AD NUMBER

ADB160453

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; Feb 89. Other requests shall be referred to Commander, USATHAMA, AMXTH-TE, Aberdeen Proving Ground, MD 21010.

AUTHORITY

USATHAMA, per DTIC Form 55, dtd 24 Mar 2003

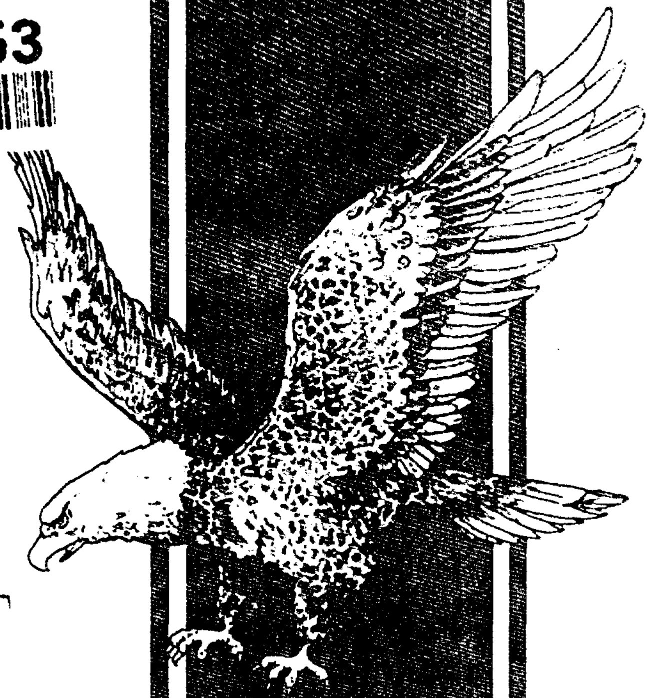
4(2)

Vw

CETH-TE-CR-89045

USATHAMA

U.S. Army Toxic and Hazardous Materials Agency



Fate of Chemical Agents on Structural Surfaces (Task 2)

DTIC
S ELECTE D
DEC 24 1991
D

Battelle Columbus Division
505 King Avenue
Columbus, Ohio 43201-2693

February 1989

Final Report

Distribution ~~limited to~~ auth U.S. government agencies only because of test and evaluation; February, 1989. Other requests for this document must be referred to: Commander, USATHAMA, AMXTH-TE, APG, MD 21010.

Prepared for:

Commander, U.S. Army Toxic and
Hazardous Materials Agency
Aberdeen Proving Ground
Maryland 21010-5401

Final Report

ATE OF CHEMICAL AGENTS ON STRUCTURAL SURFACES
(Task 2)

W. M Neill--Program Manager
Battelle Denver Operations

R. W. Coutant--Principal Investigator,
P. J. Callahan, T. Hayes, F. Girod, F. Moore
Battelle Columbus Division
505 King Avenue
Columbus, Ohio 43201-2693

February 1989
Contract No. DAAA15-86-D-0001

Prepared for

U.S. Army Toxic and
Hazardous Materials Agency
Process Development Branch
Aberdeen Proving Ground, MD 21010-5401

DISCLAIMER

This report is a work prepared for the United States Department of the Army by Battelle Columbus Division. In no event shall either the United States or Battelle have any responsibility or liability for any consequences of any use, misuse, inability to use, or reliance upon the information contained herein, nor does either warrant or otherwise represent in any way the accuracy, adequacy, efficacy, or applicability of the contents hereof.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1 REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT
2 DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Battelle Columbus Division 6b. OFFICE SYMBOL *(If applicable)*	7a. NAME OF MONITORING ORGANIZATION U.S. Army Toxic and Hazardous Materials Agency
6c. ADDRESS *(City, State, and ZIP Code)* 505 King Avenue Columbus, Ohio 43201-2693	7b. ADDRESS *(City, State, and ZIP Code)* ATTN: DRXTH-TE-D Aberdeen Proving Ground, Maryland 21010-5401
NAME OF FUNDING/SPONSORING ORGANIZATION U.S. Army Toxic and Hazardous Materials Agency 8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS *(City, State, and ZIP Code)* Building E-4585 ATTN: DRXTH-TE-D Aberdeen Proving Ground, Maryland 21010	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO.: PROJECT NO.: G8875-0200 TASK NO.: 2 WORK UNIT ACCESSION NO.:

TITLE *(Include Security Classification)*
Fate of Chemical Agents on Structural Surfaces

PERSONAL AUTHOR(S)
W. McNeill, R.W. Coutant, P.J. Callahan, T. Hayes, F. Girod, F. Moore

a. TYPE OF REPORT Final	13b. TIME COVERED FROM 10/86 TO 12/88	14. DATE OF REPORT *(Year, Month, Day)* 1989 February 15	15. PAGE COUNT 83

. SUPPLEMENTARY NOTATION

COSATI CODES			18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
FIELD	GROUP	SUB-GROUP	Hotgas Desorption Process Thermal Desorption of Agents Modeling Core Analysis

. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

The U.S. Army Toxic and Hazardous Materials Agency (USATHAMA) is charged with the responsibility for development and demonstration of technology for identification, containment and elimination of toxic and hazardous materials and related contamination. In support of this mission, Battelle Columbus Division (BCD) has conducted several bench-scale and field demonstration programs to identify and develop promising new concepts for decontamination and reclamation of contaminated structures. The focus of the current program has been on development of a better understanding of the hotgas desorption process (HDP).

The objectives of this task were to further define the fate of chemical agents adsorbed on concrete, and to develop a model describing physical and chemical aspects of the HDP so as to enable easy exploration of the effects of process variables on the effectiveness of the HDP.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE *(Include Area Code)* 22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR
83 APR edition may be used until exhausted.
All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

Block 19

The modelling effort involved development of user-oriented software that is compatible with USATHAMA computers for prediction of heatup rates and agent migration and degradation as a function of process variables with a selection of boundary conditions to represent different structural conditions. The concurrent experimental work involved (a) development of a method for laboratory simulation of HDP and monitoring desorption products using a time-of-flight mass spectrometer with a molecular beam inlet (TOFMS); (b) spiking and analysis of concrete with HD, GB, and VX by gas chromatography; and (c) use of the TOFMS to monitor thermal desorption of agent spiked concrete.

The model developed on this program provides a rapid and convenient means for examining the thermal consequences of variations in HDP parameters.

Analyses of concrete core samples obtained from Newport Army Ammunition Plant (NAAP) showed no evidence for residual VX. Spiking experiments showed that all three agents were degraded by contact with concrete, but at different rates and with different consequences. Mustard, HD, is degraded slowly by both hydrolysis to form oxathiane and by rearrangement to form dithiane. The latter process is accelerated thermally. GB reacts rapidly with the basic components of concrete by replacement of the fluorine group with a hydroxyl group. VX is degraded to a complex spectrum of products. This process is initially slow, but appears to be autocatalytic, i.e. at least some of the products accelerate the degradation process. In all three cases, many of the degradation products are relatively volatile and are readily desorbed from the concrete matrix.

The current work has verified some of the degradation chemistry proposed by other studies of agents, and suggests that the degree of contamination of concrete structures may depend not only on the extent of the original contamination but also on the history of aging of the structure. In cases of relatively casual original contamination, the long term effects of the chemical reactions with the concrete may have essentially decontaminated the structure. This was apparently the case with the core samples obtained from NAAP, where no residual VX was detected. On the other hand, experience with severely contaminated HD sites has shown that appreciable residual HD can remain trapped with the structure after many years. The slowness of the HD degradation coupled with possible saturation of the reaction capacity of the concrete could account for this observation. It is therefore recommended that consideration of application of HDP to structural decontamination be preceded by a formalized core sampling and analysis program to aid in customizing the application of the process to individual cases. It is also recommended that the model be expanded to include chemical factors developed on this program to further facilitate customization of the process.

residual HD can remain trapped with the structure after many years. The slowness of the HD degradation coupled with possible saturation of the reaction capacity of the concrete could account for this observation. It is therefore recommended that consideration of application of HDP to structural decontamination be preceded by a formalized core sampling and analysis program to aid in customizing the application of the process to individual cases. It is also recommended that the model be expanded to include chemical factors developed on this program to further facilitate customization of the process.

CONTENTS

FIGURES

FIGURES (Continued)

TABLES

1.0 Introduction

As part of their responsibilities in DOD real property disposal, the U.S. Army Toxic and Hazardous Materials Agency (USATHAMA) must develop and demonstrate technology to identify, contain, and eliminate toxic and hazardous materials and related contamination. In support of this mission, Battelle's Columbus Division (BCD) has recently completed two projects that identified promising decontamination systems on a bench scale in the laboratory. The objective of the current project is to further define the fate of chemical agents (HD, VX, and GB) adsorbed on structural components such as concrete and paint films. The project goals are to elucidate the chemistry and kinetics of such interactions through (a) development of a kinetic model for desorption and transport of agents from porous substrates and paints; (b) conduct of experiments with agent simulants to verify the kinetic model; and (c) conduct of experiments with agents to demonstrate the utility of the model and to identify specific chemical species desorbed and residual products of the process.

The overall approach included:

1. Development of a user-oriented transient heat and mass flow model for use with 80286/80287 based computers with provision for selection between multiple boundary conditions choices and multiple output modes.

2. Use of simulants (half-mustard and diisopropylmethylphosphonate) doped on concrete to develop methodology for study of thermal desorption of agents from concrete using a time-of-flight mass spectrometer with a molecular beam sampling inlet (TOFMS).

3. Analysis of core samples obtained from NAAP.

4. Spiking and gas chromatographic (GC) analyses of concrete samples doped with HD, GB, and VX.

5. Thermal desorption of fresh and aged concrete samples that had been spiked with HD, GB, and VX.

2.0 PROCEDURES

2.1 MODELING OF HDP

The hot gas desorption process involves exposing one or both sides of concrete wall to a flowing stream of gas at elevated temperatures in order to heat the wall to temperatures sufficient to cause decomposition and/or desorption of adsorbed agent. As depicted in Figure 1, practical application of the process can involve a variety of heat transfer processes, including transport to the surface of the wall (or to both sides of the wall in some instances); unsteady conduction within the wall; and heat losses through the wall to either the outside air or to the soil. This can further be complicated by the presence of an insulating barrier between the wall and the air. Mass flow within the system is similarly complicated by the fact that degradation, evaporation, and condensation can all occur within the heated wall.

The purpose of this portion of the overall task was to develop a user oriented computer code that is sufficiently flexible to deal with the multiple variations of the process and which will allow ready coupling to prediction of residual agent concentrations. The following discussion outlines the general approach taken to the solution of the heat and mass flow problem, and discusses use of the code. A detailed description of the mathematics of the problem is given in Appendix A.

2.1.1 Computational Approach

The most convenient approach to maintain flexibility in choice of process variables and boundary conditions is through the use of a finite element description of the process. Several different procedures are available for this approach, as described by Lienhardt (1981), Crank (1956), Carnahan (1969), and many other standard texts on heat and mass transfer. These procedures all involve solution of a series of simultaneous equations that are generated by expressing the fundamental differential equations in terms of Taylor series expansions of the differentials. This results in solutions at a finite intervals (nodes) within a temperature/depth or concentration/depth array. It is generally agreed that the Crank-Nicolson method (for detailed discussion see Crank, op.cit.) yields the best accuracy per step size used in

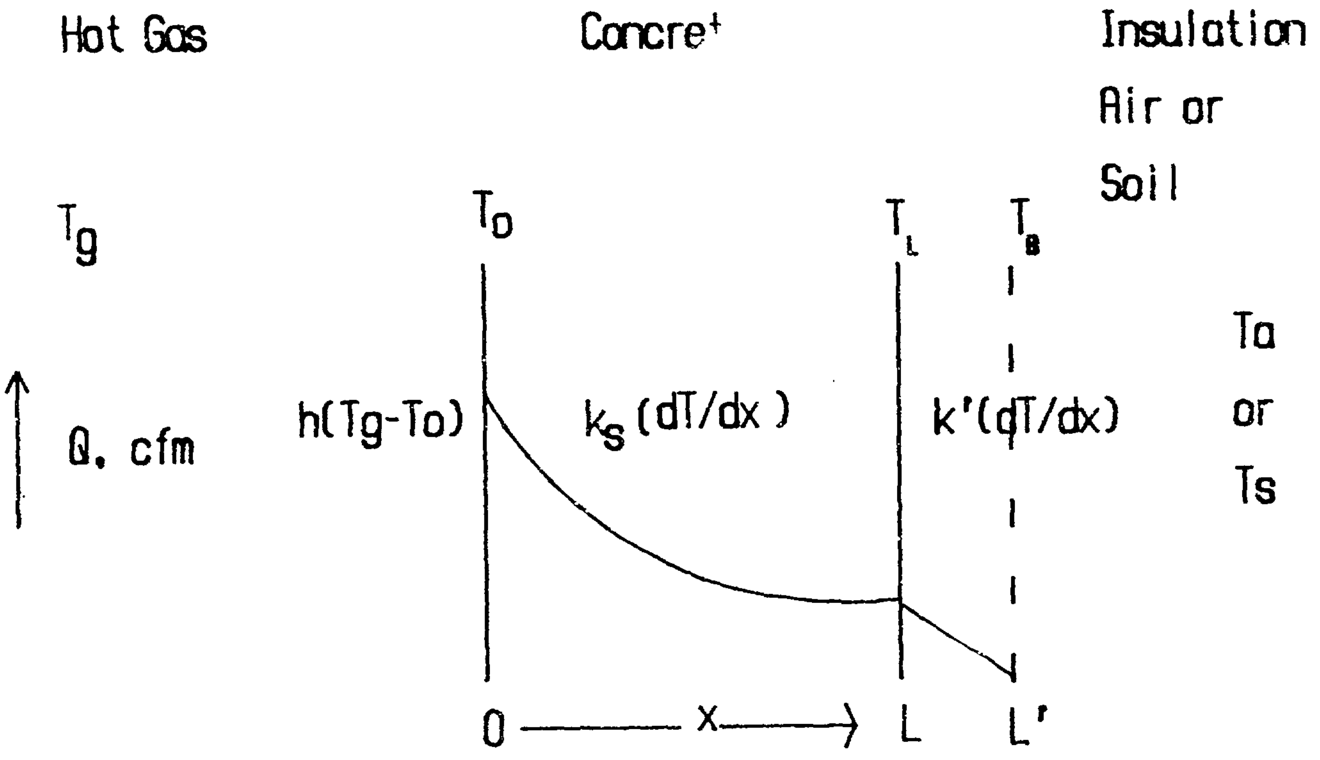


Figure 1. Schematic Representation of Heat Flow in HDP.

the calculation. For example, the accuracy of the Crank-Nicolson method is proportional to the square of the number of spatial nodes used, whereas in the implicit method used in the Battelle Phase II report (1985), the accuracy is directly proportional to the number of spatial n[illegible] Furthermore, fundamental mathematical considerations of the Crank-Nicolson method indicate that it is appropriate to use time steps that are directly proportional to the square of the spatial step, a fact that greatly simplifies the equations used.

2.1.2 Code Description

The code for the HDP model was written in C (Microsoft v5.0) for use with 80286/80287 processors. It is menu-driven with user selected (1) gas properties, (2) structural materials and dimensions, ([illegible] boundary conditions, (4) contaminant type, (5) contaminant distribution and [illegible]vel, [illegible]d (6) output modes. As examples, the user can select any of four bou[illegible]ary [illegible]ditions (both sides of wall heated, one side air cooled, one side insu[illegible] or one side in contact with moist soil); and output can be tabular or [illegible]e form of either time or depth profiles of temperature, adsorbed phas[illegible]centration, or gas phase concentration. The code supports hard copy outpu[illegible] to plotters [illegible]ing HPGL or bit-image graphics to either IBM or Epson compatible printers. [illegible]ile a default set of recommended materials properties is included in the code, virtually every parameter can be changed to simulate special cases.

In its current stage of development, only the heat transfer portion of the HDP model is functional. Provisions have been made for the mass transfer modules and peripheral I/O segments related to the mass transfer have been incorporated, but time and funding limitations related to the need for incorporation of key data from the experimental portion of the program have dictated postponement of completion and testing of the full program. The accuracy of the heat transfer module (temporarily de-coupled from mass transfer) has been checked extensively with each boundary condition and a variety of changes in input parameters. In the case of both sides of the wall being heated, the model results were compared with the explicit solution that is available. In the cases of the other boundary conditions, calculations were compared with other machine solutions that were available for specific sets of conditions. Extensive evaluations also were made by both Battelle and USATHAMA personnel for ease of use and general utility of the code, and recommendations for improvement of useability have been incorporated into the overall code design. The code also has been optimized for speed with the most complex case (determined by the 64k per data segment limitation of the 80286 processor) being completed in about 40 sec (Compaq 286).

Text introducing the use of the code is included on the diskette containing the executable image file of the code.

2.2 CORE ANALYSIS AND SPIKING EXPERIMENTS

2.2.1 Newport Core Analyses

A set of concrete samples was received from the NAAP for analyses of VX and VX degradation products. The core removed from the trench area was suspected of having the greatest degree of contamination. The other cores were also taken from areas where potential exposure could have occurred except for the background core removed from the loading area of the facility.

Each of the cores was divided into approximately 1 inch segments, and the first (top surface) and third segments were pulverized using a diamond mill to a finely divided powder. A small sample of the powder from each segment (10 ± 0.1 g) was weighed into a 4-dram vial for extraction. The samples were extracted using chloroform (10 ± 0.1 ml), and vortex mixing was utilized for 30 ± 5 seconds to facilitate the contact between the concrete particles and the solvent. The solution was allowed to settle, and the liquid layer was transferred through a 0.45-μm syringe filter to a gas chromatograph (GC) vial. The samples were analyzed using capillary GC with a flame photometric detector (GC/FPD). The analog signal from the detector was integrated using Beckman Peak Pro Chromatography software.

2.2.2 Agent Spiking Experiments

The agent spiking experiments were conducted to determine if the agents of interest, sarin (GB), mustard (HD), and VX, were degraded with time when exposed to concrete. These experiments were conducted by adding agent to 1-g quantities of pulverized concrete from the NAAP core samples. The concrete cores were pulverized using a diamond mill which produces a finely divided powder. Two types of experiments were conducted: dilute agent experiments and concentrated agent experiments.

In the dilute agent experiments, initially dilute agent solutions of agent in hexane were prepared. The concentration of these solutions was 2 mg of agent/ml of solution for each agent. One gram of the pulverized concrete core samples was added to 5 ml of each dilute agent solution. The solutions were then mixed by swirling and allowed to stand until the concrete particles had settled to the bottom of the solution. At this time, an aliquot of the liquid layer was removed from each solution and analyzed by GC/FPD to confirm the agent concentration. Subsequent aliquots were removed at multiple time intervals to determine the status of the agent concentration. If the agent concentration was observed below the measurable concentration (0.02 percent), the solution was fortified with 10 mg of neat agent and reanalyzed.

In the concentrated agent experiments, 1 ml of neat chemical surety material (CSM) was added to 1 gram of pulverized concrete core sample. Again, aliquots of the liquid were removed at multiple time intervals and analyzed by GC/FPD to determine the agent concentration. A gas chromatograph-mass spectrometry (GC-MS) analysis also was conducted on concentrated samples after the samples had aged in an attempt to identify the degradation products.

2.3 THERMAL DESORPTION

2.3.1 Apparatus

A schematic representation of the TOFMS used for the thermal desorption studies is shown in Figure 2. This unit consists of a CVC Model MA-3A time-of-flight mass spectrometer that is equipped with a molecular beam sampling inlet and a sample chamber. The sample chamber is attached to the molecular beam sampler via an O-ring flange. Within the sample chamber, a heater and sample cup assembly is mounted on a centrally located moveable pedestal. A thermocouple used for measuring sample temperature feeds through the pedestal to the base of the cup. In use, the height of the sample cup is adjusted to 1-2 mm below the sampling orifice. Air drawn through one of the ports on the sample chamber at approximately 30 cc/min sweeps over the sample and into the orifice. The first stage of the sampler is maintained at a

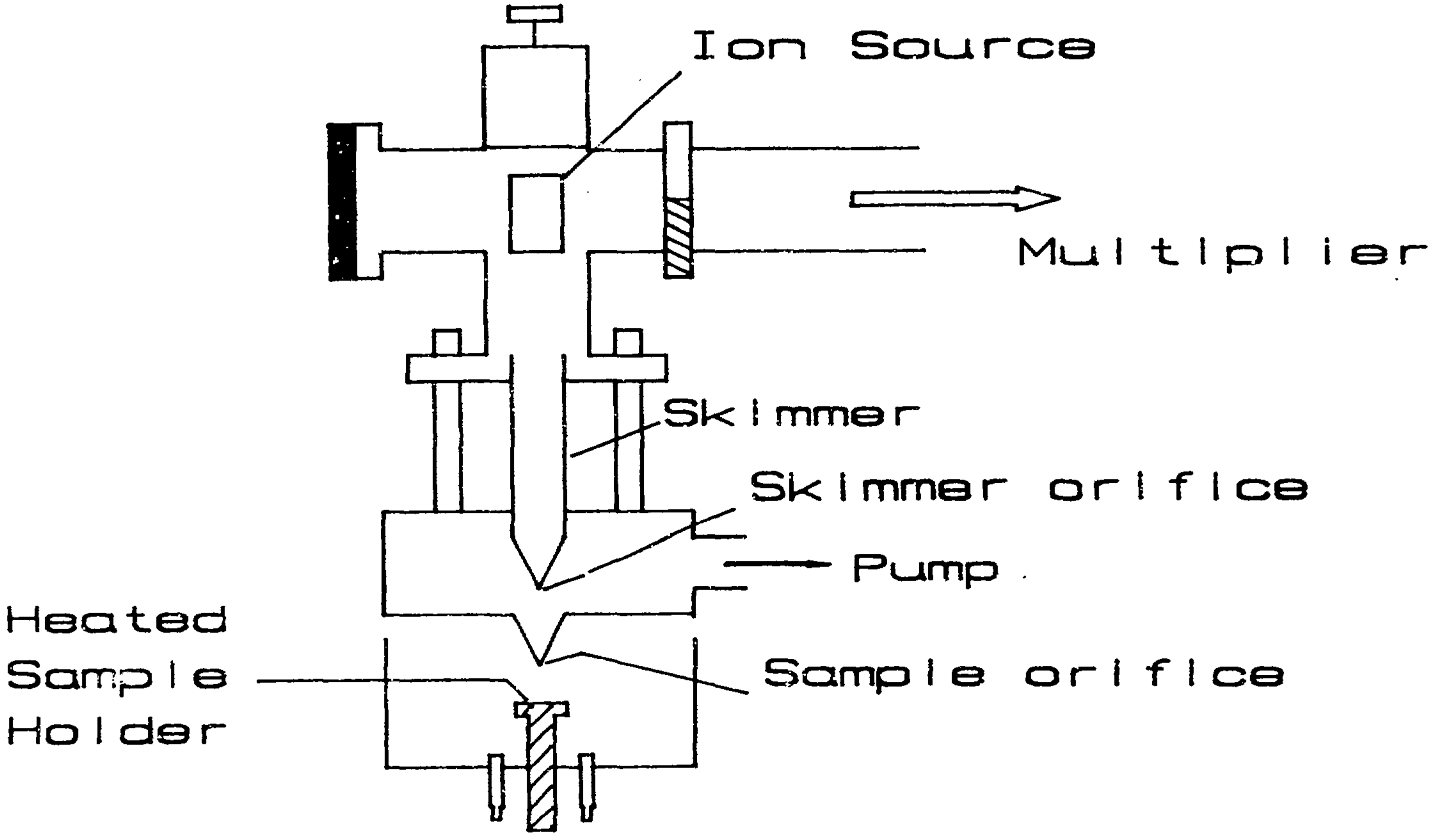


Figure 2. TOFMS used for Thermal Desorption Studies.

pressure of approximately 0.1 torr. Sample gas entering this region undergoes a rapid expansion, with only those molecules traveling along a straight line between the sampling orifice and the skimmer orifice entering the mass spectrometer. The ion source of the mass spectrometer also lies along this straight path so that the sampling process is essentially collisionless and preserves the integrity of the species being sampled. Calibration of the sensitivity of the apparatus is performed by placing samples of neat liquids having known vapor pressures in the sample cup. The principal components of the sample air (nitrogen, oxygen, argon, etc.) serve as mass markers for calibration of the mass range. Because of the dependence of the sampling rate through the orifice on temperature, the sensitivity decreases slightly with temperature, but a typical detection limit for DIMP is 0.042 µg/cc.

2.3.2 Method Development

Prior to performing the agent desorption experiments, preliminary experiments were conducted with the simulants half-mustard, HM, and diisopropylmethylphosphonate, DIMP. These preliminary experiments involved spiking 1 cm X 1 cm X 0.6 cm high porosity (17 vol %) and low porosity (12 vol %) concrete coupons with the simulants. The spiking solutions used consisted of HM and DIMP mixed with hexane at a concentration of 60 mg of simulant/ml of solution.

Each spiked sample was loaded onto the sample cup in the sample chamber. The sample chamber was then attached to the molecular-beam inlet of the TOF. A background spectrum was obtained just before mounting the sample chamber. The temperature of the sample was then programmed from ambient to 200-250°C at rates of 2-10 degrees/min., and the mass range 40 to 200 was repeatedly scanned. In analyzing the mass scans, typically the two masses that were used as calibration masses were 40 (Argon) and one of the major masses of the simulant.

2.3.3 Agent Spiked Samples

All of the agent experiments were conducted using 1 cm X 1 cm X 0.6 cm high porosity (17 vol %) concrete coupons. These coupons came from the same high porosity concrete batch that was used in the simulant experiments. The concrete samples were spiked with one of three agents: GB, HD, or VX. The purity of each agent was checked before spiking. The purities of the GB, HD, and VX were found to be 96 percent, 97 percent, and 84 percent respectively, which are within normal limits. Both fresh and aged samples were prepared for each agent. The fresh samples were prepared by applying 25 µl of the agent to top surface of a concrete coupon immediately before it was analyzed with the mass spectrometer.

Prior to preparing the aged samples, a concrete coupon was soaked in water for 24 hours to determine the approximate amount of water a coupon would absorb. By weighing the coupon before and after soaking, it was determined that the coupon absorbed 7 percent of its weight in water. It was decided that 50 percent of the absorbed water weight would be added to the aged coupons; therefore water equivalent to 3.5 percent of the sample weight was added to each aged coupon. After the addition of water, 100 µl of agent was applied to each coupon. Coupons spiked with the same agent were then placed in a closed container and allowed to age prior to the sample being analyzed with the mass spectrometer.

Before analyzing an agent spiked sample, a calibration run was performed with the agent to determine the sensitivity of the mass spectrometer for the agent and to identify a typical mass scan for the agent. Here a small stainless steel plate was placed in the sample cup. Next a small amount of the agent was placed on the plate and the sample chamber was mounted on the mass spectrometer. Mass scans were taken at room temperature and at elevated temperatures. From the mass scan for the agent itself, the following sensitivity factor was calculated:

$$S = \frac{(\text{AMT Agent})*(\text{IAr})}{(\text{AMT Ar})*(\text{Iagent})} \qquad (1)$$

where,

S = Sensitivity factor
AMT Agent = partial pressure of agent at room temperature
AMT Ar = partial pressure of argon at room temperature
IAr = Intensity of most prominent Ar peak
Iagent = Intensity of most prominent agent peak

Following the calibration run, samples spiked with the calibration agent were analyzed. Before and after the desorption run on a spiked sample, a background scan was obtained with the sample chamber removed. In a typical desorption run, the sample was loaded onto the sample cup, and the sample chamber was mounted on the molecular-beam inlet of the TOF. The temperature of the sample was then increased at 2-10 degrees/min. from near ambient to 200-277°C or until no further desorption occurred. On some occasions during heating, samples were held at one temperature for several minutes to study certain desorption products before heating was begun again. While the GB and HD spiked samples were heated, the mass range of 40 to 200 was repeatedly scanned. For the VX spiked samples, the mass range of 40 to 280 was repeatedly scanned. In analyzing the mass scans of the agent spiked samples, the masses used for calibration masses were 40 (Argon) and 149 (a moderate intensity background peak).

After performing the desorption runs an attempt was made to estimate the total amount of agent recovered from the sample. This estimate was done using the equation below:

$$N = (\text{AMT Agent})*(\Delta t) = \frac{S*(\text{Iagent}*\Delta t)*(\text{AMT Ar})*M}{(\text{IAr})} \tag{2}$$

N = Amount of agent recovered, grams

$(\text{Iagent}*\Delta t)$ = Area under the peak intensity vs time plot for agent

M = Molecular weight of agent

3.0 RESULTS

3.1 HDP MODEL

Figure 3 shows the main menu of the HDP code; several sub-menus are used depending on selections made from the main menu. Figure 4 shows two examples of the hard-copy output capabilities of the HDP code.

3.2 CORE ANALYSIS AND SPIKING EXPERIMENTS

3.2.1 Newport Core Analyses

Essentially, all core sample analyses were identical with no evidence of VX or its degradation products being observed above the 1 µg/g detection limit of the GC/FPD detector. The continuation of these analyses was stopped since the most likely candidates for contamination were found free of any measurable concentrations of VX or chromatographable degradation products.

3.2.2 Agent Spiking Experiments

A summary of the GC/FPD analyses for the dilute agent experiments is presented in Table 1. Figure 5 shows a graph of the results from the dilute experiment conducted on HD. The half-life of the HD has been indicated on Figure 5. As can be seen on Table 1, HD was found to be much more stable than VX or GB in the dilute agent experiments. Both VX and GB showed evidence of rapid decomposition when mixed with the pulverized concrete.

Because the HD appeared to be relatively stable in the dilute experiments, only VX and GB were subjected to the concentrated agent experiments in which 1 ml of neat agent was added to 1 gram of pulverized concrete. The results of the GC/FPD analyses from the concentrated experiments are shown in Table 2. These results show that even undiluted VX and GB neat agent decomposed rapidly when applied to the pulverized concrete. In comparison, the undiluted GB sample decomposed more rapidly than the undiluted VX sample because virtually no GB was remaining after 168 hours but all of the VX was still remaining after 264 hours. Figure 6 shows a graph of the results from the concentrated VX experiment. Along the 100 percent plateau shown in

Figure 3. Main Menu of HDP Code

```
                          Hot Gas Properties

Gas (F)low Rate = 5000 cfm              (G)as Temperature = 1000.0 F

                           Wall Properties

(C)onductivity = 0.70 Btu/ft-hr-F       (S)pecific Heat = 0.21 Btu/lb-F
(D)ensity = 144 lb/ft3                  (T)hickness = 8.0 in
(H)eight, Width = 12.0 x 12.0 ft        (I)nitial Temperature = 70.0 F

                         Boundary Conditions

(1).  Both sides heated                 (2).  Air cooled side
(3).  Moist soil side                   (4).  Insulated side

                           Output Choices

(5).  IBM Printer       (6).  Plotter        (7).  Disk

                           Chemical Agent

(8).  HD                (9).  GB             (0).  VX


Set up problem by pressing letter or number key for each variable you
wish to change or set.
```

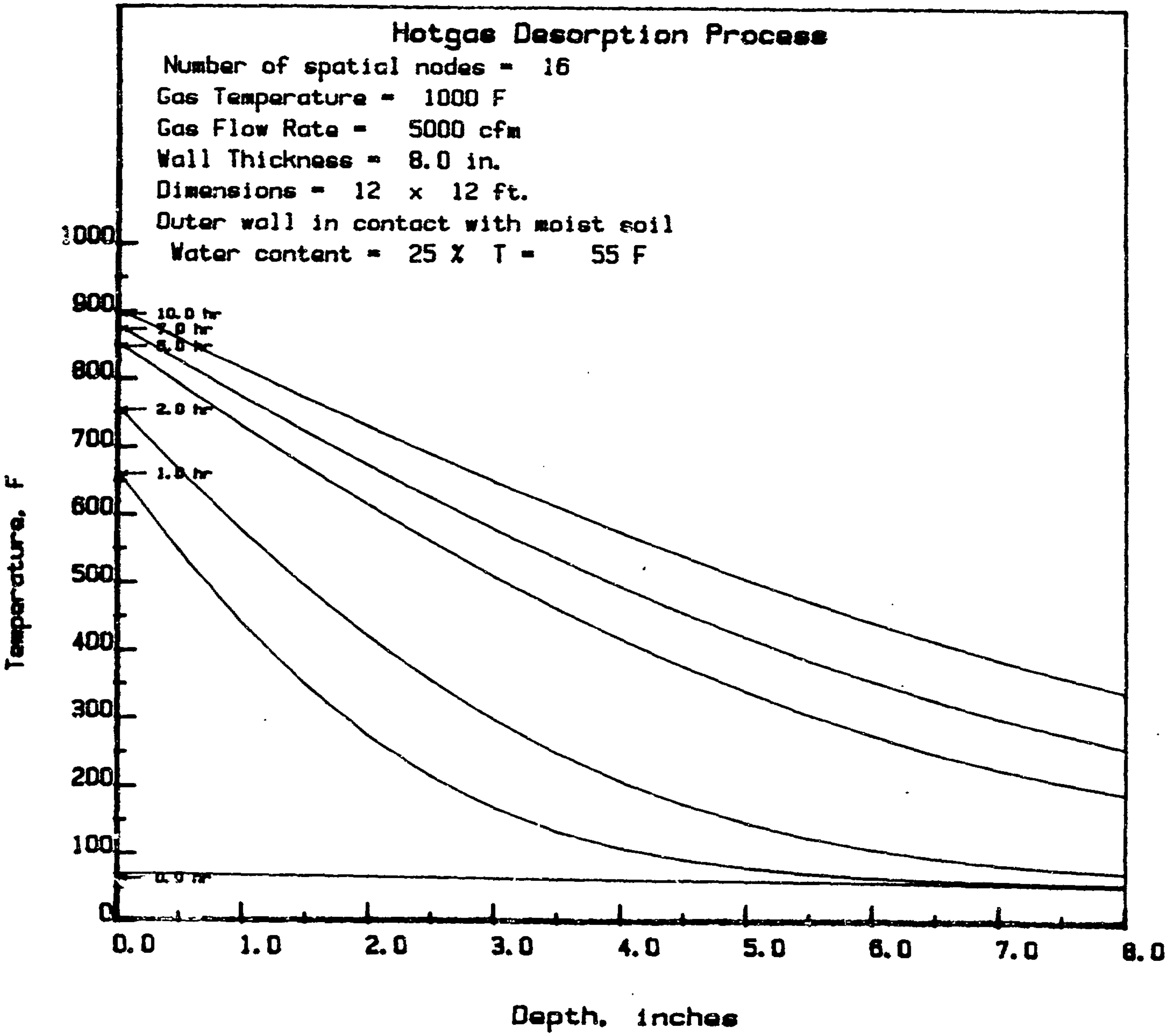


Figure 4A. Hard-Copy Output of Temperature as a Function of Depth at Several Different Times in a Concrete Wall.

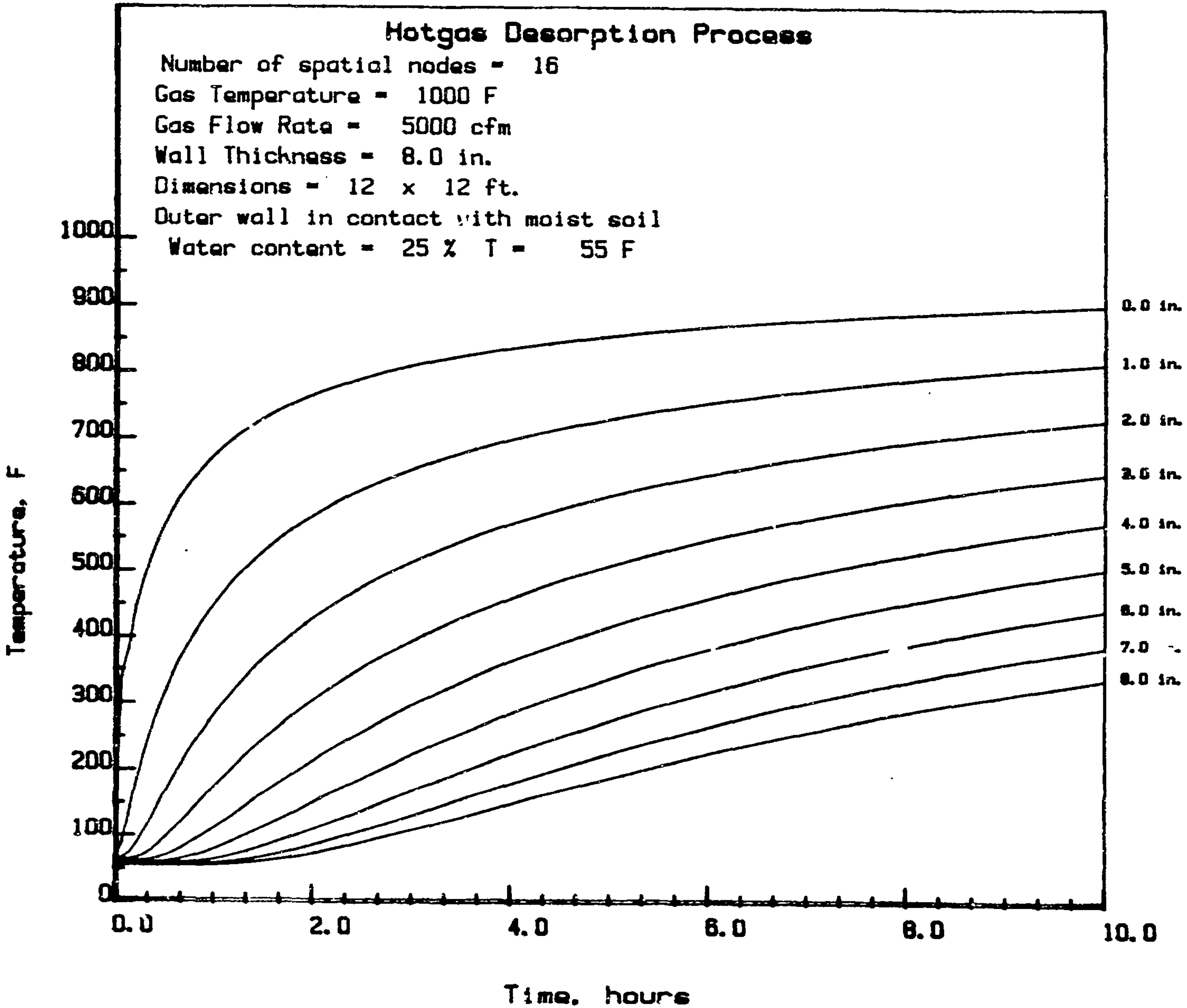


Figure 4B. Hard-Copy Output of Temperature as a Function of Time at Several Different Depths in a Concrete Wall.

TABLE 2. SUMMARY OF ANALYSIS RESULTS FOR THE CONCENTRATED NEAT AGENT CONCRETE STUDIES

(1 ml CSM on 1 gram Concrete)

Date of Sampling	Agent	Hours After Initial Dose	Percent Remaining
09/01/88	VX	0	100
09/14/88	VX	264	100
10/07/88	VX	816	0.3
11/02/88	VX	1368	0.2
11/08/88	VX	1536	0.3
11/16/88	GB	0	100
11/23/88	GB	168	0.04
12/19/88	GB	792	0.0*

*Below detectable limit (0.02 percent).

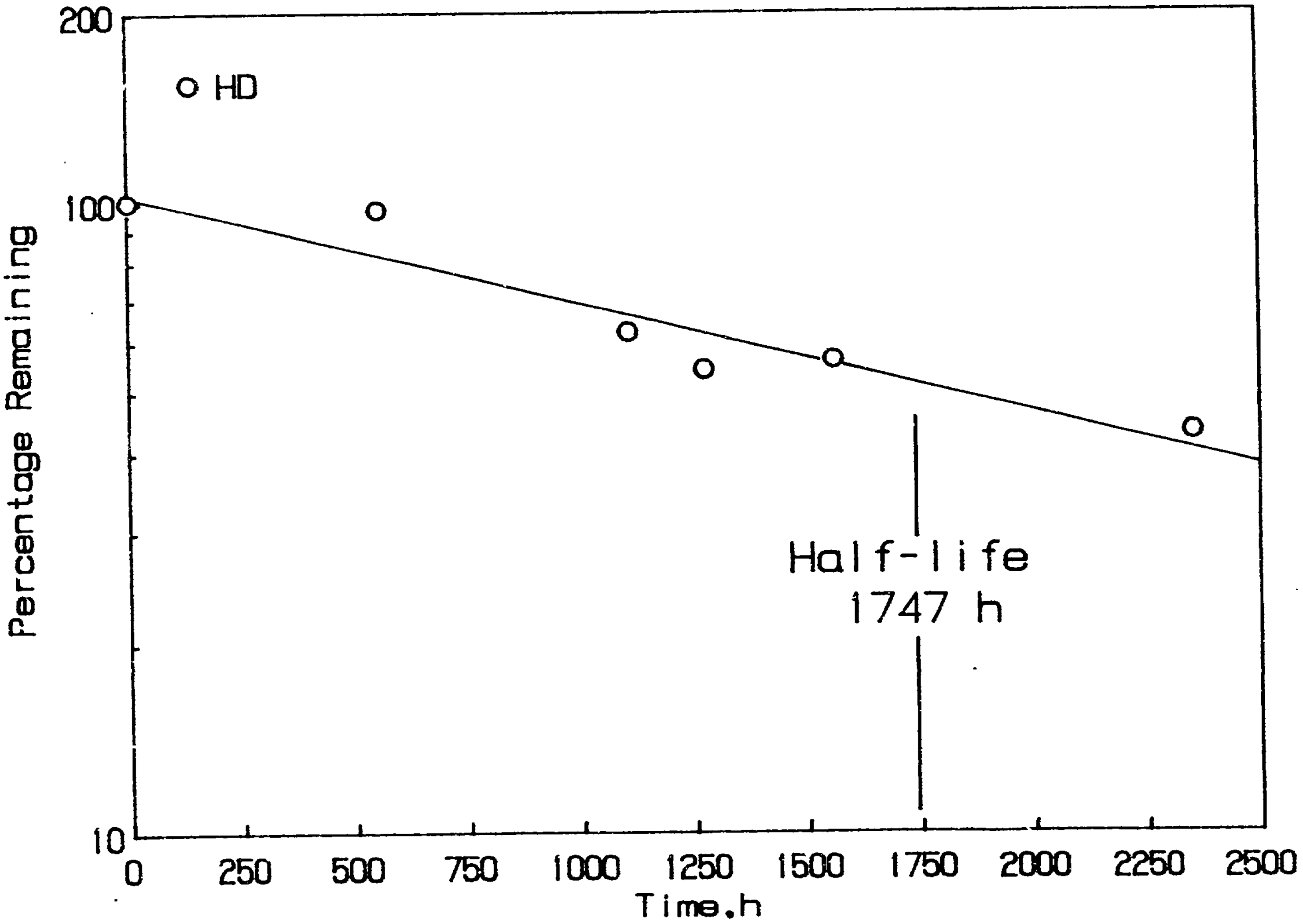


Figure 5. <u>Percentage Remaining vs Time Plot for Dilute Experiment on HD</u>.

Figure 6, the normally clear VX liquid changed to a yellow colored liquid. Along the 0 percent plateau on Figure 6, the liquid remaining was found to be viscous. A GC-MS analyses was conducted on the concentrated VX sample after the sample had aged 1824 hours in an attempt to identify any degradation products that were present. The results of the analysis are shown in Appendix B.

3.3 THERMAL DESORPTION EXPERIMENTS

3.3.1 Simulant Desorption Runs

Table 3 contains a summary of the simulant desorption runs. As noted in Table 3, the HM samples that were treated with loadings of 600 ug/cm^2 HM (in hexane) and allowed to dry yielded no detectable desorption products. The only HM spiked concrete samples that yielded desorption products were those that were either heavily spiked and were analyzed immediately without allowing hexane to evaporate completely, or those that were actually soaked in the HM solution overnight. Even in the last sample in Table 3, where the concrete was exposed to HM vapor in a closed jar, there was no evidence for retention of either HM or its reaction products. Rather, the residual HM was found only in the air contained in the jar.

With DIMP, the results were similar to the HM results in that the DIMP was observed to desorb only with freshly prepared samples. No desorption products besides the DIMP itself were identified on the DIMP spiked samples. The freshly prepared sample yielded a desorption curve as a function of temperature as shown in Figure 7. The slope of this curve represents Ep/R, the apparent energy of the process divided by the universal gas constant. By multiplying the slope by R (.001987 kcal/mole-K) it was found that the apparent energy of the process Ep was 16.5 ± 1.1 kcal/mole. Measurements of the vapor pressure of DIMP yields the following Antoine equation for the vapor pressure of DIMP (courtesy of E. C. Penski, CRDEC):

$$(43.5\text{-}102\ C) \qquad \text{Log}_{10}\ P = 5.7202 - \frac{878.624}{(110.578 + t,\ C)} \qquad (3)$$

Several vapor pressures for DIMP were calculated using Equation 3. These pressures were then plotted with respect to temperature in Figure 8. The slope of the curve in Figure 8 represents $\Delta H_v/R$ (ΔH_v = heat of vaporization of

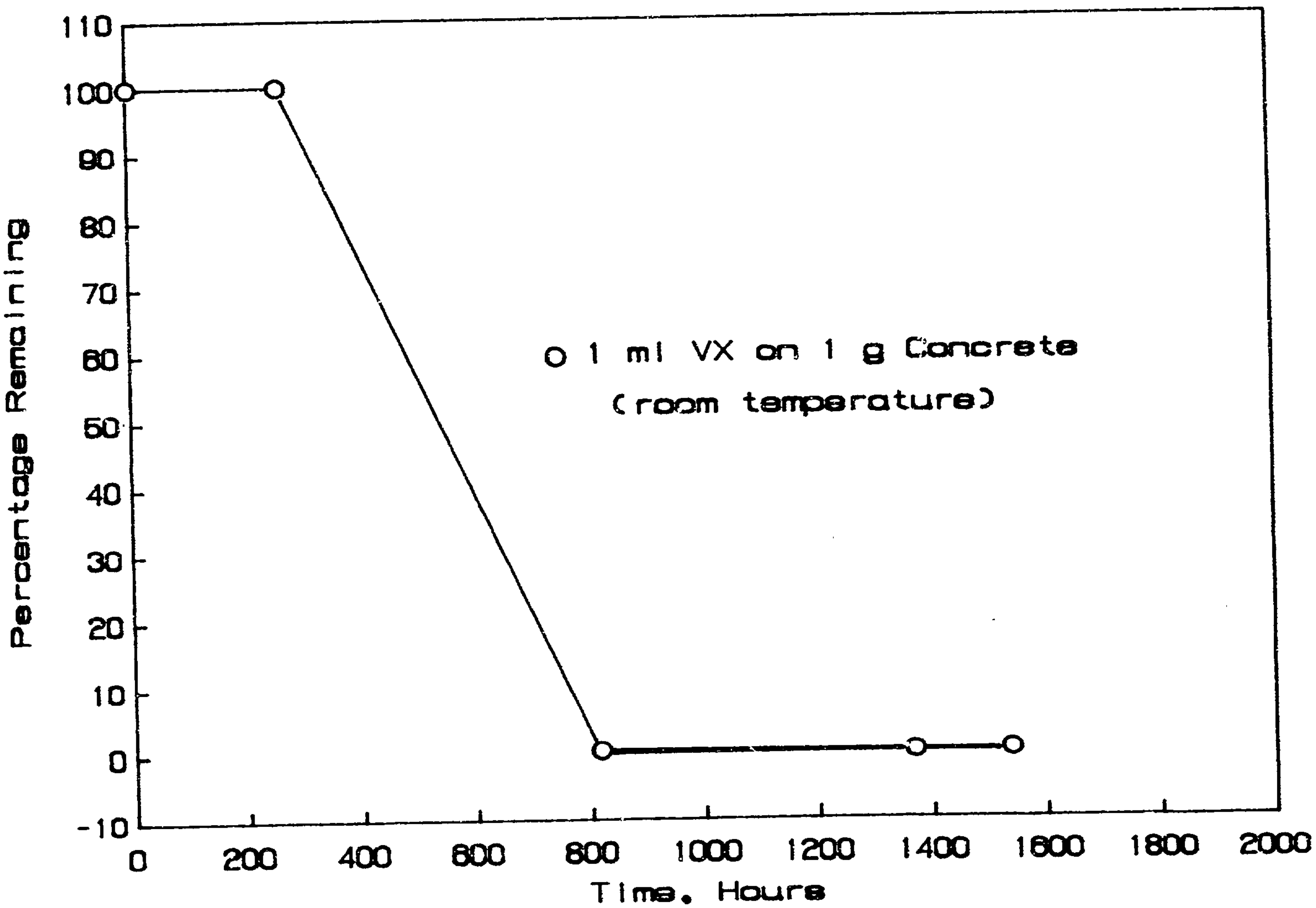


Figure 6. Percentage Remaining vs Time Plot for Concentrated VX on Concrete.

TABLE 3. SUMMARY OF SIMULANT DESORPTION RUNS

Sample Type(a)	Spike Level, mg	Product Mass to Charge Ratio	Product Identification	Temperature where Product was Detected, C
HM/LPC(b)	0.588	None detected	--	--
HM/LPC(c)	2.350	None detected	--	--
DIMP/LPC(d)	2.364	123	DIMP	26
HM/LPC(e)	Soaked	60	Product could not be identified with present information	30
		75	HM	26
		89	VES	50
		124	HM	26
HM/LPC(f)	3.525	56	Hexane	26
		60	Unknown	~45(j)
		75	HM	26
		86	Hexane	26
		89	VES	~62(j)
		124	HM	26
HM/HPC(g)	0.588	None detected	--	--
DIMP/HPC(h)	0.591	None detected	--	--
HM/Hexane(i) Vapor	5.875	51	Unknown	26
		56	Hexane	26
		75	HM	26
		86	Hexane	26
HM/Hexane Vapor plus LPC (i)	Vapor	None detected	--	--

TABLE 3. SUMMARY OF SIMULANT DESORPTION RUNS (Continued)

(a) HM = Half-mustard; DIMP = Diisopropylmethylphosphonate; LPC = Low Porosity Concrete; HPC = High Porosity Concrete ;VES = vinylethylsulfide

(b) 10 ul of HM plus hexane solution applied to top surface immediately prior to analysis

(c) 40 ul of HM plus hexane solution applied to top surface of coupon immediately prior to analysis

(d) 40 ul of DIMP plus hexane solution applied to top surface of coupon immediately prior to analysis

(e) Coupon soaked for 16 hours in a HM plus hexane solution immediately prior to analysis

(f) 60 ul of HM plus hexane solution applied to top surface of coupon immediately prior to analysis

(g) 10 ul of HM plus hexane solution was applied to the painted surface of coupon 25 days prior to analysis

(h) 10 ul of DIMP plus hexane solution was applied to the painted surface of coupon 26 days prior to analysis

(i) 100 ul of HM plus hexane solution were placed in a vial for 24 hours along with a coupon; coupon and vapor were analyzed separately

(j) Interfering peak prevented identification of temperature at which mass first appeared

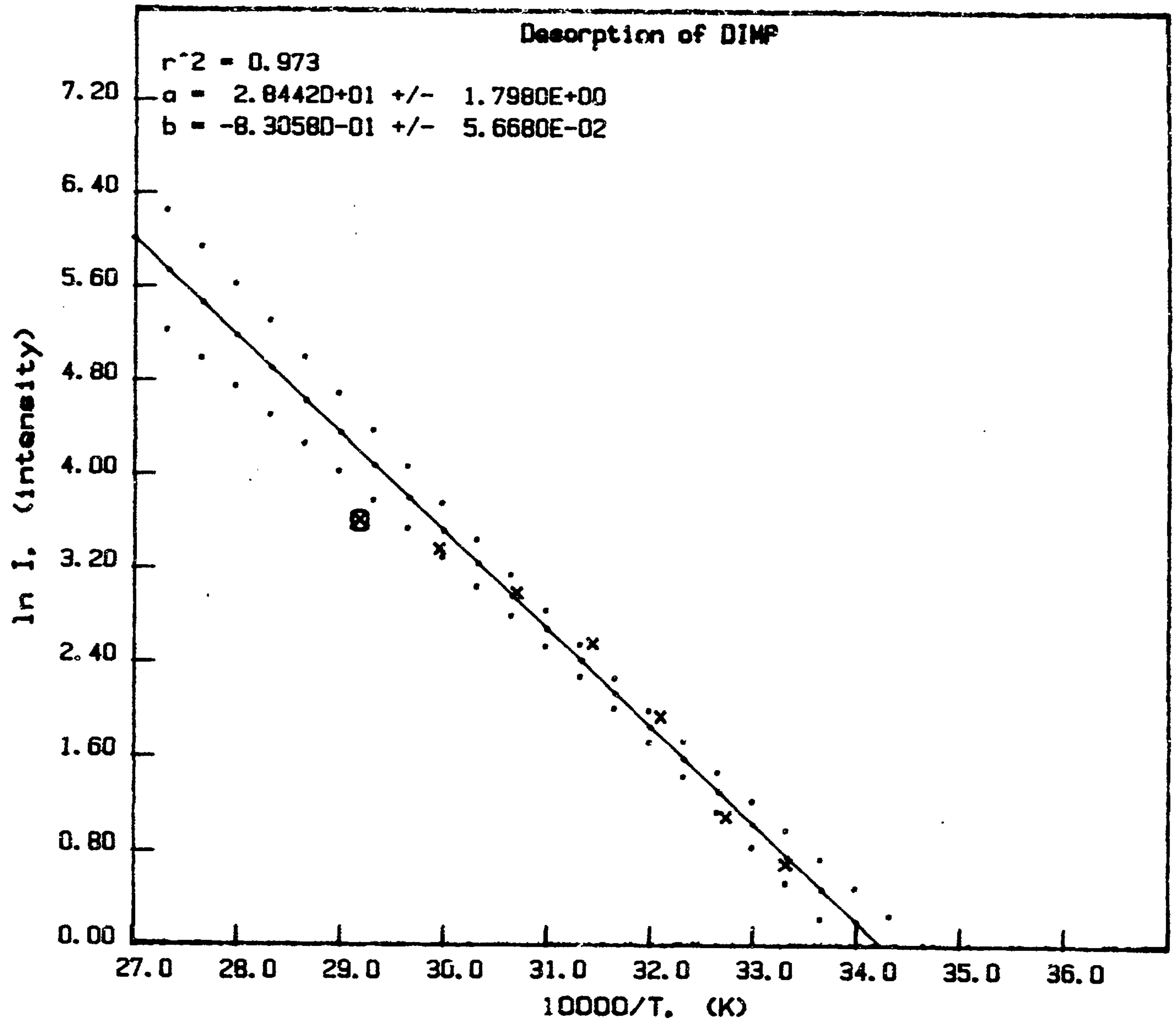


Figure 7. DIMP Desorption Curve.

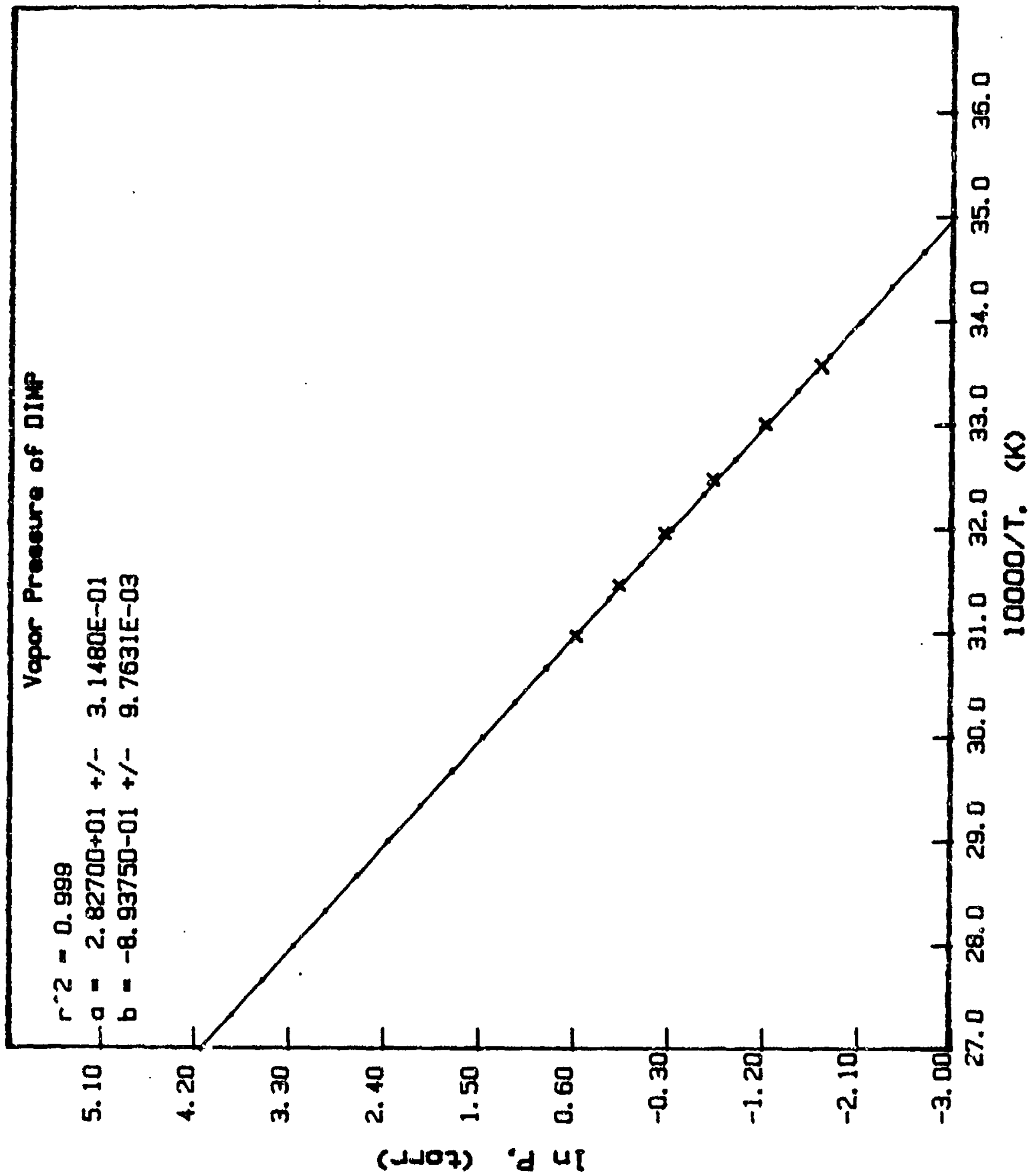


Figure 8. DIMP Vapor Pressure Curve.

DIMP). By multiplying the slope by R, a ΔH_v of 17.8 kcal/mole was found for DIMP. The similarity of the values for Ep and ΔH_v indicate that the desorption of DIMP from the concrete basically involved the evaporation of the DIMP.

With the use of Equation 2, an estimate was made on the amount of DIMP that was recovered from the freshly prepared sample. In particular, it was found that only approximately 6 percent of the DIMP was recovered. This low recovery amount indicates that the DIMP was evaporating very quickly and that most of the DIMP had evaporated during the short time span between when the sample was spiked and when the sample chamber was loaded onto the molecular-beam sampling head.

Overall, the results indicated that the samples spiked at the surface with HM and DIMP did not retain the simulants for any appreciable period of time, and that the loss of the spike can occur by reaction and or evaporation at room temperature. These results indicated that excessive loading or exposures over extended periods of time would be required in the agent desorption experiments, especially for HD, in order to achieve significant penetration into the concrete.

3.3.2 Results with Agents

Table 4 shows a summary of the agent desorption runs. Figures 9-15 show peak intensity and temperature plotted against time for all of the desorption runs.

3.3.2.1 GB Runs

For GB, the only desorption product found on the sample freshly spiked with 25 ul of agent was GB itself. For the aged samples in which water had been added, the major desorption product was found to be isopropyl methyl phosphonate (IMP) with a small amount GB still being present. This result indicates that both time and moisture are required for the GB to react with the concrete. In the two aged samples, the IMP was first detected in the temperature range 26-45°C which is within the range of ambient temperatures to which a concrete structure could be exposed. For the GB sample aged for 120 hours, GB was first detected at 98°C, while for the GB sample aged for 336 hours, GB was first detected at 181°C. The increased temperature required to detect GB in the older sample suggests that the depth of the reaction layer has

TABLE 4. SUMMARY OF AGENT DESORPTION RUNS

Sample Type	Spike Level, mg	Sample Age, hours	Product Mass to Charge Ratio	Product Identification (a)	Temp. where Product was Detected, C
GB (b)	27.3	0	99	GB	26
GB (c)	109.0	120	97	IMP	45
			99	GB	98
GB (c)	109.0	336	97	IMP	25
			99	GB	181
HD (b)	31.8	0	109	HD	47
HD (c)	127.0	312	104	Oxth	172
			109	HD	37
			120	Dith	172
HD	127.0	360	104	Oxth	167
			109	HD	29
			120	Dith	208
VX (c)	100.8	48	114	VX	87

(a) IMP = Isopropyl methyl phosphonate; Oxth = 1,4 Oxathiane; Dith = 1,4 Dithiane

(b) 25 ul of agent was applied to top surface of a fresh concrete coupon immediately prior to analysis.

(c) A quantity of water equivalent in weight to 3.5 percent of the sample weight was applied to the concrete coupon. This quantity of water was equivalent to 50 percent of the water weight that a concrete sample would be expected to absorb when saturated with water. Next 100 μl of agent was applied to the coupon. The coupon was then sealed in a container and allowed to age for a period of time prior to the sample being analyzed.

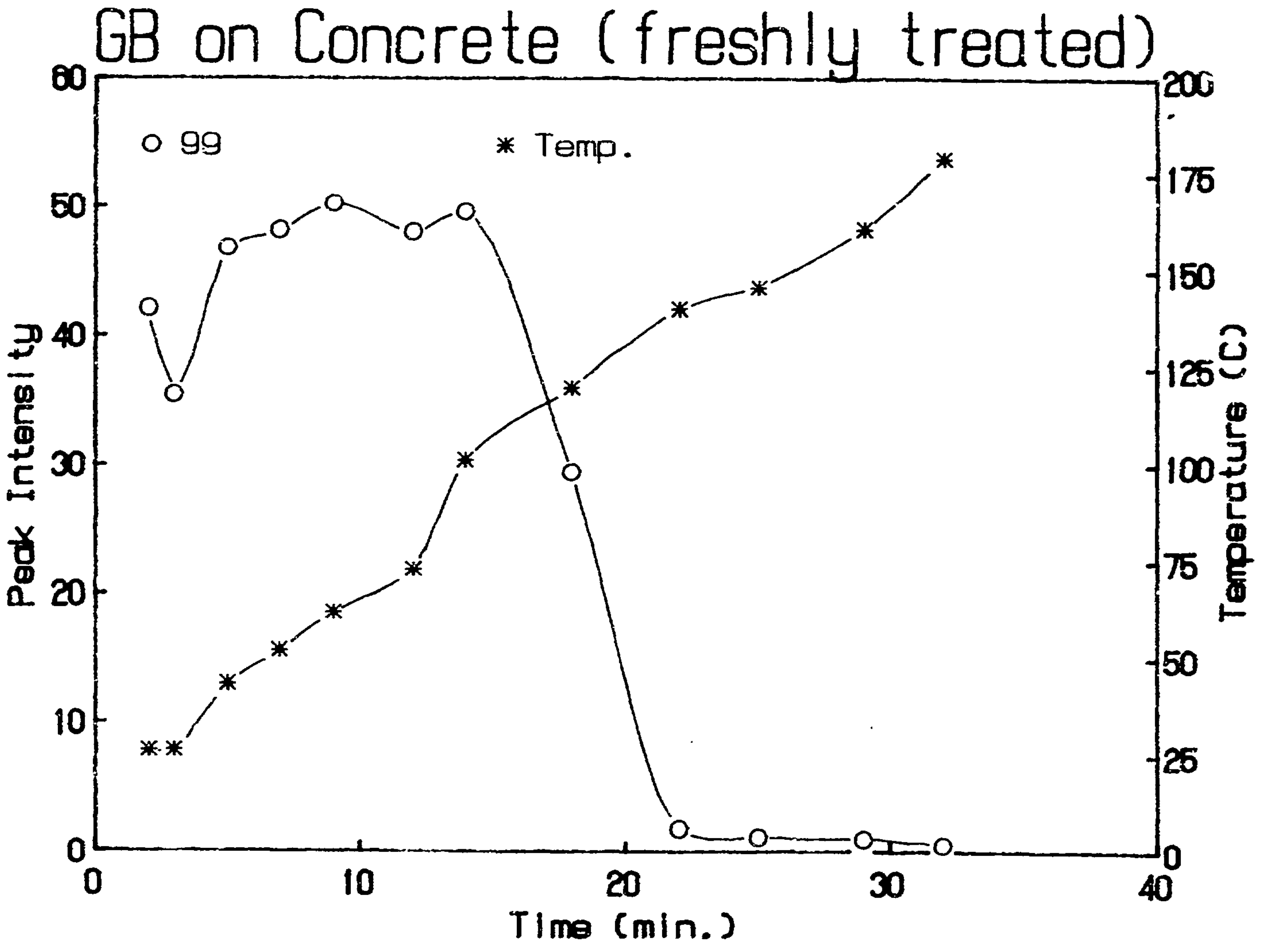


Figure 9. Plot of Peak Intensities (m/e = 99) for Fresh GB Sample.

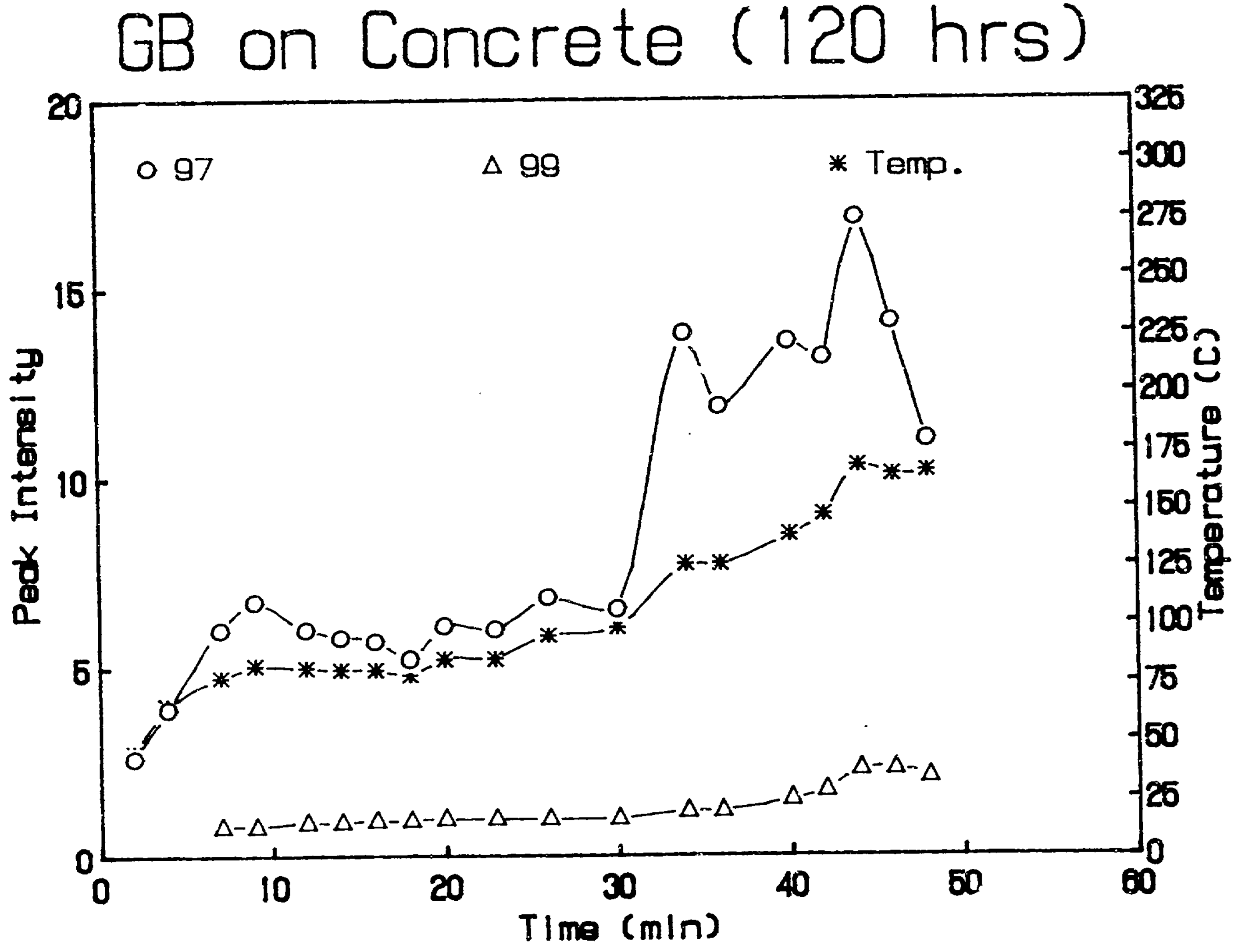


Figure 10. Plot of Peak Intensities (m/e = 97 and 99) for 120 Hour GB Sample.

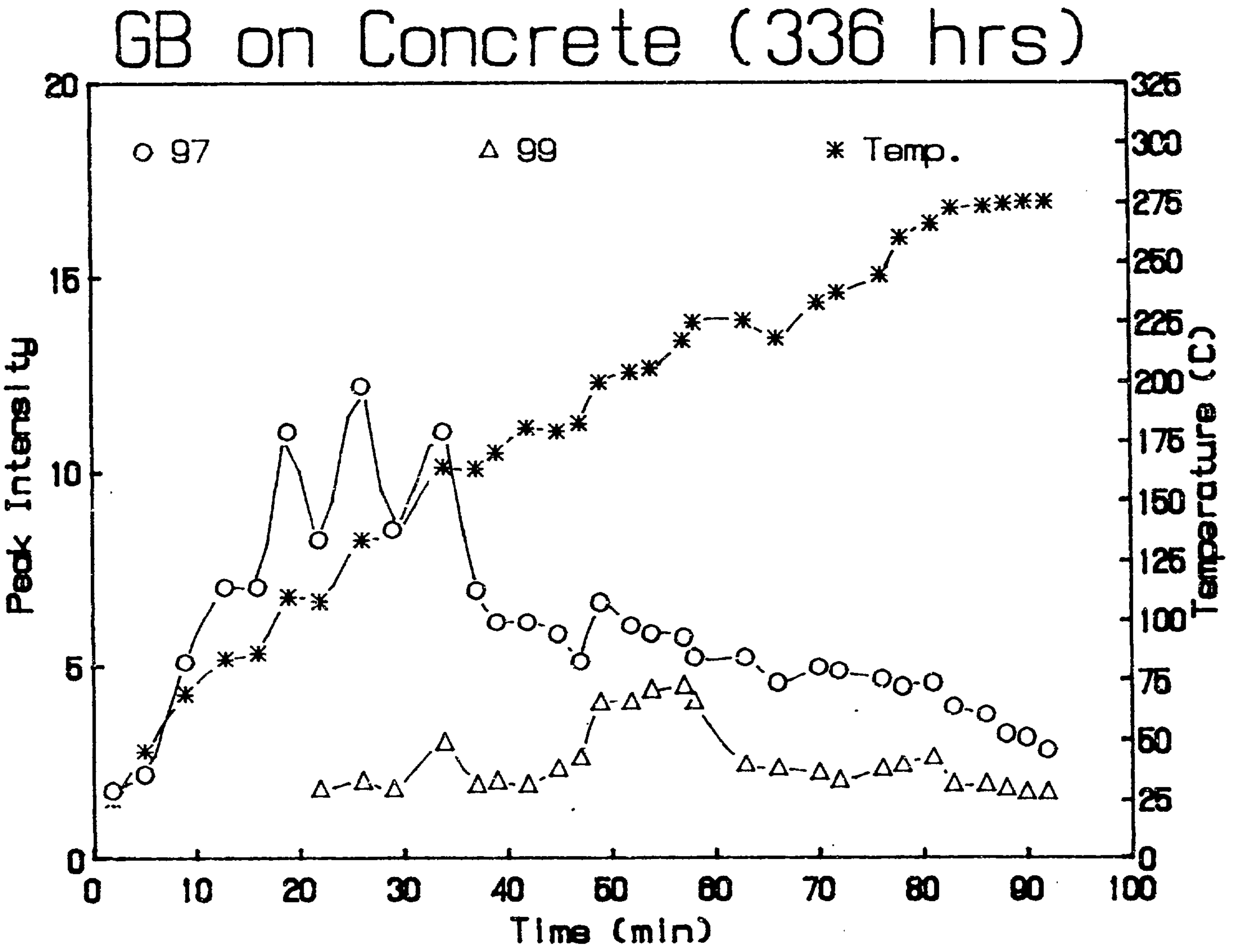


Figure 11. Plot of Peak Intensities (m/e = 97 and 99) for 336 Hour GB Sample.

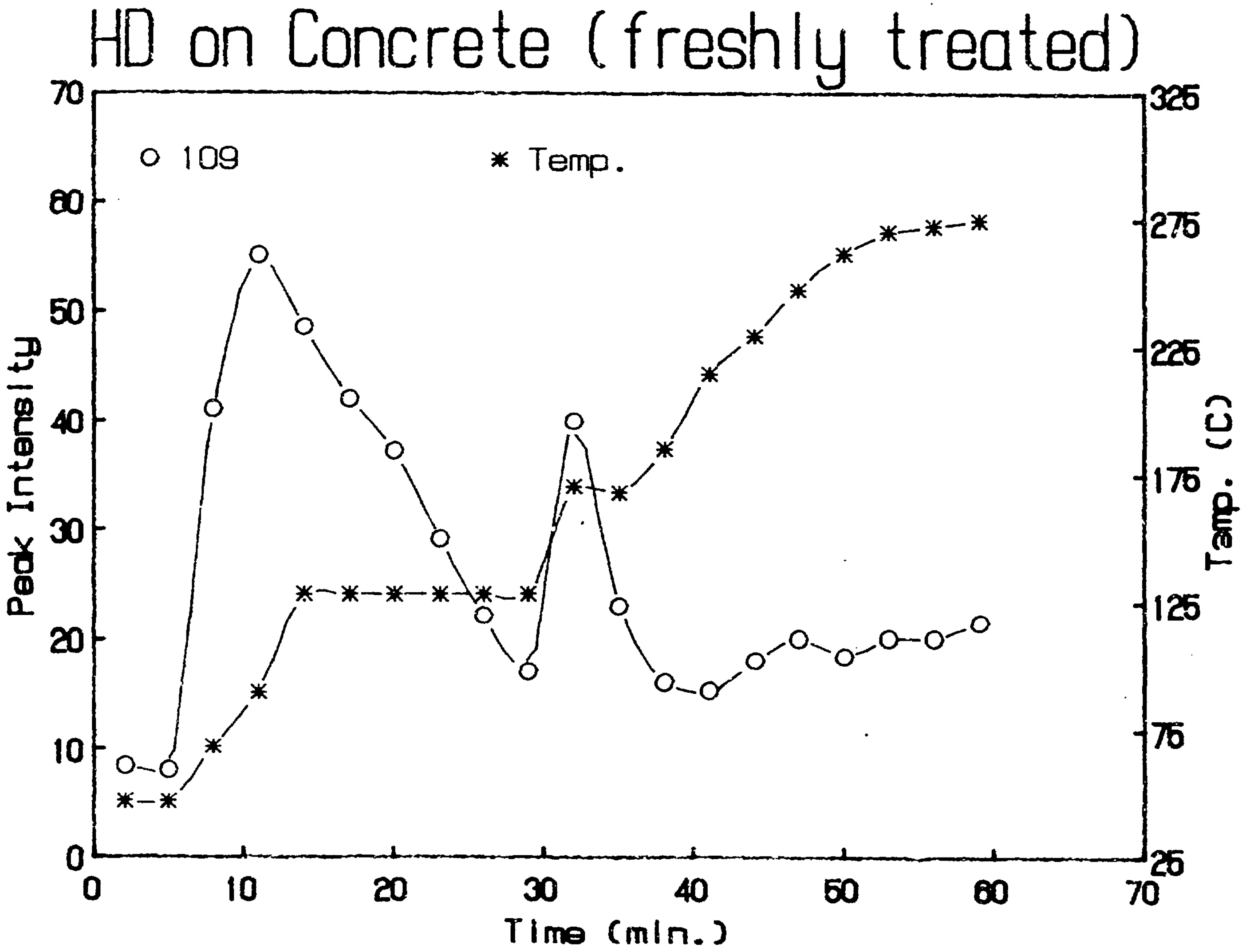


Figure 12. Plot of Peak Intensities (m/e = 109) for Fresh HD Sample.

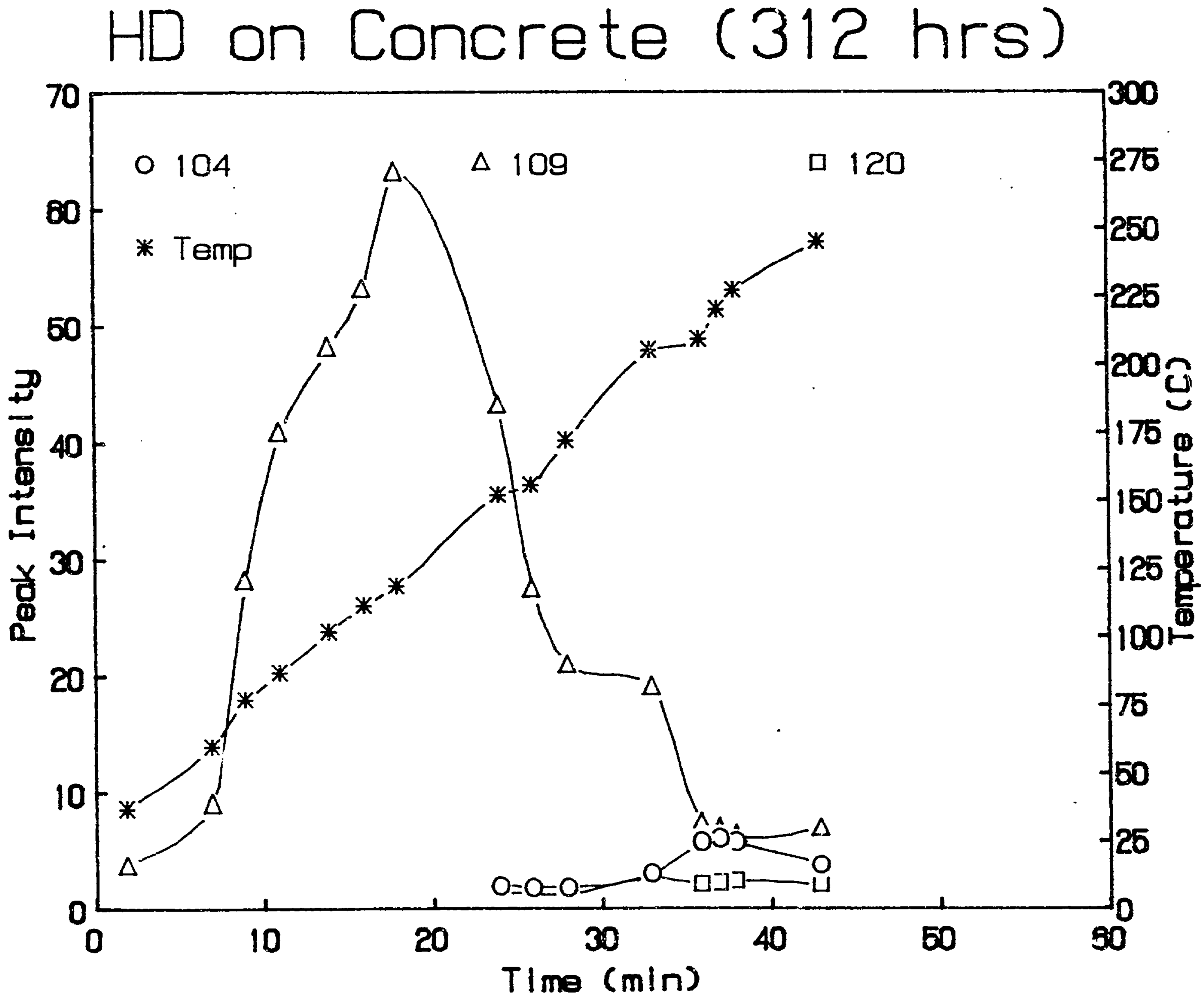


Figure 13A. Plot of Peak Intensities (m/e = 104, 109 and 120) for 312 Hour HD Sample.

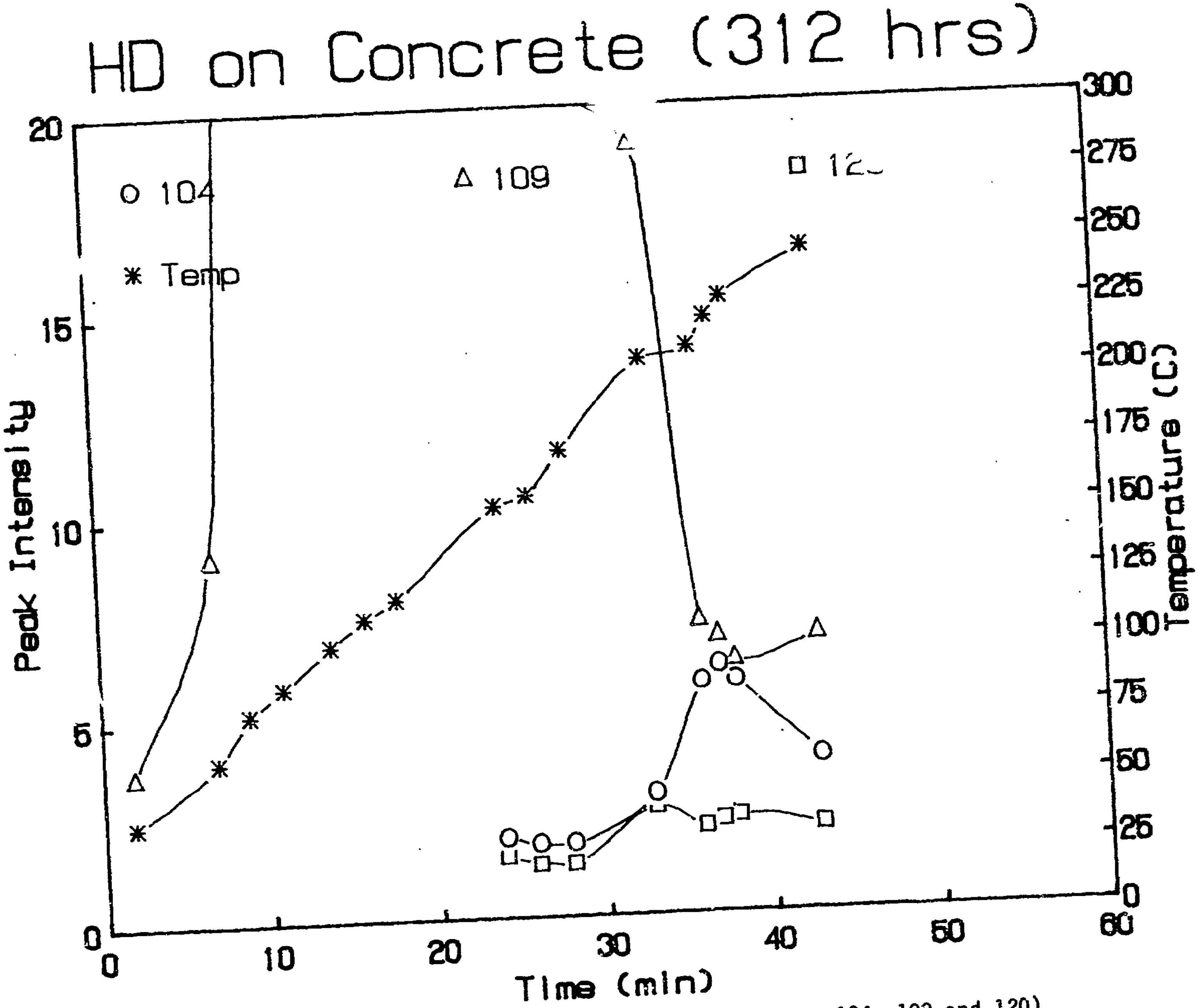


Figure 13B. Plot of Peak Intensities (m/e = 104, 109 and 120) for 312 Hour HD Sample (finer peak intensity scale).

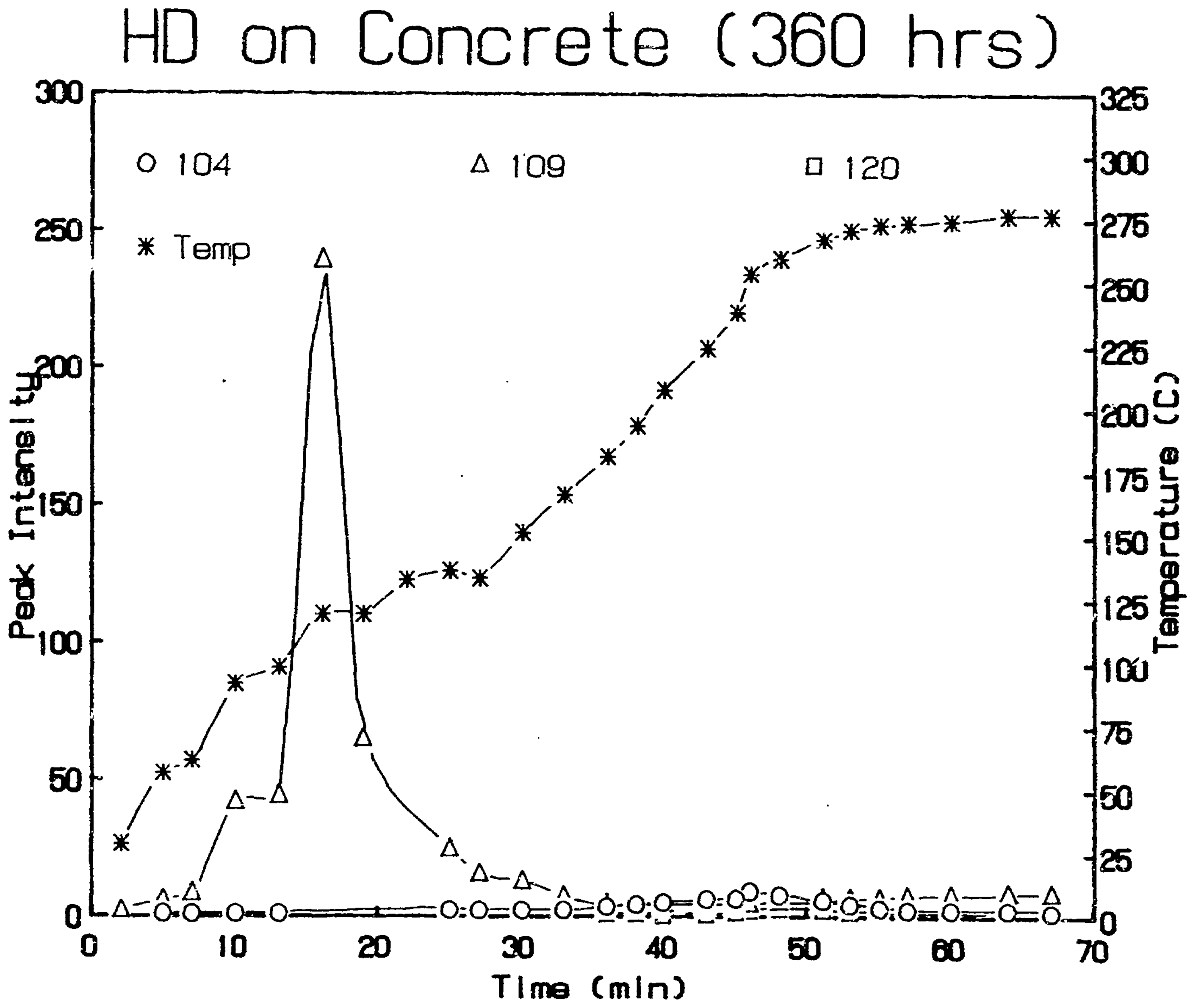


Figure 14A. Plot of Peak Intensities (m/e = 104, 109 and 120) for 360 Hour HD Sample.

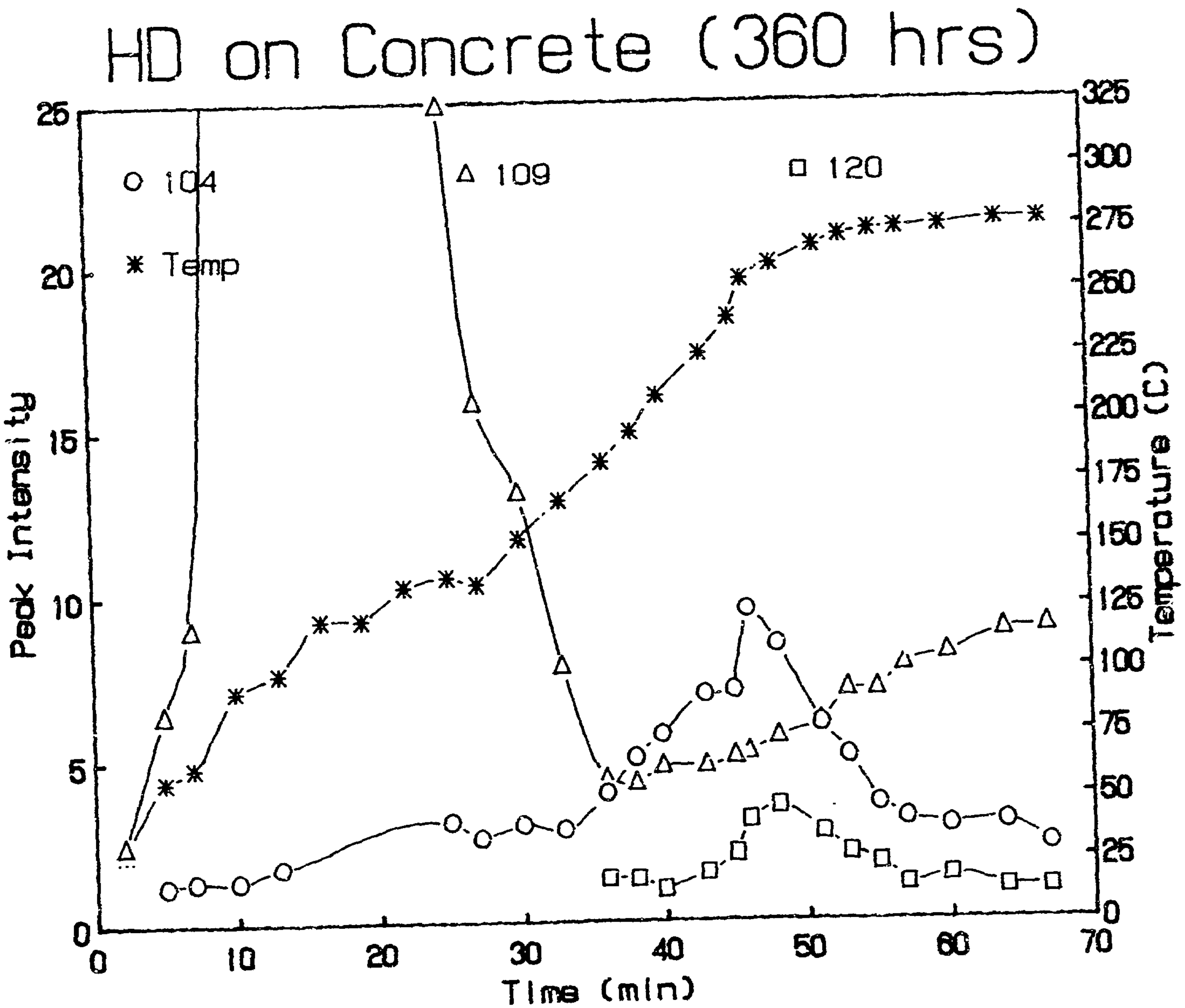


Figure 14B. Plot of Peak Intensities for 360 Hour HD Sample (finer peak intensity scale).

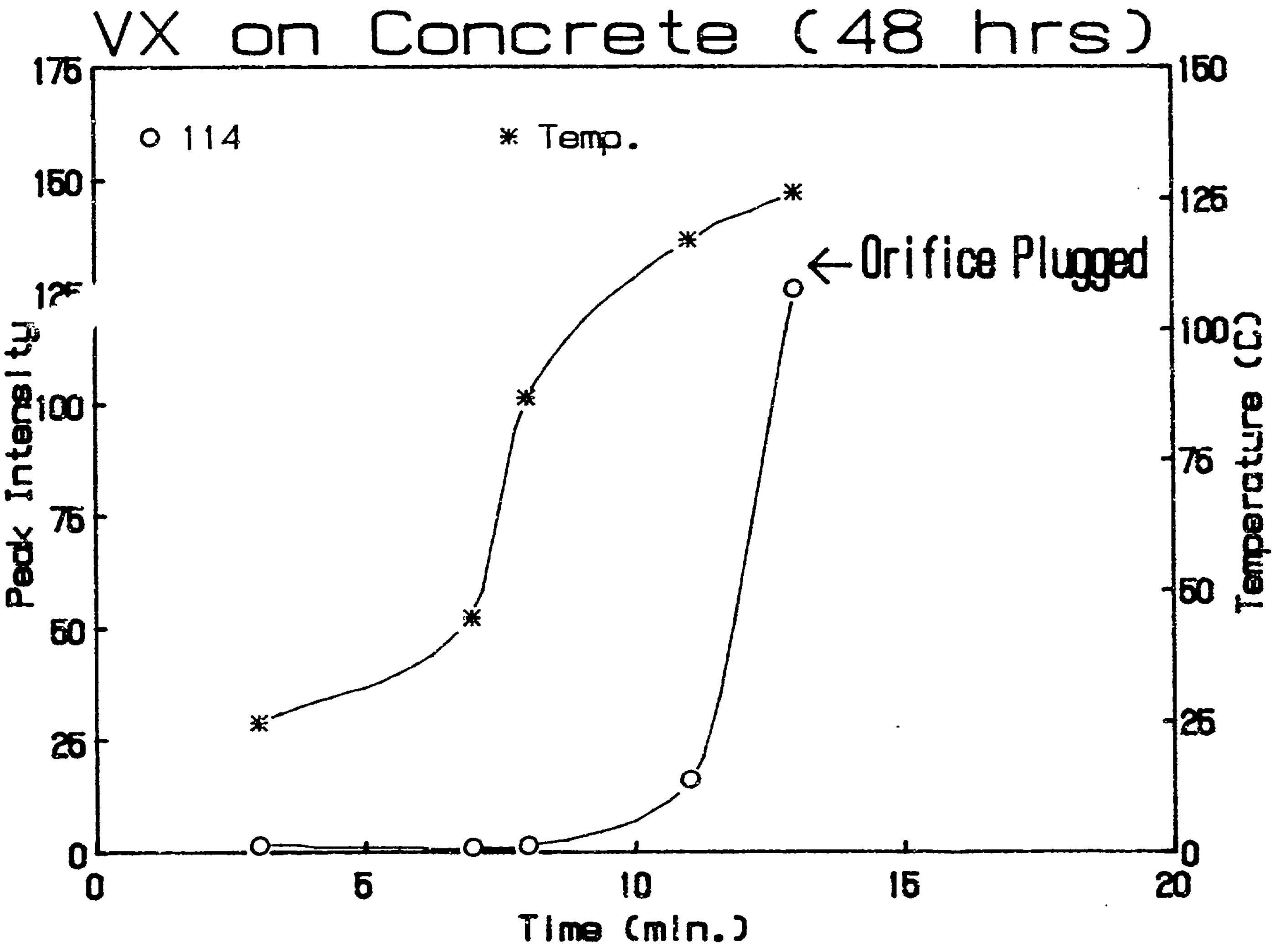
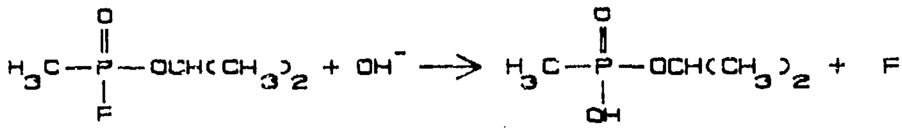


Figure 15. Plot of Peak Intensities (m/e = 114) for 48 Hour VX Sample.

increased in the older sample so that the GB has penetrated deeper into the concrete matrix.

The formation of the IMP probably results from the GB reacting with hydroxide (OH^{-1}) ions present in the moist concrete. This reaction is shown below in Equation 4. The hydroxide ion (OH^{-1}) appears to simply substitute for the fluoride ion (F^{-1}) in the GB structure. The mass spectra for GB and IMP were found to be identical with the exception that the IMP spectrum was shifted down two mass units. This shift can be accounted for by the substitution of an OH^{-1} for the F^{-1} in GB. Any major restructuring of the molecule would be evidenced by more significant differences in the two mass spectra. Figure 16 shows a comparison of the mass spectra for the GB and IMP, while Figure 17 shows a typical mass spectrum from the desorption of the aged samples in which both GB and IMP are present.



$$H_3C-\underset{\underset{\displaystyle F}{|}}{\overset{\overset{\displaystyle O}{\|}}{P}}-OCH(CH_3)_2 + OH^- \longrightarrow H_3C-\underset{\underset{\displaystyle OH}{|}}{\overset{\overset{\displaystyle O}{\|}}{P}}-OCH(CH_3)_2 + F^- \qquad (4)$$

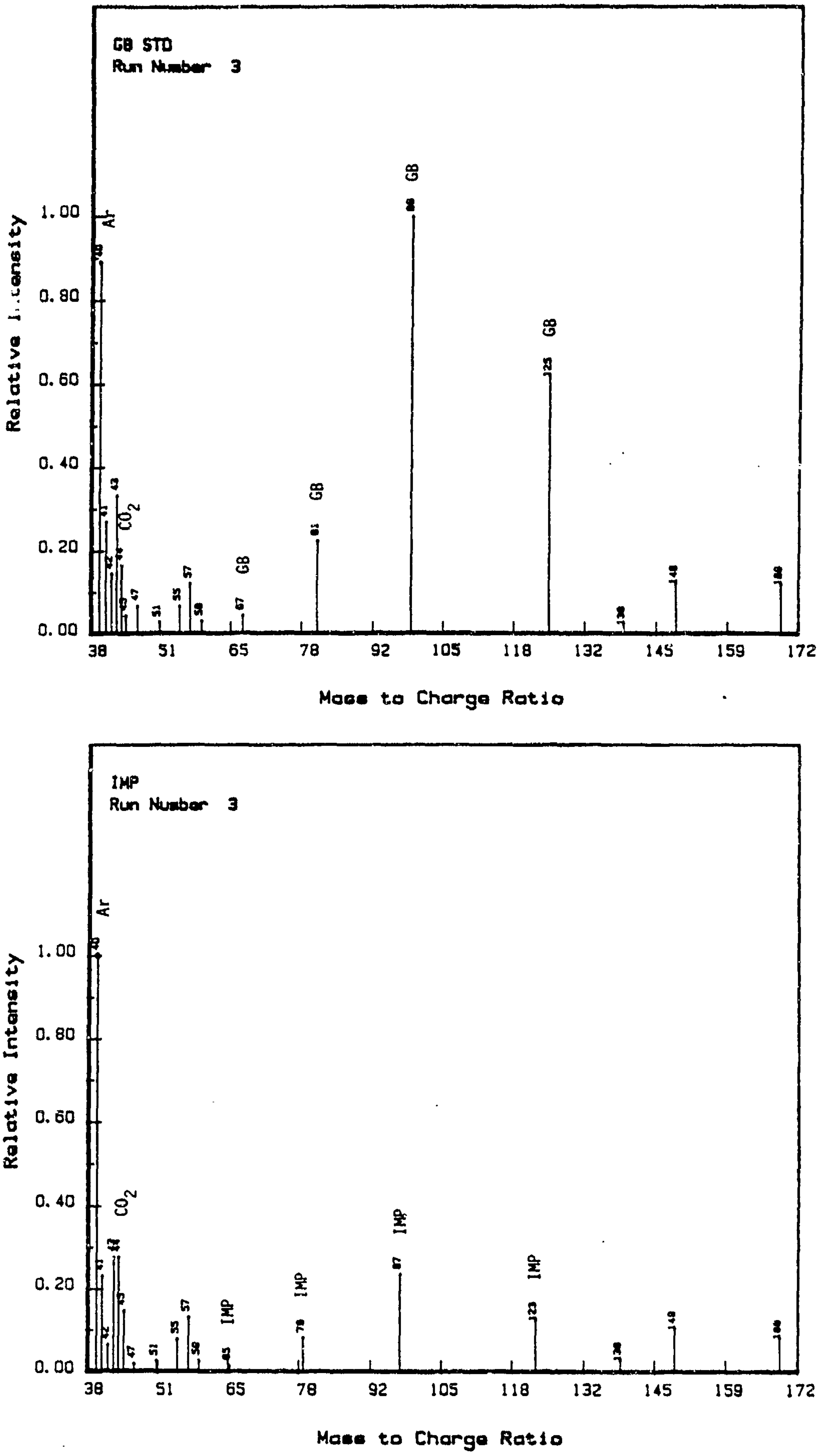
GB IMP

Using equation 2 an estimate was made of the amount of GB recovered from the three GB spiked samples. Approximately 85 percent of the 25 ul of GB applied to the freshly spiked sample was recovered. This estimate seems reasonable because some GB was lost between the time the sample was spiked and the sample chamber was loaded onto the molecular-beam inlet. For the two aged samples, recovery estimates were made by considering the sum of the GB and IMP peaks. Fifty-seven percent of the GB appeared to be recovered from the 120 hour GB sample, and 30 percent of the GB appeared to be recovered from the 336 hour GB sample.

3.3.2.2 HD Runs

As was found with GB, the only desorption product identified for the freshly spiked HD sample was the agent itself. For the two aged HD samples, the desorption products found were HD, 1,4-dithiane, and 1,4-oxathiane. Of these three desorption products, HD was by far the most abundant product as can be seen on Figures 13 and 14. Between the 1,4-dithiane and the 1,4-oxathiane, the 1,4-oxathiane appeared to be more abundant. Again the presence of desorption products besides that of the agent itself on the aged samples

Figure 16. Comparison of Mass Spectra for GB and IMP.



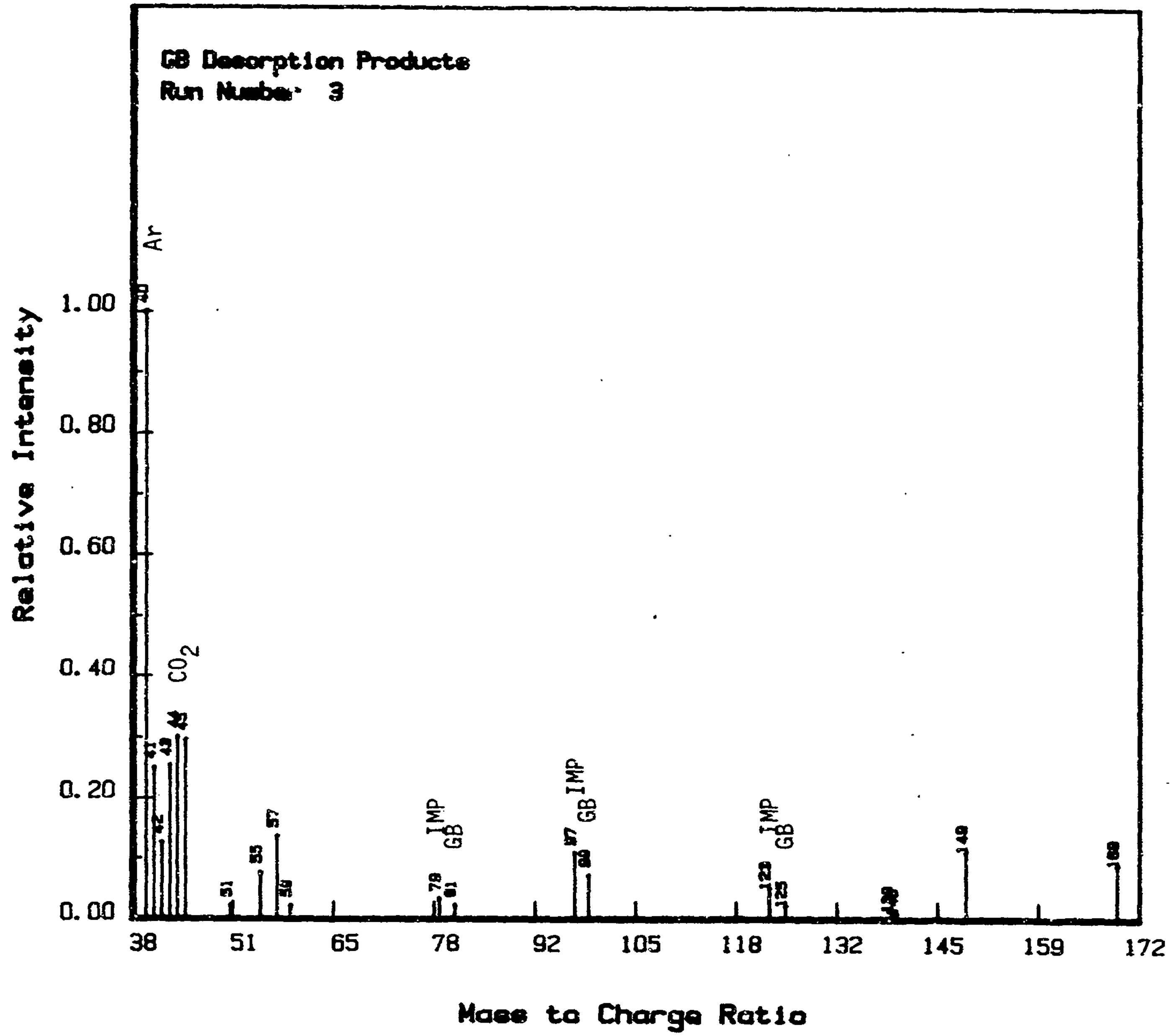


Figure 17. Mass Spectrum for Desorption of GB and IMP from an Aged GB Sample.

indicates the importance of time and moisture in order for the agent to react with the concrete. Figure 18 shows a mass spectrum for an aged HD sample in which HD, 1,4-oxathiane, and 1,4-dithiane are present.

As is shown on Table 4, HD was first detected in the temperature range of 29-47°C which is within the range of ambient temperatures to which the concrete could be exposed. On the aged samples, the 1,4-oxathiane was first detected at around 170°C in both cases. The 1,4-dithiane was first detected at the same temperature as 1,4 oxathiane in the 312 hour sample and at a temperature 41°C higher in the 360 hour sample. The greater temperature required to detect the 1,4-dithiane in the 360 hour sample suggests that the 1,4-dithiane may have been located deeper in the concrete matrix on the 360 hour sample than on the 312 hour sample.

The 1,4-oxathiane and 1,4-dithiane are cyclic compounds that appear to be the reaction products of the de-chlorination reactions which occur in the concrete. Shown below in Figure 19 is a comparison of the structure of HD with the ring structures of 1,4-oxathiane and 1,4-dithiane.

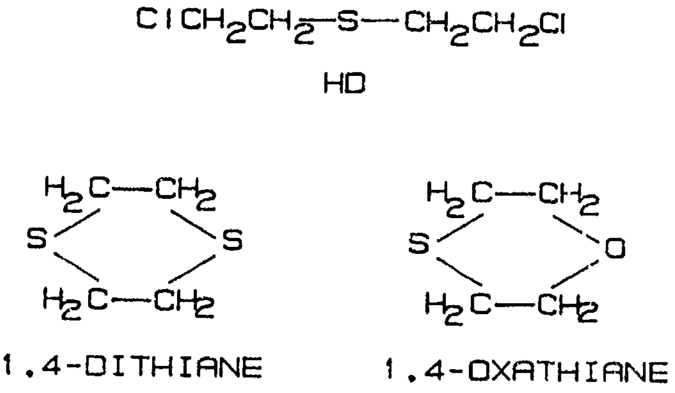


Figure 19. Structural Comparison of HD and its Desorption Products.

The formation of 1,4-oxathiane requires hydrolysis of the HD molecule. Here the HD reacts with hydroxide (OH^{-1}) ions and eventually liberates HCl to form the 1,4-oxathiane. Equation 5 shows the hydrolysis reaction.

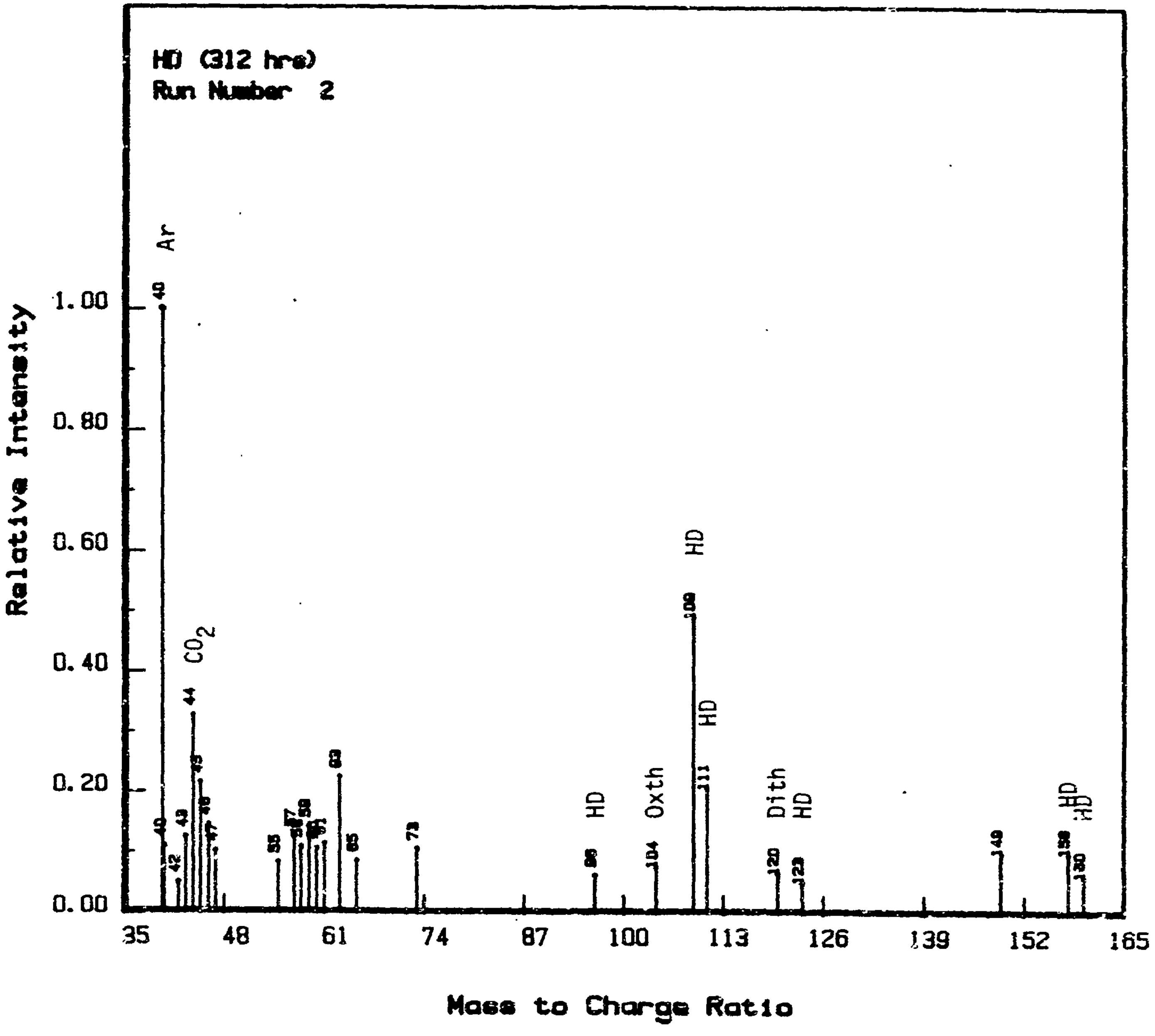
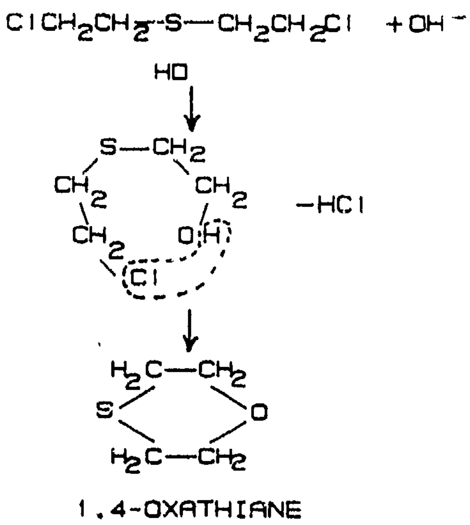


Figure 18. Mass Spectrum of Aged HD Sample Showing the Three Desorption Products.



(5)

The formation of the 1,4-dithiane results from the thermal degradation of HD. This process involves the association of 2 HD molecules with the elimination of dichloroethane to form the thermal degradation product 1,4-dithiane. Thermal degradation of HD to form 1,4-dithiane occurs at room temperature but is accelerated by heating.

From equation 2, it was estimated that 85 percent of the 25 μl of HD applied to the freshly spiked sample was recovered. As with the freshly spiked GB sample, this estimate seems reasonable because some HD was lost between the time the sample was spiked and the sample chamber was loaded onto the molecular-beam inlet. For the two aged samples, recovery estimates were considered using only the HD peaks. No attempt was made to use the 1,4-oxathiane or 1,4-dithiane peaks. On the two aged samples, 39 percent HD recovery was found on the 312 hour sample, and 50 percent HD recovery was found on the 360 hour sample.

3.3.2.3 VX Runs

The analysis of desorption products from VX spiked samples was found to be difficult. In particular, once the VX was heated above 140°C, a liquid condensed on the skimmer orifice and plugged it so that no further mass scans could be taken. An analysis of this liquid with the mass spectrometer indicated the liquid was VX. As a result of the VX condensation problem, only one run was conducted on a VX spiked sample. Modifications such as mounting a heater inside the molecular-beam sampling head would have to be made to the present apparatus before the desorption products of VX could be analyzed at temperatures above 140°C.

The one VW sample analyzed was a sample that had aged 48 hours. On this sample, the only desorption product found before the skimmer orifice plugged at 142°C was VX itself. As can be seen on Figure 15, no significant VX was detected until the sample temperature reached 87°C. The VX peak intensity continued to increase at temperatures above 87°C until the orifice became plugged at 142°C. Figure 20 shows a typical mass spectrum from the aged VX sample. No estimates were made of the amount of VW recovered because of the limited number of points that could be measured.

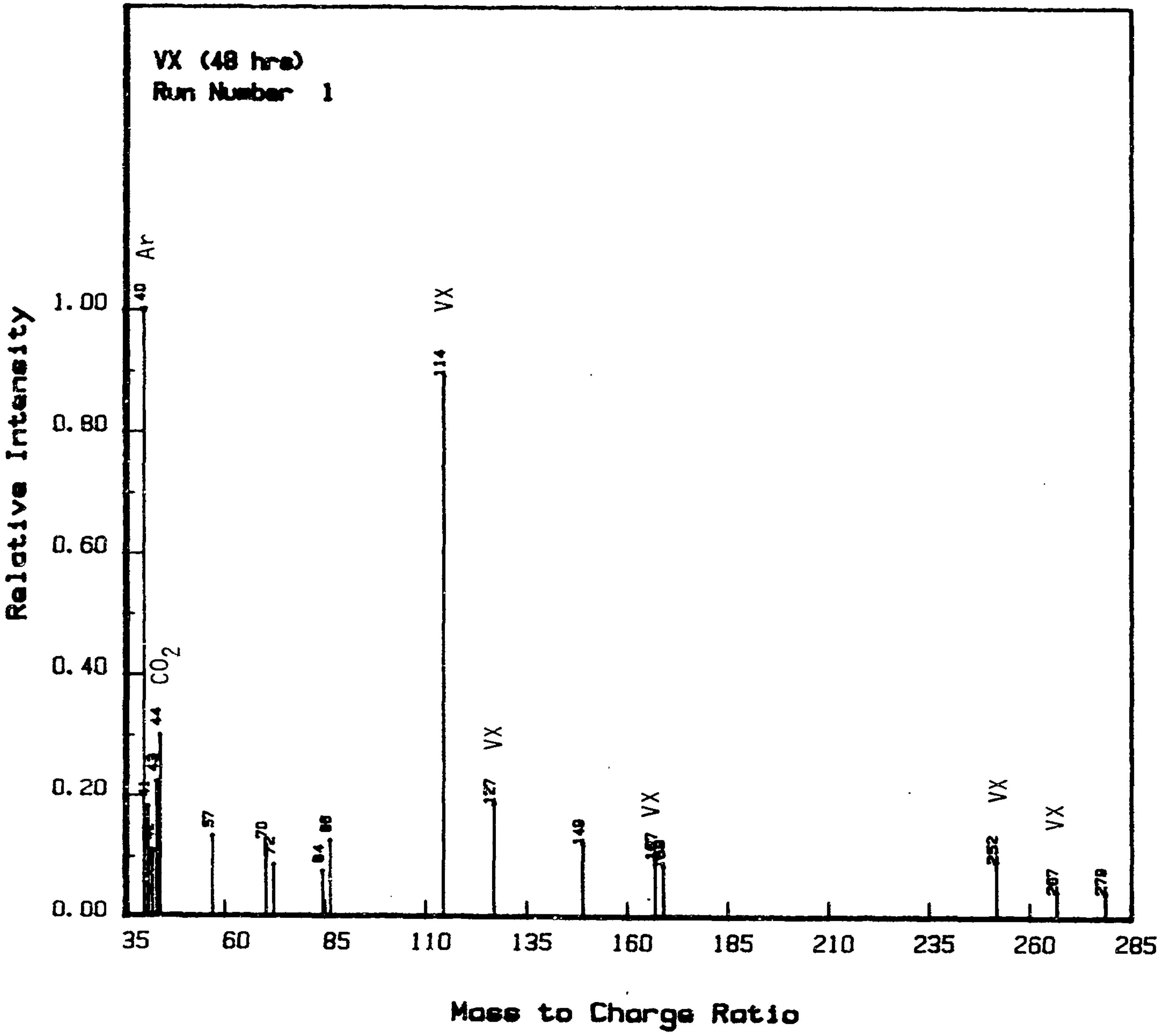


Figure 20. Mass Spectrum from Aged VX Sample.

4.0 CONCLUSIONS AND RECOMMENDATIONS

The HDP model, in its current form, provides a convenient and rapid means for examining the thermal consequences of variation in HDP parameters. However, although provisions were made for incorporation of chemical parameter, the model is currently incomplete with respect to description of chemical release rates. It is recommended that consideration be given to completion of the HDP model.

Based on the current spiking experiments, it is concluded that all three agents (HD, GB, and VX) are degraded in the presence of concrete. The degradation of HD is relatively slow, with apparent half-life of about 1747 hours. Indeed, this process may be slowed further by consumption of the basic components of the concrete. The principal product of reaction of HD with concrete is oxathiane. Dithiane also was observed in the thermal desorption experiments, but this substance is believed to be formed by a normally slow rearrangement process of the mustard which is accelerated by the heating process. That is, the dithiane does not result from reaction with the concrete. The reaction of GB with concrete is rapid and involves exchange of the flourine group of the GB with hydroxyl groups derived from the basic components of the concrete. The reaction of VX with concrete is slow initially, but the process of degradation of VX appears to be autocatalytic. A complex mixture of products is formed by this process. The current work has resulted in chromatographic isolation of 43 such products. GC/MS identification of these products is currently tentative at best because of the paucity of agent related compounds in the current NBS data base. It is recommended that a more extensive examination of the samples and spectra from the degraded VX samples be performed using reference spectra available from other agencies concerned with agent analysis and using standard chemicals for generation of reference spectra where such information is not available.

References

1. Lienhardt, J.H., *A Heat Transfer Textbook*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1981.

2. Crank, J., *The Mathematics of Diffusion*, Clarendon Press, Oxford England, 1956.

3. Carnahan, B., *Applied Numerical Methods*, John Wiley & Sons, Inc., New York, 1969.

4. Zamejc, E.R., Mezey, E.J., Hayes, T.L., Wetzel, D.K., and Garrett, B.C., *Development of Novel Decontamination Techniques for Chemical Agents (GB, VX, HD) Contaminated Facilities, Phase II - Laboratory Evaluation of Novel Agent Decontamination Concepts*, Task Final Report, March 1985, Report No. AMXTH-TE-TR-85012.

APPENDIX A
THE MATHEMATICS OF HDP

THE MATHEMATICS OF HDP

The HDP involves a series of heat and mass flow elements:

1. heat flow to and mass flow from a hot surface
2. heat flow coupled with vaporization/condensation and pyrolysis within the structure
3. heat and mass flow from the cool side of the structure

The core of the problem in forming a mathematical description of the process lies in item 2, whereas the boundary conditions are derived from consideration of items 1 and 3. In the following discussions, abbreviated terms are explained within the text but all fundamental terms are defined in the glossary of symbols.

CONSERVATION EQUATIONS

At any point within the wall, conservation of heat requires that

$$\rho C_p \left(\frac{dT}{dt} \right) = k_s \left(\frac{\partial^2 T}{\partial x^2} \right) - q_e \left(\dot{m}_e - \dot{m}_a \right) - q_r k_r \left(C_a + C_g \right) \tag{1}$$

(total heat) = (sensible heat) - (heat of evap/cond) - (heat of pyrolysis)

For the adsorbed phase, conservation of mass requires

$$\left(\frac{dC_a}{dt} \right) = \frac{-2}{r} \left(\frac{\epsilon}{1-\epsilon} \right) \left(\dot{m}_e - \dot{m}_a \right) - k_r C_a \tag{2}$$

(surface concentration) = (evaporation/condensation) - (pyrolysis)

For the vapor phase, conservation of mass requires

$$\left(\frac{dC_g}{dt} \right) = D \left(\frac{\partial^2 C_g}{\partial x^2} \right) + \frac{2}{r} \left(\frac{\epsilon}{1-\epsilon} \right) \left(\dot{m}_e - \dot{m}_a \right) - k_r C_g \tag{3}$$

$$(\text{concentration}) = \left(\begin{matrix} \text{diffusive} \\ \text{transport} \end{matrix} \right) + \left(\begin{matrix} \text{evaporation/} \\ \text{condensation} \end{matrix} \right) - (\text{pyrolysis})$$

In these equations, the rates of evaporation, m_e, and condensation, m_a, are given by

$$\dot{m}_a = \frac{\gamma C_g \bar{v}}{4} \quad \text{where } \gamma \simeq e^{-\Delta S_v / 3R} \tag{4}$$

and $$\dot{m}_e = \frac{N_s M\bar{v}}{4RT}\, e^{\frac{2\Delta S_v}{3R}}\, e^{-\Delta H_v/RT} \qquad (5)$$

The rate constant for pyrolysis, k_r, is

$$k_r = Ae^{-E^*/RT} \qquad (6)$$

The solution to the problem is gained by simultaneous solution of equations 1-3 along with the appropriate initial and boundary conditions.

Boundary Conditions

To provide flexible use of the HDP model, 4 different boundary conditions are considered.

1. Both sides of wall heated.
2. One side of wall cooled by ambient air.
3. One side in contact with moist soil.
4. One side covered with insulation.

The general approach used requires determining the effect of each of these boundary conditions on the temperature gradient at the surface of the wall. In general, we seek to find an expression of the form

$$\left(\frac{dT}{dx}\right)_s = \beta_i \left(T_i - T_s\right) \qquad (7)$$

for each of the boundary conditions.

For a heated boundary, there is no heat loss at the boundary, so,

$$k_s \left(\frac{dT}{dx}\right)_s = h\left(T_q - T_s\right) \qquad (8)$$

where h is the overall heat transfer coefficient which is composed of contributions from radiation, natural convection, and forced convection, ie.

$$h = h_r + \left(h^2_{nc} + h^2_{fc}\right)^{1/2} \qquad (9)$$

Similarly, the boundary condition for an air cooled side is

$$k_s \left(\frac{dT}{dx}\right)_s = h_a\left(T_a - T_s\right) \qquad (10)$$

When one side is insulated,

$$k_s \left(\frac{dT}{dx} \right)_s = \frac{k'}{L'} \left(T_a - T_s \right) \tag{11}$$

where k' and L' are the conductivity and thickness of the insulation respectively. In the case of one side in contact with moist soil, the soil is a semi-infinite medium, ie. the depth of soil affected by heat transfer from the wall increases with time. Lienhardt (ref) suggests representing the depth of thermal penetration by

$$L_d = (\alpha t)^{1/2} \tag{12}$$

where α is the thermal diffusivity of the soil. The boundary equation then becomes

$$k_s \left(\frac{dT}{dx} \right)_s = \frac{k_d}{L_d} \left(T_d - T_s \right) \tag{13}$$

Heat and Mass Transfer Parameters

Forced Convection:

$$h_{fc} = \frac{k_a Nu}{d}$$

where

k_a = conductivity of hot gas = $1.845 \times 10^{-4} \left(\frac{Cp}{Pr} \right) \left(\frac{T_g}{298} \right)^{0.75}$ cal, cm^3, K^{-1}

Nu = Nusselt number = $0.023\ Re^{0.79}\ Pr^{1/3}$ (dimensionless)

Re = Reynolds number = $\frac{\rho Q}{d\mu} = \frac{9.04 \times 10^{3} Q}{d T_g} \left(\frac{298}{T_g} \right)^{0.75}$ (dimensionless)

Q = volumetric flow rate, cm^3 sec^{-1}

d = hydraulic diameter of heated space (taken normal to gas flow direction), cm

μ = gas viscosity, g cm^{-1} sec^{-1}

Pr = Prandl number $\simeq \dfrac{Cp}{0.0855 + Cp}$ (dimensionless)

Cp = heat capacity of gas = $(6.449 = 1.4113 \times 10^{-3} Tg - 8.07 \times 10^{-8} T^2 g)/29$, cal $g^{-1} K^{-1}$

Tg = gas temperature, K

Natural Convection:

$$h_{mc} = 0.525 \left(\frac{ka}{d} \right) (PrGr)^{1/4}$$

where Gr = Grashof number = $\dfrac{\rho^2 \bar{\beta} g d^3}{\mu^2} \left(Tg - T_s \right)$ (dimensionless)

$\bar{\beta} = - \dfrac{1}{\rho} \left(\dfrac{d\rho}{dT} \right)$ = coefficient of volume expansion =

$7.51 \times 10^{-3} - 1.1 \times 10^{-5} T_g + 6.63 \times 10^{-9} T^2{}_g - 1.4 \times 10^{-12} T^3{}_g$, K^{-1}

g = acceleration of gravity = 980.7, cm sec^2

Radiation:

$$h_r = \sigma \epsilon^* \left(T^4{}_g - T^4{}_s \right) / \left(T_g - T_s \right)$$

σ = Stefan-Boltzmann constant = 1.355×10^{-12}, cal sec^{-1} cm^{-2} K^{-4}

ϵ^* = effective emissivity = 0.37 (dimensionless)

Agent and Degradation Product Parameters

The following table summarizes transport and degradation parameters of agents and known degradation products.

SUMMARY OF CHEMICAL PARAMETERS RELEVANT TO HDP

		Vapor Pressure		Pyrolysis	
Chemical	D@298k cm^2/sec	ΔS_v eu	ΔH_v kcal/mole	log A sec^{-1}	E* kcal/mole
HD	.0706	29.4	13.4	9.25	25
GB	.066	27	10.9	8.18	23.2
VX	.065	29.3	16.3	8.98	27.8
DEMP	.066	28.8	11.5		

GLOSSARY OF SYMBOLS

Symbol	Meaning
A	Arrhenius pre-exponential factor
α	thermal diffusivity of soil
β	coefficient of expansion
C_a	concentration of adsorbed phase
C_g	concentration of vapor phase
C_p	heat capacity at constant pressure
D	effective diffusivity of vapor
d	hydraulic diameter
E^*	activation energy
ϵ	porosity of concrete
ϵ^*	emissivity
g	acceleration of gravity
γ	collisional reaction efficiency
Gr	Grashof number
h	overall heat transfer coefficient
h_{fc}	forced convection heat transfer coefficient
h_{nc}	natural convection heat transfer coefficient
h_r	radiation heat transfer coefficient
ΔH_v	enthalpy of vaporization
k'	thermal conductivity of insulation
k_r	rate constant for pyrolysis
k_s	thermal conductivity of concrete
L	thickness of concrete
L'	thickness of insulation

GLOSSARY OF SYMBOLS (Continued)

L_d	thickness of thermal penetration in soil
m_a	mass rate of adsorption
m_e	mass rate of vaporization
μ	viscosity
N_s	mole fraction of sorbate on surface
Nu	Nusselt number
Pr	Prandl number
Q	volumetric flow rate
q_e	heat of vaporization per unit mass
q_r	heat of pyrolysis
r	average pore size in concrete
R	gas constant
Re	Reynolds number
ρ	density
σ	Boltzmann constant
ΔS_v	entropy of vaporization
T	absolute temperature
t	time
T_a	air temperature
T_d	soil temperature
T_g	hot gas temperature
T_s	surface temperature
$\bar{v}$	mean molecular speed
x	distance

APPENDIX B
GC-MS ANALYSES OF CONCENTRATED VX ON CONCRETE SAMPLE

GC-MS ANALYSES OF CONCENTRATED VX ON CONCRETE SAMPLE

GC-MS analyses were conducted on the liquid taken from the sample in which 1 ml of VX was applied to 1 gram of pulverized concrete. These analyses were conducted after the sample had aged 1824 hours in an attempt to identify the degradation products of VX.

Figure 1 shows the GC scan that was obtained from the liquid. The mass spectra that were identified for each GC peak were subjected to a Hewlett-Packard probability matching (PBM) search routine in an effort to identify the possible degradation products. This routine searches the NBS library and uses probability matching to identify the compounds whose mass spectra best match the mass spectra found for each GC peak. Table 1 shows a list of the possible product compounds that were identified for each of the 19 major GC peaks. It should be noted that the compounds shown in Table 1 only provide the best mass spectral match. However, additional testing would be required to determine if the peak retention times of the identified compounds match the peak retention times found for the sample liquid. Only by matching both retention time and mass spectra can a positive identification of the degradation products be made.

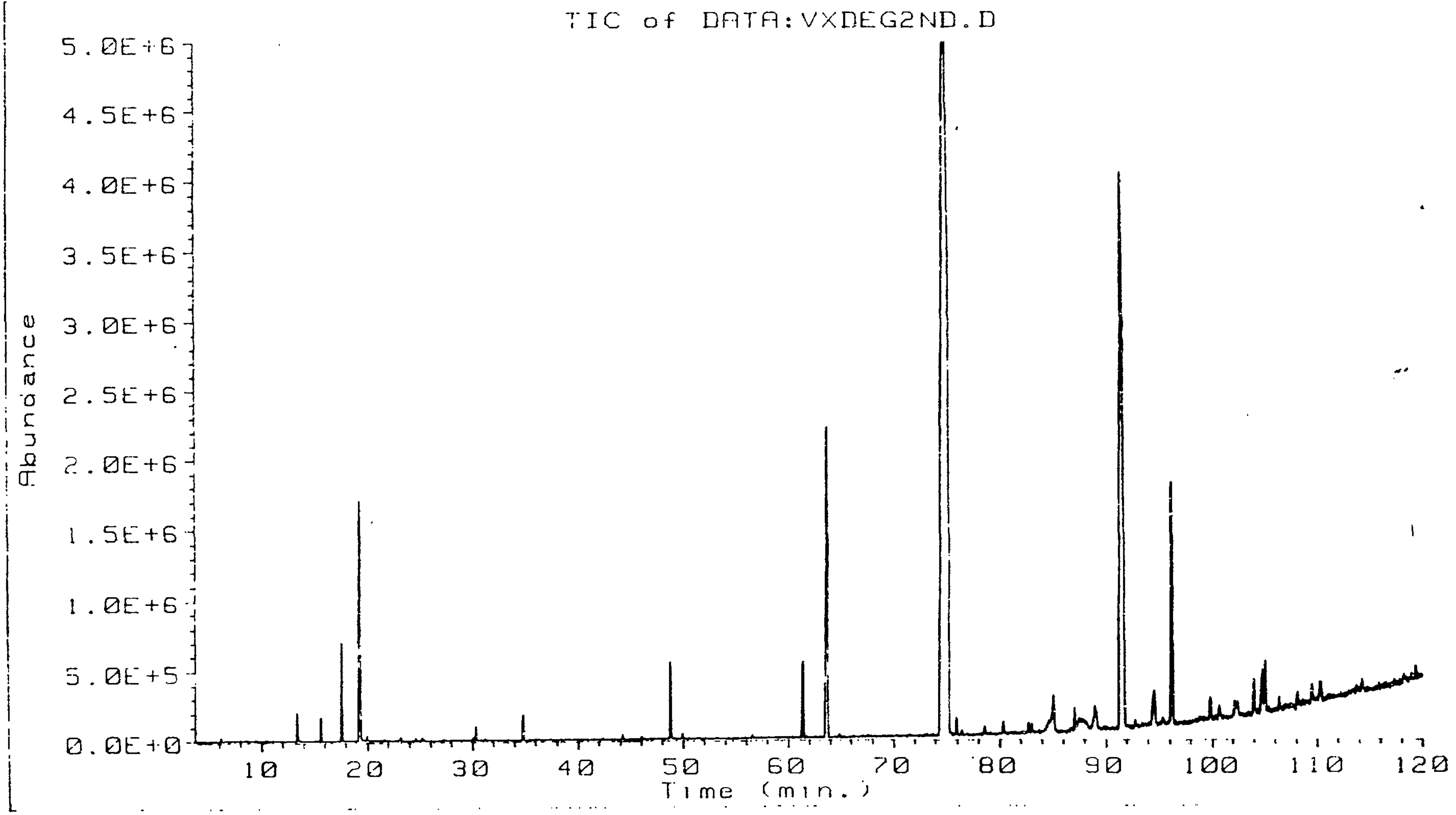


Figure B-1. GC Scan for VX on Concrete Sample Aged 1824 Hours.

TABLE B-1. POSSIBLE DEGRADATION PRODUCTS IDENTIFIED IN PBM MASS SPECTRA SEARCH

Peak Retention Time (1) (min.)	Possible Product(s)		
	Name	Mol Wt (g)	Formula
13.488	Phosphonic acid, methyl-, diethyl ester (8CI)	152	$C_5H_{13}O_3P$
17.718	Could not be identified		
19.467	Ethanol, 2-[bis(1-methylethyl)amino]- (9CI)	145	$C_8H_{19}NO$
48.863	Phosphonic acid, methyl-, 2-[bis(1-methylethyl) amino] ethyl ester (9CI)	251	$C_{11}H_{26}NO_3P$
61.45	Glycine, anhydride (8CI9CI)	132	$C_4H_8N_2O_3$
63.811	2,4 (1H,3H)-Pyrimidinedione, dihydro- (9CI)	114	$C_4H_6N_2O_2$
74.943	1-Propanamine, N,N-dipropyl- (9CI)	143	$C_9H_{21}N$
	2,5-Piperazinedione (8CI9CI)	114	$C_4H_6N_2O_2$
	2,4(1H,3H)-Pyrimidinedione, dihydro-(9CI)	114	$C_4H_6N_2O_2$
75.048	2-Imidazolidinone, 1,3-dimethyl- (8CI9CI)	114	$C_5H_{10}N_2O$
	2,4(1H,3H)-Pyrimidinedione, dihydro- (9CI)	114	$C_4H_6N_2O_2$
75.124	Ethanol, 2-[bis(1-methylethyl)amino]- (9CI)	145	$C_8H_{19}NO$
75.183	2,4-Imidazolidinedione,3-methyl- (9CI)	114	$C_4H_6N_2O_2$
75.228	1-Propanamine, N,N-dipropyl- (9CI)	143	$C_9H_{21}N$
75.278	2-Imidazolidinone, 1,3-dimethyl- (8CI9CI)	114	$C_5H_{10}N_2O$
91.625	2,5-Piperazinedione (8CI9CI)	114	$C_4H_6N_2O_2$
	2,4(1H,3H)-Pyrimidinedione, dihydro- (9CI)	114	$C_4H_6N_2O_2$
91.822	2,5-Piperazinedione (8CI9CI)	114	$C_4H_6N_2O_2$
	2-Imidazolidinone, 1,3-dimethyl- (8CI9CI)	114	$C_5H_{10}N_2O$
	2,4(1H,3H)-Pyrmidinedione, dihydro- (9CI)	114	$C_4H_6N_2O_2$
94.597	2H-1-Benzopyran-2-one,7-[(3,7-dimethyl-2, 6-Octadienyl)oxy]-(E)-9CI)	298	$C_{19}H_{22}O_3$
96.356	2,5-Piperazinedione (8CI9CI)	114	$C_4H_6N_2O_2$
	2,4(1H,3H)-Pyrimidinedione, dihydro- (9CI)	114	$C_4H_6N_2O_2$
104.035	Phosphonothioic acid, methyl-, O-[2-[bis(1-methy ethyl)amino]ethyl]] O-ethyl ester (9CI)	267	$C_{11}H_{26}NO_2PS$
104.744	Phosphonothioic acid, methyl-, O-[2-[bis(1-methy ethyl)amino]ethyl]] O-ethyl ester (9CI)	267	$C_{11}H_{26}NO_2PS$
105.108	2,4(1H,3H)-Pyriminedione, dihydro- (9CI)	114	$C_4H_6N_2O_2$

(a) Sample: 1 ml of VX applied to 1 gram of pulverized concrete and sample allowed to age 1824 hours.

APPENDIX C

Source Code
for
Hotgas Desorption Model

SOURCE CODE

```
#include <math.h>
#include <float.h>
#include <stdio.h>
#include <conio.h>
#include <graph.h>
#include <ctype.h>
#include <malloc.h>
#include <process.h>
char far *buffer;char bufl[10];
int BFL[10],i,jj,j,k,NP;float pl[15];
float q=5000.0;float ks=.0029;float cp=.21;float tg=810.77;float l=20.32;
float ti=294.111;float kp=.000207;int ST=12;float STZ;
float ts=285.78;float NX,ta,re,pr,nu,hfc,ka,gra,ha,kd,ad;
float he=12;float wi=12;float d;int ypix=350;
float rho=2.3;float rhod,cpg,cpd; float wa=25.0;
float far th[43][355];
float yp,delx,delt,hy,hx,hr,gr,hc,lp,n,x,tt,dth,ht;int nz;
float far cs[43][355];float far cg[43][355];
char resp[2];FILE *stream;char presp[2];char mode;
char fname[12]; char res;char buf[90];int far tm[640];int m[12];
#define LPT1 0;
#define COM1 0;
struct rccoord rcoord;
char cres;float coav;
double a,b,c;

main()

{
start:    for (i=0;i<=9;i++) BFL[i]=3;
     menu ();
matrix: printf ("%s","Number of hours to simulate ");
     scanf ("%f",&n);
     ST =pow(350*rho*cp*pow(l,2)/(n*ks*3600),.5)-2;if(ST>=42){ST=42;}
     printf ("%s %d","Maximum number of spatial nodes = ",ST);
     a=modf(ST*2.54/l,&b);ST=(int)(b*l/2.53);
     printf("\n%s %d","Recommended number  = ",ST);
     printf ("\n%s","Want to change ? (Y/N)");while(!kbhit());
     res=getch();res=tolower(res);if (res=='y')
     {printf ("\n%s","Input number of spatial nodes ");
     scanf ("%d",&ST);}
     _settextposition (16,30);_settextcolor (28);_outtext("Working
");
     re=(q*904000/(d*tg))*pow(298/tg,.75);
     cpg=(6.449+.001413*tg-(8.07e-08)*pow(tg,2))/29;
     pr=cpg/(.0855+cpg);nu=.023*pow(re,.793)*pow(pr,.333);
     ka=.0001845*cpg*pow(tg/298,.75)/pr;
     hfc=ka*nu/d;
     delx=(float) 1/ST;
```

```
   delt=(float) pow(delx,2)*rho*cp*pow(l,2)/(ks*3600);
   nz=(int)(1+n/delt);
   if (BFL[1]==11){for (j=0;j<=nz+4;j++)
   {for (i=0;i<=ST;i++){th[i][j]=ti/tg;}}}
   if (BFL[2]==11) {for (j=0;j<=nz+4;j++){for (i=0;i<=ST;i++){
   x=(float)(ST-i)/ST;th[i][j]=(x*ti/tg)+(1-x)*ta/tg;}}}
   if (BFL[4]==11) { ts=(ks*ti*lp+kp*ta*l)/(kp*l+ks*lp);
   for (j=0;j<=nz+4;j++){for (i=0;i<=ST;i++){
   x=(float)(ST-i)/ST;th[i][j]=(x*ti/tg)+(1-x)*ts/tg;}}}
   if (BFL[3]==11) {for (j=0;j<=nz+4;j++){for (i=0;i<=ST;i++){
   x=(float)(ST-i)/ST;th[i][j]=(x*ti/tg)+(1-x)*ts/tg;}}}
   switch(cres){  /*units of cs and cg not in right dimension*/
   case 'a':
      for (j=0;j<=nz+4;j++){for (i=0;i<=ST;i++){
      cs[i][j]=coav;}}
      break;
   case 'b':
      for (j=0;j<=nz+4;j++){for (i=0;i<=ST;i++){
      x=(float)(ST-i)/ST;cs[i][j]=coav*x*2;}}
      break;
   case 'c':
      for (j=0;j<=nz+4;j++){for (i=0;i<=ST;i++){
      x=(float)(ST-i)/ST;cs[i][j]=2*coav*(1-x);}}
      break;
   }
   jj=0;k=12;
iter:  dth=th[0][jj+1];for (j=jj;j<=k+2;j++){
   hr=(5.01e-13)*pow(tg,3)*(1-pow(th[0][j+1],4))/(1-th[0][j+1]);
   gr=fabs(((2.09e14)*pow(d,3)/pow(tg,3.5))*
   (1-th[0][j+1])/(1+pow(th[0][j+1],4.5)));
   hc=.525*(ka/d)*pow(pr*gr,.25);
   ht=sqrt(pow(hfc,2)+pow(hc,2))+hr;
   th[0][j+1]=.5*(th[1][j]+th[1][j+1]+(ht*l/ks)*
   (2-th[0][j]-th[0][j+1])*delx);
   if (BFL[1]==11){ th[ST][j+1]=th[0][j+1];}
   if (BFL[2]==11){gra=fabs((2.094e14)*(pow(d,3)/pow(ta,3.5))*
   (1-th[ST][j+1])/(1+pow(th[ST][j+1],4.5)));
   ha=.525*(ka/d)*pow(pr*gra,.25);
   th[ST][j+1]=.5*(th[ST-1][j]+th[ST-1][j+1]+
   (ha*l/ks)*((2*ta/tg)-th[ST][j]-th[ST][j+1])*delx);}
   if (BFL[3]==11){th[ST][j+1]=.5*(th[ST-1][j]+th[ST-1][j+1]+
   (kd*l/ks)*pow(((j+j+1)*.5*delt*ad),-.5)*
   (2*(ts/tg)-th[ST][j]-th[ST][j+1])*delx);}
   if (BFL[4]==11){th[ST][j+1]=.5*(th[ST-1][j+1]+th[ST-1][j]+
   (kp*l/(lp*ks))*(2*(ta/tg)-th[ST][j]-th[ST][j+1])*delx);}
   for (i=1;i<=ST-1;i++){
   th[i][j+1]=.25*(th[i+1][j+1]+th[i+1][j]+th[i-1][j]+th[i-1][j+1]);}}
   tt=dth-th[0][jj+1];
   if (fabs(tt)>=1e-07){ goto iter;}
   k=k+1;jj=jj+1;if (jj<=nz+4){goto iter;}
   output();
   if (resp[0]=='m'){goto start;}
```

```
}
output()
{
     if (BFL[7]==11) store();
     if (BFL[6]==11) plot();
     return(0);
}


menu ()
{
input:    _clearscreen (_GCLEARSCREEN);_setvideomode (_DEFAULTMODE);
       _settextcolor (14);_settextposition (1,30);
       _outtext ("Hot Gas Properties");
       d=60.96*he*wi/(he+wi);_settextcolor(2);
     _settextposition(2,5);_outtext ("Gas (F)low Rate = ");
     sprintf (buf1,"%4.0f",q);_outtext(buf1);_outtext (" cfm");
     _settextposition(2,50);_outtext ("(G)as Temperature = ");
     sprintf (buf1,"%3.1f",(tg-273)*1.8+32);_outtext(buf1);
     _outtext (" F\n");_settextcolor(14);_settextposition (3,30);
     _outtext ("Wall Properties");_settextcolor(2);_settextposition(4,5);
     _outtext ("(C)onductivity = ");sprintf(buf1,"%2.2f",ks*241);
     _outtext(buf1);_outtext(" Btu/ft-hr-F");_settextposition(4,50);
     _outtext ("(S)pecific Heat = ");sprintf(buf1,"%1.2f",cp);
     _outtext(buf1);_outtext(" Btu/lb-F");_settextposition(5,5);
     _outtext ("(D)ensity = ");sprintf(buf1,"%3.0f",(rho/0.016));
     _outtext(buf1);_outtext(" lb/ft^3");_settextposition(5,50);
     _outtext ("(T)hickness = ");sprintf(buf1,"%2.1f",(l/2.54));
     _outtext(buf1);_outtext(" in.");_settextposition(6,5);
     _outtext ("(H)eight, Width = ");
     sprintf(buf1,"%2.1f %s %2.1f",he,"x",wi);_outtext(buf1);
     _outtext(" ft.");_settextposition(6,50);
     _outtext ("(I)nitial Temperature = ");
     sprintf(buf1,"%3.1f",((ti-273)*1.8+32));_outtext(buf1);
     _outtext(" F");_settextcolor(14);_settextposition (8,30);
     _outtext ("Boundary Conditions");_settextposition (9,10);
     _settextcolor (BFL[1]);_outtext ("(1). Both sides heated");
     _settextposition (9,50);_settextcolor (BFL[2]);
     _outtext ("(2). Air cooled side");_settextposition (10,10);
     _settextcolor (BFL[3]);_outtext ("(3). Moist soil side");
     _settextposition (10,50);_settextcolor (BFL[4]);
     _outtext ("(4). Insulated side");_settextposition (12,30);
     _settextcolor (14);_outtext ("Output Choices");
     _settextposition (13,10);_settextcolor (BFL[5]);
     if (BFL[5]==3){_outtext ("(5). IBM Printer");}
     if (BFL[5]==11){_outtext ("(5). Epson Printer");}
     _settextposition (13,30);_settextcolor (BFL[6]);
     _outtext ("(6). Plotter");_settextposition (13,50);
     _settextcolor (BFL[7]);_outtext ("(7). Disk");
     _settextposition (14,30);_settextcolor (14);
     _outtext ("Chemical Agent");_settextposition (15,10);
```

```
_settextcolor (BFL[8]);_outtext ("(8). HD");_settextposition (15,30);
_settextcolor (BFL[9]);_outtext ("(9). GB");_settextposition (15,50);
_settextcolor (BFL[0]);_outtext ("(0). VX");
if ((BFL[8]+BFL[9]+BFL[0])==17){
if((BFL[6]+BFL[7])>=14){
if((BFL[1]+BFL[2]+BFL[3]+BFL[4])==20){yp=5;
_settextposition (16,30);_settextcolor (12);
_outtext ("(R)eady to Calculate");}}}
printf ("\n\n%s","Set up
problem by pressing letter or number key for each ");
printf ("\n%s\n","variable you wish to change or set");
 while(!kbhit()); resp[0]=getch();resp[0]=tolower(resp[0]);
switch (resp[0]){
case 'g':
     printf (" Input gas temperature (F) ");
     scanf ("%f",&tg);
     tg=((tg-32)/1.8)+273;
     break;
case 'f':
     printf ("Input volumetric flow rate (cfm)  ");
     scanf ("%f",&q);
     break;
case 's':
     printf ("Input specific heat (Btu/lb-F) ");
     scanf ("%f",&cp);
     break;
case 'i':
     printf ("Input initial temperature (F)");
     scanf ("%f",&ti);ti=(.5555*(ti-32)+273);
     break;
case 'c':
     printf ("Input Thermal conductivity  (Btu/hr-ft-F) ");
     scanf ("%f",&ks);ks=ks/241;
     break;
case 'd':
     printf ("Input density (lb/ft^3) ");
     scanf ("%f",&rho);rho=rho*.016;
     break;
case 't':
     printf ("Input wall thickness (in.) ");
     scanf ("%f",&l);l=l*2.54;
     break;
case 'h':
     printf ("Input room height (ft)");
     scanf ("%f",&he);
     printf ("Input room width (ft)");
     scanf ("%f",&wi);
     d=30.48*(4*he*wi/(2*he+2*wi));
     break;
case 'r':
     if (yp<=1){_settextcolor(12);_settextposition(20,1);
     _outtext("You have not selected the minimum blue items
```

```
    --press enter to continue");
    while(!kbhit());break;}
    return (0);
case '1':
    for (i=2;i<=4;i++){BFL[i]=3;}
    BFL[1]=14-BFL[1];
    break;
case '2':
    BFL[2]=14-BFL[2];BFL[1]=3;
    BFL[3]=3;BFL[4]=3;
     intf ("Input air temperature (F)");
     anf ("%f",&ta);ta=273+.5555*(ta-32);
    break;
case '4':
    BFL[4]=14-BFL[4];
    BFL[1]=3;BFL[2]=3;BFL[3]=3;
    printf ("%s\n\r","Thermal conductivity
    of insulation = 0.05 Btu/hr-ft-F ; Want to Change (Y/N)");
    while(!kbhit());res=getch();res=tolower(res);
    if(res=='y'){printf("Input conductivity of
    insulation (Btu/hr-ft-F)");
    scanf ("%f",&kp);kp=kp/241;}
    printf ("Input thickness of insulation (in)");
    scanf ("%f",&lp);lp=lp*2.54;
    printf ("\n\r%s","Input outside air temperature (F)");
    scanf ("%f",&ta);ta=273+.5555*(ta-32);
    break;
case '3':
    BFL[3]=14-BFL[3];BFL[4]=3;
    for (i=1;i<=2;i++){(BFL[i]=3);}
    printf("Soil temperature = 55 F;  Want to change ? (Y/N)");
    while (!kbhit());res=getch();res=tolower(res);
    if(res=='y'){printf ("\n\r%s","Input soil temperature (F)");
    scanf ("%f",&ts);ts=273+.5555*(ts-32);}
    printf ("\n\r%s %2.1f %s","Moisture content = ",wa,"%
    Want to change ? (Y/N) ");
    while (!kbhit());res =getch();res= tolower(res);
    if (res=='y'){printf ("\n\r%s","Input soil moisture
    content (%)");
    scanf ("%f",&wa);}
    rhod=(wa/100)+(1-wa/100)*(1.5);cpd=(wa/100)+.2*(1-wa/100);
    kd=wa*.000014+.0024*(1-wa/100);ad=kd*3600/(rhod*cpd);ta=ts;
    break;
 case '7':
    BFL[7]=14-BFL[7];
    printf ("(S)tore new data or (R)etrieve old data ?");
    while(!kbhit());res=getch();res=tolower(res);
    printf ("\n\r%s","Input file name ");
    scanf ("%s",fname);
    if (res=='r'){retrieve();
    BFL[7]=14-BFL[7];}
    break;
```

```
    case '5':
        BFL[5]=14-BFL[5];
        break;
    case '6':
        BFL[6]=14-BFL[6];
        break;
    case '8':
        BFL[8]=11;BFL[9]=3;BFL[0]=3;
        chem();
        break;
    case '9':
        BFL[9]=11;BFL[8]=3;BFL[0]=3;
        chem();
        break;
    case '0':
        BFL[0]=11;BFL[8]=3;BFL[9]=3;
        chem();
        break;
    default:
        printf ("Try again"); }
    goto input;

}

chem()
{
    _clearscreen (_GCLEARSCREEN);if(_setvideomode(_ERESCOLOR)==0)
    {ypix=200;if(_setvideomode (_HRES16COL   )==0){_setvideomode(_HRESBW);}}
    _setcolor(11);_rectangle(_GBORDER,20,0,170,(.2857*ypix));
    _rectangle(_GBORDER,210,0,370,(.2857*ypix));
    _rectangle(_GBORDER,410,0,570,(.2857*ypix));
    _setcolor(14);_moveto(20,50);_lineto(170,50);
    _moveto(210,0);_lineto(370,(.2857*ypix));
    _moveto(410,(.2857*ypix));_lineto(570,0);
    _settextcolor(12);_settextposition(4,1);_outtext("C");
    _settextposition(9,12);_outtext("x");_settextposition(9,37);
    _outtext("x");_settextposition(9,62);_outtext("x");_settextcolor(10);
    _settextposition(2,12);_outtext("(a)");_settextposition(2,37);
    _outtext("(b)");_settextposition(2,62);_outtext("(c)");
    _settextcolor(11);_settextposition(12,1);
    _outtext("Select a b or c as initial agent distributions");
    while(!kbhit());cres=getch();cres=tolower(cres);
    _settextposition(14,1);_outtext(
    "Input average agent loading (mg/g) ");
    scanf("%f",&coav);_settextposition(15,1);
    _outtext("Select (E)vaporation or (P)yrolysis as release mode ");
    while(!kbhit());mode=getch();mode=tolower(mode);
    return(0);
}

plot ()
{
```

```
   int z;
   _clearscreen (_GCLEARSCREEN);if(_setvideomode(_ERESCOLOR)==0)
   {ypix=200;if(_setvideomode (_HRES16COLOR)==0)
   {_setvideomode(_HRESBW);}} _setcolor(11);_setcolor(11);
   _moveto (0,3);_lineto (0,0);_moveto (0,3);_lineto (7,3);
   buffer =(char far *)malloc((unsigned int)
        _imagesize(0,0,1,1));
   _getimage(0,0,10,10,buffer);
replot: _clearscreen (_GCLEARSCREEN);if(_setvideomode(_ERESCOLOR)==0)
   {ypix=200;if(_setvideomode (_HRES16COLOR)==0)
   {_setvideomode(_HRESBW);}}_setcolor(11);_settextcolor(11);
   printf("Select variables to be plotted");_settextposition(2,1);
   _outtext ("Plot  (a) Temperature   (b) Conc. of adsorbed phase
   (c) Gas phase conc. ");
   _settextposition(3,1);_outtext(" (d) Fraction of agent left ?");
   while(!kbhit());cres=getch();cres=tolower(cres);
   if(cres=='d'){presp[0]='t';}
   if(cres!='d'){_settextposition(4,1);
   _outtext ("(T)ime or (D)epth Profile ?");
   while (!kbhit());presp[0]=getch();presp[0]=tolower(presp[0]);}
   switch (presp[0]){
   case 't':
        if(cres=='d'){break;}
        _settextposition(4,1);_outtext("Available Depths =");
        for (j=0;j<=ST;j++){printf ("%4.2f  ",j*1*delx/2.54);}
        rcoord=_gettextposition();rcoord.row +=(ST/12)+2;
        _settextposition(rcoord.row,1);
        _outtext("Number of Depths to plot = ? ");
        scanf ("%d",&NP);
        for (i=1;i<=NP;i++){printf ("%s %d %s","Depth #",i," = ");
        scanf ("%f",&pl[i]);}
        break;
   case 'd':
        _settextposition(4,1);_outtext("Available Times =");
        for (j=0;j<=nz;j++){printf ("%4.2f  ",j*delt);}
        rcoord=_gettextposition();rcoord.row +=(nz/12)+2;
        _settextposition(rcoord.row,1);
        _outtext("Number of Times to plot = ? ");
        scanf ("%d",&NP);
        for (i=1;i<=NP;i++){printf ("%s %d %s","Time #",i," = ");
        scanf ("%f",&pl[i]);}
        break;
        }
   _clearscreen (_GCLEARSCREEN);_setcolor(11);
   _rectangle (_GBORDER,110,1,639,.8*ypix);ovr();_settextcolor(11);
    _settextposition(24,1);_outtext("(p)lotter (r)eplot prn(s)crn
    prin(t) (m)enu or (e)xit ?");
    _settextcolor(12);
    switch (cres){
    case 'a':
        tt=(tg-273)*1.8+32;
        for (yp=0;yp<=.95*tt;yp+=tt/10){
```

```
     _settextposition (20.3*(1-yp/tt),9);sprintf(buf1,"%4.0f",yp);
     _outtext(buf1);}_settextcolor(14);_settextposition (10,1);
     _outtext ("Temp, F");
     break;
case 'b':
     tt=4*coav;
     for(yp=0;yp<=.95*tt;yp+=tt/10){
     _settextposition(20.3*(1-yp/tt),10);sprintf(buf1,"%2.1f",yp);
     _outtext(buf1);}_settextcolor(14);_settextposition(10,1);
     _outtext("C(ad),");
     _settextposition(11,1);_outtext("mg/g");
     break;
case 'c':
     tt=4*coav;
     for (yp=0;yp<=.95*tt;yp+=tt/10){
     _settextposition (20.3*(1-yp/tt),10);
     sprintf(buf1,"%2.1f",yp);_outtext(buf1);}
     _settextcolor(14);_settextposition(10,1);_outtext("C(g),");
     _settextposition(11,1);_outtext("mg/L");
     break;
case 'd':
     for (yp=0;yp<=.95;yp+=.1){
     _settextposition(20.3*(1-yp),10);sprintf(buf1,"%1.2f",yp);
     _outtext(buf1);}
     _settextcolor(14);_settextposition(10,1);_outtext("Fraction");
     _settextposition(11,1);_outtext("of agent");
     _settextposition(12,1);_outtext("left");
     break;
}
switch (presp[0]){
case 't':
     dth=nz*delt ;_settextcolor(12);
     for (tt=0;tt<=1;tt+=.1){_settextposition (22,14+65*tt);
     sprintf(buf1,"%2.1f",tt*dth);_outtext(buf1);}
     _settextcolor(14);_settextposition (23,40);
     _outtext ("Time, hours");_setcolor(14);
     switch(cres){
     case 'd':
          _moveto(110,0);
          for(j=1;j<=nz;j++){yp=fnpl();
          tt=110+(float)529*j/nz;_lineto(tt,yp);}
          break;
     default:
          _settextposition(2,75);_outtext("Depth");
          for (jj=1;jj<=NP;jj++){
          i=(int) (.1+pl[jj]/(l*delx/2.54));j=0;
          _moveto (110,fnpl());_setcolor(15-jj);
          _settextcolor(15-jj);sprintf
          (buf1,"%2.2f",i*(l*delx/2.54));
          _settextposition (jj+2,75);_outtext(buf1);
          for (j=1;j<=nz;j++){yp=fnpl();
          tt=110+(float)529*j/nz;_lineto (tt,yp);}}
```

```
               break;
          }
          for (i = 163; i<=640; i += 53){
          _putimage (i,(.8*ypix-3),buffer,_GOR);}
          tt=(tg-273)*1.8+32;
          for (yp=0;yp<=.95*tt;yp+=tt/10){z=(.8*ypix-3)*(1-yp/tt);
           _putimage (110,z,buffer,_GOR);}
          break;
     case 'd':
           dth=1/2.54;_settextcolor(12);
           for (tt=0;tt<=.99;tt+=(.1)){_settextposition (22,14+65*tt);
           sprintf(buf1,"%2.2f",tt*dth);_outtext(buf1);}
          _settextcolor(14);_settextposition (23,37);
          _outtext ("Distance, inches");_setcolor(14);
          _settextposition(2,75);_outtext("Times");
          for (jj=1;jj<=NP;jj++){j=(int) (.1+pl[jj]/delt);i=0;
          _moveto (110,fnpl()); _setcolor(15-jj);_settextcolor(15-jj);
          sprintf(buf1,"%2.2f",j*delt);
          _settextposition(jj+2,75);_outtext(buf1);
          for (i=1;i<=ST;i++){yp=fnpl();
          tt=110 +(float)529*i/ST;_lineto (tt,yp);}}
          for (i = 163; i<=640; i += 53){
          _putimage (i,(.8*ypix-3),buffer,_GOR);}
          tt=(tg-273)*1.8+32;
          for (yp=0;yp<=.95*tt;yp+=tt/10){
          z=(.8*ypix-3)*(1-yp/tt);_putimage (110,z,buffer,_GOR);}
          break;
     }

out: while(!kbhit());resp[0]=getch();resp[0]=tolower(resp[0]);
     switch (resp[0]){
     case 's':
          prnscrn();
          goto out;
          break;
     case 'p':
          hp();
          goto out;
          break;
     case 't':
          print();
          goto out;
          break;
     case 'r':
           _setvideomode(_DEFAULTMODE);
           goto replot;
           break;
     case 'e':
          goto ret;
          break;
     case 'm':
          goto ret;
```

```
          break;
     default:
          printf ("program complete"); }
          goto replot;
ret: _setvideomode (_DEFAULTMODE);
     return (0);
}
fnpl()
{
     switch (cres){
     case 'a':
          yp = .8*ypix*(tg*(1-th[i][j])/(tg-255));
          break;
     case 'b':
          yp = .8*ypix*(1-cs[i][j]/(4*coav));
          break;
     case 'c':
          yp = .8*ypix*(1-cg[i][j]/(4*coav));
          break;
     case 'd':
          yp=0;
          for(i=0;i<=ST-1;i++){
          yp=yp+.5*(cs[i][j]+cs[i+1][j]+cg[i][j]+cg[i+1][j])/ST;}
          yp=.8*ypix*(1-yp/coav);
          break;
     }
     return("%f",yp);
}
hp()
{
     stream = fopen ("COM1","wr");
     setvbuf(stream,buf,_IOFBF,sizeof(buf));
     fprintf (stream, "\33%s",".N;19:" );
     fprintf (stream,"\33%s",".H180;;17:");
     fprintf (stream,"IN;SP0;VS25;");
     fprintf (stream,"IP1700,1520,8175,7350;SP1;");
     fprintf (stream, "PUPA1700,1520;PDPA1700,1520,1700,7350,8175,
     7350,8175,1520,1700,1520;");
     fprintf (stream, "PDPA1720,1520,1720,7350,8175,7350,8175,1540,
     1720,1540;PU;");
     fprintf (stream, "Si.15,.25;tl2,0;sp2;");hand();

     switch(cres){
     case 'a':
          fprintf (stream,"%s\3", "SP1;DIO,1;PUPA1700,3737;CP-7,3;
          LBTemperature, F");hand();
          for (d=0;d<=1.05;d+=.1){
          tt=(255.3+d*(tg-255.3))/tg;
          hy=(1520+4309*(tt-255.3/tg)/(1-255.3/tg));
          hx=(32+1.8*(tg*tt-273));
          fprintf (stream,"%s %d %4.0f ","TL2,0;DI;PUPA",1700,hy);
          fprintf (stream,"%s %4.0f\3",";YT;CP-5,0;LB",hx);hand();
```

```
          if(d<1){hy=hy+216;fprintf(stream,"%s %d %4.0f %s","TL1,0;
          PUPA",1700,hy,";YT");}}
          break;
     case 'b':
          fprintf (stream,"%s\3", "SP1;DI0,1;PUPA1700,3737;CP-14,3;
          LBConcentration in Solid, mg/g");hand();
          for (d=0;d<=1.05;d+=.1){
          hy=(1520+4309*d);
          hx=d*4*coav;
          fprintf (stream,"%s %d %4.0f ","DI;PUPA",1700,hy);
          fprintf (stream,"%s %3.1f\3",";YT;CP-5,0;LB",hx);hand();}
          break;
     case 'c':
          fprintf (stream,"%s\3", "SP1;DI0,1;PUPA1700,3737;CP-15;
          LBGas Phase Concentration, mg/L");hand();
          for (d=0;d<=1.05;d+=.1){
          hy=(1520+4309*d);
          hx=d*4*coav;
          fprintf (stream,"%s %d %4.0f ","DI;PUPA",1700,hy);
          fprintf (stream,"%s %3.1f\3",";YT;CP-5,0;LB",hx);hand();}
          break;
     case 'd':
          fprintf (stream,"%s\3", "SP1;DI0,1;PUPA1700,3737;CP-11,3;
          LBFraction of Agent Left");hand();
          for (d=0;d<=1.05;d+=.1){
          hy=1520+4309*d;hx=d;
          fprintf (stream,"%s %d %4.0f ","TL2,0;DI;PUPA",1700,hy);
          fprintf (stream,"%s %1.2f\3",";YT;CP-5,0;LB",hx);hand();
          if(d<1){hy=hy+216;fprintf(stream,"%s %d %4.0f %s","TL1,0;
          PUPA",1700,hy,";YT");}}
          break;
     }
     switch (presp[0]){
     case 't':
          for (d=0;d<=(float)1.05*n;d+=1+(int)n/10){tt=d;
          hx=(1700+6475*(float)d/n);
          fprintf (stream,"%s %4.0f %d %s %2.1f\3",
          "TL2,0;DI;PUPA",hx,1520,";XT;CP-1,-1;LB",tt);hand();}
          for (d=0;d<=(float)n;d+=.1667*(1+(int)n/10)){
          hx=1700+6475*(float)d/n;
          fprintf(stream,"%s %4.0f %d %s",
          "TL1,0;PUPA",hx,1520,";XT;");hand();}
          fprintf (stream,"%s\3","SP1;SI;PUPA4937,1520;CP-6,-3;
          LBTime, hours");hand();
          switch(cres){
          case 'd':
               fprintf(stream,"SP2;PUPA1700,5829;");
               for(j=1;j<=nz;j++){hy=fnhp();
               hx=1700+(float)6475*j/nz;
               fprintf(stream,"%s %4.0f %4.0f","PDPA",hx,hy);hand();}
               break;
          default:
```

```
          for (k=1;k<=NP;k++){i=(int)(.1+pl[k]/(l*delx/2.54));
          j=0;hy=fnhp();
          fprintf (stream,"%s %4.0f","SP2;PUPA1700",hy);
          for (d=0;d<=n/delt;d++){j=(int)d;
          hx=(1700+6475*(j*delt)/n);hy=fnhp();
          fprintf (stream,"%s %4.0f %4.0f",
          "PDPA",hx,hy);hand();}
          fprintf (stream,"%s %2.1f %s\3",
          "PU;DI;SI.10,.15;CP1,0;LB",pl[k],"in.");}
     }
     break;
case 'd':
     for (d=0;d<=1.05;d+=(2.54/l)){tt=d*l/2.54;hx=(1700+6475*d);
     fprintf (stream,"%s %4.0f %d %s %2.1f\3",
     "TL2,0;DI;PUPA",hx,1520,";XT;CP-1,-1;LB",tt);hand();}
     for (d=0;d<=1;d+=2.54/(2*l)){hx=1700+6475*d;
     fprintf(stream,"%s %4.0f %d %s",
     "TL1,0;PUPA",hx,1520,";XT;");hand();}
     fprintf (stream,"%s\3","SP1;SI;PUPA4937,1520;CP-6,-3;
     LBDepth, inches");hand();
     for (k=1;k<=NP;k++){j=(int)(.1+pl[k]/delt);i=0;hy=fnhp();
     fprintf (stream,"%s %4.0f %s %2.1f %s\3",
     "SP2;PUPA1700",hy,";DI;SI.10,.15;CP.5,-.5;LB<--",pl[k],"hr");
     hand();
     fprintf (stream,"%s %4.0f","PUPA1700",hy);
     for (i=0;i<=ST;i++){d=(float)i/ST;hx=(1700+6475*d);hy=fnhp();
     fprintf (stream,"%s %4.0f %4.0f","PDPA",hx,hy);hand();}}
}
fprintf (stream,"%s\3","SP1;PUPA4937,7350;Di;Si;CP-13,-1;
LBHotgas Desorption Process");
fprintf (stream,"%s %d\3","SP2;PUPA1700,7350;CP2,-2;Si.15,.25;
LB Number of spatial nodes = ",ST);hand();
hy=(tg-273)*1.8+32;fprintf (stream,"%s %4.0f %s\3","CP;
LB   Gas Temperature = ",hy,"F");
fprintf (stream,"%s %5.0f %s\3","CP;LB   Gas Flow Rate = ",q,"cfm");
fprintf (stream,"%s %2.1f %s\3","CP;LB   Wall Thickness =
",l/2.54,"in.");
fprintf (stream,"%s %2.0f %s %2.0f %s\3","CP;
LB   Dimensions = ",he," x ",wi,"ft.");
if (BFL[1]==11){fprintf (stream,"%s\3","CP;
LB   Both sides of wall heated");}
hx=(1.8*(ta-273)+32);if (BFL[2]==11){
fprintf(stream,"%s %4.0f %s\3","CP;
LB   Outer wall air cooled with air temperature = ",hx," F");}
hx=1.8*(ts-273)+32;if (BFL[3]==11){
fprintf (stream,"%s\3","CP;
LB   Outer wall in contact with moist soil");
fprintf (stream, "%s %2.0f %s %4.0f %s\3","CP;
LB    Water content = ",wa,"%  T = ",hx,"F");}
if (BFL[4]==11){fprintf (stream,"%s %2.0f %s\3","CP;
LB   Outer wall insulated with ",lp/2.54,"in. @k = 0.05 Btu/hr-ft-F");}
fprintf (stream, "PUPA0,0;SP0");
```

```
     fflush (stream);
     fclose;
     return(0);
}
store()
{
     stream = fopen (fname, "w");
     fprintf(stream,"%d  %f  %f  %f  %f  %f  %f  %d  %f  %f  %f  %f %f %c "
     ,ST,tg,ti,q,l,he,wi,nz,lp,ta,delx,delt,coav,cres);
     for (i=0;i<=9;i++){fprintf (stream,"%d ",BFL[i]);}
     for (j=0;j<=nz;j++){
     for (i=0;i<=ST;i++){fprintf(stream,"%f ",th[i][j]);}}
     fclose (stream);
     return(0);
}
retrieve()
{
     stream = fopen (fname, "r");
     fscanf(stream,"%d  %f  %f  %f  %f  %f  %f  %d  %f  %f  %f  %f %f %c "
     ,&ST,&tg,&ti,&q,&l,&he,&wi,&nz,&lp,&ta,&delx,&delt,&coav,&cres);
     for (i=0;i<=9;i++){fscanf (stream,"%d ",&BFL[i]);}
     for (j=0;j<=nz;j++){
     for (i=0;i<=ST;i++){fscanf(stream,"%f ",&th[i][j]);}}
     fclose (stream);BFL[7]=3;BFL[6]=11;
     output();
     return(0);
}
header()
{

     fprintf (stream, " Hotgas Desorption Process");
     switch(cres){
     case 'a':
          fprintf (stream,"\n\r%s", "Temperature, (F)");
          break;
     case 'b':
          fprintf (stream,"\n\r%s",
          "Adsorbed Phase Concentration, (mg/g)");
          break;
     case 'c':
          fprintf (stream,"\n\r%s", "Gas Phase Concentration, (mg/L)");
          break;
     }
     fprintf (stream,"\n\r%s %d", " Number of spatial nodes = ",ST);
     fprintf (stream,"\n\r%s %4.1f %s",
     "Gas Temperature = ",((tg-273)*1.8)+32,"F");
     fprintf (stream,"\n\r%s %4.1f %s", "Gas Flow Rate = ",q,"cfm");
     fprintf (stream,"\n\r%s %2.1f %s", "Wall Thickness = ",l/2.54,"in.");
     fprintf (stream,"\n\r%s %2.1f %s %2.1f %s",
     "Dimensions = ",he," x ",wi,"ft.");
     if( BFL[1]==11) { fprintf (stream,"\n\r%s",
     "Both sides of wall heated");}
```

```
    if( BFL[2]==11) { fprintf (stream,"\n\r%s %4.1f %s",
    "Outer wall air cooled with air temperature = ",(ta-273)*1.8+32," F");}
    if( BFL[3]==11) { fprintf (stream,"\n\r%s",
    "Outer wall in contact with moist soil");}
    if( BFL[4]==11) { fprintf (stream,"\n\r%s %2.1f %s %1.3f %s",
    "Outer wall insulated with ",lp/2.54,"in. of insulation
    having a conductivity of ",kp*241," Btu/hr-ft-F");}
    return(0);
}
print()
{
    stream = fopen ("LPT1","w")
    header();
    fprintf (stream,"\n%s", "t,hr                    Distance,in.");
    fprintf (stream, "\n\r\017%s","          ");
     for (i=0;i<=ST;i++){fprintf (stream, "%2.1f\t",1*i*delx/2.54);}
     fprintf (stream,"\n\r");
    for (i=0;i<=8*(ST+2);i++){fprintf (stream, "_");}
    fprintf (stream,"\n\r");
    switch(cres){
    case 'a':
         for (j=0;j<=nz;j++){fprintf
         (stream,"\n\r%2.2f %s\t",j*delt,"|");
         for(i=0;i<=ST;i++){fprintf
         (stream,"%4.1f\t",1.8*(tg*th[i][j]-273)+32);}}
         break;
    case 'b':
         for (j=0;j<=nz;j++){fprintf
         (stream,"\n\r%2.2f %s\t",j*delt,"|");
         for (i=0;i<=ST;i++){fprintf(stream,"%4.1f\t",cs[i][j]);}}
         break;
    case 'c':
         for (j=0;j<=nz;j++){fprintf
         (stream,"\n\r%2.2f %s\t",j*delt,"|");
         for (i=0;i<=ST;i++){fprintf(stream,"%4.1f\t",cg[i][j]);}}
         break;
    }
    fprintf (stream,"\r");for (i=1;i<=8*(ST+2);i++){
    fprintf (stream, "_");}
    fprintf (stream,"\022\014");
    fflush (stream);
    return (0);
}
hand()
{
    int jk=0;
    while (jk<=10000){jk++;}
    return (0);
}

ovr()
{
```

```
    _settextcolor(10);_settextposition (2,16);
    if (BFL[1]==11){_outtext("Both Sides Heated");}
    if (BFL[2]==11){_outtext("One Side Air-Cooled");}
    if (BFL[3]==11){_outtext("One Side Moist Soil");}
    if (BFL[4]==11){_outtext("One Side Insulated");}
    _settextposition(3,16);
    if (BFL[8]==11){_outtext("Agent = HD");}
    if (BFL[9]==11){_outtext("Agent = GB");}
    if (BFL[0]==11){_outtext("Agent = VX");}
    sprintf(bufl,"%3.1f %s",coav," mg/g");
    _settextposition(4,16);_outtext("Ave. Init. Conc.= ");_outtext(bufl);
    return(0);
}


fnhp()
{
    switch(cres){
    case 'a':
         hy=(1520+4309 (th[i][j]-255.3/tg)/(1-255.3/tg));
         break;
    case 'b':
         hy = 1520+4309*(cs[i][j]/(4*coav));
         break;
    case 'c':
         hy = 1520+4309*cg[i][j]/(4*coav);
         break;
    case 'd':
         hy=0;
         for(i=0;i<=ST-1;i++){
         hy=hy+.5*(cs[i][j]+cs[i+1][j]+cg[i][j]+cg[i+1][j])/ST;}
         hy=1520+4309*(hy/coav);
         break;

    }
    return(hy);
}
prnscrn()
{
    stream = fopen("LPT1","w");fprintf(stream,"\v\v\033\101\007");
    if (BFL[5]==3){fprintf(stream,"\033\062");}
    for(j=0;j<=(int)(.914*ypix);j+=7){for(i=0;i<=639;i++){tm[i]=0;
    for(k=0;k<=7;k++){m[k]=_getpixel(i,k+j);if(m[k]>1){m[k]=1;}
    tm[i]=tm[i]+m[k]*pow(2,(7-k));}}
    fprintf (stream,"\v\r\t\033\114\200\002");for(i=0;i<=639;i++){
    fputc(toascii(tm[i]),stream);}
    }
    fprintf(stream,"\v\v\r");
    fprintf(stream,"\033\101\011");fprintf(stream,"\033\062");
    header();fprintf(stream,"\014");
    fflush(stream);
    fcloseall;return(0);
}
```