Modeling a Smart Sensor Network with Petri Nets

Israel L. Melo
Laboratório de Dispositivos e Nanoestruturas
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
CEP: 50670-000 Caixa Postal 7800
Recife, PE Brasil
israel@ee.ufpe.br

Tomaz C. Barros
Laboratório de Dispositivos e Nanoestruturas
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
CEP: 50670-000 Caixa Postal 7800
Recife, PE Brasil
tomaz@ee.ufpe.br

Edval J. P. Santos
Laboratório de Dispositivos e Nanoestruturas
Departamento de Eletrônica e Sistemas
Universidade Federal de Pernambuco
CEP: 50670-000 Caixa Postal 7800
Recife, PE Brasil
edval@ee.ufpe.br

ABSTRACT

In this work, the data transmission between a sensor matrix and a supervisor is modeled. The model is constructed with Petri nets to analyze synchronism and the occurrence of deadlocks.

Keywords

Petri Net, Smart sensor, network.

1. INTRODUCTION

Sensor matrix is a technique that can be used to collect signals of an area or volume. The collected information is transmitted to a supervisor for processing and/or storage.

The transmission of the information to the supervisor is a process where the competition for the access to the network can happen. The modeling of the transmission process becomes, therefore, necessary. With this objective, we have developed a model that uses Petri nets to analyze the synchronism and examines the possibility of the occurrence of deadlock.

2. DESCRIPTION OF THE PROBLEM

Consider a sensor matrix to collect temperature information of an external region. At a certain moment, m sensors, among a total of n sensors in the matrix of sensors (with $m \leq n$), desire to transmit data at the same time (or inside of a short time interval) to a supervisor. The data refers to the state of the plant in a certain moment. Therefore it cannot be mixed with data collected at different time, i.e., different state of the plant, so that the supervisor can calculate the correct state of the monitored plant.

The ideal would be that all data of a certain state of the plant is transmitted at the same time to the supervisor, as soon as available. But that is not possible because of two reasons: 1) the transmission medium holds only one connection ta the same time, and 2) the supervisor can only talk to one sensor at the same time.

It can exist the problem of some sensors not to get the chance to transmit its data at the correct moment (synchronism problem), and still the problem of the racing condition [2]. From the point of view of real time systems [4], this represents the generation of false results, because to receive the correct answer, but very late it is frequently as bad as not to receive it. Another problem is the "busy wait" [3], that is, the wait for an unavailable event, for example, free bus for transmission.

The interest of this work is in the coordination of the sensors for the transmission of data in certain instant (the synchronism concept). This will be made with the support given by the Petri nets.

3. MODELING

To study how to model with Petri nets we propose a simplified protocol. Whenever a sensor has something to transmit, it requests the attention of the supervisor. Before, however, it is verified if the bus it is free (place P13), so that there is not the risk of collision of transmitted data. In negative case, to avoid the busy wait for the event "free bus", we placed the wait for that event in a list of events that the sensor waits, and the sensor continues with other internal activities. As soon as the bus becomes free, the sensor tries again to publish your interest of transmission on the net.

When a sensor gets access to the bus and informs your interest of transmission of data (place P12), all the other sensors and the supervisor will take knowledge (places P11 and P20). And all will know how many sensors desire to transmit (places P11 and P20).

After a sensor to have informed its interest to send data, it will be ready to transmit. Before, the sensor verifies if the bus is free and if the supervisor authorized the sending of data (place P13). In case one of those conditions is negative, again to avoid a "busy wait" for the events "free bus" e "authorization of sending of data" we proceed as before (place P7 and transitions T6 and T7).

The data will be sent when the supervisor authorizes and the bus is free. The sensor waits the confirmation sending and the transmission process with this sensor will be locked up.

It should be observed that, after a first sensor to have required the supervisor's attention, it will be given a interval small of time (timed transition T10) so that the other sensors also to make its petitions. After this time, the sensors will not be able to dispose messages to be transmitted, besides those already available. And they will only come back to dispose messages when the sensor last has transmitted its data.

The supervisor, in turn, thus that detects that a first sensor desires to transmit (place P19), waits for a certain interval of time, in order to make possible that all the sensors manifest its interest in transmitting. Only after this time, the supervisor authorizes the begin of the transmission of data. The integrity of the data received by the supervisor is verified. Being all certain with the received data, the supervisor authorizes the emission next of data (transition T18). This continues until all the petitioning sensors have transmitted its referring data to the plant.

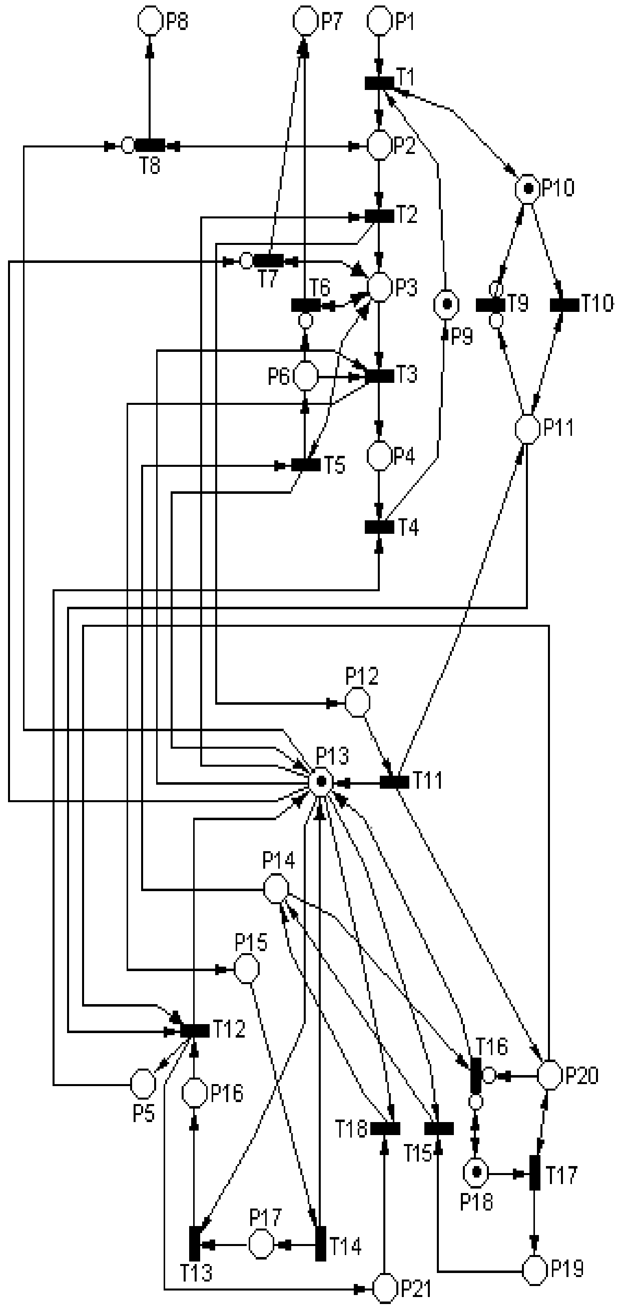

Figure 1: Sensor-Supervision Communication

4. CONCLUSIONS

The model was simulated through the "Visual Object Nets++" program [6]. The same was shown satisfactory with relation to the problems of the “competition among the sensors” and “busy wait”. However, the model doesn't guarantee the freedom of collision of data.

Other more appropriate protocols can be modeled by Petri nets, with pretension of analysis, allowing the identification of deadlocks, busy wait and collision of data.

5. ACKNOWLEDGMENTS

The authors acknowledge the support of UFPE and CNPQ institutions without which this work would not have been possible.

6. REFERENCES

[1] T. Murata, Petri nets: Properties, analysis, and applications, Proc. Of the IEEE, vol 77, p. 541-580. Apr. (1989).

[2] Andrew S. Tanenbeum, Operating Systems: desing and implementation, Prentice-Hall.

[3] Ricado A. Lopes, Sistemas de Redes para Controle e Automação, p. 280, Book Express, Rio de Janeiro (2003).

[4] Boke, C.,; Gotz, M.; Heimfarth, T.; El Kebbe, D.; Ramming, F.J.; Rips, S., Configurable Real-Time Operating Systems and Their Applications, p 148-155, Proceedings of the Eighth International Workshop on, 15-17, Jan. (2003).

[5] Paul G. Ranky, Smart sensor, Sensor Review, 312, (2002).

[6] www.systemtechnik.tu-ilmenau.de/~drath/visual_e.htm