Modeling and Simulation of Pipeline by Colored Petri Nets

[1]Behnam Barzegar, [2]Homayun Motameni, [3]Zohre Ramezani and [3]Zahra Yousefi

[1]Young Researchers Club, Nowshahr Branch, Islamic Azad University, Nowshahr, Iran
[2]Department Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran
[3]Rouzbahan Institute of Higher Education, Sari, Iran

Abstract: In this paper the adding up and subtraction problem of two floating figures is described with using pipeline technique. pipeline technique models construction in the form of Colored Petri Net is introduced. For the simulation and analysis of the model the Design/CPN tool is used and its modeling with CPN Tools.

Key words: Exponent · Floating point · Pipeline · Petri net · CPN · Mantis

INTRODUCTION

Colored Petri Net (CPN) is a tool by which validation of discrete-event systems are studied and modeled. CPNs are used to analyze and obtain significant and useful information from the structure and dynamic performance of the modeled system. Colored Petri Nets mainly focus on synchronization, concurrency and asynchronous events [1]. The graphic features of CPNs specify the applicability and visualization of the modeled system.

Furthermore, synchronous and asynchronous events present their prioritized relations and structural adaptive effects. The main difference between CPNs and Petri Nets (PN) is that in CPNs the elements are separable but in PNs they are not. Colored indicates the elements specific feature. The relation between CPNs and ordinary PNs is analogous to high level programming languages to an assembly code (Low level programming language). Theoretically, CPNs have precise computational power but practically since high level programming languages have better structural specifications, they have greater modeling power.

CPN's drawback is its non adaptivity [1,2] therefore it is not possible to access the previous information available in CPNs. If there is more than one transition activated then each transition can be considered as the next shot. This Colored Petri Net's characteristic indicates that since several events occur concurrently and event incidences are not similar, then when events occur they do not change by time and this phenomenon is in contrast with the real and dynamic world. Simulation would be similar to execution of the main program. Our Purpose is to use the simulated model for analyzing the performance of the systems, as a result here the system problems and the weak points would be identified. However, classic CPN tools can do nothing to improve and solve problems and also it would not be possible to predict the next optimized situation.

Parallel processing means using variety techniques in contemporary processing of data for increase meant of speed of calcuting computer systems. Pipeline is a technique that separates a seri process to simple operation and each simple action performs at the same time of other sections in a specific section.

Pipeline can be supposed as a complex of processor part that two by two (2-2) information circulates in it. One of specifications of pipeline is that, several calculations current at the same time at different parts. each action that be able to separate to a chain of details with the same complexity can be layed down by pipeline processor. This technique is useful for the group of applicants that need the repetition of the same operation on a complex of different data. The math pipeline usually finds in rapid computers and uses for laying down of floating point operations, multiple of numbers with stable floating point and other calculating like this that finds in scientific problems.

In this Paper an example of a pipeline is showed for adding of floating point. The rest of the paper is structured as follows. In section Colored Petri Nets, we present CPN as a basic strategy used in our proposed method. in section Pipeline, the considered problem, which is the same pipeline for add/subtraction of floating point, will be described. The proposed algorithm is introduced in last Section and in the end we have concluding remarks.
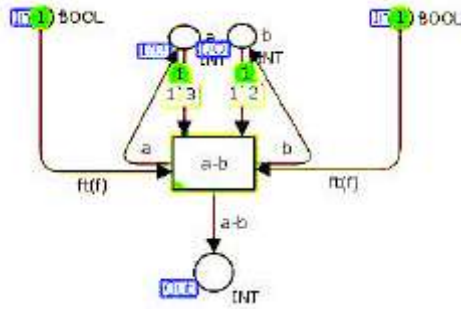

Corresponding Author: Behnam Barzegar, Young Researchers Club, Nowshahr Branch,
Islamic Azad University, Nowshahr, Iran.

Colored Petri Nets: The Colored petri net is a language for modeling and evaluating of system that co-processing connection and synchronization play the main rule in it. This language is a complex of petri net with ML programming language [3] the CPN model is an executive model system that shows position of a system and incidents which cause change in position of system.

CPN Tools is also a suitable tool for editing, modeling, analysis of space of position and analysis of operation of CPN models. graph of Petri net is a method for showing of structure of Petri nets that two shapes are in them. These places and transitions connect to each other by arcs. when an arc connects from a transition to a place, it means, the place will be the exit of the mentioned transition and if an arc be drawn from a place to a transition, it means, that place will be entrance of the mentioned transition. For description of Petri net action, tokens add to this graph, too. it causes, concept of position be defined in this graph.

Number of these tokens in the whole graph and manner of their distribution among places, determines the position of Petri net, which called it a petri net marking.

A formal definition of CPN is as follows: [1,2,4] A Colored PN (CPN) is a 6-tuple CPN (P,T,C, I^-, I^+, M_0) where:

- P = {p1, p2,..., pn} denotes a finite and non-empty set of places,
- T = {t1, t2,..., tm} denotes a finite and non-empty set of transitions, P∩T =Ø,
- C is a colour function that assigns a finite and non-empty set of colors to each place and a finite and non-empty set of modes to each transition.
- I^- and I^+ denote the backward and forward incidence functions defined by P×T, such that I^-(p,t), I^+(p,t)∈ [C(t) → C(p)MS],v(p,t) ∈ P×T².
- Mo denotes a function defined on P, describing the initial marking such that Mo (p) ∈ C(p)MS.

Pipeline For Adding Up/Distraction Of Floating Point: Here, we describe, using of pipeline for adding of two floating point numbers entrances of pipeline are add maker of floating point of two number of 2-2 (two-by-two) of normalized floating point.

$$X = A*2^a \qquad Y = B*2^b$$

A and B, are two numbers, smaller than one that make mantises and a and b are their exponent. Add and subtraction of math can be done in four parts (Figure 3).

Fig. 1: Pipeline for add up and subtraction of floating point

The constants which are shown with R, are placed among parts to store the middle results. The performed operations in parts are:

- Comparison of exponents,
- Coordination of mantises,
- Add or subtraction of mantises,
- Normalization of the result.

The bigger exponent, selects as a resulted exponent after subtraction of exponents for their comparison and finding their differences. Subtraction of exponent shows that mantis of the smaller exponent must be shifted to the right, several times. So, two mantises coordinate. Then, these two mantises will be added or subtracted in the part and finally the result will be normalized in the fourth part. When a slope be done, mantis of the complex or subtraction of a load will be shifted to the right and the exponent will have one unit increase [5,6].
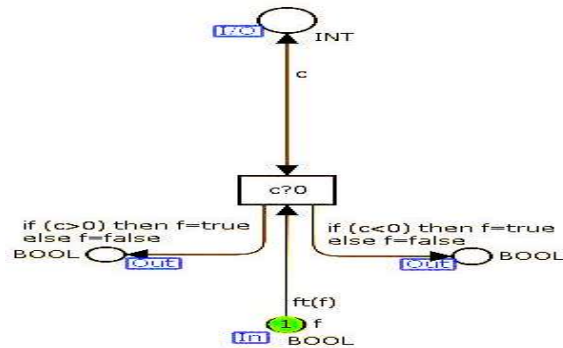
Fig. 2: Comparison of exponents

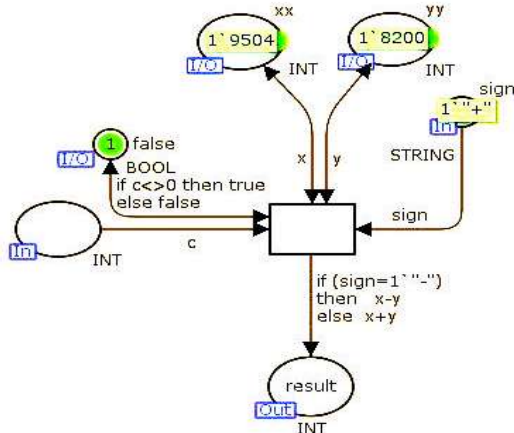Fig. 3: Comparison of exponents

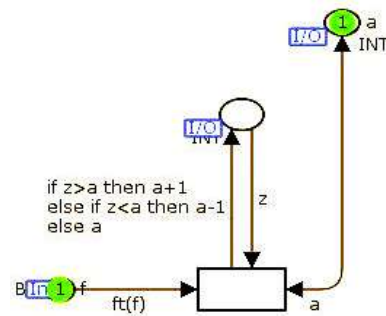Fig. 4: Coordination of mantises

Fig. 5: Add or subtraction of mantises

Fig. 6: Normalization of the result

Fig. 7: Definition of variables

If a collapse, be done, the number of zeros, beside the mantis, shows the number of left shifts of the mantis and determines the number which should be subtracted from the exponent.

The Algorithm Simulation: In this section we propose an algorithm Simulation for add up/subtraction of two floating point numbers with using of Colored Petri Net and CPN Tools. Two normalized number of below floating point are supposed:
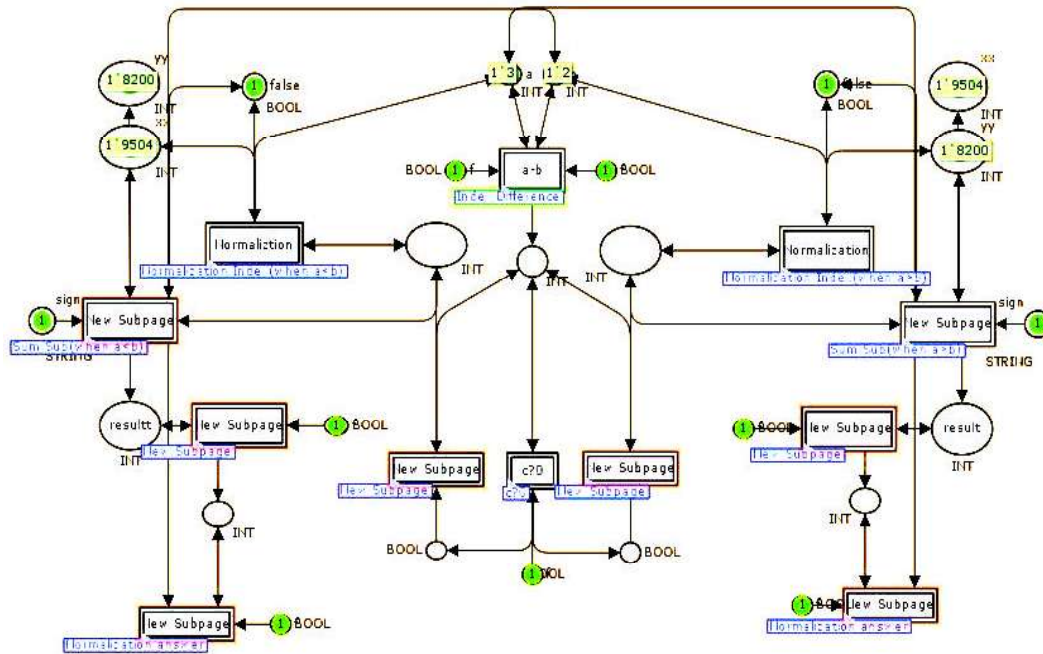
$$X = 0.9504 * 10^3, \qquad Y = 0.8200 * 10^3$$

Fig. 8: General view of Proposed Model Simulation in CPN Tools

Two exponents will be distracted of each other in the first part to 3-2=1 be caught the bigger exponent, means 3, is selected as the result exponent (Figure 2, Figure 3).

The next part shifts the Y mantis to the right to the below relations be gotten:

$$X=0.9504 * 10, \qquad Y=0.0820 * 10$$

So, two mantises will be coordinated and will have the same exponent.(Figure 4)

Plural of two mantises in part3 Figure 5) will make the below total: Z=1.0324 * 10

The total will be set by its normalization (Figure 6), so that it be the first nun zero figure in decimal side. It gets through right shift and raised exponent. Z=0.10324* 10

Figure 7, shows manner of definition of variables and Figure 8, shows general view of modeling in CPN Tools.

CONCLUSION

Pipeline is a method that can be found in very rapid computers and applies for laying down of floating point activities. Through this way, activities of can be separated into details simply and in this way, accelerate the ordinal factor of a program. In fact, pipeline is division of one ordinal action to several operational unit. we can consider pipeline as some consecutive floor that data enter and each floor, does specific act on the data.

With using CPN Tools which is a powerful tool for co processing modeling and provides more description power regard to staff nets, we can models pipeline.

REFERENCES

1. Jensen, K., L. Michael Kristensen and L. Wells, 2007. Colored Petri Nets and CPN Tools for modeling and validation of concurrent systems. International Journal of Software Tools Technology Transfer, pp: 213-254.
2. Jensen, K., 1997. Colored Petri Nets-Basic Concepts, Analysis Methods and Practical Use. Springer-Verlag, pp: 1-3.
3. Beaudouin-Lafon, M., W.E. Mackay and M. Jensen, 2001. CPN Tools: A Tool for Editing and Simulating Colored Petri Nets. LNCS 2031: Tools and Algorithms for the Construction and Analysis of Systems, pp: 574-580.
4. Peterson, J., 1981. Petri Net Theory and the Modelling of Systems, Prentice Hall.
5. Harper, R., 2000. Programming in M.L. Standard, Carnegie Mellon University.
6. Turon, A., 2006. SML/NJ Language Processing Tools: User Guide.