

DESIGN OF EMBEDDED CONTROL SYSTEMS USING HYBRID PETRI NETS

Thorsten HUMMEL, Wolfgang FENGLER

Ilmenau Technical University, Department of Computer Architectures,
P.O. Box 10 05 65, 98684 Ilmenau, GERMANY,
<thummel|wfengler>@theoinf.tu-ilmenau.de

Abstract. *The paper describes the challenges of modeling embedded hybrid control systems at a higher abstraction level. It discusses the problems of modeling such systems and suggests the use of hybrid Petri nets. Modeling an exemplary embedded control system with a special hybrid Petri net class using an object-oriented modeling and simulation tool shows the potential of hybrid Petri nets.*

Key Words. *Embedded Control Systems, Hybrid Petri Nets*

1. INTRODUCTION

The design of complex embedded systems makes high demands on the design process due to the strong combination of hardware and software components. These demands rise rapidly, if the system includes components of different time and signal concepts. That means there are systems including both event parts and continuous parts. Such systems are called heterogeneous or hybrid systems.

The behavior of such hybrid systems cannot be covered homogeneously by the well-known specification formalisms of the different hardware or software parts because of the special adaptation of these methods to their respective field of application and the different time and signal concepts the several components are described with. A continuous time model usually describes continuous components, whereas digital components are described by discrete events.

For describing both kinds of behavior 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, 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 favor of.

Therefore, we have investigated modeling methods that can describe the behavior of such systems homogeneously at a high abstraction level independently from their physical or technical details. Apart from considering the heterogeneity, the modeling method must cope with the high complexity of the modeled system. This demand requires support for modularization and partitioning, and capabilities for hierarchical structuring.

In the following, a graph based formal modeling approach is presented. It is based on a special Petri net class, which has extended capabilities for modeling of hybrid systems. To model the hybrid systems, we have used an object-oriented modeling and simulation tool based on this Petri net class. This tool can be used for modeling hybrid systems from an object-oriented point of view. It can be used for modeling and simulating 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 describing formalism in a wide range of application fields. They offer formal graphical description possibilities for modeling systems consisting of concurrent processes. Petri nets extend the automata theory by aspects like concurrency and synchronization.

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 the token fragments from the pre-place to the post-place with a certain velocity of flow. 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 applications of hybrid Petri nets described in many cases but essentially, they were concentrated on the fields of process control and 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 function of the marking from the continuous net places.

Components or subsystems are modeled separately and abstracted into classes. Classes are templates, which describe the general properties of objects. They are grouped into 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 inherits all attributes and operations defined in this class.

One of the important advantages in 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 be achieved easier. The object-oriented concept unites the advantages of the modules and hierarchies and adds useful concepts like reuse and encapsulation.

3. MODELING AN EMBEDDED CONTROL SYSTEM

The application example we have chosen to discover the possibilities of using hybrid Petri nets for modeling of embedded hybrid control 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 determining the angle of rotation ν . The current x , y_1 and y_2 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 deep interpolation. The cycle counter of these signals is a function of the grid constant and the shift 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.

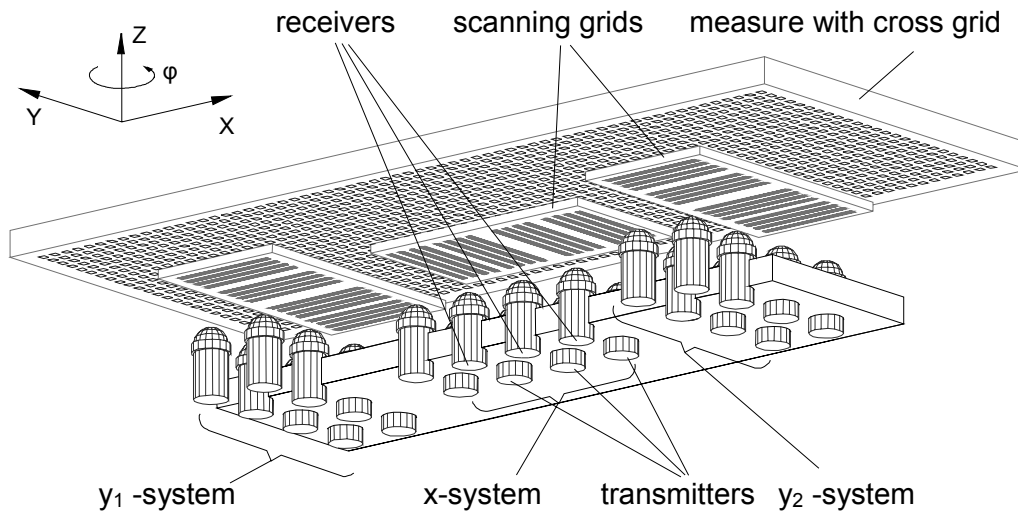


Fig. 1 Multi-coordinate measuring system

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 leads to in modeling and simulating the measure system together with the evaluation algorithm for the position detection.

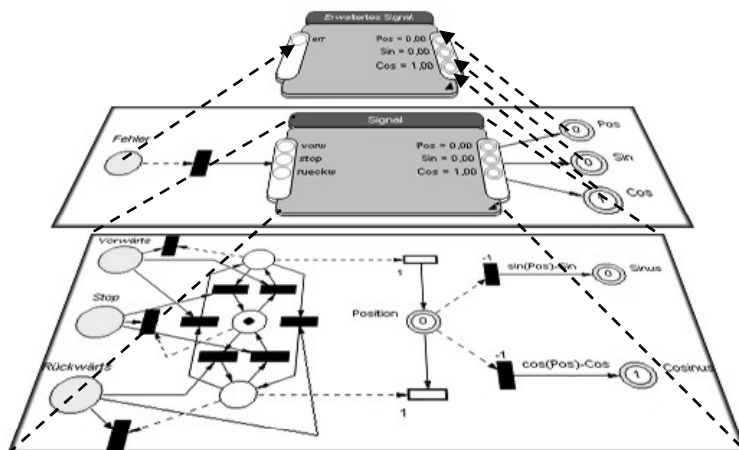


Fig. 2 The principle of hierarchical modeling

The measuring system is hierarchically modeled using components (Fig. 2).

Components with the same functionalities are abstracted into classes, put into a class library, and instantiated while modeling. The modeling of a multi-hierarchical system is possible as well.

3.1. 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. For clearness reasons this net is saved as a component into a subnet 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.

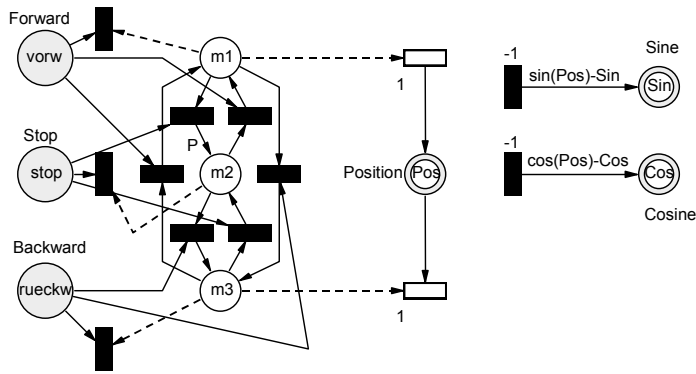


Fig. 3 Component "Signal generation"

To simulate a potential misbehavior of the measuring system, external disturbances are modeled in the subnet "Scrambler", which is included in the component "Disturbance" of the complete system.

3.2. Measuring system components

The position detection of one axis is modeled with the component "Axismess" (Fig. 4).

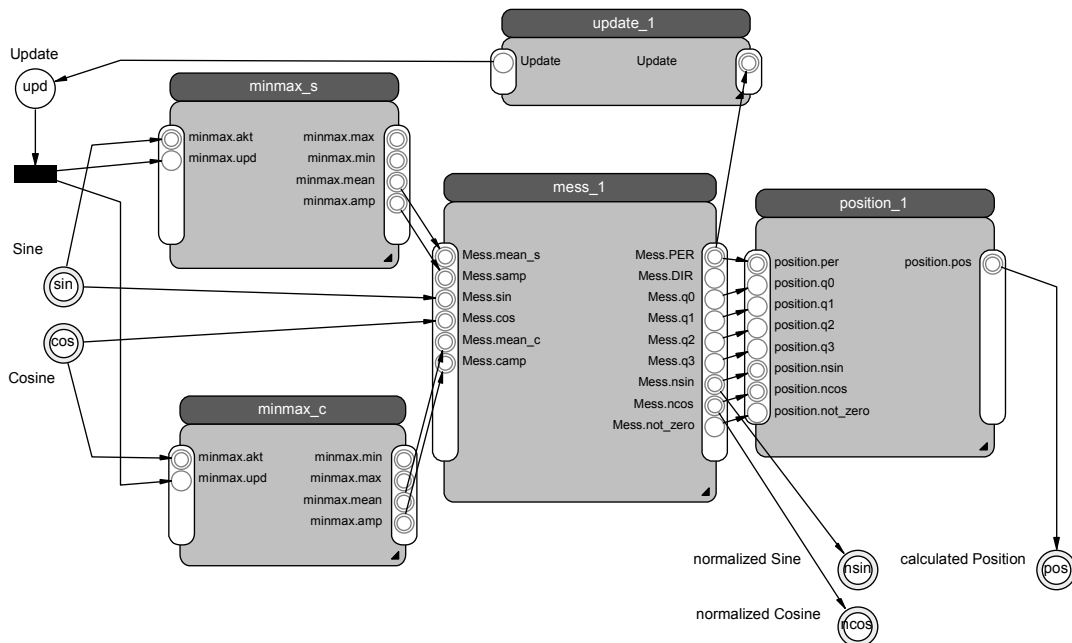


Fig. 4 Subnet "Axismess"

At first, the input signals "Sine" and "Cosine" are normalized in the subnets "Minmax_s" and "Minmax_c". These subnets are identical in its functions and were instantiated during the modeling process from the same class "Minmax" (Fig. 5).

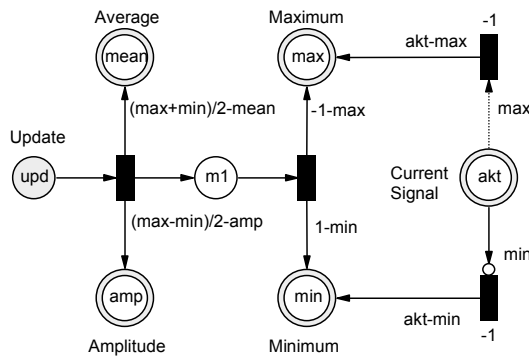


Fig. 5 Subnet "Minmax"

To find out the exact position of the carrier, the cycle number has to be determined in “Mess_1”. 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. Therefore, we changed this into logic rules and used this to model the subnet “Position_1”.

3.3. Model of the entire system

In Fig. 6, 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.

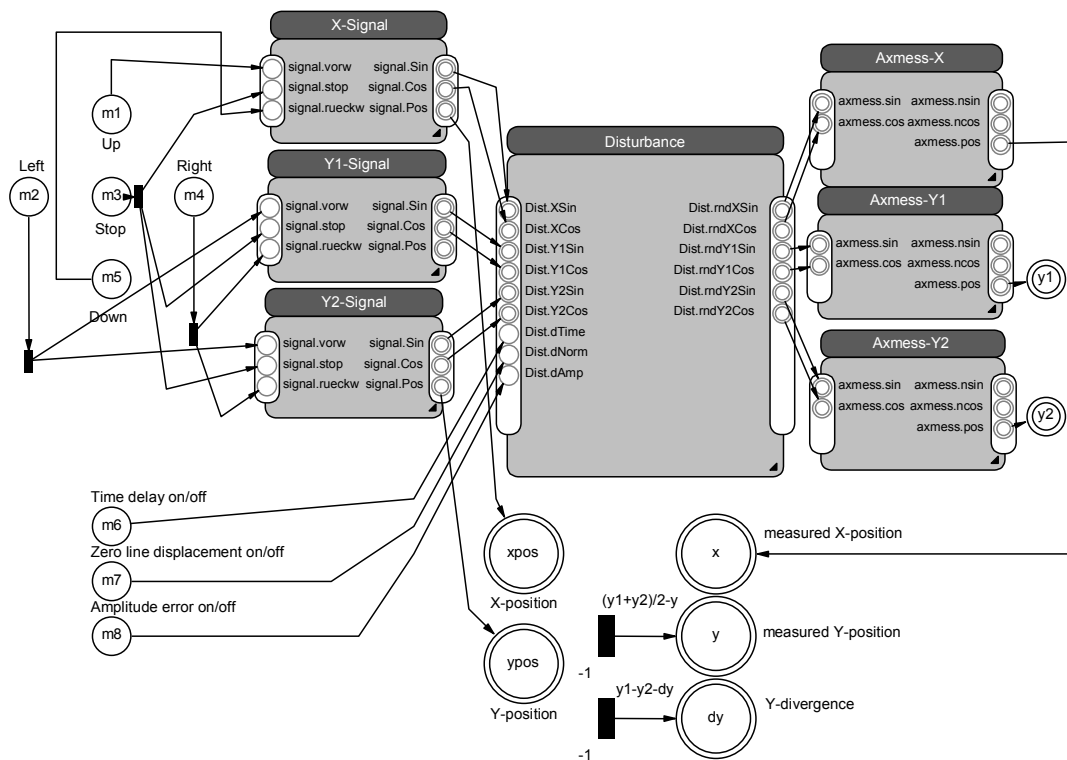


Fig. 6 Model of the entire system

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 average y-position and the divergence of the y-position arose as result of the net calculation.

3.4. System simulation

The tool “Visual Object Net++” [5] allows not only the modeling but also the simulation of systems described with Hybrid Dynamic Nets. During the simulation the firing of the transitions and the transport of the tokens are animated. The changes of the place values can be visualized by signal diagrams (Fig. 7).

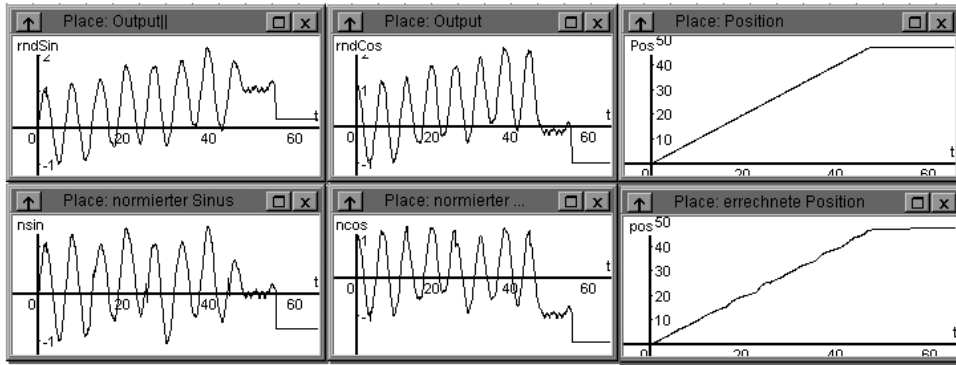


Fig. 7 System behavior with different disturbances

E.g., the middle top diagram in Fig. 7 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 homogeneous modeling of an embedded hybrid system. The object-oriented approach of the hybrid Petri net class used makes possible a clear modeling of complex hybrid systems.

Future things that have to be done are the extension and completion of the system model, and the integration of the modeling process in a complete design flow. Here we focus our future work on connecting our approach to other approaches related to hardware/software partitioning.

5. ACKNOWLEDGEMENT

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.: *Communication with Automata*. 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.: *Modeling Hybrid Systems Based on Modified Petri Nets*. PhD Thesis, TU Ilmenau, 1999. (In German)
- [5] Drath, R.: Hybrid Object Nets: An Object-oriented Concept for Modeling Complex Hybrid Systems. In: *Hybrid Dynamical Systems*. Third 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.