Dynamic Simulation of Nuclear Hydrogen Production Systems

by

Patricio D. Ramírez Muñoz

Submitted to the Department of Chemical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Chemical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2011

© Massachusetts Institute of Technology 2011. All rights reserved.

Author . . . . . .
Department of Chemical Engineering
October 15, 2010

Certified by . . . . . .
Paul I. Barton
Lammot du Pont Professor of Chemical Engineering
Thesis Supervisor

Accepted by . . . . . .
William M. Deen
Professor of Chemical Engineering
Chairman, Committee on Graduate Theses

Dynamic Simulation of Nuclear Hydrogen Production Systems

by

Patricio D. Ramírez Muñoz

Submitted to the Department of Chemical Engineering
on October 15, 2010, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Chemical Engineering

Abstract

Nuclear hydrogen production processes have been proposed as a solution to rising CO_2 emissions and low fuel yields in the production of liquid transportation fuels. In these processes, the heat of a nuclear reactor is used to run the chemical reactions in a hydrogen plant. The resulting system is tightly interconnected and operates at very high temperature and pressure, which can lead to operational disruptions and accidents. For this reason, computational studies validating the safe operation of the system are required by regulatory authorities. In the past, safety studies have been conducted by using legacy codes, such as RELAP and MELCOR, and their focus has been the operation of nuclear power plants.

However, traditional legacy codes are not appropriate to simulate nuclear hydrogen production. The simulation of a nuclear reactor itself is already complex because it involves simulating reactor kinetics and transport phenomena. To that complexity, nuclear hydrogen production adds the need to simulate chemical reactions in the hydrogen plant. These chemical reactions cannot be represented easily in legacy codes because these codes lack the flexibility, speed and accuracy required to simulate them. Therefore, only a limited number of studies on the safety of these systems exist.

Instead of using legacy codes, this thesis proposes using equation-based simulators developed by the chemical engineering community to model and study the safety of a nuclear hydrogen production plant. Equation-based simulators were designed to be flexible, extensible and fast because they have to simulate a vast range of processes from the chemical industry. Thus, they provide a good platform for the simulation of nuclear hydrogen production systems. This thesis explains the models used for the different parts in the nuclear hydrogen production plant, and then presents the response of this plant model to different accident scenarios.

The first contribution of this thesis is a novel equation-based model for the heat transfer loop connecting a nuclear reactor and a hydrogen production plant. This heat transfer loop uses helium as the heat transfer fluid, which makes simulating its behavior difficult because of the need to model gas dynamics. To resolve this, three models for gas dynamics and two set of coupling conditions for boundary variables

were tested in JACOBIAN, an equation-based simulator. The three models for gas dynamics in combination with a novel approach to set coupling conditions for boundary variables were able to represent the interesting time scales accurately in transient scenarios. The accuracy and computational speed of these simulations outperformed those produced by a reference model created in RELAP, a legacy code.

The second contribution is a model of a nuclear hydrogen production plant using high-temperature steam electrolysis to produce hydrogen. This model was created to study the effect of potential accidents on the nuclear reactor. It included detailed models of the nuclear reactor and heat transfer loop, and a partial model of the electrolysis plant. The nuclear reactor was modeled as a pebble bed modular reactor, which is one of the safest designs available. The reactor was connected to the hydrogen production plant using the heat transfer loop model already developed in this thesis. The hydrogen production plant was partially represented as a steam superheater in the heat transfer loop.

The third contribution is the demonstration of the safety characteristics of the nuclear hydrogen production plant by subjecting the plant model to three accident scenarios. The scenarios involved disruptions in the hydrogen plant or in the heat transfer loop, and all of them—directly or indirectly—lead to a loss of heat sink capacity for the nuclear reactor. This resulted in an increase of the nuclear reactor core temperature, which was quickly moderated by the fission power reduction at the fuel pebbles and by the safe design of the nuclear reactor. As a consequence, the maximum temperature reached in the core was always less than the fuel melting point and the reactor was always in a safe condition. The heat transfer loop could suffer the rupture of a pipe in one of the scenarios, and design modifications to address this were suggested.

This thesis' results partially prove that nuclear hydrogen production plants could be safe, and simultaneously, that equation-based simulators are good platforms to demonstrate the safety of these plants. Developing these models and tests further will help guarantee the safety of the plant and obtain regulatory and public approval for this new nuclear application.

Thesis Supervisor: Paul I. Barton
Title: Lammot du Pont Professor of Chemical Engineering

Acknowledgments

Thank you, Corrinne Fogg, for your love and support during the PhD; the journey was rough, but you always had a smile in your face. I could not have made it without such a wonderful wife.

Professor Paul Barton, I appreciated and enjoyed having you as my advisor. You not only taught me about dynamic systems, but also attention to detail and the highest professional standards. I know I was not the easiest student, so thank you for your patience and financial support all these years.

Professor Kazimi and Professor Green, thank you for being part of my thesis committee and for your time reviewing my work. Your help was fundmental to learn about nuclear systems and how to be a better scientific communicator. My gratitude also goes to Professor Jensen and the Chemical Engineering Department at MIT for educating me and for providing financial support at different times during my studies. Finally, my work would not have been possible without the help of gifted researchers: Dr. Memmott (MIT), Dr. Hejzlar (MIT), Dr. Sherman (SNRL), Dr. Herring (INL) and Dr. Davis (INL), Dr. Vilim (ANL).

Many thanks to my labmates in the Barton group for their friendship and academic advice; working with you helped me grow a great deal. Dr. Ajay Selot, your technical guidance and lessons made me a better researcher; listening to your point of view in different topics from economics to human nature made a better person—thank you. Professor Alexander Mitsos, thank you for your constant encouragement and strategies to do good research; you believed in me more than I did. Dr. Cha Kun Lee, I appreciated your friendship and your teaching me DSL despite the fact that you were extremely busy finishing your thesis. Dr. Derya Ozyurt, thank you for starting the project that was the foundation of my thesis and for sharing with me your knowledge about nuclear hydrogen production systems. Professor George Bollas, I am grateful to you for being my project colleague and for shouldering the project responsibilities with me. Geoff Oxberry, thank you for your discussions about how to best present my work to professors and for making the lab environment welcoming.

Matt Stuber, I am indebted to you for your fashion advice and car advice—everything was spot on. I enjoyed working with all the exceptional researchers in the lab: Achim Wechsung, Spencer Schaber, Yang Chen, Professor Thomas Adams II, Mehmet Yunt, Yang Chen, Xiang Li, Richard Lakerveld, Kamil Khan, Ruth Misener, Vibhu Prakash Saxena, Earl Solis, John Tolsma, Brian Simpson, Joe Scott, Audun Aspelund, Lars Hellemo, and Emre Armagan.

Thank you to all my friends in Boston and in Chile for the great times together and the encouragement all these years. In particular, I would like to extend my sincere gratitude to Joel Yuen, for the great discussions and the fine cooking; you are a truly reanaissance man and my hero. Arman Haidari, I enjoyed our conversations about running and school. Biz Bose, thank you for the motivation and help to get the job I wanted. Surasak Chunsrivirot, our table tennis matches were delightful. Ian Hoag, doing adventure racing with an ex-Navy SEAL is one of the best training experiences I could have ever had. Professor Luis Rademacher, your example motivated me to come to MIT and your support helped me get there. Cristian Bawlitza, you have a been wonderful friend all these years—thank you.

Thank you to all the staff members of the Chemical Engineering and Nuclear Engineering departments; you not only made my life infinitely easier, but also you cheered me on every single time. In particular, I would like to thank Suzanne Maguire, Katie Lewis, Christine Preston, Barbara Balkwill, Rebecca Hailu, Craig Abernethy, Carolyn Carrington, Melanie Miller, Alina Haverty, Jean Belbin, and Fran Miles.

I'm indebted to Kathy Neumueller, Julie Cook, Hernan Saenz, and to the people at Bain & Co. in Dallas. You let me spend a summer at your great office and gave me the opportunity to come back as a full-time employee after finishing my PhD. This gave me extra motivation to finish my studies.

Finally, thank you Darío Ramírez and Alejandra Muñoz, I could not be here without you. My achievements are only the result of being raised by a great family. Thank you, Paula Ramírez, for being there all these years to share your PhD experience with your brother. Blanca and Ana, I'm sorry I didn't spend as much time as I wanted with you; I will make up for that in the coming years.

Contents

1 Introduction **15**

2 Background **19**

2.1 Nuclear energy to aid fuel production 19

2.1.1 Current transportation energy challenges 20

2.1.2 New nuclear applications for hydrogen and heat 23

2.1.3 Nuclear hydrogen production can make the biggest impact . . 28

2.2 The need for new computational models 30

2.2.1 Legacy codes lack simulation features 31

2.2.2 Equation-based simulators are a good alternative 32

2.3 Nuclear hydrogen plant used in this thesis 36

2.3.1 PBMR and PCU Operation 37

2.3.2 HTSE 39

2.3.3 Heat transfer loop 40

3 Dynamic simulation of a heat transfer loop for alternative nuclear reactor applications **43**

3.1 Introduction 43

3.2 Using equation-based languages for the simulation of novel nuclear applications 46

3.2.1 Equation-based simulators provide flexibility, speed and accuracy 46

3.2.2 Simulation challenges in equation-based simulators are minimal and mostly related to initial conditions 48

3.3 Modeling the heat transfer loop 50
3.3.1 Dynamic models in JACOBIAN 51
3.3.2 RELAP model for comparison 61
3.4 Model Performance Evaluation 62
3.4.1 Test 1: Feasibility and steady-state profiles 62
3.4.2 Test 2: Dynamic simulation profiles 64
3.4.3 Test 3: Computational time 71
3.5 Conclusions 71

4 Transient analysis of a nuclear hydrogen production facility 75
4.1 Introduction 75
4.2 Definition of the system 77
4.3 Dynamic model of a nuclear hydrogen production system 80
4.3.1 Equation-based simulators as a simulation platform 81
4.3.2 Nuclear reactor model 82
4.3.3 Heat Transfer Loop Model 87
4.4 Transient simulations 88
4.4.1 Loss of heat sink accident 89
4.4.2 Loss of flow in the heat transfer loop 93
4.4.3 Leak in the heat transfer loop 95
4.5 Conclusion 98

5 Conclusions 101

A Discretization of Gas Dynamic Equations 105
A.1 Discretization of the continuity equation 107
A.2 Discretization of the momentum equation 109
A.3 Discretization of the energy equation 115

B JACOBIAN code 119

List of Figures

2-1 Using alternative feedstocks to produce liquid fuels increases CO_2-eq emissions [28]. 21

2-2 Equation-based simulation languages allow the variables and equations describing physical and chemical phenomena to be declared explicitly [5]. 34

2-3 The nuclear hydrogen production used in this thesis is a combination of technologies in development. 37

2-4 The design of fuel pebbles for PBRs avoids meltdowns by having a high fusion point (2000°C) and decreases the possibility of radiation spread [26]. 39

2-5 The SOEC cell dissoaciates superheated steam into hydrogen and oxygen [36]. 41

2-6 A HTSE plant is designed to be built in modular form [42]. 41

2-7 Heat transfer loop [8]. 42

3-1 Nuclear hydrogen production is an example of novel nuclear applications. The design of the heat transfer loop requires special attention. 45

3-2 Discretizing the gas dynamics equations using combination of standard grid and a staggered grid makes the solution of the system more stable. 54

3-3 One approach to set the boundary variables; it uses P_{In} as suggested in the literature [27]. 57

3-4 Alternative approach to set the boundary conditions for a gas dynamics model; it replaces the specification on P_{In} with $P_{Aux,1}$. 59

3-5 RELAP model used to benchmark JACOBIAN models. 62
3-6 Heat transfer loop's pressure profiles at steady state in JACOBIAN match RELAP's profiles with small differences. 64
3-7 Heat transfer loop's temperature profiles at steady state in JACOBIAN match RELAP's profiles with small differences. 65
3-8 The heat transfer loop's total helium mass decreases in RELAP after introducing 50% decrease in PHX's cold stream inlet flowrate. 66
3-9 Pressure of helium in the loop at the end of the PHX after a 50% decrease in PHX's cold stream flowrate. RELAP simulation reaches a lower pressure than that in JACOBIAN simulations because of RELAP's numerical error in mass conservation; however, RELAP and JACOBIAN models respond with the same speed to the step change. 67
3-10 Temperature of helium in the loop at the end of the PHX after a 50% decrease in PHX's cold stream flowrate. RELAP simulation reaches a lower temperature than that in JACOBIAN simulations because of RELAP's numerical error in mass conservation; however, RELAP and JACOBIAN models respond with the same speed to the step change. 68
3-11 Pressure of helium in loop at the outlet of the compressor; the time scale of the response to a change in the compressor is similar again in the four models, but it differs during the first second of the transient. RELAP's numerical errors in mass conservation affect the final steady state. 69
3-12 Temperature of helium in loop at the outlet of the compressor; the response to a change in the compressor is similar again in the four models, but it differs during the first second of the transient. RELAP's numerical errors in mass conservation again affect the final steady state. 69
3-13 The RELAP model's pressure profile presents oscillations during the first 0.5 s of the transient because of pressure wave effects in the helium. The simplifications used in the JACOBIAN models distort or eliminate these oscillations. 70

3-14 The RELAP model's temperature profile presents oscillations during the first 0.5 s of the transient because of pressure wave effects in the helium. The simplifications used in the JACOBIAN models distort or eliminate these oscillations. 70

3-15 Heat transfer loop simulations in JACOBIAN are more than 10 times faster than corresponding simulations in RELAP. 72

4-1 Nuclear hydrogen production system studied; the units surrounded by the dashed line will be simulated in this study. 78

4-2 The PBMR is safe, modular, and economical, and it can produce the high-temperature heat needed to run hydrogen production plants. . . 79

4-3 After a loss of heat sink accident, the temperature in the cold leg of the heat transfer loop increases more than 330°C. 90

4-4 Increased temperature in cold pipe after loss of heat sink accident decreases pipe's stress resistance from over 10^5 hrs to less than 1 hr. . 91

4-5 After a loss of heat sink accident, the helium entering the core is hotter, but its temperature is lower at the outlet. 92

4-6 After a loss of heat sink accident, the average temperature in the fuel slightly increases, which leads to a reduction in the reactor power. . . 92

4-7 After a loss of flow in the heat transfer loop, the helium temperature around the loop stays within specifications. 94

4-8 After a loss of flow in the heat transfer loop, the temperature of helium at the core inlet increases. The core response reduces the temperature at the core outlet. 94

4-9 After a loss of flow in the heat transfer loop, the core reacts safely by reducing the fission power. 95

4-10 After a leak accident in the heat transfer loop, the temperature of helium along the loop stays within specifications. 96

4-11 After a leak accident in the heat transfer loop, the temperature at the core inlet slightly increases, which results in a minor temperature decrease at the core outlet. 97
4-12 After a leak accident in the heat transfer loop, the heat transfer loop still can remove a significant amount of heat from the nuclear reactor coolant. This results in a power reduction of 24 MWth. 98

A-1 A combination of standard grid and a staggered grid is used to discretize the gas dynamics equations. 106
A-2 i^{th} control volume in the standard grid. 108
A-3 i^{th} control volume in the standard grid. 109
A-4 i^{th} control volume in the staggered grid. 110
A-5 i^{th} control volume in the staggered grid. 110
A-6 i^{th} control volume in the staggered grid with a change in section. . . . 111
A-7 i^{th} control volume in the staggered grid. 114
A-8 i^{th} control volume in the standard grid. 116

List of Tables

2.1 High-temperature gas reactors have been developed since the 1960s; the current ones being developed today are at the commercial scale [10, 65]. 29

2.2 Four different nuclear hydrogen production programs are in development [10]. 29

Chapter 1

Introduction

Replacing crude oil with more local and more affordable feedstocks, and decreasing CO_2 emissions are the most important problems that the U.S. needs to tackle regarding transportation fuels [12]. Oil is fundamental for transportation in the U.S.; it accounts for 95% all the energy used in this sector. Yet its sourcing is increasingly more complex; two thirds of oil is imported mostly from politically unstable regions, and the availability of cheap-to-process light crude oil is decreasing [13]. This has resulted in high prices [6] and an oil supply chain with a higher risk of disruption. At the same time, the processing of oil into liquid fuels and the combustion of these fuels emits large amounts of CO_2; thus, increasing the CO_2 concentration in the atmosphere. This higher CO_2 concentraion is believed to affect the climate and catastrophic consequences have been projected.

To produce liquid fuels from alternative feedstocks and decrease CO_2 emissions simultaneuously, it appears beneficial to use new nuclear applications to generate heat and hydrogen to support these novel production processes. Affordable and more local feedstocks such as oil sands, coal and biomass can be good replacements for crude oil in liquid fuel production. Unfortunately, their processing into liquid fuels requires large amounts of hydrogen and heat, which are currently generated by consuming natural gas or the same feedstock. This results in emissions of considerable amounts of CO_2 in the supply chain (*fuel chain*), which can more than double the total emissions of the fuel chain. Also, when the feedstock itself is consumed to run operations,

low fuel yields are achieved. CO_2 emissions and low yields can be avoided by using nuclear energy, which can provide the necessary hydrogen and heat. Several new nuclear applications to support liquid fuel production processes have been proposed, and special reactors are in development for these purposes.

Demonstrating the safety of these new nuclear applications with computational simulations is fundamental to obtaining regulatory approval. Producing hydrogen by using thermal power from a nuclear reactor requires a tight coupling between the reactor and the hydrogen production plant. Using heat from a nuclear reactor to support a liquid fuels production plant also requires tight coupling. These systems will operate at extreme conditions that can cause operational upsets and affect the nuclear reactor. Validating that the reactor and system will behave safely during this and other extreme upsets is required to obtain approval from authorities. This is achieved in part by studying the system with computional simulations, which allow different combinations of conditions to be tested without risking real accidents. Computational simulations have been used extensively to study nuclear reactors in power plants, and different legacy codes exist for this (e.g., RELAP [52], MELCOR [14]).

However, legacy codes do not have enough flexibility, speed and accuracy to simulate these new nuclear applications. The simulation of nuclear reactors is already complex because it involves simulating reactor kinetics and transport phenomena. To that complexity, new nuclear applications add the need to simulate chemical reactions in the hydrogen plant or in the liquid fuels plant. Chemical reactions cannot be represented easily in legacy codes because these codes lack the flexibility, speed and accuracy to simulate them. Some efforts have been made to modify legacy codes to represent chemical reactions, but they involve large teams of scientists. This situation has resulted in a limited number of studies on the safety of these systems.

This thesis proposes using equation-based simulators to model the new nuclear applications and illustrates this by creating and simulating models for a nuclear hydrogen production plant. Equation-based simulators were created to simulate a large range of process in the chemical industry. Thus, they are specifically designed to be extensible, flexible and fast, and can provide the platform needed to model new

nuclear applications. From all the proposed nuclear applications, nuclear hydrogen production is the one that can have the biggest impact. Therefore, nuclear hydrogen production was used as the case study and high temperature steam electrolysis was chosen as the specific method of production.

The specific contributions of this thesis are:

- a set of models to represent a nuclear reactor coupled to a hydrogen production plant,
- a new method to set the boundary conditions in a closed heat transfer loop using gas as the heat transfer fluid, and
- the simulation of three new possible accident scenarios encountered in a hydrogen production plant.

This thesis is structured in five chapters. Chapter 2 explains how new nuclear applications —in particular nuclear hydrogen production— can help solve problems in the production of fuels for the transportation sector. It also explains the computational difficulties in simulating the new nuclear applications and how equation-based simulators can be an appropriate platform. Chapter 3 explains how to simulate correctly a heat transfer loop connecting a nuclear plant and a hydrogen plant. Chapter 4 presents the dynamic models for the nuclear reactor, the heat transfer loop and part of the hydrogen plant. These models are used to represent and analyze three possible accident scenarios caused by upsets in the hydrogen production plant or the heat transfer loop. Chapter 5 presents the final conclusions and recommendations.

Chapter 2

Background

This chapter explains the problems associated with fuel production processes from alternative feedstocks in the U.S. It details how nuclear energy can help solve these problems by providing heat and hydrogen for the new facilities needed. It also describes the challenges in the simulation of these new nuclear applications and how equation-based simulators can provide a good platform to replace/supplement legacy codes. It finally specifies the nuclear hydrogen plant used as a case study: a pebble-bed reactor coupled to a high-temperature electrolysis plant.

2.1 Nuclear energy to aid fuel production

One of the key priorities for the transportation sector in the U.S. is to replace/reduce crude oil in the production of liquid transportation fuels. To do this, alternative feedstocks have been proposed and studied: oil sands, coal, biomass and natural gas [28, 12]. These alternative feedstocks are advantageous because they can be produced locally or in politically stable countries, and because they are either cheaper or cleaner. However, the pathways to transform them into liquid fuels require large amounts of heat and hydrogen, which are generated by consuming natural gas or the same feedstock. This results in the emission of large quantities of CO_2 and low fuel yields. Using nuclear energy to produce the heat and hydrogen needed can eliminate the high CO_2 emissions and increase fuel yields; thus, paving the way to replace crude oil

with alternative feedstocks. This section explains the problems in the production of liquid fuels using alternative feedstocks and details how nuclear energy can help with heat and hydrogen production. At the same time, this section explains that nuclear hydrogen production will be the focus of this thesis because this application might have the biggest impact in the future.

2.1.1 Current transportation energy challenges

Transforming alternative feedstocks (e.g., oil sands, coal, biomass and natural gas) into liquid transportation fuels requires a supply chain (*fuel chain*) with several steps: *extraction/production*, *conversion/refining* and *transportation/distribution* [28]. Unfortunately, these fuel chains are very energy intensive and can consume more than five times the energy needed by the fuel chain for crude oil. This also results in corresponding large CO_2 emissions, as the CO_2 emissions are approximately proportional to the energy used in each step. The energy demand and emissions vary depending on the process and feedstock used, but are caused mainly by the production of process heat and hydrogen.

1. *Light crude oil.* Production of liquid transportation fuels from light crude oil (e.g., Arab light crude) already requires large amounts of energy, which results in fuel chain emissions of 143 g CO_2 equivalent per mile driven in a SUV (g CO_2-eq/mile in SUV) or 39% of the CO_2 emitted by burning the fuel itself [28]. In particular, refineries are the largest energy consumers in the U.S. and most of this energy is used to generate heat and hydrogen. High-temperature heat (500–700°C) is required for several operations, distillation and thermal cracking being the major demands. This heat input accounts for 10% of the energy in the final fuel. The total heat input required to process the crude oil consumption in the U.S. is 142 GWth or about 50% of the nuclear capacity installed in the U.S [12]. Hydrogen is also widely used at different process steps, hydrotreating and hydrocracking being the most important ones. Refineries use approximately 4 million tonnes of hydrogen per year [64], which makes them also the largest

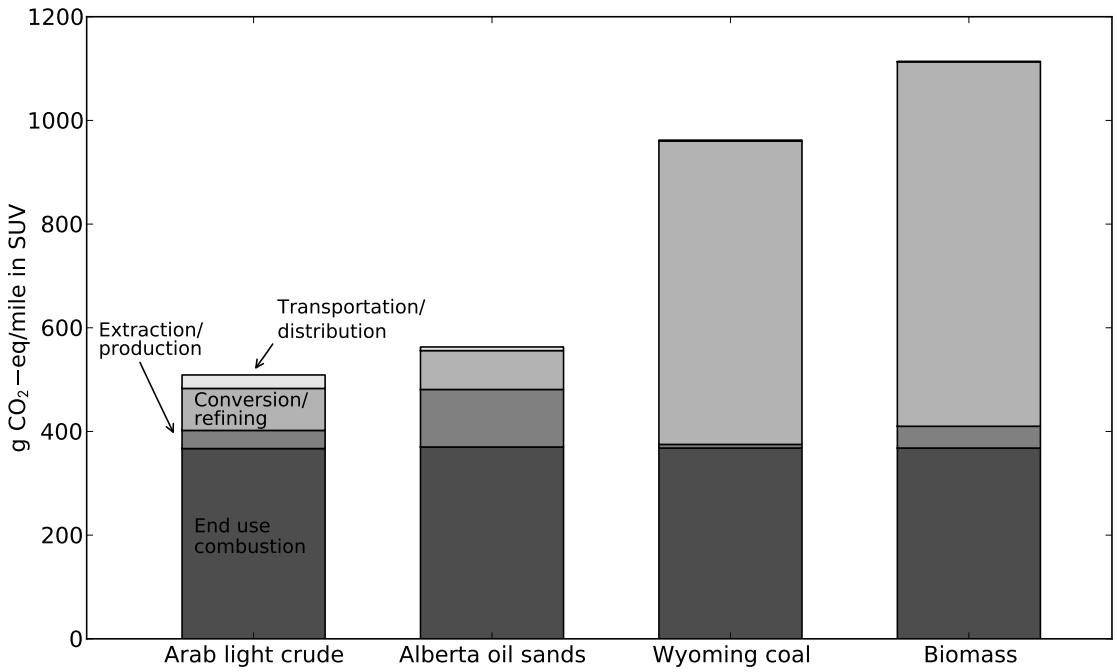

Figure 2-1: Using alternative feedstocks to produce liquid fuels increases CO_2-eq emissions [28].

consumers of hydrogen in the U.S.

2. *Heavier crudes and oil sands.* As heavier forms of crude oil are used for the production of liquid transportation fuels, the processing becomes more complex and results in higher energy consumption and higher CO_2 emissions. For example, the fuel chain for Alberta oil sands emits 193 g CO_2-eq/mile in SUV or 52% of the CO_2 emitted when burning the fuel (Fig. 2-1) [28, 31]. These emissions correspond to 35% more than the emissions generated in the processing of light crude.

   In the case of oil sands, the higher CO_2 emissions and energy requirements are explained by the increased intensity of the mining, extraction and upgrading processes. The mining of oil sands in open pits uses large quantities of fuel to operate the equipment (e.g., shovels) and to move oil sands to extraction facilities by hydrotransport pipelines. The extraction of oil sands requires large quantities of steam to decrease their viscosity and separate the bitumen from solids. For example, in situ thermal extraction requires around 1.0 Mcf of

natural gas per barrel of bitumen [34], which corresponds to approximately 17% of the energy in the bitumen. Finally, upgrading the bitumen produces syncrude that can be processed by traditional refineries. This stage involves additional steps of distillation, cracking and hydrotreating, which need large amounts of heat and hydrogen. Most of this heat and hydrogen are currently produced from natural gas, and the requirements are about 0.1 and 0.4 Mcf of natural gas per barrel of bitumen, respectively [34].

3. *Coal.* Coal, a local and abundant feedstock in the U.S. (e.g., Wyoming Coal), can also be used to produce liquid transportation fuels by the Fischer-Tropsch process. However, this fuel chain requires much more energy than the fuel chain for crude oil, and its emissions are 594 g CO_2-eq/mile in SUV or 3.2 times more (Fig. 2-1). Most of the CO_2 emissions and energy consumption are associated with the production of hydrogen from CO and H_2O in the water gas shift (WGS) reaction: $CO + H_2O \leftrightarrow CO_2 + H_2$. Additionally, this step results in very low yields in the conversion of coal into fuel, as CO is produced by the gasification of coal. Only 30% of the carbon in the coal input to the process ends in the final liquid fuel [7].

4. *Biomass.* The two most important ways to produce liquid transportation fuels from biomass are the production of ethanol by fermenting biomass-derived sugars and the production of hydrocarbon fuels by gasification and the Fischer-Tropsch process.

   The production of ethanol by fermentation is an energy-intensive process. For example, producing a gallon of ethanol from corn starch requires 120% of the energy value in the ethanol gallon, and generates a corresponding amount of CO_2-eq emissions [61]. A large part of the energy is used to produce process heat for separation steps. Currently this heat is mostly generated by burning natural gas, which results in large CO_2 emissions. In addition, the fermentation process itself generates CO_2, as yeast produces two molecules of ethanol and two of CO_2 per molecule of glucose.

A more efficient way to produce liquid fuels based on biomass is using gasification and the Fischer-Tropsch process to produce liquid fuels, because this process can use the biomass constituents as feedstock. This process works similarly to the one used for coal described previously. Here again, using the feedstock to produce hydrogen in the WGS reaction accounts for a large part of the CO_2 emissions, energy used and low conversion yields. The fuel chain emissions in this case are 746 g-CO_2-eq/mile SUV or 4.2 times more than the emissions for the light crude oil fuel chain (Fig. 2-1) [28]. These extra emissions are accounted for by the energy used for cultivating, harvesting and transporting biomass. On the other hand, such a fuel chain could be CO_2 neutral if all the energy comes from biomass or renewable sources.

Additionally in this case, it is crucial to increase the conversion yield of biomass into liquid fuels by finding alternative means of hydrogen production. Not only is it expensive and energy intensive to produce and collect biomass, but also the area that can be used to supply the biomass is limited in size. For example, one study used 17.25 miles as the average haul from the field to the plant [28]. Therefore, low conversion yields severely limit the output of a single facility. As the production of hydrogen for the Fischer-Tropsch process is the biggest factor decreasing the conversion yield, using alternative hydrogen sources can increase the output substantially.

2.1.2 New nuclear applications for hydrogen and heat

As explained above, transforming alternative feedstocks into liquid transportation fuels requires large amounts of energy to produce the hydrogen and heat needed for the processes. Producing this hydrogen and heat with conventional approaches results in high costs, high CO_2 emissions and low conversion yields, which makes it hard to use alternative feedstocks for liquid transportation fuels. However, producing the hydrogen and heat with cheaper and cleaner substitutes could solve this problem. Nuclear energy is one of the possible substitutes and extensive applications of it to

liquid transportation fuel production have been analyzed [12]. This section reviews how nuclear energy can be used to produce hydrogen – nuclear hydrogen production – and how it can provide heat for different liquid fuel production processes.

Nuclear hydrogen production

Most hydrogen production involves significant emissions of CO_2 because it relies on fossil fuels. Coupling a very high-temperature nuclear reactor to a hydrogen production plant could provide carbon neutral and inexpensive hydrogen. Two hydrogen production technologies are being developed to do this: high-temperature steam electrolysis (HTSE) and the sulphur-iodine (SI) cycle [63]. These two technologies, their applications and their benefits are explained in this section.

HTSE can produce hydrogen with no CO_2 emissions and approximately double the efficiency of traditional electrolysis. This process uses part of the heat from a high-temperature nuclear reactor to produce superheated steam with temperature in the range of 800-900°C. The rest of the high-temperature heat from the nuclear reactor is used to produce electricity using a Brayton cycle. This electricity is then used in the solid oxide electrolysis cells to produce hydrogen and oxygen from the superheated steam [63]. The use of high-temperature heat to produce electricity results in increased power cycle efficiency over previous power plants. In addition, the operation of the electrolysis cell at high temperatures enables a reduction in the electrical power needed for the electrolysis reactions. The combination of these two factors achieves thermal efficiencies in the production of hydrogen of 45% to 55% [63, 18]. This greatly exceeds efficiencies of 27% achieved by current electrolyzer technologies using electricity from earlier generations of power plants.

The SI cycle can also produce hydrogen from the heat produced by a nuclear reactor without CO_2 emissions and at high efficiency. This cycle was developed by General Atomics in the 1970s [63] and involves the dissociation of water by using

three chemical reactions:

$$I_2 + SO_2 + 2H_2O \rightarrow 2HI + H_2SO_4, \qquad (120°C)$$

$$H_2SO_4 \rightarrow SO_2 + H_2O + 1/2O_2, \qquad (830\text{-}900°C)$$

$$2HI \rightarrow I_2 + H_2. \qquad (300\text{-}450°C)$$

These reactions have been demonstrated in the laboratory and progress has been made in scale up [10]. All the reagent in these reactions is recycled, which means that only hydrogen and oxygen are the final output of the system. In this case, the high-temperature reactor is mainly used to provide high-temperature heat for the second reaction. The thermal efficiency of this process has been estimated between 45% and 55% [63].

Hydrogen produced with either of these two nuclear hydrogen production technologies could be used in different liquid transportation fuel processes to reduce emissions or increase efficiency:

- *Crude oil refining.* In the U.S., displacing the use of natural gas for hydrogen production in refining processes would decrease CO_2 emissions by 10–15% and would require 35–41 GWth of nuclear reactor capacity. In the U.S. in 2003, the 145 active refineries required 4.08 million tonnes of hydrogen for hydrotreating, hydrocracking and other processes [64]. Almost all of this hydrogen was produced by using natural gas steam reforming, and this process accounted for 10–15% of the CO_2 emissions [62, 48]. These emissions could be avoided by using hydrogen produced by nuclear hydrogen production plants using HTSE or SI. It is estimated that 60–70,000 tonnes of hydrogen per year can be produced from a 600 MWth high-temperature nuclear reactor [12]. Then, the nuclear capacity required supply all the refineries in the U.S. corresponds to 35–41 GWth. This corresponds to approximately 11–13% of the current installed nuclear capacity in the U.S., as the average power cycle efficiency of this capacity is 32% [1].

- *Oil sands upgrading.* Incorporating nuclear hydrogen production for the upgrading of Alberta oil sands could decrease CO_2 emissions in the extraction/production step by 35%. Upgrading one barrel of bitumen of Alberta oil sands requires 2.4 kg of hydrogen on average [44]. As the production of 1 kg of hydrogen from natural gas generates 11.9 kg CO_2-eq [50], the upgrading of one barrel of bitumen generates 28 kg CO_2. Overall, the production of one barrel of bitumen emits about 81 kg CO_2-eq [34], which means emissions would decrease by 35% in this step.

  Replacing all the hydrogen used for upgrading with hydrogen produced with nuclear means would require 5.1–6.0 GWth of nuclear power. In 2007, the production rate of bitumen in Alberta was approximately 700 thousand barrels per day. Thus, the corresponding requirement for hydrogen was about 0.6 million tonnes of hydrogen per day. As 60–70,000 ton-H_2/yr need 600 MWth of nuclear power for its production, satisfying this hydrogen demand would need 5.1–6.0 GWth of nuclear power.

- *Converting coal to liquids or biomass to liquids.* CO_2 emissions in a coal-to-liquids facility could be reduced by 49% by using nuclear hydrogen production. In a traditional coal-to-liquids facility, only 30% of the carbon in the coal is incorporated into the final liquid fuel [7]. Most of the remaining 70% ends as CO_2 in the WGS reaction and accounts for nearly all the CO_2 emissions. Nuclear hydrogen can replace the WGS reaction to provide hydrogen for the Fischer-Tropsch process, and this would result in 66% of the carbon in coal being incorporated in the final fuel. Therefore, emissions could be reduced by 49%. A coal-to-liquids facility processing 19 thousand tons of coal per day would require 0.5 thousand tons of hydrogen per day [7]. This rate of hydrogen can be produced by a nuclear hydrogen facility with 1.7 GWth of power. Using nuclear hydrogen production to support a biomass-to-liquids facility would have a similar effect. This is crucial in the case of biomass as the production rate of the facility is limited by the area of land that can supply the facility. Thus,

doubling the carbon conversion yield would allow a single facility to double its output and achieve economies of scale.

Nuclear heat integration

Nuclear reactors can also be used to provide heat for liquid fuel production processes and reduce CO_2 emissions. This will require a customized design for heat integration, as the different steps in each process use different amounts of heat at different temperatures. Using nuclear reactors to produce heat for industrial applications – including ethanol production – has been already tried successfully in other parts of the world. This section will describe some potential uses of nuclear heat in liquid fuel production processes.

1. *Refineries.* The heat needed to supply refineries in the U.S. would require a nuclear capacity of 142 GWth. Most of the heat used in refineries is used in distillation and thermal cracking, and it accounts for 10% of the heat content in the final liquid fuel. The total fuel output from refineries is equivalent to 20 million barrels of oil per day; then, the nuclear capacity required in this case is 142 GWth [12]. The heat provided should be at 600–700°C, which would require designing specific equipment to transfer the heat at such high temperatures.

2. *Oil sands extraction and upgrading.* In the case of oil sands, heat could be used to produce steam to extract bitumen in situ, or to separate it from solid residues [11]. In the case of extraction in situ, one pebble bed modular reactor of 250 MWth could support the production of 50 thousand barrels of bitumen per day. Other process designs are going even further by suggesting the idea of nuclear underground refining, where high-temperature heat is used to distill the bitumen in situ [12]. A process like this would avoid all the CO_2 emissions associated with producing the high-temperature heat from natural gas and it would require a 600 MWth reactor to produce 50 thousand barrels of oil per day.

3. *Ethanol production from corn.* Using low-temperature steam from nuclear re-

actors could reduce by 50% the fossil fuel use in starch-based ethanol refineries. 50% of the energy consumed in traditional ethanol refineries is used to produce steam with low temperature and pressure. The same kind of steam is produced by nuclear power plants as a by-product, and it can be easily transported for a distance of about one mile. Thus, using the steam from a nuclear power plant in an ethanol refinery is feasible, would reduce emissions and would add value to the power production in the nuclear plant. Producing 2.4 million barrels of ethanol per year would need 80 MWth of steam [12].

2.1.3 Nuclear hydrogen production can make the biggest impact

As demonstrated, nuclear hydrogen production and nuclear heat coupling can both reduce CO_2 emissions and increase feedstock conversion yields. From these two new nuclear energy applications, it is nuclear hydrogen production that can make the biggest impact in the near future. Some of the reasons for this are that nuclear hydrogen production could reduce more than 1% of CO_2 emissions in the U.S., and that its development has made significant progress. At the same time, its application could be highly standardized, contrasting the customized integration required for heat coupling projects. Furthermore, nuclear heat coupling has not been developed much. For all these reasons, this thesis will focus on nuclear hydrogen production.

The development of nuclear hydrogen production has made significant progress. Seven high-temperature gas reactors have already been built–only two of them are operational–and five more will be built by 2020 (Table 2.1) [10]. In particular Japan and China have made considerable progress. They own the two operational reactors (the 30-MW HTTR and 10-MW HTR-10, respectively) and are developing larger versions of them (the 300-MW GTHTR300 and the 250-MW HTR-PM, respectively) [10, 65]. All these reactors are also being designed to be passively safe, which will help gain public acceptability. In addition, these reactors could be used for hydrogen production, and research efforts are also being developed in this direction (Table 2.2)

[10].

Table 2.1: High-temperature gas reactors have been developed since the 1960s; the current ones being developed today are at the commercial scale [10, 65].

Reactor	Location	Power (MWth)	He temperature in/out (°C)	Core	Operation years
Dragon	UK	20	350/750	Cylindrical	1965-1975
Peach Bottom	USA	115	377/750	Cylindrical	1967-1974
AVR	Germany	46	270/950	Pebble bed	1968-1988
Fort St Vrain	USA	842	400/775	Hexagonal	1976-1989
THTR	Germany	750	270/750	Pebble bed	1985-1989
HTTR	Japan	30	395/950	Hexagonal	1998-Present
HTR-10	China	10	250-300/700-950	Pebble bed	2000-Present
PBMR	SA	500	350/950	Pebble bed	In development
ANTARES	France	600	400/850	Hexagonal	In development
GT-MHR	USA/Russia	550-600	490/950	Hexagonal	In development
HTR-PM	China	250	250/750	Pebble bed	In development
GTHTR300	Japan	600	589/850	Hexagonal	In development

Table 2.2: Four different nuclear hydrogen production programs are in development [10].

Company	JAEA	GA	PBMR/Westinghouse	AREVA NP
Country	Japan	USA	SA/USA	France
Reactor	GTHTR300	MHR-GT	PBMR	ANTARES
Power	600	550–600	500	600
PCS	Brayton	Brayton	Rankine	Rankine
HPP	SI	HTE/SI	HyS/HTE	HTE/SI
Cycle configuration	Direct PCS, series indirect HPP	Direct PCS, parallel indirect HPP	Indirect, series HPP and PCS	Indirect, parallel HPP and PCS
IHX	Helical coil S&T	Single-stage PCHE	Two stage PCHE	PCS: helical coil S&T Process: PCHE or fin plate

In addition, nuclear hydrogen production would operate as a standard standalone plant, avoiding the customized integration required by nuclear heat coupling. The hydrogen transfer is the only point of contact between a nuclear hydrogen plant supporting a fuel production plant. As hydrogen can be stored in limited quantities, the operation of the nuclear hydrogen plant is relatively independent from its fuel counterpart. Therefore, a standard modular nuclear hydrogen plant could be used to assist a large variety of fuel plants. This would make the manufacturing and deployment of nuclear hydrogen plants cheaper and faster. On the other hand, nuclear

heat coupling requires a customized integration with the fuel plant, which will make the adoption of this technology much slower.

Moreover, nuclear heat coupling has not been developed significantly. Research exploring the use of nuclear heat in refineries has not been reported yet [12]. Also, only preliminary studies have been on the use of nuclear heat to extract bitumen from oil sands [12, 11]. The development of these topics would require large multidisciplinary teams with understanding of materials, chemistry and nuclear engineering. This author does not know of such a team yet.

2.2 The need for new computational models

New nuclear applications for fuel production, such as nuclear hydrogen production, will need to demonstrate safe operation by going through a rigorous assessment, similar to the one used for nuclear power plants. This assessment involves understanding how the plant behaves during normal operation and how it responds to possible equipment failures and disruptions. In particular, the assessment will help validate the limits for the important process variables, the design of the safety procedures and the specifications for the different materials and equipment used [56]. Positive results in this assessment are fundamental to obtain a license for the reactor, because they will help guarantee the public health and safety.

The scenarios used to validate new nuclear applications will include adapted versions of the ones used for nuclear power plants and new ones depending on the chemical processes used. Scenarios including operational disruptions will comprise the decrease/increase of heat removal by the secondary system, decrease in the reactor coolant system flowrate, reactivity and power disruptions anomalies, radioactive release from a subsystem, etc. Normal operation scenarios will involve studying the procedures for the normal operational transients that the plant will experience such as start-up, shutdown and re-start.

In the case of nuclear power plants, these scenarios are studied using computer simulation codes that can model the associated physical phenomena. The leading ones

in the U.S. are MELCOR, MAAP4 and RELAP5 [57]. MELCOR and RELAP5 are used by regulatory agencies and research institutions to evaluate hypothetical severe accident events such as a station blackout or the potential for steam generator tube rupture. MAAP4 is the severe accident code most widely used by nuclear utilities and vendors because of its short run time and reduced requirements for code expertise.

However, the simulation of new nuclear applications for fuel production is more complex than the simulation of a nuclear power plant and new codes are required for this. Legacy codes lack the extensibility, flexibility and speed required to simulate chemical reactions that appear in a new nuclear application such as nuclear hydrogen production. This problem can be solved by using equation-based simulators, which are simulation platforms designed by chemical engineers to model a large variety of chemical processes. This section explains the main reasons why legacy codes, such as RELAP5, struggle to simulate nuclear hydrogen production. At the same time, this section explains how equation-based simulators can be a good platform to simulate this new kind of nuclear system.

2.2.1 Legacy codes lack simulation features

Legacy codes have been extensively used to demonstrate the safety of nuclear power plants. However, they lack the extensibility, flexibility, speed and accuracy required to represent new nuclear applications such as nuclear hydrogen production.

New nuclear applications are complex systems and their simulation requires models of a variety of physical and chemical phenomena. In the case of nuclear hydrogen production using HTSE or the SI cycle, these phenomena comprise many complex chemical reactions and unit operations. HTSE involves electrolysis cells with temperature gradients, mass transfer and steam electrolysis reactions at the cells electrodes. The SI process involves electrolyte solutions and highly corrosive chemical reactions at high-temperature: Bunsen reaction, sulfuric acid decomposition, and HI decomposition. The dynamics all of these phenomena need to be represented in order to understand the behavior of the hydrogen production plant and how it can affect the nuclear reactor.

However, legacy codes are not designed to model these chemical and physical phenomena. Legacy codes have been created to model accidents in the reactor accurately and specifically. For example, MELCOR, MAAP4 and RELAP5 can represent - with precision - the behavior of the reactor primary loop, the fission reactions in the core, and the core thermodynamics [57]. Yet, these codes do not have the necessary routines to represent the chemical reactions and processes in new nuclear applications such as hydrogen production. Nor do they have algorithms that can efficiently simulate stiff chemical reactions, which are difficult to simulate and can slow down the code's performance.

At the same time, legacy codes are hard to modify to incorporate chemical reactions. In these codes, the equations representing the reactor physics and the solution methods for these equations are tightly coupled. Additionally, these codes are collections of capabilities that have evolved and grown over the years. For example, MELCOR was initially design to be a risk assessment tool; now, it is a dynamic simulation tool [57]. This means that adding new chemical reactions and new unit operations to legacy codes requires a deep understanding of their architecture. Some research groups have attempted this, but their efforts involved years of work and large and specialized teams [43].

2.2.2 Equation-based simulators are a good alternative

Equation-based simulators used in the chemical industry are an alternative platform to legacy codes for the simulation of new nuclear applications. Chemical engineers have to deal with a large number of vastly different processes, which involve chemical reactions, different unit operations, complex thermodynamic models and heat and mass transfer phenomena. To design and optimize these processes, they often use equation-based simulators, which are general-purpose modeling environments for dynamic and steady-state simulation. Such simulators provide a high-level simulation language for the user to describe processes in terms of equations and variables. These simulators and their supporting mathematical theory have been developed for more than 30 years and some examples of them are JACOBIAN [35], Aspen Custom

Modeler [53], and gPROMS [41]. They have been developed to be extensible, flexible, fast and accurate, and this combination of features makes them ideal to simulate new nuclear applications such as nuclear hydrogen production.

The extensibility, flexibility, speed and accuracy of equation-based simulators are achieved by decoupling the description of the physical system from the numerical methods employed to simulate this system. This is a key difference between these simulators and legacy codes, where the definition of the model is tightly coupled to its solution algorithm. On the one hand, a description of the physical system is made by declaring the model, which corresponds to representing all the unit operations individually with equations and variables. These unit operations models can be coded from scratch in the simulation language or they can be imported from a library. The ability to create new unit operation models is very important in the development of new nuclear reactor applications, as the systems have not been completely defined and are constantly changing with the results of new experimental data. On the other hand, the solution of the model is done automatically by powerful general-purpose algorithms. These algorithms assemble all the equations of the model in one single system and then solve this system directly fast and accurately by automatically identifying and exploiting problem structure such as block decomposition. Model declaration and the model solution are explained in the following.

Model declaration—flexibility

Equation-based simulation languages are declarative, which means the user only needs to input the model as a set of differential-algebraic equations (DAEs) [5]. These languages can represent any physical or chemical system as a combination of DAEs:

$$\mathbf{f}(\mathbf{x}(t), \dot{\mathbf{x}}(t), \mathbf{y}(t), t) = \mathbf{0}$$
$$\mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), t) = \mathbf{0}$$

where $\mathbf{x}(t) \in \mathbb{R}^{n_x}$ and $\dot{\mathbf{x}} \in \mathbb{R}^{n_x}$ represent the differential variables (those whose time derivatives appear explicitly in the model) and the time derivatives of these variables,

```
MODEL Vessel_With_Safety_Valve

PARAMETER
  Vessel_Volume,R         AS REAL
  Set_Press, Reseat_Press        AS REAL
  Valve_Const           AS REAL

VARIABLE
  Flow_In, Flow_Out,
  Relief_Flow           AS Molar_Flowrate
  Holdup            AS Moles
  Temp              AS Temperature
  Press, Press_In       AS Pressure

STREAM
  Inlet  : Flow_In, Press_In     AS MainStream
  Outlet : Flow_Out, Press     AS MainStream
  Relief : Relief_Flow, Press    AS MainStream

SELECTOR
  Valve_Flag            AS (Closed,Open)

EQUATION

  # Mass balance
  $Holdup = Flow_In - Flow_Out - Relief_Flow ;

  # Equation of state (perfect gas)
  Press*Vessel_Volume = R*Holdup*Temp ;

  # Safety relief valve with hysteresis
  CASE Valve_Flag OF
    WHEN Closed  :Relief_Flow = 0 ;
                SWITCH TO Open IF Press >= Set_Press ;
    WHEN Open  :Relief_Flow = Valve_Const*Press/Temp^0.5 ;
                  SWITCH TO Closed IF Press <= Reseat_Press ;
  END # CASE

END # Vessel_With_Safety_Valve
```

Figure 2-2: Equation-based simulation languages allow the variables and equations describing physical and chemical phenomena to be declared explicitly [5].

respectively. $\mathbf{y} \in \mathbb{R}^{n_y}$ correspond to the algebraic variables. $\mathbf{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R} \rightarrow \mathbb{R}^{n_x}$ are the differential equations, and $\mathbf{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R} \rightarrow \mathbb{R}^{n_y}$ are the algebraic equations. The purpose is to specify the functional form of a model (e.g., a system of equations), and not a series of statements to be executed in a sequence (Fig. 2-2). The process model so defined is completely decoupled from the numerical methods used to solve it, which allows the user to focus solely on the model.

Furthermore, equation-based simulation languages use hierarchical decomposition, which greatly simplifies the coding of large models [5]. Hierarchical decomposition allows building models in levels—in the same way one would build a real plant. For example, a first level of simple models can represent basic units (e.g., pipes, heat exchangers, chemical reactors). A second level connects models from the first to

represent subsystems (e.g., heat transfer loop). Finally, a third level could connect all the subsystems to assemble a plant model. With this approach, lower-level models can be easily tested and corrected, avoiding hard-to-find errors in higher-level models.

In sum, equation declaration and hierarchical decomposition allow equation-based simulation languages to represent the details of new nuclear applications in a simple and intuitive way.

Model solution—speed and accuracy

Equally important, equation-based simulators find solutions to simulations quickly and accurately. To solve simulations quickly, they assemble all the model equations into one single equation system, which is solved simultaneously by general-purpose codes for root finding and fast numerical integration. They also automatically exploit the characteristics of the equation system (e.g., sparsity, analytical derivatives, and block decomposition) to accelerate the solution process and solve 100,000s of equations in minutes. To be accurate, they use error control based on theoretical guarantees for accuracy and stability, and these estimates are controlled automatically [4]. These features are made possible by powerful algorithms: implicit predictor-corrector methods for integration [3], automatic differentiation algorithms [16] and algorithms for sparse linear algebra [9]. These algorithms are explained here.

The best equation-based simulators use implicit predictor-corrector methods to integrate the system of equations numerically. They are implicit methods that use an explicit method to calculate an initial guess for the solution of the system of equations at each time step. The implicit methods normally used are multi-step methods, and, in particular, variable step size, variable order Backward Differentiation Formula (BDF) methods are used [15]. These methods are very stable and can solve stiff problems such as the simulation of complex systems of chemical reactions. Additionally, the step size used in the integration is controlled by the truncation error rather than the stability. Thus, the integrator can take larger time steps when the system of equations is stiff and the fast modes of the transient have decayed. This results in a faster solution of the simulation.

Automatic differentiation algorithms calculate the Jacobian (matrix of partial derivatives) of the equations analytically. These algorithms increase the speed and accuracy of the solution, because predictor-corrector methods work more efficiently with an analytical Jacobian instead of a numerical one. At the same time, these automatic algorithms avoid the tedious and prone-to-error task of deriving the analytical Jacobian manually. For these reasons, automatic differentiation algorithms are extensively used in equation-based languages [16, 54]

Algorithms for sparse linear algebra take advantage of the sparsity of an equation system to solve it faster. Normally, systems with more than about 20 equations will have only a few variables appearing in each equation. This creates a very sparse system, whose Jacobian will have mostly zeros. Using this information in numerical Jacobian evaluations and while solving a system's linear equations greatly increases the solution speed of the equation system.

2.3 Nuclear hydrogen plant used in this thesis

The goal of this thesis is to study the safety of a nuclear hydrogen production plant by simulating this plant's operation in an equation-based simulator. The plant selected for this is a combination of well-developed technologies: two modular pebble bed reactors (MPBR), a power conversion unit (PCU) using the Brayton cycle, a heat transfer loop, and a HTSE facility (Fig. 2-3). In this process, the nuclear reactor produces 500 MWth of high-temperature heat for the PCU and for the HTSE process. The PCU uses 450 MWth to produce the electricity needed in the electrolysis cell of the HTSE unit. The remaining 50 MWth of heat is transferred from the nuclear reactors to the HTSE unit by the heat transfer loop, and it is used to produce superheated steam (800-900°C). This steam is fed to the electrolysis cell where it is decomposed into oxygen and hydrogen; this reaction occurs at high temperature, which increases the efficiency of the electrolysis. The oxygen is recuperated and the hydrogen/water stream is sent to a steam separator, to be recover hydrogen.

The specific design for the MPBR and PCU was adapted from the pebble bed

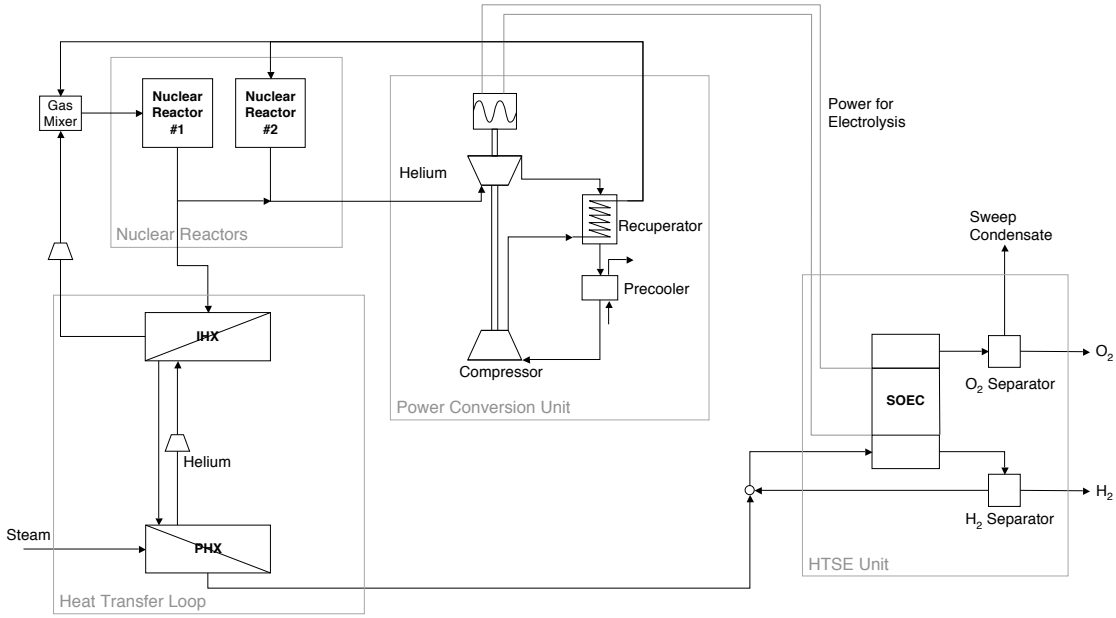

Figure 2-3: The nuclear hydrogen production used in this thesis is a combination of technologies in development.

reactor research at MIT [60, 23]. The HTSE facility and the heat tranfer loop are modeled after ones developed at INL [32, 8]. This section explains how each of these components work.

2.3.1 PBMR and PCU Operation

As explained before, nuclear hydrogen production requires very high-temperature heat (over 750°C) and electricity and different reactors/PCU combinations have been studied for this (Tables 2.1, 2.2). From these technologies, the modular pebble bed reactor technology is most promising and the one chosen for this study. Pebble bed reactors (PBR) are an attractive choice because their design provides excellent economics and safety. Also, good progress has been made in their development: Germany built two PBRs in the 1970s and 1980s, China built a 10-MWth PBR as a test and is developing a nuclear plant with two 250-MWth PBRs, and South Africa is developing a 268-MWth PBR.

In particular, this study will use the design developed at MIT called the "Modular

Pebble Bed Reactor". (MPBR) coupled to a helium Brayton cycle [60]. This design has a capacity of 250 MWth, it is based on the South African PBR and it exploits modularity to decrease production and construction costs. Two MPBRs are used in the nuclear hydrogen production plant (Fig. 2-3), which is a similar modular configuration to the new PBR project developed in China [65]. Some of the key features of the MPBR to achieve safety and good economics are the fuel pebbles used in the core, the primary coolant system and the Brayton cycle used in the PCU. These features are explained in the following.

The core in the MPBR uses fuel pebbles to simplify the reactor operation. The MPBR's fuel system is composed of 360,000 6-cm-diameter graphite pebbles containing the uranium fuel in microspheres (Fig. 2-4). These pebbles can be introduced at the top of the reactor and extracted at the bottom during operation. This online refueling capability avoids the breaks needed to refuel normal reactors, increasing the uptime and economics.

Additionally, the MPBR's fuel is designed to avoid meltdowns passively and to have a lower possibility of radiation spread. To avoid meltdowns, these reactors use fuel particles with low uranium density that cannot melt down during operational upsets or accidents. The fuel particles are designed to operate at temperatures over 1000°C, where a temperature increase results in the Doppler broadening of the fuel isotope's resonance-absorption line. As a consequence, the fission rate decreases—negative reactivity—because U-238 can attach more neutrons and less neutrons are available for the fission of U-235. The negative reactivity limits the maximum temperature the fuel can reach to 1600°C, which is well below the fusion point of the fuel (2000°C). In addition, the fuel particles are designed to feature high retention capability of fission products (Fig. 2-4), which decreases radiation spread in the case of an accident.

The coolant loop in the pebble bed reactor uses helium to decrease reactor maintenance and to decrease the possibility of radiation spread even further. Helium is an inert gas and does not corrode the components in the nuclear reactor coolant system, as water does in water-cooled reactors. Using helium, thus, avoids the cost of replac-

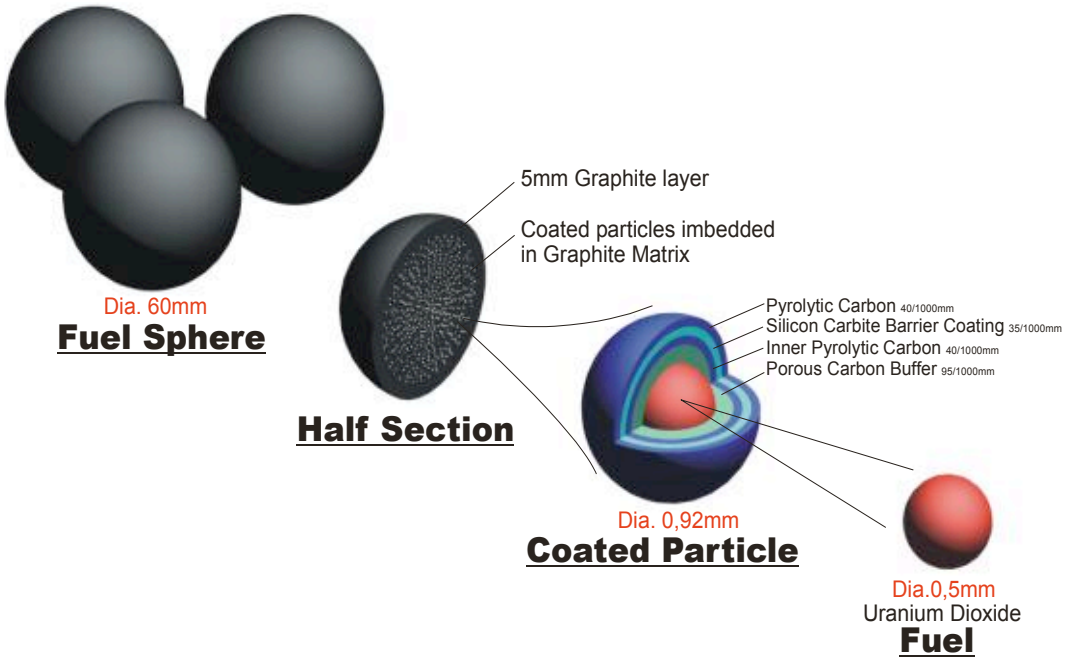

Figure 2-4: The design of fuel pebbles for PBRs avoids meltdowns by having a high fusion point (2000°C) and decreases the possibility of radiation spread [26].

ing degraded components and stopping the plant for this kind of maintenance. At the same time, helium is not transmuted to radioactive elements, which helps reduce even further the possibility of radiation dissemination in the case of an accident.

The PCU coupled to the MPBRs uses a Brayton cycle to achieve high efficiency in the production of electricity. Water-cooled reactors such as PWRs use a Rankine cycle to produce electricity, which achieves approximately 32% of power cycle efficiency. The PCU considered in this thesis, uses a Brayton cycle with helium as a coolant instead, which can achieve power cycle efficiencies of about 45%. This PCU is composed of a turbine, a compressor, a generator, a recuperator and a heater.

2.3.2 HTSE

As mentioned before, high-temperature steam electrolysis (HTSE) is an efficient way to generate hydrogen from water. One of the leading technologies being developed to do this is the solid oxide electrolysis cell (SOEC), which is based on ceramic fuel cell technology. This technology is being developed by the Idaho National Labora-

tory (INL) in collaboration with Ceramatec [8]. INL has made a lot of progress in improving cell performance and reducing cell degradation [51]. For these reasons this technology has been selected for this thesis.

The HTSE unit generates hydrogen and oxygen from water by using high-temperature heat from the nuclear reactor and electricity produced in the PCU. First, water is heated by recovered heat from product streams, and later is superheated in the process heat exchanger (PHX) (Fig. 2-3). Then, the superheated steam goes into the electrolysis cell, where the voltage applied to the cell splits water molecules into H_2 and O^{2-} at the cathode (Fig. 2-5). The O^{2-} so generated migrates through the ion-conducting solid electrolyte to the porous anode to produce O_2. As a result, the electrolysis cell dissociates the superheated steam and separates the products at the same time. The use of superheated steam allows the use of a lower voltage than the voltage needed in conventional electrolyzers, and it increases the speed of the reactions in the system.

It is important to note that the ceramic materials are very sensitive to thermal gradients, so temperature increases in the cells need to be gradual to avoid them. In the case of the cell tested at INL, the rate of temperature increase is 1°C/min. This point has to be taken into account when planning start-up and shutdown procedures.

The HTSE unit is built in a modular form. Its minimal constituent is the SOEC (10 cm x 10 cm x 3 mm). These cells are arranged in stacks of 500 cells, which then are arranged in 40 stack modules, which are finally laid out in containers containing eight modules each (Fig. 2-6). Approximately 63 of these eight-module containers would be required for the system considered in this study [42].

2.3.3 Heat transfer loop

The heat transfer loop transfers 50 MWth of heat from the nuclear reactor to the HTSE unit over a distance of 90 m. In this way, the heat transfer loop couples the nuclear reactor with the HTSE unit and, at the same time, provides safe physical separation between them. A detailed design of this loop has been developed at INL [8], and it will be the base for the loop used in this thesis.

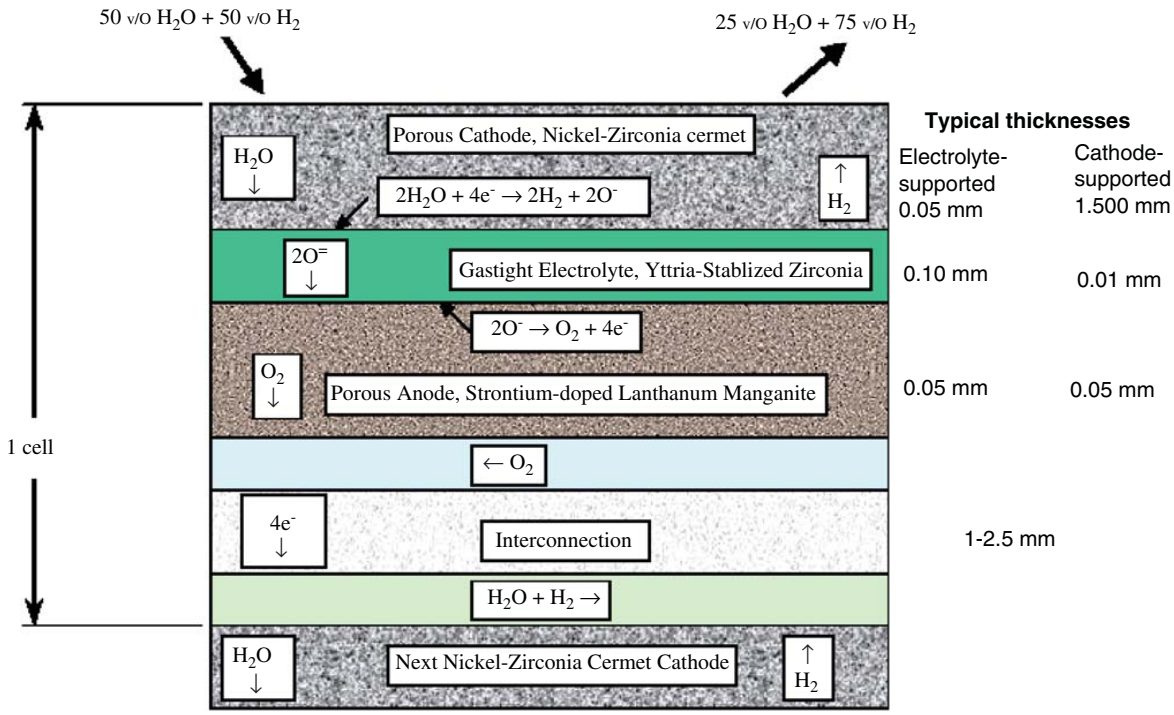

Figure 2-5: The SOEC cell dissoaciates superheated steam into hydrogen and oxygen [36].

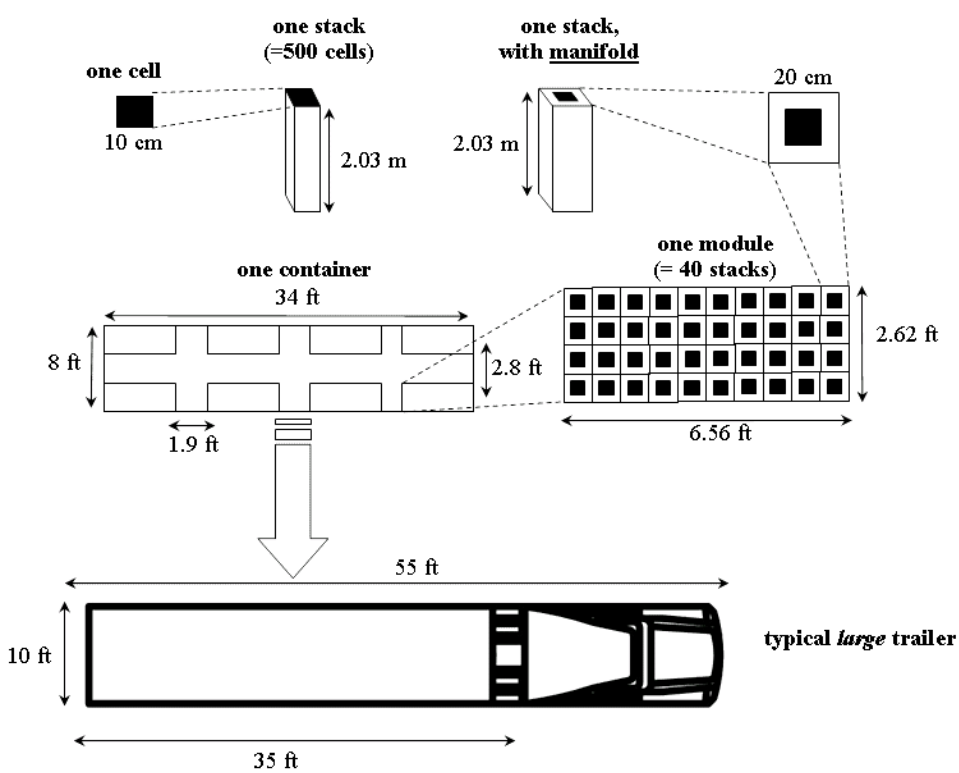

Figure 2-6: A HTSE plant is designed to be built in modular form [42].

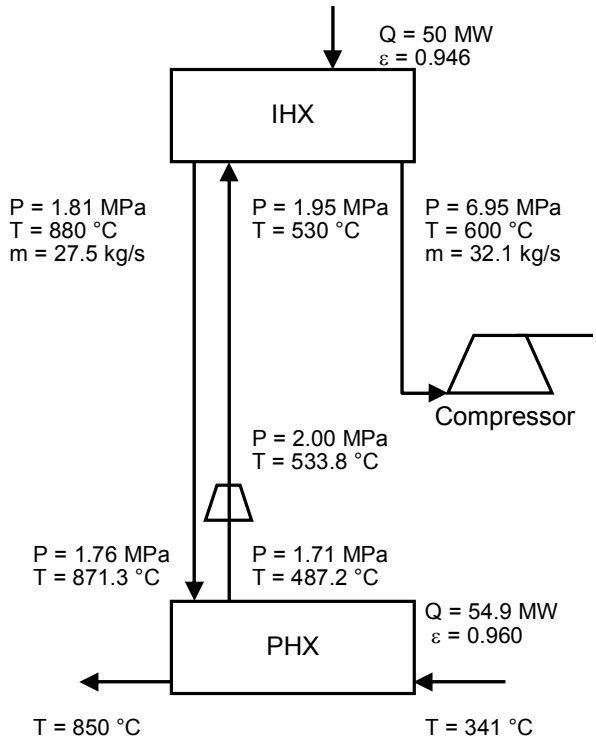

Figure 2-7: Heat transfer loop [8].

The heat transfer loop uses helium at 2 MPa and is composed of an intermediate heat exchanger (IHX), a hot leg, a process heat exchanger (PHX), a cold leg and a compressor. The IHX transfers 50 MWth from the nuclear reactor coolant at 900°C to the helium in the loop, which reaches 880°C (Fig. 2-3). Its design is based on the printed circuit heat exchanger (PCHE) technology developed at Heatric [8]. It is connected by the hot leg to the process heat exchanger (PHX), where steam is superheated to feed the electrolysis cells. The PHX is a tube-in-shell heat exchanger and the helium flows on the shell side. At the end of it, the colder helium (487.2°C) is pressurized by a compressor and then taken back to the IHX by the cold leg (Fig. 2-7).

Chapter 3

Dynamic simulation of a heat transfer loop for alternative nuclear reactor applications

3.1 Introduction

Using nuclear reactors as an energy source is an attractive alternative to reduce CO_2 emissions in energy-intensive chemical processes (e.g., oil refining, biofuels production). These processes need carbon-based fuels to provide energy and chemicals for plant operation, thus generating large amounts of CO_2. Using a nuclear reactor for these support tasks can eliminate the CO_2 emissions, and several new processes have been proposed for this [12, 2].

For example, nuclear reactors could be used for hydrogen production, which is fundamental for oil refining and other processes. Hydrogen production is currently based on steam methane reforming and releases 60 Mt of CO_2 per year in the U.S., or almost 1% of the total CO_2 emissions [55, 1]. This process could be replaced by a high-temperature nuclear reactor coupled to a hydrogen production plant [63]. In this system, the nuclear reactor provides heat and power to a solid oxide electrolysis cell (SOEC) to run high-temperature electrolysis reactions. Heat is transferred to the

SOEC by a heat transfer loop, which also provides safe separation between the nuclear reactor and the chemical plant. The rest of the heat generated by the nuclear reactor is used by the power conversion unit (PCU) to produce electricity, which powers the electrolysis in the SOEC (Fig. 3-1).

Safety and operability of these new nuclear reactor applications need to be demonstrated; in particular, the connection between the reactor and the chemical plant (the heat transfer loop) requires special attention. These new applications involve extreme conditions (e.g., high temperatures (900°C), high pressures (7 MPa), exothermic chemical reactions, and corrossive chemicals) that could cause accidents, make operation more difficult, and require exotic materials of construction. To ensure safety, the system design should prevent or isolate operational disruptions that might affect the nuclear reactor (e.g., loss of heat sink capacity, accidents). To ensure operability, the system design should be demonstrated to perform well during operational transients (e.g., start-up, shut-down, re-start). One of the key elements to achieve these goals is the heat transfer loop connecting the nuclear reactor and the chemical plant (Fig. 3-1). This loop is a new piece of engineering and presents several design challenges. It will incorporate new technologies and materials to handle the extreme conditions, and it will isolate the nuclear reactor from the chemical plant. Careful study of its performance will pave the road for the new nuclear applications.

The response of novel nuclear systems to accidents will be simulated and studied with equation-based dynamic simulators. As mentioned above, a novel system must behave safely during upsets and operational transients, and computational simulations of the system can provide some a priori assurance of this. Usually, nuclear reactor designs are tested by computational simulations in legacy simulation codes (e.g., RELAP [52]). Unfortunately, these codes cannot simulate the chemical plant because they lack flexibility, computational speed and adequate algorithms to represent chemical reactions. These problems can be overcome by using equation-based dynamic simulators from the chemical engineering field (e.g., JACOBIAN [35], Aspen Custom Modeler [53], gPROMS [41]). These simulators can easily simulate chemical reactions and they have already been used to simulate the nuclear reactor and PCU

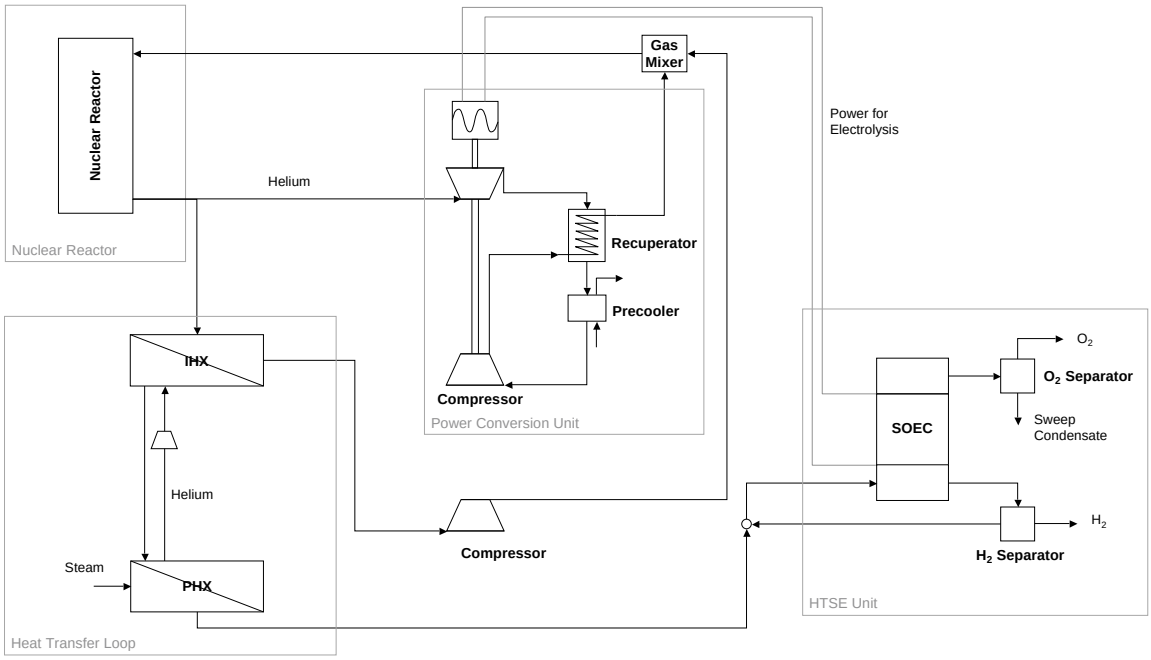

Figure 3-1: Nuclear hydrogen production is an example of novel nuclear applications. The design of the heat transfer loop requires special attention.

[24].

However, the dynamics of the heat transfer loop have not yet been studied in equation-based simulators and its simulation is technically challenging. Several studies have investigated the loop's operating point at steady state [63, 8, 38], but the work on the transient simulation front is not as well developed [59]. In addition, the use of equation-based simulators is not widespread in the nuclear engineering community, which makes finding similar studies even harder. At the same time, the simulation of the heat transfer loop using helium as a heat transfer fluid is difficult, because simulating gas dynamics rigorously involves a complex set of hyperbolic equations [25]. These equations normally require complex algorithms for their numerical integration and it is not clear if equation-based simulators can solve them accurately and efficiently [17].

This chapter, then, proposes a set of models to represent the dynamics of a heat transfer loop in equation-based simulators successfully. First, it explains equation-based simulators and their advantages. Second, it presents a description of the heat

transfer loop, six models to simulate the loop in equation-based simulators, and a reference model created in a legacy code (RELAP). Finally, recommendations concerning the best models are made by evaluating their performance and advantages.

3.2 Using equation-based languages for the simulation of novel nuclear applications

Chemical engineers have to deal routinely with a large number of vastly different processes, which involve chemical reactions, and heat and mass transfer phenomena. To design and optimize these processes, the use of equation-based simulators has become widespread; these simulators are general-purpose-modeling environments for dynamic and steady-state simulation. Such simulators provide a high-level simulation language for the user to describe process models in terms of equations and variables [5]. Then, they solve the equations automatically by using powerful general-purpose numerical algorithms. These simulators and their supporting mathematical theory have been developed over more than 30 years and some examples of them are JACOBIAN [35], Aspen Custom Modeler [53], and gPROMS [41].

Equation-based simulators are ideal to simulate nuclear reactors coupled to chemical plants because they are flexible, extensible, computationally fast and accurate. On the other hand, their use does present some challenges, for formulating large systems of equations that are mathematically correct is not easy. Yet they make these challenges less burdensome by providing error-checking algorithms to address them.

3.2.1 Equation-based simulators provide flexibility, speed and accuracy

Equation-based simulators come with simulation languages that are declarative and hierarchical, which makes them able to incorporate new user-defined models and different processes easily. They also have powerful general-purpose numerical algorithms to simulate any new models, which allow them to solve simulations quickly

and accurately.

A declarative language means the user only needs to input the model as differential-algebraic equations (DAEs) [5]. These languages can represent any physical or chemical system as a system of DAEs [3]:

$$\mathbf{f}(\mathbf{x}(t), \dot{\mathbf{x}}(t), \mathbf{y}(t), t) = \mathbf{0}$$
$$\mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), t) = \mathbf{0}$$

where $\mathbf{x}(t) \in \mathbb{R}^{n_x}$ and $\dot{\mathbf{x}}(t) \in \mathbb{R}^{n_x}$ represent the differential variables (those whose time derivatives appear explicitly in the model) and the time derivatives of these variables, respectively. $\mathbf{y}(t) \in \mathbb{R}^{n_y}$ correspond to the algebraic variables. $\mathbf{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R} \rightarrow \mathbb{R}^{n_x}$ are the differential equations, and $\mathbf{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R} \rightarrow \mathbb{R}^{n_y}$ are the algebraic equations. Although this formulation of DAEs is not the most general one, it is enough to represent most physical and chemical systems. The philosophy of an equation-based simulator is that the process model so defined is completely decoupled from the numerical methods used to solve it, which allows the user focus solely on the model formulation.

Furthermore, declarative simulation languages support hierarchical model decomposition, which greatly simplifies the coding of large models [5]. It allows building models in levels in the same way one would conceptualize a real plant. For example, a first level of simple models can represent basic units (e.g., pipes, heat exchangers, chemical reactors). A second level uses models from the first to represent subsystems (e.g., heat transfer loop). Finally, a third level could incorporate all the subsystems to assemble a plant-wide model. With this approach, lower-level models can be easily tested and corrected in isolation, avoiding hard-to-find errors in higher-level models.

Equally important, equation-based simulators find solutions to simulations quickly and accurately. They assemble the equations from all the different models in one single equation system, which is solved simultaneously by general-purpose codes for root finding and implicit numerical integration. They also exploit the characteristics of the equation system (e.g., sparsity, block decomposition) to accelerate the simulations

and solve 100,000s of equations in minutes. They are accurate too, because their error control is based on theoretical guarantees for consistency and stability, and the error can be controlled automatically.

3.2.2 Simulation challenges in equation-based simulators are minimal and mostly related to initial conditions

Equation-based simulators' flexibility, extensibility and accuracy come from using large DAE systems to represent models and from solving these systems simultaneously. However, this approach forces the user to define the system of equations, initial conditions, and parameters correctly, which is hard to accomplish for large and complex models. To address this difficulty, equation-based simulators use automatic error-checking algorithms to help the user find errors.

These automatic error-checking algorithms help verify that the system of equations is properly posed. As the number of unit operations and the number of equations in a model grow, the user finds it harder to detect missing or unnecessary equations. To assist with this task, algorithms examine the degrees of freedom of the system. This ensures the number of equations (neq) and the number of unknown variables (nev) are equal. Furthermore, sometimes a system with $neq = nev$ might have inconsistent equations (e.g., linearly dependent equations) that can render the equation system infeasible. This problem can be often be resolved by algorithms that analyze the structural singularity of the system of equations and reveal the equations with conflicting information [40].

In addition, error-checking algorithms test whether the formulated initial conditions are consistent. Not having consistent initial conditions can lead to failed simulations or results that are nonsense. Normally, finding consistent initial conditions for a simulation is a straightforward task; consider the following system of equations

[30]:

$$\dot{x}_1 = x_1 + x_2 + y, \tag{3.1}$$

$$\dot{x}_2 = x_1 - x_2 - y, \tag{3.2}$$

$$0 = x_1 + 2x_2 - y. \tag{3.3}$$

In this example, $\dot{x}_1(0)$, $\dot{x}_2(0)$, $y(0)$, $x_1(0)$ and $x_2(0)$ are the initial values needed to start the numerical integration for dynamic simulation at $t = 0$. It is simple to calculate them: any two of these variables are assigned initial values by the user and the remaining three are calculated from solving the system of equations at $t = 0$. Equation-based simulators do this automatically, thus guaranteeing that the initial condition of the simulation is consistent.

In a few cases though, error-checking algorithms can struggle to find a consistent set of initial conditions. Consider the system [30]:

$$\dot{x}_1 = x_1 + x_2 + y, \tag{3.4}$$

$$\dot{x}_2 = x_1 - x_2 - y, \tag{3.5}$$

$$0 = x_1 + 2x_2, \tag{3.6}$$

which has a similar form to the system of Eqs. (3.1)-(3.3). However, one cannot assign arbitrary values to any pair of the unknowns as before, because Eq. (3.6) imposes a very strong restriction. For example, one cannot set both $x_1(0)$ and $x_2(0)$ because they are related by Eq. (3.6). The issue is caused by an additional nonredundant "hidden" constraint derived by differentiating Eq. (3.6) with respect to time:

$$0 = \dot{x}_1 + 2\dot{x}_2. \tag{3.7}$$

So, in fact there are four equations relating the initial values of the five variables. One realizes that only one variable can be set by the user before solving for consistent initial conditions. This type of problem is referred to informally as a "high-index

problem," In general, DAEs are categorized according to their "index" [3] and DAEs with index greater than one will exhibit the following two features:

1. Additional "hidden" constraints on consistent initial conditions, as in the second example above.

2. Dependence of the solution on first and higher-order derivatives of the forcing functions.

In addition, error-controlled numerical integration of DAEs with index greater than one is not possible with standard codes, and correct numerical integration is only possible with highly-specialized algorithms specific to certain problem structures. The first example above is index 1, whereas the second example is index 2.

On the other hand, DAEs of index 1 can be solved reliably with standard codes, and the issue with additional hidden constraints on consistent initial conditions only occurs rarely. Moreover, most models in practical applications are index 1, with higher-index problems arising from modeling errors (which can be corrected) or modeling approximations (which can be relaxed). Since most equation-based simulators only support consistent initialization and numerical integration of index 1 DAEs, they apply an automatic structural analysis to detect high-index problems with the purpose of screening out DAEs with index greater than one. The first example above would pass this test, whereas the second example above would fail the test. Some simulators can provide additional feedback that Eq. (3.6) is causing the high-index problem.

3.3 Modeling the heat transfer loop

Since equation-based simulators are good environments to simulate new nuclear applications, the next step is to use them to model the heat transfer loop. We chose to model a heat transfer loop engineered at the Idaho National Laboratory (INL) [8], because of its well-thought-out design. This heat transfer loop consists of an intermediate heat exchanger (IHX), a process heat exchanger (PHX), a hot leg, a cold leg,

and a compressor (Fig. 3-1). It transfers 50 MWth from the nuclear reactor to the chemical plant, and it has to withstand very high temperatures (up to 900°C) and high pressures (up to 7 MPa) [8]. These challenging conditions require brand new design with new materials and new pieces of equipment.

Researchers at INL determined the following equipment characteristics for the heat transfer loop:

1. The IHX is a compact heat exchanger made of Alloy 617 with 786,560 semi-circular channels of 1.5 mm in diameter. The length of the IHX was 0.988 m and its width was 1.44 m.

2. The PHX is a tube-in-shell heat exchanger with 3,500 circular tubes, and each tube has an internal diameter of 10 mm. High-temperature helium flows on the inside of the tubes and the steam flows on the outside. The length of the exchanger is 8.66 m and its material is Alloy 800.

3. The hot leg and cold leg are pipes of 90 m in length with internal diameters of 0.44 m and 0.41 m, thickness-to-diameter ratio of 0.117 and 0.114, and insulation thickness 59 mm and 58 mm, respectively. Both of them are made of Alloy 800.

4. The compressor should be able to produce a pressure differential of at least 0.4 MPa.

These specifications were used to build the dynamic models for the loop in JACOBIAN—one of the best equation-based simulators available [35]. To confirm the simulation results of this model, a similar model was created using RELAP–a legacy code [8].

3.3.1 Dynamic models in JACOBIAN

The models created in JACOBIAN must contain equations for each physical phenomena considered in the loop. This was done by creating individual models containing these equations for the different parts of the loop: pipes, heat exchangers, and compressor:

1. *Pipe model.* As a first approximation, radial variations inside the pipe can be neglected, and one-dimensional models can be used to represent the gas dynamics. Three models were used for this task, and they were derived from the *Navier-Stokes* equations. The *Navier-Stokes* equations are a realistic approach to represent gas dynamics [25]; they consider mass, momentum and energy balances and they include viscosity, friction and heat transfer effects. However, in the case of pipes filled with helium, kinetic and viscosity terms in the energy equations, and viscosity terms in the momentum equation are negligible. Eliminating those terms produces the following system of equations, which will be termed the *Full Dynamics* model:

$$\text{Continuity} \qquad \frac{\partial \rho}{\partial t} = -\frac{\partial \phi}{\partial x}, \tag{3.8}$$

$$\text{Momentum} \qquad \frac{\partial \phi}{\partial t} = -\frac{\partial P}{\partial x} - \frac{\partial}{\partial x}\frac{\phi^2}{\rho} - \frac{2f\left|\phi\right|\phi}{D\rho}, \tag{3.9}$$

$$\text{Energy} \qquad \frac{\partial \psi}{\partial t} = -\frac{\partial}{\partial x}\phi h - \frac{4U\left(T - T_{Ext}\right)}{D}, \tag{3.10}$$

$$\text{Physical Properties} \qquad \psi = \rho h - P, \tag{3.11}$$

$$h = h_{form} + \frac{C_P}{M}\left(T - T_{stand}\right), \tag{3.12}$$

$$P = \frac{R_C \rho T}{M}. \tag{3.13}$$

Here ρ is the helium density, ϕ is the mass flux, P is the pressure, ψ is the internal energy, h is the enthalpy, T is the gas temperature, and T_{Ext} is the temperature surrounding the pipe. The parameters are the friction factor f, the diameter of the pipe D, the heat transfer coefficient U, the universal constant R_C, the enthalpy of formation for helium h_{form}, the heat capacity at constant pressure C_P, molecular weight M, and the standard temperature T_{stand}. Note that these equations are actually partial differential-algebraic equations (PDAEs), which can also be categorized according to their index and can exhibit similar issues with consistent initial conditions [29]. On semi-discretization of the spatial derivatives in PDAEs using the method of lines [47] a system of DAEs results.

Simplified versions of these equations were also used, in order to find faster alternative models. The *Full Dynamics* model can represent the fast time scales of gas dynamics, which means some changes in the variables can travel with the speed of sound. This makes the integration of such a model inefficient and inaccurate, because very small time steps are needed to track the variables correctly and artificial oscillations might appear depending on the spatial discretization employed. Fortunately, these fast time scales can be excluded, since the velocity in the loop is below the speed of sound and fast-moving effects become less important. In addition, those time scales are not relevant to the goal of this study, as we are interested in the longer time scale (over 0.5 s) behavior of the coupled system. To eliminate the fast phenomena, we formulated simpler models for the differential equations: the *Zero Mach Number Limit* model and the *Quasi-Steady State (QSS) Approximation* model.

The *Zero Mach Number Limit* model eliminates the accumulation term in the momentum equation [17]:

$$\text{Continuity} \qquad \frac{\partial \rho}{\partial t} = -\frac{\partial \phi}{\partial x}, \tag{3.14}$$

$$\text{Momentum} \qquad 0 = -\frac{\partial P}{\partial x} - \frac{\partial}{\partial x}\frac{\phi^2}{\rho} - \frac{2f\,|\phi|\,\phi}{D\rho}, \tag{3.15}$$

$$\text{Energy} \qquad \frac{\partial \psi}{\partial t} = -\frac{\partial}{\partial x}\phi h - \frac{4U\,(T - T_{Ext})}{D}. \tag{3.16}$$

The *QSS Approximation* model additionally eliminates the accumulation term in the continuity equation:

$$\text{Continuity} \qquad 0 = -\frac{\partial \phi}{\partial x}, \tag{3.17}$$

$$\text{Momentum} \qquad 0 = -\frac{\partial P}{\partial x} - \frac{\partial}{\partial x}\frac{\phi^2}{\rho} - \frac{2f\,|\phi|\,\phi}{D\rho}, \tag{3.18}$$

$$\text{Energy} \qquad \frac{\partial \psi}{\partial t} = -\frac{\partial}{\partial x}\phi h - \frac{4U\,(T - T_{Ext})}{D}. \tag{3.19}$$

The equations in the different models were discretized using a control volume method with a combination of a standard grid and a staggered grid (Fig. (A-1)).

The standard grid divides each section of the loop (cold leg, hot leg, PHX, IHX) into control volumes, which are used to calculate ρ, P, ψ, h and T. The staggered grid divides the loop into control volumes too, but it centers these control volumes at the faces of the control volumes in the standard grid. This grid is used to calculate the flowrate variable F, which replaces the mass flux variable ϕ in the discretized equations ($F = \phi A$; A is the total cross sectional area in pipes or heat exchangers). F is used instead of ϕ because the equations produced in this way are simpler to write and do not lose accuracy. Using this combination of grids generates a system of equations that is more stable to integrate [39]. This approach has been used to simulate gas dynamics in the past [24, 49]. The heat transfer coefficient at each control volume in the standard grid were calculated using the Dittus-Boelter equation according to the specifications from Davis et al. [8].

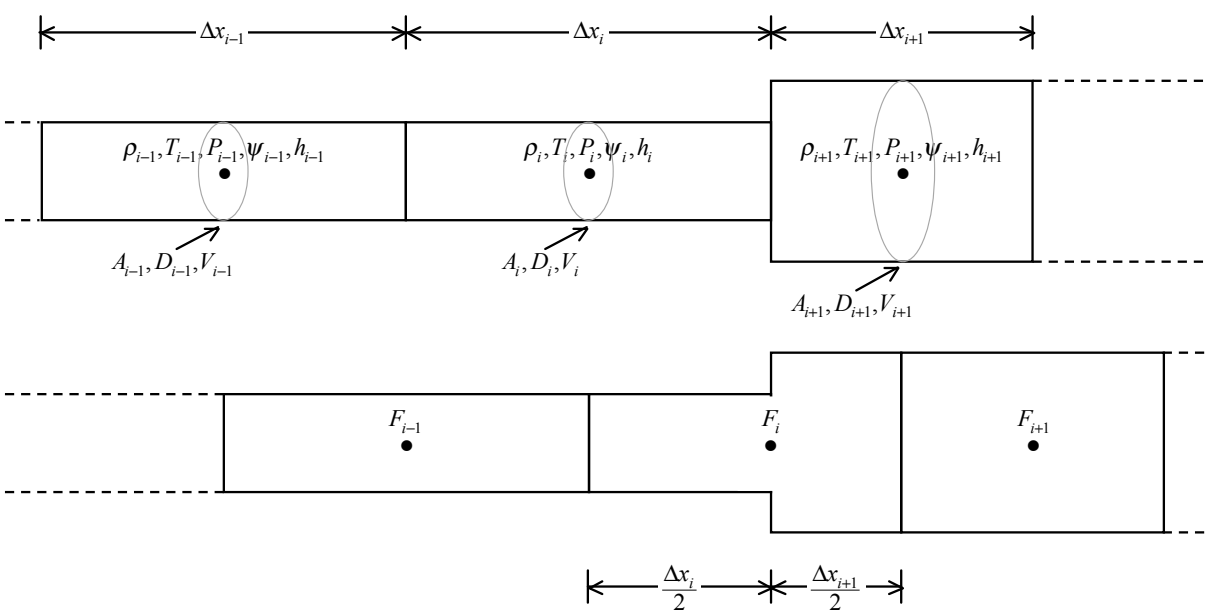

Figure 3-2: Discretizing the gas dynamics equations using combination of standard grid and a staggered grid makes the solution of the system more stable.

The discretized equations are written for the case of a change in cross sectional area in the loop, which is the most difficult case to simulate. The equations so written can also represent the case of constant cross sectional area in the loop.

The list of equations is:

- Continuity

$$\Delta x_i A_i \dot{\rho}_i = F_{i-1} - F_i \qquad i = 2 \ldots N. \tag{3.20}$$

- Momentum

$$\begin{aligned}
&\left(\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}\right)\dot{F}_i = \\
&-\left(P_{i+1}A_{i+1} - \frac{P_i\frac{\Delta x_{i+1}}{2} + P_{i+1}\frac{\Delta x_i}{2}}{\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}}(A_{i+1} - A_i) - P_i A_i\right) \\
&-\left(\frac{\left(\frac{F_i+F_{i+1}}{2}\right)^2}{\rho_{i+1}A_{i+1}} - \frac{\left(\frac{F_{i-1}+F_i}{2}\right)^2}{\rho_i A_i}\right) \\
&-\left(\frac{2f_i}{D_i}\frac{F_i^2}{\rho_i A_i}\frac{\Delta x_i}{2} + \frac{2f_{i+1}}{D_{i+1}}\frac{F_i^2}{\rho_{i+1}A_{i+1}}\frac{\Delta x_{i+1}}{2}\right) \\
&i = 2 \ldots N-1.
\end{aligned} \tag{3.21}$$

- Energy

$$\Delta x_i A_i \dot{\psi}_i = -\left(F_i h_i - F_{i-1}h_{i-1}\right) - \frac{4U_i}{D_i}\left(T_i - T_{Ext,i}\right)\Delta x_i A_i \quad i = 2 \ldots N. \tag{3.22}$$

- Algebraic equations

$$\psi_i = \rho_i h_i - P_i \qquad i = 1 \ldots N, \tag{3.23}$$

$$h_i = h_{form} + \frac{Cp}{M}\left(T_i - T_{stand}\right) \qquad i = 1 \ldots N, \tag{3.24}$$

$$P_i = \frac{R_C \rho_i T_i}{M} \qquad i = 1 \ldots N. \tag{3.25}$$

Analogous discretizations were used for the *Zero Mach Number Limit* model and for the *QSS Approximation* model.

2. *Compressor model.* At this time, our focus is on the thermal behavior of the heat

transfer loop and not on the compressor dynamics. Therefore, the compressor was represented by a simple adiabatic equation (Eq. (3.26)). In addition, the compressor equations included a mass conservation equation (Eq. (3.27)). This equation forces the mass flow entering the compressor to be the same as that leaving it:

$$\left(\frac{P_{In}}{P_{Out}}\right)^{\gamma-1} = \left(\frac{T_{In}}{T_{Out}}\right)^{\gamma}, \tag{3.26}$$

$$A_{In}\phi_{In} = A_{Out}\phi_{Out}. \tag{3.27}$$

These equations not only define the compressor, but they also define the coupling conditions for the boundary variables of the previous pipe model when used to model the loop. Two combinations of boundary conditions were used to write the coupling conditions.

The first combination of boundary conditions consists of the pressure and temperature at the left boundary and the pressure at the right boundary (P_{In}, T_{In} , P_{Out}) (Fig. 3-3). The pressure as a boundary condition at the left is normally used in the simulation of fluid dynamics [27]. In this case, the flow in the compressor was regulated by specifying the pressure differential in the compressor ($\Delta P(t)$). This approach was used for the loop and for the external pipes in heat exchangers. In addition, two extra control volumes were inserted after the compressor to model the effects of the change in cross sectional area at the compressor. These control volumes are frictionless, adiabatic and short (1 cm); therefore, they do not alter the simulation.

The differential equations that represent these coupling conditions are:

$$\Delta x_{Aux} A_N \dot{\rho}_{Aux,1} = (F_{In} - F_{Aux,1}), \tag{3.28}$$

$$\Delta x_{Aux} A_N \dot{\rho}_{Aux,2} = (F_{Aux,1} - F_{Aux,2}), \tag{3.29}$$

$$\Delta x_1 A_1 \dot{\rho}_1 = (F_{Aux,2} - F_1), \tag{3.30}$$

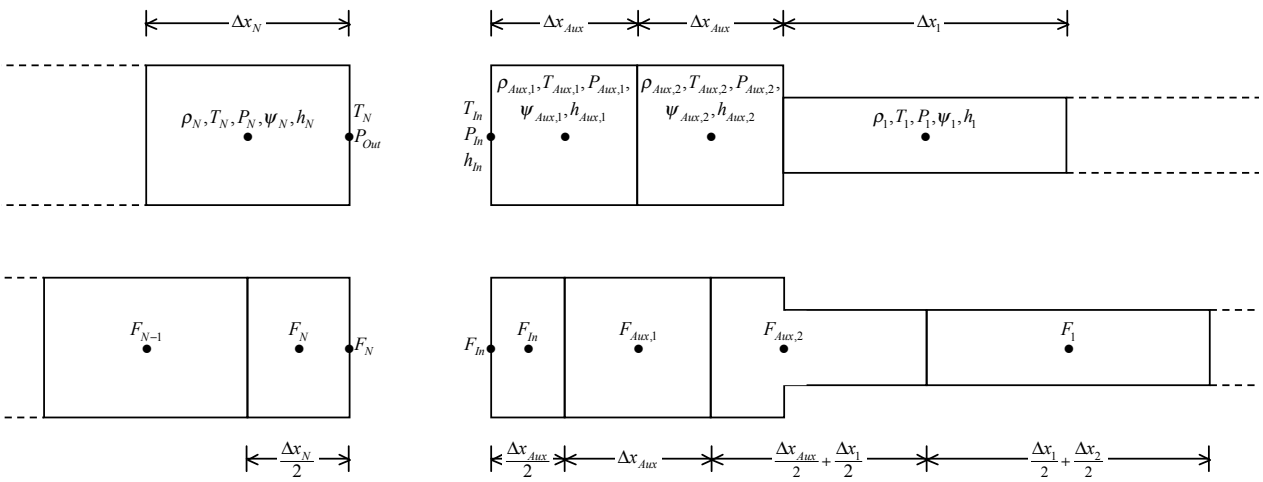

Figure 3-3: One approach to set the boundary variables; it uses P_{In} as suggested in the literature [27].

$$\frac{\Delta x_{Aux}}{2}\dot{F}_{In} = -\left(P_{Aux,1}A_N - P_{In}A_N\right) - \left(\frac{\left(\frac{F_{In}+F_{Aux,1}}{2}\right)^2}{\rho_{Aux,1}A_N} - \frac{F_{In}^2}{\rho_{Aux,1}A_N}\right), \tag{3.31}$$

$$\Delta x_{Aux}\dot{F}_{Aux,1} = -\left(P_{Aux,2}A_N - P_{Aux,1}A_N\right) - \left(\frac{\left(\frac{F_{Aux,1}+F_{Aux,2}}{2}\right)^2}{\rho_{Aux,2}A_N} - \frac{\left(\frac{F_{In}+F_{Aux,1}}{2}\right)^2}{\rho_{Aux,1}A_N}\right), \tag{3.32}$$

$$\begin{aligned}\left(\frac{\Delta x_{Aux,1}}{2} + \frac{\Delta x_1}{2}\right)\dot{F}_{Aux,2} = &-\left(P_1A_1 - \frac{P_{Aux,2}\frac{\Delta x_1}{2} + P_1\frac{\Delta x_{Aux}}{2}}{\frac{\Delta x_{Aux}}{2} + \frac{\Delta x_1}{2}}(A_1 - A_N) - P_{Aux,2}A_N\right) \\ &- \left(\frac{\left(\frac{F_{Aux,2}+F_1}{2}\right)^2}{\rho_1 A_1} - \frac{\left(\frac{F_{Aux,1}+F_{Aux,2}}{2}\right)^2}{\rho_{Aux,2}A_N}\right) \\ &- \frac{2f_1}{D_1}\frac{F_{Aux,2}^2}{\rho_1 A_1}\frac{\Delta x_1}{2},\end{aligned} \tag{3.33}$$

$$\begin{aligned}\left(\frac{\Delta x_1}{2} + \frac{\Delta x_2}{2}\right)\dot{F}_1 = &-\left(P_2A_2 - \frac{P_1\frac{\Delta x_2}{2} + P_2\frac{\Delta x_1}{2}}{\frac{\Delta x_1}{2} + \frac{\Delta x_2}{2}}(A_2 - A_1) - P_1A_1\right) \\ &- \left(\frac{\left(\frac{F_1+F_2}{2}\right)^2}{\rho_2 A_2} - \frac{\left(\frac{F_{Aux,2}+F_1}{2}\right)^2}{\rho_1 A_1}\right) \\ &- \frac{2f_1}{D_1}\frac{F_1^2}{\rho_1 A_1}\frac{\Delta x_1}{2} - \frac{2f_2}{D_2}\frac{F_1^2}{\rho_2 A_2}\frac{\Delta x_2}{2}\end{aligned} \tag{3.34}$$

$$\frac{\Delta x_N}{2}\dot{F}_N = -\left(P_{Out}A_N - P_N A_N\right) - \left(\frac{F_N^2}{\rho_N A_N} - \frac{\left(\frac{F_{N-1}+F_N}{2}\right)^2}{\rho_N A_N}\right) - \frac{2f_N}{D_N}\frac{F_N^2}{\rho_N A_N}\frac{\Delta x_N}{2}, \tag{3.35}$$

$$\Delta x_{Aux} A_N \dot{\psi}_{Aux,1} = -\left(F_{Aux,1}h_{Aux,1} - F_{In}h_{In}\right), \tag{3.36}$$

$$\Delta x_{Aux} A_N \dot{\psi}_{Aux,2} = -\left(F_{Aux,2}h_{Aux,2} - F_{Aux,1}h_{Aux,1}\right), \tag{3.37}$$

$$\Delta x_1 A_1 \dot{\psi}_1 = -\left(F_1 h_1 - F_{Aux,2}h_{Aux,2}\right). \tag{3.38}$$

The algebraic equations that represent these coupling conditions are:

$$h_{In} = h_{form} + \frac{Cp}{M}\left(T_{In} - T_{stand}\right), \tag{3.39}$$

$$F_{In} = F_N, \tag{3.40}$$

$$\Delta P(t) = P_{In} - P_{Out}, \tag{3.41}$$

$$\left(\frac{T_{In}}{T_N}\right)^{\gamma} = \left(\frac{P_{In}}{P_{Out}}\right)^{\gamma-1}. \tag{3.42}$$

The second combination of boundary conditions used was similar, but it transformed the differential variable F_{In} into an algebraic variable by eliminating its corresponding control volume in the staggered grid. In addition, P_{In} was replaced by $P_{Aux,1}$ as boundary condition (Fig. 3-4). This combination of coupling conditions was used only in the loop.

The differential equations that represent these coupling conditions are:

$$\Delta x_{Aux} A_N \dot{\rho}_{Aux,1} = (F_{In} - F_{Aux,1}), \tag{3.43}$$

$$\Delta x_{Aux} A_N \dot{\rho}_{Aux,2} = (F_{Aux,1} - F_{Aux,2}), \tag{3.44}$$

$$\Delta x_1 A_1 \dot{\rho}_1 = (F_{Aux,2} - F_1), \tag{3.45}$$

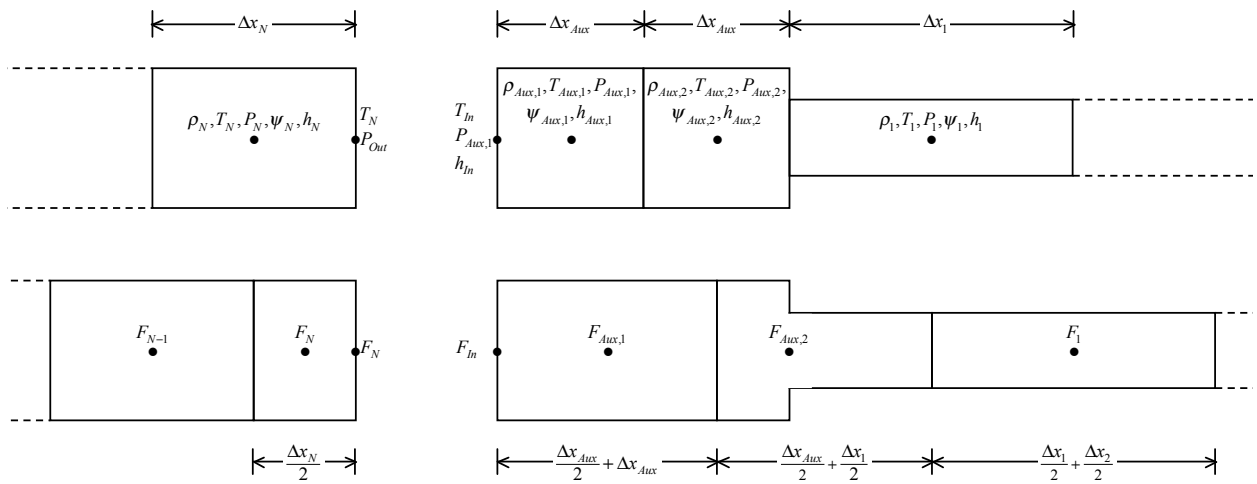

Figure 3-4: Alternative approach to set the boundary conditions for a gas dynamics model; it replaces the specification on P_{In} with $P_{Aux,1}$.

$$\left(\Delta x_{Aux} + \frac{\Delta x_{Aux}}{2}\right)\dot{F}_{Aux,1} = -\left(P_{Aux,2}A_N - P_{Aux,1}A_N\right) - \left(\frac{\left(\frac{F_{Aux,1}+F_{Aux,2}}{2}\right)^2}{\rho_{Aux,2}A_N} - \frac{F_{In}^2}{\rho_{Aux,1}A_N}\right), \tag{3.46}$$

$$\begin{aligned}\left(\frac{\Delta x_{Aux}}{2} + \frac{\Delta x_1}{2}\right)\dot{F}_{Aux,2} = &-\left(P_1A_1 - \frac{P_{Aux,2}\frac{\Delta x_1}{2} + P_1\frac{\Delta x_{Aux}}{2}}{\frac{\Delta x_{Aux}}{2} + \frac{\Delta x_1}{2}}(A_1 - A_N) - P_{Aux,2}A_N\right) \\ &- \left(\frac{\left(\frac{F_{Aux,2}+F_1}{2}\right)^2}{\rho_1 A_1} - \frac{\left(\frac{F_{Aux,1}+F_{Aux,2}}{2}\right)^2}{\rho_{Aux,2}A_N}\right) \\ &- \frac{2f_1}{D_1}\frac{F_{Aux,2}^2}{\rho_1 A_1}\frac{\Delta x_1}{2},\end{aligned} \tag{3.47}$$

$$\begin{aligned}\left(\frac{\Delta x_1}{2} + \frac{\Delta x_2}{2}\right)\dot{F}_1 = &-\left(P_2A_2 - \frac{P_1\frac{\Delta x_2}{2} + P_2\frac{\Delta x_1}{2}}{\frac{\Delta x_1}{2} + \frac{\Delta x_2}{2}}(A_2 - A_1) - P_1A_1\right) \\ &- \left(\frac{\left(\frac{F_1+F_2}{2}\right)^2}{\rho_2 A_2} - \frac{\left(\frac{F_{Aux,2}+F_1}{2}\right)^2}{\rho_1 A_1}\right) \\ &- \frac{2f_1}{D_1}\frac{F_1^2}{\rho_1 A_1}\frac{\Delta x_1}{2} - \frac{2f_2}{D_2}\frac{F_1^2}{\rho_2 A_2}\frac{\Delta x_2}{2},\end{aligned} \tag{3.48}$$

$$\frac{\Delta x_N}{2}\dot{F}_N = -\left(P_{Out}A_N - P_N A_N\right)$$

$$
- \left(\frac{F_N^2}{\rho_N A_N} - \frac{\left(\frac{F_{N-1}+F_N}{2}\right)^2}{\rho_N A_N} \right)
$$

$$
- \frac{2 f_N}{D_N} \frac{F_N^2}{\rho_N A_N} \frac{\Delta x_N}{2}, \tag{3.49}
$$

$$
\Delta x_{Aux} A_N \dot{\psi}_{Aux,1} = - \left(F_{Aux,1} h_{Aux,1} - F_{In} h_{In}, \right), \tag{3.50}
$$

$$
\Delta x_{Aux} A_N \dot{\psi}_{Aux,2} = - \left(F_{Aux,2} h_{Aux,2} - F_{Aux,1} h_{Aux,1} \right), \tag{3.51}
$$

$$
\Delta x_1 A_1 \dot{\psi}_1 = - \left(F_1 h_1 - F_{Aux,2} h_{Aux,2} \right). \tag{3.52}
$$

The algebraic equations that represent these coupling conditions are:

$$
h_{In} = h_{form} + \frac{Cp}{M} \left(T_{In} - T_{stand} \right), \tag{3.53}
$$

$$
F_{In} = F_N, \tag{3.54}
$$

$$
\Delta P(t) = P_{Aux,1} - P_{Out}, \tag{3.55}
$$

$$
\left(\frac{T_{In}}{T_N} \right)^{\gamma} = \left(\frac{P_{Aux,1}}{P_{Out}} \right)^{\gamma - 1}. \tag{3.56}
$$

Analogous coupling conditions were used for the *Zero Mach Number Limit* model and for the *QSS Approximation* model. In the case of the *QSS Approximation* model, the equation forcing mass conservation at the compressor (Eq. (3.54)) can be derived from the continuity equations, which produces an under determined system of equations. To eliminate this issue an equation forcing mass conservation was implemented: $\sum_{i=1}^{N} \rho_i A_i \Delta x_i = M(t)$, where $M(t)$ is the total mass of helium in the loop. This new user-specified variable, $M(t)$, is constant in most of the simulations, except in the case when helium leaks in the loop are simulated.

The combination of the three gas dynamics models and the two sets of boundary conditions led to six possible models for the heat transfer loop.

3. *Heat exchanger model.* The heat exchangers are represented by a thermal model for the metal structure (Eq. (3.57)) and by two pipe models—as the ones mentioned above. The thermal model accounts for the thermal inertia of the heat exchanger and considers conduction in the axial direction. This model assumes that heat losses are negligible. The equations in this model are discretized using a control volume formulation and a centered difference scheme. Each one of the control volumes has two corresponding control volumes in the pipe models (a hot pipe and a cold pipe) for the heat exchange.

$$c_{Metal}\rho_{Metal}\frac{\partial T_{Metal}}{\partial t} = \frac{\partial}{\partial x}\left(k\frac{\partial T_{Metal}}{\partial x}\right) + Q_{ToColdPipe} - Q_{FromHotPipe} \quad (3.57)$$

After creating the individual modules for all these units, they were assembled to create a model of the full heat transfer loop. Six different heat transfer loop models were created from the combination of three types of pipe models and two combinations of boundary conditions. In these models, the IHX and PHX models used 40 control volumes for the discretization, and the hot and cold legs connecting the heat exchangers used 10. The models so built consisted of 1,550 equations and variables.

3.3.2 RELAP model for comparison

We created a similar model for the heat transfer loop in RELAP [52], in order to have a good reference for comparison with the equation-based model. RELAP is a powerful simulator used to study thermal hydraulics and accidents in nuclear reactors. Because it cannot include chemical reactions easily, it cannot simulate a nuclear reactor coupled to a chemical plant. However, its basic elements are more than enough to represent the heat transfer loop studied in this chapter (Fig. 3-5). In particular, the heat transfer loop can be represented by a combination of heat structures, pipe components and junction components. The compressor is represented as a junction, whose flowrate is specified directly. A similar heat transfer loop model in RELAP had already been created by Davis et al. [8]; this model used molten salt as the heat

transfer fluid. This model was used as a base for the RELAP model in this study.

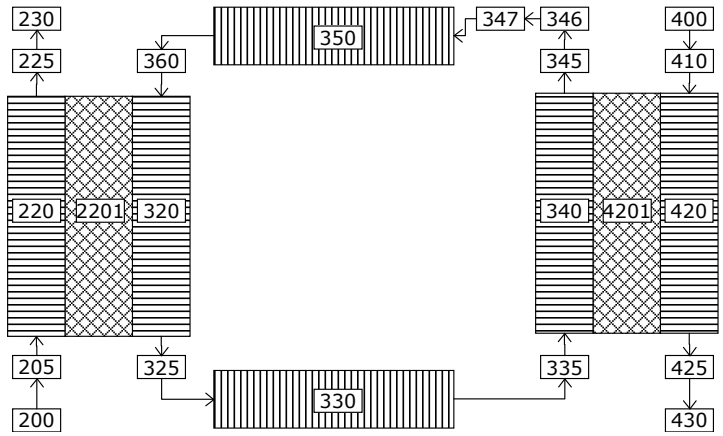

Figure 3-5: RELAP model used to benchmark JACOBIAN models.

3.4 Model Performance Evaluation

The heat transfer loop models from the previous section were tested for feasibility, accuracy and speed. The goal of running these tests is to find the models that can provide an accurate simulation of the heat transfer loop over the interesting time scales (greater than 0.5 s).

3.4.1 Test 1: Feasibility and steady-state profiles

The first step to evaluate the models is to show that the corresponding systems of equations are consistent and that the models can match the designed operating point. To do the former we used the error-checking tools included in JACOBIAN, and to do the latter we compared JACOBIAN's steady-state profiles to RELAP's.

JACOBIAN's error-checking tools revealed that for the loop Eqs. (3.28)-(3.42) were an incorrect choice of coupling conditions, whereas Eqs. (3.43)-(3.56) were the right one. As mentioned before, consistent initial conditions are needed to start numerical integration of this kind of model and to produce reliable results. Unfortunately, using Eqs. (3.28)-(3.42) as coupling conditions for the loop created an inconsistent system of equations for the calculation of initial conditions. This meant

that one of the equations was inconsistent and incompatible with the rest of the system. On the other hand, using Eqs. (3.43)-(3.56) as boundary conditions for the loop did not create this problem.

A high-index problem with a hidden constraint explains the issue caused by using Eqs. (3.28)-(3.42) as coupling conditions for the loop. The system of equations for the loop created by the combination of the semi-discretized version of the *Full Dynamics* model (Eqs. (3.20)-(3.25)) and the coupling conditions (Eqs. (3.28)-(3.42)) can be written as a semi-explcit DAE system:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t), t), \tag{3.58}$$

$$\mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), t) = \mathbf{0}. \tag{3.59}$$

A semi-explicit DAE system is index 1 if and only if $\frac{\partial \mathbf{g}}{\partial \mathbf{y}}$ is nonsingular. Closer examination shows that the equation forcing mass conservation (Eq. (3.40)) at the compressor is the one conflicting with the rest of the equations:

$$F_{In} = F_N.$$

In this case, F_{In} and F_N are both differential variables on the grid shown in Fig. 3-3, which means that $\frac{\partial \mathbf{g}}{\partial \mathbf{y}}$ has a zero row from Eq. (3.40). Therefore, the system of equations for the loop is high-index and cannot be solved with standard codes.

On the other hand, using Eqs. (3.43)-(3.56) as coupling conditions does not generate this problem. When using Eqs. (3.43)-(3.56) as coupling conditions on the grid shown in Fig. 3-4, the equation forcing mass conservation does not create a high-index problem because F_{In} is an algebraic variable and not a differential one. Then, $\frac{\partial \mathbf{g}}{\partial \mathbf{y}}$ is nonsingular and the system of equations for the loop is index 1. Consequently, only the JACOBIAN models using this set of coupling conditions are used in the following tests.

Next, we simulated the plant's designed steady-state operating point using the JACOBIAN and RELAP models and we confirmed that the resulting variable profiles

matched. The operating point has been calculated by Davis et al. [8] using lumped models. It consists of a heat transfer loop operating at a pressure of 2.0 MPa, with a flow of 27.5 kg/s, a maximum temperature of 885°C at the outlet of the IHX and a minimum temperature of 460°C at the outlet of the PHX. The hot stream in the IHX–nuclear reactor coolant–enters at 900°C and 7.0 MPa, and leaves at 6.95 MPa. The cold stream of the PHX–steam–enters at 341°C and 1.0 MPa and leaves at 850°C. We implemented these conditions in the JACOBIAN and RELAP models.

Under these conditions, the pressure and temperature profiles calculated by RELAP and JACOBIAN agreed with some small differences (Figs. 3-6 and 3-7). These differences appear because JACOBIAN models and the RELAP model use different equations to represent the physical properties of helium and the heat structures in the loop (friction factors, heat transfer factors, and equation of state for helium). Overall, RELAP's and JACOBIAN's steady-states profiles are similar enough that they can be used as a starting point for the comparison of their dynamic behavior.

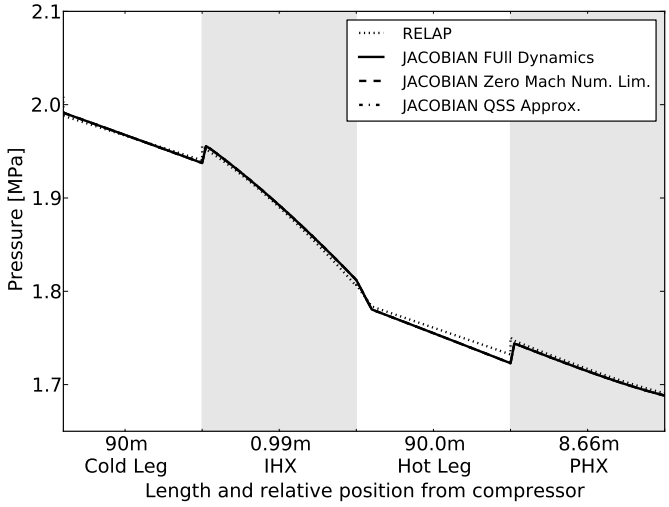

Figure 3-6: Heat transfer loop's pressure profiles at steady state in JACOBIAN match RELAP's profiles with small differences.

3.4.2 Test 2: Dynamic simulation profiles

The next step is to understand if JACOBIAN models can reproduce the dynamics of the heat transfer loop accurately. In particular, we are interested in understanding

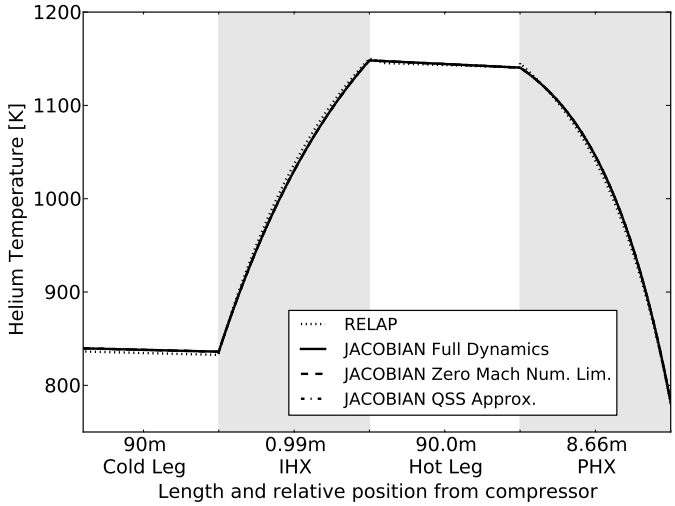

Figure 3-7: Heat transfer loop's temperature profiles at steady state in JACOBIAN match RELAP's profiles with small differences.

how fast the models respond to changes in the system. To do this, we simulated the models at the operating points calculated in the previous section, and then we introduced fast changes in some variables. The first transient was a linear decrease in the PHX's inlet pressure over 0.2 s that produced a 50% decrease in cold stream flowrate, to simulate a heat sink capacity loss in the chemical plant. The second transient was a linear increase in the pressure differential at the compressor over 0.2 s that generated a 10% increase in the loop's gas flow, to study how the sudden change in pressure propagates around the loop.

A 50% decrease in the PHX's cold stream flowrate can illustrate how the heat transfer loop responds to disruptions in the chemical plant. The chemical plant behavior can have unexpected consequences for the nuclear reactor; understanding how the loop performs in these situations will help guarantee the safety of the system. In particular, process disruptions in the chemical plant will produce a loss of its heat sink capacity and this might lead to an increase in the nuclear reactor temperature. One way to illustrate this loss of heat sink capacity is to decrease the flowrate in the PHX's cold stream. This was represented by using a linear decrease in pressure at the inlet of the PHX that produced a 50% decrease in the flowrate, and it was implemented in the JACOBIAN and RELAP models. The time span of the pressure

change was 0.2 s.

To compare the models' dynamic responses in JACOBIAN and RELAP, scaled profiles of temperature and pressure had to be used (Figs. 3-9 and 3-10). RELAP is designed to simulate very fast transients in nuclear reactors and it does not use rigorous error control for mass conservation [52]. Thus, the mass in the loop can change during transients, which produces distorted pressure and temperature profiles that cannot be compared to JACOBIAN's profiles. For example, after introducing a 50% decrease in the PHX's cold stream flowrate, the mass in the loop decreased almost 4% (Fig. 3-8). In this simulation, the total helium mass at $t = 0$ s is greater in the RELAP model than in the JACOBIAN models (Fig. 3-8) because this model requires additional control volumes to connect the different components in the loop (i.e., to connect the cold leg to the IHX).

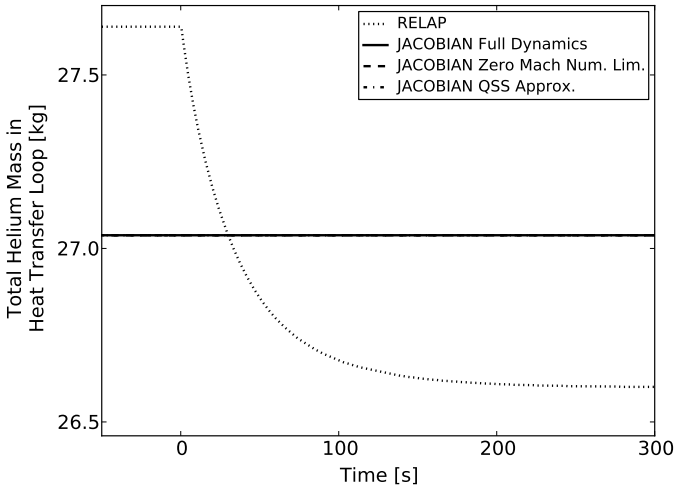

Figure 3-8: The heat transfer loop's total helium mass decreases in RELAP after introducing 50% decrease in PHX's cold stream inlet flowrate.

The three JACOBIAN models and the RELAP model responded in the same time scale to the 50% decrease in the PHX's cold stream flowrate, but their final steady states differed. To study this response we tracked the pressure and temperature of helium in the heat transfer loop at the outlet of the PHX. This point is the closest to the cold stream's inlet and it should experience the largest and fastest temperature change. Thus, it can show how quickly the system can respond when the linear

pressure change is introduced at $t = 0.0$ s. The scaled pressure and temperature profiles (Figs. 3-9 and 3-10) at this point are almost identical for the three JACOBIAN models and for the RELAP model. At the same time, these profiles are smooth, in contrast to the fast linear change originally introduced. This occurs because the heat conduction between the two PHX streams introduces a lag in the heat transfer. However, the final temperature and pressure at this point after 300 s in the RELAP model's result were lower from those in the JACOBIAN models' results. This is explained by the loss of helium mass in RELAP.

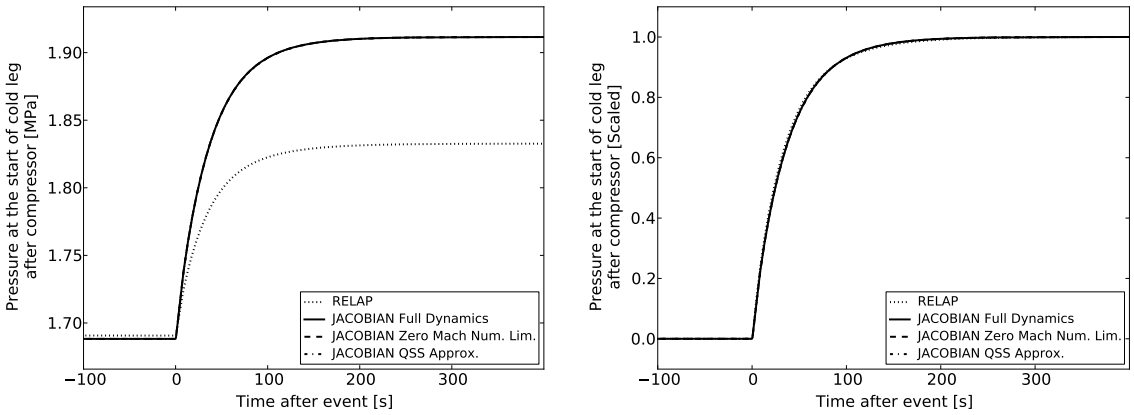

Figure 3-9: Pressure of helium in the loop at the end of the PHX after a 50% decrease in PHX's cold stream flowrate. RELAP simulation reaches a lower pressure than that in JACOBIAN simulations because of RELAP's numerical error in mass conservation; however, RELAP and JACOBIAN models respond with the same speed to the step change.

A more demanding scenario is to introduce an increase of 10% in the flowrate inside the heat transfer loop. Introducing such a change can show the differences in the responses of the different models more clearly, because it directly changes the conditions of the gas inside the heat transfer loop. Additionally, this kind of change does not face the smoothing effects of intermediate units (i.e., PHX) as in the previous scenario. The change in flowrate at the compressor was represented as a linear increase in $\Delta P(t)$ in the JACOBIAN models and as a linear increase in the flowrate in the RELAP model. The time span of the change was 0.2 s, pressure and temperature of helium were tracked at the outlet of the compressor.

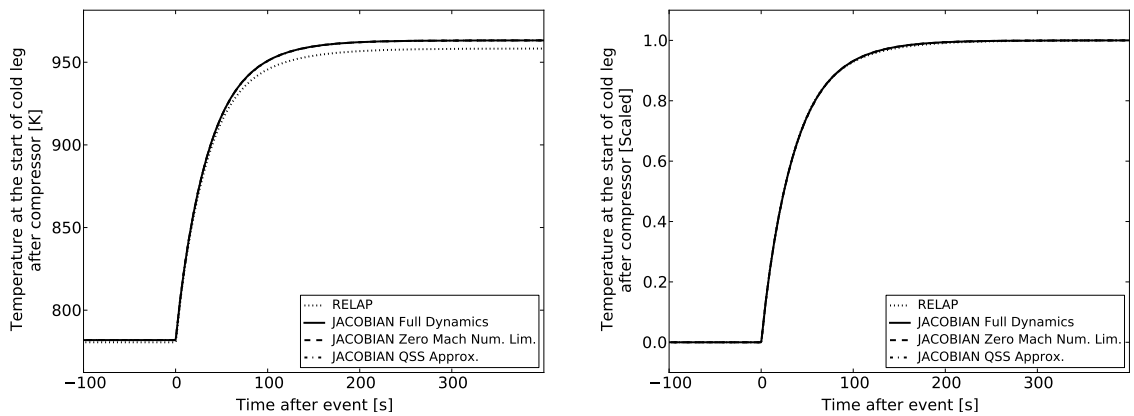

Figure 3-10: Temperature of helium in the loop at the end of the PHX after a 50% decrease in PHX's cold stream flowrate. RELAP simulation reaches a lower temperature than that in JACOBIAN simulations because of RELAP's numerical error in mass conservation; however, RELAP and JACOBIAN models respond with the same speed to the step change.

A step change in the variables would be an even more demanding scenario. This was not used because JACOBIAN models are simplified representations of gas dynamics and lack the features to track fast pressure waves generated in such a scenario. Thus, the simulation of a step change in pressure makes the simulator fail or produces results that are physically incorrect. However, this is not an issue because strict step changes in pressure are not expected as accidents and because JACOBIAN models are designed to represent slower transients (those with a time span of over 1.0 s).

The increase of 10% in the flowrate of the heat transfer loop produces pressure and temperature profiles that change with similar time scales in all the models. Yet again, the the final steady state generated by the RELAP simulation differs from those produced by the JACOBIAN simulations because of the numerical errors in RELAP (Figs. 3-11 and 3-12). The pressure takes 53 s to achieve 90% of its final increase and the temperature takes 60 s to do the same.

However, the models perform differently during the first second after the change in pressure/flowrate is introduced (Figs. 3-13 and 3-14). The RELAP model's profiles for pressure and temperature present oscillations which result from RELAP's ability to simulate pressure waves in the gas. The JACOBIAN *Full Dynamics* model produces

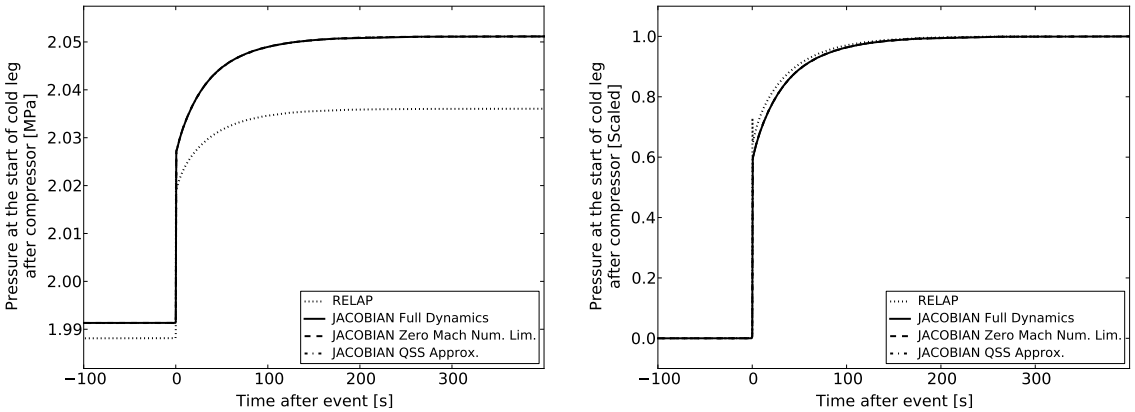

Figure 3-11: Pressure of helium in loop at the outlet of the compressor; the time scale of the response to a change in the compressor is similar again in the four models, but it differs during the first second of the transient. RELAP's numerical errors in mass conservation affect the final steady state.

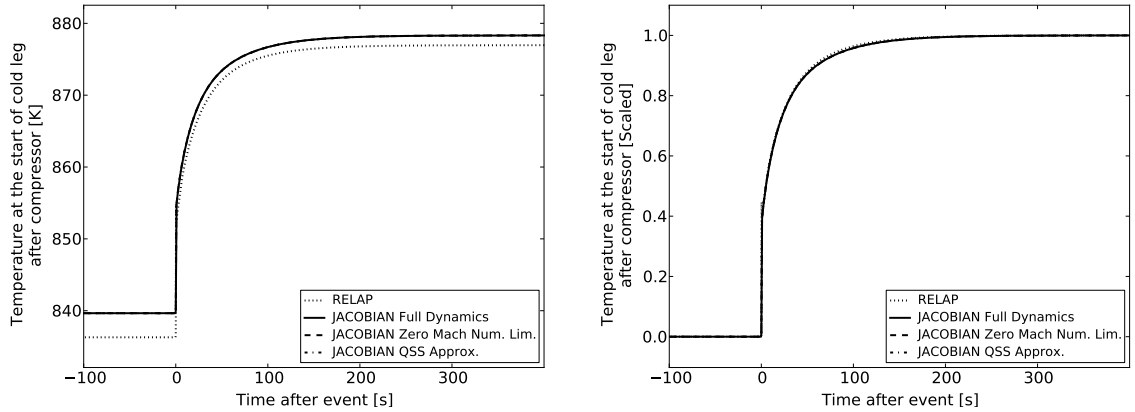

Figure 3-12: Temperature of helium in loop at the outlet of the compressor; the response to a change in the compressor is similar again in the four models, but it differs during the first second of the transient. RELAP's numerical errors in mass conservation again affect the final steady state.

small and distorted oscillations. This occurs because the method of lines used to represent to the partial differential equations is not enough to track the pressure waves in the gas. On the other hand, the JACOBIAN *Zero Mach Number Limit* model and the *QSS Approximation* model produce a profiles without oscillations in pressure and temperature. These models eliminate the fast time scales in the gas dynamics that create the oscillations, but they are still accurate after 0.5 s.

From this analysis, the *Zero Mach Number Limit* model and the *QSS Approximation* model emerge as the best JACOBIAN models to simulate the heat transfer loop.

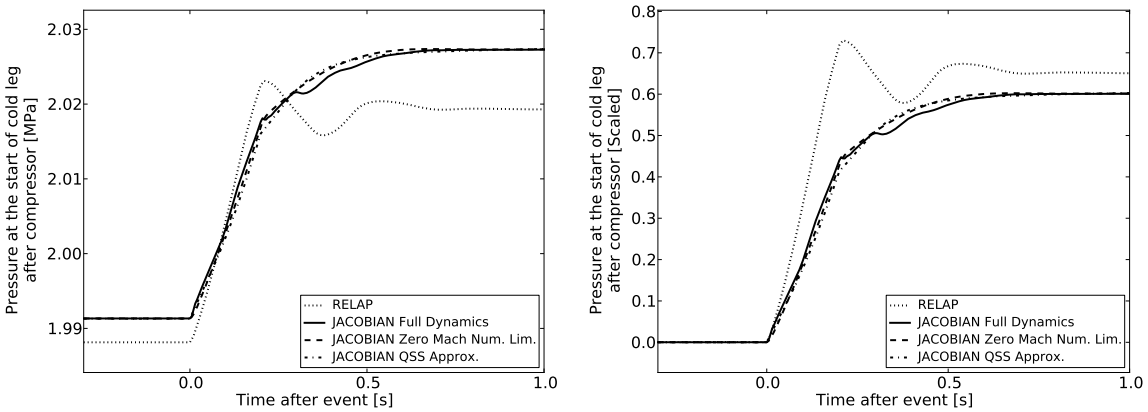

Figure 3-13: The RELAP model's pressure profile presents oscillations during the first 0.5 s of the transient because of pressure wave effects in the helium. The simplifications used in the JACOBIAN models distort or eliminate these oscillations.

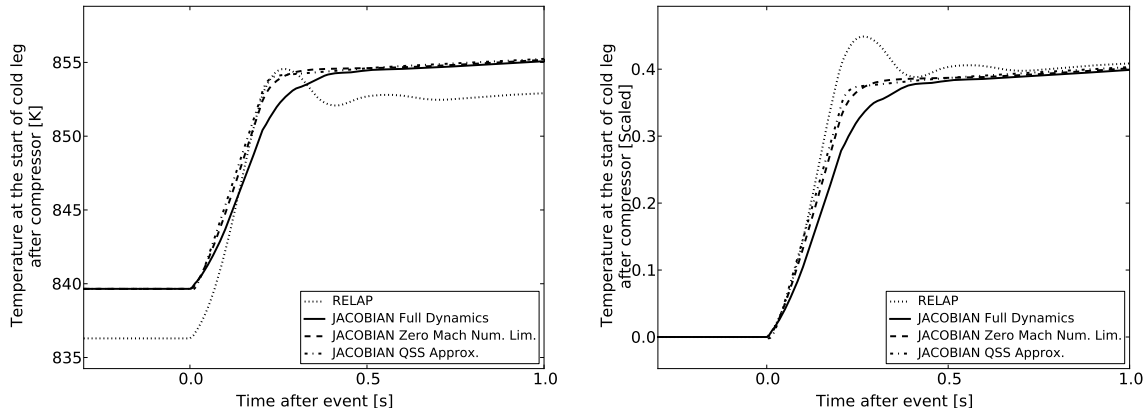

Figure 3-14: The RELAP model's temperature profile presents oscillations during the first 0.5 s of the transient because of pressure wave effects in the helium. The simplifications used in the JACOBIAN models distort or eliminate these oscillations.

As mentioned earlier, the purpose of formulating these models is to create a model of a nuclear reactor coupled to a hydrogen production plant. Such a model will allow simulating scenarios such as system start-up or the system behavior after a disruption in the chemical plant. These scenarios happen over time-scales of more than 1.0 s, and the behavior of the model before this time is not so relevant. Then a good heat transfer loop model is a model that behaves reasonably well during the first second of the transient and is accurate afterwards. The *Zero Mach Number Limit* model and the *QSS Approximation* model satisfy these conditions.

3.4.3 Test 3: Computational time

Tests so far have shown two of the JACOBIAN models to be feasible and accurate in the relevant time scales; CPU times required for simulations show that they are fast too. Simulating 160 s of simulation time for the step change in the compressor (10 s before step change and 150 s after it) can be more to 25 times faster in JACOBIAN than in RELAP. The *Full Dynamics* model takes 3 min and 25 s, the *Zero Mach Number Limit* model takes 3 min and 20 s, and the *QSS Approximation* model takes 3 min 7 s. In contrast, the RELAP model takes 86 min and 55 s (Fig. 3-15). The simulations were run using JACOBIAN v. 4.0 and RELAP v. 2.3.6 on a Windows virtual machine supported by VMWare Fusion v. 2.6 on Mac OS X v. 10.5.8. The system used was an Apple MacBook with 2.0 GHz Intel Core 2 Duo processors and 4 GB of memory.

These performance difference is explained by how RELAP and JACOBIAN were designed. On the one hand, RELAP was designed to simulate light-water reactor coolant systems, and it is particularly good at simulating two-phase systems. In order to do this, RELAP involves complex tailored numerical schemes, which increase calculation time. On the other hand, JACOBIAN is flexible and allows simulating only the necessary models. In addition, JACOBIAN uses state-of-the-art stiff numerical integration algorithms. The combination of all these characteristics makes JACOBIAN faster.

3.5 Conclusions

The proposed models in this chapter show that it is possible to simulate the heat transfer loop in an equation-based simulator such as JACOBIAN. The models represented the loop's steady state at the designed operating point and they simulated the loop's dynamics during different transient scenarios. They did all this with similar accuracy as the models in RELAP (during relevant time scales), but additionally they were faster and better at mass conservation.

The proposed models and this study are limited and can be improved, yet their

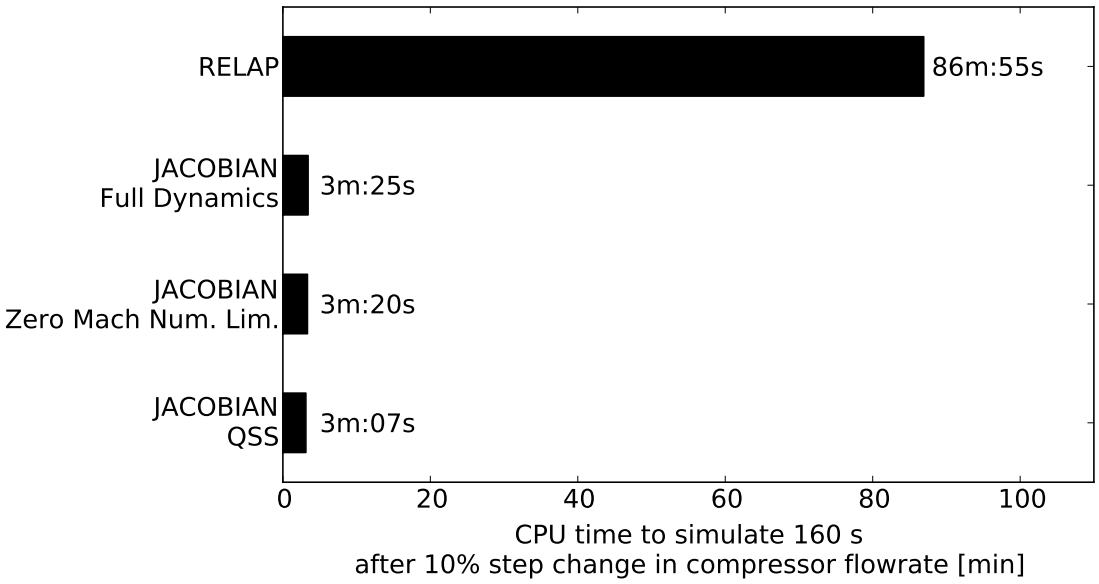

Figure 3-15: Heat transfer loop simulations in JACOBIAN are more than 10 times faster than corresponding simulations in RELAP.

implications are still valid. First, it is hard to show that JACOBIAN models will agree with RELAP models in all situations, and our set of test scenarios was restricted. Nevertheless, we believe that the scenarios simulated using fast changes in key variables give a good estimate of the model's behavior. Second, a more sophisticated compressor model could make the simulation more realistic. However, this is not necessary for this study, as this work's main goal was to show that gas dynamics over relevant time scales can be simulated effectively and we achieved that. Third, a different finite difference scheme could be used to discretize the partial derivatives in space. This would help to better track the waves and transients during the first 0.5 s after a change in the system is introduced in the *Full Dynamics* model.

Creating these models has a long-term goal, which is to propose equation-based simulators as an approach to simulate alternative nuclear applications. Equation-based simulators can comfortably incorporate these new processes, because they are modular, extendable and flexible platforms. They can also be easily audited by regulators, because they are open and declarative modelling environments. They are also faster and more precise than legacy codes because of the fast and accurate

algorithms they use for integration. For these reasons, equation-based simulators are a good choice to prove the safety and reliability of new nuclear applications, and hopefully their use will be included in future developments.

Chapter 4

Transient analysis of a nuclear hydrogen production facility

4.1 Introduction

As explained in Chapter 2, nuclear hydrogen production is a promising alternative to reduce CO_2 emissions associated to hydrogen production using traditional methods. Hydrogen is a very important commodity used for oil refining and chemicals production. Currently, its production releases 1% of the CO_2 produced in the U.S. because steam methane reforming is the predominant technology. These emissions can be eliminated by powering hydrogen production from water with a nuclear reactor. This system consists of a nuclear reactor, a heat transfer loop and a chemical plant. The nuclear reactor generates heat and electricity, and the heat transfer loop supplies heat to the chemical plant. The chemical plant uses the heat and electricity to split water into H_2 and O_2. To split water, two chemical processes are being considered: high-temperature steam electrolysis (HTSE) and the Sulfur-Iodine (SI) cycle [63]. Both processes operate at high temperature, which helps them deliver efficiencies of over 45%.

In particular, HTSE can achieve high operating efficiencies and its underlying technology has made considerable progress. High-temperature electrolysis is the decomposition of steam at high temperatures (>850°C) using solid oxide electrolysis

cells (SOECs). Operating at high temperatures offers thermodynamic advantages and the electrical demand is reduced [63]. Also, at high temperatures the kinetics of the reactions are faster [20]. These factors allow for a higher thermal-to-hydrogen conversion efficiency than that achieved with traditional electrolysis methods. This efficiency is defined as the ratio of energy content of the hydrogen produced (heating value) over the thermal energy input from the reactor (as heat or electricity). This efficiency has been projected to be 45% to 55% [63, 18], which will allow competitive hydrogen production costs to be achieved [19]. The Idaho National Laboratory (INL) has made a lot of progress in this area, and the efficiency and reliability of the cells has increased over the years [51].

Showing the safety of such a system with computational simulations is fundamental to obtain regulatory approval. The system comprising the HTSE plant, heat transfer loop and nuclear reactor operates at extreme conditions that can cause operational upsets and affect the nuclear reactor. For example, the system will lose its heat sink capacity if the HTSE plant suddenly shuts down, and this will perturb the temperature of the coolant in the nuclear reactor. Verification that the reactor and system will behave safely during this and other extreme upsets is required to obtain approval from authorities. This can be achieved by studying the system with computational simulations, which allow different combinations of conditions to be tested without risking real accidents. Computational simulations of nuclear reactors in power plants have been used extensively and different legacy codes exist for this (e.g., RELAP [52], MELCOR [14]); nuclear hydrogen production needs to undergo similar rigorous tests.

However, the complexity of nuclear hydrogen production systems has made it difficult to make progress with the simulation of accident scenarios. The simulation of nuclear reactors is already complex because it involves simulating reactor kinetics and transport phenomena. In the case of nuclear hydrogen production using HTSE, it is necessary to also simulate the chemical reactions happening in the hydrogen plant. Chemical reactions cannot be represented easily in legacy codes because these codes were not designed to incorporate them. Some efforts have been made to modify

legacy codes to represent chemical reactions, but they involve large teams of scientists [43]. This situation has resulted in a limited number of studies on the safety of these systems [43, 58].

This chapter makes progress in the study of accident scenarios in nuclear hydrogen production systems by using more flexible and powerful simulation codes: equation-based simulators. Equation-based simulators support very flexible modeling languages that allow the representation of diverse phenomena, and can be used to model nuclear hydrogen production systems easily. At the same time, they are powerful simulation codes and their use can increase the computational speed of the safety studies. In this work, we study three potential accidents and their effects on the core: a loss of heat sink at the HTSE, a loss of flow in the heat transfer loop, and a leak in the heat transfer loop.

This chapter first describes the nuclear hydrogen production system modeled; then it describes the models used to represent it. Finally, it explains the different transients studied and presents results on how the system responds.

4.2 Definition of the system

The nuclear hydrogen production facility considered in this study comprises two nuclear reactors, a power conversion unit (PCU), a heat transfer loop and a HTSE plant (Fig. 4-1). The nuclear reactors are modular, can produce the high-temperature heat needed for hydrogen production (900°C), and each of them produces 250 MWth. Fifty MWth of the heat produced by nuclear reactor #1 is delivered by the heat transfer loop to the HTSE plant. This provides the heat necessary to maintain a high-temperature environment in the HTSE unit. The remaining heat produced by nuclear reactor #1 and all of the heat produced by nuclear reactor #2 is transferred to the PCU to produce the electricity necessary to drive electrolysis in the HTSE unit.

The focus of this preliminary study will be on the behavior of nuclear reactor #1 and the heat transfer loop during accidents. The safety of nuclear reactors coupled

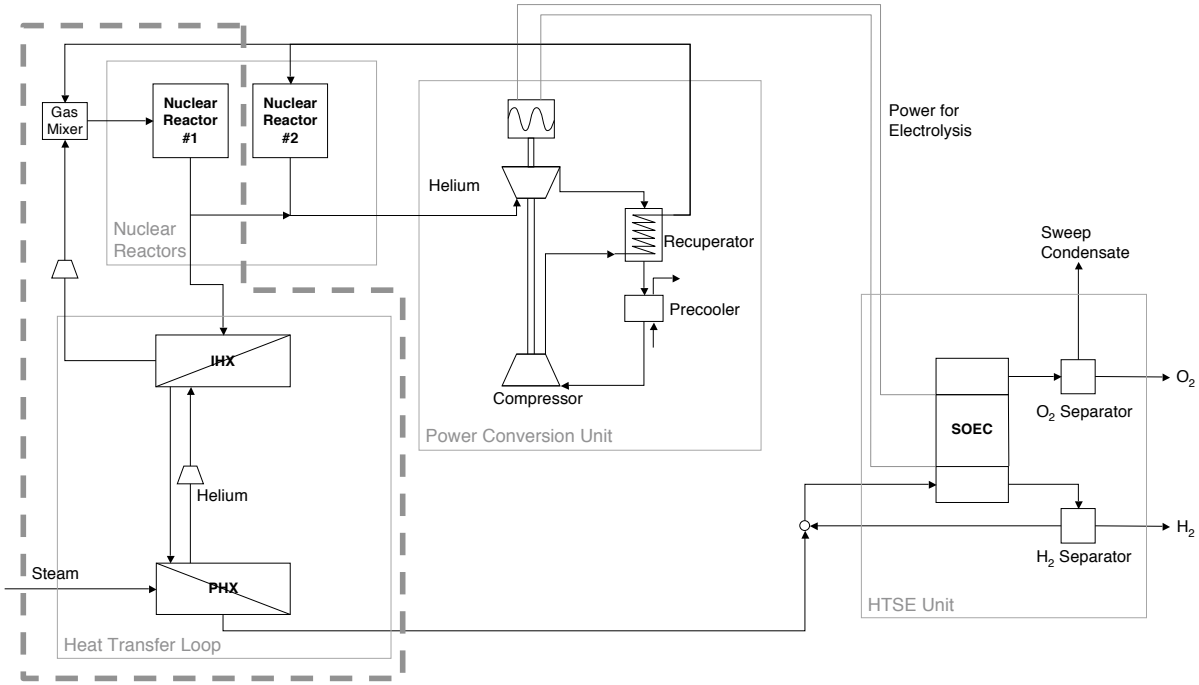

Figure 4-1: Nuclear hydrogen production system studied; the units surrounded by the dashed line will be simulated in this study.

to PCUs has been studied extensively in the past and we do not need to re-study their interaction. In this new system then, it is necessary to understand how the new parts, the heat transfer loop and the HTSE unit, affect safety. Three accident scenarios with potential severe impact on the nuclear reactor were studied: (1) a loss of heat sink accident, (2) a loss of flow in the heat transfer loop, and (3) a leak in the heat transfer loop. The study of these transients requires a detailed simulation of nuclear reactor #1 coupled with the heat transfer loop; the HTSE unit can be reduced to a heat sink at the cold side of the PHX (the steam superheater) (Fig. 4-1).

Nuclear reactor

The nuclear reactor used in the system is a bundle of two pebble bed modular reactors (PBMR). PBMRs are an ideal choice for the nuclear hydrogen production system because they can produce the high temperatures needed [37]. At the same time, they are safe, modular and present good economics [22, 65]. The specifications for the

reactors are based on the PBMR designed at MIT [23, 60] and each of these reactors can produce 250 MWth [22]. Their cores consist of a combination of fuel pebble balls and pure graphite pebble balls that allow online refueling (Fig. 4-2). Helium is used as the heat transfer fluid to bring the heat to the IHX and PCU. The cold helium enters in the lower part of the reactor, rises through channels in the side reflector, then is heated in the core, and then leaves the reactor at the bottom. A similar bundle of reactors has been proven to be technically and economically more convenient than the construction of a single nuclear reactor with the same power output [65].

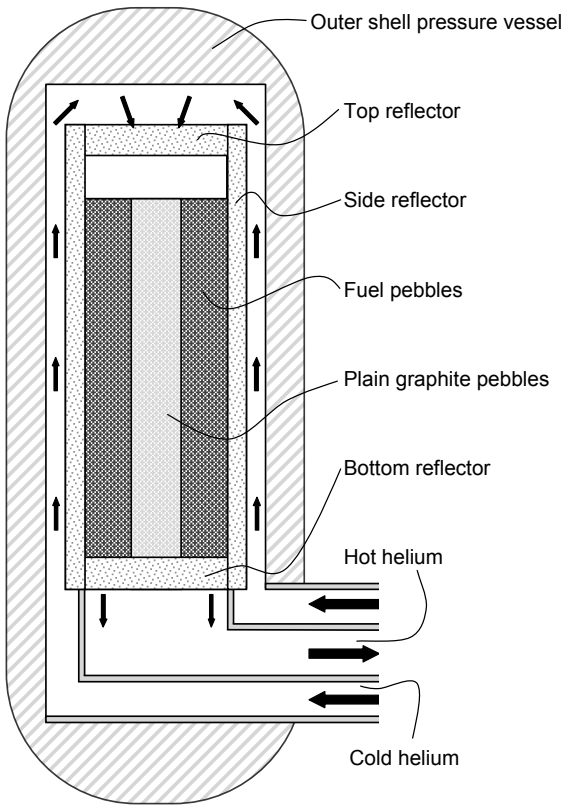

Figure 4-2: The PBMR is safe, modular, and economical, and it can produce the high-temperature heat needed to run hydrogen production plants.

Heat transfer loop

The heat transfer loop in this system is based on the heat transfer loop engineered at the INL [8], because of its detailed design. This heat transfer loop consists of an

intermediate heat exchanger (IHX), a process heat exchanger (PHX), a hot leg, a cold leg, and a compressor (Fig. 4-1). It transfers 50 MWth from the nuclear reactor to the chemical plant, and it has to withstand very high temperatures (up to 900°C) and high pressures (up to 7 MPa) [8]. These challenging conditions require a design created from scratch using new materials and new pieces of equipment.

Researchers at INL determined the following equipment characteristics for the heat transfer loop [8]:

1. The IHX is a compact heat exchanger made of Alloy 617 with 786,560 semi-circular channels 1.5 mm in diameter. The length of the IHX was 0.99 m and its width was 1.44 m.

2. The PHX is a tube-in-shell heat exchanger with 3,500 circular tubes; each tube has an internal diameter of 10 mm. High-temperature helium flows on the inside of the tubes and the steam flows on the outside. The length of the exchanger is 8.66 m and its material is Alloy 800.

3. The hot leg and cold leg are pipes of 90 m in length with internal diameters of 0.44 m and 0.41 m, thickness-to-diameter ratio of 0.117 and 0.114, and insulation thickness 59 mm and 58 mm, respectively. Both legs are made of Alloy 800.

4. The compressor should be able to produce a pressure differential of at least 0.4 MPa.

These specifications were used to build the dynamic models for the loop in JACOBIAN–one of the best equation-based simulators available [35].

4.3 Dynamic model of a nuclear hydrogen production system

As mentioned before, legacy codes normally used to study safety in nuclear reactors cannot be easily used to study nuclear hydrogen production systems. The reason for

this is that these codes cannot simulate all the phenomena involved in nuclear hydrogen production systems. In particular, they cannot incorporate chemical reactions. Equation-based simulators, on the other hand, are flexible and extensible platforms for system simulation that do not present these problems; we decided to use them instead to model our system. This section explains what equation-based simulators are, their strengths, and the models created for each part of the system considered.

4.3.1 Equation-based simulators as a simulation platform

Equation-based simulators are simulation tools created by the chemical engineering discipline to model the vast range of processes the chemical industry handles. They have been developed for more than 30 years to have flexibility, speed and accuracy. Their flexibility comes from supporting a declarative and hierarchical equation-based simulation language, which allows them to accommodate new models and different processes easily. Their speed and accuracy comes from having powerful numerical algorithms. These characteristics make them a good alternative for the simulation of new nuclear applications, which require the simulation of new phenomena and system-wide simulation of several interacting processes. We decided to use JACOBIAN [35], one of the best equation-based languages available.

A declarative equation-based language such as the one supported by JACOBIAN lets the user represent any physical or chemical system as a set of differential-algebraic equations (DAEs) [3]:

$$\mathbf{f}(\mathbf{x}(t), \dot{\mathbf{x}}(t), \mathbf{y}(t), t) = \mathbf{0}$$
$$\mathbf{g}(\mathbf{x}(t), \mathbf{y}(t), t) = \mathbf{0}$$

where $\mathbf{x}(t) \in \mathbb{R}^{n_x}$ and $\dot{\mathbf{x}}(t) \in \mathbb{R}^{n_x}$ represent the differential variables (those whose time derivatives appear explicitly in the model) and the time derivatives of these variables, respectively. $\mathbf{y}(t) \in \mathbb{R}^{n_y}$ corresponds to the algebraic variables. $\mathbf{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R} \rightarrow \mathbb{R}^{n_x}$ are the differential equations, and $\mathbf{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_y} \times \mathbb{R} \rightarrow \mathbb{R}^{n_y}$ are the algebraic equations. The philosophy of an equation-based simulator is that the

process model so defined is completely decoupled from the numerical methods used to solve it, which lets the user focus solely on the model formulation. This gives the user complete control over the model and he can create models from scratch, modify existing models accordingly and reuse other models.

Furthermore, declarative simulation languages support hierarchical model decomposition, which greatly simplifies the coding of large models [5]. It allows building models in levels in the same way one would conceptualize a real plant. For example, a first level of simple models can represent basic units (e.g., pipes, heat exchangers, chemical reactors). A second level uses models from the first to represent subsystems (e.g., heat transfer loop). Finally, a third level could incorporate all the subsystems to assemble a plant-wide model. With this approach, lower-level models can be easily tested and corrected, avoiding hard-to-find errors in higher-level models.

Equally important, equation-based simulators find solutions to simulations quickly and accurately. They assemble all the models' equations in one single equation system, which is solved simultaneously by general-purpose codes for root finding and implicit numerical integration. They also exploit the characteristics of the equation system (e.g., sparsity, block decomposition) to accelerate the solutions and solve 100,000s of equations in minutes. They are accurate too, because their error control is based on theoretical guarantees for consistency and stability, and the error can be controlled automatically.

4.3.2 Nuclear reactor model

The flexibility of JACOBIAN enables formulation of all the models required for simulation of the nuclear hydrogen production plant. As stated previously, only the nuclear reactor and the heat transfer loop need to be simulated to understand the system's behavior during the selected accidents. In this section, the model for the nuclear reactor is explained.

The nuclear reactor model is adapted from the model developed at MIT [60], which has the adequate level of detail for this study. This model only represents the pebble bed region in the core and the reflectors (Fig. 4-2), and it assumes the heat loss

from the reactor vessel is negligible. The pebble bed region in the core is cylindrical, its height is 8.5 m, the diameter of its passive region (plain graphite pebbles) is 1.75 m, and the total diameter of its active region (fuel pebbles) is 3.5 m (Fig. 4-2). The thickness of all reflectors is 1.15 m.

The model has to represent the conduction, convection and radiation of heat occurring among the different elements (pebbles, reflectors, helium), and the fission reactions in the core. To represent these phenomena, the core and reflectors are discretized using cylindrical coordinates. The axial discretization consists of 12 intervals; the top one and the bottom one represent the top and bottom reflectors, and 10 intervals in the center represent the core. The radial discretization consists of eight intervals; the center one represents the passive region, the next five represent the active region, and the final two represent the side reflectors. At each of the intervals, it is necessary to determine the temperature of the solid phases (pebbles and reflectors), and the temperature and pressure of the helium. The equations to determine these values are explained in the next section.

Temperature model for solid phases in the reactor

The behavior of the temperature of solid phases in the reactor (pebbles or reflectors) can be represented by the following general formula:

$$\rho C_P \frac{\partial T_{Solid}}{\partial t} = q_{Conduction} + q_{Radiation} + q_{Fission} + q_{Convection} \tag{4.1}$$

where T_{Solid} is the temperature of the solid phase (K), ρ is the density of the solid phase (kg/m^3) and C_P is heat capacity (J/(kg·K)). $q_{Conduction}$ corresponds to the heat conducted through the pebbles and the reflectors, and $q_{Radiation}$ is the heat radiated through the voids between the pebbles. $q_{Fission}$ represents the heat generated from fission in the pebbles, and $q_{Convection}$ is the heat transferred to the helium by convection. The definition of each of these terms in the equation is as follows.

1. Calculation of ρC_P. The heat capacity of the pebble bed can be calculated by

using the following equation [46]:

$$\rho C_P = 1.75 \cdot 10^6 (1-\varepsilon) \left[0.645 + 3.14 \left(\frac{T_{Solid} - T_0}{1000} \right) \right.$$
$$\left. -2.809 \left(\frac{T_{Solid} - T_0}{1000} \right)^2 + 0.959 \left(\frac{T_{Solid} - T_0}{1000} \right)^3 \right]. \tag{4.2}$$

In this equation, ε corresponds to the void fraction in the pebble bed, and T_0 is the reference temperature (273.15 K).

For the reflectors, ρ can be treated as a constant, with the value 1394.8 kg/m^3 [45]; C_P can be expressed as:

$$C_P = \frac{0.42 \cdot T_{Solid}}{1500}. \tag{4.3}$$

2. Calculation of $q_{Conduction}$ and $q_{Radiation}$. For the pebbles, $q_{Conduction}$ and $q_{Radiation}$ can be calculated as a single term by using

$$q_{Conduction} + q_{Radiation} = \frac{1}{r}\frac{\partial}{\partial r}\left(kr \frac{\partial T_{Solid}}{\partial r} \right) + \frac{\partial}{\partial z}\left(kr \frac{\partial T_{Solid}}{\partial z} \right), \tag{4.4}$$

and by using a conductivity constant that incorporates the radiation effect [45]:

$$k = 1.1536 \cdot 10^{-4} \left(T_{Solid} - 173.16 \right)^{1.6622}. \tag{4.5}$$

The same equation is used for the reflectors, but in this case there is no radiation effect and the conductivity in this case is roughly constant at 40 W/(m·K) [46].

3. Calculation of $q_{Fission}$. To calculate $q_{Fission}$, it is assumed that the power distribution in the pebble bed region is constant over time. Then $q_{Fission}$ for the control volume in the axial interval i (starting at the top of the core) and in the radial interval j (starting at the center of the core) can be calculated by using

the following equation:

$$q_{Fission,i,j} = \alpha_i \alpha_j Q_T \tag{4.6}$$

where Q_T is the total reactor power, and α_i and α_j are the normalized fission power factors in the axial and radial directions, respectively. The α_i factors correspond to the eight axial core intervals and their values are (from top to bottom): 0.025, 0.034, 0.053, 0.071, 0.091, 0.121, 0.148, 0.171, 0.162, 0.124 [60]. The radial α_j factors correspond to the six radial core intervals and their values are (from center to outer): 0.0 (passive section), 0.218, 0.207, 0.195, 0.19, 0.19 [60].

The nuclear reactor power can be calculated as a function of the neutronics of the reactor using the point kinetics equations. These equations use the mean values of the different variables in the reactor to calculate the total reactor power (Q_T) [21]. In this study, we use a simplified version:

$$Q_T = \omega_f V \Sigma_f \phi, \tag{4.7}$$

$$\frac{d\phi}{dt} = \frac{\rho - \beta}{\Lambda}\phi + \upsilon \lambda C_L, \tag{4.8}$$

$$\frac{dC_L}{dt} = \frac{\beta}{\Lambda}\phi - \lambda C_L. \tag{4.9}$$

In these equations Q_T, ϕ, C_L and ρ are variables. Eq. (4.7) calculates the total reactor power Q_T in terms of the average neutron flux density ϕ. The value of the neutron density ϕ is calculated using Eq. (4.8), and it depends on its own value, the reactivity ρ and the lumped delayed precursor concentration C_L. Eq. (4.8) does not consider a term to represent the control rods, because the accidents considered are unprotected. C_L lumps the six precursor groups used traditionally in the point kinetics equations, so only one equation, Eq. (4.9), is needed to represent them.

ω_f, V, Σ_f, λ, β and Λ are parameters. ω_f is the usable energy released per fission event (200 MeV), V is the active core volume, Σ_f is the macroscopic fission cross section (0.133 m^2), λ is the decay constant of the delayed precursor

(0.0584), and β is the total effective delayed-neutron fraction (0.005), and Λ is the prompt-neutron life time (4.0 $\times 10^{-4}$ s).

The reactivity ρ, the remaining variable to calculate, can be calculated from

$$\rho = \Delta\rho_X + \Delta\rho_T \tag{4.10}$$

where $\Delta\rho_X$ is the reactivity contribution from the fission product poisoning by Xenon (Xe^{135}) and $\Delta\rho_T$ is the reactivity contribution from the core temperature feedback. $\Delta\rho_X$ can be eliminated in this study, because its effects are very slow [60] in relation to the timeframe of interest (1 min to 1 hr). $\Delta\rho_T$ is calculated using

$$\Delta\rho_T = \alpha_T(T_{Core} - T_{Core,ref}). \tag{4.11}$$

T_{Core} is the average temperature in the core and a variable; it is calculated using $T_{Solid,i,j}$ with i and j corresponding to the core axial and radial intervals. α_T and $T_{Core,ref}$ are parameters. α_T is the temperature coefficient of reactivity (-3.5×10^{-5}) and $T_{Core,ref}$ is the reference temperature (999.075 K) [60].

4. Calculation of $q_{Convection}$. $q_{Convection}$ is the heat transferred to the helium gas by the pebbles. Given the high flow rates, it is assumed that radial flow can be ignored and only axial flow takes place in each node. Then the convected heat in each node (i,j) can be calculated by:

$$q_{Convection,i,j} = F_{i,j}C_{pg,i,j}\left(T_{g,out,i,j} - T_{g,in,i,j}\right) \tag{4.12}$$

where $F_{i,j}$ is the helium mass flowrate, and $T_{g,in,i,j}$ and $T_{g,out,i,j}$ are the temperature of the helium gas entering and exiting the node, respectively. These variables are calculated using the equations in the following section that explains the models for the flow of helium inside the core. $C_{pg,i,j}$ is the helium heat capacity parameter.

Now each of the terms in Eq. (4.1) is defined and they can be discretized using finite differences to find an expression for the temperature $T_{Solid,i,j}$ in each control volume in the core.

Temperature and pressure models for the helium flow in the reactor

This section explains how the temperature and pressure of the helium flowing through the core are calculated at each node. The flow of helium in the core, F_{Total} is an input variable, and the flow of helium at each node $F_{i,j}$ is calculated as a fraction of the total flowrate inside the core. The temperature of the gas in the core is calculated using an analytical expression for the heat equilibrium at each node:

$$T_{g,out,i,j} = T_{g,in,i,j} + (T_{Solid,i,j} - T_{g,in,i,j}) \left[1 - \exp\left(-\frac{hA_c}{F_{i,j}C_{pg,i,j}}\right)\right] \tag{4.13}$$

where $T_{g,out,i,j}$, $T_{g,in,i,j}$, $T_{Solid,i,j}$, $F_{i,j}$, and $C_{pg,i,j}$ are the same variables and parameters as defined before. h is the convective heat transfer coefficient calculated according to Wang [60] and A_c is the total heat transfer area in a node. To calculate the pressure drop in the core we use

$$\frac{R_{\Delta P}}{F_{Total}^2} = \frac{R_{\Delta P,ref}}{F_{Total,ref}^2} \tag{4.14}$$

with

$$R_{\Delta P} = \frac{P_{in} - P_{out}}{P_{in}} \tag{4.15}$$

P_{in} and P_{out} are the pressure values at the reactor's inlet and outlet, respectively. $R_{\Delta P}$ is the pressure drop ratio for the core and F_{Total} is the total helium mass flowrate inside the core as defined before. $R_{\Delta P,ref}$ and $F_{Total,ref}$ are the values of $R_{\Delta P}$ and F_{Total} at the designed operating point.

4.3.3 Heat Transfer Loop Model

The heat transfer loop is the other part of system that needs to be modeled in detail for this study. The model for the heat transfer loop is similar to the one explained in the previous chapter, and it is based on the design from INL specified in the

previous subsection [8]. The heat transfer model needs to represent two main type of phenomena: the gas dynamics of helium in the pipes, and the heat exchange with the nuclear reactor loop and the chemical plant.

The models used to represent the behavior of pipes, compressor and heat exchanger are the same ones developed in Chapter 3. In particular, to represent the gas dynamics in the loop the *QSS Approximation* model was used in combination with the correct coupling conditions.

4.4 Transient simulations

The model of the nuclear hydrogen plant developed in the previous section was tested against several different accident scenarios. Nuclear reactors and their behavior in response to accidents have been extensively studied in the past, because this is fundamental to obtain regulatory approval. However, their coupling to a chemical plant adds complexity and the transient behavior of this new system is not immediately obvious. To show that this new system is safe, this study focused on understanding how sudden changes in the chemical plant and heat transfer loop affect the nuclear reactor. We simulated three scenarios that can easily occur and might have an impact on the nuclear reactor:

1. a loss of heat sink,

2. a loss of flow in the heat transfer loop, and

3. a helium leak.

We focused on the effect of the different scenarios on the temperature in the nuclear reactor core, in terms of magnitude and time scale. We also explored other interesting variables depending on the scenario. The following sections explain how each of this scenarios were implemented and how the system responded.

4.4.1 Loss of heat sink accident

The loss of heat sink accident occurs when the chemical plant side loses its capacity to remove heat from the system at the PHX. This failure can be triggered by problems with valves, pumps or membranes at the electrolysis unit, or problems with the supply of steam. To model it, the steam flowrate was ramped down by 99% over 15 s. The system's response to this accident was studied by tracking the temperature and other relevant variables in the heat transfer loop and in the nuclear reactor. The accident causes minor effects on the nuclear reactor because of the reactor's inherent safety, but it has the potential to cause structural damage to the heat transfer loop.

1. *Heat transfer loop Response.* After the steam flowrate reduction at the PHX, the helium temperature stays within specifications at the IHX, hot leg and PHX. To understand the helium temperature's behavior against specifications along the IHX, hot leg and PHX, it suffices to study the maximum helium temperature in each section. These maximum temperatures are close to each other in a range of 10°C, so a good proxy for them is the helium temperature at the end of the IHX. This is the maximum helium temperature reached in the loop, and it increases slightly shortly after the steam reduction event (first 100 s). Then it drops to reach 851°C, which is 27°C colder than before the event is introduced (Fig. 4-3). This behavior is explained by the safe response of the nuclear reactor, which is explained in the next subsection.

   In contrast, the maximum helium temperature along the cold leg increases more than 330°C over the design point, reaching 908°C. The reduction of the cold stream at the PHX decreases the PHX's heat removal capacity, which leads to an increased temperature at the PHX's outlet (Fig. 4-3). Thus, the temperature increase at the PHX's outlet is over 310°C. This increase in temperature is amplified by the compressor, and the cold leg receives gas that is 330°C hotter than during normal operation.

   The high temperature reached in the cold leg decreases the stress resistance of the pipe's alloy below safety levels. The pressure difference between the inside

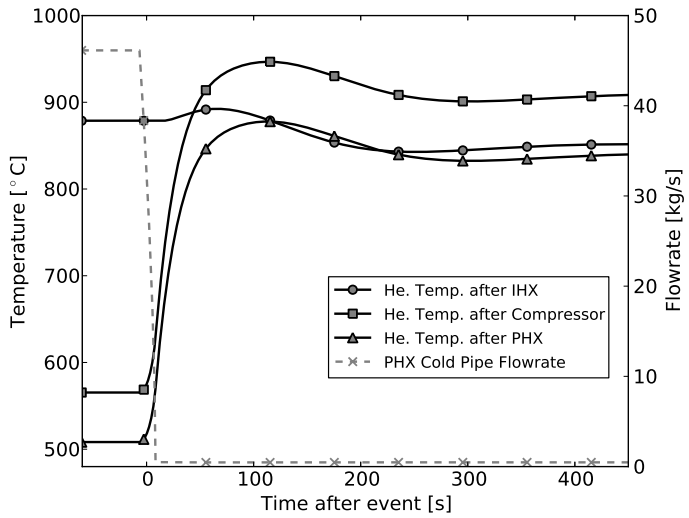

Figure 4-3: After a loss of heat sink accident, the temperature in the cold leg of the heat transfer loop increases more than 330°C.

of the pipe (2 MPa) and the outside (0.1 MPa) results in a stress on the pipe walls of 120 MPa [8]. The pipe design can easily tolerate this stress at the operating temperature (approx. 560°C) because it is made of Alloy 800 [33]. This alloy has a creep rupture strength in those conditions of 240 MPa for 10^5 hrs. However, when the temperature in the pipe reaches over 870°C, the creep rupture strength decreases dramatically. In these conditions, the pipe can only sustain 120 MPa for less than 1 hr (Fig. 4-4).

The lower stress resistance in the cold pipe could cause the pipe to rupture with serious consequences for the heat transfer loop but not for the reactor. The rupture of the cold pipe would not only damage the cold pipe, but also the surrounding hot pipe, compressor, PHX and IHX. This will render the whole system unusable and the heat sink capacity will be completely lost. Yet, the accident will not damage the reactor because the reactor's safety features can endure these changes without affecting the core. This is explained in the next section.

This accident can be prevented by using the hot pipe's material specifications to build the cold pipe, because the hot pipe is designed to withstand similar pressures at 900°C.

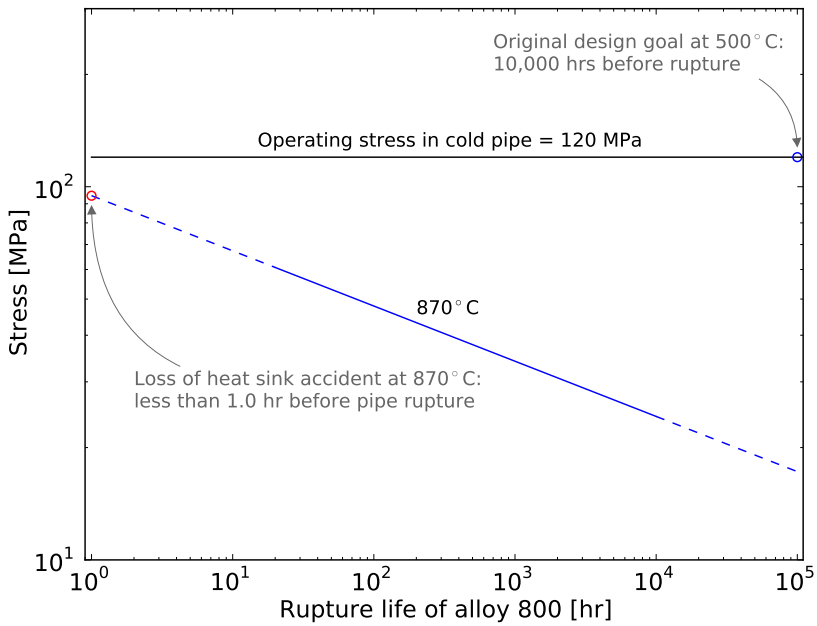

Figure 4-4: Increased temperature in cold pipe after loss of heat sink accident decreases pipe's stress resistance from over 10^5 hrs to less than 1 hr.

2. *Nuclear reactor response.* The temperature increase in the heat transfer loop's cold leg is propagated to the nuclear reactor's coolant loop, increasing the temperature of the coolant entering the core. The coolant temperature rises 260°C at the end of the IHX (Fig. 4-5) because of the loss of the heat sink at the heat transfer loop. This hotter coolant is mixed with the coolant coming at 540°C from the PCU, so the temperature of the coolant entering the core rises in only 65°C. The temperature of the coolant exiting the core actually drops because of the core response; this is explained in the next paragraph.

   The nuclear reactor core responds to the increased coolant temperature by decreasing its power. The increase in the coolant temperature entering the core slightly raises the average temperature of the fuel in the reactor core (Fig. 4-6). This temperature rise of the fuel introduces negative reactivity by increasing the fuel non-fission absorption for neutrons, which reduces the power produced and stabilizes the average temperature in the core (Fig. 4-6). Even though the average temperature in the core is the same, the lower power in the core changes the core temperature profile. The temperature at the coldest point in the core

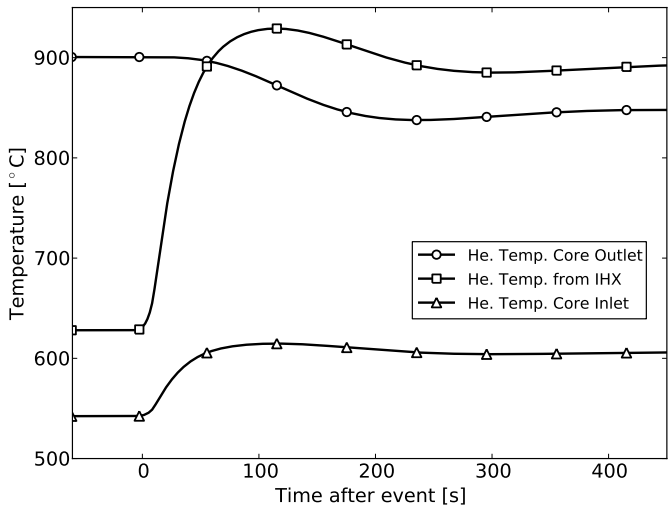

Figure 4-5: After a loss of heat sink accident, the helium entering the core is hotter, but its temperature is lower at the outlet.

(inlet) increases 60°C, and the temperature at the hottest point in the core (outlet) decreases 80°C. The lower core power explains the temperatures drop of the coolant at the core outlet and of the helium at the IHX outlet in the heat transfer loop.

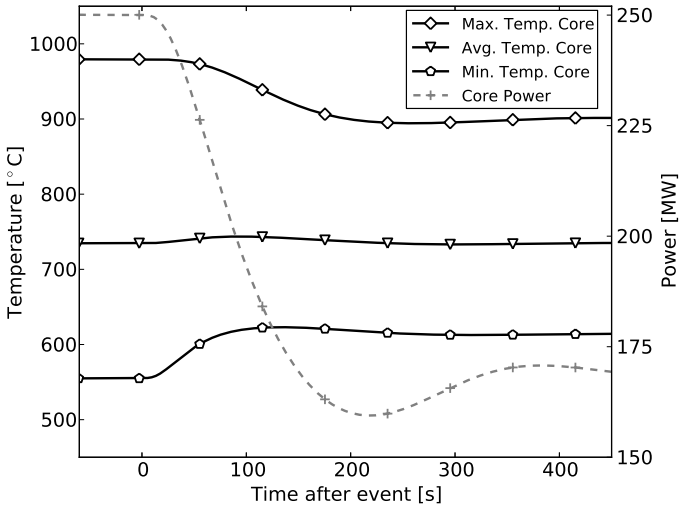

Figure 4-6: After a loss of heat sink accident, the average temperature in the fuel slightly increases, which leads to a reduction in the reactor power.

4.4.2 Loss of flow in the heat transfer loop

The loss of flow accident in the heat transfer loop happens when the compressor circulating helium inside the heat transfer loop stops. The lack of forced convection in the heat transfer loop halts the heat removal from the nuclear reactor's coolant loop. This failure can happen when the compressor surges or malfunctions. A similar accident is normally used to test the nuclear reactor's coolant loop. To represent this accident, the compressor flowrate was reduced to 1% of the operating flowrate over 5 s. The heat transfer loop and the reactor react safely in this scenario.

1. *Heat transfer loop response.* The lack of forced convection affects the helium temperature profile along the heat transfer loop, but this profile remains within design limits. The low helium flowrate along the PHX provides enough time to remove all the possible heat from it. As a result, the helium temperature at the end of the PHX drops from 510°C to 341°C (Fig. 4-7). On the other side of the loop, the helium temperature at the end of the IHX quickly increases 22°C and then drops below its normal operating temperature. This increase is induced again by the lower helium flowrate, which allows for a better heat exchange with the reactor coolant. The drop is a consequence of the reactor's power response, which is explained in the following paragraphs.

2. *Nuclear reactor response.* The low flowrate in the heat transfer loop stops the heat removal from the reactor coolant. The lack of heat removal at the IHX causes the coolant temperature at the IHX's outlet to rise initially from 628°C to 894°C (after 40 s) (Fig. 4-8). Afterwards, the coolant then reaches 850°C (after 400 s). This hotter coolant is mixed with the coolant coming from the PCU at 540°C. As a consequence, the temperature of the coolant entering the core rises 64° (after 40 s). This temperature rise produces a power response in the core, which results in a lower coolant temperature at the outlet of the core. The characteristics of this response are explained next.

   Again, the nuclear reactor core decreases its power in response to the higher coolant temperature at its inlet. The response of the core in this case is similar

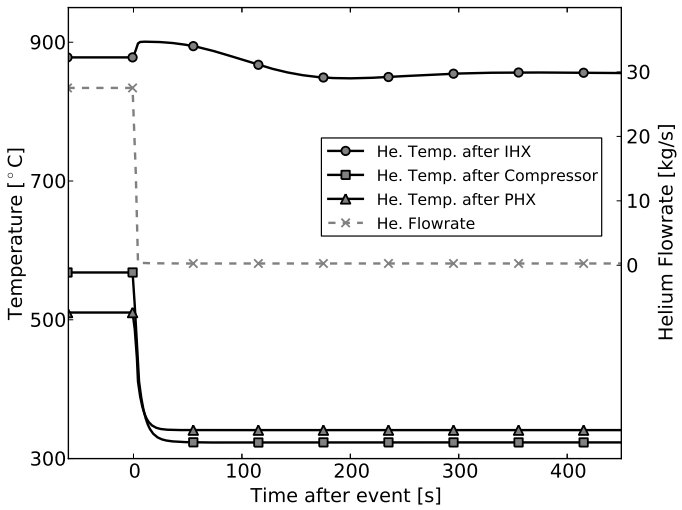

Figure 4-7: After a loss of flow in the heat transfer loop, the helium temperature around the loop stays within specifications.

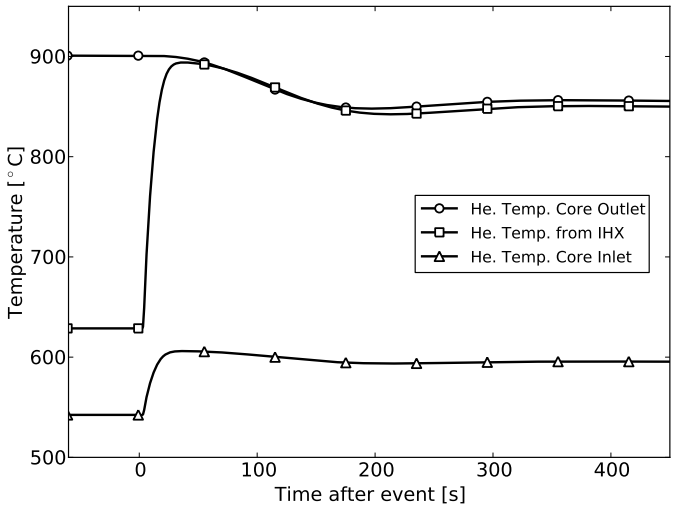

Figure 4-8: After a loss of flow in the heat transfer loop, the temperature of helium at the core inlet increases. The core response reduces the temperature at the core outlet.

to the response in the loss of heat sink accident. The higher coolant temperature increases slightly the average temperature of the core. As a response, the fuel's negative reactivity rises, which decreases the power produced in the core to 175 MWth (Fig. 4-9). Consequently, the temperature of the coolant at the core outlet is lower. So is the helium temperature at the outlet of the IHX's cold stream.

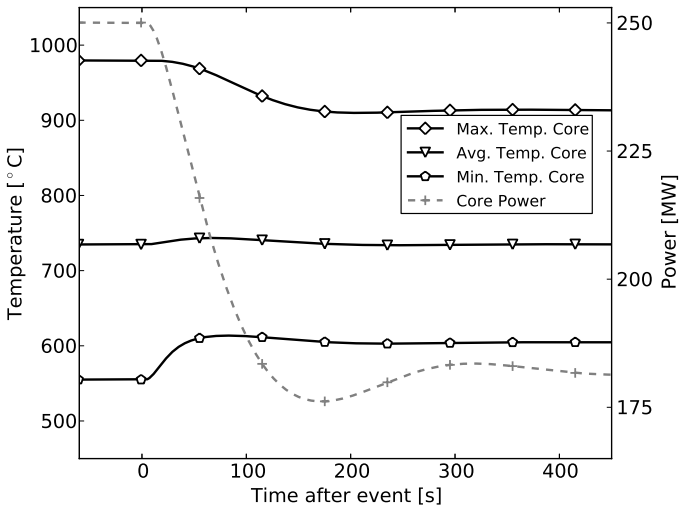

Figure 4-9: After a loss of flow in the heat transfer loop, the core reacts safely by reducing the fission power.

4.4.3 Leak in the heat transfer loop

Another potential accident that can affect the system is the loss of helium in the heat transfer loop. This failure can occur when a pipe leaks or a valve malfunctions. It results in less helium mass in the loop, which makes the loop less capable to remove heat from the nuclear reactor. To represent this accident, the total mass in the loop was linearly decreased by 50% over 1 min.

1. *Heat transfer loop response.* After introducing the leak, the heat transfer loop's response is similar, but milder, to the response to the loss of flow accident. These responses are similar because the loss of compressor power and the loss of helium mass have the same effect: a lower helium flowrate in the loop. The lower flowrate in this accident causes the helium temperature at the end of the PHX to drop from 508°C to 353°C in 400 s. On the other side of the loop, the temperature of helium at the end of the IHX rises from 878°C to 898°C in the first 70 s. The response in this case is milder compared to the response in the previous accident because the loop still has 50% of the original flowrate and can still remove a significant amount of heat from the nuclear reactor coolant. Then this temperature drops to 890°C after 400 s from the event introduction;

this drop is explained by the nuclear reactor response.

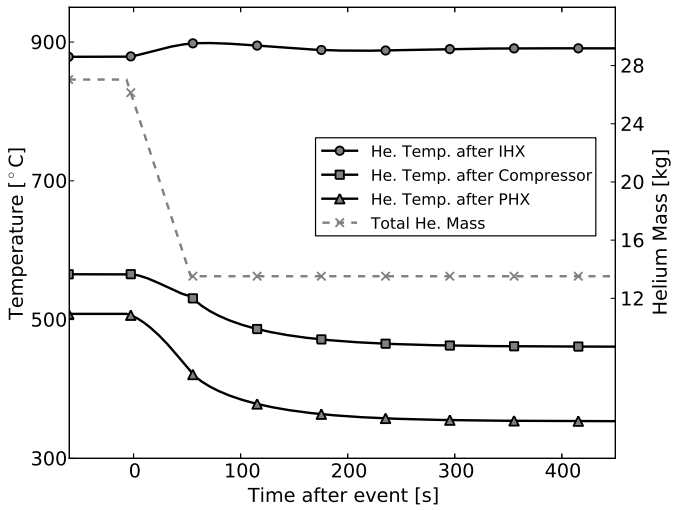

Figure 4-10: After a leak accident in the heat transfer loop, the temperature of helium along the loop stays within specifications.

The lower pressure inside the loop caused by the loss of helium increases the stress on the IHX, but the design specifications are enough to resist this stress. The IHX's walls experience a large pressure differential because the coolant from the nuclear reactor has a pressure of 7 MPa and the helium in the heat transfer loop has a pressure of 2 MPa. To withstand this pressure differential, the IHX's walls are designed using a pitch to channel diameter ratio of 1.5. This ratio is calculated as

$$\frac{p}{d} = 1 + \frac{\Delta P}{\sigma_D} \tag{4.16}$$

where ΔP is the pressure differential inside the IHX (5 MPa) and σ_D corresponds to the allowable stress over the internal walls of the IHX. σ_D is set as 10 MPa, which corresponds to half of the creep rupture stress of Alloy 800 at 900°C for 10^5 hrs [33]. ΔP increases to 6.1 MPa after the leak, because the pressure on the cold side of the IHX drops to 0.9 MPa. Under this new pressure difference, the allowable stress in the IHX's walls rises to 12.2 MPa, which is still below the creep rupture strength. Thus, this accident does not pose risks to the heat transfer loop.

2. *Nuclear reactor response.* The lower heat removal capacity in the heat transfer loop increases the temperature of the coolant entering the core. The temperature of the coolant at the end of the IHX rises from 628°C to 673°C after 400 s. This temperature change is propagated to the coolant at the core inlet, whose temperature rises from 542°C to 553°C. At the core outlet, the coolant temperature drops from 900°C to 891°C because the reactor power reduction in response to the higher coolant temperature at the core inlet.

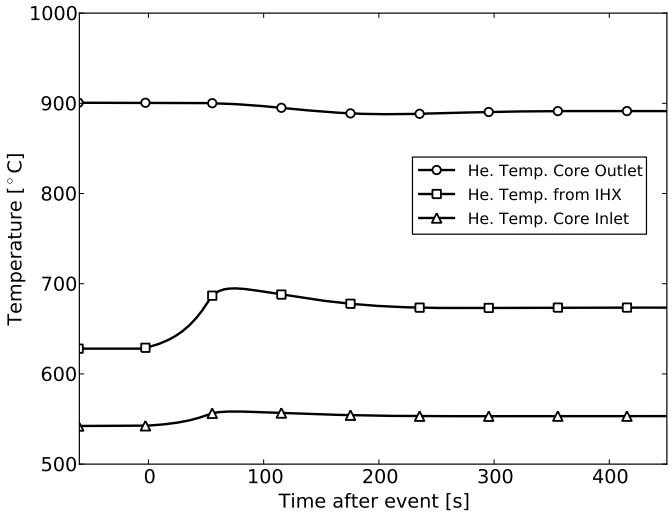

Figure 4-11: After a leak accident in the heat transfer loop, the temperature at the core inlet slightly increases, which results in a minor temperature decrease at the core outlet.

The core power reduction in this case is lower than in the previous accidents. The slightly hotter coolant at the inlet of the core increases the average temperature and the negative reactivity in the core (Fig. 4-12). This reduces the power in the reactor in 24 MWth. The lower power reduction compared to the previous accident is explained by the higher heat removal capacity remaining in the heat transfer loop.

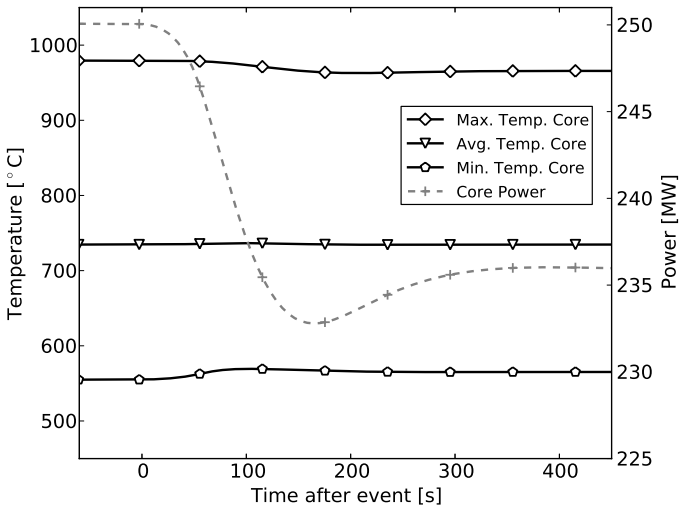

Figure 4-12: After a leak accident in the heat transfer loop, the heat transfer loop still can remove a significant amount of heat from the nuclear reactor coolant. This results in a power reduction of 24 MWth.

4.5 Conclusion

The nuclear hydrogen production plant studied is proven to have good safety features in the scenarios tested, but changes to the design might be needed. All the scenarios tested—directly or indirectly—lead to a loss of heat sink capacity at the nuclear reactor. This resulted in an increase of the nuclear reactor core temperature, which was quickly stopped by the fission power reduction at the fuel pebbles and by the safe design of the PBMR. As a result, the maximum temperature reached in the core was always less than 1000°C, which is below the melting point of the fuel pebbles (2000°C). Thus, the nuclear reactor was always safe. At the heat transfer loop, pressure and temperature remained within specified limits in two of the scenarios tested. However, in the *loss of heat sink* scenario the temperature in the cold leg increased by more than 330°C, which could result in the rupture of this pipe. This could be prevented by replacing the design specifications of the cold leg with those of the hot leg.

Adding more details to the studied nuclear hydrogen plant model would help to understand its behavior further. This study's purpose was representing the response to accidents of a nuclear hydrogen plant, and in particular, it focused on the nuclear reactor and heat transfer loop. Consequently, these two units were modeled in detail.

In contrast, the PCU was assumed as a set of external conditions that remained constant, and the interaction with the HTSE unit was represented by the steam flowing through the PHX. Including a model of the PCU would help to verify the simplifying assumption, or show potential disruptions in its operation during accidents. At the same time, including a model of the HTSE unit would help to study the behavior of the plant during normal operational transients (e.g., start-up, shutdown, and restart). The SOECs in the HTSE unit are very sensitive to thermal gradients and it will be particularly important to include this behavior in operational transients.

Finally, the analysis of the nuclear hydrogen plant behavior in this chapter has shown that equation-based simulators can successfully simulate transients of new nuclear applications. Equation-based simulators are a good platform to simulate this kind of process because they are flexible, extendable, fast and accurate. Also, they can be easily audited by regulators, because they are open and declarative modeling environments. Future safety studies should use them and focus on adding more details to the plant model and on representing the plant's normal operation. These efforts could demonstrate the safety of nuclear hydrogen production using HTSE, and help develop a cleaner way to produce liquid fuels based on alternative feedstocks.

Chapter 5

Conclusions

This thesis has shown that nuclear hydrogen production can be safe, and that equation-based simulators are a good platform to test the safety of this kind of facility. Nuclear hydrogen production is one of the new nuclear applications that could have a large impact in the production liquid fuels, but its safety needs to be demonstrated using computational simulations. For that, a nuclear hydrogen production plant was represented using equation-based simulators, and models for all its units were created. The models were tested in different accident scenarios showing that the system was safe. Here, these results are summarized and potential future work is proposed.

The second chapter of this thesis explained the rational for new nuclear applications supporting liquid fuel production, and the need for equation-based simulators to demonstrate their safety. Using more local and affordable feedstocks—such as oil sands, coal or biomass—to produce liquid fuels requires large amounts of energy and generates significant CO_2 emissions. These issues can be resolved by using nuclear hydrogen production and nuclear heat integration with production facilities. It is nuclear hydrogen production, however, that offers the most possibilities, as it is the most versatile to use. Thus, this technology was selected to study the safety in new nuclear applications. Unfortunately, the study of the safety of these new applications cannot be done with legacy codes used for the analysis of nuclear power plants. The reason for this is that legacy codes lack the required flexibility, computational speed and accuracy to represent the different chemical and physical phenomena. Instead,

equation-based simulators are the appropriate simulation platform, because they have been designed to represent complex and transient processes such as nuclear hydrogen production.

Chapter 3 proposed a set of equation-based models to simulate the heat transfer loop in a nuclear hydrogen plant and found that the models based on the *Zero Mach Number Limit* model and the *QSS Approximation* model for gas dynamics were suitable to model the interesting transients. The heat transfer loop uses helium as the heat transfer fluid, and this makes simulating its behavior difficult. To resolve this, three models for gas dynamics and two set of coupling conditions for boundary variables were tested in JACOBIAN, an equation-based simulator. The *Zero Mach Number Limit* model and the *QSS Approximation* model in combination with a new approach to set coupling conditions for boundary variables were able to represent the interesting time scales accurately in transient scenarios. These simulations were confirmed with a reference model written in the legacy code RELAP. Additionally, the computation of these simulations in JACOBIAN was faster and more accurate than the computation of the simulations in RELAP.

Chapter 4 used equation-based models to show the safety of a nuclear hydrogen production plant during accidents. The analysis of the hydrogen production plant focused on the nuclear reactor and heat transfer loop, and detailed dynamic models for these units were created. This plant model was then subjected to three accident scenarios: a loss of heat sink at the HTSE, a loss of flow in the heat transfer loop, and a leak in the heat transfer loop. All the scenarios—directly or indirectly—lead to a loss of heat sink capacity at the nuclear reactor. This resulted in an increase of the nuclear reactor core temperature, which was quickly moderated by the fission power reduction in the fuel pebbles and by the safe design of the PBMR. As a result, the maximum temperature reached in the core was always less than 1000°C, which is below the melting point of the fuel pebbles (2000°C). Thus, the nuclear reactor was always safe. At the heat transfer loop, pressure and temperature remained within specified limits in two of the scenarios tested. However, in the *loss of heat sink* scenario the temperature in the cold leg increased by more than 330°C, which could

result in the rupture of this pipe. This could be prevented by replacing the design specifications of the cold leg with those of the hot leg.

Future work in this area involves testing other scenarios and expanding the models used. This preliminary study of the safety in a nuclear hydrogen production plant was focused on the behavior of the plant during selected accidents. In particular, emphasis was on the behavior of the nuclear reactor and the heat transfer loop. However, to show a complete picture of the safety of the plant, accidents involving the operation of the PCU and HTSE unit should be included. To do this, the dynamic model of the plant will need to incorporate models representing the different parts in the PCU (e.g., turbine, compressor, generator) and in the HTSE unit (e.g., SOEC, separators). Additionally, the behavior of the system during normal operational transients should also be tested (e.g., start-up, shut-down, and re-start) to guarantee safety and operability. Indeed, the SOECs in the HTSE unit are very sensitive to thermal gradients and it will be particularly important to include this behavior in operational transients. Expanding and modifying the nuclear hydrogen plant model created in this thesis could be easily done, for these models have been created in equation-based simulators with flexibility and extendability features.

This thesis' results show that nuclear hydrogen production plants could be safe, and simultaneously, that equation-based simulators are good platforms to prove the safety of this plants. Developing these models and tests further will help to guarantee the safety of the plant and obtain regulatory and public approval. In the end, this will help to transition from the current liquid fuel supply chain based on foreign oil, which is polluting, expensive and risky, to a liquid fuel supply chain based on more affordable, local and secure feedstocks. In particular, using biomass as a feedstock in combination with nuclear hydrogen production and nuclear heat integration could achieve liquid fuel supply chain that will be clean, sustainable, efficient and local.

Appendix A

Discretization of Gas Dynamic Equations

The partial differential algebraic equations used to describe gas dynamics in the loop were the following:

$$\text{Continuity} \qquad \frac{\partial \rho}{\partial t} = -\frac{\partial \phi}{\partial x}, \tag{A.1}$$

$$\text{Momentum} \qquad \frac{\partial \phi}{\partial t} = -\frac{\partial P}{\partial x} - \frac{\partial}{\partial x}\frac{\phi^2}{\rho} - \frac{2f\left|\phi\right|\phi}{D\rho}, \tag{A.2}$$

$$\text{Energy} \qquad \frac{\partial \psi}{\partial t} = -\frac{\partial}{\partial x}\phi h - \frac{4U\left(T - T_{Ext}\right)}{D}, \tag{A.3}$$

$$\text{Physical Properties} \qquad \psi = \rho h - P, \tag{A.4}$$

$$h = h_{form} + \frac{C_P}{M}\left(T - T_{stand}\right), \tag{A.5}$$

$$P = \frac{R_C \rho T}{M}. \tag{A.6}$$

To simplify the writing of these equations, ϕ was replaced by $\frac{F}{A}$; then the equations are the following:

$$\text{Continuity} \qquad \frac{\partial \rho}{\partial t} = -\frac{\partial}{\partial x}\frac{F}{A}, \tag{A.7}$$

$$\text{Momentum} \qquad \frac{\partial}{\partial t}\frac{F}{A} = -\frac{\partial P}{\partial x} - \frac{\partial}{\partial x}\frac{F^2}{\rho A^2} - \frac{2f\,|F|\,F}{D\rho A^2}, \tag{A.8}$$

$$\text{Energy} \qquad \frac{\partial \psi}{\partial t} = -\frac{\partial}{\partial x}\frac{Fh}{A} - \frac{4U\left(T - T_{Ext}\right)}{D}, \tag{A.9}$$

$$\text{Physical Properties} \qquad \psi = \rho h - P, \tag{A.10}$$

$$h = h_{form} + \frac{C_P}{M}\left(T - T_{stand}\right), \tag{A.11}$$

$$P = \frac{R_C \rho T}{M}. \tag{A.12}$$

These equations were discretized using a combination of a standard and a staggered grid (Fig. A-1).

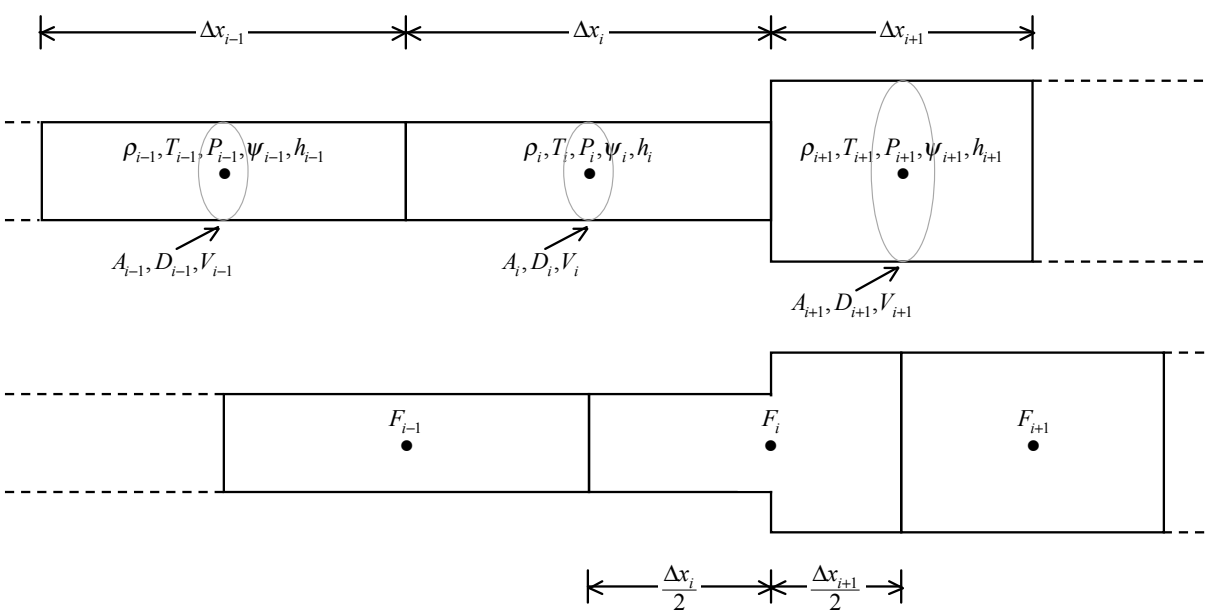

Figure A-1: A combination of standard grid and a staggered grid is used to discretize the gas dynamics equations.

In this grid, the values of the discretized differential variables ρ_i and ψ_i were assumed to be the average value of ρ and ψ over the control volume i in the standard

grid. Then these variables can be represented as

$$\rho_i = \frac{1}{V_i} \int_{V_i} \rho dV, \tag{A.13}$$

$$\psi_i = \frac{1}{V_i} \int_{V_i} \psi dV. \tag{A.14}$$

Similarly, the value of F_i is defined as the average value of F over the control volume i in the staggred grid,

$$F_i = \frac{1}{\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}} \int_{\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}} F dx. \tag{A.15}$$

The values of the algebraic variables calculated using the algebraic equations definitions:

$$\psi_i = \rho_i h_i - P_i, \tag{A.16}$$

$$h_i = h_{form} + \frac{C_P}{M} \left(T_i - T_{stand} \right), \tag{A.17}$$

$$P_i = \frac{R_C \rho_i T_i}{M}. \tag{A.18}$$

The following explains how Eq. (A.7)-(A.9) were discretized.

A.1 Discretization of the continuity equation

The continuity equation,

$$\frac{\partial \rho}{\partial t} dV = -\frac{\partial}{\partial x} \frac{F}{A} dV, \tag{A.19}$$

was integrated over each control volume in the standard grid. The ith control volume is used to explain the discretization.

Integrating this equation over the volume of the ith control volume produces the

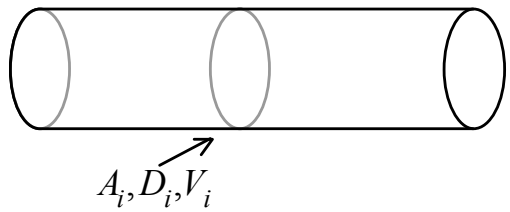

Figure A-2: i^{th} control volume in the standard grid.

equation

$$\underbrace{\int_{V_i} \frac{\partial \rho}{\partial t} dV}_{\text{Term (1)}} = -\underbrace{\int_{V_i} \frac{\partial}{\partial x} \frac{F}{A} dV}_{\text{Term (2)}} . \tag{A.20}$$

Term (1) can be simplified by using the fact that V_i is constant over time,

$$\int_{V_i} \frac{\partial \rho}{\partial t} dV = \frac{\partial}{\partial t} \int_{V_i} \rho dV. \tag{A.21}$$

Then, we can use the definition of ρ_i to get the discretized approximation of Term (1):

$$\frac{\partial}{\partial t} \int_{V_i} \rho dV = \frac{\partial \rho_i}{\partial t} V_i. \tag{A.22}$$

Term (2) can be calculated by using the divergence theorem:

$$\int_{V_i} \frac{\partial}{\partial x} \frac{F}{A} dV = \int_{S_i} \frac{F}{A} \left(\mathbf{n} \cdot \mathbf{e}_x \right) dS. \tag{A.23}$$

In this case, S_i corresponds to the sum of the surfaces surrounding the ith control volume [$S_i = S_1 + S_2 + S_3$ (Fig. A-3)].

The right hand side of Eq. (A.23) can be separated into the integrals over each surface surrounding the ith control volume and the resulting expression can be sim-

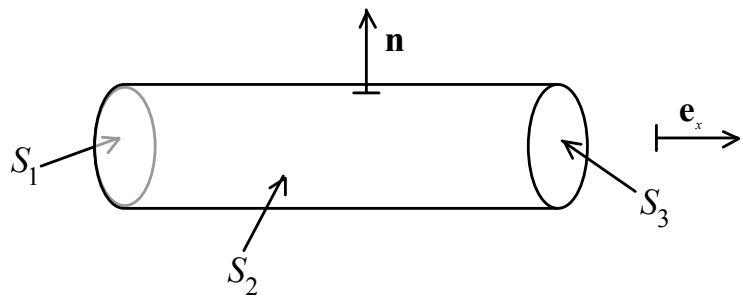

Figure A-3: i^{th} control volume in the standard grid.

plified:

$$\int_{S_i} \frac{F}{A}(\mathbf{n}\cdot\mathbf{e}_x)dS = \int_{S_1} \frac{F}{A}(\underbrace{\mathbf{n}\cdot\mathbf{e}_x}_{-1})dS + \int_{S_2} \frac{F}{A}(\underbrace{\mathbf{n}\cdot\mathbf{e}_x}_{0})dS + \int_{S_3} \frac{F}{A}(\underbrace{\mathbf{n}\cdot\mathbf{e}_x}_{1})dS$$
$$= -\int_{S_1} \frac{F}{A}dS + \int_{S_3} \frac{F}{A}dS. \tag{A.24}$$

Approximating F at S_1 by F_{i-1} and F at S_3 we obtain

$$\int_{S_i} \frac{F}{A}(\mathbf{n}\cdot\mathbf{e}_x)dS = -F_{i-1} + F_i. \tag{A.25}$$

Finally, we can write the discretized continuity equation as

$$\frac{d\rho_i}{dt}V_i = F_{i-1} - F_i. \tag{A.26}$$

A.2 Discretization of the momentum equation

The momentum equation,

$$\frac{\partial}{\partial t}\frac{F}{A} = -\frac{\partial P}{\partial x} - \frac{\partial}{\partial x}\frac{F^2}{\rho A^2} - \frac{2f\,|F|\,F}{D\rho A^2}, \tag{A.27}$$

is discretized using the staggered grid. The most general case corresponds to a staggered control volume that includes a change in section (Fig. A-4).

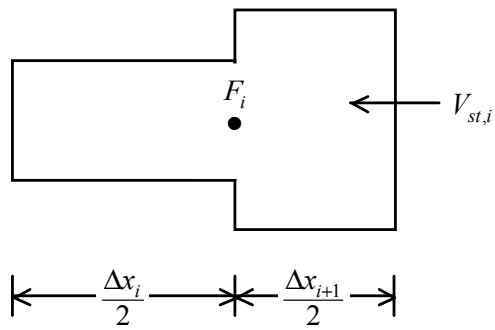

Figure A-4: i^{th} control volume in the staggered grid.

Integrating Eq. (A.27) over $V_{st,i}$ produces

$$\underbrace{\int_{V_{st,i}} \frac{\partial}{\partial t}\frac{F}{A}dV}_{\text{Term (1)}} = -\underbrace{\int_{V_{st,i}} \frac{\partial P}{\partial x}dV}_{\text{Term (2)}} - \underbrace{\int_{V_{st,i}} \frac{\partial}{\partial x}\frac{F^2}{\rho A^2}dV}_{\text{Term (3)}} - \underbrace{\int_{V_{st,i}} \frac{2f\,|F|\,F}{D\rho A^2}dV}_{\text{Term (4)}} \tag{A.28}$$

Term (1) can be simplified by considering that $V_{st,i}$ is constant over time, and by dividing the control volume into two control volumes with different section (Fig. A-8).

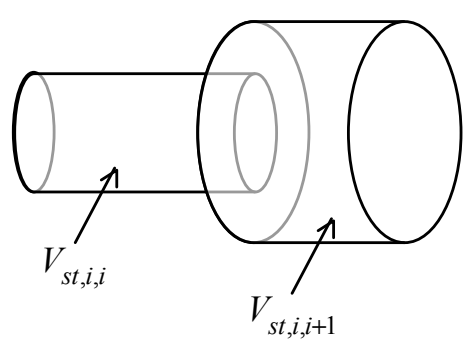

Figure A-5: i^{th} control volume in the staggered grid.

$$\int_{V_{st,i}} \frac{\partial}{\partial t}\frac{F}{A}dV = \frac{\partial}{\partial t}\int_{V_{st,i}} \frac{F}{A}dV \tag{A.29}$$

$$= \frac{\partial}{\partial t}\left(\int_{V_{st,i,i}} \frac{F}{A}dV + \int_{V_{st,i,i+1}} \frac{F}{A}dV\right) \tag{A.30}$$

$$= \frac{\partial}{\partial t}\left(\int_{\frac{\Delta x_i}{2}} \frac{F}{A_i}A_i dx + \int_{\frac{\Delta x_{i+1}}{2}} \frac{F}{A_{i+1}}A_{i+1} dx\right) \tag{A.31}$$

$$= \frac{\partial}{\partial t}\int_{\frac{\Delta x_i}{2}+\frac{\Delta x_{i+1}}{2}} F dx \tag{A.32}$$

$$= \frac{\partial}{\partial t}\left[\left(\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}\right)F_i\right] \tag{A.33}$$

$$= \left(\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}\right)\frac{\partial F_i}{\partial t} \tag{A.34}$$

Term (2) can be approximated by using the divergence theorem and by decomposing the surface surrounding the ith control volume in the staggered grid (Fig. A-6).

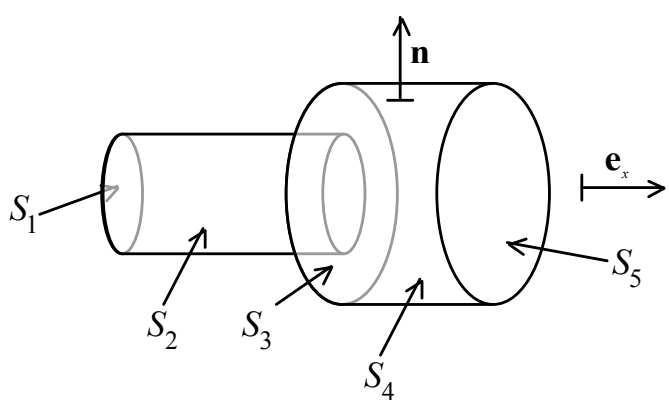

Figure A-6: i^{th} control volume in the staggered grid with a change in section.

$$\int_{V_{st,i}} \frac{\partial P}{\partial x} dV = \int_{S_{st,i}} P \mathbf{n} \cdot \mathbf{e}_x dS \tag{A.35}$$

Considering that $S_{st,i}$ can be decomposed in five surfaces [$S_{st,i} = S_1 + S_2 + S_3 + S_4 + S_5$ (Fig. A-6)], then the previous equation can be written and simplified as

$$\int_{V_{st,i}} \frac{\partial P}{\partial x} dV = \int_{S_1} P \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{-1} dS + \int_{S_2} P \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{0} dS + \int_{S_3} P \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{-1} dS + \int_{S_4} P \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{0} dS + \int_{S_5} P \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{1} dS, \tag{A.36}$$

$$\int_{V_{st,i}} \frac{\partial P}{\partial x} dV = - \int_{S_1} P dS - \int_{S_3} P dS + \int_{S_5} P dS. \tag{A.37}$$

The three terms in the right hand side of Eq. (A.37) represent the forces exerted over the gas in the staggered control volume at the different vertical faces. In particular, the force at S_3 represents the force exerted by the wall over the gas as a reaction to the force exerted by the gas over the wall. This force is significant and it needs to be included to correctly model the behavior of the gas.

P at S_1 can be approximated by P_i and P at S_5 can be approximated by using P_{i+1}. P at S_3 can be approximated by $\overline{P}_{(i,i+1)}$, a linear interpolation between P_i and P_{i+1}:

$$\overline{P}_{(i,i+1)} = \frac{P_i \frac{\Delta x_{i+1}}{2} + P_{i+1} \frac{\Delta x_i}{2}}{\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}}. \tag{A.38}$$

Then Eq. (A.37) can be written as

$$\int_{V_{st,i}} \frac{\partial P}{\partial x} dV = - P_i A_i - \overline{P}_{(i,i+1)} (A_{i+1} - A_i) + P_{i+1} A_{i+1}. \tag{A.39}$$

Finally,

$$\int_{V_{st,i}} \frac{\partial P}{\partial x} dV = -P_i A_i - \frac{P_i \frac{\Delta x_{i+1}}{2} + P_{i+1} \frac{\Delta x_i}{2}}{\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}} (A_{i+1} - A_i) + P_{i+1} A_{i+1}. \tag{A.40}$$

Term (3) can be approximated using a similar approach:

$$\int_{V_{st,i}} \frac{\partial}{\partial x} \frac{F^2}{\rho A^2} dV = \int_{S_{st,i}} \frac{F^2}{\rho A^2} \mathbf{n} \cdot \mathbf{e}_x dS \tag{A.41}$$

$$= \int_{S_1} \frac{F^2}{\rho A^2} \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{-1} dS + \int_{S_2} \frac{F^2}{\rho A^2} \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{0} dS + \int_{S_3} \overbrace{\frac{F^2}{\rho A^2}}^{0} \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{-1} dS \tag{A.42}$$

$$+ \int_{S_4} \frac{F^2}{\rho A^2} \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{0} dS + \int_{S_5} \frac{F^2}{\rho A^2} \underbrace{\mathbf{n} \cdot \mathbf{e}_x}_{1} dS. \tag{A.43}$$

The resulting equation is

$$\int_{V_{st,i}} \frac{\partial}{\partial x} \frac{F^2}{\rho A^2} dV = -\int_{S_1} \frac{F^2}{\rho A^2} dS + \int_{S_5} \frac{F^2}{\rho A^2} dS. \tag{A.44}$$

In this case, the term at S_3 is discarded because there is no flow through that wall.

F at S_1 can be approximated by using a linear interpolation between F_{i-1} and F_i:

$$\overline{F}_{(i-1,i)} = \frac{F_{i-1} \frac{\Delta x_i}{2} + F_i \frac{\Delta x_i}{2}}{\frac{\Delta x_i}{2} + \frac{\Delta x_i}{2}} = \frac{F_{i-1} + F_i}{2}. \tag{A.45}$$

Similarly, the linear approximation of F at S_5 is

$$\overline{F}_{(i,i+1)} = \frac{F_i \frac{\Delta x_{i+1}}{2} + F_{i+1} \frac{\Delta x_{i+1}}{2}}{\frac{\Delta x_{i+1}}{2} + \frac{\Delta x_{i+1}}{2}} = \frac{F_i + F_{i+1}}{2}. \tag{A.46}$$

Considering that $S_1 = A_i$ and $S_5 = A_{i+1}$, and that ρ at S_1 and S_5 can be approx-

imated by ρ_i and ρ_{i+1} respectively then Eq. (A.44) can be written as

$$\int_{V_{st,i}} \frac{\partial}{\partial x}\frac{F^2}{\rho A^2}dV = -\int_{S_1} \frac{\overline{F}^2_{(i-1,i)}}{\rho_i A_i^2}dS + \int_{S_5} \frac{\overline{F}^2_{(i,i+1)}}{\rho_{i+1} A_{i+1}^2}dS \tag{A.47}$$

$$= -\frac{\overline{F}^2_{(i-1,i)}}{\rho_i A_i} + \frac{\overline{F}^2_{(i,i+1)}}{\rho_{i+1} A_{i+1}}. \tag{A.48}$$

Finally, Term (3) can be written as

$$\int_{V_{st,i}} \frac{\partial}{\partial x}\frac{F^2}{\rho A^2}dV = -\left(\frac{F_{i-1}+F_i}{2}\right)^2 \frac{1}{\rho_i A_i} + \left(\frac{F_i+F_{i+1}}{2}\right)^2 \frac{1}{\rho_{i+1} A_{i+1}}. \tag{A.49}$$

Term (4) can be calculated by using the ith control volume in the staggered grid. For this calculation, the control volume is separated into two subcontrol volumes with different section (Fig. A-7).

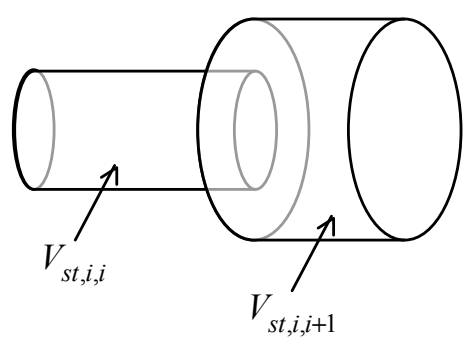

Figure A-7: i^{th} control volume in the staggered grid.

$$\int_{V_{st,i}} \frac{2f\,|F|\,F}{D\rho A^2}dV = \int_{V_{st,i,i}} \frac{2f\,|F|\,F}{D\rho A^2}dV + \int_{V_{st,i,i+1}} \frac{2f\,|F|\,F}{D\rho A^2}dV \tag{A.50}$$

It is assumed that the prevailing value for ρ in each of the subcontrol volumes is the value of ρ in the corresponding control volume in the standard grid. Then

Eq. (A.50) can be written as

$$\int\limits_{V_{st,i}} \frac{2f\left|F\right|F}{D\rho A^2}dV = \frac{2f_i}{D_i\rho_i A_i^2}\int\limits_{V_{st,i,i}} \left|F\right|FdV + \frac{2f_{i+1}}{D_{i+1}\rho_{i+1}A_{i+1}}\int\limits_{V_{st,i,i+1}} \left|F\right|FdV \tag{A.51}$$

$\int\limits_{V_{st,i,i}} \left|F\right|FdV$ and $\int\limits_{V_{st,i,i+1}} \left|F\right|FdV$ can be approximated by $\left|F_i\right|F_iA_i\frac{\Delta x_i}{2}$ and $\left|F_i\right|F_iA_{i+1}\frac{\Delta x_{i+1}}{2}$, respectively. Then Eq. (A.51) can be written as

$$\int\limits_{V_{st,i}} \frac{2f\left|F\right|F}{D\rho A^2}dV = \frac{2f_i}{D_i\rho_i A_i^2}\left|F_i\right|F_iA_i\frac{\Delta x_i}{2} + \frac{2f_{i+1}}{D_{i+1}\rho_{i+1}A_{i+1}^2}\left|F_i\right|F_iA_{i+1}\frac{\Delta x_{i+1}}{2} \tag{A.52}$$

$$= \frac{2f_i\left|F_i\right|F_i}{D_i\rho_iA_i}\frac{\Delta x_i}{2} + \frac{2f_{i+1}\left|F_i\right|F_i}{D_{i+1}\rho_{i+1}A_{i+1}}\frac{\Delta x_{i+1}}{2} \tag{A.53}$$

Having calculated all the terms for the momentum equation (Eq. (A.28)), the discretized momentum equation can be written as

$$\left(\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}\right)\frac{dF_i}{dt} = +P_iA_i + \frac{P_i\frac{\Delta x_{i+1}}{2} + P_{i+1}\frac{\Delta x_i}{2}}{\frac{\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}}(A_{i+1} - A_i) - P_{i+1}A_{i+1} \tag{A.54}$$

$$+ \left(\frac{F_{i-1} + F_i}{2}\right)^2\frac{1}{\rho_iA_i} - \left(\frac{F_i + F_{i+1}}{2}\right)^2\frac{1}{\rho_{i+1}A_{i+1}} \tag{A.55}$$

$$- \frac{2f_i\left|F_i\right|F_i}{D_i\rho_iA_i}\frac{\Delta x_i}{2} - \frac{2f_{i+1}\left|F_i\right|F_i}{D_{i+1}\rho_{i+1}A_{i+1}}\frac{\Delta x_{i+1}}{2} \tag{A.56}$$

A.3 Discretization of the energy equation

The energy equation,

$$\frac{\partial\psi}{\partial t} = -\frac{\partial}{\partial x}\frac{Fh}{A} - \frac{4U\left(T - T_{Ext}\right)}{D}, \tag{A.57}$$

is discretized using the standard grid and the ith control volume. Integrating Eq. (A.57) over this volume produces the following equation:

$$\underbrace{\int_{V_i} \frac{\partial \psi}{\partial t} dV}_{\text{Term (1)}} = \underbrace{-\int_{V_i} \frac{\partial}{\partial x} \frac{Fh}{A} dV}_{\text{Term (2)}} \underbrace{- \int_{V_i} \frac{4U}{D}(T - T_{ext}) dV}_{\text{Term (3)}} . \tag{A.58}$$

Term (1) can be simplified by using the fact that V_i is constant over time and by using the definition of ψ_i (Eq. (A.14)):

$$\int_{V_i} \frac{\partial \psi}{\partial t} dV = \frac{\partial}{\partial t} \int_{V_i} \psi dV = \frac{\partial \psi_i}{\partial t} V_i. \tag{A.59}$$

Term (2) can be calculated using the divergence theorem:

$$\int_{V_i} \frac{\partial}{\partial x} \frac{Fh}{A} dV = \int_{S_i} \frac{Fh}{A} \mathbf{n} \cdot \mathbf{e}_x dS. \tag{A.60}$$

Again, S_i corresponds to the sum of the surfaces surrounding the ith control volume [$S_i = S_1 + S_2 + S_3$ (Fig. A-8)].

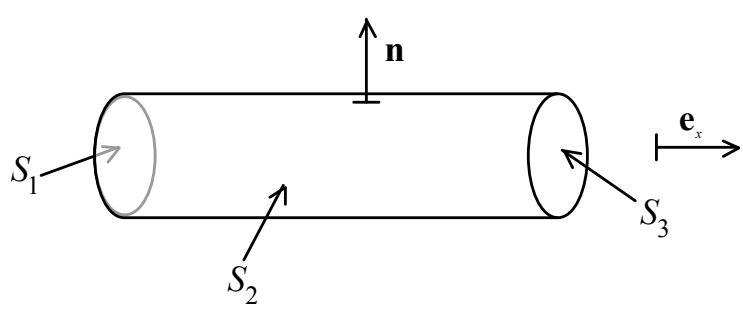

Figure A-8: i^{th} control volume in the standard grid.

Then, the right hand side of Eq. (A.60) can be separated into the integrals over each surface surrounding the ith control volume and the resulting expression can be

simplified:

$$\int_{V_i} \frac{\partial}{\partial x}\frac{Fh}{A}dV = \int_{S_1} \frac{Fh}{A}\underbrace{\mathbf{n}\cdot\mathbf{e}_x}_{-1}dS + \int_{S_2} \frac{Fh}{A}\underbrace{\mathbf{n}\cdot\mathbf{e}_x}_{0}dS + \int_{S_3} \frac{Fh}{A}\underbrace{\mathbf{n}\cdot\mathbf{e}_x}_{1}dS \tag{A.61}$$

$$= -\int_{S_1} \frac{Fh}{A}dS + \int_{S_3} \frac{Fh}{A}dS. \tag{A.62}$$

At S_1, A corresponds to A_{i-1}, F corresponds to F_{i-1}, and h corresponds to h_{i-1} (upwind differencing scheme [27]). At S_3, A corresponds to A_i, F corresponds to F_i, and h corresponds to h_i. Then Eq. (A.62) can be written as

$$\int_{V_i} \frac{\partial}{\partial x}\frac{Fh}{A}dV = -\frac{F_{i-1}h_{i-1}}{A_{i-1}}\underbrace{\int_{S_1} dS}_{A_{i-1}} + \frac{F_i h_i}{A_i}\underbrace{\int_{S_3} dS}_{A_i} \tag{A.63}$$

$$= -h_{i-1}F_{i-1} + h_i F_i. \tag{A.64}$$

Solving for Term (3). At the ith control volume U,D,T and T_{ext} can be approximated by using U_i, D_i, T_i and $T_{ext,i}$, respectively. Then Term (3) can be written as

$$\int_{V_i} \frac{4U}{D}(T - T_{ext})dV = \frac{4U_i}{D_i}(T_i - T_{ext,i})V_i. \tag{A.65}$$

Finally, the discretized energy equation is:

$$V_i\frac{d\psi_i}{dt} = h_{i-1}F_{i-1} - h_i F_i - \frac{4U_i}{D_i}(T_i - T_{ext,i})V_i. \tag{A.66}$$

Appendix B

JACOBIAN code

MODELS_PHYSICALPROPERTIES.JAC

```
##########################################################################
DECLARE
##########################################################################
TYPE    #Default    Min   Max   Units
  Area  = 1 : -1e10 : 1e2 UNIT = "m2"
  ConcPrecursors = 0:-1.0e-7 : 1.0e10 unit = ""
  DeltaTemperature = 0 : -13 : 13 UNIT = "K"
  Density = 0.755 : -1e-10 : 1e5 UNIT = "kg/m3"
  DynViscosity =  0.1 : -1.0e-12 : 1.0e10 unit = "N*s/m^2"
  Energy = 0 : -1e10 : 1e10 UNIT = "J/m3"
  Flowrate = 0.001: -1.0e5 : 1.0e5 unit ="Kg/s"
  Heat = 1: -1e10 : 1e10 UNIT = "W"
  HTCoeff = 0.1 : -1.0e-12:1.0E5  unit = "W/(m^2*K)"   # heat transfer coefficient
  IConcentration = 0:-1.0e-7 : 1.0e25 unit = ""
  KinViscosity = 0.1 : -1.0e-12 : 1.0e10 unit = "m^2/s"
  Length = 0.01 : -1.0e-10:1.0e3    unit = "m"  # length
  MassFlowRate = 0 : -1.0e6 : 1.0e6 unit = "kg/s"
  MassFlux = 218.0 : 0 : 1e8 UNIT = "Kg/(m2*s)"
  MassVelocity = 1 : -1e-7 : 1e4 UNIT = "kg/m^2/s"
  Momentum = 218.0 : 0 : 1e8 UNIT = "Kg/(m2*s)"
  Nondimensional =  0.01 : -1.0e-10 : 1.0e12 unit = "dimensionless" # for the nondimensional numbers
  PhiConcentration = 0:-1.0e-7 : 1.0e25 unit = "neutrons/(m2*s)"
  Power = 0:-1.0e-7 : 1.0e10 unit = ""
  Pressure = 1.8e6 : 0   : 2e8   UNIT = "Pa"
  Reactivity = 0:-1.0e6 : 1.0e6 unit = ""
  RealVar = 0: -1.0e10 : 1.0e10 unit = ""
  SpecificEnergy = 0 : -1e10 : 1e10 UNIT = "J/Kg"
  SpHeatCapacity = 0 : -1.0e-7 : 1.0e6 unit = "J/(kg*K)"
  SpHeatCapacityV = 0: -1.0e-7 :1.0e10 unit = "J/(m3*K)"
  Temperature = 0: 0 : 1e10 UNIT = "K"
  ThConduct = 0.01 : -1.0e-10 : 1.0e5 unit = "W/(m*K)"  #thermal conductivity
  Velocity = 2 : -1.0e-7 : 1.0e5 unit = "m/s"
  XeConcentration = 0:-1.0e-7 : 1.0e25 unit = ""
END # declare
##########################################################################



##########################################################################
MODEL HTCoeffGasPipeInsAir
##########################################################################
  PARAMETER
    #SET
    dynViscHe AS REAL  # dynamic viscosity of Helium inside the pipe unit = "Pa*s" : set
    CpHe AS REAL # heat capacity unit=  "J/(K*kg)" : set
    k1 AS REAL # ThConduct  # thermal conductivity of Alloy 617 unit = "W/(m*K)" : set
    k2 AS REAL # ThConduct  # thermal conductivity of insulation  unit = "W/(m*K)" : set
    kHe AS REAL # thermal conductivity of Helium inside pipe  unit = "W/(m*K)" : set
    #EXTERNAL
    r0 AS REAL # pipe internal radius unit= "m"  : external
    r1 AS REAL # pipe internal radius + metal thickness unit = "m": external
    r2 AS REAL # pipe internal radius + metal thickness + insulations thickness unit = "m": external
  VARIABLE
    h2 AS HTCoeff  # heat transfer coefficient of Air outside pipe, I will set it as a constant for now : set
```

```
55
56      rhoPipeHe AS Density # average density of Helium inside the pipe : external
57      vPipe AS Velocity # velocity inside the pipe :  external
58
59      h0 AS HTCoeff # heat transfer coefficient of Helium inside pipe, depends on v and rho : calculated
60      U AS HTCoeff # overall heat transfer coefficient : calculated
61    SET
62      dynViscHe := 3.963e-5; # INL calculations  # 34.9e-6; #  from NIST
63      CpHe := 5.192e3; # from NIST at 400 C , 1-2 MPa
64      k1 := 26.0; # from INL calculations  #19.3; # thermal conductivity of Alloy 617 at 400C from material sheet
            information
65      k2 := 0.1; # thermal conductivity of insulation from INL OK
66      kHe := 0.31102; # from INL calculations #  0.274; # thermal conductivity Helium at 400C , 1-2 MPa
67    INTERMEDIATE
68      kinViscHe := dynViscHe/rhoPipeHe; # kinViscHe AS KinViscosity # kinematic viscosity of Helium inside the pipe
             : calculated
69      ReHe := vPipe * 2*r0/ kinViscHe; # ReHe AS Nondimensional # reynolds number : calculated
70      PrHe := dynViscHe * CpHe/kHe; # PrHe AS Nondimensional # Prandtl number : calculated
71      Nu01 := 4.364;
72      Nu03 := 0.021*ReHe^0.8*PrHe^0.4;
73      Nu03ReHe5000 := 0.021*5000^0.8*PrHe^0.4; # Nu03 at ReHe = 5000
74      Nu02 := Nu01 + (Nu03ReHe5000 - Nu01)/(5000-2300)*(ReHe -2300);  # linearized Nu0 between
75    EQUATION
76      IF ReHe <= 2300 THEN
77        h0*2*r0 = kHe*Nu01; # linear model Oh paper June 2007 ANS
78      ELSE
79        IF ReHe <=5000 THEN
80          h0*2*r0 = kHe*Nu02;
81        ELSE
82          h0*2*r0 = kHe*Nu03; #Dittus-Boelter equation [Kays and Crawford, 1980]
83        END #if
84      END # if
85      h2 = 17.344; # INL calculations  # 0.0515;#engineeringtoolbox.com
86      U = 1/(r0* ( 1/(r0*h0) + log(r1/r0)/k1 + log(r2/r1)/k2 + 1/(r2*h2)  ) ); # overall heat transfer coefficient,
             [Bird et al]
87  END # HTCoefPipeAir
88  ######################################################################
89
90
91  ######################################################################
92  MODEL HTCoeffGasMetalHX
93  ######################################################################
94    PARAMETER
95      #SET
96      dynViscHe AS REAL  # dynamic viscosity of Helium inside the pipe unit = "Pa*s" : set
97      CpHe AS REAL # heat capacity unit=  "J/(K*kg)" : set
98      k1 AS REAL # ThConduct  # thermal conductivity of Alloy 617 unit = "W/(m*K)" : set
99      Pi AS REAL
100     #EXTERNAL
101     kHe AS REAL # thermal conductivity of Helium inside pipe  unit = "W/(m*K)" : set
102     r0 AS REAL # pipe internal radius unit= "m"  : external
103     t AS REAL # plate thickness = "m": external
104     #CALCULATED
105     r0eq AS REAL # pipe internal radius unit = "m": calculated
106   VARIABLE
107     #EXTERNAL
108     rhoPipeHe AS Density # average density of Helium inside the pipe : external
109     vPipe AS Velocity # velocity inside the pipe :  external
110     #CALCULATED
111     h0 AS HTCoeff # heat transfer coefficient of Helium inside pipe, depends on v and rho : calculated
112     U AS HTCoeff # overall heat transfer coefficient : calculated
113   SET
114     #SET
115     dynViscHe := 34.9e-6; #  from NIST
116     CpHe := 5.191e3; # from NIST at 400 C , 1-2 MPa
117     k1 := 22.0; # from INL calculations  # 19.3; # thermal conductivity of Alloy 617 at 400C from material sheet
            information
118     Pi := 4*ATAN(1);
119     #CALCULATED
120     r0eq:= Pi*r0^2/(2*(Pi*r0/2 + r0));
121   INTERMEDIATE
122     kinViscHe := dynViscHe/rhoPipeHe; # kinViscHe AS KinViscosity # kinematic viscosity of Helium inside the pipe
             : calculated
123     ReHe := vPipe * 2*r0/ kinViscHe; # ReHe AS Nondimensional # reynolds number : calculated
124     PrHe := dynViscHe * CpHe/kHe; # PrHe AS Nondimensional # Prandtl number : calculated
125     Nu01 := 4.364;
126     Nu03 := 0.021*ReHe^0.8*PrHe^0.4;
127     Nu03ReHe5000 := 0.021*5000^0.8*PrHe^0.4; # Nu03 at ReHe = 5000
128     Nu02 := Nu01 + (Nu03ReHe5000 - Nu01)/(5000-2300)*(ReHe -2300);  # linearized Nu0 between
129   EQUATION
130     IF ReHe <= 1e8 THEN
131       h0*2*r0 = kHe*Nu01; # linear model Oh paper June 2007 ANS
132     ELSE
133       IF ReHe <=5000 THEN
134         h0*2*r0 = kHe*Nu02;
135       ELSE
136         h0*2*r0 = kHe*Nu03; #Dittus-Boelter equation [Kays and Crawford, 1980]
137       END #if
138     END # if
139     U = 1/ (1/h0 + (t/2)/k1);
140 END # HTCoeffGasMetalHX
141 ######################################################################
```

```
142
143
144   ######################################################################
145   MODEL HTCoeffGasMetalHXDummyInner
146   ######################################################################
147     PARAMETER
148       # SET
149       k1 AS REAL # ThConduct  # thermal conductivity of Alloy 617 unit = "W/(m*K)" : set
150       rOuter AS REAL # pipe external radius unit= "m"  : set
151       rInner AS REAL # pipe internal radius unit= "m" : set
152       hInner AS REAL # heat transfer coefficient inside the pipe= "W/(m^2*K)"
153       # CALCULATED
154       rM AS REAL # pipe intermediate radius unit = "m" : calculated
155     VARIABLE
156       #EXTERNAL
157       rhoPipeHe AS Density # average density of Helium inside the pipe : external
158       vPipe AS Velocity # velocity inside the pipe :  external
159       #CALCULATED
160       U AS HTCoeff # overall heat transfer coefficient : calculated
161     SET
162       #SET
163       k1 := 22.0; # from INL calculations  # 19.3; # thermal conductivity of Alloy 617 at 400C from material sheet
            information
164       rInner:=  0.01/2;   rOuter:= 0.013/2;   hInner := 2000;
165       # CALCULATED
166       rM:= (rInner + rOuter)/2;
167     INTERMEDIATE
168     EQUATION
169       U = 1/(1/hInner + rInner*log(rM/rInner)/k1); # this one is calculated with respect to AInner
170   END # HTCoeffGasMetalHX
171   ######################################################################
172
173   ######################################################################
174   MODEL HTCoeffGasMetalHXDummyOuter
175   ######################################################################
176     PARAMETER
177       # SET
178       k1 AS REAL # ThConduct  # thermal conductivity of Alloy 617 unit = "W/(m*K)" : set
179       rOuter AS REAL # pipe external radius unit= "m"  : set
180       rInner AS REAL # pipe internal radius unit= "m" : set
181       hOuter AS REAL # heat transfer coefficient inside the pipe= "W/(m^2*K)"
182
183       # CALCULATED
184       rM AS REAL # pipe intermediate radius unit = "m" : calculated
185     VARIABLE
186       #EXTERNAL
187       rhoPipeHe AS Density # average density of Helium inside the pipe : external
188       vPipe AS Velocity # velocity inside the pipe :  external
189       #CALCULATED
190       U AS HTCoeff # overall heat transfer coefficient : calculated
191     SET
192       #SET
193       k1 := 22.0; # from INL calculations  # 19.3; # thermal conductivity of Alloy 617 at 400C from material sheet
            information
194       rInner:=  0.01/2;   rOuter:= 0.013/2;     hOuter:= 1364; # INL report
195       # CALCULATED
196       rM:= (rInner + rOuter)/2;
197     INTERMEDIATE
198     EQUATION
199       U = 1/(rOuter*log(rOuter/rM)/k1 + 1/hOuter); # this one is calculated with respect to AOuter
200   END # HTCoeffGasMetalHX
201   ######################################################################
202
203
204   ######################################################################
205   MODEL HEEFM # heat transfer coefficient solid to helium in nuclear reactor core
206   ######################################################################
207   PARAMETER
208       # SET here
209       d AS REAL
210       # EXTERNAL
211       VoidF AS REAL # VoidFraction
212       Area AS REAL
213   VARIABLE
214       # OUTPUT
215       HTCoeffR AS HTCoeff
216       # EXTERNAL
217       Temp AS Temperature
218       W AS MassFlowRate
219   SET
220       d:= 0.06; # dPebble= 60 mm, L 533
221   INTERMEDIATE
222       Ac:= Area*VoidF;
223       HeVis:= 1e-7*3.953*Temp^0.687; # temp in K, L37
224       Renold:= W*d/(Ac*HeVis);
225       HePr:= 0.740*Temp^(-0.014);
226       Nu:=1.27*HePr^0.333*Renold^0.36/VoidF^1.18+ 0.033*HePr^0.5*Renold^0.86/VoidF^1.07;
227       KGas:= 2.774*Temp^0.701*0.001;
228   EQUATION
229       HTCoeffR=KGas*Nu/d;
230   END # HEEFM
231   ######################################################################
```

```
232
233
234  ######################################################################
235  MODEL HEEFM2 # heat transfer coefficient solid to helium in nuclear reactor core
236  ######################################################################
237  PARAMETER
238      # SET here
239      d AS REAL
240      # EXTERNAL
241      VoidF AS REAL # VoidFraction
242      Area AS REAL
243  VARIABLE
244      # OUTPUT
245      HTCoeffR AS HTCoeff
246      # EXTERNAL
247      Temp AS Temperature
248      W AS MassFlowRate
249  SET
250      d:= 0.06; # dPebble= 60 mm, L 533
251  INTERMEDIATE
252      Ac:= Area*VoidF;
253      HeVis:= 1e-7*3.953*Temp^0.687; # temp in K, L37
254      Renold:= W*d/(Ac*HeVis);
255      HePr:= 0.740*Temp^(-0.014);
256      Nu:=1.27*HePr^0.333*Renold^0.36/VoidF^1.18+ 0.033*HePr^0.5*Renold^0.86/VoidF^1.07;
257      KGas:= 2.774*Temp^0.701*0.001;
258  EQUATION
259      HTCoeffR=KGas*Nu/d;
260  END # HEEFM
261  ######################################################################
262
263
264  ######################################################################
265  MODEL CPRF # reflector heat capacity
266  ######################################################################
267  PARAMETER
268      # SET here
269      # EXTERNAL
270  VARIABLE
271      # OUTPUT
272      Cp AS SpHeatCapacity
273      # EXTERNAL
274      Temp AS Temperature
275  SET
276  INTERMEDIATE
277      TF:= (Temp - 273.16)*9/5 + 32; # transformation from K to F
278      CPi:= 0.127+0.374*(1.0-EXP(-TF/960.0));
279  EQUATION
280      CP=CPi*4.1868*1000;
281  END # CPRF
282  ######################################################################
283
284
285  ######################################################################
286  MODEL CPFuel # fuel heat capacity
287  ######################################################################
288  PARAMETER
289      # SET here
290      # EXTERNAL
291      VoidF AS REAL
292      T0 AS REAL #reference temperature, freezing point in K
293  VARIABLE
294      # OUTPUT
295      Cp AS SpHeatCapacityV
296      # EXTERNAL
297      Temp AS Temperature
298  SET
299      T0:= 273.16;
300  INTERMEDIATE
301      CPF:=1.75*(1-VoidF);
302      F1:=0.645+3.14*(Temp-T0)/1000-2.809*((Temp-T0)/1000)^2+0.959*((Temp-T0)/1000)^3;
303  EQUATION
304      CP=CPF*F1*(100^3);
305  END # CPFuel
306  ######################################################################
```

MODELS_PIPEMODELS.JAC

```
1  ######################################################################
2  MODEL LoopFullDyn
3  ######################################################################
4    PARAMETER
5      Pi AS REAL
```

```
    N_Pipes AS INTEGER # number of sections with different properties
    R_c AS REAL   #Rc constant "J/(mol*K)
    Mw AS REAL    #Molecular weight "Kg/mol"
    T_stand AS REAL   #Standard Temp "K"
    Cp AS REAL    #specific heat at constant pressure "J/Kg/K"
    h_form AS REAL    #specific enthalpy of formation "J/Kg"
    gamma AS REAL     #gamma helium "no units"
    InitT AS REAL #Temperature "K"
    InitP AS REAL     #initial pressure "Pa"
    NG AS INTEGER     #number of control volumes in the grid : external
    SectionL AS ARRAY(N_Pipes) OF REAL # length of each section "m"
    r0 AS ARRAY(N_Pipes) OF REAL # radus of each section "m"
    muVisc AS ARRAY(N_Pipes) OF REAL # average viscosity in pipe
    Tw AS REAL # In Plant Model
    InitDeltaP AS REAL # In Simulation
    MaxDeltaP AS REAL # In Simulation
    A AS ARRAY(NG) OF REAL #area of each section 'm^2'
    A_In AS REAL
    A_Out AS REAL
    SectionEnd AS ARRAY(N_Pipes) OF INTEGER # end point sections
    SectionSpan AS ARRAY(N_Pipes) OF INTEGER # length section
    SectionDelta_x AS ARRAY(N_Pipes) OF REAL # length of control volume "m"
    Delta_x AS ARRAY(NG) OF REAL # length of control volume for each point "m"
    Delta_xShift AS ARRAY(NG) OF REAL # length of control volume for each point in the shifted grid (non v) "m"
    Delta_xExtra AS REAL
    Vol AS ARRAY(NG) OF REAL
  UNIT
    HT1 AS HTCoeffGasPipeInsAir # section 1-10
    HT2 AS HTCoeffGasMetalHX # section 11-30
    HT3 AS HTCoeffGasPipeInsAir # section 31-50
    HT4 AS HTCoeffGasMetalHXDummyOuter  # section 51-70 # this is not necessary, the number is defined directly INL report p 12
  VARIABLE
    AdaptFactor1 AS Nondimensional
    AdaptFactor2 AS Nondimensional
    rho AS ARRAY(NG) OF Density
    P AS ARRAY(NG) OF Pressure
    P_Out AS Pressure
    h AS ARRAY(NG) OF SpecificEnergy
    h0 AS SpecificEnergy
    T AS ARRAY(NG) OF Temperature
    T0 AS Temperature
    TExtrnl AS ARRAY(NG) OF Temperature
    DeltaP AS Pressure
    psi AS ARRAY(NG) OF Energy # 2 to NG Differential
    F AS ARRAY(NG) OF Flowrate
    F_In AS Flowrate
    rho_avg AS Density
    totalMass AS Nondimensional
    ModifyFlow AS Nondimensional
    comp_leak AS Nondimensional
    #Extra Variables For Section Change at Compressor
    rhoExtra AS ARRAY(2) OF Density
    PExtra AS ARRAY(2) OF Pressure
    hExtra AS ARRAY(2) OF SpecificEnergy
    TExtra AS ARRAY(2) OF Temperature
    psiExtra AS ARRAY(2) OF Energy
    FExtra AS ARRAY(2) OF Flowrate
  SET
    Pi:= 4*ATAN(1);
    R_c := 8.314;       Mw := 4.0E-3;
    T_stand := 298;       Cp := 20.78;        h_form := 0;    gamma := 1.667;
    InitT:=291.15;        InitP:= 2.0e6;        NG:=100;        N_Pipes := 4;
    A((SectionEnd(1)+1):SectionEnd(2)):= 0.3475;  #= achan*nhot = PI*d^2/8*nhot # A(2)
    A((SectionEnd(3)+1):SectionEnd(4)):= 0.4012; # from filewith LMTD # A(4)
    HT2.kHe:= 0.34981; # from INL calculations, file with LMTD
    SectionEnd(1) := 10;
    SectionEnd(2) := SectionEnd(1) + 40;
    SectionEnd(3) := SectionEnd(2) + 10;
    SectionEnd(N_Pipes):= NG;
    SectionL(1) := 90;    SectionL(2) := 0.98807; SectionL(3) := 90;
    SectionL(4) :=8.656;
    r0(1) := 0.413/2;     r0(2) := (9.1652e-4)/2; # dhyd in the PCHE, this is needed to calculate Re
    r0(3) := 0.444/2;     r0(4):= 0.01123/2; # dhyd in the PHX
    A(1:SectionEnd(1)):=Pi*r0(1)^2;  # A(1) # 0.133964579
    A_In:= A(1);
    A_Out := A(NG);
    A((SectionEnd(2)+1):SectionEnd(3)):=Pi*r0(3)^2;  # A(3)  # 0.154830252
    HT1.r0 := r0(1);
    HT1.r1 := HT1.r0*(2*0.01 + 1); # r1 depends on the t/d ratio, t/d = 0.01 for cold leg with P = 2MPa, INL report p16 OK
    HT1.r2 := HT1.r1 + 0.0058; # INL report p34
    HT2.r0 := r0(2);
    HT2.t :=   (HT2.r0*2*0.78*2 - HT2.r0)/3; # according to INL, an average of 2*tmax + 1*tmin
    HT3.r0 := r0(3);
    HT3.r1 := HT3.r0*(2*0.11 + 1); # r1 depends on the t/d ratio, t/d = 0.11 for hot leg with P = 2MPa, INL report p16
    HT3.r2 := HT3.r1 + 0.0059; # INL report p34
    SectionSpan(1) := SectionEnd(1);
    FOR I:=2 TO N_Pipes DO
      SectionSpan(I) := SectionEnd(I) - SectionEnd(I-1);
    END # for
```

```
95      SectionDelta_x(1) := SectionL(1)/(SectionSpan(1));
96      SectionDelta_x(2):= SectionL(2)/SectionSpan(2);
97      SectionDelta_x(3):=SectionL(3)/SectionSpan(3);
98      SectionDelta_x(4):=SectionL(4)/SectionSpan(4);
99      Delta_x(1:SectionEnd(1)):= SectionDelta_x(1);
100     Delta_xExtra:= 0.01;
101     FOR I:=1 TO N_Pipes-1 DO
102       Delta_x((SectionEnd(I)+1):SectionEnd(I+1)):= SectionDelta_x(I+1);
103     END # for I  # 20071112 changed
104     Delta_xShift(1:SectionEnd(1)):= SectionDelta_x(1);
105     FOR I := 1 TO N_Pipes-1 DO
106       Delta_xShift(SectionEnd(I)+1):= (SectionDelta_x(I)+SectionDelta_x(I+1))/2;
107       Delta_xShift(SectionEnd(I)+2:SectionEnd(I+1)):= SectionDelta_x(I+1);
108     END # for
109     Vol(1) := A(1)*SectionDelta_x(1);
110     Vol(2:SectionEnd(1)):= A(2:SectionEnd(1))*SectionDelta_x(1);
111     FOR I:= 1 TO (N_Pipes-1) DO
112       Vol(SectionEnd(I)+1:SectionEnd(I+1)):= A(SectionEnd(I)+1:SectionEnd(I+1))*SectionDelta_x(I+1);
113     END # for
114   INTERMEDIATE
115     Dimension i_U(NG), i_r0(NG), fr(NG)
116     Dimension i_mdot(NG) # mdot at each grid point, all the values should be equal
117     DIMENSION intRe(NG), intfArray(NG),intEps(N_Pipes)
118     i_U(1:SectionEnd(1)):=HT1.U ;
119     i_U((SectionEnd(1)+1):SectionEnd(2)):=HT2.U;
120     i_U((SectionEnd(2)+1):SectionEnd(3)):=HT3.U;
121     i_U((SectionEnd(3)+1):SectionEnd(4)):=HT4.U;
122     i_r0(1:SectionEnd(1)):=r0(1) ;
123     i_r0((SectionEnd(1)+1):SectionEnd(2)):=r0(2);
124     i_r0((SectionEnd(2)+1):SectionEnd(3)):=r0(3);
125     i_r0((SectionEnd(3)+1):SectionEnd(4)):=r0(4);
126     i_rhoPipeHe1:= sigma(rho(1:SectionEnd(1)))/SectionSpan(1);
127     i_vPipe1:=sigma(F(1:SectionEnd(1))/A(1:SectionEnd(1))/rho(1:SectionEnd(1)))/SectionSpan(1);
128     i_rhoPipeHe2:= sigma(rho(SectionEnd(1)+1:SectionEnd(2)))/SectionSpan(2);
129     i_vPipe2:=sigma(F(SectionEnd(1)+1:SectionEnd(2))/A(SectionEnd(1)+1:SectionEnd(2))/rho(SectionEnd(1)+1:
            SectionEnd(2)))/SectionSpan(2);
130     i_rhoPipeHe3:= sigma(rho(SectionEnd(2)+1:SectionEnd(3)))/SectionSpan(3);
131     i_vPipe3:=sigma(F(SectionEnd(2)+1:SectionEnd(3))/A(SectionEnd(2)+1:SectionEnd(3))/rho(SectionEnd(2)+1:
            SectionEnd(3)))/SectionSpan(3);
132     i_rhoPipeHe4:= sigma(rho(SectionEnd(3)+1:SectionEnd(4)))/SectionSpan(4); #20080311
133     i_vPipe4:= sigma(F(SectionEnd(3)+1:SectionEnd(4))/A(SectionEnd(3)+1:SectionEnd(4))/rho(SectionEnd(3)+1:
            SectionEnd(4)))/SectionSpan(4);
134     intEps(1):=4.57e-5;
135     intEps(2):=r0(2)*1e-9;
136     intEps(3):=4.57e-5;
137     intEps(4):=r0(4)*1e-9;
138     intRe(1):= (F_In/A_In + F(1)/A(1))/2*r0(1)*2/muVisc(1);
139     intRe(2:SectionEnd(1)):= (F(1:SectionEnd(1)-1)/A(1:SectionEnd(1)-1)+F(2:SectionEnd(1))/A(2:SectionEnd(1)))/2*
            r0(1)*2/muVisc(1);
140     intRe(SectionEnd(1)+1):= (F(SectionEnd(1))/A(SectionEnd(1))+F(SectionEnd(1)+1)/A(SectionEnd(1)+1))/2*r0(2)*2/
            muVisc(2);
141     intRe(SectionEnd(1)+2:SectionEnd(2)):=(F(SectionEnd(1)+1:SectionEnd(2)-1)/A(SectionEnd(1)+
142       1:SectionEnd(2)-1)+F(SectionEnd(1)+2:SectionEnd(2))/A(SectionEnd(1)+2:SectionEnd(2)))/2*r0(2)*2/muVisc(2);
143     intRe(SectionEnd(2)+1):= (F(SectionEnd(2))/A(SectionEnd(2))+F(SectionEnd(2)+1)/A(SectionEnd(2)+1))/2*r0(3)*2/
            muVisc(3);
144     intRe(SectionEnd(2)+2:SectionEnd(3)):=(F(SectionEnd(2)+1:SectionEnd(3)-1)/A(SectionEnd(2)+
145       1:SectionEnd(3)-1)+F(SectionEnd(2)+2:SectionEnd(3))/A(SectionEnd(2)+2:SectionEnd(3)))/2*r0(3)*2/muVisc(3);
146     intRe(SectionEnd(3)+1):= (F(SectionEnd(3))/A(SectionEnd(3))+F(SectionEnd(3)+1)/A(SectionEnd(3)+1))/2*r0(4)*2/
            muVisc(4);
147     intRe(SectionEnd(3)+2:SectionEnd(4)):=(F(SectionEnd(3)+1:SectionEnd(4)-1)/A(SectionEnd(3)+
148       1:SectionEnd(4)-1)+F(SectionEnd(3)+2:SectionEnd(4))/A(SectionEnd(3)+2:SectionEnd(4)))/2*r0(4)*2/muVisc(4);
149     intfArray(1:SectionEnd(1)):=1/(-2*LOG10(intEps(1)/(3.7*r0(1)) + 2.51/intRe(1:SectionEnd(1))
150       *(1.14 -2*LOG10(intEps(1)/r0(1) + 21.25/(intRe(1:SectionEnd(1))^0.9)))))^2/4;
151     intfArray(SectionEnd(1)+1:SectionEnd(2)):=64/intRe(SectionEnd(1)+1:SectionEnd(2))/4;
152     intfArray(SectionEnd(2)+1:SectionEnd(3)):=1/(-2*LOG10(intEps(3)/(3.7*r0(3))
153       + 2.51/intRe(SectionEnd(2)+1:SectionEnd(3))
154       *(1.14 -2*LOG10(intEps(3)/r0(3) + 21.25/(intRe(SectionEnd(2)+1:SectionEnd(3))^0.9)))))^2/4;
155     intfArray(SectionEnd(3)+1:SectionEnd(4)):=1/(-2*LOG10(intEps(4)/(3.7*r0(4))
156       + 2.51/intRe(SectionEnd(3)+1:SectionEnd(4))
157       *(1.14 -2*LOG10(intEps(4)/r0(4) + 21.25/(intRe(SectionEnd(3)+1:SectionEnd(4))^0.9)))))^2/4;
158     fr(1:SectionEnd(1))                 := intfArray(1:SectionEnd(1));#0.01294/4; # OK
159     fr((SectionEnd(1)+1):SectionEnd(2))   := intfArray(SectionEnd(1)+1:SectionEnd(2));#4.009E-02/4; # OK
160     fr((SectionEnd(2)+1):SectionEnd(3))   := intfArray(SectionEnd(2)+1:SectionEnd(3));#0.01348/4; # OK
161     fr((SectionEnd(3)+1):SectionEnd(4))   := intfArray(SectionEnd(3)+1:SectionEnd(4));#2.758E-02/4; # OK
162   EQUATION
163     # Mass Balance
164     Delta_xExtra*A(1)*$rhoExtra(1) = (F_In - FExtra(1));
165     Delta_xExtra*A(1)*$rhoExtra(2)  = (FExtra(1) - FExtra(2));
166     Delta_x(1)*A(1)*$rho(1) = (FExtra(2) - F(1));
167     FOR I:= 2 TO NG DO
168       Delta_x(I)*A(I)*$rho(I) = (F(I-1) - F(I));
169     END # for
170     # Momentum Balance
171     (Delta_xExtra + Delta_xExtra/2)*$FExtra(1) =
172       -(PExtra(2)*A(NG) - PExtra(1)*A(NG))
173       +(   F_In                    )^2/     (rhoExtra(1)*A(NG))
174       -(   (  FExtra(1) + FExtra(2)  )/2   )^2   /   (rhoExtra(2)*A(NG));
175     (Delta_xExtra/2 + Delta_x(1)/2)*$FExtra(2) =
176       -(    P(1)*A(1)          -          ( PExtra(2)*Delta_x(1)/2 + P(1)*Delta_xExtra/2 ) / ( Delta_x(1)/2 +
              Delta_xExtra/2)  * ( A(1) - A(NG) ) -  PExtra(2)*A(NG)  )
177       +(   (     FExtra(1) + FExtra(2)   )/2   )^2  /   (  rhoExtra(2)*A(NG)  )
178       -(   (     F(1)    + FExtra(2)      )/2    )^2  /   (  rho(1)*A(1)  )
```

```
      - 2*fr(1)/( 2*i_r0(1) )    *   (  ABS(FExtra(2))*FExtra(2)   )/  (  rho(1)*A(1)  )  *  Delta_x(1)/2;
    (Delta_x(1)/2 + Delta_x(2)/2) *$F(1)  =
      -(    P(2)*A(2)     -         ( P(1)*Delta_x(2)/2 + P(2)*Delta_x(1)/2 )  /  ( Delta_x(1)/2+Delta_x(2)/2 )
           *  ( A(2) - A(1) )      -     P(1)*A(1)    )
      +(   (    FExtra(2)  +  F(1)    )/2   )^2     /    (  rho(1)*A(1)  )
      -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A(2)   )
      - 2*fr(1)/( 2*i_r0(1) )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A(1)  )  *  Delta_x(1)/2
      - 2*fr(2)/( 2*i_r0(2) )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A(2)  )  *  Delta_x(2)/2;
    FOR I:=2 TO SectionEnd(4)-1 DO
      (  Delta_x(I)/2 + Delta_x(I+1)/2  )    * $F(I) =
        -(    P(I+1)*A(I+1)     -         ( P(I)*Delta_x(I+1)/2 + P(I+1)*Delta_x(I)/2 )  /  ( Delta_x(I)/2+
             Delta_x(I+1)/2 )  *  ( A(I +1) - A(I) )      -     P(I)*A(I)    )
        +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A(I)  )
        -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A(I+1)   )
        - 2*fr(I)/( 2*i_r0(I) )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A(I)  )  *  Delta_x(I)/2
        - 2*fr(I+1)/( 2*i_r0(I+1) )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A(I+1)  )  *  Delta_x(I+1)/2;
    END
    Delta_x(NG)/2    * $F(NG) =
        -(    P_Out*A_Out    -     P(NG)*A(NG)    )
        +(   (    F(NG-1)  +  F(NG)    )/2   )^2     /    (  rho(NG)*A(NG)  )
        -(   F(NG)    )^2    /     (   rho(NG)*A(NG)   )
        - 2*fr(NG)/( 2*i_r0(NG) )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A(NG)  )  *  Delta_x(NG)/2;

    # Energy Balance
      Delta_xExtra*A(NG)*$psiExtra(1) = (F_In*h0 - FExtra(1)*hExtra(1));
      Delta_xExtra*A(NG)*$psiExtra(2) = (FExtra(1)*hExtra(1) - FExtra(2)*hExtra(2));
      Vol(1)*$psi(1) = (FExtra(2)*hExtra(2) - F(1)*h(1)
          +4*i_U(1)/(2*i_r0(1))*(TExtrnl(1)-T(1))*Vol(1));
      FOR I:=2 TO NG DO
        Vol(I)*$psi(I) = (F(I-1)*h(I-1) - F(I)*h(I)
          +4*i_U(I)/(2*i_r0(I))*(TExtrnl(I)-T(I))*Vol(I));
      END # for

    # Algebraic Equations
        0 = (psiExtra(1)  - (rhoExtra(1)*hExtra(1) - PExtra(1)));
        0 = (psiExtra(2)  - (rhoExtra(2)*hExtra(2) - PExtra(2)));
    FOR I:=1 TO NG DO
        0 = (psi(I)  - (rho(I)*h(I) - P(I)));
    END # for

    # 2- Thermodynamical Part
    0 = 1e-7*(hExtra(1) - (h_form + Cp/Mw*(TExtra(1)-T_stand)));
    0 = 1e-7*(hExtra(2) - (h_form + Cp/Mw*(TExtra(2)-T_stand)));
    FOR I:= 1 TO NG DO
       0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
    END # for
    0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));

    # Equation of State
    0= (PExtra(1)*Mw - rhoExtra(1)*R_c*TExtra(1));
    0= (PExtra(2)*Mw - rhoExtra(2)*R_c*TExtra(2));
    FOR I:=1 TO NG DO
        0= (P(I)*Mw - rho(I)*R_c*T(I));
    END # for

    # 3 - Forced Continuity
    # Equation to set the amount of Gas <=> average density
    rho_avg = (SIGMA(rho*Vol)+ (rhoExtra(1) + rhoExtra(2))*Delta_xExtra*A(NG)) /(SIGMA(Vol) + 2*Delta_xExtra*A(NG
          )) ;
    totalMass = SIGMA(rho*Vol) + (rhoExtra(1) + rhoExtra(2))*Delta_xExtra*A(NG);
    F_In = F(NG)*(1-comp_leak) + ModifyFlow;

    # Compressor Equations
    # deltaP
    0 = (DeltaP  - (PExtra(1) - P_Out));

    # energy 1
    0 = P_Out^(gamma-1)*T0^gamma - PExtra(1)^(gamma-1)*T(NG)^gamma;

    #HT1 equations
    HT1.rhoPipeHe = i_rhoPipeHe1;
    HT1.vPipe = i_vPipe1;
    HT2.rhoPipeHe = i_rhoPipeHe2;
    HT2.vPipe = i_vPipe2;
    HT3.rhoPipeHe =i_rhoPipeHe3;
    HT3.vPipe = i_vPipe3;
    HT4.rhoPipeHe= i_rhoPipeHe4;
    HT4.vPipe = i_vPipe4;
END
######################################################################


######################################################################
MODEL Loop0Mach
######################################################################
  PARAMETER
    Pi AS REAL
    N_Pipes AS INTEGER # number of sections with different properties
    R_c AS REAL       #Rc constant "J/(mol*K)
    Mw AS REAL        #Molecular weight "Kg/mol"
    T_stand AS REAL   #Standard Temp "K"
    Cp AS REAL        #specific heat at constant pressure "J/Kg/K"
```

```
268       h_form AS REAL    #specific enthalpy of formation "J/Kg"
269       gamma AS REAL     #gamma helium "no units"
270       InitT AS REAL     #Temperature "K"
271       InitP AS REAL     #initial pressure "Pa"
272       NG AS INTEGER     #number of control volumes in the grid : external
273       SectionL AS ARRAY(N_Pipes) OF REAL # length of each section "m"
274       r0 AS ARRAY(N_Pipes) OF REAL # radus of each section "m"
275       muVisc AS ARRAY(N_Pipes) OF REAL # average viscosity in pipe
276       Tw AS REAL # In Plant Model
277       InitDeltaP AS REAL # In Simulation
278       MaxDeltaP AS REAL # In Simulation
279       A AS ARRAY(NG) OF REAL #area of each section 'm^2'
280       A_In AS REAL
281       A_Out AS REAL
282       SectionEnd AS ARRAY(N_Pipes) OF INTEGER # end point sections
283       SectionSpan AS ARRAY(N_Pipes) OF INTEGER # length section
284       SectionDelta_x AS ARRAY(N_Pipes) OF REAL # length of control volume "m"
285       Delta_x AS ARRAY(NG) OF REAL # length of control volume for each point "m"
286       Delta_xShift AS ARRAY(NG) OF REAL # length of control volume for each point in the shifted grid (non v) "m"
287       Delta_xExtra AS REAL
288       Vol AS ARRAY(NG) OF REAL
289     UNIT
290       HT1 AS HTCoeffGasPipeInsAir # section 1-10
291       HT2 AS HTCoeffGasMetalHX # section 11-30
292       HT3 AS HTCoeffGasPipeInsAir # section 31-50
293       HT4 AS HTCoeffGasMetalHXDummyOuter  # section 51-70 # this is not necessary, the number is defined directly
            INL report p 12
294     VARIABLE
295       AdaptFactor1 AS Nondimensional
296       AdaptFactor2 AS Nondimensional
297       rho AS ARRAY(NG) OF Density
298       P AS ARRAY(NG) OF Pressure
299       P_Out AS Pressure
300       h AS ARRAY(NG) OF SpecificEnergy
301       h0 AS SpecificEnergy
302       T AS ARRAY(NG) OF Temperature
303       T0 AS Temperature
304       TExtrnl AS ARRAY(NG) OF Temperature
305       DeltaP AS Pressure
306       psi AS ARRAY(NG) OF Energy # 2 to NG Differential
307       F AS ARRAY(NG) OF Flowrate
308       F_In AS Flowrate
309       rho_avg AS Density
310       totalMass AS Nondimensional
311       ModifyFlow AS Nondimensional
312       comp_leak AS Nondimensional
313       #Extra Variables For Section Change at Compressor
314       rhoExtra AS ARRAY(2) OF Density
315       PExtra AS ARRAY(2) OF Pressure
316       hExtra AS ARRAY(2) OF SpecificEnergy
317       TExtra AS ARRAY(2) OF Temperature
318       psiExtra AS ARRAY(2) OF Energy
319       FExtra AS ARRAY(2) OF Flowrate
320     SET
321       Pi:= 4*ATAN(1);
322       R_c := 8.314;       Mw := 4.0E-3;
323       T_stand := 298;       Cp := 20.78;        h_form := 0;    gamma := 1.667;
324       InitT:=291.15;        InitP:= 2.0e6;        NG:=100;        N_Pipes := 4;
325       A((SectionEnd(1)+1):SectionEnd(2)):= 0.3475;  #= achan*nhot = PI*d^2/8*nhot # A(2)
326       A((SectionEnd(3)+1):SectionEnd(4)):= 0.4012; # from filewith LMTD # A(4)
327       HT2.kHe:= 0.34981; # from INL calculations, file with LMTD
328       SectionEnd(1) := 10;
329       SectionEnd(2) := SectionEnd(1) + 40;
330       SectionEnd(3) := SectionEnd(2) + 10;
331       SectionEnd(N_Pipes):= NG;
332       SectionL(1) := 90;    SectionL(2) := 0.98807; SectionL(3) := 90;
333       SectionL(4) :=8.656;
334       r0(1) := 0.413/2;     r0(2) := (9.1652e-4)/2; # dhyd in the PCHE, this is needed to calculate Re
335       r0(3) := 0.444/2;     r0(4):= 0.01123/2; # dhyd in the PHX
336       A(1:SectionEnd(1)):=Pi*r0(1)^2;  # A(1) # 0.133964579
337       A_In:= A(1);
338       A_Out := A(NG);
339       A((SectionEnd(2)+1):SectionEnd(3)):=Pi*r0(3)^2;  # A(3)  # 0.154830252
340       HT1.r0 := r0(1);
341       HT1.r1 := HT1.r0*(2*0.01 + 1); # r1 depends on the t/d ratio, t/d = 0.01 for cold leg with P = 2MPa, INL
            report p16 OK
342       HT1.r2 := HT1.r1 + 0.0058; # INL report p34
343       HT2.r0 := r0(2);
344       HT2.t :=   (HT2.r0*2*0.78*2 - HT2.r0)/3; # according to INL, an average of 2*tmax + 1*tmin
345       HT3.r0 := r0(3);
346       HT3.r1 := HT3.r0*(2*0.11 + 1); # r1 depends on the t/d ratio, t/d = 0.11 for hot leg with P = 2MPa, INL
            report p16
347       HT3.r2 := HT3.r1 + 0.0059; # INL report p34
348       SectionSpan(1) := SectionEnd(1);
349       FOR I:=2 TO N_Pipes DO
350         SectionSpan(I) := SectionEnd(I) - SectionEnd(I-1);
351       END # for
352       SectionDelta_x(1) := SectionL(1)/(SectionSpan(1));
353       SectionDelta_x(2):= SectionL(2)/SectionSpan(2);
354       SectionDelta_x(3):=SectionL(3)/SectionSpan(3);
355       SectionDelta_x(4):=SectionL(4)/SectionSpan(4);
356       Delta_xExtra:= 0.01;
```

```
    Delta_x(1:SectionEnd(1)):= SectionDelta_x(1);
    FOR I:=1 TO N_Pipes-1 DO
      Delta_x((SectionEnd(I)+1):SectionEnd(I+1)):= SectionDelta_x(I+1);
    END # for I  # 20071112 changed
    Delta_xShift(1:SectionEnd(1)):= SectionDelta_x(1);
    FOR I := 1 TO N_Pipes-1 DO
      Delta_xShift(SectionEnd(I)+1):= (SectionDelta_x(I)+SectionDelta_x(I+1))/2;
      Delta_xShift(SectionEnd(I)+2:SectionEnd(I+1)):= SectionDelta_x(I+1);
    END # for
    Vol(1) := A(1)*SectionDelta_x(1);
    Vol(2:SectionEnd(1)):= A(2:SectionEnd(1))*SectionDelta_x(1);
    FOR I:= 1 TO (N_Pipes-1) DO
      Vol(SectionEnd(I)+1:SectionEnd(I+1)):= A(SectionEnd(I)+1:SectionEnd(I+1))*SectionDelta_x(I+1);
    END # for
  INTERMEDIATE
    Dimension i_U(NG), i_r0(NG), fr(NG)
    Dimension i_mdot(NG) # mdot at each grid point, all the values should be equal
    DIMENSION intRe(NG), intfArray(NG),intEps(N_Pipes)
    i_U(1:SectionEnd(1)):=HT1.U ;
    i_U((SectionEnd(1)+1):SectionEnd(2)):=HT2.U;
    i_U((SectionEnd(2)+1):SectionEnd(3)):=HT3.U;
    i_U((SectionEnd(3)+1):SectionEnd(4)):=HT4.U;
    i_r0(1:SectionEnd(1)):=r0(1) ;
    i_r0((SectionEnd(1)+1):SectionEnd(2)):=r0(2);
    i_r0((SectionEnd(2)+1):SectionEnd(3)):=r0(3);
    i_r0((SectionEnd(3)+1):SectionEnd(4)):=r0(4);
    i_rhoPipeHe1:= sigma(rho(1:SectionEnd(1)))/SectionSpan(1);
    i_vPipe1:=sigma(F(1:SectionEnd(1))/A(1:SectionEnd(1))/rho(1:SectionEnd(1)))/SectionSpan(1);
    i_rhoPipeHe2:= sigma(rho(SectionEnd(1)+1:SectionEnd(2)))/SectionSpan(2);
    i_vPipe2:=sigma(F(SectionEnd(1)+1:SectionEnd(2))/A(SectionEnd(1)+1:SectionEnd(2))/rho(SectionEnd(1)+1:
        SectionEnd(2)))/SectionSpan(2);
    i_rhoPipeHe3:= sigma(rho(SectionEnd(2)+1:SectionEnd(3)))/SectionSpan(3);
    i_vPipe3:=sigma(F(SectionEnd(2)+1:SectionEnd(3))/A(SectionEnd(2)+1:SectionEnd(3))/rho(SectionEnd(2)+1:
        SectionEnd(3)))/SectionSpan(3);
    i_rhoPipeHe4:= sigma(rho(SectionEnd(3)+1:SectionEnd(4)))/SectionSpan(4); #20080311
    i_vPipe4:= sigma(F(SectionEnd(3)+1:SectionEnd(4))/A(SectionEnd(3)+1:SectionEnd(4))/rho(SectionEnd(3)+1:
        SectionEnd(4)))/SectionSpan(4);
    intEps(1):=4.57e-5;
    intEps(2):=r0(2)*1e-9;
    intEps(3):=4.57e-5;
    intEps(4):=r0(4)*1e-9;
    intRe(1):= (F_In/A_In + F(1)/A(1))/2*r0(1)*2/muVisc(1);
    intRe(2:SectionEnd(1)):= (F(1:SectionEnd(1)-1)/A(1:SectionEnd(1)-1)+F(2:SectionEnd(1))/A(2:SectionEnd(1)))/2*
        r0(1)*2/muVisc(1);
    intRe(SectionEnd(1)+1):= (F(SectionEnd(1))/A(SectionEnd(1))+F(SectionEnd(1)+1)/A(SectionEnd(1)+1))/2*r0(2)*2/
        muVisc(2);
    intRe(SectionEnd(1)+2:SectionEnd(2)):=(F(SectionEnd(1)+1:SectionEnd(2)-1)/A(SectionEnd(1)+
      1:SectionEnd(2)-1)+F(SectionEnd(1)+2:SectionEnd(2))/A(SectionEnd(1)+2:SectionEnd(2)))/2*r0(2)*2/muVisc(2);
    intRe(SectionEnd(2)+1):= (F(SectionEnd(2))/A(SectionEnd(2))+F(SectionEnd(2)+1)/A(SectionEnd(2)+1))/2*r0(3)*2/
        muVisc(3);
    intRe(SectionEnd(2)+2:SectionEnd(3)):=(F(SectionEnd(2)+1:SectionEnd(3)-1)/A(SectionEnd(2)+
      1:SectionEnd(3)-1)+F(SectionEnd(2)+2:SectionEnd(3))/A(SectionEnd(2)+2:SectionEnd(3)))/2*r0(3)*2/muVisc(3);
    intRe(SectionEnd(3)+1):= (F(SectionEnd(3))/A(SectionEnd(3))+F(SectionEnd(3)+1)/A(SectionEnd(3)+1))/2*r0(4)*2/
        muVisc(4);
    intRe(SectionEnd(3)+2:SectionEnd(4)):=(F(SectionEnd(3)+1:SectionEnd(4)-1)/A(SectionEnd(3)+
      1:SectionEnd(4)-1)+F(SectionEnd(3)+2:SectionEnd(4))/A(SectionEnd(3)+2:SectionEnd(4)))/2*r0(4)*2/muVisc(4);
    intfArray(1:SectionEnd(1)):=1/(-2*LOG10(intEps(1)/(3.7*r0(1)) + 2.51/intRe(1:SectionEnd(1))
      *(1.14 -2*LOG10(intEps(1)/r0(1) + 21.25/(intRe(1:SectionEnd(1))^0.9)))))^2/4;
    intfArray(SectionEnd(1)+1:SectionEnd(2)):=64/intRe(SectionEnd(1)+1:SectionEnd(2))/4;
    intfArray(SectionEnd(2)+1:SectionEnd(3)):=1/(-2*LOG10(intEps(3)/(3.7*r0(3))
      + 2.51/intRe(SectionEnd(2)+1:SectionEnd(3))
      *(1.14 -2*LOG10(intEps(3)/r0(3) + 21.25/(intRe(SectionEnd(2)+1:SectionEnd(3))^0.9)))))^2/4;
    intfArray(SectionEnd(3)+1:SectionEnd(4)):=1/(-2*LOG10(intEps(4)/(3.7*r0(4))
      + 2.51/intRe(SectionEnd(3)+1:SectionEnd(4))
      *(1.14 -2*LOG10(intEps(4)/r0(4) + 21.25/(intRe(SectionEnd(3)+1:SectionEnd(4))^0.9)))))^2/4;
    fr(1:SectionEnd(1))                 := intfArray(1:SectionEnd(1));#0.01294/4; # OK
    fr((SectionEnd(1)+1):SectionEnd(2))   := intfArray(SectionEnd(1)+1:SectionEnd(2));#4.009E-02/4; # OK
    fr((SectionEnd(2)+1):SectionEnd(3))   := intfArray(SectionEnd(2)+1:SectionEnd(3));#0.01348/4; # OK
    fr((SectionEnd(3)+1):SectionEnd(4))   := intfArray(SectionEnd(3)+1:SectionEnd(4));#2.758E-02/4; # OK
  EQUATION
    # Mass Balance
    Delta_xExtra*A(1)*$rhoExtra(1) = (F_In - FExtra(1));
    Delta_xExtra*A(1)*$rhoExtra(2)  = (FExtra(1) - FExtra(2));
    Delta_x(1)*A(1)*$rho(1) = (FExtra(2) - F(1));
    FOR I:= 2 TO NG DO
      Delta_x(I)*A(I)*$rho(I) = (F(I-1) - F(I));
    END # for
    # Momentum Balance
    0 =
      -(PExtra(2)*A(NG) - PExtra(1)*A(NG))
      +(   F_In                    )^2/     (rhoExtra(1)*A(NG))
      -(   (  FExtra(1) + FExtra(2)  )/2   )^2   /   (rhoExtra(2)*A(NG));
    0 =
      -(    P(1)*A(1)          -          ( PExtra(2)*Delta_x(1)/2 + P(1)*Delta_xExtra/2 ) / ( Delta_x(1)/2 +
          Delta_xExtra/2)  * ( A(1) - A(NG) ) -  PExtra(2)*A(NG)  )
      +(   (    FExtra(1) + FExtra(2)   )/2   )^2  /   (  rhoExtra(2)*A(NG)  )
      -(   (     F(1)    + FExtra(2)      )/2    )^2  /   (  rho(1)*A(1)  )
      - 2*fr(1)/( 2*i_r0(1) )    *   (  ABS(FExtra(2))*FExtra(2)   )/  (  rho(1)*A(1)  )  *  Delta_x(1)/2;
    0  =
      -(    P(2)*A(2)     -         ( P(1)*Delta_x(2)/2 + P(2)*Delta_x(1)/2 )  /  ( Delta_x(1)/2+Delta_x(2)/2 )
          *  ( A(2) - A(1) )      -     P(1)*A(1)    )
      +(   (    FExtra(2)  +  F(1)    )/2   )^2     /    (  rho(1)*A(1)  )
```

```
440         -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A(2)   )
441         - 2*fr(1)/( 2*i_r0(1) )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A(1)  )  *  Delta_x(1)/2
442         - 2*fr(2)/( 2*i_r0(2) )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A(2)  )  *  Delta_x(2)/2;
443       FOR I:=2 TO SectionEnd(4)-1 DO
444         0 =
445           -(    P(I+1)*A(I+1)     -         ( P(I)*Delta_x(I+1)/2 + P(I+1)*Delta_x(I)/2 )  /  ( Delta_x(I)/2+
                 Delta_x(I+1)/2 )  *  ( A(I +1) - A(I) )      -     P(I)*A(I)    )
446           +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A(I)  )
447           -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A(I+1)   )
448           - 2*fr(I)/( 2*i_r0(I) )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A(I)  )  *  Delta_x(I)/2
449           - 2*fr(I+1)/( 2*i_r0(I+1) )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A(I+1)  )  *  Delta_x(I+1)/2;
450       END
451       0 =
452         -(    P_Out*A_Out    -     P(NG)*A(NG)    )
453         +(   (    F(NG-1)  +  F(NG)    )/2   )^2    /    (  rho(NG)*A(NG)  )
454         -(   F(NG)    )^2    /     (   rho(NG)*A(NG)   )
455         - 2*fr(NG)/( 2*i_r0(NG) )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A(NG)  )  *  Delta_x(NG)/2;
456
457       # Energy Balance
458         Delta_xExtra*A(NG)*$psiExtra(1) = (F_In*h0 - FExtra(1)*hExtra(1));
459         Delta_xExtra*A(NG)*$psiExtra(2) = (FExtra(1)*hExtra(1) - FExtra(2)*hExtra(2));
460         Vol(1)*$psi(1) = (FExtra(2)*hExtra(2) - F(1)*h(1)
461             +4*i_U(1)/(2*i_r0(1))*(TExtrnl(1)-T(1))*Vol(1));
462         FOR I:=2 TO NG DO
463           Vol(I)*$psi(I) = (F(I-1)*h(I-1) - F(I)*h(I)
464             +4*i_U(I)/(2*i_r0(I))*(TExtrnl(I)-T(I))*Vol(I));
465         END # for
466
467       # Algebraic Equations
468         0 = (psiExtra(1)  - (rhoExtra(1)*hExtra(1) - PExtra(1)));
469         0 = (psiExtra(2)  - (rhoExtra(2)*hExtra(2) - PExtra(2)));
470       FOR I:=1 TO NG DO
471           0 = (psi(I)  - (rho(I)*h(I) - P(I)));
472       END # for
473
474       # 2- Thermodynamical Part
475       0 = 1e-7*(hExtra(1) - (h_form + Cp/Mw*(TExtra(1)-T_stand)));
476       0 = 1e-7*(hExtra(2) - (h_form + Cp/Mw*(TExtra(2)-T_stand)));
477       FOR I:= 1 TO NG DO
478          0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
479       END # for
480       0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));
481
482       # Equation of State
483       0= (PExtra(1)*Mw - rhoExtra(1)*R_c*TExtra(1));
484       0= (PExtra(2)*Mw - rhoExtra(2)*R_c*TExtra(2));
485       FOR I:=1 TO NG DO
486           0= (P(I)*Mw - rho(I)*R_c*T(I));
487       END # for
488
489       # 3 - Forced Continuity
490       # Equation to set the amount of Gas <=> average density
491       rho_avg = (SIGMA(rho*Vol)+ (rhoExtra(1) + rhoExtra(2))*Delta_xExtra*A(NG)) /(SIGMA(Vol) + 2*Delta_xExtra*A(NG
             )) ;
492       totalMass = SIGMA(rho*Vol) + (rhoExtra(1) + rhoExtra(2))*Delta_xExtra*A(NG);
493       F_In = F(NG)*(1-comp_leak) + ModifyFlow;
494
495       # Compressor Equations
496       # deltaP
497       0 = (DeltaP  - (PExtra(1) - P_Out));
498
499       # energy 1
500       0 = P_Out^(gamma-1)*T0^gamma - PExtra(1)^(gamma-1)*T(NG)^gamma;
501
502       #HT1 equations
503       HT1.rhoPipeHe = i_rhoPipeHe1;
504       HT1.vPipe = i_vPipe1;
505       HT2.rhoPipeHe = i_rhoPipeHe2;
506       HT2.vPipe = i_vPipe2;
507       HT3.rhoPipeHe =i_rhoPipeHe3;
508       HT3.vPipe = i_vPipe3;
509       HT4.rhoPipeHe= i_rhoPipeHe4;
510       HT4.vPipe = i_vPipe4;
511   END
512   ######################################################################
513
514
515   ######################################################################
516   MODEL LoopQSS
517   ######################################################################
518     PARAMETER
519       Pi AS REAL
520       N_Pipes AS INTEGER # number of sections with different properties
521       R_c AS REAL        #Rc constant "J/(mol*K)
522       Mw AS REAL         #Molecular weight "Kg/mol"
523       T_stand AS REAL    #Standard Temp "K"
524       Cp AS REAL         #specific heat at constant pressure "J/Kg/K"
525       h_form AS REAL     #specific enthalpy of formation "J/Kg"
526       gamma AS REAL      #gamma helium "no units"
527       InitT AS REAL      #Temperature "K"
528       InitP AS REAL      #initial pressure "Pa"
529       NG AS INTEGER      #number of control volumes in the grid : external
```

```
530       SectionL AS ARRAY(N_Pipes) OF REAL # length of each section "m"
531       r0 AS ARRAY(N_Pipes) OF REAL # radus of each section "m"
532       muVisc AS ARRAY(N_Pipes) OF REAL # average viscosity in pipe
533       Tw AS REAL # In Plant Model
534       InitDeltaP AS REAL # In Simulation
535       MaxDeltaP AS REAL # In Simulation
536       A AS ARRAY(NG) OF REAL #area of each section 'm^2'
537       A_In AS REAL
538       A_Out AS REAL
539       SectionEnd AS ARRAY(N_Pipes) OF INTEGER # end point sections
540       SectionSpan AS ARRAY(N_Pipes) OF INTEGER # length section
541       SectionDelta_x AS ARRAY(N_Pipes) OF REAL # length of control volume "m"
542       Delta_x AS ARRAY(NG) OF REAL # length of control volume for each point "m"
543       Delta_xShift AS ARRAY(NG) OF REAL # length of control volume for each point in the shifted grid (non v) "m"
544       Delta_xExtra AS REAL
545       Vol AS ARRAY(NG) OF REAL
546     UNIT
547       HT1 AS HTCoeffGasPipeInsAir # section 1-10
548       HT2 AS HTCoeffGasMetalHX # section 11-30
549       HT3 AS HTCoeffGasPipeInsAir # section 31-50
550       HT4 AS HTCoeffGasMetalHXDummyOuter  # section 51-70 # this is not necessary, the number is defined directly
            INL report p 12
551     VARIABLE
552       AdaptFactor1 AS Nondimensional
553       AdaptFactor2 AS Nondimensional
554       rho AS ARRAY(NG) OF Density
555       P AS ARRAY(NG) OF Pressure
556       P_Out AS Pressure
557       h AS ARRAY(NG) OF SpecificEnergy
558       h0 AS SpecificEnergy
559       T AS ARRAY(NG) OF Temperature
560       T0 AS Temperature
561       TExtrnl AS ARRAY(NG) OF Temperature
562       DeltaP AS Pressure
563       psi AS ARRAY(NG) OF Energy # 2 to NG Differential
564       F AS ARRAY(NG) OF Flowrate
565       F_In AS Flowrate
566       rho_avg AS Density
567       totalMass AS Nondimensional
568       ModifyFlow AS Nondimensional
569       #Extra Variables For Section Change at Compressor
570       rhoExtra AS ARRAY(2) OF Density
571       PExtra AS ARRAY(2) OF Pressure
572       hExtra AS ARRAY(2) OF SpecificEnergy
573       TExtra AS ARRAY(2) OF Temperature
574       psiExtra AS ARRAY(2) OF Energy
575       FExtra AS ARRAY(2) OF Flowrate
576     SET
577       Pi:= 4*ATAN(1);
578       R_c := 8.314;       Mw := 4.0E-3;
579       T_stand := 298;       Cp := 20.78;        h_form := 0;    gamma := 1.667;
580       InitT:=291.15;        InitP:= 2.0e6;        NG:=100;        N_Pipes := 4;
581       A((SectionEnd(1)+1):SectionEnd(2)):= 0.3475;  #= achan*nhot = PI*d^2/8*nhot # A(2)
582       A((SectionEnd(3)+1):SectionEnd(4)):= 0.4012; # from filewith LMTD # A(4)
583       HT2.kHe:= 0.34981; # from INL calculations, file with LMTD
584       SectionEnd(1) := 10;
585       SectionEnd(2) := SectionEnd(1) + 40;
586       SectionEnd(3) := SectionEnd(2) + 10;
587       SectionEnd(N_Pipes):= NG;
588       SectionL(1) := 90;    SectionL(2) := 0.98807; SectionL(3) := 90;
589       SectionL(4) :=8.656;
590       r0(1) := 0.413/2;     r0(2) := (9.1652e-4)/2; # dhyd in the PCHE, this is needed to calculate Re
591       r0(3) := 0.444/2;     r0(4):= 0.01123/2; # dhyd in the PHX
592       A(1:SectionEnd(1)):=Pi*r0(1)^2;  # A(1) # 0.133964579
593       A_In:= A(1);
594       A_Out := A(NG);
595       A((SectionEnd(2)+1):SectionEnd(3)):=Pi*r0(3)^2;  # A(3)  # 0.154830252
596       HT1.r0 := r0(1);
597       HT1.r1 := HT1.r0*(2*0.01 + 1); # r1 depends on the t/d ratio, t/d = 0.01 for cold leg with P = 2MPa, INL
            report p16 OK
598       HT1.r2 := HT1.r1 + 0.0058; # INL report p34
599       HT2.r0 := r0(2);
600       HT2.t :=   (HT2.r0*2*0.78*2 - HT2.r0)/3; # according to INL, an average of 2*tmax + 1*tmin
601       HT3.r0 := r0(3);
602       HT3.r1 := HT3.r0*(2*0.11 + 1); # r1 depends on the t/d ratio, t/d = 0.11 for hot leg with P = 2MPa, INL
            report p16
603       HT3.r2 := HT3.r1 + 0.0059; # INL report p34
604       SectionSpan(1) := SectionEnd(1);
605       FOR I:=2 TO N_Pipes DO
606         SectionSpan(I) := SectionEnd(I) - SectionEnd(I-1);
607       END # for
608       SectionDelta_x(1) := SectionL(1)/(SectionSpan(1));
609       SectionDelta_x(2):= SectionL(2)/SectionSpan(2);
610       SectionDelta_x(3):=SectionL(3)/SectionSpan(3);
611       SectionDelta_x(4):=SectionL(4)/SectionSpan(4);
612       Delta_x(1:SectionEnd(1)):= SectionDelta_x(1);
613       Delta_xExtra:= 0.01;
614       FOR I:=1 TO N_Pipes-1 DO
615         Delta_x((SectionEnd(I)+1):SectionEnd(I+1)):= SectionDelta_x(I+1);
616       END # for I  # 20071112 changed
617       Delta_xShift(1:SectionEnd(1)):= SectionDelta_x(1);
618       FOR I := 1 TO N_Pipes-1 DO
```

```
619         Delta_xShift(SectionEnd(I)+1):= (SectionDelta_x(I)+SectionDelta_x(I+1))/2;
620         Delta_xShift(SectionEnd(I)+2:SectionEnd(I+1)):= SectionDelta_x(I+1);
621       END # for
622       Vol(1) := A(1)*SectionDelta_x(1);
623       Vol(2:SectionEnd(1)):= A(2:SectionEnd(1))*SectionDelta_x(1);
624       FOR I:= 1 TO (N_Pipes-1) DO
625         Vol(SectionEnd(I)+1:SectionEnd(I+1)):= A(SectionEnd(I)+1:SectionEnd(I+1))*SectionDelta_x(I+1);
626       END # for
627     INTERMEDIATE
628       Dimension i_U(NG), i_r0(NG), fr(NG)
629       Dimension i_mdot(NG) # mdot at each grid point, all the values should be equal
630       DIMENSION intRe(NG), intfArray(NG),intEps(N_Pipes)
631       i_U(1:SectionEnd(1)):=HT1.U ;
632       i_U((SectionEnd(1)+1):SectionEnd(2)):=HT2.U;
633       i_U((SectionEnd(2)+1):SectionEnd(3)):=HT3.U;
634       i_U((SectionEnd(3)+1):SectionEnd(4)):=HT4.U;
635       i_r0(1:SectionEnd(1)):=r0(1) ;
636       i_r0((SectionEnd(1)+1):SectionEnd(2)):=r0(2);
637       i_r0((SectionEnd(2)+1):SectionEnd(3)):=r0(3);
638       i_r0((SectionEnd(3)+1):SectionEnd(4)):=r0(4);
639       i_rhoPipeHe1:= sigma(rho(1:SectionEnd(1)))/SectionSpan(1);
640       i_vPipe1:=sigma(F(1:SectionEnd(1))/A(1:SectionEnd(1))/rho(1:SectionEnd(1)))/SectionSpan(1);
641       i_rhoPipeHe2:= sigma(rho(SectionEnd(1)+1:SectionEnd(2)))/SectionSpan(2);
642       i_vPipe2:=sigma(F(SectionEnd(1)+1:SectionEnd(2))/A(SectionEnd(1)+1:SectionEnd(2))/rho(SectionEnd(1)+1:
            SectionEnd(2)))/SectionSpan(2);
643       i_rhoPipeHe3:= sigma(rho(SectionEnd(2)+1:SectionEnd(3)))/SectionSpan(3);
644       i_vPipe3:=sigma(F(SectionEnd(2)+1:SectionEnd(3))/A(SectionEnd(2)+1:SectionEnd(3))/rho(SectionEnd(2)+1:
            SectionEnd(3)))/SectionSpan(3);
645       i_rhoPipeHe4:= sigma(rho(SectionEnd(3)+1:SectionEnd(4)))/SectionSpan(4); #20080311
646       i_vPipe4:= sigma(F(SectionEnd(3)+1:SectionEnd(4))/A(SectionEnd(3)+1:SectionEnd(4))/rho(SectionEnd(3)+1:
            SectionEnd(4)))/SectionSpan(4);
647       intEps(1):=4.57e-5;
648       intEps(2):=r0(2)*1e-9;
649       intEps(3):=4.57e-5;
650       intEps(4):=r0(4)*1e-9;
651       intRe(1):= (F_In/A_In + F(1)/A(1))/2*r0(1)*2/muVisc(1);
652       intRe(2:SectionEnd(1)):= (F(1:SectionEnd(1)-1)/A(1:SectionEnd(1)-1)+F(2:SectionEnd(1))/A(2:SectionEnd(1)))/2*
            r0(1)*2/muVisc(1);
653       intRe(SectionEnd(1)+1):= (F(SectionEnd(1))/A(SectionEnd(1))+F(SectionEnd(1)+1)/A(SectionEnd(1)+1))/2*r0(2)*2/
            muVisc(2);
654       intRe(SectionEnd(1)+2:SectionEnd(2)):=(F(SectionEnd(1)+1:SectionEnd(2)-1)/A(SectionEnd(1)+
655         1:SectionEnd(2)-1)+F(SectionEnd(1)+2:SectionEnd(2))/A(SectionEnd(1)+2:SectionEnd(2)))/2*r0(2)*2/muVisc(2);
656       intRe(SectionEnd(2)+1):= (F(SectionEnd(2))/A(SectionEnd(2))+F(SectionEnd(2)+1)/A(SectionEnd(2)+1))/2*r0(3)*2/
            muVisc(3);
657       intRe(SectionEnd(2)+2:SectionEnd(3)):=(F(SectionEnd(2)+1:SectionEnd(3)-1)/A(SectionEnd(2)+
658         1:SectionEnd(3)-1)+F(SectionEnd(2)+2:SectionEnd(3))/A(SectionEnd(2)+2:SectionEnd(3)))/2*r0(3)*2/muVisc(3);
659       intRe(SectionEnd(3)+1):= (F(SectionEnd(3))/A(SectionEnd(3))+F(SectionEnd(3)+1)/A(SectionEnd(3)+1))/2*r0(4)*2/
            muVisc(4);
660       intRe(SectionEnd(3)+2:SectionEnd(4)):=(F(SectionEnd(3)+1:SectionEnd(4)-1)/A(SectionEnd(3)+
661         1:SectionEnd(4)-1)+F(SectionEnd(3)+2:SectionEnd(4))/A(SectionEnd(3)+2:SectionEnd(4)))/2*r0(4)*2/muVisc(4);
662       intfArray(1:SectionEnd(1)):=1/(-2*LOG10(intEps(1)/(3.7*r0(1)) + 2.51/intRe(1:SectionEnd(1))
663         *(1.14 -2*LOG10(intEps(1)/r0(1) + 21.25/(intRe(1:SectionEnd(1))^0.9)))))^2/4;
664       intfArray(SectionEnd(1)+1:SectionEnd(2)):=64/intRe(SectionEnd(1)+1:SectionEnd(2))/4;
665       intfArray(SectionEnd(2)+1:SectionEnd(3)):=1/(-2*LOG10(intEps(3)/(3.7*r0(3))
666         + 2.51/intRe(SectionEnd(2)+1:SectionEnd(3))
667         *(1.14 -2*LOG10(intEps(3)/r0(3) + 21.25/(intRe(SectionEnd(2)+1:SectionEnd(3))^0.9)))))^2/4;
668       intfArray(SectionEnd(3)+1:SectionEnd(4)):=1/(-2*LOG10(intEps(4)/(3.7*r0(4))
669         + 2.51/intRe(SectionEnd(3)+1:SectionEnd(4))
670         *(1.14 -2*LOG10(intEps(4)/r0(4) + 21.25/(intRe(SectionEnd(3)+1:SectionEnd(4))^0.9)))))^2/4;
671       fr(1:SectionEnd(1))                 := intfArray(1:SectionEnd(1));#0.01294/4; # OK
672       fr((SectionEnd(1)+1):SectionEnd(2))   := intfArray(SectionEnd(1)+1:SectionEnd(2));#4.009E-02/4; # OK
673       fr((SectionEnd(2)+1):SectionEnd(3))   := intfArray(SectionEnd(2)+1:SectionEnd(3));#0.01348/4; # OK
674       fr((SectionEnd(3)+1):SectionEnd(4))   := intfArray(SectionEnd(3)+1:SectionEnd(4));#2.758E-02/4; # OK
675     EQUATION
676       # Mass Balance
677       0  = (F_In - FExtra(1));
678       0  = (FExtra(1) - FExtra(2));
679       0  = (FExtra(2) - F(1));
680       FOR I:= 2 TO NG-1 DO
681         0 = (F(I-1) - F(I));
682       END # for
683       # Momentum Balance
684       0 =
685         -(PExtra(2)*A(NG) - PExtra(1)*A(NG))
686         +(   F_In                    )^2/     (rhoExtra(1)*A(NG))
687         -(   (  FExtra(1) + FExtra(2)  )/2   )^2   /   (rhoExtra(2)*A(NG));
688       0 =
689         -(    P(1)*A(1)          -          ( PExtra(2)*Delta_x(1)/2 + P(1)*Delta_xExtra/2 ) / ( Delta_x(1)/2 +
              Delta_xExtra/2)  * ( A(1) - A(NG) ) -  PExtra(2)*A(NG)  )
690         +(   (    FExtra(1) + FExtra(2)   )/2   )^2  /   (  rhoExtra(2)*A(NG)  )
691         -(   (     F(1)    + FExtra(2)      )/2    )^2  /   (  rho(1)*A(1)  )
692         - 2*fr(1)/( 2*i_r0(1) )    *   (  ABS(FExtra(2))*FExtra(2)   )/  (  rho(1)*A(1)  )  *  Delta_x(1)/2;
693       0 =
694         -(    P(2)*A(2)     -         ( P(1)*Delta_x(2)/2 + P(2)*Delta_x(1)/2 )  /  ( Delta_x(1)/2+Delta_x(2)/2 )
              *  ( A(2) - A(1) )      -     P(1)*A(1)    )
695         +(   (    FExtra(2)  +  F(1)    )/2   )^2     /    (  rho(1)*A(1)  )
696         -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A(2)   )
697         - 2*fr(1)/( 2*i_r0(1) )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A(1)  )  *  Delta_x(1)/2
698         - 2*fr(2)/( 2*i_r0(2) )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A(2)  )  *  Delta_x(2)/2;
699       FOR I:=2 TO SectionEnd(4)-1 DO
700         0 =
```

```
        -(    P(I+1)*A(I+1)     -         ( P(I)*Delta_x(I+1)/2 + P(I+1)*Delta_x(I)/2 )  /  ( Delta_x(I)/2+
              Delta_x(I+1)/2 )  *  ( A(I +1) - A(I) )      -     P(I)*A(I)    )
        +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A(I)  )
        -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A(I+1)   )
        - 2*fr(I)/( 2*i_r0(I) )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A(I)  )  *  Delta_x(I)/2
        - 2*fr(I+1)/( 2*i_r0(I+1) )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A(I+1)  )  *  Delta_x(I+1)/2;
    END
      0 =
        -(    P_Out*A_Out    -     P(NG)*A(NG)    )
        +(   (    F(NG-1)  +  F(NG)    )/2   )^2     /    (  rho(NG)*A(NG)  )
        -(   F(NG)    )^2    /     (   rho(NG)*A(NG)   )
        - 2*fr(NG)/( 2*i_r0(NG) )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A(NG)  )  *  Delta_x(NG)/2;

    # Energy Balance
      Delta_xExtra*A(NG)*$psiExtra(1) = (F_In*h0 - FExtra(1)*hExtra(1));
      Delta_xExtra*A(NG)*$psiExtra(2) = (FExtra(1)*hExtra(1) - FExtra(2)*hExtra(2));
      Vol(1)*$psi(1) = (FExtra(2)*hExtra(2) - F(1)*h(1)
          +4*i_U(1)/(2*i_r0(1))*(TExtrnl(1)-T(1))*Vol(1));
      FOR I:=2 TO NG DO
        Vol(I)*$psi(I) = (F(I-1)*h(I-1) - F(I)*h(I)
          +4*i_U(I)/(2*i_r0(I))*(TExtrnl(I)-T(I))*Vol(I));
      END # for

    # Algebraic Equations
      0 = (psiExtra(1)  - (rhoExtra(1)*hExtra(1) - PExtra(1)));
      0 = (psiExtra(2)  - (rhoExtra(2)*hExtra(2) - PExtra(2)));
    FOR I:=1 TO NG DO
        0 = (psi(I)  - (rho(I)*h(I) - P(I)));
    END # for

    # 2- Thermodynamical Part
     0 = 1e-7*(hExtra(1) - (h_form + Cp/Mw*(TExtra(1)-T_stand)));
     0 = 1e-7*(hExtra(2) - (h_form + Cp/Mw*(TExtra(2)-T_stand)));
    FOR I:= 1 TO NG DO
       0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
    END # for
    0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));

    # Equation of State
    FOR I:=1 TO NG DO
        0= (P(I)*Mw - rho(I)*R_c*T(I));
    END # for
    0= (PExtra(1)*Mw - rhoExtra(1)*R_c*TExtra(1));
    0= (PExtra(2)*Mw - rhoExtra(2)*R_c*TExtra(2));

    # 3 - Forced Continuity
    # Equation to set the amount of Gas <=> average density
    rho_avg = (SIGMA(rho*Vol)+ (rhoExtra(1) + rhoExtra(2))*Delta_xExtra*A(NG)) /(SIGMA(Vol) + 2*Delta_xExtra*A(NG
          )) ;
    totalMass = SIGMA(rho*Vol) + (rhoExtra(1) + rhoExtra(2))*Delta_xExtra*A(NG);
    F_In = F(NG) + ModifyFlow;

    # Compressor Equations
    # deltaP
      0 = (DeltaP  - (PExtra(1) - P_Out));
    # energy 1
      0 = P_Out^(gamma-1)*T0^gamma - PExtra(1)^(gamma-1)*T(NG)^gamma;

    #HT1 equations
    HT1.rhoPipeHe = i_rhoPipeHe1;
    HT1.vPipe = i_vPipe1;
    HT2.rhoPipeHe = i_rhoPipeHe2;
    HT2.vPipe = i_vPipe2;
    HT3.rhoPipeHe =i_rhoPipeHe3;
    HT3.vPipe = i_vPipe3;
    HT4.rhoPipeHe= i_rhoPipeHe4;
    HT4.vPipe = i_vPipe4;
END
######################################################################


######################################################################
MODEL PipeQSS
######################################################################
  PARAMETER
        #SET here
        Pi AS REAL
    R_c AS REAL        #Rc constant "J/(mol*K)
    Mw AS REAL         #Molecular weight "Kg/mol"
    T_stand AS REAL    #Standard Temp "K"
    Cp AS REAL         #specific heat at constant pressure "J/Kg/K"
    h_form AS REAL     #specific enthalpy of formation "J/Kg"
    A AS REAL
    r0 AS REAL       #diameter of pipe "m"
    muVisc as REAL # average viscosity in pipe

    #EXTERNALLY set
    SectionL AS REAL           #length of pipe "m"
    NG AS INTEGER       #number of control volumes in the grid
    Temp AS REAL       #Temperature "K"
    InitP AS REAL
    InitT AS REAL
```

```
791
792       #CALCULATED
793       InitRho AS REAL     InitH AS REAL
794       InitRho_e AS REAL   Delta_x AS REAL     #length of control volume "m"
795       Vol AS REAL
796     UNIT
797       HT AS HTCoeffGasMetalHX
798     VARIABLE
799       rho AS ARRAY(NG) OF Density
800       P AS ARRAY(NG) OF Pressure
801       P_In AS Pressure
802       P_Out AS Pressure
803       h as ARRAY(NG) OF SpecificEnergy
804       h0 AS SpecificEnergy
805       T as ARRAY(NG) OF Temperature
806       TExtrnl AS ARRAY(NG) OF Temperature
807       T0 AS Temperature
808       #DIFFERENTIAL
809       F AS ARRAY(NG) OF Flowrate
810       F_In AS Flowrate
811       psi AS ARRAY(NG) OF Energy
812     SET
813       #SET
814       Pi:=4*ATAN(1);
815       R_c := 8.314;     Mw := 4.0E-3;
816       T_stand := 298;   Cp := 20.78;        h_form := 0;
817       r0 := 9.1652e-4/2 ; # dhyd in the PCHE, this is needed to calculate Re
818       A:=0.3475;
819       #CALCULATED
820       Delta_x := SectionL/(NG-1);
821       InitRho := InitP*Mw/(R_c*Temp);
822       InitH := h_form + Cp/Mw*(Temp-T_stand);
823       InitRho_e := InitRho*InitH - InitP;
824       HT.r0:=r0;        HT.t:= (HT.r0*2*0.78*2 - HT.r0)/3; # according to INL, an average of 2*tmax +1*tmin
825       Vol := A*Delta_x;
826     INTERMEDIATE
827       DIMENSION intRe(NG)
828       DIMENSION intfArray(NG)
829       U:= HT.U;
830       intRe:= F*r0*2/muVisc/A;
831       intfArray:=64/intRe/4;
832     EQUATION
833       # Mass Balance
834       0 = F_In - F(1);
835       FOR I:= 2 TO NG DO
836         0 = (F(I-1) - F(I));
837       END # for
838       # Momentum Balance
839       0 =  -(P(1)*A -P_In*A)
840           +(   F_In   )^2  /     (rho(1)*A)
841           -(    ( F_In + F(1) )/2    )^2  /     (rho(1)*A)
842           - (2*intfArray(1)/(2*r0))  *  ( ABS(F_In)*F_In )   /   (rho(1)*A)  *   (Delta_x)/2;
843       0 =
844           -(    P(2)*A     -         P(1)*A    )
845           +(   (    F_In  +  F(1)    )/2   )^2     /    (  rho(1)*A  )
846           -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A   )
847           - 2*intfArray(1)/( 2*r0 )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A  )  *  Delta_x/2
848           - 2*intfArray(2)/( 2*r0 )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A  )  *  Delta_x/2;
849       FOR I:=2 TO NG-1 DO
850       0 =
851           -(    P(I+1)*A     -         P(I)*A    )
852           +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A  )
853           -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A   )
854           - 2*intfArray(I)/( 2*r0 )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A  )  *  Delta_x/2
855           - 2*intfArray(I+1)/( 2*r0 )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A  )  *  Delta_x/2;
856       END # for
857       0 =
858         -(    P_Out*A    -     P(NG)*A    )
859         +(   (    F(NG-1)  +  F(NG)    )/2   )^2     /    (  rho(NG)*A  )
860         -(   F(NG)    )^2    /     (   rho(NG)*A   )
861         - 2*intfArray(NG)/( 2*r0 )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A  )  *  Delta_x/2;
862       # Energy Balance
863         1e-7*$psi(1)*Vol = 1e-7*(F_In*h0 - F(1)*h(1)
864           +4*U/(2*r0)*(TExtrnl(1)-T(1 ))*Vol);
865       FOR I:=(2) TO (NG) DO
866           1e-7*$psi(I)*Vol = 1e-7*(F(I-1)*h(I-1) - F(I)*h(I)
867             +4*U/(2*r0)*(TExtrnl(I)-T(I))*Vol);
868       END # for
869
870       # Algebraic Equations
871       FOR I:=1 TO NG DO
872           0 = 1e-7*(psi(I) - (rho(I)*h(I) - P(I)));
873       END # for
874
875       # 2- Thermodynamical Part
876       FOR I:= 1 TO NG DO
877           0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
878       END # for
879       0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));
880
881       # Equation of State
882       FOR I:=1 TO NG DO
```

```
        0=1e-3*(P(I)*Mw - rho(I)*R_c*T(I));
    END # for

    #Equations to assign variables for heat transfer coefficients
    HT.rhoPipeHe = sigma(rho(2:NG))/NG;
    HT.vPipe = sigma(F(2:NG)/rho(2:NG)/A)/NG;
END
#####################################################################


#####################################################################
MODEL Pipe0Mach
#####################################################################
  PARAMETER
        #SET here
        Pi AS REAL
    R_c AS REAL        #Rc constant "J/(mol*K)
    Mw AS REAL         #Molecular weight "Kg/mol"
    T_stand AS REAL    #Standard Temp "K"
    Cp AS REAL         #specific heat at constant pressure "J/Kg/K"
    h_form AS REAL     #specific enthalpy of formation "J/Kg"
    A AS REAL
    r0 AS REAL       #diameter of pipe "m"
    muVisc as REAL # average viscosity in pipe

    #EXTERNALLY set
    SectionL AS REAL           #length of pipe "m"
    NG AS INTEGER       #number of control volumes in the grid
    Temp AS REAL       #Temperature "K"
    InitP AS REAL
    InitT AS REAL

    #CALCULATED
    InitRho AS REAL     InitH AS REAL
    InitRho_e AS REAL   Delta_x AS REAL     #length of control volume "m"
    Vol AS REAL
  UNIT
    HT AS HTCoeffGasMetalHX
    VARIABLE
    rho AS ARRAY(NG) OF Density
    P AS ARRAY(NG) OF Pressure
    P_In AS Pressure
    P_Out AS Pressure
    h as ARRAY(NG) OF SpecificEnergy
    h0 AS SpecificEnergy
    T as ARRAY(NG) OF Temperature
    TExtrnl AS ARRAY(NG) OF Temperature
    T0 AS Temperature
    #DIFFERENTIAL
    F AS ARRAY(NG) OF Flowrate
    F_In AS Flowrate
    psi AS ARRAY(NG) OF Energy
      SET
    #SET
    Pi:=4*ATAN(1);
    R_c := 8.314;     Mw := 4.0E-3;
    T_stand := 298;   Cp := 20.78;        h_form := 0;
    r0 := 9.1652e-4/2 ; # dhyd in the PCHE, this is needed to calculate Re
    A:=0.3475;
    #CALCULATED
    Delta_x := SectionL/(NG-1);
    InitRho := InitP*Mw/(R_c*Temp);
    InitH := h_form + Cp/Mw*(Temp-T_stand);
    InitRho_e := InitRho*InitH - InitP;
    HT.r0:=r0;        HT.t:= (HT.r0*2*0.78*2 - HT.r0)/3; # according to INL, an average of 2*tmax +1*tmin
    Vol := A*Delta_x;
  INTERMEDIATE
    DIMENSION intRe(NG)
    DIMENSION intfArray(NG)
    U:= HT.U;
    intRe:= F*r0*2/muVisc/A;
    intfArray:=64/intRe/4;
  EQUATION
    # Mass Balance
    Delta_x*A*$rho(1) = F_In - F(1);
    FOR I:= 2 TO NG DO
      Delta_x*A*$rho(I) = (F(I-1) - F(I));
    END # for
    # Momentum Balance
    0 =  -(P(1)*A -P_In*A)
        +(   F_In   )^2  /     (rho(1)*A)
        -(    ( F_In + F(1) )/2    )^2  /     (rho(1)*A)
        - (2*intfArray(1)/(2*r0))  * ( ABS(F_In)*F_In )   /   (rho(1)*A)  *   (Delta_x)/2;
    0 =
        -(    P(2)*A     -         P(1)*A    )
        +(   (    F_In  +  F(1)    )/2   )^2     /    (  rho(1)*A  )
        -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A   )
        - 2*intfArray(1)/( 2*r0 )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A  )  *  Delta_x/2
        - 2*intfArray(2)/( 2*r0 )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A  )  *  Delta_x/2;
    FOR I:=2 TO NG-1 DO
      0 =
        -(    P(I+1)*A     -         P(I)*A    )
```

```
975          +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A  )
976          -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A   )
977          - 2*intfArray(I)/( 2*r0 )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A  )  *  Delta_x/2
978          - 2*intfArray(I+1)/( 2*r0 )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A  )  *  Delta_x/2;
979      END # for
980      0 =
981        -(    P_Out*A    -     P(NG)*A    )
982        +(   (    F(NG-1)  +  F(NG)    )/2   )^2     /    (  rho(NG)*A  )
983        -(   F(NG)    )^2    /     (   rho(NG)*A   )
984        - 2*intfArray(NG)/( 2*r0 )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A  )  *  Delta_x/2;
985      # Energy Balance
986        1e-7*$psi(1)*Vol = 1e-7*(F_In*h0 - F(1)*h(1)
987          +4*U/(2*r0)*(TExtrnl(1)-T(1 ))*Vol);
988      FOR I:=(2) TO (NG) DO
989          1e-7*$psi(I)*Vol = 1e-7*(F(I-1)*h(I-1) - F(I)*h(I)
990            +4*U/(2*r0)*(TExtrnl(I)-T(I))*Vol);
991      END # for
992
993      # Algebraic Equations
994      FOR I:=1 TO NG DO
995          0 = 1e-7*(psi(I) - (rho(I)*h(I) - P(I)));
996      END # for
997
998      # 2- Thermodynamical Part
999      FOR I:= 1 TO NG DO
1000         0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
1001     END # for
1002     0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));
1003
1004     # Equation of State
1005     FOR I:=1 TO NG DO
1006         0=1e-3*(P(I)*Mw - rho(I)*R_c*T(I));
1007     END # for
1008
1009     #Equations to assign variables for heat transfer coefficients
1010     HT.rhoPipeHe = sigma(rho(2:NG))/NG;
1011     HT.vPipe = sigma(F(2:NG)/rho(2:NG)/A)/NG;
1012 END
1013 ######################################################################
1014
1015
1016 ######################################################################
1017 MODEL PipeFullDyn
1018 ######################################################################
1019   PARAMETER
1020         #SET here
1021         Pi AS REAL
1022     R_c AS REAL        #Rc constant "J/(mol*K)
1023     Mw AS REAL         #Molecular weight "Kg/mol"
1024     T_stand AS REAL    #Standard Temp "K"
1025     Cp AS REAL         #specific heat at constant pressure "J/Kg/K"
1026     h_form AS REAL     #specific enthalpy of formation "J/Kg"
1027     A AS REAL
1028     r0 AS REAL       #diameter of pipe "m"
1029     muVisc as REAL # average viscosity in pipe
1030
1031     #EXTERNALLY set
1032     SectionL AS REAL           #length of pipe "m"
1033     NG AS INTEGER       #number of control volumes in the grid
1034     Temp AS REAL       #Temperature "K"
1035     InitP AS REAL
1036     InitT AS REAL
1037
1038     #CALCULATED
1039     InitRho AS REAL     InitH AS REAL
1040     InitRho_e AS REAL   Delta_x AS REAL     #length of control volume "m"
1041     Vol AS REAL
1042   UNIT
1043     HT AS HTCoeffGasMetalHX
1044   VARIABLE
1045     rho AS ARRAY(NG) OF Density
1046     P AS ARRAY(NG) OF Pressure
1047     P_In AS Pressure
1048     P_Out AS Pressure
1049     h as ARRAY(NG) OF SpecificEnergy
1050     h0 AS SpecificEnergy
1051     T as ARRAY(NG) OF Temperature
1052     TExtrnl AS ARRAY(NG) OF Temperature
1053     T0 AS Temperature
1054     F AS ARRAY(NG) OF Flowrate
1055     F_In AS Flowrate
1056     psi AS ARRAY(NG) OF Energy
1057   SET
1058     #SET
1059     Pi:=4*ATAN(1);
1060     R_c := 8.314;     Mw := 4.0E-3;
1061     T_stand := 298;   Cp := 20.78;        h_form := 0;
1062     r0 := 9.1652e-4/2 ; # dhyd in the PCHE, this is needed to calculate Re
1063     A:=0.3475;
1064     #CALCULATED
1065     Delta_x := SectionL/(NG-1);
1066     InitRho := InitP*Mw/(R_c*Temp);
```

```
1067      InitH := h_form + Cp/Mw*(Temp-T_stand);
1068      InitRho_e := InitRho*InitH - InitP;
1069      HT.r0:=r0;        HT.t:= (HT.r0*2*0.78*2 - HT.r0)/3; # according to INL, an average of 2*tmax +1*tmin
1070      Vol := A*Delta_x;
1071    INTERMEDIATE
1072      DIMENSION intRe(NG)
1073      DIMENSION intfArray(NG)
1074      U:= HT.U;
1075      intRe:= F*r0*2/muVisc/A;
1076      intfArray:=64/intRe/4;
1077    EQUATION
1078      # Mass Balance
1079      Delta_x*A*$rho(1) = F_In - F(1);
1080      FOR I:= 2 TO NG DO
1081        Delta_x*A*$rho(I) = (F(I-1) - F(I));
1082      END # for
1083      # Momentum Balance
1084      Delta_x/2*$F_In =  -(P(1)*A -P_In*A)
1085          +(   F_In   )^2  /     (rho(1)*A)
1086          -(    ( F_In + F(1) )/2    )^2  /     (rho(1)*A)
1087          - (2*intfArray(1)/(2*r0))  *  (ABS(F_In)*F_In )  /   (rho(1)*A)  *   (Delta_x)/2;
1088      Delta_x    * $F(1) =
1089          -(    P(2)*A     -         P(1)*A    )
1090          +(   (    F_In  +  F(1)    )/2   )^2     /    (  rho(1)*A  )
1091          -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A   )
1092          - 2*intfArray(1)/( 2*r0 )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A  )  *  Delta_x/2
1093          - 2*intfArray(2)/( 2*r0 )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A  )  *  Delta_x/2;
1094      FOR I:=2 TO NG-1 DO
1095        Delta_x    * $F(I) =
1096          -(    P(I+1)*A     -         P(I)*A    )
1097          +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A  )
1098          -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A   )
1099          - 2*intfArray(I)/( 2*r0 )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A  )  *  Delta_x/2
1100          - 2*intfArray(I+1)/( 2*r0 )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A  )  *  Delta_x/2;
1101      END # for
1102      Delta_x /2    * $F(NG) =
1103        -(    P_Out*A    -     P(NG)*A    )
1104        +(   (    F(NG-1)  +  F(NG)    )/2   )^2     /    (  rho(NG)*A  )
1105        -(   F(NG)    )^2    /     (   rho(NG)*A   )
1106        - 2*intfArray(NG)/( 2*r0 )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A  )  *  Delta_x/2;
1107      # Energy Balance
1108        1e-7*$psi(1)*Vol = 1e-7*(F_In*h0 - F(1)*h(1)
1109          +4*U/(2*r0)*(TExtrnl(1)-T(1 ))*Vol);
1110      FOR I:=(2) TO (NG) DO
1111          1e-7*$psi(I)*Vol = 1e-7*(F(I-1)*h(I-1) - F(I)*h(I)
1112            +4*U/(2*r0)*(TExtrnl(I)-T(I))*Vol);
1113      END # for
1114
1115      # Algebraic Equations
1116      FOR I:=1 TO NG DO
1117          0 = 1e-7*(psi(I) - (rho(I)*h(I) - P(I)));
1118      END # for
1119
1120      # 2- Thermodynamical Part
1121      FOR I:= 1 TO NG DO
1122          0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
1123      END # for
1124      0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));
1125
1126      # Equation of State
1127      FOR I:=1 TO NG DO
1128          0=1e-3*(P(I)*Mw - rho(I)*R_c*T(I));
1129      END # for
1130
1131      #Equations to assign variables for heat transfer coefficients
1132      HT.rhoPipeHe = sigma(rho(2:NG))/NG;
1133      HT.vPipe = sigma(F(2:NG)/rho(2:NG)/A)/NG;
1134  END
1135  ######################################################################
1136
1137
1138  ######################################################################
1139  MODEL PipeInnerQSS
1140  ######################################################################
1141    PARAMETER
1142          Pi AS REAL
1143      R_c AS REAL        #Rc constant "J/(mol*K)
1144      Mw AS REAL         #Molecular weight "Kg/mol"
1145      T_stand AS REAL    #Standard Temp "K"
1146      Cp AS REAL         #specific heat at constant pressure "J/Kg/K"
1147      h_form AS REAL     #specific enthalpy of formation "J/Kg"
1148      A AS REAL
1149      r0 AS REAL       #diameter of pipe "m"
1150      muVisc AS REAL # average viscosity in
1151      SectionL AS REAL           #length of pipe "m"
1152      NG AS INTEGER       #number of control volumes in the grid
1153      Temp AS REAL       #Temperature "K"
1154      InitP AS REAL
1155      InitRho AS REAL     InitH AS REAL
1156      InitRho_e AS REAL   Delta_x AS REAL     #length of control volume "m"
1157      Vol AS REAL
1158    UNIT
```

```
    HT AS HTCoeffGasMetalHXDummyInner
  VARIABLE
    rho AS ARRAY(NG) OF Density
    P AS ARRAY(NG) OF Pressure
    P_In AS Pressure
    P_Out AS Pressure
    h as ARRAY(NG) OF SpecificEnergy
    h0 AS SpecificEnergy
    T as ARRAY(NG) OF Temperature
    T0 AS Temperature
    TExtrnl AS ARRAY(NG) OF Temperature
    F AS ARRAY(NG) OF Flowrate
    F_In AS Flowrate
    psi AS ARRAY(NG) OF Energy
  SET
    #SET
    Pi:=4*ATAN(1);
    R_c := 8.314;     Mw :=  18.02E-3; #steam # 4.0E-3;  # helium
    T_stand := 298;   Cp := 40.65; #steam #20.78; #helium
    h_form := 0.0; # -57.8; #enthalpy of formation from perry's water gas
    r0 :=0.01/2 ; # Hyd radious based on PCHE INL report,
    A:=0.2750;
    #CALCULATED
    Delta_x := SectionL/(NG-1);
    InitRho := InitP*Mw/(R_c*Temp);
    InitH := h_form + Cp/Mw*(Temp-T_stand);
    InitRho_e := InitRho*InitH - InitP;
    Vol := A*Delta_x;
  INTERMEDIATE
    DIMENSION intRe(NG)
    DIMENSION intfArray(NG)
    U:= HT.U;
    intEps:= r0*1e-9;
    intRe:=F*r0*2/A/muVisc;
    intfArray:=1/(-2*LOG10(intEps/(3.7*r0) + 2.51/intRe*(1.14 -2*LOG10(intEps/r0 + 21.25/(intRe^0.9)))))^2/4;
  EQUATION
    # Mass Balance
    0 = F_In - F(1);
    FOR I:= 2 TO NG DO
      0 = (F(I-1) - F(I));
    END # for
    # Momentum Balance
    0 =  -(P(1)*A -P_In*A)
        +(   F_In   )^2  /     (rho(1)*A)
        -(    ( F_In + F(1) )/2    )^2  /     (rho(1)*A)
        - (2*intfArray(1)/(2*r0))  * (ABS(F_In)*F_In )   /   (rho(1)*A)  *   (Delta_x)/2;
    0 =
        -(    P(2)*A     -         P(1)*A    )
        +(   (    F_In  +  F(1)    )/2   )^2     /    (  rho(1)*A  )
        -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A   )
        - 2*intfArray(1)/( 2*r0 )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A  )  *  Delta_x/2
        - 2*intfArray(2)/( 2*r0 )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A  )  *  Delta_x/2;
    FOR I:=2 TO NG-1 DO
      0 =
        -(    P(I+1)*A     -         P(I)*A    )
        +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A  )
        -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A   )
        - 2*intfArray(I)/( 2*r0 )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A  )  *  Delta_x/2
        - 2*intfArray(I+1)/( 2*r0 )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A  )  *  Delta_x/2;
    END # for
    0 =
      -(    P_Out*A    -     P(NG)*A    )
      +(   (    F(NG-1)  +  F(NG)    )/2   )^2     /    (  rho(NG)*A  )
      -(   F(NG)    )^2    /     (   rho(NG)*A   )
      - 2*intfArray(NG)/( 2*r0 )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A  )  *  Delta_x/2;
    # Energy Balance
      1e-7*$psi(1)*Vol = 1e-7*(F_In*h0 - F(1)*h(1)
        +4*U/(2*r0)*(TExtrnl(1)-T(1 ))*Vol);
    FOR I:=(2) TO (NG) DO
        1e-7*$psi(I)*Vol = 1e-7*(F(I-1)*h(I-1) - F(I)*h(I)
          +4*U/(2*r0)*(TExtrnl(I)-T(I))*Vol);  # the last term comes from heat in/volume to heat in Work: Pipe
              Simulation
    END # for

    # Algebraic Equations
    FOR I:=1 TO NG DO
        0 = 1e-7*(psi(I) - (rho(I)*h(I) - P(I)));
    END # for

    # 2- Thermodynamical Part
    FOR I:= 1 TO NG DO
        0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
    END # for
    0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));
    # Equation of State
    FOR I:=1 TO NG DO
        0=1e-3*(P(I)*Mw - rho(I)*R_c*T(I));
    END # for

    #Equations to assign variables for heat transfer coefficients
    HT.rhoPipeHe = sigma(rho(2:NG))/NG;
    HT.vPipe = sigma(F(2:NG)/rho(2:NG)/A)/NG;
```

```
END
#######################################################################


#######################################################################
MODEL PipeInner0Mach
#######################################################################
  PARAMETER
        Pi AS REAL
    R_c AS REAL        #Rc constant "J/(mol*K)
    Mw AS REAL         #Molecular weight "Kg/mol"
    T_stand AS REAL    #Standard Temp "K"
    Cp AS REAL         #specific heat at constant pressure "J/Kg/K"
    h_form AS REAL     #specific enthalpy of formation "J/Kg"
    A AS REAL
    r0 AS REAL       #diameter of pipe "m"
    muVisc AS REAL # average viscosity in
    SectionL AS REAL           #length of pipe "m"
    NG AS INTEGER       #number of control volumes in the grid
    Temp AS REAL       #Temperature "K"
    InitP AS REAL
    InitRho AS REAL     InitH AS REAL
    InitRho_e AS REAL   Delta_x AS REAL     #length of control volume "m"
    Vol AS REAL
  UNIT
    HT AS HTCoeffGasMetalHXDummyInner
  VARIABLE
    rho AS ARRAY(NG) OF Density
    P AS ARRAY(NG) OF Pressure
    P_In AS Pressure
    P_Out AS Pressure
    h as ARRAY(NG) OF SpecificEnergy
    h0 AS SpecificEnergy
    T as ARRAY(NG) OF Temperature
    T0 AS Temperature
    TExtrnl AS ARRAY(NG) OF Temperature
    F AS ARRAY(NG) OF Flowrate
    F_In AS Flowrate
    psi AS ARRAY(NG) OF Energy
  SET
    #SET
    Pi:=4*ATAN(1);
    R_c := 8.314;     Mw :=  18.02E-3; #steam # 4.0E-3;  # helium
    T_stand := 298;   Cp := 40.65; #steam #20.78; #helium
    h_form := 0.0; # -57.8; #enthalpy of formation from perry's water gas
    r0 :=0.01/2 ; # Hyd radious based on PCHE INL report,
    A:=0.2750;
    #CALCULATED
    Delta_x := SectionL/(NG-1);
    InitRho := InitP*Mw/(R_c*Temp);
    InitH := h_form + Cp/Mw*(Temp-T_stand);
    InitRho_e := InitRho*InitH - InitP;
    Vol := A*Delta_x;
  INTERMEDIATE
    DIMENSION intRe(NG)
    DIMENSION intfArray(NG)
    U:= HT.U;
    intEps:= r0*1e-9;
    intRe:=F*r0*2/A/muVisc;
    intfArray:=1/(-2*LOG10(intEps/(3.7*r0) + 2.51/intRe*(1.14 -2*LOG10(intEps/r0 + 21.25/(intRe^0.9)))))^2/4;
  EQUATION
    # Mass Balance
    Delta_x*A*$rho(1) = F_In - F(1);
    FOR I:= 2 TO NG DO
      Delta_x*A*$rho(I) = (F(I-1) - F(I));
    END # for
    # Momentum Balance
    0 =  1e-4*(   -(P(1)*A -P_In*A)
        +(   F_In   )^2  /     (rho(1)*A)
        -(    ( F_In + F(1) )/2    )^2  /     (rho(1)*A)
        - (2*intfArray(1)/(2*r0))  * ( ABS(F_In)*F_In )  /   (rho(1)*A)  *   (Delta_x)/2    );
    0 =
      1e-4*(    -(    P(2)*A     -         P(1)*A    )
        +(   (    F_In  +  F(1)    )/2   )^2     /    (  rho(1)*A  )
        -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A   )
        - 2*intfArray(1)/( 2*r0 )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A  )  *  Delta_x/2
        - 2*intfArray(2)/( 2*r0 )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A  )  *  Delta_x/2    );
    FOR I:=2 TO NG-1 DO
      0 =
        1e-4*(   -(    P(I+1)*A     -         P(I)*A    )
        +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A  )
        -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A   )
        - 2*intfArray(I)/( 2*r0 )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A  )  *  Delta_x/2
        - 2*intfArray(I+1)/( 2*r0 )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A  )  *  Delta_x/2    );
    END # for
    0 =
      1e-4*(    -(    P_Out*A    -     P(NG)*A    )
      +(   (    F(NG-1)  +  F(NG)    )/2   )^2     /    (  rho(NG)*A  )
      -(   F(NG)    )^2    /     (   rho(NG)*A   )
      - 2*intfArray(NG)/( 2*r0 )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A  )  *  Delta_x/2     );
    # Energy Balance
      1e-7*$psi(1)*Vol = 1e-7*(F_In*h0 - F(1)*h(1)
```

```
1342          +4*U/(2*r0)*(TExtrnl(1)-T(1 ))*Vol);
1343      FOR I:=(2) TO (NG) DO
1344          1e-7*$psi(I)*Vol = 1e-7*(F(I-1)*h(I-1) - F(I)*h(I)
1345            +4*U/(2*r0)*(TExtrnl(I)-T(I))*Vol);  # the last term comes from heat in/volume to heat in Work: Pipe
                  Simulation
1346      END # for
1347
1348      # Algebraic Equations
1349      FOR I:=1 TO NG DO
1350          0 = 1e-7*(psi(I) - (rho(I)*h(I) - P(I)));
1351      END # for
1352
1353      # 2- Thermodynamical Part
1354      FOR I:= 1 TO NG DO
1355          0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
1356      END # for
1357      0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));
1358      # Equation of State
1359      FOR I:=1 TO NG DO
1360          0=1e-3*(P(I)*Mw - rho(I)*R_c*T(I));
1361      END # for
1362
1363      #Equations to assign variables for heat transfer coefficients
1364      HT.rhoPipeHe = sigma(rho(2:NG))/NG;
1365      HT.vPipe = sigma(F(2:NG)/rho(2:NG)/A)/NG;
1366  END
1367  ######################################################################
1368
1369
1370  ######################################################################
1371  MODEL PipeInnerFullDyn
1372  ######################################################################
1373    PARAMETER
1374          Pi AS REAL
1375      R_c AS REAL        #Rc constant "J/(mol*K)
1376      Mw AS REAL         #Molecular weight "Kg/mol"
1377      T_stand AS REAL    #Standard Temp "K"
1378      Cp AS REAL         #specific heat at constant pressure "J/Kg/K"
1379      h_form AS REAL     #specific enthalpy of formation "J/Kg"
1380      A AS REAL
1381      r0 AS REAL       #diameter of pipe "m"
1382      muVisc AS REAL # average viscosity in
1383      SectionL AS REAL           #length of pipe "m"
1384      NG AS INTEGER       #number of control volumes in the grid
1385      Temp AS REAL       #Temperature "K"
1386      InitP AS REAL
1387      InitRho AS REAL     InitH AS REAL
1388      InitRho_e AS REAL   Delta_x AS REAL     #length of control volume "m"
1389      Vol AS REAL
1390    UNIT
1391      HT AS HTCoeffGasMetalHXDummyInner
1392    VARIABLE
1393      rho AS ARRAY(NG) OF Density
1394      P AS ARRAY(NG) OF Pressure
1395      P_In AS Pressure
1396      P_Out AS Pressure
1397      h as ARRAY(NG) OF SpecificEnergy
1398      h0 AS SpecificEnergy
1399      T as ARRAY(NG) OF Temperature
1400      T0 AS Temperature
1401      TExtrnl AS ARRAY(NG) OF Temperature
1402      F AS ARRAY(NG) OF Flowrate
1403      F_In AS Flowrate
1404      psi AS ARRAY(NG) OF Energy
1405    SET
1406      #SET
1407      Pi:=4*ATAN(1);
1408      R_c := 8.314;     Mw :=  18.02E-3; #steam # 4.0E-3;  # helium
1409      T_stand := 298;   Cp := 40.65; #steam #20.78; #helium
1410      h_form := 0.0; # -57.8; #enthalpy of formation from perry's water gas
1411      r0 :=0.01/2 ; # Hyd radious based on PCHE INL report,
1412      A:=0.2750;
1413      #CALCULATED
1414      Delta_x := SectionL/(NG-1);
1415      InitRho := InitP*Mw/(R_c*Temp);
1416      InitH := h_form + Cp/Mw*(Temp-T_stand);
1417      InitRho_e := InitRho*InitH - InitP;
1418      Vol := A*Delta_x;
1419    INTERMEDIATE
1420      DIMENSION intRe(NG)
1421      DIMENSION intfArray(NG)
1422      U:= HT.U;
1423      intEps:= r0*1e-9;
1424      intRe:=F*r0*2/A/muVisc;
1425      intfArray:=1/(-2*LOG10(intEps/(3.7*r0) + 2.51/intRe*(1.14 -2*LOG10(intEps/r0 + 21.25/(intRe^0.9)))))^2/4;
1426    EQUATION
1427      # Mass Balance
1428      Delta_x*A*$rho(1) = F_In - F(1);
1429      FOR I:= 2 TO NG DO
1430        Delta_x*A*$rho(I) = (F(I-1) - F(I));
1431      END # for
1432      # Momentum Balance
```

```
    Delta_x/2*$F_In =  -(P(1)*A -P_In*A)
        +(   F_In   )^2  /     (rho(1)*A)
        -(    ( F_In + F(1) )/2    )^2  /     (rho(1)*A)
        - (2*intfArray(1)/(2*r0))  * ( ABS(F_In)*F_In )   /   (rho(1)*A)  *   (Delta_x)/2;
    Delta_x    * $F(1) =
        -(    P(2)*A     -         P(1)*A    )
        +(   (    F_In  +  F(1)    )/2   )^2     /    (  rho(1)*A  )
        -(   (     F(2) +  F(1)    )/2   )^2    /     (   rho(2)*A   )
        - 2*intfArray(1)/( 2*r0 )    *   (  ABS(F(1))*F(1)   )/  (  rho(1)*A  )  *  Delta_x/2
        - 2*intfArray(2)/( 2*r0 )    *   (  ABS(F(1))*F(1)  )/  (  rho(2)*A  )  *  Delta_x/2;
    FOR I:=2 TO NG-1 DO
      Delta_x    * $F(I) =
        -(    P(I+1)*A     -         P(I)*A    )
        +(   (    F(I-1)  +  F(I)    )/2   )^2     /    (  rho(I)*A  )
        -(   (     F(I+1) +  F(I)    )/2   )^2    /     (   rho(I+1)*A   )
        - 2*intfArray(I)/( 2*r0 )    *   (  ABS(F(I))*F(I)   )/  (  rho(I)*A  )  *  Delta_x/2
        - 2*intfArray(I+1)/( 2*r0 )    *   (  ABS(F(I))*F(I)  )/  (  rho(I+1)*A  )  *  Delta_x/2;
    END # for
    Delta_x /2    * $F(NG) =
      -(    P_Out*A    -     P(NG)*A    )
      +(   (    F(NG-1)  +  F(NG)    )/2   )^2     /    (  rho(NG)*A  )
      -(   F(NG)    )^2    /     (   rho(NG)*A   )
      - 2*intfArray(NG)/( 2*r0 )    *   (  ABS(F(NG))*F(NG)   )  /  (  rho(NG)*A  )  *  Delta_x/2;
    # Energy Balance
      1e-7*$psi(1)*Vol = 1e-7*(F_In*h0 - F(1)*h(1)
        +4*U/(2*r0)*(TExtrnl(1)-T(1 ))*Vol);
    FOR I:=(2) TO (NG) DO
        1e-7*$psi(I)*Vol = 1e-7*(F(I-1)*h(I-1) - F(I)*h(I)
          +4*U/(2*r0)*(TExtrnl(I)-T(I))*Vol);  # the last term comes from heat in/volume to heat in Work: Pipe
                Simulation
    END # for

    # Algebraic Equations
    FOR I:=1 TO NG DO
        0 = 1e-7*(psi(I) - (rho(I)*h(I) - P(I)));
    END # for

    # 2- Thermodynamical Part
    FOR I:= 1 TO NG DO
        0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
    END # for
    0 = (h0 - (h_form + Cp/Mw*(T0 - T_stand)));
    # Equation of State
    FOR I:=1 TO NG DO
        0=1e-3*(P(I)*Mw - rho(I)*R_c*T(I));
    END # for

    #Equations to assign variables for heat transfer coefficients
    HT.rhoPipeHe = sigma(rho(2:NG))/NG;
    HT.vPipe = sigma(F(2:NG)/rho(2:NG)/A)/NG;
END
######################################################################
```

MODELS_NUCLEARREACTORPROPERTIES.JAC

```
######################################################################
MODEL ThermalHyd3
######################################################################
  PARAMETER
    # SET here
    Pi AS REAL
    NGHor AS INTEGER
    NGVer AS INTEGER # Number of grid points in the vertical direction
    Cp AS REAL
    VoidF AS REAL
    DenGraphite AS REAL
    PDropRCore0 AS REAL #
    LNode AS ARRAY(NGVer) OF REAL # |
    R AS ARRAY(NGHor) OF REAL
    RPF AS ARRAY(NGHor) OF REAL # radial peaking factors
    APF AS ARRAY(NGVer-2) OF REAL # axial peaking factors
    WF AS ARRAY(NGHor) OF REAL
    # CALCULATED
    LVer AS ARRAY(NGVer-1) OF REAL # | delta Z
    LHor AS ARRAY(NGHor-1) OF REAL # - delta R
    AVer AS ARRAY(NGHor) OF REAL # -
    AHor AS ARRAY(NGVer,NGHor-1) OF REAL# |
    # EXTERNAL
    WPrim0 AS REAL
  UNIT
    HTCore AS ARRAY(NGVer-2,NGHor-2) OF HEEFM2
    CPReflTop AS ARRAY(1,NGHor) OF CPRF
    CPReflSide AS ARRAY(NGVer-2, 2) OF CPRF
```

```
29      CPReflDown AS ARRAY(1,NGHor) OF CPRF
30      CPFuel AS ARRAY(NGVer-2,NGHor-2) OF CPFuel
31    VARIABLE
32      # CALCULATED
33      T_FuelAvg AS Temperature
34      TCoreOut AS Temperature # temperature of gas
35      PCoreOut AS Pressure ### check units
36      PDropRCore AS RealVar
37      TSolid AS ARRAY(NGVer,NGHor) OF Density
38      TGas AS ARRAY(NGVer-1,NGHor-2) OF Temperature
39      # EXTERNAL
40      WPrim AS MassFlowRate
41      QFission AS Power #
42      TCoreIn AS Temperature
43      PCoreIn AS Pressure
44    SET
45      # SET
46      Pi:= 4*ATAN(1);
47      NGHor:= 8;  NGVer:= 12;
48      Cp:=5193.0; #L1095
49      VoidF:= 0.39; #L1102
50      DenGraphite:=1394.8;   #kg/m^3
51      PDropRCore0:= 0.02; # L1171
52      LNode(1):= 1.15; # LUpperRefl L1124
53      LNode(2:NGVer-1):= 0.79309; #LCoreNode L1126
54      LNode(NGVer):= 1.15; #LLowerRefl L1124
55      R:=[0.8239,1.0726,1.2735,1.4469,1.6014,1.7425,2.3878,2.8926];
56      RPF:=[0.0,0.218,0.207,0.195,0.19,0.19,0.0,0.0];     # radial peaking factors, 5 nodes
57      APF:=[0.025,0.034,0.053,0.071,0.091,0.121,0.148,0.171,0.162,0.124];     # axial peaking factors, 10 nodes
            excluding top and bottom reflectors
58       WF:=[0.22,0.156,0.156,0.156,0.156,0.156,0.0,0.0];    # mass flow fraction
59       # CALCULATED
60      LVer(1):= (LNode(1)+LNode(2))/2;
61      LVer(2:(NGVer-2)):= LNode(2:(NGVer-2));
62      LVer(NGVer-1):=(LNode(NGVer-1)+LNode(NGVer))/2;
63      FOR J:=1 TO NGHor-1 DO
64        LHor(J):= (R(J+1)-R(J))/2;
65      END # for J
66      AVer(1):= Pi*R(1)^2;
67      FOR J:=2 TO NGHor DO
68        AVer(J):= Pi*(R(J)^2-R(J-1)^2);
69      END # for
70      FOR I:=1 TO NGVer DO
71        AHor(I,):= 2*Pi*R(1:NGHor-1)*LNode(I); # A1_Right_C or A1_Left_C
72      END # for I
73      # CALCULATED UNITS
74      HTCore.VoidF:= VoidF;     # Gas heat transfer parameters
75      FOR I:= 1 TO NGVer-2 DO
76        FOR J:= 1 TO NGHor-2 DO
77          HTCore(I,J).Area:= AVer(J);
78        END # for J
79      END # for I
80      CPFuel.VoidF:= VoidF;     # VoidF in CPFuel
81    INTERMEDIATE
82      DIMENSION QCond(NGVer,NGHor), QFissionRing(NGVer,NGHor), QC(NGVer,NGHor)
83      DIMENSION K(NGVer,NGHor), KVer(NGVer-1,NGHor), KHor(NGVer,NGHor-1)
84      DIMENSION WCore(NGHor), EXPP(NGVer-2,NGHor-2), HeatCap(NGVer,NGHor)
85      DIMENSION T_FuelAvgSlice(NGVer-2)
86      DIMENSION  QSlice(NGVer-2)
87      QSlice:= QFission*APF;
88      # K, thermal conductivity for each node
89        # reflectors, from FORTRAN function KRT(T)
90        # (1,1) - (1,8), (12,1)-(12,8)
91        FOR J:= 1 TO NGHor DO
92          K(1,J):= -0.0092*(TSolid(1,J)/200)^5+0.267*(TSolid(1,J)/200)^4 -2.9746*(TSolid(1,J)/200)^3+16.8883*(
                TSolid(1,J)/200)^2
93            -59.4691*(TSolid(1,J)/200)+170.884;
94          K(12,J):= -0.0092*(TSolid(12,J)/200)^5+0.267*(TSolid(12,J)/200)^4-2.9746*(TSolid(12,J)/200)^3+16.8883*(
                TSolid(12,J)/200)^2
95            -59.4691*(TSolid(12,J)/200)+170.884;
96        END # for
97        # (2,7)-(11,8)
98        FOR I:= 2 TO NGVer-1 DO
99          FOR J:= 7 TO NGHor DO
100           K(I,J):=  -0.0092*(TSolid(I,J)/200)^5+0.267*(TSolid(I,J)/200)^4-2.9746*(TSolid(I,J)/200)^3+16.8883*(
                  TSolid(I,J)/200)^2
101           -59.4691*(TSolid(I,J)/200)+170.884;
102         END # for
103       END # for
104       # (2,1)-(11,6) pebbles
105       FOR I:= 2 TO NGVer-1 DO
106         FOR J:= 1 TO 6 DO
107           K(I,J):=0.00011536*(TSolid(I,J)-173.16)^1.6622;
108         END # for J
109       END # for I
110     # KVer, Interface conductivity for intermediate nodes in the vertical direction
111       FOR I:= 1 TO NGVer-1 DO
112         FOR J:= 1 TO NGHor DO
113           KVer(I,J):= (K(I,J)+K(I+1,J))/2;
114         END # for J
115       END # for I
116     # KHor, Interface conductivity for intermediate nodes in the horizontal direction
```

```
117         FOR I:= 1 TO NGVer DO
118           FOR J:= 1 TO NGHor -1 DO
119             KHor(I,J):= (K(I,J)+K(I,J+1))/2;
120           END # for J
121         END # for I
122       # QCond: Heat by conduction
123         # top reflector (1,1) - (1,8)
124           # first ring (1,1)
125           QCond(1,1):=  kHor(1,1)*(TSolid(1,2)-TSolid(1,1))*AHor(1,1)/LHor(1)
126                   + kVer(1,1)*(TSolid(2,1)-TSolid(1,1))*AVer(1)/LVer(1);
127           # middle rings (1,2) - (1,7)
128           FOR J:= 2 TO NGHor-1 DO
129             QCond(1,J):=  kHor(1,J-1)*(TSolid(1,J-1)-TSolid(1,J))*AHor(1,J-1)/LHor(J-1)
130                     + kHor(1,J)*(TSolid(1,J+1)-TSolid(1,J))*AHor(1,J)/LHor(J)
131                     + kVer(1,J)*(TSolid(2,J)-TSolid(1,J))*AVer(J)/LVer(1);
132           END # for J
133           # last ring (1,8)
134           QCond(1,8):=  kHor(1,7)*(TSolid(1,7)-TSolid(1,8))*AHor(1,7)/LHor(7)
135                   + kVer(1,8)*(TSolid(2,8)-TSolid(1,8))*AVer(8)/LVer(1);
136         # lower reflector (12,1) - (12,8)
137           # first ring (12,1)
138           QCond(12,1):=kHor(12,1)*(TSolid(12,2)-TSolid(12,1))*AHor(12,1)/LHor(1)
139                   + kVer(11,1)*(TSolid(11,1)-TSolid(12,1))*AVer(1)/LVer(11);
140           # middle rings (12,2) - (12,7)
141           FOR J:= 2 TO NGHor-1 DO
142             QCond(12,J):= kHor(12,J-1)*(TSolid(12,J-1)-TSolid(12,J))*AHor(12,J-1)/LHor(J-1)
143                     + kHor(12,J)*(TSolid(12,J+1)-TSolid(12,J))*AHor(12,J)/LHor(J)
144                     + kVer(11,J)*(TSolid(11,J)-TSolid(12,J))*AVer(J)/LVer(11);
145           END # for J
146           # last ring (12,8)
147           QCond(12,8):=kHor(12,7)*(TSolid(12,7)-TSolid(12,8))*AHor(12,7)/LHor(7)
148                   + kVer(11,8)*(TSolid(11,8)-TSolid(12,8))*AVer(8)/LVer(11);
149         # center of core (2,2) - (11,7)
150         FOR I := 2 TO NGVer-1 DO
151           FOR J:= 2 TO NGHor - 1 DO
152             QCond(I,J):=  kHor(I,J-1)*(TSolid(I,J-1)-TSolid(I,J))*AHor(I,J-1)/LHor(J-1)
153                     +   kHor(I,J)*(TSolid(I,J+1)-TSolid(I,J))*AHor(I,J)/LHor(J)
154                     + kVer(I-1,J)*(TSolid(I-1,J)-TSolid(I,J))*AVer(J)/LVer(I-1)
155                     + kVer(I,J)*(TSolid(I+1,J)-TSolid(I,J))*AVer(J)/LVer(I);
156           END # for J
157         END # for I
158         # left core column or first ring J = 1, (2,1) - (11,1)
159         FOR I:= 2 TO NGVer-1 DO
160             QCond(I,1):=  kHor(I,1)*(TSolid(I,2)-TSolid(I,1))*AHor(I,1)/LHor(1)
161                     + kVer(I-1,1)*(TSolid(I-1,1)-TSolid(I,1))*AVer(1)/LVer(I-1)
162                     + kVer(I,1)*(TSolid(I+1,1)-TSolid(I,1))*AVer(1)/LVer(I);
163         END # for I
164         # right reflector J = 8, (2,8) - (11,8)
165         FOR I:= 2 TO NGVer-1 DO
166             QCond(I,8):=  kHor(I,7)*(TSolid(I,7)-TSolid(I,8))*AHor(I,7)/LHor(7)
167                     + kVer(I-1,8)*(TSolid(I-1,8)-TSolid(I,8))*AVer(8)/LVer(I-1)
168                     + kVer(I,8)*(TSolid(I+1,8)-TSolid(I,8))*AVer(8)/LVer(I);
169         END
170         # lower reflector
171       # QFissionRing: Heat by fission
172         # the nodes with a real effect
173         FOR I:= 2 TO NGVer-1 DO
174           FOR J:= 2 TO 6 DO
175             QFissionRing(I,J) := QSlice(I-1)*RPF(J);
176           END # for J
177         END # for I
178         # nodes without effect
179         QFissionRing(1,1:8):= 0; # top reflector (1,1) - (1,8)
180         QFissionRing(2:11,1):= 0; # inner graphite pebble (2,1) - (11,1)
181         QFissionRing(2:11,7:8):=0; # side reflectors (2,7) - (11,8)
182         QFissionRing(12,1:8):=0;  # bottom reflector (12,1) - (12,8)
183       # WCore
184       WCore:= WPrim*WF; # this is a vector
185       # Calculation of the intermediate parameters for Convection calculation, L791
186       FOR I:= 1 TO NGVer-2 DO
187         FOR J:= 1 TO NGHor-2 DO
188           EXPP(I,J):= HTCore(I,J).HTCoeffR*AVer(J)*74.048*LNode(I+1)/(WCore(J)*Cp);
189         END # for J
190       END # for I
191       # QC: Q by convection with gas
192         FOR I:=2 TO NGVer-1 DO # excluding top and bottom reflector
193           FOR J:=1 TO NGHor-2 DO # excluding 2 side reflectors
194             QC(I,J):= (TGas(I,J)-TGas(I-1,J))*WCore(J)*Cp;
195           END # for J
196         END # for I
197         QC(1,1:8):= 0; # top reflector (1,1) - (1,8)
198         QC(2:11,7:8):=0; # side reflectors (2,7) - (11,8)
199         QC(12,1:8):=0;  # bottom reflector (12,1) - (12,8)
200       # Calculate HeatCapacity
201         FOR J:= 1 TO NGHor DO
202           HeatCap(1,J):= DenGraphite*LNode(1)*AVer(J)*CPReflTop(1,J).Cp; # top refl
203           HeatCap(12,J):= DenGraphite*LNode(12)*AVer(J)*CPReflDown(1,J).Cp; # lower refl
204         END # for J
205         FOR I:= 2 TO NGVer-1 DO
206           HeatCap(I,7):= DenGraphite*LNode(I)*AVer(7)*CPReflSide(I-1,1).Cp; # side refl
207           HeatCap(I,8):= DenGraphite*LNode(I)*AVer(8)*CPReflSide(I-1,2).Cp; # side refl
208           FOR J:= 1 TO NGHor-2 DO
```

```
209             HeatCap(I,J):= LNode(I)*AVer(J)*CPFuel(I-1,J).Cp; # fuel
210           END # for J
211         END # for I
212       # Calculate Average Fuel Temperature per Slice
213         FOR I:= 2 TO 11 DO
214           T_FuelAvgSlice(I-1):= sigma(AVer(2:NGHor-2)*TSolid(I,2:NGHor-2))/sigma(AVer(2:NGHor-2));
215         END # for I
216     EQUATION
217       T_FuelAvg = sigma(T_FuelAvgSlice)/(NGVer-2);
218       # Temperature of Core and reflectors
219       $TSolid*HeatCap = QCond + QFissionRing - QC; # array equation
220       TGas(1,1:NGHor-2) = TCoreIn;
221       FOR I:= 2 TO NGVer-1 DO
222         FOR J:= 1 TO NGHor -2 DO
223           # Calculate temperature of gas due to convection
224           TGas(I,J)= TGas(I-1,J)+(TSolid(I,J)-TGas(I-1,J))*(1-EXP(-EXPP(I-1,J)));
225         END # for J
226       END # for I
227       # output
228       TCoreOut = sigma(TGas(NGVer-1,)*WF(1:NGHor-2)); ### check multiplication is right
229       PCoreOut = (1-PDropRCore)*PCoreIn;
230       PDropRCore*WPrim0^2 = PDropRCore0*WPrim^2;
231       # Assign the variables for the gas heat transfer coeff calculations
232       FOR I:= 2 TO NGVer-1 DO
233         FOR J:= 1 TO NGHor -2 DO
234           HTCore(I-1,J).Temp=TSolid(I,J);
235           HTCore(I-1,J).W=WCore(J);
236         END # for J
237       END # for I
238       # assign temperature for the CP units
239       # top and lower reflectors
240         CPReflTop(1,1:NGHor).Temp = TSolid(1,1:NGHor);
241         CPReflDown(1,1:NGHor).Temp = TSolid(12,1:NGHor);
242       # side reflectors
243         CPReflSide(1:NGVer-2,1:2).Temp = TSolid(2:NGVer-1,7:8);
244       # cp fuel
245         CPFuel(1:NGVer-2,1:NGHor-2).Temp = TSolid(2:NGVer-1,1:NGHor-2);
246   END # Model
247   ######################################################################
248
249
250   ######################################################################
251   MODEL ReactivityTempSimple
252   ######################################################################
253   PARAMETER
254       alpha AS REAL
255       T_FuelAvg0 AS REAL
256   UNIT
257   VARIABLE
258       T_FuelAvg AS Temperature
259       pho_temp AS Reactivity
260   SET
261       alpha:= -3.5e-5;
262       T_FuelAvg0 := 725.925 + 273.16;
263   INTERMEDIATE
264   EQUATION
265       pho_temp = alpha*(T_FuelAvg - T_FuelAvg0);
266       # the original equation is
267       # pho_temp = -3.5e-5*(TFuelAvg-T_FuelAvg0), with T_FuelAvg = 725.925
268   END
269   ######################################################################
270
271
272   ######################################################################
273   MODEL ReactivityPoison
274   ######################################################################
275   PARAMETER
276       # SET
277       gamma_I AS REAL
278       lambda_I AS REAL
279       gamma_Xe AS REAL
280       lambda_Xe AS REAL
281       sigma_total AS REAL
282       sigma_absorb AS REAL
283       sigma_Xe AS REAL
284       # EXTERNAL
285       phi0 AS REAL
286   UNIT
287   VARIABLE
288       # CALCULATED
289       phi AS PhiConcentration
290       IConc AS IConcentration # fix this
291       Xe AS XeConcentration# fix this
292       pho_poison AS Reactivity
293       # EXTERNAL
294       Q AS Power ### fix this
295   SET
296       gamma_I:=0.06386; # Wang Thesis p. 102
297       lambda_I:=2.87e-5;
298       gamma_Xe:=0.00288;
299       lambda_Xe:=2.09e-5;
300       sigma_total:=0.133; # total fission cross section, macroscopic fission section
```

```
    sigma_Absorb:=0.1487; # total absorbtion cross section
    sigma_Xe:=3e-22; #micro Xe absorbtion cross section
INTERMEDIATE
EQUATION
    0= 1e-13*(phi-Q/(250e6)*phi0); # neutron flux
    $IConc = gamma_I*sigma_total*phi-lambda_I*IConc;
    $Xe = gamma_xe*sigma_total*phi+lambda_I*IConc - (lambda_Xe + sigma_Xe*phi)*Xe;
    pho_poison=-sigma_xe*Xe/sigma_Absorb;
END
#####################################################################


#####################################################################
MODEL Kinetics
#####################################################################
PARAMETER
    #EXTERNAL
    # SET here
    beta AS REAL # total effective delayed-neutron fraction
    Lambda AS REAL # decay constant
    TimeLen AS REAL # prompt-neutron lifetime
UNIT
VARIABLE
    #EXTERNAL
    Pho AS Reactivity
    # CALCULATED
    QFission_K AS RealVar # fission power # this Q = QFull/1e8
    C AS ConcPrecursors # concentration of delayed precursors
SET
    Beta:=5*1e-3; #0.001
    Lambda:=0.0584e0;
    TimeLen:=4.0*1e-4; # 0.0001
INTERMEDIATE
EQUATION
    $QFission_K = (Pho-beta)*QFission_K/timeLen+Lambda*C;
    $C = beta*QFission_K/timeLen -Lambda*C;
END
#####################################################################
```

MODELS_HEATEXCHANGERS.JAC

```
#####################################################################
MODEL HX2
#####################################################################
  PARAMETER
    # SET here
    Pi AS REAL
    CpHX AS REAL # J/(Kg*K)
    rhoHX AS REAL
    k AS REAL # W/(m*K)
    dch AS REAL # diameter gas channels in heat exchanger
    pc AS REAL # horizontal distance between gas channels

    # EXTERNALly set
    Length AS REAL # actual length of HX 'm"
    NGHX AS INTEGER

    # CALCULATED
    tp AS REAL # vertical distance between gas channels
    delta_x AS REAL
    ASection AS REAL # vertical section of the control V of metal in the PCHE
    AEnvelope AS REAL # horizontal area of the channels in the PCHE
    VMetal AS REAL # Volume of the section of metal in the PCHE containing 1 hot and 1 cold channel

  VARIABLE
    TGasHot AS ARRAY(NGHX) OF Temperature
    TGasCold AS ARRAY(NGHX) OF Temperature
    UHot AS HTCoeff # overall heat transfer coefficient in the hot side
    UCold AS HTCoeff # overall heat transfer coefficient in the cold side
    #Differential
    THX AS ARRAY(NGHX) OF Temperature
  SET
    # SET
    Pi := 4*ATAN(1);    CpHX:=500;    rhoHX:=8.03e3;
    k:=22.0;          dch:=1.5e-3;    pc:=2.25e-3;

      #CALCULATED
    tp:= dch * 0.78;
    delta_x:= Length/NGHX;
    ASection:= 2*tp*pc - 2*(dch/2)^2*Pi;  # A of metal surrounding 2 channels, 1 hot and 1 cold
    AEnvelope:= (Pi*dch/2+dch)*delta_x; # Contact Area of 1 channel in Control Volume
    VMetal:= ASection*delta_x;
  EQUATION
```

```
    # zero heat flux out of HX
    VMetal*CpHX*rhoHX*$THX(1)=k*(THX(2)-1*THX(1))/delta_x*ASection
      +UHot*(TGasHot(1)-THX(1))*AEnvelope
      +UCold*(TGasCold(1)-THX(1))*AEnvelope;
    FOR I:=2 TO NGHX-1 DO
      VMetal*CpHX*rhoHX*$THX(I)=k*(THX(I+1)-2*THX(I)+THX(I-1))/delta_x*ASection
      +UHot*(TGasHot(I)-THX(I))*AEnvelope
      +UCold*(TGasCold(I)-THX(I))*AEnvelope;
    END # for
    # zero heat flux out of HX
    VMetal*CpHX*rhoHX*$THX(NGHX)=k*(-1*THX(NGHX)+THX(NGHX-1))/delta_x*ASection
      +UHot*(TGasHot(NGHX)-THX(NGHX))*AEnvelope
      +UCold*(TGasCold(NGHX)-THX(NGHX))*AEnvelope;
END # model HeatExchanger
######################################################################


######################################################################
MODEL PHX
######################################################################
  PARAMETER
      # SET here
    Pi AS REAL
    CpHX AS REAL # J/(Kg*K)
    rhoHX AS REAL
    k AS REAL # W/(m*K)
    dinner AS REAL # inner diameter of tubes 'm'
    douter AS REAL # outer diameter of tubes 's'

    # EXTERNALly set
    Length AS REAL # actual length of HX 'm"
    NGHX AS INTEGER

    # CALCULATED
    delta_x AS REAL
    ASection AS REAL # vertical section of the control V of metal in the PHX
    AEnvelopeInner AS REAL # internal horizontal area of the pipe
    AEnvelopeOuter AS REAL # external horizontal area of the pipe
    VMetal AS REAL # Volume of the section of metal in a triangular unit, it corresponds to half a pipe

  VARIABLE
    TGasHot AS ARRAY(NGHX) OF Temperature
    TGasCold AS ARRAY(NGHX) OF Temperature
    UHot AS HTCoeff # overall heat transfer coefficient in the hot side
    UCold AS HTCoeff # overall heat transfer coefficient in the cold side
    #Differential
    THX AS ARRAY(NGHX) OF Temperature
  SET
    # SET
    Pi := 4*ATAN(1);  CpHX:=500;      rhoHX:=8.03e3;
    k:=22;          dinner:= 0.01;      douter:= 0.013;

      #CALCULATED
    delta_x:= Length/NGHX;
    ASection:= Pi*((douter/2)^2-(dinner/2)^2)/2; # it is divided by 2 because the unit of calculation only
         involves half a pipe
    AEnvelopeInner:=Pi*dinner/2*delta_x ;
    AEnvelopeOuter:= Pi*douter/2*delta_x;
    VMetal:= ASection*delta_x;
  EQUATION
    # zero heat flux out of HX
    VMetal*CpHX*rhoHX*$THX(1)=k*(THX(2)-1*THX(1))/delta_x*ASection
      +UHot*(TGasHot(1)-THX(1))*AEnvelopeOuter
      +UCold*(TGasCold(1)-THX(1))*AEnvelopeInner;
    FOR I:=2 TO NGHX-1 DO
      VMetal*CpHX*rhoHX*$THX(I)=k*(THX(I+1)-2*THX(I)+THX(I-1))/delta_x*ASection
      +UHot*(TGasHot(I)-THX(I))*AEnvelopeOuter
      +UCold*(TGasCold(I)-THX(I))*AEnvelopeInner;
    END # for
    # zero heat flux out of HX
    VMetal*CpHX*rhoHX*$THX(NGHX)=k*(-1*THX(NGHX)+THX(NGHX-1))/delta_x*ASection
      +UHot*(TGasHot(NGHX)-THX(NGHX))*AEnvelopeOuter
      +UCold*(TGasCold(NGHX)-THX(NGHX))*AEnvelopeInner;
END # model HeatExchanger
######################################################################
```

MODELS_HEATTRANSFERLOOP.JAC

```
######################################################################
MODEL HeatTransferLoopFullDynPipesFullDyn
######################################################################
  PARAMETER
    Tw AS REAL
```

```
6       Pi AS REAL
7       InitT AS REAL
8     UNIT
9       HX AS HX2
10      Loop1 AS LoopFullDyn
11      PipeH AS PipeFullDyn
12      HX2 AS PHX
13      PipeC2 AS PipeInnerFullDyn
14    SET
15      # SET here
16      # Plant parameters
17      Pi := 4*ATAN(1);
18      Tw:= 291.15; # setting Loop.Tw and Plant.Tw  2 x
19      PipeH.HT.kHe:= 0.36983;
20      PipeH.muVisc:=4.445E-05;
21      Loop1.muVisc(1):=3.963E-05; #from excel
22      Loop1.muVisc(2):=4.319E-05; # avg from relap
23      Loop1.muVisc(3):=5.094E-05; #from excel
24      Loop1.muVisc(4):=4.291E-05; # avg from relap
25      #CALCULATED
26      Loop1.Tw:=Tw;
27      # HX parameters
28      HX.NGHX := Loop1.SectionSpan(2); # this was set in the simulation before
29      HX.Length := Loop1.SectionL(2);
30      # PipeH
31      PipeH.NG := HX.NGHX;
32      PipeH.SectionL:= HX.Length;
33      PipeH.Temp:= Tw;
34      # HX2
35      HX2.NGHX:= Loop1.SectionSpan(4);
36      HX2.Length:= Loop1.SectionL(4);
37      # PipeC, PHX
38      PipeC2.NG:= HX.NGHX;      PipeC2.SectionL:=HX2.Length;
39      PipeC2.Temp:=InitT;
40      PipeC2.muVisc:=4.135E-05;
41    EQUATION
42      #Loop1
43      FOR I:=1 TO Loop1.SectionEnd(1) DO # 1-SectionEnd(1)
44        Loop1.TExtrnl(I)=Loop1.Tw;
45      END # for
46      FOR I:=1 TO HX.NGHX DO  # SectionEnd(1)+1 - SectionEnd(2)
47        HX.TGasCold(I)=Loop1.T(I+ Loop1.SectionEnd(1));
48        HX.THX(I)=Loop1.TExtrnl(I+Loop1.SectionEnd(1));
49      END # for
50      FOR I:=Loop1.SectionEnd(2) + 1 TO Loop1.SectionEnd(3) DO # SectionEnd(2)+1 - SectionEnd(3)
51        Loop1.TExtrnl(I)=Loop1.Tw;
52      END # for
53      # this section corresponds to the PHX and it is reset in the simulation file # SectionEnd(3)+1 - SectionEnd
          (4)
54      FOR I:=1 TO HX2.NGHX DO # SectionEnd(3) + 1 - SectionEnd(4)
55        HX2.TGasHot(I) = Loop1.T(Loop1.SectionEnd(3) + I);
56        HX2.THX(I) = Loop1.TExtrnl(I + Loop1.SectionEnd(3));
57      END # for
58      #HX
59      FOR I:= 1 TO HX.NGHX DO
60        HX.TGasHot(I) = PipeH.T(PipeH.NG + 1 - I);
61      END #for
62      HX.UHot = PipeH.HT.U;
63      HX.UCold = Loop1.HT2.U;
64      #HX2
65      FOR I:=1  TO HX2.NGHX DO
66        HX2.TGasCold(I) = PipeC2.T(PipeC2.NG + 1 -I);
67      END # for
68      HX2.UHot = Loop1.HT4.U; #20070311
69      HX2.UCold = PipeC2.HT.U;
70      #PipeH
71      FOR I:=1 TO PipeH.NG DO
72        PipeH.TExtrnl(PipeH.NG+1-I) = HX.THX(I); #PipeH.Temp; #
73      END # for
74      #PipeC PHX
75      FOR I:= 1 TO PipeC2.NG DO
76        PipeC2.TExtrnl(PipeC2.NG+1-I)=HX2.THX(I);
77      END
78  END # Model
79  ######################################################################
80
81
82  ######################################################################
83  MODEL HeatTransferLoop0MachPipesFullDyn
84  ######################################################################
85    PARAMETER
86      Tw AS REAL
87      Pi AS REAL
88      InitT AS REAL
89    UNIT
90      HX AS HX2
91      Loop1 AS Loop0Mach
92      PipeH AS PipeFullDyn
93      HX2 AS PHX
94      PipeC2 AS PipeInnerFullDyn
95    SET
96      # SET here
```

```
 97      # Plant parameters
 98      Pi := 4*ATAN(1);
 99      Tw:= 291.15; # setting Loop.Tw and Plant.Tw  2 x
100      PipeH.HT.kHe:= 0.36983;
101      PipeH.muVisc:=4.445E-05;
102      Loop1.muVisc(1):=3.963E-05; #from excel
103      Loop1.muVisc(2):=4.319E-05; # avg from relap
104      Loop1.muVisc(3):=5.094E-05; #from excel
105      Loop1.muVisc(4):=4.291E-05; # avg from relap
106      #CALCULATED
107      Loop1.Tw:=Tw;
108      # HX parameters
109      HX.NGHX := Loop1.SectionSpan(2); # this was set in the simulation before
110      HX.Length := Loop1.SectionL(2);
111      # PipeH
112      PipeH.NG := HX.NGHX;
113      PipeH.SectionL:= HX.Length;
114      PipeH.Temp:= Tw;
115      # HX2
116      HX2.NGHX:= Loop1.SectionSpan(4);
117      HX2.Length:= Loop1.SectionL(4);
118      # PipeC, PHX
119      PipeC2.NG:= HX.NGHX;      PipeC2.SectionL:=HX2.Length;
120      PipeC2.Temp:=InitT;
121      PipeC2.muVisc:=4.135E-05;
122    EQUATION
123      #Loop1
124      FOR I:=1 TO Loop1.SectionEnd(1) DO # 1-SectionEnd(1)
125        Loop1.TExtrnl(I)=Loop1.Tw;
126      END # for
127      FOR I:=1 TO HX.NGHX DO  # SectionEnd(1)+1 - SectionEnd(2)
128        HX.TGasCold(I)=Loop1.T(I+ Loop1.SectionEnd(1));
129        HX.THX(I)=Loop1.TExtrnl(I+Loop1.SectionEnd(1));
130      END # for
131      FOR I:=Loop1.SectionEnd(2) + 1 TO Loop1.SectionEnd(3) DO # SectionEnd(2)+1 - SectionEnd(3)
132        Loop1.TExtrnl(I)=Loop1.Tw;
133      END # for
134      # this section corresponds to the PHX and it is reset in the simulation file # SectionEnd(3)+1 - SectionEnd
             (4)
135      FOR I:=1 TO HX2.NGHX DO # SectionEnd(3) + 1 - SectionEnd(4)
136        HX2.TGasHot(I) = Loop1.T(Loop1.SectionEnd(3) + I);
137        HX2.THX(I) = Loop1.TExtrnl(I + Loop1.SectionEnd(3));
138      END # for
139      #HX
140      FOR I:= 1 TO HX.NGHX DO
141        HX.TGasHot(I) = PipeH.T(PipeH.NG + 1 - I);
142      END #for
143      HX.UHot = PipeH.HT.U;
144      HX.UCold = Loop1.HT2.U;
145      #HX2
146      FOR I:=1  TO HX2.NGHX DO
147        HX2.TGasCold(I) = PipeC2.T(PipeC2.NG + 1 -I);
148      END # for
149      HX2.UHot = Loop1.HT4.U; #20070311
150      HX2.UCold = PipeC2.HT.U;
151      #PipeH
152      FOR I:=1 TO PipeH.NG DO
153        PipeH.TExtrnl(PipeH.NG+1-I) = HX.THX(I); #PipeH.Temp; #
154      END # for
155      #PipeC PHX
156      FOR I:= 1 TO PipeC2.NG DO
157        PipeC2.TExtrnl(PipeC2.NG+1-I)=HX2.THX(I);
158      END
159    END # Model
160  ######################################################################
161
162
163  ######################################################################
164  MODEL HeatTransferLoopQSSPipesFullDyn
165  ######################################################################
166    PARAMETER
167      Tw AS REAL
168      Pi AS REAL
169      InitT AS REAL
170    UNIT
171        HX AS HX2
172        Loop1 AS LoopQSS
173        PipeH AS PipeFullDyn
174      HX2 AS PHX
175      PipeC2 AS PipeInnerFullDyn
176    SET
177      # SET here
178      # Plant parameters
179        Pi := 4*ATAN(1);
180      Tw:= 291.15; # setting Loop.Tw and Plant.Tw  2 x
181      PipeH.HT.kHe:= 0.36983;
182      PipeH.muVisc:=4.445E-05;
183      Loop1.muVisc(1):=3.963E-05; #from excel
184      Loop1.muVisc(2):=4.319E-05; # avg from relap
185      Loop1.muVisc(3):=5.094E-05; #from excel
186      Loop1.muVisc(4):=4.291E-05; # avg from relap
187      #CALCULATED
```

```
188       Loop1.Tw:=Tw;
189       # HX parameters
190       HX.NGHX := Loop1.SectionSpan(2); # this was set in the simulation before
191       HX.Length := Loop1.SectionL(2);
192       # PipeH
193       PipeH.NG := HX.NGHX;
194       PipeH.SectionL:= HX.Length;
195       PipeH.Temp:= Tw;
196       # HX2
197       HX2.NGHX:= Loop1.SectionSpan(4);
198       HX2.Length:= Loop1.SectionL(4);
199       # PipeC, PHX
200       PipeC2.NG:= HX.NGHX;      PipeC2.SectionL:=HX2.Length;
201       PipeC2.Temp:=InitT;
202       PipeC2.muVisc:=4.135E-05;
203     EQUATION
204       #Loop1
205       FOR I:=1 TO Loop1.SectionEnd(1) DO # 1-SectionEnd(1)
206         Loop1.TExtrnl(I)=Loop1.Tw;
207       END # for
208       FOR I:=1 TO HX.NGHX DO  # SectionEnd(1)+1 - SectionEnd(2)
209         HX.TGasCold(I)=Loop1.T(I+ Loop1.SectionEnd(1));
210         HX.THX(I)=Loop1.TExtrnl(I+Loop1.SectionEnd(1));
211       END # for
212       FOR I:=Loop1.SectionEnd(2) + 1 TO Loop1.SectionEnd(3) DO # SectionEnd(2)+1 - SectionEnd(3)
213         Loop1.TExtrnl(I)=Loop1.Tw;
214       END # for
215       # this section corresponds to the PHX and it is reset in the simulation file # SectionEnd(3)+1 - SectionEnd
            (4)
216       FOR I:=1 TO HX2.NGHX DO # SectionEnd(3) + 1 - SectionEnd(4)
217         HX2.TGasHot(I) = Loop1.T(Loop1.SectionEnd(3) + I);
218         HX2.THX(I) = Loop1.TExtrnl(I + Loop1.SectionEnd(3));
219       END # for
220       #HX
221       FOR I:= 1 TO HX.NGHX DO
222         HX.TGasHot(I) = PipeH.T(PipeH.NG + 1 - I);
223       END #for
224       HX.UHot = PipeH.HT.U;
225       HX.UCold = Loop1.HT2.U;
226       #HX2
227       FOR I:=1  TO HX2.NGHX DO
228         HX2.TGasCold(I) = PipeC2.T(PipeC2.NG + 1 -I);
229       END # for
230       HX2.UHot = Loop1.HT4.U; #20070311
231       HX2.UCold = PipeC2.HT.U;
232       #PipeH
233       FOR I:=1 TO PipeH.NG DO
234         PipeH.TExtrnl(PipeH.NG+1-I) = HX.THX(I); #PipeH.Temp; #
235       END # for
236       #PipeC PHX
237       FOR I:= 1 TO PipeC2.NG DO
238         PipeC2.TExtrnl(PipeC2.NG+1-I)=HX2.THX(I);
239       END
240   END # Model
241   ######################################################################
242
243
244   ######################################################################
245   MODEL HeatTransferLoopQSSPipesQSS
246   ######################################################################
247   PARAMETER
248     Tw AS REAL
249     Pi AS REAL
250     InitT AS REAL
251   UNIT
252       HX AS HX2
253       Loop1 AS LoopQSS
254       PipeH AS PipeQSS
255     HX2 AS PHX
256     PipeC2 AS PipeInnerQSS
257   SET
258     # SET here
259     # Plant parameters
260       Pi := 4*ATAN(1);
261     Tw:= 291.15; # setting Loop.Tw and Plant.Tw  2 x
262     PipeH.HT.kHe:= 0.36983;
263     PipeH.muVisc:=4.445E-05;
264     Loop1.muVisc(1):=3.963E-05; #from excel
265     Loop1.muVisc(2):=4.319E-05; # avg from relap
266     Loop1.muVisc(3):=5.094E-05; #from excel
267     Loop1.muVisc(4):=4.291E-05; # avg from relap
268     #CALCULATED
269     Loop1.Tw:=Tw;
270     # HX parameters
271     HX.NGHX := Loop1.SectionSpan(2); # this was set in the simulation before
272     HX.Length := Loop1.SectionL(2);
273     # PipeH
274     PipeH.NG := HX.NGHX;
275     PipeH.SectionL:= HX.Length;
276     PipeH.Temp:= Tw;
277     # HX2
278     HX2.NGHX:= Loop1.SectionSpan(4);
```

```
279     HX2.Length:= Loop1.SectionL(4);
280     # PipeC, PHX
281     PipeC2.NG:= HX.NGHX;      PipeC2.SectionL:=HX2.Length;
282     PipeC2.Temp:=InitT;
283     PipeC2.muVisc:=4.135E-05;
284   EQUATION
285     #Loop1
286     FOR I:=1 TO Loop1.SectionEnd(1) DO # 1-SectionEnd(1)
287       Loop1.TExtrnl(I)=Loop1.Tw;
288     END # for
289     FOR I:=1 TO HX.NGHX DO  # SectionEnd(1)+1 - SectionEnd(2)
290       HX.TGasCold(I)=Loop1.T(I+ Loop1.SectionEnd(1));
291       HX.THX(I)=Loop1.TExtrnl(I+Loop1.SectionEnd(1));
292     END # for
293     FOR I:=Loop1.SectionEnd(2) + 1 TO Loop1.SectionEnd(3) DO # SectionEnd(2)+1 - SectionEnd(3)
294       Loop1.TExtrnl(I)=Loop1.Tw;
295     END # for
296     # this section corresponds to the PHX and it is reset in the simulation file # SectionEnd(3)+1 - SectionEnd(4)
297     FOR I:=1 TO HX2.NGHX DO # SectionEnd(3) + 1 - SectionEnd(4)
298       HX2.TGasHot(I) = Loop1.T(Loop1.SectionEnd(3) + I);
299       HX2.THX(I) = Loop1.TExtrnl(I + Loop1.SectionEnd(3));
300     END # for
301     #HX
302     FOR I:= 1 TO HX.NGHX DO
303       HX.TGasHot(I) = PipeH.T(PipeH.NG + 1 - I);
304     END #for
305     HX.UHot = PipeH.HT.U;
306     HX.UCold = Loop1.HT2.U;
307     #HX2
308     FOR I:=1  TO HX2.NGHX DO
309       HX2.TGasCold(I) = PipeC2.T(PipeC2.NG + 1 -I);
310     END # for
311     HX2.UHot = Loop1.HT4.U; #20070311
312     HX2.UCold = PipeC2.HT.U;
313     #PipeH
314     FOR I:=1 TO PipeH.NG DO
315       PipeH.TExtrnl(PipeH.NG+1-I) = HX.THX(I); #PipeH.Temp; #
316     END # for
317     #PipeC PHX
318     FOR I:= 1 TO PipeC2.NG DO
319       PipeC2.TExtrnl(PipeC2.NG+1-I)=HX2.THX(I);
320     END
321   END # Model
322   ######################################################################
323
324
325   ######################################################################
326   MODEL HeatTransferLoopFullDynNoPipeH
327   ######################################################################
328     PARAMETER
329       Tw AS REAL
330       Pi AS REAL
331       InitT AS REAL
332     UNIT
333         HX AS HX2
334         Loop1 AS LoopFullDyn
335       HX2 AS PHX
336       PipeC2 AS PipeInnerFullDyn
337     SET
338       # SET here
339       # Plant parameters
340       Pi := 4*ATAN(1);
341       Tw:= 291.15; # setting Loop.Tw and Plant.Tw  2 x
342       Loop1.muVisc(1):=3.963E-05; #from excel
343       Loop1.muVisc(2):=4.319E-05; # avg from relap
344       Loop1.muVisc(3):=5.094E-05; #from excel
345       Loop1.muVisc(4):=4.291E-05; # avg from relap
346       #CALCULATED
347       Loop1.Tw:=Tw;
348       # HX parameters
349       HX.NGHX := Loop1.SectionSpan(2); # this was set in the simulation before
350       HX.Length := Loop1.SectionL(2);
351       # HX2
352       HX2.NGHX:= Loop1.SectionSpan(4);
353       HX2.Length:= Loop1.SectionL(4);
354       # PipeC, PHX
355       PipeC2.NG:= HX.NGHX;      PipeC2.SectionL:=HX2.Length;
356       PipeC2.Temp:=InitT;
357       PipeC2.muVisc:=4.135E-05;
358     EQUATION
359       #Loop1
360       FOR I:=1 TO Loop1.SectionEnd(1) DO # 1-SectionEnd(1)
361         Loop1.TExtrnl(I)=Loop1.Tw;
362       END # for
363       FOR I:=1 TO HX.NGHX DO  # SectionEnd(1)+1 - SectionEnd(2)
364         HX.TGasCold(I)=Loop1.T(I+ Loop1.SectionEnd(1));
365         HX.THX(I)=Loop1.TExtrnl(I+Loop1.SectionEnd(1));
366       END # for
367       FOR I:=Loop1.SectionEnd(2) + 1 TO Loop1.SectionEnd(3) DO # SectionEnd(2)+1 - SectionEnd(3)
368         Loop1.TExtrnl(I)=Loop1.Tw;
369       END # for
```

```
370      # this section corresponds to the PHX and it is reset in the simulation file # SectionEnd(3)+1 - SectionEnd
             (4)
371      FOR I:=1 TO HX2.NGHX DO # SectionEnd(3) + 1 - SectionEnd(4)
372        HX2.TGasHot(I) = Loop1.T(Loop1.SectionEnd(3) + I);
373        HX2.THX(I) = Loop1.TExtrnl(I + Loop1.SectionEnd(3));
374      END # for
375      #HX
376      HX.UCold = Loop1.HT2.U;
377      #HX2
378      FOR I:=1  TO HX2.NGHX DO
379        HX2.TGasCold(I) = PipeC2.T(PipeC2.NG + 1 -I);
380      END # for
381      HX2.UHot = Loop1.HT4.U; #20070311
382      HX2.UCold = PipeC2.HT.U;
383      #PipeC PHX
384      FOR I:= 1 TO PipeC2.NG DO
385        PipeC2.TExtrnl(PipeC2.NG+1-I)=HX2.THX(I);
386      END
387  END # Model
388  ######################################################################
389
390
391  ######################################################################
392  MODEL HeatTransferLoop0MachNoPipeH
393  ######################################################################
394    PARAMETER
395      Tw AS REAL
396      Pi AS REAL
397      InitT AS REAL
398    UNIT
399      HX AS HX2
400      Loop1 AS Loop0Mach
401      HX2 AS PHX
402      PipeC2 AS PipeInner0Mach
403    SET
404      # SET here
405      # Plant parameters
406      Pi := 4*ATAN(1);
407      Tw:= 291.15; # setting Loop.Tw and Plant.Tw  2 x
408      Loop1.muVisc(1):=3.963E-05; #from excel
409      Loop1.muVisc(2):=4.319E-05; # avg from relap
410      Loop1.muVisc(3):=5.094E-05; #from excel
411      Loop1.muVisc(4):=4.291E-05; # avg from relap
412      #CALCULATED
413      Loop1.Tw:=Tw;
414      # HX parameters
415      HX.NGHX := Loop1.SectionSpan(2); # this was set in the simulation before
416      HX.Length := Loop1.SectionL(2);
417      # HX2
418      HX2.NGHX:= Loop1.SectionSpan(4);
419      HX2.Length:= Loop1.SectionL(4);
420      # PipeC, PHX
421      PipeC2.NG:= HX.NGHX;      PipeC2.SectionL:=HX2.Length;
422      PipeC2.Temp:=InitT;
423      PipeC2.muVisc:=4.135E-05;
424    EQUATION
425      #Loop1
426      FOR I:=1 TO Loop1.SectionEnd(1) DO # 1-SectionEnd(1)
427        Loop1.TExtrnl(I)=Loop1.Tw;
428      END # for
429      FOR I:=1 TO HX.NGHX DO  # SectionEnd(1)+1 - SectionEnd(2)
430        HX.TGasCold(I)=Loop1.T(I+ Loop1.SectionEnd(1));
431        HX.THX(I)=Loop1.TExtrnl(I+Loop1.SectionEnd(1));
432      END # for
433      FOR I:=Loop1.SectionEnd(2) + 1 TO Loop1.SectionEnd(3) DO # SectionEnd(2)+1 - SectionEnd(3)
434        Loop1.TExtrnl(I)=Loop1.Tw;
435      END # for
436      # this section corresponds to the PHX and it is reset in the simulation file # SectionEnd(3)+1 - SectionEnd
             (4)
437      FOR I:=1 TO HX2.NGHX DO # SectionEnd(3) + 1 - SectionEnd(4)
438        HX2.TGasHot(I) = Loop1.T(Loop1.SectionEnd(3) + I);
439        HX2.THX(I) = Loop1.TExtrnl(I + Loop1.SectionEnd(3));
440      END # for
441      #HX
442      HX.UCold = Loop1.HT2.U;
443      #HX2
444      FOR I:=1  TO HX2.NGHX DO
445        HX2.TGasCold(I) = PipeC2.T(PipeC2.NG + 1 -I);
446      END # for
447      HX2.UHot = Loop1.HT4.U; #20070311
448      HX2.UCold = PipeC2.HT.U;
449      #PipeC PHX
450      FOR I:= 1 TO PipeC2.NG DO
451        PipeC2.TExtrnl(PipeC2.NG+1-I)=HX2.THX(I);
452      END
453  END # Model
454  ######################################################################
455
456
457  ######################################################################
458  MODEL HeatTransferLoopQSSNoPipeH
459  ######################################################################
```

```
460     PARAMETER
461       Tw AS REAL
462       Pi AS REAL
463       InitT AS REAL
464     UNIT
465         HX AS HX2
466       Loop1 AS LoopQSS
467       HX2 AS PHX
468       PipeC2 AS PipeInnerQSS
469     SET
470       # SET here
471       # Plant parameters
472       Pi := 4*ATAN(1);
473       Tw:= 291.15; # setting Loop.Tw and Plant.Tw  2 x
474       Loop1.muVisc(1):=3.963E-05; #from excel
475       Loop1.muVisc(2):=4.319E-05; # avg from relap
476       Loop1.muVisc(3):=5.094E-05; #from excel
477       Loop1.muVisc(4):=4.291E-05; # avg from relap
478       #CALCULATED
479       Loop1.Tw:=Tw;
480       # HX parameters
481       HX.NGHX := Loop1.SectionSpan(2); # this was set in the simulation before
482       HX.Length := Loop1.SectionL(2);
483       # HX2
484       HX2.NGHX:= Loop1.SectionSpan(4);
485       HX2.Length:= Loop1.SectionL(4);
486       # PipeC, PHX
487       PipeC2.NG:= HX.NGHX;      PipeC2.SectionL:=HX2.Length;
488       PipeC2.Temp:=InitT;
489       PipeC2.muVisc:=4.135E-05;
490     EQUATION
491       #Loop1
492       FOR I:=1 TO Loop1.SectionEnd(1) DO # 1-SectionEnd(1)
493         Loop1.TExtrnl(I)=Loop1.Tw;
494       END # for
495       FOR I:=1 TO HX.NGHX DO  # SectionEnd(1)+1 - SectionEnd(2)
496         HX.TGasCold(I)=Loop1.T(I+ Loop1.SectionEnd(1));
497         HX.THX(I)=Loop1.TExtrnl(I+Loop1.SectionEnd(1));
498       END # for
499       FOR I:=Loop1.SectionEnd(2) + 1 TO Loop1.SectionEnd(3) DO # SectionEnd(2)+1 - SectionEnd(3)
500         Loop1.TExtrnl(I)=Loop1.Tw;
501       END # for
502       # this section corresponds to the PHX and it is reset in the simulation file # SectionEnd(3)+1 - SectionEnd
            (4)
503       FOR I:=1 TO HX2.NGHX DO # SectionEnd(3) + 1 - SectionEnd(4)
504         HX2.TGasHot(I) = Loop1.T(Loop1.SectionEnd(3) + I);
505         HX2.THX(I) = Loop1.TExtrnl(I + Loop1.SectionEnd(3));
506       END # for
507       #HX
508       HX.UCold = Loop1.HT2.U;
509       #HX2
510       FOR I:=1  TO HX2.NGHX DO
511         HX2.TGasCold(I) = PipeC2.T(PipeC2.NG + 1 -I);
512       END # for
513       HX2.UHot = Loop1.HT4.U; #20070311
514       HX2.UCold = PipeC2.HT.U;
515       #PipeC PHX
516       FOR I:= 1 TO PipeC2.NG DO
517         PipeC2.TExtrnl(PipeC2.NG+1-I)=HX2.THX(I);
518       END
519   END # Model
520   ######################################################################
```

MODELS_NUCLEARREACTORANDAUXS.JAC

```
1   ######################################################################
2   MODEL PipeRhoConst # oneboundary_tube2
3   ######################################################################
4     PARAMETER
5       #SET here
6       Pi AS REAL
7       R_c AS REAL        #Rc constant "J/(mol*K)
8       Mw AS REAL         #Molecular weight "Kg/mol"
9       f AS REAL          #Friction Coefficient
10      T_stand AS REAL    #Standard Temp "K"
11      Cp AS REAL         #specific heat at constant pressure "J/Kg/K"
12      h_form AS REAL     #specific enthalpy of formation "J/Kg"
13      A AS REAL
14      r0 AS REAL       #diameter of pipe "m"
15      SectionL AS REAL           #length of pipe "m"
16
17      #EXTERNALLY set
18      NG AS INTEGER       #number of control volumes in the grid
```

```
    Temp AS REAL       #Temperature "K"
    InitP AS REAL
    rho AS ARRAY(NG) OF REAL # I am setting rho as an array of parameters to avoid big

    #CALCULATED
    InitRho AS REAL     InitH AS REAL
    InitRho_e AS REAL   Delta_x AS REAL     #length of control volume "m"
    Vol AS REAL
  UNIT
    HT AS HTCoeffGasMetalHX
  VARIABLE
    v AS ARRAY(NG) OF Velocity
    P AS ARRAY(NG+1) OF Pressure
    h as ARRAY(NG) OF SpecificEnergy
    T as ARRAY(NG) OF Temperature
    TExtrnl AS ARRAY(NG) OF Temperature
    #DIFFERENTIAL
    rho_e AS ARRAY(NG) OF Energy
  SET
    #SET
    Pi:=4*ATAN(1);
    R_c := 8.314;     Mw := 4.0E-3;
    T_stand := 298;   Cp := 20.78;        h_form := 0;
    r0 := 9.1652e-4/2 ; # Hyd radious based on PCHE INL report p18 #14
    A:=0.34749156;
    #CALCULATED
    Delta_x := SectionL/(NG-1);
    InitRho := InitP*Mw/(R_c*Temp);
    InitH := h_form + Cp/Mw*(Temp-T_stand);
    InitRho_e := InitRho*InitH - InitP;
    HT.r0:=r0;        HT.t:= (HT.r0*2*0.78*3 - HT.r0)/3; # according to INL, an average of 2*tmax +1*tmin
    Vol := A*Delta_x;
  INTERMEDIATE
    U:= HT.U;
  EQUATION
    # Mass Balance
    FOR I:= 2 TO NG DO
        0 = 1e-7*(rho(I-1)*v(I-1)*A - rho(I)*v(I)*A);
    END # for
    # Momentum Balance
    0 = -(P(2)-P(1))
        - ((2*f/(2*r0))*rho(1)*v(1)^2)*(Delta_x);
    FOR I:=2 TO NG-1 DO
        0 = -(P(I+1)-P(I))
            - ((2*f/(2*r0))*(rho(I)+rho(I+1))/2*v(I)^2)*Delta_x;
    END # for
    0 = -(P(NG+1)-P(NG))
        -((2*f/(2*r0))*rho(NG)*v(NG)^2)*(Delta_x);
    # Energy Balance
    FOR I:=(2) TO (NG) DO
        1e-7*$rho_E(I)*Vol = 1e-7*(rho(I-1)*h(I-1)*v(I-1)*A - rho(I)*h(I)*v(I)*A
          +4*U/(2*r0)*(TExtrnl(I)-T(I))*Vol);
    END # for

    # Algebraic Equations
    FOR I:=1 TO NG DO
        0 = 1e-7*(rho_e(I) - (rho(I)*h(I) - P(I)));
    END # for

    # 2- Thermodynamical Part
    FOR I:= 1 TO NG DO
        0 = 1e-7*(h(I) - (h_form + Cp/Mw*(T(I)-T_stand)));
    END # for

    #Equations to assign variables for heat transfer coefficients
    HT.rhoPipeHe = sigma(rho(2:NG))/NG;
    HT.vPipe = sigma(v(2:NG))/NG;
END
######################################################################


######################################################################
MODEL Comp
######################################################################
  PARAMETER
    # EXTERNAL
    Eff AS REAL # isentropic efficiency
    # EffCir=0.9
    Gamma AS REAL
    # Gamma=1.6667;
    Cp AS REAL
    #  Cp=5193.0d0
  UNIT
  VARIABLE
    # CALCULATED
    T2 AS Temperature # outlet temperature
    P2 AS Pressure # outlet pressure
    PWR AS Power # compressor power
    # EXTERNAL
    PR AS RealVar # pressure ratio  # PRCir = 1.038794831;
    P1 AS Pressure # pressure in
    T1 AS Temperature # temperature in
```

```
111       W AS MassFlowRate # mass flowrate
112     SET
113     INTERMEDIATE
114       Rgas:=2077.22;
115       C1:=(Gamma-1)/Gamma;
116       C2:=PR^C1-1;
117       C3:=1+C2/Eff;
118     EQUATION
119       T2=(T1)*C3;
120       P2=P1*PR;
121       PWR=W*Cp*(T2-T1);
122   END
123   ######################################################################
124
125
126   ######################################################################
127   MODEL NReactor
128   ######################################################################
129     PARAMETER
130         # EXTERNAL
131         WPrim0 AS REAL
132     UNIT
133         THyd AS ThermalHyd3
134         Kinetics AS Kinetics
135         Poison AS ReactivityPoison
136         ReacT AS ReactivityTempSimple
137     VARIABLE
138         TCoreIn AS Temperature # temperature of gas going in the core
139         PCoreIn AS Pressure
140         # CALCULATED
141         PhoTotal AS Reactivity #
142         PhoPoison AS Reactivity #
143         PhoTemp AS Reactivity #
144         PhoControl AS Reactivity
145         PhoExtra1 AS Reactivity
146         wPhoExtra1 AS Nondimensional
147         PhoExtra2 AS Reactivity
148         # INPUTS
149     SET
150         THyd.WPrim0 := WPrim0;
151     EQUATION
152         # Calculating units variables
153         THyd.QFission = Kinetics.QFission_K*1e+8; # L 1696
154         Poison.Q = THyd.QFission;
155         PhoPoison = 0;
156         ReacT.T_FuelAvg = THyd.T_FuelAvg;
157         PhoTemp = ReacT.pho_temp;
158         PhoTotal = PhoTemp + PhoControl + wPhoExtra1*PhoExtra1 + PhoExtra2; #PhoPoison + PhoTemp + PhoControl;
159         Kinetics.Pho = PhoTotal;
160         THyd.TCoreIn = TCoreIn;
161         THyd.PCoreIn = PCoreIn;
162         PhoTemp = -PhoExtra1;
163   END # model
164   ######################################################################
```

MODELS_NUCLEARREACTORLOOP.JAC

```
1   ######################################################################
2   MODEL PlantNR
3   ######################################################################
4     PARAMETER
5         NG AS INTEGER
6         L AS REAL
7         InitT AS REAL
8         InitDeltaP AS REAL
9         MaxDeltaP AS REAL
10
11        T_stand AS REAL
12        Cp AS REAL
13        Mw AS REAL
14        rhomix AS REAL
15        R_c AS REAL
16        Tpcu AS REAL
17        Ppcu AS REAL
18        h_form AS REAL
19        WpcupipeRatio AS REAL
20    UNIT
21        NR AS NReactor
22        PipeH AS PipeRhoConst
23        Comp AS Comp
24    VARIABLE
25        DeltaP AS Pressure
26        Tpipe AS Temperature
```

```
        Tmix AS Temperature
        Wpipe AS MassFlowRate
        Wpcu AS MassFlowRate
        Wmix AS MassFlowRate
        hpcu AS SpecificEnergy
        hpipe AS SpecificEnergy
        hmix AS SpecificEnergy
        Pmix AS Pressure
        Ppipe AS Pressure
    SET
        # SET
        NG:= 40;              L:= 0.98807;
        PipeH.f := 3.546e-2/4;
        PipeH.HT.kHe:= 0.36983;
        Comp.Eff:= 0.9;
        Comp.Gamma:=1.6667;
        Comp.Cp:=5193.0e0;
        NR.WPrim0 := 126.7;
        NR.Poison.phi0 := 1e18;

        #CALCULATED
        PipeH.NG:= NG;
        PipeH.SectionL:=L;
        PipeH.Temp:= InitT;

        T_stand:= 298;
        Cp:= 20.78;
        Mw:= 4.0e-3;
        R_c := 8.314;
        rhomix := 3.93;#4.7; # from page 186 in Wang's thesis; calculated using P and T at the outlet of the IHX.
        Tpcu := 512.0 +273.15;
        Ppcu := 7.00e6;
        h_form:= 5557;
    EQUATION
        PipeH.T(1) = NR.THyd.TCoreOut;
        PipeH.P(1) = NR.THyd.PCoreOut;
        0 = 1e-7*(DeltaP - (Comp.P2 - Comp.P1));

        Tpipe = PipeH.T(PipeH.NG);
        Ppipe = PipeH.P(PipeH.NG +1);
        Wpipe = PipeH.A*PipeH.v(PipeH.NG)*PipeH.rho(PipeH.NG);
        Wpcu = Wpipe*WpcupipeRatio;
        Wmix = Wpipe + Wpcu;
        hpcu = h_form + Cp/Mw*(Tpcu - T_stand);
        hpipe = h_form + Cp/Mw*(Tpipe - T_stand);
        Wmix*hmix =  Wpcu*(hpcu) + Wpipe*hpipe; # Wpcu*hpcu + Wpipe*hpipe;
        Wmix*Pmix = Wpcu*Ppcu + Wpipe*Ppipe;
        hmix = h_form + Cp/Mw*(Tmix - T_stand);

        Comp.T1 = Tmix;  #PipeH.T(PipeH.NG);
        Comp.P1 = Ppipe*1/(WpcupipeRatio + 1)  + Ppcu*WpcupipeRatio/(WpcupipeRatio + 1); #PipeH.P(PipeH.NG+1);
        Comp.W = PipeH.A*PipeH.v(PipeH.NG)*PipeH.rho(PipeH.NG)*(WpcupipeRatio + 1);

        NR.TCoreIn = Comp.T2;
        NR.PCoreIn = Comp.P2;
        NR.THyd.WPrim = Comp.W;
END # model
######################################################################
```

HTLFULLDYN_SS_IN_INI.INITIAL

```
# Values for differential variables in computation HTLFULLDYN_SS
# Saved at time 1000
PLANT.HX.THX(1) = 879.893 ; # 1
PLANT.HX.THX(2) = 892.184 ; # 2
PLANT.HX.THX(3) = 904.161 ; # 3
PLANT.HX.THX(4) = 915.766 ; # 4
PLANT.HX.THX(5) = 927.009 ; # 5
PLANT.HX.THX(6) = 937.904 ; # 6
PLANT.HX.THX(7) = 948.459 ; # 7
PLANT.HX.THX(8) = 958.687 ; # 8
PLANT.HX.THX(9) = 968.596 ; # 9
PLANT.HX.THX(10) = 978.197 ; # 10
PLANT.HX.THX(11) = 987.5 ; # 11
PLANT.HX.THX(12) = 996.513 ; # 12
PLANT.HX.THX(13) = 1005.25 ; # 13
PLANT.HX.THX(14) = 1013.71 ; # 14
PLANT.HX.THX(15) = 1021.91 ; # 15
PLANT.HX.THX(16) = 1029.85 ; # 16
PLANT.HX.THX(17) = 1037.55 ; # 17
PLANT.HX.THX(18) = 1045 ; # 18
PLANT.HX.THX(19) = 1052.23 ; # 19
PLANT.HX.THX(20) = 1059.23 ; # 20
```

```
PLANT.HX.THX(21) = 1066.01 ; # 21
PLANT.HX.THX(22) = 1072.58 ; # 22
PLANT.HX.THX(23) = 1078.95 ; # 23
PLANT.HX.THX(24) = 1085.12 ; # 24
PLANT.HX.THX(25) = 1091.1 ; # 25
PLANT.HX.THX(26) = 1096.89 ; # 26
PLANT.HX.THX(27) = 1102.5 ; # 27
PLANT.HX.THX(28) = 1107.94 ; # 28
PLANT.HX.THX(29) = 1113.21 ; # 29
PLANT.HX.THX(30) = 1118.31 ; # 30
PLANT.HX.THX(31) = 1123.26 ; # 31
PLANT.HX.THX(32) = 1128.05 ; # 32
PLANT.HX.THX(33) = 1132.69 ; # 33
PLANT.HX.THX(34) = 1137.19 ; # 34
PLANT.HX.THX(35) = 1141.55 ; # 35
PLANT.HX.THX(36) = 1145.77 ; # 36
PLANT.HX.THX(37) = 1149.86 ; # 37
PLANT.HX.THX(38) = 1153.83 ; # 38
PLANT.HX.THX(39) = 1157.67 ; # 39
PLANT.HX.THX(40) = 1161.37 ; # 40
PLANT.LOOP1.RHO(1) = 1.15317 ; # 41
PLANT.LOOP1.RHO(2) = 1.15067 ; # 42
PLANT.LOOP1.RHO(3) = 1.14817 ; # 43
PLANT.LOOP1.RHO(4) = 1.14565 ; # 44
PLANT.LOOP1.RHO(5) = 1.14313 ; # 45
PLANT.LOOP1.RHO(6) = 1.14059 ; # 46
PLANT.LOOP1.RHO(7) = 1.13804 ; # 47
PLANT.LOOP1.RHO(8) = 1.13549 ; # 48
PLANT.LOOP1.RHO(9) = 1.13292 ; # 49
PLANT.LOOP1.RHO(10) = 1.13034 ; # 50
PLANT.LOOP1.RHO(11) = 1.12115 ; # 51
PLANT.LOOP1.RHO(12) = 1.1026 ; # 52
PLANT.LOOP1.RHO(13) = 1.08477 ; # 53
PLANT.LOOP1.RHO(14) = 1.06791 ; # 54
PLANT.LOOP1.RHO(15) = 1.05196 ; # 55
PLANT.LOOP1.RHO(16) = 1.03683 ; # 56
PLANT.LOOP1.RHO(17) = 1.02247 ; # 57
PLANT.LOOP1.RHO(18) = 1.00882 ; # 58
PLANT.LOOP1.RHO(19) = 0.99583 ; # 59
PLANT.LOOP1.RHO(20) = 0.983453 ; # 60
PLANT.LOOP1.RHO(21) = 0.971645 ; # 61
PLANT.LOOP1.RHO(22) = 0.960369 ; # 62
PLANT.LOOP1.RHO(23) = 0.949588 ; # 63
PLANT.LOOP1.RHO(24) = 0.939271 ; # 64
PLANT.LOOP1.RHO(25) = 0.929389 ; # 65
PLANT.LOOP1.RHO(26) = 0.919912 ; # 66
PLANT.LOOP1.RHO(27) = 0.910817 ; # 67
PLANT.LOOP1.RHO(28) = 0.90208 ; # 68
PLANT.LOOP1.RHO(29) = 0.893679 ; # 69
PLANT.LOOP1.RHO(30) = 0.885595 ; # 70
PLANT.LOOP1.RHO(31) = 0.87781 ; # 71
PLANT.LOOP1.RHO(32) = 0.870305 ; # 72
PLANT.LOOP1.RHO(33) = 0.863066 ; # 73
PLANT.LOOP1.RHO(34) = 0.856077 ; # 74
PLANT.LOOP1.RHO(35) = 0.849325 ; # 75
PLANT.LOOP1.RHO(36) = 0.842796 ; # 76
PLANT.LOOP1.RHO(37) = 0.83648 ; # 77
PLANT.LOOP1.RHO(38) = 0.830363 ; # 78
PLANT.LOOP1.RHO(39) = 0.824437 ; # 79
PLANT.LOOP1.RHO(40) = 0.818691 ; # 80
PLANT.LOOP1.RHO(41) = 0.813116 ; # 81
PLANT.LOOP1.RHO(42) = 0.807703 ; # 82
PLANT.LOOP1.RHO(43) = 0.802444 ; # 83
PLANT.LOOP1.RHO(44) = 0.797332 ; # 84
PLANT.LOOP1.RHO(45) = 0.792358 ; # 85
PLANT.LOOP1.RHO(46) = 0.787518 ; # 86
PLANT.LOOP1.RHO(47) = 0.782803 ; # 87
PLANT.LOOP1.RHO(48) = 0.778207 ; # 88
PLANT.LOOP1.RHO(49) = 0.773726 ; # 89
PLANT.LOOP1.RHO(50) = 0.769358 ; # 90
PLANT.LOOP1.RHO(51) = 0.757959 ; # 91
PLANT.LOOP1.RHO(52) = 0.756103 ; # 92
PLANT.LOOP1.RHO(53) = 0.754256 ; # 93
PLANT.LOOP1.RHO(54) = 0.752399 ; # 94
PLANT.LOOP1.RHO(55) = 0.750533 ; # 95
PLANT.LOOP1.RHO(56) = 0.748658 ; # 96
PLANT.LOOP1.RHO(57) = 0.746773 ; # 97
PLANT.LOOP1.RHO(58) = 0.744879 ; # 98
PLANT.LOOP1.RHO(59) = 0.742974 ; # 99
PLANT.LOOP1.RHO(60) = 0.74106 ; # 100
PLANT.LOOP1.RHO(61) = 0.750585 ; # 101
PLANT.LOOP1.RHO(62) = 0.751689 ; # 102
PLANT.LOOP1.RHO(63) = 0.752856 ; # 103
PLANT.LOOP1.RHO(64) = 0.754135 ; # 104
PLANT.LOOP1.RHO(65) = 0.755533 ; # 105
PLANT.LOOP1.RHO(66) = 0.757058 ; # 106
PLANT.LOOP1.RHO(67) = 0.75872 ; # 107
PLANT.LOOP1.RHO(68) = 0.760528 ; # 108
PLANT.LOOP1.RHO(69) = 0.762492 ; # 109
PLANT.LOOP1.RHO(70) = 0.764623 ; # 110
PLANT.LOOP1.RHO(71) = 0.766934 ; # 111
PLANT.LOOP1.RHO(72) = 0.769438 ; # 112
```

```
PLANT.LOOP1.RHO(73) = 0.772149 ; # 113
PLANT.LOOP1.RHO(74) = 0.775082 ; # 114
PLANT.LOOP1.RHO(75) = 0.778256 ; # 115
PLANT.LOOP1.RHO(76) = 0.781688 ; # 116
PLANT.LOOP1.RHO(77) = 0.785399 ; # 117
PLANT.LOOP1.RHO(78) = 0.789411 ; # 118
PLANT.LOOP1.RHO(79) = 0.793749 ; # 119
PLANT.LOOP1.RHO(80) = 0.79844 ; # 120
PLANT.LOOP1.RHO(81) = 0.803515 ; # 121
PLANT.LOOP1.RHO(82) = 0.809005 ; # 122
PLANT.LOOP1.RHO(83) = 0.814949 ; # 123
PLANT.LOOP1.RHO(84) = 0.821387 ; # 124
PLANT.LOOP1.RHO(85) = 0.828364 ; # 125
PLANT.LOOP1.RHO(86) = 0.835933 ; # 126
PLANT.LOOP1.RHO(87) = 0.844151 ; # 127
PLANT.LOOP1.RHO(88) = 0.853082 ; # 128
PLANT.LOOP1.RHO(89) = 0.8628 ; # 129
PLANT.LOOP1.RHO(90) = 0.873389 ; # 130
PLANT.LOOP1.RHO(91) = 0.884944 ; # 131
PLANT.LOOP1.RHO(92) = 0.897573 ; # 132
PLANT.LOOP1.RHO(93) = 0.911403 ; # 133
PLANT.LOOP1.RHO(94) = 0.926579 ; # 134
PLANT.LOOP1.RHO(95) = 0.94327 ; # 135
PLANT.LOOP1.RHO(96) = 0.961674 ; # 136
PLANT.LOOP1.RHO(97) = 0.982026 ; # 137
PLANT.LOOP1.RHO(98) = 1.0046 ; # 138
PLANT.LOOP1.RHO(99) = 1.02974 ; # 139
PLANT.LOOP1.RHO(100) = 1.05783 ; # 140
PLANT.LOOP1.PSI(1) = 1.23127e+006 ; # 141
PLANT.LOOP1.PSI(2) = 1.22711e+006 ; # 142
PLANT.LOOP1.PSI(3) = 1.22294e+006 ; # 143
PLANT.LOOP1.PSI(4) = 1.21877e+006 ; # 144
PLANT.LOOP1.PSI(5) = 1.2146e+006 ; # 145
PLANT.LOOP1.PSI(6) = 1.21042e+006 ; # 146
PLANT.LOOP1.PSI(7) = 1.20624e+006 ; # 147
PLANT.LOOP1.PSI(8) = 1.20206e+006 ; # 148
PLANT.LOOP1.PSI(9) = 1.19787e+006 ; # 149
PLANT.LOOP1.PSI(10) = 1.19368e+006 ; # 150
PLANT.LOOP1.PSI(11) = 1.23208e+006 ; # 151
PLANT.LOOP1.PSI(12) = 1.25743e+006 ; # 152
PLANT.LOOP1.PSI(13) = 1.28069e+006 ; # 153
PLANT.LOOP1.PSI(14) = 1.30237e+006 ; # 154
PLANT.LOOP1.PSI(15) = 1.32259e+006 ; # 155
PLANT.LOOP1.PSI(16) = 1.34148e+006 ; # 156
PLANT.LOOP1.PSI(17) = 1.35911e+006 ; # 157
PLANT.LOOP1.PSI(18) = 1.37558e+006 ; # 158
PLANT.LOOP1.PSI(19) = 1.39098e+006 ; # 159
PLANT.LOOP1.PSI(20) = 1.40537e+006 ; # 160
PLANT.LOOP1.PSI(21) = 1.41883e+006 ; # 161
PLANT.LOOP1.PSI(22) = 1.4314e+006 ; # 162
PLANT.LOOP1.PSI(23) = 1.44316e+006 ; # 163
PLANT.LOOP1.PSI(24) = 1.45415e+006 ; # 164
PLANT.LOOP1.PSI(25) = 1.46441e+006 ; # 165
PLANT.LOOP1.PSI(26) = 1.474e+006 ; # 166
PLANT.LOOP1.PSI(27) = 1.48294e+006 ; # 167
PLANT.LOOP1.PSI(28) = 1.49129e+006 ; # 168
PLANT.LOOP1.PSI(29) = 1.49907e+006 ; # 169
PLANT.LOOP1.PSI(30) = 1.50631e+006 ; # 170
PLANT.LOOP1.PSI(31) = 1.51304e+006 ; # 171
PLANT.LOOP1.PSI(32) = 1.5193e+006 ; # 172
PLANT.LOOP1.PSI(33) = 1.5251e+006 ; # 173
PLANT.LOOP1.PSI(34) = 1.53047e+006 ; # 174
PLANT.LOOP1.PSI(35) = 1.53543e+006 ; # 175
PLANT.LOOP1.PSI(36) = 1.54001e+006 ; # 176
PLANT.LOOP1.PSI(37) = 1.54421e+006 ; # 177
PLANT.LOOP1.PSI(38) = 1.54807e+006 ; # 178
PLANT.LOOP1.PSI(39) = 1.5516e+006 ; # 179
PLANT.LOOP1.PSI(40) = 1.5548e+006 ; # 180
PLANT.LOOP1.PSI(41) = 1.55771e+006 ; # 181
PLANT.LOOP1.PSI(42) = 1.56032e+006 ; # 182
PLANT.LOOP1.PSI(43) = 1.56266e+006 ; # 183
PLANT.LOOP1.PSI(44) = 1.56474e+006 ; # 184
PLANT.LOOP1.PSI(45) = 1.56657e+006 ; # 185
PLANT.LOOP1.PSI(46) = 1.56816e+006 ; # 186
PLANT.LOOP1.PSI(47) = 1.56951e+006 ; # 187
PLANT.LOOP1.PSI(48) = 1.57065e+006 ; # 188
PLANT.LOOP1.PSI(49) = 1.57158e+006 ; # 189
PLANT.LOOP1.PSI(50) = 1.5723e+006 ; # 190
PLANT.LOOP1.PSI(51) = 1.54707e+006 ; # 191
PLANT.LOOP1.PSI(52) = 1.54135e+006 ; # 192
PLANT.LOOP1.PSI(53) = 1.53567e+006 ; # 193
PLANT.LOOP1.PSI(54) = 1.52997e+006 ; # 194
PLANT.LOOP1.PSI(55) = 1.52427e+006 ; # 195
PLANT.LOOP1.PSI(56) = 1.51856e+006 ; # 196
PLANT.LOOP1.PSI(57) = 1.51284e+006 ; # 197
PLANT.LOOP1.PSI(58) = 1.50712e+006 ; # 198
PLANT.LOOP1.PSI(59) = 1.50138e+006 ; # 199
PLANT.LOOP1.PSI(60) = 1.49564e+006 ; # 200
PLANT.LOOP1.PSI(61) = 1.50921e+006 ; # 201
PLANT.LOOP1.PSI(62) = 1.50544e+006 ; # 202
PLANT.LOOP1.PSI(63) = 1.50142e+006 ; # 203
PLANT.LOOP1.PSI(64) = 1.49724e+006 ; # 204
```

```
PLANT.LOOP1.PSI(65) = 1.49288e+006 ; # 205
PLANT.LOOP1.PSI(66) = 1.48832e+006 ; # 206
PLANT.LOOP1.PSI(67) = 1.48357e+006 ; # 207
PLANT.LOOP1.PSI(68) = 1.47859e+006 ; # 208
PLANT.LOOP1.PSI(69) = 1.47338e+006 ; # 209
PLANT.LOOP1.PSI(70) = 1.46791e+006 ; # 210
PLANT.LOOP1.PSI(71) = 1.46218e+006 ; # 211
PLANT.LOOP1.PSI(72) = 1.45616e+006 ; # 212
PLANT.LOOP1.PSI(73) = 1.44982e+006 ; # 213
PLANT.LOOP1.PSI(74) = 1.44315e+006 ; # 214
PLANT.LOOP1.PSI(75) = 1.43612e+006 ; # 215
PLANT.LOOP1.PSI(76) = 1.4287e+006 ; # 216
PLANT.LOOP1.PSI(77) = 1.42087e+006 ; # 217
PLANT.LOOP1.PSI(78) = 1.41258e+006 ; # 218
PLANT.LOOP1.PSI(79) = 1.40379e+006 ; # 219
PLANT.LOOP1.PSI(80) = 1.39448e+006 ; # 220
PLANT.LOOP1.PSI(81) = 1.38459e+006 ; # 221
PLANT.LOOP1.PSI(82) = 1.37407e+006 ; # 222
PLANT.LOOP1.PSI(83) = 1.36287e+006 ; # 223
PLANT.LOOP1.PSI(84) = 1.35093e+006 ; # 224
PLANT.LOOP1.PSI(85) = 1.33817e+006 ; # 225
PLANT.LOOP1.PSI(86) = 1.32451e+006 ; # 226
PLANT.LOOP1.PSI(87) = 1.30988e+006 ; # 227
PLANT.LOOP1.PSI(88) = 1.29416e+006 ; # 228
PLANT.LOOP1.PSI(89) = 1.27725e+006 ; # 229
PLANT.LOOP1.PSI(90) = 1.25903e+006 ; # 230
PLANT.LOOP1.PSI(91) = 1.23934e+006 ; # 231
PLANT.LOOP1.PSI(92) = 1.21801e+006 ; # 232
PLANT.LOOP1.PSI(93) = 1.19486e+006 ; # 233
PLANT.LOOP1.PSI(94) = 1.16967e+006 ; # 234
PLANT.LOOP1.PSI(95) = 1.14217e+006 ; # 235
PLANT.LOOP1.PSI(96) = 1.11205e+006 ; # 236
PLANT.LOOP1.PSI(97) = 1.07896e+006 ; # 237
PLANT.LOOP1.PSI(98) = 1.04248e+006 ; # 238
PLANT.LOOP1.PSI(99) = 1.00208e+006 ; # 239
PLANT.LOOP1.PSI(100) = 957155 ; # 240
PLANT.LOOP1.F(1) = 26.308 ; # 241
PLANT.LOOP1.F(2) = 26.308 ; # 242
PLANT.LOOP1.F(3) = 26.308 ; # 243
PLANT.LOOP1.F(4) = 26.308 ; # 244
PLANT.LOOP1.F(5) = 26.308 ; # 245
PLANT.LOOP1.F(6) = 26.308 ; # 246
PLANT.LOOP1.F(7) = 26.308 ; # 247
PLANT.LOOP1.F(8) = 26.308 ; # 248
PLANT.LOOP1.F(9) = 26.308 ; # 249
PLANT.LOOP1.F(10) = 26.308 ; # 250
PLANT.LOOP1.F(11) = 26.308 ; # 251
PLANT.LOOP1.F(12) = 26.308 ; # 252
PLANT.LOOP1.F(13) = 26.308 ; # 253
PLANT.LOOP1.F(14) = 26.308 ; # 254
PLANT.LOOP1.F(15) = 26.308 ; # 255
PLANT.LOOP1.F(16) = 26.308 ; # 256
PLANT.LOOP1.F(17) = 26.308 ; # 257
PLANT.LOOP1.F(18) = 26.308 ; # 258
PLANT.LOOP1.F(19) = 26.308 ; # 259
PLANT.LOOP1.F(20) = 26.308 ; # 260
PLANT.LOOP1.F(21) = 26.308 ; # 261
PLANT.LOOP1.F(22) = 26.308 ; # 262
PLANT.LOOP1.F(23) = 26.308 ; # 263
PLANT.LOOP1.F(24) = 26.308 ; # 264
PLANT.LOOP1.F(25) = 26.308 ; # 265
PLANT.LOOP1.F(26) = 26.308 ; # 266
PLANT.LOOP1.F(27) = 26.308 ; # 267
PLANT.LOOP1.F(28) = 26.308 ; # 268
PLANT.LOOP1.F(29) = 26.308 ; # 269
PLANT.LOOP1.F(30) = 26.308 ; # 270
PLANT.LOOP1.F(31) = 26.308 ; # 271
PLANT.LOOP1.F(32) = 26.308 ; # 272
PLANT.LOOP1.F(33) = 26.308 ; # 273
PLANT.LOOP1.F(34) = 26.308 ; # 274
PLANT.LOOP1.F(35) = 26.308 ; # 275
PLANT.LOOP1.F(36) = 26.308 ; # 276
PLANT.LOOP1.F(37) = 26.308 ; # 277
PLANT.LOOP1.F(38) = 26.308 ; # 278
PLANT.LOOP1.F(39) = 26.308 ; # 279
PLANT.LOOP1.F(40) = 26.308 ; # 280
PLANT.LOOP1.F(41) = 26.308 ; # 281
PLANT.LOOP1.F(42) = 26.308 ; # 282
PLANT.LOOP1.F(43) = 26.308 ; # 283
PLANT.LOOP1.F(44) = 26.308 ; # 284
PLANT.LOOP1.F(45) = 26.308 ; # 285
PLANT.LOOP1.F(46) = 26.308 ; # 286
PLANT.LOOP1.F(47) = 26.308 ; # 287
PLANT.LOOP1.F(48) = 26.308 ; # 288
PLANT.LOOP1.F(49) = 26.308 ; # 289
PLANT.LOOP1.F(50) = 26.308 ; # 290
PLANT.LOOP1.F(51) = 26.308 ; # 291
PLANT.LOOP1.F(52) = 26.308 ; # 292
PLANT.LOOP1.F(53) = 26.308 ; # 293
PLANT.LOOP1.F(54) = 26.308 ; # 294
PLANT.LOOP1.F(55) = 26.308 ; # 295
PLANT.LOOP1.F(56) = 26.308 ; # 296
```

```
PLANT.LOOP1.F(57) = 26.308 ; # 297
PLANT.LOOP1.F(58) = 26.308 ; # 298
PLANT.LOOP1.F(59) = 26.308 ; # 299
PLANT.LOOP1.F(60) = 26.308 ; # 300
PLANT.LOOP1.F(61) = 26.308 ; # 301
PLANT.LOOP1.F(62) = 26.308 ; # 302
PLANT.LOOP1.F(63) = 26.308 ; # 303
PLANT.LOOP1.F(64) = 26.308 ; # 304
PLANT.LOOP1.F(65) = 26.308 ; # 305
PLANT.LOOP1.F(66) = 26.308 ; # 306
PLANT.LOOP1.F(67) = 26.308 ; # 307
PLANT.LOOP1.F(68) = 26.308 ; # 308
PLANT.LOOP1.F(69) = 26.308 ; # 309
PLANT.LOOP1.F(70) = 26.308 ; # 310
PLANT.LOOP1.F(71) = 26.308 ; # 311
PLANT.LOOP1.F(72) = 26.308 ; # 312
PLANT.LOOP1.F(73) = 26.308 ; # 313
PLANT.LOOP1.F(74) = 26.308 ; # 314
PLANT.LOOP1.F(75) = 26.308 ; # 315
PLANT.LOOP1.F(76) = 26.308 ; # 316
PLANT.LOOP1.F(77) = 26.308 ; # 317
PLANT.LOOP1.F(78) = 26.308 ; # 318
PLANT.LOOP1.F(79) = 26.308 ; # 319
PLANT.LOOP1.F(80) = 26.308 ; # 320
PLANT.LOOP1.F(81) = 26.308 ; # 321
PLANT.LOOP1.F(82) = 26.308 ; # 322
PLANT.LOOP1.F(83) = 26.308 ; # 323
PLANT.LOOP1.F(84) = 26.308 ; # 324
PLANT.LOOP1.F(85) = 26.308 ; # 325
PLANT.LOOP1.F(86) = 26.308 ; # 326
PLANT.LOOP1.F(87) = 26.308 ; # 327
PLANT.LOOP1.F(88) = 26.308 ; # 328
PLANT.LOOP1.F(89) = 26.308 ; # 329
PLANT.LOOP1.F(90) = 26.308 ; # 330
PLANT.LOOP1.F(91) = 26.308 ; # 331
PLANT.LOOP1.F(92) = 26.308 ; # 332
PLANT.LOOP1.F(93) = 26.308 ; # 333
PLANT.LOOP1.F(94) = 26.308 ; # 334
PLANT.LOOP1.F(95) = 26.308 ; # 335
PLANT.LOOP1.F(96) = 26.308 ; # 336
PLANT.LOOP1.F(97) = 26.308 ; # 337
PLANT.LOOP1.F(98) = 26.308 ; # 338
PLANT.LOOP1.F(99) = 26.308 ; # 339
PLANT.LOOP1.F(100) = 26.308 ; # 340
PLANT.LOOP1.RHOEXTRA(1) = 1.1654 ; # 341
PLANT.LOOP1.RHOEXTRA(2) = 1.1654 ; # 342
PLANT.LOOP1.PSIEXTRA(1) = 1.24585e+006 ; # 343
PLANT.LOOP1.PSIEXTRA(2) = 1.24585e+006 ; # 344
PLANT.LOOP1.FEXTRA(1) = 26.308 ; # 345
PLANT.LOOP1.FEXTRA(2) = 26.308 ; # 346
PLANT.PIPEH.RHO(1) = 2.87869 ; # 347
PLANT.PIPEH.RHO(2) = 2.88664 ; # 348
PLANT.PIPEH.RHO(3) = 2.89491 ; # 349
PLANT.PIPEH.RHO(4) = 2.90353 ; # 350
PLANT.PIPEH.RHO(5) = 2.9125 ; # 351
PLANT.PIPEH.RHO(6) = 2.92184 ; # 352
PLANT.PIPEH.RHO(7) = 2.93156 ; # 353
PLANT.PIPEH.RHO(8) = 2.9417 ; # 354
PLANT.PIPEH.RHO(9) = 2.95225 ; # 355
PLANT.PIPEH.RHO(10) = 2.96325 ; # 356
PLANT.PIPEH.RHO(11) = 2.97471 ; # 357
PLANT.PIPEH.RHO(12) = 2.98666 ; # 358
PLANT.PIPEH.RHO(13) = 2.99912 ; # 359
PLANT.PIPEH.RHO(14) = 3.01211 ; # 360
PLANT.PIPEH.RHO(15) = 3.02567 ; # 361
PLANT.PIPEH.RHO(16) = 3.03982 ; # 362
PLANT.PIPEH.RHO(17) = 3.05458 ; # 363
PLANT.PIPEH.RHO(18) = 3.07 ; # 364
PLANT.PIPEH.RHO(19) = 3.08611 ; # 365
PLANT.PIPEH.RHO(20) = 3.10294 ; # 366
PLANT.PIPEH.RHO(21) = 3.12053 ; # 367
PLANT.PIPEH.RHO(22) = 3.13892 ; # 368
PLANT.PIPEH.RHO(23) = 3.15816 ; # 369
PLANT.PIPEH.RHO(24) = 3.1783 ; # 370
PLANT.PIPEH.RHO(25) = 3.19939 ; # 371
PLANT.PIPEH.RHO(26) = 3.22148 ; # 372
PLANT.PIPEH.RHO(27) = 3.24463 ; # 373
PLANT.PIPEH.RHO(28) = 3.2689 ; # 374
PLANT.PIPEH.RHO(29) = 3.29437 ; # 375
PLANT.PIPEH.RHO(30) = 3.32112 ; # 376
PLANT.PIPEH.RHO(31) = 3.34921 ; # 377
PLANT.PIPEH.RHO(32) = 3.37874 ; # 378
PLANT.PIPEH.RHO(33) = 3.4098 ; # 379
PLANT.PIPEH.RHO(34) = 3.4425 ; # 380
PLANT.PIPEH.RHO(35) = 3.47695 ; # 381
PLANT.PIPEH.RHO(36) = 3.51328 ; # 382
PLANT.PIPEH.RHO(37) = 3.55161 ; # 383
PLANT.PIPEH.RHO(38) = 3.59209 ; # 384
PLANT.PIPEH.RHO(39) = 3.6349 ; # 385
PLANT.PIPEH.RHO(40) = 3.68012 ; # 386
PLANT.PIPEH.F(1) = 32.3305 ; # 387
PLANT.PIPEH.F(2) = 32.3305 ; # 388
```

```
391   PLANT.PIPEH.F(3) = 32.3305 ; # 389
392   PLANT.PIPEH.F(4) = 32.3305 ; # 390
393   PLANT.PIPEH.F(5) = 32.3305 ; # 391
394   PLANT.PIPEH.F(6) = 32.3305 ; # 392
395   PLANT.PIPEH.F(7) = 32.3305 ; # 393
396   PLANT.PIPEH.F(8) = 32.3305 ; # 394
397   PLANT.PIPEH.F(9) = 32.3305 ; # 395
398   PLANT.PIPEH.F(10) = 32.3305 ; # 396
399   PLANT.PIPEH.F(11) = 32.3305 ; # 397
400   PLANT.PIPEH.F(12) = 32.3305 ; # 398
401   PLANT.PIPEH.F(13) = 32.3305 ; # 399
402   PLANT.PIPEH.F(14) = 32.3305 ; # 400
403   PLANT.PIPEH.F(15) = 32.3305 ; # 401
404   PLANT.PIPEH.F(16) = 32.3305 ; # 402
405   PLANT.PIPEH.F(17) = 32.3305 ; # 403
406   PLANT.PIPEH.F(18) = 32.3305 ; # 404
407   PLANT.PIPEH.F(19) = 32.3305 ; # 405
408   PLANT.PIPEH.F(20) = 32.3305 ; # 406
409   PLANT.PIPEH.F(21) = 32.3305 ; # 407
410   PLANT.PIPEH.F(22) = 32.3305 ; # 408
411   PLANT.PIPEH.F(23) = 32.3305 ; # 409
412   PLANT.PIPEH.F(24) = 32.3305 ; # 410
413   PLANT.PIPEH.F(25) = 32.3305 ; # 411
414   PLANT.PIPEH.F(26) = 32.3305 ; # 412
415   PLANT.PIPEH.F(27) = 32.3305 ; # 413
416   PLANT.PIPEH.F(28) = 32.3305 ; # 414
417   PLANT.PIPEH.F(29) = 32.3305 ; # 415
418   PLANT.PIPEH.F(30) = 32.3305 ; # 416
419   PLANT.PIPEH.F(31) = 32.3305 ; # 417
420   PLANT.PIPEH.F(32) = 32.3305 ; # 418
421   PLANT.PIPEH.F(33) = 32.3305 ; # 419
422   PLANT.PIPEH.F(34) = 32.3305 ; # 420
423   PLANT.PIPEH.F(35) = 32.3305 ; # 421
424   PLANT.PIPEH.F(36) = 32.3305 ; # 422
425   PLANT.PIPEH.F(37) = 32.3305 ; # 423
426   PLANT.PIPEH.F(38) = 32.3305 ; # 424
427   PLANT.PIPEH.F(39) = 32.3305 ; # 425
428   PLANT.PIPEH.F(40) = 32.3305 ; # 426
429   PLANT.PIPEH.F_IN = 32.3305 ; # 427
430   PLANT.PIPEH.PSI(1) = 6.03823e+006 ; # 428
431   PLANT.PIPEH.PSI(2) = 6.02385e+006 ; # 429
432   PLANT.PIPEH.PSI(3) = 6.00898e+006 ; # 430
433   PLANT.PIPEH.PSI(4) = 5.9936e+006 ; # 431
434   PLANT.PIPEH.PSI(5) = 5.97767e+006 ; # 432
435   PLANT.PIPEH.PSI(6) = 5.96117e+006 ; # 433
436   PLANT.PIPEH.PSI(7) = 5.94408e+006 ; # 434
437   PLANT.PIPEH.PSI(8) = 5.92638e+006 ; # 435
438   PLANT.PIPEH.PSI(9) = 5.90802e+006 ; # 436
439   PLANT.PIPEH.PSI(10) = 5.88899e+006 ; # 437
440   PLANT.PIPEH.PSI(11) = 5.86924e+006 ; # 438
441   PLANT.PIPEH.PSI(12) = 5.84876e+006 ; # 439
442   PLANT.PIPEH.PSI(13) = 5.82749e+006 ; # 440
443   PLANT.PIPEH.PSI(14) = 5.8054e+006 ; # 441
444   PLANT.PIPEH.PSI(15) = 5.78245e+006 ; # 442
445   PLANT.PIPEH.PSI(16) = 5.75859e+006 ; # 443
446   PLANT.PIPEH.PSI(17) = 5.73379e+006 ; # 444
447   PLANT.PIPEH.PSI(18) = 5.70799e+006 ; # 445
448   PLANT.PIPEH.PSI(19) = 5.68113e+006 ; # 446
449   PLANT.PIPEH.PSI(20) = 5.65317e+006 ; # 447
450   PLANT.PIPEH.PSI(21) = 5.62404e+006 ; # 448
451   PLANT.PIPEH.PSI(22) = 5.59367e+006 ; # 449
452   PLANT.PIPEH.PSI(23) = 5.56201e+006 ; # 450
453   PLANT.PIPEH.PSI(24) = 5.52897e+006 ; # 451
454   PLANT.PIPEH.PSI(25) = 5.49447e+006 ; # 452
455   PLANT.PIPEH.PSI(26) = 5.45844e+006 ; # 453
456   PLANT.PIPEH.PSI(27) = 5.42078e+006 ; # 454
457   PLANT.PIPEH.PSI(28) = 5.38139e+006 ; # 455
458   PLANT.PIPEH.PSI(29) = 5.34016e+006 ; # 456
459   PLANT.PIPEH.PSI(30) = 5.29699e+006 ; # 457
460   PLANT.PIPEH.PSI(31) = 5.25174e+006 ; # 458
461   PLANT.PIPEH.PSI(32) = 5.20427e+006 ; # 459
462   PLANT.PIPEH.PSI(33) = 5.15446e+006 ; # 460
463   PLANT.PIPEH.PSI(34) = 5.10212e+006 ; # 461
464   PLANT.PIPEH.PSI(35) = 5.0471e+006 ; # 462
465   PLANT.PIPEH.PSI(36) = 4.98919e+006 ; # 463
466   PLANT.PIPEH.PSI(37) = 4.92819e+006 ; # 464
467   PLANT.PIPEH.PSI(38) = 4.86388e+006 ; # 465
468   PLANT.PIPEH.PSI(39) = 4.796e+006 ; # 466
469   PLANT.PIPEH.PSI(40) = 4.7244e+006 ; # 467
470   PLANT.HX2.THX(1) = 1133.71 ; # 468
471   PLANT.HX2.THX(2) = 1130.67 ; # 469
472   PLANT.HX2.THX(3) = 1127.46 ; # 470
473   PLANT.HX2.THX(4) = 1124.06 ; # 471
474   PLANT.HX2.THX(5) = 1120.46 ; # 472
475   PLANT.HX2.THX(6) = 1116.65 ; # 473
476   PLANT.HX2.THX(7) = 1112.63 ; # 474
477   PLANT.HX2.THX(8) = 1108.36 ; # 475
478   PLANT.HX2.THX(9) = 1103.85 ; # 476
479   PLANT.HX2.THX(10) = 1099.08 ; # 477
480   PLANT.HX2.THX(11) = 1094.03 ; # 478
481   PLANT.HX2.THX(12) = 1088.69 ; # 479
482   PLANT.HX2.THX(13) = 1083.03 ; # 480
```

```
PLANT.HX2.THX(14) = 1077.05 ; # 481
PLANT.HX2.THX(15) = 1070.72 ; # 482
PLANT.HX2.THX(16) = 1064.02 ; # 483
PLANT.HX2.THX(17) = 1056.93 ; # 484
PLANT.HX2.THX(18) = 1049.43 ; # 485
PLANT.HX2.THX(19) = 1041.49 ; # 486
PLANT.HX2.THX(20) = 1033.09 ; # 487
PLANT.HX2.THX(21) = 1024.2 ; # 488
PLANT.HX2.THX(22) = 1014.8 ; # 489
PLANT.HX2.THX(23) = 1004.84 ; # 490
PLANT.HX2.THX(24) = 994.314 ; # 491
PLANT.HX2.THX(25) = 983.171 ; # 492
PLANT.HX2.THX(26) = 971.38 ; # 493
PLANT.HX2.THX(27) = 958.903 ; # 494
PLANT.HX2.THX(28) = 945.701 ; # 495
PLANT.HX2.THX(29) = 931.731 ; # 496
PLANT.HX2.THX(30) = 916.948 ; # 497
PLANT.HX2.THX(31) = 901.306 ; # 498
PLANT.HX2.THX(32) = 884.754 ; # 499
PLANT.HX2.THX(33) = 867.239 ; # 500
PLANT.HX2.THX(34) = 848.706 ; # 501
PLANT.HX2.THX(35) = 829.095 ; # 502
PLANT.HX2.THX(36) = 808.344 ; # 503
PLANT.HX2.THX(37) = 786.385 ; # 504
PLANT.HX2.THX(38) = 763.15 ; # 505
PLANT.HX2.THX(39) = 738.563 ; # 506
PLANT.HX2.THX(40) = 712.554 ; # 507
PLANT.PIPEC2.RHO(1) = 3.72389 ; # 508
PLANT.PIPEC2.RHO(2) = 3.55368 ; # 509
PLANT.PIPEC2.RHO(3) = 3.4055 ; # 510
PLANT.PIPEC2.RHO(4) = 3.27543 ; # 511
PLANT.PIPEC2.RHO(5) = 3.16041 ; # 512
PLANT.PIPEC2.RHO(6) = 3.05803 ; # 513
PLANT.PIPEC2.RHO(7) = 2.96638 ; # 514
PLANT.PIPEC2.RHO(8) = 2.88388 ; # 515
PLANT.PIPEC2.RHO(9) = 2.80928 ; # 516
PLANT.PIPEC2.RHO(10) = 2.74152 ; # 517
PLANT.PIPEC2.RHO(11) = 2.67973 ; # 518
PLANT.PIPEC2.RHO(12) = 2.62316 ; # 519
PLANT.PIPEC2.RHO(13) = 2.57121 ; # 520
PLANT.PIPEC2.RHO(14) = 2.52333 ; # 521
PLANT.PIPEC2.RHO(15) = 2.47909 ; # 522
PLANT.PIPEC2.RHO(16) = 2.43808 ; # 523
PLANT.PIPEC2.RHO(17) = 2.39997 ; # 524
PLANT.PIPEC2.RHO(18) = 2.36446 ; # 525
PLANT.PIPEC2.RHO(19) = 2.3313 ; # 526
PLANT.PIPEC2.RHO(20) = 2.30026 ; # 527
PLANT.PIPEC2.RHO(21) = 2.27113 ; # 528
PLANT.PIPEC2.RHO(22) = 2.24375 ; # 529
PLANT.PIPEC2.RHO(23) = 2.21796 ; # 530
PLANT.PIPEC2.RHO(24) = 2.19361 ; # 531
PLANT.PIPEC2.RHO(25) = 2.17058 ; # 532
PLANT.PIPEC2.RHO(26) = 2.14875 ; # 533
PLANT.PIPEC2.RHO(27) = 2.12804 ; # 534
PLANT.PIPEC2.RHO(28) = 2.10834 ; # 535
PLANT.PIPEC2.RHO(29) = 2.08957 ; # 536
PLANT.PIPEC2.RHO(30) = 2.07166 ; # 537
PLANT.PIPEC2.RHO(31) = 2.05454 ; # 538
PLANT.PIPEC2.RHO(32) = 2.03814 ; # 539
PLANT.PIPEC2.RHO(33) = 2.02242 ; # 540
PLANT.PIPEC2.RHO(34) = 2.00733 ; # 541
PLANT.PIPEC2.RHO(35) = 1.9928 ; # 542
PLANT.PIPEC2.RHO(36) = 1.97881 ; # 543
PLANT.PIPEC2.RHO(37) = 1.96531 ; # 544
PLANT.PIPEC2.RHO(38) = 1.95226 ; # 545
PLANT.PIPEC2.RHO(39) = 1.93964 ; # 546
PLANT.PIPEC2.RHO(40) = 1.92741 ; # 547
PLANT.PIPEC2.F(1) = 43.3743 ; # 548
PLANT.PIPEC2.F(2) = 43.3743 ; # 549
PLANT.PIPEC2.F(3) = 43.3743 ; # 550
PLANT.PIPEC2.F(4) = 43.3743 ; # 551
PLANT.PIPEC2.F(5) = 43.3743 ; # 552
PLANT.PIPEC2.F(6) = 43.3743 ; # 553
PLANT.PIPEC2.F(7) = 43.3743 ; # 554
PLANT.PIPEC2.F(8) = 43.3743 ; # 555
PLANT.PIPEC2.F(9) = 43.3743 ; # 556
PLANT.PIPEC2.F(10) = 43.3743 ; # 557
PLANT.PIPEC2.F(11) = 43.3743 ; # 558
PLANT.PIPEC2.F(12) = 43.3743 ; # 559
PLANT.PIPEC2.F(13) = 43.3743 ; # 560
PLANT.PIPEC2.F(14) = 43.3743 ; # 561
PLANT.PIPEC2.F(15) = 43.3743 ; # 562
PLANT.PIPEC2.F(16) = 43.3743 ; # 563
PLANT.PIPEC2.F(17) = 43.3743 ; # 564
PLANT.PIPEC2.F(18) = 43.3743 ; # 565
PLANT.PIPEC2.F(19) = 43.3743 ; # 566
PLANT.PIPEC2.F(20) = 43.3743 ; # 567
PLANT.PIPEC2.F(21) = 43.3743 ; # 568
PLANT.PIPEC2.F(22) = 43.3743 ; # 569
PLANT.PIPEC2.F(23) = 43.3743 ; # 570
PLANT.PIPEC2.F(24) = 43.3743 ; # 571
PLANT.PIPEC2.F(25) = 43.3743 ; # 572
```

```
PLANT.PIPEC2.F(26) = 43.3743 ; # 573
PLANT.PIPEC2.F(27) = 43.3743 ; # 574
PLANT.PIPEC2.F(28) = 43.3743 ; # 575
PLANT.PIPEC2.F(29) = 43.3743 ; # 576
PLANT.PIPEC2.F(30) = 43.3743 ; # 577
PLANT.PIPEC2.F(31) = 43.3743 ; # 578
PLANT.PIPEC2.F(32) = 43.3743 ; # 579
PLANT.PIPEC2.F(33) = 43.3743 ; # 580
PLANT.PIPEC2.F(34) = 43.3743 ; # 581
PLANT.PIPEC2.F(35) = 43.3743 ; # 582
PLANT.PIPEC2.F(36) = 43.3743 ; # 583
PLANT.PIPEC2.F(37) = 43.3743 ; # 584
PLANT.PIPEC2.F(38) = 43.3743 ; # 585
PLANT.PIPEC2.F(39) = 43.3743 ; # 586
PLANT.PIPEC2.F(40) = 43.3743 ; # 587
PLANT.PIPEC2.F_IN = 43.3743 ; # 588
PLANT.PIPEC2.PSI(1) = 1.81063e+006 ; # 589
PLANT.PIPEC2.PSI(2) = 1.91725e+006 ; # 590
PLANT.PIPEC2.PSI(3) = 2.00881e+006 ; # 591
PLANT.PIPEC2.PSI(4) = 2.08797e+006 ; # 592
PLANT.PIPEC2.PSI(5) = 2.1568e+006 ; # 593
PLANT.PIPEC2.PSI(6) = 2.21692e+006 ; # 594
PLANT.PIPEC2.PSI(7) = 2.26963e+006 ; # 595
PLANT.PIPEC2.PSI(8) = 2.31599e+006 ; # 596
PLANT.PIPEC2.PSI(9) = 2.35686e+006 ; # 597
PLANT.PIPEC2.PSI(10) = 2.39296e+006 ; # 598
PLANT.PIPEC2.PSI(11) = 2.42488e+006 ; # 599
PLANT.PIPEC2.PSI(12) = 2.45312e+006 ; # 600
PLANT.PIPEC2.PSI(13) = 2.47811e+006 ; # 601
PLANT.PIPEC2.PSI(14) = 2.50021e+006 ; # 602
PLANT.PIPEC2.PSI(15) = 2.51973e+006 ; # 603
PLANT.PIPEC2.PSI(16) = 2.53693e+006 ; # 604
PLANT.PIPEC2.PSI(17) = 2.55205e+006 ; # 605
PLANT.PIPEC2.PSI(18) = 2.5653e+006 ; # 606
PLANT.PIPEC2.PSI(19) = 2.57684e+006 ; # 607
PLANT.PIPEC2.PSI(20) = 2.58684e+006 ; # 608
PLANT.PIPEC2.PSI(21) = 2.59544e+006 ; # 609
PLANT.PIPEC2.PSI(22) = 2.60275e+006 ; # 610
PLANT.PIPEC2.PSI(23) = 2.60889e+006 ; # 611
PLANT.PIPEC2.PSI(24) = 2.61395e+006 ; # 612
PLANT.PIPEC2.PSI(25) = 2.61803e+006 ; # 613
PLANT.PIPEC2.PSI(26) = 2.62119e+006 ; # 614
PLANT.PIPEC2.PSI(27) = 2.62352e+006 ; # 615
PLANT.PIPEC2.PSI(28) = 2.62507e+006 ; # 616
PLANT.PIPEC2.PSI(29) = 2.62591e+006 ; # 617
PLANT.PIPEC2.PSI(30) = 2.62607e+006 ; # 618
PLANT.PIPEC2.PSI(31) = 2.62562e+006 ; # 619
PLANT.PIPEC2.PSI(32) = 2.6246e+006 ; # 620
PLANT.PIPEC2.PSI(33) = 2.62304e+006 ; # 621
PLANT.PIPEC2.PSI(34) = 2.62099e+006 ; # 622
PLANT.PIPEC2.PSI(35) = 2.61847e+006 ; # 623
PLANT.PIPEC2.PSI(36) = 2.61551e+006 ; # 624
PLANT.PIPEC2.PSI(37) = 2.61215e+006 ; # 625
PLANT.PIPEC2.PSI(38) = 2.6084e+006 ; # 626
PLANT.PIPEC2.PSI(39) = 2.6043e+006 ; # 627
PLANT.PIPEC2.PSI(40) = 2.59986e+006 ; # 628
```

HTLFULLDYN_SS_IN_PRE.PRESETS

```
# Values for computation HTLFULLDYN_SS
# Saved at time 1000
PLANT.HX.TGASHOT(1) := 908.671 ; # 1
PLANT.HX.TGASHOT(2) := 920.116 ; # 2
PLANT.HX.TGASHOT(3) := 931.225 ; # 3
PLANT.HX.TGASHOT(4) := 941.988 ; # 4
PLANT.HX.TGASHOT(5) := 952.416 ; # 5
PLANT.HX.TGASHOT(6) := 962.521 ; # 6
PLANT.HX.TGASHOT(7) := 972.311 ; # 7
PLANT.HX.TGASHOT(8) := 981.796 ; # 8
PLANT.HX.TGASHOT(9) := 990.987 ; # 9
PLANT.HX.TGASHOT(10) := 999.892 ; # 10
PLANT.HX.TGASHOT(11) := 1008.52 ; # 11
PLANT.HX.TGASHOT(12) := 1016.88 ; # 12
PLANT.HX.TGASHOT(13) := 1024.98 ; # 13
PLANT.HX.TGASHOT(14) := 1032.83 ; # 14
PLANT.HX.TGASHOT(15) := 1040.43 ; # 15
PLANT.HX.TGASHOT(16) := 1047.8 ; # 16
PLANT.HX.TGASHOT(17) := 1054.94 ; # 17
PLANT.HX.TGASHOT(18) := 1061.85 ; # 18
PLANT.HX.TGASHOT(19) := 1068.55 ; # 19
PLANT.HX.TGASHOT(20) := 1075.05 ; # 20
PLANT.HX.TGASHOT(21) := 1081.34 ; # 21
PLANT.HX.TGASHOT(22) := 1087.43 ; # 22
```

```
PLANT.HX.TGASHOT(23) := 1093.34 ; # 23
PLANT.HX.TGASHOT(24) := 1099.06 ; # 24
PLANT.HX.TGASHOT(25) := 1104.6 ; # 25
PLANT.HX.TGASHOT(26) := 1109.98 ; # 26
PLANT.HX.TGASHOT(27) := 1115.18 ; # 27
PLANT.HX.TGASHOT(28) := 1120.22 ; # 28
PLANT.HX.TGASHOT(29) := 1125.11 ; # 29
PLANT.HX.TGASHOT(30) := 1129.84 ; # 30
PLANT.HX.TGASHOT(31) := 1134.43 ; # 31
PLANT.HX.TGASHOT(32) := 1138.87 ; # 32
PLANT.HX.TGASHOT(33) := 1143.18 ; # 33
PLANT.HX.TGASHOT(34) := 1147.35 ; # 34
PLANT.HX.TGASHOT(35) := 1151.39 ; # 35
PLANT.HX.TGASHOT(36) := 1155.31 ; # 36
PLANT.HX.TGASHOT(37) := 1159.11 ; # 37
PLANT.HX.TGASHOT(38) := 1162.78 ; # 38
PLANT.HX.TGASHOT(39) := 1166.35 ; # 39
PLANT.HX.TGASHOT(40) := 1169.8 ; # 40
PLANT.HX.TGASCOLD(1) := 849.368 ; # 41
PLANT.HX.TGASCOLD(2) := 862.676 ; # 42
PLANT.HX.TGASCOLD(3) := 875.571 ; # 43
PLANT.HX.TGASCOLD(4) := 888.065 ; # 44
PLANT.HX.TGASCOLD(5) := 900.17 ; # 45
PLANT.HX.TGASCOLD(6) := 911.899 ; # 46
PLANT.HX.TGASCOLD(7) := 923.264 ; # 47
PLANT.HX.TGASCOLD(8) := 934.274 ; # 48
PLANT.HX.TGASCOLD(9) := 944.943 ; # 49
PLANT.HX.TGASCOLD(10) := 955.279 ; # 50
PLANT.HX.TGASCOLD(11) := 965.295 ; # 51
PLANT.HX.TGASCOLD(12) := 974.998 ; # 52
PLANT.HX.TGASCOLD(13) := 984.4 ; # 53
PLANT.HX.TGASCOLD(14) := 993.51 ; # 54
PLANT.HX.TGASCOLD(15) := 1002.34 ; # 55
PLANT.HX.TGASCOLD(16) := 1010.89 ; # 56
PLANT.HX.TGASCOLD(17) := 1019.17 ; # 57
PLANT.HX.TGASCOLD(18) := 1027.2 ; # 58
PLANT.HX.TGASCOLD(19) := 1034.98 ; # 59
PLANT.HX.TGASCOLD(20) := 1042.52 ; # 60
PLANT.HX.TGASCOLD(21) := 1049.82 ; # 61
PLANT.HX.TGASCOLD(22) := 1056.9 ; # 62
PLANT.HX.TGASCOLD(23) := 1063.75 ; # 63
PLANT.HX.TGASCOLD(24) := 1070.39 ; # 64
PLANT.HX.TGASCOLD(25) := 1076.83 ; # 65
PLANT.HX.TGASCOLD(26) := 1083.06 ; # 66
PLANT.HX.TGASCOLD(27) := 1089.11 ; # 67
PLANT.HX.TGASCOLD(28) := 1094.96 ; # 68
PLANT.HX.TGASCOLD(29) := 1100.63 ; # 69
PLANT.HX.TGASCOLD(30) := 1106.13 ; # 70
PLANT.HX.TGASCOLD(31) := 1111.45 ; # 71
PLANT.HX.TGASCOLD(32) := 1116.61 ; # 72
PLANT.HX.TGASCOLD(33) := 1121.61 ; # 73
PLANT.HX.TGASCOLD(34) := 1126.45 ; # 74
PLANT.HX.TGASCOLD(35) := 1131.14 ; # 75
PLANT.HX.TGASCOLD(36) := 1135.69 ; # 76
PLANT.HX.TGASCOLD(37) := 1140.09 ; # 77
PLANT.HX.TGASCOLD(38) := 1144.36 ; # 78
PLANT.HX.TGASCOLD(39) := 1148.5 ; # 79
PLANT.HX.TGASCOLD(40) := 1152.5 ; # 80
PLANT.HX.UHOT := 1738.41 ; # 81
PLANT.HX.UCOLD := 1645.45 ; # 82
PLANT.HX.THX(1) := 879.893 ; # 83
PLANT.HX.THX(2) := 892.184 ; # 84
PLANT.HX.THX(3) := 904.161 ; # 85
PLANT.HX.THX(4) := 915.766 ; # 86
PLANT.HX.THX(5) := 927.009 ; # 87
PLANT.HX.THX(6) := 937.904 ; # 88
PLANT.HX.THX(7) := 948.459 ; # 89
PLANT.HX.THX(8) := 958.687 ; # 90
PLANT.HX.THX(9) := 968.596 ; # 91
PLANT.HX.THX(10) := 978.197 ; # 92
PLANT.HX.THX(11) := 987.5 ; # 93
PLANT.HX.THX(12) := 996.513 ; # 94
PLANT.HX.THX(13) := 1005.25 ; # 95
PLANT.HX.THX(14) := 1013.71 ; # 96
PLANT.HX.THX(15) := 1021.91 ; # 97
PLANT.HX.THX(16) := 1029.85 ; # 98
PLANT.HX.THX(17) := 1037.55 ; # 99
PLANT.HX.THX(18) := 1045 ; # 100
PLANT.HX.THX(19) := 1052.23 ; # 101
PLANT.HX.THX(20) := 1059.23 ; # 102
PLANT.HX.THX(21) := 1066.01 ; # 103
PLANT.HX.THX(22) := 1072.58 ; # 104
PLANT.HX.THX(23) := 1078.95 ; # 105
PLANT.HX.THX(24) := 1085.12 ; # 106
PLANT.HX.THX(25) := 1091.1 ; # 107
PLANT.HX.THX(26) := 1096.89 ; # 108
PLANT.HX.THX(27) := 1102.5 ; # 109
PLANT.HX.THX(28) := 1107.94 ; # 110
PLANT.HX.THX(29) := 1113.21 ; # 111
PLANT.HX.THX(30) := 1118.31 ; # 112
PLANT.HX.THX(31) := 1123.26 ; # 113
PLANT.HX.THX(32) := 1128.05 ; # 114
```

```
PLANT.HX.THX(33) := 1132.69 ; # 115
PLANT.HX.THX(34) := 1137.19 ; # 116
PLANT.HX.THX(35) := 1141.55 ; # 117
PLANT.HX.THX(36) := 1145.77 ; # 118
PLANT.HX.THX(37) := 1149.86 ; # 119
PLANT.HX.THX(38) := 1153.83 ; # 120
PLANT.HX.THX(39) := 1157.67 ; # 121
PLANT.HX.THX(40) := 1161.37 ; # 122
PLANT.LOOP1.RHO(1) := 1.15317 ; # 123
PLANT.LOOP1.RHO(2) := 1.15067 ; # 124
PLANT.LOOP1.RHO(3) := 1.14817 ; # 125
PLANT.LOOP1.RHO(4) := 1.14565 ; # 126
PLANT.LOOP1.RHO(5) := 1.14313 ; # 127
PLANT.LOOP1.RHO(6) := 1.14059 ; # 128
PLANT.LOOP1.RHO(7) := 1.13804 ; # 129
PLANT.LOOP1.RHO(8) := 1.13549 ; # 130
PLANT.LOOP1.RHO(9) := 1.13292 ; # 131
PLANT.LOOP1.RHO(10) := 1.13034 ; # 132
PLANT.LOOP1.RHO(11) := 1.12115 ; # 133
PLANT.LOOP1.RHO(12) := 1.1026 ; # 134
PLANT.LOOP1.RHO(13) := 1.08477 ; # 135
PLANT.LOOP1.RHO(14) := 1.06791 ; # 136
PLANT.LOOP1.RHO(15) := 1.05196 ; # 137
PLANT.LOOP1.RHO(16) := 1.03683 ; # 138
PLANT.LOOP1.RHO(17) := 1.02247 ; # 139
PLANT.LOOP1.RHO(18) := 1.00882 ; # 140
PLANT.LOOP1.RHO(19) := 0.99583 ; # 141
PLANT.LOOP1.RHO(20) := 0.983453 ; # 142
PLANT.LOOP1.RHO(21) := 0.971645 ; # 143
PLANT.LOOP1.RHO(22) := 0.960369 ; # 144
PLANT.LOOP1.RHO(23) := 0.949588 ; # 145
PLANT.LOOP1.RHO(24) := 0.939271 ; # 146
PLANT.LOOP1.RHO(25) := 0.929389 ; # 147
PLANT.LOOP1.RHO(26) := 0.919912 ; # 148
PLANT.LOOP1.RHO(27) := 0.910817 ; # 149
PLANT.LOOP1.RHO(28) := 0.90208 ; # 150
PLANT.LOOP1.RHO(29) := 0.893679 ; # 151
PLANT.LOOP1.RHO(30) := 0.885595 ; # 152
PLANT.LOOP1.RHO(31) := 0.87781 ; # 153
PLANT.LOOP1.RHO(32) := 0.870305 ; # 154
PLANT.LOOP1.RHO(33) := 0.863066 ; # 155
PLANT.LOOP1.RHO(34) := 0.856077 ; # 156
PLANT.LOOP1.RHO(35) := 0.849325 ; # 157
PLANT.LOOP1.RHO(36) := 0.842796 ; # 158
PLANT.LOOP1.RHO(37) := 0.83648 ; # 159
PLANT.LOOP1.RHO(38) := 0.830363 ; # 160
PLANT.LOOP1.RHO(39) := 0.824437 ; # 161
PLANT.LOOP1.RHO(40) := 0.818691 ; # 162
PLANT.LOOP1.RHO(41) := 0.813116 ; # 163
PLANT.LOOP1.RHO(42) := 0.807703 ; # 164
PLANT.LOOP1.RHO(43) := 0.802444 ; # 165
PLANT.LOOP1.RHO(44) := 0.797332 ; # 166
PLANT.LOOP1.RHO(45) := 0.792358 ; # 167
PLANT.LOOP1.RHO(46) := 0.787518 ; # 168
PLANT.LOOP1.RHO(47) := 0.782803 ; # 169
PLANT.LOOP1.RHO(48) := 0.778207 ; # 170
PLANT.LOOP1.RHO(49) := 0.773726 ; # 171
PLANT.LOOP1.RHO(50) := 0.769358 ; # 172
PLANT.LOOP1.RHO(51) := 0.757959 ; # 173
PLANT.LOOP1.RHO(52) := 0.756103 ; # 174
PLANT.LOOP1.RHO(53) := 0.754256 ; # 175
PLANT.LOOP1.RHO(54) := 0.752399 ; # 176
PLANT.LOOP1.RHO(55) := 0.750533 ; # 177
PLANT.LOOP1.RHO(56) := 0.748658 ; # 178
PLANT.LOOP1.RHO(57) := 0.746773 ; # 179
PLANT.LOOP1.RHO(58) := 0.744879 ; # 180
PLANT.LOOP1.RHO(59) := 0.742974 ; # 181
PLANT.LOOP1.RHO(60) := 0.74106 ; # 182
PLANT.LOOP1.RHO(61) := 0.750585 ; # 183
PLANT.LOOP1.RHO(62) := 0.751689 ; # 184
PLANT.LOOP1.RHO(63) := 0.752856 ; # 185
PLANT.LOOP1.RHO(64) := 0.754135 ; # 186
PLANT.LOOP1.RHO(65) := 0.755533 ; # 187
PLANT.LOOP1.RHO(66) := 0.757058 ; # 188
PLANT.LOOP1.RHO(67) := 0.75872 ; # 189
PLANT.LOOP1.RHO(68) := 0.760528 ; # 190
PLANT.LOOP1.RHO(69) := 0.762492 ; # 191
PLANT.LOOP1.RHO(70) := 0.764623 ; # 192
PLANT.LOOP1.RHO(71) := 0.766934 ; # 193
PLANT.LOOP1.RHO(72) := 0.769438 ; # 194
PLANT.LOOP1.RHO(73) := 0.772149 ; # 195
PLANT.LOOP1.RHO(74) := 0.775082 ; # 196
PLANT.LOOP1.RHO(75) := 0.778256 ; # 197
PLANT.LOOP1.RHO(76) := 0.781688 ; # 198
PLANT.LOOP1.RHO(77) := 0.785399 ; # 199
PLANT.LOOP1.RHO(78) := 0.789411 ; # 200
PLANT.LOOP1.RHO(79) := 0.793749 ; # 201
PLANT.LOOP1.RHO(80) := 0.79844 ; # 202
PLANT.LOOP1.RHO(81) := 0.803515 ; # 203
PLANT.LOOP1.RHO(82) := 0.809005 ; # 204
PLANT.LOOP1.RHO(83) := 0.814949 ; # 205
PLANT.LOOP1.RHO(84) := 0.821387 ; # 206
```

```
PLANT.LOOP1.RHO(85) := 0.828364 ; # 207
PLANT.LOOP1.RHO(86) := 0.835933 ; # 208
PLANT.LOOP1.RHO(87) := 0.844151 ; # 209
PLANT.LOOP1.RHO(88) := 0.853082 ; # 210
PLANT.LOOP1.RHO(89) := 0.8628 ; # 211
PLANT.LOOP1.RHO(90) := 0.873389 ; # 212
PLANT.LOOP1.RHO(91) := 0.884944 ; # 213
PLANT.LOOP1.RHO(92) := 0.897573 ; # 214
PLANT.LOOP1.RHO(93) := 0.911403 ; # 215
PLANT.LOOP1.RHO(94) := 0.926579 ; # 216
PLANT.LOOP1.RHO(95) := 0.94327 ; # 217
PLANT.LOOP1.RHO(96) := 0.961674 ; # 218
PLANT.LOOP1.RHO(97) := 0.982026 ; # 219
PLANT.LOOP1.RHO(98) := 1.0046 ; # 220
PLANT.LOOP1.RHO(99) := 1.02974 ; # 221
PLANT.LOOP1.RHO(100) := 1.05783 ; # 222
PLANT.LOOP1.P(1) := 2.01181e+006 ; # 223
PLANT.LOOP1.P(2) := 2.00646e+006 ; # 224
PLANT.LOOP1.P(3) := 2.00109e+006 ; # 225
PLANT.LOOP1.P(4) := 1.99571e+006 ; # 226
PLANT.LOOP1.P(5) := 1.99032e+006 ; # 227
PLANT.LOOP1.P(6) := 1.98491e+006 ; # 228
PLANT.LOOP1.P(7) := 1.9795e+006 ; # 229
PLANT.LOOP1.P(8) := 1.97407e+006 ; # 230
PLANT.LOOP1.P(9) := 1.96863e+006 ; # 231
PLANT.LOOP1.P(10) := 1.96317e+006 ; # 232
PLANT.LOOP1.P(11) := 1.97929e+006 ; # 233
PLANT.LOOP1.P(12) := 1.97704e+006 ; # 234
PLANT.LOOP1.P(13) := 1.97414e+006 ; # 235
PLANT.LOOP1.P(14) := 1.9712e+006 ; # 236
PLANT.LOOP1.P(15) := 1.96822e+006 ; # 237
PLANT.LOOP1.P(16) := 1.96519e+006 ; # 238
PLANT.LOOP1.P(17) := 1.96212e+006 ; # 239
PLANT.LOOP1.P(18) := 1.95902e+006 ; # 240
PLANT.LOOP1.P(19) := 1.95587e+006 ; # 241
PLANT.LOOP1.P(20) := 1.95269e+006 ; # 242
PLANT.LOOP1.P(21) := 1.94947e+006 ; # 243
PLANT.LOOP1.P(22) := 1.94622e+006 ; # 244
PLANT.LOOP1.P(23) := 1.94293e+006 ; # 245
PLANT.LOOP1.P(24) := 1.93961e+006 ; # 246
PLANT.LOOP1.P(25) := 1.93625e+006 ; # 247
PLANT.LOOP1.P(26) := 1.93286e+006 ; # 248
PLANT.LOOP1.P(27) := 1.92943e+006 ; # 249
PLANT.LOOP1.P(28) := 1.92598e+006 ; # 250
PLANT.LOOP1.P(29) := 1.92249e+006 ; # 251
PLANT.LOOP1.P(30) := 1.91897e+006 ; # 252
PLANT.LOOP1.P(31) := 1.91543e+006 ; # 253
PLANT.LOOP1.P(32) := 1.91185e+006 ; # 254
PLANT.LOOP1.P(33) := 1.90824e+006 ; # 255
PLANT.LOOP1.P(34) := 1.90461e+006 ; # 256
PLANT.LOOP1.P(35) := 1.90095e+006 ; # 257
PLANT.LOOP1.P(36) := 1.89726e+006 ; # 258
PLANT.LOOP1.P(37) := 1.89354e+006 ; # 259
PLANT.LOOP1.P(38) := 1.8898e+006 ; # 260
PLANT.LOOP1.P(39) := 1.88603e+006 ; # 261
PLANT.LOOP1.P(40) := 1.88224e+006 ; # 262
PLANT.LOOP1.P(41) := 1.87842e+006 ; # 263
PLANT.LOOP1.P(42) := 1.87458e+006 ; # 264
PLANT.LOOP1.P(43) := 1.87071e+006 ; # 265
PLANT.LOOP1.P(44) := 1.86681e+006 ; # 266
PLANT.LOOP1.P(45) := 1.8629e+006 ; # 267
PLANT.LOOP1.P(46) := 1.85896e+006 ; # 268
PLANT.LOOP1.P(47) := 1.855e+006 ; # 269
PLANT.LOOP1.P(48) := 1.85101e+006 ; # 270
PLANT.LOOP1.P(49) := 1.847e+006 ; # 271
PLANT.LOOP1.P(50) := 1.84297e+006 ; # 272
PLANT.LOOP1.P(51) := 1.81438e+006 ; # 273
PLANT.LOOP1.P(52) := 1.80865e+006 ; # 274
PLANT.LOOP1.P(53) := 1.80295e+006 ; # 275
PLANT.LOOP1.P(54) := 1.79723e+006 ; # 276
PLANT.LOOP1.P(55) := 1.7915e+006 ; # 277
PLANT.LOOP1.P(56) := 1.78576e+006 ; # 278
PLANT.LOOP1.P(57) := 1.78e+006 ; # 279
PLANT.LOOP1.P(58) := 1.77423e+006 ; # 280
PLANT.LOOP1.P(59) := 1.76844e+006 ; # 281
PLANT.LOOP1.P(60) := 1.76263e+006 ; # 282
PLANT.LOOP1.P(61) := 1.78151e+006 ; # 283
PLANT.LOOP1.P(62) := 1.78014e+006 ; # 284
PLANT.LOOP1.P(63) := 1.77866e+006 ; # 285
PLANT.LOOP1.P(64) := 1.77719e+006 ; # 286
PLANT.LOOP1.P(65) := 1.77573e+006 ; # 287
PLANT.LOOP1.P(66) := 1.77427e+006 ; # 288
PLANT.LOOP1.P(67) := 1.77281e+006 ; # 289
PLANT.LOOP1.P(68) := 1.77136e+006 ; # 290
PLANT.LOOP1.P(69) := 1.76991e+006 ; # 291
PLANT.LOOP1.P(70) := 1.76847e+006 ; # 292
PLANT.LOOP1.P(71) := 1.76703e+006 ; # 293
PLANT.LOOP1.P(72) := 1.7656e+006 ; # 294
PLANT.LOOP1.P(73) := 1.76417e+006 ; # 295
PLANT.LOOP1.P(74) := 1.76275e+006 ; # 296
PLANT.LOOP1.P(75) := 1.76134e+006 ; # 297
PLANT.LOOP1.P(76) := 1.75993e+006 ; # 298
```

```
PLANT.LOOP1.P(77) := 1.75854e+006 ; # 299
PLANT.LOOP1.P(78) := 1.75715e+006 ; # 300
PLANT.LOOP1.P(79) := 1.75577e+006 ; # 301
PLANT.LOOP1.P(80) := 1.75441e+006 ; # 302
PLANT.LOOP1.P(81) := 1.75305e+006 ; # 303
PLANT.LOOP1.P(82) := 1.7517e+006 ; # 304
PLANT.LOOP1.P(83) := 1.75037e+006 ; # 305
PLANT.LOOP1.P(84) := 1.74905e+006 ; # 306
PLANT.LOOP1.P(85) := 1.74774e+006 ; # 307
PLANT.LOOP1.P(86) := 1.74645e+006 ; # 308
PLANT.LOOP1.P(87) := 1.74518e+006 ; # 309
PLANT.LOOP1.P(88) := 1.74392e+006 ; # 310
PLANT.LOOP1.P(89) := 1.74267e+006 ; # 311
PLANT.LOOP1.P(90) := 1.74145e+006 ; # 312
PLANT.LOOP1.P(91) := 1.74025e+006 ; # 313
PLANT.LOOP1.P(92) := 1.73907e+006 ; # 314
PLANT.LOOP1.P(93) := 1.73791e+006 ; # 315
PLANT.LOOP1.P(94) := 1.73677e+006 ; # 316
PLANT.LOOP1.P(95) := 1.73566e+006 ; # 317
PLANT.LOOP1.P(96) := 1.73458e+006 ; # 318
PLANT.LOOP1.P(97) := 1.73353e+006 ; # 319
PLANT.LOOP1.P(98) := 1.7325e+006 ; # 320
PLANT.LOOP1.P(99) := 1.73151e+006 ; # 321
PLANT.LOOP1.P(100) := 1.73056e+006 ; # 322
PLANT.LOOP1.P_OUT := 1.73003e+006 ; # 323
PLANT.LOOP1.H(1) := 2.81232e+006 ; # 324
PLANT.LOOP1.H(2) := 2.81015e+006 ; # 325
PLANT.LOOP1.H(3) := 2.80798e+006 ; # 326
PLANT.LOOP1.H(4) := 2.80581e+006 ; # 327
PLANT.LOOP1.H(5) := 2.80364e+006 ; # 328
PLANT.LOOP1.H(6) := 2.80148e+006 ; # 329
PLANT.LOOP1.H(7) := 2.79931e+006 ; # 330
PLANT.LOOP1.H(8) := 2.79715e+006 ; # 331
PLANT.LOOP1.H(9) := 2.79499e+006 ; # 332
PLANT.LOOP1.H(10) := 2.79283e+006 ; # 333
PLANT.LOOP1.H(11) := 2.86435e+006 ; # 334
PLANT.LOOP1.H(12) := 2.93349e+006 ; # 335
PLANT.LOOP1.H(13) := 3.00048e+006 ; # 336
PLANT.LOOP1.H(14) := 3.06539e+006 ; # 337
PLANT.LOOP1.H(15) := 3.12828e+006 ; # 338
PLANT.LOOP1.H(16) := 3.18921e+006 ; # 339
PLANT.LOOP1.H(17) := 3.24824e+006 ; # 340
PLANT.LOOP1.H(18) := 3.30545e+006 ; # 341
PLANT.LOOP1.H(19) := 3.36087e+006 ; # 342
PLANT.LOOP1.H(20) := 3.41457e+006 ; # 343
PLANT.LOOP1.H(21) := 3.4666e+006 ; # 344
PLANT.LOOP1.H(22) := 3.51701e+006 ; # 345
PLANT.LOOP1.H(23) := 3.56585e+006 ; # 346
PLANT.LOOP1.H(24) := 3.61317e+006 ; # 347
PLANT.LOOP1.H(25) := 3.65903e+006 ; # 348
PLANT.LOOP1.H(26) := 3.70345e+006 ; # 349
PLANT.LOOP1.H(27) := 3.7465e+006 ; # 350
PLANT.LOOP1.H(28) := 3.78821e+006 ; # 351
PLANT.LOOP1.H(29) := 3.82862e+006 ; # 352
PLANT.LOOP1.H(30) := 3.86777e+006 ; # 353
PLANT.LOOP1.H(31) := 3.90571e+006 ; # 354
PLANT.LOOP1.H(32) := 3.94246e+006 ; # 355
PLANT.LOOP1.H(33) := 3.97808e+006 ; # 356
PLANT.LOOP1.H(34) := 4.01258e+006 ; # 357
PLANT.LOOP1.H(35) := 4.04601e+006 ; # 358
PLANT.LOOP1.H(36) := 4.07841e+006 ; # 359
PLANT.LOOP1.H(37) := 4.10979e+006 ; # 360
PLANT.LOOP1.H(38) := 4.1402e+006 ; # 361
PLANT.LOOP1.H(39) := 4.16967e+006 ; # 362
PLANT.LOOP1.H(40) := 4.19822e+006 ; # 363
PLANT.LOOP1.H(41) := 4.22588e+006 ; # 364
PLANT.LOOP1.H(42) := 4.25268e+006 ; # 365
PLANT.LOOP1.H(43) := 4.27864e+006 ; # 366
PLANT.LOOP1.H(44) := 4.3038e+006 ; # 367
PLANT.LOOP1.H(45) := 4.32818e+006 ; # 368
PLANT.LOOP1.H(46) := 4.3518e+006 ; # 369
PLANT.LOOP1.H(47) := 4.37468e+006 ; # 370
PLANT.LOOP1.H(48) := 4.39685e+006 ; # 371
PLANT.LOOP1.H(49) := 4.41834e+006 ; # 372
PLANT.LOOP1.H(50) := 4.43912e+006 ; # 373
PLANT.LOOP1.H(51) := 4.43487e+006 ; # 374
PLANT.LOOP1.H(52) := 4.43062e+006 ; # 375
PLANT.LOOP1.H(53) := 4.42637e+006 ; # 376
PLANT.LOOP1.H(54) := 4.42213e+006 ; # 377
PLANT.LOOP1.H(55) := 4.41789e+006 ; # 378
PLANT.LOOP1.H(56) := 4.41366e+006 ; # 379
PLANT.LOOP1.H(57) := 4.40943e+006 ; # 380
PLANT.LOOP1.H(58) := 4.40521e+006 ; # 381
PLANT.LOOP1.H(59) := 4.40099e+006 ; # 382
PLANT.LOOP1.H(60) := 4.39677e+006 ; # 383
PLANT.LOOP1.H(61) := 4.38421e+006 ; # 384
PLANT.LOOP1.H(62) := 4.37092e+006 ; # 385
PLANT.LOOP1.H(63) := 4.35686e+006 ; # 386
PLANT.LOOP1.H(64) := 4.34197e+006 ; # 387
PLANT.LOOP1.H(65) := 4.32623e+006 ; # 388
PLANT.LOOP1.H(66) := 4.30957e+006 ; # 389
PLANT.LOOP1.H(67) := 4.29193e+006 ; # 390
```

```
PLANT.LOOP1.H(68) := 4.27328e+006 ; # 391
PLANT.LOOP1.H(69) := 4.25354e+006 ; # 392
PLANT.LOOP1.H(70) := 4.23265e+006 ; # 393
PLANT.LOOP1.H(71) := 4.21054e+006 ; # 394
PLANT.LOOP1.H(72) := 4.18715e+006 ; # 395
PLANT.LOOP1.H(73) := 4.1624e+006 ; # 396
PLANT.LOOP1.H(74) := 4.13621e+006 ; # 397
PLANT.LOOP1.H(75) := 4.1085e+006 ; # 398
PLANT.LOOP1.H(76) := 4.07917e+006 ; # 399
PLANT.LOOP1.H(77) := 4.04814e+006 ; # 400
PLANT.LOOP1.H(78) := 4.01531e+006 ; # 401
PLANT.LOOP1.H(79) := 3.98056e+006 ; # 402
PLANT.LOOP1.H(80) := 3.9438e+006 ; # 403
PLANT.LOOP1.H(81) := 3.90489e+006 ; # 404
PLANT.LOOP1.H(82) := 3.86373e+006 ; # 405
PLANT.LOOP1.H(83) := 3.82017e+006 ; # 406
PLANT.LOOP1.H(84) := 3.77408e+006 ; # 407
PLANT.LOOP1.H(85) := 3.7253e+006 ; # 408
PLANT.LOOP1.H(86) := 3.67369e+006 ; # 409
PLANT.LOOP1.H(87) := 3.61908e+006 ; # 410
PLANT.LOOP1.H(88) := 3.5613e+006 ; # 411
PLANT.LOOP1.H(89) := 3.50015e+006 ; # 412
PLANT.LOOP1.H(90) := 3.43544e+006 ; # 413
PLANT.LOOP1.H(91) := 3.36698e+006 ; # 414
PLANT.LOOP1.H(92) := 3.29453e+006 ; # 415
PLANT.LOOP1.H(93) := 3.21787e+006 ; # 416
PLANT.LOOP1.H(94) := 3.13674e+006 ; # 417
PLANT.LOOP1.H(95) := 3.05091e+006 ; # 418
PLANT.LOOP1.H(96) := 2.96008e+006 ; # 419
PLANT.LOOP1.H(97) := 2.86396e+006 ; # 420
PLANT.LOOP1.H(98) := 2.76226e+006 ; # 421
PLANT.LOOP1.H(99) := 2.65465e+006 ; # 422
PLANT.LOOP1.H(100) := 2.54078e+006 ; # 423
PLANT.LOOP1.H0 := 2.81449e+006 ; # 424
PLANT.LOOP1.T(1) := 839.351 ; # 425
PLANT.LOOP1.T(2) := 838.933 ; # 426
PLANT.LOOP1.T(3) := 838.515 ; # 427
PLANT.LOOP1.T(4) := 838.098 ; # 428
PLANT.LOOP1.T(5) := 837.681 ; # 429
PLANT.LOOP1.T(6) := 837.264 ; # 430
PLANT.LOOP1.T(7) := 836.847 ; # 431
PLANT.LOOP1.T(8) := 836.431 ; # 432
PLANT.LOOP1.T(9) := 836.015 ; # 433
PLANT.LOOP1.T(10) := 835.6 ; # 434
PLANT.LOOP1.T(11) := 849.368 ; # 435
PLANT.LOOP1.T(12) := 862.676 ; # 436
PLANT.LOOP1.T(13) := 875.571 ; # 437
PLANT.LOOP1.T(14) := 888.065 ; # 438
PLANT.LOOP1.T(15) := 900.17 ; # 439
PLANT.LOOP1.T(16) := 911.899 ; # 440
PLANT.LOOP1.T(17) := 923.264 ; # 441
PLANT.LOOP1.T(18) := 934.274 ; # 442
PLANT.LOOP1.T(19) := 944.943 ; # 443
PLANT.LOOP1.T(20) := 955.279 ; # 444
PLANT.LOOP1.T(21) := 965.295 ; # 445
PLANT.LOOP1.T(22) := 974.998 ; # 446
PLANT.LOOP1.T(23) := 984.4 ; # 447
PLANT.LOOP1.T(24) := 993.51 ; # 448
PLANT.LOOP1.T(25) := 1002.34 ; # 449
PLANT.LOOP1.T(26) := 1010.89 ; # 450
PLANT.LOOP1.T(27) := 1019.17 ; # 451
PLANT.LOOP1.T(28) := 1027.2 ; # 452
PLANT.LOOP1.T(29) := 1034.98 ; # 453
PLANT.LOOP1.T(30) := 1042.52 ; # 454
PLANT.LOOP1.T(31) := 1049.82 ; # 455
PLANT.LOOP1.T(32) := 1056.9 ; # 456
PLANT.LOOP1.T(33) := 1063.75 ; # 457
PLANT.LOOP1.T(34) := 1070.39 ; # 458
PLANT.LOOP1.T(35) := 1076.83 ; # 459
PLANT.LOOP1.T(36) := 1083.06 ; # 460
PLANT.LOOP1.T(37) := 1089.11 ; # 461
PLANT.LOOP1.T(38) := 1094.96 ; # 462
PLANT.LOOP1.T(39) := 1100.63 ; # 463
PLANT.LOOP1.T(40) := 1106.13 ; # 464
PLANT.LOOP1.T(41) := 1111.45 ; # 465
PLANT.LOOP1.T(42) := 1116.61 ; # 466
PLANT.LOOP1.T(43) := 1121.61 ; # 467
PLANT.LOOP1.T(44) := 1126.45 ; # 468
PLANT.LOOP1.T(45) := 1131.14 ; # 469
PLANT.LOOP1.T(46) := 1135.69 ; # 470
PLANT.LOOP1.T(47) := 1140.09 ; # 471
PLANT.LOOP1.T(48) := 1144.36 ; # 472
PLANT.LOOP1.T(49) := 1148.5 ; # 473
PLANT.LOOP1.T(50) := 1152.5 ; # 474
PLANT.LOOP1.T(51) := 1151.68 ; # 475
PLANT.LOOP1.T(52) := 1150.86 ; # 476
PLANT.LOOP1.T(53) := 1150.04 ; # 477
PLANT.LOOP1.T(54) := 1149.23 ; # 478
PLANT.LOOP1.T(55) := 1148.41 ; # 479
PLANT.LOOP1.T(56) := 1147.6 ; # 480
PLANT.LOOP1.T(57) := 1146.78 ; # 481
PLANT.LOOP1.T(58) := 1145.97 ; # 482
```

```
PLANT.LOOP1.T(59) := 1145.16 ; # 483
PLANT.LOOP1.T(60) := 1144.35 ; # 484
PLANT.LOOP1.T(61) := 1141.93 ; # 485
PLANT.LOOP1.T(62) := 1139.37 ; # 486
PLANT.LOOP1.T(63) := 1136.66 ; # 487
PLANT.LOOP1.T(64) := 1133.8 ; # 488
PLANT.LOOP1.T(65) := 1130.77 ; # 489
PLANT.LOOP1.T(66) := 1127.56 ; # 490
PLANT.LOOP1.T(67) := 1124.17 ; # 491
PLANT.LOOP1.T(68) := 1120.57 ; # 492
PLANT.LOOP1.T(69) := 1116.77 ; # 493
PLANT.LOOP1.T(70) := 1112.75 ; # 494
PLANT.LOOP1.T(71) := 1108.5 ; # 495
PLANT.LOOP1.T(72) := 1104 ; # 496
PLANT.LOOP1.T(73) := 1099.23 ; # 497
PLANT.LOOP1.T(74) := 1094.19 ; # 498
PLANT.LOOP1.T(75) := 1088.86 ; # 499
PLANT.LOOP1.T(76) := 1083.21 ; # 500
PLANT.LOOP1.T(77) := 1077.24 ; # 501
PLANT.LOOP1.T(78) := 1070.92 ; # 502
PLANT.LOOP1.T(79) := 1064.23 ; # 503
PLANT.LOOP1.T(80) := 1057.15 ; # 504
PLANT.LOOP1.T(81) := 1049.66 ; # 505
PLANT.LOOP1.T(82) := 1041.74 ; # 506
PLANT.LOOP1.T(83) := 1033.36 ; # 507
PLANT.LOOP1.T(84) := 1024.48 ; # 508
PLANT.LOOP1.T(85) := 1015.09 ; # 509
PLANT.LOOP1.T(86) := 1005.16 ; # 510
PLANT.LOOP1.T(87) := 994.647 ; # 511
PLANT.LOOP1.T(88) := 983.524 ; # 512
PLANT.LOOP1.T(89) := 971.753 ; # 513
PLANT.LOOP1.T(90) := 959.298 ; # 514
PLANT.LOOP1.T(91) := 946.119 ; # 515
PLANT.LOOP1.T(92) := 932.173 ; # 516
PLANT.LOOP1.T(93) := 917.416 ; # 517
PLANT.LOOP1.T(94) := 901.801 ; # 518
PLANT.LOOP1.T(95) := 885.278 ; # 519
PLANT.LOOP1.T(96) := 867.793 ; # 520
PLANT.LOOP1.T(97) := 849.292 ; # 521
PLANT.LOOP1.T(98) := 829.715 ; # 522
PLANT.LOOP1.T(99) := 809 ; # 523
PLANT.LOOP1.T(100) := 787.081 ; # 524
PLANT.LOOP1.T0 := 839.77 ; # 525
PLANT.LOOP1.TEXTRNL(1) := 291.15 ; # 526
PLANT.LOOP1.TEXTRNL(2) := 291.15 ; # 527
PLANT.LOOP1.TEXTRNL(3) := 291.15 ; # 528
PLANT.LOOP1.TEXTRNL(4) := 291.15 ; # 529
PLANT.LOOP1.TEXTRNL(5) := 291.15 ; # 530
PLANT.LOOP1.TEXTRNL(6) := 291.15 ; # 531
PLANT.LOOP1.TEXTRNL(7) := 291.15 ; # 532
PLANT.LOOP1.TEXTRNL(8) := 291.15 ; # 533
PLANT.LOOP1.TEXTRNL(9) := 291.15 ; # 534
PLANT.LOOP1.TEXTRNL(10) := 291.15 ; # 535
PLANT.LOOP1.TEXTRNL(11) := 879.893 ; # 536
PLANT.LOOP1.TEXTRNL(12) := 892.184 ; # 537
PLANT.LOOP1.TEXTRNL(13) := 904.161 ; # 538
PLANT.LOOP1.TEXTRNL(14) := 915.766 ; # 539
PLANT.LOOP1.TEXTRNL(15) := 927.009 ; # 540
PLANT.LOOP1.TEXTRNL(16) := 937.904 ; # 541
PLANT.LOOP1.TEXTRNL(17) := 948.459 ; # 542
PLANT.LOOP1.TEXTRNL(18) := 958.687 ; # 543
PLANT.LOOP1.TEXTRNL(19) := 968.596 ; # 544
PLANT.LOOP1.TEXTRNL(20) := 978.197 ; # 545
PLANT.LOOP1.TEXTRNL(21) := 987.5 ; # 546
PLANT.LOOP1.TEXTRNL(22) := 996.513 ; # 547
PLANT.LOOP1.TEXTRNL(23) := 1005.25 ; # 548
PLANT.LOOP1.TEXTRNL(24) := 1013.71 ; # 549
PLANT.LOOP1.TEXTRNL(25) := 1021.91 ; # 550
PLANT.LOOP1.TEXTRNL(26) := 1029.85 ; # 551
PLANT.LOOP1.TEXTRNL(27) := 1037.55 ; # 552
PLANT.LOOP1.TEXTRNL(28) := 1045 ; # 553
PLANT.LOOP1.TEXTRNL(29) := 1052.23 ; # 554
PLANT.LOOP1.TEXTRNL(30) := 1059.23 ; # 555
PLANT.LOOP1.TEXTRNL(31) := 1066.01 ; # 556
PLANT.LOOP1.TEXTRNL(32) := 1072.58 ; # 557
PLANT.LOOP1.TEXTRNL(33) := 1078.95 ; # 558
PLANT.LOOP1.TEXTRNL(34) := 1085.12 ; # 559
PLANT.LOOP1.TEXTRNL(35) := 1091.1 ; # 560
PLANT.LOOP1.TEXTRNL(36) := 1096.89 ; # 561
PLANT.LOOP1.TEXTRNL(37) := 1102.5 ; # 562
PLANT.LOOP1.TEXTRNL(38) := 1107.94 ; # 563
PLANT.LOOP1.TEXTRNL(39) := 1113.21 ; # 564
PLANT.LOOP1.TEXTRNL(40) := 1118.31 ; # 565
PLANT.LOOP1.TEXTRNL(41) := 1123.26 ; # 566
PLANT.LOOP1.TEXTRNL(42) := 1128.05 ; # 567
PLANT.LOOP1.TEXTRNL(43) := 1132.69 ; # 568
PLANT.LOOP1.TEXTRNL(44) := 1137.19 ; # 569
PLANT.LOOP1.TEXTRNL(45) := 1141.55 ; # 570
PLANT.LOOP1.TEXTRNL(46) := 1145.77 ; # 571
PLANT.LOOP1.TEXTRNL(47) := 1149.86 ; # 572
PLANT.LOOP1.TEXTRNL(48) := 1153.83 ; # 573
PLANT.LOOP1.TEXTRNL(49) := 1157.67 ; # 574
```

```
PLANT.LOOP1.TEXTRNL(50) := 1161.37 ; # 575
PLANT.LOOP1.TEXTRNL(51) := 291.15 ; # 576
PLANT.LOOP1.TEXTRNL(52) := 291.15 ; # 577
PLANT.LOOP1.TEXTRNL(53) := 291.15 ; # 578
PLANT.LOOP1.TEXTRNL(54) := 291.15 ; # 579
PLANT.LOOP1.TEXTRNL(55) := 291.15 ; # 580
PLANT.LOOP1.TEXTRNL(56) := 291.15 ; # 581
PLANT.LOOP1.TEXTRNL(57) := 291.15 ; # 582
PLANT.LOOP1.TEXTRNL(58) := 291.15 ; # 583
PLANT.LOOP1.TEXTRNL(59) := 291.15 ; # 584
PLANT.LOOP1.TEXTRNL(60) := 291.15 ; # 585
PLANT.LOOP1.TEXTRNL(61) := 1133.71 ; # 586
PLANT.LOOP1.TEXTRNL(62) := 1130.67 ; # 587
PLANT.LOOP1.TEXTRNL(63) := 1127.46 ; # 588
PLANT.LOOP1.TEXTRNL(64) := 1124.06 ; # 589
PLANT.LOOP1.TEXTRNL(65) := 1120.46 ; # 590
PLANT.LOOP1.TEXTRNL(66) := 1116.65 ; # 591
PLANT.LOOP1.TEXTRNL(67) := 1112.63 ; # 592
PLANT.LOOP1.TEXTRNL(68) := 1108.36 ; # 593
PLANT.LOOP1.TEXTRNL(69) := 1103.85 ; # 594
PLANT.LOOP1.TEXTRNL(70) := 1099.08 ; # 595
PLANT.LOOP1.TEXTRNL(71) := 1094.03 ; # 596
PLANT.LOOP1.TEXTRNL(72) := 1088.69 ; # 597
PLANT.LOOP1.TEXTRNL(73) := 1083.03 ; # 598
PLANT.LOOP1.TEXTRNL(74) := 1077.05 ; # 599
PLANT.LOOP1.TEXTRNL(75) := 1070.72 ; # 600
PLANT.LOOP1.TEXTRNL(76) := 1064.02 ; # 601
PLANT.LOOP1.TEXTRNL(77) := 1056.93 ; # 602
PLANT.LOOP1.TEXTRNL(78) := 1049.43 ; # 603
PLANT.LOOP1.TEXTRNL(79) := 1041.49 ; # 604
PLANT.LOOP1.TEXTRNL(80) := 1033.09 ; # 605
PLANT.LOOP1.TEXTRNL(81) := 1024.2 ; # 606
PLANT.LOOP1.TEXTRNL(82) := 1014.8 ; # 607
PLANT.LOOP1.TEXTRNL(83) := 1004.84 ; # 608
PLANT.LOOP1.TEXTRNL(84) := 994.314 ; # 609
PLANT.LOOP1.TEXTRNL(85) := 983.171 ; # 610
PLANT.LOOP1.TEXTRNL(86) := 971.38 ; # 611
PLANT.LOOP1.TEXTRNL(87) := 958.903 ; # 612
PLANT.LOOP1.TEXTRNL(88) := 945.701 ; # 613
PLANT.LOOP1.TEXTRNL(89) := 931.731 ; # 614
PLANT.LOOP1.TEXTRNL(90) := 916.948 ; # 615
PLANT.LOOP1.TEXTRNL(91) := 901.306 ; # 616
PLANT.LOOP1.TEXTRNL(92) := 884.754 ; # 617
PLANT.LOOP1.TEXTRNL(93) := 867.239 ; # 618
PLANT.LOOP1.TEXTRNL(94) := 848.706 ; # 619
PLANT.LOOP1.TEXTRNL(95) := 829.095 ; # 620
PLANT.LOOP1.TEXTRNL(96) := 808.344 ; # 621
PLANT.LOOP1.TEXTRNL(97) := 786.385 ; # 622
PLANT.LOOP1.TEXTRNL(98) := 763.15 ; # 623
PLANT.LOOP1.TEXTRNL(99) := 738.563 ; # 624
PLANT.LOOP1.TEXTRNL(100) := 712.554 ; # 625
PLANT.LOOP1.PSI(1) := 1.23127e+006 ; # 626
PLANT.LOOP1.PSI(2) := 1.22711e+006 ; # 627
PLANT.LOOP1.PSI(3) := 1.22294e+006 ; # 628
PLANT.LOOP1.PSI(4) := 1.21877e+006 ; # 629
PLANT.LOOP1.PSI(5) := 1.2146e+006 ; # 630
PLANT.LOOP1.PSI(6) := 1.21042e+006 ; # 631
PLANT.LOOP1.PSI(7) := 1.20624e+006 ; # 632
PLANT.LOOP1.PSI(8) := 1.20206e+006 ; # 633
PLANT.LOOP1.PSI(9) := 1.19787e+006 ; # 634
PLANT.LOOP1.PSI(10) := 1.19368e+006 ; # 635
PLANT.LOOP1.PSI(11) := 1.23208e+006 ; # 636
PLANT.LOOP1.PSI(12) := 1.25743e+006 ; # 637
PLANT.LOOP1.PSI(13) := 1.28069e+006 ; # 638
PLANT.LOOP1.PSI(14) := 1.30237e+006 ; # 639
PLANT.LOOP1.PSI(15) := 1.32259e+006 ; # 640
PLANT.LOOP1.PSI(16) := 1.34148e+006 ; # 641
PLANT.LOOP1.PSI(17) := 1.35911e+006 ; # 642
PLANT.LOOP1.PSI(18) := 1.37558e+006 ; # 643
PLANT.LOOP1.PSI(19) := 1.39098e+006 ; # 644
PLANT.LOOP1.PSI(20) := 1.40537e+006 ; # 645
PLANT.LOOP1.PSI(21) := 1.41883e+006 ; # 646
PLANT.LOOP1.PSI(22) := 1.4314e+006 ; # 647
PLANT.LOOP1.PSI(23) := 1.44316e+006 ; # 648
PLANT.LOOP1.PSI(24) := 1.45415e+006 ; # 649
PLANT.LOOP1.PSI(25) := 1.46441e+006 ; # 650
PLANT.LOOP1.PSI(26) := 1.474e+006 ; # 651
PLANT.LOOP1.PSI(27) := 1.48294e+006 ; # 652
PLANT.LOOP1.PSI(28) := 1.49129e+006 ; # 653
PLANT.LOOP1.PSI(29) := 1.49907e+006 ; # 654
PLANT.LOOP1.PSI(30) := 1.50631e+006 ; # 655
PLANT.LOOP1.PSI(31) := 1.51304e+006 ; # 656
PLANT.LOOP1.PSI(32) := 1.5193e+006 ; # 657
PLANT.LOOP1.PSI(33) := 1.5251e+006 ; # 658
PLANT.LOOP1.PSI(34) := 1.53047e+006 ; # 659
PLANT.LOOP1.PSI(35) := 1.53543e+006 ; # 660
PLANT.LOOP1.PSI(36) := 1.54001e+006 ; # 661
PLANT.LOOP1.PSI(37) := 1.54421e+006 ; # 662
PLANT.LOOP1.PSI(38) := 1.54807e+006 ; # 663
PLANT.LOOP1.PSI(39) := 1.5516e+006 ; # 664
PLANT.LOOP1.PSI(40) := 1.5548e+006 ; # 665
PLANT.LOOP1.PSI(41) := 1.55771e+006 ; # 666
```

```
669   PLANT.LOOP1.PSI(42) := 1.56032e+006 ; # 667
670   PLANT.LOOP1.PSI(43) := 1.56266e+006 ; # 668
671   PLANT.LOOP1.PSI(44) := 1.56474e+006 ; # 669
672   PLANT.LOOP1.PSI(45) := 1.56657e+006 ; # 670
673   PLANT.LOOP1.PSI(46) := 1.56816e+006 ; # 671
674   PLANT.LOOP1.PSI(47) := 1.56951e+006 ; # 672
675   PLANT.LOOP1.PSI(48) := 1.57065e+006 ; # 673
676   PLANT.LOOP1.PSI(49) := 1.57158e+006 ; # 674
677   PLANT.LOOP1.PSI(50) := 1.5723e+006 ; # 675
678   PLANT.LOOP1.PSI(51) := 1.54707e+006 ; # 676
679   PLANT.LOOP1.PSI(52) := 1.54135e+006 ; # 677
680   PLANT.LOOP1.PSI(53) := 1.53567e+006 ; # 678
681   PLANT.LOOP1.PSI(54) := 1.52997e+006 ; # 679
682   PLANT.LOOP1.PSI(55) := 1.52427e+006 ; # 680
683   PLANT.LOOP1.PSI(56) := 1.51856e+006 ; # 681
684   PLANT.LOOP1.PSI(57) := 1.51284e+006 ; # 682
685   PLANT.LOOP1.PSI(58) := 1.50712e+006 ; # 683
686   PLANT.LOOP1.PSI(59) := 1.50138e+006 ; # 684
687   PLANT.LOOP1.PSI(60) := 1.49564e+006 ; # 685
688   PLANT.LOOP1.PSI(61) := 1.50921e+006 ; # 686
689   PLANT.LOOP1.PSI(62) := 1.50544e+006 ; # 687
690   PLANT.LOOP1.PSI(63) := 1.50142e+006 ; # 688
691   PLANT.LOOP1.PSI(64) := 1.49724e+006 ; # 689
692   PLANT.LOOP1.PSI(65) := 1.49288e+006 ; # 690
693   PLANT.LOOP1.PSI(66) := 1.48832e+006 ; # 691
694   PLANT.LOOP1.PSI(67) := 1.48357e+006 ; # 692
695   PLANT.LOOP1.PSI(68) := 1.47859e+006 ; # 693
696   PLANT.LOOP1.PSI(69) := 1.47338e+006 ; # 694
697   PLANT.LOOP1.PSI(70) := 1.46791e+006 ; # 695
698   PLANT.LOOP1.PSI(71) := 1.46218e+006 ; # 696
699   PLANT.LOOP1.PSI(72) := 1.45616e+006 ; # 697
700   PLANT.LOOP1.PSI(73) := 1.44982e+006 ; # 698
701   PLANT.LOOP1.PSI(74) := 1.44315e+006 ; # 699
702   PLANT.LOOP1.PSI(75) := 1.43612e+006 ; # 700
703   PLANT.LOOP1.PSI(76) := 1.4287e+006 ; # 701
704   PLANT.LOOP1.PSI(77) := 1.42087e+006 ; # 702
705   PLANT.LOOP1.PSI(78) := 1.41258e+006 ; # 703
706   PLANT.LOOP1.PSI(79) := 1.40379e+006 ; # 704
707   PLANT.LOOP1.PSI(80) := 1.39448e+006 ; # 705
708   PLANT.LOOP1.PSI(81) := 1.38459e+006 ; # 706
709   PLANT.LOOP1.PSI(82) := 1.37407e+006 ; # 707
710   PLANT.LOOP1.PSI(83) := 1.36287e+006 ; # 708
711   PLANT.LOOP1.PSI(84) := 1.35093e+006 ; # 709
712   PLANT.LOOP1.PSI(85) := 1.33817e+006 ; # 710
713   PLANT.LOOP1.PSI(86) := 1.32451e+006 ; # 711
714   PLANT.LOOP1.PSI(87) := 1.30988e+006 ; # 712
715   PLANT.LOOP1.PSI(88) := 1.29416e+006 ; # 713
716   PLANT.LOOP1.PSI(89) := 1.27725e+006 ; # 714
717   PLANT.LOOP1.PSI(90) := 1.25903e+006 ; # 715
718   PLANT.LOOP1.PSI(91) := 1.23934e+006 ; # 716
719   PLANT.LOOP1.PSI(92) := 1.21801e+006 ; # 717
720   PLANT.LOOP1.PSI(93) := 1.19486e+006 ; # 718
721   PLANT.LOOP1.PSI(94) := 1.16967e+006 ; # 719
722   PLANT.LOOP1.PSI(95) := 1.14217e+006 ; # 720
723   PLANT.LOOP1.PSI(96) := 1.11205e+006 ; # 721
724   PLANT.LOOP1.PSI(97) := 1.07896e+006 ; # 722
725   PLANT.LOOP1.PSI(98) := 1.04248e+006 ; # 723
726   PLANT.LOOP1.PSI(99) := 1.00208e+006 ; # 724
727   PLANT.LOOP1.PSI(100) := 957155 ; # 725
728   PLANT.LOOP1.F(1) := 26.308 ; # 726
729   PLANT.LOOP1.F(2) := 26.308 ; # 727
730   PLANT.LOOP1.F(3) := 26.308 ; # 728
731   PLANT.LOOP1.F(4) := 26.308 ; # 729
732   PLANT.LOOP1.F(5) := 26.308 ; # 730
733   PLANT.LOOP1.F(6) := 26.308 ; # 731
734   PLANT.LOOP1.F(7) := 26.308 ; # 732
735   PLANT.LOOP1.F(8) := 26.308 ; # 733
736   PLANT.LOOP1.F(9) := 26.308 ; # 734
737   PLANT.LOOP1.F(10) := 26.308 ; # 735
738   PLANT.LOOP1.F(11) := 26.308 ; # 736
739   PLANT.LOOP1.F(12) := 26.308 ; # 737
740   PLANT.LOOP1.F(13) := 26.308 ; # 738
741   PLANT.LOOP1.F(14) := 26.308 ; # 739
742   PLANT.LOOP1.F(15) := 26.308 ; # 740
743   PLANT.LOOP1.F(16) := 26.308 ; # 741
744   PLANT.LOOP1.F(17) := 26.308 ; # 742
745   PLANT.LOOP1.F(18) := 26.308 ; # 743
746   PLANT.LOOP1.F(19) := 26.308 ; # 744
747   PLANT.LOOP1.F(20) := 26.308 ; # 745
748   PLANT.LOOP1.F(21) := 26.308 ; # 746
749   PLANT.LOOP1.F(22) := 26.308 ; # 747
750   PLANT.LOOP1.F(23) := 26.308 ; # 748
751   PLANT.LOOP1.F(24) := 26.308 ; # 749
752   PLANT.LOOP1.F(25) := 26.308 ; # 750
753   PLANT.LOOP1.F(26) := 26.308 ; # 751
754   PLANT.LOOP1.F(27) := 26.308 ; # 752
755   PLANT.LOOP1.F(28) := 26.308 ; # 753
756   PLANT.LOOP1.F(29) := 26.308 ; # 754
757   PLANT.LOOP1.F(30) := 26.308 ; # 755
758   PLANT.LOOP1.F(31) := 26.308 ; # 756
759   PLANT.LOOP1.F(32) := 26.308 ; # 757
760   PLANT.LOOP1.F(33) := 26.308 ; # 758
```

```
761   PLANT.LOOP1.F(34) := 26.308 ; # 759
762   PLANT.LOOP1.F(35) := 26.308 ; # 760
763   PLANT.LOOP1.F(36) := 26.308 ; # 761
764   PLANT.LOOP1.F(37) := 26.308 ; # 762
765   PLANT.LOOP1.F(38) := 26.308 ; # 763
766   PLANT.LOOP1.F(39) := 26.308 ; # 764
767   PLANT.LOOP1.F(40) := 26.308 ; # 765
768   PLANT.LOOP1.F(41) := 26.308 ; # 766
769   PLANT.LOOP1.F(42) := 26.308 ; # 767
770   PLANT.LOOP1.F(43) := 26.308 ; # 768
771   PLANT.LOOP1.F(44) := 26.308 ; # 769
772   PLANT.LOOP1.F(45) := 26.308 ; # 770
773   PLANT.LOOP1.F(46) := 26.308 ; # 771
774   PLANT.LOOP1.F(47) := 26.308 ; # 772
775   PLANT.LOOP1.F(48) := 26.308 ; # 773
776   PLANT.LOOP1.F(49) := 26.308 ; # 774
777   PLANT.LOOP1.F(50) := 26.308 ; # 775
778   PLANT.LOOP1.F(51) := 26.308 ; # 776
779   PLANT.LOOP1.F(52) := 26.308 ; # 777
780   PLANT.LOOP1.F(53) := 26.308 ; # 778
781   PLANT.LOOP1.F(54) := 26.308 ; # 779
782   PLANT.LOOP1.F(55) := 26.308 ; # 780
783   PLANT.LOOP1.F(56) := 26.308 ; # 781
784   PLANT.LOOP1.F(57) := 26.308 ; # 782
785   PLANT.LOOP1.F(58) := 26.308 ; # 783
786   PLANT.LOOP1.F(59) := 26.308 ; # 784
787   PLANT.LOOP1.F(60) := 26.308 ; # 785
788   PLANT.LOOP1.F(61) := 26.308 ; # 786
789   PLANT.LOOP1.F(62) := 26.308 ; # 787
790   PLANT.LOOP1.F(63) := 26.308 ; # 788
791   PLANT.LOOP1.F(64) := 26.308 ; # 789
792   PLANT.LOOP1.F(65) := 26.308 ; # 790
793   PLANT.LOOP1.F(66) := 26.308 ; # 791
794   PLANT.LOOP1.F(67) := 26.308 ; # 792
795   PLANT.LOOP1.F(68) := 26.308 ; # 793
796   PLANT.LOOP1.F(69) := 26.308 ; # 794
797   PLANT.LOOP1.F(70) := 26.308 ; # 795
798   PLANT.LOOP1.F(71) := 26.308 ; # 796
799   PLANT.LOOP1.F(72) := 26.308 ; # 797
800   PLANT.LOOP1.F(73) := 26.308 ; # 798
801   PLANT.LOOP1.F(74) := 26.308 ; # 799
802   PLANT.LOOP1.F(75) := 26.308 ; # 800
803   PLANT.LOOP1.F(76) := 26.308 ; # 801
804   PLANT.LOOP1.F(77) := 26.308 ; # 802
805   PLANT.LOOP1.F(78) := 26.308 ; # 803
806   PLANT.LOOP1.F(79) := 26.308 ; # 804
807   PLANT.LOOP1.F(80) := 26.308 ; # 805
808   PLANT.LOOP1.F(81) := 26.308 ; # 806
809   PLANT.LOOP1.F(82) := 26.308 ; # 807
810   PLANT.LOOP1.F(83) := 26.308 ; # 808
811   PLANT.LOOP1.F(84) := 26.308 ; # 809
812   PLANT.LOOP1.F(85) := 26.308 ; # 810
813   PLANT.LOOP1.F(86) := 26.308 ; # 811
814   PLANT.LOOP1.F(87) := 26.308 ; # 812
815   PLANT.LOOP1.F(88) := 26.308 ; # 813
816   PLANT.LOOP1.F(89) := 26.308 ; # 814
817   PLANT.LOOP1.F(90) := 26.308 ; # 815
818   PLANT.LOOP1.F(91) := 26.308 ; # 816
819   PLANT.LOOP1.F(92) := 26.308 ; # 817
820   PLANT.LOOP1.F(93) := 26.308 ; # 818
821   PLANT.LOOP1.F(94) := 26.308 ; # 819
822   PLANT.LOOP1.F(95) := 26.308 ; # 820
823   PLANT.LOOP1.F(96) := 26.308 ; # 821
824   PLANT.LOOP1.F(97) := 26.308 ; # 822
825   PLANT.LOOP1.F(98) := 26.308 ; # 823
826   PLANT.LOOP1.F(99) := 26.308 ; # 824
827   PLANT.LOOP1.F(100) := 26.308 ; # 825
828   PLANT.LOOP1.F_IN := 26.308 ; # 826
829   PLANT.LOOP1.RHO_AVG := 0.919789 ; # 827
830   PLANT.LOOP1.TOTALMASS := 27.4242 ; # 828
831   PLANT.LOOP1.RHOEXTRA(1) := 1.1654 ; # 829
832   PLANT.LOOP1.RHOEXTRA(2) := 1.1654 ; # 830
833   PLANT.LOOP1.PEXTRA(1) := 2.03415e+006 ; # 831
834   PLANT.LOOP1.PEXTRA(2) := 2.03415e+006 ; # 832
835   PLANT.LOOP1.HEXTRA(1) := 2.81449e+006 ; # 833
836   PLANT.LOOP1.HEXTRA(2) := 2.81449e+006 ; # 834
837   PLANT.LOOP1.TEXTRA(1) := 839.77 ; # 835
838   PLANT.LOOP1.TEXTRA(2) := 839.77 ; # 836
839   PLANT.LOOP1.PSIEXTRA(1) := 1.24585e+006 ; # 837
840   PLANT.LOOP1.PSIEXTRA(2) := 1.24585e+006 ; # 838
841   PLANT.LOOP1.FEXTRA(1) := 26.308 ; # 839
842   PLANT.LOOP1.FEXTRA(2) := 26.308 ; # 840
843   PLANT.LOOP1.HT1.H2 := 17.344 ; # 841
844   PLANT.LOOP1.HT1.RHOPIPEHE := 1.14182 ; # 842
845   PLANT.LOOP1.HT1.VPIPE := 171.996 ; # 843
846   PLANT.LOOP1.HT1.H0 := 1500.09 ; # 844
847   PLANT.LOOP1.HT1.U := 8.93421 ; # 845
848   PLANT.LOOP1.HT2.RHOPIPEHE := 0.902801 ; # 846
849   PLANT.LOOP1.HT2.VPIPE := 84.86 ; # 847
850   PLANT.LOOP1.HT2.H0 := 1665.62 ; # 848
851   PLANT.LOOP1.HT2.U := 1645.45 ; # 849
852   PLANT.LOOP1.HT3.H2 := 17.344 ; # 850
```

```
853   PLANT.LOOP1.HT3.RHOPIPEHE := 0.749559 ; # 851
854   PLANT.LOOP1.HT3.VPIPE := 226.698 ; # 852
855   PLANT.LOOP1.HT3.H0 := 1316.87 ; # 853
856   PLANT.LOOP1.HT3.U := 10.3572 ; # 854
857   PLANT.LOOP1.HT4.RHOPIPEHE := 0.833125 ; # 855
858   PLANT.LOOP1.HT4.VPIPE := 79.4405 ; # 856
859   PLANT.LOOP1.HT4.U := 1299.78 ; # 857
860   PLANT.PIPEH.RHO(1) := 2.87869 ; # 858
861   PLANT.PIPEH.RHO(2) := 2.88664 ; # 859
862   PLANT.PIPEH.RHO(3) := 2.89491 ; # 860
863   PLANT.PIPEH.RHO(4) := 2.90353 ; # 861
864   PLANT.PIPEH.RHO(5) := 2.9125 ; # 862
865   PLANT.PIPEH.RHO(6) := 2.92184 ; # 863
866   PLANT.PIPEH.RHO(7) := 2.93156 ; # 864
867   PLANT.PIPEH.RHO(8) := 2.9417 ; # 865
868   PLANT.PIPEH.RHO(9) := 2.95225 ; # 866
869   PLANT.PIPEH.RHO(10) := 2.96325 ; # 867
870   PLANT.PIPEH.RHO(11) := 2.97471 ; # 868
871   PLANT.PIPEH.RHO(12) := 2.98666 ; # 869
872   PLANT.PIPEH.RHO(13) := 2.99912 ; # 870
873   PLANT.PIPEH.RHO(14) := 3.01211 ; # 871
874   PLANT.PIPEH.RHO(15) := 3.02567 ; # 872
875   PLANT.PIPEH.RHO(16) := 3.03982 ; # 873
876   PLANT.PIPEH.RHO(17) := 3.05458 ; # 874
877   PLANT.PIPEH.RHO(18) := 3.07 ; # 875
878   PLANT.PIPEH.RHO(19) := 3.08611 ; # 876
879   PLANT.PIPEH.RHO(20) := 3.10294 ; # 877
880   PLANT.PIPEH.RHO(21) := 3.12053 ; # 878
881   PLANT.PIPEH.RHO(22) := 3.13892 ; # 879
882   PLANT.PIPEH.RHO(23) := 3.15816 ; # 880
883   PLANT.PIPEH.RHO(24) := 3.1783 ; # 881
884   PLANT.PIPEH.RHO(25) := 3.19939 ; # 882
885   PLANT.PIPEH.RHO(26) := 3.22148 ; # 883
886   PLANT.PIPEH.RHO(27) := 3.24463 ; # 884
887   PLANT.PIPEH.RHO(28) := 3.2689 ; # 885
888   PLANT.PIPEH.RHO(29) := 3.29437 ; # 886
889   PLANT.PIPEH.RHO(30) := 3.32112 ; # 887
890   PLANT.PIPEH.RHO(31) := 3.34921 ; # 888
891   PLANT.PIPEH.RHO(32) := 3.37874 ; # 889
892   PLANT.PIPEH.RHO(33) := 3.4098 ; # 890
893   PLANT.PIPEH.RHO(34) := 3.4425 ; # 891
894   PLANT.PIPEH.RHO(35) := 3.47695 ; # 892
895   PLANT.PIPEH.RHO(36) := 3.51328 ; # 893
896   PLANT.PIPEH.RHO(37) := 3.55161 ; # 894
897   PLANT.PIPEH.RHO(38) := 3.59209 ; # 895
898   PLANT.PIPEH.RHO(39) := 3.6349 ; # 896
899   PLANT.PIPEH.RHO(40) := 3.68012 ; # 897
900   PLANT.PIPEH.P(1) := 6.99931e+006 ; # 898
901   PLANT.PIPEH.P(2) := 6.99793e+006 ; # 899
902   PLANT.PIPEH.P(3) := 6.99656e+006 ; # 900
903   PLANT.PIPEH.P(4) := 6.99519e+006 ; # 901
904   PLANT.PIPEH.P(5) := 6.99383e+006 ; # 902
905   PLANT.PIPEH.P(6) := 6.99247e+006 ; # 903
906   PLANT.PIPEH.P(7) := 6.99111e+006 ; # 904
907   PLANT.PIPEH.P(8) := 6.98977e+006 ; # 905
908   PLANT.PIPEH.P(9) := 6.98842e+006 ; # 906
909   PLANT.PIPEH.P(10) := 6.98708e+006 ; # 907
910   PLANT.PIPEH.P(11) := 6.98575e+006 ; # 908
911   PLANT.PIPEH.P(12) := 6.98442e+006 ; # 909
912   PLANT.PIPEH.P(13) := 6.9831e+006 ; # 910
913   PLANT.PIPEH.P(14) := 6.98179e+006 ; # 911
914   PLANT.PIPEH.P(15) := 6.98048e+006 ; # 912
915   PLANT.PIPEH.P(16) := 6.97917e+006 ; # 913
916   PLANT.PIPEH.P(17) := 6.97788e+006 ; # 914
917   PLANT.PIPEH.P(18) := 6.97659e+006 ; # 915
918   PLANT.PIPEH.P(19) := 6.97531e+006 ; # 916
919   PLANT.PIPEH.P(20) := 6.97403e+006 ; # 917
920   PLANT.PIPEH.P(21) := 6.97276e+006 ; # 918
921   PLANT.PIPEH.P(22) := 6.97151e+006 ; # 919
922   PLANT.PIPEH.P(23) := 6.97025e+006 ; # 920
923   PLANT.PIPEH.P(24) := 6.96901e+006 ; # 921
924   PLANT.PIPEH.P(25) := 6.96778e+006 ; # 922
925   PLANT.PIPEH.P(26) := 6.96655e+006 ; # 923
926   PLANT.PIPEH.P(27) := 6.96534e+006 ; # 924
927   PLANT.PIPEH.P(28) := 6.96413e+006 ; # 925
928   PLANT.PIPEH.P(29) := 6.96294e+006 ; # 926
929   PLANT.PIPEH.P(30) := 6.96175e+006 ; # 927
930   PLANT.PIPEH.P(31) := 6.96058e+006 ; # 928
931   PLANT.PIPEH.P(32) := 6.95941e+006 ; # 929
932   PLANT.PIPEH.P(33) := 6.95826e+006 ; # 930
933   PLANT.PIPEH.P(34) := 6.95712e+006 ; # 931
934   PLANT.PIPEH.P(35) := 6.95599e+006 ; # 932
935   PLANT.PIPEH.P(36) := 6.95487e+006 ; # 933
936   PLANT.PIPEH.P(37) := 6.95377e+006 ; # 934
937   PLANT.PIPEH.P(38) := 6.95268e+006 ; # 935
938   PLANT.PIPEH.P(39) := 6.9516e+006 ; # 936
939   PLANT.PIPEH.P(40) := 6.95054e+006 ; # 937
940   PLANT.PIPEH.H(1) := 4.52899e+006 ; # 938
941   PLANT.PIPEH.H(2) := 4.51106e+006 ; # 939
942   PLANT.PIPEH.H(3) := 4.49255e+006 ; # 940
943   PLANT.PIPEH.H(4) := 4.47345e+006 ; # 941
944   PLANT.PIPEH.H(5) := 4.45373e+006 ; # 942
```

```
945   PLANT.PIPEH.H(6) := 4.43339e+006 ; # 943
946   PLANT.PIPEH.H(7) := 4.41239e+006 ; # 944
947   PLANT.PIPEH.H(8) := 4.39071e+006 ; # 945
948   PLANT.PIPEH.H(9) := 4.36834e+006 ; # 946
949   PLANT.PIPEH.H(10) := 4.34525e+006 ; # 947
950   PLANT.PIPEH.H(11) := 4.32143e+006 ; # 948
951   PLANT.PIPEH.H(12) := 4.29683e+006 ; # 949
952   PLANT.PIPEH.H(13) := 4.27145e+006 ; # 950
953   PLANT.PIPEH.H(14) := 4.24525e+006 ; # 951
954   PLANT.PIPEH.H(15) := 4.21821e+006 ; # 952
955   PLANT.PIPEH.H(16) := 4.19031e+006 ; # 953
956   PLANT.PIPEH.H(17) := 4.16151e+006 ; # 954
957   PLANT.PIPEH.H(18) := 4.13178e+006 ; # 955
958   PLANT.PIPEH.H(19) := 4.1011e+006 ; # 956
959   PLANT.PIPEH.H(20) := 4.06944e+006 ; # 957
960   PLANT.PIPEH.H(21) := 4.03675e+006 ; # 958
961   PLANT.PIPEH.H(22) := 4.00302e+006 ; # 959
962   PLANT.PIPEH.H(23) := 3.96821e+006 ; # 960
963   PLANT.PIPEH.H(24) := 3.93228e+006 ; # 961
964   PLANT.PIPEH.H(25) := 3.8952e+006 ; # 962
965   PLANT.PIPEH.H(26) := 3.85693e+006 ; # 963
966   PLANT.PIPEH.H(27) := 3.81742e+006 ; # 964
967   PLANT.PIPEH.H(28) := 3.77666e+006 ; # 965
968   PLANT.PIPEH.H(29) := 3.73458e+006 ; # 966
969   PLANT.PIPEH.H(30) := 3.69115e+006 ; # 967
970   PLANT.PIPEH.H(31) := 3.64633e+006 ; # 968
971   PLANT.PIPEH.H(32) := 3.60007e+006 ; # 969
972   PLANT.PIPEH.H(33) := 3.55232e+006 ; # 970
973   PLANT.PIPEH.H(34) := 3.50304e+006 ; # 971
974   PLANT.PIPEH.H(35) := 3.45219e+006 ; # 972
975   PLANT.PIPEH.H(36) := 3.39969e+006 ; # 973
976   PLANT.PIPEH.H(37) := 3.34552e+006 ; # 974
977   PLANT.PIPEH.H(38) := 3.2896e+006 ; # 975
978   PLANT.PIPEH.H(39) := 3.23189e+006 ; # 976
979   PLANT.PIPEH.H(40) := 3.17244e+006 ; # 977
980   PLANT.PIPEH.H0 := 4.5464e+006 ; # 978
981   PLANT.PIPEH.T(1) := 1169.8 ; # 979
982   PLANT.PIPEH.T(2) := 1166.35 ; # 980
983   PLANT.PIPEH.T(3) := 1162.78 ; # 981
984   PLANT.PIPEH.T(4) := 1159.11 ; # 982
985   PLANT.PIPEH.T(5) := 1155.31 ; # 983
986   PLANT.PIPEH.T(6) := 1151.39 ; # 984
987   PLANT.PIPEH.T(7) := 1147.35 ; # 985
988   PLANT.PIPEH.T(8) := 1143.18 ; # 986
989   PLANT.PIPEH.T(9) := 1138.87 ; # 987
990   PLANT.PIPEH.T(10) := 1134.43 ; # 988
991   PLANT.PIPEH.T(11) := 1129.84 ; # 989
992   PLANT.PIPEH.T(12) := 1125.11 ; # 990
993   PLANT.PIPEH.T(13) := 1120.22 ; # 991
994   PLANT.PIPEH.T(14) := 1115.18 ; # 992
995   PLANT.PIPEH.T(15) := 1109.98 ; # 993
996   PLANT.PIPEH.T(16) := 1104.6 ; # 994
997   PLANT.PIPEH.T(17) := 1099.06 ; # 995
998   PLANT.PIPEH.T(18) := 1093.34 ; # 996
999   PLANT.PIPEH.T(19) := 1087.43 ; # 997
1000  PLANT.PIPEH.T(20) := 1081.34 ; # 998
1001  PLANT.PIPEH.T(21) := 1075.05 ; # 999
1002  PLANT.PIPEH.T(22) := 1068.55 ; # 1000
1003  PLANT.PIPEH.T(23) := 1061.85 ; # 1001
1004  PLANT.PIPEH.T(24) := 1054.94 ; # 1002
1005  PLANT.PIPEH.T(25) := 1047.8 ; # 1003
1006  PLANT.PIPEH.T(26) := 1040.43 ; # 1004
1007  PLANT.PIPEH.T(27) := 1032.83 ; # 1005
1008  PLANT.PIPEH.T(28) := 1024.98 ; # 1006
1009  PLANT.PIPEH.T(29) := 1016.88 ; # 1007
1010  PLANT.PIPEH.T(30) := 1008.52 ; # 1008
1011  PLANT.PIPEH.T(31) := 999.892 ; # 1009
1012  PLANT.PIPEH.T(32) := 990.987 ; # 1010
1013  PLANT.PIPEH.T(33) := 981.796 ; # 1011
1014  PLANT.PIPEH.T(34) := 972.311 ; # 1012
1015  PLANT.PIPEH.T(35) := 962.521 ; # 1013
1016  PLANT.PIPEH.T(36) := 952.416 ; # 1014
1017  PLANT.PIPEH.T(37) := 941.988 ; # 1015
1018  PLANT.PIPEH.T(38) := 931.225 ; # 1016
1019  PLANT.PIPEH.T(39) := 920.116 ; # 1017
1020  PLANT.PIPEH.T(40) := 908.671 ; # 1018
1021  PLANT.PIPEH.TEXTRNL(1) := 1161.37 ; # 1019
1022  PLANT.PIPEH.TEXTRNL(2) := 1157.67 ; # 1020
1023  PLANT.PIPEH.TEXTRNL(3) := 1153.83 ; # 1021
1024  PLANT.PIPEH.TEXTRNL(4) := 1149.86 ; # 1022
1025  PLANT.PIPEH.TEXTRNL(5) := 1145.77 ; # 1023
1026  PLANT.PIPEH.TEXTRNL(6) := 1141.55 ; # 1024
1027  PLANT.PIPEH.TEXTRNL(7) := 1137.19 ; # 1025
1028  PLANT.PIPEH.TEXTRNL(8) := 1132.69 ; # 1026
1029  PLANT.PIPEH.TEXTRNL(9) := 1128.05 ; # 1027
1030  PLANT.PIPEH.TEXTRNL(10) := 1123.26 ; # 1028
1031  PLANT.PIPEH.TEXTRNL(11) := 1118.31 ; # 1029
1032  PLANT.PIPEH.TEXTRNL(12) := 1113.21 ; # 1030
1033  PLANT.PIPEH.TEXTRNL(13) := 1107.94 ; # 1031
1034  PLANT.PIPEH.TEXTRNL(14) := 1102.5 ; # 1032
1035  PLANT.PIPEH.TEXTRNL(15) := 1096.89 ; # 1033
1036  PLANT.PIPEH.TEXTRNL(16) := 1091.1 ; # 1034
```

```
1037   PLANT.PIPEH.TEXTRNL(17) := 1085.12 ; # 1035
1038   PLANT.PIPEH.TEXTRNL(18) := 1078.95 ; # 1036
1039   PLANT.PIPEH.TEXTRNL(19) := 1072.58 ; # 1037
1040   PLANT.PIPEH.TEXTRNL(20) := 1066.01 ; # 1038
1041   PLANT.PIPEH.TEXTRNL(21) := 1059.23 ; # 1039
1042   PLANT.PIPEH.TEXTRNL(22) := 1052.23 ; # 1040
1043   PLANT.PIPEH.TEXTRNL(23) := 1045 ; # 1041
1044   PLANT.PIPEH.TEXTRNL(24) := 1037.55 ; # 1042
1045   PLANT.PIPEH.TEXTRNL(25) := 1029.85 ; # 1043
1046   PLANT.PIPEH.TEXTRNL(26) := 1021.91 ; # 1044
1047   PLANT.PIPEH.TEXTRNL(27) := 1013.71 ; # 1045
1048   PLANT.PIPEH.TEXTRNL(28) := 1005.25 ; # 1046
1049   PLANT.PIPEH.TEXTRNL(29) := 996.513 ; # 1047
1050   PLANT.PIPEH.TEXTRNL(30) := 987.5 ; # 1048
1051   PLANT.PIPEH.TEXTRNL(31) := 978.197 ; # 1049
1052   PLANT.PIPEH.TEXTRNL(32) := 968.596 ; # 1050
1053   PLANT.PIPEH.TEXTRNL(33) := 958.687 ; # 1051
1054   PLANT.PIPEH.TEXTRNL(34) := 948.459 ; # 1052
1055   PLANT.PIPEH.TEXTRNL(35) := 937.904 ; # 1053
1056   PLANT.PIPEH.TEXTRNL(36) := 927.009 ; # 1054
1057   PLANT.PIPEH.TEXTRNL(37) := 915.766 ; # 1055
1058   PLANT.PIPEH.TEXTRNL(38) := 904.161 ; # 1056
1059   PLANT.PIPEH.TEXTRNL(39) := 892.184 ; # 1057
1060   PLANT.PIPEH.TEXTRNL(40) := 879.893 ; # 1058
1061   PLANT.PIPEH.F(1) := 32.3305 ; # 1059
1062   PLANT.PIPEH.F(2) := 32.3305 ; # 1060
1063   PLANT.PIPEH.F(3) := 32.3305 ; # 1061
1064   PLANT.PIPEH.F(4) := 32.3305 ; # 1062
1065   PLANT.PIPEH.F(5) := 32.3305 ; # 1063
1066   PLANT.PIPEH.F(6) := 32.3305 ; # 1064
1067   PLANT.PIPEH.F(7) := 32.3305 ; # 1065
1068   PLANT.PIPEH.F(8) := 32.3305 ; # 1066
1069   PLANT.PIPEH.F(9) := 32.3305 ; # 1067
1070   PLANT.PIPEH.F(10) := 32.3305 ; # 1068
1071   PLANT.PIPEH.F(11) := 32.3305 ; # 1069
1072   PLANT.PIPEH.F(12) := 32.3305 ; # 1070
1073   PLANT.PIPEH.F(13) := 32.3305 ; # 1071
1074   PLANT.PIPEH.F(14) := 32.3305 ; # 1072
1075   PLANT.PIPEH.F(15) := 32.3305 ; # 1073
1076   PLANT.PIPEH.F(16) := 32.3305 ; # 1074
1077   PLANT.PIPEH.F(17) := 32.3305 ; # 1075
1078   PLANT.PIPEH.F(18) := 32.3305 ; # 1076
1079   PLANT.PIPEH.F(19) := 32.3305 ; # 1077
1080   PLANT.PIPEH.F(20) := 32.3305 ; # 1078
1081   PLANT.PIPEH.F(21) := 32.3305 ; # 1079
1082   PLANT.PIPEH.F(22) := 32.3305 ; # 1080
1083   PLANT.PIPEH.F(23) := 32.3305 ; # 1081
1084   PLANT.PIPEH.F(24) := 32.3305 ; # 1082
1085   PLANT.PIPEH.F(25) := 32.3305 ; # 1083
1086   PLANT.PIPEH.F(26) := 32.3305 ; # 1084
1087   PLANT.PIPEH.F(27) := 32.3305 ; # 1085
1088   PLANT.PIPEH.F(28) := 32.3305 ; # 1086
1089   PLANT.PIPEH.F(29) := 32.3305 ; # 1087
1090   PLANT.PIPEH.F(30) := 32.3305 ; # 1088
1091   PLANT.PIPEH.F(31) := 32.3305 ; # 1089
1092   PLANT.PIPEH.F(32) := 32.3305 ; # 1090
1093   PLANT.PIPEH.F(33) := 32.3305 ; # 1091
1094   PLANT.PIPEH.F(34) := 32.3305 ; # 1092
1095   PLANT.PIPEH.F(35) := 32.3305 ; # 1093
1096   PLANT.PIPEH.F(36) := 32.3305 ; # 1094
1097   PLANT.PIPEH.F(37) := 32.3305 ; # 1095
1098   PLANT.PIPEH.F(38) := 32.3305 ; # 1096
1099   PLANT.PIPEH.F(39) := 32.3305 ; # 1097
1100   PLANT.PIPEH.F(40) := 32.3305 ; # 1098
1101   PLANT.PIPEH.F_IN := 32.3305 ; # 1099
1102   PLANT.PIPEH.PSI(1) := 6.03823e+006 ; # 1100
1103   PLANT.PIPEH.PSI(2) := 6.02385e+006 ; # 1101
1104   PLANT.PIPEH.PSI(3) := 6.00898e+006 ; # 1102
1105   PLANT.PIPEH.PSI(4) := 5.9936e+006 ; # 1103
1106   PLANT.PIPEH.PSI(5) := 5.97767e+006 ; # 1104
1107   PLANT.PIPEH.PSI(6) := 5.96117e+006 ; # 1105
1108   PLANT.PIPEH.PSI(7) := 5.94408e+006 ; # 1106
1109   PLANT.PIPEH.PSI(8) := 5.92638e+006 ; # 1107
1110   PLANT.PIPEH.PSI(9) := 5.90802e+006 ; # 1108
1111   PLANT.PIPEH.PSI(10) := 5.88899e+006 ; # 1109
1112   PLANT.PIPEH.PSI(11) := 5.86924e+006 ; # 1110
1113   PLANT.PIPEH.PSI(12) := 5.84876e+006 ; # 1111
1114   PLANT.PIPEH.PSI(13) := 5.82749e+006 ; # 1112
1115   PLANT.PIPEH.PSI(14) := 5.8054e+006 ; # 1113
1116   PLANT.PIPEH.PSI(15) := 5.78245e+006 ; # 1114
1117   PLANT.PIPEH.PSI(16) := 5.75859e+006 ; # 1115
1118   PLANT.PIPEH.PSI(17) := 5.73379e+006 ; # 1116
1119   PLANT.PIPEH.PSI(18) := 5.70799e+006 ; # 1117
1120   PLANT.PIPEH.PSI(19) := 5.68113e+006 ; # 1118
1121   PLANT.PIPEH.PSI(20) := 5.65317e+006 ; # 1119
1122   PLANT.PIPEH.PSI(21) := 5.62404e+006 ; # 1120
1123   PLANT.PIPEH.PSI(22) := 5.59367e+006 ; # 1121
1124   PLANT.PIPEH.PSI(23) := 5.56201e+006 ; # 1122
1125   PLANT.PIPEH.PSI(24) := 5.52897e+006 ; # 1123
1126   PLANT.PIPEH.PSI(25) := 5.49447e+006 ; # 1124
1127   PLANT.PIPEH.PSI(26) := 5.45844e+006 ; # 1125
1128   PLANT.PIPEH.PSI(27) := 5.42078e+006 ; # 1126
```

```
PLANT.PIPEH.PSI(28) := 5.38139e+006 ; # 1127
PLANT.PIPEH.PSI(29) := 5.34016e+006 ; # 1128
PLANT.PIPEH.PSI(30) := 5.29699e+006 ; # 1129
PLANT.PIPEH.PSI(31) := 5.25174e+006 ; # 1130
PLANT.PIPEH.PSI(32) := 5.20427e+006 ; # 1131
PLANT.PIPEH.PSI(33) := 5.15446e+006 ; # 1132
PLANT.PIPEH.PSI(34) := 5.10212e+006 ; # 1133
PLANT.PIPEH.PSI(35) := 5.0471e+006 ; # 1134
PLANT.PIPEH.PSI(36) := 4.98919e+006 ; # 1135
PLANT.PIPEH.PSI(37) := 4.92819e+006 ; # 1136
PLANT.PIPEH.PSI(38) := 4.86388e+006 ; # 1137
PLANT.PIPEH.PSI(39) := 4.796e+006 ; # 1138
PLANT.PIPEH.PSI(40) := 4.7244e+006 ; # 1139
PLANT.PIPEH.HT.RHOPIPEHE := 3.09587 ; # 1140
PLANT.PIPEH.HT.VPIPE := 28.7109 ; # 1141
PLANT.PIPEH.HT.H0 := 1760.94 ; # 1142
PLANT.PIPEH.HT.U := 1738.41 ; # 1143
PLANT.HX2.TGASHOT(1) := 1141.93 ; # 1144
PLANT.HX2.TGASHOT(2) := 1139.37 ; # 1145
PLANT.HX2.TGASHOT(3) := 1136.66 ; # 1146
PLANT.HX2.TGASHOT(4) := 1133.8 ; # 1147
PLANT.HX2.TGASHOT(5) := 1130.77 ; # 1148
PLANT.HX2.TGASHOT(6) := 1127.56 ; # 1149
PLANT.HX2.TGASHOT(7) := 1124.17 ; # 1150
PLANT.HX2.TGASHOT(8) := 1120.57 ; # 1151
PLANT.HX2.TGASHOT(9) := 1116.77 ; # 1152
PLANT.HX2.TGASHOT(10) := 1112.75 ; # 1153
PLANT.HX2.TGASHOT(11) := 1108.5 ; # 1154
PLANT.HX2.TGASHOT(12) := 1104 ; # 1155
PLANT.HX2.TGASHOT(13) := 1099.23 ; # 1156
PLANT.HX2.TGASHOT(14) := 1094.19 ; # 1157
PLANT.HX2.TGASHOT(15) := 1088.86 ; # 1158
PLANT.HX2.TGASHOT(16) := 1083.21 ; # 1159
PLANT.HX2.TGASHOT(17) := 1077.24 ; # 1160
PLANT.HX2.TGASHOT(18) := 1070.92 ; # 1161
PLANT.HX2.TGASHOT(19) := 1064.23 ; # 1162
PLANT.HX2.TGASHOT(20) := 1057.15 ; # 1163
PLANT.HX2.TGASHOT(21) := 1049.66 ; # 1164
PLANT.HX2.TGASHOT(22) := 1041.74 ; # 1165
PLANT.HX2.TGASHOT(23) := 1033.36 ; # 1166
PLANT.HX2.TGASHOT(24) := 1024.48 ; # 1167
PLANT.HX2.TGASHOT(25) := 1015.09 ; # 1168
PLANT.HX2.TGASHOT(26) := 1005.16 ; # 1169
PLANT.HX2.TGASHOT(27) := 994.647 ; # 1170
PLANT.HX2.TGASHOT(28) := 983.524 ; # 1171
PLANT.HX2.TGASHOT(29) := 971.753 ; # 1172
PLANT.HX2.TGASHOT(30) := 959.298 ; # 1173
PLANT.HX2.TGASHOT(31) := 946.119 ; # 1174
PLANT.HX2.TGASHOT(32) := 932.173 ; # 1175
PLANT.HX2.TGASHOT(33) := 917.416 ; # 1176
PLANT.HX2.TGASHOT(34) := 901.801 ; # 1177
PLANT.HX2.TGASHOT(35) := 885.278 ; # 1178
PLANT.HX2.TGASHOT(36) := 867.793 ; # 1179
PLANT.HX2.TGASHOT(37) := 849.292 ; # 1180
PLANT.HX2.TGASHOT(38) := 829.715 ; # 1181
PLANT.HX2.TGASHOT(39) := 809 ; # 1182
PLANT.HX2.TGASHOT(40) := 787.081 ; # 1183
PLANT.HX2.TGASCOLD(1) := 1126.32 ; # 1184
PLANT.HX2.TGASCOLD(2) := 1122.86 ; # 1185
PLANT.HX2.TGASCOLD(3) := 1119.19 ; # 1186
PLANT.HX2.TGASCOLD(4) := 1115.31 ; # 1187
PLANT.HX2.TGASCOLD(5) := 1111.2 ; # 1188
PLANT.HX2.TGASCOLD(6) := 1106.86 ; # 1189
PLANT.HX2.TGASCOLD(7) := 1102.26 ; # 1190
PLANT.HX2.TGASCOLD(8) := 1097.39 ; # 1191
PLANT.HX2.TGASCOLD(9) := 1092.24 ; # 1192
PLANT.HX2.TGASCOLD(10) := 1086.8 ; # 1193
PLANT.HX2.TGASCOLD(11) := 1081.03 ; # 1194
PLANT.HX2.TGASCOLD(12) := 1074.93 ; # 1195
PLANT.HX2.TGASCOLD(13) := 1068.48 ; # 1196
PLANT.HX2.TGASCOLD(14) := 1061.65 ; # 1197
PLANT.HX2.TGASCOLD(15) := 1054.42 ; # 1198
PLANT.HX2.TGASCOLD(16) := 1046.77 ; # 1199
PLANT.HX2.TGASCOLD(17) := 1038.68 ; # 1200
PLANT.HX2.TGASCOLD(18) := 1030.12 ; # 1201
PLANT.HX2.TGASCOLD(19) := 1021.06 ; # 1202
PLANT.HX2.TGASCOLD(20) := 1011.47 ; # 1203
PLANT.HX2.TGASCOLD(21) := 1001.32 ; # 1204
PLANT.HX2.TGASCOLD(22) := 990.587 ; # 1205
PLANT.HX2.TGASCOLD(23) := 979.228 ; # 1206
PLANT.HX2.TGASCOLD(24) := 967.207 ; # 1207
PLANT.HX2.TGASCOLD(25) := 954.488 ; # 1208
PLANT.HX2.TGASCOLD(26) := 941.029 ; # 1209
PLANT.HX2.TGASCOLD(27) := 926.787 ; # 1210
PLANT.HX2.TGASCOLD(28) := 911.717 ; # 1211
PLANT.HX2.TGASCOLD(29) := 895.77 ; # 1212
PLANT.HX2.TGASCOLD(30) := 878.896 ; # 1213
PLANT.HX2.TGASCOLD(31) := 861.041 ; # 1214
PLANT.HX2.TGASCOLD(32) := 842.147 ; # 1215
PLANT.HX2.TGASCOLD(33) := 822.155 ; # 1216
PLANT.HX2.TGASCOLD(34) := 801 ; # 1217
PLANT.HX2.TGASCOLD(35) := 778.614 ; # 1218
```

```
1221   PLANT.HX2.TGASCOLD(36) := 754.927 ; # 1219
1222   PLANT.HX2.TGASCOLD(37) := 729.862 ; # 1220
1223   PLANT.HX2.TGASCOLD(38) := 703.34 ; # 1221
1224   PLANT.HX2.TGASCOLD(39) := 675.275 ; # 1222
1225   PLANT.HX2.TGASCOLD(40) := 645.577 ; # 1223
1226   PLANT.HX2.UHOT := 1299.78 ; # 1224
1227   PLANT.HX2.UCOLD := 1880.53 ; # 1225
1228   PLANT.HX2.THX(1) := 1133.71 ; # 1226
1229   PLANT.HX2.THX(2) := 1130.67 ; # 1227
1230   PLANT.HX2.THX(3) := 1127.46 ; # 1228
1231   PLANT.HX2.THX(4) := 1124.06 ; # 1229
1232   PLANT.HX2.THX(5) := 1120.46 ; # 1230
1233   PLANT.HX2.THX(6) := 1116.65 ; # 1231
1234   PLANT.HX2.THX(7) := 1112.63 ; # 1232
1235   PLANT.HX2.THX(8) := 1108.36 ; # 1233
1236   PLANT.HX2.THX(9) := 1103.85 ; # 1234
1237   PLANT.HX2.THX(10) := 1099.08 ; # 1235
1238   PLANT.HX2.THX(11) := 1094.03 ; # 1236
1239   PLANT.HX2.THX(12) := 1088.69 ; # 1237
1240   PLANT.HX2.THX(13) := 1083.03 ; # 1238
1241   PLANT.HX2.THX(14) := 1077.05 ; # 1239
1242   PLANT.HX2.THX(15) := 1070.72 ; # 1240
1243   PLANT.HX2.THX(16) := 1064.02 ; # 1241
1244   PLANT.HX2.THX(17) := 1056.93 ; # 1242
1245   PLANT.HX2.THX(18) := 1049.43 ; # 1243
1246   PLANT.HX2.THX(19) := 1041.49 ; # 1244
1247   PLANT.HX2.THX(20) := 1033.09 ; # 1245
1248   PLANT.HX2.THX(21) := 1024.2 ; # 1246
1249   PLANT.HX2.THX(22) := 1014.8 ; # 1247
1250   PLANT.HX2.THX(23) := 1004.84 ; # 1248
1251   PLANT.HX2.THX(24) := 994.314 ; # 1249
1252   PLANT.HX2.THX(25) := 983.171 ; # 1250
1253   PLANT.HX2.THX(26) := 971.38 ; # 1251
1254   PLANT.HX2.THX(27) := 958.903 ; # 1252
1255   PLANT.HX2.THX(28) := 945.701 ; # 1253
1256   PLANT.HX2.THX(29) := 931.731 ; # 1254
1257   PLANT.HX2.THX(30) := 916.948 ; # 1255
1258   PLANT.HX2.THX(31) := 901.306 ; # 1256
1259   PLANT.HX2.THX(32) := 884.754 ; # 1257
1260   PLANT.HX2.THX(33) := 867.239 ; # 1258
1261   PLANT.HX2.THX(34) := 848.706 ; # 1259
1262   PLANT.HX2.THX(35) := 829.095 ; # 1260
1263   PLANT.HX2.THX(36) := 808.344 ; # 1261
1264   PLANT.HX2.THX(37) := 786.385 ; # 1262
1265   PLANT.HX2.THX(38) := 763.15 ; # 1263
1266   PLANT.HX2.THX(39) := 738.563 ; # 1264
1267   PLANT.HX2.THX(40) := 712.554 ; # 1265
1268   PLANT.PIPEC2.RHO(1) := 3.72389 ; # 1266
1269   PLANT.PIPEC2.RHO(2) := 3.55368 ; # 1267
1270   PLANT.PIPEC2.RHO(3) := 3.4055 ; # 1268
1271   PLANT.PIPEC2.RHO(4) := 3.27543 ; # 1269
1272   PLANT.PIPEC2.RHO(5) := 3.16041 ; # 1270
1273   PLANT.PIPEC2.RHO(6) := 3.05803 ; # 1271
1274   PLANT.PIPEC2.RHO(7) := 2.96638 ; # 1272
1275   PLANT.PIPEC2.RHO(8) := 2.88388 ; # 1273
1276   PLANT.PIPEC2.RHO(9) := 2.80928 ; # 1274
1277   PLANT.PIPEC2.RHO(10) := 2.74152 ; # 1275
1278   PLANT.PIPEC2.RHO(11) := 2.67973 ; # 1276
1279   PLANT.PIPEC2.RHO(12) := 2.62316 ; # 1277
1280   PLANT.PIPEC2.RHO(13) := 2.57121 ; # 1278
1281   PLANT.PIPEC2.RHO(14) := 2.52333 ; # 1279
1282   PLANT.PIPEC2.RHO(15) := 2.47909 ; # 1280
1283   PLANT.PIPEC2.RHO(16) := 2.43808 ; # 1281
1284   PLANT.PIPEC2.RHO(17) := 2.39997 ; # 1282
1285   PLANT.PIPEC2.RHO(18) := 2.36446 ; # 1283
1286   PLANT.PIPEC2.RHO(19) := 2.3313 ; # 1284
1287   PLANT.PIPEC2.RHO(20) := 2.30026 ; # 1285
1288   PLANT.PIPEC2.RHO(21) := 2.27113 ; # 1286
1289   PLANT.PIPEC2.RHO(22) := 2.24375 ; # 1287
1290   PLANT.PIPEC2.RHO(23) := 2.21796 ; # 1288
1291   PLANT.PIPEC2.RHO(24) := 2.19361 ; # 1289
1292   PLANT.PIPEC2.RHO(25) := 2.17058 ; # 1290
1293   PLANT.PIPEC2.RHO(26) := 2.14875 ; # 1291
1294   PLANT.PIPEC2.RHO(27) := 2.12804 ; # 1292
1295   PLANT.PIPEC2.RHO(28) := 2.10834 ; # 1293
1296   PLANT.PIPEC2.RHO(29) := 2.08957 ; # 1294
1297   PLANT.PIPEC2.RHO(30) := 2.07166 ; # 1295
1298   PLANT.PIPEC2.RHO(31) := 2.05454 ; # 1296
1299   PLANT.PIPEC2.RHO(32) := 2.03814 ; # 1297
1300   PLANT.PIPEC2.RHO(33) := 2.02242 ; # 1298
1301   PLANT.PIPEC2.RHO(34) := 2.00733 ; # 1299
1302   PLANT.PIPEC2.RHO(35) := 1.9928 ; # 1300
1303   PLANT.PIPEC2.RHO(36) := 1.97881 ; # 1301
1304   PLANT.PIPEC2.RHO(37) := 1.96531 ; # 1302
1305   PLANT.PIPEC2.RHO(38) := 1.95226 ; # 1303
1306   PLANT.PIPEC2.RHO(39) := 1.93964 ; # 1304
1307   PLANT.PIPEC2.RHO(40) := 1.92741 ; # 1305
1308   PLANT.PIPEC2.P(1) := 1.10918e+006 ; # 1306
1309   PLANT.PIPEC2.P(2) := 1.10717e+006 ; # 1307
1310   PLANT.PIPEC2.P(3) := 1.1051e+006 ; # 1308
1311   PLANT.PIPEC2.P(4) := 1.10297e+006 ; # 1309
1312   PLANT.PIPEC2.P(5) := 1.10079e+006 ; # 1310
```

```
1313   PLANT.PIPEC2.P(6) := 1.09855e+006 ; # 1311
1314   PLANT.PIPEC2.P(7) := 1.09626e+006 ; # 1312
1315   PLANT.PIPEC2.P(8) := 1.09392e+006 ; # 1313
1316   PLANT.PIPEC2.P(9) := 1.09154e+006 ; # 1314
1317   PLANT.PIPEC2.P(10) := 1.08911e+006 ; # 1315
1318   PLANT.PIPEC2.P(11) := 1.08663e+006 ; # 1316
1319   PLANT.PIPEC2.P(12) := 1.08412e+006 ; # 1317
1320   PLANT.PIPEC2.P(13) := 1.08156e+006 ; # 1318
1321   PLANT.PIPEC2.P(14) := 1.07897e+006 ; # 1319
1322   PLANT.PIPEC2.P(15) := 1.07634e+006 ; # 1320
1323   PLANT.PIPEC2.P(16) := 1.07368e+006 ; # 1321
1324   PLANT.PIPEC2.P(17) := 1.07098e+006 ; # 1322
1325   PLANT.PIPEC2.P(18) := 1.06825e+006 ; # 1323
1326   PLANT.PIPEC2.P(19) := 1.06548e+006 ; # 1324
1327   PLANT.PIPEC2.P(20) := 1.06269e+006 ; # 1325
1328   PLANT.PIPEC2.P(21) := 1.05986e+006 ; # 1326
1329   PLANT.PIPEC2.P(22) := 1.05701e+006 ; # 1327
1330   PLANT.PIPEC2.P(23) := 1.05413e+006 ; # 1328
1331   PLANT.PIPEC2.P(24) := 1.05123e+006 ; # 1329
1332   PLANT.PIPEC2.P(25) := 1.04829e+006 ; # 1330
1333   PLANT.PIPEC2.P(26) := 1.04533e+006 ; # 1331
1334   PLANT.PIPEC2.P(27) := 1.04235e+006 ; # 1332
1335   PLANT.PIPEC2.P(28) := 1.03935e+006 ; # 1333
1336   PLANT.PIPEC2.P(29) := 1.03632e+006 ; # 1334
1337   PLANT.PIPEC2.P(30) := 1.03326e+006 ; # 1335
1338   PLANT.PIPEC2.P(31) := 1.03019e+006 ; # 1336
1339   PLANT.PIPEC2.P(32) := 1.02709e+006 ; # 1337
1340   PLANT.PIPEC2.P(33) := 1.02398e+006 ; # 1338
1341   PLANT.PIPEC2.P(34) := 1.02084e+006 ; # 1339
1342   PLANT.PIPEC2.P(35) := 1.01768e+006 ; # 1340
1343   PLANT.PIPEC2.P(36) := 1.0145e+006 ; # 1341
1344   PLANT.PIPEC2.P(37) := 1.0113e+006 ; # 1342
1345   PLANT.PIPEC2.P(38) := 1.00808e+006 ; # 1343
1346   PLANT.PIPEC2.P(39) := 1.00485e+006 ; # 1344
1347   PLANT.PIPEC2.P(40) := 1.00159e+006 ; # 1345
1348   PLANT.PIPEC2.H(1) := 784075 ; # 1346
1349   PLANT.PIPEC2.H(2) := 851066 ; # 1347
1350   PLANT.PIPEC2.H(3) := 914376 ; # 1348
1351   PLANT.PIPEC2.H(4) := 974206 ; # 1349
1352   PLANT.PIPEC2.H(5) := 1.03075e+006 ; # 1350
1353   PLANT.PIPEC2.H(6) := 1.08418e+006 ; # 1351
1354   PLANT.PIPEC2.H(7) := 1.13468e+006 ; # 1352
1355   PLANT.PIPEC2.H(8) := 1.1824e+006 ; # 1353
1356   PLANT.PIPEC2.H(9) := 1.2275e+006 ; # 1354
1357   PLANT.PIPEC2.H(10) := 1.27012e+006 ; # 1355
1358   PLANT.PIPEC2.H(11) := 1.3104e+006 ; # 1356
1359   PLANT.PIPEC2.H(12) := 1.34847e+006 ; # 1357
1360   PLANT.PIPEC2.H(13) := 1.38444e+006 ; # 1358
1361   PLANT.PIPEC2.H(14) := 1.41843e+006 ; # 1359
1362   PLANT.PIPEC2.H(15) := 1.45056e+006 ; # 1360
1363   PLANT.PIPEC2.H(16) := 1.48092e+006 ; # 1361
1364   PLANT.PIPEC2.H(17) := 1.50962e+006 ; # 1362
1365   PLANT.PIPEC2.H(18) := 1.53673e+006 ; # 1363
1366   PLANT.PIPEC2.H(19) := 1.56236e+006 ; # 1364
1367   PLANT.PIPEC2.H(20) := 1.58657e+006 ; # 1365
1368   PLANT.PIPEC2.H(21) := 1.60946e+006 ; # 1366
1369   PLANT.PIPEC2.H(22) := 1.63109e+006 ; # 1367
1370   PLANT.PIPEC2.H(23) := 1.65153e+006 ; # 1368
1371   PLANT.PIPEC2.H(24) := 1.67085e+006 ; # 1369
1372   PLANT.PIPEC2.H(25) := 1.6891e+006 ; # 1370
1373   PLANT.PIPEC2.H(26) := 1.70635e+006 ; # 1371
1374   PLANT.PIPEC2.H(27) := 1.72265e+006 ; # 1372
1375   PLANT.PIPEC2.H(28) := 1.73806e+006 ; # 1373
1376   PLANT.PIPEC2.H(29) := 1.75262e+006 ; # 1374
1377   PLANT.PIPEC2.H(30) := 1.76638e+006 ; # 1375
1378   PLANT.PIPEC2.H(31) := 1.77939e+006 ; # 1376
1379   PLANT.PIPEC2.H(32) := 1.79168e+006 ; # 1377
1380   PLANT.PIPEC2.H(33) := 1.80329e+006 ; # 1378
1381   PLANT.PIPEC2.H(34) := 1.81427e+006 ; # 1379
1382   PLANT.PIPEC2.H(35) := 1.82464e+006 ; # 1380
1383   PLANT.PIPEC2.H(36) := 1.83444e+006 ; # 1381
1384   PLANT.PIPEC2.H(37) := 1.8437e+006 ; # 1382
1385   PLANT.PIPEC2.H(38) := 1.85246e+006 ; # 1383
1386   PLANT.PIPEC2.H(39) := 1.86073e+006 ; # 1384
1387   PLANT.PIPEC2.H(40) := 1.86855e+006 ; # 1385
1388   PLANT.PIPEC2.H0 := 713180 ; # 1386
1389   PLANT.PIPEC2.T(1) := 645.577 ; # 1387
1390   PLANT.PIPEC2.T(2) := 675.275 ; # 1388
1391   PLANT.PIPEC2.T(3) := 703.34 ; # 1389
1392   PLANT.PIPEC2.T(4) := 729.862 ; # 1390
1393   PLANT.PIPEC2.T(5) := 754.927 ; # 1391
1394   PLANT.PIPEC2.T(6) := 778.614 ; # 1392
1395   PLANT.PIPEC2.T(7) := 801 ; # 1393
1396   PLANT.PIPEC2.T(8) := 822.155 ; # 1394
1397   PLANT.PIPEC2.T(9) := 842.147 ; # 1395
1398   PLANT.PIPEC2.T(10) := 861.041 ; # 1396
1399   PLANT.PIPEC2.T(11) := 878.896 ; # 1397
1400   PLANT.PIPEC2.T(12) := 895.77 ; # 1398
1401   PLANT.PIPEC2.T(13) := 911.717 ; # 1399
1402   PLANT.PIPEC2.T(14) := 926.787 ; # 1400
1403   PLANT.PIPEC2.T(15) := 941.029 ; # 1401
1404   PLANT.PIPEC2.T(16) := 954.488 ; # 1402
```

```
1405   PLANT.PIPEC2.T(17) := 967.207 ; # 1403
1406   PLANT.PIPEC2.T(18) := 979.228 ; # 1404
1407   PLANT.PIPEC2.T(19) := 990.587 ; # 1405
1408   PLANT.PIPEC2.T(20) := 1001.32 ; # 1406
1409   PLANT.PIPEC2.T(21) := 1011.47 ; # 1407
1410   PLANT.PIPEC2.T(22) := 1021.06 ; # 1408
1411   PLANT.PIPEC2.T(23) := 1030.12 ; # 1409
1412   PLANT.PIPEC2.T(24) := 1038.68 ; # 1410
1413   PLANT.PIPEC2.T(25) := 1046.77 ; # 1411
1414   PLANT.PIPEC2.T(26) := 1054.42 ; # 1412
1415   PLANT.PIPEC2.T(27) := 1061.65 ; # 1413
1416   PLANT.PIPEC2.T(28) := 1068.48 ; # 1414
1417   PLANT.PIPEC2.T(29) := 1074.93 ; # 1415
1418   PLANT.PIPEC2.T(30) := 1081.03 ; # 1416
1419   PLANT.PIPEC2.T(31) := 1086.8 ; # 1417
1420   PLANT.PIPEC2.T(32) := 1092.24 ; # 1418
1421   PLANT.PIPEC2.T(33) := 1097.39 ; # 1419
1422   PLANT.PIPEC2.T(34) := 1102.26 ; # 1420
1423   PLANT.PIPEC2.T(35) := 1106.86 ; # 1421
1424   PLANT.PIPEC2.T(36) := 1111.2 ; # 1422
1425   PLANT.PIPEC2.T(37) := 1115.31 ; # 1423
1426   PLANT.PIPEC2.T(38) := 1119.19 ; # 1424
1427   PLANT.PIPEC2.T(39) := 1122.86 ; # 1425
1428   PLANT.PIPEC2.T(40) := 1126.32 ; # 1426
1429   PLANT.PIPEC2.TEXTRNL(1) := 712.554 ; # 1427
1430   PLANT.PIPEC2.TEXTRNL(2) := 738.563 ; # 1428
1431   PLANT.PIPEC2.TEXTRNL(3) := 763.15 ; # 1429
1432   PLANT.PIPEC2.TEXTRNL(4) := 786.385 ; # 1430
1433   PLANT.PIPEC2.TEXTRNL(5) := 808.344 ; # 1431
1434   PLANT.PIPEC2.TEXTRNL(6) := 829.095 ; # 1432
1435   PLANT.PIPEC2.TEXTRNL(7) := 848.706 ; # 1433
1436   PLANT.PIPEC2.TEXTRNL(8) := 867.239 ; # 1434
1437   PLANT.PIPEC2.TEXTRNL(9) := 884.754 ; # 1435
1438   PLANT.PIPEC2.TEXTRNL(10) := 901.306 ; # 1436
1439   PLANT.PIPEC2.TEXTRNL(11) := 916.948 ; # 1437
1440   PLANT.PIPEC2.TEXTRNL(12) := 931.731 ; # 1438
1441   PLANT.PIPEC2.TEXTRNL(13) := 945.701 ; # 1439
1442   PLANT.PIPEC2.TEXTRNL(14) := 958.903 ; # 1440
1443   PLANT.PIPEC2.TEXTRNL(15) := 971.38 ; # 1441
1444   PLANT.PIPEC2.TEXTRNL(16) := 983.171 ; # 1442
1445   PLANT.PIPEC2.TEXTRNL(17) := 994.314 ; # 1443
1446   PLANT.PIPEC2.TEXTRNL(18) := 1004.84 ; # 1444
1447   PLANT.PIPEC2.TEXTRNL(19) := 1014.8 ; # 1445
1448   PLANT.PIPEC2.TEXTRNL(20) := 1024.2 ; # 1446
1449   PLANT.PIPEC2.TEXTRNL(21) := 1033.09 ; # 1447
1450   PLANT.PIPEC2.TEXTRNL(22) := 1041.49 ; # 1448
1451   PLANT.PIPEC2.TEXTRNL(23) := 1049.43 ; # 1449
1452   PLANT.PIPEC2.TEXTRNL(24) := 1056.93 ; # 1450
1453   PLANT.PIPEC2.TEXTRNL(25) := 1064.02 ; # 1451
1454   PLANT.PIPEC2.TEXTRNL(26) := 1070.72 ; # 1452
1455   PLANT.PIPEC2.TEXTRNL(27) := 1077.05 ; # 1453
1456   PLANT.PIPEC2.TEXTRNL(28) := 1083.03 ; # 1454
1457   PLANT.PIPEC2.TEXTRNL(29) := 1088.69 ; # 1455
1458   PLANT.PIPEC2.TEXTRNL(30) := 1094.03 ; # 1456
1459   PLANT.PIPEC2.TEXTRNL(31) := 1099.08 ; # 1457
1460   PLANT.PIPEC2.TEXTRNL(32) := 1103.85 ; # 1458
1461   PLANT.PIPEC2.TEXTRNL(33) := 1108.36 ; # 1459
1462   PLANT.PIPEC2.TEXTRNL(34) := 1112.63 ; # 1460
1463   PLANT.PIPEC2.TEXTRNL(35) := 1116.65 ; # 1461
1464   PLANT.PIPEC2.TEXTRNL(36) := 1120.46 ; # 1462
1465   PLANT.PIPEC2.TEXTRNL(37) := 1124.06 ; # 1463
1466   PLANT.PIPEC2.TEXTRNL(38) := 1127.46 ; # 1464
1467   PLANT.PIPEC2.TEXTRNL(39) := 1130.67 ; # 1465
1468   PLANT.PIPEC2.TEXTRNL(40) := 1133.71 ; # 1466
1469   PLANT.PIPEC2.F(1) := 43.3743 ; # 1467
1470   PLANT.PIPEC2.F(2) := 43.3743 ; # 1468
1471   PLANT.PIPEC2.F(3) := 43.3743 ; # 1469
1472   PLANT.PIPEC2.F(4) := 43.3743 ; # 1470
1473   PLANT.PIPEC2.F(5) := 43.3743 ; # 1471
1474   PLANT.PIPEC2.F(6) := 43.3743 ; # 1472
1475   PLANT.PIPEC2.F(7) := 43.3743 ; # 1473
1476   PLANT.PIPEC2.F(8) := 43.3743 ; # 1474
1477   PLANT.PIPEC2.F(9) := 43.3743 ; # 1475
1478   PLANT.PIPEC2.F(10) := 43.3743 ; # 1476
1479   PLANT.PIPEC2.F(11) := 43.3743 ; # 1477
1480   PLANT.PIPEC2.F(12) := 43.3743 ; # 1478
1481   PLANT.PIPEC2.F(13) := 43.3743 ; # 1479
1482   PLANT.PIPEC2.F(14) := 43.3743 ; # 1480
1483   PLANT.PIPEC2.F(15) := 43.3743 ; # 1481
1484   PLANT.PIPEC2.F(16) := 43.3743 ; # 1482
1485   PLANT.PIPEC2.F(17) := 43.3743 ; # 1483
1486   PLANT.PIPEC2.F(18) := 43.3743 ; # 1484
1487   PLANT.PIPEC2.F(19) := 43.3743 ; # 1485
1488   PLANT.PIPEC2.F(20) := 43.3743 ; # 1486
1489   PLANT.PIPEC2.F(21) := 43.3743 ; # 1487
1490   PLANT.PIPEC2.F(22) := 43.3743 ; # 1488
1491   PLANT.PIPEC2.F(23) := 43.3743 ; # 1489
1492   PLANT.PIPEC2.F(24) := 43.3743 ; # 1490
1493   PLANT.PIPEC2.F(25) := 43.3743 ; # 1491
1494   PLANT.PIPEC2.F(26) := 43.3743 ; # 1492
1495   PLANT.PIPEC2.F(27) := 43.3743 ; # 1493
1496   PLANT.PIPEC2.F(28) := 43.3743 ; # 1494
```

```
PLANT.PIPEC2.F(29) := 43.3743 ; # 1495
PLANT.PIPEC2.F(30) := 43.3743 ; # 1496
PLANT.PIPEC2.F(31) := 43.3743 ; # 1497
PLANT.PIPEC2.F(32) := 43.3743 ; # 1498
PLANT.PIPEC2.F(33) := 43.3743 ; # 1499
PLANT.PIPEC2.F(34) := 43.3743 ; # 1500
PLANT.PIPEC2.F(35) := 43.3743 ; # 1501
PLANT.PIPEC2.F(36) := 43.3743 ; # 1502
PLANT.PIPEC2.F(37) := 43.3743 ; # 1503
PLANT.PIPEC2.F(38) := 43.3743 ; # 1504
PLANT.PIPEC2.F(39) := 43.3743 ; # 1505
PLANT.PIPEC2.F(40) := 43.3743 ; # 1506
PLANT.PIPEC2.F_IN := 43.3743 ; # 1507
PLANT.PIPEC2.PSI(1) := 1.81063e+006 ; # 1508
PLANT.PIPEC2.PSI(2) := 1.91725e+006 ; # 1509
PLANT.PIPEC2.PSI(3) := 2.00881e+006 ; # 1510
PLANT.PIPEC2.PSI(4) := 2.08797e+006 ; # 1511
PLANT.PIPEC2.PSI(5) := 2.1568e+006 ; # 1512
PLANT.PIPEC2.PSI(6) := 2.21692e+006 ; # 1513
PLANT.PIPEC2.PSI(7) := 2.26963e+006 ; # 1514
PLANT.PIPEC2.PSI(8) := 2.31599e+006 ; # 1515
PLANT.PIPEC2.PSI(9) := 2.35686e+006 ; # 1516
PLANT.PIPEC2.PSI(10) := 2.39296e+006 ; # 1517
PLANT.PIPEC2.PSI(11) := 2.42488e+006 ; # 1518
PLANT.PIPEC2.PSI(12) := 2.45312e+006 ; # 1519
PLANT.PIPEC2.PSI(13) := 2.47811e+006 ; # 1520
PLANT.PIPEC2.PSI(14) := 2.50021e+006 ; # 1521
PLANT.PIPEC2.PSI(15) := 2.51973e+006 ; # 1522
PLANT.PIPEC2.PSI(16) := 2.53693e+006 ; # 1523
PLANT.PIPEC2.PSI(17) := 2.55205e+006 ; # 1524
PLANT.PIPEC2.PSI(18) := 2.5653e+006 ; # 1525
PLANT.PIPEC2.PSI(19) := 2.57684e+006 ; # 1526
PLANT.PIPEC2.PSI(20) := 2.58684e+006 ; # 1527
PLANT.PIPEC2.PSI(21) := 2.59544e+006 ; # 1528
PLANT.PIPEC2.PSI(22) := 2.60275e+006 ; # 1529
PLANT.PIPEC2.PSI(23) := 2.60889e+006 ; # 1530
PLANT.PIPEC2.PSI(24) := 2.61395e+006 ; # 1531
PLANT.PIPEC2.PSI(25) := 2.61803e+006 ; # 1532
PLANT.PIPEC2.PSI(26) := 2.62119e+006 ; # 1533
PLANT.PIPEC2.PSI(27) := 2.62352e+006 ; # 1534
PLANT.PIPEC2.PSI(28) := 2.62507e+006 ; # 1535
PLANT.PIPEC2.PSI(29) := 2.62591e+006 ; # 1536
PLANT.PIPEC2.PSI(30) := 2.62607e+006 ; # 1537
PLANT.PIPEC2.PSI(31) := 2.62562e+006 ; # 1538
PLANT.PIPEC2.PSI(32) := 2.6246e+006 ; # 1539
PLANT.PIPEC2.PSI(33) := 2.62304e+006 ; # 1540
PLANT.PIPEC2.PSI(34) := 2.62099e+006 ; # 1541
PLANT.PIPEC2.PSI(35) := 2.61847e+006 ; # 1542
PLANT.PIPEC2.PSI(36) := 2.61551e+006 ; # 1543
PLANT.PIPEC2.PSI(37) := 2.61215e+006 ; # 1544
PLANT.PIPEC2.PSI(38) := 2.6084e+006 ; # 1545
PLANT.PIPEC2.PSI(39) := 2.6043e+006 ; # 1546
PLANT.PIPEC2.PSI(40) := 2.59986e+006 ; # 1547
PLANT.PIPEC2.HT.RHOPIPEHE := 2.35217 ; # 1548
PLANT.PIPEC2.HT.VPIPE := 65.6203 ; # 1549
PLANT.PIPEC2.HT.U := 1880.53 ; # 1550
```

HTL0MACH_SS_IN_INI.INITIAL

```
# Values for differential variables in computation HTL0MACH_SS
# Saved at time 1000
PLANT.HX.THX(1) = 879.893 ; # 1
PLANT.HX.THX(2) = 892.184 ; # 2
PLANT.HX.THX(3) = 904.161 ; # 3
PLANT.HX.THX(4) = 915.766 ; # 4
PLANT.HX.THX(5) = 927.009 ; # 5
PLANT.HX.THX(6) = 937.904 ; # 6
PLANT.HX.THX(7) = 948.459 ; # 7
PLANT.HX.THX(8) = 958.687 ; # 8
PLANT.HX.THX(9) = 968.596 ; # 9
PLANT.HX.THX(10) = 978.197 ; # 10
PLANT.HX.THX(11) = 987.5 ; # 11
PLANT.HX.THX(12) = 996.513 ; # 12
PLANT.HX.THX(13) = 1005.25 ; # 13
PLANT.HX.THX(14) = 1013.71 ; # 14
PLANT.HX.THX(15) = 1021.91 ; # 15
PLANT.HX.THX(16) = 1029.85 ; # 16
PLANT.HX.THX(17) = 1037.55 ; # 17
PLANT.HX.THX(18) = 1045 ; # 18
PLANT.HX.THX(19) = 1052.23 ; # 19
PLANT.HX.THX(20) = 1059.23 ; # 20
PLANT.HX.THX(21) = 1066.01 ; # 21
PLANT.HX.THX(22) = 1072.58 ; # 22
```

```
PLANT.HX.THX(23) = 1078.95 ; # 23
PLANT.HX.THX(24) = 1085.12 ; # 24
PLANT.HX.THX(25) = 1091.1 ; # 25
PLANT.HX.THX(26) = 1096.89 ; # 26
PLANT.HX.THX(27) = 1102.5 ; # 27
PLANT.HX.THX(28) = 1107.94 ; # 28
PLANT.HX.THX(29) = 1113.21 ; # 29
PLANT.HX.THX(30) = 1118.31 ; # 30
PLANT.HX.THX(31) = 1123.26 ; # 31
PLANT.HX.THX(32) = 1128.05 ; # 32
PLANT.HX.THX(33) = 1132.69 ; # 33
PLANT.HX.THX(34) = 1137.19 ; # 34
PLANT.HX.THX(35) = 1141.55 ; # 35
PLANT.HX.THX(36) = 1145.77 ; # 36
PLANT.HX.THX(37) = 1149.86 ; # 37
PLANT.HX.THX(38) = 1153.83 ; # 38
PLANT.HX.THX(39) = 1157.67 ; # 39
PLANT.HX.THX(40) = 1161.37 ; # 40
PLANT.LOOP1.RHO(1) = 1.15317 ; # 41
PLANT.LOOP1.RHO(2) = 1.15067 ; # 42
PLANT.LOOP1.RHO(3) = 1.14817 ; # 43
PLANT.LOOP1.RHO(4) = 1.14565 ; # 44
PLANT.LOOP1.RHO(5) = 1.14313 ; # 45
PLANT.LOOP1.RHO(6) = 1.14059 ; # 46
PLANT.LOOP1.RHO(7) = 1.13804 ; # 47
PLANT.LOOP1.RHO(8) = 1.13549 ; # 48
PLANT.LOOP1.RHO(9) = 1.13292 ; # 49
PLANT.LOOP1.RHO(10) = 1.13034 ; # 50
PLANT.LOOP1.RHO(11) = 1.12115 ; # 51
PLANT.LOOP1.RHO(12) = 1.1026 ; # 52
PLANT.LOOP1.RHO(13) = 1.08477 ; # 53
PLANT.LOOP1.RHO(14) = 1.06791 ; # 54
PLANT.LOOP1.RHO(15) = 1.05196 ; # 55
PLANT.LOOP1.RHO(16) = 1.03683 ; # 56
PLANT.LOOP1.RHO(17) = 1.02247 ; # 57
PLANT.LOOP1.RHO(18) = 1.00882 ; # 58
PLANT.LOOP1.RHO(19) = 0.995831 ; # 59
PLANT.LOOP1.RHO(20) = 0.983453 ; # 60
PLANT.LOOP1.RHO(21) = 0.971645 ; # 61
PLANT.LOOP1.RHO(22) = 0.960369 ; # 62
PLANT.LOOP1.RHO(23) = 0.949588 ; # 63
PLANT.LOOP1.RHO(24) = 0.939272 ; # 64
PLANT.LOOP1.RHO(25) = 0.929389 ; # 65
PLANT.LOOP1.RHO(26) = 0.919912 ; # 66
PLANT.LOOP1.RHO(27) = 0.910817 ; # 67
PLANT.LOOP1.RHO(28) = 0.90208 ; # 68
PLANT.LOOP1.RHO(29) = 0.89368 ; # 69
PLANT.LOOP1.RHO(30) = 0.885596 ; # 70
PLANT.LOOP1.RHO(31) = 0.87781 ; # 71
PLANT.LOOP1.RHO(32) = 0.870305 ; # 72
PLANT.LOOP1.RHO(33) = 0.863066 ; # 73
PLANT.LOOP1.RHO(34) = 0.856077 ; # 74
PLANT.LOOP1.RHO(35) = 0.849325 ; # 75
PLANT.LOOP1.RHO(36) = 0.842796 ; # 76
PLANT.LOOP1.RHO(37) = 0.83648 ; # 77
PLANT.LOOP1.RHO(38) = 0.830364 ; # 78
PLANT.LOOP1.RHO(39) = 0.824437 ; # 79
PLANT.LOOP1.RHO(40) = 0.818691 ; # 80
PLANT.LOOP1.RHO(41) = 0.813116 ; # 81
PLANT.LOOP1.RHO(42) = 0.807703 ; # 82
PLANT.LOOP1.RHO(43) = 0.802444 ; # 83
PLANT.LOOP1.RHO(44) = 0.797332 ; # 84
PLANT.LOOP1.RHO(45) = 0.792359 ; # 85
PLANT.LOOP1.RHO(46) = 0.787518 ; # 86
PLANT.LOOP1.RHO(47) = 0.782803 ; # 87
PLANT.LOOP1.RHO(48) = 0.778208 ; # 88
PLANT.LOOP1.RHO(49) = 0.773727 ; # 89
PLANT.LOOP1.RHO(50) = 0.769359 ; # 90
PLANT.LOOP1.RHO(51) = 0.757959 ; # 91
PLANT.LOOP1.RHO(52) = 0.756103 ; # 92
PLANT.LOOP1.RHO(53) = 0.754256 ; # 93
PLANT.LOOP1.RHO(54) = 0.752399 ; # 94
PLANT.LOOP1.RHO(55) = 0.750533 ; # 95
PLANT.LOOP1.RHO(56) = 0.748658 ; # 96
PLANT.LOOP1.RHO(57) = 0.746773 ; # 97
PLANT.LOOP1.RHO(58) = 0.744879 ; # 98
PLANT.LOOP1.RHO(59) = 0.742975 ; # 99
PLANT.LOOP1.RHO(60) = 0.741061 ; # 100
PLANT.LOOP1.RHO(61) = 0.750585 ; # 101
PLANT.LOOP1.RHO(62) = 0.751689 ; # 102
PLANT.LOOP1.RHO(63) = 0.752856 ; # 103
PLANT.LOOP1.RHO(64) = 0.754135 ; # 104
PLANT.LOOP1.RHO(65) = 0.755533 ; # 105
PLANT.LOOP1.RHO(66) = 0.757059 ; # 106
PLANT.LOOP1.RHO(67) = 0.75872 ; # 107
PLANT.LOOP1.RHO(68) = 0.760528 ; # 108
PLANT.LOOP1.RHO(69) = 0.762492 ; # 109
PLANT.LOOP1.RHO(70) = 0.764623 ; # 110
PLANT.LOOP1.RHO(71) = 0.766934 ; # 111
PLANT.LOOP1.RHO(72) = 0.769438 ; # 112
PLANT.LOOP1.RHO(73) = 0.772149 ; # 113
PLANT.LOOP1.RHO(74) = 0.775083 ; # 114
```

```
PLANT.LOOP1.RHO(75) = 0.778256 ; # 115
PLANT.LOOP1.RHO(76) = 0.781688 ; # 116
PLANT.LOOP1.RHO(77) = 0.785399 ; # 117
PLANT.LOOP1.RHO(78) = 0.789411 ; # 118
PLANT.LOOP1.RHO(79) = 0.793749 ; # 119
PLANT.LOOP1.RHO(80) = 0.798441 ; # 120
PLANT.LOOP1.RHO(81) = 0.803515 ; # 121
PLANT.LOOP1.RHO(82) = 0.809005 ; # 122
PLANT.LOOP1.RHO(83) = 0.814949 ; # 123
PLANT.LOOP1.RHO(84) = 0.821387 ; # 124
PLANT.LOOP1.RHO(85) = 0.828365 ; # 125
PLANT.LOOP1.RHO(86) = 0.835933 ; # 126
PLANT.LOOP1.RHO(87) = 0.844151 ; # 127
PLANT.LOOP1.RHO(88) = 0.853082 ; # 128
PLANT.LOOP1.RHO(89) = 0.8628 ; # 129
PLANT.LOOP1.RHO(90) = 0.873389 ; # 130
PLANT.LOOP1.RHO(91) = 0.884944 ; # 131
PLANT.LOOP1.RHO(92) = 0.897573 ; # 132
PLANT.LOOP1.RHO(93) = 0.911403 ; # 133
PLANT.LOOP1.RHO(94) = 0.926579 ; # 134
PLANT.LOOP1.RHO(95) = 0.94327 ; # 135
PLANT.LOOP1.RHO(96) = 0.961674 ; # 136
PLANT.LOOP1.RHO(97) = 0.982027 ; # 137
PLANT.LOOP1.RHO(98) = 1.0046 ; # 138
PLANT.LOOP1.RHO(99) = 1.02974 ; # 139
PLANT.LOOP1.RHO(100) = 1.05783 ; # 140
PLANT.LOOP1.PSI(1) = 1.23127e+006 ; # 141
PLANT.LOOP1.PSI(2) = 1.22711e+006 ; # 142
PLANT.LOOP1.PSI(3) = 1.22294e+006 ; # 143
PLANT.LOOP1.PSI(4) = 1.21877e+006 ; # 144
PLANT.LOOP1.PSI(5) = 1.2146e+006 ; # 145
PLANT.LOOP1.PSI(6) = 1.21042e+006 ; # 146
PLANT.LOOP1.PSI(7) = 1.20624e+006 ; # 147
PLANT.LOOP1.PSI(8) = 1.20206e+006 ; # 148
PLANT.LOOP1.PSI(9) = 1.19787e+006 ; # 149
PLANT.LOOP1.PSI(10) = 1.19368e+006 ; # 150
PLANT.LOOP1.PSI(11) = 1.23208e+006 ; # 151
PLANT.LOOP1.PSI(12) = 1.25743e+006 ; # 152
PLANT.LOOP1.PSI(13) = 1.28069e+006 ; # 153
PLANT.LOOP1.PSI(14) = 1.30237e+006 ; # 154
PLANT.LOOP1.PSI(15) = 1.3226e+006 ; # 155
PLANT.LOOP1.PSI(16) = 1.34148e+006 ; # 156
PLANT.LOOP1.PSI(17) = 1.35911e+006 ; # 157
PLANT.LOOP1.PSI(18) = 1.37558e+006 ; # 158
PLANT.LOOP1.PSI(19) = 1.39098e+006 ; # 159
PLANT.LOOP1.PSI(20) = 1.40537e+006 ; # 160
PLANT.LOOP1.PSI(21) = 1.41883e+006 ; # 161
PLANT.LOOP1.PSI(22) = 1.4314e+006 ; # 162
PLANT.LOOP1.PSI(23) = 1.44316e+006 ; # 163
PLANT.LOOP1.PSI(24) = 1.45415e+006 ; # 164
PLANT.LOOP1.PSI(25) = 1.46441e+006 ; # 165
PLANT.LOOP1.PSI(26) = 1.474e+006 ; # 166
PLANT.LOOP1.PSI(27) = 1.48294e+006 ; # 167
PLANT.LOOP1.PSI(28) = 1.49129e+006 ; # 168
PLANT.LOOP1.PSI(29) = 1.49907e+006 ; # 169
PLANT.LOOP1.PSI(30) = 1.50631e+006 ; # 170
PLANT.LOOP1.PSI(31) = 1.51304e+006 ; # 171
PLANT.LOOP1.PSI(32) = 1.5193e+006 ; # 172
PLANT.LOOP1.PSI(33) = 1.5251e+006 ; # 173
PLANT.LOOP1.PSI(34) = 1.53047e+006 ; # 174
PLANT.LOOP1.PSI(35) = 1.53543e+006 ; # 175
PLANT.LOOP1.PSI(36) = 1.54001e+006 ; # 176
PLANT.LOOP1.PSI(37) = 1.54421e+006 ; # 177
PLANT.LOOP1.PSI(38) = 1.54807e+006 ; # 178
PLANT.LOOP1.PSI(39) = 1.5516e+006 ; # 179
PLANT.LOOP1.PSI(40) = 1.5548e+006 ; # 180
PLANT.LOOP1.PSI(41) = 1.55771e+006 ; # 181
PLANT.LOOP1.PSI(42) = 1.56032e+006 ; # 182
PLANT.LOOP1.PSI(43) = 1.56266e+006 ; # 183
PLANT.LOOP1.PSI(44) = 1.56474e+006 ; # 184
PLANT.LOOP1.PSI(45) = 1.56657e+006 ; # 185
PLANT.LOOP1.PSI(46) = 1.56816e+006 ; # 186
PLANT.LOOP1.PSI(47) = 1.56951e+006 ; # 187
PLANT.LOOP1.PSI(48) = 1.57065e+006 ; # 188
PLANT.LOOP1.PSI(49) = 1.57158e+006 ; # 189
PLANT.LOOP1.PSI(50) = 1.5723e+006 ; # 190
PLANT.LOOP1.PSI(51) = 1.54707e+006 ; # 191
PLANT.LOOP1.PSI(52) = 1.54135e+006 ; # 192
PLANT.LOOP1.PSI(53) = 1.53567e+006 ; # 193
PLANT.LOOP1.PSI(54) = 1.52997e+006 ; # 194
PLANT.LOOP1.PSI(55) = 1.52427e+006 ; # 195
PLANT.LOOP1.PSI(56) = 1.51856e+006 ; # 196
PLANT.LOOP1.PSI(57) = 1.51284e+006 ; # 197
PLANT.LOOP1.PSI(58) = 1.50712e+006 ; # 198
PLANT.LOOP1.PSI(59) = 1.50138e+006 ; # 199
PLANT.LOOP1.PSI(60) = 1.49564e+006 ; # 200
PLANT.LOOP1.PSI(61) = 1.50921e+006 ; # 201
PLANT.LOOP1.PSI(62) = 1.50544e+006 ; # 202
PLANT.LOOP1.PSI(63) = 1.50142e+006 ; # 203
PLANT.LOOP1.PSI(64) = 1.49724e+006 ; # 204
PLANT.LOOP1.PSI(65) = 1.49288e+006 ; # 205
PLANT.LOOP1.PSI(66) = 1.48833e+006 ; # 206
```

```
PLANT.LOOP1.PSI(67) = 1.48357e+006 ; # 207
PLANT.LOOP1.PSI(68) = 1.47859e+006 ; # 208
PLANT.LOOP1.PSI(69) = 1.47338e+006 ; # 209
PLANT.LOOP1.PSI(70) = 1.46791e+006 ; # 210
PLANT.LOOP1.PSI(71) = 1.46218e+006 ; # 211
PLANT.LOOP1.PSI(72) = 1.45616e+006 ; # 212
PLANT.LOOP1.PSI(73) = 1.44982e+006 ; # 213
PLANT.LOOP1.PSI(74) = 1.44315e+006 ; # 214
PLANT.LOOP1.PSI(75) = 1.43612e+006 ; # 215
PLANT.LOOP1.PSI(76) = 1.4287e+006 ; # 216
PLANT.LOOP1.PSI(77) = 1.42087e+006 ; # 217
PLANT.LOOP1.PSI(78) = 1.41258e+006 ; # 218
PLANT.LOOP1.PSI(79) = 1.40379e+006 ; # 219
PLANT.LOOP1.PSI(80) = 1.39448e+006 ; # 220
PLANT.LOOP1.PSI(81) = 1.38459e+006 ; # 221
PLANT.LOOP1.PSI(82) = 1.37407e+006 ; # 222
PLANT.LOOP1.PSI(83) = 1.36287e+006 ; # 223
PLANT.LOOP1.PSI(84) = 1.35093e+006 ; # 224
PLANT.LOOP1.PSI(85) = 1.33817e+006 ; # 225
PLANT.LOOP1.PSI(86) = 1.32451e+006 ; # 226
PLANT.LOOP1.PSI(87) = 1.30988e+006 ; # 227
PLANT.LOOP1.PSI(88) = 1.29416e+006 ; # 228
PLANT.LOOP1.PSI(89) = 1.27725e+006 ; # 229
PLANT.LOOP1.PSI(90) = 1.25903e+006 ; # 230
PLANT.LOOP1.PSI(91) = 1.23934e+006 ; # 231
PLANT.LOOP1.PSI(92) = 1.21801e+006 ; # 232
PLANT.LOOP1.PSI(93) = 1.19486e+006 ; # 233
PLANT.LOOP1.PSI(94) = 1.16967e+006 ; # 234
PLANT.LOOP1.PSI(95) = 1.14217e+006 ; # 235
PLANT.LOOP1.PSI(96) = 1.11205e+006 ; # 236
PLANT.LOOP1.PSI(97) = 1.07896e+006 ; # 237
PLANT.LOOP1.PSI(98) = 1.04248e+006 ; # 238
PLANT.LOOP1.PSI(99) = 1.00208e+006 ; # 239
PLANT.LOOP1.PSI(100) = 957156 ; # 240
PLANT.LOOP1.RHOEXTRA(1) = 1.1654 ; # 241
PLANT.LOOP1.RHOEXTRA(2) = 1.1654 ; # 242
PLANT.LOOP1.PSIEXTRA(1) = 1.24585e+006 ; # 243
PLANT.LOOP1.PSIEXTRA(2) = 1.24585e+006 ; # 244
PLANT.PIPEH.RHO(1) = 2.87869 ; # 245
PLANT.PIPEH.RHO(2) = 2.88664 ; # 246
PLANT.PIPEH.RHO(3) = 2.89491 ; # 247
PLANT.PIPEH.RHO(4) = 2.90353 ; # 248
PLANT.PIPEH.RHO(5) = 2.9125 ; # 249
PLANT.PIPEH.RHO(6) = 2.92184 ; # 250
PLANT.PIPEH.RHO(7) = 2.93156 ; # 251
PLANT.PIPEH.RHO(8) = 2.9417 ; # 252
PLANT.PIPEH.RHO(9) = 2.95225 ; # 253
PLANT.PIPEH.RHO(10) = 2.96325 ; # 254
PLANT.PIPEH.RHO(11) = 2.97471 ; # 255
PLANT.PIPEH.RHO(12) = 2.98666 ; # 256
PLANT.PIPEH.RHO(13) = 2.99912 ; # 257
PLANT.PIPEH.RHO(14) = 3.01211 ; # 258
PLANT.PIPEH.RHO(15) = 3.02567 ; # 259
PLANT.PIPEH.RHO(16) = 3.03982 ; # 260
PLANT.PIPEH.RHO(17) = 3.05458 ; # 261
PLANT.PIPEH.RHO(18) = 3.07 ; # 262
PLANT.PIPEH.RHO(19) = 3.08611 ; # 263
PLANT.PIPEH.RHO(20) = 3.10294 ; # 264
PLANT.PIPEH.RHO(21) = 3.12053 ; # 265
PLANT.PIPEH.RHO(22) = 3.13892 ; # 266
PLANT.PIPEH.RHO(23) = 3.15816 ; # 267
PLANT.PIPEH.RHO(24) = 3.1783 ; # 268
PLANT.PIPEH.RHO(25) = 3.19939 ; # 269
PLANT.PIPEH.RHO(26) = 3.22148 ; # 270
PLANT.PIPEH.RHO(27) = 3.24463 ; # 271
PLANT.PIPEH.RHO(28) = 3.2689 ; # 272
PLANT.PIPEH.RHO(29) = 3.29437 ; # 273
PLANT.PIPEH.RHO(30) = 3.32112 ; # 274
PLANT.PIPEH.RHO(31) = 3.34921 ; # 275
PLANT.PIPEH.RHO(32) = 3.37874 ; # 276
PLANT.PIPEH.RHO(33) = 3.4098 ; # 277
PLANT.PIPEH.RHO(34) = 3.4425 ; # 278
PLANT.PIPEH.RHO(35) = 3.47695 ; # 279
PLANT.PIPEH.RHO(36) = 3.51328 ; # 280
PLANT.PIPEH.RHO(37) = 3.55161 ; # 281
PLANT.PIPEH.RHO(38) = 3.59209 ; # 282
PLANT.PIPEH.RHO(39) = 3.6349 ; # 283
PLANT.PIPEH.RHO(40) = 3.68012 ; # 284
PLANT.PIPEH.F(1) = 32.3305 ; # 285
PLANT.PIPEH.F(2) = 32.3305 ; # 286
PLANT.PIPEH.F(3) = 32.3305 ; # 287
PLANT.PIPEH.F(4) = 32.3305 ; # 288
PLANT.PIPEH.F(5) = 32.3305 ; # 289
PLANT.PIPEH.F(6) = 32.3305 ; # 290
PLANT.PIPEH.F(7) = 32.3305 ; # 291
PLANT.PIPEH.F(8) = 32.3305 ; # 292
PLANT.PIPEH.F(9) = 32.3305 ; # 293
PLANT.PIPEH.F(10) = 32.3305 ; # 294
PLANT.PIPEH.F(11) = 32.3305 ; # 295
PLANT.PIPEH.F(12) = 32.3305 ; # 296
PLANT.PIPEH.F(13) = 32.3305 ; # 297
PLANT.PIPEH.F(14) = 32.3305 ; # 298
```

```
PLANT.PIPEH.F(15) = 32.3305 ; # 299
PLANT.PIPEH.F(16) = 32.3305 ; # 300
PLANT.PIPEH.F(17) = 32.3305 ; # 301
PLANT.PIPEH.F(18) = 32.3305 ; # 302
PLANT.PIPEH.F(19) = 32.3305 ; # 303
PLANT.PIPEH.F(20) = 32.3305 ; # 304
PLANT.PIPEH.F(21) = 32.3305 ; # 305
PLANT.PIPEH.F(22) = 32.3305 ; # 306
PLANT.PIPEH.F(23) = 32.3305 ; # 307
PLANT.PIPEH.F(24) = 32.3305 ; # 308
PLANT.PIPEH.F(25) = 32.3305 ; # 309
PLANT.PIPEH.F(26) = 32.3305 ; # 310
PLANT.PIPEH.F(27) = 32.3305 ; # 311
PLANT.PIPEH.F(28) = 32.3305 ; # 312
PLANT.PIPEH.F(29) = 32.3305 ; # 313
PLANT.PIPEH.F(30) = 32.3305 ; # 314
PLANT.PIPEH.F(31) = 32.3305 ; # 315
PLANT.PIPEH.F(32) = 32.3305 ; # 316
PLANT.PIPEH.F(33) = 32.3305 ; # 317
PLANT.PIPEH.F(34) = 32.3305 ; # 318
PLANT.PIPEH.F(35) = 32.3305 ; # 319
PLANT.PIPEH.F(36) = 32.3305 ; # 320
PLANT.PIPEH.F(37) = 32.3305 ; # 321
PLANT.PIPEH.F(38) = 32.3305 ; # 322
PLANT.PIPEH.F(39) = 32.3305 ; # 323
PLANT.PIPEH.F(40) = 32.3305 ; # 324
PLANT.PIPEH.F_IN = 32.3305 ; # 325
PLANT.PIPEH.PSI(1) = 6.03823e+006 ; # 326
PLANT.PIPEH.PSI(2) = 6.02385e+006 ; # 327
PLANT.PIPEH.PSI(3) = 6.00898e+006 ; # 328
PLANT.PIPEH.PSI(4) = 5.9936e+006 ; # 329
PLANT.PIPEH.PSI(5) = 5.97767e+006 ; # 330
PLANT.PIPEH.PSI(6) = 5.96117e+006 ; # 331
PLANT.PIPEH.PSI(7) = 5.94408e+006 ; # 332
PLANT.PIPEH.PSI(8) = 5.92638e+006 ; # 333
PLANT.PIPEH.PSI(9) = 5.90802e+006 ; # 334
PLANT.PIPEH.PSI(10) = 5.88899e+006 ; # 335
PLANT.PIPEH.PSI(11) = 5.86924e+006 ; # 336
PLANT.PIPEH.PSI(12) = 5.84876e+006 ; # 337
PLANT.PIPEH.PSI(13) = 5.82749e+006 ; # 338
PLANT.PIPEH.PSI(14) = 5.8054e+006 ; # 339
PLANT.PIPEH.PSI(15) = 5.78245e+006 ; # 340
PLANT.PIPEH.PSI(16) = 5.75859e+006 ; # 341
PLANT.PIPEH.PSI(17) = 5.73379e+006 ; # 342
PLANT.PIPEH.PSI(18) = 5.70799e+006 ; # 343
PLANT.PIPEH.PSI(19) = 5.68113e+006 ; # 344
PLANT.PIPEH.PSI(20) = 5.65317e+006 ; # 345
PLANT.PIPEH.PSI(21) = 5.62404e+006 ; # 346
PLANT.PIPEH.PSI(22) = 5.59367e+006 ; # 347
PLANT.PIPEH.PSI(23) = 5.56201e+006 ; # 348
PLANT.PIPEH.PSI(24) = 5.52897e+006 ; # 349
PLANT.PIPEH.PSI(25) = 5.49447e+006 ; # 350
PLANT.PIPEH.PSI(26) = 5.45844e+006 ; # 351
PLANT.PIPEH.PSI(27) = 5.42078e+006 ; # 352
PLANT.PIPEH.PSI(28) = 5.38139e+006 ; # 353
PLANT.PIPEH.PSI(29) = 5.34016e+006 ; # 354
PLANT.PIPEH.PSI(30) = 5.29699e+006 ; # 355
PLANT.PIPEH.PSI(31) = 5.25174e+006 ; # 356
PLANT.PIPEH.PSI(32) = 5.20427e+006 ; # 357
PLANT.PIPEH.PSI(33) = 5.15446e+006 ; # 358
PLANT.PIPEH.PSI(34) = 5.10212e+006 ; # 359
PLANT.PIPEH.PSI(35) = 5.0471e+006 ; # 360
PLANT.PIPEH.PSI(36) = 4.98919e+006 ; # 361
PLANT.PIPEH.PSI(37) = 4.92819e+006 ; # 362
PLANT.PIPEH.PSI(38) = 4.86388e+006 ; # 363
PLANT.PIPEH.PSI(39) = 4.796e+006 ; # 364
PLANT.PIPEH.PSI(40) = 4.7244e+006 ; # 365
PLANT.HX2.THX(1) = 1133.71 ; # 366
PLANT.HX2.THX(2) = 1130.67 ; # 367
PLANT.HX2.THX(3) = 1127.46 ; # 368
PLANT.HX2.THX(4) = 1124.06 ; # 369
PLANT.HX2.THX(5) = 1120.46 ; # 370
PLANT.HX2.THX(6) = 1116.65 ; # 371
PLANT.HX2.THX(7) = 1112.63 ; # 372
PLANT.HX2.THX(8) = 1108.36 ; # 373
PLANT.HX2.THX(9) = 1103.85 ; # 374
PLANT.HX2.THX(10) = 1099.08 ; # 375
PLANT.HX2.THX(11) = 1094.03 ; # 376
PLANT.HX2.THX(12) = 1088.69 ; # 377
PLANT.HX2.THX(13) = 1083.03 ; # 378
PLANT.HX2.THX(14) = 1077.05 ; # 379
PLANT.HX2.THX(15) = 1070.72 ; # 380
PLANT.HX2.THX(16) = 1064.02 ; # 381
PLANT.HX2.THX(17) = 1056.93 ; # 382
PLANT.HX2.THX(18) = 1049.43 ; # 383
PLANT.HX2.THX(19) = 1041.49 ; # 384
PLANT.HX2.THX(20) = 1033.09 ; # 385
PLANT.HX2.THX(21) = 1024.2 ; # 386
PLANT.HX2.THX(22) = 1014.8 ; # 387
PLANT.HX2.THX(23) = 1004.84 ; # 388
PLANT.HX2.THX(24) = 994.314 ; # 389
PLANT.HX2.THX(25) = 983.171 ; # 390
```

```
PLANT.HX2.THX(26) = 971.38 ; # 391
PLANT.HX2.THX(27) = 958.903 ; # 392
PLANT.HX2.THX(28) = 945.701 ; # 393
PLANT.HX2.THX(29) = 931.731 ; # 394
PLANT.HX2.THX(30) = 916.948 ; # 395
PLANT.HX2.THX(31) = 901.306 ; # 396
PLANT.HX2.THX(32) = 884.754 ; # 397
PLANT.HX2.THX(33) = 867.239 ; # 398
PLANT.HX2.THX(34) = 848.706 ; # 399
PLANT.HX2.THX(35) = 829.095 ; # 400
PLANT.HX2.THX(36) = 808.344 ; # 401
PLANT.HX2.THX(37) = 786.385 ; # 402
PLANT.HX2.THX(38) = 763.15 ; # 403
PLANT.HX2.THX(39) = 738.563 ; # 404
PLANT.HX2.THX(40) = 712.554 ; # 405
PLANT.PIPEC2.RHO(1) = 3.72389 ; # 406
PLANT.PIPEC2.RHO(2) = 3.55368 ; # 407
PLANT.PIPEC2.RHO(3) = 3.4055 ; # 408
PLANT.PIPEC2.RHO(4) = 3.27543 ; # 409
PLANT.PIPEC2.RHO(5) = 3.16041 ; # 410
PLANT.PIPEC2.RHO(6) = 3.05803 ; # 411
PLANT.PIPEC2.RHO(7) = 2.96638 ; # 412
PLANT.PIPEC2.RHO(8) = 2.88388 ; # 413
PLANT.PIPEC2.RHO(9) = 2.80928 ; # 414
PLANT.PIPEC2.RHO(10) = 2.74152 ; # 415
PLANT.PIPEC2.RHO(11) = 2.67973 ; # 416
PLANT.PIPEC2.RHO(12) = 2.62316 ; # 417
PLANT.PIPEC2.RHO(13) = 2.57121 ; # 418
PLANT.PIPEC2.RHO(14) = 2.52333 ; # 419
PLANT.PIPEC2.RHO(15) = 2.47909 ; # 420
PLANT.PIPEC2.RHO(16) = 2.43808 ; # 421
PLANT.PIPEC2.RHO(17) = 2.39997 ; # 422
PLANT.PIPEC2.RHO(18) = 2.36446 ; # 423
PLANT.PIPEC2.RHO(19) = 2.3313 ; # 424
PLANT.PIPEC2.RHO(20) = 2.30026 ; # 425
PLANT.PIPEC2.RHO(21) = 2.27113 ; # 426
PLANT.PIPEC2.RHO(22) = 2.24375 ; # 427
PLANT.PIPEC2.RHO(23) = 2.21796 ; # 428
PLANT.PIPEC2.RHO(24) = 2.19361 ; # 429
PLANT.PIPEC2.RHO(25) = 2.17058 ; # 430
PLANT.PIPEC2.RHO(26) = 2.14875 ; # 431
PLANT.PIPEC2.RHO(27) = 2.12804 ; # 432
PLANT.PIPEC2.RHO(28) = 2.10834 ; # 433
PLANT.PIPEC2.RHO(29) = 2.08957 ; # 434
PLANT.PIPEC2.RHO(30) = 2.07166 ; # 435
PLANT.PIPEC2.RHO(31) = 2.05454 ; # 436
PLANT.PIPEC2.RHO(32) = 2.03814 ; # 437
PLANT.PIPEC2.RHO(33) = 2.02242 ; # 438
PLANT.PIPEC2.RHO(34) = 2.00733 ; # 439
PLANT.PIPEC2.RHO(35) = 1.9928 ; # 440
PLANT.PIPEC2.RHO(36) = 1.97881 ; # 441
PLANT.PIPEC2.RHO(37) = 1.96531 ; # 442
PLANT.PIPEC2.RHO(38) = 1.95226 ; # 443
PLANT.PIPEC2.RHO(39) = 1.93964 ; # 444
PLANT.PIPEC2.RHO(40) = 1.92741 ; # 445
PLANT.PIPEC2.F(1) = 43.3743 ; # 446
PLANT.PIPEC2.F(2) = 43.3743 ; # 447
PLANT.PIPEC2.F(3) = 43.3743 ; # 448
PLANT.PIPEC2.F(4) = 43.3743 ; # 449
PLANT.PIPEC2.F(5) = 43.3743 ; # 450
PLANT.PIPEC2.F(6) = 43.3743 ; # 451
PLANT.PIPEC2.F(7) = 43.3743 ; # 452
PLANT.PIPEC2.F(8) = 43.3743 ; # 453
PLANT.PIPEC2.F(9) = 43.3743 ; # 454
PLANT.PIPEC2.F(10) = 43.3743 ; # 455
PLANT.PIPEC2.F(11) = 43.3743 ; # 456
PLANT.PIPEC2.F(12) = 43.3743 ; # 457
PLANT.PIPEC2.F(13) = 43.3743 ; # 458
PLANT.PIPEC2.F(14) = 43.3743 ; # 459
PLANT.PIPEC2.F(15) = 43.3743 ; # 460
PLANT.PIPEC2.F(16) = 43.3743 ; # 461
PLANT.PIPEC2.F(17) = 43.3743 ; # 462
PLANT.PIPEC2.F(18) = 43.3743 ; # 463
PLANT.PIPEC2.F(19) = 43.3743 ; # 464
PLANT.PIPEC2.F(20) = 43.3743 ; # 465
PLANT.PIPEC2.F(21) = 43.3743 ; # 466
PLANT.PIPEC2.F(22) = 43.3743 ; # 467
PLANT.PIPEC2.F(23) = 43.3743 ; # 468
PLANT.PIPEC2.F(24) = 43.3743 ; # 469
PLANT.PIPEC2.F(25) = 43.3743 ; # 470
PLANT.PIPEC2.F(26) = 43.3743 ; # 471
PLANT.PIPEC2.F(27) = 43.3743 ; # 472
PLANT.PIPEC2.F(28) = 43.3743 ; # 473
PLANT.PIPEC2.F(29) = 43.3743 ; # 474
PLANT.PIPEC2.F(30) = 43.3743 ; # 475
PLANT.PIPEC2.F(31) = 43.3743 ; # 476
PLANT.PIPEC2.F(32) = 43.3743 ; # 477
PLANT.PIPEC2.F(33) = 43.3743 ; # 478
PLANT.PIPEC2.F(34) = 43.3743 ; # 479
PLANT.PIPEC2.F(35) = 43.3743 ; # 480
PLANT.PIPEC2.F(36) = 43.3743 ; # 481
PLANT.PIPEC2.F(37) = 43.3743 ; # 482
```

```
PLANT.PIPEC2.F(38) = 43.3743 ; # 483
PLANT.PIPEC2.F(39) = 43.3743 ; # 484
PLANT.PIPEC2.F(40) = 43.3743 ; # 485
PLANT.PIPEC2.F_IN = 43.3743 ; # 486
PLANT.PIPEC2.PSI(1) = 1.81063e+006 ; # 487
PLANT.PIPEC2.PSI(2) = 1.91725e+006 ; # 488
PLANT.PIPEC2.PSI(3) = 2.00881e+006 ; # 489
PLANT.PIPEC2.PSI(4) = 2.08797e+006 ; # 490
PLANT.PIPEC2.PSI(5) = 2.1568e+006 ; # 491
PLANT.PIPEC2.PSI(6) = 2.21692e+006 ; # 492
PLANT.PIPEC2.PSI(7) = 2.26963e+006 ; # 493
PLANT.PIPEC2.PSI(8) = 2.31599e+006 ; # 494
PLANT.PIPEC2.PSI(9) = 2.35686e+006 ; # 495
PLANT.PIPEC2.PSI(10) = 2.39296e+006 ; # 496
PLANT.PIPEC2.PSI(11) = 2.42488e+006 ; # 497
PLANT.PIPEC2.PSI(12) = 2.45312e+006 ; # 498
PLANT.PIPEC2.PSI(13) = 2.47811e+006 ; # 499
PLANT.PIPEC2.PSI(14) = 2.50021e+006 ; # 500
PLANT.PIPEC2.PSI(15) = 2.51973e+006 ; # 501
PLANT.PIPEC2.PSI(16) = 2.53693e+006 ; # 502
PLANT.PIPEC2.PSI(17) = 2.55205e+006 ; # 503
PLANT.PIPEC2.PSI(18) = 2.5653e+006 ; # 504
PLANT.PIPEC2.PSI(19) = 2.57684e+006 ; # 505
PLANT.PIPEC2.PSI(20) = 2.58684e+006 ; # 506
PLANT.PIPEC2.PSI(21) = 2.59544e+006 ; # 507
PLANT.PIPEC2.PSI(22) = 2.60275e+006 ; # 508
PLANT.PIPEC2.PSI(23) = 2.60889e+006 ; # 509
PLANT.PIPEC2.PSI(24) = 2.61395e+006 ; # 510
PLANT.PIPEC2.PSI(25) = 2.61803e+006 ; # 511
PLANT.PIPEC2.PSI(26) = 2.62119e+006 ; # 512
PLANT.PIPEC2.PSI(27) = 2.62352e+006 ; # 513
PLANT.PIPEC2.PSI(28) = 2.62507e+006 ; # 514
PLANT.PIPEC2.PSI(29) = 2.62591e+006 ; # 515
PLANT.PIPEC2.PSI(30) = 2.62607e+006 ; # 516
PLANT.PIPEC2.PSI(31) = 2.62562e+006 ; # 517
PLANT.PIPEC2.PSI(32) = 2.6246e+006 ; # 518
PLANT.PIPEC2.PSI(33) = 2.62304e+006 ; # 519
PLANT.PIPEC2.PSI(34) = 2.62099e+006 ; # 520
PLANT.PIPEC2.PSI(35) = 2.61847e+006 ; # 521
PLANT.PIPEC2.PSI(36) = 2.61551e+006 ; # 522
PLANT.PIPEC2.PSI(37) = 2.61215e+006 ; # 523
PLANT.PIPEC2.PSI(38) = 2.6084e+006 ; # 524
PLANT.PIPEC2.PSI(39) = 2.6043e+006 ; # 525
PLANT.PIPEC2.PSI(40) = 2.59986e+006 ; # 526
```

HTL0MACH_SS_IN_PRE.PRESETS

```
# Values for computation HTL0MACH_SS
# Saved at time 1000
PLANT.HX.TGASHOT(1) := 908.671 ; # 1
PLANT.HX.TGASHOT(2) := 920.116 ; # 2
PLANT.HX.TGASHOT(3) := 931.225 ; # 3
PLANT.HX.TGASHOT(4) := 941.988 ; # 4
PLANT.HX.TGASHOT(5) := 952.416 ; # 5
PLANT.HX.TGASHOT(6) := 962.521 ; # 6
PLANT.HX.TGASHOT(7) := 972.311 ; # 7
PLANT.HX.TGASHOT(8) := 981.796 ; # 8
PLANT.HX.TGASHOT(9) := 990.987 ; # 9
PLANT.HX.TGASHOT(10) := 999.892 ; # 10
PLANT.HX.TGASHOT(11) := 1008.52 ; # 11
PLANT.HX.TGASHOT(12) := 1016.88 ; # 12
PLANT.HX.TGASHOT(13) := 1024.98 ; # 13
PLANT.HX.TGASHOT(14) := 1032.83 ; # 14
PLANT.HX.TGASHOT(15) := 1040.43 ; # 15
PLANT.HX.TGASHOT(16) := 1047.8 ; # 16
PLANT.HX.TGASHOT(17) := 1054.94 ; # 17
PLANT.HX.TGASHOT(18) := 1061.85 ; # 18
PLANT.HX.TGASHOT(19) := 1068.55 ; # 19
PLANT.HX.TGASHOT(20) := 1075.05 ; # 20
PLANT.HX.TGASHOT(21) := 1081.34 ; # 21
PLANT.HX.TGASHOT(22) := 1087.43 ; # 22
PLANT.HX.TGASHOT(23) := 1093.34 ; # 23
PLANT.HX.TGASHOT(24) := 1099.06 ; # 24
PLANT.HX.TGASHOT(25) := 1104.6 ; # 25
PLANT.HX.TGASHOT(26) := 1109.98 ; # 26
PLANT.HX.TGASHOT(27) := 1115.18 ; # 27
PLANT.HX.TGASHOT(28) := 1120.22 ; # 28
PLANT.HX.TGASHOT(29) := 1125.11 ; # 29
PLANT.HX.TGASHOT(30) := 1129.84 ; # 30
PLANT.HX.TGASHOT(31) := 1134.43 ; # 31
PLANT.HX.TGASHOT(32) := 1138.87 ; # 32
PLANT.HX.TGASHOT(33) := 1143.18 ; # 33
PLANT.HX.TGASHOT(34) := 1147.35 ; # 34
```

```
PLANT.HX.TGASHOT(35) := 1151.39 ; # 35
PLANT.HX.TGASHOT(36) := 1155.31 ; # 36
PLANT.HX.TGASHOT(37) := 1159.11 ; # 37
PLANT.HX.TGASHOT(38) := 1162.78 ; # 38
PLANT.HX.TGASHOT(39) := 1166.35 ; # 39
PLANT.HX.TGASHOT(40) := 1169.8 ; # 40
PLANT.HX.TGASCOLD(1) := 849.368 ; # 41
PLANT.HX.TGASCOLD(2) := 862.676 ; # 42
PLANT.HX.TGASCOLD(3) := 875.571 ; # 43
PLANT.HX.TGASCOLD(4) := 888.065 ; # 44
PLANT.HX.TGASCOLD(5) := 900.17 ; # 45
PLANT.HX.TGASCOLD(6) := 911.899 ; # 46
PLANT.HX.TGASCOLD(7) := 923.264 ; # 47
PLANT.HX.TGASCOLD(8) := 934.274 ; # 48
PLANT.HX.TGASCOLD(9) := 944.943 ; # 49
PLANT.HX.TGASCOLD(10) := 955.279 ; # 50
PLANT.HX.TGASCOLD(11) := 965.295 ; # 51
PLANT.HX.TGASCOLD(12) := 974.998 ; # 52
PLANT.HX.TGASCOLD(13) := 984.4 ; # 53
PLANT.HX.TGASCOLD(14) := 993.51 ; # 54
PLANT.HX.TGASCOLD(15) := 1002.34 ; # 55
PLANT.HX.TGASCOLD(16) := 1010.89 ; # 56
PLANT.HX.TGASCOLD(17) := 1019.17 ; # 57
PLANT.HX.TGASCOLD(18) := 1027.2 ; # 58
PLANT.HX.TGASCOLD(19) := 1034.98 ; # 59
PLANT.HX.TGASCOLD(20) := 1042.52 ; # 60
PLANT.HX.TGASCOLD(21) := 1049.82 ; # 61
PLANT.HX.TGASCOLD(22) := 1056.9 ; # 62
PLANT.HX.TGASCOLD(23) := 1063.75 ; # 63
PLANT.HX.TGASCOLD(24) := 1070.39 ; # 64
PLANT.HX.TGASCOLD(25) := 1076.83 ; # 65
PLANT.HX.TGASCOLD(26) := 1083.06 ; # 66
PLANT.HX.TGASCOLD(27) := 1089.11 ; # 67
PLANT.HX.TGASCOLD(28) := 1094.96 ; # 68
PLANT.HX.TGASCOLD(29) := 1100.63 ; # 69
PLANT.HX.TGASCOLD(30) := 1106.13 ; # 70
PLANT.HX.TGASCOLD(31) := 1111.45 ; # 71
PLANT.HX.TGASCOLD(32) := 1116.61 ; # 72
PLANT.HX.TGASCOLD(33) := 1121.61 ; # 73
PLANT.HX.TGASCOLD(34) := 1126.45 ; # 74
PLANT.HX.TGASCOLD(35) := 1131.14 ; # 75
PLANT.HX.TGASCOLD(36) := 1135.69 ; # 76
PLANT.HX.TGASCOLD(37) := 1140.09 ; # 77
PLANT.HX.TGASCOLD(38) := 1144.36 ; # 78
PLANT.HX.TGASCOLD(39) := 1148.5 ; # 79
PLANT.HX.TGASCOLD(40) := 1152.5 ; # 80
PLANT.HX.UHOT := 1738.41 ; # 81
PLANT.HX.UCOLD := 1645.45 ; # 82
PLANT.HX.THX(1) := 879.893 ; # 83
PLANT.HX.THX(2) := 892.184 ; # 84
PLANT.HX.THX(3) := 904.161 ; # 85
PLANT.HX.THX(4) := 915.766 ; # 86
PLANT.HX.THX(5) := 927.009 ; # 87
PLANT.HX.THX(6) := 937.904 ; # 88
PLANT.HX.THX(7) := 948.459 ; # 89
PLANT.HX.THX(8) := 958.687 ; # 90
PLANT.HX.THX(9) := 968.596 ; # 91
PLANT.HX.THX(10) := 978.197 ; # 92
PLANT.HX.THX(11) := 987.5 ; # 93
PLANT.HX.THX(12) := 996.513 ; # 94
PLANT.HX.THX(13) := 1005.25 ; # 95
PLANT.HX.THX(14) := 1013.71 ; # 96
PLANT.HX.THX(15) := 1021.91 ; # 97
PLANT.HX.THX(16) := 1029.85 ; # 98
PLANT.HX.THX(17) := 1037.55 ; # 99
PLANT.HX.THX(18) := 1045 ; # 100
PLANT.HX.THX(19) := 1052.23 ; # 101
PLANT.HX.THX(20) := 1059.23 ; # 102
PLANT.HX.THX(21) := 1066.01 ; # 103
PLANT.HX.THX(22) := 1072.58 ; # 104
PLANT.HX.THX(23) := 1078.95 ; # 105
PLANT.HX.THX(24) := 1085.12 ; # 106
PLANT.HX.THX(25) := 1091.1 ; # 107
PLANT.HX.THX(26) := 1096.89 ; # 108
PLANT.HX.THX(27) := 1102.5 ; # 109
PLANT.HX.THX(28) := 1107.94 ; # 110
PLANT.HX.THX(29) := 1113.21 ; # 111
PLANT.HX.THX(30) := 1118.31 ; # 112
PLANT.HX.THX(31) := 1123.26 ; # 113
PLANT.HX.THX(32) := 1128.05 ; # 114
PLANT.HX.THX(33) := 1132.69 ; # 115
PLANT.HX.THX(34) := 1137.19 ; # 116
PLANT.HX.THX(35) := 1141.55 ; # 117
PLANT.HX.THX(36) := 1145.77 ; # 118
PLANT.HX.THX(37) := 1149.86 ; # 119
PLANT.HX.THX(38) := 1153.83 ; # 120
PLANT.HX.THX(39) := 1157.67 ; # 121
PLANT.HX.THX(40) := 1161.37 ; # 122
PLANT.LOOP1.RHO(1) := 1.15317 ; # 123
PLANT.LOOP1.RHO(2) := 1.15067 ; # 124
PLANT.LOOP1.RHO(3) := 1.14817 ; # 125
PLANT.LOOP1.RHO(4) := 1.14565 ; # 126
```

```
PLANT.LOOP1.RHO(5) := 1.14313 ; # 127
PLANT.LOOP1.RHO(6) := 1.14059 ; # 128
PLANT.LOOP1.RHO(7) := 1.13804 ; # 129
PLANT.LOOP1.RHO(8) := 1.13549 ; # 130
PLANT.LOOP1.RHO(9) := 1.13292 ; # 131
PLANT.LOOP1.RHO(10) := 1.13034 ; # 132
PLANT.LOOP1.RHO(11) := 1.12115 ; # 133
PLANT.LOOP1.RHO(12) := 1.1026 ; # 134
PLANT.LOOP1.RHO(13) := 1.08477 ; # 135
PLANT.LOOP1.RHO(14) := 1.06791 ; # 136
PLANT.LOOP1.RHO(15) := 1.05196 ; # 137
PLANT.LOOP1.RHO(16) := 1.03683 ; # 138
PLANT.LOOP1.RHO(17) := 1.02247 ; # 139
PLANT.LOOP1.RHO(18) := 1.00882 ; # 140
PLANT.LOOP1.RHO(19) := 0.995831 ; # 141
PLANT.LOOP1.RHO(20) := 0.983453 ; # 142
PLANT.LOOP1.RHO(21) := 0.971645 ; # 143
PLANT.LOOP1.RHO(22) := 0.960369 ; # 144
PLANT.LOOP1.RHO(23) := 0.949588 ; # 145
PLANT.LOOP1.RHO(24) := 0.939272 ; # 146
PLANT.LOOP1.RHO(25) := 0.929389 ; # 147
PLANT.LOOP1.RHO(26) := 0.919912 ; # 148
PLANT.LOOP1.RHO(27) := 0.910817 ; # 149
PLANT.LOOP1.RHO(28) := 0.90208 ; # 150
PLANT.LOOP1.RHO(29) := 0.89368 ; # 151
PLANT.LOOP1.RHO(30) := 0.885596 ; # 152
PLANT.LOOP1.RHO(31) := 0.87781 ; # 153
PLANT.LOOP1.RHO(32) := 0.870305 ; # 154
PLANT.LOOP1.RHO(33) := 0.863066 ; # 155
PLANT.LOOP1.RHO(34) := 0.856077 ; # 156
PLANT.LOOP1.RHO(35) := 0.849325 ; # 157
PLANT.LOOP1.RHO(36) := 0.842796 ; # 158
PLANT.LOOP1.RHO(37) := 0.83648 ; # 159
PLANT.LOOP1.RHO(38) := 0.830364 ; # 160
PLANT.LOOP1.RHO(39) := 0.824437 ; # 161
PLANT.LOOP1.RHO(40) := 0.818691 ; # 162
PLANT.LOOP1.RHO(41) := 0.813116 ; # 163
PLANT.LOOP1.RHO(42) := 0.807703 ; # 164
PLANT.LOOP1.RHO(43) := 0.802444 ; # 165
PLANT.LOOP1.RHO(44) := 0.797332 ; # 166
PLANT.LOOP1.RHO(45) := 0.792359 ; # 167
PLANT.LOOP1.RHO(46) := 0.787518 ; # 168
PLANT.LOOP1.RHO(47) := 0.782803 ; # 169
PLANT.LOOP1.RHO(48) := 0.778208 ; # 170
PLANT.LOOP1.RHO(49) := 0.773727 ; # 171
PLANT.LOOP1.RHO(50) := 0.769359 ; # 172
PLANT.LOOP1.RHO(51) := 0.757959 ; # 173
PLANT.LOOP1.RHO(52) := 0.756103 ; # 174
PLANT.LOOP1.RHO(53) := 0.754256 ; # 175
PLANT.LOOP1.RHO(54) := 0.752399 ; # 176
PLANT.LOOP1.RHO(55) := 0.750533 ; # 177
PLANT.LOOP1.RHO(56) := 0.748658 ; # 178
PLANT.LOOP1.RHO(57) := 0.746773 ; # 179
PLANT.LOOP1.RHO(58) := 0.744879 ; # 180
PLANT.LOOP1.RHO(59) := 0.742975 ; # 181
PLANT.LOOP1.RHO(60) := 0.741061 ; # 182
PLANT.LOOP1.RHO(61) := 0.750585 ; # 183
PLANT.LOOP1.RHO(62) := 0.751689 ; # 184
PLANT.LOOP1.RHO(63) := 0.752856 ; # 185
PLANT.LOOP1.RHO(64) := 0.754135 ; # 186
PLANT.LOOP1.RHO(65) := 0.755533 ; # 187
PLANT.LOOP1.RHO(66) := 0.757059 ; # 188
PLANT.LOOP1.RHO(67) := 0.75872 ; # 189
PLANT.LOOP1.RHO(68) := 0.760528 ; # 190
PLANT.LOOP1.RHO(69) := 0.762492 ; # 191
PLANT.LOOP1.RHO(70) := 0.764623 ; # 192
PLANT.LOOP1.RHO(71) := 0.766934 ; # 193
PLANT.LOOP1.RHO(72) := 0.769438 ; # 194
PLANT.LOOP1.RHO(73) := 0.772149 ; # 195
PLANT.LOOP1.RHO(74) := 0.775083 ; # 196
PLANT.LOOP1.RHO(75) := 0.778256 ; # 197
PLANT.LOOP1.RHO(76) := 0.781688 ; # 198
PLANT.LOOP1.RHO(77) := 0.785399 ; # 199
PLANT.LOOP1.RHO(78) := 0.789411 ; # 200
PLANT.LOOP1.RHO(79) := 0.793749 ; # 201
PLANT.LOOP1.RHO(80) := 0.798441 ; # 202
PLANT.LOOP1.RHO(81) := 0.803515 ; # 203
PLANT.LOOP1.RHO(82) := 0.809005 ; # 204
PLANT.LOOP1.RHO(83) := 0.814949 ; # 205
PLANT.LOOP1.RHO(84) := 0.821387 ; # 206
PLANT.LOOP1.RHO(85) := 0.828365 ; # 207
PLANT.LOOP1.RHO(86) := 0.835933 ; # 208
PLANT.LOOP1.RHO(87) := 0.844151 ; # 209
PLANT.LOOP1.RHO(88) := 0.853082 ; # 210
PLANT.LOOP1.RHO(89) := 0.8628 ; # 211
PLANT.LOOP1.RHO(90) := 0.873389 ; # 212
PLANT.LOOP1.RHO(91) := 0.884944 ; # 213
PLANT.LOOP1.RHO(92) := 0.897573 ; # 214
PLANT.LOOP1.RHO(93) := 0.911403 ; # 215
PLANT.LOOP1.RHO(94) := 0.926579 ; # 216
PLANT.LOOP1.RHO(95) := 0.94327 ; # 217
PLANT.LOOP1.RHO(96) := 0.961674 ; # 218
```

```
PLANT.LOOP1.RHO(97) := 0.982027 ; # 219
PLANT.LOOP1.RHO(98) := 1.0046 ; # 220
PLANT.LOOP1.RHO(99) := 1.02974 ; # 221
PLANT.LOOP1.RHO(100) := 1.05783 ; # 222
PLANT.LOOP1.P(1) := 2.01181e+006 ; # 223
PLANT.LOOP1.P(2) := 2.00646e+006 ; # 224
PLANT.LOOP1.P(3) := 2.00109e+006 ; # 225
PLANT.LOOP1.P(4) := 1.99571e+006 ; # 226
PLANT.LOOP1.P(5) := 1.99032e+006 ; # 227
PLANT.LOOP1.P(6) := 1.98491e+006 ; # 228
PLANT.LOOP1.P(7) := 1.9795e+006 ; # 229
PLANT.LOOP1.P(8) := 1.97407e+006 ; # 230
PLANT.LOOP1.P(9) := 1.96863e+006 ; # 231
PLANT.LOOP1.P(10) := 1.96317e+006 ; # 232
PLANT.LOOP1.P(11) := 1.97929e+006 ; # 233
PLANT.LOOP1.P(12) := 1.97704e+006 ; # 234
PLANT.LOOP1.P(13) := 1.97414e+006 ; # 235
PLANT.LOOP1.P(14) := 1.9712e+006 ; # 236
PLANT.LOOP1.P(15) := 1.96822e+006 ; # 237
PLANT.LOOP1.P(16) := 1.96519e+006 ; # 238
PLANT.LOOP1.P(17) := 1.96212e+006 ; # 239
PLANT.LOOP1.P(18) := 1.95902e+006 ; # 240
PLANT.LOOP1.P(19) := 1.95587e+006 ; # 241
PLANT.LOOP1.P(20) := 1.95269e+006 ; # 242
PLANT.LOOP1.P(21) := 1.94947e+006 ; # 243
PLANT.LOOP1.P(22) := 1.94622e+006 ; # 244
PLANT.LOOP1.P(23) := 1.94293e+006 ; # 245
PLANT.LOOP1.P(24) := 1.93961e+006 ; # 246
PLANT.LOOP1.P(25) := 1.93625e+006 ; # 247
PLANT.LOOP1.P(26) := 1.93286e+006 ; # 248
PLANT.LOOP1.P(27) := 1.92943e+006 ; # 249
PLANT.LOOP1.P(28) := 1.92598e+006 ; # 250
PLANT.LOOP1.P(29) := 1.92249e+006 ; # 251
PLANT.LOOP1.P(30) := 1.91897e+006 ; # 252
PLANT.LOOP1.P(31) := 1.91543e+006 ; # 253
PLANT.LOOP1.P(32) := 1.91185e+006 ; # 254
PLANT.LOOP1.P(33) := 1.90824e+006 ; # 255
PLANT.LOOP1.P(34) := 1.90461e+006 ; # 256
PLANT.LOOP1.P(35) := 1.90095e+006 ; # 257
PLANT.LOOP1.P(36) := 1.89726e+006 ; # 258
PLANT.LOOP1.P(37) := 1.89354e+006 ; # 259
PLANT.LOOP1.P(38) := 1.8898e+006 ; # 260
PLANT.LOOP1.P(39) := 1.88603e+006 ; # 261
PLANT.LOOP1.P(40) := 1.88224e+006 ; # 262
PLANT.LOOP1.P(41) := 1.87842e+006 ; # 263
PLANT.LOOP1.P(42) := 1.87458e+006 ; # 264
PLANT.LOOP1.P(43) := 1.87071e+006 ; # 265
PLANT.LOOP1.P(44) := 1.86682e+006 ; # 266
PLANT.LOOP1.P(45) := 1.8629e+006 ; # 267
PLANT.LOOP1.P(46) := 1.85896e+006 ; # 268
PLANT.LOOP1.P(47) := 1.855e+006 ; # 269
PLANT.LOOP1.P(48) := 1.85101e+006 ; # 270
PLANT.LOOP1.P(49) := 1.847e+006 ; # 271
PLANT.LOOP1.P(50) := 1.84297e+006 ; # 272
PLANT.LOOP1.P(51) := 1.81438e+006 ; # 273
PLANT.LOOP1.P(52) := 1.80865e+006 ; # 274
PLANT.LOOP1.P(53) := 1.80295e+006 ; # 275
PLANT.LOOP1.P(54) := 1.79723e+006 ; # 276
PLANT.LOOP1.P(55) := 1.7915e+006 ; # 277
PLANT.LOOP1.P(56) := 1.78576e+006 ; # 278
PLANT.LOOP1.P(57) := 1.78e+006 ; # 279
PLANT.LOOP1.P(58) := 1.77423e+006 ; # 280
PLANT.LOOP1.P(59) := 1.76844e+006 ; # 281
PLANT.LOOP1.P(60) := 1.76263e+006 ; # 282
PLANT.LOOP1.P(61) := 1.78151e+006 ; # 283
PLANT.LOOP1.P(62) := 1.78014e+006 ; # 284
PLANT.LOOP1.P(63) := 1.77866e+006 ; # 285
PLANT.LOOP1.P(64) := 1.7772e+006 ; # 286
PLANT.LOOP1.P(65) := 1.77573e+006 ; # 287
PLANT.LOOP1.P(66) := 1.77427e+006 ; # 288
PLANT.LOOP1.P(67) := 1.77281e+006 ; # 289
PLANT.LOOP1.P(68) := 1.77136e+006 ; # 290
PLANT.LOOP1.P(69) := 1.76991e+006 ; # 291
PLANT.LOOP1.P(70) := 1.76847e+006 ; # 292
PLANT.LOOP1.P(71) := 1.76703e+006 ; # 293
PLANT.LOOP1.P(72) := 1.7656e+006 ; # 294
PLANT.LOOP1.P(73) := 1.76417e+006 ; # 295
PLANT.LOOP1.P(74) := 1.76275e+006 ; # 296
PLANT.LOOP1.P(75) := 1.76134e+006 ; # 297
PLANT.LOOP1.P(76) := 1.75993e+006 ; # 298
PLANT.LOOP1.P(77) := 1.75854e+006 ; # 299
PLANT.LOOP1.P(78) := 1.75715e+006 ; # 300
PLANT.LOOP1.P(79) := 1.75577e+006 ; # 301
PLANT.LOOP1.P(80) := 1.75441e+006 ; # 302
PLANT.LOOP1.P(81) := 1.75305e+006 ; # 303
PLANT.LOOP1.P(82) := 1.7517e+006 ; # 304
PLANT.LOOP1.P(83) := 1.75037e+006 ; # 305
PLANT.LOOP1.P(84) := 1.74905e+006 ; # 306
PLANT.LOOP1.P(85) := 1.74774e+006 ; # 307
PLANT.LOOP1.P(86) := 1.74645e+006 ; # 308
PLANT.LOOP1.P(87) := 1.74518e+006 ; # 309
PLANT.LOOP1.P(88) := 1.74392e+006 ; # 310
```

```
PLANT.LOOP1.P(89) := 1.74267e+006 ; # 311
PLANT.LOOP1.P(90) := 1.74145e+006 ; # 312
PLANT.LOOP1.P(91) := 1.74025e+006 ; # 313
PLANT.LOOP1.P(92) := 1.73907e+006 ; # 314
PLANT.LOOP1.P(93) := 1.73791e+006 ; # 315
PLANT.LOOP1.P(94) := 1.73677e+006 ; # 316
PLANT.LOOP1.P(95) := 1.73566e+006 ; # 317
PLANT.LOOP1.P(96) := 1.73458e+006 ; # 318
PLANT.LOOP1.P(97) := 1.73353e+006 ; # 319
PLANT.LOOP1.P(98) := 1.7325e+006 ; # 320
PLANT.LOOP1.P(99) := 1.73151e+006 ; # 321
PLANT.LOOP1.P(100) := 1.73056e+006 ; # 322
PLANT.LOOP1.P_OUT := 1.73003e+006 ; # 323
PLANT.LOOP1.H(1) := 2.81232e+006 ; # 324
PLANT.LOOP1.H(2) := 2.81015e+006 ; # 325
PLANT.LOOP1.H(3) := 2.80798e+006 ; # 326
PLANT.LOOP1.H(4) := 2.80581e+006 ; # 327
PLANT.LOOP1.H(5) := 2.80364e+006 ; # 328
PLANT.LOOP1.H(6) := 2.80148e+006 ; # 329
PLANT.LOOP1.H(7) := 2.79931e+006 ; # 330
PLANT.LOOP1.H(8) := 2.79715e+006 ; # 331
PLANT.LOOP1.H(9) := 2.79499e+006 ; # 332
PLANT.LOOP1.H(10) := 2.79283e+006 ; # 333
PLANT.LOOP1.H(11) := 2.86435e+006 ; # 334
PLANT.LOOP1.H(12) := 2.93349e+006 ; # 335
PLANT.LOOP1.H(13) := 3.00048e+006 ; # 336
PLANT.LOOP1.H(14) := 3.06539e+006 ; # 337
PLANT.LOOP1.H(15) := 3.12828e+006 ; # 338
PLANT.LOOP1.H(16) := 3.18921e+006 ; # 339
PLANT.LOOP1.H(17) := 3.24824e+006 ; # 340
PLANT.LOOP1.H(18) := 3.30545e+006 ; # 341
PLANT.LOOP1.H(19) := 3.36087e+006 ; # 342
PLANT.LOOP1.H(20) := 3.41457e+006 ; # 343
PLANT.LOOP1.H(21) := 3.4666e+006 ; # 344
PLANT.LOOP1.H(22) := 3.51701e+006 ; # 345
PLANT.LOOP1.H(23) := 3.56585e+006 ; # 346
PLANT.LOOP1.H(24) := 3.61317e+006 ; # 347
PLANT.LOOP1.H(25) := 3.65903e+006 ; # 348
PLANT.LOOP1.H(26) := 3.70345e+006 ; # 349
PLANT.LOOP1.H(27) := 3.7465e+006 ; # 350
PLANT.LOOP1.H(28) := 3.78821e+006 ; # 351
PLANT.LOOP1.H(29) := 3.82862e+006 ; # 352
PLANT.LOOP1.H(30) := 3.86777e+006 ; # 353
PLANT.LOOP1.H(31) := 3.90571e+006 ; # 354
PLANT.LOOP1.H(32) := 3.94246e+006 ; # 355
PLANT.LOOP1.H(33) := 3.97808e+006 ; # 356
PLANT.LOOP1.H(34) := 4.01258e+006 ; # 357
PLANT.LOOP1.H(35) := 4.04601e+006 ; # 358
PLANT.LOOP1.H(36) := 4.07841e+006 ; # 359
PLANT.LOOP1.H(37) := 4.10979e+006 ; # 360
PLANT.LOOP1.H(38) := 4.1402e+006 ; # 361
PLANT.LOOP1.H(39) := 4.16967e+006 ; # 362
PLANT.LOOP1.H(40) := 4.19821e+006 ; # 363
PLANT.LOOP1.H(41) := 4.22588e+006 ; # 364
PLANT.LOOP1.H(42) := 4.25268e+006 ; # 365
PLANT.LOOP1.H(43) := 4.27864e+006 ; # 366
PLANT.LOOP1.H(44) := 4.3038e+006 ; # 367
PLANT.LOOP1.H(45) := 4.32818e+006 ; # 368
PLANT.LOOP1.H(46) := 4.3518e+006 ; # 369
PLANT.LOOP1.H(47) := 4.37468e+006 ; # 370
PLANT.LOOP1.H(48) := 4.39685e+006 ; # 371
PLANT.LOOP1.H(49) := 4.41834e+006 ; # 372
PLANT.LOOP1.H(50) := 4.43912e+006 ; # 373
PLANT.LOOP1.H(51) := 4.43487e+006 ; # 374
PLANT.LOOP1.H(52) := 4.43062e+006 ; # 375
PLANT.LOOP1.H(53) := 4.42637e+006 ; # 376
PLANT.LOOP1.H(54) := 4.42213e+006 ; # 377
PLANT.LOOP1.H(55) := 4.41789e+006 ; # 378
PLANT.LOOP1.H(56) := 4.41366e+006 ; # 379
PLANT.LOOP1.H(57) := 4.40943e+006 ; # 380
PLANT.LOOP1.H(58) := 4.40521e+006 ; # 381
PLANT.LOOP1.H(59) := 4.40099e+006 ; # 382
PLANT.LOOP1.H(60) := 4.39677e+006 ; # 383
PLANT.LOOP1.H(61) := 4.38421e+006 ; # 384
PLANT.LOOP1.H(62) := 4.37092e+006 ; # 385
PLANT.LOOP1.H(63) := 4.35686e+006 ; # 386
PLANT.LOOP1.H(64) := 4.34197e+006 ; # 387
PLANT.LOOP1.H(65) := 4.32623e+006 ; # 388
PLANT.LOOP1.H(66) := 4.30957e+006 ; # 389
PLANT.LOOP1.H(67) := 4.29193e+006 ; # 390
PLANT.LOOP1.H(68) := 4.27328e+006 ; # 391
PLANT.LOOP1.H(69) := 4.25354e+006 ; # 392
PLANT.LOOP1.H(70) := 4.23265e+006 ; # 393
PLANT.LOOP1.H(71) := 4.21054e+006 ; # 394
PLANT.LOOP1.H(72) := 4.18715e+006 ; # 395
PLANT.LOOP1.H(73) := 4.1624e+006 ; # 396
PLANT.LOOP1.H(74) := 4.13621e+006 ; # 397
PLANT.LOOP1.H(75) := 4.1085e+006 ; # 398
PLANT.LOOP1.H(76) := 4.07917e+006 ; # 399
PLANT.LOOP1.H(77) := 4.04814e+006 ; # 400
PLANT.LOOP1.H(78) := 4.01531e+006 ; # 401
PLANT.LOOP1.H(79) := 3.98056e+006 ; # 402
```

```
405   PLANT.LOOP1.H(80) := 3.9438e+006 ; # 403
406   PLANT.LOOP1.H(81) := 3.90489e+006 ; # 404
407   PLANT.LOOP1.H(82) := 3.86373e+006 ; # 405
408   PLANT.LOOP1.H(83) := 3.82017e+006 ; # 406
409   PLANT.LOOP1.H(84) := 3.77408e+006 ; # 407
410   PLANT.LOOP1.H(85) := 3.7253e+006 ; # 408
411   PLANT.LOOP1.H(86) := 3.67369e+006 ; # 409
412   PLANT.LOOP1.H(87) := 3.61908e+006 ; # 410
413   PLANT.LOOP1.H(88) := 3.5613e+006 ; # 411
414   PLANT.LOOP1.H(89) := 3.50015e+006 ; # 412
415   PLANT.LOOP1.H(90) := 3.43544e+006 ; # 413
416   PLANT.LOOP1.H(91) := 3.36698e+006 ; # 414
417   PLANT.LOOP1.H(92) := 3.29453e+006 ; # 415
418   PLANT.LOOP1.H(93) := 3.21787e+006 ; # 416
419   PLANT.LOOP1.H(94) := 3.13675e+006 ; # 417
420   PLANT.LOOP1.H(95) := 3.05091e+006 ; # 418
421   PLANT.LOOP1.H(96) := 2.96008e+006 ; # 419
422   PLANT.LOOP1.H(97) := 2.86396e+006 ; # 420
423   PLANT.LOOP1.H(98) := 2.76226e+006 ; # 421
424   PLANT.LOOP1.H(99) := 2.65465e+006 ; # 422
425   PLANT.LOOP1.H(100) := 2.54078e+006 ; # 423
426   PLANT.LOOP1.H0 := 2.81449e+006 ; # 424
427   PLANT.LOOP1.T(1) := 839.351 ; # 425
428   PLANT.LOOP1.T(2) := 838.933 ; # 426
429   PLANT.LOOP1.T(3) := 838.516 ; # 427
430   PLANT.LOOP1.T(4) := 838.098 ; # 428
431   PLANT.LOOP1.T(5) := 837.681 ; # 429
432   PLANT.LOOP1.T(6) := 837.264 ; # 430
433   PLANT.LOOP1.T(7) := 836.847 ; # 431
434   PLANT.LOOP1.T(8) := 836.431 ; # 432
435   PLANT.LOOP1.T(9) := 836.015 ; # 433
436   PLANT.LOOP1.T(10) := 835.6 ; # 434
437   PLANT.LOOP1.T(11) := 849.368 ; # 435
438   PLANT.LOOP1.T(12) := 862.676 ; # 436
439   PLANT.LOOP1.T(13) := 875.571 ; # 437
440   PLANT.LOOP1.T(14) := 888.065 ; # 438
441   PLANT.LOOP1.T(15) := 900.17 ; # 439
442   PLANT.LOOP1.T(16) := 911.899 ; # 440
443   PLANT.LOOP1.T(17) := 923.264 ; # 441
444   PLANT.LOOP1.T(18) := 934.274 ; # 442
445   PLANT.LOOP1.T(19) := 944.943 ; # 443
446   PLANT.LOOP1.T(20) := 955.279 ; # 444
447   PLANT.LOOP1.T(21) := 965.295 ; # 445
448   PLANT.LOOP1.T(22) := 974.998 ; # 446
449   PLANT.LOOP1.T(23) := 984.4 ; # 447
450   PLANT.LOOP1.T(24) := 993.51 ; # 448
451   PLANT.LOOP1.T(25) := 1002.34 ; # 449
452   PLANT.LOOP1.T(26) := 1010.89 ; # 450
453   PLANT.LOOP1.T(27) := 1019.17 ; # 451
454   PLANT.LOOP1.T(28) := 1027.2 ; # 452
455   PLANT.LOOP1.T(29) := 1034.98 ; # 453
456   PLANT.LOOP1.T(30) := 1042.52 ; # 454
457   PLANT.LOOP1.T(31) := 1049.82 ; # 455
458   PLANT.LOOP1.T(32) := 1056.9 ; # 456
459   PLANT.LOOP1.T(33) := 1063.75 ; # 457
460   PLANT.LOOP1.T(34) := 1070.39 ; # 458
461   PLANT.LOOP1.T(35) := 1076.83 ; # 459
462   PLANT.LOOP1.T(36) := 1083.06 ; # 460
463   PLANT.LOOP1.T(37) := 1089.11 ; # 461
464   PLANT.LOOP1.T(38) := 1094.96 ; # 462
465   PLANT.LOOP1.T(39) := 1100.63 ; # 463
466   PLANT.LOOP1.T(40) := 1106.13 ; # 464
467   PLANT.LOOP1.T(41) := 1111.45 ; # 465
468   PLANT.LOOP1.T(42) := 1116.61 ; # 466
469   PLANT.LOOP1.T(43) := 1121.61 ; # 467
470   PLANT.LOOP1.T(44) := 1126.45 ; # 468
471   PLANT.LOOP1.T(45) := 1131.14 ; # 469
472   PLANT.LOOP1.T(46) := 1135.69 ; # 470
473   PLANT.LOOP1.T(47) := 1140.09 ; # 471
474   PLANT.LOOP1.T(48) := 1144.36 ; # 472
475   PLANT.LOOP1.T(49) := 1148.5 ; # 473
476   PLANT.LOOP1.T(50) := 1152.5 ; # 474
477   PLANT.LOOP1.T(51) := 1151.68 ; # 475
478   PLANT.LOOP1.T(52) := 1150.86 ; # 476
479   PLANT.LOOP1.T(53) := 1150.04 ; # 477
480   PLANT.LOOP1.T(54) := 1149.23 ; # 478
481   PLANT.LOOP1.T(55) := 1148.41 ; # 479
482   PLANT.LOOP1.T(56) := 1147.6 ; # 480
483   PLANT.LOOP1.T(57) := 1146.78 ; # 481
484   PLANT.LOOP1.T(58) := 1145.97 ; # 482
485   PLANT.LOOP1.T(59) := 1145.16 ; # 483
486   PLANT.LOOP1.T(60) := 1144.35 ; # 484
487   PLANT.LOOP1.T(61) := 1141.93 ; # 485
488   PLANT.LOOP1.T(62) := 1139.37 ; # 486
489   PLANT.LOOP1.T(63) := 1136.66 ; # 487
490   PLANT.LOOP1.T(64) := 1133.8 ; # 488
491   PLANT.LOOP1.T(65) := 1130.77 ; # 489
492   PLANT.LOOP1.T(66) := 1127.56 ; # 490
493   PLANT.LOOP1.T(67) := 1124.17 ; # 491
494   PLANT.LOOP1.T(68) := 1120.57 ; # 492
495   PLANT.LOOP1.T(69) := 1116.77 ; # 493
496   PLANT.LOOP1.T(70) := 1112.75 ; # 494
```

```
PLANT.LOOP1.T(71) := 1108.5 ; # 495
PLANT.LOOP1.T(72) := 1104 ; # 496
PLANT.LOOP1.T(73) := 1099.23 ; # 497
PLANT.LOOP1.T(74) := 1094.19 ; # 498
PLANT.LOOP1.T(75) := 1088.86 ; # 499
PLANT.LOOP1.T(76) := 1083.21 ; # 500
PLANT.LOOP1.T(77) := 1077.24 ; # 501
PLANT.LOOP1.T(78) := 1070.92 ; # 502
PLANT.LOOP1.T(79) := 1064.23 ; # 503
PLANT.LOOP1.T(80) := 1057.15 ; # 504
PLANT.LOOP1.T(81) := 1049.66 ; # 505
PLANT.LOOP1.T(82) := 1041.74 ; # 506
PLANT.LOOP1.T(83) := 1033.36 ; # 507
PLANT.LOOP1.T(84) := 1024.48 ; # 508
PLANT.LOOP1.T(85) := 1015.09 ; # 509
PLANT.LOOP1.T(86) := 1005.16 ; # 510
PLANT.LOOP1.T(87) := 994.647 ; # 511
PLANT.LOOP1.T(88) := 983.524 ; # 512
PLANT.LOOP1.T(89) := 971.753 ; # 513
PLANT.LOOP1.T(90) := 959.298 ; # 514
PLANT.LOOP1.T(91) := 946.119 ; # 515
PLANT.LOOP1.T(92) := 932.173 ; # 516
PLANT.LOOP1.T(93) := 917.416 ; # 517
PLANT.LOOP1.T(94) := 901.801 ; # 518
PLANT.LOOP1.T(95) := 885.278 ; # 519
PLANT.LOOP1.T(96) := 867.793 ; # 520
PLANT.LOOP1.T(97) := 849.292 ; # 521
PLANT.LOOP1.T(98) := 829.716 ; # 522
PLANT.LOOP1.T(99) := 809 ; # 523
PLANT.LOOP1.T(100) := 787.082 ; # 524
PLANT.LOOP1.T0 := 839.77 ; # 525
PLANT.LOOP1.TEXTRNL(1) := 291.15 ; # 526
PLANT.LOOP1.TEXTRNL(2) := 291.15 ; # 527
PLANT.LOOP1.TEXTRNL(3) := 291.15 ; # 528
PLANT.LOOP1.TEXTRNL(4) := 291.15 ; # 529
PLANT.LOOP1.TEXTRNL(5) := 291.15 ; # 530
PLANT.LOOP1.TEXTRNL(6) := 291.15 ; # 531
PLANT.LOOP1.TEXTRNL(7) := 291.15 ; # 532
PLANT.LOOP1.TEXTRNL(8) := 291.15 ; # 533
PLANT.LOOP1.TEXTRNL(9) := 291.15 ; # 534
PLANT.LOOP1.TEXTRNL(10) := 291.15 ; # 535
PLANT.LOOP1.TEXTRNL(11) := 879.893 ; # 536
PLANT.LOOP1.TEXTRNL(12) := 892.184 ; # 537
PLANT.LOOP1.TEXTRNL(13) := 904.161 ; # 538
PLANT.LOOP1.TEXTRNL(14) := 915.766 ; # 539
PLANT.LOOP1.TEXTRNL(15) := 927.009 ; # 540
PLANT.LOOP1.TEXTRNL(16) := 937.904 ; # 541
PLANT.LOOP1.TEXTRNL(17) := 948.459 ; # 542
PLANT.LOOP1.TEXTRNL(18) := 958.687 ; # 543
PLANT.LOOP1.TEXTRNL(19) := 968.596 ; # 544
PLANT.LOOP1.TEXTRNL(20) := 978.197 ; # 545
PLANT.LOOP1.TEXTRNL(21) := 987.5 ; # 546
PLANT.LOOP1.TEXTRNL(22) := 996.513 ; # 547
PLANT.LOOP1.TEXTRNL(23) := 1005.25 ; # 548
PLANT.LOOP1.TEXTRNL(24) := 1013.71 ; # 549
PLANT.LOOP1.TEXTRNL(25) := 1021.91 ; # 550
PLANT.LOOP1.TEXTRNL(26) := 1029.85 ; # 551
PLANT.LOOP1.TEXTRNL(27) := 1037.55 ; # 552
PLANT.LOOP1.TEXTRNL(28) := 1045 ; # 553
PLANT.LOOP1.TEXTRNL(29) := 1052.23 ; # 554
PLANT.LOOP1.TEXTRNL(30) := 1059.23 ; # 555
PLANT.LOOP1.TEXTRNL(31) := 1066.01 ; # 556
PLANT.LOOP1.TEXTRNL(32) := 1072.58 ; # 557
PLANT.LOOP1.TEXTRNL(33) := 1078.95 ; # 558
PLANT.LOOP1.TEXTRNL(34) := 1085.12 ; # 559
PLANT.LOOP1.TEXTRNL(35) := 1091.1 ; # 560
PLANT.LOOP1.TEXTRNL(36) := 1096.89 ; # 561
PLANT.LOOP1.TEXTRNL(37) := 1102.5 ; # 562
PLANT.LOOP1.TEXTRNL(38) := 1107.94 ; # 563
PLANT.LOOP1.TEXTRNL(39) := 1113.21 ; # 564
PLANT.LOOP1.TEXTRNL(40) := 1118.31 ; # 565
PLANT.LOOP1.TEXTRNL(41) := 1123.26 ; # 566
PLANT.LOOP1.TEXTRNL(42) := 1128.05 ; # 567
PLANT.LOOP1.TEXTRNL(43) := 1132.69 ; # 568
PLANT.LOOP1.TEXTRNL(44) := 1137.19 ; # 569
PLANT.LOOP1.TEXTRNL(45) := 1141.55 ; # 570
PLANT.LOOP1.TEXTRNL(46) := 1145.77 ; # 571
PLANT.LOOP1.TEXTRNL(47) := 1149.86 ; # 572
PLANT.LOOP1.TEXTRNL(48) := 1153.83 ; # 573
PLANT.LOOP1.TEXTRNL(49) := 1157.67 ; # 574
PLANT.LOOP1.TEXTRNL(50) := 1161.37 ; # 575
PLANT.LOOP1.TEXTRNL(51) := 291.15 ; # 576
PLANT.LOOP1.TEXTRNL(52) := 291.15 ; # 577
PLANT.LOOP1.TEXTRNL(53) := 291.15 ; # 578
PLANT.LOOP1.TEXTRNL(54) := 291.15 ; # 579
PLANT.LOOP1.TEXTRNL(55) := 291.15 ; # 580
PLANT.LOOP1.TEXTRNL(56) := 291.15 ; # 581
PLANT.LOOP1.TEXTRNL(57) := 291.15 ; # 582
PLANT.LOOP1.TEXTRNL(58) := 291.15 ; # 583
PLANT.LOOP1.TEXTRNL(59) := 291.15 ; # 584
PLANT.LOOP1.TEXTRNL(60) := 291.15 ; # 585
PLANT.LOOP1.TEXTRNL(61) := 1133.71 ; # 586
```

```
PLANT.LOOP1.TEXTRNL(62) := 1130.67 ; # 587
PLANT.LOOP1.TEXTRNL(63) := 1127.46 ; # 588
PLANT.LOOP1.TEXTRNL(64) := 1124.06 ; # 589
PLANT.LOOP1.TEXTRNL(65) := 1120.46 ; # 590
PLANT.LOOP1.TEXTRNL(66) := 1116.65 ; # 591
PLANT.LOOP1.TEXTRNL(67) := 1112.63 ; # 592
PLANT.LOOP1.TEXTRNL(68) := 1108.36 ; # 593
PLANT.LOOP1.TEXTRNL(69) := 1103.85 ; # 594
PLANT.LOOP1.TEXTRNL(70) := 1099.08 ; # 595
PLANT.LOOP1.TEXTRNL(71) := 1094.03 ; # 596
PLANT.LOOP1.TEXTRNL(72) := 1088.69 ; # 597
PLANT.LOOP1.TEXTRNL(73) := 1083.03 ; # 598
PLANT.LOOP1.TEXTRNL(74) := 1077.05 ; # 599
PLANT.LOOP1.TEXTRNL(75) := 1070.72 ; # 600
PLANT.LOOP1.TEXTRNL(76) := 1064.02 ; # 601
PLANT.LOOP1.TEXTRNL(77) := 1056.93 ; # 602
PLANT.LOOP1.TEXTRNL(78) := 1049.43 ; # 603
PLANT.LOOP1.TEXTRNL(79) := 1041.49 ; # 604
PLANT.LOOP1.TEXTRNL(80) := 1033.09 ; # 605
PLANT.LOOP1.TEXTRNL(81) := 1024.2 ; # 606
PLANT.LOOP1.TEXTRNL(82) := 1014.8 ; # 607
PLANT.LOOP1.TEXTRNL(83) := 1004.84 ; # 608
PLANT.LOOP1.TEXTRNL(84) := 994.314 ; # 609
PLANT.LOOP1.TEXTRNL(85) := 983.171 ; # 610
PLANT.LOOP1.TEXTRNL(86) := 971.38 ; # 611
PLANT.LOOP1.TEXTRNL(87) := 958.903 ; # 612
PLANT.LOOP1.TEXTRNL(88) := 945.701 ; # 613
PLANT.LOOP1.TEXTRNL(89) := 931.731 ; # 614
PLANT.LOOP1.TEXTRNL(90) := 916.948 ; # 615
PLANT.LOOP1.TEXTRNL(91) := 901.306 ; # 616
PLANT.LOOP1.TEXTRNL(92) := 884.754 ; # 617
PLANT.LOOP1.TEXTRNL(93) := 867.239 ; # 618
PLANT.LOOP1.TEXTRNL(94) := 848.706 ; # 619
PLANT.LOOP1.TEXTRNL(95) := 829.095 ; # 620
PLANT.LOOP1.TEXTRNL(96) := 808.344 ; # 621
PLANT.LOOP1.TEXTRNL(97) := 786.385 ; # 622
PLANT.LOOP1.TEXTRNL(98) := 763.15 ; # 623
PLANT.LOOP1.TEXTRNL(99) := 738.563 ; # 624
PLANT.LOOP1.TEXTRNL(100) := 712.554 ; # 625
PLANT.LOOP1.PSI(1) := 1.23127e+006 ; # 626
PLANT.LOOP1.PSI(2) := 1.22711e+006 ; # 627
PLANT.LOOP1.PSI(3) := 1.22294e+006 ; # 628
PLANT.LOOP1.PSI(4) := 1.21877e+006 ; # 629
PLANT.LOOP1.PSI(5) := 1.2146e+006 ; # 630
PLANT.LOOP1.PSI(6) := 1.21042e+006 ; # 631
PLANT.LOOP1.PSI(7) := 1.20624e+006 ; # 632
PLANT.LOOP1.PSI(8) := 1.20206e+006 ; # 633
PLANT.LOOP1.PSI(9) := 1.19787e+006 ; # 634
PLANT.LOOP1.PSI(10) := 1.19368e+006 ; # 635
PLANT.LOOP1.PSI(11) := 1.23208e+006 ; # 636
PLANT.LOOP1.PSI(12) := 1.25743e+006 ; # 637
PLANT.LOOP1.PSI(13) := 1.28069e+006 ; # 638
PLANT.LOOP1.PSI(14) := 1.30237e+006 ; # 639
PLANT.LOOP1.PSI(15) := 1.3226e+006 ; # 640
PLANT.LOOP1.PSI(16) := 1.34148e+006 ; # 641
PLANT.LOOP1.PSI(17) := 1.35911e+006 ; # 642
PLANT.LOOP1.PSI(18) := 1.37558e+006 ; # 643
PLANT.LOOP1.PSI(19) := 1.39098e+006 ; # 644
PLANT.LOOP1.PSI(20) := 1.40537e+006 ; # 645
PLANT.LOOP1.PSI(21) := 1.41883e+006 ; # 646
PLANT.LOOP1.PSI(22) := 1.4314e+006 ; # 647
PLANT.LOOP1.PSI(23) := 1.44316e+006 ; # 648
PLANT.LOOP1.PSI(24) := 1.45415e+006 ; # 649
PLANT.LOOP1.PSI(25) := 1.46441e+006 ; # 650
PLANT.LOOP1.PSI(26) := 1.474e+006 ; # 651
PLANT.LOOP1.PSI(27) := 1.48294e+006 ; # 652
PLANT.LOOP1.PSI(28) := 1.49129e+006 ; # 653
PLANT.LOOP1.PSI(29) := 1.49907e+006 ; # 654
PLANT.LOOP1.PSI(30) := 1.50631e+006 ; # 655
PLANT.LOOP1.PSI(31) := 1.51304e+006 ; # 656
PLANT.LOOP1.PSI(32) := 1.5193e+006 ; # 657
PLANT.LOOP1.PSI(33) := 1.5251e+006 ; # 658
PLANT.LOOP1.PSI(34) := 1.53047e+006 ; # 659
PLANT.LOOP1.PSI(35) := 1.53543e+006 ; # 660
PLANT.LOOP1.PSI(36) := 1.54001e+006 ; # 661
PLANT.LOOP1.PSI(37) := 1.54421e+006 ; # 662
PLANT.LOOP1.PSI(38) := 1.54807e+006 ; # 663
PLANT.LOOP1.PSI(39) := 1.5516e+006 ; # 664
PLANT.LOOP1.PSI(40) := 1.5548e+006 ; # 665
PLANT.LOOP1.PSI(41) := 1.55771e+006 ; # 666
PLANT.LOOP1.PSI(42) := 1.56032e+006 ; # 667
PLANT.LOOP1.PSI(43) := 1.56266e+006 ; # 668
PLANT.LOOP1.PSI(44) := 1.56474e+006 ; # 669
PLANT.LOOP1.PSI(45) := 1.56657e+006 ; # 670
PLANT.LOOP1.PSI(46) := 1.56816e+006 ; # 671
PLANT.LOOP1.PSI(47) := 1.56951e+006 ; # 672
PLANT.LOOP1.PSI(48) := 1.57065e+006 ; # 673
PLANT.LOOP1.PSI(49) := 1.57158e+006 ; # 674
PLANT.LOOP1.PSI(50) := 1.5723e+006 ; # 675
PLANT.LOOP1.PSI(51) := 1.54707e+006 ; # 676
PLANT.LOOP1.PSI(52) := 1.54135e+006 ; # 677
PLANT.LOOP1.PSI(53) := 1.53567e+006 ; # 678
```

```
681   PLANT.LOOP1.PSI(54) := 1.52997e+006 ; # 679
682   PLANT.LOOP1.PSI(55) := 1.52427e+006 ; # 680
683   PLANT.LOOP1.PSI(56) := 1.51856e+006 ; # 681
684   PLANT.LOOP1.PSI(57) := 1.51284e+006 ; # 682
685   PLANT.LOOP1.PSI(58) := 1.50712e+006 ; # 683
686   PLANT.LOOP1.PSI(59) := 1.50138e+006 ; # 684
687   PLANT.LOOP1.PSI(60) := 1.49564e+006 ; # 685
688   PLANT.LOOP1.PSI(61) := 1.50921e+006 ; # 686
689   PLANT.LOOP1.PSI(62) := 1.50544e+006 ; # 687
690   PLANT.LOOP1.PSI(63) := 1.50142e+006 ; # 688
691   PLANT.LOOP1.PSI(64) := 1.49724e+006 ; # 689
692   PLANT.LOOP1.PSI(65) := 1.49288e+006 ; # 690
693   PLANT.LOOP1.PSI(66) := 1.48833e+006 ; # 691
694   PLANT.LOOP1.PSI(67) := 1.48357e+006 ; # 692
695   PLANT.LOOP1.PSI(68) := 1.47859e+006 ; # 693
696   PLANT.LOOP1.PSI(69) := 1.47338e+006 ; # 694
697   PLANT.LOOP1.PSI(70) := 1.46791e+006 ; # 695
698   PLANT.LOOP1.PSI(71) := 1.46218e+006 ; # 696
699   PLANT.LOOP1.PSI(72) := 1.45616e+006 ; # 697
700   PLANT.LOOP1.PSI(73) := 1.44982e+006 ; # 698
701   PLANT.LOOP1.PSI(74) := 1.44315e+006 ; # 699
702   PLANT.LOOP1.PSI(75) := 1.43612e+006 ; # 700
703   PLANT.LOOP1.PSI(76) := 1.4287e+006 ; # 701
704   PLANT.LOOP1.PSI(77) := 1.42087e+006 ; # 702
705   PLANT.LOOP1.PSI(78) := 1.41258e+006 ; # 703
706   PLANT.LOOP1.PSI(79) := 1.40379e+006 ; # 704
707   PLANT.LOOP1.PSI(80) := 1.39448e+006 ; # 705
708   PLANT.LOOP1.PSI(81) := 1.38459e+006 ; # 706
709   PLANT.LOOP1.PSI(82) := 1.37407e+006 ; # 707
710   PLANT.LOOP1.PSI(83) := 1.36287e+006 ; # 708
711   PLANT.LOOP1.PSI(84) := 1.35093e+006 ; # 709
712   PLANT.LOOP1.PSI(85) := 1.33817e+006 ; # 710
713   PLANT.LOOP1.PSI(86) := 1.32451e+006 ; # 711
714   PLANT.LOOP1.PSI(87) := 1.30988e+006 ; # 712
715   PLANT.LOOP1.PSI(88) := 1.29416e+006 ; # 713
716   PLANT.LOOP1.PSI(89) := 1.27725e+006 ; # 714
717   PLANT.LOOP1.PSI(90) := 1.25903e+006 ; # 715
718   PLANT.LOOP1.PSI(91) := 1.23934e+006 ; # 716
719   PLANT.LOOP1.PSI(92) := 1.21801e+006 ; # 717
720   PLANT.LOOP1.PSI(93) := 1.19486e+006 ; # 718
721   PLANT.LOOP1.PSI(94) := 1.16967e+006 ; # 719
722   PLANT.LOOP1.PSI(95) := 1.14217e+006 ; # 720
723   PLANT.LOOP1.PSI(96) := 1.11205e+006 ; # 721
724   PLANT.LOOP1.PSI(97) := 1.07896e+006 ; # 722
725   PLANT.LOOP1.PSI(98) := 1.04248e+006 ; # 723
726   PLANT.LOOP1.PSI(99) := 1.00208e+006 ; # 724
727   PLANT.LOOP1.PSI(100) := 957156 ; # 725
728   PLANT.LOOP1.F(1) := 26.308 ; # 726
729   PLANT.LOOP1.F(2) := 26.308 ; # 727
730   PLANT.LOOP1.F(3) := 26.308 ; # 728
731   PLANT.LOOP1.F(4) := 26.308 ; # 729
732   PLANT.LOOP1.F(5) := 26.308 ; # 730
733   PLANT.LOOP1.F(6) := 26.308 ; # 731
734   PLANT.LOOP1.F(7) := 26.308 ; # 732
735   PLANT.LOOP1.F(8) := 26.308 ; # 733
736   PLANT.LOOP1.F(9) := 26.308 ; # 734
737   PLANT.LOOP1.F(10) := 26.308 ; # 735
738   PLANT.LOOP1.F(11) := 26.308 ; # 736
739   PLANT.LOOP1.F(12) := 26.308 ; # 737
740   PLANT.LOOP1.F(13) := 26.308 ; # 738
741   PLANT.LOOP1.F(14) := 26.308 ; # 739
742   PLANT.LOOP1.F(15) := 26.308 ; # 740
743   PLANT.LOOP1.F(16) := 26.308 ; # 741
744   PLANT.LOOP1.F(17) := 26.308 ; # 742
745   PLANT.LOOP1.F(18) := 26.308 ; # 743
746   PLANT.LOOP1.F(19) := 26.308 ; # 744
747   PLANT.LOOP1.F(20) := 26.308 ; # 745
748   PLANT.LOOP1.F(21) := 26.308 ; # 746
749   PLANT.LOOP1.F(22) := 26.308 ; # 747
750   PLANT.LOOP1.F(23) := 26.308 ; # 748
751   PLANT.LOOP1.F(24) := 26.308 ; # 749
752   PLANT.LOOP1.F(25) := 26.308 ; # 750
753   PLANT.LOOP1.F(26) := 26.308 ; # 751
754   PLANT.LOOP1.F(27) := 26.308 ; # 752
755   PLANT.LOOP1.F(28) := 26.308 ; # 753
756   PLANT.LOOP1.F(29) := 26.308 ; # 754
757   PLANT.LOOP1.F(30) := 26.308 ; # 755
758   PLANT.LOOP1.F(31) := 26.308 ; # 756
759   PLANT.LOOP1.F(32) := 26.308 ; # 757
760   PLANT.LOOP1.F(33) := 26.308 ; # 758
761   PLANT.LOOP1.F(34) := 26.308 ; # 759
762   PLANT.LOOP1.F(35) := 26.308 ; # 760
763   PLANT.LOOP1.F(36) := 26.308 ; # 761
764   PLANT.LOOP1.F(37) := 26.308 ; # 762
765   PLANT.LOOP1.F(38) := 26.308 ; # 763
766   PLANT.LOOP1.F(39) := 26.308 ; # 764
767   PLANT.LOOP1.F(40) := 26.308 ; # 765
768   PLANT.LOOP1.F(41) := 26.308 ; # 766
769   PLANT.LOOP1.F(42) := 26.308 ; # 767
770   PLANT.LOOP1.F(43) := 26.308 ; # 768
771   PLANT.LOOP1.F(44) := 26.308 ; # 769
772   PLANT.LOOP1.F(45) := 26.308 ; # 770
```

```
773   PLANT.LOOP1.F(46) := 26.308 ; # 771
774   PLANT.LOOP1.F(47) := 26.308 ; # 772
775   PLANT.LOOP1.F(48) := 26.308 ; # 773
776   PLANT.LOOP1.F(49) := 26.308 ; # 774
777   PLANT.LOOP1.F(50) := 26.308 ; # 775
778   PLANT.LOOP1.F(51) := 26.308 ; # 776
779   PLANT.LOOP1.F(52) := 26.308 ; # 777
780   PLANT.LOOP1.F(53) := 26.308 ; # 778
781   PLANT.LOOP1.F(54) := 26.308 ; # 779
782   PLANT.LOOP1.F(55) := 26.308 ; # 780
783   PLANT.LOOP1.F(56) := 26.308 ; # 781
784   PLANT.LOOP1.F(57) := 26.308 ; # 782
785   PLANT.LOOP1.F(58) := 26.308 ; # 783
786   PLANT.LOOP1.F(59) := 26.308 ; # 784
787   PLANT.LOOP1.F(60) := 26.308 ; # 785
788   PLANT.LOOP1.F(61) := 26.308 ; # 786
789   PLANT.LOOP1.F(62) := 26.308 ; # 787
790   PLANT.LOOP1.F(63) := 26.308 ; # 788
791   PLANT.LOOP1.F(64) := 26.308 ; # 789
792   PLANT.LOOP1.F(65) := 26.308 ; # 790
793   PLANT.LOOP1.F(66) := 26.308 ; # 791
794   PLANT.LOOP1.F(67) := 26.308 ; # 792
795   PLANT.LOOP1.F(68) := 26.308 ; # 793
796   PLANT.LOOP1.F(69) := 26.308 ; # 794
797   PLANT.LOOP1.F(70) := 26.308 ; # 795
798   PLANT.LOOP1.F(71) := 26.308 ; # 796
799   PLANT.LOOP1.F(72) := 26.308 ; # 797
800   PLANT.LOOP1.F(73) := 26.308 ; # 798
801   PLANT.LOOP1.F(74) := 26.308 ; # 799
802   PLANT.LOOP1.F(75) := 26.308 ; # 800
803   PLANT.LOOP1.F(76) := 26.308 ; # 801
804   PLANT.LOOP1.F(77) := 26.308 ; # 802
805   PLANT.LOOP1.F(78) := 26.308 ; # 803
806   PLANT.LOOP1.F(79) := 26.308 ; # 804
807   PLANT.LOOP1.F(80) := 26.308 ; # 805
808   PLANT.LOOP1.F(81) := 26.308 ; # 806
809   PLANT.LOOP1.F(82) := 26.308 ; # 807
810   PLANT.LOOP1.F(83) := 26.308 ; # 808
811   PLANT.LOOP1.F(84) := 26.308 ; # 809
812   PLANT.LOOP1.F(85) := 26.308 ; # 810
813   PLANT.LOOP1.F(86) := 26.308 ; # 811
814   PLANT.LOOP1.F(87) := 26.308 ; # 812
815   PLANT.LOOP1.F(88) := 26.308 ; # 813
816   PLANT.LOOP1.F(89) := 26.308 ; # 814
817   PLANT.LOOP1.F(90) := 26.308 ; # 815
818   PLANT.LOOP1.F(91) := 26.308 ; # 816
819   PLANT.LOOP1.F(92) := 26.308 ; # 817
820   PLANT.LOOP1.F(93) := 26.308 ; # 818
821   PLANT.LOOP1.F(94) := 26.308 ; # 819
822   PLANT.LOOP1.F(95) := 26.308 ; # 820
823   PLANT.LOOP1.F(96) := 26.308 ; # 821
824   PLANT.LOOP1.F(97) := 26.308 ; # 822
825   PLANT.LOOP1.F(98) := 26.308 ; # 823
826   PLANT.LOOP1.F(99) := 26.308 ; # 824
827   PLANT.LOOP1.F(100) := 26.308 ; # 825
828   PLANT.LOOP1.F_IN := 26.308 ; # 826
829   PLANT.LOOP1.RHO_AVG := 0.91979 ; # 827
830   PLANT.LOOP1.TOTALMASS := 27.4242 ; # 828
831   PLANT.LOOP1.RHOEXTRA(1) := 1.1654 ; # 829
832   PLANT.LOOP1.RHOEXTRA(2) := 1.1654 ; # 830
833   PLANT.LOOP1.PEXTRA(1) := 2.03415e+006 ; # 831
834   PLANT.LOOP1.PEXTRA(2) := 2.03415e+006 ; # 832
835   PLANT.LOOP1.HEXTRA(1) := 2.81449e+006 ; # 833
836   PLANT.LOOP1.HEXTRA(2) := 2.81449e+006 ; # 834
837   PLANT.LOOP1.TEXTRA(1) := 839.77 ; # 835
838   PLANT.LOOP1.TEXTRA(2) := 839.77 ; # 836
839   PLANT.LOOP1.PSIEXTRA(1) := 1.24585e+006 ; # 837
840   PLANT.LOOP1.PSIEXTRA(2) := 1.24585e+006 ; # 838
841   PLANT.LOOP1.FEXTRA(1) := 26.308 ; # 839
842   PLANT.LOOP1.FEXTRA(2) := 26.308 ; # 840
843   PLANT.LOOP1.HT1.H2 := 17.344 ; # 841
844   PLANT.LOOP1.HT1.RHOPIPEHE := 1.14182 ; # 842
845   PLANT.LOOP1.HT1.VPIPE := 171.996 ; # 843
846   PLANT.LOOP1.HT1.H0 := 1500.09 ; # 844
847   PLANT.LOOP1.HT1.U := 8.93421 ; # 845
848   PLANT.LOOP1.HT2.RHOPIPEHE := 0.902801 ; # 846
849   PLANT.LOOP1.HT2.VPIPE := 84.86 ; # 847
850   PLANT.LOOP1.HT2.H0 := 1665.62 ; # 848
851   PLANT.LOOP1.HT2.U := 1645.45 ; # 849
852   PLANT.LOOP1.HT3.H2 := 17.344 ; # 850
853   PLANT.LOOP1.HT3.RHOPIPEHE := 0.74956 ; # 851
854   PLANT.LOOP1.HT3.VPIPE := 226.698 ; # 852
855   PLANT.LOOP1.HT3.H0 := 1316.87 ; # 853
856   PLANT.LOOP1.HT3.U := 10.3572 ; # 854
857   PLANT.LOOP1.HT4.RHOPIPEHE := 0.833125 ; # 855
858   PLANT.LOOP1.HT4.VPIPE := 79.4405 ; # 856
859   PLANT.LOOP1.HT4.U := 1299.78 ; # 857
860   PLANT.PIPEH.RHO(1) := 2.87869 ; # 858
861   PLANT.PIPEH.RHO(2) := 2.88664 ; # 859
862   PLANT.PIPEH.RHO(3) := 2.89491 ; # 860
863   PLANT.PIPEH.RHO(4) := 2.90353 ; # 861
864   PLANT.PIPEH.RHO(5) := 2.9125 ; # 862
```

```
PLANT.PIPEH.RHO(6) := 2.92184 ; # 863
PLANT.PIPEH.RHO(7) := 2.93156 ; # 864
PLANT.PIPEH.RHO(8) := 2.9417 ; # 865
PLANT.PIPEH.RHO(9) := 2.95225 ; # 866
PLANT.PIPEH.RHO(10) := 2.96325 ; # 867
PLANT.PIPEH.RHO(11) := 2.97471 ; # 868
PLANT.PIPEH.RHO(12) := 2.98666 ; # 869
PLANT.PIPEH.RHO(13) := 2.99912 ; # 870
PLANT.PIPEH.RHO(14) := 3.01211 ; # 871
PLANT.PIPEH.RHO(15) := 3.02567 ; # 872
PLANT.PIPEH.RHO(16) := 3.03982 ; # 873
PLANT.PIPEH.RHO(17) := 3.05458 ; # 874
PLANT.PIPEH.RHO(18) := 3.07 ; # 875
PLANT.PIPEH.RHO(19) := 3.08611 ; # 876
PLANT.PIPEH.RHO(20) := 3.10294 ; # 877
PLANT.PIPEH.RHO(21) := 3.12053 ; # 878
PLANT.PIPEH.RHO(22) := 3.13892 ; # 879
PLANT.PIPEH.RHO(23) := 3.15816 ; # 880
PLANT.PIPEH.RHO(24) := 3.1783 ; # 881
PLANT.PIPEH.RHO(25) := 3.19939 ; # 882
PLANT.PIPEH.RHO(26) := 3.22148 ; # 883
PLANT.PIPEH.RHO(27) := 3.24463 ; # 884
PLANT.PIPEH.RHO(28) := 3.2689 ; # 885
PLANT.PIPEH.RHO(29) := 3.29437 ; # 886
PLANT.PIPEH.RHO(30) := 3.32112 ; # 887
PLANT.PIPEH.RHO(31) := 3.34921 ; # 888
PLANT.PIPEH.RHO(32) := 3.37874 ; # 889
PLANT.PIPEH.RHO(33) := 3.4098 ; # 890
PLANT.PIPEH.RHO(34) := 3.4425 ; # 891
PLANT.PIPEH.RHO(35) := 3.47695 ; # 892
PLANT.PIPEH.RHO(36) := 3.51328 ; # 893
PLANT.PIPEH.RHO(37) := 3.55161 ; # 894
PLANT.PIPEH.RHO(38) := 3.59209 ; # 895
PLANT.PIPEH.RHO(39) := 3.6349 ; # 896
PLANT.PIPEH.RHO(40) := 3.68012 ; # 897
PLANT.PIPEH.P(1) := 6.99931e+006 ; # 898
PLANT.PIPEH.P(2) := 6.99793e+006 ; # 899
PLANT.PIPEH.P(3) := 6.99656e+006 ; # 900
PLANT.PIPEH.P(4) := 6.99519e+006 ; # 901
PLANT.PIPEH.P(5) := 6.99383e+006 ; # 902
PLANT.PIPEH.P(6) := 6.99247e+006 ; # 903
PLANT.PIPEH.P(7) := 6.99111e+006 ; # 904
PLANT.PIPEH.P(8) := 6.98977e+006 ; # 905
PLANT.PIPEH.P(9) := 6.98842e+006 ; # 906
PLANT.PIPEH.P(10) := 6.98708e+006 ; # 907
PLANT.PIPEH.P(11) := 6.98575e+006 ; # 908
PLANT.PIPEH.P(12) := 6.98442e+006 ; # 909
PLANT.PIPEH.P(13) := 6.9831e+006 ; # 910
PLANT.PIPEH.P(14) := 6.98179e+006 ; # 911
PLANT.PIPEH.P(15) := 6.98048e+006 ; # 912
PLANT.PIPEH.P(16) := 6.97917e+006 ; # 913
PLANT.PIPEH.P(17) := 6.97788e+006 ; # 914
PLANT.PIPEH.P(18) := 6.97659e+006 ; # 915
PLANT.PIPEH.P(19) := 6.97531e+006 ; # 916
PLANT.PIPEH.P(20) := 6.97403e+006 ; # 917
PLANT.PIPEH.P(21) := 6.97276e+006 ; # 918
PLANT.PIPEH.P(22) := 6.97151e+006 ; # 919
PLANT.PIPEH.P(23) := 6.97025e+006 ; # 920
PLANT.PIPEH.P(24) := 6.96901e+006 ; # 921
PLANT.PIPEH.P(25) := 6.96778e+006 ; # 922
PLANT.PIPEH.P(26) := 6.96655e+006 ; # 923
PLANT.PIPEH.P(27) := 6.96534e+006 ; # 924
PLANT.PIPEH.P(28) := 6.96413e+006 ; # 925
PLANT.PIPEH.P(29) := 6.96294e+006 ; # 926
PLANT.PIPEH.P(30) := 6.96175e+006 ; # 927
PLANT.PIPEH.P(31) := 6.96058e+006 ; # 928
PLANT.PIPEH.P(32) := 6.95941e+006 ; # 929
PLANT.PIPEH.P(33) := 6.95826e+006 ; # 930
PLANT.PIPEH.P(34) := 6.95712e+006 ; # 931
PLANT.PIPEH.P(35) := 6.95599e+006 ; # 932
PLANT.PIPEH.P(36) := 6.95487e+006 ; # 933
PLANT.PIPEH.P(37) := 6.95377e+006 ; # 934
PLANT.PIPEH.P(38) := 6.95268e+006 ; # 935
PLANT.PIPEH.P(39) := 6.9516e+006 ; # 936
PLANT.PIPEH.P(40) := 6.95054e+006 ; # 937
PLANT.PIPEH.H(1) := 4.52899e+006 ; # 938
PLANT.PIPEH.H(2) := 4.51106e+006 ; # 939
PLANT.PIPEH.H(3) := 4.49255e+006 ; # 940
PLANT.PIPEH.H(4) := 4.47345e+006 ; # 941
PLANT.PIPEH.H(5) := 4.45373e+006 ; # 942
PLANT.PIPEH.H(6) := 4.43339e+006 ; # 943
PLANT.PIPEH.H(7) := 4.41239e+006 ; # 944
PLANT.PIPEH.H(8) := 4.39071e+006 ; # 945
PLANT.PIPEH.H(9) := 4.36834e+006 ; # 946
PLANT.PIPEH.H(10) := 4.34525e+006 ; # 947
PLANT.PIPEH.H(11) := 4.32143e+006 ; # 948
PLANT.PIPEH.H(12) := 4.29683e+006 ; # 949
PLANT.PIPEH.H(13) := 4.27145e+006 ; # 950
PLANT.PIPEH.H(14) := 4.24525e+006 ; # 951
PLANT.PIPEH.H(15) := 4.21821e+006 ; # 952
PLANT.PIPEH.H(16) := 4.19031e+006 ; # 953
PLANT.PIPEH.H(17) := 4.16151e+006 ; # 954
```

```
PLANT.PIPEH.H(18) := 4.13178e+006 ; # 955
PLANT.PIPEH.H(19) := 4.1011e+006 ; # 956
PLANT.PIPEH.H(20) := 4.06944e+006 ; # 957
PLANT.PIPEH.H(21) := 4.03675e+006 ; # 958
PLANT.PIPEH.H(22) := 4.00302e+006 ; # 959
PLANT.PIPEH.H(23) := 3.96821e+006 ; # 960
PLANT.PIPEH.H(24) := 3.93228e+006 ; # 961
PLANT.PIPEH.H(25) := 3.8952e+006 ; # 962
PLANT.PIPEH.H(26) := 3.85693e+006 ; # 963
PLANT.PIPEH.H(27) := 3.81742e+006 ; # 964
PLANT.PIPEH.H(28) := 3.77666e+006 ; # 965
PLANT.PIPEH.H(29) := 3.73458e+006 ; # 966
PLANT.PIPEH.H(30) := 3.69115e+006 ; # 967
PLANT.PIPEH.H(31) := 3.64633e+006 ; # 968
PLANT.PIPEH.H(32) := 3.60007e+006 ; # 969
PLANT.PIPEH.H(33) := 3.55232e+006 ; # 970
PLANT.PIPEH.H(34) := 3.50304e+006 ; # 971
PLANT.PIPEH.H(35) := 3.45219e+006 ; # 972
PLANT.PIPEH.H(36) := 3.39969e+006 ; # 973
PLANT.PIPEH.H(37) := 3.34552e+006 ; # 974
PLANT.PIPEH.H(38) := 3.2896e+006 ; # 975
PLANT.PIPEH.H(39) := 3.23189e+006 ; # 976
PLANT.PIPEH.H(40) := 3.17244e+006 ; # 977
PLANT.PIPEH.H0 := 4.5464e+006 ; # 978
PLANT.PIPEH.T(1) := 1169.8 ; # 979
PLANT.PIPEH.T(2) := 1166.35 ; # 980
PLANT.PIPEH.T(3) := 1162.78 ; # 981
PLANT.PIPEH.T(4) := 1159.11 ; # 982
PLANT.PIPEH.T(5) := 1155.31 ; # 983
PLANT.PIPEH.T(6) := 1151.39 ; # 984
PLANT.PIPEH.T(7) := 1147.35 ; # 985
PLANT.PIPEH.T(8) := 1143.18 ; # 986
PLANT.PIPEH.T(9) := 1138.87 ; # 987
PLANT.PIPEH.T(10) := 1134.43 ; # 988
PLANT.PIPEH.T(11) := 1129.84 ; # 989
PLANT.PIPEH.T(12) := 1125.11 ; # 990
PLANT.PIPEH.T(13) := 1120.22 ; # 991
PLANT.PIPEH.T(14) := 1115.18 ; # 992
PLANT.PIPEH.T(15) := 1109.98 ; # 993
PLANT.PIPEH.T(16) := 1104.6 ; # 994
PLANT.PIPEH.T(17) := 1099.06 ; # 995
PLANT.PIPEH.T(18) := 1093.34 ; # 996
PLANT.PIPEH.T(19) := 1087.43 ; # 997
PLANT.PIPEH.T(20) := 1081.34 ; # 998
PLANT.PIPEH.T(21) := 1075.05 ; # 999
PLANT.PIPEH.T(22) := 1068.55 ; # 1000
PLANT.PIPEH.T(23) := 1061.85 ; # 1001
PLANT.PIPEH.T(24) := 1054.94 ; # 1002
PLANT.PIPEH.T(25) := 1047.8 ; # 1003
PLANT.PIPEH.T(26) := 1040.43 ; # 1004
PLANT.PIPEH.T(27) := 1032.83 ; # 1005
PLANT.PIPEH.T(28) := 1024.98 ; # 1006
PLANT.PIPEH.T(29) := 1016.88 ; # 1007
PLANT.PIPEH.T(30) := 1008.52 ; # 1008
PLANT.PIPEH.T(31) := 999.892 ; # 1009
PLANT.PIPEH.T(32) := 990.987 ; # 1010
PLANT.PIPEH.T(33) := 981.796 ; # 1011
PLANT.PIPEH.T(34) := 972.311 ; # 1012
PLANT.PIPEH.T(35) := 962.521 ; # 1013
PLANT.PIPEH.T(36) := 952.416 ; # 1014
PLANT.PIPEH.T(37) := 941.988 ; # 1015
PLANT.PIPEH.T(38) := 931.225 ; # 1016
PLANT.PIPEH.T(39) := 920.116 ; # 1017
PLANT.PIPEH.T(40) := 908.671 ; # 1018
PLANT.PIPEH.TEXTRNL(1) := 1161.37 ; # 1019
PLANT.PIPEH.TEXTRNL(2) := 1157.67 ; # 1020
PLANT.PIPEH.TEXTRNL(3) := 1153.83 ; # 1021
PLANT.PIPEH.TEXTRNL(4) := 1149.86 ; # 1022
PLANT.PIPEH.TEXTRNL(5) := 1145.77 ; # 1023
PLANT.PIPEH.TEXTRNL(6) := 1141.55 ; # 1024
PLANT.PIPEH.TEXTRNL(7) := 1137.19 ; # 1025
PLANT.PIPEH.TEXTRNL(8) := 1132.69 ; # 1026
PLANT.PIPEH.TEXTRNL(9) := 1128.05 ; # 1027
PLANT.PIPEH.TEXTRNL(10) := 1123.26 ; # 1028
PLANT.PIPEH.TEXTRNL(11) := 1118.31 ; # 1029
PLANT.PIPEH.TEXTRNL(12) := 1113.21 ; # 1030
PLANT.PIPEH.TEXTRNL(13) := 1107.94 ; # 1031
PLANT.PIPEH.TEXTRNL(14) := 1102.5 ; # 1032
PLANT.PIPEH.TEXTRNL(15) := 1096.89 ; # 1033
PLANT.PIPEH.TEXTRNL(16) := 1091.1 ; # 1034
PLANT.PIPEH.TEXTRNL(17) := 1085.12 ; # 1035
PLANT.PIPEH.TEXTRNL(18) := 1078.95 ; # 1036
PLANT.PIPEH.TEXTRNL(19) := 1072.58 ; # 1037
PLANT.PIPEH.TEXTRNL(20) := 1066.01 ; # 1038
PLANT.PIPEH.TEXTRNL(21) := 1059.23 ; # 1039
PLANT.PIPEH.TEXTRNL(22) := 1052.23 ; # 1040
PLANT.PIPEH.TEXTRNL(23) := 1045 ; # 1041
PLANT.PIPEH.TEXTRNL(24) := 1037.55 ; # 1042
PLANT.PIPEH.TEXTRNL(25) := 1029.85 ; # 1043
PLANT.PIPEH.TEXTRNL(26) := 1021.91 ; # 1044
PLANT.PIPEH.TEXTRNL(27) := 1013.71 ; # 1045
PLANT.PIPEH.TEXTRNL(28) := 1005.25 ; # 1046
```

```
1049   PLANT.PIPEH.TEXTRNL(29) := 996.513 ; # 1047
1050   PLANT.PIPEH.TEXTRNL(30) := 987.5 ; # 1048
1051   PLANT.PIPEH.TEXTRNL(31) := 978.197 ; # 1049
1052   PLANT.PIPEH.TEXTRNL(32) := 968.596 ; # 1050
1053   PLANT.PIPEH.TEXTRNL(33) := 958.687 ; # 1051
1054   PLANT.PIPEH.TEXTRNL(34) := 948.459 ; # 1052
1055   PLANT.PIPEH.TEXTRNL(35) := 937.904 ; # 1053
1056   PLANT.PIPEH.TEXTRNL(36) := 927.009 ; # 1054
1057   PLANT.PIPEH.TEXTRNL(37) := 915.766 ; # 1055
1058   PLANT.PIPEH.TEXTRNL(38) := 904.161 ; # 1056
1059   PLANT.PIPEH.TEXTRNL(39) := 892.184 ; # 1057
1060   PLANT.PIPEH.TEXTRNL(40) := 879.893 ; # 1058
1061   PLANT.PIPEH.F(1) := 32.3305 ; # 1059
1062   PLANT.PIPEH.F(2) := 32.3305 ; # 1060
1063   PLANT.PIPEH.F(3) := 32.3305 ; # 1061
1064   PLANT.PIPEH.F(4) := 32.3305 ; # 1062
1065   PLANT.PIPEH.F(5) := 32.3305 ; # 1063
1066   PLANT.PIPEH.F(6) := 32.3305 ; # 1064
1067   PLANT.PIPEH.F(7) := 32.3305 ; # 1065
1068   PLANT.PIPEH.F(8) := 32.3305 ; # 1066
1069   PLANT.PIPEH.F(9) := 32.3305 ; # 1067
1070   PLANT.PIPEH.F(10) := 32.3305 ; # 1068
1071   PLANT.PIPEH.F(11) := 32.3305 ; # 1069
1072   PLANT.PIPEH.F(12) := 32.3305 ; # 1070
1073   PLANT.PIPEH.F(13) := 32.3305 ; # 1071
1074   PLANT.PIPEH.F(14) := 32.3305 ; # 1072
1075   PLANT.PIPEH.F(15) := 32.3305 ; # 1073
1076   PLANT.PIPEH.F(16) := 32.3305 ; # 1074
1077   PLANT.PIPEH.F(17) := 32.3305 ; # 1075
1078   PLANT.PIPEH.F(18) := 32.3305 ; # 1076
1079   PLANT.PIPEH.F(19) := 32.3305 ; # 1077
1080   PLANT.PIPEH.F(20) := 32.3305 ; # 1078
1081   PLANT.PIPEH.F(21) := 32.3305 ; # 1079
1082   PLANT.PIPEH.F(22) := 32.3305 ; # 1080
1083   PLANT.PIPEH.F(23) := 32.3305 ; # 1081
1084   PLANT.PIPEH.F(24) := 32.3305 ; # 1082
1085   PLANT.PIPEH.F(25) := 32.3305 ; # 1083
1086   PLANT.PIPEH.F(26) := 32.3305 ; # 1084
1087   PLANT.PIPEH.F(27) := 32.3305 ; # 1085
1088   PLANT.PIPEH.F(28) := 32.3305 ; # 1086
1089   PLANT.PIPEH.F(29) := 32.3305 ; # 1087
1090   PLANT.PIPEH.F(30) := 32.3305 ; # 1088
1091   PLANT.PIPEH.F(31) := 32.3305 ; # 1089
1092   PLANT.PIPEH.F(32) := 32.3305 ; # 1090
1093   PLANT.PIPEH.F(33) := 32.3305 ; # 1091
1094   PLANT.PIPEH.F(34) := 32.3305 ; # 1092
1095   PLANT.PIPEH.F(35) := 32.3305 ; # 1093
1096   PLANT.PIPEH.F(36) := 32.3305 ; # 1094
1097   PLANT.PIPEH.F(37) := 32.3305 ; # 1095
1098   PLANT.PIPEH.F(38) := 32.3305 ; # 1096
1099   PLANT.PIPEH.F(39) := 32.3305 ; # 1097
1100   PLANT.PIPEH.F(40) := 32.3305 ; # 1098
1101   PLANT.PIPEH.F_IN := 32.3305 ; # 1099
1102   PLANT.PIPEH.PSI(1) := 6.03823e+006 ; # 1100
1103   PLANT.PIPEH.PSI(2) := 6.02385e+006 ; # 1101
1104   PLANT.PIPEH.PSI(3) := 6.00898e+006 ; # 1102
1105   PLANT.PIPEH.PSI(4) := 5.9936e+006 ; # 1103
1106   PLANT.PIPEH.PSI(5) := 5.97767e+006 ; # 1104
1107   PLANT.PIPEH.PSI(6) := 5.96117e+006 ; # 1105
1108   PLANT.PIPEH.PSI(7) := 5.94408e+006 ; # 1106
1109   PLANT.PIPEH.PSI(8) := 5.92638e+006 ; # 1107
1110   PLANT.PIPEH.PSI(9) := 5.90802e+006 ; # 1108
1111   PLANT.PIPEH.PSI(10) := 5.88899e+006 ; # 1109
1112   PLANT.PIPEH.PSI(11) := 5.86924e+006 ; # 1110
1113   PLANT.PIPEH.PSI(12) := 5.84876e+006 ; # 1111
1114   PLANT.PIPEH.PSI(13) := 5.82749e+006 ; # 1112
1115   PLANT.PIPEH.PSI(14) := 5.8054e+006 ; # 1113
1116   PLANT.PIPEH.PSI(15) := 5.78245e+006 ; # 1114
1117   PLANT.PIPEH.PSI(16) := 5.75859e+006 ; # 1115
1118   PLANT.PIPEH.PSI(17) := 5.73379e+006 ; # 1116
1119   PLANT.PIPEH.PSI(18) := 5.70799e+006 ; # 1117
1120   PLANT.PIPEH.PSI(19) := 5.68113e+006 ; # 1118
1121   PLANT.PIPEH.PSI(20) := 5.65317e+006 ; # 1119
1122   PLANT.PIPEH.PSI(21) := 5.62404e+006 ; # 1120
1123   PLANT.PIPEH.PSI(22) := 5.59367e+006 ; # 1121
1124   PLANT.PIPEH.PSI(23) := 5.56201e+006 ; # 1122
1125   PLANT.PIPEH.PSI(24) := 5.52897e+006 ; # 1123
1126   PLANT.PIPEH.PSI(25) := 5.49447e+006 ; # 1124
1127   PLANT.PIPEH.PSI(26) := 5.45844e+006 ; # 1125
1128   PLANT.PIPEH.PSI(27) := 5.42078e+006 ; # 1126
1129   PLANT.PIPEH.PSI(28) := 5.38139e+006 ; # 1127
1130   PLANT.PIPEH.PSI(29) := 5.34016e+006 ; # 1128
1131   PLANT.PIPEH.PSI(30) := 5.29699e+006 ; # 1129
1132   PLANT.PIPEH.PSI(31) := 5.25174e+006 ; # 1130
1133   PLANT.PIPEH.PSI(32) := 5.20427e+006 ; # 1131
1134   PLANT.PIPEH.PSI(33) := 5.15446e+006 ; # 1132
1135   PLANT.PIPEH.PSI(34) := 5.10212e+006 ; # 1133
1136   PLANT.PIPEH.PSI(35) := 5.0471e+006 ; # 1134
1137   PLANT.PIPEH.PSI(36) := 4.98919e+006 ; # 1135
1138   PLANT.PIPEH.PSI(37) := 4.92819e+006 ; # 1136
1139   PLANT.PIPEH.PSI(38) := 4.86388e+006 ; # 1137
1140   PLANT.PIPEH.PSI(39) := 4.796e+006 ; # 1138
```

```
1141   PLANT.PIPEH.PSI(40) := 4.7244e+006 ; # 1139
1142   PLANT.PIPEH.HT.RHOPIPEHE := 3.09587 ; # 1140
1143   PLANT.PIPEH.HT.VPIPE := 28.7109 ; # 1141
1144   PLANT.PIPEH.HT.H0 := 1760.94 ; # 1142
1145   PLANT.PIPEH.HT.U := 1738.41 ; # 1143
1146   PLANT.HX2.TGASHOT(1) := 1141.93 ; # 1144
1147   PLANT.HX2.TGASHOT(2) := 1139.37 ; # 1145
1148   PLANT.HX2.TGASHOT(3) := 1136.66 ; # 1146
1149   PLANT.HX2.TGASHOT(4) := 1133.8 ; # 1147
1150   PLANT.HX2.TGASHOT(5) := 1130.77 ; # 1148
1151   PLANT.HX2.TGASHOT(6) := 1127.56 ; # 1149
1152   PLANT.HX2.TGASHOT(7) := 1124.17 ; # 1150
1153   PLANT.HX2.TGASHOT(8) := 1120.57 ; # 1151
1154   PLANT.HX2.TGASHOT(9) := 1116.77 ; # 1152
1155   PLANT.HX2.TGASHOT(10) := 1112.75 ; # 1153
1156   PLANT.HX2.TGASHOT(11) := 1108.5 ; # 1154
1157   PLANT.HX2.TGASHOT(12) := 1104 ; # 1155
1158   PLANT.HX2.TGASHOT(13) := 1099.23 ; # 1156
1159   PLANT.HX2.TGASHOT(14) := 1094.19 ; # 1157
1160   PLANT.HX2.TGASHOT(15) := 1088.86 ; # 1158
1161   PLANT.HX2.TGASHOT(16) := 1083.21 ; # 1159
1162   PLANT.HX2.TGASHOT(17) := 1077.24 ; # 1160
1163   PLANT.HX2.TGASHOT(18) := 1070.92 ; # 1161
1164   PLANT.HX2.TGASHOT(19) := 1064.23 ; # 1162
1165   PLANT.HX2.TGASHOT(20) := 1057.15 ; # 1163
1166   PLANT.HX2.TGASHOT(21) := 1049.66 ; # 1164
1167   PLANT.HX2.TGASHOT(22) := 1041.74 ; # 1165
1168   PLANT.HX2.TGASHOT(23) := 1033.36 ; # 1166
1169   PLANT.HX2.TGASHOT(24) := 1024.48 ; # 1167
1170   PLANT.HX2.TGASHOT(25) := 1015.09 ; # 1168
1171   PLANT.HX2.TGASHOT(26) := 1005.16 ; # 1169
1172   PLANT.HX2.TGASHOT(27) := 994.647 ; # 1170
1173   PLANT.HX2.TGASHOT(28) := 983.524 ; # 1171
1174   PLANT.HX2.TGASHOT(29) := 971.753 ; # 1172
1175   PLANT.HX2.TGASHOT(30) := 959.298 ; # 1173
1176   PLANT.HX2.TGASHOT(31) := 946.119 ; # 1174
1177   PLANT.HX2.TGASHOT(32) := 932.173 ; # 1175
1178   PLANT.HX2.TGASHOT(33) := 917.416 ; # 1176
1179   PLANT.HX2.TGASHOT(34) := 901.801 ; # 1177
1180   PLANT.HX2.TGASHOT(35) := 885.278 ; # 1178
1181   PLANT.HX2.TGASHOT(36) := 867.793 ; # 1179
1182   PLANT.HX2.TGASHOT(37) := 849.292 ; # 1180
1183   PLANT.HX2.TGASHOT(38) := 829.716 ; # 1181
1184   PLANT.HX2.TGASHOT(39) := 809 ; # 1182
1185   PLANT.HX2.TGASHOT(40) := 787.082 ; # 1183
1186   PLANT.HX2.TGASCOLD(1) := 1126.32 ; # 1184
1187   PLANT.HX2.TGASCOLD(2) := 1122.86 ; # 1185
1188   PLANT.HX2.TGASCOLD(3) := 1119.19 ; # 1186
1189   PLANT.HX2.TGASCOLD(4) := 1115.31 ; # 1187
1190   PLANT.HX2.TGASCOLD(5) := 1111.2 ; # 1188
1191   PLANT.HX2.TGASCOLD(6) := 1106.86 ; # 1189
1192   PLANT.HX2.TGASCOLD(7) := 1102.26 ; # 1190
1193   PLANT.HX2.TGASCOLD(8) := 1097.39 ; # 1191
1194   PLANT.HX2.TGASCOLD(9) := 1092.24 ; # 1192
1195   PLANT.HX2.TGASCOLD(10) := 1086.8 ; # 1193
1196   PLANT.HX2.TGASCOLD(11) := 1081.03 ; # 1194
1197   PLANT.HX2.TGASCOLD(12) := 1074.93 ; # 1195
1198   PLANT.HX2.TGASCOLD(13) := 1068.48 ; # 1196
1199   PLANT.HX2.TGASCOLD(14) := 1061.65 ; # 1197
1200   PLANT.HX2.TGASCOLD(15) := 1054.42 ; # 1198
1201   PLANT.HX2.TGASCOLD(16) := 1046.77 ; # 1199
1202   PLANT.HX2.TGASCOLD(17) := 1038.68 ; # 1200
1203   PLANT.HX2.TGASCOLD(18) := 1030.12 ; # 1201
1204   PLANT.HX2.TGASCOLD(19) := 1021.06 ; # 1202
1205   PLANT.HX2.TGASCOLD(20) := 1011.47 ; # 1203
1206   PLANT.HX2.TGASCOLD(21) := 1001.32 ; # 1204
1207   PLANT.HX2.TGASCOLD(22) := 990.587 ; # 1205
1208   PLANT.HX2.TGASCOLD(23) := 979.228 ; # 1206
1209   PLANT.HX2.TGASCOLD(24) := 967.207 ; # 1207
1210   PLANT.HX2.TGASCOLD(25) := 954.488 ; # 1208
1211   PLANT.HX2.TGASCOLD(26) := 941.029 ; # 1209
1212   PLANT.HX2.TGASCOLD(27) := 926.787 ; # 1210
1213   PLANT.HX2.TGASCOLD(28) := 911.717 ; # 1211
1214   PLANT.HX2.TGASCOLD(29) := 895.77 ; # 1212
1215   PLANT.HX2.TGASCOLD(30) := 878.896 ; # 1213
1216   PLANT.HX2.TGASCOLD(31) := 861.041 ; # 1214
1217   PLANT.HX2.TGASCOLD(32) := 842.147 ; # 1215
1218   PLANT.HX2.TGASCOLD(33) := 822.155 ; # 1216
1219   PLANT.HX2.TGASCOLD(34) := 801 ; # 1217
1220   PLANT.HX2.TGASCOLD(35) := 778.614 ; # 1218
1221   PLANT.HX2.TGASCOLD(36) := 754.927 ; # 1219
1222   PLANT.HX2.TGASCOLD(37) := 729.862 ; # 1220
1223   PLANT.HX2.TGASCOLD(38) := 703.34 ; # 1221
1224   PLANT.HX2.TGASCOLD(39) := 675.275 ; # 1222
1225   PLANT.HX2.TGASCOLD(40) := 645.577 ; # 1223
1226   PLANT.HX2.UHOT := 1299.78 ; # 1224
1227   PLANT.HX2.UCOLD := 1880.53 ; # 1225
1228   PLANT.HX2.THX(1) := 1133.71 ; # 1226
1229   PLANT.HX2.THX(2) := 1130.67 ; # 1227
1230   PLANT.HX2.THX(3) := 1127.46 ; # 1228
1231   PLANT.HX2.THX(4) := 1124.06 ; # 1229
1232   PLANT.HX2.THX(5) := 1120.46 ; # 1230
```

```
PLANT.HX2.THX(6) := 1116.65 ; # 1231
PLANT.HX2.THX(7) := 1112.63 ; # 1232
PLANT.HX2.THX(8) := 1108.36 ; # 1233
PLANT.HX2.THX(9) := 1103.85 ; # 1234
PLANT.HX2.THX(10) := 1099.08 ; # 1235
PLANT.HX2.THX(11) := 1094.03 ; # 1236
PLANT.HX2.THX(12) := 1088.69 ; # 1237
PLANT.HX2.THX(13) := 1083.03 ; # 1238
PLANT.HX2.THX(14) := 1077.05 ; # 1239
PLANT.HX2.THX(15) := 1070.72 ; # 1240
PLANT.HX2.THX(16) := 1064.02 ; # 1241
PLANT.HX2.THX(17) := 1056.93 ; # 1242
PLANT.HX2.THX(18) := 1049.43 ; # 1243
PLANT.HX2.THX(19) := 1041.49 ; # 1244
PLANT.HX2.THX(20) := 1033.09 ; # 1245
PLANT.HX2.THX(21) := 1024.2 ; # 1246
PLANT.HX2.THX(22) := 1014.8 ; # 1247
PLANT.HX2.THX(23) := 1004.84 ; # 1248
PLANT.HX2.THX(24) := 994.314 ; # 1249
PLANT.HX2.THX(25) := 983.171 ; # 1250
PLANT.HX2.THX(26) := 971.38 ; # 1251
PLANT.HX2.THX(27) := 958.903 ; # 1252
PLANT.HX2.THX(28) := 945.701 ; # 1253
PLANT.HX2.THX(29) := 931.731 ; # 1254
PLANT.HX2.THX(30) := 916.948 ; # 1255
PLANT.HX2.THX(31) := 901.306 ; # 1256
PLANT.HX2.THX(32) := 884.754 ; # 1257
PLANT.HX2.THX(33) := 867.239 ; # 1258
PLANT.HX2.THX(34) := 848.706 ; # 1259
PLANT.HX2.THX(35) := 829.095 ; # 1260
PLANT.HX2.THX(36) := 808.344 ; # 1261
PLANT.HX2.THX(37) := 786.385 ; # 1262
PLANT.HX2.THX(38) := 763.15 ; # 1263
PLANT.HX2.THX(39) := 738.563 ; # 1264
PLANT.HX2.THX(40) := 712.554 ; # 1265
PLANT.PIPEC2.RHO(1) := 3.72389 ; # 1266
PLANT.PIPEC2.RHO(2) := 3.55368 ; # 1267
PLANT.PIPEC2.RHO(3) := 3.4055 ; # 1268
PLANT.PIPEC2.RHO(4) := 3.27543 ; # 1269
PLANT.PIPEC2.RHO(5) := 3.16041 ; # 1270
PLANT.PIPEC2.RHO(6) := 3.05803 ; # 1271
PLANT.PIPEC2.RHO(7) := 2.96638 ; # 1272
PLANT.PIPEC2.RHO(8) := 2.88388 ; # 1273
PLANT.PIPEC2.RHO(9) := 2.80928 ; # 1274
PLANT.PIPEC2.RHO(10) := 2.74152 ; # 1275
PLANT.PIPEC2.RHO(11) := 2.67973 ; # 1276
PLANT.PIPEC2.RHO(12) := 2.62316 ; # 1277
PLANT.PIPEC2.RHO(13) := 2.57121 ; # 1278
PLANT.PIPEC2.RHO(14) := 2.52333 ; # 1279
PLANT.PIPEC2.RHO(15) := 2.47909 ; # 1280
PLANT.PIPEC2.RHO(16) := 2.43808 ; # 1281
PLANT.PIPEC2.RHO(17) := 2.39997 ; # 1282
PLANT.PIPEC2.RHO(18) := 2.36446 ; # 1283
PLANT.PIPEC2.RHO(19) := 2.3313 ; # 1284
PLANT.PIPEC2.RHO(20) := 2.30026 ; # 1285
PLANT.PIPEC2.RHO(21) := 2.27113 ; # 1286
PLANT.PIPEC2.RHO(22) := 2.24375 ; # 1287
PLANT.PIPEC2.RHO(23) := 2.21796 ; # 1288
PLANT.PIPEC2.RHO(24) := 2.19361 ; # 1289
PLANT.PIPEC2.RHO(25) := 2.17058 ; # 1290
PLANT.PIPEC2.RHO(26) := 2.14875 ; # 1291
PLANT.PIPEC2.RHO(27) := 2.12804 ; # 1292
PLANT.PIPEC2.RHO(28) := 2.10834 ; # 1293
PLANT.PIPEC2.RHO(29) := 2.08957 ; # 1294
PLANT.PIPEC2.RHO(30) := 2.07166 ; # 1295
PLANT.PIPEC2.RHO(31) := 2.05454 ; # 1296
PLANT.PIPEC2.RHO(32) := 2.03814 ; # 1297
PLANT.PIPEC2.RHO(33) := 2.02242 ; # 1298
PLANT.PIPEC2.RHO(34) := 2.00733 ; # 1299
PLANT.PIPEC2.RHO(35) := 1.9928 ; # 1300
PLANT.PIPEC2.RHO(36) := 1.97881 ; # 1301
PLANT.PIPEC2.RHO(37) := 1.96531 ; # 1302
PLANT.PIPEC2.RHO(38) := 1.95226 ; # 1303
PLANT.PIPEC2.RHO(39) := 1.93964 ; # 1304
PLANT.PIPEC2.RHO(40) := 1.92741 ; # 1305
PLANT.PIPEC2.P(1) := 1.10918e+006 ; # 1306
PLANT.PIPEC2.P(2) := 1.10717e+006 ; # 1307
PLANT.PIPEC2.P(3) := 1.1051e+006 ; # 1308
PLANT.PIPEC2.P(4) := 1.10297e+006 ; # 1309
PLANT.PIPEC2.P(5) := 1.10079e+006 ; # 1310
PLANT.PIPEC2.P(6) := 1.09855e+006 ; # 1311
PLANT.PIPEC2.P(7) := 1.09626e+006 ; # 1312
PLANT.PIPEC2.P(8) := 1.09392e+006 ; # 1313
PLANT.PIPEC2.P(9) := 1.09154e+006 ; # 1314
PLANT.PIPEC2.P(10) := 1.08911e+006 ; # 1315
PLANT.PIPEC2.P(11) := 1.08663e+006 ; # 1316
PLANT.PIPEC2.P(12) := 1.08412e+006 ; # 1317
PLANT.PIPEC2.P(13) := 1.08156e+006 ; # 1318
PLANT.PIPEC2.P(14) := 1.07897e+006 ; # 1319
PLANT.PIPEC2.P(15) := 1.07634e+006 ; # 1320
PLANT.PIPEC2.P(16) := 1.07368e+006 ; # 1321
PLANT.PIPEC2.P(17) := 1.07098e+006 ; # 1322
```

```
1325   PLANT.PIPEC2.P(18) := 1.06825e+006 ; # 1323
1326   PLANT.PIPEC2.P(19) := 1.06548e+006 ; # 1324
1327   PLANT.PIPEC2.P(20) := 1.06269e+006 ; # 1325
1328   PLANT.PIPEC2.P(21) := 1.05986e+006 ; # 1326
1329   PLANT.PIPEC2.P(22) := 1.05701e+006 ; # 1327
1330   PLANT.PIPEC2.P(23) := 1.05413e+006 ; # 1328
1331   PLANT.PIPEC2.P(24) := 1.05123e+006 ; # 1329
1332   PLANT.PIPEC2.P(25) := 1.04829e+006 ; # 1330
1333   PLANT.PIPEC2.P(26) := 1.04533e+006 ; # 1331
1334   PLANT.PIPEC2.P(27) := 1.04235e+006 ; # 1332
1335   PLANT.PIPEC2.P(28) := 1.03935e+006 ; # 1333
1336   PLANT.PIPEC2.P(29) := 1.03632e+006 ; # 1334
1337   PLANT.PIPEC2.P(30) := 1.03326e+006 ; # 1335
1338   PLANT.PIPEC2.P(31) := 1.03019e+006 ; # 1336
1339   PLANT.PIPEC2.P(32) := 1.02709e+006 ; # 1337
1340   PLANT.PIPEC2.P(33) := 1.02398e+006 ; # 1338
1341   PLANT.PIPEC2.P(34) := 1.02084e+006 ; # 1339
1342   PLANT.PIPEC2.P(35) := 1.01768e+006 ; # 1340
1343   PLANT.PIPEC2.P(36) := 1.0145e+006 ; # 1341
1344   PLANT.PIPEC2.P(37) := 1.0113e+006 ; # 1342
1345   PLANT.PIPEC2.P(38) := 1.00808e+006 ; # 1343
1346   PLANT.PIPEC2.P(39) := 1.00485e+006 ; # 1344
1347   PLANT.PIPEC2.P(40) := 1.00159e+006 ; # 1345
1348   PLANT.PIPEC2.H(1) := 784075 ; # 1346
1349   PLANT.PIPEC2.H(2) := 851066 ; # 1347
1350   PLANT.PIPEC2.H(3) := 914376 ; # 1348
1351   PLANT.PIPEC2.H(4) := 974206 ; # 1349
1352   PLANT.PIPEC2.H(5) := 1.03075e+006 ; # 1350
1353   PLANT.PIPEC2.H(6) := 1.08418e+006 ; # 1351
1354   PLANT.PIPEC2.H(7) := 1.13468e+006 ; # 1352
1355   PLANT.PIPEC2.H(8) := 1.1824e+006 ; # 1353
1356   PLANT.PIPEC2.H(9) := 1.2275e+006 ; # 1354
1357   PLANT.PIPEC2.H(10) := 1.27012e+006 ; # 1355
1358   PLANT.PIPEC2.H(11) := 1.3104e+006 ; # 1356
1359   PLANT.PIPEC2.H(12) := 1.34847e+006 ; # 1357
1360   PLANT.PIPEC2.H(13) := 1.38444e+006 ; # 1358
1361   PLANT.PIPEC2.H(14) := 1.41843e+006 ; # 1359
1362   PLANT.PIPEC2.H(15) := 1.45056e+006 ; # 1360
1363   PLANT.PIPEC2.H(16) := 1.48092e+006 ; # 1361
1364   PLANT.PIPEC2.H(17) := 1.50962e+006 ; # 1362
1365   PLANT.PIPEC2.H(18) := 1.53673e+006 ; # 1363
1366   PLANT.PIPEC2.H(19) := 1.56236e+006 ; # 1364
1367   PLANT.PIPEC2.H(20) := 1.58657e+006 ; # 1365
1368   PLANT.PIPEC2.H(21) := 1.60946e+006 ; # 1366
1369   PLANT.PIPEC2.H(22) := 1.63109e+006 ; # 1367
1370   PLANT.PIPEC2.H(23) := 1.65153e+006 ; # 1368
1371   PLANT.PIPEC2.H(24) := 1.67085e+006 ; # 1369
1372   PLANT.PIPEC2.H(25) := 1.6891e+006 ; # 1370
1373   PLANT.PIPEC2.H(26) := 1.70635e+006 ; # 1371
1374   PLANT.PIPEC2.H(27) := 1.72265e+006 ; # 1372
1375   PLANT.PIPEC2.H(28) := 1.73806e+006 ; # 1373
1376   PLANT.PIPEC2.H(29) := 1.75262e+006 ; # 1374
1377   PLANT.PIPEC2.H(30) := 1.76638e+006 ; # 1375
1378   PLANT.PIPEC2.H(31) := 1.77939e+006 ; # 1376
1379   PLANT.PIPEC2.H(32) := 1.79168e+006 ; # 1377
1380   PLANT.PIPEC2.H(33) := 1.80329e+006 ; # 1378
1381   PLANT.PIPEC2.H(34) := 1.81427e+006 ; # 1379
1382   PLANT.PIPEC2.H(35) := 1.82464e+006 ; # 1380
1383   PLANT.PIPEC2.H(36) := 1.83444e+006 ; # 1381
1384   PLANT.PIPEC2.H(37) := 1.8437e+006 ; # 1382
1385   PLANT.PIPEC2.H(38) := 1.85246e+006 ; # 1383
1386   PLANT.PIPEC2.H(39) := 1.86073e+006 ; # 1384
1387   PLANT.PIPEC2.H(40) := 1.86855e+006 ; # 1385
1388   PLANT.PIPEC2.H0 := 713180 ; # 1386
1389   PLANT.PIPEC2.T(1) := 645.577 ; # 1387
1390   PLANT.PIPEC2.T(2) := 675.275 ; # 1388
1391   PLANT.PIPEC2.T(3) := 703.34 ; # 1389
1392   PLANT.PIPEC2.T(4) := 729.862 ; # 1390
1393   PLANT.PIPEC2.T(5) := 754.927 ; # 1391
1394   PLANT.PIPEC2.T(6) := 778.614 ; # 1392
1395   PLANT.PIPEC2.T(7) := 801 ; # 1393
1396   PLANT.PIPEC2.T(8) := 822.155 ; # 1394
1397   PLANT.PIPEC2.T(9) := 842.147 ; # 1395
1398   PLANT.PIPEC2.T(10) := 861.041 ; # 1396
1399   PLANT.PIPEC2.T(11) := 878.896 ; # 1397
1400   PLANT.PIPEC2.T(12) := 895.77 ; # 1398
1401   PLANT.PIPEC2.T(13) := 911.717 ; # 1399
1402   PLANT.PIPEC2.T(14) := 926.787 ; # 1400
1403   PLANT.PIPEC2.T(15) := 941.029 ; # 1401
1404   PLANT.PIPEC2.T(16) := 954.488 ; # 1402
1405   PLANT.PIPEC2.T(17) := 967.207 ; # 1403
1406   PLANT.PIPEC2.T(18) := 979.228 ; # 1404
1407   PLANT.PIPEC2.T(19) := 990.587 ; # 1405
1408   PLANT.PIPEC2.T(20) := 1001.32 ; # 1406
1409   PLANT.PIPEC2.T(21) := 1011.47 ; # 1407
1410   PLANT.PIPEC2.T(22) := 1021.06 ; # 1408
1411   PLANT.PIPEC2.T(23) := 1030.12 ; # 1409
1412   PLANT.PIPEC2.T(24) := 1038.68 ; # 1410
1413   PLANT.PIPEC2.T(25) := 1046.77 ; # 1411
1414   PLANT.PIPEC2.T(26) := 1054.42 ; # 1412
1415   PLANT.PIPEC2.T(27) := 1061.65 ; # 1413
1416   PLANT.PIPEC2.T(28) := 1068.48 ; # 1414
```

```
1417   PLANT.PIPEC2.T(29) := 1074.93 ; # 1415
1418   PLANT.PIPEC2.T(30) := 1081.03 ; # 1416
1419   PLANT.PIPEC2.T(31) := 1086.8 ; # 1417
1420   PLANT.PIPEC2.T(32) := 1092.24 ; # 1418
1421   PLANT.PIPEC2.T(33) := 1097.39 ; # 1419
1422   PLANT.PIPEC2.T(34) := 1102.26 ; # 1420
1423   PLANT.PIPEC2.T(35) := 1106.86 ; # 1421
1424   PLANT.PIPEC2.T(36) := 1111.2 ; # 1422
1425   PLANT.PIPEC2.T(37) := 1115.31 ; # 1423
1426   PLANT.PIPEC2.T(38) := 1119.19 ; # 1424
1427   PLANT.PIPEC2.T(39) := 1122.86 ; # 1425
1428   PLANT.PIPEC2.T(40) := 1126.32 ; # 1426
1429   PLANT.PIPEC2.TEXTRNL(1) := 712.554 ; # 1427
1430   PLANT.PIPEC2.TEXTRNL(2) := 738.563 ; # 1428
1431   PLANT.PIPEC2.TEXTRNL(3) := 763.15 ; # 1429
1432   PLANT.PIPEC2.TEXTRNL(4) := 786.385 ; # 1430
1433   PLANT.PIPEC2.TEXTRNL(5) := 808.344 ; # 1431
1434   PLANT.PIPEC2.TEXTRNL(6) := 829.095 ; # 1432
1435   PLANT.PIPEC2.TEXTRNL(7) := 848.706 ; # 1433
1436   PLANT.PIPEC2.TEXTRNL(8) := 867.239 ; # 1434
1437   PLANT.PIPEC2.TEXTRNL(9) := 884.754 ; # 1435
1438   PLANT.PIPEC2.TEXTRNL(10) := 901.306 ; # 1436
1439   PLANT.PIPEC2.TEXTRNL(11) := 916.948 ; # 1437
1440   PLANT.PIPEC2.TEXTRNL(12) := 931.731 ; # 1438
1441   PLANT.PIPEC2.TEXTRNL(13) := 945.701 ; # 1439
1442   PLANT.PIPEC2.TEXTRNL(14) := 958.903 ; # 1440
1443   PLANT.PIPEC2.TEXTRNL(15) := 971.38 ; # 1441
1444   PLANT.PIPEC2.TEXTRNL(16) := 983.171 ; # 1442
1445   PLANT.PIPEC2.TEXTRNL(17) := 994.314 ; # 1443
1446   PLANT.PIPEC2.TEXTRNL(18) := 1004.84 ; # 1444
1447   PLANT.PIPEC2.TEXTRNL(19) := 1014.8 ; # 1445
1448   PLANT.PIPEC2.TEXTRNL(20) := 1024.2 ; # 1446
1449   PLANT.PIPEC2.TEXTRNL(21) := 1033.09 ; # 1447
1450   PLANT.PIPEC2.TEXTRNL(22) := 1041.49 ; # 1448
1451   PLANT.PIPEC2.TEXTRNL(23) := 1049.43 ; # 1449
1452   PLANT.PIPEC2.TEXTRNL(24) := 1056.93 ; # 1450
1453   PLANT.PIPEC2.TEXTRNL(25) := 1064.02 ; # 1451
1454   PLANT.PIPEC2.TEXTRNL(26) := 1070.72 ; # 1452
1455   PLANT.PIPEC2.TEXTRNL(27) := 1077.05 ; # 1453
1456   PLANT.PIPEC2.TEXTRNL(28) := 1083.03 ; # 1454
1457   PLANT.PIPEC2.TEXTRNL(29) := 1088.69 ; # 1455
1458   PLANT.PIPEC2.TEXTRNL(30) := 1094.03 ; # 1456
1459   PLANT.PIPEC2.TEXTRNL(31) := 1099.08 ; # 1457
1460   PLANT.PIPEC2.TEXTRNL(32) := 1103.85 ; # 1458
1461   PLANT.PIPEC2.TEXTRNL(33) := 1108.36 ; # 1459
1462   PLANT.PIPEC2.TEXTRNL(34) := 1112.63 ; # 1460
1463   PLANT.PIPEC2.TEXTRNL(35) := 1116.65 ; # 1461
1464   PLANT.PIPEC2.TEXTRNL(36) := 1120.46 ; # 1462
1465   PLANT.PIPEC2.TEXTRNL(37) := 1124.06 ; # 1463
1466   PLANT.PIPEC2.TEXTRNL(38) := 1127.46 ; # 1464
1467   PLANT.PIPEC2.TEXTRNL(39) := 1130.67 ; # 1465
1468   PLANT.PIPEC2.TEXTRNL(40) := 1133.71 ; # 1466
1469   PLANT.PIPEC2.F(1) := 43.3743 ; # 1467
1470   PLANT.PIPEC2.F(2) := 43.3743 ; # 1468
1471   PLANT.PIPEC2.F(3) := 43.3743 ; # 1469
1472   PLANT.PIPEC2.F(4) := 43.3743 ; # 1470
1473   PLANT.PIPEC2.F(5) := 43.3743 ; # 1471
1474   PLANT.PIPEC2.F(6) := 43.3743 ; # 1472
1475   PLANT.PIPEC2.F(7) := 43.3743 ; # 1473
1476   PLANT.PIPEC2.F(8) := 43.3743 ; # 1474
1477   PLANT.PIPEC2.F(9) := 43.3743 ; # 1475
1478   PLANT.PIPEC2.F(10) := 43.3743 ; # 1476
1479   PLANT.PIPEC2.F(11) := 43.3743 ; # 1477
1480   PLANT.PIPEC2.F(12) := 43.3743 ; # 1478
1481   PLANT.PIPEC2.F(13) := 43.3743 ; # 1479
1482   PLANT.PIPEC2.F(14) := 43.3743 ; # 1480
1483   PLANT.PIPEC2.F(15) := 43.3743 ; # 1481
1484   PLANT.PIPEC2.F(16) := 43.3743 ; # 1482
1485   PLANT.PIPEC2.F(17) := 43.3743 ; # 1483
1486   PLANT.PIPEC2.F(18) := 43.3743 ; # 1484
1487   PLANT.PIPEC2.F(19) := 43.3743 ; # 1485
1488   PLANT.PIPEC2.F(20) := 43.3743 ; # 1486
1489   PLANT.PIPEC2.F(21) := 43.3743 ; # 1487
1490   PLANT.PIPEC2.F(22) := 43.3743 ; # 1488
1491   PLANT.PIPEC2.F(23) := 43.3743 ; # 1489
1492   PLANT.PIPEC2.F(24) := 43.3743 ; # 1490
1493   PLANT.PIPEC2.F(25) := 43.3743 ; # 1491
1494   PLANT.PIPEC2.F(26) := 43.3743 ; # 1492
1495   PLANT.PIPEC2.F(27) := 43.3743 ; # 1493
1496   PLANT.PIPEC2.F(28) := 43.3743 ; # 1494
1497   PLANT.PIPEC2.F(29) := 43.3743 ; # 1495
1498   PLANT.PIPEC2.F(30) := 43.3743 ; # 1496
1499   PLANT.PIPEC2.F(31) := 43.3743 ; # 1497
1500   PLANT.PIPEC2.F(32) := 43.3743 ; # 1498
1501   PLANT.PIPEC2.F(33) := 43.3743 ; # 1499
1502   PLANT.PIPEC2.F(34) := 43.3743 ; # 1500
1503   PLANT.PIPEC2.F(35) := 43.3743 ; # 1501
1504   PLANT.PIPEC2.F(36) := 43.3743 ; # 1502
1505   PLANT.PIPEC2.F(37) := 43.3743 ; # 1503
1506   PLANT.PIPEC2.F(38) := 43.3743 ; # 1504
1507   PLANT.PIPEC2.F(39) := 43.3743 ; # 1505
1508   PLANT.PIPEC2.F(40) := 43.3743 ; # 1506
```

```
PLANT.PIPEC2.F_IN := 43.3743 ; # 1507
PLANT.PIPEC2.PSI(1) := 1.81063e+006 ; # 1508
PLANT.PIPEC2.PSI(2) := 1.91725e+006 ; # 1509
PLANT.PIPEC2.PSI(3) := 2.00881e+006 ; # 1510
PLANT.PIPEC2.PSI(4) := 2.08797e+006 ; # 1511
PLANT.PIPEC2.PSI(5) := 2.1568e+006 ; # 1512
PLANT.PIPEC2.PSI(6) := 2.21692e+006 ; # 1513
PLANT.PIPEC2.PSI(7) := 2.26963e+006 ; # 1514
PLANT.PIPEC2.PSI(8) := 2.31599e+006 ; # 1515
PLANT.PIPEC2.PSI(9) := 2.35686e+006 ; # 1516
PLANT.PIPEC2.PSI(10) := 2.39296e+006 ; # 1517
PLANT.PIPEC2.PSI(11) := 2.42488e+006 ; # 1518
PLANT.PIPEC2.PSI(12) := 2.45312e+006 ; # 1519
PLANT.PIPEC2.PSI(13) := 2.47811e+006 ; # 1520
PLANT.PIPEC2.PSI(14) := 2.50021e+006 ; # 1521
PLANT.PIPEC2.PSI(15) := 2.51973e+006 ; # 1522
PLANT.PIPEC2.PSI(16) := 2.53693e+006 ; # 1523
PLANT.PIPEC2.PSI(17) := 2.55205e+006 ; # 1524
PLANT.PIPEC2.PSI(18) := 2.5653e+006 ; # 1525
PLANT.PIPEC2.PSI(19) := 2.57684e+006 ; # 1526
PLANT.PIPEC2.PSI(20) := 2.58684e+006 ; # 1527
PLANT.PIPEC2.PSI(21) := 2.59544e+006 ; # 1528
PLANT.PIPEC2.PSI(22) := 2.60275e+006 ; # 1529
PLANT.PIPEC2.PSI(23) := 2.60889e+006 ; # 1530
PLANT.PIPEC2.PSI(24) := 2.61395e+006 ; # 1531
PLANT.PIPEC2.PSI(25) := 2.61803e+006 ; # 1532
PLANT.PIPEC2.PSI(26) := 2.62119e+006 ; # 1533
PLANT.PIPEC2.PSI(27) := 2.62352e+006 ; # 1534
PLANT.PIPEC2.PSI(28) := 2.62507e+006 ; # 1535
PLANT.PIPEC2.PSI(29) := 2.62591e+006 ; # 1536
PLANT.PIPEC2.PSI(30) := 2.62607e+006 ; # 1537
PLANT.PIPEC2.PSI(31) := 2.62562e+006 ; # 1538
PLANT.PIPEC2.PSI(32) := 2.6246e+006 ; # 1539
PLANT.PIPEC2.PSI(33) := 2.62304e+006 ; # 1540
PLANT.PIPEC2.PSI(34) := 2.62099e+006 ; # 1541
PLANT.PIPEC2.PSI(35) := 2.61847e+006 ; # 1542
PLANT.PIPEC2.PSI(36) := 2.61551e+006 ; # 1543
PLANT.PIPEC2.PSI(37) := 2.61215e+006 ; # 1544
PLANT.PIPEC2.PSI(38) := 2.6084e+006 ; # 1545
PLANT.PIPEC2.PSI(39) := 2.6043e+006 ; # 1546
PLANT.PIPEC2.PSI(40) := 2.59986e+006 ; # 1547
PLANT.PIPEC2.HT.RHOPIPEHE := 2.35217 ; # 1548
PLANT.PIPEC2.HT.VPIPE := 65.6203 ; # 1549
PLANT.PIPEC2.HT.U := 1880.53 ; # 1550
```

HTLQSS_SS_IN_INI.INITIAL

```
# Values for differential variables in computation HTLFULLDYN_SS
# Saved at time 1000
PLANT.HX.THX(1) = 889.835 ; # 1
PLANT.HX.THX(2) = 900.086 ; # 2
PLANT.HX.THX(3) = 910.162 ; # 3
PLANT.HX.THX(4) = 920.011 ; # 4
PLANT.HX.THX(5) = 929.638 ; # 5
PLANT.HX.THX(6) = 939.048 ; # 6
PLANT.HX.THX(7) = 948.245 ; # 7
PLANT.HX.THX(8) = 957.235 ; # 8
PLANT.HX.THX(9) = 966.021 ; # 9
PLANT.HX.THX(10) = 974.61 ; # 10
PLANT.HX.THX(11) = 983.004 ; # 11
PLANT.HX.THX(12) = 991.209 ; # 12
PLANT.HX.THX(13) = 999.229 ; # 13
PLANT.HX.THX(14) = 1007.07 ; # 14
PLANT.HX.THX(15) = 1014.73 ; # 15
PLANT.HX.THX(16) = 1022.22 ; # 16
PLANT.HX.THX(17) = 1029.54 ; # 17
PLANT.HX.THX(18) = 1036.69 ; # 18
PLANT.HX.THX(19) = 1043.69 ; # 19
PLANT.HX.THX(20) = 1050.52 ; # 20
PLANT.HX.THX(21) = 1057.2 ; # 21
PLANT.HX.THX(22) = 1063.73 ; # 22
PLANT.HX.THX(23) = 1070.12 ; # 23
PLANT.HX.THX(24) = 1076.36 ; # 24
PLANT.HX.THX(25) = 1082.45 ; # 25
PLANT.HX.THX(26) = 1088.41 ; # 26
PLANT.HX.THX(27) = 1094.24 ; # 27
PLANT.HX.THX(28) = 1099.93 ; # 28
PLANT.HX.THX(29) = 1105.5 ; # 29
PLANT.HX.THX(30) = 1110.94 ; # 30
PLANT.HX.THX(31) = 1116.26 ; # 31
PLANT.HX.THX(32) = 1121.45 ; # 32
PLANT.HX.THX(33) = 1126.53 ; # 33
PLANT.HX.THX(34) = 1131.5 ; # 34
```

```
PLANT.HX.THX(35) = 1136.35 ; # 35
PLANT.HX.THX(36) = 1141.1 ; # 36
PLANT.HX.THX(37) = 1145.73 ; # 37
PLANT.HX.THX(38) = 1150.27 ; # 38
PLANT.HX.THX(39) = 1154.7 ; # 39
PLANT.HX.THX(40) = 1159 ; # 40
PLANT.LOOP1.PSIEXTRA(1) = 1.28551e+006 ; # 141
PLANT.LOOP1.PSIEXTRA(2) = 1.28551e+006 ; # 141
PLANT.LOOP1.PSI(1) = 1.28551e+006 ; # 141
PLANT.LOOP1.PSI(2) = 1.27902e+006 ; # 142
PLANT.LOOP1.PSI(3) = 1.27441e+006 ; # 143
PLANT.LOOP1.PSI(4) = 1.26979e+006 ; # 144
PLANT.LOOP1.PSI(5) = 1.26517e+006 ; # 145
PLANT.LOOP1.PSI(6) = 1.26054e+006 ; # 146
PLANT.LOOP1.PSI(7) = 1.25591e+006 ; # 147
PLANT.LOOP1.PSI(8) = 1.25127e+006 ; # 148
PLANT.LOOP1.PSI(9) = 1.24663e+006 ; # 149
PLANT.LOOP1.PSI(10) = 1.24198e+006 ; # 150
PLANT.LOOP1.PSI(11) = 1.27601e+006 ; # 151
PLANT.LOOP1.PSI(12) = 1.29599e+006 ; # 152
PLANT.LOOP1.PSI(13) = 1.31449e+006 ; # 153
PLANT.LOOP1.PSI(14) = 1.33196e+006 ; # 154
PLANT.LOOP1.PSI(15) = 1.34847e+006 ; # 155
PLANT.LOOP1.PSI(16) = 1.36409e+006 ; # 156
PLANT.LOOP1.PSI(17) = 1.37885e+006 ; # 157
PLANT.LOOP1.PSI(18) = 1.39281e+006 ; # 158
PLANT.LOOP1.PSI(19) = 1.40601e+006 ; # 159
PLANT.LOOP1.PSI(20) = 1.4185e+006 ; # 160
PLANT.LOOP1.PSI(21) = 1.4303e+006 ; # 161
PLANT.LOOP1.PSI(22) = 1.44146e+006 ; # 162
PLANT.LOOP1.PSI(23) = 1.45201e+006 ; # 163
PLANT.LOOP1.PSI(24) = 1.46198e+006 ; # 164
PLANT.LOOP1.PSI(25) = 1.4714e+006 ; # 165
PLANT.LOOP1.PSI(26) = 1.4803e+006 ; # 166
PLANT.LOOP1.PSI(27) = 1.48869e+006 ; # 167
PLANT.LOOP1.PSI(28) = 1.49661e+006 ; # 168
PLANT.LOOP1.PSI(29) = 1.50407e+006 ; # 169
PLANT.LOOP1.PSI(30) = 1.5111e+006 ; # 170
PLANT.LOOP1.PSI(31) = 1.51772e+006 ; # 171
PLANT.LOOP1.PSI(32) = 1.52393e+006 ; # 172
PLANT.LOOP1.PSI(33) = 1.52977e+006 ; # 173
PLANT.LOOP1.PSI(34) = 1.53525e+006 ; # 174
PLANT.LOOP1.PSI(35) = 1.54038e+006 ; # 175
PLANT.LOOP1.PSI(36) = 1.54518e+006 ; # 176
PLANT.LOOP1.PSI(37) = 1.54965e+006 ; # 177
PLANT.LOOP1.PSI(38) = 1.55382e+006 ; # 178
PLANT.LOOP1.PSI(39) = 1.5577e+006 ; # 179
PLANT.LOOP1.PSI(40) = 1.56129e+006 ; # 180
PLANT.LOOP1.PSI(41) = 1.56461e+006 ; # 181
PLANT.LOOP1.PSI(42) = 1.56766e+006 ; # 182
PLANT.LOOP1.PSI(43) = 1.57047e+006 ; # 183
PLANT.LOOP1.PSI(44) = 1.57303e+006 ; # 184
PLANT.LOOP1.PSI(45) = 1.57536e+006 ; # 185
PLANT.LOOP1.PSI(46) = 1.57746e+006 ; # 186
PLANT.LOOP1.PSI(47) = 1.57935e+006 ; # 187
PLANT.LOOP1.PSI(48) = 1.58102e+006 ; # 188
PLANT.LOOP1.PSI(49) = 1.58249e+006 ; # 189
PLANT.LOOP1.PSI(50) = 1.58376e+006 ; # 190
PLANT.LOOP1.PSI(51) = 1.55653e+006 ; # 191
PLANT.LOOP1.PSI(52) = 1.5504e+006 ; # 192
PLANT.LOOP1.PSI(53) = 1.5443e+006 ; # 193
PLANT.LOOP1.PSI(54) = 1.53819e+006 ; # 194
PLANT.LOOP1.PSI(55) = 1.53207e+006 ; # 195
PLANT.LOOP1.PSI(56) = 1.52594e+006 ; # 196
PLANT.LOOP1.PSI(57) = 1.5198e+006 ; # 197
PLANT.LOOP1.PSI(58) = 1.51364e+006 ; # 198
PLANT.LOOP1.PSI(59) = 1.50748e+006 ; # 199
PLANT.LOOP1.PSI(60) = 1.5013e+006 ; # 200
PLANT.LOOP1.PSI(61) = 1.51696e+006 ; # 201
PLANT.LOOP1.PSI(62) = 1.51365e+006 ; # 202
PLANT.LOOP1.PSI(63) = 1.51012e+006 ; # 203
PLANT.LOOP1.PSI(64) = 1.50644e+006 ; # 204
PLANT.LOOP1.PSI(65) = 1.5026e+006 ; # 205
PLANT.LOOP1.PSI(66) = 1.49858e+006 ; # 206
PLANT.LOOP1.PSI(67) = 1.49439e+006 ; # 207
PLANT.LOOP1.PSI(68) = 1.49e+006 ; # 208
PLANT.LOOP1.PSI(69) = 1.48539e+006 ; # 209
PLANT.LOOP1.PSI(70) = 1.48056e+006 ; # 210
PLANT.LOOP1.PSI(71) = 1.47548e+006 ; # 211
PLANT.LOOP1.PSI(72) = 1.47014e+006 ; # 212
PLANT.LOOP1.PSI(73) = 1.46451e+006 ; # 213
PLANT.LOOP1.PSI(74) = 1.45857e+006 ; # 214
PLANT.LOOP1.PSI(75) = 1.45229e+006 ; # 215
PLANT.LOOP1.PSI(76) = 1.44565e+006 ; # 216
PLANT.LOOP1.PSI(77) = 1.43862e+006 ; # 217
PLANT.LOOP1.PSI(78) = 1.43116e+006 ; # 218
PLANT.LOOP1.PSI(79) = 1.42323e+006 ; # 219
PLANT.LOOP1.PSI(80) = 1.4148e+006 ; # 220
PLANT.LOOP1.PSI(81) = 1.40582e+006 ; # 221
PLANT.LOOP1.PSI(82) = 1.39623e+006 ; # 222
PLANT.LOOP1.PSI(83) = 1.38599e+006 ; # 223
PLANT.LOOP1.PSI(84) = 1.37502e+006 ; # 224
```

```
PLANT.LOOP1.PSI(85) = 1.36326e+006 ; # 225
PLANT.LOOP1.PSI(86) = 1.35062e+006 ; # 226
PLANT.LOOP1.PSI(87) = 1.33702e+006 ; # 227
PLANT.LOOP1.PSI(88) = 1.32234e+006 ; # 228
PLANT.LOOP1.PSI(89) = 1.30649e+006 ; # 229
PLANT.LOOP1.PSI(90) = 1.28931e+006 ; # 230
PLANT.LOOP1.PSI(91) = 1.27066e+006 ; # 231
PLANT.LOOP1.PSI(92) = 1.25037e+006 ; # 232
PLANT.LOOP1.PSI(93) = 1.22822e+006 ; # 233
PLANT.LOOP1.PSI(94) = 1.20397e+006 ; # 234
PLANT.LOOP1.PSI(95) = 1.17735e+006 ; # 235
PLANT.LOOP1.PSI(96) = 1.14802e+006 ; # 236
PLANT.LOOP1.PSI(97) = 1.11558e+006 ; # 237
PLANT.LOOP1.PSI(98) = 1.07957e+006 ; # 238
PLANT.LOOP1.PSI(99) = 1.03941e+006 ; # 239
PLANT.LOOP1.PSI(100) = 994398 ; # 240
PLANT.PIPEH.RHO(1) = 2.88031 ; # 341
PLANT.PIPEH.RHO(2) = 2.88987 ; # 342
PLANT.PIPEH.RHO(3) = 2.89973 ; # 343
PLANT.PIPEH.RHO(4) = 2.90991 ; # 344
PLANT.PIPEH.RHO(5) = 2.92042 ; # 345
PLANT.PIPEH.RHO(6) = 2.93127 ; # 346
PLANT.PIPEH.RHO(7) = 2.94248 ; # 347
PLANT.PIPEH.RHO(8) = 2.95405 ; # 348
PLANT.PIPEH.RHO(9) = 2.96601 ; # 349
PLANT.PIPEH.RHO(10) = 2.97836 ; # 350
PLANT.PIPEH.RHO(11) = 2.99112 ; # 351
PLANT.PIPEH.RHO(12) = 3.00432 ; # 352
PLANT.PIPEH.RHO(13) = 3.01796 ; # 353
PLANT.PIPEH.RHO(14) = 3.03206 ; # 354
PLANT.PIPEH.RHO(15) = 3.04666 ; # 355
PLANT.PIPEH.RHO(16) = 3.06175 ; # 356
PLANT.PIPEH.RHO(17) = 3.07737 ; # 357
PLANT.PIPEH.RHO(18) = 3.09354 ; # 358
PLANT.PIPEH.RHO(19) = 3.11029 ; # 359
PLANT.PIPEH.RHO(20) = 3.12763 ; # 360
PLANT.PIPEH.RHO(21) = 3.1456 ; # 361
PLANT.PIPEH.RHO(22) = 3.16422 ; # 362
PLANT.PIPEH.RHO(23) = 3.18352 ; # 363
PLANT.PIPEH.RHO(24) = 3.20354 ; # 364
PLANT.PIPEH.RHO(25) = 3.22431 ; # 365
PLANT.PIPEH.RHO(26) = 3.24587 ; # 366
PLANT.PIPEH.RHO(27) = 3.26824 ; # 367
PLANT.PIPEH.RHO(28) = 3.29149 ; # 368
PLANT.PIPEH.RHO(29) = 3.31563 ; # 369
PLANT.PIPEH.RHO(30) = 3.34074 ; # 370
PLANT.PIPEH.RHO(31) = 3.36684 ; # 371
PLANT.PIPEH.RHO(32) = 3.394 ; # 372
PLANT.PIPEH.RHO(33) = 3.42228 ; # 373
PLANT.PIPEH.RHO(34) = 3.45173 ; # 374
PLANT.PIPEH.RHO(35) = 3.48241 ; # 375
PLANT.PIPEH.RHO(36) = 3.5144 ; # 376
PLANT.PIPEH.RHO(37) = 3.54778 ; # 377
PLANT.PIPEH.RHO(38) = 3.58262 ; # 378
PLANT.PIPEH.RHO(39) = 3.61901 ; # 379
PLANT.PIPEH.RHO(40) = 3.65698 ; # 380
PLANT.PIPEH.F(1) = 32.4687 ; # 381
PLANT.PIPEH.F(2) = 32.4687 ; # 382
PLANT.PIPEH.F(3) = 32.4687 ; # 383
PLANT.PIPEH.F(4) = 32.4687 ; # 384
PLANT.PIPEH.F(5) = 32.4687 ; # 385
PLANT.PIPEH.F(6) = 32.4687 ; # 386
PLANT.PIPEH.F(7) = 32.4687 ; # 387
PLANT.PIPEH.F(8) = 32.4687 ; # 388
PLANT.PIPEH.F(9) = 32.4687 ; # 389
PLANT.PIPEH.F(10) = 32.4687 ; # 390
PLANT.PIPEH.F(11) = 32.4687 ; # 391
PLANT.PIPEH.F(12) = 32.4687 ; # 392
PLANT.PIPEH.F(13) = 32.4687 ; # 393
PLANT.PIPEH.F(14) = 32.4687 ; # 394
PLANT.PIPEH.F(15) = 32.4687 ; # 395
PLANT.PIPEH.F(16) = 32.4687 ; # 396
PLANT.PIPEH.F(17) = 32.4687 ; # 397
PLANT.PIPEH.F(18) = 32.4687 ; # 398
PLANT.PIPEH.F(19) = 32.4687 ; # 399
PLANT.PIPEH.F(20) = 32.4687 ; # 400
PLANT.PIPEH.F(21) = 32.4687 ; # 401
PLANT.PIPEH.F(22) = 32.4687 ; # 402
PLANT.PIPEH.F(23) = 32.4687 ; # 403
PLANT.PIPEH.F(24) = 32.4687 ; # 404
PLANT.PIPEH.F(25) = 32.4687 ; # 405
PLANT.PIPEH.F(26) = 32.4687 ; # 406
PLANT.PIPEH.F(27) = 32.4687 ; # 407
PLANT.PIPEH.F(28) = 32.4687 ; # 408
PLANT.PIPEH.F(29) = 32.4687 ; # 409
PLANT.PIPEH.F(30) = 32.4687 ; # 410
PLANT.PIPEH.F(31) = 32.4687 ; # 411
PLANT.PIPEH.F(32) = 32.4687 ; # 412
PLANT.PIPEH.F(33) = 32.4687 ; # 413
PLANT.PIPEH.F(34) = 32.4687 ; # 414
PLANT.PIPEH.F(35) = 32.4687 ; # 415
PLANT.PIPEH.F(36) = 32.4687 ; # 416
```

```
PLANT.PIPEH.F(37) = 32.4687 ; # 417
PLANT.PIPEH.F(38) = 32.4687 ; # 418
PLANT.PIPEH.F(39) = 32.4687 ; # 419
PLANT.PIPEH.F(40) = 32.4687 ; # 420
PLANT.PIPEH.F_IN = 32.4687 ; # 421
PLANT.PIPEH.PSI(1) = 6.03571e+006 ; # 422
PLANT.PIPEH.PSI(2) = 6.01884e+006 ; # 423
PLANT.PIPEH.PSI(3) = 6.00151e+006 ; # 424
PLANT.PIPEH.PSI(4) = 5.9837e+006 ; # 425
PLANT.PIPEH.PSI(5) = 5.96538e+006 ; # 426
PLANT.PIPEH.PSI(6) = 5.94655e+006 ; # 427
PLANT.PIPEH.PSI(7) = 5.92717e+006 ; # 428
PLANT.PIPEH.PSI(8) = 5.90723e+006 ; # 429
PLANT.PIPEH.PSI(9) = 5.88672e+006 ; # 430
PLANT.PIPEH.PSI(10) = 5.86559e+006 ; # 431
PLANT.PIPEH.PSI(11) = 5.84383e+006 ; # 432
PLANT.PIPEH.PSI(12) = 5.82142e+006 ; # 433
PLANT.PIPEH.PSI(13) = 5.79833e+006 ; # 434
PLANT.PIPEH.PSI(14) = 5.77452e+006 ; # 435
PLANT.PIPEH.PSI(15) = 5.74998e+006 ; # 436
PLANT.PIPEH.PSI(16) = 5.72466e+006 ; # 437
PLANT.PIPEH.PSI(17) = 5.69854e+006 ; # 438
PLANT.PIPEH.PSI(18) = 5.67158e+006 ; # 439
PLANT.PIPEH.PSI(19) = 5.64374e+006 ; # 440
PLANT.PIPEH.PSI(20) = 5.61499e+006 ; # 441
PLANT.PIPEH.PSI(21) = 5.58528e+006 ; # 442
PLANT.PIPEH.PSI(22) = 5.55457e+006 ; # 443
PLANT.PIPEH.PSI(23) = 5.52282e+006 ; # 444
PLANT.PIPEH.PSI(24) = 5.48997e+006 ; # 445
PLANT.PIPEH.PSI(25) = 5.45597e+006 ; # 446
PLANT.PIPEH.PSI(26) = 5.42078e+006 ; # 447
PLANT.PIPEH.PSI(27) = 5.38432e+006 ; # 448
PLANT.PIPEH.PSI(28) = 5.34653e+006 ; # 449
PLANT.PIPEH.PSI(29) = 5.30735e+006 ; # 450
PLANT.PIPEH.PSI(30) = 5.26672e+006 ; # 451
PLANT.PIPEH.PSI(31) = 5.22454e+006 ; # 452
PLANT.PIPEH.PSI(32) = 5.18075e+006 ; # 453
PLANT.PIPEH.PSI(33) = 5.13524e+006 ; # 454
PLANT.PIPEH.PSI(34) = 5.08794e+006 ; # 455
PLANT.PIPEH.PSI(35) = 5.03874e+006 ; # 456
PLANT.PIPEH.PSI(36) = 4.98752e+006 ; # 457
PLANT.PIPEH.PSI(37) = 4.93419e+006 ; # 458
PLANT.PIPEH.PSI(38) = 4.8786e+006 ; # 459
PLANT.PIPEH.PSI(39) = 4.82064e+006 ; # 460
PLANT.PIPEH.PSI(40) = 4.76024e+006 ; # 461
PLANT.HX2.THX(1) = 1132.12 ; # 462
PLANT.HX2.THX(2) = 1129.7 ; # 463
PLANT.HX2.THX(3) = 1127.12 ; # 464
PLANT.HX2.THX(4) = 1124.36 ; # 465
PLANT.HX2.THX(5) = 1121.43 ; # 466
PLANT.HX2.THX(6) = 1118.29 ; # 467
PLANT.HX2.THX(7) = 1114.96 ; # 468
PLANT.HX2.THX(8) = 1111.4 ; # 469
PLANT.HX2.THX(9) = 1107.6 ; # 470
PLANT.HX2.THX(10) = 1103.55 ; # 471
PLANT.HX2.THX(11) = 1099.23 ; # 472
PLANT.HX2.THX(12) = 1094.63 ; # 473
PLANT.HX2.THX(13) = 1089.72 ; # 474
PLANT.HX2.THX(14) = 1084.49 ; # 475
PLANT.HX2.THX(15) = 1078.91 ; # 476
PLANT.HX2.THX(16) = 1072.96 ; # 477
PLANT.HX2.THX(17) = 1066.61 ; # 478
PLANT.HX2.THX(18) = 1059.84 ; # 479
PLANT.HX2.THX(19) = 1052.63 ; # 480
PLANT.HX2.THX(20) = 1044.93 ; # 481
PLANT.HX2.THX(21) = 1036.73 ; # 482
PLANT.HX2.THX(22) = 1027.98 ; # 483
PLANT.HX2.THX(23) = 1018.65 ; # 484
PLANT.HX2.THX(24) = 1008.7 ; # 485
PLANT.HX2.THX(25) = 998.095 ; # 486
PLANT.HX2.THX(26) = 986.784 ; # 487
PLANT.HX2.THX(27) = 974.722 ; # 488
PLANT.HX2.THX(28) = 961.86 ; # 489
PLANT.HX2.THX(29) = 948.145 ; # 490
PLANT.HX2.THX(30) = 933.521 ; # 491
PLANT.HX2.THX(31) = 917.926 ; # 492
PLANT.HX2.THX(32) = 901.297 ; # 493
PLANT.HX2.THX(33) = 883.565 ; # 494
PLANT.HX2.THX(34) = 864.657 ; # 495
PLANT.HX2.THX(35) = 844.495 ; # 496
PLANT.HX2.THX(36) = 822.996 ; # 497
PLANT.HX2.THX(37) = 800.07 ; # 498
PLANT.HX2.THX(38) = 775.624 ; # 499
PLANT.HX2.THX(39) = 749.557 ; # 500
PLANT.HX2.THX(40) = 721.768 ; # 501
PLANT.PIPEC2.RHO(1) = 3.7063 ; # 502
PLANT.PIPEC2.RHO(2) = 3.52381 ; # 503
PLANT.PIPEC2.RHO(3) = 3.36718 ; # 504
PLANT.PIPEC2.RHO(4) = 3.2314 ; # 505
PLANT.PIPEC2.RHO(5) = 3.11269 ; # 506
PLANT.PIPEC2.RHO(6) = 3.0081 ; # 507
PLANT.PIPEC2.RHO(7) = 2.91535 ; # 508
```

```
PLANT.PIPEC2.RHO(8) = 2.83259 ; # 509
PLANT.PIPEC2.RHO(9) = 2.75835 ; # 510
PLANT.PIPEC2.RHO(10) = 2.69142 ; # 511
PLANT.PIPEC2.RHO(11) = 2.63081 ; # 512
PLANT.PIPEC2.RHO(12) = 2.5757 ; # 513
PLANT.PIPEC2.RHO(13) = 2.52539 ; # 514
PLANT.PIPEC2.RHO(14) = 2.47931 ; # 515
PLANT.PIPEC2.RHO(15) = 2.43696 ; # 516
PLANT.PIPEC2.RHO(16) = 2.39791 ; # 517
PLANT.PIPEC2.RHO(17) = 2.3618 ; # 518
PLANT.PIPEC2.RHO(18) = 2.32832 ; # 519
PLANT.PIPEC2.RHO(19) = 2.29719 ; # 520
PLANT.PIPEC2.RHO(20) = 2.26818 ; # 521
PLANT.PIPEC2.RHO(21) = 2.24106 ; # 522
PLANT.PIPEC2.RHO(22) = 2.21566 ; # 523
PLANT.PIPEC2.RHO(23) = 2.19182 ; # 524
PLANT.PIPEC2.RHO(24) = 2.16939 ; # 525
PLANT.PIPEC2.RHO(25) = 2.14823 ; # 526
PLANT.PIPEC2.RHO(26) = 2.12824 ; # 527
PLANT.PIPEC2.RHO(27) = 2.10932 ; # 528
PLANT.PIPEC2.RHO(28) = 2.09136 ; # 529
PLANT.PIPEC2.RHO(29) = 2.07429 ; # 530
PLANT.PIPEC2.RHO(30) = 2.05802 ; # 531
PLANT.PIPEC2.RHO(31) = 2.0425 ; # 532
PLANT.PIPEC2.RHO(32) = 2.02766 ; # 533
PLANT.PIPEC2.RHO(33) = 2.01345 ; # 534
PLANT.PIPEC2.RHO(34) = 1.99981 ; # 535
PLANT.PIPEC2.RHO(35) = 1.9867 ; # 536
PLANT.PIPEC2.RHO(36) = 1.97407 ; # 537
PLANT.PIPEC2.RHO(37) = 1.96189 ; # 538
PLANT.PIPEC2.RHO(38) = 1.95011 ; # 539
PLANT.PIPEC2.RHO(39) = 1.93872 ; # 540
PLANT.PIPEC2.RHO(40) = 1.92768 ; # 541
PLANT.PIPEC2.F(1) = 43.1466 ; # 542
PLANT.PIPEC2.F(2) = 43.1466 ; # 543
PLANT.PIPEC2.F(3) = 43.1466 ; # 544
PLANT.PIPEC2.F(4) = 43.1466 ; # 545
PLANT.PIPEC2.F(5) = 43.1466 ; # 546
PLANT.PIPEC2.F(6) = 43.1466 ; # 547
PLANT.PIPEC2.F(7) = 43.1466 ; # 548
PLANT.PIPEC2.F(8) = 43.1466 ; # 549
PLANT.PIPEC2.F(9) = 43.1466 ; # 550
PLANT.PIPEC2.F(10) = 43.1466 ; # 551
PLANT.PIPEC2.F(11) = 43.1466 ; # 552
PLANT.PIPEC2.F(12) = 43.1466 ; # 553
PLANT.PIPEC2.F(13) = 43.1466 ; # 554
PLANT.PIPEC2.F(14) = 43.1466 ; # 555
PLANT.PIPEC2.F(15) = 43.1466 ; # 556
PLANT.PIPEC2.F(16) = 43.1466 ; # 557
PLANT.PIPEC2.F(17) = 43.1466 ; # 558
PLANT.PIPEC2.F(18) = 43.1466 ; # 559
PLANT.PIPEC2.F(19) = 43.1466 ; # 560
PLANT.PIPEC2.F(20) = 43.1466 ; # 561
PLANT.PIPEC2.F(21) = 43.1466 ; # 562
PLANT.PIPEC2.F(22) = 43.1466 ; # 563
PLANT.PIPEC2.F(23) = 43.1466 ; # 564
PLANT.PIPEC2.F(24) = 43.1466 ; # 565
PLANT.PIPEC2.F(25) = 43.1466 ; # 566
PLANT.PIPEC2.F(26) = 43.1466 ; # 567
PLANT.PIPEC2.F(27) = 43.1466 ; # 568
PLANT.PIPEC2.F(28) = 43.1466 ; # 569
PLANT.PIPEC2.F(29) = 43.1466 ; # 570
PLANT.PIPEC2.F(30) = 43.1466 ; # 571
PLANT.PIPEC2.F(31) = 43.1466 ; # 572
PLANT.PIPEC2.F(32) = 43.1466 ; # 573
PLANT.PIPEC2.F(33) = 43.1466 ; # 574
PLANT.PIPEC2.F(34) = 43.1466 ; # 575
PLANT.PIPEC2.F(35) = 43.1466 ; # 576
PLANT.PIPEC2.F(36) = 43.1466 ; # 577
PLANT.PIPEC2.F(37) = 43.1466 ; # 578
PLANT.PIPEC2.F(38) = 43.1466 ; # 579
PLANT.PIPEC2.F(39) = 43.1466 ; # 580
PLANT.PIPEC2.F(40) = 43.1466 ; # 581
PLANT.PIPEC2.F_IN = 43.1466 ; # 582
PLANT.PIPEC2.PSI(1) = 1.82247e+006 ; # 583
PLANT.PIPEC2.PSI(2) = 1.93726e+006 ; # 584
PLANT.PIPEC2.PSI(3) = 2.03443e+006 ; # 585
PLANT.PIPEC2.PSI(4) = 2.11734e+006 ; # 586
PLANT.PIPEC2.PSI(5) = 2.18855e+006 ; # 587
PLANT.PIPEC2.PSI(6) = 2.25006e+006 ; # 588
PLANT.PIPEC2.PSI(7) = 2.30342e+006 ; # 589
PLANT.PIPEC2.PSI(8) = 2.34986e+006 ; # 590
PLANT.PIPEC2.PSI(9) = 2.3904e+006 ; # 591
PLANT.PIPEC2.PSI(10) = 2.42585e+006 ; # 592
PLANT.PIPEC2.PSI(11) = 2.45689e+006 ; # 593
PLANT.PIPEC2.PSI(12) = 2.48407e+006 ; # 594
PLANT.PIPEC2.PSI(13) = 2.50788e+006 ; # 595
PLANT.PIPEC2.PSI(14) = 2.5287e+006 ; # 596
PLANT.PIPEC2.PSI(15) = 2.54688e+006 ; # 597
PLANT.PIPEC2.PSI(16) = 2.56271e+006 ; # 598
PLANT.PIPEC2.PSI(17) = 2.57643e+006 ; # 599
PLANT.PIPEC2.PSI(18) = 2.58827e+006 ; # 600
```

```
PLANT.PIPEC2.PSI(19) = 2.59842e+006 ; # 601
PLANT.PIPEC2.PSI(20) = 2.60703e+006 ; # 602
PLANT.PIPEC2.PSI(21) = 2.61426e+006 ; # 603
PLANT.PIPEC2.PSI(22) = 2.62023e+006 ; # 604
PLANT.PIPEC2.PSI(23) = 2.62506e+006 ; # 605
PLANT.PIPEC2.PSI(24) = 2.62884e+006 ; # 606
PLANT.PIPEC2.PSI(25) = 2.63167e+006 ; # 607
PLANT.PIPEC2.PSI(26) = 2.63363e+006 ; # 608
PLANT.PIPEC2.PSI(27) = 2.63479e+006 ; # 609
PLANT.PIPEC2.PSI(28) = 2.63522e+006 ; # 610
PLANT.PIPEC2.PSI(29) = 2.63496e+006 ; # 611
PLANT.PIPEC2.PSI(30) = 2.63409e+006 ; # 612
PLANT.PIPEC2.PSI(31) = 2.63263e+006 ; # 613
PLANT.PIPEC2.PSI(32) = 2.63064e+006 ; # 614
PLANT.PIPEC2.PSI(33) = 2.62816e+006 ; # 615
PLANT.PIPEC2.PSI(34) = 2.62522e+006 ; # 616
PLANT.PIPEC2.PSI(35) = 2.62186e+006 ; # 617
PLANT.PIPEC2.PSI(36) = 2.6181e+006 ; # 618
PLANT.PIPEC2.PSI(37) = 2.61397e+006 ; # 619
PLANT.PIPEC2.PSI(38) = 2.6095e+006 ; # 620
PLANT.PIPEC2.PSI(39) = 2.60471e+006 ; # 621
PLANT.PIPEC2.PSI(40) = 2.59962e+006 ; # 622
```

HTLQSS_SS_IN_PRE.PRESETS

```
# Values for computation HTLFULLDYN_SS
# Saved at time 1000
PLANT.HX.TGASHOT(1) := 914.423 ; # 1
PLANT.HX.TGASHOT(2) := 924.159 ; # 2
PLANT.HX.TGASHOT(3) := 933.692 ; # 3
PLANT.HX.TGASHOT(4) := 943.011 ; # 4
PLANT.HX.TGASHOT(5) := 952.118 ; # 5
PLANT.HX.TGASHOT(6) := 961.02 ; # 6
PLANT.HX.TGASHOT(7) := 969.722 ; # 7
PLANT.HX.TGASHOT(8) := 978.227 ; # 8
PLANT.HX.TGASHOT(9) := 986.539 ; # 9
PLANT.HX.TGASHOT(10) := 994.665 ; # 10
PLANT.HX.TGASHOT(11) := 1002.61 ; # 11
PLANT.HX.TGASHOT(12) := 1010.37 ; # 12
PLANT.HX.TGASHOT(13) := 1017.96 ; # 13
PLANT.HX.TGASHOT(14) := 1025.37 ; # 14
PLANT.HX.TGASHOT(15) := 1032.62 ; # 15
PLANT.HX.TGASHOT(16) := 1039.71 ; # 16
PLANT.HX.TGASHOT(17) := 1046.63 ; # 17
PLANT.HX.TGASHOT(18) := 1053.4 ; # 18
PLANT.HX.TGASHOT(19) := 1060.02 ; # 19
PLANT.HX.TGASHOT(20) := 1066.48 ; # 20
PLANT.HX.TGASHOT(21) := 1072.8 ; # 21
PLANT.HX.TGASHOT(22) := 1078.98 ; # 22
PLANT.HX.TGASHOT(23) := 1085.02 ; # 23
PLANT.HX.TGASHOT(24) := 1090.92 ; # 24
PLANT.HX.TGASHOT(25) := 1096.69 ; # 25
PLANT.HX.TGASHOT(26) := 1102.33 ; # 26
PLANT.HX.TGASHOT(27) := 1107.84 ; # 27
PLANT.HX.TGASHOT(28) := 1113.23 ; # 28
PLANT.HX.TGASHOT(29) := 1118.5 ; # 29
PLANT.HX.TGASHOT(30) := 1123.64 ; # 30
PLANT.HX.TGASHOT(31) := 1128.67 ; # 31
PLANT.HX.TGASHOT(32) := 1133.59 ; # 32
PLANT.HX.TGASHOT(33) := 1138.4 ; # 33
PLANT.HX.TGASHOT(34) := 1143.1 ; # 34
PLANT.HX.TGASHOT(35) := 1147.69 ; # 35
PLANT.HX.TGASHOT(36) := 1152.18 ; # 36
PLANT.HX.TGASHOT(37) := 1156.56 ; # 37
PLANT.HX.TGASHOT(38) := 1160.85 ; # 38
PLANT.HX.TGASHOT(39) := 1165.04 ; # 39
PLANT.HX.TGASHOT(40) := 1169.14 ; # 40
PLANT.HX.TGASCOLD(1) := 863.757 ; # 41
PLANT.HX.TGASCOLD(2) := 874.654 ; # 42
PLANT.HX.TGASCOLD(3) := 885.305 ; # 43
PLANT.HX.TGASCOLD(4) := 895.715 ; # 44
PLANT.HX.TGASCOLD(5) := 905.89 ; # 45
PLANT.HX.TGASCOLD(6) := 915.836 ; # 46
PLANT.HX.TGASCOLD(7) := 925.557 ; # 47
PLANT.HX.TGASCOLD(8) := 935.058 ; # 48
PLANT.HX.TGASCOLD(9) := 944.346 ; # 49
PLANT.HX.TGASCOLD(10) := 953.423 ; # 50
PLANT.HX.TGASCOLD(11) := 962.296 ; # 51
PLANT.HX.TGASCOLD(12) := 970.968 ; # 52
PLANT.HX.TGASCOLD(13) := 979.445 ; # 53
PLANT.HX.TGASCOLD(14) := 987.731 ; # 54
PLANT.HX.TGASCOLD(15) := 995.829 ; # 55
PLANT.HX.TGASCOLD(16) := 1003.74 ; # 56
```

```
PLANT.HX.TGASCOLD(17) := 1011.48 ; # 57
PLANT.HX.TGASCOLD(18) := 1019.04 ; # 58
PLANT.HX.TGASCOLD(19) := 1026.44 ; # 59
PLANT.HX.TGASCOLD(20) := 1033.66 ; # 60
PLANT.HX.TGASCOLD(21) := 1040.72 ; # 61
PLANT.HX.TGASCOLD(22) := 1047.62 ; # 62
PLANT.HX.TGASCOLD(23) := 1054.37 ; # 63
PLANT.HX.TGASCOLD(24) := 1060.96 ; # 64
PLANT.HX.TGASCOLD(25) := 1067.41 ; # 65
PLANT.HX.TGASCOLD(26) := 1073.71 ; # 66
PLANT.HX.TGASCOLD(27) := 1079.87 ; # 67
PLANT.HX.TGASCOLD(28) := 1085.89 ; # 68
PLANT.HX.TGASCOLD(29) := 1091.77 ; # 69
PLANT.HX.TGASCOLD(30) := 1097.52 ; # 70
PLANT.HX.TGASCOLD(31) := 1103.14 ; # 71
PLANT.HX.TGASCOLD(32) := 1108.63 ; # 72
PLANT.HX.TGASCOLD(33) := 1114 ; # 73
PLANT.HX.TGASCOLD(34) := 1119.25 ; # 74
PLANT.HX.TGASCOLD(35) := 1124.38 ; # 75
PLANT.HX.TGASCOLD(36) := 1129.4 ; # 76
PLANT.HX.TGASCOLD(37) := 1134.3 ; # 77
PLANT.HX.TGASCOLD(38) := 1139.09 ; # 78
PLANT.HX.TGASCOLD(39) := 1143.77 ; # 79
PLANT.HX.TGASCOLD(40) := 1148.34 ; # 80
PLANT.HX.UHOT := 1738.41 ; # 81
PLANT.HX.UCOLD := 1645.45 ; # 82
PLANT.HX.THX(1) := 889.835 ; # 83
PLANT.HX.THX(2) := 900.086 ; # 84
PLANT.HX.THX(3) := 910.162 ; # 85
PLANT.HX.THX(4) := 920.011 ; # 86
PLANT.HX.THX(5) := 929.638 ; # 87
PLANT.HX.THX(6) := 939.048 ; # 88
PLANT.HX.THX(7) := 948.245 ; # 89
PLANT.HX.THX(8) := 957.235 ; # 90
PLANT.HX.THX(9) := 966.021 ; # 91
PLANT.HX.THX(10) := 974.61 ; # 92
PLANT.HX.THX(11) := 983.004 ; # 93
PLANT.HX.THX(12) := 991.209 ; # 94
PLANT.HX.THX(13) := 999.229 ; # 95
PLANT.HX.THX(14) := 1007.07 ; # 96
PLANT.HX.THX(15) := 1014.73 ; # 97
PLANT.HX.THX(16) := 1022.22 ; # 98
PLANT.HX.THX(17) := 1029.54 ; # 99
PLANT.HX.THX(18) := 1036.69 ; # 100
PLANT.HX.THX(19) := 1043.69 ; # 101
PLANT.HX.THX(20) := 1050.52 ; # 102
PLANT.HX.THX(21) := 1057.2 ; # 103
PLANT.HX.THX(22) := 1063.73 ; # 104
PLANT.HX.THX(23) := 1070.12 ; # 105
PLANT.HX.THX(24) := 1076.36 ; # 106
PLANT.HX.THX(25) := 1082.45 ; # 107
PLANT.HX.THX(26) := 1088.41 ; # 108
PLANT.HX.THX(27) := 1094.24 ; # 109
PLANT.HX.THX(28) := 1099.93 ; # 110
PLANT.HX.THX(29) := 1105.5 ; # 111
PLANT.HX.THX(30) := 1110.94 ; # 112
PLANT.HX.THX(31) := 1116.26 ; # 113
PLANT.HX.THX(32) := 1121.45 ; # 114
PLANT.HX.THX(33) := 1126.53 ; # 115
PLANT.HX.THX(34) := 1131.5 ; # 116
PLANT.HX.THX(35) := 1136.35 ; # 117
PLANT.HX.THX(36) := 1141.1 ; # 118
PLANT.HX.THX(37) := 1145.73 ; # 119
PLANT.HX.THX(38) := 1150.27 ; # 120
PLANT.HX.THX(39) := 1154.7 ; # 121
PLANT.HX.THX(40) := 1159 ; # 122
PLANT.LOOP1.RHOEXTRA(1) := 1.14734 ; # 123
PLANT.LOOP1.RHOEXTRA(2) := 1.14734 ; # 123
PLANT.LOOP1.RHO(1) := 1.14734 ; # 123
PLANT.LOOP1.RHO(2) := 1.14286 ; # 124
PLANT.LOOP1.RHO(3) := 1.14003 ; # 125
PLANT.LOOP1.RHO(4) := 1.1372 ; # 126
PLANT.LOOP1.RHO(5) := 1.13435 ; # 127
PLANT.LOOP1.RHO(6) := 1.1315 ; # 128
PLANT.LOOP1.RHO(7) := 1.12862 ; # 129
PLANT.LOOP1.RHO(8) := 1.12574 ; # 130
PLANT.LOOP1.RHO(9) := 1.12285 ; # 131
PLANT.LOOP1.RHO(10) := 1.11994 ; # 132
PLANT.LOOP1.RHO(11) := 1.1156 ; # 133
PLANT.LOOP1.RHO(12) := 1.1004 ; # 134
PLANT.LOOP1.RHO(13) := 1.08551 ; # 135
PLANT.LOOP1.RHO(14) := 1.07124 ; # 136
PLANT.LOOP1.RHO(15) := 1.05755 ; # 137
PLANT.LOOP1.RHO(16) := 1.0444 ; # 138
PLANT.LOOP1.RHO(17) := 1.03177 ; # 139
PLANT.LOOP1.RHO(18) := 1.01963 ; # 140
PLANT.LOOP1.RHO(19) := 1.00793 ; # 141
PLANT.LOOP1.RHO(20) := 0.996671 ; # 142
PLANT.LOOP1.RHO(21) := 0.985813 ; # 143
PLANT.LOOP1.RHO(22) := 0.975337 ; # 144
PLANT.LOOP1.RHO(23) := 0.965222 ; # 145
PLANT.LOOP1.RHO(24) := 0.95545 ; # 146
```

```
PLANT.LOOP1.RHO(25) := 0.946002 ; # 147
PLANT.LOOP1.RHO(26) := 0.936861 ; # 148
PLANT.LOOP1.RHO(27) := 0.928012 ; # 149
PLANT.LOOP1.RHO(28) := 0.91944 ; # 150
PLANT.LOOP1.RHO(29) := 0.91113 ; # 151
PLANT.LOOP1.RHO(30) := 0.903071 ; # 152
PLANT.LOOP1.RHO(31) := 0.89525 ; # 153
PLANT.LOOP1.RHO(32) := 0.887655 ; # 154
PLANT.LOOP1.RHO(33) := 0.880276 ; # 155
PLANT.LOOP1.RHO(34) := 0.873102 ; # 156
PLANT.LOOP1.RHO(35) := 0.866125 ; # 157
PLANT.LOOP1.RHO(36) := 0.859335 ; # 158
PLANT.LOOP1.RHO(37) := 0.852723 ; # 159
PLANT.LOOP1.RHO(38) := 0.846282 ; # 160
PLANT.LOOP1.RHO(39) := 0.840005 ; # 161
PLANT.LOOP1.RHO(40) := 0.833883 ; # 162
PLANT.LOOP1.RHO(41) := 0.82791 ; # 163
PLANT.LOOP1.RHO(42) := 0.82208 ; # 164
PLANT.LOOP1.RHO(43) := 0.816387 ; # 165
PLANT.LOOP1.RHO(44) := 0.810825 ; # 166
PLANT.LOOP1.RHO(45) := 0.805388 ; # 167
PLANT.LOOP1.RHO(46) := 0.800071 ; # 168
PLANT.LOOP1.RHO(47) := 0.794869 ; # 169
PLANT.LOOP1.RHO(48) := 0.789778 ; # 170
PLANT.LOOP1.RHO(49) := 0.784793 ; # 171
PLANT.LOOP1.RHO(50) := 0.779915 ; # 172
PLANT.LOOP1.RHO(51) := 0.767418 ; # 173
PLANT.LOOP1.RHO(52) := 0.765306 ; # 174
PLANT.LOOP1.RHO(53) := 0.763202 ; # 175
PLANT.LOOP1.RHO(54) := 0.761087 ; # 176
PLANT.LOOP1.RHO(55) := 0.758961 ; # 177
PLANT.LOOP1.RHO(56) := 0.756824 ; # 178
PLANT.LOOP1.RHO(57) := 0.754676 ; # 179
PLANT.LOOP1.RHO(58) := 0.752516 ; # 180
PLANT.LOOP1.RHO(59) := 0.750345 ; # 181
PLANT.LOOP1.RHO(60) := 0.748162 ; # 182
PLANT.LOOP1.RHO(61) := 0.758154 ; # 183
PLANT.LOOP1.RHO(62) := 0.758842 ; # 184
PLANT.LOOP1.RHO(63) := 0.759576 ; # 185
PLANT.LOOP1.RHO(64) := 0.760408 ; # 186
PLANT.LOOP1.RHO(65) := 0.761346 ; # 187
PLANT.LOOP1.RHO(66) := 0.762396 ; # 188
PLANT.LOOP1.RHO(67) := 0.763568 ; # 189
PLANT.LOOP1.RHO(68) := 0.76487 ; # 190
PLANT.LOOP1.RHO(69) := 0.766313 ; # 191
PLANT.LOOP1.RHO(70) := 0.767908 ; # 192
PLANT.LOOP1.RHO(71) := 0.769667 ; # 193
PLANT.LOOP1.RHO(72) := 0.771602 ; # 194
PLANT.LOOP1.RHO(73) := 0.773727 ; # 195
PLANT.LOOP1.RHO(74) := 0.776059 ; # 196
PLANT.LOOP1.RHO(75) := 0.778613 ; # 197
PLANT.LOOP1.RHO(76) := 0.781409 ; # 198
PLANT.LOOP1.RHO(77) := 0.784467 ; # 199
PLANT.LOOP1.RHO(78) := 0.787809 ; # 200
PLANT.LOOP1.RHO(79) := 0.79146 ; # 201
PLANT.LOOP1.RHO(80) := 0.795447 ; # 202
PLANT.LOOP1.RHO(81) := 0.799801 ; # 203
PLANT.LOOP1.RHO(82) := 0.804555 ; # 204
PLANT.LOOP1.RHO(83) := 0.809747 ; # 205
PLANT.LOOP1.RHO(84) := 0.815419 ; # 206
PLANT.LOOP1.RHO(85) := 0.821618 ; # 207
PLANT.LOOP1.RHO(86) := 0.828396 ; # 208
PLANT.LOOP1.RHO(87) := 0.835814 ; # 209
PLANT.LOOP1.RHO(88) := 0.843939 ; # 210
PLANT.LOOP1.RHO(89) := 0.852847 ; # 211
PLANT.LOOP1.RHO(90) := 0.862626 ; # 212
PLANT.LOOP1.RHO(91) := 0.873377 ; # 213
PLANT.LOOP1.RHO(92) := 0.885216 ; # 214
PLANT.LOOP1.RHO(93) := 0.898277 ; # 215
PLANT.LOOP1.RHO(94) := 0.912717 ; # 216
PLANT.LOOP1.RHO(95) := 0.928718 ; # 217
PLANT.LOOP1.RHO(96) := 0.946498 ; # 218
PLANT.LOOP1.RHO(97) := 0.966314 ; # 219
PLANT.LOOP1.RHO(98) := 0.988474 ; # 220
PLANT.LOOP1.RHO(99) := 1.01335 ; # 221
PLANT.LOOP1.RHO(100) := 1.0414 ; # 222
PLANT.LOOP1.PEXTRA(1) := 2.04197e+006 ; # 223
PLANT.LOOP1.PEXTRA(2) := 2.04197e+006 ; # 223
PLANT.LOOP1.P(1) := 2.04197e+006 ; # 223
PLANT.LOOP1.P(2) := 2.03301e+006 ; # 224
PLANT.LOOP1.P(3) := 2.02702e+006 ; # 225
PLANT.LOOP1.P(4) := 2.02101e+006 ; # 226
PLANT.LOOP1.P(5) := 2.01499e+006 ; # 227
PLANT.LOOP1.P(6) := 2.00895e+006 ; # 228
PLANT.LOOP1.P(7) := 2.0029e+006 ; # 229
PLANT.LOOP1.P(8) := 1.99683e+006 ; # 230
PLANT.LOOP1.P(9) := 1.99074e+006 ; # 231
PLANT.LOOP1.P(10) := 1.98464e+006 ; # 232
PLANT.LOOP1.P(11) := 2.00285e+006 ; # 233
PLANT.LOOP1.P(12) := 2.00049e+006 ; # 234
PLANT.LOOP1.P(13) := 1.99745e+006 ; # 235
PLANT.LOOP1.P(14) := 1.99437e+006 ; # 236
```

```
PLANT.LOOP1.P(15) := 1.99125e+006 ; # 237
PLANT.LOOP1.P(16) := 1.98809e+006 ; # 238
PLANT.LOOP1.P(17) := 1.9849e+006 ; # 239
PLANT.LOOP1.P(18) := 1.98166e+006 ; # 240
PLANT.LOOP1.P(19) := 1.9784e+006 ; # 241
PLANT.LOOP1.P(20) := 1.97509e+006 ; # 242
PLANT.LOOP1.P(21) := 1.97176e+006 ; # 243
PLANT.LOOP1.P(22) := 1.96838e+006 ; # 244
PLANT.LOOP1.P(23) := 1.96498e+006 ; # 245
PLANT.LOOP1.P(24) := 1.96154e+006 ; # 246
PLANT.LOOP1.P(25) := 1.95806e+006 ; # 247
PLANT.LOOP1.P(26) := 1.95456e+006 ; # 248
PLANT.LOOP1.P(27) := 1.95102e+006 ; # 249
PLANT.LOOP1.P(28) := 1.94745e+006 ; # 250
PLANT.LOOP1.P(29) := 1.94385e+006 ; # 251
PLANT.LOOP1.P(30) := 1.94021e+006 ; # 252
PLANT.LOOP1.P(31) := 1.93655e+006 ; # 253
PLANT.LOOP1.P(32) := 1.93286e+006 ; # 254
PLANT.LOOP1.P(33) := 1.92913e+006 ; # 255
PLANT.LOOP1.P(34) := 1.92538e+006 ; # 256
PLANT.LOOP1.P(35) := 1.9216e+006 ; # 257
PLANT.LOOP1.P(36) := 1.91778e+006 ; # 258
PLANT.LOOP1.P(37) := 1.91394e+006 ; # 259
PLANT.LOOP1.P(38) := 1.91007e+006 ; # 260
PLANT.LOOP1.P(39) := 1.90617e+006 ; # 261
PLANT.LOOP1.P(40) := 1.90225e+006 ; # 262
PLANT.LOOP1.P(41) := 1.8983e+006 ; # 263
PLANT.LOOP1.P(42) := 1.89432e+006 ; # 264
PLANT.LOOP1.P(43) := 1.89031e+006 ; # 265
PLANT.LOOP1.P(44) := 1.88627e+006 ; # 266
PLANT.LOOP1.P(45) := 1.88221e+006 ; # 267
PLANT.LOOP1.P(46) := 1.87813e+006 ; # 268
PLANT.LOOP1.P(47) := 1.87401e+006 ; # 269
PLANT.LOOP1.P(48) := 1.86987e+006 ; # 270
PLANT.LOOP1.P(49) := 1.86571e+006 ; # 271
PLANT.LOOP1.P(50) := 1.86152e+006 ; # 272
PLANT.LOOP1.P(51) := 1.83045e+006 ; # 273
PLANT.LOOP1.P(52) := 1.82419e+006 ; # 274
PLANT.LOOP1.P(53) := 1.81794e+006 ; # 275
PLANT.LOOP1.P(54) := 1.81169e+006 ; # 276
PLANT.LOOP1.P(55) := 1.80541e+006 ; # 277
PLANT.LOOP1.P(56) := 1.79911e+006 ; # 278
PLANT.LOOP1.P(57) := 1.7928e+006 ; # 279
PLANT.LOOP1.P(58) := 1.78646e+006 ; # 280
PLANT.LOOP1.P(59) := 1.78011e+006 ; # 281
PLANT.LOOP1.P(60) := 1.77374e+006 ; # 282
PLANT.LOOP1.P(61) := 1.7945e+006 ; # 283
PLANT.LOOP1.P(62) := 1.793e+006 ; # 284
PLANT.LOOP1.P(63) := 1.7914e+006 ; # 285
PLANT.LOOP1.P(64) := 1.78981e+006 ; # 286
PLANT.LOOP1.P(65) := 1.78821e+006 ; # 287
PLANT.LOOP1.P(66) := 1.78662e+006 ; # 288
PLANT.LOOP1.P(67) := 1.78503e+006 ; # 289
PLANT.LOOP1.P(68) := 1.78345e+006 ; # 290
PLANT.LOOP1.P(69) := 1.78187e+006 ; # 291
PLANT.LOOP1.P(70) := 1.78029e+006 ; # 292
PLANT.LOOP1.P(71) := 1.77872e+006 ; # 293
PLANT.LOOP1.P(72) := 1.77715e+006 ; # 294
PLANT.LOOP1.P(73) := 1.77559e+006 ; # 295
PLANT.LOOP1.P(74) := 1.77404e+006 ; # 296
PLANT.LOOP1.P(75) := 1.77249e+006 ; # 297
PLANT.LOOP1.P(76) := 1.77095e+006 ; # 298
PLANT.LOOP1.P(77) := 1.76942e+006 ; # 299
PLANT.LOOP1.P(78) := 1.76789e+006 ; # 300
PLANT.LOOP1.P(79) := 1.76638e+006 ; # 301
PLANT.LOOP1.P(80) := 1.76487e+006 ; # 302
PLANT.LOOP1.P(81) := 1.76337e+006 ; # 303
PLANT.LOOP1.P(82) := 1.76189e+006 ; # 304
PLANT.LOOP1.P(83) := 1.76042e+006 ; # 305
PLANT.LOOP1.P(84) := 1.75896e+006 ; # 306
PLANT.LOOP1.P(85) := 1.75751e+006 ; # 307
PLANT.LOOP1.P(86) := 1.75608e+006 ; # 308
PLANT.LOOP1.P(87) := 1.75467e+006 ; # 309
PLANT.LOOP1.P(88) := 1.75327e+006 ; # 310
PLANT.LOOP1.P(89) := 1.75189e+006 ; # 311
PLANT.LOOP1.P(90) := 1.75054e+006 ; # 312
PLANT.LOOP1.P(91) := 1.7492e+006 ; # 313
PLANT.LOOP1.P(92) := 1.74789e+006 ; # 314
PLANT.LOOP1.P(93) := 1.7466e+006 ; # 315
PLANT.LOOP1.P(94) := 1.74534e+006 ; # 316
PLANT.LOOP1.P(95) := 1.7441e+006 ; # 317
PLANT.LOOP1.P(96) := 1.7429e+006 ; # 318
PLANT.LOOP1.P(97) := 1.74173e+006 ; # 319
PLANT.LOOP1.P(98) := 1.74059e+006 ; # 320
PLANT.LOOP1.P(99) := 1.73949e+006 ; # 321
PLANT.LOOP1.P(100) := 1.73843e+006 ; # 322
PLANT.LOOP1.P_OUT := 1.73785e+006 ; # 323
PLANT.LOOP1.HEXTRA(1) := 2.90016e+006 ; # 324
PLANT.LOOP1.HEXTRA(2) := 2.90016e+006 ; # 324
PLANT.LOOP1.H(1) := 2.90016e+006 ; # 324
PLANT.LOOP1.H(2) := 2.89803e+006 ; # 325
PLANT.LOOP1.H(3) := 2.8959e+006 ; # 326
```

```
PLANT.LOOP1.H(4) := 2.89378e+006 ; # 327
PLANT.LOOP1.H(5) := 2.89166e+006 ; # 328
PLANT.LOOP1.H(6) := 2.88954e+006 ; # 329
PLANT.LOOP1.H(7) := 2.88742e+006 ; # 330
PLANT.LOOP1.H(8) := 2.8853e+006 ; # 331
PLANT.LOOP1.H(9) := 2.88318e+006 ; # 332
PLANT.LOOP1.H(10) := 2.88106e+006 ; # 333
PLANT.LOOP1.H(11) := 2.93911e+006 ; # 334
PLANT.LOOP1.H(12) := 2.99572e+006 ; # 335
PLANT.LOOP1.H(13) := 3.05105e+006 ; # 336
PLANT.LOOP1.H(14) := 3.10513e+006 ; # 337
PLANT.LOOP1.H(15) := 3.15799e+006 ; # 338
PLANT.LOOP1.H(16) := 3.20966e+006 ; # 339
PLANT.LOOP1.H(17) := 3.26016e+006 ; # 340
PLANT.LOOP1.H(18) := 3.30952e+006 ; # 341
PLANT.LOOP1.H(19) := 3.35777e+006 ; # 342
PLANT.LOOP1.H(20) := 3.40492e+006 ; # 343
PLANT.LOOP1.H(21) := 3.45102e+006 ; # 344
PLANT.LOOP1.H(22) := 3.49607e+006 ; # 345
PLANT.LOOP1.H(23) := 3.54011e+006 ; # 346
PLANT.LOOP1.H(24) := 3.58315e+006 ; # 347
PLANT.LOOP1.H(25) := 3.62522e+006 ; # 348
PLANT.LOOP1.H(26) := 3.66634e+006 ; # 349
PLANT.LOOP1.H(27) := 3.70654e+006 ; # 350
PLANT.LOOP1.H(28) := 3.74582e+006 ; # 351
PLANT.LOOP1.H(29) := 3.78422e+006 ; # 352
PLANT.LOOP1.H(30) := 3.82175e+006 ; # 353
PLANT.LOOP1.H(31) := 3.85844e+006 ; # 354
PLANT.LOOP1.H(32) := 3.8943e+006 ; # 355
PLANT.LOOP1.H(33) := 3.92935e+006 ; # 356
PLANT.LOOP1.H(34) := 3.9636e+006 ; # 357
PLANT.LOOP1.H(35) := 3.99709e+006 ; # 358
PLANT.LOOP1.H(36) := 4.02981e+006 ; # 359
PLANT.LOOP1.H(37) := 4.0618e+006 ; # 360
PLANT.LOOP1.H(38) := 4.09307e+006 ; # 361
PLANT.LOOP1.H(39) := 4.12363e+006 ; # 362
PLANT.LOOP1.H(40) := 4.15351e+006 ; # 363
PLANT.LOOP1.H(41) := 4.1827e+006 ; # 364
PLANT.LOOP1.H(42) := 4.21124e+006 ; # 365
PLANT.LOOP1.H(43) := 4.23914e+006 ; # 366
PLANT.LOOP1.H(44) := 4.2664e+006 ; # 367
PLANT.LOOP1.H(45) := 4.29305e+006 ; # 368
PLANT.LOOP1.H(46) := 4.3191e+006 ; # 369
PLANT.LOOP1.H(47) := 4.34456e+006 ; # 370
PLANT.LOOP1.H(48) := 4.36944e+006 ; # 371
PLANT.LOOP1.H(49) := 4.39377e+006 ; # 372
PLANT.LOOP1.H(50) := 4.41751e+006 ; # 373
PLANT.LOOP1.H(51) := 4.41348e+006 ; # 374
PLANT.LOOP1.H(52) := 4.40946e+006 ; # 375
PLANT.LOOP1.H(53) := 4.40545e+006 ; # 376
PLANT.LOOP1.H(54) := 4.40144e+006 ; # 377
PLANT.LOOP1.H(55) := 4.39743e+006 ; # 378
PLANT.LOOP1.H(56) := 4.39343e+006 ; # 379
PLANT.LOOP1.H(57) := 4.38943e+006 ; # 380
PLANT.LOOP1.H(58) := 4.38543e+006 ; # 381
PLANT.LOOP1.H(59) := 4.38144e+006 ; # 382
PLANT.LOOP1.H(60) := 4.37745e+006 ; # 383
PLANT.LOOP1.H(61) := 4.3678e+006 ; # 384
PLANT.LOOP1.H(62) := 4.35751e+006 ; # 385
PLANT.LOOP1.H(63) := 4.34653e+006 ; # 386
PLANT.LOOP1.H(64) := 4.33483e+006 ; # 387
PLANT.LOOP1.H(65) := 4.32236e+006 ; # 388
PLANT.LOOP1.H(66) := 4.30905e+006 ; # 389
PLANT.LOOP1.H(67) := 4.29487e+006 ; # 390
PLANT.LOOP1.H(68) := 4.27974e+006 ; # 391
PLANT.LOOP1.H(69) := 4.26361e+006 ; # 392
PLANT.LOOP1.H(70) := 4.24641e+006 ; # 393
PLANT.LOOP1.H(71) := 4.22806e+006 ; # 394
PLANT.LOOP1.H(72) := 4.20851e+006 ; # 395
PLANT.LOOP1.H(73) := 4.18765e+006 ; # 396
PLANT.LOOP1.H(74) := 4.16541e+006 ; # 397
PLANT.LOOP1.H(75) := 4.1417e+006 ; # 398
PLANT.LOOP1.H(76) := 4.11641e+006 ; # 399
PLANT.LOOP1.H(77) := 4.08945e+006 ; # 400
PLANT.LOOP1.H(78) := 4.06069e+006 ; # 401
PLANT.LOOP1.H(79) := 4.03003e+006 ; # 402
PLANT.LOOP1.H(80) := 3.99734e+006 ; # 403
PLANT.LOOP1.H(81) := 3.96248e+006 ; # 404
PLANT.LOOP1.H(82) := 3.9253e+006 ; # 405
PLANT.LOOP1.H(83) := 3.88566e+006 ; # 406
PLANT.LOOP1.H(84) := 3.8434e+006 ; # 407
PLANT.LOOP1.H(85) := 3.79832e+006 ; # 408
PLANT.LOOP1.H(86) := 3.75026e+006 ; # 409
PLANT.LOOP1.H(87) := 3.69901e+006 ; # 410
PLANT.LOOP1.H(88) := 3.64436e+006 ; # 411
PLANT.LOOP1.H(89) := 3.58609e+006 ; # 412
PLANT.LOOP1.H(90) := 3.52395e+006 ; # 413
PLANT.LOOP1.H(91) := 3.45769e+006 ; # 414
PLANT.LOOP1.H(92) := 3.38703e+006 ; # 415
PLANT.LOOP1.H(93) := 3.31169e+006 ; # 416
PLANT.LOOP1.H(94) := 3.23135e+006 ; # 417
PLANT.LOOP1.H(95) := 3.14568e+006 ; # 418
```

```
427   PLANT.LOOP1.H(96) := 3.05433e+006 ; # 419
428   PLANT.LOOP1.H(97) := 2.95692e+006 ; # 420
429   PLANT.LOOP1.H(98) := 2.85305e+006 ; # 421
430   PLANT.LOOP1.H(99) := 2.74229e+006 ; # 422
431   PLANT.LOOP1.H(100) := 2.62419e+006 ; # 423
432   PLANT.LOOP1.H0 := 2.90229e+006 ; # 424
433   PLANT.LOOP1.TEXTRA(1) := 856.26 ; # 425
434   PLANT.LOOP1.TEXTRA(2) := 856.26 ; # 425
435   PLANT.LOOP1.T(1) := 856.26 ; # 425
436   PLANT.LOOP1.T(2) := 855.85 ; # 426
437   PLANT.LOOP1.T(3) := 855.441 ; # 427
438   PLANT.LOOP1.T(4) := 855.032 ; # 428
439   PLANT.LOOP1.T(5) := 854.623 ; # 429
440   PLANT.LOOP1.T(6) := 854.215 ; # 430
441   PLANT.LOOP1.T(7) := 853.807 ; # 431
442   PLANT.LOOP1.T(8) := 853.399 ; # 432
443   PLANT.LOOP1.T(9) := 852.991 ; # 433
444   PLANT.LOOP1.T(10) := 852.584 ; # 434
445   PLANT.LOOP1.T(11) := 863.757 ; # 435
446   PLANT.LOOP1.T(12) := 874.654 ; # 436
447   PLANT.LOOP1.T(13) := 885.305 ; # 437
448   PLANT.LOOP1.T(14) := 895.715 ; # 438
449   PLANT.LOOP1.T(15) := 905.89 ; # 439
450   PLANT.LOOP1.T(16) := 915.836 ; # 440
451   PLANT.LOOP1.T(17) := 925.557 ; # 441
452   PLANT.LOOP1.T(18) := 935.058 ; # 442
453   PLANT.LOOP1.T(19) := 944.346 ; # 443
454   PLANT.LOOP1.T(20) := 953.423 ; # 444
455   PLANT.LOOP1.T(21) := 962.296 ; # 445
456   PLANT.LOOP1.T(22) := 970.968 ; # 446
457   PLANT.LOOP1.T(23) := 979.445 ; # 447
458   PLANT.LOOP1.T(24) := 987.731 ; # 448
459   PLANT.LOOP1.T(25) := 995.829 ; # 449
460   PLANT.LOOP1.T(26) := 1003.74 ; # 450
461   PLANT.LOOP1.T(27) := 1011.48 ; # 451
462   PLANT.LOOP1.T(28) := 1019.04 ; # 452
463   PLANT.LOOP1.T(29) := 1026.44 ; # 453
464   PLANT.LOOP1.T(30) := 1033.66 ; # 454
465   PLANT.LOOP1.T(31) := 1040.72 ; # 455
466   PLANT.LOOP1.T(32) := 1047.62 ; # 456
467   PLANT.LOOP1.T(33) := 1054.37 ; # 457
468   PLANT.LOOP1.T(34) := 1060.96 ; # 458
469   PLANT.LOOP1.T(35) := 1067.41 ; # 459
470   PLANT.LOOP1.T(36) := 1073.71 ; # 460
471   PLANT.LOOP1.T(37) := 1079.87 ; # 461
472   PLANT.LOOP1.T(38) := 1085.89 ; # 462
473   PLANT.LOOP1.T(39) := 1091.77 ; # 463
474   PLANT.LOOP1.T(40) := 1097.52 ; # 464
475   PLANT.LOOP1.T(41) := 1103.14 ; # 465
476   PLANT.LOOP1.T(42) := 1108.63 ; # 466
477   PLANT.LOOP1.T(43) := 1114 ; # 467
478   PLANT.LOOP1.T(44) := 1119.25 ; # 468
479   PLANT.LOOP1.T(45) := 1124.38 ; # 469
480   PLANT.LOOP1.T(46) := 1129.4 ; # 470
481   PLANT.LOOP1.T(47) := 1134.3 ; # 471
482   PLANT.LOOP1.T(48) := 1139.09 ; # 472
483   PLANT.LOOP1.T(49) := 1143.77 ; # 473
484   PLANT.LOOP1.T(50) := 1148.34 ; # 474
485   PLANT.LOOP1.T(51) := 1147.56 ; # 475
486   PLANT.LOOP1.T(52) := 1146.79 ; # 476
487   PLANT.LOOP1.T(53) := 1146.02 ; # 477
488   PLANT.LOOP1.T(54) := 1145.25 ; # 478
489   PLANT.LOOP1.T(55) := 1144.47 ; # 479
490   PLANT.LOOP1.T(56) := 1143.7 ; # 480
491   PLANT.LOOP1.T(57) := 1142.93 ; # 481
492   PLANT.LOOP1.T(58) := 1142.16 ; # 482
493   PLANT.LOOP1.T(59) := 1141.39 ; # 483
494   PLANT.LOOP1.T(60) := 1140.63 ; # 484
495   PLANT.LOOP1.T(61) := 1138.77 ; # 485
496   PLANT.LOOP1.T(62) := 1136.79 ; # 486
497   PLANT.LOOP1.T(63) := 1134.68 ; # 487
498   PLANT.LOOP1.T(64) := 1132.42 ; # 488
499   PLANT.LOOP1.T(65) := 1130.02 ; # 489
500   PLANT.LOOP1.T(66) := 1127.46 ; # 490
501   PLANT.LOOP1.T(67) := 1124.73 ; # 491
502   PLANT.LOOP1.T(68) := 1121.82 ; # 492
503   PLANT.LOOP1.T(69) := 1118.71 ; # 493
504   PLANT.LOOP1.T(70) := 1115.4 ; # 494
505   PLANT.LOOP1.T(71) := 1111.87 ; # 495
506   PLANT.LOOP1.T(72) := 1108.11 ; # 496
507   PLANT.LOOP1.T(73) := 1104.09 ; # 497
508   PLANT.LOOP1.T(74) := 1099.81 ; # 498
509   PLANT.LOOP1.T(75) := 1095.25 ; # 499
510   PLANT.LOOP1.T(76) := 1090.38 ; # 500
511   PLANT.LOOP1.T(77) := 1085.19 ; # 501
512   PLANT.LOOP1.T(78) := 1079.65 ; # 502
513   PLANT.LOOP1.T(79) := 1073.75 ; # 503
514   PLANT.LOOP1.T(80) := 1067.46 ; # 504
515   PLANT.LOOP1.T(81) := 1060.75 ; # 505
516   PLANT.LOOP1.T(82) := 1053.59 ; # 506
517   PLANT.LOOP1.T(83) := 1045.96 ; # 507
518   PLANT.LOOP1.T(84) := 1037.83 ; # 508
```

```
PLANT.LOOP1.T(85) := 1029.15 ; # 509
PLANT.LOOP1.T(86) := 1019.9 ; # 510
PLANT.LOOP1.T(87) := 1010.03 ; # 511
PLANT.LOOP1.T(88) := 999.513 ; # 512
PLANT.LOOP1.T(89) := 988.296 ; # 513
PLANT.LOOP1.T(90) := 976.334 ; # 514
PLANT.LOOP1.T(91) := 963.58 ; # 515
PLANT.LOOP1.T(92) := 949.979 ; # 516
PLANT.LOOP1.T(93) := 935.476 ; # 517
PLANT.LOOP1.T(94) := 920.011 ; # 518
PLANT.LOOP1.T(95) := 903.52 ; # 519
PLANT.LOOP1.T(96) := 885.936 ; # 520
PLANT.LOOP1.T(97) := 867.185 ; # 521
PLANT.LOOP1.T(98) := 847.191 ; # 522
PLANT.LOOP1.T(99) := 825.87 ; # 523
PLANT.LOOP1.T(100) := 803.137 ; # 524
PLANT.LOOP1.T0 := 856.67 ; # 525
PLANT.LOOP1.TEXTRNL(1) := 291.15 ; # 526
PLANT.LOOP1.TEXTRNL(2) := 291.15 ; # 527
PLANT.LOOP1.TEXTRNL(3) := 291.15 ; # 528
PLANT.LOOP1.TEXTRNL(4) := 291.15 ; # 529
PLANT.LOOP1.TEXTRNL(5) := 291.15 ; # 530
PLANT.LOOP1.TEXTRNL(6) := 291.15 ; # 531
PLANT.LOOP1.TEXTRNL(7) := 291.15 ; # 532
PLANT.LOOP1.TEXTRNL(8) := 291.15 ; # 533
PLANT.LOOP1.TEXTRNL(9) := 291.15 ; # 534
PLANT.LOOP1.TEXTRNL(10) := 291.15 ; # 535
PLANT.LOOP1.TEXTRNL(11) := 889.835 ; # 536
PLANT.LOOP1.TEXTRNL(12) := 900.086 ; # 537
PLANT.LOOP1.TEXTRNL(13) := 910.162 ; # 538
PLANT.LOOP1.TEXTRNL(14) := 920.011 ; # 539
PLANT.LOOP1.TEXTRNL(15) := 929.638 ; # 540
PLANT.LOOP1.TEXTRNL(16) := 939.048 ; # 541
PLANT.LOOP1.TEXTRNL(17) := 948.245 ; # 542
PLANT.LOOP1.TEXTRNL(18) := 957.235 ; # 543
PLANT.LOOP1.TEXTRNL(19) := 966.021 ; # 544
PLANT.LOOP1.TEXTRNL(20) := 974.61 ; # 545
PLANT.LOOP1.TEXTRNL(21) := 983.004 ; # 546
PLANT.LOOP1.TEXTRNL(22) := 991.209 ; # 547
PLANT.LOOP1.TEXTRNL(23) := 999.229 ; # 548
PLANT.LOOP1.TEXTRNL(24) := 1007.07 ; # 549
PLANT.LOOP1.TEXTRNL(25) := 1014.73 ; # 550
PLANT.LOOP1.TEXTRNL(26) := 1022.22 ; # 551
PLANT.LOOP1.TEXTRNL(27) := 1029.54 ; # 552
PLANT.LOOP1.TEXTRNL(28) := 1036.69 ; # 553
PLANT.LOOP1.TEXTRNL(29) := 1043.69 ; # 554
PLANT.LOOP1.TEXTRNL(30) := 1050.52 ; # 555
PLANT.LOOP1.TEXTRNL(31) := 1057.2 ; # 556
PLANT.LOOP1.TEXTRNL(32) := 1063.73 ; # 557
PLANT.LOOP1.TEXTRNL(33) := 1070.12 ; # 558
PLANT.LOOP1.TEXTRNL(34) := 1076.36 ; # 559
PLANT.LOOP1.TEXTRNL(35) := 1082.45 ; # 560
PLANT.LOOP1.TEXTRNL(36) := 1088.41 ; # 561
PLANT.LOOP1.TEXTRNL(37) := 1094.24 ; # 562
PLANT.LOOP1.TEXTRNL(38) := 1099.93 ; # 563
PLANT.LOOP1.TEXTRNL(39) := 1105.5 ; # 564
PLANT.LOOP1.TEXTRNL(40) := 1110.94 ; # 565
PLANT.LOOP1.TEXTRNL(41) := 1116.26 ; # 566
PLANT.LOOP1.TEXTRNL(42) := 1121.45 ; # 567
PLANT.LOOP1.TEXTRNL(43) := 1126.53 ; # 568
PLANT.LOOP1.TEXTRNL(44) := 1131.5 ; # 569
PLANT.LOOP1.TEXTRNL(45) := 1136.35 ; # 570
PLANT.LOOP1.TEXTRNL(46) := 1141.1 ; # 571
PLANT.LOOP1.TEXTRNL(47) := 1145.73 ; # 572
PLANT.LOOP1.TEXTRNL(48) := 1150.27 ; # 573
PLANT.LOOP1.TEXTRNL(49) := 1154.7 ; # 574
PLANT.LOOP1.TEXTRNL(50) := 1159 ; # 575
PLANT.LOOP1.TEXTRNL(51) := 291.15 ; # 576
PLANT.LOOP1.TEXTRNL(52) := 291.15 ; # 577
PLANT.LOOP1.TEXTRNL(53) := 291.15 ; # 578
PLANT.LOOP1.TEXTRNL(54) := 291.15 ; # 579
PLANT.LOOP1.TEXTRNL(55) := 291.15 ; # 580
PLANT.LOOP1.TEXTRNL(56) := 291.15 ; # 581
PLANT.LOOP1.TEXTRNL(57) := 291.15 ; # 582
PLANT.LOOP1.TEXTRNL(58) := 291.15 ; # 583
PLANT.LOOP1.TEXTRNL(59) := 291.15 ; # 584
PLANT.LOOP1.TEXTRNL(60) := 291.15 ; # 585
PLANT.LOOP1.TEXTRNL(61) := 1132.12 ; # 586
PLANT.LOOP1.TEXTRNL(62) := 1129.7 ; # 587
PLANT.LOOP1.TEXTRNL(63) := 1127.12 ; # 588
PLANT.LOOP1.TEXTRNL(64) := 1124.36 ; # 589
PLANT.LOOP1.TEXTRNL(65) := 1121.43 ; # 590
PLANT.LOOP1.TEXTRNL(66) := 1118.29 ; # 591
PLANT.LOOP1.TEXTRNL(67) := 1114.96 ; # 592
PLANT.LOOP1.TEXTRNL(68) := 1111.4 ; # 593
PLANT.LOOP1.TEXTRNL(69) := 1107.6 ; # 594
PLANT.LOOP1.TEXTRNL(70) := 1103.55 ; # 595
PLANT.LOOP1.TEXTRNL(71) := 1099.23 ; # 596
PLANT.LOOP1.TEXTRNL(72) := 1094.63 ; # 597
PLANT.LOOP1.TEXTRNL(73) := 1089.72 ; # 598
PLANT.LOOP1.TEXTRNL(74) := 1084.49 ; # 599
PLANT.LOOP1.TEXTRNL(75) := 1078.91 ; # 600
```

```
PLANT.LOOP1.TEXTRNL(76) := 1072.96 ; # 601
PLANT.LOOP1.TEXTRNL(77) := 1066.61 ; # 602
PLANT.LOOP1.TEXTRNL(78) := 1059.84 ; # 603
PLANT.LOOP1.TEXTRNL(79) := 1052.63 ; # 604
PLANT.LOOP1.TEXTRNL(80) := 1044.93 ; # 605
PLANT.LOOP1.TEXTRNL(81) := 1036.73 ; # 606
PLANT.LOOP1.TEXTRNL(82) := 1027.98 ; # 607
PLANT.LOOP1.TEXTRNL(83) := 1018.65 ; # 608
PLANT.LOOP1.TEXTRNL(84) := 1008.7 ; # 609
PLANT.LOOP1.TEXTRNL(85) := 998.095 ; # 610
PLANT.LOOP1.TEXTRNL(86) := 986.784 ; # 611
PLANT.LOOP1.TEXTRNL(87) := 974.722 ; # 612
PLANT.LOOP1.TEXTRNL(88) := 961.86 ; # 613
PLANT.LOOP1.TEXTRNL(89) := 948.145 ; # 614
PLANT.LOOP1.TEXTRNL(90) := 933.521 ; # 615
PLANT.LOOP1.TEXTRNL(91) := 917.926 ; # 616
PLANT.LOOP1.TEXTRNL(92) := 901.297 ; # 617
PLANT.LOOP1.TEXTRNL(93) := 883.565 ; # 618
PLANT.LOOP1.TEXTRNL(94) := 864.657 ; # 619
PLANT.LOOP1.TEXTRNL(95) := 844.495 ; # 620
PLANT.LOOP1.TEXTRNL(96) := 822.996 ; # 621
PLANT.LOOP1.TEXTRNL(97) := 800.07 ; # 622
PLANT.LOOP1.TEXTRNL(98) := 775.624 ; # 623
PLANT.LOOP1.TEXTRNL(99) := 749.557 ; # 624
PLANT.LOOP1.TEXTRNL(100) := 721.768 ; # 625
PLANT.LOOP1.PSIEXTRA(1) := 1.28551e+006 ; # 626
PLANT.LOOP1.PSIEXTRA(2) := 1.28551e+006 ; # 626
PLANT.LOOP1.PSI(1) := 1.28551e+006 ; # 626
PLANT.LOOP1.PSI(2) := 1.27902e+006 ; # 627
PLANT.LOOP1.PSI(3) := 1.27441e+006 ; # 628
PLANT.LOOP1.PSI(4) := 1.26979e+006 ; # 629
PLANT.LOOP1.PSI(5) := 1.26517e+006 ; # 630
PLANT.LOOP1.PSI(6) := 1.26054e+006 ; # 631
PLANT.LOOP1.PSI(7) := 1.25591e+006 ; # 632
PLANT.LOOP1.PSI(8) := 1.25127e+006 ; # 633
PLANT.LOOP1.PSI(9) := 1.24663e+006 ; # 634
PLANT.LOOP1.PSI(10) := 1.24198e+006 ; # 635
PLANT.LOOP1.PSI(11) := 1.27601e+006 ; # 636
PLANT.LOOP1.PSI(12) := 1.29599e+006 ; # 637
PLANT.LOOP1.PSI(13) := 1.31449e+006 ; # 638
PLANT.LOOP1.PSI(14) := 1.33196e+006 ; # 639
PLANT.LOOP1.PSI(15) := 1.34847e+006 ; # 640
PLANT.LOOP1.PSI(16) := 1.36409e+006 ; # 641
PLANT.LOOP1.PSI(17) := 1.37885e+006 ; # 642
PLANT.LOOP1.PSI(18) := 1.39281e+006 ; # 643
PLANT.LOOP1.PSI(19) := 1.40601e+006 ; # 644
PLANT.LOOP1.PSI(20) := 1.4185e+006 ; # 645
PLANT.LOOP1.PSI(21) := 1.4303e+006 ; # 646
PLANT.LOOP1.PSI(22) := 1.44146e+006 ; # 647
PLANT.LOOP1.PSI(23) := 1.45201e+006 ; # 648
PLANT.LOOP1.PSI(24) := 1.46198e+006 ; # 649
PLANT.LOOP1.PSI(25) := 1.4714e+006 ; # 650
PLANT.LOOP1.PSI(26) := 1.4803e+006 ; # 651
PLANT.LOOP1.PSI(27) := 1.48869e+006 ; # 652
PLANT.LOOP1.PSI(28) := 1.49661e+006 ; # 653
PLANT.LOOP1.PSI(29) := 1.50407e+006 ; # 654
PLANT.LOOP1.PSI(30) := 1.5111e+006 ; # 655
PLANT.LOOP1.PSI(31) := 1.51772e+006 ; # 656
PLANT.LOOP1.PSI(32) := 1.52393e+006 ; # 657
PLANT.LOOP1.PSI(33) := 1.52977e+006 ; # 658
PLANT.LOOP1.PSI(34) := 1.53525e+006 ; # 659
PLANT.LOOP1.PSI(35) := 1.54038e+006 ; # 660
PLANT.LOOP1.PSI(36) := 1.54518e+006 ; # 661
PLANT.LOOP1.PSI(37) := 1.54965e+006 ; # 662
PLANT.LOOP1.PSI(38) := 1.55382e+006 ; # 663
PLANT.LOOP1.PSI(39) := 1.5577e+006 ; # 664
PLANT.LOOP1.PSI(40) := 1.56129e+006 ; # 665
PLANT.LOOP1.PSI(41) := 1.56461e+006 ; # 666
PLANT.LOOP1.PSI(42) := 1.56766e+006 ; # 667
PLANT.LOOP1.PSI(43) := 1.57047e+006 ; # 668
PLANT.LOOP1.PSI(44) := 1.57303e+006 ; # 669
PLANT.LOOP1.PSI(45) := 1.57536e+006 ; # 670
PLANT.LOOP1.PSI(46) := 1.57746e+006 ; # 671
PLANT.LOOP1.PSI(47) := 1.57935e+006 ; # 672
PLANT.LOOP1.PSI(48) := 1.58102e+006 ; # 673
PLANT.LOOP1.PSI(49) := 1.58249e+006 ; # 674
PLANT.LOOP1.PSI(50) := 1.58376e+006 ; # 675
PLANT.LOOP1.PSI(51) := 1.55653e+006 ; # 676
PLANT.LOOP1.PSI(52) := 1.5504e+006 ; # 677
PLANT.LOOP1.PSI(53) := 1.5443e+006 ; # 678
PLANT.LOOP1.PSI(54) := 1.53819e+006 ; # 679
PLANT.LOOP1.PSI(55) := 1.53207e+006 ; # 680
PLANT.LOOP1.PSI(56) := 1.52594e+006 ; # 681
PLANT.LOOP1.PSI(57) := 1.5198e+006 ; # 682
PLANT.LOOP1.PSI(58) := 1.51364e+006 ; # 683
PLANT.LOOP1.PSI(59) := 1.50748e+006 ; # 684
PLANT.LOOP1.PSI(60) := 1.5013e+006 ; # 685
PLANT.LOOP1.PSI(61) := 1.51696e+006 ; # 686
PLANT.LOOP1.PSI(62) := 1.51365e+006 ; # 687
PLANT.LOOP1.PSI(63) := 1.51012e+006 ; # 688
PLANT.LOOP1.PSI(64) := 1.50644e+006 ; # 689
PLANT.LOOP1.PSI(65) := 1.5026e+006 ; # 690
```

```
PLANT.LOOP1.PSI(66) := 1.49858e+006 ; # 691
PLANT.LOOP1.PSI(67) := 1.49439e+006 ; # 692
PLANT.LOOP1.PSI(68) := 1.49e+006 ; # 693
PLANT.LOOP1.PSI(69) := 1.48539e+006 ; # 694
PLANT.LOOP1.PSI(70) := 1.48056e+006 ; # 695
PLANT.LOOP1.PSI(71) := 1.47548e+006 ; # 696
PLANT.LOOP1.PSI(72) := 1.47014e+006 ; # 697
PLANT.LOOP1.PSI(73) := 1.46451e+006 ; # 698
PLANT.LOOP1.PSI(74) := 1.45857e+006 ; # 699
PLANT.LOOP1.PSI(75) := 1.45229e+006 ; # 700
PLANT.LOOP1.PSI(76) := 1.44565e+006 ; # 701
PLANT.LOOP1.PSI(77) := 1.43862e+006 ; # 702
PLANT.LOOP1.PSI(78) := 1.43116e+006 ; # 703
PLANT.LOOP1.PSI(79) := 1.42323e+006 ; # 704
PLANT.LOOP1.PSI(80) := 1.4148e+006 ; # 705
PLANT.LOOP1.PSI(81) := 1.40582e+006 ; # 706
PLANT.LOOP1.PSI(82) := 1.39623e+006 ; # 707
PLANT.LOOP1.PSI(83) := 1.38599e+006 ; # 708
PLANT.LOOP1.PSI(84) := 1.37502e+006 ; # 709
PLANT.LOOP1.PSI(85) := 1.36326e+006 ; # 710
PLANT.LOOP1.PSI(86) := 1.35062e+006 ; # 711
PLANT.LOOP1.PSI(87) := 1.33702e+006 ; # 712
PLANT.LOOP1.PSI(88) := 1.32234e+006 ; # 713
PLANT.LOOP1.PSI(89) := 1.30649e+006 ; # 714
PLANT.LOOP1.PSI(90) := 1.28931e+006 ; # 715
PLANT.LOOP1.PSI(91) := 1.27066e+006 ; # 716
PLANT.LOOP1.PSI(92) := 1.25037e+006 ; # 717
PLANT.LOOP1.PSI(93) := 1.22822e+006 ; # 718
PLANT.LOOP1.PSI(94) := 1.20397e+006 ; # 719
PLANT.LOOP1.PSI(95) := 1.17735e+006 ; # 720
PLANT.LOOP1.PSI(96) := 1.14802e+006 ; # 721
PLANT.LOOP1.PSI(97) := 1.11558e+006 ; # 722
PLANT.LOOP1.PSI(98) := 1.07957e+006 ; # 723
PLANT.LOOP1.PSI(99) := 1.03941e+006 ; # 724
PLANT.LOOP1.PSI(100) := 994398 ; # 725
PLANT.LOOP1.FEXTRA(1) := 27.6937 ; # 726
PLANT.LOOP1.FEXTRA(2) := 27.6937 ; # 726
PLANT.LOOP1.F(1) := 27.6937 ; # 726
PLANT.LOOP1.F(2) := 27.6937 ; # 727
PLANT.LOOP1.F(3) := 27.6937 ; # 728
PLANT.LOOP1.F(4) := 27.6937 ; # 729
PLANT.LOOP1.F(5) := 27.6937 ; # 730
PLANT.LOOP1.F(6) := 27.6937 ; # 731
PLANT.LOOP1.F(7) := 27.6937 ; # 732
PLANT.LOOP1.F(8) := 27.6937 ; # 733
PLANT.LOOP1.F(9) := 27.6937 ; # 734
PLANT.LOOP1.F(10) := 27.6937 ; # 735
PLANT.LOOP1.F(11) := 27.6937 ; # 736
PLANT.LOOP1.F(12) := 27.6937 ; # 737
PLANT.LOOP1.F(13) := 27.6937 ; # 738
PLANT.LOOP1.F(14) := 27.6937 ; # 739
PLANT.LOOP1.F(15) := 27.6937 ; # 740
PLANT.LOOP1.F(16) := 27.6937 ; # 741
PLANT.LOOP1.F(17) := 27.6937 ; # 742
PLANT.LOOP1.F(18) := 27.6937 ; # 743
PLANT.LOOP1.F(19) := 27.6937 ; # 744
PLANT.LOOP1.F(20) := 27.6937 ; # 745
PLANT.LOOP1.F(21) := 27.6937 ; # 746
PLANT.LOOP1.F(22) := 27.6937 ; # 747
PLANT.LOOP1.F(23) := 27.6937 ; # 748
PLANT.LOOP1.F(24) := 27.6937 ; # 749
PLANT.LOOP1.F(25) := 27.6937 ; # 750
PLANT.LOOP1.F(26) := 27.6937 ; # 751
PLANT.LOOP1.F(27) := 27.6937 ; # 752
PLANT.LOOP1.F(28) := 27.6937 ; # 753
PLANT.LOOP1.F(29) := 27.6937 ; # 754
PLANT.LOOP1.F(30) := 27.6937 ; # 755
PLANT.LOOP1.F(31) := 27.6937 ; # 756
PLANT.LOOP1.F(32) := 27.6937 ; # 757
PLANT.LOOP1.F(33) := 27.6937 ; # 758
PLANT.LOOP1.F(34) := 27.6937 ; # 759
PLANT.LOOP1.F(35) := 27.6937 ; # 760
PLANT.LOOP1.F(36) := 27.6937 ; # 761
PLANT.LOOP1.F(37) := 27.6937 ; # 762
PLANT.LOOP1.F(38) := 27.6937 ; # 763
PLANT.LOOP1.F(39) := 27.6937 ; # 764
PLANT.LOOP1.F(40) := 27.6937 ; # 765
PLANT.LOOP1.F(41) := 27.6937 ; # 766
PLANT.LOOP1.F(42) := 27.6937 ; # 767
PLANT.LOOP1.F(43) := 27.6937 ; # 768
PLANT.LOOP1.F(44) := 27.6937 ; # 769
PLANT.LOOP1.F(45) := 27.6937 ; # 770
PLANT.LOOP1.F(46) := 27.6937 ; # 771
PLANT.LOOP1.F(47) := 27.6937 ; # 772
PLANT.LOOP1.F(48) := 27.6937 ; # 773
PLANT.LOOP1.F(49) := 27.6937 ; # 774
PLANT.LOOP1.F(50) := 27.6937 ; # 775
PLANT.LOOP1.F(51) := 27.6937 ; # 776
PLANT.LOOP1.F(52) := 27.6937 ; # 777
PLANT.LOOP1.F(53) := 27.6937 ; # 778
PLANT.LOOP1.F(54) := 27.6937 ; # 779
PLANT.LOOP1.F(55) := 27.6937 ; # 780
```

```
PLANT.LOOP1.F(56) := 27.6937 ; # 781
PLANT.LOOP1.F(57) := 27.6937 ; # 782
PLANT.LOOP1.F(58) := 27.6937 ; # 783
PLANT.LOOP1.F(59) := 27.6937 ; # 784
PLANT.LOOP1.F(60) := 27.6937 ; # 785
PLANT.LOOP1.F(61) := 27.6937 ; # 786
PLANT.LOOP1.F(62) := 27.6937 ; # 787
PLANT.LOOP1.F(63) := 27.6937 ; # 788
PLANT.LOOP1.F(64) := 27.6937 ; # 789
PLANT.LOOP1.F(65) := 27.6937 ; # 790
PLANT.LOOP1.F(66) := 27.6937 ; # 791
PLANT.LOOP1.F(67) := 27.6937 ; # 792
PLANT.LOOP1.F(68) := 27.6937 ; # 793
PLANT.LOOP1.F(69) := 27.6937 ; # 794
PLANT.LOOP1.F(70) := 27.6937 ; # 795
PLANT.LOOP1.F(71) := 27.6937 ; # 796
PLANT.LOOP1.F(72) := 27.6937 ; # 797
PLANT.LOOP1.F(73) := 27.6937 ; # 798
PLANT.LOOP1.F(74) := 27.6937 ; # 799
PLANT.LOOP1.F(75) := 27.6937 ; # 800
PLANT.LOOP1.F(76) := 27.6937 ; # 801
PLANT.LOOP1.F(77) := 27.6937 ; # 802
PLANT.LOOP1.F(78) := 27.6937 ; # 803
PLANT.LOOP1.F(79) := 27.6937 ; # 804
PLANT.LOOP1.F(80) := 27.6937 ; # 805
PLANT.LOOP1.F(81) := 27.6937 ; # 806
PLANT.LOOP1.F(82) := 27.6937 ; # 807
PLANT.LOOP1.F(83) := 27.6937 ; # 808
PLANT.LOOP1.F(84) := 27.6937 ; # 809
PLANT.LOOP1.F(85) := 27.6937 ; # 810
PLANT.LOOP1.F(86) := 27.6937 ; # 811
PLANT.LOOP1.F(87) := 27.6937 ; # 812
PLANT.LOOP1.F(88) := 27.6937 ; # 813
PLANT.LOOP1.F(89) := 27.6937 ; # 814
PLANT.LOOP1.F(90) := 27.6937 ; # 815
PLANT.LOOP1.F(91) := 27.6937 ; # 816
PLANT.LOOP1.F(92) := 27.6937 ; # 817
PLANT.LOOP1.F(93) := 27.6937 ; # 818
PLANT.LOOP1.F(94) := 27.6937 ; # 819
PLANT.LOOP1.F(95) := 27.6937 ; # 820
PLANT.LOOP1.F(96) := 27.6937 ; # 821
PLANT.LOOP1.F(97) := 27.6937 ; # 822
PLANT.LOOP1.F(98) := 27.6937 ; # 823
PLANT.LOOP1.F(99) := 27.6937 ; # 824
PLANT.LOOP1.F(100) := 27.6937 ; # 825
PLANT.LOOP1.F_IN := 27.6937 ; # 826
PLANT.LOOP1.RHO_AVG := 0.919725 ; # 827
PLANT.LOOP1.TOTALMASS := 27.4149 ; # 828
PLANT.LOOP1.HT1.H2 := 17.344 ; # 829
PLANT.LOOP1.HT1.RHOPIPEHE := 1.13304 ; # 830
PLANT.LOOP1.HT1.VPIPE := 182.461 ; # 831
PLANT.LOOP1.HT1.H0 := 1563 ; # 832
PLANT.LOOP1.HT1.U := 8.93635 ; # 833
PLANT.LOOP1.HT2.RHOPIPEHE := 0.915592 ; # 834
PLANT.LOOP1.HT2.VPIPE := 87.9967 ; # 835
PLANT.LOOP1.HT2.H0 := 1665.62 ; # 836
PLANT.LOOP1.HT2.U := 1645.45 ; # 837
PLANT.LOOP1.HT3.H2 := 17.344 ; # 838
PLANT.LOOP1.HT3.RHOPIPEHE := 0.75785 ; # 839
PLANT.LOOP1.HT3.VPIPE := 236.032 ; # 840
PLANT.LOOP1.HT3.H0 := 1372.09 ; # 841
PLANT.LOOP1.HT3.U := 10.3605 ; # 842
PLANT.LOOP1.HT4.RHOPIPEHE := 0.829069 ; # 843
PLANT.LOOP1.HT4.VPIPE := 83.9116 ; # 844
PLANT.LOOP1.HT4.U := 1299.78 ; # 845
PLANT.PIPEH.RHO(1) := 2.88031 ; # 846
PLANT.PIPEH.RHO(2) := 2.88987 ; # 847
PLANT.PIPEH.RHO(3) := 2.89973 ; # 848
PLANT.PIPEH.RHO(4) := 2.90991 ; # 849
PLANT.PIPEH.RHO(5) := 2.92042 ; # 850
PLANT.PIPEH.RHO(6) := 2.93127 ; # 851
PLANT.PIPEH.RHO(7) := 2.94248 ; # 852
PLANT.PIPEH.RHO(8) := 2.95405 ; # 853
PLANT.PIPEH.RHO(9) := 2.96601 ; # 854
PLANT.PIPEH.RHO(10) := 2.97836 ; # 855
PLANT.PIPEH.RHO(11) := 2.99112 ; # 856
PLANT.PIPEH.RHO(12) := 3.00432 ; # 857
PLANT.PIPEH.RHO(13) := 3.01796 ; # 858
PLANT.PIPEH.RHO(14) := 3.03206 ; # 859
PLANT.PIPEH.RHO(15) := 3.04666 ; # 860
PLANT.PIPEH.RHO(16) := 3.06175 ; # 861
PLANT.PIPEH.RHO(17) := 3.07737 ; # 862
PLANT.PIPEH.RHO(18) := 3.09354 ; # 863
PLANT.PIPEH.RHO(19) := 3.11029 ; # 864
PLANT.PIPEH.RHO(20) := 3.12763 ; # 865
PLANT.PIPEH.RHO(21) := 3.1456 ; # 866
PLANT.PIPEH.RHO(22) := 3.16422 ; # 867
PLANT.PIPEH.RHO(23) := 3.18352 ; # 868
PLANT.PIPEH.RHO(24) := 3.20354 ; # 869
PLANT.PIPEH.RHO(25) := 3.22431 ; # 870
PLANT.PIPEH.RHO(26) := 3.24587 ; # 871
PLANT.PIPEH.RHO(27) := 3.26824 ; # 872
```

```
PLANT.PIPEH.RHO(28) := 3.29149 ; # 873
PLANT.PIPEH.RHO(29) := 3.31563 ; # 874
PLANT.PIPEH.RHO(30) := 3.34074 ; # 875
PLANT.PIPEH.RHO(31) := 3.36684 ; # 876
PLANT.PIPEH.RHO(32) := 3.394 ; # 877
PLANT.PIPEH.RHO(33) := 3.42228 ; # 878
PLANT.PIPEH.RHO(34) := 3.45173 ; # 879
PLANT.PIPEH.RHO(35) := 3.48241 ; # 880
PLANT.PIPEH.RHO(36) := 3.5144 ; # 881
PLANT.PIPEH.RHO(37) := 3.54778 ; # 882
PLANT.PIPEH.RHO(38) := 3.58262 ; # 883
PLANT.PIPEH.RHO(39) := 3.61901 ; # 884
PLANT.PIPEH.RHO(40) := 3.65698 ; # 885
PLANT.PIPEH.P(1) := 6.9993e+006 ; # 886
PLANT.PIPEH.P(2) := 6.99792e+006 ; # 887
PLANT.PIPEH.P(3) := 6.99655e+006 ; # 888
PLANT.PIPEH.P(4) := 6.99518e+006 ; # 889
PLANT.PIPEH.P(5) := 6.99382e+006 ; # 890
PLANT.PIPEH.P(6) := 6.99246e+006 ; # 891
PLANT.PIPEH.P(7) := 6.9911e+006 ; # 892
PLANT.PIPEH.P(8) := 6.98976e+006 ; # 893
PLANT.PIPEH.P(9) := 6.98841e+006 ; # 894
PLANT.PIPEH.P(10) := 6.98708e+006 ; # 895
PLANT.PIPEH.P(11) := 6.98575e+006 ; # 896
PLANT.PIPEH.P(12) := 6.98442e+006 ; # 897
PLANT.PIPEH.P(13) := 6.9831e+006 ; # 898
PLANT.PIPEH.P(14) := 6.98179e+006 ; # 899
PLANT.PIPEH.P(15) := 6.98049e+006 ; # 900
PLANT.PIPEH.P(16) := 6.97919e+006 ; # 901
PLANT.PIPEH.P(17) := 6.9779e+006 ; # 902
PLANT.PIPEH.P(18) := 6.97661e+006 ; # 903
PLANT.PIPEH.P(19) := 6.97534e+006 ; # 904
PLANT.PIPEH.P(20) := 6.97407e+006 ; # 905
PLANT.PIPEH.P(21) := 6.97281e+006 ; # 906
PLANT.PIPEH.P(22) := 6.97155e+006 ; # 907
PLANT.PIPEH.P(23) := 6.97031e+006 ; # 908
PLANT.PIPEH.P(24) := 6.96907e+006 ; # 909
PLANT.PIPEH.P(25) := 6.96784e+006 ; # 910
PLANT.PIPEH.P(26) := 6.96662e+006 ; # 911
PLANT.PIPEH.P(27) := 6.9654e+006 ; # 912
PLANT.PIPEH.P(28) := 6.9642e+006 ; # 913
PLANT.PIPEH.P(29) := 6.96301e+006 ; # 914
PLANT.PIPEH.P(30) := 6.96182e+006 ; # 915
PLANT.PIPEH.P(31) := 6.96065e+006 ; # 916
PLANT.PIPEH.P(32) := 6.95948e+006 ; # 917
PLANT.PIPEH.P(33) := 6.95833e+006 ; # 918
PLANT.PIPEH.P(34) := 6.95718e+006 ; # 919
PLANT.PIPEH.P(35) := 6.95605e+006 ; # 920
PLANT.PIPEH.P(36) := 6.95493e+006 ; # 921
PLANT.PIPEH.P(37) := 6.95381e+006 ; # 922
PLANT.PIPEH.P(38) := 6.95271e+006 ; # 923
PLANT.PIPEH.P(39) := 6.95162e+006 ; # 924
PLANT.PIPEH.P(40) := 6.95055e+006 ; # 925
PLANT.PIPEH.H(1) := 4.52556e+006 ; # 926
PLANT.PIPEH.H(2) := 4.50427e+006 ; # 927
PLANT.PIPEH.H(3) := 4.4825e+006 ; # 928
PLANT.PIPEH.H(4) := 4.46023e+006 ; # 929
PLANT.PIPEH.H(5) := 4.43744e+006 ; # 930
PLANT.PIPEH.H(6) := 4.41412e+006 ; # 931
PLANT.PIPEH.H(7) := 4.39027e+006 ; # 932
PLANT.PIPEH.H(8) := 4.36587e+006 ; # 933
PLANT.PIPEH.H(9) := 4.3409e+006 ; # 934
PLANT.PIPEH.H(10) := 4.31535e+006 ; # 935
PLANT.PIPEH.H(11) := 4.28922e+006 ; # 936
PLANT.PIPEH.H(12) := 4.26248e+006 ; # 937
PLANT.PIPEH.H(13) := 4.23513e+006 ; # 938
PLANT.PIPEH.H(14) := 4.20714e+006 ; # 939
PLANT.PIPEH.H(15) := 4.17851e+006 ; # 940
PLANT.PIPEH.H(16) := 4.14921e+006 ; # 941
PLANT.PIPEH.H(17) := 4.11924e+006 ; # 942
PLANT.PIPEH.H(18) := 4.08858e+006 ; # 943
PLANT.PIPEH.H(19) := 4.05721e+006 ; # 944
PLANT.PIPEH.H(20) := 4.02511e+006 ; # 945
PLANT.PIPEH.H(21) := 3.99227e+006 ; # 946
PLANT.PIPEH.H(22) := 3.95868e+006 ; # 947
PLANT.PIPEH.H(23) := 3.92431e+006 ; # 948
PLANT.PIPEH.H(24) := 3.88914e+006 ; # 949
PLANT.PIPEH.H(25) := 3.85317e+006 ; # 950
PLANT.PIPEH.H(26) := 3.81636e+006 ; # 951
PLANT.PIPEH.H(27) := 3.7787e+006 ; # 952
PLANT.PIPEH.H(28) := 3.74017e+006 ; # 953
PLANT.PIPEH.H(29) := 3.70076e+006 ; # 954
PLANT.PIPEH.H(30) := 3.66043e+006 ; # 955
PLANT.PIPEH.H(31) := 3.61917e+006 ; # 956
PLANT.PIPEH.H(32) := 3.57696e+006 ; # 957
PLANT.PIPEH.H(33) := 3.53378e+006 ; # 958
PLANT.PIPEH.H(34) := 3.48959e+006 ; # 959
PLANT.PIPEH.H(35) := 3.44439e+006 ; # 960
PLANT.PIPEH.H(36) := 3.39814e+006 ; # 961
PLANT.PIPEH.H(37) := 3.35083e+006 ; # 962
PLANT.PIPEH.H(38) := 3.30242e+006 ; # 963
PLANT.PIPEH.H(39) := 3.2529e+006 ; # 964
```

```
979   PLANT.PIPEH.H(40) := 3.20232e+006 ; # 965
980   PLANT.PIPEH.H0 := 4.5464e+006 ; # 966
981   PLANT.PIPEH.T(1) := 1169.14 ; # 967
982   PLANT.PIPEH.T(2) := 1165.04 ; # 968
983   PLANT.PIPEH.T(3) := 1160.85 ; # 969
984   PLANT.PIPEH.T(4) := 1156.56 ; # 970
985   PLANT.PIPEH.T(5) := 1152.18 ; # 971
986   PLANT.PIPEH.T(6) := 1147.69 ; # 972
987   PLANT.PIPEH.T(7) := 1143.1 ; # 973
988   PLANT.PIPEH.T(8) := 1138.4 ; # 974
989   PLANT.PIPEH.T(9) := 1133.59 ; # 975
990   PLANT.PIPEH.T(10) := 1128.67 ; # 976
991   PLANT.PIPEH.T(11) := 1123.64 ; # 977
992   PLANT.PIPEH.T(12) := 1118.5 ; # 978
993   PLANT.PIPEH.T(13) := 1113.23 ; # 979
994   PLANT.PIPEH.T(14) := 1107.84 ; # 980
995   PLANT.PIPEH.T(15) := 1102.33 ; # 981
996   PLANT.PIPEH.T(16) := 1096.69 ; # 982
997   PLANT.PIPEH.T(17) := 1090.92 ; # 983
998   PLANT.PIPEH.T(18) := 1085.02 ; # 984
999   PLANT.PIPEH.T(19) := 1078.98 ; # 985
1000  PLANT.PIPEH.T(20) := 1072.8 ; # 986
1001  PLANT.PIPEH.T(21) := 1066.48 ; # 987
1002  PLANT.PIPEH.T(22) := 1060.02 ; # 988
1003  PLANT.PIPEH.T(23) := 1053.4 ; # 989
1004  PLANT.PIPEH.T(24) := 1046.63 ; # 990
1005  PLANT.PIPEH.T(25) := 1039.71 ; # 991
1006  PLANT.PIPEH.T(26) := 1032.62 ; # 992
1007  PLANT.PIPEH.T(27) := 1025.37 ; # 993
1008  PLANT.PIPEH.T(28) := 1017.96 ; # 994
1009  PLANT.PIPEH.T(29) := 1010.37 ; # 995
1010  PLANT.PIPEH.T(30) := 1002.61 ; # 996
1011  PLANT.PIPEH.T(31) := 994.665 ; # 997
1012  PLANT.PIPEH.T(32) := 986.539 ; # 998
1013  PLANT.PIPEH.T(33) := 978.227 ; # 999
1014  PLANT.PIPEH.T(34) := 969.722 ; # 1000
1015  PLANT.PIPEH.T(35) := 961.02 ; # 1001
1016  PLANT.PIPEH.T(36) := 952.118 ; # 1002
1017  PLANT.PIPEH.T(37) := 943.011 ; # 1003
1018  PLANT.PIPEH.T(38) := 933.692 ; # 1004
1019  PLANT.PIPEH.T(39) := 924.159 ; # 1005
1020  PLANT.PIPEH.T(40) := 914.423 ; # 1006
1021  PLANT.PIPEH.TEXTRNL(1) := 1159 ; # 1007
1022  PLANT.PIPEH.TEXTRNL(2) := 1154.7 ; # 1008
1023  PLANT.PIPEH.TEXTRNL(3) := 1150.27 ; # 1009
1024  PLANT.PIPEH.TEXTRNL(4) := 1145.73 ; # 1010
1025  PLANT.PIPEH.TEXTRNL(5) := 1141.1 ; # 1011
1026  PLANT.PIPEH.TEXTRNL(6) := 1136.35 ; # 1012
1027  PLANT.PIPEH.TEXTRNL(7) := 1131.5 ; # 1013
1028  PLANT.PIPEH.TEXTRNL(8) := 1126.53 ; # 1014
1029  PLANT.PIPEH.TEXTRNL(9) := 1121.45 ; # 1015
1030  PLANT.PIPEH.TEXTRNL(10) := 1116.26 ; # 1016
1031  PLANT.PIPEH.TEXTRNL(11) := 1110.94 ; # 1017
1032  PLANT.PIPEH.TEXTRNL(12) := 1105.5 ; # 1018
1033  PLANT.PIPEH.TEXTRNL(13) := 1099.93 ; # 1019
1034  PLANT.PIPEH.TEXTRNL(14) := 1094.24 ; # 1020
1035  PLANT.PIPEH.TEXTRNL(15) := 1088.41 ; # 1021
1036  PLANT.PIPEH.TEXTRNL(16) := 1082.45 ; # 1022
1037  PLANT.PIPEH.TEXTRNL(17) := 1076.36 ; # 1023
1038  PLANT.PIPEH.TEXTRNL(18) := 1070.12 ; # 1024
1039  PLANT.PIPEH.TEXTRNL(19) := 1063.73 ; # 1025
1040  PLANT.PIPEH.TEXTRNL(20) := 1057.2 ; # 1026
1041  PLANT.PIPEH.TEXTRNL(21) := 1050.52 ; # 1027
1042  PLANT.PIPEH.TEXTRNL(22) := 1043.69 ; # 1028
1043  PLANT.PIPEH.TEXTRNL(23) := 1036.69 ; # 1029
1044  PLANT.PIPEH.TEXTRNL(24) := 1029.54 ; # 1030
1045  PLANT.PIPEH.TEXTRNL(25) := 1022.22 ; # 1031
1046  PLANT.PIPEH.TEXTRNL(26) := 1014.73 ; # 1032
1047  PLANT.PIPEH.TEXTRNL(27) := 1007.07 ; # 1033
1048  PLANT.PIPEH.TEXTRNL(28) := 999.229 ; # 1034
1049  PLANT.PIPEH.TEXTRNL(29) := 991.209 ; # 1035
1050  PLANT.PIPEH.TEXTRNL(30) := 983.004 ; # 1036
1051  PLANT.PIPEH.TEXTRNL(31) := 974.61 ; # 1037
1052  PLANT.PIPEH.TEXTRNL(32) := 966.021 ; # 1038
1053  PLANT.PIPEH.TEXTRNL(33) := 957.235 ; # 1039
1054  PLANT.PIPEH.TEXTRNL(34) := 948.245 ; # 1040
1055  PLANT.PIPEH.TEXTRNL(35) := 939.048 ; # 1041
1056  PLANT.PIPEH.TEXTRNL(36) := 929.638 ; # 1042
1057  PLANT.PIPEH.TEXTRNL(37) := 920.011 ; # 1043
1058  PLANT.PIPEH.TEXTRNL(38) := 910.162 ; # 1044
1059  PLANT.PIPEH.TEXTRNL(39) := 900.086 ; # 1045
1060  PLANT.PIPEH.TEXTRNL(40) := 889.835 ; # 1046
1061  PLANT.PIPEH.F(1) := 32.4687 ; # 1047
1062  PLANT.PIPEH.F(2) := 32.4687 ; # 1048
1063  PLANT.PIPEH.F(3) := 32.4687 ; # 1049
1064  PLANT.PIPEH.F(4) := 32.4687 ; # 1050
1065  PLANT.PIPEH.F(5) := 32.4687 ; # 1051
1066  PLANT.PIPEH.F(6) := 32.4687 ; # 1052
1067  PLANT.PIPEH.F(7) := 32.4687 ; # 1053
1068  PLANT.PIPEH.F(8) := 32.4687 ; # 1054
1069  PLANT.PIPEH.F(9) := 32.4687 ; # 1055
1070  PLANT.PIPEH.F(10) := 32.4687 ; # 1056
```

```
1071   PLANT.PIPEH.F(11) := 32.4687 ; # 1057
1072   PLANT.PIPEH.F(12) := 32.4687 ; # 1058
1073   PLANT.PIPEH.F(13) := 32.4687 ; # 1059
1074   PLANT.PIPEH.F(14) := 32.4687 ; # 1060
1075   PLANT.PIPEH.F(15) := 32.4687 ; # 1061
1076   PLANT.PIPEH.F(16) := 32.4687 ; # 1062
1077   PLANT.PIPEH.F(17) := 32.4687 ; # 1063
1078   PLANT.PIPEH.F(18) := 32.4687 ; # 1064
1079   PLANT.PIPEH.F(19) := 32.4687 ; # 1065
1080   PLANT.PIPEH.F(20) := 32.4687 ; # 1066
1081   PLANT.PIPEH.F(21) := 32.4687 ; # 1067
1082   PLANT.PIPEH.F(22) := 32.4687 ; # 1068
1083   PLANT.PIPEH.F(23) := 32.4687 ; # 1069
1084   PLANT.PIPEH.F(24) := 32.4687 ; # 1070
1085   PLANT.PIPEH.F(25) := 32.4687 ; # 1071
1086   PLANT.PIPEH.F(26) := 32.4687 ; # 1072
1087   PLANT.PIPEH.F(27) := 32.4687 ; # 1073
1088   PLANT.PIPEH.F(28) := 32.4687 ; # 1074
1089   PLANT.PIPEH.F(29) := 32.4687 ; # 1075
1090   PLANT.PIPEH.F(30) := 32.4687 ; # 1076
1091   PLANT.PIPEH.F(31) := 32.4687 ; # 1077
1092   PLANT.PIPEH.F(32) := 32.4687 ; # 1078
1093   PLANT.PIPEH.F(33) := 32.4687 ; # 1079
1094   PLANT.PIPEH.F(34) := 32.4687 ; # 1080
1095   PLANT.PIPEH.F(35) := 32.4687 ; # 1081
1096   PLANT.PIPEH.F(36) := 32.4687 ; # 1082
1097   PLANT.PIPEH.F(37) := 32.4687 ; # 1083
1098   PLANT.PIPEH.F(38) := 32.4687 ; # 1084
1099   PLANT.PIPEH.F(39) := 32.4687 ; # 1085
1100   PLANT.PIPEH.F(40) := 32.4687 ; # 1086
1101   PLANT.PIPEH.F_IN := 32.4687 ; # 1087
1102   PLANT.PIPEH.PSI(1) := 6.03571e+006 ; # 1088
1103   PLANT.PIPEH.PSI(2) := 6.01884e+006 ; # 1089
1104   PLANT.PIPEH.PSI(3) := 6.00151e+006 ; # 1090
1105   PLANT.PIPEH.PSI(4) := 5.9837e+006 ; # 1091
1106   PLANT.PIPEH.PSI(5) := 5.96538e+006 ; # 1092
1107   PLANT.PIPEH.PSI(6) := 5.94655e+006 ; # 1093
1108   PLANT.PIPEH.PSI(7) := 5.92717e+006 ; # 1094
1109   PLANT.PIPEH.PSI(8) := 5.90723e+006 ; # 1095
1110   PLANT.PIPEH.PSI(9) := 5.88672e+006 ; # 1096
1111   PLANT.PIPEH.PSI(10) := 5.86559e+006 ; # 1097
1112   PLANT.PIPEH.PSI(11) := 5.84383e+006 ; # 1098
1113   PLANT.PIPEH.PSI(12) := 5.82142e+006 ; # 1099
1114   PLANT.PIPEH.PSI(13) := 5.79833e+006 ; # 1100
1115   PLANT.PIPEH.PSI(14) := 5.77452e+006 ; # 1101
1116   PLANT.PIPEH.PSI(15) := 5.74998e+006 ; # 1102
1117   PLANT.PIPEH.PSI(16) := 5.72466e+006 ; # 1103
1118   PLANT.PIPEH.PSI(17) := 5.69854e+006 ; # 1104
1119   PLANT.PIPEH.PSI(18) := 5.67158e+006 ; # 1105
1120   PLANT.PIPEH.PSI(19) := 5.64374e+006 ; # 1106
1121   PLANT.PIPEH.PSI(20) := 5.61499e+006 ; # 1107
1122   PLANT.PIPEH.PSI(21) := 5.58528e+006 ; # 1108
1123   PLANT.PIPEH.PSI(22) := 5.55457e+006 ; # 1109
1124   PLANT.PIPEH.PSI(23) := 5.52282e+006 ; # 1110
1125   PLANT.PIPEH.PSI(24) := 5.48997e+006 ; # 1111
1126   PLANT.PIPEH.PSI(25) := 5.45597e+006 ; # 1112
1127   PLANT.PIPEH.PSI(26) := 5.42078e+006 ; # 1113
1128   PLANT.PIPEH.PSI(27) := 5.38432e+006 ; # 1114
1129   PLANT.PIPEH.PSI(28) := 5.34653e+006 ; # 1115
1130   PLANT.PIPEH.PSI(29) := 5.30735e+006 ; # 1116
1131   PLANT.PIPEH.PSI(30) := 5.26672e+006 ; # 1117
1132   PLANT.PIPEH.PSI(31) := 5.22454e+006 ; # 1118
1133   PLANT.PIPEH.PSI(32) := 5.18075e+006 ; # 1119
1134   PLANT.PIPEH.PSI(33) := 5.13524e+006 ; # 1120
1135   PLANT.PIPEH.PSI(34) := 5.08794e+006 ; # 1121
1136   PLANT.PIPEH.PSI(35) := 5.03874e+006 ; # 1122
1137   PLANT.PIPEH.PSI(36) := 4.98752e+006 ; # 1123
1138   PLANT.PIPEH.PSI(37) := 4.93419e+006 ; # 1124
1139   PLANT.PIPEH.PSI(38) := 4.8786e+006 ; # 1125
1140   PLANT.PIPEH.PSI(39) := 4.82064e+006 ; # 1126
1141   PLANT.PIPEH.PSI(40) := 4.76024e+006 ; # 1127
1142   PLANT.PIPEH.HT.RHOPIPEHE := 3.1094 ; # 1128
1143   PLANT.PIPEH.HT.VPIPE := 28.7014 ; # 1129
1144   PLANT.PIPEH.HT.H0 := 1760.94 ; # 1130
1145   PLANT.PIPEH.HT.U := 1738.41 ; # 1131
1146   PLANT.HX2.TGASHOT(1) := 1138.77 ; # 1132
1147   PLANT.HX2.TGASHOT(2) := 1136.79 ; # 1133
1148   PLANT.HX2.TGASHOT(3) := 1134.68 ; # 1134
1149   PLANT.HX2.TGASHOT(4) := 1132.42 ; # 1135
1150   PLANT.HX2.TGASHOT(5) := 1130.02 ; # 1136
1151   PLANT.HX2.TGASHOT(6) := 1127.46 ; # 1137
1152   PLANT.HX2.TGASHOT(7) := 1124.73 ; # 1138
1153   PLANT.HX2.TGASHOT(8) := 1121.82 ; # 1139
1154   PLANT.HX2.TGASHOT(9) := 1118.71 ; # 1140
1155   PLANT.HX2.TGASHOT(10) := 1115.4 ; # 1141
1156   PLANT.HX2.TGASHOT(11) := 1111.87 ; # 1142
1157   PLANT.HX2.TGASHOT(12) := 1108.11 ; # 1143
1158   PLANT.HX2.TGASHOT(13) := 1104.09 ; # 1144
1159   PLANT.HX2.TGASHOT(14) := 1099.81 ; # 1145
1160   PLANT.HX2.TGASHOT(15) := 1095.25 ; # 1146
1161   PLANT.HX2.TGASHOT(16) := 1090.38 ; # 1147
1162   PLANT.HX2.TGASHOT(17) := 1085.19 ; # 1148
```

```
PLANT.HX2.TGASHOT(18) := 1079.65 ; # 1149
PLANT.HX2.TGASHOT(19) := 1073.75 ; # 1150
PLANT.HX2.TGASHOT(20) := 1067.46 ; # 1151
PLANT.HX2.TGASHOT(21) := 1060.75 ; # 1152
PLANT.HX2.TGASHOT(22) := 1053.59 ; # 1153
PLANT.HX2.TGASHOT(23) := 1045.96 ; # 1154
PLANT.HX2.TGASHOT(24) := 1037.83 ; # 1155
PLANT.HX2.TGASHOT(25) := 1029.15 ; # 1156
PLANT.HX2.TGASHOT(26) := 1019.9 ; # 1157
PLANT.HX2.TGASHOT(27) := 1010.03 ; # 1158
PLANT.HX2.TGASHOT(28) := 999.513 ; # 1159
PLANT.HX2.TGASHOT(29) := 988.296 ; # 1160
PLANT.HX2.TGASHOT(30) := 976.334 ; # 1161
PLANT.HX2.TGASHOT(31) := 963.58 ; # 1162
PLANT.HX2.TGASHOT(32) := 949.979 ; # 1163
PLANT.HX2.TGASHOT(33) := 935.476 ; # 1164
PLANT.HX2.TGASHOT(34) := 920.011 ; # 1165
PLANT.HX2.TGASHOT(35) := 903.52 ; # 1166
PLANT.HX2.TGASHOT(36) := 885.936 ; # 1167
PLANT.HX2.TGASHOT(37) := 867.185 ; # 1168
PLANT.HX2.TGASHOT(38) := 847.191 ; # 1169
PLANT.HX2.TGASHOT(39) := 825.87 ; # 1170
PLANT.HX2.TGASHOT(40) := 803.137 ; # 1171
PLANT.HX2.TGASCOLD(1) := 1126.15 ; # 1172
PLANT.HX2.TGASCOLD(2) := 1123.33 ; # 1173
PLANT.HX2.TGASCOLD(3) := 1120.32 ; # 1174
PLANT.HX2.TGASCOLD(4) := 1117.12 ; # 1175
PLANT.HX2.TGASCOLD(5) := 1113.7 ; # 1176
PLANT.HX2.TGASCOLD(6) := 1110.06 ; # 1177
PLANT.HX2.TGASCOLD(7) := 1106.17 ; # 1178
PLANT.HX2.TGASCOLD(8) := 1102.03 ; # 1179
PLANT.HX2.TGASCOLD(9) := 1097.61 ; # 1180
PLANT.HX2.TGASCOLD(10) := 1092.9 ; # 1181
PLANT.HX2.TGASCOLD(11) := 1087.88 ; # 1182
PLANT.HX2.TGASCOLD(12) := 1082.52 ; # 1183
PLANT.HX2.TGASCOLD(13) := 1076.81 ; # 1184
PLANT.HX2.TGASCOLD(14) := 1070.72 ; # 1185
PLANT.HX2.TGASCOLD(15) := 1064.23 ; # 1186
PLANT.HX2.TGASCOLD(16) := 1057.3 ; # 1187
PLANT.HX2.TGASCOLD(17) := 1049.92 ; # 1188
PLANT.HX2.TGASCOLD(18) := 1042.04 ; # 1189
PLANT.HX2.TGASCOLD(19) := 1033.65 ; # 1190
PLANT.HX2.TGASCOLD(20) := 1024.69 ; # 1191
PLANT.HX2.TGASCOLD(21) := 1015.15 ; # 1192
PLANT.HX2.TGASCOLD(22) := 1004.97 ; # 1193
PLANT.HX2.TGASCOLD(23) := 994.111 ; # 1194
PLANT.HX2.TGASCOLD(24) := 982.535 ; # 1195
PLANT.HX2.TGASCOLD(25) := 970.192 ; # 1196
PLANT.HX2.TGASCOLD(26) := 957.03 ; # 1197
PLANT.HX2.TGASCOLD(27) := 942.994 ; # 1198
PLANT.HX2.TGASCOLD(28) := 928.028 ; # 1199
PLANT.HX2.TGASCOLD(29) := 912.069 ; # 1200
PLANT.HX2.TGASCOLD(30) := 895.052 ; # 1201
PLANT.HX2.TGASCOLD(31) := 876.906 ; # 1202
PLANT.HX2.TGASCOLD(32) := 857.556 ; # 1203
PLANT.HX2.TGASCOLD(33) := 836.923 ; # 1204
PLANT.HX2.TGASCOLD(34) := 814.921 ; # 1205
PLANT.HX2.TGASCOLD(35) := 791.46 ; # 1206
PLANT.HX2.TGASCOLD(36) := 766.443 ; # 1207
PLANT.HX2.TGASCOLD(37) := 739.767 ; # 1208
PLANT.HX2.TGASCOLD(38) := 711.321 ; # 1209
PLANT.HX2.TGASCOLD(39) := 680.988 ; # 1210
PLANT.HX2.TGASCOLD(40) := 648.643 ; # 1211
PLANT.HX2.UHOT := 1299.78 ; # 1212
PLANT.HX2.UCOLD := 1880.53 ; # 1213
PLANT.HX2.THX(1) := 1132.12 ; # 1214
PLANT.HX2.THX(2) := 1129.7 ; # 1215
PLANT.HX2.THX(3) := 1127.12 ; # 1216
PLANT.HX2.THX(4) := 1124.36 ; # 1217
PLANT.HX2.THX(5) := 1121.43 ; # 1218
PLANT.HX2.THX(6) := 1118.29 ; # 1219
PLANT.HX2.THX(7) := 1114.96 ; # 1220
PLANT.HX2.THX(8) := 1111.4 ; # 1221
PLANT.HX2.THX(9) := 1107.6 ; # 1222
PLANT.HX2.THX(10) := 1103.55 ; # 1223
PLANT.HX2.THX(11) := 1099.23 ; # 1224
PLANT.HX2.THX(12) := 1094.63 ; # 1225
PLANT.HX2.THX(13) := 1089.72 ; # 1226
PLANT.HX2.THX(14) := 1084.49 ; # 1227
PLANT.HX2.THX(15) := 1078.91 ; # 1228
PLANT.HX2.THX(16) := 1072.96 ; # 1229
PLANT.HX2.THX(17) := 1066.61 ; # 1230
PLANT.HX2.THX(18) := 1059.84 ; # 1231
PLANT.HX2.THX(19) := 1052.63 ; # 1232
PLANT.HX2.THX(20) := 1044.93 ; # 1233
PLANT.HX2.THX(21) := 1036.73 ; # 1234
PLANT.HX2.THX(22) := 1027.98 ; # 1235
PLANT.HX2.THX(23) := 1018.65 ; # 1236
PLANT.HX2.THX(24) := 1008.7 ; # 1237
PLANT.HX2.THX(25) := 998.095 ; # 1238
PLANT.HX2.THX(26) := 986.784 ; # 1239
PLANT.HX2.THX(27) := 974.722 ; # 1240
```

```
PLANT.HX2.THX(28) := 961.86 ; # 1241
PLANT.HX2.THX(29) := 948.145 ; # 1242
PLANT.HX2.THX(30) := 933.521 ; # 1243
PLANT.HX2.THX(31) := 917.926 ; # 1244
PLANT.HX2.THX(32) := 901.297 ; # 1245
PLANT.HX2.THX(33) := 883.565 ; # 1246
PLANT.HX2.THX(34) := 864.657 ; # 1247
PLANT.HX2.THX(35) := 844.495 ; # 1248
PLANT.HX2.THX(36) := 822.996 ; # 1249
PLANT.HX2.THX(37) := 800.07 ; # 1250
PLANT.HX2.THX(38) := 775.624 ; # 1251
PLANT.HX2.THX(39) := 749.557 ; # 1252
PLANT.HX2.THX(40) := 721.768 ; # 1253
PLANT.PIPEC2.RHO(1) := 3.7063 ; # 1254
PLANT.PIPEC2.RHO(2) := 3.52381 ; # 1255
PLANT.PIPEC2.RHO(3) := 3.36718 ; # 1256
PLANT.PIPEC2.RHO(4) := 3.2314 ; # 1257
PLANT.PIPEC2.RHO(5) := 3.11269 ; # 1258
PLANT.PIPEC2.RHO(6) := 3.0081 ; # 1259
PLANT.PIPEC2.RHO(7) := 2.91535 ; # 1260
PLANT.PIPEC2.RHO(8) := 2.83259 ; # 1261
PLANT.PIPEC2.RHO(9) := 2.75835 ; # 1262
PLANT.PIPEC2.RHO(10) := 2.69142 ; # 1263
PLANT.PIPEC2.RHO(11) := 2.63081 ; # 1264
PLANT.PIPEC2.RHO(12) := 2.5757 ; # 1265
PLANT.PIPEC2.RHO(13) := 2.52539 ; # 1266
PLANT.PIPEC2.RHO(14) := 2.47931 ; # 1267
PLANT.PIPEC2.RHO(15) := 2.43696 ; # 1268
PLANT.PIPEC2.RHO(16) := 2.39791 ; # 1269
PLANT.PIPEC2.RHO(17) := 2.3618 ; # 1270
PLANT.PIPEC2.RHO(18) := 2.32832 ; # 1271
PLANT.PIPEC2.RHO(19) := 2.29719 ; # 1272
PLANT.PIPEC2.RHO(20) := 2.26818 ; # 1273
PLANT.PIPEC2.RHO(21) := 2.24106 ; # 1274
PLANT.PIPEC2.RHO(22) := 2.21566 ; # 1275
PLANT.PIPEC2.RHO(23) := 2.19182 ; # 1276
PLANT.PIPEC2.RHO(24) := 2.16939 ; # 1277
PLANT.PIPEC2.RHO(25) := 2.14823 ; # 1278
PLANT.PIPEC2.RHO(26) := 2.12824 ; # 1279
PLANT.PIPEC2.RHO(27) := 2.10932 ; # 1280
PLANT.PIPEC2.RHO(28) := 2.09136 ; # 1281
PLANT.PIPEC2.RHO(29) := 2.07429 ; # 1282
PLANT.PIPEC2.RHO(30) := 2.05802 ; # 1283
PLANT.PIPEC2.RHO(31) := 2.0425 ; # 1284
PLANT.PIPEC2.RHO(32) := 2.02766 ; # 1285
PLANT.PIPEC2.RHO(33) := 2.01345 ; # 1286
PLANT.PIPEC2.RHO(34) := 1.99981 ; # 1287
PLANT.PIPEC2.RHO(35) := 1.9867 ; # 1288
PLANT.PIPEC2.RHO(36) := 1.97407 ; # 1289
PLANT.PIPEC2.RHO(37) := 1.96189 ; # 1290
PLANT.PIPEC2.RHO(38) := 1.95011 ; # 1291
PLANT.PIPEC2.RHO(39) := 1.93872 ; # 1292
PLANT.PIPEC2.RHO(40) := 1.92768 ; # 1293
PLANT.PIPEC2.P(1) := 1.10918e+006 ; # 1294
PLANT.PIPEC2.P(2) := 1.10715e+006 ; # 1295
PLANT.PIPEC2.P(3) := 1.10506e+006 ; # 1296
PLANT.PIPEC2.P(4) := 1.10291e+006 ; # 1297
PLANT.PIPEC2.P(5) := 1.1007e+006 ; # 1298
PLANT.PIPEC2.P(6) := 1.09844e+006 ; # 1299
PLANT.PIPEC2.P(7) := 1.09613e+006 ; # 1300
PLANT.PIPEC2.P(8) := 1.09377e+006 ; # 1301
PLANT.PIPEC2.P(9) := 1.09136e+006 ; # 1302
PLANT.PIPEC2.P(10) := 1.0889e+006 ; # 1303
PLANT.PIPEC2.P(11) := 1.08641e+006 ; # 1304
PLANT.PIPEC2.P(12) := 1.08387e+006 ; # 1305
PLANT.PIPEC2.P(13) := 1.0813e+006 ; # 1306
PLANT.PIPEC2.P(14) := 1.07869e+006 ; # 1307
PLANT.PIPEC2.P(15) := 1.07604e+006 ; # 1308
PLANT.PIPEC2.P(16) := 1.07336e+006 ; # 1309
PLANT.PIPEC2.P(17) := 1.07065e+006 ; # 1310
PLANT.PIPEC2.P(18) := 1.06791e+006 ; # 1311
PLANT.PIPEC2.P(19) := 1.06513e+006 ; # 1312
PLANT.PIPEC2.P(20) := 1.06233e+006 ; # 1313
PLANT.PIPEC2.P(21) := 1.05951e+006 ; # 1314
PLANT.PIPEC2.P(22) := 1.05665e+006 ; # 1315
PLANT.PIPEC2.P(23) := 1.05377e+006 ; # 1316
PLANT.PIPEC2.P(24) := 1.05087e+006 ; # 1317
PLANT.PIPEC2.P(25) := 1.04794e+006 ; # 1318
PLANT.PIPEC2.P(26) := 1.04499e+006 ; # 1319
PLANT.PIPEC2.P(27) := 1.04201e+006 ; # 1320
PLANT.PIPEC2.P(28) := 1.03902e+006 ; # 1321
PLANT.PIPEC2.P(29) := 1.036e+006 ; # 1322
PLANT.PIPEC2.P(30) := 1.03297e+006 ; # 1323
PLANT.PIPEC2.P(31) := 1.02991e+006 ; # 1324
PLANT.PIPEC2.P(32) := 1.02684e+006 ; # 1325
PLANT.PIPEC2.P(33) := 1.02374e+006 ; # 1326
PLANT.PIPEC2.P(34) := 1.02063e+006 ; # 1327
PLANT.PIPEC2.P(35) := 1.0175e+006 ; # 1328
PLANT.PIPEC2.P(36) := 1.01435e+006 ; # 1329
PLANT.PIPEC2.P(37) := 1.01118e+006 ; # 1330
PLANT.PIPEC2.P(38) := 1.008e+006 ; # 1331
PLANT.PIPEC2.P(39) := 1.00479e+006 ; # 1332
```

```
1347  PLANT.PIPEC2.P(40) := 1.00158e+006 ; # 1333
1348  PLANT.PIPEC2.H(1) := 790991 ; # 1334
1349  PLANT.PIPEC2.H(2) := 863955 ; # 1335
1350  PLANT.PIPEC2.H(3) := 932380 ; # 1336
1351  PLANT.PIPEC2.H(4) := 996549 ; # 1337
1352  PLANT.PIPEC2.H(5) := 1.05673e+006 ; # 1338
1353  PLANT.PIPEC2.H(6) := 1.11316e+006 ; # 1339
1354  PLANT.PIPEC2.H(7) := 1.16608e+006 ; # 1340
1355  PLANT.PIPEC2.H(8) := 1.21572e+006 ; # 1341
1356  PLANT.PIPEC2.H(9) := 1.26226e+006 ; # 1342
1357  PLANT.PIPEC2.H(10) := 1.30591e+006 ; # 1343
1358  PLANT.PIPEC2.H(11) := 1.34685e+006 ; # 1344
1359  PLANT.PIPEC2.H(12) := 1.38523e+006 ; # 1345
1360  PLANT.PIPEC2.H(13) := 1.42123e+006 ; # 1346
1361  PLANT.PIPEC2.H(14) := 1.455e+006 ; # 1347
1362  PLANT.PIPEC2.H(15) := 1.48666e+006 ; # 1348
1363  PLANT.PIPEC2.H(16) := 1.51635e+006 ; # 1349
1364  PLANT.PIPEC2.H(17) := 1.54419e+006 ; # 1350
1365  PLANT.PIPEC2.H(18) := 1.57031e+006 ; # 1351
1366  PLANT.PIPEC2.H(19) := 1.59479e+006 ; # 1352
1367  PLANT.PIPEC2.H(20) := 1.61776e+006 ; # 1353
1368  PLANT.PIPEC2.H(21) := 1.6393e+006 ; # 1354
1369  PLANT.PIPEC2.H(22) := 1.65949e+006 ; # 1355
1370  PLANT.PIPEC2.H(23) := 1.67843e+006 ; # 1356
1371  PLANT.PIPEC2.H(24) := 1.6962e+006 ; # 1357
1372  PLANT.PIPEC2.H(25) := 1.71285e+006 ; # 1358
1373  PLANT.PIPEC2.H(26) := 1.72848e+006 ; # 1359
1374  PLANT.PIPEC2.H(27) := 1.74313e+006 ; # 1360
1375  PLANT.PIPEC2.H(28) := 1.75686e+006 ; # 1361
1376  PLANT.PIPEC2.H(29) := 1.76975e+006 ; # 1362
1377  PLANT.PIPEC2.H(30) := 1.78183e+006 ; # 1363
1378  PLANT.PIPEC2.H(31) := 1.79316e+006 ; # 1364
1379  PLANT.PIPEC2.H(32) := 1.80379e+006 ; # 1365
1380  PLANT.PIPEC2.H(33) := 1.81375e+006 ; # 1366
1381  PLANT.PIPEC2.H(34) := 1.8231e+006 ; # 1367
1382  PLANT.PIPEC2.H(35) := 1.83186e+006 ; # 1368
1383  PLANT.PIPEC2.H(36) := 1.84008e+006 ; # 1369
1384  PLANT.PIPEC2.H(37) := 1.84779e+006 ; # 1370
1385  PLANT.PIPEC2.H(38) := 1.85502e+006 ; # 1371
1386  PLANT.PIPEC2.H(39) := 1.8618e+006 ; # 1372
1387  PLANT.PIPEC2.H(40) := 1.86815e+006 ; # 1373
1388  PLANT.PIPEC2.H0 := 713180 ; # 1374
1389  PLANT.PIPEC2.T(1) := 648.643 ; # 1375
1390  PLANT.PIPEC2.T(2) := 680.988 ; # 1376
1391  PLANT.PIPEC2.T(3) := 711.321 ; # 1377
1392  PLANT.PIPEC2.T(4) := 739.767 ; # 1378
1393  PLANT.PIPEC2.T(5) := 766.443 ; # 1379
1394  PLANT.PIPEC2.T(6) := 791.46 ; # 1380
1395  PLANT.PIPEC2.T(7) := 814.921 ; # 1381
1396  PLANT.PIPEC2.T(8) := 836.923 ; # 1382
1397  PLANT.PIPEC2.T(9) := 857.556 ; # 1383
1398  PLANT.PIPEC2.T(10) := 876.906 ; # 1384
1399  PLANT.PIPEC2.T(11) := 895.052 ; # 1385
1400  PLANT.PIPEC2.T(12) := 912.069 ; # 1386
1401  PLANT.PIPEC2.T(13) := 928.028 ; # 1387
1402  PLANT.PIPEC2.T(14) := 942.994 ; # 1388
1403  PLANT.PIPEC2.T(15) := 957.03 ; # 1389
1404  PLANT.PIPEC2.T(16) := 970.192 ; # 1390
1405  PLANT.PIPEC2.T(17) := 982.535 ; # 1391
1406  PLANT.PIPEC2.T(18) := 994.111 ; # 1392
1407  PLANT.PIPEC2.T(19) := 1004.97 ; # 1393
1408  PLANT.PIPEC2.T(20) := 1015.15 ; # 1394
1409  PLANT.PIPEC2.T(21) := 1024.69 ; # 1395
1410  PLANT.PIPEC2.T(22) := 1033.65 ; # 1396
1411  PLANT.PIPEC2.T(23) := 1042.04 ; # 1397
1412  PLANT.PIPEC2.T(24) := 1049.92 ; # 1398
1413  PLANT.PIPEC2.T(25) := 1057.3 ; # 1399
1414  PLANT.PIPEC2.T(26) := 1064.23 ; # 1400
1415  PLANT.PIPEC2.T(27) := 1070.72 ; # 1401
1416  PLANT.PIPEC2.T(28) := 1076.81 ; # 1402
1417  PLANT.PIPEC2.T(29) := 1082.52 ; # 1403
1418  PLANT.PIPEC2.T(30) := 1087.88 ; # 1404
1419  PLANT.PIPEC2.T(31) := 1092.9 ; # 1405
1420  PLANT.PIPEC2.T(32) := 1097.61 ; # 1406
1421  PLANT.PIPEC2.T(33) := 1102.03 ; # 1407
1422  PLANT.PIPEC2.T(34) := 1106.17 ; # 1408
1423  PLANT.PIPEC2.T(35) := 1110.06 ; # 1409
1424  PLANT.PIPEC2.T(36) := 1113.7 ; # 1410
1425  PLANT.PIPEC2.T(37) := 1117.12 ; # 1411
1426  PLANT.PIPEC2.T(38) := 1120.32 ; # 1412
1427  PLANT.PIPEC2.T(39) := 1123.33 ; # 1413
1428  PLANT.PIPEC2.T(40) := 1126.15 ; # 1414
1429  PLANT.PIPEC2.TEXTRNL(1) := 721.768 ; # 1415
1430  PLANT.PIPEC2.TEXTRNL(2) := 749.557 ; # 1416
1431  PLANT.PIPEC2.TEXTRNL(3) := 775.624 ; # 1417
1432  PLANT.PIPEC2.TEXTRNL(4) := 800.07 ; # 1418
1433  PLANT.PIPEC2.TEXTRNL(5) := 822.996 ; # 1419
1434  PLANT.PIPEC2.TEXTRNL(6) := 844.495 ; # 1420
1435  PLANT.PIPEC2.TEXTRNL(7) := 864.657 ; # 1421
1436  PLANT.PIPEC2.TEXTRNL(8) := 883.565 ; # 1422
1437  PLANT.PIPEC2.TEXTRNL(9) := 901.297 ; # 1423
1438  PLANT.PIPEC2.TEXTRNL(10) := 917.926 ; # 1424
```

```
1439   PLANT.PIPEC2.TEXTRNL(11) := 933.521 ; # 1425
1440   PLANT.PIPEC2.TEXTRNL(12) := 948.145 ; # 1426
1441   PLANT.PIPEC2.TEXTRNL(13) := 961.86 ; # 1427
1442   PLANT.PIPEC2.TEXTRNL(14) := 974.722 ; # 1428
1443   PLANT.PIPEC2.TEXTRNL(15) := 986.784 ; # 1429
1444   PLANT.PIPEC2.TEXTRNL(16) := 998.095 ; # 1430
1445   PLANT.PIPEC2.TEXTRNL(17) := 1008.7 ; # 1431
1446   PLANT.PIPEC2.TEXTRNL(18) := 1018.65 ; # 1432
1447   PLANT.PIPEC2.TEXTRNL(19) := 1027.98 ; # 1433
1448   PLANT.PIPEC2.TEXTRNL(20) := 1036.73 ; # 1434
1449   PLANT.PIPEC2.TEXTRNL(21) := 1044.93 ; # 1435
1450   PLANT.PIPEC2.TEXTRNL(22) := 1052.63 ; # 1436
1451   PLANT.PIPEC2.TEXTRNL(23) := 1059.84 ; # 1437
1452   PLANT.PIPEC2.TEXTRNL(24) := 1066.61 ; # 1438
1453   PLANT.PIPEC2.TEXTRNL(25) := 1072.96 ; # 1439
1454   PLANT.PIPEC2.TEXTRNL(26) := 1078.91 ; # 1440
1455   PLANT.PIPEC2.TEXTRNL(27) := 1084.49 ; # 1441
1456   PLANT.PIPEC2.TEXTRNL(28) := 1089.72 ; # 1442
1457   PLANT.PIPEC2.TEXTRNL(29) := 1094.63 ; # 1443
1458   PLANT.PIPEC2.TEXTRNL(30) := 1099.23 ; # 1444
1459   PLANT.PIPEC2.TEXTRNL(31) := 1103.55 ; # 1445
1460   PLANT.PIPEC2.TEXTRNL(32) := 1107.6 ; # 1446
1461   PLANT.PIPEC2.TEXTRNL(33) := 1111.4 ; # 1447
1462   PLANT.PIPEC2.TEXTRNL(34) := 1114.96 ; # 1448
1463   PLANT.PIPEC2.TEXTRNL(35) := 1118.29 ; # 1449
1464   PLANT.PIPEC2.TEXTRNL(36) := 1121.43 ; # 1450
1465   PLANT.PIPEC2.TEXTRNL(37) := 1124.36 ; # 1451
1466   PLANT.PIPEC2.TEXTRNL(38) := 1127.12 ; # 1452
1467   PLANT.PIPEC2.TEXTRNL(39) := 1129.7 ; # 1453
1468   PLANT.PIPEC2.TEXTRNL(40) := 1132.12 ; # 1454
1469   PLANT.PIPEC2.F(1) := 43.1466 ; # 1455
1470   PLANT.PIPEC2.F(2) := 43.1466 ; # 1456
1471   PLANT.PIPEC2.F(3) := 43.1466 ; # 1457
1472   PLANT.PIPEC2.F(4) := 43.1466 ; # 1458
1473   PLANT.PIPEC2.F(5) := 43.1466 ; # 1459
1474   PLANT.PIPEC2.F(6) := 43.1466 ; # 1460
1475   PLANT.PIPEC2.F(7) := 43.1466 ; # 1461
1476   PLANT.PIPEC2.F(8) := 43.1466 ; # 1462
1477   PLANT.PIPEC2.F(9) := 43.1466 ; # 1463
1478   PLANT.PIPEC2.F(10) := 43.1466 ; # 1464
1479   PLANT.PIPEC2.F(11) := 43.1466 ; # 1465
1480   PLANT.PIPEC2.F(12) := 43.1466 ; # 1466
1481   PLANT.PIPEC2.F(13) := 43.1466 ; # 1467
1482   PLANT.PIPEC2.F(14) := 43.1466 ; # 1468
1483   PLANT.PIPEC2.F(15) := 43.1466 ; # 1469
1484   PLANT.PIPEC2.F(16) := 43.1466 ; # 1470
1485   PLANT.PIPEC2.F(17) := 43.1466 ; # 1471
1486   PLANT.PIPEC2.F(18) := 43.1466 ; # 1472
1487   PLANT.PIPEC2.F(19) := 43.1466 ; # 1473
1488   PLANT.PIPEC2.F(20) := 43.1466 ; # 1474
1489   PLANT.PIPEC2.F(21) := 43.1466 ; # 1475
1490   PLANT.PIPEC2.F(22) := 43.1466 ; # 1476
1491   PLANT.PIPEC2.F(23) := 43.1466 ; # 1477
1492   PLANT.PIPEC2.F(24) := 43.1466 ; # 1478
1493   PLANT.PIPEC2.F(25) := 43.1466 ; # 1479
1494   PLANT.PIPEC2.F(26) := 43.1466 ; # 1480
1495   PLANT.PIPEC2.F(27) := 43.1466 ; # 1481
1496   PLANT.PIPEC2.F(28) := 43.1466 ; # 1482
1497   PLANT.PIPEC2.F(29) := 43.1466 ; # 1483
1498   PLANT.PIPEC2.F(30) := 43.1466 ; # 1484
1499   PLANT.PIPEC2.F(31) := 43.1466 ; # 1485
1500   PLANT.PIPEC2.F(32) := 43.1466 ; # 1486
1501   PLANT.PIPEC2.F(33) := 43.1466 ; # 1487
1502   PLANT.PIPEC2.F(34) := 43.1466 ; # 1488
1503   PLANT.PIPEC2.F(35) := 43.1466 ; # 1489
1504   PLANT.PIPEC2.F(36) := 43.1466 ; # 1490
1505   PLANT.PIPEC2.F(37) := 43.1466 ; # 1491
1506   PLANT.PIPEC2.F(38) := 43.1466 ; # 1492
1507   PLANT.PIPEC2.F(39) := 43.1466 ; # 1493
1508   PLANT.PIPEC2.F(40) := 43.1466 ; # 1494
1509   PLANT.PIPEC2.F_IN := 43.1466 ; # 1495
1510   PLANT.PIPEC2.PSI(1) := 1.82247e+006 ; # 1496
1511   PLANT.PIPEC2.PSI(2) := 1.93726e+006 ; # 1497
1512   PLANT.PIPEC2.PSI(3) := 2.03443e+006 ; # 1498
1513   PLANT.PIPEC2.PSI(4) := 2.11734e+006 ; # 1499
1514   PLANT.PIPEC2.PSI(5) := 2.18855e+006 ; # 1500
1515   PLANT.PIPEC2.PSI(6) := 2.25006e+006 ; # 1501
1516   PLANT.PIPEC2.PSI(7) := 2.30342e+006 ; # 1502
1517   PLANT.PIPEC2.PSI(8) := 2.34986e+006 ; # 1503
1518   PLANT.PIPEC2.PSI(9) := 2.3904e+006 ; # 1504
1519   PLANT.PIPEC2.PSI(10) := 2.42585e+006 ; # 1505
1520   PLANT.PIPEC2.PSI(11) := 2.45689e+006 ; # 1506
1521   PLANT.PIPEC2.PSI(12) := 2.48407e+006 ; # 1507
1522   PLANT.PIPEC2.PSI(13) := 2.50788e+006 ; # 1508
1523   PLANT.PIPEC2.PSI(14) := 2.5287e+006 ; # 1509
1524   PLANT.PIPEC2.PSI(15) := 2.54688e+006 ; # 1510
1525   PLANT.PIPEC2.PSI(16) := 2.56271e+006 ; # 1511
1526   PLANT.PIPEC2.PSI(17) := 2.57643e+006 ; # 1512
1527   PLANT.PIPEC2.PSI(18) := 2.58827e+006 ; # 1513
1528   PLANT.PIPEC2.PSI(19) := 2.59842e+006 ; # 1514
1529   PLANT.PIPEC2.PSI(20) := 2.60703e+006 ; # 1515
1530   PLANT.PIPEC2.PSI(21) := 2.61426e+006 ; # 1516
```

```
PLANT.PIPEC2.PSI(22) := 2.62023e+006 ; # 1517
PLANT.PIPEC2.PSI(23) := 2.62506e+006 ; # 1518
PLANT.PIPEC2.PSI(24) := 2.62884e+006 ; # 1519
PLANT.PIPEC2.PSI(25) := 2.63167e+006 ; # 1520
PLANT.PIPEC2.PSI(26) := 2.63363e+006 ; # 1521
PLANT.PIPEC2.PSI(27) := 2.63479e+006 ; # 1522
PLANT.PIPEC2.PSI(28) := 2.63522e+006 ; # 1523
PLANT.PIPEC2.PSI(29) := 2.63496e+006 ; # 1524
PLANT.PIPEC2.PSI(30) := 2.63409e+006 ; # 1525
PLANT.PIPEC2.PSI(31) := 2.63263e+006 ; # 1526
PLANT.PIPEC2.PSI(32) := 2.63064e+006 ; # 1527
PLANT.PIPEC2.PSI(33) := 2.62816e+006 ; # 1528
PLANT.PIPEC2.PSI(34) := 2.62522e+006 ; # 1529
PLANT.PIPEC2.PSI(35) := 2.62186e+006 ; # 1530
PLANT.PIPEC2.PSI(36) := 2.6181e+006 ; # 1531
PLANT.PIPEC2.PSI(37) := 2.61397e+006 ; # 1532
PLANT.PIPEC2.PSI(38) := 2.6095e+006 ; # 1533
PLANT.PIPEC2.PSI(39) := 2.60471e+006 ; # 1534
PLANT.PIPEC2.PSI(40) := 2.59962e+006 ; # 1535
PLANT.PIPEC2.HT.RHOPIPEHE := 2.32481 ; # 1536
PLANT.PIPEC2.HT.VPIPE := 65.9555 ; # 1537
PLANT.PIPEC2.HT.U := 1880.53 ; # 1538
```

HTLNOPIPEHQSS_SS_OUT_INI.INITIAL

```
# Values for differential variables in computation HTLQSS_SS
# Saved at time 2120
PLANT.HX.THX(1) = 875.997 ; # 1
PLANT.HX.THX(2) = 887.105 ; # 2
PLANT.HX.THX(3) = 898.003 ; # 3
PLANT.HX.THX(4) = 908.635 ; # 4
PLANT.HX.THX(5) = 919.007 ; # 5
PLANT.HX.THX(6) = 929.126 ; # 6
PLANT.HX.THX(7) = 938.998 ; # 7
PLANT.HX.THX(8) = 948.629 ; # 8
PLANT.HX.THX(9) = 958.024 ; # 9
PLANT.HX.THX(10) = 967.19 ; # 10
PLANT.HX.THX(11) = 976.133 ; # 11
PLANT.HX.THX(12) = 984.857 ; # 12
PLANT.HX.THX(13) = 993.368 ; # 13
PLANT.HX.THX(14) = 1001.67 ; # 14
PLANT.HX.THX(15) = 1009.77 ; # 15
PLANT.HX.THX(16) = 1017.67 ; # 16
PLANT.HX.THX(17) = 1025.38 ; # 17
PLANT.HX.THX(18) = 1032.9 ; # 18
PLANT.HX.THX(19) = 1040.24 ; # 19
PLANT.HX.THX(20) = 1047.4 ; # 20
PLANT.HX.THX(21) = 1054.38 ; # 21
PLANT.HX.THX(22) = 1061.2 ; # 22
PLANT.HX.THX(23) = 1067.84 ; # 23
PLANT.HX.THX(24) = 1074.33 ; # 24
PLANT.HX.THX(25) = 1080.65 ; # 25
PLANT.HX.THX(26) = 1086.83 ; # 26
PLANT.HX.THX(27) = 1092.85 ; # 27
PLANT.HX.THX(28) = 1098.72 ; # 28
PLANT.HX.THX(29) = 1104.45 ; # 29
PLANT.HX.THX(30) = 1110.04 ; # 30
PLANT.HX.THX(31) = 1115.49 ; # 31
PLANT.HX.THX(32) = 1120.82 ; # 32
PLANT.HX.THX(33) = 1126.01 ; # 33
PLANT.HX.THX(34) = 1131.07 ; # 34
PLANT.HX.THX(35) = 1136.01 ; # 35
PLANT.HX.THX(36) = 1140.83 ; # 36
PLANT.HX.THX(37) = 1145.53 ; # 37
PLANT.HX.THX(38) = 1150.12 ; # 38
PLANT.HX.THX(39) = 1154.59 ; # 39
PLANT.HX.THX(40) = 1158.94 ; # 40
PLANT.LOOP1.PSI(1) = 1.21853e+006 ; # 41
PLANT.LOOP1.PSI(2) = 1.21407e+006 ; # 42
PLANT.LOOP1.PSI(3) = 1.2096e+006 ; # 43
PLANT.LOOP1.PSI(4) = 1.20513e+006 ; # 44
PLANT.LOOP1.PSI(5) = 1.20066e+006 ; # 45
PLANT.LOOP1.PSI(6) = 1.19618e+006 ; # 46
PLANT.LOOP1.PSI(7) = 1.1917e+006 ; # 47
PLANT.LOOP1.PSI(8) = 1.18721e+006 ; # 48
PLANT.LOOP1.PSI(9) = 1.18271e+006 ; # 49
PLANT.LOOP1.PSI(10) = 1.17821e+006 ; # 50
PLANT.LOOP1.PSI(11) = 1.21421e+006 ; # 51
PLANT.LOOP1.PSI(12) = 1.23642e+006 ; # 52
PLANT.LOOP1.PSI(13) = 1.25693e+006 ; # 53
PLANT.LOOP1.PSI(14) = 1.27623e+006 ; # 54
PLANT.LOOP1.PSI(15) = 1.2944e+006 ; # 55
PLANT.LOOP1.PSI(16) = 1.3115e+006 ; # 56
```

```
PLANT.LOOP1.PSI(17) = 1.32762e+006 ; # 57
PLANT.LOOP1.PSI(18) = 1.3428e+006 ; # 58
PLANT.LOOP1.PSI(19) = 1.35711e+006 ; # 59
PLANT.LOOP1.PSI(20) = 1.37059e+006 ; # 60
PLANT.LOOP1.PSI(21) = 1.38329e+006 ; # 61
PLANT.LOOP1.PSI(22) = 1.39525e+006 ; # 62
PLANT.LOOP1.PSI(23) = 1.40653e+006 ; # 63
PLANT.LOOP1.PSI(24) = 1.41714e+006 ; # 64
PLANT.LOOP1.PSI(25) = 1.42714e+006 ; # 65
PLANT.LOOP1.PSI(26) = 1.43655e+006 ; # 66
PLANT.LOOP1.PSI(27) = 1.4454e+006 ; # 67
PLANT.LOOP1.PSI(28) = 1.45372e+006 ; # 68
PLANT.LOOP1.PSI(29) = 1.46153e+006 ; # 69
PLANT.LOOP1.PSI(30) = 1.46887e+006 ; # 70
PLANT.LOOP1.PSI(31) = 1.47574e+006 ; # 71
PLANT.LOOP1.PSI(32) = 1.48219e+006 ; # 72
PLANT.LOOP1.PSI(33) = 1.48821e+006 ; # 73
PLANT.LOOP1.PSI(34) = 1.49385e+006 ; # 74
PLANT.LOOP1.PSI(35) = 1.4991e+006 ; # 75
PLANT.LOOP1.PSI(36) = 1.50399e+006 ; # 76
PLANT.LOOP1.PSI(37) = 1.50854e+006 ; # 77
PLANT.LOOP1.PSI(38) = 1.51275e+006 ; # 78
PLANT.LOOP1.PSI(39) = 1.51665e+006 ; # 79
PLANT.LOOP1.PSI(40) = 1.52024e+006 ; # 80
PLANT.LOOP1.PSI(41) = 1.52354e+006 ; # 81
PLANT.LOOP1.PSI(42) = 1.52657e+006 ; # 82
PLANT.LOOP1.PSI(43) = 1.52932e+006 ; # 83
PLANT.LOOP1.PSI(44) = 1.53181e+006 ; # 84
PLANT.LOOP1.PSI(45) = 1.53406e+006 ; # 85
PLANT.LOOP1.PSI(46) = 1.53607e+006 ; # 86
PLANT.LOOP1.PSI(47) = 1.53785e+006 ; # 87
PLANT.LOOP1.PSI(48) = 1.53941e+006 ; # 88
PLANT.LOOP1.PSI(49) = 1.54075e+006 ; # 89
PLANT.LOOP1.PSI(50) = 1.54188e+006 ; # 90
PLANT.LOOP1.PSI(51) = 1.51428e+006 ; # 91
PLANT.LOOP1.PSI(52) = 1.50808e+006 ; # 92
PLANT.LOOP1.PSI(53) = 1.50192e+006 ; # 93
PLANT.LOOP1.PSI(54) = 1.49574e+006 ; # 94
PLANT.LOOP1.PSI(55) = 1.48956e+006 ; # 95
PLANT.LOOP1.PSI(56) = 1.48336e+006 ; # 96
PLANT.LOOP1.PSI(57) = 1.47715e+006 ; # 97
PLANT.LOOP1.PSI(58) = 1.47093e+006 ; # 98
PLANT.LOOP1.PSI(59) = 1.46469e+006 ; # 99
PLANT.LOOP1.PSI(60) = 1.45845e+006 ; # 100
PLANT.LOOP1.PSI(61) = 1.47351e+006 ; # 101
PLANT.LOOP1.PSI(62) = 1.46929e+006 ; # 102
PLANT.LOOP1.PSI(63) = 1.46481e+006 ; # 103
PLANT.LOOP1.PSI(64) = 1.46016e+006 ; # 104
PLANT.LOOP1.PSI(65) = 1.45533e+006 ; # 105
PLANT.LOOP1.PSI(66) = 1.4503e+006 ; # 106
PLANT.LOOP1.PSI(67) = 1.44506e+006 ; # 107
PLANT.LOOP1.PSI(68) = 1.4396e+006 ; # 108
PLANT.LOOP1.PSI(69) = 1.4339e+006 ; # 109
PLANT.LOOP1.PSI(70) = 1.42795e+006 ; # 110
PLANT.LOOP1.PSI(71) = 1.42173e+006 ; # 111
PLANT.LOOP1.PSI(72) = 1.41522e+006 ; # 112
PLANT.LOOP1.PSI(73) = 1.4084e+006 ; # 113
PLANT.LOOP1.PSI(74) = 1.40125e+006 ; # 114
PLANT.LOOP1.PSI(75) = 1.39374e+006 ; # 115
PLANT.LOOP1.PSI(76) = 1.38585e+006 ; # 116
PLANT.LOOP1.PSI(77) = 1.37755e+006 ; # 117
PLANT.LOOP1.PSI(78) = 1.36881e+006 ; # 118
PLANT.LOOP1.PSI(79) = 1.35959e+006 ; # 119
PLANT.LOOP1.PSI(80) = 1.34986e+006 ; # 120
PLANT.LOOP1.PSI(81) = 1.33957e+006 ; # 121
PLANT.LOOP1.PSI(82) = 1.32868e+006 ; # 122
PLANT.LOOP1.PSI(83) = 1.31713e+006 ; # 123
PLANT.LOOP1.PSI(84) = 1.30488e+006 ; # 124
PLANT.LOOP1.PSI(85) = 1.29185e+006 ; # 125
PLANT.LOOP1.PSI(86) = 1.27797e+006 ; # 126
PLANT.LOOP1.PSI(87) = 1.26318e+006 ; # 127
PLANT.LOOP1.PSI(88) = 1.24738e+006 ; # 128
PLANT.LOOP1.PSI(89) = 1.23046e+006 ; # 129
PLANT.LOOP1.PSI(90) = 1.21233e+006 ; # 130
PLANT.LOOP1.PSI(91) = 1.19284e+006 ; # 131
PLANT.LOOP1.PSI(92) = 1.17186e+006 ; # 132
PLANT.LOOP1.PSI(93) = 1.14922e+006 ; # 133
PLANT.LOOP1.PSI(94) = 1.12472e+006 ; # 134
PLANT.LOOP1.PSI(95) = 1.09815e+006 ; # 135
PLANT.LOOP1.PSI(96) = 1.06925e+006 ; # 136
PLANT.LOOP1.PSI(97) = 1.03771e+006 ; # 137
PLANT.LOOP1.PSI(98) = 1.00319e+006 ; # 138
PLANT.LOOP1.PSI(99) = 965262 ; # 139
PLANT.LOOP1.PSI(100) = 923433 ; # 140
PLANT.LOOP1.PSIEXTRA(1) = 1.23448e+006 ; # 141
PLANT.LOOP1.PSIEXTRA(2) = 1.23448e+006 ; # 142
PLANT.HX2.THX(1) = 1127.77 ; # 264
PLANT.HX2.THX(2) = 1124.33 ; # 265
PLANT.HX2.THX(3) = 1120.69 ; # 266
PLANT.HX2.THX(4) = 1116.87 ; # 267
PLANT.HX2.THX(5) = 1112.85 ; # 268
PLANT.HX2.THX(6) = 1108.61 ; # 269
```

```
PLANT.HX2.THX(7) = 1104.16 ; # 270
PLANT.HX2.THX(8) = 1099.46 ; # 271
PLANT.HX2.THX(9) = 1094.52 ; # 272
PLANT.HX2.THX(10) = 1089.32 ; # 273
PLANT.HX2.THX(11) = 1083.85 ; # 274
PLANT.HX2.THX(12) = 1078.09 ; # 275
PLANT.HX2.THX(13) = 1072.03 ; # 276
PLANT.HX2.THX(14) = 1065.64 ; # 277
PLANT.HX2.THX(15) = 1058.92 ; # 278
PLANT.HX2.THX(16) = 1051.85 ; # 279
PLANT.HX2.THX(17) = 1044.4 ; # 280
PLANT.HX2.THX(18) = 1036.57 ; # 281
PLANT.HX2.THX(19) = 1028.32 ; # 282
PLANT.HX2.THX(20) = 1019.63 ; # 283
PLANT.HX2.THX(21) = 1010.49 ; # 284
PLANT.HX2.THX(22) = 1000.87 ; # 285
PLANT.HX2.THX(23) = 990.74 ; # 286
PLANT.HX2.THX(24) = 980.078 ; # 287
PLANT.HX2.THX(25) = 968.854 ; # 288
PLANT.HX2.THX(26) = 957.041 ; # 289
PLANT.HX2.THX(27) = 944.605 ; # 290
PLANT.HX2.THX(28) = 931.514 ; # 291
PLANT.HX2.THX(29) = 917.735 ; # 292
PLANT.HX2.THX(30) = 903.231 ; # 293
PLANT.HX2.THX(31) = 887.962 ; # 294
PLANT.HX2.THX(32) = 871.891 ; # 295
PLANT.HX2.THX(33) = 854.973 ; # 296
PLANT.HX2.THX(34) = 837.165 ; # 297
PLANT.HX2.THX(35) = 818.42 ; # 298
PLANT.HX2.THX(36) = 798.688 ; # 299
PLANT.HX2.THX(37) = 777.917 ; # 300
PLANT.HX2.THX(38) = 756.054 ; # 301
PLANT.HX2.THX(39) = 733.039 ; # 302
PLANT.HX2.THX(40) = 708.82 ; # 303
PLANT.PIPEC2.PSI(1) = 1.81868e+006 ; # 385
PLANT.PIPEC2.PSI(2) = 1.91901e+006 ; # 386
PLANT.PIPEC2.PSI(3) = 2.00606e+006 ; # 387
PLANT.PIPEC2.PSI(4) = 2.08199e+006 ; # 388
PLANT.PIPEC2.PSI(5) = 2.14854e+006 ; # 389
PLANT.PIPEC2.PSI(6) = 2.20709e+006 ; # 390
PLANT.PIPEC2.PSI(7) = 2.25877e+006 ; # 391
PLANT.PIPEC2.PSI(8) = 2.30451e+006 ; # 392
PLANT.PIPEC2.PSI(9) = 2.34506e+006 ; # 393
PLANT.PIPEC2.PSI(10) = 2.38106e+006 ; # 394
PLANT.PIPEC2.PSI(11) = 2.41306e+006 ; # 395
PLANT.PIPEC2.PSI(12) = 2.44149e+006 ; # 396
PLANT.PIPEC2.PSI(13) = 2.46677e+006 ; # 397
PLANT.PIPEC2.PSI(14) = 2.48921e+006 ; # 398
PLANT.PIPEC2.PSI(15) = 2.50911e+006 ; # 399
PLANT.PIPEC2.PSI(16) = 2.52672e+006 ; # 400
PLANT.PIPEC2.PSI(17) = 2.54225e+006 ; # 401
PLANT.PIPEC2.PSI(18) = 2.55591e+006 ; # 402
PLANT.PIPEC2.PSI(19) = 2.56786e+006 ; # 403
PLANT.PIPEC2.PSI(20) = 2.57824e+006 ; # 404
PLANT.PIPEC2.PSI(21) = 2.5872e+006 ; # 405
PLANT.PIPEC2.PSI(22) = 2.59485e+006 ; # 406
PLANT.PIPEC2.PSI(23) = 2.60129e+006 ; # 407
PLANT.PIPEC2.PSI(24) = 2.60662e+006 ; # 408
PLANT.PIPEC2.PSI(25) = 2.61094e+006 ; # 409
PLANT.PIPEC2.PSI(26) = 2.6143e+006 ; # 410
PLANT.PIPEC2.PSI(27) = 2.61679e+006 ; # 411
PLANT.PIPEC2.PSI(28) = 2.61846e+006 ; # 412
PLANT.PIPEC2.PSI(29) = 2.61938e+006 ; # 413
PLANT.PIPEC2.PSI(30) = 2.6196e+006 ; # 414
PLANT.PIPEC2.PSI(31) = 2.61915e+006 ; # 415
PLANT.PIPEC2.PSI(32) = 2.6181e+006 ; # 416
PLANT.PIPEC2.PSI(33) = 2.61647e+006 ; # 417
PLANT.PIPEC2.PSI(34) = 2.6143e+006 ; # 418
PLANT.PIPEC2.PSI(35) = 2.61163e+006 ; # 419
PLANT.PIPEC2.PSI(36) = 2.60848e+006 ; # 420
PLANT.PIPEC2.PSI(37) = 2.60489e+006 ; # 421
PLANT.PIPEC2.PSI(38) = 2.60088e+006 ; # 422
PLANT.PIPEC2.PSI(39) = 2.59648e+006 ; # 423
PLANT.PIPEC2.PSI(40) = 2.59171e+006 ; # 424
```

HTLNOPIPEHQSS_SS_OUT_PRE.PRESETS

```
# Values for computation HTLQSS_SS
# Saved at time 2120
PLANT.HX.TGASHOT(1) := 902.624 ; # 1
PLANT.HX.TGASHOT(2) := 913.125 ; # 2
PLANT.HX.TGASHOT(3) := 923.388 ; # 3
PLANT.HX.TGASHOT(4) := 933.4 ; # 4
```

```
PLANT.HX.TGASHOT(5) := 943.167 ; # 5
PLANT.HX.TGASHOT(6) := 952.696 ; # 6
PLANT.HX.TGASHOT(7) := 961.993 ; # 7
PLANT.HX.TGASHOT(8) := 971.062 ; # 8
PLANT.HX.TGASHOT(9) := 979.91 ; # 9
PLANT.HX.TGASHOT(10) := 988.542 ; # 10
PLANT.HX.TGASHOT(11) := 996.962 ; # 11
PLANT.HX.TGASHOT(12) := 1005.18 ; # 12
PLANT.HX.TGASHOT(13) := 1013.19 ; # 13
PLANT.HX.TGASHOT(14) := 1021.01 ; # 14
PLANT.HX.TGASHOT(15) := 1028.64 ; # 15
PLANT.HX.TGASHOT(16) := 1036.08 ; # 16
PLANT.HX.TGASHOT(17) := 1043.34 ; # 17
PLANT.HX.TGASHOT(18) := 1050.42 ; # 18
PLANT.HX.TGASHOT(19) := 1057.33 ; # 19
PLANT.HX.TGASHOT(20) := 1064.07 ; # 20
PLANT.HX.TGASHOT(21) := 1070.65 ; # 21
PLANT.HX.TGASHOT(22) := 1077.07 ; # 22
PLANT.HX.TGASHOT(23) := 1083.33 ; # 23
PLANT.HX.TGASHOT(24) := 1089.43 ; # 24
PLANT.HX.TGASHOT(25) := 1095.39 ; # 25
PLANT.HX.TGASHOT(26) := 1101.2 ; # 26
PLANT.HX.TGASHOT(27) := 1106.87 ; # 27
PLANT.HX.TGASHOT(28) := 1112.4 ; # 28
PLANT.HX.TGASHOT(29) := 1117.8 ; # 29
PLANT.HX.TGASHOT(30) := 1123.06 ; # 30
PLANT.HX.TGASHOT(31) := 1128.2 ; # 31
PLANT.HX.TGASHOT(32) := 1133.21 ; # 32
PLANT.HX.TGASHOT(33) := 1138.1 ; # 33
PLANT.HX.TGASHOT(34) := 1142.87 ; # 34
PLANT.HX.TGASHOT(35) := 1147.52 ; # 35
PLANT.HX.TGASHOT(36) := 1152.06 ; # 36
PLANT.HX.TGASHOT(37) := 1156.48 ; # 37
PLANT.HX.TGASHOT(38) := 1160.8 ; # 38
PLANT.HX.TGASHOT(39) := 1165.02 ; # 39
PLANT.HX.TGASHOT(40) := 1169.13 ; # 40
PLANT.HX.TGASCOLD(1) := 847.757 ; # 41
PLANT.HX.TGASCOLD(2) := 859.616 ; # 42
PLANT.HX.TGASCOLD(3) := 871.186 ; # 43
PLANT.HX.TGASCOLD(4) := 882.472 ; # 44
PLANT.HX.TGASCOLD(5) := 893.484 ; # 45
PLANT.HX.TGASCOLD(6) := 904.226 ; # 46
PLANT.HX.TGASCOLD(7) := 914.706 ; # 47
PLANT.HX.TGASCOLD(8) := 924.93 ; # 48
PLANT.HX.TGASCOLD(9) := 934.904 ; # 49
PLANT.HX.TGASCOLD(10) := 944.635 ; # 50
PLANT.HX.TGASCOLD(11) := 954.128 ; # 51
PLANT.HX.TGASCOLD(12) := 963.39 ; # 52
PLANT.HX.TGASCOLD(13) := 972.425 ; # 53
PLANT.HX.TGASCOLD(14) := 981.239 ; # 54
PLANT.HX.TGASCOLD(15) := 989.838 ; # 55
PLANT.HX.TGASCOLD(16) := 998.228 ; # 56
PLANT.HX.TGASCOLD(17) := 1006.41 ; # 57
PLANT.HX.TGASCOLD(18) := 1014.4 ; # 58
PLANT.HX.TGASCOLD(19) := 1022.19 ; # 59
PLANT.HX.TGASCOLD(20) := 1029.79 ; # 60
PLANT.HX.TGASCOLD(21) := 1037.2 ; # 61
PLANT.HX.TGASCOLD(22) := 1044.43 ; # 62
PLANT.HX.TGASCOLD(23) := 1051.49 ; # 63
PLANT.HX.TGASCOLD(24) := 1058.37 ; # 64
PLANT.HX.TGASCOLD(25) := 1065.09 ; # 65
PLANT.HX.TGASCOLD(26) := 1071.64 ; # 66
PLANT.HX.TGASCOLD(27) := 1078.03 ; # 67
PLANT.HX.TGASCOLD(28) := 1084.27 ; # 68
PLANT.HX.TGASCOLD(29) := 1090.35 ; # 69
PLANT.HX.TGASCOLD(30) := 1096.28 ; # 70
PLANT.HX.TGASCOLD(31) := 1102.07 ; # 71
PLANT.HX.TGASCOLD(32) := 1107.72 ; # 72
PLANT.HX.TGASCOLD(33) := 1113.23 ; # 73
PLANT.HX.TGASCOLD(34) := 1118.61 ; # 74
PLANT.HX.TGASCOLD(35) := 1123.85 ; # 75
PLANT.HX.TGASCOLD(36) := 1128.97 ; # 76
PLANT.HX.TGASCOLD(37) := 1133.96 ; # 77
PLANT.HX.TGASCOLD(38) := 1138.83 ; # 78
PLANT.HX.TGASCOLD(39) := 1143.58 ; # 79
PLANT.HX.TGASCOLD(40) := 1148.21 ; # 80
PLANT.HX.UHOT := 1738.41 ; # 81
PLANT.HX.UCOLD := 1645.45 ; # 82
PLANT.HX.THX(1) := 875.997 ; # 83
PLANT.HX.THX(2) := 887.105 ; # 84
PLANT.HX.THX(3) := 898.003 ; # 85
PLANT.HX.THX(4) := 908.635 ; # 86
PLANT.HX.THX(5) := 919.007 ; # 87
PLANT.HX.THX(6) := 929.126 ; # 88
PLANT.HX.THX(7) := 938.998 ; # 89
PLANT.HX.THX(8) := 948.629 ; # 90
PLANT.HX.THX(9) := 958.024 ; # 91
PLANT.HX.THX(10) := 967.19 ; # 92
PLANT.HX.THX(11) := 976.133 ; # 93
PLANT.HX.THX(12) := 984.857 ; # 94
PLANT.HX.THX(13) := 993.368 ; # 95
PLANT.HX.THX(14) := 1001.67 ; # 96
```

```
PLANT.HX.THX(15) := 1009.77 ; # 97
PLANT.HX.THX(16) := 1017.67 ; # 98
PLANT.HX.THX(17) := 1025.38 ; # 99
PLANT.HX.THX(18) := 1032.9 ; # 100
PLANT.HX.THX(19) := 1040.24 ; # 101
PLANT.HX.THX(20) := 1047.4 ; # 102
PLANT.HX.THX(21) := 1054.38 ; # 103
PLANT.HX.THX(22) := 1061.2 ; # 104
PLANT.HX.THX(23) := 1067.84 ; # 105
PLANT.HX.THX(24) := 1074.33 ; # 106
PLANT.HX.THX(25) := 1080.65 ; # 107
PLANT.HX.THX(26) := 1086.83 ; # 108
PLANT.HX.THX(27) := 1092.85 ; # 109
PLANT.HX.THX(28) := 1098.72 ; # 110
PLANT.HX.THX(29) := 1104.45 ; # 111
PLANT.HX.THX(30) := 1110.04 ; # 112
PLANT.HX.THX(31) := 1115.49 ; # 113
PLANT.HX.THX(32) := 1120.82 ; # 114
PLANT.HX.THX(33) := 1126.01 ; # 115
PLANT.HX.THX(34) := 1131.07 ; # 116
PLANT.HX.THX(35) := 1136.01 ; # 117
PLANT.HX.THX(36) := 1140.83 ; # 118
PLANT.HX.THX(37) := 1145.53 ; # 119
PLANT.HX.THX(38) := 1150.12 ; # 120
PLANT.HX.THX(39) := 1154.59 ; # 121
PLANT.HX.THX(40) := 1158.94 ; # 122
PLANT.LOOP1.RHO(1) := 1.14186 ; # 123
PLANT.LOOP1.RHO(2) := 1.13901 ; # 124
PLANT.LOOP1.RHO(3) := 1.13615 ; # 125
PLANT.LOOP1.RHO(4) := 1.13328 ; # 126
PLANT.LOOP1.RHO(5) := 1.13039 ; # 127
PLANT.LOOP1.RHO(6) := 1.1275 ; # 128
PLANT.LOOP1.RHO(7) := 1.12458 ; # 129
PLANT.LOOP1.RHO(8) := 1.12166 ; # 130
PLANT.LOOP1.RHO(9) := 1.11873 ; # 131
PLANT.LOOP1.RHO(10) := 1.11578 ; # 132
PLANT.LOOP1.RHO(11) := 1.10995 ; # 133
PLANT.LOOP1.RHO(12) := 1.09332 ; # 134
PLANT.LOOP1.RHO(13) := 1.07711 ; # 135
PLANT.LOOP1.RHO(14) := 1.06165 ; # 136
PLANT.LOOP1.RHO(15) := 1.04688 ; # 137
PLANT.LOOP1.RHO(16) := 1.03275 ; # 138
PLANT.LOOP1.RHO(17) := 1.01923 ; # 139
PLANT.LOOP1.RHO(18) := 1.00627 ; # 140
PLANT.LOOP1.RHO(19) := 0.993839 ; # 141
PLANT.LOOP1.RHO(20) := 0.981903 ; # 142
PLANT.LOOP1.RHO(21) := 0.970434 ; # 143
PLANT.LOOP1.RHO(22) := 0.959403 ; # 144
PLANT.LOOP1.RHO(23) := 0.948785 ; # 145
PLANT.LOOP1.RHO(24) := 0.938555 ; # 146
PLANT.LOOP1.RHO(25) := 0.928692 ; # 147
PLANT.LOOP1.RHO(26) := 0.919175 ; # 148
PLANT.LOOP1.RHO(27) := 0.909986 ; # 149
PLANT.LOOP1.RHO(28) := 0.901106 ; # 150
PLANT.LOOP1.RHO(29) := 0.89252 ; # 151
PLANT.LOOP1.RHO(30) := 0.884211 ; # 152
PLANT.LOOP1.RHO(31) := 0.876165 ; # 153
PLANT.LOOP1.RHO(32) := 0.868369 ; # 154
PLANT.LOOP1.RHO(33) := 0.860811 ; # 155
PLANT.LOOP1.RHO(34) := 0.853477 ; # 156
PLANT.LOOP1.RHO(35) := 0.846358 ; # 157
PLANT.LOOP1.RHO(36) := 0.839443 ; # 158
PLANT.LOOP1.RHO(37) := 0.832722 ; # 159
PLANT.LOOP1.RHO(38) := 0.826186 ; # 160
PLANT.LOOP1.RHO(39) := 0.819826 ; # 161
PLANT.LOOP1.RHO(40) := 0.813634 ; # 162
PLANT.LOOP1.RHO(41) := 0.807602 ; # 163
PLANT.LOOP1.RHO(42) := 0.801723 ; # 164
PLANT.LOOP1.RHO(43) := 0.795989 ; # 165
PLANT.LOOP1.RHO(44) := 0.790396 ; # 166
PLANT.LOOP1.RHO(45) := 0.784935 ; # 167
PLANT.LOOP1.RHO(46) := 0.779602 ; # 168
PLANT.LOOP1.RHO(47) := 0.774391 ; # 169
PLANT.LOOP1.RHO(48) := 0.769296 ; # 170
PLANT.LOOP1.RHO(49) := 0.764313 ; # 171
PLANT.LOOP1.RHO(50) := 0.759442 ; # 172
PLANT.LOOP1.RHO(51) := 0.746737 ; # 173
PLANT.LOOP1.RHO(52) := 0.744574 ; # 174
PLANT.LOOP1.RHO(53) := 0.74242 ; # 175
PLANT.LOOP1.RHO(54) := 0.740254 ; # 176
PLANT.LOOP1.RHO(55) := 0.738076 ; # 177
PLANT.LOOP1.RHO(56) := 0.735886 ; # 178
PLANT.LOOP1.RHO(57) := 0.733685 ; # 179
PLANT.LOOP1.RHO(58) := 0.731471 ; # 180
PLANT.LOOP1.RHO(59) := 0.729245 ; # 181
PLANT.LOOP1.RHO(60) := 0.727007 ; # 182
PLANT.LOOP1.RHO(61) := 0.737706 ; # 183
PLANT.LOOP1.RHO(62) := 0.738969 ; # 184
PLANT.LOOP1.RHO(63) := 0.740294 ; # 185
PLANT.LOOP1.RHO(64) := 0.741736 ; # 186
PLANT.LOOP1.RHO(65) := 0.7433 ; # 187
PLANT.LOOP1.RHO(66) := 0.744995 ; # 188
```

```
PLANT.LOOP1.RHO(67) := 0.74683 ; # 189
PLANT.LOOP1.RHO(68) := 0.748813 ; # 190
PLANT.LOOP1.RHO(69) := 0.750955 ; # 191
PLANT.LOOP1.RHO(70) := 0.753266 ; # 192
PLANT.LOOP1.RHO(71) := 0.755759 ; # 193
PLANT.LOOP1.RHO(72) := 0.758444 ; # 194
PLANT.LOOP1.RHO(73) := 0.761337 ; # 195
PLANT.LOOP1.RHO(74) := 0.764452 ; # 196
PLANT.LOOP1.RHO(75) := 0.767805 ; # 197
PLANT.LOOP1.RHO(76) := 0.771413 ; # 198
PLANT.LOOP1.RHO(77) := 0.775295 ; # 199
PLANT.LOOP1.RHO(78) := 0.779473 ; # 200
PLANT.LOOP1.RHO(79) := 0.783969 ; # 201
PLANT.LOOP1.RHO(80) := 0.788809 ; # 202
PLANT.LOOP1.RHO(81) := 0.79402 ; # 203
PLANT.LOOP1.RHO(82) := 0.799632 ; # 204
PLANT.LOOP1.RHO(83) := 0.805679 ; # 205
PLANT.LOOP1.RHO(84) := 0.812199 ; # 206
PLANT.LOOP1.RHO(85) := 0.819233 ; # 207
PLANT.LOOP1.RHO(86) := 0.826828 ; # 208
PLANT.LOOP1.RHO(87) := 0.835034 ; # 209
PLANT.LOOP1.RHO(88) := 0.843911 ; # 210
PLANT.LOOP1.RHO(89) := 0.853523 ; # 211
PLANT.LOOP1.RHO(90) := 0.863944 ; # 212
PLANT.LOOP1.RHO(91) := 0.875257 ; # 213
PLANT.LOOP1.RHO(92) := 0.887559 ; # 214
PLANT.LOOP1.RHO(93) := 0.900957 ; # 215
PLANT.LOOP1.RHO(94) := 0.915576 ; # 216
PLANT.LOOP1.RHO(95) := 0.931562 ; # 217
PLANT.LOOP1.RHO(96) := 0.949081 ; # 218
PLANT.LOOP1.RHO(97) := 0.96833 ; # 219
PLANT.LOOP1.RHO(98) := 0.989539 ; # 220
PLANT.LOOP1.RHO(99) := 1.01298 ; # 221
PLANT.LOOP1.RHO(100) := 1.03897 ; # 222
PLANT.LOOP1.P(1) := 1.99164e+006 ; # 223
PLANT.LOOP1.P(2) := 1.98572e+006 ; # 224
PLANT.LOOP1.P(3) := 1.97979e+006 ; # 225
PLANT.LOOP1.P(4) := 1.97384e+006 ; # 226
PLANT.LOOP1.P(5) := 1.96788e+006 ; # 227
PLANT.LOOP1.P(6) := 1.9619e+006 ; # 228
PLANT.LOOP1.P(7) := 1.9559e+006 ; # 229
PLANT.LOOP1.P(8) := 1.94989e+006 ; # 230
PLANT.LOOP1.P(9) := 1.94386e+006 ; # 231
PLANT.LOOP1.P(10) := 1.93782e+006 ; # 232
PLANT.LOOP1.P(11) := 1.95581e+006 ; # 233
PLANT.LOOP1.P(12) := 1.95345e+006 ; # 234
PLANT.LOOP1.P(13) := 1.95039e+006 ; # 235
PLANT.LOOP1.P(14) := 1.9473e+006 ; # 236
PLANT.LOOP1.P(15) := 1.94417e+006 ; # 237
PLANT.LOOP1.P(16) := 1.94099e+006 ; # 238
PLANT.LOOP1.P(17) := 1.93778e+006 ; # 239
PLANT.LOOP1.P(18) := 1.93452e+006 ; # 240
PLANT.LOOP1.P(19) := 1.93123e+006 ; # 241
PLANT.LOOP1.P(20) := 1.92789e+006 ; # 242
PLANT.LOOP1.P(21) := 1.92452e+006 ; # 243
PLANT.LOOP1.P(22) := 1.92111e+006 ; # 244
PLANT.LOOP1.P(23) := 1.91767e+006 ; # 245
PLANT.LOOP1.P(24) := 1.91419e+006 ; # 246
PLANT.LOOP1.P(25) := 1.91067e+006 ; # 247
PLANT.LOOP1.P(26) := 1.90712e+006 ; # 248
PLANT.LOOP1.P(27) := 1.90353e+006 ; # 249
PLANT.LOOP1.P(28) := 1.89991e+006 ; # 250
PLANT.LOOP1.P(29) := 1.89626e+006 ; # 251
PLANT.LOOP1.P(30) := 1.89257e+006 ; # 252
PLANT.LOOP1.P(31) := 1.88885e+006 ; # 253
PLANT.LOOP1.P(32) := 1.8851e+006 ; # 254
PLANT.LOOP1.P(33) := 1.88132e+006 ; # 255
PLANT.LOOP1.P(34) := 1.8775e+006 ; # 256
PLANT.LOOP1.P(35) := 1.87365e+006 ; # 257
PLANT.LOOP1.P(36) := 1.86978e+006 ; # 258
PLANT.LOOP1.P(37) := 1.86587e+006 ; # 259
PLANT.LOOP1.P(38) := 1.86193e+006 ; # 260
PLANT.LOOP1.P(39) := 1.85796e+006 ; # 261
PLANT.LOOP1.P(40) := 1.85397e+006 ; # 262
PLANT.LOOP1.P(41) := 1.84994e+006 ; # 263
PLANT.LOOP1.P(42) := 1.84589e+006 ; # 264
PLANT.LOOP1.P(43) := 1.8418e+006 ; # 265
PLANT.LOOP1.P(44) := 1.83769e+006 ; # 266
PLANT.LOOP1.P(45) := 1.83355e+006 ; # 267
PLANT.LOOP1.P(46) := 1.82939e+006 ; # 268
PLANT.LOOP1.P(47) := 1.82519e+006 ; # 269
PLANT.LOOP1.P(48) := 1.82097e+006 ; # 270
PLANT.LOOP1.P(49) := 1.81672e+006 ; # 271
PLANT.LOOP1.P(50) := 1.81245e+006 ; # 272
PLANT.LOOP1.P(51) := 1.78092e+006 ; # 273
PLANT.LOOP1.P(52) := 1.77456e+006 ; # 274
PLANT.LOOP1.P(53) := 1.76822e+006 ; # 275
PLANT.LOOP1.P(54) := 1.76187e+006 ; # 276
PLANT.LOOP1.P(55) := 1.75549e+006 ; # 277
PLANT.LOOP1.P(56) := 1.7491e+006 ; # 278
PLANT.LOOP1.P(57) := 1.74268e+006 ; # 279
PLANT.LOOP1.P(58) := 1.73625e+006 ; # 280
```

```
PLANT.LOOP1.P(59) := 1.72979e+006 ; # 281
PLANT.LOOP1.P(60) := 1.72331e+006 ; # 282
PLANT.LOOP1.P(61) := 1.74441e+006 ; # 283
PLANT.LOOP1.P(62) := 1.74289e+006 ; # 284
PLANT.LOOP1.P(63) := 1.74128e+006 ; # 285
PLANT.LOOP1.P(64) := 1.73966e+006 ; # 286
PLANT.LOOP1.P(65) := 1.73806e+006 ; # 287
PLANT.LOOP1.P(66) := 1.73645e+006 ; # 288
PLANT.LOOP1.P(67) := 1.73485e+006 ; # 289
PLANT.LOOP1.P(68) := 1.73326e+006 ; # 290
PLANT.LOOP1.P(69) := 1.73167e+006 ; # 291
PLANT.LOOP1.P(70) := 1.73009e+006 ; # 292
PLANT.LOOP1.P(71) := 1.72851e+006 ; # 293
PLANT.LOOP1.P(72) := 1.72694e+006 ; # 294
PLANT.LOOP1.P(73) := 1.72538e+006 ; # 295
PLANT.LOOP1.P(74) := 1.72383e+006 ; # 296
PLANT.LOOP1.P(75) := 1.72228e+006 ; # 297
PLANT.LOOP1.P(76) := 1.72075e+006 ; # 298
PLANT.LOOP1.P(77) := 1.71922e+006 ; # 299
PLANT.LOOP1.P(78) := 1.7177e+006 ; # 300
PLANT.LOOP1.P(79) := 1.7162e+006 ; # 301
PLANT.LOOP1.P(80) := 1.7147e+006 ; # 302
PLANT.LOOP1.P(81) := 1.71322e+006 ; # 303
PLANT.LOOP1.P(82) := 1.71175e+006 ; # 304
PLANT.LOOP1.P(83) := 1.71029e+006 ; # 305
PLANT.LOOP1.P(84) := 1.70885e+006 ; # 306
PLANT.LOOP1.P(85) := 1.70742e+006 ; # 307
PLANT.LOOP1.P(86) := 1.70601e+006 ; # 308
PLANT.LOOP1.P(87) := 1.70462e+006 ; # 309
PLANT.LOOP1.P(88) := 1.70324e+006 ; # 310
PLANT.LOOP1.P(89) := 1.70189e+006 ; # 311
PLANT.LOOP1.P(90) := 1.70055e+006 ; # 312
PLANT.LOOP1.P(91) := 1.69924e+006 ; # 313
PLANT.LOOP1.P(92) := 1.69795e+006 ; # 314
PLANT.LOOP1.P(93) := 1.69668e+006 ; # 315
PLANT.LOOP1.P(94) := 1.69543e+006 ; # 316
PLANT.LOOP1.P(95) := 1.69422e+006 ; # 317
PLANT.LOOP1.P(96) := 1.69303e+006 ; # 318
PLANT.LOOP1.P(97) := 1.69187e+006 ; # 319
PLANT.LOOP1.P(98) := 1.69075e+006 ; # 320
PLANT.LOOP1.P(99) := 1.68965e+006 ; # 321
PLANT.LOOP1.P(100) := 1.6886e+006 ; # 322
PLANT.LOOP1.P_OUT := 1.68802e+006 ; # 323
PLANT.LOOP1.H(1) := 2.81134e+006 ; # 324
PLANT.LOOP1.H(2) := 2.80926e+006 ; # 325
PLANT.LOOP1.H(3) := 2.80719e+006 ; # 326
PLANT.LOOP1.H(4) := 2.80511e+006 ; # 327
PLANT.LOOP1.H(5) := 2.80304e+006 ; # 328
PLANT.LOOP1.H(6) := 2.80097e+006 ; # 329
PLANT.LOOP1.H(7) := 2.7989e+006 ; # 330
PLANT.LOOP1.H(8) := 2.79683e+006 ; # 331
PLANT.LOOP1.H(9) := 2.79476e+006 ; # 332
PLANT.LOOP1.H(10) := 2.7927e+006 ; # 333
PLANT.LOOP1.H(11) := 2.85599e+006 ; # 334
PLANT.LOOP1.H(12) := 2.9176e+006 ; # 335
PLANT.LOOP1.H(13) := 2.9777e+006 ; # 336
PLANT.LOOP1.H(14) := 3.03633e+006 ; # 337
PLANT.LOOP1.H(15) := 3.09354e+006 ; # 338
PLANT.LOOP1.H(16) := 3.14934e+006 ; # 339
PLANT.LOOP1.H(17) := 3.20379e+006 ; # 340
PLANT.LOOP1.H(18) := 3.2569e+006 ; # 341
PLANT.LOOP1.H(19) := 3.30872e+006 ; # 342
PLANT.LOOP1.H(20) := 3.35927e+006 ; # 343
PLANT.LOOP1.H(21) := 3.40859e+006 ; # 344
PLANT.LOOP1.H(22) := 3.4567e+006 ; # 345
PLANT.LOOP1.H(23) := 3.50364e+006 ; # 346
PLANT.LOOP1.H(24) := 3.54943e+006 ; # 347
PLANT.LOOP1.H(25) := 3.5941e+006 ; # 348
PLANT.LOOP1.H(26) := 3.63768e+006 ; # 349
PLANT.LOOP1.H(27) := 3.6802e+006 ; # 350
PLANT.LOOP1.H(28) := 3.72168e+006 ; # 351
PLANT.LOOP1.H(29) := 3.76215e+006 ; # 352
PLANT.LOOP1.H(30) := 3.80163e+006 ; # 353
PLANT.LOOP1.H(31) := 3.84014e+006 ; # 354
PLANT.LOOP1.H(32) := 3.87771e+006 ; # 355
PLANT.LOOP1.H(33) := 3.91437e+006 ; # 356
PLANT.LOOP1.H(34) := 3.95013e+006 ; # 357
PLANT.LOOP1.H(35) := 3.98502e+006 ; # 358
PLANT.LOOP1.H(36) := 4.01905e+006 ; # 359
PLANT.LOOP1.H(37) := 4.05226e+006 ; # 360
PLANT.LOOP1.H(38) := 4.08465e+006 ; # 361
PLANT.LOOP1.H(39) := 4.11626e+006 ; # 362
PLANT.LOOP1.H(40) := 4.14709e+006 ; # 363
PLANT.LOOP1.H(41) := 4.17717e+006 ; # 364
PLANT.LOOP1.H(42) := 4.20651e+006 ; # 365
PLANT.LOOP1.H(43) := 4.23514e+006 ; # 366
PLANT.LOOP1.H(44) := 4.26306e+006 ; # 367
PLANT.LOOP1.H(45) := 4.29031e+006 ; # 368
PLANT.LOOP1.H(46) := 4.31689e+006 ; # 369
PLANT.LOOP1.H(47) := 4.34282e+006 ; # 370
PLANT.LOOP1.H(48) := 4.36812e+006 ; # 371
PLANT.LOOP1.H(49) := 4.3928e+006 ; # 372
```

```
375   PLANT.LOOP1.H(50) := 4.41684e+006 ; # 373
376   PLANT.LOOP1.H(51) := 4.41279e+006 ; # 374
377   PLANT.LOOP1.H(52) := 4.40875e+006 ; # 375
378   PLANT.LOOP1.H(53) := 4.4047e+006 ; # 376
379   PLANT.LOOP1.H(54) := 4.40067e+006 ; # 377
380   PLANT.LOOP1.H(55) := 4.39663e+006 ; # 378
381   PLANT.LOOP1.H(56) := 4.3926e+006 ; # 379
382   PLANT.LOOP1.H(57) := 4.38857e+006 ; # 380
383   PLANT.LOOP1.H(58) := 4.38455e+006 ; # 381
384   PLANT.LOOP1.H(59) := 4.38053e+006 ; # 382
385   PLANT.LOOP1.H(60) := 4.37651e+006 ; # 383
386   PLANT.LOOP1.H(61) := 4.36206e+006 ; # 384
387   PLANT.LOOP1.H(62) := 4.34684e+006 ; # 385
388   PLANT.LOOP1.H(63) := 4.33083e+006 ; # 386
389   PLANT.LOOP1.H(64) := 4.31397e+006 ; # 387
390   PLANT.LOOP1.H(65) := 4.29622e+006 ; # 388
391   PLANT.LOOP1.H(66) := 4.27754e+006 ; # 389
392   PLANT.LOOP1.H(67) := 4.25788e+006 ; # 390
393   PLANT.LOOP1.H(68) := 4.23718e+006 ; # 391
394   PLANT.LOOP1.H(69) := 4.21539e+006 ; # 392
395   PLANT.LOOP1.H(70) := 4.19246e+006 ; # 393
396   PLANT.LOOP1.H(71) := 4.16832e+006 ; # 394
397   PLANT.LOOP1.H(72) := 4.14291e+006 ; # 395
398   PLANT.LOOP1.H(73) := 4.11616e+006 ; # 396
399   PLANT.LOOP1.H(74) := 4.088e+006 ; # 397
400   PLANT.LOOP1.H(75) := 4.05836e+006 ; # 398
401   PLANT.LOOP1.H(76) := 4.02716e+006 ; # 399
402   PLANT.LOOP1.H(77) := 3.99432e+006 ; # 400
403   PLANT.LOOP1.H(78) := 3.95975e+006 ; # 401
404   PLANT.LOOP1.H(79) := 3.92336e+006 ; # 402
405   PLANT.LOOP1.H(80) := 3.88505e+006 ; # 403
406   PLANT.LOOP1.H(81) := 3.84473e+006 ; # 404
407   PLANT.LOOP1.H(82) := 3.80228e+006 ; # 405
408   PLANT.LOOP1.H(83) := 3.75761e+006 ; # 406
409   PLANT.LOOP1.H(84) := 3.71058e+006 ; # 407
410   PLANT.LOOP1.H(85) := 3.66107e+006 ; # 408
411   PLANT.LOOP1.H(86) := 3.60896e+006 ; # 409
412   PLANT.LOOP1.H(87) := 3.55411e+006 ; # 410
413   PLANT.LOOP1.H(88) := 3.49637e+006 ; # 411
414   PLANT.LOOP1.H(89) := 3.43559e+006 ; # 412
415   PLANT.LOOP1.H(90) := 3.37161e+006 ; # 413
416   PLANT.LOOP1.H(91) := 3.30426e+006 ; # 414
417   PLANT.LOOP1.H(92) := 3.23337e+006 ; # 415
418   PLANT.LOOP1.H(93) := 3.15875e+006 ; # 416
419   PLANT.LOOP1.H(94) := 3.0802e+006 ; # 417
420   PLANT.LOOP1.H(95) := 2.99751e+006 ; # 418
421   PLANT.LOOP1.H(96) := 2.91047e+006 ; # 419
422   PLANT.LOOP1.H(97) := 2.81886e+006 ; # 420
423   PLANT.LOOP1.H(98) := 2.72242e+006 ; # 421
424   PLANT.LOOP1.H(99) := 2.6209e+006 ; # 422
425   PLANT.LOOP1.H(100) := 2.51405e+006 ; # 423
426   PLANT.LOOP1.H0 := 2.81342e+006 ; # 424
427   PLANT.LOOP1.T(1) := 839.163 ; # 425
428   PLANT.LOOP1.T(2) := 838.763 ; # 426
429   PLANT.LOOP1.T(3) := 838.363 ; # 427
430   PLANT.LOOP1.T(4) := 837.964 ; # 428
431   PLANT.LOOP1.T(5) := 837.565 ; # 429
432   PLANT.LOOP1.T(6) := 837.166 ; # 430
433   PLANT.LOOP1.T(7) := 836.767 ; # 431
434   PLANT.LOOP1.T(8) := 836.369 ; # 432
435   PLANT.LOOP1.T(9) := 835.971 ; # 433
436   PLANT.LOOP1.T(10) := 835.574 ; # 434
437   PLANT.LOOP1.T(11) := 847.757 ; # 435
438   PLANT.LOOP1.T(12) := 859.616 ; # 436
439   PLANT.LOOP1.T(13) := 871.186 ; # 437
440   PLANT.LOOP1.T(14) := 882.472 ; # 438
441   PLANT.LOOP1.T(15) := 893.484 ; # 439
442   PLANT.LOOP1.T(16) := 904.226 ; # 440
443   PLANT.LOOP1.T(17) := 914.706 ; # 441
444   PLANT.LOOP1.T(18) := 924.93 ; # 442
445   PLANT.LOOP1.T(19) := 934.904 ; # 443
446   PLANT.LOOP1.T(20) := 944.635 ; # 444
447   PLANT.LOOP1.T(21) := 954.128 ; # 445
448   PLANT.LOOP1.T(22) := 963.39 ; # 446
449   PLANT.LOOP1.T(23) := 972.425 ; # 447
450   PLANT.LOOP1.T(24) := 981.239 ; # 448
451   PLANT.LOOP1.T(25) := 989.838 ; # 449
452   PLANT.LOOP1.T(26) := 998.228 ; # 450
453   PLANT.LOOP1.T(27) := 1006.41 ; # 451
454   PLANT.LOOP1.T(28) := 1014.4 ; # 452
455   PLANT.LOOP1.T(29) := 1022.19 ; # 453
456   PLANT.LOOP1.T(30) := 1029.79 ; # 454
457   PLANT.LOOP1.T(31) := 1037.2 ; # 455
458   PLANT.LOOP1.T(32) := 1044.43 ; # 456
459   PLANT.LOOP1.T(33) := 1051.49 ; # 457
460   PLANT.LOOP1.T(34) := 1058.37 ; # 458
461   PLANT.LOOP1.T(35) := 1065.09 ; # 459
462   PLANT.LOOP1.T(36) := 1071.64 ; # 460
463   PLANT.LOOP1.T(37) := 1078.03 ; # 461
464   PLANT.LOOP1.T(38) := 1084.27 ; # 462
465   PLANT.LOOP1.T(39) := 1090.35 ; # 463
466   PLANT.LOOP1.T(40) := 1096.28 ; # 464
```

```
PLANT.LOOP1.T(41) := 1102.07 ; # 465
PLANT.LOOP1.T(42) := 1107.72 ; # 466
PLANT.LOOP1.T(43) := 1113.23 ; # 467
PLANT.LOOP1.T(44) := 1118.61 ; # 468
PLANT.LOOP1.T(45) := 1123.85 ; # 469
PLANT.LOOP1.T(46) := 1128.97 ; # 470
PLANT.LOOP1.T(47) := 1133.96 ; # 471
PLANT.LOOP1.T(48) := 1138.83 ; # 472
PLANT.LOOP1.T(49) := 1143.58 ; # 473
PLANT.LOOP1.T(50) := 1148.21 ; # 474
PLANT.LOOP1.T(51) := 1147.43 ; # 475
PLANT.LOOP1.T(52) := 1146.65 ; # 476
PLANT.LOOP1.T(53) := 1145.87 ; # 477
PLANT.LOOP1.T(54) := 1145.1 ; # 478
PLANT.LOOP1.T(55) := 1144.32 ; # 479
PLANT.LOOP1.T(56) := 1143.54 ; # 480
PLANT.LOOP1.T(57) := 1142.77 ; # 481
PLANT.LOOP1.T(58) := 1141.99 ; # 482
PLANT.LOOP1.T(59) := 1141.22 ; # 483
PLANT.LOOP1.T(60) := 1140.45 ; # 484
PLANT.LOOP1.T(61) := 1137.66 ; # 485
PLANT.LOOP1.T(62) := 1134.74 ; # 486
PLANT.LOOP1.T(63) := 1131.65 ; # 487
PLANT.LOOP1.T(64) := 1128.41 ; # 488
PLANT.LOOP1.T(65) := 1124.99 ; # 489
PLANT.LOOP1.T(66) := 1121.4 ; # 490
PLANT.LOOP1.T(67) := 1117.61 ; # 491
PLANT.LOOP1.T(68) := 1113.63 ; # 492
PLANT.LOOP1.T(69) := 1109.43 ; # 493
PLANT.LOOP1.T(70) := 1105.02 ; # 494
PLANT.LOOP1.T(71) := 1100.37 ; # 495
PLANT.LOOP1.T(72) := 1095.48 ; # 496
PLANT.LOOP1.T(73) := 1090.33 ; # 497
PLANT.LOOP1.T(74) := 1084.91 ; # 498
PLANT.LOOP1.T(75) := 1079.2 ; # 499
PLANT.LOOP1.T(76) := 1073.2 ; # 500
PLANT.LOOP1.T(77) := 1066.88 ; # 501
PLANT.LOOP1.T(78) := 1060.22 ; # 502
PLANT.LOOP1.T(79) := 1053.22 ; # 503
PLANT.LOOP1.T(80) := 1045.84 ; # 504
PLANT.LOOP1.T(81) := 1038.08 ; # 505
PLANT.LOOP1.T(82) := 1029.91 ; # 506
PLANT.LOOP1.T(83) := 1021.31 ; # 507
PLANT.LOOP1.T(84) := 1012.26 ; # 508
PLANT.LOOP1.T(85) := 1002.73 ; # 509
PLANT.LOOP1.T(86) := 992.699 ; # 510
PLANT.LOOP1.T(87) := 982.14 ; # 511
PLANT.LOOP1.T(88) := 971.025 ; # 512
PLANT.LOOP1.T(89) := 959.325 ; # 513
PLANT.LOOP1.T(90) := 947.01 ; # 514
PLANT.LOOP1.T(91) := 934.046 ; # 515
PLANT.LOOP1.T(92) := 920.4 ; # 516
PLANT.LOOP1.T(93) := 906.036 ; # 517
PLANT.LOOP1.T(94) := 890.915 ; # 518
PLANT.LOOP1.T(95) := 874.999 ; # 519
PLANT.LOOP1.T(96) := 858.245 ; # 520
PLANT.LOOP1.T(97) := 840.609 ; # 521
PLANT.LOOP1.T(98) := 822.045 ; # 522
PLANT.LOOP1.T(99) := 802.504 ; # 523
PLANT.LOOP1.T(100) := 781.936 ; # 524
PLANT.LOOP1.T0 := 839.563 ; # 525
PLANT.LOOP1.TEXTRNL(1) := 291.15 ; # 526
PLANT.LOOP1.TEXTRNL(2) := 291.15 ; # 527
PLANT.LOOP1.TEXTRNL(3) := 291.15 ; # 528
PLANT.LOOP1.TEXTRNL(4) := 291.15 ; # 529
PLANT.LOOP1.TEXTRNL(5) := 291.15 ; # 530
PLANT.LOOP1.TEXTRNL(6) := 291.15 ; # 531
PLANT.LOOP1.TEXTRNL(7) := 291.15 ; # 532
PLANT.LOOP1.TEXTRNL(8) := 291.15 ; # 533
PLANT.LOOP1.TEXTRNL(9) := 291.15 ; # 534
PLANT.LOOP1.TEXTRNL(10) := 291.15 ; # 535
PLANT.LOOP1.TEXTRNL(11) := 875.997 ; # 536
PLANT.LOOP1.TEXTRNL(12) := 887.105 ; # 537
PLANT.LOOP1.TEXTRNL(13) := 898.003 ; # 538
PLANT.LOOP1.TEXTRNL(14) := 908.635 ; # 539
PLANT.LOOP1.TEXTRNL(15) := 919.007 ; # 540
PLANT.LOOP1.TEXTRNL(16) := 929.126 ; # 541
PLANT.LOOP1.TEXTRNL(17) := 938.998 ; # 542
PLANT.LOOP1.TEXTRNL(18) := 948.629 ; # 543
PLANT.LOOP1.TEXTRNL(19) := 958.024 ; # 544
PLANT.LOOP1.TEXTRNL(20) := 967.19 ; # 545
PLANT.LOOP1.TEXTRNL(21) := 976.133 ; # 546
PLANT.LOOP1.TEXTRNL(22) := 984.857 ; # 547
PLANT.LOOP1.TEXTRNL(23) := 993.368 ; # 548
PLANT.LOOP1.TEXTRNL(24) := 1001.67 ; # 549
PLANT.LOOP1.TEXTRNL(25) := 1009.77 ; # 550
PLANT.LOOP1.TEXTRNL(26) := 1017.67 ; # 551
PLANT.LOOP1.TEXTRNL(27) := 1025.38 ; # 552
PLANT.LOOP1.TEXTRNL(28) := 1032.9 ; # 553
PLANT.LOOP1.TEXTRNL(29) := 1040.24 ; # 554
PLANT.LOOP1.TEXTRNL(30) := 1047.4 ; # 555
PLANT.LOOP1.TEXTRNL(31) := 1054.38 ; # 556
```

```
PLANT.LOOP1.TEXTRNL(32) := 1061.2 ; # 557
PLANT.LOOP1.TEXTRNL(33) := 1067.84 ; # 558
PLANT.LOOP1.TEXTRNL(34) := 1074.33 ; # 559
PLANT.LOOP1.TEXTRNL(35) := 1080.65 ; # 560
PLANT.LOOP1.TEXTRNL(36) := 1086.83 ; # 561
PLANT.LOOP1.TEXTRNL(37) := 1092.85 ; # 562
PLANT.LOOP1.TEXTRNL(38) := 1098.72 ; # 563
PLANT.LOOP1.TEXTRNL(39) := 1104.45 ; # 564
PLANT.LOOP1.TEXTRNL(40) := 1110.04 ; # 565
PLANT.LOOP1.TEXTRNL(41) := 1115.49 ; # 566
PLANT.LOOP1.TEXTRNL(42) := 1120.82 ; # 567
PLANT.LOOP1.TEXTRNL(43) := 1126.01 ; # 568
PLANT.LOOP1.TEXTRNL(44) := 1131.07 ; # 569
PLANT.LOOP1.TEXTRNL(45) := 1136.01 ; # 570
PLANT.LOOP1.TEXTRNL(46) := 1140.83 ; # 571
PLANT.LOOP1.TEXTRNL(47) := 1145.53 ; # 572
PLANT.LOOP1.TEXTRNL(48) := 1150.12 ; # 573
PLANT.LOOP1.TEXTRNL(49) := 1154.59 ; # 574
PLANT.LOOP1.TEXTRNL(50) := 1158.94 ; # 575
PLANT.LOOP1.TEXTRNL(51) := 291.15 ; # 576
PLANT.LOOP1.TEXTRNL(52) := 291.15 ; # 577
PLANT.LOOP1.TEXTRNL(53) := 291.15 ; # 578
PLANT.LOOP1.TEXTRNL(54) := 291.15 ; # 579
PLANT.LOOP1.TEXTRNL(55) := 291.15 ; # 580
PLANT.LOOP1.TEXTRNL(56) := 291.15 ; # 581
PLANT.LOOP1.TEXTRNL(57) := 291.15 ; # 582
PLANT.LOOP1.TEXTRNL(58) := 291.15 ; # 583
PLANT.LOOP1.TEXTRNL(59) := 291.15 ; # 584
PLANT.LOOP1.TEXTRNL(60) := 291.15 ; # 585
PLANT.LOOP1.TEXTRNL(61) := 1127.77 ; # 586
PLANT.LOOP1.TEXTRNL(62) := 1124.33 ; # 587
PLANT.LOOP1.TEXTRNL(63) := 1120.69 ; # 588
PLANT.LOOP1.TEXTRNL(64) := 1116.87 ; # 589
PLANT.LOOP1.TEXTRNL(65) := 1112.85 ; # 590
PLANT.LOOP1.TEXTRNL(66) := 1108.61 ; # 591
PLANT.LOOP1.TEXTRNL(67) := 1104.16 ; # 592
PLANT.LOOP1.TEXTRNL(68) := 1099.46 ; # 593
PLANT.LOOP1.TEXTRNL(69) := 1094.52 ; # 594
PLANT.LOOP1.TEXTRNL(70) := 1089.32 ; # 595
PLANT.LOOP1.TEXTRNL(71) := 1083.85 ; # 596
PLANT.LOOP1.TEXTRNL(72) := 1078.09 ; # 597
PLANT.LOOP1.TEXTRNL(73) := 1072.03 ; # 598
PLANT.LOOP1.TEXTRNL(74) := 1065.64 ; # 599
PLANT.LOOP1.TEXTRNL(75) := 1058.92 ; # 600
PLANT.LOOP1.TEXTRNL(76) := 1051.85 ; # 601
PLANT.LOOP1.TEXTRNL(77) := 1044.4 ; # 602
PLANT.LOOP1.TEXTRNL(78) := 1036.57 ; # 603
PLANT.LOOP1.TEXTRNL(79) := 1028.32 ; # 604
PLANT.LOOP1.TEXTRNL(80) := 1019.63 ; # 605
PLANT.LOOP1.TEXTRNL(81) := 1010.49 ; # 606
PLANT.LOOP1.TEXTRNL(82) := 1000.87 ; # 607
PLANT.LOOP1.TEXTRNL(83) := 990.74 ; # 608
PLANT.LOOP1.TEXTRNL(84) := 980.078 ; # 609
PLANT.LOOP1.TEXTRNL(85) := 968.854 ; # 610
PLANT.LOOP1.TEXTRNL(86) := 957.041 ; # 611
PLANT.LOOP1.TEXTRNL(87) := 944.605 ; # 612
PLANT.LOOP1.TEXTRNL(88) := 931.514 ; # 613
PLANT.LOOP1.TEXTRNL(89) := 917.735 ; # 614
PLANT.LOOP1.TEXTRNL(90) := 903.231 ; # 615
PLANT.LOOP1.TEXTRNL(91) := 887.962 ; # 616
PLANT.LOOP1.TEXTRNL(92) := 871.891 ; # 617
PLANT.LOOP1.TEXTRNL(93) := 854.973 ; # 618
PLANT.LOOP1.TEXTRNL(94) := 837.165 ; # 619
PLANT.LOOP1.TEXTRNL(95) := 818.42 ; # 620
PLANT.LOOP1.TEXTRNL(96) := 798.688 ; # 621
PLANT.LOOP1.TEXTRNL(97) := 777.917 ; # 622
PLANT.LOOP1.TEXTRNL(98) := 756.054 ; # 623
PLANT.LOOP1.TEXTRNL(99) := 733.039 ; # 624
PLANT.LOOP1.TEXTRNL(100) := 708.82 ; # 625
PLANT.LOOP1.PSI(1) := 1.21853e+006 ; # 626
PLANT.LOOP1.PSI(2) := 1.21407e+006 ; # 627
PLANT.LOOP1.PSI(3) := 1.2096e+006 ; # 628
PLANT.LOOP1.PSI(4) := 1.20513e+006 ; # 629
PLANT.LOOP1.PSI(5) := 1.20066e+006 ; # 630
PLANT.LOOP1.PSI(6) := 1.19618e+006 ; # 631
PLANT.LOOP1.PSI(7) := 1.1917e+006 ; # 632
PLANT.LOOP1.PSI(8) := 1.18721e+006 ; # 633
PLANT.LOOP1.PSI(9) := 1.18271e+006 ; # 634
PLANT.LOOP1.PSI(10) := 1.17821e+006 ; # 635
PLANT.LOOP1.PSI(11) := 1.21421e+006 ; # 636
PLANT.LOOP1.PSI(12) := 1.23642e+006 ; # 637
PLANT.LOOP1.PSI(13) := 1.25693e+006 ; # 638
PLANT.LOOP1.PSI(14) := 1.27623e+006 ; # 639
PLANT.LOOP1.PSI(15) := 1.2944e+006 ; # 640
PLANT.LOOP1.PSI(16) := 1.3115e+006 ; # 641
PLANT.LOOP1.PSI(17) := 1.32762e+006 ; # 642
PLANT.LOOP1.PSI(18) := 1.3428e+006 ; # 643
PLANT.LOOP1.PSI(19) := 1.35711e+006 ; # 644
PLANT.LOOP1.PSI(20) := 1.37059e+006 ; # 645
PLANT.LOOP1.PSI(21) := 1.38329e+006 ; # 646
PLANT.LOOP1.PSI(22) := 1.39525e+006 ; # 647
PLANT.LOOP1.PSI(23) := 1.40653e+006 ; # 648
```

```
PLANT.LOOP1.PSI(24) := 1.41714e+006 ; # 649
PLANT.LOOP1.PSI(25) := 1.42714e+006 ; # 650
PLANT.LOOP1.PSI(26) := 1.43655e+006 ; # 651
PLANT.LOOP1.PSI(27) := 1.4454e+006 ; # 652
PLANT.LOOP1.PSI(28) := 1.45372e+006 ; # 653
PLANT.LOOP1.PSI(29) := 1.46153e+006 ; # 654
PLANT.LOOP1.PSI(30) := 1.46887e+006 ; # 655
PLANT.LOOP1.PSI(31) := 1.47574e+006 ; # 656
PLANT.LOOP1.PSI(32) := 1.48219e+006 ; # 657
PLANT.LOOP1.PSI(33) := 1.48821e+006 ; # 658
PLANT.LOOP1.PSI(34) := 1.49385e+006 ; # 659
PLANT.LOOP1.PSI(35) := 1.4991e+006 ; # 660
PLANT.LOOP1.PSI(36) := 1.50399e+006 ; # 661
PLANT.LOOP1.PSI(37) := 1.50854e+006 ; # 662
PLANT.LOOP1.PSI(38) := 1.51275e+006 ; # 663
PLANT.LOOP1.PSI(39) := 1.51665e+006 ; # 664
PLANT.LOOP1.PSI(40) := 1.52024e+006 ; # 665
PLANT.LOOP1.PSI(41) := 1.52354e+006 ; # 666
PLANT.LOOP1.PSI(42) := 1.52657e+006 ; # 667
PLANT.LOOP1.PSI(43) := 1.52932e+006 ; # 668
PLANT.LOOP1.PSI(44) := 1.53181e+006 ; # 669
PLANT.LOOP1.PSI(45) := 1.53406e+006 ; # 670
PLANT.LOOP1.PSI(46) := 1.53607e+006 ; # 671
PLANT.LOOP1.PSI(47) := 1.53785e+006 ; # 672
PLANT.LOOP1.PSI(48) := 1.53941e+006 ; # 673
PLANT.LOOP1.PSI(49) := 1.54075e+006 ; # 674
PLANT.LOOP1.PSI(50) := 1.54188e+006 ; # 675
PLANT.LOOP1.PSI(51) := 1.51428e+006 ; # 676
PLANT.LOOP1.PSI(52) := 1.50808e+006 ; # 677
PLANT.LOOP1.PSI(53) := 1.50192e+006 ; # 678
PLANT.LOOP1.PSI(54) := 1.49574e+006 ; # 679
PLANT.LOOP1.PSI(55) := 1.48956e+006 ; # 680
PLANT.LOOP1.PSI(56) := 1.48336e+006 ; # 681
PLANT.LOOP1.PSI(57) := 1.47715e+006 ; # 682
PLANT.LOOP1.PSI(58) := 1.47093e+006 ; # 683
PLANT.LOOP1.PSI(59) := 1.46469e+006 ; # 684
PLANT.LOOP1.PSI(60) := 1.45845e+006 ; # 685
PLANT.LOOP1.PSI(61) := 1.47351e+006 ; # 686
PLANT.LOOP1.PSI(62) := 1.46929e+006 ; # 687
PLANT.LOOP1.PSI(63) := 1.46481e+006 ; # 688
PLANT.LOOP1.PSI(64) := 1.46016e+006 ; # 689
PLANT.LOOP1.PSI(65) := 1.45533e+006 ; # 690
PLANT.LOOP1.PSI(66) := 1.4503e+006 ; # 691
PLANT.LOOP1.PSI(67) := 1.44506e+006 ; # 692
PLANT.LOOP1.PSI(68) := 1.4396e+006 ; # 693
PLANT.LOOP1.PSI(69) := 1.4339e+006 ; # 694
PLANT.LOOP1.PSI(70) := 1.42795e+006 ; # 695
PLANT.LOOP1.PSI(71) := 1.42173e+006 ; # 696
PLANT.LOOP1.PSI(72) := 1.41522e+006 ; # 697
PLANT.LOOP1.PSI(73) := 1.4084e+006 ; # 698
PLANT.LOOP1.PSI(74) := 1.40125e+006 ; # 699
PLANT.LOOP1.PSI(75) := 1.39374e+006 ; # 700
PLANT.LOOP1.PSI(76) := 1.38585e+006 ; # 701
PLANT.LOOP1.PSI(77) := 1.37755e+006 ; # 702
PLANT.LOOP1.PSI(78) := 1.36881e+006 ; # 703
PLANT.LOOP1.PSI(79) := 1.35959e+006 ; # 704
PLANT.LOOP1.PSI(80) := 1.34986e+006 ; # 705
PLANT.LOOP1.PSI(81) := 1.33957e+006 ; # 706
PLANT.LOOP1.PSI(82) := 1.32868e+006 ; # 707
PLANT.LOOP1.PSI(83) := 1.31713e+006 ; # 708
PLANT.LOOP1.PSI(84) := 1.30488e+006 ; # 709
PLANT.LOOP1.PSI(85) := 1.29185e+006 ; # 710
PLANT.LOOP1.PSI(86) := 1.27797e+006 ; # 711
PLANT.LOOP1.PSI(87) := 1.26318e+006 ; # 712
PLANT.LOOP1.PSI(88) := 1.24738e+006 ; # 713
PLANT.LOOP1.PSI(89) := 1.23046e+006 ; # 714
PLANT.LOOP1.PSI(90) := 1.21233e+006 ; # 715
PLANT.LOOP1.PSI(91) := 1.19284e+006 ; # 716
PLANT.LOOP1.PSI(92) := 1.17186e+006 ; # 717
PLANT.LOOP1.PSI(93) := 1.14922e+006 ; # 718
PLANT.LOOP1.PSI(94) := 1.12472e+006 ; # 719
PLANT.LOOP1.PSI(95) := 1.09815e+006 ; # 720
PLANT.LOOP1.PSI(96) := 1.06925e+006 ; # 721
PLANT.LOOP1.PSI(97) := 1.03771e+006 ; # 722
PLANT.LOOP1.PSI(98) := 1.00319e+006 ; # 723
PLANT.LOOP1.PSI(99) := 965262 ; # 724
PLANT.LOOP1.PSI(100) := 923433 ; # 725
PLANT.LOOP1.F(1) := 27.5042 ; # 726
PLANT.LOOP1.F(2) := 27.5042 ; # 727
PLANT.LOOP1.F(3) := 27.5042 ; # 728
PLANT.LOOP1.F(4) := 27.5042 ; # 729
PLANT.LOOP1.F(5) := 27.5042 ; # 730
PLANT.LOOP1.F(6) := 27.5042 ; # 731
PLANT.LOOP1.F(7) := 27.5042 ; # 732
PLANT.LOOP1.F(8) := 27.5042 ; # 733
PLANT.LOOP1.F(9) := 27.5042 ; # 734
PLANT.LOOP1.F(10) := 27.5042 ; # 735
PLANT.LOOP1.F(11) := 27.5042 ; # 736
PLANT.LOOP1.F(12) := 27.5042 ; # 737
PLANT.LOOP1.F(13) := 27.5042 ; # 738
PLANT.LOOP1.F(14) := 27.5042 ; # 739
PLANT.LOOP1.F(15) := 27.5042 ; # 740
```

```
PLANT.LOOP1.F(16) := 27.5042 ; # 741
PLANT.LOOP1.F(17) := 27.5042 ; # 742
PLANT.LOOP1.F(18) := 27.5042 ; # 743
PLANT.LOOP1.F(19) := 27.5042 ; # 744
PLANT.LOOP1.F(20) := 27.5042 ; # 745
PLANT.LOOP1.F(21) := 27.5042 ; # 746
PLANT.LOOP1.F(22) := 27.5042 ; # 747
PLANT.LOOP1.F(23) := 27.5042 ; # 748
PLANT.LOOP1.F(24) := 27.5042 ; # 749
PLANT.LOOP1.F(25) := 27.5042 ; # 750
PLANT.LOOP1.F(26) := 27.5042 ; # 751
PLANT.LOOP1.F(27) := 27.5042 ; # 752
PLANT.LOOP1.F(28) := 27.5042 ; # 753
PLANT.LOOP1.F(29) := 27.5042 ; # 754
PLANT.LOOP1.F(30) := 27.5042 ; # 755
PLANT.LOOP1.F(31) := 27.5042 ; # 756
PLANT.LOOP1.F(32) := 27.5042 ; # 757
PLANT.LOOP1.F(33) := 27.5042 ; # 758
PLANT.LOOP1.F(34) := 27.5042 ; # 759
PLANT.LOOP1.F(35) := 27.5042 ; # 760
PLANT.LOOP1.F(36) := 27.5042 ; # 761
PLANT.LOOP1.F(37) := 27.5042 ; # 762
PLANT.LOOP1.F(38) := 27.5042 ; # 763
PLANT.LOOP1.F(39) := 27.5042 ; # 764
PLANT.LOOP1.F(40) := 27.5042 ; # 765
PLANT.LOOP1.F(41) := 27.5042 ; # 766
PLANT.LOOP1.F(42) := 27.5042 ; # 767
PLANT.LOOP1.F(43) := 27.5042 ; # 768
PLANT.LOOP1.F(44) := 27.5042 ; # 769
PLANT.LOOP1.F(45) := 27.5042 ; # 770
PLANT.LOOP1.F(46) := 27.5042 ; # 771
PLANT.LOOP1.F(47) := 27.5042 ; # 772
PLANT.LOOP1.F(48) := 27.5042 ; # 773
PLANT.LOOP1.F(49) := 27.5042 ; # 774
PLANT.LOOP1.F(50) := 27.5042 ; # 775
PLANT.LOOP1.F(51) := 27.5042 ; # 776
PLANT.LOOP1.F(52) := 27.5042 ; # 777
PLANT.LOOP1.F(53) := 27.5042 ; # 778
PLANT.LOOP1.F(54) := 27.5042 ; # 779
PLANT.LOOP1.F(55) := 27.5042 ; # 780
PLANT.LOOP1.F(56) := 27.5042 ; # 781
PLANT.LOOP1.F(57) := 27.5042 ; # 782
PLANT.LOOP1.F(58) := 27.5042 ; # 783
PLANT.LOOP1.F(59) := 27.5042 ; # 784
PLANT.LOOP1.F(60) := 27.5042 ; # 785
PLANT.LOOP1.F(61) := 27.5042 ; # 786
PLANT.LOOP1.F(62) := 27.5042 ; # 787
PLANT.LOOP1.F(63) := 27.5042 ; # 788
PLANT.LOOP1.F(64) := 27.5042 ; # 789
PLANT.LOOP1.F(65) := 27.5042 ; # 790
PLANT.LOOP1.F(66) := 27.5042 ; # 791
PLANT.LOOP1.F(67) := 27.5042 ; # 792
PLANT.LOOP1.F(68) := 27.5042 ; # 793
PLANT.LOOP1.F(69) := 27.5042 ; # 794
PLANT.LOOP1.F(70) := 27.5042 ; # 795
PLANT.LOOP1.F(71) := 27.5042 ; # 796
PLANT.LOOP1.F(72) := 27.5042 ; # 797
PLANT.LOOP1.F(73) := 27.5042 ; # 798
PLANT.LOOP1.F(74) := 27.5042 ; # 799
PLANT.LOOP1.F(75) := 27.5042 ; # 800
PLANT.LOOP1.F(76) := 27.5042 ; # 801
PLANT.LOOP1.F(77) := 27.5042 ; # 802
PLANT.LOOP1.F(78) := 27.5042 ; # 803
PLANT.LOOP1.F(79) := 27.5042 ; # 804
PLANT.LOOP1.F(80) := 27.5042 ; # 805
PLANT.LOOP1.F(81) := 27.5042 ; # 806
PLANT.LOOP1.F(82) := 27.5042 ; # 807
PLANT.LOOP1.F(83) := 27.5042 ; # 808
PLANT.LOOP1.F(84) := 27.5042 ; # 809
PLANT.LOOP1.F(85) := 27.5042 ; # 810
PLANT.LOOP1.F(86) := 27.5042 ; # 811
PLANT.LOOP1.F(87) := 27.5042 ; # 812
PLANT.LOOP1.F(88) := 27.5042 ; # 813
PLANT.LOOP1.F(89) := 27.5042 ; # 814
PLANT.LOOP1.F(90) := 27.5042 ; # 815
PLANT.LOOP1.F(91) := 27.5042 ; # 816
PLANT.LOOP1.F(92) := 27.5042 ; # 817
PLANT.LOOP1.F(93) := 27.5042 ; # 818
PLANT.LOOP1.F(94) := 27.5042 ; # 819
PLANT.LOOP1.F(95) := 27.5042 ; # 820
PLANT.LOOP1.F(96) := 27.5042 ; # 821
PLANT.LOOP1.F(97) := 27.5042 ; # 822
PLANT.LOOP1.F(98) := 27.5042 ; # 823
PLANT.LOOP1.F(99) := 27.5042 ; # 824
PLANT.LOOP1.F(100) := 27.5042 ; # 825
PLANT.LOOP1.F_IN := 27.5042 ; # 826
PLANT.LOOP1.TOTALMASS := 27.0521 ; # 827
PLANT.LOOP1.RHOEXTRA(1) := 1.15546 ; # 828
PLANT.LOOP1.RHOEXTRA(2) := 1.15546 ; # 829
PLANT.LOOP1.PEXTRA(1) := 2.01632e+006 ; # 830
PLANT.LOOP1.PEXTRA(2) := 2.01632e+006 ; # 831
PLANT.LOOP1.HEXTRA(1) := 2.81342e+006 ; # 832
```

```
PLANT.LOOP1.HEXTRA(2) := 2.81342e+006 ; # 833
PLANT.LOOP1.TEXTRA(1) := 839.563 ; # 834
PLANT.LOOP1.TEXTRA(2) := 839.563 ; # 835
PLANT.LOOP1.PSIEXTRA(1) := 1.23448e+006 ; # 836
PLANT.LOOP1.PSIEXTRA(2) := 1.23448e+006 ; # 837
PLANT.LOOP1.FEXTRA(1) := 27.5042 ; # 838
PLANT.LOOP1.FEXTRA(2) := 27.5042 ; # 839
PLANT.LOOP1.HT1.H2 := 17.344 ; # 840
PLANT.LOOP1.HT1.RHOPIPEHE := 1.12889 ; # 841
PLANT.LOOP1.HT1.VPIPE := 181.878 ; # 842
PLANT.LOOP1.HT1.H0 := 1554.44 ; # 843
PLANT.LOOP1.HT1.U := 8.93607 ; # 844
PLANT.LOOP1.HT2.RHOPIPEHE := 0.898512 ; # 845
PLANT.LOOP1.HT2.VPIPE := 89.1714 ; # 846
PLANT.LOOP1.HT2.H0 := 1665.62 ; # 847
PLANT.LOOP1.HT2.U := 1645.45 ; # 848
PLANT.LOOP1.HT3.H2 := 17.344 ; # 849
PLANT.LOOP1.HT3.RHOPIPEHE := 0.736935 ; # 850
PLANT.LOOP1.HT3.VPIPE := 241.071 ; # 851
PLANT.LOOP1.HT3.H0 := 1364.58 ; # 852
PLANT.LOOP1.HT3.U := 10.3601 ; # 853
PLANT.LOOP1.HT4.RHOPIPEHE := 0.821936 ; # 854
PLANT.LOOP1.HT4.VPIPE := 84.1899 ; # 855
PLANT.LOOP1.HT4.U := 1299.78 ; # 856
PLANT.HX2.TGASHOT(1) := 1137.66 ; # 1143
PLANT.HX2.TGASHOT(2) := 1134.74 ; # 1144
PLANT.HX2.TGASHOT(3) := 1131.65 ; # 1145
PLANT.HX2.TGASHOT(4) := 1128.41 ; # 1146
PLANT.HX2.TGASHOT(5) := 1124.99 ; # 1147
PLANT.HX2.TGASHOT(6) := 1121.4 ; # 1148
PLANT.HX2.TGASHOT(7) := 1117.61 ; # 1149
PLANT.HX2.TGASHOT(8) := 1113.63 ; # 1150
PLANT.HX2.TGASHOT(9) := 1109.43 ; # 1151
PLANT.HX2.TGASHOT(10) := 1105.02 ; # 1152
PLANT.HX2.TGASHOT(11) := 1100.37 ; # 1153
PLANT.HX2.TGASHOT(12) := 1095.48 ; # 1154
PLANT.HX2.TGASHOT(13) := 1090.33 ; # 1155
PLANT.HX2.TGASHOT(14) := 1084.91 ; # 1156
PLANT.HX2.TGASHOT(15) := 1079.2 ; # 1157
PLANT.HX2.TGASHOT(16) := 1073.2 ; # 1158
PLANT.HX2.TGASHOT(17) := 1066.88 ; # 1159
PLANT.HX2.TGASHOT(18) := 1060.22 ; # 1160
PLANT.HX2.TGASHOT(19) := 1053.22 ; # 1161
PLANT.HX2.TGASHOT(20) := 1045.84 ; # 1162
PLANT.HX2.TGASHOT(21) := 1038.08 ; # 1163
PLANT.HX2.TGASHOT(22) := 1029.91 ; # 1164
PLANT.HX2.TGASHOT(23) := 1021.31 ; # 1165
PLANT.HX2.TGASHOT(24) := 1012.26 ; # 1166
PLANT.HX2.TGASHOT(25) := 1002.73 ; # 1167
PLANT.HX2.TGASHOT(26) := 992.699 ; # 1168
PLANT.HX2.TGASHOT(27) := 982.14 ; # 1169
PLANT.HX2.TGASHOT(28) := 971.025 ; # 1170
PLANT.HX2.TGASHOT(29) := 959.325 ; # 1171
PLANT.HX2.TGASHOT(30) := 947.01 ; # 1172
PLANT.HX2.TGASHOT(31) := 934.046 ; # 1173
PLANT.HX2.TGASHOT(32) := 920.4 ; # 1174
PLANT.HX2.TGASHOT(33) := 906.036 ; # 1175
PLANT.HX2.TGASHOT(34) := 890.915 ; # 1176
PLANT.HX2.TGASHOT(35) := 874.999 ; # 1177
PLANT.HX2.TGASHOT(36) := 858.245 ; # 1178
PLANT.HX2.TGASHOT(37) := 840.609 ; # 1179
PLANT.HX2.TGASHOT(38) := 822.045 ; # 1180
PLANT.HX2.TGASHOT(39) := 802.504 ; # 1181
PLANT.HX2.TGASHOT(40) := 781.936 ; # 1182
PLANT.HX2.TGASCOLD(1) := 1118.89 ; # 1183
PLANT.HX2.TGASCOLD(2) := 1114.97 ; # 1184
PLANT.HX2.TGASCOLD(3) := 1110.85 ; # 1185
PLANT.HX2.TGASCOLD(4) := 1106.51 ; # 1186
PLANT.HX2.TGASCOLD(5) := 1101.94 ; # 1187
PLANT.HX2.TGASCOLD(6) := 1097.13 ; # 1188
PLANT.HX2.TGASCOLD(7) := 1092.07 ; # 1189
PLANT.HX2.TGASCOLD(8) := 1086.74 ; # 1190
PLANT.HX2.TGASCOLD(9) := 1081.13 ; # 1191
PLANT.HX2.TGASCOLD(10) := 1075.22 ; # 1192
PLANT.HX2.TGASCOLD(11) := 1069.01 ; # 1193
PLANT.HX2.TGASCOLD(12) := 1062.47 ; # 1194
PLANT.HX2.TGASCOLD(13) := 1055.58 ; # 1195
PLANT.HX2.TGASCOLD(14) := 1048.33 ; # 1196
PLANT.HX2.TGASCOLD(15) := 1040.7 ; # 1197
PLANT.HX2.TGASCOLD(16) := 1032.67 ; # 1198
PLANT.HX2.TGASCOLD(17) := 1024.21 ; # 1199
PLANT.HX2.TGASCOLD(18) := 1015.31 ; # 1200
PLANT.HX2.TGASCOLD(19) := 1005.94 ; # 1201
PLANT.HX2.TGASCOLD(20) := 996.081 ; # 1202
PLANT.HX2.TGASCOLD(21) := 985.7 ; # 1203
PLANT.HX2.TGASCOLD(22) := 974.772 ; # 1204
PLANT.HX2.TGASCOLD(23) := 963.27 ; # 1205
PLANT.HX2.TGASCOLD(24) := 951.162 ; # 1206
PLANT.HX2.TGASCOLD(25) := 938.417 ; # 1207
PLANT.HX2.TGASCOLD(26) := 925.001 ; # 1208
PLANT.HX2.TGASCOLD(27) := 910.879 ; # 1209
PLANT.HX2.TGASCOLD(28) := 896.013 ; # 1210
```

```
927   PLANT.HX2.TGASCOLD(29) := 880.365 ; # 1211
928   PLANT.HX2.TGASCOLD(30) := 863.894 ; # 1212
929   PLANT.HX2.TGASCOLD(31) := 846.555 ; # 1213
930   PLANT.HX2.TGASCOLD(32) := 828.304 ; # 1214
931   PLANT.HX2.TGASCOLD(33) := 809.093 ; # 1215
932   PLANT.HX2.TGASCOLD(34) := 788.87 ; # 1216
933   PLANT.HX2.TGASCOLD(35) := 767.582 ; # 1217
934   PLANT.HX2.TGASCOLD(36) := 745.175 ; # 1218
935   PLANT.HX2.TGASCOLD(37) := 721.587 ; # 1219
936   PLANT.HX2.TGASCOLD(38) := 696.759 ; # 1220
937   PLANT.HX2.TGASCOLD(39) := 670.623 ; # 1221
938   PLANT.HX2.TGASCOLD(40) := 643.112 ; # 1222
939   PLANT.HX2.UHOT := 1299.78 ; # 1223
940   PLANT.HX2.UCOLD := 1880.53 ; # 1224
941   PLANT.HX2.THX(1) := 1127.77 ; # 1225
942   PLANT.HX2.THX(2) := 1124.33 ; # 1226
943   PLANT.HX2.THX(3) := 1120.69 ; # 1227
944   PLANT.HX2.THX(4) := 1116.87 ; # 1228
945   PLANT.HX2.THX(5) := 1112.85 ; # 1229
946   PLANT.HX2.THX(6) := 1108.61 ; # 1230
947   PLANT.HX2.THX(7) := 1104.16 ; # 1231
948   PLANT.HX2.THX(8) := 1099.46 ; # 1232
949   PLANT.HX2.THX(9) := 1094.52 ; # 1233
950   PLANT.HX2.THX(10) := 1089.32 ; # 1234
951   PLANT.HX2.THX(11) := 1083.85 ; # 1235
952   PLANT.HX2.THX(12) := 1078.09 ; # 1236
953   PLANT.HX2.THX(13) := 1072.03 ; # 1237
954   PLANT.HX2.THX(14) := 1065.64 ; # 1238
955   PLANT.HX2.THX(15) := 1058.92 ; # 1239
956   PLANT.HX2.THX(16) := 1051.85 ; # 1240
957   PLANT.HX2.THX(17) := 1044.4 ; # 1241
958   PLANT.HX2.THX(18) := 1036.57 ; # 1242
959   PLANT.HX2.THX(19) := 1028.32 ; # 1243
960   PLANT.HX2.THX(20) := 1019.63 ; # 1244
961   PLANT.HX2.THX(21) := 1010.49 ; # 1245
962   PLANT.HX2.THX(22) := 1000.87 ; # 1246
963   PLANT.HX2.THX(23) := 990.74 ; # 1247
964   PLANT.HX2.THX(24) := 980.078 ; # 1248
965   PLANT.HX2.THX(25) := 968.854 ; # 1249
966   PLANT.HX2.THX(26) := 957.041 ; # 1250
967   PLANT.HX2.THX(27) := 944.605 ; # 1251
968   PLANT.HX2.THX(28) := 931.514 ; # 1252
969   PLANT.HX2.THX(29) := 917.735 ; # 1253
970   PLANT.HX2.THX(30) := 903.231 ; # 1254
971   PLANT.HX2.THX(31) := 887.962 ; # 1255
972   PLANT.HX2.THX(32) := 871.891 ; # 1256
973   PLANT.HX2.THX(33) := 854.973 ; # 1257
974   PLANT.HX2.THX(34) := 837.165 ; # 1258
975   PLANT.HX2.THX(35) := 818.42 ; # 1259
976   PLANT.HX2.THX(36) := 798.688 ; # 1260
977   PLANT.HX2.THX(37) := 777.917 ; # 1261
978   PLANT.HX2.THX(38) := 756.054 ; # 1262
979   PLANT.HX2.THX(39) := 733.039 ; # 1263
980   PLANT.HX2.THX(40) := 708.82 ; # 1264
981   PLANT.PIPEC2.RHO(1) := 3.77479 ; # 1265
982   PLANT.PIPEC2.RHO(2) := 3.61285 ; # 1266
983   PLANT.PIPEC2.RHO(3) := 3.47031 ; # 1267
984   PLANT.PIPEC2.RHO(4) := 3.34393 ; # 1268
985   PLANT.PIPEC2.RHO(5) := 3.23116 ; # 1269
986   PLANT.PIPEC2.RHO(6) := 3.12995 ; # 1270
987   PLANT.PIPEC2.RHO(7) := 3.03864 ; # 1271
988   PLANT.PIPEC2.RHO(8) := 2.95586 ; # 1272
989   PLANT.PIPEC2.RHO(9) := 2.8805 ; # 1273
990   PLANT.PIPEC2.RHO(10) := 2.81162 ; # 1274
991   PLANT.PIPEC2.RHO(11) := 2.74842 ; # 1275
992   PLANT.PIPEC2.RHO(12) := 2.69025 ; # 1276
993   PLANT.PIPEC2.RHO(13) := 2.63652 ; # 1277
994   PLANT.PIPEC2.RHO(14) := 2.58675 ; # 1278
995   PLANT.PIPEC2.RHO(15) := 2.54053 ; # 1279
996   PLANT.PIPEC2.RHO(16) := 2.49748 ; # 1280
997   PLANT.PIPEC2.RHO(17) := 2.45729 ; # 1281
998   PLANT.PIPEC2.RHO(18) := 2.41967 ; # 1282
999   PLANT.PIPEC2.RHO(19) := 2.38439 ; # 1283
1000  PLANT.PIPEC2.RHO(20) := 2.35122 ; # 1284
1001  PLANT.PIPEC2.RHO(21) := 2.31998 ; # 1285
1002  PLANT.PIPEC2.RHO(22) := 2.29049 ; # 1286
1003  PLANT.PIPEC2.RHO(23) := 2.2626 ; # 1287
1004  PLANT.PIPEC2.RHO(24) := 2.23618 ; # 1288
1005  PLANT.PIPEC2.RHO(25) := 2.2111 ; # 1289
1006  PLANT.PIPEC2.RHO(26) := 2.18725 ; # 1290
1007  PLANT.PIPEC2.RHO(27) := 2.16453 ; # 1291
1008  PLANT.PIPEC2.RHO(28) := 2.14286 ; # 1292
1009  PLANT.PIPEC2.RHO(29) := 2.12215 ; # 1293
1010  PLANT.PIPEC2.RHO(30) := 2.10233 ; # 1294
1011  PLANT.PIPEC2.RHO(31) := 2.08332 ; # 1295
1012  PLANT.PIPEC2.RHO(32) := 2.06508 ; # 1296
1013  PLANT.PIPEC2.RHO(33) := 2.04754 ; # 1297
1014  PLANT.PIPEC2.RHO(34) := 2.03064 ; # 1298
1015  PLANT.PIPEC2.RHO(35) := 2.01436 ; # 1299
1016  PLANT.PIPEC2.RHO(36) := 1.99863 ; # 1300
1017  PLANT.PIPEC2.RHO(37) := 1.98342 ; # 1301
1018  PLANT.PIPEC2.RHO(38) := 1.96869 ; # 1302
```

```
PLANT.PIPEC2.RHO(39) := 1.95441 ; # 1303
PLANT.PIPEC2.RHO(40) := 1.94055 ; # 1304
PLANT.PIPEC2.P(1) := 1.12004e+006 ; # 1305
PLANT.PIPEC2.P(2) := 1.11785e+006 ; # 1306
PLANT.PIPEC2.P(3) := 1.11559e+006 ; # 1307
PLANT.PIPEC2.P(4) := 1.11327e+006 ; # 1308
PLANT.PIPEC2.P(5) := 1.11089e+006 ; # 1309
PLANT.PIPEC2.P(6) := 1.10845e+006 ; # 1310
PLANT.PIPEC2.P(7) := 1.10596e+006 ; # 1311
PLANT.PIPEC2.P(8) := 1.10341e+006 ; # 1312
PLANT.PIPEC2.P(9) := 1.10081e+006 ; # 1313
PLANT.PIPEC2.P(10) := 1.09816e+006 ; # 1314
PLANT.PIPEC2.P(11) := 1.09547e+006 ; # 1315
PLANT.PIPEC2.P(12) := 1.09272e+006 ; # 1316
PLANT.PIPEC2.P(13) := 1.08994e+006 ; # 1317
PLANT.PIPEC2.P(14) := 1.0871e+006 ; # 1318
PLANT.PIPEC2.P(15) := 1.08423e+006 ; # 1319
PLANT.PIPEC2.P(16) := 1.08132e+006 ; # 1320
PLANT.PIPEC2.P(17) := 1.07837e+006 ; # 1321
PLANT.PIPEC2.P(18) := 1.07538e+006 ; # 1322
PLANT.PIPEC2.P(19) := 1.07235e+006 ; # 1323
PLANT.PIPEC2.P(20) := 1.06929e+006 ; # 1324
PLANT.PIPEC2.P(21) := 1.06619e+006 ; # 1325
PLANT.PIPEC2.P(22) := 1.06306e+006 ; # 1326
PLANT.PIPEC2.P(23) := 1.05989e+006 ; # 1327
PLANT.PIPEC2.P(24) := 1.0567e+006 ; # 1328
PLANT.PIPEC2.P(25) := 1.05347e+006 ; # 1329
PLANT.PIPEC2.P(26) := 1.05022e+006 ; # 1330
PLANT.PIPEC2.P(27) := 1.04693e+006 ; # 1331
PLANT.PIPEC2.P(28) := 1.04361e+006 ; # 1332
PLANT.PIPEC2.P(29) := 1.04027e+006 ; # 1333
PLANT.PIPEC2.P(30) := 1.0369e+006 ; # 1334
PLANT.PIPEC2.P(31) := 1.0335e+006 ; # 1335
PLANT.PIPEC2.P(32) := 1.03008e+006 ; # 1336
PLANT.PIPEC2.P(33) := 1.02662e+006 ; # 1337
PLANT.PIPEC2.P(34) := 1.02315e+006 ; # 1338
PLANT.PIPEC2.P(35) := 1.01965e+006 ; # 1339
PLANT.PIPEC2.P(36) := 1.01612e+006 ; # 1340
PLANT.PIPEC2.P(37) := 1.01257e+006 ; # 1341
PLANT.PIPEC2.P(38) := 1.00899e+006 ; # 1342
PLANT.PIPEC2.P(39) := 1.00539e+006 ; # 1343
PLANT.PIPEC2.P(40) := 1.00177e+006 ; # 1344
PLANT.PIPEC2.H(1) := 778513 ; # 1345
PLANT.PIPEC2.H(2) := 840573 ; # 1346
PLANT.PIPEC2.H(3) := 899531 ; # 1347
PLANT.PIPEC2.H(4) := 955540 ; # 1348
PLANT.PIPEC2.H(5) := 1.00875e+006 ; # 1349
PLANT.PIPEC2.H(6) := 1.0593e+006 ; # 1350
PLANT.PIPEC2.H(7) := 1.10732e+006 ; # 1351
PLANT.PIPEC2.H(8) := 1.15294e+006 ; # 1352
PLANT.PIPEC2.H(9) := 1.19627e+006 ; # 1353
PLANT.PIPEC2.H(10) := 1.23745e+006 ; # 1354
PLANT.PIPEC2.H(11) := 1.27656e+006 ; # 1355
PLANT.PIPEC2.H(12) := 1.31372e+006 ; # 1356
PLANT.PIPEC2.H(13) := 1.34901e+006 ; # 1357
PLANT.PIPEC2.H(14) := 1.38255e+006 ; # 1358
PLANT.PIPEC2.H(15) := 1.41441e+006 ; # 1359
PLANT.PIPEC2.H(16) := 1.44467e+006 ; # 1360
PLANT.PIPEC2.H(17) := 1.47342e+006 ; # 1361
PLANT.PIPEC2.H(18) := 1.50073e+006 ; # 1362
PLANT.PIPEC2.H(19) := 1.52668e+006 ; # 1363
PLANT.PIPEC2.H(20) := 1.55133e+006 ; # 1364
PLANT.PIPEC2.H(21) := 1.57475e+006 ; # 1365
PLANT.PIPEC2.H(22) := 1.597e+006 ; # 1366
PLANT.PIPEC2.H(23) := 1.61813e+006 ; # 1367
PLANT.PIPEC2.H(24) := 1.63821e+006 ; # 1368
PLANT.PIPEC2.H(25) := 1.65728e+006 ; # 1369
PLANT.PIPEC2.H(26) := 1.6754e+006 ; # 1370
PLANT.PIPEC2.H(27) := 1.69261e+006 ; # 1371
PLANT.PIPEC2.H(28) := 1.70897e+006 ; # 1372
PLANT.PIPEC2.H(29) := 1.7245e+006 ; # 1373
PLANT.PIPEC2.H(30) := 1.73926e+006 ; # 1374
PLANT.PIPEC2.H(31) := 1.75328e+006 ; # 1375
PLANT.PIPEC2.H(32) := 1.7666e+006 ; # 1376
PLANT.PIPEC2.H(33) := 1.77926e+006 ; # 1377
PLANT.PIPEC2.H(34) := 1.79128e+006 ; # 1378
PLANT.PIPEC2.H(35) := 1.8027e+006 ; # 1379
PLANT.PIPEC2.H(36) := 1.81355e+006 ; # 1380
PLANT.PIPEC2.H(37) := 1.82385e+006 ; # 1381
PLANT.PIPEC2.H(38) := 1.83364e+006 ; # 1382
PLANT.PIPEC2.H(39) := 1.84294e+006 ; # 1383
PLANT.PIPEC2.H(40) := 1.85178e+006 ; # 1384
PLANT.PIPEC2.H0 := 713180 ; # 1385
PLANT.PIPEC2.T(1) := 643.112 ; # 1386
PLANT.PIPEC2.T(2) := 670.623 ; # 1387
PLANT.PIPEC2.T(3) := 696.759 ; # 1388
PLANT.PIPEC2.T(4) := 721.587 ; # 1389
PLANT.PIPEC2.T(5) := 745.175 ; # 1390
PLANT.PIPEC2.T(6) := 767.582 ; # 1391
PLANT.PIPEC2.T(7) := 788.87 ; # 1392
PLANT.PIPEC2.T(8) := 809.093 ; # 1393
PLANT.PIPEC2.T(9) := 828.304 ; # 1394
```

```
PLANT.PIPEC2.T(10) := 846.555 ; # 1395
PLANT.PIPEC2.T(11) := 863.894 ; # 1396
PLANT.PIPEC2.T(12) := 880.365 ; # 1397
PLANT.PIPEC2.T(13) := 896.013 ; # 1398
PLANT.PIPEC2.T(14) := 910.879 ; # 1399
PLANT.PIPEC2.T(15) := 925.001 ; # 1400
PLANT.PIPEC2.T(16) := 938.417 ; # 1401
PLANT.PIPEC2.T(17) := 951.162 ; # 1402
PLANT.PIPEC2.T(18) := 963.27 ; # 1403
PLANT.PIPEC2.T(19) := 974.772 ; # 1404
PLANT.PIPEC2.T(20) := 985.7 ; # 1405
PLANT.PIPEC2.T(21) := 996.081 ; # 1406
PLANT.PIPEC2.T(22) := 1005.94 ; # 1407
PLANT.PIPEC2.T(23) := 1015.31 ; # 1408
PLANT.PIPEC2.T(24) := 1024.21 ; # 1409
PLANT.PIPEC2.T(25) := 1032.67 ; # 1410
PLANT.PIPEC2.T(26) := 1040.7 ; # 1411
PLANT.PIPEC2.T(27) := 1048.33 ; # 1412
PLANT.PIPEC2.T(28) := 1055.58 ; # 1413
PLANT.PIPEC2.T(29) := 1062.47 ; # 1414
PLANT.PIPEC2.T(30) := 1069.01 ; # 1415
PLANT.PIPEC2.T(31) := 1075.22 ; # 1416
PLANT.PIPEC2.T(32) := 1081.13 ; # 1417
PLANT.PIPEC2.T(33) := 1086.74 ; # 1418
PLANT.PIPEC2.T(34) := 1092.07 ; # 1419
PLANT.PIPEC2.T(35) := 1097.13 ; # 1420
PLANT.PIPEC2.T(36) := 1101.94 ; # 1421
PLANT.PIPEC2.T(37) := 1106.51 ; # 1422
PLANT.PIPEC2.T(38) := 1110.85 ; # 1423
PLANT.PIPEC2.T(39) := 1114.97 ; # 1424
PLANT.PIPEC2.T(40) := 1118.89 ; # 1425
PLANT.PIPEC2.TEXTRNL(1) := 708.82 ; # 1426
PLANT.PIPEC2.TEXTRNL(2) := 733.039 ; # 1427
PLANT.PIPEC2.TEXTRNL(3) := 756.054 ; # 1428
PLANT.PIPEC2.TEXTRNL(4) := 777.917 ; # 1429
PLANT.PIPEC2.TEXTRNL(5) := 798.688 ; # 1430
PLANT.PIPEC2.TEXTRNL(6) := 818.42 ; # 1431
PLANT.PIPEC2.TEXTRNL(7) := 837.165 ; # 1432
PLANT.PIPEC2.TEXTRNL(8) := 854.973 ; # 1433
PLANT.PIPEC2.TEXTRNL(9) := 871.891 ; # 1434
PLANT.PIPEC2.TEXTRNL(10) := 887.962 ; # 1435
PLANT.PIPEC2.TEXTRNL(11) := 903.231 ; # 1436
PLANT.PIPEC2.TEXTRNL(12) := 917.735 ; # 1437
PLANT.PIPEC2.TEXTRNL(13) := 931.514 ; # 1438
PLANT.PIPEC2.TEXTRNL(14) := 944.605 ; # 1439
PLANT.PIPEC2.TEXTRNL(15) := 957.041 ; # 1440
PLANT.PIPEC2.TEXTRNL(16) := 968.854 ; # 1441
PLANT.PIPEC2.TEXTRNL(17) := 980.078 ; # 1442
PLANT.PIPEC2.TEXTRNL(18) := 990.74 ; # 1443
PLANT.PIPEC2.TEXTRNL(19) := 1000.87 ; # 1444
PLANT.PIPEC2.TEXTRNL(20) := 1010.49 ; # 1445
PLANT.PIPEC2.TEXTRNL(21) := 1019.63 ; # 1446
PLANT.PIPEC2.TEXTRNL(22) := 1028.32 ; # 1447
PLANT.PIPEC2.TEXTRNL(23) := 1036.57 ; # 1448
PLANT.PIPEC2.TEXTRNL(24) := 1044.4 ; # 1449
PLANT.PIPEC2.TEXTRNL(25) := 1051.85 ; # 1450
PLANT.PIPEC2.TEXTRNL(26) := 1058.92 ; # 1451
PLANT.PIPEC2.TEXTRNL(27) := 1065.64 ; # 1452
PLANT.PIPEC2.TEXTRNL(28) := 1072.03 ; # 1453
PLANT.PIPEC2.TEXTRNL(29) := 1078.09 ; # 1454
PLANT.PIPEC2.TEXTRNL(30) := 1083.85 ; # 1455
PLANT.PIPEC2.TEXTRNL(31) := 1089.32 ; # 1456
PLANT.PIPEC2.TEXTRNL(32) := 1094.52 ; # 1457
PLANT.PIPEC2.TEXTRNL(33) := 1099.46 ; # 1458
PLANT.PIPEC2.TEXTRNL(34) := 1104.16 ; # 1459
PLANT.PIPEC2.TEXTRNL(35) := 1108.61 ; # 1460
PLANT.PIPEC2.TEXTRNL(36) := 1112.85 ; # 1461
PLANT.PIPEC2.TEXTRNL(37) := 1116.87 ; # 1462
PLANT.PIPEC2.TEXTRNL(38) := 1120.69 ; # 1463
PLANT.PIPEC2.TEXTRNL(39) := 1124.33 ; # 1464
PLANT.PIPEC2.TEXTRNL(40) := 1127.77 ; # 1465
PLANT.PIPEC2.F(1) := 46.175 ; # 1466
PLANT.PIPEC2.F(2) := 46.175 ; # 1467
PLANT.PIPEC2.F(3) := 46.175 ; # 1468
PLANT.PIPEC2.F(4) := 46.175 ; # 1469
PLANT.PIPEC2.F(5) := 46.175 ; # 1470
PLANT.PIPEC2.F(6) := 46.175 ; # 1471
PLANT.PIPEC2.F(7) := 46.175 ; # 1472
PLANT.PIPEC2.F(8) := 46.175 ; # 1473
PLANT.PIPEC2.F(9) := 46.175 ; # 1474
PLANT.PIPEC2.F(10) := 46.175 ; # 1475
PLANT.PIPEC2.F(11) := 46.175 ; # 1476
PLANT.PIPEC2.F(12) := 46.175 ; # 1477
PLANT.PIPEC2.F(13) := 46.175 ; # 1478
PLANT.PIPEC2.F(14) := 46.175 ; # 1479
PLANT.PIPEC2.F(15) := 46.175 ; # 1480
PLANT.PIPEC2.F(16) := 46.175 ; # 1481
PLANT.PIPEC2.F(17) := 46.175 ; # 1482
PLANT.PIPEC2.F(18) := 46.175 ; # 1483
PLANT.PIPEC2.F(19) := 46.175 ; # 1484
PLANT.PIPEC2.F(20) := 46.175 ; # 1485
PLANT.PIPEC2.F(21) := 46.175 ; # 1486
```

```
PLANT.PIPEC2.F(22) := 46.175 ; # 1487
PLANT.PIPEC2.F(23) := 46.175 ; # 1488
PLANT.PIPEC2.F(24) := 46.175 ; # 1489
PLANT.PIPEC2.F(25) := 46.175 ; # 1490
PLANT.PIPEC2.F(26) := 46.175 ; # 1491
PLANT.PIPEC2.F(27) := 46.175 ; # 1492
PLANT.PIPEC2.F(28) := 46.175 ; # 1493
PLANT.PIPEC2.F(29) := 46.175 ; # 1494
PLANT.PIPEC2.F(30) := 46.175 ; # 1495
PLANT.PIPEC2.F(31) := 46.175 ; # 1496
PLANT.PIPEC2.F(32) := 46.175 ; # 1497
PLANT.PIPEC2.F(33) := 46.175 ; # 1498
PLANT.PIPEC2.F(34) := 46.175 ; # 1499
PLANT.PIPEC2.F(35) := 46.175 ; # 1500
PLANT.PIPEC2.F(36) := 46.175 ; # 1501
PLANT.PIPEC2.F(37) := 46.175 ; # 1502
PLANT.PIPEC2.F(38) := 46.175 ; # 1503
PLANT.PIPEC2.F(39) := 46.175 ; # 1504
PLANT.PIPEC2.F(40) := 46.175 ; # 1505
PLANT.PIPEC2.F_IN := 46.175 ; # 1506
PLANT.PIPEC2.PSI(1) := 1.81868e+006 ; # 1507
PLANT.PIPEC2.PSI(2) := 1.91901e+006 ; # 1508
PLANT.PIPEC2.PSI(3) := 2.00606e+006 ; # 1509
PLANT.PIPEC2.PSI(4) := 2.08199e+006 ; # 1510
PLANT.PIPEC2.PSI(5) := 2.14854e+006 ; # 1511
PLANT.PIPEC2.PSI(6) := 2.20709e+006 ; # 1512
PLANT.PIPEC2.PSI(7) := 2.25877e+006 ; # 1513
PLANT.PIPEC2.PSI(8) := 2.30451e+006 ; # 1514
PLANT.PIPEC2.PSI(9) := 2.34506e+006 ; # 1515
PLANT.PIPEC2.PSI(10) := 2.38106e+006 ; # 1516
PLANT.PIPEC2.PSI(11) := 2.41306e+006 ; # 1517
PLANT.PIPEC2.PSI(12) := 2.44149e+006 ; # 1518
PLANT.PIPEC2.PSI(13) := 2.46677e+006 ; # 1519
PLANT.PIPEC2.PSI(14) := 2.48921e+006 ; # 1520
PLANT.PIPEC2.PSI(15) := 2.50911e+006 ; # 1521
PLANT.PIPEC2.PSI(16) := 2.52672e+006 ; # 1522
PLANT.PIPEC2.PSI(17) := 2.54225e+006 ; # 1523
PLANT.PIPEC2.PSI(18) := 2.55591e+006 ; # 1524
PLANT.PIPEC2.PSI(19) := 2.56786e+006 ; # 1525
PLANT.PIPEC2.PSI(20) := 2.57824e+006 ; # 1526
PLANT.PIPEC2.PSI(21) := 2.5872e+006 ; # 1527
PLANT.PIPEC2.PSI(22) := 2.59485e+006 ; # 1528
PLANT.PIPEC2.PSI(23) := 2.60129e+006 ; # 1529
PLANT.PIPEC2.PSI(24) := 2.60662e+006 ; # 1530
PLANT.PIPEC2.PSI(25) := 2.61094e+006 ; # 1531
PLANT.PIPEC2.PSI(26) := 2.6143e+006 ; # 1532
PLANT.PIPEC2.PSI(27) := 2.61679e+006 ; # 1533
PLANT.PIPEC2.PSI(28) := 2.61846e+006 ; # 1534
PLANT.PIPEC2.PSI(29) := 2.61938e+006 ; # 1535
PLANT.PIPEC2.PSI(30) := 2.6196e+006 ; # 1536
PLANT.PIPEC2.PSI(31) := 2.61915e+006 ; # 1537
PLANT.PIPEC2.PSI(32) := 2.6181e+006 ; # 1538
PLANT.PIPEC2.PSI(33) := 2.61647e+006 ; # 1539
PLANT.PIPEC2.PSI(34) := 2.6143e+006 ; # 1540
PLANT.PIPEC2.PSI(35) := 2.61163e+006 ; # 1541
PLANT.PIPEC2.PSI(36) := 2.60848e+006 ; # 1542
PLANT.PIPEC2.PSI(37) := 2.60489e+006 ; # 1543
PLANT.PIPEC2.PSI(38) := 2.60088e+006 ; # 1544
PLANT.PIPEC2.PSI(39) := 2.59648e+006 ; # 1545
PLANT.PIPEC2.PSI(40) := 2.59171e+006 ; # 1546
PLANT.PIPEC2.HT.RHOPIPEHE := 2.39784 ; # 1547
PLANT.PIPEC2.HT.VPIPE := 68.6177 ; # 1548
PLANT.PIPEC2.HT.U := 1880.53 ; # 1549
```

SIMULATIONS_CH3_LOOPONLY_FULLDYN.JAC

```
INCLUDE MODELS_PHYSICALPROPERTIES
INCLUDE MODELS_HEATEXCHANGERS
INCLUDE MODELS_PIPEMODELS
INCLUDE MODELS_HEATTRANSFERLOOP
#####################################################################
SIMULATION HTLFullDyn_SS
#####################################################################
  OPTIONS
    INIT_PRINT_LEVEL:=0;
    DYNAMIC_PRINT_LEVEL:=1;
    DYNAMIC_REPORTING_INTERVAL:=10;
    CSVOUTPUT:=TRUE;
    DYNAMIC_BOUNDS := TRUE;
    INIT_BLOCK_SOLVE:= TRUE;
    INIT_RELATIVE_TOLERANCE := 1e-9;
    INIT_ABSOLUTE_TOLERANCE := 1e-8;
    REINIT_RELATIVE_TOLERANCE:= 1e-7;
```

```
18      REINIT_ABSOLUTE_TOLERANCE:= 1e-7;
19      DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
20      DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
21      REINIT_MAX_ITERATIONS:= 1000;
22      REINIT_PRINT_LEVEL:= 0;
23      CHECK_MATH_EXCEPTIONS:= FALSE;
24      DYNAMIC_SCALING:= TRUE;
25      INITIALIZE_SENSITIVITIES := TRUE;
26    UNIT
27      Plant AS HeatTransferLoopFullDynPipesFullDyn
28    SET
29      #Plant.Loop1
30      Plant.Loop1.InitDeltaP:=500.0;
31      Plant.Loop1.MaxDeltaP:= 304125;
32      # Plant.PipeH
33      Plant.PipeH.InitP:=6.95e6;
34      Plant.PipeH.InitT:= 273.15 + 530;
35      Plant.PipeC2.InitP:=1.00e6;
36      Plant.InitT:= 273.15+18;
37    INTERMEDIATE
38    INPUT
39      #Loop1
40      Plant.Loop1.DeltaP:=  304125;
41      Plant.Loop1.AdaptFactor1:=1;
42      Plant.Loop1.AdaptFactor2:=1;
43      Plant.Loop1.ModifyFlow := 0.00;
44      Plant.Loop1.comp_leak := 0.0;
45      # PipeH
46      WITHIN Plant DO
47        PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
48        PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
49        PipeH.P_Out:=6950000.0;#PipeH.InitP;
50      END # within
51      # PipeC2
52      WITHIN Plant DO
53        PipeC2.T0:= 614.15;#InitT;
54        PipeC2.P_In:=1110000.0; #PipeC2.InitP+100;
55        PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
56      END # within
57    PRESET
58      INCLUDE  ..\input\HTLFULLDYN_SS_IN_PRE
59    INITIAL
60      INCLUDE  ..\input\HTLFULLDYN_SS_IN_INI
61    SCHEDULE
62      SEQUENCE
63        CONTINUE FOR 100
64        RESET
65          Plant.PipeC2.P_In:= 1110000.0 + 10950.0*(TIME-OLD(TIME))/10;
66          Plant.Loop1.DeltaP:= 304125 + (331300 - 304125)*(TIME - OLD(TIME))/10;
67        END # reset
68        CONTINUE FOR 10
69        RESET
70          Plant.PipeC2.P_In:=1110000.0 + 10950.0;
71          Plant.Loop1.DeltaP:= 331300;
72        END # reset
73        CONTINUE FOR 1000
74        RESET
75          Plant.Loop1.ModifyFlow:= -0.01;
76        END # reset
77        CONTINUE FOR 38.55
78        RESET
79          Plant.Loop1.ModifyFlow:= 0.0;
80        END # reset
81        CONTINUE FOR 1000
82        DISPLAY  TIME, "seg" END
83        SAVE PRESETS HTLFullDyn_SS_Out_PRE
84        SAVE INITIAL HTLFullDyn_SS_Out_INI
85      END # sequence
86  END # simulation
87  ######################################################################
88
89
90  ######################################################################
91  SIMULATION HTLFullDyn_QStepDecrFPHX_Event30s
92  ######################################################################
93    OPTIONS
94      INIT_PRINT_LEVEL:=0;
95      DYNAMIC_PRINT_LEVEL:=0;
96      DYNAMIC_REPORTING_INTERVAL:=0.1;
97      CSVOUTPUT:=TRUE;
98      DYNAMIC_BOUNDS := TRUE;
99      INIT_BLOCK_SOLVE:= TRUE;
100     INIT_RELATIVE_TOLERANCE := 1e-9;
101     INIT_ABSOLUTE_TOLERANCE := 1e-8;
102     REINIT_RELATIVE_TOLERANCE:= 1e-7;
103     REINIT_ABSOLUTE_TOLERANCE:= 1e-7;
104     DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
105     DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
106     REINIT_MAX_ITERATIONS:= 1000;
107     REINIT_PRINT_LEVEL:= 0;
108     CHECK_MATH_EXCEPTIONS:= FALSE;
109     DYNAMIC_SCALING:= TRUE;
```

```
110       INITIALIZE_SENSITIVITIES := TRUE;
111     UNIT
112       Plant AS HeatTransferLoopFullDynPipesFullDyn
113     SET
114       #Plant.Loop1
115       Plant.Loop1.InitDeltaP:=500.0;
116       Plant.Loop1.MaxDeltaP:= 304125;
117       # Plant.PipeH
118       Plant.PipeH.InitP:=6.95e6;
119       Plant.PipeH.InitT:= 273.15 + 530;
120       Plant.PipeC2.InitP:=1.00e6;
121       Plant.InitT:= 273.15+18;
122     INTERMEDIATE
123     INPUT
124       #Loop1
125       Plant.Loop1.DeltaP:=  331300;
126       Plant.Loop1.AdaptFactor1:=1;
127       Plant.Loop1.AdaptFactor2:=1;
128       Plant.Loop1.ModifyFlow := 0.00;
129       Plant.Loop1.comp_leak := 0.0;
130       # PipeH
131       WITHIN Plant DO
132         PipeH.T0:=1173.15;  #PipeH.InitT;
133         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
134         PipeH.P_Out:=6950000.0;#PipeH.InitP;
135       END # within
136       # PipeC2
137       WITHIN Plant DO
138         PipeC2.T0:= 614.15;#InitT;
139         PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
140         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
141       END # within
142     PRESET
143       INCLUDE  ..\HTLFullDyn_SS_sout\HTLFullDyn_SS_Out_PRE
144     INITIAL
145       INCLUDE  ..\HTLFullDyn_SS_sout\HTLFullDyn_SS_Out_INI
146     SCHEDULE
147       SEQUENCE
148         CONTINUE FOR 10
149         DISPLAY  Plant.PipeC2.F(1) END
150         RESET
151           Plant.PipeC2.P_In:= 1120950.0 - (1120950.0 - 1000000.0 )*0.658*(TIME - OLD(TIME))/0.2;
152         END # reset
153         CONTINUE FOR 0.2
154         RESET
155           Plant.PipeC2.P_In:= 1120950.0 - (1120950.0 - 1000000.0 )*0.658;
156         END # reset
157         CONTINUE FOR 19.8 #20
158         DISPLAY  TIME, "seg" END
159         DISPLAY  Plant.PipeC2.F(1) END
160         SAVE PRESETS HTLFullDyn_QStepDecrFPHX_Event30s_Out_PRE
161         SAVE INITIAL HTLFullDyn_QStepDecrFPHX_Event30s_Out_INI
162       END # sequence
163   END # simulation
164   ######################################################################
165
166
167   ######################################################################
168   SIMULATION HTLFullDyn_QStepDecrFPHX_PostEvent1000s
169   ######################################################################
170     OPTIONS
171       INIT_PRINT_LEVEL:=0;
172       DYNAMIC_PRINT_LEVEL:=0;
173       DYNAMIC_REPORTING_INTERVAL:=1.0;
174       CSVOUTPUT:=TRUE;
175       DYNAMIC_BOUNDS := TRUE;
176       INIT_BLOCK_SOLVE:= TRUE;
177       INIT_RELATIVE_TOLERANCE := 1e-9;
178       INIT_ABSOLUTE_TOLERANCE := 1e-8;
179       REINIT_RELATIVE_TOLERANCE:= 1e-9;
180       REINIT_ABSOLUTE_TOLERANCE:= 1e-9;
181       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
182       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
183       REINIT_MAX_ITERATIONS:= 1000;
184       REINIT_PRINT_LEVEL:= 0;
185       CHECK_MATH_EXCEPTIONS:= FALSE;
186       DYNAMIC_SCALING:= TRUE;
187       INITIALIZE_SENSITIVITIES := TRUE;
188     UNIT
189       Plant AS HeatTransferLoopFullDynPipesFullDyn
190     SET
191       #Plant.Loop1
192       Plant.Loop1.InitDeltaP:=500.0;
193       Plant.Loop1.MaxDeltaP:= 304125;
194       # Plant.PipeH
195       Plant.PipeH.InitP:=6.95e6;
196       Plant.PipeH.InitT:= 273.15 + 530;
197       Plant.PipeC2.InitP:=1.00e6;
198       Plant.InitT:= 273.15+18;
199     INTERMEDIATE
200     INPUT
201       #Loop1
```

```
202       Plant.Loop1.DeltaP:=  331300;
203       Plant.Loop1.AdaptFactor1:=1;
204       Plant.Loop1.AdaptFactor2:=1;
205       Plant.Loop1.ModifyFlow := 0.00;
206       Plant.Loop1.comp_leak := 0.0;
207       # PipeH
208       WITHIN Plant DO
209         PipeH.T0:=1173.1500;  #PipeH.InitT;
210         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
211         PipeH.P_Out:=6950000.0;#PipeH.InitP;
212       END # within
213       # PipeC2
214       WITHIN Plant DO
215         PipeC2.T0:= 614.15;#InitT;
216         PipeC2.P_In:=1120950.0 - (1120950.0 - 1000000.0 )*0.658;     # 1110000.0; #PipeC2.InitP+100;
217         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
218       END # within
219     PRESET
220       INCLUDE  ..\HTLFullDyn_QStepDecrFPHX_Event30s_sout\HTLFullDyn_QStepDecrFPHX_Event30s_Out_PRE
221     INITIAL
222       INCLUDE  ..\HTLFullDyn_QStepDecrFPHX_Event30s_sout\HTLFullDyn_QStepDecrFPHX_Event30s_Out_INI
223     SCHEDULE
224       SEQUENCE
225         CONTINUE FOR 1000
226         DISPLAY  TIME, "seg" END
227         DISPLAY  Plant.PipeC2.F(1) END
228         SAVE PRESETS HTLFullDyn_QStepDecrFPHX_PostEvent1000s_Out_PRE
229         SAVE INITIAL HTLFullDyn_QStepDecrFPHX_PostEvent1000s_Out_INI
230       END # sequence
231   END # simulation
232   ######################################################################
233
234
235   ######################################################################
236   SIMULATION HTLFullDyn_QStepIncDPLoopComp_Event160sToTime
237   ######################################################################
238     OPTIONS
239       INIT_PRINT_LEVEL:=0;
240       DYNAMIC_PRINT_LEVEL:=0;
241       DYNAMIC_REPORTING_INTERVAL:=0.01;
242       CSVOUTPUT:=TRUE;
243       DYNAMIC_BOUNDS := TRUE;
244       INIT_BLOCK_SOLVE:= TRUE;
245       INIT_RELATIVE_TOLERANCE := 1e-9;
246       INIT_ABSOLUTE_TOLERANCE := 1e-8;
247       REINIT_RELATIVE_TOLERANCE:= 1e-7;
248       REINIT_ABSOLUTE_TOLERANCE:= 1e-7;
249       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
250       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
251       REINIT_MAX_ITERATIONS:= 1000;
252       REINIT_PRINT_LEVEL:= 0;
253       CHECK_MATH_EXCEPTIONS:= FALSE;
254       DYNAMIC_SCALING:= TRUE;
255       INITIALIZE_SENSITIVITIES := TRUE;
256     UNIT
257       Plant AS HeatTransferLoopFullDynPipesFullDyn
258     SET
259       #Plant.Loop1
260       Plant.Loop1.InitDeltaP:=500.0;
261       Plant.Loop1.MaxDeltaP:= 304125;
262       # Plant.PipeH
263       Plant.PipeH.InitP:=6.95e6;
264       Plant.PipeH.InitT:= 273.15 + 530;
265       Plant.PipeC2.InitP:=1.00e6;
266       Plant.InitT:= 273.15+18;
267     INTERMEDIATE
268     INPUT
269       #Loop1
270       Plant.Loop1.DeltaP:=  331300;
271       Plant.Loop1.AdaptFactor1:=1;
272       Plant.Loop1.AdaptFactor2:=1;
273       Plant.Loop1.ModifyFlow := 0.00;
274       Plant.Loop1.comp_leak := 0.0;
275       # PipeH
276       WITHIN Plant DO
277         PipeH.T0:=1173.15;  #PipeH.InitT;
278         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
279         PipeH.P_Out:=6950000.0;#PipeH.InitP;
280       END # within
281       # PipeC2
282       WITHIN Plant DO
283         PipeC2.T0:= 614.15;#InitT;
284         PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
285         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
286       END # within
287     PRESET
288       INCLUDE  ..\HTLFullDyn_SS_sout\HTLFullDyn_SS_Out_PRE
289     INITIAL
290       INCLUDE  ..\HTLFullDyn_SS_sout\HTLFullDyn_SS_Out_INI
291     SCHEDULE
292       SEQUENCE
293         CONTINUE FOR 10
```

```
294         RESET
295           Plant.Loop1.DeltaP:= 331300 + 47515*(TIME-10)/0.2;
296         END # reset
297         CONTINUE FOR 0.2
298         RESET
299           Plant.Loop1.DeltaP:= 331300 + 47515;
300         END # reset
301         CONTINUE FOR 149.8 #20
302         DISPLAY  TIME, "seg" END
303         DISPLAY  Plant.PipeC2.F(1) END
304         SAVE PRESETS HTLFullDyn_QStepDecrFPHX_Event160s_Out_PRE
305         SAVE INITIAL HTLFullDyn_QStepDecrFPHX_Event160s_Out_INI
306       END # sequence
307   END # simulation
308   ######################################################################
309
310
311   ######################################################################
312   SIMULATION HTLFullDyn_QStepIncDPLoopComp_Event30s
313   ######################################################################
314     OPTIONS
315       INIT_PRINT_LEVEL:=0;
316       DYNAMIC_PRINT_LEVEL:=0;
317       DYNAMIC_REPORTING_INTERVAL:=0.005;
318       CSVOUTPUT:=TRUE;
319       DYNAMIC_BOUNDS := TRUE;
320       INIT_BLOCK_SOLVE:= TRUE;
321       INIT_RELATIVE_TOLERANCE := 1e-9;
322       INIT_ABSOLUTE_TOLERANCE := 1e-8;
323       REINIT_RELATIVE_TOLERANCE:= 1e-7;
324       REINIT_ABSOLUTE_TOLERANCE:= 1e-7;
325       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
326       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
327       REINIT_MAX_ITERATIONS:= 1000;
328       REINIT_PRINT_LEVEL:= 0;
329       CHECK_MATH_EXCEPTIONS:= FALSE;
330       DYNAMIC_SCALING:= TRUE;
331       INITIALIZE_SENSITIVITIES := TRUE;
332     UNIT
333       Plant AS HeatTransferLoopFullDynPipesFullDyn
334     SET
335       #Plant.Loop1
336       Plant.Loop1.InitDeltaP:=500.0;
337       Plant.Loop1.MaxDeltaP:= 304125;
338       # Plant.PipeH
339       Plant.PipeH.InitP:=6.95e6;
340       Plant.PipeH.InitT:= 273.15 + 530;
341       Plant.PipeC2.InitP:=1.00e6;
342       Plant.InitT:= 273.15+18;
343     INTERMEDIATE
344     INPUT
345       #Loop1
346       Plant.Loop1.DeltaP:=  331300;
347       Plant.Loop1.AdaptFactor1:=1;
348       Plant.Loop1.AdaptFactor2:=1;
349       Plant.Loop1.ModifyFlow := 0.00;
350       Plant.Loop1.comp_leak := 0.0;
351       # PipeH
352       WITHIN Plant DO
353         PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
354         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
355         PipeH.P_Out:=6950000.0;#PipeH.InitP;
356       END # within
357       # PipeC2
358       WITHIN Plant DO
359         PipeC2.T0:= 614.15;#InitT;
360         PipeC2.P_In:=1110000.0 + 10950.0; # 1110000.0; #PipeC2.InitP+100;
361         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
362       END # within
363     PRESET
364       INCLUDE  ..\HTLFullDyn_SS_sout\HTLFullDyn_SS_Out_PRE
365     INITIAL
366       INCLUDE  ..\HTLFullDyn_SS_sout\HTLFullDyn_SS_Out_INI
367     SCHEDULE
368       SEQUENCE
369         CONTINUE FOR 10
370         RESET
371           Plant.Loop1.DeltaP:= 331300 + 47515*(TIME-10)/0.2;
372         END # reset
373         CONTINUE FOR 0.2
374         RESET
375           Plant.Loop1.DeltaP:= 331300 + 47515;
376         END # reset
377         CONTINUE FOR 19.8# 20
378         DISPLAY  TIME, "seg" END
379         DISPLAY Plant.Loop1.F(1) END
380         SAVE PRESETS HTLFullDyn_QStepIncDPLoopComp_Event30s_Out_PRE
381         SAVE INITIAL HTLFullDyn_QStepIncDPLoopComp_Event30s_Out_INI
382       END # sequence
383   END # simulation
384   ######################################################################
385
```

```
386
387 ######################################################################
388 SIMULATION HTLFullDyn_QStepIncDPLoopComp_PostEvent1000s
389 ######################################################################
390   OPTIONS
391     INIT_PRINT_LEVEL:=0;
392     DYNAMIC_PRINT_LEVEL:=0;
393     DYNAMIC_REPORTING_INTERVAL:=1.0;
394     CSVOUTPUT:=TRUE;
395     DYNAMIC_BOUNDS := TRUE;
396     INIT_BLOCK_SOLVE:= TRUE;
397     INIT_RELATIVE_TOLERANCE := 1e-9;
398     INIT_ABSOLUTE_TOLERANCE := 1e-8;
399     REINIT_RELATIVE_TOLERANCE:= 1e-7;
400     REINIT_ABSOLUTE_TOLERANCE:= 1e-7;
401     DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
402     DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
403     REINIT_MAX_ITERATIONS:= 1000;
404     REINIT_PRINT_LEVEL:= 0;
405     CHECK_MATH_EXCEPTIONS:= FALSE;
406     DYNAMIC_SCALING:= TRUE;
407     INITIALIZE_SENSITIVITIES := TRUE;
408   UNIT
409     Plant AS HeatTransferLoopFullDynPipesFullDyn
410   SET
411     #Plant.Loop1
412     Plant.Loop1.InitDeltaP:=500.0;
413     Plant.Loop1.MaxDeltaP:= 304125;
414     # Plant.PipeH
415     Plant.PipeH.InitP:=6.95e6;
416     Plant.PipeH.InitT:= 273.15 + 530;
417     Plant.PipeC2.InitP:=1.00e6;
418     Plant.InitT:= 273.15+18;
419   INTERMEDIATE
420   INPUT
421     #Loop1
422     Plant.Loop1.DeltaP:=  331300 + 47515;
423     Plant.Loop1.AdaptFactor1:=1;
424     Plant.Loop1.AdaptFactor2:=1;
425     Plant.Loop1.ModifyFlow := 0.00;
426     Plant.Loop1.comp_leak := 0.0;
427     # PipeH
428     WITHIN Plant DO
429       PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
430       PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
431       PipeH.P_Out:=6950000.0;#PipeH.InitP;
432     END # within
433     # PipeC2
434     WITHIN Plant DO
435       PipeC2.T0:= 614.15;#InitT;
436       PipeC2.P_In:=1110000.0 + 10950.0; #PipeC2.InitP+100;
437       PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
438     END # within
439   PRESET
440     INCLUDE  ..\HTLFullDyn_QStepIncDPLoopComp_Event30s_sout\HTLFullDyn_QStepIncDPLoopComp_Event30s_Out_PRE
441   INITIAL
442     INCLUDE  ..\HTLFullDyn_QStepIncDPLoopComp_Event30s_sout\HTLFullDyn_QStepIncDPLoopComp_Event30s_Out_INI
443   SCHEDULE
444     SEQUENCE
445       CONTINUE FOR 1000
446       DISPLAY  TIME, "seg" END
447       DISPLAY Plant.Loop1.F(1) END
448       SAVE PRESETS HTLFullDyn_StepIncDPLoopComp_QPostEvent1000s_Out_PRE
449       SAVE INITIAL HTLFullDyn_StepIncDPLoopComp_QPostEvent1000s_Out_INI
450     END # sequence
451 END # simulation
452 ######################################################################
```

SIMULATIONS_CH3_LOOPONLY_0MACH.JAC

```
 1 INCLUDE MODELS_PHYSICALPROPERTIES
 2 INCLUDE MODELS_HEATEXCHANGERS
 3 INCLUDE MODELS_PIPEMODELS
 4 INCLUDE MODELS_HEATTRANSFERLOOP
 5 ######################################################################
 6 SIMULATION HTL0Mach_SS
 7 ######################################################################
 8   OPTIONS
 9     INIT_PRINT_LEVEL:=0;
10     DYNAMIC_PRINT_LEVEL:=1;
11     DYNAMIC_REPORTING_INTERVAL:=10;
12     CSVOUTPUT:=TRUE;
13     DYNAMIC_BOUNDS := TRUE;
```

```
14      INIT_BLOCK_SOLVE:= TRUE;
15      INIT_RELATIVE_TOLERANCE := 1e-9;
16      INIT_ABSOLUTE_TOLERANCE := 1e-8;
17      REINIT_RELATIVE_TOLERANCE:= 1e-8;
18      REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
19      DYNAMIC_RELATIVE_TOLERANCE:= 1e-8;
20      DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
21      REINIT_MAX_ITERATIONS:= 2000;
22      REINIT_PRINT_LEVEL:= 0;
23      CHECK_MATH_EXCEPTIONS:= FALSE;
24      DYNAMIC_SCALING:= TRUE;
25      INITIALIZE_SENSITIVITIES := TRUE;
26    UNIT
27      Plant AS HeatTransferLoop0MachPipesFullDyn
28    SET
29      #Plant.Loop1
30      Plant.Loop1.InitDeltaP:=500.0;
31      Plant.Loop1.MaxDeltaP:= 304125;
32      # Plant.PipeH
33      Plant.PipeH.InitP:=6.95e6;
34      Plant.PipeH.InitT:= 273.15 + 530;
35      Plant.PipeC2.InitP:=1.00e6;
36      Plant.InitT:= 273.15+18;
37    INTERMEDIATE
38    INPUT
39      #Loop1
40      Plant.Loop1.DeltaP:= 304125;
41      Plant.Loop1.AdaptFactor1:=1;
42      Plant.Loop1.AdaptFactor2:=1;
43      Plant.Loop1.ModifyFlow := 0.00;
44      Plant.Loop1.comp_leak := 0.0;
45      # PipeH
46      WITHIN Plant DO
47        PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
48        PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
49        PipeH.P_Out:=6950000.0;#PipeH.InitP;
50      END # within
51      # PipeC2
52      WITHIN Plant DO
53        PipeC2.T0:= 614.15;#InitT;
54        PipeC2.P_In:=1110000.0; #PipeC2.InitP+100;
55        PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
56      END # within
57    PRESET
58      INCLUDE  ..\input\HTL0MACH_SS_IN_PRE
59    INITIAL
60      INCLUDE  ..\input\HTL0MACH_SS_IN_INI
61    SCHEDULE
62      SEQUENCE
63        CONTINUE FOR 100
64        RESET
65          Plant.PipeC2.P_In:= 1110000.0 + 10950.0*(TIME-OLD(TIME))/10;
66          Plant.Loop1.DeltaP:= 304125 + (331300 - 304125)*(TIME - OLD(TIME))/10;
67        END # reset
68        CONTINUE FOR 10
69        RESET
70          Plant.PipeC2.P_In:=1110000.0 + 10950.0;
71          Plant.Loop1.DeltaP:= 331300;
72        END # reset
73        CONTINUE FOR 1000
74        RESET
75          Plant.Loop1.ModifyFlow:= -0.01;
76        END # reset
77        CONTINUE FOR 38.55
78        RESET
79          Plant.Loop1.ModifyFlow:= 0.0;
80        END # reset
81        CONTINUE FOR 1000
82        DISPLAY  TIME, "seg" END
83        SAVE PRESETS HTL0MACH_SS_OUT_PRE
84        SAVE INITIAL HTL0MACH_SS_OUT_INI  END # sequence
85  END # simulation
86  ######################################################################
87
88
89  ######################################################################
90  SIMULATION HTL0Mach_QStepDecrFPHX_Event30s
91  ######################################################################
92    OPTIONS
93      INIT_PRINT_LEVEL:=0;
94      DYNAMIC_PRINT_LEVEL:=0;
95      DYNAMIC_REPORTING_INTERVAL:=0.1;
96      CSVOUTPUT:=TRUE;
97      DYNAMIC_BOUNDS := TRUE;
98      INIT_BLOCK_SOLVE:= TRUE;
99      INIT_RELATIVE_TOLERANCE := 1e-9;
100     INIT_ABSOLUTE_TOLERANCE := 1e-8;
101     REINIT_RELATIVE_TOLERANCE:= 1e-8;
102     REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
103     DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
104     DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
105     REINIT_MAX_ITERATIONS:= 2000;
```

```
106       REINIT_PRINT_LEVEL:= 0;
107       CHECK_MATH_EXCEPTIONS:= FALSE;
108       DYNAMIC_SCALING:= TRUE;
109       INITIALIZE_SENSITIVITIES := TRUE;
110     UNIT
111       Plant AS HeatTransferLoop0MachPipesFullDyn
112     SET
113       #Plant.Loop1
114       Plant.Loop1.InitDeltaP:=500.0;
115       Plant.Loop1.MaxDeltaP:= 304125;
116       # Plant.PipeH
117       Plant.PipeH.InitP:=6.95e6;
118       Plant.PipeH.InitT:= 273.15 + 530;
119       Plant.PipeC2.InitP:=1.00e6;
120       Plant.InitT:= 273.15+18;
121     INTERMEDIATE
122     INPUT
123       #Loop1
124       Plant.Loop1.DeltaP:= 331300;
125       Plant.Loop1.AdaptFactor1:=1;
126       Plant.Loop1.AdaptFactor2:=1;
127       Plant.Loop1.ModifyFlow := 0.00;
128       Plant.Loop1.comp_leak := 0.0;
129       # PipeH
130       WITHIN Plant DO
131         PipeH.T0:=1173.1500;  #PipeH.InitT;
132         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
133         PipeH.P_Out:=6950000.0;#PipeH.InitP;
134       END # within
135       # PipeC2
136       WITHIN Plant DO
137         PipeC2.T0:= 614.15;#InitT;
138         PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
139         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
140       END # within
141     PRESET
142       INCLUDE  ..\HTL0Mach_SS_sout\HTL0Mach_SS_Out_PRE
143     INITIAL
144       INCLUDE  ..\HTL0Mach_SS_sout\HTL0Mach_SS_Out_INI
145     SCHEDULE
146       SEQUENCE
147         CONTINUE FOR 10
148         DISPLAY  Plant.PipeC2.F(1) END
149         RESET
150           Plant.PipeC2.P_In:= 1120950.0 - (1120950.0 - 1000000.0 )*0.658*(TIME - OLD(TIME))/0.2;
151         END # reset
152         CONTINUE FOR 0.2
153         RESET
154           Plant.PipeC2.P_In:= 1120950.0 - (1120950.0 - 1000000.0 )*0.658;
155         END # reset
156         CONTINUE FOR 19.8 #20
157         DISPLAY  TIME, "seg" END
158         DISPLAY  Plant.PipeC2.F(1) END
159         SAVE PRESETS HTL0Mach_QStepDecrFPHX_Event30s_Out_PRE
160         SAVE INITIAL HTL0Mach_QStepDecrFPHX_Event30s_Out_INI
161       END # sequence
162   END # simulation
163   ######################################################################
164
165
166   ######################################################################
167   SIMULATION HTL0Mach_QStepDecrFPHX_PostEvent1000s
168   ######################################################################
169     OPTIONS
170       INIT_PRINT_LEVEL:=0;
171       DYNAMIC_PRINT_LEVEL:=0;
172       DYNAMIC_REPORTING_INTERVAL:=1.0;
173       CSVOUTPUT:=TRUE;
174       DYNAMIC_BOUNDS := TRUE;
175       INIT_BLOCK_SOLVE:= TRUE;
176       INIT_RELATIVE_TOLERANCE := 1e-9;
177       INIT_ABSOLUTE_TOLERANCE := 1e-8;
178       REINIT_RELATIVE_TOLERANCE:= 1e-8;
179       REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
180       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
181       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
182       REINIT_MAX_ITERATIONS:= 2000;
183       REINIT_PRINT_LEVEL:= 0;
184       CHECK_MATH_EXCEPTIONS:= FALSE;
185       DYNAMIC_SCALING:= TRUE;
186       INITIALIZE_SENSITIVITIES := TRUE;
187     UNIT
188       Plant AS HeatTransferLoop0MachPipesFullDyn
189     SET
190       #Plant.Loop1
191       Plant.Loop1.InitDeltaP:=500.0;
192       Plant.Loop1.MaxDeltaP:= 304125;
193       # Plant.PipeH
194       Plant.PipeH.InitP:=6.95e6;
195       Plant.PipeH.InitT:= 273.15 + 530;
196       Plant.PipeC2.InitP:=1.00e6;
197       Plant.InitT:= 273.15+18;
```

```
198     INTERMEDIATE
199     INPUT
200       #Loop1
201       Plant.Loop1.DeltaP:= 331300;
202       Plant.Loop1.AdaptFactor1:=1;
203       Plant.Loop1.AdaptFactor2:=1;
204       Plant.Loop1.ModifyFlow := 0.00;
205       Plant.Loop1.comp_leak := 0.0;
206       # PipeH
207       WITHIN Plant DO
208         PipeH.T0:=1173.1500;  #PipeH.InitT;
209         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
210         PipeH.P_Out:=6950000.0;#PipeH.InitP;
211       END # within
212       # PipeC2
213       WITHIN Plant DO
214         PipeC2.T0:= 614.15;#InitT;
215         PipeC2.P_In:=1120950.0 - (1120950.0 - 1000000.0 )*0.658;     # 1110000.0; #PipeC2.InitP+100;
216         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
217       END # within
218     PRESET
219       INCLUDE  ..\HTL0Mach_QStepDecrFPHX_Event30s_sout\HTL0Mach_QStepDecrFPHX_Event30s_Out_PRE
220     INITIAL
221       INCLUDE  ..\HTL0Mach_QStepDecrFPHX_Event30s_sout\HTL0Mach_QStepDecrFPHX_Event30s_Out_INI
222     SCHEDULE
223       SEQUENCE
224         CONTINUE FOR 1000
225         DISPLAY  TIME, "seg" END
226         SAVE PRESETS HTL0Mach_QStepDecrFPHX_PostEvent1000s_Out_PRE
227         SAVE INITIAL HTL0Mach_QStepDecrFPHX_PostEvent1000s_Out_INI
228       END # sequence
229   END # simulation
230   ######################################################################
231
232
233   ######################################################################
234   SIMULATION HTL0Mach_QStepIncDPLoopComp_Event160sToTime
235   ######################################################################
236     OPTIONS
237       INIT_PRINT_LEVEL:=0;
238       DYNAMIC_PRINT_LEVEL:=0;
239       DYNAMIC_REPORTING_INTERVAL:=0.01;
240       CSVOUTPUT:=TRUE;
241       DYNAMIC_BOUNDS := TRUE;
242       INIT_BLOCK_SOLVE:= TRUE;
243       INIT_RELATIVE_TOLERANCE := 1e-9;
244       INIT_ABSOLUTE_TOLERANCE := 1e-8;
245       REINIT_RELATIVE_TOLERANCE:= 1e-8;
246       REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
247       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
248       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
249       REINIT_MAX_ITERATIONS:= 2000;
250       REINIT_PRINT_LEVEL:= 0;
251       CHECK_MATH_EXCEPTIONS:= FALSE;
252       DYNAMIC_SCALING:= TRUE;
253       INITIALIZE_SENSITIVITIES := TRUE;
254     UNIT
255       Plant AS HeatTransferLoop0MachPipesFullDyn
256     SET
257       #Plant.Loop1
258       Plant.Loop1.InitDeltaP:=500.0;
259       Plant.Loop1.MaxDeltaP:= 304125;
260       # Plant.PipeH
261       Plant.PipeH.InitP:=6.95e6;
262       Plant.PipeH.InitT:= 273.15 + 530;
263       Plant.PipeC2.InitP:=1.00e6;
264       Plant.InitT:= 273.15+18;
265     INTERMEDIATE
266     INPUT
267       #Loop1
268       Plant.Loop1.DeltaP:= 331300;
269       Plant.Loop1.AdaptFactor1:=1;
270       Plant.Loop1.AdaptFactor2:=1;
271       Plant.Loop1.ModifyFlow := 0.00;
272       Plant.Loop1.comp_leak := 0.0;
273       # PipeH
274       WITHIN Plant DO
275         PipeH.T0:=1173.1500;  #PipeH.InitT;
276         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
277         PipeH.P_Out:=6950000.0;#PipeH.InitP;
278       END # within
279       # PipeC2
280       WITHIN Plant DO
281         PipeC2.T0:= 614.15;#InitT;
282         PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
283         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
284       END # within
285     PRESET
286       INCLUDE  ..\HTL0Mach_SS_sout\HTL0MACH_SS_OUT_PRE
287     INITIAL
288       INCLUDE  ..\HTL0Mach_SS_sout\HTL0MACH_SS_OUT_INI
289     SCHEDULE
```

```
290       SEQUENCE
291         CONTINUE FOR 10
292         RESET
293           Plant.Loop1.DeltaP:= 331300 + 47515*(TIME-10)/0.2;
294         END # reset
295         CONTINUE FOR 0.2
296         RESET
297           Plant.Loop1.DeltaP:= 331300 + 47515;
298         END # reset
299         CONTINUE FOR 149.8 #20
300         DISPLAY  TIME, "seg" END
301         SAVE PRESETS HTL0Mach_QStepDecrFPHX_Event160s_Out_PRE
302         SAVE INITIAL HTL0Mach_QStepDecrFPHX_Event160s_Out_INI
303       END # sequence
304   END # simulation
305   ######################################################################
306
307
308
309
310
311
312
313
314
315
316   ######################################################################
317   SIMULATION HTL0Mach_QStepIncDPLoopComp_Event30s
318   ######################################################################
319     OPTIONS
320       INIT_PRINT_LEVEL:=0;
321       DYNAMIC_PRINT_LEVEL:=0;
322       DYNAMIC_REPORTING_INTERVAL:= 0.005; # 0.1;
323       CSVOUTPUT:=TRUE;
324       DYNAMIC_BOUNDS := TRUE;
325       INIT_BLOCK_SOLVE:= TRUE;
326       INIT_RELATIVE_TOLERANCE := 1e-9;
327       INIT_ABSOLUTE_TOLERANCE := 1e-8;
328       REINIT_RELATIVE_TOLERANCE:= 1e-8;
329       REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
330       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
331       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
332       REINIT_MAX_ITERATIONS:= 2000;
333       REINIT_PRINT_LEVEL:= 0;
334       CHECK_MATH_EXCEPTIONS:= FALSE;
335       DYNAMIC_SCALING:= TRUE;
336       INITIALIZE_SENSITIVITIES := TRUE;
337     UNIT
338       Plant AS HeatTransferLoop0MachPipesFullDyn
339     SET
340       #Plant.Loop1
341       Plant.Loop1.InitDeltaP:=500.0;
342       Plant.Loop1.MaxDeltaP:= 304125;
343       # Plant.PipeH
344       Plant.PipeH.InitP:=6.95e6;
345       Plant.PipeH.InitT:= 273.15 + 530;
346       Plant.PipeC2.InitP:=1.00e6;
347       Plant.InitT:= 273.15+18;
348     INTERMEDIATE
349     INPUT
350       #Loop1
351       Plant.Loop1.DeltaP:= 331300;
352       Plant.Loop1.AdaptFactor1:=1;
353       Plant.Loop1.AdaptFactor2:=1;
354       Plant.Loop1.ModifyFlow := 0.00;
355       Plant.Loop1.comp_leak := 0.0;
356       # PipeH
357       WITHIN Plant DO
358         PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
359         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
360         PipeH.P_Out:=6950000.0;#PipeH.InitP;
361       END # within
362       # PipeC2
363       WITHIN Plant DO
364         PipeC2.T0:= 614.15;#InitT;
365         PipeC2.P_In:=1110000.0 + 10950.0; # 1110000.0; #PipeC2.InitP+100;
366         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
367       END # within
368     PRESET
369       INCLUDE  ..\HTL0Mach_SS_sout\HTL0Mach_SS_Out_PRE
370     INITIAL
371       INCLUDE  ..\HTL0Mach_SS_sout\HTL0Mach_SS_Out_INI
372     SCHEDULE
373       SEQUENCE
374         CONTINUE FOR 10
375         RESET
376           Plant.Loop1.DeltaP:= 331300 + 47515*(TIME-10)/0.2;
377         END # reset
378         CONTINUE FOR 0.2
379         RESET
380           Plant.Loop1.DeltaP:= 331300 + 47515;
381         END # reset
```

```
382         CONTINUE FOR 19.8# 20
383         DISPLAY  TIME, "seg" END
384         DISPLAY Plant.Loop1.F(1) END
385         SAVE PRESETS HTL0Mach_QStepIncDPLoopComp_Event30s_Out_PRE
386         SAVE INITIAL HTL0Mach_QStepIncDPLoopComp_Event30s_Out_INI
387       END # sequence
388   END # simulation
389   ######################################################################
390
391
392   ######################################################################
393   SIMULATION HTL0Mach_QStepIncDPLoopComp_PostEvent1000s
394   ######################################################################
395     OPTIONS
396       INIT_PRINT_LEVEL:=0;
397       DYNAMIC_PRINT_LEVEL:=0;
398       DYNAMIC_REPORTING_INTERVAL:= 1.0; # 0.1;
399       CSVOUTPUT:=TRUE;
400       DYNAMIC_BOUNDS := TRUE;
401       INIT_BLOCK_SOLVE:= TRUE;
402       INIT_RELATIVE_TOLERANCE := 1e-9;
403       INIT_ABSOLUTE_TOLERANCE := 1e-8;
404       REINIT_RELATIVE_TOLERANCE:= 1e-8;
405       REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
406       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
407       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
408       REINIT_MAX_ITERATIONS:= 2000;
409       REINIT_PRINT_LEVEL:= 0;
410       CHECK_MATH_EXCEPTIONS:= FALSE;
411       DYNAMIC_SCALING:= TRUE;
412       INITIALIZE_SENSITIVITIES := TRUE;
413     UNIT
414       Plant AS HeatTransferLoop0MachPipesFullDyn
415     SET
416       #Plant.Loop1
417       Plant.Loop1.InitDeltaP:=500.0;
418       Plant.Loop1.MaxDeltaP:= 304125;
419       # Plant.PipeH
420       Plant.PipeH.InitP:=6.95e6;
421       Plant.PipeH.InitT:= 273.15 + 530;
422       Plant.PipeC2.InitP:=1.00e6;
423       Plant.InitT:= 273.15+18;
424     INTERMEDIATE
425     INPUT
426       #Loop1
427       Plant.Loop1.DeltaP:=  331300 + 47515;
428       Plant.Loop1.AdaptFactor1:=1;
429       Plant.Loop1.AdaptFactor2:=1;
430       Plant.Loop1.ModifyFlow := 0.00;
431       Plant.Loop1.comp_leak := 0.0;
432       # PipeH
433       WITHIN Plant DO
434         PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
435         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
436         PipeH.P_Out:=6950000.0;#PipeH.InitP;
437       END # within
438       # PipeC2
439       WITHIN Plant DO
440         PipeC2.T0:= 614.15;#InitT;
441         PipeC2.P_In:=1110000.0 + 10950.0; #PipeC2.InitP+100;
442         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
443       END # within
444     PRESET
445       INCLUDE  ..\HTL0Mach_QStepIncDPLoopComp_Event30s_sout\HTL0Mach_QStepIncDPLoopComp_Event30s_Out_PRE
446     INITIAL
447       INCLUDE  ..\HTL0Mach_QStepIncDPLoopComp_Event30s_sout\HTL0Mach_QStepIncDPLoopComp_Event30s_Out_INI
448     SCHEDULE
449       SEQUENCE
450         CONTINUE FOR 1000
451         DISPLAY  TIME, "seg" END
452         DISPLAY Plant.Loop1.F(1) END
453         SAVE PRESETS HTL0Mach_QStepIncDPLoopComp_PostEvent1000s_Out_PRE
454         SAVE INITIAL HTL0Mach_QStepIncDPLoopComp_PostEvent1000s_Out_PREs_Out_INI
455       END # sequence
456   END # simulation
457   ######################################################################
```

SIMULATIONS_CH3_LOOPONLY_QSS.JAC

```
1   INCLUDE MODELS_PHYSICALPROPERTIES
2   INCLUDE MODELS_HEATEXCHANGERS
3   INCLUDE MODELS_PIPEMODELS
4   INCLUDE MODELS_HEATTRANSFERLOOP
```

```
5   ######################################################################
6   SIMULATION HTLQSS_SS
7   ######################################################################
8     OPTIONS
9       INIT_PRINT_LEVEL:=0;
10      DYNAMIC_PRINT_LEVEL:=1;
11      DYNAMIC_REPORTING_INTERVAL:=10;
12      CSVOUTPUT:=TRUE;
13      DYNAMIC_BOUNDS := TRUE;
14      INIT_BLOCK_SOLVE:= TRUE;
15      INIT_RELATIVE_TOLERANCE := 1e-9;
16      INIT_ABSOLUTE_TOLERANCE := 1e-8;
17      REINIT_RELATIVE_TOLERANCE:= 1e-8;
18      REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
19      DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
20      DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
21      REINIT_MAX_ITERATIONS:= 2000;
22      REINIT_PRINT_LEVEL:= 0;
23      CHECK_MATH_EXCEPTIONS:= FALSE;
24      DYNAMIC_SCALING:= TRUE;
25      INITIALIZE_SENSITIVITIES := TRUE;
26    UNIT
27      Plant AS HeatTransferLoopQSSPipesFullDyn
28    SET
29      #Plant.Loop1
30      Plant.Loop1.InitDeltaP:=500.0;
31      Plant.Loop1.MaxDeltaP:=0.321e6;
32      # Plant.PipeH
33      Plant.PipeH.InitP:=6.95e6;
34      Plant.PipeH.InitT:= 273.15 + 530;
35      Plant.PipeC2.InitP:=1.00e6;
36      Plant.InitT:= 273.15+18;
37    INTERMEDIATE
38    INPUT
39      #Loop1
40      Plant.Loop1.DeltaP:=  304125;
41      Plant.Loop1.AdaptFactor1:=1;
42      Plant.Loop1.AdaptFactor2:=1;
43      Plant.Loop1.ModifyFlow := 0.00;
44      Plant.Loop1.rho_avg := 0.919725239276886;
45      # PipeH
46      WITHIN Plant DO
47        PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
48        PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
49        PipeH.P_Out:=6950000.0;#PipeH.InitP;
50      END # within
51      # PipeC2
52      WITHIN Plant DO
53        PipeC2.T0:= 614.15;#InitT;
54        PipeC2.P_In:=1110000.0; #PipeC2.InitP+100;
55        PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
56      END # within
57    PRESET
58      INCLUDE  ..\input\HTLQSS_SS_IN_PRE
59    INITIAL
60      INCLUDE  ..\input\HTLQSS_SS_IN_INI
61    SCHEDULE
62      SEQUENCE
63        CONTINUE FOR 100
64        RESET
65          Plant.PipeC2.P_In:= 1110000.0 + 10950.0*(TIME-OLD(TIME))/10;
66          Plant.Loop1.DeltaP:= 304125 + (331300 - 304125)*(TIME - OLD(TIME))/10;
67        END # reset
68        CONTINUE FOR 10
69        RESET
70          Plant.PipeC2.P_In:=1110000.0 + 10950.0;
71          Plant.Loop1.DeltaP:=331300;
72        END # reset
73        CONTINUE FOR 1000
74        RESET
75          Plant.Loop1.rho_avg:=  0.919725239276886 - 0.0140*0.919725239276886*(TIME-OLD(TIME))/10;
76        END # reset
77        CONTINUE FOR 10
78        RESET
79          Plant.Loop1.rho_avg:= (1.0 - 0.0140)*0.919725239276886;
80        END
81        CONTINUE FOR 1000
82        DISPLAY  TIME, "seg" END
83        SAVE PRESETS HTLQSS_SS_Out_PRE
84        SAVE INITIAL HTLQSS_SS_Out_INI
85      END # sequence
86  END # simulation
87  ######################################################################
88
89
90  ######################################################################
91  SIMULATION HTLQSS_QStepDecrFPHX_Event30s
92  ######################################################################
93    OPTIONS
94      INIT_PRINT_LEVEL:=0;
95      DYNAMIC_PRINT_LEVEL:=0;
96      DYNAMIC_REPORTING_INTERVAL:=0.01;
```

```
 97      CSVOUTPUT:=TRUE;
 98      DYNAMIC_BOUNDS := TRUE;
 99      INIT_BLOCK_SOLVE:= TRUE;
100      INIT_RELATIVE_TOLERANCE := 1e-9;
101      INIT_ABSOLUTE_TOLERANCE := 1e-8;
102      REINIT_RELATIVE_TOLERANCE:= 1e-8;
103      REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
104      DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
105      DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
106      REINIT_MAX_ITERATIONS:= 2000;
107      REINIT_PRINT_LEVEL:= 0;
108      CHECK_MATH_EXCEPTIONS:= FALSE;
109      DYNAMIC_SCALING:= TRUE;
110      INITIALIZE_SENSITIVITIES := TRUE;
111    UNIT
112      Plant AS HeatTransferLoopQSSPipesFullDyn
113    SET
114      #Plant.Loop1
115      Plant.Loop1.InitDeltaP:=500.0;
116      Plant.Loop1.MaxDeltaP:=0.321e6;
117      # Plant.PipeH
118      Plant.PipeH.InitP:=6.95e6;
119      Plant.PipeH.InitT:= 273.15 + 530;
120      Plant.PipeC2.InitP:=1.00e6;
121      Plant.InitT:= 273.15+18;
122    INTERMEDIATE
123    INPUT
124      #Loop1
125      Plant.Loop1.DeltaP:=  331300;
126      Plant.Loop1.AdaptFactor1:=1;
127      Plant.Loop1.AdaptFactor2:=1;
128      Plant.Loop1.ModifyFlow := 0.00;
129      Plant.Loop1.rho_avg := (1.0 - 0.0140)*0.919725239276886;
130      # PipeH
131      WITHIN Plant DO
132        PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
133        PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
134        PipeH.P_Out:=6950000.0;#PipeH.InitP;
135      END # within
136      # PipeC2
137      WITHIN Plant DO
138        PipeC2.T0:= 614.15;#InitT;
139        PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
140        PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
141      END # within
142    PRESET
143      INCLUDE  ..\HTLQSS_SS_sout\HTLQSS_SS_OUT_PRE
144    INITIAL
145      INCLUDE  ..\HTLQSS_SS_sout\HTLQSS_SS_OUT_INI
146    SCHEDULE
147      SEQUENCE
148        CONTINUE FOR 10  #10
149        DISPLAY  Plant.PipeC2.F(1) END
150        RESET
151          Plant.PipeC2.P_In:= 1120950.0 - (1120950.0 - 1000000.0 )*0.658*(TIME - OLD(TIME))/0.2;
152        END # reset
153        CONTINUE FOR 0.2
154        RESET
155          Plant.PipeC2.P_In:= 1120950.0 - (1120950.0 - 1000000.0 )*0.658;
156        END # reset
157        CONTINUE FOR 19.8 #20
158        DISPLAY  TIME, "seg" END
159        DISPLAY  Plant.PipeC2.F(1) END
160        SAVE PRESETS HTLQSS_QStepDecrFPHX_Event30s_Out_PRE
161        SAVE INITIAL HTLQSS_QStepDecrFPHX_Event30s_Out_INI
162      END # sequence
163  END # simulation
164  ######################################################################
165
166
167  ######################################################################
168  SIMULATION HTLQSS_QStepDecrFPHX_PostEvent1000s
169  ######################################################################
170    OPTIONS
171      INIT_PRINT_LEVEL:=0;
172      DYNAMIC_PRINT_LEVEL:=0;
173      DYNAMIC_REPORTING_INTERVAL:=1.0;
174      CSVOUTPUT:=TRUE;
175      DYNAMIC_BOUNDS := TRUE;
176      INIT_BLOCK_SOLVE:= TRUE;
177      INIT_RELATIVE_TOLERANCE := 1e-9;
178      INIT_ABSOLUTE_TOLERANCE := 1e-8;
179      REINIT_RELATIVE_TOLERANCE:= 1e-8;
180      REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
181      DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
182      DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
183      REINIT_MAX_ITERATIONS:= 2000;
184      REINIT_PRINT_LEVEL:= 0;
185      CHECK_MATH_EXCEPTIONS:= FALSE;
186      DYNAMIC_SCALING:= TRUE;
187      INITIALIZE_SENSITIVITIES := TRUE;
188    UNIT
```

```
189       Plant AS HeatTransferLoopQSSPipesFullDyn
190     SET
191       #Plant.Loop1
192       Plant.Loop1.InitDeltaP:=500.0;
193       Plant.Loop1.MaxDeltaP:= 304125;
194       # Plant.PipeH
195       Plant.PipeH.InitP:=6.95e6;
196       Plant.PipeH.InitT:= 273.15 + 530;
197       Plant.PipeC2.InitP:=1.00e6;
198       Plant.InitT:= 273.15+18;
199     INTERMEDIATE
200     INPUT
201       #Loop1
202       Plant.Loop1.DeltaP:=  331300;
203       Plant.Loop1.AdaptFactor1:=1;
204       Plant.Loop1.AdaptFactor2:=1;
205       Plant.Loop1.ModifyFlow := 0.00;
206       Plant.Loop1.rho_avg :=  (1.0 - 0.0140)*0.919725239276886;
207       # PipeH
208       WITHIN Plant DO
209         PipeH.T0:=1173.1500;  #PipeH.InitT;
210         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
211         PipeH.P_Out:=6950000.0;#PipeH.InitP;
212       END # within
213       # PipeC2
214       WITHIN Plant DO
215         PipeC2.T0:= 614.15;#InitT;
216         PipeC2.P_In:=1120950.0 - (1120950.0 - 1000000.0 )*0.658;     # 1110000.0;  #PipeC2.InitP+100;
217         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
218       END # within
219     PRESET
220       INCLUDE  ..\HTLQSS_QStepDecrFPHX_Event30s_sout\HTLQSS_QStepDecrFPHX_Event30s_Out_PRE
221     INITIAL
222       INCLUDE  ..\HTLQSS_QStepDecrFPHX_Event30s_sout\HTLQSS_QStepDecrFPHX_Event30s_Out_INI
223     SCHEDULE
224       SEQUENCE
225         CONTINUE FOR 1000
226         DISPLAY  TIME, "seg" END
227         DISPLAY  Plant.PipeC2.F(1) END
228         SAVE PRESETS HTLQSS_QStepDecrFPHX_PostEvent1000s_Out_PRE
229         SAVE INITIAL HTLQSS_QStepDecrFPHX_PostEvent1000s_Out_INI
230       END # sequence
231   END # simulation
232   ######################################################################
233
234
235   ######################################################################
236   SIMULATION HTLQSS_QStepIncDPLoopComp_Event160sToTime
237   ######################################################################
238     OPTIONS
239       INIT_PRINT_LEVEL:=0;
240       DYNAMIC_PRINT_LEVEL:=0;
241       DYNAMIC_REPORTING_INTERVAL:=0.01;
242       CSVOUTPUT:=TRUE;
243       DYNAMIC_BOUNDS := TRUE;
244       INIT_BLOCK_SOLVE:= TRUE;
245       INIT_RELATIVE_TOLERANCE := 1e-9;
246       INIT_ABSOLUTE_TOLERANCE := 1e-8;
247       REINIT_RELATIVE_TOLERANCE:= 1e-8;
248       REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
249       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
250       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
251       REINIT_MAX_ITERATIONS:= 2000;
252       REINIT_PRINT_LEVEL:= 0;
253       CHECK_MATH_EXCEPTIONS:= FALSE;
254       DYNAMIC_SCALING:= TRUE;
255       INITIALIZE_SENSITIVITIES := TRUE;
256     UNIT
257       Plant AS HeatTransferLoopQSSPipesFullDyn
258     SET
259       #Plant.Loop1
260       Plant.Loop1.InitDeltaP:=500.0;
261       Plant.Loop1.MaxDeltaP:=0.321e6;
262       # Plant.PipeH
263       Plant.PipeH.InitP:=6.95e6;
264       Plant.PipeH.InitT:= 273.15 + 530;
265       Plant.PipeC2.InitP:=1.00e6;
266       Plant.InitT:= 273.15+18;
267     INTERMEDIATE
268     INPUT
269       #Loop1
270       Plant.Loop1.DeltaP:=  331300;
271       Plant.Loop1.AdaptFactor1:=1;
272       Plant.Loop1.AdaptFactor2:=1;
273       Plant.Loop1.ModifyFlow := 0.00;
274       Plant.Loop1.rho_avg := (1.0 - 0.0140)*0.919725239276886;
275       # PipeH
276       WITHIN Plant DO
277         PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
278         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
279         PipeH.P_Out:=6950000.0;#PipeH.InitP;
280       END # within
```

```
281       # PipeC2
282       WITHIN Plant DO
283         PipeC2.T0:= 614.15;#InitT;
284         PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
285         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
286       END # within
287     PRESET
288       INCLUDE  ..\HTLQSS_SS_sout\HTLQSS_SS_Out_PRE
289     INITIAL
290       INCLUDE  ..\HTLQSS_SS_sout\HTLQSS_SS_Out_INI
291     SCHEDULE
292       SEQUENCE
293         CONTINUE FOR 10  #10
294         RESET
295           Plant.Loop1.DeltaP:= 331300 + 47515*(TIME-10)/0.2;
296         END # reset
297         CONTINUE FOR 0.2
298         RESET
299           Plant.Loop1.DeltaP:= 331300 + 47515;
300         END # reset
301         CONTINUE FOR 149.8 #20
302         DISPLAY  TIME, "seg" END
303         DISPLAY  Plant.PipeC2.F(1) END
304         SAVE PRESETS HTLQSS_QStepDecrFPHX_Event160s_Out_PRE
305         SAVE INITIAL HTLQSS_QStepDecrFPHX_Event160s_Out_INI
306       END # sequence
307   END # simulation
308   ######################################################################
309
310
311   ######################################################################
312   SIMULATION HTLQSS_QStepIncDPLoopComp_Event30s
313   ######################################################################
314     OPTIONS
315       INIT_PRINT_LEVEL:=0;
316       DYNAMIC_PRINT_LEVEL:=0;
317       DYNAMIC_REPORTING_INTERVAL:= 0.005; # 0.1;
318       CSVOUTPUT:=TRUE;
319       DYNAMIC_BOUNDS := TRUE;
320       INIT_BLOCK_SOLVE:= TRUE;
321       INIT_RELATIVE_TOLERANCE := 1e-9;
322       INIT_ABSOLUTE_TOLERANCE := 1e-8;
323       REINIT_RELATIVE_TOLERANCE:= 1e-8;
324       REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
325       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
326       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
327       REINIT_MAX_ITERATIONS:= 2000;
328       REINIT_PRINT_LEVEL:= 0;
329       CHECK_MATH_EXCEPTIONS:= FALSE;
330       DYNAMIC_SCALING:= TRUE;
331       INITIALIZE_SENSITIVITIES := TRUE;
332     UNIT
333       Plant AS HeatTransferLoopQSSPipesFullDyn
334     SET
335       #Plant.Loop1
336       Plant.Loop1.InitDeltaP:=500.0;
337       Plant.Loop1.MaxDeltaP:=304125;
338       # Plant.PipeH
339       Plant.PipeH.InitP:=6.95e6;
340       Plant.PipeH.InitT:= 273.15 + 530;
341       Plant.PipeC2.InitP:=1.00e6;
342       Plant.InitT:= 273.15+18;
343     INTERMEDIATE
344     INPUT
345       #Loop1
346       Plant.Loop1.DeltaP:=  331300;
347       Plant.Loop1.AdaptFactor1:=1;
348       Plant.Loop1.AdaptFactor2:=1;
349       Plant.Loop1.ModifyFlow := 0.00;
350       Plant.Loop1.rho_avg := (1.0 - 0.0140)*0.919725239276886;
351       # PipeH
352       WITHIN Plant DO
353         PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
354         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
355         PipeH.P_Out:=6950000.0;#PipeH.InitP;
356       END # within
357       # PipeC2
358       WITHIN Plant DO
359         PipeC2.T0:= 614.15;#InitT;
360         PipeC2.P_In:=1110000.0 + 10950.0;  #PipeC2.InitP+100;
361         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
362       END # within
363     PRESET
364       INCLUDE  ..\HTLQSS_SS_sout\HTLQSS_SS_Out_PRE
365     INITIAL
366       INCLUDE  ..\HTLQSS_SS_sout\HTLQSS_SS_Out_INI
367     SCHEDULE
368       SEQUENCE
369         CONTINUE FOR 10
370         RESET
371           Plant.Loop1.DeltaP:= 331300 + 47515*(TIME-10)/0.2;
372         END # reset
```

```
373         CONTINUE FOR 0.2
374         RESET
375           Plant.Loop1.DeltaP:= 331300 + 47515;
376         END # reset
377         CONTINUE FOR 19.9# 20
378         DISPLAY  TIME, "seg" END
379         DISPLAY Plant.Loop1.F(1) END
380         SAVE PRESETS HTLQSS_QStepIncDPLoopComp_Event30s_Out_PRE
381         SAVE INITIAL HTLQSS_QStepIncDPLoopComp_Event30s_Out_INI
382       END # sequence
383   END # simulation
384   ######################################################################
385
386
387   ######################################################################
388   SIMULATION HTLQSS_QStepIncDPLoopComp_PostEvent1000s
389   ######################################################################
390     OPTIONS
391       INIT_PRINT_LEVEL:=0;
392       DYNAMIC_PRINT_LEVEL:=0;
393       DYNAMIC_REPORTING_INTERVAL:= 1.0; # 0.1;
394       CSVOUTPUT:=TRUE;
395       DYNAMIC_BOUNDS := TRUE;
396       INIT_BLOCK_SOLVE:= TRUE;
397       INIT_RELATIVE_TOLERANCE := 1e-9;
398       INIT_ABSOLUTE_TOLERANCE := 1e-8;
399       REINIT_RELATIVE_TOLERANCE:= 1e-8;
400       REINIT_ABSOLUTE_TOLERANCE:= 1e-8;
401       DYNAMIC_RELATIVE_TOLERANCE:= 1e-6;
402       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8;
403       REINIT_MAX_ITERATIONS:= 2000;
404       REINIT_PRINT_LEVEL:= 0;
405       CHECK_MATH_EXCEPTIONS:= FALSE;
406       DYNAMIC_SCALING:= TRUE;
407       INITIALIZE_SENSITIVITIES := TRUE;
408     UNIT
409       Plant AS HeatTransferLoopQSSPipesFullDyn
410     SET
411       #Plant.Loop1
412       Plant.Loop1.InitDeltaP:=500.0;
413       Plant.Loop1.MaxDeltaP:=304125;
414       # Plant.PipeH
415       Plant.PipeH.InitP:=6.95e6;
416       Plant.PipeH.InitT:= 273.15 + 530;
417       Plant.PipeC2.InitP:=1.00e6;
418       Plant.InitT:= 273.15+18;
419     INTERMEDIATE
420     INPUT
421       #Loop1
422       Plant.Loop1.DeltaP:=  331300 + 47515;
423       Plant.Loop1.AdaptFactor1:=1;
424       Plant.Loop1.AdaptFactor2:=1;
425       Plant.Loop1.ModifyFlow := 0.00;
426       Plant.Loop1.rho_avg := (1.0 - 0.0140)*0.919725239276886;
427       # PipeH
428       WITHIN Plant DO
429         PipeH.T0:=1173.1500244140625;  #PipeH.InitT;
430         PipeH.P_In:= 7000000.0; #PipeH.InitP + 100;
431         PipeH.P_Out:=6950000.0;#PipeH.InitP;
432       END # within
433       # PipeC2
434       WITHIN Plant DO
435         PipeC2.T0:= 614.15;#InitT;
436         PipeC2.P_In:=1110000.0 + 10950.0;  #PipeC2.InitP+100;
437         PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
438       END # within
439     PRESET
440       INCLUDE  ..\HTLQSS_QStepIncDPLoopComp_Event30s_sout\HTLQSS_QStepIncDPLoopComp_Event30s_Out_PRE
441     INITIAL
442       INCLUDE  ..\HTLQSS_QStepIncDPLoopComp_Event30s_sout\HTLQSS_QStepIncDPLoopComp_Event30s_Out_INI
443     SCHEDULE
444       SEQUENCE
445         CONTINUE FOR 1000
446         DISPLAY  TIME, "seg" END
447         DISPLAY Plant.Loop1.F(1) END
448         SAVE PRESETS HTLQSS_QStepIncDPLoopComp_PostEvent1000s_Out_PRE
449         SAVE INITIAL HTLQSS_QStepIncDPLoopComp_PostEvent1000s_Out_INI
450       END # sequence
451   END # simulation
452   ######################################################################
```

SIMULATIONS_CH4_HTLANDNUCLEARREACTOR_QSS.JAC

```
1   INCLUDE MODELS_PHYSICALPROPERTIES
2   INCLUDE MODELS_HEATEXCHANGERS
3   INCLUDE MODELS_NUCLEARREACTORPROPERTIES
4   INCLUDE MODELS_NUCLEARREACTORANDAUXS
5   INCLUDE MODELS_NUCLEARREACTORLOOP
6   INCLUDE MODELS_PIPEMODELS
7   INCLUDE MODELS_HEATTRANSFERLOOP
8   ######################################################################
9   SIMULATION NuclearReactorAndHTLQSS_ULOF_FullEvent
10  ######################################################################
11    OPTIONS
12      INIT_PRINT_LEVEL:=0; DYNAMIC_PRINT_LEVEL:=0;
13      DYNAMIC_REPORTING_INTERVAL:=1; CSVOUTPUT:=TRUE;
14      DYNAMIC_BOUNDS := TRUE;
15      INIT_BLOCK_SOLVE:= TRUE; INIT_RELATIVE_TOLERANCE := 1e-9;
16      INIT_ABSOLUTE_TOLERANCE := 1e-8; REINIT_RELATIVE_TOLERANCE:= 1e-7;
17      REINIT_ABSOLUTE_TOLERANCE:= 1e-8; DYNAMIC_RELATIVE_TOLERANCE:= 1e-7;
18      DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8; REINIT_MAX_ITERATIONS:= 1000;
19      REINIT_PRINT_LEVEL:= 0; CHECK_MATH_EXCEPTIONS:= FALSE;
20      DYNAMIC_SCALING:= TRUE; INITIALIZE_SENSITIVITIES := TRUE;
21    PARAMETER
22      # Loop2
23        QFission_K0 AS REAL
24        C0 AS REAL
25        TTrans AS REAL
26        TCoreOut_SetPoint AS REAL
27        Xe0 AS REAL
28        IConc0 AS REAL
29    UNIT
30      # Loop2
31        Plant2 AS PlantNR
32      #HeatTransferLoopQSSNoPipeH
33        Plant AS HeatTransferLoopQSSNoPipeH
34    VARIABLE
35      TCoreOutGas As Temperature
36      TFuelAvg AS Temperature
37      TFuelMax AS Temperature
38    SET
39      # Simulation
40        #SET
41        QFission_K0:=  250e6/1e8*1/100;
42        C0:= 5.3510274e10/1e8*1/100;
43        TTrans:= 300; # s? ### fix this
44        TCoreOut_SetPoint := 900 + 273.16; # L1530
45        #CALCULATED
46        IConc0:= Plant2.NR.Poison.gamma_I*Plant2.NR.Poison.sigma_total/Plant2.NR.Poison.lambda_I*Plant2.NR.Poison.
              phi0;
47        Xe0:= (Plant2.NR.Poison.gamma_xe+Plant2.NR.Poison.gamma_I)*
48          Plant2.NR.Poison.sigma_total/(Plant2.NR.Poison.lambda_xe+Plant2.NR.Poison.sigma_xe*Plant2.NR.Poison.phi0)
                *Plant2.NR.Poison.phi0;
49      # Loop2
50        Plant2.NR.WPrim0 := 126.6;
51        Plant2.NR.Poison.phi0 := 1e18;
52        Plant2.PipeH.InitP:=6.95e6;
53        Plant2.PipeH.rho:= 3.93;
54      # Plant2
55        Plant2.InitDeltaP := 100;
56        Plant2.MaxDeltaP := 0.29e6;
57        Plant2.InitT:= 300; # 273.15+530;
58        Plant2.WpcupipeRatio:=3.18;
59      # HeatTransferLoopQSSNoPipeH
60        Plant.Loop1.InitDeltaP:=500.0;
61        Plant.Loop1.MaxDeltaP:=0.321e6;
62        Plant.PipeC2.InitP:=1.00e6;
63        Plant.InitT:= 273.15+18;
64    INTERMEDIATE
65      # Loop2
66        # PIPEH
67        i_UH:= Plant2.PipeH.HT.U;
68        i_r0H:= Plant2.PipeH.r0;
69        i_VH2:= Plant2.PipeH.Vol;
70        # heat calculation
71        QH:= sigma(4*i_UH/(2*i_r0H)*(Plant2.PipeH.TExtrnl(1:Plant2.PipeH.NG)
72          -Plant2.PipeH.T(1:Plant2.PipeH.NG))*i_VH2);
73        QHdiff:= Plant2.PipeH.A*Plant2.PipeH.rho(Plant2.PipeH.NG)*Plant2.PipeH.v(Plant2.PipeH.NG)*Plant2.PipeH.h(
              Plant2.PipeH.NG) -
74        Plant2.PipeH.A*Plant2.PipeH.rho(1)*Plant2.PipeH.v(1)*Plant2.PipeH.h(1);
75        mdotH:= Plant2.PipeH.rho(1)*Plant2.PipeH.v(1)*Plant2.PipeH.A;
76  EQUATION
77      # Loop2
78        FOR I:=1 TO Plant2.PipeH.NG DO
79          Plant2.PipeH.TExtrnl(Plant2.PipeH.NG+1-I) = Plant.HX.THX(I);  ###
80        END # for
81      # Dummy variables for reporting
82        TCoreOutGas = Plant2.NR.THyd.TCoreOut - 273.16;
83        TFuelAvg = Plant2.NR.Thyd.T_FuelAvg - 273.16;
84        TFuelMax = Plant2.NR.THyd.TSolid(10,2) - 273.16;
85      # HeatTransferLoopQSSNoPipeH
86        Plant.HX.UHot = Plant2.PipeH.HT.U;
87        FOR I:= 1 TO Plant.HX.NGHX DO
88          Plant.HX.TGasHot(I) = Plant2.PipeH.T(Plant2.PipeH.NG + 1 - I);
89        END # for
```

```
INPUT
    # Loop2
      Plant2.NR.PhoControl := 0.0; # 0.0001;
      Plant2.NR.wPhoExtra1 := 1.0;
      Plant2.NR.PhoExtra2 := 0.0;
      Plant2.Comp.P2:= 7.0e6;
      Plant2.Comp.PR:= 1.0001;  # we start at a low number
    # HeatTransferLoopQSSNoPipeH
      Plant.Loop1.DeltaP:=  331300;
      Plant.Loop1.AdaptFactor1:=1;
      Plant.Loop1.AdaptFactor2:=1;
      Plant.Loop1.ModifyFlow := 0.00;
      Plant.Loop1.rho_avg := (1.0 - 0.0140)*0.919725239276886;
      WITHIN Plant DO
        PipeC2.T0:= 614.15;#InitT;
        PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
        PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
      END # within
PRESET
    # Loop2
      WITHIN Plant2 DO
        FOR I:= 1 TO PipeH.NG DO
          PipeH.P(I):= PipeH.InitP;
          PipeH.T(I):= PipeH.Temp;
          Comp.P1:= PipeH.InitP;
        END # for
      END # within
    # HeatTransferLoopQSSNoPipeH
      INCLUDE ..\input\HTLNOPIPEHQSS_SS_OUT_PRE
INITIAL
    # Loop2
      WITHIN Plant2 DO
        FOR I:=2 TO PipeH.NG DO
          PipeH.$rho_E(I)= 0;
        END # for
      END # within
          Plant2.NR.THyd.TSolid = 600;
      Plant2.NR.Kinetics.QFission_K = QFission_K0;
      Plant2.NR.Kinetics.C = C0;
      Plant2.NR.Poison.IConc = IConc0;
      Plant2.NR.Poison.Xe = Xe0;
      INCLUDE ..\input\HTLNOPIPEHQSS_SS_OUT_INI
SCHEDULE
    SEQUENCE
      CONTINUE FOR 10*60
      # Loop2: start compressor 2 until reaching target speed in coolant loop
      RESET
        Plant2.Comp.PR := 1.0001 + 0.03*(TIME - OLD(TIME))/(6*60);
      END # reset
      CONTINUE UNTIL mdotH > 32.0 # Plant2.NR.THyd.WPrim > 124
      RESET
        Plant2.Comp.PR := OLD(Plant2.Comp.PR);
      END # reset
      CONTINUE FOR 1*60
      # Loop2: increase reactivity power reaches 250e6 W
      RESET
          Plant2.NR.PhoControl := 0.0001*(TIME-OLD(TIME))/(5*60);
      END # reset
      CONTINUE FOR 5*60
      RESET
          Plant2.NR.PhoControl := OLD(Plant2.NR.PhoControl);
      END # reset
      CONTINUE UNTIL Plant2.NR.THyd.QFission > 250e6
      CONTINUE FOR 17.1  # this extra time lets QFission reach the right value.
      RESET
          Plant2.NR.PhoControl := 0.0;
      END # reset
      CONTINUE FOR 1*60
      DISPLAY Plant2.NR.THyd.QFission END
      DISPLAY "17.1" END
      CONTINUE FOR 800*60
      SAVE PRESETS PRESETS_NuclearReactorOperating
      SAVE INITIAL INITIAL_NuclearReactorOperating
      # Insert Temperature Reactivity
      RESET
        Plant2.NR.wPhoExtra1:= 0;
        Plant2.NR.PhoExtra2:=OLD(Plant2.NR.PhoExtra1);
      END # reset
      CONTINUE FOR 1*60
      DISPLAY TIME END
      RESET
        Plant2.NR.PhoExtra2:=OLD(Plant2.NR.PhoExtra2); # trick to eliminate problems of phoextra2 jumping during
              transient
      END
      CONTINUE FOR 1*60
      DISPLAY Plant.Loop1.F(1) END
      DISPLAY Plant.Loop1.DeltaP END
      DISPLAY "This is QH", QH, "This is QHdiff", QHdiff END
      # This is the new transient.  I will stop the flow in the intermediate loop.
      RESET
        Plant.Loop1.DeltaP := 328300 - (328300 - 2500.0)*(TIME - OLD(TIME))/5;
      END # reset
```

```
181         CONTINUE FOR 5
182         RESET
183           Plant.Loop1.DeltaP := 2500.0;
184         END # reset
185         CONTINUE FOR 100*60
186         DISPLAY Plant.Loop1.F(1) END
187         DISPLAY "This is QH", QH, "This is QHdiff", QHdiff END
188       END # sequence
189   END # simulation
190   ######################################################################
191
192
193   ######################################################################
194   SIMULATION NuclearReactorAndHTLQSS_HTLLeak_FullEvent
195   ######################################################################
196     OPTIONS
197       INIT_PRINT_LEVEL:=0; DYNAMIC_PRINT_LEVEL:=0;
198       DYNAMIC_REPORTING_INTERVAL:=1; CSVOUTPUT:=TRUE;
199       DYNAMIC_BOUNDS := TRUE;
200       INIT_BLOCK_SOLVE:= TRUE; INIT_RELATIVE_TOLERANCE := 1e-9;
201       INIT_ABSOLUTE_TOLERANCE := 1e-8; REINIT_RELATIVE_TOLERANCE:= 1e-7;
202       REINIT_ABSOLUTE_TOLERANCE:= 1e-8; DYNAMIC_RELATIVE_TOLERANCE:= 1e-7;
203       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8; REINIT_MAX_ITERATIONS:= 1000;
204       REINIT_PRINT_LEVEL:= 0; CHECK_MATH_EXCEPTIONS:= FALSE;
205       DYNAMIC_SCALING:= TRUE; INITIALIZE_SENSITIVITIES := TRUE;
206     PARAMETER
207       # Loop2
208         QFission_K0 AS REAL
209         C0 AS REAL
210         TTrans AS REAL
211         TCoreOut_SetPoint AS REAL
212         Xe0 AS REAL
213         IConc0 AS REAL
214     UNIT
215       # Loop2
216         Plant2 AS PlantNR
217       #HeatTransferLoopQSSNoPipeH
218         Plant AS HeatTransferLoopQSSNoPipeH
219     VARIABLE
220       TCoreOutGas As Temperature
221       TFuelAvg AS Temperature
222       TFuelMax AS Temperature
223     SET
224       # Simulation
225         #SET
226         QFission_K0:=  250e6/1e8*1/100;
227         C0:= 5.3510274e10/1e8*1/100;
228         TTrans:= 300; # s? ### fix this
229         TCoreOut_SetPoint := 900 + 273.16; # L1530
230         #CALCULATED
231         IConc0:= Plant2.NR.Poison.gamma_I*Plant2.NR.Poison.sigma_total/Plant2.NR.Poison.lambda_I*Plant2.NR.Poison.
              phi0;
232         Xe0:= (Plant2.NR.Poison.gamma_xe+Plant2.NR.Poison.gamma_I)*
233           Plant2.NR.Poison.sigma_total/(Plant2.NR.Poison.lambda_xe+Plant2.NR.Poison.sigma_xe*Plant2.NR.Poison.phi0)
                *Plant2.NR.Poison.phi0;
234       # Loop2
235         Plant2.NR.WPrim0 := 126.6;
236         Plant2.NR.Poison.phi0 := 1e18;
237         Plant2.PipeH.InitP:=6.95e6;
238         Plant2.PipeH.rho:= 3.93;
239       # Plant2
240         Plant2.InitDeltaP := 100;
241         Plant2.MaxDeltaP := 0.29e6;
242         Plant2.InitT:= 300; # 273.15+530;
243         Plant2.WpcupipeRatio:=3.17;
244       # HeatTransferLoopQSSNoPipeH
245         Plant.Loop1.InitDeltaP:=500.0;
246         Plant.Loop1.MaxDeltaP:=0.321e6;
247         Plant.PipeC2.InitP:=1.00e6;
248         Plant.InitT:= 273.15+18;
249     INTERMEDIATE
250         # Loop2
251           # PIPEH
252           i_UH:= Plant2.PipeH.HT.U;
253           i_r0H:= Plant2.PipeH.r0;
254           i_VH2:= Plant2.PipeH.Vol;
255           # heat calculation
256           QH:= sigma(4*i_UH/(2*i_r0H)*(Plant2.PipeH.TExtrnl(1:Plant2.PipeH.NG)
257             -Plant2.PipeH.T(1:Plant2.PipeH.NG))*i_VH2);
258           QHdiff:= Plant2.PipeH.A*Plant2.PipeH.rho(Plant2.PipeH.NG)*Plant2.PipeH.v(Plant2.PipeH.NG)*Plant2.PipeH.h(
                Plant2.PipeH.NG) -
259           Plant2.PipeH.A*Plant2.PipeH.rho(1)*Plant2.PipeH.v(1)*Plant2.PipeH.h(1);
260           mdotH:= Plant2.PipeH.rho(1)*Plant2.PipeH.v(1)*Plant2.PipeH.A;
261     EQUATION
262       # Loop2
263         FOR I:=1 TO Plant2.PipeH.NG DO
264           Plant2.PipeH.TExtrnl(Plant2.PipeH.NG+1-I) = Plant.HX.THX(I);  ###
265         END # for
266       # Dummy variables for reporting
267         TCoreOutGas = Plant2.NR.THyd.TCoreOut - 273.16;
268         TFuelAvg = Plant2.NR.Thyd.T_FuelAvg - 273.16;
269         TFuelMax = Plant2.NR.THyd.TSolid(10,2) - 273.16;
```

```
270       # HeatTransferLoopQSSNoPipeH
271         Plant.HX.UHot = Plant2.PipeH.HT.U;
272         FOR I:= 1 TO Plant.HX.NGHX DO
273           Plant.HX.TGasHot(I) = Plant2.PipeH.T(Plant2.PipeH.NG + 1 - I);
274         END # for
275     INPUT
276       # Loop2
277         Plant2.NR.PhoControl := 0.0; # 0.0001;
278         Plant2.NR.wPhoExtra1 := 1.0;
279         Plant2.NR.PhoExtra2 := 0.0;
280         Plant2.Comp.P2:= 7.0e6;
281         Plant2.Comp.PR:= 1.0001;  # we start at a low number
282       # HeatTransferLoopQSSNoPipeH
283         Plant.Loop1.DeltaP:=  328300;
284         Plant.Loop1.AdaptFactor1:=1;
285         Plant.Loop1.AdaptFactor2:=1;
286         Plant.Loop1.ModifyFlow := 0.00;
287         Plant.Loop1.rho_avg :=  (1.0 - 0.0140)*0.919725239276886;
288         WITHIN Plant DO
289           PipeC2.T0:= 614.15;#InitT;
290           PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
291           PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
292         END # within
293     PRESET
294       # Loop2
295         WITHIN Plant2 DO
296           FOR I:= 1 TO PipeH.NG DO
297             PipeH.P(I):= PipeH.InitP;
298             PipeH.T(I):= PipeH.Temp;
299             Comp.P1:= PipeH.InitP;
300           END # for
301         END # within
302       # HeatTransferLoopQSSNoPipeH
303         INCLUDE ..\input\HTLNOPIPEHQSS_SS_OUT_PRE
304     INITIAL
305       # Loop2
306         WITHIN Plant2 DO
307           FOR I:=2 TO PipeH.NG DO
308             PipeH.$rho_E(I)= 0;
309           END # for
310         END # within
311             Plant2.NR.THyd.TSolid = 600;
312         Plant2.NR.Kinetics.QFission_K = QFission_K0;
313         Plant2.NR.Kinetics.C = C0;
314         Plant2.NR.Poison.IConc = IConc0;
315         Plant2.NR.Poison.Xe = Xe0;
316         INCLUDE ..\input\HTLNOPIPEHQSS_SS_OUT_INI
317     SCHEDULE
318       SEQUENCE
319         CONTINUE FOR 10*60
320         # Loop2: start compressor 2 until reaching target speed in coolant loop
321         RESET
322           Plant2.Comp.PR := 1.0001 + 0.03*(TIME - OLD(TIME))/(6*60);
323         END # reset
324         CONTINUE UNTIL mdotH > 32.0 # Plant2.NR.THyd.WPrim > 124
325         RESET
326           Plant2.Comp.PR := OLD(Plant2.Comp.PR);
327         END # reset
328         CONTINUE FOR 1*60
329         # Loop2: increase reactivity power reaches 250e6 W
330         RESET
331             Plant2.NR.PhoControl := 0.0001*(TIME-OLD(TIME))/(5*60);
332         END # reset
333         CONTINUE FOR 5*60
334         RESET
335             Plant2.NR.PhoControl := OLD(Plant2.NR.PhoControl);
336         END # reset
337         CONTINUE UNTIL Plant2.NR.THyd.QFission > 250e6
338         CONTINUE FOR 17.1  # this extra time lets QFission reach the right value.
339         RESET
340             Plant2.NR.PhoControl := 0.0;
341         END # reset
342         CONTINUE FOR 1*60
343         DISPLAY Plant2.NR.THyd.QFission END
344         DISPLAY "17.1" END
345         CONTINUE FOR 800*60
346         SAVE PRESETS PRESETS_NuclearReactorOperating
347         SAVE INITIAL INITIAL_NuclearReactorOperating
348         # Insert Temperature Reactivity
349         RESET
350           Plant2.NR.wPhoExtra1:= 0;
351           Plant2.NR.PhoExtra2:=OLD(Plant2.NR.PhoExtra1);
352         END # reset
353         CONTINUE FOR 1*60
354         DISPLAY TIME END
355         RESET
356           Plant2.NR.PhoExtra2:=OLD(Plant2.NR.PhoExtra2); # trick to eliminate problems of phoextra2 jumping during
                transient
357         END
358         CONTINUE FOR 1*60
359         DISPLAY Plant.Loop1.F(1) END
360         DISPLAY Plant.Loop1.DeltaP END
```

```
361         # This is the new transient.  50\% decrease in helium mass in loop
362         RESET
363           Plant.Loop1.rho_avg := (1.0 - 0.0140)*0.919725239276886 - 0.50* (1.0 - 0.0140)*0.919725239276886*(TIME-
                OLD(TIME))/60;
364         END
365         CONTINUE FOR 60
366         RESET
367           Plant.Loop1.rho_avg := 0.50* (1.0 - 0.0140)*0.919725239276886;
368         END # reset
369         CONTINUE FOR 100*60
370         DISPLAY "This is QH", QH, "This is QHdiff", QHdiff END
371         DISPLAY "This is mdotH", mdotH END
372         DISPLAY "Thotout", (Plant2.PipeH.T(Plant2.PipeH.NG) - 273.15) END
373         DISPLAY Plant2.NR.THyd.TSolid, Plant2.NR.THyd.TGas END
374         DISPLAY "This is QC2", QC2, "This is QC2diff", QC2diff END
375         DISPLAY "This is mdotC2", mdotC2 END
376         DISPLAY "This is mdotH", mdotH END
377         DISPLAY "This is mdotC2", mdotC2 END
378       END # sequence
379   END # simulation
380   ######################################################################
381
382
383   ######################################################################
384   SIMULATION NuclearReactorAndHTLQSS_LossHeatSink_FullEvent
385   ######################################################################
386     OPTIONS
387       INIT_PRINT_LEVEL:=0; DYNAMIC_PRINT_LEVEL:=0;
388       DYNAMIC_REPORTING_INTERVAL:=1; CSVOUTPUT:=TRUE;
389       DYNAMIC_BOUNDS := TRUE;
390       INIT_BLOCK_SOLVE:= TRUE; INIT_RELATIVE_TOLERANCE := 1e-9;
391       INIT_ABSOLUTE_TOLERANCE := 1e-8; REINIT_RELATIVE_TOLERANCE:= 1e-7;
392       REINIT_ABSOLUTE_TOLERANCE:= 1e-8; DYNAMIC_RELATIVE_TOLERANCE:= 1e-7;
393       DYNAMIC_ABSOLUTE_TOLERANCE:= 1e-8; REINIT_MAX_ITERATIONS:= 1000;
394       REINIT_PRINT_LEVEL:= 0; CHECK_MATH_EXCEPTIONS:= FALSE;
395       DYNAMIC_SCALING:= TRUE; INITIALIZE_SENSITIVITIES := TRUE;
396     PARAMETER
397       # Loop2
398         QFission_K0 AS REAL
399         C0 AS REAL
400         TTrans AS REAL
401         TCoreOut_SetPoint AS REAL
402         Xe0 AS REAL
403         IConc0 AS REAL
404     UNIT
405       # Loop2
406         Plant2 AS PlantNR
407       #HeatTransferLoopQSSNoPipeH
408         Plant AS HeatTransferLoopQSSNoPipeH
409     VARIABLE
410       TCoreOutGas As Temperature
411       TFuelAvg AS Temperature
412       TFuelMax AS Temperature
413     SET
414       # Simulation
415         #SET
416         QFission_K0:=  250e6/1e8*1/100;
417         C0:= 5.3510274e10/1e8*1/100;
418         TTrans:= 300; # s? ### fix this
419         TCoreOut_SetPoint := 900 + 273.16; # L1530
420         #CALCULATED
421         IConc0:= Plant2.NR.Poison.gamma_I*Plant2.NR.Poison.sigma_total/Plant2.NR.Poison.lambda_I*Plant2.NR.Poison.
              phi0;
422         Xe0:= (Plant2.NR.Poison.gamma_xe+Plant2.NR.Poison.gamma_I)*
423           Plant2.NR.Poison.sigma_total/(Plant2.NR.Poison.lambda_xe+Plant2.NR.Poison.sigma_xe*Plant2.NR.Poison.phi0)
                *Plant2.NR.Poison.phi0;
424       # Loop2
425         Plant2.NR.WPrim0 := 126.6;
426         Plant2.NR.Poison.phi0 := 1e18;
427         Plant2.PipeH.InitP:=6.95e6;
428         Plant2.PipeH.rho:= 3.93;
429       # Plant2
430         Plant2.InitDeltaP := 100;
431         Plant2.MaxDeltaP := 0.29e6;
432         Plant2.InitT:= 300; # 273.15+530;
433         Plant2.WpcupipeRatio:=3.17;
434       # HeatTransferLoopQSSNoPipeH
435         Plant.Loop1.InitDeltaP:=500.0;
436         Plant.Loop1.MaxDeltaP:=0.321e6;
437         Plant.PipeC2.InitP:=1.00e6;
438         Plant.InitT:= 273.15+18;
439     INTERMEDIATE
440         # Loop2
441           # PIPEH
442           i_UH:= Plant2.PipeH.HT.U;
443           i_r0H:= Plant2.PipeH.r0;
444           i_VH2:= Plant2.PipeH.Vol;
445           # heat calculation
446           QH:= sigma(4*i_UH/(2*i_r0H)*(Plant2.PipeH.TExtrnl(1:Plant2.PipeH.NG)
447             -Plant2.PipeH.T(1:Plant2.PipeH.NG))*i_VH2);
448           QHdiff:= Plant2.PipeH.A*Plant2.PipeH.rho(Plant2.PipeH.NG)*Plant2.PipeH.v(Plant2.PipeH.NG)*Plant2.PipeH.h(
                Plant2.PipeH.NG) -
```

```
449         Plant2.PipeH.A*Plant2.PipeH.rho(1)*Plant2.PipeH.v(1)*Plant2.PipeH.h(1);
450         mdotH:= Plant2.PipeH.rho(1)*Plant2.PipeH.v(1)*Plant2.PipeH.A;
451     EQUATION
452       # Loop2
453         FOR I:=1 TO Plant2.PipeH.NG DO
454           Plant2.PipeH.TExtrnl(Plant2.PipeH.NG+1-I) = Plant.HX.THX(I);  ###
455         END # for
456       # Dummy variables for reporting
457         TCoreOutGas = Plant2.NR.THyd.TCoreOut - 273.16;
458         TFuelAvg = Plant2.NR.Thyd.T_FuelAvg - 273.16;
459         TFuelMax = Plant2.NR.THyd.TSolid(10,2) - 273.16;
460       # HeatTransferLoopQSSNoPipeH
461         Plant.HX.UHot = Plant2.PipeH.HT.U;
462         FOR I:= 1 TO Plant.HX.NGHX DO
463           Plant.HX.TGasHot(I) = Plant2.PipeH.T(Plant2.PipeH.NG + 1 - I);
464         END # for
465     INPUT
466       # Loop2
467         Plant2.NR.PhoControl := 0.0; # 0.0001;
468         Plant2.NR.wPhoExtra1 := 1.0;
469         Plant2.NR.PhoExtra2 := 0.0;
470         Plant2.Comp.P2:= 7.0e6;
471         Plant2.Comp.PR:= 1.0001;  # we start at a low number
472       # HeatTransferLoopQSSNoPipeH
473         Plant.Loop1.DeltaP:=  328300;
474         Plant.Loop1.AdaptFactor1:=1;
475         Plant.Loop1.AdaptFactor2:=1;
476         Plant.Loop1.ModifyFlow := 0.00;
477         Plant.Loop1.rho_avg :=  (1.0 - 0.0140)*0.919725239276886;
478         WITHIN Plant DO
479           PipeC2.T0:= 614.15;#InitT;
480           PipeC2.P_In:=1120950.0; #PipeC2.InitP+100;
481           PipeC2.P_Out:=1000000.0; #PipeC2.InitP;
482         END # within
483     PRESET
484       # Loop2
485         WITHIN Plant2 DO
486           FOR I:= 1 TO PipeH.NG DO
487             PipeH.P(I):= PipeH.InitP;
488             PipeH.T(I):= PipeH.Temp;
489             Comp.P1:= PipeH.InitP;
490           END # for
491         END # within
492       # HeatTransferLoopQSSNoPipeH
493         INCLUDE ..\input\HTLNOPIPEHQSS_SS_OUT_PRE
494     INITIAL
495       # Loop2
496         WITHIN Plant2 DO
497           FOR I:=2 TO PipeH.NG DO
498             PipeH.$rho_E(I)= 0;
499           END # for
500         END # within
501             Plant2.NR.THyd.TSolid = 600;
502         Plant2.NR.Kinetics.QFission_K = QFission_K0;
503         Plant2.NR.Kinetics.C = C0;
504         Plant2.NR.Poison.IConc = IConc0;
505         Plant2.NR.Poison.Xe = Xe0;
506         INCLUDE ..\input\HTLNOPIPEHQSS_SS_OUT_INI
507     SCHEDULE
508       SEQUENCE
509         CONTINUE FOR 10*60
510         # Loop2: start compressor 2 until reaching target speed in coolant loop
511         RESET
512           Plant2.Comp.PR := 1.0001 + 0.03*(TIME - OLD(TIME))/(6*60);
513         END # reset
514         CONTINUE UNTIL mdotH > 32.0 # Plant2.NR.THyd.WPrim > 124
515         RESET
516           Plant2.Comp.PR := OLD(Plant2.Comp.PR);
517         END # reset
518         CONTINUE FOR 1*60
519         # Loop2: increase reactivity power reaches 250e6 W
520         RESET
521             Plant2.NR.PhoControl := 0.0001*(TIME-OLD(TIME))/(5*60);
522         END # reset
523         CONTINUE FOR 5*60
524         RESET
525             Plant2.NR.PhoControl := OLD(Plant2.NR.PhoControl);
526         END # reset
527         CONTINUE UNTIL Plant2.NR.THyd.QFission > 250e6
528         CONTINUE FOR 17.1  # this extra time lets QFission reach the right value.
529         RESET
530             Plant2.NR.PhoControl := 0.0;
531         END # reset
532         CONTINUE FOR 1*60
533         DISPLAY Plant2.NR.THyd.QFission END
534         DISPLAY "17.1" END
535         CONTINUE FOR 800*60
536         SAVE PRESETS PRESETS_NuclearReactorOperating
537         SAVE INITIAL INITIAL_NuclearReactorOperating
538         # Insert Temperature Reactivity
539         RESET
540           Plant2.NR.wPhoExtra1:= 0;
```

```
541          Plant2.NR.PhoExtra2:=OLD(Plant2.NR.PhoExtra1);
542        END # reset
543        CONTINUE FOR 1*60
544        DISPLAY TIME END
545        RESET
546          Plant2.NR.PhoExtra2:=OLD(Plant2.NR.PhoExtra2); # trick to eliminate problems of phoextra2 jumping during
               transient
547        END
548        CONTINUE FOR 1*60
549        DISPLAY Plant.Loop1.F(1) END
550        DISPLAY Plant.Loop1.DeltaP END
551        # This is the new transient.  50\% decrease in helium mass in loop
552        RESET
553          Plant.PipeC2.P_In:= 1120950.0 - 120894.0*(TIME-OLD(TIME))/15;
554        END
555        CONTINUE FOR 15
556        RESET
557          Plant.PipeC2.P_In:= 1120950.0 - 120894.0;
558        END # reset
559        CONTINUE FOR 100*60
560        DISPLAY "This is QH", QH, "This is QHdiff", QHdiff END
561      END # sequence
562  END # simulation
563  ######################################################################
```

Bibliography

[1] Energy Information Administration. International energy outlook 2009. Technical report, U.S. Department of Energy, Washington, DC, 2009.

[2] R. Agrawal, N. R. Singh, F. H. Ribeiro, and W. N. Delgass. Sustainable fuel for the transportation sector. *Proceedings of the National Academy of Sciences*, 104(12):4828, 2007.

[3] U. M. Ascher and L. R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Society for Industrial & Applied Mathematics, Philadelphia, PA, USA, 1998.

[4] P. I. Barton. Dyanmic modeling and simulation notes. Class notes, M.I.T, 1997.

[5] P. I. Barton and C. C. Pantelides. Modeling of combined discrete/continuous processes. *AIChE Journal*, 40(6):966–979, 1994.

[6] BP. BP Statistical Review of World Energy June 2010. bp.com/statisticalreview.

[7] R. S. Cherry and R. A. Wood. Use of a nuclear high temperature gas reactor in a coal-to-liquids process. Technical Report INL/EXT-06-11667, Idaho National Laboratory, 2006.

[8] C. B. Davis, R. B. Barner, S. R. Sherman, and D. F. Wilson. Thermal-hydraulic analyses of heat transfer fluid requirements and characteristics for coupling a hydrogen product plant to a high-temperature nuclear reactor. Technical Report INL/EXT-05-00453, Idaho National Laboratory, 2005.

[9] I. S. Duff, A. M. Ersiman, and J. K. Reid. *Direct Methods for Sparse Matrices*. Oxford University Press, 1986.

[10] R. Elder and R. Allen. Nuclear heat for hydrogen production: Coupling a very high/high temperature reactor to a hydrogen production plant. *Progress in Nuclear Energy*, 51(3):500–525, 2009.

[11] A. E. Finan and A. C. Kadak. Integration of Nuclear Energy Into Oil Sands Projects. *Journal of Engineering for Gas Turbines and Power-Transactions of the ASME*, 132(4):042902–1–8, 2010.

[12] C. W. Forsberg. Nuclear energy for a low-carbon-dioxide-emission transportation system with liquid fuels. *Nuclear Technology*, 164(3):348–367, 2008.

[13] C. W. Forsberg. Sustainability by combining nuclear, fossil, and renewable energy sources. *Progress in nuclear energy*, 51(1):192–200, 2009.

[14] R. O. Gauntt. *MELCOR Computer Code Manuals, Vol. 1: Primer and Users' Giude Version 1.8.6*. Sandia National Laboratories, 2005.

[15] C. W. Gear. The automatic integration of ordinary differential equations. *Communications of the ACM*, 14(3):179, 1971.

[16] A. Griewank and A. Walther. *Evaluating derivatives: principles and techniques of algorithmic differentiation*. Society for Industrial and Applied Mathematics (SIAM), 2nd edition, 2008.

[17] M. Hanke, K. Henrik, A. Olsson, and M. Strömgren. Stability analysis of a degenerate hyperbolic system modelling a heat exchanger. *Mathematics and Computers in Simulation*, 74(1):8–19, 2007.

[18] E. A. Harvego, M. G. McKellar, J. E. O'Brien, and J. S. Herring. Parametric evaluation of large-scale high-temperature electrolysis hydrogen production using different advanced nuclear reactor heat sources. *Nuclear Engineering and Design*, 239:1571–1580, Sep 2009.

[19] E. A. Harvego, M. G. McKellar, M. S. Sohal, J. E. O'Brien, and J. S. Herring. Economic analysis of a nuclear reactor powered high-temperature electrolysis hydrogen production plant. In *ES2008: Proceedings of the 2nd International Conference on Energy Sustainability - 2008*, pages 549–558. Amer. Soc. Mechanical Engineers, 2009.

[20] A. Hauch, S. D. Ebbesen, S. H. Jensen, and M. Mogensen. Highly efficient high temperature electrolysis. *Journal of Materials Chemistry*, 18(20):2331–2340, May 2008.

[21] A. F. Henry. *Nuclear Reactor Analysis*. MIT Press, 1975.

[22] A. C. Kadak. A future for nuclear energy: pebble bed reactors. *Int. J. Critical Infrastructures*, 1(4):330–345, 2005.

[23] A. C. Kadak. MIT pebble bed reactor projects. *Nuclear Engineering and Technology*, 39:95–102, 2007.

[24] J. F. Kikstra and A. H. M. Verkooijen. Dynamic Modeling of a Cogenerating Nuclear Gas Turbine Plant - Part I: Modeling and Validation. *Journal of Engineering for Gas Turbines and Power*, 124:725, 2002.

[25] R. J. Leveque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, Cambridge, UK, 2002.

[26] Pebble Bed Modular Reactor Limited. Photo gallery—fuel composition. http://www.pbmr.co.za/index.asp?Content=213.

[27] W. Malalasekera and H. K. Versteeg. *An introduction to computational fluid dynamics*. Longman Scientific & Technical, 1995.

[28] J. J. Marano and J. P. Ciferno. Life-cycle greenhouse-gas emissions inventory for Fischer-Tropsch fuels. Technical report, Energy and Environmental Solutions, LLC for the U.S. Department of Energy, National Energy Laboratory, 2001.

[29] W. E. Martinson and P. I. Barton. A differentiation index for partial differential-algebraic equations. *SIAM Journal on Scientific Computing*, 21(6):2295–2315, 2000.

[30] S. E. Mattsson. On modelling and differential/algebraic systems. *Simulation*, 52(1):24, 1989.

[31] T. McCann and P. Magee. Crude oil greenhouse gas life cycle analysis helps assign values for CO_2 emissions trading. *Oil & Gas J.*, 97:38–44, Feb. 22 1999.

[32] M. G. McKellar, J. E. O'Brien, E. A. Harvego, and J. S. Herring. Optimized flow sheet for a reference commercial-scale nuclear-driven high-temperature electrolysis hydrogen production plant. Technical Report INL/EXT-07-13573, Idaho National Laboratory, November 2007.

[33] Special Metals. Incoloy alloy 800. http://www.specialmetals.com/products/incoloyalloy800h&ht.php, May 2010.

[34] National Energy Board (NEB). Canada's oil sands: Opportunities and challenges to 2015: An energy market assessment. Technical report, NEB, Calgary, Alta., May 2004.

[35] Numerica Technology, Cambridge, MA, U.S.A. *JACOBIAN User Guide*, 2006.

[36] J. O'Brien, C. Stoots, J. S. Herring, and G. L. Hawkes. Hydrogen production from nuclear energy via high temperature electrolysis. Technical Report INL/CON-06-01375, INL, 2006.

[37] U.S. Department of Energy Research Advisory Committee and the Generation IV International Forum. A technology roadmap for generation IV nuclear energy systems. Technical report, 2002.

[38] C. H. Oh, E. S. Kim, S. R. Sherman, and R. Vilim. HyPEP FY-07 Report: System Integration Model Development. Technical Report INL/EXT-07-12470, Idaho National Laboratory, 2007.

[39] S. V. Patankar. *Numerical heat transfer and fluid flow*. Taylor & Francis, 1980.

[40] A. Pothen and C. J. Fan. Computing the block triangular form of a sparse matrix. *ACM Trans. Math. Software*, 16(4):303–324, 1990.

[41] Process System Enterprise Ltd. *gPROMS Introductory User Guide*, 2004.

[42] M. Richards, A. Shenoy, K. Schultz, L. Brown, E. Harvego, M. McKellar, J. P. Coupey, S. M. M. Reza, F. Okamoto, and N. Handa. H2-MHR conceptual designs based on the sulphur-iodine process and high-temperature electrolysis. *International Journal of Nuclear Hydrogen Production and Applications*, 1(1):36–50, 2006.

[43] S. B. Rodriguez, R. O. Gauntt, R. Cole, F. Gelbard, K. Mcfadden, T. Drennen, B. Martin, D. Louie, L. Archuleta, M. El-Genk, J. Tournier, F. Espinoza, S.T. Revankar, and K. Vierow. Transient analysis of Sulfur-Iodine cycle experiments and very high temperature reactor simulations using MELCOR-H_2. *Nuclear Technology*, 166(1):76–85, Apr 2009.

[44] D. K. Ryland, H. Li, and R. R. Sadhankar. Electrolytic hydrogen generation using CANDU nuclear reactors. *International Journal of Energy Research*, 31(12):1142–1155, 2007.

[45] D. Saphier. HTGR Transient Analysis with the DSNP Simulation Language. Technical Report RASG-111-84, Soreq Nuclear Research Center, Israel, December 1984.

[46] M.G. Savage. A one-dimensional modeling of radial heat removal during depressurized heatup transients in modular pebble-bed and prismatic high temperature gas-cooled reactors. Technical Report ORNL/TM-9215, Oak Ridge National Laboratory, July 1984.

[47] W. E. Schiesser. *The numerical method of lines: integration of partial differential equations*. Academic Press San Diego, 1991.

[48] M. Schipper. Energy-related carbon dioxide emissions in U. S. manufacturing. Technical Report DOE/EIA-0573, Energy Information Administration, 2006.

[49] D. Sedes. Modelling, simulation and process safety analysis. A case study: The formaldehyde process. Technical report, Massachusetts Institute of Technology, 1994.

[50] P. L. Spath and M. K. Mann. Life cycle assessment of hydrogen production via natural gas steam reforming. Technical report, National Renewable Energy Laboratory, 2001.

[51] C. Stoots, J. O'Brien, S. Herring, K. Condie, L. Moore-McAteer, J. J. Hartvigsen, and D. Larsen. High Temperature Solid-Oxide Electrolyzer 2500 Hour Test Results At The Idaho National Laboratory. In *2009 AIChE Annual Meeting, Conference Proceedings*. INL/CON-09-16888, Idaho National Laboratory (INL), 2009.

[52] The RELAP5-3D Code Development Team. *RELAP-3D Code Manual*. Idaho National Laboratory, Idaho Falls, Idaho, U.S.A., 2005.

[53] Aspen Tech. *Aspen Custom Modeler User Manuals*, 2005.

[54] J. E. Tolsma and P. I. Barton. On computational differentiation. *Computers & Chemical Engineering*, 22(4/5):475–490, 1998.

[55] U.S. Environmental Protection Agency, Office of Air and Radiation. Technical support document for hydrogen production: Proposed rule for mandatory reporting of greenhouse gases. Technical report, 2008.

[56] U.S. Nuclear Regulatory Commission, Washington, DC 20555-0001. *Nuclear Power Plant Licensing Process*, nureg/br-0298, rev. 2 edition, July 2004.

[57] K. Vierow, Y. Liao, J. Johnson, M. Kenton, and R. Gauntt. Severe accident analysis of a PWR station blackout with the MELCOR, MAAP4 and SCDAP/RELAP5 codes. *Nuclear Engineering and Design*, 234(1-3):129–145, 2004.

[58] R. B. Vilim. Dynamic modeling efforts for system interface studies. Technical Report ANL-07/16, Argonne National Laboratory, December 2006.

[59] R. B. Vilim. Initial Assessment of the Operability of the VHTR-HTSE Nuclear-Hydrogen Plant. Technical Report ANL-08/01, Nuclear Engineering Division, Argonne National Laboratory, September 2007.

[60] C. Wang. *Design, Analysis and Optimization of the Power Conversion System for the Modular Pebble Bed Reactor Systems*. Ph.D. Thesis, Massachusetts Institute of Technology, 2003.

[61] M. Wang, M. Wu, and H. Huo. Life-cycle energy and greenhouse gas emission impacts of different corn ethanol plant types. *Environmental Research Letters*, 2:024001, 2007.

[62] M. Q. Wang. Estimation of energy efficiencies of U. S. petroleum refineries. Technical report, Argonne National Laboratory, Argonne, IL, 2008.

[63] B. Yildiz and M. S. Kazimi. Efficiency of hydrogen production systems using alternative nuclear energy technologies. *International Journal of Hydrogen Energy*, 31:77–92, 2006.

[64] B. Yildiz, M. C. Petri, G. Conzelmann, and C. W. Forsberg. Configuration and technology implications of potential nuclear hydrogen system applications. Technical Report ANL-05/30, Argonne National Laboratory, 2005.

[65] Z. Zhang, Z. Wu, D. Wang, Y. Xu, Y. Sun, F. Li, and Y. Dong. Current status and technical description of Chinese 2 x 250 MWth HTR-PM demonstration plant. *Nuclear Engineering and Design*, 239(7):1212–1219, Jul 2009.