AN ABSTRACT OF THE THESIS OF

JEROME ALAN ZEBROWKSI for the degree of MASTER OF SCIENCE

in Chemistry presented on May 11, 1981

Title: EXPERIMENTAL ASPECTS OF OPTICAL THREE WAVE MIXING SPECTROSCOPY

Abstract approved: Redacted for Privacy

Professor J. W. Nibler

Experimental aspects of Optical Three Wave Mixing Spectroscopy are considered with emphasis on the identification and reduction of noise sources to improve detection sensitivity. Raman (ω_{Ram}) and two photon (ω_{TP}) resonances in the third order susceptibility ($\chi^{(3)}_{TOT}$) are examined by the Coherent Anti-Stokes Raman Scattering (CARS) and two photon-three wave mixing processes which generate new coherent signals at $\omega_{TW} = \omega_1 + \omega_2 \pm \omega_3$. Fundamental theory is outlined in terms of the resonant (χ_{res}) and nonresonant (χ_{NR}) contributions to $\chi^{(3)}_{TOT}$, and the traditional expression for the power of the new three wave signal, P_{TW}, is expanded to include the experimental dependence on time and spatial overlap.

Digital signal averaging techniques are developed and employed in the examination of the highly nonlinear behavior of the resonant and nonresonant mixing processes. A ratio between these processes is constructed in an experimental sample/reference scheme to aid in the isolation of noise components which contribute to the large

uncertainty in the estimation of $\overline{P}_{TW}$. Distributions for the sample populations are illustrated and suggest heavy contamination in the sample set due to the nonlinear nature of $\overline{P}_{TW}$. Results of the noise source analysis has yielded the magnitudes of the noise components in the variance expansion of $\overline{P}_{CARS}$. Robust averaging methods are also considered as a possible means of reducing the variance in the three wave signals.

The computer programs used in the recording of spectral data are discussed in terms of the opto-electronic and analog to digital conversion processes needed for data collection. Fitting routines are employed to smooth the collected data and to assign transition energies to the spectrum. The versatility and effectiveness of these programs is demonstrated by several examples in this thesis.

Two Photon Electronic Spectroscopy (TPES) is discussed and a comparison is made between conventional techniques and two photon-three wave mixing spectroscopy. The experimental feasibility of two photon-three wave processes is further developed for the case when $\omega_{TP} = \omega_1 + \omega_2$. Choice of a U.V. laser frequency for ω_1 thus can permit measurement of two-photon transitions in the vacuum U.V. Results are given for the $X^2\pi \rightarrow \tilde{A}^2\Sigma$ transitions of gas phase Nitric Oxide at 140 torr and examples of other systems for possible study are suggested.

Experimental Aspects of Optical Three Wave Mixing Spectroscopy

by

Jerome Alan Zebrowski

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Commencement June 1982

APPROVED:

Redacted for Privacy

Professor of Chemistry in charge of major

Redacted for Privacy

Head of Department (Chemistry) or Chairperson

Redacted for Privacy

Dean of Graduate School

Date thesis is presented May 11, 1981

Typed by Laina R. Hardenburger for JEROME ALAN ZEBROWSKI

TABLE OF CONTENTS

LIST OF FIGURES

EXPERIMENTAL ASPECTS OF OPTICAL THREE WAVE MIXING SPECTROSCOPY

CHAPTER 1

INTRODUCTION AND FUNDAMENTAL THEORY OF THREE WAVE MIXING

Introduction

Two of the most fundamentally important and versatile tools available to the spectroscopist of today are the laser and the mini-computer. The laser, with its intense optical attributes, has opened the doors for the development of new spectroscopic techniques, an example of which is optical three wave mixing spectroscopy. The development of high energy pulsed lasers, with peak powers in the megawatt range, has allowed for their use in the refinement of tunable dye lasers. Pulsed dye lasers of this type, with line widths on the order of .15 cm^{-1}, can then be used as tunable sources for high resolution spectroscopy. It is the employment of both frequency fixed and frequency tunable pulsed lasers which establishes the basis for the experimental techniques used in the optical three wave mixing processes described in this thesis.

Although the pulse width of these lasers is rather short (10 nsec), the experimental duty cycle is only 10 Hz, well within the time limits of mini-computers used in the laboratory today. In the past few years, the efficiency and affordability of micro processors and mini-computer systems has allowed for their employment in rapid data collection, data reduction and graphic display. The versatility of such computer systems has made them a powerful experimental tool,

readily adaptable for interface with pulsed laser experiments. Thus, the purpose of this thesis has been not only to show applications of the three wave mixing process, but to also bring the power and effectiveness of the computer to bear in the data collection and analysis.

Because the scope of this thesis is broad, dealing with several applications of the optical mixing process as well as the experimental interface and data processing systems, each aspect is best presented in a separate chapter. We start in the present chapter by providing an outline for the fundamental theory of the optical three wave mixing process in terms of the third order susceptibility tensor and the power efficiency of the generated three wave mixing beam, P_{TW}. Different types of mixing processes are distinguished according to the frequency dependence of both the sample and the optical beams used in the mixing.

Chapter two deals with signal averaging techniques and examines the individual components of the total "noise" or uncertainty in the averaging of the nonlinear three wave beam. In addition, we can utilize this analysis to improve experimental conditions and to investigate a more obscure but potentially useful method of signal averaging which arises from Robust statistical methods.

In chapter three are given the details of the data collection and processing system developed in this work. The computer interface for data collection is discussed in terms of the opto-electronic conversions, and pulse network which controls the analog to digital

conversion sequence, real time plotting and the frequency tuning of the dye laser. Further it describes the methods used to statistically reduce and smooth the stored data, to calibrate the frequencies of the spectrum and to then display the data graphically.

Finally, in chapter four we apply the results of chapters two and three to a novel resonant three wave process in which the sum of two optical frequencies allows access to an excited electronic state via a two photon transition.

A. Polarization Properties of Matter

When an atom or molecule is placed in an external electric field, $\vec{E}$, it becomes electronically polarized. The extent to which that substance is polarized is a vector quantity known as the polarization, $\vec{P}$. For isotropic media $\vec{P}$ has a direction parallel to $\vec{E}$, and a magnitude proportional to the electric field strength. This induced dipole can be expressed in terms of its electronic and nuclear components as:

$$\vec{d}_{ind} = \underset{\sim}{\alpha}_{el} \cdot \vec{E} + \underset{\sim}{\alpha}_{v} \cdot E \qquad (1.1)$$

Here, $\vec{E}$ is the applied electric field strength, $\underset{\sim}{\alpha}_{el}$ is the electronic polarizability tensor (which is associated with electron cloud distortion) and $\underset{\sim}{\alpha}_{v}$ is the vibrational polarizability tensor resulting from skeletal distortion. In terms of the dielectric susceptibility, χ, the total polarization becomes

$$\vec{P} = (\underset{\sim}{\alpha}_{el} + \underset{\sim}{\alpha}_{v}) \cdot \vec{E} = \chi \cdot E \qquad (1.2)$$

In the optical three wave mixing process, we are concerned with conditions which produce a large, induced dipole moment, when the applied fields arise from very intense electromagnetic radiation. The higher optical field strengths, presented by the laser, cause nonlinear terms of the polarization to become important. To demonstrate this effect we expand $\vec{P}$ in a power series [1]

$$\vec{P} = \chi \cdot \vec{E} + \chi^{(2)} \cdot \vec{E}_1 \cdot \vec{E}_2 + \chi^{(3)} \cdot \vec{E}_1 \cdot \vec{E}_2 \cdot \vec{E}_3 \qquad (1.3)$$

For isotropic gas or liquid phase media, it can be shown [2] that the second order susceptability, $\chi^{(2)}$ must vanish, in which case equation (1.3) reduces to

$$\vec{P} = \chi \cdot \vec{E} + \chi^{(3)} \cdot \vec{E}_1 \cdot \vec{E}_2 \cdot \vec{E}_3 \qquad (1.4)$$

Equation (1.4) shows that the nonlinear polarization for gaseous molecules, is then dependent on the third order susceptibility, $\chi^{(3)}$. This coefficient describes the interaction of three electric fields with a dielectric substance to produce a polarization effect and a three wave mixing process occurs.

In the optical three wave mixing process of this thesis, three colinear laser beams at angular frequencies, ω_1, ω_2, and ω_3, are focused into a gaseous sample. When the three electric fields are in phase, $\chi^{(3)}$ produces a coherent dipole source which generates a colinear beam at a new frequency [3]

$$\omega_{TW} = \omega_1 \pm \omega_2 \pm \omega_3 \qquad (1.5)$$

Under these conditions, the power of the three wave mixing beam can be expressed as

$$P_{TW} = k \cdot P_1 \cdot P_2 \cdot P_3 \, |\chi^{(3)}_{TOT}|^2 \qquad (1.6)$$

Here k is a proportionality constant and P_i is the laser power density at the focus. $\chi^{(3)}_{TOT}$ is taken to be a sum of contributions from all resonant and nonresonant processes.

According to equation (1.1), there are two components of the induced dipole which contribute to the dielectric susceptibility (and to P_{TW}). If the mixing process occurs in the absence of an active electronic or vibrational (rotational) resonance, the electronic part of the susceptibility would still give rise to the coupling of three optical waves (and produce a fourth at ω_{TW}) by use of virtual electronic states. The nonresonant power thereby generated at ω_{TW} is given by

$$P_{NR} = k \cdot P_1 \cdot P_2 \cdot P_3 \, |\chi_{NR}|^2 \qquad (1.7)$$

and is characterized by the nonresonant susceptibility χ_{NR}. Thus, any dielectric substance will produce a three wave mixing signal which can be detected when both field strengths ($\vec{E}_i$) and χ_{NR} become sufficiently large.

The third order, nonlinear susceptibility may become very large and an increase in P_{TW} is observed when any intermediate mixing state corresponds to a real or resonant state. The optical three wave mixing described in this thesis illustrates the enhancement of P_{TW}

by the resonant processes

$$\omega_1 \pm \omega_2 = \omega_{res} \tag{1.8}$$

For the case in which the sum of two optical frequencies, $\omega_1 + \omega_2$ match an allowed two photon electronic transition, equation (1.5) becomes

$$\omega_{TW} = \omega_1 + \omega_2 - \omega_3 = \omega_{TP} - \omega_3. \tag{1.9}$$

This optical three wave process is known as Two Photon Electronic Spectroscopy or TPES. Chapter four deals extensively with this form of the mixing process and further discussion is resumed then.

P_{TW} may also be increased when the vibrational susceptibility becomes large. This type of resonant process can occur (equation 1.8) when the difference between two optical frequencies correspond to a Raman active vibrational (rotational) resonance, ω_{Ram}, such that

$$\omega_{Ram} = \omega_1 - \omega_2 = \omega_{res} \tag{1.10}$$

Experimentally it is convenient to choose $\omega_3 = \omega_1$ so that the three wave mixing frequency of equation (1.5) becomes

$$\omega_{TW} = 2\omega_1 - \omega_2 \tag{1.11}$$

Since ω_{TW} corresponds to a higher energy anti-Stokes frequency (compared to ω_1), this three wave process is known as Coherent Anti-Stokes Raman Scattering or CARS. The transition frequency dependence

of the sample is characterized by the parameter $\chi^{(3)}_{CARS}$. It is discussed more specifically in the following section.

Overall there are several unique three wave mixing processes. Figure (1.1) depicts the four possible cases of concern in our experimental measurements. Two of these cases (c,d) involve an active resonance of the sample, which causes the induced dipole to become large, and P_{TW} is increased above the nonresonant signal level produced by processes a,b.

B. The Third Order Susceptibility

Both the resonant and nonresonant three wave processes are collectively described by the total third order susceptibility, $\chi^{(3)}_{TOT}$. This quantity is a bulk susceptibility which includes resonant and nonresonant contributions from all solvent and solute molecules present in the sampled volume. For the single, pure compounds used in this work, χ is simpler but the resonant part of $\chi^{(3)}_{TOT}$ still involves a summation to include the contributions of all the molecular transitions in the region near $\omega_1 - \omega_2$. Usually only a single resonance is important, in which case, $\chi^{(3)}_{TOT}$ can be written as

$$\chi^{(3)}_{TOT} = \chi_{NR} + \chi'_{CARS} + i\,\chi''_{CARS} \qquad (1.12)$$

Here χ'_{CARS} is the real part of the frequency dependent susceptibility and χ''_{CARS} is the imaginary part. When derived [4] in terms of the Raman cross section, the frequency dependent CARS susceptibility can be written [1] as

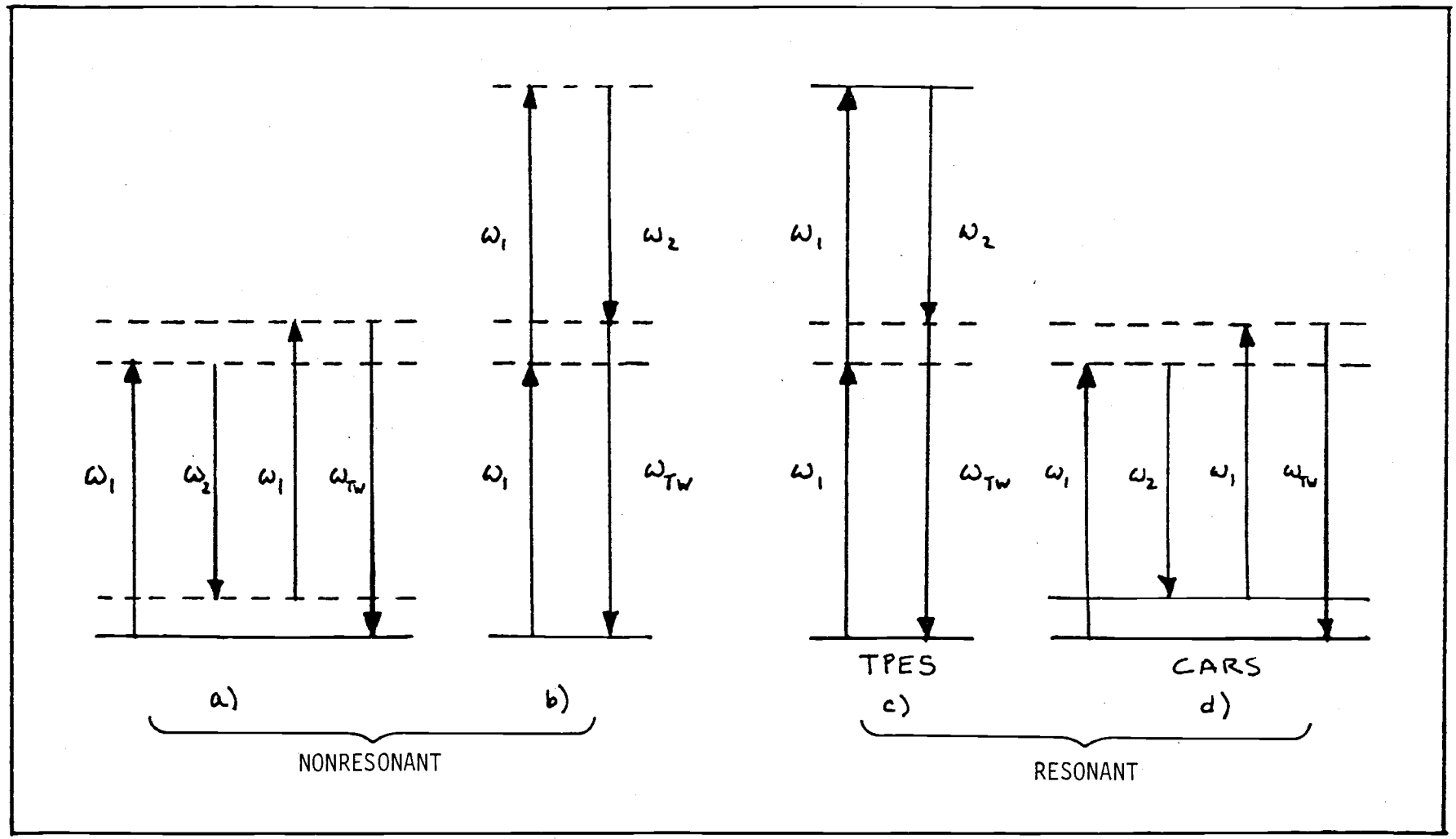


Figure 1.1. Energy level diagrams showing resonant and nonresonant processes which contribute to the intensity at ω_{TW}.

$$\chi^{(3)}_{CARS} = \frac{2NC^4}{h\omega_2^4}\left(\frac{d\sigma}{d\Omega}\right)\frac{\omega_{Ram}}{\omega_{Ram}^2 - (\omega_1-\omega_2)^2 - i\,\Gamma_{Ram}(\omega_1-\omega_2)} \qquad (1.13)$$

Here N is the number density, ω_{Ram} is the Raman transition frequency and Γ is the transition line width.

According to equation (1.6) it is the square of the modulus of $\chi^{(3)}_{TOT}$ which determines the signal strength of the generated three wave beam. Since the electronic χ_{NR} term is usually real, the square of equation (1.12) yields

$$|\chi|^2 = \chi^2_{NR} + 2\chi_{NR}\chi'_{CARS} + \chi'^2_{CARS} + \chi''^2_{CARS} \qquad (1.14)$$

Each term in equation (1.14) contributes to the frequency dependent, line shape structure of the anti-Stokes signal and each is proportional to the square of the number density for a pure compound. We observe that the frequency independent nonresonant component interferes in the determination of the absolute CARS amplitude but fortunately χ_{NR} is small compared to χ_{CARS} in the region of a resonance. Such is the case of the pure resonant samples examined in this work. However, for the ratioing experiments described below, we have used some nonresonant reference samples (e.g. glass, argon, nitrogen) for which $\chi_{NR}(ref)$ is the only important term in equation (1.14).

C. Ratioing Experiments

To this point we have shown a power equation (1.6) for the CARS process which reveals the high nonlinearity of the experiment. The

CARS signal is very noisy because of shot-to-shot fluctuations in the time, spatial and frequency distributions of the laser pulses. Experimentally we define a general expression for the energy detected at ω_{TW} by introducing a spatial and time dependence as well as a detection efficiency parameter. Assuming $\omega_3 = \omega_1$, equation (1.6) for a resonant sample then becomes

$$P_{CARS} = k \cdot P_1^2 \cdot P_2 \cdot T(t) \cdot S_s(q) \cdot D_s(n) \cdot |\chi^3_{TOT}|^2 \quad (1.15)$$

where $T(t)$ is a time overlap function, $S_s(q)$ is the spatial overlap distribution function for the laser pulses and $D_s(n)$ is the total detection efficiency including n components.

For an experimental reference leg containing a nonresonant sample the power generated at ω_{TW} is

$$P_{ref} = k \cdot P_1^2 \cdot P_2 \cdot T(t) \cdot S_r(q) \cdot D_r(n) \cdot |\chi_{NR}(ref)|^2 \quad (1.16)$$

Thus, we can construct a ratio between the reference and sample signals such that

$$P_{ratio} = P_{CARS}/P_{ref} = \frac{S_s(q)}{S_r(q)} \cdot \frac{D_s(n)}{D_r(n)} \cdot \frac{|\chi^3(\omega)_{TOT}|^2}{|\chi_{NR}|^2} \quad (1.17)$$

The advantages of performing this type of ratio becomes clear when it is noticed that the power dependence, P_1 and P_2 and the time dependence $T(t)$ are eliminated in the ratio. Therefore, by employing this ratio we can hope to improve the experimental determination of

the relative CARS resonance amplitudes over the frequency range defined by the spectrum. Chapter two of this thesis is concerned with this ratioing and also with the noise contributions of the various parts of equation (1.16).

CHAPTER TWO

SIGNAL AVERAGING TECHNIQUES FOR NONLINEAR OPTICAL PROCESSES

A. Introduction

In this chapter, we shall examine the efficiency and reliability of the techniques used to determine the frequency dependent, nonlinear polarization of a gaseous molecule. This determination involves the digital time averaging of an analog signal which is proportional to the three wave mixing beam, P_{CARS}. Section B provides a description of the experimental apparatus used in this process. Due to the nonlinear nature of the experiment, standard deviations are often as high as 60% [5]. Thus our major objective is to examine statistical aspects which will improve our estimation of $\overline{P}_{TW}(\omega)$ by reducing the experimental uncertainty or "noise".

We can minimize the experimental uncertainty in two ways. First, by analyzing the experimental parameters which contribute to the total noise, we gain insight into conditions which optimize the experimental reproducibility. Secondly, we can examine the structure of experimental sample distributions, thereby providing a means to seek out a more effective signal averaging technique. For the first case, a detailed analysis of the noise components is provided by the use of the computer program TWEEK (Appendix III). In this analysis, resonant and nonresonant three wave mixings beam are generated by the CARS process in a sample and reference leg respectively. Parameters which can be measured directly (e.g. P_1 or P_2) are averaged over N shots

and the standard deviation is taken to represent the magnitude of that noise component. For those parameters which cannot be measured directly (e.g. spatial jitter S(q)), a resonant to nonresonant ratio is constructed to aid in the isolation and subsequent determination of the noise magnitude. Statistical operators which govern the procedure for these measurements are outlined in Section C. The results of the noise source analysis are presented in Section D.

The second optimization in the measurement of $\bar{P}_{TW}$ or $\bar{P}_{CARS}$ can be arrived at by examining the behavior of the sample distribution function when acted upon by various signal averaging operators. Such an analysis involves the classical average operator, which produces the usual mean of N data points, and the employment of two averaging operators which have the ability to change the sample population set. This alteration process is dependent on an operator judgment technique which may exclude the outlying members of a data set. Operators of this type arise from the less common procedures of Robust statistics. Although these statistical methods are difficult to justify in a quantitative sense, they have found widespread application in the statistical treatment of contaminated data sets [6]. The effects of Robust operators on a typical experimental distribution are discussed in the final section of this chapter.

B. Experimental

Experimentally, the CARS process involves the mixing of three photons within a dielectric substance at a well defined spatial

element (the focal point). These photons arrive at the focal point in pulses ($\sim 10^{18}$ photon/pulse) generated by two colinear laser beams $P_1(\omega_1)$ and $P_2(\omega_2)$. Normally, $P_1(\omega_1)$ is fixed at a given frequency, and the Stokes beam, $P_2(\omega_2)$ is scanned across a given frequency range which produces a change in the anti-Stokes beam. This change in the CARS beam is then monitored and a spectrum of $\omega_1 - \omega_2$ is recorded. Thus, the experimental CARS process has three requirements:

1. Two laser beams must be generated at frequencies ω_1 and ω_2, and ω_2 must be tunable.

2. These two beams must be coincident in time, be combined in a colinear, coherent fashion and be focused into the same time and spatial element.

3. The generated anti-Stokes beam must be isolated from the other colinear beams for measurement by an optical detector.

The experimental conditions for the CARS process used in the analysis of the noise components of S_{CARS} are now described in terms of a resonant sample leg, nonresonant reference leg and the ratio between these two processes. Since in these experiments we are only interested in measuring a peak resonant signal and not in obtaining a CARS spectrum (at the present time) both $P_1(\omega_1)$ and $P_2(\omega_2)$ are frequency fixed. We shall generate $P_1(\omega_1)$ at 532 nm (18796 cm^{-1}) in the green and determine the frequency of $P_2(\omega_2)$ according to the peak resonant frequency of the CARS sample. We have chosen Nitric Oxide (NO) as the resonance substance in the sample leg. The Raman

frequency shift for the Q branch ($\Delta J=0$) of NO occurs at 1877 cm^{-1} [7], which requires the frequency at $P_2(\omega_2)$ to be 591 nm (16913 cm^{-1} in the red). The anti-Stokes beam is then generated at 484 nm (20667 c (20667 cm^{-1}) in the blue.

1. Combining $P_1(\omega_1)$ and $P_2(\omega_2)$ in the Resonant and Nonresonant Legs.

A Quanta-Ray Nd:YAG laser is pulsed at 10 HZ. The IR (1064 nm) output is doubled using a KD*P crystal to produce the second harmonic at 532 nm (green). The first and second harmonic, are summed to generate the third harmonic at 355 nm (UV), a rhombic prism is used to disperse these three frequencies in a 90° left hand bend. When not used, the invisible leftover 1064 nm light is promptly dumped for safety reasons. The UV (~30 mj) is expanded through a scope to match the size of the dye laser cells (~1 cm). 33% of the UV output is split off using a coated reflector and focused through a cyclindrical lens to transversely pump a dye laser oscillator cell. The remainder of the UV is used to transversely pump a dye laser amplifier.

The green beam is also expanded (~X2) and is optically delayed relative to the pulse output of the dye laser to assure proper time overlap. The average energy of the green beam, measured after the delay (and scope) is about 50 mj/pulse and it has a pulse width of ~10 nsec. A dichroic beam splitter, designed to reflect green and pass red at 45°, is used to combine the red and green beams in a colinear manner just before the sample cell. An adjustable 6"

focal length lens is used to focus both beams through a pinhole (made by the green) positioned on the optical axis. This assures maximum spatial overlap. During the experiment, a 1" x 6" gas cell is centered about the focus. The static cell pressure is controlled by a roughing pump and is monitored by a Teledyne 1000 torr pressure head connected to a digital voltmeter. Just after the cell, an 8" focal length lens is used to collimate the mixing beams. The anti-Stokes beam (484 nm) is isolated by a dichroic which reflects both the red (591 nm) and green (532 nm) beams into the reference leg. Background along the optical axis is further reduced by a Corning 7-60 filter designed to transmit 70% of the anti-Stokes beam. The small trace of green leftover after the filter is finally separated by a dispersing grating which sends the anti-Stokes beam into a McPherson double pass monochromator. The optical signal is collected by an RCA 31034 A photomultiplier tube operated at 1750 V.

The reference (nonresonant) leg is established by a second dichroic (similar to the one used in the sample leg) which reflects the mixing beams in a direction parallel to the propagation of the anti-Stokes beam. A filter is used to take out any blue light present in the reference leg before the focus. The green and red are focused by a 10" focal length lens through a pinhole defined by the green focus to assure good spatial overlap in the reference leg. A glass cell window (1" x 1/16") is used as the nonresonant substance and is placed just behind the focal point. The mixing beams are then collimated using an 8" exit lens, dispersed on a grating (similar to

the sample leg grating) and collected by an IP28 photomultiplier tube at 700 V. Several blue filters (7-59, 7-60, 7-61) are used to reduce the green and red light levels at the photomultiplier tube entrance. Both the sample and reference photomultiplier tubes produce a negative going output pulse. Further details are given by Guthals [8]. The schematic representation of this process is given in Figure 2.1.

2. The Dye Laser

$P_2(\omega_2)$ is generated by a UV pumped, grazing incidence tunable dye laser whose lasing range can be extended across the entire visible region by selecting the appropriate dye [9,10]. Although it has been designed with a sine (wavelength) drive mechanism, over small tuning intervals it is stepped in approximately linear .02 cm^{-1} (wavenumber) intervals by an Aerotech stepper motor unit.

The dye laser system consists of two components: the oscillator which creates the standing wave, and the amplifier.

1. The oscillator dye cells are transversely pumped with 10 mj of UV (355 nm) generated by the Nd:YAG laser. The pumping beam is cylindrically focused into the dye cell at 4" above the table plane and penetrates into the dye ~2 mm. The emission along the optical axis is expanded by a series of two 90° prisms onto a 2400 line/mm reflection grating. The expansion factor is chosen so that the grazing incidence grating is completely illuminated when the grazing angle is 8-9° (relative to the incident axis). The first order diffracted beam is plane polarized and is reflected back toward the

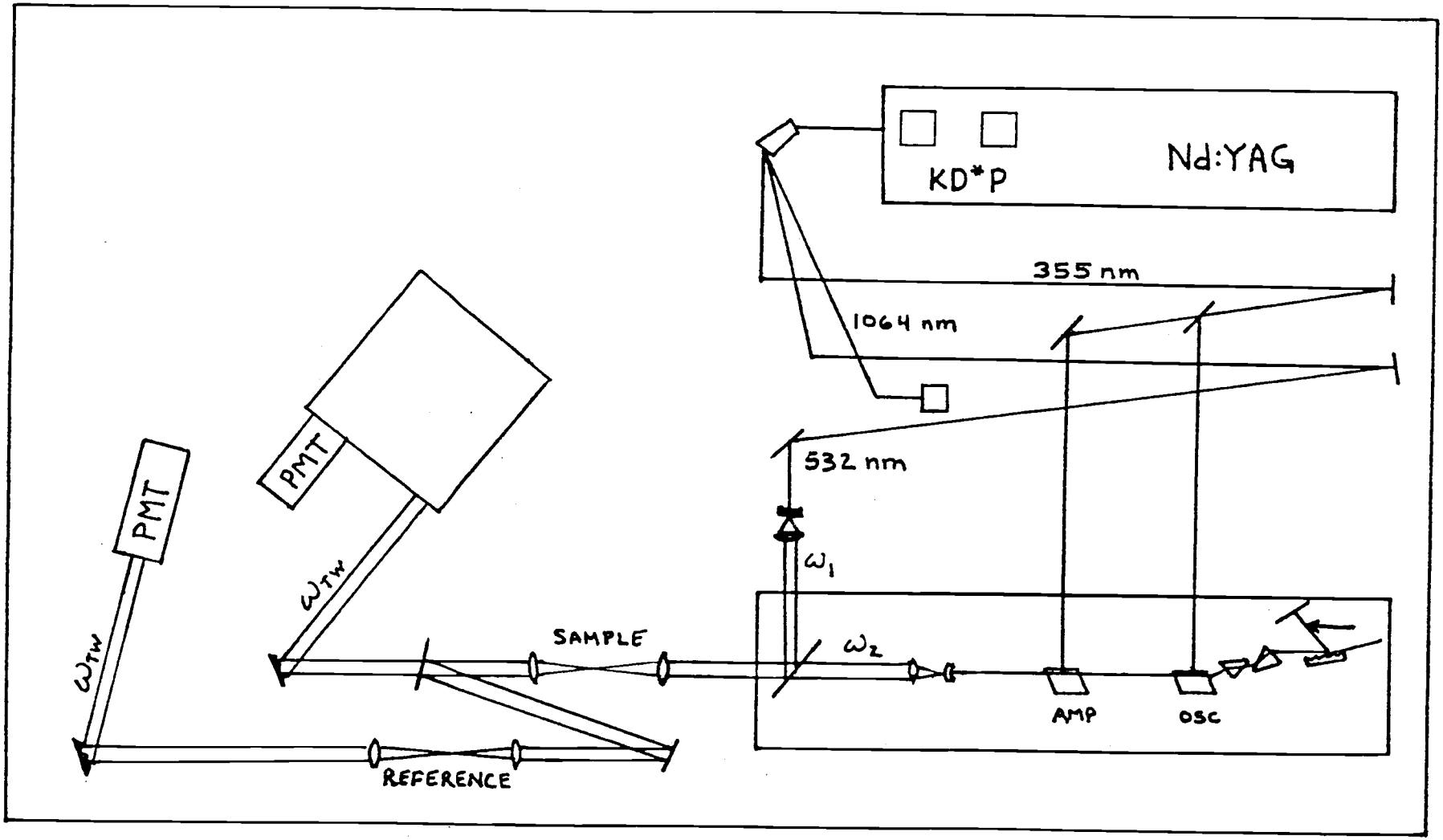


Figure 2.1. Experimental apparatus used in the resonant/nonresonant ratio.

grating by a 4" x ½" front surface mirror which is moved for tuning. In the present experiment on NO, the dye cell contains a 1:1 mixture of Rhodamine 610-640 dye dissolved in ETOH/H_2O in a 3:2 ratio to make a total concentration of ~2.5 x 10^{-3} M.

2. Amplification of the oscillator output is accomplished by transversely pumping a dye cell with ~20 mj of UV in the same manner as the oscillator. The amplifier dye may be varied for optimum efficiency, but for our experiments it is usually the same as the dye mixture for the oscillator.

C. Classical Statistics

Consider an infinite number of independent measurements occurring in the Y dimension (e.g. the power level in P_1). Since each event is independent, we should not expect consecutive measurements to yield the same value. The infinite set of all possible values, $\{Y_i\}$, is known as the parent population. This set is characterized in terms of its measurement parameters by a probability distribution function $Pr(Y_i,\mu,\sigma)$ where μ is the mean and σ is proportional to the characteristic width of the distribution (FWHM).

Experimentally, we cannot sample the entire parent population. Rather, we define a finite subset of this population which represents the parent population. This finite set of possible magnitudes, $\{X_i\}$ is the sample population and is described by the sample distribution function $P(X_i,\bar{X},S)$. Since we do not know the exact nature of the

distribution function we shall assume that each of the N elements in $\{X_i\}$ have an equal probability of occurrence. The expression for the experimental average (mean) of a sample population is then

$$\bar{X} = \frac{1}{N} \sum_{i}^{N} X_i \tag{2.1}$$

Increasing N to N' elements (such that N'>>N) offers an experimental approximation to the true average $\bar{X}(N')$, since in the limit $\bar{X}$ must approach μ.

The second moment about a distribution may also be defined and is taken to represent the uncertainty in the determination of the average (μ) of a population. This second moment is known as the variance, σ^2, for the parent population and as S^2 for a finite subset of this population. In the latter case, the experimental second moment becomes

$$S(n) = \frac{1}{N-1} \left[\sum_{i}^{N} |X_i - \bar{X}(n)|^2\right]. \tag{2.2}$$

Here N-1 is the number of degrees of freedom left after using N elements to determine $\bar{X}$. The standard deviation is defined to be the square root of the variance.

At this point, it is convenient to introduce the notation of statistical operators. We define a statistical operator to be any procedure which estimates a physical property of the distribution function $P(X_i,\bar{X},S)$. The measurable parameters of interest here are the first and second moments $\bar{X}$ and S^2. Since these parameters pertain to the sample set $\{X_i\}$, applying an operator $\hat{X}$ to the sample

population described by the function $f(X_i)$ must yield $\bar{X}$. The first and second moment operators are procedure operators and are denoted by $\hat{X}$ and $\hat{S}^2$ respectively. Using operator notation the estimation procedure becomes

$$\hat{X}[f(X_i)] = \bar{X}[f(X_i)] \tag{2.3}$$

$$\hat{S}^2[f(X_i)] = S^2[f(X_i)] \tag{2.4}$$

Notice is made that the classical operators $\hat{X},\hat{S}^2$ do not alter the sample set $f(X_i)$. However this is not a necessary restriction for the use of statistical operators.

In summary, the characteristics of an experimental sample set $\{X_i\}$ can be described by a probability distribution function $P(X_i,\hat{X},\hat{S})$. Both the first and second moments of P(N) are well defined and each moment has a corresponding operator.

D. Experimental Noise Sources

In this section we examine the experimental noise components inherent in the CARS optical three wave mixing process. The procedure used to deduce a value for the magnitude of a CARS noise source is described. Both reference (nonresonant) and sample (resonant) legs are provided experimentally to establish a means by which otherwise inaccessible parameters may be isolated.

The analysis of an experimental noise component occurs through a two step process. First, the parameter which constitutes the active noise source must be experimentally isolated. Once that source

becomes measurable it is averaged over a large number of measurements (N' = 2500) and the standard deviation is taken to be the true representative magnitude of the noise source. However, it is the percent variance $(S\%)^2$ which is used in the actual calculations of the noise components.

Under current experimental conditions, an optical signal is converted to the digital domain via the Fortran program TWEEK. This program then employs the general signal averaging operators $\hat{P}(N)$, and $\hat{S}^2(N)$. The average power operator $\hat{P}(N)$ yields an average $\bar{P}$ of the N measurements,

$$\bar{P} = \frac{1}{N} \sum_{i}^{N} (P_i). \tag{2.5}$$

$\hat{S}^2(n)$ is the second moment operator which gives S^2 relative to $\bar{P}$. The value for this uncertainty is obtained experimentally in the form [11],

$$S^2 = \frac{1}{N-1} [\sum_{i}^{N} (P_i)^2 - N(\bar{P})^2]. \tag{2.6}$$

Using operator notation, the experimental noise estimation processes become

$$\hat{P}[f(P_i)] = \bar{P}[f(P_i)] \tag{2.7}$$

$$\hat{S}[f(P_i)] = (S^2)^{\frac{1}{2}}[f(P_i)] \tag{2.8}$$

Program TWEEK has been designed (see chapter three) to apply these operators simultaneously on a reference (r), a sample (s) and the

ratio between reference and sample (s/r).

Ultimately, we are interested in optimizing the techniques used to measure $\bar{P}_{CARS}$, and restrict our present analysis to the noise components which contribute to S^2_{CARS}. As mentioned in chapter one, the expression relating the CARS signal amplitude to the experimental parameters can be written as

$$P_{CARS} = K \cdot P_1^2 \cdot P_2 \cdot T(t) \cdot S(q) \cdot \chi^2(\omega) \cdot D(n) \qquad (2.9)$$

The total variance for such multivariable functions must include a weighted contribution from each experimental parameter. For a general nonlinear function of the type

$$f = g^n \cdot h^m \qquad (2.10)$$

where the variables g and h are uncorrelated, the variance in f is

$$\frac{S_f^2}{\bar{f}^2} = n^2 \frac{S_g^2}{\bar{g}^2} + m^2 \frac{S_h^2}{\bar{h}^2} \qquad (2.11)$$

When all S values are expressed as a percentage deviation in the mean equation (2.11) simplifies to

$$S_f^2 = n^2 S_g^2 + m^2 S_h^2 \qquad (2.12)$$

Accordingly, the percent variance in P_{CARS} can be expressed in terms of the measured percent standard deviations for each experimental parameter as

$$S^2_{CARS} = 4\,S^2_1 + S^2_2 + S^2_t + S^2_q + 4\,S^2_\omega + S^2_d \qquad (2.13)$$

Here the noise components S_i are considered to arise from the following:

1. S_1: fluctuations in ω_1 pulse energy or mean power
2. S_2: fluctuations in ω_2 pulse energy or mean power
3. S_t: jitter in the time overlap of ω_1 and ω_2 due to time modulation in ω_1 and ω_2
4. S_q: jitter in the spatial overlap of ω_1 and ω_2
5. S_ω: $\omega_1 - \omega_2$ frequency jitter relative to ω_{res} in X_{res}
6. S_d: total noise due to detection.

Each of these components is considered below in terms of their origin, isolation and percent-standard deviation. Numerical methods used in this analysis are described by the percent variance expansions in the order of their simplicity.

1. S_1: Fluctuations in the pulse energy or mean power are naturally inherent to the pulsed Nd:YAG laser. Since P_1 is an independent parameter it may be monitored interchangeably in the reference or sample leg. A typical sample set of 2500 measurements of P_1 is shown in Figure 2.2. The magnitude of S_1 has been determined by averaging the standard deviation of $\bar{P}_1$ over 40 data groups of 500 shots each, and is taken to be 1.8 ± .2% after correcting for detection noise (see below) and optimization (tweeking) of the laser. This value is used in the remaining calculations requiring a knowledge of S_1.

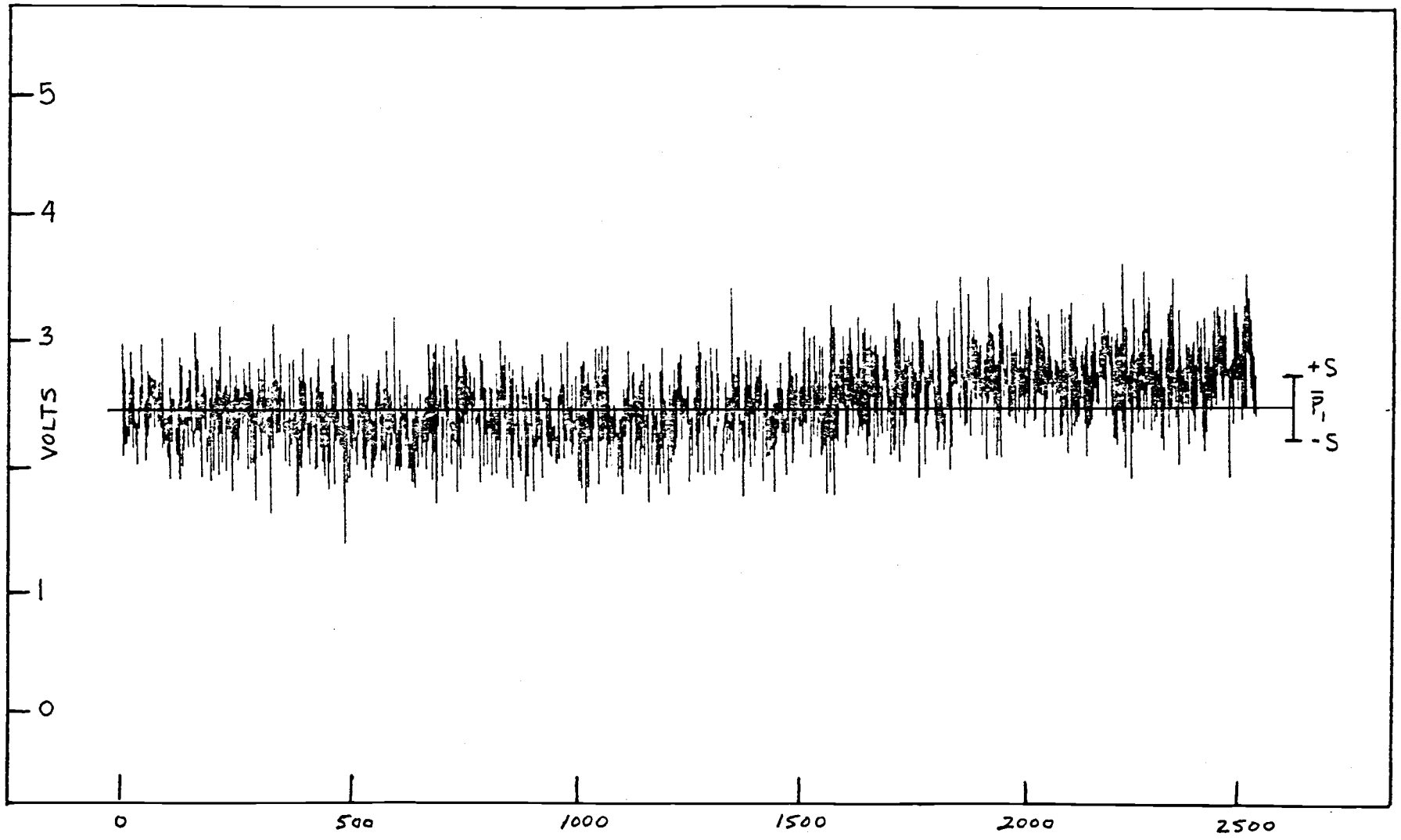


Figure 2.2. Shot-to-Shot fluctuations in $P_1(\omega_1)$ over 2500 shots.

2. S_2: Fluctuations in the dye laser output $P_2(\omega_2)$ depend on several internal parameters such as dye concentration and efficiency at ω_2, stability of the 355 nm pumping beam, and reflection efficiency of the grating as a function of angle. P_2 is also directly accessible for measurement and Figure 2.3 shows the P_2 power fluctuations for the red output at 591 nm. The magnitude of S_2 has been assigned a value of 18.8 ± 3.2%, but this value changes with ω_2 and typically ranges from ~10 to 30%. S_2 represents the mean of 6 data groups of 500 shots each.

3. S_d: The detection noise is considered to be a sum of two components

$$S_d^2 = S_{A/D}^2 + S_\gamma^2 \tag{2.14}$$

$S_{A/D}$ arises from noise in the electronic amplification and the analog to digital conversion process while S_γ results from photon statistics [12]. Total noise due to both sources is estimated by constructing a ratio between sample and reference detectors which measures the same events (e.g. the green laser intensity). From equation 2.13, it can be seen that

$$S_{s/r}^2(D) = S_r^2(D) + S_s^2(D) \tag{2.15}$$

The detector noise in the ratio represents the mean of 50 data sets of 500 shots each, and is 5.16 ± .8%. If we assume that each detector leg introduces an equal noise component, then we can assign a magnitude of 3.7 ± .4% to $S_r(D) \approx S_s(D) = S_d$.

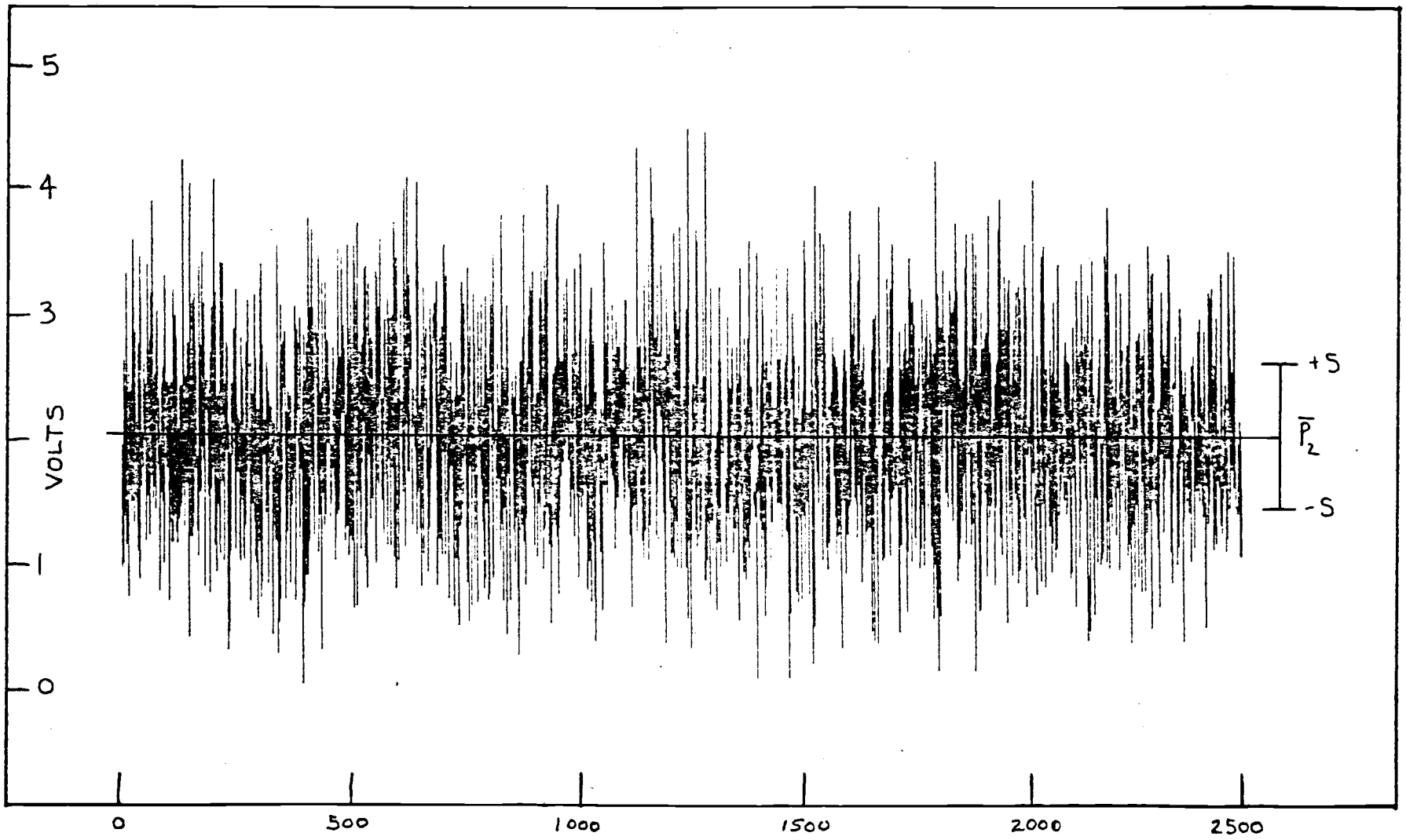


Figure 2.3. Shot-to-Shot fluctuations in $P_2(\omega_2)$ over 2500 shots.

4. S_q: In the CARS process we require that $P_1(\omega_1)$ and $P_2(\omega_2)$ interact simultaneously and expect the power of P_{CARS} to be proportional to the spatial overlap of laser pulses at the focus. We assume that both laser beams have a Gaussian profile and define the spatial coordinate functions to be $G\omega_1(r)$ and $G\omega_2(r)$. The spatial overlap is then described by

$$S(q) = \int G\omega_1(q)\ G\omega_2(q)\ dq \tag{2.16}$$

and $S(q)$ is restricted to the interval $0 \leq S(q) \leq 1$. Although we shall not specify the exact form of $S(q)$, values are suggested in Figure 2.4. Spatial jitter cannot be determined directly but can be deduced from the results of a nonresonant/nonresonant ratio and S_d. Using equation (2.13) the variance expansion for such a ratio can be written as

$$S^2_{N/N} = S^2_r(q) + S_s(q) + S_r(D) + S_s(D) \tag{2.17}$$

As before, we shall assume that both the reference and the sample legs introduce an equal noise and define a collective parameter S^2_L to be

$$S^2_L = S^2_q + S^2_d \tag{2.18}$$

The value for $S^2_{N/N}$ is represented by the mean of 5 data groups of 500 elements each and has a magnitude of 27.0 ± 1.3% after tweeking. Since $2S^2_L \simeq S^2_{N/N}$ we can isolate S_q and have assigned 18.7 ± 2.1 as

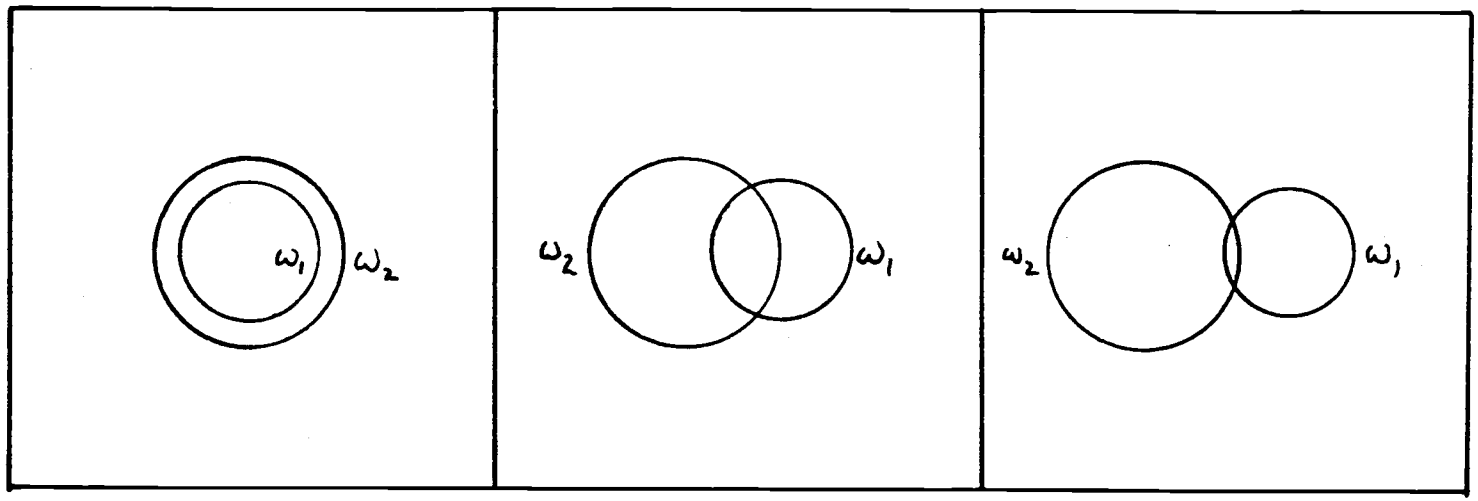


Figure 2.4. $P_1(\omega_1)$, $P_2(\omega_2)$ overlap variation due to spatial jitter.

its characteristic magnitude. Further, a value of 19.1 ± .65 will be used for S_L^2.

5. S_ω: Frequency jitter relative to ω_{res}, arises directly from the factor of $\omega_1 - \omega_2$ in the expression for $\chi^{(3)}_{CARS}$ (the χ^2 dependence of the signal level produces the factor of 4 in equation 2.13). There are two numerical methods which can be used to arrive at a value for S_ω. First, by using equations (1.15) and (1.16) we can write the variance relationship

$$4 \cdot S_\omega^2 = S_{CARS}^2 - S_{NR}^2. \tag{2.19}$$

The total experimental uncertainty S_{CARS}^2 has been determined as the mean of 10 data groups (500 shots each) measured at the Q branch maximum for NO in the sample leg. After optimization of experimental parameters, the signal has a characteristic standard deviation of 54.7 ± 2.1%. Figure 2.5 shows the shot-to-shot fluctuations of a typical resonant data set. For a similar nonresonant data set, a value of 45.3 ± 1.7% is deduced for S_{NR}. Solving equation (2.19) then gives a value of 15.2 ± .85% for S_ω. A second method for evaluating S_ω involves the ratio between a resonant and nonresonant sample. Rearranging the variance expansion for equation (1.17) yields

$$4 \cdot S_\omega^2 = 2 \cdot S_L^2 - S_{R/N}^2 \tag{2.20}$$

and S_ω is determined to be 12.5 ± .13%. Since both methods produce a comparable value (within numerical uncertainty) we have adopted the first value.

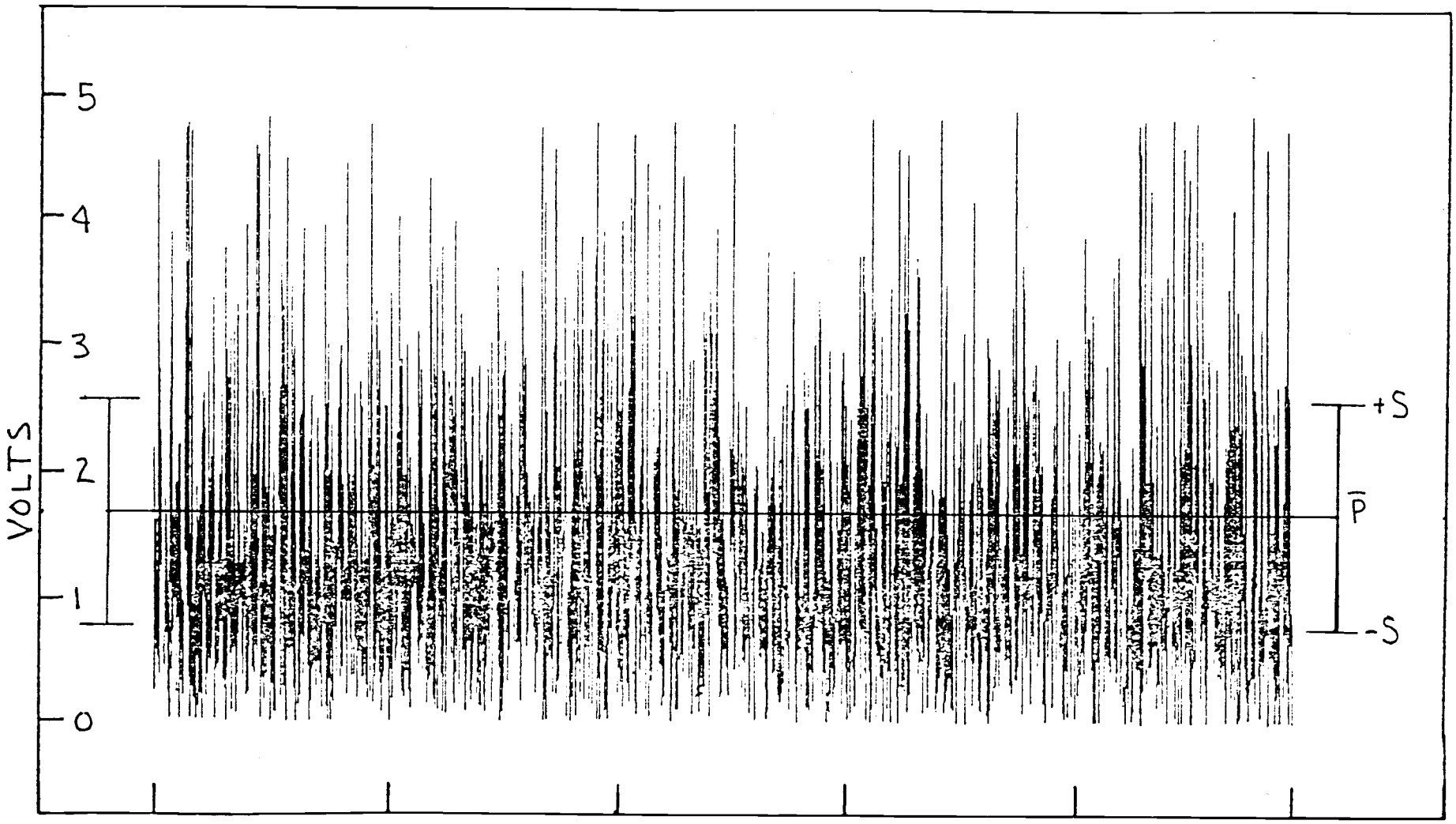


Figure 2.5. Shot-to-Shot fluctuations in $P_{CARS}(\omega_{TW})$ over 2500 shots.

6. S_t: Very high frequency time modulation jitter is naturally inherent to multimode laser experiments and is a significant noise component in S^2_{CARS}. If we define the pulse energy distribution in time by the functions $f\omega_1(t)$ and $f\omega_2(t)$, then the time overlap of ω_1 and ω_2 can be written as

$$T(t) = \int f\omega_1(t) \cdot f\omega_2(t)\, dt \qquad (2.21)$$

Again we do not specify the nature of f (this will vary from laser shot to laser shot) and restrict T(t) to the interval $0 \leq T(t) \leq 1$. Figure 2.6 illustrates overlap jitter due to time modulation. The magnitude for S_t can be deduced from the variance expansion of equation (1.15)

$$S_t^2 = S^2_{CARS} - 4 \cdot S_1^2 - S_2^2 - 4 \cdot S_\omega^2 - S_L^2. \qquad (2.22)$$

The appropriate substitutions yield $S_t = 36.4 \pm 17.0\%$.

Numerical substitution of the above individual S_i values into a variance expansion for a resonant/nonresonant ratio gives a calculated experimental uncertainty for $S_{R/N} = 40.6\%$. As a check we compare this value to the measured value of 31.4%. Table 2.1 summarizes our results and the last line shows the relative importance of the various noise terms. Clearly, the dominant noise contributions come from the time (~46%) and frequency jitter (~32%) associated with the multimode character of the Nd:YAG and dye lasers. Under the conditions of these experiments,

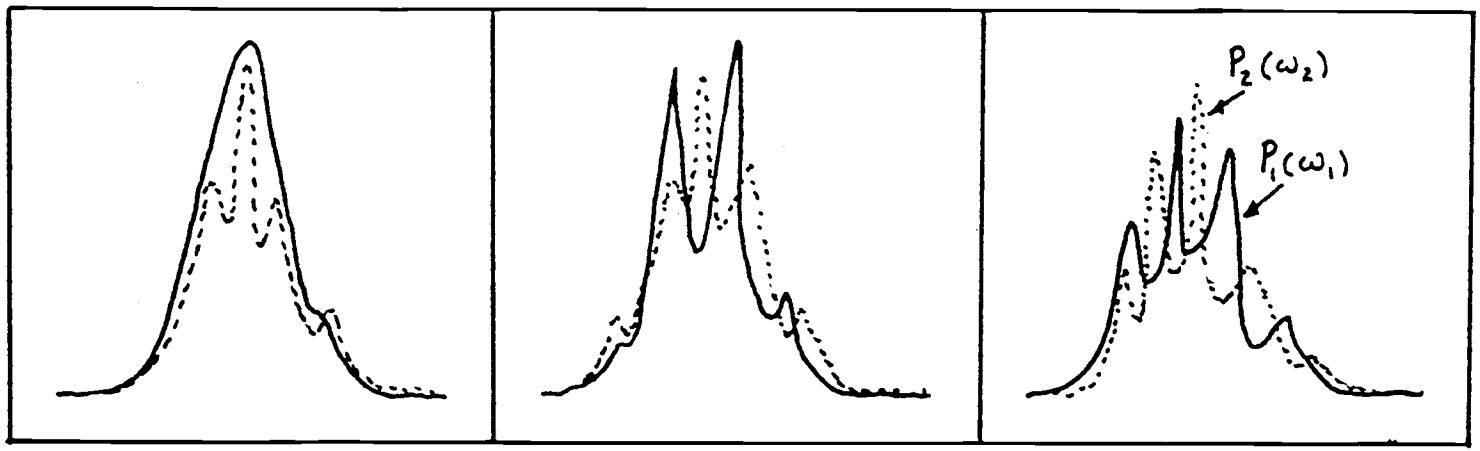


Figure 2.6. Time modulation jitter in the overlap of two laser pulses.

TABLE 2.1 % Noise Contributions in a CARS signal for NO

S^2_{CARS}	=	$4 \cdot S_1^2$	+	S_2^2	+	$4 \cdot S_w^2$	+	S_t^2	+	S_q^2	+	S_d^2
$(53.8)^2$	=	$4(1.9)^2$	+	$(18.8)^2$	+	$(3.7)^2$	+	$4(15.2)^2$	+	$(18.7)^2$	+	$(36.4)^2$
1.000	=	.005	+	.122	+	.005	+	.319	+	.121	+	.458

the ω_1 linewidth was about 0.01 cm^{-1}, corresponding to simultaneous lasing of about four longitudinal cavity modes. By careful adjustment of the two intracavity etalons and by activation of the electronic line narrowing accessory of the laser (a slowly opening Q-switch which allows one longitudinal mode to dominate), it is possible to reduce the linewidth to ~0.02 cm^{-1} and to achieve single mode operation about 50% of the time. Unfortunately, from a time jitter standpoint, two modes produce a greater fluctuation than do four [13]. The optimum solution would be to use a reliable, single mode operational laser. Unfortunately this would require redesign of the laser. Alternatively, one might sacrifice the narrow linewidth by going to a very broad source in which the time modulations are so rapid that the overlap functions of the ω_1 and ω_2 beams start to become constant again.

Since time modulation jitter is reduced when either ω_1 or ω_2 beams act as a single mode source, one might consider a second approach. That is, to use a single mode C.W. tunable laser as the basic ω_2 oscillator source and increase the power of this beam by passing it through amplifier cells pumped by the Nd:YAG laser. This method would not only reduce the time jitter, but would also aid in the reduction of the third largest noise component (12.2%) due to energy fluctuations in $P_2(\omega_2)$. Such a C.W. oscillator source was not available in this work.

Some improvement in the noise arising from spatial jitter (12.1%) may be achieved by using more rigid optical mounts and shorter optical paths. Reducing the vibrations along the optical table may also prove

to be beneficial although such vibrations are thought to be small. The energy fluctuations in $P_1(\omega_1)$ and detector noise sources are also considered to be negligible.

Sample/reference ratioing experiments have proven to be quite significant since the standard deviation in $\overline{P}_{CARS}$ is reduced by almost a factor of two. That is, the deviations in the resonant/nonresonant ratio are only 31.4%. This is due to the elimination of the fluctuations in P_1,P_2 and in the time jitter in the ratio. However, it should be noted that to achieve this improvement requires great care in assuring comparable spatial overlap in both the reference and sample focal volumes. The use of long focal length lenses has helped to achieve this goal and it is recommended that the resonant/nonresonant ratio scheme be adopted in a routine manner for future work.

E. Experimental Distributions of Nonlinear Processes and Robust Operators.

In this section we shall consider the possibility of improving our estimation of $\overline{P}_{CARS}$ by examining the distribution which underlies the sample population. Due to the high nonlinearity of the three wave mixing process and the large uncertainties (~55%) in the average, we have reason to suspect that the sample population may be contaminated with outlying or "bad" data points. Distributions which characterize populations containing outliers are normally heavy tailed thereby reducing the precision of the classical averaging operator. That is, they pull the average toward them. In the case

of such abnormal distributions, Robust statistical operators have been designed to modify the classical operator in an accenting fashion, so that erroneous data has much less influence on the estimation process. We shall propose two such Robust operators which may prove effective in this type of operation for the CARS process.

A typical sample set for P_{CARS} (Figure 2.5) has shown that the majority of the fluctuations in P_{CARS} extend above the average value for the sample set. This suggests that this nonlinear distribution set must be heavily contaminated. Figure 2.7 illustrates this point by comparing the sample sets (N = 2500) of P_1 and P_{CARS} in terms of their distribution structure. We should expect that on the more typical experimental interval $5 \leq N \leq 100$, discontinuity (contamination) in the sample set will be greatly enhanced, and our estimation $\bar{P}_{CARS}$ may be quite poor. Thus we now examine the nature of the Robust operators.

Robust statistics have been shown [14,15] to be effective in restoring the distribution continuity by their ability to alter the sample population set. The first Robust operator which we have considered was suggested by Quenoville (1956) [16], and is denoted $\hat{R}^{-}(N)$. The Robust estimation process is

$$\hat{R}^{-}(N)[f(X_i)] = \bar{P}(M)[g(X_i)] \tag{2.23}$$

In this process, the Robust operator eliminates data points from the set by judging whether a given data point is contained in the interval

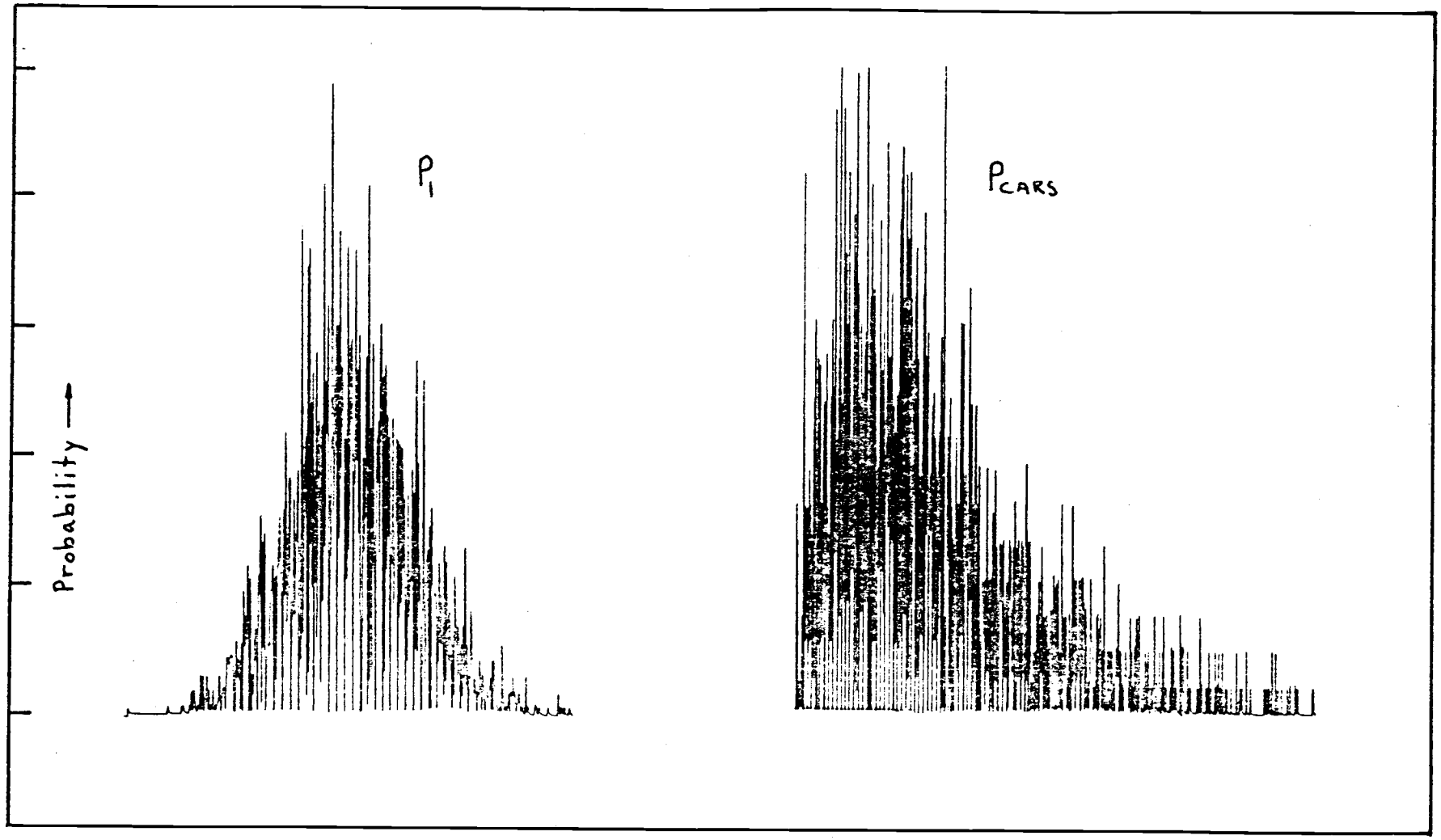


Figure 2.7. Sample set distributions for P_1 (left) and P_{CARS} (right).

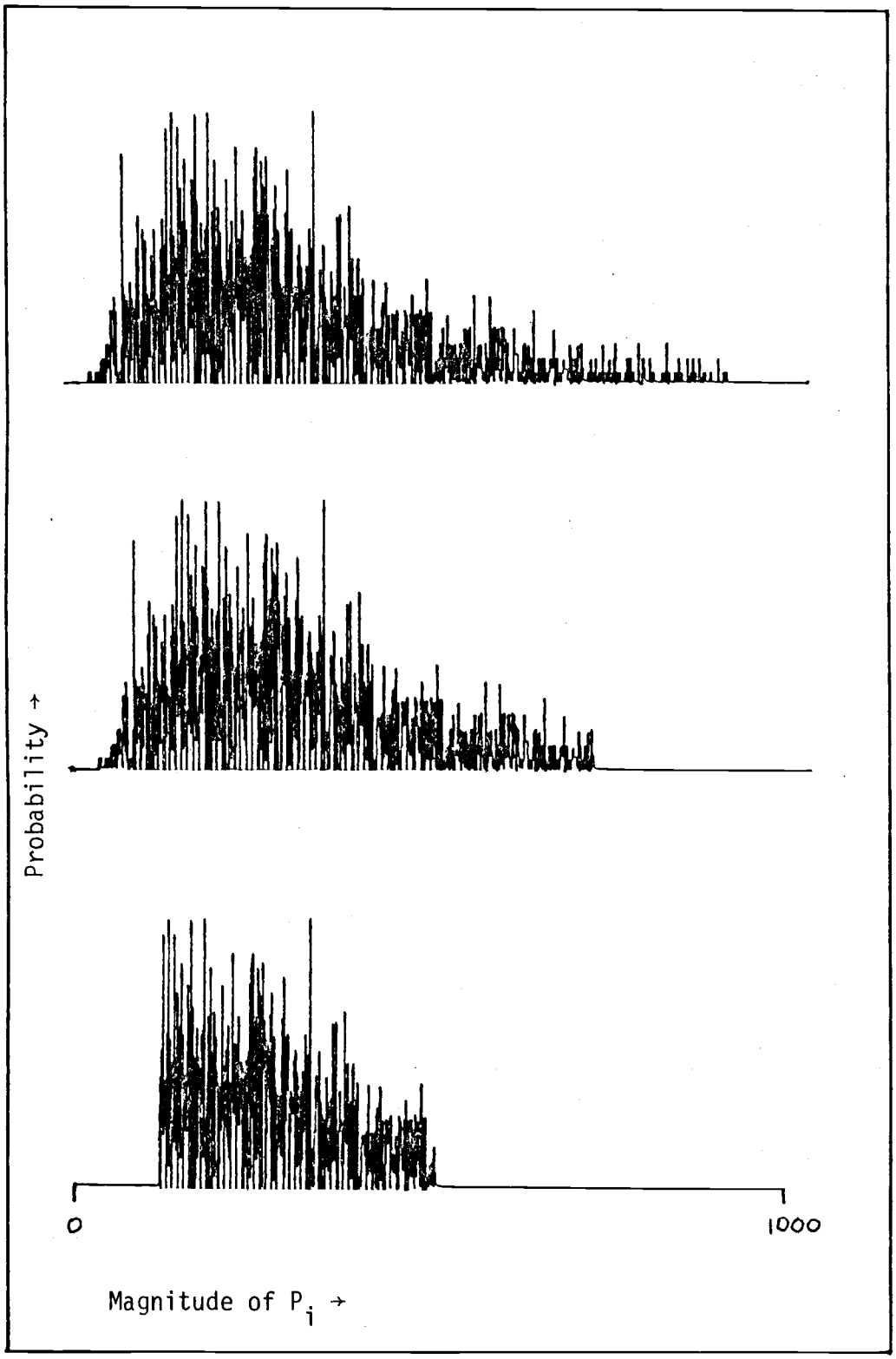


Figure 2.8. Effects of $\hat{R}^-$ on P_{CARS} sample set.

$$\bar{P} - b \cdot \hat{S} \leq P_i \leq \bar{P} + b \cdot \hat{S} \qquad (2.24)$$

where b is a scaling factor which is chosen to define an acceptable width to the distribution. All data points outside this interval are eliminated and the sample set decreases from N to M elements such that $g(X_i)$ $f(X_i)$. The average of this set is then computed to give $\bar{P}(M)$. Thus we say that $R^-(N)$ is the elimination operator. Figure 2.8 shows the effect of this operator on the CARS population set of Figure 2.5 for various values of b.

The second Robust operator of interest, $R^+(N)$, expands the data set according to equation (2.24) by the estimation process

$$\hat{R}^+(N)[f(X_i)] = \bar{P}(N)[h(X_i)] \qquad (2.25)$$

but does not alter the total number of data points N. We call $\hat{R}^+(N)$ the replacer operator since it functions to first reject the outlying points of the initial N data points and to then replace these with sufficient new values to restore the set to N values. This process is then repeated until finally all values fall within the acceptable width.

Because of their definitions, the effect of these Robust operators is to reduce the standard deviation of the average. The critical test is, of course, to see whether the Robust average more accurately represents the "true" or absolute average. This has been the case for other studies of non-normal distributions and we believe that it is likely to be true for the nonlinear CARS process as well. Unfortunately, the absolute (true) value of $\bar{P}_{CARS}$ is not available

for analytical analysis and so we cannot prove this assumption. In any future work to test the Robust averaging methods, it may be worthwhile to examine the ability of different operators to accurately reproduce relative peak intensities; for example, the Q branch structure of the D_2 vibrational transition [17]. Since these relative intensities can be accurately calculated, this should permit a more quantitative judgment as to the merits of Robust operators in their application to spectroscopic techniques.

CHAPTER 3

COMPUTER PROGRAMS FOR PULSED LASER SPECTROSCOPY

This chapter provides a detailed description of the computer programs developed as part of this research for data collection, data reduction, frequency calibration and graphic display. Although we are interested in three wave mixing experiments, the structure of the trigger pulse network and data storage system may provide for the adaptation of this program system to other optical techniques.

The experiments presented in this thesis involve a sequence of four individual run programs. The versatility of the program system is determined by the ability of each element (program) in the run sequence to stand alone as a useful program. As an example, the first element, known as TWEEK, was used in chapter two as a stand alone program to evaluate the experimental noise components of the optical three wave mixing process. In addition to this application, TWEEK is used in the run sequence to optimize experimental conditions prior to the recording of a spectrum.

The four functions which define the experimental sequence are discussed in sections A to D. The first functional procedure is the experimental optimization. The second procedure is to collect and store the spectrum as a function of the dye laser frequency. Once the data is stored, the third element processes that data so that it may be displayed as a function of the three wave mixing frequency. The final function is to graphically display the processed data in convenient form.

The programs developed for the run sequence are FORTRAN control programs which access MACRO subroutines. In addition to the explicit description of these control programs this chapter will provide details pertaining to the FORTRAN - MACRO interaction.

A. Experimental Optimization

The experiment section of chapter two describes the means by which two optical three wave mixing beams are generated. We now consider the steps involved in converting these analog signals to the digital domain. This conversion procedure then defines the experimental interface between the optical three wave process and the minicomputer used in the laboratory.

1. Analog to Digital Converters

A PDP 11/10 computer system controls the interface process and offers the use of two isolated analog to digital (A/D) converters. These A/D converters are best characterized in terms of the major functional components which collectively define the A/D conversion process. They are the sample and hold units (S/H), channel select multiplexer, and the actual successive approximation converter.

One of the A/D converters which is available is a 12 bit converter and is program accessible via the DR11-K parallel I/O interface board. The SHA5 S/H unit associated with this converter is controlled by an external input control start line hereby referred to as the CNTL ST L. When CNTL ST L is held at +5 volts, the S/H unit continually samples

the input signal level. The high to low transition of the input trigger (CNTL ST) causes the S/H unit to hold the input signal with a settling time <5 nsec. At present we do not use a multiplexer with this converter. After the signal is held in a fixed state it is sent directly to the input of the approximator. The approximator is designed to respond over a 0 to +5 volt range, which allows a resolution of one part in 2^{12} = 4096. Thus one count by the converter corresponds to a voltage change of ~1.2 mv. The time for one conversion cycle is on the order of 25 μsec.

A 10 bit converter is also available for use and is program accessible through the AR-11 interface board of the PDP 11/10 computer system. Unlike the SHA5 S/H unit of the 12 bit converter, the S/H associated with the AR-11 cannot be directly triggered from an external pulse. The external start function for this converter is only available through program control. When bit #4 of the A/D control status register (CSR) is set the high to low transition of a trigger pulse on the EXT ST L, will cause the S/H unit to hold the input level. In the absence of an external trigger, the S/H unit may be activated when the A/D start bit of the CSR is set. A four bit multiplexer which is also program controlled by the CSR, allows up to 16 sequential A/D conversions. Presently, we are concerned with only the first four channels. Again the input voltage range is 0 to +5 volts so that each count by the converter corresponds to ~5 mv or a resolution of one part in 2^{10} = 1024. The conversion cycle for this converter is ~30 μsec.

2. Parallel A/D Conversions

With two, isolated A/D converters analog signals can be converted in a parallel or sequential manner. In the parallel process, two input signals are captured by the S/H units simultaneously and then converted to a digital value sequentially. In the sequential process one signal is held and converted and the second signal is processed in a similar fashion. Due to the nature of our pulsed experiments, this distinction becomes important when considering the requirements of the pulse behavior in time. For the parallel process, one might think that we require only the coincidence of both pulse maxima. However, we have determined experimentally that, in the case of a ratio between two signals, less uncertainty is introduced in the conversion process when both pulses have the same shape (i.e. the same rise and fall time). In sequential conversions, we further require the fall time to be as long as possible. This assures a small change in the signal intensity over the time interval of two conversion cycles (~60 μsec). In this section we are concerned with only two signal inputs and therefore restrict our discussion to the more efficient parallel process. For this type of conversion process we can summarize the required conditions for the input pulses as follows. Both pulses should have the same shape in time and be positive going. To preserve resolution in the conversion, the peak maxima should approach the +5 volt limit.

The photomultiplier outputs which characterize the three wave mixing signal are typically on the order of 250 - 500 mv in magnitude.

To satisfy the previous requirements for parallel conversion, the photomultiplier output signals are inverted, amplified and shaped using RC frequency filters. The output of the sample leg (RCA) photomultiplier is manipulated using an ITHACO 1201 low noise pre-amplifier whose inverting input is terminated at 1 MΩ. A Tektronix 1A7A amplifier is used for the reference leg IP28 phototube and the output of both amplifiers is monitored on a Tektronix dual beam oscilloscope.

3. A/D Control Start Triggers

When an A/D conversion takes place between pulses, or when the light to the photomultiplier is blocked, a digital value of Ø should be obtained. However, this is not always the case; a DC offset or low frequency ripple (60 HZ) may contribute to the input signal level. We can compensate for the positive components of such background by sampling the pulse at the base of the peak and subtracting this digital value from the peak value. Thus, we make two A/D conversions on each pulse; first the background and then the peak.

As mentioned earlier, the 12 bit S/H unit is controlled by the high to low transition of the CNTL ST trigger pulse. The actual conversion of the held signal starts when the Input Data Buffer register (DR) is referenced by a program instruction (e.g. MOV @#DR, RØ starts the A/D conversion). Conveniently, the AR-11, 10 bit converter may also be started externally by a high to low transition on the EXT ST line. Thus, the same trigger pulse may be used to

simultaneously sample two pulses on the 10 and 12 bit converters. The 10 bit converter will finish first since the approximator is started by the S/H unit. We shall use the CNTR ST and EXT ST lines to sample the pulse peaks. The background A/D processes may also be controlled externally but not in the way previously defined (since there is only one EXT ST input). The AR-11 interface board contains a display control network intended for interface with a storage scope. The display status register (DSR) controls the display pulse network and has available one return line. This return line, (ERASE RTN L) will set a bit of the DSR on the low to high transition at its input. We can then test this bit, and when set, proceed with the A/D conversions. In this case the 10 bit converter is started by the CSR A/D start bit. A program reference is made to DR and causes the 12 bit approximator to signal the 12 bit S/H unit, and a conversion is started. The delay between the 10 and 12 bit S/H is the time for one executable program statement (~2 μsec). Therefore, by using the ERASE RTN L of the display network, we can mimic an external A/D start and, essentially, hold both input signals at the same time.

The ND:YAG laser provides two output pulses which are used to control the A/D conversion sequence. The first pulse is sent out when the flashlamps arc, corresponding to the excitation of the ND:YAG rod. This is known as the lamp pulse. A second pulse occurs ~250 μsec later when the Q switch is opened and the laser fires. The lamp pulse is terminated at 50 Ω at the input of a Hewlett Packard 222A pulse generator. The output from this generator is defined to be a

+5 volt, 50 μsec pulse using the pulse width and amplitude controls. This output pulse is then connected to the ERASE RTN L, and the rising edge is positioned at the background A/D site using the pulse delay control. The Q switch sync out pulse is also terminated at 50 Ω at the input of a second Hewlett Packard pulse generator. The pulse generator output is connected to both the CNRT ST and EXT ST lines. We position the falling edge of the +5 volt output pulse at the sample pulse peak using the pulse width control. Both the background and peak A/D trigger pulses are set using an oscilloscope which displays a trigger pulse and the sample pulse.

4. Program Tweek

Tweek is an interactive Fortran program designed to monitor the experimental behavior of two analog signals and their ratio, as a function of time (i.e. over X shots). This behavior is characterized in terms of an average and % standard deviation over N shots (which is calculated and displayed by the program). An unformated, sequential data file may be created to allow for the storage of the sample peak values returned from the A/D conversion process. The shot data, which is collected from the A/D buffers, is arranged in data groups of the N shots. A maximum of 200 groups may be constructed, but there is no limit on N, since no storage arrays are allocated. Thus a total of M·N shots are collected (and stored) where M is the number of groups. In addition to calculating the average (and S) for N shots, the program computes an average at the end of each data group

corresponding to the total number of shots collected up to that point. This average is termed the running total and may represent the parent or absolute experimental average. The program printout displays the average and % standard deviation for the two analog inputs and their ratio, as a function of the group number (and N shots). In addition, the running total is presented next to the data for each group. The structure of the data printout allows the experimentalist to evaluate the experimental parameter of interest by a digital analysis of the % standard deviation. The optimization procedure is continued by an experimental adjustment and a program loop to start another run. Program output and documentation of TWEEK are given in the Fortran Library of Appendix III. The general flowchart diagram for TWEEK is illustrated in Figure 3.1.

The FORTRAN control program collects data from the 10 and 12 bit converters via a call to the MACRO subroutine known as COLTWO. This subroutine is a general collecting routine whose function is to return the peak and background values of two sample pulse inputs to the control program. It becomes the responsibility of the control program to subtract the background and construct the ratio. COLTWO.MAC is listed in the Macro library of Appendix II and the details of the subroutine program execution are discussed in the next section under Macro library. For now, we briefly state that COLTWO may use either the parallel or the sequential A/D process and its associated trigger pulse network, both of which have been previously defined. This option is set in the first interactive statement of TWEEK.

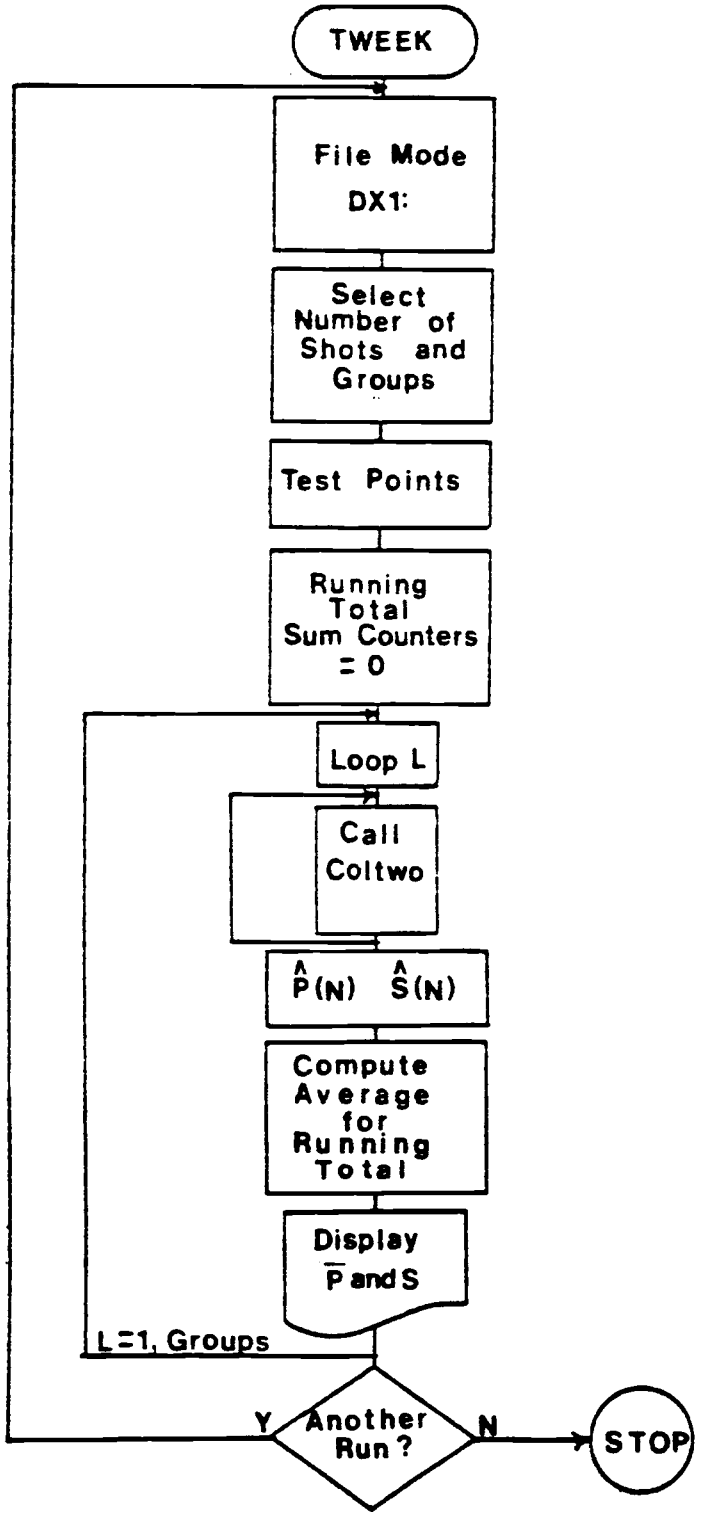


Figure 3.1. Flowchart diagram for program TWEEK.

The interactive mode then prompts for a file name (if one is desired), selects the number of data groups and the number of shots/groups to be averaged. A scale factor which multiplies the ratio is set and the test point portion of the program is entered. The user may select any integer number of test points which consists of an average over 30 shots. Each shot value is displayed along with the average so that the input signal levels may be adjusted to approach the +5 volt limit. The inner loop, of a nested DO LOOP series, uses COLTWO to collect N shots, subtracts the background, constructs a scaled ratio and adds the values to an appropriate sum counter used in computing the average and standard deviation. The outer loop counts the number of data groups, initializes the sum counters, computes the averages and displays the data. After M groups have been processed an interactive prompt occurs for another run.

B. Spectral Data Collection

Once experimental conditions are optimized, the second element of the run sequence is called upon to record the optical three wave mixing spectrum. Two Fortran control programs (AVRAGE and RATIO) are available to serve this purpose and are listed and documented in the Fortran Library (Appendix III). It is the objective of these programs to produce an output file which offers information pertaining to the spectral envelope structure as a function of transition energies (frequencies). As indicated previously, it is the change in ω_2 which allows us to probe the resonant three wave mixing transitions and

therefore, we seek out methods for using the change in ω_2 to calibrate the transition energies. This is accomplished by time averaging the power generated in the three wave mixing beam (P_{CARS}) at a fixed dye laser frequency ω_2. As a calibration reference, ω_2 is measured at the beginning of each scan. The dye laser is then stepped by $\Delta\omega_2$ increments and the average power operator is again applied (over N shots) at a new dye laser frequency $\omega_2 + \Delta\omega_2$. Thus each record of the data file contains the average transition intensity and the average change in dye laser frequency, $\Delta\omega_2$. The change, $\Delta\omega_2$, is determined by averaging the change in the interference fringe intensity produced by passing the dye laser zero order output through a .9975 cm^{-1} etalon. A maximum in the fringe intensity will occur at every .9975 cm^{-1}. Further the standard deviation in $\overline{P}_{CARS}$ is recorded to yield information about the uncertainty of the transition amplitude.

1. Data File Structure

The PDP 11/10 computer system which has been employed is controlled by the RT-11 V04-B monitor system and uses dual RX01 single density, floppy disk units known as DX0: and DX1:. Data file storage is defined in terms of the blocksize and each block contains 512 bytes of disk memory. A logical unit number may be assigned to a data file with the Fortran OPEN or CALL ASSIGN statement. Execution of this statement causes the file name to be placed in the disk directory. Data space is then allocated and an input data buffer zone is created in program memory. A series of Fortran WRITE statements

causes the input data to be stored in the buffer. When the buffer is full, it is written into disc memory as ASCII coded characters (i.e. one byte/character). The record size is defined to be the number of bytes used in one WRITE statement and is determined by the total FORMAT (or field size) of the variables in the WRITE statement. Data files are closed when all the input data has been written and before the program terminates by the CLOSE or CALL CLOSE statements.

2. The Macro Library

All programs listed in the Macro Library are general purpose subroutines and carry the RT-11.MAC extension. They are .GLOBL subroutines and may interact with Fortran variables through the Fortran CALL statement. When Fortran variables are passed as arguments to the Macro routine, a stack is created in register #5 (R5). This stack contains the numbers of arguments in the pass and the corresponding address of each variable. Because Fortran passes addresses we will use the @(R5)+ register mode to manipulate the stack in a sequential manner.

We begin the construction of the Macro Library with a discussion of the three subroutines called by the Fortran control program, AVRAGE. Other members of this Library are described in later sections as they are used. Information pertaining to the status and buffer registers used by these subroutines is located in Appendix I.

COLTWO.MAC is a general data collecting routine which samples the background and peak values for two input signals. The Fortran

CALL statement has the form

CALL COLTWO(IN, B1, B2, PEAK1, PEAK2)

IN is the input variable and determines whether the 12 bit converter is used in the parallel A/D process or the 10 bit conversion are used in the sequential process. External A/D start triggers are assigned by the trigger network according to the value of IN. The integer output variables are returned from the stack to the Fortran program in the order they appear in the CALL statement.

PLTTWO.MAC is a subroutine which utilizes the 10 bit digital to analog converter of the AR-11 to plot the integer variables IX and IY on a strip chart recorder. The CALL statement is

CALL PLTTWO(IY, IX)

IX and IY analog values occur at the XOUT and YOUT BNC connectors and range from -2.5 (0) to +2.5 (1024) volts. When bit #11 of the DSR is cleared a +5 volt level is maintained at the WRITE THRU BNC output. Setting bit #11 causes WRITE THUR to go low and a pulse is created to drive a linear dual pen strip chart recorder. The pulse width is controlled by the time elapse between execution of the BIC and BIS statements.

STEP.MAC is used to create a pulse output at the DISP CH2 BNC connector, which controls the stepping function of the dye laser. This pulse width is determined by bit #9 of the DSR in the reverse

of bit #11. One pulse is generated each time CALL STEP is executed in the Fortran program.

3. Program AVRAGE

AVRAGE is an interactive Fortran control program which calls upon the three Macro subroutines COLTWO, PLTTWO, and STEP, to aid in the data collecting, real time plotting, and frequency tuning of an optical three wave mixing spectrum. AVRAGE, like other members of the run sequence, may also be employed as a stand alone program. The general function of this program is to record the average values for two analog input signals. However, program notation and variable names are representative of the CARS process. Figure 3.2 illustrates the general flow chart diagram for AVRAGE.

After the initialization of program arrays and variable modes, the program user enters the parameter (IAD) to select the parallel (12 bit A/D on P_{CARS}) or sequential (10 bit A/D on P_{CARS}) conversion process which then defines the trigger pulse network. For the novice program user, an operational checklist (CHECK) has been included, which displays instructions for the BNC cable assignments needed for the interface.

AVRAGE is designed to record any number of data points (records) restricted only to the number of available bytes in the disk memory. When more than one spectrum is to be recorded on a single disk, it becomes desirable to know when the disk is full without checking after each run. This capacity has been incorporated in AVRAGE

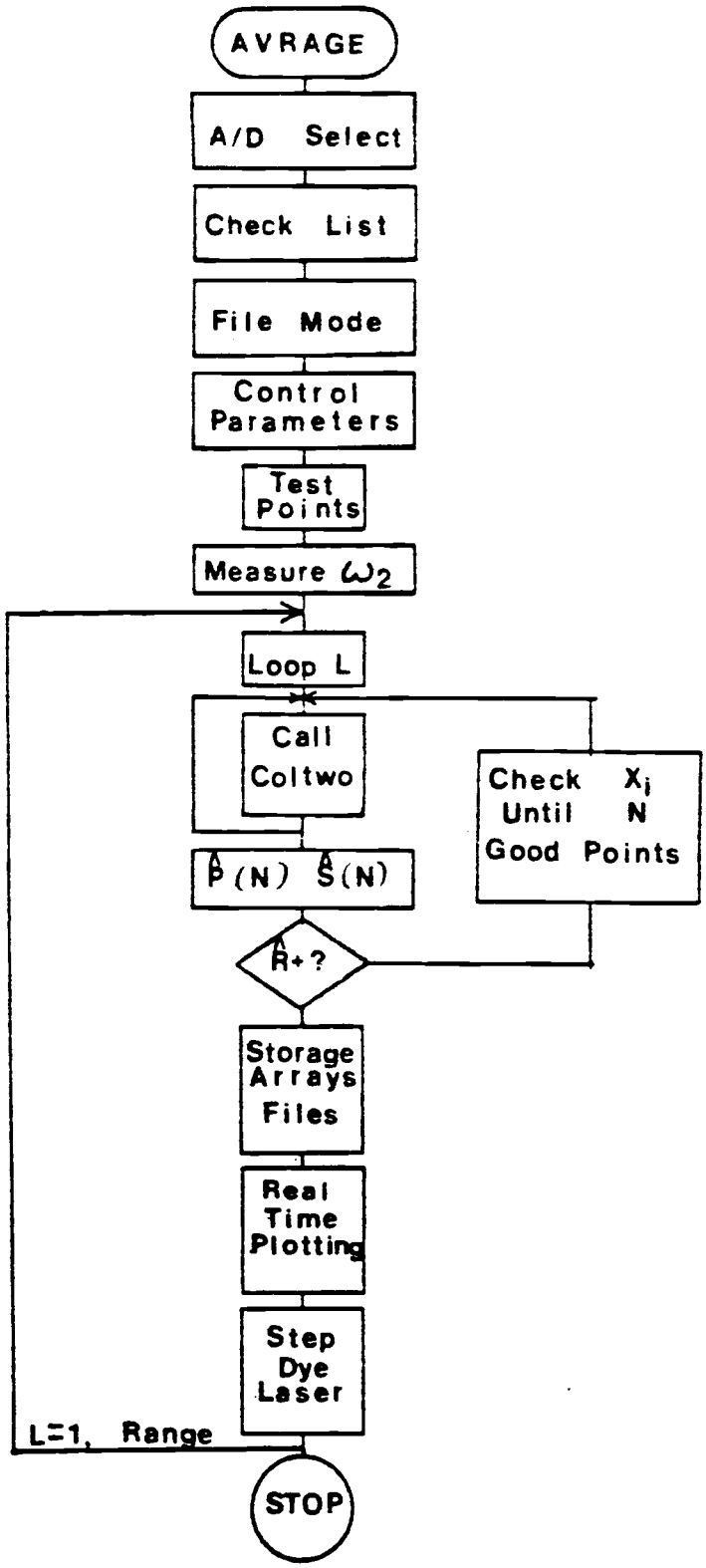


Figure 3.2. Flowchart diagram for program AVRAGE.

through the interactive file mode. AVRAGE uses the CALL ASSIGN statement to create an output data file on DX1: and assigns a logical unit number to that file. The file name is entered by the program user in the form X'XXXXX. DAT where X' must be an alpha character and X is any other alpha numeric character. Since DX1: data file space is counted by blocks, we must know the number of bytes used in one record of order to keep track of available disk memory. The record WRITE FORMAT is (X, I4, F5, 1, I4). This defines the record size to 16 bytes including carriage return and line feed. Thus, it requires one block of disk memory to record 32 data points. Further, RX01 units only allow the use of half the remaining (available) blocks [17]. In the file mode the number of free data blocks on DX1: is entered along with the desired number of data points (records). The program then calculates the number of blocks that are required and determines if there is enough available memory. If so a file name is entered and placed into the disk directory. The output data file is then opened. The program user may also enter an eight character I.D. label which is written as the first record of the data file. AVRAGE now enters the first of three parts belonging to the interactive mode functional block.

AVRAGE is capable of storing up to 99 shot values in core memory, which will be used to calculate the average and standard deviations. The parameter SHOTS determines the sample set size and is entered by the program user. The dye laser stepping resolution may also be changed by entering an integer value stored in IRES.

Once the data record is ready for storage, the average values for the CARS intensity and fringes are plotted on a strip chart recorder in the real time plotting mode. Because of the large fluctuations in the three wave mixing signal, the data is best displayed in a smoothed fashion. The plotting mode uses an exponential growth and decay scheme in which the intensities (IAVEW3, IAVEFR) to be plotted are fitted to the function

$$Y_{PLOT} = Y_{i-1} \cdot e^{-1/\tau \cdot \Delta X} + (1 - e^{-1/\tau \cdot \Delta X}) Y_i \qquad (3.1)$$

Here ΔX is the increment of the X-axis expansion factor and τ is a plotting time constant. The second part of the program interactive mode sets the parameters TOW (τ) and JMAX (ΔX). A scaling factor of 3 has been included in JMAX so that the number of external clock pulses to the recorder corresponds to the internal clock, X-axis distance in cm. The first and second coefficient terms of equation (3.1) are defined by the program variables AI and SI respectively.

AVRAGE offers the use of two average power operators for the signal averaging process described in chapter two. They are $\hat{P}_{CARS}$ and $\hat{R}^{+}_{CARS}$. The final part of the interactive mode selects the operator; $\hat{P}_{CARS}$ is the default. If the Robust operator is chosen, the variance range (in units of sigma) is entered into program memory by the program user as the variable XLIM.

A test point mode has been included which displays the four untreated shot values, returned as the output variables of COLTWO. This allows the program user to adjust the input signal levels using

the amplifier gain controls. It is intended that this adjustment occur at the maximum peak intensity in the spectrum. After the signal levels have been optimized the dye laser is scanned to the initial ω_2 frequency which is then measured and entered into the variable SWIN (in angstroms). The program is now ready to record the spectrum.

Recording of the spectral data occurs in a series of nested DO loops. The most outer loop in L counts the number of data records and initializes the sum counters used in the averaging process. It is the inner loop in I which actually controls the data processing. In this loop COLTWO is called in a DO loop which counts from I=1, SHOTS. The background is subtracted from the peak value and the resultant value is added to a sum counter $\sum_{i}^{SHOTS} X_i$, squared and added to the counter $\sum_{i}^{SHOTS} (X_i)^2$. These values are used by $\hat{P}_{CARS}$ (or $\hat{R}^{+}_{CARS}$) and $\hat{S}_{CARS}$ respectively. The operators $\hat{P}_{CARS}$ and $\hat{S}_{CARS}$ are then applied and the data record is constructed. If $\hat{R}^{+}_{CARS}$ has been chosen, each member of the sample set is checked against XLIM, and may be eliminated. In this case a DO which recalls COLTWO is entered until the sample population set contains SHOTS "good" elements. This changes the data records to reflect the robust operators and the record is written into disk memory. Finally, the data record is plotted in a DO loop which calls PLTTWO from J=1, JMAX. PLTTWO plots the CARS intensity and the fringe intensity values. The dye laser is then stepped by IRES increments and the loop in L starts the data processing for the next record.

The versatility of AVRAGE as a stand alone program has been demonstrated by its use in the recording of a Photoacoustic Raman Scattering (PARS) spectrum of CH_4. Figure 3.3 shows the results compared to the CARS spectrum of CH_4.

4. Program Ratio

Ratio is a hybrid mixture of programs TWEEK and AVRAGE. It has been designed to collect the background and peak values for three analog input signals. RATIO is used in the run sequence as an alternative to AVRAGE under the experimental conditions of a resonant/nonresonant ratio. RATIO will store up to 600 records on a DX1: output data file. It stores these records by an unformated write statement. Each record contains the values $\overline{P}_{CARS}$, $\overline{S}_{CARS}$, $\overline{P}_{R/N}$, $\overline{S}_{R/N}$, and $\overline{P}_{FRING}$.

COL123.MAC is used to collect the background and peak values for the three analog input signals. The Fortran control program subtracts the background and constructs the resonant/nonresonant ratio. Like AVRAGE, Robust operator $\hat{R}^{+}$ is available in Ratio.

The remaining functional blocks of Ratio behave the same as AVRAGE with one exception: elements of the data record are stored in arrays and the entire file is written at the end of the program in a DO loop. Ratio has been included in Fortran Library and COL123 is listed in the MACRO Library.

METHANE Q – Branch at 2916 cm^{-1}

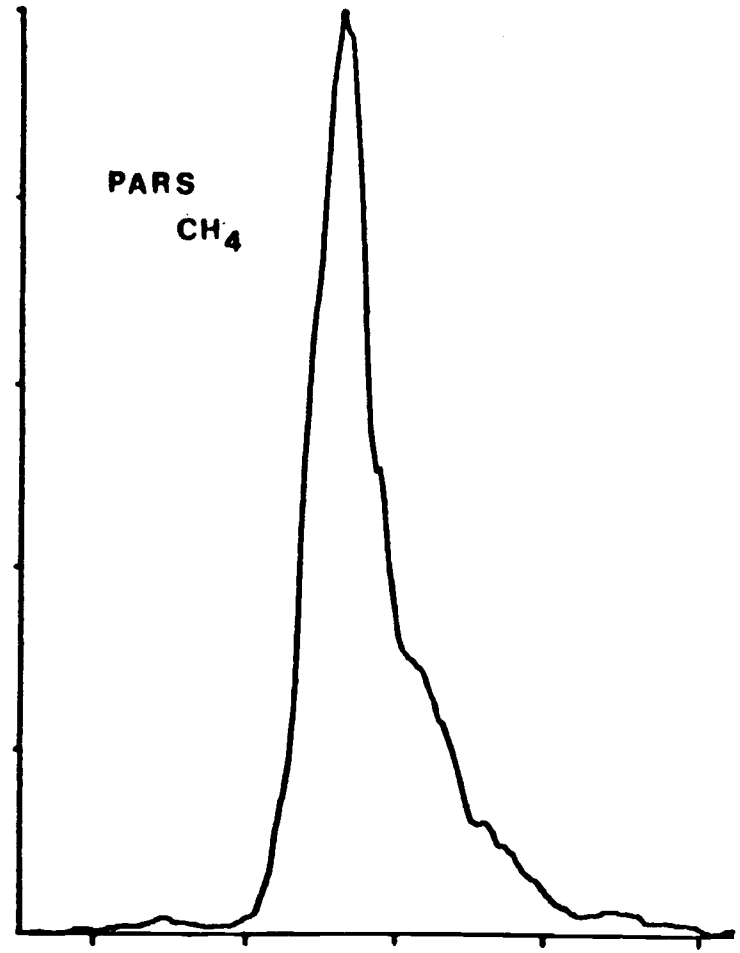
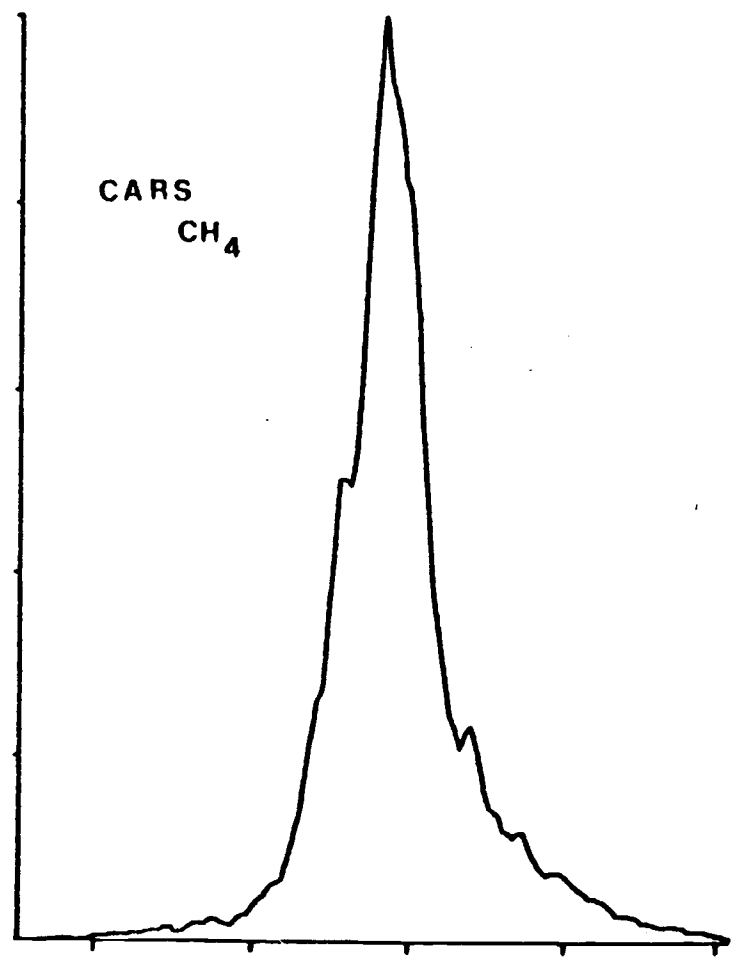


Figure 3.3 $\omega_1 - \omega_2$ cm^{-1}



$\omega_1 - \omega_2$ cm^{-1}

C. Data Processing

After the three wave mixing spectrum has been recorded, the need arises for a program that will process that data. The program we use for this purpose is known as CRUNCH; it occupies the third position in the run sequence. The objective of CRUNCH is to smooth the spectral intensity data and assign a frequency in wavenumbers to each data point in the spectrum. This task is accomplished by a series of Fortran subroutines which are called by the control program OMEGA. Once the spectral data has been processed, the resulting reduced data is recorded in an output data file for later use. Each record of the output file contains the fitted intensities as a function of transition energy, in cm^{-1}.

OMEGA is an interactive Fortran program whose main responsibility is to control the manipulation of data files, common arrays, and fitting parameters used in the data reduction process. The spectral intensities undergo a least squares fit to a seven point, cubic polynomial function using the control subroutine POLFIT given by Bevington [18]. Frequency calibration of the spectrum is obtained by the subroutine FRING which controls the linearization fit to the dye laser fringes. All programs which compose CRUNCH are listed and documented in the Fortran Library. Figure 3.4 shows the schematic representation.

The first function of OMEGA is to open the input data file on DS1: and assign it a logical unit number. Input data is read into

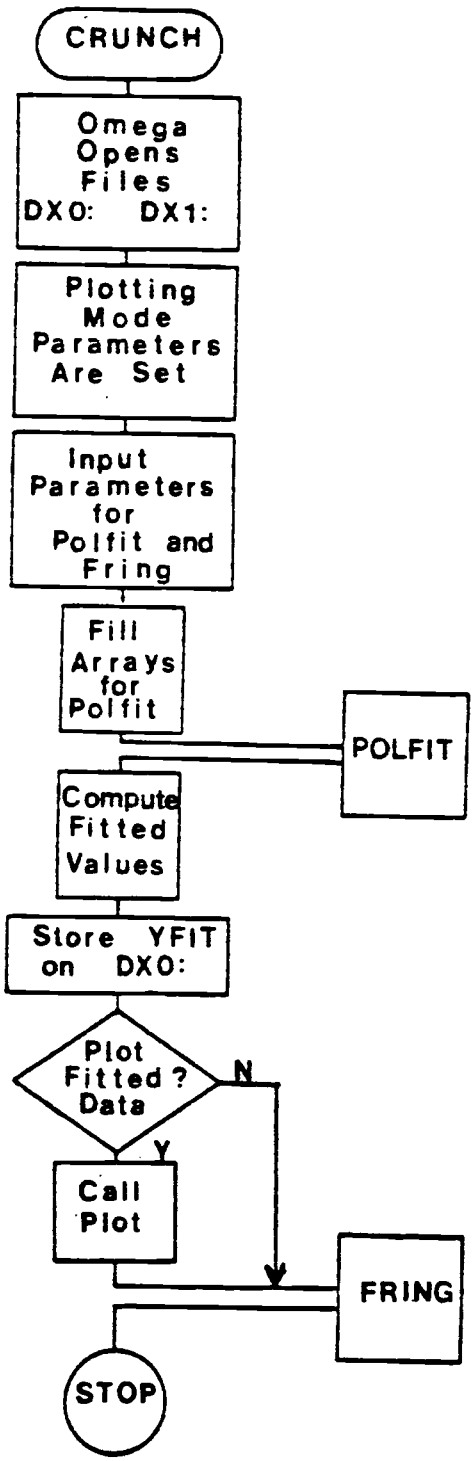


Figure 3.4. Flowchart diagram for program CRUNCH.

core memory as Fortran array elements IY(I), SY(I), and IFR(I). These variables represent $\overline{P}_{CARS}$, S_{CARS}, P_{FRING}, respectively. IY(I) and IFR(I) are searched for minimum and maximum values, which are used to normalize the data. An output data file is allocated disk memory on DX0:, and will contain the processed spectrum for graphic display. The first part of the interactive mode of OMEGA, may allow the program user to plot data on a strip chart recorder as it is being processed. Two plotting modes are designed to accomplish this desire. First, the input data file may be plotted to display the fringes and normalized intensities. The second plot which may be obtained illustrates the normalized fitted intensities of the output file. If either plotting mode is activated, the program user interactively sets the chart scales, and enters a X-axis expansion factor (ISLICE). The raw data is then normalized and plotted.

A weighting mode for POLFIT is then entered into MODE. This variable determines the method used for the calculation of the weighting factor $1/\sigma_i^2$. If MODE is positive σ_i^2 is taken to be the elements of SY(I). A negative value for MODE assumes that $\sigma_i^2 \approx$ IY(I) and zero produces an equal weighting mode. The X(I) and Y(I) data point arrays are then constructed and a call to POLFIT is executed in the form

CALL POLFIT(NPTS, NTERMS, MODE, CHISQR).

POLFIT is used to calculate the coefficients, A(I), for a seven point least squares fit to the polynomial function

$$Y = A(1) + A(2) \cdot X + A(3) \cdot X^2 + A(4) \cdot X^3 \qquad (3.2)$$

These coefficients are returned to OMEGA and then used to compute the seven fitted y values YFIT(I). Fitted spectral intensities are then stored on DXO: and a loop is executed until the entire spectrum is fitted and stored. OMEGA then calls the subroutine FRING to calibrate the data point frequencies. Since we are interested in frequency values representing $\omega_1 \pm \omega_2$, parameters which are passed to FRING include the frequency of ω_1 (in cm^{-1}) and the initial ω_2 wavelength (in angstroms). The program user selects either $\omega_1 + \omega_2$ or $\omega_1 - \omega_2$.

FRINGE is used as a control program to monitor the calibration process and pass parameters to the fitting routines which aid in the calculations. The frequency calibration involves a nonlinear least squares fit on the dye laser fringes, by the subroutine CURFIT (see Bevington for details). The fitting function is a Gaussian peak plus quadratic polynomial of the form

$$Y = A(1) \cdot e^{-z^2/2} + A(4) + A(5) \cdot X + A(6) \cdot X^2 \qquad (3.3)$$

where $z = (X - A(2))/A(3)$. Figure 3.5 shows a typical fringe pattern. Once the frequency is calibrated control is passed back to OMEGA and the output data file on DXO: is closed. Output data records have the FORMAT (F9.3,X,I4) which represents the frequency (in wavenumbers) and the fitted intensities respectively. The spectrum is now ready for graphical display.

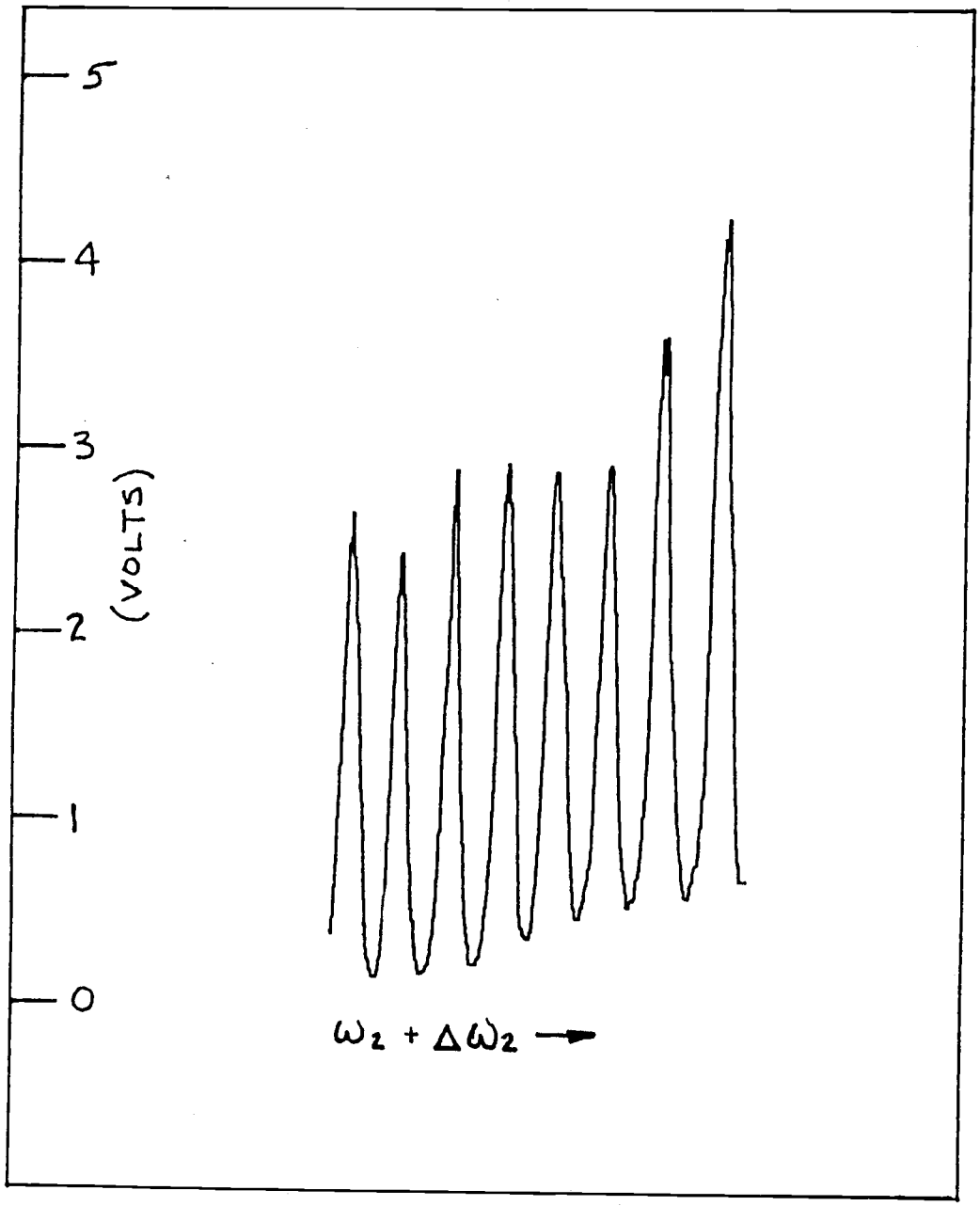


Figure 3.5. Typical dye laser interference (fringe) pattern used to calibrate transition energies.

D. Graphic Display of the Fitted Spectrum

The final element of the run sequence has been designed to display the reduced three wave mixing spectrum as a function of transition energies. The program used for this purpose is known as PLTFIL [19] and serves two main functions. In the insepction mode, it allows the program user to visually inspect the input file at the TEK terminal. Secondly, a X-Y plotting mode displays and labels the spectrum on a X-Y plotter.

The inspection mode has associated with it, the following options:

1. It will isolate and expand any portion of the spectrum and pass that data to the X-Y mode.

2. It will measure and display individual spectral frequencies by the use of cursor crosshairs.

In the X-Y plotting mode final display may exhibit:

1. Tic marked calibrated axis

2. Horizontal and vertical alphanumeric labeling

3. Numerical frequency labeling

All of the spectra illustrated in this thesis have resulted from PLTFIL.

CHAPTER FOUR

TWO PHOTON ELECTRONIC SPECTROSCOPY

A. Introduction

As stated in chapter one of this thesis, the nonlinear polarization of a dielectric substance will become large when the sum of two optical frequencies $\omega_1 + \omega_2$ matches an allowed two photon electronic resonance, ω_{TP}. For this reason optical three wave mixing spectroscopy has found a place in the rapidly growing field of two photon spectroscopy. It has already been shown experimentally that the two photon resonances in the third order susceptibility will yield two photon spectra when $\omega_{TP} = 2\omega_1$ [20]. In this work, we have further developed the application of optical three wave mixing by considering the molecular absorption of two different photons such that $\omega_{TP} = \omega_1 + \omega_2$, where ω_2 is the tunable source. By using a UV frequency for ω_1, states at higher energies are thereby accessible. Successful results on the $\tilde{A}$ state of Nitric Oxide are reported.

Two photon spectroscopy has received considerable attention in recent years because of its ability to offer information pertaining to dipole forbidden transitions into electronic excited states. For example, $g \leftrightarrow g$ ($u \leftrightarrow u$) transitions in molecules possessing a center of symmetry are parity forbidden in one photon spectroscopy. However, McClain has shown that the simultaneous absorption of two photons is possible and has discussed selection rules for this process [21]. When considering the "allowedness" of a two photon transition, we must

examine the total electronic vibrational symmetry [22]. We note that two photon absorption processes are highly dependent on the photon polarization [23]. By selectively choosing the polarization of the second photon, relative to the first, we can gain information about the symmetry of the two photon resonance state, and may further provide a means of isolating excited states from other nearby states of different symmetry. We shall illustrate such an application in the final section of this chapter.

Several techniques are available to the experimentalist to probe the allowed two photon transitions in a molecule. Direct two photon absorption is extremely difficult, but has the distinct advantage that it yields absolute values for the two photon absorption coefficient δ_{TP}. For experiments involving two laser beams P_1 and P_2, the absorption strength is defined by [21]

$$-\Delta P_1 = -\Delta P_2 = P_1 \cdot P_2 \cdot \delta_{TP} \cdot C \cdot L \cdot A^{-1} \qquad (4.1)$$

where P_1 and P_2 are the laser powers [photons/sec], C is the sample concentration, L is the path length and A is the beam area. A typical value for δ_{TP} is about 10^{-50} $cm^4 \cdot sec \cdot photon^{-1} \cdot molecule^{-1}$.

Other methodology is available in which two photon absorption processes are described by the result of absorption. These indirect methods have the advantage that fewer absorption events can be detected, allowing for greater sensitivity than direct absorption. However, indirect measurements may yield only relative values for δ_{TP}.

Moderate resolution, two photon excitation fluorescence has been applied to Nitric Oxide at pressures as low as 50 m torr [24]. This study has demonstrated the occurrence of O ($\Delta J = -2$) and S ($\Delta J = +2$) branches seen only with the two photon absorption process and the frequencies agree well with the calculated transition energies. It should be realized that, even if two photon absorption occurs, the one photon fluorescence from the excited state may be symmetry forbidden. Further, low quantum yields and nonradiative energy transfer can create experimental difficultues in detecting the one proton fluoresence.

In the case of nonradiative or thermal relaxation, the two photon absorption event is detectable using photoacoustic techniques. This method also offers high sensitivity when no one photon absorption process interferes. Another advantage to photoacoustic spectroscopy is that detection is independent of the large background of scattered light usually present in laser experiments, thereby increasing the signal/noise ratio.

Once a molecule is excited into a higher electronic state it is still acted upon by other incident photons of the radiation field. The molecule can absorb one or more additional photons until it is removed from the resonant state by ionization, dissociation, or other possible photochemical processes. For many excited electronic states the most probable fate in such an intense field is ionization [25], yielding a free electron whose production as a function of ω_2 can characterize the two photon state. Multiphoton Ionization

Spectroscopy (MPI) of this type has been used to observe two photon transitions in trans butadiene [26] and to aid in the study of the $^1E_{2g}$ state of benzene [27]. Although MPI techniques are often destructive and usually not so selective, they can offer high sensitivity and experimental simplicity.

Finally, we consider the detection of a two photon resonance in the third order susceptibility via the optical three wave mixing process. We shall examine the form of χ_{TP} in the next section but note here that, as long as χ_{TP} remains large compared to χ_{NR}, this technique offers several distinctive advantages. The virtue of the three wave mixing process lies in the fact that two photon transitions occur through an intermediate state in the mixing process: the molecule only serves as a mixing medium and the measured signal does not require a change in the ground state population. It is therefore a direct and nondestructive technique which may be employed to obtain high resolution ($<.5\ cm^{-1}$) two photon electronic spectra at gas pressures less than 100 torr [28]. As shown in chapter two of this thesis, three wave signal beams have unusually large experimental uncertainty in their amplitude. However, this does not detract from the effectiveness of this technique for assigning transition energies and J quantum numbers.

It is our intent to assess the experimental feasibility of optical three wave mixing spectroscopy in the study of electronically excited states when $\omega_{TP} = \omega_1 + \omega_2$ and ω_1 = 355 nm in the UV at 28185 cm^{-1}. In addition to the added consideration needed in the

experimental mixing of UV radiation, it should be realized that both the spatial (S(q)) and the time (T(t)) overlap functions now involve three pulses. The first objective of this study is therefore, to evaluate the experimental "sensitivity" in the detection and the frequency assignment of two photon electronic transitions. We have employed the well documented $\tilde{X}^2\pi_{\frac{1}{2}} \rightarrow \tilde{A}^2\Sigma$ transition of Nitric Oxide as the experimental test case and use the computer program run sequence to aid in the measurement of transitions energies at $\omega_1 + \omega_2$. A second investigation is conducted to determine the sensitivity of the dipole selection rules on $P_2(\omega_2)$ polarization changes. In these experiments we have used $\omega_1 = 28185\ cm^{-1}$ in combination with red photons to access two photon transitions near $44200\ cm^{-1}$ (226 nm). Results of this analysis are given in Section D and future applications of two photon-three wave mixing spectroscopy are suggested in Section E of this chapter.

B. Two Photon Resonances in $\chi^{(3)}_{TOT}$

One of our main interests in the application of the two photon-three wave mixing process is in the assessment of the magnitude of χ_{TP} compared to χ_{CARS} and to χ_{NR}. The total third order susceptibility representing the contributions from all molecular resonances can be written as

$$\chi^{(3)}_{TOT} = \chi_{NR} + \chi'_{CARS} + i\chi''_{CARS} + \chi'_{TP} + i\chi''_{TP} \qquad (4.2)$$

In the consideration of the CARS susceptibility, we had assumed that the intermediate mixing states did not involve two photon resonances at $\omega_1 + \omega_2$. Similarly, here we shall consider only resonant mixing that is independent of $\omega_1 - \omega_2$ processes and note that one must be careful to assure this in an actual experiment. Thus, we can reduce equation (4.2) to yield

$$\chi^{(3)}_{TOT} = \chi_{NR} + \chi_{TP} \tag{4.3}$$

Here, χ_{TP} actually involves a sum over all two photon resonances but, as before, we consider only a single transition to be important. From [29] and [30], we can obtain an expression relating χ_{TP} to δ_{TP} (defined in equation 4.1) which shows

$$\chi_{TP} = \frac{NC^2 n_1 n_2 \Gamma_{TP}}{32\pi^3 h\omega_1\omega_2} (\delta_{TP}) \frac{1}{\omega_{TP}-(\omega_1+\omega_2)-i\Gamma_{TP}} \tag{4.4}$$

The power generated in the mixing process at $\omega_{TW} = \omega_1 + \omega_2 - \omega_3$ is then

$$P_{TP} = K \cdot P_1 \cdot P_2 \cdot P_3 \cdot N^2 \cdot |\chi_{NR} + \chi_{TP}|^2. \tag{4.5}$$

Since we are probing levels in an excited electronic state in the UV region, nonresonant intermediate mixing states near ω_1 and ω_{TP} are closer (in energy) to resonance states. This then may produce a larger contribution by χ_{NR} than for the CARS process, a concern we wished to test experimentally.

C. Experimental Considerations

The experimental apparatus used in this work is changed only by the addition of a 355 nm beam in the mixing process. A second beam combiner is placed between the sample cell and the green-red beam combiner, to colinearily mix ~10 mj of UV taken from the dye laser amplifier beam. Quartz lenses are used to reduce the possibility of interference caused by UV induced fluorescence. This is an important consideration since we are detecting a frequency which is shifted to the red of the 355 nm beam. The three mixing beams (UV, green and red) are commonly focused through a pinhole drilled by the UV to assure maximum spatial overlap. Inserted after the sample cell is a yellow filter designed to reduce the UV light level along the optical axis. The dye used for the $\tilde{A}$ state of NO is Rhodamine 640/ETOH whose concentration was optimized for power at ω_2 = 625 nm (red) and is approximately 2.0×10^{-3} M.

Experimentally it was not feasible to use sample/reference legs to give a resonant/nonresonant ratio because of the special dichroics required by the presence of the UV beam. However, to reduce the standard deviation in the signal, we did employ the $\hat{R}^{+}$ Robust operator via the program AVRAGE, which is used to collect the data. This data is then processed using the run sequence programs to yield the measured transition energies at $\omega_1 + \omega_2$.

D. Results

Experimentally, it was pleasing to find that the presence of UV radiation did not impose on the relative efficiency of the apparatus or in the determination of $\bar{P}_{TW}$. However, we have found that this experiment is very sensitive to proper spatial overlap of the three beams. To ensure this, the CARS signal of CH_4 (ω_{Ram} = 2916 cm^{-1}) was used to properly align the green and red beams within the focal volume defined by the green beam. Once this spatial overlap had been optimized, the UV was introduced and adjusted for maximum overlap by monitoring the two photon signal. This stepwise type of procedure is strongly recommended when the spatial overlap function is dependent on three pulses.

Spectra obtained for NO at 140 torr are shown in Figures 4.1 to 4.4 and demonstrate that we have been successful in obtaining high resolution spectra corresponding to the $\tilde{X}\ ^2\pi_{\frac{1}{2}}$ (V=0) $\rightarrow$ $\tilde{A}\ ^2\Sigma$ (v = 0) transition. Transition energies were determined and some J quantum number assignments have been made using previous calculations by Bray et al. [24]. In general, the signal levels were smaller than for CARS spectra but not by more than a factor of 10. X_{NR} was not significant in these spectra.

A polarization study was conducted by rotating the polarization of $P_2(\omega_2)$ and scanning across two adjacent J states. As shown in the inset of Figure 4.1, the two photon transition $\pi \rightarrow \Sigma$ requires that the polarization of ω_1 and ω_2 be orthogonal and that the signal beam

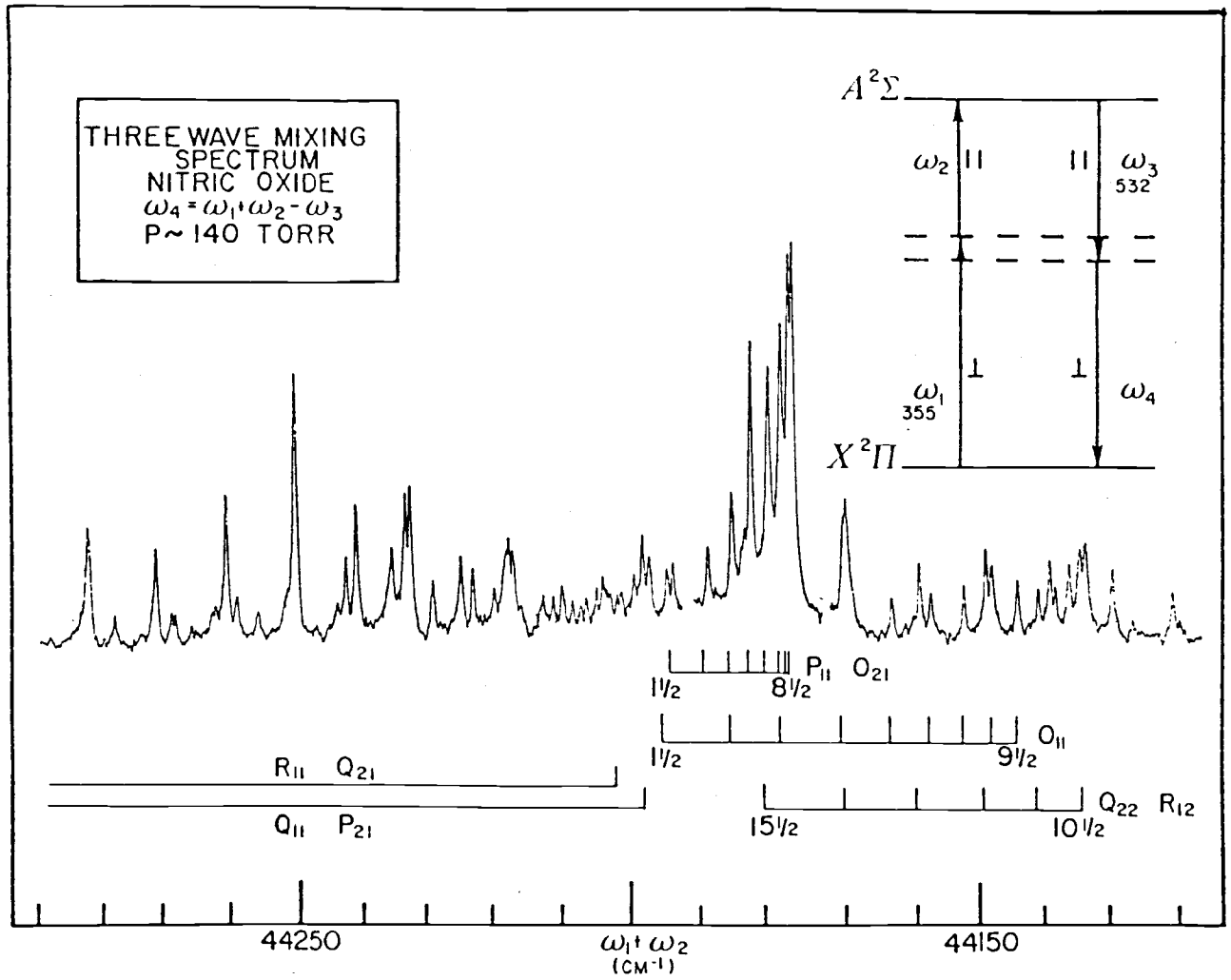


Figure 4.1. Three wave mixing spectrum of Nitric Oxide.

$\omega_4 = \omega_{TW}$ have a polarization perpendicular to that of ω_3. Although this was found to be the case, the ω_{TW} intensity when ω_1 and ω_2 were made parallel was only reduced by a factor of ~3, much less than might be expected. This is believed to be largely caused by poor linear polarization of ω_1 and ω_2 due to scrambling characteristics of the dichroics and other optical elements. Nonetheless, the $\pi \rightarrow \Sigma$ character of the transition is clearly established.

Finally, the spectra demonstrate the power and effectiveness of the computer run sequence in recording an entire spectrum in pieces without loss in the accuracy of determining the transition energies. Figures 4.2, 4.3 and 4.4 show the spectrum of Figure 4.1 is such a division. The quality of these spectra is excellent and the digital smoothing routines permit resolution of even low intensity transitions.

E. Possible Future Applications

In the previous section, we have shown that the two photon-three wave mixing process involving resonances at $\omega_{TP} = \omega_1 + \omega_2$ is not only experimentally feasible but is also fairly sensitive in the detection of $\bar{P}_{TP}$. In addition, transition energies may be readily measured. For these reasons, we feel that this technique holds considerable promise in future application of two photon electronic spectroscopy.

One of the most appealing attributes of the two photon-three wave mixing process is the potential for determining vacuum UV ($\omega_{TP} >$ 50,000 cm^{-1}) transition energies using visible and UV lasers. In

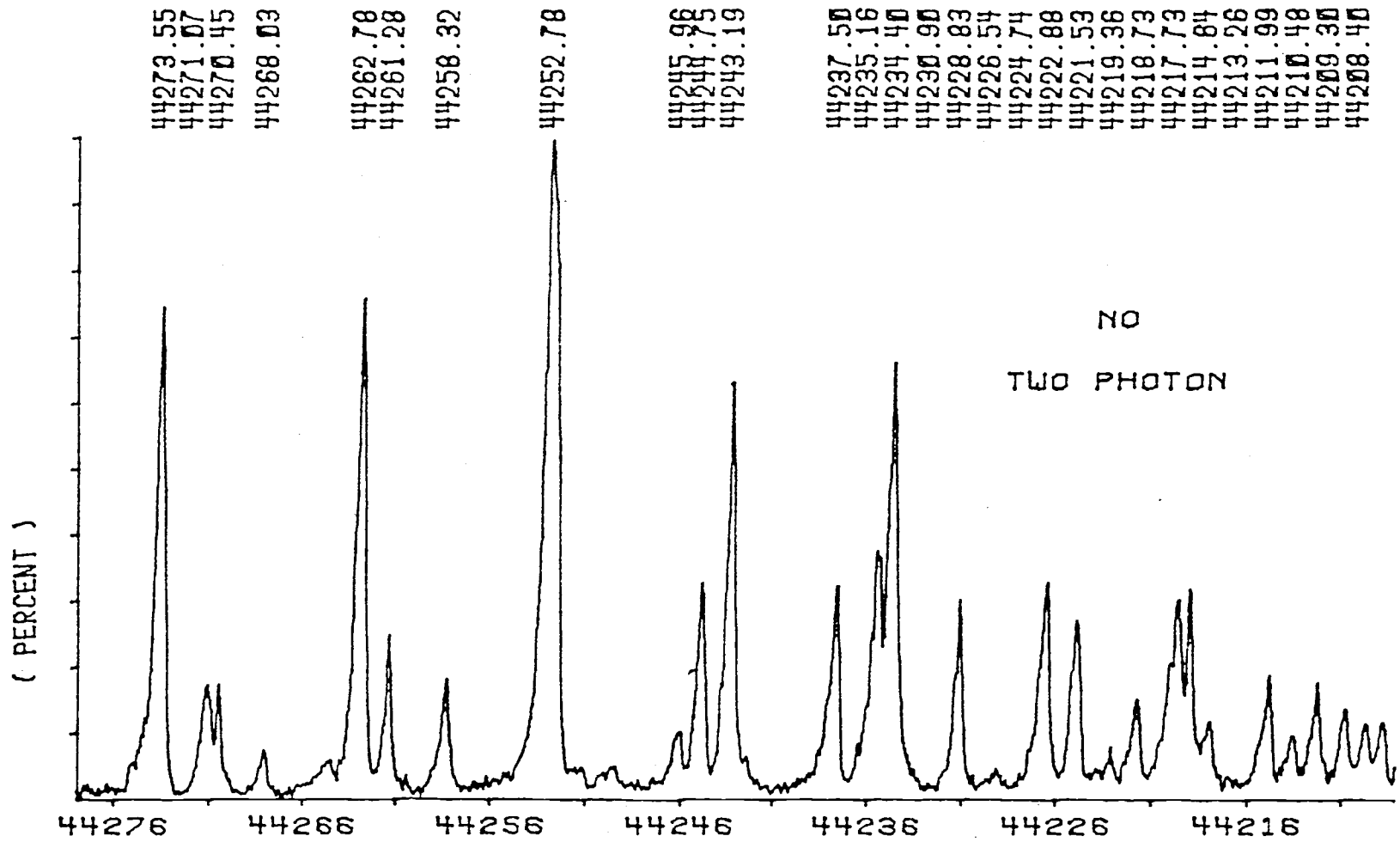


Figure 4.2. Two photon spectrum.

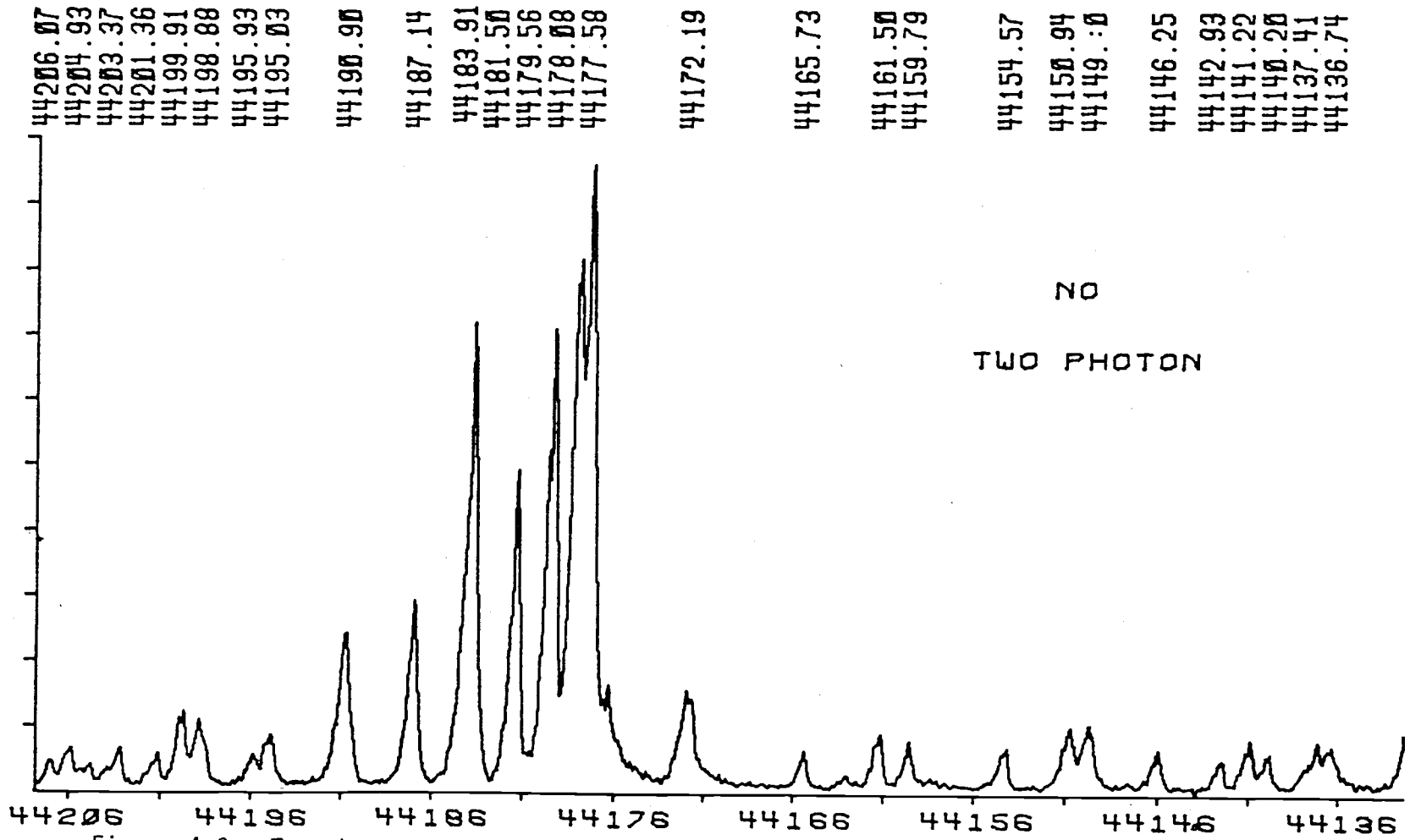


Figure 4.3. Two photon resonances of NO.

p.80 missing from original. Author unavailable to supply.

practice, this is only experimentally possible when $\omega_1 \geq \omega_2$. Two possible candidates for such application, whose two photon transitions have been well documented [26,27] using MPI techniques, are benzene ($\omega_{TP} \simeq 51{,}020\ cm^{-1}$) and trans-butadiene ($\omega_{TP} \simeq 50{,}112\ cm^{-1}$). Both of these transitions can be accessed using ω_1 = 355 nm in combination with tunable blue photons (~440 nm). Alternatively, one can use ω_1 = 266 nm (the fourth harmonic) plus tunable red photons. This permits one to access higher energies but, in some preliminary work on benzene using this scheme, fluorescence caused by ω_{266} proved to be overwhelming. Better spectral discrimination or possible use of the 1064 nm Nd:YAG beam as ω_3 (to blue shift ω_4 relative to ω_{266}), may be necessary to minimize this problem.

A second important application of the three wave technique lies in its ability to selectively isolate close lying electronic states of different symmetry, by altering incident photon polarizations. The $\tilde{C}\ ^2\pi$ and $\tilde{D}\ ^2\Sigma$ states of Nitric Oxide lie within 800 cm^{-1} of each other and are recommended as a most interesting test case since both of these states lie in the vacuum UV.

REFERENCES

1. W. M. Tolles, J. W. Nibler, J. R. McDonald, and A. B. Harvey, J. Appl. Spectrose. 31, 253 (1977).

2. J. W. Nibler, Private communication (1980).

3. H. P. Trommsdorff, R. M. Hochstrasser, and G. R. Meredith, J. Luminescence 18/19, 687 (1979).

4. J. W. Nibler and G. V. Knighten, "Coherent Anti-Stokes Raman Spectroscopy", Topics in Current Physics, edited by A. Weber, Vol. 11, Chap. 7, Springer-Verlag, Berlin (1979).

5. J. A. Zebrowski and J. W. Nibler, current data on file (1981).

6. W. J. Rey, "Robust Statistical Methods", Lecture Notes in Mathematics, edited by A. Dold and B. Eckmann, No. 690, Springer-Verlag, Berlin (1978).

7. G. Herzberg, K. P. Huber, Molecular Spectra and Molecular Structure, Van Nostrand, Princeton (1979), Vol. IV.

8. D. M. Guthals, Ph.D. Thesis (1981).

9. D. M. Guthals and J. W. Nibler, Opt. Commun. 29, 322 (1979).

10. Quanta-Ray "Pulsed Dye Laser Users Manual" (1978).

11. D. P. Shoemaker, C. W. Garland, and J. I. Steinfeld, Experiments in Physical Chemistry, McGraw-Hill, N.Y. (1974).

12. W. M. Tolles and R. D. Turner, J. Appl. Spectrose. 31, 96 (1977).

13. J. W. Nibler, Private Communication (1981).

14. Robustness in Statistics, edited by R. L. Lavner and G. N. Wilkinson, Academic Press, N.Y. (1979).

15. P. J. Huber, "Robust Statistica Procedures", Soc. Indust. Appl. Math., No. 27, J. W. Arrowsmith Ltd., Bristol (1977).

16. M. H. Quenoville, Biometrika, 43, 353 (1956).

17. DEC, "RX01 users reference manual".

18. P. R. Bevington, Data Reduction and Error Analysis for the Physical Sciences, McGraw-Hill, N.Y. (1969).

19. T. Lundeen, "Program Manual", (1981).

20. R. M. Hochstrasser, G. R. Meredith, and H. P. Trommsdorff, Chem. Phys. Lett. 53, 423 (1978).

21. W. M. McClain, Acc. Chem. Resh. 7, 129 (1974).

22. R. M. Hochstrasser, H. N. Sung, and J. E. Wessel, Chem. Phys. Lett. 24, 16 (1974).

23. R. G. Bray and R. M. Hochstrasser, Mol. Phys. 31, 1199 (1975).

24. R. G. Bray, R. M. Hochstrasser, and J. E. Wessel, Chem. Phys. Lett. 27, 167 (1974).

25. P. M. Johnson, Acc. Chem. Resh. 13, 20 (1980).

26. V. Vaida, R. E. Turner, J. L. Casey, and S. D. Colson, Chem. Phys. Lett. 54, 25 (1978).

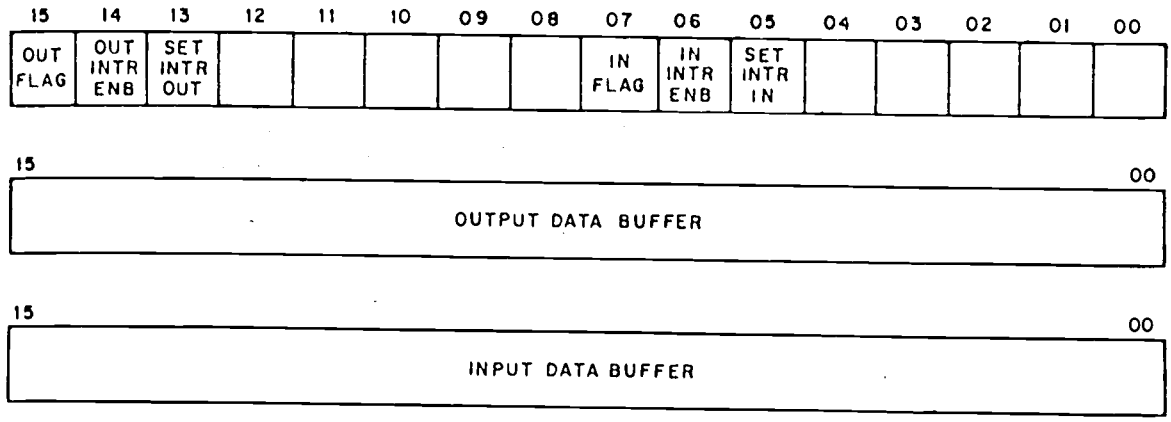
27. V. Vaida, M. B. Robin, and N. A. Kuebler, Chem. Phys. Lett. 58, 557 (1978).

28. J. A. Zebrowski, and J. W. Nibler, current data on file (1980).

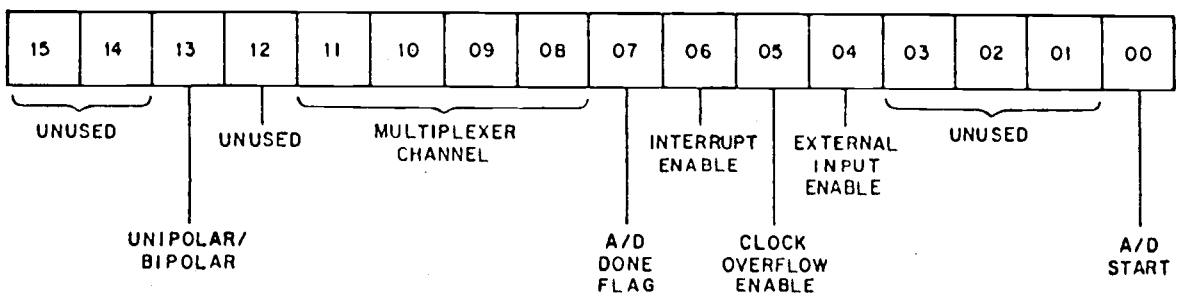
29. R. J. Anderson, G. R. Holtom, and W. M. McClain, J. Chem. Phys. 66, 3832 (1977).

30. J. H. Bechtel and W. L. Smith, Phys. Rev. B 13, 3515 (1976).

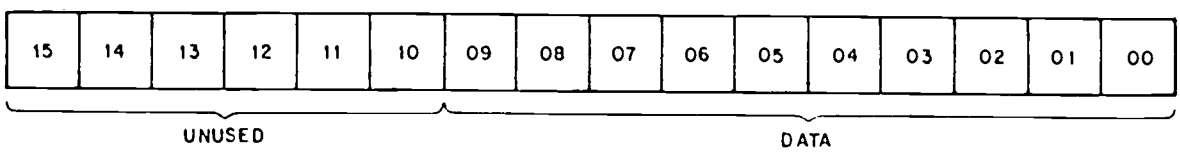
Appendix I: AR11 and DR11-K Program Reference Guide



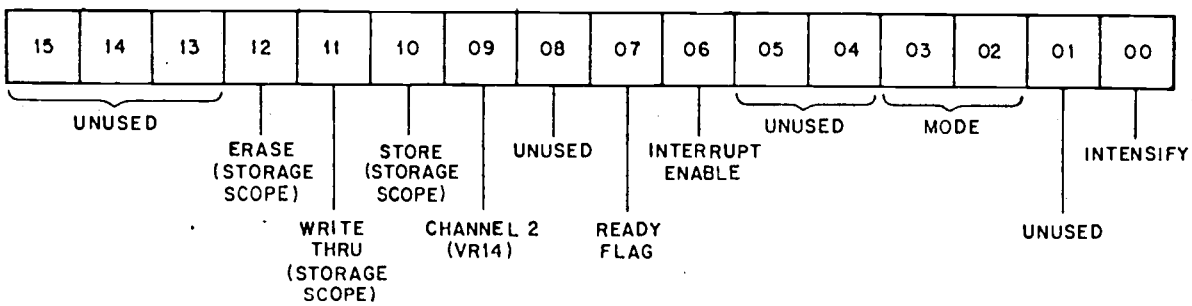
DR11-K Register Assignments



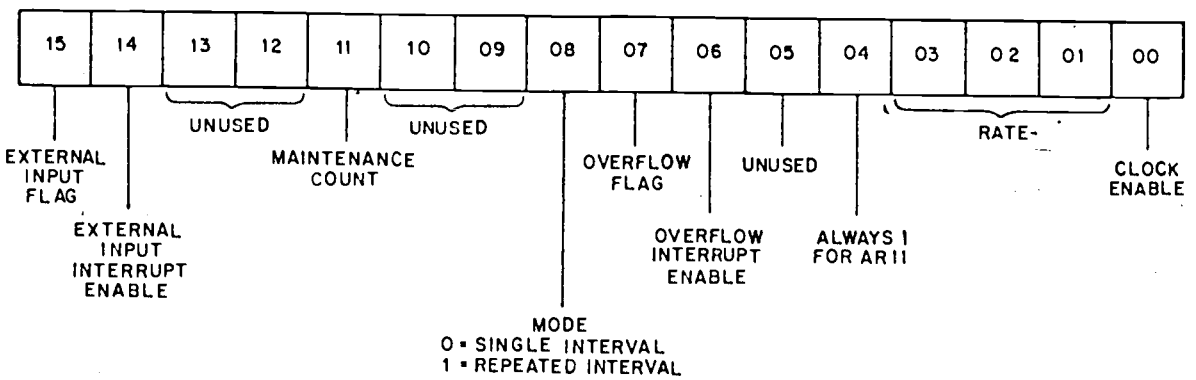
A/D Status Register



A/D Buffer Register



Display Control Status Register



Clock Status Register

APPENDICES

Appendix II: Macro Library

This Library contains the following documented listings:

COLTWO.MAC
PLTTWO.MAC
STEP.MAC
COL123.MAC
PLOT.MAC

These subroutines are general purpose routines which are Fortran compatible using the .GLOBL label and are program accessible through the Fortran CALL statement.

```
 GLOBL  COLTWO
R0=%0
R1=%1
R2=%2
R3=%3
R4=%4
R5=%5
R6=%6
PC=%7
DSR=170410
CSR=170400
DBR=170402
SR=167770
DR=167772
CH00EX=020020
CH00=020001
FRING=021401
FRINGX=021420
COLTWO: TST(R5)+                ;NO NEED TO SAVE # OF ARGUMENTS
        MOV@(R5)+,R0            ;10 OR 12 BIT A/D ?
        MOV#DONE,@#300          ;LOAD 12 BIT INTERUPT VECTOR
        MOV#340,@#304           ;PRIORITY 6
        CMP R0,#12.             ;GO TO 12 BIT PART IF EQUAL
        BEQ TWL
MORE:   BIS#011000,@#DSR        ;ALLOW EXT START ON ERASE RETURN
        BIC#2000,@#DSR
QQ:     TSTB@#DSR               ;WAIT FOR BACKGROUND TRIGGER
        BPL QQ
        MOV#FRING,@#CSR         ;BCKGND ON FRINGES
C:      TSTB@#CSR               ;WAIT FOR 10 BIT FINISH
        BPL C
        MOV@#DBR,R2             ;STORE FRING BCKGND IN REGISTER
        MOV#CH00,@#CSR          ;10 BIT SAMPLE ON W3 BACKGND
B:      TSTB@#CSR               ;WAIT FOR DONE BIT
        BPL B
        MOV@#DBR,R1             ;STORE W3 BCKGND
        MOV#CH00EX,@#CSR        ;SAMPLE W3 ON CH 00 A/D EXT START
TTST:   TSTB@#CSR               ;A/D DONE?
        BPL TTST
        MOV@#DBR,R0
        CMP R0,#1023.           ;TEST TO SEE IF FULL SCALE
        BEQ MORE                ;IF SO DON'T KEEP
        MOV#FRING,@#CSR         ;SAMPLE FRINGES
G:      TSTB@#CSR               ;WAIT FOR DONE BIT
        BPL G
        MOV@#DBR,R3             ;MOVE TO REGISTER
        CMP R3,#1023.           ;TEST FULL SCALE SATURATION
        BEQ MORE                ;IF SO GET ANOTHER
        MOV R2,@(R5)+           ;STORE BCK FRINGES
        MOV R1,@(R5)+           ;STORE BCK W3
        MOV R0,@(R5)+           ;STORE W3
        MOV R3,@(R5)+           ;STORE FRINGES
        RTS PC
TWL:    BIS#011000,@#DSR        ;ALLOW EXT START ON ERA RTN L
        BIC#2000,@#DSR
TA:     TSTB@#DSR               ;WAIT FOR TRIGGER PULSE
        BPL TA
        MOV#FRING,@#CSR         ;10 BIT SAMPLE ON FRINGE BCKGND
TB:     TSTB@#CSR               ;WAIT FOR DONE BIT
        BPL TB
*
```

```
$
        MOV@#DBR,R1             ;SAVE IN REGISTER
        MOV@#DR,R0              ;START 12 BIT A/D ON W3 BCKGND
        MOV#100,@#SR            ;ENABLE INTERUPT
        WAIT                    ;FOR THE INTERUPT
        MOV@#DR,R2              ;SAVE IN REGISTER
        MOV#FRINGX,@#CSR        ;10 BIT EXT SAMPLE ON FRING
TC:     TSTB@#CSR               ;WAIT FOR DONE BIT
        BPL TC
        MOV@#DR,R0              ;12 BIT SAMPLE ON W3
        MOV#100,@#SR            ;ENABLE INTERUPT
        WAIT                    ;FOR INTERUPT
        MOV@#DR,R3              ;SAVE IN REGISTER
        CMP R3,#4095.           ;TEST FOR FULL SCALE SATURATION
        BEQ TWL                 ;IF SO TAKE ANOTHER
        MOV@#DBR,R4             ;SAVE FRINGES IN REGISTER
        CMP R4,#1023.           ;TEST FOR FULL SCALE
        BEQ TWL                 ;IF SO TAKE ANOTHER
        MOV R1,@(R5)+           ;STORE BCK FRINGES
        MOV R2,@(R5)+           ;STORE BCK W3
        MOV R3,@(R5)+           ;STORE W3
        MOV R4,@(R5)+           ;STORE FRINGES
        RTS PC
DONE:   NOP                     ;INTERUPT ROUTINE
        RTI
.END COLTWO
*
```

```
.GLOBL  PLTTWO
R0=%0
R5=%5
PC=%7
XBUF=170412
YBUF=170414
CLKSR=170404
CLKBR=170406
DSR=170410
PLTTWO: TST(R5)+                ;PASS THE NUMBER OF ARGUMENTS
        BIS#11000,@#DSR         ;NO PULSES OUT--PLEASE!
        BIC#2000,@#DSR
        BIS#4000,@#DSR          ;NO PULSE OUT ON WRITE THRU
        MOV#CHART,@#344         ;LOAD CLOCK INTER. VECTOR
        MOV#340,@#346           ;PRIORITY 6
        BIC#1,@#CLKSR
        MOV#5.,R0
        NEG R0
        MOV R0,@#CLKBR          ;LOAD THE COUNTER
        MOV#111,@#CLKSR         ;START THE CLOCK
        MOV@(R5)+,@#YBUF        ;PLOT THE AVERAGE OF W3
        MOV@(R5)+,@#XBUF        ;PLOT FRINGES
        WAIT                    ;FOR CLOCK INTERUPT
        RTS PC
CHART:  BIC#4000,@#DSR          ;PULSE OUT FOR RECORDER ON WRITE THRU
        CLR@#CLKSR              ;DISABLE INTERUPT
        BIS#4000,@#DSR
        RTI
.END PLTTWO
*
```

```
.GLOBL STEP
R0=%0
R5=%5
PC=%7
DSP=170410
STEP:TST(R5)+    ;PASS THE NUMBER OF ARGUMENTS
BIS#1000,@#DSR   ;PULSE OUT TO STEPPER MOTOR
BIC#1000,@#DSR   ;PULSE DONE--NOT VERY WIDE
AGN:TSTB@#DSR
BPL AGN
RTS PC
.END STEP
*
```

```
 .GLOBL COL10
R0=%0
R1=%1
R2=%2
R3=%3
R4=%4
R5=%5
R6=%6
R7=%7
DSR=170410
CSR=170400
DBR=170402
SR=167770
DR=167772
XBUF=170412
YBUF=170414
CH00EX=002020
CH00=020001
CH01=020401
CH02=021001
FRING=021401
CH01EX=020420
COL10:TST(R5)+              ;NO NEED TO SAVE # OF ARGUMENTS
MOV#015000,@#DSR              ;ENABLE INTERUPT
GG:TSTB@#DSR
BPL GG
MOV#FRING,@#CSR         ;BCKGND ON FRINGES
C:TSTB@#CSR
BPL C
MOV@#DBR,@(R5)+           ;STORE FRING BCKGND
MOV#CH02,@#CSR            ;BCKGND ON W2
D:TSTB@#CSR
BPL D
MOV@#DBR,@(R5)+          ;STORE W2 BCKGND
MOV#CH01,@#CSR             ;10 BIT SAMPLE ON W1
A:TSTB@#CSR                     ;WAIT FOR DONE BIT
BPL A
MOV@#DBR,@(R5)+                     ;STORE W3 BCKGND
MOV#CH00,@#CSR                  ;10 BIT SAMPLE ON W3
B:TSTB@#CSR                     ;WAIT FOR DONE BIT
BPL B
MOV@#DBR,@(R5)+                     ;STORE W3 BCKGND
MORE:MOV#CH00EX,@#CSR               ;SAMPLE W3 ON CH 00 A/D EXT START
TTST:TSTB@#CSR                  ;A/D DONE?
BPL TTST
MOV@#DBR,R0
CMP R0,#1023.              ;TEST TO SEE IF FULL SCALE
BEQ MORE                  ;IF SO DON'T KEEP
MOV#CH01,@#CSR            ;SAMPLE ON CH01 FOR W1
AGN:TSTB@#CSR           ;WAIT FOR DONE BIT
BPL AGN
MOV@#DBR,R1
CMP R1,#1023.
BEQ MORE
MOV#CH02,@#CSR          ;SAMPLE W2
F:TSTB@#CSR
BPL F
MOV@#DBR,R2
CMP R2,#1023.
BEQ MORE
*
```

```
MOV#FRING,@#CSR        ;SAMPLE FRINGES
G:TSTB@#CSR
BPL G
MOV@#DBR,R3
CMP R3,#1023.
BEQ MORE
MOV R0,@(R5)+          ;STORE THE SIGNAL IN AN ARRAY
MOV R1,@(R5)+          ;STORE W1 IN ARRAY
MOV R2,@(R5)+          ;STORE W2
MOV R3,@(R5)+          ;STORE FRINGES
RTS PC
.END COL10
*
```

```
.GLOBL  PLOT
R0=%0
R5=%5
PC=%7
XBUF=170412
YBUF=170414
CLKSR=170404
CLKBR=170406
DSR=170410
PLOT:   TST(R5)+            ;PASS THE NUMBER OF ARGUMENTS
        MOV#5000,@#DSR      ;NO PULSES OUT--PLEASE!
        MOV#CHART,@#344     ;LOAD CLOCK INTER. VECTOR
        MOV#340,@#346       ;PRIORITY 6
        MOV#3.,R0
        NEGB R0
        MOVB R0,@#CLKBR     ;LOAD THE COUNTER
        MOV#113,@#CLKSR     ;START THE CLOCK
        MOV@(R5)+,@#XBUF    ;PLOT AVEW3
        MOV@(R5)+,@#YBUF    ;PLOT THE RATIO
        WAIT                ;FOR CLOCK INTERUPT
        RTS PC
CHART:  MOV#3000,@#DSR      ;PULSE OUT FOR RECORDER ON WRITE THRU
        CLR@#CLKSR          ;DISABLE INTERUPT
        MOV#5000,@#DSR
        RTI
.END PLOT
*
```

```
.GLOBL CLOCK
R0=%0
R5=%5
PC=%7
XBUF=170412
YBUF=170414
CLKSR=170404
CLKBR=170406
DSR=170410
CLOCK:  TST(R5)+                ;PASS THE NUMBER OF ARGUMENTS
        MOV#5000,@#DSR          ;NO PULSES OUT--PLEASE!
        MOV#CHART,@#344         ;LOAD CLOCK INTER. VECTOR
        MOV#340,@#346           ;PRIORITY 6
        BIC#1,@#CLKSR
        MOV#5.,R0
        NEG R0
        MOV R0,@#CLKBR          ;LOAD THE COUNTER
        MOV#111,@#CLKSR         ;START THE CLOCK
        MOV@(R5)+,@#XBUF        ;PLOT AVEW3
        MOV@(R5)+,@#YBUF        ;PLOT THE RATIO
        WAIT                    ;FOR CLOCK INTERUPT
        RTS PC
CHART:  MOV#3000,@#DSR          ;PULSE OUT FOR RECORDER ON WRITE THRU
        CLR@#CLKSR              ;DISABLE INTERUPT
        MOV#5000,@#DSR
        RTI
.END CLOCK
*
```

Appendix III. Fortran Library

Contained in this Library are the documented listings of the Computer Program Run Sequence, with the exception of PLTFIL. The .SAV extension indicates the program RUN name. Fortran control programs (.MAIN.) are listed with their associated subroutines in the following order:

TWEEK.SAV

    TWEEK.FOR(.MAIN.)

AVRAGE.SAV

    AVRAGE.FOR(.MAIN.)

RATIO.SAV

    RATIO.FOR(.MAIN.)

CRUNCH.SAV

    OMEGA.FOR(.MAIN.)
    POLFIT.FOR
    FRING.FOR
    CURFIT.FOR
    MATINV.FOR

These programs are compiled and linked under the direction of the RT-11 V04-B monitor system.

TWEEK.SAV

```
C       TWEEK HAS BEEN DESIGNED TO SERVE AS THE FIRST
C       ELEMENT OF THE PROGRAM RUN SEQUENCE. ITS MAIN
C       FUNCTION IS TO COMPUTE AND DISPLAY THE AVERAGE
C       AND STANDARD DEVIATION FOR N SHOTS COLLECTED IN
C       M DATA GROUPS. A RUNNING TOTAL FOR THE AVERAGE
C       AND DEVIATION IS ALSO DISPLAYED.
C
      INTEGER*2 SHOTS,RANGE
      DATA YES,NO,ALL/'Y','N','A'/
      DOUBLE PRECISION SQRT,ABS,EXP
      REAL*8 NO,YES
      IMPLICIT INTEGER*4(B,F,W)
      IMPLICIT REAL*8(T,S,R,C,A,D)
      DIMENSION AVEW3(200),AVENR(200),AVEPOW(200),DEV(200)
      DIMENSION CW3(200),CNR(200),NAMR(3)
C
C       A FILE MAY BE CREATED TO SAVE THE TOTAL NUMBER
C       OF COLLECTED DATA POINTS
C
      TYPE *,'DO YOU WANT TO SAVE A DATA FILE ?'
      ACCEPT 62,ANS01
      IF(ANS01.EQ.NO)GO TO 25
      TYPE *,'ASSIGN A FILE NAME  NAME.DAT'
      TYPE *,' '
      ACCEPT 3,NAMR
3     FORMAT(3A2)
      OPEN(UNIT=12,NAME=NAMR)
25    CONTINUE
C
C       SET THE AVERAGING PARAMETERS
C
      TYPE 15
15    FORMAT(X,'HOW MANY DATA GROUPS DO YOU WANT ?')
      ACCEPT 20,RANGE
20    FORMAT(I4)
      TYPE 5
5     FORMAT(X,'SELECT THE NUMBER OF SHOTS/GROUP TO BE AVERAGED')
      ACCEPT 10,SHOTS
10    FORMAT(I5)
      IF(ANS01.EQ.NO)GO TO 1
      BTOT=RANGE*SHOTS
      WRITE(12,*)BTOT
C
C       SCALE THE RATIO
C
1     TYPE 63
63    FORMAT(X,'SET AN INTENSITY SCALE FACTOR FOR THE RATIO (F6.1)')
      ACCEPT 64,SCLSET
64    FORMAT(F6.1)
C
C       ENTER THE TEST POINT MODE. SET DIGITAL LEVELS
C       TO APPROACH THE 5 VOLT VALUES
C
      TYPE 53
53    FORMAT(X,'HOW MANY TEST POINTS DO YOU WANT ? 0 FOR NONE')
      ACCEPT 54,BCOUNT
54    FORMAT(I6)
      IF(BCOUNT.EQ.0)GO TO 2
      DO 2300 L=1,BCOUNT
*
```

```
      SUMNR=0.0
      RATSUM=0.0
      SMRTSQ=0.0
      SUMW3=0.0
      SUMSQR=0.0
      TYPE 47
47    FORMAT(X,'  W3',2X,'  NR',2X,'  W2',2X,'  FR',3X,
     *'  B3',2X,'  BNR',2X,'  B2',2X,'  BF')
      DO 2200 I=1,30
17    CONTINUE
      CALL COL12(B1,B3,W1,W3)
      TYPE 46,W3,W1,W2,FRING,B3,B1,B2,BFR
46    FORMAT(8I6)
      STORW3=W3
      STORW1=W1
      IF(STORW3.LE.0.0)GO TO 17
      IF(STORW1.LE.0.0)GO TO 17
      SUMW3=SUMW3+STORW3
      SUMNR=SUMNR+STORW1
      RATIO=STORW3/STORW1
      RATIO=RATIO*SCLSET
      RATSUM=RATSUM+RATIO
2200  CONTINUE
      XSHOTS=30.
      AVEW3(L)=SUMW3/XSHOTS
      AVENR(L)=SUMNR/XSHOTS
      AVEPOW(L)=RATSUM/XSHOTS
      TYPE 228,AVEW3(L),AVEPOW(L),AVENR(L)
2300  CONTINUE
      TYPE 51
51    FORMAT(X,'DO YOU WANT MORE TEST POINTS ? (YES OR NO)')
      ACCEPT 52,ANS5
52    FORMAT(A1)
      IF(ANS5.EQ.YES)GO TO 1
2     CONTINUE
C
C       START THE DATA COLLECTION AND INITIALIZE THE RUNNING
C       TOTAL SUM COUNTERS.
C
      TYPE 61
61    FORMAT(X,'ARE YOU READY TO START A REAL SCAN ? (YES OR NO)',/)
      ACCEPT 62,ANS6
62    FORMAT(A1)
      IF(ANS6.EQ.NO)GO TO 1
        TSUMW3=0.0
        TSUMNR=0.0
        TSUMRA=0.0
        TSQRW3=0.0
        TSQRNR=0.0
        TSQRRA=0.0
      TYPE 30
30    FORMAT(X,'START')
C
C       THE LOOP IN L COUNTS THE DATA GROUPS
C
      DO 100 L=1,RANGE
      SUMNR=0.0
      SMNRSQ=0.0
      RATSUM=0.0
      SMRTSQ=0.0
*
```

```
      SUMW3=0.0
      SUMSQR=0.0
C
C       THE LOOP IN I COUNTS THE DATA POINTS
C
      DO 200 I=1,SHOTS
18    CONTINUE
      CALL COL12(B1,B3,W1,W3)
      STORW3=W3
      STORW1=W1
      IF(STORW3.LE.0.0)GO TO 18
      IF(STORW1.LE.0.0)GO TO 18
      SUMW3=SUMW3+STORW3
      SQRW3=STORW3*STORW3
      SUMSQR=SUMSQR+SQRW3
      SUMNR=SUMNR+STORW1
      SQRNR=STORW1*STORW1
      SMNRSQ=SMNRSQ+SQRNR
      RATIO=STORW3/STORW1
      RATIO=RATIO*SCLSET
      RATSUM=RATSUM+RATIO
      SQRRAT=RATIO*RATIO
      SMRTSQ=SMRTSQ+SQRRAT
      IF(ANS01.EQ.NO)GO TO 200
      WRITE(12,201)W3,W1
201   FORMAT(X,I4,X,I4)
200   CONTINUE
1000  CONTINUE
C
C       DISPLAY THE AVERAGE AND DEVIATION OF THE GROUP
C
      XSHOTS=SHOTS
      AVEW3(L)=SUMW3/XSHOTS
      AVENR(L)=SUMNR/XSHOTS
      AVEPOW(L)=RATSUM/XSHOTS
      IF(AVEW3(L).EQ.0.0)GO TO 229
      SW3=SQRT((1/(XSHOTS-1))*ABS(SUMSQR-XSHOTS*AVEW3(L)*
     *AVEW3(L)))
      SNR=SQRT((1/(XSHOTS-1))*ABS(SMNRSQ-XSHOTS*AVENR(L)*
     *AVENR(L)))
      SRAT=SQRT((1/(XSHOTS-1))*ABS(SMRTSQ-XSHOTS*AVEPOW(L)*
     *AVEPOW(L)))
240   CONTINUE
      CW3(L)=SW3/AVEW3(L)*100.
      CNR(L)=SNR/AVENR(L)*100.
      DEV(L)=SRAT/AVEPOW(L)*100.
      GO TO 230
229   CW3(L)=0.0
       CNR(L)=0.0
      DEV(L)=0.0
      AVEPOW(L)=0.0
230   CONTINUE
228   FORMAT(X,'AVE W3 = ',F7.1,2X,'AVE OF RATIO = ',F7.1
     *,2X,'AVE OF NR = ',F7.1)
C
C       COMPUTE AND DISPLAY THE RUNNING TOTAL
C
        TSUMW3=TSUMW3+SUMW3
        TSUMNR=TSUMNR+SUMNR
        TSUMRA=TSUMRA+RATSUM
*
```

```
      TSQRW3=TSQRW3+SUMSQR
      TSQRNR=TSQRNR+SMNRSQ
      TSQRRA=TSQRRA+SMRTSQ
      TSHOTS=SHOTS*L
      TAVEW3=TSUMW3/TSHOTS
      TAVENR=TSUMNR/TSHOTS
      TAVERA=TSUMRA/TSHOTS
      TSW3=SQRT((1/(TSHOTS-1))*ABS(TSQRW3-TSHOTS*TAVEW3*
     *TAVEW3))
      TSNR=SQRT((1/(TSHOTS-1))*ABS(TSQRNR-TSHOTS*TAVENR*
     *TAVENR))
      TSRA=SQRT((1/(TSHOTS-1))*ABS(TSQRRA-TSHOTS*TAVERA*
     *TAVERA))
      TSW3=TSW3/TAVEW3*100.
      TSNR=TSNR/TAVENR*100.
      TSRA=TSRA/TAVERA*100.
      TYPE 500
500   FORMAT(X,'GROUP',3X,'W3',6X,'S(W3)',4X,'NR',6X,'S(NR)',
     *4X,'RA',4X,'S(RA)',5X,'RUNNING TOTAL')
      TYPE 501,L,AVEW3(L),CW3(L),AVENR(L),CNR(L),AVEPOW(L),DEV(L),
     *TAVEW3,TSW3
501   FORMAT(I5,2X,F6.1,'+/-',F5.1,'%',X,F6.1,'+/-',
     *F5.1,'%',X,F6.1,'+/-',F5.1,'%',2X,'W3='F6.1,
     *'+/-',F5.1,'%')
      TYPE 502,TAVENR,TSNR
502   FORMAT(56X,'NR=',F6.1,'+/-',F5.1,'%')
      TYPE 503,TAVERA,TSRA
503   FORMAT(56X,'RA=',F6.1,'+/-',F5.1,'%')
100   CONTINUE
      TYPE 101
101   FORMAT(/,/,X,'FINISH')
      WRITE(12,9)TAVEW3,TSW3,TAVENR,TSNR,TAVERA,TSRA
9     FORMAT(8G12.5)
      CLOSE(UNIT=12,DISPOSE='SAVE')
      STOP
      END
*
```

AVRAGE.SAV

```
C       THIS PROGRAM IS DESIGNED TO AVERAGE W3 AND TO COLLECT THE
C       ASSOCIATED FRINGE PATTERN.THE PROGRAM WILL COLLECT ANY
C       NUMBER OF POINTS---WITHIN THE ALLOWED SPACE OF THE DISK.
C       THE STORAGE SIZE IS 32 POINTS PER BLOCK
C
      INTEGER*2 JMAX,SHOTS,RALIM,RANGE,INTW3,INTRAT
      DATA YES,NO,ALL/'Y','N','A'/
      REAL*4 NO
      REAL*8 FILE,NAME
      DOUBLE PRECISION SQRW3,SUMSQR,ABS
      INTEGER*4 NUMEND,ISTART,IEND,KSPACE
      IMPLICIT INTEGER*2(B,F,W)
      IMPLICIT REAL*4(M,A,S,R,X,Y,T,C,D)
      DIMENSION W3(99),FRING(99),B3(99),BFR(99),
     *STORW3(99),STORFR(99),IFIX(99),CW3(1000),
     *AVEW3(1000),AVEFR(1000),FILE(4),NAME(4)
C
      TYPE *,' '
      TYPE *,'THIS PROGRAM COLLECTS W3,SW3 AND FRINGES'
      TYPE *,'DATA STORAGE REQUIRES 2 BLOCKS/32 POINTS'
      TYPE *,' '
      TYPE *,'10 OR 12 BIT COVERSION ON W3 ?'
      ACCEPT *,IAD
      TYPE *,'DO YOU WANT TO SEE THE OPERATION CHECKLIST ?'
      ACCEPT 40,ANSCHK
      IF(ANSCHK.EQ.NO)GO TO 3
      CALL CHECK(IAD)
C
3     PAUSE'PLACE OUTPUT DATA DISK IN DX1'
      TYPE *,'HOW MANY FREE BLOCKS ON DATA DISK'
      ACCEPT *,IBLOCK
C
8     TYPE *,'HOW MANY DATA POINTS DO YOU WANT ?'
      ACCEPT *,RANGE
      IF(RANGE.GT.7600)RANGE=7600
      IDISK=RANGE/16
      IF(IDISK.LE.IBLOCK)GO TO 4
      TYPE *,''
      TYPE *,'NOT ENOUGH SPACE ON DISK--USE ANOTHER'
      TYPE *,''
      GO TO 3
C
C       OPEN A FILE TO SAVE DATA AND CREATE A FILE ID
C
4     TYPE 5
5     FORMAT(X,'YOU NEED TO ASSIGN A FILE NAME--DX1:NAME.DAT',/)
      CALL ASSIGN(11,'DX1:XXXXXX.DAT',-6)
      TYPE *,'ENTER A FILE I.D. NAME  8 CHARACTERS'
      ACCEPT 248,FILE,NAME
C
      WRITE(11,249)FILE,NAME,RANGE
C
      TYPE *,'SELECT THE NUMBER OF SHOTS TO BE AVERAGED'
      ACCEPT *,SHOTS
C
      TYPE *,'HOW MANY STEPS/DATA POINT ?'
      ACCEPT *,IRES
C
C       THE REAL TIME PLOTTING PARAMETERS ARE NOW SET
C
      TYPE *,'SET THE PLOTTING TIME CONSTANT IN SEC (F4.2)'
      ACCEPT *,TCW
      TYPE *,'HOW MANY CM/WAVENUMBER (1,2,3,OR,4) IN THE PLOT'
      ACCEPT *,JMAX
*
```

```
      JMAX=JMAX*3
      SLICE=JMAX
      AI=EXP(-1./(TOW*SLICE))
      SI=1.-AI
      YINTFR=0.0
      YINTW3=0.0
C
C       ASK ABOUT THE SIGMA CHECK.THIS ALLOWS THE AVERAGE
C       ESTIMATOR TO BEHAVE AS A ROBUST ESTIMATOR.
C
      XLIM=0.0
      TYPE 35
35    FORMAT(X,'DO YOU WANT A DATA CHECK AGAINST SIGMA?  (YES OR NO)')
      ACCEPT 40,ANS1
40    FORMAT(A1)
      IF(ANS1.EQ.NO)GO TO 26
      TYPE 45
45    FORMAT(X,'WITHIN HOW MANY SIGMA? (F4.2)')
      ACCEPT 50,XLIM
50    FORMAT(F4.2)
26    CONTINUE
C
C       COLLECT TEST POINTS TO SET THE DIGITAL SIGNAL LEVELS
C
      TYPE 63
63    FORMAT(X,'YOU NEED TO SET THE SIGNAL LEVEL INPUTS',/,/)
      PAUSE  'SCAN TO MAXIMUM IN W3 '
C
1     CONTINUE
      TYPE 53
53    FORMAT(X,'HOW MANY TEST POINTS DO YOU WANT ? 0 FOR NONE')
      ACCEPT 54,BCOUNT
54    FORMAT(I6)
      IF(BCOUNT.EQ.0)GO TO 2
C
C       COLLECT THE TEST POINTS,SUBTRACT THE BACKGROUND
C       AND KEEP A RUNNING SUM FOR THE AVERAGE
C
      DO 2300 L=1,BCOUNT
      SUMW3=0.0
      SUMSQR=0.0
      SUMFR=0.0
      IFLAG=0
      TYPE 47
      DO 2200 I=1,SHOTS
      CALL COLTWO(IAD,BFR(I),B3(I),W3(I),FRING(I))
      TYPE 46,W3(I),B3(I),FRING(I),BFR(I)
      STORW3(I)=W3(I)-B3(I)
      STORFR(I)=FRING(I)-BFR(I)
      SUMW3=SUMW3+STORW3(I)
      SUMFR=SUMFR+STORFR(I)
2200  CONTINUE
C
C       COMPUTE AND DISPLAY TEST AVERAGES FOR W3 AND FRINGES
C
      XSHOTS=SHOTS
      AVEW3I=SUMW3/XSHOTS
      AVEFRI=SUMFR/XSHOTS
      TYPE 228,AVEW3I,AVEFRI
C
2300  CONTINUE
      TYPE 51
      ACCEPT 40,ANS5
      IF(ANS5.EQ.YES)GO TO 1
*
```

```
2     CONTINUE
      TYPE *,'ARE YOU READY TO START A REAL SCAN ? (YES OR NO)'
      ACCEPT 40,ANS6
      IF(ANS6.EQ.NO)GO TO 1
C
C       ENTER W2 FREQUENCY FOR LATER CALIBRATION
C
      TYPE *,'MEASURE THE RED AT INITIAL FREQ (ENTER AS F7.2 ANG)'
      ACCEPT *,SWIN
      WRITE(11,151)SWIN,SHOTS,XLIM
C
C       THIS PORTION COLLECTS THE DATA, COMPUTES THE AVERAGE AND
C       THE STANDARD DEVIATION, PLOTS THE DATA IN REAL
C       TIME AND STEPS THE DYE LASER.
C
      PAUSE'START THE SCAN'
      TYPE *,'START'
33    CONTINUE
      IF(RANGE.GT.1000)GO TO 31
      RALIM=RANGE
      GO TO 32
31    CONTINUE
      RALIM=1000
32    CONTINUE
C
C       THE LOOP IN L COLLECTS N DATA POINTS
C
      DO 100 L=1,RALIM
      SUMW3=0.0
      SUMSQR=0.0
      SUMFR=0.0
      IFLAG=0
C
C       THE LOOP IN I COLLECTS THE SHOTS TO BE AVERAGED
C
      DO 200 I=1,SHOTS
34    CONTINUE
C
C       COLLECT W3 AND FRINGE, SUBTRACT THE BACKGROUND
C       ADD TO THE SUM FOR COMPUTATION OF AVERAGE AND DEVIATION
C
      CALL COLTWO(IAD,BFR(I),B3(I),W3(I),FRING(I))
      STORW3(I)=W3(I)-B3(I)
      STORFR(I)=FRING(I)-BFR(I)
      IF(STORW3(I).LE.0.0)GO TO 34
      IF(STORFR(I).LE.0.0)GO TO 34
      SUMW3=SUMW3+STORW3(I)
      SQRW3=STORW3(I)*STORW3(I)
      SUMSQR=SUMSQR+SQRW3
      SUMFR=SUMFR+STORFR(I)
200   CONTINUE
C
C       CALCULATE AVERAGE AND SIGMA
C
1000  CONTINUE
      XSHOTS=SHOTS
      AVEW3(L)=SUMW3/XSHOTS
      IF(AVEW3(L).EQ.0.0)GO TO 229
      SW3=SQRT((1/(XSHOTS-1))*ABS(SUMSQR-XSHOTS*AVEW3(L)*
     *AVEW3(L)))
C
C       IF USING A ROBUST CHECK WITH SIGMA
C       FIND THE SIGMA LIMITS.
C
*
```

```
      IF(ANS1.EQ.NO)GO TO 240
      IFLAG=0
      XLIMIT=XLIM*SW3
C
C       CHECK EACH SHOT.KEEP THOSE SHOTS WITHIN THE SIGMA LIMIT
C
      DO 1002 I=1,SHOTS
      DIFFW3=ABS(AVEW3(L)-STORW3(I))
      IF(DIFFW3.LE.XLIMIT)GO  TO 1001
      IFLAG=IFLAG+1
      IFIX(IFLAG)=I
1001  CONTINUE
C
C       THE INDEX OF THE BAD POINTS ARE SAVED IN ARRAY IFIX
C       THESE POINTS ARE THEN REPLACED
C
1002  CONTINUE
      IF(IFLAG.EQ.0)GO TO 240
      DO 1003 J=1,IFLAG
      I=IFIX(J)
      CALL COLTWO(IAD,BFR(I),B3(I),W3(I),FRING(I))
1003  CONTINUE
C
C       A NEW AVERAGE IS COMPUTED AND THE NEW SET OF POINTS
C       IS CHECKED AGAINST THE SIGMA LIMITS UNTIL THERE
C       ARE THE DESIRED NUMBER OF GOOD SHOTS
C
      SUMW3=0.0
      SUMSQR=0.0
      SUMFR=0.0
      DO 1100 I=1,SHOTS
      STORW3(I)=W3(I)-B3(I)
      STORFR(I)=FRING(I)-BFR(I)
      SUMW3=SUMW3+STORW3(I)
      SQRW3=STORW3(I)*STORW3(I)
      SUMSQR=SUMSQR+SQRW3
      SUMFR=SUMFR+STORFR(I)
1100  CONTINUE
      GO TO 1000
C
C       AFTER THE AVERAGE IS SATISFIED BY SIGMA,
C       PARAMETERS ARE STORED IN AN ARRAY FOR LATER USE
C
240   CONTINUE
      AVEFR(L)=SUMFR/XSHOTS
      CW3(L)=SW3
      GO TO 230
229   CW3(L)=0.0
      AVEFR(L)=SUMFR/XSHOTS
230   CONTINUE
C
C       THE AVERAGE VALUES ARE NOW PLOTTED ON A STRIP CHART RECORDER
C
      DO 2000 J=1,JMAX
      YINTW3=AI*YINTW3+SI*AVEW3(L)
      YINTFR=AI*YINTFR+SI*AVEFR(L)
      INTW3=YINTW3+.5
      INTFR=YINTFR+.5
      CALL PLTTWO(INTW3,INTFR)
2000  CONTINUE
C
C       THE DYE LASER IS STEPPED
C
      DO 100 I=1,IRES
*
```

```
      CALL STEP
100   CONTINUE
C
C       STORE THE AVERAGE VALUE OF W3, ITS STANDARD DEVIATION
C       AND THE AVERAGE VALUE FOR THE FRINGES
C       THEY ARE STORED ON DX1 AS X,I4,F5.1,I4
C
      DO 155 I=1,RALIM
      IAVEW3=AVEW3(I)+.5000
      IAVEFR=AVEFR(I)+.50
      WRITE(11,243)IAVEW3,CW3(I),IAVEFR
155   CONTINUE
C
C       GO BACK FOR MORE POINTS IF NEED BE
C
      RANGE=RANGE-1000
      IF(RANGE.GT.0)GO TO 33
C
C       FINISHED WITH SCAN--CLOSE THE FILE
C
      TYPE *,'FINISH'
      ENDFILE 11
      CALL CLOSE(11)
C
C       GO TO BEGINNING FOR ANOTHER SCAN
C
      TYPE *,'DO YOU WANT ANOTHER SCAN ?'
      ACCEPT 40,ANSSCN
      IBLOCK=IBLOCK-IDISK
      IF(ANSSCN.EQ.YES)GO TO 8
46    FORMAT(4I6)
47    FORMAT(X,'  W3',2X,'  B3',2X,
     *'  FR',2X,'  BFR')
51    FORMAT(/,X,'DO YOU WANT MORE TEST POINTS ? (YES OR NO)')
151   FORMAT(X,F7.2,I2,F4.2)
228   FORMAT(X,'AVE W3 = ',F7.1,6X,'AVE FRINGES = ',F7.1)
243   FORMAT(X,I4,F5.1,I4)
248   FORMAT(8A1)
249   FORMAT(8A1,I5)
      STOP
      END
C
C
C
C
C
        SUBROUTINE CHECK(IAD)
        TYPE *,'THE CABLE ASSIGNMENTS ARE:'
        TYPE *,'        W3 TO MCPHERSON TO INVERTING AMP'
        IF(IAD.EQ.10)GO TO 5
        TYPE *,'        INVERTING AMP TO 12 BIT INPUT'
        GO TO 10
5       TYPE *,'        INVERTING AMP TO CH00 OF 10 BIT'
10      TYPE *,'        FRINGES TO PHOTO DIODE TO AMP'
        TYPE *,'        AMP TO CH03 OF 10 BIT A/D'
        TYPE *,''
        TYPE *,'TRIGGERS:'
        TYPE *,'        THE BACKGROUND TRIGGERS ON THE RISING EDGE'
        TYPE *,'        YAG LAMP OUTPUT TO PULSE GENERATOR'
        TYPE *,'        PULSE GENRATOR OUTPUT TO ERA RET'
        TYPE *,'        SAMPLE A/D ON THE FALLING EDGE OF A PULSE'
        TYPE *,'        YAG Q SWITCH OUTPUT TO PULSE GENERATOR'
        IF(IAD.EQ.10)GO TO 15
        TYPE *,'        PULSE GENERATOR OUTPUT TO CNTR ST OF 12 BIT'
*
```

```
$
        GO TO 20
15      TYPE *,'         PULSE GENERATOR OUT TO EXT ST'
20      TYPE *,''
        TYPE *,'PLOTTING:'
        TYPE *,'         X OUT TO RECORDER (FRINGES)'
        TYPE *,'         Y OUT TO RECORDER (W3)'
        TYPE *,'         WRT THRU TO SYNC IN  SYNC OUT'
        TYPE *,'         TO EXT PULSES INPUT OF RECORDER'
        TYPE *,''
        TYPE *,'DYE LASER:'
        TYPE *,'         DISP CH2 OUTPUT TO REMOTE CLOCK'
        TYPE *,'         INPUT OF STEPPER MOTOR'
        TYPE *,''
        TYPE *,''
        RETURN
        END
*
```

```
C       RATIO IS INTENDED AS A SUBSTITUTE FOR AVRAGE
C       WHEN A RESONANT/NONRESONANT RATIO IS POSSIBLE.
C       THIS PROGRAM BEHAVES THE SAME AS AVRAGE
C       AND HAS THE SAME TRIGGERS, PLOTTING AND DYE
C       LASER STEPPING ASSIGNMENTS.
C
C
      INTEGER*4 JMAX, SHOTS, RA, RANGE, INTW3, INTRAT
      DATA YES, NO, ALL/'Y', 'N', 'A'/
      DOUBLE PRECISION SQRT, ABS, EXP
      REAL*8 NO
      INTEGER*4 NAVE, NAVPOW, NSTEPS
      IMPLICIT INTEGER*4(B, F, W)
      IMPLICIT REAL*8(M, A, S, R, X, Y, T, C, D)
      DIMENSION W3(50), W1(50), STORW3(50), STORW1(50), RATIO(50),
     *DIFF(50), AVEW3(200), AVEPOW(200), CW3(200), DEV(200),
     *AVENR(200), CNR(200),
     *B3(50), B1(50), BFR(50), B2(50), W2(50), FRING(50),
     *STORW2(50), STORFR(50), IFIX(50), AVEFR(200)
      TYPE 3
3     FORMAT(X, 'THE CABLE ASSIGNMENTS ARE:'/, X, 'WRITE THRU  RECORDER'
     *, /, X, 'DISP CH 2  STEPPER MOTOR', /, X, 'W3  CH00 OR 12 BIT',
     */, X, 'W1  CH01', /, X, 'W2  CH02', /, X, 'FRINGES  CH03', /, /, /)
      TYPE 5
5     FORMAT(X, 'SELECT THE NUMBER OF SHOTS TO BE AVERAGED')
      ACCEPT 10, SHOTS
10    FORMAT(I2)
      TYPE 901
901   FORMAT(X, 'SELECT THE NUMBER OF STEPS/POINT')
      ACCEPT 902, IRES
902   FORMAT(I2)
      TYPE 6
6     FORMAT(X, 'SET THE TIME CONSTANT IN SEC (F4.2)')
      ACCEPT 7, TOW
7     FORMAT(F4.2)
      TYPE 11
11    FORMAT(X, 'HOW MANY CM/WAVENUMBER (1, 2, 3, OR 4) IN THE PLOT')
      ACCEPT 12, JMAX
12    FORMAT(I2)
      TYPE 15
15    FORMAT(X, 'HOW MANY DATA POINTS DO YOU WANT ?')
      ACCEPT 20, RANGE
20    FORMAT(I4)
      TYPE 35
35    FORMAT(X, 'DO YOU WANT A DATA CHECK AGAINST SIGMA?  (YES OR NO)')
      ACCEPT 40, ANS1
40    FORMAT(A1)
      IF(ANS1.EQ.NO)GO TO 26
      TYPE 45
45    FORMAT(X, 'WITHIN HOW MANY SIGMA? (F4.2)')
      ACCEPT 50, XLIM
50    FORMAT(F4.2)
26    CONTINUE
1       TYPE 63
63      FORMAT(X, 'SET AN INTENSITY SCALE FACTOR FOR THE RATIO (F6.1)')
        ACCEPT 64, SCLSET
64      FORMAT(F6.1)
C
      TYPE 53
53    FORMAT(X, 'HOW MANY TEST POINTS DO YOU WANT ? 0 FOR NONE')
*
```

```
      ACCEPT 54,BCOUNT
54    FORMAT(I6)
      IF(BCOUNT.EQ.0)GO TO 2
      DO 2300 L=1,BCOUNT
      SUMNR=0.0
      RATSUM=0.0
      SMRTSQ=0.0
      SUMW3=0.0
      SUMSQR=0.0
      SUMFR=0.0
      IFLAG=0
      TYPE 47
47    FORMAT(X,'  W3',2X,'  NR',2X,'  W2',2X,'  FR',3X,
     *'  B3',2X,'  BNR',2X,'  B2',2X,'  BF')
      DO 2200 I=1,SHOTS
17    CONTINUE
      CALL COL10(BFR(I),B2(I),B1(I),B3(I),W3(I),W1(I),W2(I),
     *FRING(I))
      TYPE 46,W3(I),W1(I),W2(I),FRING(I),B3(I),B1(I),B2(I),BFR(I)
46    FORMAT(8I6)
      STORW3(I)=W3(I)-B3(I)
      STORW1(I)=W1(I)-B1(I)
      IF(STORW3(I).LE.0.0)GO TO 17
      IF(STORW1(I).LE.0.0)GO TO 17
      STORW2(I)=W2(I)-B2(I)
      STORFR(I)=FRING(I)-BFR(I)
      IF(STORFR(I).LE.0.0)GO TO 17
      SUMW3=SUMW3+STORW3(I)
      SQRW3=STORW3(I)*STORW3(I)
      SUMSQR=SUMSQR+SQRW3
      SUMNR=SUMNR+STORW1(I)
      RATIO(I)=STORW3(I)/STORW1(I)
      RATIO(I)=RATIO(I)*SCLSET
      RATSUM=RATSUM+RATIO(I)
      SQRRAT=RATIO(I)*RATIO(I)
      SMRTSQ=SMRTSQ+SQRRAT
      SUMFR=SUMFR+STORFR(I)
2200   CONTINUE
      XSHOTS=SHOTS
      AVEW3(L)=SUMW3/XSHOTS
      AVENR(L)=SUMNR/XSHOTS
      AVEPOW(L)=RATSUM/XSHOTS
      TYPE 228,AVEW3(L),AVEPOW(L),AVENR(L)
2300  CONTINUE
      TYPE 51
51    FORMAT(X,'DO YOU WANT MORE TEST POINTS ? (YES OR NO)')
      ACCEPT 52,ANS5
52    FORMAT(A1)
      IF(ANS5.EQ.YES)GO TO 1
2     CONTINUE
      TYPE 61
61    FORMAT(X,'ARE YOU READY TO START A REAL SCAN ? (YES OR NO)',/)
      ACCEPT 62,ANS6
62    FORMAT(A1)
      IF(ANS6.EQ.NO)GO TO 1
C
      TYPE *,'MEASURE AND ENTER OMEGA 2 IN ANG.  XXXX.XX'
      ACCEPT *,SWIN
      JMAX=JMAX*3
      SLICE=JMAX
*
```

```
      AI=EXP(-1/(TOW*SLICE))
      SI=1-AI
      YINTRA=0.0
      YINTW3=0.0
      TYPE 30
30    FORMAT(X,'START')
      CALL STEP
      DO 100 L=1,RANGE
      SUMNR=0.0
      SMNRSQ=0.0
      SQNR=0.0
      RATSUM=0.0
      SMRTSQ=0.0
      SUMW3=0.0
      SUMSQR=0.0
      SUMFR=0.0
      IFLAG=0
      DO 200 I=1,SHOTS
18    CONTINUE
       CALL COL10(BFR(I),B2(I),B1(I),B3(I),W3(I),W1(I),W2(I),
     *FRING(I))
      STORW3(I)=W3(I)-B3(I)
      STORW1(I)=W1(I)-B1(I)
      IF(STORW3(I).LE.0.0)GO TO 18
      IF(STORW1(I).LE.0.0)GO TO 18
      STORW2(I)=W2(I)-B2(I)
      STORFR(I)=FRING(I)-BFR(I)
      IF(STORFR(I).LE.0.0)GO TO 18
      SUMW3=SUMW3+STORW3(I)
      SQRW3=STORW3(I)*STORW3(I)
      SUMSQR=SUMSQR+SQRW3
      SUMNR=SUMNR+STORW1(I)
      SQRNR=STORW1(I)*STORW1(I)
      SMNRSQ=SMNRSQ+SQRNR
      RATIO(I)=STORW3(I)/STORW1(I)
      RATIO(I)=RATIO(I)*SCLSET
      RATSUM=RATSUM+RATIO(I)
      SQRRAT=RATIO(I)*RATIO(I)
      SMRTSQ=SMRTSQ+SQRRAT
      SUMFR=SUMFR+STORFR(I)
200   CONTINUE
1000  CONTINUE
      XSHOTS=SHOTS
      AVEW3(L)=SUMW3/XSHOTS
      AVENR(L)=SUMNR/XSHOTS
      AVEPOW(L)=RATSUM/XSHOTS
      IF(AVEW3(L).EQ.0.0)GO TO 229
      SW3=SQRT((1/(XSHOTS-1))*ABS(SUMSQR-XSHOTS*AVEW3(L)*
     *AVEW3(L)))
      SNR=SQRT((1/(XSHOTS-1))*ABS(SMNRSQ-XSHOTS*AVENR(L)*
     *AVENR(L)))
      SRAT=SQRT((1/(XSHOTS-1))*ABS(SMRTSQ-XSHOTS*AVEPOW(L)*
     *AVEPOW(L)))
      IF(ANSI.EQ.NO)GO TO 240
      IFLAG=0
      RATLIM=XLIM*SRAT
      XLIMIT=XLIM*SW3
      DO 1002 I=1,SHOTS
      DIFRAT=ABS(AVEPOW(L)-RATIO(I))
      DIFFW3=ABS(AVEW3(L)-STORW3(I))
*
```

```
      IF(DIFRAT.LE.RATLIM)GO  TO 1001
      IFLAG=IFLAG+1
      IFIX(IFLAG)=I
1001  CONTINUE
1002  CONTINUE
      IF(IFLAG.EQ.0)GO TO 240
      DO 1003 J=1,IFLAG
      I=IFIX(J)
19    CONTINUE
      CALL COL10(BFR(I),B2(I),B1(I),B3(I),W3(I),W1(I),W2(I),
     *FRING(I))
      STRCHK=W1(I)-B1(I)
      IF(STRCHK.LE.0.0)GO TO 19
1003  CONTINUE
      SUMNR=0.0
      SMNRSQ=0.0
      SQRNR=0.0
      RATSUM=0.0
      SMRTSQ=0.0
      SUMW3=0.0
      SUMSQR=0.0
      SUMFR=0.0
      DO 1100 I=1,SHOTS
      STORW3(I)=W3(I)-B3(I)
      STORW1(I)=W1(I)-B1(I)
      IF(STORW3(I).LE.0.0)GO TO 19
      IF(STORW1(I).LE.0.0)GO TO 19
      STORW2(I)=W2(I)-B2(I)
      STORFR(I)=FRING(I)-BFR(I)
      IF(STORFR(I).LE.0.0)GO TO 19
      SUMW3=SUMW3+STORW3(I)
      SQRW3=STORW3(I)*STORW3(I)
      SUMSQR=SUMSQR+SQRW3
      SUMNR=SUMNR+STORW1(I)
      SQRNR=STORW1(I)*STORW1(I)
      SMNRSQ=SMNRSQ+SQRNR
      RATIO(I)=STORW3(I)/STORW1(I)
      RATIO(I)=RATIO(I)*SCLSET
      RATSUM=RATSUM+RATIO(I)
      SQRRAT=RATIO(I)*RATIO(I)
      SMRTSQ=SMRTSQ+SQRRAT
      SUMFR=SUMFR+STORFR(I)
1100  CONTINUE
      GO TO 1000
240   CONTINUE
      AVEFR(L)=SUMFR/XSHOTS
      CW3(L)=SW3
      CNR(L)=SNR
      DEV(L)=SRAT
      GO TO 230
229   CW3(L)=0.0
       CNR(L)=0.0
      DEV(L)=0.0
      AVEPOW(L)=0.0
      AVEFR(L)=SUMFR/XSHOTS
      NAVPOW=0
230   CONTINUE
228   FORMAT(X,'AVE W3 = ',F7.1,2X,'AVE OF RATIO = ',F7.1
     *,2X,'AVE OF NR = ',F7.1)
      DO 2000 J=1,JMAX
*
```

```
      YINTW3=AI*YINTW3+SI*AVEW3(L)
      YINTRA=AI*YINTRA+SI*AVEPOW(L)
      INTW3=YINTW3+.5
      INTRAT=YINTRA+.5
      IF(INTRAT.GT.1023)INTRAT=1023
      CALL CLOCK(INTW3,INTRAT)
2000  CONTINUE
      DO 900 I=1,IRES
      CALL STEP
900   CONTINUE
100   CONTINUE
      TYPE 101
101   FORMAT(X,'FINISH')
      TYPE 241
241   FORMAT(X,'YOU NEED TO ASSIGN A FILE NAME--DX1:NAME.DAT',/)
      CALL ASSIGN(11,'DX1:XXXXXX.DAT',-6)
      TYPE 247
247   FORMAT(X,'ENTER A FILE I.D. NAME  6 CHARACTERS')
      ACCEPT 248,FILE,NAME
248   FORMAT(2A4)
      WRITE(11,249)FILE,NAME,RANGE
249   FORMAT(2A4,I5)
      WRITE(11,*)SWIN,SHOTS,XLIM
      DO 250 I=1,RANGE
      WRITE(11,*)AVEW3(I),CW3(I),AVENR(I),CNR(I),AVEPOW(I),DEV(I)
     *,AVEFR(I)
250   CONTINUE
      ENDFILE 11
245   FORMAT(F7.1,'+OR-',F7.3,'%',X,F7.1,'+OR-',F7.3,'%',
     *F7.1,'+OR-',F7.3,'%')
      TYPE 103
103   FORMAT(X,'DO YOU WANT TO SEE THE DATA FILE (YES OR NO)')
      ACCEPT 104,ANS2
104   FORMAT(A1)
      IF(ANS2.EQ.NO)GO TO 105
      TYPE 246
246   FORMAT(2X,'AVEW3',6X,'S(W3)',2X,'AVENR',6X,'S(NR)',2X
     *,'AVE OF RATIO',5X,'S(RAT)')
      DO 300 K=1,RANGE
      IF(AVEW3(K).NE.0.0)GO TO 106
      CW3(K)=0.0
      CNR(K)=0.0
      DEV(K)=0.0
      GO TO 107
106   CONTINUE
      CW3(K)=CW3(K)/AVEW3(K)*100.
      CNR(K)=CNR(K)/AVENR(K)*100.
      DEV(K)=DEV(K)/AVEPOW(K)*100.
107   CONTINUE
      TYPE 245,AVEW3(K),CW3(K),AVENR(K),CNR(K),AVEPOW(K),DEV(K)
300   CONTINUE
105   CONTINUE
      CALL CLOSE(11)
      STOP
      END
*
```

CRUNCH.SAV

```
C       OMEGA IS THE CONTROL OR MAIN PROGRAM USED IN THE
C       PROCESSING OF THE DATA STORED AS FORMAT(I4,F5.1,I4) BY
C       'AVRAGE' OR 'RATIO'.OMEGA CALLS SEVERAL SUBROUTINES
C       IN WHICH THE SPECTRUM IS FITTED TO A POLYNOMIAL
C       FUNCTION FOR SMOOTHING,AND EACH DATA
C       POINT CALIBRATED TO THE INITIAL FREQUENCY ACCORDING TO
C       W1+W2 OR W1-W2.IN ADDITION TO CONTROLLING THE
C       SUBROUTINE CALLS,OMEGA PLOTS THE RAW DATA,THE SMOOTHED
C       DATA,AND SEARCHES THE FITTED DATA FOR MAXIMA.
C
C
        DIMENSION IY(2200),FREQ(7),IFIT(7)
        COMMON/PASS/IFR(2200),SY(2200),IFROUT
        COMMON/POLYIN/Y(7),X(7),SIGMAY(7)
        COMMON/POLOUT/YFIT(7),A(6)
        DATA YES,NO/'Y','N'/
        REAL*8 NFREQ,YES,NO,ANSP1,ANSP2,ANS1,ANS2,ANSFRG
        INTEGER*2 FLAG,SHOTS,RANGE
C
        TYPE *,'NOTE: RETURN NEEDED AFTER PAUSE STATEMENT'
        TYPE *,' '
        TYPE *,'  PLOTTING INSTUCTIONS'
        TYPE *,'        X OUT --- FRINGES'
        TYPE *,'        Y OUT --- SPECTRUM'
        TYPE *,'        WRITHRU TO LINEAR RECORDER FOR DRIVING PULSES'
        TYPE *,' '
C
C       ALLOW DX0 AND DX1 TO ACCESS USER FILES
C
        PAUSE' REPLACE DX0 WITH SHORT MONITOR'
        PAUSE' REPLACE DX1 WITH DATA DISK TO BE READ'
C
C       OPEN THE DATA FILE ON DX1 AND READ ITS PARAMETERS
C
        TYPE *,'ENTER THE FILE TO BE CRUNCHED -- DX1:NAME.DAT '
        TYPE *,' '
        CALL ASSIGN(13,'DX1:XXXXXX.DAT',-6)
        READ(13,1)FILE,NAME,RANGE
        READ(13,2)SWIN,SHOTS,XLIM
1       FORMAT(2A4,I5)
2       FORMAT(F7.2,I2,F4.2)
C
C       READ THE DATA FROM DISK INTO ARRAYS,FIND THE MAXIMUM
C       AND THE MINIMUM FOR THE THREE WAVE SIGNAL.
C
        MAX=0
        MIN=4095
        INTMAX=0
        INTMIN=4095
        I=1
3       CONTINUE
        READ(13,4,END=5)IY(I),SY(I),IFR(I)
        IY(I)=IABS(IY(I))
        IFR(I)=IABS(IFR(I))
        IF(IY(I).GT.MAX)MAX=IY(I)
        IF(IY(I).LT.MIN)MIN=IY(I)
        IF(IFR(I).GT.INTMAX)INTMAX=IFR(I)
        IF(IFR(I).LT.INTMIN)INTMIN=IFR(I)
4       FORMAT(X,I4,F5.1,I4)
*
```

```
          I=I+1
          GO TO 3
5         IY(I)=IY(I-1)
          IFR(I)=IFR(I-1)
C
C         PRINT THE # OF DATA POINTS IN THE CURRENT FILE
C
          NCOUNT=I-1
          TYPE 6,NCOUNT
6         FORMAT(X,'THERE ARE ',I6,' DATA POINTS IN THIS FILE',/)
C
C         OPEN THE SCRATCH FILE ON DX0
C
          OPEN(UNIT=12,NAME='DX0:FITTED.DAT')
C
C         EVALUATE THE PLOTTING MODE CHOICES
C
          TYPE *,'DO YOU WANT A PLOT OF THE RAW DATA ?'
          ACCEPT 7,ANSP1
7         FORMAT(A1)
          TYPE *,'DO YOU WANT A PLOT OF THE FITTED DATA ?'
          ACCEPT 7,ANSP2
          IF(ANSP1.EQ.NO.AND.ANSP2.EQ.NO)GO TO 20
C
C         SET THE PLOTTING SCALES BY %
C
          TYPE *,'SET THE PLOTTING SCALES BY % FULL SCALE'
          CALL PLOT(0,0)
          PAUSE 'SET 0'
          CALL PLOT(1000,1000)
          PAUSE 'SET 100%'
          CALL PLOT(0,0)
          PAUSE'ADVANCE THE CHART PAPER'
C
C         NUMBER OF PULSES TO THE STRIP CHART RECORDER DETERMINES
C         THE X AXIS EXPANSION IN THE PLOTTING.
C
          TYPE *,'SET X AXIS EXPANSION FACTOR -- RECORDER @ 30 CM/MIN'
          TYPE *,'  10 IS NORMAL EXPANSION'
          ACCEPT *,ISLICE
          IF(ANSP1.EQ.NO)GO TO 20
C
C         NORMALIZE THE DATA AND CONVERT TO INTEGER FOR PLOTTING
C
          DO 9 K=1,NCOUNT+1
          IY(K)=FLOAT(IY(K)-MIN)/FLOAT(MAX-MIN)*1000.+.5
9         CONTINUE
C
C         PLOT THE NORMALIZED RAW DATA
C
          DO 10 K=1,NCOUNT
          DO 10 I=1,ISLICE
          IPLT=IY(K)+(IY(K+1)-IY(K))*I/ISLICE
          IFRPLT=IFR(K)+(IFR(K+1)-IFR(K))*I/ISLICE
          CALL PLOT(IFRPLT,IPLT)
10        CONTINUE
C
C         GET READY FOR THE NEXT PLOT
C
          DO 11 I=1,1000
*
```

```
      CALL PLOT(0,0)
11    CONTINUE
C
C     THE WEIGHTING MODE IS PASSED TO SUBROUTINE POLFIT.
C     IT REFLECTS THE RELATIVE WEIGHTS FOR EACH DATA POINT.
C
      TYPE *,' '
20    TYPE *,'SET WEIGHTING MODE FOR POLY SMOOTH'
      TYPE *,'         1 -- INSTRUMENTAL  1/SIGMAY**2'
      TYPE *,'         0 -- EQUAL WEIGHT'
      TYPE *,'        -1 -- STAT 1/Y'
      ACCEPT *,MODE
C
C     THE PARAMETERS OMEGA1 AND ICAL ARE PASSED TO
C     THE SUBROUTINE FRING WHICH ASSIGNS THE CHANNEL
C     FREQUENCIES.
C
      TYPE *,'ENTER OMEGA 1 IN WAVENUMBERS XXXXX.XX'
      ACCEPT *,OMEGA1
      TYPE *,' '
      TYPE *,'ENTER 1 FOR W1+W2   -1 FOR W1-W2'
      ACCEPT *,ICAL
      TYPE *,' '
      TYPE *,'DO YOU WANT FRINGE DATA OUTPUT ?'
      ACCEPT 7,ANSFRG
      IFROUT=0
      IF(ANSFRG.EQ.YES)IFROUT=1
C
C     SET PARAMETERS FOR POLFIT 7 POINT SMOOTH
C
      J=0
      NTERMS=4
      NPTS=7
      IN=1
      IFIN=7
C
C     FILL X ARRAY FOR 7 POINT CUBIC POLYINOMIAL FIT BY POLFIT
C
      DO 25 L=1,7
      X(L)=L
25    CONTINUE
C
C     READ RAW DATA BACK INTO ARRAY
C
      IF(ANSP1.EQ.NO)GO TO 27
      REWIND 13
      READ(13,1)FILE,NAME,RANGE
      READ(13,2)SWIN,SHOTS,XLIM
      DO 26 I=1,NCOUNT
      READ(13,4,END=26)IY(I),SY(I),IFR(I)
26    CONTINUE
C
C     FILL Y ARRAY WITH 7 POINTS FOR POLYINOMIAL FIT
C
27    M=1
      DO 28 L=IN,IFIN
      Y(M)=IY(L)
      SIGMAY(M)=SY(L)
      M=M+1
28    CONTINUE
*
```

```
C
      CALL POLFIT(NPTS,NTERMS,MODE,CHISQR)
C
C     COMPUTE FITTED Y VALUES WITH FUNCTION  POLYFN
C
      DO 29 I=1,7
      YFIT(I)=POLYFN(X,I,A)
29    CONTINUE
C
C     STORE THE FITTED DATA ON DX0: UNIT 12.ALLOW SPACE
C     ON FILE 12 FOR THE FREQUENCY(F9.3).THIS DETERMINES
C     THE ACTUAL SIZE OF FILE 12 AND AN ENDFILE MARK IS
C     WRITTEN.IN ADDITION TO THE STORAGE,THE FITTED DATA
C     IS PLOTTED ON THE STRIP CHART RECORDER
C
      NFREQ=00000.000
      IF(J.NE.0)GO TO 33
C
C     STORE THE DATA
C
      DO 33 K=1,3
      WRITE(12,32)NFREQ,YFIT(K)
32    FORMAT(F9.3,X,F8.3)
33    CONTINUE
      J=J+1
      WRITE(12,32)NFREQ,YFIT(4)
      IN=IN+1
      IFIN=IFIN+1
      IF(IFIN.LE.NCOUNT)GO TO 27
      DO 34 K=5,7
      J=J+1
      WRITE(12,32)NFREQ,YFIT(K)
34    CONTINUE
C
C
      ENDFILE 12
      REWIND 12
      YYMAX=0.0
      YYMIN=4095.00
C
C     READ FITTED DATA INTO ARRAY SY(I) FOR PRING
C     FIND THE MAX AND MIN FOR FITTED DATA
C
      DO 35 I=1,NCOUNT
      READ(12,32)NFREQ,SY(I)
      IF(SY(I).GT.YYMAX)YYMAX=SY(I)
      IF(SY(I).LT.YYMIN)YYMIN=SY(I)
35    CONTINUE
C
C     NORMALIZE THE FITTED DATA TO 1000 FOR PLOTTING
C
      SY(NCOUNT+1)=SY(NCOUNT)
      IF(ANSP2.EQ.NO)GO TO 38
      DO 36 I=1,NCOUNT+1
      SY(I)=(SY(I)-YYMIN)/(YYMAX-YYMIN)*1000.
36    CONTINUE
C
C     PLOT THE FITTED DATA
C
      DO 37 K=1,NCOUNT
*
```

```
      DO 37 I=1,ISLICE
      IPLT=SY(K)+(SY(K+1)-SY(K))*FLOAT(I/ISLICE)+.5
      IFRPLT=IFR(K)+(IFR(K+1)-IFR(K))*I/ISLICE
      CALL PLOT(IFRPLT,IPLT)
37    CONTINUE
38    REWIND 12
C
C     SUBROUTINE FRING WILL ASSIGN A FREQUENCY TO EACH
C     DATA POINT ON FILE 12, AND WRITE THIS FREQUENCY
C     ONTO THAT FILE.
C
      CALL FRING(INTMAX,INTMIN,NCOUNT,OMEGA1,ICAL,SWIN)
C
C
C
C     THIS PORTION OF THE CONTROL PROGRAM SEARCHES THE
C     FITTED SPECTRUM FOR PEAK MAXIMA.IT SEARCHES WITHIN
C     ANY NUMBER OF DATA POINTS AND MUST SATISFY THE
C     MINIMUM INTENSITY REQUIREMENT.
C
      TYPE *,' '
      TYPE *,'DO YOU WANT A SEARCH FOR MAXIMA IN FITTED DATA'
      ACCEPT 40,ANS1
40    FORMAT(A1)
      IF(ANS1.EQ.NO)GO TO 70
      REWIND 12
      DO 42 M=1,NCOUNT
      READ(12,41,END=42)SY(M),IY(M)
41    FORMAT(F9.3,X,I4)
42    CONTINUE
43    TYPE *,'SET MIN. INTENSITY BY COUNTS FOR PEAK SEARCH'
      ACCEPT *,INTSRH
      TYPE *,'HOW MANY POINTS IN PEAK SEARCH (7,9,11 ETC) ?'
      ACCEPT *,NUMPEK
      IPEAK=(NUMPEK+1)/2
      KCNT=IPEAK
C
C     SEARCH THE NUMBER OF POINTS IN THE PEAK SEARCH FOR
C     A MAXIMUM. MAXIMUM MUST BE GREATER THEN MIN INTENSITY
C
44    CONTINUE
      IF(NCOUNT-IPEAK.LT.NUMPEK-1)GO TO 60
      IF(IY(IPEAK).LT.INTSRH)GO TO 50
      KIN=IPEAK-KCNT+1
      K=KIN
45    CONTINUE
      IF(IY(K).GT.IY(IPEAK))GO TO 50
      K=K+1
      IF(K.LE.NUMPEK)GO TO 45
      TYPE 46,IY(IPEAK),IPEAK,SY(IPEAK)
46    FORMAT(X,'A MAX OF',I4,' OCCURS AT # ',I4,'= ',F9.3,' WVENMS')
50    CONTINUE
      IPEAK=IPEAK+1
      GO TO 44
60    CONTINUE
C
C     CAN GO BACK FOR ANOTHER SEARCH WITH DIFFERENT
C     CHOICE OF PARAMETERS.
C
      TYPE *,'DO YOU WANT ANOTHER SEARCH ?'
*
```

p.119 missing from original. Author unavailable to supply.


```
C       SUBROUTINE POLFIT IS A CONTROL SUBROUTINE USED
C       TO FIT THE OMEGA DATA AND CALLS ALL OTHER
C       SUBROUTINES NEEDED IN THE FITTING PROCESS.
C       POLFIT CALCULATES THE COEFFICIENTS A(I)'S
C       FOR A LEAST SQUARES FIT TO AN ARRAY OF DATA
C       POINTS WITH THE CUBIC POLYINOMIAL
C
C       Y=A(1)+A(2)*X+A(3)*X**2+A(4)*X**3
C
C       THIS ROUTINE IS DISCUSSED IN BEVINGTON P.140
C
        SUBROUTINE POLFIT(NPTS,NTERMS,MODE,CHISQR)
        COMMON/POLYIN/Y(7),X(7),SIGMAY(7)
        COMMON/POLOUT/YFIT(7),A(6)
        DIMENSION SUMX(10),SUMY(10),ARRAY(5,5)
        DOUBLE PRECISION ARRAY
C
C       ACCUMULATE WEIGHTED SUMS
C
11      NMAX=2*NTERMS-1
        DO 13 N=1,NMAX
13      SUMX(N)=0.
        DO 15 J=1,NTERMS
15      SUMY(J)=0.
        CHISQ=0.
21      DO 50 I=1,NPTS
        XI=X(I)
        YI=Y(I)
31      IF(MODE) 32,37,39
32      IF(YI) 35,37,33
33      WEIGHT=1./YI
        GO TO 41
35      WEIGHT=1./(-YI)
        GO TO 41
37      WEIGHT=1.
        GO TO 41
39      WEIGHT=1./SIGMAY(I)**2
41      XTERM=WEIGHT
        DO 44 N=1,NMAX
        SUMX(N)=SUMX(N)+XTERM
44      XTERM=XTERM*XI
45      YTERM=WEIGHT*YI
        DO 48 N=1,NTERMS
        SUMY(N)=SUMY(N)+YTERM
48      YTERM=YTERM*XI
49      CHISQ=CHISQ+WEIGHT*YI**2
50      CONTINUE
C
C       CALCULATE COEFFICIENTS
C
51      DO 54 J=1,NTERMS
        DO 54 K=1,NTERMS
        N=J+K-1
54      ARRAY(J,K)=SUMX(N)
        DELTA=DETERM(ARRAY,NTERMS)
        IF(DELTA) 61,57,61
57      CHISQR=0.
        DO 59 J=1,NTERMS
59      A(J)=0.
        GO TO 80
*
```

```
61      DO 70 L=1,NTERMS
62      DO 66 J=1,NTERMS
        DO 65 K=1,NTERMS
        N=J+K-1
65      ARRAY(J,K)=SUMX(N)
66      ARRAY(J,L)=SUMY(J)
70      A(L)=DETERM(ARRAY,NTERMS)/DELTA
C
C       CALCULATE CHI SQUARE
C
71      DO 75 J=1,NTERMS
        CHISQ=CHISQ-2.*A(J)*SUMY(J)
        DO 75 K=1,NTERMS
        N=J+K-1
75      CHISQ=CHISQ+A(J)*A(K)*SUMX(N)
76      FREE=NPTS-NTERMS
77      CHISQR=CHISQ/FREE
80      RETURN
        END
C
C
C
C       FUNCTION DETERM EVALUATES THE DETERM
C       OF A SQUARE MATRIX -- SEE BEVINGTON P.293
C
C
        FUNCTION DETERM(ARRAY,NORDER)
        DOUBLE PRECISION ARRAY,SAVE
        DIMENSION ARRAY(5,5)
        NORDER=4
10      DETERM=1.
11      DO 50 K=1,NORDER
C
C       INTERCHANGE COLUMNS IF DIAGNOL ELEMENT IS ZERO
C
        IF(ARRAY(K,K)) 41,21,41
21      DO 23 J=K,NORDER
        IF(ARRAY(K,J)) 31,23,31
23      CONTINUE
        DETERM=0.
        GO TO 60
31      DO 34 I=K,NORDER
        SAVE=ARRAY(I,J)
        ARRAY(I,J)=ARRAY(I,K)
34      ARRAY(I,K)=SAVE
        DETERM=-DETERM
C
C       SUBTRACT ROW K FROM LOWER ROWS TO GET DIAGNOL MATRIX
C
41      DETERM=DETERM*ARRAY(K,K)
        IF(K-NORDER) 43,50,50
43      K1=K+1
        DO 46 I=K1,NORDER
        DO 46 J=K1,NORDER
46      ARRAY(I,J)=ARRAY(I,J)-ARRAY(I,K)*ARRAY(K,J)/ARRAY(K,K)
50      CONTINUE
60      RETURN
        END
C
C
*
```

```
$
C       FUNCTION POLYFN EVALUATES THE Y VALUE FOR A GIVEN
C       SET OF A(I)'S BY THE FOLLOWING:
C
C       Y=A(1)+A(2)*X+A(3)*X**2+A(4)*X**3
C
C
        FUNCTION POLYFN(X,I,A)
        DIMENSION A(6),X(7)
        XI=X(I)
        YFUN=0.
        YFUN=A(1)+A(2)*XI+A(3)*XI**2+A(4)*XI**3
        POLYFN=YFUN
        RETURN
        END
*
```

```
C       SUBROUTINE FRING IA A CONTROL SUBRUOTINE WHICH
C       CALIBRATES THE FREQUENCY OF A SPECTRUM RELATIVE
C       TO THE MEASURED INITIAL FREQUENCY OF THE
C       COLLECTING PROGRAM.IT IS THE RESPONSIBILITY OF
C       THIS PROGRAM TO MONITOR THE VARIATION OF CHISQR
C       AND THEN TO TERMINATE THE LINEARIZATION OF THE
C       FITTING FUNCTION.THE DATA IS FIT TO A GAUSSIAN
C       PEAK PLUS A QUADRATIC POLYNOMIAL BACKGROUND.
C
C
C
        SUBROUTINE FRING(INTMAX,INTMIN,NCOUNT,OMEGA1,ICAL,SWIN)
        DIMENSION MAX(3),YMAX(3),YSET(75),
     *  ASAV(6),CHISAV(2),ATWO(120),ATHREE(120)
        COMMON/PASS/INTY(2200),SY(2200),IFROUT
        COMMON /INPUT/Y(75),X(75)
        COMMON/OUTPUT/YFIT(75),A(6),SIGMAA(6)
        INTEGER*2 SHOTS,RANGE,FLAG
C
C       CONTROL PARAMETERS ARE SET
C
        J=1
        FLAG=0
        ISTOP=NCOUNT
        M=1
        SCLMAX=INTMAX
        SCLMIN=INTMIN
        KI=1
        NIT=3
C
C       SEARCH FOR FIRST COMPLETE FRING.THIS IS DONE BY
C       LOOKING FOR THREE MAXIMA IN A LOCAL FITTING
C       NEIGHBORHOOD.IF THE FRING HAS BOTH
C       A WELL DEFINED MAXIMUM AND A MINIMUM ON BOTH
C       SIDES IT IS FITTED.
C
5       CONTINUE
        M=KI
        IF(M.GE.ISTOP)GO TO 65
10      CONTINUE
        IF(FLAG.EQ.NIT)GO TO 18
13      CONTINUE
        IF(INTY(M+1).LE.INTY(M))GO TO 14
        M=M+1
        GO TO 13
14      CONTINUE
        IF(INTY(M+2).LE.INTY(M))GO TO 15
        M=M+2
        GO TO 13
15      CONTINUE
        IF(INTY(M+3).LE.INTY(M))GO TO 16
        M=M+1
        GO TO 13
16      CONTINUE
        INCHEK=INTMAX/4+.5
        IF(INTY(M).GE.INCHEK)GO TO 17
        M=M+1
        GO TO 13
17      CONTINUE
        FLAG=FLAG+1
*
```

```
      MAX(FLAG)=M
      YMAX(FLAG)=INTY(M)
      M=M+15
      GO TO 10
18    CONTINUE
C
C     SET THE INITIAL AND FINAL CHAN. NUMBERS FOR THE
C     FIRST FRING TO BE FITTED. SET THE FIRST TWO
C     COEFFICIENTS: A(1) IS THE INTENSITY OVER THE BACKGROUND
C     A(2) IS THE CHAN NUMBER OF THE FRING MAXIMUM.
C
      IDIFF=(MAX(2)-MAX(1))/2
      IF(NIT.EQ.2)MAX(1)=MAX(2)
      KI=MAX(1)-IDIFF
      IF(KI.LE.0)GO TO 21
      KF=MAX(1)+IDIFF
      IF(KF.GT.NCOUNT)KF=NCOUNT
      INTMAX=0
      INTMIN=1023
      DO 20 I=KI,KF
      IF(I.GT.NCOUNT)GO TO 20
      IF(INTY(I).GT.INTMAX)INTMAX=INTY(I)
      IF(INTY(I).LT.INTMIN)INTMIN=INTY(I)
20    CONTINUE
      A(1)=YMAX(1)-FLOAT(INTMIN)
      A(2)=1+IDIFF
      GO TO 23
C
C     IF FIRST MAXIMUM DOES NOT CORRESPOND TO A COMPLETE
C     FRING, START AT THE SECOND MAXIMUM FOUND IN THE
C     SEARCH. A(1) AND A(2) ARE SET.
C
21    CONTINUE
      IDIFF=(MAX(3)-MAX(2))/2+.5
      KI=MAX(2)-IDIFF
      KF=MAX(2)+IDIFF
      A(2)=1+IDIFF
      INTMAX=0
      INTMIN=1023
      DO 22 I=KI,KF
      IF(I.GT.NCOUNT)GO TO 22
      IF(INTY(I).GT.INTMAX)INTMAX=INTY(I)
      IF(INTY(I).LT.INTMIN)INTMIN=INTY(I)
22    CONTINUE
      A(1)=YMAX(2)-FLOAT(INTMIN)
23    CONTINUE
C
C     DETERMINE THE EXACT NUMBER OF POINTS IN THE FRING
C     AND FILL AN X AND Y ARRAY FOR SUBROUTINE CURFIT.
C
      NPTS=KF-KI+1
      L=0
      DO 30 K=KI,KF
      YSET(L+1)=INTY(K)
      L=L+1
30    CONTINUE
      I=1
31    CONTINUE
      IF(I.GT.NPTS)GO TO 32
      Y(I)=YSET(I)
*
```

```
      X(I)=I
      I=I+1
      GO TO 31
32    CONTINUE
C
C     SET INITIAL PARAMETERS FOR CURFIT
C
      FLAMDA=.001
      NTERMS=6
      MODE=0
      DIFF=IDIFF
      YMIN=INTMIN
      A(3)=.30*DIFF
      A(4)=YMIN
      A(5)=0.0
      A(6)=0.0
C
C     SUBROUTINE CURFIT USES THE INITIAL COEFFICIENTS A(I)
C     COMPUTES A REDUCED CHISQR AND RETURNS BOTH CHISQR
C     AND NEW COEFFICIENTS.CHISQR IS MONITORED FOR
C     A MINIMUM OR SIGN CHANGE.
C
      ILOOP=0
40    IF(ILOOP.EQ.2)GO TO 50
      CALL CURFIT(NPTS,NTERMS,MODE,FLAMDA,CHISQR)
      CHISAV(ILOOP+1)=CHISQR
      DO 41 I=1,6
      ASAV(I)=A(I)
41    CONTINUE
      ILOOP=ILOOP+1
      GO TO 40
50    CONTINUE
      IF(CHISAV(2).GE.CHISAV(1))GO TO 52
      CHISAV(1)=CHISAV(2)
      DO 51 L=1,6
      ASAV(L)=A(L)
51    CONTINUE
      CALL CURFIT(NPTS,NTERMS,MODE,FLAMDA,CHISQR)
      CHISAV(2)=CHISQR
      GO TO 50
52    CONTINUE
C
C     SAVE THE COEFFICIENTS WHICH CORRESPOND TO THE
C     REDUCED CHISQR.COMPUTE AND DISPLAY FRINGE OUTPUT
C     PARAMETERS IF DESIRED.
C
      DO 53 L=1,6
      A(L)=ASAV(L)
53    CONTINUE
      ATWO(J)=A(2)+FLOAT(KI-1)
      ATHREE(J)=A(3)
      IF(IFROUT.NE.1)GO TO 60
      TYPE *,IDIFF,KI,KF,NPTS
      TYPE 56,J,ATWO(J),SIGMAA(2)
56    FORMAT(X,'ATWO(',I3,') = ',F7.2,3X,'SIGMA(ATWO) = 'F9.4,/)
      GAMMA=ATHREE(J)*2.354/FLOAT(NPTS)
      TYPE 59,J,GAMMA,SIGMAA(3)
59    FORMAT(X,'FWHM AT ATWO(',I3,') = ',F9.4
     *  , '+/-',X,F9.4,X,'WAVENUMBERS',/)
60    J=J+1
*
```

```
C
C       ADJUST INITIAL PARAMETERS AND LOOP BACK FOR ANOTHER FRING
C
        IF(NCOUNT-KF.LT.NPTS-2)GO TO 65
        NIT=2
        FLAG=0
        MAX(3)=0
        GO TO 5
65      CONTINUE
C
C       THIS PORTION OF FRING ASSIGNES A FREQUENCY FOR EACH
C       CHANNEL NUMBER ACCORDING TO EITHER W1+W2 OR
C       W1-W2.THIS INFORMATION ALONG WITH THE FITTED
C       INTENSITIES OF THE SPECTRUM ARE WRITTEN IN THEIR
C       FINAL FORM FOR USE IN A PLOTTING ROUTINE.
C
        J=J-1
        OMEGA2=10000./SWIN*10000.
        FREQ=OMEGA1+OMEGA2
        IF(ICAL.LT.0)FREQ=OMEGA1-OMEGA2.
        I=1
        M=1
        ISTR=1
66      CONTINUE
        IF(M+1.GT.J)GO TO 70
        FRDIF=ATWO(M+1)-ATWO(M)
        FACTOR=.9975/FRDIF
        IEND=ATWO(M+1)+.5
67      DO 69 I=ISTR,IEND
        IF(ICAL.EQ.1)FREQ=FREQ-FACTOR
        IF(ICAL.LT.0)FREQ=FREQ+FACTOR
        IIYI=SY(I)+.5
        WRITE(12,68)FREQ,IIYI
68      FORMAT(F9.3,X,I4)
69      CONTINUE
        IF(I.GE.NCOUNT)GO TO 71
        M=M+1
        ISTR=IEND+1
        GO TO 66
70      CONTINUE
        ISTR=IEND+1
        IEND=NCOUNT
        GO TO 67
71      CONTINUE
        RETURN
        END
*
```

```
      FUNCTION FUNCTN(X,I,A)
C     PURPOSE: TO EVALUATE A GAUSSIAN PEAK PLUS QUADRATIC PLOYINOMIAL
C     SEE BEVINGTON P.214
      DIMENSION X(75),A(6)
11    XI=X(I)
12    FUNCTN=A(4)+A(5)*XI+A(6)*XI**2
13    Z=(XI-A(2))/A(3)
      Z2=Z**2
      IF(Z2-50.)16,20,20
16    FUNCTN=FUNCTN+A(1)*EXP(-Z2/2.)
20    RETURN
      END
      SUBROUTINE FDERIV(X,I,A,DELTAA,NTERMS,DERIV)
      DIMENSION X(75),A(6),DELTAA(6),DERIV(10)
      XI=X(I)
      Z=(XI-A(2))/A(3)
      Z2=Z**2
      IF(Z2-50.)21,15,15
15    DO 16 J=1,3
16    DERIV(J)=0.
      GO TO 24
21    DERIV(1)=EXP(-Z**2/2.)
      DERIV(2)=A(1)*DERIV(1)*Z/A(3)
      DERIV(3)=DERIV(2)*Z
24    DERIV(4)=1.
      DERIV(5)=XI
      DERIV(6)=XI**2
      RETURN
      END
      FUNCTION FCHISQ (Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
C        PURPOSE:  EVALUATE REDUCED CHI SQUARE FOR FIT TO DATA
C          SEE BEVINGTON P194
      DOUBLE PRECISION CHISQ,WEIGHT
      DIMENSION Y(75),YFIT(75)
      SIGMAY=1.0
11    CHISQ=0.
12    IF (NFREE) 13, 13, 20
13    FCHISQ=0.
      GO TO 40
20    DO 30 I=1,NPTS
      IF(MODE) 22,27,29
22    IF(Y(I))25,27,23
23    WEIGHT=1./Y(I)
      GO TO 30
25    WEIGHT=1./(-Y(I))
      GO TO 30
27    WEIGHT=1.
      GO TO 30
29    WEIGHT=1./SIGMAY
30    CHISQ=CHISQ+WEIGHT*(Y(I)-YFIT(I))**2
31    FREE=NFREE
32    FCHISQ=CHISQ/FREE
40    RETURN
      END
*
```

```
      SUBROUTINE CURFIT(NPTS,NTERMS,MODE,FLAMDA,CHISQR)
      DOUBLE PRECISION ARRAY,BETA
      COMMON/INPUT/Y(75),X(75)
      COMMON/OUTPUT/YFIT(75),A(6),SIGMAA(6)
      DIMENSION B(10),WEIGHT(100),ALPHA(10,10),BETA(10),
     *  DERIV(10),ARRAY(10,10),DELTAA(6)
      SIGMAY=0.0
11    NFREE=NPTS-NTERMS
      IF(NFREE.GT.0) GO TO 20
13    CHISQR = 0.
      RETURN
20    DO 30 I=1,NPTS
      IF(MODE.NE.-2)GO TO 21
      WEIGHT(I)=1./(Y(I)**2)
      GO TO 30
21    IF(MODE) 22,27,29
22    IF(Y(I)) 25,27,23
23    WEIGHT(I)= 1./Y(I)
      GO TO 30
25    WEIGHT(I)=1./(-Y(I))
      GO TO 30
27    WEIGHT(I)=1.
      GO TO 30
29    CONTINUE
30    CONTINUE
31    DO 34 J=1,NTERMS
      BETA(J)=0.
      DO 34 K=1,J
34    ALPHA(J,K)=0
41    DO 50 I=1,NPTS
      CALL FDERIV (X,I,A,DELTAA,NTERMS,DERIV)
      DO 46 J=1,NTERMS
      BETA(J)=BETA(J)+WEIGHT(I)*(Y(I)-FUNCTN(X,I,A))*DERIV(J)
      DO 46 K=1,J
46    ALPHA(J,K)=ALPHA(J,K)+WEIGHT(I)*DERIV(J)*DERIV(K)
50    CONTINUE
51    DO 53J=1,NTERMS
      DO 53K=1,J
53    ALPHA(K,J)=ALPHA(J,K)
61    DO 62 I=1,NPTS
62    YFIT(I) = FUNCTN(X,I,A)
63    CHISQ1=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
71    DO 74 J=1,NTERMS
      DO 73K=1,NTERMS
      TERM=SQRT(ALPHA(J,J)*ALPHA(K,K))
      IF(TERM.EQ.0)TYPE 1000,J,K
1000  FORMAT('  TERM,J,K',2I3)
73    ARRAY(J,K) = ALPHA(J,K)/TERM
74    ARRAY(J,J) = 1.+FLAMDA
80    CALL MATINV (ARRAY,NTERMS,DET)
81    DO 84 J=1,NTERMS
      B(J) =A(J)
      DO 84 K=1,NTERMS
84    B(J)= B(J) + BETA(K)* ARRAY(J,K)/SQRT(ALPHA(J,J)*ALPHA(K,K))
91    DO 92 I=1,NPTS
92    YFIT(I) =FUNCTN(X,I,A)
93    CHISQR=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
      IF(FLAMDA.EQ.0.) GO TO 101
      IF(CHISQ1-CHISQR) 95,101,101
95    FLAMDA=10.*FLAMDA
*
```

```
$
      GO TO 71
101   DO 103 J=1,NTERMS
      A(J)=B(J)
103   SIGMAA(J)=SQRT(ARRAY(J,J)/ALPHA(J,J))
      FLAMDA=FLAMDA/10.
110   RETURN
      END
*
```

```
C       SUBROUTINE MATINV
C
C       INVERTS A SYMMETRIC MATRIX AND CALCULATES ITS DETERMINANT
C       BY THE GAUSS-JORDN METHOD.  SEE P302 OF BEVINGTON
C       FOR DETAILS.
C
C       ARRAY - INPUT MATRIX WHICH IS REPLACED BY ITS INVERSE
C       NORDER - DEGREE OF MATRIX
C       DET - DETERMINANT OF INPUT MATRIX
C
        SUBROUTINE MATINV (ARRAY,NORDER,DET)
        DOUBLE PRECISION ARRAY,AMAX,SAVE
        DIMENSION ARRAY(10,10),IK(6),JK(6)
10      DET = 1
11      DO 100 K=1,NORDER
        AMAX=0.
21      DO 30 I=K,NORDER
        DO 30 J=K,NORDER
23      IF (DABS(AMAX) - DABS(ARRAY(I,J))) 24,24,30
24      AMAX=ARRAY(I,J)
        IK(K)=I
        JK(K)=J
30      CONTINUE
31      IF (AMAX) 41,32,41
32      DET = 0.
        GOTO 140
41      I=IK(K)
        IF (I-K) 21,51,43
43      DO 50 J=1,NORDER
        SAVE=ARRAY(K,J)
        ARRAY(K,J)=ARRAY(I,J)
50      ARRAY(I,J)=-SAVE
51      J=JK(K)
        IF (J-K) 21,61,53
53      DO 60 I=1,NORDER
        SAVE=ARRAY(I,K)
        ARRAY(I,K)=ARRAY(I,J)
60      ARRAY(I,J)=-SAVE
61      DO 70 I=1,NORDER
        IF (I-K) 63,70,63
63      ARRAY(I,K)=-ARRAY(I,K)/AMAX
70      CONTINUE
71      DO 80 I=1,NORDER
        DO 80 J=1,NORDER
        IF (I-K) 74,80,74
74      IF (J-K) 75,80,75
75      ARRAY(I,J)=ARRAY(I,J)+ARRAY(I,K)*ARRAY(K,J)
80      CONTINUE
81      DO 90 J=1,NORDER
        IF (J-K) 83,90,83
83      ARRAY(K,J)=ARRAY(K,J)/AMAX
90      CONTINUE
        ARRAY(K,K)=1./AMAX
100     DET=DET*AMAX
101     DO 130 L=1,NORDER
        K=NORDER-L+1
        J=IK(K)
        IF (J-K) 111,111,105
105     DO 110 I=1,NORDER
*
```

```
$
      SAVE=ARRAY(I,K)
      ARRAY(I,K)=-ARRAY(I,J)
110   ARRAY(I,J)=SAVE
111   I=JK(K)
      IF (I-K) 130,130,113
113   DO 120 J=1,NORDER
      SAVE=ARRAY(K,J)
      ARRAY(K,J)=-ARRAY(I,J)
120   ARRAY(I,J)=SAVE
130   CONTINUE
140   RETURN
      END
*
```