MODELLING OF EMBEDDED MECHATRONIC SYSTEMS USING HYBRID PETRI NETS

VESSELKA DURIDANOVA, THORSTEN HUMMEL

Ilmenau Technical University
Department of Computer Architectures
P.O. Box 100565, 98684 Ilmenau, Germany
email: *vesselka, thummel@theoinf.tu-ilmenau.de*

Abstract: *The paper describes the challenges of modelling hybrid embedded systems. It discusses the problems of modelling such systems and suggests the use of hybrid Petri nets. The potential of hybrid Petri nets is shown by modelling an exemplary embedded mechatronic system with a special hybrid Petri net class using a special modelling tool.*

Keywords: Modelling, Embedded systems, Mechatronic systems, Hybrid Petri nets

1. INTRODUCTION

The design of complex embedded systems makes high demands on the design process due to the close combination of hardware and software components. These demands rise rapidly, if the system includes components of different time and signal concepts. Such systems are called heterogeneous or hybrid systems.

The special characteristic of mechatronic systems in comparison to classical systems originates from their higher heterogeneity and complexity. Even the simplest mechatronic system consists of subsystems of different physical natures. Modern electric drive systems comprise mechanical, electro mechanical and electronic subsystems. Furthermore each mechatronic system includes components of different time and signal concepts: continuous, discrete, and mixed-mode components. So this heterogeneity concerns both the physical principles in the whole system and the behaviour of variables inside the subsystems.

The behaviour of such heterogeneous systems cannot be covered in a homogeneous model by the well-known specification formalisms of the different mechanical, hardware or software parts because of the special adaption of these methods to their respective field of application and the different time and signal concepts the several components are described with. Continuous components are usually described by a continuous time model, whereas digital components are described by discrete events.

For describing both kinds of behaviour in its interaction, there are different approaches to describe such systems. On the one hand the different components can be described by their special formalisms. On the other hand a homogeneous description formalism can be used to model the complete system with its different time and signal concepts, and that is what we are in favour of.

So we have investigated modelling methods that can describe the behaviour of such systems homogeneously at a high abstraction level independently from their physical or technical details. Apart from considering the heterogeneity, the modelling method must cope with the high complexity of mechatronic systems. In addition to their basic functions (e.g. motion generation for electric drives) further auxiliary functions have to be performed (positioning, air supply observation, laser control, error recognition in every subsystem etc.). These demands require support for modularisation, partitioning and capabilities for hierarchical structuring.

In the following a graph based formal modelling approach is presented. It is based on a special Petri net class, which has extended capabilities for modelling of hybrid systems. To model the hybrid systems, we have used an object-oriented modelling and simulation tool based on this Petri net class. This tool can be used for modelling of hybrid systems from an object-oriented point of view. It can be used for modelling of components or subsystems and offers capabilities for hierarchical structuring.

2. HYBRID PETRI NETS

The theory of Petri nets has its origin in C.A. Petri's dissertation "Communication with Automata" [1], submitted in 1962. Petri nets are used as a description formalism in a wide range of application fields. They offer formal graphical description possibilities for modelling of systems consisting of concurrent processes. Petri nets extend the automata theory by aspects like concurrency and synchronisation.

A method to describe embedded hybrid systems homogeneously is the use of hybrid Petri nets [2]. They originate from continuous Petri nets introduced by David and Alla [3]. A basic difference between continuous and ordinary Petri nets is the interpretation of the token value. A token is not an individual anymore, but a real quantity of token fragments. The transition moves with a velocity of flow the token fragments from the place before to the place thereafter. The essence of hybrid Petri nets is the combination of continuous and discrete net elements in order to model hybrid systems.

In the past there were described applications of hybrid Petri nets in many cases, but essentially they were concentrated on the fields of process control or automation. In the following we demonstrate the possibilities of using hybrid Petri nets to model embedded hybrid systems. The used Petri net class of Hybrid Dynamic Nets (HDN) and its object-oriented extension is described in [4] and [5]. This class is derived from the above-mentioned approach of David and Alla and defines the firing speed as a function of the marking from the continuous net places.

Components or subsystems are modelled separately and abstracted into classes. Classes are templates, which describe the general properties of objects. They are grouped in class libraries. Classes can be used to create objects, which are called instances of these classes. If an object is created by a class it gets all attributes and operations defined in this class.

One of the important advantages of using this concept is the ability to describe a larger system by decomposition into interacting objects. Because of the properties of objects, the modification of the system model could by easier achieved. The object-oriented concept unites the advantages of the modules and hierarchies and adds useful concepts like reuse and encapsulation.

3. MODELLING AN EMBEDDED MECHATRONIC SYSTEM

The application example we have chosen to discover the possibilities of using hybrid Petri nets for modelling of embedded hybrid systems, is an integrated multi-coordinate drive [6]. This is a complex mechatronic system including a so called multi-coordinate measuring system.

Fig. 1 shows this incremental, incident light measuring system consisting of three scanning units fixed in the stator and a cross-grid measure integrated into the stage. The two y-systems allow to determine the angle of rotation φ. The current x, y1 and y2 position is determined by the cycle detection of its corresponding sine and cosine signals. The full cycle counter keeps track of completed periods of the incremental measuring system. This is a precondition for the following high interpolation. The cycle counter of these signals is a function of the cross grid constant and the shift

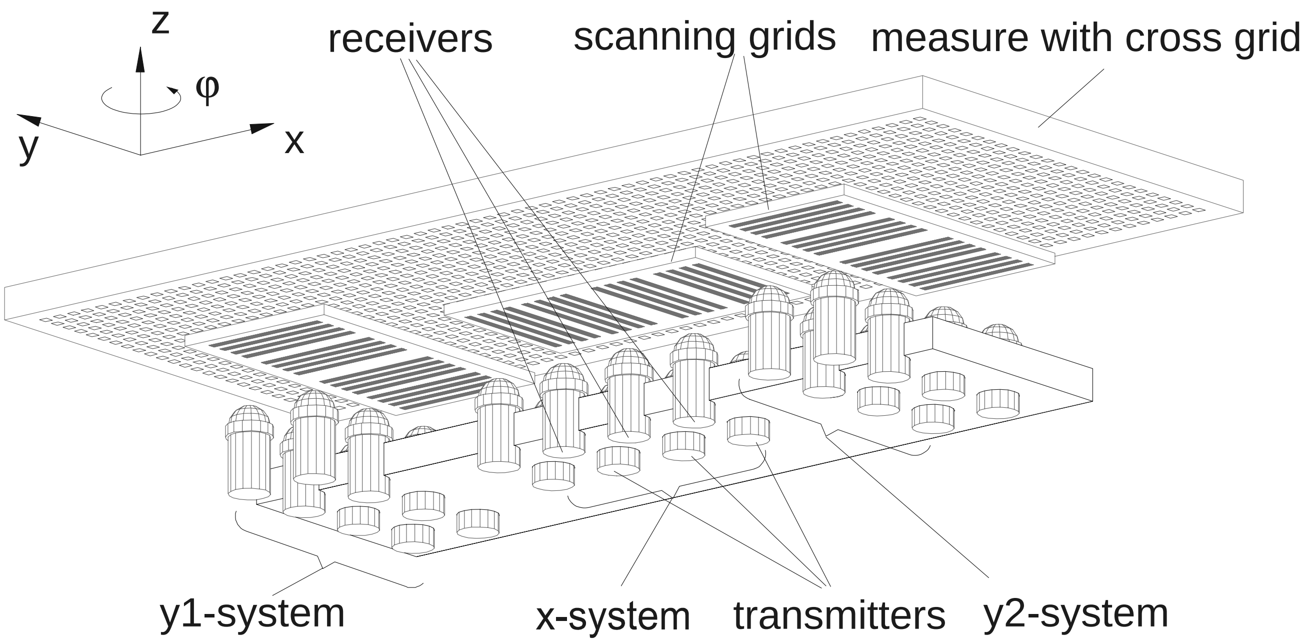

Fig. 1 Multi-coordinate measuring system

between the scanning grids and the measure. The cycle counter provides a discrete position and in many cases, this precision is sufficient for the motive control algorithm. To support a very precise position control with µm or nm resolution, it must be decided, which possibility of increasing the measure precision is the most cost-efficient. There is a limit of improving the optic and mechanical properties because of the minimum distances in the grid.

Alternatively, an interpolation within a signal period can be used, whereby the sampling rate of the A/D-Converter is increased, which would allow a more detailed evaluation of the continuous signals of the receiver. The problem to be solved in this application example results in modelling the measure system together with the evaluation algorithm for the position detection.

The measuring system is modelled hierarchically using components (Fig. 2). Components with the same functionalities are abstracted into classes, put into a class library, and instantiated while modelling. The modelling of a multi-hierarchical system is possible as well.

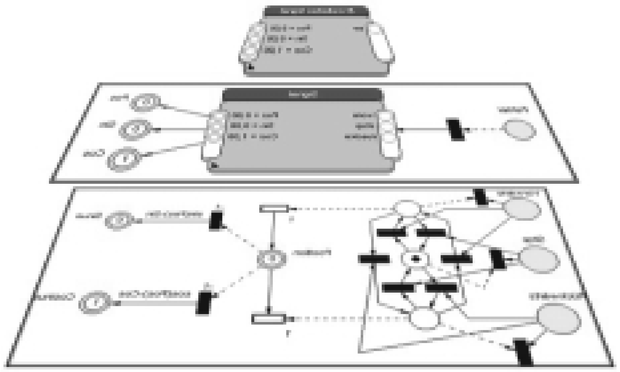

Fig. 2 The principle of hierarchical modelling

System environment

The component “Signal generation“ (Fig. 3) simulates the sensor data and provides the sine and cosine signals as well as a position value.

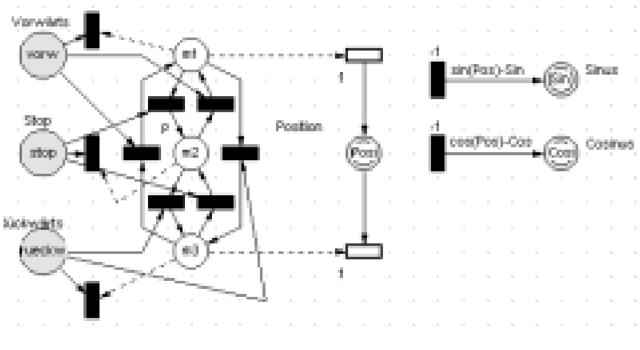

Fig. 3 Component “Signal generation”

For clearness reasons this net is saved as a component into a subnet (Fig. 4) and gets the input places “Forward”, “Stop”, and “Backward”. It provides a sine and a cosine signal and additionally a position signal as a comparative value for a later error control function.

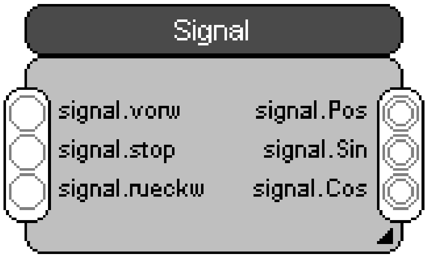

Fig. 4 Component “Signal”

To simulate a potential misbehaviour of the measuring system, external disturbances are modelled in the subnet “Scrambler“ (Fig. 5), which is included in the component “Disturbance“ of the complete system.

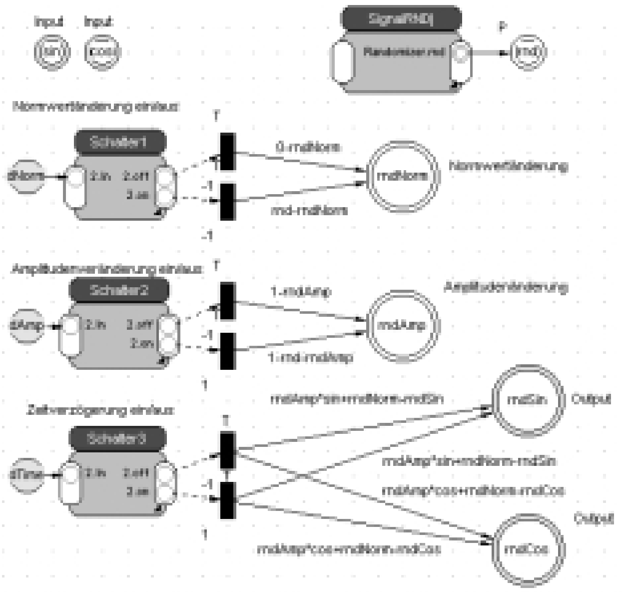

Fig. 5 Subnet “Scrambler”

Measuring system components

The position detection of one axis is modelled with the component “Axismess” (Fig. 6).

At first the input signals “Sine” and “Cosine” are normalized in the subnets “Minmax_s” and “Minmax_c” (Fig. 7). These subnets are identical in its functions and were instanced during the modelling process from the same class “Minmax”.

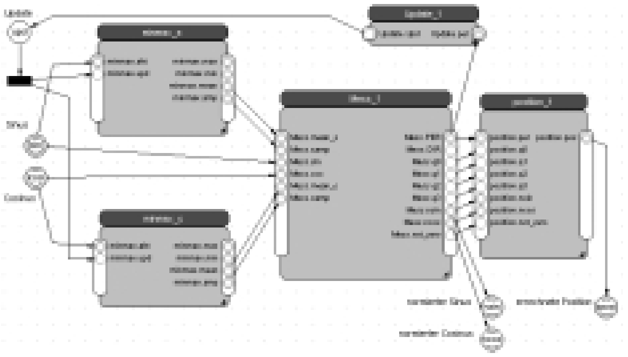

Fig. 6 Subnet “Axismess”

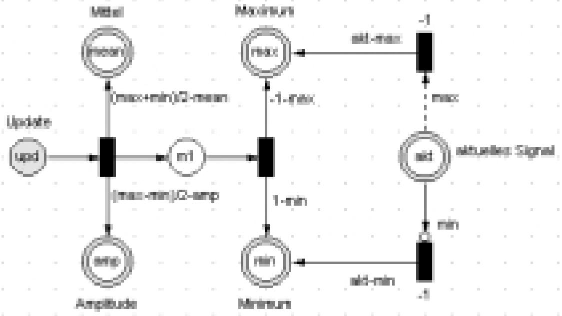

Fig. 7 Subnet “Minmax”

Following this, the cycle number is determined in “Mess_1” (Fig. 8) and finally in “Position_1” (Fig. 9) the exact position is determined.

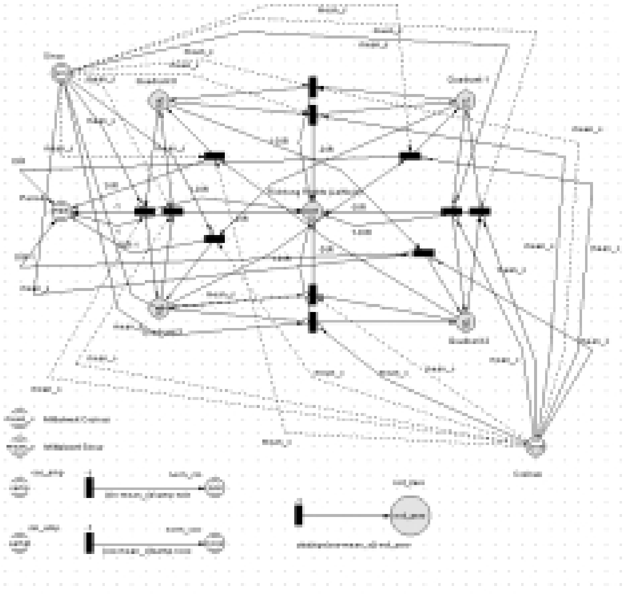

Fig. 8 Subnet “Mess_1”

To find out the exact position of the carrier, the cycle number has to be determined. To determine this correctly, the measuring system has to detect the moving direction of the carrier and with it the increasing or decreasing of the cycle number. The original measuring system used a look-up table, but this was very hard to model with Petri nets. So we changed this into logic rules and used this to model the subnet “Position_1” (Fig. 9).

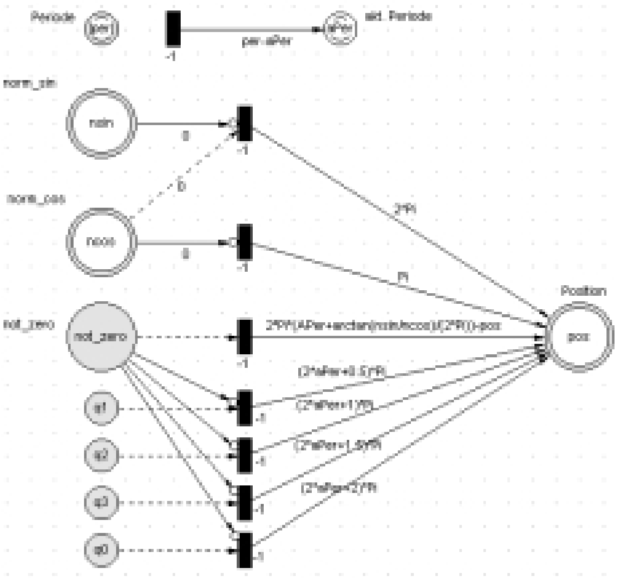

Fig. 9 Subnet “Position_1”

Model of the entire system

In Fig. 10 the model of the entire system is shown. Besides the measuring system it includes the components for signal generation and external disturbance simulation. The components for signal generation “x/y1/y2-direction” are instances of the class “Signal” and model the signals of an ideal environment.

The component “Disturbance” includes the simulation of various kinds of signal disturbances (displacement of the zero line, amplitude errors, time delay etc.). The signal disturbances can be turned on and off at any time during the simulation.

The objects “Axismess_x/y1/y2” are based on the class “Axismess” and include the evaluation algorithm for the three directions. The motion of any desired direction can be controlled by feeding marks into the places m1 to m8.

The x-position, the middle y-position and the divergence of the y-position arose as result of the net calculation.

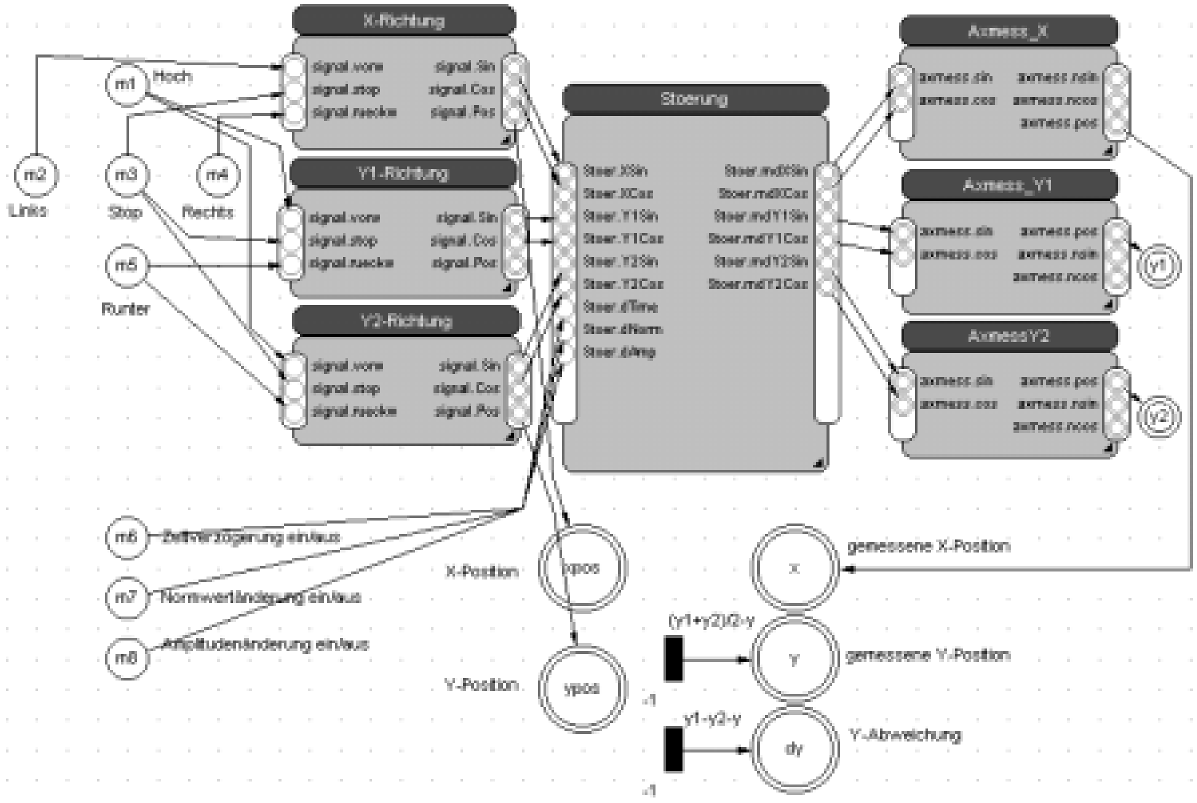

Fig. 10 Model of the entire system

System simulation

The tool "Visual Object Net" allows not only the modelling but also the simulation of systems described with Hybrid Dynamic Nets.

During the simulation the firing of the transitions and the transport of the marks are shown as animation. The changes of the place values can be visualized by signal diagrams (Fig. 11).

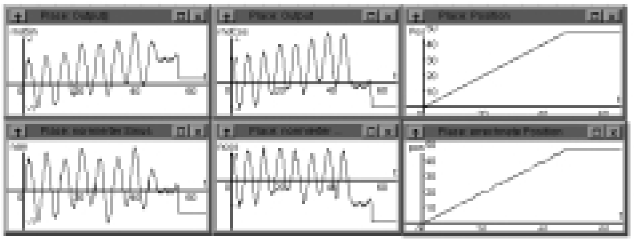

Fig. 11 System behaviour with different disturbances

E.g., the middle top diagram in Fig. 11 shows an extreme example of a simulation with disturbances. It shows a clear exceeding of the zero line of the cosine signal. Nevertheless the normal values are correctly calculated and the position of the machine is correctly displayed.

4. CONCLUSION

Our investigation has shown the advantages of using hybrid Petri nets for a homogeneous modelling of an embedded mechatronic system. The object-oriented approach of the used hybrid Petri net class makes a clear modelling of complex hybrid systems possible.

Future things that have to be done are the extension and completion of the system model and the integration of the modelling process in a complete design flow.

5. ACKNOWLEDGMENT

This research work is supported by the DFG (Deutsche Forschungsgemeinschaft, (German research association)) as part of the investigation project "Design and design methodology of embedded systems" with the subject "Design of embedded parallel control systems for integrated multi-axial motive systems" under grant FE373/13-1.

REFERENCES

[1] Petri, C.A.: *Kommunikation mit Automaten*. Schriften des IIM Nr. 2, Institut für Instrumentelle Mathematik, Bonn, 1962. (in german)

[2] Alla, H., David, R., Le Bail, J.: Hybrid Petri nets. *Proceedings of the European Control Conference*, Grenoble, 1991.

[3] Alla, H., David, R.: Continuous Petri nets. *Proceedings of the 8th European Workshop on Application and Theory of Petri nets*, Saragossa, 1987.

[4] Drath, R.: *Modellierung hybrider Systeme auf der Basis modifizierter Petri-Netze*. PhD Thesis, TU Ilmenau, 1999. (in german)

[5] Drath, R.: Hybrid Object Nets: An Object-oriented Concept for Modelling Complex Hybrid Systems. In: *Hybrid Dynamical Systems*. 3rd International Conference on Automation of Mixed Processes, ADPM'98, Reims, 1998.

[6] Saffert, E., Schäffel, C., Kallenbach, E.: Control of an Integrated Multi-coordinate Drive. *Mechatronics'96*, 18.-20.09.1996, Guimaraes, Portugal, Proceedings Vol. 1, S. 151-156.