John Foster
and Associates Inc.
September 16, 1977

NASA Representative
Scientific & Technical Information Facility
P.O. Box 8757
Baltimore/Washington International Airport,
Maryland 21240


Reference: Program Code 358-41-06

The subject report prepared under Contract NAS 2-9432 has been reviewed at Ames and is recommended for release in STAR as CR-152004.

Ralph W. Lewis
Chief, Library Branch

Enclosure:
1 cy subject report

cc:
NASA Hqrs., Code KSI (w/o encls.)
# TABLE OF CONTENTS

1.0 INTRODUCTION ............................................. 1

2.0 TECHNICAL ASPECTS ........................................ 3

2.1 Detection of Signals in Noise ......................... 3
2.2 Correlation Detection .................................. 5
2.3 Matched Filters and Non-White Noise ............... 12
2.4 Karhunen-Loeve Filtering .............................. 14
2.5 Digital Filtering ..................................... 16

3.0 ALGORITHMS AND SOFTWARE ............................. 24

3.1 A Digital Filter Design Aid ............................ 24
3.2 Infrared Filter Calibration ............................ 34
3.3 A Basic Source Detection Routine .................... 47

4.0 APPENDIX .................................................. 62

4.1 A Signal Processing Game .............................. 62
LIST OF ILLUSTRATIONS

1. Error Rate for Correlation Detection ............... 8
2. Probability of Detection for Correlation Receivers ........ 10
3. Bandwidth Variations with Source Temperature ........... 36
4. Effective Wavelength Variations .................. 37
5. Color Temperature-Magnitude Calibration ............. 38
1.0 INTRODUCTION

This is the second report on data processing techniques useful for infrared astronomy data analysis systems. As with the first report (NASA CR-151943), the investigation is restricted to consideration of data from space-based telescope systems operating as survey instruments. In this report the theoretical background for specific point-source detection schemes is completed, and the development of specific algorithms and software for the broad range of requirements is begun.

Section 2 develops the detail detection tests and processing requirements for point-source surveys and evaluates the performance measurement processes. The details of peak detection decisions and correlation detection are covered for the case of general bandlimited white gaussian noise. For non-white noise, a modified correlation test and a matched filter test are presented. A technique for resampling the data which is equivalent to a matched filtering approach is discussed which automatically decorrelates the noise. Implementation of this Karhunen-Loeve filtering is necessarily complicated, but for some kinds of noise an acceptable approach.

Section 3 then reviews a basic processing task to indicate where computation is needed outside of the normal data stream. While the processing used in the primary data reduction task is important, the actual software depends heavily on the specific mission hardware and is best approached anew for each task using the theories of Section 2 of this report and of the previous report, and of several cited authors. For the general signal processing task, a routine for designing digital filters is given based on the theory of Section 2.5. The calibration of detector-filter systems is the most complicated of the tasks off the main processing line; a routine which provides this calibration for blackbody or other input spectra. Finally, the preliminary processing routine for a previous survey
program is presented and briefly discussed to indicate how much processing can be done in a single pass of the data.

The Appendix in Section 4 presents an interesting game which can develop a fuller appreciation and understanding of the complexities of data analysis.
2.0 TECHNICAL ASPECTS

This section completes the task begun in the first report of reviewing the theoretical basis for the design of point-source survey data analysis software. The detection techniques for single-channel signal and noise processing are reviewed. The schemes reviewed include peak detection, optimal filtering, correlation, and Karhunen-Loeve filtering. The details of digital filtering, which is applicable to many aspects of data processing, are reviewed in the final section.

2.1 Detection of Signals in Noise

In most communication systems the errors (false detections and missed signals) are assumed to be of equal importance and with known probabilities. In more general detection problems, however, the a priori probabilities and costs of those errors are difficult to determine. The Neyman-Pearson test was first applied in such a case to radar detection with a peak measurement technique. The criterion can also be applied to more sophisticated detection methods, and in all cases, will give the highest probability of detection at a chosen false-alarm rate. The type of technique used depends on the amount of information available about the expected signal; generally, more information used will result in a higher detection probability at the chosen false-alarm rate. The likelihood ratio is the test used where the hypothesis is chosen if:

\[ \lambda = \frac{p(s)}{p(n)} \geq \eta \]

and the counter-hypothesis (no signal) is chosen otherwise. Here \( p(s) \) is the probability density function of the data with a signal present and \( p(n) \) is the p.d.f. of the noise alone, and \( \eta \) is the decision level chosen to satisfy the false-alarm constraint.

Consider the case of a signal in white noise, such that the signal has a normalized mean value of one. The probability functions are:
\[ p(s) = \frac{1}{\sqrt{4\pi}} e^{-\frac{(y-1)^2}{4}} \quad \text{and} \quad p(n) = \frac{1}{\sqrt{4\pi}} e^{-\frac{y^2}{4}} \] 2.1-2

Then the likelihood ratio test is:

\[ \lambda(y) = e^{\frac{y}{2} - 1} \geq \lambda_0 \] 2.1-3

To determine the threshold \( \lambda_0 \), the false-alarm probability is found from:

\[ P(f.a.) = \int_{\infty}^{\lambda_0} \frac{1}{\sqrt{4\pi}} e^{-\frac{y^2}{4}} \, dy \] 2.1-4

If we want a false-alarm rate of 10% or less, then \( \gamma = 1.8 \), and we choose the hypothesis if \( y \geq 1.8 \).

The probability of detection for a single test observation is:

\[ P(\text{det.}) = \int_{\infty}^{\gamma} \frac{1}{\sqrt{4\pi}} e^{-\frac{(y-1)^2}{4}} \, dy = 0.285 \] 2.1-5

In terms of the likelihood ratio, note that \( \lambda(y) = \lambda_0 = 1.9 \) and we make a detection whenever \( \lambda(y) \geq 1.9 \). To improve this rather mediocre performance, several measurements may be tested. With the same false-alarm rate, we choose the decision level differently. If we take \( n \) independent samples, the signal-present probability distribution has unity mean and a variance of \( \sigma^2 \), and:

\[ p_s(y_1, y_2, \ldots, y_n) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(y_1-1)^2}{2\sigma^2}\right] \times \ldots \] 2.1-6

\[ \times \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(y_n-1)^2}{2\sigma^2}\right] \]

Similarly, the noise-only probability distribution is:

\[ p_n(y) = \left(\frac{1}{2\pi\sigma^2}\right)^n \exp\left[-\frac{1}{2} \sum_{i=1}^{n} \frac{y_i^2}{\sigma^2}\right] \] 2.1-7
Taking the logarithm of the likelihood ratio, the decision test is:

\[ \frac{1}{n} \sum_{i=1}^{n} y_i \geq \lambda'_0 \]

where

\[ \lambda'_0 = \frac{1}{2} + \frac{\sigma^2}{n} \ln \lambda_0 \]

Now the probability of each kind of error is different, and we choose \( \lambda'_0 \) by evaluating \( P_1 (\text{f.a.}) + P_2 (\text{missed signal}) \), where

\[ P_1 (\text{f.a.}) = \int_{\lambda'_0}^{\infty} p_n(y) \, dy = \int_{\lambda'_0}^{\infty} \left( \frac{n}{2\pi\sigma^2} \right)^{1/2} e^{-n(y-1)^2/2\sigma^2} \, dy \]

and

\[ P_2 (\text{m.s.}) = \int_{-\infty}^{\lambda'_0} p_s(y) \, dy = \int_{-\infty}^{\lambda'_0} \left( \frac{n}{2\pi\sigma^2} \right)^{1/2} e^{-n(z-1)^2/2\sigma^2} \, dz \]

It now becomes clear that improving the performance of simple peak-detection schemes becomes a complicated task even using very little information about the signal. If we use more of the information available, and some of the knowledge about the nature of the noise, a more successful detection scheme can be derived. When multiple detections are made on a single source, the above can be used to evaluate the detection probability.

### 2.2 Correlation Detection

Rather than make a detection test based on only the peaks of the data stream as in the previous example, consider how we might deal with detecting a signal that we know. Let \( r_k, k=1, \ldots, m \) be the sequential data samples. Assuming additive noise,

\[ r_k = \{S_k\} + n_k \]

\[ 2.2-1 \]
where $S_R$ is the $k^{th}$ value of our expected signal, and $n_k$ is the noise sample. We may now derive a likelihood ratio test which uses this information.

First, assume that the noise is bandlimited white noise with power spectral density $S(\omega) = N_0/2$ for $|\omega| < \Omega$ and zero otherwise. The noise autocorrelation function then is given by:

$$R(\tau) = \frac{N_0 \Omega}{2\pi} \frac{\sin (\Omega \tau)}{\Omega \tau}$$  

2.2-2

This has its first zero at $\tau = \pi/\Omega$ so that if the received signal is sampled at intervals $\Delta t = \pi/\Omega$ the samples will be uncorrelated, and being gaussian they then will be statistically independent.

The probability density functions of the two cases will be:

$$p_s(r) = \left(\frac{1}{2\pi \sigma_s^2}\right)^{m/2} \exp \left[ -\sum_{k=1}^{m} \frac{(r_k - S_k)^2}{2\sigma_s^2} \right]$$

and

$$p_n(r) = \left(\frac{1}{2\pi \sigma_n^2}\right)^{m/2} \exp \left[ -\sum_{k=1}^{m} \frac{(r_k)^2}{2\sigma_n^2} \right]$$

and the logarithm of the likelihood ratio test results in the decision test:

$$\sum_{k=1}^{m} \frac{r_k S_k}{\sigma_n^2} \geq \ln \lambda_0 + \frac{1}{2} \sum_{k=1}^{m} \frac{S_k^2}{\sigma_n^2}$$  

2.2-4

Now the left-hand side of 2.2-4 is just the normalized cross-correlation coefficient of the signal with its expected template. Furthermore, the variance of the noise $\sigma_n^2$ is just the noise autocovariance function at zero frequency,

$$\sigma_n^2 = \frac{N_0 \Omega}{2\pi}$$  

2.2-5
Since one of our two signals is zero (noise only), we may define the average signal energy $E$ and the time cross-correlation coefficient $\rho$ by:

$$E = \frac{1}{2\pi} \sum_{k=1}^{m} S_k^2$$

and

$$\rho = 0$$

By extending equations 2.2-3 to infinite bandwidth $\Omega \rightarrow \infty$, the probability density functions for the signal case and the noise-only case can be derived as:

$$P_n(G) = \left[ \frac{1}{2\pi N_0} \right]^{1/2} \exp \left[ -\frac{(G+E)^2}{2N_0E} \right]$$

$$P_s(G) = \left[ \frac{1}{2\pi N_0} \right]^{1/2} \exp \left[ -\frac{(G-E)^2}{2N_0E} \right]$$

Since the false-alarm rate and the missed sources probabilities are equal when the samples are uncorrelated, the error rate is:

$$P_e = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-z^2/2} \, dz$$

where

$$\gamma = (E/N_0)^{1/2}$$

and thus we can determine an error rate based on the signal-to-noise ratio, independent of the shape of the signal. Figure 1 shows the error rate as a function of the signal-to-noise power ratio. Note that as long as the noise samples are uncorrelated, the error rate is also independent of the number of samples in the correlation sum. This apparently unreasonable result is directly related to the assumption of statistically independent samples. For bandlimited
\[
\frac{E}{N_0} (dB) = 10 \log \left( \frac{E}{N_0} \right)
\]

**Figure 1.** Error Rate Performance for Correlation Detection
white noise there must be \( m = \Delta t/\pi = \) constant independent samples in the
interval 0 to \( T \). Oversampling the signal may actually result in degraded performance, as will be discussed in section 2.4. If we choose the value of \( \gamma \) in 2.2-8 to achieve a desired error rate, then the probability of detection is

\[
P_D = \int_{-\infty}^{\infty} \frac{1}{(2\pi)^{1/2}} \exp \left[ - \frac{z^2}{2} \right] dz
\]

which is shown in Figure 2 as a function of signal-to-noise ratio and error rate.

If we have chosen the normalized signal template properly, our detection test simultaneously makes a best estimate of the signal amplitude. If the signal model is written as a function of a constant-amplitude factor \( A \), then the maximum likelihood estimate of that amplitude is the solution of:

\[
\sum_{i=1}^{m} \left[ r_i - s_i (A) \right] \frac{\partial s_i (A)}{\partial A} = 0
\]

2.2-10

or, writing \( s = A s' \), we want the solution of:

\[
\sum_{i=1}^{m} \left( r_i - \hat{A} S_i \right) S_i = 0
\]

2.2-11

That solution is

\[
\hat{A} = \frac{\sum_{i=1}^{m} (r_i S_i)}{\sum_{i=1}^{m} S_i^2}
\]

2.2-12

and now if \( S_i \) was normalized such that \( \sum S_i^2 = 1 \), and we re-arrange the terms in 2.2-4, we have the detection test and amplitude estimate simultaneously:

\[
\hat{A} = \sum_{k=1}^{m} r_k S_R \geq \sigma_n^2 \ln \lambda_o + \frac{1}{2}
\]

2.2-13
Figure 2.
Now it is clear how the correlation test is a better detector than a peak test. The correlation test, takes an average of the signals weighted by the expected response as a best estimate of the amplitude. Because it is using n samples of the signal, the improvement in error rate can be as much as $\sqrt{n}$. The uncertainty in the estimate is determined from the noise autocovariance function, as in section 2.3 as:

$$
\sigma^2_A = \int_0^T \int_0^T s(\tau) s(z) R_n(z-\tau) \, dz \, d\tau \tag{2.2-14}
$$

If we have N multiple pulses available from a single source, the decision test 2.2-4 can be modified to:

$$
\sum_{i=1}^{N} \sum_{k=1}^{m} r_k s_k \geq \sigma^2_n / n + 1 \sum_{i=1}^{N} \sum_{k=1}^{m} s_k^2
$$

The false-alarm rate given by 2.2-8 is not changed, since we are designing our test for a chosen error performance. However, the detection probability improves; the new detection rate is given by 2.2-9 by replacing $E$ with $E'$:

$$
E' = \sum_{i=1}^{N} E_i
$$

And since all signal energies for a given source are equal, there will be a 3 dB increase in the equivalent performance for each doubling of the number of signals.

As a final note to this discussion, the signal-to-noise ratio used here is the more useful signal power-to-noise power ratio, not the typical peak-to-rms value which has little physical meaning.
2.3 Matched Filters and Non-White Noise

The cross-correlation term on the left-hand side of equation 2.2-4 can be replaced by the equivalent matched filter. If the filter's transfer function is $h(t_i)$, then the output of the filter is

$$e(t_k) = \sum_{i=1}^{m} h_i r_{k-i}$$

and by inspection the filter output matches the correlator output if

$$h_i = S_{m-i}$$

That is, the matched filter is the time-reversed image of the signal expected. It is important to note, however, that because of the time reversal, the matched filter and the correlator output are equal only at time $T$, where the entire signal train (in samples) is within the bounds of the filter or of the correlator.

The matched filter representation is well suited to the case of non-white noise. We will show that the optimal detector for non-white noise replaces the left-hand side of 2.2-4 with a filter which is the product of the white-noise matched filter and a pre-whitening filter described in terms of the autocovariance function of the noise. To avoid confusing subscripts, we shall write the filter transformations in terms of time integrals which are the generalized extensions of the summations in section 2.2. The output of the filter at time $T$ is:

$$e(T) = \int_{0}^{T} h(\tau) r(T-\tau) \, d\tau$$

The signal and noise components are easily identified as

$$S(T) = \int_{0}^{T} h(\tau) S(T-\tau) \, d\tau$$

$$N(T) = \int_{0}^{T} h(\tau) N(T-\tau) \, d\tau$$
The noise power then can be written in terms of the autocovariance function as

\[ N(T) = \int_{0}^{T} h(\tau) n(T-\tau) \, d\tau \]

The optimum signal-to-noise ratio can be found by minimizing the Lagrangian:

\[ L = \int_{0}^{T} \int_{0}^{T} h(\tau) h(z) R_n(z-\tau) \, d\tau \, dz - \mu \int_{0}^{T} h(\tau) s(T-\tau) \, d\tau \]

The resulting variation yields:

\[ \int_{0}^{T} h_0(z) R_n(T-z) \, dz = s(T-T) \]

The filter which satisfies this relation will maximize the signal-to-noise ratio for a known signal in any noise with autocovariance function \( R_n(\tau) \). Equation 2.3-6 is, of course, a Fredholm integral equation of the first kind which is solvable only for a restricted group of covariance functions \( R_n(\tau) \). If, however, we can adequately approximate the integration by replacing the 0 to T limits with \(-\infty\) to \(+\infty\), then the Fourier transform of 2.3-6 gives immediately

\[ H(s) = \frac{S^*(s)e^{-sT}}{S_n(s)} \]

where \( s = iw \) and \( S_n(s) \) is the actual power spectral density function of the noise. This matched filter is then just the white-noise matched filter convolved with the actual noise spectrum. This result was derived for the limit \( T \rightarrow \pm \infty \), but a detailed derivation shows that it holds wherever the data samples are uncorrelated, which was determined from the zeros of the noise autocovariance function.
As in the previous section, a best estimate of the signal amplitude exists in the presence of non-white noise. That estimate is given by:

\[
\hat{A} = \frac{\int_{0}^{T} h(\tau)r(\tau) d\tau}{\int_{0}^{T} h(\tau)s(\tau) d\tau}
\]

where \(h(\tau)\) is the solution of:

\[
s(t) = \int_{0}^{T} R_n(t-\tau) h(\tau) d\tau
\]

Comparing this result with 2.3-6, we see that the optimal whitening filter is the best weighting function for the correlation detector and the amplitude estimate in the presence of non-white noise.

2.4 Karhunen-Loeve Filtering

The emphasis in the preceding section was an additive white noise. Since this is often invalid, we derived a test based on the noise autocovariance function. For the white noise case we considered a flat bandlimited spectrum and found that appropriate uniformly spaced amplitude samples were statistically independent. For colored noise we considered the continuous sampling limit and wrote the detection equations as integral relationships. However, uniformly spaced samples in colored noise are correlated and the sampled case is difficult to evaluate explicitly. There is, however, another method which can be used to generate statistically independent samples. While these are not amplitude samples, they can be used to construct the same detection and performance tests as previously described. The approach used will be to expand the signal in a series of functions which are orthogonal over the region 0 to T.

The functions we seek are a set of \(f_i(t)\)'s with the normality condition:
\[
\int_0^T f_i(t) f_j^*(t) \, dt = \begin{cases} 1 & \text{if } i=j, \\ 0 & \text{otherwise}. \end{cases} \tag{2.4-1}
\]

and given these functions, the new samples \(r_k^r\) of the data are given by:

\[
r_k^r = \int_0^T r(t) f_k(t) \, dt \tag{2.4-2}
\]

we also need the re-sampled signal template:

\[
s_k = \int_0^T s(t) f_k(t) \, dt \tag{2.4-3}
\]

The eigenfunctions \(f_k(t)\) are the solutions of the integral equation:

\[
\lambda_j f_j(t) = \int_0^T f_j(x) R_n(t-x) \, dx \tag{2.4-4}
\]

Now we may write the probability density functions for the new sample set, as:

\[
P_s = \prod_{k=1}^{N} \left( \frac{1}{2\pi\lambda_k} \right)^{1/2} \exp \left[ -\frac{(r_k^r - s_k)^2}{2\lambda_k} \right] \tag{2.4-5}
\]

\[
P_n = \prod_{k=1}^{N} \left( \frac{1}{2\pi\lambda_k} \right)^{1/2} \exp \left[ -\frac{r_k^2}{2\lambda_k} \right]
\]

and the detection test becomes:

\[
\sum_{k=1}^{N} \frac{s_k r_k}{\lambda_k} \geq \ln \lambda_0 + \frac{1}{2} \sum_{k=1}^{N} \frac{s_k^2}{\lambda_k} \tag{2.4-5}
\]

which is identical to 2.2-4 except that \(\sigma_n^2\) has been replaced by the eigenvalues \(\lambda_k\), and the signal samples have been transformed by a weighting function similar to the whitening filter of section 2.3. In this case, however, equations 2.4-2 through 2.4-4 can be written as sums over the time sampled values with no loss of generality, hence, with no degradation in performance caused by correlated samples.
2.5 Digital Filtering

The transformations of sections 2.2 through 2.4 can be written as filter transfer functions. Additionally, empirical methods can be used to synthesize a desired transfer function and the equations of those sections can then be used to evaluate the error rate and detection performance. This latter course is often followed when the sampling rate is constrained by some considerations other than those requiring uncorrelated noise samples. Typically the desired filtering is matched to the sample rate and the signal dwell time by the Nyquist theorem and we wish to evaluate the detection performance of such systems. Additionally, it may be desirable to further filter the data to improve the signal-to-noise ratio based on the observed noise spectrum. In this section we will discuss how such a transfer function could be synthesized and then derive the algorithm for converting that analog transfer function to a digital difference equation.

Given an analog impulse response function \( H(S) \), the difference equation for the filter function can be derived. Also, given the nominal characteristics desired, the transfer function can be synthesized. Both of these techniques are described below.

The frequency response can generally be described as a series of first-order filters. The transfer function of a low-pass filter is:

\[
H_1(S) = \left( \frac{a_i}{a_i + S} \right) G_L
\]

where \( a_i = 2\pi f_i \), \( G_L \) is the gain of the filter.

\( f_i \) = the corner frequency of the filter (Hz).

Graphically:
That is, the response of a low-pass filter is flat for $f < f_i$, and falls at 6 dB per octave (linearly on a log A-log f graph).

For a high-pass filter,

$$H_i(S) = \left(\frac{S}{a_i + S}\right) G_H$$

2.5-2

which appears as:

![Diagram of a high-pass filter response]

and the slope is the same as before. Higher order forms of these filters have transfer functions which are powers of the above $H(S)'s$, with the exponent $n$ equal to the order of the filter. That is, a 3rd order high-pass filter is:

$$H(S) = \left(\frac{S}{a_i + S}\right)^3 G_H$$

2.5-3

and its response slope increases by a factor of $n$ (3 in this example):

![Diagram of a 3rd order high-pass filter response]

Finally, a circuit which can be described by a series of such filters has a transfer function which is a product of the elemental $H_i(S)$ terms, and a response curve which is a series of line segments with $n(\pm 6 \text{ dB})$ quantum slope changes at each characteristic frequency.
A representative example is demonstrated by the following. The filter consists of a first-order high-pass filter of $f_1 = 4$ Hz, and a second-order low-pass filter of $f_2 = 40$ Hz. In addition, the detector acts as a low-pass filter of order 1 at $f_3 = 1$ Hz. The overall transfer function is then:

$$H(S) = \left( \frac{S}{a_3 + S} \right) \left( \frac{a_2}{a_2 + S} \right) \left( \frac{a_1}{a_1 + S} \right)^2 G$$

2.5-4

\[ a_3 = 2\pi \cdot 4 \text{ Hz} \]
\[ a_2 = 2\pi \cdot 1 \text{ Hz} \]
\[ a_1 = 2\pi \cdot 40 \text{ Hz} \]

which is pictured as:

![Diagram showing the frequency response of the system with marked frequencies and a representative graph of the transfer function.](image-url)
The definition of the corner frequency is important here. If we have defined the \( f_i \) points as the traditional 3 dB or half-power response frequencies, then the values used in 2.5-1 and 2.5-2 for \( a_i \) must be altered somewhat. Note, for example, that for a single order filter \( (m=1) \), we have the half-power response at

\[
H(S) \times H'(S) = \frac{a_i^2}{a_i^2 + S^2} \Rightarrow S = a_i
\]

as expected. For an \( m \)th order filter, we find

\[
H(S) \times H'(S) = \frac{1}{2} \left( \frac{a_i^2}{a_i^2 + S^2} \right)^m \Rightarrow S = \left( \frac{1}{\sqrt{2}} - 1 \right) a_i^2
\]

and similarly, for a high-pass filter:

\[
S = \frac{a_i}{\left( \frac{1}{\sqrt{2}} - 1 \right)^{1/2}}
\]

Given the transfer function \( H(S) \), the difference equation can now be determined as follows. First, transform the frequencies. Since we desire the digital equivalent frequency, determine \( A_i \) by:

\[
A_i = \tan \left( \frac{2\pi f_i T}{2} \right) = \tan \left( \frac{a_i T}{2} \right)
\]

where \( T \) is the sampling interval, \( = 1/\text{SR} \) (SR is the sample rate in Hz). Second, transform the \( H(S) \) function to an \( H(Z) \) function by the substitution:

\[
S \rightarrow \frac{Z-1}{Z+1}
\]

This transformation preserves the square-magnitude response of the system except for a warping of the frequency scale as given by the first relationship (2.5-5). The advantage of this transformation is that aliasing is not introduced by the digital representation, thus avoiding the necessity of "guard" filters (anti-aliasing) which would result in a digital filter of higher order than the original analog transfer function.
The technique then is to express \( H(Z) \) in terms of a polynomial in \( Z \), as:

\[
H(Z) = G \times \frac{P(Z)}{Q(Z)}
\]

with the order of \( Q(Z) \) equal to or greater than \( P(Z) \). Most easily, the substitution used is:

\[
(A_i + S) + \frac{(A_i - 1)(Z + U_i)}{(Z+1)}
\]

where \( U_i = \frac{(A_i - 1)}{(A_i + 1)} \) and \( P_i = A_i + 1 \).

The numerator and denominator of \( H(Z) \) are then divided by \( Z^n \), where \( n \) is the order of the denominator, resulting in a transfer function which is a ratio of two polynomials of equal order in powers of \( Z^{-1} \). Equation 2.5-9 is used for \( S \) terms in the numerator. The general form of this \( H(Z) \) is:

\[
H(Z) = \frac{Z^{-n}(Z+1)^{n-m}(Z-1)^m}{(1 + \sum_{i=1}^{n} S_i Z^{-i})} \prod_{i=1}^{n} \frac{A_i}{P_i} \cdot G_i
\]

where:

- \( A_i \) is as defined previously;
- \( n \) = the number of elemental filters;
- \( m \) = the number of high-pass filters, and the \( A_i \)'s are ordered with the low-pass frequencies first.

\( P_i = A_i + 1 \)

and

\( S_i \) are the expanded coefficients of the product function:
\[ n \prod_{i=1}^{n}(U_i + Z) \]
such that:

\[ S_1 = \sum_{i=1}^{n} U_i \]

\[ S_2 = \sum_{j=2}^{n} U_j \left( \sum_{i=1}^{j} U_i \right) \]

\[ S_3 = \sum_{j=3}^{n} U_R \left[ \sum_{j=2}^{k-1} U_j \left( \sum_{i=1}^{j-1} U_i \right) \right] \]

\[ S_r = \sum_{m=r}^{n} U_R \left[ \sum_{m=1}^{m-1} U_R \sum_{k=4-2}^{\infty} \left( \sum_{i=1}^{j-1} A_i \right) \right] \]

Again from our example 2.5-4, the \( H(S) \) transforms to:

\[ H(Z) = \frac{Z^{-4}(Z-1)(Z+1)^3}{Z^{-4}(Z+U_1)^2(Z+U_2)(Z+U_3)} \frac{A_1^2 A_2}{P_1^2 P_2 P_3} G \]

The \( Z \)-transform corresponds exactly with 2.5-12 if we consider the four elements of the filter as having frequencies (in the original form) of 1, 4, 40, and 40 Hz. That is, \( U_1 \) is repeated, but treated as if it were two different terms. Ignoring the constant factor temporarily:

\[ H(Z) = \frac{1-2Z^{-1} - 2Z^{-3} - Z^{-4}}{1 + \sum_{i=1}^{S_1} Z^{-1}} \]
where $S_1 = 2U_1 + U_2 + U_3$

\[ S_2 = U_1U_1 + 2U_1U_2 + 2U_1U_3 + U_2U_3 \]

\[ S_3 = U_1U_1U_2 + U_1U_1U_3 + 2U_1U_2U_3 \]

\[ S_4 = U_1^2U_2U_3 \]

Now that we have the Z-transform of the $H(S)$ transfer function, the difference equation for the system can be written. Noting that $Z^{-1}$ is the unit delay function, and writing $H(Z)$ as:

\[ H(Z) = \frac{1 + \sum_{i=1}^{n} T_i Z^{-i}}{1 + \sum_{i=1}^{n} S_i Z^{-i}} = \frac{Y(Z)}{X(Z)} \]

where $Y(Z)$ is the Z-transform of the output, and $X(Z)$ is the input transform. Inverting the transform we find:

\[ Y_j = X_j + \sum_{i=1}^{n} T_i X_{j-i} - \sum_{i=1}^{n} S_i Y_{j-i} \]

where $Y_j$ is the $j$th sample of the output series, and $X_j$ is the corresponding $j$th input value. Continuing our example,

\[ Y_n = X_n - 2X_{n-1} - 2X_{n-3} - X_{n-4} - S_1 Y_{n-1} - S_2 Y_{n-2} - S_3 Y_{n-3} - S_4 Y_{n-4} \]

It is interesting to note that because the transformation 2.5-9 is bilinear, the difference equation will always be the same order in $X_{n-i}$ and $Y_{n-i}$ terms (except for canceling of some $Z^i$ terms by the expansion of $(Z+1)^m(Z-1)^k$).

To find the $T_i$ coefficients of equations 2.5-17 and 2.5-18, we must expand

\[(1+Z)^{n-m}(1-Z)^m = 1 + T_1 Z + \ldots + T_n Z^n \]
but the gain constant in 2.2-12 of

\[
\prod_{i=1}^{m} A_i \prod_{i=1}^{n} \left( \frac{G_i}{\rho_i} \right) = \text{COEFF} \quad 2.5-21
\]

must be included as a factor of all the \( X_i \) terms.

Expanding 2.5-20 can be done using the binomial expansion subroutine attached to expand \((1+Z)^{n-m}\) and \((1-Z)^m\), and the polynomial product routine to find the resulting terms. Then if we redefine the subscript of \( T \) by one to absorb the 1 on the r.h.s. of 2.5-20, we can include the factor 2.5-21 easily into the definition of \( T_i \):

\[
(l+Z)^{n-m}(1-Z)^m - T_1 + T_2 Z + \ldots + T_{n+1} Z^n \quad 2.5-22
\]

if we also redefine the \( S_i \) and set \( S_1 = 0 \), then the relations 2.2-17 and 2.2-18 can be written:

\[
Y_i = \sum_{i=1}^{n+1} T_i \ X_{j-i+1} - \sum_{i=1}^{n+1} S_i \ Y_{j-i+1} \quad 2.5-23
\]

\[
= \sum_{i=1}^{n+1} (T_i \ X_{j-i+1} - S_i \ Y_{j-i+1})
\]

This redefinition of \( S_i \) has been included in the attached algorithm (see section 3.1).
3.0 ALGORITHMS AND SOFTWARE

Section 3 of the first report covered in detail the basic survey point-source processing scheme. However, in order to successfully complete the sky survey and to define the detection test gates, a number of peripheral routines are needed. This section will discuss several of the most important of these routines. Some routines are simple, such as coordinate transformations used for positional matching. Others, although complex and worthy of discussion, are very specifically written for each mission. These routines generally are part of the primary data processing system and are assembled from the formula and algorithms of section 2 of this report and section 3 of the previous paper. Yet, other routines which are a part of the database merging are decisions and tests for specific types of astronomical sources and depend on the sensor bandpasses and sensitivities and on the spectral characteristics of the sources being searched for. Some routines which are only peripheral to the primary detection scheme are so basic and important that they are worthy of individual discussion here. The complete set of programs designed and tested on the Celestial Mapping Program (CMP) data will be published at a later date when that task is completed.

Two basic programs will be covered here and one front-end detection scheme used on a previous survey program. First, we will discuss an application of the digital filter design scheme of section 2.5. Then the calibration of infrared detectors is discussed and a routine to evaluate the spectral response of a detector plus filter combination to a variety of stellar spectra.

3.1 A Digital Filter Design Aid

The discussion of section 2.5 covered the algorithm for digital filter design; here, we consider the specific use of the following routines. The program attached does two things. First, the coefficients of the difference equation
\[ Y_j = \sum_{i=1}^{N+1} (T_i X_{j-i+1} - S_i Y_{j-i+1}) \]

where the \( X_i \)'s are the input data, \( Y_i \)'s the output data, and \( S_i \) and \( T_i \) the filter coefficients determined from the desired frequency response.

Second, the routine creates a pair of sample response sequences. One is the impulse response function of the filter. If the filter characteristics were chosen to duplicate the response of a detector and its electronics, then this impulse response will model the radiation-hit response. The other response is the system reaction to a square wave. Since the duration of the square wave is equal to the point-source dwell time, the response is approximately the same as a source signature.
PROGRAM TRNSFN

PROGRAM TRNSFN(INPUT,OUTPUT)
COMPLEX C,SUM,COEFF,T

THE FOLLOWING DIMENSION STATEMENT IS USED TO CREATE A SAMPLE RESPONSE SEQUENCE FOR A RECTANGULAR INPUT EQUAL IN LENGTH TO THE STAR DWELL TIME, AND THE IMPULSE RESPONSE SEQUENCE.

DIMENSION XX(200),XP(200),YY(200),YP(200),TIM(200)

THE ROUTINE IS DIMENSIONED FOR TRANSFER FUNCTIONS OF TOTAL ORDER 9 OR LESS; FOR HIGHER ORDERS, CHANGE THE DIMENSIONS OF ALL THE FOLLOWING TO N+1, WHERE N IS THE DESIRED ORDER.

DIMENSION C(10),SUM(10),T(10),X(10),Y(10),Z(10),RS(10),TR(10)

PRINT 99
99 FORMAT(1HR)
A1=1HC & A2=1HS & A3=1HT & A4=1HF

N AND M DESCRIBE THE ORDER OF THE TRANSFER FUNCTION.
N = THE TOTAL ORDER
M = THE ORDER OF THE HIGH-PASS FUNCTIONS

THE PROGRAM WILL LOOP FOR NEW TRANSFER FUNCTIONS, TO END, SET M = 0. (THE LAST DATA CARD CAN BE A BLANK TO STOP)

CONTINUE
READ 100,N,M
100 FORMAT(2I5).
IF(N.EQ.0) GO TO 51

THE SAMPLE RATE SR GOVERNS THE FREQUENCY WARPING FOR THE S TO Z TRANSFORMATION.
THE DWELL TIME IS DETERMINED FROM THE SCAN RATE SCNRC AND THE DETECTOR SIZE SIZE.
SIZE AND SCNRC ARE VARIABLES USED ONLY FOR GENERATING THE TEST CASES.

READ 101, SH,SIZE,SCNRC
LIST THE INPUT PARAMETERS

PRINT 201, N, M, SR, LN
201 FORMAT(1H1, //, 20X, *N = *, I5, *, M = *, I5, *, SAMPLE RATE = *, F10.4, *, DWELL = *, I5, //)

THE CORNER FREQUENCIES FOR THE FILTER ELEMENTS CAN BE SPECIFIED AS COMPLEX (REAL + IMAGINARY)
THEY SHOULD BE ORDERED WITH THE M HIGH-PASS ELEMENTS LAST
PROGRAM TRNSFN  76/76  OPT=1  

C EACH ELEMENT HAS A (COMPLEX) GAIN WHICH CANNOT BE ZERO
C THE GAIN CAN BE SET AS (1,00,0,00)
C THE ORDER MUST BE EQUAL TO OR GREATER THAN ONE
C
C THE FREQUENCY CARDS ARE FORMATTED (4F10,6,I5) AS FOLLOWS:
C FREQ(REAL),FREQ(IMAG),GAIN(REAL),GAIN(IMAG),ORDER

65 PI=3.141592653579
COEFF=CMPLX(1,0,0,0)
NT=0
DO 1 I=1,N
READ 102,WDR,WDI,GAINR,GAINI,NORD
102 FORMAT(4F10,6,I5)
PRINT 205, I, WDR, WDI, NORD
205 FORMAT(IH ,1OX,*WDR(*2,*),*WDI(*1,*,I2,*),*)=*,2F10,6,5X,*ORDER=*,I5)
NN=NORD
IF((NT+M),GE,N) GO TO 7
WDR=WDR/(2o**(i/NORD)-#),*)
WDI=WDI/(2o**(i/NORD)-1e)
GO TO 8
7 WDR=WDR**(2o**(i/NORD)-1e)
WDI=WDI**(2o**(i/NORD)-1e)
8 CONTINUE
WAI=TAN(HDI*PI/ SR)
WAR=TAN(HDR*PI/ SR)

C LIST THE FREQUENCIES
131 CONTINUE
NT=NT+1
PRINT 202, NT, WDR, WDI, I, WAR, WAI, NT, GAINR, GAINI
202 FORMAT(IH ,1OX,*WDR(*2,*),*WDI(*1,*,I2,*),*)=*,2F10,6,5X
1,*GAIN(*,I2,*),*)=*,2F10,6)
C(NT)=CMPLX(WAR+1,*,WAI)/CMPLX(WAR+1,*,WAI)
COEFF=COEFF*CMPLX(GAINR,GAINI)/CMPLX(WAR+1,*,WAI)
IF((NT,LE,(N-M)) .COEFF=COEFF*CMPLX(WAR,WAI)
NN=NN+1
95 IF(NN,GT,0) GO TO 131
IF(NT,GE,N) GO TO 141
1 CONTINUE
141 CONTINUE

C LIST THE C(I) TERMS AND THE VALUE OF COEFF

100 C
DETERMINE THE S(I) COEFFICIENTS OF THE Y(J-I) TERMS.

DO 2 I=1,N
SUM(I)=CMPLX(0,0,0,0)
DO 4 I=1,N
SUM(I)=SUM(I)+C(I)
IF(I.EQ.N) GO TO 4
DO 3 J=2,N
IF(I+J-1.LE.N) SUM(J)=SUM(J)+SUM(J-1)+C(I+J-1)
CONTINUE

DO 5 I=1,N
      SUM(N+2) = SUM(N+1+I)
      SUM(I) = CMPLX(0,0,0,0)

LIST THE S(I) COEFFICIENTS

I=N+1
PRINT 203, (A2,ISUM(I),I=1,NN)

EXPAND THE NUMERATOR TO DETERMINE THE T(I) COEFFICIENTS OF X(J=1)

NOTE THAT T(I) HAS EXACTLY (N+M+1)*(M+1)+1 = N+1 TERMS

NP=N+M
CALL BINEXP(X,NP,1,00)
IDIMX=NP
NP=M
CALL BINEXP(Y,NP,1,00)
IDIMY=NP
CALL PMPY(X,IDIMX,Y,IDIMY,Z,IDIMZ)
DO 6 I=1,IDIMZ
   T(I) = Z(I)*COEFF
   TR(I) = REAL(T(I))
   RS(I) = REAL(SUM(I))
6 CONTINUE

NOW LIST THE T(I) COEFFICIENTS OF THE Y(J=1) TERMS

PRINT 203, (A3,IT(C);I=1,IDIMZ)

NOW ALL THE FACTORS ARE DETERMINED,
WE MAY NOW CALCULATE THE RESPONSE TO AN ARBITRARY INPUT SEQUENCE
OF X(I) VALUES AS

Y(J) = SUM(I=1 TO N+1) OF (( T(I)*X(J-I+1) - S(I)*Y(J-I+1) ))

THE FOLLOWING CARDS MARKED **** CREATE A SAMPLE RESPONSE ****
SEQUENCE, THE INPUT RECTANGULAR PULSE IS CREATED IN XX(I) ****
WITH A LENGTH EQUAL TO THE STAR Dwell TIME FOR THE INSTRUMENT ****
INPUT THE RESPONSE SEQUENCE IS IN YY(I) THE IMPULSE ****
DO 11 I=1,200
TIM(I)=0.0
IF(I,GE,10) TIM(I)=1000.*(I-20)/SR
XX(I)=0.00 $ XP(I)=0.00
YY(I)=0.00 $ YP(I)=0.00
11 CONTINUE
LL=I9+LN
DO 12 I=20,LL
PROGRAM TRNSFN 76/76 OPT=1

12 XX(I)=1,00

175 C
175 C GENERATE GAUSSIAN NOISE AND ADD IT TO THE INPUT DATA.
175 C
175 CALL RANSET(863211)
180 A=0.
180 NOISE=1 $ SNR=100.,
180 NOISE=1 $ SNR=10.,
180 NOISE=0
180 IF(NOISE,EQ,0) GO TO 123
180 DO 122 I=10,200
180 GNOISE=0.
185 DO 121 J=1,12
185 GNOISE = GNOISE+ RANF(A)
190 GNOISE = (GNOISE=6,0)/SNR
190 XX(I)=XX(I)+GNOISE
190 CONTINUE
190 CONTINUE

195 XP(10)=1,00
195 IL=N+1
195 JF=N+2
195 YYM=-99999., $ YPM=-99999.,
200 DO 14 J=JF,200
200 DO 13 I=1,IL
200 YY(J)=YY(J)+TR(I)*XX(J-I+1)-RS(I)*YY(J-I+1).
200 YP(J)=YP(J)+TR(I)*XP(J-I+1)-RS(I)*YP(J-I+1)
200 CONTINUE
200 CONTINUE
205 IF(YP(J),GT,YPM) YPM=YP(J)
205 IF(YY(J),GT,YYM) YYM=YY(J)
205 CONTINUE
205 DO 15 J=1,200
205 YY(J)=YY(J)/YYM $ YP(J)=YP(J)/YPM
205 CONTINUE
210 PRINT 300, YYM, YPM
300 FORMAT(1H1,10X,*YYMAX = *,F10.6,*, YP.MAX = *,F10.6,/) $ PRINT 111, (I,TIM(I),XX(I),YY(I),XP(I),YP(I),I=1,200)
310 111 FORMAT(1H1, *SEC X 1 XP YP*,//(3X,I3,5(5X,F10.3)))
215 GO TO 50.
ENTRY POINTS
45  TRNSFN

SYMBOLIC REFERENCE MAP (R=1).
3.2 Infrared Filter Calibration

The calibration of infrared brightnesses is the single most difficult aspect of a scanning sky survey. For ground-based point-and-integrate systems, it is possible, and in fact common, to make all measurements of the same signal-to-noise ratio by varying the integration. Since the amplitude uncertainty was shown to be a function of the signal-to-noise value in section 2.2, it is clear that uniform photometric accuracy is readily achieved. For survey instruments, the uncertainty of the initial measurements is inversely related to the signal-to-noise value, applying a fundamental limit to the accuracy of the survey measurements which varies both with brightness of the source and its location in the sky.

Further complicating the problem is the fact that the sources have a wide variety of spectrum so that the broad band detectors typical of infrared survey instruments do not have a well-defined intrinsic calibration. It is possible to calibrate the detector voltage in terms of the watts per cm² it receives. However, if the survey is measure sources in several colors, or if the calibrations are to be derived from measurements made in a difficult wavelength region, the measurements must be referred to a spectral intensity. The wavelength bandwidth that is needed, however, is dependent on the spectrum of the source being measured. Furthermore, the effective wavelength of that measurement varies with the input spectrum.

The units of the brightness measurement are another problem. The most useful form is the brightness magnitude, defined by

\[
m = -2.5 \log_{10} \left( \frac{B}{B_0} \right)
\]

where \(B\) is the observed brightness and \(B_0\) is the reference value. This reference is different for every filter, since it is defined as the response that filter-detector would observe from a particular "standard" star - the archetype is α-Lyra, which is defined as a
10,000°K blackbody source with an angular diameter of 1.5697 E-16 steradians. The great benefit of this magnitude measurement is that we skirt the question of effective bandwidth. These magnitude measurements still need an effective wavelength, but for blackbody spectra at wavelengths less than 50 micrometers, the effective wavelength varies only very slowly until the source temperature falls below 5000°K. Finally, the magnitude measurements defined by 3.2-1 can be used inversely to find the equivalent blackbody color temperature if measurements are available in two or more bands.

Figures 3, 4, and 5 show the variations in bandwidth, effective wavelength, and magnitude difference for three infrared filters similar to ones commonly used in previous surveys and measurements. The results were derived from the attached filter calibration routine which is self-explanatory.
Figure 4.
Figure 6.
PROGRAM INGRT (INPUT,OUTPUT,PUNCH,TAPES=INPUT,TAPE6=OUTPUT)

C PLANCK2 (MODIFIED)

--------------INTEGRATION CAN BE DONE WITH OR WITHOUT THE PLANCK FUNCTION,

C PROGRAM FOR INTEGRATING THE PLANCK BLACKBODY RADIATION FUNCTION

C OVER AN INTERVAL DETERMINED AND ATTENUATED BY FILTER-SENSITIVITY

C TYPE FUNCTIONS,

C DATA SHOULD BE IN ORDER OF INCREASING WAVELENGTH,

C NFUNCTION = NO. OF RESPONSE FUNCTION

C WRITE = 1 - WRITE WAVELENGTH, INTENSITY, RADIANCE, PLANCK INTENSITY,

C NFUNCTION = 0 - DO NOT WRITE,

C NFUNCTION = 1 - PLANCK FUNCTION TO BE CALCULATED,

C NFUNCTION = 0 - PLANCK FUNCTION NOT TO BE CALCULATED,

C TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C DATA DECK CONSISTS OF TRANS, DATA ONLY,

C NFUNCTION = 1 - NORMALIZED FUNCTION1(I) * FUNCTION2(I)

C NFUNCTION = 0 - DO 'T.Normalize

C I = 1 PLCT

C = 0 DO NOT PLOT

C = 1 c NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,

C = 1 - NORMAIZED FUNCTION1(I) * FUNCTION2(I)

C = 0 - DO 'T Normalize

C = 1 PLCT

C = 0 DO NOT PLOT

C = 1 TRANSMISSION DATA DECK CONSISTS OF WAV, AND TRANS,

C = 0 TRANSMISSION DATA DECK CONSISTS OF TRANS, DATA ONLY,
DEFINED WAVELENGTH IS THE FLAT RESPONSE EFFECTIVE WAVELENGTH.

WIDTH ZERO IS THE FLAT RESPONSE BANDWIDTH.

WAVELENGTH IS THE TRUE EFFECTIVE WAVELENGTH FOR $BB$, SPECTRA AT THE GIVEN TEMPERATURE.

THE TRUE EFFECTIVE WAVELENGTH IS THE INTEGRAL OF $B(\Lambda, T) \times R(\Lambda) \Lambda D\Lambda$ DIVIDED BY THE INTEGRAL OF $B(\Lambda, T) \times R(\Lambda) D\Lambda$.

WIDTH IS THE TRUE BANDWIDTH AT THE TRUE EFFECTIVE WAVELENGTH FOR THE GIVEN TEMPERATURE.

THE TRUE EFFECTIVE BANDWIDTH IS THE INTEGRAL OF $B(\Lambda, T) \times R(\Lambda) \Lambda D\Lambda$ DIVIDED BY $B(\Lambda, T)$.

WIDTHDP IS THE PROPER BANDWIDTH FOR THE DEFINED EFFECTIVE WAVELENGTH.

BR MAG IS $-2.5 \times \log_{10}(B(\Lambda, T) / B(\Lambda, 0, 0))$ WHERE $B(\Lambda, T)$ IS THE $BB$, EMISSION.

COL' MAG IS $-2.5 \times \log_{10}(\text{INT FLUX} / \text{REFERENCE FLUX})$.

INT FLUX IS THE INTEGRATED FLUX ON THE DETECTOR IN W*CM$^{-2}$STR$^{-1}$.

VIEW FLUX IS THE INTEGRATED FLUX MULTIPLIED BY THE FIELD OF VIEW $= 4.5 \times 10^{-6}$ STR AND IS IN W*CM$^{-2}$.

REAL MAGFAC

DIMENSION WAV(150), RES(150), TRA(150), TEMP(150), N(150), W(150), XX(150), RESP(150), WXX(150), TYPE(4), EXPER(4)

NJ = 0
OMEGA = 1.5697E-16
PI = 3.141592654
ICOL = 0
L = 0
READ (5, 94) NFUNCT, IRITE, NFUX, IAVTR, NORMAL, IPlot

94 FORMAT (6I3)
95 NJ = NJ + 1
NFUX = 0
DO 92 I = 1, 4
92 TYPE(I) = 6H
LOLD = L
ICOL = ICOL + 1
NN=0
READ (5,99) NTRANS
FURMAT(113)
REA (5,100) M,N,L
FURMAT(111,213)
TEMP(1)=10000.

* INSERT TEMP DEFINING CARDS; L=XX, TEMP(2)=XX, ETC.
L=70
TEMP(2)=100,
DO 20 I=1,42
20 TEMP(I)=TEMP(I-1)+10.
DO 21 I=43,52
21 TEMP(I)=TEMP(I-1)+50.
DO 22 I=53,70
22 TEMP(I)=TEMP(I-1)+500,
N1=1, N2=N
READ(5,101) (EXP1R(K),K=1,4),(WAV(I),RES(I),I=N1,N2)
FURMAT(4A6/(12F6,4))
N1=N+1
DO 500 I=L1,150
WAV(I)=0.00
RES(I)=0.00
TRAN(I)=0.00
RESP(I)=0.00
500 CONTINUE
N2=N
99 NN=NN+1
IF(M=1) 1,1,3
3 IF(IWAVTR.EQ.1) GO TO 601
READ (5,112) (TYPE(K),K=1,4),(TRAN(I),I=1,N)
FURMAT(4A6/(12F6,4))
GO TO 600
601 READ (5,112) (TYPE(K),K=1,4),(WAV(I),TRAN(I),I=1,N)
600 CONTINUE
DO 2 I=1,N,1
2 RESP(I)=RESP(I)*TRAN(I)
GO TO 11
1 DO 9 I=1,N,1
TRAN(I)=0.
9 RESP(I)=RESP(I)
887 CONTINUE
11 CONTINUE
IF(NORMAL.FG.1) GO TO 700
M0AT=1.
N1=1
220 NSPACE=0
   LINE=0.
   WRITE (6,712) (EXP(K),K=1,4), (TYPE(K),K=1,4)
212 FORMAT(1H1,37X,446/ 91X,19HTABLE OF INPUT DATA/ 38X,446/20X,
   110WAVELENGTH,3X,11FUNCTION(1),3X,11FUNCTION(2),5X,10WAVELENGTH
   2,3X,11FUNCTION(1),3X,11FUNCTION(2)/)
   IG=N1  $  INC=N2
   DO 114 I=IBG,INC
   NSPACE=NSPACE+1
   LINE=LINE+1
206 IF (5<LINE)204,206,206
   WRITE (6,210) WAV(I),RES(I),TRAN(I),WAV(I+35),RES(I+35),TRAN(I+35)
210 FORMAT(22X,F5,2,7X,F7,4,7X,F7,4,10X,F5,2,7X,F7,4,7X,F7,4)
   GO TO 114
204 NSPACE=1.
   IF (55<LINE)219,214,214
   WRITE (6,211) WAV(I),RES(I),TRAN(I),WAV(I+35),RES(I+35),TRAN(I+35)
211 FORMAT(1H0,21X,F5,2,7X,F7,4,7X,F7,4,10X,F5,2,7X,F7,4,7X,F7,4)
   GO TO 114
219 N=1+LINE-1
   IF (N2-N)250,220,220
114 CONTINUE
   GO TO 250
700 BIGP=RESP(I)
   DO 30 I=2,N
   IF (BIGP=RESP(I))32,30,30
   BIGP=RESP(I)
30 CONTINUE
   DO 33 I=1,N
   RESP(I)=RESP(I)/BIGP
33 CONTINUE
   N=1
720 NSPACE=0
   WRITE (6,712) (EXP(K),K=1,4), (TYPE(K),K=1,4)
712 FORMAT(1H1,46X,446/50X,19HTABLE OF INPUT DATA/ 47X,446/10X,
   110WAVELENGTH,3X,11FUNCTION(1),3X,11FUNCTION(2),3X,
   211FUNCTION(1X2),5X,10WAVELENGTH,3X,11FUNCTION(1),3X,
   311FUNCTION(2),3X,11FUNCTION(1Y2),5X,10NORMALIZED,47X,
   410NORMALIZED/)
   IG=N1  $  INC=N2
   DO 614 I=IG,INC
   NSPACE=NSPACE+1
   LINE=LINE+1
   IF (5<NSPACE)704,706,706.
506 WRITE (6,710) WAV(I),RES(I),TRAN(I),RESP(I),WAV(I+35),RES(I+35),
      TRAN(I+35),RESP(I+35)
510 FORMAT(12X,F5,2,3(7X,F7,4),10X,F5,2,3(7X,F7,4))
GO TO 614
504 NSPACE=1
514 WRITE (6,711) WAV(I),RES(I),TRAN(I),RESP(I),WAV(I+35),RES(I+35),
      TRAN(I+35),RESP(I+35)
511 FORMAT(1H0,11X,F5,2,3(7X,F7,4),10X,F5,2,3(7X,F7,4))
GO TO 614
519 I=A+LINE-1
512 IF(N2-A/I50,720,720
614 CONTINUE
750 CONTINUE
1111 FORMAT(F10.7,F10.8)
250 IF(NFLUX,EG,0) GO TO 307
507 WRITE(6,107) (TEMP(I),I=1,L)
107 FORMAT(// L5X,11HTEMPERATURE//(9F13.2))
809 J=0
61 J=J+1
50 IF(NFLUX,EG,0) GO TO 307
4 CALL SPECTRM (TEMP(J),WAV(I),RESP(I),X(I),WX(I),WXX(I),ZINT)
GO TO 501
307 DO 302 I=1,N
5 WXX=RESP(I)
5 WXX=WX(I)/PI
302 CONTINUE
50 IF(I=NUT,EG,0) GO TO 300
301 NSPACE=5
5 LINE=45
5 DO 216 I=1,N
5 NSPACE=NSPACE+1
5 LINE=LINE+1
5 IF(S-NSPACE)240,242,242
242 WRITE (6,120) WAV(I),WX(I),WXX(I),W(I)
120 FORMAT(30X,1PE12.5,4X,1PE12.5,5X,1PE12.5,4X,1PE12.5)
GO TO 216
246 NSPACE=1
5 IF(45-LINE)244,246,246
246 WRITE (6,122) WAV(I),WX(I),WXX(I),W(I)
122 FORMAT(1H0,29X,1PE12.5,4X,1PE12.5,5X,1PE12.5,9X,1PE12.5)
GO TO 216
244 LINE=1
5 IF(1=(EG,1) GO TO 242
GO TO 300
216 CONTINUE
300 H=AV(2)-WAV(1)
   X1H7=0,0
   KN=K-2
   DO 5 I=1,KN,2
   XIJ=XYX+(I/2)*X(I)+4,0*W(I+1)+W(I+2))
   X=INT
5 CONTINUE
   IF(NFLUX,EG.1) GO TO 140
   H=IFE (6,142) XINT
   HZERO=XINT
142 FORMAT(1/1X,J11B12W12H1M =1PE12.5,1H1M1K1M1K)
   AC=BC2=0.
   GO TO 144
140 XINT=XINT/PI
   X=PR=XINT
   YINT=0,0
   DO 7 I=1,KN,2
   YINT=XINT+(I/3)*X(I)+4,0*W(I+1)+W(I+2))
   CONTINUE
   X=INT
   EFX=XINT*BIGP/YINT
   TEFX=XINT*BIGP/ZINT
   DC=2,5*ALOG10(1/EFX)
   BC2=2,5*ALOG10(1/TEFX)
7 DO 126 I=1,KN
126 WWX(I)=X(I)*WAV(I)
   XINT=0,0
   DO 128 I=1,KN,2
   XINT=XINT+(I/3)*WWX(I)+4,0*WWX(I+1)+WWX(I+2))
128 CONTINUE
   XINT=XINT/PI
   EFFT=XINT/PI
   IF(NFLUX,EG.1) GO TO 146
   EFFT=PIEF0
   H=IFE (6,130) EFFT
130 FORMAT(1/1X,25H-EFFECTIVE WAVELENGTH = 1PE12.5)
146 CONTINUE
   IF(NFLUX,EG.0) WAVDEF=EFFT
   IF(NFLUX,EG.0) GO TO 8
   ZNON=0.
   CALL SPECNTF (TF,EF,T,ZNON,WAV,ZNON,ZNON,ZNON)
   WAV=WAV/PI
   IF(J,NJ,NJ) GO TO 4451
   H=HAY
AHREF=XIRP*OMEGA

PUNCH 222, AHREF

FORMAT(5X,**REFERENCE FLUX = *,1PE12.5)
PUNCH 223, KAVDEF, BWZERO

FORMAT(5X,**DEFINED WAVELENGTH = *,F12.8,** ZERO BANDWIDTH = *,F12.18)
PUNCH 221

FORMAT(1X,**NO,** TEMP WAVELENGTH BWIDTH BWTHDF BR MAG COL

1 MAG INT**FLUX VIEW FLUX**)

CONTINUE

CALL SPCTRIV (TEMP(J), KAVDEF, ZNON, BWAVDF, ZNON, ZNON, ZNON)

BWAVDF=BWAVDF/PI

IF(BWAV, EQ, 0, 0) GO TO 4441

DELWAV=XINT/BWAV

DELWDF=XINT/BWAVDF

CONTINUE

MAGFAC= BWAV/ AHREF

MAGFAC = 2.5*MAGFAC

BRMAG = 2.5*ALOG10((XIRP*OMEGA/AHREF)

AIRP = XIRP*4.5L-06

PUNCH 200, ICOL, TEMP(J), EFFT, DELWAV, DELWDF, MAGFAC, BRMAG, XIRP, AIRP

FORMAT(1X, I5, 2X, 6F9.3, 2E10.5)

IF(J-L) 6, 8, 8

CONTINUE

IF(NFLUX, EQ, 1) GO TO 688

NFLUX = 1

GO TO 887

CONTINUE

IF(NHNFUNCT) 98, 97, 97

IF(NHNFUNCT) 95, 96, 96

CONTINUE

STOP

END

SUBROUTINE SPCTRIV (TEMP, KAV, RESP, NAV, XX, XXX, XINT)

INSERT DESIRED EMISSIVITY HERE AS EMS = XXX

EMS = 1.00

PI = 3.141592654

IF(NAV, EQ, 0) GO TO 10

AI = 1.43879/(NAV*TEMP/10000.)

IF(NAV<.88) GO TO 10

BI = EXP(AI)

W = (3.741832E-16)/((NAV/10000.)*S)(BI-1.)*EMS

GO TO 10 12
\[ W = (3,741832 \times 10^{-16}) \exp(-A) / (C \times (M/0.0001) \times 5) \times \text{EMS} \]

\[ x = \text{KSP} \times n \]
\[ w = x / \pi \]
\[ \text{WNTG} = (5,6666 \times 12) \times (\text{TEMP} \times 4) / \pi \times \text{EMS} \]

RETURN

END
A Point-Source Detection Routine

The last routine presented here is a program developed for an early sky survey. The purpose in reviewing it here is to illustrate both the breadth of processing which can be done in a single pass of the data and also the complexity of the required software. The routine unpacks and de-commutates the data and checks for errors and gaps. Three background channels are processed, the running noise computed, and the data plotted. Within the basic detection loop, the data is tested for signal peaks, correlation peaks, and signal length, and the radiation hits are separated from the data. Estimates of the amplitude and bias level are made, and the position of the position of the signal is found from the time of detection.

Inspection of a sample portion of the preliminary detection list reveals some of the basic problems which the following merging routines will need to deal with. The most complex problem is that the correlation coefficient does not track well with some of the other measurements of a good signal. For example, source number 22 has a good correlation coefficient, but the estimated amplitude is less than half the peak height, and the amplitude estimate peak is significantly shifted in time from the data peak and also from the peak correlation coefficient. The correlation coefficient is below a reasonable error gate, but the same is true for signal number 10, where except for a slightly low ρ, the signal is very good. This pattern persisted throughout the data, and a careful study revealed that the poor ρ values were the result of an uncertainty in the detector bias. Since the sensor used bidirectional logarithmic amplifiers, the uncertainty in bias led to a possible error in de-compressing the amplifier functions which would tend to warp the signal shape significantly and degrade the value of the correlation coefficient, and also warping the noise spectrum.

The software presented here was not designed to minimize its use of computer resources which would probably have resulted in a separation of the multiple functions of this routine. Furthermore,
using a maximum sensitivity test which allowed a 10% error rate, the
resulting data was not significantly compressed. Of course, making
multiple measurements of the signal quality and not immediately testing
on them. This allowed manual inspection of the data quality and
careful adjustment of the tests which followed providing a sound study
basis for a larger detection scheme.
INITIALIZE DATA AND DO PRELIMINARY CALCULATIONS

DATA (CYSN(I), I=1,65)

1,1984,2820,2232,2912,12-1,1551,28-9,1589,232,2703
2,1321,716,797,9,266,2299,2343,2395,1816,8921,33,6,1
3,1327,2717,559,4,72,175,135,2912,1759,1952,2273,1765
4,3253,31,5,1990,2260,2336,2119,1542,2653,2839,1735,2186
5,2398,26,1,1954,21,6,345,3332,2171,0,0

DATA (CFST(I), I=1,65)

1,524,52,58,5,676,497,51,496,502,533,599,6,12,13,483,497,6814
2,82,49,475,499,499,464,497,438,513,521,498,5,39,533,52,2
3,53,513,35,467,499,491,515,477,523,519,478,4,2,39,513,342,0

DATA (CM(I), I=1,21)+0,723932E+02,-0,1731253E+03,0,1507014E+03,
1,16.7,3.73476E+2,0,8152275E+01,-0,116358E+32,0,1525229E+02,-5.439
2347E+01,1,3,676524E+06,0,215142E+01,0,1236007E+02,0,2436290E+0
3,24,23,245,81,4,1331119E+02,0,265275E+01,-0,244939E+02,0,4
1,57265E+03,-0,2140656E+03,0,17972825E+03,-0,88,7811E+02,0,15625
5E+027

ORIGINAL PAGE IS OF POOR QUALITY
DTA=MAX(J)=-.709;
KR(J)=K(J)=0.
ON=Q(J)=3.
PR=CONT(J)=0.
COUNT(J)=0.
KPRIM(J)=0.
YNT(J)=0.
YN*1(J)=1.
YN*2(J)=1.
KPRF(J)=0.
KSTART(J)=.
MARKER(J)=.TRUE.
XNT(J)=X*NT(J)=X*N2(J)=L.
XAR(J)=1.
PI=3.1415926535.
SADD=350.
FRQA=1.0.
FRGB=5.0.
ALPHA=2.*PI*.FRQA.
RETA=2.*PI*.FRGB.
TAN=TAH(TETA/2.,*SMPF).
TANH=TAN(RETA/2.*SMPR).
OPA=1.+TANA.
QMA=1.-TANA.
OP3=1.+TANB.
OMB=1.-TANB.
DO 9=1,3.
10 CONTINUE
BEGIN=TRUE.
END=1.0.
IF 9=1,2,3 GO TO 10.
CALL AXIS (2.0,0,0,7,HAZMUTH,-7,XM1X,0.0,FSTAZ,DX,10.9).
CALL AXIS (2.0,0,0,8,HZMITHF,7,3,0,31.2,-0X,20.0).
CALL AXIS (1.0,3.0,12,HSIGNAL SCALE,12,5.0,90.0,-2.*DY,DY,20.0).
CALL PLOT (2.0,1.0,0,0,0,0).
DO 11 J=1,NSYM.
YSYM=0.
XSYM=XSYM+2.
DO 11 K=1,18.
YSYM=YSYM+0.5.
11 CALL SYMBOL (XSYM,YSYM,SYMHT,3.0,0,-1).
TL=.5=0.5-DY.
YMOST=9.5*DY.
-3 CONTINUE.
PROGRAM MINISP 4/7, OPT=3

DO J=1,15,
XN(J)=XN1(J)=XN2(J)=0.
Y(J)=0.
YN1(J)=0.
YN2(J)=0.
CONTINUE

DO 22 J=1,15,
DO 22 K=1,39
AMP(J,K)=SC(J,K)=0.5*(J,K)=39.
SAS(J,K)=SAS(J,K)=39.

22 CTG(J,K)=DNX(J,K)=0.

READ IN A PHYSICAL RECORD, E = BY 33 ELEMENTS. (12 LINES EACH)

CONTINUE

CTIM=SECOND,A)

IF (TJX3-CLIM,LE,0.) GO TO 999
PHR=U=MKEL=-1.
BUFF=. IN - T1(I) (IA(I),IA(J))

CALL PECALL

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE
SUBJ=J
SUNSQ=DNX*X(J)
DNY*N(J)=0.
QG(1,E,K)=1
FV(J,K)=CNFST(J)
IF (ILVL.LT.5) ILVL=5
IF (ILVL.EQ.5) ILVL=ILVL+1
IF (ILVL.LE.9) QVX(J,K)=-32*(HLVL)
IF (ILVL.GT.0) CNY(J,K)=CNF(NLVL)

IF (ICCL0R.EQ.1) GO TO 63
IF (NLVL.GE.0) NLVL=NLVL+1
?ERR=((97-ILVL+1)-CNF(ILVL-1))/2.)*ERR
NLVL(J)=NLVL(J)+CNF(J)
CONTINUE

IF (ICCL0R.EQ.1) GO TO 15
DO 33 K=1,39
WSU(J,K)=FSU(J,K+39)
BIAS(J,K)=BIAS(J,K+39)
AM(J,K)=-1FJ(J,K+39)
SUMX=SUMXY=SUMXX=0.
DO 31 K=K,39
SUMX=SUMX+DNX(J,KL)
SUMXY=SUMXY+DNX(J,KL)*YY(K-L+1)
SUMXX=SUMXX+DNX(J,KL)*YY(K-L+1)

IF (YU(J,K+3) .LE. UY(J,K+3)) GO TO 36
RO(J,K+39) = RNUM/RNUM/(XDENOM*YDENOM)

GO TO 37

35 RSO(J,K+39) = 0.
37 IF (YDENO3 .LE. 0) GO TO 39
AM(J,K+39) = RNUM/YDENOM
GO TO 39

38 AM(J,K+39) = 0.
39 IF (YDENO3 .LE. 0) GO TO 32
BIAS(J,K+39) = (SUMYY*SUMX-SUMXY)/YDENOM
GO TO 33

42 BIAS(J,K+39) = 0.
CONTINUE
CONTINUE
IF (BEGIN) GO TO 70

C FOLLOWING IS THE DIGITAL FILTER ROUTINE

C DO 14 K=1,79
XN(J)=DTA(J,K)
YN(J)=J**1.5+DTA(J,K)+1.3*Y(NM1(J))=J**4.73544*YNM2(J)
CONTINUE
IF(ICOLOR NE 1) GO TO 70
XBAR(J) = SUM/J
DO 65 K = 1, J
DEL = OTA(J, K) - XBAR(J)
SUMSC = SUM + DEL * DEL
65 CONTINUE
SIGSQ(J) = SUMSC/J

C SET MSN EQUAL TO NOISE IF READING FIRST RECORD. OTHERWISE USE THE NOISE FORMULA
C SET THE REFERENCE LEVEL EQUAL TO THE CURRENT AVERAGE IF READING THE FIRST RECORD. OTHERWISE USE THE REFLVL FORMULA.
C
IF (BEGIN) REFLVL(J) = XBAR(J)
IF (BEGIN) MSN(J) = SIGSQ(J)
IF (((SIGSQ(J) - 3.0 * MSN(J)) GT 0.0) OR (.NOT. MARKER(J))) GO TO 67
MSN(J) = 6.0 * MSN(J) + 1.0 * SIGSQ(J)
67 CONTINUE

C SEARCH FOR RAPID RISE AND SOURCE SIGNALS IN THE CURRENT BLOCK OF DATA. DISCARD ALL SIGNALS WHICH MEET THE RRD TEST.
C
RMS(J) = 3.0 * SORT(MSN(J))

C
DO 31 K = 1, J
IF (MARKER(J)) GO TO 99
IF (IOTA(J, K) .LT. UMAJX(J)) KP = K(J) = K
OMAX(J) = OMAJX(J, K), OMAJX(J)
IF (IOTA(J, K) .LT. REFLVL(J) - RMS(J)) LT, J(J) = K(J)
KPIME(J) = K(J)
GO TO 101
99 IF (OMAX(J, K) .LT. REFLVL(J) - RMS(J)) LT, J(J) GO TO 100
GO TO 101

100 MARKER(J) = .FALSE.
KSPB = K + 28
KSPR = K + 33
DO 35 I = KSRB, KSEP
IF (ISQ(J, I) LT RMAX(J)) GO TO 34
K(R(J)) = I - 28
34 CONTINUE

I(AV(J, J) LT AYAX(J)) GO TO 35
KAI(J) = I - 28
AOMAX(J) = AAV(J, J)
BIMAX(J) = BIAS(J, J)
35 CONTINUE
AV = VAL(J) = REFLVL(J)
ZNO(J) = PS(J)
**Code Snippet**

```
KSTAR(J)=K
DTAMAX(J)=DTA(J,K)
KP:AK(J)=K
BGNTIM(J)=TA+(KSTAF(J)+J/70.)/350.

CONTINUE
IF(KPRAKE(J),CE,J) GO TO 90
IF(KPRAKE(J),CE,J) GO TO 100
IF(KPRAKE(J),CE,J) GO TO 102
IF(KPRAKE(J),CE,J) GO TO 125
IF(KPRAKE(J),CE,J) GO TO 192
1 GO TO 192

CONTINUE
PKTI(K,J)=TA+(KSTAF(J)+J/70.)/350.
RPA:CT(J)=TCPNT(J)+1
NRJCT=NRJCT+1
WRITE(327,HREC,J,KSTAR(J),KPEAK(J),KPRIME(J),DTAMAX(J),BGNTIM(J)
1,PKTI(J),AVRVAL(J),ZNO(J),NRJCT,RPONT(J),KJ(J),ANAX(J),KPR(J)
2RMAX(J),LIPAX(J)
2 FORMAT(515,5E12.5,215,2(I5,E12.5),12.5)
GO TO 103

C
IF A SOURCE HAS BEEN FOUND WHICH EXCEEDS THE REFERENCE LEVEL BY
THREE SIGMA, CALCULATE THE BEGINNING TIME AND THE TIME OF PEAK,
AND OUTPUT THE STAR DATA TO THE RECORDING FILE.
C

CONTINUE
PKTI(J)=TA+(KSTAF(J)+J/70.)/350.
COUNT(J)=COUNT(J)+1
MSTAR=MSTAR+1
WRITE(327,HREC,J,KSTAR(J),KPEAK(J),KPRIME(J),DTAMAX(J),BGNTIM(J)
1,PKTI(J),AVRVAL(J),ZNO(J),NRJCT,RPONT(J),KJ(J),ANAX(J),KPR(J)
2RMAX(J),LIPAX(J)
2 FORMAT(515,5E12.5,215,2(I5,E12.5),12.5)
GO TO 103

C
IF KJ(J) COINCIDES WITH A PREVIOUS MAKER(J), MARK THE PEAK,
AND OUTPUT THE STAR DATA TO THE RECORDING FILE.
C

CONTINUE
MSTAR=MSTAR+1
WRITE(327,HREC,J,KSTAR(J),KPEAK(J),KPRIME(J),DTAMAX(J),BGNTIM(J)
1,PKTI(J),AVRVAL(J),ZNO(J),NRJCT,RPONT(J),KJ(J),ANAX(J),KPR(J)
2RMAX(J),LIPAX(J)
2 FORMAT(515,5E12.5,215,2(I5,E12.5),12.5)
GO TO 103

C
BACKGROUND DATA CHANNELS, IN MV(J,K)
C
J=25,45,65... K=1 TO 39
C
SUMK=0.
SUMBK=0.
IBK=(J=7)/19
DO 16 K=1,39
POINT=VV(J,K)/2048
VAL=POINT*3KFA(J,IBK)
TIME=TA*(KJ(J)+J)/350.
IF(J,J,EQ,28) WRITE(9) VAL, TIME
IF(J,J,EQ,35) WRITE(9) VAL, TIME
```
IF (IPLC1.EQ.0) GO TO 2002
ISTCH=(ICOLOR-1)*18+1
IF (ISTCH.GE.17) GO TO 2001
DO 2001 LL=ISTCH,1EN0CH
F=M(0,LL-1,16)+1
DO 3001 KK=1,39
L=LCHN(LL)
R=TA+(KK+L/70.)/350.
X(KK)=251.37904+1.3082-0.5*2-2.7721-3E-05*R^2 -0.6376111111111E+01+1.8757556786*(-0.136*(CHAN(L)/130)-2)+(13.3353/2.1)-((MOD((MOD(CHAN(L),130)),2))^6,0.3353))^-0.18=-FSTAY
Y(KK)=(FSTAY*DOT+DTAIL,KK)
IF (Y(KK).LT.YLEAST) Y(KK)=YLEAST
IF (Y(KK).GT.YMOST) Y(KK)=YMOST
3001 CONTINUE
CALL LINE (X,Y,39,1,0,1,0,0,1,0,0,0,35)
2001 CONTINUE
2002 CONTINUE
IF (ICCOLOR.NE.1) GO TO 5
WRITE(*,1001) ((BKSN(K),RKSIG(K)),K=1,3),PHREC
1001 FORMAT(1X,IPSL,-)=X,1PE,15.),,3X,15)
IF ((MCO(PHREC,NU)).NE.0) GO TO 5
DO 50 IC=1,55
DEL=PHS(I)*RMS(I)-SYSN(I)
DEL=L-DEL, D=L=0.
50 QMS(I)=SORT(DEL)
WRITE(*,1002) PHREC,TA,(CHAN(I),RMS(I),QMS(I),I=6,25),(CHAN(I),RMS(I),QMS(I),I=6,33)
WRITE(*,1002) PHREC,TA,(CHAN(I),QMS(I),I=6,25),(CHAN(I),RMS(I),QMS(I),I=6,33)
1002 FORMAT(1H9,4X,14,4X,1PE,2,1D(3(* 1 *),13,-1X,-10),,3X,10,-1),/19X 1)
GO TO 5
70 WRITE(*,1003) PHREC
1003 FORMAT(1H9,4X,*PARITY ERROR ON RECORD *,I19,* CONTINUE READING*/1)
GO TO 5
999 ILSTP=PHREC
WRITE(*,1004) PHREC,TA
1004 FORMAT(1H9,5X,*TIME MAXIMUM REACHED AT RECORD */,I10,5X,* FINAL TI
THE IS */10,4,///)
20 IF (ICCOLOR.NE.1) GO TO 18.
ENDFILE,REWIND 4
POINT 5.
IF(EOF(4)) STOP

FILE 7

READ (7,1072) PHREC,TA,(CHAN(I),RMS(I),OMS(I),I=8,23),(CHAN(I),RMS(I),OMS(I),I=27,44)
IF(EOF(7)) STOP

WRITE (*,1004)

ENDFILE 4

NOW COUNT UP THE OBJECTS DETECTED, AND THE REJECTS, AND LIST TABLES

DO 200 I=1,65
J=CHAN(I)/100
IF(J,NE.0) ICOLCNT(J)=ICOLCNT(J)+COUNT(I)
200 OBJECT=OBJECT+COUNT(I)

IF(IPLOT.NE.0) GO TO 19
WRITE(*,1009) ICOLOR,ICOLCNT(ICOLOR)
19 FORMAT(146,5X,*END OF INPUT FILE*,5X,*NO PLOTS*)
GO TO 19
17 WRITE(*,1009) ICOLOR,ICOLCNT(ICOLOR)
169 FORMAT(146,5X,*END OF INPUT FILE*,5X,*PLOT COLOR NUMBER*,13,2Y,19,1* STARS*)
18 CONTINUE
IF(ICOLCNNE.1) GO TO 202
WRITE(*,1005) FSTTIM,TA
1005 FORMAT(141,5X,*THE FIRST RECORD IS *,I16,5X,*THE LAST RECORD IS *,111*)
WRITE(*,1011) FSTTIM,TA
1011 FORMAT(5X,*THE DATA BEGINS AT T = *,F12,4,* SECONDS AND T = 2:05
1AT T = *,F12,4,* SECONDS*)
WRITE(*,1015) OBJECT,REJECT
1015 FORMAT(146,5X,*OBJECTS FOUND *,F10,5X,*SIGNALS REJECTED *)
WRITE(*,1014)
1014 FORMAT(146,5X,*CHANNEL NCHENCLATURE *,13,2X,*160 IS A SHORT WAVE
161,80TH*,7,3X,*260S ARE MEDIUM WAVELENGTH*,7,3X,*360S ARE LONG
2 WAVELENGTH*,7)
WRITE(*,1007)
1007 FORMAT(146,5X,*OBJECTS DETECTED ON EACH CHANNEL*)
WRITE(*,1008) (CHAN(I),COUNT(I),I=8,23),(CHAN(I),COUNT(I),I=27,44)
1, (CHAN(I),COUNT(I),I=46,63)
1008 FORMAT(146,5X,*SOURCES ON EACH COLOR*,7,27X,*SHORT*****,15,7,27X,*1MEDIUM*****,15,7,27X,*LONG*****,15)
C
ENDFL= 2
ENDFile 3
REWRD 2
REWIND 3
RT=2
PRINT 46
WRITE(7',ITF6)
1016 FORMAT((1HC, 56X, "STARS DETECTED", 1/))
1021 WRITE(7',ITF2)
1021 FC=MAT(IX,*,NUMBEr_CHAN_ST PK E A R LNH TIME AMPL H)
1021 WRITE(*,ITF2)
1021 WRITE(*,ITF2)
500 READ(KT,2) IREG,IND,ISAMP,PEAK,1E-6,PK,MEAN,NOISE,ISTL, ICIC, KACH,ACH,KRCH,ACH,SHIFT
1021 IF (C0F(IT)) =3C2,3C1
301 HT=PEAK-MEAN
301 IF(NICE.EQ.0.0.) GO TO 12
301 SN=P*HT/NOISE
301 GO TO 13
12 SN=0.
13 CONTINUE
301 SN=SN*FACTUAL(INW)
301 ZRS=(PEAK-ISAMP)*1.30814/359.
301 ZE=31.53976929+9.67935530+7.0(+130TCHAN(IND),1J))-9.723
301 ZE=Z0*PI/180.
301 PK=PK
301 PK=PK-PEAK+360.
301 PK=PK+360.
301 COSZ=COS(ZERAD)
301 SINZ=SIN(ZERAD)
301 COSA=COS(AZRA)
301 SINH=SIN(AZRAD)
301 SINDEC=COSZ*SIDECP+SNZ*COECP*COSAZ
301 DEC=ASIN(SINDEC)
301 COSDEC=SNRT(1.-SINDC*SINDC)
301 SINDC=SINDC/SINDEC
301 COSHA=(COSZ*SIDEC**SINDC)/(COECP*COSDEC)
301 HATAN2(SINDC,COSHA)
301 RA=HA
301 DECDEC=DEC*180./PI
301 WRITF((6,1022) ISTTL,ISTCH,CHAN(IND),ISAMP,PEAK,1E-6,PK,RAMS,DECDEC
1021 1 FORMAT((1X,14,7(1X,13)),1X,F5.3,1X,F5.3,1X,F5.2,1X,F7.2,2(1X,F6.2),1X,F7.
1021 1 FORMAT((1X,14,7(1X,13)),1X,F5.3,1X,F5.3,1X,F5.2,1X,F7.2,2(1X,F6.2),1X,F7.
301 1 HTIF=-(1.022) ISTTL,ISTCH,CHAN(IND),ISAMP,PEAK,1E-6,PK,RAMS,DECDEC
301 1 FORMAT((1X,14,7(1X,13)),1X,F5.3,1X,F5.3,1X,F5.2,1X,F7.2,2(1X,F6.2),1X,F7.
301 1 GO TO 300
301 1
**Program History**

```
PROGRAM THIST

92 IF(IT,IT<0.3) GO TO 92
PRINT 1
WRITE ('(A,A)'), IT
917 FCYMAT(INL,'X',STARS,REJECTED,/) NT=3

402 CONTINUE
END FILE=4
REWIND 4
IF(IFLOT .GT. 0.5) GO TO 405
ICOLOR=ICOLOR+1
IF(ICCCLC .GT. 3) GO TO 6
XH=XH/X+10.
GO 3 I=I,IFACE
BACKSPACE 1
CALL PLOT(X:EXPL,1,1,3)
IF(ICCORDER.EQ.1) DX=NX*2.5
GO TO 4 CONTINUE
END FILE=6
IF(IFLOT.EQ.0) GO TO 777
CALL ENDFILE
777 CALL END
STOP
END
```

**Error Details Diagnosis of Problem**

1. **560** THE RECORD LENGTH IS GREATER THAN 137 CHARACTERS. IT MAY EXCEED.

2. **IVAL** ARRAY NAME OPERAND NOT SUBSCRIPTED, FIRST ELEMENT WILL BE USED.

**Actual Reference Map (R=3)**

**NTS DEFS LINE REFERENCES**

<table>
<thead>
<tr>
<th>SN</th>
<th>TYPE</th>
<th>&quot;LOCATION&quot;</th>
<th>REFS</th>
<th>187</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>REAL</td>
<td></td>
<td></td>
<td>563</td>
</tr>
<tr>
<td>2</td>
<td>REAL</td>
<td></td>
<td></td>
<td>557</td>
</tr>
<tr>
<td>3</td>
<td>REAL</td>
<td></td>
<td></td>
<td>335</td>
</tr>
<tr>
<td>4</td>
<td>REAL</td>
<td></td>
<td>375</td>
<td>336</td>
</tr>
<tr>
<td>5</td>
<td>REAL</td>
<td></td>
<td>270</td>
<td>272</td>
</tr>
<tr>
<td>6</td>
<td>REAL</td>
<td></td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>REAL</td>
<td></td>
<td>1</td>
<td>351</td>
</tr>
<tr>
<td>8</td>
<td>REAL</td>
<td></td>
<td>552</td>
<td>357</td>
</tr>
<tr>
<td>9</td>
<td>REAL</td>
<td></td>
<td>21</td>
<td>307</td>
</tr>
<tr>
<td>10</td>
<td>REAL</td>
<td></td>
<td>22</td>
<td>279</td>
</tr>
<tr>
<td>11</td>
<td>REAL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Muster Chart:**

1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
4.0 APPENDIX: A SIGNAL PROCESSING GAME

The aim of this game is to develop skills in signal processing. The input data for this game are the recorded data \( U_R(t) \). It is assumed that the non-uniform scanning velocity has been corrected for already. The time coordinate is given in discrete numbers \( t = 0, 1, 2...63 \). We may consider \( U_R(t) \) as being about one quarter of a single horizontal scan (\( \beta = \text{constants} \)).

The rules of the games are as follows. The "investigator" gets the sheet "Recorded Data \( U_R(t) \)" and the sheet "Problem #1." After solving this problem he will give the solution to the "monitor" and to the "game constructor." Now he may start on problem #2, and so on. But it is important that the investigator does not get the next problem sheet before he has finished the previous problem. The reason is that the formulation of the later problems contains parts of the answers to the earlier problems. This has to do with the basic structure of this simulation game: for performing any meaningful signal processing operation one must have some knowledge about the original signal and/or the noise. For example, in problem #1 the investigator is told that the noise is additive and non-negative. In the later problems, the investigator will be supplied with even more a priori information. Naturally, this should enable him to extract the signals better and better. But the methods for doing this increase in complexity.

On the very last pages, following the problems, the design of the "recorded data" is explained, and the true original signal is unveiled. Obviously those pages should not be given to the investigator before he has solved all the problems.
**RECORDED DATA $U_R(t)$**

$t$ is the discrete time variable running from $t=0$ to $t=63$.

<table>
<thead>
<tr>
<th>$t$</th>
<th>$U_R(t)$</th>
<th>$t$</th>
<th>$U_R(t)$</th>
<th>$t$</th>
<th>$U_R(t)$</th>
<th>$t$</th>
<th>$U_R(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>55</td>
<td>16</td>
<td>110</td>
<td>32</td>
<td>39</td>
<td>48</td>
<td>383</td>
</tr>
<tr>
<td>1</td>
<td>25</td>
<td>17</td>
<td>184</td>
<td>33</td>
<td>56</td>
<td>49</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>85</td>
<td>18</td>
<td>29</td>
<td>34</td>
<td>05</td>
<td>50</td>
<td>69</td>
</tr>
<tr>
<td>3</td>
<td>61</td>
<td>19</td>
<td>51</td>
<td>35</td>
<td>15</td>
<td>51</td>
<td>58</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>20</td>
<td>42</td>
<td>36</td>
<td>95</td>
<td>52</td>
<td>52</td>
</tr>
<tr>
<td>5</td>
<td>95</td>
<td>21</td>
<td>78</td>
<td>37</td>
<td>09</td>
<td>53</td>
<td>66</td>
</tr>
<tr>
<td>6</td>
<td>07</td>
<td>22</td>
<td>09</td>
<td>38</td>
<td>81</td>
<td>54</td>
<td>79</td>
</tr>
<tr>
<td>7</td>
<td>00</td>
<td>23</td>
<td>15</td>
<td>39</td>
<td>21</td>
<td>55</td>
<td>134</td>
</tr>
<tr>
<td>8</td>
<td>62</td>
<td>24</td>
<td>13</td>
<td>40</td>
<td>81</td>
<td>56</td>
<td>94</td>
</tr>
<tr>
<td>9</td>
<td>79</td>
<td>25</td>
<td>50</td>
<td>41</td>
<td>399</td>
<td>57</td>
<td>102</td>
</tr>
<tr>
<td>10</td>
<td>148</td>
<td>26</td>
<td>99</td>
<td>42</td>
<td>312</td>
<td>58</td>
<td>108</td>
</tr>
<tr>
<td>11</td>
<td>105</td>
<td>27</td>
<td>54</td>
<td>43</td>
<td>348</td>
<td>59</td>
<td>94</td>
</tr>
<tr>
<td>12</td>
<td>125</td>
<td>28</td>
<td>99</td>
<td>44</td>
<td>303</td>
<td>60</td>
<td>56</td>
</tr>
<tr>
<td>13</td>
<td>125</td>
<td>29</td>
<td>35</td>
<td>45</td>
<td>383</td>
<td>61</td>
<td>67</td>
</tr>
<tr>
<td>14</td>
<td>173</td>
<td>30</td>
<td>98</td>
<td>46</td>
<td>317</td>
<td>62</td>
<td>51</td>
</tr>
<tr>
<td>15</td>
<td>181</td>
<td>31</td>
<td>02</td>
<td>47</td>
<td>317</td>
<td>63</td>
<td>63</td>
</tr>
</tbody>
</table>
Problem #1

Given are the recorded data $U_R(t)$ with $t = 0, 1, \ldots, 63$. Wanted are the original data $U_0(t)$, which represent the "one-dimensional equivalent object radiation" $S_E(\alpha)$. We assume that the known influences of the telescope $[M(x', y'); R(x', y')]$ and of the electrical system $[G(t)]$ have been compensated already or are negligible. But the recorded signal $U_R(t)$ is corrupted by additive noise $N(t)$:

$$U_R(t) = U_0(t) + N(t)$$

The only features known about the original signal $U_0(t)$ and about the noise $N(t)$ are that they are non-negative:

$$U_0(t) \geq 0; \quad N(t) \geq 0.$$

Furthermore, the noise $N(t)$ is stationary, which means that the noise properties are not "drifting." In other words, short-term average features of the noise remain the same from the beginning to the end of the observation.

Try to utilize the given a priori information for computing a new signal $U_1(t)$ from $U_R(t)$, which somehow is better than $U_R(t)$ as an approximated representation of $U_0(t)$. Plot $U_1(t)$ as a continuous curve, and also $U_R(t)$ for comparison.
Problem #2

Given are the facts:

\[ U_0(t) \geq 0; \quad N(t) \geq 0; \quad \bar{N} = 50. \]

By \( \bar{N} \) we mean the linear average of the noise. This \( \bar{N} \) can be visualized as the dark current of the photoreceiver as measured with an instrument which rejects high frequencies.

Based on these facts, try to compute a better signal \( U_2(t) \) from \( U_R(t) \). Plot both \( U_2(t) \) and \( U_R(t) \).
Problem #3

Given are the same facts as in the previous problem. In addition, it is known that the noise is approximately "white."

\[ \mathcal{N}(t) = \bar{N} + \mathcal{n}(t); \quad \tilde{\mathcal{n}}(\nu) = \int \mathcal{n}(t) e^{-2\pi i \nu t} \, dv; \]

\[ \nu = \frac{m}{64}; \quad m = -32, -31, \ldots, -1, 0, +1, \ldots, +30, +31; \]

|\tilde{\mathcal{n}}(\nu)|^2 constant. The amount of this "constant" is not known. Try to deduce it from the recorded data \( U_R(t) \). You might have to make an intelligent guess.
Problem #4

Given are the same facts as in the previous problems, including the "constant" which describes the noise power level.

\[ |\tilde{n}(\nu)|^2 \approx \frac{16}{3} \times 10^4 \text{ in } -\frac{1}{2} \leq \nu \leq +\frac{1}{2}. \]

Now that \( |\tilde{n}(\nu)|^2 \) is known and \( U_R(\cdot) \) is computable, can you apply the Wiener-filter theory, at least in a guessed approximation? Try it and compute \( U_4(t) \). Plot \( U_4(t) \) and \( U_R(t) \). Hint: represent \( |\tilde{U}_0(\nu)|^2 \) by a gaussian function of suitable peak power and width. Signal processing specialists always try it with a gaussian function if they don't know a better way.

\[ |\tilde{U}_0(\nu)|^2 \approx p_4 e^{-\pi (\nu/v_4)^2}. \]
Problem #5

Try the same approach as in the previous problem, but with a guessed sinc^2-shaped \(|U_0(v)|^2\)

\[ |U_0(v)|^2 \approx P_5 \cdot \text{sinc}^2(v/v_5); \quad \text{sinc } z = \frac{\sin \pi z}{\pi z} \]

Plot the result \(U_5(t)\) and also \(U_R(t)\) for comparison.
Problem #6

Try the same approach as in the previous problem, but a somewhat different guess for $|\tilde{U}_0(\nu)|^2$

$$|\tilde{U}_0(\nu)|^2 = P_6 \text{sinc}^2(\nu/\nu_0) + (P_0 - P_6) \delta_0;$$

Herein $\delta_0$ means a function which is equal to 1 for $\nu \neq 0$ and equal to 0 for $\nu \neq 0$. Plot $U_6(t)$ and $U_R(t)$. 
Problem #7

Based on all of the accumulated experience, try your own signal processing approach or simply guess what $U_0(t)$ might have been. Call it $U_7(t)$. Plot $U_7(t)$ and $U_R(t)$. 