ANL-77-XX-90

THE REMOTE WORKING LEVEL MONITOR

Prepared for

UNITED STATES DEPARTMENT OF THE INTERIOR

BUREAU OF MINES

by

Argonne National Laboratory

with the

U. S. Energy Research and Development Administration

Argonne, Illinois

FINAL REPORT

on

The Remote Working Level Monitor

Contract No. HO 252053

Design and Construction of a Remote Working Level Meter

18 November 1977

DISCLAIMER NOTICE

November 1977

Performing Organization Rept. No:
P7447C
Final Report
Contract No. HO 252053

THE REMOTE WORKING LEVEL MONITOR

Donald J. Keefe

William P. McDowell

Peter G. Groer

Argonne National Laboratory
9700 S. Cass Avenue
Argonne, Illinois 60439

Sponsoring Organization

Office of the Assistant Director
Mining, Bureau of Mines, Department of Interior
Washington, D.C. 20241

Abstract

The Remote Working Level Monitor (RWLM) is an instrument used to remotely monitor the Rn-daughter concentrations and the Working Level (WL). It is an ac powered, microprocessor based instrument which multiplexes two independent detector units to a single central processor unit (CPU). The CPU controls the actuation of the detector units and processes and outputs the data received from these remote detector units. The remote detector units are fully automated and require no manual operation once they are set up. They detect and separate the alpha emitters of RaA and RaC' as well as detecting the beta emitters of RaB and RaC. The resultant pulses from these detected radioisotopes are transmitted to the CPU for processing. The programmed microprocessor performs the mathematical manipulations necessary to output accurate Rn-daughter concentrations and the WL. A special subroutine within the program enables the RWLM to run and output a calibration procedure on command. The data resulting from this request can then be processed in a separate program on most computers capable of BASIC programming. The calibration program results in the derivation of coefficients and beta efficiencies which provides calibrated coefficients and beta efficiencies.

Key Words
Remote Working Level Monitor (RWLM)
Working Level (WL)
Rn-daughter Concentrations

U.S. Security Classif. of Report and this Page:

UNCLASSIFIED

FOREWORD

This report was prepared by Argonne National Laboratory, Electronics Division, 9700 S. Cass Avenue, Argonne, Illinois 60439 under Contract Number HO 252053. The contract was initiated under the Metal and Nonmetal Health and Safety Research Program. It was administered under the technical direction of DMRC with Mr. Robert Droullard acting as the Technical Project Officer. Mr. David Askin was the contract administrator for the Bureau of Mines.

This report is a summary of the work recently completed as part of this contract during the period April 1975 to June 1977. This report was submitted by the authors on June 1977.

Subject Inventions

The design of the filter-advance mechanism was recently submitted to the Chicago Patent Group of the Energy Research and Development Administration for a patent review. The Chicago Patent Group of ERDA will contact the Bureau of Mines regarding the disposition of this patent application in accordance with the contract.

TABLE OF CONTENTS

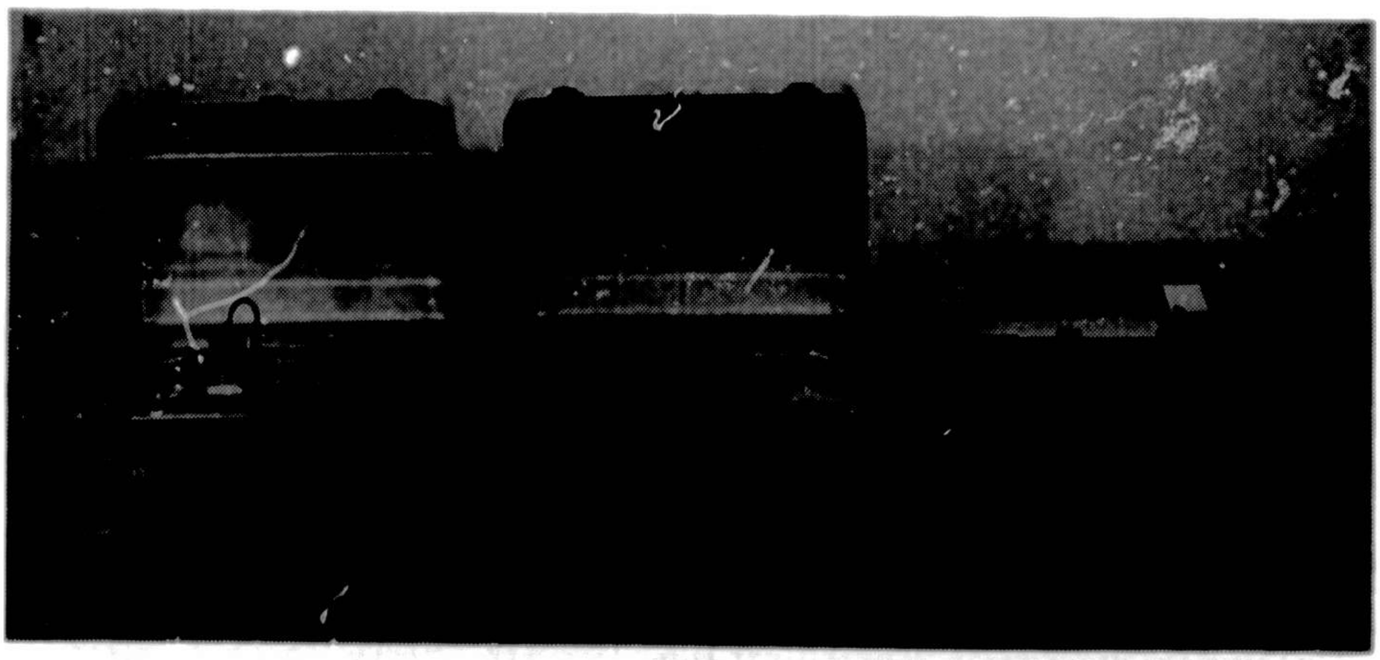

Figure 1 RW/LM System

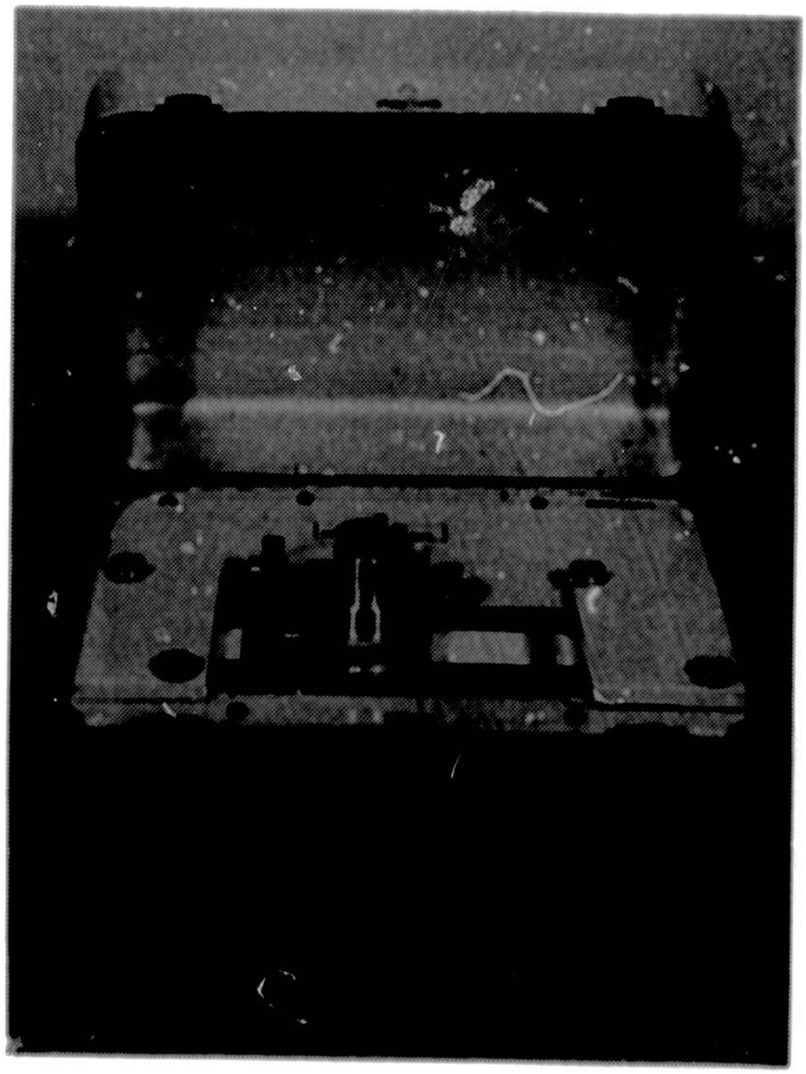

Figure 2 RW/LM Detector Unit #1 - Covers in place

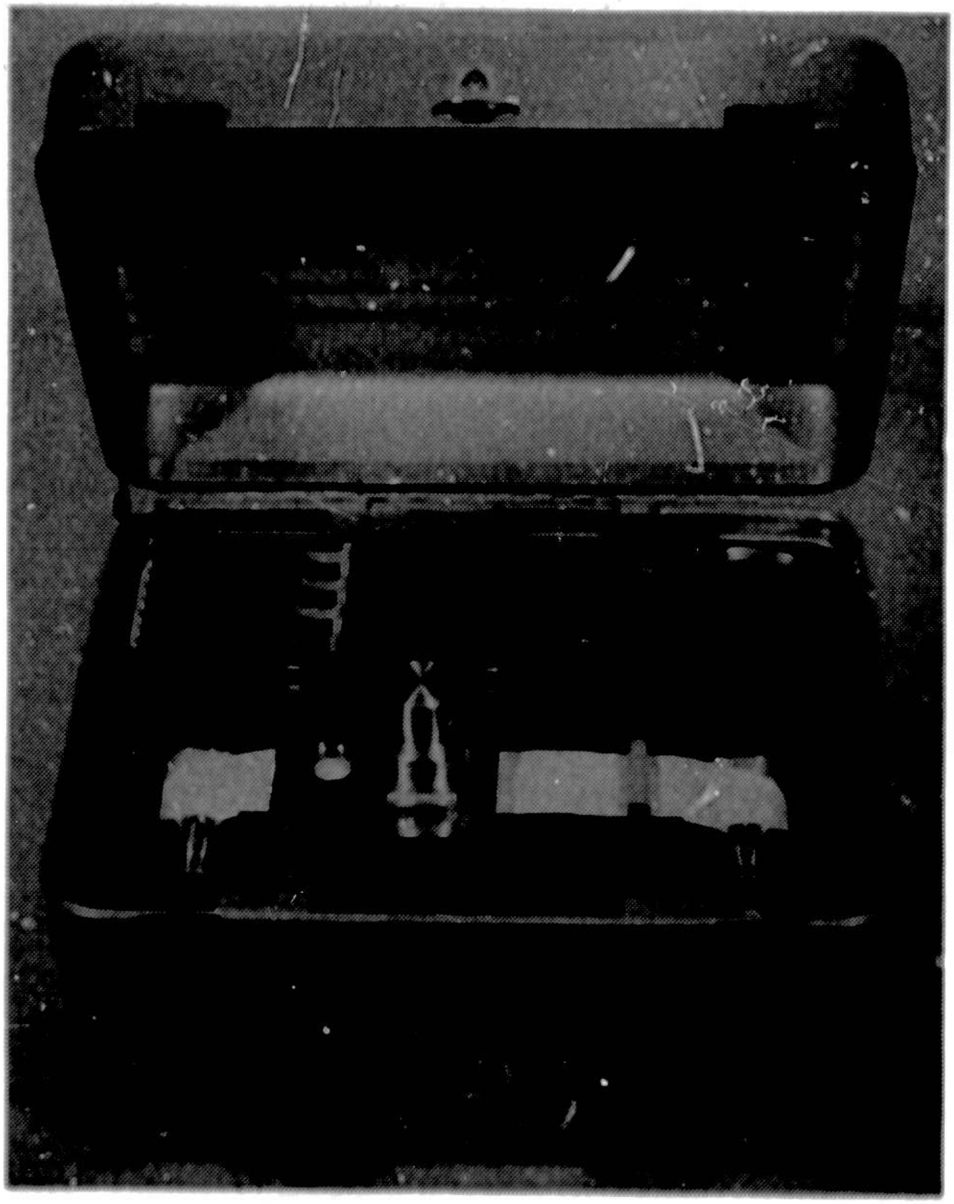

Figure 3 RW/LM Detector Unit #2 - Covers removed

Figure 4 Central Processing Unit (CPU)

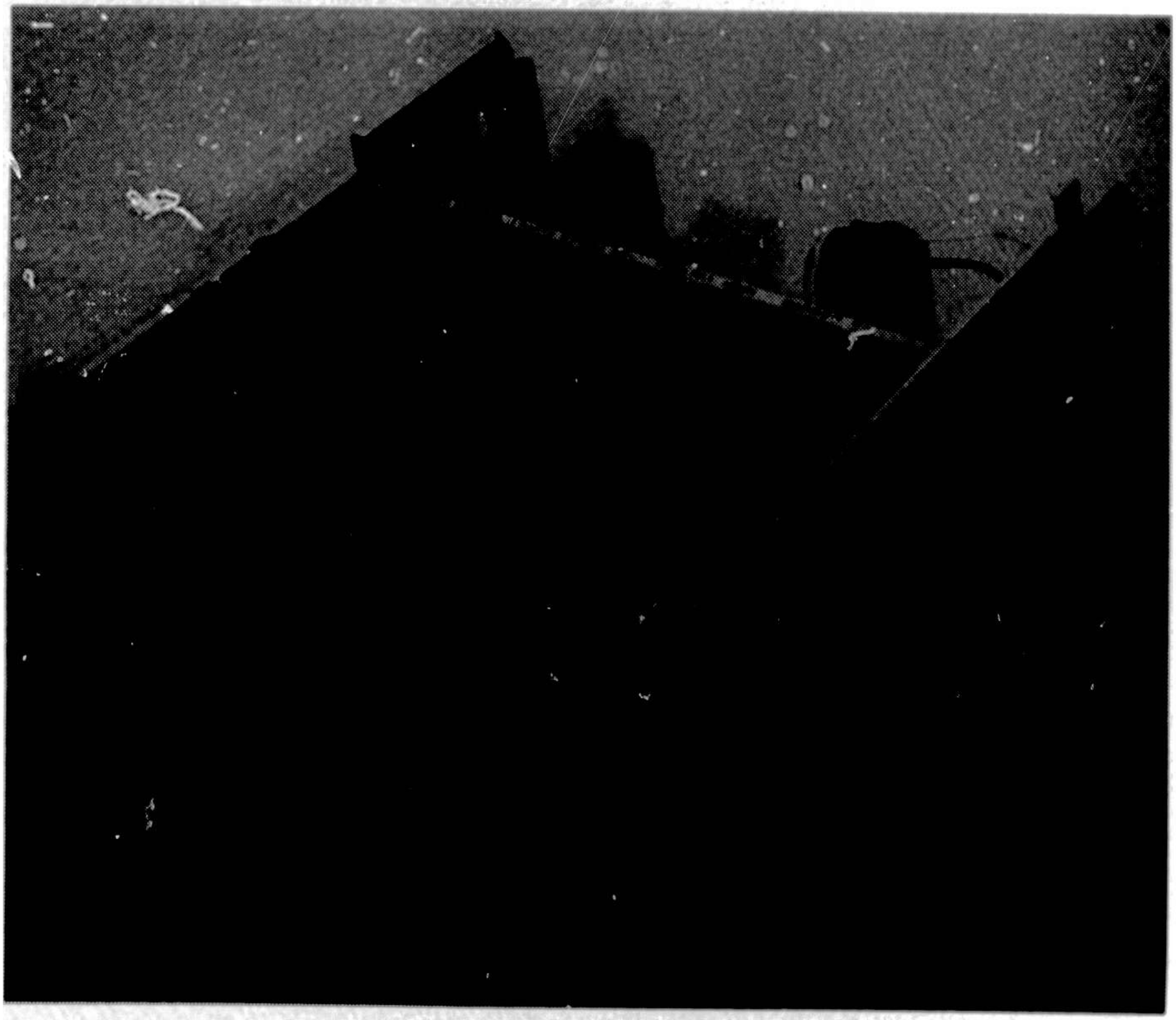

Figure 5 Exposed View of Central Processing Unit (CPU)

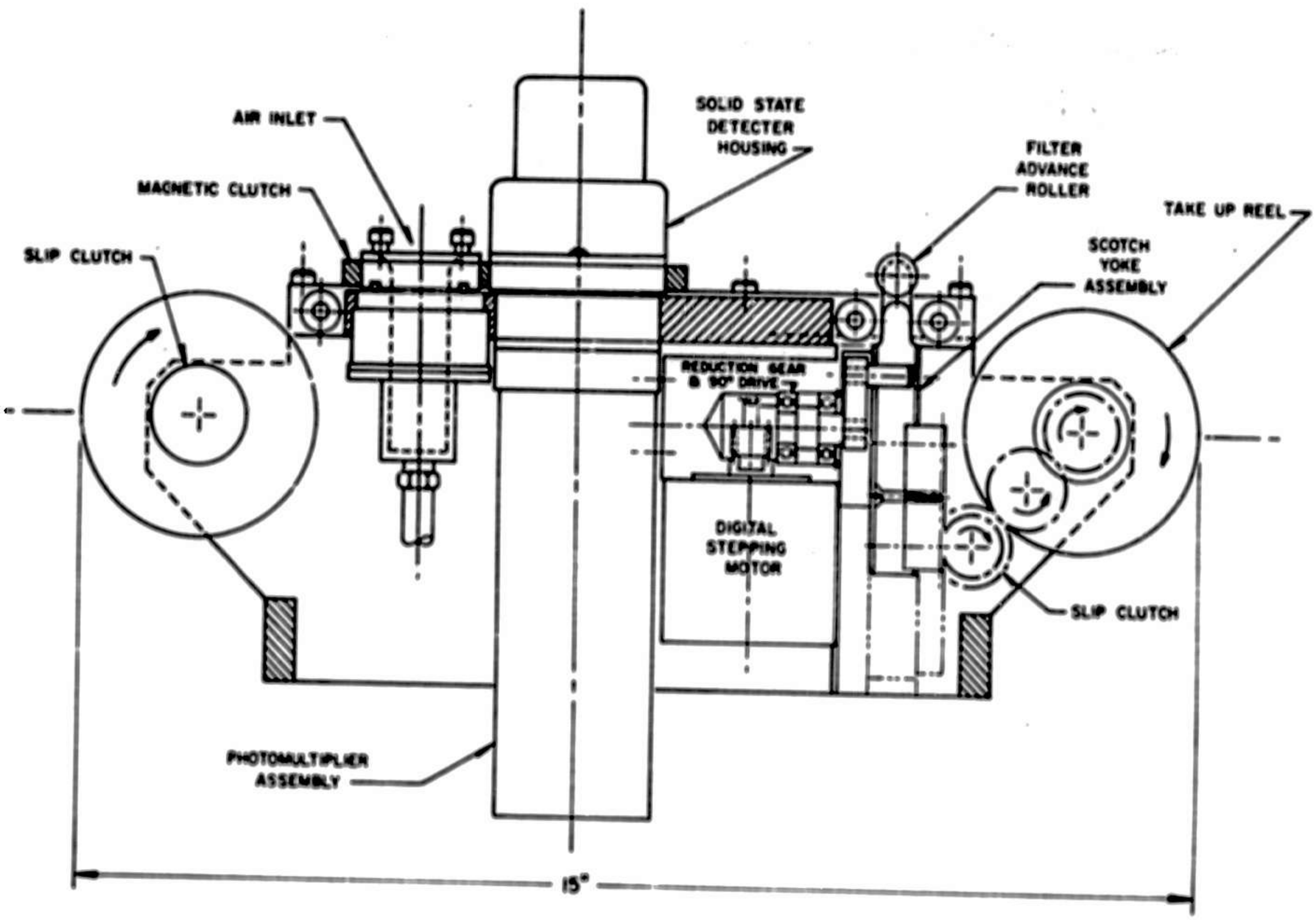

Figure 6 Schematic of Filter Transport Mechanism

1.0 INTRODUCTION

The exact methods commonly used to determine Working Level (WL) measure the total alpha activity on a filter membrane during three preselected time periods. Due to the half-lives of the isotopes involved (TRaB = 26.8 min. TRaC = 19.7 min.), these methods require 35 minutes to complete. Methods of measuring WL using the total alpha counts, where the counting time is less than about one RaB half-life after the end of sampling, are unsatisfactory because of statistical problems. In addition, these methods are almost blind to RaB (Pb^{214}) as a consequence of its relatively long half-life and the fact that RaB is a beta emitter. The solution to this dilemma is to measure RaB directly through its beta activity. Since RaB's daughter RaC (Bi^{214}) is also a beta emitter, the two beta spectra have to be separated to obtain RaB. How this is accomplished is described in subsequent sections describing the theory of the Remote Working Level Monitor/Instant Working Level Meter (RWLM/IWLM) and its calibration.

It is important to note that the IWLM, described in the Bureau of Mines Final Report prepared under contract No. HO 122106, and the RWLM described here use the same method to obtain the measurement of the radon-daughter concentrations and the WL. The IWLM is a portable battery-operated instrument which automatically performs the sequence of operations required after initiation of the start button. Its output is displayed visually and must be manually recorded. The RWLM is a multichannel (two channel in this case) totally automatic instrument designed to measure working levels and individual Rn-daughter concentrations and record the data. The RWLM is capable of taking up to 200 samples at operator-selected time intervals. The capability to perform these automatic operations is obtained through the use of a microprocessor as the instrument controller. The microprocessor enables the RWLM to keep real time, provide timing logic for the electronic sequencing necessary for proper operation, perform the necessary mathematical operations, communicate with the outside world through the data terminal, and perform its three different modes of operation. The microprocessor, in addition to performing the circuit logic through the use of software, also provides an automatic operational check of four mechanical functions and a functional check on remote power. It checks for proper operation of the filter positioning, adequate filter supply, release of the solenoid-operated air seal brake and monitors the pump speed within preselected limits. It any one of these parameters are beyond limits or not operational, the computer is flagged, resulting in a vectored interrupt which provides the specific error message to be printed on the data terminal and stops the measurement sequence. A Texas Instrument's Silent 700 data terminal is used for data recording and instrument control.

The instrument is thoroughly described in section 2. Its theory of operation and calibration are described in sections 2 and 4. Test results are tabulated in section 5, and section 6 contains the operating procedures for its three modes of operation. The drawings and computer programs are contained in Appendices A, B, and C.

2.0 THEORY

After taking a background count for two minutes, the RWLM collects an air sample for two minutes onto a filter membrane, moves the filter membrane to a counting station and then counts this air sample for two minutes starting 13 sec after the end of the sampling period. The instrument has three counting channels, the lower-energy alpha channel, the upper-energy alpha channel, and a beta sensitive channel. The RaA counts observed are accumulated in the lower-energy alpha channel; the RaC' counts are stored in the upper-energy alpha channel while the total beta counts from RaB and RaC are recorded in the beta channel. The counts in these three channels are functions of N_A, N_B, N_C, the unknown concentrations of RaA, RaB, and RaC in the ambient air (units are atoms/liter). The following relationships hold:

$$TA = 0.558814\ E_1 V N_A$$

$$B + C = (0.038498E_2 + 0.001793E_3)VN_A + (0.097712E_2 + 0.007473_3)VN_B + 0.130234E_3 VN_C$$

$$TC' = (0.001793N_A + 0.007473N_B + 0.130234N_C)E_1 V \qquad (1)$$

where:

TA = alpha counts from RaA accumulated during the 2-min counting period starting 13 sec after the end of sampling.

B + C = beta counts from RaB and RaC in beta channel accumulated during the same time interval as above.

TC' = alpha counts from RaC' (same period of accumulation as above).

V = flowrate (liters/min).

E_1 = detection efficiency for RaA and RaC'.

E_2 = detection efficiency for RaB.

E_3 = detection efficiency for RaC.

N_A = RaA concentration in units of atoms/liter.

N_B = RaB concentration in units of atoms/liter.

N_C = RaC concentration in units of atoms/liter.

The numerical coefficients in (1) follow from the laws of radioactive-series decay. The half-lives used are:

Nuclide	Half-life
RaA(^{218}Po)	3.05 min
RaB(^{214}Pb)	26.8 min
RaC(^{214}Bi)	19.7 min

To show the principle of calculation of these numerical coefficients, a sample calculation is given below:

EXAMPLE: Calculate the numerical coefficient for the first of (1).

This coefficient is the product of three factors:

1) Buildup factor = $1 - \exp(-\lambda_A t_B)$

t_B = buildup time = 2 min.

2) Delay factor = $\exp(-\lambda_A D)$

D = Delay before start of the counting interval

= 13 sec or 13/60 min.

3) Decay factor = $\left[1 - \exp(-\lambda_A t_D)\right] / \lambda_A$

t_D = decay time = 2 min.

λ_A = RaA decay constant = 0.2272614.

Multiplying all these factors one obtains:

$$\left[1 - \exp\left(-\lambda_A t_B\right)\right] \exp\left(-\lambda_A D\right) \left[1 - \exp\left(-\lambda_A t_D\right)\right] / \lambda_A = 0.558814.$$

The derivations of the analogous coefficients for the other equations are more complex but easily obtainable by the computer programs given in Appendix C. If all the efficiencies and the flowrate are known, (1) contains only numerical coefficients and the unknowns N_A, N_B, and N_C. Inverting (1) results in another set of equations which give a N_A, N_B, and N_C for every observed set of TA, (B+C), and TC'. The WL can be calculated easily from the resulting N_A, N_B, and N_C.

The RWLM method does not make any assumptions about the equilibrium conditions between the Rn daughters, but does assume that the airborne concentrations N_A, N_B, and N_C remain constant during the sampling period of two minutes. A detailed description of how the efficiencies are obtained and how (1) is inverted is given in section 4.0 and in the computer programs in Appendix C.

3.0 DESCRIPTIONS

The Remote Working Level Monitor (RWLM) consists of two independent detector assemblies, a central processing unit and a data terminal. (See Figs. 1 through 6.)

3.1 The Detector Assembly

Each detector assembly (see Figs. 2 and 3) is housed in a portable aluminum instrument case. This high-density package contains:

1. The mechanical drive mechanisms for the filter-transport assembly.
2. The air pump and its motor-tachometer assembly.
3. The necessary power supplies for the mechanical and electronic subassemblies.
4. The alpha- and beta-detector assemblies.
5. The electronic preamplifiers, discriminators, and single-channel analyzer.
6. The electronic control package.

3.1.1 The mechanical filter-transport assembly (see Fig. 6) advances the filter membrane 2.000 ± 0.005 inches each transport cycle. The sequence of mechanical operation is as follows:

1. The filter paper is advanced two cycles to provide a clean surface for background measurements.
2. The pump draws a sample.
3. The filter paper is advanced again to move the filtered sample to the detectors for counting.

Six inches of filter membrane* are used per measurement. Each sensor has a sufficient filter supply to allow up to 200 samples, either one sample per hour for eight days or any other programmed frequency limited by a total of 200 samples and a 14 min or greater sampling period.

This operation is motor driven, and computer controlled. As shown in Fig. 6, a 5V, bifilar-wound stepping motor** provides 200 steps per revolution. a 5:1-90° drive and reduction gear couples the motor to a "Scotch Yoke" drive assembly. One revolution of this drive assembly will provide one filter-advance cycle and requires 1000 pulses from the stepping motor per cycle.

The filter advance cycle is implemented as follows: The power to the magnetic clutch is turned off, releasing the clutch which is under mechanical spring pressure. Upon activation of pulses to the digital motor, the filter advance roller moves down due to the mechanical conversion of angular to linear motion of the scotch yoke mechanism. This advances the paper from the storage reel. At the 500th pulse the filter advance roller is at its bottom dead center position, which completes the filter advance. At this time, the computer energizes the magnetic clutch,

---

*GELMAN ACROPOR AN-800

**Superior Electronics SLO-SYN Model M061-FC02.

braking the filter membrane and preventing further motion of the filter. Due to the action of the slip clutch and the one-way bearing on the take-up reel and its drive, the filter paper is taken up during the last 500 pulses of the motor operation. The four-phase drive signal to the stepping motor is generated by a digital motor-translator circuit explained in section 3.1.6.3.

3.1.2 The Air-Pump and Motor-Tachometer Drive

A GAST rotary-vane vacuum pump, Model 1033, is coupled with a universal self-aligning coupler to a printed circuit motor-tachometer combination, Model U9M4T. The pump-motor system is servo-regulated using tachometer feedback for a constant-speed operation. It is activated under computer control. The interface circuit is described in section 3.1.6.3. The pump speed is adjusted to provide a flowrate of 12 liters per min. The computer-controlled solenoid-operated magnetic clutch assures a tight air seal of the filter against the inlet air port.

3.1.3 Regulated DC Power Supplies

The following regulated DC power supplies are included within each portable detector assembly:

1. An ELEXON Model OLV30-15 (15V @ 3.3 amp) regulated power supply provides power for the air pump motor.

2. An Analog Devices Model PS 933 (±24V @ 50 mA) regulated power supply and a ±12V @ 50 mA regulated supply (drawing EL-C-7171) provide power for the alpha detector and preamp.
3. A Semiconductor Circuits, Model MP 1.5. 750/ 2.15.100 (±15V @ 100 mA plus +5V @ 750 mA) regulated power supply provides power for the linear circuits and the digital logic.
4. A Semiconductor Circuits, Model DPS 1.5.1500 (+5V @ 1.5 A) regulated power supply provides power for the solenoid.
5. A 5V @ 3 A regulated supply (see drawing EL-C-7171, sheet 7, Appendix B) provides power for the stepping motor.

The AC line power is supplied to the detector assemblies via a power cable from the central processing unit and is independently fused within the detector assembly. Each detector assembly also contains its own power "ON/OFF" switch.

3.1.4 Radiation Detectors

The alpha detector is a silicon-surface barrier detector.* The face of the detector is protected with double-aluminized Mylar (25 gauge). The detector is sensitive to both the 6.00 MeV RaA and the 7.68 MeV RaC' alpha particles.

*Ortec #CA-29-300-100.

The beta detector is a NE 102 Scintillator optically coupled to a 10-stage, low-noise, high-gain photomultiplier (EMI 9633B). The gamma-background sensitivity has been reduced by a factor of 50 over that of the original MIT IWLM by the provision of an integral lead shield for the beta detector and the use of a thin scintillator (0.003 in).

3.1.5 Detector Preamplifier and Single-Channel Analyzer

The alpha channel utilizes a Canberra preamplifier Model 1406, whose output is further amplified by a dual integrated-circuit amplifier having a gain of 100. This combination nets a charge gain of 20 V/pico-coulomb for the alpha channel. The output of this charge-sensitive amplifier is delivered to a single-channel analyzer which separates the 6.00 MeV RaA energy peaks from the 7.68 MeV RaC' energy peaks. (See Appendix C, drawing EL-C-7171, sheet 12.) These separated energy pulses are routed via line drivers to the appropriate accumulators in the central processing unit (CPU).

The beta channel's preamplifier is contained within the photomultiplier tube, voltage divider and preamplifier housing. The charge gain of this amplifier is fixed at .05 V/pico-coulomb. A regulated high-voltage converter (Venus K30) is used to supply a well regulated voltage to the dynode bias divider to insure the gain stability of the photomultiplier.

A voltage of approximately 1250V was found to be a satisfactory value for the PM tube. At this value the optimum signal-to-noise ratio is obtained. The output pulses from the PM tubes' charge-sensitive amplifier are routed to a low-level discriminator which prevents the counting of noise pulses. The signals that satisfy the discriminator requirements are shaped, routed to a line driver and then to the beta-channel accumulator in the CPU.

3.1.6 <u>The Electronic Control Package</u>

This package contains four printed circuit cards which perform the following functions:

Card 1: Line drivers and opto-isolated line receivers (see drawing EL-C-7171, sheet 10 of Appendix B).

Card 2: Tachometer output window comparator, high-voltage regulator, solenoid driver and the pump-motor speed control (see drawing EL-C-7171, sheet 11 of Appendix B).

Card 3: Slo-Syn motor translator control (see drawing EL-C-7171, sheet 4 of Appendix B).

Card 4: Source check logic control and paper position and paper empty comparators (see drawing EL-C-7171, sheet 9 of Appendix B).

3.1.6.1 <u>Line Drivers and Opto-Isolated Line Receivers</u>

The two detector assemblies and the central processing unit are separated by up to 500 ft of shielded cable. To drive digital pulses, with minimum distortion over this distance, requires line drivers and line receivers. The line drivers are designed to drive a 50 ohm line which is terminated in the characteristic impedance of the cable. Opto-isolators are used to terminate the line, as well as to provide electrical isolation between the three units. This further reduces noise-induced signals and electrically isolates the computer from the detection heads. Signals from the CPU are terminated within the particular detector assembly and the return signal is returned to the CPU ground via the remaining wire of the twisted pair. Signals from the detector assembly are terminated within the CPU and the ground is returned in a similar fashion to the detector assembly.

3.1.6.2 Tachometer Output Window Comparator, High-Voltage Regulator, Solenoid Driver and Pump-Motor Speed Control Circuit

Since the pump speed is approximately proportional to the flowrate, the subject of course to wear the efficiency, pump speed is regulated using a tachometer. (See Appendix B, sheet 6, tachometer vs flowrate graph.) The tachometer voltage is sensed and electrically compared to an upper and lower limit. If the tachometer voltage does not fall within this preselected window, it is sensed as an error by the comparator. The comparator output is monitored by the CPU and an off normal signal causes the error message "FLOWRATE OUT-OF-RANGE" to be types on the terminal. If this occurs, the measurement cycle is aborted and the CPU returns to command loop.

A buffered reference signal is fed to the low-voltage input of a high-voltage DC to DC converter (see drawing EL-C-7171, sheet 11). A temperature compensated zener reference feeds a potentiometer which provides a variable reference for a closed-loop feed-back-buffer amplifer. This amplifier supplies the regulated voltage and current necessary for the regulated high-voltage converter.

This regulated high voltage is used by the beta channel's high-voltage photomultiplier bias string. Regulating this voltage provides a measure of stability in the gain of the photomultiplier.

The solenoid driver provides the necessary current to activate the solenoid-driven clutch.

The circuit itself is basically a current-controlled switch. A provision for a solenoid status check signal is fed back to the computer which checks for this signal at the appropriate times. If this flag signal is present when checked, the message "SOLENOID NOT RELEASED" is printed on the terminal and the measurement operation is aborted.

A closed-loop servo-controlled motor-drive circuit is provided to regulate the angular velocity of the pump motor, keeping the pump at a constant regulated speed. Pump activation is computer controlled.

3.1.6.3 Slo-Syn Motor Translator Control

A four phase, 5V, one Amp, step sequence (full-step mode) is required to drive the stepping motor. This four-bit driver code is generated by the translator control card. This circuit operates as follows: A hexidecimal divider is either counted up or down depending on the motor-rotation direction desired. The binary output of this divider provides a four-bit address to a programmable read-only memory which has the proper motor code stored. The memory outputs provide the current drivers with the proper input code sequence. The chip-enable input of the PROM provides an On/Off switch for these drivers. The motor direction is controlled by either an up or down count, the motor stepping speed is controlled by the pulse-repetition rate to the counter, and power is provided by the PROM chip-enable input. This motor can be driven to a maximum speed of 330 steps/sec of a 3 sec paper-advance cycle.

3.1.6.4 <u>Source Check Paper Position and Paper Empty Logic</u>

In the source-check mode of operation, the discriminator setting for the RaC' energy must be lowered in order to detect the Am^{241} source used in the source check. To accomplish this, a linear switch is automatically actuated by the CPU to provide a lower voltage to the upper-level channel when a C' source check operation is selected. The upper-level discriminator is usually set at approximately 1.3 volts to detect the 7.6 MeV RaC' energy. When a source check mode is entered, this voltage is automatically set to approximately 0.4V.

The paper position and paper empty comparators provide logic signals to the CPU which enable the CPU to check these parameters. To accomplish this, each comparator receives a signal from a light-sensitive detector (Skan-a-matic). These detectors provide their own infrared light source and are able to detect reflected IR energy and provide a current output proportional to the magnitude of IR-reflected light received. The current output is fed through a high-impedance resistor which provides a signal voltage to the comparators where it is compared to a common reference voltage. As long as the light is not reflected, the current switch is turned off. However, when this light source is detected, due to reflection, the current switch is turned on until the paper is exhausted. When the filter feed reel is empty, the light source projects against a black surface causing no reflection. The paper-position detector looks for a white scribe mark etched on one end of the scotch yoke cam drive assembly which appears within the view of the detector only when the filter advance roller is at top dead center.

3.2 Central Processing Unit (CPU)

The RWLM central processing unit consists of a chassis with an integral power unit containing the following:

1) an IMSAI MPU-A-8080 processor board
2) an IMSAI RAM 4-1 random access memory board with 1K of memory.
3) a Cromemco 8K read only memory board
4) a Processor Technology 2KRO -2K read only memory board
5) a Processor Technology 3P+S - serial I/O board
6) a vectored interrupt board
7) a head control and line driver board
8) a line receiver board
9) an accumulator board
10) a front panel substitute board
11) a system bus board
12) a system power supply

The CPU is housed in an IMSAI corporation main frame modified to meet the needs of the RWLM. The modification consists of the addition of system I/O cable connectors and system power connectors.

3.2.1 IMSAI MPU-A Theory of Operation

The IMSAI MPU-A board is structured around the Intel 8080A microprocessor chip, and much of the MPU-A board is wired to support the 8080A device. The MPU-A board provides interfacing between the 8080A chip and the data and address busses, clock and synchronization signals, and the voltage regulation necessary for the 8080A and other chips.

The address lines from the 8080A drive the address bus on the back plane through 8T97 tri-state buffer drivers. These drivers may be disabled through the ADDRESS DISABLE line on pin 22 of the back plane. Intel 8216 bi-directional bus drivers connect the 8080's bi-directional data bus to the back plane's dual uni-directional DATA IN and DATA OUT busses. The direction of data transmission is determined by the DIRECTION ENABLE line. The DIRECTION ENABLE line is in turn controlled by the front panel and the processor status signals DATA BUS IN and HALF ACKNOWLEDGE. The 8216 can be disabled by the DATA OUT DISABLE line on pin 23 of the back plane.

The 8080A's bi-directional data bus is also connected to the data bus socket and the 8212 status byte latch. The data bus socket is used to connect a front panel (not included) to the bi-directional bus, while the 8312 latch transfers the status byte to the back plane via 8T97 drivers. These drivers are disabled by the STATUS DISABLE line on pin 18 of the back plane. The 8212 is latched up by the STATUS STROBE signal of the 8224 clock chip to store the status information for each instruction cycle.

One K pullup resistors to +5 volts are connected to all the bi-directional bus lines to ensure that during the time the bus is not driven, the 8080A reads all 1's.

The 8824 clock ship and crystal oscillator, provide the two-phase non-overlapping 2 megacycle system clock for the 8080A. These clocks are also driven onto the back plane through 8T97 tri-state buffered drivers.

The CLOCK line on the back plane is driven from the TTL Phase II clock line through a delay so that the phase relation of the clock signal to the Phase II and Phase I back plane signals is nearly identical to that produced by the MITS Altair 8800 system. Six sections of a 7404 are used for this delay to provide greater simplicity and higher reliability than a one shot. The 8224 chip also provides the power-on reset function through use of a 4.7K resistor and 33 uF capacitor connected to the reset input of the 8224. The power-on reset is applied to the 8080A and is applied to the POWER ON CLEAR line, pin 99 on the back plane.

The two BACK PLANE READY signals are ANDed and connected to the 8224 for synchronization with the Phase II clock before being connected to the 8080A chip. The INTERRUPT line is connected directly to the 8080A, while the HOLD REQUEST line is synchronized with the Phase II clock and then connected to the 8080A.

The six processor status signals (SYNC WRITE, STROBE DATA BIT IN, READ STROBE, INTERRUPT ENABLED, HOLD ACKNOWLEDGE, and WAIT ACKNOWLEDGE) are all driven onto the back plant through 8T97 tri-state buffered drivers. These drivers may be disabled by the CONTROL DISABLE line, pin 19 on the back plane.

The +5 volts is regulated from the +8 volts by a 7805 integrated circuit regulator, while the -5 volts is regulated by a 5-volt zener and a 470 ohm resistor from the 16-volt bus. The +12 volts is regulated by a 12-volt zener and connected to the +16-volt line by two 82 ohm 1/2 watt resistors in parallel. All voltages are filtered with .33 microfarad tantalum and disc ceramic capacitors.

3.2.2 <u>IMSAI RAM 4-1 Theory of Operation</u>

The RAM-4 board has space for 4K bytes of memory which consist of 32 chips of Intel 8111 or 2111 type random access memory organized 456 words x 4 bits wide in each chip. In the RWLM system only 1K bytes are implemented.

These RAM devices are arranged on the board in a 2 x N ($1 \leq N \leq 16$) array, with the top row A containing bits 0,1 2,3 of all the data and Row B containing bits 4,5,6, and 7 of all the data. Read/write and address control is provided by a support network of Gates (C8, C9, C13) and a decoder (C10). Bi-directional tri-state bus drivers (C15, C16) are used to receive and transmit data to and from the IMSAI 8080 System bus.

To begin the Read or Write Cycles, the board must be enabled. As shown in the schematic, the board enable is produced by an 8-input NAND (741LS30 in position C13). Four of the NAND inputs are the jumper selected board address bits (A12, A13, A14, A15 or complements), and the remaining two are the inverted status bits SINP and SOUT. When the board is properly addressed, the NAND output is driven low. The 8205 1-of-8 decoder is then enabled, addressing a particular memory chip pair uniquely determined by the states of A8, A9, A10 and A11.

The 8T97 bus driver (C14) is also driven by the NAND (C13). Also enabled at this time are the 8216 (C15, C16) tri-state bi-directional bus drivers.

The direction of data flow is determined by the 7402 in position C8, which when low selects a data path going from the IMSAI 8080 data bus to the RAM-4 board's data bus. This is made low by either the memory write line from the control panel or the complement of the memory read status signal from the processor. Thus for normal operation, with the machine running, the status signal memory read determines whether these data bus drivers are driving to the IMSAI 8080 data-out bus. When the machine is stopped and the front panel is being used, the direction of data transmission is selected by the memory write pulse from the front panel. When writing from the front panel, a delay is necessary before turning off the data on the memory chips (so that there is time for the memory chips to write on the trailing edge of the write strobe before the data disappears) and this delay is provided by the disc capacitor to ground connected at the output of the inverter at C9 pin 8. In addition to selecting the direction of data flow through the bi-directional data bus drivers, the direction control signal is also inverted and applied to the output disable pin on the 8111's so that during writing the 8111 is receiving data on its bi-directional data pins and not attempting to drive. The write strobe is applied to the 8111's through a 4 section data out DIP switch which enables the programmer to turn off the write pulse for each K for debugging purposes. When the machine is running normally, the write comes from the processor write strobe line (pin 77 on the back plane) and when the front panel is being used, the write strobe line comes from the front panel on the memory write line (pin 60 on the back plane). Two other sections of the 7402 are used to take either one of these write strobes and buffer them to drive the memory chips.

3.2.3 Cromenco 8K Bytesaver Read Only Memory

This board is a read only memory board plug compatible with the standard S-100 microcomputer bus. It has the capacity to hold eight 2708 type UV erasable read only memories.

The board also contains provisions for programming the 2708 type memories; however, the RWLM does not support the software required to perform this function. In the RWLM seven of the eight sockets are used. This broad is addressed at Zero and Stores the main RWLM program.

3.2.4 Processor Technology 2KRO 2K Read Only Memory

This board is a read only memory, socket compatible with the standard S-100 microcomputer bus. It has the capacity to hold eight 1702A type UV erasable read only memories. Each 1702A is capable of holding 256 bytes of program. In the RWLM this board holds the floating point mathematics package and the floating point-PLM transfer routines.

3.2.5 Processor Technology 3P+S Input Output Module

This board is designed to provide the interface between the microcomputer system and the terminal device. In the RWLM this board is wired to implement the standard RS-232C signals required by the TI Silent 700.

The board contains a serial I/O port which links the terminal to the system and a parallel port which is designed to provide I/O status information to the processor. It also contains two parallel I/O ports that are not used by the RWLM.

3.2.6 Vectored Interrupt Board

The vectored interrupt board provides the processor with an eight level priority interrupt capability and a controlled interval clock that can be used to interrupt the processor on a regular basis. In the RWLM a one second interrupt is used to provide the timing information. The program control of the vector interrupt board is performed via output port OFBH. The output of the address select gate (74L30) is ANDed with processor signals PWR and SOUT. These signals are used to latch the lower four bits of data into the 8214 priority interrupt chip. When 8214 is enabled and one of its priority request lines is low, the 8314 INT line is used to strobe an 8212 IC. This causes the 8212's INT flip-flop to be set and requests an interrupt from the processor. When the processor acknowledges the interrupt with an INTA, the 8212 outputs are enabled and this puts the interrupt request address on bits 3, 4, and 5 of the DATA IN bus. The remaining bits of DATA IN are held high. The byte thus formed on the DATA IN lines is a restart instruction with bits 3, 4, and 5 directing the processor to one of eight restart locations.

The clock circuit consists of a set of frequency dividers which count the phase 2 clock pulses. In the RWLM, the one second intervals are selected. The clock interrupt flip-flop is reset by ANDing the RST 7 instruction with the INTA signal. Details of operation on the 8212 and the 8214 can be found in the Intel Data Book.

3.2.7 Head Control on Select Board

The Head control board consists of an I/O address decoder, an 8212 used as an output port latch, and two sets of line drivers. When a command is to be sent to the remote heads an OUT 7 instruction is placed on the bus. This is decoded by the 8205 whose output, along with PWR and SOUT cause the 8212 to latch the output instruction. The output lines of 8212 are then used to select the head and to hold the required information. Bit 0 selects the head to be addressed via a set of 7408 AND gates. When Bit 0 is equal to 0, head one is selected and when Bit 0 is equal to one, head two is selected. Bits 2 and 3 are not used. Bit 4 delivers pulses to the digital motor, Bit 5 turns on the digital motor, Bit 6 activates the solenoid and Bit 7 activates the pump.

3.2.8 Line Receiver Board

The line receiver board consists of a set of opto-isolator line receivers which detect the signals sent by each remote head. Head selection is performed by 72LS158 multiplexers which select either head one or head two using the head select bit from the output port on the Head control board.

This data is placed on the processors data bus whenever an IN 7 instruction is performed. Address decoding is performed on the Head Control Board and signals are bussed to the 8212 on the line receiver board.

3.2.9 The Accumulator Board

In order to count the pulses in each of the detection channels (RaA, Ra(B+C) and RaC'). A seven-decade accumulator has been implemented for each channel. The accumulator section consists of a MC 14518 decade prescaler which holds the least significant decade with a Mostek MK 50395N six decade up/down counter with latch and comparator.

The accumulator board is controlled as output port seven by an 8212 latch. The data is read out of the accumulators onto the data bus by a set of 8212 latches used as input ports, four, five and six. Output port 4's output commands are:

1) $\overline{\text{STOREL}}$ - This strobes the output 8212 and causes Them to hold the data being impressed on their input lines.

2) $\overline{\text{SET}}$ - This resets the multiplexed output of the MK50395N to the most significant digit.

3) SCAN - This strobes the MK50395N output selector to output the next digit.

4) $\overline{\text{CLEAR}}$ - Resets all counters.

5) COUNT ENABLE - opens the count gates and allows the accumulators to collect data.

6) STORES - Enables latch in MK50395N.

The MK50395N is a N-MOS device and requires a VCC of +12 volts; therefore, open collector 7406 devices are used to interface with the output port and the RCA 3081. Seven transistor common emitter IC devices are used to interface with the input port.

3.2.10 Front Panel Replacement Card

In the normal S-100 system, a front panel is used which generates a MWRITE signal. When no front panel is installed, a way of generating this signal must be provided. A single 7400 NAND gate is used to implement the function.

$\text{MWRITE} = \text{PWR}\cdot\overline{\text{SOUT}}$

3.2.11 System Bus

The S100 system bus structure consists of 100 lines. These are arranged 50 on each side of the plug-in board, with pins 1 through 50 on the component side and pins 51 through 100 on the back side.

As the board is viewed right-side up (components up, 100 pin connector towards you) pin #1 is on the left end on the top and pin 51 is on the back side directly opposite pin #1.

Conventions:

SYMBOLS: "P" prefix indicates a processor command or control signal
"S" prefix indicates a processor status signal

LOADING: All inputs to a card should be loaded with a maximum of 1 TTL low power load

LEVELS: All bus signals except the power supply are TTL. All Data and Address lines are positive TRUE (ground = 0 bit)

BUS DEFINITION

Front Side

No.	SYMBOL	NAME	FUNCTION
1	+8V	+8 volts	Unregulated input to 5 V regulators
2	+16V	+16 volts	Positive unregulated voltage
3	XRDY	External Ready	Used by Front Panel: Pulling this line low will cause the processor to enter a WAIT state and allows the status of the normal Ready Line (PRDY) to be examined.

BUS DEFINITION

Front Side

No.	SYMBOL	NAME	FUNCTION
4	V10	Vectored Interrupt Line #0	
5	V11	Vectored Interrupt Line #1	
6	V12	Vectored Interrupt Line #2	
7	V13	Vectored Interrupt Line #3	
8	V14	Vectored Interrupt Line #4	
9	V15	Vectored Interrupt Line #5	
10	V16	Vectored Interrupt Line #6	
11	V17	Vectored Interrupt Line #7	
12	HS	Head Select Bit	
13		SINP·PDBIN for Head Output Port	
14-16	UNUSED		
17		HEAD ADDRESS SELECT	$\overline{\text{I/O SEL}}$
18	$\overline{\text{STATUS DSBL}}$	STATUS DISABLE	Allows the buffers for the 8 status lines to be tristated

BUS DEFINITION

Front Side

No.	SYMBOL	NAME	FUNCTION
19	$\overline{\text{CC DSB}}$	COMMAND CONTROL DISABLE	Allows the buffers for the 6 output c command/ control lines to be tri-stated
20	UNPROT	UNPROTECT	Reserved for input to the memory protect flip-flop on a given memory board
21	SS	SINGLE STEP	Used by Front Panel to disable input buffer while panel drives bidirectional data bus
22	$\overline{\text{ADDR DSBL}}$	ADDRESS DISABLE	Allows the buffers for the 16 address lines to be tri-stated
23	$\overline{\text{DO DSBL}}$	DATA OUT DISABLE	Allows the bidirectional data bus drivers for the 8 data lines to be tri-stated for both input and output data buses
24	Ø2	Phase 2 Clock	
25	Ø1	Phase 1 Clock	
26	PHLDA	Hold Acknowledge	Processor control output signal which appears in response to the HOLD signal;

BUS DEFINITION

Front Side

No.	SYMBOL	NAME	FUNCTION
26(continued)			indicates that the data and address bus will go to the high impedance state on the 8080. Note: $\overline{\text{ADDR DSBL}}$ and $\overline{\text{DO DSBL}}$ must be driven to the tri-state the system bus
27	PWAIT	WAIT	Processor control output signal which acknowleges that the processor is in a WAIT state
28	PINTE	INTERRUPT ENABLE	Processor control output signal indicating interrupts are enabled: may be set or reset by EI and DI instruction and inhibits interrupts from being accepted by the CPU if it is reset
29	A5	Address Line #5	
30	A4	Address Line #4	
31	A3	Address Line #3	
32	A15	Address Line #15	
33	A12	Address Line #12	

BUS DEFINITION

Front Side

No.	SYMBOL	NAME	FUNCTION
34	A9	Address Line #9	
35	DO	Data Out Line #1	
36	DO0	Data Out Line #0	
37	A10	Address Line #10	
38	DO4	Data Out Line #4	
39	DO5	Data Out Line #5	
40	DO6	Data Out Line #6	
41	DI2	Data In Line #2	
42	DI3	Data In Line #3	
43	DI7	Data In Line #7	
44	SM1	M1	Status output signal that indicates that the processor in in the fetch cycle for the first byte of an instruction
45	SOUT	OUT	Status output signal which indicates that the address bus contains the address of an output device and the data bus will contain the output data when $\overline{\text{PWR}}$ is active

BUS DEFINITION

Front Side

No.	SYMBOL	NAME	FUNCTION
46	SNIP	INP	Status output signal which indicates that the address bus contains the address of an input device and the input data should be placed on the data bus when PDBIN is active
47	SMEMR	MEMR	Status output signal which indicates that the data bus will be used for memory read data
48	SHLTA	HLTA	Status output signal which acknowledges a HALT instruction
49	$\overline{\text{CLOCK}}$	CLOCK	2 MHz clock signal
50	GND	GROUND	

Back Side

No.	SYMBOL	NAME	FUNCTION
51	-8V	+8 volts	Unregulated input to 5V regulators
52	-16V	-16 volts	Negative unregulated voltage

BUS DEFINITION

Back Side

No.	SYMBOL	NAME	FUNCTION
53	$\overline{\text{SSW DSB}}$	SENSE SWITCH DISABLE	Disables the data input buffers so the input from the sense switches may be strobed onto the bidirectional data bus
54	$\overline{\text{EXT CLR}}$	EXTERNAL CLEAR	Clear signal for I/O devices (front panel switch closure to ground)
55	CGND	CHASSIS GROUND	
56-57	UNUSED		
68	MWRT	MEMORY WRITE	From the Front Panel replacement card indicates that the current data on the Data Out Bus is to be written into the memory location currently on the address bus
69	$\overline{\text{PS}}$	PROTECT STATUS	Reserved to indicate the status of the memory protect flip-flop on the memory board currently addressed
70	PROT	PROTECT	Reserved for input to the memory protect flip-flop on the memory board currently addressed

BUS DEFINITION

Back Side

No.	SYMBOL	NAME	FUNCTION
71	RUN	RUN	Indicates that the RUN/STOP flip-flop is set to run on the front panel
72	PRDY	READY	Processor command/control input that controls the run state of the processor; if the line is pulled low the processor will enter a wait state until the line is released
73	$\overline{\text{PINT}}$	INTERRUPT REQUEST	The processor recognizes an interrupt request on this line at the end of the current instruction or while halted. If the processor is in the HOLD state or the Interrupt Enable flip-flop is reset, it will not honor the request
74	$\overline{\text{PHOLD}}$	HOLD	Processor command input signal which requests the processor to enter the HOLD state; allows an external device to gain control of address and data buses as soon as the processor

BUS DEFINITION

Back Side

No.	SYMBOL	NAME	FUNCTION
74(continued)			has completed its use of these buses for the current machine cycle
75	$\overline{\text{PRESET}}$	RESET	Processor command input; while activated the content of the program counter is cleared and the instruction register is set to 0
76	PSYNC	SYNC	Processor control output provides a signal to indicate the beginning of each machine cycle
77	$\overline{\text{PWR}}$	WRITE	Processor control output used for memory write or I/O output control; data on the data bus is stable while the $\overline{\text{PWR}}$ is active
78	PDBIN	DATA BUS IN	Processor control output signal indicates to external circuits that the data bus is in the input mode
79	A0	Address Line #0	
80	A1	Address Line #1	

BUS DEFINITION

Back Side

No.	SYMBOL	NAME	FUNCTION
81	A2	Address Line #2	
82	A6	Address Line #6	
83	A7	Address Line #7	
84	A8	Address Line #8	
85	A13	Address Line #13	
86	A14	Address Line #14	
87	A11	Address Line #11	
88	DO2	Data Out Line #2	
89	DO3	Data Out Line #3	
90	DO7	Data Out Line #7	
91	DI4	Data In Line #4	
92	DI5	Data In Line #5	
93	DI6	Data In Line #6	
94	DI7	Data In Line #1	
95	DI0	Data In Line #0	
96	SINTA	INTA	Status output signal to acknowledge signal for INTERRUPT request
97	SWO	WO	Status output signal indicates that the operation in the current machine cycle will be a WRITE memory or output function

BUS DEFINITION

Back Side

No.	SYMBOL	NAME	FUNCTION
98	SSTACK	STACK	Status output signal indicates that the address bus holds the pushdown stack address from the Stack Pointer
99	$\overline{POC}$	Power-On Clear	
100	GND	GROUND	

3.2.12 CPU Power Supply - Functional Description

The power supply is designed to be used in the CPU with boards that have on-board regulators. It provides +10 volts and ±18 volts at no load and approximately +7 and ±15.8 volts at full load.

The rectifiers and 115 volt power switching and fusing functions are contained on a small power supply PC board. Large computer-grade filter capacitors are used to provide minimum ripple. A custom-wound transformer is used to provide the maximum amount of power in the available space to match the exact requirements of the 8080 system. One hundred fifteen volt output terminals are available on the board, switched both fused for such functions as the ventilating fan and auxiliary power outlets on the back panel. A position is provided on the rectifier board for a power switch, for use when the front panel is not installed in the system.

The secondary windings of the power transformer are all connected in series to provide a winding with a center tap and one tap on each side of the center tap midway up the secondary out of which three separate full wave center tap rectifier circuits provide the three required voltages.

Two heavy duty rectifiers are connected to the mid points of the winding and provide +10 volts at the output. A full wave bridge for both plus and negative voltages is connected to the end points of the secondary winding to provide +18 and -18 volts.

Large computer grade electrolytic capacitors are used to filter the rectifier output before applying the voltages to the 8080 back plane.

Since the 7805 regulators used on the 8080 system boards require 7 volts minimum to regulate at 5 volts, a maximum power supply voltage drop of 3 volts is permissible between the no load voltage of 10 volts and the minimum permissible full load voltage of 7 volts.

Many components contribute significantly to the full load power supply voltage drop due to the heavy currents involved. Among these are the resistance of the transformer winding, the resistance of the hook-up wire used in assembling the power supply, the voltage drop accross the diode, the voltage drop between cycles from the discharge of the filter capacitor, and contact resistance in any of the joints. A similar situation is true for the ±18 volt supplies; however, because of the lower currents involved, this design situation is less stringent.

Additionally, three series diodes were put in both the +18 and -18 volt power supply leads to hold the no load voltage down to an acceptable value for a number of boards which use a simple resistor and zener diode to regulate for +12 volts.

3.3 Control Program

3.3.1 Program Description

The control program for the RWLM is written in the PL/M Language (Programming Language/Microcomputers). This language is block structured and somewhat self-documenting. The complete program listing is contained in Appendix C. Most procedures are explained by the comments in the listing.

3.3.2 The Language

A PL/M program is a sequence of "declarations" and "executable statements."

The declarations allow the programmer to control allocation of storage, define simple textual substitutions (macros), and define procedures. PL/M is a "block structured" language: procedures may contain further declarations which control storage allocation and define other procedures.

The procedure definition facility of PL/M allows modular programming: a program can be divided into sections (e.g., Teletype input, conversion from binary to decimal forms, and printing putput messages). Each of these sections is written as a PL/M procedure.

PL/M handles two kinds of data. Its two basic "data types" are BYTE and ADDRESS. A BYTE variable or constant is one that can be represented as an 8-bit quantity; an ADDRESS variable or constant is a 16-bit or double-byte quantity.

The programmer can DECLARE variable names to represent BYTE or ADDRESS values. One can also declare vectors (or array) of the type of BYTE or ADDRESS.

In general, executable statements specify the computational processes that are to take place. To achieve this, arithmetic, logical (Boolean), and comparison (relational) operators are defined for variables and constants of both types (BYTE and ADDRESS). These operators and operands are combined to form EXPRESSIONS which resemble those of elementary algebra. For example, the PL/M expression

```
X * (Y - 3)/R
```

represents this calculation: the value of X multiplied by the quantity Y-3, divided by the value of R. Expressions are a major component of PL/M statements. A simple statement which computes a result and stores it in a memory location defined by a variable name. The assignment

```
Q = X * (Y - 3)/R
```

first causes the computation to the right of the equal sign, as described above. The result of this computation is then saved in a memory location labeled by the variable name "Q."

Other statements in PL/M perform conditional tests and branching, loop control, and procedure invocation with parameter passing. The flow of program execution is specified by means of control structures that take advantage of the block-structured nature of the language. Input and output statements read and write 8-bit values from and to the 8080 microprocessor's input and output ports. Procedures can be defined which use these basic input and output statements to perform more complicated I/O operations.

3.3.3 Calibration Coefficients

The Rn-daughter and WL coefficients which were calculated during the calibration runs are stored in the READ ONLY memory of the RWLM in matrix form and are accessed by the CAL procedure on page 9 of the listing. Each constant is allocated 16 bytes of memory starting at location 110 H. The format of the data is as follows:

ADDRESS (HEX)

110-11F	WL	COEF. #1	HEAD #1
120-12F	WL	COEF. #2	
130-13F	WL	COEF. #3	
140-14F	RaA	COEF. #1	
150-15F	RaA	COEF. #2	
160-16F	RaA	COEF. #3	
170-17F	RaB	COEF. #1	
180-18F	RaB	COEF. #2	
190-19F	RaB	COEF. #3	
1A0-1AF	RaC	COEF. #1	
1B0-1BF	RaC	COEF. #2	
1C0-1CF	RaC	COEF. #3	

1D0-1DF	WL	COEF.	#1	HEAD #2
1E0-1EF	WL	COEF.	#2	
1F0-1FF	WL	COEF.	#3	
200-20F	RaA	COEF.	#1	
210-21F	RaA	COEF.	#2	
220-22F	RaA	COEF.	#3	
230-23F	RaB	COEF.	#1	
240-24F	RaB	COEF.	#2	
250-25F	RaB	COEF.	#3	
260-26F	RaC	COEF.	#1	
270-27F	RaC	COEF.	#2	
280-28F	RaC	COEF.	#3	

Note from the listing that the last character in each data string is a 24H. The math pack interprets this character as the end of a number.

The arrangement of data in this structure allows the routines that calculate the WL and Rn-daughter levels to be tightly coded and it also allows using the same routines to do the calculation for each head with only the change of an indexing variable.

For example, in the CAL procedure the fixed point numbers in the above matrix must be converted to four-byte floating point numbers to be used in the calculations. The code to perform this conversion is as follows:

```
DO I=0 to 11;
CALL INSTR(.CONSTR + (I*16) + (192 * SELDET),
.FCON + (I*4)0;
END;
```

The INSTR routine is passed to the address of the number it is to convert to floating point as the first parameter:

CONSTR + (I*16) + (192*SELDET)

when I = 0 and SELDET = 0 (HEAD #1).

The first number converted starts at location 110H, i.e., the first WL coefficient for Head #1. If SELDET = 1 as it would if Head #2 were selected, the first number converted would at location 100H, i.e., 110H+0+(192D x 1) = 1D0H

The INSTR routine then stores the converted number at location pointed by .FCON + (I*4). This, of course, forms a second matrix holding the floating point values of the constant data starting at the address of FCON (21BFH).

The altitude correction factor, the ta counts and the background counts are also converted to floating point and stored by the CAL PROCEDURE.

The calculations are then performed on the data and constants as shown in the listing. The output data is stored in a vector called WL for use by the PRINTREPORT Procedure.

3.4 Floating Point Math Pack

The floating point system consists of a set of subroutines designed to perform arithmetic operations on numeric quantities represented in memory.

The software constituting the floating point system is divided into two sections.

Section 1 contains all the arithmetic routines, while section 2 contains routines which are used to convert data between a binary floating point format and a decimal format suitable for entry or display on input/output equipment.

3.4.1 Floating Point Routines

The 8080 binary floating point system consists of a set of subroutines designed to perform operations on numeric quantities represented in a specific notation. Subroutines are provided to perform a variety of arithmetic and related operations.

The subroutines are designed to be stored and executed in READ-ONLY-MEMORY (ROM) and require 64 bytes of RAM. Scratchpad memory is initialized by a utility subroutine which must be executed before other subroutines are executed for the first time (INT routine called by PL/M).

In general, the subroutines have the following characteristics. Subroutines requiring one operand take it from an internal floating point accumulator. Subroutines requiring two operands take one from the accumulator and the other from the memory location indicated by the contents of the H and L registers upon entry. The numeric result of each operation is stored in the accumulator and is returned to the called in the A,B,C, and D registers.

Upon exit from the arithmetic subroutines, the properties of the result are indicated by the settings of the control bits.

Carry Bit = 1	The result exceeds the capacity of the accumulator. The other control bits, the contents of the hardware registers, and the contents of the accumulators are meaningless. This situation is also indicated by a non-zero quantity being stored in a flag word.
Carry Bit = 0	The result is in range. The zero and sign bits are properly set, and the A,B,C, and D registers contain a representation of the value in the accumulator.
Zero Bit = 1	The result of the operation is zero or a quantity too small to be represented.
Zero Bit = 0	The result is non-zero.
Sign Bit = 0	The result is non-zero.
Sign Bit = 1	The result is negative.
Sign Bit = 0	The result is positive.

Data are represented in a notation which records eight bits of exponent. One bit of sign, and 24 bits of fraction. The largest magnitude that can be represented is approximately $3.6 * 10^{38}$. The smallest non-zero magnitude is approximately $2.7 * 10^{-39}$. The resolution of the notation is approximately $6.2 * 10^{-8}$, i.e., better than seven-decimal digit precision.

Data values are represented in four consecutive memory words which must be in the same bank of memory. The interpretation of these words is shown below.

Word 1	If non-zero, this word contains the exponent plus a bias of 200 octal. The exponent indicates the power of 2 by which the fraction is multiplied to obtain the represented value. If this word is zero the represented value is zero and words 2, 3, and 4 are meaningless.
Word 2, Bit 7	This bit indicates the sign of the value: 0 if positive, 1 if negative.
Word 2, Bits 6-0	These bits plus as sssumed 1 in the bit 7 are the most significant bits of the fraction. The fraction is stored in absolute form (unsigned) with the radix point positioned to the left of bit 7. The value of the fraction is thus less than 1.0 and equal to or greater than 0.5.
Word 3	This word contains the second most significant eight bits of the fraction.
Word 4	This word contains the least significant eight bits of the fraction.

Examples of Data Notation (Octal Notation).

Value	Word 1	Word 2	Word 3	Word 4
0.0	000	xxx	xxx	xxx x = don't care
+1.0	201	000	000	000
-1.0	201	200	000	000
+0.1	175	114	314	314
-100.1	207	310	063	063

3.4.2 Floating Point Accumulator

The floating point accumulator consists of 5 scratchpad words containing, respectively, the accumulator exponent, the accumulator sign, and three words of accumulator fraction. The exponent is recorded with a bias of 200 octal. An exponent word of zero indicates that the value in the accumulator is zero and the remaining words of the accumulator are meaningless. The sign word holds 000 if the accumulator is negative, 200 octal if positive. The fraction is recorded as a normalized positive value with the radix point to the left of the most significant bit of the first fraction word.

3.5 Format Conversion Package

The format conversion package of the 8080 binary floating point system contains subroutines for the conversion of data between the floating point system notation and two other formats. The non-floating, point formats are four-word fixed-point format and variable-length character-string format.

The format conversion package is contained in 512 consecutive words of memory (2 banks of ROM) and requires for its execution that the arithmetic and utility packages be available in memory. The combination of this format conversion package and the arithmetic and utility packages uses the first 64 words of a bank of RAM as scratchpad memory.

The fixed point format data processes by this package consist of 32-bit binary numbers occupying four words. (Two's complement notation is used to represent negative values.) The position of the binary point relative to the bits representing the value is denoted by a binary scaling factor. The binary scaling factor is not normally recorded in the computer; but, when a format conversion subroutine is called, the binary scaling factor must be specified (in the E register). A binary scaling factor of zero indicates the binary point is immediately to the left of the most significant of the 32 bits representing the value. A binary scaling factor of 32 indicates the binary point is immediately to the right of the least significant bit. The permissible range of the binary scaling factor is -128 (200 octal) to +127 (177 octal).

The character string format data processed by this package consist of binary representations of characters occupying consecutive words of memory. A character string may not cross a memory-bank boundary. The characters which may be included in a character string, and the corresponding octal representations are listed below.

Decimal Digits	00B-011B BCD digits*
Space	360B
+	373B
-	375B
.	376
Exponential sign	025B

*(These octal representations can be converted to the corresponding ASCII characters by adding 060B to each.)

The out subroutine generates character strings in two formats. Each consists of 13 characters. The format used in a specific case is dependent upon the magnitude of the value represented.

Zero and magnitudes between .10000000 and 9999999. are represented by a space or minus sign, seven decimal digits and an appropriately positioned decimal point, and four spaces.

Magnitudes outside the above range are represented by a space or minus sign, a value between 1.000000 and 9.999999, an exponential sign, and a signed two-digit power of ten.

The input subroutine converts character strings in either of the above formats, or a modified version of them. The leading sign character may be included or omitted. Up to 37 digits may be used to indicate the value, with or without an included decimal point. If a power-of-ten multiplier is indicated it may be signed or unsigned and may contain one or two digits. An input character string is terminated by the first character which departs from the specified format.

3.6 Interface Between PL/M and Assembly Language Floating Point Math Pack Routines

There are two sets of routines tha' interface the PL/M and the assembly language routines. One set of these interfaces appears on page 8 of the PL/M control program listing in Appendix C. The assembly language part of the routine is labeled Math Pack Transfer Routine and is also found in Appendix C.

The PL/M procedures are called by elements in the control program. In turn, they transfer control to the assembly language transfer control to the assembly language transfer routines. These routines move parameters passed by PL/M and make these parameters correspond with the requirements of the Math Pack.

4.0 CALIBRATION

It is evident from (1) in section 2.0 that the numerical values for N_A, N_B, and N_C depend on V, E_1, E_2, E_3, and the corrected counts TA, TC' see (5) and the measured counts (B + C). Therefore, to calibrate the RWLM, V, E_1, E_2, and E_3 (the flowrate and the efficiencies of RaA, RaB, and RaC) have to be determined, and one has to ensure that the counts used to calculate N_A, N_B, and N_C are TA, (B + C), and TC' as defined in section 2.0.

4.1 Measurement of Flowrate V

The flowrate, V (liters/min), can easily be measured with a Wet Test Meter (Precision Scientific Co.). The numerical value of V (in liters/min) for the particular detector measured is used in the calibration procedure (see Fig. 8).

4.2 Determinations of E_1, E_2, and E_3

The alpha efficiency of the RWLM detectors, E_1, was determined by comparison with a calibrated hemispherical proportional counter. The proportional counter was calibrated with an NBS ^{241}Am source. An air sample from a plastic drum containing uranium ore was taken. About 40 minutes after the ending of the sampling period the circular portion of the filter strip containing the filtered Rn daughters was cut out. This circular disc was then counted with the proportional counter and with the RWLM. The counting period was always 30 seconds. About 10 measurements with each counter were taken. These results were averaged and E_1 was calculated according to the formula:

$$E_1 = (C_{RWLM}/C_{PC})E_{PC}$$

where:

C_{RWLM} = average number of alpha counts observed with RWLM

C_{PC} = average number of alpha counts observed with the proportional counter.

E_{PC} = known alpha efficiency of the proportional counter.

The measurements with the two counters should be interlaced. The first measurement with the RWLM, for example, should be followed by the first measurement with the proportional counter, and so on. Counting in this interlaced fashion avoids bias due to decay. The counts from the RWLM detection system can be recorded with an external scaler.

Once E_1 is know, E_2 and E_3 can be determined by the following procedure: Knowledge of E_1 allows a complete analysis of any air sample by a method using total alpha counts. This method allows calculation of N_A, N_B, and N_C for this particular air sample. N_A, N_B, and N_C can then be used to calculate the beta disintegrations from RaB and RaC. By comparing the calculated beta disintegrations with the observed total beta counts, the values for E_2 and E_3 are obtained. The necessary calculations are all performed by the "RWLM CALIBRATION PROGRAM" (see Fig. 8).

To calculate the beta efficiencies E_2 and E_3, the Rn-daughter concentrations (atoms/liter) N_A, N_B, and N_C must first be determined from the alpha counts during several counting periods using the following equations:

$$N_A = 0.962617*A_5/E_1*V$$

$$N_B = (-0.919056*A_5 - 11.17412*C_5 + 2.764731*C_3)/E_1*V \quad (2)$$

$$N_C = (0.054135*A_5 + 4.305161*C_5 - 0.264945*C_3)/E_1*V$$

where: *

A_5 = RaA counts observed during 5 min starting 13 sec after the end of the 2-min sampling time.

C_5 = RaC' counts observed during the same time interval as above.

C_3 = RaC' counts observed during 30 min, starting at the same time as above.

The numerical coefficients in (2) are again derived from the laws of radioactive-series decay. This derivation is straightforward but lengthy and will therefore be omitted.

With N_A, N_B, and N_C known, E_2 and E_3 can be determined from the following equations:

$$Q_1 = (0.131911*N_A + 0.235122*N_B)*V*E_2 + (0.010901*N_A + 0.029629*N_B + 0.309182*N_C*V*E_3 \quad (3)$$

$$Q_2 = (0.984198*N_A + 1.046109*N_B)*V*E_2 + (0.389388*N_A + 0.481449*N_B + 1.24961*N_C)*V*E_3$$

where

Q_1 = total beta counts observed during 5 min starting 13 sec after the end of the 2-min sampling time.

Q_2 = total beta counts observed during 30 min same time as above.

---

*The overlap corrections must be incorporated into the alpha counts (A_5, C_5 and C_3). See (5).

With the beta efficiencies E_2 and E_3 so determined, (1) can be inverted and properly scaled. When properly scaled this yields a set of equations which gives the Rn-daughter concentrations in pCi/liter. The WL can also be expressed as a linear combination of A, (B + C), and C'. These counts are net counts, i.e., the background has been subtracted. The equations are of the following form:

$$\begin{aligned} WL &= C_1(A) + C_2(B + C) + C_3(C') \\ RaA(pCi/liter) &= C_4(A) + C_5(B + C) + C_6(C') \\ RaB(pCi/liter) &= C_7(A) + C_8(B + C) + C_9(C') \\ RaC(pCi/liter) &= C_{10}(A) + C_{11}(B + C) + C_{12}(C') \end{aligned} \qquad (4)$$

C_1 through C_{12} are the derived weighting coefficients which are atored in the memory of the RWLM. It is clear from this description of the calibration that N_A, N_B, and N_C are treated as independent unknowns (i.e., no a priori relationship between these quantities other than radioactive series decay is assumed). The RWLM determines, therefore, the Rn-daughter concentrations and WL without any assumptions about the Rn-daughter equilibrium. Since all weighting coefficients are strictly proportional to the inverse of the flowrate, or 1/V, a flowrate correction can be made by including a multiplication factor in (4). This implies that a recalibration of the RWLM is unnecessary if it is operated at different altitudes, or any condition which may result in a flowrate different from that used in the calibration, such as a loss of pump efficiency due to wear.

4.3 Discriminator Adjustments

V, E_1, E_2, and E_3 have been determined so far. We will now discuss the adjustments necessary to ensure that the other input parameters VO (overlap), A, (B + C) and C' are properly determined.

VO = fraction of RaC' counts appearing in the RaA channel during the corresponding counting period.

A = total alpha counts in RaA channel during 2 min counting period starting 13 sec after the end of sampling.

(B+C) = total beta counts from RaB and RaC (same period of accumulation as above).

C = total alpha counts in RaC' channel (same as above).

CKUS = total alpha counts from 39 to 41 min used to calculate WLKUS, the Kusnetz WL (with time base starting 13 sec after the end of sampling).

First, the lower alpha discriminator level has to be set above the noise level. This is done easily by gradually increasing the reference voltage level until no background counts due to noise are observed. This adjustment has to be done in an area with negligible airborne radioactivity. In a similar manner the beta channel must be adjusted. A certain number of counts due to ambient gamma radiation will always appear since the beta detector is sensitive to gamma rays. The RWLM channel accumulator registers approximately 250 counts/min for 1 mr/hr of gamma radiation from a ^{226}Ra source. The next step is the adjustment of the upper alpha discriminator level. For this adjustment, a multi-channel analyzer (MCA) is needed. The multichannel-analyzer displys shows clearly the separation between the RaA and the RaC' alpha peaks. Integrating the combined spectrum between the analyzer's baseline and the channel with the minimum number of counts between the two alpha peaks gives A, the number of counts in the lower alpha channel. Integrating from this channel to the descent of the RaC' spectrum gives C', the number of counts in the upper alpha channel. The upper level alpha discriminator must then be adjusted to obtain agreement between A and C' by comparing the integrated counts under each peak on the MCA with the A and C' counts observed in the alpha channels of the RWLM. This data is collected simultaneously from the same air sample. This agreement can be achieved by gradually adjusting the upper alpha discriminator level in the RWLM and comparing the different sets of A counts and C' counts. An initial minimum setting of the RWLM's upper alpha discriminator can be achieved with a ^{241}Am standard. This initial setting should be slightly higher than the pulses from the 5.5 MeV ^{241}alpha's. After the upper alpha discriminator has been properly set, the RWLM will record the quantities A and C' properly. The A counts and C' counts are, however, not yet suitable for the necessary calculations. The A and C' have to be corrected for the degradation of the RaC' spectrum.

VO (overlap) is the fraction of the RaC' counts that is detected by the lower alpha channel when a sample is made with the counting delayed 45 min to assure complete decay of RaA. It is algebraically defined by:

VO = (RaC' counts in RaA channel)/(RaC' counts in C channel).

With VO known, all TA and TC counts can be determined from:

$$TA = A - ((VO) \cdot C') \quad (5)$$

$$TC' = (1 + VO) \cdot C'$$

These relationships (5) are used in the "RWLM CALIBRATION PROGRAM."

4.4 Calibration Program (see Fig. 8)

The calibration program automates the solution of the equations for radioactive-series decay, as applied to the RWLM, and calculates the required efficiencies and coefficients. This program is available in both the FORTRAN and BASIC languages. The program performs the following functions:

1) It calculates the Rn-daughter concentrations by three different methods and the WL by four different methods from the same air sample.

   The four different methods are:

   a) The alpha-spectroscopic method

   b) The total-alpha method

   c) The Kusnetz method

   d) The IWLM/RWLM method

2) It calculates the RaB and RaC efficiencies using the Rn-daughter concentrations calculated by the total-alpha method, and the beta counts at two different periods of time.

3) It calculates the 12 coefficients needed for the IWLM/RWLM method. See (4).

4) It calculates the Rn-daughter concentrations from 9 of these calculated coefficients, in units of atoms/liter, for

comparison to the other two methods, and also in units of pCi/liter which is the unit used by the RWLM.

5) It calculates the WL from the Rn-daughter concentrations, and with the use of the three remaining coefficients it also determines the WL directly from the measured counts.

In order to understand the program, it will be necessary to refer to the calibration program and to the "Table of Definitions of Symbols in the calibration program" (see Appendix C and Fig. 8). The program has two branches, one which calculates beta efficiencies from the input data, and another which uses the mean value of the beta efficiencies to derive the final weighted coefficients (see Figs. 9 and 10). In both branches the Rn-daughter concentrations are calculated by using the alpha-spectroscopic method (line 140) and the total-alpha method (line 150).

Lines 50 through 52 are the coefficients for these two equations. Lines 100 through 120 are statements that adapt the input data for these methods. Line 170 calculates the WL with the alpha-spectroscopic method. Line 180 uses the total-alpha method for the same calculation. Line 190 calculates the WL by the Kusnetz method. These three WL calculations are available for comparison with the IWLM/RWLM method to be explained later.

As shown in Fig. 9, the first question asked by the computer is "Calculate or Input E_B, E_C" (E_1, E_2). When the operator types a "C", the program is routed into Branch 1 (lines 2000 to 2100) which uses the inverted equations calculate the beta efficiencies for RaB and RaC (E_2 and E_3, respectively) by using the observed beta counts (Q_1 and Q_2) and the Rn-daughter concentrations previously determined by the total-alpha method. If an "I" is entered (see Fig. 10), the input branch is selected and the beta efficiencies are entered as data (line 210).

Statements 3000 to 3210 perform a matrix inversion and give the final weighting coefficients C_1 through C_{12} for (4) after conversion to the proper units.

The Rn-daughter concentrations are calculated in units of atoms/liter by the statements for FO(1), FO(2), and FO(3) and are printed out to allow comparison with the values calculated by the alpha methods. FO(1), FO(2), and FO(3) are then converted to units of pCi/liter (line 260) and printed. The WL is calculated from the Rn-daughter concentrations which were derived using the IWLM/RWLM method (line 270) and directly using the coefficients of the IWLM/RWLM method (line 280). Note that these two different equations give exactly the same results. See Fig. 9, "WL from IWLM (direct and from Rn daughters)."

4.5 Calibration Procedure

The calibration procedure consists of the following steps: first, 10 sets of data are automatically collected; second, the RWLM calibration program is used to calculate the beta efficiencies; third, the RWLM calibration program is used to calculate the coefficients for the RWLM using the mean value of the calculated beta efficiencies; and, fourth, these new coefficients

```
*G
TODAY IS 00/00/00  THE TIME IS 00:00
DATE CORRECT (Y/N)?N
ENTER YEAR-77 ENTER MONTH-03 ENTER DAY -21 ENTER HOUR-11 ENTER MINUTE-45

TODAY IS 03/21/77  THE TIME IS 11:45
DO YOU WANT  DETECTOR  1 RUN(Y/N)?Y
DO YOU WANT  DETECTOR  2 RUN(Y/N)?Y

ENTER TIME INTERVAL BETWEEN SAMPLES HOURS-00 MIN-20
ENTER TIME TO START HOUR-12  MIN-00
INPUT TIME TO STOP
ENTER YEAR-77 ENTER MONTH-03 ENTER DAY -22 ENTER HOUR-00 ENTER MINUTE-00
```

Figure 7 Data Terminal Dialog between Operator and System

```
1 REM        RWLM CALIBRATION PROGRAM
2 REM     ARGONNE NATIONAL LABORATORY ELECTRONICS DIVISION
20 PRINT "RWLMCAL.BAS   V02.1": GOSUB 4000
30 FOR I=0 TO 8
35 READ C(I),CO(I)
40 NEXT I
50 DATA .926838,1.918637,0,-3.196779,0,2.506266
51 DATA -.879403,1.701921,-11.12606,-18.620853,2.752840,16.756924
52 DATA .049957,-3.6592397,4.232080,12.896776,-.251541,-10.308247
100 S5=A5+C5:S3=A3+C3:S2=A2+C2
110 T1=A5-V0*C5:T2=(1+V0)*C5:T3=(1+V0)*C3:T4=A-V0*C
120 T5=(1+V0)*C
130 FOR O=0 TO 6 STEP 3
140 F1(O/3)=(C(O)*T1+C(O+1)*T2+C(O+2)*T3)/(E1*V)
150 F2(O/3)=(CO(O)*S5+CO(O+1)*S3+CO(O+2)*S2)/(E1*V)
160 NEXT O
170 W1=(13.68*F1(0)+7.68*(F1(1)+F1(2)))/130000
180 W2=(13.68*F2(0)+7.68*(F2(1)+F2(2)))/130000
190 W3=X1/(E1*2*V*150)
200 IF S0=1 THEN GOSUB 2000:GOTO 250
210 INPUT "ER,EC PLEASE";E2,E3
250 GOSUB 3000
260 P1=.227261*FO(1)/2.22:P2=.025864*FO(2)/2.22:P3=.035185*FO(3)/2.22
270 P4=(13.68*FO(1)+7.68*(FO(2)+FO(3)))/130000
280 F9=W(1)*A+W(2)*B3+W(3)*C
299 PRINT:PRINT:PRINT:PRINT
300 PRINT "               R W L M   C A L I B R A T I O N ":PRINT
301 PRINT:PRINT:PRINT A$:PRINT D$,P$
302 PRINT:PRINT "FNA=";F1(0),,"FNB=";F1(1),,"FNC=";F1(2):PRINT
305 PRINT:PRINT "FNA2=";F2(0),,"FNB2=";F2(1),"FNC2="F2(2)
307 PRINT:PRINT"WL1=";W1,"WLKUS=";W3,"ER=";E2,"EC="E3
310 PRINT:PRINT"WL2=";W2
312 PRINT:PRINT"RN-DAUGHTER COEFFICIENTS FOR IWL-MEMORY"
320 FOR I=1 TO 3: FOR J=1 TO 3
330 PRINTP(I,J),:NEXTJ:PRINT:NEXT I
335 PRINT:PRINT"RN-DAUGHTER (RWLM) IN ATOMS"
336 PRINT FO(1),FO(2),FO(3)
340 PRINT:PRINT"RN-DAUGHTER (RWLM) IN PCI/L"
345 PRINT P1,P2,P3
350 PRINT:PRINT:PRINT"WL FROM IWLM (DIRECT AND FROM RN-DAUGHTERS)"
355 PRINT P4,F9
360 PRINT:PRINT:PRINT"WL COFFICIENTS FOR IWLM-MEMORY"
370 PRINT W(1),W(2),W(3)
490 PRINT:PRINT:PRINT:PRINT:PRINT
500 PRINT"NORMAL STOP AT LINE 500":STOP
2000 X1=(.127907*F2(0)+.236138*F2(1))*V
2010 X2=(.385895*F2(0)+.478116*F2(1)+1.256959*F2(2))*V
2020 X3=(.981722*F2(0)+1.050628*F2(1))*V
2030 X4=(.010200*F2(0)+.028418*F2(1)+.311*F2(2))*V
2040 O9=X1*X2-X3*X4
2050 Y1=(.385895*F2(0)+.478116*F2(1)+1.256959*F2(2))*V*O1
2060 Y2=(.0102*F2(0)+.028418*F2(1)+.311*F2(2))*V*O2
2070 O8=Y1-Y2:E2=O8/O9
2080 Z1=(.127907*F2(0)+.236138*F2(1))*V*O2
2090 Z2=(.981722*F2(0)+1.050628*F2(1))*V*O1
2100 E3=(Z1-Z2)/O9
2110 RETURN
3000 A1(1)=.580386*E1*V:A2(1)=.036204*E2*V+.001584*E3*V
3010 A2(2)=.098134*E2*V+.006941*E3*V:A2(3)=.131*E3*V:A3(1)=.001584*E1*V
```

Figure 8 RWLM Calibration Program

```
3020 A3(2)=.006941*E1*V:A3(3)=.131*E1*V
3030 B4=A1(1)*(A2(2)*A3(3)-A3(2)*A2(3))
3040 C1(1)=(A2(2)*A3(3)-A2(3)*A3(2))/B4
3050 C2(1)=(A2(3)*A3(1)-A2(1)*A3(3))/B4
3060 C2(2)=(A1(1)*A3(3))/B4
3065 C2(3)=-(A1(1)*A2(3))/B4
3070 C3(1)=(A2(1)*A3(2)-A2(2)*A3(1))/B4
3080 C3(2)=-(A1(1)*A3(2))/B4
3090 C3(3)=(A1(1)*A2(2))/B4
3100 P(1,1)=C1(1)*.227261/2.22
3105 P(1,2)=0
3110 P(1,3)=-P(1,1)*V0:F0(1)=C1(1)*T4
3120 F0(2)=C2(1)*T4+C2(2)*B3+C2(3)*T5
3130 P(2,1)=C2(1)*.025864/2.22:P(2,2)=C2(2)*.025864/2.22
3140 P(2,3)=(C2(3)*(1+V0)-C2(1)*V0)*.025864/2.22
3150 F0(3)=C3(1)*T4+C3(2)*B3+C3(3)*T5
3160 P(3,1)=C3(1)*.035185/2.22:P(3,2)=C3(2)*.035185/2.22
3170 P(3,3)=(C3(3)*(1+V0)-C3(1)*V0)*.035185/2.22
3180 W(1)=(13.68*C1(1)+7.68*(C2(1)+C3(1)))/130000
3190 W(2)=7.68*(C2(2)+C3(2))/130000
3200 W(3)=(7.68*(1+V0)*(C2(3)+C3(3)))/130000-V0*W(1)
3210 RETURN
4000 INPUT "CALCULATE OR INPUT EB,EC (C/I)"; B$
4001 A$="DATE              PLACE"
4010 IF NOT(B$="I" OR B$="C") THEN 4000
4020 IF B$="I" THEN  S0=0
4021 IF B$="C" THEN S0=1
4030 INPUT "DATE,PLACE" ; D$,P$
4040 INPUT "FLOWRATE (LITERS/MIN)";V
4050 INPUT "EFFICIENCY OF ALPHA DETECTOR"; E1
4055 INPUT "OVLAP"; V0
4057 INPUT "TOTAL BETA COUNTS FROM RAB AND RAC DURING 5 MINUTES";B1
4060 INPUT "TOTAL BETA COUNTS FROM RAB AND RAC DURING 30 MINUTES";B2
4070 INPUT "TOTAL ALPHA COUNTS IN RAA CHANNEL DURING 30 MINUTES";A3
4080 INPUT "TOTAL ALPHA COUNTS IN RAA CHANNEL DURING 35 MINUTES";A2
4090 INPUT "TOTAL ALPHA COUNTS IN RAC' CHANNEL DURING 35 MINUTES";C2
4100 INPUT "TOTAL ALPHA COUNTS IN RAA CHANNEL DURING 5 MINUTES";A5
4110 INPUT "TOTAL ALPHA COUNTS IN RAC' CHANNEL DURING 5 MINUTES";C5
4120 INPUT "TOTAL ALPHA COUNTS IN RAC' CHANNEL DURING 30 MINUTES"; C3
4130 INPUT "TOTAL ALPHA COUNTS IN RAA CHANNEL DURING 2 MINUTES";A
4140 INPUT "TOTAL BETA COUNTS FROM RAB AND RAC DURING 2 MINUTES"; B1
4150 INPUT "TOTAL ALPHA COUNTS IN RAC' CHANNEL DURING 2 MINUTES";C
4160 INPUT "TOTAL ALPHA COUNTS FROM 39-41 MINUTES";K1
4999 RETURN
5000 END
OK
```

Figure 8. (Continued)

```
RWLMCAL.BAS   V02.1
CALCULATE ØR INPUT EB,EC (C/I)? C
DATE,PLACE? 8/4/76,      QUIRK 1
FLØWRATE (LITERS/MIN)? 11.2
EFFICIENCY ØF ALPHA DETECTØR? .20
OVLAP? .18
TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 5 MINUTES? 56416.5
TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 30 MINUTES? 294583
TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 30 MINUTES? 25419
TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 35 MINUTES? 27405.5
TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 35 MINUTES? 107085
TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 5 MINUTES? 10151.5
TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 5 MINUTES? 16980
TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 30 MINUTES? 94182
TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 2 MINUTES? 5157
TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 2 MINUTES? 22898
TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 2 MINUTES? 6993
TØTAL ALPHA CØUNTS FRØM 39-41 MINUTES? 2685


            R W L M   C A L I B R A T I Ø N


DATE          PLACE
8/4/76        QUIRK 1

FNA= 2935.72                 FNB= 34272.6                 FNC= 25533.5


FNA2= 3029.67                              FNB2= 32477.7
FNC2= 25368.5

WL1= 3.84209  WLKUS= 3.99554               EB= .191372   EC= .395248

WL2= 3.73619

RN-DAUGHTER CØEFFICIENTS FØR IWL-MEMØRY
 .0787421      0             -.0141736
-3.30609E-03   .0553894      -.128571
 9.08931E-05  -3.99245E-03    .0730273

RN-DAUGHTER (RWLM) IN ATØMS
 2998.51       30227.2       26482.9

RN-DAUGHTER (RWLM) IN PCI/L
 306.957       352.161       419.73


WL FRØM IWLM (DIRECT AND FRØM RN-DAUGHTERS)
 3.66579       3.66579


WL CØFFICIENTS FØR IWLM-MEMØRY
 6.45171E-05   2.65986E-04  -3.94318E-04




NØRMAL STØP AT LINE 500

BREAK IN 500
ØK
```

Figure 9 RWLM Calibration Program - Readout with calculate (C) beta efficiency branch

```
RWLMCAL.BAS   V02.1
CALCULATE ØR INPUT EB,EC (C/I)? I
DATE,PLACE? 8/4/76,       QUIRK 1
FLØWRATE (LITERS/MIN)? 11.2
EFFICIENCY ØF ALPHA DETECTØR? .20
OVLAP? .18
TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 5 MINUTES? 56416.5
TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 30 MINUTES? 294583
TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 30 MINUTES? 25419
TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 35 MINUTES? 27405.5
TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 35 MINUTES? 107085
TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 5 MINUTES? 10151.5
TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 5 MINUTES? 16980
TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 30 MINUTES? 94182
TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 2 MINUTES? 5137
TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 2 MINUTES? 22898
TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 2 MINUTES? 6993
TØTAL ALPHA CØUNTS FRØM 39-41 MINUTES? 2685
EB,EC PLEASE? .205, .3865


            R W L M   C A L I B R A T I Ø N


DATE          PLACE
8/4/76        QUIRK 1

FNA= 2935.72              FNB= 34272.6              FNC= 25533.5


FNA2= 3029.67                          FNB2= 32477.7
FNC2= 25368.5

WL1= 3.84209  WLKUS= 3.99554           EB= .205      EC= .3865

WL2= 3.73619

RN-DAUGHTER CØEFFICIENTS FØR IWL-MEMØRY
 .0787421       0             -.0141736
-3.30609E-03    .0517072      -.117316
 9.08931E-05   -3.72704E-03    .072216

RN-DAUGHTER (RWLM) IN ATØMS
 2998.51        29746.1        26508.4

RN-DAUGHTER (RWLM) IN PCI/L
 306.957        346.555        420.134


WL FRØM IWLM (DIRECT AND FRØM RN-DAUGHTERS)
 3.63888        3.63888


WL CØFFICIENTS FØR IWLM-MEMØRY
 6.45171E-05    2.48304E-04   -3.40269E-04



NØRMAL STØP AT LINE 500

BREAK IN 500
ØK
```

Figure 10 RWLM Calibration Program - Readout with input (I) beta efficiency branch

*** R W L M ***
DETECTØR 1

TØDAY IS 08/04/76 THE TIME IS 16:45
ALTITUDE CØRRECTIØN FACTØR = 1

WØRKING LEVEL	3.66579	
RADIUM A	306.957	PC/L
RADIUM B	352.161	PC/L
RADIUM C	419.73	PC/L

	SAMPLE CØUNTS	BACKGRØUND CØUNTS	NET CØUNTS
RADIUM A	5184	27	5157.0
RADIUM (B+C)	23569	671	22898.0
RADIUM C'	7015	22	6993.0

05 MINUTE CØUNT

	SAMPLE CØUNTS	BACKGRØUND CØUNTS	NET CØUNTS
RADIUM A	10219	27	10151.5
RADIUM (B+C)	58094	671	56416.5
RADIUM C'	17035	22	16980.0

30 MINUTE CØUNT

	SAMPLE CØUNTS	BACKGRØUND CØUNTS	NET CØUNTS
RADIUM A	25824	27	25419.0
RADIUM (B+C)	304648	671	294583.0
RADIUM C'	94512	22	94182.0

35 MINUTE CØUNT

	SAMPLE CØUNTS	BACKGRØUND CØUNTS	NET CØUNTS
RADIUM A	27878	27	27405.5
RADIUM (B+C)	344188	671	332445.5
RADIUM C'	107470	22	107085.0

39-41 MINUTE CØUNT

	SAMPLE CØUNTS	BACKGRØUND CØUNTS	NET CØUNTS
RADIUM A	762	27	735.0
RADIUM (B+C)	13850	671	13179.0
RADIUM C'	4657	22	4635.0

Figure 11 Sample Calibration Data Readout

are loaded into the RWLM memory. This procedure assumes that the alpha efficiency, the flowrate, and the RaC' overlap were previously determined. The first step is initiated by entering a "C" command via the keyboard. The RWLM then proceeds automatically to complete 10 successive calibration sampling runs (see Fig. 11).

When the calibration mode is selected by typing a "C" command, the following events occur:

1) a normal sampling run is taken, but the accumulators are not reset at the end of the 2-min counting time.*
2) the total count for 5 min is recorded.
3) the total count for 30 min is recorded.
4) the total count for 35 min is recorded.
5) the accumulators are reset and the total count from 39 to 41 min is recorded.

This procedure is automatically repeated a total of 10 times for each head in order to obtain a statistically sufficient number of runs. It is possible to acquire the data for a complete calibration run in approximately 14 hours. A calibration run can be completed overnight. Unlike the earlier IWLM which had no microprocessor, the RWLM does not require an operator to be present to record the data. A sample calibration run is shown in Fig. 11.

On completion of the data collection, the RWLM calibration program is run, using the collected data to calculate beta efficiencies E_1

*The zero time-base is selected as 13 sec after the end of the sampling period.

and E_2. The beta efficiencies calculated by the program are averaged to obtain the mean beta efficiencies.

The program is then rerun in the input branch (see Fig. 10) to calculate the final coefficients C_1 through C_{12} for the RWLM equations.

These calculated coefficients C_1 through C_{12} are then programmed into the RWLM PROM memory and the calibration is complete. Note that since the RWLM has two detection heads, two individual calibrations are required for the total system.

5.0 TEST RESULTS

The measurements of WL and Rn-daughter concentrations taken with the RWLM in the Twilight Mine on November 16-18, 1976, are tabulated in Table 5.1. The agreement of the WL, N_A, N_B, and N_C values obtained during the calibration runs is very good. WL is compared with three different and independent methods of measurement. One sample is used to obtain data for all the different methods of measurement. The WL's as measured by the total-alpha method, alpha-spectroscopic method, and the RWLM method show consistently good agreement. Occasionally the WL as measured by Kusnetz method reads a higher value than the other three methods; however, the WL using the Kusnetz method is in good agreement with the other three methods in more than 80% of the measurements. The Rn-daughter concentrations as measured by the RWLM show an absence of bias which can be seen easily by inspection of the tabulated data in Table 5.1.

TABLE 5.1

Data Calculated from Calibration Runs
in Twilight Mine

		WL	RaA	RaB	RaC	Method
			(in units of atoms/liter)			
11/16/76	Det 1					
	Run #1	1.640	2187	15496	8365	alpha spectroscopic
	Time: 1353	1.610	2112	14799	8687	total alpha
		1.686				Kusnetz
		1.645	2163	15330	8664	RWLM
	Det 1					
	Run #2	1.607	2216	15110	8137	alpha spectroscopic
	Time: 1439	1.566	2116	14168	8569	total alpha
		1.623				Kusnetz
		1.547	2215	13993	8243	RWLM
	Det 1					
	Run #3	1.569	2153	14595	8130	alpha spectroscopic
	Time: 1525	1.607	2327	15615	7441	total alpha
		1.669				Kusnetz
		1.606	2128	15402	7998	RWLM
	Det 1					
	Run #4	1.574	2095	15159	7752	alpha spectroscopic
	Time: 1630	1.540	2012	14364	8111	total alpha
		2.139				Kusnetz
		1.584	2125	15395	7638	RWLM
	Det 1					
	Run #5	1.500	2147	13824	7740	alpha spectroscopic
	Time: 1716	1.461	1938	12721	8554	total alpha
		1.635				Kusnetz
		1.461	2125	12944	7996	RWLM
	Det 1					
	Run #6	1.481	2157	13351	7881	alpha spectroscopic
	Time: 1801	1.509	2189	13938	7713	total alpha
		2.075				Kusnetz
		1.527	2163	13940	8057	RWLM
	Det 1					
	Run #7	1.482	2025	14200	7276	alpha spectroscopic
	Time: 1847	1.419	1722	12489	8469	total alpha
		1.512				Kusnetz
		1.475	1971	14215	7235	RWLM
	Det 1					
	Run #8	1.481	2064	13759	7642	alpha spectroscopic
	Time: 1933	1.487	1999	13747	7862	total alpha
		1.591				Kusnetz
		1.487	2090	13915	7540	RWLM
	Det 1					
	Run #9	1.442	2052	13396	7361	alpha spectroscopic
	Time: 2019	1.461	2070	13774	7262	total alpha
		1.523				Kusnetz
		1.482	2070	14125	7270	RWLM

		WL	RaA	RaB	RaC	Method
11/16/76	Det 1					
	Run #10	1.470	2037	13760	7488	alpha spectroscopic
	Time: 2105	1.472	1998	13731	7624	total alpha
		1.593				Kusnetz
		1.460	2051	13471	7597	RWLM
	Det 1					
	Run #11	1.437	2026	13280	7434	alpha spectroscopic
	Time: 2150	1.454	1999	13553	7498	total alpha
		1.537				Kusnetz
		1.474	2040	13916	7404	RWLM
	Det 1					
	Run #12	1.459	2075	13380	7628	alpha spectroscopic
	Time: 2236	1.459	1971	13190	8002	total alpha
		1.598				Kusnetz
		1.510	2101	14447	7379	RWLM
	Det 1					
	Run #13	1.427	2118	13266	7120	alpha spectroscopic
	Time: 2322	1.428	2040	13137	7396	total alpha
		1.484				Kusnetz
		1.463	2121	13854	7130	RWLM
11/17/76	Det 2					
	Run #1	1.761	2762	16266	8628	alpha spectroscopic
	Time: 0008	1.752	2743	16056	8710	total alpha
		2.322				Kusnetz
		1.759	2752	16587	8288	RWLM
	Det 2					
	Run #2	1.810	2800	17298	8359	alpha spectroscopic
	Time: 0053	1.761	2515	15874	9462	total alpha
		1.832				Kusnetz
		1.801	2801	17053	8436	RWLM
	Det 2					
	Run #3	1.793	2805	16725	8633	alpha spectroscopic
	Time: 0139	1.800	2768	16786	8751	total alpha
		1.878				Kusnetz
		1.827	2804	17378	8552	RWLM
	Det 2					
	Run #4	1.790	2881	16404	8764	alpha spectroscopic
	Time: 0225	1.813	2867	16812	8772	total alpha
		1.813				Kusnetz
		1.776	2849	16779	8810	RWLM
	Det 2					
	Run #5	1.788	2827	15973	9250	alpha spectroscopic
	Time: 0311	1.838	2933	17104	8781	total alpha
		1.890				Kusnetz
		1.864	2831	17246	9263	RWLM
	Det 2					
	Run #6	1.852	2903	17142	9042	alpha spectroscopic
	Time: 0356	1.875	2902	17565	9005	total alpha
		2.574				Kusnetz
		1.841	2897	16721	9278	RWLM

		WL	RaA	RaB	RaC	Method
11/17/76	Det 2					
	Run #7	1.858	2903	16689	9593	alpha spectroscopic
	Time: 0442	1.884	2905	17181	9537	total alpha
		1.932				Kusnetz
		1.886	2950	16745	9922	RWLM
	Det 2					
	Run #8	1.844	2858	17030	9097	alpha spectroscopic
	Time: 0528	1.877	2911	17744	8849	total alpha
		2.017				Kusnetz
		1.886	2852	18003	8843	RWLM
	Det 2					
	Run #9	1.888	3053	16553	9967	alpha spectroscopic
	Time: 0614	1.957	3187	18084	9365	total alpha
		2.062				Kusnetz
		1.953	3032	17769	9886	RWLM
	Det 2					
	Run #10	1.916	2894	17874	9409	alpha spectroscopic
	Time: 0700	1.880	2674	16793	10259	total alpha
		1.994				Kusnetz
		1.913	2875	17450	9818	RWLM

Data Calculated from Calibration Runs
in Twilight Mine

Date	Run	WL	RaA	RaB	RaC	Method
			(in units of atoms/liter)			
11/17/76	Det 1 Run #1 Time: 1529	0.333	640	3055	1438	alpha spectroscopic
		0.311	562	2510	1756	total alpha
		0.342				Kusnetz
		0.329	657	2882	1514	RWLM
	Det 1 Run #2 Time: 1615	0.330	650	3027	1402	alpha spectroscopic
		0.323	648	2895	1422	total alpha
		0.325				Kusnetz
		0.318	669	2752	1447	RWLM
	Det 1 Run #3 Time: 1701	0.306	596	2766	1348	alpha spectroscopic
		0.299	587	2633	1388	total alpha
		0.312				Kusnetz
		0.296	611	2519	1405	RWLM
	Det 1 Run #4 Time: 1746	0.293	554	2772	1201	alpha spectroscopic
		0.265	466	2083	1567	total alpha
		0.374				Kusnetz
		0.281	527	2629	1188	RWLM
	Det 1 Run #5 Time: 1832	0.290	532	2727	1237	alpha spectroscopic
		0.279	512	2483	1327	total alpha
		0.289				Kusnetz
		0.279	513	2499	1310	RWLM
	Det 1 Run #6 Time: 1918	0.267	513	2265	1343	alpha spectroscopic
		0.276	557	2509	1173	total alpha
		0.278				Kusnetz
		0.257	505	2036	1406	RWLM
	Det 1 Run #7 Time: 2004	0.253	492	2257	1147	alpha spectroscopic
		0.250	469	2163	1235	total alpha
		0.267				Kusnetz
		0.252	500	2250	1123	RWLM
	Det 1 Run # 8 Time: 2050	0.255	475	2456	1017	alpha spectroscopic
		0.245	441	2211	1155	total alpha
		0.244				Kusnetz
		0.256	477	2545	933	RWLM
	Det 1 Run #9 Time: 2135	0.222	469	1894	1021	alpha spectroscopic
		0.221	464	1864	1042	total alpha
		0.228				Kusnetz
		0.207	464	1597	1074	RWLM

		WL	RaA	RaB	RaC	Method
11/17/76	Det 1					
	Run #10	0.213	431	1889	942	alpha spectroscopic
	Time: 2221	0.205	405	1697	1047	total alpha
		0.223				Kusnetz
		0.230	427	2192	944	RWLM
	Det 2					
	Run #1	1.90	2871	17707	9268	alpha spectroscopic
	Time: 2301	1.90	2882	17821	9218	total alpha
		2.00				Kusnetz
		1.83	2891	16511	9299	RWLM
	Det 2					
	Run #2	1.85	2847	16555	9690	alpha spectroscopic
	Time: 2353	1.82	2772	15806	10018	total alpha
		1.91				Kusnetz
		1.85	2873	16402	9765	RWLM
11/18/76	Det 2					
	Run #3	1.864	2937	16752	9569	alpha spectroscopic
	Time: 0038	1.867	2952	16838	9509	total alpha
		2.017				Kusnetz
		1.869	2919	17028	9406	RWLM
	Det 2					
	Run #4	1.893	2848	17795	9167	alpha spectroscopic
	Time: 0124	1.865	2664	16959	9872	total alpha
		1.993				Kusnetz
		1.811	2857	16199	9367	RWLM
	Det 2					
	Run #5	1.836	2822	16839	9204	alpha spectroscopic
	Time: 0210	1.822	2803	16544	9297	total alpha
		1.913				Kusnetz
		1.835	2823	16695	9339	RWLM
	Det 2					
	Run #6	1.961	2975	18096	9797	alpha spectroscopic
	Time: 0256	1.902	2722	16549	10805	total alpha
		2.603				Kusnetz
		1.921	2963	17350	9890	RWLM
	Det 2					
	Run #7	2.007	2998	18929	9707	alpha spectroscopic
	Time: 0341	1.975	2863	18075	10249	total alpha
		2.089				Kusnetz
		2.006	2943	19097	9618	RWLM
	Det 2					
	Run #8	1.983	3022	18304	9874	alpha spectroscopic
	Time: 0427	1.980	3000	18213	9956	total alpha
		2.670				Kusnetz
		1.879	3011	16436	10012	RWLM
	Det 2					
	Run #9	1.973	2950	18687	9461	alpha spectroscopic
	Time: 0513	1.877	2596	16240	10897	total alpha
		2.048				Kusnetz
		1.929	2986	17934	9391	RWLM
	Det 2					
	Run #10	1.985	2976	18546	9750	alpha spectroscopic
	Time: 0559	1.972	2934	18234	9923	total alpha
		2.123				Kusnetz
		1.946	2937	17748	9955	RWLM

6.0 OPERATING INSTRUCTIONS

6.1 Normal Operation

6.1.1 Load the filter-paper transport mechanism. This is accomplished by removing the alpha-detector hold-down screw and swinging the detector assembly up. The feed and takeup reel covers can then be removed and the reels freed from the instrument by pulling the release plunger for each reel. Note that the reels are identical. The filter paper is then loaded onto the feed reel in such a manner that it is pulled from the top of the reel as it feeds to the sampling port (see Fig. 6). The paper is then placed in the guide slots, under the paper-advance bar, and attached to the takeup reel with a small piece of tape. The takeup reel is then advanced manually several revolutions to assume proper action. If the paper advance bar is not in the highest position, type a "J" on the terminal.

6.1.2 Assure that both the signal cable and the power cable are properly connected to the detector assemblies, and that these cables are also connected to the appropriate channel connector on the CPU.

6.1.3 Apply power to the central processing unit and the TI Silent 700 terminal.

6.1.4 Assure that the AC power in ON in the remote unit by observing the red lamp above the name plate.

6.1.5 If the lamp does not come on the power is applied, the internal switch is in the "off" position. Remove the cover plate and place the switch in the "on" position.

6.1.6 Place the control switch on the terminal to the "ON-LINE" position and press the restart button. The terminal should respond with:

REMOTE WORKING LEVEL MONITOR VERSION 11.0

TODAY IS XX/XX/XX THE TIME IS XX:XX

6.1.7 When first energized, the contents of the data memory locations are unspecified; to reset the system type:

R

The system will respond:

REMOTE WORKING LEVEL MONITOR VERSION 11.0

TODAY IS 00/00/00 THE TIME IS 00:00

6.1.8 To set the date and time, type

D

The system will respond:

ENTER YEAR-XX ENTER MONTH-XX ENTER DAY-XX

ENTER HOUR-XX ENTER MINUTE-XX

The operator must always enter a two-digit number, i.e., January is 01, etc. If a mistake is made in entry, continue typing or press the restart button and re-enter the "D": command. When all data has been entered, the system will respond:

TODAY IS XX/XX/XX THE TIME IS XX:XX

Note that the time function assumes a 24-hour clock. When in the command loop (indicated by an asterisk on the terminal), an "F" can be types to check the date and time.

6.1.9 If an altitude-correction factor other than 1.0 is desired, the operator can now type an "A". The system will respond:

SELECT HEAD 1 OR 2 -

The operator then types 1 or 2.

The system responds:

ENTER ALTITUDE CORRECTION FACTOR=

The operator then types in an appropriate number such as 1.053 or 0.9562. The sequence repeats to allow the operator to set the correction factor on the remaining head and then returns to the command loop.

6.1.10 The system is now ready for normal operation. To enter this mode the operator types a "G".

The system responds:

TODAY IS XX/XX/XX THE TIME IS XX:XX DATE CORRECT (Y/N)?

If date is correct, type "Y"; if not, type "N" and system will enter correction routine. When date is correct, system responds: DO YOU WANT DETECTOR 1 RUN (Y/N)? and waits for operator response.

After response the system types -

DO YOU WANT DETECTOR 2 RUN (Y/N)?

The system responds: (Operator response is indicated as Y or N)

ENTER TIME INTERVAL BETWEEN SAMPLES HOURS-XX MINS-XX

(Time interval must be greater than 14 minutes.)

ENTER TIME TO START HOUR-XX

MIN-XX

INPUT TIME TO STOP

ENTER YEAR-XX ENTER MONTH-XX ENTER DAY-XX

ENTER HOUR-XX ENTER MIN-XX

The system now enters the sampling loop and remains in it until:

a) tie time interval is complete.

b) an error is detected.

c) the reset button on the terminal is depressed.

Fig. 7 shows a complete dialogue between an operator and the system.

6.2 Source Check Mode

6.2.1 In order to provide a first-order check of the system functions, a source check mode has been provided. To use this mode the alpha-detector hold-down screw must be removed and the source holder inserted between the detectors. When it is properly inserted, it cannot be removed without slightly lifting the alpha detector holder assembly.

6.2.2 When the source holder is in place in the desired head or heads, type "S" to enter the Source Check Mode. The system will respond with:

SELECT HEAD 1 OR 2 - 1

SELECT MODE A - B - C - A

The system now proceeds to accumulate a two-minute count in the detector RaA channel. At the end of the counting time it will print a report. All normal calculations will be performed with the count data in the unselected channels set to 0.

This mode allows the operator to assure that the system is operating normally as the calculations for all source check modes should be consistent within statistical variances from day to day.

6.3 Calibration Mode

The calibration mode which has been described in section 4.0 is entered by typing a "C".

7.0 REFERENCES

An Instant Working Level Meter with Automatic Individual Radon Daughter Readout for Uranium Mines,
P. G. Groer, D. J. Keefe, W. P. McDowell, and R. F. Selman,
Proc. 3rd Int. Congress of the International Radiation Protection Association, Part II: 950-956, 1973 (September). National Technical Information Service

Rapid Determination of Radon Daughter Concentrations and Working Level with the Instant Working Level Meter
P. G. Groer, D. J. Keefe, W. P. McDowell, and R. F. Selman,
Int. Symp. on Radiation Protection in Mining and Milling of Uranium and Thorium, Bordeaux, France, 1974 (September).

A Portable Alpha Counter for Uranium Mines with Preset, Updated Readout,
D. J. Keefe, W. P. McDowell, and P. G. Groer,
Int. Symp. on Radiation Protection in Mining and Milling of Uranium and Thorium, Bordeaux, France, 1974 (September).

Interfacing Calculator Chips to Nuclear-Counting Systems,
W. P. McDowell, E. J. Keefe, and P. G. Groer,
IEEE Trans. Nucl. Sci., Vol. NS22,
467-472, 1975 (February).

APPENDIX A

This appendix contains the drawings of all mechanical parts. Copies of these drawings can be obtained from the reproducible master.

TABLE OF CONTENTS
FILTER TRANSPORT ASSEMBLY
MODEL 1A

Part #	Drawing #	Part Name and Description
	G0230-0003-PL-00	Parts List
	G0230-0003-DE-00	Assembly Drawing
1	G0230-0004-DC-00	Top Guide Plate
2	G0240-0005-DC-00	Right Side Plate
3	G0230-0006-DC-00	Left Side Plate
4	G0230-0007-DC-00	Actuating Plate Housing
5	G0230-0008-DC-00	Actuating Plate
6	G0230-0009-DC-00	Spool
7	G0230-0010-DC-00	Gear Housing
8	G0230-0011-DB-00	Gear Housing Shaft
9	G0230-0012-DB-00	Actuating Plate Cam
10	G0230-0013-DB-00	Actuating Plate Drive Rack
11	G0230-0014-DB-00	Actuating Plate Retaining Bar
12	G0230-0015-DB-00	Spool Plunger Spring.
13	G0230-0016-DB-00	Plunger Knob
14	G0230-0017-DB-00	Spool Retaining Plunger
15	G0230-0018-DB-00	Retaining Plunger Housing
16	G0230-0019-DB-00	Spacer Washer
17	G0230-0020-DB-00	Spool Shaft
18	G0230-0021-DB-00	Spool Bearing Housing
19	G0230-0022-DB-00	Spacer
20	G0230-0023-DB-00	Idler Gear Shaft
21	G0230-0024-DB-00	Drive Gear Bearing Housing
22	G0230-0025-DB-00	Drive Gear Shaft
23	G0230-0026-DB-00	Lower Tie Bar
24	G0230-0027-DB-00	Roller
25	G0230-0028-DB-00	Roller Shaft
26	G0230-0029-DB-00	Actuating Plate Roller Shaft
27	G0230-0030-DB-00	Hinge Pin
28	G0230-0031-DB-00	Mounting Plate Hinge Piece
29	G0230-0032-DB-00	Mounting Plate Hinge Bracket
30	G0230-0033-DB-00	Mounting Plate
31	G0230-0034-DB-00	Pressure Plate
32	G0230-0035-DB-00	Pressure Plate Guide Screw
33	G0230-0036-DB-00	Pressure Plate Support Spring
34	G0230-0037-DB-00	Stand off
35	G0230-0038-DB-00	Stand Off Shield
36	G0230-0039-DB-00	Stand Off Shield Insert
37	G0230-0040-DB-00	Shielding Cover
38	G0230-0041-DB-00	Cover Upper Shield Slug
39	G0230-0042-DB-00	Cover Lower Shield Disc
40	G0230-0043-DB-00	Tube Holder
41	G0230-0044-DB-00	Tube Holder Insert
42	G0230-0045-DB-00	Tube

Part #	Drawing #	Part Name and Description
43	G0230-0046-DB-00	Tube End Cap
44	G0230-0047-DB-00	Spacer Ring
45	G0230-0048-DB-00	Coil Bobbin
46	G0230-0049-DB-00	Supply Spool Shaft
47	G0230-0050-DB-00	Friction Washer
48	G0230-0051-DB-00	Tape Tension Adjustment Spring
49	G0230-0052-DB-00	Source Holder
50	G0230-0053-DB-00	Coil Shield Washer
51	G0230-0054-DB-00	Led Proximity Sensor Collet
52	G0230-0055-DB-00	Retaining Cap
53	G0230-0056-DB-00	Led Proximity sensor collet Housing
54	G0230-0057-DB-00	Counter Shield
55	G0230-0058-DB-00	Skanner Spacer
56	G0230-0059-DB-00	Worm 32P. 4th'Ds, 438PD

*APPENDIX B

TABLE OF CONTENTS
ANL - ELECTRONICS DIVISION
DRAWING # EL-C-7171

REMOTE WORKING LEVEL MONITOR (RWLM)

*This appendix contains the electronic circuit diagrams. Copies of these diagrams can be obtained from the reproducible master.

APPENDIX C

This appendix contains the computer programs used in the development and calibration of the RWLM.

1. RWLM Floating Point Math Pack - Arithmetic and Utility Routines

2. RWLM Floating Point Math Pack - Format Conversion Routines

3. Math Pack Transfer Routine

4. Remote Working Level Monitor Control Program

5. RWLM Calibration Program

6. Table of Definitions of Symbols in the Calibration Program

7. Rn-daughter and Working Level Coefficient Program

RFLM FLOATING POINT MATH PACK-ARITHMETIC AND UTILITY ROUTINES

```
00000006 SYSIN   LINE  ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE    1
00000007 SYSIN      1  3200                                ORG    3200H
00000008 SYSIN      2  3200  00 32                 ARTHB         EQU       32H ;   BANK NUMBER OF ARITH PKG
00000009 SYSIN      3  3200  32 00                 ARITH         EQU       $
00000010 SYSIN      4  3200  22 00                 SCR     EQU    2200H
00000011 SYSIN      5  3200  00 22                 SCRB          EQU       22H ;      BANK NUMBER OF SCRATCH
00000012 SYSIN      6                              ;       8080 BINARY FLOATING POINT SYSTEM
00000013 SYSIN      7                              ;       PROGRAMMER CAL OHME
00000014 SYSIN      8                              ;       DATE 26 DECEMBER 1973
00000015 SYSIN      9                              ;       ARITH IS THE BEGINNING ADDRESS OF THE
00000016 SYSIN     10                              ;       ARITHMETIC AND UTILITY PACKAGE OF THE FLOATING
00000017 SYSIN     11                              ;       POINT SYSTEM.
00000018 SYSIN     12                              ;       SCR IS THE BEGINNING ADDRESS OF THE
00000019 SYSIN     13                              ;       RAM USED AS SCRATCPAD FOR THE SYSTEM.
00000020 SYSIN     14                              ;       THE RAM MULTIPLY AND DIVIDE SUBROUTINES
00000021 SYSIN     15                              ;       ARE MOVED FROM ROM TO RAM BY SUBROUTINE
00000022 SYSIN     16                              ;       INIT AND ARE EXECUTED IN RAM ONLY.
00000023 SYSIN     17                              ;       RAM MULTIPLY SUBROUTINE.
00000024 SYSIN     18  3200  22 00                 MULX4         EQU       $-ARITH+SCR
00000025 SYSIN     19  3200  C6 00                         ADI   0 ;        ADD OPERAND 3RD FRACTION
00000026 SYSIN     20  3202  00 01                 MULP3         EQU       $-1-ARITH
00000027 SYSIN     21  3202  5F                            MOV   E,A ;      4TH PARTIAL PRODUCT
00000028 SYSIN     22  3203  7A                            MOV   A,D ;      3RD PARTIAL PRODUCT
00000029 SYSIN     23  3204  CE 00                         ACI   0 ;        ADD OPERAND 2ND FRACTION
00000030 SYSIN     24  3206  00 05                 MULP2         EQU       $-1-ARITH
00000031 SYSIN     25  3206  57                            MOV   D,A ;      3RD PARTIAL PRODUCT
00000032 SYSIN     26  3207  79                            MOV   A,C ;      2ND PARTIAL PRODUCT
00000033 SYSIN     27  3208  CE 00                         ACI   0 ;        ADD OPERAND 1ST FRACTION
00000034 SYSIN     28  320A  00 09                 MULP1         EQU       $-1-ARITH
00000035 SYSIN     29  320A  C3 73 34                      JMP   MULX5 ; TO ROM CODE
00000036 SYSIN     30                              ;       RAM DIVIDE SUBROUTINE.
00000037 SYSIN     31  320D  22 0D                 DIVX5         EQU       $-ARITH+SCR
00000038 SYSIN     32  320D  D6 00                         SUI   0 ;        SUB DIVISOR 4TH FRACTION
00000039 SYSIN     33  320F  00 0E                 OP4S          EQU       $-1-ARITH
00000040 SYSIN     34  320F  7D                            MOV   A,L ;      REMAINDER 3RD FRACTION
00000041 SYSIN     35  3210  DE 00                         SBI   0 ;        SUB DIVISOR 3RD FRACTION
00000042 SYSIN     36  3212  00 11                 OP3S          EQU       $-1-ARITH
00000043 SYSIN     37  3212  6F                            MOV   L,A ;      REMAINDER 3RD FRACTION
00000044 SYSIN     38  3213  7C                            MOV   A,H ;      REMAINDER 2ND FRACTION
00000045 SYSIN     39  3214  DE 00                         SBI   0 ;        SUB DIVISOR 2ND FRACTION
00000046 SYSIN     40  3216  00 15                 OP2S          EQU       $-1-ARITH
00000047 SYSIN     41  3216  67                            MOV   H,A ;      REMAINDER 2ND FRACTION
00000048 SYSIN     42  3217  7B                            MOV   A,E ;      REMAINDER 1ST FRACTION
00000049 SYSIN     43  3218  DE 00                         SBI   0 ;        SUB DIVISOR 1ST FRACTION
00000050 SYSIN     44  321A  00 19                 OP1S          EQU       $-1-ARITH
00000051 SYSIN     45  321A  5F                            MOV   E,A ;      REMAINDER 1ST FRACTION
00000052 SYSIN     46  321B  3E 00                         MVI   A,0 ;      REMAINDER 4TH FRACTION
00000053 SYSIN     47  321D  00 1C                 OP4A          EQU       $-1-ARITH
00000054 SYSIN     48  321D  C9                            RET     ;        RETURN TO ROM
00000055 SYSIN     49  321E  22 1E                 DIVX6         EQU       $-ARITH+SCR
00000056 SYSIN     50  321E  C6 00                         ADI   0 ;        ADD DIVISOR 3RD FRACTION
00000057 SYSIN     51  3220  00 1F                 OP3A          EQU       $-1-ARITH
00000058 SYSIN     52  3220  6F                            MOV   L,A ;      REMAINDER 3RD FRACTION
```

```
ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE   2
3221  7C                          MOV    A,H ;        REMAINDER 2ND FRACTION
3222  CE 00                       ACI    0 ;          ADD DIVISOR 2ND FRACTION
3224  00 23               OP2A           EQU         $-1-ARITH
3224  67                          MOV    H,A ;        REMAINDER 2ND FRACTION
3225  7B                          MOV    A,E ;        REMAINDER 1ST FRACTION
3226  CE 00                       ACI    0 ;          ADD DIVISOR 1ST FRACTION
3228  00 27               OP1A           EQU         $-1-ARITH
3228  5F                          MOV    E,A ;        REMAINDER 1ST FRACTION
3229  3E 00                       MVI    A,0 ;        REMAINDER 4TH FRACTION
322B  00 2A               OP4X           EQU         $-1-ARITH
322B  C3 DF 34                    JMP    DIVX2 ; TO ROM CODE
                          ;       RAM LOCATIONS USED BY THE BINARY
                          ;       FLOATING POINT SYSTEM.
322E  00 2E               OVER           EQU         $-ARITH
322E  00                          DB     0 ;         INITIALLY CLEAR
322F  00 2F               PREX           EQU         OVER+1 ; PREVIOUS EXPONENT
322F  00 30               ACCE           EQU         PREX+1 ; ACCUMULATOR EXPONENT
322F  00 31               ACCS           EQU         ACCE+1 ; ACCUMULATOR SIGN
322F  00 32               ACC1           EQU         ACCS+1 ; ACCUMULATOR 1ST FRACTION
322F  00 33               ACC2           EQU         ACC1+1 ; ACCUMULATOR 2ND FRACTION
322F  00 34               ACC3           EQU         ACC2+1 ; ACCUMULATOR 3RD FRACTION
322F  00 35               SF      EQU    ACC3+1 ; SUBTRACTION FLAG
                          ;       INIT SUBROUTINE ENTRY POINT
322F  2E 2F               INIT:          MVI         L,PREX ;  TO ADDR LAST WD TO MOVE
3231  26 32               INIT1:         MVI         H,ARTHR ; TO ADDRESS ROM COPY
3233  5E                          MOV    E,M ;        CURRENT WORD OF ROM COPY
3234  26 22                       MVI    H,SCRB ;     TO ADDRESS RAM COPY
3236  73                          MOV    M,E ;        WRITE CURRENT WD TO RAM
3237  2D                          DCR    L ;          DECREMENT WORD ADDRESS
3238  F2 31 32                    JP     INIT1 ; IF MORE TO MOVE
323B  C9                          RET    ;            RETURN TO CALLER
                   .      ;       STR SUBROUTINE ENTRY POINT.
323C  73                  STR0:          MOV         M,E ;     STORE ZEROETH WORD
323D  2C                          INR    L ;          TO ADDRESS FIRST WORD
323E  77                  STR:    MOV    M,A ;        STORE FIRST WORD
323F  2C                  STR1:          INR         L ;   TO ADDRESS SECOND WORD
3240  70                          MOV    M,B ;        STORE SECOND WORD
3241  2C                          INR    L ;          TO ADDRESS THIRD WORD
3242  71                          MOV    M,C ;        STORE THIRD WORD
3243  2C                          INR    L ;          TO ADDRESS FOURTH WORD
3244  72                          MOV    M,D ;        STORE FOURTH WORD
3245  C9                          RET    ;            RETURN TO CALLER
                          ;       FLOATING POINT ZRO SUBROUTINE ENT. PNT.
3246  26 22               ZRO:    MVI    H,SCRB ;     TO ADDRESS SCRATCH BANK
3248  2E 30               ZRO1:          MVI         L,ACCE    ;   TO ADDR ACCUM EXPON
324A  AF                          XRA    A ;          ZERO
324B  77                          MOV    M,A ;        CLEAR ACCUMULATOR EXPONENT
324C  C9                          RET    ;            RETURN TO CALLER
                          ;       FLOATING POINT CHS SUBROUTINE ENT. PNT.
324D  3E 80               CHS:    MVI    A,200Q ;     MASK FOR SIGN BIT
324F  0E                          DB     016Q ;       LBI INST TO SKIP NEXT WD
                          ;       FLOATING POINT ABS SUBROUTINE ENT. PNT.
```

```
                          LINE  ADDR  B1 B2 B3 B4 ERROR  SOURCE  INTEL 8080 ASSEMBLER  PAGE    3
00000116 SYSIN             105  3250  AF                 ABS:  XRA   A :          ZERO
00000117 SYSIN             106  3251  26 22                    MVI   H,SCRB :     TO ADDRESS SCRATCH BANK
00000118 SYSIN             107  3253  2E 31                    MVI   L,ACCS :     TO ADDRESS ACCUM SIGN
00000119 SYSIN             108  3255  A6                       ANA   M            :    COMPLEMENT OF SIGN
00000120 SYSIN             109  3256  EE 80                    XRI   200Q :       COMPLEMENT THE SIGN BIT
00000121 SYSIN             110  3258  77                       MOV   M,A :        ACCUMULATOR SIGN
00000122 SYSIN             111                           :     FLOATING POINT TEST ENTRY POINT.
00000123 SYSIN             112  3259  26 22              TST:  MVI   H,SCRB :     TO ADDRESS SCRATCH BANK
00000124 SYSIN             113  325B  2E 30              TST1:       MVI       L,ACCE :   TO ADDR ACCUM EXPONENT
00000125 SYSIN             114  325D  7E                       MOV   A,M :        ACCUMULATOR EXPONENT
00000126 SYSIN             115  325E  A7                       ANA   A            :    SET CONTROL BITS
00000127 SYSIN             116  325F  CA 46 32                 JZ    ZRO :        IF ACCUMULATOR IS ZERO
00000128 SYSIN             117  3262  5F                       MOV   E,A :        ACCUMULATOR EXPONENT
00000129 SYSIN             118  3263  2C                       INR   L :          TO ADDR ACCUMULATOR SIGN
00000130 SYSIN             119  3264  7E                       MOV   A,M :        ACCUMULATOR SIGN
00000131 SYSIN             120  3265  2C                       INR   L :          TO ADDR ACCUM 1ST FRCTN
00000132 SYSIN             121  3266  AE                       XRA   M :          ACCUM SIGN AND 1ST FRCTN
00000133 SYSIN             122  3267  2C                       INR   L :          TO ADDR ACCUM 2ND FRCTN
00000134 SYSIN             123  3268  4E                       MOV   C,M :        ACCUMULATOR 2ND FRACTION
00000135 SYSIN             124  3269  2C                       INR   L :          TO ADDR ACCUM 3RD FRCTN
00000136 SYSIN             125  326A  56                       MOV   D,M :        ACCUMULATOR 3RD FRCTN
00000137 SYSIN             126  326B  C3 7A 33                 JMP   ADD12 :      TO SET EXIT CONDITIONS
00000138 SYSIN             127                           :     FLOATING POINT LOAD ENTRY POINT.
00000139 SYSIN             128  326E  7E                 LOD:  MOV   A,M :        OPERAND EXPONENT
00000140 SYSIN             129  326F  A7                       ANA   A            :    SET CONTROL BITS
00000141 SYSIN             130  3270  CA 46 32                 JZ    ZRO :        IF OPERAND IS ZERO
00000142 SYSIN             131  3273  5F                       MOV   E,A :        OPERAND EXPONENT
00000143 SYSIN             132  3274  2C                       INR   L :          TO ADDR OP SIGN AND 1ST
00000144 SYSIN             133  3275  7E                       MOV   A,M :        OPERAND SIGN AND 1ST FRCTN
000001-5 SYSIN             134  3276  2C                       INR   L :          TO ADDRESS OPERAND 2ND FRACTION
00000146 SYSIN             135  3277  4E                       MOV   C,M :        OPERAND 2ND FRACTION
00000147 SYSIN             136  3278  2C                       INR   L :          TO ADDRESS OPERAND 3RD FRACTION
00000148 SYSIN             137  3279  56                       MOV   D,M :        OPERAND 3RD FRACTION
00000149 SYSIN             138                           :     STORE THE OPERAND IN THE ACCUMULATOR.
00000150 SYSIN             139  327A  6F                       MOV   L,A :        OPERAND SIGN AND 1ST FRCTN
00000151 SYSIN             140  327B  F6 80              LOD1:       ORI       200Q :     ACCUMULATOR 1ST FRACTI
00000152 SYSIN             141  327D  47                       MOV   B,A :        ACCUMULATOR 1ST FRACTION
00000153 SYSIN             142  327E  AD                       XRA   L :          ACCUMULATOR SIGN
00000154 SYSIN             143  327F  26 22                    MVI   H,SCRB :     TO ADDRESS SCRATCH BANK
00000155 SYSIN             144  3281  2E 30                    MVI   L,ACCE :     TO ADDR ACCUM EXPONENT
00000156 SYSIN             145  3283  CD 3C 32                 CALL  STRO :       SET THE ACCUMULATOR
00000157 SYSIN             146  3286  A8                       XRA   B :          ACCUM SIGN AND 1ST FRCTN
00000158 SYSIN             147                           :     SET CONTROL BITS AND EXIT
00000159 SYSIN             148  3287  47                       MOV   B,A :        ACCUM SIGN AND 1ST FRACTION
00000160 SYSIN             149  3288  F6 01                    ORI   1 :          SET SIGN BIT FOR EXIT
00000161 SYSIN             150  328A  7B                       MOV   A,E :        ACCUMULATOR EXPONENT
00000162 SYSIN             151  328B  C9                       RET   :            RETURN TO CALLER
00000163 SYSIN             152                           :     FLOATING POINT MUL SUBROUTINE ENT. PNT.
00000164 SYSIN             153  328C  7E                 MUL:  MOV   A,M :        OPERAND EXPONENT
00000165 SYSIN             154  328D  A7                       ANA   A            :    SET CONTROL BITS
00000166 SYSIN             155  328E  C4 95 33                 CNZ   MDEX :       READ OPERAND IF NOT ZERO
00000167 SYSIN             156  3291  CA 46 32                 JZ    ZRO     :       IF ZERO OR UNDERFLOW
```

RVLM FLOATING POINT MATH PACK-ARITHMETIC AND UTILITY ROUTINES

```
LINE  ADDR  B1 B2 B3 B4 ERROR  SOURCE  INTEL 8080 ASSEMBLER  PAGE    4
 157  3294  DA CA 32                   JC      OVERF : IF OVERFLOW
 158  3297  CD 4D 34                   CALL    MULX :         CALL FIXED MULT SUBRTN
 159                           :       NORMALIZE IF NECESSARY.
 160  329A  78                         MOV     A,B :          1ST PRODUCT
 161  329B  A7                         ANA     A              :    SET CONTROL BITS
 162  329C  FA A9 32                   JM      RNDA :         IF NO NORMALIZATION REQUIRED
 163  329F  2E 3C                      MVI     L,ACCE    :      TO ADDR ACCUM EXPONENT
 164  32A1  7E                         MOV     A,M :          ACCUMULATOR EXPONENT
 165  32A2  DE 01                      SBI     1 :            DECREMENT ACCUMULATOR EXPONENT
 166  32A4  77                         MOV     M,A :          ACCUMULATOR EXPONENT
 167  32A5  C8                         RZ       :             RETURN TO CALLER IF UNDERFLOW
 168  32A6  CD BC 33                   CALL    LSH :          CALL LEFT SHIFT SUBROUTINE
 169                           :       ROUND IF NECESSARY.
 170  32A9  CD 30 34           RNDA:           CALL          ROND :     CALL ROUNDING SUBROUTI
 171  32AC  DA CA 32                   JC      OVERF : IF OVERFLOW
 172  32AF  47                         MOV     B,A :          ACCUM SIGN AND 1ST FRACTION
 173  32B0  F6 01                      ORI     1 :            SET SIGN BIT
 174  32B2  7B                         MOV     A,E :          ACCUMULATOR EXPONENT
 175  32B3  C9                         RET      :             RETURN TO CALLER
 176                           :       FLOATING POINT DIV SUBROUTINE ENT. PNT.
 177  32B4  AF                 DIV:    XRA     A :            ZERO
 178  32B5  96                         SUB     M :            COMPLEMENT OF DIVISOR EXPONENT
 179  32B6  FE 01                      CPI     1 :            SET CARRY IF DIVISION BY ZERO
 180  32B8  D4 95 33                   CNC     MDEX :         READ OPERAND IF NOT ZERO
 181  32BB  DA CA 32                   JC      OVERF :        IF OVERFLOW OR DIVISION BY ZERO
 182  32BE  CA 48 32                   JZ      ZRO1 :         IF UNDERFLOW OR ZERO
 183  32C1  4F                         MOV     C,A :          DIVISOR 1ST FRACTION
 184  32C2  CD 90 34                   CALL    DIVX :         CALL FIXED DIV SUBRTN
 185  32C5  26 22                      MVI     H,SCRB :       TO ADDRESS SCRATCH BANK
 186  32C7  DA A9 32                   JC      RNDA :         IF NO OVERFLOW
 187                           :       SET OVERFLOW FLAG.
 188  32CA  26 22              OVERF:          MVI           H,SCRB :   TO ADDRESS SCRATCH BAN
 189  32CC  2E 2E                      MVI     L,OVER :       TO ADDR OVERFLOW FLAG
 190  32CE  3E FF                      MVI     A,377Q :       OVERFLOW FLAG
 191  32D0  77                         MOV     M,A :          OVERFLOW FLAG
 192  32D1  07                         RLC      :             SET CARRY BIT FOR EXIT
 193  32D2  C9                         RET      :             RETURN TO CALLER
 194  32D3  00                         DB      0 :            CHECK SUM WORD
 195                           :       FLOATING POINT SUB SUBROUTINE ENT. PNT.
 196  32D4  3E 80              SB:     MVI     A,200Q :       MASK TO CHANGE OP SIGN
 197  32D6  0E                         DB      016Q :         LBI INST TO SKIP NEXT WD
 198                           :       FLOATING POINT ADD SUBROUTINE ENT. PNT.
 199  32D7  AF                 AD:     XRA     A :            ZERO
 200                           :       LOAD THE OPERAND.
 201  32D8  5E                         MOV     E,M :          OPERAND EXPONENT
 202  32D9  2C                         INR     L :            TO ADDR OP SIGN, 1ST FRCTN
 203  32DA  AE                         XRA     M :            OPERAND SIGN AND 1ST FRCTN
 204  32DB  47                         MOV     B,A :          OPERAND SIGN AND 1ST FRCTN
 205  32DC  2C                         INR     L :            TO ADDR OPERAND 2ND
 206  32DD  4E                         MOV     C,M :          OPERAND 2ND FRACTION
 207  32DE  2C                         INR     L :            TO ADDR OPERAND 3RD FRCTN
 208  32DF  56                         MOV     D,M :          OPERAND 3RD FRACTION
```

```
00000226 SYSIN   LINE  ADDR  B1 B2 B3 B4 ERROR  SOURCE  INTEL 8080 ASSEMBLER  PAGE    5
00000227 SYSIN    209                           :       SAVE INITIAL EXPONENT.
00000228 SYSIN    210  32E0  26 22                      MVI     H,SCRB :      TO ADDRESS SCRATCH BANK
00000229 SYSIN    211  32E2  2E 30                      MVI     L,ACCE :      TO ADDR ACCUM EXPONENT
00000230 SYSIN    212  32E4  7E                         MOV     A,M :         ACCUMULATOR EXPONENT
00000231 SYSIN    213  32E5  2D                         DCR     L :           TO ADDR INITIAL EXPONENT
00000232 SYSIN    214  32E6  77                         MOV     M,A :         INITIAL EXPONENT
00000233 SYSIN    215                           :       CHECK FOR ZERO OPERAND.
00000234 SYSIN    216  32E7  7B                         MOV     A,E :         OPERAND EXPONENT
00000235 SYSIN    217  32E8  A7                         ANA     A             :     SET CONTROL BITS
00000236 SYSIN    218  32E9  CA 5B 32                   JZ      TST1 :        IF OPERAND IS ZERO
00000237 SYSIN    219                           :       GENERATE SUBTRACTION FLAG, RESTORE
00000238 SYSIN    220                           :       SUPPRESSED FRACTION BIT.
00000239 SYSIN    221  32EC  68                         MOV     L,B :         OPERAND SIGN AND 1ST FRCTN
00000240 SYSIN    222  32ED  78                         MOV     A,B :         OPERAND SIGN AND 1ST FRACTION
00000241 SYSIN    223  32EE  F6 80                      ORI     200Q :        OPERAND 1ST FRACTION
00000242 SYSIN    224  32F0  47                         MOV     B,A :         OPERAND 1ST FRACTION
00000243 SYSIN    225  32F1  AD                         XRA     L :           OPERAND SIGN
00000244 SYSIN    226  32F2  2E 31                      MVI     L,ACCS :      TO ADDRESS ACCUMULATOR SIGN
00000245 SYSIN    227  32F4  AE                         XRA     M :           SUBTRACTION FLAG
00000246 SYSIN    228  32F5  2E 35                      MVI     L,SF :        TO ADDRESS SUBTRACTION FLAG
00000247 SYSIN    229  32F7  77                         MOV     M,A :         SUBTRACTION FLAG
00000248 SYSIN    230                           :       DETERMINE RELATIVE MAGNITUDES OF
00000249 SYSIN    231                           :       OPERAND AND ACCUMULATOR.
00000250 SYSIN    232  32F8  2E 30                      MVI     L,ACCE :      TO ADDRESS ACCUMULATOR EXPONENT
00000251 SYSIN    233  32FA  7E                         MOV     A,M :         ACCUMULATOR EXPONENT
00000252 SYSIN    234  32FB  A7                         ANA     A             :     SET CONTROL BITS
00000253 SYSIN    235  32FC  CA 86 33                   JZ      ADD17 : IF ACCUMULATOR IS ZERO
00000254 SYSIN    236  32FF  93                         SUB     E :           DIFFERENCE IN EXPONENTS
00000255 SYSIN    237  3300  DA 0E 33                   JC      ADD2 :        IF ACCUM SMALLER THAN OP
00000256 SYSIN    238                           :       CHECK FOR INSIGNIFICANT OPERAND.
00000257 SYSIN    239  3303  FA 5B 32                   JM      TST1 :        IF THE OPERAND IS INSIGNIFICANT
00000258 SYSIN    240  3306  FE 19                      CPI     031Q :        COMPARE SHIFT COUNT TO 25
00000259 SYSIN    241  3308  DA 2D 33                   JC      ADD3 :        JOIN EXCH PATH IF OP SIGNIF
00000260 SYSIN    242  330B  C3 5B 32                   JMP     TST1 :        OPERAND IS INSIGNIFICANT
00000261 SYSIN    243                           :       CHECK FOR INSIGNIFICANT ACCUMULATOR
00000262 SYSIN    244  330E  F2 86 33           ADD2:           JP            ADD17 :     IF ACCUM IS INSIGNIFIC
00000263 SYSIN    245  3311  FE E7                      CPI     347Q :        COMPARE SHIFT COUNT TO MINUS 25
00000264 SYSIN    246  3313  DA 86 33                   JC      ADD17 :       IF ACCUM IS INSIGNIFICANT
00000265 SYSIN    247  3316  73                         MOV     M,E :         OPERAND EXPONENT
00000266 SYSIN    248  3317  5F                         MOV     E,A :         SHIFT COUNT
00000267 SYSIN    249  3318  2E 35                      MVI     L,SF :        TO ADDRESS THE SUBTRACTION FLAG
00000268 SYSIN    250  331A  7E                         MOV     A,M :         SUBTRACTION FLAG
00000269 SYSIN    251  331B  2E 31                      MVI     L,ACCS :      TO ADDRESS THE ACCUMULATOR SIGN
00000270 SYSIN    252  331D  AE                         XRA     M :           OPERAND SIGN
00000271 SYSIN    253  331E  77                         MOV     M,A :         ACCUMULATOR SIGN
00000272 SYSIN    254  331F  AF                         XRA     A :           ZERO
00000273 SYSIN    255  3320  93                         SUB     E :           COMPLEMENT SHIFT COUNT
00000274 SYSIN    256                           :       EXCHANGE ACCUMULATOR AND OPERAND.
00000275 SYSIN    257  3321  2C                         INR     L :           TO ADDR ACCUM 1ST FRACTION
00000276 SYSIN    258  3322  5E                         MOV     E,M :         ACCUMULATOR 1ST FRACTION
00000277 SYSIN    259  3323  70                         MOV     M,B :         OPERAND 1ST FRACTION
00000278 SYSIN    260  3324  43                         MOV     B,E :         ACCUMULATOR 1ST FRACTION
```

```
LINE  ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE    6
261   3325  2C                          INR    L ;          TO ADDR ACCUM 2ND FRACTION
262   3326  5E                          MOV    E,M ;        ACCUMULATOR 2ND FRACTION
263   3327  71                          MOV    M,C ;        OPERAND 2ND FRACTION
264   3328  4B                          MOV    C,E ;        ACCUMULATOR 2ND FRACTION
265   3329  2C                          INR    L ;          TO ADDR ACCUM 3RD FRACTION
266   332A  5E                          MOV    E,M ;        ACCUMULATOR 3RD FRACTION
267   332B  72                          MOV    M,D ;        OPERAND 3RD FRACTION
268   332C  53                          MOV    D,E ;        ACCUMULATOR 3RD FRACTION
269                             ;       POSITION THE OPERAND.
270   332D  CD C9 33            ADD3:          CALL         RSH ;      POSITION THE OPERAND
271   3330  2E 35                       MVI    L,SF ;       TO ADDRESS SUBTRACTION FLAG
272   3332  7E                          MOV    A,M ;        SUBTRACTION FLAG
273   3333  A7                          ANA    A            ;   SET CONTROL BITS
274   3334  2E 34                       MVI    L,ACC3 ;     TO ADDR ACCUM 3RD FRCTN
275   3336  FA 5D 33                    JM     ADD9 ;       IF SUBTRACTION REQUIRED
276                             ;       ADD ADDEND TO AUGEND.
277   3339  7E                          MOV    A,M ;        AUGEND 3RD FRACTION
278   333A  82                          ADD    D ;          ADDEND 3RD FRACTION
279   333B  57                          MOV    D,A ;        SUM 3RD FRACTION
280   333C  2D                          DCR    L ;          TO ADDRESS AUGEND 2ND FRACTION
281   333D  7E                          MOV    A,M ;        AUGEND 2ND FRACTION
282   333E  89                          ADC    C ;          ADDEND 2ND FRACTION
283   333F  4F                          MOV    C,A ;        SUM 2ND FRACTION
284   3340  2D                          DCR    L ;          TO ADDRESS AUGEND 1ST FRACTION
285   3341  7E                          MOV    A,M ;        AUGEND 1ST FRACTION
286   3342  88                          ADC    B ;          ADDEND 1ST FRACTION
287   3343  47                          MOV    B,A ;        SUM 1ST FRACTION
288   3344  D2 74 33                    JNC    ADD11 ;      IF NO CARRY FROM 1ST FRCTN
289                             ;       RIGHT SHIFT SUM TO NORMALIZED POSITION.
290   3347  1F                          RAR      ;          RIGHT SHIFT SUM 1ST FRACTION
291   3348  47                          MOV    B,A ;        SUM 1ST FRACTION
292   3349  79                          MOV    A,C ;        SUM 2ND FRACTION
293   334A  1F                          RAR      ;          RIGHT SHIFT SUM 2ND FRACTION
294   334B  4F                          MOV    C,A ;        SUM 2ND FRACTION
295   334C  7A                          MOV    A,D ;        SUM 3RD FRACTION
296   334D  1F                          RAR      ;          RIGHT SHIFT SUM 3RD FRACTION
297   334E  57                          MOV    D,A ;        SUM 3RD FRACTION
298   334F  1F                          RAR      ;          4TH FRCTN = LOW BIT OF 3RD
299   3350  5F                          MOV    E,A ;        SUM 4TH FRACTION
300   3351  2E 30                       MVI    L,ACCE ;     TO ADDRESS ACCUMULATOR EXPONENT
301   3353  7E                          MOV    A,M ;        ACCUMULATOR EXPONENT
302   3354  C6 01                       ADI    1 ;          INCREMENT ACCUMULATOR EXPONENT
303   3356  DA CA 32                    JC     OVERF ;      IF OVERFLOW
304   3359  77                          MOV    M,A ;        ACCUMULATOR EXPONENT
305   335A  C3 74 33                    JMP    ADD11 ;      TO ROUND FRACTION
306                             ;       SUBTRACT SUBTRAHEND FROM MINUEND.
307   335D  AF                  ADD9:          XRA          A ;   MINUEND 4TH FRCTN IS ZERO
308   335E  93                          SUB    E ;          SUBTRAHEND 4TH FRACTION
309   335F  5F                          MOV    E,A ;        DIFFERENCE 4TH FRACTION
310   3360  7E                          MOV    A,M ;        MINUEND 3RD FRACTION
311   3361  9A                          SBB    D ;          SUBTRAHEND 3RD FRACTION
312   3362  57                          MOV    D,A ;        DIFFERENCE 3RD FRACTION
```

```
            LINE  ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE    7
00000337 SYSIN   313  3363  2D                        DCR    L ;           TO ADDRESS MINUEND 2ND FRACTION
00000338 SYSIN   314  3364  7E                        MOV    A,M :         MINUEND 2ND FRACTION
00000339 SYSIN   315  3365  99                        SBB    C ;           SUBTRAHEND 2ND FRACTION
00000340 SYSIN   316  3366  4F                        MOV    C,A :         DIFFERENCE 2ND FRACTION
00000341 SYSIN   317  3367  2D                        DCR    L ;           TO ADDRESS MINUEND 1ST FRACTION
00000342 SYSIN   318  3368  7E                        MOV    A,M :         MINUEND 1ST FRACTION
00000343 SYSIN   319  3369  98                        SBB    B ;           SUBTRAHEND 1ST FRACTION
00000344 SYSIN   320  336A  47                        MOV    B,A :         DIFFERENCE 1ST FRACTION
00000345 SYSIN   321  336B  DC EF 33          ADD10:         CC           COMP ;     COMPLEMENT IF NEGATIVE
00000346 SYSIN   322  336E  F4 02 34                  CP     NORM ;        NORMALIZE IF NECESSARY
00000347 SYSIN   323  3371  F2 48 32                  JP     ZRO1 ;        IF UNDERFLOW OR ZERO
00000348 SYSIN   324  3374  CD 30 34          ADD11:         CALL         ROND ;     CALL ROUNDING SUBROUTI
00000349 SYSIN   325  3377  DA CA 32                  JC     OVERF ;      IF OVERFLOW
00000350 SYSIN   326  337A  47                ADD12:         MOV          B,A ;      ACCUM SIGN AND 1ST FRC
00000351 SYSIN   327  337B  2E 2F                     MVI    L,PREX ;      TO ADDRESS PREV EXPONENT
00000352 SYSIN   328  337D  7B                        MOV    A,E ;         ACCUMULATOR EXPONENT
00000353 SYSIN   329  337E  96                        SUB    M ;           DIFFERENCE IN EXPONENTS
00000354 SYSIN   330  337F  6F                        MOV    L,A ;         DIFFERENCE IN EXPONENTS
00000355 SYSIN   331  3380  78                        MOV    A,B :         ACCUM SIGN AND 1ST FRCTN
00000356 SYSIN   332  3381  F6 01                     ORI    1 ;           SET SIGN BIT FOR EXIT
00000357 SYSIN   333  3383  7B                        MOV    A,E ;         ACCUMULATOR EXPONENT
00000358 SYSIN   334  3384  5D                        MOV    E,L ;         SIGNIFICANCE INDEX
00000359 SYSIN   335  3385  C9                        RET     ;            RETURN TO CALLER
00000360 SYSIN   336                          ;       LOAD THE ACCUMULATOR WITH THE OPERAND.
00000361 SYSIN   337  3386  2E 35             ADD17:         MVI          L,SF ;     TO ADDR SUBTRACTION FL
00000362 SYSIN   338  3388  7E                        MOV    A,M ;         SUBTRACTION FLAG
00000363 SYSIN   339  3389  2E 31                     MVI    L,ACCS ;      TO ADDR ACCUMULATOR SIGN
00000364 SYSIN   340  338B  AE                        XRA    M ;           OPERAND SIGN
00000365 SYSIN   341  338C  2D                        DCR    L ;           TO ADDR ACCUM EXPONENT
00000366 SYSIN   342  338D  CD 3C 32                  CALL   STRO ;        SET THE ACCUMULATOR
00000367 SYSIN   343  3390  A8                        XRA    B ;           ACCUM SIGN AND 1ST FRCTN
00000368 SYSIN   344  3391  C3 7A 33                  JMP    ADD12 ;       JOIN EXIT CODE
00000369 SYSIN   345  3394  00                        DB     0 ;           CHECK SUM WORD
00000370 SYSIN   346                          :       SUBROUTINE TO READ THE OPERAND AND
00000371 SYSIN   347                          :       CHECK THE ACCUMULATOR EXPONENT.
00000372 SYSIN   348  3395  47                MDEX:          MOV          B,A ;      EXPONENT MODIFIER
00000373 SYSIN   349  3396  2C                        INR    L ;           TO ADDR OF SIGN, 1ST FRCTN
00000374 SYSIN   350  3397  4E                        MOV    C,M :         OPERAND SIGN AND 1ST FRACTION
00000375 SYSIN   351  3398  2C                        INR    L .           TO ADDRESS OPERAND 2ND FRACTION
00000376 SYSIN   352  3399  56                        MOV    D,M -         OPERAND 2ND FRACTION
00000377 SYSIN   353  339A  2C                        INR    L ;           TO ADDRESS OPERAND 3RD FRACTION
00000378 SYSIN   354  339B  5E                        MOV    E,M .         OPERAND 3RD FRACTION
00000379 SYSIN   355  339C  26 22                     MVI    H,SCRB ;      TO ADDRESS SCRATCH BANK
00000380 SYSIN   356  339E  2E 30                     MVI    L,ACCE ;      TO ADDRESS ACCUMULATOR EXPONENT
00000381 SYSIN   357  33A0  7E                        MOV    A,M :         ACCUMULATOR EXPONENT
00000382 SYSIN   358  33A1  A7                        ANA    A             ;    SET CONTROL BITS
00000383 SYSIN   359  33A2  C8                        RZ      :            RETURN IF ACCUM IS ZERO
00000384 SYSIN   360  33A3  80                        ADD    B ;           RESULT EXPONENT PLUS BIAS
00000385 SYSIN   361  33A4  47                        MOV    B,A :         RESULT EXPONENT PLUS BIAS
00000386 SYSIN   362  33A5  1F                        RAR     ;            CARRY TO SIGN
00000387 SYSIN   363  33A6  A8                        XRA    B ;           CARRY AND SIGN MUST DIFFER
00000388 SYSIN   364  33A7  78                        MOV    A,B ;         RESULT EXPONENT PLUS BIAS
```

```
ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE    8
33A8  06 80                       MVI    B,200Q :     EXP BIAS, SIGN MASK, MS BIT
33AA  F2 B8 33                    JP     OVUN :       IF OVERFLOW OR UNDERFLOW
33AD  90                          SUB    B :          REMOVE EXCESS EXP BIAS
33AE  C8                          RZ      :           RETURN IF UNDERFLOW
33AF  77                          MOV    M,A :        RESULT EXPONENT
33B0  2C                          INR    L :          TO ADDRESS ACCUMULATOR SIGN
33B1  7E                          MOV    A,M :        ACCUMULATOR SIGN
33B2  A9                          XRA    C :          RESULT SIGN IN SIGN BIT
33B3  A0                          ANA    B            :    RESULT SIGN
33B4  77                          MOV    M,A :        RESULT SIGN
33B5  79                          MOV    A,C :        OPERAND SIGN AND 1ST FRCTN
33B6  B0                          ORA    B :          OPERAND 1ST FRACTION
33B7  C9                          RET     :           RETURN TO CALLER
33B8  07                  OVUN:          RLC          :    SET CARRY BIT IF OVERFLOW
33B9  D8                          RC      :           RETURN IF OVERFLOW
33BA  AF                          XRA    A :          ZERO
33BB  C9                          RET     :           RETURN IF UNDERFLOW
                          ;       SUBROUTINE TO LEFT SHIFT THE B, C,
                          ;       D, AND E REGISTERS ONE BIT.
33BC  7B                  LSH:    MOV    A,E :        ORIGINAL CONTENTS OF E
33BD  17                          RAL     :           LEFT SHIFT E
33BE  5F                          MOV    E,A :        RESTORE CONTENTS OF E REGISTER
33BF  7A                  LSH1:          MOV        A,D :     ORIGINAL CONTENTS OF D
33C0  17                          RAL     :           LEFT SHIFT D
33C1  57                          MOV    D,A :        RESTORE CONTENTS OF D REGISTER
33C2  79                          MOV    A,C :        ORIGINAL CONTENTS OF C REGISTER
33C3  17                          RAL     :           LEFT SHIFT C
33C4  4F                          MOV    C,A :        RESTORE CONTENTS OF C REGISTER
33C5  78                          MOV    A,B :        ORIGINAL CONTENTS OF B REGISTER
33C6  8F                          ADC    A :          LEFT SHIFT B
33C7  47                          MOV    B,A :        RESTORE CONTENTS OF B REGISTER
33C8  C9                          RET     :           RETURN TO CALLER
                          ;       RIGHT SHIFT THE B, C, D AND E REGISTERS
                          ;       BY THE SHIFT COUNT IN THE A REGISTER
                          ;       SHIFT OPERAND TO REGISTER INDICATED BY
                          ;       SHIFT COUNT
33C9  1E 00               RSH:    MVI    E,0 :        OPERAND 4TH FRCTN IS ZERO
33CB  2E 08               RSH0:          MVI        L,010Q :   EACH REG IS 8 BITS OF
33CD  BD                  RSH1:          CMP        L :   COMPARE SHIFT COUNT TO 8
33CE  FA DA 33                    JM     RSH2 :       IF REQ SHIFT LESS THAN 8
33D1  5A                          MOV    E,D :        OPERAND 4TH FRACTION
33D2  51                          MOV    D,C :        OPERAND 3RD FRACTION
33D3  48                          MOV    C,B :        OPERAND 2ND FRACTION
33D4  06 00                       MVI    B,0 :        OPERAND 1ST FRACTION IS ZERO
33D6  95                          SUB    L :          REDUCE SHIFT COUNT BY 1 REG
33D7  C2 CD 33                    JNZ    RSH1 :       IF MORE SHIFTS REQUIRED
                          ;       SHIFT OPERAND RIGHT BY -SHIFT COUNT-
                          ;       BITS.
33DA  A7                  RSH2:          ANA        A     :    SET CONTROL BITS
33DB  C8                          RZ      :           RETURN IF SHIFT COMPLETE
33DC  6F                          MOV    L,A :        SHIFT COUNT
33DD  A7                  RSH3:          ANA        A     :    CLEAR CARRY BIT
```

```
LINE  ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE    9
 417  33DE  78                          MOV    A,B :        OPERAND 1ST FRACTION
 418  33DF  1F                          RAR     :           RIGHT SHIFT OP 1ST FRCTN
 419  33E0  47                          MOV    B,A :        OPERAND 1ST FRACTION
 420  33E1  79                          MOV    A,C :        OPERAND 2ND FRACTION
 421  33E2  1F                          RAR     :           RIGHT SHIFT OP 2ND FRCTN
 422  33E3  4F                          MOV    C,A :        OPERAND 2ND FRACTION
 423  33E4  7A                          MOV    A,D :        OPERAND 3RD FRACTION
 424  33E5  1F                          RAR     :           RIGHT SHIFT OP 3RD FRCTN
 425  33E6  57                          MOV    D,A :        OPERAND 3RD FRACTION
 426  33E7  7B                          MOV    A,E :        OPERAND 4TH FRACTION
 427  33E8  1F                          RAR     :           RIGHT SHIFT OP 4TH FRCTN
 428  33E9  5F                          MOV    E,A :        OPERAND 4TH FRACTION
 429  33EA  2D                          DCR    L :          DECREMENT SHIFT COUNT
 430  33EB  C2 DD 33                    JNZ    RSH3 :       IF MORE SHIFTS REQUIRED
 431  33EE  C9                          RET     :           RETURN TO CALLER
 432                            ;       COMPLEMENT THE B, C, D, AND E REGISTERS.
 433  33EF  2D                  COMP:          DCR          L :   TO ADDR ACCUM SIGN
 434  33F0  7E                          MOV    A,M :        ACCUMULATOR SIGN
 435  33F1  EE 80                       XRI    200Q :       CHANGE SIGN
 436  33F3  77                          MOV    M,A :        ACCUMULATOR SIGN
 437  33F4  AF                  COMP1:         XRA          A :   ZERO
 438  33F5  6F                          MOV    L,A :        ZERO
 439  33F6  93                          SUB    E :          COMPLEMENT 4TH FRCTN
 440  33F7  5F                          MOV    E,A :        4TH FRACTION
 441  33F8  7D                          MOV    A,L :        ZERO
 442  33F9  9A                          SBB    D :          COMPLEMENT 3RD FRCTN
 443  33FA  57                          MOV    D,A :        3RD FRACTION
 444  33FB  7D                          MOV    A,L :        ZERO
 445  33FC  99                          SBB    C :          COMPLEMENT 2ND FRCTN
 446  33FD  4F                          MOV    C,A :        2ND FRACTION
 447  33FE  7D                          MOV    A,L :        ZERO
 448  33FF  98                          SBB    B :          COMPLEMENT 1ST FRCTN
 449  3400  47                          MOV    B,A :        1ST FRACTION
 450  3401  C9                          RET     :           RETURN TO CALLER
 451                            ;       NORMALIZE THE REGISTERS.
 452  3402  2E 20               NORM:          MVI          L,040Q :   MAX NORMALIZING SHIFT
 453  3404  78                  NORM1:         MOV          A,B :      1ST FRACTION
 454  3405  A7                          ANA    A            :   SET CONTROL BITS
 455  3406  C2 ?2 34                    JNZ    NORM3 :      IF 1ST FRACTION NONZERO
 456  3409  41                          MOV    B,C :        1ST FRACTION
 457  340A  4A                          MOV    C,D :        2ND FRACTION
 458  340B  53                          MOV    D,E :        3RD FRACTION
 459  340C  5F                          MOV    E,A :        ZERO 4TH FRACTION
 460  340D  7D                          MOV    A,L :        NORMALIZING SHIFT COUNT
 461  340E  D6 08                       SUI    010Q :       REDUCE SHIFT COUNT
 462  3410  6F                          MOV    L,A :        NORMALIZING SHIFT COUNT
 463  3411  C2 04 34                    JNZ    NORM1 :      IF FRACTION NONZERO
 464  3414  C9                          RET     :           IF FRACTION IS ZERO
 465  3415  2D                  NORM2:         DCR          L :   DECREMENT SHIFT COUNT
 466  3416  7B                          MOV    A,E :        ORIGINAL CONTENTS OF E
 467  3417  17                          RAL     :           LEFT SHIFT E
 468  3418  5F                          MOV    E,A :        RESTORE CONTENTS OF E REGISTER
```

```
00000501 SYSIN    LINE  ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE    10
00000502 SYSIN     469  3419  7A                          MOV    A,D ;         ORIGINAL CONTENTS OF D REGISTER
00000503 SYSIN     470  341A  17                          RAL     ;            LEFT SHIFT D
00000504 SYSIN     471  341B  57                          MOV    D,A ;         RESTORE CONTENTS OF D REGISTER
00000505 SYSIN     472  341C  79                          MOV    A,C ;         ORIGINAL CONTENTS OF C REGISTER
00000506 SYSIN     473  341D  17                          RAL     ;            LEFT SHIFT C
00000507 SYSIN     474  341E  4F                          MOV    C,A ;         RESTORE CONTENTS OF C REGISTER
00000508 SYSIN     475  341F  78                          MOV    A,B ;         ORIGINAL CONTENTS OF B REGISTER
00000509 SYSIN     476  3420  8F                          ADC    A ;           LEFT SHIFT B
00000510 SYSIN     477  3421  47                          MOV    B,A ;         RESTORE CONTENTS OF B REGISTER
00000511 SYSIN     478  3422  F2 15 34              NORM3:       JP           NORM2 ;    IF NOT NORMALIZED
00000512 SYSIN     479  3425  7D                          MOV    A,L ;         NORMALIZING SHIFT COUNT
00000513 SYSIN     480  3426  D6 20                       SUI    040Q ;        REMOVE BIAS
00000514 SYSIN     481  3428  2E 30                       MVI    L,ACCE ;      TO ADDR ACCUM EXPONENT
00000515 SYSIN     482  342A  86                          ADD    M ;           ADJUST ACCUM EXPONENT
00000516 SYSIN     483  342B  77                          MOV    M,A ;         NEW ACCUM EXPONENT
00000517 SYSIN     484  342C  C8                          RZ      ;            RETURN IF ZERO EXP
00000518 SYSIN     485  342D  1F                          RAR     ;            BORROW BIT TO SIGN
00000519 SYSIN     486  342E  A7                          ANA    A             ;    SET SIGN TO IND. UNDERFLOW
00000520 SYSIN     487  342F  C9                          RET     ;            RETURN TO CALLER
00000521 SYSIN     488                              ;     SUBROUTINE TO ROUND THE B, C, D REGISTERS.
00000522 SYSIN     489  3430  2E 30                 ROND:        MVI          L,ACCE ;   TO ADDR ACCUM EXPONENT
00000523 SYSIN     490  3432  7B                          MOV    A,E ;         4TH FRACTION
00000524 SYSIN     491  3433  A7                          ANA    A             ;    SET CONTROL BITS
00000525 SYSIN     492  3434  5E                          MOV    E,M ;         ACCUMULATOR EXPONENT
00000526 SYSIN     493  3435  FC 3F 34                    CM     RNDR ;        CALL 2ND LEVEL ROUNDER
00000527 SYSIN     494  3438  D8                          RC      ;            IF OVERFLOW
00000528 SYSIN     495  3439  78                          MOV    A,B ;         1ST FRACTION
00000529 SYSIN     496  343A  2C                          INR    L ;           TO ADDR ACCUM SIGN
00000530 SYSIN     497  343B  AE                          XRA    M ;           ACCUM SIGN AND 1ST FRCTN
00000531 SYSIN     498  343C  C3 3F 32                    JMP    STR1 ;        RETURN THRU STORE SUBR.
00000532 SYSIN     499                              ;     SECOND LEVEL ROUNDING SUBROUTINE.
00000533 SYSIN     500  343F  14                    RNDR:        INR          D ;    ROUND 3RD FRACTION
00000534 SYSIN     501  3440  C0                          RNZ     ;            RETURN IF NO CARRY
00000535 SYSIN     502  3441  0C                          INR    C ;           CARRY TO 2ND FRACTION
00000536 SYSIN     503  3442  C0                          RNZ     ;            RETURN IF NO CARRY
00000537 SYSIN     504  3443  04                          INR    B ;           CARRY TO 1ST FRACTION
00000538 SYSIN     505  3444  C0                          RNZ     ;            RETURN IF NO CARRY
00000539 SYSIN     506  3445  7B                          MOV    A,E ;         ACCUMULATOR EXPONENT
00000540 SYSIN     507  3446  C6 01                       ADI    1 ;           INCREMENT ACCUM EXPONENT
00000541 SYSIN     508  3448  5F                          MOV    E,A ;         NEW ACCUM EXPONENT
00000542 SYSIN     509  3449  06 80                       MVI    B,200Q ;      NEW 1ST FRACTION
00000543 SYSIN     510  344B  77                          MOV    M,A ;         NEW ACCUM EXPONENT
00000544 SYSIN     511  344C  C9                          RET     ;            RETURN TO ROND SUBROUTINE
00000545 SYSIN     512                              ;     FIXED POINT MULTIPLY SUBROUTINE.
00000546 SYSIN     513  344D  2E 09                 MULX:        MVI          L,MULP1 ;  TO ADDR 1ST MULTIPLICA
00000547 SYSIN     514  344F  77                          MOV    M,A ;         1ST MULTIPLICAND
00000548 SYSIN     515  3450  2E 05                       MVI    L,MULP2 ;     TO ADDR 2ND MULTIPLICAND
00000549 SYSIN     516  3452  72                          MOV    M,D ;         2ND MULTIPLICAND
00000550 SYSIN     517  3453  2E 01                       MVI    L,MULP3 ;     TO ADDR 3RD MULTIPLICAND
00000551 SYSIN     518  3455  73                          MOV    M,E ;         3RD MULTIPLICAND
00000552 SYSIN     519  3456  AF                          XRA    A ;           CLEAR 6TH PRODUCT
00000553 SYSIN     520  3457  5F                          MOV    E,A ;         CLEAR 5TH PRODUCT
```

```
ADDR  B1 B2 B3 B4 ERROR    SOURCE  INTEL 8080 ASSEMBLER  PAGE   11
3458  57                            MOV     D,A ;       CLEAR 4TH PRODUCT
                           ;        MULTIPLY BY EACH ACCUMULATOR
                           ;        FRACTION IN TURN.
3459  2E 34                         MVI     L,ACC3 ;    TO ADDRESS 3RD FRCTN
345B  CD 68 34                      CALL    MULX2 ;     MULTIPLY BY ACCUM 3RD FRCTN
345E  2E 33                         MVI     L,ACC2 ;    TO ADDRESS 2ND FRCTN
3460  CD 65 34                      CALL    MULX1 ;     MULTIPLY BY ACCUM 2ND FRCTN
3463  2E 32                         MVI     L,ACC1 ;    TO ADDRESS 1ST FRCTN
                           ;        MULTIPLY BY ONE ACCUMULATOR WORD.
3465  7A                   MULX1:           MOV         A,D ;       5TH PARTIAL PRODUCT
3466  59                            MOV     E,C ;       4TH PARTIAL PRODUCT
3467  50                            MOV     D,B ;       3RD PARTIAL PRODUCT
3468  46                   MULX2:           MOV         B,M ; MULTIPLIER
3469  6F                            MOV     L,A ;       5TH PARTIAL PRODUCT
346A  AF                            XRA     A ;         ZERO
346B  4F                            MOV     C,A ;       2ND PARTIAL PRODUCT
346C  90                            SUB     B ;         SET CARRY BIT FOR EXIT FLAG
346D  DA 79 34                      JC      MULX3 ;     IF MULTIPLIER IS NOT ZERO
3470  4A                            MOV     C,D ;       2ND PARTIAL PRODUCT
3471  53                            MOV     D,E ;       3RD PARTIAL PRODUCT
3472  C9                            RET      ;          MULT BY ZERO COMPLETE
                           ;        COMPLETE ADDITION OF MULTIPLICAND.
3473  4F                   MULX5:           MOV         C,A ;       2ND PARTIAL PRODUCT
3474  D2 79 34                      JNC     MULX3 ;     IF NO CARRY TO 1ST PRODUCT
3477  04                            INR     B ;         ADD CARRY TO 1ST PRODUCT
3478  A7                            ANA     A           ;   CLEAR CARRY BIT
                           ;        LOOP FOR EACH BIT OF MULTIPLIER WORD.
3479  7D                   MULX3:           MOV         A,L ;       5TH PART PRODUCT, EXIT
347A  8F                            ADC     A ;         SHIFT EXIT FLAG OUT IF DONE
347B  C8                            RZ       ;          EXIT IF MULTIPLICATION DONE
347C  6F                            MOV     L,A ;       5TH PART PRODUCT, EXIT FLAG
347D  7B                            MOV     A,E ;       4TH PARTIAL PRODUCT
347E  17                            RAL      ;          SHIFT 4TH PARTIAL PRODUCT
347F  5F                            MOV     E,A ;       4TH PARTIAL PRODUCT
3480  7A                            MOV     A,D ;       3RD PARTIAL PRODUCT
3481  17                            RAL      ;          SHIFT 3RD PARTIAL PRODUCT
3482  57                            MOV     D,A ;       3RD PARTIAL PRODUCT
3483  79                            MOV     A,C ;       2ND PARTIAL PRODUCT
3484  17                            RAL      ;          SHIFT 2ND PARTIAL PRODUCT
3485  4F                            MOV     C,A ;       2ND PARTIAL PRODUCT
3486  78                            MOV     A,B ;       1ST PART PROD AND MULTPLIER
3487  17                            RAL      ;          SHIFT 1ST PROD AND MULTIPLIER
3488  47                            MOV     B,A ;       1ST PART PROD AND MULTIPLIER
3489  D2 79 34                      JNC     MULX3 ;     IF NO ADDITION REQUIRED
                           ;        ADD THE MULTIPLICAND TO THE PRODUCT
                           ;        IF THE MULTIPLIER BIT IS ONE.
348C  7B                            MOV     A,E ;       4TH PARTIAL PRODUCT
348D  C3 00 22                      JMP     MULX4 ;     TO RAM CODE
                           ;        FIXED POINT DIVIDE SUBROUTINE.
                           ;        SUBTRACT DIVISOR FROM ACCUMULATOR TO
                           ;        OBTAIN 1ST REMAINDER.
3490  2E 34                DIVX:            MVI         L,ACC3 ;   TO ADDRESS ACCUM 3RD F
```

```
00000611 SYSIN   LINE  ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE   12
00000612 SYSIN    573  3492  7E                            MOV   A,M :        ACCUMULATOR 3RD FRACTION
00000613 SYSIN    574  3493  93                            SUB   E :          DIVISOR 3RD FRACTION
00000614 SYSIN    575  3494  77                            MOV   M,A :        REMAINDER 3RD FRACTION
00000615 SYSIN    576  3495  2D                            DCR   L :          TO ADDRESS ACCUM 2ND FRCTN
00000616 SYSIN    577  3496  7E                            MOV   A,M :        ACCUMULATOR 2ND FRACTION
00000617 SYSIN    578  3497  9A                            SBB   D :          DIVISOR 2ND FRACTION
00000618 SYSIN    579  3498  77                            MOV   M,A :        REMAINDER 2ND FRACTION
00000619 SYSIN    580  3499  2D                            DCR   L :          TO ADDRESS ACCUM 1ST FRCTN
00000620 SYSIN    581  349A  7E                            MOV   A,M :        ACCUMULATOR 1ST FRACTION
00000621 SYSIN    582  349B  99                            SBB   C :          DIVISOR 1ST FRACTION
00000622 SYSIN    583  349C  77                            MOV   M,A :        REMAINDER 1ST FRACTION
00000623 SYSIN    584                            :         HALVE THE DIVISOR AND STORE FOR
00000624 SYSIN    585                            :         ADDITION OR SUBTRACTION.
00000625 SYSIN    586  349D  79                            MOV   A,C :        DIVISOR 1ST FRACTION
00000626 SYSIN    587  349E  17                            RAL    :           SET CARRY BIT
00000627 SYSIN    588  349F  79                            MOV   A,C :        DIVISOR 1ST FRACTION
00000628 SYSIN    589  34A0  1F                            RAR    :           HALF OF DIVISOR 1ST FRCTN
00000629 SYSIN    590                            :         + 200B TO CORRECT QUOTIENT
00000630 SYSIN    591  34A1  2E 19                         MVI   L,OP1S :     TO ADDRESS 1ST SUBTRACT DIVISOR
00000631 SYSIN    592  34A3  77                            MOV   M,A :        1ST SUBTRACT DIVISOR
00000632 SYSIN    593  34A4  2E 27                         MVI   L,OP1A :     TO ADDRESS 1ST ADD DIVISOR
00000633 SYSIN    594  34A6  77                            MOV   M,A :        1ST ADD DIVISOR
00000634 SYSIN    595  34A7  7A                            MOV   A,D :        DIVISOR 2ND FRACTION
00000635 SYSIN    596  34A8  1F                            RAR    :           HALF OF DIVISOR 2ND FRACTION
00000636 SYSIN    597  34A9  2E 15                         MVI   L,OP2S :     TO ADDRESS 2ND SUBTRACT DIVISOR
00000637 SYSIN    598  34AB  77                            MOV   M,A :        2ND SUBTRACT DIVISOR
00000638 SYSIN    599  34AC  2E 23                         MVI   L,OP2A :     TO ADDRESS 2ND ADD DIVISOR
00000639 SYSIN    600  34AE  77                            MOV   M,A :        2ND ADD DIVISOR
00000640 SYSIN    601  34AF  7B                            MOV   A,E :        DIVISOR 3RD FRACTION
00000641 SYSIN    602  34B0  1F                            RAR    :           HALF OF DIVISOR 3RD FRACTION
00000642 SYSIN    603  34B1  2E 11                         MVI   L,OP3S :     TO ADDRESS 3RD SUBTRACT DIVISOR
00000643 SYSIN    604  34B3  77                            MOV   M,A :        3RD SUBTRACT DIVISOR
00000644 SYSIN    605  34B4  2E 1F                         MVI   L,OP3A :     TO ADDRESS 3RD ADD DIVISOR
00000645 SYSIN    606  34B6  77                            MOV   M,A :        3RD ADD DIVISOR
00000646 SYSIN    607  34B7  06 00                         MVI   B,0 :        INIT QUOTIENT 1ST FRCTN
00000647 SYSIN    608  34B9  78                            MOV   A,B :        DIVISOR FOURTH FRACTION IS ZERO
00000648 SYSIN    609  34BA  1F                            RAR    :           LOW BIT OF DIVISOR 3RD FRACTION
00000649 SYSIN    610  34BB  2E 0E                         MVI   L,OP4S :     TO ADDRESS 4TH SUBTRACT DIVISOR
00000650 SYSIN    611  34BD  77                            MOV   M,A :        4TH SUBTRACT DIVISOR
00000651 SYSIN    612  34BE  2E 1C                         MVI   L,OP4A :     TO ADDRESS 4TH ADD DIVISOR
00000652 SYSIN    613  34C0  77                            MOV   M,A :        4TH ADD DIVISOR
00000653 SYSIN    614  34C1  2E 2A                         MVI   L,OP4X :     TO ADDRESS 4TH ADD DIVISOR
00000654 SYSIN    615  34C3  77                            MOV   M,A :        4TH ADD DIVISOR
00000655 SYSIN    616                            :         LOAD 1ST REMAINDER, CHECK SIGN.
00000656 SYSIN    617  34C4  2E 32                         MVI   L,ACC1 :     TO ADDR REMAINDER 1ST FRCTN
00000657 SYSIN    618  34C6  7E                            MOV   A,M :        REMAINDER 1ST FRACTION
00000658 SYSIN    619  34C7  2C                            INR   L :          TO ADDR REMAINDER 2ND FRCTN
00000659 SYSIN    620  34C8  56                            MOV   D,M :        REMAINDER 2ND FRACTION
00000660 SYSIN    621  34C9  2C                            INR   L :          TO ADDR REMAINDER 3RD FRCTN
00000661 SYSIN    622  34CA  5E                            MOV   E,M :        REMAINDER 3RD FRACTION
00000662 SYSIN    623  34CB  A7                            ANA   A            :     SET CONTROL BITS
00000663 SYSIN    624  34CC  FA F6 34                      JM    DIVX4 :      IF REMAINDER IS NEGATIVE
```

```
ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE  13
                          ;       ADJUST EXPONENT,POSITION REMAINDER
                          ;       AND INITIALIZE THE QUOTIENT.
34CF  2E 30                       MVI     L,ACCE ;     TO ADDRESS ACCUMULATOR EXPONENT
34D1  4E                          MOV     C,M ;        QUOTIENT EXPONENT
34D2  0C                          INR     C ;          INCREMENT QUOTIENT EXPONENT
34D3  C8                          RZ       ;           RETURN IF OVERFLOW
34D4  71                          MOV     M,C ;        QUOTIENT EXPONENT
34D5  6B                          MOV     L,E ;        REMAINDER 3RD FRACTION
34D6  62                          MOV     H,D ;        REMAINDER 2ND FRACTION
34D7  5F                          MOV     E,A ;        REMAINDER 1ST FRACTION
34D8  16 01                       MVI     D,1 ;        INITIALIZE QUOT 3RD FRCTN
34DA  48                          MOV     C,B ;        INITIALIZE QUOT 2ND FRCTN
                          ;       SUBTRACT THE DIVISOR FROM THE REMAINDER
                          ;       IF IT IS POSITIVE
34DB  AF                  DIVX1:          XRA     A ;   REMAINDER 4TH FRCTN IS ZERO
34DC  CD 0D 22                    CALL    DIVX5 ;      CALL RAM SECTION
34DF  07                  DIVX2:          RLC      ;   SHFT REM 4TH FRCTN TO CY
                          ;       SHIFT THE REMAINDER LEFT ONE BIT.
34E0  78                          MOV     A,B ;        QUOTIENT 1ST FRACTION
34E1  17                          RAL      ;           MS BIT OF QUOTIENT TO CY
34E2  D8                          RC       ;           IF DIVISION COMPLETE
34E3  1F                          RAR      ;           REMAINDER 4TH FRCTN TO CY
34E4  7D                          MOV     A,L ;        REMAINDER 3RD FRACTION
34E5  17                          RAL      ;           LEFT SHIFT REM 3RD FRCTN
34E6  6F                          MOV     L,A ;        REMAINDER 3RD FRACTION
34E7  7C                          MOV     A,H ;        REMAINDER 2ND FRACTION
34E8  17                          RAL      ;           LEFT SHIFT REM 2ND FRCTN
34E9  67                          MOV     H,A ;        REMAINDER 2ND FRACTION
34EA  CD BC 33                    CALL    LSH ;        CALL LEFT SHIFT SUBROUTINE
                          ;       BRANCH IF SUBTRACTION IS REQUIRED
34ED  7A                          MOV     A,D ;        QUOTIENT 3RD FRACTION
34EE  0F                          RRC      ;           REM SIGN INDIC TO CARRY BIT
34EF  DA DB 34                    JC      DIVX1 ;      TO SUB DIVISOR IF REM POS
                          ;       ADD THE DIVISOR IF THE REMAINDER
                          ;       IS NEGATIVE.
34F2  7D                  DIVX3:          MOV     A,L ;     REMAINDER 3RD FRACTION
34F3  C3 1E 22                    JMP     DIVX6 ;      TO RAM CODE
                          ;       POSITION THE REMAINDER AND INITIALIZE
                          ;       THE QUOTIENT.
34F6  6B                  DIVX4:          MOV     L,E ;     REMAINDER 3RD FRACTION
34F7  62                          MOV     H,D ;        REMAINDER 2ND FRACTION
34F8  5F                          MOV     E,A ;        REMAINDER 1ST FRACTION
34F9  50                          MOV     D,B ;        INITIALIZE QUOT 3RD FRCTN
34FA  48                          MOV     C,B ;        INITIALIZE QUOT 2ND FRCTN
34FB  C3 F2 34                    JMP     DIVX3 ;      ADD DIVISOR IF REM IS NEG
34FE  00                          DB      0 ;          CHECKSUM WORD
34FF                              END
TOTAL ASSEMBLER ERRORS =     0
```

RWLM FLOATING POINT MATH PACK-FORMAT CONVERSION ROUTINES

```
00000006 SYSIN   ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE    1
00000007 SYSIN   3500                              ORG    3500H
00000008 SYSIN   3500  22 00                SCR    EQU    2200H
00000009 SYSIN   3500  00 22                SCRB          EQU        22H  ;      BANK NUMBER OF SCRATC
00000010 SYSIN   3500  32 00                ARITH         EQU        3200H ;     BASE ADDRESS OF ARITHM
00000011 SYSIN                              ;      8080 BINARY FLOATING POINT SYSTEM
00000012 SYSIN                              ;      FORMAT CONVERSION PACKAGE
00000013 SYSIN                              ;      PROGRAMMER CAL OHME
00000014 SYSIN                              ;      DATE 26 DECEMBER 1973
00000015 SYSIN                              ;      ARITH IS THE BEGINNING ADDRESS OF THE
00000016 SYSIN                              ;      ARITHMETIC AND UTILITY PACKAGE OF THE FLOATING
00000017 SYSIN                              ;      POINT SYSTEM.
00000018 SYSIN                              ;      SCR IS THE BEGINNING ADDRESS OF THE
00000019 SYSIN                              ;      RAM USED AS SCRATCPAD FOR THE SYSTEM.
00000020 SYSIN                              ;      RAM LOCATIONS USED BY THE BINARY
00000021 SYSIN                              ;      FLOATING POINT SYSTEM.
00000022 SYSIN   3500  00 2E                OVER          EQU        56Q ;       OVERFLOW FLAG
00000023 SYSIN   3500  00 30                ACCE          EQU        60Q ;       ACCUMULATOR EXPONENT
00000024 SYSIN   3500  00 31                ACCS          EQU        ACCE+1 ;    ACCUMULATOR SIGN
00000025 SYSIN   3500  00 32                ACC1          EQU        ACCS+1 ;    ACCUMULATOR 1ST FRACTI
00000026 SYSIN   3500  00 33                ACC2          EQU        ACC1+1 ;    ACCUMULATOR 2ND FRACTI
00000027 SYSIN   3500  00 34                ACC3          EQU        ACC2+1 ;    ACCUMULATOR 3RD FRACTI
00000028 SYSIN   3500  00 35                SF     EQU    ACC3+1 ;     SUBTRACTION FLAG
00000029 SYSIN   3500  00 36                ADRL          EQU        SF+1 ;      CHARACTER STRING WORD
00000030 SYSIN   3500  00 37                ADRH          EQU        ADRL+1 ;    CHARACTER STRING BANK
00000031 SYSIN   3500  00 38                TMP1          EQU        ADRH+1 ;    TEMPORARY STORAGE
00000032 SYSIN   3500  00 39                TMP2          EQU        TMP1+1 ;    TEMPORARY STORAGE
00000033 SYSIN   3500  00 3A                TMP3          EQU        TMP2+1 ;    TEMPORARY STORAGE
00000034 SYSIN   3500  00 3B                VALE          EQU        TMP3+1 ;    VALUE EXPONENT
00000035 SYSIN   3500  00 3C                VAL1          EQU        VALE+1 ;    VALUE 1ST FRACTION
00000036 SYSIN   3500  00 3D                VAL2          EQU        VAL1+1 ;    VALUE 2ND FRACTION
00000037 SYSIN   3500  00 3E                VAL3          EQU        VAL2+1 ;    VALUE 3RD FRACTION
00000038 SYSIN   3500  00 3F                TMP4          EQU        VAL3+1 ;    TEMPORARY STORAGE
00000039 SYSIN                              ;      ADDRESSES IN THE ARITHMETIC AND UTILITY
00000040 SYSIN                              ;      PACKAGE REFERENCED BY THE FORMAT CONVERSION
00000041 SYSIN                              ;      PACKAGE.
00000042 SYSIN   3500  32 3E                STR    EQU    ARITH+76Q
00000043 SYSIN   3500  32 46                ZRO    EQU    ARITH+106Q
00000044 SYSIN   3500  32 50                ABS    EQU    ARITH+120Q
00000045 SYSIN   3500  32 59                TST    EQU    ARITH+131Q
00000046 SYSIN   3500  32 6E                LOD    EQU    ARITH+156Q
00000047 SYSIN   3500  32 8C                MUL    EQU    ARITH+214Q
00000048 SYSIN   3500  32 B4                DIV    EQU    ARITH+264Q
00000049 SYSIN   3500  32 D7                AD     EQU    ARITH+327Q
00000050 SYSIN   3500  33 6B                ADD10         EQU        ARITH+553Q
00000051 SYSIN   3500  33 BC                LSH    EQU    ARITH+674Q
00000052 SYSIN   3500  33 C9                RSH    EQU    ARITH+711Q
00000053 SYSIN   3500  33 EF                COMP          EQU        ARITH+757Q
00000054 SYSIN                              ;      SUBROUTINE TO CONVERT FROM FIXED
00000055 SYSIN                              ;      POINT TO FLOATING POINT FORMAT.
00000056 SYSIN   3500  6B                   FLT:   MOV    L,E ;         INPUT EXPONENT
00000057 SYSIN   3501  5A                          MOV    E,D ;         4TH INPUT FRACTION
00000058 SYSIN   3502  51                          MOV    D,C ;         3RD INPUT FRACTION
```

```
00000061 SYSIN      LINE  ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE    2
00000062 SYSIN       53   3503  48                          MOV    C,B ;          2ND INPUT FRACTION
00000063 SYSIN       54   3504  47                          MOV    B,A ;          1ST INPUT FRACTION
00000064 SYSIN       55   3505  7D                          MOV    A,L ;          INPUT EXPONENT
00000065 SYSIN       56   3506  EE 80                       XRI    200Q ;         APPLY EXPONENT BIAS
00000066 SYSIN       57   3508  26 22                       MVI    H,SCRB ;       TO ADDRESS SCRATCH BANK
00000067 SYSIN       58   350A  2E 30                       MVI    L,ACCE ;       TO ADDR ACCUM EXPONENT
00000068 SYSIN       59   350C  77                          MOV    M,A ;          ACCUMULATOR EXPONENT
00000069 SYSIN       60   350D  2C                          INR    L ;            TO ADDRESS ACCUM SIGN
00000070 SYSIN       61   350E  36 80                       MVI    M,200Q ;       SET ACCUM SIGN POSITIVE
00000071 SYSIN       62   3510  2C                          INR    L ;            TO ADDR ACCUM 1ST FRCTN
00000072 SYSIN       63   3511  78                          MOV    A,B ;          1ST INPUT FRACTION
00000073 SYSIN       64   3512  A7                          ANA    A              ;    SET SIGN BIT
00000074 SYSIN       65   3513  17                          RAL     ;             INPUT SIGN TO CARRY
00000075 SYSIN       66   3514  C3 6B 33                    JMP    ADD10 ;        COMPLETE CONVERSION
00000076 SYSIN       67                             ;       SUBROUTINE TO CONVERT FROM FLOATING
00000077 SYSIN       68                             ;       POINT TO FIXED POINT FORMAT.
00000078 SYSIN       69   3517  26 22               FIX:    MVI    H,SCRB ;       TO ADDRESS SCRATCH BANK
00000079 SYSIN       70   3519  2E 30                       MVI    L,ACCE ;       TO ADDR ACCUM EXPONENT
00000080 SYSIN       71   351B  7E                          MOV    A,M ;          ACCUMULATOR EXPONENT
00000081 SYSIN       72   351C  A7                          ANA    A              ;    SET CONTROL BITS
00000082 SYSIN       73   351D  CA 45 35                    JZ     FIX1 ;         IF ACCUMULATOR IS ZERO
00000083 SYSIN       74   3520  7B                          MOV    A,E ;          INPUT EXPONENT
00000084 SYSIN       75   3521  C6 7F                       ADI    177Q ;         APPLY BIAS - 1
00000085 SYSIN       76   3523  96                          SUB    M ;            SHIFT COUNT - 1
00000086 SYSIN       77   3524  D8                          RC      ;             RETURN IF ACCUM TOO LARGE
00000087 SYSIN       78   3525  FE 1F                       CPI    037Q ;         COMPARE TO LARGE SHIFT
00000088 SYSIN       79   3527  D2 45 35                    JNC    FIX1 ;         IF ACCUMULATOR TOO SMALL
00000089 SYSIN       80   352A  C6 01                       ADI    1 ;            SHIFT COUNT
00000090 SYSIN       81   352C  2E 32                       MVI    L,ACC1 ;       TO ADDR ACCUM 1ST FRCTN
00000091 SYSIN       82   352E  46                          MOV    B,M ;          ACCUMULATOR 1ST FRACTION
00000092 SYSIN       83   352F  2C                          INR    L ;            TO ADDR ACCUM 2ND FRCTN
00000093 SYSIN       84   3530  4E                          MOV    C,M ;          ACCUMULATOR 2ND FRCTN
00000094 SYSIN       85   3531  2C                          INR    L ;            TO ADDR ACCUM 3RD FRCTN
00000095 SYSIN       86   3532  56                          MOV    D,M ;          ACCUMULATOR 3RD FRCTN
00000096 SYSIN       87   3533  CD C9 33                    CALL   RSH ;          POSITION THE FRACTION
00000097 SYSIN       88   3536  2E 31                       MVI    L,ACCS ;       TO ADDR ACCUM SIGN
00000098 SYSIN       89   3538  7E                          MOV    A,M ;          ACCUMULATOR SIGN
00000099 SYSIN       90   3539  A7                          ANA    A              ;    SET CONTROL BITS
00000100 SYSIN       91   353A  F4 EF 33                    CP     COMP ;         COMPLEMENT FRCTN IF NEG
00000101 SYSIN       92   353D  3E 01                       MVI    A,1 ;          NON-ZERO
00000102 SYSIN       93   353F  B0                          ORA    B ;            SET CONTROL BITS FOR EXIT
00000103 SYSIN       94   3540  78                          MOV    A,B ;          1ST RESULT
00000104 SYSIN       95   3541  41                          MOV    B,C ;          2ND RESULT
00000105 SYSIN       96   3542  4A                          MOV    C,D ;          3RD RESULT
00000106 SYSIN       97   3543  53                          MOV    D,E ;          4TH RESULT
00000107 SYSIN       98   3544  C9                          RET     ;             RETURN TO CALLER
00000108 SYSIN       99   3545  AF                  FIX1:          XRA          A ;   ZERO
00000109 SYSIN      100   3546  47                          MOV    B,A ;          ZERO
00000110 SYSIN      101   3547  4F                          MOV    C,A ;          ZERO
00000111 SYSIN      102   3548  57                          MOV    D,A ;          ZERO
00000112 SYSIN      103   3549  C9                          RET     ;             RETURN TO CALLER
00000113 SYSIN      104   354A  00                          DB     0 ;            CHECKSUM WORD
```

```
ADDR  B1 B2 B3 B4 ERROR  SOURCE  INTEL 8080 ASSEMBLER  PAGE  3
                         ;      INP SUBROUTINE ENTRY POINT.
                         ;      INITIALIZE TEMPORARY STORAGE.
354B  5E                 INP:   MOV   E,M ;      FIRST CHARACTER OF STRING
354C  CD D5 36                  CALL  SVAD ;     SET CHAR ADDR, PNT FLG, EXP
354F  2C                        INR   L ;        TO ADDRESS VALUE SIGN
3550  36 80                     MVI   M,200Q ;   SET VALUE SIGN POSITIVE
3552  2E 30                     MVI   L,ACCE ;   TO ADDR ACCUM EXPONENT
3554  72                        MOV   M,D ;      SET ACCUM TO ZERO
3555  7B                        MOV   A,E ;      FIRST CHARACTER
3556  FE F0                     CPI   360Q ;     COMPARE TO SPACE
3558  CA 68 35                  JZ    INP1 ;     IF SPACE CHARACTER
355B  FE FB                     CPI   373Q ;     COMPARE CHAR TO PLUS
355D  CA 68 35                  JZ    INP1 ;     IF PLUS SIGN
3560  FE FD                     CPI   375Q ;     COMPARE TO MINUS
3562  C2 6B 35                  JNZ   INP2 ;     IF NOT MINUS SIGN
3565  2E 3A                     MVI   L,TMP3 ;   TO ADDR VALUE SIGN
3567  72                        MOV   M,D ;      SET VALUE SIGN NEGATIVE
                         ;      ANALYZE NEXT CHARACTER IN STRING.
3568  CD E2 36           INP1:        CALL       CHAD ;  CALL CHAR ADDR SBRTN
356B  7E                        MOV   A,M ;      NEXT CHARACTER
356C  26 22                     MVI   H,SCRB ;   TO ADDRESS SCRATCH BANK
356E  06 00              INP2:        MVI        B,0 ;      DIGIT 2ND WD OR DEC EX
3570  FE FE                     CPI   376Q ;     COMPARE TO DECIMAL POINT
3572  CA AB 35                  JZ    INP3 ;     IF DECIMAL POINT
3575  FE 15                     CPI   025Q ;     COMPARE TO EXPONENT SIGN
3577  CA B5 35                  JZ    INP4 ;     IF EXPONENT SIGN
357A  FE 0A                     CPI   12Q ;      SET CARRY IF CHAR IS DIGIT
357C  D2 E6 35                  JNC   INP8 ;     IF CHAR IS NOT A DIGIT
357F  2E 3F                     MVI   L,TMP4 ;   TO ADDR CURRENT DIGIT
3581  77                        MOV   M,A ;      SAVE CURRENT DIGIT
3582  21 ED 36                  LXI   H,FTEN ;   TO ADDR FLOATING TEN
3585  CD 8C 32                  CALL  MUL ;      MULTIPLY BY TEN
3588  2E 3B                     MVI   L,VALE ;   TO ADDR VALUE
358A  CD 3E 32                  CALL  STR ;      STORE OLD VALUE TIMES TEN
358D  2C                        INR   L ;        TO ADDR CURRENT DIGIT
358E  7E                        MOV   A,M ;      CURRENT DIGIT
358F  06 00                     MVI   B,0 ;      CLEAR 2ND WORD OF DIGIT
3591  48                        MOV   C,B ;      CLEAR 3RD WORD OF DIGIT
3592  50                        MOV   D,B ;      CLEAR 4TH WORD OF DIGIT
3593  1E 08                     MVI   E,010Q ;   INDICATE DIGIT IS IN REG A
3595  CD 00 35                  CALL  FLT ;      CONVERT DIGIT TO FLOATING PNT
3598  2E 3B                     MVI   L,VALE ;   TO ADDR VALUE
359A  CD D7 32                  CALL  AD ;       ADD OLD VALUE TIMES TEN
359D  2E 39                     MVI   L,TMP2 ;   TO ADDR DEC PNT FLAG
359F  7E                        MOV   A,M ;      DECIMAL POINT FLAG
35A0  A7                        ANA   A          ;   SET CONTROL BITS
35A1  CA 68 35                  JZ    INP1 ;     IF NO DEC PNT ENCOUNTERED
35A4  2D                        DCR   L ;        TO ADDR INPUT EXPONENT
35A5  46                        MOV   B,M ;      INPUT EXPONENT
35A6  05                        DCR   B ;        DECREMENT INPUT EXPONENT
35A7  70                        MOV   M,B ;      UPDATE INPUT EXPONENT
35A8  C3 68 35                  JMP   INP1 ;     TO GET NEXT CHARACTER
```

```
ADDR  B1 B2 B3 B4 ERROR  SOURCE  INTEL 8080 ASSEMBLER  PAGE    4
35AB  2E 39              INP3:          MVI        L,TMP2 :   TO ADDR DEC PNT FLAG
35AD  AE                         XRA    M :         ZERO IF FLAG SET
35AE  77                         MOV    M,A :       SET DEC PNT FLAG
35AF  C2 68 35                   JNZ    INP1 :      IF FLAG NOT ALREAY SET
35B2  C3 E6 35                   JMP    INP8 :      IF 2ND DEC PNT
                         :       PROCESS DECIMAL EXPONENT.
35B5  CD E2 36           INP4:          CALL       CHAR :  CALL CHAR ADDR SBRTN
35B8  7E                         MOV    A,M :       NEXT CHARACTER OF STRING
35B9  47                         MOV    B,A :       CURRENT CHARACTER
35BA  D6 FD                      SUI    375Q :      COMPARE TO MINUS CHAR
35BC  5F                         MOV    E,A :       CHAR - MINUS SIGN
35BD  CA C6 35                   JZ     INP5 :      IF MINUS SIGN
35C0  C6 02                      ADI    2 :         COMPARE TO PLUS CHAR
35C2  78                         MOV    A,B :       CURRENT CHARACTER
35C3  C2 C8 35                   JNZ    INP6 :      IF NOT PLUS SIGN
35C6  2C                 INP5:          INR        L :   TO ADDRESS NEXT CHAR
35C7  7E                         MOV    A,M :       NEXT CHARACTER OF STRING
35C8  06 00              INP6:          MVI        B,0 :       POSSIBLE DEC EXPONENT
35CA  FE 0A                      CPI    12Q :       SET CARRY IF CHAR IS DIGIT
35CC  D2 E6 35                   JNC    INP8 :      IF CHAR IS NOT A DIGIT
35CF  47                         MOV    B,A :       DEC EXP EQUAL DIGIT
35D0  2C                         INR    L :         TO ADDRESS NEXT CHAR
35D1  7E                         MOV    A,M :       NEXT CHARACTER OF STRING
35D2  FE 0A                      CPI    12Q :       SET CARRY IF CHAR IS DIGIT
35D4  D2 DF 35                   JNC    INP7 :      IF CHAR IS NOT A DIGIT
                         :       FORM COMPLETE DECIMAL EXPONENT.
35D7  4F                         MOV    C,A :       LS DIGIT OF DEC EXP
35D8  78                         MOV    A,B :       MS DIGIT OF DEC EXP
35D9  87                         ADD    A :         2 * MS DIGIT
35DA  87                         ADD    A :         4 * MS DIGIT
35DB  80                         ADD    B :         5 * MS DIGIT
35DC  87                         ADD    A :         10 * MS DIGIT
35DD  81                         ADD    C :         10 * MS + LS DIGIT
35DE  47                         MOV    B,A :       DECIMAL EXPONENT
35DF  7B                 INP7:          MOV        A,E :       SIGN OF DEC EXPONENT
35E0  A7                         ANA    A           :     SET CONTROL BITS
35E1  C2 E6 35                   JNZ    INP8 :      IF SIGN PLUS
35E4  90                         SUB    B :         COMPLEMENT DEC EXP
35E5  47                         MOV    B,A :       DECIMAL EXPONENT
35E6  26 22              INP8:          MVI        H,SCRB :  TO ADDRESS SCRATCH BANK
35E8  2E 3A                      MVI    L,TMP3 :    TO ADDRESS INPUT SIGN
35EA  4E                         MOV    C,M :       INPUT SIGN
35EB  2E 31                      MVI    L,ACCS :    TO ADDRESS ACCUM SIGN
35ED  71                         MOV    M,C :       ACCUMULATOR SIGN
35EE  78                         MOV    A,B :       DECIMAL EXPONENT
                         :       CONVERT DECIMAL EXPONENT TO BINARY.
35EF  2E 38              INP9:          MVI        L,TMP1 :  TO ADDRESS DEC EXPONENT
35F1  86                         ADD    M :         ADJUST DECIMAL EXPONENT
35F2  CA 59 32                   JZ     TST :       IN DEC EXP IS ZERO
35F5  77                         MOV    M,A :       CURRENT DECIMAL EXPONENT
35F6  21 BD 36                   LXI    H,FTEN :    TO ADDR FLOATING TEN
35F9  F2 04 36                   JP     INP10 :     IF MULTIPLY REQUIRED
```

```
ADDR  B1 B2 B3 B4 ERROR   SOURCE  INTEL 8080 ASSEMBLER  PAGE   5
35FC  CD B4 32                    CALL   DIV ;       DIVIDE BY TEN
35FF  3E 01                       MVI    A,1 ;       TO INCREMENT DEC EXP
3601  C3 EF 35                    JMP    INP9 ;      TO TEST FOR COMPLETION
3604  CD 8C 32            INP10:  CALL   MUL ;       MULTIPLY BY TEN
3607  D8                          RC     ;           RETURN IF OVERFLOW
3608  3E FF                       MVI    A,377Q ;    TO DECREMENT DEC EXP
360A  C3 EF 35                    JMP    INP9 ;      TO TEST FOR COMPLETION
                          ;       OUT SUBROUTINE ENTRY POINT.
                          ;       SAVE CHARACTER ADDRESS AND ACCUMULATOR.
360D  2D                  OU:     DCR    L ;         DECREMENT CHARACTER ADDRESS
360E  CD D5 36                    CALL   SVAD ;      SET CHAR ADDR, DIG CNT, DEC EXP
3611  CD 59 32                    CALL   TST ;       LOAD ACCUM TO REGISTERS
3614  2E 3B                       MVI    L,VALE ;    TO ADDR ACCUM SAVE AREA
3616  CD 3E 32                    CALL   STR ;       CALL REG STR SUBROUTINE
                          ;       OUTPUT SIGN CHARACTER.
3619  CD E2 36                    CALL   CHAD ;      CALL CHAR ADDR SBRTN
361C  36 F0                       MVI    M,360Q ;    STORE SPACE CHARACTER
361E  A7                          ANA    A           ;   SET CONTROL BITS
361F  CA 3B 36                    JZ     OUT3 ;      IF ACCUMULATOR IS ZERO
3622  5F                          MOV    E,A ;       ACCUMULATOR EXPONENT
3623  78                          MOV    A,B ;       ACCUM SIGN AND 1ST FRCTN
3624  A7                          ANA    A           ;   SET CONTROL BITS
3625  7B                          MOV    A,E ;       ACCUMULATOR EXPONENT
3626  F2 2B 36                    JP     OUT1 ;      IF ACCUM IS POSITIVE
3629  36 FD                       MVI    M,375Q ;    CHANGE SIGN TO MINUS
                          ;       SCALE ACCUMULATOR TO .1 - 1. RANGE.
362B  FE 7E               OUT1:   CPI    176Q ;      COMPARE TO SMALL EXPON
362D  21 ED 36            OUT2:   LXI    H,FTEN ;    TO ADDR FLOATING TEN
3630  DA 45 36                    JC     OUT4 ;      IF EXPONENT TOO SMALL
3633  FE 81                       CPI    201Q ;      COMPARE TO LARGE EXP
3635  DA 50 36                    JC     OUT5 ;      IF EXP SMALL ENOUGH
3638  CD B4 32                    CALL   DIV ;       DIVIDE BY TEN
363B  26 22               OUT3:   MVI    H,SCRH ;    TO ADDRESS SCRATCH BAN
363D  2E 39                       MVI    L,TMP2 ;    TO ADDR DECIMAL EXPONENT
363F  5E                          MOV    E,M ;       DECIMAL EXPONENT
3640  1C                          INR    E ;         INCREMENT DECIMAL EXPONENT
3641  73                          MOV    M,E ;       DECIMAL EXPONENT
3642  C3 2D 36                    JMP    OUT2 ;      TO TEST FOR SCALING COMPLETE
3645  CD 8C 32            OUT4:   CALL   MUL ;       MULTIPLY BY TEN
3648  2E 39                       MVI    L,TMP2 ;    TO ADDR DECIMAL EXPONENT
364A  5E                          MOV    E,M ;       DECIMAL EXPONENT
364B  1D                          DCR    E ;         DECREMENT DECIMAL EXPONENT
364C  73                          MOV    M,E ;       DECIMAL EXPONENT
364D  C3 2B 36                    JMP    OUT1 ;      TO TEST FOR SCALING COMPLETE
                          ;       ROUND THE VALUE BY ADDING ,00000005.
3650  CD 50 32            OUT5:   CALL   ABS ;       SET ACCUM POSITIVE
3653  21 F1 36                    LXI    H,RNDO ;    TO ADDRESS ROUNDER
3656  CD D7 32                    CALL   AD  ;       ADD THE ROUNDER
3659  FE 81                       CPI    201Q ;      CHECK FOR OVERFLOW
365B  D2 2D 36                    JNC    OUT2 ;      IF EXP TOO LARGE
                          ;       SET DIGIT COUNTS.
365E  2E 39                       MVI    L,TMP2 ;    TO ADDR DECIMAL EXPONENT
```

```
                 ADDR  B1 B2 B3 B4 ERROR  SOURCE  INTEL 8080 ASSEMBLER  PAGE    6
00000281 SYSIN   3660  7E                         MOV    A,M ;        DECIMAL EXPONENT
00000282 SYSIN   3661  5F                         MOV    E,A ;        DIGITS BEFORE DEC POINT
00000283 SYSIN   3662  FE 08                      CPI    010Q ;       COMPARE TO LARGE EXP
00000284 SYSIN   3664  DA 69 36                   JC     OUT6 ;       IF EXPONENT IN RANGE
00000285 SYSIN   3667  1E 01                      MVI    E,1 ;        DIGITS BEFORE DEC POINT
00000286 SYSIN   3669  93                 OUT6:          SUB        E ;   ADJUST DEC EXPONENT
00000287 SYSIN   366A  77                         MOV    M,A ;        DECIMAL EXPONENT
00000288 SYSIN   366B  3E 07                      MVI    A,7 ;        TOTAL NUMBER OF DIGITS
00000289 SYSIN   366D  93                         SUB    E ;          DIGITS AFTER DECIMAL PNT
00000290 SYSIN   366E  2C                         INR    L ;          TO ADDR 2ND DIGIT CNT
00000291 SYSIN   366F  77                         MOV    M,A ;        DIGITS AFTER DECIMAL POINT
00000292 SYSIN   3670  1D                         DCR    E ;          DECREMENT DIGIT COUNT
00000293 SYSIN   3671  7B                         MOV    A,E ;        DIGITS BEFORE DEC PNT
00000294 SYSIN                            ;       OUTPUT SIGNIFICANT DIGITS.
00000295 SYSIN   3672  2E 38              OUT7:          MVI        L,TMP1 ;   TO ADDR DIGIT COUNT
00000296 SYSIN   3674  86                         ADD    M ;          ADJUST DIGIT COUNT
00000297 SYSIN   3675  77                         MOV    M,A ;        NEW DIGIT COUNT
00000298 SYSIN   3676  FA 93 36                   JM     OUT8 ;       IF COUNT RUN OUT
00000299 SYSIN   3679  21 ED 36                   LXI    H,FTEN ;     TO ADDR FLOATING TEN
00000300 SYSIN   367C  CD 8C 32                   CALL   MUL ;        MULTIPLY BY TEN
00000301 SYSIN   367F  1E 08                      MVI    E,10Q ;      TO PLACE DIGIT IN REG A
00000302 SYSIN   3681  CD 17 35                   CALL   FIX ;        CONVERT TO FIXED FORMAT
00000303 SYSIN   3684  CD E2 36                   CALL   CHAD ;       CALL CHAR ADDR SBRTN
00000304 SYSIN   3687  77                         MOV    M,A ;        OUTPUT DECIMAL DIGIT
00000305 SYSIN   3688  AF                         XRA    A ;          CLEAR CURRENT DIGIT
00000306 SYSIN   3689  1E 08                      MVI    E,010Q ;     BINARY SCALING FACTOR
00000307 SYSIN   368B  CD 00 35                   CALL   FLT ;        RESTORE VALUE MINUS DIGIT
00000308 SYSIN   368E  3E FF                      MVI    A,377Q ;     TO ADJUST DIGIT CNT
00000309 SYSIN   3690  C3 72 36                   JMP    OUT7 ;       LOOP FOR NEXT DIGIT
00000310 SYSIN   3693  2E 3A              OUT8:          MVI        L,TMP3 ;   TO ADDR 2ND DIGIT CNT
00000311 SYSIN   3695  7E                         MOV    A,M ;        DIGITS AFTER DECIMAL PNT
00000312 SYSIN   3696  36 FF                      MVI    M,377Q ;     SET 2ND COUNT NEG
00000313 SYSIN   3698  A7                         ANA    A            ;   SET CONTROL BITS
00000314 SYSIN   3699  FA A6 36                   JM     OUT9 ;       IF 2ND COUNT RAN OUT
00000315 SYSIN   369C  CD E2 36                   CALL   CHAD ;       CALL CHAR ADDR SBRTN
00000316 SYSIN   369F  36 FE                      MVI    M,376Q ;     STORE DECIMAL POINT
00000317 SYSIN   36A1  26 22                      MVI    H,SCRB ;     TO ADDRESS SCRATCH BANK
00000318 SYSIN   36A3  C3 72 36                   JMP    OUT7 ;       LOOP FOR NEXT DIGIT
00000319 SYSIN   36A6  2D                 OUT9:          DCR        L ;   TO ADDR DECIMAL EXP
00000320 SYSIN   36A7  A6                         ANA    M            ;   DECIMAL EXPONENT
00000321 SYSIN   36A8  CA CD 36                   JZ     OUT13 ;      IF DECIMAL EXPONENT IS ZERO
00000322 SYSIN                            ;       OUTPUT DECIMAL EXPONENT.
00000323 SYSIN   36AB  06 FB                      MVI    B,373Q ;     PLUS CHARACTER
00000324 SYSIN   36AD  F2 B5 36                   JP     OUT10 ;      IF EXPONENT IS POSITIVE
00000325 SYSIN   36B0  06 FD                      MVI    B,375Q ;     CHANGE SIGN TO MINUS
00000326 SYSIN   36B2  4F                         MOV    C,A ;        NEGATIVE EXPONENT
00000327 SYSIN   36B3  AF                         XRA    A ;          ZERO
00000328 SYSIN   36B4  91                         SUB    C ;          COMPLEMENT EXPONENT
00000329 SYSIN   36B5  0E FF              OUT10:         MVI        C,377Q ;   EMBRYO TENS DIGIT
00000330 SYSIN   36B7  57                 OUT11:         MOV        D,A ;      UNITS DIGIT
00000331 SYSIN   36B8  0C                         INR    C ;          INCREMENT TENS DIGIT
00000332 SYSIN   36B9  D6 0A                      SUI    012Q ;       REDUCE REMAINDER
```

RWLH FLOATING POINT MATH PACK-FORMAT CONVERSION ROUTINES

```
LINE  ADDR  B1 B2 B3 B4 ERROR  SOURCE  INTEL 8080 ASSEMBLER  PAGE   7
 313  36BB  D2 B7 36                   JNC     OUT11 ;      IF MORE TENS
 314  36BE  3E 15                      MVI     A,025Q ;     EXPONENT SIGN
 315  36C0  CD E2 36           OUT12:          CALL        CHAD ;     CALL CHAR ADDR SBRTN
 316  36C3  CD 3E 32                   CALL    STR ;        STORE LAST 4 CHARACTERS
 317  36C6  26 22                      MVI     H,SCRB ;     TO ADDRESS SCRATCH BANK
 318  36C8  2E 3B                      MVI     L,VALE ;     TO ADDRESS ACCUM SAVE AREA
 319  36CA  C3 6E 32                   JMP     LOD ;        RESTORE ACCUM AND EXIT
 320                           ;       OUTPUT 4 SPACES IF EXPONENT IS ZERO.
 321  36CD  3E F0              OUT13:          MVI         A,360Q ;   SPACE CHARACTER
 322  36CF  47                         MOV     B,A ;        SPACE CHARACTER
 323  36D0  4F                         MOV     C,A ;        SPACE CHARACTER
 324  36D1  57                         MOV     D,A ;        SPACE CHARACTER
 325  36D2  C3 C0 36                   JMP     OUT12 ;      TO STORE CHARACTERS
 326                           ;       SUBROUTINE TO SAVE CHARACTER STRING ADDR.
 327  36D5  7D                 SVAD:           MOV         A,L ;      CHARACTER STRING WORD
 328  36D6  44                         MOV     B,H ;        CHARACTER STRING BANK
 329  36D7  0E 00                      MVI     C,0 ;        INPUT EXP OR DIGIT CNT
 330  36D9  51                         MOV     D,C ;        DEC PNT FLAG OR DEC EXP
 331  36DA  26 22                      MVI     H,SCRB ;     TO ADDRESS SCRATCH BANK
 332  36DC  2E 36                      MVI     L,ADRL ;     TO ADDR CHAR STRING WORD
 333  36DE  CD 3E 32                   CALL    STR ;        STORE A, B, C, AND D
 334  36E1  C9                         RET       ;          RETURN TO CALLER
 335                           ;       SUBROUTINE TO OBTAIN NEXT CHARACTER ADDR.
 336  36E2  26 22              CHAD:           MVI         H,SCRB ;   TO ADDRESS SCRATCH BAN
 337  36E4  2E 36                      MVI     L,ADRL ; TO ADDR CHAR STRING WORD
 338  36E6  5E                         MOV     E,M ;        CHARACTER STRING WORD
 339  36E7  1C                         INR     E ;          TO ADDR NEXT CHARACTER
 340  36E8  73                         MOV     M,E ;        UPDATE CHAR STRING WORD
 341  36E9  2C                         INR     L ;          TO ADDR CHAR STRING BANK
 342  36EA  66                         MOV     H,M ;        CHARACTER STRING BANK
 343  36EB  6B                         MOV     L,E ;        CHARACTER STRING WORD
 344  36EC  C9                         RET       ;          RETURN TO CALLER
 345  36ED  84 20 00 00        FTEN:           DB          204Q,040Q,0,0 ; FLOATING TEN
 346  36F1  68 56 BF AD        RNDO:           DB          150Q,126Q,277Q,255Q ; .00000005
 347  36F5  00                         DB      0 ;          CHECKSUM WORD
 348  36F6                             END
TOTAL ASSEMBLER ERRORS =     0
```

```

                        ;MATH PACK TRANS ROUTINE
                        ; MOVES B TO H AND C TO L THEN JUMPS TO
                        ; MATH PACK
                        ;
                        ;
                        ;                                 RTW     3/9/76
                        ;
                        ;
0000 32 59              TST     EQU     3259H
0000 32 3E              STR     EQU     323EH
0000 32 6E              LOD     EQU     326EH
0000 32 D7              ADD0    EQU     32D7H
0000 32 D4              SUB0    EQU     32D4H
0000 32 8C              MUL     EQU     328CH
0000 32 B4              DIV     EQU     32B4H
0000 35 4B              INP     EQU     354BH
0000 36 0D              OUTPUT  EQU     360DH
                        ;
                        ;
31D5                            ORG     31D5H
                        ;
                        ;SET SO BUTS UP TO MATH PACK
                        ;
31D5 60                 STRT:   MOV     H,B
31D6 69                         MOV     L,C
31D7 E5                         PUSH    H
31D8 CD 59 32                   CALL    TST
31DB E1                         POP     H
31DC C3 3E 32                   JMP     STR
31DF 60                 LODT:   MOV     H,B
31E0 69                         MOV     L,C
31E1 C3 6E 32                   JMP     LOD
31E4 60                 ADDT:   MOV     H,B
31E5 69                         MOV     L,C
31E6 C3 D7 32                   JMP     ADD0
31E9 60                 SUBT:   MOV     H,B
31EA 69                         MOV     L,C
31EB C3 D4 32                   JMP     SUB0
31EE 60                 MULT:   MOV     H,B
31EF 69                         MOV     L,C
31F0 C3 8C 32                   JMP     MUL
31F3 60                 DIVT:   MOV     H,B
31F4 69                         MOV     L,C
31F5 C3 B4 32                   JMP     DIV
31F8 60                 INPT:   MOV     H,B
31F9 69                         MOV     L,C
31FA C3 4B 35                   JMP     INP
31FD 60                 OUTT:   MOV     H,B
31FE 69                         MOV     L,C
31FF C3 0D 36                   JMP     OUTPUT
3202 00                         END
```

TOTAL ASSEMBLER ERRORS = 0

SYMBOL TABLE (600 MAX. 27 USED)

1	A	0007	ADD0	32D7	ADDT	31E4	B	0000	C	0
6	D	0002	DIV	32B4	DIVT	31F3	E	0003	H	0
11	INP	354B	INPT	31F8	L	0005	LOD	326E	LODT	3
16	M	0006	MUL	328C	MULT	31EE	OUTPUT	360D	OUTT	3
21	PSW	0006	SP	0006	STR	323E	STRT	31D5	SUB0	3.
26	SUBT	31E9	TST	3259						

```
        /* RWLM PROGRAM */
     /* BY W.P. MCDOWELL  AND D.J.KEEFE  AND R.T.WITEK*/
     /*FOR BOM CONTRACT # HO 252053          */
     /*ARGONNE NATIONAL LABORATORY*/
      /* NOVEMBER 8,1976*/
      /*COPYRIGHT 1976*/
         /*V11*/
 /*THIS PROGRAM REQUIRES THE USE OF THE INTEL PLM COMPILER TO
   GENERATE 8080 CODE-USE THE FOLLOWING COMPILER SWITCH SETTINGS
   $*=9215(SETS STACK POINTER TO 23FFH FOR 1K RAM,$V=32-RAM=2000H,
   $H=256 ORG FOR PROPER MATHPACK OPERATION,$P,$G,$B=6,$O=7 */
        /* DATA DECLARATIONS*/
DECLARE LIT LITERALLY 'LITERALLY';
DECLARE SP16 LIT '0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,24H';
DECLARE MASK DATA(1,2,4,8,16,32,64,128,255,0);
/* 255 AND 0 IN ABOVE MASK ARE DUMMIES INSERTED TO PREVENT
 CONSTANT DATA WHICH FOLLOWS FROM OVERLAPING A MEMORY PAGE
 THE MATH-PACK REQUIRES THAT REG H NOT CHANGE IN THE
 ADDRESS OF A VARIABLE          */
DECLARE CONSTF DATA(SP16,SP16,SP16,SP16,SP16,SP16,SP16,
SP16,SP16,SP16,SP16,SP16,SP16,SP16,SP16,
SP16,SP16,SP16,SP16,SP16,SP16,SP16,SP16,SP16);
DECLARE TRUE LITERALLY '1111$1111B';
DECLARE FALSE LITERALLY '0';
DECLARE QM LITERALLY '3FH';
DECLARE OUT BYTE;
DECLARE BIT$07(8) BYTE; DECLARE BIT$04(8) BYTE;
DECLARE (NET1,NET2,NET3)(13) BYTE;
DECLARE (B0,B1,B2,D0,D1,D2) (8)  BYTE;
DECLARE LF LITERALLY '0AH', CR LITERALLY '0DH',BELL LITERALLY '7';
DECLARE (FAILTST,SELDET)BYTE;
DECLARE DETSEL(2) BYTE;
DECLARE ( TIMERMINS,TIMERHRS,TIT) BYTE;
DECLARE (TIMERSECS) ADDRESS;
DECLARE (TIM,TIH,SMIN,SHRS) BYTE;
DECLARE  HEAD(2) BYTE;
DECLARE (SAMPLE$HOURS,SAMPLE$MINS )  BYTE;
DECLARE ( DETHRS,DETMINS) (3) BYTE;
DECLARE  RWLM$HEAD LITERALLY '7';
DECLARE (YRS,MONS,DAYS,HRS ,MINS,SECS) BYTE ;
DECLARE (I,N,M,TT) BYTE;
DECLARE (TSTPWR,TSTSOL)BYTE;
DECLARE (TSTART,TSTOP) ADDRESS;
DECLARE TTYSTT LITERALLY '0',TTYBUF LITERALLY '1';
DECLARE DATEOK DATA ('DATE CORRECT (Y/N)',QM);
DECLARE WISH DATA('DO YOU WANT ');
DECLARE WISH2 DATA(' RUN(Y/N)',QM);
DECLARE STOPTIME DATA ('INPUT TIME TO STOP');
DECLARE TQH DATA ('ENTER TIME INTERVAL BETWEEN SAMPLES HOURS-');
DECLARE TQM DATA (' MIN-');
DECLARE SHORT DATA(CR,LF,'TIME TOO SHORT MUST BE >14 MINS',CR,LF);
DECLARE STH DATA('ENTER TIME TO START HOUR-');
DECLARE STMM DATA ('  MIN-');
DECLARE ALT$COR$FACTOR(13) BYTE;
DECLARE WL(52) BYTE;
DECLARE (YS,MNS,DS,HS,MS) BYTE;
DECLARE (CAL$TEST,CAL$TEST1,CALTEST2,CAL$TIME)  BYTE;
DECLARE (ALTFACTOR1,ALTFACTOR2) (13) BYTE;
```

```
      /*PROCEDURES*/
 TYPEOUT: PROCEDURE(CHAR); /* TYPE A CHARACTER */
  DECLARE (CHAR,I) BYTE;
   DO WHILE((INPUT(TTYSTT) AND 80H)=FALSE); END; /* WAIT FOR TTY */
   OUTPUT(TTYBUF)=CHAR;/* PRINT THE CHAR*/
   IF CHAR=CR THEN /* WAIT ON CR FOR S700*/
        DO I=0 TO 8; CALL TIME(250); END;/*200 MSEC WAIT*/
  END TYPEOUT;/*THATS ALL*/

  TYPEIN:  PROCEDURE BYTE ;/* INPUT A CHAR FROM TTY */
    DO WHILE((INPUT(TTYSTT) AND 1)=FALSE);END;/*WAIT FOR DATA*/
     RETURN INPUT(TTYBUF);/*GET THE CHAR*/
   END TYPEIN;

TYINOUT:PROCEDURE; /*ECHOS A CHAR TO TTY */
    N=TYPEIN AND 7FH;
    CALL TYPEOUT(N);
    END TYINOUT;

  ECHO:PROCEDURE  BYTE;
  CALL TYINOUT;RETURN(N);
  END ECHO;

DECLARE CNT BYTE;

  INP$LINE:PROCEDURE(P);
  DECLARE P ADDRESS,(T,C BASED P)BYTE;
  CNT=0;
  DO WHILE((T:=ECHO)<>CR);
  C(CNT)=T;CNT=CNT+1; END;
 C(CNT)='$'-'0';
  CALL TYPEOUT(LF);
  END INP$LINE;

  CRLF: PROCEDURE; /* OUTPUTS A CR-LF TO THE TTY*/
   CALL TYPEOUT(0DH);
   CALL TYPEOUT(0AH);
  END CRLF;

  PRINTPROMPT:PROCEDURE; /*PRINTS PRCMPT CHAR*/
   CALL CRLF;
   CALL TYPEOUT('*');
  END PRINTPPOMPT;

PACKBCD: PROCEDURE; /*PRODUCES A PACKED BCD BYTE*/
   CALL TYINOUT;
   N=N-'0';
       CALL TYINOUT;
   N=SHL(N,4)+(N-'0');
   END PACKBCD;
```

```
 PRINTSTRING:PROCEDURE(NAME, LENGTH); /* TYPES A DATA STRING*/
   DECLARE NAME ADDRESS,(LENGTH,I,CHAR BASED NAME) BYTE;
   DO I=0 TO LENGTH-1;
   CALL TYPEOUT(CHAR(I));
    END;
 END PRINTSTRING;

  PRINTBCD: PROCEDURE(B) ; /*PRINT BCD CHAR PAIR*/
   DECLARE B BYTE;
   CALL TYPEOUT(SHR(B,4)+'0');
   CALL TYPEOUT((B AND 0FH)+'0');
  END PRINTBCD;

  ERROR:   PROCEDURE;
   /*  ERROR RESTART PROCEDURE RESTARTS PROGRAM
       AFTER ANY ERROR*/
  DECLARE ERRMSG DATA (CR,LF,BELL,BELL,
     '**  E R R O R  **',BELL,BELL,CR,LF,LF);
        CALL PRINTSTRING(.ERRMSG,LENGTH(ERRMSG));
        GOTO START;
     END ERROR;

DELAY$1MS: PROCEDURE;  /*1 MS DELAY */
     DISABLE;
     CALL TIME(10);
     ENABLE;
  END DELAY$1MS;

DELAY$5MS:PROCEDURE;  /*5 MS.DELAY*/
     DISABLE;
    CALL TIME(48);
    ENABLE;
    END DELAY$5MS;

  TYPEDATE: PROCEDURE; /*OUTPUTS CLOCK DATA*/
   DECLARE DATE  DATA(CR,LF,'TODAY IS ');
   DECLARE TIME DATA('  THE TIME IS ');
   CALL PRINTSTRING(.DATE,LENGTH(DATE));
   CALL PRINTBCD(MONS);
   CALL TYPEOUT('/');
   CALL PRINTBCD(DAYS);
   CALL TYPEOUT('/');
   CALL PRINTBCD(YRS);
   CALL PRINTSTRING(.TIME,LENGTH(TIME));
   CALL PRINTBCD(HRS);
   CALL TYPEOUT(':');
   CALL PRINTBCD(MINS);
   CALL CRLF;
  END TYPEDATE;
```

```
/* SET DATE AND TIME FROM KEYBOARD*/
   DECLARE SETYEAR DATA(CR,LF,'ENTER YEAR-');
   DECLARE SETMON DATA(' ENTER MONTH-');
   DECLARE SETDAY  DATA(' ENTER DAY -');
   DECLARE SETHR   DATA(' ENTER HOUR-');
   DECLARE SETMIN  DATA(' ENTER MINUTE-');
   CHANGEDATE: PROCEDURE;
  DISABLE;
 CALL PRINTSTRING(.SETYEAR,LENGTH(SETYEAR));
  CALL PACKBCD;
   YRS=M;
CALL PRINTSTRING(.SETMON,LENGTH(SETMON));
  CALL PACKBCD;
   MONS=M;
 CALL PRINTSTRING(.SETDAY,LENGTH(SETDAY));
    CALL PACKBCD;
   DAYS=M;
CALL PRINTSTRING(.SETHR,LENGTH(SETHR));
  CALL PACKBCD;
   HRS=M;
CALL PRINTSTRING(.SETMIN,LENGTH(SETMIN));
  CALL PACKBCD;
   MINS=M;
  CALL CRLF;
       SECS=0;
        OUTPUT(OFBH)=0;
   ENABLE;
   CALL TYPEDATE;
END CHANGEDATE;

     RESTART: PROCEDURE INTERRUPT 0;
      /* STARTUP AND RESTART PROCEDURE INT 0*/
       GO TO START;/* ALL YOU DO ON INT 0 IS GO TO START*/
       END RESTART;
```

```
        TIMEKEEPER: PROCEDURE INTERRUPT 7;
/*MACHINE IS INTERRUPTED ONCE PER SEC TO UPDATE CLOCK*/
/* DAYMAX IS A VECTOR FOR LENGTH OF MONTH */
  DECLARE DAYMAX DATA(0,31H,28H,31H,30H,31H,30H,31H,31H,30H,
   0,0,0,0,0,0,31H,30H,31H);
   TIMERSECS=TIMERSECS+1;
SECS=SECS OR SECS;/*RESETS CARRY BIT TO ACCOUNT FOR COMPILER BUG*/
  IF(SECS := DEC(SECS+1))=60H THEN    /*INCR SECONDS*/
  DO;
    SECS=0;
     SECS=SECS OR SECS;  /*SEE ABOVE*/
      TIMERMINS=DEC( TIMERMINS+1);  /*EXTERNAL TIMER*/
          IF TIMERMINS=60H THEN
  DO;TIMERMINS=0;SECS=SECS OR SECS;TIMERHRS=DEC(TIMER$HRS+1);END;
      MINS=MINS OR MINS;  /*SEE ABOVE*/
    IF (MINS:=DEC(MINS+1))=60H THEN   /*INCR MINS*/
    DO;
       MINS=0;
      HRS=HRS OR HRS;  /*SEE ABOVE*/
       IF(HRS:=DEC(HRS+1))=24H THEN   /*INCR HRS*/
       DO;
       HRS=0;
/*NOW INCR DAYS AND THEN CHECK FOR LENGTH OF MONTH */
      DAYS=DAYS OR DAYS;  /*SEE ABOVE*/
        IF (DAYS:=DEC(DAYS+1)) > DAYMAX(MONS)
   THEN DO;
         DAYS=1;
         DAYS=DAYS OR DAYS;
          IF(MONS:=DEC(MONS+1))=13H THEN
           DO; YRS=DEC(YRS+1);
           MONS=1;
           END;
         END;
        END;
      END;
   END;
  OUTPUT(0FBH)=0;
 END TIMEKEEPER;
```

```
   DETMSG: PROCEDURE; /*PRINTS DET NUMBER*/
   DECLARE COMMSG DATA(' DETECTOR  ');
    CALL PRINTSTRING(.COMMSG,LENGTH(COMMSG));
      CALL TYPEOUT(SELDET + '1');
    END DETMSG;

  /* CHAN 7 OUTPUT PROCEDURE  */
  /* OUTPUT 7 CONTROLS THE RWLM HEAD*/
  /* BIT 0 HEAD SELECT-0=HEAD 1,1=HEAD 2 */
   /*BIT 1                       */
  /*BIT 2                      */
  /*BIY 3 MOTOR PULSES         */
  /*BIT 4 MOTOR POWER          */
  /*BIT 5 SOLENOID POWER       */
  /*BIT 6 PUMP ON              */
  /*BIT 7 SOURCE CHECK                        */
   DET$HEAD$OUT:   PROCEDURE;
   DECLARE I BYTE;
      OUT=FALSE;
      DO I=0 TO 7;
      CUT=OUT OR (BIT$07(I) AND MASK(I));
      END;
   OUTPUT(7)=OUT;
   END DET$HEAD$OUT;

    /* CHAN 4 OUTPUT  PROCEDURE */
/* OUTPUT 4 CONTROLS THE DATA ACCUMULATORS  */
 /* BIT 0  STORE(NOT)                */
 /* BIT 1 SET(NOT)                   */
 /* BIT 2 SCAN                       */
 /* BIT 3 CLEAR(NOT)                 */
 /* BIT 4 COUNT ENABLE               */
 /* BIT 5                            */
 /* BIT 6                            */
 /* BIT 7                            */
    CHN$4$OUT:  PROCEDURE;
    DECLARE I BYTE;
         OUT=FALSE;
         DO I=0 TO 7;
         OUT=OUT OR (BIT$04(I) AND MASK(I));
         END;
    OUTPUT(4)=OUT;
    END CHN$4$OUT;

   RELEASE$SOLENOID: PROCEDURE;
   DECLARE SOLMSG DATA(' SOLENOID NOT RELEASED');
      FAILTST=FALSE;
         BIT$07(5)=FALSE;
         CALL DET$HEAD$OUT;
         IF (INPUT(RWLM$HEAD) AND MASK(5))=FALSE THEN
           DO;
           CALL DETMSG;
           CALL PRINTSTRING(.SOLMSG,LENGTH(SOLMSG));
           FAILTST=TRUE; CALL ERROR;
           END;
   END RELEASE$SOLENOID;
```

```
/*PAPADV MOVES FILTER 1 CYCLE,CHECKS 0 POS AND FILTER QUANT*/
 PAPADV:PROCEDURE;
      DECLARE PULSES ADDRESS;
      DECLARE FINISH LABEL;
      DECLARE NOPAPER DATA( ' FILTER SUPPLY EXHAUSTED',CR,LF);
      DECLARE EXTRA$ADVANCE DATA(' FILTER ADVANCE ERROR',CR,LF);
       FAILTST=FALSE;
       BIT$07(4)=MASK(4);
       CALL RELEASE$SOLENOID;
       IF FAILTST=0 THEN
          DO PULSES = 0 TO 1150;
          /*TURN ON SOLENOID WHEN STARTING PAPER TAKE UP*/
          IF PULSES > 500 THEN BIT$07(5)=TRUE;
          /*START FORMING MOTOR PULSE*/
          BIT$07(3)=FALSE;
          CALL DET$HEAD$OUT;
          CALL DELAY$1MS;
          BIT$07(3)=MASK(3);
          CALL DET$HEAD$OUT;
          CALL DELAY$5MS;
IF ((INPUT(RWLM$HEAD)AND MASK(7))=MASK(7) AND PULSES>100 ) THEN
             DO;
             IF(INPUT(RWLM$HEAD) AND MASK(4))= 0 THEN
                DO; CALL DETMSG;
                CALL PRINTSTRING(.NOPAPER,LENGTH(NOPAPER));
                FAILTST=TRUE;/* SET ERROR FLAG */
                END;
             GO TO FINISH;/*GO HOME NICELY*/
             END;
          END;
          /*IF HERE THERE HAS BEEN AN ERROR*/
      CALL DETMSG;
      CALL PRINTSTRING(.EXTRA$ADVANCE,LENGTH(EXTRA$ADVANCE));
      FAILTST=TRUE;
FINISH:BIT$07(4)=FALSE;/*TURN OFF POWER */
      BIT$07(5)=FALSE;/*AND THE SOLENOID*/
       CALL DET$HEAD$OUT;/*DO IT NOW*/
         IF FAILTST=TRUE THEN CALL ERROR;/*ERROR IF BADNESS SET*/
      END  PAPADV;/*ELSE GO BACK TO WERE YOU CAME FROM*/
```

```
 /* THIS SECTION INTERFACES PLM TO THE MATH PACK*/
 DECLARE ADDT LIT '31E2H';
 DECLARE SUBT LIT '31E7H';
 DECLARE INITT LIT '322FH';
 DECLARE MULT LIT '31ECH';
 DECLARE DIVT LIT '31F1H';
 DECLARE INPT LIT '31F6H';
 DECLARE OUTT LIT '31FBH';
 DECLARE STRT LIT '31D3H';
 DECLARE LODT LIT '31DDH';
 INIT:    PROCEDURE ;
 GOTO INITT;
 END INIT;
 ADD:    PROCEDURE(LOC);
         DECLARE LOC ADDRESS;
         GOTO ADDT;
 END ADD;
 SUB:    PROCEDURE(LOC);
         DECLARE LOC ADDRESS;
         GOTO SUBT;
         END SUB;
 MUL:    PROCEDURE(LOC);
         DECLARE LOC ADDRESS;
         GO TO MULT;
         END MUL;
 DIV:    PROCEDURE(LOC);
         DECLARE LOC ADDRESS;
         GOTO DIVT;
 END DIV;
 LOD:    PROCEDURE(LOC);
         DECLARE LOC ADDRESS;
         GO TO LODT;
 END LOD;
 STR:    PROCEDURE(LOC);
         DECLARE LOC ADDRESS;
         GOTO STRT;
 END STR;
 INP:    PROCEDURE(LOC);
         DECLARE LOC ADDRESS;
         GOTO INPT;
 END INP;
 OUTPT:   PROCEDURE(LOC);
          DECLARE LOC ADDRESS;
          GO TO OUTT;
 END OUTPT;
   INSTR: PROCEDURE(A1,A2);
     DECLARE (A1,A2) ADDRESS;
     CALL INP(A1); CALL STR(A2);
   END INSTR;
/* THIS END THE MATH PACK -PLM INTERFACE*/
```

```
      DECLARE TEMP(48) BYTE;
      DECLARE  (ASUB,FSUB,TSUB) ADDRESS;
    DECLARE ALTCONF(4) BYTE;
      DECLARE  FT(12) BYTE,FB(24) BYTE, FCON(48) BYTE;

 NETSCOUNT:PROCEDURE; /*SUBTRACTS BACKGROUND FROM SAMPLE*/
      DO I=0 TO 2;
      ASUB=.FB+I*4;FSUB=ASUB+12;TSUB=.FT+I*4;
      CALL LOD(FSUB);CALL SUB(ASUB);CALL STR(TSUB);
        CALL OUTPT(.NET1+I*13);
      END;
   END NETSCOUNT;

CAL:   PROCEDURE; /* PERFORMES CALCULATIONS*/
      DECLARE (I,I1) BYTE;
      CALL INIT;
/* CONVERT CONSTANT VALUES TO FLOATING POINT AND STORE*/
      DO I=0 TO 11;
      CALL INSTR(.CONSTR+(I*16)+(192*SELDET),.FCON+(I*4));
      END;
/* CONVERT ALTITUDE CORRECTION FACTOR TO FP AND STORE*/
         /* DETERMINE WHICH HEAD IS SELECTED*/
        IF SELDET=0 THEN
         DO I=0 TO 13;ALT$COR$FACTOR(I)=ALTFACTOR1(I);END;
      ELSE
         DO I=0 TO 13;ALT$COR$FACTOR(I)=ALTFACTOR2(I);END;
      CALL INSTR(.ALT$COR$FACTOR,.ALTCONF);
/*CONVERT BACKGROUND AND SAMPLE COUNTS TO FP AND STORE*/
      DO I=0 TO 5;
      CALL INSTR(.B0+I*8,.FB+I*4);
      END;
/* SUBTRACT BACKGROUND FROM SAMPLE AND STORE*/
    CALL NETSCOUNT;
  /* PERFORM MULTIPLICATIONS AND STORE RESULTS IN TEMP*/
      DO I1=0 TO 3 ;
         DO I=0 TO 8 BY 4;
         CALL LOD(.FT+I);
         CALL MUL(.FCON+I+(I1*12));
         CALL STR(.TEMP+I+(I1*12));
         END;
      CALL LOD(.TEMP+(I1*12));
         DO I=4 TO 8 BY 4;
/* PERFORM ADDITIONS*/
         CALL ADD(.TEMP+I+(I1*12));
         END;
/* MULTIPLY BY ALTITUDE CORRECTION FACTOR*/
      CALL MUL(.ALTCONF);
/*STORE RESULTS*/
      CALL OUTPT(.WL+(I1*13));
      END;
END CAL;
```

```
/*THIS SECTION PRINTS THE REPORT ON THE TERMINAL*/
DECLARE WLM DATA ('WORKING LEVEL          RADIUM',
' A                    RADIUM B                    RADIUM C',
'                ');
DECLARE L DATA('      PC/L PC/L PC/L');
DECLARE HEAD1 DATA (CR,LF,LF,'                         '
,'       *** R W L M ***',CR,LF,'                         ');
DECLARE CL DATA('RADIUM A        RADIUM (B+C)    RADIUM C''  ');
DECLARE CH DATA('                  SAMPLE COUNTS',
'          BACKGROUND COUNTS     NET COUNTS',CR,LF);
 DECLARE ALTMSG DATA(CR,LF,'ENTER ALTITUDE CORRECTION FACTOR= ');

PRINT$REPORT:  PROCEDURE;
DECLARE I BYTE;
   /*CONVERT COUNT DATA FROM BCD TO ASCII*/
DO I=0 TO 47 ; B0(I)=B0(I)+'0'; END;
DO I=0 TO 38;NET1(I)=NET1(I)+'0';END;
    /*CHECK IF CALIBRATION REPORT*/
   IF CAL$TEST2=TRUE THEN GOTO CAL$PRINT;
     DO I=0 TO 51; WL(I)=WL(I)+'0';END;
   CALL PRINTSTRING(.HEAD1,LENGTH(HEAD1));
     CALL DETMSG;
   CALL CRLF;
CALL TYPEDATE;
CALL PRINTSTRING(.ALTMSG+8,27);
 DO I=0 TO CNT-1; N=ALT$COR$FACTOR(I)+30H;CALL TYPEOUT(N);END;
CALL CRLF;CALL CRLF;
   DO I=0 TO 3;
   CALL PRINTSTRING(.WLM+I*24,24);
   CALL PRINTSTRING(.WL+I*13,13);
   CALL PRINTSTRING(.L+I*5,5);
   CALL CRLF;
   END;
   CALL CRLF;CALL CRLF;CALL CRLF;
    /*PRINT CALIBRATION REPORT*/
  CAL$PRINT:;
   CALL PRINTSTRING(.CH,LENGTH(CH));
   DO I=0 TO 2;
   CALL PRINTSTRING(.CL+I*16,16);
   CALL PRINTSTRING(.B0+I*8+24,7);
   CALL PRINTSTRING(.HEAD1+3,16);
   CALL PRINTSTRING(.B0+I*8,7);
   CALL PRINTSTRING(.HEAD1+3,16);
    CALL PRINTSTRING(.NET1+I*13,13);
   CALL CRLF;
   END;
   CALL CRLF;CALL CRLF;
      /* RETURN DATA TO ORIGINAL CONDITION*/
  /* RECONVERT TO BCD*/
DO I=0 TO 47; B0(I)=B0(I)-'0';END;
DO I=0 TO 38;NET1(I)=NET1(I)-'0';END;
DO I=0 TO 51;WL(I)=WL(I)-'0';END;
   END PRINTREPORT;
```

```
     COUNT: PROCEDURE;
     /*  TURN COUNTERS ON AT TSTART AND OFF
         AT TSTOP , CLEAR BEFORE COUNT */
  IF CALSTEST1=TRUE THEN GO TO CALSLOOP;
          BIT$O4(3)=FALSE; BIT$O4(4)=FALSE;CALL CHN$4$OUT;
          BIT$O4(3)=MASK(3);BIT$O4(4)=MASK(4);
          DO WHILE (TIMER$SECS<TSTART); END;
          CALL CHN$4$OUT;
     /* FOR CALIBRATION ONLY */
  CAL$LOOP:;
          BIT$O4(4)=FALSE;
 DECLARE TS ADDRESS;
STARTTIM:  DISABLE;
    TS= TIMER$SECS;   ENABLE;
  IF TSTOP>TS THEN GOTO STARTTIM;
          CALL CHN$4$OUT;
     END COUNT;
```

```
     READ$ACCUM: PROCEDURE(A,B,C);
     DECLARE I BYTE;
/* READ ACCUM PROCEDURE
WILL READ DATA FROM SELECTED HEAD AND RETURN DATA */
DECLARE (A,B,C) ADDRESS, (R1 BASED A,R2 BASED B, R3 BASED C) BYTE;
          /* LATCH LSD*/
     BIT$O4(0)=MASK(0);CALL CHN$4$OUT;
     BIT$O4(0)=FALSE; CALL CHN$4$OUT;
          /*INPUT LSD AND RIGHT JUSTIFY */
     R1(6)=SHR(INPUT(4),4);
     R2(6)=SHR(INPUT(5),4);
     R3(6)=SHR(INPUT(6),4);
          /* SET THE COUNTER DISPLAY COUNTER*/
     BIT$O4(1)=MASK(1); CALL CHN$4$OUT;
     BIT$O4(1)=FALSE;   CALL CHN$4$OUT;
          /* THIS LOOP READS THE SIX DIGITS FROM THE COUNTERS*/
        DO I=0 TO 5;
          /*LATCH THE DIGIT */
        BIT$O4(0)=MASK(0) ;CALL CHN$4$OUT;
        BIT$O4(0)=FALSE ;CALL CHN$4$OUT;
          /*INPUT DIGIT AND STRIP OFF LSD*/
        R1(I)=NOT(INPUT(4)) AND 15;
        R2(I)=NOT(INPUT(5)) AND 15;
        R3(I)=NOT(INPUT(6)) AND 15;
          /*STROBE THE SCAN INPUT HAS TO BE SLOW FOR THE CMOS*/
        BIT$O4(2)=MASK(2);CALL CHN$4$OUT;
     CALL DELAY$1MS;
        BIT$O4(2)=FALSE; CALL CHN$4$OUT;
     CALL DELAY$1MS;
     END;
          /* PUT $-0 ON THE END OF EACH NUMBER FOR THE MATH PACK*/
     R1(7)='$'-'0';R2(7)='$'-'0';R3(7)='$'-'0';
     END READ$ACCUM;

  DECLARE FLO$MSG DATA(' FLOW RATE OUT OF RANGE ');
  FLOW$CHECK: PROCEDURE;
      IF (INPUT(RWLM$HEAD) AND MASK(3))=0 THEN
      DO;
        CALL DET$HEAD$OUT;    /*STOP PUMP*/
        CALL DETMSG;
        CALL PRINTSTRING(.FLO$MSG,LENGTH(FLO$MSG));
        CALL ERROR;
      END;
  END FLOW$CHECK;
```

```
TEST$AC$POWER:  PROCEDURE;
 DECLARE PWRMSG DATA(' POWER FAIL',CR,LF);
   FAILTST=FALSE;
   IF (INPUT(RWLM$HEAD) AND MASK(6)) =0 THEN
    DO;
   CALL DETMSG;
  CALL PRINTSTRING(.PWRMSG,LENGTH(PWRMSG));   /* POWER FAILURE*/
  FAILTST = TRUE;
    END;
   END TEST$AC$POWER ;

   SAMPLE:  PROCEDURE;
   /* SAMPLE PROCEDURE  TAKES SAMPLE FROM HEAD
       HEAD SCHOULD BE SELECTED BEFORE CALL*/
  /* MAKE SURE THE POWER IS ON SO THAT THINGS WILL RUN*/
 CALL TEST$AC$POWER;IF FAILTST THEN CALL ERROR;
   /* TOP OUT THE PAPER BEFORE YOU START TO WORK*/
   /* CHECK FOR ZERO POSITION*/
IF (INPUT(RWLM$HEAD) AND MASK(7)) = FALSE THEN CALL PAPADV;
   CALL PAPADV; CALL PAPADV;
   /* CLEAR OUT PAPER */
   TIMER$SECS=0;/* SET REAL TIME TO ZERO*/
        /* TAKE BACKGROUND COUNT FROM 1 SEC TO 121 SEC*/
   TSTART=1;TSTOP=121;CALL COUNT;
        /* READ IN BACKGROOUND COUNT*/
   CALL READ$ACCUM(.B0,.B1,.B2);
        /* SET TO START PUMP IN OUTPUT VECTOR*/
   BIT$07(5)=TRUE;BIT$07(6)=TRUE;
          /* WAIT TILL TIME TO START PUMP*/
   DO WHILE (TIMER$SECS<122);END;
        /* START PUMP*/
   CALL DET$HEAD$OUT;BIT$07(6)=FALSE;
       /*TEST FLOW RATE AT 200 SEC*/
    DO WHILE (TIMER$SECS<200);  END;
    CALL FLOW$CHECK;
        /* LET PUMP RUN FOR 2 MIN*/
   DO WHILE (TIMER$SECS<242); END;
        /* STOP PUMP*/
   CALL DET$HEAD$OUT;
        /* MOVE SAMPLE UNDER HEAD*/
   CALL PAPADV;
        /*SET UP TO COUNT SAMPLE*/
   TSTART=256;TSTOP=376;CALL COUNT;
        /*READ DATA FOR SAMPLE*/
   CALL READ$ACCUM(.D0,.D1,.D2);
 /*WHEN CALIBRATING JUMP BACK TO CALIBRATE PROCEDURE*/
  IF CAL$TEST=TRUE THEN GOTO FINISH;
        /* DO MATH AND PRINT IT OUT*/
   CALL CAL; CALL PRINT$REPORT;
        /* GO ON HOME NOW ALL IS DONE*/
FINISH:;
   END SAMPLE;
```

```
CYCLE:     PROCEDURE;
   CAL$TEST=FALSE;  /*NORMAL RUN*/
     /* WAIT TILL TIME TO START*/
      DO WHILE (SMIN<>MINS OR SHRS<>HRS);END;
      /*ONCE IN INTERRUPT ONLY WAY OUT */
   TT=FALSE;
      DO WHILE NOT(TT);/*MAKE A STATMENT AFTER TEST*/
      /* CLEAR MIN AND HRS TIMER*/
      TIMER$MINS=0; TIMER$HRS=0;
      /*HEAD 0 SELECT*/
      IF HEAD(0)   THEN
           DO;
           BIT$07(0)=FALSE;SELDET=0;CALL DET$HEAD$OUT;CALL SAMPLE;
      DO WHILE (TIMER$MINS<07H);END;
           END;
      IF HEAD(1)  THEN
           DO;
           BIT$07(0)=TRUE;SELDET=1;CALL DET$HEAD$OUT;CALL SAMPLE;
           END;
      DO WHILE NOT(TIMER$MINS=TIM AND TIMER$HRS=TIH);END;
TT=(YRS>=YS AND MONS>=MNS AND DAYS>=DS AND HRS>=HS AND MINS>=MS);
   END;
      END CYCLE; /* THATS ALL THERE IS*/
```

```
MAIN:      PROCEDURE;/* THE ONE THAT STARTS IT OFF*/
  CALL TYPEDATE;CALL PRINTSTRING(.DATEOK,LENGTH(DATEOK));
       /*IF DATE OK THEN JUST TURN ON CLOCK ELSE CHANGE DATE*/
  /*PRINT DATE AND SEE IF IT IS OK*/
  CALL TYINOUT;IF N='N' THEN CALL CHANGEDATE;
       ELSE DO;OUTPUT(OFBH)=0;ENABLE;CALL CRLF;END;
       DO SELDET=0 TO 1; /*ASK FOR EACH HEAD*/
       CALL PRINTSTRING(.WISH,LENGTH(WISH));CALL DETMSG;
       CALL PRINTSTRING(.WISH2,LENGTH(WISH2)); CALL TYINOUT;
       IF N='Y' THEN HEAD(SELDET)=TRUE;
           ELSE HEAD(SELDET)=FALSE;
       CALL CRLF;
       END;
     TIT=TRUE; DO WHILE (TIT=TRUE);
       CALL CRLF; CALL PRINTSTRING(.TQH,LENGTH(TQH));
       CALL PACKBCD;TIH=N;
        CALL PRINTSTRING(.TQM,LENGTH(TQM));
       CALL PACKBCD; TIM=N;
CALL CRLF; IF((TIH<1 AND TIM<14H) OR TIM>59H ) THEN
     TIT=TRUE;
           ELSE TIT=FALSE;
         IF TIT THEN CALL PRINTSTRING(.SHORT,LENGTH(SHORT));
     END;
       CALL PRINTSTRING(.STH,LENGTH(STH));CALL PACKBCD;
     SHRS=N;CALL PRINTSTRING(.STHM,LENGTH(STHM));CALL PACKBCD;
       SMIN=N;
       CALL CRLF;
   CALL PRINTSTRING(.STOPTIME,LENGTH(STOPTIME));
    CALL PRINTSTRING(.SETYEAR,LENGTH(SETYEAR));
    CALL PACKBCD; YS=N;
    CALL PRINTSTRING(.SETMON,LENGTH(SETMON));
    CALL PACKBCD; MNS=N;
    CALL PRINTSTRING (.SETDAY,LENGTH(SETDAY));
    CALL PACKBCD; DS=N;
    CALL PRINTSTRING(.SETHR,LENGTH(SETHR));
    CALL PACKBCD; HS=N;
    CALL PRINTSTRING(.SETMIN,LENGTH(SETMIN));
    CALL PACKBCD; MS=N;
    CALL CRLF;
   CALL CYCLE;
 END MAIN;


DECLARE LST DATA(CR,LF,'ILLEGAL CHAR PLEASE REENTER',
   CR,LF);
```

```
 DECLARE SELECTSHEAD DATA(CR,LF,'SELECT HEAD 1 OR 2-');
  SEL$HEAD:PROCEDURE; /* SETS UP HEAD BITS FROM INPUT DATA*/
CALL PRINTSTRING(.SELECTSHEAD,LENGTH(SELECTSHEAD));
CALL TTINOUT;CALL CRLF;
N=N-'0';IF N=0 THEN GOTO EXIT;IF N>2 THEN GOTO EXIT;
   IF N=1 THEN DO;
   SELDET=0;BIT$07(0)=FALSE;  END;
  ELSE DO;
      SELDET=1;BIT$07(0)=TRUE;END;
 GOTO FINISH;
 DECLARE NPOSS DATA(CR,LF,'NOT POSSIBLE');
 EXIT:CALL PRINTSTRING(.NPOSS,LENGTH(NPOSS));
GOTO LOOP;
 FINISH:;
 END SEL$HEAD;

RSTART$ACCUM:PROCEDURE; /*RESTARTS ACCUM AFTER READ */
          BIT$04(3)=MASK(3);BIT$04(4)=MASK(4);
          CALL CHN$4$OUT;
  END RSTART$ACCUM;

 /*THIS PROCEDURE PRINTS THE CALIBRATION DATA*/
PRINT$CALDATA:PROCEDURE;
DECLARE CAL$HEAD DATA(' MINUTE COUNT',CR,LF,LF);
CALL PRINTSTRING(.HEAD1+3,20);CALL PRINTBCD(CAL$TIME);
 CALL PRINTSTRING(.CAL$HEAD,LENGTH(CAL$HEAD));
 CALL PRINT$REPORT;
END PRINTCALDATA;

/*THIS PROCEDURE ADJUSTS THE 2 MIN BACKGROUND FOR
  THE CALIBRATION RUN REQUIREMENTS                */
NET$BKG:PROCEDURE;
 CALL CAL;
 DECLARE MLP1 DATA(2,0FEH,5,0,0,24H);
 DECLARE MLP2 DATA(1,5,0FEH,0,0,24H);
 DECLARE MLP3 DATA(1,7,0FEH,5,0,24H);
  DECLARE MLP4 DATA(1,0FEH,0,0,0,24H);
  DO I=0 TO 2;
CALL INP(.MLP1+M*6);CALL MUL(.FB+I*4);CALL STR(.FB+I*4);
  END;
CALL NET$COUNT;
  END NET$BKG;

CAL$SUB:PROCEDURE; /*SUBROUTINE FOR CALIBRATE PROCEDURE*/
CALL CCUNT; CALL READ$ACCUM(.D0,.D1,.D2); CALL RSTART$ACCUM;
  /* CALCULATE AND PRINT*/
  CALL NET$BKG;  CALL PRINT$CALDATA;
END CAL$SUB;
```

```
     CALIBRATE:    PROCEDURE;  /*CONTROLS CALIBRATION CYCLE*/
  DECLARE(N1,N2) BYTE;
  DO N1=1 TO 2;
   IF N1 =1 THEN DO;  SELDET=0; BIT$07(0)=FALSE;  END;
   ELSE DO; SELDET=1; BIT$07(0)=TRUE;END;
    DO N2=0 TO 9;
  OUTPUT(OFBH)=0;ENABLE; /*BE SURE CLOCK IS RUNNING*/
 CAL$TEST=TRUE;
 CALL SAMPLE;  /*TAKE A NORMAL RUN*/
 CALL RSTART$ACCUM;
          /* DO MATH AND PRINT IT OUT*/
     CALL CAL; CALL PRINT$REPORT;
  CAL$TEST1=TRUE;CAL$TEST2=TRUE;
  /* SET UP TO GET 5MIN DATA*/
  CAL$TIME=05H;
TSTOP=556; M=0; CALL CAL$SUB;
 /*SET UP TO GET 30 MIN DATA */
   CAL$TIME=30H;
 TSTOP=2056; M=1; CALL CAL$SUB;
  /*SET UP FOR 35 MIN DATA*/
  CAL$TIME=35H;
  TSTOP=2356; M=2; CALL CALSUB;
/*   SET UP TO DO KUSNETZ METHOD*/
 M=3;
   CAL$TIME=39H;
CAL$TEST1=FALSE;
  CAL$TEST = FALSE; TSTART=2596;TSTOP=2716;
CALL CAL$SUB;
  CAL$TEST2=FALSE;
IF INPUT(TTYBUF)AND 7FH=1BH THEN GOTO FINISH;
  END; /*END OF INNER LOOP-10 CAL RUNS*/
  END; /*END OF OUTER LOOP -BOTH HEADS*/
FINISH:; /*ESCAPE ROUTE*/
 END CALIBRATE;
```

```
   SOURCE$CHECK:   PROCEDURE; /*ALLOWS CALIBRATION WITH SOURCES */
DECLARE SCHK DATA('SOURCE CHECK MODE  ');
DECLARE SEL$SOURCE DATA(CR,LF,'SELECT MODE  A-B-C -');
DECLARE I BYTE;
   START:CALL SEL$HEAD;CALL DET$HEAD$OUT;
   CALL TEST$AC$POWER;
    IF FAILTST=TRUE THEN GOTO LOOP;
   CALL PRINTSTRING(.SEL$SOURCE,LENGTH(SEL$SOURCE));
CALL TYINOUT;CALL CRLF;
     /* CHECK ALLOWABLE VALUES OF INPUT*/
IF N<'A' THEN GOTO EXIT;IF N>'C' THEN GOTO EXIT;
    /*CHECK MODE AND SET UP OUTPUT PORTS*/
   IF N='C' THEN BIT$07(7)=TRUE;
   ELSE BIT$07(7)=FALSE;CALL DET$HEAD$OUT;
    /*TAKE A TWO MINUTE COUNT*/
   OUTPUT(0FBH)=0;ENABLE; /*BE SURE CLOCK IS RUNNING*/
   TIMER$SECS=0;TSTART=1;TSTOP=121;CALL COUNT;/*GET DATA*/
    /* READ COUNT AND STORE*/
     CALL READ$ACCUM(.D0,.D1,.D2);
     /*ZERO UNSELECTED DATA AREAS*/
    DO CASE N-'A';
       DO;   /* CASE 0,N=A,ZERO (B+C) AND C'*/
         DO I=0 TO 6;
             D1(I)=0;D2(I)=0;END;
       END;
        DO;  /* CASE 1,N=B,  ZERO A AND C'*/
         DO I=0 TO 6;
             D0(I)=0;D2(I)=0;END;
          END;
       DO; /*CASE 2,N=C, ZERO A AND (B+C)*/
         DO I=0 TO 6;
           D0(I)=0;D1(I)=0;END;
         END;
      END; /*END CASE*/
  /* ZERO BACKGROUND DATA AREA*/
  DO I=0 TO 23;B0(I)=0;END;
B0(7)=24H;B1(7)=24H;B2(7)=24H; /*PLACE $ FOR MATH PACK DELIMITER*/
    /* PERFORM MATH AND PRINT IT OUT*/
CALL PRINTSTRING(.HEAD1+3,20);
CALL PRINTSTRING(.SCHK,LENGTH(SCHK));
CALL PRINTSTRING(.N,1);
   CALL CAL;CALL PRINT$REPORT;
    GO TO START;
EXIT:;
DECLARE PRNTEND DATA(CR,LF,'END ');
CALL PRINTSTRING(.PRNTEND,LENGTH(PRNTEND));
CALL PRINTSTRING(.SCHK,LENGTH(SCHK));CALL CRLF;
    END SOURCE$CHECK;
```

```
ALT$FACTOR:PROCEDURE;
    DO M=1 TO 2     ;
    CALL SEL$HEAD;
CALL PRINTSTRING(.ALTMSG,LENGTH(ALTMSG));
CALL IMP$LINE(.ALT$COR$FACTOR);
DO I=0 TO CNT;ALT$COR$FACTOR(I)=ALT$COR$FACTOR(I)-30H;END;
ALT$COR$FACTOR(13)=24H;
      IF SELDET=0 THEN
    DO I=0 TO 13;ALTFACTOR1(I)=ALT$COR$FACTOR(I);    END;
    ELSE
    DO I=0 TO 13;ALTFACTOR2(I)=ALT$COR$FACTOR(I);    END;
    END;
END ALT$FACTOR;

        /* MAIN */
/*  THE MAIN PROGRAM LOOP  CALLS THE REQUIRED PROCEDURES*/
DECLARE SIGNON DATA('REMOTE WORKING LEVEL MONITOR VERSION',
  '   11  04-26-77',CR,LF);
RESET:
   OUTPUT(RWLM$HEAD)=FALSE; /*RESET HEAD*/
DO I=0 TO 5 ;YRS(I)=0;END; /*RESET DATE AND TIME TO 0*/
   CAL$TEST=FALSE; /*BE SURE NOT IN CAL MODE*/
    CAL$TEST1=FALSE;CAL$TEST2=FALSE;
  /*RESET ALTITUDE CORECTION FACTOR TO 1.0*/
  DO I=0 TO 12;ALTFACTOR1(I)=0;ALTFACTOR2(I)=0;END;
  ALTFACTOR1(0)=1;ALTFACTOR1(1)=24H;
  ALTFACTOR2(0)=1;ALTFACTOR2(1)=24H;
CNT=1; SELDET=0;
START: CALL CRLF;
 CALL PRINTSTRING(.SIGNON, LENGTH(SIGNON));
 CALL TYPEDATE;
DO I=0 TO 7;BIT$07(I)=0;BIT$04(I)=0; END; /*CLEAR CONTROL VECTOR*/
LOOP: CALL PRINTPROMPT;   /* INPUT CHAR FROM TTY,ECHO,BRANCH*/
       OUTPUT(0FBH)=0;  ENABLE;
       CALL TYINOUT;
        /*PROCESS TYPED COMMAND INTO NUMBER AND EXECUTE*/
     IF (N:=(N-40H)) > 19
      THEN GOTO MESSAGE;
```

```
      /* THIS SECTION SELECTS THE BRANCH SELECTED BY THE KBRD*/
     DO CASE N;
       GOTO MESSAGE;  /* CASE 0 */
       CALL ALT$FACTOR; /*CHAR A*/
       GOTO MESSAGE;  /* CHAR=B */
       CALL CALIBRATE; /*CHAR C*/
       CALL CHANGEDATE;   /*CHAR = D*/
       CALL CAL;   /* CHAR = E*/
       CALL TYPEDATE;     /* CHAR = F*/
       CALL MAIN;    /*CHAR = G */
        CALL SEL$HEAD;      /*CHAR H*/
       GOTO MESSAGE;    /* CHAR = I*/
       CALL PAPADV;     /*CHAR = J*/
       GOTO MESSAGE;  /* CHAR=K */
       GOTO MESSAGE;  /* CHAR=L */
       GO TO 3800H;/*CHAR =M,RETURN TO MONITOR*/
       GOTO MESSAGE;  /* CHAR=N */
       GOTO MESSAGE;  /* CHAR=O */
      CALL PRINT$REPORT;  /* CHAR =P*/
DO;
 CALL READ$ACCUM(.B0,.B1,.B2);CALL READACCUM(.D0,.D1,.D2);
END;    /*CHAR Q*/
       GOTO RESET;     /* CHAR =R */
       CALL SOURCE$CHECK; /*CHAR=S*/
     END;
    GOTO LOOP;
MESSAGE:    CALL PRINTSTRING(.LST,LENGTH(LST));
GOTO LOOP;
     EOF;
```

PIN TO ASSEMBLY LANGUAGE CROSS INDEX

00000005 SYSIN	1=0103H	19=0106H	20=011CH	26=029CH	51=02A3H	52=02AFH
00000006 SYSIN	53=02B9H	54=02CBH	55=02E5H	56=02FAH	61=031DH	62=0336H
00000007 SYSIN	63=033CH	76=0340H	77=0342H	78=0349H	79=034FH	80=0352H
00000008 SYSIN	81=0372H	82=0373H	84=0375H	85=037CH	86=037FH	89=0382H
00000009 SYSIN	90=0384H	91=038CH	92=038DH	94=0390H	95=0395H	99=039BH
00000010 SYSIN	101=039FH	102=03ABH	103=03C1H	104=03CFH	105=03D5H	106=03D6H
00000011 SYSIN	110=03DBH	111=03E0H	112=03E1H	115=03F4H	116=03F9H	117=03FAH
00000012 SYSIN	120=03EDH	121=03F3H	122=03F5H	123=03F8H	124=0406H	125=0409H
00000013 SYSIN	130=0412H	131=0415H	132=0420H	133=042CH	134=0433H	135=0434H
00000014 SYSIN	138=0438H	139=0441H	140=0447H	141=0453H	142=0454H	146=0457H
00000015 SYSIN	147=0471H	148=0479H	149=047FH	154=048CH	155=048DH	156=048EH
00000016 SYSIN	157=048FH	160=0490H	161=049DH	162=049EH	163=049FH	166=04A2H
00000017 SYSIN	167=04ADH	168=04BBH	169=04C3H	170=04CAH	171=04CFH	172=04D6H
00000018 SYSIN	173=04DBH	174=04E2H	175=04EAH	176=04F1H	177=04F6H	178=04FDH
00000019 SYSIN	179=0500H	180=0501H	183=050FH	184=051BH	185=0527H	186=0533H
00000020 SYSIN	187=0541H	191=0542H	192=054AH	193=054DH	194=0551H	195=0559H
00000021 SYSIN	196=055CH	197=056CH	198=056BH	199=056EH	200=056FH	201=0577H
00000022 SYSIN	202=057AH	203=057EH	204=0586H	205=0589H	206=058DH	207=0590H
00000023 SYSIN	208=0595H	209=0598H	210=0599H	211=059CH	212=059DH	217=05A1H
00000024 SYSIN	218=05A7H	228=05ABH	229=05C1H	231=05C6H	233=05C8H	234=05CDH
00000025 SYSIN	235=05D5H	236=05D8H	237=05DAH	238=05DCH	239=05E3H	240=05E6H
00000026 SYSIN	241=05F7H	242=05FBH	243=0603H	244=0606H	245=0608H	246=060BH
00000027 SYSIN	247=0613H	248=0616H	250=061BH	251=061EH	252=0621H	253=062EH
00000028 SYSIN	254=0634H	255=0636H	256=063FH	257=0647H	258=0648H	259=064BH
00000029 SYSIN	264=064EH	265=0654H	267=0657H	268=0662H	269=066AH	270=0674H
00000030 SYSIN	271=0675H	285=067AH	286=0687H	287=069FH	288=06A5H	289=06AAH
00000031 SYSIN	290=06ABH	305=06B0H	306=06BEH	307=06D7H	308=06DFH	309=06E5H
00000032 SYSIN	310=06E6H	312=06E9H	313=06FFH	314=0704H	315=070BH	316=070FH
00000033 SYSIN	317=071DH	318=0720H	319=0723H	320=072BH	321=0733H	322=0734H
00000034 SYSIN	329=0737H	330=0751H	331=0768H	332=076DH	333=077CH	334=0781H
00000035 SYSIN	335=0787H	336=07A1H	337=07B6H	338=07BFH	339=07C3H	340=07C6H
00000036 SYSIN	341=07D4H	342=07D9H	343=07DCH	344=0805H	345=0809H	346=0817H
00000037 SYSIN	347=081DH	348=0825H	349=082AH	350=082DH	352=0836H	353=083BH
00000038 SYSIN	354=083EH	355=0846H	356=084FH	357=0853H	358=085CH	359=0860H
00000039 SYSIN	360=086BH	361=0869H	374=086CH	375=086DH	377=0873H	378=0876H
00000040 SYSIN	379=0877H	381=087DH	382=0880H	383=0881H	385=0887H	386=088AH
00000041 SYSIN	387=088BH	389=0891H	390=0894H	391=0895H	393=089BH	394=089EH
00000042 SYSIN	395=089FH	397=08A5H	398=08ACH	399=08ADH	401=08AFH	402=08B2H
00000043 SYSIN	403=08B3H	405=08B9H	406=08BCH	407=08BDH	409=08C7H	410=08CFH
00000044 SYSIN	411=08D8H	412=08D9H	420=08DFH	421=08E7H	422=0953H	423=0970H
00000045 SYSIN	424=0994H	425=099BH	426=099CH	429=099FH	431=09A4H	432=09BDH
00000046 SYSIN	433=0A08H	434=0A0FH	435=0A15H	436=0A18H	437=0A41H	438=0A44H
00000047 SYSIN	439=0A6DH	440=0A76H	442=0A7EH	443=0A84H	444=0ACFH	445=0AD6H
00000048 SYSIN	447=0AD9H	453=0ADEH	454=0AE7H	455=0B0CH	456=0B0FH	457=0B3DH
00000049 SYSIN	458=0B6BH	459=0B6EH	460=0B92H	461=0BADH	462=0BB0H	463=0BD8H
00000050 SYSIN	464=0BDFH	465=0BDEH	466=0BE1H	467=0BF4H	468=0C05H	469=0C0CH
00000051 SYSIN	470=0C3DH	473=0C6DH	474=0C81H	476=0CD1H	477=0CFCH	479=0D45H
00000052 SYSIN	481=0D69H	486=0D6EH	487=0D95H	488=0DBFH	489=0DC9H	490=0DE1H
00000053 SYSIN	491=0DF9H	492=0DFCH	493=0DFFH	494=0E02H	495=0E11H	496=0E3AH
00000054 SYSIN	497=0E40H	498=0E4EH	499=0E6BH	500=0E8EH	501=0EA8H	502=0EABH
00000055 SYSIN	503=0EB2H	504=0EBBH	505=0EC3H	506=0EC7H	507=0EDCH	508=0EEAH
00000056 SYSIN	509=0F16H	510=0F25H	511=0F4BH	512=0F5AH	513=0F80H	514=0F83H
00000057 SYSIN	515=0F8AH	516=0F90H	517=0F95H	518=0FBCH	519=0FE5H	520=100CH
00000058 SYSIN	521=100FH	527=1015H	528=101BH	529=102DH	530=104FH	531=1060H
00000059 SYSIN	532=1063H	533=106BH	534=106CH	536=106DH	537=1079H	538=108CH
00000060 SYSIN	539=108DH	541=1097H	547=109FH	548=10A2H	549=10A9H	550=10B1H
00000061 SYSIN	551=10BCH	552=10D1H	553=10E6H	554=10E8H	555=10F7H	556=10FFH
00000062 SYSIN	557=1104H	558=110DH	559=1115H	560=1119H	561=111FH	562=112AH

```
00000063 SYSIN
00000064 SYSIN
00000065 SYSIN
00000066 SYSIN
00000067 SYSIN
00000068 SYSIN
00000069 SYSIN
00000070 SYSIN
00000071 SYSIN
00000072 SYSIN
00000073 SYSIN
00000074 SYSIN
00000075 SYSIN
00000076 SYSIN
00000077 SYSIN
00000078 SYSIN
00000079 SYSIN
00000080 SYSIN
00000081 SYSIN
00000082 SYSIN
00000083 SYSIN
00000084 SYSIN
00000085 SYSIN
00000086 SYSIN
00000087 SYSIN
00000088 SYSIN
00000089 SYSIN
00000090 SYSIN
00000091 SYSIN
00000092 SYSIN
00000093 SYSIN
00000094 SYSIN
00000095 SYSIN
00000096 SYSIN
00000097 SYSIN
00000098 SYSIN
00000099 SYSIN
00000100 SYSIN
00000101 SYSIN
00000102 SYSIN
00000103 SYSIN
00000104 SYSIN
00000105 SYSIN
00000106 SYSIN
00000107 SYSIN
00000108 SYSIN
00000109 SYSIN
00000110 SYSIN
00000111 SYSIN
00000112 SYSIN
00000113 SYSIN
00000114 SYSIN
```

PLM TO ASSEMBLY LANGUAGE CROSS INDEX

```
 563=112FH  564=1140H  565=1151H  566=1152H  567=1167H  568=116AH
 569=1175H  570=1178H  571=117FH  572=1187H  573=11A5H  574=11A7H
 576=11BFH  579=11C1H  580=11CDH  581=11D0H  582=11D3H  583=11D6H
 584=11DEH  585=11E1H  586=11F2H  590=11F5H  591=11F2H  592=11F7H
 593=1205H  594=1208H  595=120BH  596=1213H  597=1218H  598=1219H
 607=121CH  608=1227H  609=1224H  610=1238H  611=123BH  614=1242H
 615=1252H  616=1255H  617=1264H  618=126CH  619=1275H  620=1276H
 621=1285H  622=1288H  623=129EH  624=12BFH  625=129EH  626=12A1H
 627=12ADH  628=12B0H  629=12B3H  631=12B6H  633=12BDH  634=12C6H
 635=12C9H  636=12D8H  637=12DEH  638=12E0H  639=12E1H  640=12E7H
 642=12E8H  650=12EDH  651=1306H  653=130AH  654=1313H  655=1317H
 657=131AH  658=1321H  659=1325H  660=132EH  661=1338H  662=133CH
 663=1340H  664=1345H  665=134FH  666=1356H  667=1369H  668=13A2H
 669=13A9H  670=13AAH  673=13ADH  674=13B5H  676=13B8H  677=13C4H
 678=13CEH  679=13DCH  680=13E7H  681=13F2H  682=1404H  683=1411H
 684=1415H  685=141CH  686=1429H  687=1434H  688=1437H  689=1443H
 690=1446H  691=1466H  692=146AH  693=1471H  694=1482H  695=1485H
 696=1490H  697=149FH  698=14A3H  699=14A6H  700=14AFH  701=14B6H
 702=14B9H  703=14C6H  704=14C9H  705=14D6H  706=14D9H  707=14E6H
 708=14E9H  709=14F6H  710=14F9H  711=1501H  712=1504H  713=1505H
 714=1524H  717=1539H  719=1541H  720=1544H  721=1547H  722=155CH
 723=1559H  724=155EH  725=1566H  726=1569H  727=1571H  728=1574H
 729=1582H  730=158AH  731=1590H  732=1591H  736=1599H  737=15B3H
 738=15B8H  739=15B9H  741=15BCH  742=15CCH  743=15E1H  744=15E9H
 745=15ECH  746=15EDH  747=15F0H  748=15F3H  749=15F9H  750=15FFH
 751=1605H  752=160BH  753=1619H  754=1679H  755=1680H  756=1683H
 757=1684H  759=1687H  760=169CH  761=169FH  762=16A2H  763=16A3H
 768=16A8H  769=16B1H  770=16BFH  771=16CFH  772=16D9H  774=16DCH
 775=16DDH  776=16E1H  777=16E4H  778=16E7H  779=16EAH  780=16EDH
 781=16F1H  782=16F4H  783=16F7H  784=1706H  786=170AH  787=1719H
 790=171DH  791=172CH  793=1731H  794=1736H  795=173AH  796=1745H
 797=174EH  799=1752H  800=1764H  801=1767H  802=176CH  803=176DH
 807=1770H  808=1783H  809=1799H  811=179CH  812=179FH  813=17A2H
 814=17B1H  815=17B9H  816=17BCH  817=17BFH  818=17C5H  819=17D1H
 820=17CEH  821=17DBH  822=17EAH  823=17FDH  824=17EEH  825=1805H
 826=1808H  827=1817H  828=181DH  829=1823H  830=1828H  831=1831H
 832=184DH  833=1850H  834=1855H  835=185FH  836=187AH  837=187DH
 838=1882H  839=188BH  840=18A7H  841=18AAH  842=18BAH  843=18BFH
 844=18D8H  645=18F2H  847=18F3H  848=1902H  849=190AH  850=1912H
 851=1918H  852=191BH  853=191EH  854=1924H  855=192CH  856=1937H
 857=1938H  859=193DH  860=1946H  861=1949H  862=1951H  863=1957H
 864=1983H  865=198AH  866=1991H  867=19BCH  868=19BFH  869=19E7H
 870=19EEH  871=19EFH  876=1A24H  879=1A27H  880=1A44H  881=1A49H
 892=1A4CH  884=1A4FH  885=1A79H  886=1A80H  887=1A88H  888=1A8FH
 889=1A93H  890=1A96H  891=1A9FH  892=1AA1H  903=1AC9H  894=1ACCH
 895=1ACFH  896=1AD3H  897=1ADAH  898=1AE4H  901=1AE1H  903=1AE9H
 904=1AECH  905=1AEFH  906=1AF5H  907=1AF8H  909=1AFEH  909=1B04H
 910=1B0AH  911=1B10H  912=1B16H  913=1B1CH  914=1B1FH  915=1B25H
 916=1B28H  917=1B2BH  918=1B31H  919=1B34H  920=1B37H  921=1B3AH
 922=1B5EH  923=1B61H  924=1B64H  925=1B67H  926=1B9CH  927=1B9FH
 928=1BA7H  929=1BAAH
STACK SIZE =  36 BYTES
```

VARIABLE LOCATION IN RAM

```
MEMORY..........................2200H
MASK............................0106H
CONSTB..........................0110H
CNT.............................2049H
BIT07...........................204AH
BIT08...........................2052H
NET1............................205AH
NET2............................2067H
NET3............................2074H
R0..............................2081H
R1..............................2089H
R2..............................2091H
D0..............................2099H
D1..............................20A1H
D2..............................20A9H
FAILTST.........................20B1H
SHIPET..........................20B2H
DETSPL..........................20B3H
TIMERMINS.......................20B5H
TIMERHRS........................20B6H
TIT.............................20B7H
TIMERSECS.......................20B9H
TIM.............................20BAH
TIH.............................20BBH
SMIN............................20BCH
SHRS............................20BDH
HEAD............................20BFH
SAMPLEHOURS.....................20C0H
SAMPLEMINS......................20C1H
RETHRS..........................20C2H
RETMINS.........................20C5H
YRS.............................20C8H
MONS............................20C9H
DAYS............................20CAH
HRS.............................20CBH
MINS............................20CCH
SECS............................20CDH
I...............................20CEH
N...............................20CFH
M...............................20D0H
TT..............................20D1H
TSTPER..........................20D2H
TSTSOL..........................20D3H
TSTART..........................20D4H
TSTOP...........................20D6H
DATEOK..........................0290H
MISM............................02A3H
MISM2...........................02AFH
STOPTIME........................02B9H
TOP.............................02C3H
TON.............................02F5H
SHORT...........................02FAH
STM.............................031DH
STMM............................0336H
ALTCORFACTOR....................20D9H
FL..............................20E6H
YS..............................211AH
MNS.............................211BH
```

VARIABLE LOCATION IN RAM

```
DS..............................211CH
HS..............................211DH
MS..............................211EH
CALTEST.........................211FH
CALTEST1........................2120H
CALTEST2........................2121H
CALTIME.........................2122H
AITFACTOR1......................2123H
AITFACTOR2......................2130H
TYPEOUT.........................033CH
CHAR............................213DH
I...............................213EH
TYPEIN..........................0373H
TYINOUT.........................037FH
ECHO............................038DH
CNT.............................213FH
INPLINE.........................0395H
F...............................2140H
T...............................2143H
CRLF............................03D6H
PRINTPROMPT.....................03E1H
PACKBCD.........................03EAH
PRINTSTRING.....................0409H
NAME............................2144H
LENGTH..........................2147H
I...............................2148H
PRINTBCD........................043AH
P...............................2149H
ERROR...........................0454H
ERRMSG..........................0457H
START...........................1A93H
DELAY1MS........................047FH
DELAY5MS........................048FH
TYPEDATE........................049FH
DATE............................04A2H
TIME............................04ADH
SETYEAR.........................0501H
SETMON..........................050EH
SETDAY..........................051BH
SETHR...........................0527H
SETMIN..........................0533H
CHANGEDATE......................0541H
RESTART.........................059DH
TIMEKEEPER......................05A7H
DAYMAX..........................05AEH
DETMSG..........................0654H
CONMSG..........................0657H
DETHPADOUT......................0675H
I...............................214AH
CHNAOUT.........................06ABH
I...............................214BH
RELEASESOLENOID.................06B6H
SOLMSG..........................06B9H
PAPADV..........................0730H
PULSES..........................214CH
FINISH..........................084BH
NOPAPER.........................0737H
EXTRAADVANCE....................0751H
```

VARIABLE LOCATION IN RAM

```
00000233 SYSIN     INIT...........................0869H
00000234 SYSIN     ADD............................086DH
00000235 SYSIN     LOC............................214FH
00000236 SYSIN     SUB............................0877H
00000237 SYSIN     LOC............................2150H
00000238 SYSIN     MUL............................0881H
00000239 SYSIN     LOC............................2152H
00000240 SYSIN     DIV............................088BH
00000241 SYSIN     LOC............................2154H
00000242 SYSIN     MOD............................0895H
00000243 SYSIN     LOC............................2156H
00000244 SYSIN     STR............................089FH
00000245 SYSIN     LOC............................2158H
00000246 SYSIN     INP............................08A9H
00000247 SYSIN     LOC............................215AH
00000248 SYSIN     OUTPT..........................08B3H
00000249 SYSIN     LOC............................215CH
00000250 SYSIN     INSTR..........................08BDH
00000251 SYSIN     A1.............................215EH
00000252 SYSIN     A2.............................2160H
00000253 SYSIN     TEMP...........................2162H
00000254 SYSIN     ASUB...........................2192H
00000255 SYSIN     FSUB...........................2194H
00000256 SYSIN     TSUB...........................2196H
00000257 SYSIN     ALTCONF........................2199H
00000258 SYSIN     FT.............................219DH
00000259 SYSIN     FB.............................21A0H
00000260 SYSIN     FCON...........................21C1H
00000261 SYSIN     NETCOUNT.......................09D9H
00000262 SYSIN     CAL............................099CH
00000263 SYSIN     I..............................21B1H
00000264 SYSIN     I1.............................21B2H
00000265 SYSIN     FLM............................0C0DH
00000266 SYSIN     L..............................0C6DH
00000267 SYSIN     READ1..........................0CA1H
00000268 SYSIN     CL.............................0CD1H
00000269 SYSIN     CH.............................0CFCH
00000270 SYSIN     ALTMSG.........................0D45H
00000271 SYSIN     PRINTREPORT....................0D69H
00000272 SYSIN     I..............................21B3H
00000273 SYSIN     CALPRINT.......................0FBEH
00000274 SYSIN     COUNT..........................100EH
00000275 SYSIN     CALLOOP........................1063H
00000276 SYSIN     1S.............................21B4H
00000277 SYSIN     STARTTIM.......................106CH
00000278 SYSIN     READACCTM......................108DH
00000279 SYSIN     A..............................21B6H
00000280 SYSIN     B..............................21B8H
00000281 SYSIN     C..............................21BAH
00000282 SYSIN     T..............................21BCH
00000283 SYSIN     FLOWSG.........................11A7H
00000284 SYSIN     FLOWCHECK......................11BFH
00000285 SYSIN     TESTACPOWER....................11B2H
00000286 SYSIN     PWRMSG.........................11E5H
00000287 SYSIN     SAMPLE.........................1219H
00000288 SYSIN     FINISH.........................12E7H
00000289 SYSIN     CYCLE..........................12E0H
00000290 SYSIN     EATW...........................13AAH
```

```
                                        VARIABLE LOCATION IN RAM

00000291 SYSIN                           LST..............................1505H
00000292 SYSIN                           SELECTHEAD.......................1524H
00000293 SYSIN                           SELHEAD..........................1539H
00000294 SYSIN                           EXIT.............................1582H
00000295 SYSIN                           FINISH...........................1590H
00000296 SYSIN                           NPOSS............................1574H
00000297 SYSIN                           LOOP.............................1AC9H
00000298 SYSIN                           RSTARTACCUM......................1591H
00000299 SYSIN                           PRINTCALDATA.....................15B9H
00000300 SYSIN                           CALHEAD..........................15BCH
00000301 SYSIN                           NETBKG...........................15EDH
00000302 SYSIN                           MLP1.............................15F3H
00000303 SYSIN                           MLP2.............................15F9H
00000304 SYSIN                           MLP3.............................15FFH
00000305 SYSIN                           MLP4.............................1605H
00000306 SYSIN                           CALSUB...........................1684H
00000307 SYSIN                           CALIBRATE........................16A3H
00000308 SYSIN                           F1...............................21FDH
00000309 SYSIN                           N2...............................21FEH
00000310 SYSIN                           FINISH...........................176CH
00000311 SYSIN                           SOURCECHECK......................176DH
00000312 SYSIN                           SCHK.............................1770H
00000313 SYSIN                           SELSOURCE........................1783H
00000314 SYSIN                           I................................21FFH
00000315 SYSIN                           START............................1799H
00000316 SYSIN                           EXIT.............................191BH
00000317 SYSIN                           FRNTEND..........................191EH
00000318 SYSIN                           ALTFACTOR........................1938H
00000319 SYSIN                           SIGNON...........................19FFH
00000320 SYSIN                           RESET............................1A24H
00000321 SYSIN                           MESSAGE..........................1B9BH
```

PLM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
00000324 SYSIN   0000H JMP    9DH    05H
00000325 SYSIN   0038H JMP    A7H    05H
00000326 SYSIN   0100H LXI SP FFH    23H    JMP    24H    1AH
00000327 SYSIN   0106H 01H 02H 04H 08H 10H 20H 40H 80H FFH 00H
00000329 SYSIN   0110H 00H 00H 00H 00H 00H 00H 00H 00H 00H 00H 00H 00H 00H 00H 01H 20H
00000330 SYSIN   0299H 44H 41H 54H 45H 20H 43H 4FH 52H 52H 45H 43H 54H 20H 28H 59H 2FH
00000331 SYSIN   02A0H 4EH 29H 3FH
00000332 SYSIN   02A3H 44H 4FH 20H 59H 4FH 55H 20H 57H 41H 4EH 54H 20H
00000333 SYSIN   02AFH 20H 52H 55H 4EH 28H 59H 2FH 4EH 29H 3FH
00000334 SYSIN   02B9H 49H 4EH 50H 55H 54H 20H 54H 49H 4DH 45H 20H 54H 4FH 20H 53H 54H
00000335 SYSIN   02C9H 4FH 50H
00000336 SYSIN   02CBH 45H 4EH 54H 45H 52H 20H 54H 49H 4DH 45H 20H 49H 4EH 54H 45H 52H
00000337 SYSIN   02DBH 56H 41H 4CH 20H 42H 45H 54H 57H 45H 45H 4EH 20H 53H 41H 4DH 50H
00000338 SYSIN   02EBH 4CH 45H 53H 20H 48H 4FH 55H 52H 53H 20H
00000339 SYSIN   02F5H 20H 4DH 49H 4EH 2DH
00000340 SYSIN   02FAH 0DH 0AH 54H 49H 4DH 45H 20H 54H 4FH 4FH 20H 53H 48H 4FH 52H 54H
00000341 SYSIN   030AH 20H 4DH 55H 53H 54H 20H 42H 45H 20H 3EH 31H 36H 20H 4DH 49H 4EH
00000342 SYSIN   031AH 53H 0DH 0AH
00000343 SYSIN   031DH 45H 4EH 54H 45H 52H 20H 54H 49H 4DH 45H 20H 54H 4FH 20H 53H 54H
00000344 SYSIN   032DH 41H 52H 54H 20H 48H 4FH 55H 52H 2DH
00000345 SYSIN   0336H 20H 20H 4DH 49H 4EH 2DH
00000346 SYSIN   033CH LXI H  3DH    21H    MOV MC IN     00H    ANA I  80H    SUB I
00000347 SYSIN   0345H 00H    JZ     40H    03H    LXI H  3DH    21H    MOV AM OUT
00000348 SYSIN   034EH 01H    MOV AM SUB I  0DH    JNZ    72H    03H    INR L  MOV MI
00000349 SYSIN   0357H 00H    MOV AI 08H    LXI H  3EH    21H    SUB M  JC     72H
00000350 SYSIN   0360H 03H    MOV AI FAH    MOV BI 0CH    MOV CB DCR C  JNZ    66H
00000351 SYSIN   0369H 03H    DCR A  JNZ    65H    03H    INR M  JNZ    5BH    03H
00000352 SYSIN   0372H RET    IN     00H    ANA I  01H    SUB I  00H    JZ     73H
00000353 SYSIN   037BH 03H    IN     01H    RET    CALL   73H    03H    ANA I  7FH
00000354 SYSIN   0384H LXI H  CFH    20H    MOV MA MOV CM CALL   3CH    03H    RET
00000355 SYSIN   038DH CALL   7FH    03H    LXI H  CFH    20H    MOV AM RET    LXI H
00000356 SYSIN   0396H 40H    21H    MOV MC INX H  MOV MB MOV LI 3FH    MOV MI 00H
00000357 SYSIN   039FH CALL   8DH    03H    INR H  MOV LI 43H    MOV MA SUB I  0DH
00000358 SYSIN   03A8H JZ     C1H    03H    MOV LI 3FH    MOV CM MOV BI 00H    LHLD
00000359 SYSIN   03B1H 40H    21H    DAD B  XCHG   LXI H  43H    21H    MOV CM MOV AC
00000360 SYSIN   03BAH STAX D MOV LI 3FH    INR M  JMP    9FH    03H    MOV LI 3FH
00000361 SYSIN   03C3H MOV CM MOV BI 00H    LHLD   40H    21H    DAD B  XCHG   MOV AI
00000362 SYSIN   03CCH 24H    SUB I  30H    STAX D MOV CI CAH    CALL   3CH    03H
00000363 SYSIN   03D5H RET    MOV CI 0DH    CALL   3CH    03H    MOV CI 0AH    CALL
00000364 SYSIN   03DEH 3CH    03H    RET    CALL   D6H    03H    MOV CI 2AH    CALL
00000365 SYSIN   03E7H 3CH    03H    RET    CALL   7FH    03H    LXI H  CFH    20H
00000366 SYSIN   03F0H MOV AM SUB I  30H    INR L  MOV MA CALL   7FH    03H    LXI H
00000367 SYSIN   03F9H D0H    20H    MOV AM ADD A  ADD A  ADD A  ADD A  DCR L  MOV CA
00000368 SYSIN   0402H MOV AM SUB I  30H    ADD C  INR L  MOV MA RET    LXI H  40H
00000369 SYSIN   040BH 21H    MOV MC INX H  MOV MB MOV LI 47H    MOV ME INR L  MOV MI
00000370 SYSIN   0414H 00H    LXI H  47H    21H    MOV CM DCR C  MOV AC INR L  SUB M
00000371 SYSIN   041DH JC     33H    04H    MOV CM MOV BI 00H    LHLD   44H    21H
00000372 SYSIN   0426H DAD B  MOV AM MOV CA CALL   3CH    03H    LXI H  49H    21H
00000373 SYSIN   042FH INR M  JNZ    15H    04H    RET    LXI H  49H    21H    MOV MC
00000374 SYSIN   0438H MOV CI 04H    MOV AM ORA A  RAR    DCR C  JNZ    3BH    04H
00000375 SYSIN   0441H ADD I  30H    MOV CA CALL   3CH    03H    LXI H  49H    21H
00000376 SYSIN   044AH MOV AM ANA I  0FH    ADD I  30H    MOV CA CALL   3CH    03H
00000377 SYSIN   0453H RET    JMP    71H    04H
00000378 SYSIN   0457H 0DH 0AH 07H 07H 2AH 2AH 20H 20H 45H 20H 52H 20H 52H 20H 4FH 20H
00000379 SYSIN   0467H 52H 20H 20H 2AH 2AH 07H 07H 0DH 0AH 00H
00000380 SYSIN   0471H LXI B  57H    04H    MOV EI 1AH    CALL   09H    C0H    LXI SP
00000381 SYSIN   047AH FFH    23H    JMP    93H    1AH    DI     MOV AI CAH    MOV BI
00000382 SYSIN   0483H 0CH    MOV CB DCR C  JNZ    85H    04H    DCR A  JNZ    84H
```

PLM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
00000383 SYSIN          048CH 04H    EI     RET    DI     MOV AI 30H    MOV BI 0CH    MOV CB
00000384 SYSIN          0495H DCR C  JNZ    95H    04H    DCR A  JNZ    94H    04H    EI
00000385 SYSIN          049EH RET    JMP    BBH    04H
00000386 SYSIN          04A2H 0DH 0AH 54H 4FH 44H 41H 59H 20H 49H 53H 20H
00000387 SYSIN          04ADH 20H 20H 54H 48H 45H 20H 54H 49H 4DH 45H 20H 49H 53H 20H
00000388 SYSIN          04BBH LXI B  A2H    04H    MOV EI 0BH    CALL   09H    04H    DCR H
00000389 SYSIN          04C4H MOV LI C9H    MOV CM CALL   34H    04H    MOV CI 2FH    CALL
00000390 SYSIN          04CDH 3CH    03H    LXI H  CAH    20H    MOV CM CALL   34H    04H
00000391 SYSIN          04D6H MOV CI 2FH    CALL   3CH    03H    LXI H  C8H    20H    MOV CM
00000392 SYSIN          04DFH CALL   34H    04H    LXI B  ADH    04H    MOV EI 0FH    CALL
00000393 SYSIN          04E8H 09H    04H    DCR H  MOV LI CBH    MOV CM CALL   34H    04H
00000394 SYSIN          04F1H MOV CI 3AH    CALL   3CH    03H    LXI H  CCH    20H    MOV CM
00000395 SYSIN          04FAH CALL   34H    04H    CALL   D6H    03H    RET
00000396 SYSIN          0501H 0DH 0AH 45H 4EH 54H 45H 52H 20H 59H 45H 41H 52H 2DH
00000397 SYSIN          050EH 20H 45H 4EH 54H 45H 52H 2CH 4DH 4FH 4EH 54H 48H 2DH
00000398 SYSIN          051BH 20H 45H 4EH 54H 45H 52H 2CH 44H 41H 59H 20H 2DH
00000399 SYSIN          0527H 20H 45H 4EH 54H 45H 52H 2CH 48H 4FH 55H 52H 2DH
00000400 SYSIN          0533H 20H 45H 4EH 54H 45H 52H 2CH 4DH 49H 4EH 55H 54H 45H 2DH
00000401 SYSIN          0541H DI     LXI B  01H    05H    MOV EI 0DH    CALL   09H    04H
00000402 SYSIN          054AH CALL   EAH    03H    MOV CM MOV LI C8H    MOV MC LXI B  0FH
00000403 SYSIN          0553H 05H    MOV EI 0DH    CALL   09H    04H    CALL   EAH    03H
00000404 SYSIN          055CH MOV CM MOV LI C9H    MOV MC LXI B  1BH    05H    MOV EI 0CH
00000405 SYSIN          0565H CALL   09H    04H    CALL   EAH    03H    MOV CM MOV LI CAH
00000406 SYSIN          056EH MOV MC LXI B  27H    05H    MOV EI 0CH    CALL   09H    04H
00000407 SYSIN          0577H CALL   EAH    03H    MOV CM MOV LI CBH    MOV MC LXI B  33H
00000408 SYSIN          0580H 05H    MOV EI 0EH    CALL   09H    04H    CALL   EAH    03H
00000409 SYSIN          0589H MOV CM MOV LI CCH    MOV MC CALL   D6H    03H    LXI H  CDH
00000410 SYSIN          0592H 20H    MOV MI 00H    XRA A  OUT    FBH    EI     CALL   9FH
00000411 SYSIN          059BH 04H    RET    PUSH H PUSH D PUSH B PUSH A LXI SP FFH    23H
00000412 SYSIN          05A4H JMP    93H    1AH    PUSH H PUSH D PUSH B PUSH A JMP    C1H
00000413 SYSIN          05ADH 05H
00000414 SYSIN          05AEH 00H 31H 28H 31H 30H 31H 30H 31H 31H 30H 00H 00H 00H 00H 00H 00H
00000415 SYSIN          05BEH 31H 30H 31H
00000416 SYSIN          05C1H LHLD   B8H    20H    INX H  SHLD   B8H    20H    LXI H  CDH
00000417 SYSIN          05CAH 20H    MOV AM ORA M  MOV MA MOV CM INR C  MOV AC DAA    MOV MA
00000418 SYSIN          05D3H SUB I  60H    JNZ    4BH    06H    MOV MI 00H    MOV AM ORA M
00000419 SYSIN          05DCH MOV MA MOV LI B5H    MOV CM INR C  MOV AC DAA    MOV MA SUB I
00000420 SYSIN          05E5H 60H    JNZ    F7H    05H    MOV MI 00H    MOV LI CDH    MOV AM
00000421 SYSIN          05EEH ORA M  MOV MA MOV LI B6H    MOV CM INR C  MOV AC DAA    MOV MA
00000422 SYSIN          05F7H MOV LI CCH    MOV AM ORA M  MOV MA MOV CM INR C  MOV AC DAA
00000423 SYSIN          0600H MOV MA SUB I  60H    JNZ    4BH    06H    MOV MI 00H    DCR L
00000424 SYSIN          0609H MOV AM ORA M  MOV MA MOV CM INR C  MOV AC DAA    MOV MA SUB I
00000425 SYSIN          0612H 24H    JNZ    4BH    06H    MOV MI 00H    DCR L  MOV AM ORA M
00000426 SYSIN          061BH MOV MA MOV CM INR C  MOV AC DAA    MOV MA DCR L  MOV EM MOV DI
00000427 SYSIN          0624H 00H    LXI H  AEH    05H    DAD D  MOV CA MOV AM SUB C  JNC
00000428 SYSIN          062DH 4BH    06H    LXI H  CAH    20H    MOV MI 01H    MOV AM ORA M
00000429 SYSIN          0636H MOV MA DCR L  MOV CM INR C  MOV AC DAA    MOV MA SUB I  13H
00000430 SYSIN          063FH JNZ    4BH    06H    DCR L  MOV CM INR C  MOV AC DAA    MOV MA
00000431 SYSIN          0648H INR L  MOV MI 01H    XRA A  OUT    FBH    POP A  POP B  POP D
00000432 SYSIN          0651H POP H  EI     RET    JMP    62H    06H
00000433 SYSIN          0657H 20H 44H 45H 54H 45H 43H 54H 4FH 52H 20H 20H
00000434 SYSIN          0662H LXI B  57H    06H    MOV EI 0BH    CALL   09H    04H    DCR H
00000435 SYSIN          066BH MOV LI B2H    MOV AM ADD I  31H    MOV CA CALL   3CH    03H
00000436 SYSIN          0674H RET    LXI H  49H    20H    MOV MI 00H    INR H  INR L  MOV MI
00000437 SYSIN          067DH 00H    MOV AI 07H    LXI H  4AH    21H    SUB M  JC     A5H
00000438 SYSIN          0686H 06H    MOV CM MOV BI 00H    DCR H  DAD B  MOV AM LXI H  4AH
00000439 SYSIN          068FH 21H    MOV EM MOV DI 00H    LXI H  06H    01H    DAD D  MOV CA
00000440 SYSIN          0698H MOV AM ANA C  LXI H  49H    20H    ORA M  MOV MA INR H  INR L
```

PLM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
00000441 SYSIN   06A1H INR B  JNZ    7EH    06H    DCR H  DCR L  MOV AM OUT    07H
00000442 SYSIN   06AAH RET    LXI B  [illegible]  20H    MOV MI 00H    INR H  MOV LI 4BH
00000443 SYSIN   06B3H MOV MI 0CH    MOV AI 07H    LXI H  4BH    21H    SUB M  JC
00000444 SYSIN   06BCH DFB    C6H    MOV CM MOV BI 00H    DCR H  MOV LI 52H    DAD B
00000445 SYSIN   06C5H MOV AM LXI H  4BH    21H    MOV EM MOV DI 00H    LXI H  06H
00000446 SYSIN   06CEH 01H    DAD D  MOV CA MOV AM ANA C  LXI H  49H    20H    ORA M
00000447 SYSIN   06D7H MOV MA INR H  MOV LI 4BH    INR M  JNZ    B5H    C6H    DCR H
00000448 SYSIN   06E0H MOV LI 49H    MOV AM OUT    04H    RET    JMP    FFH    06H
00000449 SYSIN   06E9H 20H 53H 4FH 4CH 45H 4EH 4FH 49H 44H 20H 4EH 4FH 54H 20H 52H 45H
00000450 SYSIN   06F9H 4CH 45H 41H 53H 45H 44H
00000451 SYSIN   06FFH LXI H  B1H    20H    MOV MI 0CH    LXI B  05H    00H    MOV LI
00000452 SYSIN   0708H 4AH    DAD B  MOV MI 00H    CALL   75H    06H    IN     07H
00000453 SYSIN   0711H LXI D  05H    00H    LXI H  06H    01H    DAD D  MOV CA MOV AM
00000454 SYSIN   071AH ANA C  SUB I  00H    JNZ    33H    07H    CALL   54H    06H
00000455 SYSIN   0723H LXI B  E9H    06H    MOV EI 16H    CALL   09H    04H    DCR H
00000456 SYSIN   072CH MOV LI B1H    MOV MI FFH    CALL   54H    06H    RET    JMP
00000457 SYSIN   0735H 60H    07H
00000458 SYSIN   0737H 20H 46H 49H 4CH 54H 45H 52H 20H 53H 55H 50H 50H 4CH 59H 20H 45H
00000459 SYSIN   0747H 58H 48H 41H 55H 53H 54H 45H 44H 0DH 0AH
00000460 SYSIN   0751H 20H 46H 49H 4CH 54H 45H 52H 20H 41H 44H 56H 41H 4EH 43H 45H 20H
00000461 SYSIN   0761H 45H 52H 52H 4FH 52H 0DH 0AH
00000462 SYSIN   0768H LXI H  B1H    20H    MOV MI 00H    LXI B  04H    00H    MOV LI
00000463 SYSIN   0771H 4AH    DAD B  PUSH H LXI B  04H    00H    LXI H  C6H    01H
00000464 SYSIN   077AH DAD B  MOV AM POP H  MOV MA CALL   E6H    06H    LXI H  B1H
00000465 SYSIN   0783H 20H    MOV AM SUB I  00H    JNZ    3EH    08H    INR H  MOV LI
00000466 SYSIN   078CH 4CH    MOV MI 00H    INX H  MOV MI 00H    MOV AI 7EH    MOV BI
00000467 SYSIN   0795H 04H    LXI H  4CH    21H    SUB M  INR L  MOV CA MOV AB SBC M
00000468 SYSIN   079EH JC     3EH    08H    MOV AI F4H    MOV BI 01H    DCR L  SUB M
00000469 SYSIN   07A7H INR L  MOV CA MOV AB SBC M  JNC    B7H    07H    LXI B  05H
00000470 SYSIN   07B0H 00H    DCR H  MOV LI 4AH    DAD B  MOV MI FFH    LXI B  03H
00000471 SYSIN   07B9H 00H    LXI H  4AH    20H    DAD B  MOV MI 00H    CALL   75H
00000472 SYSIN   07C2H 06H    CALL   7FH    04H    LXI B  03H    00H    INR L  DAD B
00000473 SYSIN   07CBH PUSH H LXI B  03H    00H    LXI H  06H    01H    DAD B  MOV AM
00000474 SYSIN   07D4H POP H  MOV MA CALL   75H    06H    CALL   9FH    04H    IN
00000475 SYSIN   07DDH 07H    LXI D  07H    00H    LXI H  06H    01H    DAD D  MOV CA
00000476 SYSIN   07E6H MOV AM ANA C  MOV EA PUSH D LXI B  07H    00H    LXI H  06H
00000477 SYSIN   07EFH 01H    DAD B  MOV AM POP D  SUB E  SUB I  01H    SBC A  MOV EA
00000478 SYSIN   07F8H MOV AI 64H    LXI H  4CH    21H    SUB M  INR L  MOV EA MOV AI
00000479 SYSIN   0801H 00H    SBC M  SBC A  ANA C  RRC    JNC    20H    08H    IN
00000480 SYSIN   080AH 07H    LXI D  04H    00H    LXI H  06H    01H    DAD D  MOV CA
00000481 SYSIN   0813H MOV AM ANA C  SUB I  00H    JNZ    2AH    08H    CALL   54H
00000482 SYSIN   081CH 06H    LXI B  37H    07H    MOV EI 1AH    CALL   09H    04H
00000483 SYSIN   0825H DCR H  MOV LI B1H    MOV MI FFH    JMP    49H    08H    DCR L
00000484 SYSIN   082EH MOV CM INR L  MOV BM LXI H  01H    00H    DAD B  SHLD   4CH
00000485 SYSIN   0837H 21H    JMP    92H    07H    CALL   54H    06H    LXI B  51H
00000486 SYSIN   0840H 07H    MOV EI 17H    CALL   09H    04H    DCR H  MOV LI B1H
00000487 SYSIN   0849H MOV MI FFH    LXI B  04H    00H    LXI H  4AH    20H    DAD B
00000488 SYSIN   0852H MOV MI 0CH    LXI B  05H    00H    LXI H  4AH    20H    DAD B
00000489 SYSIN   085BH MOV MI 00H    CALL   75H    06H    MOV LI B1H    MOV AM SUB I
00000490 SYSIN   0864H FFH    CZ     54H    04H    RET    JMP    27H    32H    RET
00000491 SYSIN   086DH LXI H  4EH    21H    MOV MC INX H  MOV MB JMP    E2H    31H
00000492 SYSIN   0876H RET    LXI H  50H    21H    MOV MC INX H  MOV MB JMP    E7H
00000493 SYSIN   087FH 31H    RET    LXI H  52H    21H    MOV MC INX H  MOV MB JMP
00000494 SYSIN   0888H ECH    31H    RET    LXI H  54H    21H    MOV MC INX H  MOV MB
00000495 SYSIN   0891H JMP    F1H    31H    RET    LXI H  56H    21H    MOV MC INX H
00000496 SYSIN   089AH MOV MB JMP    DDH    31H    RET    LXI H  58H    21H    MOV MC
00000497 SYSIN   08A3H INX H  MOV MB JMP    D3H    31H    RET    LXI H  5AH    21H
00000498 SYSIN   08ACH MOV MC INX H  MOV MB JMP    F6H    31H    RET    LXI H  5CH
```

PIM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
00000499 SYSIN    08B5H 21H    MOV MC INX H  MOV MB JMP    FBH    31H    RET    LXI H
00000500 SYSIN    08BEH 5EH    21H    MOV MC INX H  MOV MB INR L  MOV ME INX H  MOV MD
00000501 SYSIN    08C7H MOV LI 5EH    MOV CM INR L  MOV BM CALL   A9H    08H    LXI H
00000502 SYSIN    08D0H 60H    21H    MOV CM INR L  MOV BM CALL   9FH    08H    RET
00000503 SYSIN    08D9H LXI H  CEH    20H    MOV MI 00H    MOV AI 02H    LXI H  CEH
00000504 SYSIN    08E2H 20H    SUB M  JC     9BH    09H    LXI B  A9H    21H    MOV LI
00000505 SYSIN    08EBH 46H    MOV MC INR L  MOV MB MOV EI 04H    MOV DI 00H    LXI H
00000506 SYSIN    08F4H CEH    20H    MOV CM MOV BI 00H    JMP    1FH    09H    MOV AC
00000507 SYSIN    08FDH SUB E  MOV AB SBC D  JP     08H    09H    MOV HB MOV IC XCHG
00000508 SYSIN    0906H MOV BH MOV CL LXI H  00H    00H    XCHG   MOV AB ORA C  RZ
00000509 SYSIN    090FH XCHG   MOV AB RAR    MOV BA MOV AC RAR    MOV CA JNC    1AH
00000510 SYSIN    0918H 09H    DAD D  XCHG   DAD H  JMP    0CH    09H    CALL   FCH
00000511 SYSIN    0921H 08H    LXI H  46H    20H    MOV CM INR L  MOV BM XCHG   DAD B
00000512 SYSIN    092AH SHLD   92H    21H    LXI B  0CH    00H    DAD B  SHLD   94H
00000513 SYSIN    0933H 21H    LXI B  9DH    21H    LXI H  44H    20H    MOV MC INR L
00000514 SYSIN    093CH MOV MB MOV EI 04H    MOV DI 00H    LXI H  CFH    20H    MOV CM
00000515 SYSIN    0945H MOV BI 00H    CALL   FCH    08H    LXI H  44H    20H    MOV CM
00000516 SYSIN    094EH INR L  MOV BM XCHG   DAD B  SHLD   96H    21H    LXI H  94H
00000517 SYSIN    0957H 21H    MOV CM INR L  MOV BM CALL   95H    08H    LXI H  92H
00000518 SYSIN    0960H 21H    MOV CM INR L  MOV BM CALL   77H    08H    LXI H  96H
00000519 SYSIN    0969H 21H    MOV CM INR L  MOV BM CALL   9FH    08H    LXI B  5AH
00000520 SYSIN    0972H 20H    LXI H  42H    20H    MOV MC INR L  MOV MB MOV EI 0DH
00000521 SYSIN    097BH MOV DI 0CH    LXI H  CFH    20H    MOV CM MOV BI 00H    CALL
00000522 SYSIN    0984H FCH    08H    LXI H  42H    20H    MOV CM INR L  MOV BM XCHG
00000523 SYSIN    098DH DAD B  XCHG   MOV CE MOV BD CALL   B3H    08H    LXI H  CFH
00000524 SYSIN    0996H 20H    INR M  JNZ    DEH    08H    RET    CALL   69H    08H
00000525 SYSIN    099FH LXI H  F1H    21H    MOV MI 00H    MOV AI 08H    LXI H  F1H
00000526 SYSIN    09A8H 21H    SUB M  JC     0FH    0AH    MOV EI 10H    MOV DI 00H
00000527 SYSIN    09B1H LXI H  F1H    21H    MOV CM MOV BI 00H    CALL   FCH    08H
00000528 SYSIN    09BAH LXI B  10H    01H    XCHG   DAD B  XCHG   LXI H  40H    20H
00000529 SYSIN    09C3H MOV ME INR L  MOV MD LXI H  B2H    20H    MOV EM MOV DI 00H
00000530 SYSIN    09CCH MOV CI C0H    MOV BI 00H    CALL   FCH    08H    LXI H  40H
00000531 SYSIN    09D5H 20H    MOV CM INR L  MOV BM XCHG   DAD B  XCHG   LXI B  C1H
00000532 SYSIN    09DEH 21H    LXI H  3CH    20H    MOV MC INR L  MOV MB INR L  MOV ME
00000533 SYSIN    09E7H INR L  MOV MD MOV EI 04H    MOV DI 00H    LXI H  F1H    21H
00000534 SYSIN    09F0H MOV CM MOV BI 00H    CALL   FCH    08H    LXI H  3CH    20H
00000535 SYSIN    09F9H MOV CM INR L  MOV BM XCHG   DAD B  XCHG   LXI H  3EH    20H
00000536 SYSIN    0A02H MOV CM INR L  MOV BM CALL   BDH    08H    LXI H  F1H    21H
00000537 SYSIN    0A0BH INR M  JNZ    A4H    09H    DCR H  MOV LI B2H    MOV AM SUB I
00000538 SYSIN    0A14H 0CH    JNZ    44H    0AH    INR H  MOV LI F1H    MOV MI 00H
00000539 SYSIN    0A1DH MOV AI 0DH    LXI H  F1H    21H    SUB M  JC     6DH    0AH
00000540 SYSIN    0A26H MOV CM MOV BI 00H    DCR H  MOV LI D9H    DAD B  PUSH H LXI H
00000541 SYSIN    0A2FH F1H    21H    MOV CM MOV BI 00H    MOV LI 23H    DAD B  MOV AM
00000542 SYSIN    0A38H POP H  MOV MA LXI H  F1H    21H    INR M  JNZ    1DH    0AH
00000543 SYSIN    0A41H JMP    6DH    0AH    INR H  MOV LI F1H    MOV MI 00H    MOV AI
00000544 SYSIN    0A4AH 0DH    LXI H  F1H    21H    SUB M  JC     6DH    0AH    MOV CM
00000545 SYSIN    0A53H MOV BI 00H    DCR H  MOV LI D9H    DAD B  PUSH H LXI H  F1H
00000546 SYSIN    0A5CH 21H    MOV CM MOV BI 00H    MOV LI 30H    DAD B  MOV AM POP H
00000547 SYSIN    0A65H MOV MA LXI H  F1H    21H    INR M  JNZ    49H    0AH    LXI H
00000548 SYSIN    0A6EH D9H    20H    LXI D  99H    21H    CALL   BDH    08H    LXI H
00000549 SYSIN    0A77H F1H    21H    MOV MI 00H    MOV AI 05H    LXI H  F1H    21H
00000550 SYSIN    0A80H SUB M  JC     D6H    0AH    LXI B  81H    20H    DCR H  MOV LI
00000551 SYSIN    0A89H 3AH    MOV MC INR L  MOV MB MOV EI 08H    MOV DI 00H    LXI H
00000552 SYSIN    0A92H F1H    21H    MOV CM MOV BI 00H    CALL   FCH    08H    LXI H
00000553 SYSIN    0A9BH 3AH    20H    MOV CM INR L  MOV BM XCHG   DAD B  XCHG   LXI B
00000554 SYSIN    0AA4H A9H    21H    LXI H  36H    20H    MOV MC INR L  MOV MB INR L
00000555 SYSIN    0AADH MOV ME INR L  MOV MD MOV EI 04H    MOV DI 00H    LXI H  F1H
00000556 SYSIN    0AB6H 21H    MOV CM MOV BI 00H    CALL   FCH    08H    LXI H  36H
```

PLM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
0ABFH 20H    MOV CM INR L  MOV BM XCHG   DAD B  XCHG   LXI H  38H
0AC8H 20H    MOV CM INR L  MOV BM CALL   BDH    08H    LXI H  F1H
0AD1H 21H    INR M  JNZ    7BH    0AH    CALL   D9H    08H    INR H
0ADAH MOV LI F2H    MOV MI 00H    MOV AI 03H    LXI H  F2H    21H
0AE3H SUB M  JC     0CH    0CH    DCR L  MOV MI 00H    MOV AI 08H
0AECH LXI H  F1H    21H    SUB M  JC     6EH    0BH    JMP    00H
0AF5H 0BH    LXI H  F1H    21H    MOV AM ADD I  04H    MOV MA JMP
0AFEH EAH    0AH    LXI B  9DH    21H    MOV EM MOV DI 00H    MOV LC
0B07H MOV HB DAD D  XCHG   MOV CE MOV BD CALL   95H    08H    LXI B
0B10H C1H    21H    LXI H  F1H    21H    MOV EM MOV DI 00H    MOV LC
0B19H MOV HB DAD D  XCHG   LXI H  34H    20H    MOV ME INR L  MOV MD
0B22H MOV EI 0CH    MOV DI 00H    LXI H  F2H    21H    MOV CM MOV BI
0B2BH 00H    CALL   FCH    08H    LXI H  34H    20H    MOV CM INR L
0B34H MOV BM XCHG   DAD B  XCHG   MOV CE MOV BD CALL   81H    0BH
0B3DH LXI B  62H    21H    LXI H  F1H    21H    MOV EM MOV DI 00H
0B46H MOV LC MOV HB DAD D  XCHG   LXI H  32H    20H    MOV ME INR L
0B4FH MOV MD MOV EI 0CH    MOV DI 00H    LXI H  F2H    21H    MOV CM
0B58H MOV BI 00H    CALL   FCH    08H    LXI H  32H    20H    MOV CM
0B61H INR L  MOV BM XCHG   DAD B  XCHG   MOV CE MOV BD CALL   9EH
0B6AH 08H    JMP    F6H    0AH    LXI B  62H    21H    DCR H  MOV LI
0B73H 30H    MOV MC INR L  MOV MB MOV EI 0CH    MOV DI 00H    LXI H
0B7CH F2H    21H    MOV CM MOV BI 00H    CALL   FCH    08H    LXI H
0B85H 30H    20H    MOV CM INR L  MOV BM XCHG   DAD B  XCHG   MOV CE
0B8EH MOV BD CALL   95H    08H    LXI H  F1H    21H    MOV MI 04H
0B97H MOV AI 08H    LXI H  F1H    21H    SUB M  JC     DBH    0BH
0BA0H JMP    ADH    0BH    LXI H  F1H    21H    MOV AM ADD I  04H
0BA9H MOV MA JMP    97H    0BH    LXI B  62H    21H    MOV EM MOV DI
0BB2H 00H    MOV LC MOV HB DAD D  XCHG   LXI H  2EH    20H    MOV ME
0BBBH INR L  MOV MD MOV EI 0CH    MOV DI 00H    LXI H  F2H    21H
0BC4H MOV CM MOV BI 00H    CALL   FCH    08H    LXI H  2EH    20H
0BCDH MOV CM INR L  MOV BM XCHG   DAD B  XCHG   MOV CE MOV BD CALL
0BD6H 6DH    08H    JMP    A3H    0BH    LXI B  99H    21H    CALL
0BDFH 81H    08H    LXI B  E6H    20H    LXI H  2CH    20H    MOV MC
0BE8H INR L  MOV MB MOV EI 0DH    MOV DI 00H    LXI H  F2H    21H
0BF1H MOV CM MOV BI 00H    CALL   FCH    08H    LXI H  2CH    20H
0BFAH MOV CM INR L  MOV BM XCHG   DAD B  XCHG   MOV CE MOV BD CALL
0C03H B3H    08H    LXI H  F2H    21H    INR M  JNZ    DEH    0AH
0C0CH RET
0C0DH 57H 4FH 52H 4BH 49H 4EH 47H 20H 4CH 45H 56H 45H 4CH 20H 20H 20H
0C1DH 20H 20H 20H 20H 20H 20H 20H 20H 52H 41H 44H 49H 55H 4DH 20H 41H
0C2DH 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H
0C3DH 52H 41H 44H 49H 55H 4DH 20H 42H 20H 20H 20H 20H 20H 20H 20H 20H
0C4DH 20H 20H 20H 20H 20H 20H 20H 20H 52H 41H 44H 49H 55H 4DH 20H 43H
0C5DH 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H
0C6DH 20H 20H 20H 20H 20H 20H 50H 43H 2FH 4CH 20H 50H 43H 2FH 4CH 20H
0C7DH 50H 43H 2FH 4CH
0C81H 0DH 0AH 0AH 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H
0C91H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H
0CA1H 2AH 2AH 2AH 20H 52H 20H 57H 20H 4CH 20H 4DH 20H 2AH 2AH 2AH 0DH
0CB1H 0AH 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H
0CC1H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H
0CD1H 52H 41H 44H 49H 55H 4DH 20H 41H 20H 20H 20H 20H 20H 20H 20H 20H
0CE1H 52H 41H 44H 49H 55H 4DH 20H 28H 42H 29H 43H 29H 20H 20H 20H 20H
0CF1H 52H 41H 44H 49H 55H 4DH 20H 43H 27H 20H 20H
0CFCH 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H 20H
0D0CH 53H 41H 4DH 50H 4CH 45H 20H 43H 4FH 55H 4EH 54H 53H 20H 20H 20H
0D1CH 20H 20H 20H 20H 20H 20H 20H 42H 41H 43H 4BH 47H 52H 4FH 55H 4EH
0D2CH 44H 20H 43H 4FH 55H 4EH 54H 53H 20H 20H 20H 20H 20H 4EH 45H 54H
```

PLM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
00000615 SYSIN   0D3CH 20H 43H 4FH 55H 4EH 54H 53H 0DH 0AH
00000616 SYSIN   0D45H 0DH 0AH 45H 4EH 54H 45H 52H 20H 41H 4CH 54H 49H 54H 55H 44H 45H
00000617 SYSIN   0D55H 20H 43H 4FH 52H 52H 45H 43H 54H 49H 4FH 4EH 20H 46H 41H 43H 54H
00000618 SYSIN   0D65H 4FH 52H 3DH 20H
00000619 SYSIN   0D69H LXI H  F3H    21H    MOV MI 00H    MOV AI 2FH    LXI H  F3H
00000620 SYSIN   0D72H 21H    SUB M  JC     95H    0DH    MOV CM MOV BI 00H    DCR H
00000621 SYSIN   0D7BH MOV LI 81H    DAD B  PUSH H LXI H  F3H    21H    MOV CM MOV BI
00000622 SYSIN   0D84H 00H    DCR H  MOV LI 81H    DAD B  MOV AM ADD I  30H    POP H
00000623 SYSIN   0D8DH MOV MA LXI H  F3H    21H    INR M  JNZ    6EH    0DH    MOV MI
00000624 SYSIN   0D96H 00H    MOV AI 26H    LXI H  F3H    21H    SUB M  JC     BFH
00000625 SYSIN   0D9FH 0DH    MOV CM MOV BI 00H    DCR H  MOV LI 5AH    DAD B  PUSH H
00000626 SYSIN   0DA8H LXI H  F3H    21H    MOV CM MOV BI 00H    DCR H  MOV LI 5AH
00000627 SYSIN   0DB1H DAD B  MOV AM ADD I  30H    POP H  MOV MA LXI H  F3H    21H
00000628 SYSIN   0DBAH INR M  JNZ    97H    0DH    MOV LI 21H    MOV AM SUB I  FFH
00000629 SYSIN   0DC3H JZ     BBH    0EH    MOV LI F3H    MOV MI 00H    MOV AI 33H
00000630 SYSIN   0DCCH LXI H  F3H    21H    SUB M  JC     F1H    0DH    MOV CM MOV BI
00000631 SYSIN   0DD5H 00H    DCR H  MOV LI E6H    DAD B  PUSH H LXI H  F3H    21H
00000632 SYSIN   0DDEH MOV CM MOV BI 00H    DCR H  MOV LI E6H    DAD B  MOV AM ADD I
00000633 SYSIN   0DE7H 30H    POP H  MOV MA LXI H  F3H    21H    INR M  JNZ    CAH
00000634 SYSIN   0DF0H 0DH    LXI B  81H    0CH    MOV EI 50H    CALL   09H    04H
00000635 SYSIN   0DF9H CALL   54H    06H    CALL   D6H    03H    CALL   9FH    04H
00000636 SYSIN   0E02H LXI B  08H    00H    LXI H  45H    0DH    DAD B  XCHG   MOV CE
00000637 SYSIN   0E0BH MOV BD MOV EI 1BH    CALL   09H    04H    MOV LI F3H    MOV MI
00000638 SYSIN   0E14H 00H    LXI H  3FH    21H    MOV CM DCR C  MOV AC MOV LI F3H
00000639 SYSIN   0E1DH SUB M  JC     3AH    0EH    MOV CM MOV BI 00H    DCR H  MOV LI
00000640 SYSIN   0E26H D9H    DAD B  MOV AM ADD I  30H    LXI H  CFH    20H    MOV MA
00000641 SYSIN   0E2FH MOV CM CALL   3CH    03H    LXI H  F3H    21H    INR M  JNZ
00000642 SYSIN   0E38H 15H    0EH    CALL   D6H    03H    CALL   D6H    03H    LXI H
00000643 SYSIN   0E41H F3H    21H    MOV MI 00H    MOV AI 03H    LXI H  F3H    21H
00000644 SYSIN   0E4AH SUB M  JC     B2H    0EH    MOV EI 18H    MOV DI 00H    LXI H
00000645 SYSIN   0E53H F3H    21H    MOV CM MOV BI 00H    CALL   FCH    08H    LXI B
00000646 SYSIN   0E5CH 0DH    0CH    XCHG   DAD B  XCHG   MOV CE MOV BD MOV EI 18H
00000647 SYSIN   0E65H CALL   09H    04H    LXI B  F6H    20H    DCR H  MOV LI 2AH
00000648 SYSIN   0E6EH MOV MC INR L  MOV MB MOV EI 0DH    MOV DI 00H    LXI H  F3H
00000649 SYSIN   0E77H 21H    MOV CM MOV BI 00H    CALL   FCH    08H    LXI H  2AH
00000650 SYSIN   0E80H 20H    MOV CM INR L  MOV BM XCHG   DAD B  XCHG   MOV CE MOV BD
00000651 SYSIN   0E89H MOV EI 0DH    CALL   09H    04H    MOV EI C5H    MOV DI 00H
00000652 SYSIN   0E92H LXI H  F3H    21H    MOV CM MOV BI 00H    CALL   FCH    08H
00000653 SYSIN   0E9BH LXI B  6DH    0CH    XCHG   DAD B  XCHG   MOV CE MOV BD MOV EI
00000654 SYSIN   0EA4H 05H    CALL   09H    04H    CALL   D6H    03H    LXI H  F3H
00000655 SYSIN   0EADH 21H    INR M  JNZ    45H    0EH    CALL   D6H    03H    CALL
00000656 SYSIN   0EB6H D6H    03H    CALL   D6H    03H    LXI B  FCH    0CH    MOV EI
00000657 SYSIN   0EBFH 49H    CALL   09H    04H    MOV LI F3H    MOV MI 00H    MOV AI
00000658 SYSIN   0EC8H 02H    LXI H  F3H    21H    SUB M  JC     8AH    0FH    MOV EI
00000659 SYSIN   0ED1H 10H    MOV DI 00H    LXI H  F3H    21H    MOV CM MOV BI 00H
00000660 SYSIN   0EDAH CALL   FCH    08H    LXI B  D1H    0CH    XCHG   DAD B  XCHG
00000661 SYSIN   0EE3H MOV CE MOV BD MOV EI 10H    CALL   09H    04H    LXI B  81H
00000662 SYSIN   0EECH 20H    DCR H  MOV LI 28H    MOV MC INR L  MOV MB MOV EI 08H
00000663 SYSIN   0EF5H MOV DI 00H    LXI H  F3H    21H    MOV CM MOV BI 00H    CALL
00000664 SYSIN   0EFEH FCH    08H    LXI H  28H    20H    MOV CM INR L  MOV BM XCHG
00000665 SYSIN   0F07H DAD B  XCHG   LXI B  18H    00H    XCHG   DAD B  XCHG   MOV CE
00000666 SYSIN   0F10H MOV BD MOV EI 07H    CALL   09H    04H    LXI B  03H    00H
00000667 SYSIN   0F19H LXI H  81H    0CH    DAD B  XCHG   MOV CE MOV BD MOV EI 10H
00000668 SYSIN   0F22H CALL   09H    04H    LXI B  81H    20H    DCR H  MOV LI 26H
00000669 SYSIN   0F2BH MOV MC INR L  MOV MB MOV EI 08H    MOV DI 00H    LXI H  F3H
00000670 SYSIN   0F34H 21H    MOV CM MOV BI 00H    CALL   FCH    08H    LXI H  26H
00000671 SYSIN   0F3DH 20H    MOV CM INR L  MOV BM XCHG   DAD B  XCHG   MOV CE MOV BD
00000672 SYSIN   0F46H MOV EI 07H    CALL   09H    04H    LXI B  03H    00H    LXI H
```

PLM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
00000673 SYSIN   0F4FH 81H     0CH     DAD B   XCHG    MOV CE  MOV BD  MOV EI  10H     CALL
00000674 SYSIN   0F58H 09H     04H     LXI B   5AH     20H     DCR H   MOV LI  24H     MOV MC
00000675 SYSIN   0F61H INR L   MOV MB  MOV EI  0DH     MOV DI  00H     LXI H   F3H     21H
00000676 SYSIN   0F6AH MOV CM  MOV BI  00H     CALL    FCH     08H     LXI H   24H     20H
00000677 SYSIN   0F73H MOV CM  INR L   MOV BM  XCHG    DAD B   XCHG    MOV CE  MOV BD  MOV EI
00000678 SYSIN   0F7CH 0DH     CALL    09H     04H     CALL    D6H     03H     LXI H   F3H
00000679 SYSIN   0F85H 21H     INR M   JNZ     C7H     0FH     CALL    D6H     03H     CALL
00000680 SYSIN   0F8EH D6H     03H     LXI H   F3H     21H     MOV MI  00H     MOV AI  2FH
00000681 SYSIN   0F97H LXI H   F3H     21H     SUB M   JC      BCH     0FH     MOV CM  MOV BI
00000682 SYSIN   0FA0H 00H     DCR H   MOV LI  81H     DAD B   PUSH H  LXI H   F3H     21H
00000683 SYSIN   0FA9H MOV CM  MOV BI  00H     DCR H   MOV LI  81H     DAD B   MOV AM  SUB I
00000684 SYSIN   0FB2H 30H     POP H   MOV MA  LXI H   F3H     21H     INR M   JNZ     95H
00000685 SYSIN   0FBBH 0FH     MOV MI  00H     MOV AI  26H     LXI H   F3H     21H     SUB M
00000686 SYSIN   0FC4H JC      E5H     0FH     MOV CM  MOV BI  00H     DCR H   MOV LI  5AH
00000687 SYSIN   0FCDH DAD B   PUSH H  LXI H   F3H     21H     MOV CM  MOV BI  00H     DCR H
00000688 SYSIN   0FD6H MOV LI  5AH     DAD B   MOV AM  SUB I   30H     POP H   MOV MA  LXI H
00000689 SYSIN   0FDFH F3H     21H     INR M   JNZ     BEH     0FH     MOV MI  00H     MOV AI
00000690 SYSIN   0FE8H 33H     LXI H   F3H     21H     SUB M   JC      0FH     10H     MOV CM
00000691 SYSIN   0FF1H MOV BI  00H     DCR H   MOV LI  E6H     DAD B   PUSH H  LXI H   F3H
00000692 SYSIN   0FFAH 21H     MOV CM  MOV BI  00H     DCR H   MOV LI  E6H     DAD B   MOV AM
00000693 SYSIN   1003H SUB I   30H     POP H   MOV MA  LXI H   F3H     21H     INR M   JNZ
00000694 SYSIN   100CH E7H     0FH     RET     LXI H   20H     21H     MOV AM  SUB I   FFH
00000695 SYSIN   1015H JZ      63H     10H     LXI B   03H     00H     DCR H   MOV LI  52H
00000696 SYSIN   101EH DAD B   MOV MI  00H     LXI B   04H     00H     LXI H   52H     20H
00000697 SYSIN   1027H DAD B   MOV MI  00H     CALL    ABH     06H     LXI B   03H     00H
00000698 SYSIN   1030H MOV LI  52H     DAD B   PUSH H  LXI B   03H     00H     LXI H   06H
00000699 SYSIN   1039H 01H     DAD B   MOV AM  POP H   MOV MA  LXI B   04H     00H     LXI H
00000700 SYSIN   1042H 52H     20H     DAD B   PUSH H  LXI B   04H     00H     LXI H   06H
00000701 SYSIN   104BH 01H     DAD B   MOV AM  POP H   MOV MA  LXI H   B8H     20H     MOV AM
00000702 SYSIN   1054H INR L   MOV BM  MOV LI  D4H     SUB M   INR L   MOV CA  MOV AB  SBC M
00000703 SYSIN   105DH JC      50H     10H     CALL    ABH     06H     LXI B   04H     00H
00000704 SYSIN   1066H LXI H   52H     20H     DAD B   MOV MI  00H     DI      LXI H   B8H
00000705 SYSIN   106FH 20H     MOV CM  INR L   MOV BM  INR H   MOV LI  F4H     MOV MC  INX H
00000706 SYSIN   1078H MOV MB  EI      DCR L   MOV AM  INR L   MOV BM  DCR H   MOV LI  D6H
00000707 SYSIN   1081H SUB M   INR L   MOV CA  MOV AB  SBC M   JC      6CH     10H     CALL
00000708 SYSIN   108AH ABH     06H     RET     LXI H   F8H     21H     MOV MC  INX H   MOV MB
00000709 SYSIN   1093H INR L   MOV ME  INX H   MOV MD  LXI H   06H     01H     MOV CM  LXI H
00000710 SYSIN   109CH 52H     20H     MOV MC  CALL    ABH     06H     MOV LI  52H     MOV MI
00000711 SYSIN   10A5H 00H     CALL    ABH     06H     LXI B   06H     00H     LHLD    F6H
00000712 SYSIN   10AEH 21H     DAD B   XCHG    IN      04H     PUSH D  MOV EI  04H     ORA A
00000713 SYSIN   10B7H RAR     DCR E   JNZ     B6H     10H     POP H   MOV MA  LXI B   06H
00000714 SYSIN   10C0H 00H     LHLD    F8H     21H     DAD B   XCHG    IN      05H     PUSH D
00000715 SYSIN   10C9H MOV EI  04H     ORA A   RAR     DCR E   JNZ     CBH     10H     POP H
00000716 SYSIN   10D2H MOV MA  LXI B   06H     00H     LHLD    FAH     21H     DAD B   XCHG
00000717 SYSIN   10DBH IN      06H     PUSH D  MOV EI  04H     ORA A   RAR     DCR E   JNZ
00000718 SYSIN   10E4H E0H     10H     POP H   MOV MA  LXI H   52H     20H     INX H   PUSH H
00000719 SYSIN   10EDH LXI H   06H     01H     INX H   MOV AM  POP H   MOV MA  CALL    ABH
00000720 SYSIN   10F6H 06H     MOV LI  52H     INX H   MOV MI  00H     CALL    ABH     06H
00000721 SYSIN   10FFH INR H   MOV LI  FCH     MOV MI  00H     MOV AI  05H     LXI H   FCH
00000722 SYSIN   1108H 21H     SUB M   JC      7FH     11H     LXI H   05H     01H     MOV CM
00000723 SYSIN   1111H LXI H   52H     20H     MOV MC  CALL    ABH     06H     MOV LI  52H
00000724 SYSIN   111AH MOV MI  00H     CALL    ABH     06H     INR H   MOV LI  FCH     MOV CM
00000725 SYSIN   1123H MOV BI  00H     LHLD    F6H     21H     DAD B   XCHG    IN      04H
00000726 SYSIN   112CH CMA     ANA I   0FH     STAX D  LXI H   FCH     21H     MOV CM  MOV BI
00000727 SYSIN   1135H 00H     LHLD    F8H     21H     DAD B   XCHG    IN      05H     CMA
00000728 SYSIN   113EH ANA I   0FH     STAX D  LXI H   FCH     21H     MOV CM  MOV BI  00H
00000729 SYSIN   1147H LHLD    FAH     21H     DAD B   XCHG    IN      06H     CMA     ANA I
00000730 SYSIN   1150H 0FH     STAX D  LXI B   02H     00H     LXI H   52H     20H     DAD B
```

```
00000731 SYSIN    1159H PUSH H LXI B  02H    00H    LXI H  06H    01H    DAD B  MOV AM
00000732 SYSIN    1162H POP H  MOV MA CALL   ABH    06H    CALL   7FH    04H    LXI B
00000733 SYSIN    116BH 02H    00H    MOV LI 52H    DAD B  MOV MI 00H    CALL   ABH
00000734 SYSIN    1174H 06H    CALL   7FH    04H    INR H  MOV LI FCH    INR M  JNZ
00000735 SYSIN    117DH 04H    11H    LXI B  07H    00H    LHLD   F6H    21H    DAD B
00000736 SYSIN    1186H XCHG   MOV AI 24H    SUB I  30H    STAX D LXI B  07H    00H
00000737 SYSIN    118FH LHLD   F8H    21H    DAD B  XCHG   MOV AI 24H    SUB I  30H
00000738 SYSIN    1198H STAX D LXI B  07H    00H    LHLD   FAH    21H    DAD B  XCHG
00000739 SYSIN    11A1H MOV AI 24H    SUB I  30H    STAX D RET
00000740 SYSIN    11A7H 20H 46H 4CH 4FH 57H 20H 52H 41H 54H 45H 20H 4FH 55H 54H 20H 4FH
00000741 SYSIN    11B7H 46H 20H 52H 41H 4EH 47H 45H 20H
00000742 SYSIN    11BFH IN     07H    LXI D  03H    00H    LXI H  06H    01H    DAD D
00000743 SYSIN    11C8H MOV CA MOV AM ANA C  SUB I  00H    JNZ    E1H    11H    CALL
00000744 SYSIN    11D1H 75H    06H    CALL   54H    06H    LXI B  A7H    11H    MOV EI
00000745 SYSIN    11DAH 18H    CALL   09H    04H    CALL   54H    04H    RET    JMP
00000746 SYSIN    11E3H F2H    11H
00000747 SYSIN    11E5H 20H 50H 4FH 57H 45H 52H 20H 46H 41H 49H 4CH 0DH 0AH
00000748 SYSIN    11F2H LXI H  B1H    20H    MOV MI 00H    IN     07H    LXI D  06H
00000749 SYSIN    11FBH 00H    LXI H  06H    01H    DAD D  MOV CA MOV AM ANA C  SUB I
00000750 SYSIN    1204H 00H    JNZ    18H    12H    CALL   54H    06H    LXI B  F5H
00000751 SYSIN    120DH 11H    MOV EI 0DH    CALL   09H    04H    DCR H  MOV LI B1H
00000752 SYSIN    1216H MOV MI FFH    RET    CALL   E2H    11H    LXI H  B1H    20H
00000753 SYSIN    121FH MOV AM RRC    CC     54H    04H    IN     07H    LXI D  07H
00000754 SYSIN    1228H 00H    LXI H  06H    01H    DAD D  MOV CA MOV AM ANA C  SUB I
00000755 SYSIN    1231H 00H    CZ     34H    07H    CALL   34H    07H    CALL   34H
00000756 SYSIN    123AH 07H    MOV LI B8H    MOV MI 00H    INX H  MOV MI 00H    MOV LI
00000757 SYSIN    1243H D4H    MOV MI 01H    INX H  MOV MI 00H    INR L  MOV MI 79H
00000758 SYSIN    124CH INX H  MOV MI 00H    CALL   0FH    10H    LXI B  81H    20H
00000759 SYSIN    1255H INR H  MOV LI F6H    MOV MC INX H  MOV MB LXI B  89H    20H
00000760 SYSIN    125EH LXI D  91H    20H    CALL   8DH    10H    LXI B  05H    00H
00000761 SYSIN    1267H LXI H  4AH    20H    DAD B  MOV MI FFH    LXI B  06H    00H
00000762 SYSIN    1270H LXI H  4AH    20H    DAD B  MOV MI FFH    LXI H  B8H    20H
00000763 SYSIN    1279H MOV AM INR L  MOV BM SUB I  7AH    MOV CA MOV AB SBC I  00H
00000764 SYSIN    1282H JC     76H    12H    CALL   75H    06H    LXI B  06H    00H
00000765 SYSIN    128BH INR L  DAD B  MOV MI 00H    LXI H  B8H    20H    MOV AM INR L
00000766 SYSIN    1294H MOV BM SUB I  C8H    MOV CA MOV AB SBC I  00H    JC     8FH
00000767 SYSIN    129DH 12H    CALL   BFH    11H    LXI H  B8H    20H    MOV AM INR L
00000768 SYSIN    12A6H MOV BM SUB I  F2H    MOV CA MOV AB SBC I  00H    JC     A1H
00000769 SYSIN    12AFH 12H    CALL   75H    06H    CALL   34H    07H    MOV LI D4H
00000770 SYSIN    12B8H MOV MI 00H    INX H  MOV MI 01H    INR L  MOV MI 78H    INX H
00000771 SYSIN    12C1H MOV MI 01H    CALL   0FH    10H    LXI B  99H    20H    INR H
00000772 SYSIN    12CAH MOV LI F6H    MOV MC INX H  MOV MB LXI B  A1H    20H    LXI D
00000773 SYSIN    12D3H A9H    20H    CALL   8DH    10H    LXI H  1FH    21H    MOV AM
00000774 SYSIN    12DCH SUB I  FFH    JZ     E7H    12H    CALL   9CH    09H    CALL
00000775 SYSIN    12E5H 69H    0DH    RET    LXI H  1FH    21H    MOV MI 00H    LXI H
00000776 SYSIN    12EEH BCH    20H    MOV AM MOV LI CCH    SUB M  ADD I  FFH    SBC A
00000777 SYSIN    12F7H MOV LI BDH    MOV CA MOV AM MOV LI CBH    SUB M  ADD I  FFH
00000778 SYSIN    1300H SBC A  ORA C  RRC    JC     EDH    12H    MOV LI D1H    MOV MI
00000779 SYSIN    1309H 00H    LXI H  D1H    20H    MOV AM CMA    RRC    JNC    A9H
00000780 SYSIN    1312H 13H    MOV LI B5H    MOV MI 00H    INR L  MOV MI 00H    MOV LI
00000781 SYSIN    131BH BEH    MOV AM RRC    JNC    38H    13H    MOV LI 4AH    MOV MI
00000782 SYSIN    1324H 00H    MOV LI B2H    MOV MI 00H    CALL   75H    06H    CALL
00000783 SYSIN    132DH 19H    12H    LXI H  B5H    20H    MOV AM SUB I  07H    JC
00000784 SYSIN    1336H 2FH    13H    MOV LI BEH    INX H  MOV AM RRC    JNC    4FH
00000785 SYSIN    133FH 13H    LXI H  4AH    20H    MOV MI FFH    MOV LI B2H    MOV MI
00000786 SYSIN    1348H 01H    CALL   75H    06H    CALL   19H    12H    LXI H  B5H
00000787 SYSIN    1351H 20H    MOV AM MOV LI BAH    SUB M  SUB I  01H    SBC A  MOV LI
00000788 SYSIN    135AH B6H    MOV CA MOV AM MOV LI BBH    SUB M  SUB I  01H    SBC A
```

PIM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
1363H ANA C  CMA      RRC      JC       4FH      13H      MOV LI C8H      MOV AM
136CH INR H  MOV LI 1AH      SUB M    SBC A    CMA      DCR H    MOV LI C9H
1375H MOV CA MOV AM INR H    MOV LI 1BH      SUB M    SBC A    CMA      ANA C
137EH DCR H  MOV LI CAH      MOV FA MOV AM INR H    MOV LI 1CH      SUB M
1387H SBC A  CMA      ANA E    DCR H    MOV LI CBH      MOV CA MOV AM INR H
1390H MOV LI 1DH      SUB M    SBC A    CMA      ANA C    DCR H    MOV LI CCH
1399H MOV EA MOV AM INR H    MOV LI 1EH      SUB M    SBC A    CMA      ANA E
13A2H DCR H  MOV LI D1H      MOV MA JMP      0AH      13H      RET      CALL
13ABH 9FH    04H      LXI B    90H      02H      MOV EI 13H      CALL     09H
13B4H 04H    CALL     7FH      03H      LXI H    C7H      20H      MOV AM SUB I
13BDH 4EH    JNZ      C7H      13H      CALL     41H      05H      JMP      CEH
13C6H 13H    XRA A    OUT      FBH      EI       CALL     D6H      03H      LXI H
13CFH B2H    20H      MOV MI 00H      MOV AI 01H      LXI H    B2H      20H
13D8H SUB M  JC       1CH      14H      LXI B    A3H      02H      MOV EI 0CH
13E1H CALL   09H      04H      CALL     54H      06H      LXI B    AFH      02H
13EAH MOV EI 0AH      CALL     09H      04H      CALL     7FH      03H      LXI H
13F3H CFH    20H      MOV AM SUB I    59H      JNZ      09H      14H      MOV LI
13FCH B2H    MOV CM MOV BI 00H      MOV LI B7H      DAD B    MOV MI FFH
1405H JMP    12H      14H      MOV LI B2H      MOV CM MOV BI 00H      MOV LI
140EH BEH    DAD B    MOV MI 00H      CALL     D6H      03H      LXI H    B2H
1417H 20H    INR M    JNZ      D3H      13H      MOV LI B7H      MOV MI FFH
1420H LXI H  B7H      20H      MOV AM SUB I    FFH      JNZ      85H      14H
1429H CALL   D6H      C3H      LXI B    CBH      02H      MOV EI 2AH      CALL
1432H 09H    04H      CALL     EAH      03H      MOV CM MOV LI BBH      MOV MC
143BH LXI B  F5H      02H      MOV EI 05H      CALL     09H      04H      CALL
1444H EAH    03H      MOV CM MOV LI BAH      MOV MC CALL     D6H      03H
144DH LXI H  BBH      20H      MOV AM CMP I    01H      MOV AI FFH      JC
1456H 59H    14H      XRA A    DCR L    MOV CA MOV AM SUB I    14H      SBC A
145FH ANA C  MOV EA MOV AI 59H      SUB M    SBC A    ORA E    RRC      JNC
1468H 71H    14H      MOV LI B7H      MOV MI FFH      JMP      75H      14H
1471H MOV LI B7H      MOV MI 00H      MOV AM RRC      JNC      20H      14H
147AH LXI B  FAH      02H      MOV EI 23H      CALL     09H      04H      JMP
1483H 20H    14H      LXI B    1DH      03H      MOV EI 19H      CALL     09H
148CH 04H    CALL     EAH      03H      MOV CM MOV LI BDH      MOV MC LXI B
1495H 36H    03H      MOV EI 06H      CALL     09H      04H      CALL     EAH
149EH 03H    MOV CM MOV LI BCH      MOV MC CALL     D6H      03H      LXI B
14A7H 39H    02H      MOV EI 12H      CALL     09H      04H      LXI B    01H
14B0H 05H    MOV EI 0DH      CALL     09H      04H      CALL     EAH      03H
14B9H MOV CM INR H    MOV LI 1AH      MOV MC LXI B    0EH      05H      MOV EI
14C2H 0DH    CALL     09H      04H      CALL     EAH      03H      MOV CM INR H
14CBH MOV LI 1BH      MOV MC LXI B    1BH      05H      MOV EI 0CH      CALL
14D4H 09H    04H      CALL     EAH      03H      MOV CM INR H    MOV LI 1CH
14DDH MOV MC LXI B    27H      05H      MOV EI 0CH      CALL     09H      04H
14E6H CALL   EAH      03H      MOV CM INR H    MOV LI 1DH      MOV MC LXI B
14EFH 33H    05H      MOV EI 0EH      CALL     09H      04H      CALL     EAH
14F8H 03H    MOV CM INR H    MOV LI 1EH      MOV MC CALL     D6H      03H
1501H CALL   E8H      12H      RET
1505H 0DH 0AH 49H 4CH 4CH 45H 47H 41H 4CH 20H 43H 48H 41H 52H 20H 50H
1515H 4CH 45H 41H 53H 45H 20H 52H 45H 45H 4EH 54H 45H 52H 0DH 0AH
1524H 0DH 0AH 53H 45H 4CH 45H 43H 54H 20H 48H 45H 41H 44H 20H 31H 20H
1534H 4FH 52H 20H 32H 2DH
1539H LXI B  24H      15H      MOV EI 15H      CALL     09H      04H      CALL
1542H 7FH    03H      CALL     D6H      03H      LXI H    CFH      20H      MOV AM
154BH SUB I  30H      MOV MA SUB I    00H      JZ       82H      15H      MOV AI
1554H 02H    SUB M    JC       82H      15H      MOV CM DCR C    JNZ      69H
155DH 15H    MOV LI B2H      MOV MI 00H      MOV LI 4AH      MOV MI 00H
1566H JMP    71H      15H      MOV LI B2H      MOV MI 01H      MOV LI 4AH
156FH MOV MI FFH      JMP      90H      15H
```

PLM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
00000847 SYSIN   1574H 0DH 0AH 42H 4FH 54H 20H 50H 4FH 53H 53H 49H 42H 4CH 45H
00000848 SYSIN   1582H LXI B  74H   15H   MOV EI 02H    CALL   09H    04H    LXI SP
00000849 SYSIN   158BH FFH    23H   JMP   C9H    1AH    RET    LXI B  03H    C0H
00000850 SYSIN   1594H LXI H  52H   20H   DAD B  PUSH H LXI B  03H    00H    LXI H
00000851 SYSIN   159DH 06H    01H   DAD B MOV AM POP H  MOV MA LXI B  04H    C0H
00000852 SYSIN   15A6H LXI H  52H   20H   DAD B  PUSH H LXI B  04H    00H    LXI H
00000853 SYSIN   15AFH 06H    C1H   DAD B MOV AM POP H  MOV MA CALL   ABH    06H
00000854 SYSIN   15B8H RET    JMP   CCH   15H
00000855 SYSIN   15BCH 20H 4DH 49H 4EH 55H 54H 45H 20H 43H 4FH 55H 4EH 54H 0DH 0AH 0AH
00000856 SYSIN   15CCH LXI B  03H   00H   LXI H  E1H    0CH    DAD B  XCHG   MOV CE
00000857 SYSIN   15D5H MOV BD MOV EI 14H  CALL   09H    04H    MOV LI 22H    MOV CM
00000858 SYSIN   15DEH CALL   34H   04H   LXI B  BCH    15H    MOV EI 10H    CALL
00000859 SYSIN   15E7H 09H    04H   CALL  69H    0DH    RET    CALL   9CH    09H
00000860 SYSIN   15F0H JMP    0BH   16H
00000861 SYSIN   15F3H 02H FEH 05H 00H 00H 24H
00000862 SYSIN   15F9H 01H 05H FEH 00H 00H 24H
00000863 SYSIN   15FFH 01H 07H FEH 05H 00H 24H
00000864 SYSIN   1605H 01H FEH 00H 00H 00H 24H
00000865 SYSIN   160BH DCR H  MOV LI CEH  MOV MI 00H    MOV AI 02H    LXI H  CEH
00000866 SYSIN   1614H 20H    SUB M JC    80H    16H    MOV EI 06H    MOV DI 00H
00000867 SYSIN   161DH LXI H  D0H   20H   MOV CM MOV BI 00H    CALL   FCH    08H
00000868 SYSIN   1626H LXI B  F3H   15H   XCHG   DAD B  XCHG   MOV CE MOV BD CALL
00000869 SYSIN   162FH A9H    08H   LXI B A9H    21H    LXI H  22H    20H    MOV MC
00000870 SYSIN   1638H INR L  MOV MB MOV EI 04H  MOV DI 00H    LXI H  CEH    20H
00000871 SYSIN   1641H MOV CM MOV BI 00H  CALL   FCH    08H    LXI H  22H    20H
00000872 SYSIN   164AH MOV CM INR L MOV BM XCHG  DAD B  XCHG   MOV CE MOV BD CALL
00000873 SYSIN   1653H 81H    08H   LXI B A9H    21H    LXI H  20H    20H    MOV MC
00000874 SYSIN   165CH INR L  MOV MB MOV EI 04H  MOV DI 00H    LXI H  CEH    20H
00000875 SYSIN   1665H MOV CM MOV BI 00H  CALL   FCH    08H    LXI H  20H    20H
00000876 SYSIN   166EH MOV CM INR L MOV BM XCHG  DAD B  XCHG   MOV CE MOV BD CALL
00000877 SYSIN   1677H 9FH    08H   LXI H CEH    20H    INR M  JNZ    10H    16H
00000878 SYSIN   1680H CALL   D9H   08H   RET    CALL   0FH    10H    LXI B  99H
00000879 SYSIN   1689H 20H    INR H MOV LI F6H   MOV MC INX H  MOV MB LXI B  A1H
00000880 SYSIN   1692H 20H    LXI D A9H   20H    CALL   8DH    10H    CALL   91H
00000881 SYSIN   169BH 15H    CALL  EDH   15H    CALL   B9H    15H    RET    LXI H
00000882 SYSIN   16A4H FDH    21H   MOV MI 01H   MOV AI 02H    LXI H  FDH    21H
00000883 SYSIN   16ADH SUB M  JC    6CH   17H    MOV CM DCR C  JNZ    C2H    16H
00000884 SYSIN   16B6H DCR H  MOV LI B2H  MOV MI 00H    MOV LI 4AH    MOV MI 00H
00000885 SYSIN   16BFH JMP    CBH   16H   DCR H  MOV LI B2H    MOV MI 01H    MOV LI
00000886 SYSIN   16C8H 4AH    MOV MI FFH  INR H  MOV LI FEH    MOV MI 00H    MOV AI
00000887 SYSIN   16D1H 09H    LXI H FEH   21H    SUB M  JC     67H    17H    XRA A
00000888 SYSIN   16DAH OUT    FBH   EI    MOV LI 1FH    MOV MI FFH    CALL   19H
00000889 SYSIN   16E3H 12H    CALL  91H   15H    CALL   9CH    09H    CALL   69H
00000890 SYSIN   16ECH 0DH    MOV LI 20H  MOV MI FFH    INR L  MOV MI FFH    INR L
00000891 SYSIN   16F5H MOV MI 05H   DCR H MOV LI D6H    MOV MI 2CH    INX H  MOV MI
00000892 SYSIN   16FEH 02H    MOV LI D0H  MOV MI 00H    CALL   84H    16H    MOV LI
00000893 SYSIN   1707H 22H    MOV MI 30H  DCR H  MOV LI D6H    MOV MI 08H    INX H
00000894 SYSIN   1710H MOV MI 08H   MOV LI D0H   MOV MI 01H    CALL   84H    16H
00000895 SYSIN   1719H MOV LI 22H   MOV MI 35H   DCR H  MOV LI D6H    MOV MI 34H
00000896 SYSIN   1722H INX H  MOV MI 09H  MOV LI D0H    MOV MI 02H    CALL   84H
00000897 SYSIN   172BH 16H    DCR H MOV LI D0H   MOV MI 03H    INR H  MOV LI 22H
00000898 SYSIN   1734H MOV MI 39H   MOV LI 20H   MOV MI 00H    DCR L  MOV MI 00H
00000899 SYSIN   173DH DCR H  MOV LI D4H  MOV MI 24H    INX H  MOV MI 0AH    INR L
00000900 SYSIN   1746H MOV MI 9CH   INX H MOV MI 0AH    CALL   84H    16H    MOV LI
00000901 SYSIN   174FH 21H    MOV MI 00H  IN     01H    MOV CA MOV AI 7FH    SUB I
00000902 SYSIN   1758H 1BH    SUB I 01H   SBC A  ANA C  RPC    JC     6CH    17H
00000903 SYSIN   1761H MOV LI FEH   INR M JNZ    D0H    16H    DCR L  INR M  JNZ
00000904 SYSIN   176AH A8H    16H   RET   JMP    99H    17H
```

```
00000905 SYSIN      1770H 53H 4FH 55H 52H 43H 45H 20H 43H 48H 45H 43H 4BH 20H 4DH 4FH 44H
00000906 SYSIN      1780H 45H 20H 20H
00000907 SYSIN      1783H 0DH 0AH 53H 45H 4CH 45H 43H 54H 20H 4DH 4FH 44H 45H 20H 20H 41H
00000908 SYSIN      1793H 2DH 42H 2DH 43H 20H 2DH
00000909 SYSIN      1799H CALL   39H    15H    CALL   75H    06H    CALL   E2H    11H
00000910 SYSIN      17A2H LXI H  B1H    20H    MOV AM SUB I  FFH    JNZ    B1H    17H
00000911 SYSIN      17ABH LXI SP FFH    23H    JMP    C9H    1AH    LXI B  83H    17H
00000912 SYSIN      17B4H MOV EI 16H    CALL   09H    04H    CALL   7FH    03H    CALL
00000913 SYSIN      17BDH D6H    03H    LXI H  CFH    20H    MOV AM SUB I  41H    JC
00000914 SYSIN      17C6H 1BH    19H    MOV AI 43H    SUB M  JC     1BH    19H    MOV AM
00000915 SYSIN      17CFH SUB I  43H    JNZ    DFH    17H    LXI B  07H    00H    MOV LI
00000916 SYSIN      17D8H 4AH    DAD B  MOV MI FFH    JMP    E7H    17H    LXI B  07H
00000917 SYSIN      17E1H 00H    MOV LI 4AH    DAD B  MOV MI 00H    CALL   75H    06H
00000918 SYSIN      17EAH XRA A  OUT    FBH    EI     MOV LI 88H    MOV MI 00H    INX H
00000919 SYSIN      17F3H MOV MI 00H    MOV LI D4H    MOV MI 01H    INX H  MOV MI 00H
00000920 SYSIN      17FCH INR L  MOV MI 79H    INX H  MOV MI 00H    CALL   CFH    10H
00000921 SYSIN      1805H LXI B  99H    20H    INR H  MOV LI F6H    MOV MC INX H  MOV MB
00000922 SYSIN      180EH LXI B  A1H    20H    LXI D  A9H    20H    CALL   8DH    10H
00000923 SYSIN      1817H LXI H  CFH    20H    MOV AM SUB I  41H    MOV LA MOV MI 00H
00000924 SYSIN      1820H JMP    AAH    18H    LXI H  FFH    21H    MOV MI 00H    MOV AI
00000925 SYSIN      1829H 06H    LXI H  FFH    21H    SUB M  JC     BAH    18H    MOV CM
00000926 SYSIN      1832H MOV BI 00H    DCR H  MOV LI A1H    DAD B  MOV MI 00H    LXI H
00000927 SYSIN      183BH FFH    21H    MOV CM MOV BI 00H    DCR H  MOV LI A9H    DAD B
00000928 SYSIN      1844H MOV MI 00H    LXI H  FFH    21H    INR M  JNZ    28H    18H
00000929 SYSIN      184DH JMP    BAH    18H    LXI H  FFH    21H    MOV MI 00H    MOV AI
00000930 SYSIN      1856H 06H    LXI H  FFH    21H    SUB M  JC     BAH    18H    MOV CM
00000931 SYSIN      185FH MOV BI 00H    DCR H  MOV LI 99H    DAD B  MOV MI 00H    LXI H
00000932 SYSIN      1868H FFH    21H    MOV CM MOV BI 00H    DCR H  MOV LI A9H    DAD B
00000933 SYSIN      1871H MOV MI 00H    LXI H  FFH    21H    INR M  JNZ    55H    18H
00000934 SYSIN      187AH JMP    BAH    18H    LXI H  FFH    21H    MOV MI 00H    MOV AI
00000935 SYSIN      1883H 06H    LXI H  FFH    21H    SUB M  JC     BAH    18H    MOV CM
00000936 SYSIN      188CH MOV BI 00H    DCR H  MOV LI 99H    DAD B  MOV MI 00H    LXI H
00000937 SYSIN      1895H FFH    21H    MOV CM MOV BI 00H    DCR H  MOV LI A1H    DAD B
00000938 SYSIN      189EH MOV MI 00H    LXI H  FFH    21H    INR M  JNZ    82H    18H
00000939 SYSIN      18A7H JMP    BAH    18H    DAD H  LXI B  B4H    18H    DAD B  MOV EM
00000940 SYSIN      18B0H INX H  MOV DM XCHG   PCHL
00000941 SYSIN      18B4H 23H 18H 50H 18H 7DH 18H
00000942 SYSIN      18BAH LXI H  FFH    21H    MOV MI 00H    MOV AI 17H    LXI H  FFH
00000943 SYSIN      18C3H 21H    SUB M  JC     D8H    18H    MOV CM MOV BI 00H    DCR H
00000944 SYSIN      18CCH MOV LI 81H    DAD B  MOV MI 00H    LXI H  FFH    21H    INR M
00000945 SYSIN      18D5H JNZ    BFH    18H    LXI B  07H    00H    DCR H  MOV LI 81H
00000946 SYSIN      18DEH DAD B  MOV MI 24H    LXI B  07H    00H    LXI H  89H    20H
00000947 SYSIN      18E7H DAD B  MOV MI 24H    LXI B  07H    00H    LXI H  91H    20H
00000948 SYSIN      18F0H DAD B  MOV MI 24H    LXI B  03H    00H    LXI H  81H    0CH
00000949 SYSIN      18F9H DAD B  XCHG   MOV CE MOV BD MOV EI 14H    CALL   09H    04H
00000950 SYSIN      1902H LXI B  70H    17H    MOV EI 13H    CALL   09H    04H    LXI B
00000951 SYSIN      190BH CFH    20H    MOV EI 01H    CALL   09H    04H    CALL   9CH
00000952 SYSIN      1914H 09H    CALL   69H    0DH    JMP    99H    17H    JMP    24H
00000953 SYSIN      191DH 19H
00000954 SYSIN      191EH 0DH 0AH 45H 4EH 44H 20H
00000955 SYSIN      1924H LXI B  1EH    19H    MOV EI 06H    CALL   09H    04H    LXI B
00000956 SYSIN      192DH 70H    17H    MOV EI 13H    CALL   09H    04H    CALL   D6H
00000957 SYSIN      1936H 03H    RET    LXI H  D0H    20H    MOV MI 01H    MOV AI 02H
00000958 SYSIN      193FH LXI H  D0H    20H    SUB M  JC     FEH    19H    CALL   39H
00000959 SYSIN      1948H 15H    LXI B  45H    0DH    MOV EI 24H    CALL   09H    04H
00000960 SYSIN      1951H LXI B  D9H    20H    CALL   95H    03H    LXI H  CEH    20H
00000961 SYSIN      195AH MOV MI 00H    LXI H  3FH    21H    MOV AM DCR H  MOV LI CEH
00000962 SYSIN      1963H SUB M  JC     83H    19H    MOV CM MOV BI 00H    MOV LI D9H
```

```
00000963 SYSIN
00000964 SYSIN
00000965 SYSIN
00000966 SYSIN
00000967 SYSIN
00000968 SYSIN
00000969 SYSIN
00000970 SYSIN
00000971 SYSIN
00000972 SYSIN
00000973 SYSIN
00000974 SYSIN
00000975 SYSIN
00000976 SYSIN
00000977 SYSIN
00000978 SYSIN
00000979 SYSIN
00000980 SYSIN
00000981 SYSIN
00000982 SYSIN
00000983 SYSIN
00000984 SYSIN
00000985 SYSIN
00000986 SYSIN
00000987 SYSIN
00000988 SYSIN
00000989 SYSIN
00000990 SYSIN
00000991 SYSIN
00000992 SYSIN
00000993 SYSIN
00000994 SYSIN
00000995 SYSIN
00000996 SYSIN
00000997 SYSIN
00000998 SYSIN
00000999 SYSIN
00001000 SYSIN
00001001 SYSIN
00001002 SYSIN
00001003 SYSIN
00001004 SYSIN
00001005 SYSIN
00001006 SYSIN
00001007 SYSIN
00001008 SYSIN
00001009 SYSIN
00001010 SYSIN
00001011 SYSIN
00001012 SYSIN
00001013 SYSIN
00001014 SYSIN
00001015 SYSIN
00001016 SYSIN
00001017 SYSIN
00001018 SYSIN
00001019 SYSIN
00001020 SYSIN
```

PLM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
196CH DAD B   PUSH H LXI H  CEH     20H     MOV CM MOV BI 00H     MOV LI
1975H D9H     DAD B  MOV AM SUB I   30H     POP H  MOV MA LXI H   CEH
197EH 20H     INR M  JNZ    5CH     19H     LXI H  0DH     00H    MOV LI
1987H D9H     DAD B  MOV MI 24H     LXI H   B2H    20H     MOV AM SUB I
1990H 00H     JNZ    BFH    19H     MOV LI  CEH    MOV MI  00H    MOV AI
1999H 0DH     LXI H  CEH    20H     SUB M   JC     E7H     19H    MOV CM
19A2H MOV BI  00H    INR H  MOV LI  23H     DAD B  PUSH H  LXI H  CEH
19ABH 20H     MOV CM MOV BI 00H     MOV LI  D9H    DAD B   MOV AM POP H
19B4H MOV MA  LXI H  CEH    20H     INR M   JNZ    98H     19H    JMP
19BDH E7H     19H    MOV LI CEH     MOV MI  00H    MOV AI  0DH    LXI H
19C6H CEH     20H    SUB M  JC      E7H     19H    MOV CM  MOV BI 00H
19CFH INR H   MOV LI 30H    DAD B   PUSH H  LXI H  CEH     20H    MOV CM
19D8H MOV BI  00H    MOV LI D9H     DAD B   MOV AM POP H   MOV MA LXI H
19E1H CEH     20H    INR M  JNZ     C3H     19H    LXI H   DCH    20H
19EAH INR M   JNZ    3DH    19H     RET
19EFH 52H 45H 4DH 4FH 54H 45H 20H 57H 4FH 52H 4BH 49H 4EH 47H 20H 4CH
19FFH 45H 56H 45H 4CH 20H 4DH 4FH 4EH 49H 54H 4FH 52H 20H 56H 45H 52H
1A0FH 53H 49H 4FH 4EH 20H 20H 20H 31H 31H 20H 20H 30H 34H 2DH 32H 36H
1A1FH 2DH 37H 37H 0DH 0AH
1A24H XRA A   OUT    07H    LXI H   CEH     20H    MOV MI  00H    MOV AI
1A2DH 05H     LXI H  CEH    20H     SUB M   JC     44H     1AH    MOV CM
1A36H MOV BI  00H    MOV LI C8H     DAD B   MOV MI 00H     LXI H  CEH
1A3FH 20H     INR M  JNZ    2CH     1AH     INR H  MOV LI  1FH    MOV MI
1A48H 00H     INR L  MOV MI 00H     INR L   MOV MI 00H     DCR H  MOV LI
1A51H CEH     MOV MI 00H    MOV AI  0CH     LXI H  CEH     20H    SUB M
1A5AH JC      79H    1AH    MOV CM  MOV BI  00H    INR H   MOV LI 23H
1A63H DAD B   MOV MI 00H    LXI H   CEH     20H    MOV CM  MOV BI 00H
1A6CH INR H   MOV LI 30H    DAD B   MOV MI  00H    LXI H   CEH    20H
1A75H INR M   JNZ    54H    1AH     INR H   MOV LI 23H     MOV MI 01H
1A7EH INX H   MOV MI 24H    LXI H   30H     21H    MOV MI  01H    INX H
1A87H MOV MI  24H    LXI H  3FH     21H     MOV MI 01H     DCR H  MOV LI
1A90H B2H     MOV MI 00H    CALL    D6H     03H    LXI B   EFH    19H
1A99H MOV EI  35H    CALL   09H     04H     CALL   9FH     04H    LXI H
1AA2H CEH     20H    MOV MI 00H     MOV AI  07H    LXI H   CEH    20H
1AABH SUB M   JC     C9H    1AH     MOV CM  MOV BI 00H     MOV LI 4AH
1AB4H DAD B   MOV MI 00H    LXI H   CEH     20H    MOV CM  MOV BI C0H
1ABDH MOV LI  52H    DAD B  MOV MI  00H     LXI H  CEH     20H    INR M
1AC6H JNZ     A6H    1AH    CALL    E1H     03H    XRA A   OUT    FBH
1ACFH EI      CALL   7FH    03H     LXI H   CEH    20H     MOV AM SUB I
1AD8H 40H     MOV MA MOV CA MOV AI  13H     SUB C  JC      9FH    1BH
1AE1H LHLD    CEH    20H    MOV MI  00H     JMP    6AH     1BH    JMP
1AEAH 9FH     1BH    CALL   38H     19H     JMP    9CH     1BH    JMP
1AF3H 9FH     1BH    CALL   A3H     16H     JMP    9CH     1BH    CALL
1AFCH 41H     05H    JMP    9CH     1BH     CALL   9CH     09H    JMP
1B05H 9CH     1BH    CALL   9FH     04H     JMP    9CH     1BH    CALL
1B0EH AAH     13H    JMP    9CH     1BH     CALL   39H     15H    JMP
1B17H 9CH     1BH    JMP    9FH     1BH     CALL   34H     07H    JMP
1B20H 9CH     1BH    JMP    9FH     1BH     JMP    9FH     1BH    JMP
1B29H 00H     38H    JMP    9CH     1BH     JMP    9FH     1BH    JMP
1B32H 9FH     1BH    CALL   69H     0DH     JMP    9CH     1BH    LXI H
1B3BH 81H     20H    LXI H  F6H     21H     MOV MC INX H   MOV MB LXI B
1B44H 89H     20H    LXI D  91H     20H     CALL   8DH     10H    LXI B
1B4DH 99H     20H    LXI H  F6H     21H     MOV MC INX H   MOV MB LXI B
1B56H A1H     20H    LXI D  A9H     20H     CALL   8DH     10H    JMP
1B5FH 9CH     1BH    JMP    24H     1AH     CALL   6DH     17H    JMP
1B68H 9CH     1BH    DAD H  LXI B   74H     1BH    DAD B   MOV EM INX H
1B71H MOV DM  XCHG   PCHL
1B74H E9H 1AH ECH 1AH F2H 1AH F5H 1AH FBH 1AH 01H 1BH 07H 1BH 0DH 1BH
```

PIM COMPILER ASSEMBLY LANGUAGE OUTPUT

```
00001021 SYSIN          1B84H 13H 1BH 19H 1BH 1CH 1BH 22H 1BH 25H 1BH 29H 1BH 2EH 1BH 31H 1BH
00001022 SYSIN          1B94H 34H 1BH 3AH 1BH 61H 1BH 64H 1BH
00001023 SYSIN          1B9CH JMP    C9H    1AH    LXI B  05H    15H    MOV EI 1FH    CALL
00001024 SYSIN          1BA5H 09H    04H    JMP    C9H    1AH    EI     HLT
00001025 SYSIN          NO PROGRAM ERRORS
```

```
1 REM        RWLM CALIBRATIØN PRØGRAM
2 REM     ARGØNNE NATIØNAL LABØRATØRY ELECTRØNICS DIVISIØN
20 PRINT "RWLMCAL.BAS   V02.1": GØSUB 4000
30 FØR I=0 TØ 8
35 READ C(I),CO(I)
40 NEXT I
50 DATA .926838,1.918637,0,-3.196779,0,2.506266
51 DATA -.879403,1.701921,-11.12606,-18.620853,2.752840,16.756924
52 DATA .049957,-3.6592397,4.232080,12.896776,-.251541,-10.308247
100 S5=A5+C5:S3=A3+C3:S2=A2+C2
110 T1=A5-V0*C5:T2=(1+V0)*C5:T3=(1+V0)*C3:T4=A-V0*C
120 T5=(1+V0)*C
130 FØR Q=0 TØ 6 STEP 3
140 F1(Q/3)=(C(Q)*T1+C(Q+1)*T2+C(Q+2)*T3)/(E1*V)
150 F2(Q/3)=(CO(Q)*S5+CO(Q+1)*S3+CO(Q+2)*S2)/(E1*V)
160 NEXT Q
170 W1=(13.68*F1(0)+7.68*(F1(1)+F1(2)))/130000
180 W2=(13.68*F2(0)+7.68*(F2(1)+F2(2)))/130000
190 W3=K1/(E1*2*V*150)
200 IF S0=1 THEN GØSUB 2000:GØTØ 250
210 INPUT "EB,EC PLEASE";E2,E3
250 GØSUB 3000
260 P1=.227261*F0(1)/2.22:P2=.025864*F0(2)/2.22:P3=.035185*F0(3)/2.22
270 P4=(13.68*F0(1)+7.68*(F0(2)+F0(3)))/130000
280 F9=W(1)*A+W(2)*B3+W(3)*C
299 PRINT:PRINT:PRINT:PRINT
300 PRINT "               R W L M   C A L I B R A T I Ø N ":PRINT
301 PRINT:PRINT:PRINT A$:PRINT D$,P$
302 PRINT:PRINT "FNA=";F1(0),,"FNB=";F1(1),,"FNC=";F1(2):PRINT
305 PRINT:PRINT "FNA2=";F2(0),,"FNB2=";F2(1),"FNC2="F2(2)
307 PRINT:PRINT"WL1=";W1,"WLKUS=";W3,"EB=";E2,"EC="E3
310 PRINT:PRINT"WL2=";W2
312 PRINT:PRINT"RN-DAUGHTER CØEFFICIENTS FØR IWL-MEMØRY"
320 FØR I=1 TØ 3: FØR J=1 TØ 3
330 PRINTP(I,J),:NEXTJ:PRINT:NEXT I
335 PRINT:PRINT"RN-DAUGHTER (RWLM) IN ATØMS"
336 PRINT F0(1),F0(2),F0(3)
340 PRINT:PRINT"RN-DAUGHTER (RWLM) IN PCI/L"
345 PRINT P1,P2,P3
350 PRINT:PRINT:PRINT"WL FRØM IWLM (DIRECT AND FRØM RN-DAUGHTERS)"
355 PRINT P4,F9
360 PRINT:PRINT:PRINT"WL CØFFICIENTS FØR IWLM-MEMØRY"
370 PRINT W(1),W(2),W(3)
490 PRINT:PRINT:PRINT:PRINT:PRINT
500 PRINT"NØRMAL STØP AT LINE 500":STØP
2000 X1=(.127907*F2(0)+.236138*F2(1))*V
2010 X2=(.385895*F2(0)+.478116*F2(1)+1.256959*F2(2))*V
2020 X3=(.981722*F2(0)+1.050628*F2(1))*V
2030 X4=(.010200*F2(0)+.028418*F2(1)+.311*F2(2))*V
2040 Q9=X1*X2-X3*X4
2050 Y1=(.385895*F2(0)+.478116*F2(1)+1.256959*F2(2))*V*Q1
2060 Y2=(.0102*F2(0)+.028418*F2(1)+.311*F2(2))*V*Q2
2070 Q8=Y1-Y2:E2=Q8/Q9
2080 Z1=(.127907*F2(0)+.236138*F2(1))*V*Q2
2090 Z2=(.981722*F2(0)+1.050628*F2(1))*V*Q1
2100 E3=(Z1-Z2)/Q9
2110 RETURN
3000 A1(1)=.580386*E1*V:A2(1)=.036204*E2*V+.001584*E3*V
3010 A2(2)=.098134*E2*V+.006941*E3*V:A2(3)=.131*E3*V:A3(1)=.001584*E1*V
```

```
3020 A3(2)=.006941*E1*V:A3(3)=.131*E1*V
3030 B4=A1(1)*(A2(2)*A3(3)-A3(2)*A2(3))
3040 C1(1)=(A2(2)*A3(3)-A2(3)*A3(2))/B4
3050 C2(1)=(A2(3)*A3(1)-A2(1)*A3(3))/B4
3060 C2(2)=(A1(1)*A3(3))/B4
3065 C2(3)=-(A1(1)*A2(3))/B4
3070 C3(1)=(A2(1)*A3(2)-A2(2)*A3(1))/B4
3080 C3(2)=-(A1(1)*A3(2))/B4
3090 C3(3)=(A1(1)*A2(2))/B4
3100 P(1,1)=C1(1)*.227261/2.22
3105 P(1,2)=0
3110 P(1,3)=-P(1,1)*VO:FO(1)=C1(1)*T4
3120 FO(2)=C2(1)*T4+C2(2)*B3+C2(3)*T5
3130 P(2,1)=C2(1)*.025864/2.22:P(2,2)=C2(2)*.025864/2.22
3140 P(2,3)=(C2(3)*(1+VO)-C2(1)*VO)*.025864/2.22
3150 FO(3)=C3(1)*T4+C3(2)*B3+C3(3)*T5
3160 P(3,1)=C3(1)*.035185/2.22:P(3,2)=C3(2)*.035185/2.22
3170 P(3,3)=(C3(3)*(1+VO)-C3(1)*VO)*.035185/2.22
3180 W(1)=(13.68*C1(1)+7.68*(C2(1)+C3(1)))/130000
3190 W(2)=7.68*(C2(2)+C3(2))/130000
3200 W(3)=(7.68*(1+VO)*(C2(3)+C3(3)))/130000-VO*W(1)
3210 RETURN
4000 INPUT "CALCULATE ØR INPUT EB,EC (C/I)"; B$
4001 A$="DATE            PLACE"
4010 IF NØT(B$="I" ØR B$="C") THEN 4000
4020 IF B$="I" THEN  SO=0
4021 IF B$="C" THEN SO=1
4030 INPUT "DATE,PLACE" ; D$,P$
4040 INPUT "FLØWRATE (LITERS/MIN)";V
4050 INPUT "EFFICIENCY ØF ALPHA DETECTØR"; E1
4055 INPUT "OVLAP"; VO
4057 INPUT "TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 5 MINUTES";Q1
4060 INPUT "TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 30 MINUTES";Q2
4070 INPUT "TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 30 MINUTES";A3
4080 INPUT "TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 35 MINUTES";A2
4090 INPUT "TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 35 MINUTES";C2
4100 INPUT "TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 5 MINUTES";A5
4110 INPUT "TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 5 MINUTES";C5
4120 INPUT "TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 30 MINUTES"; C3
4130 INPUT "TØTAL ALPHA CØUNTS IN RAA CHANNEL DURING 2 MINUTES";A
4140 INPUT "TØTAL BETA CØUNTS FRØM RAB AND RAC DURING 2 MINUTES"; B3
4150 INPUT "TØTAL ALPHA CØUNTS IN RAC' CHANNEL DURING 2 MINUTES";C
4160 INPUT "TØTAL ALPHA CØUNTS FRØM 39-41 MINUTES";K1
4999 RETURN
5000 END
ØK
```

TABLE OF DEFINITIONS OF SYMBOLS IN CALIBRATION PROGRAM

A. Constants and Data Inputs

V = Flowrate (liters/min)

VO = Overlap of RaC' counts

A = RaA counts (2 min)

A5 = RaA counts (5 min)

A3 = RaA counts (30 min)

A2 = RaA counts (35 min)

B3 = Ra(B+C) counts (2 min)

Q1 = Ra(B+C) counts (5 min)

Q2 = Ra(B+C) counts (30 min)

C = RaC' count (2 min)

C5 = RaC' count (5 min)

C3 = RaC' count (30 min)

C2 = RaC' count (35 min)

E1 = Alpha efficiency (EA)

K1 = 2-min Kusnetz count from (39 to 41 min)

B. Calculated Variables

E2 = RaB beta efficiency

E3 = RaC beta efficiency

F1(0) = FNA atoms/liter of RaA *
F1(1) = FNB atoms/liter of RaB *
F1(2) = FNC atoms/liter of RaC *

F2(0) = FNA2 atoms/liter of RaA **
F2(1) = FNB2 atoms/liter of RaB **
F2(2) = FNC2 atoms/liter of RaC **

W1 = WL_1 (WL calculated from Rn-daughters derived by spectroscopic method)

W2 = WL_2 (WL calculated from Rn-daughters derived by total-alpha method)

W3 = WLKUS (WL calculated by the Kusnetz method)

---

*Calculated by alpha-spectroscopic method.

**Calculated by total-alpha method.

P(1,1), P(1,2) P(1,3) P(2,1), P(2,2) P(2,3) P(3,1), P(3,2) P(3,3)	=	Rn-daughter coefficients for RWLM memory RaA, RaB, RaC coefficients
F0(1), F0(2), F0(3)	=	Rn-daughters (A, B, C) in atoms/liter calculated from previous coefficient
P1, P2, P3	=	Rn-daughters (A, B, C) in pCi/liter calculated from previous coefficient
F9	=	WL calculated directly from derived WL coefficients
P4	=	WL calculated from the Rn-daughters, F0's
W(1), W(2), W(3)	=	WL coefficients for RWLM memory

```
C       RWLMCN.FTN
C
C               PROGRAM TO CAL PHYSICAL CONSTANTS FOR RWLM
C
C
C               R.T.WITEK/M.K.MCGEE
C                                               4-FEB-77
C
        DOUBLE PRECISION AS(3),DELAY,TD,T,TD2(4),ANSWER(4,3,3),B(9)
        DOUBLE PRECISION D0(9),DA,DB,DC,DH,DJ,DG,DF,DE
        DOUBLE PRECISION FLAM(3),A0(3),C(3),AI(3),CS(3),A(3),AL(3),D(9)
        TYPE 50
        ACCEPT  2,T,TD2(1)
        A0(1) = 0.0
        A0(2) = 0.0
        A0(3) = 0.0
        FLAM(1) = 0.2272614
        FLAM(2) = 0.0258637
        FLAM(3) = 0.0351851
        NUM=3
        N=1
        TD2(2)=5
        TD2(3)=30
        TD2(4)=35
        DO 70 II=1,4
        DO 70 JJ=1,3
        DO 90 I=1,3
 90     C(I)=0.0
        C(JJ)=1.0
        TD=TD2(II)
        CALL BTMNEQ(NUM,FLAM,A0,C,T,AS,AI)
        DO 100 I=1,3
 100    CS(I) = 0.0
        DELAY = 13.0/60.0
        CALL BTMNEQ(NUM,FLAM,AS,CS,DELAY,AL,AI)
        CALL BTMNEQ(3,FLAM,AL,CS,TD,A,AI)
        DO 70 I=1,3
        ANSWER(II,JJ,I)=AI(I)
 70     CONTINUE
        DO 6 II = 1,4
        TYPE 200,TD2(II)
        DO 6 JJ = 1,3
6       TYPE 7,(ANSWER(II,JJ,I),I=1,3)
        K=0
        DO 10 JJ=1,3
        K=K+1
        B(K)=ANSWER(2,JJ,1)
 10     CONTINUE
        DO 11 II=2,3
        DO 11 JJ=1,3
        K=K+1
        B(K)=ANSWER(II,JJ,3)
 11     CONTINUE
C       SPECTROSCOPIC DETERM & INVERTED MAT
        TYPE 300,N
        TYPE 8,(B(K),K=1,9)
        D(1)=B(5)*B(9)-B(6)*B(8)
        D(2)=B(3)*B(8)-B(2)*B(9)
        D(3)=B(2)*B(6)-B(3)*B(5)
        D(4)=B(6)*B(7)-B(4)*B(9)
        D(5)=B(1)*B(9)-B(3)*B(7)
        D(6)=B(3)*B(4)-B(1)*B(6)
        D(7)=B(4)*B(8)-B(5)*B(7)
        D(8)=B(2)*B(7)-B(1)*B(8)
        D(9)=B(1)*B(5)-B(2)*B(4)
        DA=B(1)*((B(5)*B(9))-(B(8)*B(6)))
        DB=B(4)*((B(2)*B(9))-(B(8)*B(3)))
        DC=B(7)*((B(2)*B(6))-(B(5)*B(3)))
```

```
      DE=DA-DB+DC
      DO 17 K=1,9
      D(K)=D(K)/DE
17    CONTINUE
      TYPE 400
      TYPE 8,(D(K),K=1,9)
      N=N+1
      K=0
      III=1
      DO 13 II=2,4
      DO 13 JJ=1,3
      K=K+1
      IF (K.EQ.1) GO TO 55
      IF (K.EQ.4) GO TO 55
      IF (K.EQ.7) GO TO 55
      B(K)=ANSWER(II,JJ,3)
13    CONTINUE
C     TOTAL ALPHA DETERM & INVERTED MAT
      TYPE 300,N
      DO 14 K=1,9
      L=K-1
      TYPE 5,L,B(K)
14    CONTINUE
      D0(1)=B(5)*B(9)-B(6)*B(8)
      D0(2)=B(3)*B(8)-B(2)*B(9)
      D0(3)=B(2)*B(6)-B(3)*B(5)
      D0(4)=B(6)*B(7)-B(4)*B(9)
      D0(5)=B(1)*B(9)-B(3)*B(7)
      D0(6)=B(3)*B(4)-B(1)*B(6)
      D0(7)=B(4)*B(8)-B(5)*B(7)
      D0(8)=B(2)*B(7)-B(1)*B(8)
      D0(9)=B(1)*B(5)-B(2)*B(4)
      DF=B(1)*((B(5)*B(9))-(B(8)*B(6)))
      DG=B(4)*((B(2)*B(9))-(B(8)*B(3)))
      DH=B(7)*((B(2)*B(6))-(B(5)*B(3)))
      DJ=DF-DG+DH
      DO 18 K=1,9
      D0(K)=D0(K)/DJ
18    CONTINUE
      TYPE 700
      TYPE 8,(D0(K),K=1,9)
      TYPE 500
      TYPE 7,(ANSWER(1,1,I),I=1,3)
      TYPE 7,(ANSWER(1,2,2),ANSWER(1,2,3),ANSWER(1,3,3))
      TYPE 7,(ANSWER(1,JJ,3),JJ=1,3)
      K=0
      K=K+1
      TYPE 600
      TYPE 4,K,(ANSWER(2,JJ,2),JJ=1,2)
      K=K+1
      TYPE 3,K,(ANSWER(3,JJ,3),JJ=1,3)
      K=K+1
      TYPE 4,K,(ANSWER(3,JJ,2),JJ=1,2)
      K=K+1
      TYPE 3,K,(ANSWER(2,JJ,3),JJ=1,3)
      K=0
      DO 20 II=3,2,-1
      K=K+1
      TYPE 15,K,(ANSWER(II,JJ,3),JJ=1,3)
20    CONTINUE
      K=0
      DO 30 II=2,3
      K=K+1
      TYPE 16,K,(ANSWER(II,JJ,2),JJ=1,2)
30    CONTINUE
      GO TO 60
4     FORMAT(1X,'X',I1,'= ',E16.10,',  ',E16.10)
3     FORMAT(1X,'X',I1,'= ',2(E16.10,',  '),E16.10)
```

```
 15     FORMAT(1X,'Y',I1,'= ',2(E16.10,', '),E16.10)
 16     FORMAT(1X,'Z',I1,'= ',E16.10,', ',E16.10)
 5      FORMAT(1X,'A0(',I1,')=',1X,E16.10)
  50    FORMAT(1X,' INPUT VARIABLES ')
 200    FORMAT(//1X,F4.0,' MINUTE COUNTING TIME VALUES'/)
 300    FORMAT(//1X,'INVERSE MATRIX DATA FOR F',I1,' CONVERSION'/)
 8      FORMAT(1X,(3(E20.10,2X)))
 400    FORMAT(//1X,'SPECTROSCOPIC INVERTED MATRIX'/)
 700    FORMAT(//1X,'TOTAL ALPHA INVERTED MATRIX'/)
 500    FORMAT(//1X,'DAUGHTER COEFFICIENTS***2 MINUTE COUNTING TIME '/)
 600    FORMAT(//1X,'BETA COEFFICIENTS***5&30 MINUTE COUNTING TIME '/)
 2      FORMAT(2F10.2)
 7      FORMAT(1X,3(E16.10,2X))
 55     III=III+1
        B(K)=(ANSWER(II,JJ,3))+(ANSWER(III,1,1))
        GO TO 13
 60     CALL PETER2(ANSWER,D0,D)
        END
C
C
C
C       IWLMAL.FORT     THIS WAS A SEP PROG THAT IS MERGED
C
        SUBROUTINE PETER2(ANSWER,D0,D)
C
        DOUBLE PRECISION ANSWER(4,3,3),D0(9),D(9)
        DOUBLE PRECISION P(3,3),FM(3),WL(3)
 100    TYPE    10
  10    FORMAT(' INPUT PLEASE...')
        ACCEPT 1, SWITCH
        IF (SWITCH .EQ. 2.0) GOTO 900
 1      FORMAT(F3.1)
        ACCEPT 11,V,EA,OVLAP,BC5,BC30,A30,A35,C35
        TYPE 200,V,EA,OVLAP,BC5,BC30,A30,A35,C35
 200    FORMAT(1X,F5.2,1X,F5.3,1X,F4.2,1X,5(F8.1,1X))
 11     FORMAT(8F20.5)
 12     TYPE 13
 13     FORMAT(' COUNTS PLEASE...')
        ACCEPT 14, A5,C5,C30,A,BC,C,CKUS
 14     FORMAT(7F15.5)
        TYPE    210,A5,C5,C30,A,BC,C,CKUS
 210    FORMAT(1X,7(F10.2,1X))
        S5=A5+C5
        S30=A30+C30
        S35=C35+A35
        TA5 = A5 - OVLAP*C5
        TC5 = (1. + OVLAP)*C5
        TC30 = (1. + OVLAP)*C30
        TA = A-OVLAP*C
        TC = (1 + OVLAP) * C
        FNA = (D(1)*TA5)/(EA*V)
        FNA2 = (D0(1)*S5+D0(2)*S30+D0(3)*S35)/(EA*V)
        FNB = (D(4)*TA5+D(5)*TC5+D(6)*TC30)/(EA*V)
        FNB2 = (D0(4)*S5+D0(5)*S30+D0(6)*S35)/(EA*V)
        FNC = (D(7)*TA5+D(8)*TC5+D(9)*TC30)/(EA*V)
        FNC2 = (D0(7)*S5+D0(8)*S30+D0(9)*S35)/(EA*V)
        WL1 = (13.68*FNA+7.68*(FNB+FNC))/130000.
        WL2 = (13.68*FNA2+7.68*(FNB2+FNC2))/130000.
        WLKUS = CKUS/(EA*2.*V*150.)
        IF(SWITCH.EQ.1.0) GO TO 99
        CALL BETEFF(V,FNA2,FNB2,FNC2,BC5,BC30,EB,EC,ANSWER)
        GO TO 88
99      TYPE 3
3       FORMAT(' EB,EC PLEASE.....')
        ACCEPT 2, EB,EC
2       FORMAT(2F10.5)
88      CALL MATINV(EA,EB,EC,V,OVLAP,TA,TC,BC,P,FM,WL,ANSWER)
        PA = .227261 * FM(1)/2.22
```

```
      PB = .025864 * FM(2)/2.22
      PC = .035185 * FM(3)/2.22
      PWL = (13.68*FM(1) +7.68*(FM(2)+FM(3)))/130000.
      FIWL = WL(1)*A + WL(2) * BC + WL(3) * C
      TYPE 15, FNA,FNB,FNC
      TYPE 22, FNA2,FNB2,FNC2
22    FORMAT(' FNA2=',F8.1,' FNB2=',F8.1,' FNC2=',F8.1)
15    FORMAT(' FNA=',F8.1,' FNB=',F8.1,' FNC=',F8.1)
      TYPE 16, WL1,WLKUS,EB,EC
16    FORMAT(' WL1=',F10.6,'WLKUS=',F10.6,' EB=',F8.4,' EC=',F8.4)
      TYPE 21, WL2
21    FORMAT(' WL2=',F5.2)
      TYPE 71
71    FORMAT(' RN-DAUGHTER COEFFICIENTS FOR IWLM-MEMORY')
      TYPE 17,((P(I,K),K=1,3),I=1,3)
17    FORMAT(3(1X,F9.6,2X))
      TYPE 72
72    FORMAT(' RN-DAUGHTERS(IWLM) IN ATOMS AND PCI/LITER')
      TYPE 18, (FM(I),I=1,3),PA,PB,PC
18    FORMAT(1X,3(D15.9,2X),3(F7.3,2X))
      TYPE 73
73    FORMAT(' WL FROM IWLM(DIRECT AND FROM RN-DAUGHTERS)')
      TYPE 19,PWL,FIWL
19    FORMAT(1X,2(F5.2,2X))
      TYPE 74
74    FORMAT(' WL COEFFICENTS FOR IWLM-MEMORY')
      TYPE 20,(WL(I),I=1,3)
20    FORMAT(1X,3(D20.10,2X))
      GO TO 100
900   RETURN
      END
C

C
C
C
C     MATRINV.FORT
C
C
C
      SUBROUTINE MATINV(EA,EB,EC,V,OVLAP,TA,TC,BC,P,FM,WL,A)
      DOUBLE PRECISION A(4,3,3),P(3,3),FM(3),WL(3)
      A11=A(1,1,1)*EA*V
      A12=.0
      A13=.0
      A21=A(1,1,2)*EB*V+A(1,1,3)*EC*V
      A22=A(1,2,2)*EB*V+A(1,2,3)*EC*V
      A23=A(1,3,3)*EC*V
      A31=A(1,1,3)*EA*V
      A32=A(1,2,3)*EA*V
      A33=A(1,3,3)*EA*V
      BD=A11*(A22*A33-A32*A23)
      C11=(A22*A33-A23*A32)/BD
      C21=(A23*A31-A21*A33)/BD
      C22=(A11*A33)/BD
      C23=-(A11*A23)/BD
      C31=(A21*A32-A22*A31)/BD
      C32=-(A11*A32)/BD
      C33=(A11*A22)/BD
      P(1,1)=C11*.227261/2.22
      P(1,2)=.0
      P(1,3)=-P(1,1)*OVLAP
      FM(1)=C11*TA
      FM(2)=C21*TA+C22*BC+C23*TC
      P(2,1)=C21*.025864/2.22
      P(2,2)=C22*.025864/2.22
      P(2,3)=(C23*(1.+OVLAP)-C21*OVLAP)*.025864/2.22
      FM(3)=C31*TA+C32*BC+C33*TC
```

```
      P(3,1)=C31*.035185/2.22
      P(3,2)=C32*.035185/2.22
      P(3,3)=(C33*(1+OVLAP)-C31*OVLAP)*.035185/2.22
      WL(1)=(13.68*C11+7.68*(C21+C31))/130000.
      WL(2)=7.68*(C22+C32)/130000.
      WL(3)=(7.68*(1.+OVLAP)*(C23+C33))/130000.-OVLAP*WL(1)
      RETURN
      END
C
C
C
C
C
C
C     BETEFF SUB
C
C
C
      SUBROUTINE BETEFF(V,FNA,FNB,FNC,BC5,BC30,EB,EC,A)
      DOUBLE PRECISION A(4,3,3)
      X1=(A(2,1,2)*FNA+A(2,2,2)*FNB)*V
      X2=(A(3,1,3)*FNA+A(3,2,3)*FNB+A(3,3,3)*FNC)*V
      X3=(A(3,1,2)*FNA+A(3,2,2)*FNB)*V
      X4=(A(2,1,3)*FNA+A(2,2,3)*FNB+A(2,3,3)*FNC)*V
      DET = X1*X2-X3*X4
      Y1 = (A(3,1,3)*FNA+A(3,2,3)*FNB+A(3,3,3)*FNC)*V*BC5
      Y2=(A(2,1,3)*FNA+A(2,2,3)*FNB+A(2,3,3)*FNC)*V*BC30
      DETB=Y1-Y2
      EB=DETB/DET
      Z1=(A(2,1,2)*FNA+A(2,2,2)*FNB)*V*BC30
      Z2=(A(3,1,2)*FNA+A(3,2,2)*FNB)*V*BC5
      DETC=Z1-Z2
      EC=DETC/DET
      RETURN
      END
C
C
C
C
C
C
C
C     BTMNEQ SUBROUTINE
C
C
C
      SUBROUTINE BTMNEQ (NUM,FLAM,A0,C,T,A,AI)
      DOUBLE PRECISION FLAM(3),A0(3),C(3),A(3),AI(3),T
      DOUBLE PRECISION ANK
      DO 1 I=1,NUM
      AI(I) = 0.0
      A(I) = 0.0
 1    CONTINUE
      DO 40 N1 = 1,NUM
      DO 30 K = N1,NUM
      DO 20 N = N1,K
      ANK = A0(N1)*FLAM(N)/FLAM(N1)-C(N1)
      DO 10 I = N1,K
      IF (I-N) 9,10,9
 9    ANK = ANK*FLAM(I)/(FLAM(I) -FLAM(N))
 10   CONTINUE
      AI(K) = AI(K) + ANK*(1.0 - DEXP(-FLAM(N)*T))/FLAM(N)
 20   A(K) = A(K) + ANK*DEXP(-FLAM(N)*T)
      AI(K) = AI(K) + C(N1) * T
 30   A(K) = A(K) + C(N1)
 40   CONTINUE
      RETURN
      END
```