BG9600414

INTEGRATED COMPUTER CODES FOR NUCLEAR POWER PLANT SEVERE ACCIDENT ANALYSIS

I. D. IORDANOV, Y. CH. HRISTOV
INRNE - BAS

1. INTRODUCTION

Since the accident at Three Mile Island Unit 2 (TMI-2) increased attention has been given to the details of accidents that could take place at a nuclear power plant and that could result in radioactive releases to the environment in excess of accepTable levels. Because of the scale and complexity of the relevant postulated events, it is not possible to carry out direct experimental investigations to characterise them. Instead, these complex sequences have been modelled as a set of simpler component phenomena. Experimental programmes have been undertaken to characterise the results of occurrences of possible combinations of these phenomena in postulated events. The improved understanding of the various individual phenomena has been incorporated into analytical computer codes, thus synthesising the individual phenomena into sequences of events representing hypothetical accidents.

2. MELTDOWN ACCIDENT ANALYSIS PROGRAM

The Modular Accident Analysis Program (MAAP) is a computer code which simulates light water reactor system response to accident initiation events [1]. It is prepared as a part of the IDCOR (Industry Degraded COre Rulemaking) program to investigate the physical phenomena which might occur in the event of a serious light water reactor accident leading to core damage, possible reactor pressure vessel failure and possible containment failure and depressurization.

MAAP includes models for the important phenomena which might occur in a serious light water reactor accident. There are two parallel versions of MAAP, MAAP/BWR and MAAP/PWR, for the two general light water reactor types in use. MAAP can predict the progression of hypothetical accident sequences from a set of initiating event to either a safe, sTable, coolable state or to containment failure and depressurization. MAAP treats a wide spectrum of phenomena including steam formation, core heatup, cladding oxidation and hydrogen evolution, vessel failure, corium-concrete interactions, ignition of combustible gases, fluid (water and corium) entrainment by high velocity gases, and fission product release, transport and deposition. MAAP treats all of the important engineered safety systems such as emergency core cooling, containment sprays, fan coolers, and power operated relief valves, and the auxiliary or reactor building can be modelled for sequences in which it is important. In addition, MAAP allows operator interventions and incorporates these in a very flexible manner, permitting the user to model operator behaviour in a general way. The user models the operator by specifying a set of variable values and/or events which are the operator interventions conditions. There is a large set of actions the operator can take in response to the intervention conditions.

The user may establish one or more intervention conditions by specifying limits for any of a set of key variables or by declaring any of more than 100 event flags as key events. When a key variable reaches its specified limit, or a key event flag changes status, program execution pauses and operator actions, also specified by the user, are taken. The operator actions consist of changes to event flags numbered 200 and above. In this way, MAAP is directed by a pseudooperator who uses present plant conditions to make operational decisions.

This work was partially funded by Commission of the European Communities under contract No F13-CT92-0057 - MA No 13299(5)

Periodically during a transient, MAAP writes restart files. This allows the user to make a subsequent run starting at any time covered by the original run. This provides additional flexibility for the user to simulate operator actions or to change external events. Restart data files are written at time intervals chosen by the user and whenever a program interventions occurs. MAAP can then resume execution from any of the times at which a restart file entry was written. The restart can have a new program intervention conditions, new operator actions, and even changes to the input parameter file.

MAAP has a modular structure in which separate subprograms are dedicated to specific region models and physical phenomena. This facilitates changing the code because improvements to phenomenological or region models can be made relatively small subprograms. Each MAAP version, MAAP/BWR and MAAP/PWR, consists of a main program which directs program execution through several high level subroutines. Depending on the containment/primary system design, the program calls a sequence of system and region subroutines at each time step. These subroutines call, in turn, phenomenology subprograms as required. At the lowest level, a set of property-library subprograms are available to provide physical properties.

The MAAP code uses a two stage computational procedure in which the present values of the dynamic variables describing the state of the system (often masses and internal energies) are used to calculate their rates of change. Then the rates are integrated in a separate subroutine to provide updated values of the dynamic variables. The integration technique used during the development of the MAAP is an explicit, first order, Euler integration. An alternative method, a second order Runge-Kuta integration can be selected through the parameter file. The differences are not great; a Loss-of-All-Power PWR example transient using the second order method had a running time twice as long as that of the first order method. Key procedure events (e.g., time to reactor vessel failure) were within a few percent of each other generally much closer. A typical MAAP run has time steps as small as 0.005 seconds and as large as 20 seconds. MAAP has been written to execute both in a batch mode and in an interactive mode where the user monitors the execution at a terminal. The batch and interactive modes are entirely parallel and use the same input desks (file of card images). They differ only in method and timing of entering the cards (lines) of the input desk (file).

MAAP can model an auxiliary building which receives the discharges from either the containment or the primary system. The auxiliary building model can be run simultaneous with the primary system and containment models or in a stand-alone model. In the later, the code reads an input generated in an earlier MAAP run that supplies boundary conditions to the model.

The parameter file, required by MAAP to define the reactor system, consists mainly of plant-specific data which will not change from one run to another. These are relegated to a disk file which is read by MAAP at the start of execution. Accident specific inputs, such as accident initiators and operator actions, are contained in a separate input desk which is read by MAAP during execution. The user may change parameter file entries for individual runs by specifying those changes in the input desk. Thus, the parameter file for a specific plant needs to be prepared only once and temporary changes to any parameter entries can be made at execution time without manipulating the parameter file itself. An accident summary is printed at the end of a run and provides a chronology of significant events such as engineered system responses and operator actions.

MAAP has been issued in several versions since about 1982. The first version receiving wide distribution was MAAP 2.0B.

3. ICARE COMPUTER PROGRAM

The ICARE computer code is being developed at the Institute for Protection and Nuclear Safety of CEA (France). This analytical work [2] is a part of the Severe Fuel Damage program [3] conducted in France in order to clearly understand the main physical mechanism occurring during the core degradation of PWR.

ICARE models the progression of reactor core damage including: core heat-up, loss of geometry by melting and embrittlement, relocation of materials, crucible formation and fission products release. It works as a stand-alone code to describe both experimental facilities and reactor core in severe situations; in case of reactor calculations, ICARE uses boundary conditions provided by the French Reactor Coolant System CATHARE code [4].

During a severe accident on PWR, hydrogen and heat are produced by oxidation of zircaloy in fuel claddings and absorber rod guide tubes as well as oxidation of stainless steel in absorber rod claddings and vessel structures. The prediction of the right amount of hydrogen and heat is of great importance for the analysis of containment integrity in case of loss of coolant from primary circuit and for the analysis of whole degradation respectively.

ICARE2 code (version 2) uses a new data organisation technique allowing especially a dynamic management of the computer memory, called SIGAL (in French: Structure Informatique d'Accueil et de Gestion de Logiciels).

This technique sets up several functions concerning in the same time developers and users. Each function is associated with either a library or a code:

- a dynamic memory management library to optimise the use of central computer's memory;
- a memory organisation library to handle and modify SIGAL data bases stored in the central memory;
- a data reader;
- a data checker;
- an analyser program to interpret special user's instructions in order to perform all manipulations of SIGAL data bases;
- a self governing graphic program called T.I.C. (in French: Trace Interactif de Courbes). A general overview of tne ICARE2 architecture is given in figure 1.

The modular structure of the code allows many types of topologies to be described: PHEBUS SFD and FP, PFB, CORA, PWR core.

The main program calls in series different modules which are associated to specific physical tasks: fluid dynamics, thermics, flow down and relocation, chemical reactions, ect. The numerical scheme of the main physical modules is as implicit in time as possible, but the coupling between them is generally explicit. Therefore, the energetic error induced by coupling between hydraulic and thermic modules controls essentially the global time step management.

The data exchanges between each of these modules are performed through SIGAL data base. This logic, combined with the modular structure of the code, makes it possible new modules to be developed independently and after self-governing tests to be easily introduced in ICARE2.

ICARE2 is composed of about 30 000 statements entirely written in FORTRAN 77 and, therefore, can be implemented under any system compatible with the FORTRAN 77 language. However, the SIGAL software must be adapted according to specific logic of the different hardware (IBM, CRAY, VAX, SUN, ect.).

ICARE2 is implemented in France on IBM, CRAY and SUN computers. It has been released to SPAIN and to ISPRA.

The ICARE2 input file is presented in the form of independent blockdata, each of them defining various types of quantities such as metallic structure geometry, boundary conditions, physical exchanges, computation options (time step, graphic or restart storages, ect.). There is no compulsory order for blockdata description in the input file and no predefined relationships between each blockdata. The latter point, which gives large flexibility to introduce data, allows users to perform very easily sensitivity studies. The input data acquisition is performed by a data reader which is able to read structured user's data, to check their syntax and to store automatically them in a SIGAL data base. The data checker compares user's data with a predefined data base, called control file. This data base consists of a file created by developers and containing the correct data form as well as the physical or logical elementary rules they have to respect (such as data compulsory aspects, blockdata variable types, physical pertinence of boundary conditions, number of terms in Tables, increasing or decreasing order of these terms, ect.).

4. SOURCE TERM CODE PACKAGE

The Source Term Code Package (STCP) is a set of computer codes which allows analyses of nuclear reactor accidents to produce predictions of fission product release to the environment as a function of reactor design and specifications of the assumed accidents [5]. In figure 2 is shown the flow diagram of the STPC. The overall thermal-hydraulics is provided by the MARCH-3. Release of fission products and aerosols during core-concrete interactions is predicted with the VANESA code. Detailed thermal-hydraulics and fission product transport in the reactor coolant system are provided by the TRAP-MELT3. Finally, fission product transport in the containment is predicted by NAUA code.

MARCH-3 describes the behaviour of the reactor during a severe accident and, with approximately 185 routines, is the largest of the codes involved in the STCP. The MARCH-3 code evaluates the following phenomena:

1. Heatup of the reactor coolant inventory and pressure rise or safety valve settings with subsequent boiloff;
2. Initial blowdown of the of the coolant from the reactor coolant system;
3. Generation and transport of heat within the core, including boiloff of water from the reactor vessel;
4. Heatup of the fuel following core uncovery, including the effects of metal-water reactions;
5. Melting and slumping of the fuel onto the lower core support structures and into the vessel bottom head;
6. Interaction of the core debris with residual water in the reactor vessel;
7. Interaction of the core debris with the reactor vessel bottom head;
8. Interaction of the core debris with the water in the reactor cavity;
9. Attack of the concrete baseman by the core and structural debris;
10. Relocation of the decay heat source as fission products are released from the fuel and transported to the containment;
11. Mass and energy additions to the containment associated with all foregoing phenomena and their effects on the containment temperature, pressure, and steam condensation;

12. Effects of burning of hydrogen and carbon monoxide on the containment pressure and temperature;
13. Leakage of gases to the environment.

For MARCH-3, input is required to describe plant, to select modelling options, to describe control parameters for safety system operation.

The MARCH-3 output provides a wide variety of information on the thermal and hydraulic conditions in the reactor system, as well as the containment, throughout the course of the accident sequence being analysed. Among the key outputs are the timing of containment events, status of the core, and pressure and temperature in the containment. Additionally, extensive detail is available on core and structure compositions, distribution of water with in the entire system, mass and energy balance audits, ect.

The VANESA model is a mechanistic description of the aerosol generation and fission product release during core debris interactions with concrete. The model predicts the mass, composition, and meanparticle size of radioactive and non-radioactive materials liberated as vapors or particles during interactions. The model indicates whether mass release is by vaporisation processes or mechanical processes. VANESA takes the CORCON output of the MARCH-3 and models the reduction of the H_2O and CO_2 to H_2 and CO, as well as the loss of other materials from the pool as aerosols. The gas release from the core-concrete interaction is an important part of most accident sequences because it provides a severe load on the containment at the same time that a large amount of airborne material is being produced. The rate of deposition of vapors and aerosol moving through structures of reactor coolant system (e.g., the upper plenum or the pressurizer) can be calculated given the temperature of these structures and the flow rate, composition, and temperature of the gas. The MERGE code provides the required flow rates, gas conditions, and temperatures. TRAP-MELT2 code, which runs as a subroutine in TRAP-MELT3, handles ten species of materials, including noble gases in the STCP. The noble gases are not retained in the reactor coolant system and are thus considered in TRAP-MELT only for decay heat calculations and for bookkeeping purposes. The three species, CsI, CsOH and Te account for all the volatile fission products of interest in a TRAP-MELT calculations. These three forms are treated as vapors as they leave the core: they can condense an walls and aerosol particles, evaporate from were they have condensed, or become attached to wall surface by some chemical or physical mechanism (sorbtion).

Some portion of the aerosol produced in the vessel during core heatup and those ex-vessel aerosols produced during the core concrete interaction eventually arrive in the reactor containment structure. Natural processes of agglomeration and deposition lead to retention of aerosols within the containment. In addition, some containment structures are equipped with water sprays, ice condensers, or water suppression pools, which cause further retention of aerosol and fission products. The NAUA-MOD4 computer code is being used for analysing these effects.

5. EXPERIENCE OF STCP IMPLEMENTATION

5.1. VAX SYSTEM IMPLEMENTATION

STCP Mod.1 was implemented on a VAX/VMS computer system. The package was run on a 32-bit supper-minicomputer VAX 11/150 with VAX/VMS V4.7 operating system. Using the 192 version of the programme MARCH3, the TMLB sample problem for Zion Unit 1 NPP (PWR with a large, dry containment) was calculated successfully. However, when analysing the TMLB scenario for a VVER-440 (V 213) NPP, MARCH3 failed to calculate the entire sequence due to loss of accuracy. In order to remove this obstacle, a DOUBLE

PRECISION VAX-version MARCH3D was created. In this version, the CPUSSEC function was recoded to return the elpased real time through substitution the CDC intrinsic function SECOND by the VAX/VMS intrinsic function SECNDS. The last one returns the time of the day. Using the created VAX-version MARCH3D, the TMLB scenarios for Zion and VVER-440 NPP were also calculated.

Table 1 shows the chronology of the main accidents events during the Zion TMLB accidents calculated by MARCH3 and MARCH3D an a VAX 11/750 computer. For a comparison, the sample output results for the same scenario, provided by the Battelle Columbus Laboratories, are also given in Table 1.

In order to estimate the influence of the VAX 11/750 word length representation on the results obtained, the relative deviations of some Zion TMLB MARCH3- and MARCH3D-calculated parameters from the sample values were calculated. The relative deviation of the parameters, dev_i, was calculated according to:

$$dev_i = \frac{P_{calc,i} - P_{sample,i}}{P_{sample,i}} 100, \%, \qquad (1)$$

where

$P_{calc,i}$ is MARCH3- or MARCH3D-calculated value of the parameter at the moment of accident event i;
$P_{sample,i}$ - sample value of the parameter at the moment of accident event i.

Figures 3 and 4 show the relative deviations of the Zion TMLB MARCH3- and MARC3D-calculated reactor containment pressure and debris temperature as sample accident time functions.

TABLE 1 - CHRONOLOGY OF THE MARCH3- AND MARCH3D-CALCULATED ZION TMLB ACCIDENT EVENTS ON A VAX 11/750 COMPUTER

No	ACCIDENT EVENT	TIME (min)		
		SAMPLE	MARCH3D	MARCH3
1	Accident initiation	0.00	0.00	0.00
2	Steam generator dryout	93.30	93.30	95.65
3	Core uncovery	126.63	126.63	126.23
4	Start of core melting	149.13	149.13	148.98
5	Core slumping	168.13	168.13	167.48
6	Bottom head heatup	169.88	169.88	169.23
7	Bottom head failure	178.57	178.57	177.92
8	Start of water-debris interaction	178.57	178.57	177.92
9	Containment failure	279.61	179.61	178.96
10	Reactor cavity dryout	242.18	242.04	239.42
11	Start of debris-concrete interaction	307.18	304.04	305.42
12	Normal exit	907.18	907.04	905.42

5.2 IBM SYSTEM IMPLEMENTATION

STCP Mod.1 was also implemented on an IBM computer system. The package was run on a 32-bit IBM 3081 computer. The software included MVS/XA operating system, ISPF utility an VS-FORTRAN V-1.3 compiler. During the VS-FORTRAN compilation of the origin MARCH3 (V 192) code, some warnings (but not syntax errors) were encountered. Zion TMLB sample problem wad run on the machine and it was noted that the resultant output contained some differences from the sample output results. Consequently, the second attempt was to recoded MARCH3 into DOUBLE PRECISION. Repetitive checking of source lines was necessary during the conversion. As a result of these modifications, the full functionality of the code was retained. Using the created IBM DOUBLE PRECISION version MARCH3D, the TMLB scenarios for Zion and VVER-440 (V 213) NPPs were calculated.

Table 2 shows the chronology of the main accident events during the Zion TMLB accident sequence calculated by the MARCH3D DOUBLE PRECISION versions on a VAX 11/750 and on a IBM 3081 computers. For a comparison, the sample problem output results for the same scenario, are also given in Table 2. The figures 7 to 10 show a very good coincidence between the results from calculation and the sample problem output results.

In order to estimate the influence of the computer used on the results obtained, the relative deviations of some Zion TMLB VAX 11-750- and IBM 3081-calculated parameters, using the created MARCH3D DOUBLE PRECISION versions, from the sample output values were calculated according to Equation (1).

Figures 5 and 6 show the relative deviations of the Zion TMLB VAX 11/750- and IBM 3081-calculated reactor containment pressure and debris temperature as a function of the sample accident time.

TABLE 2 - CHRONOLOGY OF THE MARCH3- AND MARCH3D-CALCULATED ZION TMLB ACCIDENT EVENTS ON A VAX 11/750 COMPUTER

No	ACCIDENT EVENT	TIME (min)		
		SAMPLE	VAX	IBM
1	Accident initiation	0.00	0.00	0.00
2	Steam generator dryout	93.30	93.30	93.30
3	Core uncovery	126.63	126.63	126.63
4	Start of core melting	149.13	149.13	148.13
5	Core slumping	168.13	168.13	168.13
6	Bottom head heatup	169.88	169.88	169.88
7	Bottom head failure	178.57	178.57	178.57
8	Start of water-debris interaction	179.57	178.57	179.57
9	Containment failure	179.61	179.61	179.61
10	Reactor cavity dryout	242.18	242.04	240.30
11	Start of debris-concrete interaction	307.18	307.04	306.30
12	Normal exit	907.18	907.04	906.30

REFERENCES

1. MAAP User's manual, volume I and II, Fauske & Associates Inc, 1982.
2. USNRC, Nuclear Power Plant Severe Accident Research Plan, G. P. Marino Ed., NUREG -09000, Rev. 1, April, 1985.
3. A. Porrachia et al., ICARE: A Computer Code for Severe Fuel Damage Analysis Development of Models for the First Version ICARE1, Int. Conf. - Th. React. Safety, Avignon, France, October, 1988.
4. G. Gonnier et al., In Pile Investigation at Phebus Facility on the Behaviour of PWR Type Fuel Bundles in Severo Accident Conditions beyond the Desing Criteria, Int. Conf. -Th. React. Safety, Avignon, France, October, 1988.
5. J. A. Gieske et al., Source Term Code Package - a User's Guide (Mod.1), NUREG/CR-4587, Washington, July, 1986.

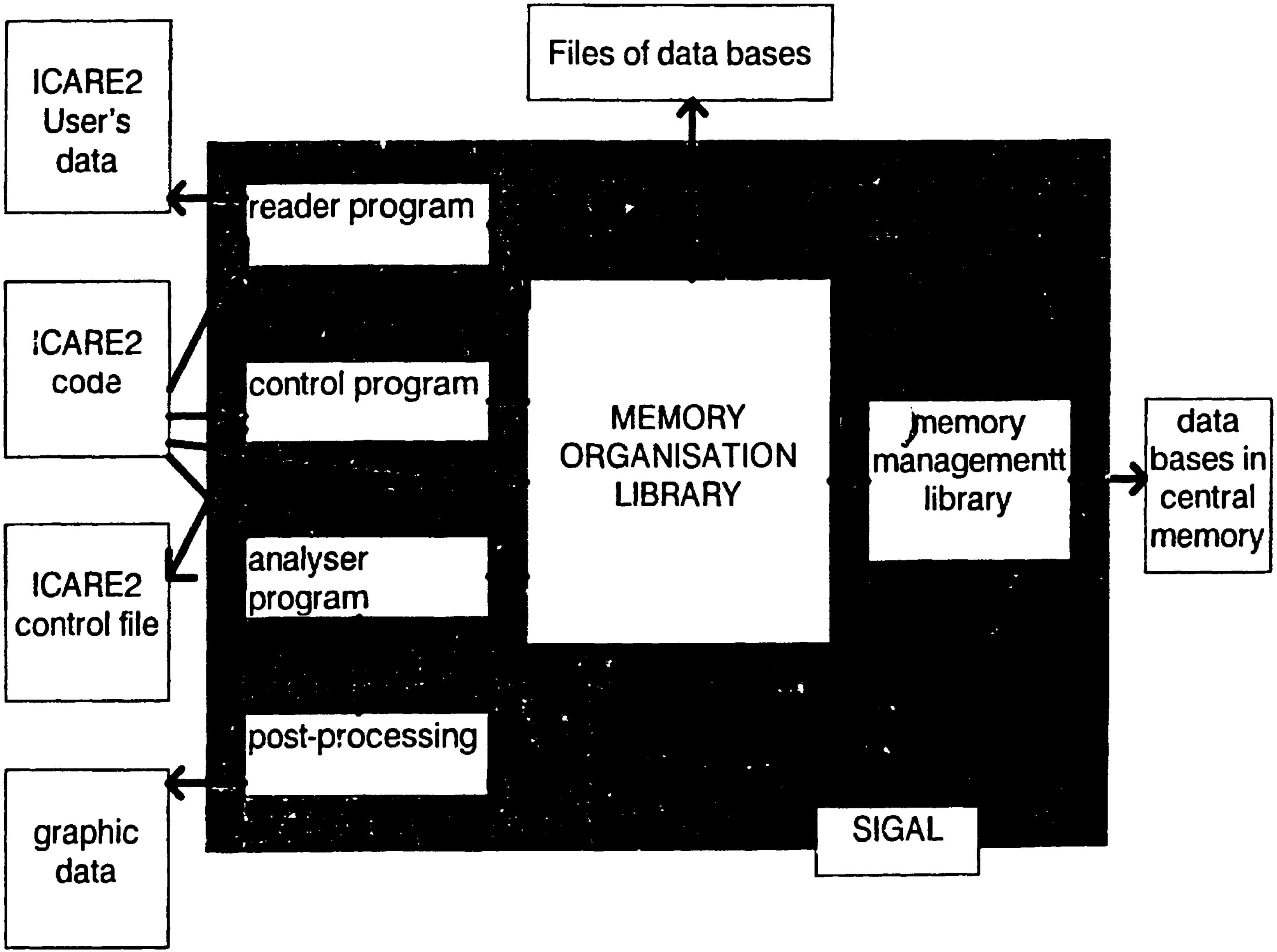
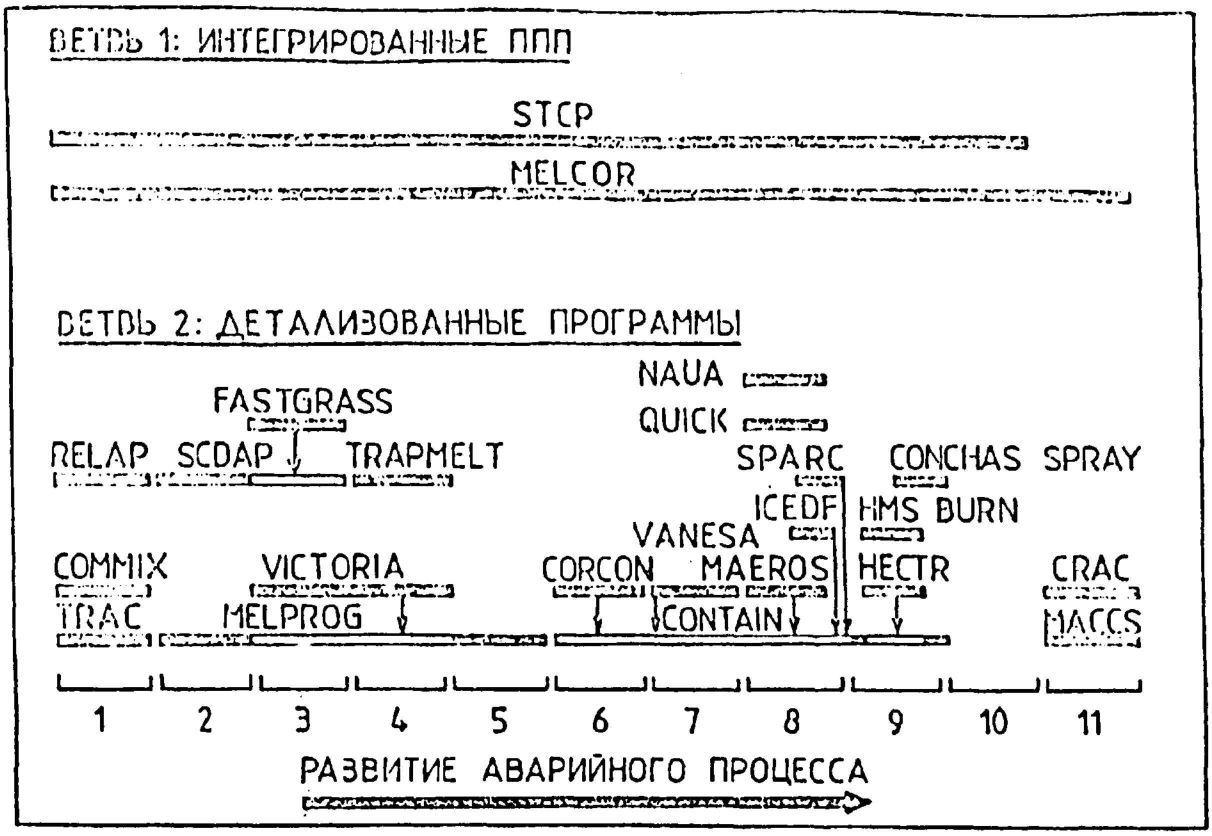
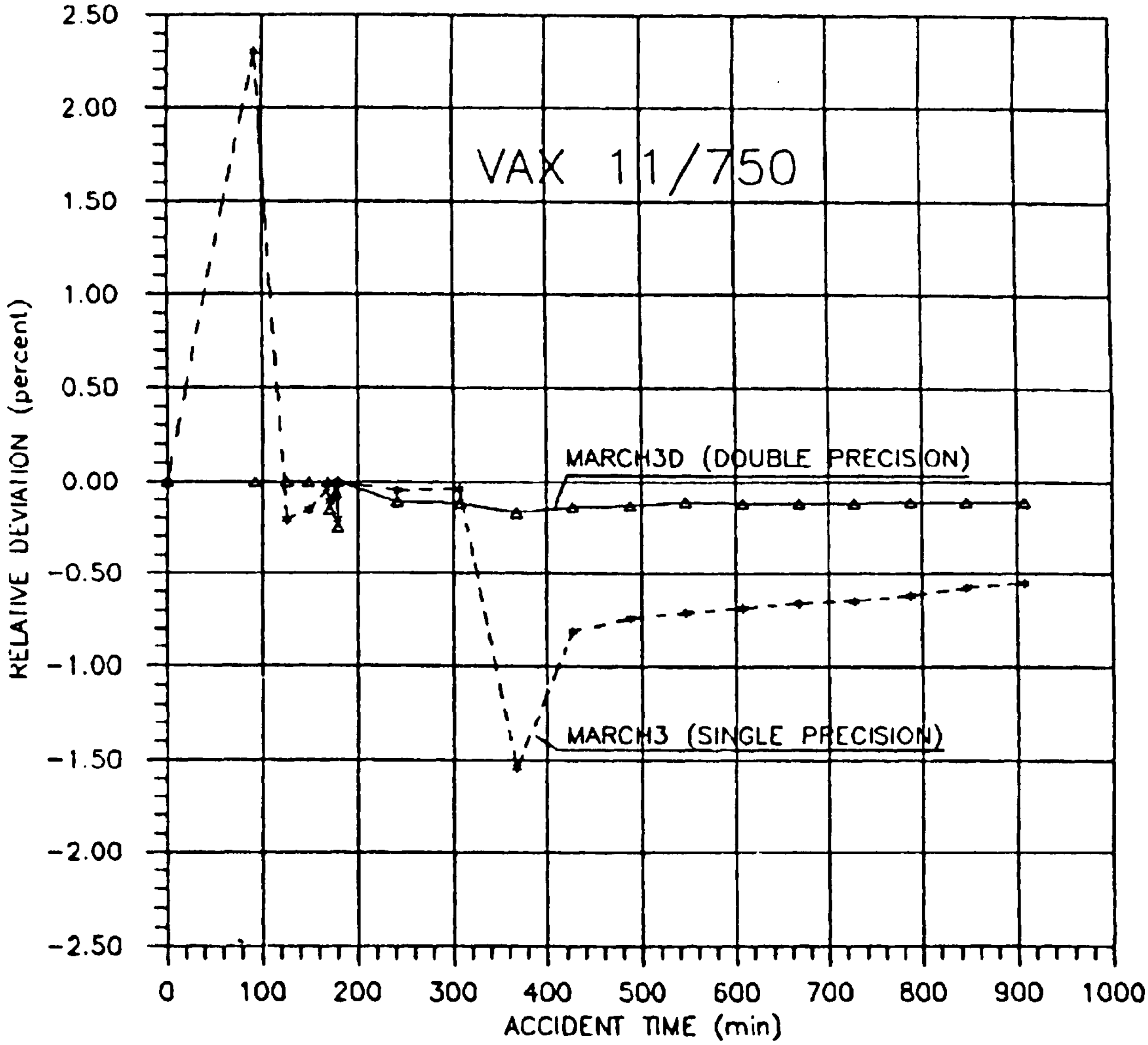


Fig. 1 - A general overview of the ICARE2 architecture



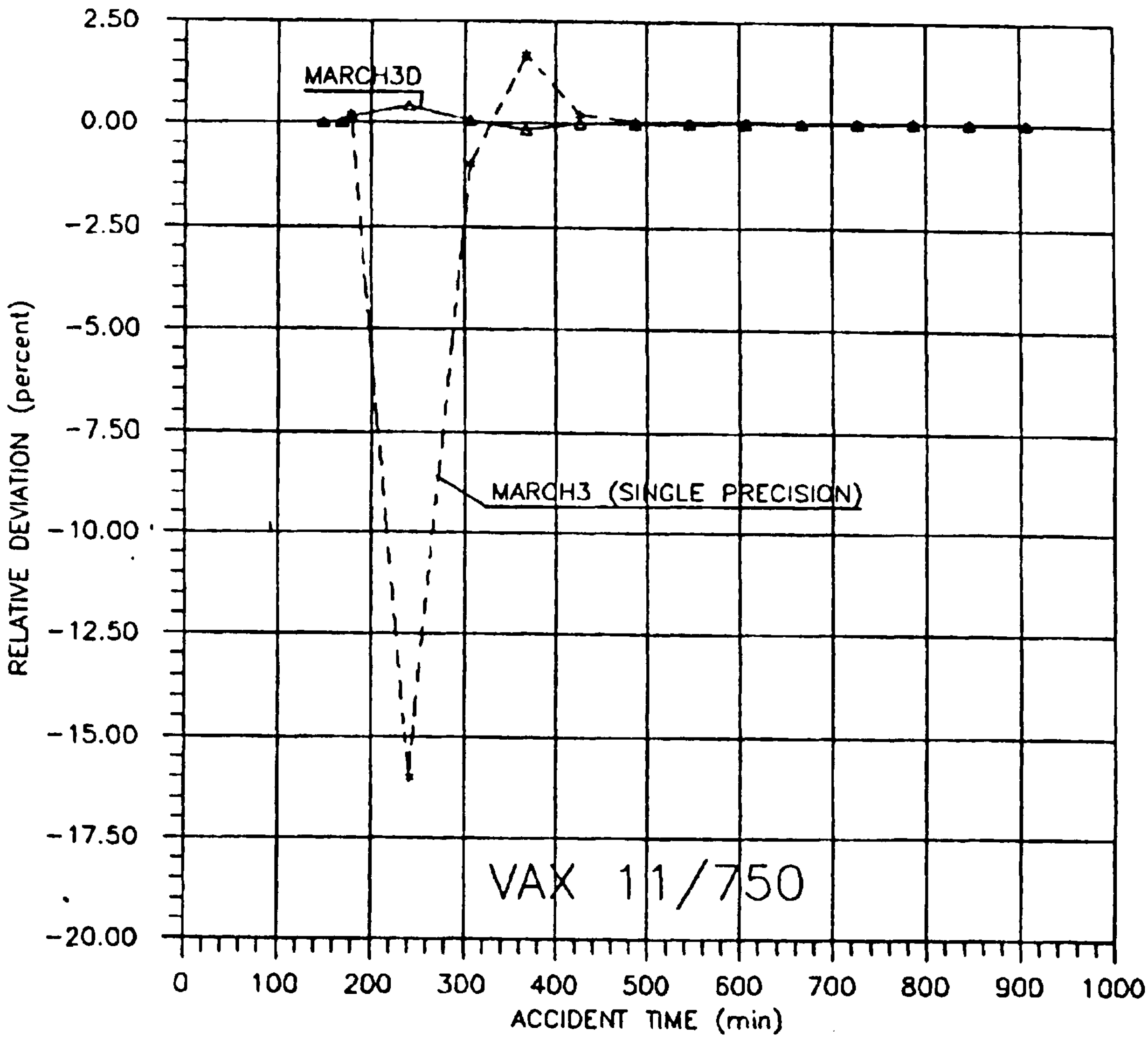
1 - Thermal Hydraulics
2 - Core Melting
3 - Release Frome Fuel
4 - Transport in Reactor Coolant System
5 - Vessel Failure
6 - Concrete Interactions
7 - Release from Debris
8 - Transport in Containment
9 - Containment Loads
10- Containment Performance
11- Off Site Consequences

Fig. 2 - Flow diagram of the STCP



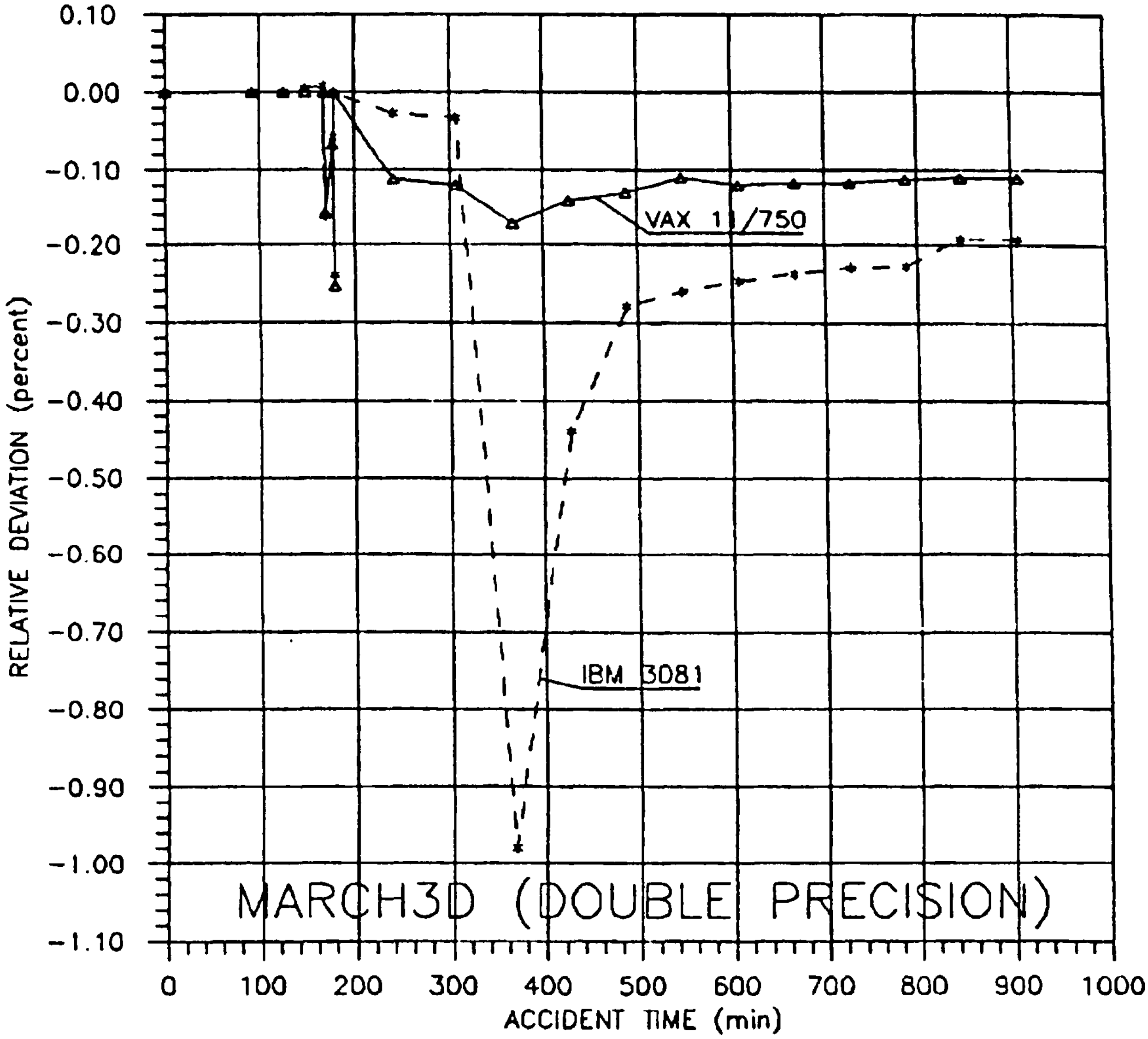
RELATIVE DEVIATION OF THE MARCH3- AND MARCH3D-
CALCULATED ZION TMLB CONTAINMENT PRESSURE
ON A VAX 11/750 FROM THE SAMPLE VALUES

Fig. 3 - Containment pressure



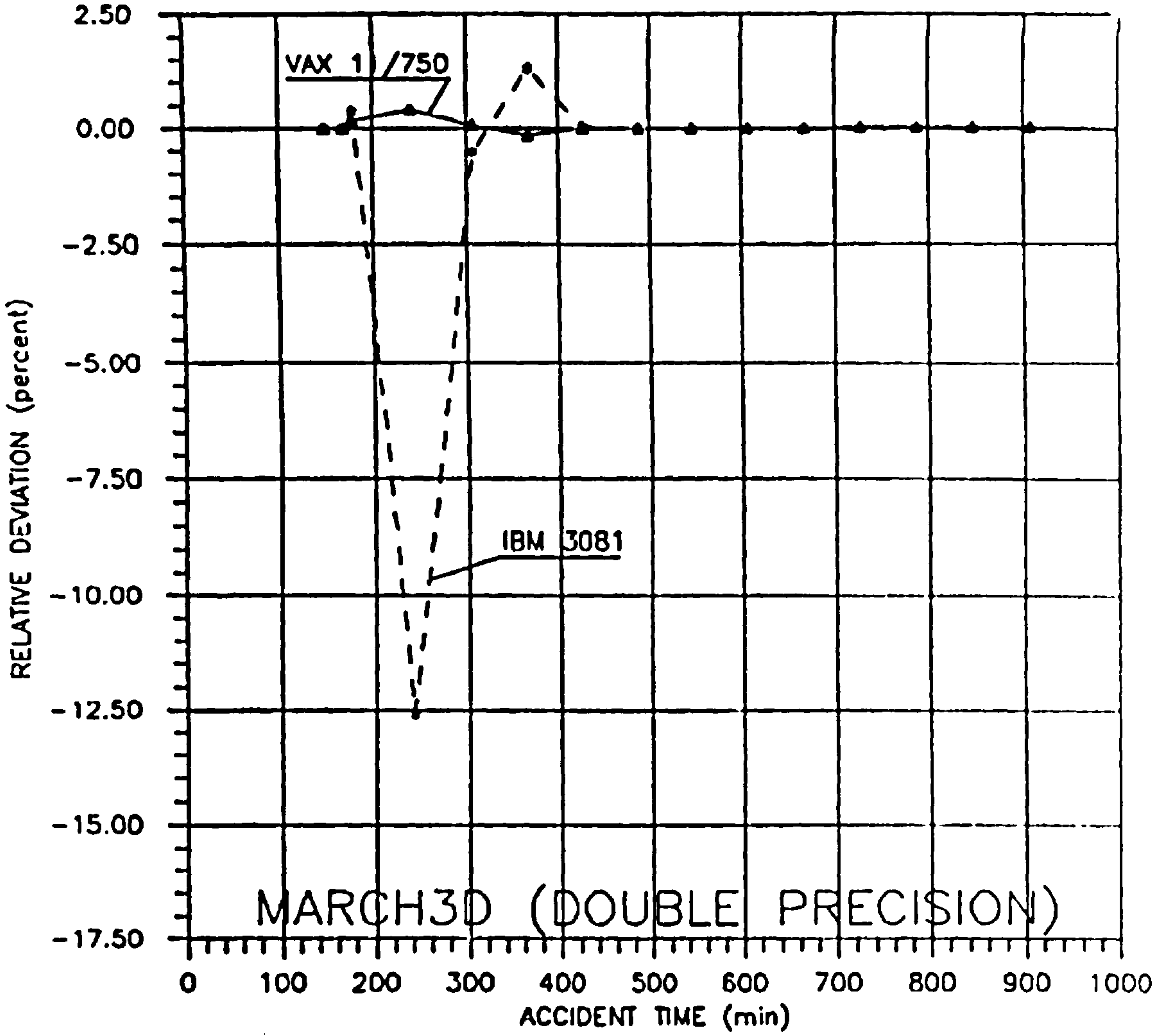
RELATIVE DEVIATION OF THE MARCH3- AND MARCH3D-
CALCULATED ZION TMLB DEBRIS TEMPERATURE
ON A VAX 11/750 FROM THE SAMPLE VALUES

Fig. 4 - Debris temperature



RELATIVE DEVIATION OF THE MARCH3D-CALCULATED ZION TMLB CONTAINMENT PRESSURE ON VAX 11/750 AND IBM 3081 COMPUTERS FROM THE SAMPLE VALUES

Fig. 5 - Containment pressure



4 RELATIVE DEVIATION OF THE MARCH3D-CALCULATED ZION TMLB DEBRIS TEMPERATURE ON VAX 11/750 AND IBM 3081 COMPUTERS FROM THE SAMPLE VALUES

Fig. 6 - Debris temperature

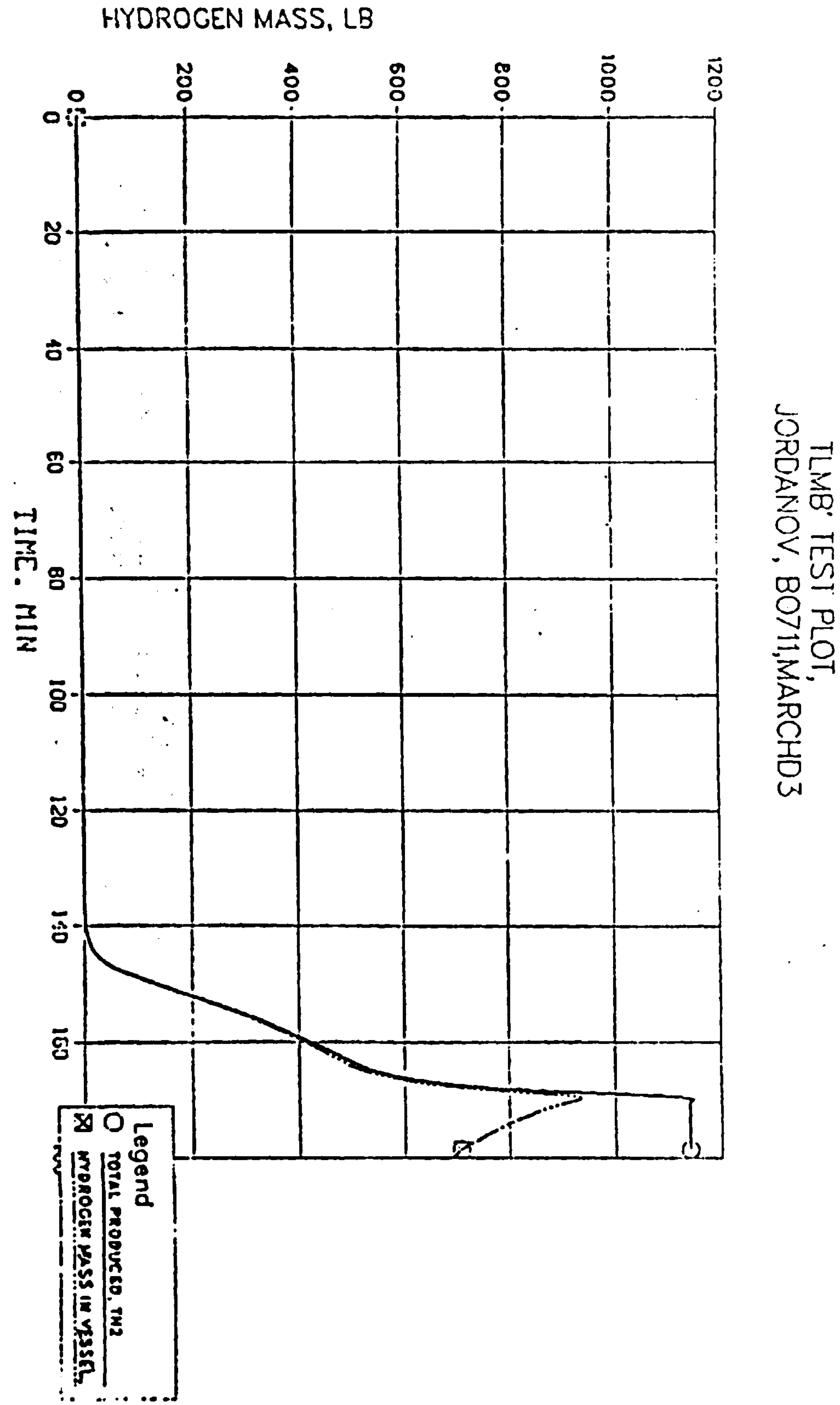


Fig. 7 - Hydrogen Mass, LB

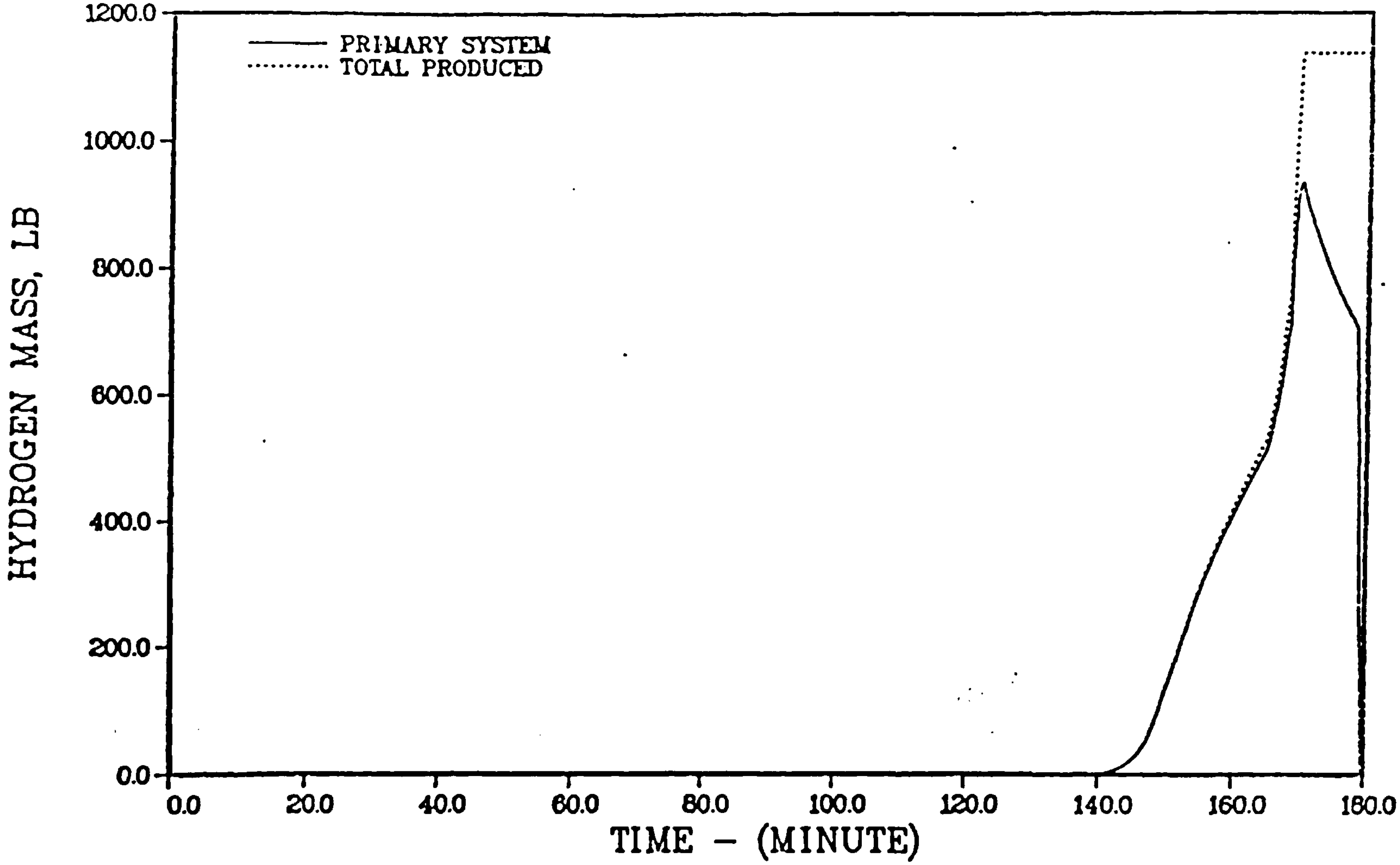


Fig. 8 - Hydrogen Mass, LB

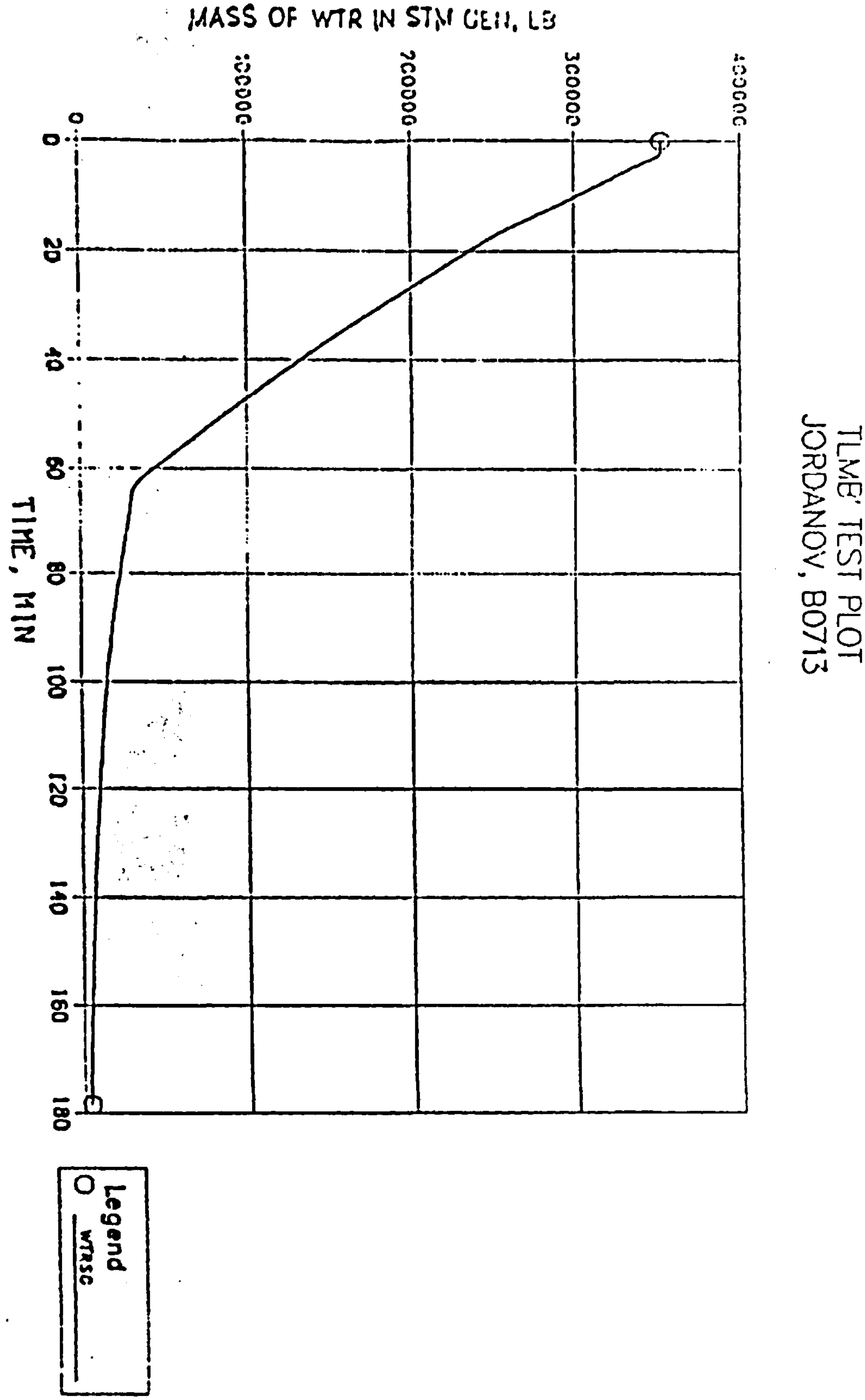


Fig. 9 - Mass of WTR in STM GEN, LB

NEXT PAGE(S) left BLANK

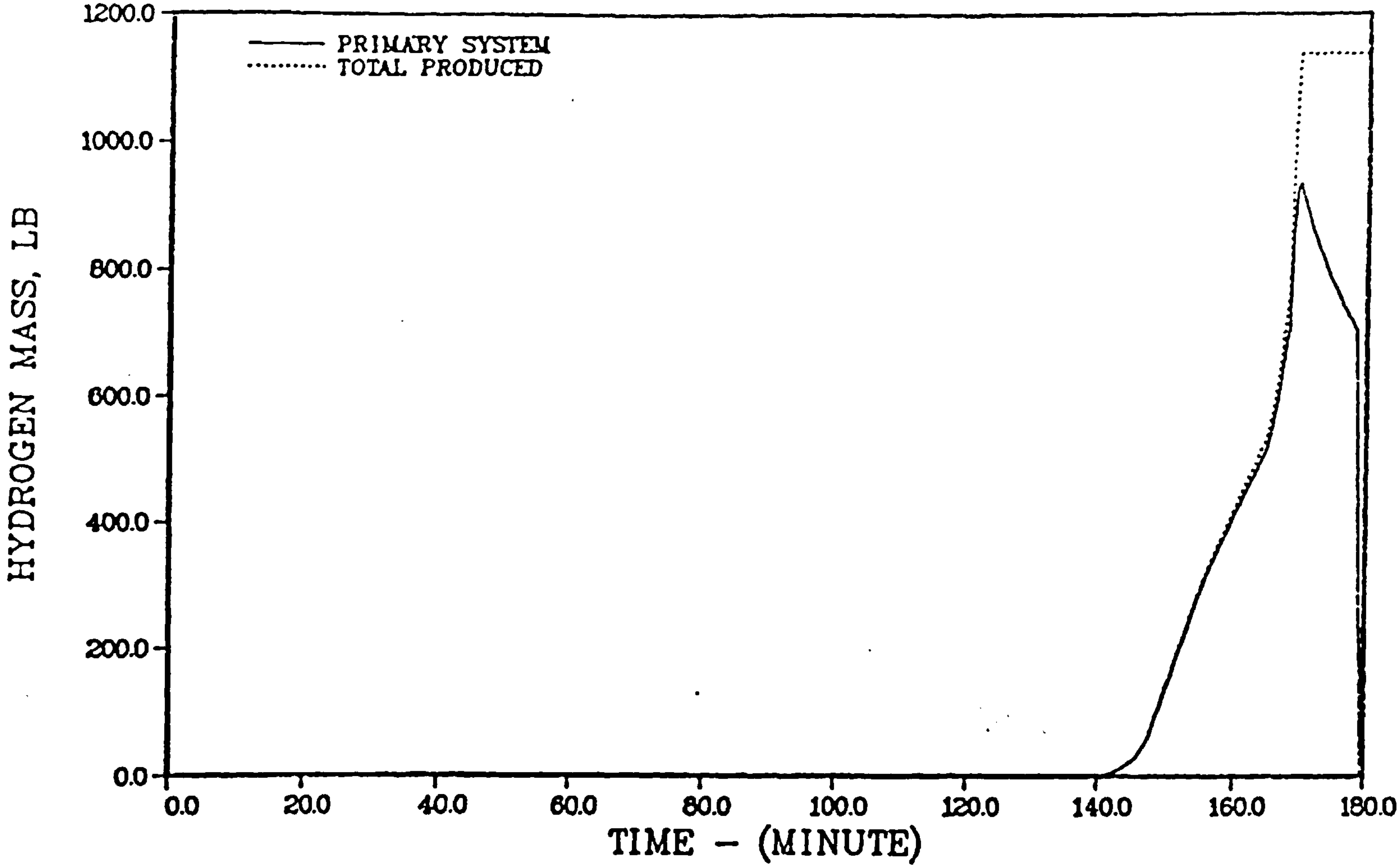


Fig. 10 - Mass of WTR in STM GEN, LB