Chapter 13
Implementation of a DES Environment

Gyorgy Lipovszki
Budapest University of Technology and Economics, Hungary

Istvan Molnar
Bloomsburg University of Pennsylvania, USA

ABSTRACT

In this chapter the authors describe a program system that implements a Discrete Event Simulation (DES) development environment. The simulation environment was created using the LabVIEW graphical programming system; a National Instruments software product. In this programming environment, the user can connect different procedures and data structures with "graphical wires" to implement a simulation model, thereby creating an executable simulation program. The connected individual objects simulate a discrete event problem. The chapter describes all simulation model objects, their attributes and methods. Another important element of the discrete event simulator is the task list, which has also been created using task type objects. The simulation system uses the "next event simulation" technique and refreshes the actual state (attribute values of all model objects) at every event. The state changes are determined by the entity objects, their input, current content, and output. Every model object can access (read) all and modify (write) a selected number of object attribute values. This property of the simulation system provides the possibility to build a complex discrete event system using predefined discrete event model objects.

GENERAL INTRODUCTION OF THE DES SIMULATOR

Implementation of a DES can be realized in different ways (Kreutzer, 1986). To simplify the procedure and to avoid difficulties related to computer science subjects (language definition, syntax and semantics discussion, translation, etc.), in this chapter the authors decided to remain focused on those issues, which are closely related to the DES and the simple implementation of the DES computational model. Doing so, the authors present a DES extension of the industry standard programming system LabVIEW.

DOI: 10.4018/978-1-60566-774-4.ch013

The LabVIEW DES simulation is the numerical computation of a series of discrete events described with the help of a set of blocks with given characteristics (the framework system of simulation processes, which can be described with discrete stochastic state variables).

- With the help of the framework system, a simulation of processes can be realized. The processes are driven by several independent, parallel events, between which information exchange is possible,
- Within the framework system, the linear, parallel and feedback connection of objects is possible in optional quantity and depth.
- The framework system is designed in an object-oriented manner. Both, the blocks with different characteristics serving the process simulation and the entities carrying information are designed in the same way.
- The programs prepared with the help of the framework system can be saved as subroutines, too. These subroutines can be reused in the simulation program; an arbitrary number of times with arbitrary input parameters.

The simulation framework operates in the following structure:

Why is it Necessary to Extend the LabVIEW Program System with Discrete Event Objects (DEO)?

The LabVIEW program system was developed for measuring, analysis and display of data generated in discrete time in equidistant (time) steps. It has numerous procedures, which make the measurement of data easier by using triggering technique based on a given time or signal level. For the data analysis, there is a wide scope procedure library at disposal, which makes it possible to realize detailed signal analysis either in time or in frequency range. As it can be seen, the program system is able to execute sample processing of continuous signals with the help of digital computers. It also contains procedures, which are able to handle elements of queues (creating queue objects, placing a new element into the queue, remove an element of a queue, and finally. deleting a queue object). These procedures and a few others, which help parallel programming (Notifier and Semaphore Operations, or Occurrences), are objects and procedures, which can be used to create (put together) a discrete event simulation model.

A new research and development has been started to make LabVIEW capable to create new object types, which make it possible to generate discrete event simulation models of arbitrary complexity in an easy way.

What is Needed to Create a Discrete Event Simulator (DES)?

Realizing a DES, there are a series of basic functionalities and elements, which must be implemented to ensure a modular program structure. When the basic elements were implemented, object-oriented data structures and object related methods were used, utilizing fully the possibilities and features of the host, the LabVIEW.

Simulation time and event handling

The most important step in a DES system is to establish the event objects (**Task**) with given data, the storage of the data according to given conditions and the removal of them from the system, when given events have already taken place. The individual event objects have to be properly stored and organized according to given conditions using another object (called **TaskList**). When different DES systems are implemented, different event lists (differing both in number and in tasks) can be used. In the DES system, realized in LabVIEW, there is only one such list, which contains all the events (events' data) of the simulation model in