Simulation of Signal and Transfer Function on XCOS simulator

[1]Garima Jain, [2]Nasreen Noorani, [3]Kavita Hedau
EC Department, RGPV University
Acropolis technical campus, Indore, India
[1]gari.2911@gmail.com
[3]kavita.kolhe@gmail.com
EC Department, RGPV University
Acropolis technical campus, Indore, India
[2]nasreen.atc@gmail.com

***Abstract*—In this paper we are familiar with the XCOS Palette browser and Identify the block for designing of signals and transfer function. now a day's open source software are popular for simulation and synthesis, XCOS is a free graphical editor and simulator based toolbox on Scilab which provide helps to design physical systems such as electrical, mechanical, automotive, hydraulics, using a graphical user interface based on a block diagram approach.**

***Keywords*— scilab 5.5.1, XCOS toolbox.**

I. Introduction

Xcos is a free graphical editor and simulator based on Scilab. It includes explicit and implicit dynamical systems and both continuous and discrete sub-systems. This toolbox is particularly useful in control theory, digital and signal processing and model-based design for multi domain simulation, especially when continuous time and discrete time components are interconnected [1]. The Xcos module is directly integrated into Scilab. It allows block diagrams and various analyze systems. Once opened Scilab, the Xcos module starts by clicking on the icon 
Two windows open:
1) An untitled window (active window) Blank will be used to create the model as a block diagram.
2) Window "Browser pallets - Xcos" which contains the pallets, i.e. library blocks that can be used in the block diagrams. You can observe the CPGE palette that includes all the commonly used blocks by category: Starters, Linear Operators, Non-linearities, Outputs, and Advanced Features. Clicking on a category shows all available blocks [1].

II. BLOCK DESCRIPTION

1. Integral Block

Palette: Continuous time systems / INTEGRAL_m

Purpose: The output of the block y(t) is the integral of the input u(t) at the current time step t .In our simulation we use this block to recover the variable L(t) starting from its derivative L'(t). The initial condition can be specified in the input mask.
Hint: Numerically it is always more robust using the integral block instead of the derivative block.

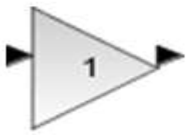
2. sum block

Palette: Math operations / BIGSOM_f

Purpose: The output of the block y(t) is the sum with sign of the input signals. The sign of the sum can be specified from the input mask with "+1" for "+" and "-1" for "-".

3. Gain Block

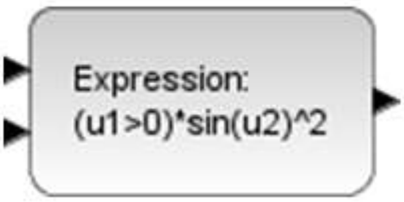
Palette: Math operations / GAINBLK_f



Purpose: The output of the block y(t) is the input signal u(t) multiplied by the gain factor. The value of the gain constant can be specified from the input mask.

4. The expression block

Palette: User defined functions / EXPRESSION



Purpose: The output of the block y is a mathematical combination of the input signals u1,u2,…,uN (max 8). The name, u, followed by a number, is mandatory. More precisely, u1 represents the first input port signal, u2 represents the second input port signal, and so on.

Note that constants that appear in the expression must first be defined in the "context menu" before their use.

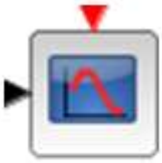
5. The clock block

Palette: Sources / CLOCK_c

Purpose: This block generates a regular sequence of time events with a specified period and starting at a given initialization time. We use this block to activate the scope block (see next step) with the desired frequency.

6. The scope block

Palette: Sinks / CSCOPE



Purpose: This block is used to display the input signal (also vector of signals) with respect to the simulation time. For a better visualization it may be necessary to specify the scope parameters like *ymin*, *ymax* values.

7.The multiplexer block

Palette: Signal routing / MUX

Purpose: This block merges the input signals (maximum 8) into unique vector output signal. We use this block for plotting more signals in the same windows. The number of input ports can be specified from the input mask.

8.The simulation starting time

Each Xcos simulation starts from the ***initial time 0*** and ends at a specified final time.

The ending simulation time should be specified in the "Simulation/Setup" menu in the "Final integration time" field [2].

III. PROCEDURE

A. Perform output of three square wave generator using xcos toolbox.

Step 1:- choose the block "GENSQR_f " three times. This block is a square wave generator: output takes values -*M* and +*M*. Every time an event is received on the input event port, the output switches from –*M* to *M*, or *M* to -*M*. set the amplitude 1 for square wave generator using right click. Typically the event input port is used to specify the signal period.

Step 2**:-** choose the clock and set period 10, 4, 1 respectively.

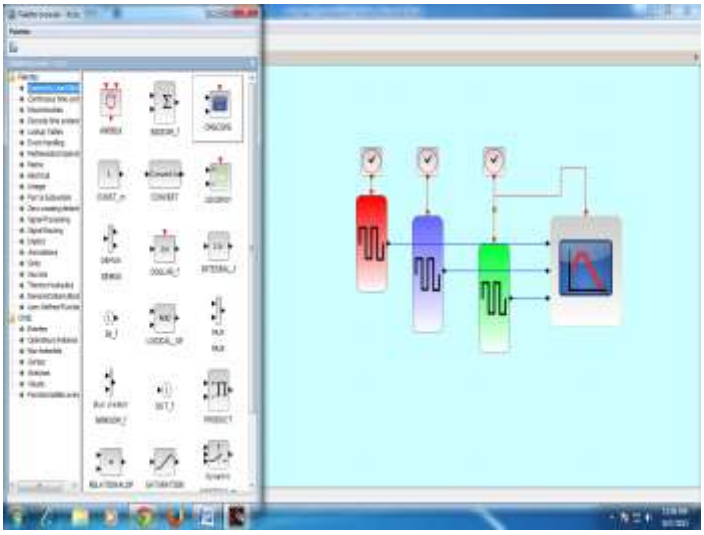


Fig. 1 Arrangement of blocks for square wave generator

First the sink block is a scope with a sample rate set to *1 second*. This parameter will affect the printed results of all the diagram branches.

The first source block (red) illustrate the typical use of this block. The period of each output level is set with a clock. The signal edge is fast and should be accurate for most of the application with a edge speed ratio of *1/10*.

The second source block (blue) illustrate a much more problematic use case. Users can use these parameters to illustrate some limitations of a design.

The third source block (green) illustrate a mis-use and a common first user error. The block activation times are the same as the scope [3].

Step3 :-connect all the block

Step4 :- set the execution time 300.

Step5:-save and run the model.

B. Perform output of transfer function of ramp input using xcos toolbox

Transfer function= 1/1+s

Step 1:- choose the block " Ramp". This block is a ramp iput: The Ramp block generates a signal that starts at a specified **Start Time** and **Initial Value** and increases by a specified rate (**Slope** parameter). The output signal is defined by the following equation:[3]

$$s(t) = \begin{cases} s_0 & \text{for } t < t_0 \\ s_0 + k.t & \text{for } t > t_0 \end{cases}$$

where t_0 is the **Start Time**, s_0 the **Initial Value** and k the **Slope**.

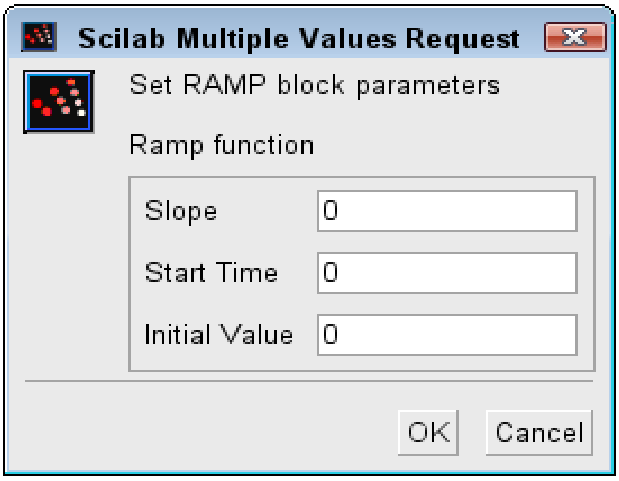


Fig. 2 Set ramp block parameter

Step 2:- fill this block

- **Slope:-0.5**

Scalar. The rate of change of the generated signal.
Properties : Type 'vec' of size 1.

- **Start Time-1**

Scalar. The time at which the signal begins to increase.
Properties : Type 'vec' of size 1.

- **Initial Value-0**

Scalar. The initial value of the signal.
Properties : Type 'vec' of size 1.

Step 3**:**- choose the Transfer function block “1/1+s”.
Step4 :-connect all the block
Step5 :- set the execution time 20.
Step6**:**-save and run the model.

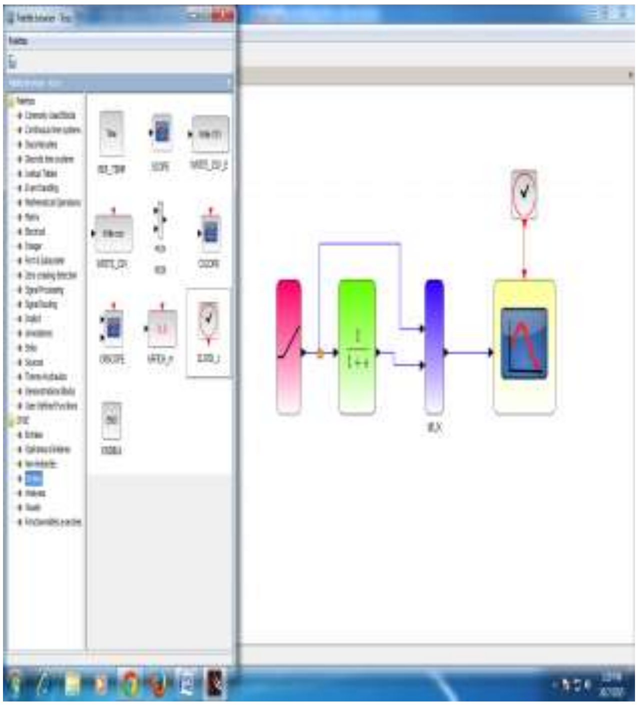


Fig. 3 Arrangement of blocks for ramp input Transfer function= 1/1+s

*C. Perform C(s)/R(s)= 10/((1+2*s)*(1+6*s)) using xcos toolbox*

Step 1:- choose the block ” step”. This block is a step input:
Step 2:- fill the parameter of block
Step 3:- choose the Transfer function block “1/1+s”. Edit with given transfer function

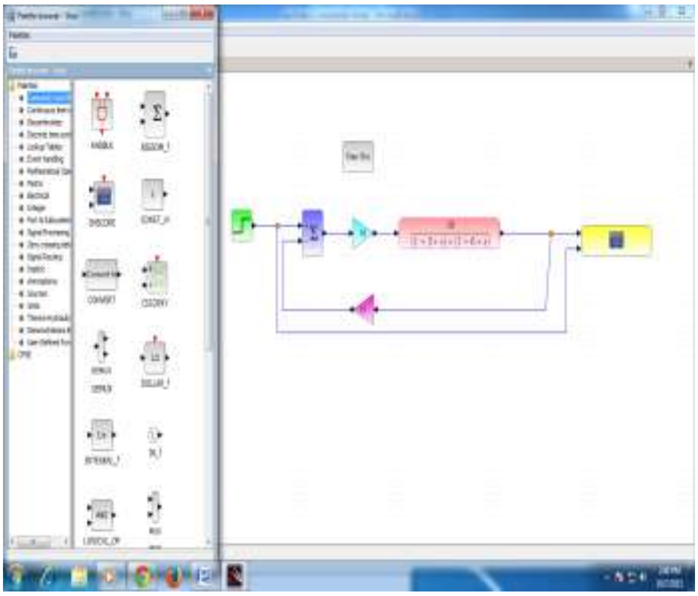


Fig. 4 Arrangement of blocks for C(s)/R(s)= 10/((1+2*s)*(1+6*s))

Step4 :-connect all the block
Step5 :- set the execution time 100.
Step6:-save and run the model.

IV.RESULT

D. output of three square wave generator:

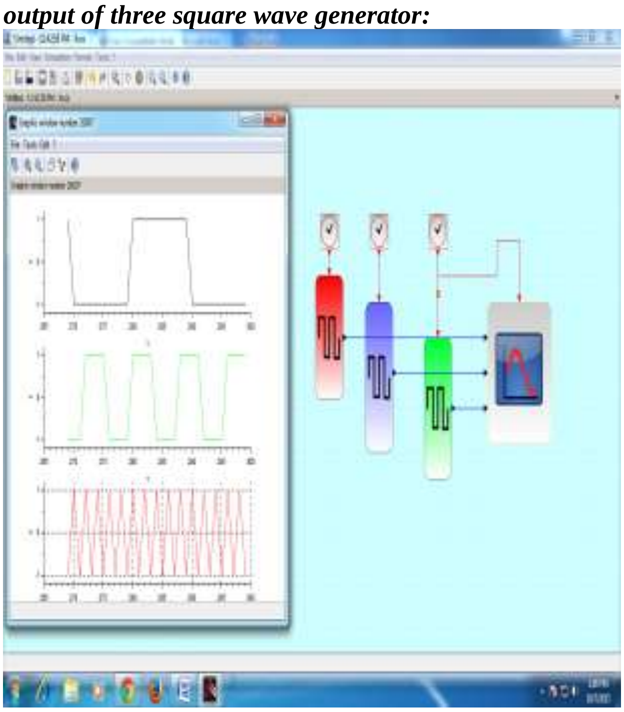


Fig. 5 output of three square wave generator

E. output of transfer function of ramp input using xcos toolbox

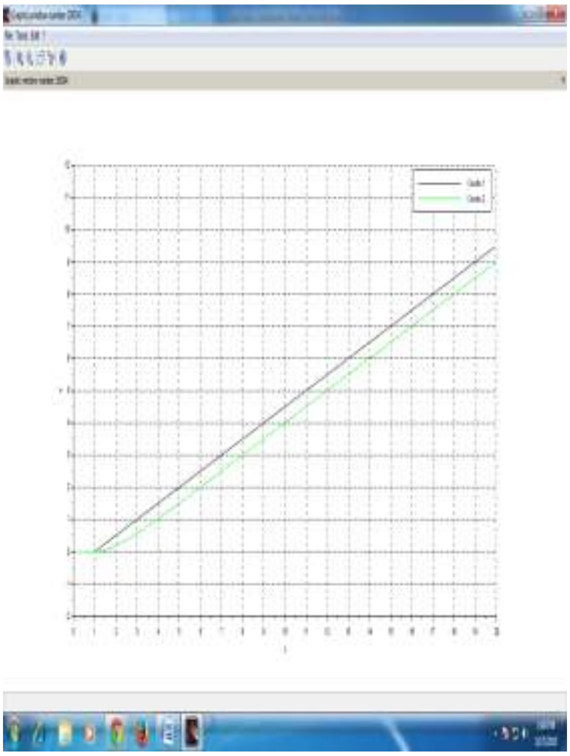
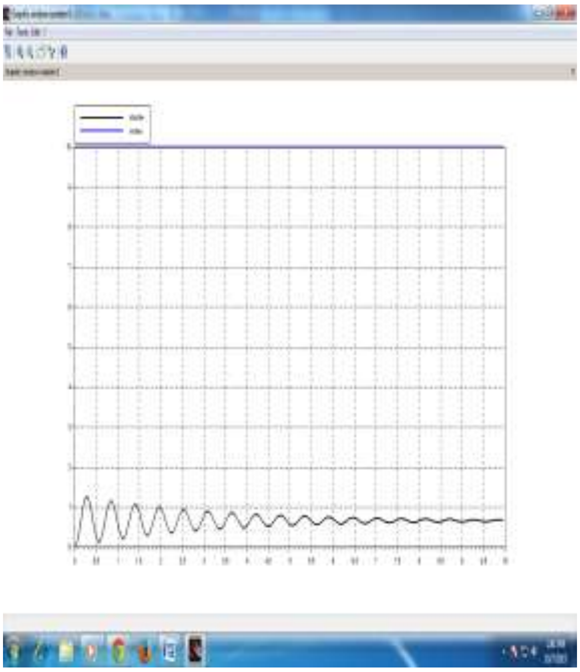


Fig. 6 output of transfer function of ramp input

*F. output of C(s)/R(s)= 10/((1+2*s)*(1+6*s)) using xcos toolbox*



V.CONCLUSION

Fig. 7 output of C(s)/R(s)= 10/((1+2*s)*(1+6*s))

In this paper, we have presented procedure to use xcos toolbox of scilab for perform the output of transfer function in graphical way and also see the way to check the output of square wave generator using number of components. This paper is used as a reference paper for future to design various signal and system of digital signal processing and control system for engineering practice.

REFERENCES

[1] Vande Wouwer, P. Saucez, C. V. Fern ández 'Simulation of ODE/PDE Models with MATLAB®, OCTAVE and SCILAB'
[2].Hema Ramchandran, Achuthsankar S. Nair'SCILAB (A FREE SOFTWARE TO MATLAB)'
[3]www.scilab..in
[4] https://atoms.scilab.org/toolboxes/xcos_toolbox_skeleton
[5]http://www.docente.unicas.it/useruploads/000822/files/scilab_presentation_ver2_3.pdf
[6]http://www.openeering.com/sites/default/files/Control%20System%20Toolbox%20in%20Scilab.pdf
[7]https://th.wikispaces.com/file/view/Using+Xcos+as+a+Teaching+Tool+in+a+Simulation+course.pdf
[8] http://www.heikell.fi/downloads/scilabpdf.pdf

