SANDIA REPORT

SAND89–1674 • UC–600
Unlimited Release
Printed September 1989

Automated Glass Fiber Drawing

William L. Smith

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550
for the United States Department of Energy
under Contract DE-AC04-76DP00789

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
PO Box 62
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from
National Technical Information Service
US Department of Commerce
5285 Port Royal Rd
Springfield, VA 22161

NTIS price codes
Printed copy: A02
Microfiche copy: A01

Sand #89-1674
Printed September 1989

Distribution Unlimited Release
UC-600

Automated Glass Fiber Drawing

by

SAND--89-1674

DE90 000894

William L. Smith
Surface Science

Sandia National Laboratories
Albuquerque, New Mexico

Abstract

The formation of pristine silica fiber sections in an ultra high vacuum environment requires the critical control of production parameters. Glass temperature and fiber draw force must be precisely controlled to produce favorable results. Process control is achieved by automating the operation of a high power CO_2 laser and the acquisition of data with a dedicated microprocessor. This combination, with the integration of ocular control through the application of digital image processing techniques, has subsequently led to a consistent and reproducible means of forming high quality pristine glass fiber test sections on the order of 25 μm to 50 μm in diameter. Fibers formed is this manner are tested to failure (in situ) by applying loads at various rates in controlled environments. This report presents the techniques and devices used to develop this automated fabrication and testing system.

This work was performed at Sandia National Laboratories and supported by the Department of Energy under contract number DE-AC04-76DP00789.

Introduction

Silica based optical fibers are rapidly assuming a pivotal role in advanced energy and defense related technologies. In many cases, the reliability of the fiber optic system is controlled by the mechanical properties of the silica glass fiber. As part of an effort to understand the mechanical properties of high purity glass fibers, it is necessary to examine the strength of pristine fibers that have not been exposed to abrasive or chemically reactive agents. Studies of silica glass fibers in the pristine condition allow us to assess the potential of the optical fibers and provide the opportunity to examine, in a controlled fashion, the role of chemical species in degrading the fiber strength.

In order to form contamination-free surfaces, we use an ultra high vacuum chamber as a means of excluding particulate and chemical species during the fabrication procedure. The silica glass is heated within the vacuum chamber by means of a CO_2 laser to facilitate fiber drawing. After formation, samples can be tested to failure in controlled environmental conditions. Since fiber drawing and testing are conducted in situ, the handling and fixturing problems associated with delicate fiber sections are eliminated.

Although this fabrication and testing procedure offers many advantages for the production and examination of pristine glass fibers, it is difficult to accurately control the heat treatment conditions and fiber draw parameters in a vacuum environment. This paper shows how microprocessor control of the critical processing parameters can be used to establish a procedure for fabricating reproducible pristine fiber test sections.

Materials & Geometry

Samples (Figure 1) are prepared from sections of 1-mm O.D. Amersil T08-WG Commercial quartz rod. A 10-mm length of flame cut rod is fused at each end to larger (2-mm O.D.) sections of generic fused silica (GFS) rod. The GFS sections are then formed into small radius hooks for connection to the drawing apparatus. After cleaning, the 1-mm O.D. section of silica is lightly fire polished to re-

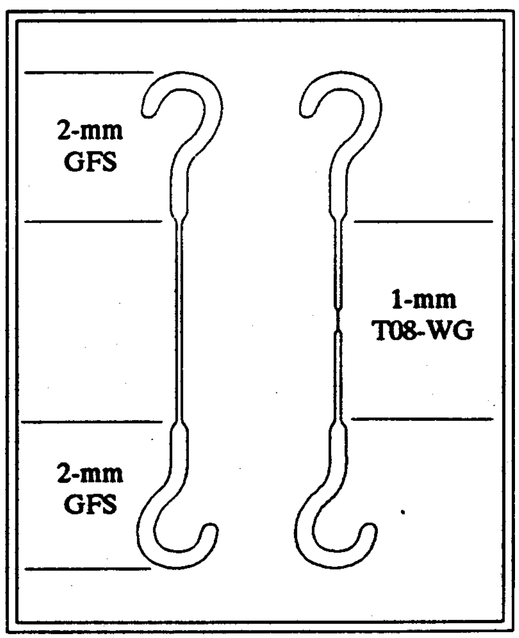


Figure 1 Prepared & Drawn Samples

move any surface imperfections or scratches which might adversely effect the ultimate strength of the fiber.

Samples are then positioned in the vacuum chamber in a vertical alignment with the base hook attached to an internal load cell, and the upper hook attached to a length of stainless steel beaded chain. This configuration provides multiple free pivot points when the sample is positioned for fiber formation. The elimination of lateral fixturing forces is essential to the linear draw of the fiber.

In order to form the fiber, the load chain is fixed to a rotary manipulator which is externally driven with a mechanical advantage of 3:1. Draw force is provided by a 100 gram external dead weight. After the fiber is formed, the manipulator drive is transferred to a linear stepper stage for the application of controlled loads.

Apparatus

The environmental chamber (Photo 1) is constructed of 7-cm stainless steel sections oriented in an elongated vertical configuration. Pumping is provided by a 50 l/s turbomolecular pump which achieves a base system pressure of 1×10^{-08} torr. The system is equipped with a residual gas analyzer, a capacitance barometer, thermocouple gauges, and an ion gauge. All instrumentation is linked to the control computer via the IEEE-488 data interface (Figure 2).

The stainless steel beaded chain is suspended from a rotary manipulator which is fixed to the top of the chamber. A 22-kg capacity load cell is attached (in vacuum) to the base of the chamber. The total separation of 46-cm creates a long draw path for fiber formation. The entire vacuum

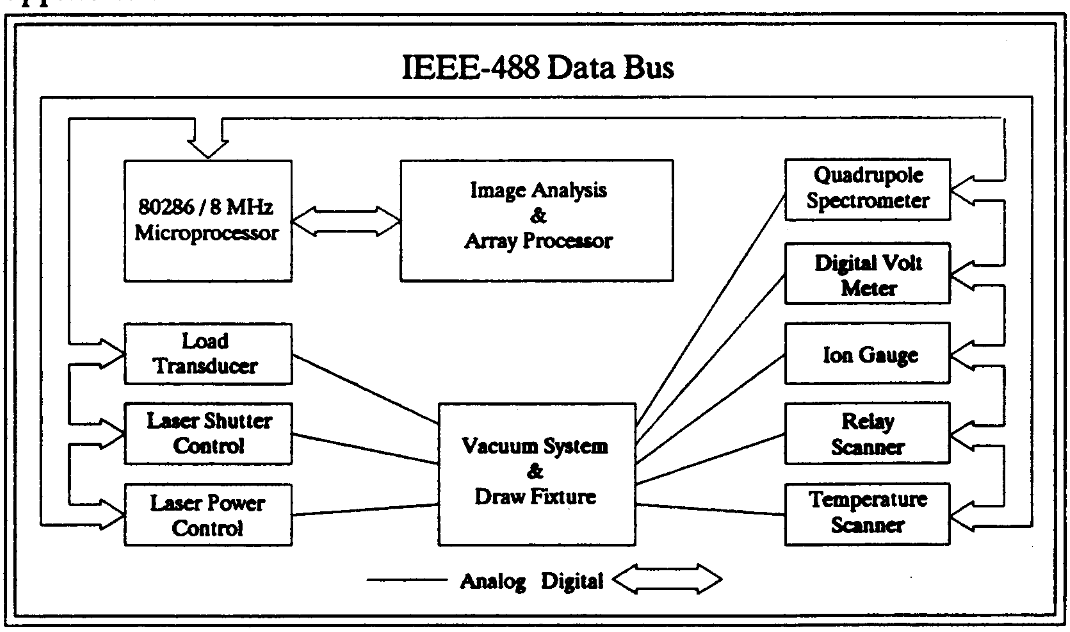


Figure 2 Block Schematic of Logical Information Flow

system is suspended from a hard fixture which provides x-y-z manipulation for sample positioning. With this arrangement, the sample can be accurately positioned in the CO_2 beam path which enters and exits the vacuum chamber through two potassium bromide (KBr) windows (2-cm O.D. x 6-mm thick).

Digital Image Processing

User written control and analysis routines control all aspects of fiber formation and measurement. Routines written in the C programming language (Listing 1) combine library functions containing image processing, IEEE-488 interface control, and numerical analysis routines.

The sample is imaged through a 31X microscope using a pixel array CCD camera. Video information from the camera is transfered via RS-170 interface to a real time image processor at a rate of 30 frames per second. The analog input signal of 0 to 7.14 mV is digitized at the image processor by a flash A/D converter at a 10 MHz rate. This 8-bit digitization produces an image with 256 discrete grey scale levels which are treated as integer values when accessed by the computer. The resulting image is stored in a 512x480 frame memory array and simultaneously displayed on a high resolution color monitor.

Fiber Formation

Due to the high softening point (>1700° C.) of fused quartz, an intense heat source is required to draw fibers. As shown in Figure 3, the carbon dioxide laser produces radiation at a nominal wavelength of 10.6 μm. Since the optical transmission of fused quartz is essentially zero above 4.0 μm[1], this laser light proves to be an excellent heat source for this application.

The laser used in this system is capable of producing 25 watts of power with an exit beam diameter of 3.6-mm. Since the power density of this beam is not sufficient for fiber formation, the original laser beam is directed through a zinc selenide focusing singlet. This lens system produces a well defined focal waist of 1.2-mm diameter with no pre-expansion of the

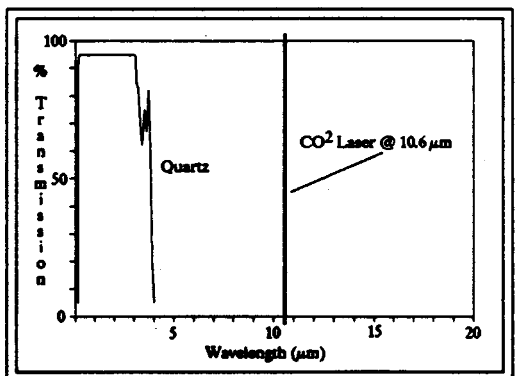


Figure 3 High Absorption of CO^2 Laser Line by Fused Quartz

laser beam. With the laser operating at a power setting of 16 watts, the necessary temperatures are achieved.

Prior to placing the sample in the vacuum chamber, the diameter of the T08-WG section is accurately measured to a resolution of 1.0-μm using a digital micrometer. This measurement is used as a calibration of the optical system once the sample has been mounted. Lighting is adjusted so that the sample section is defined as a dark bar on a light background (Figure 4). A line of pixels is scanned and stored as a single dimension integer array. Since undesirable reflections contained within the bar are difficult to identify and exclude, the array is first evaluated from left to right to identify the left edge of the sample. The array is then indexed and read from right to left to define the right edge of the sample. With this technique, the reflections and specular defects encountered within the body of the dark bar have no effect either on edge detection or the resulting calculation of sample width.

Figure 4 is a representation of the 1-mm silica section. Figure 5 is an example of a scanned line which corresponds to the filar line on Figure 4. In this cross section the edges and the reflected portion in the center of the bar are easily differentiated. The silica section of a 512 pixel scanned line typically occupies 360 pixels with a magnification of 31X.

Proper alignment of the sample is critical to the performance of the drawing process. To facilitate this, a low power helium-neon (HeNe) laser is co-aligned with the high power CO_2 laser. During the setup phase, the operator interposes a turning mirror in the CO_2 beam path to provide a visual HeNe indication of the projected beam line. The sample is translated into the beam through the manipulation of the linear manipulators which support the vacuum system. This alignment is accurately confirmed on a projected image at the exit side of the vacuum chamber.

After initial path alignment

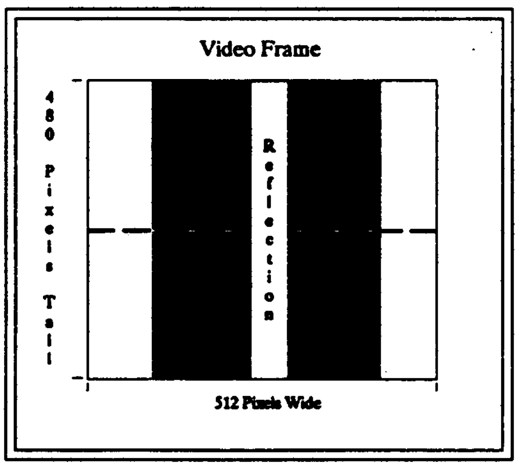


Figure 4 Scanned Line thru a Typical Video Frame

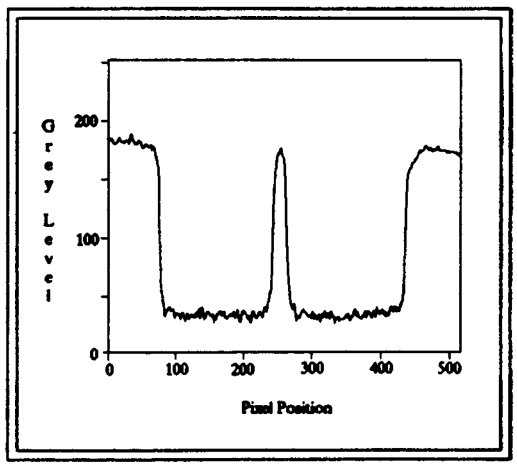


Figure 5 Line Scan of a 1-mm Silica Section

using the HeNe laser, the turning mirror is removed and a test exposure is made with the CO_2 laser. This procedure confirms the vertical positioning of the beam path with relation to the scan position of the image processor. Simulated filars on the display screen are aligned to the center of a heated area on the sample as the operator manually pulses the laser. This process serves a secondary purpose by bringing the section into line with the vertical pull of the system. Any slight alignment irregularities are eliminated as the silica is slightly softened by the laser heat.

Once the 1-mm sample section has been accurately aligned and measured, the drawing process is entirely controlled by the microprocessor. In this procedure, all artificial lighting is removed and the fiber is drawn in a darkened room. When the shutter is opened from the control program, the heating of the silica produces an increasingly intense white light. The control program continuously loops in a recursive routine which acquires a video frame, analyzes a line of pixels, and compares the results with a previously entered target intensity constant. As the silica heats, the program is triggered to a second routine when the intensity (typically - grey level 220) indicates that the sample has reached a softening point (>1700° C). At this point the program repeatedly scans a predetermined line and measures the decreasing sample width in real time (30/sec). When the measured width is equal to or less than a target width value, the shutter is closed and the draw is complete.

At this point the fiber section is illuminated and imaged. The minimum diameter of the drawn section is located by translating the measurement microscope along the length of the fiber. Repeated measurements are taken and averaged to determine the resulting minimum fiber diameter and the total length (~2-mm) of the drawn section. All system and fiber data are recorded to a file and the system is reconfigured for fiber testing.

Strength Testing

Fibers produced in vacuum can be tested to failure under controlled environmental conditions. The vacuum system is configured with an auxiliary gas manifold to provide dosing materials during the testing process. Gas pressures are controlled by monitoring the output of a high resolution capacitance barometer. Once the environment has been stabilized, the linear stage can be driven to load fibers at rates of 0.5 to 50.0 grams per second. Rate, load, and chamber information are transmitted to the control computer for data file storage.

A base line of strength data for a particular material is determined by forming and breaking a series of fibers in a high vacuum of $<5 \times 10^{-08}$ torr. Silica fibers formed in this manner have been tested[2] to stresses in excess of 10.3 GPa in the pristine state.

Summary

This communication describes a novel method for the entire formation and mechanical testing cycle of pristine silica glass fibers. After initial sample insertion and alignment, the experiment is completely microprocessor controlled and proceeds with a minimum of operator interaction. Critical parameters for fiber formation are controlled and recorded by a dedicated microprocessor and stored to disk for further analysis. Fibers drawn in this manner are pristine in nature and consistently uniform. To date, this system has been employed to gather preliminary base line data in preparation for a complete study on the environmental effects on silica materials.

Acknowledgments

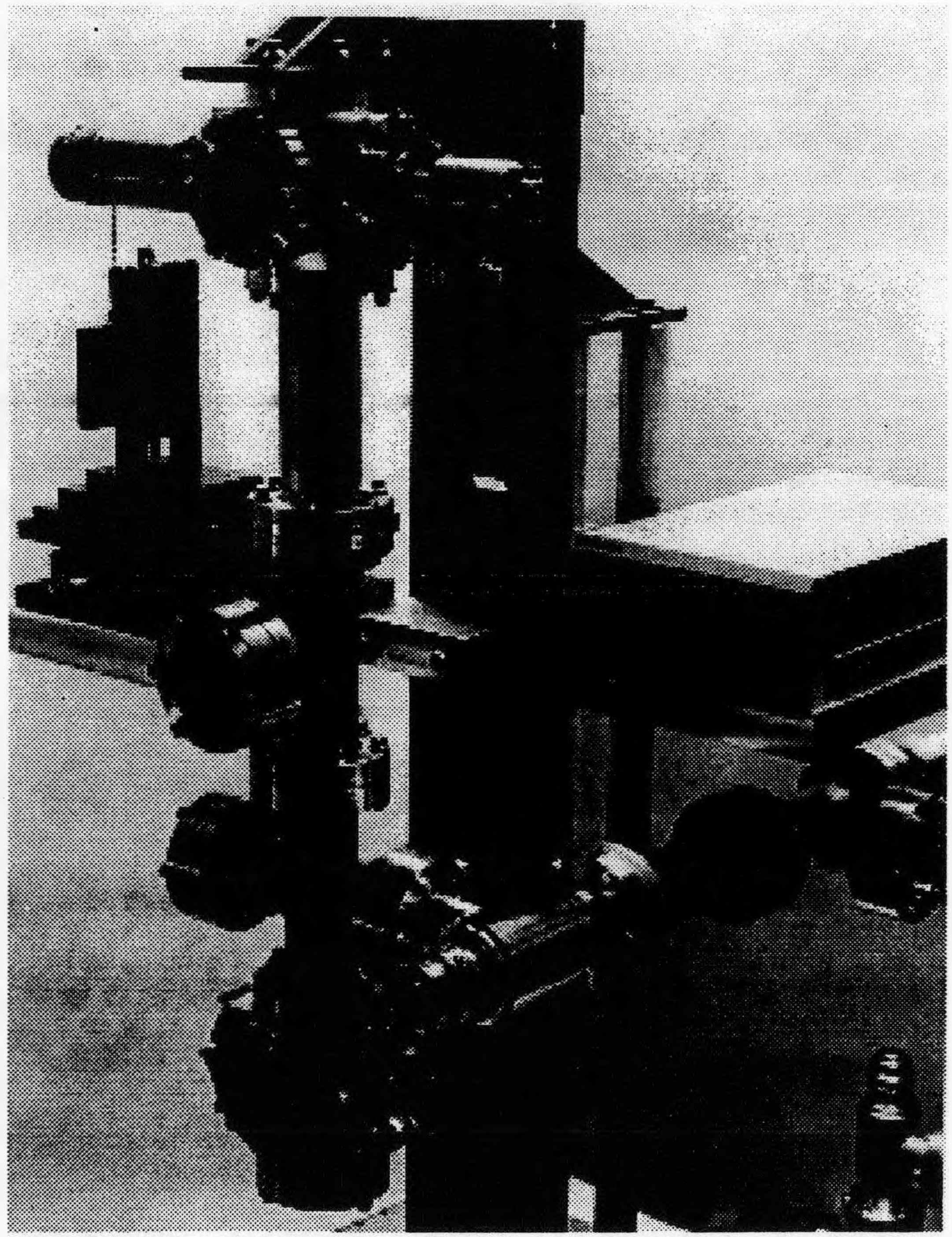
I would like to thank T. L Garcia for his assistance in code development and hardware assembly for this project.

References

1. From Amersil Inc., brochure EM-9227-2 (no date).

2. D.R. Tallant, T.A. Michalske, & W. L. Smith, "The Effects of Tensile Stress on the Raman Spectrum of Silica Glass", Journal of Non-Crystalline Solids 106 (1988) 380-383.

Photo 1



Listing 1

```
/*                    PROGRAM: LASERVC2.C

Description:  LASERVC2.C is a development program that is used to
                                control the fiber drawing operation. The user
                                is guided through the program with pop-up
                                comments that must be acknowleged before
                                continuing. This will insure the safety of oper-
                                ators and the success of the fiber drawing
                                operation. After several screen prompts, the
                                program activates a loop to draw fibers from a
                                1 mm section of silica. When the target width
                                is detected, the user is prompted for final in-
                                put and program is terminated.
Environment: Vacuum

Image
Contrast:      White on Black.

Control:       Laser control is accomplished through HP power supplies,
                                models: 6632A for the laser power, & 59501B
                                for the laser shutter. Images of samples are
                                acquired by way of an image processor furn-
                                ished by Imaging Technology model: FG-100-
                                AT and controlled with ITEX 100 program-
                                ming routines. These devices are accessed
                                through a GeneralPurpose Interface Bus
                                (GPIB) and controlled withMicrosoft C
                                Programming Language Routines.
*/
/***** Header files required for function definitions. *****/

#include "itex100.h"
#include "stdtyp.h"
#include <stdio.h>
#include <graph.h>
#include <decl.h>
#include <ctype.h>
#include <time.h>

/***** Variable definition and declaration. *****/

BYTE bdat1[512];
char rd1[512],rd2[512],fl[12];
char fname[]="color1.hff",meas[]="measure",ref[]="reference";
int a,b,c,i,j,k,fw,ps6,ps7,dvm1,dvm2,qm=0;
int lp1,rp1,cnta[4],cntb[4],width1,width2;
int M1=0,M2=0,M3=0,M4=0,i;
int x=256,y=256;
int x1=256,y1=256;
int x2,y2,ion8,n[10],d;
int v,h,z,v1,h1,z1,cnt=0;

/***** MAIN program begins. *****/

main()
{
FILE *fp;
char rd[7];
float tim,mm,mic,ma,wa;
int test,width,tot=0;
time_t start,finish,ltime;

/***** Initialization of devices. *****/

ps6=ibfind("PS6");
ps7=ibfind("PS7");
dvm1=ibfind("DVM1");
dvm2=ibfind("DVM2");
ion8=ibfind("ION8");
ibwrt(ps6,"VSET 0.00",9);
ibwrt(ps7,"1000",4);
ibwrt(dvm1,"R7R5W6",6);
ibwrt(dvm2,"R7R5W6",6);

/***** Initialization of Image Processor. *****/

_clearscreen(_GCLEARSCREEN);
sethdw(0x300,0xA0000,PSEUDO_COLOR,1);
setdim(512,512,12);
initregs();
initluts();
initialize();
setpmask(0x00FF);
sclear(0);
setpmask(0xF00);
dynamic_luts();
setpmask(0x0FF);
sclear(255,WAIT);

/***** Calculate initial width of the sample. *****/

snap(1);
brhline(0,350,512,bdat1);
beg0:
for(i=i;i<512;i++)
  if(bdat1[i]<140){
  a+=1;
  cnta[a]=i;
  for(i=i;i<513;i++)
    if(bdat1[i]>140){
    b+=1;
    cntb[b]=i;
    if(b>=2)break;
    else goto beg0;
  }
}
width1=cntb[b]-cnta[1];
i=0,a=0,b=0;
grab(0);
ibwrt(ion8,"R02\r\n",3);
for(i=0;i<=11000;i++){
k=k+1;
}
ibrd (ion8,rd,7);
i=0;

/***** User prompts Phase 1 thru Phase 8. *****/
ph1:
_clearscreen(_GCLEARSCREEN);
printf("FIBER DRAWING CONTROL PROGRAM:");
printf("\n\n\nPhase 1: Position sample in chamber and adjust optics...");
printf("\n\n\n\n\nPress RETURN to continue...");
c=getch();
putchar (c);
switch (c){
  case '\r':
  break;
  default:
  goto ph1;
  break;
  }

ph2:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 2: Ensure that power supplies have been ON for");
printf("\n\n        at least 30 minutes before continuing...");
printf("\n\n\n\n\nPress RETURN to continue...");
c=getch();
putchar (c);
switch (c){
  case '\r':
  break;
  default:
  goto ph2;
  break;
  }

ph3:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 3: Ensure that all operators have eye protection\
\n\n        and that area is secure from entry...\
\n\n\n\n\nPress RETURN to continue...");
```

```
c=getch();
putchar (c);
switch (c){
  case '\r':
  break;
  default:
  goto ph3;
  break;
  }
ph4:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 4: Turn ON controller with key switch and \n\
                press start button...");
printf("\n\n\n\n\n\n\n\n\n\n\t\t\a     **** CAUTION ****\n\n\n\
\tImmediately ensure that laser shutter is closed!\
\n\n\n\n\n\n\n\n\nPress RETURN to power up laser...");
c=getch();
putchar (c);
switch (c){
  case '\r':
  lson();
  break;
  default:
  goto ph4;
  break;
  }

ph4a:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 4a: Interpose power head in beam path and adjust");
printf("\n\n          laser power...");
printf("\n\n\n\n\nPress RETURN to continue...");
c=getch();
putchar (c);
switch (c){
  case '\r':
  break;
  default:
  goto ph4a;
  break;
  }

ph4b:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 4b: Input laser power setting (watts)... ");
scanf("%f",&wa);

ph5:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 5: Input current setting (DC mA)... ");
scanf("%f",&ma);

ph5a:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 5a: Remove power head from beam path... ");
printf("\n\n\n\n\nPress RETURN to continue...");
c=getch();
putchar (c);
switch (c){
  case '\r':
  break;
  default:
  goto ph5a;
  break;
  }

ph6:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 6: Enter sample width (mm): ");
scanf("%f",&mm);
mic=((mm*1000.0)/width1);

ph7:
_clearscreen(_GCLEARSCREEN);
printf("\nPhase 7: Enter target width in pixels: ");
scanf("%i",&test);

ph8:
_clearscreen(_GCLEARSCREEN);
printf("\007");
printf("\nPhase 8: Ready for fiber drawing operation...\n\n\
                Present sample width = %i pixels.\n\
                Target width set at    %i pixels.\n",width1,test);
printf("\n\n\n\n\n\nTurn OFF light source and room lighting...\
\n\n\n\n\n\n\n\n\n\nPress RETURN to open shutter and heat
sample...");
c=getch();
putchar (c);
switch (c){
  case '\r':
  shon();
  break;
  default:
  goto ph8;
  break;
}
/***** Detect trigger point. *****/
readlut(fname);
delta_ms();
trig:
                printf("\nTrigger...");
                snap (1);
                brhline(0,350,512,bdat1);
                line(0,351,512,351,GREEN_OVERLAY<<8);
                for(i=0;i<512;i++){
                if(bdat1[i]>qm)qm=bdat1[i];
                if(bdat1[i]<qm)qm=qm;
                 if(bdat1[i]>220)goto init;
                }
                goto trig;

/***** Calculate width of sample and compare to target. *****/
init:
i=0;
lp:
snap(1);
brhline(0,255,512,bdat1);
line(0,256,512,256,RED_OVERLAY<<8);
for(i=i;i<512;i++)
  if(bdat1[i]>50)
  {
  lp1=i;
  break;
  }
rp:
  for(i=511;i>=0;i--)
  if(bdat1[i]>50)
  {
  rp1=i;
  break;
  }
width:
   width2=rp1-lp1;
   if(width2<=test)goto end;
   else goto init;
end:

/***** Reset devices to default settings. *****/
tim=delta_ms();
time(&finish);
shoff();
lsoff();
printf("\007");
printf("\nTARGET DETECTED!");
printf("\n\nFiber drawing sequence terminated.\n\n");
ibwrt(ps6,"VSET 0.00",9);
ibwrt(ps7,"1000",4);
ibwrt(dvm1,"R7R5W6",6);
ibwrt(dvm2,"R7R5W6",6);
ibloc(dvm1);
ibloc(dvm2);
ibloc(ps6);

ph9:
printf("\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\nPhase 9: Turn laser off \
with key switch...\n\nPress RETURN to focus image...");
c=getch();
putchar (c);
switch (c){
  case '\r':
  break;
  default:
  goto ph9;
  break;
```

```
}
ph10:
_clearscreen(_GCLEARSCREEN);
initialize();
setpmask(0x00FF);
sclear(0);
setpmask(0xF00);
dynamic_luts();
setpmask(0x0FF);
grab(0);
printf("\nPhase 10: Continuous GRAB mode...\
\n\n\n\n\n\n\nPress RETURN to measure fiber...");
c=getch();
putchar (c);
switch (c){
  case '\r':
  break;
  default:

    goto ph10;

  break;
  }

/***** Routine to measure fiber. *****/
snapm:
_clearscreen(_GCLEARSCREEN);
printf("\nAuto-Measure...\n");
a=0,b=0,tot=0;
i=0;
for(k=1;k<=5;k++){
snap(1);
brhline(0,255,512,bdat1);
hlclear(0,256,512,RED_OVERLAY<<8);
text(400,241,0,1,RED_OVERLAY<<8,meas);
hlclear(0,351,512,GREEN_OVERLAY<<8);
text(400,336,0,1,GREEN_OVERLAY<<8,ref);
i=0;
begin1:
for (i=i;i<512;++i)
 if(bdat1[i]<85){
   a+=1;
   cnta[a]=i;
   if((i>=512) exit();
   else goto begin;
                 begin:
                 for(i=i;i<513;i++)
                 if(bdat1[i]>85){
                 b+=1;
                 cntb[b]=i;
                 if(i>=512) exit();
                 else goto begin1;
                 }
}
width=cntb[b]-cnta[1];            /* Output data. */
printf("\n   width%i = %d pixels.",k,width);
tot=width+tot;
}
printf("\n\n   average width = %i",tot/5);

/***** Mouse-Measure routine. ****/
START:
 MOUSECMM(&M1,&M2,&M3,&M4);       /* Initialize Mouse */
 M1=11;                     /* Zero Motion Counter */
 MOUSECMM(&M1,&M2,&M3,&M4);
 MOUSECMM(&M1,&M2,&M3,&M4);
 printf ("\n\nMouse-Measure...\n\n");
 printf ("   Click left button to read pixel position.");
 printf ("\n   Click right button to continue.\n");
 printf ("\n    X   Y   Z");
 z=rpixel(x,y);

LOOP:                       /* Display Cursor */
 hlclear(x-5,y,10,RED_OVERLAY<<8);
 vlclear(x,y-5,10,RED_OVERLAY<<8);
 M1=6;                      /* Get Right Button Release */
 M2=1;
 MOUSECMM(&M1,&M2,&M3,&M4);
 if (M2>0) goto ENDIT;
 M1=6;                      /* Get Left Button Release */
 M2=0;
 MOUSECMM(&M1,&M2,&M3,&M4);
 x2=x1;
 y2=y1;
 if (M2>0) goto RECORD;
 M1=11;                     /* Check for X-Movement */
 MOUSECMM(&M1,&M2,&M3,&M4);
 x1=x1+M3;
 y1=y1+M4;
 if (x==x1) goto CON2;
 goto MOVE;
CON2:                       /* Check for Y-Movement */
 if (y==y1) goto LOOP;
MOVE:                        /* Update X & Y - Start Again */
 x=x+M3;
 y=y+M4;
 hlclear(x2-5,y2,10,RED_OVERLAY<<0);
 vlclear(x2,y2-5,10,RED_OVERLAY<<0);
 z=rpixel(x,y);
 x1=x;
 y1=y;
 goto LOOP;
RECORD:
 cnt=cnt+1;
 printf("\n   %3i %3i %3i",x,y,z);
 n[cnt]=x;
 hlclear(x2-5,y2,10,RED_OVERLAY<<0);
 vlclear(x2,y2-5,10,RED_OVERLAY<<0);
 if(cnt==2)
 {
 d=n[1]-n[2];
 printf("\t\t   width = %3i",abs(d));
 cnt=0;
 goto LOOP;
 }
 else
 goto LOOP;
ENDIT:
 hlclear(x2-5,y2,10,RED_OVERLAY<<0);
 vlclear(x2,y2-5,10,RED_OVERLAY<<0);
 hlclear(0,256,512,RED_OVERLAY<<8);
 hlclear(0,351,512,GREEN_OVERLAY<<8);

ph11:
printf("\n\n\n\nDo you want to repeat measurement (y or n)? ");
c=getch();
putchar (c);
switch (c){
  case 'y':
  goto snapm;
  break;
  case 'n':
  break;
  default:
  goto ph11;
  break;
  }
_clearscreen(_GCLEARSCREEN);
printf("\nEnter actual fiber width: ");
scanf("%i",&fw);

/***** Output program parameters. *****/
_clearscreen(_GCLEARSCREEN);
printf("\nPROGRAM PARAMETERS:");
printf("\n\nLeft point  = %i.",lp1);
printf("\nRight point = %i.",rp1);
printf("\n\nStarting width = %d pixels = %5.2f microns.\n\
Ending width   = %d pixels = %5.2f microns.",\
width1,(mic*width1),width2,(mic*width2));
printf("\n\nTarget width        = %i pixels = %7.2f
microns.",test,(mic*test));
printf("\nActual fiber width    = %i pixels = %5.2f microns.",fw,(mic*fw));
printf("\nLaser power setting   = %3.2f watts.",wa);
printf("\nLaser current setting = %3.2f mA.",ma);
printf("\nVacuum reading        = %e",rd);
printf("\n\nProgram triggered at a value of %i.",qm);
printf("\nElapsed time        = %3.2f seconds.",tim/1000.0);
ask:
printf("\n\n\n\nDo you want program parameters written to file (y or n)?
");
c=getch();
putchar(c);
switch(c){
   case 'n':
```

```
  break;
  case 'y':
printf("\n\nEnter name of data file with '.dat' extension: ");
scanf("%s",fi);
fp=fopen(fi,"w+");
time(&ltime);
fprintf(fp,"%s",fi);
fprintf(fp,"\n\r%s",ctime(&ltime));
fprintf(fp,"\r%i",lp1);
fprintf(fp,"\r%i",rp1);
fprintf(fp,"\r%d  %5.2f\r%d
%5.2f",width1,(mic*width1),width2,(mic*width2));
fprintf(fp,"\r%i  %7.2f",test,(mic*test));
fprintf(fp,"\r%i  %5.2f",fw,(mic*fw));
fprintf(fp,"\r%3.2f",me);
fprintf(fp,"\r%3.2f",we);
fprintf(fp,"\r%s",rd);
fprintf(fp,"\r%i",qm);
fprintf(fp,"\r%3.2f",tim/1000.0);
  break;
  default:
  goto ask;
  break;
}
printf("\n\n\n\n\nLASER CONTROL PROGRAM TERMINATED!");
fclose(fp);
}
```

```
/***** MAIN program ends. *****/

/***** Definition of FUNCTION subroutines. ****/
/* Function lson */
   lson()
   {
   ibwrt(ps6,"VSET 12.00",10);
   ibrd(dvm1,rd1,16);
   }
/* Function lsoff */
   lsoff()
   {
   ibwrt(ps6,"VSET 0.00",9);
   ibrd(dvm1,rd1,16);
   }
/* Function shon */
   shon()
   {
   ibwrt(ps7,"2713",4);
   ibrd(dvm2,rd2,16);
   }
/* Function shoff */
   shoff()
   {
   ibwrt(ps7,"1000",4);
   ibrd(dvm2,rd2,16);
   }
```

Distribution

1110 S. T. Picraux
1111 B. L. Doyle
1112 S. M. Myers
1114 T. A. Michalske
1114 W. L. Smith (20)
1823 D. R. Tallant
1840 R. E. Loehman
1842 B. C. Bunker
3141 C. Ward (8)
3151 W. I. Klein (3)
8524 J. A. Wackerly