ABSTRACT

CALIBRATION OF PHOTOLUMINESCENCE EXPERIMENT

by
Chihchuan Daniel Lee

Photoluminescence in a semiconductor at room temperature is quite difficult to distinguish from the many nonlasing optical lines from the laser, therefore, a blank test is very important. In so doing, the noises of the detector has to be taken into account. The calibration, however, mainly depends on the detector and is less dependent on other components. Instead of using black-body radiation light source and pyroelectric detector, we use globar lamp and PbS detector; because of the difficulty in the former case in doing alignment of optical path (invisible) and operating at a lower signal level from the detector. The employment of a computer allows real time data acquisition, data averaging and real time displaying, plus, we can take the advantage if the scanning should terminate as the result is unexpected; better yet, store data and spectra for future need. The comparison between calibrated and uncalibrated spectrum shown that there are in fact significant deviations.

CALIBRATION OF PHOTOLUMINESCENCE EXPERIMENT

by
Chihchuan Danel Lee

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Masters of Science in Applied Physics

Department of Physics

January 1994

APPROVAL PAGE

CALIBRATION OF PHOTOLUMINESCENCE EXPERIMENT

Chihchuan Daniel Lee

Dr. K. Ken Chin, Thesis Advisor — Date
Professor of Physics, NJIT

Dr. John C. Hensel, Committee Member — Date
Distinguished Reserach Professor in Applied Physics
of the Department of Physics, NJIT

Dr. Halina Opyrchal, Committee Member — Date
Visiting Professor in Applied Physics
of the Department of Physics, NJIT

BIOGRAPHICAL SKETCH

Author: Chihchuan Daniel Lee

Degree: Master of Science in Applied Physics

Date: January 1994

Undergraduate and Graduate Education:

- Master of Science in Applied Physics,
  New Jersey Institute of Technology,
  Newark, New Jersey, 1994

- Bachelor of Science in Physics,
  Chung Yuan Christian University,
  Chung Li, Taiwan, 1989

Major: Applied Physics

This thesis is dedicated to
Dr. Chaoliang Lai and Miss Hsiuya Mao

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

Table **Page**

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

THE PRINCIPLE OF PHOTOLUMINESCENCE EXPERIMENT

An electron occupying a higher energy state than it would under equilibrium conditions makes a transition to an empty lower energy state and all or most of the energy difference between the two states can be emitted as electromagnetic radiation. Electromagnetic radiation, varying over enormous ranges of wavelength from radio waves to gamma radiation, carry energy through space at a fixed speed of about 3×10^8 meters per second in *vacuo*. The narrow range perceived by the eye, together with a small extension into the ultraviolet (UV) and infrared (IR) region, is commonly called "**light**", and its main requirement for emission is that the system not be at thermal equilibrium with its ambient. This deviation from equilibrium requires some form of excitation. The light emission process is generally called "**luminescence**".

1.1 Nature of Light

The study of luminescence, those phenomena involving the interaction of light must be take into account. Throughout history, philosophers and scientists have tried to explain what light is; in so doing, they have tested their evolving knowledge of our physical world. Although many early ideas have been proven false, others have been repeatedly verified by experimental tests. It is necessary to use two complementary and apparently contradictory models, by regarding light sometimes as a collection of particles and sometimes as a succession of waves. The wave model explains well the phenomena of reflection, refraction and diffraction without dealing with the absorption of energy. While the particle model can explain the phenomena other than the wave model's, a combination to both models can provide a complete the explanation of light. According to the *electromagnetic theory of Maxwell* the displacements forming the waves are varying electric and magnetic fields, the electric and magnetic oscillations taking place in directions at right angles to one another and to the direction of travel. The wave properties are characterized by a frequency ν, and a wavelength , λ which are related by the equation,

$$\lambda \cdot \nu = c \qquad \text{Eq.(1-1)}$$

where c is the constant velocity of light in *vacuo*.

To describe the phenomena involving absorption and emission of light it is necessary to invoke the requirements of the *Quantum Theory*. The basic idea of which is that the energy can only be absorbed in definite units, or *quanta*. The energy E, carried by a *quanta* is proportion to the frequency, i.e.,

$$E = h \cdot \nu = h \cdot c / \lambda \qquad \text{Eq.(1-2)}$$

where h is *Planck's constant* (6.624×10^{-27} erg-sec).

1.2 Conceptual Ideas of Light Absorption and Emission of Semiconductor

The electrons are bound to atoms and because of the *Uncertainty Principle*, we have to treat then electron as a wave as well as a particle. In semiconductor, the valence band contains many electrons and the conduction band has many empty states into which the electrons may be excited. The incident photons with energies greater than the band gap energy are absorbed while photons with energies less than the band gap are transmitted.

The excited electron, after absorbing a photon, may have more energy than energy band gap and resides in the conduction band. Thus, an EHP(electron-hole-pair) is created. The electron and hole created by this absorption process are "excess" nonequilibrium carriers. Due to scattering events, the excited electron loses energy to the lattice until its velocity reaches the *thermal equilibrium velocity* of other conduction band electrons. Since the hole and the electron are out of balance with their environment, they must eventually recombine. The recombination energy will be given up in the form of light emission. This is so called "**photoluminescence**". As Figure 1.1 indicates, optical absorption of a photon with $h\nu > E_g$ (energy gap) : (a) an EHP is created during photon absorption; (b) the excited electron gives up energy to the lattice by scattering events; (c) the electron recombines with a hole in the valence band and gives off a photon. Further, a photon with energy less than the energy gap is unable to excite an electron from the valence band to the conduction band, thus in pure semiconductor, there is negligible absorption of photons with $h\nu < E_g$. This well explains why some materials are transparent in certain wavelength ranges.[18]

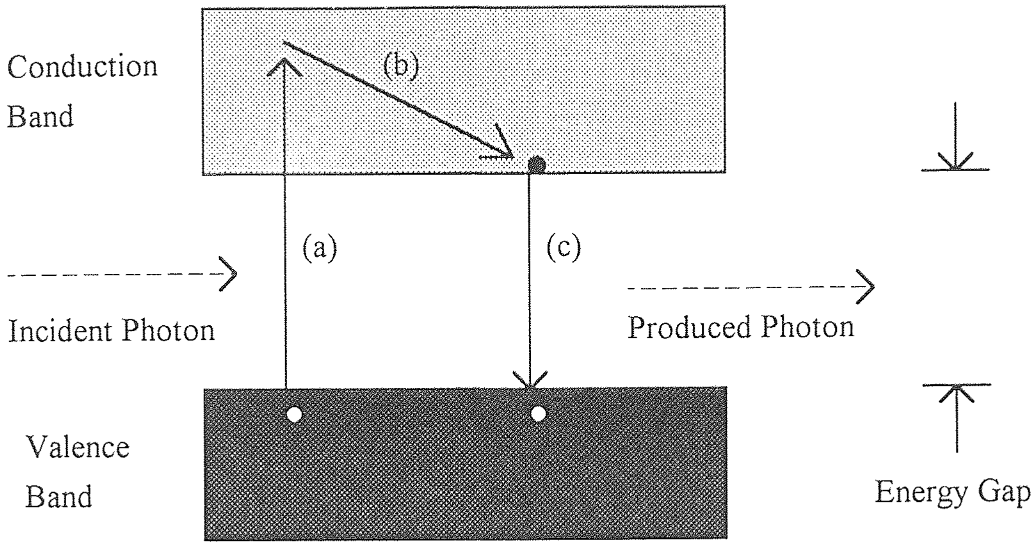


Figure 1.1

Optical Absorption

If the material is sufficiently pure and the temperature is sufficiently low, the electrons and the holes may pair off into excitons which then recombine, emitting a narrow spectral line. In a direct band gap semiconductor, where momentum is conserved in a simple radiative transition, the energy of the emitted photon is simply,

$$h\nu = E_g - E_x \qquad \text{Eq.(1-3)}$$

where E_x is the binding energy of the *exciton*.

In an indirect gap semiconductor, since the momentum of the photon p=E/c is negligible, and momentum conservation requires that a phonon be emitted or absorbed to complete the transition. The energy of the emitted or absorbed photon therefore is,

$$h\nu = E_g - E_x \pm E_p$$

Eq.(1-4)

where E_p is the energy of the emitted or absorbed *phonon*.

The narrow emission spectrum of the exciton can be replicated at several lower photon energies such that,

$$h\nu = E_g - E_x \pm mE_p$$

Eq.(1-5)

where m is the number of *optical phonons* emitted or absorbed per transition.

It should be noted that although the transition with phonon emission is less probable than the direct recombination, the resulting photon has a greater chance to escape because it occurs at a lower photon energy than the direct recombination, i.e., in a region of the spectrum where the semiconductor is more transparent. However, the reabsorption of exciton radiation is not necessarily a loss of energy because, in the process, a new exciton is formed which will provide another opportunity to emit.[5][7][11][17][18][23]

1.3 Modes of Excitation

1.3.1 By Light Absorption

The reciprocal relationships of absorption and emission of light *quanta*, and their connection with electronic energy levels, have been discussed above. For quantitative experimental work, or for clear thinking on the subject, one deals with the absorption of *monochromatic light*. The fundamental absorption refers to band-to-band or to exciton transition, i.e., to the excitation of an electron from the valence band to the conduction band. The fundamental absorption, which manifests itself by a rapid rise in absorption, can be used to determine the energy gap of the semiconductor. However, because the transitions are subject to certain selection rules, the estimation of the energy gap from the *absorption edge* is not a straightforward process -- even if competing absorption process can be accounted for.

If a beam of photons with $h\nu > E_g$ falls on a semiconductor, there will be some predictable amount of absorption, determined by the properties of material. Assume that a photon beam of intensity I (photons/cm^2-s) is directed at a sample of thickness l. The beam contains only photons of wavelength λ, selected by a *monochromator*. As the beam passes through the sample, its intensity at a distance x from the surface can be calculated by considering the probability of absorption within any increment **d**x. Since a photon which has survived to x without absorption has no memory of how far it has traveled, its probability in any **d**x is proportional to the intensity remaining at x,

$$- \mathbf{d}I(x) / \mathbf{d}x = \alpha I(x)$$

Eq.(1-6)

The solution to this equation is,

$$I(x) = I \cdot e^{-\alpha x}$$

Eq.(1-7)

and the intensity of light transmitted through the sample thickness l is,

$$I_t = I \cdot e^{-\alpha l}$$

Eq.(1-8)

The coefficient α is called the *absorption coefficient* and has units of cm^{-1}. This coefficient will of course vary with the photon wavelength and with the material. In a typical plot of a vs. wavelength, there is negligible absorption at long wavelengths ($h\nu$ small) and considerable absorption of photons with energies larger than E_g. According to Eq.(1-2), the relation between photon energy and wavelength is $E = hc / \lambda$. If E is given in electron volts and λ in micrometers, this becomes $E = 1.24 / \lambda$ (ev). When electron hole pairs are generated in a semiconductor, or when carriers are excited into higher impurity levels from which they fall to their equilibrium states, light can be given off by the material. Many of the semiconductors are well suited for light emission, particularly the compound semiconductors with direct band gap. For steady state excitation, the recombination of EHPs occurs at the same rate as the generation, and one photon is emitted for each photon absorbed. If carriers are excited by photon absorption, the radiation resulting from the recombination of the excited carrier is called "**photoluminescence**".[7][11][14][18]

1.3.2 By Electron Impact

This is the means by which the luminescence from *mercury*- and *sodium*-, vapour lamps, cathode-ray screens, etc., is excited. The electrons have energies very much larger than the quantized levels of the outer electrons of atoms and molecules, and by their passage they detach bound electrons which enter the accelerated stream. In the plasma (ion-electron gas) so formed, recombination occurs, and by downward jumps between energy levels light *quanta* are emitted. Molecules exposed to electron impact, unless very stable, become chemically broken up. And any luminescence observed is due to those products, usually radicals, which have good radiative powers. The term "*cathodeluminescence*" has been used to describe the effect of electron impact, but has also been applied to several other phenomena of quite different nature. One is the glow at electrodes observed in some electrolyses, which is due to ion-radical recombination. Another is the luminescence excited by application of voltages to thin layers of specially prepared Zinc sulphide, where the light emission is probably due to electronic effects at points of local high potential gradient. Electroluminescence also describes the light produced when a current is passed across a p-n junction of a semiconductor such as gallium arsenide or phosphide; the effect is here due to "*recombination*" of the charge carriers". To sum up, a high electric field ($E > 2 \times 10^5$ V/cm) can cause across-the gap ionization of electron of electron-hole pairs. The carriers thus generated are accelerated by the electric field and interact with valence electrons via a collision process which causes an avalanche of electron-hole-pair generation. Some of the pairs recombine radiatively, but because of their high kinetic energy, the emission spectrum is quite broad. However, after the electric field is turned off, the carriers thermalize to the band edges and recombine in a narrow spectrum.[7]

1.3.3 By Heat

Temperature radiation from hot, totally absorbing substances is calculable by *Planck's theory* and is expressed by the well-known "**black body**" curves of intensity against wavelength. Imperfectly absorbing materials are equally imperfectly emitting, and their temperature radiation -- called *incandescence* in the luminous region, never exceeds in intensity the contours of the corresponding "**black body**" curve. True luminescence is the emission of greater quantities of light than thermal radiation theory allows, and therefore always involves some momentary disturbance of the thermal equilibrium; in fact, it interferes with the concept of a true temperature. The ambiguous term "**thermoluminescence**" does not refer to normal temperature emission, but to a special effect found in some crystal phosphors. After these have been excited by light, radioactive exposure, etc., electrons liberated within the crystal become "**trapped**" (especially at low temperatures) in some way, as at lattice imperfections or foreign ion inclusions. Energy so stored may then sometimes be re-emitted as luminescence when heating drives electrons out of their trapped positions. Illumination by infrared radiation may produce a similar result; this is not due to heating but to the specific absorption of inferred by the traps. The phenomenon is complex because of the several imperfectly understood stages involved, some of which lead to energy degradation instead of emission.[7]

1.3.4 By Chemical Reaction

Radiative recombination of ground state atoms or small radical is rare and weak. The collision time for a bimolecular encounter, about 10^{-13} s, is very short compared with radiative times of 10^{-8} s or more. Even if collisions are stabilized by a third body, interaction in pairs of, say, **H**-atoms, give singlet ground state and triplet excited state molecules, between which radiation is *forbidden*. In cases where the potential energy curve *cross* -- come close together with interaction, *inverse predissociation* may lead to formation of an excited molecule, but the probability of its radiation becomes large only when a number of conditions are fulfilled. Higher energy states of atoms or small radicals, capable of forming excited molecules which radiationally *combine* with the ground state, are needed for strong emission effects. For complex oxidation reactions or organic molecules in solution, potential energy surface considerations become less useful. Chemiluminescence reactions in this field are uncommon either because the reaction energy is insufficient of because the molecules present are incapable of radiation, i.e., "**nonfluorescent**".[7]

1.4 Modes of Emission

The terms "**fluorescence**" and "**phosphorescence**" have no strictly agreed meaning, and have been used variously at different times. Emission from atoms excited by light absorption is usually called "**atomic fluorescence**"; if the atoms start from and return to the ground state, the emission is termed resonance radiation. The atomic electron waves can be said to be in resonance with the absorbed or emitted light waves, of an identical sharp frequency. If light of a different, lower frequency is shone on the gas, the electrons are merely subjected to *forced vibrations* which slow up the light without real energy transfer -- refractive index effect. With molecules effects of intermolecular atomic vibrations intrude into the picture. Under conditions of forced vibrations -- use longer wavelength non-absorbed light, the refractive index effect operates for the greater part of the light beam, but a few molecules behave differently. They abstract from the light a small quantity of energy to set their nuclei into vibration, leaving a few light *quanta* of diminished energy and therefore frequency, whose waves cannot combine with the incident beam, and so emerge in all directions. The spectrum of such light therefore shows a relatively strong line of the incident (monochromatic) light, scattered sideways by local density fluctuations, and having a weak line or lines on the longer wavelength side; the frequency differences give molecular vibration frequencies. This is the *Raman Effect*, and it is a common phenomenon for all molecules because the "lifetime" of the radiation-matter interaction is less than 10^{-15} s -- too short for any degradatory processes to occur. The simplest example of light emission from a semiconductor occurs for direct excitation and recombination of an EHP. If the recombination of EHPs occurs at the same rate as the generation, and one photon is emitted for each photon absorbed. Direct recombination is a fast process; the mean lifetime of the EHP is usually on the order of 10^{-8} s or less. Thus the emission of photons stops within approximately 10^{-8} s after the excitation is turned off. Such fast luminescent processes are often referred to as *fluorescence*. In some materials, however, emission continues for periods up to seconds or minutes after the excitation is removed. These slow processes are called *phosphorescence*, and the materials are called *phosphors*. An example of slow process is that a material contains a defect level -- perhaps due to an impurity, in the band gap which has a strong tendency to temporarily capture (*trap*) electrons from the conduction band . In such a material the emission of phosphorescent light persists for a relatively long time after the excitation is removed. In photoluminescence, there is a great deal of confusion concerning the definitions of terms used to discriminate between the different types of *long-lived* luminescence. Originally phosphorescence was used to describe any *long-lived* emission, and fluorescence for any *short-lived* emission. In order that this definition be useful, it was necessary to give a more precise interpretation of *long-lived* and this was usually done by considering any luminescence with a lifetime greater than 10^{-4} s as *long-lived*. However, this presupposes that all luminescence decay is exponential. Therefore, we can come to a conclusion that *fluorescence* is a luminescence which occurs only during excitation. *Phosphorescence* is a luminescence which continues for some time after the excitation is terminated.[7][17][18]

1.5 Photoluminescence

Optical excitation (via absorption) produces *photoluminescence*.

1.5.1 Direct Recombination of Electrons and Holes

Electrons in the conduction band of a semiconductor may make transitions to the valence band -- *recombination*, an excess population of electrons and holes decays by electrons falling from the conduction band to empty states (holes) in the valence band. Energy lost by an electron-hole-pair in making the transition is given up as a photon. Direct recombination occurs spontaneously; that is, the probability that an electron and a hole will recombine is constant in time. As in the case of carrier scattering, this constant probability leads us to expect an exponential solution for decay of the excess carriers. In this case the rate of decay of electrons at any time t is proportional to the number of electrons and the number of holes at t. The net rate of change in the conduction and electron concentration is the *thermal generation rate* $\alpha \cdot n_i^2$ (α is absorption coefficient) minus the *recombination rate*,

$$dn(t) / dt = \alpha n_i^2 - \alpha n(t)p(t)$$

Eq.(1-9)

Let us assume that the excess electron-hole population is created at t = 0, for example, by a short flash of light, and the initial excess electron and hole concentrations δn and δp are equal. Then as the electrons and holes recombine in pairs, the instantaneous concentrations of excess carriers $\delta n(t)$ and $\delta p(t)$ are also equal. Thus we can write the total concentrations of Eq. (1-9) in terms of the equilibrium values n_o and p_o and the excess carrier concentrations $\delta n(t) = \delta p(t)$. And because for the intrinsic material at equilibrium, we have $n_o \cdot p_o = n_i^2$, therefore,

$$\mathbf{d}\delta n(t) / \mathbf{d}t = \alpha n_i^2 - \alpha[\, n_o + \delta n(t)\,][\, p_o + \delta p(t)\,]$$
$$= - \alpha[\, (n_o + p_o)\delta n(t) + \delta n^2(t)\,]$$

Eq.(1-10)

This nonlinear equation would be difficult to solve in its present form. However, it can be simplified for the case of low-level injection. If the excess carrier concentrations are small, we can neglect the δn^2 term. Furthermore, if the material is extrinsic, we can usually neglect the term representing the equilibrium minority carriers. For example, if the material is p-type ($p_o >> n_o$), Eq.(1-10) becomes,

$$\mathbf{d}\delta n(t) / \mathbf{d}t = - \alpha p_o \delta n(t)$$

Eq.(1-11-n)

if the material is n type ($n_o >> p_o$) then,

$$\mathbf{d}\delta p(t) / \mathbf{d}t = -\alpha n_o \delta p(t)$$

Eq.(1-11-p)

The solution to Eq.(1-11-n) is an exponential decay from the original excess carrier concentration δn,

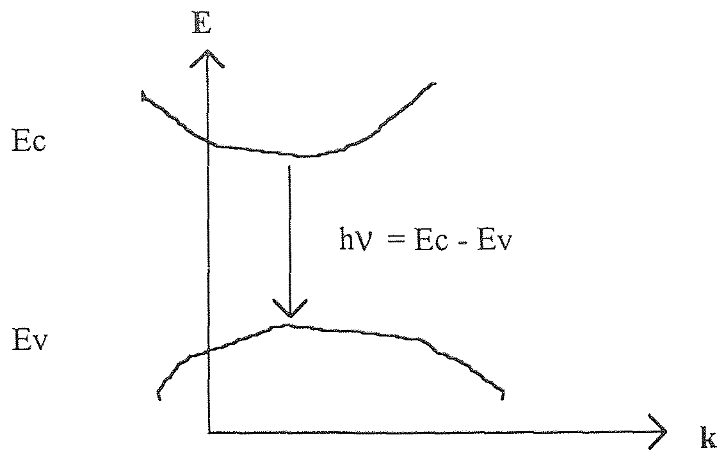
$$\delta n(t) = \delta n \cdot \mathbf{e}^{-\alpha p_o t}$$
$$= \delta n \cdot \mathbf{e}^{-t/\tau_n}$$

Eq.(1-12-n)

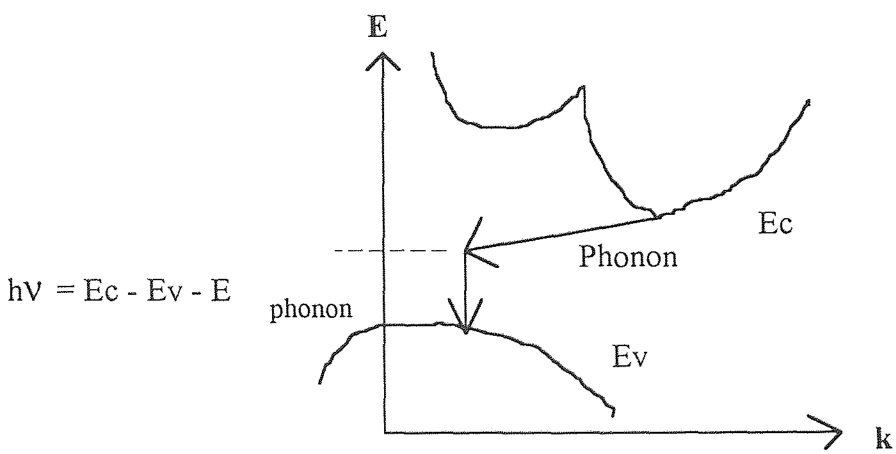
And, the solution to Eq.(1-11-p) is,

$$\delta p(t) = \delta p \cdot e^{-\alpha n_o t}$$
$$= \delta p \cdot e^{-t/\tau_n}$$

Eq.(1-12-p)



(a) Direct Gap Transition



(b) Indirect Gap Transition

Figure 1.2

Radiative Transition from Conduction Band to Valence Band

Excess electrons in a p-type semiconductor recombine with a decay constant $\tau_n = (\alpha p_o)^{-1}$, called the *recombination lifetime*. Since the calculation is made in terms of the minority carriers, τ_n is often called the *minority carrier lifetime*. The decay of excess holes in n-type material occurs with $\tau_p = (\alpha n_o)^{-1}$. In the case of direct recombination, the excess majority carriers decay at exactly the same rate as the minority carriers. [13][17][18][21]

Figure 1.2 is the diagram of direct and indirect transition in the semiconductor.

1.5.2 Indirect Recombination; Trapping

In column IV semiconductors and as well as some certain compound semiconductors, the probability of direct electron-hole recombination is very small. There are photons with energy equal to energy gap given off by materials such as Si and Ge during recombination, but this radiation is very weak and may be detected only by sensitive equipment. The vast majority of the recombination events in indirect materials occur via recombination levels within the band gap, and the resulting energy loss by recombining electrons is usually given up to the lattices as heat rather than by the emission of photons. Any impurity or lattice defect can serve as a recombination center if it is capable of receiving a carrier of one type and subsequently capturing the opposite type of carrier, thereby annihilating the pair. The carrier lifetime resulting from indirect recombination is somewhat more complicated than is the case for direct recombination, since it is necessary to account for unequal times required for capturing each type of carrier. In particular, recombination is often delayed by the tendency for a captured carrier to be thermally reexcited to its original band before capture of the opposite type of carrier can occur. When a carrier is trapped temporarily at a center and then is reexcited without recombination taking place, the process is often called "**temporary**

trapping". Although the nomenclature varies somewhat, it is common to refer to an impurity or defect center as a trapping center (or simply trap) if, after capture of one type of carrier, the most probable next event is re excitation. If the most probable next event is capture of the opposite type of carrier, the center is pre-dominantly a recombination center. The recombination can be slow or fast, depending on the average time the first carrier is held before the second carrier is captured. In general, trapping levels located deep in the band gap are slower in releasing trapped carriers than are the levels located near one of the bands. This results from the fact that more energy is required, for example, to reexcite a trapped electron from a center near the middle of the gap to the conduction band than is required to reexcited an electron from a level closer to the conduction band.[13][17][18][21]

1.5.3 Intraband Transitions

In addition to the conduction-to-valence band transition, electron transitions in a semiconductor may occur within the same energy band. As is shown in Figure 1.3.

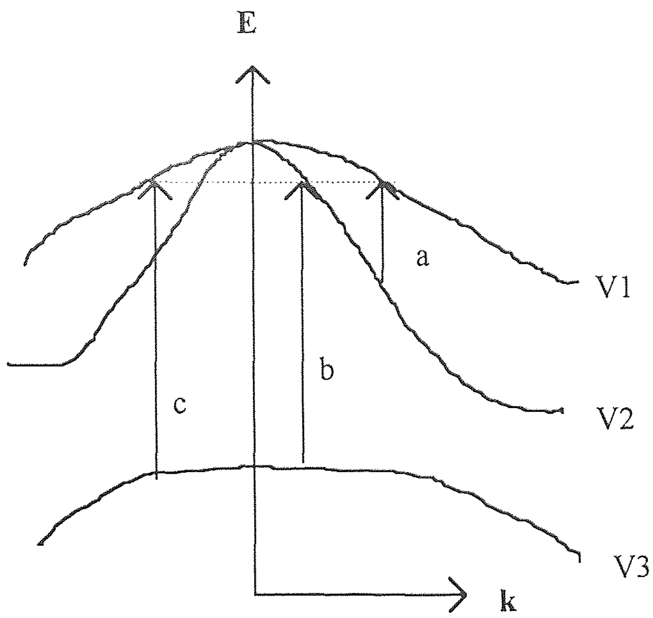


Figure 1.3

Valence Subband Structure and
Intraband Transitions[11]

A. p-type semiconductors --

The valence band of most semiconductors consists of three subbands which are separated by spin-orbit interaction. In p-type semiconductors, when the top of the valence band is populated with holes, it is possible to make three types of photon absorbing transitions: (**a**) from the light-hole band to the heavy-hole band; (**b**) from the split-off band to the heavy-hole band; an (**c**) from the split-off band to the light-hole band. These transitions have been observed in a number of semiconductors, and their interpretation can be verified by changing the position of the *Fermi level*, i.e., by doping. This absorption is proportional to the hole density, and it disappears when the material is made n-type. Because of selection rules, the transition probabilities between the valence subbands vanish at $k = 0$ but increase with k^2.[11][17]

B. n-type semiconductors --

In an n-type semiconductor intraband transitions between the set of conduction subbands are conceivable. The basic idea is about the same as p-type semiconductors. The absorption peak has been observed in many experiments, this peak increases with the electron concentration. Indirect transitions between minima at different k of the same conduction subband have been proposed to explain a bump in the low-energy absorption edge of other n-type material.[11][17]

1.6 Quantum Yields

Because of the nonradiational ways in which excited electrons can return to the ground state, fewer *quanta* are usually emitted as luminescence than are absorbed; the ratio of τ / τ_0(τ is the measured lifetime and τ_0 is the radiational lifetime) is called **quantum yield**. It is very difficult to measure the quantum yield directly because emission is in all directions (usually not equally), and for molecules is a wide band which has to be measured with detectors of unequal wavelength sensitivity. Once agreed values for certain standard substances are available, however, matters are simplified, and direct comparison of spectral intensities under equal conditions of illumination and light absorption gives the magnitudes sought. *Quantum yields* of emission depend on the relative rates of the processes; rates of emission depend on more complex factors which include the proximity of the levels. Theoretical treatment of degradation is very difficult. We can qualitatively explain facts about yields from a knowledge of the nature and relative positions of energy levels, which is obtained from absorption and emission spectral data, but quantitative prediction has yet to be developed. [6][10][13]

1.7 Quantum Efficiency

It was Einstein who emphasized the importance of determining the quantum efficiency when investigating the mechanism of photochemical reactions. The quantum efficiency may be defined in terms of the number of molecules of reactant that disappear, or the number of molecules of a particular product that are produced, per quantum of light absorbed, thus,

quantum efficiency of photoluminescence =

einsteins emitted / *einsteins* absorbed

Eq.(1-13)

where one *einstein* is equal to 6.023 x 10^{23} *quanta* (Nhv-erg).

For a given input excitation energy, the radiative recombination process is in direct competition with the nonradiative process. The *quantum efficient* is the fraction of the excited carriers that combine radiatively to the total recombination and may be written in terms of the lifetimes as,

$$\begin{aligned} \text{quantum efficiency} &= Rr / R \\ &= (1 / \tau_r) / (1 / \tau_r + 1 / \tau_{nr}) \\ &= \tau_{nr} / (\tau_{nr} + \tau_r) \end{aligned}$$

Eq.(1-14)

where Rr and R are the radiative recombination rate and total recombination rate, respectively. Whereas τ_{nr} is the *nonradiative lifetime* and τ_r is the *radiative lifetime*.

The *recombination rate* and *lifetime* of minority carriers are related to p-type layers by,

$$R = (n - n_o) / \tau$$

Eq.(1-15)

and to n-type layers by,

$$R = (p - p_o) / \tau$$

Eq.(1-16)

where n_o and p_o are electron and hole concentrations in *thermal equilibrium*, respectively, and n and p are electron and hole concentrations under optical excitations, respectively.

The minority carrier lifetime is given by,

$$\tau = \tau_r \cdot \tau_{nr} / (\tau_{nr} + \tau_r)$$

Eq.(1-17)

Eq. (1-14) shows that the *radiative lifetime* τ_r needs to be small to give high *quantum efficiency*.[6][10][13]

CHAPTER 2

SETUP OF PHOTOLUMINESCENCE EXPERIMENT

2.1 Apparatus and Principles

An interference band pass filter is used to guarantee that the exciting light source has a single wavelength, either 488 nm (blue) or 514.5 nm (green) for Ar ion Laser and 632.8 nm (red) for He-Ne Laser.

Our experimental setup is shown in Figure 2.1. The laser beam (filtered), First, passes through an aperture, second, is chopped, and, lastly, is focused onto the sample. The photoluminescence by this excitation is focused by lens L2 onto the entrance slit of the monochromator. The PbS detector is employed here to detect the light emerging from the exit slit of the monochromator. The output from the PbS detector is fed to a lock-in amplifier to process the signal and to communicate with the computer via GPIB (**G**eneral **P**urpose **I**nstrument **B**us) Card for the purpose of data acquisition. The computer also control the monochromator via JY488 interface equipped controller.

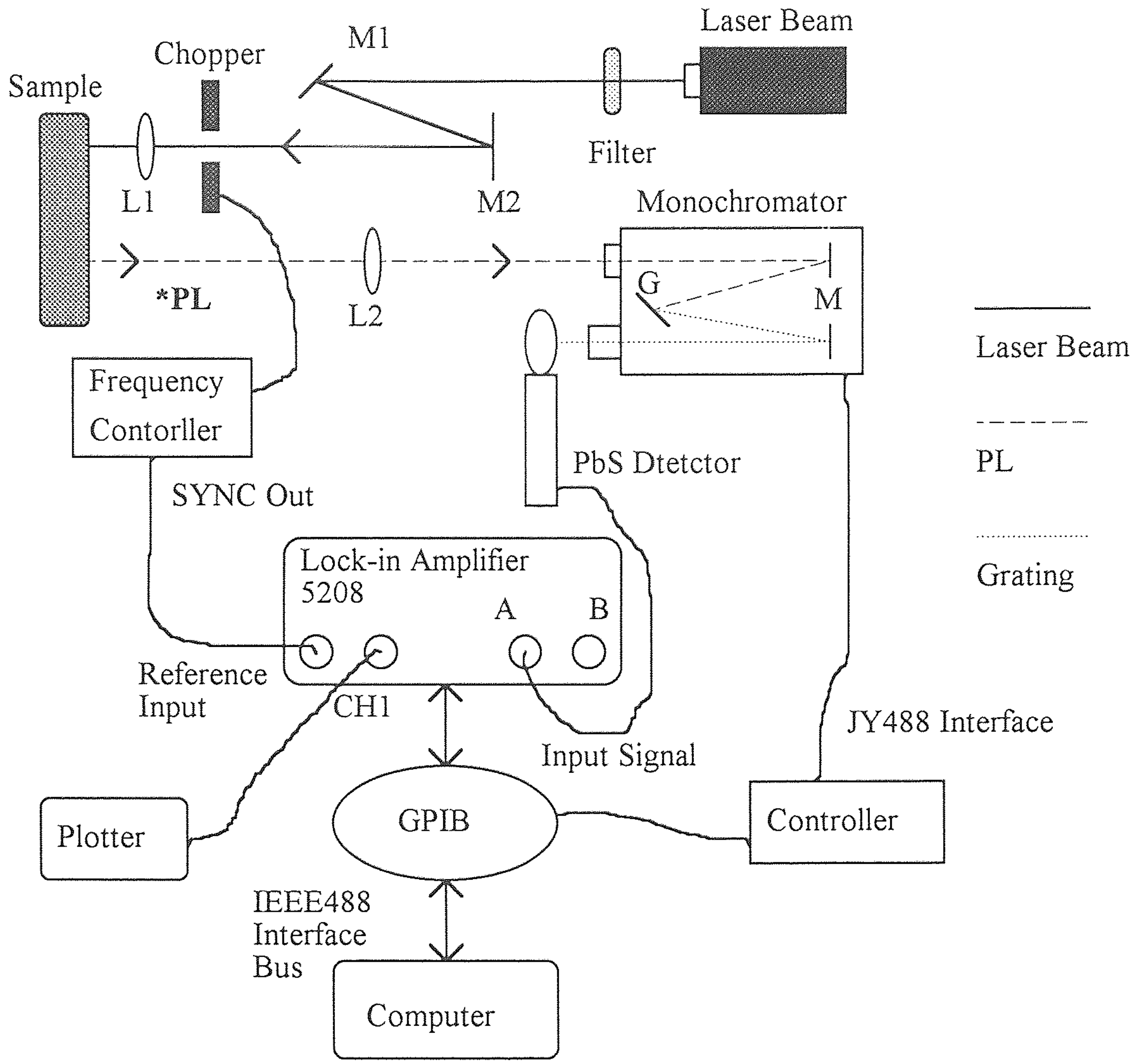


Figure 2.1

Experiment Setup

* PL might also include nonlasing optical lines from the laser beam.

2.2 Light Sources

The various gas lasers used as xciting light sources are listed in table 2.1.

Table 2.1 Some Gas Lasers *continuous wave, cw.

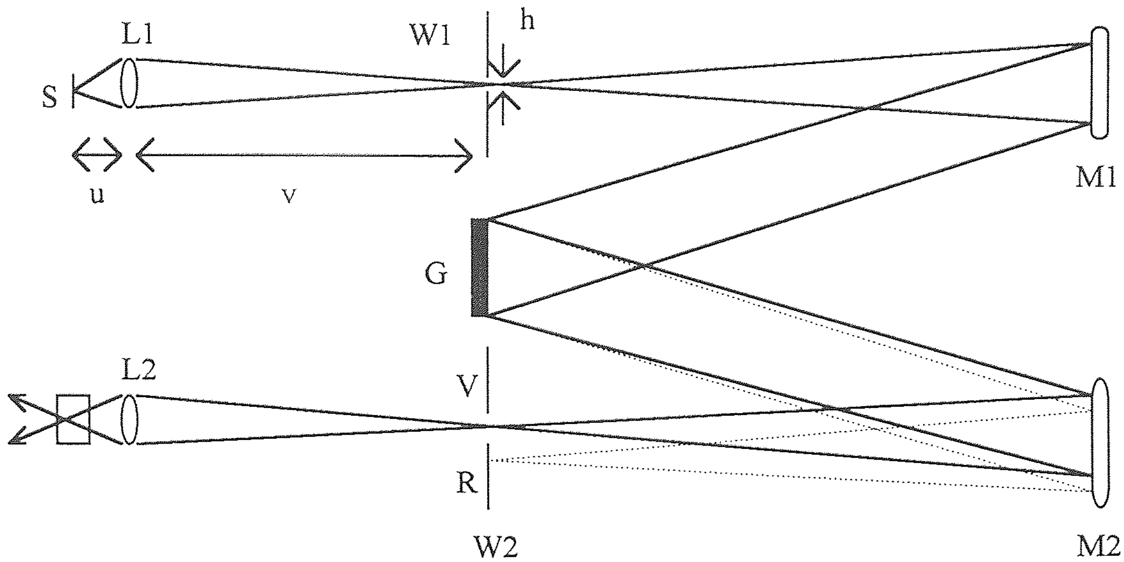
Gas or Gas Mixture	Active Species	Principal Laser Wavelengths (μ)	Remarks
He-Ne	Ne	0.6328, 1.15, 3.39	cw*
Ne	Ne	0.5401, 0.6143,1.15	pulsed high gain
Ne	Ne^+	0.3323, 0.3378, 0.3392	cw or pulsed
Ar	Ar^+	0.4765, 0.488, 0.5145	cw or pulsed
Kr	Kr^+	0.5208, 0.5309, 0.5862, 0.6471	cw or pulsed
Xe-He	Xe	3.506, 5.574	cw high gain
Xe	Xe^+	0.4603, 0.5419, 0.5971	cw or pulsed
Ne-O_2, Ar-O_2	O	0.8446	cw
N_2	N_2	0.3371	pulsed high gain
Air, N_2	N^+	0.5679	pulsed
He-Cd	Cd^+	0.3250, 0.4416	cw high efficiency

The argon ion laser, which we used in this experiment, generates visible light at several wavelengths in blue and green region (488 nm ~ 514.5 nm). And the helium-neon laser is chosen at 632.8 nm (red). Both of the lasers are cw (continuous wavelength).

2.3 Monochromator

2.3.1 Working Principle

All monochromator systems contain the following components (see figure 2.2). An entrance slit, W_1, of adjustable width; a collimator, M_1, which may be a mirror or a lens; a dispersing element, G, which may be a grating or a prism; a mirror, M_2, or a lens, to focus the dispersed light; and an exit slit, W_2, also of adjustable width. We shall assume for the sake of simplicity that the heights of W_1, W_2 and the focal length of mirror M1 and M2 are equal. Light from the source S is passed via the lens, L_1, to the entrance slit, W_1, of the monochromator. The diverging light from W1 is rendered parallel by the mirror or the collimator, M_1, and passed to the dispersing element, G. Different wavelengths of light leave G at slightly different angles and the dispersed light is focused by the mirror or lens, M_2, on the plane VR, so that each specific wavelength produces an image of the entrance slit at some point along VR. The images corresponding to the various wavelengths in the light from the source thus form a band of light varying in wavelength from one end of VR to the other, i.e. a spectrum of the source. The exit slit W_2, is situated at a suitable point along VR and thus selects a narrow band of wavelengths which are passed to the specimen (or the detector as the case may be), via another lens L_2, or other suitable optics. On rotating the dispersing element G, the spectrum VR moves laterally across the exit slit so that any particular wavelength region can be made to pass to the specimen or the detector. The wavelength may be selected by manual rotation of G, or more frequently in spectrometers, G may be rotated at a constant speed by a synchronous motor so that the complete spectrum can be scanned automatically and presented on the computer screen.



Grating

Figure 2.2

Working Principle of a Monochromator

Refer to Figure 2.2, let F be the aperture of the monochromator defined by,

$$F = d / f$$

Eq.(2-1)

where f is the focal length of the collimating mirror M_1, and d is the effective width of the dispersing element.

The relationship between u and v is then given by the lens magnification formula as follows,

$$v / u = h / s$$

Eq.(2-2)

where h is the height of the entrance slit and s the height of the source.

The cross section of the parallel beam of rays from M_1 dispersed by the prism or grating, is assumed to be square, and hence its area is d^2. The solid angle, θ_2, subtended by the effective area of M_1 at the entrance slit, W_1, is thus given approximately by,

$$\theta_2 = d^2 / f^2 = F^2$$

Eq.(2-3)

From Eq.(2-2):

$$\theta_1 / \theta_2 = v^2 / u^2 = h^2 / s^2$$

Eq.(2-4)

Therefore,

$$\theta_1 = F^2 h^2 / s^2$$

Eq.(2-5)

If the area of the source is A_u, the area of the image of the source, A_v, on the plane of the entrance slit is given by,

$$A_v / A_u = h^2 / s^2$$

Eq.(2-6)

Hence, the intensity of lumination on the plane of the entrance slit is given by,

$$I_\lambda \theta_1 / A_v = I_\lambda F^2 / A_u \qquad \textit{einstein}\ \text{mm}^{-2}\text{sec}^{-1}\text{nm}^{-1}$$

Eq.(2-7)

For low values of F, it is not possible to increase the intrinsic intensity from a source by focusing with a lens. What we have achieved with the lens is to ensure that the entrance slit is filled with light no matter how great its size in relation to the size of the source. If the entrance slit of the monochromator is opened to a width w_1, the amount of light passing through the slit and collected by the effective area of M_1 is,

$$(I_\lambda / A_u)\, F^2 h w_1 \qquad \textit{einstein}\ \text{sec}^{-1}\ \text{nm}^{-1}$$

Eq.(2-8)

Thus to collect the greatest possible amount of light at the entrance slit we require a source with the greatest possible value of I_λ/A_u, i.e. the greatest intrinsic brightness (*einsteins* mm^{-2} sec^{-1}). For a source of a given power, this implies the smallest area of source, and with a shape to match that of the fully opened entrance slit.

2.3.2 Light Gathering Power

To measure the amount of light within a given band width that the monochromator will pass from a source of given brightness, we have to define the light gathering power **(LGP)**,

$$\textbf{LGP} = F^2 hm = \alpha h d^2 / f$$

Eq.(2-9)

where α is the angular dispersion, and m is linear dispersion of the monochromator usually is expressed in millimeters per nanometer.

The relationship between α and m is given as follows,

$$m = \alpha f$$

Eq.(2-10)

where f is the focal length of the mirror, M_2, and let α is measured in radians, the distance, m, along VR occupied by the unit band of wavelengths is

From Eq.(2-9), therefore, the light gathering power depends on the slit height, h, the focal length, f, the angular dispersion, α, and the working height, d, of the prism or grating, and proportions to the square of the aperture (f^2), slit height, h and the linear dispersion (m).

The experiment use a grating monochromator. Therefore, to isolate high intensities of light of comparatively broad band width we use large **LGP**.

2.4 Filters

Filters may be used as an inexpensive substitute for a monochromator of large **LGP** value, or they may be used in conjunction with a monochromator to increase its efficiency. They may be used for isolating a band of wavelengths of exciting light when measuring a luminescence emission spectrum, or for isolating the required luminescence band when measuring an excitation spectrum. In our experiment filter is used for both purpose and monochromator is then dispensed with entire. It should be noted that when high intensities of exciting light are required it is sometimes an advantage to use a broad band pass filter to separate a combination of two or more spectrum lines. To obtain a sufficiently narrow band-pass, it is customary to combine a broad band-pass filter having the required long-wavelength cut-off with a suitable short-wavelength cut-off filter. If a weak luminescence spectrum in one spectral region has to be measured in the presence of a very intense luminescence in another spectral region, it is often desirable to use a filter to prevent the intense luminescence from entering the analyzing monochromator. Because in this experiment we use a laser beam as a light source with a very narrow spectrum, or in fact, a line spectrum. Therefore, instead of using a monochromator to isolate the exciting light, we use a filter. Furthermore, for high accuracy measurement, we use long-pass cut-on filter, because in our experiment we expect to have photoluminescence in IR range.

2.5 Detector

There are three types of physical instruments that are generally used to detect or measure light in the visible and ultra-violet regions of the spectrum -- the thermopile, the various types of photo-cell, and the photo-electron multiplier tube (photomultiplier). For the measurement of photoluminescence, particularly at low intensity, the photomultiplier is used almost exclusively because of its extremely high sensitivity, but its sensitivity falls dramatically at the wavelength beyond 1.1 micrometer. Therefore in our experiment (IR range), we tried to use a pyroelectric detector as the standard detector because it is a very low noise, stable, (within 0.2μ to 5μ) detector.[20] Unfortunately, this experiment did not succeed because of undetectable signal. Next we turned to a PbS detector which is a photoconductive type of detector. Because the PbS detector does not have a flat response. We need to obtain the spectral dependence of the response of the PbS detector and normalize it.

CHAPTER 3

CALIBRATION OF PHOTOLUMINESCENCE EXPERIMENT

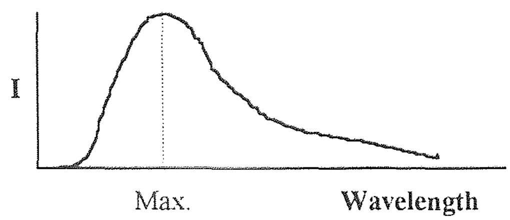
3.1 Calibration of Emission Spectra

When the monochromator is scanned at constant slit width and constant photomultiplier sensitivity, the curve obtained is the apparent or uncorrected emission spectrum. Each of the component will contribute signal which is undesired to the actual spectrum; we only consider three factors here, light source, monochromator and detector (PbS detector). To determine the true spectrum, the apparent curve has to be corrected under the consideration of those three factors by using Eq.(3-1).

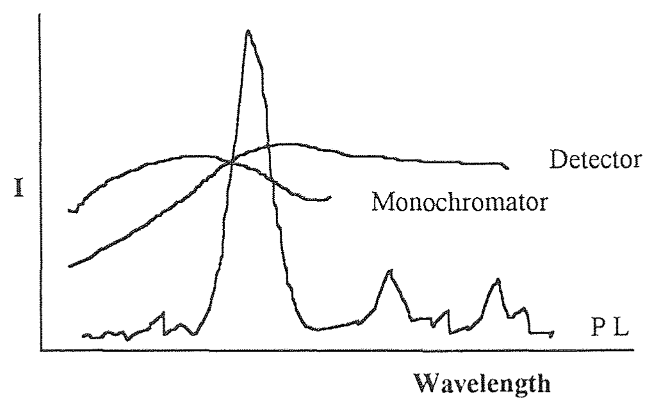
$$I_{normalized} = I_{read\ back} / M_n \cdot D_n$$

Eq.(3-1)

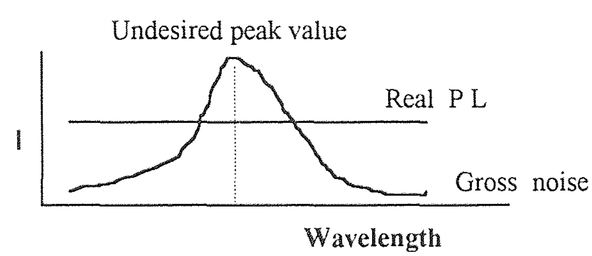
where I is the intensity, M_n is the normalization coefficient for the monochromator and so is D_n for the detector. All of them are wavelength-dependent.



(a). 500K black body radiation light source (BBRLS).



(b). Read back PL spectrum (uncorrected) depends on the instruments we used to measure it.



(c). Real PL spectrum will be distorted by the gross noise.

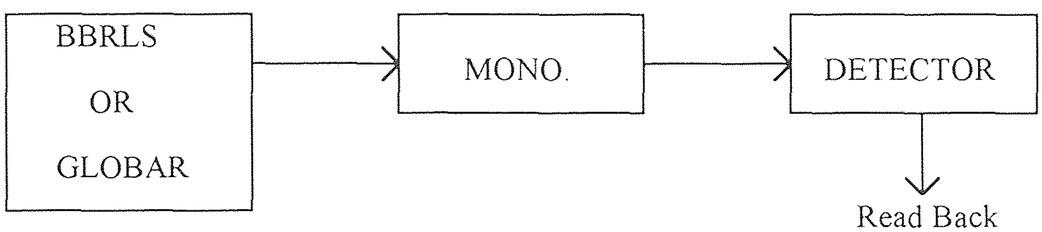
Figure 3.1

Distribution Curves

Figure 3.1(a) illustrates the distribution curve of black body radiation at 500 K, it is a pretty good ideal light source because we can predict the behavior easily by *Planck's*

Law. The experiment was not feasible because the light source is in the invisible IR range, making the alignment very difficult. Therefore, we use globar to do the calibration of the detector. Figure 3.1(b) shows that the component might contribute noise to the real signal and end up with distorted spectrum. Calibration is very important in order to get a correct spectrum, especially when a flat-like part of the luminescence spectrum is distorted by gross noise with peaks which can seriously affect our results as shown in Figure 3.1(c).

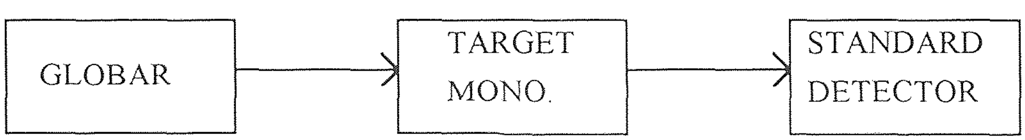
The following is the block diagram for the three components that we are using in the experiment, and need to be take into account when we are trying to correct the spectrum.



We need to calibrate each component individually in any event. Block diagrams are shown as follows,

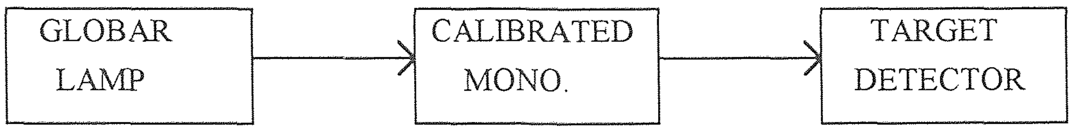
To calibrate monochromator -- M_n :

We use globar as light source and pyroelectric detector as standard detector to calibrate monochromator,



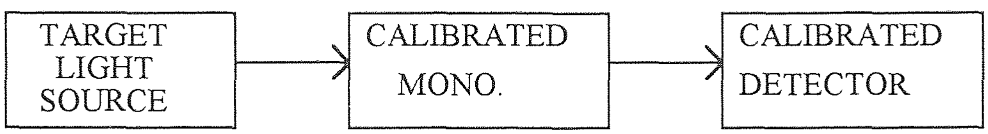
To calibrate detector -- D_n :

Employ the calibrated monochromator that we have done from last procedure and black body radiation light source to calibrate detector (PbS detector).



To calibrate light source :

Since the monochromator and the detector have been calibrated, the light source thus can also be calibrated.



3.2 The Radiation Laws of Stefan, Wien and Planck

The total radiant flux per second from a black body (full) radiator maintained at a temperature TK (where T_0 is the temperature of the surroundings) is given by *Stefan's equation*,

$$M = \sigma(T^4 - T_0^4)$$

Eq.(3-2)

where M is radiant emittance in the unit of watt/m^{-2}, and $\sigma = 5.67 \times 10^{-8}$ $W \cdot m^{-2} \cdot k^{-4}$.

Within the wavelength interval λ to $(\lambda + d\lambda)$, the energy per unit volume is,

$$E_\lambda d\lambda = C_1\lambda^{-5}d\lambda \,/\, \exp(C_2/\lambda T)-1$$

Eq.(3-3)

$$C_1 = 8\pi hc, \quad C_2 = hc/k$$

c : speed of light, k : *Boltzmann constant*, h : *Planck's constant*

For a given value of T, E_λ will be a maximum at a certain wavelength λ_m. By differentiating Eq.(3-3) with respect to λ and equating to zero gives the *equation of Wien*,

$$\lambda_m T = c \cdot h \,/\, 4.965$$

Eq.(3-4)

where $\lambda_m T$ is recognized as *Wien's constant.*

Therefore,

$$\lambda_m \propto T^{-1}$$

Eq.(3-5)

Eq.(3-2) the correction is needed because the thermal emissivities are not unit,

$$M = \varepsilon\sigma(T^4 - T_o{}^4)$$

Eq.(3-6)

where ε is the emissivity.[5][16]

Planck's great contribution in the field of blackbody radiation was to treat the energy of the electromagnetic standing waves, oscillating sinusoidally in time, as a *discrete* instead of a *continuous* quantity. This resulted in a successful explanation of blackbody radiation. *Planck's blackbody spectrum* obtained from his theory is given in Eq.(3-7). The experimental results are in complete agreement with *Planck's formula* at all temperature. See Figure 3.1(a).

$$\rho_T(\nu)d\nu = \frac{8\pi\nu^2}{c^3}\frac{h\nu}{e^{\frac{h\nu}{kT}}-1}d\nu$$

Eq(3-7)

$$\nu = c/\lambda,$$

$$\rho_T(\lambda)d\lambda = \frac{8\pi hc}{\lambda^5}\frac{d\lambda}{e^{\frac{hc}{\lambda kT}}-1}$$

Eq(3-8)

It is convenient in analyzing experimental results to express the *Planck blackbody spectrum* in terms of wavelength λ rather than frequency ν. See Eq.(3-8).[16]

3.3 Experiment

A. Use Ar+ Laser (λ=514.5 nm) and He-Ne Laser (λ=632.8 nm) as the light source.

A cw(continuous wavelength) laser beam differs from conventional sources is that it is highly monochromatic, extremely intense, and in some cases polarized.

B. Samples :

1. Material : InGaAs, GaAs as substract, wavelength of photoluminescence : 1.0μ
2. Material : GaSb, GaSb as substract, wavelength of photoluminescence : $I_{peak}(1.56\mu)$

C. Apparatus :

1. Monochromator : Model # HR320 (Instrument SA, Inc.). Grating change from 1200 *l*/mm to 300 *l*/mm; actual wavelength thus should multiply by 4.
2. JY488/SPEX488 Interface for Spectrometer Controller to communicate with computer via GPIB.
3. Detector : PbS detector.
4. Lock-in Amplifier : Model # 5208 (EG&G Brookdeal Electronics).
5. (a) 400 mW Argon Ion Laser (457-514 nm); Laser Power--0.2 V/mW, tuned at the wavelength of 514 nm (green).
   (b) 15 mW Helium-Neon Laser (632.8 nm).
   (c) Globar (IR emitter).
6. Interference Filter : Removes any undesired wavelengths from the laser beam, guarantee that the wavelength passing through is unique.
7. Chopper : Model # CTX-534 (Laser Precision Corp.); frequency -- 100~200 Hz.
8. Computer : IBM XT™ with GPIB interface card.

D. Temperature : Room temperature (293K).

3.3.1 PbS Detector

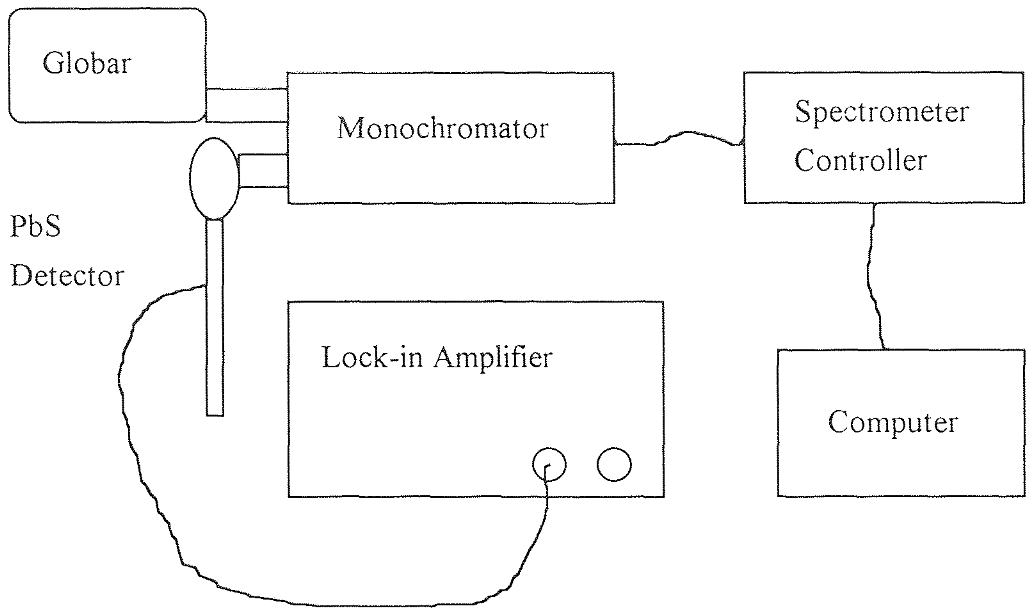


Figure 3.2

Setup for the experiment to Acquire
the Spectrum of PbS Detector

To compare the photoluminescence spectra with or without applying the spectrum of PbS detector can help us to determine the influence of the detector to the photoluminescence spectra. As indicated in Figure 3.2, we use globar to get the spectrum of PbS detector from 960 nm to 3240 nm, assuming that the spectrum of the globar in the above range is flat (according to the manufacturer's catalog, the spectrum of the globar in the range of 1.5μm~4μm is nearly flat).[20] Therefore, the normalized spectrum of the PbS detector can be obtained by dividing the intensity of each point of the curve by the maximum intensity. Figure 3.3 is the spectrum of PbS detector and, Figure 3.4 is the normalized spectrum.

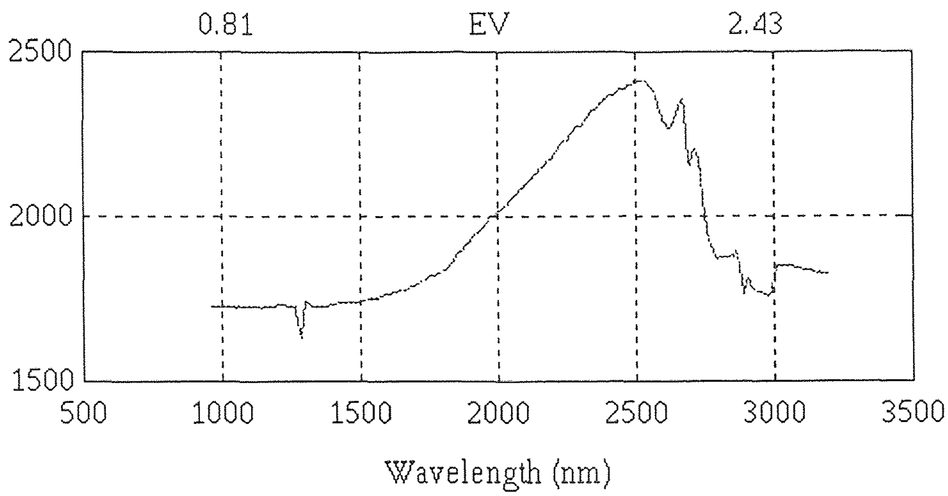


Figure 3.3

PbS Detector with Globar Lamp

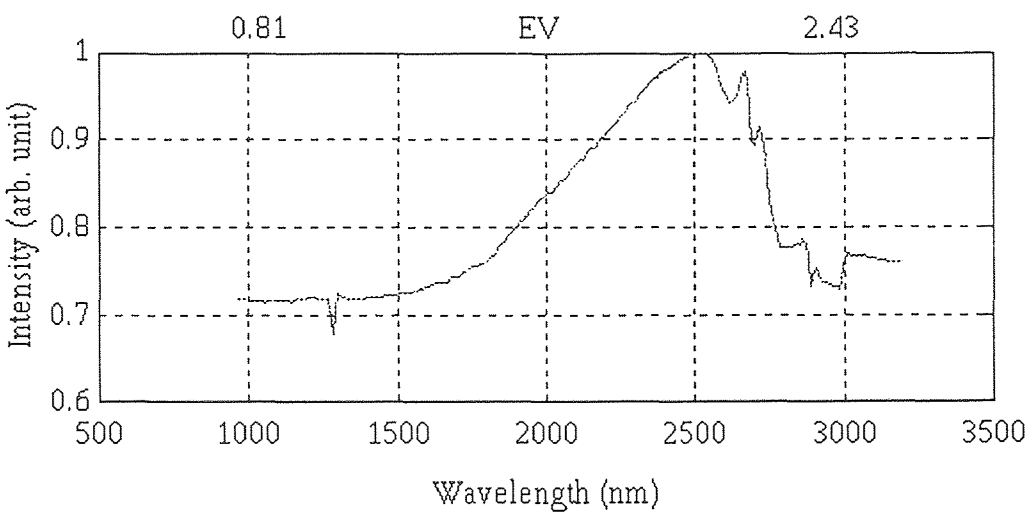


Figure 3.4

PbS Detector with Globar Lamp

Normalized spectrum

3.3.2 Blank Test for Argon Ion Laser and Helium-Neon Laser

We need to process blank test to find out the spectrum of the laser beams before we proceed the experiment with the sample. The spectra obtained from this procedure are used to compare with the photoluminescence spectra we are going to get later on for the purpose of finding the real photoluminescence spectrum. Setup the equipment as indicated in Figure 2.1, instead of using sample, we put aluminum plate to reflect laser beam, because we can't expect photoluminescence for aluminum in IR at room temperature (293K).

a. Argon Ion Laser :

Process the test through the range of 1.0μm~2.7μm. The result is as follow,

There are four peaks found within this range,

1032 - 1072* nm (gain : 5mV)

1520 - 1600* nm (gain : 1mV)

2060 - 2100* nm (gain : 20mV)

2560 - 2620* nm (gain : 50mV)

Figure 3.5 is one of the spectrum in the range of 1520 - 1600 nm, and others are in appendix E.

b. Helium-Neon Laser :

Process the test through the range of 1.26μm~2.60μm, there are three peaks found,

1260 - 1320* nm (gain : 200μV)

1900 - 1960* nm (gain : 500μV)

2520 - 2600* nm (gain : 5mV)

Figure 3.6 is one of the spectrum in the range of 1260 - 1320 nm, and others are in appendix E.

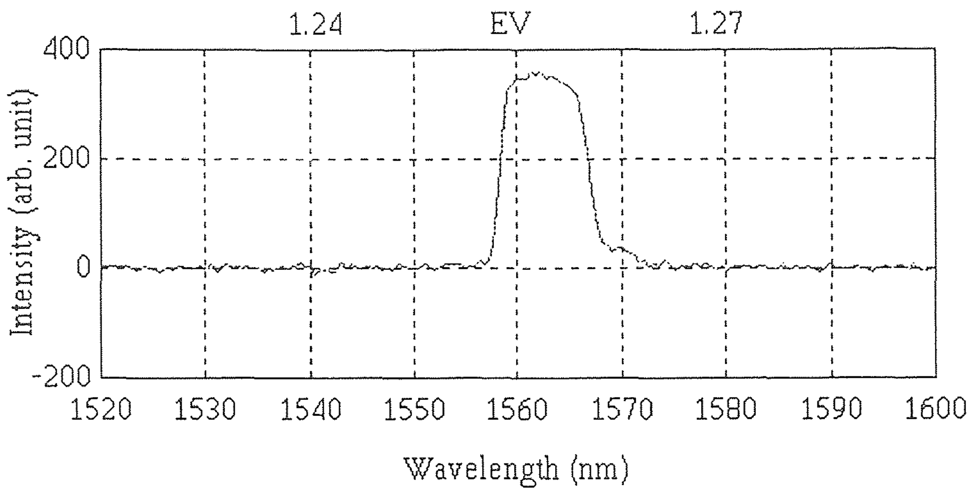


Figure 3.5

Argon ion Laser

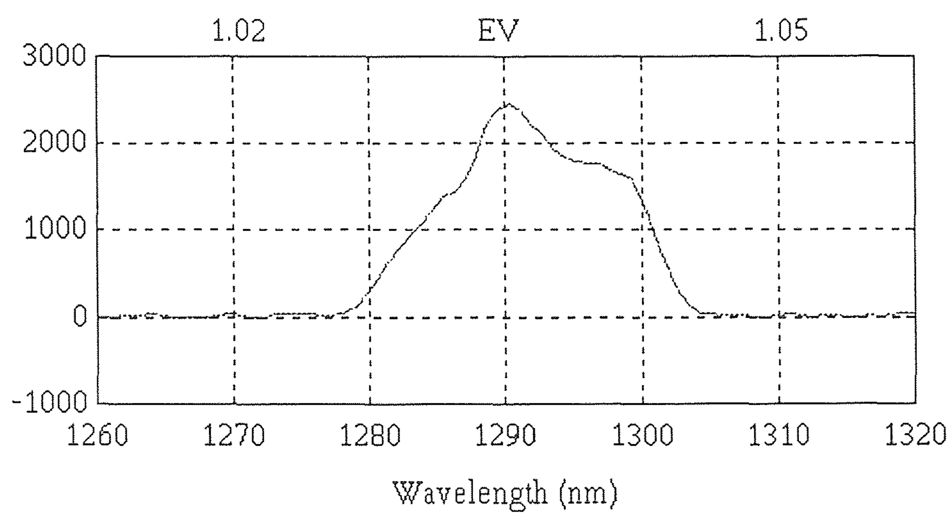


Figure 3.6

Helium-Neon Laser

3.3.3 GaSb Sample

Set up all the devices, components and peripherals as was shown in Figure 2.1. Then choose GaSb as the sample. We expect to have photoluminescence at the wavelength around 1.56μ, which is in the infrared region.

a. Use Argon Ion Laser as Light Source:

Scan from 1.0μ to 2.7μ, four peaks related to blank test were found,

1032 - 1112* nm (gain : 1mV)

1544 - 1600* nm (gain 200μV)

2040 - 2120* nm (gain : 10mV)

2548 - 2628* nm (gain : 20mV)

1544 - 1600 nm is the range that we expect to have photoluminescence. In comparison of the spectrum of blank test and the spectrum of sample GaSb in this range, it does contain deviation. Nonetheless, there are no differences between both in other ranges. Figure 3.7 is one of the spectra and others are in appendix E. Figure 3.8 is the comparison between the blank test of argon ion laser and GaSb spectrum.

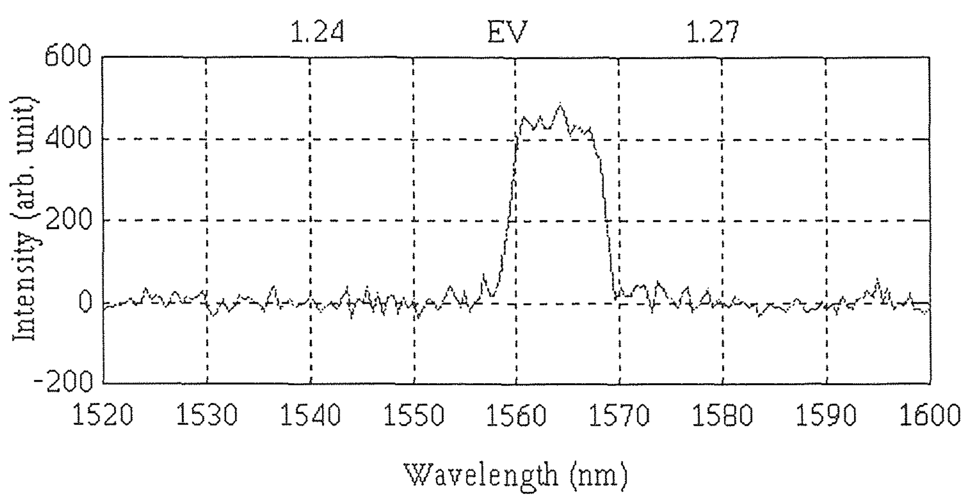


Figure 3.7

Sample : GaSb, Light Source : Ar Ion Laser

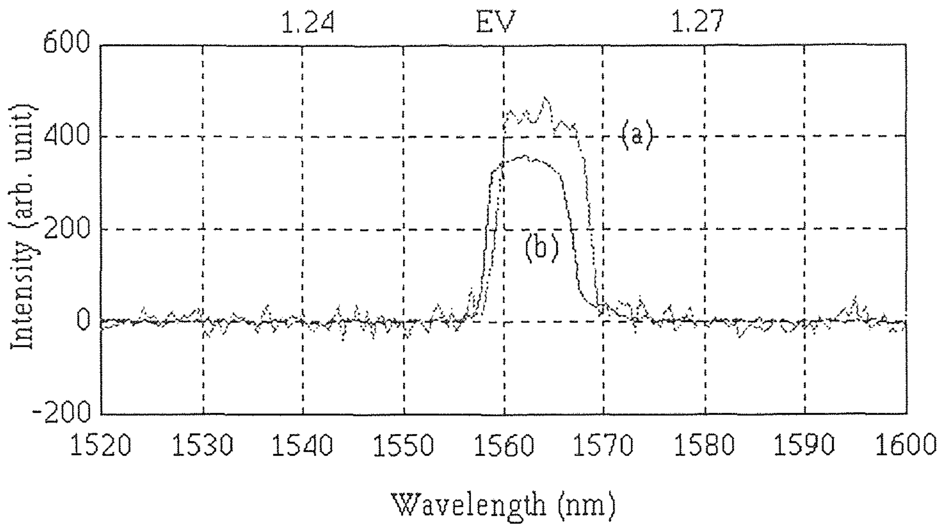


Figure 3.8

(a) Spectrum for GaSb, (b) Spectrum for the Blank Test of Ar Ion Laser

b. Use Helium-Neon Laser :

Scan from 1.24μm to 2.6μm, three peaks related to blank test were found,

1240 - 1320* nm (gain : 10mV)

1900 - 1960* nm (gain : 50mV)

2520 - 2600* nm (gain : 2mV)

Figure 3.9 is one of the spectra and others are in appendix E. Figure 3.10 is the comparison between the blank test of He-Ne laser and GaSb spectrum in the same range.

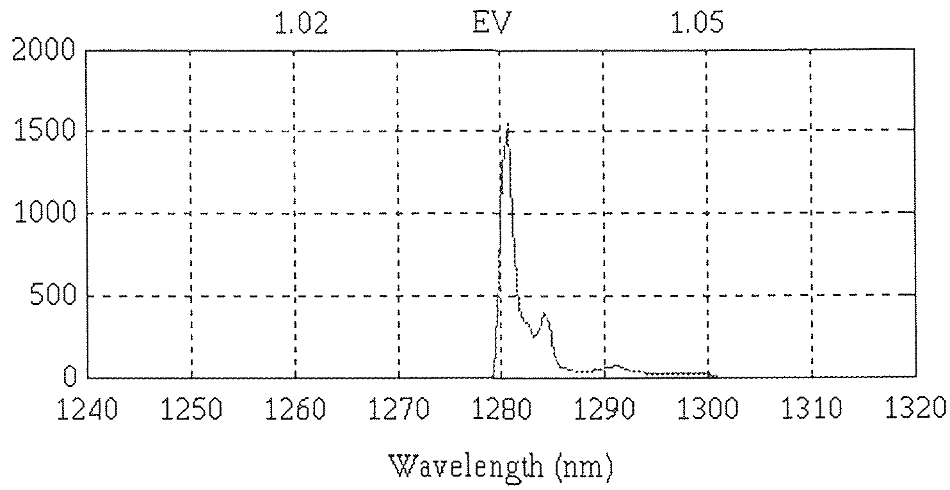


Figure 3.9

Sample : GaSb, Light Source : He-Ne Laser

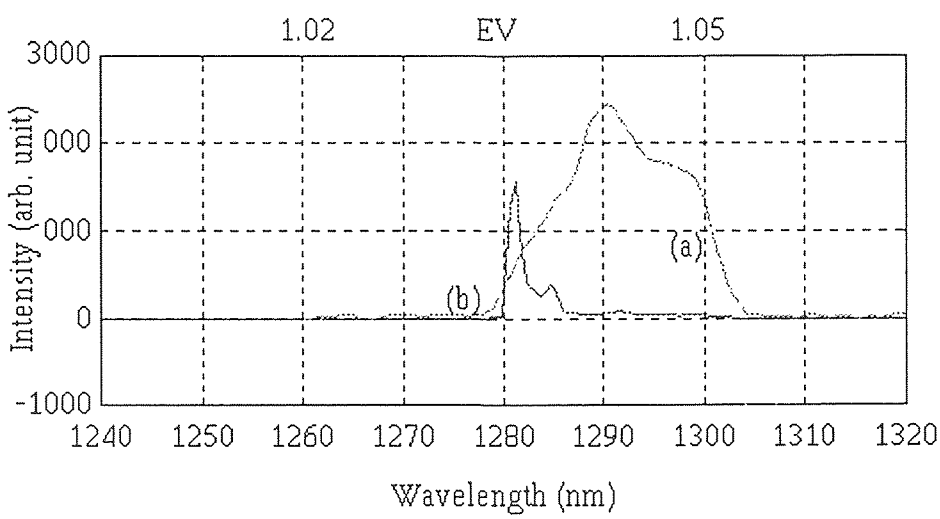


Figure 3.10

(a) Spectrum for the Blank Test of He-Ne Laser, (b) Spectrum for GaSb

3.3.4 InGaAs Sample

Scan from 1.0μm to 2.6μm, with He-Ne Laser as light source,

1272 - 1312* nm (gain : 2mV)

1900 - 1952* nm (gain : 2mV)

2544 - 2620* nm (gain : 10mV)

Figure 3.11 is one of the spectra and others are in appendix E. Figure 3.12 is the comparison between the blank test of He-Ne laser and InGaAs spectrum.

* The data was acquired via the application of BASIC program in appendix G.

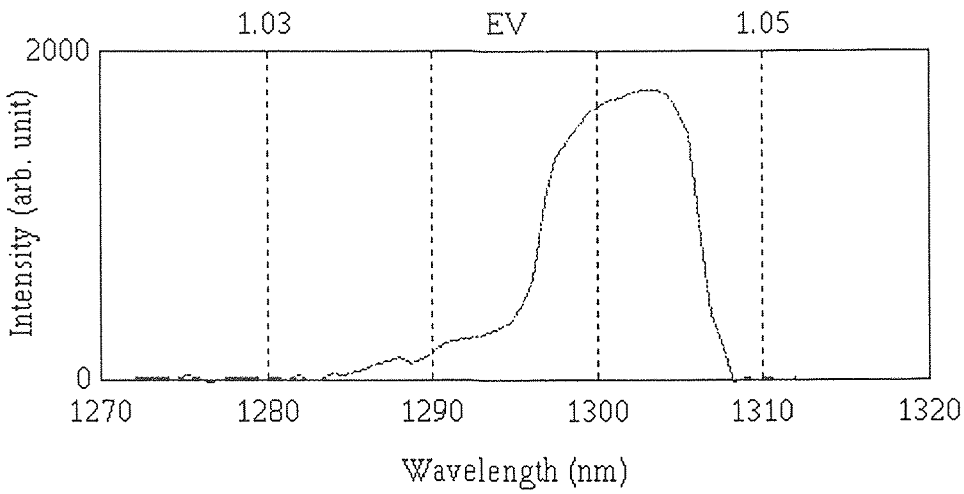


Figure 3.11

Sample : InGaAs, Light Source : He-Ne Laser

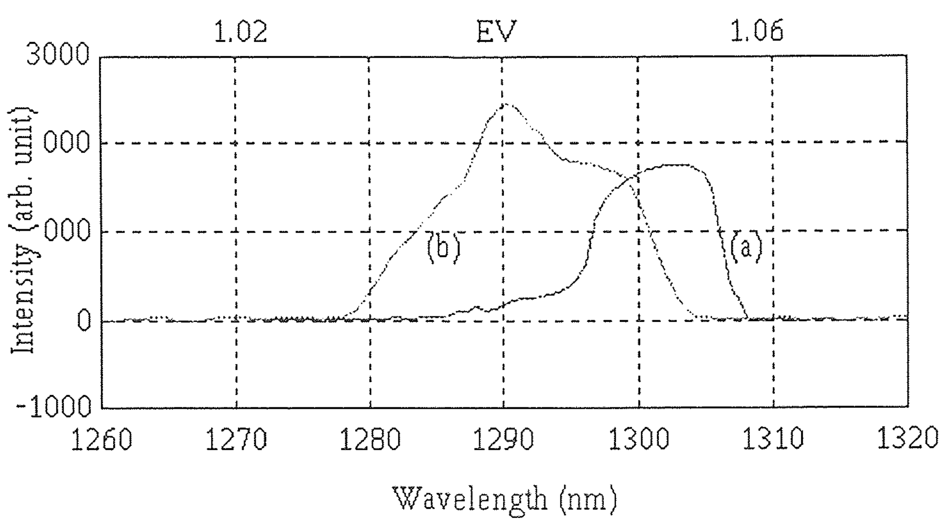


Figure 3.12

(a) Spectrum for InGaAs, (b) Spectrum for the Blank Test of He-Ne Laser

CHAPTER 4

DATA ACQUISITION

Refer to Figure 2.1 The photoluminescence emitted from the sample is detected by the photomultiplier, then we can get read-out signal from the lock-in amplifier. While, the monochromator is controlled by the spectrometer. **GPIB** is the interface card to make communication between peripheral devices and the computer via **IEEE488** interface bus (in the experiment, lock-in amplifier and spectrometer are the devices we need to make connection to). We use **BASIC** as the coding language to write the software to control all the devices and acquire data. We need to send command to the spectrometer about where to start scan, where to stop scan, and scan speed, then the spectrometer controls monochromator. Meanwhile, also read back the signal from the lock-in amplifier, plot the uncorrected spectrum on screen and stores it. The main task we are trying to do is to have real time data acquisition, data averaging and displaying. Thereafter, employ the spectrum and datum we obtained to calibrate and normalize all the components, eventually get the real spectrum.

4.1 IEEE488 Communication with a Computer

IEEE488 interface bus acts just like **RS232** serial. Via **GPIB** we cad connect up to 32 devices. However, depending on the configuration of the interface board may limited to 16 devices. First thing after we install our **GPIB** card and softwares is to configure **IEEE488** by running the software named *install.exe*. Just follow the instruction to setup all the configuration required to use **IEEE488** interface bus. The optional **488** interface is designed as an **IEEE488** *Talker/Listener*. The primary **IEEE488** address is set at the factory to 1, however, it can be reconfigured with a command to set the address from 1 to 31. **IEEE488** can take control of the instrument at any time by asserting the REM line. If **IEEE488** is initializing the system, the entire auto-baud sequence used for serial communication is skipped. The system is automatically forced into "INDEPENDENT MODE", which means that all the parameters needed to perform the experiment are sent at one time. These parameters include the monochromator positions, integration time, and gains. Once the experiment is defined in this manner using the command SET SCAN PARAMETERS, the command START SCAN can be issued. The host program, at this point, is free to perform other tasks until the scan is finished. If desired, data can be read "*on the fly*" as the scan is progressing, or after the scan is completed. For TIME BASE scans, the results will be much better if the data is read after completion of the scan, removing the uncertain host communication time from the interval between data points. For less time critical experiments, however, data can easily be read as soon as it is acquired in hardware. When controlling the DataScan from a PC program, the "INDEPENDENT ID" character (which is **ASCII** character 248 in DECIMAL, F8 in HEX) would be sent to force the DataScan into "INDEPENDENT MODE". This is the mode where it expects to receive the commands as a string of parameters to set up the instrument to run autonomously without host intervention. After all these have been done, the DataScan receives this "INDEPENDENT ID" character, it responds with the EQUAL character ("="). At this point you must send the character string "O2000" (the letter 'O', followed by the numbers '2', '0', '0', '0'), followed by the NULL character (which is 0 in DECIMAL, 0 in HEX). This transfers control to the main SPEX232/DataScan program, residing in Flash RAM or EPROM in the instrument. The instrument will respond with an ASTERISK character (*) to acknowledge that it is in the Independent Mode. At this point you are ready to send the more interesting commands. Fig. 4-1 is the flowchart of **IEEE488** start up procedure. Also refer to appendix G, the

BASIC program for this experiment. At the first 100 lines, they are the communication establishing procedures.[3][19]

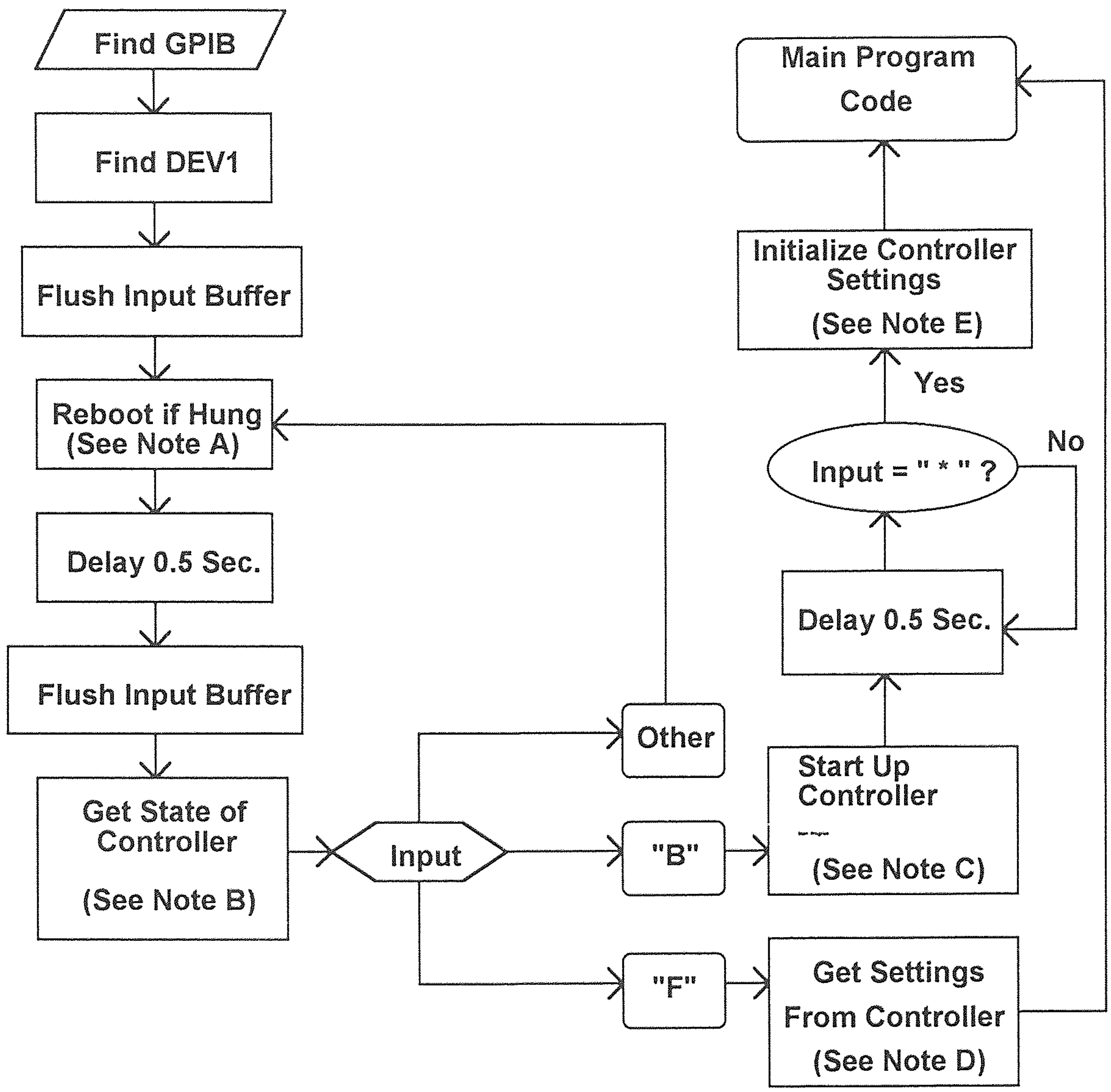


Figure 4.1

IEEE488 Start Up Procedure [19]

***** Notes on Figure 4.1 :

A: Send decimal value 222. This will force a reboot if hung from a previous run.

B: Send WHERE AM I command <SPACE> response will be "B" (for BOOT) or "F" (for MAIN) depending on the previous state of the controller.

C: Send "O2000" + NULL. Transfers control from the BOOT to the MAIN program. You must send the NULL.[19]

4.2 Establishing GPIB Communications

4.2.1 Install GPIB and Setup

The driver must be installed via computer's *config.sys* file. Connect **GPIB** cable between the interface board in the computer and **SPEX488** or **DS488**. If other devices are to be used on the same bus, it is recommended that they be disconnected temporarily, to reduce the possible sources of problems while establishing communication for the first time. Run the *ibconf.exe* file and set the configuration as shown in appendix A Assume that the first device name as DEV1.[3][19]

4.2.2 Test GPIB Communications

Run *ibic.exe* file, and issue these commands at the **:** prompt,

IBFIND GPIB0	Find the PCII board in the PC
IBFIND DEV1	Find the DS488 or SPEX488 at address 1
IBWRT " "	Send one space character (must be enclosed in Quotation marks)
IBRD 1	Read 1 character.

If succeeded, should received a **B** or an **F**, otherwise, will received error message. To gain further assurance of communication to the instrument, run *hwcfg488.exe* file by typing "hwcfg488 1" will read the configuration from the nonvolatile memory in the **DS488** or **SPEX488**. Then will see hardware configuration screen on the monitor. Press [F8] key to print out the configuration for future need.

CAUTION : DO NOT PRESS THE [F3] KEY, AS THIS WILL CHANGE THE CONFIGURATION IN THE INSTRUMENT.[3][19]

4.3 Data Acquisition

Appendix E is the BASIC program for data acquisition, data averaging and real time displaying. A few operating parameters must be set before the data acquisition can take place :

1. The integration time must be set.
2. The gain must be set for the channel being used.
3. The high voltage may need to be turned on.

4.3.1 Devices' Address Setup

In the program we set **GPIB0** at address **0**, spectrometer controller (DS488 or SPEX488) as **device 1** at address **1** and lock-in amplifier (mode 5208) as **device 4** at address **4**, on any other device connected to GPIB board.[3][8]

4.3.2 Several Precautions When Running the Program

Because of the restriction of the conventional memory (640KB), we have to access SCAN at the very first time when we get into the main menu, otherwise, insufficient memory error will occur (it depends on how many points we want to get). Also, for the resolution restriction, one horizontal line can has up to 640 dots. In other words, we can only have up to 640 scanning points at one time. There is an another problem has to be taken into account. Delay for response. Instrument delay is 0.5 sec., in the program has to fix the delay time for specific computer. After scan finished, the data will be stored in the file named "DATA", it is necessary to print out the file right away or rename it, or will lost the data when second time we run the program and do the scan process. The program has to be "exit" back to DOS environment, otherwise the files have been called will cause errors later on when we try to run the program again for the reason that the files were improperly closed. We also need to make couple backups for the source program in case of error occurs.

APPENDIX A

GPIB0 AND DEV1 SETUP REFERENCES

GPIB0

Primary GPIB Address	0
Secondary GPIB Address	None
Timeout Setting	T3s
EOS Byte	00H
Terminate Read on EOS	No
Set EOI With EOS on Write	No
Type of Compare on EOS	7-Bit
Set EOI W/Last Byte of Write	Yes
System Controller	Yes
Assert REN When SC	No
Enable Auto Serial Polling	Yes
Timing	500 nanoseconds
CIC Protocols	Yes
Interrupt Setting	None
Base I/O Address	02B8H
DMA Channel	1

DEV1

Primary GPIB Address	1
Secondary GPIB Address	None
Timeout Setting	T10s
EOS Byte	00H
Terminate Read on EOS	No
Set EOI With EOS on Write	No
Type of Compare on EOS	7-Bit
Set EOI W/Last Byte of Write	Yes
Repeat Addressing	No

* Other devices' setup are same as DEV1 shown above.[19]

APPENDIX B

GPIB CARD SETTINGS

GPIB-PCII	Default	Available
Base I/O Address (HEX)	2B8	000 to 3F8
DMA Channel	1	1, 2, 3, and Not Used
Interrupt Line (IRQ)	7	2, 3, 4, 5, 6, 7, and Not Used
7210 / 9914 Mode	7210	7210 And 9914
Shield Ground	Connected	Connected, Disconnected

Table B-1

Factory Default Settings and Available Configurations for GPIB-PCII Mode [3]

APPENDIX C

EG&G (LOCK-IN AMPLIFIER) COMMANDS
MODEL 5208

COMMAND	FUNCTION	PARAMETER RANGE
A1 n [0]	Turns the AUTORANGE on or off.	(0 or 1)
A2 n [0]	Turns the AUTOSET function on or off.	(0 or 1)
A3	Causes the AUTO-OFFSET routine to run.	
A4 n [0]	Turns AUTONORMALIZE function on or off.	(0 or 1)
A5	Controls RATIO function.	
A6	Reserved for future use.	
A7	Causes the AUTOPHASE routine to run.	
B n	Same as MSK command.	
C n	Same as DD command.	
CH n [0]	Turns Crystal Het on or off.	(0 or 1)
D n [1]	Selects which data DPM will display	(0 to 5)
DD n	Defines delimiter to be used between numbers sent to host computer (any ASCII character)	
E	Same as ERR command.	
ERR	Causes ERROR NUMBER of previous command to be sent.	
F n [1]	Selects FREQUENCY BAND	(0 to 2)
G n [1638]	Sets output gain (both channels in 5208).	(0 to 4095)
H n [1]	Selects mode of front-panel LED's.	(0 to 2)
ID	Reads model number.	
J n1 n2 [400 6]	Selects frequency of Oscillator and Reference.	
	n1 is the numerical frequency.	(100 to 1000)
	n2 is the band.	(0 to 7)
K n	Causes the routine associated with the specified key to run exactly the same as if the key had been pressed at the front panel.	(0 to 13)
L n [20]	Selects the sensitivity LIMIT.	(0 to 20)
M n [0]	Selects data indicated by panel meter(s).	(0 to 2)
MSK n	Specifies SRQ MASK byte.	(0 to 255)
N	Reads status of Overload Indicators	(1 to 7)
0 n1 n2 [0 0]	Sets OFFSET	

COMMAND	FUNCTION	PARAMETER RANGE
	n1 is offset	(0 to 1000)
	n2 is the polarity	(0 to 2)
00 n [1]	Turns Oscillator on or off.	(0 to 1)
OPTION n1 n2	Reads status of specified option	
	n1 (sent by host) specifies option.	(92 to 99)
	n2 (sent by 5207/8) specifies	
	presence of absence of option.	(0 to 1)
P n1 n2 [0 200]	Selects Phase Shift.	
	n1 is the Quadrant.	(0 to 3)
	n2 is the value.	(100 to 3900)
	(100 gives -2.5%)	
	(3900 gives 92.5%)	
Q1	Reads integer value of CH 1 output.	
Q2	Reads integer value of CH 2 output.	
Q3	Reads integer value of AUX output.	
Q5	Reads both CH 1 and CH 2 output values.	
R n [2]	Selects RESERVE.	(0 to 2)
S n [0]	Selects the SENSITIVITY.	(0 to 20)
T	Causes STATUS byte to be sent.	
T n1 n2 [6 1]	Selects TIME CONSTANT and dB/octave.	
	n1 selects the Time Constant.	(0 to 10)
	n2 selects the dB/octave	(0 or 1)
V	Same as VER command.	
VER	Turns EXPAND on or off	(0 or 1)
Z	Same as ST command	

[8]

APPENDIX D

MONOCHROMATOR TYPES

Type	Model	Steps/Unit	Min. Freq.	Max. Freq.	Ramp Time	Backlash	Base Grating
1	180D	16	2560	2560	1000	320	1200
2	MSD2	400	1000	36000	3000	20000	1200
3	1269	500	1000	36000	2000	25000	1200
4	1403	400	1000	28000	2000	20000	1200
5	1404	400	1000	28000	2000	20000	1200
6	1680	50	400	400	1000	500	1200
7	1681	50	400	400	1000	500	1200
8	1870B	50	400	400	1000	5000	1200
9	1870C	400	1000	32000	2000	20000	1200
10	1877A	50	400	400	1000	1000	1200
11	1877B	4000	1000	40000	2000	40000	1200
12	340E	50	400	400	1000	500	1200
13	220M	32	2560	2560	1000	320	1200
14	H10	20	300	450	2000	200	1200
15	H20V	20	300	450	2000	200	1200
16	H25	20	300	450	2000	200	1200
17	HR250	20	300	450	2000	200	1200
18	****	20	300	450	2000	200	600

Remarks:

- 500M, 500X, 750M, 1000M, 1250M, 1702, 1704 spectrometers when controlled via the MSD2 (Type 2).
- Model 1704 is also Type 7.
- Model 1877C is also Type 10.
- Model 1877D, 1877E are also Type 11, where Model 1877E is spectrograph stage.
- Model 1877E is also Type 12 and is prefilter stage.
- Model 270M is also Type 13.
- Model HR320 is also Type 17.
- **** Type 18 is Model HR320D.[19]

APPENDIX E

SPECTRA

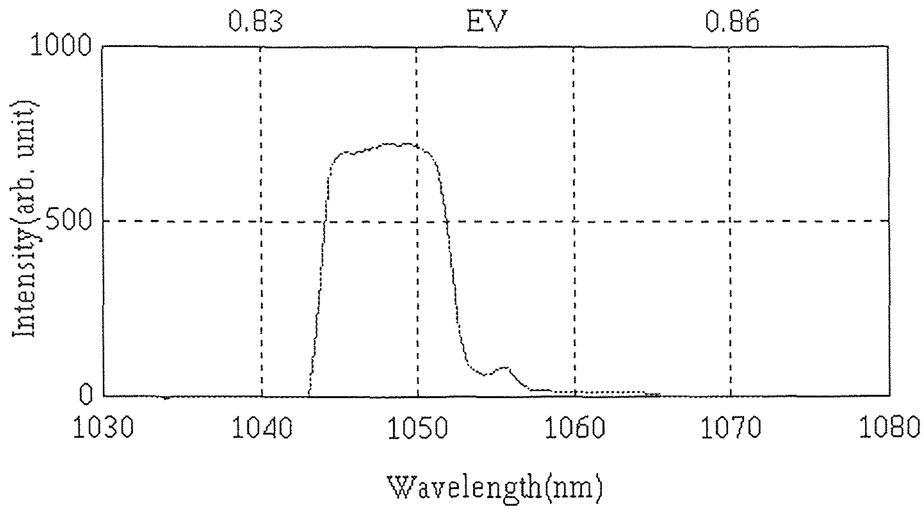


Figure E.1
Blank Test for Argon Ion Laser

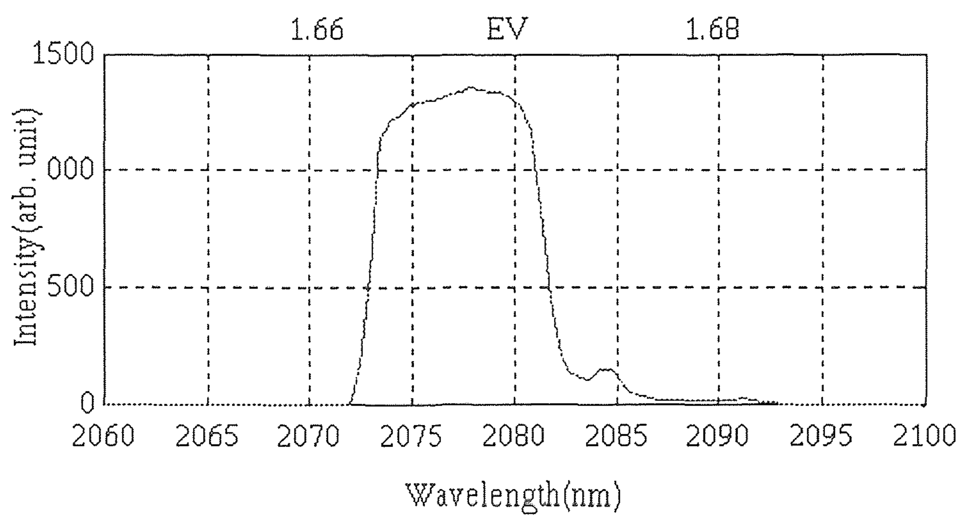


Figure E.2
Blank Test for Argon Ion Laser

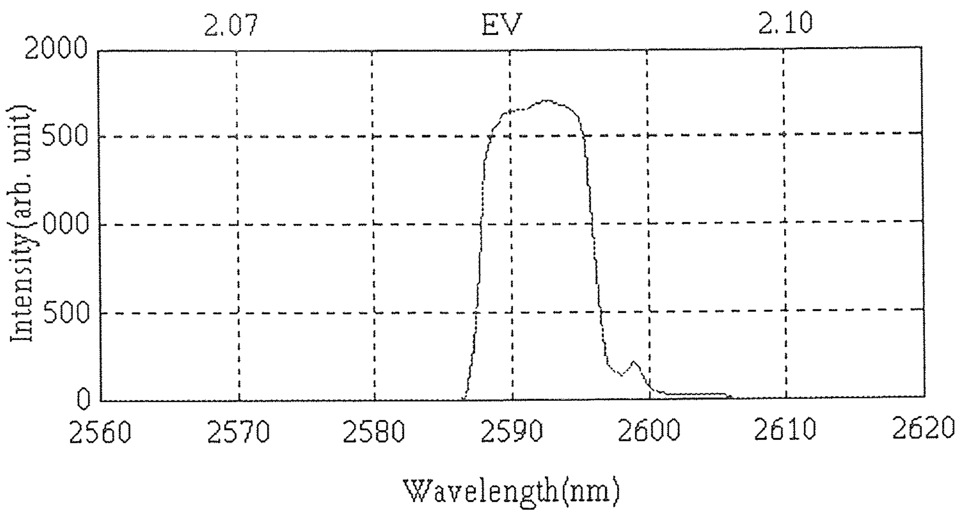


Figure E.3
Blank Test for Argon Ion Laser

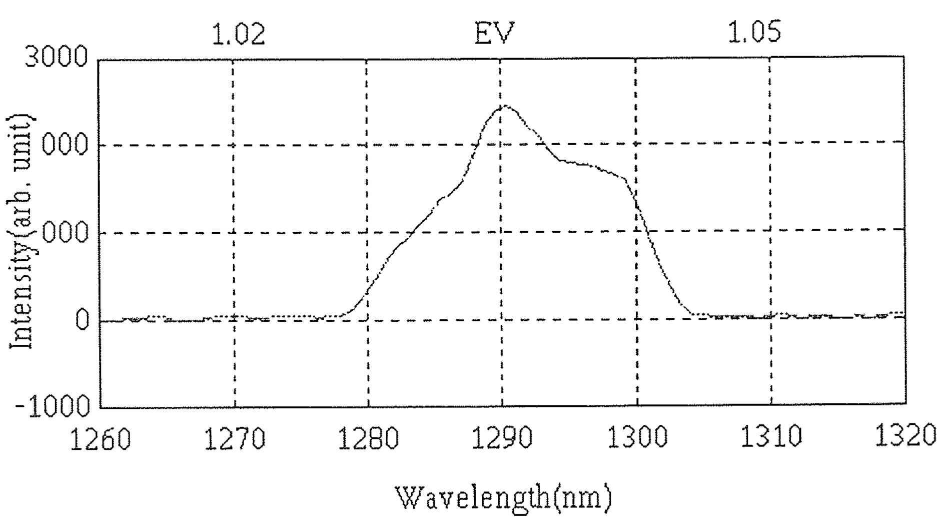


Figure E.4
Blank Test for Helium-Neon Laser

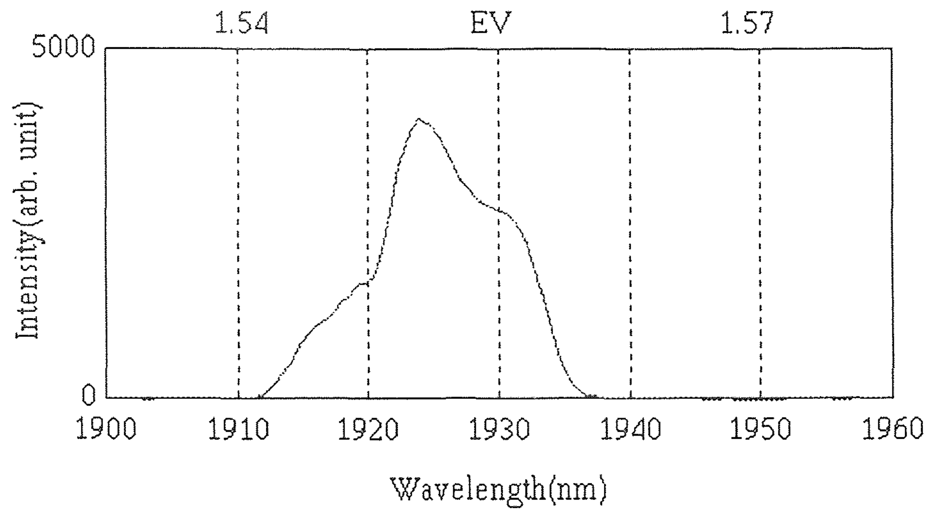


Figure E.5
Blank Test for Helium-Neon Laser

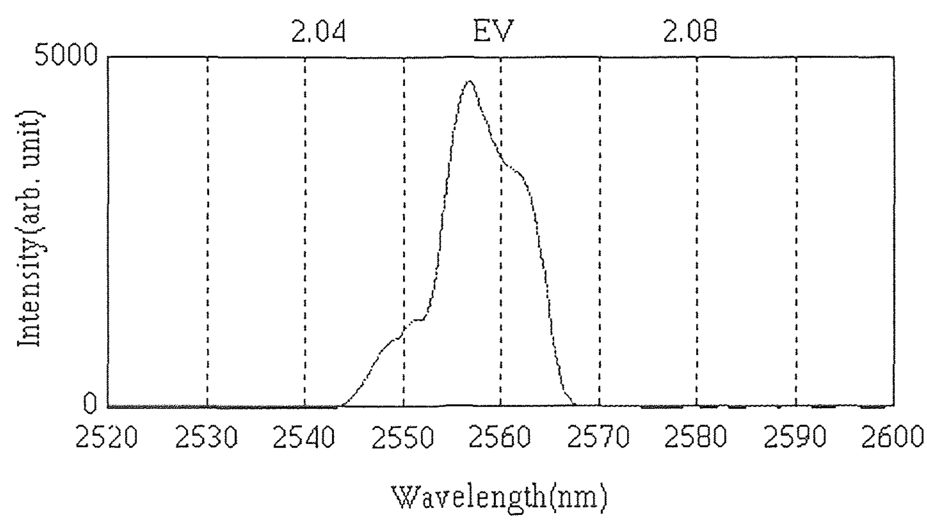


Figure E.6
Blank Test for Helium-Neon Laser

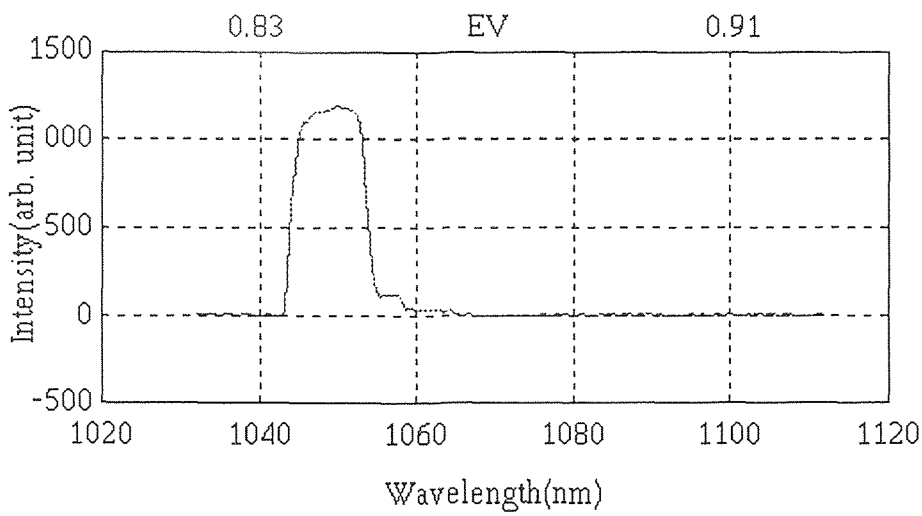


Figure E.7
GaSb Spectrum
Light Source : Ar Ion Laser

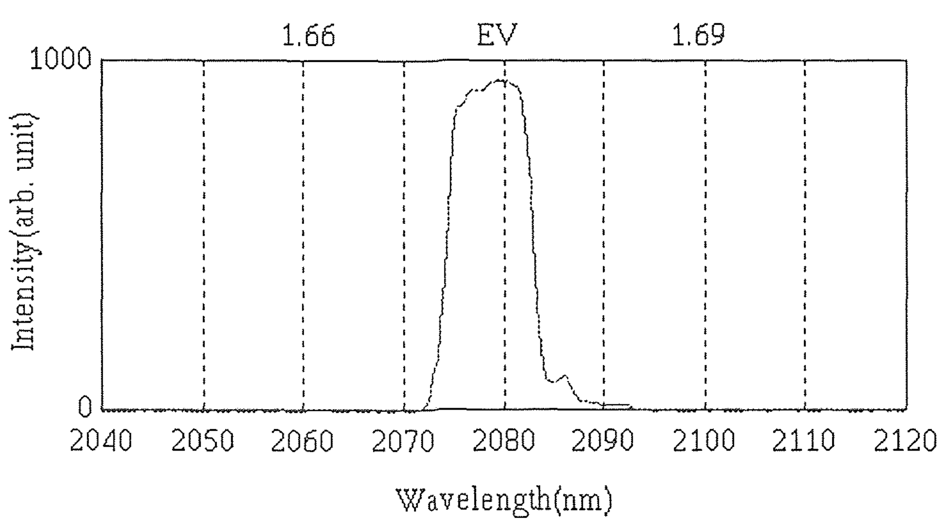


Figure E.8
GaSb Spectrum
Light Source : Ar Ion Laser

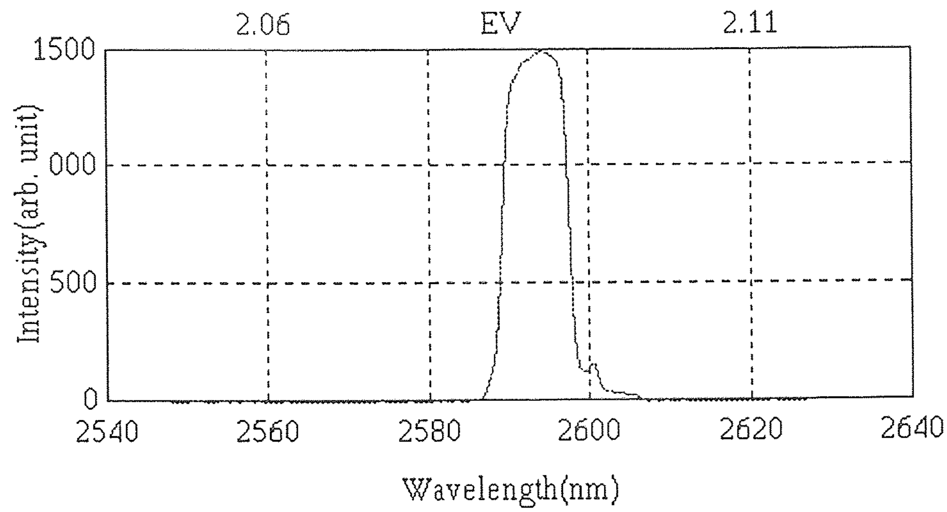


Figure E.9
GaSb Spectrum
Light Source : Ar Ion Laser

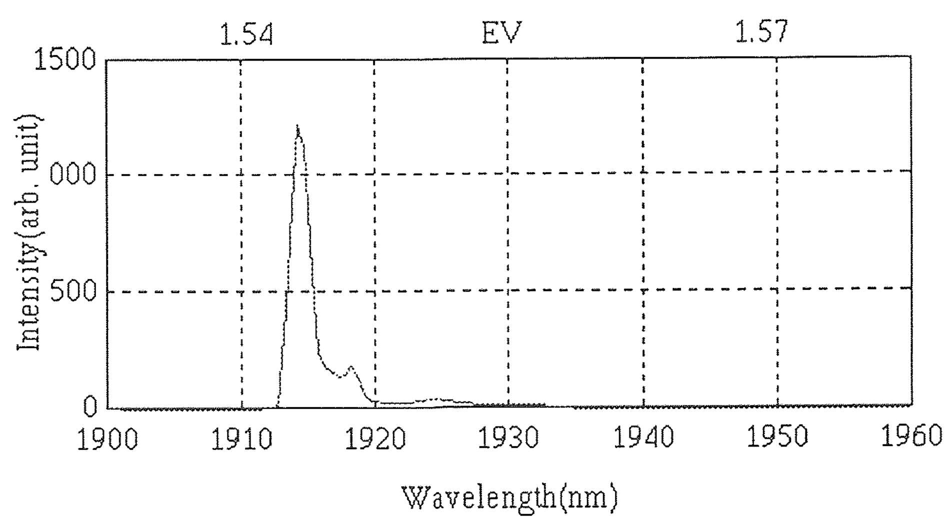


Figure E.10
GaSb Spectrum
Light Source : Helium-Neon Laser

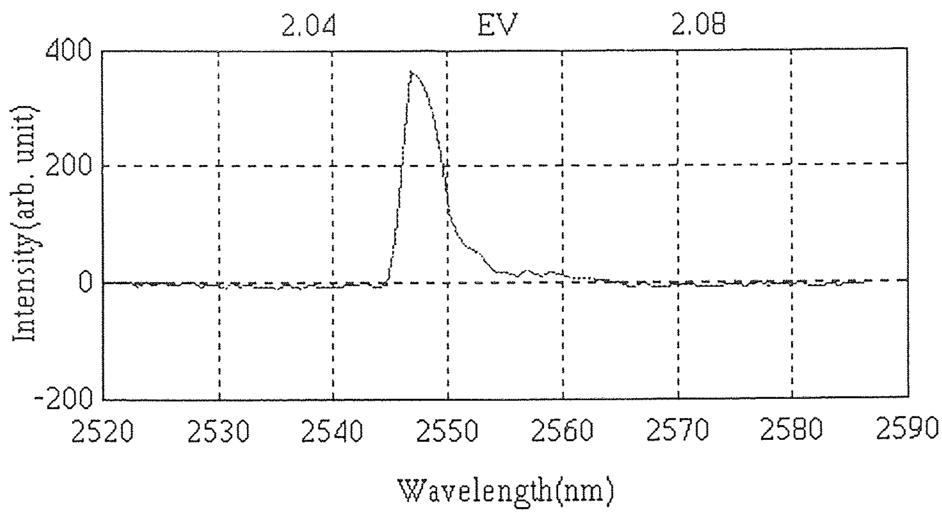


Figure E.11
GaSb Spectrum
Light Source : Helium-Neon Laser

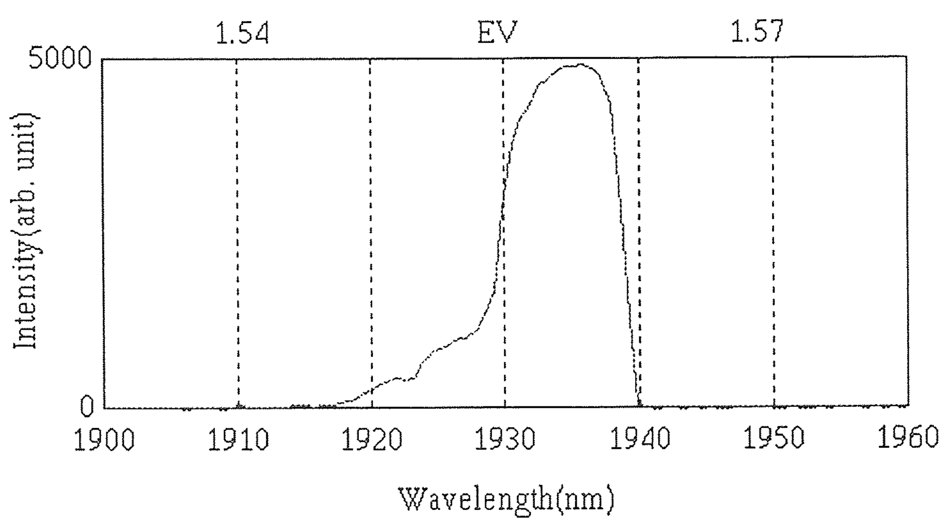
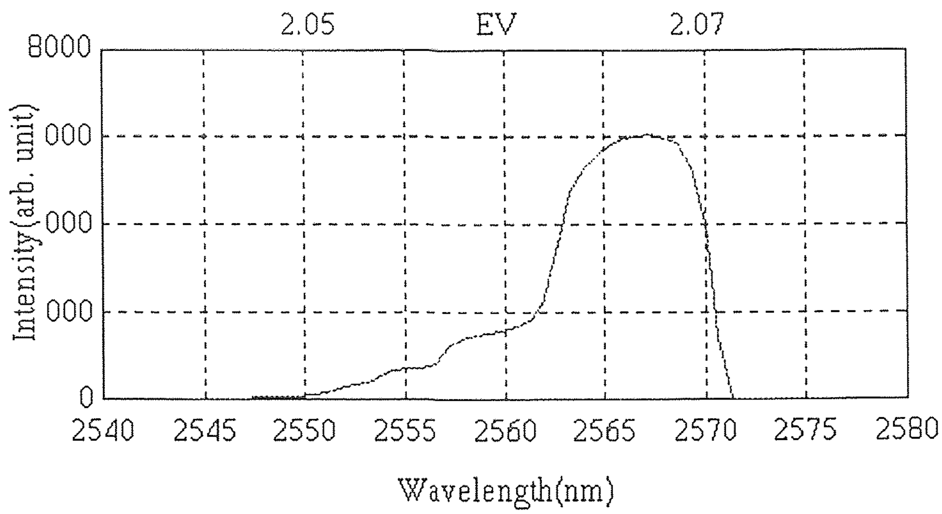


Figure E.12
InGaAs Spectrum
Light Source : Helium-Neon Laser



FigureE.13
InGaAs Spectrum
Light Source : Helium-Neon Laser

APPENDIX F

NORMALIZED SPECTRA

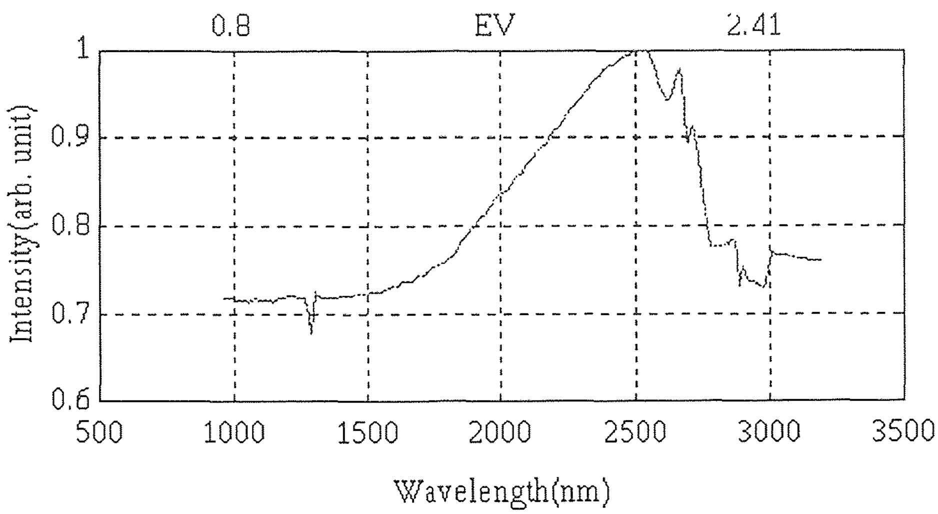


Figure F.1
Normalized PbS Detector Spectrum
Light Source : Globar Lamp

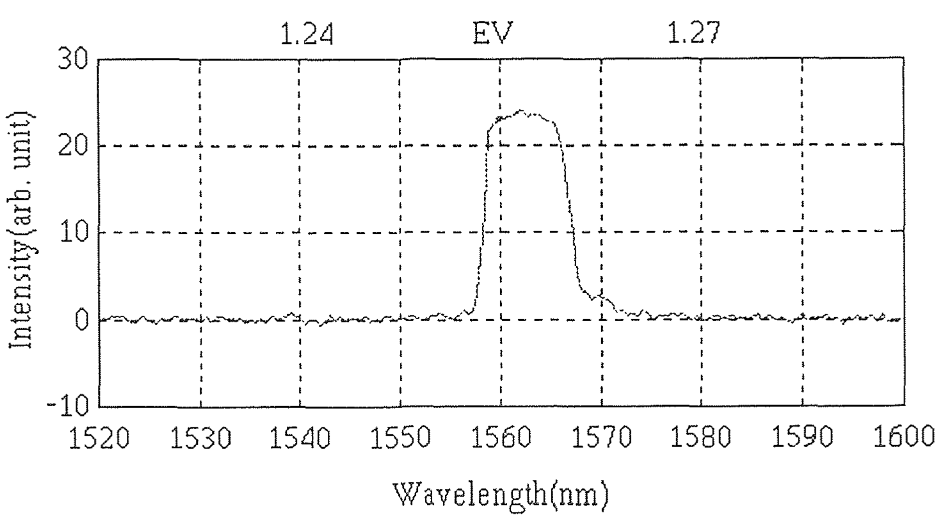


Figure F.2
Blank Test Spectrum for Ar Ion Laser with
Normalized PbS Detector Spectrum Applied

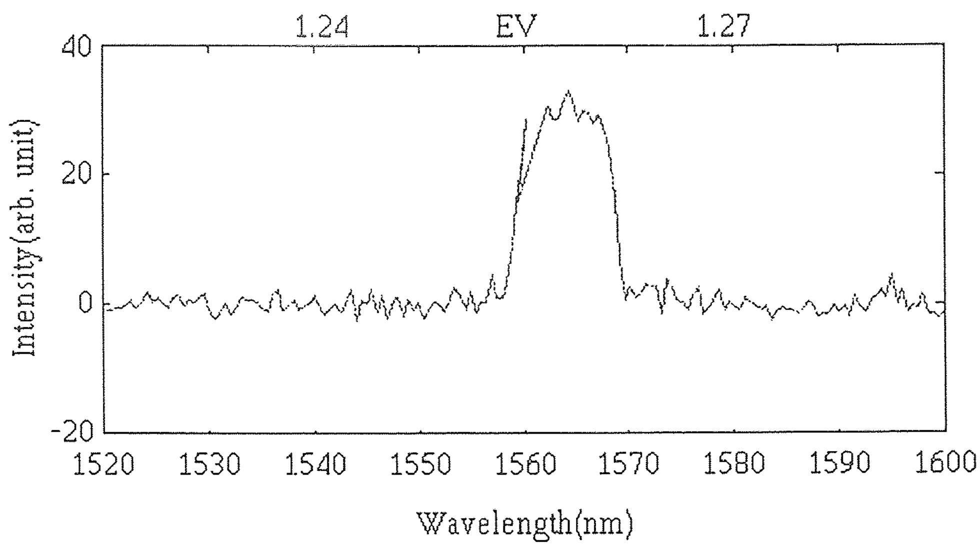


Figure F.3
Spectrum for GaSb with Normalized
PbS Detector Spectrum Applied
Light Source : Ar Ion Laser

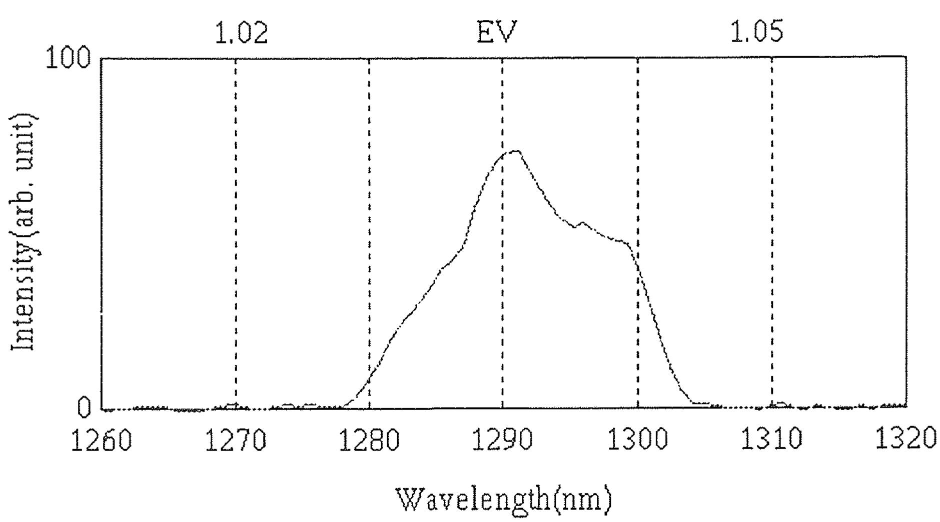


Figure F.4
Blank Test Spectrum for He-Ne Laser with
Normalized PbS Detector Spectrum Applied

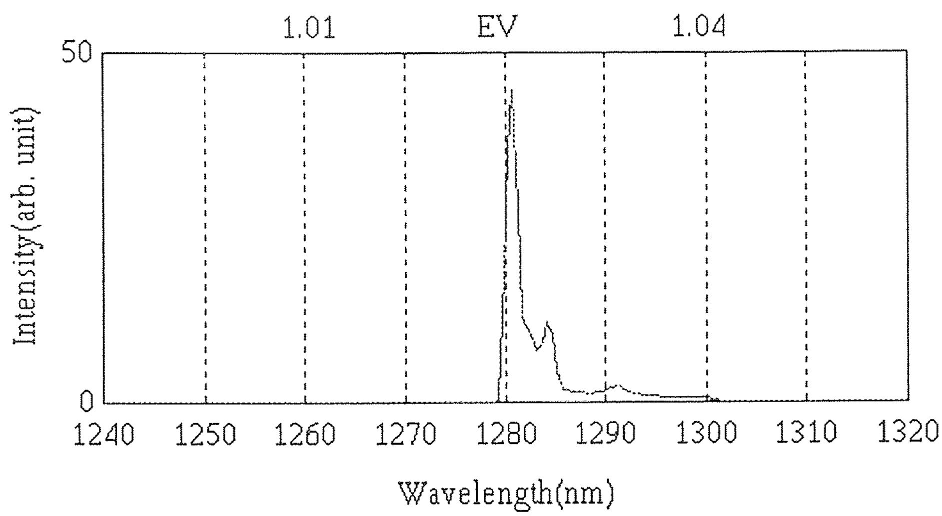


Figure F.5
Spectrum for GaSb with Normalized
PbS Detector Applied
Light Source : He-Ne Laser

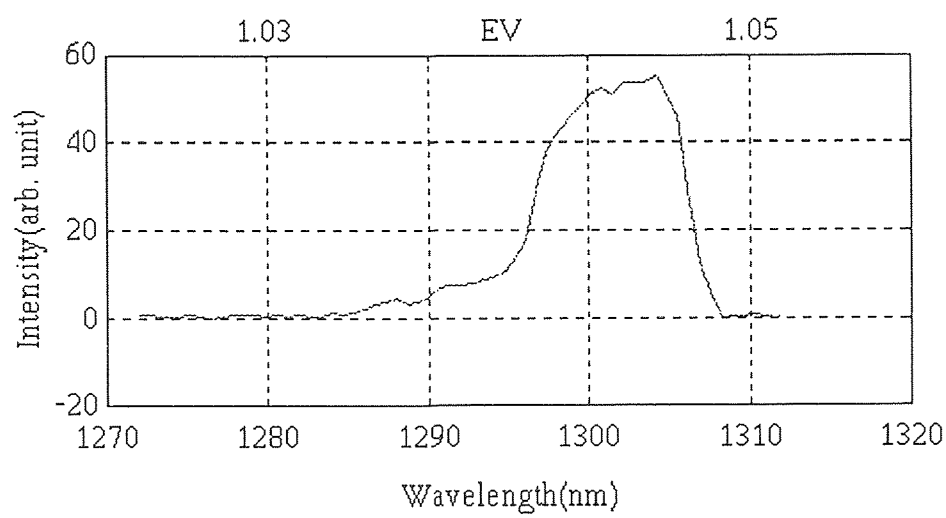


Figure F.6
Spectrum for InGaAs with Normalized
PbS Detector Spectrum Applied
Light Source : He-Ne Laser

APPENDIX G

DATA ACQUISITION PROGRAM OF PL EXPERIMENT

```
1   CLEAR ,60000! : IBINIT1=60000! : IBINIT2=IBINIT1+3 : IBINIT3=IBINIT1+6 : IBINIT4=IBINIT1+9 : BLOAD "bib.m", IBINIT1
2   CALL IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,IBLOC,IBPPC,IBBNA,IBONC,IBRSC,IBSRE,IBRSV,IBPAD,IBSAD,IBIST,IBNMA,IBEOS,IBTMO,IBEOT,IBRDF,IBWRTF,IBTRAP,IBDEV,IBLN,IBLINES,IBCONFIG)
3   CALL IBINIT2(IBGTS, IBCAC, IBWAIT, IBPOKE, IBWRT, IBWRTA, IBCMD, IBCMDA, IBRD, IBRDA, IBSTOP, IBRPP, IBRSP, IBDIAG, IBXTRC, IBRDI, IBWRTI, IBRDIA, IBWRTIA, IBEVENT, IBSTA%, IBERR%, IBCNT%)
4   CALL IBINIT3(SENDCMDS, SENDSETUP, SENDDATABYTES, SEND, SENDLIST, RECEIVESETUP, RCVRESPMSG, RECEIVE, SENDIFC, DEVCLEAR, DEVCLEARLIST, ENABLELOCAL, ENABLEREMOTE, SETRWLS, SENDLLO)
5   CALL IBINIT4(PASSCONTROL, READSTATUSBYTE, TRIGGER, TRIGGERLIST, PPOLLCONFIG, PPOLLUNCONFIG, PPOLL, RESETSYS, FINDRQS, ALLSPOLL, FINDLSTN, TESTSYS, IBRDKEY, IBWRTKEY, WAITSRQ, TESTSRQ)
6   REM
7   REM GPIB COMMAND
8   REM
9       UNL% = &H3F : UNT% = &H5F : GTL% = &H1 : SDC% = &H4 : PPC% = &H5 : GGET% = &H8 : TCT% = &H9
10      LLO% = &H11 : DCL% = &H14 : PPU% = &H15 : SPE% = &H18 : SPD% = &H19 : PPE% = &H60 : PPD% = &H70
11  REM
12  REM GPIB STATUS BIT VECTOR
13  REM
14      EERR = &H8000 : TIMO = &H4000 : EEND = &H2000 : SRQI = &H1000 : RQS = &H800 : SPOLL = &H400 : EEVENT = &H200
15      CMPL = &H100 : LOK = &H80 : RREM = &H40 : CIC = &H20 : AATN = &H10 : TACS = &H8 : LACS = &H4 : DTAS = &H2 : DCAS = &H1
16  REM
17  REM ERROR MESSAGE RETURNED IN GLOBAL VARIABLE IBERR
18  REM
19      EDVR = 0 : ECIC = 1 : ENOL = 2 : EADR = 3 : EARG = 4 : ESAC = 5 : EABO = 6 : ENEB = 7
20      EOIP = 10 : ECAP = 11 : EFSO = 12 : EBUS = 14 : ESTB = 15 : ESRQ = 16 : ETAB = 20
21  REM
22  REM EOS MODE BITS
23  REM
24      BIN% = &H1000 : XEOS% = &H800 : REOS% = &H400
25  REM
26  REM TIMEOUT VALUES AND MEANINGS
27  REM
28      TNONE% = 0 : T10US% = 1 : T30US% = 2 : T100US% = 3 : T300US% = 4 : T1MS% = 5 : T3MS% = 6 : T10MS% = 7
29      T30MS% = 8 : T100MS% = 9 : T300MS% = 10 : T1S% = 11 : T3S% = 12 : T10S% = 13 : T30S% = 14 : T100S% = 15
30      T300S% = 16 : T1000S% = 17
```

```
31  REM
32  REM MISCELLANEOUS
33  REM
34        S% = &H8 : LF% = &HA
35  REM
36  REM IBLN CONSTANTS
37  REM
38        ALL.SAD% = -1 : NO.SAD% = 0
39  REM
40  REM IBEVENT CONSTANTS
41  REM
42        EVENTDTAS% = 1 : EVENTDCAS% =2
43  REM
44  REM "OPTION" SELECTION CODES FOR IBCONFIG FUNCTION
45  REM
46        IBCPAD% = &H1 : IBCSAD% = &H2 : IBCTMO% = &H3 : IBCEOT% = &H4 :
IBCPPC% = &H5 : IBCREADDR% = &H6
47        IBCAUTOPOLL% = &H7 : IBCCICPROT% = &H8 : IBCIRQ% = &H9 : IBCSC& = &HA
: IBCSRE% = &HB : IBCEOSRD% = &HC
48        IBCEOSWRT% &HD : IBCEOSCMP% = &HE : IBCEOSCHAR% = &HF : IBCPP2% =
&H10 : IBCTIMING% = &H11 : IBCDMA% = &H12
49        IBCREADADJUST% = &H13 : IBCWRITEADJUST% = &H14 : IBCEVENTQUEUE% =
&H15 : IBCSPOLLBIT% = &H16 : IBCSENDLLO% = &H17
50        IBCSPOLLTIME% = &H18 : IBCPPOLLTIME% = &H19 : IBCNOENDBITONEOS% =
&H1A
51  REM
52  REM These values are used by the Send 488.2 command:
53  REM
54        NULLEND% = &HO : NLEND% = &H1 : DABEND% = &H2
55  REM
56  REM This value is used by the 488.2 Receive command:
57  REM
58        STOPEND% = &H100
59  REM
60  REM The following values are used by the IBLINES function:
61  REM
62        VALIDEOI = &H80 : VALIDATN = &H40 : VALIDSRQ = &H20 : VALIDREN = &H10 :
VALIDIFC = &H8 : VALIDNRFD = &H4 : VALIDDAV = &H1
63        BUSEOI = &H8000 : BUSATN = &H4000 : BUSSRQ = &H2000 : BUSREN = &H1000 :
BUSIFC = &H800 : BUSNRFD = &H400 : BUSNDAC = &H200 : BUSDAV = &H100
64  REM
65  REM THIS VALUE IS USED TO TERMINATE AN ADDRESS LIST. IT
66  REM SHOULD BE ASSIGNED TO THE LAST ENTRY
67  REM
68        NOADDR% = &HFFFF
70 REM ************************************
71 KEY OFF
72 REM ************************************
100 REM ------------------------------------------------------------
200  REM
300  REM National Instruments IEEE-488 card and driver were loaded.
350  REM The driver device address [1..31] are referenced as
400  REM "DEV1".."DEV31". see the National Instruments Software Ref. Manual
```

```
450  REM for further details.
500  REM
550  REM Assign IEEE-488 address for Photomutiplier to 1, and Lock-in
600  Amplifier to 4.
630  REM
650  REM MUST merge this code with DECL.BAS and BIB.M MUST be in this
700  REM directory. These files are supplied by National Instruments.
750  REM-----------------------------------------------------------
800  REM-----------------------------------------------------------
850        PRINT "Begin IEEE-488 Communications Setup"
900  REM-----------------------------------------------------------
950  REM
1000        BOARD$ = "GPIBO"
1050        DEVICE$ = "DEV"
1100        RBUF$ = SPACE$(132) : PBUF$ = SPACE$(132)
1150        OBUF$ = SPACE$(132)
1200        JUNK$ = SPACE$(132)
1250        ICH$ = SPACE$(1)
1300        ACK$ = SPACE$(1)
1350        CR$ = CHR$(13)
1360        MINHARDWAREFREQ# = 1#
1400  REM..
1450        INPUT "Device Address [1..31]: ",DA$
1500        DA = VAL(DA$)
1550        IF  (DA > 0)  AND  (DA < 32)  GOTO 1650
1600        SOUND 350,3 : GOTO 1400
1650  REM..
1700        DEVICE$ = DEVICE$ + DA$
1750  REM
1800        CALL IBFIND (DEVICE$, BD%)
1850        MSG$ = "IBFIND ERROR on " + BOARD$
1900        IF (DV% < 0) THEN GOSUB 7250 : STOP
1950  REM
2000        CALL IBFIND (DEVICE$, DV%)
2050        MSG$ = "IBFIND ERROR on " + DEVICE$
2100        IF  (DV% < 0) THEN GOSUB 7250  :  STOP
2150        EOSV% = &HD
2200        V% = EOSV% + &H1400
2250        CALL IBEOS  ( DV%, V%)
2300        CALL IBTMO ( DV%, T1S%)
2350  REM
2400  REM     Flush anything in input buffer
2450        CALL IBRD(DV%,JUNK$)  :  SOUND 2000,1 : PRINT JUNK$
2500        REM        May time out
2550  REM
2600  REM   If auto bauded, these commands will put us in
2650  REM    a know state without restarting the instrument.
2700  REM
2750        ICH$ = CHR$(222) : CALL IBWRT(DV%,ICH$)
2800        DELAY =  .5 : GOSUB 6950          'Wait for possible reset
2850  REM     Flush anything in input buffer
2900        CALL IBRD (DV%,JUNK$) :  SOUND 2000,1 : PRINT JUNK$
2950        REM        May time out
```

```
3000  REM
3050  REM ..
3100        IICH$ = " ": CALL IBWRT(DV%, ICH$): SOUND 400,2
3150        MSG$ = "IBWRT ERROR"
3200        IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
3250  REM
3300        CALL IBRD (IV%, ICH$): SOUND 1000,2
3350        MSG$ = "IBWRT ERROR"
3400        IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
3450  REM
3500        PRINT "Received". ICH$
3550  REM
3600        IF (ICH$ = "B") GOTO 3850           'Init Program O.K.
3650        IF (ICH$ = "F") GOTO 4450           'Main Program O.K.
3700        GOTO 2350                           'Bad Response. Try Again
3750  REM ..
3800  REM
3850  REM..  Jump to Main Program
3900        PRINT "Jump tto Main Program"
3950        OBUF$ = "02000" + CHR$(0)
4000        CALL IBWRT (DV%, OBUF$)  :  MSG$ = "IBWRT ERROR"
4050        IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
4100  REM
4150  REM
4200        DELAY =  .5 : GOSUB 6950            'Wait for main to start up
4250        CALL IBRD(DV%, ICH$) : MSG$ ="IBRD ERROR"
4300        IF (IBSTA% AND EERR) THEN GOSUB 7250 : RESUME
4350  REM May time out
4400        IF (ICH$ <>."*") GOTO 4150          'Wait for Main Program Response
4450  REM
4500        SOUND 500, 5: SOUND 1000, SOUND 500,5
4550        PRINT "IEEE-488 Communications Established" : print""
4600        PRINT "Getting EPROM and FLASH Version Numbers"
4650        SAYOK = 0          'Turn OFF ACK sound and OK print out flag
4700  REM   ' Put up version number of BOOT and Flash
4750  REM ----Boot Version
4800        ICH$ = "Y" : CALL IBWRT(DV%, ICH$)  'No comma or <CR> after command
4850        MSG$ = "IBWRT ERROR"
4900        IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
4950        GOSUB 6200       'Wait For Confirmation'
5000  REM
5050        GOSUB 14250      'Get RBUF$ from 488
5100  REM
5150        PBUF$ = LEFT$(RBUF$, INSTR(RBUF$,CR$) -1)
5200        PRINT "Boot Version:", PBUF$
5250  REM----Flash Version
5300        ICH$ = "z" : CALL IBWRT(DV%,ICH$) : MSG$ = "IBWRT ERROR"
5350        IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
5400  REM
5450        GOSUB 6200    'Wait For Confirmation'
5500        GOSUB 14250      'Get RBUF$ from 488
55550  REM
5600        PBUF$ = LEFT$(RBUF$, INSTR(RBUF$,CR$) -1)
```

```
5650        PRINT "Flash Version:", PBUF$
5700    REM
5750    REM---- Send Mono Init Command
5755        PRINT "Init Mono, Please Wait..."
5760        CALL IBTMO(DV%, T100S%)    'Allow for long init of 220M, 270M
5800        ICH$ ="A" : CALL IBWRT(DV%, ICH$) : MSG$ = "IBWRT IRROR"
5850        IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
5900        GOSUB 6    200     'Wait For Confirmation'
5910        PRINT " Mono Init Done"
5950        PRINT "" : PRINT "" : PRINT "Hit any key:"
6000        WHILE (INKEY$ ="") : WEND
6050        CALL IBTMO(DV%, T10S%)   'reset time out to something reasonable
6100    GOTO 16600  '***Jump To User Interface
6150    REM____________________________________________________________
6200    REM *****Input ACK **** and test for valid
6250        CALL IBRD(DV%,ACK$)  :  IF (SAYOK =1) THEN SOUND 1000,2
6300        MSG$ = "IBRD ERROR"
6350        IF (IBSTA% AND EERR ) THEN GOSUB 7250 : STOP
6400        IF (ACK$ = "0") THEN 6450 ELSE 6600
6450        REM ..
6500        IF (SAYOK = 1) THEN PRINT "Receive was O>K>"
6550        GOTO 6800
6600    REM ..
6650    IF (ACK$ = "b") THEN PRINT "Receive was bad"
6700    PRINT "Character Received was: ";ASC(ACK$),ACK$
6750    STOP
6800    REM ..
6850    RETURN
6900    REM ____________________________________________________________
6950    REM *****  Time Delay of Delay
7000     T0 = TIMER
7050     WHILE ((TIMER - T0) < DELAY)
7100     WEND
7150      RETURN
7200    REM ____________________________________________________________
7250    REM *****  IEEE-488 ERROR
7300    REM THIS ROUTINE WILL NOTIFY YOU THAT AB IB CALL FAILED
7350    REM AND PRINT THE STATUS VARIABLES.
7400    REM ____________________________________________________________
7450    REM
7500    PRINT MSG$
7550    REM
7600    PRINT "ibsta=$H"; HEX$(IBSTA%); "<";
7650    IF IBSTA% AND EERR THEN PRINT " ERR";
7700    IF IBSTA% AND TIMO THEN PRINT " TIMO";
7750    IF IBSTA% AND EEND THEN PRINT " END";
7800    IF IBSTA% AND SRQI THEN PRINT " SRQI";
7850    IF IBSTA% AND RQS THEN PRINT " RQS";
7900    IF IBSTA% AND CMPL THEN PRINT " CMPL";
7950    IF IBSTA% AND LOK THEN PRINT " LOK";
8000    IF IBSTA% AND RREM THEN PRINT " REM";
8050    IF IBSTA% AND CIC THEN PRINT " CIC";
8100    IF IBSTA% AND AATN THEN PRINT " ATN";
```

```
8150  IF IBSTA% AND TACS THEN PRINT " TACS";
8200  IF IBSTA% AND LACS THEN PRINT " LACS";
8250  IF IBSTA% AND DTAS THEN PRINT "DTAS";
8300  IF IBSTA% AND DCAS THEN PRINT " DCAS";
8350  PRINT ">"
8400 REM
8450 PRINT "iberr= "; IBERR%;
8500 IF IBERR% = EDVR THEN PRINT " EDVR <DOS ERROR>"
8550 IF IBERR% = ECIC THEN PRINT " ECIC <NOT CIC>"
8600 IF IBERR% = ENOL THEN PRINT " ENOL <NO LISTENER>"
8650 IF IBERR% = EADR THEN PRINT " EADR <ADDRESS ERROR>"
8700 IF IBERR% = EARG THEN PRINT " EARG <INVALID ARGUMENT>"
8750 IF IBERR% = ESAC THEN PRINT " ESAC <NOT SYS Ctrlr>"
8800 IF IBERR% = EABO THEN PRINT " EABO <OP. ABORTED>"
8850 IF IBERR% = ENEB THEN PRINT " ENEB <NO GPIB BOARD>"
8900 IF IBERR% = EOIP THEN PRINT " EOIP < Asnc I/O in prg>"
8950 IF IBERR% = ECAP THEN PRINT " ECAP <NO CAPABILITY>"
9000 IF IBERR% = EFSO THEN PRINT " EFSO <FILE SYS. ERROR>"
9050 IF IBERR% = EBUS THEN PRINT " EBUS <COMMAND ERROR>"
9100 IF IBERR% = ESTB THEN PRINT " ESTB <STATUS BYTE LOST>"
9150 IF IBERR% = ESRQ THEN PRINT " ESRQ < SRQ STUCK ON>"
9200 IF IBERR% = ETAB THEN PRINT " ETAB <TABLE OVERFLOW>"
9250 REM
9300 PRINT "ibcnt = ";IBCNT%
9350 REM
9400 REM CALL THE IBCONL FUNCTION TO DISABLE THE HARDWARE AND
9450 REM SOFTWARE UNLESS ERROR WAS A TIME OUT.
9500 IF (IBSTA% AND TIMO) THEN RETURN
9550 REM
9600 V% = 0 : CALL IBCONL(BD%, V%)
9650 RETURN
9700 REM ______________________________________________________
9750 REM * SHORT MOVE TO TARGET AND DISPLAY POSITION TILL DONE
9800 GOSUB 11650    'MOVE RELATIVE
9850 WHILE (MOTORBUSY = 1)
9900 GOSUB 17100    'GET AND DISPLAY POSITION
9950 GOSUB 12250    'TEST MOTOR BUSY
10000 WEND
10050 GOSUB 17100    'GET AND DISPLAY POSITION
10100 RETURN
10150 REM ______________________________________________________
10200 REM *****  BACKLASH MOVE TO TARGET AND DISPLAY POSITION TILL DONE
*****
10250 IF (TARGET# > STEPS#) THEN GOTO 10450
10300 TMP# = TARGET# : TARGET# = TMP# - BACKLASH# : GOSUB 10600
10350 IF (HALTSCAN = 1) THEN RETURN
10400 TARGET# =TMP#
10450 REM ...
10500 GOSUB 10600
10550 RETURN
10600 REM ______________________________________________________
10650 REM ***  MOVE TO TARGET AND DISPLAY POSITION TILL DONE  ***
10700 GOSUB 11650    'MOVE RELATIVE
```

```
10750 HALTSCAN = 0
10800 WHILE (MOTORBUSY = 1)
10850 GOSUB 26050   'GET KEY
10900 IF (KEYVALUE = 0) THEN GOTO 11350   'NO KEY
10950 IF (KEYVALUE = F6KV) THEN GOSUB 30650 : GOTO 11350   'SHUTTER
11000 IF (KEYVALUE = F8KV) THEN 11050 ELSE 11250   'HALT
11050 REM ...   'STOP SCAN
11100 GOSUB 12000   'STOP MOTOR
11150 HALTSCAN = 1   'FLAG TO OTHERS THAN SCAN WAS HALTED
11200 GOTO 11350
11250 REM ... INVALID KEY DURING SCAN
11300 SOUND 500,1
11350 REM ...
11400 GOSUB 17100   'GET AND DISPLAY POSITION
11450 GOSUB 12250   'TEST MOTOR BUSY
11500 WEND
11550 GOSUB 17100   'GET AND DISPLAY POSITION
11600 RETURN
11650 REM ____________________________________________________________
11700 REM *****  MOVE MONO RELATIVE IN STEPS  *****
11750 RELSTEPS# = TARGET# - STEPS#
11800 OBUF$ = "F" + "0" + "," + STR$(RELSTEPS#) + CR$
11850 GOSUB 13950   'PUT OBUF$ TO 488 AND WAIT FOR CONFIRM
11900 MOTORBUSY = 1
11950 RETURN
12000 REM ____________________________________________________________
12050 REM *****  MOTOR STOP  *****
12100 ICH$ = "L"
12150 GOSUB 14500   'PUT ICH$ TO 488 AND WAIT FOR CONFIRM
12200 RETURN
12250 REM ____________________________________________________________
12300 REM *****  MOTOR BUSY  *****
12350 ICH$ = "E"
12400 GOSUB 14500   'PUT ICH$ TO 488 AND WAIT FOR CONFIRM
12450 REM
12500 CALL IBRD(DV%,ACK$)
12550 MSG$ = "IBRD ERROR"
12600 IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
12650 IF (ACK$ = "q") THEN 12700 ELSE 12750
12700 MOTORBUSY = 1 : GOTO 12800
12750 MOTORBUSY = 0 : GOTO 12800
12800 REM ...
12850 RETURN
12900 REM ____________________________________________________________
12950 REM *****  SET MONO POSITION  *****
13000 STEPS# = WAVE# * STEPSPERUNIT#
13050 OBUF$ = "G0," + STR$(STEPS#) + CR#
13100 GOSUB 13950   'PUT OBUF$ TO 488 AND WAIT FOR CONFIRM
13150 RETURN
13200 REM ____________________________________________________________
13250 REM *****  READ MONO POSITION  *****
13300 OBUF$ = "H0" + CR$
13350 GOSUB 13950   'PUT OBUF$ TO 488 AND WAIT FOR CONFIRM
```

```
13400 REM
13450 GOSUB 14250    'GET RBUF$ FROM 488
13500 PBUF$ = LEFT$(RBUF$, INSTR(RBUF$, CR$) -1)
13550 STEPS# = VAL(PUBF$) : WAVE# = EPS# / STEPSPERUNIT#
13600 RETURN
13650 REM____________________________________________________
13700 REM ***** Set Motor Speed
13750 FREQMIN$ = STR$(F1#) : FREQMAX$ = STR$(F2#) ": RAMPTIME$ = STR$(RT#)
13800 OBUF$ "B"+"0,"+ FREQMIN$ +"," + FREQMAX$ +"," + RAMPTIME$ + CR$
13850 GOSUB 13950    'Put OBUF$ to 488 and wait for confirm
13900 RETRUN
13950 REM____________________________________________________
14000 REM ***** Put OBUF$) : MSG$ = "IBWRT ERROR"
14050 CALL IBWRT(DV%, OBUF$) : MSG$ = "IBWRT ERROR"
14100 IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
14150 GOSUB 6200     ' Wait For Confirmation'
14200 RETURN
14250 REM____________________________________________________
14300 REM ***** Get rbuf$ from 488 *****
14350 CALL IBRD(DV%,RBUF$) : MSG$ = "IBRD ERROR"
14400 IF (IBSTA% AND EERR ) THEN GOSUB 7250 : STOP
14450 RETURN
14500 REM____________________________________________________
14550 REM ***** Put ICH$ to 488 ***** and wait for confirm
14600 CALL IBWRT (DV%, ICH$) : MSG$ = "IBWRT ERROR"
14650 IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
14700 GOSUB 6200     'Wait For Confirmation'
14750 RETURN
14800 REM____________________________________________________
14850 REM ***** Accessory Busy *****
14900 ICH$ = "1" : Gosub 14500 'Put ICH$ to 4988 and confirm
14950 REM
15000 CALL IBRD(DV%, ACK$) : MSG$ = "IBRD ERROR"
15050 IF (IBSTA% AND EERR) THEN GOSUB 7250 : STOP
15100 IF (ACK$ + "q") THEN ACCESSORYBUSY = 1 ELSE ACCESSORYBUSY =0
15150 RETURN
15200 REM____________________________________________________
15250 REM____________________________________________________
15300 REM *** SHUTTER _OPEN else SHUTTER_ CLOSE ***
15350 IF (SHUTTER + 1) THEN OBUF$ = "WO"+ CR$ ELSE OBUF$ = "XO" +CR$
15400 GOSUB 13950    'Put OBUF$ to 488 and wait for confirm
15450 GOSUB 19050    'Display Shutter Position
15452 ACCESSORYBUSY = 1
15500 RETURN
15550 REM____________________________________________________
15600 REM *** TURRET_POSITION_1 else TURRET _POSITION_0 ***
15650 IF (TURRET+1) THEN OBUF$ = "a0" + CR$ ELSE OBUF$ +"b0" +CR$
15700 GOSUB 13950    ' Put OBUF$ to 488 and wait for confirm
15750 GOSUB 18600    'Display Accessories
15752 ACCCESSORYBUSY = 1
15800 RETURN
15850 REM____________________________________________________
15900 REM *** ENTR_MIRROR_SIDE_POSITION else ENTR_MIRROR_FRONT_POSITION ***
```

```
15950 IF (ENTMIRROR +1) THEN OBUF$ ="C0" + CR$ELSE OBUF$ = "d0" + CR$
16000 GOSUB 13950   'Put OBUF$ to 488 and wait for confirm
16050 GOSUB 18600     'Display Accessories
16052 ACCESSORYBUSY = 1
16100 RETURN
16150 REM_______________________________________________________________
16200 REM *** EXIT_MIRROR_SIDE_POSITION else EXIT_MIRROR_FRONT_POSITION ***
16250 IF (EXTMIRROR = 1) THEN OBUF$ = "e0" +CR$ ELSE OBUF$ ="f0" + CR$
16300 GOSUB 13950    'Put OBUF$ to 488 and wait for confirm
16350 GOSUB 18600     'Display Accessories
16352 ACCESSORYBUSY = 1
16400 RETURN
16450 REM_______________________________________________________________
16500 REM
16550 REM-----------------------------------------------------------------------------
16600   ON ERROR GOTO 16900
16650 REM-----------------------------------------------------------------------------
16700 REM ...
16750  GOSUB 36350      'Input Last Set of User Parameters
16800  GOTO 19350      'Jump To Main
16850 REM_______________________________________________________________
16900 REM .....ERROR HANDLER
16950  PRINT "***** ERROR=", ERR, " on line number", ERL
17000  LASTERROR = ERR
17050  IF (LASTERROR = 62) THEN RESUME NEXT ELSE STOP
17100 REM_______________________________________________________________
17150 REM ***** GET AND DISPLAY POSITION *****
17200  GOSUB 13200     'READ MONO POSITION
17210  GOSUB 12860      'Check limits
17250 LOCATE 1,40
17300 PRINT USING ": #####.###   ########   ##"; WAVE#, STEPS#, LIMIT
17310 IF ((REPEAT >0) AND (FLAG =2)) THEN GOTO 38000   'JUMP TO EG&G
17320 REM IF ((MESSAGE$ = "F7") AND (FLAG = 2)) THEN GOTO 38000  'JUMP TO EG&G
17340 REM
17350 RETURN
17400 REM_______________________________________________________________
17450 REM ***** Display Scan *****
17500 LOCATE 3,40
17550 PRINT USING ": #####.###"; STARTSCAN#
17600 LOCATE 4,40
17650 PRINT USING ":#####.###"; ENDSCAN#
17700 LOCATE 5, 40
17750 PRINT USING ":#####.#          ######"; SCANSPEED#, SCANFREQ#
17800 RETURN
17850 REM_______________________________________________________________
17900 REM ***** Display Tweek Speed *****
17950 LOCATE 13, 40
18000 IF (TWEEKSPEED<>1) THEN PRINT ": Fast Increment"
18050 IF (TWEEKSPEED= 1) THEN PRINT": Slow Increment"
18100 RETURN
18150 REM_______________________________________________________________
18200 REM ***** Markers *****
18250 LOCATE 16, 10
```

```
18300 PRINT USING ": #####.###"; MARKERA#
18350 LOCATE 17, 10
18400 PRINT USING ": #####.###"; MARKERB#
18450 LOCATE 18, 10
18500 PRINT USING ": #####.###"; MARKERC#
18550 RETRUN
18600 REM______________________________________________________________
18650 REM ***** Display Accessories *****
18700 LOCATE 8, 40
18750 PRINT ": ", TURRET
18800 LOCATE 9, 40
18850 PRINT ": ", ENTMIRROR
18900 LOCATE 10, 40
18950 PRINT ": ", EXTMIRROR
19000 RETURN
19050 REM______________________________________________________________
19100 REM ***** Display Shutter *****
19150 LOCATE 7, 40
19200 IF (SHUTTER = 0) THEN PRINT ": COLOSED" ELSE PRINT": OPEN "
19250 RETURN
19300 REM______________________________________________________________
19350 REM______________________________________________________________
19400 REM ***** Main ****
19450  F1KV = 187
19500  F2KV = 188
19550  F3KV = 189
19600  F4KV = 190
19650  F5KV = 191
19700  F6KV = 192
19750  F7KV = 193
19800  F8KV = 194
19850  F9KV = 195
19900  F10KV = 196
19950 LEFTKV = 203
20000 RIGHTKV = 205
20020 RKV =70
20030 PKV = 80
20050 TKV = 84
20100 EKV = 69
20150 XKV = 88
20200 CTRLA = 1
20250 CTRLB = 2
20300 CTRLC = 3
20350 REM
20400  ON KEY (1) GOSUB 21050
20450  ON KEY (2) GOSUB 21100
20500  ON KEY (3) GOSUB 21150
20550  ON KEY (4) GOSUB 21200
20600  ON KEY (5) GOSUB 21250
20650  ON KEY (6) GOSUB 21300
20700  ON KEY (7) GOSUB 21350
20750  ON KEY (8) GOSUB 21400
20800  ON KEY (9) GOSUB 21450
```

```
20850  ON KEY (10) GOSUB 21500
20900  ON KEY (11) GOSUB 21550
20950  ON KEY (12) GOSUB 21600
21000  GOTO 21700
21050  HOTKEY = F1KV : RETURN
21100  HOTKEY = F2KV : RETURN
21150  HOTKEY = F3KV : RETURN
21200  HOTKEY = F4KV : RETURN
21250  HOTKEY = F5KV : RETURN
21300  HOTKEY = F6KV : RETURN
21350  HOTKEY = F7KV : RETURN
21400  HOTKEY = F8KV : RETURN
21450  HOTKEY = F9KV : RETURN
21500   HOTKEY = F10KV : RETURN
21550   HOTKEY = LEFTKV : RETURN
21600   HOTKEY = RIGHTKV : RETURN
21650  REM
21700   KEY(1) ON: KEY(2) ON: KEY(3) ON: KEY(4) ON: KEY(5) ON
21750   KEY(6) ON: KEY(7) ON: KEY(8) ON: KEY(9) ON: KEY(10) ON
21800 KET(12) ON: KEY(13) ON
21850  REM
21900   GOTO 22100
21950   CLS
22000   LOCATE 15,20
22050   INPUT "Mono Type : ",MONOWANTED
22100   GOSUB 26700
22150   PRINT "": PRINT "HIT ANY KEY:"
22200   WHILE (INKEY$ = "") : WEND
22250  REM
22300   CLS
22350  REM
22400   SAYOK = 0     'TURN OFF 488 ACK SOUND AND OK PRINT OUT FLAG
22450   MOTORBUSY = 0 : MOVEBUSY = 0 : HALTSCAN = 0
22452   ACCESSORYBUSY = 0 : TUPDATE = TIMER
22500   WAVE# = 0# : TARGET# = 0#
22550   TWEEKINC = 1
22600   TWEEKSPEED = 1 : TWEEKMAX = STEPSPERUNIT# + 1
22650   STEPS# = 0
22700   SHUTTER = 0
22750   TURRET = 0
22800   ENTMIRROR = 0
22850   EXTMIRROR = 0
22900   SCANSPEED# = SCANFREQ# / STEPSERUNIT#
22950  REM
23000   MENUROW = 1 : MENUCOL = 15 : GOSUB 25000  'PULL UP MAIN MENU
23050   ROW = 20 : COL = 20 : MESSAGE$ = "Command: " : GOSUB 28100
23100   ROW = 20 : COL = 20+9
23150  REM
23160   IF FRE(0) < 5000 THEN COLOR 26,0 : LOCATE 1,1 : PRINT USING "#####";FRE(0) :
COLOR 7,0
23170   LOCATE 1,1 : PRINT USING "#####";FRE(0)
23200   GOSUB 30150   'CALIBRATE
23250   GOSUB 17100   'DISPLAY POSITION
```

```
23300   GOSUB 17400   'DISPLAY SCAN
23350   GOSUB 17850   'DISPLAY TWEEK SPEED
23400   GOSUB 18600   'DISPALY ACCESSORIES
23450   GOSUB 15250   'DO AND DISPLAY SHUTTER
23500   GOSUB 18150   'DISPLAY MARKERS
23550   F1# = MINFREQ# : F2# = MAXFREQ# : RT# = RAMPTIME#
23600   GOSUB 13650   'SET MOTOR SPEED
23650  REM
23670  CHECKPT = 0
23700  WHILE (1)
23750   GOSUB 26050   'GET KEY
23800   IF (KEYVALUE = 0) THEN GOTO 24850
23850   IF (KEYVALUE = 19) THEN GOTO 21950   '(Ctrl-s) CHANG MONO TYPE
23900   IF (KEYVALUE = F1KV) THEN GOSUB 28350 : GOTO 24850
23950   IF (KEYVALUE = F2KV) THEN GOSUB 28800 : GOTO 24850
24000   IF (KEYVALUE = F3KV) THEN GOSUB 29200 : GOTO 24850
24050   IF (KEYVALUE = F4KV) THEN GOSUB 29600 : GOTO 24850
24100   IF (KEYVALUE = F5KV) THEN GOSUB 30150 : GOTO 24850
24150   IF (KEYVALUE = F6KV) THEN GOSUB 30650 : GOTO 24850
24200   IF (KEYVALUE = F7KV) THEN GOSUB 30950 : GOTO 24850
24250   IF (KEYVALUE = F8KV) THEN GOSUB 32550 : GOTO 24850
24300   IF (KEYVALUE = F9KV) THEN GOSUB 32850 : GOTO 24850
24350   IF (KEYVALUE = F10KV) THEN GOSUB33150 : GOTO 24850
24400   IF (KEYVALUE = LEFTKV) THEN GOSUB 33400 : GOTO 24850
24450   IF (KEYVALUE = RIGHTKV) THEN GOSUB 34100 : GOTO 24850
24460 IF ( (AKEY$ = "F") OR (AKEY$ = "f") ) THEN GOSUB 43000 : GOTO 24850
24470 IF ( (AKEY$ = "P") OR (AKEY$ = "p") ) THEN GOSUB 57000 : GOTO 24850
24500 IF ( (AKEY$ = "T") OR (AKEY$ = "t") ) THEN GOSUB 34750 : GOTO 24850
24550 IF ( (AKEY$ = "E") OR (AKEY$ = "e") ) THEN GOSUB 35050 : GOTO 24850
24600 IF ((AKEY$ = "X") OR (AKEY$ = "x")) THEN GOSUB 35350 : GOTO 24850
24650 IF ((AKEY$ = "A") OR (AKEY$ = "a")) THEN GOSUB 35650 : GOTO 24850
24700 IF (KEYVALUE = CTRLA) THEN GOSUB 35950 : GOTO 24850
24701 IF ((AKEY$ = "B") OR (AKEY$ = "b")) THEN GOSUB 36310 : GOTO 24850
24702 IF (KEYVALUE = CTRLB) THEN GOSUB 36316 : GOTO 24850
24703 IF ((AKEY$ = "C") OR (AKEY$ = "c")) THEN GOSUB 36330 : GOTO 24850
24704 IF (KEYVALUE = CTRLC) THEN GOSUB 36336 : GOTO 24850
24750 LOCATE 23,1 : PRINT "Key Value: ",KEYVALUE : SOUND 1000,1
24800 IF (KEYVALUE = 26) THEN STOP
24850 REM ...
24852 IF (TIMER - TUPDATE < .2) THEN GOTO 24890
24853   TUPDATE = TIMER
24854     GOSUB 17100   'UPDATE POSITION AND LIMITS
24856     GOSUB 14800   'TEST ACCESSORY BUSY
24858     IF (ACCESSORYBUSY = 1) THEN LOCATE 23,20 : PRINT "Accessory Busy"
24860     IF (ACCESSORYBUSY = 0) THEN LOCATE 23,1 : PRINT SPACE$(70)
24870 IF (CHECKPT = 1) THEN GOTO 22250  'PULL UP MAIN MENU AGAIN
24890 REM ...
24900 WEND
24950 STOP
25000 REM
25050 REM____________________________________________________________
25100 REM ***** Main Menu *****
25150 LOCATE MENUROW+0,MENUCOL: PRINT "<- Arrow -> to Tweek Mono"
```

```
25200 LOCATE MENUROW+1,MENUCOL: PRINT "F1   - Set"
25250 LOCATE MENUROW+2,MENUCOL: PRINT "F2   - Start Position"
25300 LOCATE MENUROW+3,MENUCOL : PRINT "F3   - End Position"
25350 LOCATE MENUROW+4,MENUCOL : PRINT "F4   - Scan Speed"
25400 LOCATE MENUROW+5,MENUCOL : PRINT "F5   - Calibrate"
25450 LOCATE MENUROW+6,MENUCOL : PRINT "F6   - Shutter"
25500 LOCATE MENUROW+7,MENUCOL : PRINT "  T   - Turret"
25550 LOCATE MENUROW+8,MENUCOL : PRINT "  E   - Entrance Mirror"
25600 LOCATE MENUROW+9,MENUCOL : PRINT "  X   -Exit Mirror"
25650 LOCATE MENUROW+10,MENUCOL : PRINT "F7   - Scan"
25700 LOCATE MENUROW+11,MENUCOL : PRINT "F8   - Halt"
25750 LOCATE MENUROW+12,MENUCOL : PRINT "F9   - Tweek Speed"
25800 LOCATE MENUROW+13,MENUCOL : PRINT "F10   - EXIT"
25830 LOCATE MENUROW+14,MENUCOL : PRINT "   P    - PRINT OUT THE RESULT"
25850 LOCATE MENU+15,1 : PRINT "Maker A"
25900 LOCATE MENUROW+16,1 : PRINT "Maker B"
25950 LOCATE MENUROW+17,1 : PRINT "Maker C"
25980 LOCATE MENUROW+18,MENUCOL : PRINT "   F   - FLUSH"
26000 RETURN
26050 REM ______________________________________________________
26100 REM   *****  GetKey  *****
26150 IF (HOTKEY = 0) THEN GOTO 26350
26200 KEYVALUE = HOTKEY : HOTKEY = 0
26250 RETURN
26300 REM
26350 AKEY$ = INKEY$
26400 L = LEN(AKEY$) + 1
26450 ON L GOTO 26500, 26550, 26600
26500 KEYVALUE = 0 : GOTO 26650
26550 KEYVALUE = ASC(AKEY$) : GOTO 26650
26600 KEYVALUE = ASC(MID$(AKEY$, 2)) + 128 : GOTO 26650
26650 RETURN
26700 REM______________________________________________________
26750 REM *****  Read Mono Parameter File  *****
26800 OPEN "488_scan.ini" FOR INPUT AS #2
26850 NRECSKIP = 7 : LASTERROR = 0
26900 REM ...
26950 INPUT #2, MONOTYPE, COMMENT$: PRINT MONOTYPE, COMMENT$
27000 IF (LASTERROR <> 62) THEN 27200
27050     CLOSE #2
27100     PRINT "Parameters for Mono Type ",MONOTYPE, " NOT Found."
27150     GOTO 22000
27200 REM ...
27250 IF (MONOTYPE = MONOWANTED) THEN 27300 ELSE 27700
27300     INPUT #2, STEPSPERUNIT#, COMMENT$: PRINT STEPSERUNIT#, COMMENT$
27301 STEPERUNIT# = 500/25
27350     INPUT #2, MINFREQ#,COMMENT$: PRINT MINFREQ#,COMMENT$
27400     INPUT #2,MAXFREQ#,COMMENT$: PRINT MAXFREQ#,COMMENT$
27450     INPUT #2,RAMPTIME#,COMMENT$: PRINT RAMPTIME#, COMMENT$
27500     INPUT #2,BACKLASH#,COMMENT$: PRINT BACKLASH#, COMMENT$
27550     INPUT #2,JUNK#,COMMENT$ : PRITN SCANFREQ#,COMMENT$
27600     INPUT #2,COMMENT$ : COMMENT$
27650 GOTO 27950
```

```
27700     FOR I = 1 TO NRECSKIP
27750          INPUT #2, COMMENT$
27800     NEXT I
27850     PRINT COMMENT$
27900 GOTO 26900
27950 REM ...
28000 CLOSE #2
28050 RETURN
28100 REM ________________________________________________
28150 REM ***** Print Message *****
28200 LOCATE ROW,COL : PRINT SPACE$(50) : LOCATE ROW,COL
28250 PRINT MESSAGE$
28300 RETURN
28350 REM ________________________________________________
28400 REM ***** F1 *****
28450 MESSAGE$ = "F1" : GOSUB 28100
28500 LOCATE 22,20
28550 INPUT "Move to: ",W#
28600 TARGET# = W# * STEPSPERUNIT#
28650 LOCATE 21,1 : PRINT SPACE$(70)
28700 GOSUB 10150    'MOVE TO TARGET,DISPLAY AND WAIT TILL DONE
28710 LOCATE 21,20 : INPUT :DO YOU WANT TO READ BACK THE INTENSITY (Y/N)?",YN$
28720 IF (YN$ = "Y" OR YN$ = "y") THEN GOTO 38000 ELSE GOTO 28745
28740 REM
28745 LOCATE 21,1 : PRINT SPACE$(70)
28750 RETURN
28800 REM ________________________________________________
28850 REM ***** F2 *****
28900 MESSAGE$ = "F2" : GOSUB 28100
28950 LOCATE 22,20
29000 INPUT "Start of Scan: ",STARTSCAN#
29050 LOCATE 22,1 : PRINT SPACE$(&)
29100 GOSUB 17400    'DISPLAY SCAN
29150 RETURN
29200 REM
29250 REM ***** F3 :*****
29300 MESSAGE$ = "F3" : GOSUB 28100
29350 LOCATE 22,20
29400 INPUT "End of Scan: ", endscan#
29450 LOCATE 22,1 : PRINT SPACE$(70)
29500 GOSUB 17400                    'DisplayScan
29550 RETRUN
29600 REM________________________________________________
29650 REM ***** F4 *****
29700 MESSAGES = "F4" : GOSUB 28100
29750 LOCATE 22,20
29800 INPUT "Scan Speed: ", SCANSPEED#
29850 LOCATE 22,1:PRINT SPACE$(70)
29900 SCANFREQ# = SCANSPEED# * STEPSPERUNIT#
29950 IF (SCANFREQ# >= MINHARDWAREFREQ# ) THEN GOTO 30000
29962 SCANFREQ# = MINHARDWAREFREQ# : SCANSPEED# = SCANFREQ# /
STEPSPERUNIT#
29964 GOSUB 17400     'DisplayScan
```

```
30100 RETURN
30150 REM_______________________________________________________
30200 REM *****   F5   *****
30250 MESSAGE$ = "F5"  :  GOSUB 28100
30300 LOCATE 22,20
30350 INPUT "Where are you: ", WAVE#
30400 LOCATE 22,1 : PRINT SPACE$(70)
30450 REM
30500 GOSUB 12900      'Set Mono Position
30550 GOSUB 17100      'Display Position
30600 RETRUN
30650 REM_______________________________________________________
30700 REM *****   F6   *****
30750 MESSAGE$ = "F6" :  GOSUB 28100
30800 SHUTTER = 1- SHUTTER
30850 GOSUB 15250
30900 RETURN
30950 REM_______________________________________________________
31000 REM *****   F7   *****
31050 MESSAGE$ = "F7" : GOSUB 28100
31055 LOCATE 22,1 : PRINT SPACE$(70)
31060 LOCATE 22,10
31070 INPUT "HOW MANY TIMES YOU WANT TO REPEAT?", REPEAT
31080 ON ERROR GOTO 32470
31090 LOCATE 22,1 : PRINT SPACE$(70)
31095 IF REPEAT = 0 THEN RETURN
31100 IF REPEAT > 0 THEN : LOCATE 22,10 : INPUT "PLEASE ENTER YOUR SCALE
?",SCAL:INPUT "PLEASE ENTER YOUR GRATING ?",GRATE : GOTO 46000 'MUTIPLE SCAN
AND GET SCALE FOR PLOTTING
31110 IF REPEAT < 1 THEN GOTO 32330
31120 GOTO 32500
31140 REM
32150 REM...
32200 HALTSCAN = 0
32250 LOCATE 11,40
32300 PRINT ": STOPPED  "
32320 GOTO 32350
32330 LOCATE 22,20: : PRINT "YOU HAVE ENTERRED INVALID NUMBER !" : BEEP
32340 GOTO 31070
32350 REM  lOAD nORMAL Speed
32400 F1# = MINFREQ# : F2# = MAXFREQ# : RT# = RAMPTIME#
32450 GOSUB 13650      'Set Motor Speed
32460 GOTO 32485
32470 LOCATE 22,20 : PRINT "ERRORS OCCURS!"
32480 RESUME 31070
32485 FLAG =1
32490 GOSUB 55000   'STORE DATA INTO FILE
32495 COUNT# = 0
32497 REPEEAT = 0
32500 RETURN
32550 REM_______________________________________________________
32600 REM *****  F8  *****
32650 MESSAGE$ = "F8" : GOSUB 28100
```

```
32700 HALTSCAN = 1
32730 SENS = 0
32750 WHILE (INKEY$ <> "") : WEND   'Flush input buffer
32800 RETURN
32850 REM____________________________________________________
32900 REM *****  F9  *****
32950 MESSAGE$ = "F9" : GOSUB 28100
33000 TWEEKSPEED = TWEEKMAX - TWEEKSPEED
33050 GOSUB 36950     ' Shut dowm
33350 RETURN
33400 REM____________________________________________________
33450 REM *****  Left Arrow  *****
33500 MESSAGE$ = "Left" : GOSUB 28100
33550 IF ( TWEEKSPEED <> 1) THEN 33650
33600 IF ( (TIMER-T1) > .1) THEN 33650 ELSE 33700
33650 F =  1 : GOTO 33750
33700 F = F + 3 : IF (F > 5*TWEEKMAX ) THEN F = 5*TWEEKMAX
33750 TARGET# = (STEPS# - TWEEKINC * TWEEKSPEED * F)
33800 IF (TARGET# < 0#) THEN TARGET# = 0# : SOUND 1000,1
33850 REM
33900 GOSUB 9700     ' Short Move to target, display and wait till done
33950 T1 = TIMER
34050 RETURN
34100 REM____________________________________________________
34150 REM *****  Right Arrow  *****
34200 MESSAGE$ = " Right " : GOSUB 28100
34250 IF (TWEEKSPEED <> 1 ) THEN 34350
34300 IF (   (TIMER-T2)   > .1 ) THEN 34350 ELSE 34400
34350 F = 1 : GOTO 34450
34400 F = F + 3 : IF (F> 5*TWEEKMAX)  THEN  F = 5*TWEEKMAX
34450 TARGET# = (STEPS# + TWEEKINC * TWKKESPEED * F)
34500 REM
34550 GOSUB 9700      ' Shout Move to target, display and wait till done
34600 T2 = TIMER
34700 RETURN
34750 REM____________________________________________________
34800 REM *****  "T"  Turret  *****
34850 MESSAGE$ = "T" : GOSUB 28100
34900 TURRET = 1 - TURRET
34950 GOSUB 15550   'DO TURRET AND DISPLAY ACCESSORIS
35000 RETURN
35050 REM ____________________________________________________
35100 REM *****  "E"  ENTmirror  *****
35150 MESSAGE$ = "E" : GOSUB 28100
35200 ENTMIRROR = 1- ENTMIRROR
35250 GOSUB 15850   'DO ENTRANCE MIRROR AND DISPLAY ACCESSORIES
35300 RETURN
35350 REM ____________________________________________________
35400 REM *****  "X"  EXTmirror  *****
35450 MESSAGE$ = "X" : GOSUB 28100
35500 EXTMIRROR = 1 - EXTMIRROR
35550 GOSUB 16150   'DO EXIT MIRROR AND DISPLAY ACCESSORIES
35600 RETURN
```

```
35650 REM ________________________________________________
35700 REM *****  "A"  GOTO MARKER A  *****
35750 MESSAGE$ = "GO TO A" : GOSUB 28100
35800 TARGET# = MARKERA# * STEPSPERUNIT#
35850 GOSUB 10150    'MOVE TO TARGET, DISPLAY AND WAIT TILL DONE
35900 RETURN
35950 REM ________________________________________________
36000 REM *****  "Ctrl-A" SET MARKER A *****
36050 MESSAGE$ = "SET A" : GOSUB 28100
36100 LOCATE 22,20
36150 INPUT "MARKER A: ",MARKERA#
36200 LOCATE 22,1 : PRINT SPACE$(70)
36250 GOSUB 18150    'DISPLAY MARKERS
36300 RETURN
36310 REM ________________________________________________
36311 REM *****  "B" GO TO MARKER B  *****
36312 MESSAGE$ = "GO TO B" : GOSUB 28100
36313 TARGET# = MARKERB# * STEPSPERUNIT#
36314 GOSUB 10150    'MOVE TO TARGET, DISPLAY AND WAIT TILL DONE
36315 RETURN
36316 REM ________________________________________________
36317 REM *****  "Ctrl-B" SET MARKER B  *****
36318 MESSAGE$ = "SET B" : GOSUB 28100
36319 LOCATE 22,20
36320 INPUT "MARKER B: ",MARKERB#
36321 LOCATE 22,1 : PRINT SPACE$(70)
36322 GOSUB 18150    'DISPLAY MARKERS
36323 RETURN
36330 REM ________________________________________________
36331 REM *****  "C" GO TO MARKER C  *****
36332 MESSAGE$ = "GO TO C" : GOSUB 28100
36339 LOCATE 22,20
36340 INPUT "MARKER C: ",MARKERC#
36341 LOCATE 22,1 : PRINT SPACE$(70)
36342 GOSUB 18150    'DISPLAY MARKERS
36343 RETURN
36350 REM ________________________________________________
36400 REM *****  INPUT USER PARAMETERS  *****
36450 OPEN "488_scan.usr" FOR INPUT AS #2
36500 INPUT #2, MONOWANTED, COMMENT$
36550 INPUT #2, STARTSCAN#, COMMENT$
36600 INPUT #2, ENDSCAN#, COMMENT$
36650 INPUT #2, SCANFREQ#, COMMENT$
36700 INPUT #2, MARKERA#, COMMENT$
36750 INPUT #2, MARKERB#, COMMENT$
36800 INPUT #2, MARKERC#, COMMENT$
36850 CLOSE #2
36900 RETURN
36950 REM ________________________________________________
37000 REM *****  EXIT  *****
37050 REM SAVE USER PARAMETERS ON EXIT
37100 OPEN "488_scan.usr" FOR OUTPUT AS #2
37150 PRINT #2, MONOTYPE,      " ;MONO TYPE"
```

```
37200 PRINT #2, STARTSCAN#,  " ;START OF SCAN"
37250 PRINT #2, ENDSCAN#,     ";END OF SCAN"
37300 PRINT #2, SCANFREQ#,    ";SCAN FREQ"
37350 PRINT #2, MARKERA#,     ";MARKER A"
37400 PRINT #2, MARKERB#,     ";MARKER B"
37450 PRINT #2, MARKERC#,     ";MARKER C"
37500 CLOSE #2
37550 GOTO 60000   'TO END THE PROGRAM
37600 GOTO 60000   'TO END THE PROGRAM
38000 REM ____________________________________________________________
38050 REM === SUBROUTINE FOR THE INTENSITY READS BACK --> F1
38055 REM
38057 LOCATE 21,1 : PRINT SPACE$(70)
38060 ON ERROR GOTO 45000
38070 LOCATE 21,20 : PRINT "PLEASE WAIT FOR RESPONSE ..."
38100 GPIBO% = 0 : DEV4% = 4
38150 BDNAME$ = "DEV4"
38200 CALL IBFIND(BDNAME$, DEV4%) 'UNIT DESCRIPTOR RETURNED
38250 BDNAME$ = "GPIB0"
38300 CALL IBFIND(BDNAME$, GPIB0%) 'UNIT DESCRIPTIOR RETURNED
38450 A$ = "Q1" : CALL IBWRT(DEV4%, A%)
38500 FOR I = 1 TO 200 : NEXT I   'WAIT FOR CONFIRM
38550 B$ = SPACE$(30)
38600 CALL IBRD(DEV4%, B$)
38620 LOCATE 19,49 : PRINT "                    "   'CLEAR SPACE FOR DATA
38630 LOCATE 19,30 : PRINT "THE WAVELENGTH IS : ";WAVE#
38650 LOCATE 21,20 : PRINT "THE INTENSITY READS BACK IS : ";B$
38670 FOR I = 1 TO 1000   'DELAY FOR THE INTENSITY DISPLAY
38680 NEXT I
38700 BDNAME$ = "DEV1"
38750 LOCATE 19,30 : PRINT SPACE$(45)   'CLEAR UP THE SPACE
38800 GOTO 28740   'JUMP BACK FROM EG&G TO GPIB
40000 REM
40005 REM ---> INITIALIZING
40010 REM
40050 COUNT# = COUNT# + 1
40100 IF COUNT# > 1 THEN GOTO 40500
40130 IF (REPEAT > 0) THEN GOTO 40200
40150 LOCATE 21,20 : PRINT "PLEASE WAIT >> INITIALIZING !"
40200 ABC# = (ENDSCAN# - STARTSCAN#) \ SCANSPEED# +1
40230 IF (SCANCO = 1) THEN GOTO 40270
40240 ON ERROR GOTO 42000
40250 DIM EGGW#[ABC#], EGGI[ABC#]
40255 ARRDEC = 99   'ARRAY DECLARATION CHECK
40260 SCANCO = 1
40270 ON ERROR GOTO 45000
40300 FOR I = 1 TO ABC#
40350 EGGI#[I] = 0
40400 EGGW#[I] = 0
40450 NEXT I
40500 RETURN
41000 REM
41010 REM ---> STORE DATA ---> FROM LINE #4860
```

```
41020 REM
41030 IF K > 1 THEN COUNT# = COUNT# + 1 : GOSUB 45500
41050 IF K = 1 THEN EGGI#[COUNT#] = VAL(B$)
41150 EGGW#[COUNT#] = WAVE#
41170 RETURN
42000 REM
42050 REM ---> OUT OF MEMORY HANDLING
42100 REM
42150 CLS
42170 COLOR 26,0    'HIGH WHITE BLINKING
42180 LOCATE 1,1 : PRINT USING "#####";FRE(0)
42200 LOCATE 13,5 : PRINT "          ... INSUFFICIENT MEMORY ! ..."
42230 COLOR 7,0    'RETURN TO DEFAULT
42250 LOCATE 15,5 : PRINT "PLEASE USE  F  COMMAND ON MAIN MENU TO FLUSH
MEMORY"
42260 LOCATE 16,5 : PRINT "BETTER YET, RESTART THE PROGRAM AND DO THE SCAN AT
THE VERY FIRST TIME."
42300 LOCATE 21,5 : PRINT "... PLEASE HIT ANY KEY ..."
42350 WHILE(INKEY$ = "") : WEND
42400 GOTO 60000    'ABORT
43000 REM
43050 REM ---> MEMORY MANEGER
43100 REM
43130 MESSAGE$ = "F" : GOSUB 28100
43150 IF (ARRDEC < 99) THEN GOTO 43250
43200 ERASE EGGI#, EGGW# : ARRDEC = 0
43250 LOCATE 1,1 : PRINT USING "#####";FRE(0)
43300 LOCATE 22,20 : PRINT "DONE !"
43320 FOR I = 1 TO 500 : NEXT I    'DELAY FOR DISPLAY
43350 LOCATE 22,1 : PRINT SPACE$(70)
43400 RETURN
45000 REM
45005 REM ---> ON-ERROR HANDLING
45010 REM
45030 NUMBER = NUMBER + 1
45050 LOCATE 21,55 : PRINT "ERRORS !";NUMBER;" TIMES"
45070 IF NUMBER > 50 THEN GOTO 60000
45100 IF ( FRE(0) < 100 ) THEN GOTO 42000 ELSE RESUME
45500 REM
45505 REM ---> DATA AVERAGING
45510 REM
45550 BBB# = VAL(B$)
45600 AVEEGG# = EGGI[COUNT#]
45650 EGGI[COUNT#] = ( BBB# + AVEEGG# ) * .5
45700 RETURN
46000 REM
46005 REM ---> SCANNING ---> FROM F7
46010 REM
46020 GRATING = 1200 / GRATE
46030 CHECKPT = 1    'REPEATED SCAN CHECK
46050 FOR K = 1 TO REPEAT
46100 CLS
46200 HALTSCAN = 0
```

```
46220 FLAG = 1    'POSITION CHECK
46250 TARGET# = STARTSCAN# * STEPSPERUNIT#
46300 GOSUB 10150    'MOVE TO TARGET, DISPLAY AND WAIT TILL DONE
46350 REM
46400 IF (HALTSCAN = 1) THEN GOTO 32150    'USER HALT
46450 REM
46500 DELAY = 1! : GOSUB 6950    'DELAY BEFORE START
46520 FLAG = 2    'SCANNING CHECK
46550 TARGET# = ENDSCAN# * STEPSPERUNIT#
46600 REM
46650 F1# = MINFREQ# : F2# = SCANFREQ# : RT# = RAMPTIME#
46700 IF (SCANFREQ# < MINFREQ#) THEN F1# = SCANFREQ#
46750 GOSUB 13650    'SET MOTOR SPEED
46800 REM
46850 GOSUB 10150    'MOVE TO TARGET, DISPLAY AND WAIT FOR INTENSITY READS
BACK
46900 REM
46950 GOSUB 32550    'STOP SCAN AND RE-ENABLE HOT KEYS
47000 REM
47050 HALTSCAN = 0
47100 REM
47150 REM
47200 F1# = MINFREQ# : F2# = MAXFREQ# : RT# = RAMPTIME#
47250 GOSUB 13650    'SET MOTOR SPEED
47300 COUNT# = 0
47330 PRINT "    ... PLEASE HIT ANY KEY ..."    'BREAK BETWEEN REPEATED SCREEN
47340 WHILE (INKEY$ = "") : WEND
47350 NEXT K
47370 REPEAT = 0
47380 K = 0
47390 FLAG = 1
47400 GOTO 32490    'JUMP BACK TO GPIB
48000 REM
48005 REM ---> SUBROUTINE FOR REPEATED INTENSITY READS BACK ---> F7 (LINE#46850)
48010 REM
48030 IF ( K > 1 ) THEN GOTO 48100
48050 GOSUB 40000    'INITIALIZE
48100 GPIB0% = 0 : DEV4% = 4
48150 BDNAME$ = "DEV4"
48200 CALL IBFIND(BDNAME$, DEV4%)
48250 BDNAME$ = "GPIB0"
48300 CALL IBFIND(BDNAME$, GPIB0%)
48420 A$ = "Q1" : CALL IBWRT(DEV4%,A$)
48440 FOR I = 1 TO 100 : NEXT I
48450 B$ = SPACE$(30)
48500 CALL IBRD(DEV4%, B$)
48550 BDNAME$ = "DEV1"
48600 GOSUB 41000    'STORE DATA
48650 GOSUB 49000    'ON-SCREEN DISPLAYING
48700 GOTO 17340    'JUMP BACK TO GPIB
49000 REM
49005 REM ---> ON-SCREEN DISPLAYING
49010 REM
```

```
49903 PP = 140 + (INT((EGGI#[1] - EGGI#[COUNT#])/SCAL + .5 ))    '140 IS THE GROUND
VALUE
49004 IF ( SENS = 1 ) THEN GOTO 49930
49905 REM ____________________________________________________________
49906 SENS = 1
49907 CLS : KEY OFF : SCREEN2
49908 STARTEV# = 12400 / (STARTSCAN# * GRATING) : ENDEV# = 12400 / (ENDSCAN# *
GRATING)
49909 LOCATE 4,5 : PRINT "EV" : LOCATE 4,26 : PRINT USING " FROM  ###.##    TO
###.##";STARTEV#;ENDEV# : LOCATE 4,61 : PRINT USING "SCALE : ###.#";SCAL
49912 LOCATE 15,1 : PRINT "I"
49915 LOCATE 25,5 : PRINT "A" : LOCATE 25,25 : PRINT USING "FROM  #####.##    TO
#####.##";STARTSCAN# * GRATING; ENDSCAN# * GRATING
49918 LINE(1,15)-(640,15)    'EV SCALE LINE
49921 LINE(1,191)-(640,191)    'WAVELENGTH SCALE LINE
49924 LINE(17,15)-(17,191)    'INTENSITY SCALE LINE
49925 LINE(8,140)-(17,140)    'GROUND LINE
49927 X = 16 : SBP = 2    'SBP STANDS FOR SPACE BETWEEN POINTS
49928 REM ____________________________________________________________
49930 X = X +SBP
49932 IF ( PP > 190 OR PP < 16 ) THEN GOTO 49950
49933 IF ( X > 640 ) THEN LOCATE 25,8 : PRINT "< STILL GOING WITHOUT PT. ON SCREEN
BECAUSE OF MORE THAN";624\SBP;"PT.S >" : GOTO 49950
49936 PSET(X,PP)
49950 RETURN
55000 REM
55001 REM ---> STORE DATA
55002 REM
55050 OPEN "DATA" FOR OUTPUT AS #3
55100 FOR I = 1 TO ABC#
55150 AA# = EGGW#[I] * GRATING
55200 BB# = EGGI#[I]
55250 WRITE #3, AA#, BB#
55300 NEXT I
55350 CLOSE #3
55400 RETURN
57000 REM
57001 REM ---> PRINT OUT DATA
57002 REM
57050 MESSAGE$ ="P" : GOSUB 28100
57055
CLS
57060 PRINT "----->> PLEASE ENTER  1  TO PRINT OUT ON SCREEN,
                                   AND  2  TO THE PRINTER ..."
57065 INPUT STATUS
57070 IF ( STATUS < 1 OR STATUS > 2 ) THEN GOTO 57075 ELSE GOTO 57100
57075 PRINT " <<< INVALID ENTRY !  PLEASE TRY AGAIN ! >>>" : GOTO 57060
57100 PRINT " WAVELENGTH (A)           INTENSITY "
57120 I = 0
57130 OPEN "DATA" FOR INPUT AS #3
57140 IF ( STATUS = 2 ) THEN LPRINT " WAVELENGTH (A)          INTENSITY"
57150 IF EOF(3) THEN GOTO 57380
57170 I = I + 1
```

```
57200 INPUT #3, AA#, BB#
57220 IF ( AA# = 0 ) THEN GOTO 57380
57240 IF ( STATUS = 2 ) THEN GOTO 57320
57250 PRINT AA#,"    >",BB#
57300 IF ( I MOD 18 = 0 ) THEN PRINT "    ... PLEASE HIT ANY KEY ..." : WHILE (INKEY$ = "") :
WEND
57310 GOTO 57350
57320 LPRINT AA#,"    >",BB#
57350 GOTO 57150
57380 PRINT "    ... PLEASE HIT ANY KEY ..." : WHILE (INKEY$ = "") : WEND
57400 CLOSE #3
57450 CHECHPT = 1
57500 RETURN
60000 REM
60050 CALL IBLOC(DV%)    'PUT 488 OFF LINE
60100 SYSTEM    'EXIT BASIC
```

REFERENCES

1. Bartee, Thomas C., BASIC Computer Programing, 2nd Ed., Cambridge[Mass.] : Harper & Row, 1985.

2. BAISC Handbook - General Programing Information, 3rd Ed., Personal Computer Hardware Reference Labrorary, IBM Corp., 1984.

3. Getting Start With Your GPIB-PCII/IIA and the NI-488.2 for MS-DOS, National Instruments Corp., 1992.

4. Hsiu, Ching-Fang, IBM PC Introduction and BASIC, 7th Ed., Taipei : Song-Gan, Inc., 1985.

5. Kittle, Charles, Introduction to Solid State Physics, 6th Ed., John Wiley & Sons, Inc., 1986

6. Verdeyen, Joseph T., Laser Electronics, 2nd Ed., New York : Prentice-Hall, Inc., 1989.

7. Bowen, E. J. and F.R.S., Luminescence In Chemistry, Princeton, N.J. : Van Nostrand, 1968.

8. Model 5207 & 5208 Lock-in Amplifiers Instruction Manual, EG&G Princeton Applied Research.

9. Klein, Miles V., Furtak, Thomas E., Optics, 2nd Ed., New York : John Wiley & Sons, Inc., 1986.

10. Yariv, Amnon, Optical Electronics, 4th Ed., Standard College Publishing, a Division of Holt, Rinehart and Winston, Inc., 1991.

11. Pankove, Jacques I., Optical Processes In Semiconductors, New York : Dover Publications, Inc., 1971.

12. Yariv, Amnon and Yeh, Pochi, Optical Waves In Crystals, New York : John Wiley & Sons, Inc., 1984

13. Parker, C. A., F. R. M. S. and F. R. I. C., Photoluminescence of Solutions, New York : Elsevier Publishing Co, 1968.

14. Sze, S. M., Physics of Semiconductor Devices, 2nd Ed., New York : John Wiley & Sons, Inc., 1981.

15. Hammond III, H. K. and Mason, H. L., Editors, Precision Measurement and Calibration -- Selected NBS Papers on Radiometry & Photometry, U.S. Department of Commerce, NBS Special Publication 300,Volume 7, 1971.

REFERENCES
(Continued)

16. Eisberg, Robert and Resnick, Robert, Quantum Physics, 2nd Ed., John wiley & Sons, Inc., 1985.

17. Sze, S. M., Semiconductor Devices Physics & Technology, New York : McGraw-Hill, Inc.,

18. Streetman, Ben G., Solid State Electronic Devices, 3rd Ed., New York : Prentice Hall, Inc., 1990.

19. Spectrometer Control Via JY232, JY488 And DataLink Interface Manual, JY Optical Systems Instruments., 1993.

20. The Oriel Catalog-Light Sources, Monochromators, Detection System Volume II, Oriel Corporation, 1989.

21. Pankove, J. I., Topics In Applied Physics --Eltroluminescence, Volume 17, New York : Springer-Verlag Berlin Heidelberg, 1977.

22. Sze, S.M., VLSI Technology, 2nd Ed., New York : McGraw-Hill, Inc., 1988.