

DESIGNING WITH COMPUTERS  
AT  
LAWRENCE BERKELEY LABORATORY \*

John S. Colonias  
Lawrence Berkeley Laboratory  
University of California  
Berkeley, California

NOTICE  
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, sub-contractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

ABSTRACT

This paper explores the application of digital computers to the solution of engineering problems relating to accelerator design as practiced at the Lawrence Berkeley Laboratory. First, the existing computer environment is described in terms of hardware and software available for direct communication between the engineer and the computer, then some examples of useful programs are outlined not so much to advertise their extensive use, but rather to show the ease of their utilization and the manner by which the communication between the machine and the designer is conducted. The author believes that the practicing engineer looking into conventional computer installations sees programmers using a variety of special esoteric languages, and his first introduction to computer languages usually convinces him that communicating with a computer is an awkward and constraining process. Thus he is persuaded to ignore the computer installation unless he is absolutely forced to use it.

We hope that this paper will dispel some of these ideas and convince engineers that they can communicate with the computer in ordinary English and mathematics, rather than in intermediate artificial languages, and that they can benefit from the attractive features offered by such a cooperative adventure.

INTRODUCTION

The past few years have been years of exciting growth in the field of computations. Digital computers have increased their computational speed by orders of magnitude, mathematical analysis has experienced dramatic advances, time-sharing has provided the important (economic) capability of on-line interaction, and computer users have increased in number and sophistication. The net effect of this rapid growth is that cost per computation has decreased rapidly, so that we can afford to be less efficient in computation utilization and concentrate instead on the development of computer programs that can be used interactively with "computer assisted instructions," relieving the user from the drudgery of having to learn programming tasks. In this paper we restrict ourselves to the interactive aspect of designing with computers since it offers very attractive features particularly suited to engineering calculation. The examples described in this paper indicate the broad range of interests and capabilities that computer graphics offer.

COMPUTER ENVIRONMENT

HARDWARE

The Lawrence Berkeley Laboratory facility is equipped with a CDC 7600, a CDC 6600 and a CDC 6400 computer system, plus an array of temporary and permanent storage devices, arranged in the fashion sketched in Fig. 1.

The system provides for handling large volumes of output by producing microfiche at 48X or 24X, or microfilm output in 35 mm or 16 mm strip form. With 48X reduction up to 270 pages of line printer output can fit on a single 4X6 inch microfiche. Graphic and alphanumeric output are both available.

For communication with outside users, the computing facility is equipped with telephone lines to communicate with remote terminals. Thus, a remote terminal can be connected to the system simply by telephoning a dial-up connection.

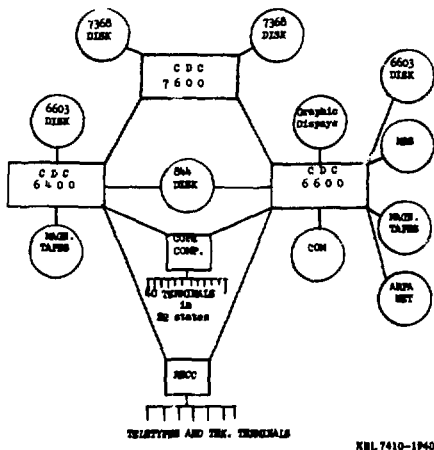


Fig. 1. Simplified block diagram of LBL computing facility.

In addition to interfacing with remote terminals for interactive computing and remote job entry, the Berkeley installation is also equipped with a remote batch interface system (UC Cope controller). Users who have remote batch stations at their own facilities may submit batch jobs and receive printed output directly.

The operating systems at the LBL facility evolved from original CDC systems. The BKY 7600 system has grown out of the CDC SCOPE 1 system, and the 6000 system has its roots in the original CHIPPEWA operating system. Both have been tailored to fit the particular needs of the Berkeley installation, and are maintained by a staff of system programmers at the Laboratory.

SOFTWARE

For interactive and remote communications with the computers, the following software packages have been developed.

1. SESAME
2. PTSS (People's Time-Sharing System)
3. GRAPHIC
4. PICASSO

1. **SESAME.** This is the main BKY teletype system, consisting of many subsystems with which the user may create and edit files, edit files already in the system, copy files to and from the

"data cell", execute small pseudo-Fortran programs, doing all reads and writes at the remote terminal or use the terminal as a sophisticated desk calculator (see example in Fig. 2).

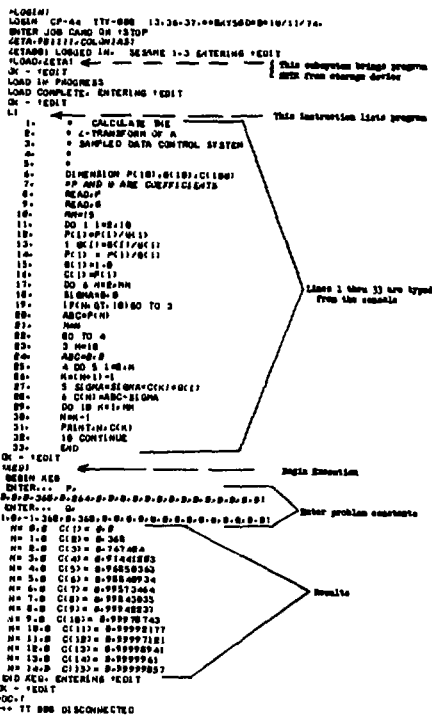


Fig. 2. Example of SESAME use, showing the ease of use of various subsystems.

2. **PTSS.** This is a general purpose system for entering, running and debugging programs from a remote console (CDC 252 display or Tektronix 4002). It consists of two principal parts: A "text-editor" which allows the creation, deletion, or modification of programs, and an interactive "debugger" which helps the user debug his programs, by allowing him to examine core, examine the compilation listing, the central registers, etc. When an error occurs, while the user is interacting (either editing or debugging), PTSS occupies less than 6K of core. When the user

wants to compile and run the program, PTSS increases its field-length to accommodate the compiler.

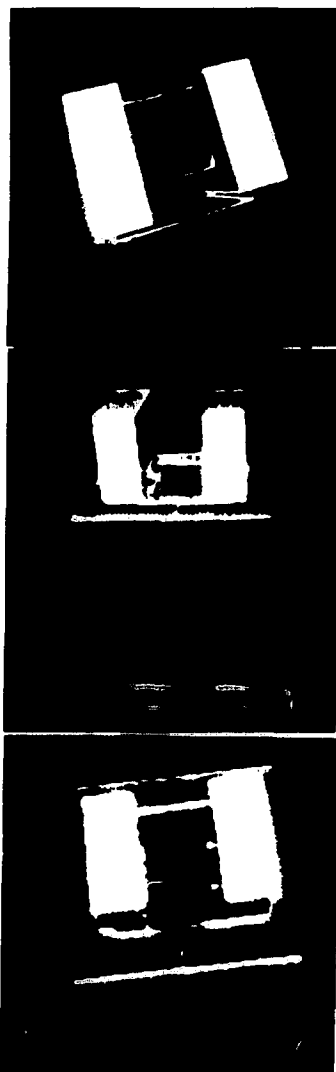
One main difference between PTSS and SESAME is that the latter allows one to quickly and easily enter, debug, or execute simple programs using a limited subset of FORTRAN.

3. GRAPHIC. This is a set of low level routines to allow device independent graphic displays. This independence is achieved by having the routines for each device, interpret only those parameters supported by each device. Devices to be ultimately supported, include CDC 252 consoles, GT40 terminals, Tektronix 4010/12, ARPA net protocol, Stromberg Carlson 4460 and CALCOMP. The package allows the user to clear the screen and plot points, lines and character strings for on-line or off-line devices.

4. PICASSO. This is a general interactive graphics program<sup>2</sup> for constructing a complicated data structure by the time-proven method of drawing pictures. The program can serve as a "front-end" module for any model-building program which interprets and analyzes a user-defined data structure. In many applications the picture itself is the desired output, as in printed circuit design layout, or flowcharting of procedures and program. For these PICASSO is self-contained. The philosophy of the program is to allow the user to create and edit diagrams, which are treated as entities. The diagrams may consist of basic elements (lines and/or text) or they may be built from other (previously defined) diagrams as well. The recursive nature of the data structure provides for the construction of models based on user-defined components, such as logical diagrams, mechanical engineering drawing physical plant layout, optical systems design, or almost any problem where the model is easier to draw than to encode in digital form by hand. Figure 3 shows some successive views of a window-frame magnet rotated for three dimensional projection.

#### DESIGN APPLICATIONS

The complexity of modern accelerators and beam lines is such that the required standards of beam quality and operational flexibility cannot be achieved without the use of a computer.



XBB 7410-7079

Fig. 3. Example of PICASSO use, showing a three dimensional view of a bending magnet.

Computer programs to simulate various aspects of accelerator design have been developed by many laboratories, and are being extensively used to

perform their specified functions. Here, we will touch only some of the areas emphasizing the ease of use and the manner by which the communication between the machine and the designer is conducted.

## MAGNET DESIGN

Perhaps the most expensive single component of a particle accelerator is the magnet whose field shape determines the function of the magnet in the accelerator. The precision required in the design of these magnets, as well as the high cost of modern accelerators, demand the study of many possible models before construction of the actual magnet begins. Therefore computer programs like LINDA and TRIM that simulate the mathematical model of a two-dimensional magnet with a variety of boundary conditions are very useful.

Specifically, program LINDA<sup>3</sup> uses a combination of scalar and vector potentials applied to a non-uniform rectangular grid of about 10000. It produces accurate results, allowing the experimenter to perform perturbation studies on the pole face profile with very fine detail.

Program TRIM<sup>4</sup> also calculates magnetic fields for two-dimensional magnets with arbitrary geometry and general boundary conditions. Here the problem space or universe consists of an irregular triangular mesh of about 4000 points in which the magnet geometry is superposed. TRIM users vector potential throughout the mesh. Both programs are large and involved, but once the user has learned a few simple rules, these programs are easy to use.

One of the attractive characteristics of these programs is that they have been adapted for on-line interactive use. Specifically, program TRIM displays the problem mesh on the screen and allows manipulation of data to conform with the requirements of the problem. The man-machine communication is done in a manner meaningful to the user, with frequent instructions guiding him to the next step. Upon examination of the displayed mesh, the engineer corrects or alters any part of the mesh that does not agree with the input specification. He continues this procedure of altering and viewing the corresponding change until he is satisfied that the geometry is free of errors, then he terminates the on-line session and the

resulting final mesh is submitted to the computer for the off-line calculation of the magnetic field distribution. Figure 4 shows successive stages of a mesh design, while Fig. 5 shows a complicated mesh structure and the resulting field distribution for a preliminary design of a superconducting dipole magnet.



Fig. 4. (a) Preparation of mesh on the console. (b) Final generated mesh.

At present we are in the process of converting the TRIM program to use a Tektronix 4012 terminal and at the same time reduce the core requirements of the program from 150000<sub>g</sub> locations to less than 50000<sub>g</sub>.

## BEAM TRANSPORT DESIGN

The design of beam transport systems involves the guiding of a beam of particles from the accelerator source to the target. This transport system consists of various beam transforming elements whose behavior is modeled by pro-

gram is written in such a way that the CRT console is used not only as a means of displaying data but as a suggestive medium, guiding the user throughout the design by either informing him quickly of the overall characteristics of the system or by displaying warning messages to avoid invalid conditions. Thus errors are detected easily and loss of time because of erroneous data is effectively minimized.

Figure 6 shows various phases of the development of a beam transport system. The rectangles represent a magnet while the size of the rectangle reflects the magnet length and aperture. Below the beam envelope, the cumulative transfer matrix is displayed along the beam matrix. Both these matrices include useful information pertaining to the

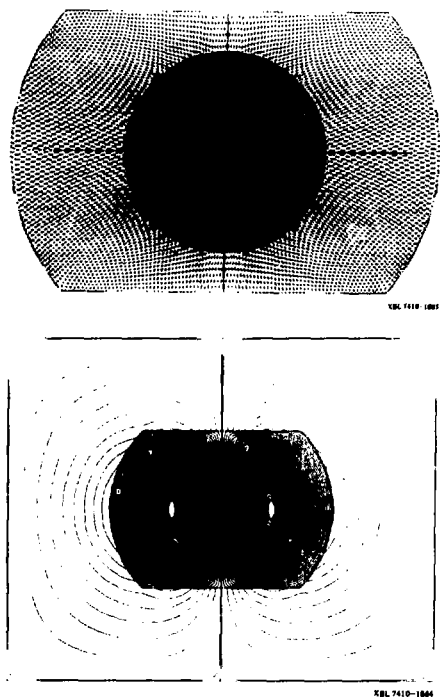


Fig. 5. (a) Calcomp plot of the generated mesh for a superconducting dipole magnet. (b) Calcomp plot of the resulting field distribution.

gram TRANSPORT.<sup>5</sup> This program, used interactively, occupies 60000 words of memory and provides means for describing an injected beam and a magnet system. The beam is specified by a six-dimensional ellipsoid, while the magnet system includes most of the commonly used beam elements such as bending magnets, quadrupoles, sextupoles, drift spaces, etc.

The program is versatile and allows for a variety of constraints to be applied to the system. Parameter fitting is also possible, allowing the experimenter to specify the parameters he wants to adjust, such as length, gradients, or apertures; and the program computes the first order correction by a least-squares method.

The outstanding feature of this program is its interactive aspect. The pro-

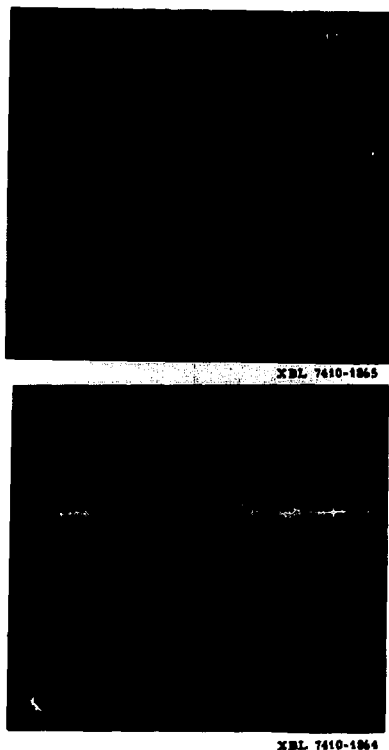


Fig. 6. (a) Beam envelope and phase space projection. (b) Ray trace beam envelope and phase space projection at different point and beam line.

projection of the beam ellipsoid on the coordinate axes, the tilt of the ellipse, dispersion, etc. The phase space ellipses at various points along the trajectory may be displayed next to the matrices.

The interaction between user and machine consists of altering various parameters and observing the corresponding change, or modifying the convergence conditions demanded to produce a more sophisticated design.

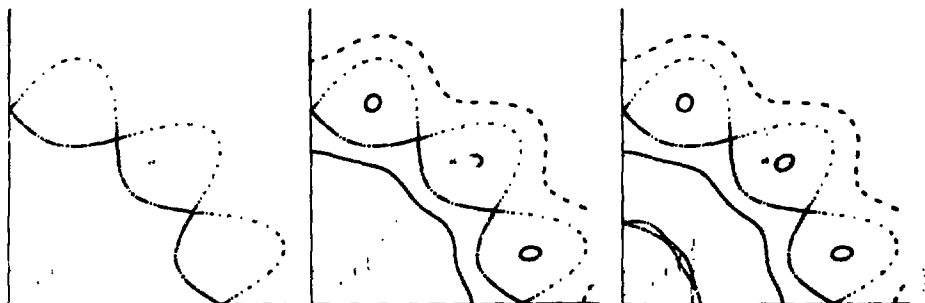
### BEAM-BEAM INTERACTION

This program<sup>6</sup> allows the theoretician or the storage ring designer to investigate the beam-beam interaction effect of a strong beam of charged particles a test particle as well as the effect of moving the beam through the ring to the next interaction region. The program accepts various storage ring parameters such as the shape and intensity of the strong beam

and the position and slope of the test particle. As the beam interacts the test particle the values of the new position and slope are plotted. As the process progresses an orbit is traced in the phase space,  $(x, x')$ , ( $x$  = position of particle,  $x'$  = slope) that shows the position and slope of the test particle. (See Fig. 7).

This visual representation of the effect of the beam-beam interaction allows one to find parameter values that give stable phase plane plots (i.e., bounded), and thus the test particle stays within the physical aperture of the system. It also shows when the test particle is unbounded and thus is lost from the physical system.

The curves shown in Fig. 7 were made on the Tektronix 4012 terminal and represent phase plane orbits traced for various initial starting condition of the test particle. The dialog between man and machine is shown below the plots,



```
THIS IS QMPTTY. TYPE START TO EXECUTE.
**
START PROGRAM
```

```
TYPE NO. SECTIONS
```

```
** TYPE SIGN(1)
```

```
TYPE TUNE
```

```
TYPE DELTA-HU/SECTOR (MAX.)
```

```
TYPE BETA
```

```
TYPE DATA
```

```
FOR EXP., TYPE 4 -- FOR SERIES, TYPE NO. OF TERMS
```

```
** TYPE SIGN(1)
```

```
TYPE SIGN(1) SIGN = 3.927100 SMALL ANGLE TUNE = .000000
```

```
TO CONTINUE WITH THESE DATA, TYPE 1 -- OTHERWISE 0
```

```
**
```

```
TYPE INITIAL X & Y (F-FORMAT ONLY, 2 LINES)
```

```
TYPE INITIAL
```

```
**
```

```
X = 0. Y = 2.9710000000000000 Y = 0.
```

```
TO CONTINUE WITH THESE DATA, TYPE 1 -- OTHERWISE 0
```

```
**
```

```
TO ADD NOISE, TYPE 1 -- OTHERWISE 0
```

```
TYPE NO. OF PRINT STOPS
```

```
TYPE REVS/PRINT
```

```
FOR TTY OUTPUT, TYPE 1 -- FOR PLOT, TYPE 0
```

```
**
```

```
TYPE XMIN, XMAX
```

```
TYPE YMIN, YMAX
```

```
**
```

```
XMIN = 0. YMIN = 0.
```

```
XMAX = 4.0000000000000000
```

```
YMAX = 4.0000000000000000
```

```
TO PLOT AREA, TYPE 1 -- OTHERWISE 0
```

```
**
```

```
X Y
```

```
0 0.9710000000000000
```

```
500 0.0010000000000000
```

```
**
```

```
N = 500, TYPE ADDIT. NO. OF PRINT STOPS
```

```
1000
```

```
TYPE REVS/PRINT
```

```
**
```

```
1500 1.5174000000000000
```

```
**
```

```
N = 1500, TYPE ADDIT. NO. OF PRINT STOPS
```

```
1000
```

```
TYPE REVS/PRINT
```

KEL 7410 1043

Fig. 7. Phase plane orbits traced with various initial conditions of the test particle.

and consists of question and answer conversation in which the computer asks pertinent questions about the type of problem that it is asked to compute. The structure seen in these plots is interesting to those who study the properties of non-linear mapping. It can also be suggestive to the theoretician trying to understand the mechanism that causes the actual physical beam blowup seen in real storage rings.

Here, as in the previous examples, the interaction between man and machine is not only desirable but mandatory if the behavior of the system to a variety of external conditions is to be properly investigated.

### CONCLUSIONS

We have attempted to show the manner in which interactive graphics are being applied to some applications pertaining to accelerator design. We also showed that these programs are written in a way that requires a minimum of computer experience. However, no matter how clever the programmer is, or how well written these programs are for these uses, it is mandatory that he have some degree of "computer literacy." The user must also be willing to accept as perfectly natural the use of a computer to amplify his intellect, just as he now accepts as perfectly natural and desirable the use of machinery to extend his physical capabilities. Thus, he should not hesitate to perform elementary computations until he gains the confidence to perform the more complex tasks.

### REFERENCES

1. BKY Users Handbook, Lawrence Berkeley Laboratory Rept. UCID-3409.
2. D. M. Austin and H. H. Holmes, PICASSO: A General Interactive Graphics Modeling Program, Lawrence Berkeley Laboratory Rept. LBL-580 (1972).
3. J. S. Colonias, Particle Accelerator Design: Computer Programs (Academic Press, New York, 1974) p. 39.
4. J. S. Colonias, TRIM: A Magneto-static Computer for the CDC 6600, Lawrence Berkeley Laboratory Rept. UCRL-18439 (1968).
5. A. C. Paul, TRANSPORT: An Ion Optic Program, LBL Version, Lawrence Berkeley Laboratory Rept. LBL-2697 (1974).
6. L. J. Laslett and E. R. Close, Personal communication (1974).

\* This work was performed under the auspices of the U. S. Atomic Energy Commission.