CHAPTER 6

DATA PROCESSING

"Science cannot solve the ultimate mystery of nature. And it is because in the last analysis we ourselves are part of the mystery we are trying to solve."
M Planck

6.1 OVERVIEW OF THE DATA PROCESSING

Once the phase-stepping has been performed as detailed in Chapter 5 and parameters such as bar temperature, air temperature, pressure, humidity and CO_2 content have been measured, the data must be processed to produce results for: the central length of the bar (corrected to 20 °C), the form of the exposed measurement face, the flatness of the face and its parallelism with respect to the wrung face. This processing forms the major part of the computer program which controls the interferometer. The majority of the computer processing is devoted to extracting the phase data into a form in which it can be used in the multiple-wavelength analysis, to calculate the length of the bar.

The stages of the data processing are illustrated in figure 6.1. The raw data is stored in the framestore in the form of 15 images and a mask. The images are stored as 256 x 256 pixel arrays at a resolution of 8 bits (256 levels). The phase data is calculated from the images, one wavelength at a time, by applying the phase stepping equation (5.57) to each set of 5 images. The 3 phase maps then contain phase data in the range $-\pi$ to $+\pi$, including discontinuities of magnitude 2π at the boundaries between fringes and a discontinuity at the edge of the bar. The discontinuities are then removed by a 3-pass routine developed specifically for the interferometer. A surface is fitted to the data of the platen to account for any deviation from flatness so that the phase of the fitted surface can be subtracted from the phase data over the whole image to remove tilt. The resulting phase maps are scaled to fringe fractions by dividing by 2π. The phase maps are then representations of the difference in phase between the measured phase and the phase fitted to the surface of the platen, *i.e.* the phase data on the surface of the bar is now directly related to the fringe fractions required for length calculation.

The red phase map is then used to calculate the flatness of the exposed face of the bar and its parallelism to the wrung face by fitting a plane to the phase data. Fringe fractions at the centre of the bar are averaged for each of the 3 wavelengths and the

resulting fractions used in the multiple-wavelength calculations. The results are displayed on the screen with the option for hardcopy.

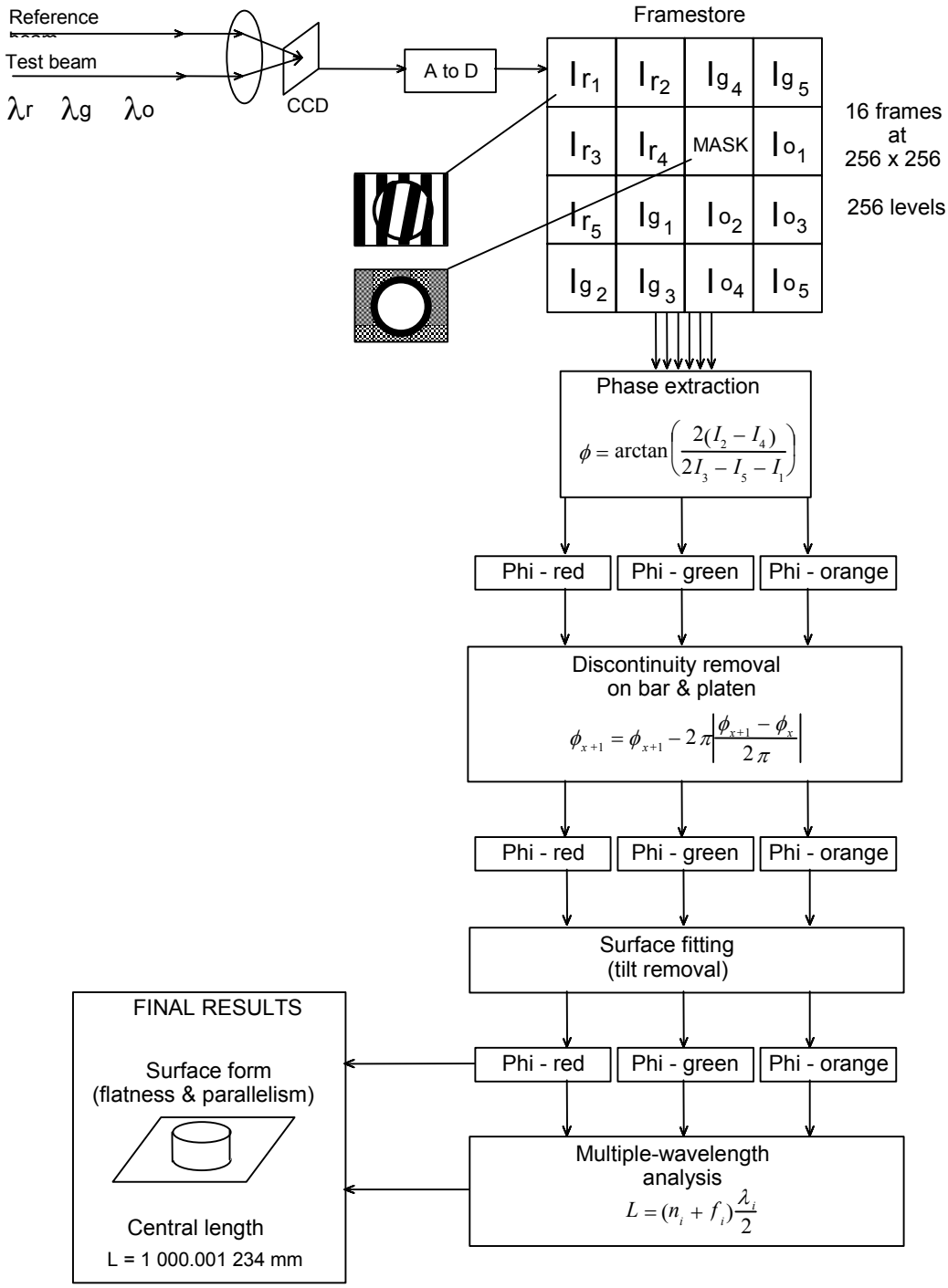


Figure 6.1 *- Flow diagram of data processing*

6.2 COMPUTING SYSTEM

The original computer used for the data analysis and instrument control was a Hewlett-Packard Vectra ES12, an 80286 PC compatible with a maths co-processor, VGA compatible display and a clock speed of 12 MHz. A dot-matrix printer was used for hard-copy. The following cards were installed inside the PC: A Matrox PIP1024B framestore, a CEC IEEE interface card, an Amplicon PC14A digital I-O card. Eventually the speed of the computer proved to be a limiting factor: a compilation of the program took up to 8 minutes and a single measurement (including set-up and analysis) took 6 minutes to perform.

The computer was replaced with an Elonex PC450, an 80486 DX2 system, operating at a clock-doubled speed of 50 MHz, with an on-board co-processor and 64 K cache. An S-VGA compatible monitor and a Hewlett-Packard DeskJet 550C colour inkjet printer were added.

With the PC450, it was possible to place DOS in High Memory and to set up a RAMDRIVE and SMARTDRIVE disc cache, enabling much faster compilation and program execution. The compilation time was reduced to 31 seconds and the measurement time to 2 minutes 17 seconds. The actual time taken for the calculations is 45 seconds, with the remainder of the time required for the setting up, phase-stepping and temperature measurements.

The program is approximately 4500 lines of Pascal code which is compiled by a Microsoft Pascal 4.0 compiler into an executable file of size 140 K. Microsoft Pascal was chosen as the programming language as it is well structured and could interface with the libraries of routines provided with the interface boards used inside the PC which were supplied as compiled Microsoft Pascal and C object modules.

The major limitation of Microsoft Pascal is that there is a maximum limit of 64 K allowed for data, with no single data structure allowed to be greater than this limit. Thus there were two problems for the data processing. Firstly, a 256 x 256 array of REAL numbers (which would be required for each 256 x 256 phase map) would be stored as 4 bytes per number, thus resulting in an array size of 256 K. Thus the images had to be sampled at 128 x 128 resolution to result in 64 K phase arrays. Secondly, since a maximum of 64 K was allowed for ALL variables, some alternative method had to be used to store the 3 phase maps and the phase-step map. These arrays were placed high in memory, *i.e.* outside the default data segment. Unfortunately this meant that they were not protected from being violated by other programs and some further programming was necessary to avoid clashes. Microsoft Pascal is not able to use extended or expanded memory, so only a maximum of 640 K was available to the program.

6.3 IMAGE PROCESSING

6.3.1 Interferogram digitisation

Each interferogram is digitised by being imaged onto the CCD array of a Sony AVC-D5CE monochrome video camera. The array size is 8.8 mm x 6.6 mm at 500 x 582 pixels. The image of the interferogram over-fills the array and only the central region is digitised - this avoids the inclusion of diffracted beams at the edge of the image. The camera is synchronised to the monitor signal derived from the Matrox framestore board ensuring that the image position is fixed with respect to the framestore pixels. The camera and framestore are connected with standard 75 Ω BNC cable.

The Matrox framestore is configured as a single store of size 1024 x 1024 pixels, with the zoom option enabled allowing digitisation to a 256 x 256 image (total 16 images) at 8 bit resolution. The organisation of the 16 image areas in the framestore is shown in figure 6.2. Fifteen images are used for storing the interferograms for the 3 wavelengths and the remaining image is used to store the 3-level mask. Access to the intensity data stored on the framestore is via library routines with speed of access below that of direct memory access of main computer RAM hence all calculations are performed on arrays stored in conventional memory.

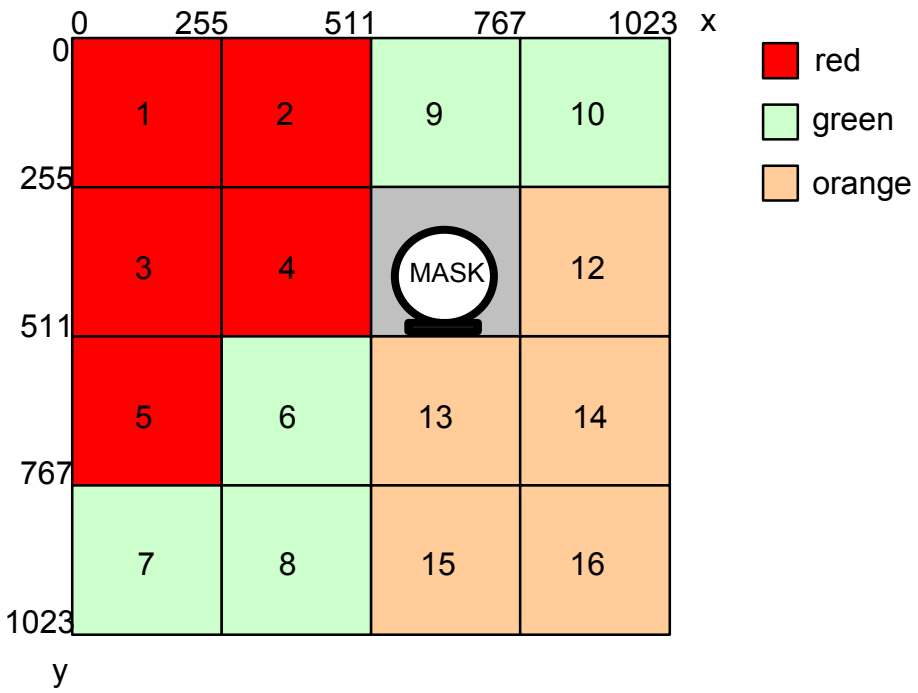


Figure 6.2 - Organisation of framestore memory

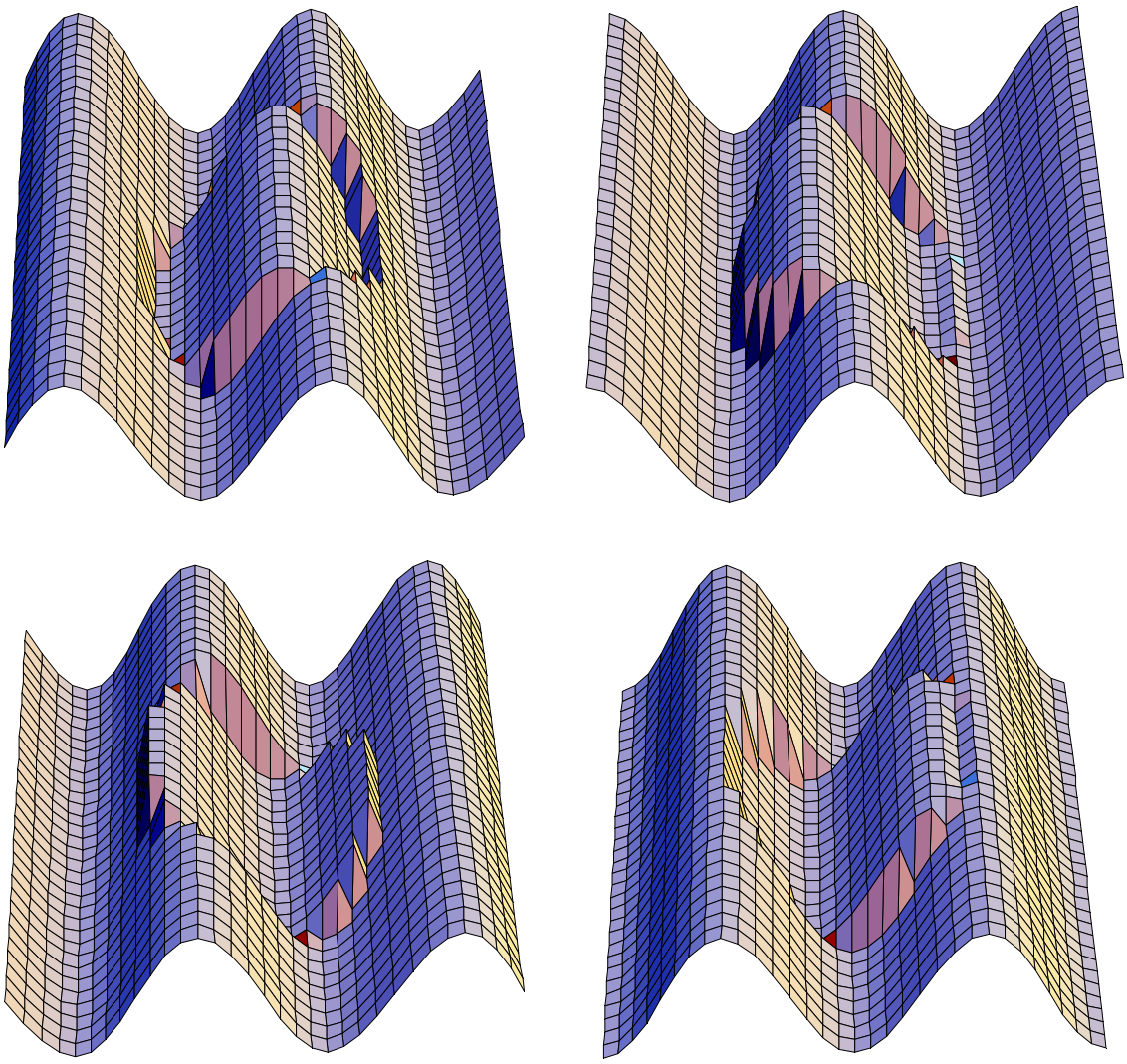


Figure 6.3 *- Simulated intensity arrays for the first 4 digitised images*

The gain and offset of the analogue to digital converter on the framestore card can be manually adjusted using the computer program so that most of the 256 level range of the framestore is used. Typical histograms of digitised interferograms are shown in figures 6.4 & 6.5. The extra peak at level 128 is due to the areas in the image where the surfaces are not aligned or smooth enough for interference to be visible, such as the supports, or the edge of the bar, and hence have an average intensity of half the digitised range. The widths of the low and high level peaks are dependent on the number of fringes in view: by adjusting the numbers of fringes it is possible to alter the relative widths of these peaks. When the fringes are adjusted such that a bright fringe is at both the left and right edges of the screen, the high intensity peak is widest. The most important feature is that almost the whole range of the digitiser is used, decreasing the

noise in the phase measurements due to quantisation noise in the digitiser (see § 5.3.7.4), but without the peaks being truncated at the extremes of the digitisation range.

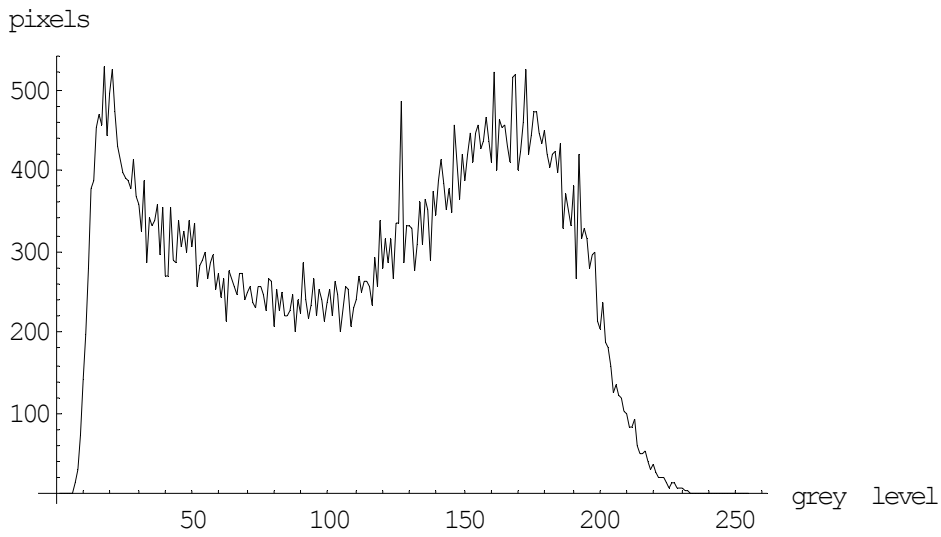


Figure 6.4 *- Typical digitisation histogram showing number of pixels within given intensity levels*

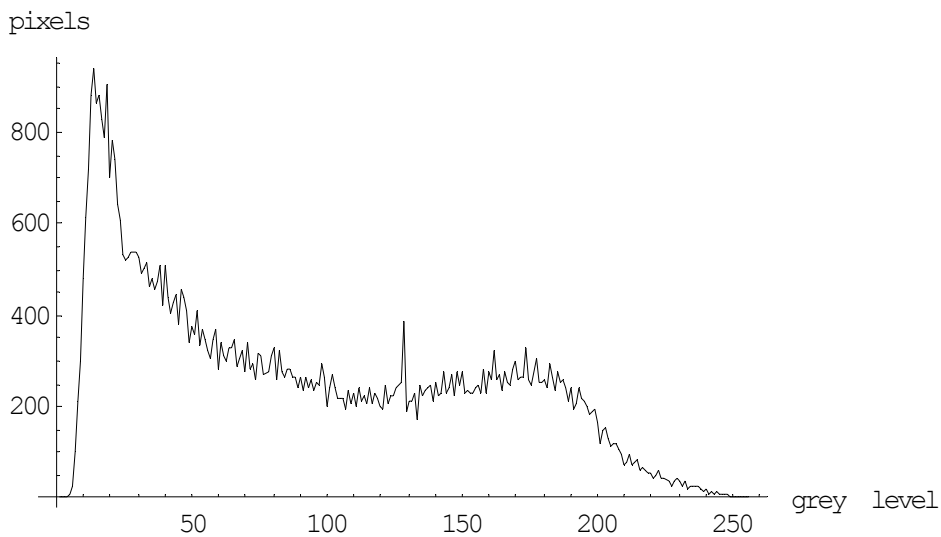


Figure 6.5 *- Typical digitisation histogram showing number of pixels within given intensity levels, but with more dark fringes in the image*

6.3.2 Phase extraction

After image digitisation and measurement of temperature, pressure *etc*, the phase is extracted from the 15 digitised images, one wavelength at a time by applying equation

6.1 to the intensity data stored in the framestore. The phase is extracted on a pixel-by-pixel basis, starting at the top left pixel, and progressing along to the end of the row, before extracting the next row. The results are stored in 3 arrays, one for each wavelength (`phi_red`, `phi_green`, `phi_orange`). Next, use is made of a 4-quadrant arctangent routine (described in § 5.3.3) which returns a value in the range -π to +π depending on the sign of the numerator and denominator in equation 6.1. At each pixel, the actual phase-step value (nominally 90° or π/2) is also calculated and stored in an array (`alpha`).

$$\phi(x,y) = \arctan\left[\frac{2\left(I_2(x,y) - I_4(x,y)\right)}{2I_3(x,y) - 2I_5(x,y) - I_1(x,y)}\right] \tag{6.1}$$

After phase extraction, the three arrays `phi_red`, `phi_green` and `phi_orange` contain phase information including tilt and 2π discontinuities at the boundary between fringes. A simulated example of one of these phase maps is shown in figure 6.6.

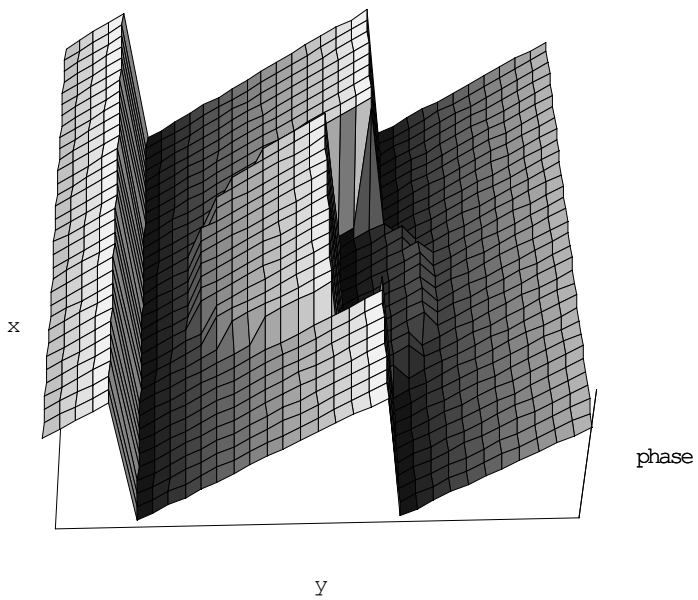


Figure 6.6 *- Simulated phase map containing 2π discontinuities, tilt, and phase difference due to the bar*

6.3.3 Discontinuity removal

The modulo 2π discontinuities in each phase-map are removed on a line-by-line basis using a three-pass routine. Initially, a three-level mask is generated by the user and stored in the framestore before any measurements are made. This mask is used in the phase-unwrapping to distinguish between (i) data on the platen, (ii) data on the end of the bar and (iii) invalid data such as at the edges of the bar which are radiused.

During the phase-unwrapping, the phase is unwrapped first for data corresponding to the surface of the platen. The first line to be unwrapped is the top horizontal line of the image. The unwrapping algorithm scans across this line on a pixel-by-pixel basis, removing discontinuities of magnitude ~ 2π by adding or subtracting multiples of 2π to the pixel following the discontinuity.

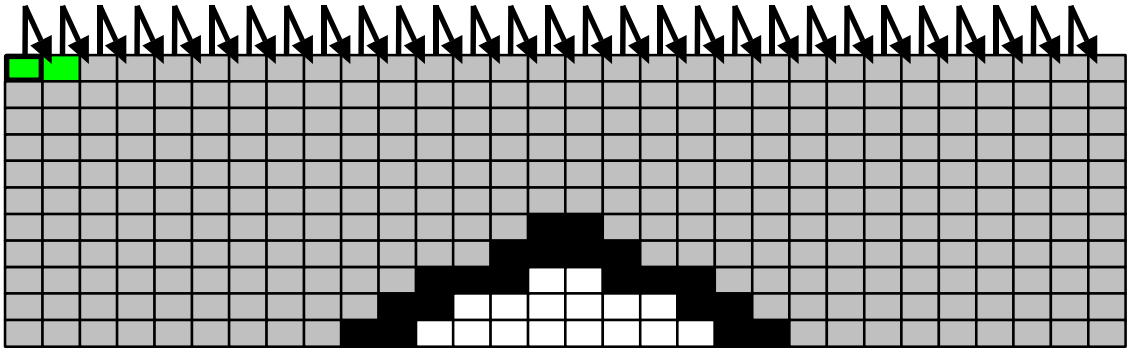


Figure 6.7(a) *- Discontinuity removal 1st pass - removal of discontinuities across top line of image using pixel-by-pixel comparison*

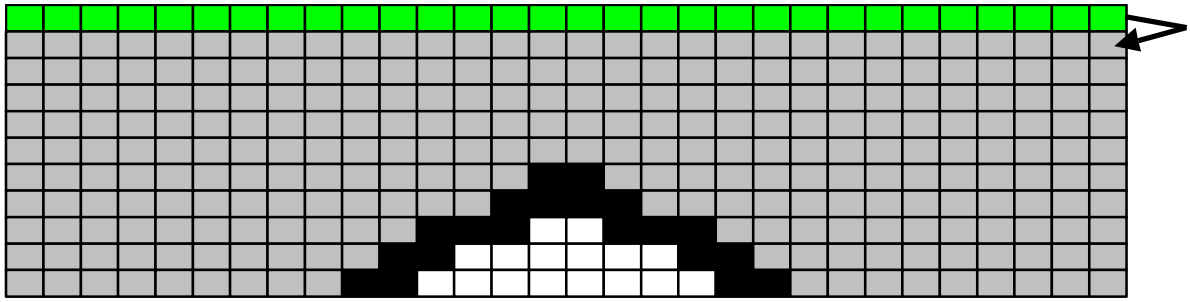


Figure 6.7(b) *- Discontinuity removal 1st pass - comparison of phase of top line with next line below*

$$\phi_{x+1} = \phi_{x+1} - 2\pi\left(\phi_{x-1} - \phi_x \middle| \mathrm{MOD}\, 2\pi\right) \tag{6.2}$$

This unwrapped line is the used as a reference for the rest of the data on the platen. The unwrapping algorithm is thus dependent on this line being a section through smoothly-varying phase values corresponding to the surface of the platen. More generalised algorithms exist which can cope with discontinuous phase [**1**,**2**,**3**] though they are generally more computationally intensive.

The phase values of the next line below are compared to those of the unwrapped line, and adjustments made as necessary. This procedure continues down the image. Each pixel is checked before being unwrapped to verify that it corresponds to valid data from the platen. This first pass halts at the centre of the image.

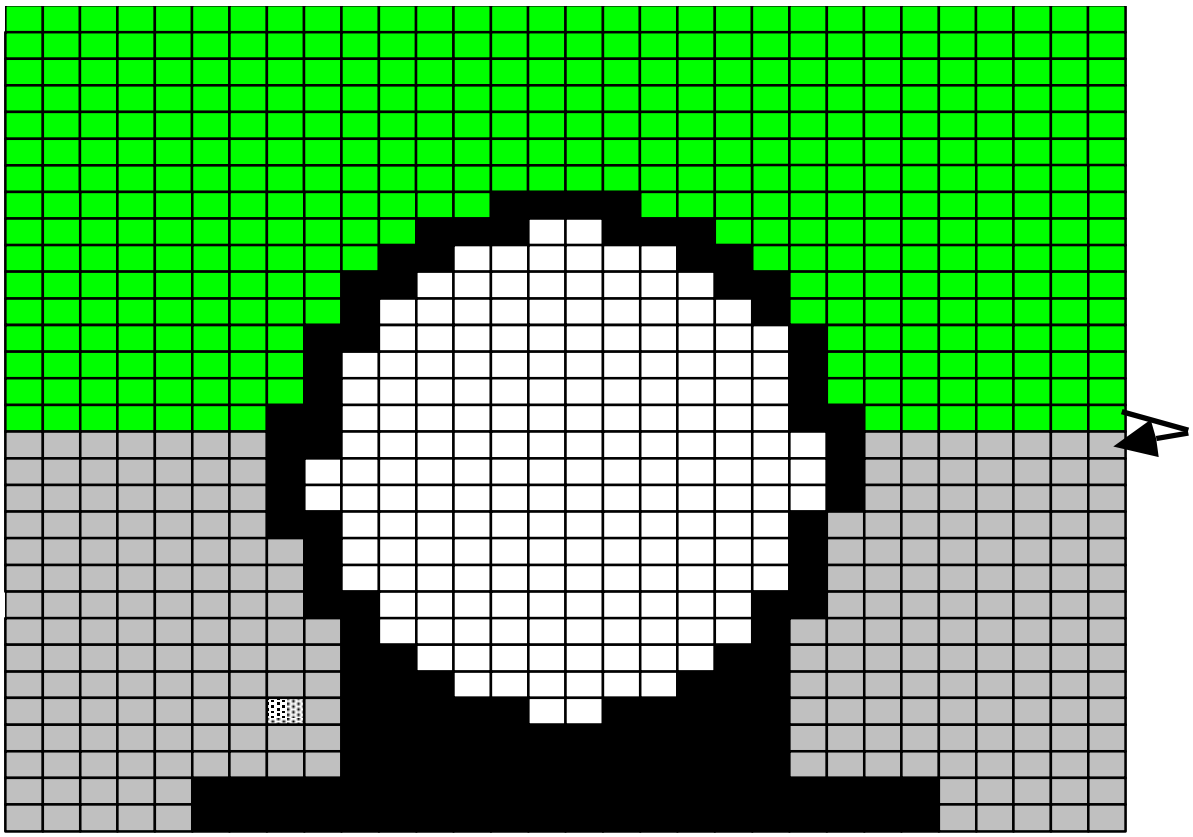


Figure 6.7(c) *- Discontinuity removal 1st pass - continuation until middle of image*

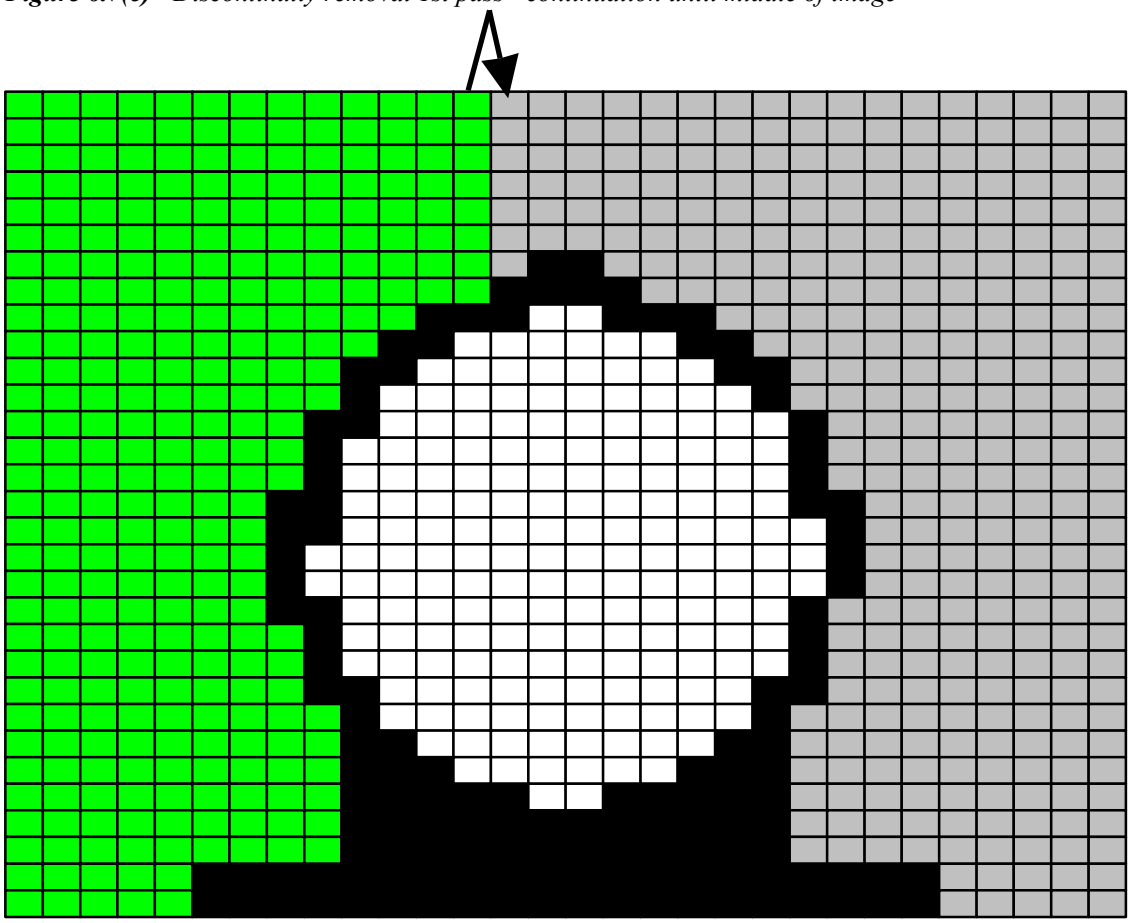


Figure 6.8 *- Discontinuity removal 2nd pass from left to right*

The second pass starts by unwrapping the vertical line along the left edge of the image. This line is then used as a reference for subsequent lines using the same procedure as the first pass. The second pass stops at the centre of the image.

The third pass is the same as the second, though starting from the right edge of the image. The three pass algorithm thus fits together the phase data around the edge of the image, then moves inwards towards the bar.

The phase data of the bar are unwrapped using a similar two pass algorithm, starting at the centre of the bar and then moving upwards and downwards.

After discontinuity removal the three phase maps are smoothly-varying but contain tilt due to the presence of tilt fringes in the original images, as shown in figure 6.9.

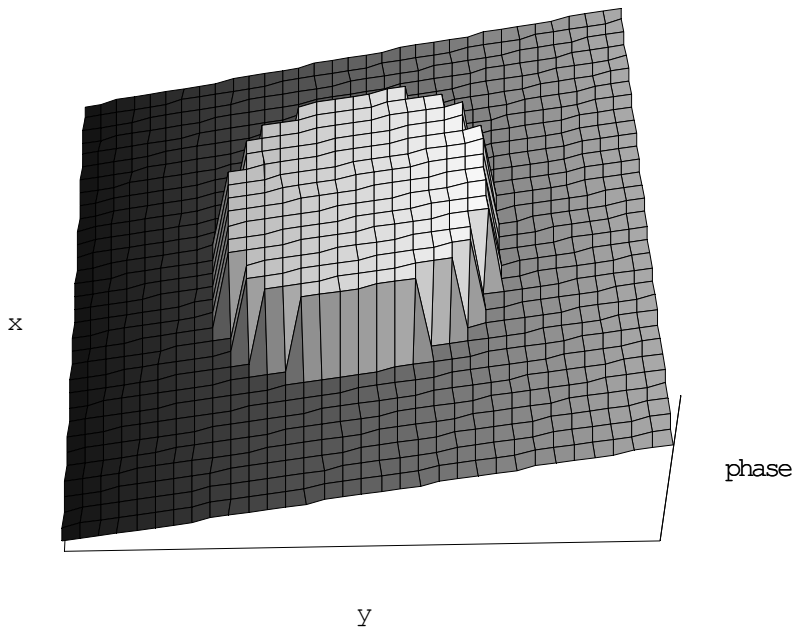


Figure 6.9 *- Simulated phase map after discontinuity removal, showing tilt and phase difference due to bar*

The tilt in each phase map is removed by fitting a suitable polynomial surface to the phase data of the platen and then subtracting this fitted surface from the measured data. The results of the image processing are three phase maps, at the three measurement wavelengths corresponding to the difference between the measured phase and the fitted surface, at each point in the image. Thus the phase maps represent (1) the deviation of the platen from the fitted surface and (2) the phase of the end of the bar with respect to

the fitted surface, *i.e.* the length of the bar, as defined in BS 5317. Typical phase maps after discontinuity and tilt removal are shown in figure 6.10.

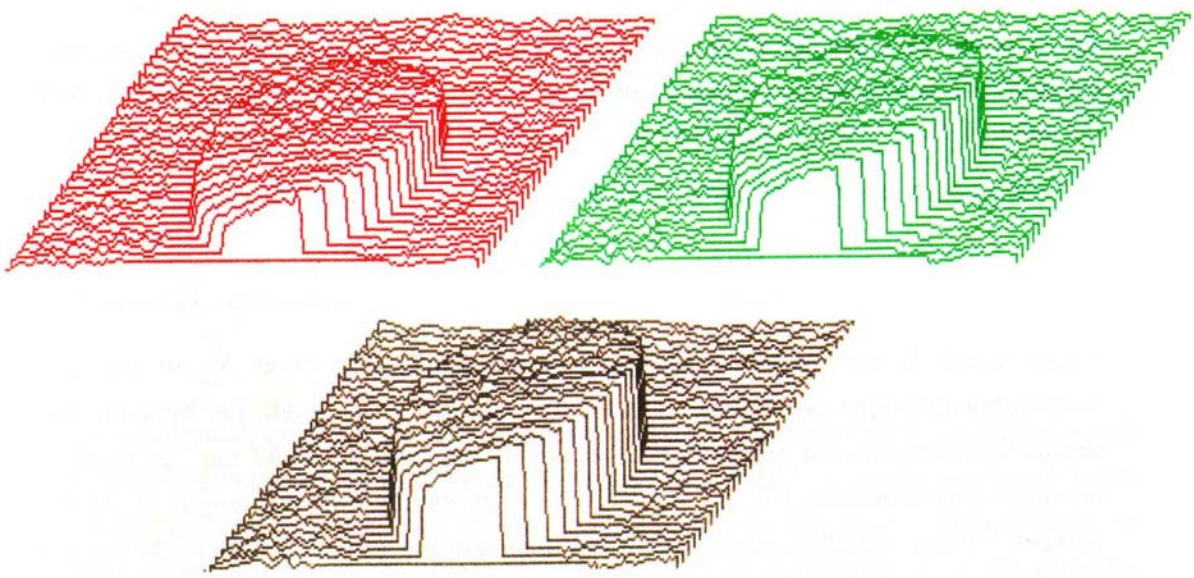


Figure 6.10 *- Phase maps after discontinuity and tilt removal, clockwise from top left: 633 nm, 543 nm, 612 nm*

The values in the three phase maps are converted to fringe fractions by dividing by 2π. Statistics of the surface measurements such as peak-valley variation, flatness and parallelism of the faces are easily calculated from the phase maps. In practice, these parameters are calculated from only the 633 nm phase map, `phi_red`.

For the calculation of the length of the bar, the fringe fractions of 9x9 pixels at the centre of the bar are averaged for each of the three wavelengths. These three fractions, f_1, f_2 and f_3 corresponding to the red, green and orange wavelengths respectively, can be combined in the technique of multiple-wavelength interferometry to calculate the length of the bar, based on solution of equation (6.3).

$$L = (n + f)\lambda' / 2 \tag{6.3}$$

6.4 MULTIPLE-WAVELENGTH INTERFEROMETRY

6.4.1 Multiple wavelength analysis

The reason for using three wavelengths rather than just one will now be explained. With one wavelength the corresponding fringe fraction, *f*, can be measured by the

interferometer. In order to solve equation 6.3, it is then necessary to know the value of n, *i.e.* it requires prior knowledge of L to within ± 1/4 of a fringe or approximately 150 nm. This accuracy cannot be achieved with conventional techniques.

To overcome this, use can be made of a second wavelength using the method of exact fractions [**4**]. If the effective range S_1 of the single wavelength technique is $\lambda_1 / 2$, then the range of the two wavelength system, $S_{1,2}$, is given by

$$S_{1,2} = \frac{\lambda_1 \lambda_2}{2(\lambda_1 - \lambda_2)} \tag{6.4}$$

where λ_1 and λ_2 are the two wavelengths used [**5**,**6**,**7**]. The range $S_{1,2}$ of the two-wavelength technique can thus be increased by making $\lambda_1 - \lambda_2$ small, *i.e.* by using two similar wavelengths. For example, with $\lambda_1 = 633$ nm and $\lambda_2 = 543$ nm, as used in previous interferometers [**8**], $S_{1,2} = 1.9$ µm, *i.e.* an estimate of the length of the bar within ± 0.9 µm will allow unambiguous calculation of the accurate length of the bar. For long bars in particular, such as those over 1 m in length, this accuracy is difficult to achieve, especially without accurate knowledge of the thermal expansion coefficient of the length bar (see Chapter 8). It is possible to increase the range $S_{1,2}$ by using other wavelengths, *e.g.* $\lambda_1 = 633$ nm and $\lambda_2 = 612$ nm, for which $S_{1,2} = 9.2$ µm, however the effective range is actually smaller than $S_{1,2}$ because it is limited by the accuracy of the measurement of the fringe fractions f_1 and f_2. This can be overcome by using a third wavelength, λ_3. To see why a third wavelength is necessary, the system of solutions to equation (6.3) for two wavelengths will be examined.

Rewriting equation (6.3) for the two ambient wavelengths λ_1' and λ_2' gives

$$L_1 = (n_1 + f_1)\lambda_1' / 2 \tag{6.5a}$$
$$L_2 = (n_2 + f_2)\lambda_2' / 2 \tag{6.5b}$$

Values of f_1 and f_2 are measured in the interferometer. With no *a priori* knowledge of L, values of n_1 and n_2 are undetermined and solutions of (6.5a) and (6.5b) are periodic in $\lambda_1' / 2$ & $\lambda_2' / 2$, respectively (see figure 6.11). For certain values of n_1 and n_2 the solutions of (6.5a) and (6.5b) are equal to within a small margin of error. In figure 6.11 this occurs for ($n_1 = 1$, $n_2 = 1$), ($n_1 = 6$, $n_2 = 7$), ($n_1 = 12$, $n_2 = 14$) and ($n_1 = 17$, $n_2 = 20$). Only one of these solutions corresponds to the length of the bar. The correct solution is deemed to be the one for which the two individual solutions agree most closely, in this case either ($n_1 = 1$, $n_2 = 1$) or ($n_1 = 17$, $n_2 = 20$) would be selected, *i.e.* the effective range of the technique is limited to 17 orders of $\lambda_1' / 2$ (in this example), unless these two close matches can be resolved.

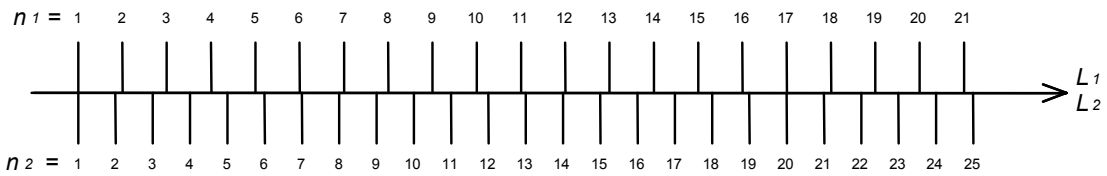


Figure 6.11 *- Coincidences for two wavelengths 633 nm & 543 nm*

To distinguish between the close matches requires a measurement resolution which depends on the accuracy with which the values of $(n+f)\lambda'/2$ can be measured. In the interferometer, the fundamental limit on the accuracy of measurement is the knowledge of the ambient wavelength λ' caused by uncertainty in the determination of the refractive index of the ambient air at different wavelengths (see Chapter 7). The uncertainty in this dispersion correction is approximately ± 2.48 x 10^{-8}. This is equal to a measurement uncertainty of ± 25 nm for a 1000 mm bar, or 0.08 fringes at wavelength 633 nm. Thus any solutions which agree to closer than 0.08 $\lambda_1'/2$ will not be resolved. In the case of λ_1 = 633 nm and λ_2 = 612 nm, the corresponding solutions occur for (n_1 = 1, n_2 = 1) and (n_1 = 2, n_2 = 2) because the two wavelengths are so similar. Thus using these 2 wavelengths the technique is limited to only 1 order of $\lambda_1'/2$, or 306 nm which is no better than using single wavelength interferometry.

To distinguish between these solutions, a third 543 nm wavelength is used. This leads to a set of 3 equations of the form of (6.5a) and (6.5b) with three periodic sets of solutions. The correct solution is identified by close matches at all three wavelengths with the coincidence at n_1 = 17 resolved, as shown in figure 6.12. In the interferometer, with λ_1 = 633 nm, λ_2 = 543 nm and λ_3 = 612 nm, the effective range is extended to 30 orders of $\lambda_1'/2$, or approximately 9.5 µm, which is where the next close match between all three solutions occurs. It is relatively easy to measure the length of a bar to within ± 9 µm by comparison with other measured bars *e.g.* using a CMM. Also, the length tolerances for length bars according to BS 5317 mean that any bar should be well within ± 9 µm of its nominal length, *e.g.* 1000 mm ± 1.55 µm is a typical tolerance.

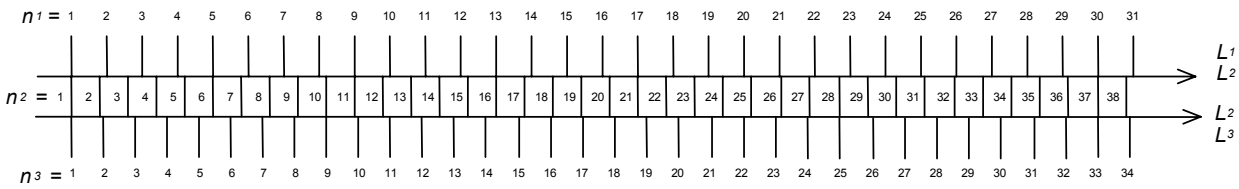


Figure 6.12 *- Coincidences for three wavelengths 633 nm, 543 nm, 612 nm*

Thus to find the length of the bar, based on a nominal value and three measured fringe fractions, the system of coincidences between the three wavelengths is examined for the closest match over a range of ± 15 orders of the red wavelength, centred on the order corresponding to the nominal size of the bar input by the user. This produces one solution which corresponds to the length of the bar.

This system of matching solutions for functions of different periods is similar to the integer analysis proposed by Guzhov & Solodkin [**9**], however the technique of integer decoding of interferograms is not valid in this particular situation as the uncertainty associated with the measurement of the fringe fractions precludes the use of integers to describe them, *i.e.* the values are real numbers with associated uncertainties.

The accuracy of the matching of the 3 solutions is measured by examining the 'residuals' - these are calculated as follows.

Let L_r, L_g, L_o be the lengths of the bar calculated using the 3 measured fringe fractions (red, green and orange respectively). The length of the bar can be calculated as the mean of these 3 results:

$$L = \frac{L_r, L_g, L_o}{3} \tag{6.6}$$

The red, green and orange residuals r_r, r_g, r_o are calculated as follows

$$r_r = \frac{2(L - L_r)}{\lambda'_r} \qquad r_g = \frac{2(L - L_g)}{\lambda'_g} \qquad r_o = \frac{2(L - L_o)}{\lambda'_o} \tag{6.7}$$

i.e. they represent the departure from perfect agreement, in units of one fringe (at each wavelength). Typical values of the residuals are dependent on the length being measured (showing the uncertainty in measurement to be length dependent and hence

due to factors such as alignment, refractive index, *etc*): for a short bar (up to 300 mm) they are typically less than about ± 0.01 fringe (± 3 nm) and for long bars (approximately 1000 mm), they are ± 0.03 fringe (± 9 nm). Note that some of the residuals will be positive, some negative. When two of the residuals have one sign for all measurements, and the third has the opposite sign, then that indicates that the single laser has drifted out of calibration more than the mean of the two other lasers. If the values of all the residuals increases with time, but with randomly varying signs, then this indicates that the equipment used for refractive index determination or temperature measurement is due for recalibrating (*e.g.* Druck, Michell, Tinsley, PRTs). In the interferometer, the frequency stability of the red laser is better than the green and orange lasers, so the measured length is the red measured length, rather than the average of the three measured lengths.

6.4.2 Limit to multiple-wavelength technique due to source instability

Walsh [**10**] has proposed a technique for calculating the limit to the length which can be measured using multiple-wavelength interferometry due to frequency instability of the light source(s). This will be examined for the case of the interferometer presented here.

Firstly, the equations for the length measurement are re-written as follows:

The equation

$$2L = \lambda_i (N_i + f_i) \tag{6.8}$$

is written as

$$2L = \lambda_s (N_s + f_s) \tag{6.9}$$

where

$$\lambda_s = \frac{1}{\sum A_i \sigma_i} \qquad \sigma_i = \frac{1}{\lambda_i} \qquad N_s = \sum A_i N_i \qquad f_s = \sum A_i f_i \tag{6.10}$$

the f_i are fringe fractions, the N_i are the integer interference orders and the A_i are coefficients, chosen such that

$$\lambda_s >> \lambda_i \quad \text{and} \quad \sum A_i = 0 \tag{6.11}$$

(Note that the effective wavelength is given by λ_s, which for the two-wavelength case, is the same as that given by (6.4)). The length of the object can then be calculated from

$$2L = \frac{1}{\sum A_i \sigma_i}\left(\sum A_i N_i + \sum A_i f_i\right) \tag{6.12}$$

provided that the length of the object is already known to within ± $\lambda_s/2$.

Walsh shows that the maximum length that can be measured using this technique is given by

$$L \leq \frac{1}{2}\frac{c}{\sum |A_i| \Delta\upsilon} \tag{6.13}$$

where $\Delta\nu$ is the uncertainty in the frequency of the light source. For the interferometer, the largest value of $\Delta\nu$ is approximately 5 MHz (for the green laser). The choice of values for the A_i coefficients depends on the operational mode. For two-wavelength interferometry, A_1 = 1 and A_2 = -1, leading to a value for $\sum |A_i|$ of 2. The maximum length measurable with this technique using the green and red wavelengths is 15 m, provided that the length is already known to within ± 1.9 μm (as before). For three-wavelength interferometry, A_1 = -1, A_2 = 2, A_3 = -1, giving $\sum |A_i|$ = 4. The maximum length measurable with the three-wavelength technique using the three wavelengths of the interferometer is 7.5 m, provided the length is already known to within ± 3.2 μm.

Using the above equations for the calculation of the length of the object shows that the frequency instability of the green laser places a tighter tolerance on the initial length estimate, unless the effect of the laser drift can be decreased, for example by more frequent calibration of the lasers. The interferometer does not use equation (6.12) to calculate the length of the bar directly because the N_i are not directly known: the method of excess fractions is used. These two techniques are fundamentally the same and the limit of 7.5 m does indeed apply to the interferometer and represents the maximum length which can be measured using this technique with these laser wavelengths.

6.4.3 Multiple wavelength algorithm - method of exact fractions

The procedure from the computer program which calculates the result is presented here. Comments on the code are in bold type.

```
procedure calc_length;

{procedure to calculate multiple-wavelength solution to measured length}
{works with two or three wavelengths}

var
    alter,range                 :integer;
    i,x,y,best_solution,again   :integer;
    amb_order_red               :integer4;
    amb_order_green             :integer4;
    amb_order_orange            :integer4;
    amb_red_est                 :real8;
    amb_green_est               :real8;
    amb_orange_est              :real8;
    nom_red_est                 :real8;
    nom_green_est               :real8;
    nom_orange_est              :real8;
    red_residual                :real8;
    green_residual              :real8;
    orange_residual             :real8;
    nom_mean_length,minimum     :real8;
    wavelength                  :real8;
    amb_length                  :real8;
    amb_est                     :real8;
    residuals                   :array[1..3,-20..20]of real8;
    key                         :byte;

begin

{resets length and wavelength variables}

precis_length:=0.0;
red_wave_corr:=9.99;
green_wave_corr:=9.99;
orange_wave_corr:=9.99;

{calculates wavelengths of operational lasers to ambient conditions}

if (red_used) then
red_wave_corr:=wave_correction(red_wavelength,air_temp,air_pressure,air_hum
idity);
if (green_used) then
green_wave_corr:=wave_correction(green_wavelength,air_temp,air_pressure,air
_humidity);
if (orange_used) then
orange_wave_corr:=wave_correction(orange_wavelength,air_temp,air_pressure,a
ir_humidity);

{converts nominal length to ambient conditions}
{estimates nominal red interference order at ambient conditions}

amb_length:=nom_to_amb(nom_length,bar_temp);
amb_order_red:=round4(amb_length*2/red_wave_corr);

{resets fringe fraction totals for before averaging 9x9 points at image
centre}

red_fraction:=0;
green_fraction:=0;
orange_fraction:=0;
for x:=cent_x-4 to cent_x+4 do
  begin
  for y:=cent_y-4 to cent_y+4 do
      begin
      if (red_used) then red_fraction:=red_fraction+phi_red[x,y];
      if (green_used) then green_fraction:=green_fraction+phi_green[x,y];
      if (orange_used) then
orange_fraction:=orange_fraction+phi_orange[x,y];
  end;
end;
red_fraction:=red_fraction/81;
green_fraction:=green_fraction/81;
orange_fraction:=orange_fraction/81;

{displays some headings and other information on the screen}
{chooses order scanning range: 20 for normal, 50 if debugging}

if (debugging) then rewrite(debug_file);
if (debugging)
    then range:=50
    else range:=20;
```

```
{scans over range of orders, around ambient estimated order}
{calculates solutions to interference equation at each order}
{and converts back to nominal conditions}

for i:=-range to range do
   begin
   amb_red_est:=(i+amb_order_red+red_fraction)*red_wave_corr/2;
   nom_red_est:=amb_to_nom(amb_red_est,bar_temp);

{if green wavelength operating, works out corresponding green order at}
{ambient, and then solves interference equation for green data}
{then converts to nominal conditions}

   if (green_used) then
     begin
     amb_order_green:=round4(2*amb_red_est/green_wave_corr-green_fraction);
     amb_green_est:=(amb_order_green+green_fraction)*green_wave_corr/2;
     nom_green_est:=amb_to_nom(amb_green_est,bar_temp);
   end;

{repeats this process if orange wavelength operating}

   if (orange_used) then
      begin
      amb_order_orange:=round4(2*amb_red_est/orange_wave_corr-
orange_fraction);

amb_orange_est:=(amb_order_orange+orange_fraction)*orange_wave_corr/2;
      nom_orange_est:=amb_to_nom(amb_orange_est,bar_temp);
   end;

{declares red answer correct, calculates orange and green residuals}
{which are departures from the red answer, in units of fringes}

   nom_mean_length:=nom_red_est;
   amb_length:=nom_to_amb(nom_mean_length,bar_temp);

   if (green_used) then
       begin
       amb_est:=nom_to_amb(nom_green_est,bar_temp);
       green_residual:=(amb_est-amb_length)/green_wave_corr;
   end;

   if (orange_used) then
       begin
       amb_est:=nom_to_amb(nom_orange_est,bar_temp);
       orange_residual:=(amb_est-amb_length)/orange_wave_corr;
   end;

{resets residuals to zero if wavelength not used}

   if not(green_used) then green_residual:=0.0;
   if not(orange_used) then orange_residual:=0.0;
   red_residual:=0.0;

{writes debugging information to file, if requested}

   if (debugging) then
      begin
     write(debug_file,i+amb_order_red,chr(9),green_residual:5:3,chr(9),
      orange_residual:5:3,chr(9));
      writeln(debug_file,abs(green_residual) + abs(orange_residual));
   end;

{for central +- 20 orders, stores resiuals in an array}

   if (abs(i) <=20) then
      begin
      residuals[1,i]:=red_residual;
      residuals[2,i]:=green_residual;
      residuals[3,i]:=orange_residual;
   end;
end;

{scans through array, picking lowest absolute residuals}
{marks this as best solution}

minimum:=99999.9;

for i:=-20 to 20 do
   begin
   if ((abs(residuals[1,i]) + abs(residuals[2,i]) + abs(residuals[3,i])) <
minimum)
       then
```

```
       begin
       minimum:=abs(residuals[1,i]) + abs(residuals[2,i]) +
abs(residuals[3,i]);
       best_solution:=i;
   end;
end;

{for 3 orders either side of best solution, displays results}

for i:=best_solution-3 to best_solution+3 do
   begin
   amb_red_est:=(i+amb_order_red+red_fraction)*red_wave_corr/2;
   nom_red_est:=amb_to_nom(amb_red_est,bar_temp);

   if (green_used) then
      begin
      amb_order_green:=round4(2*amb_red_est/green_wave_corr-
green_fraction);
      amb_green_est:=(amb_order_green+green_fraction)*green_wave_corr/2;
      nom_green_est:=amb_to_nom(amb_green_est,bar_temp);
    end;

   if (orange_used) then
      begin
      amb_order_orange:=round4(2*amb_red_est/orange_wave_corr-
orange_fraction);

amb_orange_est:=(amb_order_orange+orange_fraction)*orange_wave_corr/2;
      nom_orange_est:=amb_to_nom(amb_orange_est,bar_temp);
    end;

   nom_mean_length:=nom_red_est;
   amb_length:=nom_to_amb(nom_mean_length,bar_temp);

   if (green_used) then
       begin
       amb_est:=nom_to_amb(nom_green_est,bar_temp);
       green_residual:=(amb_est-amb_length)/green_wave_corr;
    end;

   if (orange_used) then
       begin
       amb_est:=nom_to_amb(nom_orange_est,bar_temp);
       orange_residual:=(amb_est-amb_length)/orange_wave_corr;
   end;

   if not(green_used) then green_residual:=0;
   if not(orange_used) then orange_residual:=0;
   red_residual:=0.0;

   residuals[1,i]:=red_residual;
   residuals[2,i]:=green_residual;
   residuals[3,i]:=orange_residual;

{prints information, and flags best solution}


write((i+amb_order_red):8,red_residual:10:3,green_residual:10:3,orange_resi
dual:10:3,
'     ',amb_length*1000:11:6);
   if (i = best_solution)
       then
       begin
       write('  <--- BEST SOLUTION');
       precis_length:=nom_mean_length;
       uncorrected_length:=nom_to_amb(precis_length,bar_temp);
       best_red_residual:=residuals[1,i];
       best_green_residual:=residuals[2,i];
       best_orange_residual:=residuals[3,i];
   end;
   writeln;
end;
end;
```

6.4.4 Order scanning

Although the expected repeat distance for the three-wavelength technique is 30 orders (± 15 either side of nominal), it is often possible to extend this range, with care. At the coincidence at n_1 = 30, the difference between the red and green coincidences is -0.061 green fringes and between the red and orange coincidences is 0.031 orange fringes.

$$
\begin{aligned}
&30.000 \times \lambda_1 \\
&= 31.031 \times \lambda_2 \\
&= 34.939 \times \lambda_3
\end{aligned}
$$

It is shown in chapter 7 and chapter 10 that the uncertainty of the refractive index correction is ± 2.5 x 10^{-8} at a confidence level of 95%. Thus it is 95% certain that the errors in absolute fringe fraction measurements will be within ± 0.08 λ_1, ± 0.09 λ_2, ± 0.08 λ_3 for bars up to 1 m in length. For shorter bars the errors will be expected to be correspondingly smaller and so it is possible to extend the order scanning. It has been found that ± 20 orders is a suitable scanning range though the software allows this to be extended to ± 50 orders for debugging purposes.

6.5 FLATNESS AND PARALLELISM MEASUREMENTS

As well as measuring the central length of the bar, the interferometer also measures the flatness of the exposed face of the bar and the parallelism of the faces, measured as the variation in length of the bar measured at different points on the exposed face. Because these are only relative measurements (from one point to another) and do not require absolute determination, the values of flatness and parallelism are calculated from only the red wavelength phase map.

6.5.1 Measurement of parallelism (variation)

The parallelism (strictly the deviation from parallelism) is defined as the difference between the maximum and minimum lengths, measured at any points on the measurement faces, measured perpendicular to the wrung face (see § 1.4.5).

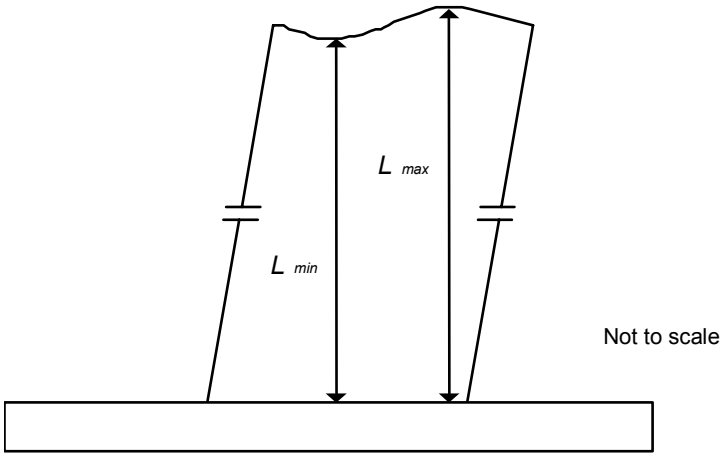


Figure 6.13 *- Measurement of deviation from parallelism of a length bar's measurement faces*

Note that the definitions of L_{max} and L_{min} are the same as for the central length, *i.e.* the distance, perpendicular to the wrung surface, to the measurement point. This is why the interferometer is aligned with the platen face perpendicular to the measurement beam, rather than the exposed face of the bar (see § 4.1.8). The phase measurements are also referred to the best fit surface through the platen data, which should correspond closely with the mean tangent to the platen surface at any point, if the platen surface is smoothly varying.

The parallelism is thus calculated as the difference between the maximum and minimum values in the phase data measured on the surface of the bar (the software mask is used to check that the data corresponds to the surface of the bar).

```
{having fitted least squares plane to data on the bar}
{now finds variation (parallelism)}

min:=9999.9;
max:=-9999.9;
for x:=0 to 126 do
    begin
    for y:=0 to 126 do
        begin

{checks to see if data corresponds to bar surface}

        if (imagestore(11,x,y)=254) then
            begin

{data is on bar, takes max and min values}

            if (phi_red[x,y] < min) then min:=phi_red[x,y];
            if (phi_red[x,y] > max) then max:=phi_red[x,y];
        end;
    end;
end;

variation:=max-min;
```

When measuring gauge blocks, this result is termed the *variation* in length, since the parallelism is defined differently according to which standard is being used. Thus in the results, the parallelism is also referred to as variation.

6.5.2 Measurement of flatness

The (deviation from) flatness is defined as the minimum distance between two parallel planes which just envelop the measuring face.

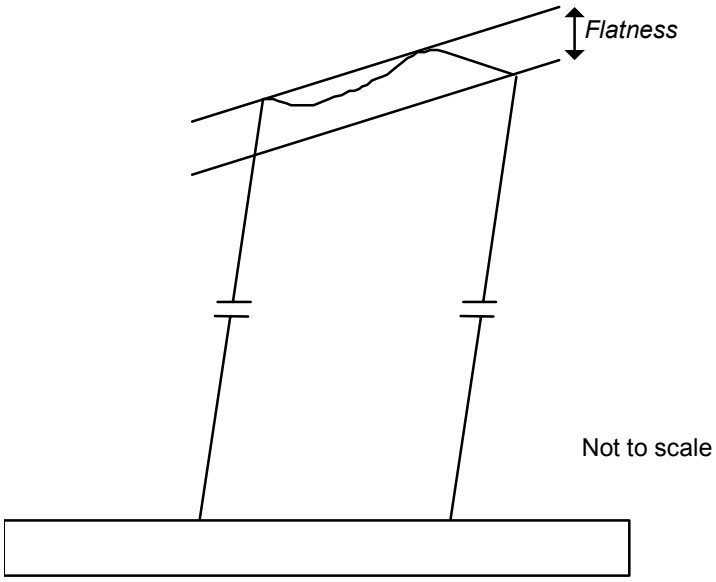


Figure 6.14 *- Measurement of deviation from flatness of length bar's measurement face*

This definition is difficult to realise in practice due to the difficulty of fitting two parallel planes to the data. Except for pathological cases, this definition of flatness is equivalent to that used in the interferometer, which is the difference between the maximum positive and negative departures from the best fit plane through the surface of the bar.

To measure the flatness, the program performs a least squares fit of a plane to the data on the bar. The difference between the maximum positive and negative departures from this plane is the flatness.

```
{fits least squares plane to data on bar}
{now finds flatness}

min:=9999.9;
max:=-9999.9;
for x:=0 to 126 do
    begin
    for y:=0 to 126 do
        begin

{checks to see if data corresponds to bar surface}

        if (imagestore(11,x,y)=254) then
            begin

{data is on bar, now take difference between fitted and real data}
{max-min = flatness}

            if ((phi_red[x,y]-(b0+(b1*x)+(b2*y))) < min) then
min:=(phi_red[x,y]-(b0+(b1*x)+(b2*y)));
            if ((phi_red[x,y]-(b0+(b1*x)+(b2*y))) > max) then
max:=(phi_red[x,y]-(b0+(b1*x)+(b2*y)));
        end;
    end;
end;
flatness:=max-min;
```

6.5.3 Example measurements

The following are some example measurements of flatness and parallelism performed on length bars in the interferometer. For each bar, the red phase map is displayed, with numerical data for the flatness and parallelism (variation).

These measurements were performed with bars from the same set (Set 1455). In these printouts the end of the bar appears slightly oval, this is due to the camera and framestore pixels corresponding to rectangles in the image, rather than squares. A correction factor is used in the display software to try to correct the images.

The area around the bar which is masked and is set to zero phase can be seen in the phase maps as the flat area immediately around the edge of the bar. The smoothness of the platen surface can be judged from the phase data in the remainder of the image.

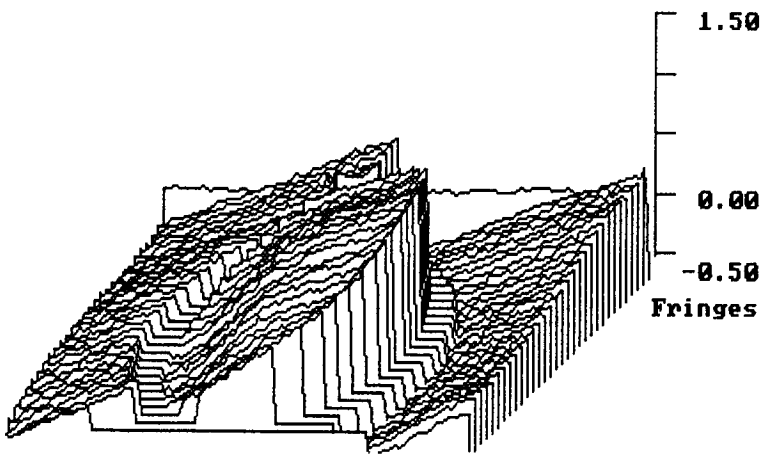
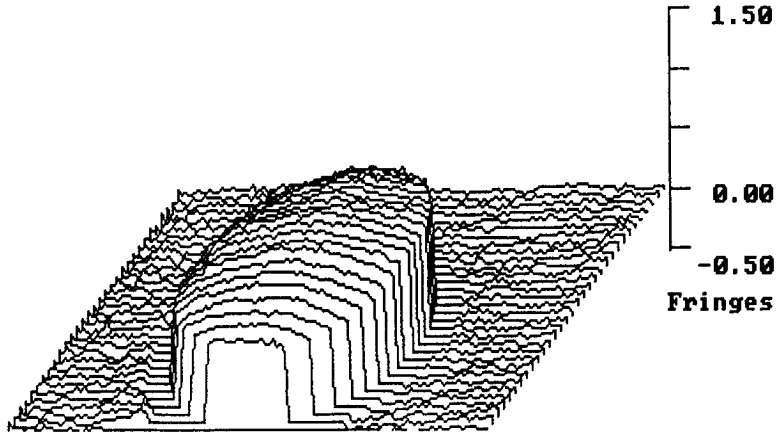
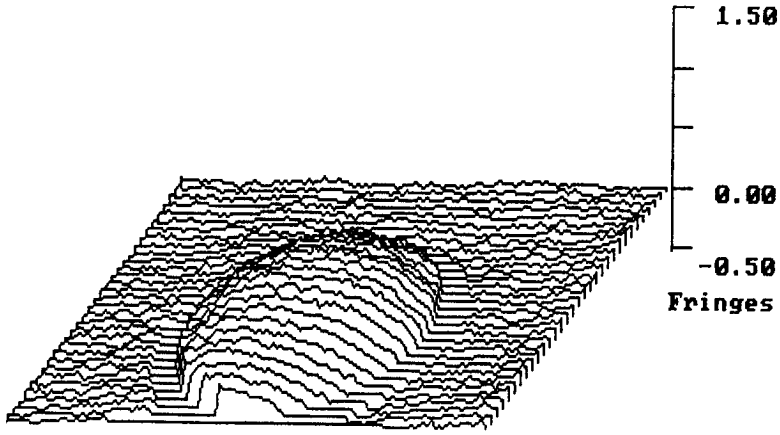


Figure 6.15 *- Example measurement showing errors in the phase map due to improper discontinuity removal across top line, caused by incorrect positioning of platen in image*

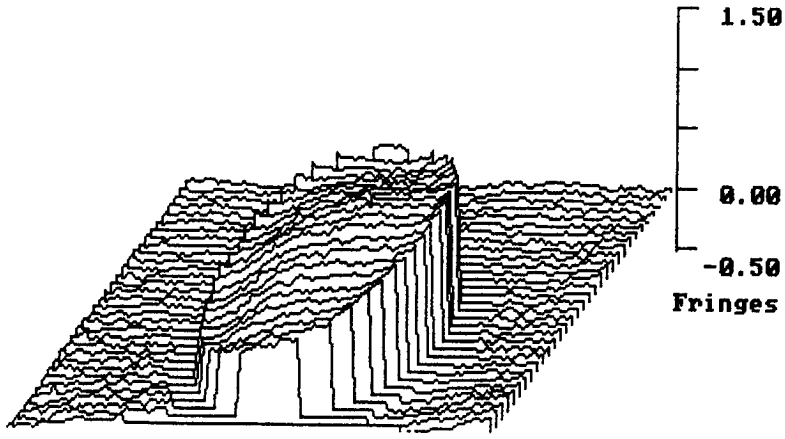
Flatness: 0.43 fringes
Variation: 0.48 fringes



Flatness: 0.55 fringes
Variation: 0.61 fringes



Flatness: 0.33 fringes
Variation: 0.74 fringes



Flatness: 0.42 fringes
Variation: 1.11 fringes

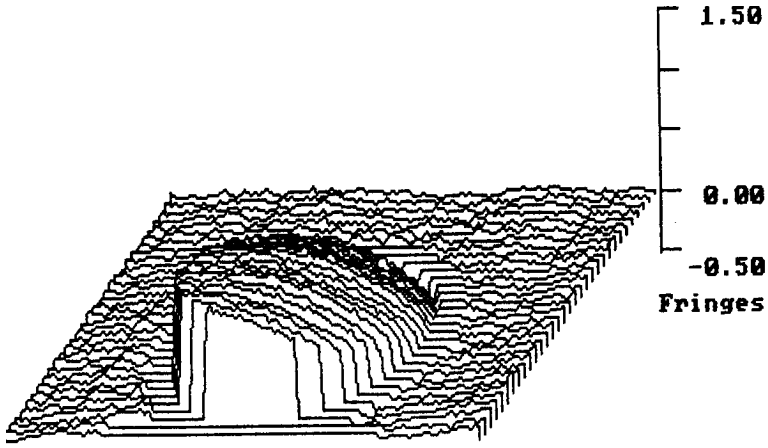


Figure 6.16 *- Example measurements of flatness and parallelism (variation) of length bars*

6.6 COMPUTER PROGRAM

The computer program is approximately 4500 lines long so only an overview of the routines is given here for reference. Figure 6.17 flowcharts the program which controls the interferometer.

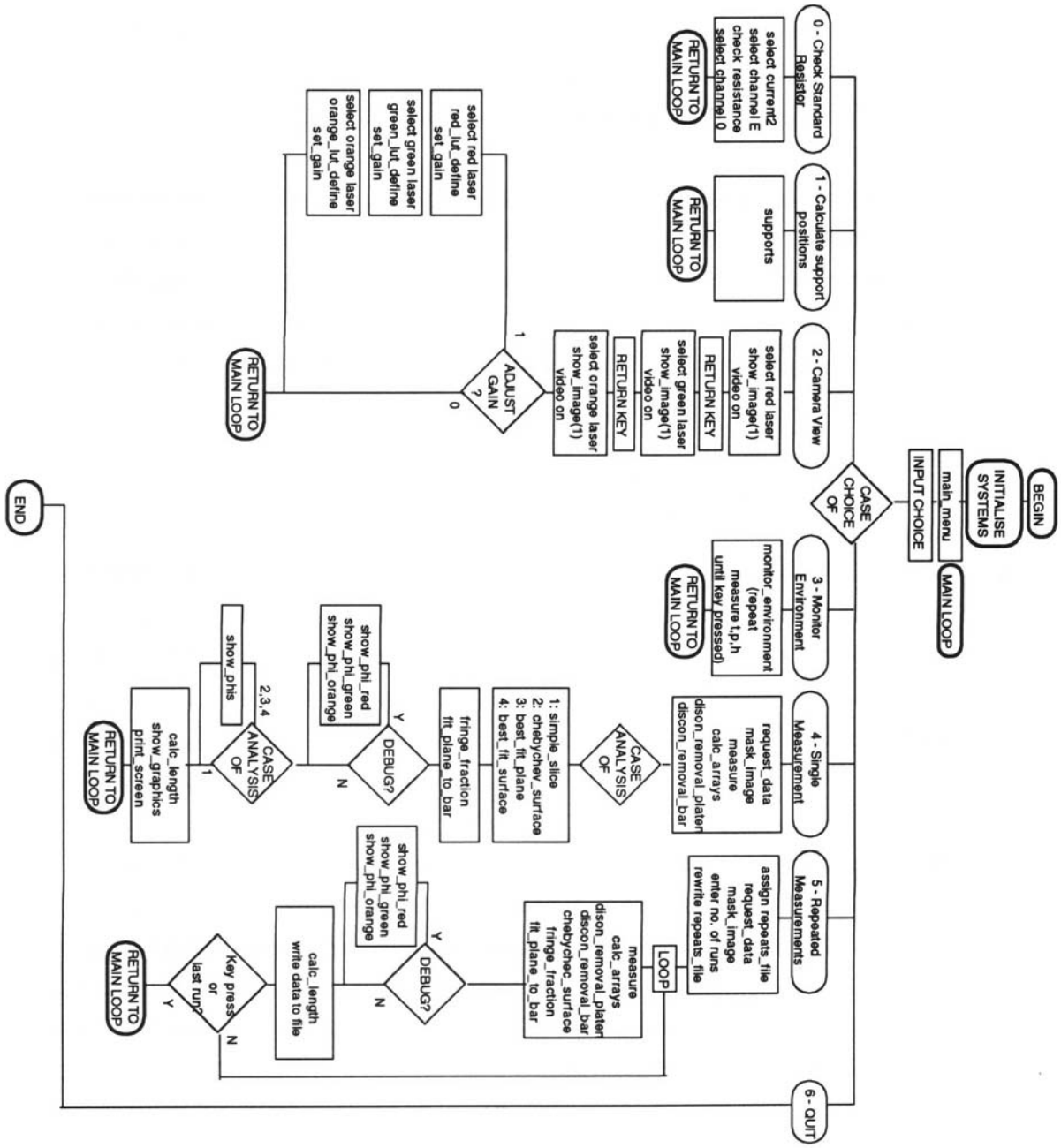


Figure 6.17 *- Program flowchart (simplified)*

After initialisation, the program is based around a main loop which allows the user six choices, selected from a menu. These choices are as follows.

0 - Check standard resistor

This option is used periodically to calibrate the Tinsley resistance bridge. It selects channel E, which has the 100 Ω resistor connected. The display of the bridge then shows the value of the resistor. The front panel potentiometer is then adjusted until the resistance reading is the same as the calibrated value for the resistor.

1 - Calculate support positions

This option is used to calculate the position of the supports for the length bars. An input of the bar length is required. The calculation assumes a platen thickness and diameter corresponding to the six specially manufactured interferometer platens. The calculation uses the analysis developed in appendix C. The results given are the positions of the supports from either end of the bar and the support separation.

2 - Camera view

This option is used either when aligning the interferometer (to switch on the laser beam) or when adjusting the tilt of the mirrors. The lasers are selected in sequence: red, green, orange. If necessary, there is a further option which allows the user to adjust the offset and gain of the digitiser to ensure that the full range of the A-to-D converter is used, but without saturation. A histogram is shown on screen similar to figures 6.4 and 6.5.

3 - Monitor environment

This option continuously monitors conditions inside the chamber in a loop, until the user selects halt. The temperatures are measured in sequence with the pressure, humidity and CO_2 concentration. The sample pump is halted before the pressure reading is taken. The results can be saved to a file on disc.

4 - Single measurement

This option is the one normally used for making measurements of length bars. Firstly it requests data from the user for: Reference, Nominal size of bar, nominal thermal expansion coefficient, which of the three positions on the carriage the bar is occupying

(to select the correct PRTs), and the type of analysis to use. The program then allows the user to adjust the mask using a video overlay system which shows the image of the bar on screen with superimposed cursor lines which the user adjusts to enclose the edge of the bar in the image. Next the full measurement procedure begins with selection of the relevant PRTs and measurement of the humidity and CO_2 content. The sample pump is then stopped and the PRT resistance measured. The air PRT is then read. The phase-stepping is then performed, including synchronisation to the modulation of the red laser and a suitable wait period between successive steps to allow the DPT to reach correct position. In the middle of the phase-stepping, the pressure is read. The second of the PRTs in contact with the bar is measured and the sample pump re-started.

Next the program calculates the `phi` and `alpha` arrays (phase and phase step size, respectively) for the three wavelengths. If the debugging mode is active, the three phase step maps are displayed on the framestore monitor as grey level maps with histograms showing the spread in phase step. The discontinuities are removed from the platen and bar surfaces and the appropriate analysis is used to fit to the data on the platen. The fringe fractions are then calculated as the difference between the measured data and the fitted data, divided by 360°. A least-squares plane is fitted to the phase data of the bar surface to allow calculation of the flatness and parallelism. If the debugging mode is active, the three phase maps are displayed on the framestore monitor as grey level maps. The three phase maps are displayed on the monitor in pseudo-three dimensional form (*e.g.* figure 6.16). Next the multiple-wavelength analysis is used to calculate the length of the bar. The results are displayed with an option for printout.

5 - Repeated measurements

This option performs the same as option 4, but with no graphical displays and with results saved to disc file. The user can select the number of measurement runs, which can be interrupted if necessary at the end of any measurement.

6 - Quit

This option simply terminates the main loop, closes files and quits the program.

Figure 6.18 shows an example printout from the interferometer for a 1000 mm length bar measured at approximately 20 °C. The bar temperature is 20.008 °C, the air temperature is 20.012 °C. The bar is flat to within 0.46 fringes (145 nm) and exhibits a

variation in length (parallelism error) of 1.16 fringes (367 nm). The measured length of the bar is 1000.003 601 mm, hence it shows a departure of 3601 nm from its nominal length. The results show a slight rounding error between the two stated values for departure of 1 nm.

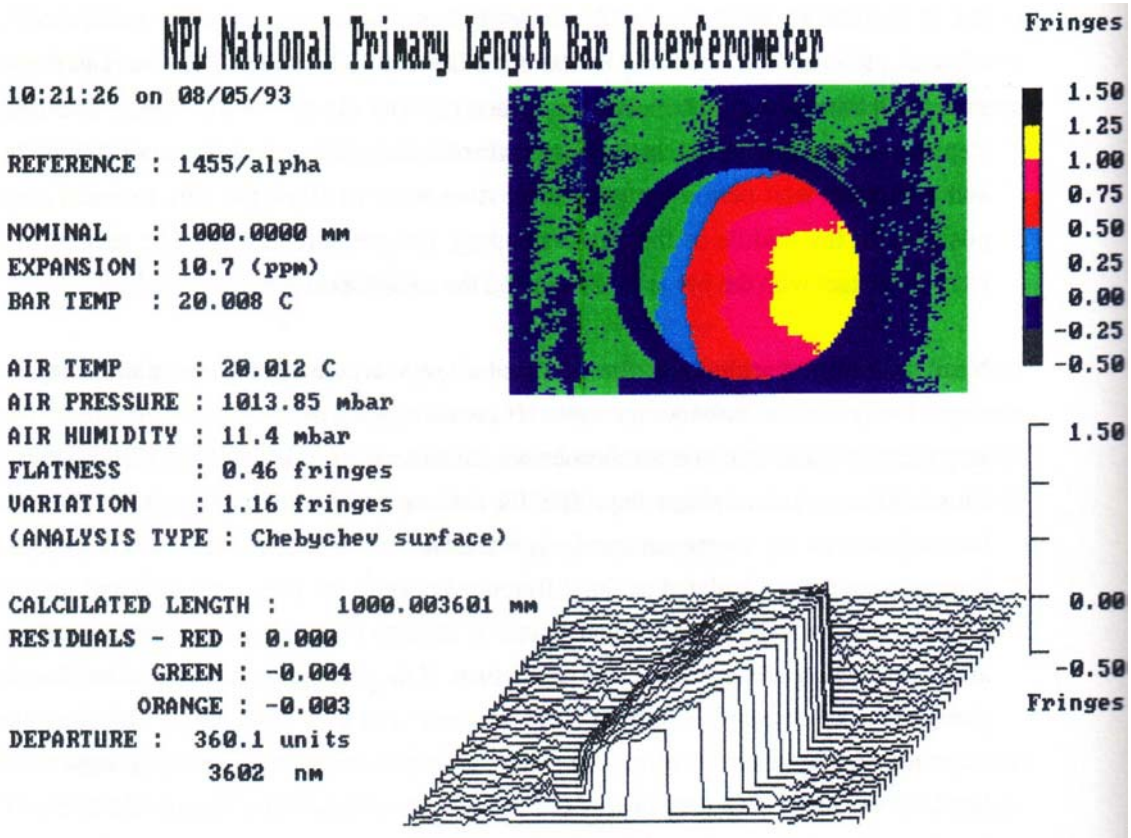


Figure 6.18 *- Example printout from interferometer: results for a 1000 mm bar*

6.7 DOUBLE-ENDED ANALYSIS

Initially the double-ended analysis was tried as a part of the normal interferometer program but this increased the code size beyond the limit imposed by the compiler - 64 K bytes of user generated code per compiled unit, module or program. Splitting the program into separate completed modules which could be combined at run time was tried, but cross-module global variable compatibility was difficult to achieve. The final solution was to produce a new program based on the original interferometer control program, but with non-essential procedures and functions removed to make space for the double-ended procedures. The new program has no options for environmental monitoring, disc file usage or single-ended measurements. It simply allows a camera view (for alignment purposes) and double-ended phase-stepping. New routines were

developed for: double-ended masking, double-ended discontinuity removal, double-ended data fitting and double-ended multiple-wavelength analysis. The main functionality of the program is the same as option 4 of the previous single-ended program, except for the following.

1 Masking now has to mask 2 bar images, a central discontinuity or join, and a larger area of masked-out data beneath the bar images.

2 When phase-stepping, it is necessary to re-align the interferometer for each wavelength because otherwise the extra tilt due to dispersion in the beamsplitter would defeat the discontinuity-removal algorithms (see figure 4.23).

3 The discontinuity removal for the bar now has to work on 2 separate bar faces.

4 The platen discontinuity removal has to 'bridge' the join in the centre of the image - it maps the pixels to the immediate left and right of the join to the same phase modulo 2π and then uses linear interpolation to fill the gap between them.

5 The 3-pass routine now needs a fourth pass in the lower-centre of the image to unwrap data in between the bars, near the supports.

6 Two fringe fractions are measured after using a least-squares-plane analysis - there is insufficient data to use multiple chebychev fits.

7 The multiple-wavelength calculation routine now uses double-ended fringe fraction results (see § 3.3.4).

The lower contrast of the fringes in the background of the image (previously the platen area) and the necessity to use best-fit-plane analysis result in larger phase-measurement errors in this region. This means that the fringe fraction measurements are not as accurate as the single-ended measurements and the reference surface (background) phase data is not flat. Measurements made using double-ended interferometry will not be as accurate as single-ended measurements and the scanning range of the multiple-wavelength analysis is restricted to prevent spurious results being selected. Results of some double-ended measurements are given in chapter 9.

The phase-stepping and refractive index and temperature measurements are performed as before. The spread in `alpha` (phase-step angle) appears larger than for single-ended measurements due to the low fringe contrast producing errors in the phase-step size calculation, but the mean step sizes for the three wavelengths are unchanged. This is

because it is the reference mirror which is stepped so the phase-stepping algorithm is the same as used for the single-ended measurements with the same step size. The fringe fractions are calculated as before, except that the signs are reversed: the fitted data of the background is subtracted from the measured phase maps then the phase values at the centres of the bar images are summed to give three complementary fringe fractions, *i.e.* $1-f_r$, $1-f_g$ and $1-f_o$. This is because equation (3.17) requires fringe fractions of the opposite sign to those measured in single-ended interferometry.

REFERENCES FOR CHAPTER 6

[**1**] Ghiglia D C, Mastin G A & Romero L A Cellular automata method for phase-unwrapping *J. Opt. Soc. Am.* **4** (1987) 267-280

[**2**] Stetson K A Phase-step interferometry of irregular shapes by using an edge-following algorithm *Appl. Opt.* **31** (1992) 5320-5325

[**3**] Gierloff J J Phase unwrapping by regions *SPIE* **818** (1987) 2-9

[**4**] Rolt F H *Gauges and Fine Measurement* (MacMillan & Co.: London) (1929) 49-50, 212-213

[**5**] Creath K Step height measurement using two-wavelength phase-shifting interferometry *Appl. Opt.* **26** (1987) 2810-2816

[**6**] Cheng YY & Wyant J C Two wavelength phase shifting interferometry *Appl. Opt.* **23** (1984) 4539-4543

[**7**] Cheng YY & Wyant J C Multiple wavelength phase-shifting interferometry *Appl. Opt.* **24** (1985) 804-807

[**8**] Pugh D J & Jackson K Automatic gauge block measurement using multiple wavelength interferometry *Proc. SPIE* **656** (1986) 244-249

[**9**] Guzhov V I & Solodkin Y N Using the properties of integers to decode interferograms *Opt. Spectrosc. (USSR)* **65** (1988) 661-665

[**10**] Walsh C J Limit to multiwavelength interferometry imposed by frequency instability of the source *Appl. Opt.* **26** (1987) 29-31