GPSS EXTENDED CAPABILITIES AND MODIFICATIONS

J. W. Walton/L. W. Terry
Systems Simulations
The Boeing Company
Huntsville, Alabama

The following paper is a description of a Boeing Modeling Technique (BMT) System. The system was derived from GPSS II-B and contains some desirable GPSS/360 capabilities in addition to extended capabilities and modifications.

EXTENDED CAPABILITIES

Debug and Tracing Aids

Snap Block. Two additional blocks have been added to BMT to provide checkout and debug capability for users. A snap block feature has been implemented to give the user the ability, at the execution of the snap block, to print out the transaction chains (current, future, interrupt, user), block counts, clocks, savex cells, and statistics associated with queues, stores, facilities, and tables. Also the logic switches which are set at the time the snap is executed may be requested. The option to print any or all of these entities is given by coding a particular number in the proper field of the block.

Output Block. An output block, along with format control cards, also gives the user the ability to obtain statistical output and system information. It is very useful in obtaining report information as well as in debugging a model. A format control card is provided to tell the BMT assembler that a format follows. Columns 1-6 of this card contain the format number. Immediately following the format card, the modeler's title cards are placed. Following these format title cards, a card is found containing the names of the system variables to be printed out immediately after the titles. An end-format card is placed immediately after the system variable indicator card to indicate the end of this format. Whenever the user wishes to print these titles followed by the values of the system variables indicated, he merely places an output block in his model statements to be executed and associates a particular format with this output block by indicating the format number in Columns 19-24.

MODIFICATIONS

Interrupt Handling

Spie Macro. A spie macro is issued by the BMT Control Monitor to disable the normal control program interrupt handling routines and enables the user provided exit upon a desired interrupt occurring. All types of interrupts are enabled by the spie. The user's exit specified in the spie macro passes control to bring the output module into croe, if not already in core, and all simulation statistics accumulated to that point are printed. This enables the user not to lose his output if a system error occurs, e.g. an operation or addressing error. Also if a computational error occurs such as exponential overflow, the user still gets his results up to the point of the overflow occurring.

Synad Exit. The synad exit, which is taken if an asynchronous input/output error occurs, has been changed in BMT to be the address of a users specified program. This program automatically dumps core to enable the I/O error to be debugged and then passes control to the output module which prints all model statistics accumulated to the point of the error. After printing the accumulated results, control is passed to the assembler to process the next model. Therefore, the modeler is not deprived of the possibility of the results being useful even though an error occurred.

Allocation

Entities in BMT are allotted through the use of an allot card.

Allot Card. The allocation of entities in BMT is provided through an allot control card. An allot card may be written for each entity used, but if not, the assembler module will generate one. If the user's model uses more of a particular entity than he allots, the assembler will correct his error and allot the correct number used. If the user's allot card for a particular entity contains a number greater than the quantity he used, the assembler will allocate the number that is higher, in this case the number coded on the allot card. Entities may be coded either as absolute numbers or symbolically. The assembler places all numeric or symbolic names representing a particular entity in a pool for that entity and reassigns each a unique value. All entity allocation is done by means of these allot cards.