N89-12917

AUTOMATION SOFTWARE FOR A MATERIALS TESTING LABORATORY

Michael A. McGaw and Peter J. Bonacuse
NASA Lewis Research Center
Cleveland, Ohio 44135

A comprehensive software system for automating much of the experimental process has recently been completed at the Lewis Research Center's high-temperature fatigue and structures laboratory. The system was designed to support experiment definition and conduct, results analysis and archiving, and report generation activities. This was accomplished through the design and construction of several software systems, as well as through the use of several commercially available software products, all operating on a local, distributed minicomputer system (fig. 1). Experimental capabilities currently supported in an automated fashion include both isothermal and thermomechanical fatigue and deformation testing capabilities. The future growth and expansion of this system will be directed toward providing multiaxial test control, enhanced thermomechanical test control, and higher test frequency (hundreds of hertz).

Research Project Model

A model of a typical research project was developed by examining the process used by researchers in the course of conducting materials behavior research (fig. 2). The principal emphasis of the automation effort at the Lewis Research Center has been on supporting the formulation and conduct of experiments, the analysis of the resulting data, and the reporting of research progress.

Hypothesis formulation is an intensely creative (human) process and therefore is not easily subjectable to automation. Whether this will remain true in the future is a topic of fervent debate and will not be discussed here, save to say that automation tools can do much to support this creative process. An identical statement can be made for the conclusion formulation process.

Experiment Formulation and Conduct

A basic model of this process is given in figure 3. The researcher, attempting to prove a hypothesis, first formulates an experiment or set of experiments. Having a suitable description (generally symbolic in nature), a parametrization is made to fix the precise nature of the tests desired. In this way all control parameters and measurement variables are defined, as well as their strategies. At this point these requirements must be translated into the form of a computer program in order that the desired test can be executed. Generally, this has meant the creation of unique programs, a consequence we are seeking to minimize.

Our present capabilities consist of a very general uniaxial, isothermal test-creation and control capability, as well as a number of unique programs for conducting thermomechanical tests. The unique programs are generated in the usual sense; the general development process and supporting tools are described in figures 4 and 5. A program is developed and tested (to the maximum extent possible) on the host processor, where a full complement of tools are present to support such activities.

When the requirements of an experiment fall within the uniaxial, isothermal test catagory, an automated process is used. A set of command waveform and data acquisition requirements are generally related: a materials test command waveform and data acquisition requirement can always be decomposed into "blocks" where certain sequences of command and data acquisition are fixed in relation to one another and usually repeated as a block for a finite number of interations. These "blocks" differ from one to another based on changing data acquisition needs, control mode differences, command waveform differences, or a combination of these. At times, these blocks are concatenated, forming yet another "block". Because of this, a capability to nest a series of "blocks" exists. These test control requirements are implemented through the creation of a "control tree", an ordered path connecting elements, or nodes, of operations. There are five types of nodes, each possessing a set of usage rules for implementing its functions. A typical test, the constant-amplitude strain-controlled fatigue test, and its control tree are given in figure 6. As can be seen, node types exist for expressing repetition, command waveform character, and data acquisition parameters and strategies. This structure effectively provides for the creation of virtually any kind of test - it is adaptable to include other capabilities as well. In fact, the extensions of multiaxial test control, computed variable control and thermomechanical test control were incorporated into the basic data structure and program design and will be implemented in the future.

Once the control tree is generated, the actual experiment can be conducted. This is accomplished through a program which interprets the control tree and effects the operations called for. This multitasked and interrupt-driven program performs in real-time. The interface provides for the usual controls during execution: begin the test, pause, resume, status request, and abort. At this time, it is not possible to alter either the tree structure nor node (test) parameters dynamically other than through the aforementioned interface. This capability is latent in the basic design, however, and will be implemented in the future. A number of programs (an environment) support the creation and execution of tests using the control-tree approach (fig. 7). Support functions include capabilities to:

(a) Uniquely describe a given uniaxial materials testing system in terms of transducer complement, calibration data, etc.
(b) Create control trees from parametric descriptions of desired tests.
(c) Conduct materials tests from control trees previously generated.
(d) Produce formatted test reports from the data acquired during the execution of an actual test.
(e) Provide interface mappings between a given computer system and a given servohydraulic testing system.
(f) Provide general communications capabilities among the lab computer systems, including file transfer and virtual interface capabilities.

A more complete description of this environment as well as a thorough discussion of the control-tree concept will appear in a future paper.

Data Analysis Cycle

A basic model of this process is given in figure 8. The usual procedure for analyzing experimental test results involves organizing the primative data, deciding how to analyze the results (task definition), and generating (unique) programs for the analysis. This procedure is quite lengthy and cumbersome, especially within the context of software available to automate much, if not all of this process. We have elected to use commercially available software systems for this portion of the research process; our choice of software includes a relational data base management system (residing on the host processor), and an integrated graphics/statistics/modeling program (residing on personal computers). Key features of these systems are shown in figures 9 and 10.

The manner in which these system are used in our laboratory is characterized by two separate environments: the relational DBMS resides on the host computer system and is the primary organizing and archiving system. Data acquired in the laboratory are loaded into this system after each experiment (or set of experiments). The interactive analysis system, residing on personal computers distributed throughout the building, accesses the data through the DBMS (all systems are networked over the Lewis CATV local area network). The user need not physically handle the data at all - electronic transfer of compatible data files throughout the laboratory computer system is possible. We have found this environment to be both powerful and efficient.

The final portion of the analysis process is reporting. For this function, we are using a variety of work processing and text editing programs: each individual is using his or her choice. The fundamental characteristic of this process, however, is the ability of these programs to exchange text files with the secretarial word processing systems.

CONCLUSION

The system in use for conducting research at the Lewis Research Center's high temperature fatigue and structures laboratory has been described. Those areas of the research process that could be automated in an effective manner through the use of commercially available software systems were. For those areas not effectively automated through commercially available systems, custom systems were developed. A key characteristic of the test conduct portion of the described system is the notion of a control tree, the principal means of describing and executing materials tests under computer control. The environment supporting the creation and execution of control trees was described. Finally, future extensions planned for enhanced control capability were described.

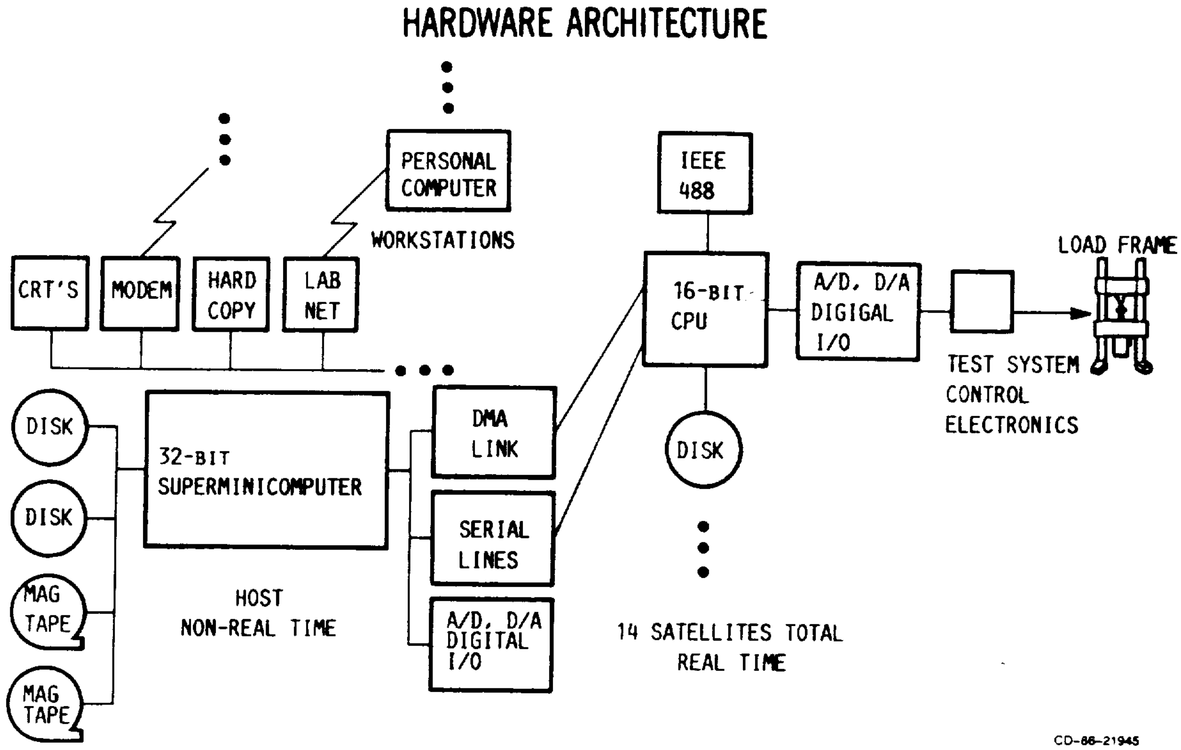

Figure 1

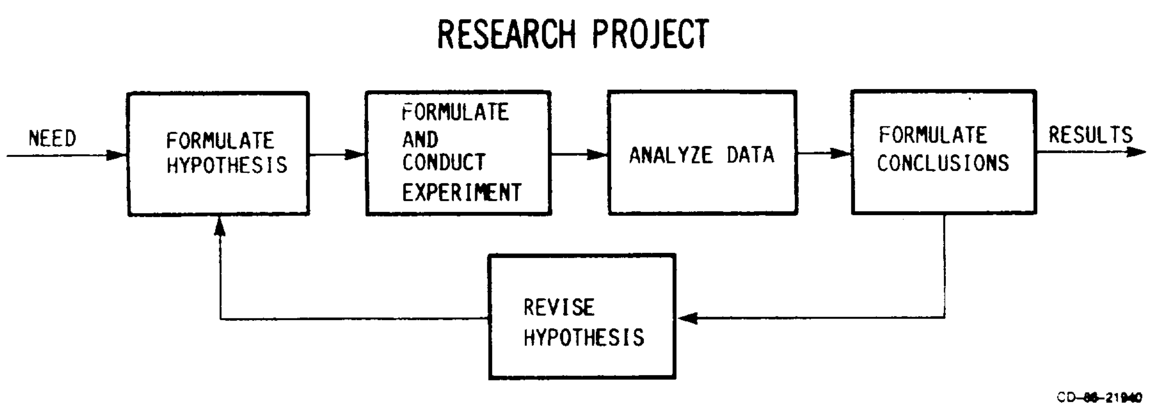

Figure 2

FORMULATION AND CONDUCT OF EXPERIMENT

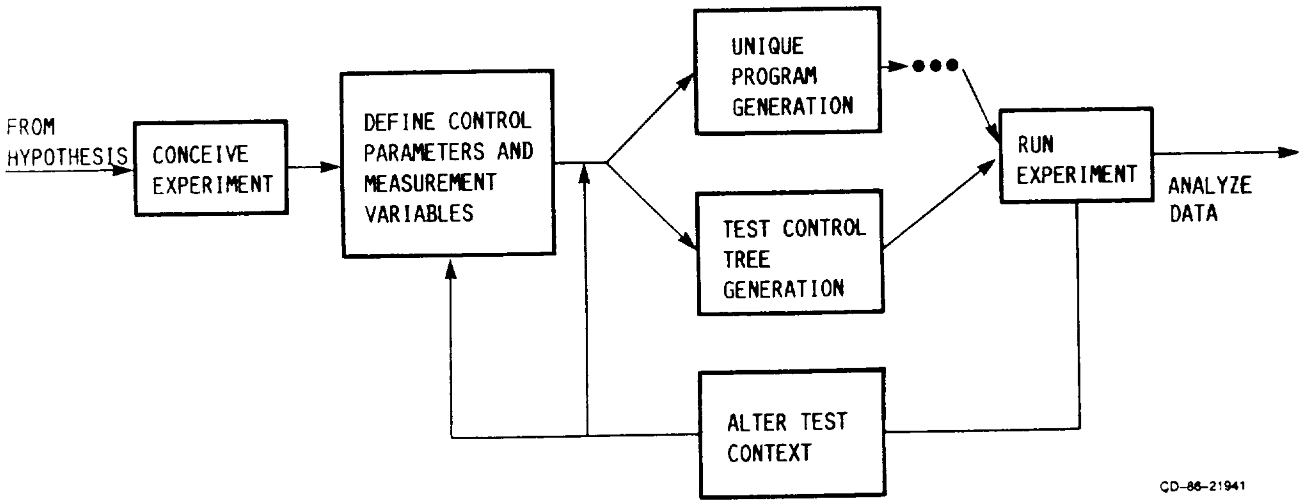

Figure 3

PROGRAM DEVELOPMENT CYCLE

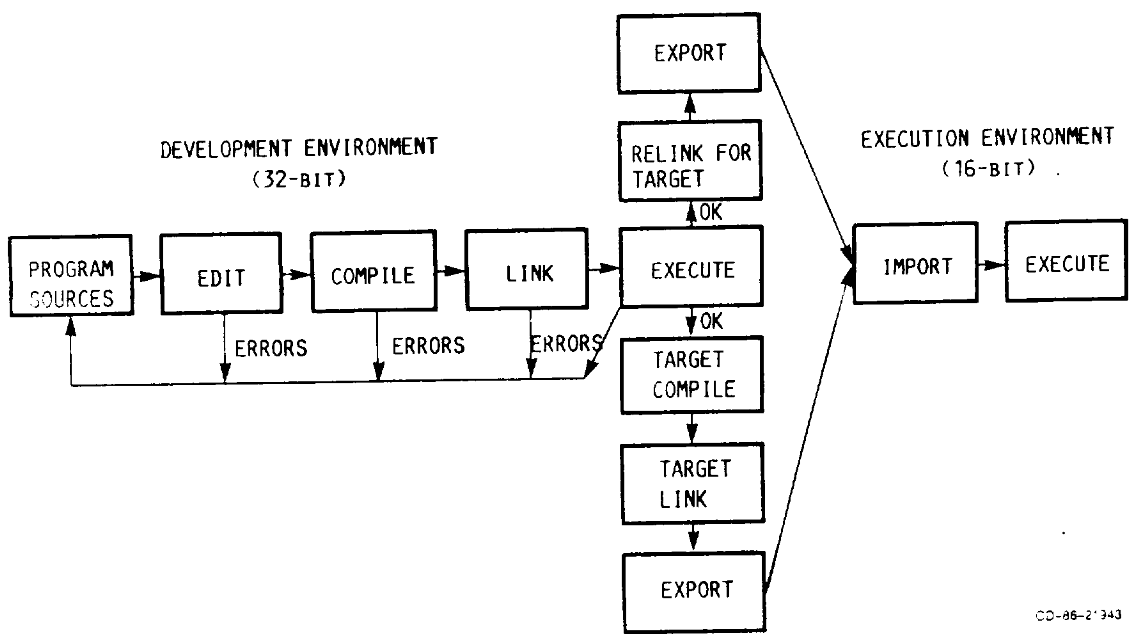

Figure 4

PROGRAM DEVELOPMENT TOOLS

- SOURCE EDITORS
- LANGUAGE PROCESSORS
  - ADA
  - PASCAL
  - FORTRAN-77
  - BASIC
  - ASSEMBLER
- LINKER
- MISCELLANEOUS TOOLS
  - CONFIGURATION CONTROL UTILITY
  - SYMBOLIC DEBUGGERS
  - LIBRARY EDITOR
  - FILE EDITOR
  - ETC.
- LIBRARIES
  - SENSOR INPUT/OUTPUT
  - MATHEMATICS
  - STATISTICS
  - GRAPHICS

CD-86-21939

Figure 5

TYPICAL FATIGUE TEST

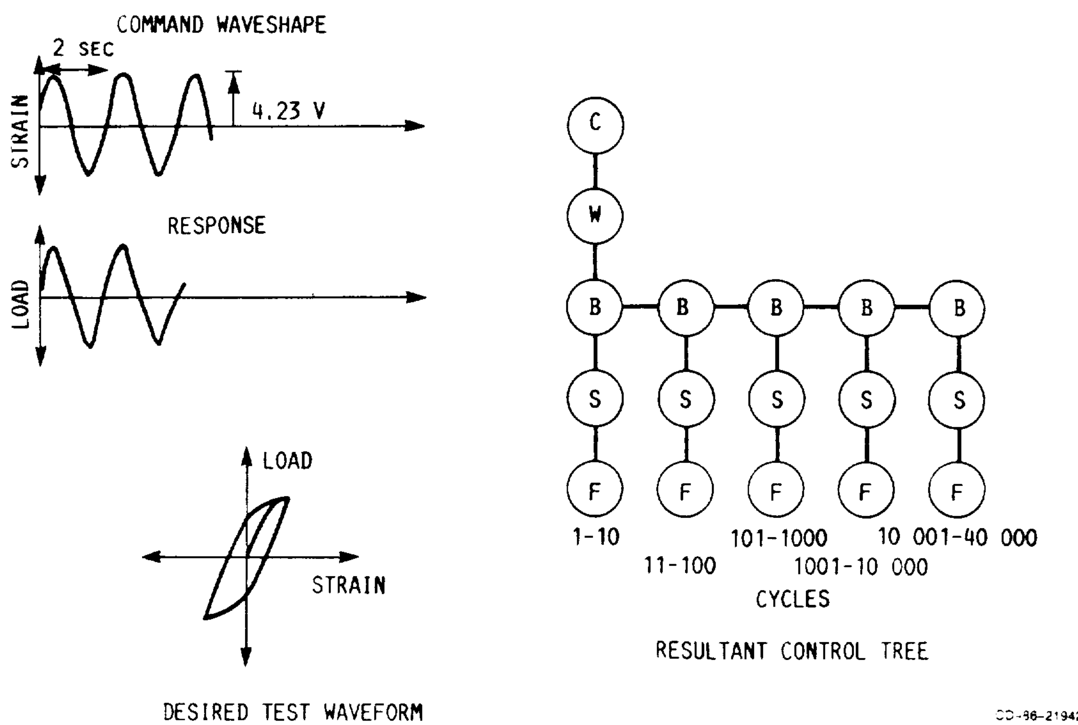

Figure 6

CONTROL-TREE SUPPORTING ENVIRONMENT

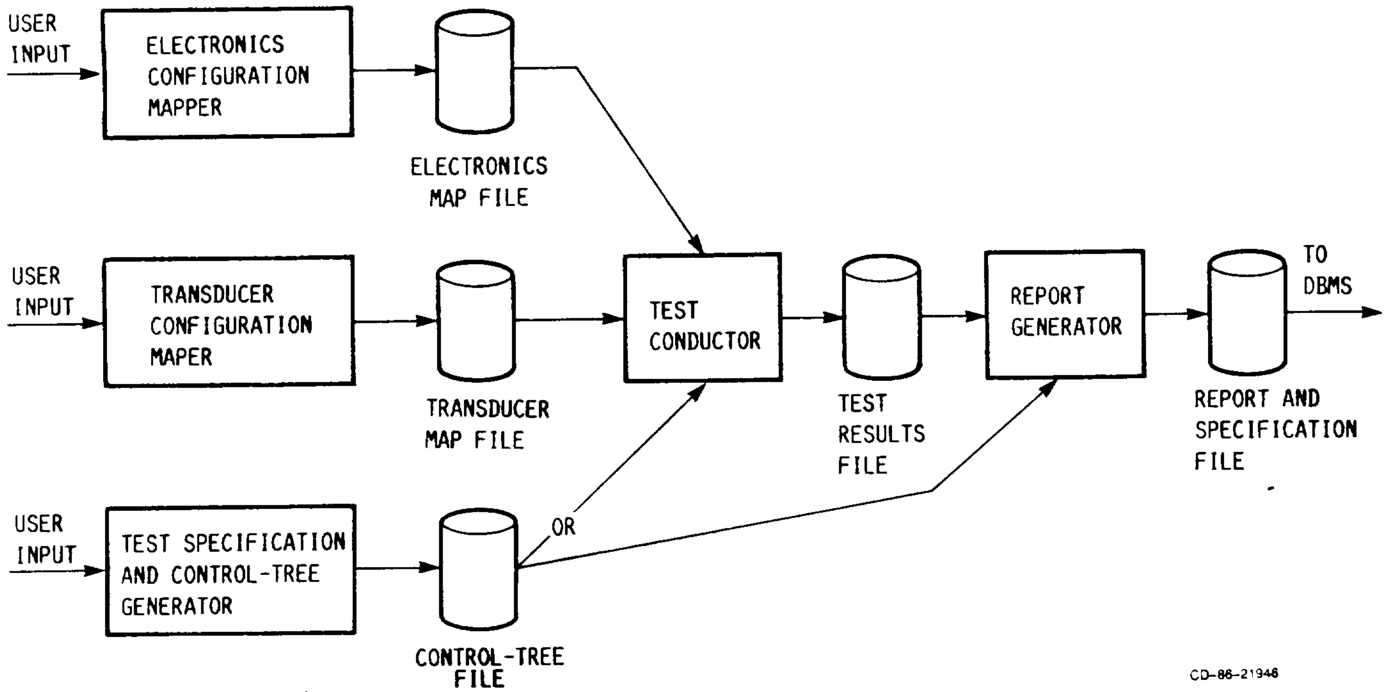

Figure 7

DATA ANALYSIS CYCLE

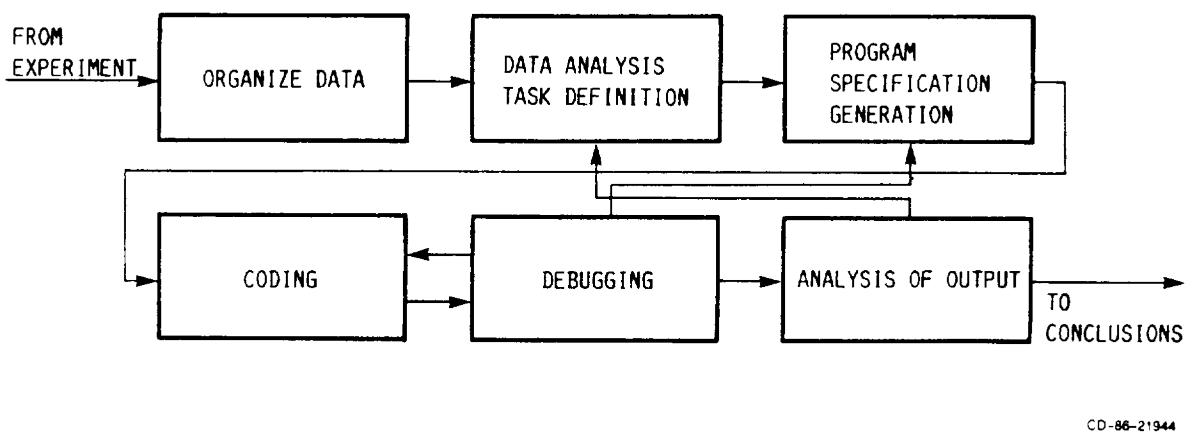

Figure 8

RELATIONAL DBMS

- MULTIUSER DATA ACCESS, SHARING
- RELATIONAL; USERS VIEW DATA AS COLLECTIONS OF TABLES
- SQL DATA MANIPULATION/DEFINITION LANGUAGE
- INTERACTIVE AND PROGRAM CONTROLLED ACCESS

CD-86-21938

Figure 9

INTERACTIVE ANALYSIS ENVIRONMENT

- DATA ENTRY AND RETRIEVAL FUNCTIONS
- DATA TRANSFORMATION AND ANALYSIS
- GRAPHICS: COLOR, 2- AND 3-D, DIGITAL PLOTTER
- CURVE FITTING
- STATISTICS
- ANALYTICAL MODELING
- INTERACTIVE OR PROGRAM-DRIVEN

CD-86-21937

Figure 10