

Full Surface Interferometric Testing of Grazing Incidence Mirrors

AD-A219 394

Final Report

Principal Investigator: John L. Remo, Ph.D.

E.R.G. Systems Inc.
Brackenwood Path
St. James, New York 11780

February, 1990


John L. Remo, Ph.D.

Table of Contents

1. Final Report Abstract and Executive Summary.....	1
2. Achievement of the Phase I Objectives.....	3
3. Description of the Interferometer Breadboard.....	5
4. Testing and Data Reduction.....	9
5. Software for Cylindrical Surface Measurement.....	58
6. Conclusions.....	71

This research is sponsored by SDIO/IST and managed by ONR
Contract Number: N 00014-89-C-0199

DTIC
ELECTE
MAR 16 1990
S E D

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

90 05 06 09

Final Report Abstract and Executive Summary

The aim of this six month SDIO SBIR contract was to demonstrate the proof-of-principle of the "Full Surface Interferometric Scanner" (FSIS), an instrument which can rapidly and reliably measure both the full surface figure as well as the macroroughness of grazing incidence optics. The FSIS has the potential to fill the need of SDIO to characterize and qualify the necessary off-axis aspherical mirror technology that will be used for weapon pointing, beam control, and beam propagation through several environments and countermeasures.

This instrument design is based on the use of normal incidence, sub-aperture interferometry and wavefront shearing interferometry which surmounts many of the problems encountered by other (eg, long trace profilers) techniques. This new system, the FSIS, for which we have developed a breadboard system which makes novel use of three sequential operations: sub-aperture slope measurement, wavefront integration, and surface profile synthesis.

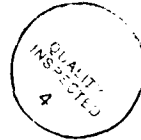
The FSIS breadboard construction was completed during December 1989 and tests were carried out on various optical components such as those typically used for imaging X-ray and hard UV-radiation at grazing incidence. These tests were extended through February 15, 1990, at which time the first phase of the project was concluded.

During the last period of this research project additional development of the software algorithms was carried out; a source code listing is enclosed in this report.

From our tests on various optical surfaces it appears that the FSIS will find application in X-ray and UV high resolution lithography, medical imaging, astronomy, physics, microbiology, and industrial (surface) quality control.

During Phase II two completely automated prototype versions of a rugged and self-contained FSIS will be developed to measure a broad range of grazing incidence optical surfaces.

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	



STATEMENT "A" per Dr. Vernon Smiley
 OIR/Pasadena/1262
 TELECON 3/15/90

VG

Achievement of the Phase I Objectives

The Phase I objective was to design, construct, and test an instrument that would be capable of measuring the full surface figure and macroroughness (mm spatial frequency surface error) for aspheric grazing incidence optics. This instrument, the FSIS, was built in a breadboard configuration and measurements were carried out on (cylindrical) sub-aperture segments. Several of these sub-aperture can be synthesized in the software. These measurements can be rapidly carried out (in under 1 second) thereby minimizing sensitivity to thermal fluctuations as well as to other environmental effects that have seriously impaired the effective operation of other instruments. This very short measurement time is necessary when attempting to measure the figure of optical surfaces to the tolerances necessary for grazing incidence X-ray and UV wavelengths in addition to imaging at other wavelengths by off-axis aspheric optics.

For very large mirrors which are greater than the size of the beam width, the full surface scanning capability was achieved through the use of fitting algorithms for overlapping sub-apertures which allowed the full surface shape to be synthesized. A least squares fit is currently being used for the Phase I proof-of-principle demonstration.

During the phase II work more sophisticated algorithms corresponding to particular aspheric surfaces will be tested and their accuracy evaluated. Additional improvements will be made in through the use of a higher resolution CCD camera, improved video frame grabbing and enhanced signal processing.

The present (Phase I) breadboard system stage was not motorized and the system was manually operated. The Phase II instruments will be fully motorized and automated.

Description of the Interferometer Breadboard

A schematic diagram of the interferometer system is shown in figure 1 and was described in detail in the interim report (December, 1989). A photograph of this system is shown in figure 2. To allow for the necessary degrees of freedom a five axis mirror holding system was designed and constructed and includes;

- 1) An in-plane rotation to align the mirror on to a cylindrical reference surface.
- 2) A two axis tilt
- 3) A two axis translation

A photograph of the assembly is shown in figure 3.

The phase shifting was obtained by the appropriate translation of the grating which served as the shearing device.

The test surface was the convex surface of a cylindrical lens having a 500mm focal length and a 250mm radius of curvature.

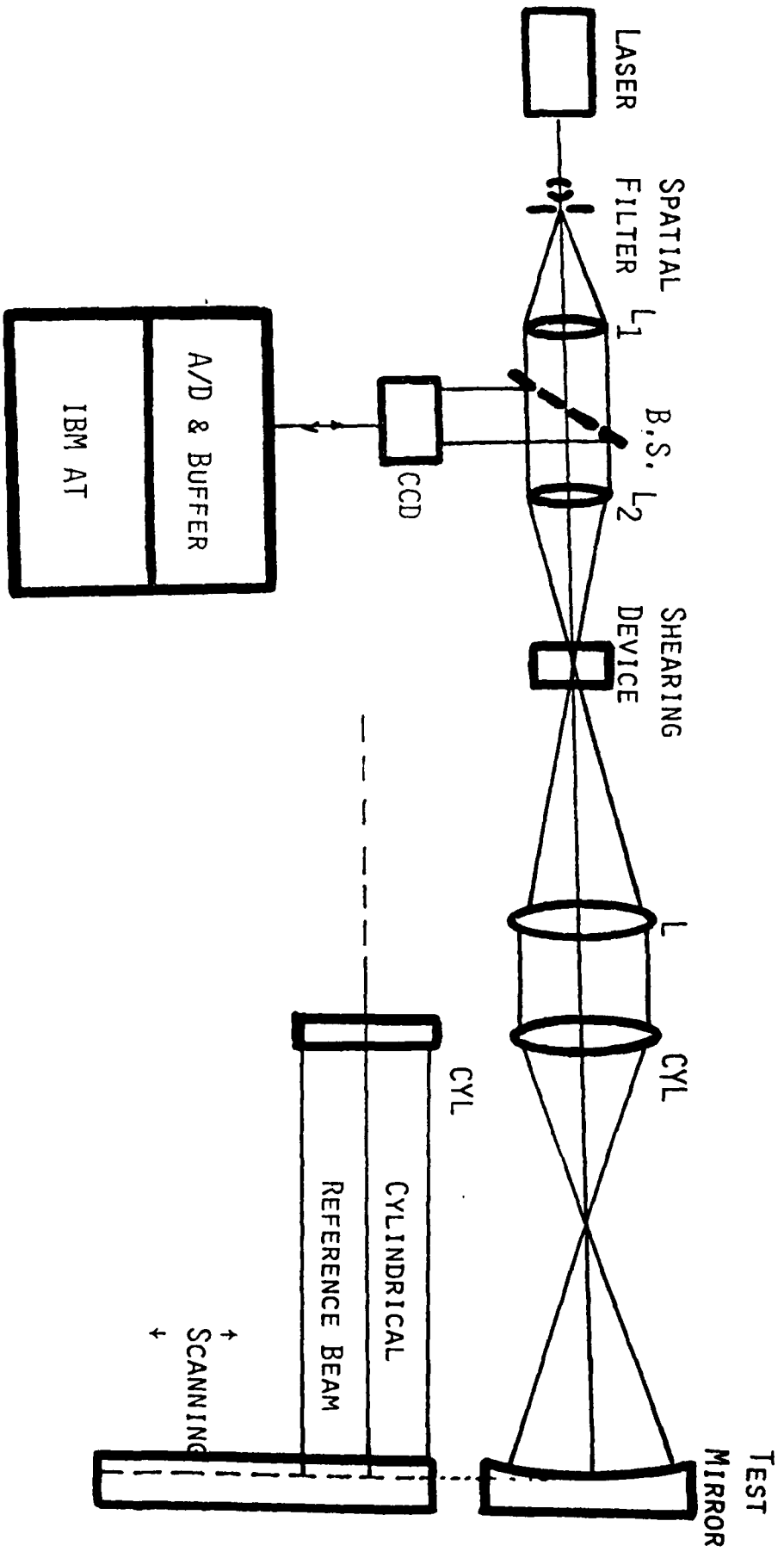


FIGURE 1: SCHEMATIC DIAGRAM OF INTERFEROMETER FOR FULL SURFACE TESTING OF CYLINDRICAL TYPE MIRRORS

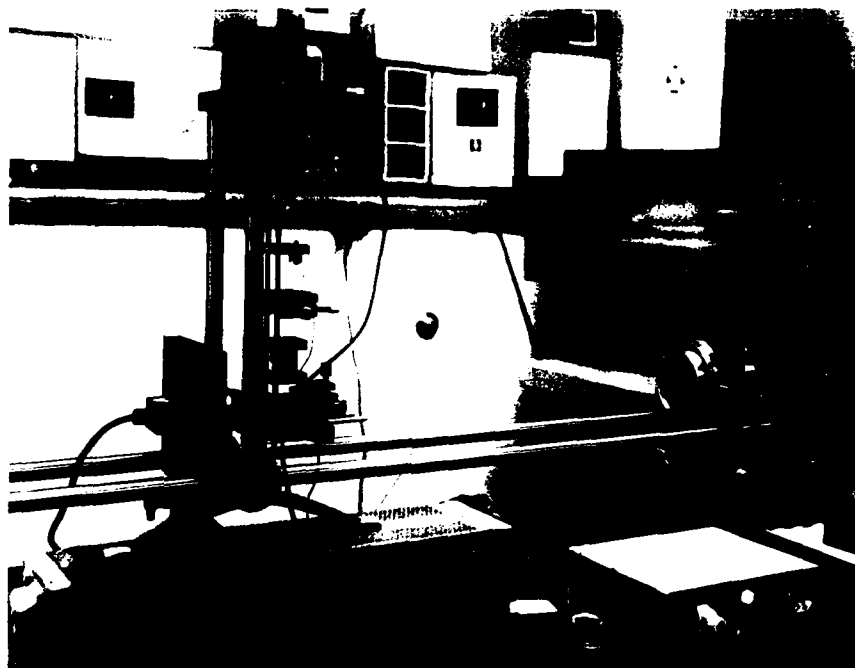


FIGURE 2: PHOTOGRAPH OF COMPLETE INTERFEROMETER BREADBOARD

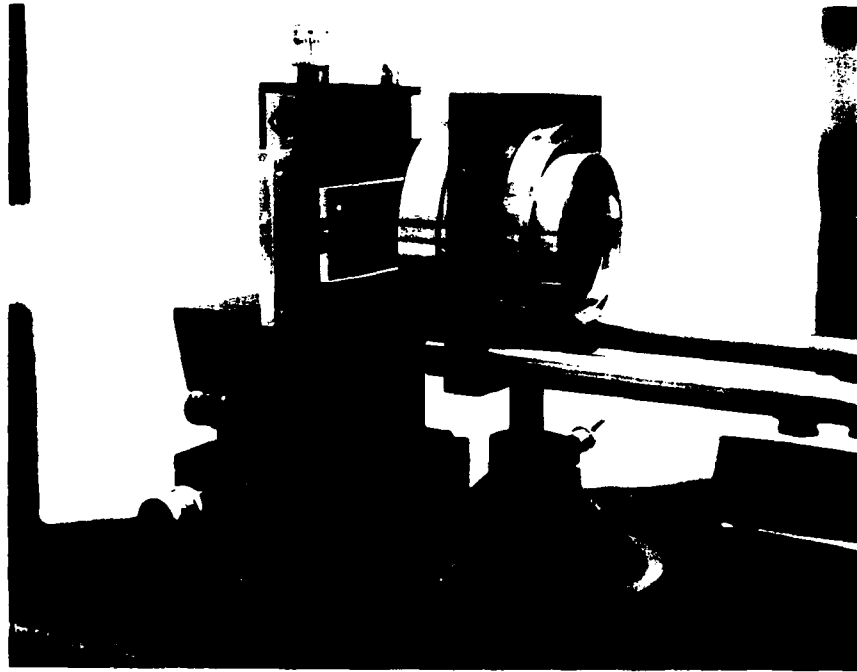


FIGURE 3 : DETAILED VIEW OF CYLINDRICAL MIRROR HOLDING
ASSEMBLY (5 AXES : IN-PLANE ROTATION, 2 TILTS
AND 2 TRANSLATIONS)

Testing and Data Reduction

In order to appropriately test the FSIS on our sample optic, four cases were considered:

1. The mirror assembly is in-focus and the mirror surface well with the reference surface.
2. The mirror assembly is in-focus, but the surface was misaligned by 1° with respect to the reference surface.
3. The mirror assembly was out-of-focus and the surface well aligned with the reference surface.
4. The mirror assembly was out-of-focus and the surface was misaligned by 1° with respect to the reference surface.

Figures 4, 8, 12, and 16 and figures 6, 10, 14, and 18 show photographs of the corresponding interferograms for X-shear and Y-shear respectively, ie, the X and Y slope functions.

Figures 5, 9, 13, and 17 and 7, 11, 15, and 19 show the phase measurement of the corresponding interferograms for the X-shear and Y-shear respectively. Each phase interferogram has the :

- A) Original, 3-D plot.
- B) Tilt removed, 3-D plot.
- C) Original, contour plot.
- D) Tilt removed, contour plot.

The interferograms were imaged on to a CCD camera, A/D converted and stored in a frame buffer. Phase measurement was then carried out digitally and the results shown in the accompanying figures. These results show in particular that these interferometer measurements are insensitive to focus or mirror alignment. This is in strong contrast to the scanning interferometer systems.

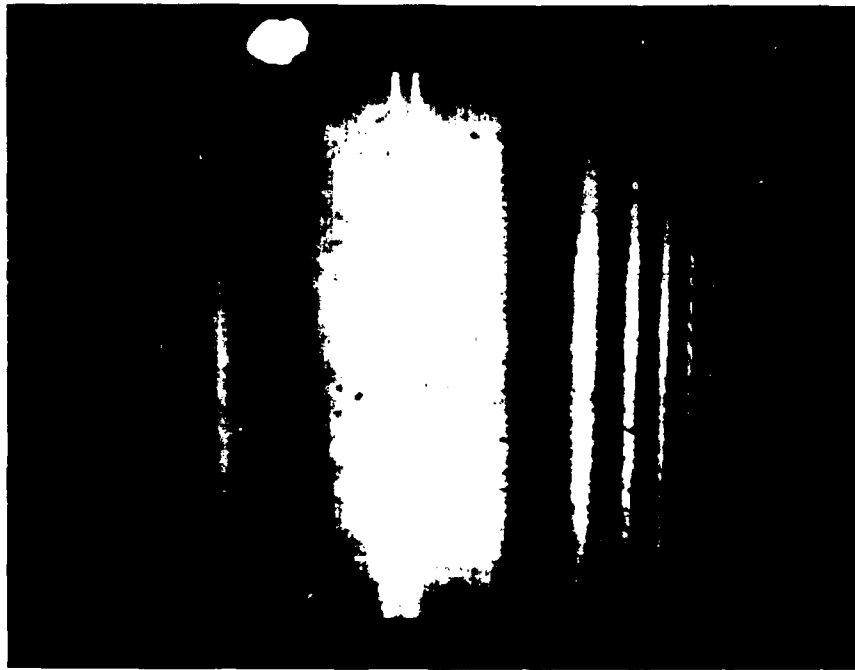


FIGURE 4: X-SHEAR INTERFEROGRAM (CURVED SIDE OF CYLINDER),
WITH MIRROR IN FOCUS AND ALIGNED

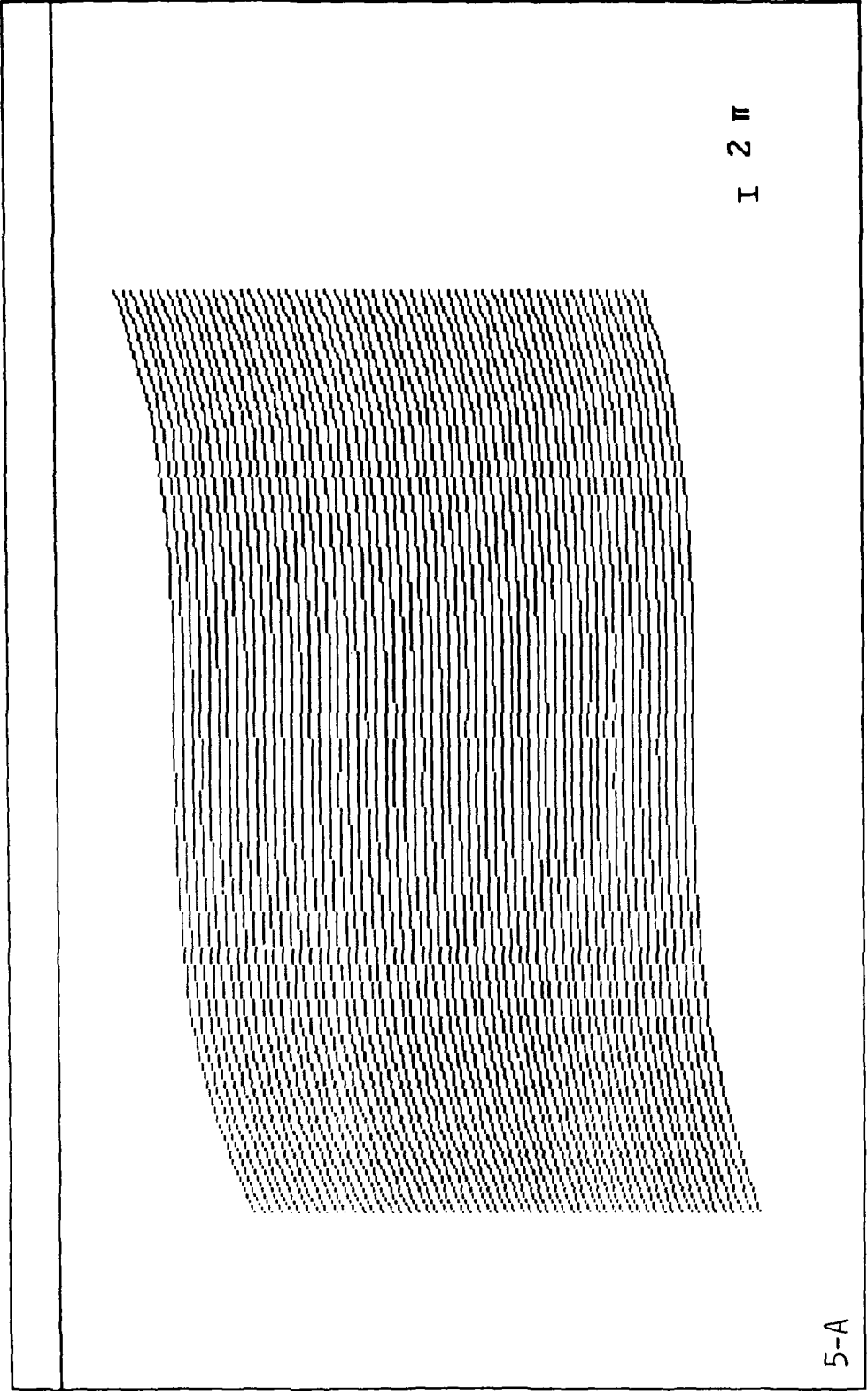
FIGURE 5: PHASE MEASUREMENT FOR X-SHEAR INTERFEROGRAM (FIG.4)

5-A: ORIGINAL, 3-D PLOT

5-B: TILT REMOVED, 3-D PLOT

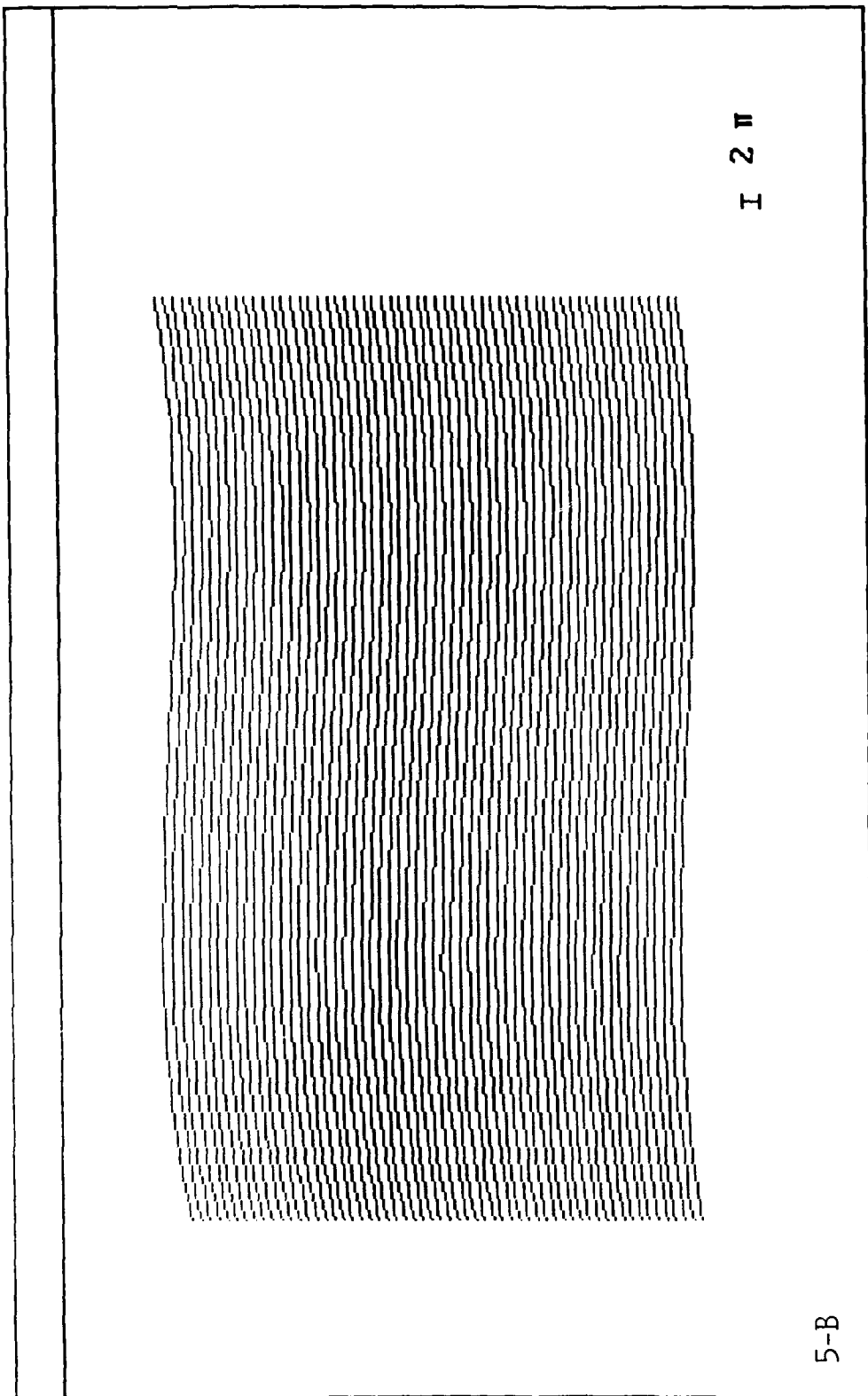
5-C: ORIGINAL, CONTOUR PLOT

5-D: TILT REMOVED, CONTOUR PLOT



I 2 II

5-A

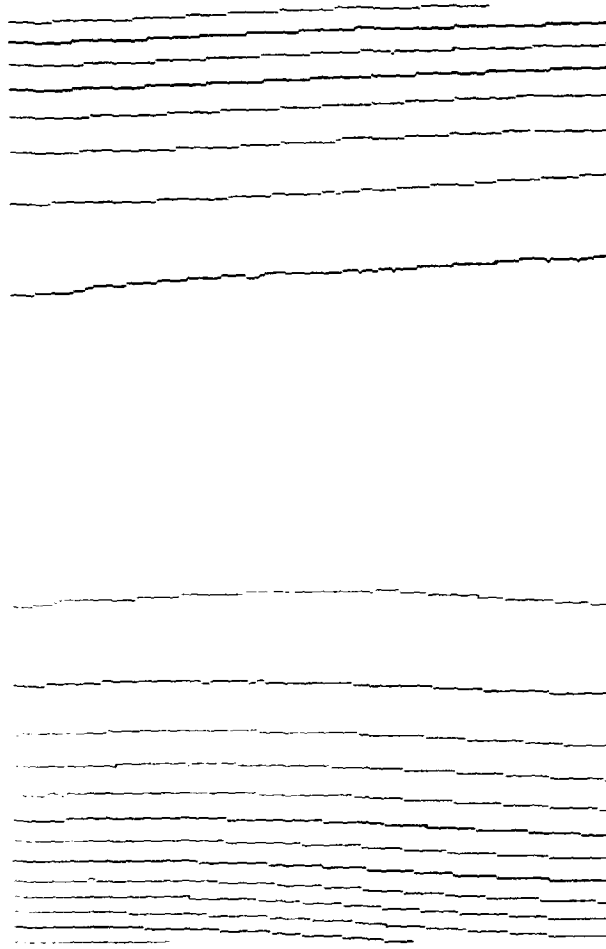


I 2 II

5-B



THIN LINE CONTOUR



1.00 Wave / Level

5-C

THIN LINE CONTOUR



1.00 Wave / Level

5-D

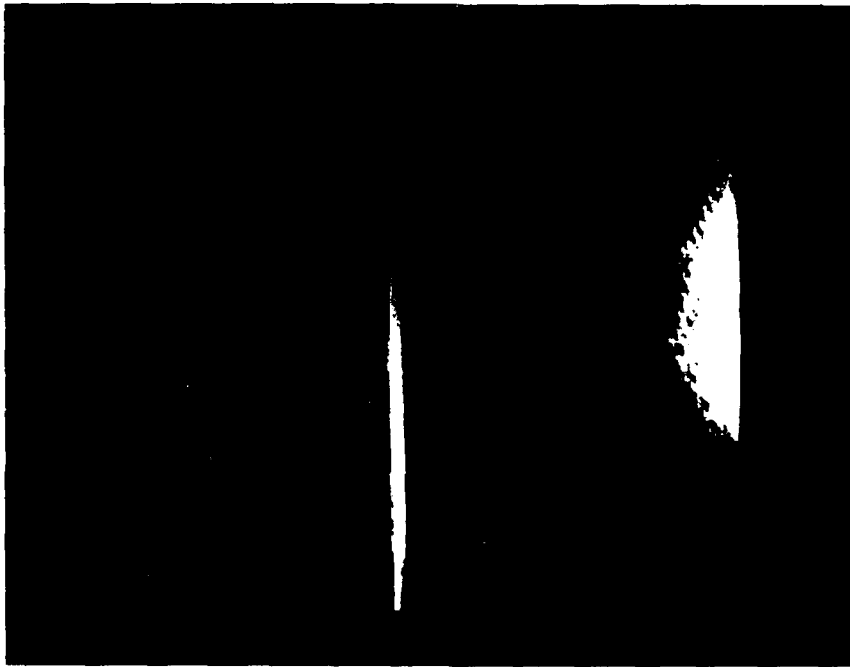


FIGURE 6: Y-SHEAR INTERFEROGRAM (STRAIGHT SIDE OF CYLINDER),
WITH MIRROR IN-FOCUS AND ALIGNED

FIGURE 7: PHASE MEASUREMENT FOR Y-SHEAR INTERFEROGRAM (FIG. 6)

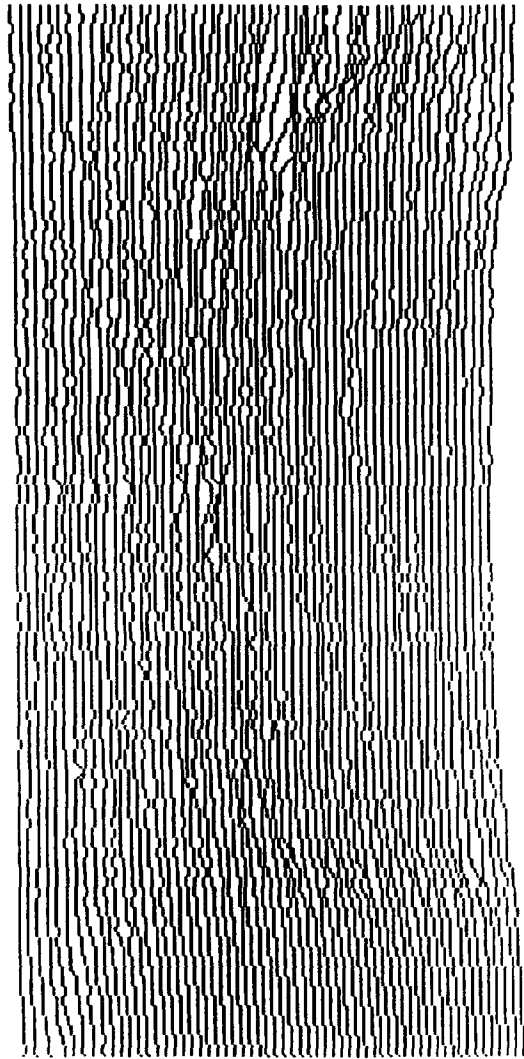
7-A: ORIGINAL, 3-D PLOT

7-B: TILT REMOVED, 3-D PLOT

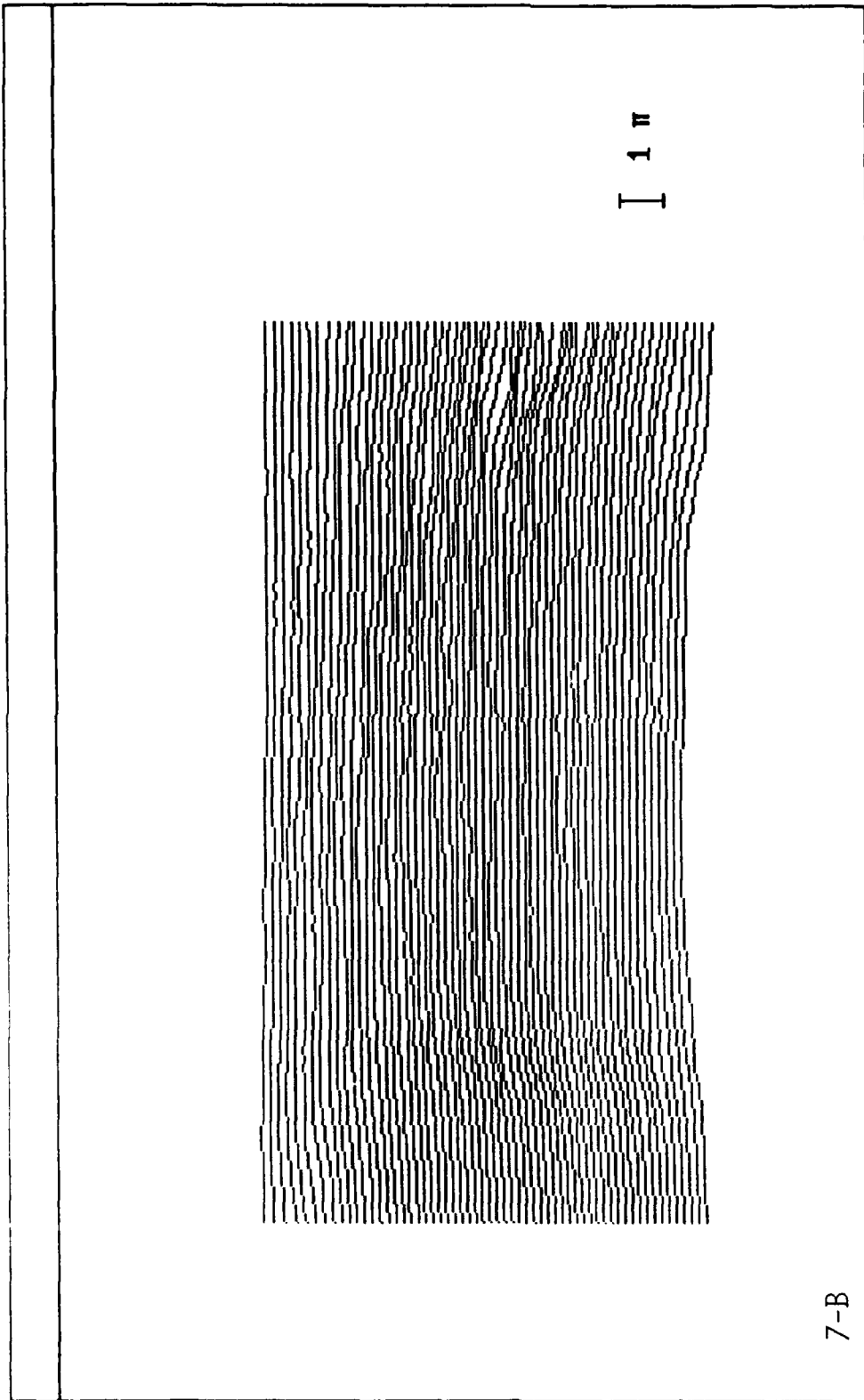
7-C: ORIGINAL, CONTOUR PLOT

7-D: TILT REMOVED, CONTOUR PLOT

I 1 II



7-A

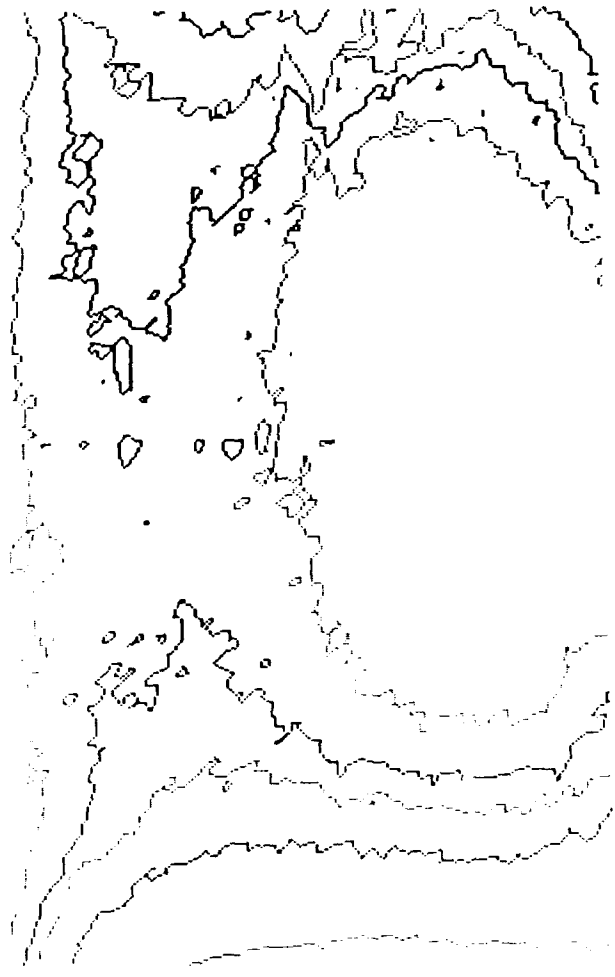


I 1 H

7-B



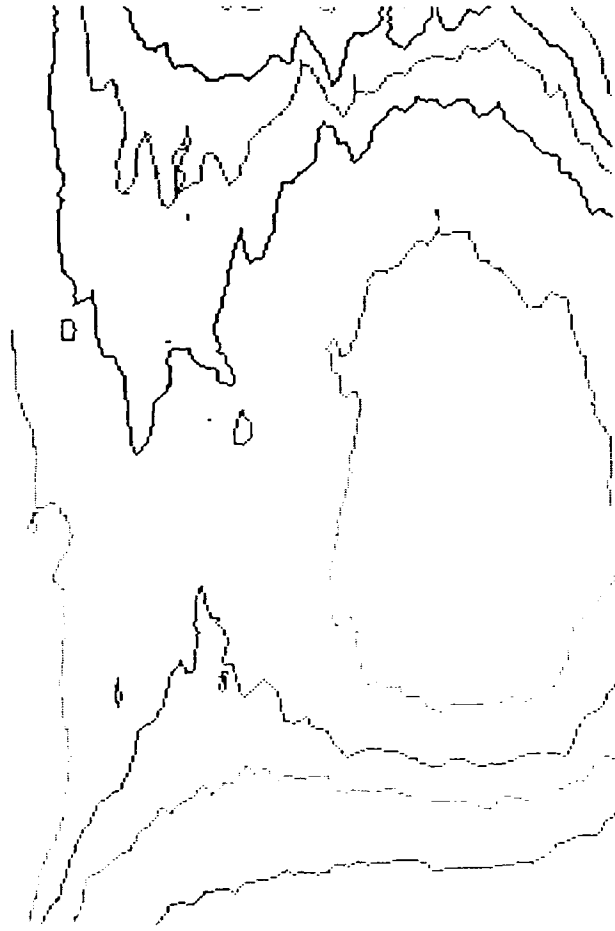
THIN LINE CONTOUR



0.20 Wave / Level

7-C

THIN LINE CONTOUR



7-D

0.20 Wave / Level

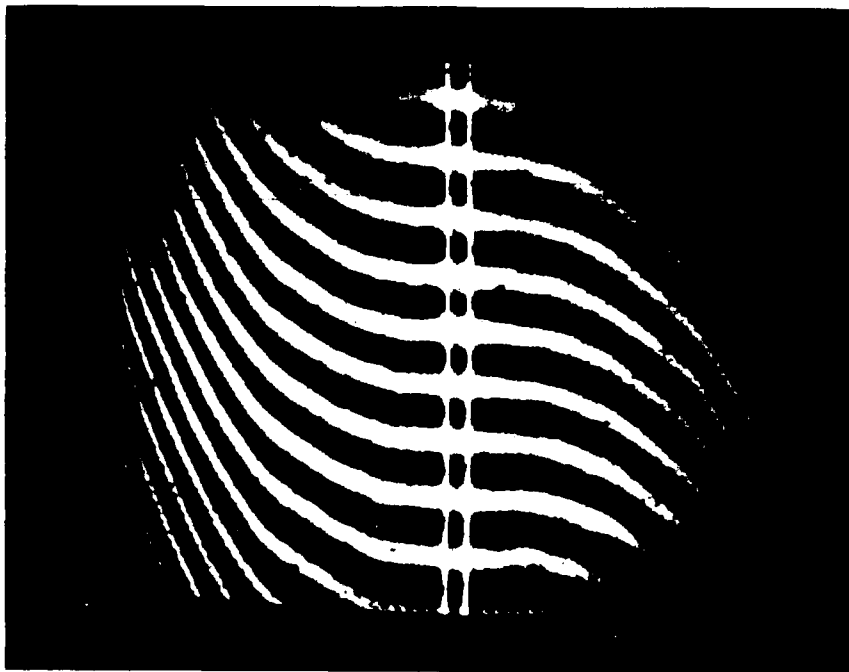


FIGURE 8: X-SHEAR INTERFEROGRAM (CURVED SIDE OF CYLINDER),
WITH MIRROR IN-FOCUS AND MISALIGNED BY 1°

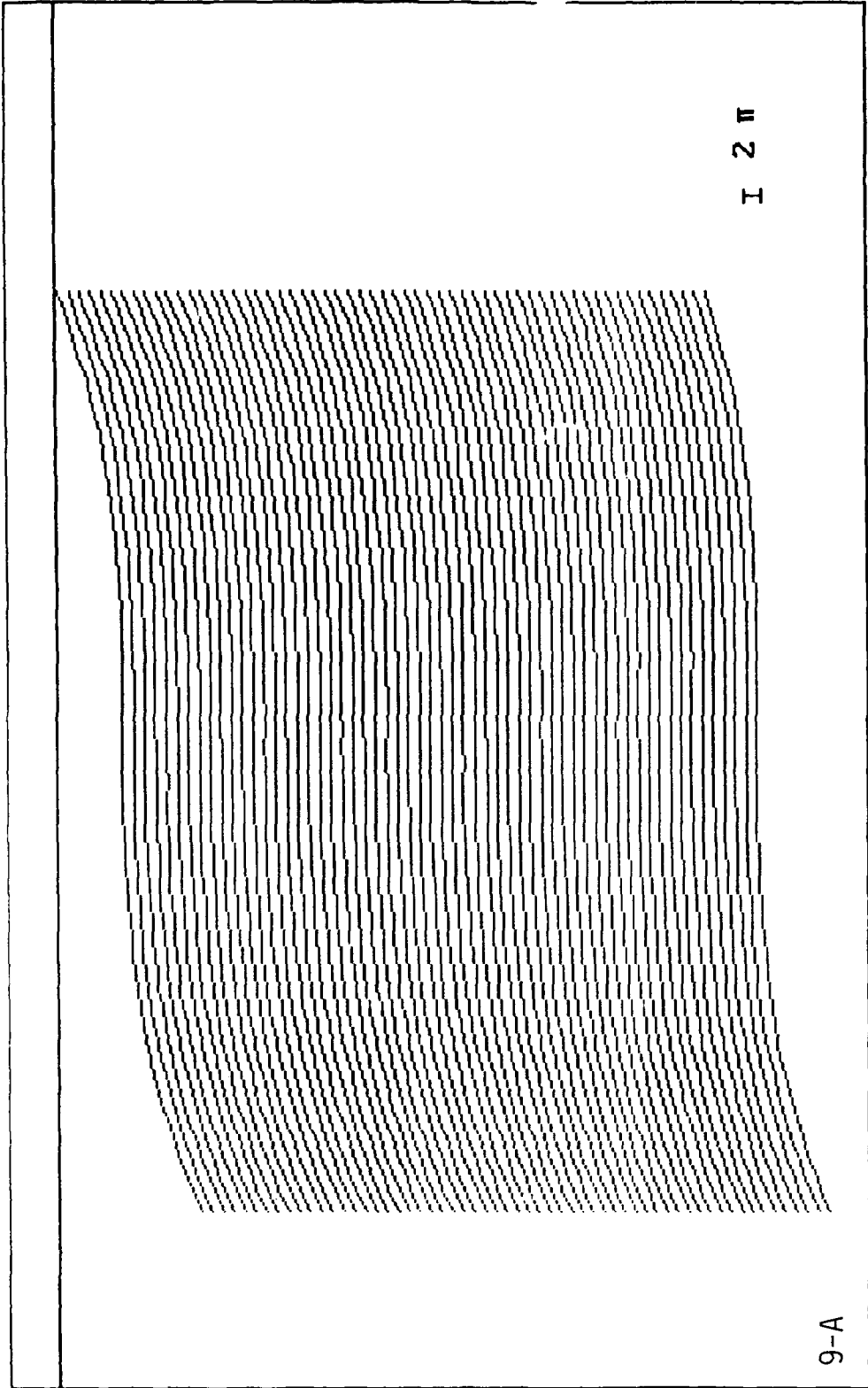
FIGURE 9: PHASE MEASUREMENT FOR X-SHEAR INTERFEROGRAM (FIG.8)

9-A: ORIGINAL 3-D PLOT

9-B: TILT REMOVED, 3-D PLOT

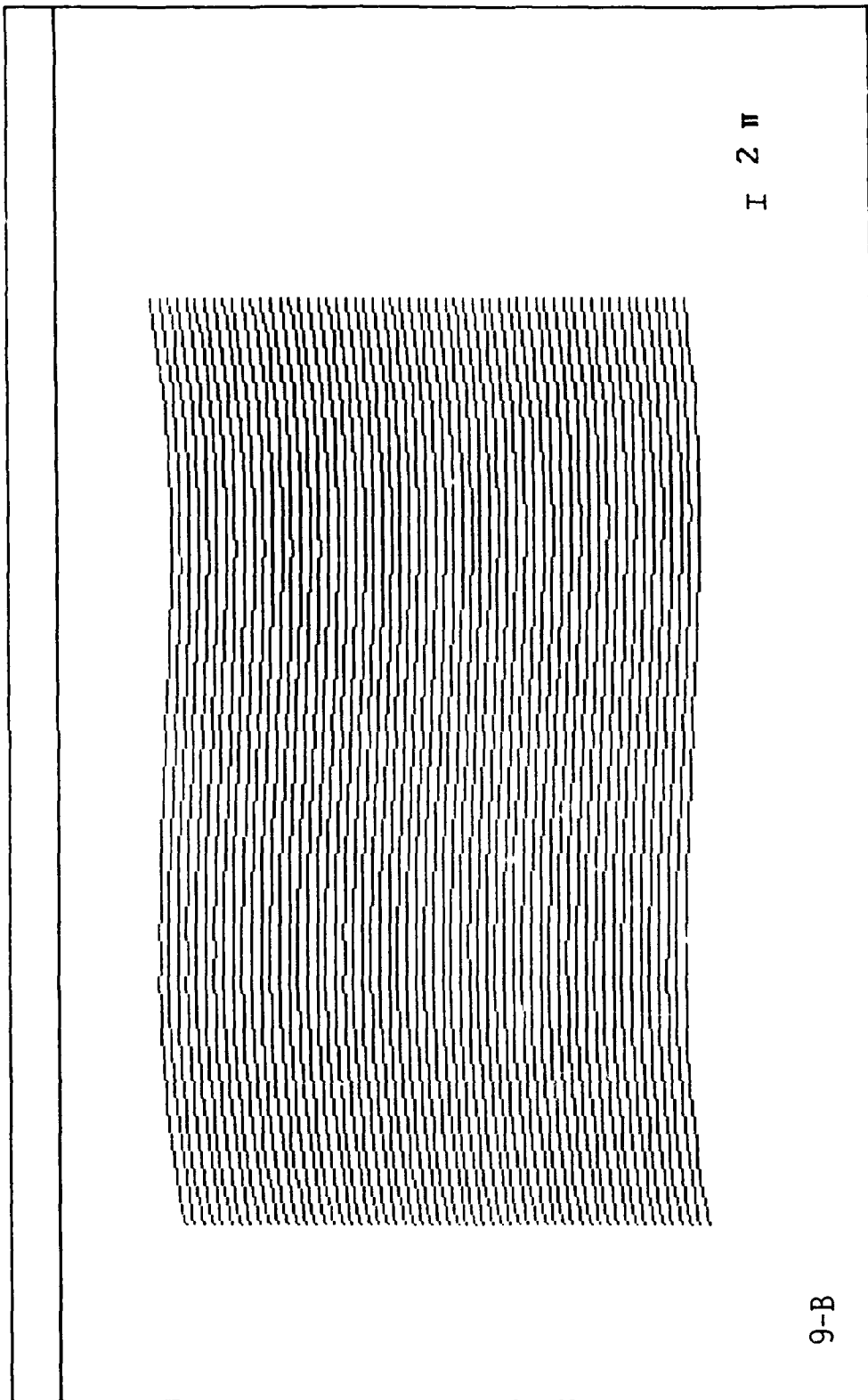
9-C: ORIGINAL, CONTOUR PLOT

9-D: TILT REMOVED, CONTOUR PLOT



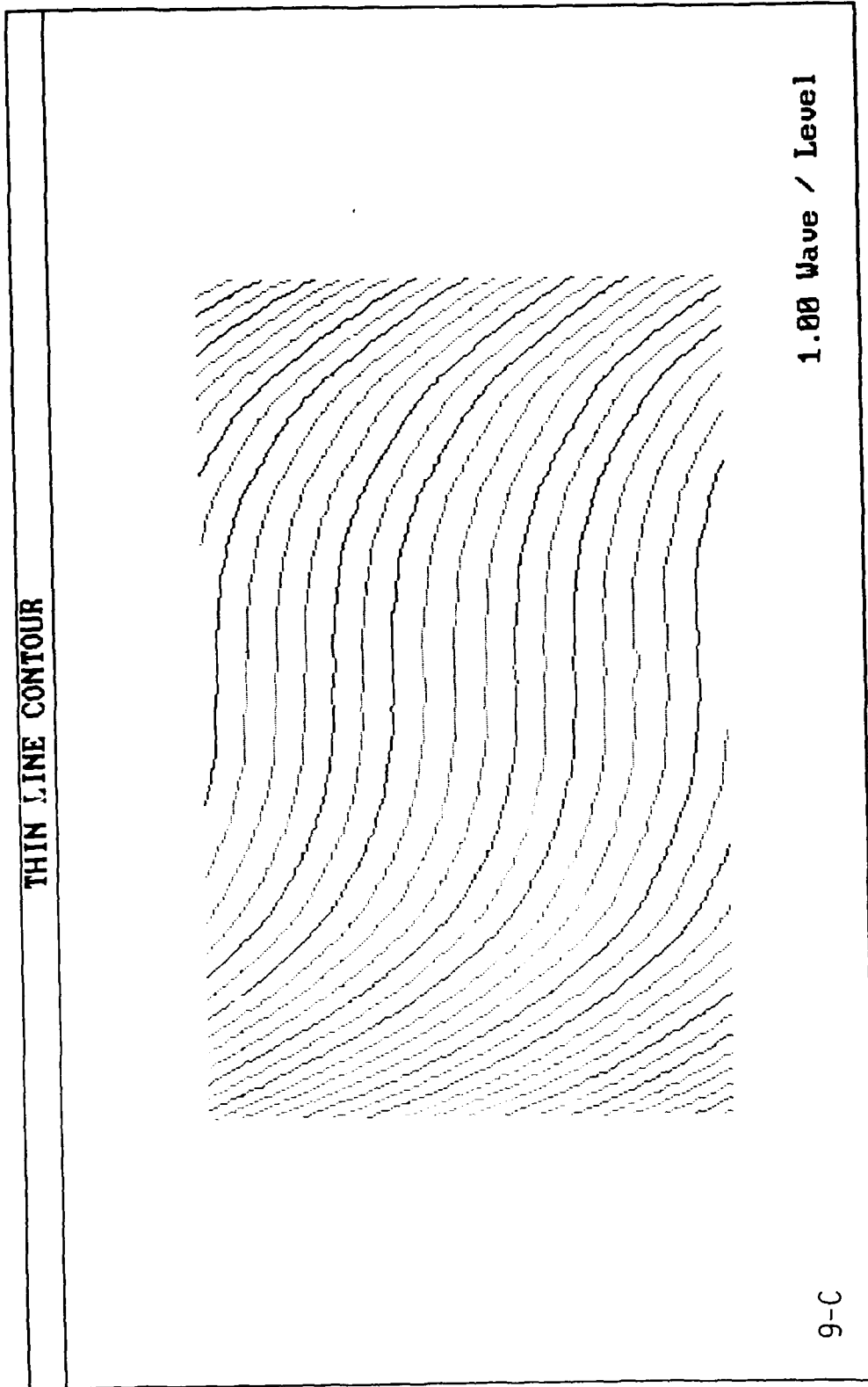
9-A

I 2 II



I 2 II

9-B



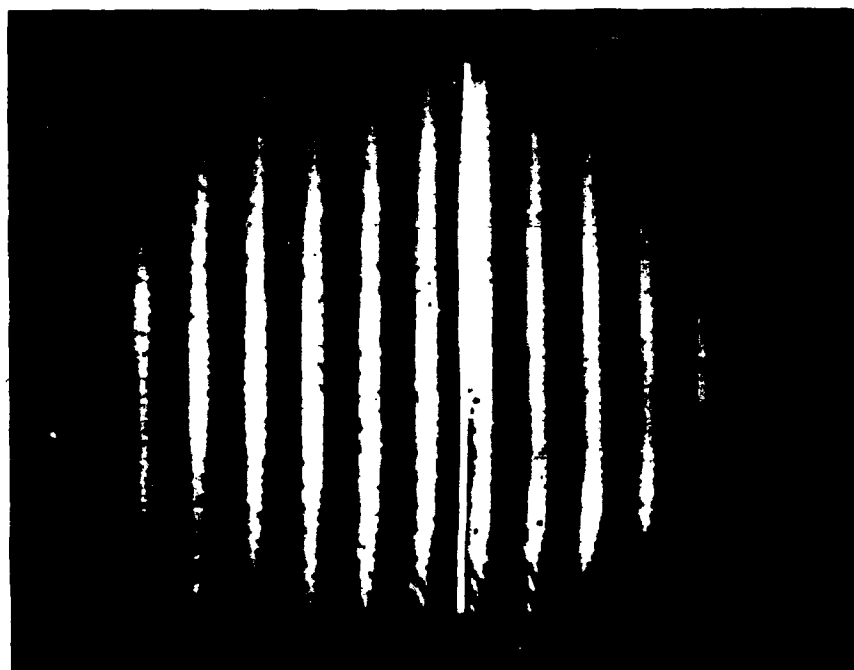


FIGURE 10: Y-SHEAR INTERFEROGRAM (STRAIGHT SIDE OF CYLINDER),
WITH MIRROR IN-FOCUS AND MISALIGNED BY 1°

FIGURE 11: PHASE MEASUREMENT OF Y-SHEAR INTERFEROGRAM (FIG.10)

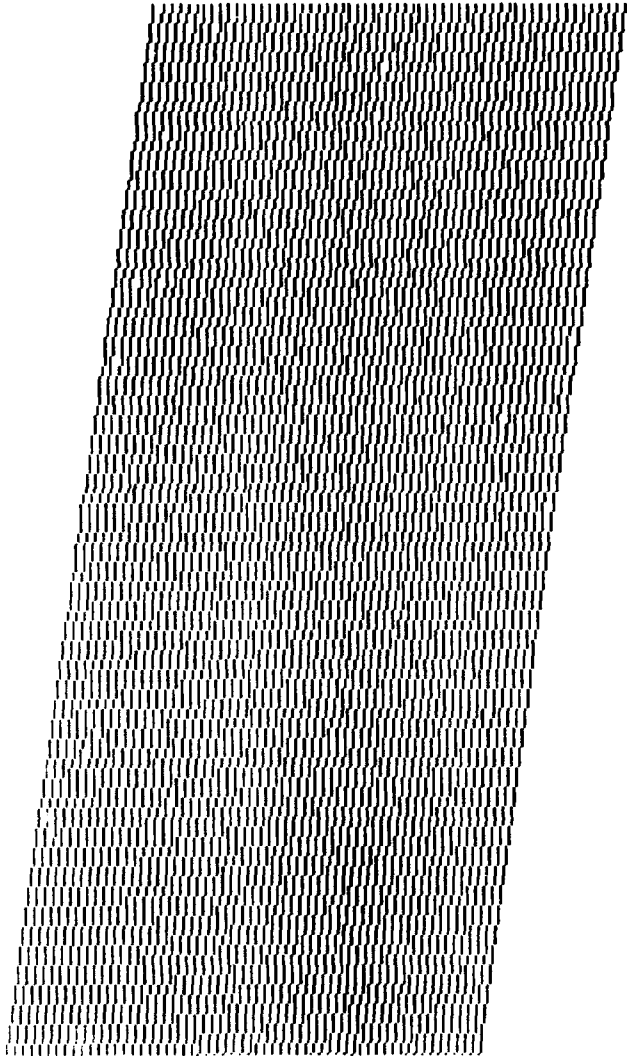
11-A: ORIGINAL, 3-D PLOT

11-B: TILT REMOVED, 3-D PLOT

11-C: ORIGINAL, CONTOUR PLOT

11-D: TILT REMOVED, CONTOUR PLOT

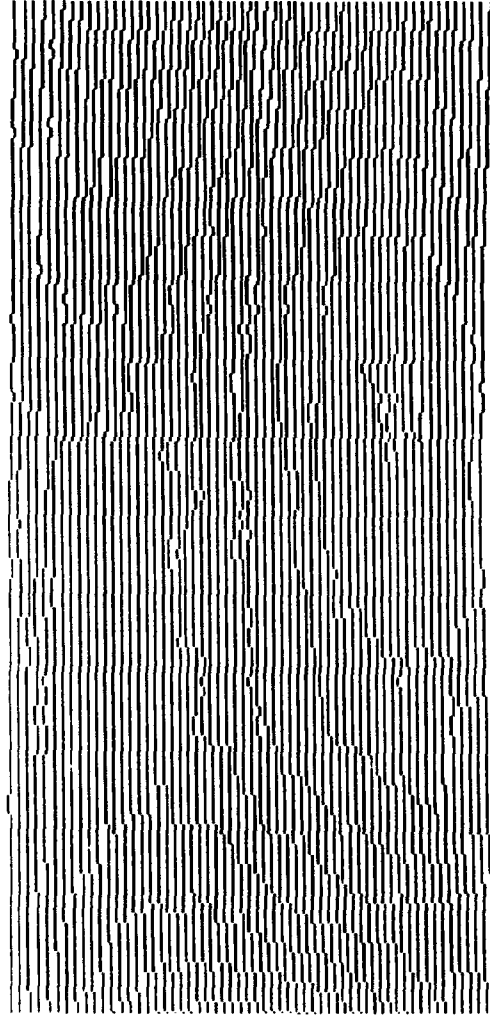
I 2 II



11-A

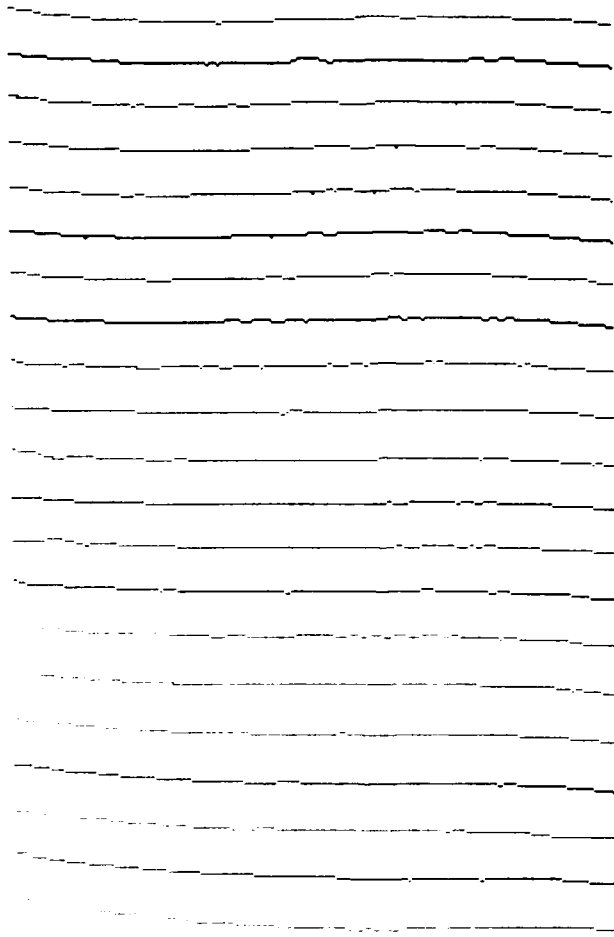


I 1 II



11-B

THIN LINE CONTOUR



1.00 Wave / Level

11-C

THIN LINE CONTOUR



0.20 Wave / Level

11-D



FIGURE 12: X-SHEAR INTERFEROGRAM (CURVED SIDE OF CYLINDER),
WITH MIRROR OUT-OF-FOCUS AND ALIGNED

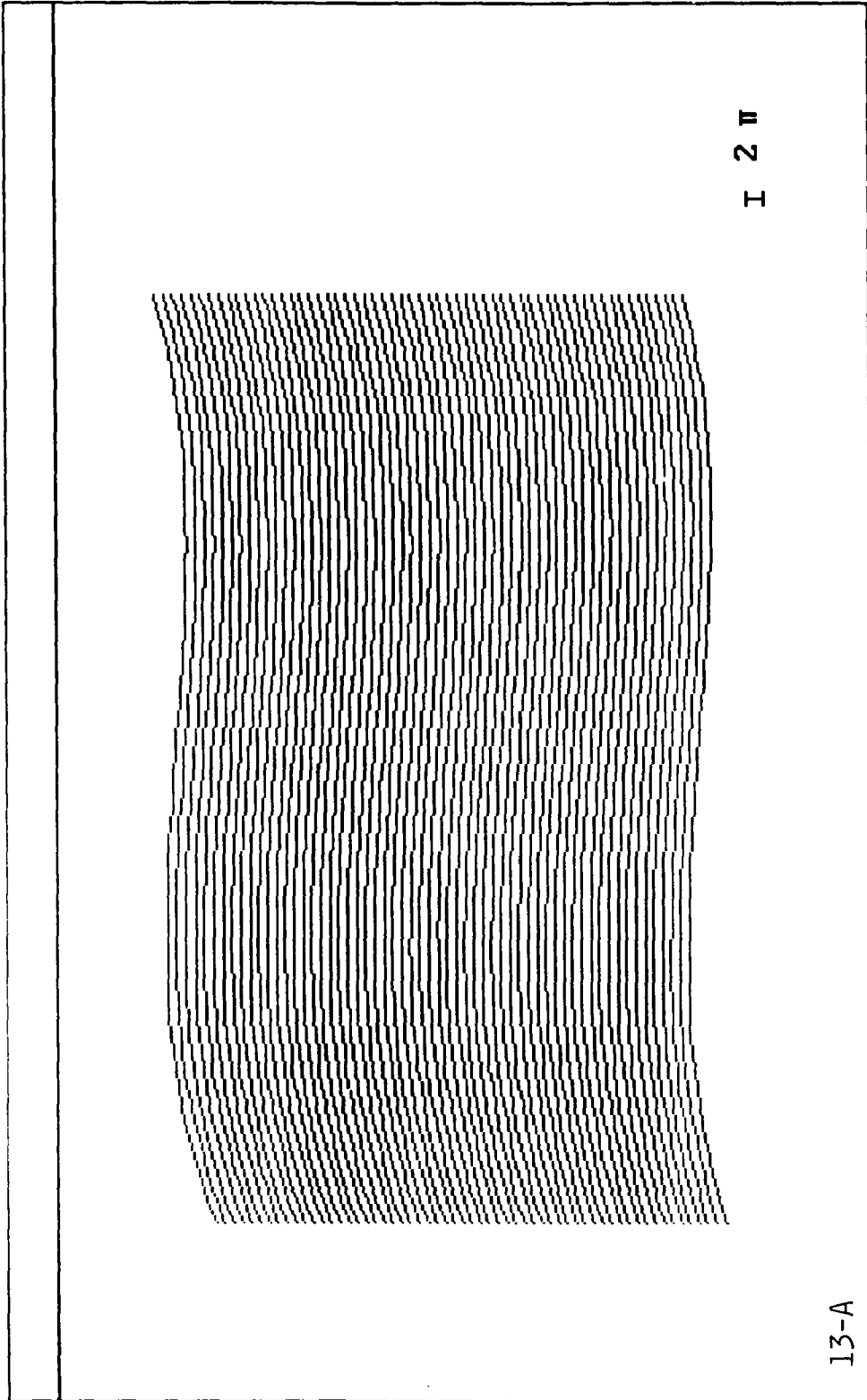
FIGURE 13: PHASE MEASUREMENT OF X-SHEAR INTERFEROGRAM (FIG. 12)

13-A: ORIGINAL, 3-D PLOT

13-B: TILT REMOVED, 3-D PLOT

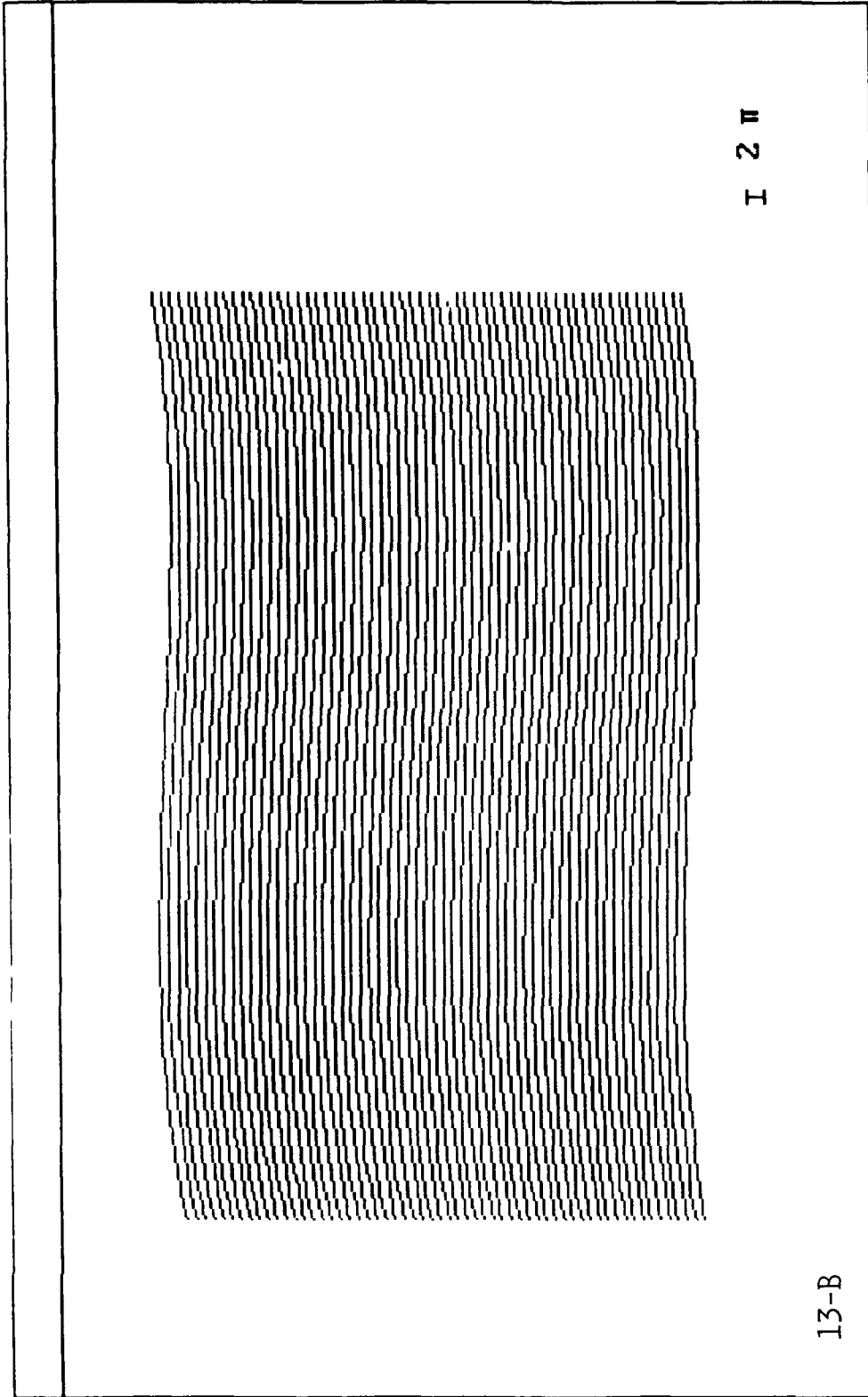
13-C: ORIGINAL, CONTOUR PLOT

13-D: TILT REMOVED, CONTOUR PLOT



I 2 n

13-A

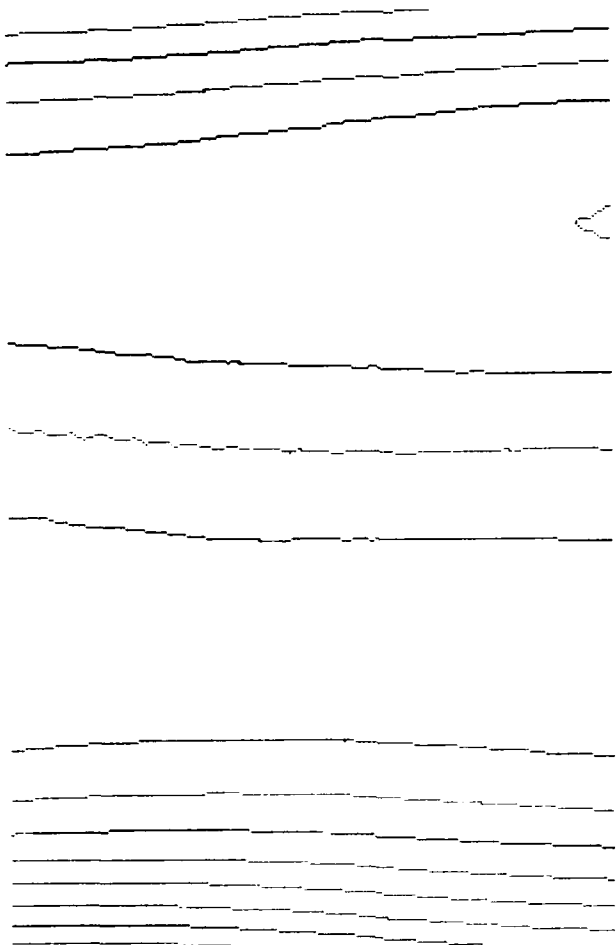


I 2 II

13-B



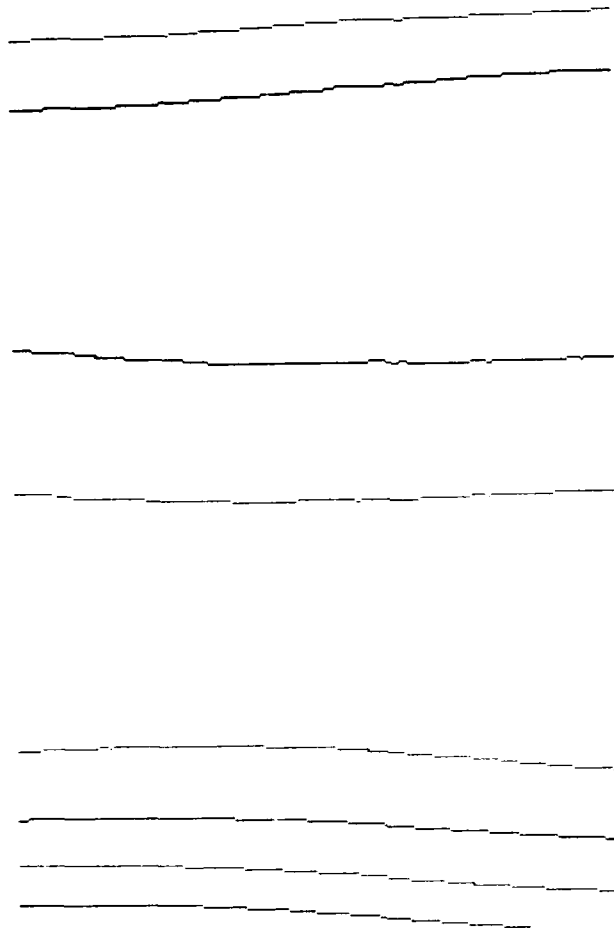
THIN LINE CONTOUR



1.00 Wave / Level

13-C

THIN LINE CONTOUR



1.00 Wave / Level

13-D

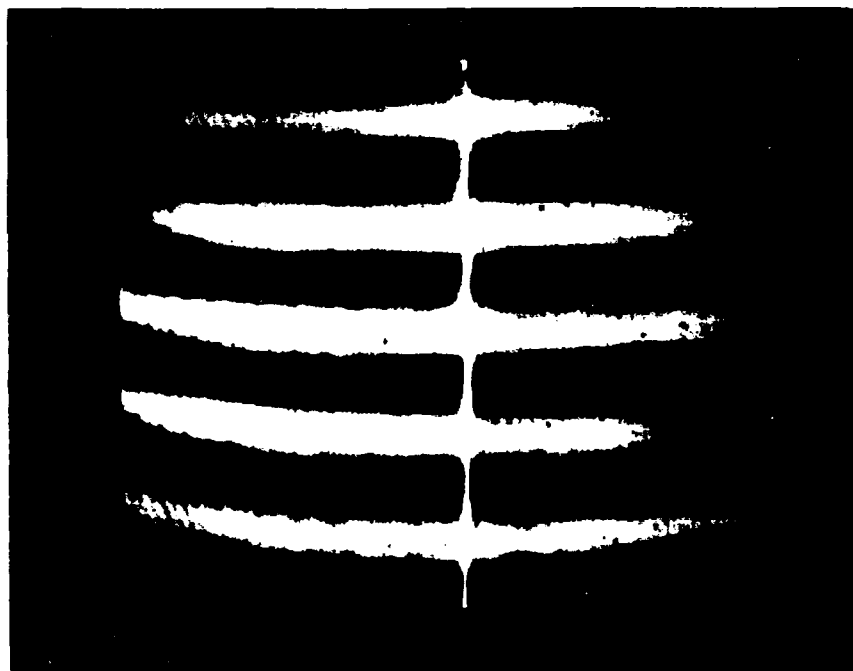


FIGURE 14: Y-SHEAR INTERFEROGRAM (STRAIGHT SIDE OF CYLINDER)
WITH MIRROR OUT-OF-FOCUS AND ALIGNED

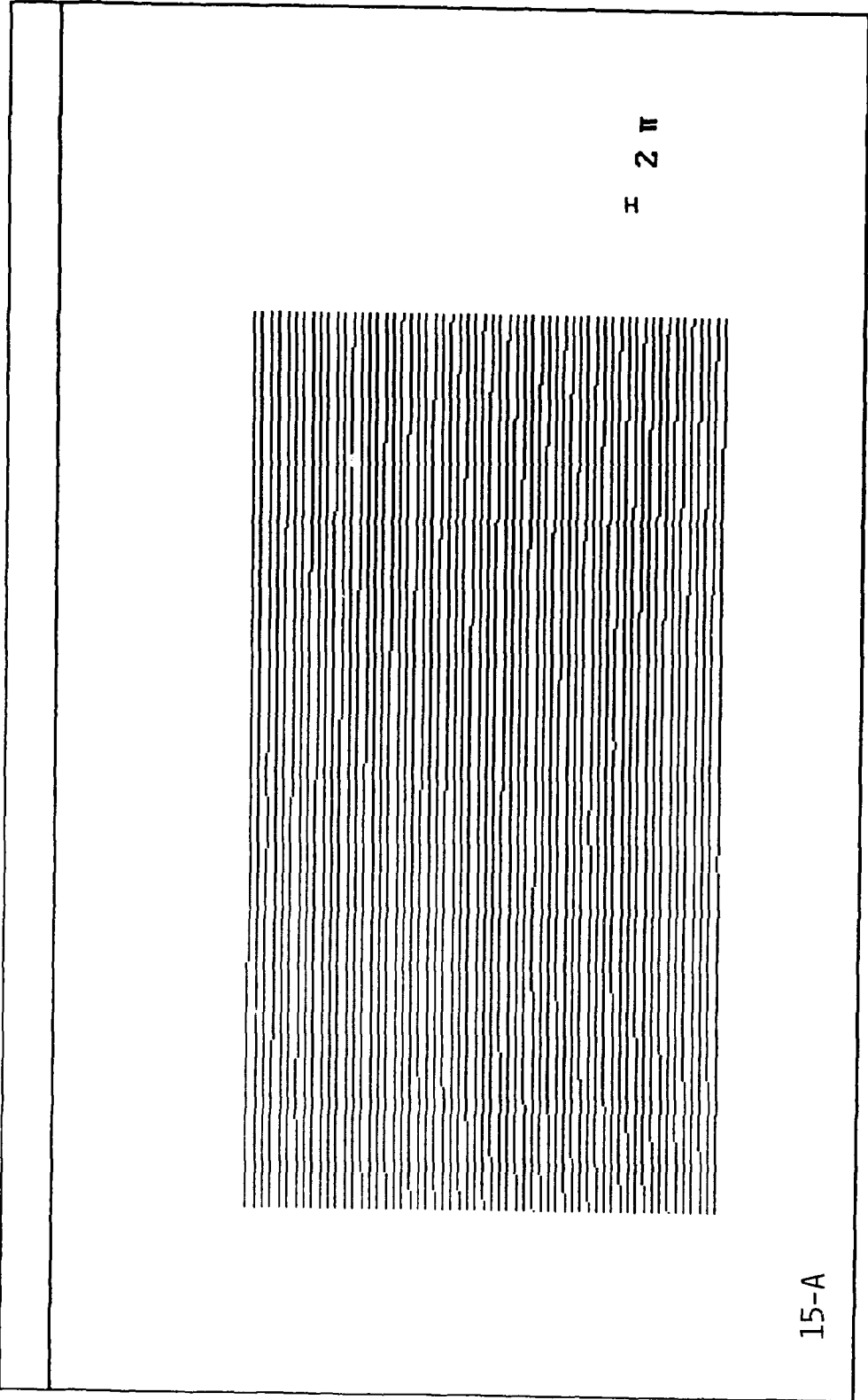
FIGURE 15: PHASE MEASUREMENT OF Y-SHEAR INTERFEROGRAM (FIG.14)

15-A: ORIGINAL, 3-D PLOT

15-B: TILT REMOVED, 3-D PLOT

15-C: ORIGINAL, CONTOUR PLOT

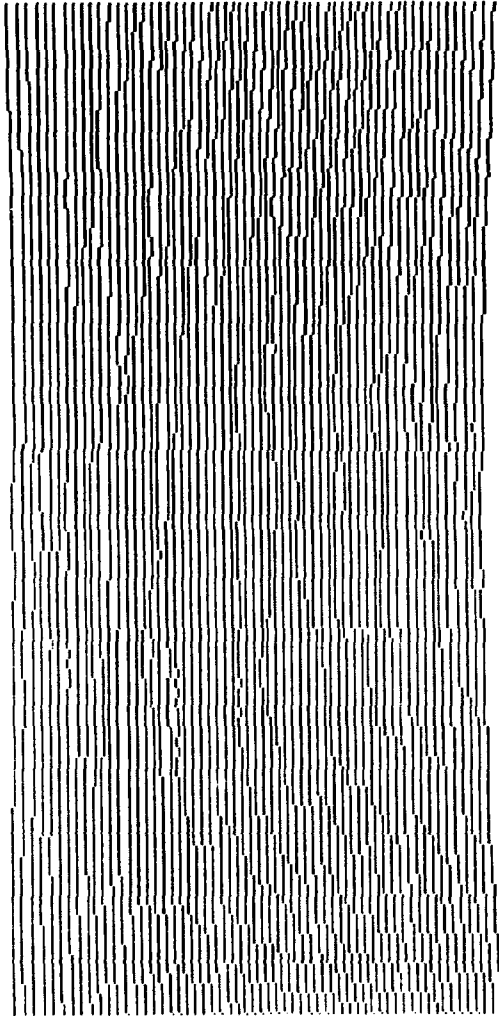
15-D: TILT REMOVED, CONTOUR PLOT



И 2 П
I

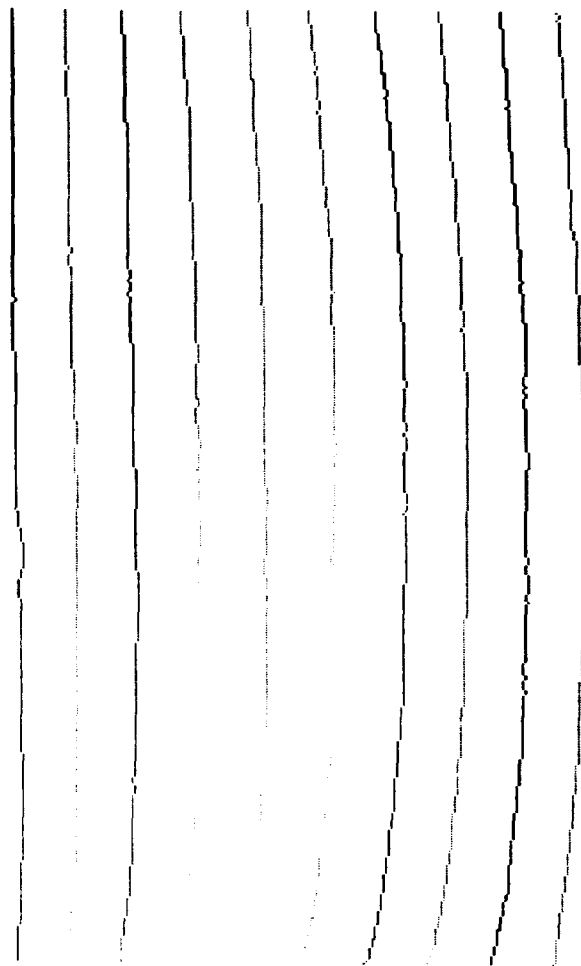
15-A

I 1 II



15-B

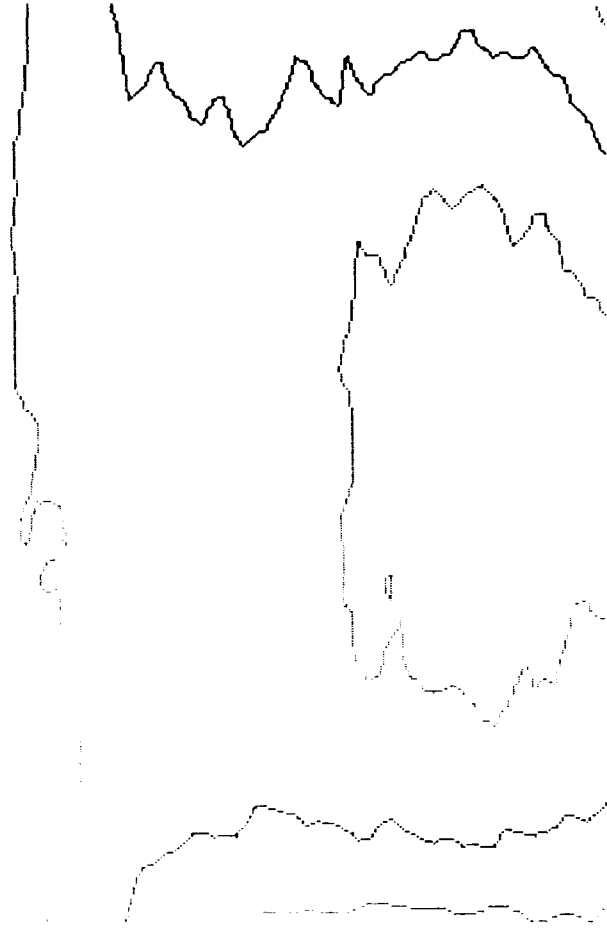
THIN LINE CONTOUR



1.00 Wave / Level

15-C

THIN LINE CONTOUR



0.20 Wave / Level

15-D

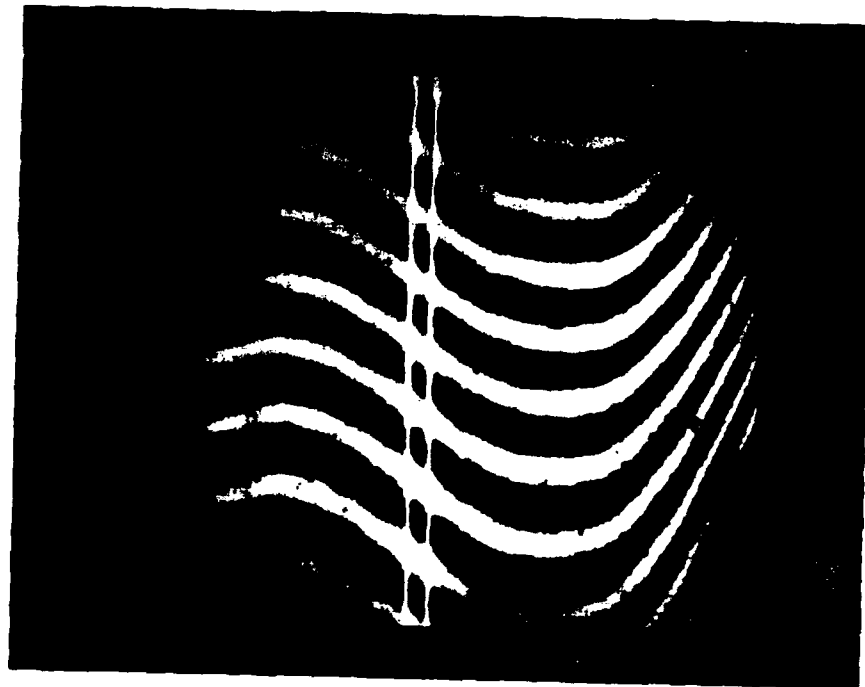


FIGURE 16: X-SHEAR INTERFEROGRAM (CURVED SIDE OF CYLINDER),
WITH MIRROR OUT-OF-FOCUS AND MISALIGNED BY 1°

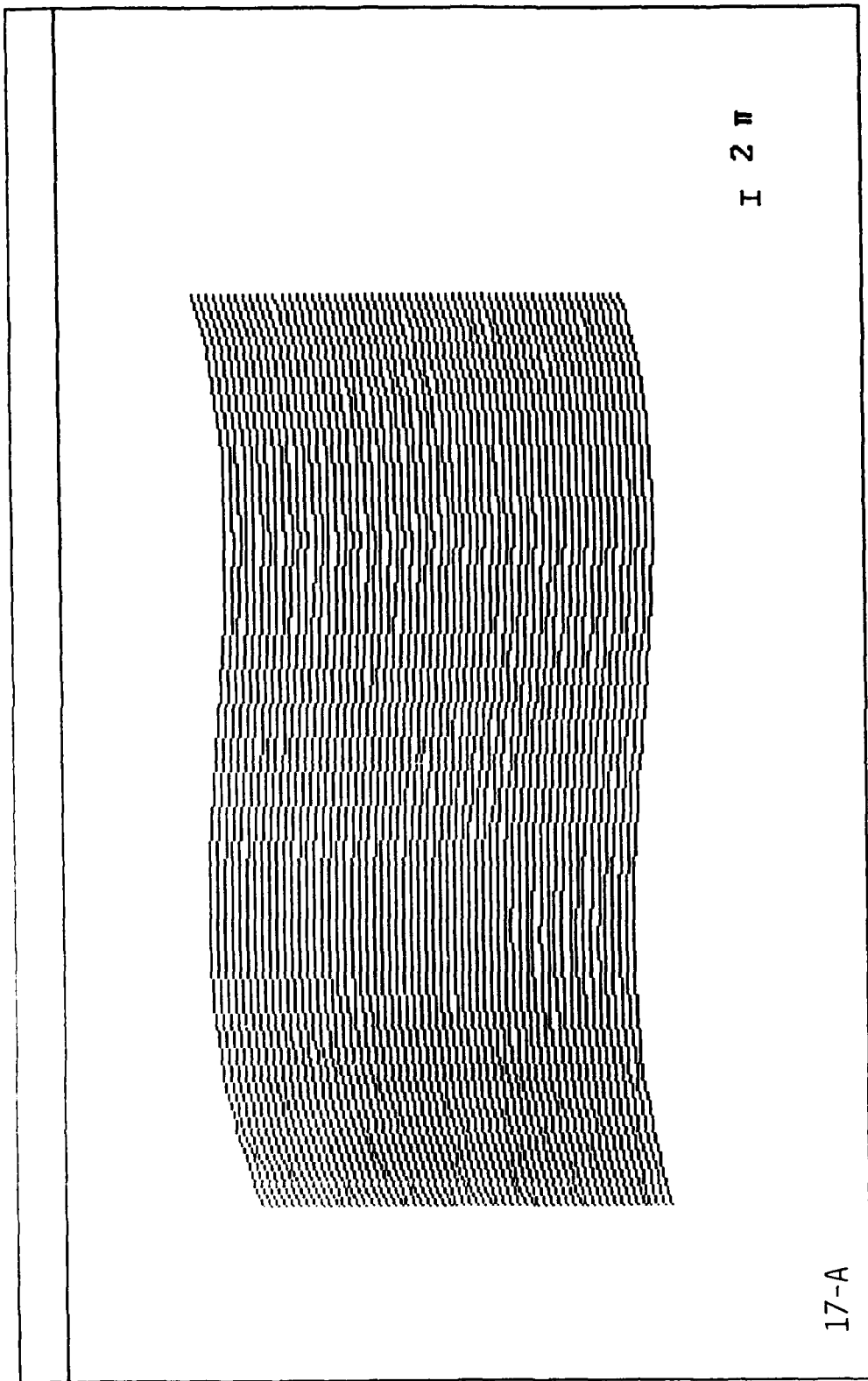
FIGURE 17: PHASE MEASUREMENT OF X-SHEAR INTERFEROGRAM (FIG.16)

17-A: ORIGINAL, 3-D PLOT

17-B: TILT REMOVED, 3-D PLOT

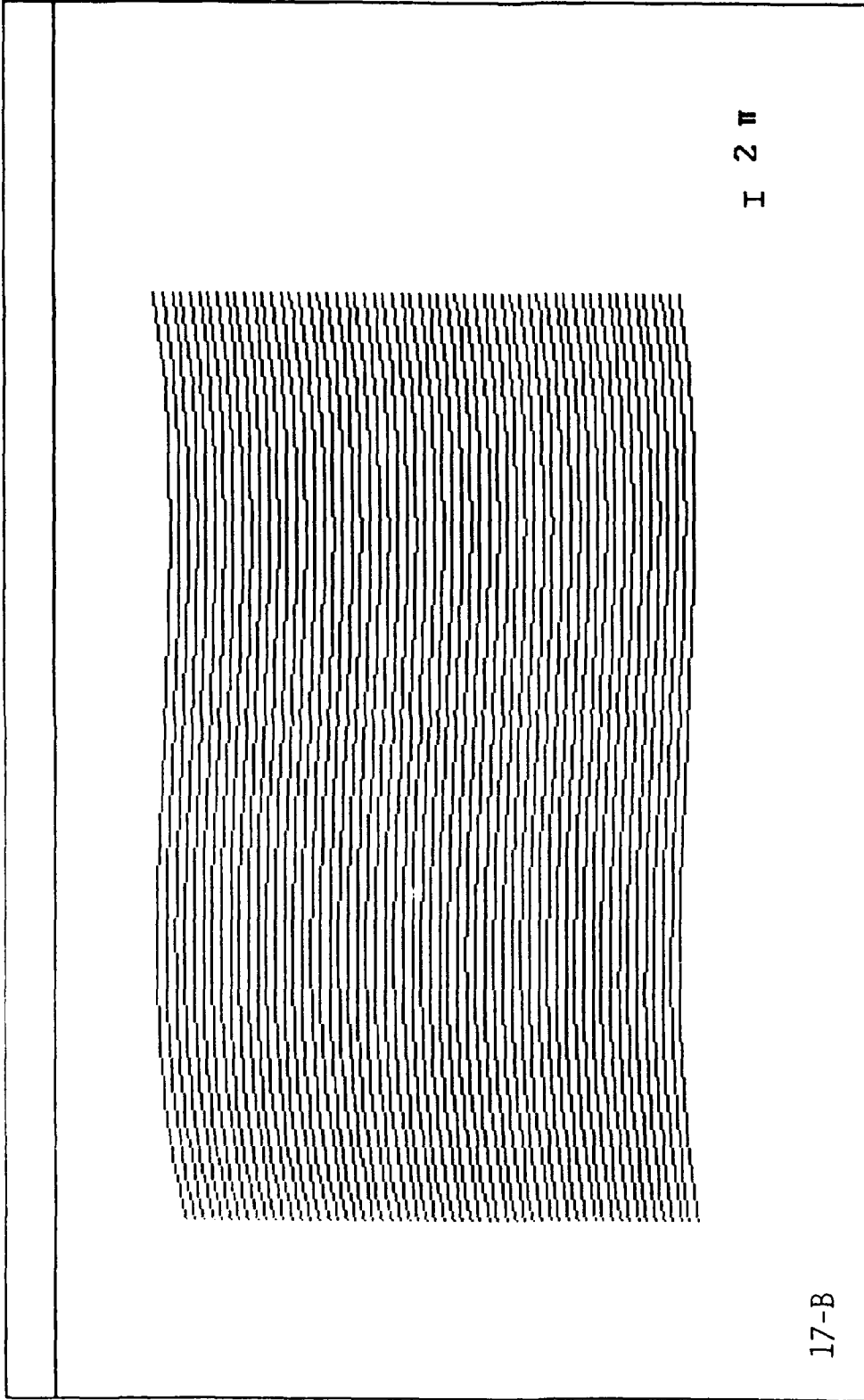
17-C: ORIGINAL, CONTOUR PLOT

17-D: TILT REMOVED, CONTOUR PLOT



I 2 II

17-A

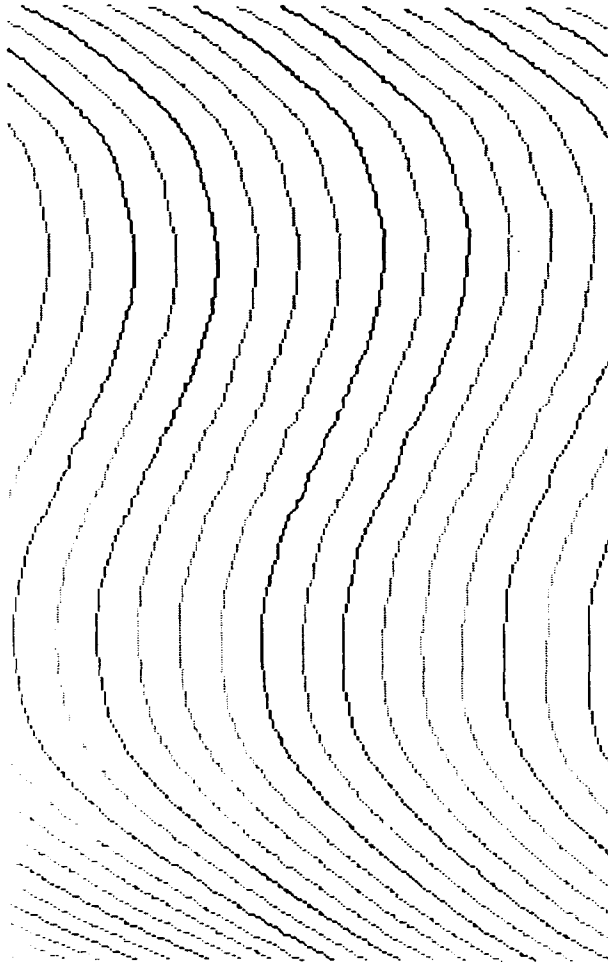


I 2 II

17-B



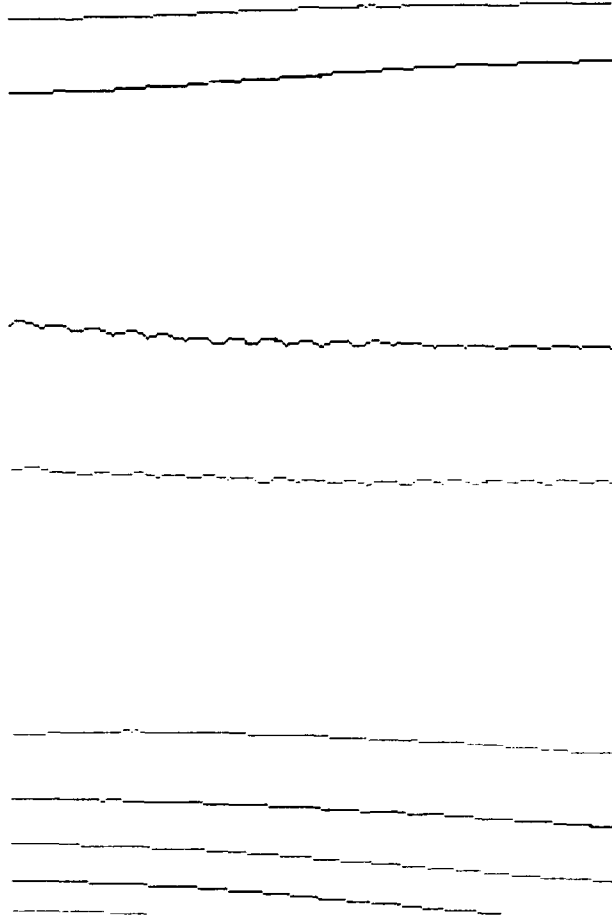
THIN LINE CONTOUR



1.00 Wave / Level

17-C

THIN LINE CONTOUR



17-D

1.00 Wave / Level

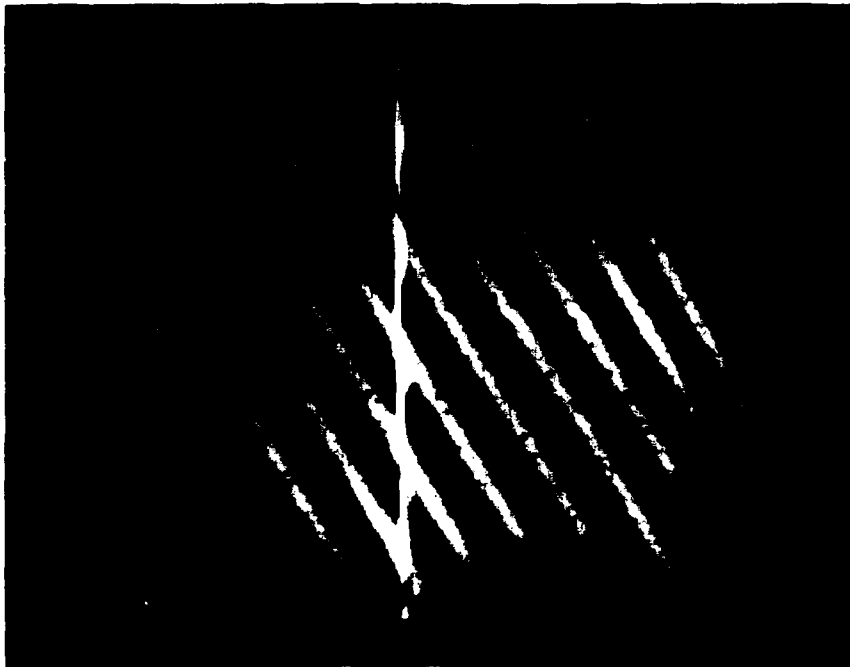


FIGURE 18: Y-SHEAR INTERFEROGRAM (STRAIGHT SIDE OF CYLINDER),
WITH MIRROR OUT-OF-FOCUS AND MISALIGNED BY 1°

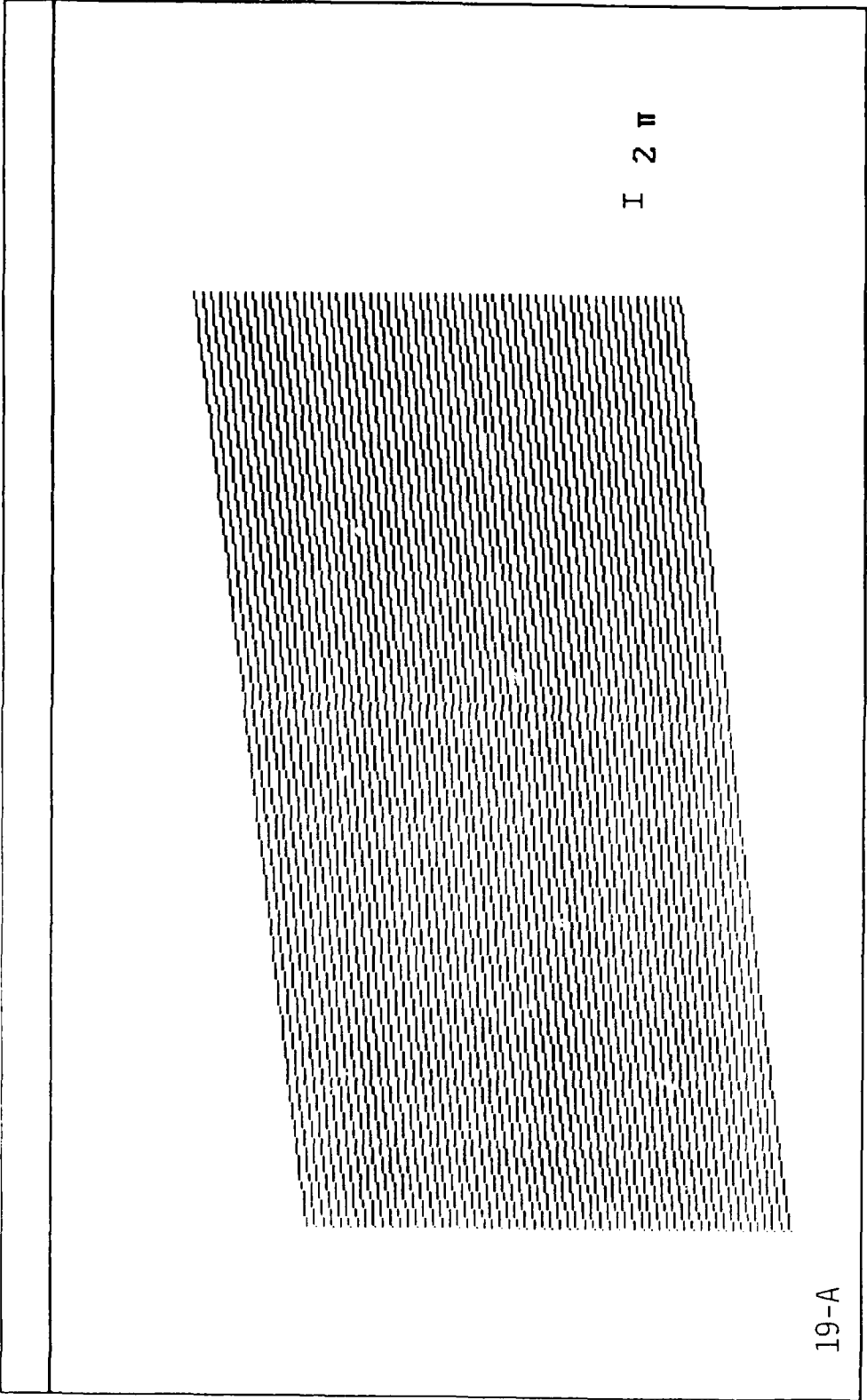
FIGURE 19: PHASE MEASUREMENT OF Y-SHEAR INTERFEROGRAM (FIG.18)

19-A: ORIGINAL, 3-D PLOT

19-B: TILT REMOVED, 3-D PLOT

19-C: ORIGINAL, CONTOUR PLOT

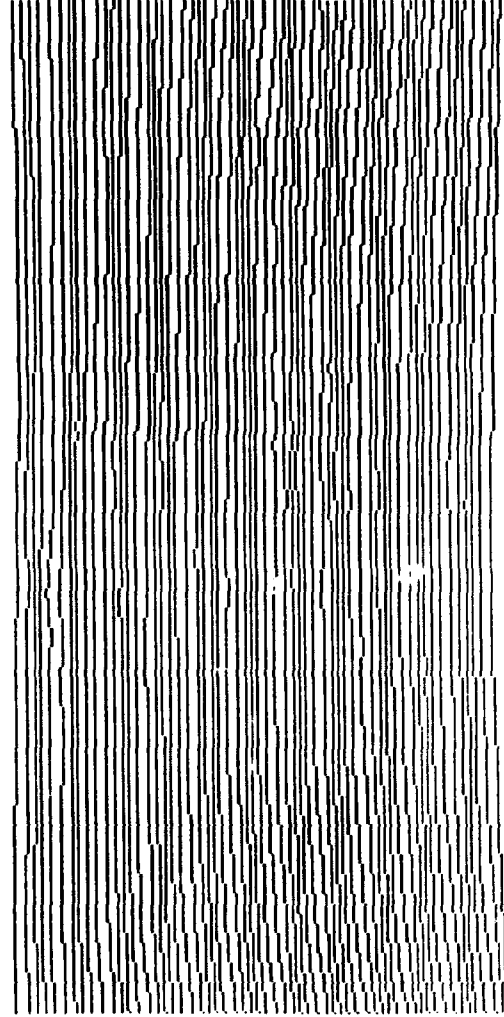
19-D: TILT REMOVED, CONTOUR PLOT



I 2 II

19-A

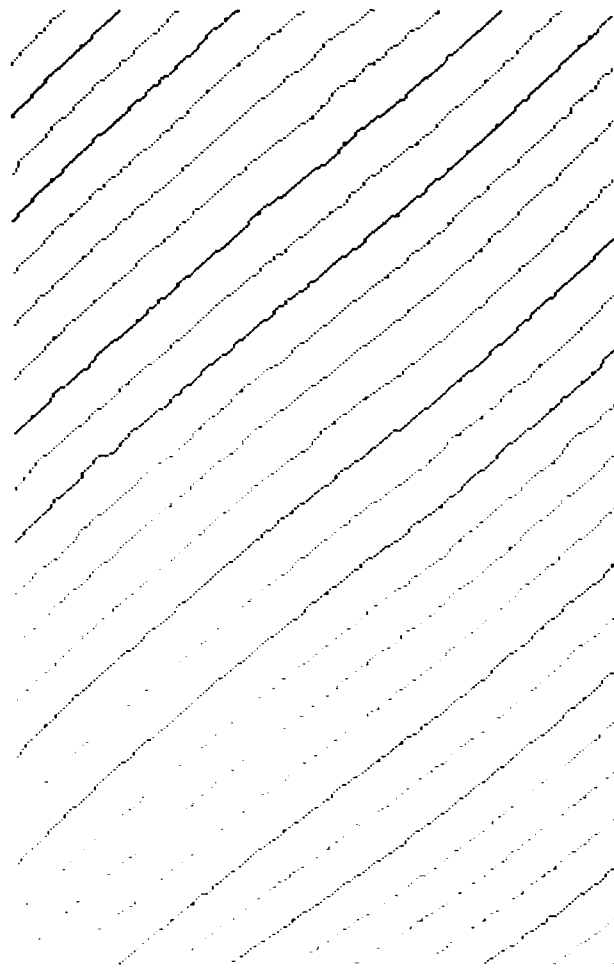
I 1 n



19-B



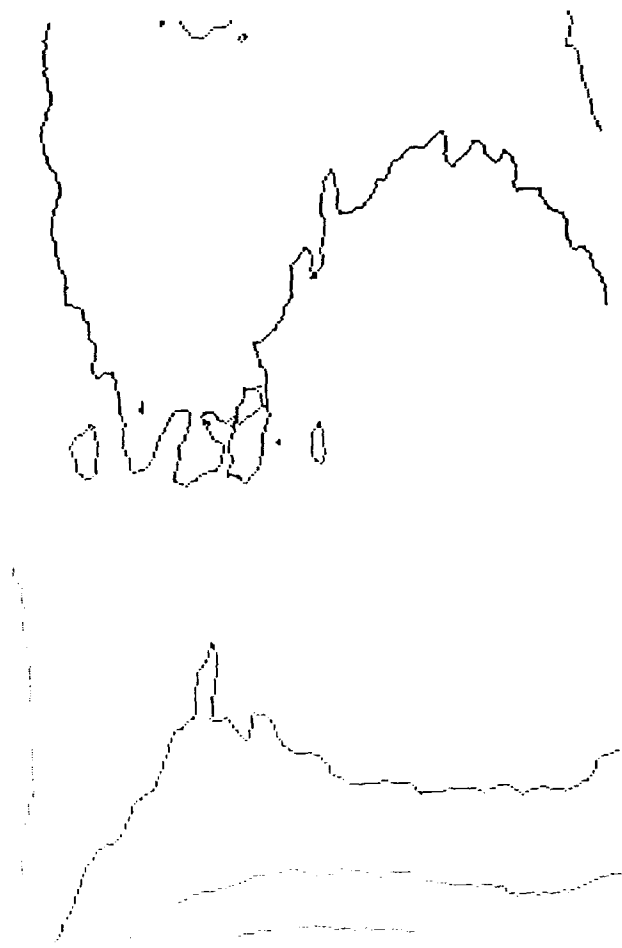
THIN LINE CONTOUR



1.00 Wave / Level

19-C

THIN LINE CONTOUR



0.20 Wave / Level

19-D

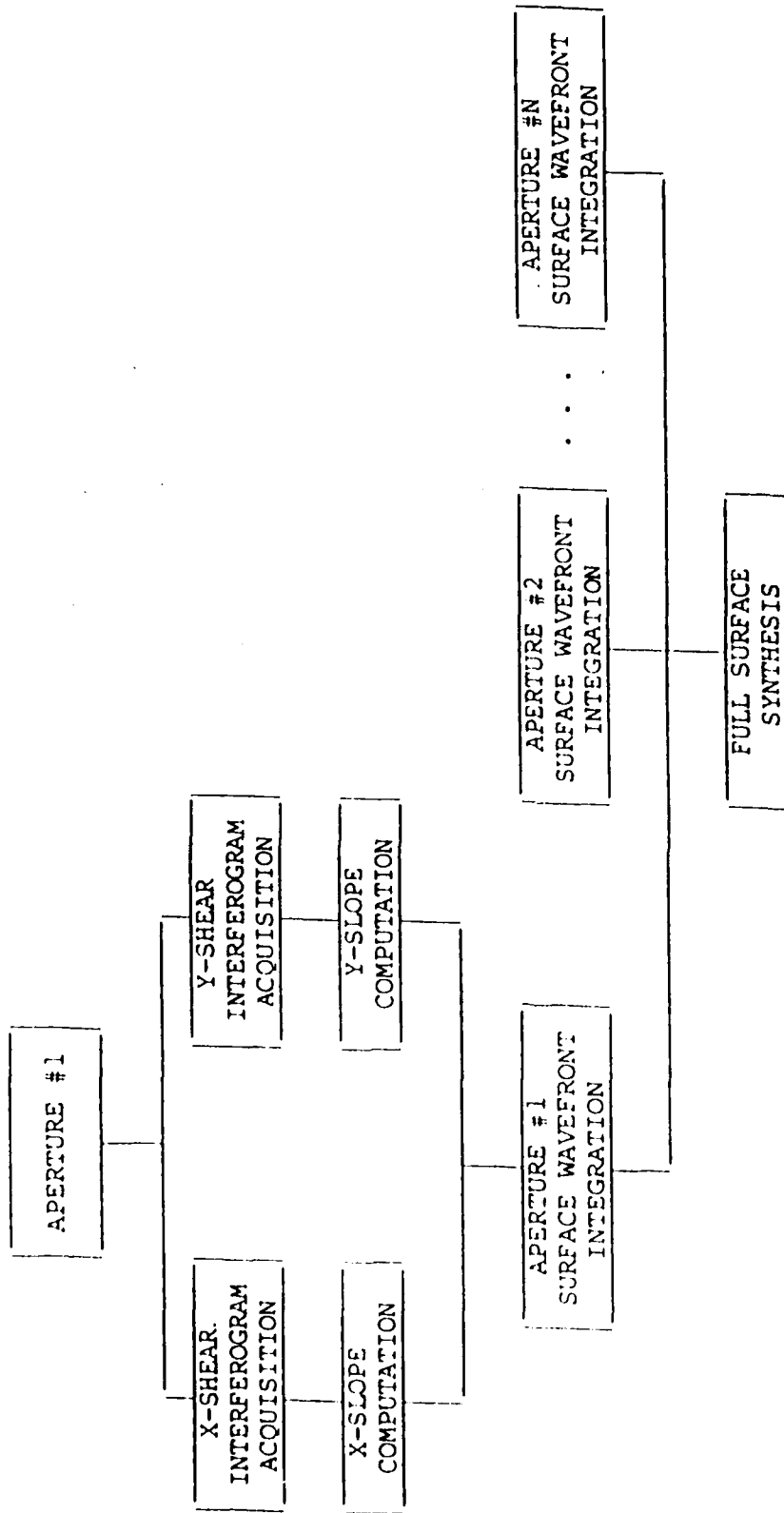
Software for Cylindrical surface Measurement

The source code software contained in this section was **specially** developed under this SDIO/ SBIR contract for use in the FSIS to carry out cylindrical surface measurement and describes the following processes:

- 1) Interferogram acquisition by means of the CCD camera.
- 2) Interferogram A/D conversion and storage on the frame buffer.
- 3) Fringe data processing to yield X and Y slope functions.
- 4) 3-D and contour displays.

The flow chart describing how the software processes data is given in figure 20.

The actual source code is contained on the next 11 pages after the flow chart.



Flow chart of preliminary software.

Figure 20

```
'*****  
'* CYLINDRICAL LENS MEASUREMENT PROGRAM *  
'*****
```

```
'*****  
' CONSTANTS AND VARIABLES AND ARRAYS  
'*****
```

```
PI=3.141596  
SIZE=240  
STEPX=4  
STEPY=4  
DEGREE=180/PI  
SCALE=5  
DIM ORDER(SIZE),Y1(SIZE),X(SIZE),Y(SIZE),P(SIZE),IP(60,60)  
DIM A(SIZE),B(SIZE),C(SIZE),D(SIZE),M(SIZE),M1(SIZE)
```

```
'*****  
' ARRAYS FOR CONTOUR MAP  
'*****
```

```
DIM SHARED z(150)  
DIM SHARED h(4)  
DIM SHARED ish(4)  
DIM SHARED xh(4)  
DIM SHARED yh(4)  
DIM SHARED im(3)  
DIM SHARED jm(3)  
DIM SHARED castab(2,2,2)  
im(0)=0 : im(1)=1 : im(2)=1 : im(3)=0  
jm(0)=0 : jm(1)=0 : jm(2)=1 : jm(3)=1  
for k=0 to 2  
for j=0 to 2  
for i=0 to 2  
read castab(k,j,i)  
next i  
next j  
next k  
data 0,0,8,0,2,5,7,6,9,0,3,4,1,0,1,4,3,0,9,6,7,5,2,0,8,0,0,0
```

```
SCREEN 9  
CLS
```

```
'*****  
' INTERFEROGRAM PARAMETER INPUTS  
'*****
```

```
50 INPUT "FILE NAME1";NAME1$  
INPUT "FILE NAME2";NAME2$  
INPUT "LEFT BOUND";LEFT  
INPUT "RIGHT BOUND";RIGHT  
INPUT "TOP BOUND";TOP  
INPUT "BOTTOM BOUND";BOTTOM  
XC=INT(RIGHT-LEFT)/2  
YC=INT(RIGHT-LEFT)/2
```

```
IF STEPY<3 THEN  
PRINT "PLEASE ENTER STEP Y > 3"  
GOTO 50  
END IF  
IF LEN(NAME1$)>7 THEN  
PRINT "PLEASE ENTER FILE NAME LENGTH < 8"  
GOTO 50  
END IF  
IF LEN(NAME2$)>7 THEN
```

```

PRINT "PLEASE ENTER FILE NAME LENGTH < 8"
GOTO 50
END IF
IF LEFT<1 THEN
PRINT "PLEASE ENTER LEFT BOUNDARY > 0 "
GOTO 50
END IF
IF RIGHT>SIZE THEN
PRINT "PLEASE ENTER RIGHT BOUNDARY <"SIZE
GOTO 50
END IF
IF TOP<1 THEN
PRINT "PLEASE ENTER TOP BOUNDARY < 1"
GOTO 50
END IF
IF BOTTOM>240 THEN
PRINT "PLEASE ENTER BOTTOM BOUNDARY < 240
GOTO 50
END IF

```

```

'*****
' SETUP IMAGE BUFFER PARAMETERS
'*****

```

```

OPEN "LOAD2V" FOR OUTPUT AS #1
PRINT #1,"L S I"
PRINT #1,"P X "SIZE;" 3 60 I N V 1 Q E P Q"
PRINT #1,"P D N Q"
PRINT #1,"M B 1 Q F R "NAME1$+"1.DAT Q"
PRINT #1,"M B 1 Q F R "NAME1$+"1.DAT Q"
PRINT #1,"M B 1 Q F R "NAME1$+"2.DAT Q"
PRINT #1,"M B 1 Q F R "NAME1$+"3.DAT Q"
PRINT #1,"M B 1 Q F R "NAME1$+"4.DAT Q"
PRINT #1,"QU"
CLOSE #1

```

```

'*****
' CONTROL IMAGE BUFFER AND ACQUIRE IMAGES
'*****

```

```

INPUT "DATA LOADED IN BUFFER (Y/N)";Y$
IF Y$="n" OR Y$="N" THEN
SHELL "LOAD2V.BAT"
CLS
END IF

```

```

'*****
' OPEN IMAGE #1 SET INPUT/OUTPUT FILES
'*****

```

```

P1$=NAME1$+"1.DAT"
P2$=NAME1$+"2.DAT"
P3$=NAME1$+"3.DAT"
P4$=NAME1$+"4.DAT"

```

```

OPEN P1$ AS #1 LEN=SIZE
OPEN P2$ AS #2 LEN=SIZE
OPEN P3$ AS #3 LEN=SIZE
OPEN P4$ AS #4 LEN=SIZE

```

```

FIELD #1,SIZE AS A$
FIELD #2,SIZE AS B$
FIELD #3,SIZE AS C$
FIELD #4,SIZE AS D$

```

```

'*****
' COMPUTE CENTER COLUMN LINE DATA
'*****

```

```

FOR I=BOTTOM TO TOP STEP -1
  GET #1,I
  GET #2,I
  GET #3,I
  GET #4,I
  E$=MID$(A$,YC,1)
  F$=MID$(B$,YC,1)
  G$=MID$(C$,YC,1)
  H$=MID$(D$,YC,1)
  A=ASC(E$)
  B=ASC(F$)
  C=ASC(G$)
  D=ASC(H$)
  AN=B-D
  DEN=A-C
  IF DEN=0 THEN DEN=0.0001
  T=ABS(AN/DEN)
  P=ATN(T)*DEGREE
  IF DEN<0 THEN 100
  IF AN<0 THEN P=360-P:GOTO 200
  GOTO 200
100  IF AN<0 THEN P=P+180:GOTO 200
     P=180-P
200  P(I)=P
     ORDER(I)=0
NEXT I

```

```

'*****
' ASSIGN THE INTERFEROGRAM FRINGE ORDERS
'*****

```

```

C=0
FOR I=YC TO BOTTOM
  IF P(I+1)-P(I)<-180 THEN C=C+1
  IF P(I+1)-P(I)>180 THEN C=C-1
  ORDER(I+1)=C
NEXT I

```

```

FOR I=YC TO TOP STEP -1
  IF I=YC THEN GOTO 300
  IF P(I+1)-P(I)<-180 THEN C=C-1
  IF P(I+1)-P(I)>180 THEN C=C+1
300 ORDER(I)=C
NEXT I

```

```

'*****
' COMPUTE ROW LINE DATA AND DISPLAY
'*****

```

```

ICOUNT=0
FOR I=TOP TO BOTTOM STEP Y
  ICOUNT=ICOUNT+1
  GET #1,I
  GET #2,I
  GET #3,I
  GET #4,I
  FOR J=LEFT TO RIGHT
    E$=MID$(A$,J,1)
    F$=MID$(B$,J,1)
    G$=MID$(C$,J,1)
    H$=MID$(D$,J,1)

```

```

      A(J)=ASC(E$)
      B(J)=ASC(F$)
      C(J)=ASC(G$)
      D(J)=ASC(H$)
NEXT J
FOR J=LEFT TO RIGHT
  AN=B-D
  DEN=A-C
  IF DEN=0 THEN DEN=0.0001
  T=ABS(AN/DEN)
  P=ATN(T)*DEGREE
  IF DEN<0 THEN 400
  IF AN<0 THEN P=360-P:GOTO 500
  GOTO 500
400   IF AN<0 THEN P=P+180:GOTO 500
      P=180-P
500   P(J)=P
NEXT J

```

```

'*****
' ASSIGN THE INTERFEROGRAM FRINGE ORDERS
'*****

```

```

C=ORDER(I)
FOR J=XC TO RIGHT
  IF P(J+1)-P(J)<-180 THEN C=C+1
  IF P(J+1)-P(J)>180 THEN C=C-1
  Y1(J+1)=P(J+1)+C*360
NEXT J
C=ORDER(I)
FOR J=XC TO LEFT STEP -1
  IF P(J+1)-P(J)<-180 THEN C=C-1
  IF P(J+1)-P(J)>180 THEN C=C+1
  Y1(J)=P(J)+C*360
NEXT J

```

```

'*****
' DISPLAY THE INTERFEROGRAM DATA
'*****

```

```

FOR J=LEFT TO RIGHT
  X2=J*1.5+(I-TOP)/2
  Y2=Y1(J)/SCALE
  IF J MOD STEPX THEN IP(ICOUNT,INT(J/4))=Y1(J)
  Y2=-Y2+120-(I-TOP)/2
  IF I=I1 THEN GOTO 700
  IF J=J1 THEN GOTO 1000
  GOSUB 10000
  GOTO 1000
700   M1(J)=Y2
      IF J=J1 THEN GOTO 1000
      LINE (X1,Y1)-(X2,Y2)
1000  X1=X2:Y1=Y2
NEXT J
FOR J=LEFT TO RIGHT
  JO=J+(I-TOP)/4
  M(JO)=M1(JO)
NEXT J
M(J2+1+(I-I1)/4)=200

```

```

NEXT I
1500 Y$=INKEY$:IF Y$="" THEN GOTO 1500

```

```

'*****
' DISPLAY CONTOUR MAP OF INTERFEROGRAM

```

'*****

GOSUB 20000
CLS
CLOSE

'*****
' SETUP IMAGE BUFFER PARAMETERS
'*****

OPEN "LOAD2V" FOR OUTPUT AS #1
PRINT #1,"L S I"
PRINT #1,"P X "SIZE;" 3 60 I N V 1 Q E P Q"
PRINT #1,"P D N Q"
PRINT #1,"M B 1 Q F R "NAME1\$+"1.DAT Q"
PRINT #1,"M B 1 Q F R "NAME1\$+"1.DAT Q"
PRINT #1,"M B 1 Q F R "NAME1\$+"2.DAT Q"
PRINT #1,"M B 1 Q F R "NAME1\$+"3.DAT Q"
PRINT #1,"M B 1 Q F R "NAME1\$+"4.DAT Q"
PRINT #1,"QU"
CLOSE #1

'*****
' CONTROL IMAGE BUFFER AND ACQUIRE IMAGES
'*****

INPUT "DATA LOADED IN BUFFER (Y/N)";Y\$
IF Y\$="n" OR Y\$="N" THEN
SHELL "LOAD2V.BAT"
CLS
END IF

'*****
' OPEN IMAGE #2 SET INPUT/OUTPUT FILES
'*****

P1\$=NAME2\$+"1.DAT"
P2\$=NAME2\$+"2.DAT"
P3\$=NAME2\$+"3.DAT"
P4\$=NAME2\$+"4.DAT"

OPEN P1\$ AS #1 LEN=SIZE
OPEN P2\$ AS #2 LEN=SIZE
OPEN P3\$ AS #3 LEN=SIZE
OPEN P4\$ AS #4 LEN=SIZE

FIELD #1,SIZE AS A\$
FIELD #2,SIZE AS B\$
FIELD #3,SIZE AS C\$
FIELD #4,SIZE AS D\$

'*****
' COMPUTE CENTER COLUMN LINE DATA
'*****

FOR I=BOTTOM TO TOP STEP -1
GET #1,I
GET #2,I
GET #3,I
GET #4,I
E\$=MID\$(A\$,YC,1)
F\$=MID\$(B\$,YC,1)
G\$=MID\$(C\$,YC,1)
H\$=MID\$(D\$,YC,1)

```

A=ASC(E$)
B=ASC(F$)
C=ASC(G$)
D=ASC(H$)
AN=B-D
DEN=A-C
IF DEN=0 THEN DEN=0.0001
T=ABS(AN/DEN)
P=ATN(T)*DEGREE
IF DEN<0 THEN 2100
IF AN<0 THEN P=360-P:GOTO 2200
GOTO 2200
2100 IF AN<0 THEN P=P+180:GOTO 2200
P=180-P
2200 P(I)=P
ORDER(I)=0
NEXT I

```

```

'*****
' ASSIGN THE INTERFEROGRAM FRINGE ORDERS
'*****

```

```

C=0
FOR I=YC TO BOTTOM
  IF P(I+1)-P(I)<-180 THEN C=C+1
  IF P(I+1)-P(I)>180 THEN C=C-1
  ORDER(I+1)=C
NEXT I

```

```

FOR I=YC TO TOP STEP -1
  IF I=YC THEN GOTO 2300
  IF P(I+1)-P(I)<-180 THEN C=C-1
  IF P(I+1)-P(I)>180 THEN C=C+1
2300 ORDER(I)=C
NEXT I

```

```

'*****
' COMPUTE ROW LINE DATA AND DISPLAY
'*****

```

```

ICOUNT=0
FOR I=TOP TO BOTTOM STEP STEPY
  ICOUNT=ICOUNT+1
  GET #1,I
  GET #2,I
  GET #3,I
  GET #4,I
  FOR J=LEFT TO RIGHT
    E$=MID$(A$,J,1)
    F$=MID$(B$,J,1)
    G$=MID$(C$,J,1)
    H$=MID$(D$,J,1)
    A(J)=ASC(E$)
    B(J)=ASC(F$)
    C(J)=ASC(G$)
    D(J)=ASC(H$)
  NEXT J
  FOR J=LEFT TO RIGHT
    AN=B-D
    DEN=A-C
    IF DEN=0 THEN DEN=0.0001
    T=ABS(AN/DEN)
    P=ATN(T)*DEGREE
    IF DEN<0 THEN 2400
    IF AN<0 THEN P=360-P:GOTO 2500

```

```

      GOTO 2500
2400   IF AN<0 THEN P=P+180:GOTO 2500
      P=180-P
2500   P(J)=P
      NEXT J

```

```

'*****
' ASSIGN THE INTERFEROGRAM FRINGE ORDERS
'*****

```

```

      C=ORDER(I)
      FOR J=XC TO RIGHT
        IF P(J+1)-P(J)<-180 THEN C=C+1
        IF P(J+1)-P(J)>180 THEN C=C-1
        Y1(J+1)=P(J+1)+C*360
      NEXT J

```

```

      C=ORDER(I)
      FOR J=XC TO LEFT STEP -1
        IF P(J+1)-P(J)<-180 THEN C=C-1
        IF P(J+1)-P(J)>180 THEN C=C+1
        Y1(J)=P(J)+C*360
      NEXT J

```

```

'*****
' DISPLAY THE INTERFEROGRAM DATA
'*****

```

```

      FOR J=LEFT TO RIGHT
        X2=J*1.5+(I-TOP)/2
        Y2=Y1(J)/SCALE
        IF J MOD STEPX THEN IP(ICOUNT,INT(J/4))=Y1(J)
        Y2=-Y2+120-(I-TOP)/2
        IF I=I1 THEN GOTO 2700
        IF J=J1 THEN GOTO 3000

```

```

      GOSUB 10000
      GOTO 3000
2700   M1(J)=Y2
      IF J=J1 THEN GOTO 3000
      LINE (X1,Y1)-(X2,Y2)
3000   X1=X2:Y1=Y2

```

```

      NEXT J
      FOR J=LEFT TO RIGHT
        J0=J+(I-TOP)/4
        M(J0)=M1(J0)
      NEXT J
      M(J2+1+(I-I1)/4)=200

```

```

NEXT I
3500 Y$=INKEY$:IF Y$="" THEN GOTO 3500

```

```

'*****
' DISPLAY CONTOUR MAP OF INTERFEROGRAM
'*****

```

```

GOSUB 20000
CLS
CLOSE
END

```

```

10000
'*****
' SUB PROGRAM OF HIDDEN LINE
'*****

```

```

      J0=J+(I-TOP)/4
      IF Y1<M(J0) THEN 10100

```

```

        IF J=J2 THEN 10050
        IF Y2=M(J0+1) THEN 10200
        RETURN
10050 M1(J0+1)=Y2
        LINE (X1,Y1)-(X2,Y2)
        RETURN
10100 IF J=J2 THEN 10100
        IF I2<M(J0+1) THEN 10200
        GOTO 10150
        M1(J0)=Y1
        M1(J0+1)=Y2
        LINE (X1,Y1)-(X2,Y2)
10150 M=(Y2-Y1)/(X2-X1)
        B=Y1-M*X1
        D=((Y1-M(J0))*(X2-X1))/(M(J0+1)-Y2+Y1-M(J0))
        X=X1+D
        Y=M*X+B
        LINE (X1,Y1)-(X,Y)
        M1(J0)=Y1
        RETURN
10200 M=(Y1-Y2)/(X1-X2)
        B=Y1-M*X1
        D=((M(J0)-Y1)*(X2-X1))/(Y2-M(J0+1)+M(J0)-Y1)
        X=X1+D
        Y=M*X+B
        LINE (X,Y)-(X2,Y2)
        M1(J0+1)=Y2
        RETURN

```

20000

```

'*****
' SUBROUTINE DISPLAY CONTOUR MAP OF INTERFEROGRAM
'*****

```

```

xdim=150
coldim=150
xn=60
yn=60
stepno=4
xf=left
yf=top
xs=stepx
ys=stepy
xsp=100
yjsp=50
redim x(xdim)
redim y(xdim)
redim colors(coldim)

```

```

' IP(2)      : z-value array
' xf        : start value of x-direction index
' yf        : start value of y-direction index
' xn        : # of point in x-direction
' yn        : # of ponit in y-direction
' stpno     : step no
' xs        : x-scale
' ys        : y-scale
' xsp       : start position of x-direction
' yjsp      : start position of y-direction

```

```

for i = 1 to xn
  x(i) = xsp + (i - 1) * xs
next i
for i = 1 to yn

```

```
y(i) = ysp + (i - 1) * ys
next i
```

```
mm = (xn - xf) mod stpno
xe = xn - mm - stpno
mm = (yn - yf) mod stpno
ye = yn - mm - stpno
```

```
if min <= 0 and max >= 0 then
  exlevel = 1
else
  exlevel = 0
end if
'***** main process *****
```

```
mainp:
if que$ = "y" then
  input "line contour step(wave):";stepcon
  yrow=4:xcol=25
  end if
if stepcon=0 then goto 10202
zma = abs(max) / (360*stepcon)
zmi = abs(min) / (360*stepcon)
zma = int(zma)
zmi = int(zmi)
mmi = min mod (360*stepcon)
mma = max mod (360*stepcon)
levell = zma + zmi
level = levell + exlevel
if min > 0 and min < (360*stepcon) then
  for i = 0 to level - 1
    z(i) = min - mmi + (i + 1) * (360*stepcon)
    colors(i)=i mod 16
  next i
else
  for i = 0 to level - 1
    z(i) = min - mmi + i * (360*stepcon)
    colors(i)=i mod 16
  next i
end if
locate 6,xmm: print cmd$
locate yrow%+2,xcol%:print "STEP:"stepcon
for j=yf to ye step -stpno
  if j<=top or j=>bottom then goto noneinbox
  for i=xf to xe step stpno
    if i<LEFT or i>RIGHT then goto noneinbox
    if (IP(i,j)<IP(i,j+stpno)) then
      dmin=IP(i,j)
    else
      dmin=IP(i,j+stpno)
    end if
    if IP(i+stpno,j)<dmin then dmin=IP(i+stpno,j)
    if IP(i+stpno,j+stpno)<dmin then dmin=IP(i+stpno,j+stpno)
    if IP(i,j)>IP(i,j+stpno) then
      dmax=IP(i,j)
    else
      dmax=IP(i,j+stpno)
    end if
    if IP(i+stpno,j)>dmax then dmax=IP(i+stpno,j)
    if IP(i+stpno,j+stpno)>dmax then dmax=IP(i+stpno,j+stpno)
    if dmax<z(0) or dmin>z(level-1) then goto noneinbox
  for k=0 to level - 1
    if z(k)<dmin or z(k)>dmax then goto noneintri
  for m=4 to 0 step -1
```

```

if m>0 then
  h(m)=IP(i+stpno*im(m-1),j+stpno*jm(m-1))-z(k)
  xh(m)=x(i+stpno*im(m-1))
  yh(m)=y(j+stpno*jm(m-1))
end if
if m=0 then
  h(0)=(h(1)+h(2)+h(3)+h(4))/4
  xh(0)=(x(i)+x(i+stpno))/2
  yh(0)=(y(j)+y(j+stpno))/2
end if
if h(m)>0 then
  ish(m)=2
elseif (h(m)<0) then
  ish(m)=0
else
  ish(m)=1
end if
next m
swskip$ = "n"
for m=1 to 4
  if swskip$ = "y" then
    swskip$ = "n"
    go to case0
  end if
  m1=m:m2=0:m3=m+1
  if m3=5 then m3=1
  casetype=cint(castab(ish(m1),ish(m2),ish(m3)))
  if casetype=0 then
    swskip$ = "n"
    goto case0
  end if
on casetype goto case1,case2,case3,case4,case5,case6,case7,case8,case9
case1:
  x1=xh(m1):y1=yh(m1):x2=xh(m2):y2=yh(m2)
  goto drawit
case2:
  x1=xh(m2):y1=yh(m2):x2=xh(m3):y2=yh(m3)
  swskip$ = "y"
  goto drawit
case3:
  x1=xh(m3):y1=yh(m3):x2=xh(m1):y2=yh(m1)
  goto drawit
case4:
  x1=xh(m1):y1=yh(m1)
  x2=(h(m3)*xh(m2)-h(m2)*xh(m3))/(h(m3)-h(m2))
  y2=(h(m3)*yh(m2)-h(m2)*yh(m3))/(h(m3)-h(m2))
  savx = x2 : savy = y2
  goto drawit
case5:
  x1=xh(m2):y1=yh(m2)
  x2=(h(m1)*xh(m3)-h(m3)*xh(m1))/(h(m1)-h(m3))
  y2=(h(m1)*yh(m3)-h(m3)*yh(m1))/(h(m1)-h(m3))
  goto drawit
case6:
  x1=xh(m3):y1=yh(m3)
  if m > 1 then
    x2 = savx : y2 = savy
  else
    x2=(h(m2)*xh(m1)-h(m1)*xh(m2))/(h(m2)-h(m1))
    y2=(h(m2)*yh(m1)-h(m1)*yh(m2))/(h(m2)-h(m1))
  end if
  goto drawit
case7:
  if m > 1 then
    x1 = savx : y1 = savy

```

```

        else
            x1=(h(m2)*xh(m1)-h(m1)*xh(m2))/(h(m2)-h(m1))
            y1=(h(m2)*yh(m1)-h(m1)*yh(m2))/(h(m2)-h(m1))
        end if
        x2=(h(m3)*xh(m2)-h(m2)*xh(m3))/(h(m3)-h(m2))
        y2=(h(m3)*yh(m2)-h(m2)*yh(m3))/(h(m3)-h(m2))
        savx = x2 : savy = y2
        goto drawit
    case8:
        x1=(h(m3)*xh(m2)-h(m2)*xh(m3))/(h(m3)-h(m2))
        y1=(h(m3)*yh(m2)-h(m2)*yh(m3))/(h(m3)-h(m2))
        x2=(h(m1)*xh(m3)-h(m3)*xh(m1))/(h(m1)-h(m3))
        y2=(h(m1)*yh(m3)-h(m3)*yh(m1))/(h(m1)-h(m3))
        savx = x1 : savy = y1
        goto drawit
    case9:
        x1=(h(m1)*xh(m3)-h(m3)*xh(m1))/(h(m1)-h(m3))
        y1=(h(m1)*yh(m3)-h(m3)*yh(m1))/(h(m1)-h(m3))
        if m > 1 then
            x2 = savx : y2 = savy
        else
            x2=(h(m2)*xh(m1)-h(m1)*xh(m2))/(h(m2)-h(m1))
            y2=(h(m2)*yh(m1)-h(m1)*yh(m2))/(h(m2)-h(m1))
        end if
        drawit:
            x1 = int(x1)
            x2 = int(x2)
            y1 = int(y1)
            y2 = int(y2)
            line (x1,y1)-(x2,y2),colors(k)
    case0: next m

noneintri: next k
noneinbox: next i : next j
circle (X(XC),Y(YC)),abs(X(XC)-X(xc-radius-1)),7
10202     RETURN

```

Conclusion

We have demonstrated that the FSIS instrument has the capability to accurately and rapidly analyze the full figure and surface roughness characteristics of grazing incidence surfaces for X-ray and hard UV imaging as well as for other cylindrical type aspheric surfaces over a broad range of wavelengths. This is possible in an affordable, reliable, and rugged instrument. We are currently in the process of obtaining grazing incidence optics from some national laboratories engaged in X-ray and UV imaging. Currently, there is no other optical instrument that we are aware of which can accurately measure both the full surface figure as well as the surface macro-roughness. The need for the immediate application of this optical surface measuring technology to chemical lasers, excimer lasers, laboratory X-ray lasers, gamma ray lasers, and free electron lasers is well documented. Other applications include general scanning of surfaces for quality control, materials defects analysis, and as a general instrument for aspheric optical quality control.

This research project demonstrated that the FSIS can work reliably in many types of environments and can be produced for under \$65,000. Commercial production with user support and additional technical refinements can be marketed for less than \$100,000. This is a small investment for such a versatile quality control instrument that can quantitatively determine whether an optical component that will be used in X-ray and UV imaging has been made to specification.