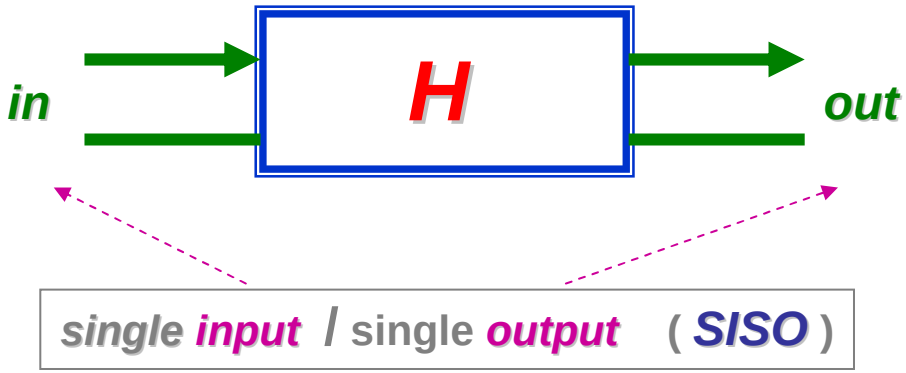
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

important NOTE



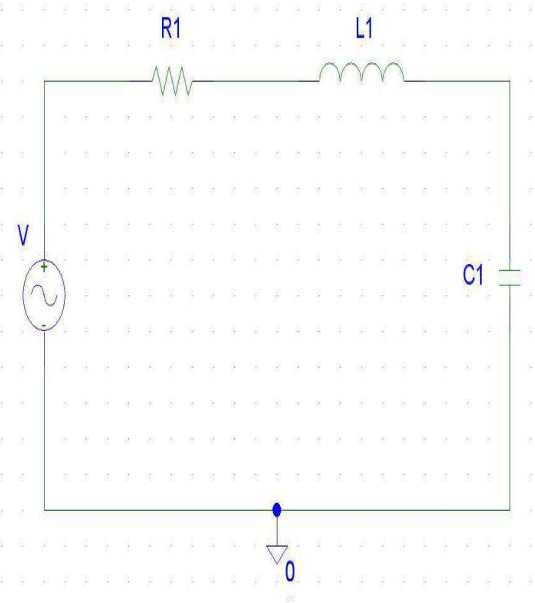
The concept of *Transfer Function* assumes that

the *input* and *output* are ideal (= *no interfering impedance*)



UNIT gain *OPAMP* may be used to safely cascade *FILTERS*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



$$\left[vr1 = \frac{R1\ V}{ZL1 + ZC1 + R1} \right]$$

$$\left[i11 = \frac{V}{ZL1 + ZC1 + R1} \right]$$

$$\left[vc1 = \frac{V\ ZC1}{ZL1 + ZC1 + R1} \right]$$

$$ZL1 = L1 * S$$

$$ZC1 = \frac{1}{C1 * S}$$

$$\frac{vr1}{V} = \frac{C1\ R1\ s}{C1\ L1\ s^2 + C1\ R1\ s + 1}$$

$$\frac{i11}{V} = \frac{C1\ s}{C1\ L1\ s^2 + C1\ R1\ s + 1}$$

$$\frac{vc1}{V} = \frac{1}{C1\ L1\ s^2 + C1\ R1\ s + 1}$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Those **3** "***transfer functions***" have the general form

(transmittance)

$$\frac{Y(s)}{X(s)} = \frac{s^2 N2 + s N1 + N0}{s^2 D2 + s D1 + D0}$$

An expression of the **Y(t)** "***time response***" can be determined

when $\frac{s^2 N2 + s N1 + N0}{s^2 D2 + s D1 + D0} X(s)$

(*or its decomposition*) can be retrieved from an

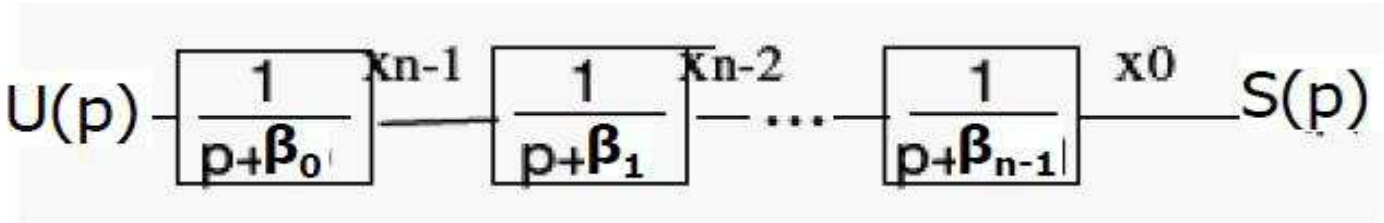
Inverse Laplace Transform Table

Obviously, this supposes that a ***"S" transform*** of the **INPUT** **X(t)** **is known !!!**

DECOMPOSITION

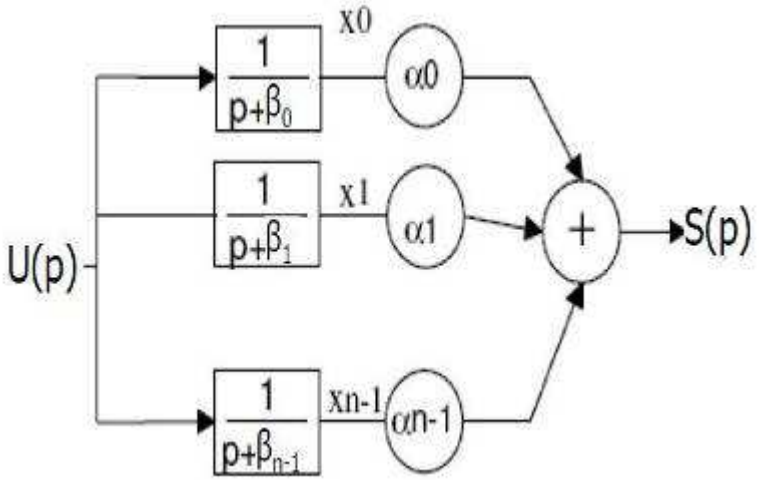
CASCADE

$$H(p) = \frac{1}{(p+\beta_0)(p+\beta_1)\ldots(p+\beta_{n-1})}$$



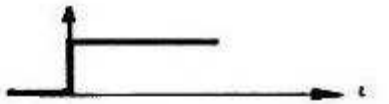
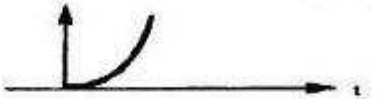
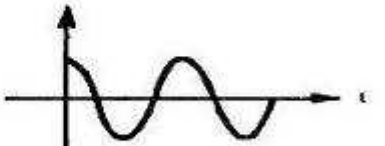
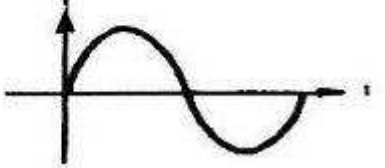
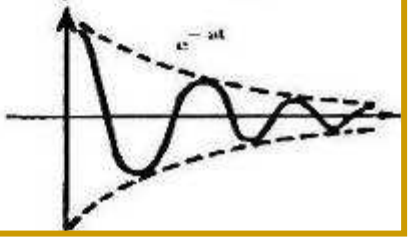
MODAL

$$H(p) = \frac{\alpha_0}{p+\beta_0} + \frac{\alpha_1}{p+\beta_1} + \ldots + \frac{\alpha_{n-1}}{p+\beta_{n-1}}$$

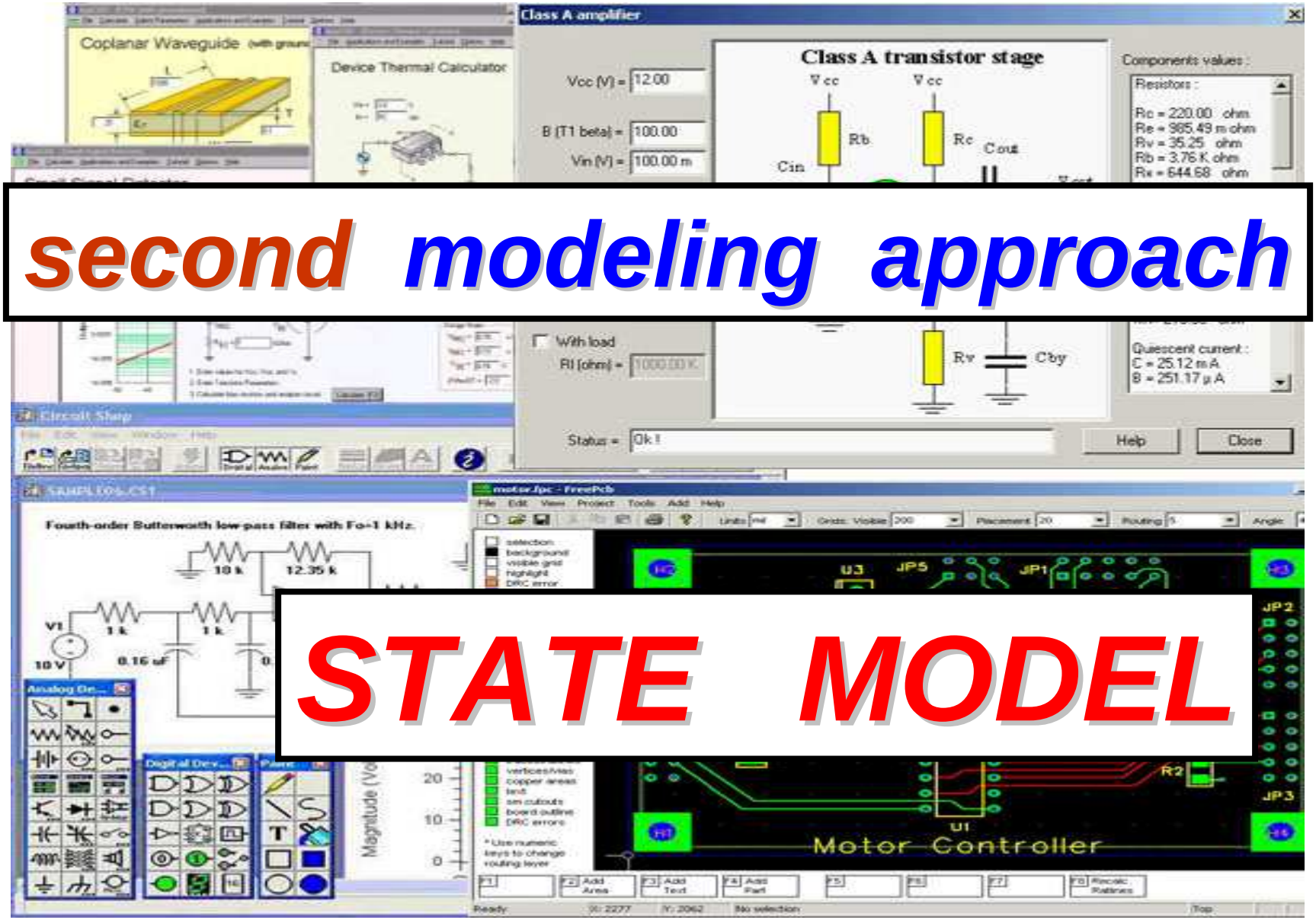


Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

TABLE EXTRACT

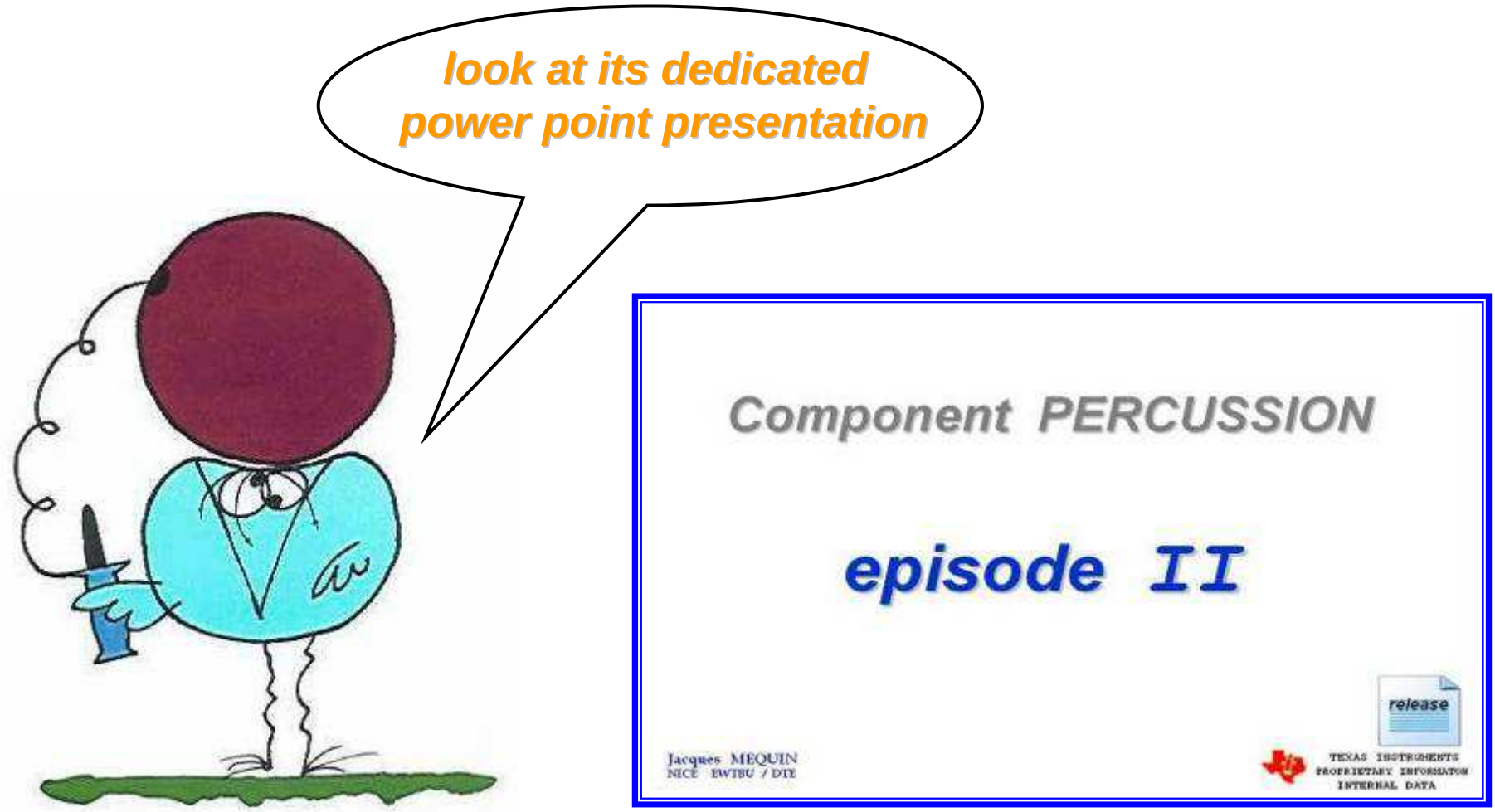
Unit		
$\frac{1}{p}$	HEAVISIDE 1	
Speed		
$\frac{1}{p^2}$	t	
Acceleration		
$\frac{1}{p^n}$	$\frac{t^{n-1}}{(n-1)!}$	
Dirac		
1	DIRAC at	
1 order		
$\frac{1}{p+a}$	e^{-at}	
2 order		
$\frac{1}{(p+a)^2}$	te^{-at}	
(n + 1) order		
$\frac{1}{(p+a)^{n+1}}$	$\frac{t^n e^{-at}}{n!}$	
Cosinus ...		
$\frac{p}{p^2+\omega^2}$	$\cos \omega t$	
$\frac{p}{p^2-\omega^2}$	$\text{ch}\ \omega t$	
Sinus ...		
$\frac{\omega}{p^2+\omega^2}$	$\sin \omega t$	
$\frac{\omega}{p^2-\omega^2}$	$\text{sh}\ \omega t$	
$\frac{p+a}{(p+a)^2+\omega^2}$	$e^{-at}\cos \omega t$	
$\frac{ap+\beta}{(p+a)^2+\omega^2}$	$Ae^{-at}\cos(\omega t+\varphi)$ $A=\frac{1}{\omega}\sqrt{a^2\omega^2+(\beta-a\alpha)^2}$ $\varphi=-\arctan(\beta-a\alpha)$	

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

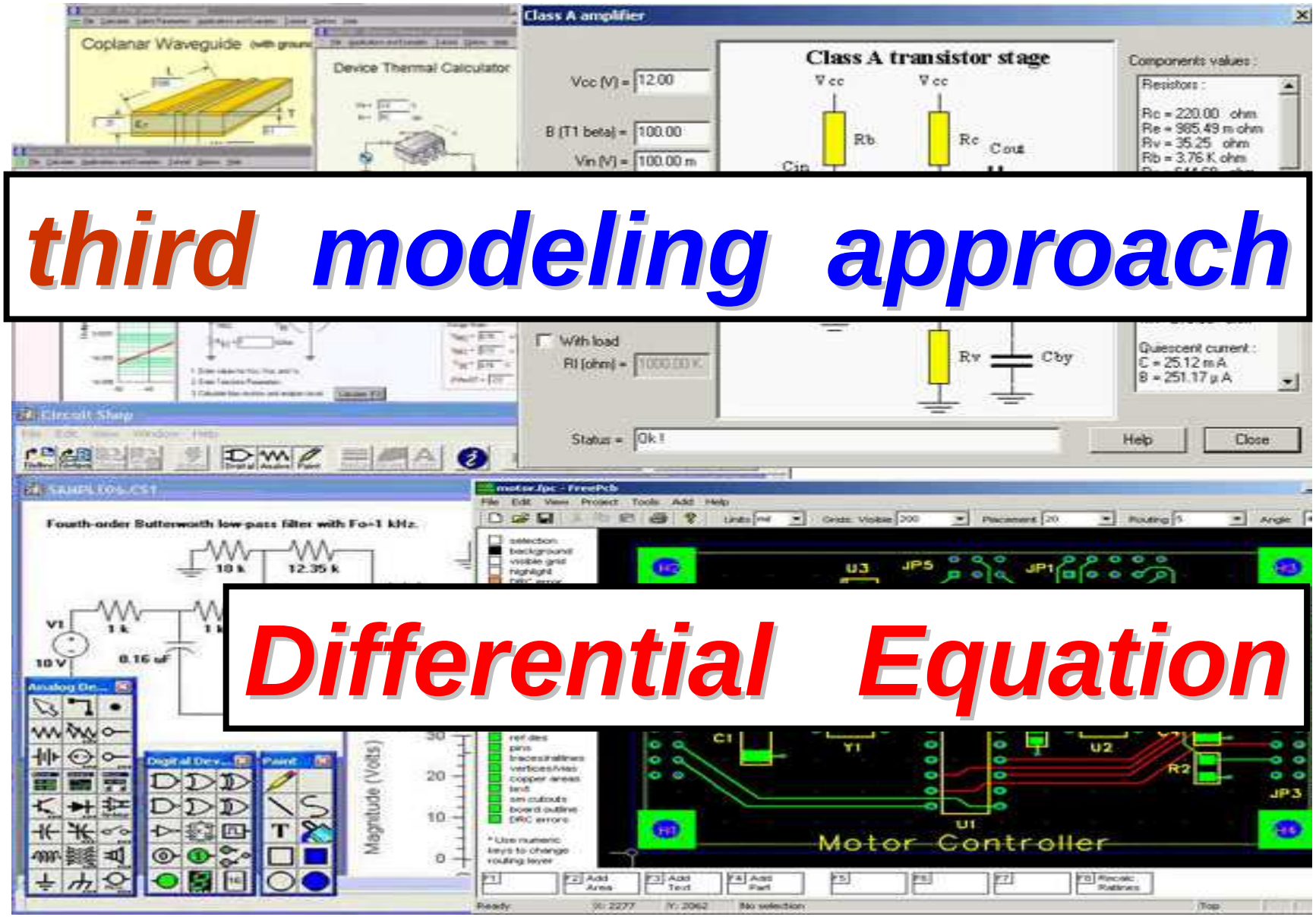


Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STATE MODEL



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



third modeling approach

Differential Equation

Differential Equation

A ***generic method*** to ***simulate*** the "***time domain***" response of the ***transfer function***

$$\frac{Y(s)}{X(s)} = \frac{s^2 N2 + s N1 + N0}{s^2 D2 + s D1 + D0}$$

is to use its **equivalent** ***time domain*** "***differential equation***"

$$\left(\frac{d^2}{dt^2}Y(t)\right)D2 + \left(\frac{d}{dt}Y(t)\right)D1 + Y(t)D0 = \left(\frac{d^2}{dt^2}X(t)\right)N2 + \left(\frac{d}{dt}X(t)\right)N1 + X(t)N0$$

This only requires that ***samples*** of the **INPUT** **X(t)** can be evaluated

LCCODE : *linear constant coefficient ordinary differential equation*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

They are numerous methods available to ***integrate*** ***Ordinary Differential Equation*** **(ODE)**

Among them, the ***RK4*** method ("*fourth order*" ***Runge-Kutta***) is both **accurate** (*error* $O(h^5)$) and **easy** to implement

Expecting for $y(t+h)$ **an expression such as**

$$y_{i+1} = y_i + (a_1 k_1 + a_2 k_2 + a_3 k_3 + a_4 k_4)\ (x_{i+1} - x_i)$$

assuming

$$\frac{dy}{dx} = f(x, y)$$

it is based on the determination of the value of the coefficients that "***match***" the ***Taylor series*** expansion at ***order 4***

$$y_{i+1} = y_i + \left.\frac{dy}{dx}\right|_{x_i, y_i}(x_{i+1} - x_i) + \left.\frac{1}{2!}\frac{d^2y}{dx^2}\right|_{x_i, y_i}(x_{i+1} - x_i)^2 + \left.\frac{1}{3!}\frac{d^3y}{dx^3}\right|_{x_i, y_i}(x_{i+1} - x_i)^3 + \left.\frac{1}{4!}\frac{d^4y}{dx^4}\right|_{x_i, y_i}(x_{i+1} - x_i)^4$$

due to supernumerary (*free*) variables, several possible set of coefficients ***k1***, …, ***k4*** are acceptable solutions

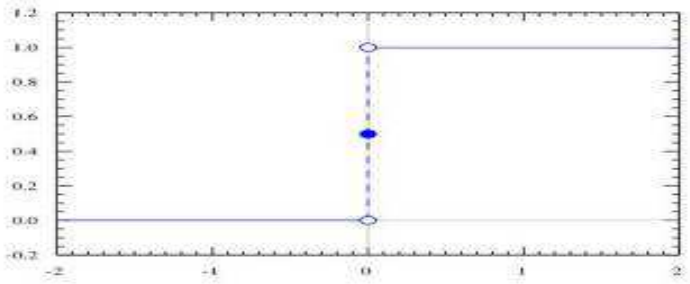
The following choice is generally very convenient

$$\begin{aligned}
k_1 &= \Delta t f(t_i, y_i) \\
k_2 &= \Delta t f(t_i + \frac{1}{2}\Delta t, y_i + \frac{1}{2}k_1) \\
k_3 &= \Delta t f(t_i + \frac{1}{2}\Delta t, y_i + \frac{1}{2}k_2) \\
k_4 &= \Delta t f(t_i + \Delta t, y_i + k_3) \\
y_{i+1} &= y_i + \frac{1}{6}(k_1 + k_2 + k_3 + k_4)
\end{aligned}$$

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

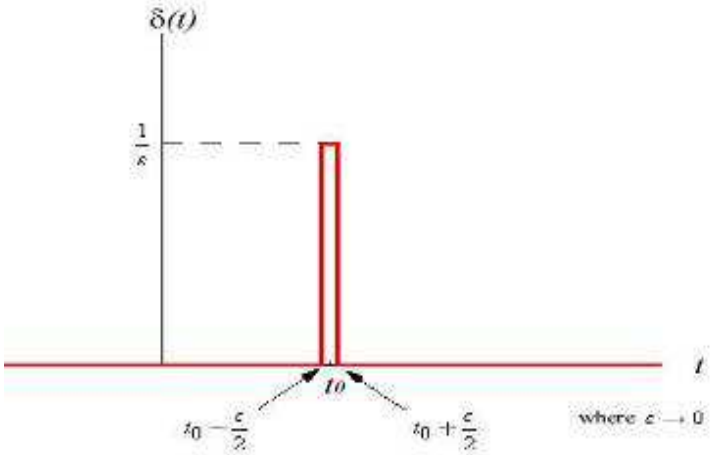
DISTRIBUTIONS $< T , \varphi >$

HEAVISIDE "generalized function"



$$H(x) = \begin{cases} 0 & x < 0 \\ 1 & x > 0 \end{cases}$$

DIRAC "generalized function"



$$\delta(x) \equiv \frac{dH}{dx} = \begin{cases} 0 & x < 0 \\ \infty & x = 0 \\ 0 & x > 0 \end{cases}$$

$$x\delta'(x) = -\delta(x)$$

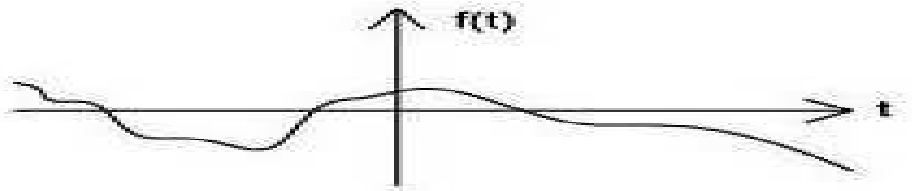
$$\int_{-\infty}^{\infty} \delta(x) f(x)\,dx = f(0) \qquad \int_{-\infty}^{\infty} a\delta(x)\,dx = a$$

Laurent Schwartz established it as a "***mesure***" (*a particular subspecies of his "**distributions**" theory*)

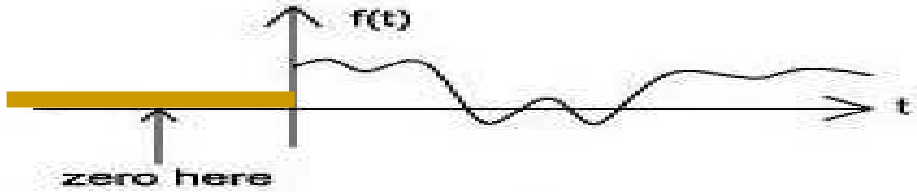
(***Fields medal** in 1950*)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

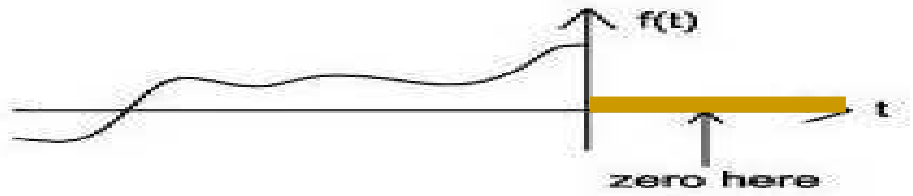
DEFINITIONS



noncausal signal



causal signal



anticausal signal

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

important relations

$$H'(t) = \delta(t)$$

$$f(t)\,\delta(t) = f(0)\,\delta(t)$$

$$\Rightarrow$$

$$H'(t-a) = \delta(t-a)$$

$$f(t)\,\delta(t-a) = f(a)\,\delta(t-a)$$

Let us consider a signal $y(t)$, *its "CAUSAL representation" is* $Y(t)$

$$Y(t) = y(t)\,H(t)$$

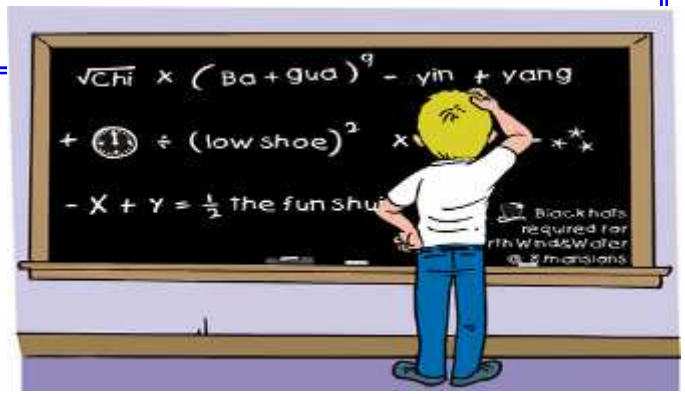
Note : $(u\,v)' = u'\,v + u\,v'$

$$Y'(t) = y'(t)\,H(t) + y(t)\,H'(t) = y'(t)\,H(t) + y(0)\,\delta(t)$$

the expression of next derivative will simplify as

$$Y''(t) = y''(t)\,H(t) + y'(0)\,\delta(t) + y(0)\,\delta'(t)$$

etc . . .



The next thing to do would be to show what is happening when we inject those definitions into the differential equations of the circuit

maybe another day . . .

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

STABILITY

Poles and stability of a system (of any order)

Look at the real part of each pole:

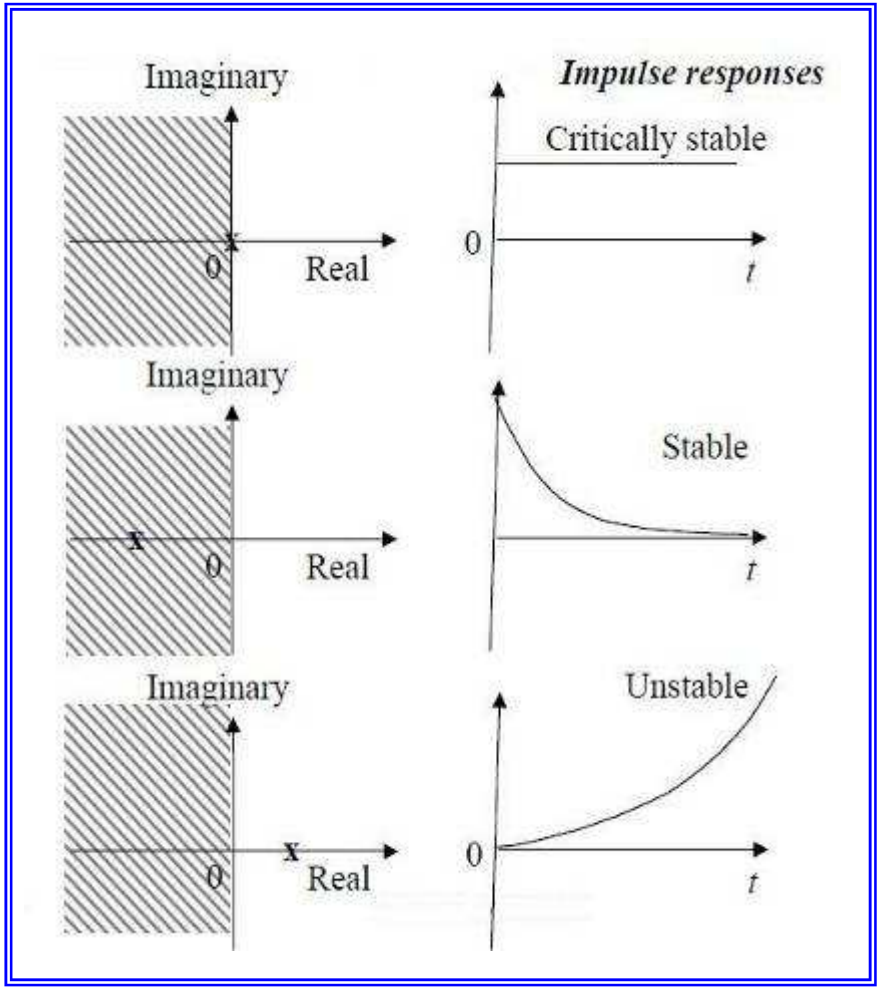
- If the **real part of ALL poles** is **negative** then all components of the system are **stable** and the **overall system is stable**
- If **ONE pole (or more)** has a **zero real part** then that component is critically/marginally stable and the **overall system is critically stable**
- If **ONE pole (or more)** has a **positive real part** then that component lead the **overall system to instability**

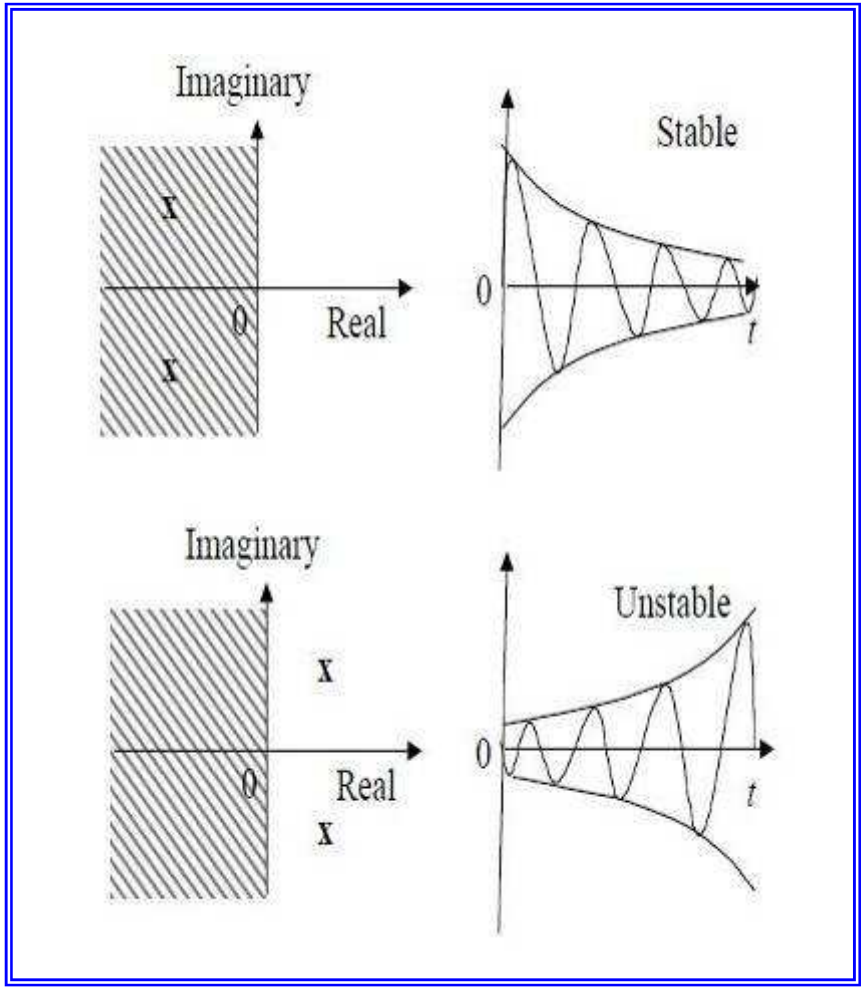
Look at the imaginary part for the presence of oscillations:

- If the imaginary part of a pole is zero then that component does not have any oscillatory contribution
- If the imaginary part is not zero then its value is the frequency of oscillation of the corresponding component of the system

The zeros of a system do not affect the stability

- The zeros affect the transient response of the system





For ***higher order*** systems that have many poles,

the poles that are ***further to the right*** of the ***s-plan*** will ***dominate*** the response

Numerically, there is another issue

pole order

$f = 100\ Mhz \Rightarrow\Rightarrow \omega = 2\pi f = 6.28\ 10^{8}$

$\omega^{2} = 3.94\ 10^{17}$

$\omega^{3} = 2.48\ 10^{26}$

. . .

Such very large ± *exponent* are also found in the Y(t) *high order derivatives*

(f ↑ ➔ step h ↓ *very small* and *powered* as h , h^{2} , h^{3} , . . .)

quantum slice of the period
must be SHANNON compliant to avoid aliasing

accordingly to TAYLOR expansion

NOTE: To minimize this issue the *ILT package* is coded *"internally"* with floating "*long double*" (*x87 standard : 80 bits / 19 digits / ≈ ±4932 exponent*)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

None of *3 methods* presented ***structurally handle*** percussions since, *by hypothesis*, the circuit is assumed ***Linear Time Invariant***

Nevertheless,

simulating the original circuit "up to" t^-

then, "starting at" t^+, simulating a ***modified circuit*** having as "***initial conditions***" the ***final values*** reached just ***before*** the ***percussion***

does play the trick !!!

This is OK !!!, but the making of such ***simulator*** changing the ***user defined circuit during the runtime*** is likely to be ***clumsy***

The aim of this presentation is precisely to claim that

an ***equational solution exist*** *to stick*, ***all along the simulation***, *with a more* ***seamless approach***

NAPA

15 years ago,

Yves Leduc who is the ***owner***, ***developer*** and ***enhancer*** of an ***"in-house" analog functional simulator*** (*named* ***NAPA***) has asked me to develop a ***plug-in*** performing the ***Inverse Laplace Transform***

Along the years, collecting new algorithms in the ***literature*** and on the **web** ***NAPA*** has succeed to become a unique collection of ***state of the art mathematical library*** feature (*sophisticated FFT windows, noise analysis, etc …*)

NAPA "ILT" plug-in hypothesis

For the *ILT package*, the output Y[t] of a ***transfer function*** is the same

if INPUT[t] is changing while INPUT[t - h] did also change value

or if INPUT[t - h] was *maintaining* a *steady value*

such heuristic* (*that neglects step "input variation"*) enables the *ILT package* to behave independently of the INPUT *derivatives

*(in line with the **STATE MODEL** activated by a **steady input** during a **step interval**)*

Therefore, such ILT processing can be viewed as " Markovian " ★

This is particularly well adapted to *non-vectorial* simulator such as *NAPA* that can *consequently* consider *any filter* as an *ordinary function* to which it has trivially to provide just a simple *"scalar" INPUT* "*sample*" at each call

★ *the state of such process at time t is independent of the history of the process before its previous state*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

$$\frac{Y(s)}{X(s)} = \frac{s^2 N2 + s N1 + N0}{s^2 D2 + s D1 + D0}$$

N2
N1
N0

$$\left(\frac{d^2}{dt^2}Y(t)\right)D2 + \left(\frac{d}{dt}Y(t)\right)D1 + Y(t)D0 = \left(\frac{d^2}{dt^2}X(t)\right)N2 + \left(\frac{d}{dt}X(t)\right)N1 + X(t)N0$$

$= 0$ $= 0$

X(t) ***causal*** ⇒ **X(t)** = $x(t)$ $H(t)$

developing this *differential equation* even with **input** $x''(t) = 0$ **and** $x'(t) = 0$

still leave the opportunity for the ***numerator coefficients*** ***N2*** **and** ***N1***

to contribute to the **Y(t)** **output** response equation

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

The ***"Markovian"*** approach of the ***ILT package*** is motivated by **another *tactical choice***

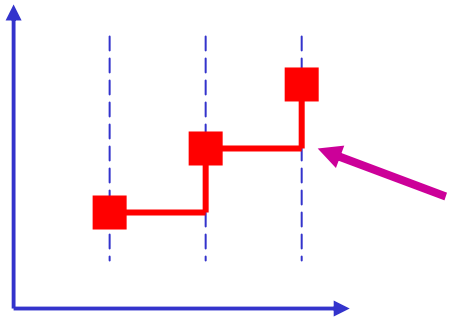
➡ **To assume** that the **INPUT** potentially may contain some ***sharp discontinuities***

(***Initialization*** **being one of them** ➡ ***ILT*** curves **at** " t_{0+} " ***match the theory***)

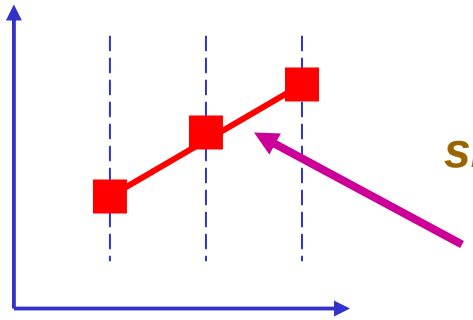
➡ and consequently, **not to assume**, estimate (nor "*interpolate*"), any ***smoothness*** of the **INPUT**

This would **cost** to store (to an *arbitrary chosen* ***depth***) a vector of the previous **INPUT**s and to ***interpolate*** them to compute ***associated derivatives***

This would also add some extra pain in case of ***STEP*** changes (***re-sampling***)



belief of a sharp INPUT based on discontinuity assumptions

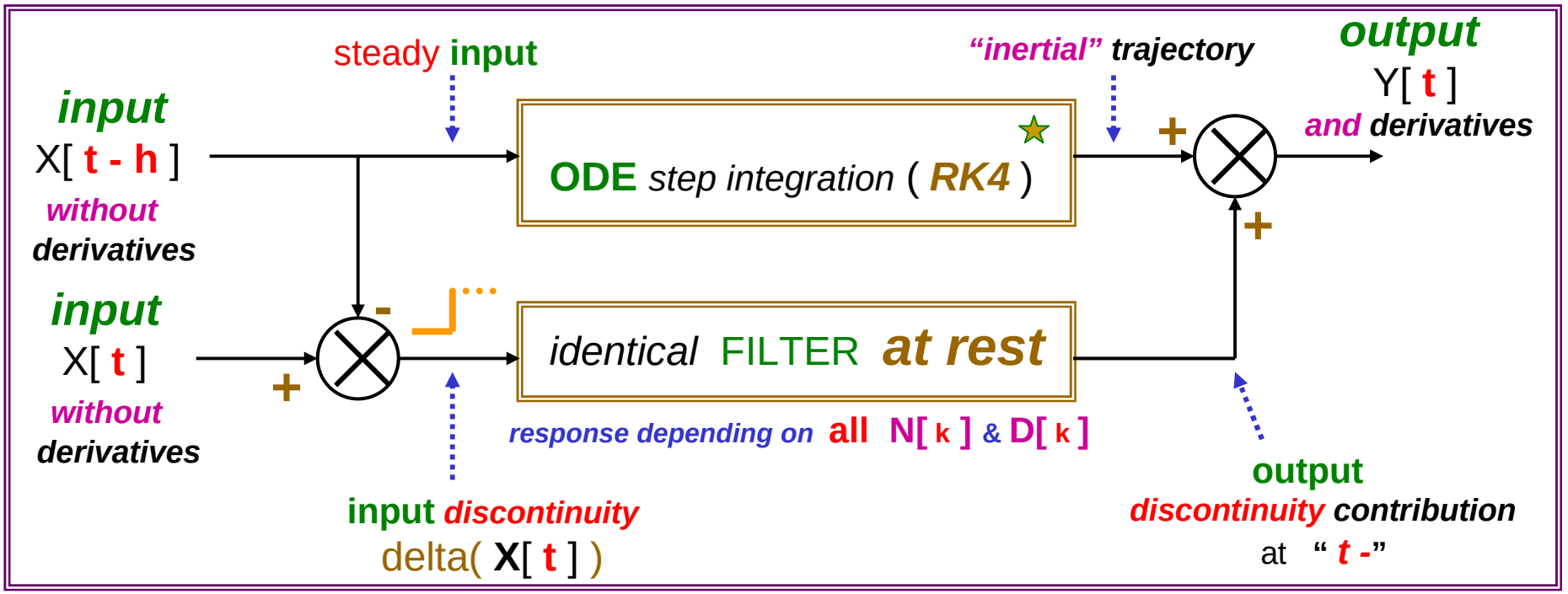


belief of a smooth INPUT based on derivatives

In other words, the ***expectation*** to accurately calculate the response to some ***sharp*** signal is somewhat tight to the **necessity** of a ***brut force*** processing of the **INPUT**

Quality of the *LINEAR* *response ???*

So, the ***output*** of the ***ILT package*** is calculated as



In summary, ***the hypothesis of sharp INPUT based on discontinuity assumptions enables accurate responses of "switched" type of circuits (in particular, the proper calculation of filter " impulse response ")***

However, ***for "smooth" input , the error is closely tight to the user STEP granularity***

★ *the ILT package uses RK4 for historical reasons, a simpler integration should be OK*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

ILT plug-in package functions :

ILT_init_LAPLACE( ) to define the *transfer function* NUMERATOR DENOMINATOR and the "*time step*"

ILT_change_h( ) enable '*on the fly*' change of the "*time step*"

ILT_step_LAPLACE( ) input(k + 1) == >> output(k + 1)

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

After all those fascinating (?) slides of introduction,

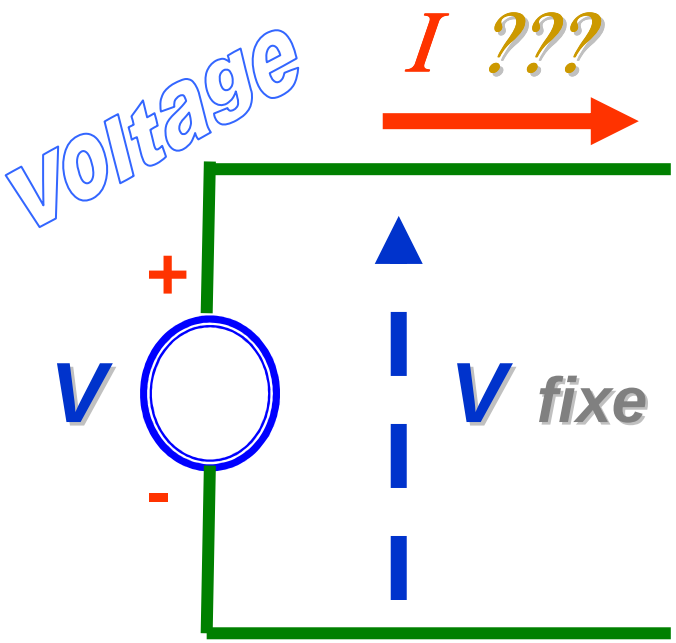
NOW, we can go back to our frightening question :

Component PERCUSSION of Linear Time Invariant analog circuit

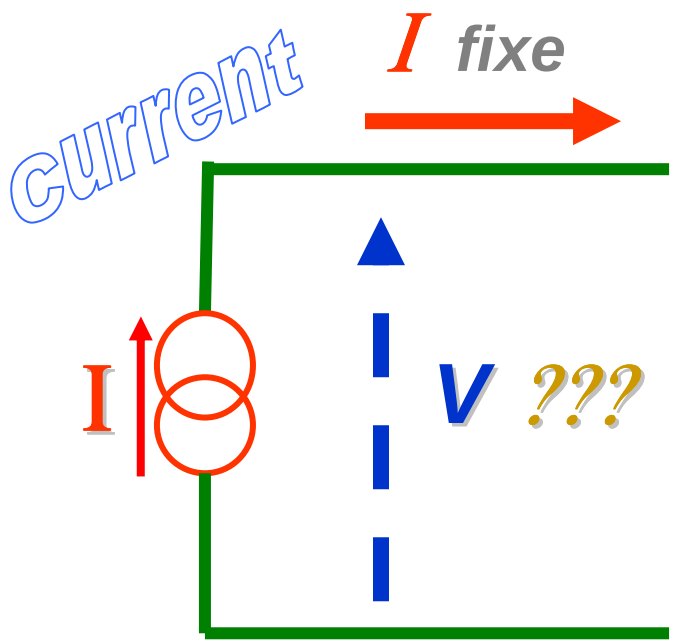
What is happening ???

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

"IDEAL" SOURCES



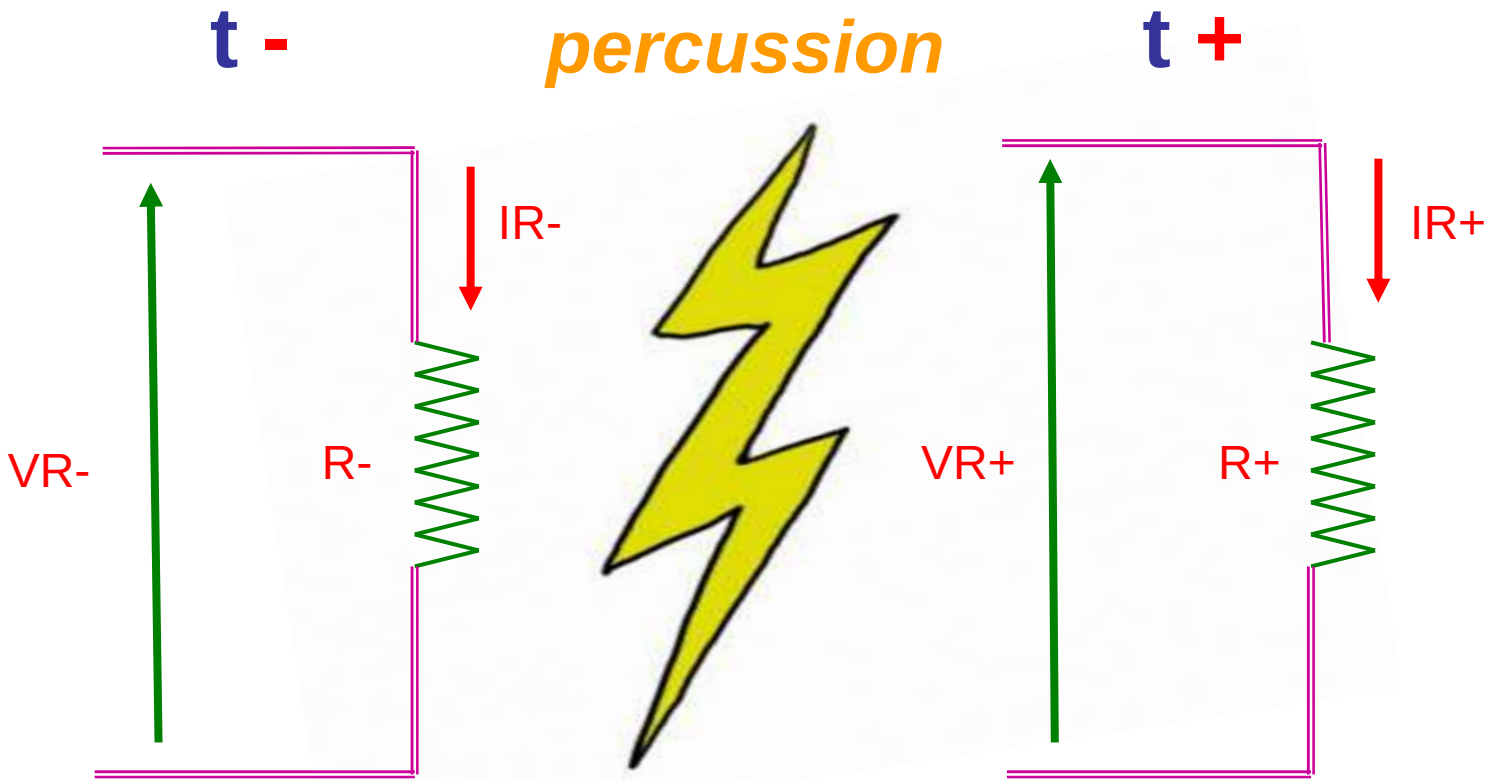
the **current**
will take whatever value
as a consequence of the
connected circuit



the **voltage**
will take whatever value
as a consequence of the
connected circuit

In both cases, the VOLTAGE and CURRENT are not "interdependent"

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com



At the precise instant of the ***percussion***,

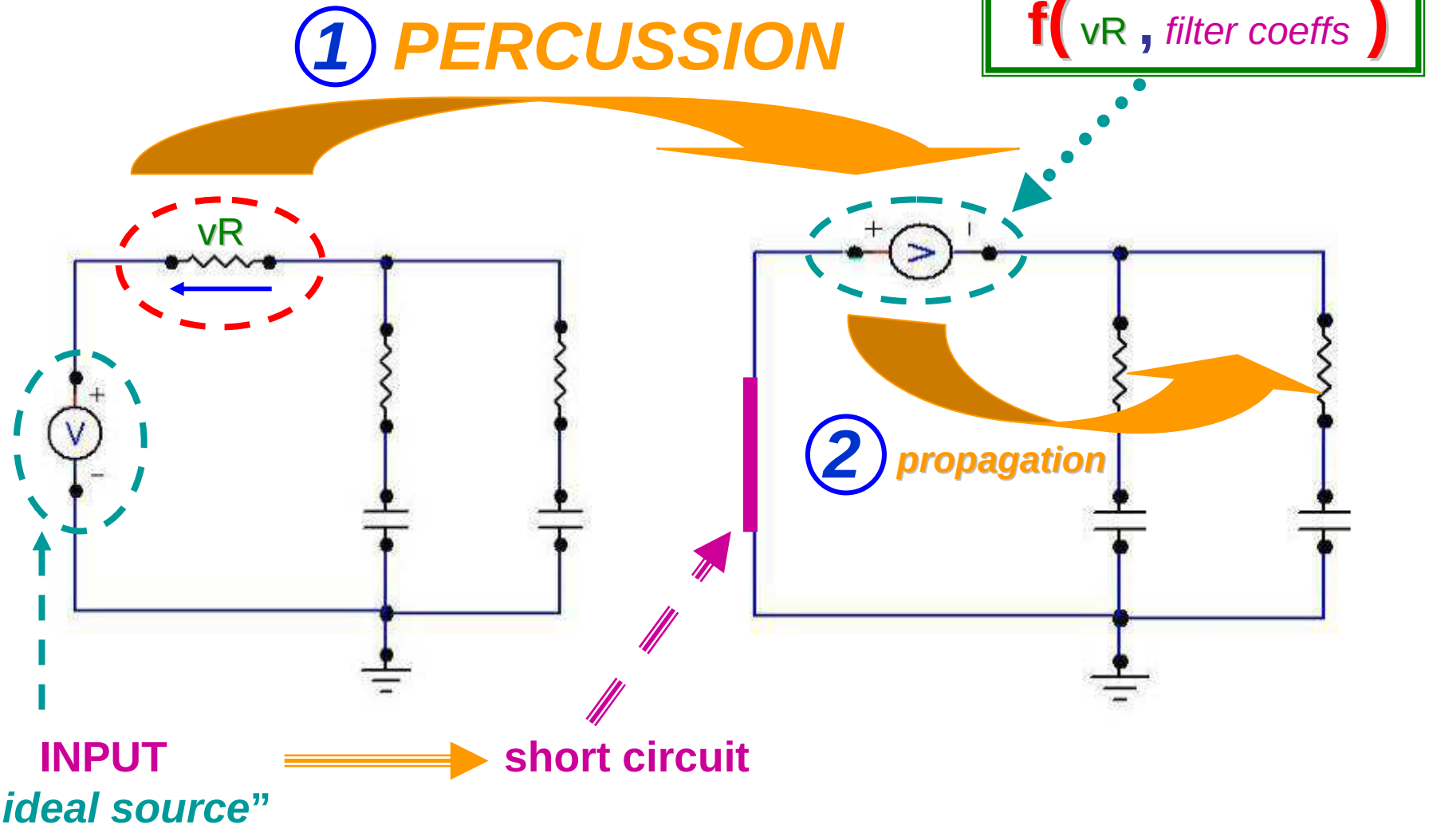
VRδ and ***IRδ*** *are paradoxically no longer* "***interdependent***"

since the "***percuted***" **component** can be viewed (during such ***null time***) as an "***ideal source***"

2 stages mystery . . .

"ideal source"
but for a null time !!!

f(vR , *filter coeffs*)



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

ILT PERCUSSION processing

FIRST stage : **"COMPONENT" PERCUSSION**

ILT plug-in package function: ILT_percussion( )

Please remember that the **ILT** package **initialization** has only defined the coefficients **Numerator** / **Denominator** of some **transfer functions**

for example, **the numerical values of the polynomial coefficients**

$N = [n_1, 0]$ and $D = [d_2, d_1, 1]$

$$\frac{vr1}{V} = \frac{C1\,R1\,s}{C1\,L1\,s^2 + C1\,R1\,s + 1}$$

Can we just make a call to a function with an updated set of polynomial coefficients (computed, for example, using a new value of "R1") **without passing**★ **the information that a R, a L, or a C has changed ???**

★ This is a very ambitious **assumption** regarding the assumed impact of the coefficients on the behavior of the **targeted differential equation variables**, demonstration is beyond this presentation

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

Yes !!!, *equations* can be found using *"DISTRIBUTION" formalism*

(as expected, they predominantly depend on the old & new denominator coefficients)

but . . .

it is OK to "*percute the Transfer Function*" representative of the *voltage*, or the *current*, across a RESISTOR

for a CAPACITANCE value change, it is mandatory to "*percute*" the *voltage*
since the *current* and the charge *Q* will remain unchanged by the *percussion*
It is the *voltage* that will *evolve*, according to the new *C*, to maintain $Q = CV$

for an INDUCTANCE value change, it is mandatory to "*percute*" the *current*
since the *voltage* and the *flux* ϕ will remain unchanged by the *percussion*
It is the *current* that will *evolve*, according to the new *L*, to maintain $\phi = L\ I$

If several components have *simultaneously* to change their values
they must do it through *several consecutive (2 stages) percussions*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

VALID SIMULATIONS

	never percuted		sometimes percuted	
R	$V_R(s) = H(s)\ input(s)$	$I_R(s) = H(s)\ input(s)$	$V_R(s) = H(s)\ input(s)$	$I_R(s) = H(s)\ input(s)$
L	$V_L(s) = H(s)\ input(s)$	$I_L(s) = H(s)\ input(s)$	~~$V_L(s) = H(s)\ input(s)$~~	$I_L(s) = H(s)\ input(s)$
C	$V_C(s) = H(s)\ input(s)$	$I_C(s) = H(s)\ input(s)$	$V_C(s) = H(s)\ input(s)$	~~$I_C(s) = H(s)\ input(s)$~~

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

ILT PERCUSSION processing

SECOND stage : **PERCUSSION "PROPAGATION"**

ILT plug-in package function: *ILT_export_percussion( )*

Now that an updated value of the voltage (or current) across the "percuted" component has been computed, we must update the voltage (or current) across any other output for which the user has defined, at initialization, an observable transfer function

That is, for our RLC example, iL1 & vC1

$$\frac{i11}{V} = \frac{C1\,s}{C1\,L1\,s^2 + C1\,R1\,s + 1}$$

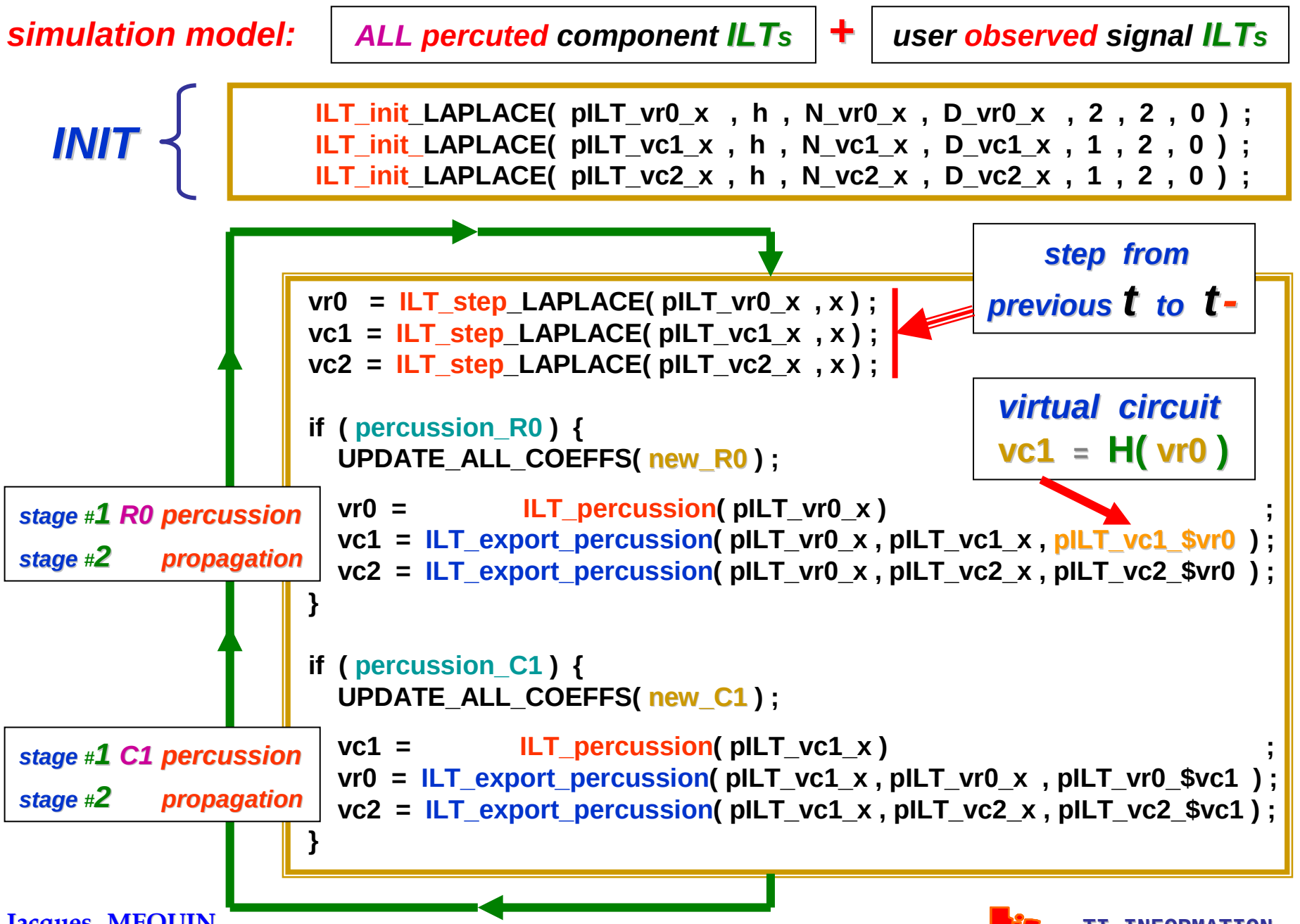
$$\frac{vc1}{V} = \frac{1}{C1\,L1\,s^2 + C1\,R1\,s + 1}$$

HOW TO DO IT ???

The "percuted" component ***will act*** *(but for a null time)*
as an ideal INPUT source within a ***virtual circuit*** *having*
a short circuit in place of the regular INPUT !!!

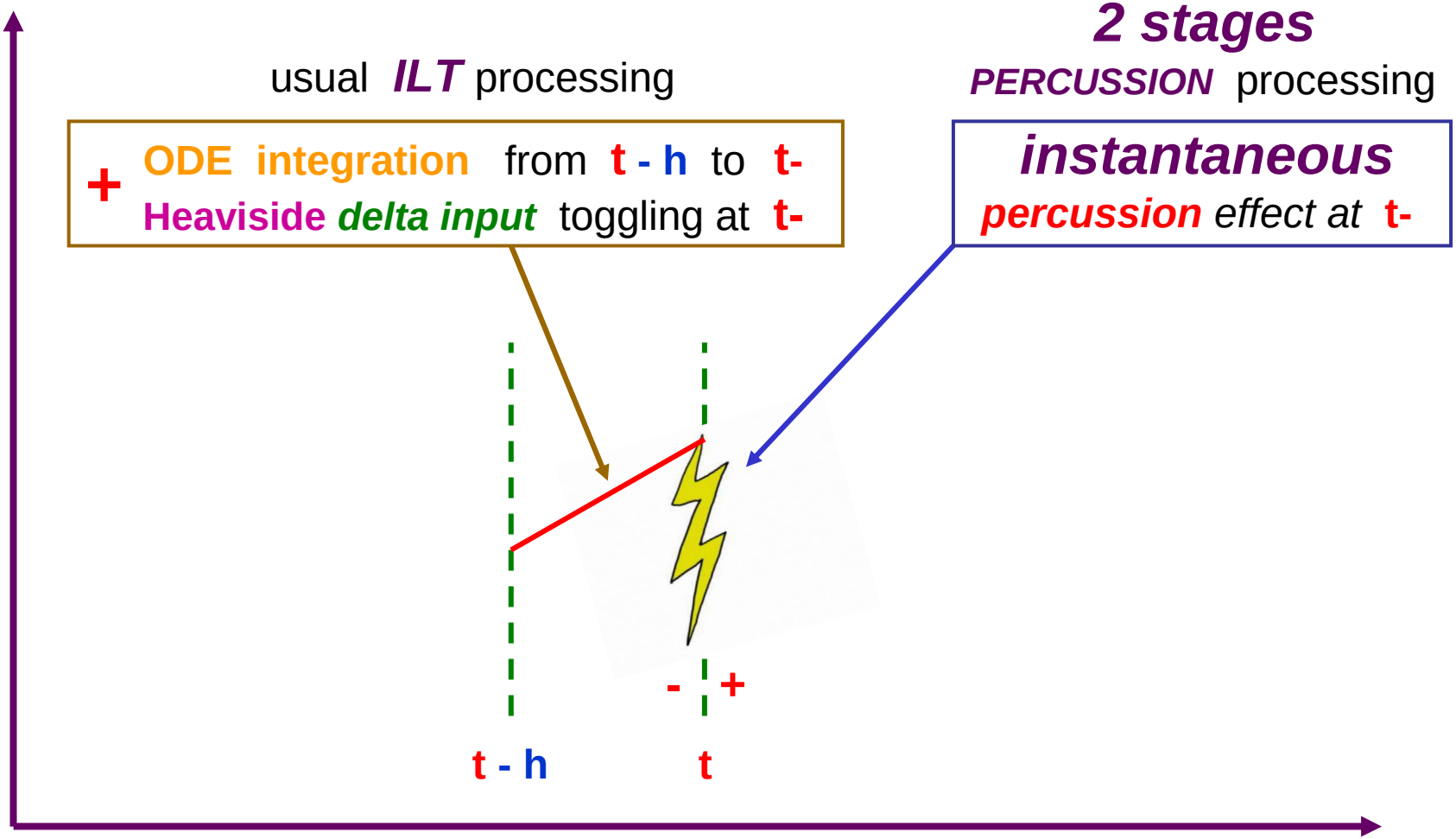
So, we have to predefine, for those ***OUTPUTs*** requiring a ***side effect*** of the ***main percussion***, a set of ***null execution time (!!!)*** ***transfer functions*** modeling those ***OUTPUTs*** in the ***virtual circuit***

equations can be found using *"DISTRIBUTION" formalism*
to ***superpose*** to their current ***OUTPUT*** value at " t " *the effect*
of the ***main discontinuity*** *propagated through the* ***virtual circuit***



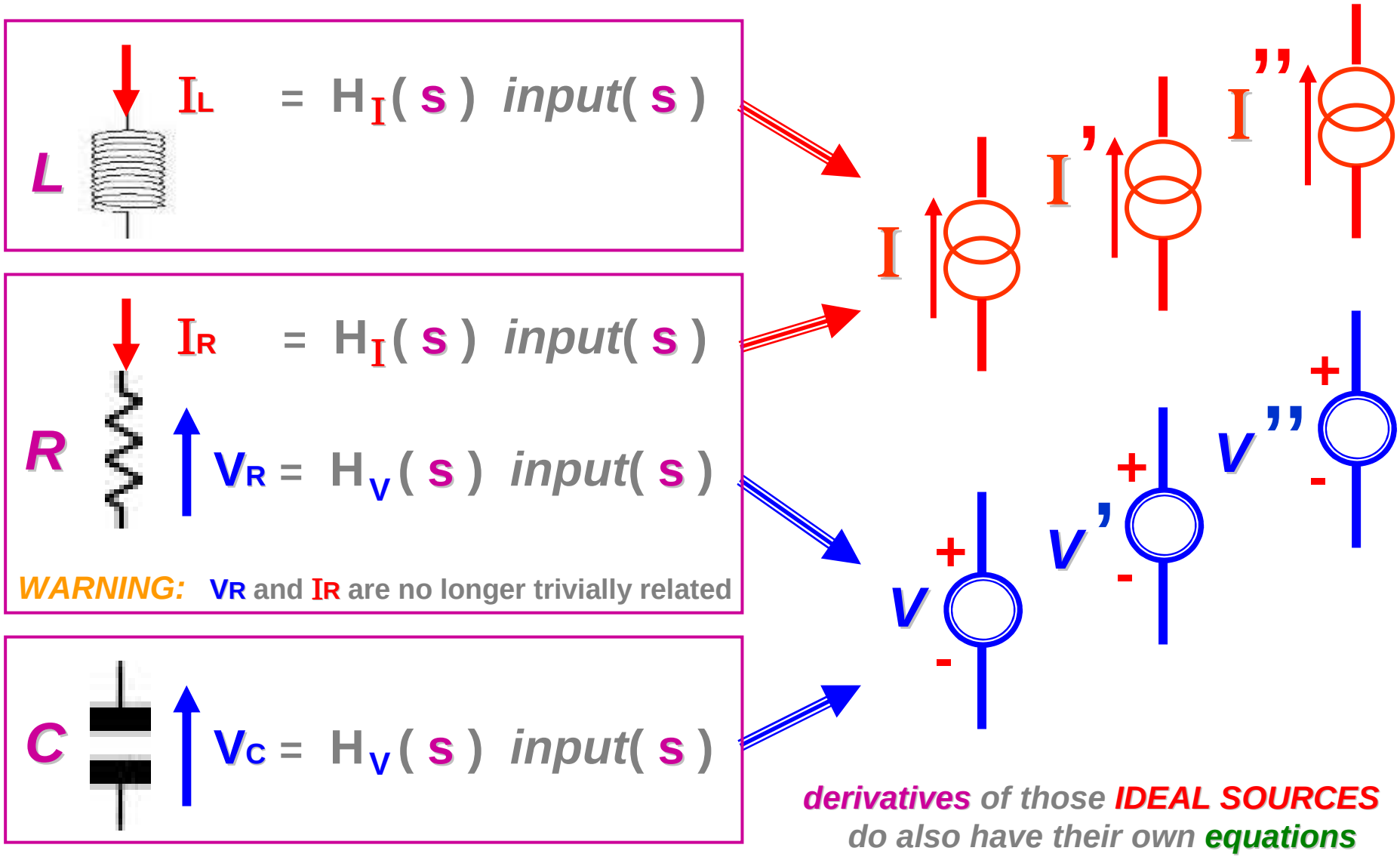
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

ILT PERCUSSION processing summary



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

COMPONENT PERCUSSION



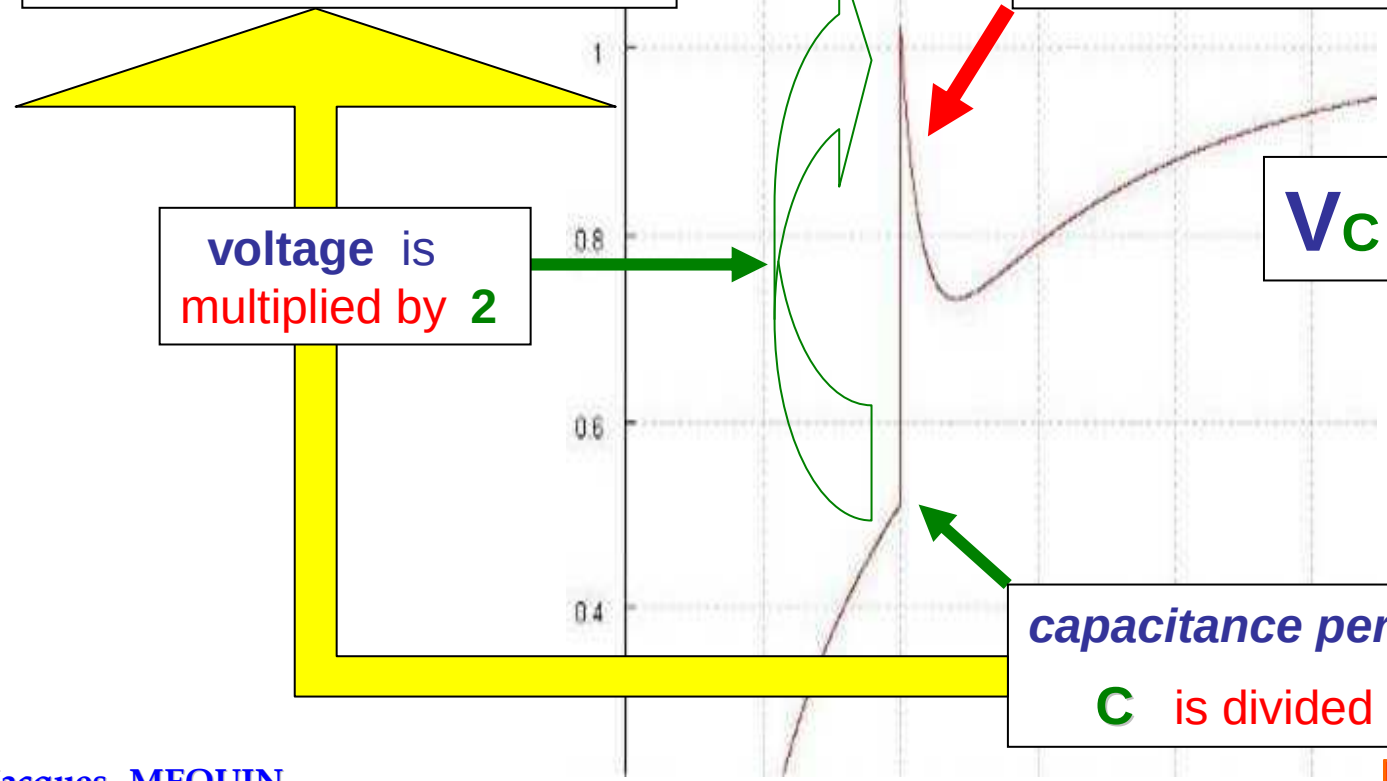
derivatives of those *IDEAL SOURCES*
do also have their own *equations*

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

from now on the curve becomes ***the resultant*** of ***a new*** ***free-mode + forced-mode***

at the precise instant
of the percussion
there is nothing very
impressive in the fact that
the equations predict
the doubling of the voltage

but the equations are priceless
in predicting how the circuit
will behave immediately after
the percussion due to
the symbolic expressions
of the *derivatives*



voltage is
multiplied by **2**

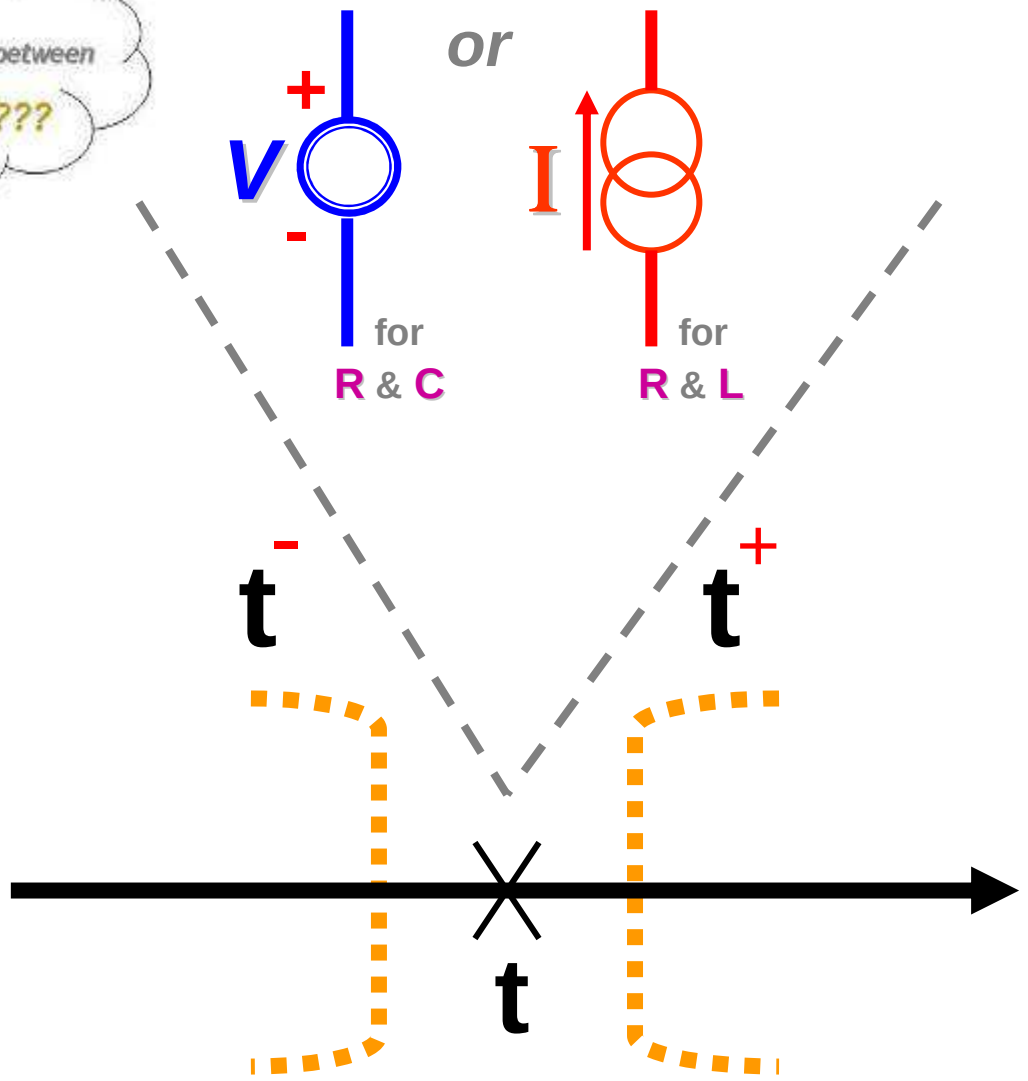
$V_C = f(\textit{input})$

capacitance percussion

C is divided by **2**

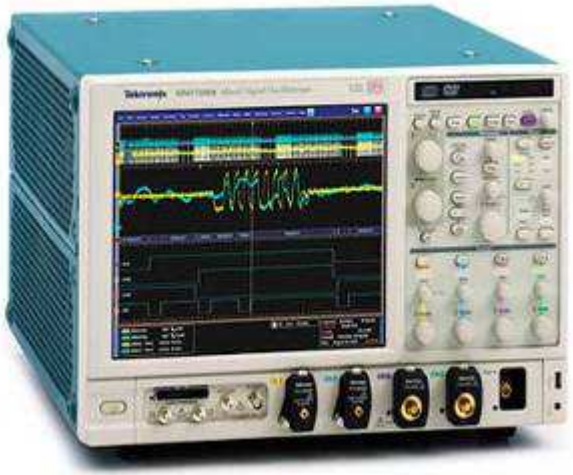
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

percuted component



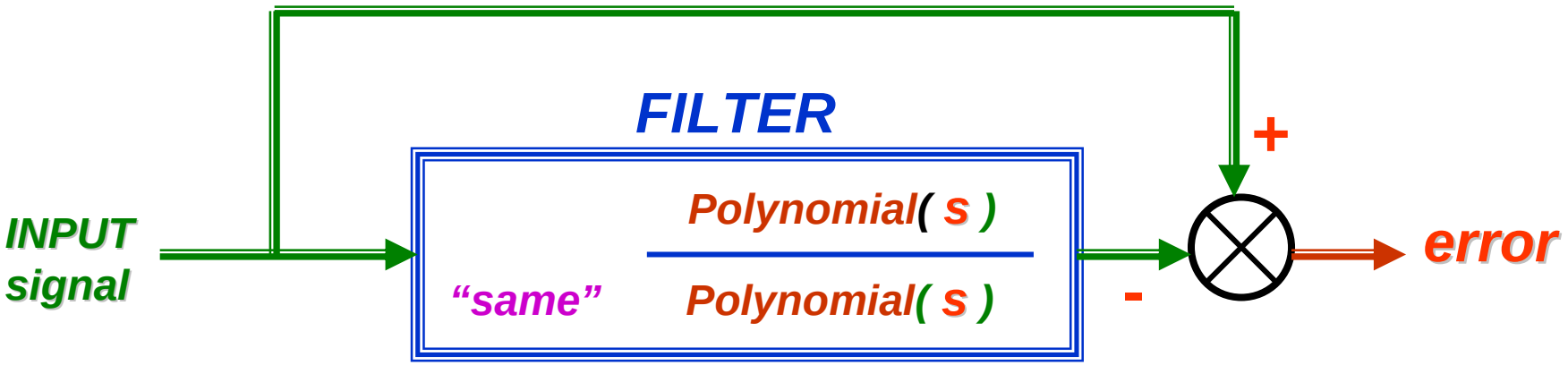
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

TEST



Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

“ WIRE ” test



Assuming that the *FILTER* is *stable*, this is expected to behave as an *ideal “wire”*

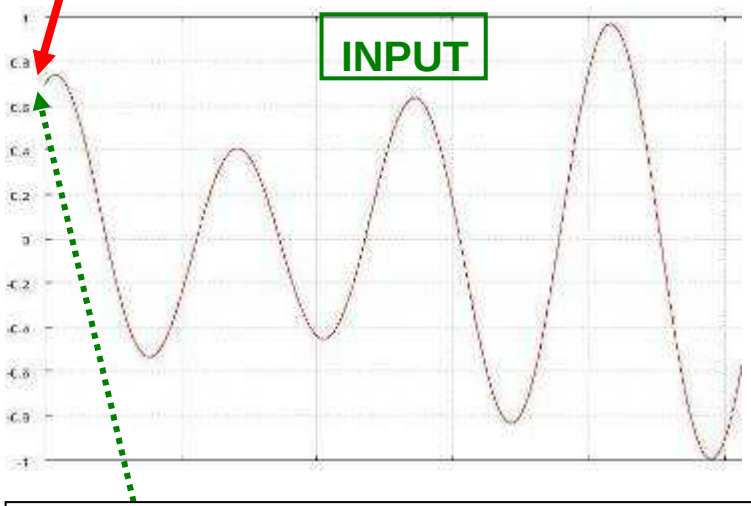
" WIRE " test

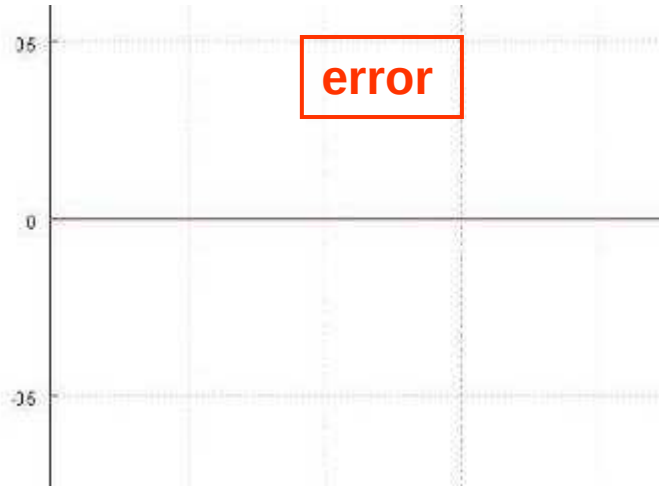
this filter is **"*stable*"** by construction

ka * *cos*(wa * t) + kb * *sin*(wb * t)

$$\frac{(s+1)(s+2)(s+3)(s+4)(s+5)}{(s+1)(s+2)(s+3)(s+4)(s+5)}$$

output(0) will only depends on
this first input sample
and the ***filter coefficients***

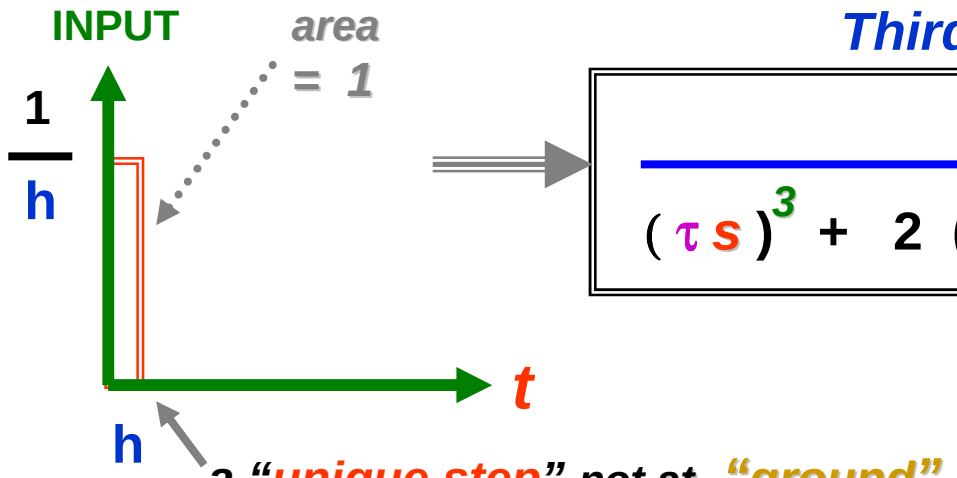




to increase ***stress*** this **activation** does not start at **0**

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

"IMPULSIONAL" response

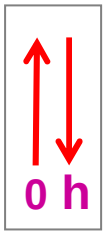


a "unique step" not at "ground"

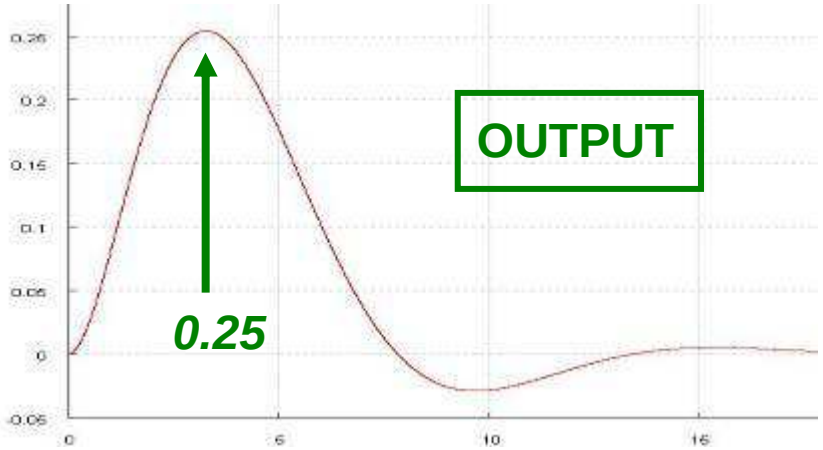
Third order Butterworth

$$\frac{1}{(\tau s)^3 + 2(\tau s)^2 + 2(\tau s) + 1}$$

only 2 discontinuity error points



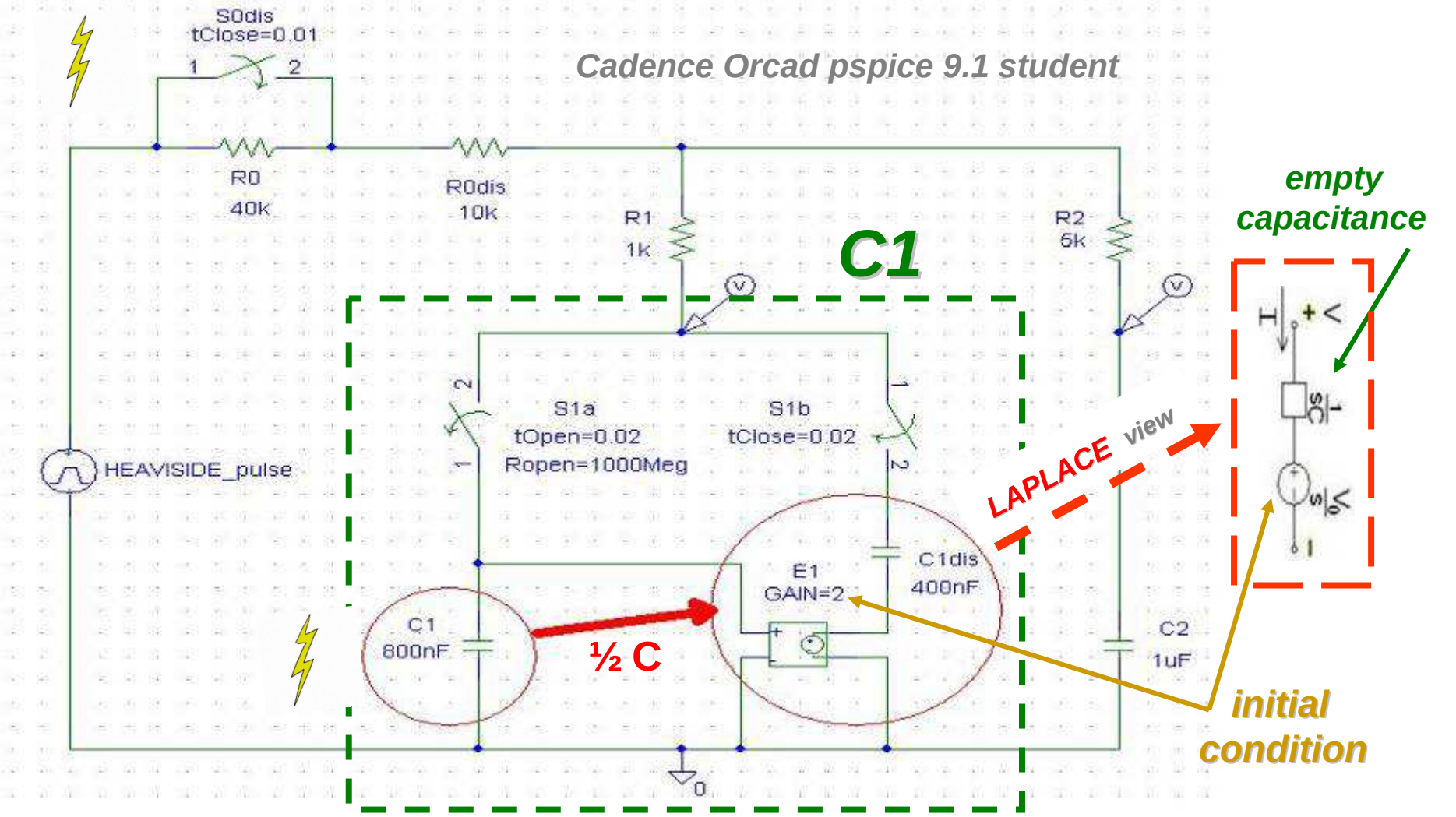
" RK4 " error



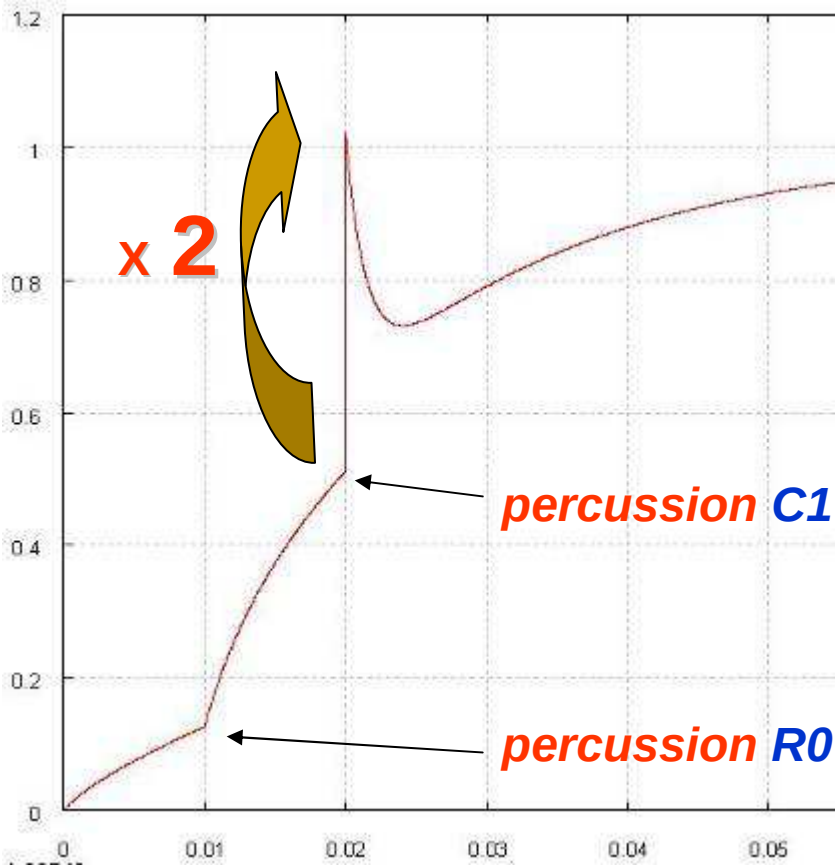
Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

PERCUSSION

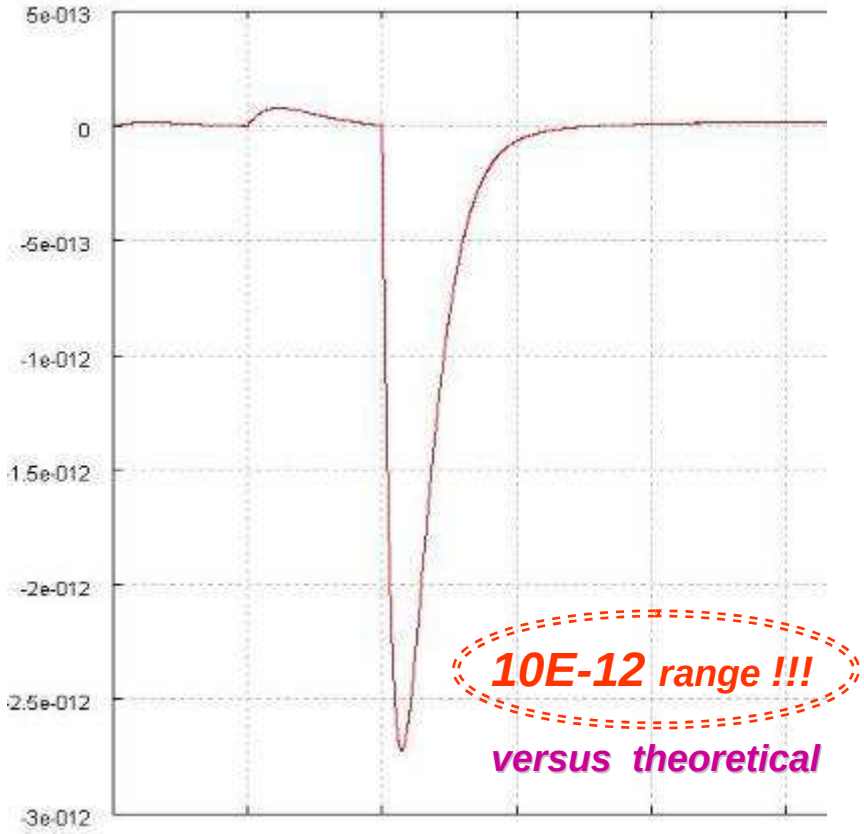
at	t =	0.01	R0	50 Kohm	==>>	10 Kohm
at	t =	0.02	C1	800 nF	==>>	400 nF



VC1 output



error on VC1 output



The **INPUT** of this test case is a trivial ***HEAVISIDE***
Therefore, there is ***no contribution*** of the ***"ILT package"*** **INPUT** ***intrinsic sampling error method*** to this filter ***output response***

This is a favorable situation to measure on its own the percussion processing quality

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

SPICE is a ***matrix based solver***
computing the behavior of some ***(linear*** and/or ***non-linear)*** ***global circuit*** by the ***full evaluation*** of its ***possibly very (very) large*** number of *individual* ***NODES***

INTEGRATION (*and* ***NON-LINEARITY***) are handled through **"*iterative*" *heuristics***

for ***Linear Time Invariant*** (***L T I***) *circuit*,
TRANSFER FUNCTION models the behavior of a ***single observed signal***

The ***DENOMINATOR*** reflects the complexity of the ***circuit STATE*** "***as a whole***"
Therefore, to limit the ***polynomial order*** this method is reserved to ***small circuit***

Nevertheless, the advantage of a ***TRANSFER FUNCTION*** is that the associated ***DIFFERENTIAL EQUATION*** allows the manipulation of the overall ***circuit STATE*** *through* ***"algorithmic equations"***

This is why this facilitates the modeling of ***PERCUSSIONS***

PERCUSSIONS also occur in a "pinball"

By analogy with a Differential Equation, the metal ball has a trajectory

An instantaneous shock on an obstacle is enough to change it forever . . .

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

HALL of Fame

Brook Taylor	1685 - 1731
Marquis Pierre-Simon Laplace	1749 - 1827
Joseph Fourier	1768 - 1830
Gustav Robert Kirchhoff	1824 - 1887
Oliver Heaviside	1850 - 1925
Carl Runge & Martin Wilhelm Kutta method	1901
Paul Adrien Maurice Dirac	1902 - 1984
Laurent Schwartz	1915 - 2002
M I T Macsyma (now *Maxima*)	*late* 1960s
SPICE 1 public domain	1972

Jacques MEQUIN
NICE EWTBU / DTE
2012 j-mequin@ti.com

QUESTIONS ?