A USER-FRIENDLY, MENU-DRIVEN, LANGUAGE-FREE LASER CHARACTERISTICS CURVES GRAPHING PROGRAM FOR DESK-TOP IBM PC COMPATIBLE COMPUTERS

by

Glenn Klutz
Associate Professor
Department of Technology
Elizabeth City State University
Elizabeth City, North Carolina 27909

A menu-driven language-free program which graphs the interrelationship of the many physical parameters of lasers and laser materials for use by laser researchers and engineers is an expressed need or requirement of the Environmental Sensors Branch of the Flight Electronics Division, NASA Langley Research Center.

The branch has already established a facility that uses collected data and feeds it into mathematical models that generate improved data arrays by correcting for various losses, base line drift, and conversion to unity scaling. These developed data arrays have headers and other identifying information affixed and are subsequently stored in a Laser Materials and Characteristics data base which is accessible to various users. The two part data base: absorption - emission spectra and tabulated data, is developed around twelve laser models. The tabulated section of the data base is divided into several parts: crystalline, optical, mechanical, and thermal properties; absorption and emission spectra information; chemical name and formulas; and miscellaneous.

This summer's project was to develop a menu-driven, language-free graphing program for use with this data base. The final version of the graphing program will reduce and/or remove the requirement that users become competent FORTRAN programmers and the concomitant requirement that they also spend several days to a few weeks becoming conversant with the GEOGRAF library and sequence of calls and the continual refreshers of both. It is the consensus within the Branch that researchers time is more important being spent in their specific research specialties.

The work this summer included becoming thoroughly conversant with or at least very familiar with GEOGRAF by GEOCOMP Corp. GEOGRAF is a FORTRAN callable graphics library that helps plot to screen, printer, or plotter during execution or to a disk file during execution for actual plotting at a later time. In GEOGRAF the programmer instructs the plot device, be it screen, printer, or plotter, with FORTRAN call statements rather than through the symbolic language required by the graphics device. Learning the FORTRAN language from scratch takes the average person several weeks to a couple of months. Learning to actually run each of the subroutines in the GEOGRAF library takes a good block of time. Also, after learning the FORTRAN language and the subroutines, each time a researcher wanted to graph a new set of data he/she

would have to spend a block of time refreshing themselves on both the language and the library routines as well as sequence of calls as has been the experience of this program author.

The development of the graphing program involved trial runs of the various callable library routines on dummy data in order to become familiar with actual implementation and sequencing. This was followed by trial runs with actual data base files and some additional data from current research that was not in the data base but currently needed graphs. These actual runs provided the knowledge as to which actual subroutines would need to be included in the menu-driven program to provide for graphing all files from the data base. After successful runs, with dummy and real data, using actual FORTRAN instructions steps were undertaken to develop the menu-driven language-free implementation of a program which would require that the user only know how to use microcomputers. The user would simply be responding to items displayed on the video screen. To assist the user in arriving at the optimum values needed for a specific graph, a paper and pencil check list was made available to use on trial runs.

Visualizing the various problems that can be encountered by neophyte programmers has proved to be a great challenge to the author of this program. For instance, since FORTRAN programs will crash if alphabetic characters are supplied when numeric data are required, reprogramming was required so that all responses be in character format and the computer then convert to the required numeric data required by the call instruction. Near the end of this summer's tenure, other areas that had not been thoroughly visualized from a beginning programmer's viewpoint began to appear. During the seventh week while talking with various researchers and making special runs on data they had collected, it became evident that methods would need to be incorporated for them to massage data before actual processing by the graphing program. This included the capability to invert the graph, to average groups of points for plotting, or to generate a paralleling array of data to serve as the other axis when their data collection system had only provided for one array of data.

There are several areas that need additional investigation as indicated above. One includes the possibility of a generic program to take any data file whether in the data base or not and the user being able to respond to items which in turn will construct correct call statements. A second option would be to investigate the utilization of the on-order FORTRAN compiler which contains integral plot commands. A third area would be to investigate the possibility of loading these onto a host computer for the Division or even Center wide.

It is expected that this graphing program will provide an added dimension to the research accomplishments of various researchers involved in laser research and that continued efforts on this project can expand its capabilities and maybe even be expanded to different aspects of research underway here at the Center.