受理 55.12.22

博士論文

"ELECTRONIC STRUCTURES OF Si-SiO_2 SYSTEM"

「Si-SiO_2 系の電子状態」

東京大学大学院工学系研究科
電子工学専門課程

桜井貴康

1980年 12月 22日

ABSTRACT

Various energy levels caused by micro-structural defects such as dangling bonds of silicon or oxygen stoms, oxygen vacancies and impurity atoms (H, OH, Cl and F) at the interface between amorphous SiO_2 and the crystalline Si substrate, and in the SiO_2 film are calculated based on Green's function formulation and the parameterized tight-binding Hamiltonians for Si and O. The extended Hückel theory in which overlap integrals are not included is used to calculate Hamiltonians for H, Cl and F.

The major results are as follows. (1) The perfect interface and the interface with oxygen dangling bonds do not have a energy level in the Si bandgap, whereas $Si_3 \equiv Si-$ dangling bond and oxygen vacancy at the interface have. (2) This dangling bond level moves out from the Si bandgap with bonding any of H, OH, Cl and F. This corresponds to the annealing behavior of the interface trap states. (3) Si-Si weak bond, Si-Si weak interaction, Si-O weak bond, Si-O weak interaction at the interface give rise to localized states in the Si bandgap whose energy vary by changing the geometrical configuration of the chemical bonding such as the bond length and the bond angles. These are thought to be possible origins of interface states continuously distributed in energy. (4) Amorphous SiO_2 without Si dangling bonds or oxygen vacancies

has no localized level in the SiO_2 bandgap, even if the Si-O-Si bond angle is varied in a wide range. (5) The Si dangling bond and Si-Si bond in SiO_2 cause gap states. These states can be origins of neutral traps in the SiO_2.

TABLE OF CONTENTS

CHAPTER 1. INTRODUCTION

Silicon is widely used for electronic devices and its oxide, especially SiO_2, also takes an important role in the recent semiconductor electronics as protecting layers of silicon ICs and insulators of MOSFETs. So that great amount of activities have been carried out on the Si-SiO_2 system but there still remains some problems about the Si-SiO_2 interface trap states and SiO_2 bulk traps.[1~45]

The trap states at the Si-SiO_2 interface are important in controlling the threshold voltage Vth, transconductance and flicker noise of MOS transistors. Although MOS technology has made possible to reduce the density of the Si-SiO_2 interface states, there typically remains a resudual density of the interface states between 10^{11} and 10^{12} $[cm^{-2}]$.[54]

On the other hand, the SiO_2 bulk traps have much to do with a "hot electron injection problem". Recently, the size of MOS devices in ICs becomes very minute, and as a result, the doping density becomes higher due to the "scaling theory". Consequently, the electric field near the Si-SiO_2 interface increases to the point where the electrons are accelarated by the field and injected into the SiO_2 layer. The injected electrons are then trapped by the SiO_2 bulk traps and shift the threshold voltage. This gives rise to the instability of MOS devices and is considered to be a technological problem. Since the SiO_2 bulk traps are much generated by the electron beam

lithography, the "hot electron injection problem" severely limits the design of VLSI.

However, the microscopic origins of the Si-SiO_2 interface states and SiO_2 neutral traps are not yet understood clearly. The purpose of this research is to obtain the theoretical background of the microscopic origins of these traps. In this thesis the following items are discussed. (1) The calculation method to treat the electronic structure of crystalline Si-amorphous SiO_2 interfaces and bulk amorphous SiO_2 with or without microscopic structural disorders such as defects and impurities, has been developed based on tight-binding Hamiltonians and Green's function formulation. (2) This method has been applied to the real system with varying physical parameters such as bonding lengths and bonding angles in a wide range. (3) According to the results of the calculation, the possible origins of the Si-SiO_2 interface trap states continuously distributed in energy and the SiO_2 bulk traps are proposed. Theoretical explanation of the annealing behaviors of these traps are also discussed.

Chapter 1 of this thesis is dedicated for the determination of the tight-binding parameters for silicon. Chapter 2 describes the Extended Hückel Theory which is used for the impurity problems in this work. Chapter 3 is dealing with the Green's function formulation as a general tool for calculating the electronic structures of solids and some applications of this method are described in this chapter. In chapter 5, the

calculation method for the interface problem and the calculated results are given together with the possible models of the interface trap states and their annealing behavior. The bonding defects in amorphous SiO_2 are discussed in chapter 6.

Chapter 7 is dedicated for general conclusions. Appendix A, B, and C treat the computer programs used in this work, the measurements of SiO_2 bulk traps, and the useful tables and figures, respectively.

CHAPTER 2. SINGLE CRYSTALLINE SILICON SUBSTRATE

SECTION 2-1. INTRODUCTORY REMARK

In this chapter the determination of a new set of tight-binding interaction parameters for silicon is described. In order to calculate the electronic structures of solids, the first thing to do is to set up Hamiltonians. The basis of the Hamiltonian can be plane waves, orthogonalized plane waves, muffin-tin orbitals, or atomic orbitals. Since the interface problems which this work deals with is itself complex, one electron Hamiltonian together with atomic orbitals as a basis set (Linear Combination of Atomic Orbitals or tight-binding method) has been adopted because the tight-binding method is the most simple and flexible, but nevertheless it is known that this method can describe the various electronic properties of solids satisfactorily if the parameters in the tight-binding method are properly chosen. The interaction parameters in the simple tight-binding method, which can be regarded as resonance integrals or hopping integrals, are usually determined so as to fit the calculated band structure to the experimental results and the band structure calculated by the pseudopotential method. As for the interaction parameters for Si several authors have published the parameters, but the agreement of the band structure calculated by using previously published parameters[55~57] with the band structure which was experimentally known

and calculated by the pseudopotential method is not necessarily satisfactory, so that the parameters up to 2nd nearest neighbor interactions have been determined to improve the results.

In this chapter, firstly, the tight-binding method is described briefly and then the fitting procedure is mentioned which is followed by the results and discussions.

SECTION 2-2. TIGHT-BINDING METHOD

The Schrödinger eqation for one electron is expressed as

$$\mathcal{H}\psi_n = \epsilon_n \psi_n \qquad \text{Eq. 2-2-1}$$

,where $\mathcal{H}$ is a Hamiltonian, ψ_n is the eigen function belonging to an eigen value ϵ_n. In the tight-binding method, ψ_n is expanded by atomic orbitals φ_j's as

$$\psi_n = \sum_j c_j^n \varphi_j \ . \qquad \text{Eq. 2-2-2}$$

Then, substituting eq. 2-2-2 into eq. 2-2-1, multiplying φ_i^* from the left and integrating over the space coordinate (including spin coordinate if spins are taken into consideration), we have

$$H|\epsilon_n\rangle = \epsilon_n S|\epsilon_n\rangle \ . \qquad \text{Eq. 2-2-3}$$

,where

$$H_{ij} \triangleq \int \varphi_i^* \mathcal{H} \varphi_j \, d\tau, \qquad \text{Eq. 2-2-4}$$

$$S_{ij} \triangleq \int \varphi_i^* \varphi_j \, d\tau \ , \qquad \text{Eq. 2-2-5}$$

and

$$|\epsilon_n\rangle \triangleq (c_1^n, c_2^n, \cdots, c_N^n)^T \ . \qquad \text{Eq. 2-2-6}$$

In the simple tight-binding method, Sij is set equal to δij. This means that Sij (i≠j) is neglected or that the atomic orbitals are orthogonalized by the Löwdin method.[72] The distinction between these two interpretations is not made clear, but the assumption of Sij=δij is generally used quite flexibly. The resulting equation is

$$H|\epsilon_n\rangle = \epsilon_n|\epsilon_n\rangle \quad . \qquad \text{Eq. 2-2-7}$$

In the simple tight-binding method, the resonance integral, Hij, is considered as a parameter which is to be determined so as to reproduce the experimentally known band structure of a crystal solid. Since Hij is usually a rapidly decreasing function on the distance of the orbitals i and j, Hij between two distant orbitals are set equal to zero. Here, it should be noted that this approximation does not mean the exclusion of the functions which spread over a large area of the space. For example, the Bloch orbitals can be constructed within this approximation. If only the Hij's between the orbitals belonging to the nearest neighbor atoms (up to the 2nd nearest neighbor atoms) are taken into account, the approximation is called the "1st nearest neighbor approximation (2nd nearest neighbor approximation)".

SECTION 2-3. FITTING PROCEDURE

The 2nd nearest neighbor approximation is adopted here due to the reason given in section 2-4, and consequently 10 free interaction parameters exist. These parameters are the self-energy of Si s orbital (Es), that of Si p orbital (Ep), and the 8 resonance integrals which are given in table 2-3-1.

Let the vectors $\mathbb{P}^\circ=(P_1^\circ, \ \cdots \ ,P_{10}^\circ)^\top$, $\mathbb{E}^\circ=(E_1^\circ, \ \cdots \ ,E_N^\circ)^\top=\mathbb{E}(\mathbb{P}^\circ)$, and $\mathbb{E}^t=(E_1^t, \ \cdots \ ,E_N^t)^\top$ be the initial values of the interaction parameters, the initial energy eigen values, and the true energy eigen values which are derived from the experiments and the pseudopotential calculation. New set of parameters $\mathbb{P}^\circ+\Delta\mathbb{P}^\circ$ are determined so as to reproduce the true energy eigen values $\mathbb{E}^t$. Using Taylor expansion the problem can be linearlized in the form as

$$\mathbb{E}(\mathbb{P}^\circ+\Delta\mathbb{P}^\circ) \sim \mathbb{E}^\circ+\frac{d\mathbb{E}^\circ}{d\mathbb{P}^\circ}\Delta\mathbb{P}^\circ=\mathbb{E}^t \ . \qquad \text{Eq. 2-3-1}$$

$\Delta\mathbb{P}^\circ$ is calculated from this equation. The algorism is shown in fig. 2-3-1 as a flowchart. Generally speaking, however, eq. 2-3-1 can not be solved uniquely because it takes the form as

$$N\left\{\overbrace{\left[\frac{d\mathbb{E}^\circ}{d\mathbb{P}^\circ}\right]}^{10}\overbrace{\left[\Delta\mathbb{P}^\circ\right]}^{1}=\left[\mathbb{E}^t-\mathbb{E}^\circ\right]\right. \qquad N>10 \qquad \text{Eq. 2-3-2}$$

But it can be solved by using the method of least squares. This method is already enrolled in the program library of Tokyo university computer center as LESW1.

The fitted eigen values are seven valence band eigen values derived from the experiments, six conduction band eigen values given by the pseudopotential calculation, and three

eigen values of the Si (111) free surface with no relaxation and no reconstruction calculated by the pseudopotential method which is in good agreement with the experimental results. The eigen value problem of the Si free surface is solved by the method presented in Section 4-4. The fitting procedure is carried out with the special care to fit the bandgap of 1.12 [eV]. The computer programs used for this fitting (SBFIT and SBFIT3) are shown in Appendix A.

SECTION 2-4. RESULTS AND DISCUSSIONS

Figure 2-4-1 is the calculated energy band structure of Si with the new interaction parameters determined in this work together with the energy band structures calculated with the previously published parameters. The values of the parameters are listed in Table 2-4-1. The free surface states calculated with various sets of interaction parameters are compared with one another in Fig. 2-4-2.

Chadi's parameters[57] are obtained on the basis of the 1st nearest neighbor approximation. These parameters can reproduce the valence band but completely fail to reproduce the conduction band and the calculated bandgap is 3.41 [eV] as seen in Table 2-4-2. Dresselhaus and Dresselhaus's (D&D) parameters[36] show good agreement about the bandgap with the experimental

results but the calculated eigen value of the bottom of the valence band is very different from the true value. With Pandey and Phillips's (P&P) parameters[55] which include 8 non-zero parameters, the energy band structure can be calculated correctly except the bandgap whereas the calculated surface states are not in good agreement with the self-consistent pseudopotential calculation[58] as seen from Fig. 2-4-2.

On the other hand, the agreement of the calculated results by the parameters determined in this work which include 10 non-zero parameters, with the experimental results and the pseudopotential calculation results[55] is satisfactory. Of course the more number of parameters are included, the better fit with the true values can be achieved. However, the calculation with many parameters is tedious and sometimes impossible. So that the 2nd nearest neighbor approximation can be said the best approximation in dealing with the complex problems such as interface problems with microstructural disorders.

SECTION 2-5. SUMMARY

The interaction parameters for Si are determined within the 2nd nearest neighbor approximation with a special care to fit the calculated electronic structures near the bandgap to the experimental results and the pseudopotential results. The calculation with the parameters reproduces the true electronic

band structures of bulk Si and the energy eigen values of the Si (111) free surfaces satisfactorily, so that the present parameters can be used in the calculation of the electronic structures of the Si-SiO_2 system.

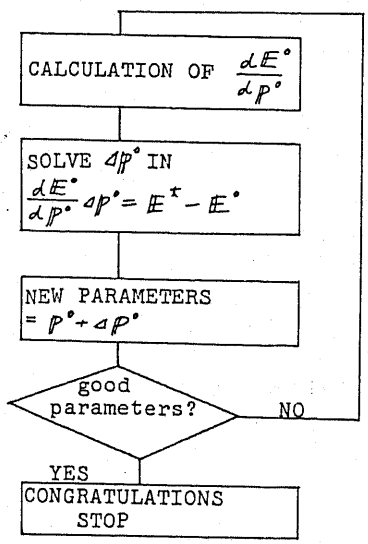


Fig. 2-3-1.

Fitting algorism

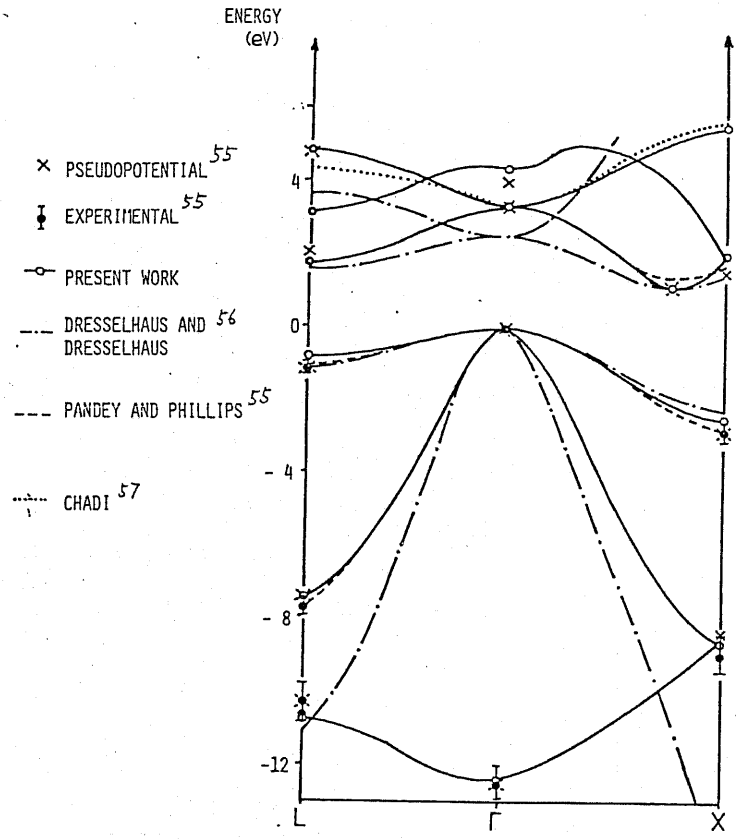


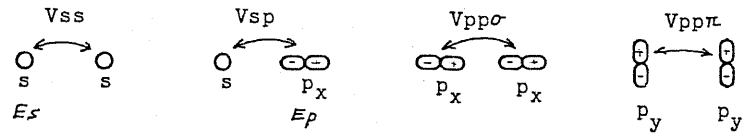
Fig. 2-4-1 CALCULATED ENERGY BAND STRUCTURE OF BULK SILICON USING TIGHT-BINDING HAMILTONIAN

In the present work Si bandgap is calculated to be 1.1 eV which agrees with the experimental result. The top of the valence band is chosen energy zero.

Table 2-4-1

Tight-binding parameters for silicon. Unit is eV. Notation is standard and schematically illustrated below the table.

	Es	Ep	V^1_{ss}	V^1_{sp}	$V^1_{pp\sigma}$	$V^1_{pp\pi}$	V^2_{ss}	V^2_{sp}	$V^2_{pp\sigma}$	$V^2_{pp\pi}$
present	-5.22	0.83	-2.05	2.09	2.33	-0.54	0.084	-0.29	0.47	-0.13
P & P [55]	-4.20	0.19	-2.08	2.12	2.32	-0.52	0.	0.	0.58	-0.1
Chadi [57]	-4.2	1.7	-2.08	2.76	3.13	-0.92	0.	0.	0.	0



Superscript 1 and 2 denote 1st and 2nd nearest neighbor interaction respectively.

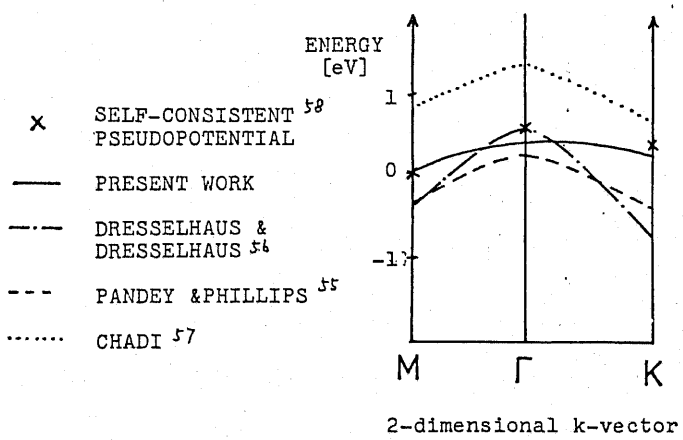


Fig. 2-4-2 Si (111) free surface state with no relaxation and no reconstruction. Calculated results with the other author's parameters are also shown.

Table 2-4-2. Calculated bandgap

	transition	bandgap [eV]
experimental [59]	indirect	1.12
present work	indirect	1.11
Dresselhaus et al [56]	indirect	1.1
Pandey et al [55]	indirect	1.40
Chadi [57]	direct	3.41

CHAPTER 3. EXTENDED HUCKEL THEORY

SECTION 3-1. INTRODUCTORY REMARK

In setting up the Hamiltonians for the Si-SiO_2 system which includes impurities such as H, Cl, and F, the simple tight-binding method described in Chap. 2 can not be directly applied because the electronic band structures of these materials are not sufficiently known. Therefore, some other method to calculate the tight-binding Hamiltonian must be investigated. The Extended Hückel Theory (EHT),[60] the Complete Neglect of Differential Overlap method (CNDO),[61] the Indirect Neglect of Differential Overlap method (INDO),[61] and the Modified Complete Neglect of Differential Overlap (MINDO) are the possible candidates for this. The INDO and MIMDO can only treat the elements up to Ne, that is, they can not treat Si properly. So that they are omitted. Shimizu et al[68] reported that the band structure of Si can not be reproduced by the CNDO method whereas it can be reproduced by the EHT. Taking this result into consideration, the EHT has been adopted here in calculating the Hamiltonians of the system including H, Cl, and F.

In Sec. 3-2, the EHT method is explained. Sections 3-3 and 3-4 are dedicated for the applications of this method to crystalline SiO_2 and to impurities (H, Cl, and F) in the

$Si-SiO_2$ system, respectively. Summary comes lastly.

SECTION 3-2. EXPLANATION OF THE METHOD

In the EHT, the ij element of the Hamiltonian is calculated as

$$Hij=-K_{\alpha(i),\alpha(j)}\cdot Sij\cdot(VOIPi+VOIPj)/2 \quad , \qquad \text{Eq. 3-2-1}$$

$$Hii=-VOIPi \qquad \text{Eq. 3-2-2}$$

,where VOIPi represents the valence orbital ionization potential of the i-th orbital whose value is listed in Table C-2, $\alpha(i)$ denotes the kind of the i-th orbital, say Si3s or Hls, $K_{\alpha(i),\alpha(j)}$ is a proprtional constant, and Sij is an overlap integral ($\int \varphi^*i\cdot\varphi j dx$). In calculating Sij, the Slater type orbital as

$$\varphi_{nlm}(r,\theta,\phi)=Nr^{n-1}e^{-\zeta_{nl}\cdot r}Y_{lm}(\theta,\phi) \qquad \text{Eq. 3-2-3}$$

is chosen as a basis. In Eq. 3-2-3, n, l, and m are principal, azimuthal, and magnetic quantum number, respectively, N denotes normalization factor, ζnl is a parameter which is given in Table C-1, and $Ylm(\theta,\phi)$ represents a spherical harmonic function. The calculation method of the overlap integrals

between two Slater type orbitals is given in Ref. 61,62.

The EHT was first successfully used for the organic materials including C, H, and O by Hoffmann in 1963[60] and afterwards great many applications are reported in the field of chemistry. The basis of the EHT approximation was given by Blyholder and Coulson[63] as follows. The LCAO molecular Hartree-Fock equations for a closed-shell configuration can be reduced to a form identical with that of the EHT if (1) we accept the Mulliken approximation[64] for overlap charge distributions, and (2) we assume a uniform charge distribution in calculating two-electron integrals over molecular orbitals. Numerical comparisons indicate that this approximation leads to results which, while unsuitable for high accuracy calculations, should be reasonably satisfactory for molecules that can not at present be handled with facility by standard LCAO molecular Hartree-Fock methods.

Although the EHT is widely used for organic molecules, the application to the amorphous Se,[65] transition metals in GaAs as impurities,[66] and Nitrgen in crystalline Si[67] have been reported with success. Good introduction for the EHT in Japanese is in the book titled "Rhoshi kagaku nyumon" by Yonezawa et al published from Kagaku- dojin K.K.

SECTION 3-3. APPLICATION TO CRYSTALLINE SiO_2

In order to see how the EHT works for the problems of

solids it is applied to the calculation of the electronic energy band structure of alpha-quartz and it is examined whether the EHT with K constants as parameters can reproduce the band structure calculated by the pseudopotential method and estimated by the experimental results. In this connection, if all $K_{\alpha(i),\alpha(j)}$ parameters are set equal to 1.75 which was used by Hoffmann[60] and believed to be valid in C-H sytem, the method is sometimes called the normal EHT, and on the other hand, if K's are varied as parameters, the method is sometimes called the modified EHT.

When all $K_{\alpha(i),\alpha(j)}$ are set equal to 1.75, the calculated bandbap of alpha-quartz was 16 eV and the top valence band width was 1 eV, which disagree with the experimental observation.[46-53] Therefore, the modified EHT is used and furthermore VOIPs are considered as parameters because these can be changed with the charge distribution. The parameters are chosen to fit the calculated energy levels of Γ and M points of alpha-quartz which have been shown[50] to reflect the peaks of density of states (DOS) to those estimated by the experiments. The structure and the 1st Brillouin zone of alpha-quartz are illustrated in Figs. 3-3-1 and 3-3-2, respectively. The 2nd nearest neighbor approximation is adopted in the calculation.

The fitting was possible. The calculated band structure are given in Fig. 3-3-3. The calculated DOS are shown in Fig. 3-3-4 and the peaks of DOS are in good agreement with the experiments[69] and other calculation results.[49, 50] Calculated

ionicity of 50% is consistent with the value estimated from Pauling's electronegativity consideration (see Table C-3 and Fig C-4 in Appendix C). Although the valence band widths are in good agreement with the results of the pseudopotential calculation,[49] the top of the highest valence band is located at point in the present calculation, which is different from the results of the pseudopotential method where the top exists at A or K. This discrepancy ,though small, can not be overcome as far as the approximation that K parameter between Px and Px is equal to that between Pz and Pz is employed. The bottom of the lowest conduction band shows the symmetry of Γ1 and consists of Si3s and O2s, which agrees well with the Fowler et al's results.[50]

The chosen parameters are listed in Table 3-3-1. Using these parameters, the band structure of beta-cristobalite is calculated. Figures 3-3-5 and 3-3-6 illustrate the struture and the 1st Brillouin zone of beta-cristobalite, respectively. The calculated band structure is shown in Fig. 3-3-7. The overall band positions are similar to those of alpha-quartz. This result indicates that the elctronic structures of various forms of SiO_2 are determined by the short range order of atomic configurations.

The application of the EHT to the problems of crystals using Bloch orbitals as a basis is briefly described in Ref. 70 . The actual calculation is carried out using the programs BAND, ALPHA, and BETA in Appendex A.

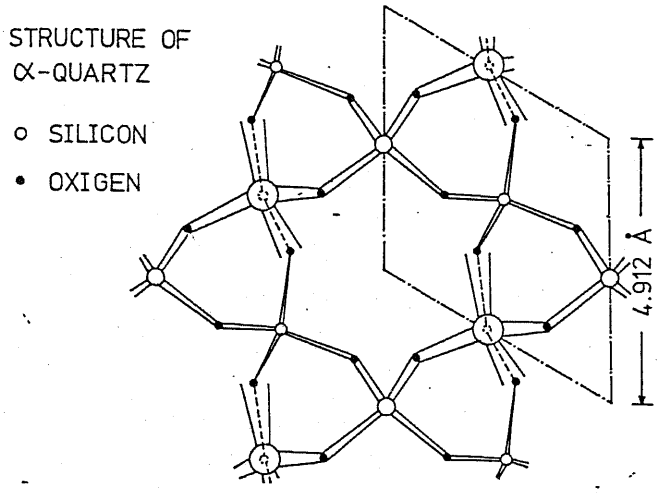


Fig. 3-3-1. Srtucture of α-quartz.

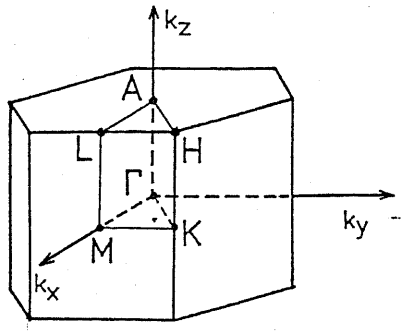


Fig. 3-3-2. 1st Brillouin zone of α-quartz.

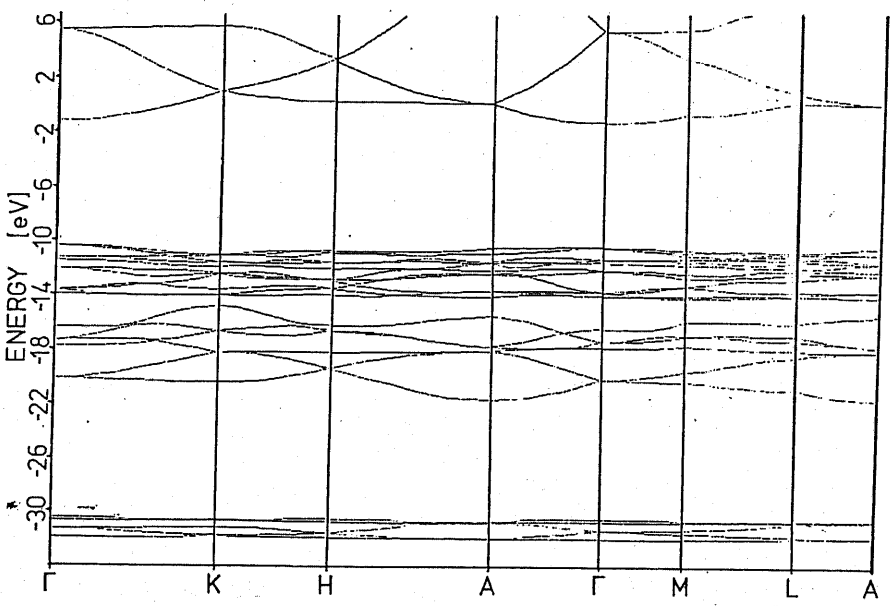


Fig. 3-3-3 Calculated band structure of α-quartz by the EHT.

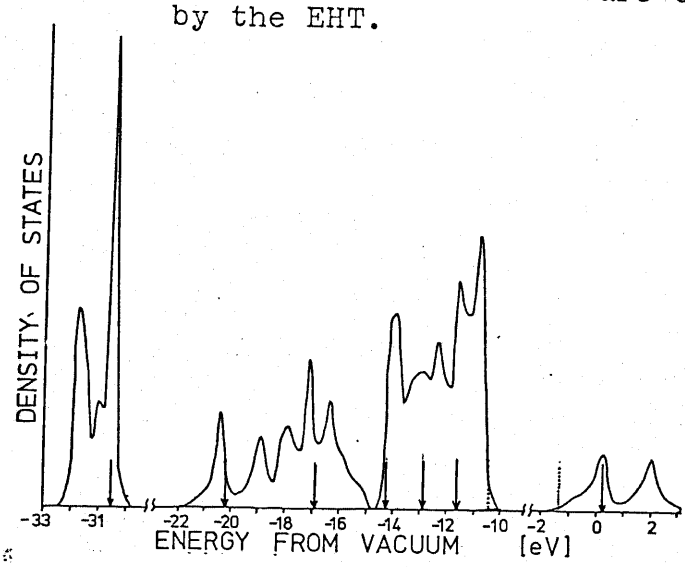


Fig. 3-3-4. Density of states of α-quartz. The arrows indicate the estimated peaks of DOS and dotted line indicate band edges.[69]

Table 3-3-1. Determined parameters.

H	Si3s,Si3s=-14.95	eV
H	Si3p,Si3p=-7.78	eV
H	O2s,O2s=-29.6	eV
H	O2p,O2p=-12.7	eV
K	Si3s,Si3s=0.87	
K	Si3p,Si3p=2.5	
K	O2s,O2s=2.5	
K	O2p,O2p=6.0	
K	Si3s,O2s=1.34	
K	Si3p,O2p=2.9	

K parameters between s and p orbitals are chosen to be an arithemetic average of those between s-s and p-p orbitals.

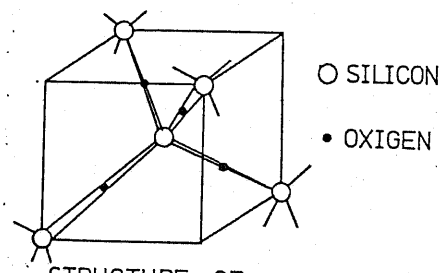


Fig. 3-3-5. STRUCTURE OF β-CRISTOBALITE.

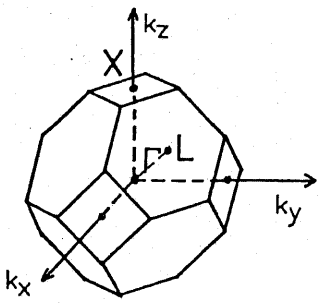


Fig. 3-3-6. 1st Brillouin zone of -cristobalite.

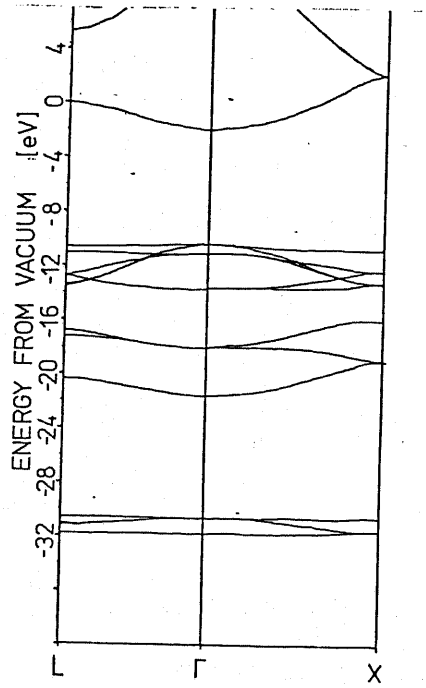


Fig. 3-3-7. Calculated band structure of -cristobalite.

SECTION 3-4. APPLICATION TO THE Si-SiO_2 SYSTEM INCLUDING IMPURITIES

The orbital exponents and the VOIPs used in the calculation of the energy levels of impurities in the Si-SiO_2 are listed in Tables 3-4-1 and 3-4-2, respectively. The nearest neighbor interaction is adopted for the impurity problem. The approximation of $Sij=\delta ij$ is also used according to Ref. 66. The normal bond lengths between O and H, Si and O, Si and H, Si and Cl, and Si and F are chosen to be 0.97, 1.61, 1.50, 1.50, and 1.50 [A], respectively. The bond lengths between O and H, and Si and H are cited from Ref. 71, and that between Si and O is a normal bond length in the SiO_2. The bond lengths between Si and Cl, and Si and F are not ascertained. But these ambiguities can be compensated by varying the K parameters in a wide range. In fact K parameters are varied between 1.0 and 4.0 in the calculation described in Chaps. 5 and 6, although the value between 1.0 and 2.0 is empirically believed to be probable for K parameters.[66,68] This variation of K parameters do not affect the results of Chaps. 5 and 6.

The EHT is also used in varying the bond lengths between Si and O, and Si and Si. That is, we assume that each simple tight-binding parameter is altered according to

$$Vij(Rkl)=Vij(R\mathring{k}l)\cdot Sij(Rkl)/Sij(R\mathring{k}l) \qquad \text{Eq. 3-4-1}$$

,where the interaction parameter and the overlap integral at the bond length of Rkl are denoted as Vij(Rkl) and Sij(Rkl), respectively and R^{o}_{kl} represents the bulk bond length.

The interactions between valence orbitals are taken into account

Table 3-4-1.

Employed exponents for Slater type orbitals. Fig. C-1

	ζ_{1s}	ζ_{2s}	ζ_{2p}	ζ_{3s}	ζ_{3p}
H	1.0				
O		2.2458	2.2266		
F		2.5638	2.5500		
Si				1.6344	1.4284
Cl				2.3561	2.0387

Table 3-4-2.

Employed values of VOIP. Fig. C-2

	VOIP1s	VOIP2s	VOIP2p	VOIP3s	VOIP3p
H	13.6				
O		32.33	15.80		
F		40.12	18.65		
Si				14.83	7.75
Cl				25.27	13.70

SECTION 3-5. SUMMARY

In conclusion, the modified EHT can be applied to crystal SiO_2 satisfactorily. This result encourages to use the EHT in the problems of solids and the determined parameters can be used in the further research. The calculated results indicate that the electronic structures of various form of SiO_2 is derived from the short range order such as the valencies and the tetrahedral SiO_4 structure.

In Sec 3-4, the values of the VOIPs and the orbital exponents are listed. The method for varying the simple tight-binding parameters is also described.

CHAPTER 4. GREEN'S FUNCTION FORMULATION

SECTION 4-1. INTRODUCTORY REMARK

Once the Hamiltonian is established, various energy levels can be calculated by solving the eigen value problem of Eq. 2-2-7. However, it is impossible to solve this eigen value problem because the size of the Hamiltonian is enormously large. But if the short range interactions are assumed, like in Chaps. 2 and 3, the calculation effort can be greatly reduced with the help of a Green's function formulation and the energy levels of the system can be obtained. In Secs. 4-2, the general Green's function theory is reviewed. Section 4-3 deals with the Green's function formulation with short range interactions which is developed in this work. This formulation is applied to the case of surface problems and the Bethe-lattices in Secs. 4-4 and 4-5, respectively.

SECTION 4-2. GENERAL GREEN'S FUNCTION THEORY

The Green's function G is defined as

$$G(E) = (EI - H)^{-1} = \sum_n \frac{|E_n >< E_n|}{E - E_n} \qquad \text{Eq. 4-2-1}$$

,where E denotes energy, I a unit matrix, H a Hamiltonian, and E_n and $|E_n\rangle$ the n-th eigenvalue and eigenvector, respectively. The local density of the j-th orbital($LDOS_j$) is expressed as

$$LDOS_j = \sum_n |\langle j|E_n\rangle|^2 \, \delta \, (E - E_n) \qquad \text{Eq. 4-2-2}$$

$$= -\frac{1}{\pi} \lim_{\delta \to \pm 0} I_m \, [G(E + i\delta)_{jj}] \qquad \text{Eq. 4-2-3}$$

, where $I_m[G(E + i\delta)_{jj}]$ is an imaginary part of the jj element of the matrix $G(E + i\delta)$.

If the LDOSj is summed over j, then it is easily shown that the resulting quantity is a total density of states. The weighting term $|\langle j|En\rangle|^2$ is the j-th orbital component of the n-th eigen function. Therefore, the LDOSj is considered as the j-th orbital contribution to the total density of states or it of interpreted as the density of states belonging to the j-th orbital. For example, if an eigen function is localized on the j-th orbital, then LDOSj shows a large peak at the corresponding energy eigen value. Since the Green's function has poles at the energy eigen values, it can be suited for the problem of defects and impurities.

When the $i\delta$ in Eq. 4-2-3 is approximated by a finite value due to the practical limitation, the δ-function in Eq. 4-2-2 has the Lorentzian broadness. That is, a sharp line spectrum at E_n is broadened by a factor of $\delta/\{(E - E_n)^2 + \delta^2\}$. However,

if the values y_1 and y_2 at two points E_1 and E_2 near the peak are given, then the peak value y_p and the peak energy E_p can be calculated by the following formulae,

$$E_p = \frac{(y_1E_1 - y_2E_2) + \sqrt{y_1y_2(E_1 - E_2)^2 - \delta^2(y_1 - y_2)^2}}{y_2 - y_1}$$

$$y_p = y_1 \left\{ 1 + \frac{(E_1 - E_p)^2}{\delta^2} \right\} . \qquad \text{Eq. 4-2-4}$$

The explanation of y_1, y_2, y_p, E_1, E_2, and E_p is given in Fig.4-2-1. Therefore, though δ=0.04 eV and energy step of 0.1 eV are adopted here, the error of calculation by this approximation is less than 0.01 eV with help of Eq. 4-2-4

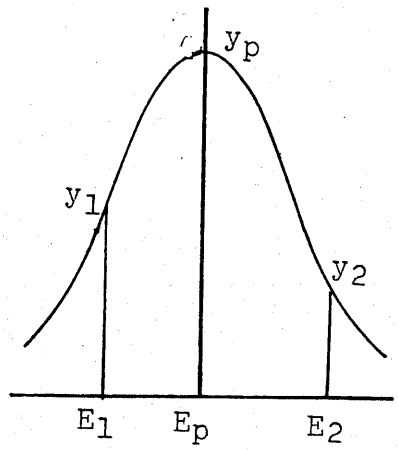


Fig. 4-2-1.

Lorentzian curve. If E_1, y_1, E_2, and y_2 are known, E_p and y_p can be calculated.

SECTION 4-3. GREEN'S FUNCTION WITH SHORT RANGE INTERACTIONS

The matrix size to be manipulated is greatly reduced if the interactions among atoms are short-ranged. When a system A and B interact each other through short-range interaction as shown in Fig. 4-3-1, the Hamiltonian and Green's function G are written in the forms

$$(EI-H)^{-1} = \left[EI - \left[\begin{array}{c|c|c|c} h_{AA}^{22} & h_{AA}^{21} & 0 & 0 \\ \hline h_{AA}^{12} & h_{AA}^{11} & V_{AB}^{11} & 0 \\ \hline 0 & V_{BA}^{11} & h_{BB}^{11} & h_{BB}^{12} \\ \hline 0 & 0 & h_{BB}^{21} & h_{BB}^{22} \end{array}\right] \right]^{-1} \qquad \text{Eq. 4-3-1}$$

$$= G(E) = \left[\begin{array}{c|c|c|c} G_{AA}^{22} & G_{AA}^{21} & G_{AB}^{21} & G_{AB}^{22} \\ \hline G_{AA}^{12} & G_{AA}^{11} & G_{AB}^{11} & G_{AB}^{12} \\ \hline G_{BA}^{12} & G_{BA}^{11} & G_{BB}^{11} & G_{BB}^{12} \\ \hline G_{BA}^{22} & G_{BA}^{21} & G_{BB}^{21} & G_{BB}^{22} \end{array}\right] \qquad \text{Eq. 4-3-2}$$

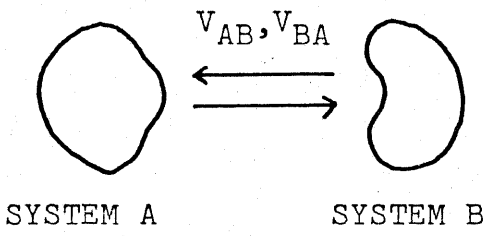


Fig. 4-3-1.

Schematic illustration that system A and B interact with each other.

On the other hand, when the system A and B are independent, as illustrated in Fig. 4-3-2, the Green's function for each system, g_{AA} and g_{BB} are written as

$$g_{AA} = \left[EI - \left[\begin{array}{c|c} h_{AA}^{11} & h_{AA}^{12} \\ \hline h_{AA}^{21} & h_{AA}^{22} \end{array} \right] \right]^{-1} = \left[\begin{array}{c|c} g_{AA}^{11} & g_{AA}^{12} \\ \hline g_{AA}^{21} & g_{AA}^{22} \end{array} \right] \quad \text{Eq. 4-3-3}$$

and

$$g_{BB} = \left[EI - \left[\begin{array}{c|c} h_{BB}^{22} & h_{BB}^{21} \\ \hline h_{BB}^{12} & h_{BB}^{11} \end{array} \right] \right]^{-1} = \left[\begin{array}{c|c} g_{BB}^{22} & g_{BB}^{21} \\ \hline g_{BB}^{12} & g_{BB}^{11} \end{array} \right] \quad \text{Eq. 4-3-4}$$

, respectively.

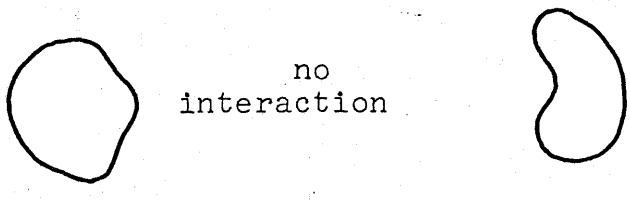
Then the following equations are obtained.

$$G_{AA}^{11} = \left((g_{AA}^{11})^{-1} - V_{AB}^{11} g_{BB}^{11} V_{BA}^{11} \right)^{-1} \quad \text{Eq. 4-3-5}$$

(equation for connection)

$$g_{AA}^{11} = \left((G_{AA}^{11})^{-1} + V_{AB}^{11} g_{BB}^{11} V_{BA}^{11} \right)^{-1} . \quad \text{Eq. 4-3-6}$$

(equation for separation)



SYSTEM A SYSTEM B

Fig. 4-3-2.

Schematic illustration that system A and B are independent.

The derivation of these equations are given afterwards.

These two equations show that the size of the matrix is small if the local density of states within a small part of the system is to be calculated. Moreover, connection and separation of atomic groups as illustrated in Fig. 4-3-3 can be done quite easily using these two equations. For example, it the system A is the crystalline Si and the system B is the amorphous SiO_2 then using Eq. 4-3-5, the Green's functions near the interface are calculated, and if G_{AA} is a Green's function of the crystalline Si at the perfect interface and g_{BB} is a Green's function of a Bethe-lattice and an extra oxygen atom, then using Eq. 4-3-6, the Green's function of the interface including a dangling bond (g_{AA}) can be obtained.

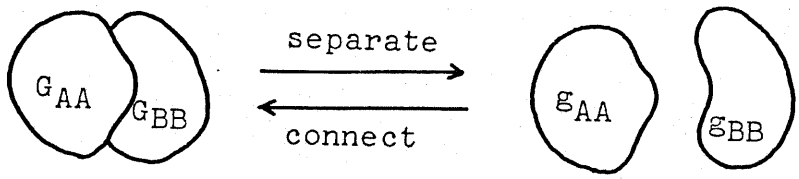


Fig. 4-3-3.
Schematic illustration of connecting and separating two systems.

The derivation of Eqs. 4-3-5 and 4-3-6 is carried out as follows. Cal culation of (EI-H)G=I using

$$\left[\begin{array}{c|c} G_{AA} & G_{AB} \\ \hline G_{BA} & G_{BB} \end{array}\right] = \left[\begin{array}{c|c} g_{AA}^{-1} & -V_{AB} \\ \hline -V_{BA} & g_{BB}^{-1} \end{array}\right] \qquad \text{Eq. 4-3-7}$$

leads

$$\left[\begin{array}{c|c} G_{AA}^{22} & G_{AA}^{21} \\ \hline G_{AA}^{12} & G_{AA}^{11} \end{array}\right] = \left[\begin{array}{c|c} g_{AA}^{22} + g_{AA}^{21} V_{AB}^{11} G_{BA}^{12} & g_{AA}^{21} + g_{AA}^{21} V_{AB}^{11} G_{AB}^{11} \\ \hline g_{AA}^{12} + g_{AA}^{11} V_{AB}^{11} G_{BA}^{12} & g_{AA}^{11} + g_{AA}^{11} V_{AB}^{11} G_{BA}^{11} \end{array}\right], \qquad \text{Eq. 4-3-8}$$

and

$$\left[\begin{array}{c|c} G_{BA}^{12} & G_{BA}^{11} \\ \hline G_{BA}^{22} & G_{BA}^{21} \end{array}\right] = \left[\begin{array}{c|c} g_{BB}^{11} V_{BA}^{11} G_{AA}^{12} & g_{BB}^{11} V_{BA}^{11} G_{AA}^{11} \\ \hline g_{BB}^{21} V_{BA}^{11} G_{AA}^{12} & g_{BB}^{21} V_{BA}^{11} G_{AA}^{11} \end{array}\right]. \qquad \text{Eq. 4-3-9}$$

Particularly, the following equations are useful,

$$G_{AA}^{11} = g_{AA}^{11} + g_{AA}^{11} V_{AB}^{11} G_{BA}^{11}, \qquad \text{Eq. 4-3-10}$$

$$G_{BA}^{11} = g_{BB}^{11} V_{BA}^{11} G_{AA}^{11}, \qquad \text{Eq. 4-3-11}$$

Substituting Eq. 4-3-11 into Eq. 4-3-10 and solving the resultant equation as to G , we have Eq. 4-3-5. Equation 4-3-6 is easily derived from Eq. 4-3-5. Other than Eq. 4-3-5 and Eq. 4-3-6,

$$G_{AA}^{11} = g_{AA}^{11} V_{AB}^{11} G_{BB}^{11} V_{BA}^{11} g_{AA}^{11} + g_{AA}^{11} \qquad \text{Eq. 4-3-12}$$

also holds.

When the approximation that Sij is set equal to δij is not valid, the Green's function G is defined as $(ES-H)^{-1}$ instead of $(EI-H)^{-1}$. Nevertheless, the discussions of this section are all valid if g and V are substituted by $(ES-h)^{-1}$ and V-ES, respectively. However, it should be notified that the quantity $-\frac{1}{\pi}\lim_{\delta\to+0}$ Im(Trace[G(E+iδ)]) does not give a DOS in this case. This fact is sometimes overlooked. The true formula which gives a DOS is

$$-\frac{1}{\pi}\lim_{\delta\to+0} \mathrm{Im}(\mathrm{Trace}[SG(E+i\delta)]), \qquad \text{Eq. 4-3-13}$$

$$-\frac{1}{\pi}\lim_{\delta\to+0} \mathrm{Im}(\mathrm{Trace}[G(E+i\delta)S]), \text{ or} \qquad \text{Eq. 4-3-14}$$

$$-\frac{1}{\pi}\lim_{\delta\to+0} \mathrm{Im}(\mathrm{Trace}[S^{\frac{1}{2}}G(E+i\delta)S^{\frac{1}{2}}]). \qquad \text{Eq. 4-3-15}$$

In this case of non-zero overlap, there is an overlap part contribution to DOS in addition to an orbital contribution which is considered in Eq. 4-2-2. The discussion of the non-zero overlap is also given in Sec. 4-5.

SECTION 4-4. APPLICATION TO SURFACE PROBLEMS

The problem of the electronic structures of solid surfaces has attracted much attention. Various methods have been developed and used to study surfaces of metals and semiconductors. However, all of the theoretical investigations based on realistic models of semiconductor surfaces belong to either of

the following major categories. One is a slab method[73] where the semi-infinite solid is simulated by a finite number of atomic layers having two-dimensional periodicity. The other is a perturbation method[74] where the semi-infinite solid is treated as a perturbed perfect solid.

Here, as an application of the method described in the previous section, we report a new method which does not belong to either of the two categories and which treats the semi-infinite solids as infinitely stacked atomic layers. It has advantages over the above-mentioned slab method in that it deals with the exactly semi-infinite solid, instead of a finite number of atomic layers. Consequently, the band continua are expressed in terms of continuous functions.

From the computational point of view, the sizes of the matrices to be manipulated in the present method are about a tenth of those in the slab method. Compared with the perturbation method, the present method is economical because there is no need for solving the perfect solid problem. Furthermore, different from the perturbation method, this method can treat the relaxed surfaces quite easily.

Since the present method is fit especially for examining the numerical validity of the widely used slab-type calculation, this course of study is carried out by using the Si (111) surface.

Although the description of the method is done by using the Si (111) surface, the method is quite general.

Let k be a two-dimensional reciprocal vector of the Si (111) surface. Then, taking Bloch orbitals as a basis set, the Hamiltonian H^k and the Green's function G^k for a certain k vector are expressed in the form

$$G^k(E)=(EI-H^k)^{-1}$$

$$= \left[EI- \begin{bmatrix} h^k_{11} & V^k_{1s} & & 0 \\ V^k_{s1} & h^k_{11} & V^k_{1s} & \\ & V^k_{s1} & h^k_{11} & V^k_{1s} \\ 0 & & V^k_{s1} & h^k_{11} & \ddots \\ & & & \ddots \end{bmatrix} \right]^{-1} \qquad \text{Eq. 4-4-1}$$

$$= \begin{bmatrix} G^k_{11} & & \\ & * & \\ & & \end{bmatrix} \qquad \text{Eq. 4-4-2}$$

,where h^k_{11}, V^k_{s1}, V^k_{1s}, and G^k_{11} are 8 x 8 matrices. The exact forms of these matrices are given in Ref. 75 .

Taking into account the fact that the semi-infinite solid should not be influenced by the addition of one atomic layer on the surface (Hereafter we will call this techniques as the layer stacking method. See Fig.4-4-1),the self-consistent equation for G^k_{11}

$$G^k_{11} = ((E + i\delta)I - h^k_{11} - V^k_{1s} G^k_{11} V^k_{s1})^{-1} \qquad \text{Eq. 4-4-3}$$

holds, which is derived by the use of Eq.4-3-5. G^k_{11} can be de-

termined by this equation. To solve Eq.4-4-3 in terms of G_{11}^{k} numerically, the relaxation method is effective. Since this relaxation process is rather oscillatory, the damping type procedure must be efficient to accelerate the convergence. In fact averaging operation over each oscillation period has shortened the computation time by a factor of 2-10. Once the G_{11}^{k}'s for various k-vectors are obtained, the Green's function G_{11} which takes normal atomic orbitals as a basis set is calculated by

$$G_{11} = \frac{1}{N}\sum_{k} G_{11}^{k} \qquad \text{Eq. 4-4-4}$$

,where N is the number of the unit cells.
The derivation of this equation is demonstrated in Appendix D. Any G_{11}^{k} in the course of the numerical relaxation is a Green's function for a finite slab, whereas the converged value is that for the semi-infinite solid. So that the comparison of surface LDOS between the slabs and the semi-infinite solid is easily carried out.

The two-dimentional Brillouin zone of the Si (111) surface is given in Fig. 4-4-2. Figure 4-4-3 shows how the surface LDOS changes as the number of atomic layers of the slab increases from 8 through infinity at the point of the Si (111) surface. The interaction parameters given in Ref. is used in the calculation. These results show that the gap state for the finite slab monotonously approaches that of the infinite solid and that the widely used slab approximation is excellent for the gap state level when the slab includes more than 30 atomic layers, but not so good for the band continua.

The calculation in this chapter is carried out by using SURF and SURF12 in Appendix A.

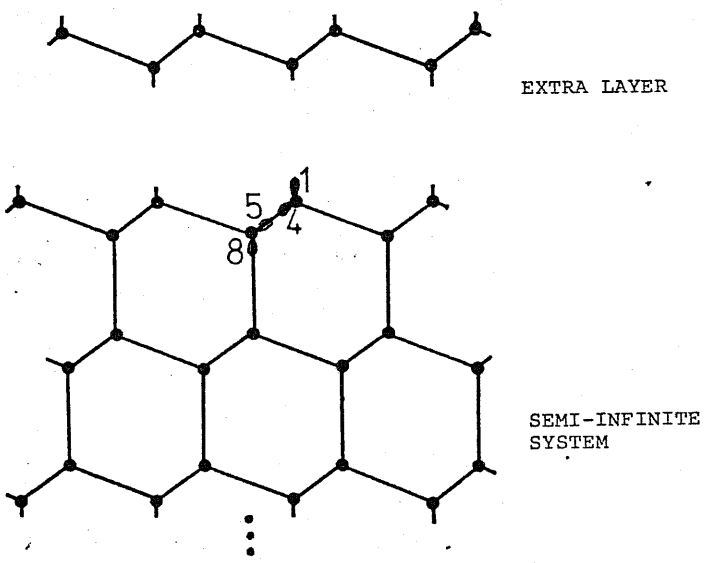


Fig. 4-4-1 Layer stacking method. When the system is semi-infinite, it should not be influenced by the addition of the extra atomic layers. The number denotes orbital number.

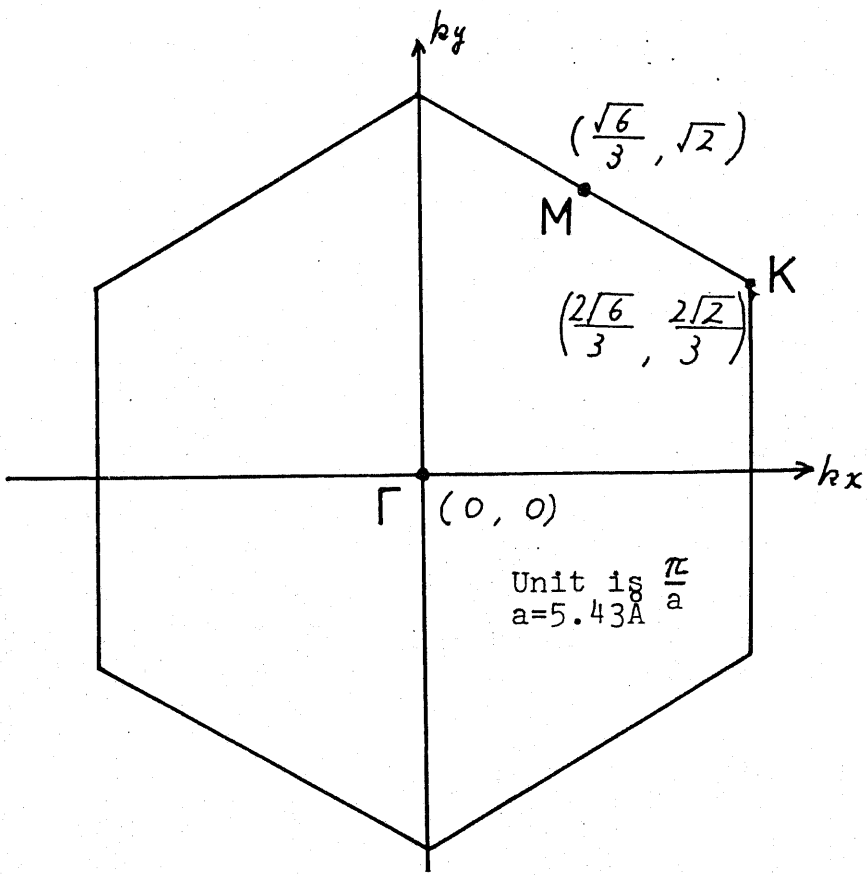


Fig. 4-4-2. Two-dimensional 1st Brillouin zone of Si (111) surface.

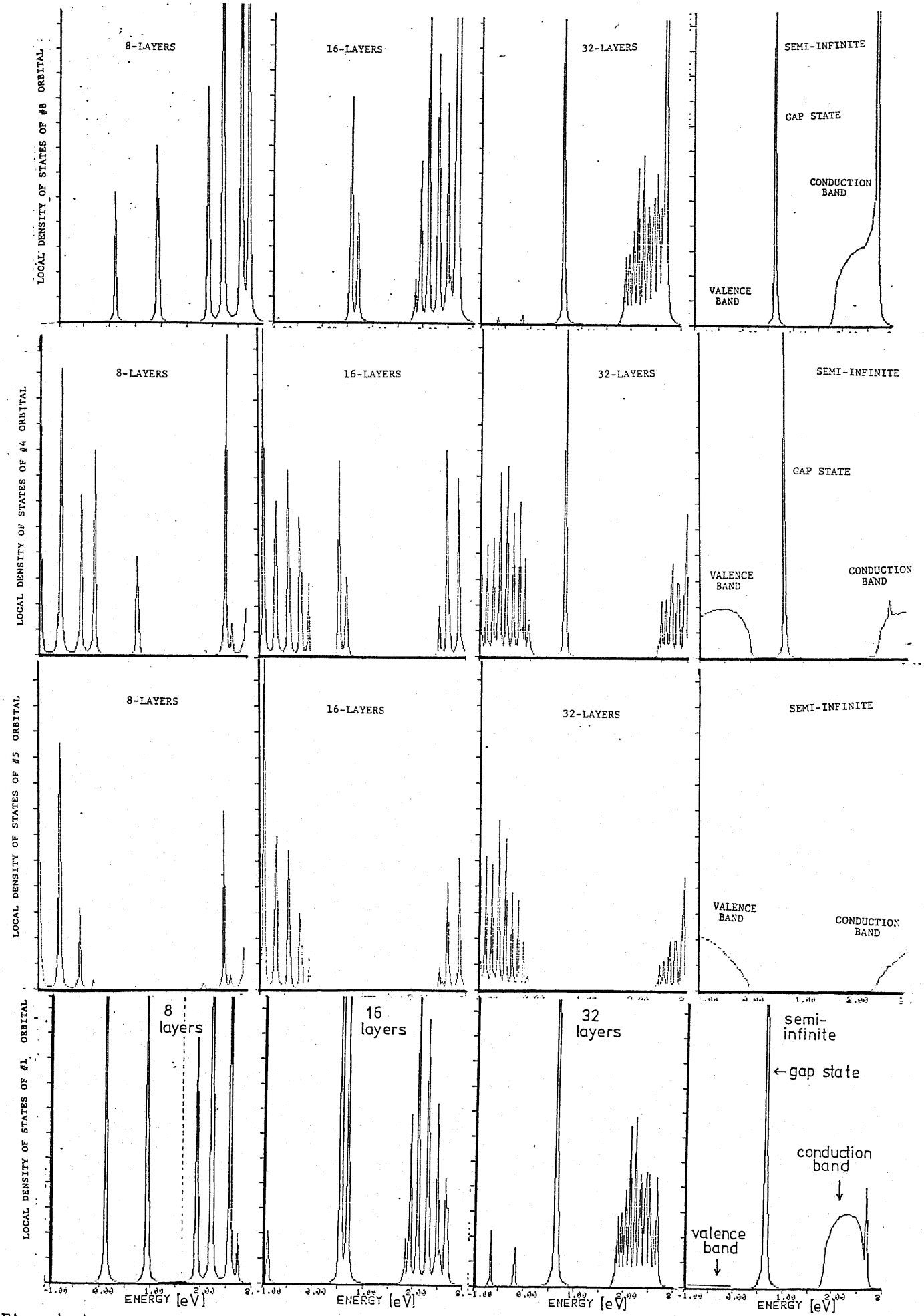


Fig. 4-4-3.

LDOS of free Si surface (at Γ). From left to right the number of layers increases. From top to bottom the orbital number is 8, 4, 5, and 1.

SECTION 4-5. APPLICATION TO BETHE-LATTICES

4-5-1.

The Green's function for the SiO_2 is also calculated by use of an equation of self-consistency. Since the SiO_2 Bethe-lattice employed here has been constructed by infinitely connecting SiO_3 units, one SiO_3 unit whose three oxygen dangling-bonds are terminated by the SiO_2 Bethe-lattices is equal to the SiO_2 Bethe-lattice itself. This fact leads to the equation of self-consistency similar to Eq. 4-4-3. The Green's function of SiO_2 Bethe-lattice is determined through this equation, where the manipulation of matrices sized 16x16 are required. The solution of the SiO_2 Bethe-lattice is first obtained by Laughlin and Joannopoulos[76] using a transfer matrix technique. In the present method, however, the Green's functions are computed directly without transfer matrices.

4-5-2. Since the case of SiO_2 is difficult as an introductory explanation, the simple case of tetrahedrally bonded solids with Weaire-Thorpe Hamiltonian is described as an example.

The Weaire-Thorpe Hamiltonian[77] takes sp^3 orbitals as a basis set and includes two interaction parameters as illustrated in Fig. 4-5-1. The energy zero is taken at the self energy of the sp^3 orbital. Let g_o denote the Green's function of the orbital 0 in Fig. 4-5-1. Then, the self-consistency mentioned above requires that the 1,1 element of

$$\left[\begin{bmatrix} E & -V1 & -V1 & -V1 \\ -V1 & E & -V1 & -V1 \\ -V1 & -V1 & E & -V1 \\ -V1 & -V1 & -V1 & E \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & V2 & 0 & 0 \\ 0 & 0 & V2 & 0 \\ 0 & 0 & 0 & V2 \end{bmatrix} \begin{bmatrix} 0 & * & * & * \\ * & g_o & * & * \\ * & * & g_o & * \\ * & * & * & g_o \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & V2 & 0 & 0 \\ 0 & 0 & V2 & 0 \\ 0 & 0 & 0 & V2 \end{bmatrix}\right]^{-1}$$

Eq. 4-5-1

is equal to g_o using Eq. 4-3-5. If the energy is measured in terms of V1 and V2/V1 is substituted by V, then we have

$$g_o = \frac{E - V^2 g_o + 2}{E(E - V^2 g_o + 2) - 3} \quad . \qquad \text{Eq. 4-5-2}$$

This quadratic equation of g_o have two roots. Solving Eq. 4-5-2 by the numerical relaxation method with the substitution of $E+i\delta$ into E corresponds in the physical sense to connecting more and more atoms to the system. So that this relaxation method gives the physically meaningful results. On the other hand, this relaxation method if applied to the problem of the type as

$$T^{(n)} = \frac{a + bT^{(n-1)}}{c + dT^{(n-1)}} \qquad \text{Eq. 4-5-3}$$

turns out to converge on the absolutely smaller root of the eqation

$$x = \frac{a + bx}{c + dx} \quad . \qquad \text{Eq. 4-5-4}$$

This is easily seen from the eqation

$$\frac{T^{(n)} - \alpha}{T^{(n)} - \beta} = \frac{\dfrac{a + bT^{(n-1)}}{c + dT^{(n-1)}} - \dfrac{a + b\alpha}{c + d\alpha}}{\dfrac{a + bT^{(n-1)}}{c + dT^{(n-1)}} - \dfrac{a + b\beta}{c + d\beta}}$$

$$= \frac{\alpha}{\beta} \cdot \frac{T^{(n-1)} - \alpha}{T^{(n-1)} - \beta} \qquad \text{Eq. 4-5-5}$$

,where α and β are two roots of Eq. 4-5-4. Therefore, the physically meaningful root of Eq.4-5-2 is that which has the smaller absolute value.

The Green's function of perfectly bonded Bethe-lattice, G, as illustrated in Fig. 4-5-2 is calculated as

$$G=(g_o - g_o V^2)^{-1} \qquad \text{Eq. 4-5-6}$$

which is derived by using Eq. 4-3-5. The LDOS calculated by Eq.4-5-6 with $V=\sqrt{3}$ is shown in Fig. 4-5-3. This figure suggests that the bandgap and the band continuum arise from the fact that the large number of atoms are bonded in the solids but not from the fact that the system has a translational symmetry.[90]

The band continuum is the region where the imaginary part of g_o or G is non-zero, that is,

$$[(E+1)^2-(V+\sqrt{3})^2-1][(E+1)^2-(V-\sqrt{3})^2-1]\leqq 0 . \qquad \text{Eq. 4-5-7}$$

This region is given in Fig. 4-5-4. When the energy is measured in terms of V2 and V1/V2 is denoted as V1, then the corresponding equation to Eq.4-5-7 is

$$[(E+\tfrac{\sqrt{3}}{4})^2+2(E+\tfrac{\sqrt{3}}{4})(V1-\tfrac{\sqrt{3}}{4})-3(V1-\tfrac{\sqrt{3}}{4})^2-\tfrac{1}{4}]\times$$
$$[(E-\tfrac{\sqrt{3}}{4})^2+2(E-\tfrac{\sqrt{3}}{4})(V1+\tfrac{\sqrt{3}}{4})-3(V1+\tfrac{\sqrt{3}}{4})^2-\tfrac{1}{4}]\leqq 0 \qquad \text{Eq. 4-5-7}$$

,whose graph is shown in Fig. 4-5-5. This graph agrees with

the band diagram calculated by the "cluster method".[78,79] The results of the cluster method are obtained through a numerical calculation, whereas the analytical expression is attained in the present method.

The Green's function of the i-th orbital (in Fig. 4-5-1) is expressed as

$$g_i = g^2V^2(y-G)(zV)^{2i-2} + G \qquad \text{Eq. 4-5-9}$$

,where

$$y = \frac{E(b+1)-2}{(b-1)(E(b+2)-3)}, \qquad \text{Eq. 4-5-10}$$

$$z = \frac{-1}{E(b+2)-3}, \qquad \text{Eq. 4-5-11}$$

and

$$b = E - g_0V^2. \qquad \text{Eq. 4-5-12}$$

These equations are derived by the use of Eq. 4-3-12. Since g_i should not be diverged, this formula shows that the LDOS of the system including a dangling bond approaches that of the perfectly bonded Bethe-lattice, as the orbital gets far away from a dangling bond.

4-5-3. Here, the effect of the non-zero overlap is investigated using the even simpler Hamiltonian. The Hamiltonian is based on tetrahedrally bonded s orbitals and consists of only one parameter as illustrated in Fig. 4-5-6.

If the non-zero overlap S is taken into account, the LDOS is formulated as

$$\mathrm{LDOS} = \frac{2}{\pi}\cdot\frac{\sqrt{12V^2-E^2}}{16V^2-E^2} + \frac{2}{\pi}\cdot\frac{SE}{V}\cdot\frac{\sqrt{12V^2-E^2}}{16V^2-E^2}. \qquad \text{Eq. 4-5-13}$$

the first term and the second term can be considered as an orbital contribution and an overlap part contribution, respectively. The second term can be 30% of the LDOS if S=0.1 and V=1 are adopted. So that Eqs. 4-3-13 ~4-3-15 must be used instead of Eq.4-2-3 when the non-zero overlap is taken into consideration. Usually, however, the proper set of interaction parameters with the neglection of the overlap integrals are employed to get rid of the complications.

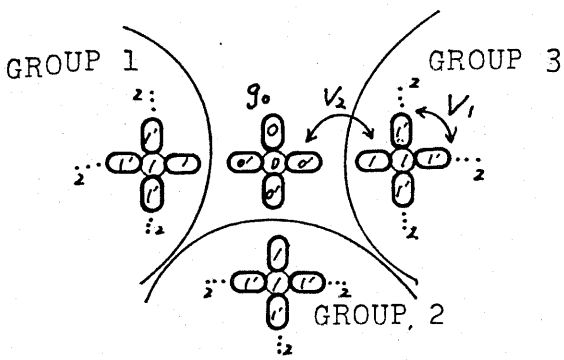


Fig. 4-5-1. Bethe-lattice with Weaire-Thorpe Hamiltonian. GROUP 1, 2, 3, and the whole system are the same one another.

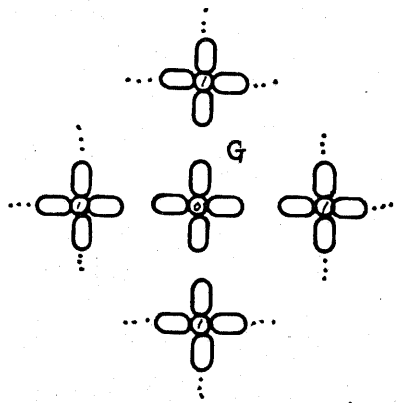


Fig. 4-5-2. Perfectly bonded Bethe-lattice.

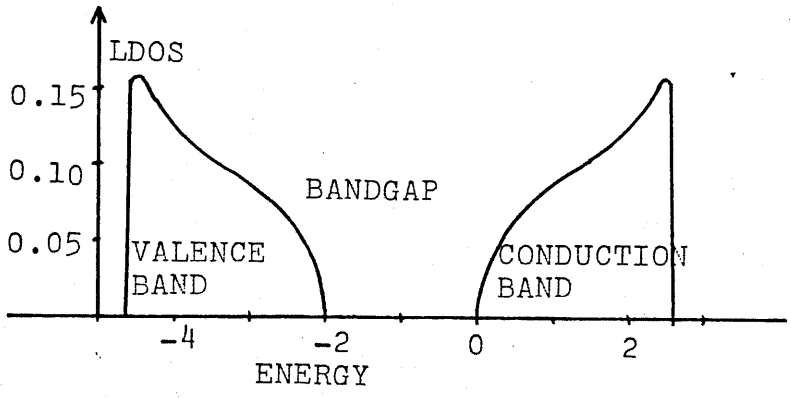


Fig. 4-5-3. LDOS of Bethe-lattice calculated by Eq. 4-5-6. This suggests that the bandgap and the band continuum arise not from the fact that the system has a translational symmetry but from the fact that the large number of atoms are bonded.

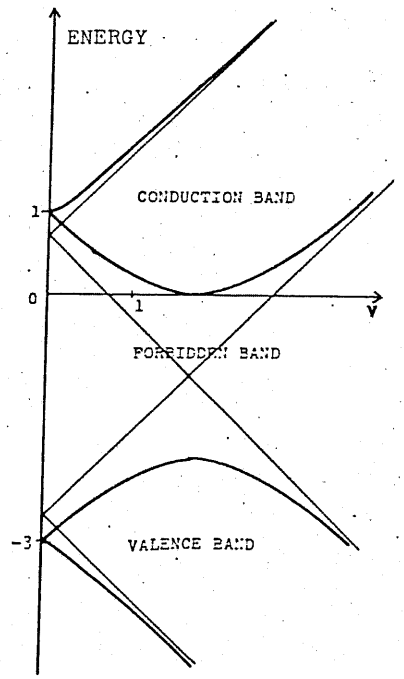


Fig. 4-5-4. Band diagram of Bethe-lattice. Parameter is V.

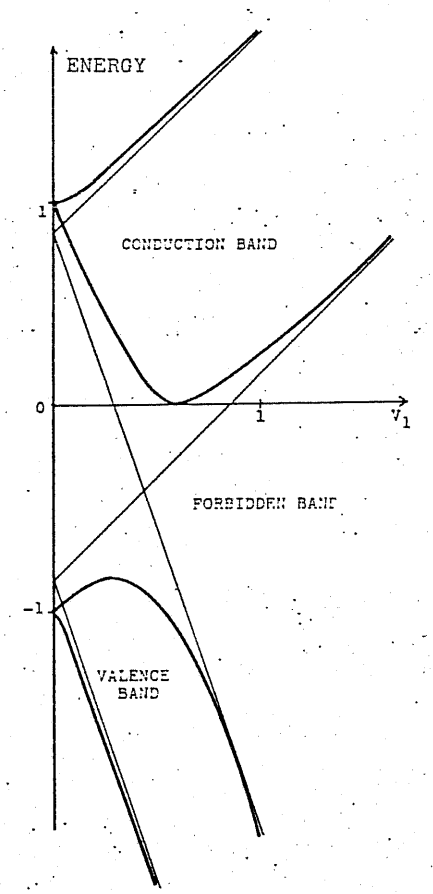


Fig. 4-5-5. Band diagram of Bethe-lattice. Parameter is Vl. This agrees with the calculated results by the "cluster method".

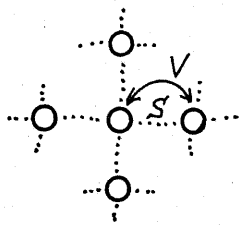


Fig. 4-5-6. Tetrahedrally bonded s orbitals.

SECTION 4-6. SUMMARY

The Green's function formulation can make use of the sparsity of the non-zero elements of the Hamiltonian and make the eigen value problem solvable. The local density of states which includes the information of the wave functions and energy levels is calculated from the Green's function.

The developed method itself is simple and flexible and can be applied to crystalline free surfaces with or without relaxation and/or reconstruction, impurity chemisorped surfaces, Bethe-lattices, bulk impurity problems, defects, and other complex problems such as superlattices and multilayer problems, if the proper tight-binding parameters are given. If a dynamic matrix is used for Hamiltonian, phonon spectra and Raman spectra can also be calculated.[80] It should be noted that the calculation method used in Chaps. 5 and 6 is mathematically exact except for the limitation caused by the practical numerical calculation such as the finite number of the sampling points in k space.

CHAPTER 5. INTERFACE STATES

SECTION 5-1. INTRODUCTORY REMARK

The trap states between Si-SiO_2 interface have been attracting much attention because these states take an important role in controlling the threshold voltage V_{th}, transconductance and flicker noise of MOS transistors. However, the chemical and physical origins of these states have not been fully understood, although some attempts were made to shed light on these problems by Chadi, Joannopoulos, and Laughlin.[80,81,82] In particular, little is konwn about the theoretical background of the fact that these interface trap states are continuously distributed in energy. Our work is an extension of Chadi et al's approach with a special emphasis on this continuous distribution of the trap states at Si-SiO_2 interfaces.

Our basic model has been constructed with a crystalline Si with (111) orientation and amorphous SiO_2 represented by a Bethe-lattice as shown in Fig.5-1-1. This can be considered as a Cluster-Bethe-Lattice model(CBLM) first introduced by Joannopoulos and Yndurain,[83] but in our case the cluster is the crystalline silicon, whose dangling bonds are terminated by the SiO_2 Bethe-lattices. The experimental evidences for very thin Si-SiO_2 transition layer[84,85] and the theoretical conclusion that the elastic energy of the Si-SiO_2 system is lowered as the width of the SiO_x layer goes zero,[86] encourage us to use this idealized abrupt-junction

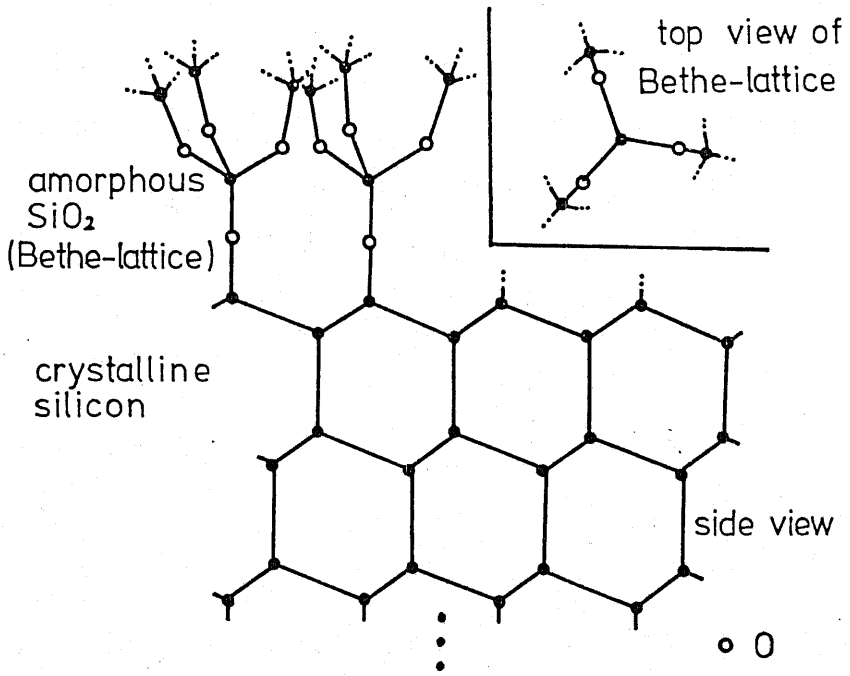


Fig. 5-1-1. BASIC MODEL

Basic model constructed with amorphous SiO_2 represented by Bethe-lattice and Si substrate with (111) orientation. Open and closed circles denote O and Si atoms, respectively.

model to simulate a thermally grown SiO_2-Si interface. Since oxygen chemisorption of 1/3 monolayer on a Si free surface is enough to cancel the surface reconstruction,[87] it is not needed to include the reconstruction of crystalline Si surface at the interface.

A Bethe-lattice used here to represent amorphous SiO_2 is a hypothetical, tree-like lattice containing no closed rings, where the valencies of O and Si are maintained 2 and 4 respectively. It is constructed by connecting SiO_4 tetrahedrals and has three-fold symmetry. Normal Si-O-Si angle is chosen to be 144° which is believed to be the peak value of Si-O-Si angle in amorphous SiO_2 system[88] and this angle is varied in the range between 120° and 180° in case that an amorphous effect is taken into consideration. The electronic structures of crystalline SiO_2 and amorphous SiO_2 are shown to be similar by optical absorption measurements[89], though the long-range atomic configurations are quite different. This suggests that the short-range orders such as valencies and SiO_4 tetrahedrals play an important role to determine the electronic structures, so that the Bethe-lattice model is thought to be a good approximation to represent amorphous SiO_2. (also cf. Sec. 3-5)

Based on this basic model we have calculated the energy levels of microscopic defects at Si-SiO_2 interfaces with changing various bonding parameters such as bond lengths, bond bending angles and bond rotation angles and investigated the possible origins of the interface traps continuously dis-

tributed in the forbidden gap of Si. Furthermore we have obtained the energy levels of impurities such as H, OH, Cl, and F bonded to Si dangling bond at the interface and discussed the annealing behavior of the interface traps.

In section 5-2, the various theories which deal with the calculation of the electronic structures of amorphous materials are discussed. Section 5-3 is dedicated for the models and procedures of the calculation of the Si-SiO_2 interface. Section 5-4 describes the calculated results of the interface states and impurities at the interface together with the proposed model for the interface states continuously distributed in energy and their annealing behavior. Section 5-5 is a summary.

SECTION 5-2. CALCULATION THEORIES OF THE ELECTRONIC STRUCTURES OF AMORPHOUS MATERIAL

The density of states is a basic physical quantity which has much to do with the photo characteristics and the electronic conduction of amorphous materials. In the following, various theories calculating the DOS are reviewed. From this review, it is shown that the 'Bethe-lattice model' is the only theory which can reproduce the electronic structure of amorphous materials and whose calculation is carried out in a practical length of time. For every theory , the following sequence is used: a) authors, b) reference number, c) year, d) materials which are actually treated by the theory, e) characteristics, and f) conclusions.

5-2-1. COHERENT POTENTIAL APPROXIMATION

a) Yonezawa et al, b) 91, c) 1975, d) actually no material is treated.

e) The extension of the theory which was successful in the theory of alloys.

f) It has not reached the stage of the application to the real material. The calculation effort is felt too heavy.

5-2-2. CLUSTER MODEL

a) McGill & Klima, Keller & Ziman, Okazaki et al, b) 92, 93, 94

c) 1972-1975, d) a-C, a-Si, vacancy in Si

e) The electron multiple scattering by many spherical muffin-tin atomic potentials (atomic cluster) which are floating in an effective potential is correctly calculated. Okazaki et al[94] introduced self-consistency in determining the effective potential.

f) The region where the DOS is small (pseudogap) certainly exists without assuming long range order of the atomic configuration (Fig. 5-2-1). If there is a vacancy of Si, then a peak in the DOS arises in the pseudogap. This corresponds to the generation of a dangling-bond state (Fig. 5-2-2). But even a large cluster which contains more than 30 atoms does not have a real bandgap where the DOS is completely zero (Fig. 5-2-3). This is due to the imperfect treatment of the surface of the

"atomic cluster".

5-2-3. BAND CALCULATION WITH POLYMORPH

a) Joannopoulos et al, Chin et al, b) 95, 96 , c) 1973-1975, d) Si, Ge

e) Band calculation by a pseudopotential method is carried out for complex structured crystals which have atoms of 2-12 in a unit cell. Chin et al calculated the band structure of a Si crystal which has 61 atoms in a unit cell by the LCAO method.

f) The differences between the DOS of a-Si and that of crystal are that the top of the valence band gets steeper in a-Si and that the fine structures in DOS disappear in a-Si,[97] as shown in Fig. 5-2-4. These effects are reproduced in Fig. 5-2-5. But the physical reasons of these effects are not clear. The calculation is tedious.

5-2-4. PERTURBED CRYSTAL MODEL

a) Kramer, b) 98 , c) 1970, d) a-Ge, a-Si, a-Se

e) If the crystal undergoes a certain kind of perturbation, the Hamiltonian becomes non-Hermite and the energy eigen values have imaginary part, that is, the band structure is blurred.

f) It is interesting in the sense that the transition from the crystal to the amorphous state can be analyzed. However, the existing amorphous material can not be condidered as a

perturbed crystal, so that the theory loses its basis. The calculated results (Fig. 5-2-6) of amorphous material shows no change from a crystal in the valence band, which does not agree with the experimental results.

5-2-5. SIMPLE TIGHT-BINDING MODEL

a) Weaire & Thorpe, b) 77, c) 1971, d) tetrahedrally bonded material

e) The theoretical discussions are made by using a very simple Hamiltonian given in Sec. 4-5 (Fig. 4-5-1).

f) They proved the mathematical fact that the bandgap larger than abs(2xV2-4xV1) is produced, only assuming that the atoms are bonded with four surrounding atoms, and without assuming the definite atomic configuration. If V1 and V2 fluctuate in the range of V1±|d1| and V2±|d2|, respectively, then the minimal bandgap decreases to

$$2(|V2|-|d2|)-4(|V1|+|d2|).$$

5-2-6. MOLECULAR ORBITAL METHOD

a) Chen, Shimizu et al, b) 99, 100, c) 1973-, d)Se, AsSe, SiO_2, defects, etc.

e) Energy levels are calculated for imaginary molecules or large molecules.

f) The method itself has been established in the field of

chemistry and is considered to be quite general. A calculated example is shown in Fig. 5-2-7. Treatments of molecular surfaces and the wave functions extended all over a material are problems.

5-2-7. CLUSTER BETHE-LATTICE MODEL

a) Joannopoulos et al, b) [101] , c) 1974-, d) Si, Ge, SiO_2

e) Bethe-lattices are connected to the surface of a certain cluster of atoms (Fig. 5-2-8). The Bethe-lattice is an imaginary, tree-like, and infinite connected network of atoms with the same coordination number as found in the real material but without rings of bonds. It provides a "neutral" background upon which the individual properties of the cluster are studied. The reasons for using the Bethe-lattices as a boundary condition are that the electronic structure of the system can be solved exactly, that DOS of the Bethe-lattice is featureless and smooth, and that the short range order of the atomic configuration found in a real material is maintained in the Bethe-lattice.

f) In Fig. 5-2-9, they concluded that the steepness of DOS near the top of the valence band is due to the short range disorder such as the distorted tetrahedral structure, whereas the smoothness of DOS at the lower half of the valence band is due to the middle range disorder.

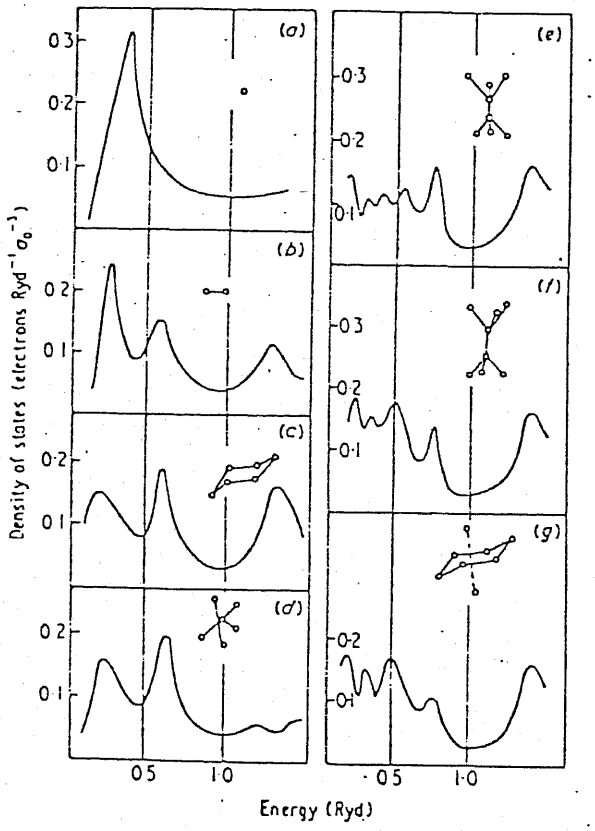


Fig. 5-2-1.[92] DOS for various carbon clusters by cluster method.

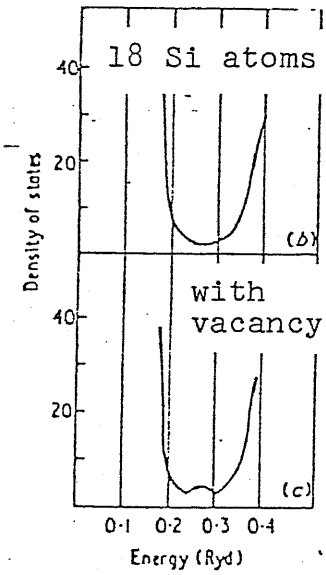


Fig. 5-2-2.[94] DOS peak is generated by Si vacancy.

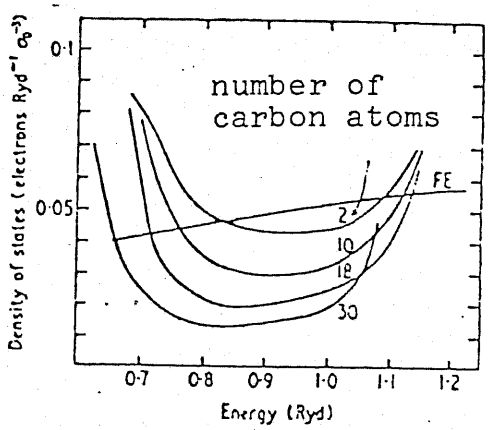


Fig. 5-2-3.[93] Complete bandgap can not be reproduced even if the cluster contains as many as 30 atoms.

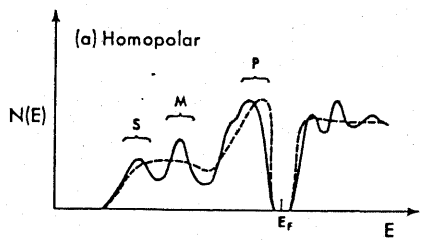


Fig. 5-2-4.[101] Sketch of DOS. Solid line is for amorphous, and dotted line is for crystal. DOS near valence band top gets steeper and fine structure disappears when in amorphous.

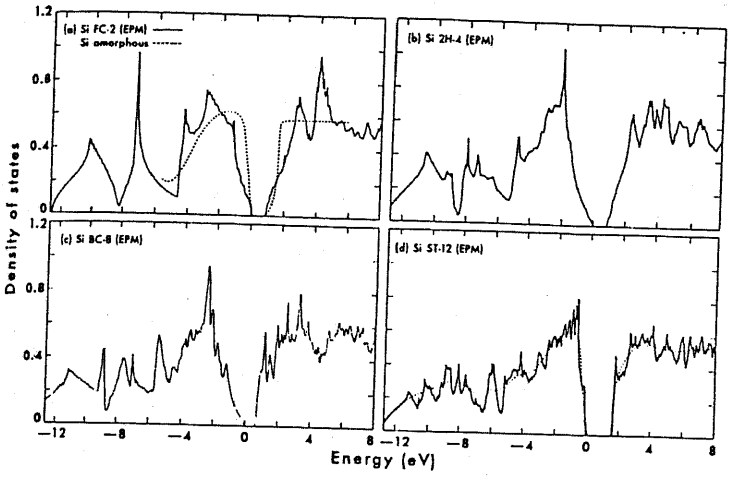


Fig. 5-2-5.[95] Band calculation for polymorph. Structure gets more complex from (a) to (d).

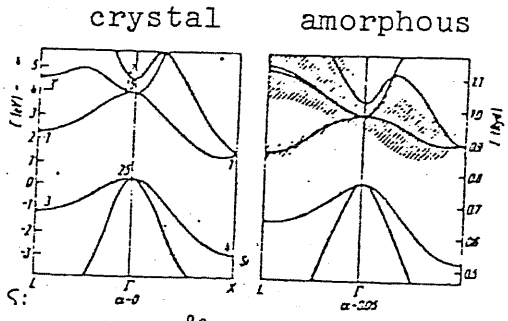


Fig. 5-2-6.[98] Band structure is blurred in amorphous phase.

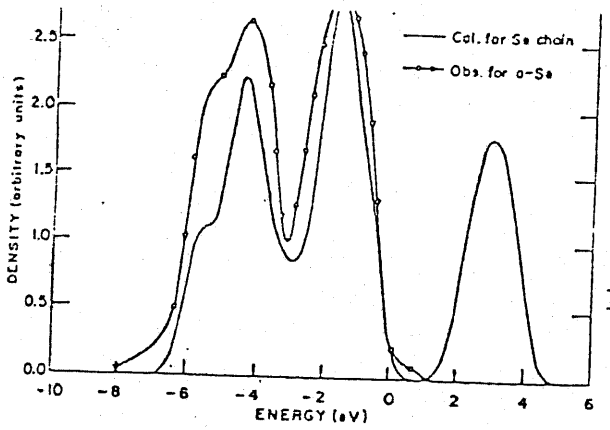


Fig. 5-2-7.[100] DOS for Se by EHT. Energy levels are Gaussian-broadened.

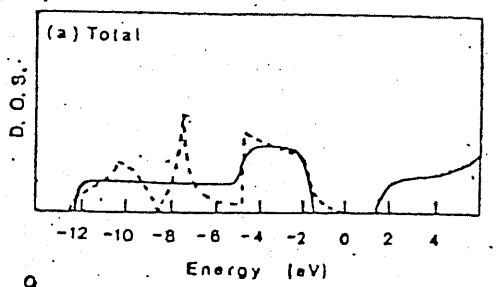


Fig. 5-2-9 Densities of states for the Bethe lattice The filled valence bands lie at negative energies. (a) Total density of states with superimposed (dashed line) crystalline density of states.

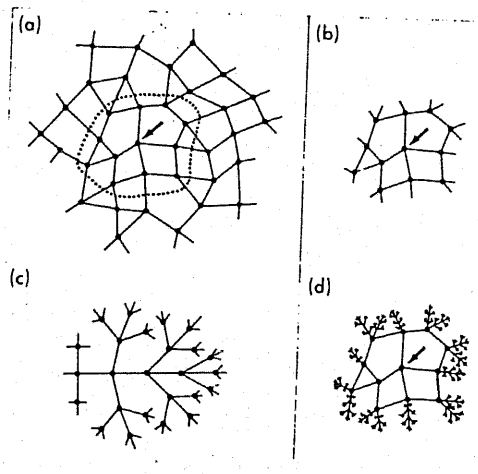


Fig. 5-2-8.[101] The "cluster-Bethe lattice" method involves (a) taking an infinite system of atoms and choosing one atom as a reference point; (b) removing a small cluster surrounding and including this atom from the system; (c) using a Bethe lattice, which is an infinite system of atoms with fixed coordination and no rings of bonds to (d) satisfy the dangling bonds of the cluster. This new structure is called a "cluster-Bethe lattice" system.

SECTION 5-3. MODELS FOR CALCULATION AND CALCULATION PROCEDURE

Models of the atomic configurations used in the calculation are shown sechematically in Fig. 5-3-1. The following is the procedure of the calculation.

(1) Fix one certain k-vector.

(2) Calculate the Green's function for Si (111) free surface (Fig. 5-3-1(a)).

(3) Connect a Bethe-lattice and one oxygen to this surface (Fig. 5-3-1(b)).

(4) Sum up the Green's functions over various k-vectors.

(5) Separate the Bethe-lattice and one oxygen to form a $Si_3 \equiv Si-$ dangling bond (Fig. 5-3-1(c)).

(6) Bring the Bethe-lattice closer to the Si dangling bond to simulate Si-Si bonding and O-Vacancy (Fig. 5-3-1(d)).

(7) Bring the Bethe-lattice and one oxygen closer to the Si dangling bond to form Si-O weak bond at the interface (Fig. 5-3-1(e)).

(8) Bond any of H, O, OH, Cl and F to the Si dangling bond to represent the bonding of impurity atom (Fig. 5-3-1(f)).

These are the rough sketches of the calculation procedure and other atomic configurations used in the calculation will be shown schematically in each time in the results.

The tight-binding parameters for SiO_2 employed in the present calculation have been determined by Chadi et al. The values are given in Table 5-3-1. The orbitals used in the calculation of the SiO_2 Bethe-lattice are illustrated in Fig. 5-3-2.

The computer programs used in this chapter are shown in Appendix A. The calculation has been carried out at the computer center of the Univ. of Tokyo with some library subroutines provided by the center.

The Hamiltonian and the basical calculation method are explained in the previous chapters.

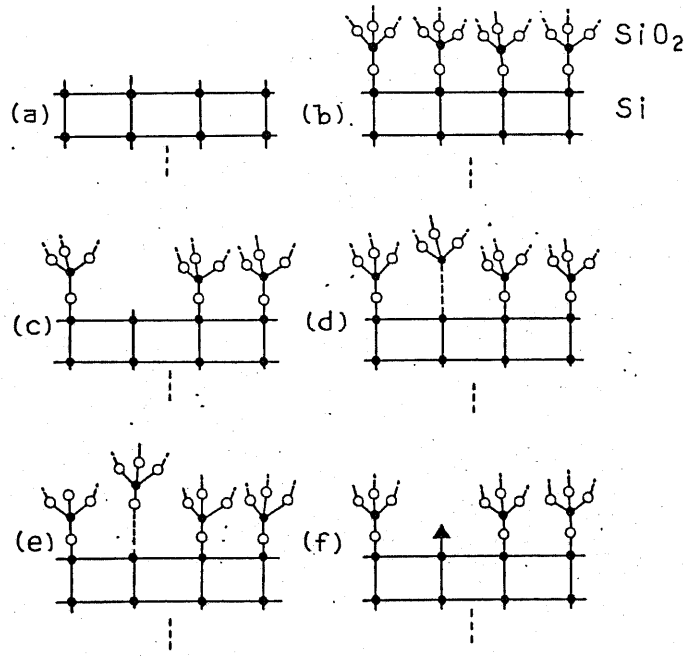


Fig. 5-3-1.
Models of various configirations used in the calculation. a) Si free surface, b) perfect interface, c) $Si_3\equiv Si$-dangling bond, d) Si-Si weak bond and O-vacancy at interface, e) Si-O weak bond at interface, f) impurity at interface. Closed circle, open circle and closed triangle denote Si, O, and impurity atoms (H, OH, Cl, and F), respectively.

Table 5-3-1.

Tight-binding interaction parameters (in eV) for SiO_2. The unprimed (primed) subscripts refer to Si(0) orbitals. Notation is standard and simlar to that used in Table

Es=4.42	Ep=10.67	Es'=-14.63	Ep'=-1.83
Vss'σ=-2.85	Vs's'σ=-0.15	Vsp'σ=-5.4	Vs'pσ=-9.5
Vpp'σ=-5.4	Vpp'π=-1.4	Vp'p'σ=-0.45	Vp'p'π=-0.45

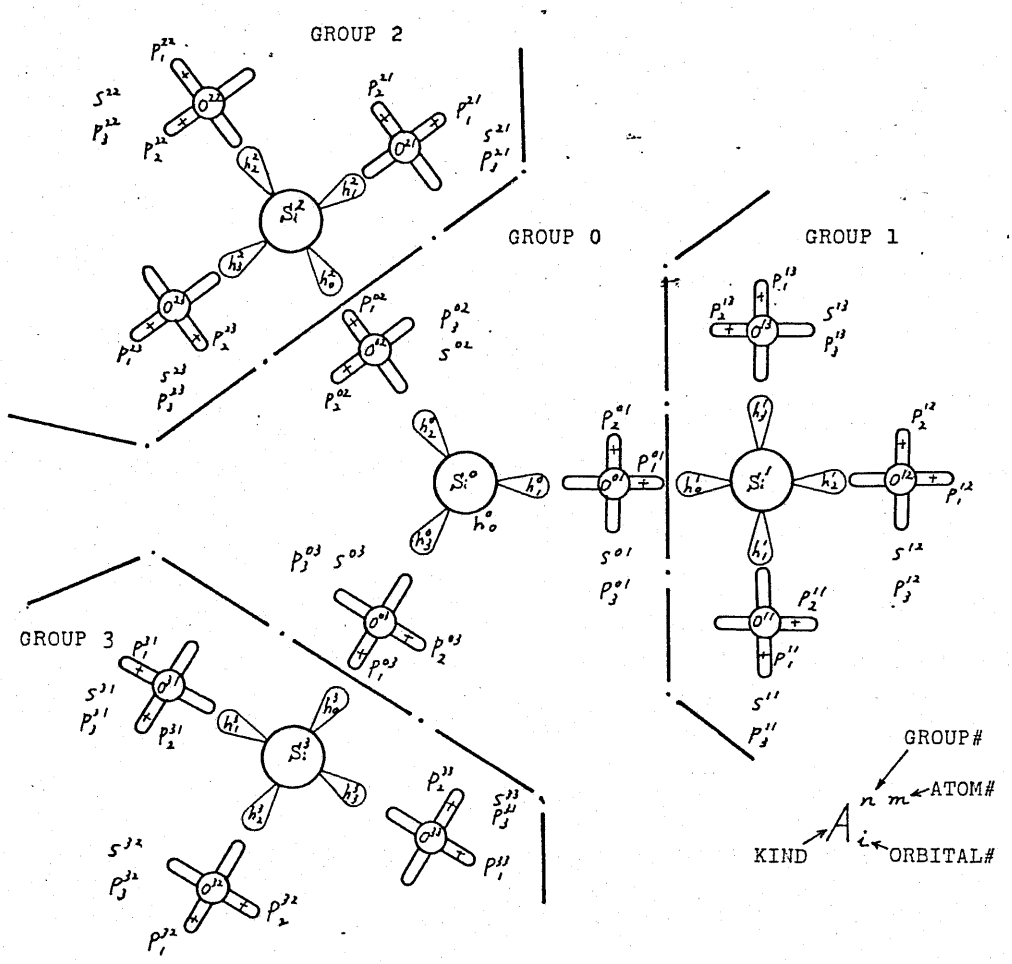


Fig. 5-3-2. Orbitals used in the calculation of the SiO_2 Bethe-lattice.

SECTION 5-4. RESULTS AND DISCUSSIONS

Both the perfect interface with the Si-O-Si bond angle ranged from 120° to 180° and the interface with oxygen dangling bond do not have a gap state as illustrated in Fig. 5-4-0. However, $Si_3\equiv Si-$ dangling bond at the interface gives rise to a gap state at about the middle of the Si bandgap as shown in fig.10, as Laughlin et al have indicated. But it should be noted that Si bandgap in their calculation was about 2.5 eV due to the nearest neighbor approximation and a Bethe-lattice approximation for the Si substrate, whereas in our model Si bandgap is calculated to be exactly 1.1 eV

The O-vacancy and Si-Si weak bond at the interface produce trap states in the Si bandgap, whose energy levels vary in the energy range lower than the midgap of Si by changing various bonding parameters. Figure 5-4-1 demonstrates the dependence of Si-Si weak bond level (including O-vacancy level) on the Si-Si bond length. Figures 5-4-2 and 5-4-3 show how O-vacancy level varies in energy depending on the bond bending angle, and the rotation angle, respectively. In these bonding parameters, bond length variation gives the strongest effect on the energy level.

When the Si dangling bond interact weakly with Si atom which is already bonded to four oxygen atoms in the SiO_2 network, the level also appears in the Si bandgap. This level moves in the lower half of the Si bandgap when the distance between these two Si atoms are varied from 2.3 Å to infinity as is shown in Fig. 5-4-4.

On the contrary, Si-O weak bond and Si-O weak interaction produce trap levels in the upper half of the Si bandgap. Si-O weak bond state changes its energy level depending on the bond length and the bond bending angle as demonstrated in Figs. 5-4-5 and 5-4-6, respectively. Si-O weak interaction indicates the situation where the Si dangling bond interacts weakly with O atom which is already bonded to two Si atoms in the SiO_2 network. Figure 5-4-7 shows the dependence of the Si-O weak interaction level on the distance between Si atom and O atom which is already bonded to two Si stoms in the SiO_2 network.
In calculating the energy levels of Si-Si weak interaction shown in Fig. 5-4-4 and those of Si-O weak interaction shown in Fig. 5-4-7, the interaction parameters among atoms having normal valency have been used. To weaken these interaction parameters is almost equivalent to increase the distance between the atoms at the interface. (cf. Fig. 5-4-8)

The bonding parameters at the actual $Si-SiO_2$ interface can be supposed to vary because of the amorphous structure and the large internal stress included in this system. Therefore, Si-Si weak bond, Si-Si weak interaction, Si-O weak bond, and Si-O weak interaction are thought to be the possible origins of the interface trap states continuously distributed in energy. Commonly observed U-shaped distribution of the interface trap densities (Fig. 5-4-9) is explained if bond length distributions in Fig. 5-4-10 and energy level dependences on the bond length as Fig. 5-4-1 and Fig. 5-4-5 are assumed. The rapid decrease of the distribution (Fig. 5-4-10) as the bond length

increases is reasonable because the normal bond lengths are shorter than 2.5 Å and the longer bond length are less likely.

The gap states move out of the energy range between 0.5 eV below the top of Si valence band and 0.5 eV above the bottom of Si conduction band when any of H, OH, Cl and F is bonded to the Si atom at the interface. This situation is illustrated in Fig. 5-4-11. Since an energy level outside the Si bandgap can not work as a trap state at the interface under normal operating conditions, this result explains the reduction of the interface trap density by H_2 annealing, trichrolo-ethylene annealing, and HCl oxidation and further suggests the possibility of F annealing.

The level moves up in the conduction band of Si in bonding with either of H, OH, Cl, and F. (cf. Fig. 5-4-12 and 5-4-13)

AT THE Si - SiO_2 INTERFACE

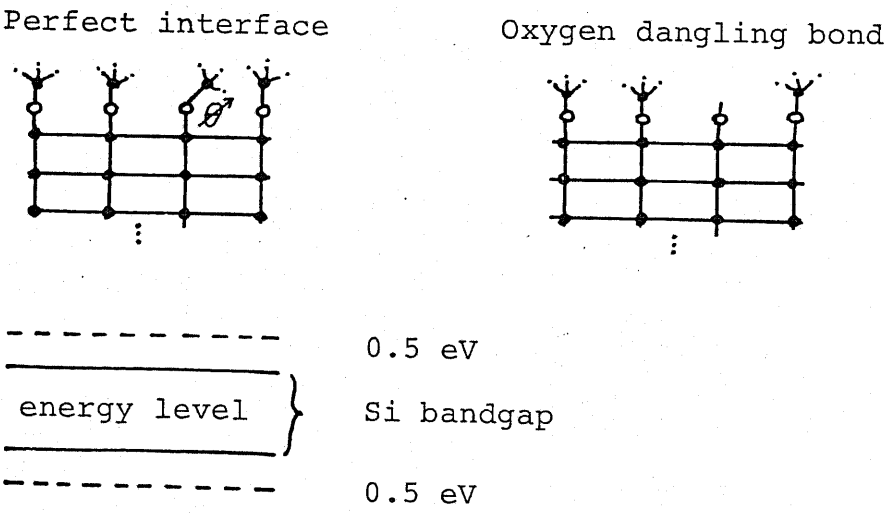


Fig. 5-4-0. Perfect interface and oxygen dangling bond at the Si-SiO_2 interface have no energy level in the range between 0.5 eV below the top of the valence band and 0.5 eV above the bottom of the conduction band of Si. Open and closed circles denote O and Si atoms, respectively.

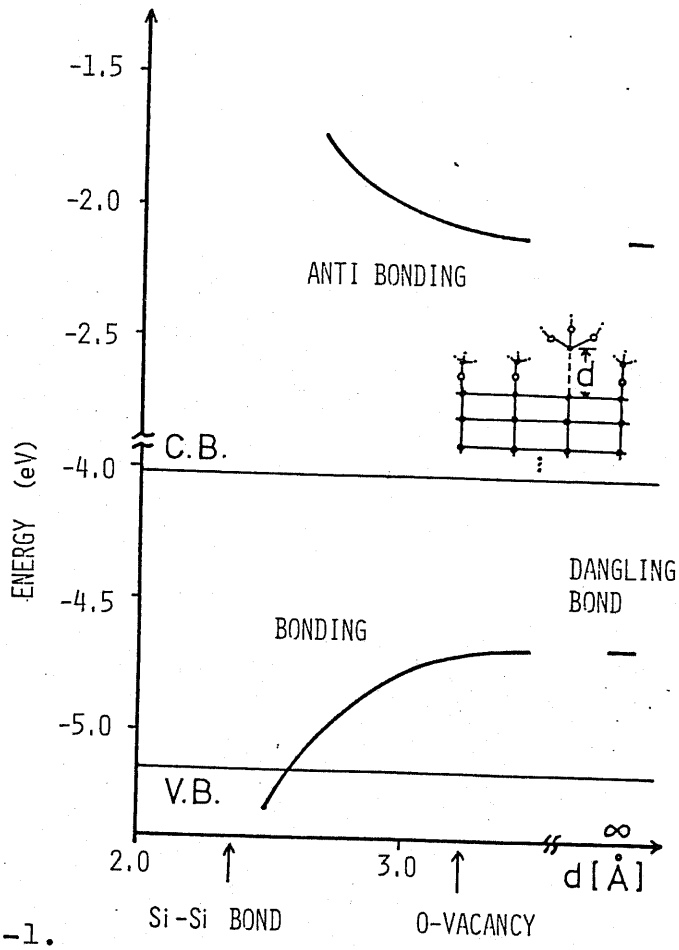


Fig. 5-4-1.
Si dangling bond, Si-Si bond, and O-vacancy level at the interface. These energy levels move in the lower half of the Si bandgap with changing the bond length d. Open and closed circles denote O and Si atoms, respectively.

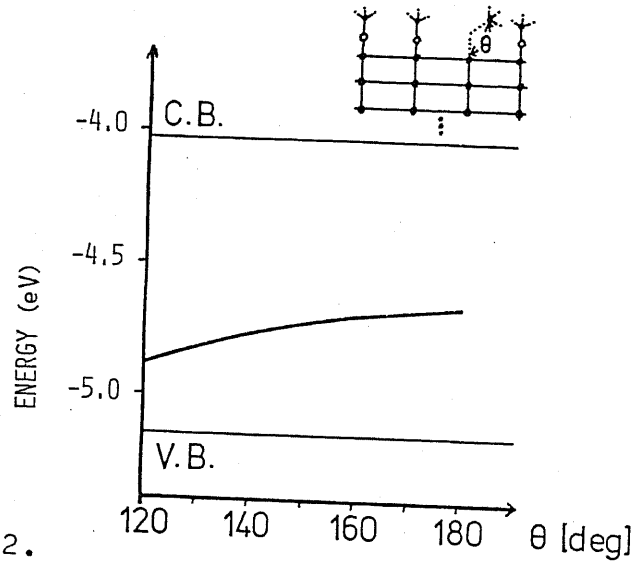


Fig. 5-4-2.

O-vacancy level and its dependence on the bond bending angle. Open and closed circles denote O and Si atoms, respectively.

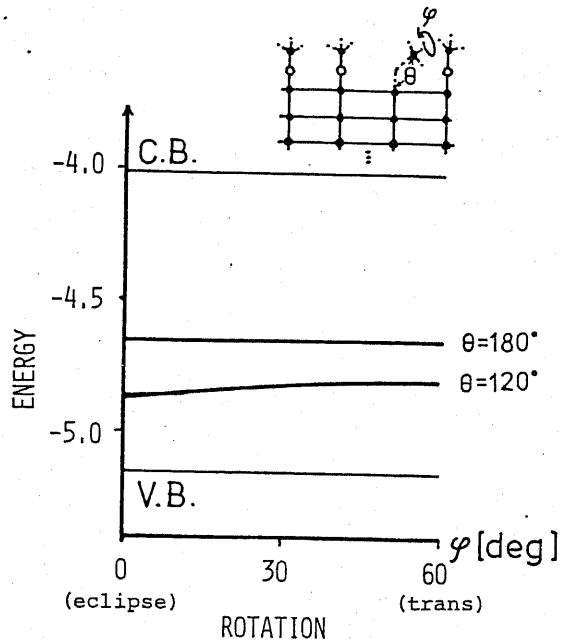


Fig. 5-4-3.

Dependence of O-vacancy level on the bond rotation angle. This dependence is very weak. Open and closed circles denote O and Si atoms, respectively.

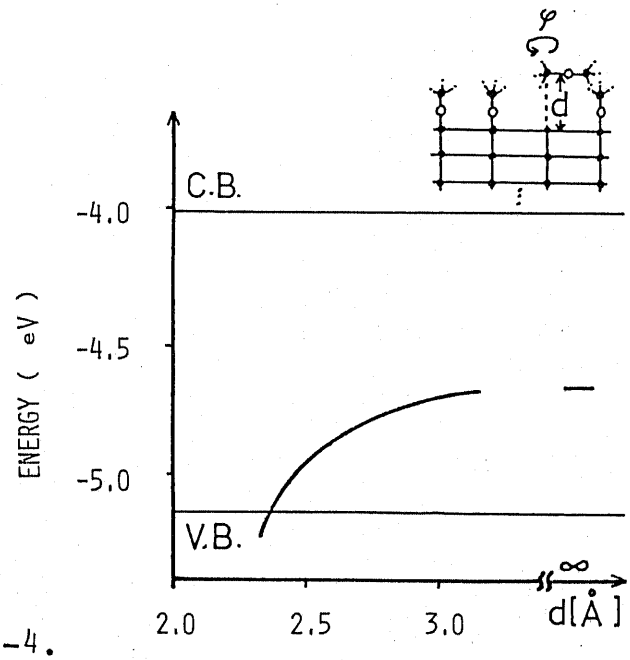


Fig. 5-4-4.

Energy level of Si dangling bond weakly interacting with Si atom which is already bonded to four oxygen atoms in SiO_2 network. The energy level varies with changing the distance d. φ-rotation gives no significant effect on the energy level. Open and closed circles denote O and Si atoms, respectively.

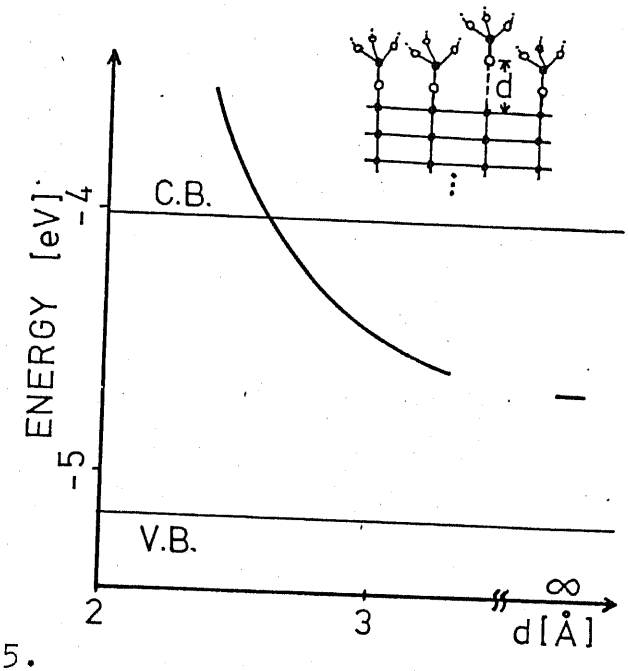


Fig. 5-4-5.

Energy level of Si-O weak bond at the interface. This energy level moves in the energy range higher than the midgap with changing the bond length d. Open and closed circles denote O and Si atoms, respectively.

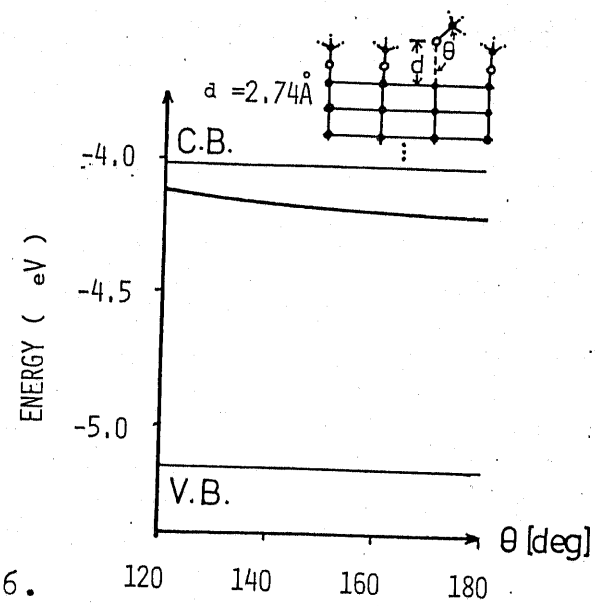


Fig. 5-4-6.

The dependence of the energy level of Si-O weak bond on bond bending angle. The dependence is minor. Open and closed circles denote O and Si atoms, respectively.

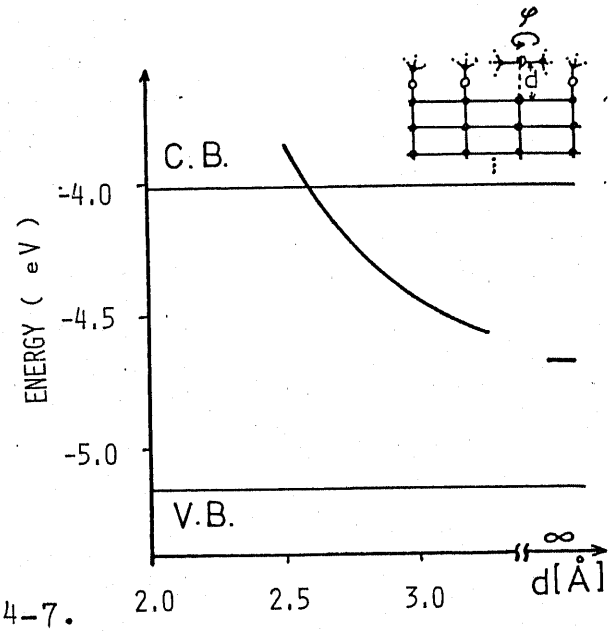


Fig. 5-4-7. Energy level of Si dangling bond weakly interacting with oxygen atom which is already bonded to two Si atoms in SiO_2 network. The energy level varies with changing the distance d. φ-rotation gives no significant effect on the energy level. Open and closed circles denote O and Si atoms, respectively.

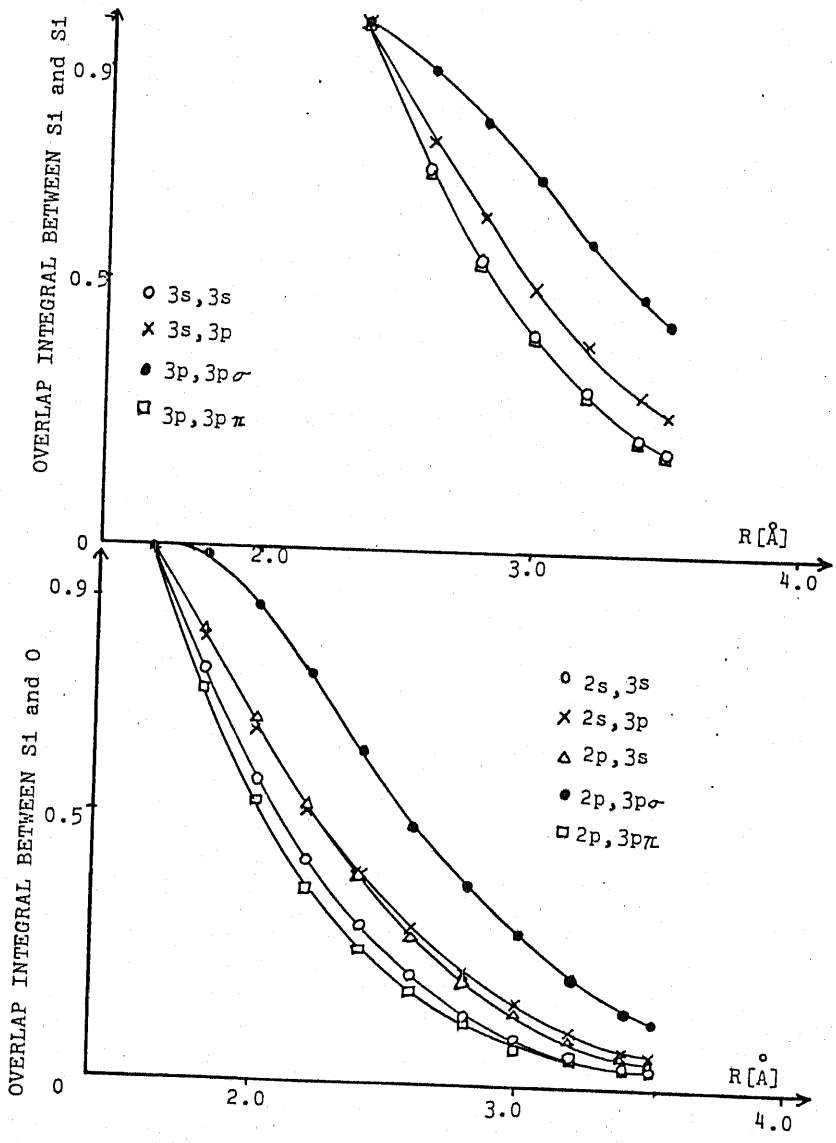


Fig. 5-4-8.

Values of overlap integrals dependent on distances of orbitals, R.

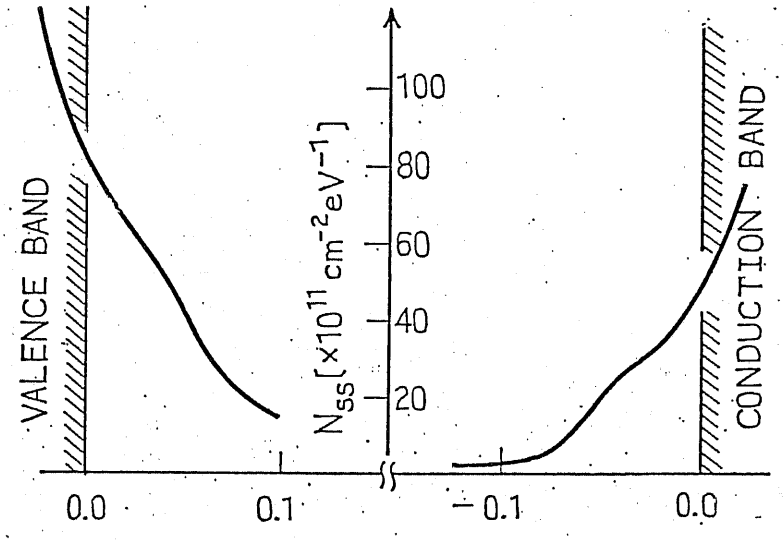


Fig. 5-4-9. Commonly observed U-shaped distribution of interface trap-state density in the forbidden gap of Si. (Cited from Ref./02)

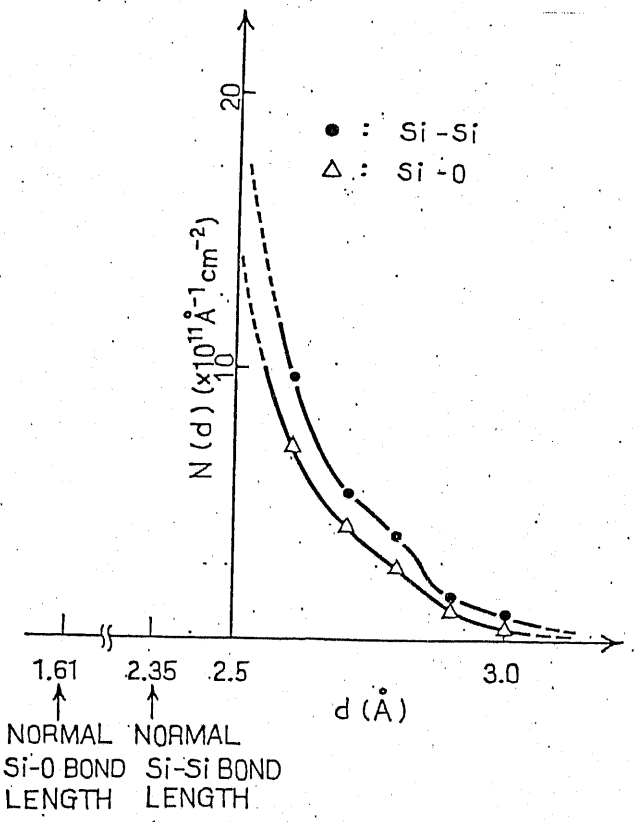
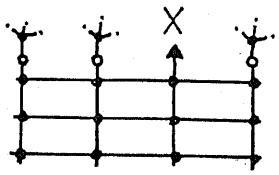


Fig. 5-4-10. Assumed bond length density N(d) ($Å^{-1}$ cm^{-2}) vs. bond length d (Å). Using this distribution together with the energy level dependence on bond length of Fig. 5-4-1 and 5-4-5, U-shaped distribution as shown in Fig. 5-4-9 can be explained.

(1) At the $Si-SiO_2$ interface



No level exists in and near the Si bandgap.

X=any of H, OH, Cl, and F

Fig. 5-4-11. Impurity at the $Si-SiO_2$ interface. If any of H,OH, CL, and F is bonded to the Si dangling bond, no energy level exists in the energy range between 0.5eV below the top of the valence band and 0.5eV above the bottom of the conduction band of Si. Open and closed circles denote O and Si atoms, respectively.

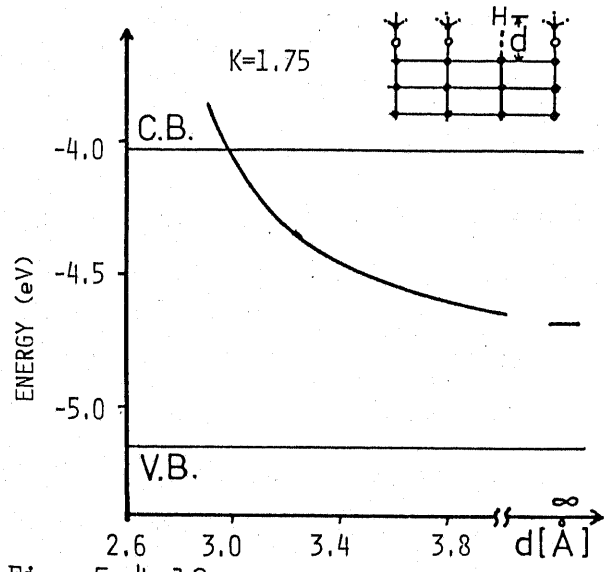


Fig. 5-4-12.
STRETCHED Si-H BOND AT INTERFACE

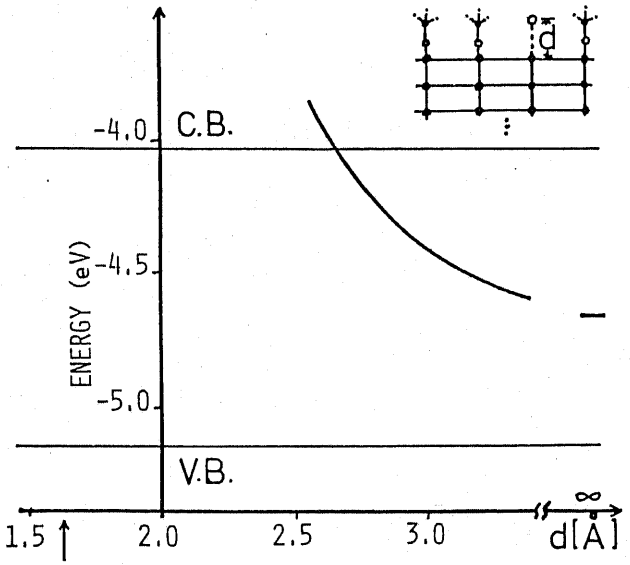


Fig. 5-4-13.
STRETCHED Si-O ATOM BOND AT INTERFACE

SECTION 5-5. SUMMARY

A calculation procedure dealing with the electronic structures of crystalline Si-amorphous SiO_2 interface including microstructural defects based on semiempirical tight-binding Hamiltonians and Green's function formulation is applied to the calculation of interface trap states between amorphous SiO_2 and the Si substrate with (111) orientation.

The following results are obtained. Perfect interface and the interface including oxygen dangling bond have no energy level in the Si bandgap, whereas $Si_3 \equiv Si-$ dangling bond has a energy level at about the middle of the Si bandgap. Si-Si weak bond and weak interaction at the interface give rise to gap states whose energy move in the energy range lower than the midgap with varying the distances between two Si atoms, while the energy levels of Si-O weak bond and weak interaction at the interface appear at the upper half of the forbidden gap of Si depending on the distances between the Si atom and the O atom.

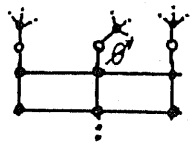
Possible origins of interface trap states which are distributed continuously in Si energy gap are suggested to be these Si-Si weak bond, Si-Si weak interaction, Si-O weak bond, and Si-O weak interaction at the interface.

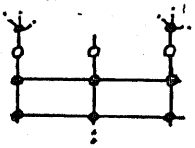
The reduction of the gap state density by H_2-annealing, trichrolo-ethylene annealing, or HCl oxidation is understood by bonding H or Cl to the $Si_3 \equiv Si-$ dangling bond at the interface.

These situations are illustrated in Fig. 5-5-1.

Fig. 5-5-1. Calculated results on Si-SiO interfaces.

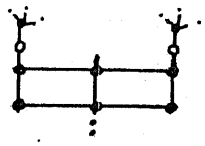
(1) (a) Perfect interface (b) oxygen dangling bond





No energy level exists in the Si bandgap.

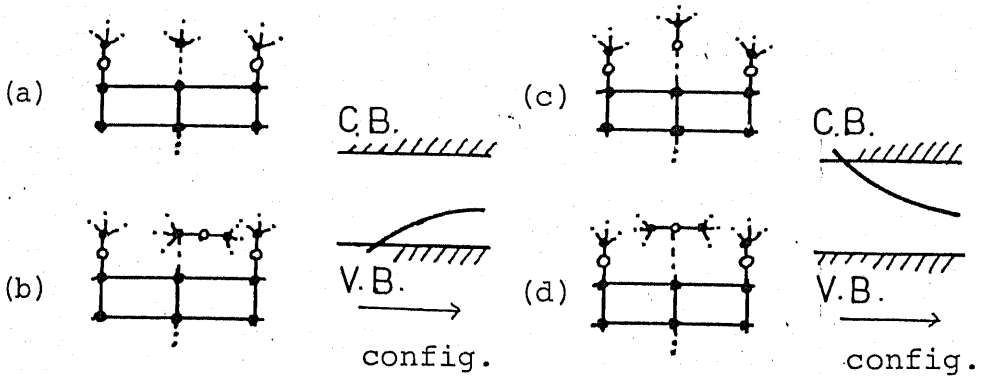
(2) Si dangling bond



A energy level appears about the Si midgap.

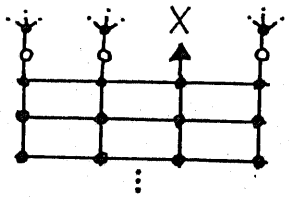
(3) (a) Si-Si bond (O-vacancy) (c) Si-O weak bond

(b) Si-Si weak interaction (d) Si-O weak interaction

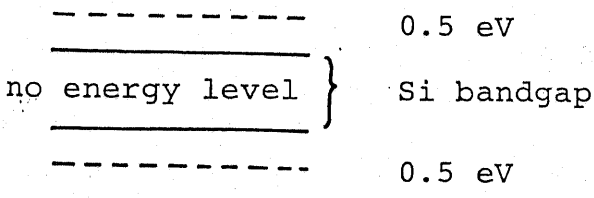


The energy level in the Si bandgap varies with the change of the configuration.

(4) Impurity at the $Si-SiO_2$ interface



X=any of H, OH, Cl, and F



No energy level exists in the energy range between 0.5 eV below the top of the valence band and 0.5 eV above the bottom of the conduction band of Si.

CHAPTER 6. BONDING DEFECTS IN SiO_2

SECTION 6-1. INTRODUCTORY REMARK [1~45]

Recently, as the sizes of MOS integrated circuits become very small to the order of sub-microns and accordingly the high doping density is employed, the electric field near the channel becomes very high. This high electric field acceralates the electrons and injects them into the SiO_2 layer. The injected electrons are then trapped in the SiO_2 bulk traps and change the Vth. This instability of MOS devices becomes a technological problem in making short channel MOS transistors. It is called the "hot electron injection problem", hot, because the injected electrons have enough energy to overcome the Si-SiO_2 electronic barrier higher than 3 eV and electrons having the energy of 3 eV are considered hot in a thermal equibrium.

The bulk traps of SiO_2 is thought neutral before they are negatively charged up by the injected electrons, because they do not affect the initial flat band voltage shift and in addition the capture cross sections of these traps are between 1.7×10^{-15} and 1.0×10^{-18} $[cm^2]$,[38] which are considered to be those of neutral traps.[103]

In addition to the problem concerning to MOS transistors, the elctron capture in the oxide gives a grave effect on the operation of FAMOS. In FAMOS devices, the avalanche voltage

increases as the write-erase cycle is repeated (so called "walk-out effect"). However, the microscopic origins of the neutral traps are not yet clear. This chapter is dedicated for obtaining the theoretical background of the neutral traps and shedding light on the electron injection problem. The calculation has been carried out assuming various atomic configurations of amorphous SiO_2 with or without bonding defects and impurities in Sec. 6-2. According to the results, some of the origins of the neutral traps are proposed in Sec. 6-3. Section 6-4 is devoted for the summary.

SECTION 6-2. MODELS FOR CALCULATION

The structure of amorphous SiO_2 is considered as a randomly connected SiO_2 network as shown in Photo 6-2-1. There is another kind of model called "micro-crystalline model", but in anyway the radial distribution function calculated by the "random network model" agrees excellently with experimental results. The Si-O-Si bond angle is thought to be varied between 120° and 180°, so that the calculations of this chapter is executed with varying the bond angle in this range. The Bethe-lattices mentioned in the previous chapter are used to represent the amorphous SiO_2. Two Bethe-lattices and an oxygen atom are connected all together with varying Si-O-Si bond angles to make a perfectly bonded amorphous SiO_2. In

order to simulate a Si-Si bond in SiO_2, two Bethe-lattices are interacted each other. Furthermore, impurities such as H, Cl, and F are bonded to a SiO_2 Bethe-lattice to calculate the effect of impurities in SiO_2. These models are illustrated in each figure of calculated results in the next section. The employed Hamiltonians are the same as those explained in Chap. 5.

SECTION 6-3. RESULTS AND DISCUSSIONS

Amorphous SiO_2 without Si dangling bonds and oxygen vacancies has no localized level in the SiO_2 bandgap, even if the Si-O-Si bond angle is varied in the range from 120° to 180°. In this perfectly bonded SiO_2 without bonding defects, the top of the valence band fluctuates about 0.4 eV by varying the Si-O-Si bond angle as shown in Fig. 6-3-1. This fluctuation may cause some effects on the hole injection from Si to SiO_2. Si dangling bonds, oxygen vacancies, and Si-Si bonds in SiO_2 produce levels in the SiO_2 bandgap, whose energy levels are given in Fig. 6-3-2. This dangling bond level changes its energy level about 1 eV[104] as shown in Fig. 6-3-3 according to the Si-O-Si bond angle variation, that is, by amorphous effect. The O-vacancy level also varies by this amorphous effect slightly, as shown in Fig. 6-3-4.

The Si dangling bond level and antibonding level of Si-Si bond

in SiO_2 can be the origins of the neutral traps. If the Si dangling bond or Si-Si bond exist in SiO_2 near the Si-SiO_2 interface, the electrons in these levels can go into the Si conduction band by tunneling and leaves positive charges, which can be one of the physical origins of the positive fixed charges commonly observed in the Si-SiO_2 system.[105]

The effects of the bonding of H, Cl, and F can not be predicted clearly because the results are dependent on the choice of K parameter of the EHT. If the value less than 1.0 is used for the K parameter, then the impurity level appears in the SiO_2 bandgap, whereas with the greater value the level dissapears. So that the situation is critical. But in anyway, the level moves upward to the conduction band of SiO_2. Since we thought that this effect can be detected by an avalanche injection and a photo-depopulation measurements, these experiments are carried out in Appendix B only to find that the sensitivity of the measurements is too low to say some definite conclusions.

SECTION 6-4. SUMMARY

Calculated results are summarized in Fig. 6-4-1. According to these results, some chemical origins of neutral traps in the thermally grown oxide might be a Si dangling bond level and an anti-bonding level of a Si-Si bond including an oxygen vacancy in the oxide.

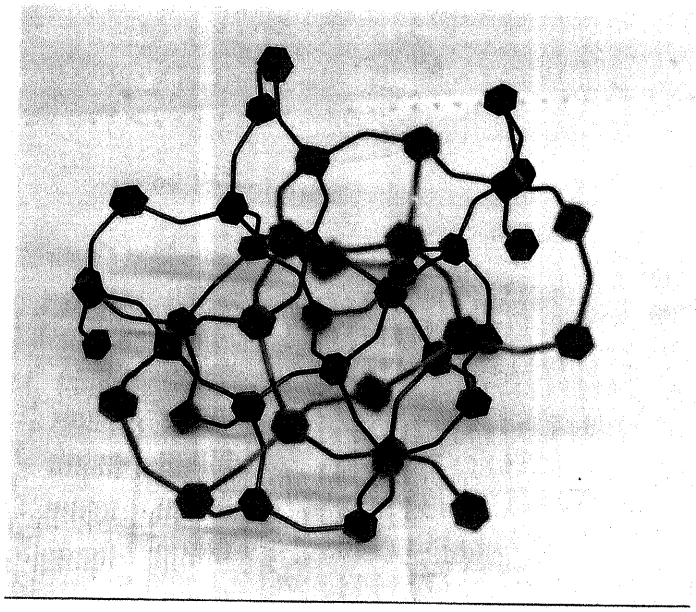


Photo. 6-2-1. Random network model of amorphous SiO_2. Black plastic balls are Si atoms and bended point of lines is oxygen atom. Si-O-Si angle is chosen to be 144° which is believed to be the peak value of the Si-O-Si angle distribution in the real a-SiO_2. It is easy to construct this kind of model without heavy stress.

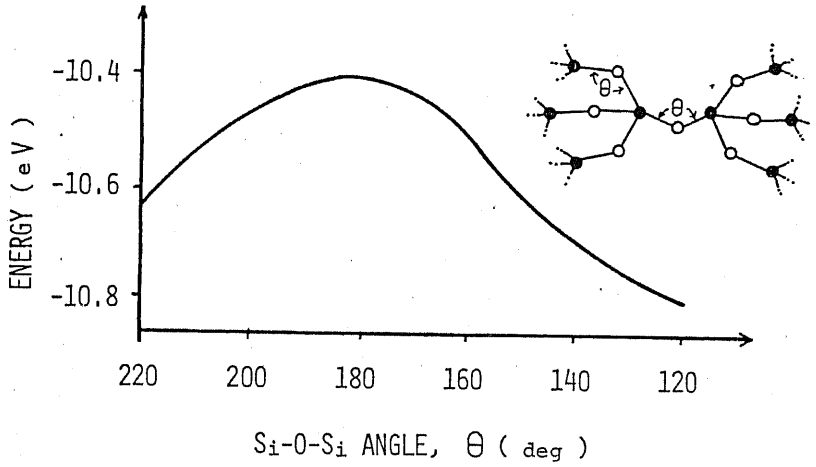


Fig. 6-3-1. Si-O-Si bond angle dependency of the top of the SiO_2 valence band. In the illustration open circles and closed circles denote O and Si atoms, respectively.

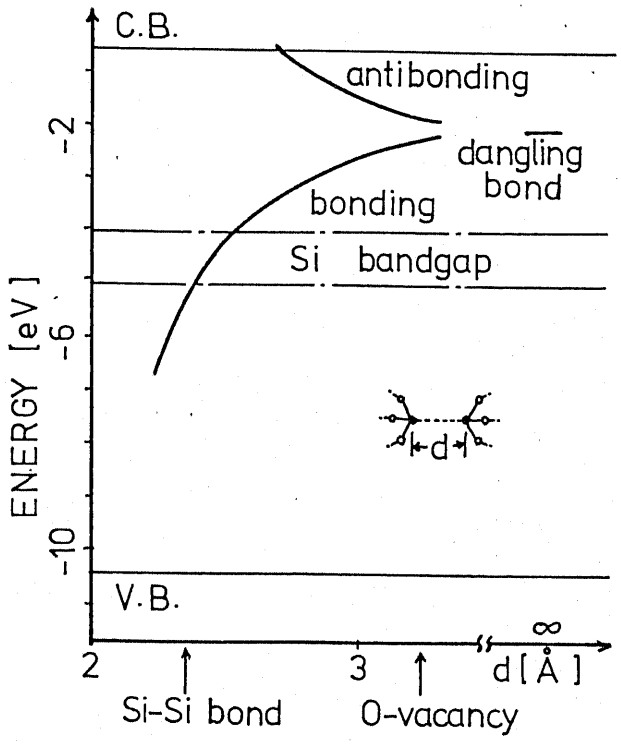


Fig. 6-3-2. Energy level of Si-Si bond in SiO_2 and its dependence on Si-Si bond length

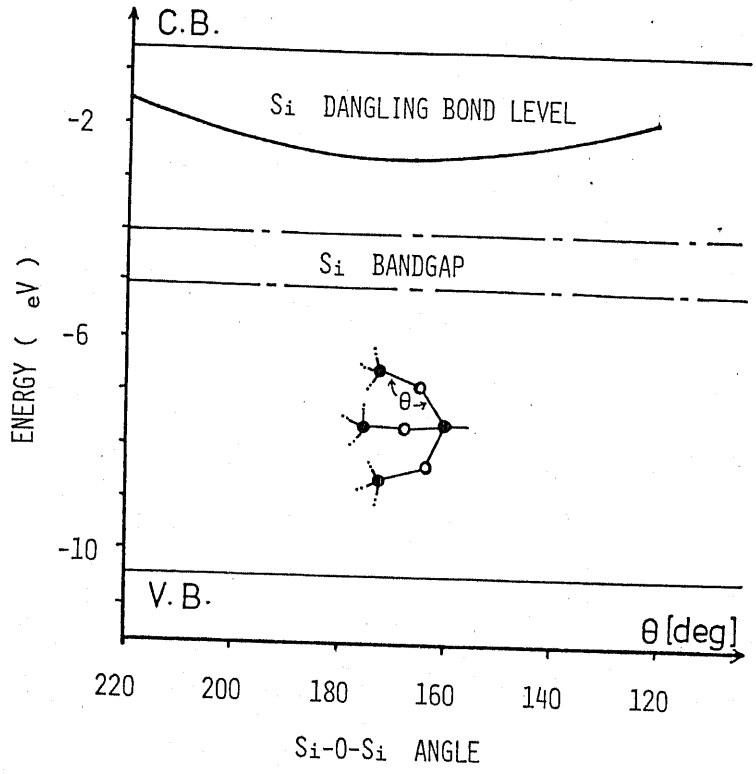


Fig. 6-3-3. Si-O-Si ANGLE DEPENDENCY OF Si DANGLING BOND LEVEL IN AMORPHOUS SiO_2

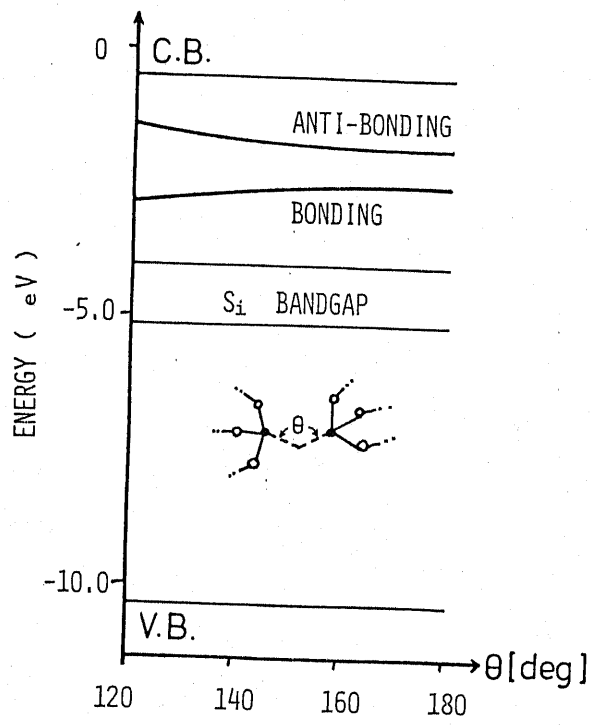
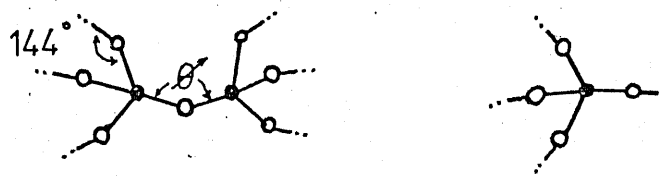


Fig. 6-3-4. O-VACANCY LEVEL IN SiO_2 AND ITS DEPENDENCE ON BOND BENDING

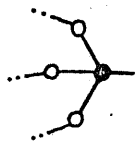
IN SiO_2

(1) (a) SiO_2 with no defects (b) oxygen dangling bond



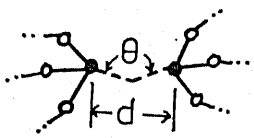
No energy level exists in the SiO_2 bandgap.

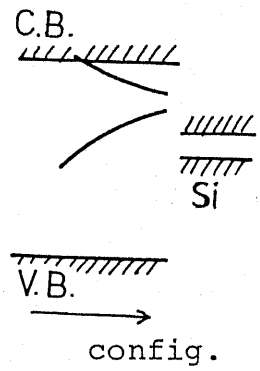
(2) Si dangling bond



A energy level exists in the SiO_2 bandgap.

(3) Si-Si bond (O-vacancy)





Two energy levels exist in the SiO_2 bandgap. These levels vary with changing d and θ.

Fig. 6-4-1. Summary of the calculated results on the amorphous SiO_2.

CHAPTER 7. CONCLUSIONS

A calculation procedure dealing with the electronic structure of crystalline Si-amorphous SiO_2 interface including microstructural defects is presented based on semiempirical tight-binding Hamiltonians and Green' function formulation and applied to the calculation of trap states at the interface between amorphous SiO_2 and the Si substrate with (111) orientation. This calculation method can be applied to the interfaces of materials other than Si-SiO_2 if the proper tight-binding parameters are given. The electronic structures of the amorphous SiO_2 with or without bonding defects are also calculated.

The following results are obtained. A perfect interface and the interface including oxygen dangling bond have no energy level in the Si bandgap, whereas $Si_3{\equiv}Si-$ dangling bond has a energy level at about the middle of the Si bandgap. Si-Si weak bond and Si-Si weak interaction at the interface give rise to gap states whose energy move in the energy range lower than the midgap with varying the distances between these two Si atoms. On the other hand, the energy levels of Si-O weak bond and Si-O weak interaction at the interface appear at the upper half of the forbidden gap of Si, depending on the distances between the Si atom and the O atom.

Posssible origins of interface trap states whose energy levels are distributed continuously in the Si bandgap are

suggested to be these Si-Si stretched bond, Si-Si weak interaction, Si-O stretched bond, and Si-O weak interaction at the interface.

The reduction of the interface trap state density by H_2 annealing, trichrolo-ethylene annealing, or HCl oxidation is understood by bonding H or Cl to the $Si_3{\equiv}Si-$ dangling bonds at the interface. According to the calculated results, the possibility of F annealing is also suspected.

As for the SiO_2 bulk traps, some of the origins of the neutral traps in the SiO_2 film are suggested to be the Si dangling bonds and Si-Si bonds in SiO_2.

These situations are illustrated in Fig. 7-1.

The developed method itself is simple and flexible and can be applied to crystalline free surfaces with or without relaxation and/or reconstruction, impurity chemisorped surfaces, Bethe-lattices, bulk impurity problems, defects, and other complex problems such as superlattices and multilayer problems, if the proper tight-binding parameters are given. If a dynamic matrix is used for Hamiltonian, phonon spectra and Raman spectra can also be calculated. It should be noted that the calculation method used in Chaps. 5 and 6 is mathematically exact except for the limitation caused by the practical numerical calculation such as the finite number of the sampling points in k space.

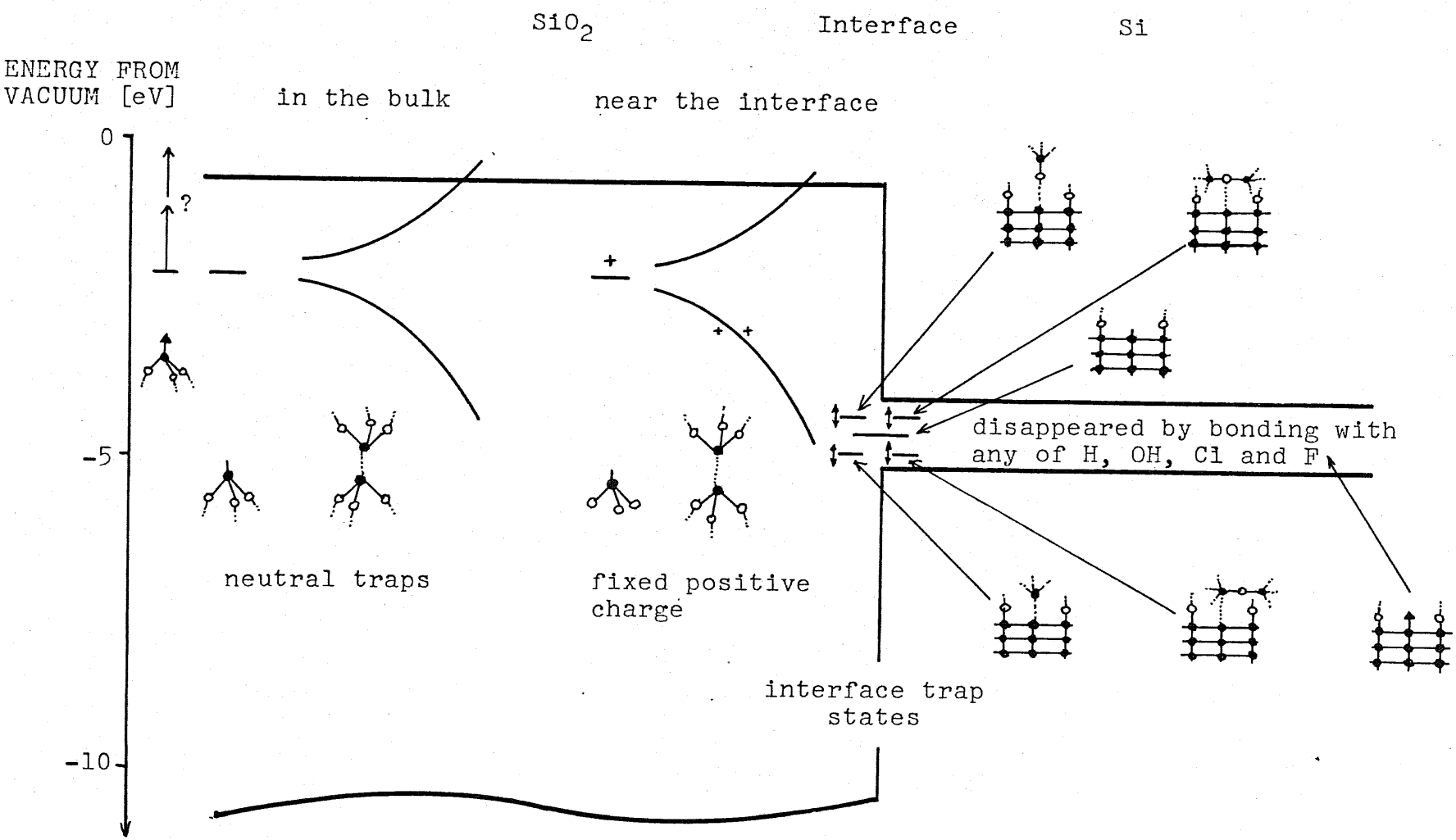


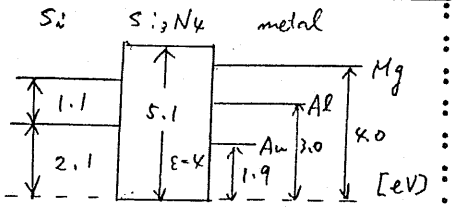
Fig. 7-1. Some of the possible origins of various gapstates of $Si-SiO_2$ system proposed in this work. Open and closed circles represent O and Si atoms, respectively. Solid triangles denote impurity atoms.

ACKNOWLEDGMENT

The author would like to express his profound appreciation to Prof. T. Sugano for his constant advices and encouragement in carrying out this work and to Prof. Y. Okabe for many useful suggestions and fruitful discussions. The author also wishes to give thanks to Dr. S. Sakai for giving him the theoretical background of the electronic structures of disordered materials and for introducing him the Green's function approach. The author is indebted to all staffs of Sugano and Okabe laboratories, especially to Dr. Asada for supporting him to make the electronic circuits, to Dr. F. Koshiga and Mr. M. Kitagawa for giving him the confortable atomosphere for the research, to Mr. C.F. Yeh and Mr. Ho Quoc Vu for their aid in the experiments, and to Mr. H. Matsumoto and Mr. T. Akiyama for stimulating the author with the intellectual problems.

REFERENCES

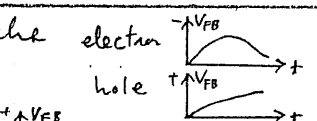
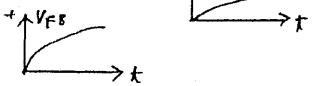
	AUTHOR	JOURNAL	SAMPLE	OBJECT	INJECTION/DETRAP	SIGNAL	CONTENTS/CONCLUSIONS
1	Goodman	Phys. Rev. 144 588 (1966)	n (111) 0.002 Ωcm 2~5μ SiO_2 1100°C 1μ gold 0.2cm²	barrier height of Al-SiO_2	1	Photo current	· mean free pass $W = \mu\tau E$ $\mu\tau \sim 10^{-13}$ m²/V
2	Deal Snow Mead	JPCS 27 1873 (1966)	P, n, (111), (100) 750μØ	barrier height of metal-SiO_2	"		barrier / work fn [eV]: Mg 2.25 / 3.70; Al 3.2 / 4.20; Ni 3.7 / 4.74; Cu 3.8 / 4.52; Au 4.1 / 4.70; Ag 4.15 / 4.31
3	Nicollian Berglund	JAP 41 3052 (1970)	P, 10^{16}~10^{17} cm⁻³ 500~2000 Å	theory of AC-avalanche	avalanche inj.	I_{DC}	· ω, F dependence of I_{DC} (theoretical, experimental) · details of setup
4	Powell Berglund	JAP 42 4390 (1971)	P, 1Ωcm, 2230 Å 1100°C dry, ~3000 Å		photo inj.	photo I-V ($I \sim 10^{-8}$ A) $h\omega < 5.4$ eV bias = +90 V	· minute theory of photo I-V · eq (7) is important · photo I-V changes in time. (Fig 5)
5	Nicollian Berglund Schmidt Andrews	JAP 42 5654 (1971)	P, 0.2Ωcm, 1500~2500 Å 250μØ, Cr-Au dot 80°C~200°C anneal	water-related center	avalanche inj.		· $\sigma \sim 1.5 \times 10^{-17}$ cm² · $E_A \sim 0.35$ eV (annealed in H_2O) · emits H when captures electron. · electro-chemical reaction
6	Thomas Feigl	J. Phys. Chem. Sol. 33 2197 (1972)	n, Si (111) 0.01Ωcm, 1~4μm wet, 2×10 mm	2.1 eV ± 0.5 trap (by Na^+?)	photo inj. photo detrap. 60 W dentrium lamp 80°K~300°K, <10^{-5} torr, in inert gas	photo Q-V ($\sim 10^{-12}$A) · sweep λ (6.3mm/sec) · photo I(t),	· photo Q-V measurement was difficult · $\frac{n_T}{L} = 0.6 \times 10^{14}$; 2.6×10^{15} cm⁻² · $\frac{\bar{x}}{L} = 0.62, 0.7$ · $\Delta E = 0.5$ eV
7	Yun	APL 23 152 (1973), APL 25 340 (1974)	n-Si, 2Ωcm MNOS	$\bar{x}$ location Technique		ΔV_{FB} & $\int I\,dt$	· $Q = \int I\,dt$ if the capacitance is constant · if there is no conduction current, $\bar{x}$ and n_T are separable
8	Dimaria Feigl	Phys. Rev. B9 1874 (1974)			photo detrap.	photo Q-V	· minute theory of photo Q-V

	AUTHOR	JOURNAL	SAMPLE	OBJECT	INJECTION/DETRAP	SIGNAL	CONTENTS/CONCLUSIONS
9	Dimaria	JAP 45 5454 (1974)	p-Si 0.001Ωcm (100), ~500Å Al_2O_3	Barrier height of metal and Al_2O_3		photo current ($\sim 10^{-12}$[A])	· full setup for photo I-V · neglect light interference · $Y = I h\nu / ST$ (photoelectrons/photon) · Al, Au, Ni, Mg 4.1 3.7 3.2 2.5 [eV]
10	Ning Yu	JAP 45 5373 (1974)	n-ch IGFET 0.5~2 Ωcm 350~1500 Å 2×40 mil	electron trap	photo inj. (non-avalanche inj.) $10^{-4} \sim 10^{-9}$ A/cm² visible light	$V_T(t)$ $\eta_{eff} = \frac{d(C_{ox} \Delta V_T)}{dt} j_g^{-1}$ $= \sum (N_{T,i})_{eff} \sigma_i e^{-\sigma_i N_{inj}}$	$\sigma = 3.3 \times 10^{-13}$ cm² 2.4×10^{-19} cm²
11	Dimaria Kerr	APL 27 505 (1975)	poly-Si	why SiO_2 on poly-Si leaky?		dark current photo current (I-V)	· barrier height of poly Si - SiO_2 = " cry-Si - SiO_2 · leakage is due to the roughness of the interface and enhancement of the field
12	Dimaria Arnett	APL 26 711 (1975)	MNS, p, n, CVD Si_3N_4 815°C SiO_2 ~ 15 Å	barrier height		photo current 2.0 ~ 5.5 eV 14 eV gate bias by hole injection	
13	Arnett Dimaria	APL 27 34 (1975)	MNS	dark current			· dark current dependence on contact metal and insulator thickness
14	Dimaria Feigl Butler	Phys. Rev. B11 5023 (1975)	1~4 μm 0.001~5 Ωcm [Au, Ni, Al]	2.4 eV trap in SiO_2	photo inj. photo detrap field detrap deuterium lamp	photo I-V (5 eV) (< 1 A/mw·cm²) photo Q-V $\frac{1}{\tau} = n v_T \sigma_e + S\sigma_p + n v_T \alpha$	· $\sigma_T \sim 10^{-18}$ cm² · $\sigma_e \sim 10^{-14}$ cm² · table of light intensity · $\bar{x} = 0.5$ · theory · 1% of traps are photo-accessible
15	Dimaria	JAP 47 4073 (1976)	MOWOS	electron trap	5 eV photo emission ±15 V gate bias at 300°K, air	photo I-V (5 eV)	· neglect of detrap · $\Delta V_g' = \Delta V_{FB}$

	AUTHOR	JOURNAL	SAMPLE	OBJECT	INJECTION/DETRAP	SIGNAL	CONTENTS/CONCLUSIONS
16	Ma Yun Dimaria Scoggan	JAP 47 1599 (1976)	MNOS	EB- irradiation		C-V (interface trap) dark current charge-centroid method (Yun)	· thick films N_{ss} ↗ · thin films N_{ss} constant
17	Dimaria Aitken Young	JAP 47 2740 (1976)	P-Si, 0.1Ωcm 1500Å dry forming gas 400°C, 20min	Na^+ trap (∓2.4eV trap)	avalanche inj. 500KHz square wave. 77°K $J_{DC} = 10^{-3} \sim 10^{-9}$ [A/cm²]	ΔV_{FB} $\left(= \frac{q\bar{x}}{\varepsilon\tau} Q \exp\left(-\frac{t}{\tau}\right)\right)$ $\tau = \frac{\varepsilon}{J\sigma}$	· no field detrap at 77°K · $\sigma = 2\times10^{-15}$, 2×10^{-19}, 5×10^{-20} [cm²]
18	Aitken Dimaria Young	IE³ NS-23 1526 (1976)	500~1000Å dry at 1000°C photo. p.m 2Ωcm ava. p 0.1~0.2Ωcm Al gate 125Å poly Si gate	EB, X-ray induced trap	avalanche inj. photo inj.	photo I-V ΔV_{FB}	· Induced positive charge is localized in { 50Å from Si-SiO_2 interface { 50Å from Al-SiO_2 interface
19	Feigl Butler Dimaria Kapoor	Thermal & Photostimulated Currents in Insulators (1976)		Review on photo Q-V		photo Q-V	· absorption & photo lumi. are complex. · $W(E, E_i) \propto \lvert \hat{\rho} \cdot \vec{r}_{if} \rvert^2 g(E_i + E)$
20	Butler Feigl Ota Dimaria	〃	n. P doped-Si (100), ~5Ωcm 1μm + Na	Na related trap (wet)	photo inj. photo detrap (4500Å)	ΔV_{FB}	· There is a difference of the several orders between doped Na density and electrically active density
21	Aitken Young	JAP 47 1196 (1976)	P, 0.1Ωcm 500Å SiO_2 dry 400°C, 20 min in forming gas	X-ray induced trap	avalanche inj 100~500 KHz 77°K & 300°K	ΔV_{FB}	· 5 traps whose σ are ranging $10^{-13} \sim 10^{-19}$ [cm²].
22	Ning	JAP 47 1079 (1976)	p-ch IGFET (100), 0.1Ωcm 1000Å	hole trap	〃	$V_T(t)$ $N_{inj}(t) = C_{ox} \cdot \Delta V_T(t)/q + \int_0^t j_G(t')/q\, dt'$	· $\eta_{eff} \sim 99\%$ · $\sigma = 3.1\times10^{-15}$ [cm²] $n_T = 1.4\times10^{18}$ [cm⁻³] · neglect of detrapp

	AUTHOR	JOURNAL	SAMPLE	OBJECT	INJECTION/DETRAP	SIGNAL	CONTENTS/CONCLUSIONS
23	Jones Embree	JAP 47 5365 (1976)	quartz a-SiO_2	O-vacancy		photo luminescence	quartz: lumi. 4.77 (2600Å), absorption 6.2 (2000Å) a-SiO_2: 4.28 (2900Å), 5.9 (2100Å) signal is due to O-vacancy. because decrease with O_2 anneal
24	Dimaria Young Ormond	APL 31 680 (1977)	MOWOS	effect of W layer		photo I-V ($\rightarrow \bar{x}$)	leakage current, breakdown voltage } are improved by W. Due to uniformization of trap density.
25	Young Dimaria Bojarczuk	JAP 48 3425 (1977)	W implanted (50keV) & evaporated	W in SiO_2	avalanche-inj. Thermal detrap 275 ~ 350°C	δV_{FB} photo I-V ($\rightarrow \bar{x}$)	· $\sigma = 4.56 \times 10^{-14} \sim 1 \times 10^{-14}$ [cm²] (clustering ?) · $E_T = 0.6$ [eV] thermal photo inaccessible
26	Dimaria Weinberg Aitken Young	J. Elec. Mat. 6, 207 (1977)	MOS MOWOS	hole & electron trap	photo inj.	photo I-V ($<10^{-13}$[A]) $\delta V_{FB} \sim 10$ [V]	· Review of photo I-V · trap center is equal to W location
27	Dimaria Weinberg Aitken	JAP 48 898 (1977)	P, n Si 2Ωcm (100) ~1000Å dry Al ~100Å 0.5×0.5mm 400°C, 20min forming	hole traps in SiO_2	Ne-discharge (VUV) 20keV Xray by W high field	photo I-V (5eV) C-V	· no bulk hole trap in SiO_2 hole traps are localized near interface (50Å) · theory of photo I-V
28	Young Dimaria Hunter	J. Elec. Mat. 6 569 (1977)	p-Si, 0.1~0.2Ωcm 700Å at 1000°C dry	Al trap in SiO_2	avalanche-inj. 500kHz, square wave	δV_{FB}	· setup of avalanche inj. · anneal T ↗ trap ↘ · comparison between $\bar{x}$ & LSS theory
29	Ning Osburn Yu	(1977)	n-ch IGFET SiO_2, Si_3N_4	hot electron trapping	avalanche inj.	V_T transconductance	· Si_3N_4 V_{FD} · SiO_2 V_{FB} & transconductance
30	Ning Osburn Yu	JAP 48 289 (1977)	poly-Si gate IGFET	theory of injection	non-avalanche inj.	gate current	· Lucky electron model · Prob. $= 2.9\, e^{-\frac{\phi d}{\lambda}}$ $\lambda = .108 \tanh(0.63/kT)$ $\phi V(d) = 3.1 - \beta E_{ox}^{\frac{1}{2}} - \alpha E_{ox}^{\frac{1}{3}}$

	AUTHOR	JOURNAL	SAMPLE	OBJECT	INJECTION/DETRAP	SIGNAL	CONTENTS/CONCLUSIONS
31	Dimaria	JAP 48 5149 (1978)	P-Si 0.1 Ωcm (100) Al ~130Å Na^+ dope by NaCl	location of Na^+ ion		photo I-V	• Na^+ is localized near Si-SiO_2 interface (50Å)
32	Dimaria Dekeersmaecker Young	IBM. Res. Rep. #29814 1/24/78 (1978)	Light Activated (As or P implanted MOS)	Storage Device			• Write and Erase by light • no damage by avalanche • volume type memory
33	Dimaria	IBM. Res. Rep. #31120 7/21/78 (1978)	Al, P, As, B implanted MOS		avalanche inj. thermal detrap photo detrap 77°K ~ 300°K	"	• σ_{ph} is const. on T • no trap by B • no field detrap (~3 MV/cm) • thermal detrap in N_2 100~350°C
34	Dekeersmaecker Dimaria Pantelides	IBM. Res. Rep. #30337 4/21/78 (1978)	P-Si (100) 0.1 ~ 0.2 Ωcm 1000°C dry 560 ~ 1430Å Al ~ 100Å 0.52 mm^2	As, P implanted	avalanche-inj. 50 kHz ramp. photo detrap λ-scan step by step	ΔV_{FB} (1 MHz) $\frac{\Delta V_{FB}(0) - \Delta V_{FB}(t)}{\Delta V_{FB}(0)} = t\gamma(h\omega) \cdot \Sigma(h\omega)$	• $\sigma_I \sim 3 \times 10^{-17}$, $\sigma_{As} \sim 10^{-15}$ cm^2 $E_T \sim 4$ [eV]
35	Dimaria Young Dekeersmaecker Hunter Serrano	JAP 49 5441 (1978)	P-Si (100) 1300 ~ 1400 Å 0.1 ~ 0.2 Ωcm	B, Al, P, As implanted (5~100 KeV)	avalanche inj	photo I-V	• comparison with LSS
36	Dimaria Young Hunter Serrano	IBM J. Res. Dev. 22 285 (1978)	"	Al implanted	avalanche inj 500 kHz sq. 4 hours	ΔV_{FB}	• 400 ~ 600 points by computor • $\bar{x} = 0.59$ σ cm^2 : N_t cm^{-2} 1.6×10^{-15} : 4.6×10^{11} 1.26×10^{-16} : 1.14×10^{12} 1.40×10^{-17} : 1.4×10^{12} 1.26×10^{-18} : 5×10^{11}
37	"	ibid. 289 (1978)	"	"	"	photo I-V	• without Al → no trapping • good agreement with $\bar{x}$ and SIMS • certain portion is active trap.

	AUTHOR	JOURNAL	SAMPLE	OBJECT	INJECTION/DETRAP	SIGNAL	CONTENTS/CONCLUSIONS
38	Aitken Young Pan	JAP 49 3386 (1978)	P 0.2 Ωcm (100) 440Å dry 1000°C	EB 25KeV induced trap	avalanche-inj 4×10^{-8} A° 2×10^{4} sec 77°K, 295°K	ΔV_{FB}	• neutral trap can not be annealed by low temp. anneal. • trap density decrease at 500°C • $\sigma \sim 10^{-15} \sim 10^{-18}$ cm²
39	Ning	Sol. St. Elec. 21 273 (1978)					• formula for emission probability • hot carrier injection phenomenology
40	Ning	JAP 49 4077 (1978)	n-ch poly-Si gate IGFET (100) 0.5Ωcm 500Å 1000°C HCl or wet oxidation	traps induced by EB deposited Al	non-avalanche inj.		• X ray induced by EB produces traps. • $\sigma \sim 10^{-13}$ cm² positive trap $\sigma \sim 10^{-15}$ cm² neutral trap • annealed in forming gas • $\sigma_{pos} \propto E_{ox}^{-3} \cdot T^{\alpha}$ $\sigma_{neu} \propto e^{-bE_{ox}}$
41	Ning	JAP 49 5997 (1978)	〃	forward bias pulsed injection in dark atm.	TSR 77°K ~300°K	ΔV_{FB}	• peak 300mV ± 50mV • 90% is shallow level • X ray induced trap is deep.
42	Dimaria	JAP 50 5826 (1979)	proposal of GI-MIS, SI-MIS				• low voltage • high reliability • no degradation of Si-SiO₂ int. • electrically "erase" & "write"
43	Dimaria Ephrath Young	JAP 50 4015 (1979)	RIE CF_4, O_2, Ar	Reactive Ion etching		C-V photo I-V	• Traps can be annealed out • charging is done by plasma induced photon
44	Weinberg Young Dimaria Rubloff	JAP 50 5757 (1979)	P, n-Si (100) 0.1~0.2 Ωcm 1050Å, 1.25mmØ Al ~150Å	water effect on MOS	avalanche inj. photo inj. (10eV)	ΔV_{FB} $\Delta V_{FB}(t)$, $\Delta V_{FB}(\lambda)$ photo I-V	• avalanche electron / hole  • VUV 
45	Ning	IE³ ED-26 546 (1979)		Voltage limitation in VLSI			• $V_{DS} - V_{GS} \leq 4.75$ V (25°C) 〃 ≤ 3.5 V (77°K)

	AUTHOR	JOURNAL	METHOD	MATERIAL	RESULTS	FIT WITH EXP.
46	Schneider Fowler	Phys. Rev. Let 25, 425 (1976)	mixed-basis method	β-cristobalite valence, conduction band	Band structure	○
47	Pantelides Harrison	Phys. Rev. B13 2667 (1976)	linear combination of bond orbitals	β-cristobalite, GeO_2 valence band	Band structure DOS	○ UV, UPS XPS, XES
48	Ciraci Batra	Phys. Rev. B15, 4923 (1977) The Phys. of SiO_2 & its interface 65	extended tight- binding method basis = GTO, STO, sp^3	β-cristobalite α-quartz valence, conduction band	Band structure DOS E_g = 9.8 eV	○ UPS XPS SXS
49	Chelikowsky Schlüter	Phys. Rev. B15 4020 (1977)	self-consistent pseudopotential	α-quartz valence, conduction band	Band structure DOS charge distribution E_g = 9.2 eV indirect	○ excellent agreement
50	Calabrese Fowler	Phys. Rev. B18 2888 (1978)	mixed-basis method	α-quartz valence, conduction band	Band structure DOS similar to β-christobalite	○
51	Calabrese Fowler Schneider	The Phys. of SiO_2 & its interface 70 (1978)	tight-binding fit	α-quartz β-cristobalite α-trimidite valence, conduction band	Band structure DOS	
52	Nucho Madhukar	ibid. 60 (1978)	simple tight- binding	α-quartz valence, conduction band	Band structure DOS	○
53	Chadi Laughlin Joannopoulos	ibid. 55 (1978)	"	α-quartz β-cristobalite Bethe-lattice	Band structure (α-quartz) DOS	

54. M.Pepper & N.F. Mott, Phys. of Semicond., Proc. 13th Int'l Conf. ,Rome, 762 (1976).

55. K.C. Pandey and J.C. Phillips, Phys. Rev. B13, 750 (1976), Solid State Commun. 14, 459 (1974).

56. G. Dresselhaus & M.S. Dresselhaus, Phys. Rev. 160, 649 (1967).

57. D.J. Chadi, Phys. Rev. B16, 790 (1977).

58. M. Schluter, J.R. Chelikowsky, S.G. Louie, & M.L. Cohen, Phys. Rev. B12, 4200 (1975).

59. S.M. Sze, Physics of Semiconductor devices, Wiley-Interscience, New York (1969).

60. R. Hoffmann, J. Chem. Phys. 39, 1393 (1963).

61. J.A. Pople & D.L. Beveridge, Approximate Molecular Orbital Theory, McGraw-Hill (1970).

62. R.S. Mulliken, C.A. Rieke, D. Orloff, & H. Orloff, J. Chem. Phys. 17, 1248 (1949).

63. G. Blyholder & C.A. Coulson, Theoret. Chim. Acta (Berl.) 10, 316 (1968).

64. R.S. Mulliken, J. Chim. Physique 46, 497,675 (1949).

65. I. Chen, Phys. Rev. B11, 3676 (1975).

66. N.P. Il'in & V.F. Masterov, Soviet Phys. Semicond. 11, 874 (1977).

67. Messmer & Watkins, Phys. Rev. Lett. 25, 656 (1970).

68. T. Shimizu & N. Ishii, Report of Eng. Dept. of Kanazawa Univ., 11, 11 (1977).

69. Griscom, J. non-crystalline solids, 14, 155 (1977).

70. R.P. Messmer, Chem. Phys. Lett., 11, 589 (1971).

71. A.J. Bennett & L.M. Roth, J. Phys. Chem. Silids, 32, 1251 (1971).

72. P.O. Löwdin, J. Chem. Phys., 18, 365 (1950).

73. K. Hirabayashi, J. Phys. Soc. Japan, 27, 1475 (1969).

74. J. Pollman & S.T. Pantelides, Phys. Rev., B18, 5524 (1978).

75. T. Iizuka, Master Thesis at Elec. Dept. of the Univ. of Tokyo, (1972).

76. R.B. Laughlin & J.D. Joannopoulos, Phys. Rev. B17, 2790 (1978).

77. M.F. Thorpe & D. Weaire, Phys. Rev. B4, 3518 (1971).

78. Goto, presented at the solid electronics meeting held at elec. dept. of the Univ. of Tokyo (1978).

79. R. Haydock & M.J. Kelly, Surf. Sci., 38, 139 (1975).

80. D.J. Chadi, R.B. Laughlin and J.D. Joannopoulos, The physics of SiO_2 and its interfaces, Pergamon Press, 55 (1978).

81. R.B. Laughlin, J.D. Joannopoulos and D.J. Chadi, ibid., 321 (1978).

82. R.B. Laughlin, J.D. Joannopoulos, and D.J. Chadi, Phys. Rev. B21, 5733 (1980).

83. J.D. Joannopoulos and F. Yndurain, Phys. Rev. B10, 5164(1974).

84. F.J. Grunthaner and J. Maserjian, ibid., 389 (1978).

85. C.R. Helms, Y.E. Strausser, and W.E. Spicer, Appl. Phys. Let. 33, 767 (1978).

86. W.T. Pantelides and M.Long, The Physics of SiO_2 and its interface, Pergamon Press, 339 (1978).

87. A. Koma, private communication.

88. R.L. Mozzi and B.E. Warren, J. Appl. Cryst. 2, 164 (1969).

89. H.R. Philipp, Sol. State Commun. 4, 73 (1966).

90. Ioffe & Regal, Progr. Semicond. 4, 239 (1960).

91. Yonezawa & watabe, Phys. Rev. B11, 4746,4753 (1975).

92. McGill & Klima, Phys. Rev. B5, 1517 (1972).

93. Keller & ziman, J. Non-cryst. Solids 8-10, 111 (1972).

94. M. Okazaki et al, J. Phys. Soc. Japan 39, 780,787 (1975).
95. Joannopoulos & Cohen, Phys. Rev. B7, 2644 (1973).
96. Ching & Lin, Phys. Rev. Lett. 34, 1223 (1975).
97. Donovan & Spicer, Phys. Rev. Lett. 21, 1572 (1968).
98. B. Kramer, Phys. Stat. Solidi 41, 649,725 (1970).
99. I. Chen, Phys. Rev. B7, 3672 (1973), B8, 1440 (1973).
100. T. Shimizu et al, Phys stat. Solidi (b) 74, K39 (1976).
101. J.D. Joannopoulos et al, Solid State Physics 31, 71, Academic Press (1976).
102. H. Sakaki, K. Hoh & T.Sugano, IEEE ED-17, 892 (1970).
103. T. Sugano, Semiconductor Physics and Devices(I), p286, Shokodo in Japan (1973).
104. T.H. Ning, J. Appl. Phys. 49, 5997 (1979).
105. C.M. Svensson, The physics of SiO_2 and its interfaces, Pergamon Press, 328 (1978).

APPENDIX A

INDEX FOR PROGRAMS

PROGRAM NAME	PAGE	CONTENTS
		at the $Si-SiO_2$ interface. Input $\dot{q}^{\gamma}_{\gamma\gamma}$ and $\dot{q}^{D}_{ss}$ from file 23 and 26. Output $\dot{q}^{STR}_{ss}$.
DSIG	A1-107	Data for SIGEEO.
WEAK	A1-108	Calculate energy levels for weak Si-O interaction at the $Si-SiO_2$ interface. Input $\dot{q}^{D}_{ss}$ and $\dot{q}^{O}_{oo}$ from file 26 and 28.
DWEAK	A1-111	Data for WEAK.
WEAKS	A1-112	Calculate energy levels for weak Si-Si interaction at the $Si-SiO_2$ interface. Input $\dot{q}^{D}_{ss}$ and $\dot{q}^{o}_{ss}$ from file 26 and 29.
DWEAKS	A1-114	Data for WEAKS.

ER SBFIT OF A1402.S.FORT

```
00000100       FUNCTION HH(ANK,LCI,MCI,ANL,LCJ,R,NANK,NANL)
00000200       INTEGER ANK,ANL
00000300       EQUIVALENCE (P(1),SESS),(P(2),SESP),
00000400      *            (P(3),SVSS),(P(4),SVSP),(P(5),SVPPS),(P(6),SVPPP),
00000500      *            (P(7),SUSS),(P(8),SUSP),(P(9),SUPPS),(P(10),SUPPP)
00000600       COMMON /COMP/P(10)
00000700       ANK=ANK;ANL=ANL;NANK=NANK;NANL=NANL
00000800       SVPS=SVSP ;SUPS=SUSP
00000900 C------SILICON------
00001000   200 CONTINUE
00001100       IF(ABS(R).LT.0.01) GO TO 220
00001200       IF(ABS(R).GT.5.) GO TO 210
00001300 C------1ST NEAREAT------
00001400   400 CONTINUE
00001500       EXPBR=1.
00001600       IF(LCI.EQ.0.AND.LCJ.EQ.0) GO TO 201
00001700       IF(LCI.EQ.0.AND.LCJ.EQ.1) GO TO 202
00001800       IF(LCI.EQ.1.AND.LCJ.EQ.0) GO TO 203
00001900       IF(MCI.EQ.0) GO TO 204
00002000       IF(MCI.EQ.1) GO TO 205
00002100       STOP 0008
00002200   201 HH=EXPBR*SVSS ;GO TO 1
00002300   202 HH=EXPBR*SVSP ;GO TO 1
00002400   203 HH=EXPBR*(-SVPS) ;GO TO 1
00002500   204 HH=EXPBR*SVPPS ;GO TO 1
00002600   205 HH=EXPBR*SVPPP ;GO TO 1
00002700 C------2ND NEAREST------
00002800   210 CONTINUE
00002900       IF(LCI.EQ.0.AND.LCJ.EQ.0) GO TO 211
00003000       IF(LCI.EQ.0.AND.LCJ.EQ.1) GO TO 212
00003100       IF(LCI.EQ.1.AND.LCJ.EQ.0) GO TO 213
00003200       IF(MCI.EQ.0) GO TO 214
00003300       IF(MCI.EQ.1) GO TO 215
00003400       STOP 0009
00003500   211 HH=SUSS ;GO TO 1
00003600   212 HH=SUSP ;GO TO 1
00003700   213 HH=-SUPS ;GO TO 1
00003800   214 HH=SUPPS ;GO TO 1
00003900   215 HH=SUPPP ;GO TO 1
00004000   220 IF(LCI.EQ.0) GO TO 221
00004100       IF(LCI.EQ.1) GO TO 222
00004200       STOP 0013
00004300   221 HH=SESS ;GO TO 1
00004400   222 HH=SESP ;GO TO 1
00004500     1 CONTINUE
00004600 C     WRITE(6,101) HH
00004700 C 101 FORMAT(' HH=',F10.3)
00004800       RETURN ;END
00004900       PROGRAM SBFIT
00005000       COMMON/COM03/ V(40),W(20,40)
00005100       COMMON/VECK/VECK(3)
00005200       COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
00005300       COMMON/NATOMS/ NATOMS,N,NK
00005400       COMMON/INFO/ AN(40),NOUT
00005500       COMMON/INFO1/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
00005600       COMMON /CON/CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX,CSS,CSX,CXS,CX)
00005700       COMMON/VOIP/ VOIP(4,10),AZETA(4,10),CONST,OM1,OM2,VP(4,10),
00005800      *             AZP(4,10)
```

```
00005900       COMMON /POS/POS(3,10),BL,AU,TH,PAI,ROOT2,ROOT3
00006000       COMMON /H/H(40,40)
00006100       COMMON /NC/NC(18),LC(10),MC(10)
00006200       COMMON /COM01/ ZS(40,40),ZH(40,40)
00006300       COMMON /COM02/ PM(40,40),RM(40,40)
00006400       COMMON /NAN/NAN(40)
00006500       COMMON /COMSF/FAC(12,40),OLDEP(6),IP1
00006600       INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
00006700       REAL*8 V
00006800       COMPLEX*16 ZS,ZH,PM,RM
00006900       COMPLEX*8 ZTHETA,ZEXP
00007000 C      DIMENSION SD(2,2,12),ST(13),DELV(8),SROT(12,12)
00007100       DIMENSION F(3,12),HD(16,16,12)
00007200       DIMENSION AKM(3,7),ROT(12,12),ROTD(4,4),ROTD1(4,4)
00007300      *,         HROT(12,12),RT(13)
00007400       DATA ROT/144*0./
00007500       DATA AKM/5*0.,0.6122,3*0.3061,0.,0.,0.4592,
00007510      *         2*0.459,0.,2*0.,0.48,2*0.,0.5/
00007600       DATA AU/.529167/,PAI/3.141592/,BL/2.35/,A/5.43/
00007700       DATA AN/14,14,14,14/,NATOMS/4/
00007800       DATA F/0.,1.,1.,1.,0.,1.,1.,1.,0.,
00007900      *       0.,-1.,-1.,-1.,0.,-1.,-1.,-1.,0.,
00008000      *       0.,1.,-1.,-1.,0.,1.,1.,-1.,0.,
00008100      *       0.,-1.,1.,1.,0.,-1.,-1.,1.,0./
00008200       DATA ROTD/1.,1.,1.,1.,1.,1.,-1.,-1.,
00008300      *          1.,-1.,1.,-1.,1.,-1.,-1.,1./
00008400       DATA TH/109.5/
00008500       DATA ROTD1/1.,-1.,-1.,-1.,1.,-1.,1.,1.,
00008600      *       1.,1.,-1.,1.,1.,1.,1.,-1./
00008700       DATA NAN/5,5,5,5/
00008800       REAL*8 WW(40,7)
00008900       DIMENSION DP(10),LW(40),OLDV(8,7)
00009000       COMMON /COMP/P(10)
00009100 C------CJ NEW PARAM NNN----
00009200 C      DATA P/-4.2,1.7,-2.075,2.7583,3.129,-0.9212,4*0./
00009300 C------PP PARAM----
00009400 C      DATA P/-4.203,0.187,-2.08,2.12,2.32,-0.52,2*0.,0.58,-0.1/
00009500 C------CJ 2 PARAM------
00009600 C      DATA P/-4.4,1.8,-2.03,2.546,4.188,-1.452,3*0.,-0.365/
00009610 C------PRESENT------
00009620 C      DATA P/-5.22,0.83,-2.05,2.09,2.33,-0.54,0.084,-0.29,.47,-.13/
00009630 C------TEST1------
00009640 C      DATA P/-5.658,1.178,-2.053,2.087,2.426,-.647,.106,-.539,.379,-
00009650 C      DIMENSION TESTD(11)
00009660 C      DATA TESTD/.509,.043,-.016,.056,-.156,.082,-.021,.18,.064,.0/
00009700       DATA DP/6*0.1,4*0.01/
00009800       EQUIVALENCE (R1,RT(1))
00009900       COMMON /R/R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13
00010000      *       ,AK1,AK2
00010010 C------PARAM FROM FILE 30-----
00010020 C      READ(30) (P(I),I=1,10)
00010030       READ(5,301) (P(I),I=1,10)
00010040   301 FORMAT(F10.0)
00010100       DO 21 J=1,12
00010200       DO 21 I=1,3
00010300    21 F(I,J)=F(I,J)*A/2./AU
00010400 C------MAKE POS------
00010500       DO 30 I=1,3
00010600       POS(J,1)=0. ;POS(I,2)=A/4./AU
00010700    30 CONTINUE
00010800       DO 1000 IPARAM=1,11
00010900         IP1=IPARAM-1 ;IP2=IPARAM-2
00011000         IF(IP1.EQ.0) GO TO 1001
00011100         IF(IP1.EQ.1) GO TO 1002
00011200         P(IP1)=P(IP1)+DP(IP1)
00011300         P(IP2)=P(IP2)-DP(IP2)
```

```
00011400           GO TO 1001
00011500  1002     P(1)=P(1)+DP(1)
00011600           GO TO 1001
00011700  1001     CONTINUE
00011800         CALL RWRIT(P,'PARAM ',10,10)
00011900 C------BULK PROBLEM------
00012000 C------MAKE SD AND HS-----
00012100         DO 40 IF=1,12
00012200         DO 41 I=1,3
00012300         POS(I,3)=F(I,IF)
00012400         POS(I,4)=F(I,IF)+A/4./AU
00012500     41  CONTINUE
00012600         DO 46 J=1,16
00012700         DO 46 I=1,16
00012800     46  S(I,J)=0.
00012900         CALL INTGRL
00013000 C       CALL RWRITE(POS,'POS  ',3,10,10)
00013100 C       CALL RWRITE(S,'S    ',40,16,8)
00013200         DO 408 I=1,16
00013300         DO 408 J=I,16
00013400    408    HD(I,J,IF)=S(I,J)
00013500     40  CONTINUE
00013600         DO 43 I=1,16
00013700         DO 43 J=I,16
00013800         HD(J,I,3)=HD(I,J,3)
00013900     43  S(J,I)=S(I,J)
00014000         DO 44 J=1,16
00014100         DO 44 I=1,16
00014200     44  H(I,J)=HD(I,J,3)
00014300 C       CALL RWRITE(H,'H    ',40,16,8)
00014400 C------SCANNING OF K VECTOR------
00014500         DO 2 MODE=1,7
00014510         IF(IP1.NE.0.AND.MODE.GE.5) GO TO 2
00014600         DO 181 I=1,3
00014700    181    VECK(I)=AKM(I,MODE)
00014800         DO 50 J=1,8
00014900         DO 50 I=1,8
00015000           ZH(I,J)=HD(I,J,3)
00015100         DO 50 IF=1,12
00015200           ZTHETA=0.
00015300         DO 51 K=1,3
00015400     51    ZTHETA=ZTHETA+VECK(K)*F(K,IF)*(0.,1.)
00015500           ZEXP=EXP(ZTHETA)
00015600           ZH(I,J)=ZH(I,J)+HD(I,J+8,IF)*ZEXP
00015700     50  CONTINUE
00015800         CALL DEIGCH(ZH,8,40,-8,8,1.E-15,WW,LW,V,ZS)
00015900         WRITE(6,12) (VECK(I),I=1,3),(V(I),I=1,8)
00016000     12  FORMAT(' EIGENVALUES AT NK=',3F10.3,' ARE',8F9.4)
00016100         IF(IP1.EQ.0) GO TO 1003
00016200         DO 1004 I=1,8
00016300   1004  FAC(IP1,8*(MODE-1)+I)=V(I)-OLDV(I,MODE)
00016400         GO TO 2
00016500   1003  DO 1005 I=1,8
00016600   1005  OLDV(I,MODE)=V(I)
00016700      2  CONTINUE
00016800 C------MAKE ROT------
00016900         DO 60 J=1,4
00017000         DO 60 I=1,4
00017100         ROT(I,J)=.5*ROTD(I,J)
00017200         ROT(I+4,J+4)=.5*ROTD1(I,J)
00017300     60  ROT(I+8,J+8)=.5*ROTD(I,J)
00017400         DO 63 J=1,12
00017500         DO 63 I=1,12
00017600         HROT(I,J)=0.
00017700         DO 63 K=1,12
00017800         DO 63 L=1,12
```

```
00017900         HROT(I,J)=HROT(I,J)+ROT(L,I)*H(L,K)*ROT(K,J)
00018000      63 CONTINUE
00018100         RT(1)=HROT(1,1)
00018200         RT(2)=HROT(1,2)
00018300         RT(3)=HROT(1,5)
00018400         RT(4)=HROT(1,8)
00018500         RT(5)=HROT(2,8)
00018600         RT(6)=HROT(2,6)
00018700         RT(7)=HROT(1,9)
00018800         RT(8)=HROT(2,10)
00018900         RT(9)=HROT(1,12)
00019000         RT(10)=HROT(1,10)
00019100         RT(11)=HROT(4,10)
00019200         RT(12)=HROT(2,11)
00019300         RT(13)=HROT(4,9)
00019400         CALL RWRIT(RT,'RT    ',13,13)
00019500 C       CALL RWRITE(HROT,'HROT ',12,12,8)
00019600         CALL SBSURF
00019700  1000 CONTINUE
00019800 C------OUT FILE 20------
00019810         P(10)=P(10)-DP(10)
00019900         DO 1006 MODE=1,4
00020000         DO 1006 I=1,8
00020100  1006 FAC(11,8*(MODE-1)+I)=OLDV(I,MODE)
00020200         DO 1007 MODE=1,3
00020300  1007 FAC(11,MODE+32)=OLDEP(MODE)
00020400         DO 1008 IP=1,10
00020500         DO 1008 I=1,40
00020600  1008 FAC(IP,I)=FAC(IP,I)/DP(IP)
00020700         DO 1010 I=1,10
00020800  1010 FAC(I,40)=P(I)
00020900         WRITE(20) ((FAC(I,J),J=1,40),I=1,12)
00021000         CALL RMAT(FAC,'FAC   ',12,40,12,8)
00021100         STOP ;END
00021200         SUBROUTINE SBSURF
00021300         IMPLICIT COMPLEX*8 (G,V,Z)
00021400         COMPLEX*8 H(8,16),V(8,8)
00021500         COMPLEX*8 G(8,8),CDET,CW(8)
00021600         COMPLEX*8 ZP(8,8)
00021700         COMPLEX*8 GST(8,8,30),ZQ(8,8)
00021800         COMMON /R/R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13
00021900        *       ,AK1,AK2
00022000         COMMON /COMH/H
00022100         COMMON /H0HDF/ H0,HD,F
00022200         DIMENSION IP(8)
00022300         EQUIVALENCE (H(1,9),V(1,1))
00022400         DIMENSION HD(8,16,6),F(3,6),H0(8,16)
00022500         COMMON /COMSF/FAC(12,40),OLDEP(6),IP1
00022600         DIMENSION AKM(2,3),RGST(30),EST(30)
00022700         DATA AKM/2*0.,2*-3.1416,2.094,4.189/
00022800         DATA V/64*0./
00022900         DATA F/-0.6124,0.3536,0.,-0.6124,-0.3536,0.,
00023000        *   0.,0.7071,0.,0.,-0.7071,0.,
00023100        *   0.6124,-0.3536,0.,0.6124,0.3536,0./
00023200         DATA H0/128*0./,HD/768*0./
00023300         AG1=1. ;EPSG=0.05
00023400         EI=-0.9 ;EF=1.6;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
00023500         NN=(EF-EI+0.01)/DE ;NN=NN+1
00023600         ICONT=0 ;BNDCT=0.33 ;BNDCT=BNDCT/0.529
00023700         ISC=0
00023800 C------SCANNING OF K VECTOR------
00023900         DO 1 MODE=1,3
00024000         AK1=AKM(1,MODE) ;AK2=AKM(2,MODE)
00024100 C------SETTING UP OF HAMILTONIAN------
00024200         CALL SETH11
00024300 C       CALL CWRITE(H,'H     ',8,16,8)
```

```
00024400 C------RESETTING OF G-------
00024500        DO 30 I=1,8
00024600        DO 30 J=1,8
00024700     30 G(J,I)=0.
00024800 C------SCANNING OF ENERGY-----
00024900        E=EI-DE ;NUME=0
00025000 C------CHANGE ENERGY------
00025100     40 NUME=NUME+1
00025200        E=E+DE
00025300        IF(E.GT.EF+0.1*DE) GO TO 2
00025400        ZE=E+(0.,1.)*DELTA
00025500        N=0 ;IN=0 ;ISW=0
00025600     50 N=N+1
00025700 C------CALCULATION OF NEW G-------
00025800        DO 51 I=1,8
00025900        DO 51 J=1,8
00026000          ZP(I,J)=0.
00026100        DO 51 L=1,8
00026200     51   ZP(I,J)=ZP(I,J)+V(I,L)*G(L,J)
00026300        DO 52 I=1,8
00026400        DO 52 J=1,8
00026500          ZQ(I,J)=-H(I,J)
00026600        DO 52 L=1,8
00026700     52   ZQ(I,J)=ZQ(I,J)-ZP(I,L)*CONJG(V(J,L))
00026800        DO 53 I=1,8
00026900     53 ZQ(I,I)=ZQ(I,I)+ZE
00027000        CALL CINV(ZQ,8,0,8,8,1.D-14,CDET,CW,IP,NSTOP)
00027100 C      WRITE(6,205) N,EPSC,G(1,1),G(4,4),G(5,5),G(8,8),IN,ISW
00027200 C 205 FORMAT(1H ,I4,9F8.3,2I2)
00027300 C-------CONVERGENCE CHECK------
00027400        C1=ABS(G(1,1)-ZQ(1,1))/(ABS(ZQ(1,1))+0.001)
00027500        C2=ABS(G(4,4)-ZQ(4,4))/(ABS(ZQ(4,4))+0.001)
00027600        C3=ABS(G(5,5)-ZQ(5,5))/(ABS(ZQ(5,5))+0.001)
00027700        C4=ABS(G(8,8)-ZQ(8,8))/(ABS(ZQ(8,8))+0.001)
00027800        EPSC=C1+C2+C3+C4
00027900        DO 230 J=1,8
00028000        DO 230 I=1,8
00028100   230 G(I,J)=ZQ(I,J)
00028200        IF(ICONT.EQ.1.AND.ISC.EQ.1) GO TO 303
00028300        IF(EPSC.GT.EPSG) GO TO 203
00028400        GO TO 204
00028500 C-------NOT YET CONVERGED------
00028600   203 AG2=ABS(G(1,1))+ABS(G(2,2))
00028700        IF(N.LT.3) GO TO 240
00028800        IF((AG2-AG1)*(AG1-AG0).GT.0.) GO TO 241
00028900 C-------MAX OR MIN-----
00029000        IF(IN.GE.29) GO TO 244
00029100        GO TO (243,244) ,ISW
00029200   243 ISW=ISW+1
00029300 C-------BEFORE AVERAGING-----
00029400   241 IF(ISW.EQ.0) GO TO 240
00029500 C-------IN AVERAGING------
00029600        IN=IN+1
00029700 C-------STORAGE-------
00029800        DO 245 I=1,8
00029900        DO 245 J=1,8
00030000   245 GST(J,I,IN)=G(J,I)
00030100        GO TO 240
00030200 C-------AVERAGE AND CHANGE-----
00030300   244 DO 247 I=1,8
00030400        DO 247 J=1,8
00030500        G(J,I)=0.
00030600        DO 248 K=1,IN
00030700   248 G(J,I)=G(J,I)+GST(J,I,K)
00030800   247 G(J,I)=G(J,I)/IN
00030900        ISW=0 ;IN=0
```

```
00031000    240 AG0=AG1 ;AG1=AG2
00031100        GO TO 50
00031200 C-------CONVERGED------
00031300    204 CONTINUE
00031400 C      CALL CWRITE(G,'G     ',8,8,8)
00031500        IF(ICONT.EQ.0) GO TO 302
00031600 C      DO 301 J=1,4
00031700 C      DO 301 I=1,4
00031800 C      H(I,J+8)=H(I,J+8)*EXP2
00031900 C      H(I+4,J+12)=H(I+4,J+12)*EXP2
00032000 C 301  H(I+4,J+8)=H(I+4,J+8)*EXP1
00032100 C      ISC=1 ;GO TO 50
00032200    303 CONTINUE
00032300 C      ISC=0
00032400 C      DO 306 J=1,4
00032500 C      DO 306 I=1,4
00032600 C      H(I,J+8)=H(I,J+8)/EXP2
00032700 C      H(I+4,J+12)=H(I+4,J+12)/EXP2
00032800 C 306  H(I+4,J+8)=H(I+4,J+8)/EXP1
00032900    302 CONTINUE
00033000 C      WRITE(6,202) N,NUME,E,EPSC,G(1,1),G(4,4),G(5,5),G(8,8)
00033100 C      CALL CWRITE(G,'G     ',8,8,8)
00033200 C 202  FORMAT(1H ,2I3,F7.2,F8.3,8F7.3)
00033300        RGST(NUME)=-AIMAG(G(1,1))
00033400        EST(NUME)=E
00033500        GO TO 40
00033600      2 CONTINUE
00033700        PEAK=0.
00033800        DO 63 NUME=1,NN
00033900        IF(RGST(NUME).LE.PEAK) GO TO 63
00034000        PEAK=RGST(NUME)
00034100        IE=NUME
00034200     63 CONTINUE
00034300        IF(RGST(IE-1)-RGST(IE+1)) 64,65,65
00034400     64 Y1=RGST(IE) ;Y2=RGST(IE+1)
00034500        E1=EST(IE) ;E2=EST(IE+1)
00034600        GO TO 66
00034700     65 Y1=RGST(IE-1) ;Y2=RGST(IE)
00034800        E1=EST(IE-1) ;E2=EST(IE)
00034900     66 EPEAK=((Y1*E1-Y2*E2)
00035000       *      +SQRT(Y1*Y2*(E1-E2)**2-DELTA**2*(Y1-Y2)**2))/(Y1-Y2)
00035100        IF(IP1.EQ.0) GO TO 67
00035200        FAC(IP1,MODE+32)=EPEAK-OLDEP(MODE)
00035300        GO TO 68
00035400     67 OLDEP(MODE)=EPEAK
00035410     68 CONTINUE
00035500        WRITE(6,69) AK1,AK2,EPEAK,MODE
00035600     69 FORMAT(1H ,'AK1,AK2,EPEAK,MODE=',3F8.4,I5)
00035800      1 CONTINUE
00035900        RETURN ;END

ER SBFIT      OF  A1402.S.FORT  LIST END
```

BER DSBFIT JF A1402.S.FORT

```
00000100 -5.29
00000200 1.049
00000300 -2.144
00000400 2.09
00000500 2.346
00000600 -0.588
00000700 0.1233
00000800 -0.366
00000900 0.435
00001000 -0.1536
```

BER DSBFIT JF A1402.S.FORT LIST END

ER SBFIT3 JF A1402.S.FORT

```
00000100        PROGRAM SBFIT3
00000200        IMPLICIT REAL*8(A-H,J-Y)
00000300        REAL*4 FAC(12,40),RP1(10)
00000400        DIMENSION ET(25),IET(25),W(25),AA(12,12),F(40,12),
00000500       *          P0(10),P1(10),DELE0(25),DELE1(25),DP0(10),
00000600       *          E1(25),NANS(12)
00000700 C------LSUBPD (F4DLESW1,F4DLINS1)------
00000800 C------FETCH FAC FROM FILE 20-------
00000900        READ(20) ((FAC(I,J),J=1,40),I=1,12)
00001000        DO 10 I=1,12
00001100        DO 10 J=1,40
00001200     10 F(J,I)=FAC(I,J)
00001300        CALL RMAT(FAC,'FAC   ',12,40,12,8)
00001400        DO 11 I=1,10
00001500     11 P0(I)=F(40,I)
00001600 C------READ-IN IET,ET,W------
00001700        READ(5,100) N
00001800    100 FORMAT(I2)
00001900        DO 20 I=1,N
00002000        READ(5,101) IET(I),ET(I),W(I)
00002100    101 FORMAT(I2,2E10.0)
00002200     20 CONTINUE
00002300 C------CHANGING ORDER----
00002400        DO 30 I=1,N
00002500        DO 30 J=1,12
00002600     30 F(I,J)=F(IET(I),J)
00002700 C------CALCULATE DELE0-----
00002800        DO 31 I=1,N
00002900     31 DELE0(I)=ET(I)-F(I,11)
00003000 C------SOLVE-----
00003100        CALL DLESW1(F,N,10,40,12,DELE0,W,AA,NANS,DP0,NSTOP)
00003200        WRITE(6,220) NSTOP
00003300    220 FORMAT(' STOP=',I2)
00003400 C------CALCULATE NEW PARAMETERS-----
00003500        DO 32 I=1,10
00003600     32 P1(I)=P0(I)+DP0(I)
00003700        CALL DWRIT(P1,'NEWP  ',10,10)
00003800 C------EVALUATION OF NEW PARAMETERS-----
00003900        DO 33 I=1,N
00004000        E1(I)=F(I,11)
00004100        DO 34 J=1,10
00004200     34 E1(I)=E1(I)+F(I,J)*DP0(J)
00004300     33 DELE1(I)=ET(I)-E1(I)
00004400 C------OUTPUT------
00004500        WRITE(6,103)
00004600    103 FORMAT(1H0,' I IET      ET          W          E0',
00004700       *          '          DELE0      E1          DELE1')
00004800        DO 40 I=1,N
00004900     40 WRITE(6,102) I,IET(I),ET(I),W(I),F(I,11),DELE0(I),E1(I),DELE1(
00005000    102 FORMAT(1H ,2I3,8F10.4)
00005100 C------OUT FILE ON 30-----
00005200        DO 41 I=1,10
00005300     41 RP1(I)=P1(I)
00005310 C      RP1(1)=RP1(1)-E1(2) ;RP1(2)=RP1(2)-E1(2)
00005400        WRITE(30) (RP1(I),I=1,10)
00005500        STOP ;END
```

ER SBFIT3 JF A1402.S.FORT LIST END

ER DFIT JF A1402.S.FORT

```
00000100 16
00000200  1 -12.5      1.
00000300  2 0.         10.
00000400  5 3.5         1.
00000500  8 4.0         1.
00000600  9 -8.7        1.
00000700 11 -3.0        1.
00000800 13 1.3         1.
00000900 17 -10.5       1.
00001000 18 -7.7        1.
00001100 19 -1.2        1.
00001200 21 1.6         1.
00001300 23 4.0         1.
00001400 29 1.12       10.
00001500 33 0.7        10.
00001600 34 0.2         1.
00001700 35 0.7        10.
```

ER DFIT JF A1402.S.FORT LIST END

ER BAND OF A1402.S.FORT

```
00000100        SUBROUTINE COEFFT
00000200        IMPLICIT REAL*4(A-H,O-Z)
00000300        COMMON/ARRAYS/S(40,40),Y(9135),Z( 765)
00000400        DO 1 I=1,9135
00000500      1 Y(I)=0.0D0
00000600        DO 2 I=1,765
00000700      2 Z(I)=0.0D0
00000800 C-----LOAD NON-ZERO Y COEFFICIENTS
00000900        Y(7039)=    64.D0
00001000        Y(7040)=    64.D0
00001100        Y(7049)=   -64.D0
00001200        Y(7032)=  -128.D0
00001300        Y(7041)=   -64.D0
00001400        Y(7033)=  -128.D0
00001500        Y(7042)=   128.D0
00001600        Y(7025)=    64.D0
00001700        Y(7034)=   128.D0
00001800        Y(7026)=    64.D0
00001900        Y(7035)=   -64.D0
00002000        Y(7027)=   -64.D0
00002100        Y(6904)=   -96.D0
00002200        Y(6913)=    32.D0
00002300        Y(6896)=  -192.D0
00002400        Y(6905)=   192.D0
00002500        Y(6906)=   288.D0
00002600        Y(6915)=   -96.D0
00002700        Y(6889)=   192.D0
00002800        Y(6907)=  -192.D0
00002900        Y(6890)=    96.D0
00003000        Y(6899)=  -288.D0
00003100        Y(6891)=  -192.D0
00003200        Y(6900)=   192.D0
00003300        Y(6892)=   -32.D0
00003400        Y(6901)=    96.D0
00003500        Y(2854)=   -16.D0
00003600        Y(2863)=    16.D0
00003700        Y(2847)=    32.D0
00003800        Y(2856)=   -16.D0
00003900        Y(2865)=   -16.D0
00004000        Y(2840)=   -16.D0
00004100        Y(2849)=   -16.D0
00004200        Y(2858)=    32.D0
00004300        Y(2842)=    16.D0
00004400        Y(2851)=   -16.D0
00004500        Y(2710)=    48.D0
00004600        Y(2719)=   -48.D0
00004700        Y(2711)=    48.D0
00004800        Y(2720)=   -96.D0
00004900        Y(2729)=    48.D0
00005000        Y(2703)=   -48.D0
00005100        Y(2712)=   -48.D0
00005200        Y(2721)=    96.D0
00005300        Y(2704)=   -48.D0
00005400        Y(2713)=    48.D0
00005500        Y(2722)=    48.D0
00005600        Y(2731)=   -48.D0
00005700        Y(2705)=    96.D0
00005800        Y(2714)=   -48.D0
```

```
00005900      Y(2723)=   -48.D0
00006000      Y(2706)=    48.D0
00006100      Y(2715)=   -96.D0
00006200      Y(2724)=    48.D0
00006300      Y(2707)=   -48.D0
00006400      Y(2716)=    48.D0
00006500      Y(5329)=    64.D0
00006600      Y(5322)=  -128.D0
00006700      Y(5340)=   -64.D0
00006800      Y(5315)=    64.D0
00006900      Y(5333)=   128.D0
00007000      Y(5326)=   -64.D0
00007100      Y(5185)=   -96.D0
00007200      Y(5194)=    32.D0
00007300      Y(5186)=   -96.D0
00007400      Y(5195)=    64.D0
00007500      Y(5204)=    32.D0
00007600      Y(5178)=    96.D0
00007700      Y(5187)=    32.D0
00007800      Y(5196)=    64.D0
00007900      Y(5179)=    96.D0
00008000      Y(5188)=   -32.D0
00008100      Y(5197)=    32.D0
00008200      Y(5206)=   -96.D0
00008300      Y(5180)=   -64.D0
00008400      Y(5189)=   -32.D0
00008500      Y(5198)=   -96.D0
00008600      Y(5181)=   -32.D0
00008700      Y(5190)=   -64.D0
00008800      Y(5199)=    96.D0
00008900      Y(5182)=   -32.D0
00009000      Y(5191)=    96.D0
00009100      Y(4375)=  -144.D0
00009200      Y(4384)=    96.D0
00009300      Y(4393)=   -16.D0
00009400      Y(4368)=   144.D0
00009500      Y(4386)=   -48.D0
00009600      Y(4395)=    96.D0
00009700      Y(4370)=   -96.D0
00009800      Y(4379)=    48.D0
00009900      Y(4397)=  -144.D0
00010000      Y(4372)=    16.D0
00010100      Y(4381)=   -96.D0
00010200      Y(4390)=   144.D0
00010300      Y(1900)=   144.D0
00010400      Y(1909)=  -144.D0
00010500      Y(1893)=  -144.D0
00010600      Y(1920)=   144.D0
00010700      Y(1895)=   144.D0
00010800      Y(1922)=  -144.D0
00010900      Y(1906)=  -144.D0
00011000      Y(1915)=   144.D0
00011100      Y( 955)=   -16.D0
00011200      Y( 964)=    32.D0
00011300      Y( 973)=   -16.D0
00011400      Y( 948)=    16.D0
00011500      Y( 966)=   -48.D0
00011600      Y( 975)=    32.D0
00011700      Y( 950)=   -32.D0
00011800      Y( 959)=    48.D0
00011900      Y( 977)=   -16.D0
00012000      Y( 952)=    16.D0
00012100      Y( 961)=   -32.D0
00012200      Y( 970)=    16.D0
00012300      Y(8155)=    64.D0
00012400      Y(8156)=   -64.D0
```

```
00012500        Y(8165)=   -64.D0
00012600        Y(8148)=   -64.D0
00012700        Y(8157)=    64.D0
00012800        Y(8149)=    64.D0
00012900        Y(8158)=    64.D0
00013000        Y(8150)=   -64.D0
00013100        Y(8020)=   -96.D0
00013200        Y(8029)=    32.D0
00013300        Y(8021)=   128.D0
00013400        Y(8013)=    96.D0
00013500        Y(8031)=   -96.D0
00013600        Y(8014)=  -128.D0
00013700        Y(8015)=   -32.D0
00013800        Y(8024)=    96.D0
00013900        Y(7084)=   -64.D0
00014000        Y(7076)=  -128.D0
00014100        Y(7085)=    64.D0
00014200        Y(7086)=   128.D0
00014300        Y(7069)=   128.D0
00014400        Y(7070)=    64.D0
00014500        Y(7079)=  -128.D0
00014600        Y(7071)=   -64.D0
00014700        Y(3205)=   -16.D0
00014800        Y(3214)=   -16.D0
00014900        Y(3206)=    16.D0
00015000        Y(3215)=   -16.D0
00015100        Y(3198)=    16.D0
00015200        Y(3216)=   -16.D0
00015300        Y(3199)=   -16.D0
00015400        Y(3217)=    16.D0
00015500        Y(3200)=   -16.D0
00015600        Y(3209)=    16.D0
00015700        Y(3201)=    16.D0
00015800        Y(3210)=   -16.D0
00015900        Y(7579)=    64.D0
00016000        Y(7580)=   -64.D0
00016100        Y(7572)=  -128.D0
00016200        Y(7573)=   128.D0
00016300        Y(7565)=    64.D0
00016400        Y(7566)=   -64.D0
00016500        Y(5680)=    64.D0
00016600        Y(5681)=   -64.D0
00016700        Y(5673)=   -64.D0
00016800        Y(5691)=   -64.D0
00016900        Y(5674)=    64.D0
00017000        Y(5692)=    64.D0
00017100        Y(5684)=    64.D0
00017200        Y(5685)=   -64.D0
00017300        Y(7435)=   -96.D0
00017400        Y(7444)=    32.D0
00017500        Y(7436)=   -96.D0
00017600        Y(7445)=   160.D0
00017700        Y(7428)=    96.D0
00017800        Y(7437)=   128.D0
00017900        Y(7446)=   -96.D0
00018000        Y(7429)=    96.D0
00018100        Y(7438)=  -128.D0
00018200        Y(7447)=   -96.D0
00018300        Y(7430)=  -160.D0
00018400        Y(7439)=    96.D0
00018500        Y(7431)=   -32.D0
00018600        Y(7440)=    96.D0
00018700        Y(5545)=   -96.D0
00018800        Y(5554)=    32.D0
00018900        Y(5546)=    32.D0
00019000        Y(5555)=    32.D0
```

```
00019100          Y(5538)=    96.D0
00019200          Y(5556)=    32.D0
00019300          Y(5539)=   -32.D0
00019400          Y(5557)=   -96.D0
00019500          Y(5540)=   -32.D0
00019600          Y(5549)=   -32.D0
00019700          Y(5541)=   -32.D0
00019800          Y(5550)=    96.D0
00019900          Y(3070)=    48.D0
00020000          Y(3079)=   -48.D0
00020100          Y(3071)=   -48.D0
00020200          Y(3080)=    48.D0
00020300          Y(3063)=   -48.D0
00020400          Y(3081)=    48.D0
00020500          Y(3064)=    48.D0
00020600          Y(3082)=   -48.D0
00020700          Y(3065)=    48.D0
00020800          Y(3074)=   -48.D0
00020900          Y(3066)=   -48.D0
00021000          Y(3075)=    48.D0
00021100          Y(8200)=   -64.D0
00021200          Y(8201)=    64.D0
00021300          Y(8193)=    64.D0
00021400          Y(8194)=   -64.D0
00021500          Y(7615)=   -64.D0
00021600          Y(7616)=   -64.D0
00021700          Y(7625)=    64.D0
00021800          Y(7608)=    64.D0
00021900          Y(7617)=    64.D0
00022000          Y(7609)=    64.D0
00022100          Y(7618)=   -64.D0
00022200          Y(7610)=   -64.D0
00022300          Y(3250)=    16.D0
00022400          Y(3259)=   -16.D0
00022500          Y(3243)=   -16.D0
00022600          Y(3261)=    16.D0
00022700          Y(3245)=    16.D0
00022800          Y(3254)=   -16.D0
00022900          Y(5725)=   -64.D0
00023000          Y(5718)=    64.D0
00023100          Y(5736)=    64.D0
00023200          Y(5729)=   -64.D0
00023300 C-----LOAD NON-ZERO Z COEFFICIENTS
00023400          Z(341)=    -1.D0
00023500          Z(343)=     3.D0
00023600          Z(345)=    -3.D0
00023700          Z(347)=     1.D0
00023800          Z(664)=    -1.D0
00023900          Z(665)=     5.D0
00024000          Z(666)=   -10.D0
00024100          Z(667)=    10.D0
00024200          Z(668)=    -5.D0
00024300          Z(669)=     1.D0
00024400          Z(154)=    -1.D0
00024500          Z(156)=     5.D0
00024600          Z(158)=   -10.D0
00024700          Z(160)=    10.D0
00024800          Z(162)=    -5.D0
00024900          Z(164)=     1.D0
00025000          Z(222)=    -1.D0
00025100          Z(223)=     1.D0
00025200          Z(224)=     4.D0
00025300          Z(225)=    -4.D0
00025400          Z(226)=    -6.D0
00025500          Z(227)=     6.D0
00025600          Z(228)=     4.D0
```

```
00025700         Z(229)=    -4.D0
00025800         Z(230)=    -1.D0
00025900         Z(231)=     1.D0
00026000         Z(307)=    -1.D0
00026100         Z(308)=     2.D0
00026200         Z(309)=     2.D0
00026300         Z(310)=    -6.D0
00026400         Z(312)=     6.D0
00026500         Z(313)=    -2.D0
00026600         Z(314)=    -2.D0
00026700         Z(315)=     1.D0
00026800         Z(409)=    -1.D0
00026900         Z(410)=     3.D0
00027000         Z(411)=    -1.D0
00027100         Z(412)=    -5.D0
00027200         Z(413)=     5.D0
00027300         Z(414)=     1.D0
00027400         Z(415)=    -3.D0
00027500         Z(416)=     1.D0
00027600         Z(528)=    -1.D0
00027700         Z(529)=     4.D0
00027800         Z(530)=    -5.D0
00027900         Z(532)=     5.D0
00028000         Z(533)=    -4.D0
00028100         Z(534)=     1.D0
00028200         Z(562)=    -1.D0
00028300         Z(563)=     2.D0
00028400         Z(565)=    -2.D0
00028500         Z(566)=     1.D0
00028600         Z(732)=    -1.D0
00028700         Z(733)=     1.D0
00028800         Z(545)=     1.D0
00028900         Z(546)=    -3.D0
00029000         Z(547)=     2.D0
00029100         Z(548)=     2.D0
00029200
00029300
00029400         Z(549)=    -3.D0
00029500         Z(550)=     1.D0
00029600         Z(579)=     1.D0
00029700         Z(580)=    -1.D0
00029800         Z(581)=    -1.D0
00029900         Z(582)=     1.D0
00030000         Z(596)=    -1.D0
00030100         Z(598)=     1.D0
00030200         Z(443)=    -1.D0
00030300         Z(444)=     1.D0
00030400         Z(445)=     2.D0
00030500         Z(446)=    -2.D0
00030600         Z(447)=    -1.D0
00030700         Z(448)=     1.D0
00030800         Z(698)=    -1.D0
00030900         Z(699)=     3.D0
00031000         Z(700)=    -3.D0
00031100         Z(701)=     1.D0
00031200         Z(324)=     1.D0
00031300         Z(325)=    -1.D0
00031400         Z(326)=    -3.D0
00031500         Z(327)=     3.D0
00031600         Z(328)=     3.D0
00031700         Z(329)=    -3.D0
00031800         Z(330)=    -1.D0
00031900         Z(331)=     1.D0
00032000         Z(460)=     1.D0
00032100         Z(462)=    -2.D0
00032200         Z(464)=     1.D0
```

```
00032300       RETURN
00032400       END
00032500       FUNCTION SS(NN1,LL1,MM,NN2,LL2,ALPHA,BETA)
00032600       IMPLICIT REAL*4(A-H,O-Z)
00032700 C-----PROCEDURE FOR CALCULATING REDUCED OVERLAP INTEGRALS
00032800       COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
00032900       COMMON/AUXINT/A(17),B(17)
00033000       INTEGER ULIM
00033100       N1=NN1
00033200       L1=LL1
00033300       M=MM
00033400       N2=NN2
00033500       L2=LL2
00033600       P =(ALPHA + BETA)/2.D0
00033700       PT=(ALPHA - BETA)/2.D0
00033800       K = 0.D0
00033900       M=IABS(M)
00034000 C-----REVERSE QUANTUM NUMBERS IF NECESSARY
00034100       IF((L2.LT.L1).OR.((L2.EQ.L1).AND.(N2.LT.N1))) GO TO 20
00034200    10 GO TO 30
00034300    20 K = N1
00034400       N1= N2
00034500       N2= K
00034600       K= L1
00034700       L1= L2
00034800       L2= K
00034900       PT=-PT
00035000    30 CONTINUE
00035100       K = MOD((N1+N2-L1-L2),2)
00035200 C-----FIND A AND B INTEGRALS
00035300       CALL AINTGS(P,N1+N2)
00035400       CALL BINTGS(PT,N1+N2)
00035500       IF((L1.GT.0).OR.(L2.GT.0)) GO TO 60
00035600 C-----BEGIN SECTION USED FOR OVERLAP INTEGRALS INVOLVING S FUNCTIONS
00035700 C-----FIND Z TABLE NUMBER L
00035800    40 L = (90-17*N1+N1**2-2*N2)/2
00035900       ULIM = N1+N2
00036000       LLIM = 0
00036100       DO 50 I=LLIM,ULIM
00036200       NNI1=N1+N2-I+1
00036300    50 X=X+Z(I+1,L)*A(I+1)*B(NNI1)/2.D0
00036400       SS=X
00036500       GO TO 80
00036600 C-----BEGIN SECTION USED FOR OVERLAPS INVOLVING NON-S FUNCTIONS
00036700 C-----FIND Y TABLE NUMBER L
00036800    60 L=(5-M)*(24-10*M+M**2)*(83-30*M+3*M**2)/120+
00036900      1  (30-9*L1+L1**2-2*N1)*(28-9*L1+L1**2-2*N1)/8+
00037000      2  (30-9*L2+L2**2-2*N2)/2
00037100       LLIM = 0
00037200       DO 70 I=LLIM,8
00037300       ULIM=4 - MOD(K+I,2)
00037400       DO 70 J=LLIM,ULIM
00037500       IIII=2*J+MOD(K+I,2)+1
00037600    70 X=X+Y(I+1,J+1,L)*A(I+1)*B(IIII)
00037700       SS = X*(FACT(M+1)/8.D0)**2* SQRT( FLOAT(2*L1+1)*FACT(L1-M)*
00037800      1  FLOAT(2*L2+1)*FACT(L2-M)/(4.D0*FACT(L1+M)*FACT(L2+M)))
00037900    80 CONTINUE
00038000       RETURN
00038100       END
00038200       SUBROUTINE HARMTR(T,MAXL,E)
00038300       IMPLICIT REAL*4(A-H,O-Z)
00038400       DIMENSION T(9,9),E(3)
00038500       COST = E(3)
00038600       IF((1.D0-COST**2).GT.0.0000000001) GO TO 20
00038700    10 SINT = 0.D0
00038800       GO TO 30
```

```
00038900      20 SINT= SQRT(1.D0-COST**2)
00039000      30 CONTINUE
00039100         IF(SINT.GT.0.000001D0)  GO TO 50
00039200      40 COSP = 1.D0
00039300         SINP = 0.D0
00039400         GO TO 70
00039500      50 COSP = E(1)/SINT
00039600      60 SINP = E(2)/SINT
00039700      70 CONTINUE
00039800         DO 80 I=1,9
00039900         DO 80 J=1,9
00040000      80 T(I,J) = 0.D0
00040100         T(1,1) =1.D0
00040200         IF (MAXL.GT.1) GO TO 100
00040300      90 IF (MAXL.GT.0) GO TO 110
00040400         GO TO 120
00040500    100 COS2T = COST**2-SINT**2
00040600         SIN2T = 2.D0*SINT*COST
00040700         COS2P = COSP**2-SINP**2
00040800         SIN2P = 2.D0*SINP*COSP
00040900 C-----TRANSFORMATION MATRIX ELEMENTS FOR D FUNCTIONS
00041000         SQRT3= SQRT(3.D0)
00041100         T(5,5) = (3.D0*COST**2-1.D0)/2.D0
00041200         T(5,6) = -SQRT3   *SIN2T/2.D0
00041300         T(5,8) = SQRT3    *SINT**2/2.D0
00041400         T(6,5) = SQRT3    *SIN2T*COSP/2.D0
00041500         T(6,6) = COS2T*COSP
00041600         T(6,7) = -COST*SINP
00041700         T(6,8) =-T(6,5)/SQRT3
00041800         T(6,9) = SINT*SINP
00041900         T(7,5) = SQRT3    *SIN2T*SINP/2.D0
00042000         T(7,6) = COS2T*SINP
00042100         T(7,7) = COST*COSP
00042200         T(7,8) = -T(7,5)/SQRT3
00042300         T(7,9) = -SINT*COSP
00042400         T(8,5) = SQRT3    *SINT**2*COS2P/2.D0
00042500         T(8,6) = SIN2T*COS2P/2.D0
00042600         T(8,7) = -SINT*SIN2P
00042700          T(8,8) = (1.D0+COST**2)*COS2P/2.D0
00042800         T(8,9) = -COST*SIN2P
00042900         T(9,5) = SQRT3    *SINT**2*SIN2P/2.D0
00043000         T(9,6) = SIN2T*SIN2P/2.D0
00043100         T(9,7) = SINT*COS2P
00043200         T(9,8) = (1.D0+COST**2)*SIN2P/2.D0
00043300         T(9,9) = COST*COS2P
00043400    110 CONTINUE
00043500 C-----TRANSFORMATION MATRIX ELEMENTS FOR P FUNCTIONS
00043600         T(2,2) = COST*COSP
00043700         T(2,3) = -SINP
00043800         T(2,4) = SINT*COSP
00043900         T(3,2) = COST*SINP
00044000         T(3,3) = COSP
00044100         T(3,4) = SINT*SINP
00044200         T(4,2) = -SINT
00044300         T(4,4) = COST
00044400    120 CONTINUE
00044500         RETURN
00044600         END
00044700         SUBROUTINE RELVEC(R,E,C1,C2)
00044800         IMPLICIT REAL*4(A-H,O-Z)
00044900         DIMENSION E(3),C1(3),C2(3)
00045000         X = 0.D0
00045100         DO 10 I=1,3
00045200         E(I) = C2(I)-C1(I)
00045300         X = X+E(I)**2
00045400      10 CONTINUE
```

```
00045500         R= SQRT(X)
00045600         DO 40 I=1,3
00045700         IF (R.GT..000001D0) GO TO 30
00045800      20 GO TO 40
00045900      30 E(I) =E(I)/R
00046000      40 CONTINUE
00046100         RETURN
00046200         END
00046300         FUNCTION FACT(N)
00046400         IMPLICIT REAL*4(A-H,O-Z)
00046500         PRODT = 1.D0
00046600      20 DO 30 I=1,N
00046700      30 PRODT=PRODT* FLOAT(I)
00046800      40 FACT=PRODT
00046900         RETURN
00047000         END
00047100         SUBROUTINE BINTGS(X,K)
00047200         IMPLICIT REAL*4(A-H,O-Z)
00047300 C-----FILLS ARRAY OF B-INTEGRALS. NOTE THAT B(I) IS B(I-1) IN THE
00047400 C-----USUAL NOTATION
00047500 C-----FOR X.GT.3                          EXPONENTIAL FORMULA IS USED
00047600 C-----FOR 2.LT.X.LE.3 AND K.LE.10         EXPONENTIAL FORMULA IS USED
00047700 C-----FOR 2.LT.X.LE.3 AND K.GT.10         15 TERM SERIES IS USED
00047800 C-----FOR 1.LT.X .E.2 AND K.LE.7          EXPONENTIAL FORMULA IS USED
00047900 C-----FOR 1.LT.X.LE.2 AND K.GT.7          12 TERM SERIES IS USED
00048000 C-----FOR .5.LT.X.LE.1 AND K.LE.5         EXPONENTIAL FORMULA IS USED
00048100 C-----FOR .5.LT.X.LE.1 AND K.GT.5          7 TERM SERIES IS USED
00048200 C-----FOR X.LE..5                          6 TERM SERIES IS USED
00048300 C-----*****************************************************************
00048400         COMMON/AUXINT/A(17),B(17)
00048500         I0=0
00048600         ABSX= ABS(X)
00048700         IF(ABSX.GT.3.D0) GO TO 120
00048800      10 IF(ABSX.GT.2.D0) GO TO 20
00048900      40 IF(ABSX.GT.1.D0) GO TO 50
00049000      70 IF(ABSX.GT..5D0) GO TO 80
00049100     100 IF(ABSX.GT..000001D0) GO TO 110
00049200         GO TO 170
00049300     110 LAST=6
00049400         GO TO 140
00049500      80 IF(K.LE.5) GO TO 120
00049600      90 LAST=7
00049700         GO TO 140
00049800      50 IF(K.LE.7) GO TO 120
00049900      60 LAST=12
00050000         GO TO 140
00050100      20 IF(K.LE.10) GO TO 120
00050200      30 LAST=15
00050300         GO TO 140
00050400
00050500     120 EXPX= EXP(X)
00050600         EXPMX=1.D0/EXPX
00050700         B(1)=(EXPX-EXPMX)/X
00050800         DO 130 I=1,K
00050900     130 B(I+1)=( FLOAT(I)*B(I)+(-1.D0)**I*EXPX-EXPMX)/X
00051000         GO TO 190
00051100     140 DO 160 I=I0,K
00051200         Y=0.D0
00051300         DO 150 M=I0,LAST
00051400     150 Y=Y+(-X)**M*(1.D0-(-1.D0)**(M+I+1))/(FACT(M)* FLOAT(M+I+1))
00051500     160 B(I+1)=Y
00051600         GO TO 190
00051700
00051800     170 DO 180 I=I0,K
00051900     180 B(I+1)=(1.D0-(-1.D0)**(I+1))/ FLOAT(I+1)
00052000     190 CONTINUE
```

```
00052100          RETURN
00052200          END
00052300          SUBROUTINE AINTGS(X,K)
00052400          IMPLICIT REAL*4(A-H,O-Z)
00052500          COMMON/AUXINT/A(17),B(17)
00052600          A(1) = EXP(-X)/X
00052700          DO 10  I=1,K
00052800       10 A(I+1) =(A(I)* FLOAT(I)+ EXP(-X))/X
00052900          RETURN
00053000          END
00053100          SUBROUTINE DEIGCH(A,N,N1,NE,NV,EPS,W,LW,E,V)
00053200          IMPLICIT REAL*8 (A-H, O-Z)
00053300          COMPLEX*16 A,V,CR
00053400          COMPLEX*32 CS
00053500          REAL*16 S
00053600          LOGICAL SW, LW
00053700          DIMENSION A(N1,1), E(1), V(N1,1), W(N1,7), LW(1)
00053800          IF(N.LE.0 .OR. NE.EQ.0 ) GO TO 910
00053900          NEA=IABS(NE)
00054000          NVA=IABS(NV)
00054100          IF(N1.LT.N .OR. N.LT.NEA .OR. NEA.LT.NVA ) GO TO 920
00054200          IF(EPS.LT.0.D0) EPS=1.D-16
00054300          NM1=N-1
00054400          N2=N-2
00054500          IF(N2) 10, 20, 50
00054600 C     WHEN N=1
00054700       10 E(1)=A(1,1)
00054800          IF( NV.NE.0 ) V(1,1) = 1.0D0
00054900          GO TO 900
00055000 C     WHEN N=2
00055100 C     COMPUTE EIGENVALUES OF 2*2 MATRIX
00055200       20 CALL ERRSET(202,256,-1,1)
00055300          W(1,1)=A(1,1)
00055400          W(2,1)=A(2,2)
00055500          W(1,2)=CDABS(A(2,1))
00055600          A(1,1)=A(2,1)/W(1,2)
00055700          T = 0.5D0*(W(1,1)+W(2,1))
00055800          R=W(1,1)*W(2,1)-W(1,2)**2
00055900          D=T*T-R
00056000          Q=DABS(T)+DSQRT(D)
00056100          IF(T.LT.0.) Q=-Q
00056200          T=T*DFLOAT(NE)
00056300          IF(T) 40, 30, 30
00056400       30 E(1)=Q
00056500          IF(NEA.EQ.2) E(2)=R/Q
00056600          GO TO 310
00056700       40 E(1)=R/Q
00056800          IF(NEA.EQ.2) E(2)=Q
00056900          GO TO 310
00057000 C     WHEN N=3,4,...
00057100 C     REDUCE TO TRIDIAGONAL FORM BY HOUSEHOLDER'S METHOD
00057200       50 DO 60 I=1,N
00057300       60 W(I,1)=A(I,I)
00057400          DO 190 K=1,N2
00057500          S=0.
00057600          DO 70 I=K+1,N
00057700       70 S=S+DREAL(A(I,K))**2+DIMAG(A(I,K))**2
00057800          SR = QSQRT(S)
00057900          T=CDABS(A(K+1,K))
00058000          W(K,2)=-SR
00058100          IF(T) 90, 80, 90
00058200       80 A(K,K) = 1.0D0
00058300          GO TO 100
00058400       90 A(K,K)=A(K+1,K)/T
00058500      100 IF(S.EQ.0.) GO TO 190
00058600          R = 1.0D0/(S+T*SR)
```

```
00058700          A(K+1,K)=A(K+1,K)+SR*A(K,K)
00058800          DO 140 I=K+1,N
00058900          CS=(0.,0.)
00059000          IF(I.EQ.K+1) GO TO 120
00059100          DO 110 J=K+1,I-1
00059200      110 CS=CS+A(I,J)*A(J,K)
00059300      120 CS=CS+W(I,1)*A(I,K)
00059400          IF(I.EQ.N) GO TO 140
00059500          DO 130 J=I+1,N
00059600      130 CS=CS+DCONJG(A(J,I))*A(J,K)
00059700      140 A(I,I)=CS*R
00059800          CS=(0.,0.)
00059900          DO 150 I=K+1,N
00060000      150 CS=CS+DCONJG(A(I,K))*A(I,I)
00060100          CS = 0.5D0*R*CS
00060200          DO 160 I=K+1,N
00060300      160 A(I,I)=A(I,I)-CS*A(I,K)
00060400          DO 170 I=K+1,N
00060500      170 W(I,1) = W(I,1)-2.0D0*DREAL(A(I,K)*DCONJG(A(I,I)))
00060600          DO 180 I=K+2,N
00060700          DO 180 J=K+1,I-1
00060800      180 A(I,J)=A(I,J)-A(I,K)*DCONJG(A(J,J))-A(I,I)*DCONJG(A(J,K))
00060900      190 CONTINUE
00061000          W(NM1,2)=CDABS(A(N,NM1))
00061100          IF(W(NM1,2)) 196,195,196
00061200      195 A(NM1,NM1)=1.D0
00061300          GO TO 198
00061400      196 CONTINUE
00061500          A(NM1,NM1)=A(N,NM1)/W(NM1,2)
00061600      198 CONTINUE
00061700 C  COMPUTE EIGENVALUES BY BISECTION METHOD
00061800          CALL ERRSET(202,256,-1,1)
00061900          R=DMAX1((DABS(W(1,1))+DABS(W(1,2))),(DABS(W(NM1,2))+DABS(W(N,1
00062000          DO 200 I=2,NM1
00062100          T=DABS(W(I-1,2))+DABS(W(I,1))+DABS(W(I,2))
00062200          IF(T.GT.R) R=T
00062300      200 CONTINUE
00062400          EPS1=R*0.1D-15
00062500          EPS2=R*EPS
00062600          DO 210 I=1,NM1
00062700      210 W(I,3)=W(I,2)**2
00062800          IF(NE.LT.0) R=-R
00062900          F=R
00063000          DO 220 I=1,NEA
00063100      220 E(I)=-R
00063200          DO 300 K=1,NEA
00063300          D=E(K)
00063400      230 T = 0.5D0*(D+F)
00063500          IF(DABS(D-T).LE.EPS2 .OR. DABS(F-T).LE.EPS2 ) GO TO 300
00063600          J=0
00063700          I=1
00063800      240 Q=W(I,1)-T
00063900      250 IF(Q.GE.0.) J=J+1
00064000          IF(Q.EQ.0.) GO TO 260
00064100          I=I+1
00064200          IF(I.GT.N) GO TO 270
00064300          CALL OVERFL(L)
00064400          Q=W(I,1)-T-W(I-1,3)/Q
00064500          CALL OVERFL(L)
00064600          IF(L.NE.1) GO TO 250
00064700          J=J+1
00064800          I=I-1
00064900      260 I=I+2
00065000          IF(I.LE.N) GO TO 240
00065100      270 IF(NE.LT.0) J=N-J
00065200          IF(J.GE.K) GO TO 280
```

```
00065300          F=T
00065400          GO TO 230
00065500      280 D=T
00065600          M=MIN0(J,NEA)
00065700          DO 290 I=K,M
00065800      290 E(I)=T
00065900          GO TO 230
00066000      300 E(K)=T
00066100  C   COMPUTE EIGENVECTORS BY INVERSE ITERATION
00066200      310 CALL ERRSET(202, 10, 5,2)
00066300          IF(NV.EQ.0) GO TO 900
00066400          MM=584287
00066500          CALL ERRSET(202,256,-1,1)
00066600          DO 490 I=1,NVA
00066700          DO 320 J=1,N
00066800          W(J,3)=W(J,1)-E(I)
00066900          W(J,4)=W(J,2)
00067000      320 W(J,7) = 1.0D0
00067100          SW=.FALSE.
00067200  C   REDUCE TO TRIANGULAR FORM
00067300          DO 340 J=1,NM1
00067400          IF(DABS(W(J,3)).LT.DABS(W(J,2))) GO TO 330
00067500          IF(W(J,3).EQ.0.) W(J,3)=1.0D-30
00067600          W(J,6)=W(J,2)/W(J,3)
00067700          LW(J)=.FALSE.
00067800          W(J+1,3)=W(J+1,3)-W(J,6)*W(J,4)
00067900          W(J,5)=0.
00068000          GO TO 340
00068100      330 W(J,6)=W(J,3)/W(J,2)
00068200          LW(J)=.TRUE.
00068300          W(J,3)=W(J,2)
00068400          T=W(J,4)
00068500          W(J,4)=W(J+1,3)
00068600          W(J,5)=W(J+1,4)
00068700          W(J+1,3)=T-W(J,6)*W(J,4)
00068800          W(J+1,4)=-W(J,6)*W(J,5)
00068900      340 CONTINUE
00069000          IF(W(N,3).EQ.0.) W(N,3)=1.0D-30
00069100  C   BEGIN BACK SUBSTITUTION
00069200          IF(I.EQ.1 .OR. DABS(E(I)-E(I-1)).GE.EPS1) GO TO 360
00069300  C   GENERATE RANDOM NUMBERS
00069400          DO 350 J=1,N
00069500          MM=MM*48828125
00069600      350 W(J,7)=FLOAT(MM)*0.4656613E-9
00069700      360 CALL OVERFL(L)
00069800          T=W(N,7)
00069900          R=W(N-1,7)
00070000      370 W(N,7)=T/W(N,3)
00070100          W(N-1,7)=(R-W(N-1,4)*W(N,7))/W(N-1,3)
00070200          CALL OVERFL(L)
00070300          IF(L.NE.1) GO TO 390
00070400          DO 380 J=1,N2
00070500      380 W(J,7)=W(J,7)*1.0D-5
00070600          T=T*1.0D-5
00070700          R=R*1.0D-5
00070800          GO TO 370
00070900      390 IF(N.EQ.2) GO TO 440
00071000          K=N2
00071100      400 T=W(K,7)
00071200      410 W(K,7)=(T-W(K,4)*W(K+1,7)-W(K,5)*W(K+2,7))/W(K,3)
00071300          CALL OVERFL(L)
00071400          IF(L.NE.1) GO TO 430
00071500          DO 420 J=1,N
00071600      420 W(J,7)=W(J,7)*1.0D-5
00071700          T=T*1.0D-5
00071800          GO TO 410
```

```
00071900    430 K=K-1
00072000        IF(K) 440,440,400
00072100    440 IF(SW) GO TO 470
00072200        SW=.TRUE.
00072300        DO 460 J=1,NM1
00072400        IF(LW(J)) GO TO 450
00072500        W(J+1,7)=W(J+1,7)-W(J,6)*W(J,7)
00072600        GO TO 460
00072700    450 T=W(J,7)
00072800        W(J,7)=W(J+1,7)
00072900        W(J+1,7)=T-W(J,6)*W(J+1,7)
00073000    460 CONTINUE
00073100        GO TO 360
00073200    470 DO 480 J=1,N
00073300    480 V(J,I)=W(J,7)
00073400    490 CONTINUE
00073500 C   BEGIN BACK TRANSFORMATION (1)
00073600        CR = 1.0D0
00073700        DO 500 J=2,N
00073800        CR=CR*A(J-1,J-1)
00073900        DO 500 I=1,NVA
00074000    500 V(J,I)=V(J,I)*CR
00074100 C   BEGIN BACK TRANSFORMATION (2)
00074200        CALL ERRSET(202, 10, 5,2)
00074300        IF(N.EQ.2) GO TO 600
00074400        DO 590 I=1,NVA
00074500        K=N2
00074600    550 CR=-A(K+1,K)*DCONJG(A(K,K))*W(K,2)
00074700        IF(DREAL(CR).EQ.0.0 .AND. DIMAG(CR).EQ.0.0) GO TO 580
00074800        CR = 1.0D0/CR
00074900        CS=(0.,0.)
00075000        DO 560 J=K+1,N
00075100    560 CS=CS+DCONJG(A(J,K))*V(J,I)
00075200        CR=CR*CS
00075300        DO 570 J=K+1,N
00075400    570 V(J,I)=V(J,I)-CR*A(J,K)
00075500    580 K=K-1
00075600        IF(K.GE.1) GO TO 550
00075700    590 CONTINUE
00075800    600 CONTINUE
00075900 C   NORMALIZE EIGENVECTORS
00076000 C   NORMALIZE AS MAXIMUM ELEMENT=1
00076100        DO 620 I=1,NVA
00076200        T=DABS(DREAL(V(1,I)))+DABS(DIMAG(V(1,I)))
00076300        K=1
00076400        DO 610 J=2,N
00076500        R=DABS(DREAL(V(J,I)))+DABS(DIMAG(V(J,I)))
00076600        IF(T.GE.R) GO TO 610
00076700        T=R
00076800        K=J
00076900    610 CONTINUE
00077000        CR = 1.0D0/V(K,I)
00077100        DO 620 J=1,N
00077200    620 V(J,I)=V(J,I)*CR
00077300        IF(NV.LT.0) GO TO 900
00077400 C   ORTHOGONALIZE AS NORM=1
00077500        DO 680 I=1,NVA
00077600        IF(I.EQ.1 .OR. DABS(E(I)-E(I-1)).GT.EPS) GO TO 650
00077700        DO 640 J=M,I-1
00077800        CS=(0.,0.)
00077900        DO 630 K=1,N
00078000    630 CS = CS + DCONJG(V(K,J))*V(K,I)
00078100        DO 640 K=1,N
00078200    640 V(K,I)=V(K,I)-CS*V(K,J)
00078300        GO TO 660
00078400    650 M=I
```

```
00078500 C  NORMALIZE AS NORM=1
00078600    660 S=0.
00078700        DO 670 J=1,N
00078800    670 S=S+DREAL(V(J,I))**2+DIMAG(V(J,I))**2
00078900        T = QSQRT(1.0Q0/S)
00079000        DO 680 J=1,N
00079100    680 V(J,I)=V(J,I)*T
00079200    900 RETURN
00079300 C  PRINT ERROR MESSAGE
00079400    910 WRITE(6,1000) N,NE
00079500        GO TO 900
00079600    920 WRITE(6,1100) NV,NE,N,N1
00079700        GO TO 900
00079800   1000 FORMAT(1H0,'(SUBR. DEIGCH) N=',I5,',NE=',I5,'  N SHOULD BE GREA
00079900       1 THAN ZERO AND NE SHOULD BE NON-ZERO. RETURN WITH NO CALCULATIO
00080000       2 ')
00080100   1100 FORMAT(1H0,'(SUBR. DEIGCH) NV=',I5,',NE=',I5,',N=',I5,',N1=',I5
00080200       1' NV,NE,N,N1 SHOULD SATISFY THE FOLLOWING INEQUALITIES, INVI <=
00080300       2EI <= N <= N1 .' /1H ,'RETURN WITH NO CALCULATION.' )
00080400        END
00080500        SUBROUTINE MATOUT(MATOP)
00080600        COMPLEX*16 ZS,ZH,PM,RM
00080700        REAL*8 V
00080800        COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
00080900        COMMON/COM01/ ZS(40,40),ZH(40,40)
00081000        COMMON/COM02/ PM(40,40),RM(40,40)
00081100        COMMON/COM03/ V(40),W(20,40)
00081200        COMMON/NATOMS/ NATOMS,N,NK
00081300        REAL*8 A(2,40,40),B(2,40,40),C(2,40,40)
00081400        EQUIVALENCE (A(1,1,1),ZS(1,1))
00081500        EQUIVALENCE (B(1,1,1),PM(1,1))
00081600        EQUIVALENCE(C(1,1,1),ZH(1,1))
00081700 C      IF(MATOP.NE.3.AND.MATOP.NE.5) GO TO 90
00081800        DO 80 M=1,N,12
00081900        K=M+11
00082000        IF (K.LE.N) GO TO 30
00082100    20  K=N
00082200    30  CONTINUE
00082300        IF(MATOP.NE.3) GO TO 11
00082400        WRITE(6,10) (V(J),J=M,K)
00082500    10  FORMAT(7X,12F9.4)
00082600    11  CONTINUE
00082700        WRITE(6,40) (J,J=M,K)
00082800    40  FORMAT(//,7X,12(4X,I2,3X),//)
00082900        DO 60 I=1,N
00083000        GO TO (41,42,43,44,45),MATOP
00083100    41  WRITE(6,50) I,(S(I,J),J=M,K) ;GO TO 60
00083200    42  WRITE(6,50) I,(A(1,I,J),J=M,K)
00083300        WRITE(6,50) I,(A(2,I,J),J=M,K) ;GO TO 60
00083400    43  WRITE(6,50) I,(A(1,I,J),J=M,K)
00083500        WRITE(6,50) I,(A(2,I,J),J=M,K) ;GO TO 60
00083600    44  WRITE(6,50) I,(C(1,I,J),J=M,K)
00083700        WRITE(6,50) I,(C(2,I,J),J=M,K) ;GO TO 60
00083800    45  WRITE(6,50) I,(B(1,I,J),J=M,K)
00083900        WRITE(6,50) I,(B(2,I,J),J=M,K) ;GO TO 60
00084000    50  FORMAT(1X,I2,4X,50(F9.4))
00084100    60  CONTINUE
00084200        WRITE(6,70)
00084300    70  FORMAT(//)
00084400    80  CONTINUE
00084500    90  RETURN
00084600        END
00084700        SUBROUTINE SOLUT
00084800        COMPLEX*16 S,H,PM,RM,SUM,DEN
00084900        REAL*8 EI,WW,V
00085000        DIMENSION LW(40),EI(40),WW(40,7)
```

```
00085100 C-------***    SOLUTION OF SECULAR DETERMINANT    ***
00085200         COMMON/COM01/ S(40,40),H(40,40)
00085300         COMMON/COM02/ PM(40,40),RM(40,40)
00085400         COMMON/COM03/ V(40),E(20,40)
00085500         COMMON/COM19/ IRET,ISTOP,NNAT
00085600         COMMON/NATOMS/ NATOMS,N,NK
00085700         DEBUG,SUBCHK
00085800 C       COMPLEX*8 ZH(24,24)
00085900 C       DO 3 I=1,24
00086000 C       DO 3 J=1,24
00086100 C     3 ZH(I,J)=H(I,J)
00086200         DO 2 I=1,N
00086300         DO 2 J=1,N
00086400       2 RM(I,J)=S(I,J)
00086500 C-----
00086600         EPS=1.E-15
00086700         CALL DEIGCH(S,N,40,-N,N,EPS,WW,LW,EI,PM)
00086800 C-----
00086900         DO 10 I=1,N
00087000         IF(EI(I) .LE.0.0) GO TO 41
00087100 C-----
00087200      10 V(I)=1.0/SQRT(EI(I) )
00087300         DO 13 I=1,N
00087400         DO 12 J=1,N
00087500      12 S(I,J)=PM(I,J)*V(J)
00087600      13 CONTINUE
00087700 C-----
00087800         DO 16 I=1,N
00087900         DO 15 J=1,N
00088000         PM(I,J)=0.0
00088100         DO 14 K=1,N
00088200      14 PM(I,J)=PM(I,J)+CONJG(S(K,I))*H(K,J)
00088300      15 CONTINUE
00088400      16 CONTINUE
00088500         DO 19 I=1,N
00088600         DO 18 J=1,N
00088700         H(I,J)=0.0
00088800         DO 17 K=1,N
00088900      17 H(I,J)=H(I,J)+PM(I,K)*S(K,J)
00089000      18 CONTINUE
00089100      19 CONTINUE
00089200 C-----
00089300 C       DO 3333 I=1,N
00089400 C       DO 3333 J=I,N
00089500 C3333 H(J,I)=CONJG(H(I,J))
00089600         CALL DEIGCH(H,N,40,-N,N,EPS,WW,LW,V,PM)
00089700 C-----
00089800         DO 21 I=1,N
00089900      21 E(NK-1,I)=V(I)
00090000         DO 52 I=1,N
00090100         DO 51 J=1,N
00090200         H(I,J)=0.0
00090300         DO 50 K=1,N
00090400      50 H(I,J)=H(I,J)+S(I,K)*PM(K,J)
00090500      51 CONTINUE
00090600      52 CONTINUE
00090700 C-----
00090800         DO 25 I=1,N
00090900         SUM=0.0
00091000         DO 24 J=1,N
00091100         DO 23 K=1,N
00091200      23 SUM=SUM+CONJG(H(J,I))*H(K,I)*RM(J,K)
00091300      24 CONTINUE
00091400         DEN=SQRT(SUM)
00091500         DO 26 L=1,N
00091600      26 S (L,I)=H(L,I)/DEN
```

```
00091700      25 CONTINUE
00091800 C       DO 31 I=1,N
00091900 C       DO 31 J=1,N
00092000 C    31 H(I,J)=0.
00092100 C       DO 32 I=1,N
00092200 C       DO 32 J=1,N
00092300 C       DO 32 K=1,N
00092400 C    32 H(I,J)=H(I,J)+(ZH(I,K)-V(J)*RM(I,K))*S(K,J)
00092500 C       WRITE(6,33)
00092600 C    33 FORMAT(' (H-ES)X=')
00092700 C       CALL MATOUT(4)
00092800         GO TO 60
00092900      41 WRITE(6,42)
00093000      42 FORMAT(' OVERLAP MATRIX IS NON-POSITIVE. STOP')
00093100         STOP
00093200      60 RETURN
00093300         END
00093400         PROGRAM PLBAND
00093500         DIMENSION W(20,40),AK(40),V(40)
00093600         READ(20,51) N,MNK1,VECK1
00093700      51 FORMAT(2I10,F10.4)
00093800         READ(20,50) ((W(I,J),J=1,N),I=1,MNK1)
00093900      50 FORMAT(24F10.4)
00094000         DO 1 I=1,MNK1
00094100       1 AK(I)=(I-1)*VECK1/(MNK1-1)
00094200         DO 4 I=1,MNK1
00094300         DO 4 J=1,N
00094400           IF(W(I,J).GE.40.) W(I,J)=40.
00094500       4 CONTINUE
00094600         NK1=MNK1+1 ;NK2=MNK1+2
00094700         AK(NK1)=0. ;AK(NK2)=-0.075 ;V(NK1)=-40. ;V(NK2)=2.
00094800 C       WRITE(6,100) N,MNK1,VECK1,((W(I,J),J=1,N),I=1,MNK1)
00094900 C 100 FORMAT(1H ,2I10,F10.4,/1H ,(12F10.4,/1H ))
00095000         CALL PLOTS(999.,999.,'EHT SP SIO2')
00095100         CALL PLOT(2.,15.,-3)
00095200         AXL=VECK1/0.075
00095300         CALL AXIS(0.,0.,'K',-1,AXL,-90.,AK(NK1),0.075)
00095400         CALL AXIS(0.,0.,'ENERGY',6,40.,0.,V(NK1),V(NK2))
00095500         DO 2 I=1,N
00095600         DO 3 J=1,MNK1
00095700       3 V(J)=W(J,I)
00095800         CALL LINE(V,AK,MNK1,1,0,0)
00095900         WRITE(6,101) I,(V(K),K=1,MNK1)
00096000     101 FORMAT(1H ,I10,/1H ,(12F10.4/1H ))
00096100       2 CONTINUE
00096200         CALL PLOTV
00096300         STOP ;END

BER  BAND        OF  A1402.S.FORT  LIST END
```

BER BETA JF A1402.S.FORT

```
00000100       SUBROUTINE MATOUT(MATOP)
00000200       COMPLEX*16 ZS,ZH,PM,RM
00000300       REAL*8 V
00000400       COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
00000500       COMMON/COM01/ ZS(40,40),ZH(40,40)
00000600       COMMON/COM02/ PM(40,40),RM(40,40)
00000700       COMMON/COM03/ V(40),W(20,40)
00000800       COMMON/NATOMS/ NATOMS,N,NK
00000900       REAL*8 A(2,40,40),B(2,40,40),C(2,40,40)
00001000       EQUIVALENCE (A(1,1,1),ZS(1,1))
00001100       EQUIVALENCE (B(1,1,1),PM(1,1))
00001200       EQUIVALENCE(C(1,1,1),ZH(1,1))
00001300 C     IF(MATOP.NE.3.AND.MATOP.NE.5) GO TO 90
00001400       DO 80 M=1,N,12
00001500       K=M+11
00001600       IF (K.LE.N) GO TO 30
00001700    20 K=N
00001800    30 CONTINUE
00001900       IF(MATOP.NE.3) GO TO 11
00002000       WRITE(6,10) (V(J),J=M,K)
00002100    10 FORMAT(7X,12F9.4)
00002200    11 CONTINUE
00002300       WRITE(6,40) (J,J=M,K)
00002400    40 FORMAT(//,7X,12(4X,I2,3X),//)
00002500       DO 60 I=1,N
00002600       GO TO (41,42,43,44,45),MATOP
00002700    41 WRITE(6,50) I,(S(I,J),J=M,K) ;GO TO 60
00002800    42 WRITE(6,50) I,(A(1,I,J),J=M,K)
00002900       WRITE(6,50) I,(A(2,I,J),J=M,K) ;GO TO 60
00003000    43 WRITE(6,50) I,(A(1,I,J),J=M,K)
00003100       WRITE(6,50) I,(A(2,I,J),J=M,K) ;GO TO 60
00003200    44 WRITE(6,50) I,(C(1,I,J),J=M,K)
00003300       WRITE(6,50) I,(C(2,I,J),J=M,K) ;GO TO 60
00003400    45 WRITE(6,50) I,(B(1,I,J),J=M,K)
00003500       WRITE(6,50) I,(B(2,I,J),J=M,K) ;GO TO 60
00003600    50 FORMAT(1X,I2,4X,50(F9.4))
00003700    60 CONTINUE
00003800       WRITE(6,70)
00003900    70 FORMAT(//)
00004000    80 CONTINUE
00004100    90 RETURN
00004200       END
00004300       PROGRAM BETA
00004400       COMMON/COM03/ V(40),W(20,40)
00004500       COMMON/VECK/VECK(3)
00004600 C--------GENERATION OF MATRIX VOIP AND AZETA---------
00004700       COMMON/VOIP/ VOIP(4,6),AZETA(4,6),CONST,OM1,OM2,VP(4,6),AZP(4,
00004800       COMMON/POS/ POS(3,6),F(3,7),BONDL,AU,BAU,NC(18),LC(10),MC(10)
00004900       COMMON/COM17/TOEL(30),EN(40)
00005000       COMMON/COM13/ TOP(10,10)
00005100       COMMON/INFO/ AN(35),NOUT
00005200       COMMON/NATOMS/ NATOMS,N,NK
00005300       COMMON /CON/CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX,CSS,CSX,CXS,CX
00005310       REAL*8 VM(40)
00005400       REAL*8 V
00005500       INTEGER AN
00005510       DIMENSION AKM(3,3)
00005520       DATA AKM/5*0.,0.4771,3*0.2236/
```

```
00005600       DATA AN/14,14,8,8,8,8/,NATOMS/6/
00005700       DATA EN/16*2./,TOEL/4.,4.,6.,6.,6.,6./,OM1,OM2/.5,.5/
00005800       DATA CONST/1.75/
00005900 C     DATA VOIP/14.95,7.78,2.05,5*0.,
00006000 C    *           32.38,15.84/
00006100       DATA VOIP/12.88,7.08,2.05,5*0.,
00006200      *           32.38,19.58/
00006300       DATA AZETA/1.38,1.38,1.38,5*0.,
00006400      *            2.28,2.28/
00006500 C     DATA AZETA/1.56,1.56,1.38,5*0.,2.1,2.1/
00006600 C     DATA AZETA/1.6344,1.4284,1.38,5*0.,2.246,2.227/
00006700       DATA POS/0.,0.,0.,2.,2.,2.,1.,1.,1.,
00006800      *         3.,3.,1.,1.,3.,3.,3.,1.,3./
00006900       DATA F/0.,0.,0.,0.,1.,1.,1.,0.,1.,1.,1.,0.,
00007000      *                0.,-1.,1.,-1.,0.,1.,1.,-1.,0./
00007100       DATA VECK/3*0./
00007200       DEBUG,SUBCHK
00007300       BONDL=1.61 ;AU=.529167 ;BAU=BONDL/AU
00007400       B4=BONDL*4./SQRT(3.)/AU ;B1=B4/4.
00007500  1000 READ(5,14) VOIP(1,1),VOIP(2,1),VOIP(1,3),VOIP(2,3)
00007600    14 FORMAT(4F10.0)
00007700       WRITE(6,15) VOIP(1,1),VOIP(2,1),VOIP(1,3),VOIP(2,3)
00007800    15 FORMAT(' VOIP=',4F10.3)
00007810       READ(5,14) AZETA(1,1),AZETA(2,1),AZETA(1,3),AZETA(2,3)
00007820       WRITE(6,16) AZETA(1,1),AZETA(2,1),AZETA(1,3),AZETA(2,3)
00007830    16 FORMAT(' AZETA=',4F10.3)
00007900       READ(5,5) CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX
00008000     5 FORMAT(3F10.0)
00008600       WRITE(6,7) CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX
00008700     7 FORMAT(' CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX=',6F10.3)
00008800       READ(5,8) CSS,CSX,CXS,CXX
00008810     8 FORMAT(4F10.0)
00008900       WRITE(6,13) CSS,CSX,CXS,CXX
00009000    13 FORMAT(' CSS,CSX,CXS,CXX=',4F10.3)
00009010       READ(5,9) MODE,MOD1,MNK,ISOL,ITDP,IGRAPH,IHAM
00009020     9 FORMAT(7I2)
00009030       IF(MODE.EQ.0) STOP
00009040       REWIND 30
00009100       DO 22 I=1,3
00009200       DO 22 J=1,7
00009300    22 F(I,J)=F(I,J)*B4
00009400       DO 4 I=1,3
00009500       DO 4 J=1,6
00009600     4 POS(I,J)=POS(I,J)*B1
00009700 C---------GENERATION OF MATRIX VOIP AND AZETA---------
00009800       DO 10 I=1,4
00009900         VOIP(I,2)=VOIP(I,1)
00010000         AZETA(I,2)=AZETA(I,1)
00010100       DO 10 J=4,6
00010200         VOIP(I,J)=VOIP(I,3)
00010300         AZETA(I,J)=AZETA(I,3)
00010400    10 CONTINUE
00010500       DO 11 I=1,4
00010600       DO 11 J=1,6
00010700         VP(I,J)=VOIP(I,J)
00010800    11   AZP(I,J)=AZETA(I,J)
00011000       CALL COEFFT
00011100       CALL INTGRL
00011200       IF(IHAM.EQ.0) GO TO 140
00011300 C-------- WRITING OUT S-----
00011400       WRITE(6,350)
00011500   350 FORMAT(1H1,1X,23HOVERLAP INTEGRAL MATRIX)
00011600       CALL MATOUT(1)
00011700   140 CONTINUE
00011800       NK=1
00011900 C------SCANNING OF K VECTOR-------
```

```
00012000      2 CONTINUE
00012100        IF(NK-1.GT.MNK) GO TO 1
00012200        GO TO (161,162) ,MODE
00012300    161 INIK=3 ;LASK=1 ;GO TO 170
00012400    162 INIK=1 ;LASK=2 ;GO TO 170
00012500    170 CONTINUE
00012510        AK1=0.
00012520        DO 180 I=1,3
00012530    180 AK1=AK1+(AKM(I,INIK)-AKM(I,LASK))**2
00012540        AK1=SQRT(AK1)
00012550        DO 181 I=1,3
00012560    181 VECK(I)=(AKM(I,INIK)*(MNK-NK+1.0001)
00012570       *          +AKM(I,LASK)*(NK-1.))/(MNK+0.0001)
00012580        NK=NK+1
00012590        CALL SETSH
00012600 C      WRITE(6,351) NK
00012610 C  351 FORMAT(' BLOCH OVERLAP MATRIX AT NK=',I5)
00012620 C      CALL MATOUT(2)
00012630 C      WRITE(6,352) NK
00012640 C  352 FORMAT(' BLOCH HAMILTONIAN MATRIX AT NK=',I5)
00012650 C      CALL MATOUT(4)
00012660        CALL SOLUT
00012670        WRITE(6,12) (VECK(I),I=1,3),(V(I),I=1,N)
00012680     12 FORMAT(' EIGENVALUES AT NK=',3F10.3,'ARE',/(6F9.4))
00012690 C------DIFFERENCE-------
00012700        GO TO (40,41,42,42) ,MOD1
00012710     41 WRITE(30) (V(I),I=1,N) ;GO TO 40
00012720     42 READ(30) (VM(I),I=1,N)
00012730        DO 43 I=1,N
00012740     43 VM(I)=V(I)-VM(I)
00012750        WRITE(6,102) NK,(VM(I),I=1,N)
00012760    102 FORMAT( ' DIFFERENCE AT NK=',I5,/(6F9.4))
00012770        IF(MOD1.LE.3) GO TO 40
00012780        WRITE(40) (VM(I),I=1,N)
00012790     40 CONTINUE
00012800 C------ISOL-----------
00012810        IF(ISOL.EQ.0) GO TO 103
00012820        WRITE(6,353) NK
00012830    353 FORMAT(' SOLUTION AT NK=',I5)
00012840        CALL MATOUT(3)
00012850    103 CONTINUE
00012860 C------TDP-----------
00012870        IF(ITDP.EQ.0) GO TO 44
00012880        CALL DENSIT
00012890     44 CONTINUE
00012900        GO TO 2
00012910      1 CONTINUE
00012920        IF(IGRAPH.EQ.0) GO TO 1000
00012930        MNK1=MNK+1
00012940        WRITE(20,51) N,MNK1,AK1
00012950     51 FORMAT(2I10,F10.4)
00012960        WRITE(20,50) ((W(I,J),J=1,N),I=1,MNK1)
00012970     50 FORMAT(24F10.4)
00013240        STOP ;END
00015800        SUBROUTINE INTGRL
00015900        IMPLICIT REAL*4(A-H,O-Z)
00016000 C-----ATOMIC INTEGRALS FOR CNDO CALCULATIONS
00016100        COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
00016200        COMMON/INFO/ AN(35),NOUT
00016300        COMMON/NATOMS/ NATOMS,N,NK
00016400        COMMON/INFO1/CZ(35),U(80),ULIM(35),LLIM(35),ULK,ULL,ANL,ANK
00016500        COMMON/GAB/T(9,9),PAIRS(9,9),TEMP(9,9),C1(3),C2(3)
00016600        COMMON/AUXINT/A(17),B(17)
00016700        COMMON /SOUTER/SOUTER(10,10,3,7),NZS(2,3,7)
00016800        COMMON/VOIP/ VOIP(4,6),AZETA(4,6),CONST,OM1,OM2,VP(4,6),AZP(4,
00016900        COMMON/POS/ POS(3,6),F(3,7),BONDL,AU,BAU,NC(18),LC(10),MC(10)
```

```
00017000       DIMENSION P(80,80)
00017100       DIMENSION E(3),Q(40)
00017200       EQUIVALENCE (P(1),Y(1))
00017300       INTEGER AN,ULIM,ULK,JLL,CZ,U,ANL,ANK
00017400 C-----DETERMINATION OF SIZE OF AO BASIS IN AND CORE CHARGE CZ
00017500       DEBUG,SUBCHK
00017600       N=0
00017700 C      WRITE(6,1100) ((POS(I,J),J=1,6),I=1,3)
00017800 C1100 FORMAT(1H ,'POS='/3(6F10.4/))
00017900 C      WRITE(6,1101) ((F(I,J),J=1,7),I=1,3)
00018000 C1101 FORMAT(1H ,'F='/3(7F10.4/))
00018100 C      WRITE(6,1102) ((VOIP(I,J),J=1,6),I=1,4)
00018200 C1102 FORMAT(1H ,'VOIP='/4(6F10.4/))
00018300 C      WRITE(6,1103) ((AZETA(I,J),J=1,6),I=1,4)
00018400 C1103 FORMAT(1H ,'AZETA='/4(6F10.4/))
00018500       DO 60 I=1,NATOMS
00018600       LLIM(I) = N+1
00018700       K=1
00018800       IF (AN(I).LT.11) GO TO 20
00018900    10 N=N+4
00019000       CZ(I)=AN(I)-10
00019100       GO TO 50
00019200    20 IF (AN(I).LT.3) GO TO 40
00019300    30 N=N+4
00019400       CZ(I) = AN(I)-2
00019500       GO TO 50
00019600    40 N=N+1
00019700       CZ(I)= AN(I)
00019800    50 CONTINUE
00019900       ULIM(I) = N
00020000    60 CONTINUE
00020100 C-----ASSIGNMENT OF ANGULAR MOMENTUM QUANTUM NOS. TO ATOMIC ORBITALS
00020200       LC(1)=0
00020300       LC(2)=1
00020400       LC(3)=1
00020500       LC(4)=1
00020600       LC(5)=2
00020700       LC(6)=2
00020800       LC(7)=2
00020900       LC(8)=2
00021000       LC(9)=2
00021100       MC(1)=0
00021200       MC(2)=1
00021300       MC(3)=-1
00021400       MC(4)=0
00021500       MC(5)=0
00021600       MC(6)=1
00021700       MC(7)=-1
00021800       MC(8)=2
00021900       MC(9)=-2
00022000 C-----FILL U ARRAY---U(J) IDENTIFIES THE ATOM TO WHICH ORBITAL J IS
00022100 C-----ATTACHED E.G. ORBITAL 32 ATTACHED TO  ATOM 7, ETC.
00022200 C-----ASSIGNMENT OF ORBITAL EXPONENTS TO ATOMS BY SLATERS RULES
00022300       MC(8)=2 ;MC(14)=3
00022400       DO 92 K=1,NATOMS
00022500       LLK=LLIM(K)
00022600       JLK=ULIM(K)
00022700       ANK=AN(K)
00022800       NORBK=ULK-LLK+1
00022900       DO 92 I=1,NORBK
00023000       LLKP=LLK+I-1
00023100       LCZETA=LC(I)+1
00023200       Q(LLKP)=AZETA(LCZETA,K)
00023300       U(LLKP)=K
00023400    92 CONTINUE
00023500 C-----STEP THRU PAIRS OF ATOMS
```

```
00023600         DO 320 IF=1,7
00023700         NIF=0
00023800         DO 320 K=1,NATOMS
00023900         DO 320 L=K,NATOMS
00024000         DO 100 I=1,3
00024100         C1(I)=POS(I,K)
00024200     100 C2(I)=POS(I,L)+F(I,IF)
00024300 C-----CALCULATE UNIT VECTOR ALONG INTERATOM AXIS,E
00024400         CALL RELVEC(R,E,C1,C2)
00024500         IF(R.GT.2*BAU+0.1) GO TO 320
00024600         IF(IF.EQ.1) GO TO 102
00024700         NIF=NIF+1
00024800         NZS(1,NIF,IF)=K
00024900         NZS(2,NIF,IF)=L
00025000     102 CONTINUE
00025100         LLK = LLIM(K)
00025200         LLL = LLIM(L)
00025300         JLK = ULIM(K)
00025400         JLL = ULIM(L)
00025500         NORBK=ULK-LLK+1
00025600         NORBL=ULL-LLL+1
00025700         ANK=AN(K)
00025800         ANL=AN(L)
00025900 C-----LOOP THRU PAIRS OF BASIS FUNCTIONS, ONE ON EACH ATOM
00026000         DO 200 I=1,NORBK
00026100         DO 200 J=1,NORBL
00026200         IF(K.EQ.L.AND.IF.EQ.1) GO TO 160
00026300     110 IF(MC(I).NE.MC(J)) GO TO 150
00026400     120 IF(MC(I).LT.0) GO TO 140
00026500         LLKP=LLK+I-1 ;LLLP=LLL+J-1
00026600     130 PAIRS(I,J)= SQRT((Q(LLKP)*R)**(2*NC(ANK)+1)*(Q(LLLP)*R)**(2*NC
00026700        1)+1)/(FACT(2*NC(ANK))*FACT(2*NC(ANL))))*(-1.D0)**(LC(J)+MC(J))
00026800        2*SS(NC(ANK),LC(I),MC(I),NC(ANL),LC(J),Q(LLKP)*R,Q(LLLP)*R)
00026900         GO TO 190
00027000     140 PAIRS(I,J)=PAIRS(I-1,J-1)
00027100         GO TO 190
00027200     150 PAIRS(I,J)=0.0D0
00027300         GO TO 190
00027400     160 IF (I.EQ.J) GO TO 170
00027500     180 PAIRS(I,J)=0.0D0
00027600         GO TO 190
00027700     170 PAIRS(I,J)=1.0D0
00027800     190 CONTINUE
00027900     200 CONTINUE
00028000         LCULK=LC(NORBK)
00028100         LCULL=LC(NORBL)
00028200         MAXL=MAX0(LCULK,LCULL)
00028300         IF(R.GT.0.000001D0) GO TO 220
00028400     210 GO TO 250
00028500 C-----ROTATE INTEGRALS FROM DIATOMIC BASIS TO MOLECULAR BASIS
00028600     220 CALL HARMTR(T,MAXL,E)
00028700         DO 230 I=1,NORBK
00028800         DO 230 J=1,NORBL
00028900         TEMP(I,J) = 0.D0
00029000         DO 230 KK=1,NORBL
00029100         TEMP(I,J) = TEMP(I,J)+T(J,KK)*PAIRS(I,KK)
00029200     230 CONTINUE
00029300         DO 240 I=1,NORBK
00029400         DO 240 J=1,NORBL
00029500         PAIRS(I,J) = 0.D0
00029600         DO 240 KK=1,NORBK
00029700         PAIRS(I,J) = PAIRS(I,J)+T(I,KK)*TEMP(KK,J)
00029800     240 CONTINUE
00029900 C-----FILL S MATRIX
00030000     250 CONTINUE
00030100         IF(IF.NE.1) GO TO 262
```

```
00030200         DO 260 I=1,NORBK
00030300         LLKP=LLK+I-1
00030400         DO 260 J=1,NORBL
00030500         LLLP=LLL+J-1
00030600     260 S(LLKP,LLLP)=PAIRS(I,J)
00030700         GO TO 320
00030800     262 CONTINUE
00030900         DO 264 I=1,NORBK
00031000         DO 264 J=1,NORBL
00031100     264 SOUTER(I,J,NIF,IF)=PAIRS(I,J)
00031200     320 CONTINUE
00031300     330 CONTINUE
00031400         RETURN
00031500         END
00031600         SUBROUTINE SETSH
00031700         IMPLICIT COMPLEX*16 (Z)
00031800         COMMON/COM01/ ZS(40,40),ZH(40,40)
00031900         COMMON/VOIP/ VOIP(4,6),AZETA(4,6),CONST,OM1,OM2,VP(4,6),AZP(4,
00032000         COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
00032100         COMMON/INFO/ AN(35),NOUT
00032200         COMMON/INFO1/CZ(35),U(80),ULIM(35),LLIM(35),ULK,ULL,ANL,ANK
00032300         COMMON /SOUTER/SOUTER(10,10,3,7),NZS(2,3,7)
00032400         COMMON/NATOMS/ NATOMS,N,NK
00032500         COMMON/VECK/VECK(3)
00032600         COMMON/POS/ POS(3,6),F(3,7),BONDL,AU,BAU,NC(18),LC(10),MC(10)
00032700         COMMON /CON/CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX,CSS,CSX,CXS,CX
00032800         INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
00032900         REAL*4 Z
00033000         DEBUG,SUBCHK
00033100 C       WRITE(6,100) (((NZS(I,J,K),J=1,3),I=1,2),K=1,7)
00033200 C 100 FORMAT(1H0,'NZS=',/1H ,2(3I2/1H ))
00033300 C       WRITE(6,101) ((((SOUTER(I,J,K,L),J=1,4),I=1,4),K=1,3),L=1,7)
00033400 C 101 FORMAT(1H0,'SOUTER=',/1H ,4(4F10.4/1H ))
00033500         DO 2 I=1,N
00033600         DO 2 J=I,N
00033700       2 ZS(I,J)=S(I,J)
00033800         DO 3 IF=2,7
00033900         ZTHETA=0.
00034000         DO 4 I=1,3
00034100       4 ZTHETA=ZTHETA+VECK(I)*F(I,IF)*(0.,1.)
00034200         ZEXP=EXP(ZTHETA)
00034300 C       WRITE(6,102) ZTHETA,ZEXP,(F(I,IF),I=1,3),IF
00034400 C 102 FORMAT(1H ,'ZTHETA,ZEXP=',4F10.4/1H ,'F,IF=',3F10.4,I3)
00034500 C--------SET UP S MATRIX FOR A CERTAIN K--------
00034600         DO 5 NIF=1,3
00034700         IF(NZS(1,NIF,IF).EQ.0.) GO TO 3
00034800         K=NZS(1,NIF,IF)
00034900         L=NZS(2,NIF,IF)
00035000         LLK=LLIM(K)
00035100         LLL=LLIM(L)
00035200         JLK=ULIM(K)
00035300         JLL=ULIM(L)
00035400         NORBK=ULK-LLK+1
00035500         NORBL=ULL-LLL+1
00035600         DO 6 I=1,NORBK
00035700         DO 6 J=1,NORBL
00035800           LLKP=LLK+I-1
00035900           LLLP=LLL+J-1
00036000       6   ZS(LLKP,LLLP)=ZS(LLKP,LLLP)+SOUTER(I,J,NIF,IF)*ZEXP
00036100       5 CONTINUE
00036200       3 CONTINUE
00036300 C--------CALCULATION OF H MATRIX--------
00036400         DO 8 K=1,NATOMS
00036500         DO 8 L=K,NATOMS
00036600           LLK=LLIM(K)
00036700           LLL=LLIM(L)
```

```
00036800           ULK=ULIM(K)
00036900           ULL=ULIM(L)
00037000           NORBK=ULK-LLK+1
00037100           NORBL=ULL-LLL+1
00037200           DO 9 I=1,NORBK
00037300           DO 9 J=1,NORBL
00037400             LLKP=LLK+I-1
00037500             LLLP=LLL+J-1
00037600             LCI=LC(I)+1
00037700             LCJ=LC(J)+1
00037800             IF(LLKP.EQ.LLLP) GO TO 10
00037900             LLKPU=U(LLKP) ;LLLPU=U(LLLP)
00038000             IF(LLKPU.LE.2.AND.LLLPU.LE.2) GO TO 11
00038100             IF(LLKPU.GE.3.AND.LLLPU.GE.3) GO TO 12
00038200             IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 16
00038300             IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 17
00038310             IF((LLKPU.LE.2.AND.LCI.EQ.1).OR.
00038320      *         (LLLPU.LE.2.AND.LCJ.EQ.1)) GO TO 18
00038330             CONST=CXS ;GO TO 13
00038340     18      CONST=CSX ;GO TO 13
00038500     16      CONST=CSS ;GO TO 13
00038600     17      CONST=CXX ;GO TO 13
00038610     11      IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 22
00038620             IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 23
00038630             CONST=CSSSX ;GO TO 13
00038640     22      CONST=CSSSS ;GO TO 13
00038650     23      CONST=CSSXX ;GO TO 13
00038710     12      IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 20
00038720             IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 21
00038730             CONST=COOSX ;GO TO 13
00038740     20      CONST=COOSS ;GO TO 13
00038750     21      CONST=COOXX ;GO TO 13
00038900     13      CONTINUE
00039000             ZH(LLKP,LLLP)=-ZS(LLKP,LLLP)*(VOIP(LCI,K)+VOIP(LCJ,L))*
00039100      *                CONST/2.
00039200           GO TO 9
00039300     10  ZH(LLKP,LLKP)=-VOIP(LCI,K)
00039400      9  CONTINUE
00039500      8  CONTINUE
00039600 C         CALL MATOUT(4)
00039700 C---------SYMMETRIZATION OF S AND H MATRIX--------
00039800           DO 7 I=1,N
00039900           DO 7 J=I,N
00040000             ZS(J,I)=CONJG(ZS(I,J))
00040100             ZH(J,I)=CONJG(ZH(I,J))
00040200      7  CONTINUE
00040300 C         DO 15 I=1,N
00040400 C         DO 14 J=1,N
00040500 C           ZS(I,J)=0.
00040600 C   14  CONTINUE
00040700 C           ZS(I,I)=1.
00040800 C   15  CONTINUE
00040900           RETURN ;END
00041000           SUBROUTINE SOLUT
00041100           COMPLEX*16 S,H,PM,RM,SUM,DEN
00041200           REAL*8 EI,WW,V
00041300           DIMENSION LW(40),EI(40),WW(40,7)
00041400 C---------***    SOLUTION OF SECULAR DETERMINANT   ***
00041500           COMMON/COM01/ S(40,40),H(40,40)
00041600           COMMON/COM02/ PM(40,40),RM(40,40)
00041700           COMMON/COM03/ V(40),W(20,40)
00041800           COMMON/COM19/ IRET,ISTOP,NNAT
00041900           COMMON/NATOMS/ NATOMS,N,NK
00042000           DEBUG,SUBCHK
00042100 C         COMPLEX*8 ZH(24,24)
00042200 C         DO 3 I=1,24
```

```
00042300 C       DO 3 J=1,24
00042400 C    3 ZH(I,J)=H(I,J)
00042500         DO 2 I=1,N
00042600         DO 2 J=1,N
00042700      2 RM(I,J)=S(I,J)
00042800 C-----
00042900         EPS=1.E-15
00043000         CALL DEIGCH(S,N,40,-N,N,EPS,WW,LW,EI,PM)
00043100 C-----
00043200         DO 10 I=1,N
00043300         IF(EI(I) .LE.0.0) GO TO 41
00043400 C-----
00043500     10 V(I)=1.0/SQRT(EI(I) )
00043600         DO 13 I=1,N
00043700         DO 12 J=1,N
00043800     12 S(I,J)=PM(I,J)*V(J)
00043900     13 CONTINUE
00044000 C-----
00044100         DO 16 I=1,N
00044200         DO 15 J=1,N
00044300         PM(I,J)=0.0
00044400         DO 14 K=1,N
00044500     14 PM(I,J)=PM(I,J)+CONJG(S(K,I))*H(K,J)
00044600     15 CONTINUE
00044700     16 CONTINUE
00044800         DO 19 I=1,N
00044900         DO 18 J=1,N
00045000         H(I,J)=0.0
00045100         DO 17 K=1,N
00045200     17 H(I,J)=H(I,J)+PM(I,K)*S(K,J)
00045300     18 CONTINUE
00045400     19 CONTINUE
00045500 C-----
00045600 C       DO 3333 I=1,N
00045700 C       DO 3333 J=I,N
00045800 C3333 H(J,I)=CONJG(H(I,J))
00045900         CALL DEIGCH(H,N,40,-N,N,EPS,WW,LW,V,PM)
00046000 C-----
00046100         DO 21 I=1,N
00046200     21 V(NK-1,I)=V(I)
00046300         DO 52 I=1,N
00046400         DO 51 J=1,N
00046500         H(I,J)=0.0
00046600         DO 50 K=1,N
00046700     50 H(I,J)=H(I,J)+S(I,K)*PM(K,J)
00046800     51 CONTINUE
00046900     52 CONTINUE
00047000 C-----
00047100         DO 25 I=1,N
00047200         SUM=0.0
00047300         DO 24 J=1,N
00047400         DO 23 K=1,N
00047500     23 SUM=SUM+CONJG(H(J,I))*H(K,I)*RM(J,K)
00047600     24 CONTINUE
00047700         DEN=SQRT(SUM)
00047800         DO 26 L=1,N
00047900     26 S (L,I)=H(L,I)/DEN
00048000     25 CONTINUE
00048100 C       DO 31 I=1,N
00048200 C       DO 31 J=1,N
00048300 C   31 H(I,J)=0.
00048400 C       DO 32 I=1,N
00048500 C       DO 32 J=1,N
00048600 C       DO 32 K=1,N
00048700 C   32 H(I,J)=H(I,J)+(ZH(I,K)-V(J)*RM(I,K))*S(K,J)
00048800 C       WRITE(6,33)
```

```
00048900 C   33 FORMAT(' (H-ES)X=')
00049000 C      CALL MATOUT(4)
00049100        GO TO 60
00049200     41 WRITE(6,42)
00049300     42 FORMAT(' OVERLAP MATRIX IS NON-POSITIVE. STOP')
00049400        STOP
00049500     60 RETURN
00049600        END
00049700        SUBROUTINE DENSIT
00049800        COMPLEX*16 S,H,PM,RM
00049900        REAL*8 V
00050000 C-------*** CALC. OF POPULATIONS  ***
00050100        COMMON/COM01/ S(40,40),H(40,40)
00050200        COMMON/COM02/ PM(40,40),RM(40,40)
00050300        COMMON/COM03/ V(40),W(20,40)
00050400        COMMON/COM17/ TOEL(30),EN(40)
00050500        COMMON/COM21/ CON,SUMEN,CONSK
00050600        COMMON/NATOMS/ NATOMS,N,NK
00050700        COMMON/INFO/ AN(35),NOUT
00050800        COMMON/COM13/ TDP(10,10)
00050900        COMMON/COM19/ IRET,ISTOP,NNAT
00051000        COMMON/INFO1/CZ(35),U(80),ULIM(35),LLIM(35),ULK,ULL,ANL,ANK
00051100        INTEGER AN,ULIM,ULK,JLL,CZ,U,ANL,ANK
00051200        SUMEN=0.0
00051300        DO 1 I=1,N
00051400      1 SUMEN=SUMEN+EN(I)*V(I)
00051500        DO 21 I=1,N
00051600        PM(I,I)=0.0
00051700        DO 21 J=1,N
00051800        DO 21 K=1,N
00051900     21 PM(I,I)=PM(I,I)+REAL(CONJG(S(I,J))*S(K,J)*RM(I,K))*EN(J)
00052000        NI=N-1
00052100        DO 22 I=1,NI
00052200        IMA=I+1
00052300        DO 22 J=IMA,N
00052400        PM(I,J)=0.0
00052500        DO 22 K=1,N
00052600     22 PM(I,J)=PM(I,J)+2.0*REAL(RM(I,J)*CONJG(S(I,K))*S(J,K))*EN(K)
00052700        DO 2 I=1,NI
00052800        IJK=I+1
00052900        DO 2 J=IJK,N
00053000      2 PM(J,I)=PM(I,J)
00053100 C-------***   CALC. OF TOTAL  ***
00053200        NNAT=NATOMS
00053300        NI=N-1
00053400        NNATA=NNAT-1
00053500        DO 31 L=1,NNATA
00053600        LKM=L+1
00053700        DO 31 K=LKM,NNAT
00053800        TDP(L,K)=0.0
00053900        DO 32 I=1,NI
00054000        IMJ=I+1
00054100        DO 32 J=IMJ,N
00054200        IF(U(I).NE.L) GO TO 32
00054300        IF(U(J).NE.K) GO TO 32
00054400        TDP(L,K)=TDP(L,K)+PM(I,J)
00054500        TDP(K,L)=TDP(L,K)
00054600     32 CONTINUE
00054700     31 CONTINUE
00054800        DO 3 I=1,NNAT
00054900        TDP(I,I)=0.0
00055000        DO 4 J=1,N
00055100        IF(U(J).NE.I) GO TO 4
00055200        TDP(I,I)=TDP(I,I)+PM(J,J)
00055300      4 CONTINUE
00055400      3 CONTINUE
```

```
00055500         WRITE(6,40) ((TDP(I,J),J=1,NATOMS),I=1,NATOMS)
00055600      40 FORMAT(1H0,'TDP='/1H ,6(6F10.4/1H ))
00055700         RETURN
00055800         END
00055900         SUBROUTINE CHANGE
00056000 C------CHANGING OF VOIP AND AZETA------
00056100         REAL*8 V
00056200         COMMON/COM03/ V(40),W(20,40)
00056300         COMMON/NATOMS/ NATOMS,N,NK
00056400         COMMON/VOIP/ VOIP(4,6),AZETA(4,6),CONST,OM1,OM2,VP(4,6),AZP(4,
00056500         COMMON/COM17/TOEL(30),EN(40)
00056600         COMMON/COM13/ TDP(10,10)
00056700         RETURN ;END

BER BETA         OF  A1402.S.FORT  LIST END
```

ER ALPHA OF A1402.S.FORT

```
00000100       PROGRAM ALPHA
00000200       COMMON/COM01/ ZS(40,40),ZH(40,40)
00000300       COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
00000400       COMMON/COM03/ V(40),W(20,40)
00000500       COMMON/VECK/VECK(3)
00000600 C--------GENERATION OF MATRIX VOIP AND AZETA---------
00000700       COMMON/VOIP/ VOIP(4,9),AZETA(4,9),CONST,OM1,OM2,VP(4,9),AZP(4,9
00000800       COMMON/POS/ POS(3,9),F(3,13),BONDL,AU,BAU,NC(18),LC(10),MC(10)
00000900       COMMON/COM17/TOEL(30),EN(40)
00001000       COMMON/COM13/ TDP(10,10)
00001100       COMMON/INFO/ AN(40),NOUT
00001110       COMMON/INFO1/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
00001200       COMMON/NATOMS/ NATOMS,N,NK
00001300       COMMON /TM/TM(3,3),GM(3,3),POST(3,9),AK(3),NF(3,13),IFMAX,NUNIT
00001400       COMMON /CON/CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX,CSS,CSX,CXS,CXX
00001410       INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
00001420       COMPLEX*16 ZS,ZH
00001500       REAL*8 VM(40)
00001600       REAL*8 V
00001605       DIMENSION OVM(40)
00001610       DIMENSION AKM(3,6)
00001620       DATA AKM/3*.0,.3908,.2256,.0,.3908,.2256,.3080,
00001630      *          .0,.0,.3080,.2256,.0,.0,.2256,.0,.3080/
00001800       DATA AN/14,14,14,8,8,8,8,8,8/,NATOMS/9/
00001900       DATA EN/24*2./,TOEL/4.,4.,4.,6.,6.,6.,6.,6.,6./,OM1,OM2/.5,.5/
00002000       DATA CONST/1.75/,IFMAX,NUNIT/13,9/
00002100 C     DATA VOIP/14.95,7.78,2.05,9*0.,
00002200 C    *           32.38,15.84/
00002300 C     DATA VOIP/12.88,7.08,2.05,9*0.,
00002400 C    *           32.38,19.58/
00002500 C     DATA AZETA/1.38,1.38,1.38,9*0.,
00002600 C    *           2.28,2.28/
00002700 C     DATA AZETA/1.56,1.56,1.38,9*0.,2.1,2.1/
00002800 C     DATA AZETA/1.6344,1.4284,1.38,9*0.,2.246,2.227/
00002900       DATA TM/4.254,0.,0.,-2.456,4.912,0.,0.,0.,5.396/,
00003000      *     GM/1.477,0.7385,0.,0.,1.279,0.,0.,0.,1.1643/
00003100       DATA POST/0.465,0.,0.,0.535,0.535,0.333,0.,0.465,0.667,
00003200      *     0.415,0.272,0.120,0.585,0.857,0.213,0.857,0.585,0.453,
00003300      *     0.143,0.728,0.880,0.728,0.143,0.787,0.272,0.415,0.547/
00003400       DATA NF/0,0,0,1,0,0,0,1,0,0,0,1,-1,0,0,0,-1,0,0,0,-1,
00003500      *        0,-1,-1,0,-1,1,0,1,1,0,1,-1,-1,0,1,1,0,-1/
00003600       DEBUG,SUBCHK
00003700 C-------MAKING TM AND GM MATRIX-----
00003800       AU=.529167
00003900       DO 31 I=1,3
00004000       DO 31 J=1,3
00004100         TM(I,J)=TM(I,J)/AU
00004200    31   GM(I,J)=GM(I,J)*AU
00004300 C-------MAKING POSITION AND F(TRANSLATION) MATRIX-----
00004400       DO 32 I=1,3
00004500       DO 32 J=1,9
00004600       POS(I,J)=0.
00004700       DO 32 K=1,3
00004800    32   POS(I,J)=POS(I,J)+TM(K,I)*POST(K,J)
00004900       DO 33 I=1,3
00005000       DO 33 J=1,13
00005100       F(I,J)=0.
00005200       DO 33 K=1,3
```

```
00005300     33   F(I,J)=F(I,J)+TM(K,I)*NF(K,J)
00005400 C------VOIP,AZETA,C,MODE,ETC INPUT------
00005500  1000 READ(5,14) VOIP(1,1),VOIP(2,1),VOIP(1,4),VOIP(2,4)
00005600     14 FORMAT(4F10.0)
00005700        READ(5,14) AZETA(1,1),AZETA(2,1),AZETA(1,4),AZETA(2,4)
00005800        READ(5,5) CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX
00005900      5 FORMAT(3F10.0)
00006000        READ(5,8) CSS,CSX,CXS,CXX
00006100      8 FORMAT(4F10.0)
00006200        READ(5,9) MODE,MOD1,MNK,ISOL,ITDP,IGRAPH,IHAM
00006300      9 FORMAT(7I2)
00006310        IF(MODE.EQ.0) STOP
00006320        REWIND 30
00006400        CALL COEFFT
00006500 C      IF(MODE.NE.5) GO TO 120
00006600 C      NNNN=0
00006700 C   17 NNNN=NNNN+1
00006800 C      REWIND 30
00006900 C      GO TO (110,111,112,113,121,122,123,124,
00007000 C     *       125,126,127,128,129,130,131),NNNN
00007100 C 110 VOIP(1,1)=VOIP(1,1)-.5 ;GO TO 120
00007200 C 111 VOIP(1,1)=VOIP(1,1)+.5 ;VOIP(2,1)=VOIP(2,1)-.5 ;GO TO 120
00007300 C 112 VOIP(2,1)=VOIP(2,1)+.5 ;VOIP(1,4)=VOIP(1,4)-.5 ;GO TO 120
00007400 C 113 VOIP(1,4)=VOIP(1,4)+.5 ;VOIP(2,4)=VOIP(2,4)-.5 ;GO TO 120
00007500 C 121 VOIP(2,4)=VOIP(2,4)+.5 ;CSSSS=CSSSS-.05 ;GO TO 120
00007600 C 122 CSSSS=CSSSS+.05 ;CSSSX=CSSSX-.05 ;GO TO 120
00007700 C 123 CSSSX=CSSSX+.05 ;CSSXX=CSSXX-.05 ;GO TO 120
00007800 C 124 CSSXX=CSSXX+.05 ;COOSS=COOSS-.05 ;GO TO 120
00007900 C 125 COOSS=COOSS+.05 ;COOSX=COOSX-.05 ;GO TO 120
00008000 C 126 COOSX=COOSX+.05 ;COOXX=COOXX-.05 ;GO TO 120
00008100 C 127 COOXX=COOXX+.05 ;CSS=CSS-.05 ;GO TO 120
00008200 C 128 CSS=CSS+.05 ;CSX=CSX-.05 ;GO  TO 120
00008300 C 129 CSX=CSX+.05 ;CXS=CXS-.05 ;GO TO 120
00008400 C 130 CXS=CXS+.05 ;CXX=CXX-.05 ;GO TO 120
00008500 C 131 STOP 0001
00008600 C 120 CONTINUE
00008700 C--------GENERATION OF MATRIX VOIP AND AZETA---------
00008800        DO 10 I=1,2
00008900          VOIP(I,2)=VOIP(I,1)
00009000          AZETA(I,2)=AZETA(I,1)
00009100          VOIP(I,3)=VOIP(I,1)
00009200          AZETA(I,3)=AZETA(I,1)
00009300        DO 10 J=5,9
00009400          VOIP(I,J)=VOIP(I,4)
00009500          AZETA(I,J)=AZETA(I,4)
00009600     10 CONTINUE
00009700        WRITE(6,15) VOIP(1,1),VOIP(2,1),VOIP(1,4),VOIP(2,4)
00009800     15 FORMAT(' VOIP=',4F10.3)
00009900        WRITE(6,16) AZETA(1,1),AZETA(2,1),AZETA(1,4),AZETA(2,4)
00010000     16 FORMAT(' AZETA=',4F10.3)
00010100        WRITE(6,7) CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX
00010200      7 FORMAT(' CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX=',6F10.3)
00010300        WRITE(6,13) CSS,CSX,CXS,CXX
00010400     13 FORMAT(' CSS,CSX,CXS,CXX=',4F10.3)
00010500        CALL INTGRL
00010510        IF(IHAM.EQ.0) GO TO 140
00010600 C-------- WRITING OUT S-----
00010700        WRITE(6,350)
00010800    350 FORMAT(1H1,1X,23HOVERLAP INTEGRAL MATRIX)
00010900        CALL MATOUT(1)
00011100 C-------CALCULATION OF H MATRIX--------
00011200        DO 408 K=1,NATOMS
00011300        DO 408 L=K,NATOMS
00011400          LLK=LLIM(K)
00011500          LLL=LLIM(L)
00011600          ULK=ULIM(K)
```

```
00011700           ULL=ULIM(L)
00011800           NORBK=ULK-LLK+1
00011900           NORBL=ULL-LLL+1
00012000           DO 409 I=1,NORBK
00012100           DO 409 J=1,NORBL
00012200             LLKP=LLK+I-1
00012300             LLLP=LLL+J-1
00012400             LCI=LC(I)+1
00012500             LCJ=LC(J)+1
00012600             IF(LLKP.EQ.LLLP) GO TO 410
00012700             LLKPU=U(LLKP) ;LLLPU=U(LLLP)
00012800             IF(LLKPU.LE.3.AND.LLLPU.LE.3) GO TO 411
00012900             IF(LLKPU.GE.4.AND.LLLPU.GE.4) GO TO 412
00013000             IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 416
00013100             IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 417
00013200             IF((LLKPU.LE.3.AND.LCI.EQ.1).OR.
00013300          *        (LLLPU.LE.3.AND.LCJ.EQ.1)) GO TO 418
00013400             CONST=CXS ;GO TO 413
00013500      418    CONST=CSX ;GO TO 413
00013600      416    CONST=CSS ;GO TO 413
00013700      417    CONST=CXX ;GO TO 413
00013800      411    IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 422
00013900             IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 423
00014000             CONST=CSSSX ;GO TO 413
00014100      422    CONST=CSSSS ;GO TO 413
00014200      423    CONST=CSSXX ;GO TO 413
00014300      412    IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 420
00014400             IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 421
00014500             CONST=COOSX ;GO TO 413
00014600      420    CONST=COOSS ;GO TO 413
00014700      421    CONST=COOXX ;GO TO 413
00014800      413    CONTINUE
00014900             ZH(LLKP,LLLP)=-S(LLKP,LLLP)*(VOIP(LCI,K)+VOIP(LCJ,L))*
00015000          *                CONST/2.
00015100           GO TO 409
00015200      410  ZH(LLKP,LLKP)=-VOIP(LCI,K)
00015300      409  CONTINUE
00015400      408  CONTINUE
00015500           WRITE(6,400) NK
00015600      400  FORMAT('  HAMILTONIAN MATRIX AT NK=',I5)
00015700           CALL MATOUT(4)
00015800      140  CONTINUE
00015900           NK=1
00016000  C------SCANNING OF K VECTOR------
00016100        2  CONTINUE
00016200           IF(NK-1.GT.MNK) GO TO 1
00016210           GO TO (161,162,163,164,165,166,167,168,168,
00016220          *        190,191,192,193) ,MODE
00016400      161  INIK=1 ;LASK=2 ;GO TO 170
00016500      162  INIK=2 ;LASK=3 ;GO TO 170
00016600      163  INIK=3 ;LASK=4 ;GO TO 170
00016700      164  INIK=4 ;LASK=1 ;GO TO 170
00016800      165  INIK=1 ;LASK=5 ;GO TO 170
00016900      166  INIK=5 ;LASK=6 ;GO TO 170
00017000      167  INIK=6 ;LASK=4 ;GO TO 170
00017100      168  STOP
00017200      170  CONTINUE
00017300           AK1=0.
00017400           DO 180 I=1,3
00017500      180  AK1=AK1+(AKM(I,INIK)-AKM(I,LASK))**2
00017600           AK1=SQRT(AK1)
00017700           DO 181 I=1,3
00017800      181  VECK(I)=(AKM(I,INIK)*(MNK-NK+1.0001)
00017900          *          +AKM(I,LASK)*(NK-1.))/(MNK+0.0001)
00018200           NK=NK+1
00018300           CALL SETSH
```

```
0018400 C       WRITE(6,351) NK
0018500 C 351 FORMAT(' BLOCH OVERLAP MATRIX AT NK=',I5)
0018600 C       CALL MATOUT(2)
0018700 C       WRITE(6,352) NK
0018800 C 352 FORMAT(' BLOCH HAMILTONIAN MATRIX AT NK=',I5)
0018900 C       CALL MATOUT(4)
0019000         CALL SOLUT
0019100         WRITE(6,12) (VECK(I),I=1,3),(V(I),I=1,N)
0019200    12 FORMAT(' EIGENVALUES AT NK=',3F10.3,'ARE',/(6F9.4))
0019210 C------DIFFERENCE-------
0019220         GO TO (40,41,42,42) ,MOD1
0019400    41 WRITE(30) (V(I),I=1,N) ;GO TO 40
0019500    42 READ(30) (VM(I),I=1,N)
0019600         DO 43 I=1,N
0019700    43 VM(I)=V(I)-VM(I)
0019800         WRITE(6,102) NK,(VM(I),I=1,N)
0019900   102 FORMAT( ' DIFFERENCE AT NK=',I5,/(6F9.4))
0020000         IF(MOD1.LE.3) GO TO 40
0020100         WRITE(40) (VM(I),I=1,N)
0020200    40 CONTINUE
0020210 C------ISOL----------
0020300         IF(ISOL.EQ.0) GO TO 103
0020400         WRITE(6,353) NK
0020500   353 FORMAT(' SOLUTION AT NK=',I5)
0020600         CALL MATOUT(3)
0020700   103 CONTINUE
0020710 C------TDP----------
0020800         IF(ITDP.EQ.0) GO TO 44
0020900         CALL DENSIT
0021000    44 CONTINUE
0021100         GO TO 2
0021200     1 CONTINUE
0021300         IF(IGRAPH.EQ.0) GO TO 1000
0021500         MNK1=MNK+1
0021600         WRITE(20,51) N,MNK1,AK1
0021700    51 FORMAT(2I10,F10.4)
0021800         WRITE(20,50) ((W(I,J),J=1,N),I=1,MNK1)
0021900    50 FORMAT(24F10.4)
0021902         STOP 0002
0021904 C------DOS CALCULATION----------
0021906   190 IKZ=1 ;LKZ=2 ;GO TO 194
0021908   191 IKZ=3 ;LKZ=4 ;GO TO 194
0021910   192 IKZ=5 ;LKZ=6 ;GO TO 194
0021912   193 IKZ=7 ;LKZ=7 ;GO TO 194
0021914   194 CONTINUE
0021916         NK=0
0021918         DO 195 KZ=IKZ,LKZ
0021920           VECK(3)=0.0238+(KZ-1)*0.0434
0021922         DO 195 KX=1,9
0021924           VECK(1)=0.0217+(KX-1)*0.0434
0021926           PORN=0.5773*VECK(1)-0.026
0021928           IF(PORN.LT.0.) GO TO 195
0021930           LKY=PORN/0.0434 ;LKY=LKY+1
0021932         DO 195 KY=1,LKY
0021934           VECK(2)=0.0260+(KY-1)*0.0434
0021936           NK=NK+1
0021938           CALL SETSH
0021940           CALL SOLUT
0021942           WRITE(6,197) NK,(VECK(I),I=1,3),(V(I),I=1,N)
0021944   197     FORMAT(' NK=',I4,',K=',3F10.3,/(6F9.4))
0021946         DO 196 I=1,N
0021948   196 DVM(I)=V(I)
0021950         WRITE(50) (DVM(I),I=1,N)
0021952   195 CONTINUE
0022000         STOP ;END
0022100         SUBROUTINE INTGRL
```

```
      IMPLICIT REAL*4(A-H,O-Z)
C-----ATOMIC INTEGRALS FOR CNDO CALCULATIONS
      COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
      COMMON/INFO/ AN(40),NOUT
      COMMON/NATOMS/ NATOMS,N,NK
      COMMON/INFO1/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
      COMMON/GAB/T(9,9),PAIRS(9,9),TEMP(9,9),C1(3),C2(3)
      COMMON/AUXINT/A(17),B(17)
      COMMON /SOUTER/SOUTER(4,4,4,13),NZS(2,4,13)
      COMMON/VOIP/ VOIP(4,9),AZETA(4,9),CONST,OM1,OM2,VP(4,9),AZP(4,9
      COMMON/POS/ POS(3,9),F(3,13),BONDL,AU,BAU,NC(18),LC(10),MC(10)
      COMMON /TM/TM(3,3),GM(3,3),POST(3,9),AK(3),NF(3,13),IFMAX,NUNIT
      DIMENSION P(80,80)
      DIMENSION E(3),Q(40)
      EQUIVALENCE (P(1),Y(1))
      INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
C-----DETERMINATION OF SIZE OF AO BASIS IN AND CORE CHARGE CZ
      DEBUG,SUBCHK
      N=0
C     WRITE(6,1100) ((POS(I,J),J=1,9),I=1,3)
C1100 FORMAT(1H ,'POS='/3(9F10.4/))
C     WRITE(6,1101) ((F(I,J),J=1,13),I=1,3)
C1101 FORMAT(1H ,'F='/3(13F10.4/))
C     WRITE(6,1102) ((VOIP(I,J),J=1,9),I=1,4)
C1102 FORMAT(1H ,'VOIP='/4(9F10.4/))
C     WRITE(6,1103) ((AZETA(I,J),J=1,9),I=1,4)
C1103 FORMAT(1H ,'AZETA='/4(9F10.4/))
      DO 60 I=1,NATOMS
      LLIM(I) = N+1
      K=1
      IF (AN(I).LT.11) GO TO 20
   10 N=N+4
      CZ(I)=AN(I)-10
      GO TO 50
   20 IF (AN(I).LT.3) GO TO 40
   30 N=N+4
      CZ(I) = AN(I)-2
      GO TO 50
   40 N=N+1
      CZ(I)= AN(I)
   50 CONTINUE
      ULIM(I) = N
   60 CONTINUE
C-----ASSIGNMENT OF ANGULAR MOMENTUM QUANTUM NOS. TO ATOMIC ORBITALS
      LC(1)=0
      LC(2)=1
      LC(3)=1
      LC(4)=1
      LC(5)=2
      LC(6)=2
      LC(7)=2
      LC(8)=2
      LC(9)=2
      MC(1)=0
      MC(2)=1
      MC(3)=-1
      MC(4)=0
      MC(5)=0
      MC(6)=1
      MC(7)=-1
      MC(8)=2
      MC(9)=-2
C-----FILL U ARRAY---U(J) IDENTIFIES THE ATOM TO WHICH ORBITAL J IS
C-----ATTACHED E.G. ORBITAL 32 ATTACHED TO  ATOM 7, ETC.
C-----ASSIGNMENT OF ORBITAL EXPONENTS TO ATOMS BY SLATERS RULES
      NC(8)=2 ;NC(14)=3
```

```
00028800         DO 92 K=1,NATOMS
00028900         LLK=LLIM(K)
00029000         JLK=ULIM(K)
00029100         ANK=AN(K)
00029200         NORBK=ULK-LLK+1
00029300         DO 92 I=1,NORBK
00029400         LLKP=LLK+I-1
00029500         LCZETA=LC(I)+1
00029600         A(LLKP)=AZETA(LCZETA,K)
00029700         J(LLKP)=K
00029800      92 CONTINUE
00029900 C-----STEP THRU PAIRS OF ATOMS
00030000         DO 320 IF=1,13
00030100         NIF=0
00030200         DO 320 K=1,NATOMS
00030300         DO 320 L=K,NATOMS
00030400         DO 100 I=1,3
00030500         C1(I)=POS(I,K)
00030600    100  C2(I)=POS(I,L)+F(I,IF)
00030700 C-----CALCULATE UNIT VECTOR ALONG INTERATOM AXIS,E
00030800         CALL RELVEC(R,E,C1,C2)
00030900         IF(R.GT.3.1/AU) GO TO 320
00031000         IF(IF.EQ.1) GO TO 102
00031100         NIF=NIF+1
00031200         NZS(1,NIF,IF)=K
00031300         NZS(2,NIF,IF)=L
00031400    102  CONTINUE
00031500         LLK = LLIM(K)
00031600         LLL = LLIM(L)
00031700         JLK = ULIM(K)
00031800         JLL = ULIM(L)
00031900         NORBK=ULK-LLK+1
00032000         NORBL=ULL-LLL+1
00032100         ANK=AN(K)
00032200         ANL=AN(L)
00032300 C-----LOOP THRU PAIRS OF BASIS FUNCTIONS, ONE ON EACH ATOM
00032400         DO 200 I=1,NORBK
00032500         DO 200 J=1,NORBL
00032600         IF(K.EQ.L.AND.IF.EQ.1) GO TO 160
00032700    110  IF(MC(I).NE.MC(J)) GO TO 150
00032800    120  IF(MC(I).LT.0) GO TO 140
00032900         LLKP=LLK+I-1 ;LLLP=LLL+J-1
00033000    130  PAIRS(I,J)= SQRT((Q(LLKP)*R)**(2*NC(ANK)+1)*(Q(LLLP)*R)**(2*NC(
00033100        1)+1)/(FACT(2*NC(ANK))*FACT(2*NC(ANL))))*(-1.D0)**(LC(J)+MC(J))
00033200        2*SS(NC(ANK),LC(I),MC(I),NC(ANL),LC(J),Q(LLKP)*R,Q(LLLP)*R)
00033300         GO TO 190
00033400    140  PAIRS(I,J)=PAIRS(I-1,J-1)
00033500         GO TO 190
00033600    150  PAIRS(I,J)=0.0D0
00033700         GO TO 190
00033800    160  IF (I.EQ.J) GO TO 170
00033900    180  PAIRS(I,J)=0.0D0
00034000         GO TO 190
00034100    170  PAIRS(I,J)=1.0D0
00034200    190  CONTINUE
00034300    200  CONTINUE
00034400         LCULK=LC(NORBK)
00034500         LCULL=LC(NORBL)
00034600         MAXL=MAX0(LCULK,LCULL)
00034700         IF(R.GT.0.000001D0) GO TO 220
00034800    210  GO TO 250
00034900 C-----ROTATE INTEGRALS FROM DIATOMIC BASIS TO MOLECULAR BASIS
00035000    220  CALL HARMTR(T,MAXL,E)
00035100         DO 230 I=1,NORBK
00035200         DO 230 J=1,NORBL
00035300         TEMP(I,J) = 0.D0
```


A1-40

```
      DO 230 KK=1,NORBL
      TEMP(I,J) = TEMP(I,J)+T(J,KK)*PAIRS(I,KK)
  230 CONTINUE
      DO 240 I=1,NORBK
      DO 240 J=1,NORBL
      PAIRS(I,J) = 0.D0
      DO 240 KK=1,NORBK
      PAIRS(I,J) = PAIRS(I,J)+T(I,KK)*TEMP(KK,J)
  240 CONTINUE
C-----FILL S MATRIX
  250 CONTINUE
      IF(IF.NE.1) GO TO 262
      DO 260 I=1,NORBK
      LLKP=LLK+I-1
      DO 260 J=1,NORBL
      LLLP=LLL+J-1
  260 S(LLKP,LLLP)=PAIRS(I,J)
      GO TO 320
  262 CONTINUE
      DO 264 I=1,NORBK
      DO 264 J=1,NORBL
  264 SOUTER(I,J,NIF,IF)=PAIRS(I,J)
  320 CONTINUE
  330 CONTINUE
      RETURN
      END
      SUBROUTINE SETSH
      IMPLICIT COMPLEX*16 (Z)
      COMMON/COM01/ ZS(40,40),ZH(40,40)
      COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
      COMMON/NATOMS/ NATOMS,N,NK
      COMMON/VECK/VECK(3)
      COMMON/VOIP/ VOIP(4,9),AZETA(4,9),CONST,OM1,OM2,VP(4,9),AZP(4,9)
      COMMON/POS/ POS(3,9),F(3,13),BONDL,AU,BAU,NC(18),LC(10),MC(10)
      COMMON/INFO/ AN(40),NOUT
      COMMON/INFO1/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
      COMMON /SOUTER/SOUTER(4,4,4,13),NZS(2,4,13)
      COMMON /TM/TM(3,3),GM(3,3),POST(3,9),AK(3),NF(3,13),IFMAX,NUNIT
      COMMON /CON/CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX,CSS,CSX,CXS,CXX
      INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
      REAL*4 Z
      DEBUG,SUBCHK
C     WRITE(6,100) (((NZS(I,J,K),J=1,4),I=1,2),K=1,13)
C 100 FORMAT(1H0,'NZS=',/1H ,2(4I2/1H ))
C     WRITE(6,101) ((((SOUTER(I,J,K,L),J=1,4),I=1,4),K=1,4),L=1,13)
C 101 FORMAT(1H0,'SOUTER=',/1H ,4(4F10.4/1H ))
      DO 2 I=1,N
      DO 2 J=I,N
    2 ZS(I,J)=S(I,J)
      DO 3 IF=2,13
      ZTHETA=0.
      DO 4 I=1,3
    4 ZTHETA=ZTHETA+VECK(I)*F(I,IF)*(0.,1.)
      ZEXP=EXP(ZTHETA)
C     WRITE(6,102) ZTHETA,ZEXP,(F(I,IF),I=1,3),IF
C 102 FORMAT(1H ,'ZTHETA,ZEXP=',4F10.4/1H ,'F,IF=',3F10.4,I3)
C-------SET UP S MATRIX FOR A CERTAIN K--------
      DO 5 NIF=1,4
      IF(NZS(1,NIF,IF).EQ.0.) GO TO 3
      K=NZS(1,NIF,IF)
      L=NZS(2,NIF,IF)
      LLK=LLIM(K)
      LLL=LLIM(L)
      ULK=ULIM(K)
      ULL=ULIM(L)
      NORBK=ULK-LLK+1
```

```
      NORBL=ULL-LLL+1
      DO 6 I=1,NORBK
      DO 6 J=1,NORBL
        LLKP=LLK+I-1
        LLLP=LLL+J-1
    6   ZS(LLKP,LLLP)=ZS(LLKP,LLLP)+SOUTER(I,J,NIF,IF)*ZEXP
    5 CONTINUE
    3 CONTINUE
C-------CALCULATION OF H MATRIX--------
      DO 8 K=1,NATOMS
      DO 8 L=K,NATOMS
        LLK=LLIM(K)
        LLL=LLIM(L)
        ULK=ULIM(K)
        ULL=ULIM(L)
        NORBK=ULK-LLK+1
        NORBL=ULL-LLL+1
        DO 9 I=1,NORBK
        DO 9 J=1,NORBL
          LLKP=LLK+I-1
          LLLP=LLL+J-1
          LCI=LC(I)+1
          LCJ=LC(J)+1
          IF(LLKP.EQ.LLLP) GO TO 10
          LLKPU=U(LLKP) ;LLLPU=U(LLLP)
          IF(LLKPU.LE.3.AND.LLLPU.LE.3) GO TO 11
          IF(LLKPU.GE.4.AND.LLLPU.GE.4) GO TO 12
          IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 16
          IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 17
          IF((LLKPU.LE.3.AND.LCI.EQ.1).OR.
     *       (LLLPU.LE.3.AND.LCJ.EQ.1)) GO TO 18
          CONST=CXS ;GO TO 13
   18     CONST=CSX ;GO TO 13
   16     CONST=CSS ;GO TO 13
   17     CONST=CXX ;GO TO 13
   11     IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 22
          IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 23
          CONST=CSSSX ;GO TO 13
   22     CONST=CSSSS ;GO TO 13
   23     CONST=CSSXX ;GO TO 13
   12     IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 20
          IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 21
          CONST=COOSX ;GO TO 13
   20     CONST=COOSS ;GO TO 13
   21     CONST=COOXX ;GO TO 13
   13     CONTINUE
          ZH(LLKP,LLLP)=-ZS(LLKP,LLLP)*(VOIP(LCI,K)+VOIP(LCJ,L))*
     *             CONST/2.
        GO TO 9
   10   ZH(LLKP,LLKP)=-VOIP(LCI,K)
    9 CONTINUE
    8 CONTINUE
C     IF(NK.EQ.1)  CALL MATOUT(4)
C--------SYMMETRIZATION OF S AND H MATRIX--------
      DO 7 I=1,N
      DO 7 J=I,N
        ZS(J,I)=CONJG(ZS(I,J))
        ZH(J,I)=CONJG(ZH(I,J))
    7 CONTINUE
C     DO 15 I=1,N
C     DO 14 J=1,N
C       ZS(I,J)=0.
C  14 CONTINUE
C       ZS(I,I)=1.
C  15 CONTINUE
      RETURN ;END
```

```
      SUBROUTINE DENSIT
      COMPLEX*16 S,H,PM,RM
      REAL*8 V
C-------*** CALC. OF POPULATIONS  ***
      COMMON/COM01/ S(40,40),H(40,40)
      COMMON/COM02/ PM(40,40),RM(40,40)
      COMMON/COM03/ V(40),W(20,40)
      COMMON/COM17/ TOEL(30),EN(40)
      COMMON/COM21/ CON,SUMEN,CONSK
      COMMON/NATOMS/ NATOMS,N,NK
      COMMON/INFO/ AN(40),NOUT
      COMMON/COM13/ TDP(10,10)
      COMMON/COM19/ IRET,ISTOP,NNAT
      COMMON/INF01/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
      INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
      SUMEN=0.0
      DO 1 I=1,N
    1 SUMEN=SUMEN+EN(I)*V(I)
      DO 21 I=1,N
      PM(I,I)=0.0
      DO 21 J=1,N
      DO 21 K=1,N
   21 PM(I,I)=PM(I,I)+REAL(CONJG(S(I,J))*S(K,J)*RM(I,K))*EN(J)
      NI=N-1
      DO 22 I=1,NI
      IMA=I+1
      DO 22 J=IMA,N
      PM(I,J)=0.0
      DO 22 K=1,N
   22 PM(I,J)=PM(I,J)+2.0*REAL(RM(I,J)*CONJG(S(I,K))*S(J,K))*EN(K)
      DO 2 I=1,NI
      IJK=I+1
      DO 2 J=IJK,N
    2 PM(J,I)=PM(I,J)
C-------***   CALC. OF TOTAL  ***
      NAT=NATOMS
      NI=N-1
      NNATA=NNAT-1
      DO 31 L=1,NNATA
      LKM=L+1
      DO 31 K=LKM,NNAT
      TDP(L,K)=0.0
      DO 32 I=1,NI
      IMJ=I+1
      DO 32 J=IMJ,N
      IF(U(I).NE.L) GO TO 32
      IF(U(J).NE.K) GO TO 32
      TDP(L,K)=TDP(L,K)+PM(I,J)
      TDP(K,L)=TDP(L,K)
   32 CONTINUE
   31 CONTINUE
      DO 3 I=1,NNAT
      TDP(I,I)=0.0
      DO 4 J=1,N
      IF(U(J).NE.I) GO TO 4
      TDP(I,I)=TDP(I,I)+PM(J,J)
    4 CONTINUE
    3 CONTINUE
      WRITE(6,40) ((TDP(I,J),J=1,NATOMS),I=1,NATOMS)
   40 FORMAT(1H0,'TDP='/1H ,9(9F8.4/1H ))
      RETURN
      END

ALPHA      OF  A1402.S.FORT  LIST END
```

```
ER DOS          OF  A1402.S.FORT

0000100         PROGRAM DOS
0000200         DIMENSION V(160,36),D(500),E(500)
0000210          DIMENSION A(70)
0000220          DATA B/' '/,S/'*'/
0000300         DO 10 I=1,61
0000400      10 E(I)=-33.+(I-1)*0.05
0000500         DO 11 I=62,303
0000600      11 E(I)=-22.+(I-62)*0.05
0000700         DO 12 I=304,425
0000800      12 E(I)=-2.+(I-304)*0.05
0000900         N=425 ;DEL=0.05 ;DEL2=0.05**2
0001000         PAID=DEL/3.141592/154
0001100         DO 50 J=1,44
0001200      50 READ(50) (V(J,I),I=1,36)
0001300         DO 51 J=45,88
0001400      51 READ(51) (V(J,I),I=1,36)
0001500         DO 52 J=89,132
0001600      52 READ(52) (V(J,I),I=1,36)
0001700         DO 53 J=133,154
0001800      53 READ(53) (V(J,I),I=1,36)
0001900         DO 20 I=1,N
0002000           D(I)=0.
0002100         DO 21 J=1,154
0002200         DO 21 K=1,30
0002300      21 D(I)=D(I)+1./((E(I)-V(J,K))**2+DEL2)
0002400         D(I)=D(I)*PAID
0002500 C       IF(D(I).GT.25.) D(I)=25.
0002600      20 CONTINUE
0002610 C       WRITE(6,100) (E(I),D(I),I=1,N)
0002620 C  100 FORMAT(1H ,2F9.4,3X,2F9.4,3X,2F9.4)
0002625 C       DO 31 I=1,70
0002630 C   31 A(I)=B
0002635 C       DO 30 I=1,N
0002640 C       IP=5.83*D(I)
0002645 C       A(IP)=S
0002650 C       WRITE(6,101) E(I),(A(J),J=1,70)
0002655 C  101 FORMAT(1H ,F9.4,70A1)
0002660 C   30 A(IP)=B
0002700         CALL PLOTS(999.,999.,'DOS')
0002800         CALL PLOT(3.,3.,-3)
0002810         M=3 ;NOM=N/M
0002820         N=NOM
0002830         DO 33 I=1,NOM
0002840         II=M*(I-1)+1
0002850         D(I)=D(II)
0002860         E(I)=E(II)
0002870      33 CONTINUE
0002880         N1=N+1 ;N2=N+2
0003000         E(N1)=-33.
0003100         E(N2)=1.
0003200         D(N1)=0.
0003300         D(N2)=0.5
0003400         CALL AXIS(0.,0.,'ENERGY',-6,40.,0.,E(N1),E(N2))
0003500         CALL AXIS(0.,0.,'DOS',3,25.,90.,D(N1),D(N2))
0003800         CALL LINE(E,D,N,1,0,0)
0003900         CALL PLOTV
0004000         STOP ;END
```

ER SURF JF A1402.I.FORT

```
00000100        PROGRAM SURF
00000200        IMPLICIT COMPLEX*8 (G,V,Z)
00000300        COMPLEX*8 H(8,16),V(8,8)
00000400        COMPLEX*8 G(8,8),CDET,CW(8)
00000500        COMPLEX*8 ZP(8,8)
00000600        COMPLEX*8 GST(8,8,30),ZQ(8,8)
00000700        COMPLEX*8 GOUT(4,4,130)
00000800        COMMON /R/R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13
00000900       *       ,AK1,AK2
00001000        COMMON /COMH/H
00001100        COMMON /H0HDF/ H0,HD,F
00001200        DIMENSION IP(8),RG3M(130),RG4M(130)
00001300        EQUIVALENCE (H(1,9),V(1,1))
00001400        DIMENSION AV1(2),AV2(2),XV(2),YV(2),AKV(2)
00001500        DIMENSION HD(8,16,6),F(3,6),H0(8,16)
00001510 C      DIMENSION H1(8,16)
00001600 C      DIMENSION CM(2,2,3),PQ(2),SIGN(2,4)
00001700 C      DATA CM/1.,0.,0.,1.,.5,.866,.866,-.5,.5,.866,-.866,.5/
00001800 C      DATA PQ/1.,.25/,SIGN/1.,1.,-1.,1.,1.,-1.,-1.,-1./
00001900        DATA AV1/-.6124,.3535/,AV2/-.6124,-.3535/,
00002000       *     XV/-.8165,1.4142/,YV/-1.4142,-.8165/
00002100        DATA V/64*0./
00002110 C------PRESENT1-----
00002120        DATA R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13/
00002130       *-.536,-1.585,-4.105,-.553,-.324,.264,.21,-.084,.034,
00002140       *.199,-.06,.069,-.484/
00002200 C-------PP PARAMETERS-----
00002300 C      DATA R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13/
00002400 C     *-0.911,-1.098,-4.091,-0.551,-0.274,0.245,0.265,
00002500 C     *-0.075,-0.315,0.025,0.025,0.025,-0.315/
00002600 C--------DRESSELHAUS PARAMETERS------
00002700 C      DATA R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13/
00002800 C     * -2.095,-2.965,-0.4,0.61,-1.74,-1.02,
00002900 C     * 0.04125,0.19125,-0.41375,0.18125,0.09125,0.15625,0.04625/
00002910 C-------CHADI PARAMETAERS-----
00002920 C      DATA R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13/
00002930 C     * 0.2250,-1.475,-5.2541,-0.5328,-0.2903,0.6310,
00002940 C     * 7*0./
00003000        DATA F/-0.6124,0.3536,0.,-0.6124,-0.3536,0.,
00003100       *   0.,0.7071,0.,0.,-0.7071,0.,
00003200       *   0.6124,-0.3536,0.,0.6124,0.3536,0./
00003300        DATA H0/128*0./,HD/768*0./
00003400        DATA BETA/2.58/
00003500        DATA PAI/3.141592/
00003510        R1=R1-5.15
00003520        WRITE(6,299) R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13
00003530    299 FORMAT(' R1-R13=',13F8.3)
00003600        XV(1)=XV(1) ;YV(1)=YV(1)
00003700        AG1=1. ;EPSG=0.05
00003710        WRITE(6,8)
00003720      8 FORMAT(' INPUT EI,EF,DE,DELTA,EPSG IN 5F10.0')
00003730        EI=-11. ;EF=0. ;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
00003740        READ(5,9) DEI,DEF,DDE,DDELTA,DEPSG
00003750      9 FORMAT(5F10.0)
00003760        IF(DEI.NE.0.) EI=DEI
00003770        IF(DEF.NE.0.) EF=DEF
00003780        IF(DDE.NE.0.) DE=DDE
00003790        IF(DDELTA.NE.0.) DELTA=DDELTA
```

```
00003800         IF(DEPSG.NE.0.) EPSG=DEPSG
00003810         NN=(EF-EI+0.01)/DE ;NN=NN+1
00003900         ICONT=0 ;BNDCT=0
00004000         READ(5,300) ICONT,BNDCT
00004100     300 FORMAT(I1,F10.3)
00004200         B1=2.35-BNDCT
00004300         B2=SQRT(5.5225+B1**2+1.569*B1)
00004400         EXP1=EXP(BETA*(2.35-B1))
00004500         EXP2=EXP(BETA*(3.84-B2))
00004600         ISC=0
00004700         REWIND 20
00004800         READ(5,11) ISKIP,NKEND
00004900 C       CALL SETHD
00005000 C       CALL RWRITE(H0,'H0    ',8,16,8)
00005100 C       CALL RWRITE(F,'F     ',3,6,6)
00005200 C       DO 14 IF=1,6
00005300 C       DO 15 J=1,16
00005400 C       DO 15 I=1,8
00005500 C    15 H1(I,J)=HD(I,J,IF)
00005600 C    14 CALL RWRITE(H1,'H1    ',8,16,8)
00005700     11 FORMAT(2I3)
00005800         IF(ISKIP.EQ.0) GO TO 13
00005900         DO 12 IS=1,ISKIP
00006000     12 READ(20) (((GOUT(I,J,NUME),I=1,4),J=1,4),NUME=1,NN),
00006100       *             (RG3M(I),RG4M(I),I=1,NN)
00006200     13 CONTINUE
00006300 C------SCANNING OF K-------
00006400          NUMK=0
00006500         DO 1 K1=1,12
00006600         AKV(1)=-1.62+0.28*(K1-1)
00006700         DO 1 K2=1,14
00006800         AKV(2)=-1.88+0.26*(K2-1)
00006900         IF(AKV(2).LT.(0.5774*AKV(1)-1.8856).OR.
00007000       *    AKV(2).LT.(-0.5774*AKV(1)-1.8856).OR.
00007100       *    AKV(2).GT.(0.5774*AKV(1)+1.8856).OR.
00007200       *    AKV(2).GT.(-0.5774*AKV(1)+1.8856)) GO TO 1
00007300            AK1=PAI*(AV1(1)*AKV(1)+AV1(2)*AKV(2))
00007400            AK2=PAI*(AV2(1)*AKV(1)+AV2(2)*AKV(2))
00007500         NUMK=NUMK+1
00007600         IF(NUMK.LE.ISKIP.OR.NUMK.GT.NKEND) GO TO 1
00007700     10 WRITE(6,105) AK1,AK2,K1,K2,NUMK,AKV
00007800 C       WRITE(6,106) AK1,AK2,AKV
00007900 C 106 FORMAT(' AK1,AK2,AKV=',4F10.4)
00008000 C       WRITE(6,107) ((CM(J1,J2,I3),J2=1,2),J1=1,2),
00008100 C     *             (SIGN(J3,I4),J3=1,2)
00008200 C 107 FORMAT(' CM,SIGN=',6F10.4)
00008300   105 FORMAT(' AK1,AK2,K1,K2,NUMK,AKV=',2F8.3,3I3,2F8.3)
00008400 C------SETTING UP OF HAMILTONIAN------
00008500         CALL SETH11
00008600 C       CALL H111K(H,H0,HD,AKV,F)
00008700 C       CALL SETH
00008800 C       CALL CWRITE(H,'H     ',8,16,8)
00009000 C------RESETTING OF G-------
00009100         DO 30 I=1,8
00009200         DO 30 J=1,8
00009300     30 G(J,I)=0.
00009400 C------SCANNING OF ENERGY-----
00009500         E=EI-DE ;NUME=0
00009600 C------CHANGE ENERGY------
00009700     40 NUME=NUME+1
00009800         E=E+DE
00009900         IF(E.GT.EF+0.1*DE) GO TO 2
00010000         ZE=E+(0.,1.)*DELTA
00010100         N=0 ;IN=0 ;ISW=0
00010200     50 N=N+1
00010300 C------CALCULATION OF NEW G-------
```

```
0010305         DO 51 I=1,8
0010310         DO 51 J=1,8
0010315           ZP(I,J)=0.
0010320         DO 51 L=1,8
0010325      51   ZP(I,J)=ZP(I,J)+V(I,L)*G(L,J)
0010330         DO 52 I=1,8
0010335         DO 52 J=1,8
0010340           ZQ(I,J)=-H(I,J)
0010345         DO 52 L=1,8
0010350      52   ZQ(I,J)=ZQ(I,J)-ZP(I,L)*CONJG(V(J,L))
0010400 C       DO 51 I=1,8
0010500 C       DO 51 J=1,8
0010600 C       ZQ(I,J)=-H(I,J)
0010700 C       DO 51 K=1,8
0010800 C       DO 51 L=1,8
0010900 C    51 ZQ(I,J)=ZQ(I,J)-V(I,L)*G(L,K)*CONJG(V(J,K))
0011000         DO 53 I=1,8
0011100      53 ZQ(I,I)=ZQ(I,I)+ZE
0011200         CALL CINV(ZQ,8,0,8,8,1.D-14,CDET,CW,IP,NSTOP)
0011300 C       WRITE(6,205) N,EPSC,G(1,1),G(4,4),G(5,5),G(8,8),IN,ISW
0011400 C 205 FORMAT(1H ,I4,9F8.3,2I2)
0011500 C-------CONVERGENCE CHECK------
0011600         C1=ABS(G(1,1)-ZQ(1,1))/(ABS(ZQ(1,1))+0.001)
0011700         C2=ABS(G(4,4)-ZQ(4,4))/(ABS(ZQ(4,4))+0.001)
0011800         C3=ABS(G(5,5)-ZQ(5,5))/(ABS(ZQ(5,5))+0.001)
0011900         C4=ABS(G(8,8)-ZQ(8,8))/(ABS(ZQ(8,8))+0.001)
0012000         EPSC=C1+C2+C3+C4
0012100         DO 230 J=1,8
0012200         DO 230 I=1,8
0012300    230  G(I,J)=ZQ(I,J)
0012400         IF(ICONT.EQ.1.AND.ISC.EQ.1) GO TO 303
0012500         IF(EPSC.GT.EPSG) GO TO 203
0012600         GO TO 204
0012700 C-------NOT YET CONVERGED------
0012800    203  AG2=ABS(G(1,1))+ABS(G(2,2))
0012900         IF(N.LT.3) GO TO 240
0013000         IF((AG2-AG1)*(AG1-AG0).GT.0.) GO TO 241
0013100 C-------MAX OR MIN-----
0013200         IF(IN.GE.29) GO TO 244
0013300         GO TO (243,244) ,ISW
0013400    243  ISW=ISW+1
0013500 C-------BEFORE AVERAGING-----
0013600    241  IF(ISW.EQ.0) GO TO 240
0013700 C-------IN AVERAGING------
0013800         IN=IN+1
0013900 C-------STORAGE-------
0014000         DO 245 I=1,8
0014100         DO 245 J=1,8
0014200    245  GST(J,I,IN)=G(J,I)
0014300         GO TO 240
0014400 C-------AVERAGE AND CHANGE-----
0014500    244  DO 247 I=1,8
0014600         DO 247 J=1,8
0014700         G(J,I)=0.
0014800         DO 248 K=1,IN
0014900    248  G(J,I)=G(J,I)+GST(J,I,K)
0015000    247  G(J,I)=G(J,I)/IN
0015100         ISW=0 ;IN=0
0015200    240  AG0=AG1 ;AG1=AG2
0015300         GO TO 50
0015400 C-------CONVERGED------
0015500    204  CONTINUE
0015600 C       CALL CWRITE(G,'G     ',8,8,8)
0015700         IF(ICONT.EQ.0) GO TO 302
0015800         DO 301 J=1,4
0015900         DO 301 I=1,4
```

```
00016000         H(I,J+8)=H(I,J+8)*EXP2
00016100         H(I+4,J+12)=H(I+4,J+12)*EXP2
00016200    301  H(I+4,J+8)=H(I+4,J+8)*EXP1
00016300 C       CALL CWRITE(H,'H     ',8,16,8)
00016400 C       WRITE(6,305) B1,B2,EXP1,EXP2
00016500 C  305  FORMAT(' B1,B2,EXP1,EXP2',4F10.3)
00016600         ISC=1 ;GO TO 50
00016700    303  CONTINUE
00016800         ISC=0
00016900         DO 306 J=1,4
00017000         DO 306 I=1,4
00017100         H(I,J+8)=H(I,J+8)/EXP2
00017200         H(I+4,J+12)=H(I+4,J+12)/EXP2
00017300    306  H(I+4,J+8)=H(I+4,J+8)/EXP1
00017400    302  CONTINUE
00017500         WRITE(6,202) N,NUME,E,EPSC,G(1,1),G(4,4),G(5,5),G(8,8)
00017600 C       CALL CWRITE(G,'G     ',8,8,8)
00017700    202  FORMAT(1H ,2I3,F7.2,F8.3,8F7.3)
00017800         RG3=AIMAG(G(5,5)) ;RG4=AIMAG(G(8,8))
00017900         DO 60 J=1,4
00018000         DO 60 I=1,4
00018100     60  GOUT(I,J,NUME)=G(I,J)
00018200         RG3M(NUME)=RG3 ;RG4M(NUME)=RG4
00018300         GO TO 40
00018400 C------OUTPUT ON FILE-----
00018500      2  CONTINUE
00018600         WRITE(20) (((GOUT(I,J,NUME),I=1,4),J=1,4),NUME=1,NN),
00018700        *              (RG3M(I),RG4M(I),I=1,NN)
00018800      1  CONTINUE
00018900         STOP ;END
00019000         SUBROUTINE SETH11
00019100         IMPLICIT COMPLEX*8 (P)
00019200         COMPLEX*8 H(8,16)
00019300         COMMON /R/R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13
00019400        *       ,AK1,AK2
00019500         COMMON /COMH/H
00019600         AK3=AK1-AK2
00019700         P1=EXP((0.,1.)*AK1) ;PI1=CONJG(P1)
00019800         P2=EXP((0.,1.)*AK2) ;PI2=CONJG(P2)
00019900         P3=EXP((0.,1.)*AK3) ;PI3=CONJG(P3)
00020000         C1=COS(AK1) ;C2=COS(AK2) ;C3=COS(AK3)
00020100 C-------SUBMATRIX 1-1--------
00020200         H(1,1)=R1+2*R8*(C3+C1+C2)
00020300         H(2,2)=R1+2*R7*(C3+C2)+2*R8*C1
00020400         H(3,3)=R1+2*R7*(C3+C1)+2*R8*C2
00020500         H(4,4)=R1+2*R7*(C1+C2)+2*R8*C3
00020600         H(1,2)=R2+R10*(P3+PI2)+R11*(PI3+P2)+2*R12*C1
00020700         H(1,3)=R2+R10*(PI3+PI1)+R11*(P3+P1)+2*R12*C2
00020800         H(1,4)=R2+R10*(P1+P2)+R11*(PI1+PI2)+2*R12*C3
00020900         H(2,3)=R2+R10*(PI1+P2)+R11*(P1+PI2)+R9*PI3+R13*P3
00021000         H(2,4)=R2+R10*(PI3+P1)+R11*(P3+PI1)+R9*P2+R13*PI2
00021100         H(3,4)=R2+R10*(P2+P3)+R11*(PI2+PI3)+R9*P1+R13*PI1
00021200 C-------SUBMATRIX 2-2-------
00021300         DO 20 I=5,8
00021400         DO 20 J=I,8
00021500     20  H(I,J)=H(9-J,9-I)
00021600 C------SUBMATRIX 1-2-------
00021700         H(1,5)=R4+R5*(P1+P2)
00021800         H(4,8)=H(1,5)
00021900         H(1,6)=R4*P1+R5*(1.+P2)
00022000         H(3,8)=H(1,6)
00022100         H(1,7)=R4*P2+R5*(1.+P1)
00022200         H(2,8)=H(1,7)
00022300         H(1,8)=R6*(1.+P1+P2)
00022400         H(2,5)=R4*(1.+P2)+R5*P1
00022500         H(4,7)=H(2,5)
```

```
0022600         H(2,6)=R4*(P1+P2)+R5
0022700         H(3,7)=H(2,6)
0022800         H(2,7)=R3*P2+R6*(1.+P1)
0022900         H(3,5)=R4*(1.+P1)+R5*P2
0023000         H(4,6)=H(3,5)
0023100         H(3,6)=R3*P1+R6*(1.+P2)
0023200         H(4,5)=R3+R6*(P1+P2)
0023300         DO 30 I=1,7
0023400         I1=I+1
0023500         DO 30 J=I1,8
0023600      30 H(J,I)=CONJG(H(I,J))
0023700 C-------SUBMATRIX 2-3-----
0023800         H(5,9)=R4 ;H(6,9)=R4 ;H(7,9)=R4 ;H(8,10)=R4 ;H(8,11)=R4
0023900         H(8,12)=R4
0024000         H(5,10)=R5 ;H(5,11)=R5 ;H(6,10)=R5 ;H(6,12)=R5 ;H(7,11)=R5
0024100         H(7,12)=R5
0024200         H(5,12)=R6 ;H(6,11)=R6 ;H(7,10)=R6 ;H(8,9)=R3
0024300 C-------SUBMATRIX 1-3--------
0024400         H(1,9)=R7*(1.+P1+P2)
0024500         H(1,10)=R11*(1.+P1)+R13*P2
0024600         H(1,11)=R11*(1.+P2)+R13*P1
0024700         H(1,12)=R11*(P1+P2)+R13
0024800         H(2,9)=R9*P2+R10*(1.+P1)
0024900         H(2,10)=R7*P2+R8*(1.+P1)
0025000         H(2,11)=R10*P2+R11*P1+R12
0025100         H(2,12)=R10*P2+R11+R12*P1
0025200         H(3,9)=R9*P1+R10*(1.+P2)
0025300         H(3,10)=R10*P1+R11*P2+R12
0025400         H(3,11)=R7*P1+R8*(1.+P2)
0025500         H(3,12)=R10*P1+R11+R12*P2
0025600         H(4,9)=R9+R10*(P1+P2)
0025700         H(4,10)=R10+R11*P2+R12*P1
0025800         H(4,11)=R10+R11*P1+R12*P2
0025900         H(4,12)=R7+R8*(P1+P2)
0026000 C------SUBMATRIX 2-4------
0026100         DO 10 I=5,8
0026200         DO 10 J=13,16
0026300      10 H(I,J)=H(17-J,17-I)
0026400         RETURN ;END
0026500         SUBROUTINE SETH
0026600         IMPLICIT COMPLEX*8 (P)
0026700         COMPLEX*8 H(8,16)
0026800         COMMON /R/R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13
0026900        *       ,AK1,AK2
0027000         COMMON /COMH/H
0027100         PH1=EXP((0.,1.)*AK1)
0027200         PH2=CONJG(PH1)
0027300         PH3=EXP((0.,1.)*AK2)
0027400         PH4=CONJG(PH3)
0027500         PH6=PH1*PH3
0027600         H(1,1)=R1+R7*(PH3+PH4)+R8*(PH1+PH2)
0027700         H(2,2)=H(1,1)
0027800         H(3,3)=R1+R7*(PH1+PH2)+R8*(PH3+PH4)
0027900         H(4,4)=H(3,3)
0028000         H(1,2)=R2+R12*(PH1+PH2)+R9*PH4+R13*PH3
0028100         H(1,3)=R2+R10*(PH2+PH4)+R11*(PH1+PH3)
0028200         H(1,4)=R2+R10*(PH1+PH4)+R11*(PH2+PH3)
0028300         H(2,3)=R2+R10*(PH2+PH3)+R11*(PH1+PH4)
0028400         H(2,4)=R2+R10*(PH1+PH3)+R11*(PH2+PH4)
0028500         H(3,4)=R2+R12*(PH3+PH4)+R9*PH1+R13*PH2
0028600 C
0028700         H(5,5)=H(3,3)
0028800         H(6,6)=H(3,3)
0028900         H(7,7)=H(1,1)
0029000         H(8,8)=H(1,1)
0029100         H(5,6)=H(3,4)
```

```
00029200          H(5,7)=H(2,4)
00029300          H(5,8)=H(1,4)
00029400          H(6,7)=H(2,3)
00029500          H(6,8)=H(1,3)
00029600          H(7,8)=H(1,2)
00029700  C
00029800          H(1,5)=R4+R5*PH1
00029900          H(2,5)=H(1,5)
00030000          H(4,7)=H(1,5)
00030100          H(4,8)=H(1,5)
00030200          H(1,6)=R4*PH1+R5
00030300          H(2,6)=H(1,6)
00030400          H(3,7)=H(1,6)
00030500          H(3,8)=H(1,6)
00030600          H(1,7)=R5*(1.+PH1)
00030700          H(2,8)=H(1,7)
00030800          H(1,8)=R6*(1.+PH1)
00030900          H(2,7)=H(1,8)
00031000          H(3,5)=R4*(1.+PH1)
00031100          H(4,6)=H(3,5)
00031200          H(3,6)=R3*PH1+R6
00031300          H(4,5)=R3+R6*PH1
00031400  C
00031500          DO 10 I=1,7
00031600          I1=I+1
00031700          DO 10 J=I1,8
00031800     10   H(J,I)=CONJG(H(I,J))
00031900  C
00032000          H(5,9)=R4*PH3+R5
00032100          H(6,9)=H(5,9)
00032200          H(8,11)=H(5,9)
00032300          H(8,12)=H(5,9)
00032400          H(5,10)=R4+R5*PH3
00032500          H(6,10)=H(5,10)
00032600          H(7,11)=H(5,10)
00032700          H(7,12)=H(5,10)
00032800          H(5,11)=R5*(1.+PH3)
00032900          H(6,12)=H(5,11)
00033000          H(5,12)=R6*(1.+PH3)
00033100          H(6,11)=H(5,12)
00033200          H(7,9)=R4*(1.+PH3)
00033300          H(8,10)=H(7,9)
00033400          H(7,10)=R3+R6*PH3
00033500          H(8,9)=R3*PH3+R6
00033600  C
00033700          H(1,9)=R7*(PH3+PH6)+R8*(1.+PH1)
00033800          H(1,10)=R10*(1.+PH1)+R11*(PH3+PH6)
00033900          H(1,11)=R11*(PH1+PH3)+R12+R13*PH6
00034000          H(1,12)=R11*(1.+PH6)+R12*PH1+R13*PH3
00034100          H(2,9)=R10*(PH3+PH6)+R11*(1.+PH1)
00034200          H(2,10)=R7*(1.+PH1)+R8*(PH3+PH6)
00034300          H(2,11)=R11*(1.+PH6)+R12*PH3+R13*PH1
00034400          H(2,12)=R11*(PH1+PH3)+R12*PH6+R13
00034500          H(3,9)=R9*PH6+R10*(PH1+PH3)+R12
00034600          H(3,10)=R9*PH1+R10*(1.+PH6)+R12*PH3
00034700          H(3,11)=R7*(PH1+PH6)+R8*(1.+PH3)
00034800          H(3,12)=R10*(PH1+PH6)+R11*(1.+PH3)
00034900          H(4,9)=R9*PH3+R10*(1.+PH6)+R12*PH1
00035000          H(4,10)=R9+R10*(PH1+PH3)+R12*PH6
00035100          H(4,11)=R10*(1.+PH3)+R11*(PH1+PH6)
00035200          H(4,12)=R7*(1.+PH3)+R8*(PH1+PH6)
00035300          DO 20 I=1,4
00035400          H(9-I,16)=H(1,8+I)
00035500          H(9-I,15)=H(2,8+I)
00035600          H(9-I,14)=H(3,8+I)
00035700     20   H(9-I,13)=H(4,8+I)
```

```
0035800          RETURN ;END
0035900          SUBROUTINE CINV(A,N,M,N1,M1,EPS,DET,W,IP,NSTOP)
0036000          IMPLICIT REAL*4(A-H,O-Z)
0036100          COMPLEX*8 A,DET,PIVOT,W,AWK, PVT
0036200          DIMENSION A(N1,1),IP(1),W(1)
0036300          REAL MAX
0036400          NM = N+M
0036500          EPSS = 1.0E-2*EPS
0036600          DET = 1.0E0
0036700          DO 160 I=1,N
0036800          IP(I) = 0
0036900      160 CONTINUE
0037000  C
0037100  C       PIVOT SERCH.
0037200  C
0037300          DO 270 K=1,N
0037400          MAX = -1.0E0
0037500          DO 190 I=1,N
0037600          IF ( IP(I) ) 170,170,190
0037700      170 CONTINUE
0037800          ABSS = ABS(A(I,K))
0037900          IF ( MAX-ABSS) 180,190,190
0038000      180 CONTINUE
0038100          MAX = ABSS
0038200          L = I
0038300      190 CONTINUE
0038400  C
0038500  C       CHECK SINGULARITY AND CALCULATE DETERMINANT.
0038600  C
0038700          IF ( MAX-EPS ) 1010,1010,200
0038800      200 CONTINUE
0038900          PIVOT = A(L,K)
0039000          DET = DET*PIVOT
0039100  C
0039200  C       START SWEEP OUT.
0039300  C
0039400          IP(L) = K
0039500          PVT = -1.0D+0/PIVOT
0039600          DO 250 J=1,NM
0039700          IF ( J-K) 210,250,210
0039800      210 CONTINUE
0039900          AWK = A(L,J) * PVT
0040000          IF ( ABS(AWK)-EPSS ) 240,240,220
0040100      220 CONTINUE
0040200          DO 230 I=1,N
0040300          A(I,J) = A(I,J) + A(I,K)*AWK
0040400      230 CONTINUE
0040500      240 CONTINUE
0040600          A(L,J) = -AWK
0040700      250 CONTINUE
0040800          DO 260 I=1,N
0040900          A(I,K) = A(I,K) * PVT
0041000      260 CONTINUE
0041100          A(L,K) = -PVT
0041200      270 CONTINUE
0041300  C
0041400  C       EXCHANGE COLUMN.
0041500  C
0041600          DO 290 J=1,NM
0041700          DO 280 I=1,N
0041800          W(I) = A(I,J)
0041900      280 CONTINUE
0042000          DO 290 I=1,N
0042100          A( IP(I), J) = W(I)
0042200      290 CONTINUE
0042300          IF ( N-1 ) 350,350,300
```

```
00042400    300 CONTINUE
00042500        DO 320 I=1,N
00042600        DO 310 J=1,N
00042700        W(J) = A(I,J)
00042800    310 CONTINUE
00042900        DO 320 J=1,N
00043000        A(I,J) = W( IP(J) )
00043100    320 CONTINUE
00043200        DO 340 I=1, N-1
00043300        DO 340 J=I+1,N
00043400        IF ( IP(I)-IP(J) ) 340,340,330
00043500    330 CONTINUE
00043600        DET = -DET
00043700    340 CONTINUE
00043800    350 CONTINUE
00043900        ISTOP = 0
00044000        RETURN
00044100 C
00044200 C       ARGUMENT ERROR RETURN.
00044300 C
00044400   1000 CONTINUE
00044500        ISTOP = 3
00044600        WRITE(6,10) N,M,N1,M1
00044700        RETURN
00044800 C
00044900 C       MATRIX IS SINGULAR.
00045000 C
00045100   1010 CONTINUE
00045200        DET = 0.0E0
00045300        IF ( MAX ) 1011,1020,1011
00045400   1011 CONTINUE
00045500        WRITE(6,20) K
00045600        ISTOP = 2
00045700        RETURN
00045800   1020 CONTINUE
00045900        WRITE(6,30)
00046000        ISTOP = 1
00046100        RETURN
00046200     10 FORMAT(1H0,'(SUBR. CINV) INVALID ARGUMENT. N,M,N1,M1 =',4I5)
00046300     20 FORMAT(1H0,'(SUBR. CINV) MATRIX IS SINGULAR AT STEP #',I5)
00046400     30 FORMAT(1H0,'(SUBR. CINV) MATRIX IS SINGULAR.')
00046500        END

ER SURF         OF  A1402.I.FORT  LIST END
```

ER SURF12 JF A1402.I.FORT

```
00000100        PROGRAM SURF12
00000200        IMPLICIT COMPLEX*8 (G,V,Z)
00000300        COMPLEX*8 H(8,16),V(8,8)
00000400        COMPLEX*8 G(8,8),ZP(8,8),CDET,CW(8)
00000500        COMPLEX*8 GST(8,8,30),ZQ(8,8)
00000600        COMMON /R/R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13
00000700       *       ,AK1,AK2
00000800        COMMON /COMH/H
00000900        DIMENSION IP(8)
00001000        EQUIVALENCE (H(1,9),V(1,1))
00001100        DATA V/64*0./
00001200        DATA R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,R13/
00001300       * -2.095,-2.965,-0.4,0.61,-1.74,-1.02,
00001400       * 0.04125,0.19125,-0.41375,0.18125,0.09125,0.15625,0.04625/
00001500        DELTA=0.01
00001600        AK1=0. ;AK2=0.
00001700        CALL SETH11
00001800        DO 1 IE=1,160
00001900        E=-1.+(IE-1)*0.025
00002000 C------RESETTING OF G-------
00002100        DO 30 I=1,8
00002200        DO 30 J=1,8
00002300     30 G(J,I)=0.
00002400        ZE=E+(0.,1.)*DELTA
00002500        N=0
00002600        DO 2 NM=1,16
00002700        N=N+1
00002800 C------CALCULATION OF NEW G-------
00002900        DO 51 I=1,8
00003000        DO 51 J=1,8
00003100        ZP(I,J)=0.
00003200        DO 51 L=1,8
00003300     51 ZP(I,J)=ZP(I,J)+V(I,L)*G(L,J)
00003400        DO 52 I=1,8
00003500        DO 52 J=1,8
00003600        ZQ(I,J)=-H(I,J)
00003700        DO 52 L=1,8
00003800     52 ZQ(I,J)=ZQ(I,J)-ZP(I,L)*CONJG(V(J,L))
00003900        DO 53 I=1,8
00004000     53 ZQ(I,I)=ZQ(I,I)+ZE
00004100        CALL CINV(ZQ,8,0,8,8,1.D-14,CDET,CW,IP,NSTOP)
00004200        DO 230 J=1,8
00004300        DO 230 I=1,8
00004400    230 G(I,J)=ZQ(I,J)
00004500        IF(N.EQ.1.OR.N.EQ.2.OR.N.EQ.4.OR.N.EQ.8.OR.N.EQ.16)
00004600       *   GO TO 3
00004700        GO TO 2
00004800      3 RG1=AIMAG(G(1,1)) ;RG2=AIMAG(G(4,4))
00004900        RG3=AIMAG(G(5,5)) ;RG4=AIMAG(G(8,8))
00005000        WRITE(20) E,RG1,RG2,RG3,RG4
00005100        WRITE(6,100) N,E,RG1,RG2,RG3,RG4
00005200    100 FORMAT(1H ,I3,5F9.3)
00005300      2 CONTINUE
00005400      1 CONTINUE
00005500        STOP ;END
00005600        PROGRAM PLS
00005700        DIMENSION G(280,5,4),E(282),R(282)
00005800        N=160 ;N1=N+1 ;N2=N+2
```

```
00005900          XM=8. ;YM=20.
00006000          DO 1 J=1,160
00006100          DO 1 I=1,5
00006200          READ(20) E(J),G(J,I,1),G(J,I,2),G(J,I,3),G(J,I,4)
00006300        1 CONTINUE
00006400          DO 6 I=1,N
00006500          DO 6 J=1,5
00006600          DO 6 K=1,4
00006700          G(I,J,K)=-G(I,J,K)
00006800        6 IF(G(I,J,K).GT.12.) G(I,J,K)=12.
00006900 C        DO 7 J=1,N
00007000 C        DO 7 I=1,5
00007100 C      7 WRITE(6,100) E(J),G(J,I,1),G(J,I,2),G(J,I,3),G(J,I,4)
00007200 C  100 FORMAT(1H ,5F8.3)
00007300          CALL PLOTS(0.,0.,'SURF')
00007400          CALL PLOT(2.,2.,-3)
00007500          R(N1)=0. ;R(N2)=0.5 ;E(N1)=-1. ;E(N2)=0.5
00007600          DO 2 K=1,4
00007700          DO 3 I=1,5
00007800          CALL AXIS(0.,0.,'ENERGY',-6,XM,0.,-1.,0.5)
00007900          CALL AXIS(0.,0.,'LDOS',4,YM,90.,0.,0.5)
00008000          DO 4 J=1,N
00008100        4 R(J)=G(J,I,K)
00008200          CALL LINE(E,R,N,1,0,0)
00008300        3 CALL PLOT(10.,0.,-3)
00008400        2 CONTINUE
00008500          CALL PLOTV
00008600          STOP ;END
00008700          PROGRAM PLS1
00008800          DIMENSION G(280,5,4),E(282),R(282)
00008900          N=160 ;N1=N+1 ;N2=N+2
00009000          XM=8. ;YM=20.
00009100          DO 1 J=1,N
00009200          I=1
00009300          READ(20) E(J),G(J,I,1),G(J,I,2),G(J,I,3),G(J,I,4)
00009400        1 CONTINUE
00009500          DO 6 I=1,N
00009600          J=1
00009700          DO 6 K=1,4
00009800          G(I,J,K)=-G(I,J,K)
00009900        6 IF(G(I,J,K).GT.12.) G(I,J,K)=12.
00010000 C        DO 7 J=1,N
00010100 C        DO 7 I=1,5
00010200 C      7 WRITE(6,100) E(J),G(J,I,1),G(J,I,2),G(J,I,3),G(J,I,4)
00010300 C  100 FORMAT(1H ,5F8.3)
00010400          CALL PLOTS(0.,0.,'SURF')
00010500          CALL PLOT(2.,2.,-3)
00010600          DO 2 K=1,4
00010700          R(N1)=0. ;R(N2)=0.5 ;E(N1)=-1. ;E(N2)=0.5
00010800          CALL AXIS(0.,0.,'ENERGY',-6,XM,0.,-1.,0.5)
00010900          CALL AXIS(0.,0.,'LDOS',4,YM,90.,0.,0.5)
00011000          I=1
00011100          DO 4 J=1,N
00011200        4 R(J)=G(J,I,K)
00011300          CALL LINE(E,R,N,1,0,0)
00011400          CALL PLOT(10.,0.,-3)
00011500        2 CONTINUE
00011600          CALL PLOTV
00011700          STOP ;END

BER  SURF12     OF  A1402.I.FORT  LIST END
```

ER CBLM3 JF A1402.SS.FORT

```
0000100       PROGRAM CBLM
0000200       IMPLICIT COMPLEX*8 (C)
0000300       DIMENSION GSMAT(800),GPMAT(800),EMAT(800)
0000400       DIMENSION CTEMP(4,4),SGSMAT(800),SGPMAT(800)
0000500       DATA CA,CB,CD,CF,CG,CA0,CB0,CD0,CF0,CG0/
0000600      *     10*0./
0000700       DATA PAI/3.141592/
0000800       NK=0 ;IGRAPH=0
0000900       ITS=0
0001000       SU=0 ;SV=0 ;ST=0 ;SX=0
0001100 C------SILICON EHT INCLUDING OVERLAP-------
0001200 C     U0=-5. ;V0=0. ;T0=2.5 ;X0=3.1 ;ES=-15.7 ;EP=-7.8
0001300 C     SU=.24 ;SV=0. ;ST=-.19 ;SX=-.21
0001400 C------GERMANIUM CHADI--------
0001500       U0=-1.7 ;V0=0.7 ;T0=1.7 ;X0=1.4 ;ES=-6.3 ;EP=2.1
0001600 C------SILICON CHADI--------
0001700 C     U0=-2.0 ;V0=0.8 ;T0=1.9 ;X0=1.5 ;ES=-6. ;EP=1.2
0001800       WRITE(6,113) ES,EP,U0,V0,T0,X0,SU,SV,ST,SX
0001900   113 FORMAT(1H1,'ES,EP,U0,V0,T0,X0,SU,SV,ST,SX=',10F7.3)
0002000       READ(5,63) IGRAPH,EI,EF,DE,EPS,EPSF
0002100    63 FORMAT(I1,5F10.0)
0002200       IF(EPSF.EQ.0.) EPSF=0.001
0002300       IF(EPS.EQ.0.) EPS=1.
0002400       IF(DE.EQ.0.) DE=0.1
0002500       IF(EF.EQ.0.) EF=5.
0002600       E=EI
0002700       WRITE(6,64)
0002800    64 FORMAT(1H1,'     E   EPS  IT',
0002900      *       12X,'GP',12X,'GS',11X,'SGP',11X,'SGS',11X,'TGP',11X,'TGS
0003000      *       )
0003100    60 CONTINUE
0003200       IF(E.GT.EF) GO TO 100
0003300       NN=10
0003400       IT=0
0003500       U=U0-E*SU ;V=V0-E*SV ;T=T0-E*ST ;X=X0-E*SX
0003600    62 CONTINUE
0003700       IT=IT+1
0003800       CE=CMPLX(0.,EPS)
0003900       CQ=-3*X*CG+U*CD
0004000       CP=V*CA+2*T*CB+X*CF
0004100       CR=T*CA+V*CB+T*CB+X*CF
0004200       CY=-X*CA-2*X*CB+U*CF
0004300       CZ=V*CG+2*T*CG+X*CD
0004400       CPC=E+CE-EP-3*CP
0004500       CQC=E+CE-ES-3*CQ
0004600       DO 20 I=1,3
0004700       DO 20 J=1,3
0004800    20 CTEMP(I,J)=CR
0004900       DO 30 I=1,3
0005000         CTEMP(I,I)=CPC
0005100         CTEMP(4,I)=CY
0005200    30   CTEMP(I,4)=CZ
0005300       CTEMP(4,4)=CQC
0005400       CALL INV(CTEMP)
0005500       CQD=CTEMP(4,4)
0005600       CPD=CTEMP(1,1)
0005700       CRD=CTEMP(2,1)
0005800       CYD=CTEMP(4,1)
```

```
005900          CZD=CTEMP(1,4)
006000          CA0=CA ;CB0=CB ;CD0=CD ;CF0=CF ;CG0=CG
006100          CA=CPD*V+2*CRD*T-CZD*X
006200          CB=CRD*V+CPD*T+CRD*T-CZD*X
006300          CD=3*CYD*X+CQD*U
006400          CF=-CYD*V-2*CYD*T+CQD*X
006500          CG=-CPD*X-2*CRD*X-CZD*U
006600          ERR=MAX(ABS(CA-CA0),ABS(CB-CB0),ABS(CD-CD0),
006700         *         ABS(CF-CF0),ABS(CG-CG0))
006800 C        CA =(OMEGA*CA+OMEGA1*CA0)
006900 C        CB =OMEGA*CB+OMEGA1*CB0
007000 C        CD =OMEGA*CD+OMEGA1*CD0
007100 C        CF =OMEGA*CF+OMEGA1*CF0
007200 C        CG =OMEGA*CG+OMEGA1*CG0
007300          N=MOD(IT,NNN)+1
007400          CAR=(CA+CAR*(N-1))/N
007500          CBR=(CB+CBR*(N-1))/N
007600          CDR=(CD+CDR*(N-1))/N
007700          CFR=(CF+CFR*(N-1))/N
007800          CGR=(CG+CGR*(N-1))/N
007900          IF(N.NE.NNN) GO TO 503
008000          CA=CAR ;CB=CBR ;CD=CDR ;CF=CFR ;CG=CGR
008100      503 CONTINUE
008200          IF(IT.GE.300) GO TO 601
008300          IF(ERR.LE.0.0001) GO TO 500
008400          GO TO 62
008500      500 CONTINUE
008600          IF(E.EQ.EI.AND.EPS.GT.EPSF) GO TO 600
008700          GO TO 601
008800      600 EPS=EPS/10
008900          GO TO 62
009000      601 CONTINUE
009100          ITS=ITS+IT
009200          IF(IGRAPH.EQ.9) GO TO 60
009300          CGP=-1./(E-EP-4*(CA*V+2*CB*T+CF*X))/PAI
009400          CGS=-1./(E-ES+4*(3*CG*X-CD*U))/PAI
009500          CSGP=4*(CA*SV+2*CB*ST+CF*SX)*CGP
009600          CSGS=4*(-3*CG*SX+CD*SU)*CGS
009700          CTGP=CGP+CSGP
009800          CTGS=CGS+CSGS
009900          IF(IGRAPH.EQ.1) GO TO 401
010000          WRITE(6,400) E,EPS,IT,CGP,CGS,CSGP,CSGS,CTGP,CTGS
010100      400 FORMAT(1H ,F6.2,F6.4,I4,4F7.4,10F7.3)
010200      401 NK=NK+1
010300          GPMAT(NK)=AIMAG(CGP)*3.
010400          GSMAT(NK)=AIMAG(CGS)
010500          SGPMAT(NK)=AIMAG(CSGP)*3.
010600          SGSMAT(NK)=AIMAG(CSGS)
010700          EMAT(NK)=E ;E=E+DE
010800          GO TO 60
010900      100 WRITE(6,106) ITS
011000      106 FORMAT(' TOTAL ITERATION NUMBER=',I10)
011100          IF(IGRAPH.EQ.0.OR.IGRAPH.EQ.9) STOP
011200          WRITE(20,105) NK
011300      105 FORMAT(I5)
011400          WRITE(20,104) (GPMAT(I),GSMAT(I),SGPMAT(I),SGSMAT(I),EMAT(I),
011500         *                  I=1,NK)
011600      104 FORMAT(5F10.4)
011700          STOP ;END
011800          SUBROUTINE INV(A)
011900          COMMON NSTOP
012000          COMPLEX A,DET,PIVOT,W,AWK, PVT
012100          DIMENSION A(4,4),IP(4),W(4)
012200          REAL MAX
012300 C
012400 C        ARGUMENT CHECK.
```

```
012500 C
012600       N=4 ;M=0 ;N1=4 ;M1=4 ;EPS=1.0E-5
012700       NM = N+M
012800       IF ( N ) 1000,1000,100
012900   100 CONTINUE
013000       IF ( N-N1 ) 110,110,1000
013100   110 CONTINUE
013200       IF ( M ) 1000,120,120
013300   120 CONTINUE
013400       IF ( NM-M1 ) 130,130,1000
013500   130 CONTINUE
013600       IF ( EPS ) 140,150,150
013700   140 CONTINUE
013800       EPS = 1.0E-5
013900   150 CONTINUE
014000 C
014100 C     INITIALIZATION.
014200 C
014300       EPSS = 1.0E-2*EPS
014400       DET = 1.0E0
014500       DO 160 I=1,N
014600       IP(I) = 0
014700   160 CONTINUE
014800 C
014900 C     PIVOT SERCH.
015000 C
015100       DO 270 K=1,N
015200       MAX = -1.0E0
015300       DO 190 I=1,N
015400       IF ( IP(I) ) 170,170,190
015500   170 CONTINUE
015600       ABSS = CABS(A(I,K))
015700       IF ( MAX-ABSS) 180,190,190
015800   180 CONTINUE
015900       MAX = ABSS
016000       L = I
016100   190 CONTINUE
016200 C
016300 C     CHECK SINGULARITY AND CALCULATE DETERMINANT.
016400 C
016500       IF ( MAX-EPS ) 1010,1010,200
016600   200 CONTINUE
016700       PIVOT = A(L,K)
016800       DET = DET*PIVOT
016900 C
017000 C     START SWEEP OUT.
017100 C
017200       IP(L) = K
017300       PVT = -1.0D+0/PIVOT
017400       DO 250 J=1,NM
017500       IF ( J-K) 210,250,210
017600   210 CONTINUE
017700       AWK = A(L,J) * PVT
017800       IF ( CABS(AWK)-EPSS ) 240,240,220
017900   220 CONTINUE
018000       DO 230 I=1,N
018100       A(I,J) = A(I,J) + A(I,K)*AWK
018200   230 CONTINUE
018300   240 CONTINUE
018400       A(L,J) = -AWK
018500   250 CONTINUE
018600       DO 260 I=1,N
018700       A(I,K) = A(I,K) * PVT
018800   260 CONTINUE
018900       A(L,K) = -PVT
019000   270 CONTINUE
```

```
C
C      EXCHANGE COLUMN.
C
       DO 290 J=1,NM
       DO 280 I=1,N
       W(I) = A(I,J)
  280  CONTINUE
       DO 290 I=1,N
       A( IP(I), J) = W(I)
  290  CONTINUE
       IF ( N-1 ) 350,350,300
  300  CONTINUE
       DO 320 I=1,N
       DO 310 J=1,N
       W(J) = A(I,J)
  310  CONTINUE
       DO 320 J=1,N
       A(I,J) = W( IP(J) )
  320  CONTINUE
       DO 340 I=1, N-1
       DO 340 J=I+1,N
       IF ( IP(I)-IP(J) ) 340,340,330
  330  CONTINUE
       DET = -DET
  340  CONTINUE
  350  CONTINUE
       NSTOP = 0
       RETURN
C
C      ARGUMENT ERROR RETURN.
C
 1000  CONTINUE
       NSTOP = 3
       WRITE(6,10) N,M,N1,M1
       RETURN
C
C      MATRIX IS SINGULAR.
C
 1010  CONTINUE
       DET = 0.0E0
       IF ( MAX ) 1011,1020,1011
 1011  CONTINUE
       WRITE(6,20) K
       NSTOP = 2
       RETURN
 1020  CONTINUE
       WRITE(6,30)
       NSTOP = 1
       RETURN
   10  FORMAT(1H0,'(SUBR. CINV) INVALID ARGUMENT. N,M,N1,M1 =',4I5)
   20  FORMAT(1H0,'(SUBR. CINV) MATRIX IS SINGULAR AT STEP #',I5)
   30  FORMAT(1H0,'(SUBR. CINV) MATRIX IS SINGULAR.')
       END
       PROGRAM PLOOS
       DIMENSION GSMAT(800),GPMAT(800),SGSMAT(800),SGPMAT(800)
       DIMENSION Y1(800),Y2(800),X(800),Y3(800)
       READ(20,100) NK
  100  FORMAT(I5)
       READ(20,101)(GPMAT(I),GSMAT(I),SGPMAT(I),SGSMAT(I),X(I),I=1,NK)
  101  FORMAT(5F10.4)
       CALL PLOTS(999.,999.,'SAKURAI TAKAYASU')
       CALL PLOT(1.,5.,-3)
       DO 10 J=1,3
       GO TO (1,2,3),J
     1 DO 11 K=1,NK
       Y1(K)=GPMAT(K)
```


A1-58

```
0025700      11  Y2(K)=GSMAT(K)
0025800          GO TO 4
0025900       2  DO 12 K=1,NK
0026000          Y1(K)=SGPMAT(K)
0026100      12  Y2(K)=SGSMAT(K)
0026200          GO TO 4
0026300       3  DO 13 K=1,NK
0026400          Y1(K)=GPMAT(K)+SGPMAT(K)
0026500      13  Y2(K)=GSMAT(K)+SGSMAT(K)
0026600          GO TO 4
0026700       4  DO 14 K=1,NK
0026800      14  Y3(K)=Y1(K)+Y2(K)
0026900          NK1=NK+1 ;NK2=NK+2
0027000          Y1(NK1)=Y1(1) ;Y1(NK2)=.2
0027100          Y2(NK1)=Y1(1) ;Y2(NK2)=.2
0027200          Y3(NK1)=Y1(1) ;Y3(NK2)=.2
0027300          X(NK1)=X(1) ;X(NK2)=2.
0027400          CALL AXIS(0.,0.,'ENERGY',-6,20.,0.,X(NK1),X(NK2))
0027500          CALL AXIS(0.,0.,'DOS',3,10.,90.,Y1(NK1),Y1(NK2))
0027600          CALL LINE(X,Y1,NK,1,0,0)
0027700          CALL DASHLN(X,Y2,NK,1)
0027800          CALL LINE(X,Y3,NK,1,0,0)
0027900          IF(J.EQ.2) GO TO 15
0028000          CALL PLOT(0.,15.,-3)
0028100          GO TO 16
0028200      15  CALL PLOT(28.,-15.,-3)
0028300      16  CONTINUE
0028400      10  CONTINUE
0028500          CALL PLOTV
0028600          STOP ;END

ER CBLM3      JF  A1402.SS.FORT  LIST END
```

```
ER GEE        OF  A1402.S.FORT

0000100       FUNCTION HH(ANK,LCI,MCI,ANL,LCJ,R,NANK,NANL)
0000200       INTEGER ANK,ANL
0000300 C------PRESENT1-----
0000400       DATA SVSS,SVSP,SVPS,SVPPS,SVPPP,
0000500      *     SUSS,SUSP,SUPS,SUPPS,SUPPP,
0000600      *     DSESS,DSESP/
0000700      *-2.144,2.09,2.09,2.346,-0.588,
0000800      *).123,-0.366,-0.366,0.435,-0.154,
0000900      *-5.29,1.049/
0001000       DATA VSS,VSP,VPS,VPPS,VPPP,
0001100      *     USS,USP,UPS,UPPS,UPPP,
0001200      *     DESS,DESP,DEOS,DEOP/
0001300      *     -2.85,5.4,9.5,5.4,-1.4,
0001400      *     -0.15,0.,0.,0.45,-0.45,
0001500      *     4.42,10.67,-14.63,-1.83/
0001600 C    *-1.5,3.76,3.5,5.71,-0.64,
0001700 C    *-0.6,0.8,0.8,1.29,-0.16,
0001800 C    *3.86,8.36,-16.36,-1.77/
0001900 C------PP PARAMETERS------
0002000 C     DATA SVSS,SVSP,SVPS,SVPPS,SVPPP,
0002100 C    *     SUSS,SUSP,SUPS,SUPPS,SUPPP,
0002200 C    *     DSESS,DSESP/
0002300 C    *     -2.08,2.12,2.12,2.32,-0.52,
0002400 C    *     0.,0.,0.,0.58,-0.1,
0002500 C    *     -4.203,0.187/
0002600 C-------CJ PARAMETERS-----
0002700 C     DATA SVSS,SVSP,SVPS,SVPPS,SVPPP,
0002800 C    *     SUSS,SUSP,SUPS,SUPPS,SUPPP,
0002900 C    *     DSESS,DSESP/
0003000 C    *     -2.075,2.7583,2.7583,3.129,-0.9212,
0003100 C    *     5*0.,
0003200 C    *     -4.2,1.7/
0003300       DATA ZOS,ZOP,ZSS,ZSP/2.246,2.227,1.6344,1.4284/,
0003400      *     ROS/3.0434/
0003500       DATA B/2.58/
0003600       DATA DSO,DSS/-10.4,-5.15/
0003700       ESS=DSO+DESS
0003800       ESP=DSO+DESP
0003900       EOS=DSO+DEOS
0004000       EOP=DSO+DEOP
0004100       SESS=DSESS+DSS
0004200       SESP=DSESP+DSS
0004300       IF(NANK.EQ.5) GO TO 200
0004400       IF(NANK.EQ.2.AND.NANL.EQ.4) GO TO 400
0004500 C      IF((NANK.NE.2.AND.NANL.EQ.4).OR.
0004600 C    *    (NANK.EQ.2.AND.NANL.NE.4)) GO TO 30
0004700       IF(ABS(R).GT.0.01) GO TO 20
0004800 C------ DIAGANAL------
0004900 C     IF(NANK.EQ.4) GO TO 15
0005000       IF(ANK.EQ.14.AND.LCI.EQ.0) GO TO 11
0005100       IF(ANK.EQ.14.AND.LCI.EQ.1) GO TO 12
0005200       IF(ANK.EQ.8.AND.LCI.EQ.0) GO TO 13
0005300       IF(ANK.EQ.8.AND.LCI.EQ.1) GO TO 14
0005400       STOP 0005
0005500    11 HH=ESS ;GO TO 1
0005600    12 HH=ESP ;GO TO 1
0005700    13 HH=EOS ;GO TO 1
0005800    14 HH=EOP ;GO TO 1
```

```
005900 C   15 CONTINUE
006000 C      IF(LCI.EQ.0) GO TO 16
006100 C      IF(LCI.EQ.1) GO TO 17
006200 C   16 HH=SESS ;GO TO 1
006300 C   17 HH=SESP ;GO TO 1
006400       20 CONTINUE
006500          IF(ANK.EQ.14.AND.ANL.EQ.14) GO TO 30
006600          IF(ANK.EQ.14.AND.ANL.EQ.8) GO TO 40
006700          IF(ANK.EQ.8.AND.ANL.EQ.14) GO TO 50
006800          IF(ANK.EQ.8.AND.ANL.EQ.8) GO TO 60
006900          STOP 0006
007000 C------SI-SI-------
007100       30 CONTINUE
007200          HH=0. ;GO TO 1
007300 C------SI-0-------
007400       40 CONTINUE
007500          IF(LCI.EQ.0.AND.LCJ.EQ.0) GO TO 41
007600          IF(LCI.EQ.0.AND.LCJ.EQ.1) GO TO 42
007700          IF(LCI.EQ.1.AND.LCJ.EQ.0) GO TO 43
007800          IF(MCI.EQ.0) GO TO 44
007900          IF(MCI.EQ.1) GO TO 45
008000          STOP 0007
008100       41 HH=VSS*OVERS(3,0,0,ZSS,2,0,0,ZOS,R,1)/
008200         *         OVERS(3,0,0,ZSS,2,0,0,ZOS,ROS,1);GO TO 1
008300       42 HH=VSP*OVERS(3,0,0,ZSS,2,1,0,ZOP,R,1)/
008400         *         OVERS(3,0,0,ZSS,2,1,0,ZOP,ROS,1);GO TO 1
008500       43 HH=-VPS*OVERS(3,1,0,ZSP,2,0,0,ZOS,R,1)/
008600         *         OVERS(3,1,0,ZSP,2,0,0,ZOS,ROS,1);GO TO 1
008700       44 HH=VPPS*OVERS(3,1,0,ZSP,2,1,0,ZOP,R,1)/
008800         *         OVERS(3,1,0,ZSP,2,1,0,ZOP,ROS,1);GO TO 1
008900       45 HH=VPPP*OVERS(3,1,1,ZSP,2,1,1,ZOP,R,1)/
009000         *         OVERS(3,1,1,ZSP,2,1,1,ZOP,ROS,1);GO TO 1
009100 C   41 HH=VSS ;GO TO 1
009200 C   42 HH=VSP ;GO TO 1
009300 C   43 HH=-VPS ;GO TO 1
009400 C   44 HH=VPPS ;GO TO 1
009500 C   45 HH=VPPP ;GO TO 1
009600 C------0-SI-------
009700       50 CONTINUE
009800          IF(LCI.EQ.0.AND.LCJ.EQ.0) GO TO 51
009900          IF(LCI.EQ.0.AND.LCJ.EQ.1) GO TO 52
010000          IF(LCI.EQ.1.AND.LCJ.EQ.0) GO TO 53
010100          IF(MCI.EQ.0) GO TO 54
010200          IF(MCI.EQ.1) GO TO 55
010300          STOP 00011
010400       51 HH=VSS*OVERS(2,0,0,ZOS,3,0,0,ZSS,R,1)/
010500         *         OVERS(2,0,0,ZOS,3,0,0,ZSS,ROS,1);GO TO 1
010600       52 HH=VPS*OVERS(2,0,0,ZOS,3,1,0,ZSP,R,1)/
010700         *         OVERS(2,0,0,ZOS,3,1,0,ZSP,ROS,1);GO TO 1
010800       53 HH=-VSP*OVERS(2,1,0,ZOP,3,0,0,ZSS,R,1)/
010900         *         OVERS(2,1,0,ZOP,3,0,0,ZSS,ROS,1);GO TO 1
011000       54 HH=VPPS*OVERS(2,1,0,ZOP,3,1,0,ZSP,R,1)/
011100         *         OVERS(2,1,0,ZOP,3,1,0,ZSP,ROS,1);GO TO 1
011200       55 HH=VPPP*OVERS(2,1,1,ZOP,3,1,1,ZSP,R,1)/
011300         *         OVERS(2,1,1,ZOP,3,1,1,ZSP,ROS,1);GO TO 1
011400 C   51 HH=VSS ;GO TO 1
011500 C   52 HH=VPS ;GO TO 1
011600 C   53 HH=-VSP ;GO TO 1
011700 C   54 HH=VPPS ;GO TO 1
011800 C   55 HH=VPPP ;GO TO 1
011900 C------0-0-------
012000       60 CONTINUE
012100          IF(LCI.EQ.0.AND.LCJ.EQ.0) GO TO 61
012200          IF(LCI.EQ.0.AND.LCJ.EQ.1) GO TO 62
012300          IF(LCI.EQ.1.AND.LCJ.EQ.0) GO TO 63
012400          IF(MCI.EQ.0) GO TO 64
```

```
0012500         IF(MCI.EQ.1) GO TO 65
0012600         STOP 0012
0012700      61 HH=USS ;GO TO 1
0012800      62 HH=USP ;GO TO 1
0012900      63 HH=-UPS ;GO TO 1
0013000      64 HH=UPPS ;GO TO 1
0013100      65 HH=UPPP ;GO TO 1
0013200 C------SILICON------
0013300     200 CONTINUE
0013400         IF(ABS(R).LT.0.01) GO TO 220
0013500         IF(ABS(R).GT.5.) GO TO 210
0013600 C------1ST NEAREAT------
0013700     400 CONTINUE
0013900         EXPBR=EXP(B*(2.35-.529*R))
0014000 C       B=B
0014100         IF(LCI.EQ.0.AND.LCJ.EQ.0) GO TO 201
0014200         IF(LCI.EQ.0.AND.LCJ.EQ.1) GO TO 202
0014300         IF(LCI.EQ.1.AND.LCJ.EQ.0) GO TO 203
0014400         IF(MCI.EQ.0) GO TO 204
0014500         IF(MCI.EQ.1) GO TO 205
0014600         STOP 0008
0014700     201 HH=EXPBR*SVSS ;GO TO 1
0014800     202 HH=EXPBR*SVSP ;GO TO 1
0014900     203 HH=EXPBR*(-SVPS) ;GO TO 1
0015000     204 HH=EXPBR*SVPPS ;GO TO 1
0015100     205 HH=EXPBR*SVPPP ;GO TO 1
0015200 C 201 HH=SVSS/0.239*OVERS(3,0,0,ZSS,3,0,0,ZSS,R,1);GO TO 1
0015300 C 202 HH=SVSP/(-0.358)*OVERS(3,0,0,ZSS,3,1,0,ZSP,R,1);GO TO 1
0015400 C 203 HH=-SVPS/(0.358)*OVERS(3,1,0,ZSP,3,0,0,ZSS,R,1);GO TO 1
0015500 C 204 HH=SVPPS/(-0.378)*OVERS(3,1,0,ZSP,3,1,0,ZSP,R,1);GO TO 1
0015600 C 205 HH=SVPPP/0.189*OVERS(3,1,1,ZSP,3,1,1,ZSP,R,1);GO TO 1
0015700 C------2ND NEAREST------
0015800     210 CONTINUE
0016000         IF(LCI.EQ.0.AND.LCJ.EQ.0) GO TO 211
0016100         IF(LCI.EQ.0.AND.LCJ.EQ.1) GO TO 212
0016200         IF(LCI.EQ.1.AND.LCJ.EQ.0) GO TO 213
0016300         IF(MCI.EQ.0) GO TO 214
0016400         IF(MCI.EQ.1) GO TO 215
0016500         STOP 0009
0016600     211 HH=SUSS ;GO TO 1
0016700     212 HH=SUSP ;GO TO 1
0016800     213 HH=-SUPS ;GO TO 1
0016900     214 HH=SUPPS ;GO TO 1
0017000     215 HH=SUPPP ;GO TO 1
0017100 C 211 HH=SUSS/0.026*OVERS(3,0,0,ZSS,3,0,0,ZSS,R,1);GO TO 1
0017200 C 212 HH=SUSP/(-0.059)*OVERS(3,0,0,ZSS,3,1,0,ZSP,R,1);GO TO 1
0017300 C 213 HH=-SUPS/(0.059)*OVERS(3,1,0,ZSP,3,0,0,ZSS,R,1);GO TO 1
0017400 C 214 HH=SUPPS/(-0.111)*OVERS(3,1,0,ZSP,3,1,0,ZSP,R,1);GO TO 1
0017500 C 215 HH=SUPPP/0.020*OVERS(3,1,1,ZSP,3,1,1,ZSP,R,1);GO TO 1
0017600     220 IF(LCI.EQ.0) GO TO 221
0017700         IF(LCI.EQ.1) GO TO 222
0017800         STOP 0013
0017900     221 HH=SESS ;GO TO 1
0018000     222 HH=SESP ;GO TO 1
0018100       1 CONTINUE
0018200 C       WRITE(6,101) HH
0018300 C 101 FORMAT(' HH=',F10.3)
0018400         RETURN ;END
0018500         SUBROUTINE INTGRL
0018600         COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
0018700         COMMON/NATOMS/ NATOMS,N,NK
0018800         COMMON/INFO/ AN(40),NOUT
0018900         COMMON/INFO1/CZ(40),J(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
0019000         COMMON/GAB/T(9,9),PAIRS(9,9),TEMP(9,9),C1(3),C2(3)
0019100         COMMON/AUXINT/A(17),B(17)
0019200         COMMON/VOIP/ VOIP(4,10),AZETA(4,10),CONST,OM1,OM2,VP(4,10),
```

```
0019300       *                AZP(4,10)
0019400        COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0019500        COMMON /NC/NC(18),LC(10),MC(10)
0019600        COMMON /NAN/NAN(40)
0019700        DIMENSION P(80,80)
0019800        DIMENSION E(3),Q(40)
0019900        EQUIVALENCE (P(1),Y(1))
0020000        INTEGER AN,ULIM,ULK,JLL,CZ,U,ANL,ANK
0020100 C      DEBUG,SUBCHK
0020200 C-----DETERMINATION OF SIZE OF AO BASIS IN AND CORE CHARGE CZ
0020300        CALL MAKEDN(1,3)
0020400        N=0
0020500        DO 60 I=1,NATOMS
0020600        LLIM(I) = N+1
0020700        K=1
0020800        IF(AN(I).EQ.0) GO TO 10
0020900        IF (AN(I).LT.11) GO TO 20
0021000     10 N=N+4
0021100        CZ(I)=AN(I)-10
0021200        GO TO 50
0021300     20 IF (AN(I).LT.3) GO TO 40
0021400     30 N=N+4
0021500        CZ(I) = AN(I)-2
0021600        GO TO 50
0021700     40 N=N+1
0021800        CZ(I)= AN(I)
0021900     50 CONTINUE
0022000        JLIM(I) = N
0022100     60 CONTINUE
0022200 C-----ASSIGNMENT OF ANGULAR MOMENTUM QUANTUM NOS. TO ATOMIC ORBITALS
0022300        LC(1)=0
0022400        LC(2)=1
0022500        LC(3)=1
0022600        LC(4)=1
0022700        LC(5)=2
0022800        LC(6)=2
0022900        LC(7)=2
0023000        LC(8)=2
0023100        LC(9)=2
0023200        MC(1)=0
0023300        MC(2)=1
0023400        MC(3)=-1
0023500        MC(4)=0
0023600        MC(5)=0
0023700        MC(6)=1
0023800        MC(7)=-1
0023900        MC(8)=2
0024000        MC(9)=-2
0024100 C-----FILL U ARRAY---U(J) IDENTIFIES THE ATOM TO WHICH ORBITAL J IS
0024200 C-----ATTACHED E.G. ORBITAL 32 ATTACHED TO  ATOM 7, ETC.
0024300 C-----ASSIGNMENT OF ORBITAL EXPONENTS TO ATOMS BY SLATERS RULES
0024400        NC(8)=2 ;NC(14)=3
0024500        DO 92 K=1,NATOMS
0024600        LLK=LLIM(K)
0024700        JLK=ULIM(K)
0024800        ANK=AN(K)
0024900        NORBK=ULK-LLK+1
0025000        DO 92 I=1,NORBK
0025100        LLKP=LLK+I-1
0025200        LCZETA=LC(I)+1
0025300        Q(LLKP)=AZETA(LCZETA,K)
0025400        Q(LLKP)=Q(LLKP)
0025500        J(LLKP)=K
0025600     92 CONTINUE
0025700 C-----STEP THRU PAIRS OF ATOMS
0025800        DO 320 K=1,NATOMS
```

```
0025900         DO 320 L=K,NATOMS
0026000         DO 100 I=1,3
0026100         C1(I)=POS(I,K)
0026200     100 C2(I)=POS(I,L)
0026300 C-----CALCULATE UNIT VECTOR ALONG INTERATOM AXIS,E
0026400         CALL RELVEC(R,E,C1,C2)
0026500         THD=(180.-TH)/2.
0026600         CRD=(2.*COS(THD*PAI/180.)+0.1)*BL/AU
0026610         IF(K.EQ.5.AND.L.EQ.6) GO TO 101
0026620         IF(K.EQ.1.AND.L.EQ.6) GO TO 101
0026700         IF(K.EQ.8.AND.L.EQ.10) GO TO 101
0026800         IF(R.GT.CRD) GO TO 320
0026900     101 CONTINUE
0027000         LLK = LLIM(K)
0027100         LLL = LLIM(L)
0027200         JLK = ULIM(K)
0027300         JLL = ULIM(L)
0027400         NORBK=ULK-LLK+1
0027500         NORBL=ULL-LLL+1
0027600         ANK=AN(K)
0027700         ANL=AN(L)
0027800         IF(ANK.EQ.0.OR.ANL.EQ.0) GO TO 150
0027900 C-----LOOP THRU PAIRS OF BASIS FUNCTIONS, ONE ON EACH ATOM
0028000         DO 200 I=1,NORBK
0028100         DO 200 J=1,NORBL
0028200         IF(K.EQ.L) GO TO 160
0028300     110 IF(MC(I).NE.MC(J)) GO TO 150
0028400     120 IF(MC(I).LT.0) GO TO 140
0026500         LLKP=LLK+I-1 ;LLLP=LLL+J-1
0028600 C 130 PAIRS(I,J)= SQRT((Q(LLKP)*R)**(2*NC(ANK)+1)*(Q(LLLP)*R)**(2*NC(
0028700 C     1)+1)/(FACT(2*NC(ANK))*FACT(2*NC(ANL))))*(-1.D0)**(LC(J)+MC(J))
0028800 C     2*SS(NC(ANK),LC(I),MC(I),NC(ANL),LC(J),Q(LLKP)*R,Q(LLLP)*R)
0028900     130 PAIRS(I,J)=HH(ANK,LC(I),MC(I),ANL,LC(J),R,NAN(K),NAN(L))
0029000         GO TO 190
0029100     140 PAIRS(I,J)=PAIRS(I-1,J-1)
0029200         GO TO 190
0029300     150 PAIRS(I,J)=0.0D0
0029400         GO TO 190
0029500     160 IF (I.EQ.J) GO TO 170
0029600     180 PAIRS(I,J)=0.0D0
0029700         GO TO 190
0029800 C 170 PAIRS(I,J)=1.0D0
0029900     170 IF(MC(I).LT.0) GO TO 171
0030000         PAIRS(I,J)=HH(ANK,LC(I),MC(I),ANL,LC(J),R,NAN(K),NAN(L))
0030100         GO TO 190
0030200     171 PAIRS(I,J)=PAIRS(I-1,J-1)
0030300     190 CONTINUE
0030400     200 CONTINUE
0030500         LCULK=LC(NORBK)
0030600         LCULL=LC(NORBL)
0030700         MAXL=MAX0(LCULK,LCULL)
0030800         IF(R.GT.0.000001D0) GO TO 220
0030900     210 GO TO 250
0031000 C-----ROTATE INTEGRALS FROM DIATOMIC BASIS TO MOLECULAR BASIS
0031100     220 CALL HARMTR(T,MAXL,E)
0031200         DO 230 I=1,NORBK
0031300         DO 230 J=1,NORBL
0031400         TEMP(I,J) = 0.D0
0031500         DO 230 KK=1,NORBL
0031600         TEMP(I,J) = TEMP(I,J)+T(J,KK)*PAIRS(I,KK)
0031700     230 CONTINUE
0031800         DO 240 I=1,NORBK
0031900         DO 240 J=1,NORBL
0032000         PAIRS(I,J) = 0.D0
0032100         DO 240 KK=1,NORBK
0032200         PAIRS(I,J) = PAIRS(I,J)+T(I,KK)*TEMP(KK,J)
```

```
032300   240 CONTINUE
032400 C-----FILL S MATRIX
032500   250 CONTINUE
032600       DO 260 I=1,NORBK
032700       LLKP=LLK+I-1
032800       DO 260 J=1,NORBL
032900       LLLP=LLL+J-1
033000   260 S(LLKP,LLLP)=PAIRS(I,J)
033100   320 CONTINUE
033200   380 CONTINUE
033300       RETURN
033400       END
033500       PROGRAM GEE
033600       IMPLICIT COMPLEX (G,V,Z)
033700       COMMON/ARRAYS/S(40,40),Y(9,5,203),AZ(17,45)
033800       COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
033900       COMMON /H/H(40,40)
034000       DIMENSION GLCC(16,16),GSCCM1(16,16),VCED01(16,16),
034100      *          ZP(16,16),ZQ(16,16),IT1(16),IT2(16)
034200       DIMENSION GST(16,16,15)
034300       DIMENSION GSSSM1(4,4),GBBBM1(16,16),GOOO(4,4),
034400      *          GOOOM1(4,4),VOSO(4,4),VBXO(16,8),GXSS(4,4),
034500      *           GXOO(4,4),GXSO(4,4),GXOS(4,4),GXXX(8,8),
034600      *          GXXXM1(8,8),GPBB(16,16),GPXX(8,8),
034700      *          GPBX(16,8),GPXB(8,16),ZW4(4),ZW8(8),ZW16(16),
034800      *           IP4(4),IP8(8),IP16(16),GSSSK(4,4)
034900       DIMENSION GOUT(4,4,20)
035000       DATA IT1/1,4,2,3,13,14,15,16,5,6,7,8,9,10,11,12/,
035100      *     IT2/1,3,4,2,9,10,11,12,13,14,15,16,5,6,7,8/
035200       DATA AU/.529167/,PAI/3.141592/,BL/1.61/
035300 C     DEBUG,SUBCHK
035400       NN=20
035500       READ(21) (((GOUT(I,J,K),I=1,4),J=1,4),K=1,NN)
035600 C     READ(5,100) EI,DE,EF,DELTA,EPSG
035700 C 100 FORMAT(5F10.0)
035800       EI=-5.5 ;EF=-3.6 ;EPSG=0.05
035900       CALL SETSH
036000 C     CALL RWRITE(H,'H    ',40,40,8)
036100 C     CALL RWRITE(S,'S    ',40,40,8)
036200 C------RESET GLCC--------------
036300       DO 10 J=1,16
036400       DO 10 I=1,16
036500    10 GLCC(I,J)=0.
036600       E=EI-0.1 ;NUME=0
036700 C------CHANGE ENERGY------
036800     1 DE=0.1 ;DELTA=0.04 ;NUME=NUME+1
036900 C     IF(E.GE.-6.01.AND.E.LT.-4.01) DE=0.05
037000       E=E+DE
037100 C     IF(E.GT.-5.99.AND.E.LT.-3.99) DELTA=0.02
037200       IF(E.GT.EF+0.1*DE) STOP 0001
037300       ZE=CMPLX(E,DELTA)
037400       N=0
037500       IN=0 ;ISW=0
037600 C------CALCULATE GSCCM1 & VCED01------
037700       DO 11 J=1,16
037800       DO 11 I=1,16
037900         GSCCM1(I,J)=ZE*S(I,J)-H(I,J)
038000    11   VCED01(I,J)=H(I,J+16)-ZE*S(I,J+16)
038100 C------ITERATION FOR GLCC(E)-------
038200     2 N=N+1
038300       DO 20 I=1,16
038400       DO 20 J=I,16
038500         ZP(J,I)=0.
038600       DO 21 K=1,16
038700       DO 21 L=1,16
038800    21 ZP(J,I)=ZP(J,I)+VCED01(J,K)*GLCC(K,L)*VCED01(I,L)
```

```
0038900     20 ZP(I,J)=ZP(J,I)
0039000        DO 22 I=1,16
0039100        DO 22 J=I,16
0039200        ZQ(J,I)=GSCCM1(J,I)-ZP(J,I)-ZP(IT1(J),IT1(I))-ZP(IT2(J),IT2(I))
0039300     22 ZQ(I,J)=ZQ(J,I)
0039400        CALL INVS(ZQ)
0039500 C------CONVERGENCE CHECK---------
0039600        C1=ABS(GLCC(1,1)-ZQ(1,1))/(ABS(ZQ(1,1))+0.001)
0039700        C2=ABS(GLCC(2,2)-ZQ(2,2))/(ABS(ZQ(2,2))+0.001)
0039800        C3=ABS(GLCC(5,5)-ZQ(5,5))/(ABS(ZQ(5,5))+0.001)
0039900        C4=ABS(GLCC(6,6)-ZQ(6,6))/(ABS(ZQ(6,6))+0.001)
0040000        EPSC=C1+C2+C3+C4
0040100        DO 30 I=1,16
0040200        DO 30 J=I,16
0040300          GLCC(J,I)=ZQ(J,I)
0040400     30   GLCC(I,J)=ZQ(J,I)
0040500        IF(EPSC.GT.EPSG) GO TO 3
0040600 C------WRITE ON FILE---------
0040700 C      CALL CWRITE(GLCC,'GLCC ',16,16,8)
0040800        WRITE(20) GLCC(1,1) ,GLCC(2,2),GLCC(1,2),GLCC(2,3)
0040900       *          , ((GLCC(I,J),I=5,8),J=5,8)
0041000       *          , ((GLCC(I,J),I=1,4),J=5,8)
0041100       *          , ((GLCC(I,J),I=5,8),J=9,12)
0041200        WRITE(6,102) N,NUME,E,EPSC,GLCC(1,1),GLCC(2,2),GLCC(5,5),GLCC(6
0041300        DO 303 J=1,4
0041400        DO 303 I=1,4
0041500   303 GSSSK(I,J)=GOUT(I,J,NUME)
0041600 C      CALL CWRITE(GSSSK,'GSSSK',4,4,4)
0041700 C------MAKE GSSSM1 ,GBBBM1, GOOO, GOOOM1---------
0041800        DO 300 J=1,4
0041900        DO 300 I=1,4
0042000   300 GSSSM1(I,J)=GSSSK(I,J)
0042100        CALL CINV(GSSSM1,4,0,4,4,1,D-14,ZDET,ZW4,IP4,NSTOP)
0042200 C      CALL CWRITE(GSSSM1,'GSSSM1',4,4,4)
0042300        DO 301 J=1,16
0042400        DO 301 I=1,16
0042500   301 GBBBM1(I,J)=GLCC(I,J)
0042600        CALL CINV(GBBBM1,16,0,16,16,1.D-14,ZDET,ZW16,IP16,NSTOP)
0042700 C      CALL CWRITE(GBBBM1,'GBBBM1',16,16,8)
0042800        DO 302 J=1,4
0042900        DO 302 I=1,4
0043000        GOOOM1(I,J)=ZE*S(I+32,J+32)-H(I+32,J+32)
0043100   302 GOOO(I,J)=GOOOM1(I,J)
0043200        CALL CINV(GOOO,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0043300 C      CALL CWRITE(GOOO,'GOOO ',4,4,4)
0043400 C      CALL CWRITE(GOOOM1,'GOOOM1',4,4,4)
0043500 C------MAKE VOSD AND VSXD------
0043600        DO 310 J=1,4
0043700        DO 310 I=1,4
0043800   310 VOSD(I,J)=H(I+32,J+36)-ZE*S(I+32,J+36)
0043900        DO 311 J=1,8
0044000        DO 311 I=1,16
0044100   311 VBXD(I,J)=H(I,J+32)-ZE*S(I,J+32)
0044200 C      CALL CWRITE(VOSD,'VOSD ',4,4,4)
0044300 C      CALL CWRITE(VBXD,'VBXD ',16,8,8)
0044400 C------CALCULATE GXSS ,GXOO, GXSO, GXOS------
0044500        DO 320 J=1,4
0044600        DO 320 I=1,4
0044700        GXSS(I,J)=GSSSM1(I,J) ;GXOO(I,J)=GOOOM1(I,J)
0044800        DO 320 K=1,4
0044900        DO 320 L=1,4
0045000        GXOO(I,J)=GXOO(I,J)-VOSD(I,L)*GSSSK(L,K)*VOSD(J,K)
0045100   320 GXSS(I,J)=GXSS(I,J)-VOSD(L,I)*GOOO(L,K)*VOSD(K,J)
0045200        CALL CINV(GXSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0045300        CALL CINV(GXOO,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0045400 C      CALL CWRITE(GXSS,'GXSS ',4,4,4)
```

```
0045500 C       CALL CWRITE(GXOO,'GXOO ',4,4,4)
0045600         DO 322 J=1,4
0045700         DO 322 I=1,4
0045800         GXSO(I,J)=0. ;GXOS(I,J)=0.
0045900         DO 322 K=1,4
0046000         DO 322 L=1,4
0046100         GXSO(I,J)=GXSO(I,J)+GSSSK(I,L)*VOSD(K,L)*GXOO(K,J)
0046200    322  GXOS(I,J)=GXOS(I,J)+GOOO(I,L)*VOSD(L,K)*GXSS(K,J)
0046300 C------MAKE GXXX AND GXXXM1------
0046400         DO 330 J=1,4
0046500         DO 330 I=1,4
0046600         GXXX(I,J)=GXOO(I,J)
0046700         GXXX(I+4,J+4)=GXSS(I,J)
0046800         GXXX(I,J+4)=GXOS(I,J)
0046900    330  GXXX(I+4,J)=GXSO(I,J)
0047000         DO 331 J=1,8
0047100         DO 331 I=1,8
0047200    331  GXXXM1(I,J)=GXXX(I,J)
0047300         CALL CINV(GXXXM1,8,0,8,8,1.D-14,ZDET,ZW8,IP8,NSTOP)
0047400 C       CALL CWRITE(GXXX,'GXXX ',8,8,8)
0047500 C       CALL CWRITE(GXXXM1,'GXXXM1',8,8,8)
0047600 C------CALCULATE GPBB, GPXX, GPBX, GPXB-------
0047700         DO 340 J=1,16
0047800         DO 340 I=1,16
0047900         GPBB(I,J)=GBBBM1(I,J)
0048000         DO 340 K=1,8
0048100         DO 340 L=1,8
0048200    340  GPBB(I,J)=GPBB(I,J)-VBXD(I,L)*GXXX(L,K)*VBXD(J,K)
0048300         CALL CINV(GPBB,16,0,16,16,1.D-14,ZDET,ZW16,IP16,NSTOP)
0048400         DO 341 J=1,8
0048500         DO 341 I=1,8
0048600         GPXX(I,J)=GXXXM1(I,J)
0048700         DO 341 K=1,16
0048800         DO 341 L=1,16
0048900    341  GPXX(I,J)=GPXX(I,J)-VBXD(L,I)*GLCC(L,K)*VBXD(K,J)
0049000         CALL CINV(GPXX,8,0,8,8,1.D-14,ZDET,ZW8,IP8,NSTOP)
0049100 C       CALL CWRITE(GPBB,'GPBB ',16,16,8)
0049200 C       CALL CWRITE(GPXX,'GPXX ',8,8,8)
0049300         DO 342 J=1,8
0049400         DO 342 I=1,16
0049500         GPBX(I,J)=0. ;GPXB(J,I)=0.
0049600         DO 342 K=1,8
0049700         DO 342 L=1,16
0049800         GPBX(I,J)=GPBX(I,J)+GLCC(I,L)*VBXD(L,K)*GPXX(K,J)
0049900    342  GPXB(J,I)=GPXB(J,I)+GXXX(J,K)*VBXD(L,K)*GPBB(L,I)
0050000 C       CALL CWRITE(GPBX,'GPBX ',16,8,8)
0050100 C       CALL CWRITE(GPXB,'GPXB ',8,16,8)
0050200         WRITE(6,343) (GXOO(I,I),I=1,4),(GXSS(I,I),I=1,4),
0050300        *             (GPBB(I,I),I=1,16),(GPXX(I,I),I=1,8)
0050400    343  FORMAT(' GXOO4,GXSS4,GPBB16,GPXX8=',8(/1H ,8F10.3))
0050500         GO TO 1
0050600      3  CONTINUE
0050700    102  FORMAT(1H ,2I3,F7.2,F8.3,8F7.3,I2)
0050800 C------AVERAGING-------
0050900         AG2=AIMAG(GLCC(1,1))
0051000         IF(E.LT.-10.) AG2=AIMAG(GLCC(6,6))
0051100         IF(N.LT.3) GO TO 40
0051200         GO TO 46
0051300     46  IF((AG2-AG1)*(AG1-AG0).GT.0.) GO TO 41
0051400         IF(IN.GE.14) GO TO 44
0051500         GO TO(43,44) ,ISW
0051600     43  ISW=ISW+1
0051700     41  IF(ISW.EQ.0) GO TO 40
0051800         IN=IN+1
0051900         DO 45 I=1,16
0052000         DO 45 J=I,16
```

```
0052100     45 GST(J,I,IN)=GLCC(J,I)
0052200        GO TO 40
0052300     44 DO 47 I=1,16
0052400        DO 47 J=I,16
0052500          GLCC(J,I)=0.
0052600        DO 48 K=1,IN
0052700     48   GLCC(J,I)=GLCC(J,I)+GST(J,I,K)
0052800          GLCC(J,I)=GLCC(J,I)/IN
0052900     47   GLCC(I,J)=GLCC(J,I)
0053000        ISW=0 ;IN=0
0053100     40 AG0=AG1 ;AG1=AG2
0053200        GO TO 2
0053300        END
0053400        SUBROUTINE SETSH
0053500        COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
0053600        COMMON/NATOMS/ NATOMS,N,NK
0053700        COMMON/INFO/ AN(40),NOUT
0053800        COMMON/INFO1/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
0053900        COMMON /CON/CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX,CSS,CSX,CXS,CXX
0054000        INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
0054100        COMMON/VOIP/ VOIP(4,10),AZETA(4,10),CONST,OM1,OM2,VP(4,10),
0054200       *              AZP(4,10)
0054300        COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0054400        COMMON /H/H(40,40)
0054500        COMMON /NC/NC(18),LC(10),MC(10)
0054600        COMMON /NAN/NAN(40)
0054700        DIMENSION TVEC(3),POS0(3,4),POS1(3,4),ROT0(4,4,4),ROT1(4,4,4),
0054800       *          ROT(40,40),SD(40,40),HD(40,40)
0054900        DATA AN/14,8,8,8,14,8,8,8,8,14/,NATOMS/10/
0055000        DATA NAN/2,1,1,1,1,1,1,1,3,4/
0055100 C      DEBUG,SUBCHK
0055200   1000 READ(5,14) VOIP(1,1),VOIP(2,1),VOIP(1,4),VOIP(2,4)
0055300     14 FORMAT(4F10.0)
0055400        READ(5,14) AZETA(1,1),AZETA(2,1),AZETA(1,4),AZETA(2,4)
0055500        READ(5,5) CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX
0055600      5 FORMAT(3F10.0)
0055700        READ(5,8) CSS,CSX,CXS,CXX
0055800      8 FORMAT(4F10.0)
0055900        CALL COEFFT
0056000        WRITE(6,15) VOIP(1,1),VOIP(2,1),VOIP(1,4),VOIP(2,4)
0056100     15 FORMAT(' VOIP=',4F10.3)
0056200        WRITE(6,16) AZETA(1,1),AZETA(2,1),AZETA(1,4),AZETA(2,4)
0056300     16 FORMAT(' AZETA=',4F10.3)
0056400        WRITE(6,7) CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX
0056500      7 FORMAT(' CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX=',6F10.3)
0056600        WRITE(6,13) CSS,CSX,CXS,CXX
0056700     13 FORMAT(' CSS,CSX,CXS,CXX=',4F10.3)
0056800        DO 20 I=1,2
0056900          VOIP(I,5)=VOIP(I,1) ;AZETA(I,5)=AZETA(I,1)
0057000        VOIP(I,10)=VOIP(I,1) ;AZETA(I,10)=AZETA(I,1)
0057100        DO 21 J=2,3
0057200          VOIP(I,J)=VOIP(I,4) ;AZETA(I,J)=AZETA(I,4)
0057300     21  CONTINUE
0057400        DO 22 J=6,9
0057500          VOIP(I,J)=VOIP(I,4) ;AZETA(I,J)=AZETA(I,4)
0057600     22   CONTINUE
0057700     20 CONTINUE
0057800 C------CALCULATE POSITIONS FOR GROUP 0-------
0057900        TVEC(1)=0. ;TVEC(2)=0. ;TVEC(3)=0.
0058000        CALL POSROT(0.,0.,0.,BL,AU,TVEC,POS0,ROT0)
0058100        DO 30 I=1,3
0058200        DO 30 J=1,4
0058300     30 POS(I,J)=POS0(I,J)
0058400 C------CALCULATE POSITIONS FOR GROUP 1-------
0058500        READ(5,32) TH
0058600     32 FORMAT(F10.0)
```

```
0058700         ALPHA=199.47-TH ;ALPHAR=ALPHA*PAI/180.
0058800         TVEC(1)=(.9428+COS(ALPHAR))*BL/AU
0058900         TVEC(2)=0.
0059000         TVEC(3)=(.3333+SIN(ALPHAR))*BL/AU
0059100         RY=19.47-(128.94-TH)
0059200         CALL POSROT(0.,RY,0.,BL,AU,TVEC,POS1,ROT1)
0059300         DO 33 I=1,3
0059400         DO 33 J=1,4
0059500      33 POS(I,J+4)=POS1(I,J)
0059600 C------CALCULATE POSITIONS FOR OX-------
0059700         POS(1,9)=0. ;POS(2,9)=0. ;POS(3,9)=-BL/AU
0059800         POS(1,10)=0. ;POS(2,10)=0. ;POS(3,10)=-2*BL/AU
0059900         WRITE(6,1001) ((POS(I,J),J=1,10),I=1,3)
0060000    1001 FORMAT(10F9.3)
0060100         CALL INTGRL
0060200         DO 35 J=1,40
0060300         DO 35 I=1,40
0060400         H(I,J)=S(I,J)
0060500      35 S(I,J)=0.
0060600         DO 36 I=1,40
0060700      36 S(I,I)=1.
0060800 C--------CALCULATION OF H MATRIX--------
0060900 C       DO 408 K=1,NATOMS
0061000 C       DO 408 L=K,NATOMS
0061100 C         LLK=LLIM(K)
0061200 C         LLL=LLIM(L)
0061300 C         ULK=ULIM(K)
0061400 C         ULL=ULIM(L)
0061500 C         NORBK=ULK-LLK+1
0061600 C         NORBL=ULL-LLL+1
0061700 C         DO 409 I=1,NORBK
0061800 C         DO 409 J=1,NORBL
0061900 C           LLKP=LLK+I-1
0062000 C           LLLP=LLL+J-1
0062100 C           LCI=LC(I)+1
0062200 C           LCJ=LC(J)+1
0062300 C           IF(LLKP.EQ.LLLP) GO TO 410
0062400 C           LLKPU=U(LLKP) ;LLLPU=U(LLLP)
0062500 C       IF(AN(LLKPU).EQ.14.AND.AN(LLLPU).EQ.14) GO TO 411
0062600 C       IF(AN(LLKPU).EQ.8 .AND.AN(LLLPU).EQ.8 ) GO TO 412
0062700 C           IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 416
0062800 C           IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 417
0062900 C       IF((AN(LLKPU).EQ.14.AND.LCI.EQ.1).OR.
0063000 C     *    (AN(LLLPU).EQ.14.AND.LCJ.EQ.1)) GO TO 418
0063100 C           CONST=CXS ;GO TO 413
0063200 C 418       CONST=CSX ;GO TO 413
0063300 C 416       CONST=CSS ;GO TO 413
0063400 C 417       CONST=CXX ;GO TO 413
0063500 C 411       IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 422
0063600 C           IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 423
0063700 C           CONST=CSSSX ;GO TO 413
0063800 C 422       CONST=CSSSS ;GO TO 413
0063900 C 423       CONST=CSSXX ;GO TO 413
0064000 C 412       IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 420
0064100 C           IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 421
0064200 C           CONST=COOSX ;GO TO 413
0064300 C 420       CONST=COOSS ;GO TO 413
0064400 C 421       CONST=COOXX ;GO TO 413
0064500 C 413       CONTINUE
0064600 C           H(LLKP,LLLP)=-S(LLKP,LLLP)*(VOIP(LCI,K)+VOIP(LCJ,L))*
0064700 C     *                 CONST/2.
0064800 C       GO TO 409
0064900 C 410   H(LLKP,LLKP)=-VOIP(LCI,K)
0065000 C 409   CONTINUE
0065100 C 408   CONTINUE
0065200 C------ROTATION TO DESIRED ATOMIC BASIS-------
```

```
0065300          DO 161 J=1,40
0065400          DO 161 I=1,40
0065500      161 ROT(I,J)=0.
0065600          DO 162 J=1,4
0065700            ROT(J+32,J+32)=1.
0065800          DO 162 I=1,4
0065900            ROT(I,J)=ROT0(I,J,1)
0066000            ROT(I+4,J+4)=ROT0(I,J,2)
0066100            ROT(I+8,J+8)=ROT0(I,J,3)
0066200            ROT(I+12,J+12)=ROT0(I,J,4)
0066300            ROT(I+16,J+16)=ROT1(I,J,1)
0066400            ROT(I+20,J+20)=ROT1(I,J,2)
0066500            ROT(I+24,J+24)=ROT1(I,J,3)
0066600      162   ROT(I+28,J+28)=ROT1(I,J,4)
0066700          TVEC(1)=0. ;TVEC(2)=0. ;TVEC(3)=-2*BL/AU
0066800          CALL POSROT(0.,180.,0.,BL,AU,TVEC,POS0,ROT0)
0066900          DO 166 J=1,4
0067000          DO 166 I=1,4
0067100      166 ROT(I+36,J+36)=ROT0(I,J,1)
0067200          DO 163 I=1,40
0067300          DO 163 J=I,40
0067400            S(J,I)=S(I,J)
0067500      163   H(J,I)=H(I,J)
0067600 C        CALL RWRITE(H,'H     ',40,40,8)
0067700 C        CALL RWRITE(S,'S     ',40,40,8)
0067800 C        CALL RWRITE(ROT,'ROT   ',40,40,8)
0067900          DO 164 J=1,40
0068000          DO 164 I=1,40
0068100            SD(I,J)=0. ;HD(I,J)=0.
0068200          DO 164 K=1,40
0068300            SD(I,J)=SD(I,J)+S(I,K)*ROT(K,J)
0068400      164   HD(I,J)=HD(I,J)+H(I,K)*ROT(K,J)
0068500          DO 165 J=1,40
0068600          DO 165 I=1,40
0068700            S(I,J)=0. ;H(I,J)=0.
0068800          DO 165 K=1,40
0068900            S(I,J)=S(I,J)+ROT(K,I)*SD(K,J)
0069000      165   H(I,J)=H(I,J)+ROT(K,I)*HD(K,J)
0069100          RETURN ;END
0069200          SUBROUTINE POSROT(RX,RY,RZ,BL,AU,TVEC,POS,ROT)
0069300          DIMENSION TVEC(3),POS(3,4),ROT(4,4,4),HM(3,4),PM(3,3,3),
0069400         *          T(4,4),TD(4,4)
0069500          DATA HM/0.,0.,-1.,.9428,.0,.3333,
0069600         *       -.4714,.8165,.3333,-.4714,-.8165,.3333/,
0069700         *    PM/.9428,.0,.3333,.0,1.,.0,-.3333,.0,.9428,
0069800         *       -.4714,.8165,.3333,-.8660,-.5,.0,.1667,-.2887,.9428,
0069900         *       -.4714,-.8165,.3333,.8660,-.5,.0,.1667,.2887,.9428/
0070000 C        DEBUG,SUBCHK
0070100          PAI=3.141592
0070200          X=RX*PAI/180. ;Y=RY*PAI/180. ;Z=RZ*PAI/180.
0070300          DO 50 I=1,3
0070400          DO 50 J=1,4
0070500       50 POS(I,J)=BL*HM(I,J)/AU
0070600          POS(3,1)=0.
0070700          DO 51 I=1,4
0070800       51 ROT(1,I,1)=.5
0070900          DO 52 I=2,4
0071000            ROT(1,1,I)=1.
0071100          DO 52 J=2,4
0071200            ROT(1,J,I)=0.
0071300       52   ROT(J,1,I)=0.
0071400          DO 53 I=1,3
0071500          DO 53 J=1,4
0071600       53   ROT(I+1,J,1)=HM(I,J)*.8660
0071700          DO 54 K=1,3
0071800          DO 54 J=1,3
```

```
0071900          DO 54 I=1,3
0072000       54 ROT(I+1,J+1,K+1)=PM(I,J,K)
0072100          IF(X.EQ.0.) GO TO 20
0072200 C--------X AXIS ROTATION---------
0072300          DO 11 J=1,4
0072400          DO 11 I=1,4
0072500       11 T(I,J)=0.
0072600          T(1,1)=1. ;T(2,2)=1.
0072700          SX=SIN(X) ;CX=COS(X)
0072800          T(3,3)=CX ;T(3,4)=-SX
0072900          T(4,3)=SX ;T(4,4)=CX
0073000          DO 12 J=1,4
0073100          DO 12 I=1,4
0073200            TD(I,J)=0.
0073300          DO 12 K=1,4
0073400       12   TD(I,J)=TD(I,J)+T(I,K)*ROT(K,J,1)
0073500          DO 13 J=1,4
0073600          DO 13 I=1,4
0073700       13   ROT(I,J,1)=TD(I,J)
0073800          DO 15 L=2,4
0073900          DO 14 J=1,4
0074000          DO 14 I=1,4
0074100            TD(I,J)=0.
0074200          DO 14 K=1,4
0074300       14 TD(I,J)=TD(I,J)+T(I,K)*ROT(K,J,L)
0074400          DO 15 J=1,4
0074500          DO 15 I=1,4
0074600       15 ROT(I,J,L)=TD(I,J)
0074700          DO 16 J=1,4
0074800          DO 16 I=1,3
0074900            TD(I,J)=0.
0075000          DO 16 K=1,3
0075100       16   TD(I,J)=TD(I,J)+T(I+1,K+1)*POS(K,J)
0075200          DO 17 J=1,4
0075300          DO 17 I=1,3
0075400       17 POS(I,J)=TD(I,J)
0075500 C--------Y AXIS ROTATION---------
0075600       20 IF(Y.EQ.0.) GO TO 30
0075700          DO 21 J=1,4
0075800          DO 21 I=1,4
0075900       21 T(I,J)=0.
0076000          T(1,1)=1. ;T(3,3)=1.
0076100          SY=SIN(Y) ;CY=COS(Y)
0076200          T(2,2)=CY ;T(2,4)=SY
0076300          T(4,2)=-SY ;T(4,4)=CY
0076400          DO 22 J=1,4
0076500          DO 22 I=1,4
0076600            TD(I,J)=0.
0076700          DO 22 K=1,4
0076800       22   TD(I,J)=TD(I,J)+T(I,K)*ROT(K,J,1)
0076900          DO 23 J=1,4
0077000          DO 23 I=1,4
0077100       23   ROT(I,J,1)=TD(I,J)
0077200          DO 25 L=2,4
0077300          DO 24 J=1,4
0077400          DO 24 I=1,4
0077500            TD(I,J)=0.
0077600          DO 24 K=1,4
0077700       24 TD(I,J)=TD(I,J)+T(I,K)*ROT(K,J,L)
0077800          DO 25 J=1,4
0077900          DO 25 I=1,4
0078000       25 ROT(I,J,L)=TD(I,J)
0078100          DO 26 J=1,4
0078200          DO 26 I=1,3
0078300            TD(I,J)=0.
0078400          DO 26 K=1,3
```

```
0078500     26    TD(I,J)=TD(I,J)+T(I+1,K+1)*POS(K,J)
0078600           DO 27 J=1,4
0078700           DO 27 I=1,3
0078800     27 POS(I,J)=TD(I,J)
0078900 C------Z AXIS ROTATION--------
0079000     30 IF(Z.EQ.0.) GO TO 40
0079100           DO 31 J=1,4
0079200           DO 31 I=1,4
0079300     31 T(I,J)=0.
0079400           T(1,1)=1. ;T(4,4)=1.
0079500           SZ=SIN(Z) ;CZ=COS(Z)
0079600           T(2,2)=CZ ;T(2,3)=-SZ
0079700           T(3,2)=SZ ;T(3,3)=CZ
0079800           DO 32 J=1,4
0079900           DO 32 I=1,4
0080000             TD(I,J)=0.
0080100           DO 32 K=1,4
0080200     32    TD(I,J)=TD(I,J)+T(I,K)*ROT(K,J,1)
0080300           DO 33 J=1,4
0080400           DO 33 I=1,4
0080500     33    ROT(I,J,1)=TD(I,J)
0080600           DO 35 L=2,4
0080700           DO 34 J=1,4
0080800           DO 34 I=1,4
0080900             TD(I,J)=0.
0081000           DO 34 K=1,4
0081100     34 TD(I,J)=TD(I,J)+T(I,K)*ROT(K,J,L)
0081200           DO 35 J=1,4
0081300           DO 35 I=1,4
0081400     35 ROT(I,J,L)=TD(I,J)
0081500           DO 36 J=1,4
0081600           DO 36 I=1,3
0081700             TD(I,J)=0.
0081800           DO 36 K=1,3
0081900     36    TD(I,J)=TD(I,J)+T(I+1,K+1)*POS(K,J)
0082000           DO 37 J=1,4
0082100           DO 37 I=1,3
0082200     37 POS(I,J)=TD(I,J)
0082300     40 DO 41 J=1,4
0082400           DO 41 I=1,3
0082500     41 POS(I,J)=POS(I,J)+TVEC(I)
0082600           RETURN ;END
0082700           SUBROUTINE INVS(A)
0082800           COMPLEX A,W
0082900           DIMENSION A(16,16),NOSEQ(16)
0083000 C         DEBUG,SUBCHK
0083100           EPSL=1.E-10 ;N=16
0083200           DO 10 NN=1,N
0083300     10 NOSEQ(NN)=NN
0083400           DO 100 NN=1,N
0083500           P=0.
0083600           DO 20 I=NN,N
0083700           IF(P-ABS(A(I,1))) 21,20,20
0083800     21 P=ABS(A(I,1))
0083900           IP=I
0084000     20 CONTINUE
0084100           IF(P-EPSL) 4,4,5
0084200      4 WRITE(6,22)
0084300     22 FORMAT(' PIVOT IS TOO SMALL. I DON''T THINK YOU ARE SMART.')
0084400      5 NW=NOSEQ(IP)
0084500           NOSEQ(IP)=NOSEQ(NN)
0084600           NOSEQ(NN)=NW
0084700           DO 30 J=1,N
0084800             W=A(IP,J)
0084900             A(IP,J)=A(NN,J)
0085000     30      A(NN,J)=W
```

```
085100          W=A(NN,1)
085200          DO 40 J=2,N
085300       40 A(NN,J-1)=A(NN,J)/W
085400          A(NN,N)=1./W
085500          DO 50 I=1,N
085600          IF(I-NN) 51,50,51
085700       51 W=A(I,1)
085800          DO 60 J=2,N
085900       60 A(I,J-1)=A(I,J)-W*A(NN,J-1)
086000          A(I,N)=-W*A(NN,N)
086100       50 CONTINUE
086200      100 CONTINUE
086300          DO 200 NN=1,N
086400          DO 70 J=NN,N
086500          IF(NOSEQ(J)-NN) 70,71,70
086600       70 CONTINUE
086700       71 NOSEQ(J)=NOSEQ(NN)
086800          DO 80 I=1,N
086900          W=A(I,J)
087000          A(I,J)=A(I,NN)
087100       80 A(I,NN)=W
087200      200 CONTINUE
087300          RETURN ;END

ER GEE          OF  A1402.S.FORT  LIST END
```

ER DATA0 JF A1402.S.FORT

```
000010 -5.5        -3.6         0.1         0.04        0.05
000020 179.
000100 14.95       7.78         29.6        12.7
000200 1.6344      1.4284       2.2460       2.227
000300 .87         1.69         2.5
000400 2.5         4.3          6.0
000500 1.34        2.1          2.1         2.9
000600 180.
000700
000800
000900
001000
001100
001200
001300
001400
001500
001600
```

ER DATA0 JF A1402.S.FORT LIST END

ER GEEO OF A1402.I.FORT

```
0000100        PROGRAM GEEO
0000200        IMPLICIT COMPLEX*8 (G,V,Z)
0000300        COMMON/ARRAYS/S(40,40),Y(9,5,203),AZ(17,45)
0000400        COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0000500        COMMON /H/H(40,40)
0000600        DIMENSION GB11(16,16),GB11M1(16,16),V1AD(16,4),GAAA(4,4),
0000700       *          GAAAM1(4,4),GABAA(4,4),GAB11(16,16),GABA1(4,16),
0000800       *          GAB1A(16,4),ZW4(4),ZW16(16),IP4(4),IP16(16)
0000900        DIMENSION GOUT(8,8)
0001000        DIMENSION GCCCM1(20,20),GICC(20,20),V2CD(16,20),ZW20(20),
0001100       *          IP20(20)
0001200        DIMENSION GD(16,16),RM(130,20)
0001300        DATA AU/.529167/,PAI/3.141592/,BL/1.61/
0001400        WRITE(6,8)
0001500      8 FORMAT(' INPUT EI,EF,DE,DELTA IN 4F10.0')
0001600        EI=-11. ;EF=0. ;DE=0.1 ;DELTA=0.04
0001700        READ(5,9) DEI,DEF,DDE,DDELTA
0001800      9 FORMAT(4F10.0)
0001900        IF(DEI.NE.0.) EI=DEI
0002000        IF(DEF.NE.0.) EF=DEF
0002100        IF(DDE.NE.0.) DE=DDE
0002200        IF(DDELTA.NE.0.) DELTA=DDELTA
0002300        NN=(EF-EI+0.01)/DE ;NN=NN+1
0002400        REWIND 22
0002500        ISKIP=(EI-(-11.)+0.01)/DE
0002600        IF(ISKIP.EQ.0) GO TO 103
0002700        DO 104 I=1,ISKIP
0002800        CALL TREAD(GB11)
0002900    104 CONTINUE
0003000    103 CONTINUE
0003100        READ(5,105) IOUT
0003200    105 FORMAT(I3)
0003300 C------SET UP S AND H-------
0003400        CALL SHBAB(IAN5)
0003500 C      CALL RWRITE(H,'H     ',40,40,10)
0003600 C      CALL RWRITE(S,'S     ',40,40,10)
0003700        E=EI-DE ;NUME=0
0003800 C------CHANGE ENERGE------
0003900      1 NUME=NUME+1
0004000        E=E+DE
0004100        IF(E.GT.EF+0.1*DE) GO TO 3
0004200        ZE=CMPLX(E,DELTA)
0004300 C------CALL TREAD--------
0004400        CALL TREAD(GB11)
0004500 C      CALL CWRITE(GB11, 'GB11 ',16,16,8)
0004600        IF(IAN5.EQ.0) GO TO 101
0004700 C------MAKE V1AD AND GAAAM1-----
0004800        DO 10 J=1,4
0004900        DO 10 I=1,16
0005000     10 V1AD(I,J)=H(I,J+16)-ZE*S(I,J+16)
0005100        IF(NUME.EQ.1.AND.IOUT.EQ.1)
0005200       * CALL CWRITE(V1AD,'V1AD ',16,4,4)
0005300        DO 11 J=1,4
0005400        DO 11 I=1,4
0005500     11 GAAAM1(I,J)=ZE*S(I+16,J+16)-H(I+16,J+16)
0005510        IF(NUME.EQ.1.AND.IOUT.EQ.1)
0005520       * CALL CWRITE(GAAAM1,'GAAAM1',4,4,4)
0005600 C      CALL CWRITE(GAAAM1,'GAAAM1',4,4,4)
```

```
0005700 C------MAKE GAAA AND GB11M1-----
0005800        DO 20 J=1,4
0005900        DO 20 I=1,4
0006000     20 GAAA(I,J)=GAAAM1(I,J)
0006100        CALL CINV(GAAA,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0006200 C      CALL CWRITE(GAAA, 'GAAA ',4,4,4)
0006300        DO 21 J=1,16
0006400        DO 21 I=1,16
0006500     21 GB11M1(I,J)=GB11(I,J)
0006600        CALL CINV(GB11M1,16,0,16,16,1.D-14,ZDET,ZW16,IP16,NSTOP)
0006700 C      CALL CWRITE(GB11M1,'GB11M1',16,16,8)
0006800        GO TO 150
0006900 C------CALCULATE GABAA AND GAB11-----
0007000 C------- SI-SI BOND IN SIO2------
0007100   101 CONTINUE
0007200 C------MAKE V1AD AND GAAAM1 AND GAAA----
0007300        DO 110 J=1,4
0007400        DO 110 I=1,16
0007500   110 V1AD(I,J)=H(I,J+20)-ZE*S(I,J+20)
0007600        DO 112 J=1,4
0007700        DO 112 I=5,16
0007800   112 V1AD(I,J)=0.
0007900 C      CALL CWRITE(V1AD,'V1AD ',16,4,4)
0008000        DO 111 J=1,4
0008100        DO 111 I=1,4
0008200          GAAA(I,J)=GB11(I,J)
0008300   111    GAAAM1(I,J)=GB11(I,J)
0008400        CALL CINV(GAAAM1,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0008500 C      CALL CWRITE(GAAA,'GAAA ',4,4,4)
0008600 C      CALL CWRITE(GAAAM1,'GAAAM1',4,4,4)
0008700 C-------MAKE GB11M1-------
0008800        DO 120 J=1,16
0008900        DO 120 I=1,16
0009000   120 GB11M1(I,J)=GB11(I,J)
0009100        CALL CINV(GB11M1,16,0,16,16,1.D-14,ZDET,ZW16,IP16,NSTOP)
0009200 C      CALL CWRITE(GB11M1,'GB11M1',16,16,8)
0009300   150 CONTINUE
0009400        DO 31 J=1,4
0009500        DO 31 I=1,4
0009600        GABAA(I,J)=GAAAM1(I,J)
0009700        DO 31 K=1,16
0009800        DO 31 L=1,16
0009900     31 GABAA(I,J)=GABAA(I,J)-V1AD(L,I)*GB11(L,K)*V1AD(K,J)
0010000        CALL CINV(GABAA,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0010100 C      CALL CWRITE(GABAA,'GABAA',4,4,4)
0010200        DO 32 J=1,16
0010300        DO 32 I=1,16
0010400        GAB11(I,J)=GB11M1(I,J)
0010500        DO 32 K=1,4
0010600        DO 32 L=1,4
0010700     32 GAB11(I,J)=GAB11(I,J)-V1AD(I,L)*GAAA(L,K)*V1AD(J,K)
0010800        CALL INVS(GAB11)
0010900 C      CALL CWRITE(GAB11,'GAB11',16,16,8)
0011000 C------CALCULATE GABA1 AND GAB1A------
0011100        DO 40 J=1,16
0011200        DO 40 I=1,4
0011300        GABA1(I,J)=0.
0011400        DO 40 K=1,16
0011500        DO 40 L=1,4
0011600     40 GABA1(I,J)=GABA1(I,J)+GAAA(I,L)*V1AD(K,L)*GAB11(K,J)
0011700 C      CALL CWRITE(GABA1,'GABA1',4,16,8)
0011800        DO 41 J=1,4
0011900        DO 41 I=1,16
0012000        GAB1A(I,J)=0.
0012100        DO 41 K=1,4
0012200        DO 41 L=1,16
```

```
012300     41  GAB1A(I,J)=GAB1A(I,J)+GB11(I,L)*V1AD(L,K)*GABAA(K,J)
012400 C       CALL CWRITE(GAB1A,'GAB1A',16,4,4)
012500 C------MAKE GOUT-------
012600         DO 42 J=1,4
012700         DO 42 I=1,4
012800           GOUT(I,J)=GAB11(I,J)
012900           GOUT(I+4,J+4)=GABAA(I,J)
013000           GOUT(I,J+4)=GAB1A(I,J)
013100     42    GOUT(I+4,J)=GABA1(I,J)
013200 C       CALL CWRITE(GOUT, 'GOUT ',8,8,8)
013300 C------OUTPUT------
013400         WRITE(6,100) E,(GOUT(I,I),I=1,8)
013500    100 FORMAT(1H ,17F7.3)
013600         IF(IAN5.EQ.0) GO TO 72
013700         WRITE(20) ((GOUT(I,J),J=I,8),I=1,8)
013800 C------MAKE GCCM1 AND V2CD------
013900         DO 50 J=1,20
014000         DO 50 I=1,16
014100     50  V2CD(I,J)=H(I+20,J)-ZE*S(I+20,J)
014200         JMIN=5
014300         IF(IAN5.NE.0) JMIN=1
014400         DO 54 J=JMIN,16
014500         DO 54 I=1,4
014600     54  V2CD(I,J)=0.
014700         IF(NUME.EQ.1.AND.IOUT.EQ.1)
014800        * CALL CWRITE(V2CD,'V2CD ',16,20,8)
014900         DO 51 J=1,4
015000         DO 51 I=1,4
015100     51  GCCCM1(I+16,J+16)=GABAA(I,J)
015200         DO 52 J=1,16
015300         DO 52 I=1,16
015400     52  GCCCM1(I,J)=GAB11(I,J)
015500         DO 53 J=1,4
015600         DO 53 I=1,16
015700         GCCCM1(I,J+16)=GAB1A(I,J)
015800     53  GCCCM1(J+16,I)=GABA1(J,I)
015810 C       CALL CWRITE(GCCCM1,'GCCCCM1',20,20,8)
015900         CALL CINV(GCCCM1,20,0,20,20,1.D-14,ZDET,ZW20,IP20,NSTOP)
015910 C       CALL CWRITE(GCCCM1,'GCCCCM1',20,20,8)
016000 C------CALCULATE GICC------
016100         DO 60 J=1,16
016200         DO 60 I=1,20
016300           GD(I,J)=0.
016400         DO 60 L=1,16
016500     60    GD(I,J)=GD(I,J)+V2CD(L,I)*GB11(L,J)
016600         DO 61 J=1,20
016700         DO 61 I=1,20
016800           GICC(I,J)=GCCCM1(I,J)
016900         DO 61 L=1,16
017000     61    GICC(I,J)=GICC(I,J)-GD(I,L)*V2CD(L,J)
017010 C       CALL CWRITE(GICC,'GICC ',20,20,8)
017100         CALL CINV(GICC,20,0,20,20,1.D-14,ZDET,ZW20,IP20,NSTOP)
017110 C       CALL CWRITE(GICC,'GICC ',20,20,8)
017120 C------OUTPUT GOOD FOR WEAK-----
017130 C       WRITE(28) ((GICC(I,J),J=17,20),I=17,20)
017140         WRITE(29) ((GICC(I,J),J=1,4),I=1,4)
017200     72  CONTINUE
017300         DO 70 I=1,8
017400     70    RM(NUME,I)=AIMAG(GOUT(I,I))
017500         DO 71 I=1,12
017600     71    RM(NUME,I+8)=AIMAG(GICC(I+8,I+8))
017700         WRITE(6,73) E,(RM(NUME,I),I=1,20)
017800     73  FORMAT(1H ,13F7.3)
017900         GO TO 1
018000      3  CONTINUE
018100         WRITE(30) ((RM(NUME,I),I=1,20),NUME=1,NN)
```

```
0018200       CALL RMAT(RM,'RM    ',130,20,NN,10)
0018300       STOP 0002
0018400       END
0018500       SUBROUTINE SHBAB(IAN5)
0018600       COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
0018700       COMMON/NATOMS/ NATOMS,N,NK
0018800       COMMON/INFO/ AN(40),NOUT
0018900       COMMON/INFO1/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
0019000       COMMON /CON/CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX,CSS,CSX,CXS,CX
0019100       INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
0019200       COMMON/VOIP/ VOIP(4,10),AZETA(4,10),CONST,OM1,OM2,VP(4,10),
0019300      *                AZP(4,10)
0019400       COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0019500       COMMON /H/H(40,40)
0019600       COMMON /NC/NC(18),LC(10),MC(10)
0019700       COMMON /NAN/NAN(40)
0019800       DIMENSION TVEC(3),POS0(3,4),POS1(3,4),ROT0(4,4,4),ROT1(4,4,4),
0019900      *          ROT(40,40),SD(40,40),HD(40,40)
0020000       DATA AN/14,8,8,8,8,14,8,8,8/,NATOMS/9/
0020100       DATA NAN/2,1,1,1,3,4,1,1,1/
0020200  1000 READ(5,14) VOIP(1,1),VOIP(2,1),VOIP(1,4),VOIP(2,4)
0020300    14 FORMAT(4F10.0)
0020400       READ(5,14) AZETA(1,1),AZETA(2,1),AZETA(1,4),AZETA(2,4)
0020500       READ(5,5) CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX
0020600     5 FORMAT(3F10.0)
0020700       READ(5,8) CSS,CSX,CXS,CXX
0020800     8 FORMAT(4F10.0)
0020900       CALL COEFFT
0021000       WRITE(6,15) VOIP(1,1),VOIP(2,1),VOIP(1,4),VOIP(2,4)
0021100    15 FORMAT(' VOIP=',4F10.3)
0021200       WRITE(6,16) AZETA(1,1),AZETA(2,1),AZETA(1,4),AZETA(2,4)
0021300    16 FORMAT(' AZETA=',4F10.3)
0021400       WRITE(6,7) CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX
0021500     7 FORMAT(' CSSSS,CSSSX,CSSXX,COOSS,COOSX,COOXX=',6F10.3)
0021600       WRITE(6,13) CSS,CSX,CXS,CXX
0021700    13 FORMAT(' CSS,CSX,CXS,CXX=',4F10.3)
0021800       DO 20 I=1,2
0021900         VOIP(I,6)=VOIP(I,1) ;AZETA(I,6)=AZETA(I,1)
0022000       DO 21 J=2,3
0022100         VOIP(I,J)=VOIP(I,4) ;AZETA(I,J)=AZETA(I,4)
0022200    21  CONTINUE
0022300       DO 22 J=7,9
0022400         VOIP(I,J)=VOIP(I,4) ;AZETA(I,J)=AZETA(I,4)
0022500    22   CONTINUE
0022600    20 CONTINUE
0022700 C------INPUT INFORMATION ON IMPURITY-----
0022800       READ(5,200) AN(5),ETA1,ETA2
0022900   200 FORMAT(I2,2F10.0)
0023000       WRITE(6,202) AN(5),ETA1,ETA2
0023100       IAN5=AN(5)
0023200   202 FORMAT(' AN(5),ETA1,ETA2=',I2,2F10.3)
0023300       READ(5,201) VOIP(1,5),VOIP(2,5)
0023400       READ(5,201) AZETA(1,5),AZETA(2,5)
0023500   201 FORMAT(2F10.0)
0023600       WRITE(6,203) VOIP(1,5),VOIP(2,5),AZETA(1,5),AZETA(2,5)
0023700   203 FORMAT(' VOIP=',2F10.3,' AZETA=',2F10.3)
0023800       READ(5,204) CASSS,CASSX,CASXS,CASXX,
0023900      *            CAOSS,CAOSX,CAOXS,CAOXX
0024000   204 FORMAT(8F10.0)
0024100       WRITE(6,205) CASSS,CASSX,CASXS,CASXX,
0024200      *            CAOSS,CAOSX,CAOXS,CAOXX
0024300   205 FORMAT(' CASSS,CASSX,CASXS,CASXX,CAOSS,CAOSX,CAOXS,CAOXX=',
0024400      *       8F10.3)
0024500       IF(AN(5).NE.8) GO TO 23
0024600       DO 24 I=1,2
0024700       VOIP(I,5)=VOIP(I,4) ;AZETA(I,5)=AZETA(I,4)
```

```
0024800     24 CONTINUE
0024900     23 CONTINUE
0025000 C------CALCULATE POSITIONS FOR GROUP 0-------
0025100        BLAUE1=BL/AU*(1.+ETA1)
0025200        BLAUE2=BL/AU*(1.+ETA2)
0025300        TVEC(1)=0. ;TVEC(2)=0. ;TVEC(3)=BLAUE1
0025400        CALL POSROT(0.,0.,0.,BL,AU,TVEC,POS0,ROT0)
0025500        DO 30 I=1,3
0025600        DO 30 J=1,4
0025700     30 POS(I,J)=POS0(I,J)
0025800 C------CALCULATE POSITIONS FOR GROUP A-----
0025900        POS(1,5)=0.; POS(2,5)=0. ;POS(3,5)=0.
0026000 C------CALCULATE POSITIONS FOR GROUP 1-------
0026100        READ(5,32) TH,RZ
0026200     32 FORMAT(2F10.0)
0026300        THR=TH*PAI/180.
0026400        WRITE(6,34) TH,RZ
0026500     34 FORMAT(' TH,RZ=',2F10.3)
0026600        TVEC(1)=SIN(THR)*BLAUE2 ;TVEC(2)=0. ;TVEC(3)=COS(THR)*BLAUE2
0026700        CALL POSROT(0.,TH,RZ,BL,AU,TVEC,POS1,ROT1)
0026800        DO 33 I=1,3
0026900        DO 33 J=1,4
0027000     33 POS(I,J+5)=POS1(I,J)
0027100        CALL RWRITE(POS,'POS  ',3,10,10)
0027200        CALL INTGRL
0027300        DO 35 J=1,40
0027400        DO 35 I=1,40
0027500        H(I,J)=S(I,J)
0027600     35 S(I,J)=0.
0027700        DO 36 I=1,40
0027800     36 S(I,I)=1.
0027900 C-------CALCULATION OF H MATRIX--------
0028000 C      DO 408 K=1,NATOMS
0028100 C      DO 408 L=K,NATOMS
0028200 C        LLK=LLIM(K)
0028300 C        LLL=LLIM(L)
0028400 C        ULK=ULIM(K)
0028500 C        ULL=ULIM(L)
0028600 C        NORBK=ULK-LLK+1
0028700 C        NORBL=ULL-LLL+1
0028800 C        DO 409 I=1,NORBK
0028900 C        DO 409 J=1,NORBL
0029000 C          LLKP=LLK+I-1
0029100 C          LLLP=LLL+J-1
0029200 C          LCI=LC(I)+1
0029300 C          LCJ=LC(J)+1
0029400 C          IF(LLKP.EQ.LLLP) GO TO 410
0029500 C          LLKPU=U(LLKP) ;LLLPU=U(LLLP)
0029600 C      NA=AN(5)
0029700 C      IF(NA.EQ.8) GO TO 414
0029800 C      IF(AN(LLKPU).EQ.NA.OR.AN(LLLPU).EQ.NA) GO TO 430
0029900 C 414 CONTINUE
0030000 C      IF(AN(LLKPU).EQ.14.AND.AN(LLLPU).EQ.14) GO TO 411
0030100 C      IF(AN(LLKPU).EQ.8 .AND.AN(LLLPU).EQ.8 ) GO TO 412
0030200 C          IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 416
0030300 C          IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 417
0030400 C      IF((AN(LLKPU).EQ.14.AND.LCI.EQ.1).OR.
0030500 C     *   (AN(LLLPU).EQ.14.AND.LCJ.EQ.1)) GO TO 418
0030600 C          CONST=CXS ;GO TO 413
0030700 C 418      CONST=CSX ;GO TO 413
0030800 C 416      CONST=CSS ;GO TO 413
0030900 C 417      CONST=CXX ;GO TO 413
0031000 C 411      IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 422
0031100 C          IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 423
0031200 C          CONST=CSSSX ;GO TO 413
0031300 C 422      CONST=CSSSS ;GO TO 413
```

```
C 423      CONST=CSSXX ;GO TO 413
C 412      IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 420
C          IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 421
C          CONST=COOSX ;GO TO 413
C 420      CONST=COOSS ;GO TO 413
C 421      CONST=COOXX ;GO TO 413
C 430  IF(AN(LLKPU).EQ.14.OR.AN(LLLPU).EQ.14) GO TO 431
C      IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 432
C      IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 433
C      IF((AN(LLKPU).EQ.NA.AND.LCI.EQ.1).OR.
C     *   (AN(LLLPU).EQ.NA.AND.LCJ.EQ.1)) GO TO 434
C        CONST=CAOXS ;GO TO 413
C 432  CONST=CAOSS ;GO TO 413
C 433  CONST=CAOXX ;GO TO 413
C 434  CONST=CAOSX ;GO TO 413
C 431  IF(LCI.EQ.1.AND.LCJ.EQ.1) GO TO 435
C      IF(LCI.GE.2.AND.LCJ.GE.2) GO TO 436
C      IF((AN(LLKPU).EQ.NA.AND.LCI.EQ.1).OR.
C     *   (AN(LLLPU).EQ.NA.AND.LCJ.EQ.1)) GO TO 437
C        CONST=CASXS ;GO TO 413
C 435  CONST=CASSS ;GO TO 413
C 436  CONST=CASXX ;GO TO 413
C 437  CONST=CASSX ;GO TO 413
C 413      CONTINUE
C          H(LLKP,LLLP)=-S(LLKP,LLLP)*(VOIP(LCI,K)+VOIP(LCJ,L))*
C     *                 CONST/2.
C        GO TO 409
C 410  H(LLKP,LLKP)=-VOIP(LCI,K)
C 409  CONTINUE
C 408  CONTINUE
C      DO 1005 KK=1,3
C      JMIN=16*(KK-1)+1 ;JMAX=MIN(40,16*KK)
C------ROTATION TO DESIRED ATOMIC BASIS-------
       DO 161 J=1,40
       DO 161 I=1,40
  161  ROT(I,J)=0.
       DO 162 J=1,4
       ROT(J+16,J+16)=1.
       DO 162 I=1,4
         ROT(I,J)=ROT0(I,J,1)
         ROT(I+4,J+4)=ROT0(I,J,2)
         ROT(I+8,J+8)=ROT0(I,J,3)
         ROT(I+12,J+12)=ROT0(I,J,4)
         ROT(I+20,J+20)=ROT1(I,J,1)
         ROT(I+24,J+24)=ROT1(I,J,2)
         ROT(I+28,J+28)=ROT1(I,J,3)
  162    ROT(I+32,J+32)=ROT1(I,J,4)
C      WRITE(6,1002) ((ROT(II,JJ),JJ=1,40),II=1,40)
C1002  FORMAT(' ROT=',10F8.3)
       DO 163 I=1,40
       DO 163 J=I,40
         S(J,I)=S(I,J)
  163    H(J,I)=H(I,J)
       DO 164 J=1,40
       DO 164 I=1,40
         SD(I,J)=0. ;HD(I,J)=0.
       DO 164 K=1,40
         SD(I,J)=SD(I,J)+S(I,K)*ROT(K,J)
  164    HD(I,J)=HD(I,J)+H(I,K)*ROT(K,J)
       DO 165 J=1,40
       DO 165 I=1,40
         S(I,J)=0. ;H(I,J)=0.
       DO 165 K=1,40
         S(I,J)=S(I,J)+ROT(K,I)*SD(K,J)
  165    H(I,J)=H(I,J)+ROT(K,I)*HD(K,J)
       RETURN ;END
```

```
      SUBROUTINE TREAD(G)
      COMPLEX G(16,16)
      READ(22) G(1,1) ,G(2,2),G(1,2),G(2,3)
     *       , ((G(I,J),I=5,8),J=5,8)
     *       , ((G(I,J),I=1,4),J=5,8)
     *       , ((G(I,J),I=5,8),J=9,12)
      DO 201 I=3,4
        G(I,I)=G(2,2)
  201   G(1,I)=G(1,2)
      G(2,4)=G(2,3) ;G(3,4)=G(2,3)
      DO 202 I=1,4
        I1=I+1
      DO 202 J=I1,4
  202   G(J,I)=G(I,J)
      DO 203 I=5,8
      DO 203 J=5,8
        G(J+4,I+4)=G(J,I)
  203   G(J+8,I+8)=G(J,I)
      DO 204 I=5,8
      DO 204 J=9,12
        G(I+4,J+4)=G(I,J)
  204   G(I,J+4)=G(I,J)
      DO 205 I=1,4
        G(7,I+12)=-G(7,I+12)
  205   G(I+4,15)=-G(I+4,15)
      DO 206 I=5,8
        G(1,I+4)=G(1,I) ;G(1,I+8)=G(1,I)
        G(2,I+4)=G(4,I) ;G(2,I+8)=G(3,I)
        G(3,I+4)=G(2,I) ;G(3,I+8)=G(4,I)
        G(4,I+4)=G(3,I) ;G(4,I+8)=G(2,I)
  206   CONTINUE
      DO 207 I=1,16
      DO 207 J=I,16
  207   G(J,I)=G(I,J)
      RETURN ;END
      SUBROUTINE RWRITE(A,IC,N,M,MO)
      REAL A
      DIMENSION A(N,M),IC(2)
      WRITE(6,100) IC,N,M,N,MO
  100 FORMAT(1H0,2A4,'(',I3,',',I3,') IN (',I3,',',I3,')',
     *       30('*'))
      KMAX=M/MO
      IF(MO*KMAX.NE.M) KMAX=KMAX+1
      DO 10 K=1,KMAX
      JMIN=1+(K-1)*MO ;JMAX=MIN(M,MO*K)
      WRITE(6,103) (J,J=JMIN,JMAX)
  103 FORMAT(1H ,12I10)
      DO 11 I=1,N
   11 WRITE(6,101) I,(A(I,J),J=JMIN,JMAX)
  101 FORMAT(1H ,I2,12F10.3)
      WRITE(6,102)
  102 FORMAT(' ----------------------------------------')
   10 CONTINUE
      RETURN ;END
      SUBROUTINE CWRITE(A,IC,N,M,MO)
      COMPLEX*8 A
      DIMENSION A(N,M),IC(2)
      WRITE(6,100) IC,N,M,N,MO
  100 FORMAT(1H0,2A4,'(',I3,',',I3,') IN (',I3,',',I3,')',
     *       30('*'))
      KMAX=M/MO
      IF(MO*KMAX.NE.M) KMAX=KMAX+1
      DO 10 K=1,KMAX
      JMIN=1+(K-1)*MO ;JMAX=MIN(M,MO*K)
      WRITE(6,103) (J,J=JMIN,JMAX)
  103 FORMAT(1H ,8I16)
```

```
0044600         DO 11 I=1,N
0044700      11 WRITE(6,101) I,(A(I,J),J=JMIN,JMAX)
0044800     101 FORMAT(1H ,I2,16F8.3)
0044900         WRITE(6,102)
0045000     102 FORMAT(' ----------------------------------------')
0045100      10 CONTINUE
0045200         RETURN ;END
0045300         SUBROUTINE TWRITE(A,IC,N,M,L,MO)
0045400         COMPLEX*8 A
0045500         DIMENSION A(N,M,L),IC(2)
0045600         WRITE(6,100) IC,N,M,L,N,MO
0045700     100 FORMAT(1H0,2A4,'(',I3,',',I3,') BY (',
0045800        *        I3,',',I3,')',30('*'))
0045900         KMAX=M/MO
0046000         IF(MO*KMAX.NE.M) KMAX=KMAX+1
0046100         DO 12 KK=1,L
0046200         DO 10 K=1,KMAX
0046300         JMIN=1+(K-1)*MO ;JMAX=MIN(M,MO*K)
0046400         WRITE(6,103) (J,J=JMIN,JMAX)
0046500     103 FORMAT(1H ,8I16)
0046600         DO 11 I=1,N
0046700      11 WRITE(6,101) I,(A(I,J,KK),J=JMIN,JMAX)
0046800     101 FORMAT(1H ,I2,16F8.3)
0046900         WRITE(6,102)
0047000     102 FORMAT(' ----------------------------------------')
0047100      10 CONTINUE
0047200      12 CONTINUE
0047300         RETURN ;END
0047400         SUBROUTINE RMAT(A,IC,N,M,NO,MO)
0047500         REAL A
0047600         DIMENSION A(N,M),IC(2)
0047700         WRITE(6,100) IC,N,M,NO,MO
0047800     100 FORMAT(1H0,2A4,'(',I3,',',I3,') IN (',I3,',',I3,')',
0047900        *        30('*'))
0048000         KMAX=M/MO
0048100         IF(MO*KMAX.NE.M) KMAX=KMAX+1
0048200         DO 10 K=1,KMAX
0048300         JMIN=1+(K-1)*MO ;JMAX=MIN(M,MO*K)
0048400         WRITE(6,103) (J,J=JMIN,JMAX)
0048500     103 FORMAT(1H ,16I8)
0048600         DO 11 I=1,NO
0048700      11 WRITE(6,101) I,(A(I,J),J=JMIN,JMAX)
0048800     101 FORMAT(1H ,I2,16F8.3)
0048900         WRITE(6,102)
0049000     102 FORMAT(' ----------------------------------------')
0049100      10 CONTINUE
0049200         RETURN ;END
0049300         SUBROUTINE CMAT(A,IC,N,M,NO,MO)
0049400         COMPLEX*8 A
0049500         DIMENSION A(N,M),IC(2)
0049600         WRITE(6,100) IC,N,M,NO,MO
0049700     100 FORMAT(1H0,2A4,'(',I3,',',I3,') IN (',I3,',',I3,')',
0049800        *        30('*'))
0049900         KMAX=M/MO
0050000         IF(MO*KMAX.NE.M) KMAX=KMAX+1
0050100         DO 10 K=1,KMAX
0050200         JMIN=1+(K-1)*MO ;JMAX=MIN(M,MO*K)
0050300         WRITE(6,103) (J,J=JMIN,JMAX)
0050400     103 FORMAT(1H ,8I16)
0050500         DO 11 I=1,NO
0050600      11 WRITE(6,101) I,(A(I,J),J=JMIN,JMAX)
0050700     101 FORMAT(1H ,I2,16F8.3)
0050800         WRITE(6,102)
0050900     102 FORMAT(' ----------------------------------------')
0051000      10 CONTINUE
0051100         RETURN ;END
```

```
051200          SUBROUTINE RWRIT(A,IC,M,MO)
051300          DIMENSION A(1),IC(2)
051400          WRITE(6,100) IC,MO
051500      100 FORMAT(1H0,2A4,' IN (',I3,')')
051600          KMAX=M/MO
051700          IF(MO*KMAX.NE.M) KMAX=KMAX+1
051800          DO 10 K=1,KMAX
051900          JMIN=1+(K-1)*MO ;JMAX=MIN(M,MO*K)
052000          WRITE(6,103) (J,J=JMIN,JMAX)
052100      103 FORMAT(1H ,16I8)
052200          WRITE(6,101) (A(J),J=JMIN,JMAX)
052300      101 FORMAT(1H ,2X,16F8.3)
052400          WRITE(6,102)
052500      102 FORMAT(' ----------------------------------------')
052600       10 CONTINUE
052700          RETURN ;END
052800          SUBROUTINE DWRIT(A,IC,M,MO)
052900          REAL*8 A
053000          DIMENSION A(1),IC(2)
053100          WRITE(6,100) IC,MO
053200      100 FORMAT(1H0,2A4,' IN (',I3,')')
053300          KMAX=M/MO
053400          IF(MO*KMAX.NE.M) KMAX=KMAX+1
053500          DO 10 K=1,KMAX
053600          JMIN=1+(K-1)*MO ;JMAX=MIN(M,MO*K)
053700          WRITE(6,103) (J,J=JMIN,JMAX)
053800      103 FORMAT(1H ,16I8)
053900          WRITE(6,101) (A(J),J=JMIN,JMAX)
054000      101 FORMAT(1H ,2X,16F8.3)
054100          WRITE(6,102)
054200      102 FORMAT(' ----------------------------------------')
054300       10 CONTINUE
054400          RETURN ;END

R GEEO          OF  A1402.I.FORT  LIST END
```

ER DGEEO OF A1402.I.FORT

```
0000010 -3.        0.          0.1         0.04        0.05
0000020   1
0000100 14.95      7.78        29.6        12.7
0000200 1.6344     1.4284      2.2460       2.227
0000300 .87        1.69        2.5
0000400 2.5        4.3         6.0
0000500 1.34       2.1         2.1         2.9
0000600  0 0.        0.
0000700
0000800
0000900
0001000 120.        0.
```

ER DGEEO OF A1402.I.FORT LIST END

ER PERFIN OF A1402.I.FORT

```
0000100        PROGRAM PERFIN
0000200        IMPLICIT COMPLEX*8 (G,V,Z)
0000300        COMMON/ARRAYS/S(40,40),Y(9,5,203),AZ(17,45)
0000400        COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0000500        COMMON /H/H(40,40)
0000510        REAL*4 VYS
0000520        DIMENSION GD(8,8)
0000600        DIMENSION GOUT(12,12,20),GAV(12,12,20)
0000700       *,          GYYY(8,8),GSSSK(4,4),GSSSM1(4,4),GYYYM1(8,8),
0000800       *           GSURF(4,4,40),
0000900       *           GPYYK(8,8),GPSSK(4,4),GPSYK(4,8),GPYSK(8,4)
0001000        DIMENSION SYS(8,4),VYS(8,4),VYSD(8,4),RG3M(130),RG4M(130),
0001100       *           ZW4(4),ZW8(8),IP4(4),IP8(8)
0001200        DIMENSION AV1(2),AV2(2),XV(2),YV(2),AKV(2)
0001210        DIMENSION VGV(8,8)
0001300        DATA AV1/-.6124,.3535/,AV2/-.6124,-.3535/,
0001400       *     XV/-.8165,1.4142/,YV/-1.4142,-.8165/
0001500        DATA AU/.529167/,PAI/3.141592/,BL/1.61/
0001505        XV(1)=XV(1) ;YV(1)=YV(1)
0001510        REWIND 21
0001515        READ(5,11) ISKIP,NKEND
0001520     11 FORMAT(2I3)
0001525        IF(ISKIP.EQ.0) GO TO 13
0001530        DO 12 IS=1,ISKIP
0001535        READ(21) (((GSURF(I,J,NUME),I=1,4),J=1,4),NUME=1,NN),
0001540       *           (RG3M(I),RG4M(I),I=1,NN)
0001545     12 CONTINUE
0001550     13 CONTINUE
0001560        WRITE(6,8)
0001570      8 FORMAT(' INPUT EI,EF,DE,DELTA,EPSG IN 5F10.0')
0001580        EI=-11. ;EF=0. ;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
0001590        READ(5,9) DEI,DEF,DDE,DDELTA,DEPSG
0001600      9 FORMAT(5F10.0)
0001610        IF(DEI.NE.0.) EI=DEI
0001620        IF(DEF.NE.0.) EF=DEF
0001630        IF(DDE.NE.0.) DE=DDE
0001640        IF(DDELTA.NE.0.) DELTA=DDELTA
0001650        IF(DEPSG.NE.0.) EPSG=DEPSG
0001660        NN=(EF-EI+0.01)/DE ;NN=NN+1
0001800 C------SET UP S AND H-------
0001900        CALL SHBAB(IAN5)
0001910        DO 6 J=1,4
0001920        DO 6 I=1,4
0001930        H(I,J+20)=0.
0001940      6 H(I+20,J)=0.
0002000 C      CALL RWRITE(H,'H     ',40,40,10)
0002100 C      CALL RWRITE(S,'S     ',40,40,10)
0002200 C------MAKE VYS AND SYS-------
0002300        DO 400 J=1,4
0002400        DO 400 I=1,4
0002500          VYS(I,J)=H(I,J+20) ;SYS(I,J)=S(I,J+20)
0002600          VYS(I+4,J)=H(I+16,J+20) ;SYS(I+4,J)=S(I+16,J+20)
0002700    400 CONTINUE
0002800 C      CALL RWRITE(VYS,'VYS   ',8,4,4)
0002900 C      CALL RWRITE(SYS,'SYS   ',8,4,4)
0003000 C------SCANNING OF K-------
0003100        NUMK=0
0003110        DO 1 K1=1,12
```

```
003120         AKV(1)=-1.62+0.28*(K1-1)
003130         DO 1 K2=1,14
003140         AKV(2)=-1.88+0.28*(K2-1)
003150         IF(AKV(2).LT.(0.5774*AKV(1)-1.8856).OR.
003160        *    AKV(2).LT.(-0.5774*AKV(1)-1.8856).OR.
003170        *    AKV(2).GT.(0.5774*AKV(1)+1.8856).OR.
003180        *    AKV(2).GT.(-0.5774*AKV(1)+1.8856)) GO TO 1
003900           AK1=PAI*(AV1(1)*AKV(1)+AV1(2)*AKV(2))
004000           AK2=PAI*(AV2(1)*AKV(1)+AV2(2)*AKV(2))
004100         NUMK=NUMK+1
004110         IF(NUMK.LE.ISKIP.OR.NUMK.GT.NKEND) GO TO 1
004120      10 WRITE(6,105) AK1,AK2,K1,K2,NUMK,AKV
004180     105 FORMAT(' AK1,AK2,K1,K2,NUMK,AKV=',2F8.3,3I3,2F8.3)
004400 C------REWIND GEEO FILE------
004500         REWIND 23
004600 C------READ-IN SURF-------
004700         READ(21) (((GSURF(I,J,NUME),I=1,4),J=1,4),NUME=1,NN),
004800        *           (RG3M(I),RG4M(I),I=1,NN)
004900         RG3M(1)=RG3M(1) ;RG4M(1)=RG4M(1)
005000 C------SCANNING OF ENERGY-----
005100         E=EI-DE ;NUME=0
005200 C------CHANGE ENERGY------
005300      40 NUME=NUME+1
005400 C       IF(E.GE.-6.01.AND.E.LT.-4.01) DE=0.05
005500         E=E+DE
005600 C       IF(E.GT.-5.99.AND.E.LT.-3.99) DELTA=0.02
005700         IF(E.GT.EF+0.1*DE) GO TO 2
005800         ZE=E+(0.,1.)*DELTA
005900 C------READ-IN GEEO(GYYY)------
006000         READ(23) ((GYYY(I,J),J=I,8),I=1,8)
006100         DO 410 I=1,7
006200         I1=I+1
006300         DO 410 J=I1,8
006400     410 GYYY(J,I)=GYYY(I,J)
006500 C       CALL CWRITE(GYYY,'GYYY  ',8,8,8)
006600 C------MAKE VYSD AND GSSS------
006700         DO 420 J=1,4
006800         DO 420 I=1,8
006900     420 VYSD(I,J)=VYS(I,J)-ZE*SYS(I,J)
007000 C       CALL CWRITE(VYSD,'VYSD  ',8,4,4)
007100         DO 421 J=1,4
007200         DO 421 I=1,4
007300           GSSSK(I,J)=GSURF(I,J,NUME)
007400     421   GSSSM1(I,J)=GSSSK(I,J)
007500 C       CALL CWRITE(GSSSK,'GSSSK',4,4,4)
007600 C------MAKE GYYYM1 AND GSSSM1------
007700         DO 430 J=1,8
007800         DO 430 I=1,8
007900     430 GYYYM1(I,J)=GYYY(I,J)
008000         CALL CINV(GYYYM1,8,0,8,8,1.D-14,ZDET,ZW8,IP8,NSTOP)
008100         CALL CINV(GSSSM1,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
008200 C       CALL CWRITE(GSSSM1,'GSSSM1',4,4,4)
008300 C       CALL CWRITE(GYYYM1,'GYYYM1',8,8,8)
008400 C------CALCULATE GPYYK AND GPSSK------
008405         DO 439 J=1,4
008410         DO 439 I=1,8
008415           GD(I,J)=0.
008420         DO 439 L=1,4
008425     439   GD(I,J)=GD(I,J)+VYSD(I,L)*GSSSK(L,J)
008430         DO 440 J=1,8
008435         DO 440 I=1,8
008440           GPYYK(I,J)=GYYYM1(I,J)
008445         DO 440 L=1,4
008450     440   GPYYK(I,J)=GPYYK(I,J)-GD(I,L)*VYSD(J,L)
008500 C       DO 440 J=1,8
008600 C       DO 440 I=1,8
```

```
0008700 C         GPYYK(I,J)=GYYYM1(I,J)
0008800 C       DO 440 K=1,4
0008900 C       DO 440 L=1,4
0009000 C 440     GPYYK(I,J)=GPYYK(I,J)-VYSD(I,L)*GSSSK(L,K)*VYSD(J,K)
0009100         CALL CINV(GPYYK,8,0,8,8,1.D-14,ZDET,ZW8,IP8,NSTOP)
0009200 C       CALL CWRITE(GPYYK,'GPYYK',8,8,8)
0009205         DO 441 J=1,8
0009210         DO 441 I=1,4
0009215           GD(I,J)=0.
0009220         DO 441 L=1,8
0009225    441  GD(I,J)=GD(I,J)+VYSD(L,I)*GYYY(L,J)
0009230         DO 442 J=1,4
0009235         DO 442 I=1,4
0009240           GPSSK(I,J)=GSSSM1(I,J)
0009245         DO 442 L=1,8
0009250    442    GPSSK(I,J)=GPSSK(I,J)-GD(I,L)*VYSD(L,J)
0009300 C       DO 441 J=1,4
0009400 C       DO 441 I=1,4
0009500 C         GPSSK(I,J)=GSSSM1(I,J)
0009600 C       DO 441 K=1,8
0009700 C       DO 441 L=1,8
0009800 C 441     GPSSK(I,J)=GPSSK(I,J)-VYSD(L,I)*GYYY(L,K)*VYSD(K,J)
0009900         CALL CINV(GPSSK,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0010000 C       CALL CWRITE(GPSSK,'GPSSK',4,4,4)
0010100 C------CALCULATE GPYSK AND GPSYK------
0010105         DO 450 J=1,4
0010110         DO 450 I=1,8
0010115           GD(I,J)=0.
0010120         DO 450 L=1,8
0010125    450    GD(I,J)=GD(I,J)+GYYY(I,L)*VYSD(L,J)
0010130         DO 451 J=1,4
0010135         DO 451 I=1,8
0010140         GPYSK(I,J)=0.
0010145         DO 451 L=1,4
0010150    451  GPYSK(I,J)=GPYSK(I,J)+GD(I,L)*GPSSK(L,J)
0010200 C       DO 451 J=1,4
0010300 C       DO 451 I=1,8
0010400 C         GPYSK(I,J)=0.
0010500 C       DO 451 K=1,4
0010600 C       DO 451 L=1,8
0010700 C 451     GPYSK(I,J)=GPYSK(I,J)+GYYY(I,L)*VYSD(L,K)*GPSSK(K,J)
0010800 C       CALL CWRITE(GPYSK,'GPYSK',8,4,4)
0010805         DO 452  J=1,8
0010810         DO 452 I=1,4
0010815           GD(I,J)=0.
0010820         DO 452 L=1,4
0010825    452    GD(I,J)=GD(I,J)+GSSSK(I,L)*VYSD(J,L)
0010830         DO 458 J=1,8
0010835         DO 458 I=1,4
0010840           GPSYK(I,J)=0.
0010845         DO 458 L=1,8
0010850    458    GPSYK(I,J)=GPSYK(I,J)+GD(I,L)*GPYYK(L,J)
0010900 C       DO 452 J=1,8
0011000 C       DO 452 I=1,4
0011100 C         GPSYK(I,J)=0.
0011200 C       DO 452 K=1,8
0011300 C       DO 452 L=1,4
0011400 C 452     GPSYK(I,J)=GPSYK(I,J)+GSSSK(I,L)*VYSD(K,L)*GPYYK(K,J)
0011500 C       CALL CWRITE(GPSYK,'GPSYK',4,8,8)
0011600 C------STORE IN GAV------
0011700         DO 453 J=1,8
0011800         DO 453 I=1,8
0011900    453    GAV(I,J,NUME)=GPYYK(I,J)
0012000         DO 454 J=1,4
0012100         DO 454 I=1,4
0012200    454    GAV(I+8,J+8,NUME)=GPSSK(I,J)
```

```
012300         DO 455 J=1,4
012400         DO 455 I=1,8
012500           GAV(I,J+8,NUME)=GPYSK(I,J)
012600    455    GAV(J+8,I,NUME)=GPSYK(J,I)
012610         DO 456 I=1,12
012620    456  RG3M(I)=AIMAG(GAV(I,I,NUME))
012630         WRITE(6,457) E,(RG3M(I),I=1,12)
012640    457 FORMAT(1H ,13F8.3)
012650         GO TO 40
012700 C------WRITE OUT-----
012710       2 CONTINUE
012800 C       CALL TWRITE(GAV,'GAV  ',12,12,20,8)
012900 C------AVERAGING------
013100         IF(NUMK.NE.1) GO TO 461
013110         REWIND 20
013200         WRITE(20) (((GAV(I,J,NUME),I=1,12),J=1,12),NUME=1,NN)
013300    461 CONTINUE
013400         REWIND 20
013500         READ(20) (((GOUT(I,J,NUME),I=1,12),J=1,12),NUME=1,NN)
013600         NUMK1=NUMK-1
013700         DO 460 NUME=1,NN
013800         DO 460 J=1,12
013900         DO 460 I=1,12
014000    460    GAV(I,J,NUME)=(GAV(I,J,NUME)+GOUT(I,J,NUME)*NUMK1)/NUMK
014100 C------WRITE ON FILE PERFIN------
014110         REWIND 20
014200         WRITE(20) (((GAV(I,J,NUME),I=1,12),J=1,12),NUME=1,NN)
014300       1 CONTINUE
014400         STOP 0002 ;END

ER  PERFIN     OF  A1402.I.FORT  LIST END
```

ER DPERF OF A1402.I.FORT

```
000010  71150
000020 -5.5          -3.6          0.1          0.04          0.05
000100 14.95         7.78         29.6         12.7
000200 1.6344        1.4284       2.2460        2.227
000300 .87           1.69         2.5
000400 2.5           4.3          6.0
000500 1.34          2.1          2.1          2.9
000600  8
000700
000800
000900
001000 180.          0.
```

ER DPERF OF A1402.I.FORT LIST END

ER DANGLE OF A1402.I.FORT

```
0000100        PROGRAM DANGLE
0000200        IMPLICIT COMPLEX*8 (G,V,Z)
0000300        COMMON/ARRAYS/S(40,40),Y(9,5,203),AZ(17,45)
0000400        COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0000500        COMMON /H/H(40,40)
0000600        REAL*4 VYS
0000700        DIMENSION SYS(8,4),VYS(8,4),VYSD(8,4),
0000800       *           GPPP(12,12,20),GPSSM1(4,4),GDSS(4,4)
0000900       *,           ZW4(4),IP4(4)
0001000        DIMENSION GYYY(8,8)
0001100        DATA AU/.529167/,PAI/3.141592/,BL/1.61/
0001110        REWIND 20
0001120        WRITE(6,8)
0001130      8 FORMAT(' INPUT EI,EF,DE,DELTA,EPSG IN 5F10.0')
0001140        EI=-5.5 ;EF=-3.6 ;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
0001150        READ(5,9) DEI,DEF,DDE,DDELTA,DEPSG
0001160      9 FORMAT(5F10.0)
0001170        IF(DEI.NE.0.) EI=DEI
0001180        IF(DEF.NE.0.) EF=DEF
0001190        IF(DDE.NE.0.) DE=DDE
0001200        IF(DDELTA.NE.0.) DELTA=DDELTA
0001210        IF(DEPSG.NE.0.) EPSG=DEPSG
0001220        NN=(EF-EI+0.01)/DE ;NN=NN+1
0001400 C------SET UP S AND H-------
0001500        CALL SHBAB(IAN5)
0001510        DO 6 J=1,4
0001520        DO 6 I=1,4
0001530        H(I,J+20)=0.
0001540      6 H(I+20,J)=0.
0001600 C      CALL RWRITE(H,'H     ',40,40,10)
0001700 C      CALL RWRITE(S,'S     ',40,40,10)
0001800 C------MAKE VYS AND SYS-------
0001900        DO 400 J=1,4
0002000        DO 400 I=1,4
0002100          VYS(I,J)=H(I,J+20) ;SYS(I,J)=S(I,J+20)
0002200          VYS(I+4,J)=H(I+16,J+20) ;SYS(I+4,J)=S(I+16,J+20)
0002300    400 CONTINUE
0002400 C      CALL RWRITE(VYS,'VYS   ',8,4,4)
0002500 C      CALL RWRITE(SYS,'SYS   ',8,4,4)
0002600 C------READ-IN PERFIN------
0002700        READ(24) (((GPPP(I,J,NUME),I=1,12),J=1,12),NUME=1,NN)
0002800 C------SCANNING OF ENERGY-----
0002900        E=EI-DE ;NUME=0
0003000 C------CHANGE ENERGY------
0003100     40 NUME=NUME+1
0003200 C      IF(E.GE.-6.01.AND.E.LT.-4.01) DE=0.05
0003300        E=E+DE
0003400 C      IF(E.GT.-5.99.AND.E.LT.-3.99) DELTA=0.02
0003500        IF(E.GT.EF+0.1*DE) GO TO 2
0003600        ZE=E+(0.,1.)*DELTA
0003700 C------READ-IN GEEO(GYYY)------
0003800        READ(23) ((GYYY(I,J),J=I,8),I=1,8)
0003900        DO 410 I=1,7
0004000        I1=I+1
0004100        DO 410 J=I1,8
0004200    410 GYYY(J,I)=GYYY(I,J)
0004300 C      CALL CWRITE(GYYY,'GYYY  ',8,8,8)
0004400 C------MAKE VYSD AND GPSSM1------
```

```
0004500         DO 420 J=1,4
0004600         DO 420 I=1,8
0004700     420 VYSD(I,J)=VYS(I,J)-ZE*SYS(I,J)
0004710 C       CALL CWRITE(VYSD,'VYSD ',8,4,4)
0004800         DO 422 J=1,4
0004900         DO 422 I=1,4
0005000     422 GPSSM1(I,J)=GPPP(I+8,J+8,NUME)
0005100         CALL CINV(GPSSM1,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0005110 C       CALL CWRITE(GPSSM1,'GPSSM1',4,4,4)
0005200 C------CALCULATE GDSS------
0005300         DO 423 J=1,4
0005400         DO 423 I=1,4
0005500         GDSS(I,J)=GPSSM1(I,J)
0005600         DO 423 K=1,8
0005700         DO 423 L=1,8
0005800     423 GDSS(I,J)=GDSS(I,J)+VYSD(L,I)*GYYY(L,K)*VYSD(K,J)
0005900         CALL CINV(GDSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0005910 C       CALL CWRITE(GDSS,'GDSS ',4,4,4)
0006000 C------OUTPUT ON FILE GDSS------
0006010         WRITE(6,424) E,(GDSS(I,I),I=1,4)
0006020     424 FORMAT(1H ,9F8.3)
0006100         WRITE(20) ((GDSS(I,J),I=1,4),J=1,4)
0006200         GO TO 40
0006300       2 STOP ;END
0006400         PROGRAM PLDANG
0006500         COMPLEX*8 GDSS(4,4)
0006600         DIMENSION EM(132),R(132),RM(130,2)
0006700         NN=20 ;NN1=NN+1 ;NN2=NN+2 ;EI=-5.5
0006800 C------MAKE EM---------
0006900 C       DO 14 I=1,51
0007000 C    14 EM(I)=-11.+(I-1)*0.1
0007100 C       DO 15 I=1,40
0007200 C    15 EM(I+51)=-5.95+(I-1)*0.05
0007300 C       DO 16 I=1,39
0007400 C    16 EM(I+91)=-3.9+(I-1)*0.1
0007500         DO 14 I=1,NN
0007600     14 EM(I)=-5.5+(I-1)*0.1
0007700         DO 10 NUME=1,NN
0007800 C------READ-IN GDSS------
0007900         READ(26) ((GDSS(I,J),I=1,4),J=1,4)
0008000 C------MAKE RM------
0008100         RM(NUME,1)=-AIMAG(GDSS(1,1))
0008200         RM(NUME,2)=-AIMAG(GDSS(2,2))
0008300     10 CONTINUE
0008400         XM=NN*0.2 ;YM=20. ;XM2=XM+2.
0008500         CALL PLOTS(0.,0.,'SURF')
0008600         CALL PLOT(2.,2.,-3)
0008700         R(NN1)=0. ;R(NN2)=0.5 ;EM(NN1)=EI ;EM(NN2)=0.5
0008800         DO 2 K=1,2
0008900         CALL AXIS(0.,0.,'ENERGY',-6,XM,0.,EI,0.5)
0009000         CALL AXIS(0.,0.,'DANGLE',6,YM,90.,0.,0.5)
0009100         DO 4 I=1,NN
0009200         R(I)=-RM(I,K)
0009300       4 CONTINUE
0009400         CALL LINE(EM,R,NN,1,0,0)
0009500         CALL PLOT(XM2,0.,-3)
0009600       2 CONTINUE
0009700         CALL PLOTV
0009800         STOP ;END

R DANGLE    OF  A1402.I.FORT  LIST END
```

ER DDANG JF A1402.I.FORT

```
000010 -5.5        -3.6          0.1          0.04        0.05
000100 14.95       7.78         29.6         12.7
000200 1.6344      1.4284        2.2460       2.227
000300 .87         1.69          2.5
000400 2.5         4.3           6.0
000500 1.34        2.1           2.1          2.9
000600  8
000700
000800
000900
001000 180.        0.
```

ER DDANG JF A1402.I.FORT LIST END

R OVAC OF A1402.I.FORT

```
000100        PROGRAM OVAC
000200        IMPLICIT COMPLEX*8 (G,V,Z)
000300        COMMON/ARRAYS/S(40,40),Y(9,5,203),AZ(17,45)
000400        COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
000500        COMMON /H/H(40,40)
000600        DIMENSION GBBB(16,16),GBBBM1(16,16),GDSS(4,4),
000700       *           GDSSM1(4,4),VBSD(16,4),GVBB(16,16)
000800       *,           GVSS(4,4),ZW4(4),ZW16(16),IP4(4),IP16(16)
000900       *,          OUT(20,130)
001000        DATA AU/.529167/,PAI/3.141592/,BL/1.61/
001010        WRITE(6,8)
001020      8 FORMAT(' INPUT EI,EF,DE,DELTA,EPSG IN 5F10.0')
001030        EI=-5.5 ;EF=-3.6 ;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
001040        READ(5,9) DEI,DEF,DDE,DDELTA,DEPSG
001050      9 FORMAT(5F10.0)
001060        IF(DEI.NE.0.) EI=DEI
001070        IF(DEF.NE.0.) EF=DEF
001080        IF(DDE.NE.0.) DE=DDE
001090        IF(DDELTA.NE.0.) DELTA=DDELTA
001100        IF(DEPSG.NE.0.) EPSG=DEPSG
001110        NN=(EF-EI+0.01)/DE ;NN=NN+1
001300 C------SET UP S AND H-------
001400        CALL SHBAB(IAN5)
001410 C      DO 6 J=1,4
001420 C      DO 6 I=1,4
001430 C      H(I,J+20)=0.
001440 C    6 H(I+20,J)=0.
001450 C      CALL RWRITE(H,'H     ',40,40,10)
001460 C      CALL RWRITE(S,'S     ',40,40,10)
001500 C------SCANNING OF ENERGY-----
001600        E=EI-DE ;NUME=0
001700 C------CHANGE ENERGY------
001800     40 NUME=NUME+1
002000        E=E+DE
002200        IF(E.GT.EF+0.1*DE) GO TO 2
002210        ZE=E+(0.,1.)*DELTA
002300 C------READ-IN GBBB AND GDSS------
002400        CALL TREAD(GBBB)
002500        READ(26) ((GDSS(I,J),I=1,4),J=1,4)
002510 C      CALL CWRITE(GDSS,'GDSS ',4,4,4)
002600 C------MAKE VBSD------
002700        DO 420 J=1,4
002800        DO 420 I=1,16
002900    420 VBSD(I,J)=H(I,J+20)-ZE*S(I,J+20)
002902        DO 421 J=1,4
002904        DO 421 I=5,16
002906    421 VBSD(I,J)=0.
002910 C      CALL CWRITE(VBSD,'VBSD ',16,4,4)
003000 C------MAKE GBBBM1 AND GDSSM1------
003100        DO 430 J=1,16
003200        DO 430 I=1,16
003300    430 GBBBM1(I,J)=GBBB(I,J)
003400        CALL CINV(GBBBM1,16,0,16,16,1.D-14,ZDET,ZW16,IP16,NSTOP)
003410 C      CALL CWRITE(GBBBM1,'GBBBM1',16,16,8)
003500        DO 431 J=1,4
003600        DO 431 I=1,4
003700    431 GDSSM1(I,J)=GDSS(I,J)
003800        CALL CINV(GDSSM1,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
```

```
003810 C      CALL CWRITE(GDSSM1,'GDSSM1',4,4,4)
003900 C------CALCULATE GVBB AND GVSS------
004000        DO 440 J=1,16
004100        DO 440 I=1,16
004200        GVBB(I,J)=GBBBM1(I,J)
004300        DO 440 K=1,4
004400        DO 440 L=1,4
004500   440  GVBB(I,J)=GVBB(I,J)-VBSD(I,L)*GDSS(L,K)*VBSD(J,K)
004600        CALL CINV(GVBB,16,0,16,16,1.D-14,ZDET,ZW16,IP16,NSTOP)
004610 C      CALL CWRITE(GVBB,'GVBB ',16,16,8)
004700        DO 441 J=1,4
004800        DO 441 I=1,4
004900        GVSS(I,J)=GDSSM1(I,J)
005000        DO 441 K=1,16
005100        DO 441 L=1,16
005200   441  GVSS(I,J)=GVSS(I,J)-VBSD(L,I)*GBBB(L,K)*VBSD(K,J)
005300        CALL CINV(GVSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
005310 C      CALL CWRITE(GVSS,'GVSS ',4,4,4)
005400 C------STORE-IN OUT------
005500        DO 450 I=1,16
005600   450  OUT(I,NUME)=AIMAG(GVBB(I,I))
005700        DO 451 I=1,4
005800   451  OUT(I+16,NUME)=AIMAG(GVSS(I,I))
005810        WRITE(6,452) E,(OUT(I,NUME),I=1,20)
005820   452 FORMAT(' E=',F10.3/1H ,16F7.3/1H ,4F8.3)
005900        GO TO 40
006000 C------OUTPUT ON FILE OVAC------
006100     2  CONTINUE
006200        WRITE(20) ((OUT(I,NUME),I=1,20),NUME=1,NN)
006300        STOP ;END
006400        PROGRAM PLOVAC
006500        DIMENSION EM(132),R(132),RM(130,20)
006600        NN=20 ;NN1=NN+1 ;NN2=NN+2 ;EI=-5.5
006700 C------MAKE EM---------
006800 C      DO 14 I=1,51
006900 C   14 EM(I)=-11.+(I-1)*0.1
007000 C      DO 15 I=1,40
007100 C   15 EM(I+51)=-5.95+(I-1)*0.05
007200 C      DO 16 I=1,39
007300 C   16 EM(I+91)=-3.9+(I-1)*0.1
007400        DO 14 I=1,NN
007500    14 EM(I)=-5.5+(I-1)*0.1
007600 C------READ-IN OVAC------
007700        READ(27) ((RM(NUME,I),I=1,20),NUME=1,NN)
007800        XM=NN*0.2 ;YM=20. ;XM2=XM+2.
007900        CALL PLOTS(0.,0.,'SURF')
008000        CALL PLOT(2.,2.,-3)
008100        R(NN1)=0. ;R(NN2)=0.5 ;EM(NN1)=EI ;EM(NN2)=0.5
008200        DO 2 K=1,20
008300        CALL AXIS(0.,0.,'ENERGY',-6,XM,0.,EI,0.5)
008400        CALL AXIS(0.,0.,'OVAC',4,YM,90.,0.,0.5)
008500        DO 4 I=1,NN
008600        R(I)=-RM(I,K)
008700     4  CONTINUE
008800        CALL LINE(EM,R,NN,1,0,0)
008900        CALL PLOT(XM2,0.,-3)
009000     2  CONTINUE
009100        CALL PLOTV
009200        STOP ;END
009300        PROGRAM PLUNI
009400        DIMENSION E(132),R(132),RM(130,20),NOUT(20)
009500        INTEGER*8 IP(10)
009600        E=999.
009700        WRITE(6,100)
009800   100 FORMAT(' INPUT NN,KIND,EI,DE IN 2I3,2F10.0')
009900        READ(5,101) NNN,NKIND,DEI,DDE
```

```
010000   101 FORMAT(2I3,2F10.0)
010100       NN=110 ;KIND=20
010200       IF(NNN.NE.0) NN=NNN
010300       IF(NKIND.NE.0) KIND=NKIND
010400 C------READ-IN-------
010410       REWIND 30
010500       READ(30) ((RM(NUME,I),I=1,KIND),NUME=1,NN)
010600       NN1=NN+1 ;NN2=NN+2
010700 C------MAKE E--------
010800       EI=-11. ;DE=0.1
010900       IF(DEI.NE.0.) EI=DEI
011000       IF(DDE.NE.0.) DE=DDE
011100       DO 20 I=1,NN
011200    20   E(I)=EI+(I-1)*DE
011300       CALL PLOTS(0.,0.,'S')
011400       CALL PLOT(2.,2.,-3)
011500 C------DX,DY-------
011600       WRITE(6,104)
011700   104 FORMAT(' INPUT DX,DY PER ONE CM')
011800       READ(5,105) DDX,DDY
011900   105 FORMAT(2F10.0)
012000       DX=0.5 ;DY=0.5
012100       IF(DDX.NE.0.) DX=DDX
012200       IF(DDY.NE.0.) DY=DDY
012300       YMAX=20./DY ;XMAX=NN*DE/DX ;XMAX2=XMAX+2.
012400       IC=0
012500       R(NN1)=0. ;R(NN2)=DY ;E(NN1)=EI ;E(NN2)=DX
012600     2 CONTINUE
012700       IC=IC+1
012800       WRITE(6,102) IC
012900   102 FORMAT(' INPUT NOUT IN 20I2 ;GRAPH #=',I3)
013000       READ(5,103) (NOUT(I),I=1,20)
013100   103 FORMAT(20I2)
013200       IF(NOUT(1).EQ.0) GO TO 6
013300       DO 10 I=1,20
013400       IF(NOUT(I).NE.0) GO TO 10
013500       LNOUT=I-1 ;GO TO 11
013600    10 CONTINUE
013700    11 CONTINUE
013800       WRITE(6,106) (NOUT(K),K=1,LNOUT)
013900   106 FORMAT(' INPUT TYTLE OF #',20I3)
014000       READ(5,107) (IP(I),I=1,10)
014100   107 FORMAT(10A8)
014200       CALL SYMBOL(0.,0.,0.3,IP(1),0.,8)
014210       DO 3 I=2,7
014300       CALL SYMBOL(S,S,0.3,IP(I),0.,8)
014310     3 CONTINUE
014400       CALL PLOT(0.,2.,-3)
014500       CALL AXIS(0.,0.,'ENERGY',-6,XMAX,0.,EI,DX)
014600       CALL AXIS(0.,0.,'LDOS',4,YMAX,90.,0.,DY)
014700       DO 4 I=1,NN
014800         R(I)=0.
014900         DO 5 K=1,LNOUT
015000     5   R(I)=R(I)-RM(I,NOUT(K))
015100         IF(R(I).GT.20.) R(I)=20.
015200     4 CONTINUE
015300       CALL LINE(E,R,NN,1,0,0)
015400       CALL PLOT(XMAX2,-2.,-3)
015500       GO TO 2
015550     6 CONTINUE
015600       CALL PLOTV
015700       STOP ;END

R OVAC       OF  A1402.I.FORT  LIST END
```

R DOVAC OF A1402.I.FORT

```
000100 -5.5        -3.6          0.1          0.04        0.05
000200 14.95       7.78         29.6         12.7
000300 1.6344      1.4284       2.2460        2.227
000400 .87         1.69         2.5
000500 2.5         4.3          6.0
000600 1.34        2.1          2.1          2.9
000700  8 -.2          0.
000800
000900
001000
001100 180.         0.0
```

R DOVAC OF A1402.I.FORT LIST END

ER OHCL OF A1402.I.FORT

```
0000100       PROGRAM OHCL
0000200       IMPLICIT COMPLEX*8 (G,V,Z)
0000300       COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
0000400       COMMON/NATOMS/ NATOMS,N,NK
0000500       COMMON/INFO/ AN(40),NOUT
0000600       COMMON/INFO1/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
0000700       INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
0000800       REAL*4 VOIP,VP,Z
0000900       COMMON/VOIP/ VOIP(4,10),AZETA(4,10),CONST,OM1,OM2,VP(4,10),
0001000      *  AZP(4,10)
0001100       COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0001200       COMMON /H/H(40,40)
0001300       COMMON /NC/NC(18),LC(10),MC(10)
0001400       COMMON /NAN/NAN(40)
0001500       DIMENSION S1(9,9)
0001600       DIMENSION RM(130,9)
0001700       DIMENSION ZW4(4),ZW5(5),IP4(4),IP5(5),GWSS(4,4),GWZZ4(4,4),
0001800      *  GWZZ5(5,5),GOSSM1(4,4),GDSS(4,4),
0001900      *  VZSD5(5,4),VZSD4(4,4),VZSD1(1,4),
0002000      *  G8BB(16,16),GZZZ5(5,5),GZZZ4(4,4),GZZZM5(5,5),
0002100      *  GZZZM4(4,4),ROT(9,9),ROT1(4,4),SD(9,9),HD(9,9)
0002200       DATA NAN/3*1/
0002300       DATA AZHS,AZCLS,AZCLP,AZOS,AZOP,AZSIS,AZSIP
0002400      *    /1.,2.3561,2.0387,2.2458,2.2266,1.6344,1.4284/
0002500       DATA EHS,ECLS,ECLP,EOS,EOP,ESIS,ESIP
0002600      *    /13.6,25.27,13.70,32.33,15.80,14.83,7.75/
0002700       DATA AU/0.52916/,POS/9*0./,ROT/81*0./
0002800       DATA ROT1/0.5,0.,0.,0.8660,0.5,-0.4082,-0.7071,-0.2887,
0002900      *  0.5,-0.4082,0.7071,-0.2887,0.5,0.8165,0.,-0.2887/
0002910     5 CONTINUE
0003000       READ(5,100) MODE,IBS,DHS,DAS,CONST,RMAX
0003100   100 FORMAT(2I1,4F10.0)
0003200       WRITE(6,107) MODE,IBS,DHS,DAS,CONST,RMAX
0003300   107 FORMAT(' MODE,IBS,DHS,DAS,CONST,RMAX=',2I3,4F10.3)
0003310       IF(IBS.EQ.1) GO TO 120
0003320       REWIND 26  ;GO TO 121
0003330   120 REWIND 22
0003340   121 CONTINUE
0003400       NATOMS=3
0003500 C------SET-UP SD AND HD------
0003600       CALL COEFFT
0003700 C------INITIALIZE VOIP AND AZETA-------
0003800       POS(3,1)=DHS/AU ;POS(3,2)=DAS/AU
0003900       AN(1)=1 ;AN(2)=8 ;AN(3)=14
0004000       VOIP(1,1)=EHS ;AZETA(1,1)=AZHS
0004100       VOIP(1,2)=EOS ;AZETA(1,2)=AZOS
0004200       VOIP(2,2)=EOP ;AZETA(2,2)=AZOP
0004300       VOIP(1,3)=ESIS ;AZETA(1,3)=AZSIS
0004400       VOIP(2,3)=ESIP ;AZETA(2,3)=AZSIP
0004500       GO TO (11,11,11,12,12,13),MODE
0004600    12 CONTINUE
0004700       AN(2)=17
0004800       VOIP(1,2)=ECLS ;AZETA(1,2)=AZCLS
0004900       VOIP(2,2)=ECLP ;AZETA(2,2)=AZCLP
0005000       GO TO 11
0005100    13 CONTINUE
0005200       AN(2)=9
0005300       VOIP(1,2)=40.12 ;AZETA(1,2)=2.5628
```

```
005400         VOIP(2,2)=32.33 ;AZETA(2,2)=2.5500
005500      11 CONTINUE
005600 C-------SET-UP S------
005700         CALL INTGS(RMAX,1)
005800 C------CALCULATION OF H MATRIX------
005900         DO 408 K=1,NATOMS
006000         DO 408 L=K,NATOMS
006100           LLK=LLIM(K)
006200           LLL=LLIM(L)
006300           ULK=ULIM(K)
006400           ULL=ULIM(L)
006500           NORBK=ULK-LLK+1
006600           NORBL=ULL-LLL+1
006700           DO 409 I=1,NORBK
006800           DO 409 J=1,NORBL
006900             LLKP=LLK+I-1
007000             LLLP=LLL+J-1
007100             LCI=LC(I)+1
007200             LCJ=LC(J)+1
007300             IF(LLKP.EQ.LLLP) GO TO 410
007400             H(LLKP,LLLP)=-S(LLKP,LLLP)*(VOIP(LCI,K)+VOIP(LCJ,L))*
007500        *  CONST/2.
007600           GO TO 409
007700    410  H(LLKP,LLKP)=-VOIP(LCI,K)
007800    409  CONTINUE
007900    408  CONTINUE
008000         DO 20 I=1,9
008100         DO 20 J=I,9
008200           S(J,I)=S(I,J)
008300     20  CONTINUE
008400         DO 21 I=1,9
008500         DO 21 J=1,9
008600     21  S1(I,J)=S(I,J)
008700         GO TO (22,22,22,26,26,26),MODE
008710     22  CONTINUE
008800         CALL INTGS(RMAX,2)
009000         DO 24 J=2,9
009100         DO 24 I=2,9
009200     24  H(I,J)=S(I,J)
009300         GO TO 26
009400     26  DO 27 I=1,9
009500         DO 27 J=I,9
009600     27  H(J,I)=H(I,J)
009700 C       CALL RMAT(S,'S     ',40,9,9,9)
009800 C       CALL RMAT(H,'H     ',40,9,9,9)
009900 C------TRANSFORMATION-------
010000         DO 30 I=1,5
010100     30    ROT(I,I)=1.
010200         DO 31 J=1,4
010300         DO 31 I=1,4
010400     31    ROT(I+5,J+5)=ROT1(I,J)
010500 C       CALL RWRITE(ROT,'ROT   ',9,9,9)
010600         DO 32 J=1,9
010700         DO 32 I=1,9
010800           SD(I,J)=0. ;HD(I,J)=0.
010900         DO 32 K=1,9
011000         DO 32 L=1,9
011100           SD(I,J)=SD(I,J)+ROT(L,I)*S1(L,K)*ROT(K,J)
011200     32    HD(I,J)=HD(I,J)+ROT(L,I)*H(L,K)*ROT(K,J)
011300         CALL RWRITE(SD,'SD    ',9,9,9)
011400         CALL RWRITE(HD,'HD    ',9,9,9)
011500 C------S=1--------
011600         DO 40 J=1,9
011700         DO 40 I=1,9
011800     40    SD(I,J)=0.
011900         DO 41 I=1,9
```

```
012000     41    SD(I,I)=1.
012100           WRITE(6,8)
012200      8 FORMAT(' INPUT EI,EF,DE,DELTA,EPSG IN 5F10.0')
012300           EI=-11. ;EF=-0.1;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
012400           READ(5,9) DEI,DEF,DDE,DDELTA,DEPSG
012500      9 FORMAT(5F10.0)
012600           IF(DEI.NE.0.) EI=DEI
012700           IF(DEF.NE.0.) EF=DEF
012800           IF(DDE.NE.0.) DE=DDE
012900           IF(DDELTA.NE.0.) DELTA=DDELTA
013000           IF(DEPSG.NE.0.) EPSG=DEPSG
013100           NN=(EF-EI+0.01)/DE ;NN=NN+1
013200           IF(IBS.EQ.2) GO TO 103
013300           REWIND 22
013400           ISKIP=(EI-(-11.)+0.01)/DE
013500           IF(ISKIP.EQ.0) GO TO 103
013600           DO 104 I=1,ISKIP
013700           CALL TREAD(GBBB)
013800    104 CONTINUE
013900    103 CONTINUE
014000           READ(5,105) IOUT
014100    105 FORMAT(I3)
014200           E=EI-DE ;NUME=0
014300 C------CHANGE ENERGE------
014400      1 NUME=NUME+1
014500           E=E+DE
014600           IF(E.GT.EF+0.1*DE) GO TO 3
014700           ZE=CMPLX(E,DELTA)
014800 C------READ-IN GDSS------
014900           GO TO (51,52),IBS
015000     51 CALL TREAD(GBBB)
015100           DO 56 J=1,4
015200           DO 56 I=1,4
015300     56    GDSS(I,J)=GBBB(I,J)
015400           GO TO 57
015500     52 READ(26) ((GDSS(I,J),I=1,4),J=1,4)
015600     57 CONTINUE
015700           DO 58 J=1,4
015800           DO 58 I=1,4
015900     58    GDSSM1(I,J)=GDSS(I,J)
016000           CALL CINV(GDSSM1,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
016100           GO TO (60,70,80,70,80,70),MODE
016200 C-------HYDROGEN-------
016300    60 CONTINUE
016400           DO 61 I=1,4
016500    61    VZSD1(1,I)=HD(1,I+5)-ZE*SD(1,I+5)
016600           GZZZ1=1./(ZE-HD(1,1))
016700           IF(IOUT.EQ.1.AND.NUME.EQ.1)
016800          * WRITE(6,106) (VZSD1(1,I),I=1,4),GZZZ1
016900   106 FORMAT(' VZSD1,GZZZ1=',10F8.3)
017000           DO 62 J=1,4
017100           DO 62 I=1,4
017200    62    GWSS(I,J)=GDSSM1(I,J)-VZSD1(1,I)*GZZZ1*VZSD1(1,J)
017300           CALL CINV(GWSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
017400           GZZZM1=1./GZZZ1
017500           GWZZ1=GZZZM1
017600           DO 63 J=1,4
017700           DO 63 I=1,4
017800    63    GWZZ1=GWZZ1-VZSD1(1,I)*GDSS(I,J)*VZSD1(1,J)
017900           GWZZ1=1./GWZZ1
018000           WRITE(6,64) E,GWZZ1,(GWSS(I,I),I=1,4)
018100    64 FORMAT(1H ,11F8.3/1H ,10F8.3)
018200           RM(NUME,1)=AIMAG(GWZZ1)
018300           DO 67 I=1,4
018400    67 RM(NUME,I+1)=AIMAG(GWSS(I,I))
018500           GO TO 1
```

```
018600 C------MATRIX 4-------
018700     70 CONTINUE
018710 C       CALL RWRITE(HD,'HD    ',9,9,9)
018720 C       CALL RWRITE(SD,'SD    ',9,9,9)
018800        DO 71 J=1,4
018900        DO 71 I=1,4
019000     71   VZSD4(I,J)=HD(I+1,J+5)-ZE*SD(I+1,J+5)
019100        DO 72 J=1,4
019200        DO 72 I=1,4
019300     72   GZZZ4(I,J)=ZE*SD(I+1,J+1)-HD(I+1,J+1)
019400        IF(IOUT.EQ.1.AND.NUME.EQ.1)
019500       * CALL CWRITE(VZSD4,'VZSD4',4,4,4)
019600        IF(IOUT.EQ.1.AND.NUME.EQ.1)
019700       * CALL CWRITE(GZZZ4,'GZZZ4M1',4,4,4)
019800        CALL CINV(GZZZ4,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
019900        DO 73 J=1,4
020000        DO 73 I=1,4
020100          GWSS(I,J)=GDSSM1(I,J)
020200        DO 73 K=1,4
020300        DO 73 L=1,4
020400     73   GWSS(I,J)=GWSS(I,J)-VZSD4(L,I)*GZZZ4(L,K)*VZSD4(K,J)
020500        CALL CINV(GWSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
020600        DO 74 J=1,4
020700        DO 74 I=1,4
020800     74   GZZZM4(I,J)=GZZZ4(I,J)
020900        CALL CINV(GZZZM4,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
021000        DO 75 J=1,4
021100        DO 75 I=1,4
021200          GWZZ4(I,J)=GZZZM4(I,J)
021300        DO 75 K=1,4
021400        DO 75 L=1,4
021500     75   GWZZ4(I,J)=GWZZ4(I,J)-VZSD4(I,L)*GDSS(L,K)*VZSD4(J,K)
021600        CALL CINV(GWZZ4,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
021700        WRITE(6,76) E,(GWZZ4(I,I),I=1,4),(GWSS(I,I),I=1,4)
021800     76 FORMAT(1H ,11F8.3/1H ,10F8.3)
021900        DO 77 I=1,4
022000        RM(NUME,I)=AIMAG(GWZZ4(I,I))
022100     77 RM(NUME,I+4)=AIMAG(GWSS(I,I))
022200        GO TO 1
022300 C------MATRIX 5--------
022400     80 CONTINUE
022500        DO 81 J=1,4
022600        DO 81 I=1,5
022700     81   VZSD5(I,J)=HD(I,J+5)-ZE*SD(I,J+5)
022800        DO 82 J=1,5
022900        DO 82 I=1,5
023000     82   GZZZ5(I,J)=ZE*SD(I,J)-HD(I,J)
023100        IF(IOUT.EQ.1.AND.NUME.EQ.1)
023200       * CALL CWRITE(VZSD5,'VZSD5',5,4,4)
023300        IF(IOUT.EQ.1.AND.NUME.EQ.1)
023400       * CALL CWRITE(GZZZ5,'GZZZ5M1',5,5,5)
023500        CALL CINV(GZZZ5,5,0,5,5,1.D-14,ZDET,ZW5,IP5,NSTOP)
023600        DO 83 J=1,4
023700        DO 83 I=1,4
023800          GWSS(I,J)=GDSSM1(I,J)
023900        DO 83 K=1,5
024000        DO 83 L=1,5
024100     83   GWSS(I,J)=GWSS(I,J)-VZSD5(L,I)*GZZZ5(L,K)*VZSD5(K,J)
024200        CALL CINV(GWSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
024300        DO 84 J=1,5
024400        DO 84 I=1,5
024500     84   GZZZM5(I,J)=GZZZ5(I,J)
024600        CALL CINV(GZZZM5,5,0,5,5,1.D-14,ZDET,ZW5,IP5,NSTOP)
024700        DO 85 J=1,5
024800        DO 85 I=1,5
024900          GWZZ5(I,J)=GZZZM5(I,J)
```

```
025000         DO 85 K=1,4
025100         DO 85 L=1,4
025200      85   GWZZ5(I,J)=GWZZ5(I,J)-VZSD5(I,L)*GDSS(L,K)*VZSD5(J,K)
025300         CALL CINV(GWZZ5,5,0,5,5,1.D-14,ZDET,ZW5,IP5,NSTOP)
025400         WRITE(6,86) E,(GWZZ5(I,I),I=1,5),(GWSS(I,I),I=1,4)
025500      86 FORMAT(1H ,11F8.3/1H ,10F8.3)
025600         DO 87 I=1,5
025700      87 RM(NUME,I)=AIMAG(GWZZ5(I,I))
025800         DO 88 I=1,4
025900      88 RM(NUME,I+5)=AIMAG(GWSS(I,I))
026000         GO TO 1
026100       3 CONTINUE
026200 C------OUTPUT------
026300         WRITE(30) ((RM(NUME,I),I=1,9),NUME=1,NN)
026400         GO TO 5
026500         END
026600         SUBROUTINE INTGS(RMAX,IP)
026700         COMMON/ARRAYS/S(40,40),Y(9,5,203),Z(17,45)
026800         COMMON/NATOMS/ NATOMS,N,NK
026900         COMMON/INFO/ AN(40),NOUT
027000         COMMON/INFO1/CZ(40),U(80),ULIM(40),LLIM(40),ULK,ULL,ANL,ANK
027100         COMMON/GAB/T(9,9),PAIRS(9,9),TEMP(9,9),C1(3),C2(3)
027200         COMMON/AUXINT/A(17),B(17)
027300         COMMON/VOIP/ VOIP(4,10),AZETA(4,10),CONST,OM1,OM2,VP(4,10),
027400        *  AZP(4,10)
027500         COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
027600         COMMON /NC/NC(18),LC(10),MC(10)
027700         COMMON /NAN/NAN(40)
027800         DIMENSION P(80,80)
027900         DIMENSION E(3),Q(40)
028000         EQUIVALENCE (P(1),Y(1))
028100         INTEGER AN,ULIM,ULK,ULL,CZ,U,ANL,ANK
028200 C       DEBUG,SUBCHK
028300 C------DETERMINATION OF SIZE OF AO BASIS IN AND CORE CHARGE CZ
028400 C       CALL RWRITE(POS,'POS  ',3,10,10)
028500 C       CALL RWRITE(VOIP,'VOIP ',4,10,10)
028600 C       CALL RWRITE(AZETA,'AZETA',4,10,10)
028700         RMAX=99.
028710         CALL MAKEDN(1,3)
028800         N=0
028900         DO 60 I=1,NATOMS
029000         LLIM(I) = N+1
029100         K=1
029200         IF (AN(I).LT.11) GO TO 20
029300      10 N=N+4
029400         CZ(I)=AN(I)-10
029500         GO TO 50
029600      20 IF (AN(I).LT.3) GO TO 40
029700      30 N=N+4
029800         CZ(I) = AN(I)-2
029900         GO TO 50
030000      40 N=N+1
030100         CZ(I)= AN(I)
030200      50 CONTINUE
030300         ULIM(I) = N
030400      60 CONTINUE
030500 C------ASSIGNMENT OF ANGULAR MOMENTUM QUANTUM NOS. TO ATOMIC ORBITALS
030600         LC(1)=0
030700         LC(2)=1
030800         LC(3)=1
030900         LC(4)=1
031000         LC(5)=2
031100         LC(6)=2
031200         LC(7)=2
031300         LC(8)=2
031400         LC(9)=2
```

```
00031500         IC(1)=0
00031600         IC(2)=1
00031700         IC(3)=-1
00031800         IC(4)=0
00031900         IC(5)=0
00032000         IC(6)=1
00032100         IC(7)=-1
00032200         IC(8)=2
00032300         IC(9)=-2
00032400 C-----FILL U ARRAY---U(J) IDENTIFIES THE ATOM TO WHICH ORBITAL J IS
00032500 C-----ATTACHED E.G. ORBITAL 32 ATTACHED TO  ATOM 7, ETC.
00032600 C-----ASSIGNMENT OF ORBITAL EXPONENTS TO ATOMS BY SLATERS RULES
00032700         NC(8)=2 ;NC(14)=3 ;NC(1)=1 ;NC(17)=3 ;NC(9)=2
00032800         DO 92 K=1,NATOMS
00032900         LLK=LLIM(K)
00033000         JLK=ULIM(K)
00033100         ANK=AN(K)
00033200         NORBK=ULK-LLK+1
00033300         DO 92 I=1,NORBK
00033400         LLKP=LLK+I-1
00033500         LCZETA=LC(I)+1
00033600         A(LLKP)=AZETA(LCZETA,K)
00033700         J(LLKP)=K
00033800      92 CONTINUE
00033900 C-----STEP THRU PAIRS OF ATOMS
00034000         KK1=1
00034100         IF(IP.EQ.1) GO TO 93
00034200         KK1=2
00034300      93 CONTINUE
00034400         DO 320 K=KK1,NATOMS
00034500         DO 320 L=K,NATOMS
00034600         DO 100 I=1,3
00034700         C1(I)=POS(I,K)
00034800    100  C2(I)=POS(I,L)
00034900 C-----CALCULATE UNIT VECTOR ALONG INTERATOM AXIS,E
00035000         CALL RELVEC(R,E,C1,C2)
00035100         IF(R.GT.RMAX) GO TO 320
00035200         LLK = LLIM(K)
00035300         LLL = LLIM(L)
00035400         JLK = ULIM(K)
00035500         JLL = ULIM(L)
00035600         NORBK=ULK-LLK+1
00035700         NORBL=ULL-LLL+1
00035800         ANK=AN(K)
00035900         ANL=AN(L)
00036000 C-----LOOP THRU PAIRS OF BASIS FUNCTIONS, ONE ON EACH ATOM
00036100         DO 200 I=1,NORBK
00036200         DO 200 J=1,NORBL
00036300         IF(K.EQ.L) GO TO 160
00036400    110  IF(MC(I).NE.MC(J)) GO TO 150
00036500    120  IF(MC(I).LT.0) GO TO 140
00036600         LLKP=LLK+I-1 ;LLLP=LLL+J-1
00036700    130  PAIRS(I,J)= SQRT((Q(LLKP)*R)**(2*NC(ANK)+1)*(Q(LLLP)*R)**(2*NC
00036800        1)+1)/(FACT(2*NC(ANK))*FACT(2*NC(ANL))))*(-1.D0)**(LC(J)+MC(J))
00036900        2*SS(NC(ANK),LC(I),MC(I),NC(ANL),LC(J),Q(LLKP)*R,Q(LLLP)*R)
00037000         IF(IP.EQ.1) GO TO 190
00037100         PAIRS(I,J)=HH(ANK,LC(I),MC(I),ANL,LC(J),R,NAN(K),NAN(L))
00037200         GO TO 190
00037300    140  PAIRS(I,J)=PAIRS(I-1,J-1)
00037400         GO TO 190
00037500    150  PAIRS(I,J)=0.0D0
00037600         GO TO 190
00037700    160  IF (I.EQ.J) GO TO 170
00037800    180  PAIRS(I,J)=0.0D0
00037900         GO TO 190
00038000    170  PAIRS(I,J)=1.0D0
```

```
038100          IF(IP.EQ.1) GO TO 190
038200          IF(MC(I).LT.0) GO TO 171
038300          PAIRS(I,J)=HH(ANK,LC(I),MC(I),ANL,LC(J),R,NAN(K),NAN(L))
038400          GO TO 190
038500      171 PAIRS(I,J)=PAIRS(I-1,J-1)
038600      190 CONTINUE
038700      200 CONTINUE
038800          LCULK=LC(NORBK)
038900          LCULL=LC(NORBL)
039000          MAXL=MAX0(LCULK,LCULL)
039100          IF(R.GT.0.00001D0) GO TO 220
039200      210 GO TO 250
039300  C-----ROTATE INTEGRALS FROM DIATOMIC BASIS TO MOLECULAR BASIS
039400      220 CALL HARMTR(T,MAXL,E)
039500          DO 230 I=1,NORBK
039600          DO 230 J=1,NORBL
039700          TEMP(I,J) = 0.D0
039800          DO 230 KK=1,NORBL
039900          TEMP(I,J) = TEMP(I,J)+T(J,KK)*PAIRS(I,KK)
040000      230 CONTINUE
040100          DO 240 I=1,NORBK
040200          DO 240 J=1,NORBL
040300          PAIRS(I,J) = 0.D0
040400          DO 240 KK=1,NORBK
040500          PAIRS(I,J) = PAIRS(I,J)+T(I,KK)*TEMP(KK,J)
040600      240 CONTINUE
040700  C-----FILL S MATRIX
040800      250 CONTINUE
040900          DO 260 I=1,NORBK
041000          LLKP=LLK+I-1
041100          DO 260 J=1,NORBL
041200          LLLP=LLL+J-1
041300      260 S(LLKP,LLLP)=PAIRS(I,J)
041400      320 CONTINUE
041500      330 CONTINUE
041600          RETURN
041700          END

R OHCL        OF  A1402.I.FORT  LIST END
```


A1-103

ER DOHCL3 OF A1402.I.FORT

```
0000100 12 3.4          3.0        1.75       20.
0000200 -5.5          -3.6
0000300   0
0000400 12 3.3          3.0        1.75      20.
0000500 -5.5          -3.6
0000600   1
0000700 12 3.2          2.          1.75        20.
0000800 -5.5          -3.6
0000900   1
0001000 12 3.1          4.         1.75        20.
0001100 -5.5          -3.6
0001200   1
0001300 12 4.           3.2         1.75       20.
0001400 -5.5          -3.6
0001500   0
```

ER DOHCL3 OF A1402.I.FORT LIST END

ER SIGEEO OF A1402.I.FORT

```
0000100         PROGRAM SIGEEO
0000200         IMPLICIT COMPLEX*8 (G,V,Z)
0000300         COMMON/ARRAYS/S(40,40),Y(9,5,203),AZ(17,45)
0000400         COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0000500         COMMON /H/H(40,40)
0000600         DIMENSION GYYY(8,8),GSTRSS(4,4),VSY(4,8),GDSS(4,4),ZW4(4),
0000700        *          IP4(4),GDSSM1(4,4)
0000800         DATA AU/.529167/,PAI/3.141592/,BL/1.61/
0000900         WRITE(6,8)
0001000       8 FORMAT(' INPUT EI,EF,DE,DELTA,EPSG IN 5F10.0')
0001100         EI=-5.5 ;EF=-3.6 ;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
0001200         READ(5,9) DEI,DEF,DDE,DDELTA,DEPSG
0001300       9 FORMAT(5F10.0)
0001400         IF(DEI.NE.0.) EI=DEI
0001500         IF(DEF.NE.0.) EF=DEF
0001600         IF(DDE.NE.0.) DE=DDE
0001700         IF(DDELTA.NE.0.) DELTA=DDELTA
0001800         IF(DEPSG.NE.0.) EPSG=DEPSG
0001900         EPSG=EPSG
0002000         NN=(EF-EI+0.01)/DE ;NN=NN+1
0002100 C------SET UP S AND H-------
0002200         CALL SHBAB(IAN5)
0002300 C       DO 6 J=1,4
0002400 C       DO 6 I=1,4
0002500 C       H(I,J+20)=0.
0002600 C     6 H(I+20,J)=0.
0002700 C       CALL RWRITE(H,'H     ',40,40,10)
0002800 C------MAKE VSY-----
0002900         DO 420 J=1,4
0003000         DO 420 I=1,4
0003100           VSY(I,J)=0.
0003200   420     VSY(I,J+4)=H(I+20,J+16)
0003300 C       CALL CWRITE(VSY,'VSY   ',4,8,8)
0003400 C------SCANNING OF ENERGY-----
0003500         E=EI-DE ;NUME=0
0003600 C------CHANGE ENERGY------
0003700    40   NUME=NUME+1
0003800         E=E+DE
0003900         IF(E.GT.EF+0.1*DE) GO TO 2
0004000         DELTA=DELTA
0004100 C       ZE=E+(0.,1.)*DELTA
0004200 C------READ-IN GEEO(GYYY)------
0004300         READ(23) ((GYYY(I,J),J=I,8),I=1,8)
0004400         DO 410 I=1,7
0004500         I1=I+1
0004600         DO 410 J=I1,8
0004700   410   GYYY(J,I)=GYYY(I,J)
0004800 C       CALL CWRITE(GYYY,'GYYY  ',8,8,8)
0004900 C------READ-IN GDSS-----
0005000         READ(26) ((GDSS(I,J),I=1,4),J=1,4)
0005100 C       CALL CWRITE(GDSS,'GDSS  ',4,4,4)
0005200 C------CALCULATE GSTRSS,GDSSM1------
0005300         DO 431 J=1,4
0005400         DO 431 I=1,4
0005500   431   GDSSM1(I,J)=GDSS(I,J)
0005600         CALL CINV(GDSSM1,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0005700 C       CALL CWRITE(GDSSM1,'GDSSM1',4,4,4)
0005800         DO 440 J=1,4
```

```
005900          DO 440 I=1,4
006000            GSTRSS(I,J)=GDSSM1(I,J)
006100          DO 440 K=1,8
006200          DO 440 L=1,8
006300      440   GSTRSS(I,J)=GSTRSS(I,J)-VSY(I,L)*GYYY(L,K)*VSY(J,K)
006400  C       CALL CWRITE(GSTRSS,'GSTRSS',4,4,4)
006500          CALL CINV(GSTRSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
006600          WRITE(6,441) E,(GSTRSS(I,I),I=1,4)
006700      441 FORMAT(1H ,9F10.3)
006800          GO TO 40
006900        2 STOP ;END

ER  SIGEEO      OF  A1402.I.FORT  LIST END
```

ER DSIG JF A1402.I.FORT

```
0000100 -5.5      -3.6       0.1       0.04      0.05
0000200 14.95     7.78      29.6      12.7
0000300 1.6344    1.4284    2.2460     2.227
0000400 .87       1.69      2.5
0000500 2.5       4.3       6.0
0000600 1.34      2.1       2.1       2.9
0000700  8 0.       .6
0000800
0000900
0001000
0001100 180.      180.
```

ER DSIG JF A1402.I.FORT LIST END

ER WEAK OF A1402.I.FORT

```
0000100       PROGRAM WEAK
0000200       IMPLICIT COMPLEX*8 (G,V,Z)
0000300       COMMON/ARRAYS/S(40,40),Y(9,5,203),AZ(17,45)
0000400       COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
0000500       COMMON /H/H(40,40)
0000600       DIMENSION GDSS(4,4),GDSSM1(4,4),VSO(4,4),ZW4(4),IP4(4),
0000700      *          GWSS(4,4),GOOO(4,4)
0000800       DIMENSION ROT(4,4)
0000900       DIMENSION HD(4,4)
0001000       DATA AU/.529167/,PAI/3.141592/,BL/1.61/
0001100       WRITE(6,8)
0001200     8 FORMAT(' INPUT EI,EF,DE,DELTA,EPSG IN 5F10.0')
0001300       EI=-5.5 ;EF=-3.6 ;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
0001400       READ(5,9) DEI,DEF,DDE,DDELTA,DEPSG
0001500     9 FORMAT(5F10.0)
0001600       IF(DEI.NE.0.) EI=DEI
0001700       IF(DEF.NE.0.) EF=DEF
0001800       IF(DDE.NE.0.) DE=DDE
0001900       IF(DDELTA.NE.0.) DELTA=DDELTA
0002000       IF(DEPSG.NE.0.) EPSG=DEPSG
0002100       EPSG=EPSG
0002200       NN=(EF-EI+0.01)/DE ;NN=NN+1
0002300 C------SET UP S AND H-------
0002400       CALL SHBAB(IAN5)
0002500 C------MAKE VSO--------
0002600       READ(5,451) RX,RY,RZ
0002700   451 FORMAT(3F10.0)
0002800       CALL ROTAT(RX,RY,RZ,ROT)
0002900       DO 452 I=1,4
0003000       DO 452 J=1,4
0003100   452 HD(I,J)=H(I+20,J+16)
0003200       CALL RWRITE(HD,'HD    ',4,4,4)
0003300 C     CALL RWRITE(ROT,'ROT   ',4,4,4)
0003400       DO 454 I=1,4
0003500       DO 454 J=1,4
0003600         VSO(I,J)=0.
0003700       DO 454 K=1,4
0003800   454   VSO(I,J)=VSO(I,J)+H(I+20,K+16)*ROT(K,J)
0003900       CALL CWRITE(VSO,'VSO   ',4,4,4)
0004000 C------SCANNING OF ENERGY-----
0004100       E=EI-DE ;NUME=0
0004200 C------CHANGE ENERGY------
0004300    40 NUME=NUME+1
0004400       E=E+DE
0004500       IF(E.GT.EF+0.1*DE) GO TO 2
0004600       DELTA=DELTA
0004700 C------READ-IN GDSS-----
0004800       READ(26) ((GDSS(I,J),I=1,4),J=1,4)
0004900 C------SET GOOO AND GDSSM1----
0005000       DO 431 J=1,4
0005100       DO 431 I=1,4
0005200   431 GDSSM1(I,J)=GDSS(I,J)
0005300       CALL CINV(GDSSM1,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0005400       READ(28) ((GOOO(I,J),J=1,4),I=1,4)
0005500 C------CALCULATE GWSS------
0005600       DO 440 J=1,4
0005700       DO 440 I=1,4
0005800       GWSS(I,J)=GDSSM1(I,J)
```

```
0005900         DO 440 K=1,4
0006000         DO 440 L=1,4
0006100     440 GWSS(I,J)=GWSS(I,J)-VSO(I,L)*GOOO(L,K)*VSO(J,K)
0006200         CALL CINV(GWSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
0006300         WRITE(6,441) E,(GWSS(I,I),I=1,4)
0006400     441 FORMAT(1H ,9F10.3)
0006500         GO TO 40
0006600       2 STOP ;END
0006700         SUBROUTINE ROTAT(RX,RY,RZ,ROT)
0006800         DIMENSION T(4,4),ROTD(4,4),ROT(4,4)
0006900 C------MAKE TX*TY*TZ WHICH IS TO BE MULTIPLIED FROM RIGHT OF SOMETHIN
0007000         PAI=3.141592
0007100         X=RX*PAI/180. ;Y=RY*PAI/180. ;Z=RZ*PAI/180.
0007200 C------RESET ROT-----
0007300         DO 50 I=1,4
0007400         DO 50 J=1,4
0007500      50 ROT(I,J)=0.
0007600         DO 51 I=1,4
0007700      51 ROT(I,I)=1.
0007800 C-------X AXIS ROTATION--------
0007900         IF(X.EQ.0.) GO TO 20
0008000         DO 11 J=1,4
0008100         DO 11 I=1,4
0008200      11 T(I,J)=0.
0008300         T(1,1)=1. ;T(2,2)=1.
0008400         SX=SIN(X) ;CX=COS(X)
0008500         T(3,3)=CX ;T(3,4)=-SX
0008600         T(4,3)=SX ;T(4,4)=CX
0008700         DO 12 J=1,4
0008800         DO 12 I=1,4
0008900           ROTD(I,J)=0.
0009000         DO 12 K=1,4
0009100      12 ROTD(I,J)=ROTD(I,J)+ROT(I,K)*T(K,J)
0009200         DO 13 I=1,4
0009300         DO 13 J=1,4
0009400      13 ROT(I,J)=ROTD(I,J)
0009500 C------Y AXIS ROTATION--------
0009600      20 IF(Y.EQ.0.) GO TO 30
0009700         DO 21 J=1,4
0009800         DO 21 I=1,4
0009900      21 T(I,J)=0.
0010000         T(1,1)=1. ;T(3,3)=1.
0010100         SY=SIN(Y) ;CY=COS(Y)
0010200         T(2,2)=CY ;T(2,4)=SY
0010300         T(4,2)=-SY ;T(4,4)=CY
0010400         DO 22 J=1,4
0010500         DO 22 I=1,4
0010600         ROTD(I,J)=0.
0010700         DO 22 K=1,4
0010800      22 ROTD(I,J)=ROTD(I,J)+ROT(I,K)*T(K,J)
0010900         DO 23 I=1,4
0011000         DO 23 J=1,4
0011100      23 ROT(I,J)=ROTD(I,J)
0011200 C------Z AXIS ROTATION--------
0011300      30 IF(Z.EQ.0.) GO TO 40
0011400         DO 31 J=1,4
0011500         DO 31 I=1,4
0011600      31 T(I,J)=0.
0011700         T(1,1)=1. ;T(4,4)=1.
0011800         SZ=SIN(Z) ;CZ=COS(Z)
0011900         T(2,2)=CZ ;T(2,3)=-SZ
0012000         T(3,2)=SZ ;T(3,3)=CZ
0012100         DO 32 J=1,4
0012200         DO 32 I=1,4
0012300         ROTD(I,J)=0.
0012400         DO 32 K=1,4
```

```
012500      32 ROTD(I,J)=ROTD(I,J)+ROT(I,K)*T(K,J)
012600         DO 33 I=1,4
012700         DO 33 J=1,4
012800      33 ROT(I,J)=ROTD(I,J)
012900      40 RETURN ;END

ER WEAK       OF  A1402.I.FORT  LIST END
```

ER DWEAK JF A1402.I.FORT

```
0000100 -5.5        -3.6          0.1         0.04        0.05
0000200 14.95       7.78         29.6        12.7
0000300 1.6344      1.4284       2.2460       2.227
0000400 .87         1.69         2.5
0000500 2.5         4.3          6.0
0000600 1.34        2.1          2.1         2.9
0000700  8 0.         .6
0000800
0000900
0001000
0001100 180.         0.0
0001200  0.         30.          0.
```

ER DWEAK JF A1402.I.FORT LIST END

A1-111

ER WEAKS OF A1402.I.FORT

```
00000100       PROGRAM WEAKS
00000200       IMPLICIT COMPLEX*8 (G,V,Z)
00000300       COMMON/ARRAYS/S(40,40),Y(9,5,203),AZ(17,45)
00000400       COMMON /POS/POS(3,10),BL,AU,TH,PAI,R2,R3
00000500       COMMON /H/H(40,40)
00000600       DIMENSION GDSS(4,4),GDSSM1(4,4),VSS(4,4),ZW4(4),IP4(4),
00000700      *          GWSS(4,4),GBSS(4,4)
00000800       DIMENSION TVEC1(3),POS1(3,4),ROT(4,4,4),ROT1(4,4),ROT2(4,4),
00000900      *          T(4,4),HD(4,4)
00001000       DATA AU/.529167/,PAI/3.141592/,BL/1.61/
00001100       WRITE(6,8)
00001200     8 FORMAT(' INPUT EI,EF,DE,DELTA,EPSG IN 5F10.0')
00001300       EI=-5.5 ;EF=-3.6 ;DE=0.1 ;DELTA=0.04 ;EPSG=0.05
00001400       READ(5,9) DEI,DEF,DDE,DDELTA,DEPSG
00001500     9 FORMAT(5F10.0)
00001600       IF(DEI.NE.0.) EI=DEI
00001700       IF(DEF.NE.0.) EF=DEF
00001800       IF(DDE.NE.0.) DE=DDE
00001900       IF(DDELTA.NE.0.) DELTA=DDELTA
00002000       IF(DEPSG.NE.0.) EPSG=DEPSG
00002100       EPSG=EPSG
00002200       NN=(EF-EI+0.01)/DE ;NN=NN+1
00002300 C------SET UP S AND H-------
00002400       CALL SHBAB(IAN5)
00002500 C------MAKE VSS--------
00002600       CALL POSROT(0.,0.,0.,BL,AU,TVEC1,POS1,ROT)
00002700       DO 450 I=1,4
00002800       DO 450 J=1,4
00002900   450 ROT1(I,J)=ROT(I,J,1)
00003000       READ(5,451) RX,RY,RZ
00003100   451 FORMAT(3F10.0)
00003200       CALL POSROT(RX,RY,RZ,BL,AU,TVEC1,POS1,ROT)
00003300       DO 452 I=1,4
00003400       DO 452 J=1,4
00003500   452 ROT2(I,J)=ROT(I,J,1)
00003600       DO 453 I=1,4
00003700       DO 453 J=1,4
00003800         T(I,J)=0.
00003900       DO 453 K=1,4
00004000   453 T(I,J)=T(I,J)+ROT1(K,I)*ROT2(K,J)
00004100       DO 455 I=1,4
00004200       DO 455 J=1,4
00004300   455 HD(I,J)=H(I+20,J)
00004400       DO 454 I=1,4
00004500       DO 454 J=1,4
00004600       VSS(I,J)=0.
00004700       DO 454 K=1,4
00004800   454 VSS(I,J)=VSS(I,J)+HD(I,K)*T(K,J)
00004900       CALL CWRITE(VSS,'VSS   ',4,4,4)
00005000 C------SCANNING OF ENERGY-----
00005100       E=EI-DE ;NUME=0
00005200 C------CHANGE ENERGY------
00005300    40 NUME=NUME+1
00005400       E=E+DE
00005500       IF(E.GT.EF+0.1*DE) GO TO 2
00005600       DELTA=DELTA
00005700 C------READ-IN GDSS-----
00005800       READ(26) ((GDSS(I,J),I=1,4),J=1,4)
```

```
005900 C------SET GBSS AND GDSSM1----
006000       DO 431 J=1,4
006100       DO 431 I=1,4
006200   431 GDSSM1(I,J)=GDSS(I,J)
006300       CALL CINV(GDSSM1,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
006400       READ(29) ((GBSS(I,J),J=1,4),I=1,4)
006500 C------CALCULATE GWSS------
006600       DO 440 J=1,4
006700       DO 440 I=1,4
006800       GWSS(I,J)=GDSSM1(I,J)
006900       DO 440 K=1,4
007000       DO 440 L=1,4
007100   440 GWSS(I,J)=GWSS(I,J)-VSS(I,L)*GBSS(L,K)*VSS(J,K)
007200       CALL CINV(GWSS,4,0,4,4,1.D-14,ZDET,ZW4,IP4,NSTOP)
007300       WRITE(6,441) E,(GWSS(I,I),I=1,4)
007400   441 FORMAT(1H ,9F10.3)
007500       GO TO 40
007600     2 STOP ;END

ER  WEAKS      OF  A1402.I.FORT  LIST END
```

ER DWEAKS JF A1402.I.FORT

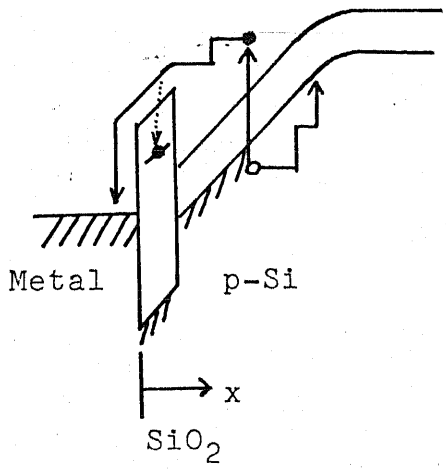
```
000100 -5.5        -3.6          0.1          0.04        0.05
000200 14.95       7.78         29.6         12.7
000300 1.6344      1.4284       2.2460        2.227
000400 .87         1.69         2.5
000500 2.5         4.3          6.0
000600 1.34        2.1          2.1          2.9
000700  0 -.45
000800
000900
001000
001100 180.         0.0
001200 0.          30.          0.
```

ER DWEAKS JF A1402.I.FORT LIST END

APPENDIX B

MEASUREMENTS OF THE SiO_2 BULK TRAPS

The preliminary results are discussed on the measurements of the SiO_2 bulk traps. The avalanche injection technique is reported in detail and the results of the photo-depopulation measurement are mentioned briefly.



B-1 PRINCIPLE

Injection is carried out by applying AC voltage of the frequency about MHz to MOS diode made with p-Si substrate. AC voltage induces avalanche breakdown and the generated electrons are injected into SiO_2, a part of which are trapped in the SiO_2 film. The rate equation which governs the trapping mechanics is expressed as

$$\frac{dn_{Ti}(x,t)}{dt} = \frac{j_{inj}}{q}\sigma_i \{ N_{Ti}(x) - n_{Ti}(x,t) \} \qquad B-1$$

,where $n_{Ti}(x,t)$ is the number of electrons per unit area which are trapped in the i-th kind of the trap at the location x and time t, j_{inj}/q is the number of the electrons which pass through the SiO_2 film, σ_i is the capture cross section of the i-th kind of the trap, and and $N_{Ti}(x)$ is the total number of the i-th kind of the trap. Solving this rate equation under the initial condition of $n_{Ti}(x,0)=0$, we have

$$n_{Ti}(x,t) = N_{Ti}(x)\cdot\left(1 - e^{-\sigma_i \frac{j_{inj}}{q}}\right) \qquad B-2$$

On the other hand, letting Cox, tox the capacitance and the thickness of the SiO film respectively, the flat band voltage is written as

$$\Delta V_{FB}(t) = \frac{q}{C_{ox} t_{ox}} \int_0^{t_{ox}} x \sum_i n_{Ti}(x,t)\, dx \qquad B-3$$

Substitution of eq. B-2 into eq. B-3 yields

$$\eta_{eff} = \frac{d\,(C_{ox}\Delta V_{FB}(t))}{dt} j_{inj}^{-1} = \sum_i (N_{tti})_{eff}\,\sigma_i\, e^{-\sigma_i N_{inj}} \qquad B-4$$

The following notation are used in obtaining eq. B-4.

$$\begin{cases} \dfrac{j_{inj}}{q}\, t \longrightarrow N_{inj} \\ \dfrac{\overline{\kappa_i}}{t_{ox}} \displaystyle\int_0^{t_{ox}} N_{Ti}(x)\,dx \longrightarrow (N_{tti})_{eff} \end{cases}$$

This eq. B-4 is the fundamental equation. The physical parameters (Ntti)eff and σ_i are obtained by semilog-plotting ηeff versus Ninj.

Here, the results of trial calculation in which two kinds of traps are assumed to exist in the SiO_2 film are shown. Figure B-1 is the calculated curve of eq. B-3, fig. B-2 is the semilog plot of fig. B-1, and fig. B-3 is the calculated curve of eq. B-4.

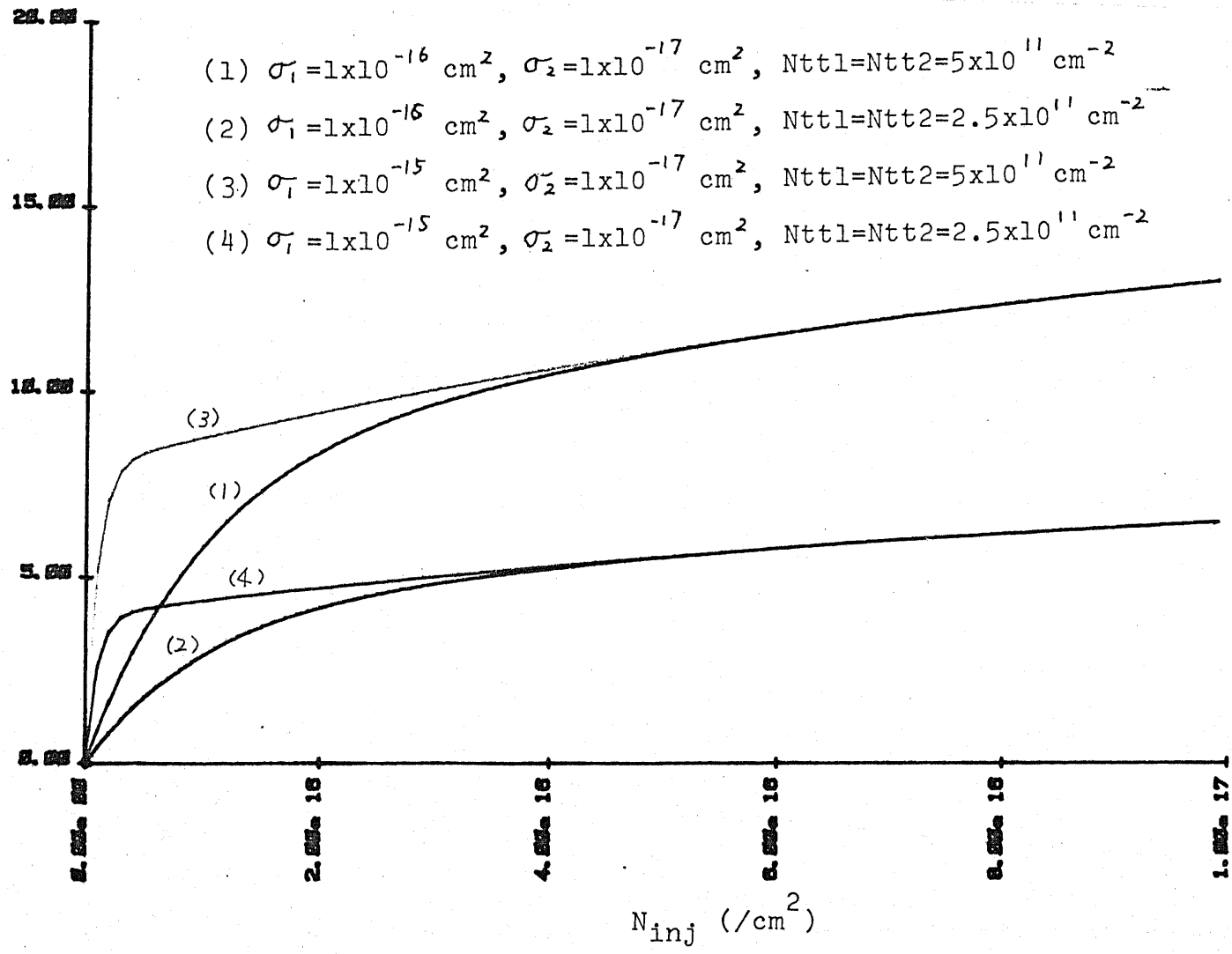


Fig. B-1 Calculated curve of $\overline{V}_{inj}$ versus N_{inj}

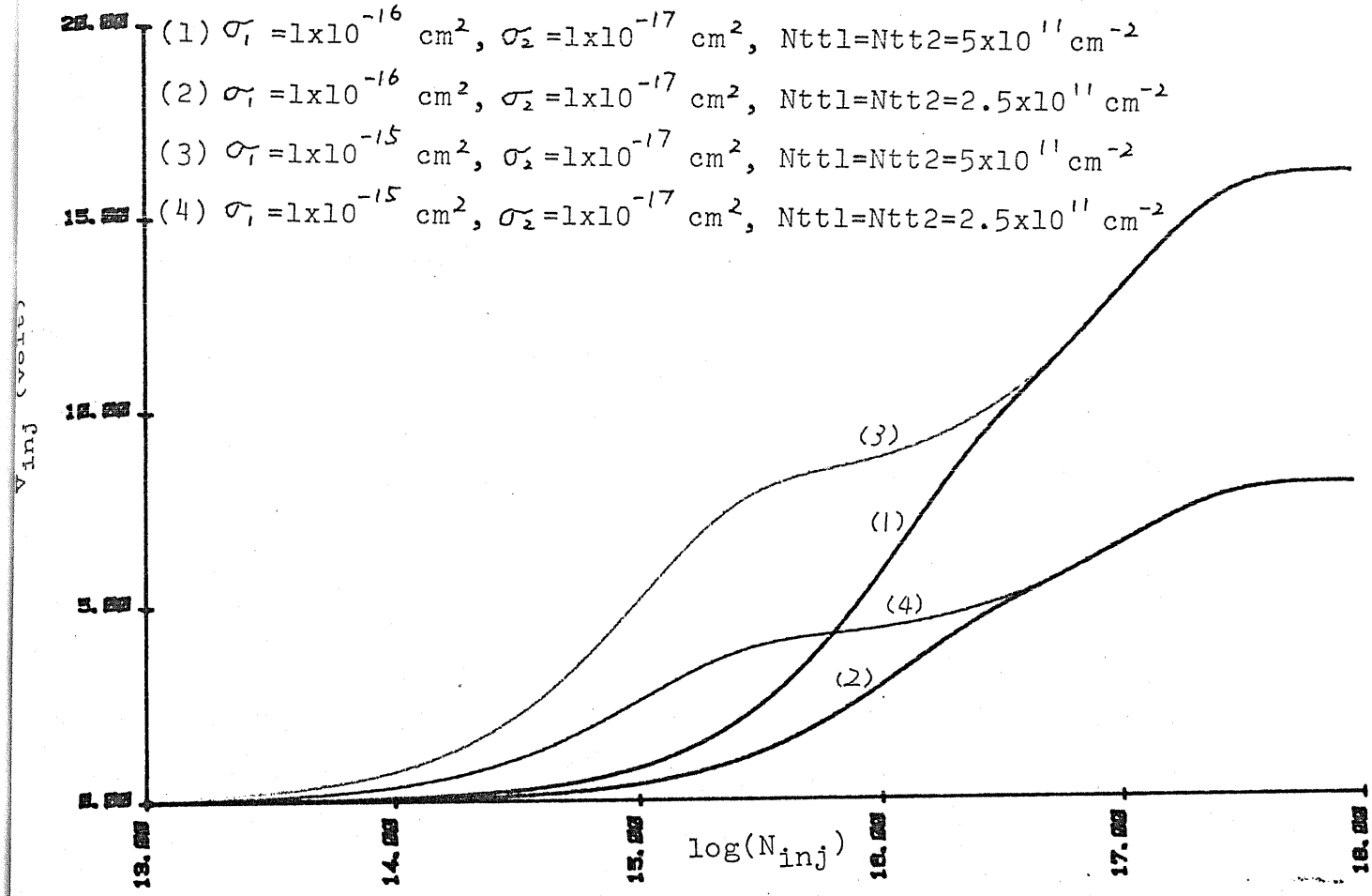


Fig. B-2 Calculated curve of $\bar{V}_{inj}$ versus log(N_{inj})

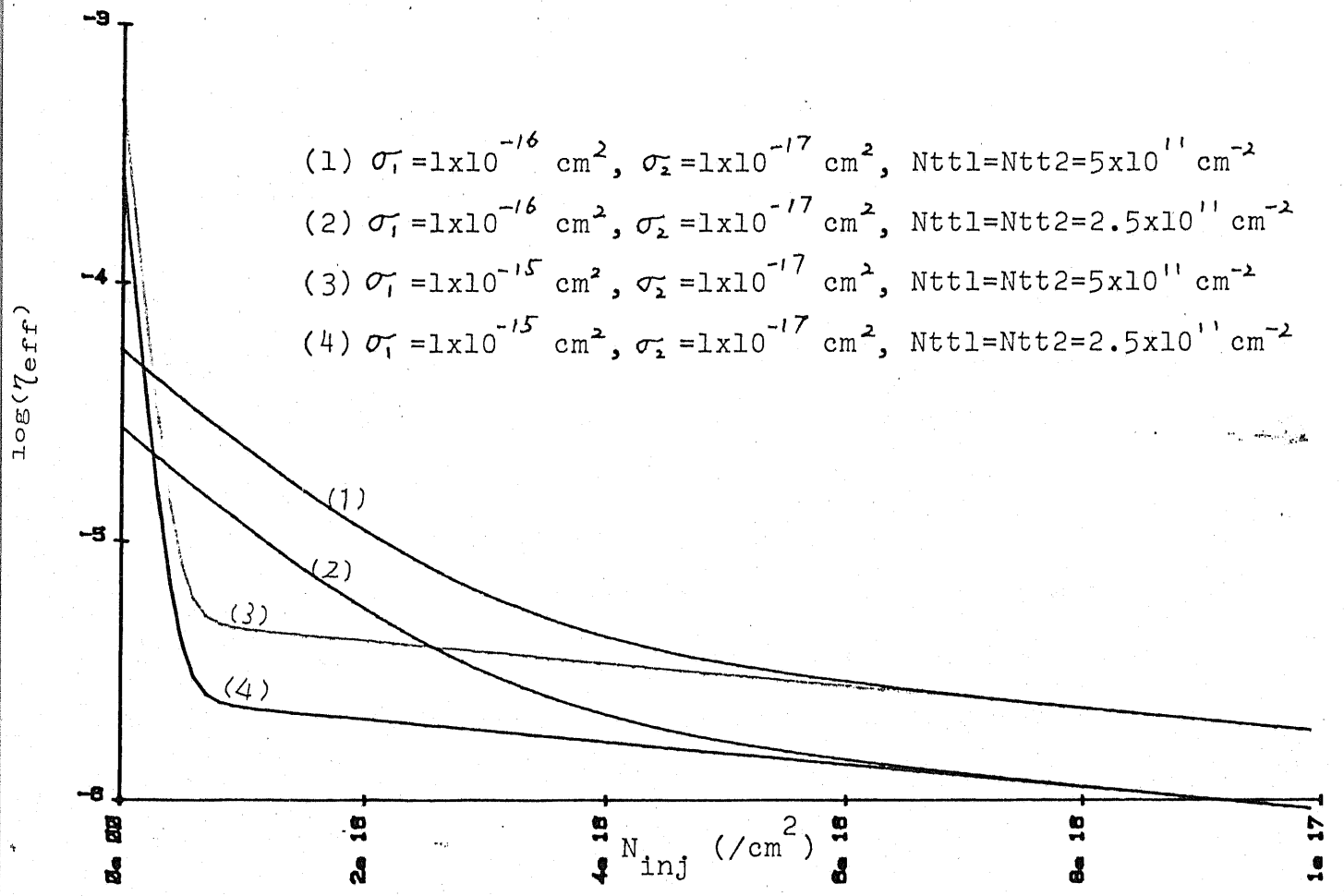


Fig. B-3 Calculated curve of log(η_{eff}) versus N_{inj}

B-2 SETUP FOR MEASUREMENT

The main electronic circuit for the measurement is shown in fig. B-4. It gives the beedback voltage $\overline{V}inj$ to the MOS sample so as to keep the DC component of the avalanche current constant. Since the feedback voltage $\overline{V}inj$ equals to the flat band voltage V_{FB} as shown in Fig. B-5, V_{FB} can be monitored if $\overline{V}inj$ is monitored. In order to monitor and process the data of $\overline{V}inj$, a micro-computer system (see fig. B-6) is constructed because the measurements are carried out for a long time, say from 10 min. to 2 hours, and the amount of the data is enormous. Figure B-7 is the interface circuit employed for connecting the micro-computer with the digital voltmeter and fig. B-8 is the program for the data aquisition.

B-3. EXPERIMENTAL RESULTS

The sample used here is 0.5 [ohm.cm] p-Si, which is oxidized in dry O_2 at 1000 [°C] and annealed in N_2 for 30 min. The thickness of the oxide is 1000 [Å] and the area of the sample is 0.36[mm^2]. The frequency used for the avalanche injection is about 500 [kHz].

The change in the C-V curve as the injection proceeds is given in fig. B-9. It is seen from this C-V curve and from the DLTS experiment (see fig. B-10) that the density of the interface states increases as the amount of injected electrons

increases. So that the correction of the effect of the increased interface states must be considered if the exact analysis is required. But as a preliminary experiment this effect is not taken into account in the following. The reproductivity of this measurement is good if the sample is made with the same process as is seen in fig. B-11. Fugures 12 and 13 show the measured $\overline{V}inj(t)$ and the measured $\log(\eta eff)$ with the theoretical fit where two kinds of traps are assumed to exist in the SiO_2 film. The fit is not satisfactory and one of the reason of this poor fit is that there exist more kinds of traps in the SiO_2. The correction of the increased interface states and the consideration of the more kinds of traps are to be done as the next step.

The avalanche current versus the avalanche voltage is plotted in fig. B-14.

B-4. PHOTO-DEPOPULATION MEASUREMENT

Photo-depopulation measurements are carried out by using 1kW Xe lamp and mercury lamp. The sample is first charged up by the avalanche injection technique mentioned above to the flat band voltage shift of about 5 volts and then exposed to the light which includes the photon of the energy up to 6 eV. The depopulation is done under the zero bias condition. The flat band voltage is measured to know how many electrons are

excited to the SiO_2 conduction band and go away. No substantial difference is observed between the sample with and without H anneal. That is, when the flat band voltage of the sample is shifted to 5 eV, then the flat band voltage shift after the bleaching process by the Hg lamp is about 0.1 eV. Since this shift is very small it is difficult to pick up the difference between the sample with and without H anneal. The fact that about 98% of the trapped electrons can not be depopulated by photon although there exist substantial amount of shallow levels (ref. 41) suggests that the photo-depopulation technique, in the present, is not so powerful in determining the characteristics of the traps in the SiO_2 film from a technological viewpoint.

The charging is carried out in the dark and all photo-depopulation experiment is done in the liquid nitrogen at the temperature of 77 [K]. The whole setup is shown in photo B-1.

B-5. CONCLUSION

The avalanche injection measurement ia carried out and is thought to be a powerful tool in characterizing the electrically active traps in the SiO_2 film, whereas the photo-depopulation experiment is not so useful because about 98% of the electrically active traps in the oxide are inaccessible by photon.

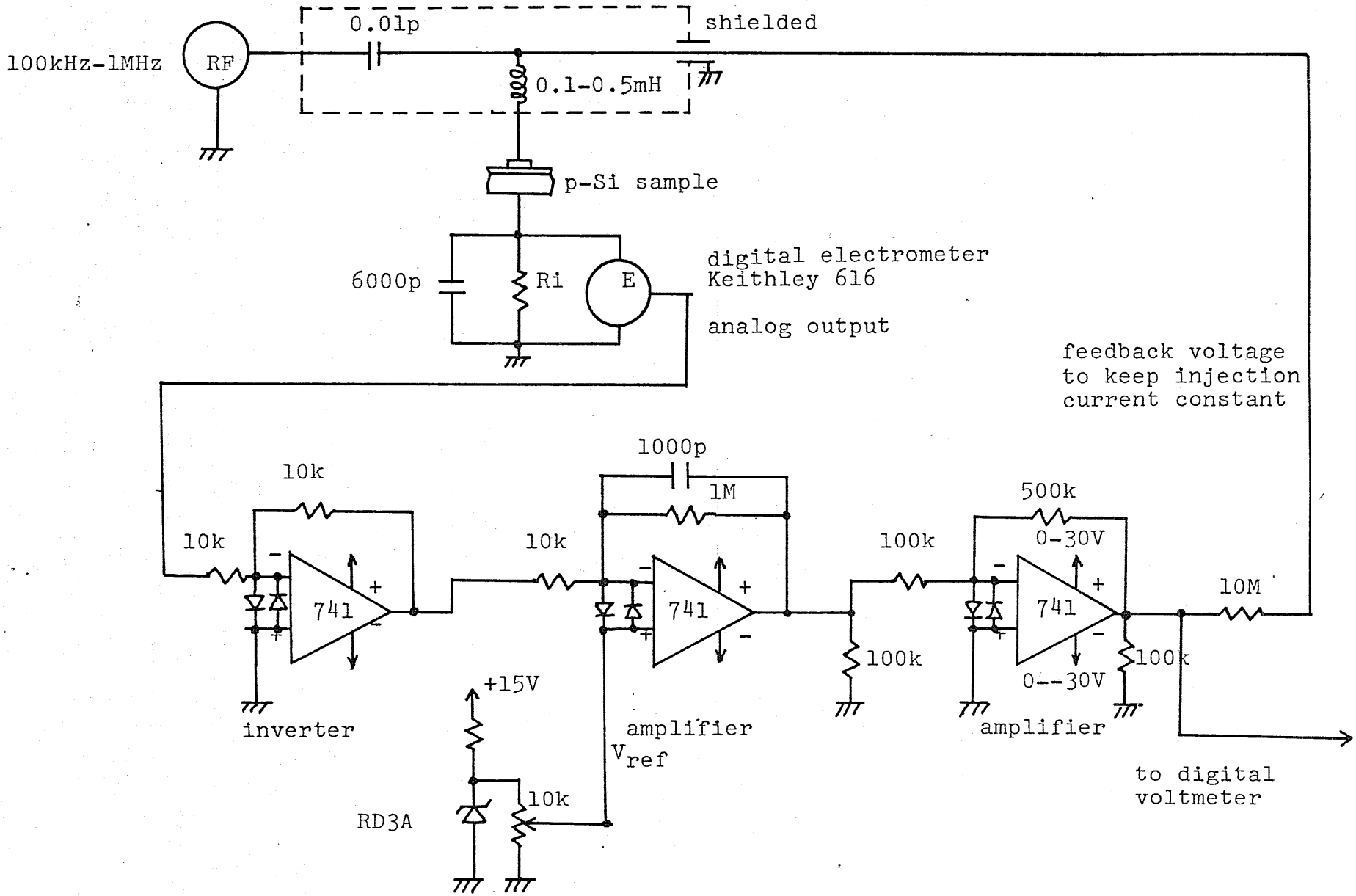


Fig. B-4 Main electronic circuit for the measurement.

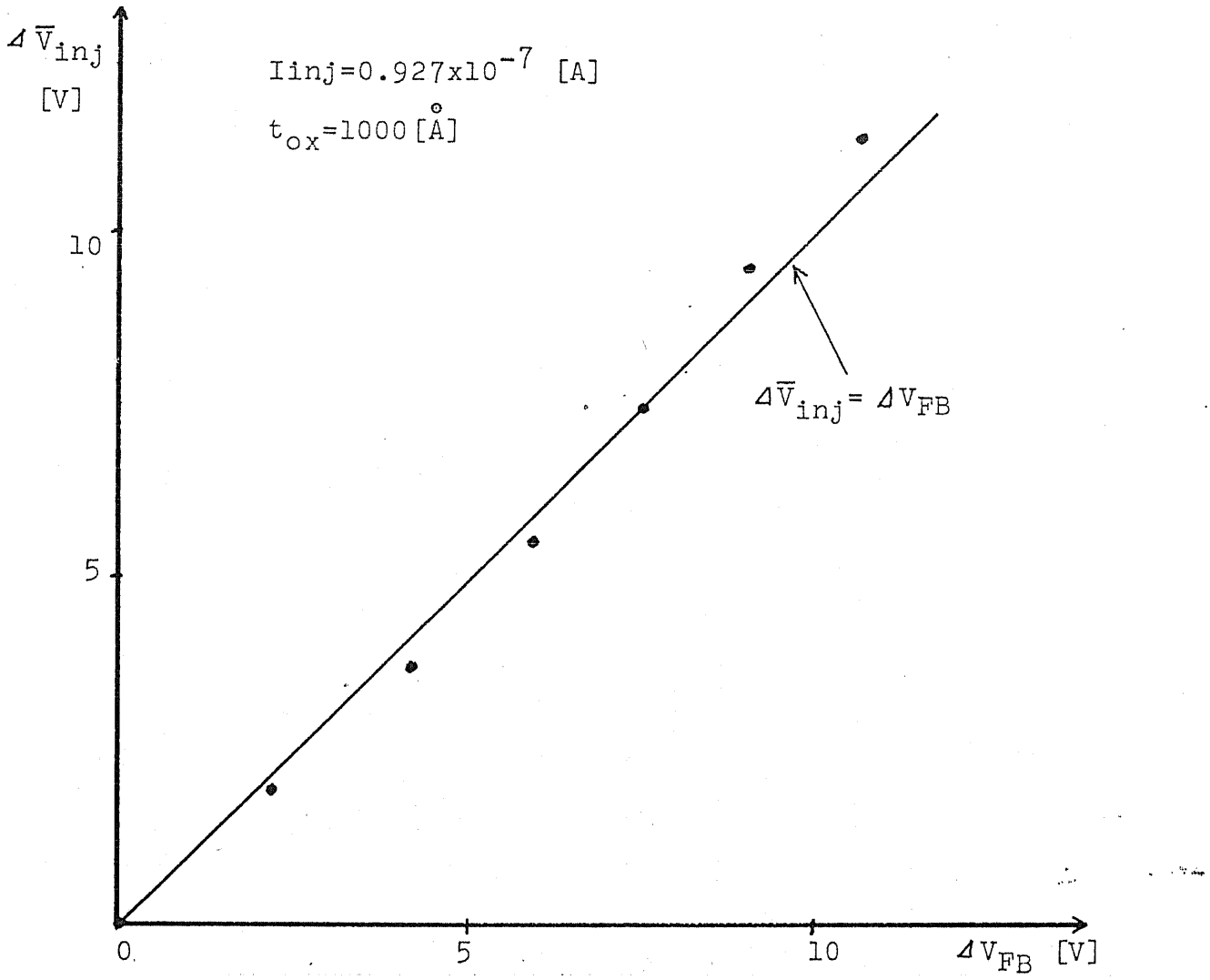


Fig. B-5 Relation between V_{FB} and $\overline{V}_{inj}$.

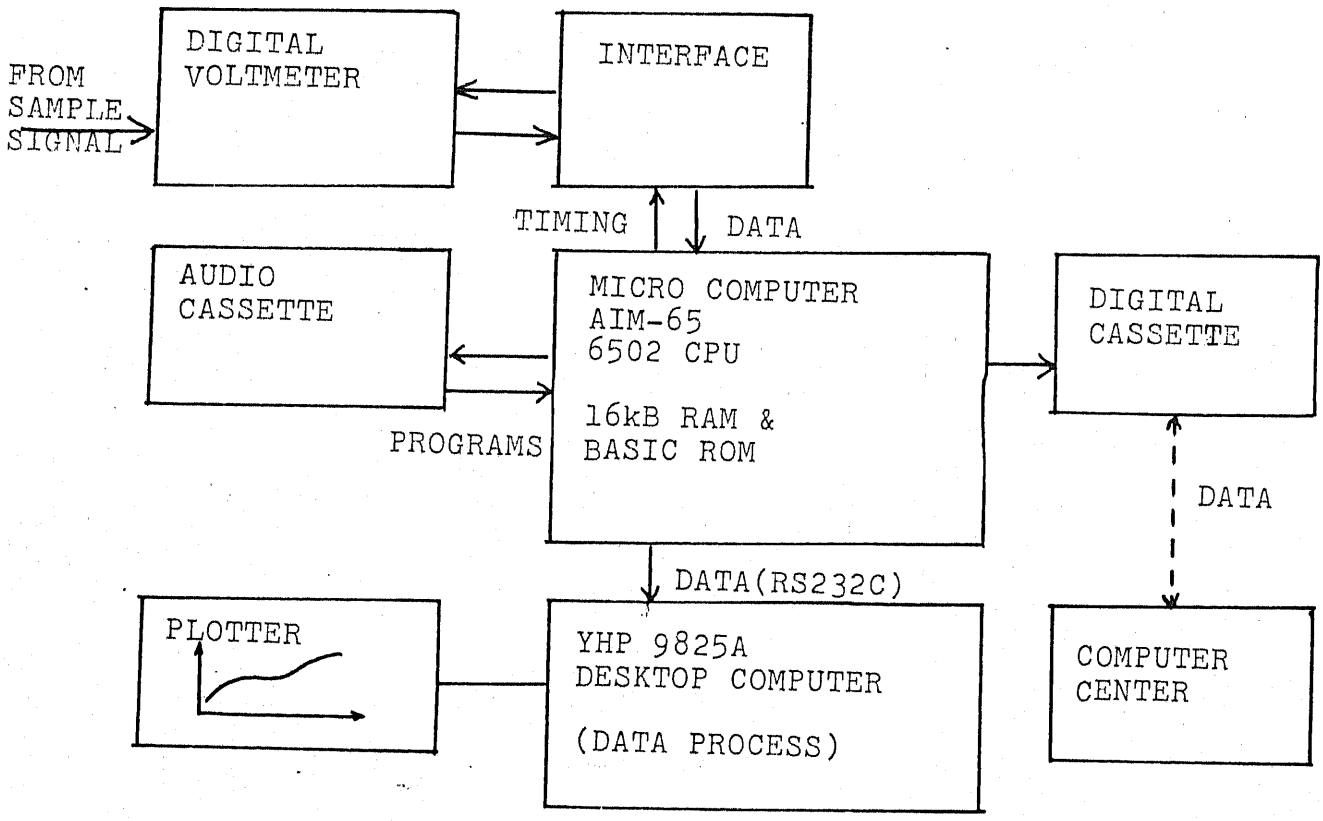


Fig. B-6 Micro-computer system for the measurement.

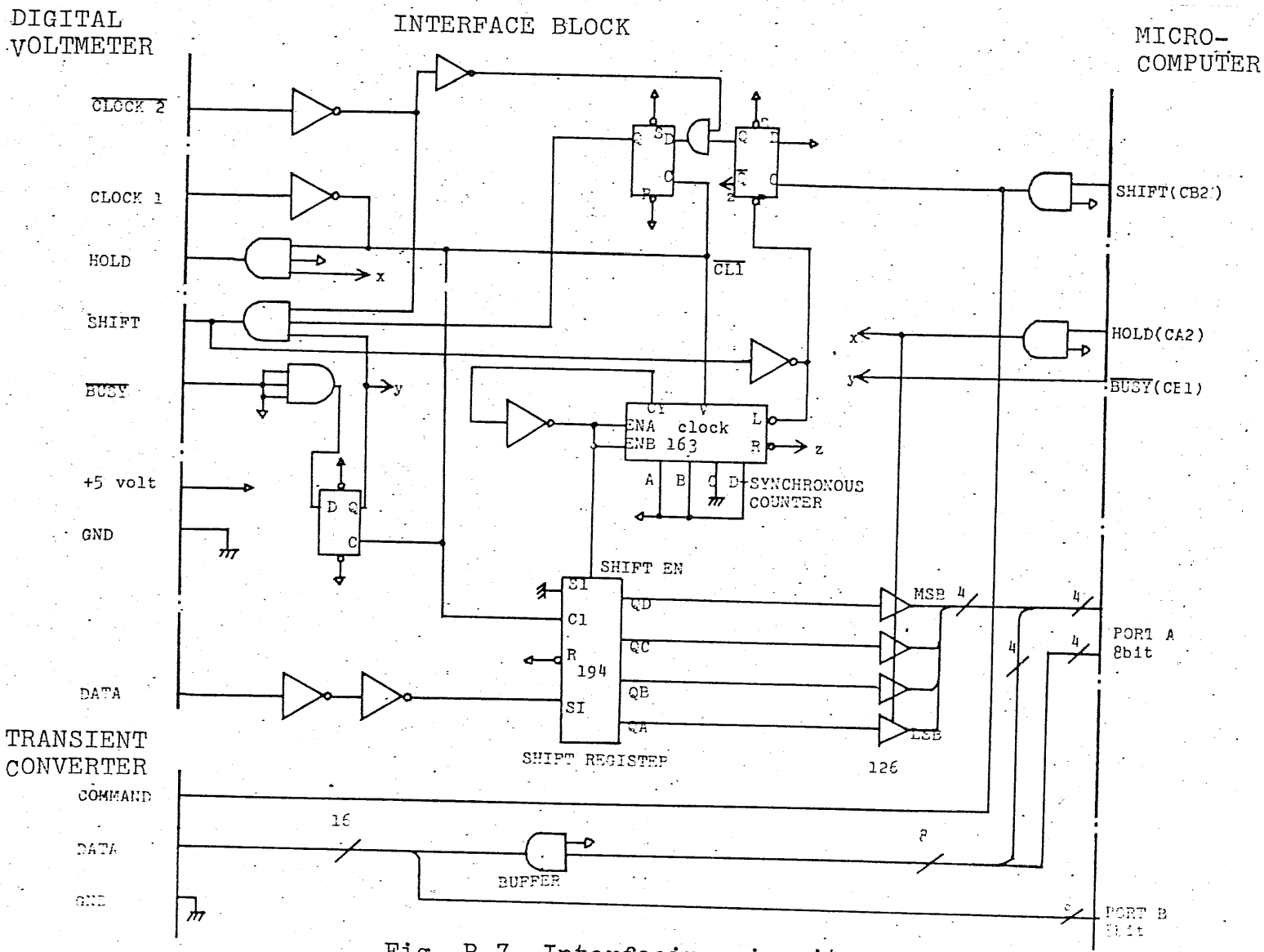


Fig. B-7 Interfacing circuit

Fig. B-8

PROGRAM FOR ELECTRON INJECTION MEASUREMENT
(DATA AQUISITION) written in BASIC

```
 10 REM===DATA FROM
DVM===
 20 INPUT!"INTVL (SE
C)";IT
 25 INPUT!"ADRS IN H
EX(EVEN)";PC$:GOSUB4
000:FR%=PA00
 30 FR%=PA%:PRINT! P
A%:GOSUB3000
 40 INPUT!"#DATA (<4
000)";DT%
 50 REM===PORT A INP
UT===
 60 POKE 40963,0
 70 REM===HOLD SET(C
A2) & SFT RESET(CB2)
===
 80 AD=40972:A$="CE"
:GOSUB2000
 90 FOR D=1TODT%*2ST
EP2
 95 D%=INT(D+.1)
 100 REM===WAIT===
 105 JT=IT*995-404
 110 FORI=1TOJT:NEXT
 120 REM===HOLD RESE
T(CA2)===
 130 AD=40972:A$="CC
":GOSUB2000
 140 REM===WAIT UNTI
L BUSY IS OFF (CB1)=
==
 145 FOR H=1TO80:NEX
TH
 150 PK=PEEK(40960):
AD=40972:A$="DC":GOS
UB2000
 160 WAIT 40973,16
 170 REM===HOLD SET(
CA2)===
 180 AD=40972:A$="CE
":GOSUB2000
 190 X%=USR(0)
 200 P1%=PEEK(16371)
:P2%=PEEK(16372)
 205 P3%=PEEK(16373)
:P4%=PEEK(16374)
 210 POKE FR%+D%+1,I
NT(P1% AND 240)+INT(
P2%/16)
 220 POKE FR%+D%,INT
(P3% AND 240)+INT(P4
%/16)
 230 NEXTD
 240 REM===HOLD RESE
T(CA2)===
 250 POKE 40972,204
 255 PRINT!"STOP STO
P"
 260 STOP:END

 1000 REM===DATA TO
TC===
 1010 INPUT!"#DATA (
<4000)";DT%
 1020 FR%=8192
 1030 REM===PORT AB
OUTPUT===
 1040 AD=40962:A$="F
F":GOSUB2000
 1050 AD=40963:GOSUB
2000
 1060 REM===AB LATCH
 CLEAR===
 1070 AD=40971:A$="0
3":GOSUB2000
 1080 REM===HOLD RES
ET(CA2) & CB2 LOW===
 1090 AD=40972:A$="C
C":GOSUB2000
 1100 FOR D=1TODT%*2
STEP2
 1105 D%=INT(D+.1)
 1110 REM===WAIT===
 1130 REM===OUTPUT D
ATA===
 1140 POKE 40961,PEE
K(FR%+D%-1)
 1150 POKE 40960,PEE
K(FR%+D%)
 1160 REM===8 PULSES
 ON CB2===
 1170 FOR I=1 TO 8
 1180 POKE 40972,236
 1200 POKE 40972,204
 1220 NEXT I
 1230 NEXT D
 1240 STOP:END
 1999 REM===POKE SUB
ROUTINE===
 2000 A1%=ASC(LEFT$(
A$,1)):A2%=ASC(RIGHT
$(A$,1))
 2010 IF A1%<58 THEN
 A1%=A1%-48:GOTO2030
 2020 IF A1%>64 THEN
 A1%=A1%-55
 2030 IF A2%<58 THEN
 A2%=A2%-48:GOTO2050
 2040 IF A2%>64 THEN
 A2%=A2%-55
 2050 AA=INT(A1%*16+
A2%+.1)
 2060 POKE INT(AD+.1
),AA:RETURN

 3000 REM===MAKE MAC
HINE SUBROUTINE===
 3010 TX$="8EF03F8DF
13FA200A9EE"
 3012 TX$=TX$+"8D0CA
0EAA9CE8D0CA0EAEA"
 3014 TX$=TX$+"AD01A
09DF23FE8E009F0034C0
81F"
 3016 TX$=TX$+"AEF03
FADF13F4CD1C0"
 3020 FOR I=1TOLEN(T
X$)STEP2
 3030 AD=7936+INT(I/
2):A$=MID$(TX$,INT(I
+.1),2)
 3040 GOSUB2000
 3050 NEXTI
 3060 POKE4,0:POKE5,
31
 3070 RETURN
 4000 REM===HEX TO D
EC===
 4010 PC$=RIGHT$("00
0"+PC$,4):PA%=0
 4020 FOR I4=1TO4
 4030 A1%=ASC(MID$(P
C$,I4,1))
 4040 IF A1%<58 THEN
 A1%=A1%-48:GOTO4060
 4050 IF A1%>64 THEN
 A1%=A1%-55
 4060 I6%=16^(4-I4)
 4065 PA%=PA%+A1%*I6
%
 4070 NEXT I4
 4080 RETURN
```

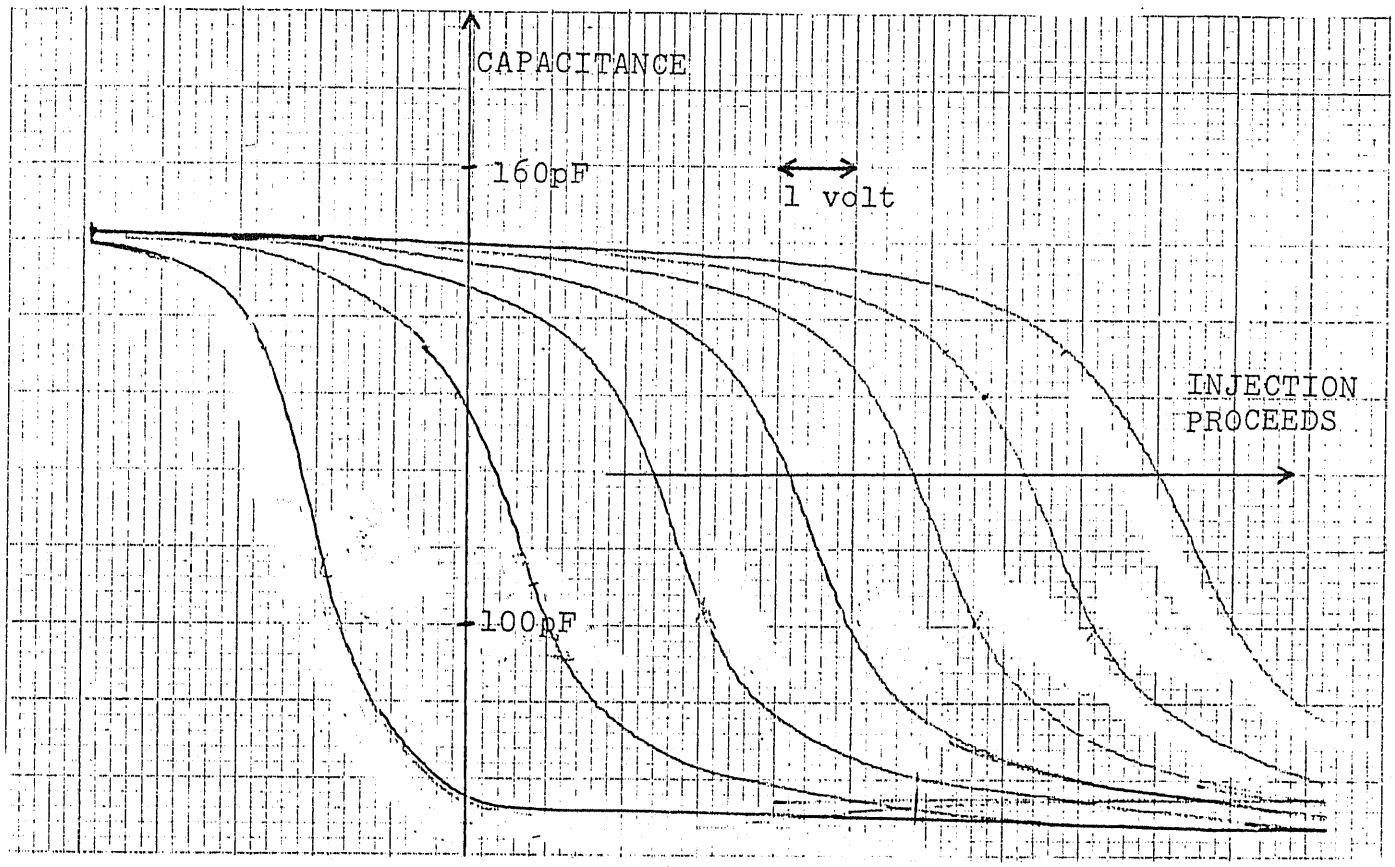


Fig. B-9 Change in C-V curve as injection proceeds.

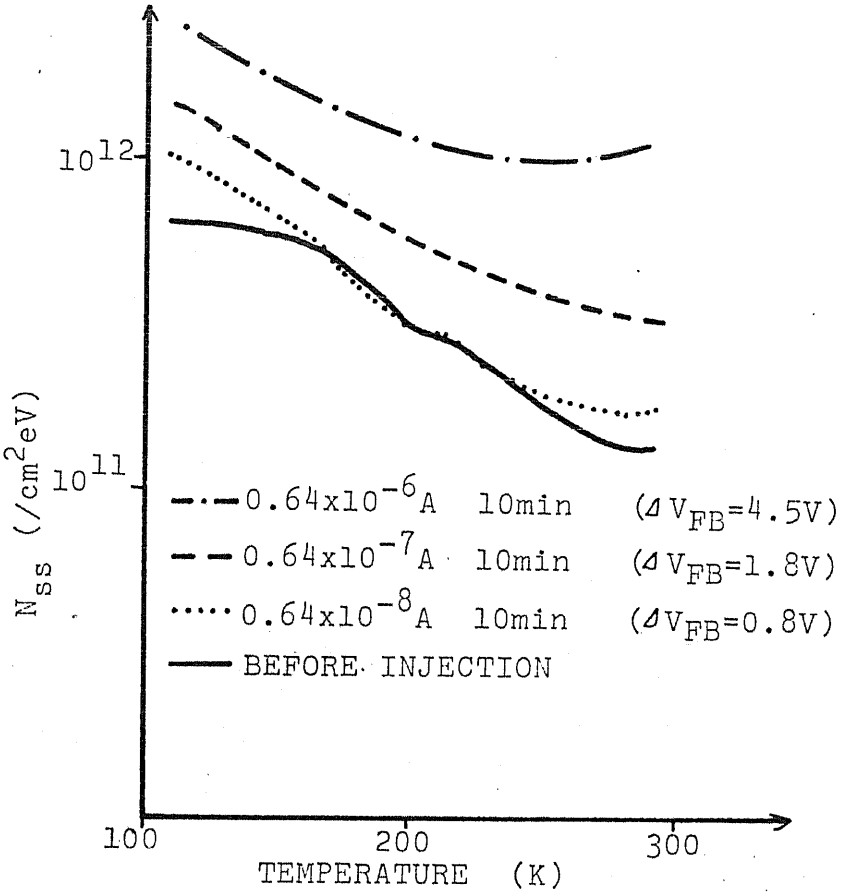


Fig. B-10 DLTS measurement on interface states for various number of injected electrons

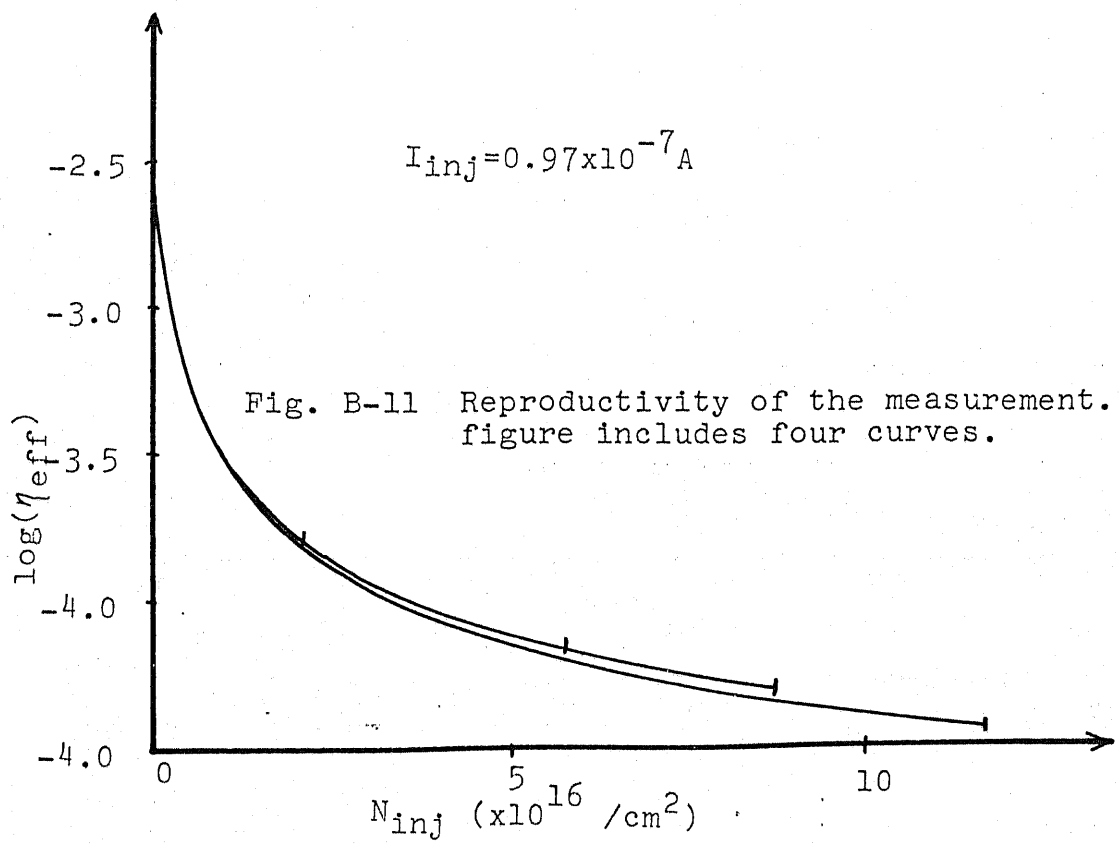


Fig. B-11 Reproductivity of the measurement. This figure includes four curves.

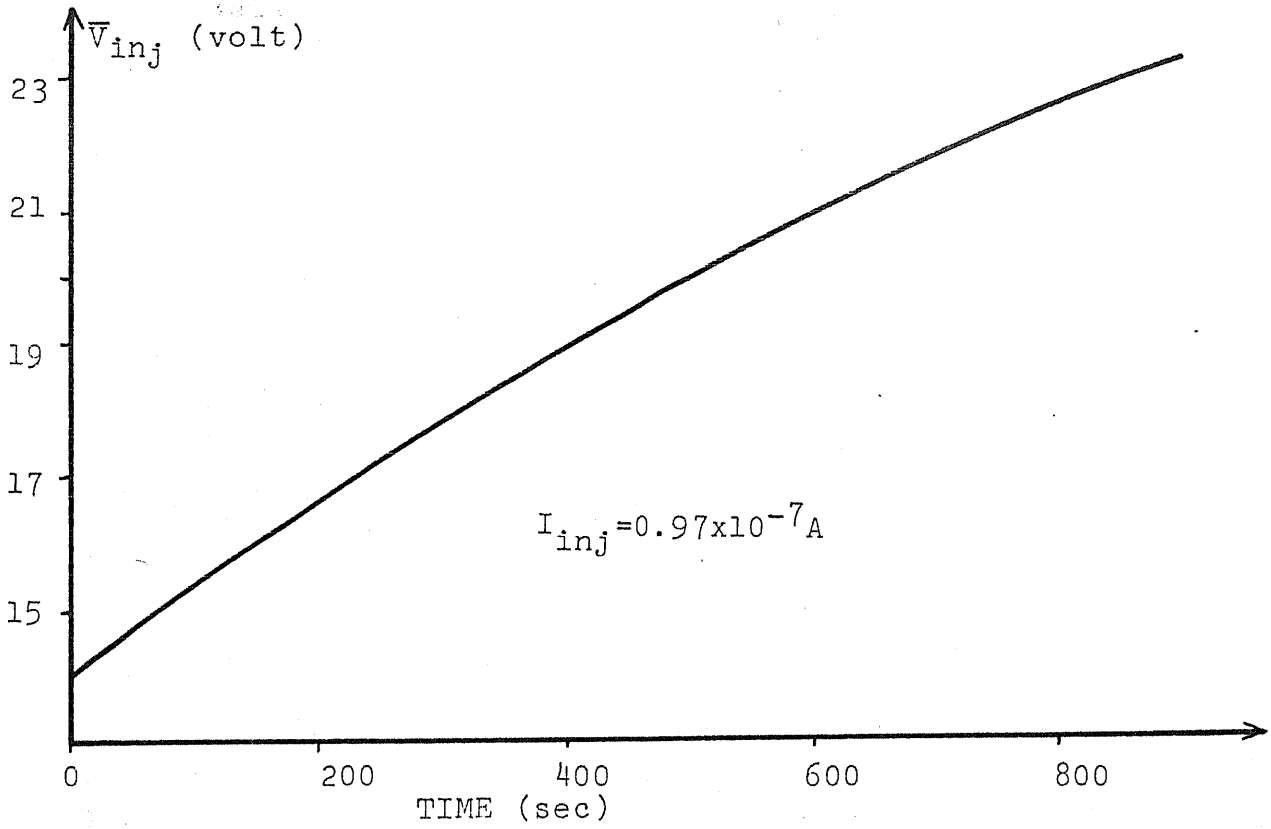


Fig. B-12 Measured $\overline{V}_{inj}$.

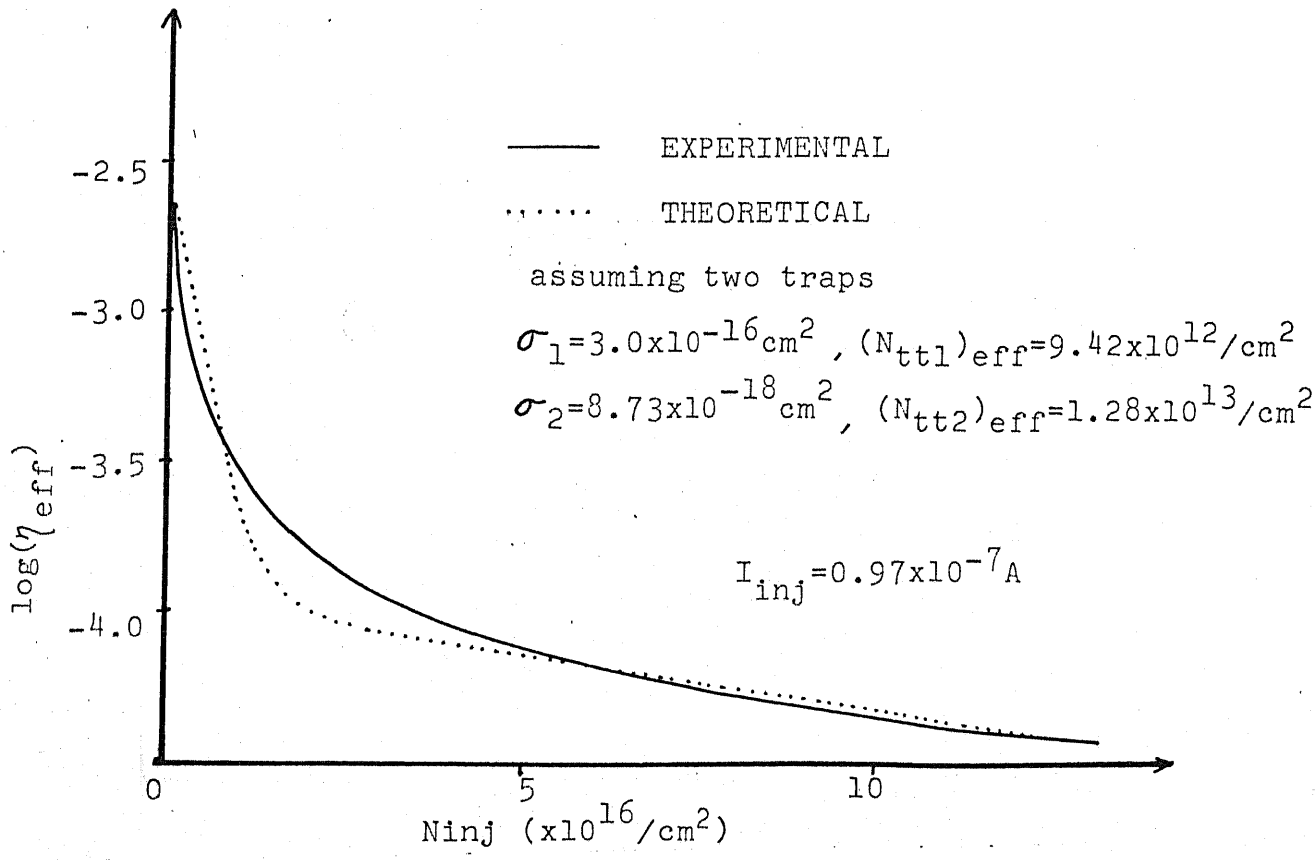


Fig. B-13 Measured log(ηeff) with the theoretical fit where two kinds of traps are assumed to exist in SiO_2 film.

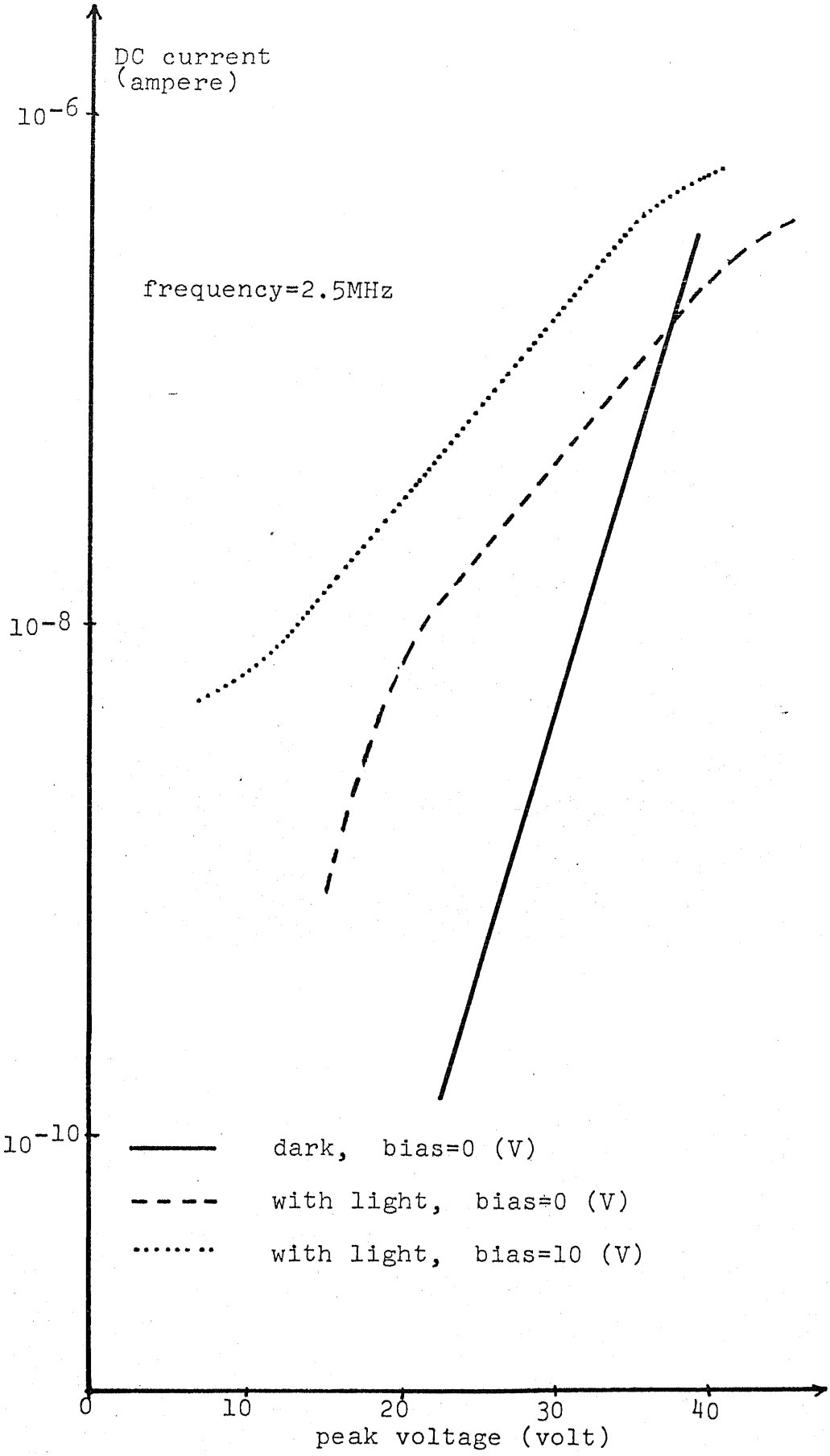


Fig. B-14 Avalanche current versus peak avalanche voltage

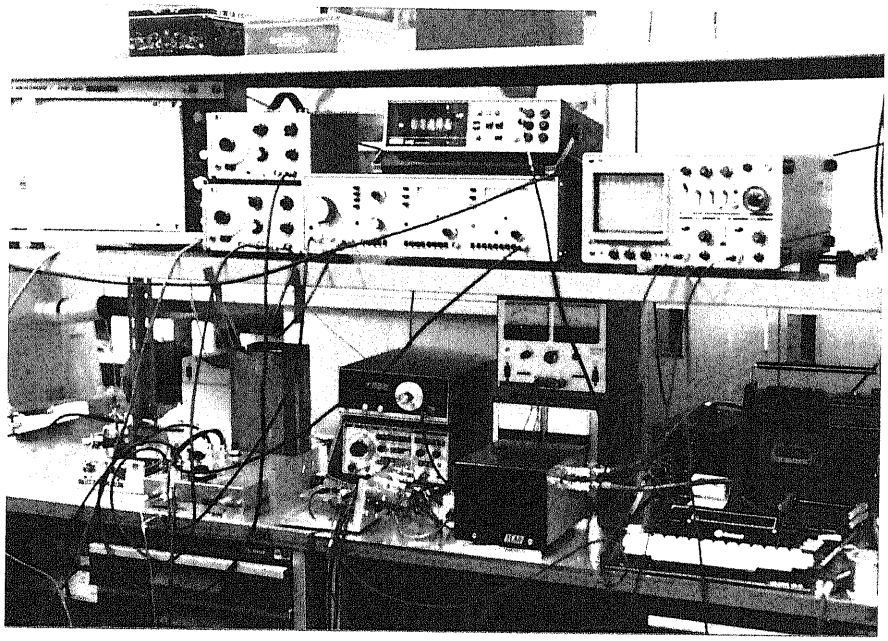


Photo. B-1. Setup for avalanche injection and photodepopulation measurements.

APPENDIX C USEFUL TABLES AND FIGURES

Some of the useful tables and figures used in this thesis are listed only for completion.

Table C-1. Best values of ξ for Slater type orbitals.

Table C-2. Valence Orbital Ionization Potential parameters.

Table C-3. Electro-negativity.

Fig. C-4. Ionicity versus electro-negativity.

Table C-1 Best values of ξ for the ground state of neutral atoms.[a]

Z	1s	2s	2p	3s	3p	4s	3d	4p
2.	1.6875							
3.	2.6906	0.6396						
4.	3.6848	0.9560						
5.	4.6795	1.2881	1.2107					
6.	5.6727	1.6083	1.5679					
7.	6.6651	1.9237	1.9170					
8.	7.6579	2.2458	2.2266					
9.	8.6501	2.5638	2.5500					
10.	9.6421	2.8792	2.8792					
11.	10.6259	3.2857	3.4009	0.8358				
12	11.6089	3.6960	3.9129	1.1025				
13.	12.5910	4.1068	4.4817	1.3724	1.3552			
14.	13.5745	4.5100	4.9725	1.6344	1.4284			
15.	14.5578	4.9125	5.4806	1.8806	1.6288			
16.	15.5409	5.3144	5.9885	2.1223	1.8273			
17.	16.5239	5.7152	6.4966	2.3561	2.0387			
18.	17.5075	6.1152	7.0041	2.5856	2.2547			
19.	18.4895	6.5031	7.5136	2.8933	2.5752	0.8738		
20.	19.4730	6.8882	8.0207	3.2005	2.8861	1.0995		
21.	20.4566	7.2868	8.5273	3.4466	3.1354	1.1581	2.3733	
22.	21.4409	7.6883	9.0324	3.6777	3.3679	1.2042	2.7138	
23.	22.4256	8.0907	9.5364	3.9031	3.5950	1.2453	2.9943	
24.	23.4138	8.4919	10.0376	4.1226	3.8220	1.2833	3.2522	
25.	24.3957	8.8969	10.5420	4.3393	4.0364	1.3208	3.5094	
26.	25.3810	9.2995	11.0444	4.5587	4.2593	1.3585	3.7266	
27.	26.3668	9.7025	11.5462	4.7741	4.4782	1.3941	3.9518	
28.	27.3526	10.1063	12.0476	4.9870	4.6950	1.4277	4.1765	
29.	28.3386	10.5099	12.5485	5.1981	4.9102	1.4606	4.4002	
30.	29.3245	10.9140	13.0490	5.4064	5.1231	1.4913	4.6261	
31.	30.3094	11.2995	13.5454	5.6654	5.4012	1.7667	5.0311	1.5554
32.	31.2937	11.6824	14.0411	5.9299	5.6712	2.0109	5.4171	1.6951
33.	32.2783	12.0635	14.5368	6.1985	5.9499	2.2360	5.7928	1.8623
34.	33.2622	12.4442	15.0326	6.4678	6.2350	2.4394	6.1590	2.0718
35.	34.2471	12.8217	15.5282	6.7395	6.5236	2.6382	6.5197	2.2570
36.	35.2316	13.1990	16.0235	7.0109	6.8114	2.8289	6.8753	2.4423

[a] cited from E. Clementi et al, J. Chem. Phys. 38, 2686 (1963).

Table C-2. *VOIP Isoelectronic Fit Parameters*[a]

Number of Electrons	Configuration	Electron	Standard Deviation	*A*[b]	*B*	*C*
1	$1s$	$1s$	0.0	109.84	219.2	109.7
2	$1s^2$	$1s$	0.1	109.82	301.7	198.4
2	$1s\,2p$	$1s$	0.1	109.96	386.8	357.85
3	$1s^2\,2p$	$1s$	0.1	110.15	467.2	524.8
1	$2s$	$2s$	0.0	27.48	54.8	27.4
3	$1s^2\,2s$	$2s$	0.1	27.62	76.0	43.4
4	(He) $2s^2$	$2s$	0.1	27.64	100.3	75.1
4	(He) $2s\,2p$	$2s$	0.3	27.76	81.3	47.9
5	(He) $2s^2\,2p$	$2s$	0.1	27.82	120.6	113.4
5	(He) $2s\,2p^2$	$2s$	0.3	27.91	119.1	122.25
6	(He) $2s^2\,2p^2$	$2s$	0.2	27.95	141.6	156.6
6	(He) $2s\,2p^3$	$2s$	0.4	28.00	141.2	171.0
7	(He) $2s^2\,2p^3$	$2s$	0.2	28.16	162.2	206.2
7	(He) $2s\,2p^4$	$2s$	0.1	28.05	163.3	226.0
8	(He) $2s^2\,2p^4$	$2s$	0.3	27.95	184.6	260.8
9	(He) $2s^2\,2p^5$	$2s$	0.5	28.07	205.7	323.6
10	(He) $2s^2\,2p^6$	$2s$	0.1	28.29	227.0	390.9
1	$2p$	$2p$	0.0	27.48	54.8	27.4
2	$1s\,2p$	$2p$	0.1	27.52	57.8	28.6
3	$1s^2\,2p$	$2p$	0.2	27.74	59.1	28.4
4	(He) $2s\,2p$	$2p$	0.2	27.72	97.6	79.8
4	(He) $2p^2$	$2p$	0.4	27.57	76.1	45.35
5	(He) $2s^2\,2p$	$2p$	0.2	27.78	102.4	66.75
5	(He) $2s\,2p^2$	$2p$	0.3	28.02	90.1	67.0
5	(He) $2p^3$	$2p$	0.1	27.25	94.0	61.4
6	(He) $2s^2\,2p^2$	$2p$	0.3	27.95	118.2	85.8
6	(He) $2s\,2p^3$	$2p$	0.4	28.03	111.95	80.9
6	(He) $2p^4$	$2p$	0.1	28.06	105.4	88.1
7	(He) $2s^2\,2p^3$	$2p$	0.2	28.16	133.2	106.4
7	(He) $2s\,2p^4$	$2p$	2.1	30.01	114.0	129.4
8	(He) $2s^2\,2p^4$	$2p$	0.4	27.04	149.75	127.4
8	(He) $2s\,2p^5$	$2p$	0.1	27.76	145.2	126.4
9	(He) $2s^2\,2p^5$	$2p$	0.4	27.93	165.5	150.4
9	(He) $2s\,2p^6$	$2p$	0.8	28.22	157.7	155.1
10	(He) $2s^2\,2p^6$	$2p$	0.3	28.25	180.2	173.9
11	(Ne) $3s$	$3s$	0.0	13.18	68.0	41.0
12	(Ne) $3s^2$	$3s$	0.5	13.13	78.2	61.25
12	(Ne) $3s\,3p$	$3s$	0.5	13.14	78.6	71.7
13	(Ne) $3s^2\,3p$	$3s$	0.8	13.15	80.0	90.8
13	(Ne) $3s\,3p^2$	$3s$	0.0[c]	9.50	103.6	89.4
14	(Ne) $3s^2\,3p^2$	$3s$	1.3	13.08	99.9	119.6
14	(Ne) $3s\,3p^3$	$3s$	2.0	11.12	118.2	111.2
15	(Ne) $3s^2\,3p^3$	$3s$	0.3	14.27	100.7	151.4
16	(Ne) $3s^2\,3p^4$	$3s$	0.3	12.23	124.0	166.7
17	(Ne) $3s^2\,3p^5$	$3s$	0.2	13.70	120.7	203.8
18	(Ne) $3s^2\,3p^6$	$3s$	0.3	13.24	138.6	235.6
11	(Ne) $3p$	$3p$	0.7	13.33	49.4	23.9
12	(Ne) $3s\,3p$	$3p$	0.6	13.21	60.5	36.0
12	(Ne) $3p^2$	$3p$	0.6	12.05	61.9	41.3
13	(Ne) $3s^2\,3p$	$3p$	0.6	13.20	71.1	47.85
13	(Ne) $3s\,3p^2$	$3p$	0.6	12.44	75.65	42.8
14	(Ne) $3s^2\,3p^2$	$3p$	0.3	13.02	81.7	62.5
14	(Ne) $3s\,3p^3$	$3p$	2.8	8.82	110.7	19.4
14	(Ne) $3s^2\,3p\,4s$	$3p$	0.9	13.36	86.3	90.5
15	(Ne) $3s^2\,3p^3$	$3p$	0.0[c]	15.25	83.9	81.6
15	(Ne) $3s\,3p^4$	$3p$	1.3	14.25	83.85	100.1
15	(Ne) $3s^2\,3p^2\,4s$	$3p$	0.7	14.63	91.3	114.8
16	(Ne) $3s^2\,3p^4$	$3p$	0.7	13.17	98.5	93.4
16	(Ne) $3s\,3p^5$	$3p$	0.7	13.88	94.9	99.7
16	(Ne) $3s^2\,3p^3\,4s$	$3p$	0.4	13.57	102.4	131.6
17	(Ne) $3s^2\,3p^5$	$3p$	0.4	13.49	106.3	110.4
17	(Ne) $3s\,3p^6$	$3p$	0.0[c]	13.40	106.4	116.0
17	(Ne) $3s^2\,3p^4\,4s$	$3p$	0.2	13.36	112.0	153.3
18	(Ne) $3s^2\,3p^6$	$3p$	0.1	13.36	116.6	127.5
18	(Ne) $3s^2\,3p^5\,4s$	$3p$	0.6	13.39	121.5	175.3
19	(Ar) $3d$	$3d$	0.1	13.10	24.5	12.2
11	(Ne) $4s$	$4s$	0.3	7.47	29.0	15.5
12	(Ne) $3s\,4s$	$4s$	0.1	7.67	32.2	19.8
13	(Ne) $3s^2\,4s$	$4s$	0.2	7.65	36.1	22.85
14	(Ne) $3s^2\,3p\,4s$	$4s$	0.8	7.79	39.7	25.15
15	(Ne) $3s^2\,3p^2\,4s$	$4s$	1.0	9.30	38.5	31.8
16	(Ne) $3s^2\,3p^3\,4s$	$4s$	0.2	7.82	45.35	30.1
17	(Ne) $3s^2\,3p^4\,4s$	$4s$	0.3	8.00	48.0	32.1
18	(Ne) $3s^2\,3p^5\,4s$	$4s$	0.4	7.99	51.9	33.6
19	(Ar) $4s$	$4s$	0.4	8.09	53.5	34.7

All numbers in $1000cm^{-1}$

Multiplication by 0.12398 gives the number in eV.

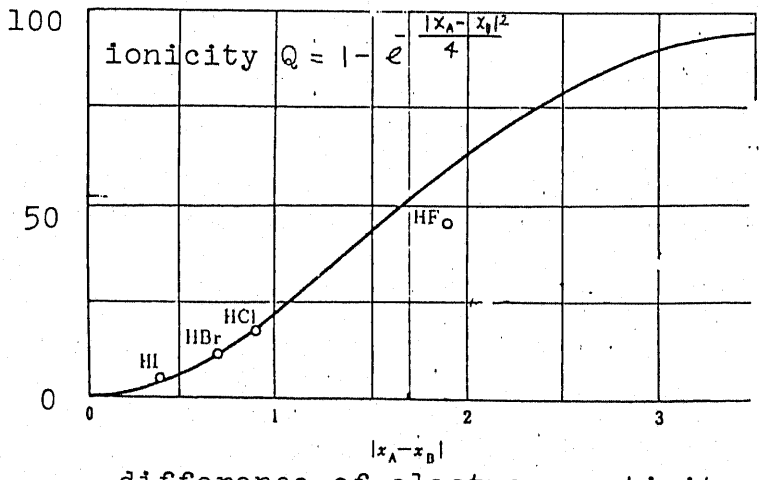
$VOIP(q)=Aq^2+Bq+C$,where q is excess charge.

H. Basch et al, Theoret. Chim. Acta 3,458 (1965)

Table C-3. Electro-negativity

H 2.1																
Li 1.0	Be 1.5											B 2.0	C 2.5	N 3.0	O 3.5	F 4.0
Na 0.9	Mg 1.2											Al 1.5	Si 1.8	P 2.1	S 2.5	Cl 3.0
K 0.8	Ca 1.0	Sc 1.3	Ti 1.5	V 1.6	Cr 1.6	Mn 1.5	Fe 1.8	Co 1.8	Ni 1.8	Cu 1.9	Zn 1.6	Ga 1.6	Ge 1.8	As 2.0	Se 2.4	Br 2.8
Rb 0.8	Sr 1.0	Y 1.2	Zr 1.4	Nb 1.6	Mo 1.8	Tc 1.9	Ru 2.2	Rh 2.2	Pd 2.2	Ag 1.9	Cd 1.7	In 1.7	Sn 1.8	Sb 1.9	Te 2.1	I 2.5
Cs 0.7	Ba 0.9	La-Lu 1.1-1.2	Hf 1.3	Ta 1.5	W 1.7	Re 1.9	Os 2.2	Ir 2.2	Pt 2.2	Au 2.4	Hg 1.9	Tl 1.8	Pb 1.8	Bi 1.9	Po 2.0	At 2.2
Fr 0.7	Ra 0.9	Ac 1.1	Th 1.3	Pa 1.5	U 1.7	Np-No 1.3										

Fig. C-4. Ionicity



APPENDIX D

DERIVATION OF Eq. 4-4-4

In the problem of crystalline solids, the eigen value is specified by a k-vector and a band number n. Therefore, the spectrum decomposition of Green's function is written as

$$(EI-H)^{-1}=\sum_{n,k}\frac{|E_n^k><E_n^k|}{E-E_n^k} \qquad \text{Eq. D-1}$$

Here, $|E_n^k>$ is expressed as

$$|E_n^k>=\frac{1}{\sqrt{N}}\begin{pmatrix}\vdots\\ |E_{no}^k>\\ |E_{no}^k>e^{ikR_1}\\ |E_{no}^k>e^{ikR_2}\\ \vdots\\ |E_{no}^k>e^{ikR_\nu}\end{pmatrix} \qquad \text{Eq. D-2}$$

This is derived from the Bloch's theorem. In Eq. D-2, $|E_{no}^k>$ represents the eigen vector belonging to the central unit cell, R_ν denotes the directional vector from the central unit cell to the ν-th unit cell, and N is the number of unit cells included in the system. Substitution of Eq. D-2 into Eq. D-1 yields

$$(EI-H)^{-1}=\frac{1}{N}\sum_k\begin{pmatrix}\ddots & & & \\ & G^k & G^k e^{-ikR_1} & G^k e^{-ikR_2}\\ & G^k e^{ikR_1} & G^k & G^k e^{-ik(R_2-R_1)}\\ & G^k e^{ikR_2} & G^k e^{ik(R_2-R_1)} & G^k\\ & & & \ddots\end{pmatrix} \qquad \text{Eq. D-3}$$

,where

$$G^k = \sum_n \frac{|En_o^k\rangle\langle En_o^k|}{E-En^k} \quad . \qquad \text{Eq. D-4}$$

Up to this point it is shown that the Green's function based on usual atomic orbitals as a basis set is written as

$$\text{The part of } (EI-H)^{-1} \text{ for the central unit cell} = \frac{1}{N}\sum_k G^k \qquad \text{Eq. D-5}$$

The next step is to know the meaning of G^k.

Substitution of Eq. D-2 into the original eigen value problem becomes

$$En^k \begin{bmatrix} \vdots \\ |En_o^k\rangle \\ |En_o^k\rangle e^{ikR_1} \\ |En_o^k\rangle e^{ikR_2} \\ \vdots \\ |En_o^k\rangle e^{ikR_\nu} \end{bmatrix} - \begin{bmatrix} \ddots & & & & \\ & H_{00} & H_{01} & H_{02} & \\ & H_{10} & H_{11} & H_{12} & \\ & H_{20} & H_{21} & H_{22} & \ddots \\ & & & & \end{bmatrix} \begin{bmatrix} \vdots \\ |En_o^k\rangle \\ |En_o^k\rangle e^{ikR_1} \\ |En_o^k\rangle e^{ikR_2} \\ \vdots \\ |En_o^k\rangle e^{ikR_\nu} \end{bmatrix} = 0$$

Eq. D-6

All the equations derived from Eq. D-6 turn out to be the same as

$$En^k|En_o^k\rangle - \left(\sum_\nu H_{o\nu} e^{ikR_\nu}\right)|En_o^k\rangle = 0 \quad . \qquad \text{Eq. D-7}$$

For example,

$$En^k\, e^{ikR_1}\,|\,En_o^k\rangle - \left(\sum_\nu H_{1\nu} e^{ikR_\nu}\right)|En_o^k\rangle = 0 \qquad \text{Eq. D-8}$$

can be rewritten as

$$e^{ikR_1}\left[En^k|En_o^k\rangle - \left(\sum_\nu H_{o\nu} e^{ikR_\nu}\right)|En_o^k\rangle\right] = 0 \quad . \qquad \text{Eq. D-9}$$

If we put

$$\sum_{\nu} H_{o\nu} e^{ikR_{\nu}} = H^{k} \quad , \qquad \text{Eq. D-10}$$

we have

$$(E^{k}_{n}I - H^{k})\,|E^{k}_{no}\rangle = 0 \quad . \qquad \text{Eq. D-11}$$

This eigen value problem happens to equal to that based on the Bloch orbitals as a basis set. This fact and the following equation

$$G^{k} = (EI - H^{k})^{-1} \qquad \text{Eq. D-12}$$

shows that G^{k} is a Green's function based on the Bloch orbitals as a basis set. So that with Eq. D-5, Eq. 4-4-4 is derived.