Implementing program generator for embedded systems based on SES model

Takaaki Koshiro

School of information,
Japan Advanced Institute of Science and Technology

February 15, 2001

Keywords: SES model, Embedded system, RTOS.

Today, the object-oriented approach which is widely used is one of the most progressed approach for software development. In the embedded system developments, we have to take their specific non-functional properties such as time and hardware constraints into consideration. Many development methods for embedded systems have been proposed, and OCTOPUS is one of such methods. OCTOPUS proposed implicit concurrent model and explicit concurrent model. We model a target system without considering asynchronous and synchronous factors in the implicit concurrent model. Then, in the explicit concurrent model, we model the target system with considering these factors from the constructed implicit concurrent model. However, how to implement software from the implicit concurrent model is not fully explained.

In this research, we adopt the SES model as design method. The SES model is an object-oriented method for time-critical embedded systems proposed by Aoki. In this approach, we construct SES model,then we implement software on real-time operating system from the design model. The SES model consists of process sequences. We refer to each of the sequences as a SES which is an abbreviation of synchronous execution sequence. This approach also provides templates for implementing software on RTOS from SES model. In this paper, We proposed a tool which automatically generates a program from SES model using template. Software on RTOS consists of a set of tasks and scheduling algorithm. These tasks are executed concurrently according to the scheduling algorithm. Our a program generator which generates an executable program which is executed on RTOS from a set of the SESs, a set of the tasks, and scheduling algolithm. As SES model did not have enough notion to implement such software, I extended SES model so that we can describe about tasks and schedule algolithm as follows.

1. I introduced the notation which specifies an aspect to assign SESs appearing in SES model to tasks

2. I introduced the notation which specifies an aspect to assign to schedule SESs

I found that we can automatically generate a program from SES model with these two aspects, and we can execute this program by implementing processes appearing in the SES model. We implemented a telephone system using this program generator. The constructed SES model contains eleven states and fifteen SESs. We succeeded in generating an executable program on RTOS. However, it is still difficult to describe complete specification to be defined. For example , SES model and the program generator can not deal with tasks comminications. These problems are taken as a future works.