# DEVELOPMENT OF A MINIATURE NEUTRON SOURCE REACTOR SIMULATOR FOR TRAINING AND EDUCATION

**DIOP-FRIMPONG, DAVID DEMBA**

**DEPARTMENT OF NUCLEAR ENGINEERING**
**SCHOOL OF NUCLEAR AND ALLIED SCIENCES (SNAS)**
**UNIVERSITY OF GHANA**

**MPhil**

**2020**

# Development Of A Miniature Neutron Source Reactor Simulator For Training And Education

This thesis was submitted to the

DEPARTMENT OF NUCLEAR ENGINEERING,
SCHOOL OF NUCLEAR AND ALLIED SCIENCES
UNIVERSITY OF GHANA

By

Diop-Frimpong, David Demba

(10373261)

In partial fulfilment of the requirements for the degree of

MASTERS OF PHILOSOPHY

in

COMPUTATIONAL NUCLEAR SCIENCE AND ENGINEERING

October, 2020

## DECLARATION

This thesis is the result of research work undertaken by David Demba Diop-Frimpong in the Department of Nuclear Engineering, School of Nuclear and Allied Sciences, University of Ghana, under the supervision of Dr. Seth Kofi Debrah and Mr. Charles Nartey.

Sign..... Date..16/06/2021.....

**David Demba Diop-Frimpong**

Student

Sign..... Date..16/06/2021.....

**Seth Kofi Debrah, PhD.**

Principal Supervisor

Sign..... Date..16/06/2021.....

**Mr. Charles Nartey**

Co-Supervisor

**ABSTRACT**

This research project is aimed at developing a virtual dynamic simulator for a Miniature Neutron Source Reactor with GHARR-1 as a model. The GHARR-1 core which is surrounded above, beneath and around by metallic beryllium reflectors has undergone core conversion and now has a LEU core. The simulator modelled the reactor's self-limiting transient behavior as well as the reaction to the control rod movement, thermal reactivity feedback and fission product poison build up during the normal reactor operation. Using the point kinetic model of neutronics, a simplified model of fission product poisoning and lumped parameter modelling of thermal hydraulic exchanges between different zones of the reactor, ordinary differential equations were developed for the time gradients of reactor power and the temperatures of the different reactor zones. These equations were solved with LabVIEW 2019 Control and Simulation Toolkit, using its Runge-Kutta 45 ordinary differential equation solver. The reactor control panel, built with the LabVIEW 2019 graphical user interface, allows the user to vary control rod position, and configure the differential equation solver parameter. Outputs of reactor inlet and outlet temperatures, fuel clad temperature and reactor power are displayed in simulation time and can be retrieved into a file. For simulation of reactor presets after control rod full insertion, the simulator depicted similar trends in output to those obtained by experiment. These values were however overestimated by the simulator by varying levels of deviation. The simulator was capable of qualitatively predicting the power, and temperature trends of the GHARR-1 at 17 kW and 34 kW.

**DEDICATION**

This work is dedicated to my lovely family, the Diop-Frimpong family.

**ACKNOWLEDGEMENTS**

With deep gratitude, I acknowledge the contributions of Dr. Seth Debrah and Mr. Charles Nartey, my supervisors for their guidance through this research.

I also wish to acknowledge Selorm Dzide of the Nuclear Regulatory Authority for his aid in the building of this simulator.

I am thankful to my family and friends for their emotional and financial assistance through this journey.

## TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

# NOMENCLATURE

| Symbol | | Units |
|---|---|---|
| **Roman Alphabet** | | |
| $T_{fm}$ | Fuel meat temperature | [°C] |
| $T_c$ | Cladding Temperature | [°C] |
| $T_1$ | Reactor inlet Temperature | [°C] |
| $T_2$ | Reactor outlet Temperature | [°C] |
| $T_{upper}$ | Temperature of water above the core | [°C] |
| $T_{dc}$ | Temperature of water in the down-comer region | [°C] |
| $T_f$ | Average coolant temperature | [°C] |
| $T_{pool}$ | Pool water temperature | [°C] |
| $T_{Be}$ | Annular beryllium shield temperature | [°C] |
| $A_s$ | Area of core heated surface area | $[mm^2]$ |
| $A_{up}$ | Cross-sectional area of flow of water above the core | $[mm^2]$ |
| $A_{dc}$ | Cross-sectional area of flow of water in down-comer | $[mm^2]$ |
| $C_{pfm}$ | Fuel meat specific heat capacity | $[J.kg^{-1}.°C^{-1}]$ |
| $C_{pc}$ | Cladding specific heat capacity | $[J.kg^{-1}.°C^{-1}]$ |
| $C_{pu}$ | Specific heat capacity of water above the core | $[J.kg^{-1}.°C^{-1}]$ |
| $C_{pdc}$ | Specific heat capacity of water in the down-comer region | $[J.kg^{-1}.°C^{-1}]$ |
| $C_{pf}$ | Average moderator specific heat capacity | $[J.kg^{-1}.°C^{-1}]$ |
| $e\ (t)$ | error between process variable and set-point variable | [_] |
| $D$ | Diffusion Coefficient | $[mm^2s^{-1}]$ |
| $J$ | Neutron current density | $[mm^{-2}.s^{-2}]$ |

| | | |
|---|---|---|
| $P_u$ | Period taken from limit cycle | $[s]$ |
| $K_u$ | Ultimate controller gain | [_] |

**Greek Symbols**

| | | |
|---|---|---|
| $\rho$ | Reactivity of nuclear reactor | [pcm] |
| $\beta$ | Delayed neutron fraction | [_] |
| $k_{eff}$ | Effective Multiplication Factor | [_] |
| $\phi$ | Neutron flux | $[mm^{-2}.s^{-2}]$ |
| $\Lambda$ | Mean neutron lifetime | $[s]$ |
| $\lambda$ | Decay constant of delayed neutron precursor | $[s^{-}]$ |
| $\Sigma_a$ | Macroscopic neutron absorption cross section | $[mm^{-1}]$ |
| $\sigma$ | Microscopic neutron interaction cross section | $[barn]$ |

**Abbreviations**

| | |
|---|---|
| LabVIEW | Laboratory Virtual Instrument Engineering Workshop |
| PID | Proportional, Integral, Derivative |
| VI | Virtual Instrument |
| GHARR-1 | Ghana Research Reactor 1 |
| MNSR | Miniature Neutron Source Reactor |
| LEU | Low Enriched Uranium |
| NAA | Neutron Activation Analysis |
| AC | Alternating Current |
| DAQ | Data Acquisition Device |
| I/O | Input/Output |
| GPIB | General Purpose Interface Bus |

| | |
|---|---|
| SP | Set Point |
| PV | Process Variable |

# CHAPTER 1:INTRODUCTION

## 1.1 Background

Scientists and engineers in order to understand and imitate nature, run experiments to check for the reaction of various elements to certain stimuli. Experiments ranging from Galileo demonstrating that objects fall at the same speed, Isaac Newton splitting white light into colors (Fara, 2015), Ernest Rutherford and associated splitting the atom (Gosling, 1999) and Enrico Fermi demonstrating the nuclear chain reaction have helped shape the way scientists and engineers view and translate the world (Nesvizhevsky & Villain, 2017).

Over the last fifty years, there has been a revolutionary development affecting almost all aspects of science. This development is the astonishing growth in the use of digital computer to study phenomena of great complexities in the sciences - the rise of computer simulations. (Winsberg, 2010).

## 1.2 Overview and Importance of Simulation

A simulation is the imitation of the operation of real-world process or system over time. The overall approach in computer simulation is to represent the dynamic characteristics of a real-world system in a computer model. The model is then used to predict the effect of certain stimuli on the model which is useful in making informed decision on the real system. Simulations are used for problems in which analytical methods present no solutions. Since most dynamic problems in practice cannot be represented and solved fully

using mathematical equations, computer simulation is a powerful and flexible methodology in complex systems analysis (Kalbasi et al., 2015).

The need for simulators in the world today cannot be overemphasized. Especially in the world of training, simulator is the preferred tool. The utility power industry, specifically the nuclear industry, has been using training simulators for many years. Specific processes which make significant use of simulators include offshore and onshore production facilities, ethylene plants, catalytic cracking units and ammonia plants (Jones & Limited, 1992). The use of simulators allows for an almost perfect representation of real-life events without the risk of failure of equipment or injury.

### 1.3 Overview and Importance of Control Systems

In order to acquire simulators that closely mimic real-life machinery or events, control systems are used to guide certain aspects of the simulator. A control system is a system of devices or set of devices, that manages, commands, directs or regulates the behavior of other devices or systems to achieve desired results (Electrical4U, 2019). In the case of simulators, controllers can be used to closely represent the outcome of events. Controllers would allow a simulation to be governed more by automation and less by human intervention. In the case of a reactor simulation, a control system can be used to control the movement of control rods based on the neutron flux in the reactor.

Building an efficient control system in a simulator can allow for precise results of a simulation as well as broadening the scope of possibilities of the simulator. When

considering the configuration of a control system there are two options available. These are,

- open-loop system and
- closed-loop system.

Figure 1.1 depicts the different components of a control system in a close-loop configuration. This is the most extended and useful configuration, as well as the most obvious way, to control the output of a given plant. This is because of the continuous feedback provided by the sensorial system that permits the controller to update its actions. However open-loop configuration (see Figure 1.2) can be also found in certain simple and affordable control systems.

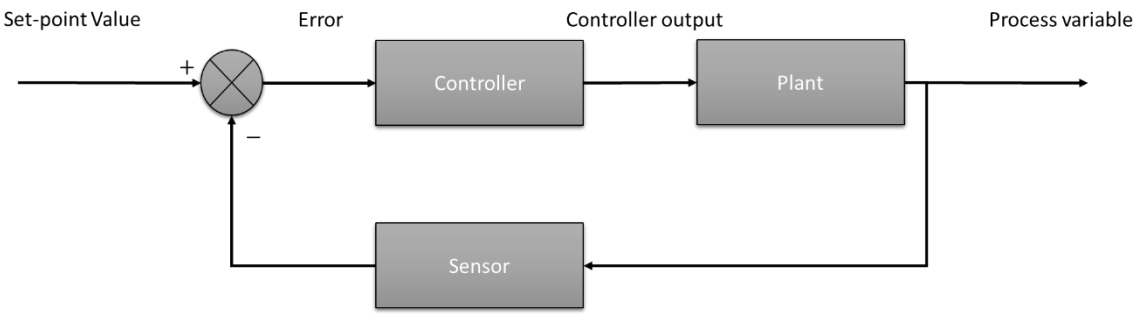


Figure 1.1: The Closed-Loop configuration

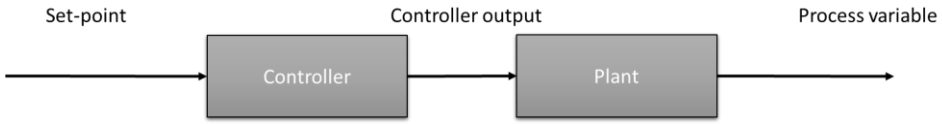


Figure 1.2: Open-Loop Configuration

### 1.3.1 Overview and Importance of PID

PID control is used to control and maintain processes. It can be used to control physical variables such as temperature, pressure, flow rate, and liquid tank level. The original technology for industrial Proportional, Integral, and Derivative (PID) control was pneumatic, hydraulic, or mechanical and the controller usually had a simple interface for manual tuning. The first theoretical analysis of a PID controller dates back to 1922 when the Russian-American engineer, Nicolas Minorsky developed an automatic ship steering system for the U.S. Navy. This was based on observations of the steersmen's use of current error, past error, and rate of change to keep the ship on course. Controllers with electrical systems were developed after World War II (Ho, 2014).

The technique is widely used in today's manufacturing industry to achieve accurate process control under different process conditions. A Process Variable (PV) temperature, for example, is measured, and a feedback signal is sent to the controller. The controller then compares the feedback signal to the set point (SP) and generates an error value. The value is examined with one or more of the three proportional, integral, and derivative methodologies. As a result, the controller issues the necessary commands or alters the control variable (CV) to correct the error (E). These procedures form an iterative process.

## 1.4 Description of GHARR-1 Control Rod Mechanism

The Ghana Research Reactor 1 (GHARR-1) is a Miniature Neutron Source Reactor (MNSR), a tank in pool type reactor. It has a thermal power of 34kW, uses Low Enriched Uranium (LEU) of 13% $UO_2$ as fuel and light water as both a moderator and coolant. It is used for Neutron Activation Analysis (NAA), education and training (Odoi et al., 2016).

In the GHARR-1, reactivity can be controlled by either raising the control rod, to increase reactivity or lowering it to decrease reactivity. The control rod operates via an electromechanical drive control mechanism, consisting of AC servomotor, an electromagnetic clutch, a gearbox, a winch drum linked with a steel rope and an autosyn.

Upon the application of power to the motor, and the activation of the "protection ready" mode on the control console, the electromagnetic clutch is closed, allowing the motor to drive the gear which in turn operates the winch drum, to raise or lower the control rod via the steel rope.

## 1.5 Overview of LabVIEW

LabVIEW is a graphical programming language that uses icons and other graphical objects in a drag and drop environment instead of the text-based environment used in traditional programming languages. LabVIEW employs dataflow programming, where data flows through nodes, a node in this case being a block of code represented in LabVIEW as an icon. LabVIEW uses two windows: the front panel and the block panel diagram panel.

LabVIEW 2009 has several features that make it amenable to systems modelling and simulation. LabVIEW 2009 can be used to model systems of linear ordinary differential equations (ODEs), representing the neutron point kinetics and thermal hydraulic models. It allows for the use of a wide range of numerical differential equation solvers, within a simulation loop VI of the Control Design and Simulation Toolkit. It has been proven to be suitable for programming in the fields of Nuclear Instrumentation and Simulation.

The Control Design and Simulation Toolkit consist of sets of VIs, namely the Control Design VIs, the Simulation VIs, the Optimal Design VIs and the Simulation Converter. They can be used for the design, analysis and deployment of controllers for dynamic system models; building, simulating and deploying dynamic models; configuration of simulation parameters; determining optimal parameters for dynamic models and also for the conversion of Math Works SIMULINK models to LabVIEW block diagrams. The integrators used in the control and simulation loops can implement one of a variety of inbuilt ODE solvers. These solvers, including the Runge-Kutta 45 and 23, compute the ODEs numerically using either fixed step solvers or variable step solvers.

## 1.6 Statement of Research Problems

Although small thermal power research reactor simulators are needed for research and training, there is limited research on small power reactor research simulators. Previously, a research project was aimed at developing a PC- based dynamic simulator for Ghana Research Reactor 1 (GHARR-1) (Dzide, 2010). A simulator was developed to model the reactor's self-limiting transient behavior as well as its reaction to control rod movement, insertion of thermal reactivity feedback and fission product poison build up during the reactor operation using the point kinetic model of neutronics. However, this previous work involved a manual system of the control of the control rod motion. Thus, limiting the scope of the simulator.

## 1.7 Objectives

The main objective of this work was to design and develop a desktop simulator of GHARR-1 for training which can simulate the automatic control of the reactor, the transient behavior due to thermal reactivity feedback as well as fission product poison build-up using LabVIEW.

### 1.7.1 Specific Objectives

To achieve the main objective, the following objectives where employed:

- Rebuild an earlier simulator using new conditions of the Low Enriched Uranium core.
- Automate the control rod motion using PID and data from the GHARR-1 research reactor
- Auto tune the PID to enable the control system perform optimally for GHARR-1

## 1.8 Justification

This research would lead to the improvement of an earlier GHARR-1 simulator and thus create a teaching apparatus for simulating Miniature Neutron Source Reactor (MNSR) nuclear research reactors.

This work would also improve on the model used in the GHARR-1 simulator by improving the control unit. It will also open the door to adding complexities to the control system such

as multiple feedback loops and intelligent control units, improving the efficiency of the model as a whole.

## 1.9 Organization of Thesis

This thesis was divided into 5 Chapters. The first presents the background, justification and objectives of the work. It focuses on the overview of topics significant to the work such as the Simulators, Ghana Research Reactor 1, LabVIEW and Control Systems. The chapter then goes ahead to justify the need of such work to be undertaken. Overall and specific objectives were presented in this chapter.

The second chapter is a literature review, which analytically discusses the similar simulations and designs of control mechanisms control rod motion. This chapter also looks at the underlying equations for the various components of the research and the relationship between these components.

The third chapter, titled “Methodology” describes the models used in the simulation. In this chapter is a breakdown of the underlying equations used in the research. It also has LabVIEW based algorithms and discusses the method of verification and validation. The use of LabView Virtual Instruments (VIs) was expanded upon in this chapter.

Chapter 4 contains results of simulations involving control rod motion and power presets. It also contains discussion of the results, their significance, as well as their accuracy and associated results.

Chapter 5 states the main conclusion of the research project as well as recommendations for the future related work.

# CHAPTER 2:LITERATURE REVIEW

## 2.1 Nuclear Reactor Theory

### 2.1.1 Miniature Neutron Source Reactors (MNSRs)

In view of the problems faced by large and medium research reactors, the prototype of Miniature Neutron Source Reactor was developed and constructed in 1984. After 2 years operation, the MNSR has demonstrated its high reliability, unique safety, facilitation of operation, tiny radiation effect, and cost attraction (Yongmao et al., 1987). This reactor has a maximum thermal neutron flux of $1 \times 10^{12}\ n.cm^{-2}.s^{-1}$ in one of its inner irradiation channels and thermal power of approximately 30kW. The MNSR, designed based on the Canadian SLOWPOKE reactor is one of the smallest commercial research reactors available. Its commercial versions currently in operation in China, Ghana, Iran, Nigeria, Pakistan and Syria, is considered as an excellent tool for Neutron Activation Analysis (NAA), training of Scientist, and Engineers in nuclear science and technology and small scale radioisotope production (Ahmed et al., 2006).

The nuclear fission reaction in GHARR-1 occurs by splitting the heavy U-235 nuclides to which in turn releases energy, through neutron bombardment. The neutrons released from said bombardment then interact with other elements in the reactor, namely the coolant which in GHARR-1 is light water ($H_2O$). Some of these neutrons undergo thermalization after being a round of bombardment in the nuclear reactor.

The reactivity of a nuclear reactor is defined as a measure of its departure from criticality. It is given by the formula below:

$$\rho = \frac{k_{eff}-1}{k_{eff}} \tag{2-1}$$

Where $k_{eff} = the\ effective\ multiplication\ factor\ of\ the\ reactor$

The reactivity, which tells the measure of the state of a reactor in relation to where it would be if it were in a critical state, can be calculated by combining reactivity feedback from the control rod, temperature changes of the coolant and poisoning of xenon and samarium.

$$\rho_{total} = \rho_o + \rho_{poison} + \rho_{thermal} + \rho_{controlRod} \tag{2-2}$$

Where $\rho_o$ represents the excess reactivity.

### 2.1.2 Point Kinetic Equation

The point kinetics model is a simplification of reactor dynamics, which treats the whole reactor core as a point source. It is widely used because of the apparent simplicity of the resulting derived equations. It therefore provides solutions that are representative of the average output parameters of the reactor and their variation with time, while providing little information of spatial variation within the reactor (Dzide, 2010).

It is generally expressed in the form:

$$\frac{1}{v_g}\frac{\partial \phi_g(\vec{r},t)}{\partial t} = -\nabla.\vec{J}(\vec{r},t) - \sum_{g'=1}^{g}\Sigma_{sgg'}\phi_{g'} + \chi_g^p(1-\beta)\sum_{g'=1}^{g} v\Sigma_{fgg'}\phi_{g'} + \sum_{i=1}^{D}\chi_{ig}^d\lambda C_i(\vec{r},t) \tag{2-3}$$

$Where\ \phi = neutron\ flux$

$g = Energy\ group$

$J = neutron\ current\ density$

$v = neutron\ speed$

$\chi_g^d = Delayed\ neutron\ spectra$

$\chi_g^p = Photoneutron\ spectra$

$\Sigma_{sgg'} = Macroscopic\ neutron\ scattering\ crossection\ of\ group\ g'$

$\Sigma_{fgg'} = Macroscopic\ fission\ crossection\ of\ group\ g'$

$\beta = Delayed\ neutron\ fraction$

$\lambda = Decay\ constant\ of\ delayed\ neutron\ precursor$

$C = Concentration\ of\ delayed\ neutron\ precursor$

Neutrons produced by the fission process are born at different energies. These energies are classified into a number of groups each representing a specific energy range. For this reason, Equation 2-3 solves for the rate of change of neutron flux by summing up all neutron flux terms over all specified neutron groups.

The point kinetic model, as employed in this work, makes the approximation of assuming the neutrons emitted for inducing fission are mono-energetic, some of which are delayed and photo-neutrons. Those neutrons produced directly from fission are termed prompt neutron while those produced later are generally classified as delayed neutrons.

The relation below gives the rate of change of neutron density with time, within a nuclear reactor system.

$$\frac{\partial n}{\partial t} = \{Rate\ of\ production\ from\ Source\} - \{Rate\ of\ loss\ by\ absorption\} - \{Rate\ of\ loss\ by\ leakage\} \quad (2\text{-}4)$$

In equation (2-5) below, the various terms in (2-4) are expressed in the same order.

$$\frac{\partial n}{\partial t} = S(r,t) - \Sigma_a \phi(r,t) - \nabla . J \quad (2\text{-}5)$$

$\Sigma_a = macroscopic\ neutron\ absorption\ cross\ section$

$\phi(r,t) = neutron\ flux,$

$J = neutron\ current\ density$

The first term represents a source term of neutrons while the remaining two represent neutron losses by absorption and leakage respectively.

According to Fick's Law (Stacey, 2007),

$$J = -D . \nabla \phi(r,t) \quad (2\text{-}6)$$

Where

$D = diffusion\ coefficient$

From (2-4), (2-5) and (2-6),

$$\frac{\partial n}{\partial t} = S(r,t) + D . \nabla^2 \phi(r,t) - \Sigma_a \phi(r,t) \quad (2\text{-}7)$$

Choosing $\beta$ = fraction of delayed neutrons, then $(1-\beta)$ = fraction of prompt neutrons.

$$\Rightarrow S(r,t) = k_\infty \Sigma_a \phi (1-\beta) \quad (2\text{-}8)$$

$where\ k_\infty = multiplication\ factor$

The term $k_\infty$ represents the number of neutrons produced per generation of absorbed neutrons. Therefore, $S(r,t)$ represents the production rate of prompt neutrons.

$$\text{Diffusion length (Stacey, 2007), L} = \sqrt{\frac{\mathrm{D}}{\Sigma_\mathrm{a}}} \quad (2\text{-}9)$$

$$\phi = \mathrm{n.v} \quad (2\text{-}10)$$

Where $v$=average neutron velocity.

The neutron flux term in (2-7) can be expressed in terms of a constant known as the geometric buckling factor, $B_g$ (Stacey, 2007). $B_g$ is specific the geometry of the reactor medium.

$$\nabla^2\phi = -\mathrm{B_g}\phi \tag{2-11}$$

Delayed neutrons are produced from the decay of fission products. Some fission products also undergo beta decay and hence release gamma radiation. These gamma photons, incident on the beryllium shims in turn eject neutrons from the beryllium nuclei these delayed neutrons are referred to as photo-neutrons (Muhammad et al., 2008). The presence of the beryllium reflector in the reactor therefore compensates for the small critical mass of the MNSR.

From equations (2-8), (2-9), (2-10), (2-11) equation (2-7) can be expressed as a simplification of (2-3), in the form below which accounts for all the delayed neutrons.

$$\frac{\partial \mathrm{n}}{\partial \mathrm{t}} = \mathrm{k_\infty}\Sigma_\mathrm{a}\phi(1-\beta) - \mathrm{L^2}\Sigma_\mathrm{a}\mathrm{B_g}\mathrm{nv} - \Sigma_\mathrm{a}\mathrm{nv} + \left\{\begin{matrix}\text{Rate of production}\\ \text{of delayed and}\\ \text{photo neutrons}\end{matrix}\right\} \tag{2-12}$$

The rate of production of delayed neutrons can be represented by the decay of several groups of precursors from fission fragments. Likewise, the production of photo neutrons can be represented by the decay of photo-neutrons precursors.

$C_i^d = ith\ group\ of\ delayed\ neutrons, C_j^p = jth\ group\ of\ delayed\ neutrons$

$$\left\{\begin{matrix}\text{Rate of change}\\ \text{of delayed neutron}\\ \text{precursors}\end{matrix}\right\} = \left\{\begin{matrix}\text{Rate of production}\\ \text{of delayed neutron}\\ \text{precursors}\end{matrix}\right\} - \left\{\begin{matrix}\text{Rate of decay}\\ \text{of delayed}\\ \text{neutrons}\\ \text{precursors}\end{matrix}\right\} \tag{2-13}$$

$$\frac{\mathrm{dC_i^d}}{\mathrm{dt}} = \gamma^\mathrm{d}\beta_\mathrm{i}^\mathrm{d}\mathrm{k_\infty}\Sigma_\mathrm{a}\phi - \lambda_\mathrm{i}^\mathrm{d}\mathrm{C_i^d} \tag{2-14}$$

Similarly,

$$\frac{dC_j^p}{dt} = \gamma^p \beta_j^p k_\infty \Sigma_a \phi - \lambda_j^p C_j^p \quad (2\text{-}15)$$

The following four relationships (Lewis, 2008) can be used to further simplify (2-11)

$$\text{Prompt neutron lifetime}, l = \frac{1}{v\Sigma_a(1+L^2B_g^2)} \quad (2\text{-}16)$$

$$\text{Neutron Non Leakage Probability}, P_{NL} = \frac{1}{1+L^2B_g^2} \quad (2\text{-}17)$$

$$\text{Effective multiplication factor}, k_{eff} = k_\infty P_{NL} \quad (2\text{-}18)$$

$$\text{Mean neutron generation time}, \Lambda = \frac{1}{k_{eff}} \quad (2\text{-}19)$$

After substituting the RHS for equations (2-1), (2-16), (2-17), (2-18) and (2-19) into (2-11) and upon further simplification the following equation results:

$$\frac{\partial n}{\partial t} = \frac{\rho-\beta}{\Lambda}.n + \sum_{i=1}^{6} \lambda_i^d C_i^d + \sum_{j=1}^{9} \lambda_j^p C_j^p \quad (2\text{-}20)$$

### 2.1.3 Fission Product Poisoning

Xenon-135 is a both a direct fission product as well as a product of the decay of other fission products. It has such a large neutron absorption cross-section ($\sim 2.65 \times 10^6\ barns$) (Lewis, 2008) that small amounts in the core might significantly affect the $k_{eff}$ by absorbing neutrons needed to sustain the fission chain reactions.

In the equations to follow in this section,

$\Sigma_f = Macroscopic\ fission\ cross\ section\ of\ fuel$

$\sigma_a^Y = Microscopic\ absorption\ cross\ section\ of\ element\ Y$

$\lambda_Y = Decay\ constant\ of\ element\ Y$

$\gamma_Y = Fission\ Yield\ of\ element\ Y$

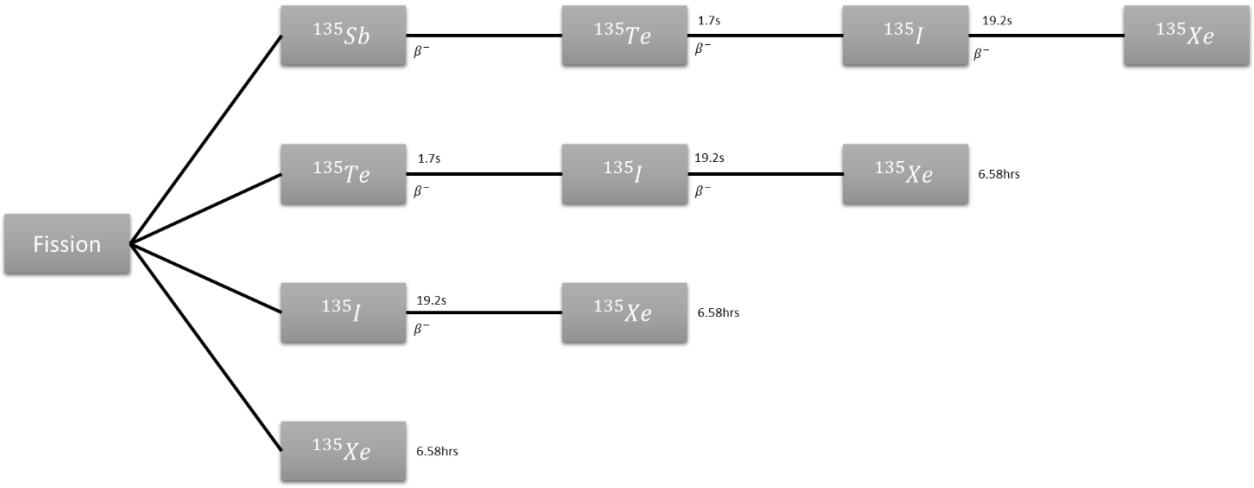


Figure 2.1: Xenon-135 Production Chain

Concentrations of Xenon -135 can be found from solutions to the following equations.

$$\frac{\partial \mathrm{I}}{\partial \mathrm{t}} = \gamma_{\mathrm{f}}\Sigma_{\mathrm{f}}\phi - \lambda_{\mathrm{I}}\mathrm{I} \qquad (2\text{-}21)$$

$$\frac{\partial \mathrm{Xe}}{\partial \mathrm{t}} = \gamma_{\mathrm{Xe}}\Sigma_{\mathrm{f}}\phi + \lambda_{\mathrm{I}}\mathrm{I} - \lambda_{\mathrm{Xe}}\mathrm{Xe} - \sigma_{\mathrm{a}}^{\mathrm{Xe}}\phi\mathrm{Xe} \qquad (2\text{-}22)$$

Samarium-149 is a product of the beta decay of Promethium-149. Promethium-149 is formed from the beta decay of a U-235 fission fragment, Neodymium-149. Neodymium-149 however has a half-life of 1.7 hours, which is much smaller than that of Promethium-149, which is 53 hours. The Promethium-149 can therefore be assumed to be a direct product of the fission process. $^{149}Sm$ has a neutron absorption cross-section of $4 \times 10^4$ barns (Lewis, 2008).

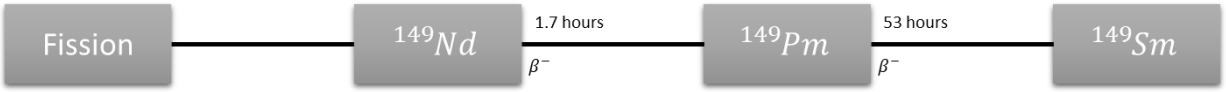


Figure 2.2 :Samarium -149 Production Chain

The following differential equations describe the rate of production of Samarium-149.

$$\frac{\partial \mathrm{Pm}}{\partial \mathrm{t}} = \gamma_{\mathrm{Pm}} \Sigma_{\mathrm{f}} \phi - \lambda_{\mathrm{Pm}} \mathrm{Pm} \tag{2-23}$$

$$\frac{\partial \mathrm{Sm}}{\partial \mathrm{t}} = \lambda_{\mathrm{Pm}} \mathrm{Pm} - \mathrm{Pm}.\, \sigma_{\mathrm{a}}^{\mathrm{Sm}}.\, \phi \tag{2-24}$$

In general the reactivity contributions of the fission product poison, Y can be proven to be of the form (US Department of Energy, 1993),

$$\rho(\mathrm{Y}_{\mathrm{concentration}}) = -\frac{\mathrm{Y}_{\mathrm{concentration}} \cdot \sigma_{\mathrm{a}}^{\mathrm{Y}}}{\Sigma_{\mathrm{a}}^{\mathrm{f}}} \tag{2-25}$$

## 2.2 Controller Theory

This glue stitches other fields of engineering. Electrical engineers designing switching power regulators for electrical devices, require feedback, and can be unstable if designed incorrectly. Communication engineers building an automatic gain circuit which automatically increases the gain in weak signals and decreases the gain in strong signals. Mechanical engineers concerned with vibration and dampening in a structure designing an isolation system in a motor mount for a system sensitive to vibrations. Civil engineer building an active or passive dampening system for tall buildings in an earthquake zone where you could design a robotic assembly line or tune PID controller gains that are ubiquitous in the industrial robotic applications. Aerospace engineer solving aircraft flutter which occurs when there is an interaction between the stiffness of a structure and its aerodynamics.

At this point it is evident control systems are more than just tuning PID controllers. It's the building models of any system and then simulating to make predictions by filtering out noise and rejecting outside disturbances selecting proper sensors and actuators, testing your system to ensure it performs as expected.

### 2.2.1 Open and Closed Loop Systems

A control system is a mechanism by which the future behavior or state of a system can be altered. The altered state tends to a desired state. Control theory is a branch of mathematics with processes of attaining said outcomes. All control units have two basic parts, namely the plant or system to be controlled and an input into the plant. The input produces an output when it works on the plant forming an open loop control system. These are used for simple processes that have well defined input output system. The drawback is the fact that the input has no way of adjusting to gain a more efficient output therefore creating an inefficient system. If the input is fed with a signal with a relation to the output the system is referred to as a closed loop system.

### 2.2.2 Time and Frequency Domain

When a spring connected to a fixed point on one side and a mass on the other side is set in motion by an impulse corresponding to instantaneous velocity, the resulting movement over time resembles a sinusoid that can be described in the time domain using newtons second law of motion.

$$m\ddot{x} = -kx \tag{2-26}$$

The general solution of this differential equation is a sinusoid in the form:

$$y = Asin(\omega t + \phi) \tag{2-27}$$

This is the description of the system in the time domain. Describing this in the frequency domain is however simply with there being a single peak of amplitude $A$ at frequency $2\pi\omega$.

Representing complex sinusoidal curves, for instance a curve of superimposed sinusoidal curves of different frequency, in the frequency domain is not as simple. The frequencies of both the superimposed waves must be taken into account as well as their corresponding amplitude.

A mathematician named Joseph Fourier in 1807 published an equation that stated, if you have a signal that repeats over a period $T$, or the frequency of the repeating pattern is $\frac{1}{T}$, this time domain signal can be represented by an infinite summation of sinusoids at ever increasing frequencies (Asher, 2013). This Fourier series is as follows:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty}\{a_n\, cos(nx) + b_n\, sin(nx)\} \qquad (2\text{-}28)$$

Where the Fourier coefficients $a_0, a_n, b_n$ are defined by the integrals

$$a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)dx$$

$$a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos(nx)\,dx$$

$$b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin(nx)\,dx$$

In this form each sinusoid must be a multiple of the lowest frequency, the first harmonic. If the period, $T$ of the repeating signal increases the first harmonic frequency would decrease and therefore the discrete signals in the time domain increase in number. Taking the limit as the period approaches infinity, making it a non-repeating signal, the first harmonic frequency would approach zero. Now every frequency is possible. Thus, turning the discrete Fourier summation into a continuous Fourier integral called the Fourier Transform. This is able to represent any signal, repeating or not, as a summation of

sinusoidal signals that includes frequencies amplitudes and phases. This is great for understanding the frequency content of signals. The Fourier transform is as follows:

$$F(\omega) = \int_{-\infty}^{+\infty} f(t)e^{-j\omega t}dt \quad (2\text{-}29)$$

Note that according the Euler's formula, when $e$ is raised to an imaginary number, in this case $-j\omega t$, you get $\cos(\omega t) - jsin(\omega t)$.

To solve differential equations, the Fourier Transform needs to be tweaked. Firstly, making the transform causal by limiting its range to exclude negative time. Adding an exponential content to simulate the decay of signals. Changing these would change the Fourier Transform to a Laplace Transform.

$$\mathcal{L}\big(f(t)\big) = \int_0^{\infty} f(t)e^{-\sigma t}e^{-j\omega t}dt$$

$$\mathcal{L}\big(f(t)\big) = \int_0^{\infty} f(t)e^{-(\sigma + j\omega)t}dt$$

Taking $s = \sigma + j\omega$

$$\mathcal{L}\big(f(t)\big) = \int_0^{\infty} f(t)e^{-st}dt \quad (2\text{-}30)$$

### 2.2.3 Linear and Time Invariant System (LTI) Systems

LTI systems are all systems that react in a specific manner when subjected to an arbitrary input and have the following defining properties:

#### 2.2.3.1 Homogeneity

This states that if an input to a system is scaled by a factor $a$ then the output will be scaled by the same factor.

#### 2.2.3.2 Superposition (Additivity)

This states that if you two different inputs being imposed on an LTI at the same time, the output will be the superposition of the output signals from each input signal.

#### 2.2.3.3 Time Invariance

This refers to the system behaving the same regardless of when in time an action takes place.

Very rarely do real life systems meet the requirement of linear time-invariant systems but linear systems still important because we can solve them.

## 2.3 PID Theory

There exist numerous algorithms to implement control systems. The most used among these is the PID control with most feedback loops based on PID control or some minor variations of it.

A version of the PID control is;

$$\mathrm{u(t) = ke(t) + k_i \int_0^t e(\tau)d\tau + k_d \frac{de}{dt}} \qquad (2\text{-}31)$$

where $u$ is the control signal and $e$ is the control error ($e = r - y$). The reference value is also called the setpoint. The control signal is thus a sum of three terms: The P-term

(which is proportional to the error), the I-term (which is proportional to the integral of the error), and the D-term (which is proportional to the derivative of the error). The controller parameters are proportional gain $k$, integral gain $k_i$ and derivative gain $k_d$. The controller can also be parameterized as

$$u(t) = k_c \left( e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right) \quad (2\text{-}32)$$

where $T_i$ is called integral time and $T_d$ derivative time. The proportional part acts on the present value of the error, the integral represent and average of past errors and the derivative can be interpreted as a prediction of future errors based on linear extrapolation (Astrom & Murray, 2007).

### 2.3.1 Proportional Control

Proportional or P- controller gives output which is proportional to current error $e\ (t)$. It compares desired or set point with actual value or feedback process value. The resulting error is multiplied with proportional constant to get the output. If the error value is zero, then this controller output is zero.

This controller requires biasing or manual reset when used alone. This is because it never reaches the steady state condition. It provides stable operation but always maintains the steady state error. Speed of the response is increased when the proportional constant Kc increase.

### 2.3.2 Integral Control

Due to limitation of p-controller where there always exists an offset between the process variable and set point, I-controller is needed, which provides necessary action to eliminate

the steady state error. It integrates the error over a period of time until error value reaches to zero. It holds the value to final control device at which error becomes zero.

Integral control decreases its output when negative error takes place. It limits the speed of response and affects stability of the system. Speed of the response is increased by decreasing integral gain $K_i$.

In above figure, as the gain of the I-controller decreases, steady state error also goes on decreasing. For most of the cases, PI controller is used particularly where high-speed response is not required.

While using the PI controller, I-controller output is limited to somewhat range to overcome the integral wind-up conditions where integral output goes on increasing even at zero error state, due to nonlinearities in the plant.

### 2.3.3 Derivative Control

I-controller doesn't have the capability to predict the future behavior of error. So, it reacts normally once the set point is changed. D-controller overcomes this problem by anticipating future behavior of the error. Its output depends on rate of change of error with respect to time, multiplied by derivative constant. It gives the kick start for the output thereby increasing system response.

In the above figure response of D controller is more, compared to PI controller and also settling time of output is decreased. It improves the stability of system by compensating phase lag caused by I-controller. Increasing the derivative gain increases speed of response.

So finally, we observed that by combining these three controllers, we can get the desired response for the system. Different manufactures design different PID algorithms.

There have been various types of techniques applied for PID tuning, one of the earliest being the Ziegler Nichols technique. These techniques can be broadly classified as classical and computational or optimization techniques.

Classical techniques make certain assumptions about the plant and the desired output and try to obtain analytically, or graphically some feature of the process that is then used to decide the controller settings. These techniques are computationally very fast and simple to implement, and are good as a first iteration. But due to the assumptions made, the controller settings usually do not give the desired results directly and further tuning is required (Bansal et al., 2012).

## 2.4 Zeigler-Nichols' Heuristic Algorithm

This is by far the most popular tuning method in use. It was proposed by John Ziegler and Nathaniel Nichols in 1942 and is still a simple, fairly effective PID tuning method. There are two methods proposed by Ziegler and Nichols. The proposed Ziegler Nichols setting is given in Table 2-1.

Table 2-1: Ziegler Nichols' settings

| **Controller** | $K_p$ | $T_i$ | $T_D$ |
|---|---|---|---|
| **P** | 1/a | - | - |
| **PI** | 0.9/a | 3.33 $\theta$ | - |
| **PID** | 1.2/a | 2$\theta$ | 0.5$\theta$ |

### 2.4.1 Process reaction curve tuning technique

In the process reaction curve method, a process reaction curve is generated in response to a disturbance. This process curve is then used to calculate the controller gain, integral time and derivative time. The method is performed in open loop so that no control action occurs and the process response can be isolated. To generate a process reaction curve, the process is allowed to reach steady state or as close to steady state as possible. Then, in open loop, so that there is no control action, a small step disturbance is introduced and the reaction of the process variable is recorded. Figure 2.3 shows a typical process reaction curve for the process variable (PV) generated using the above method for a generic self-regulating process. The term self-regulating refers to a process where the controlled variable eventually returns to a stable value or levels out without external intervention (Svrcek et al., 2007).

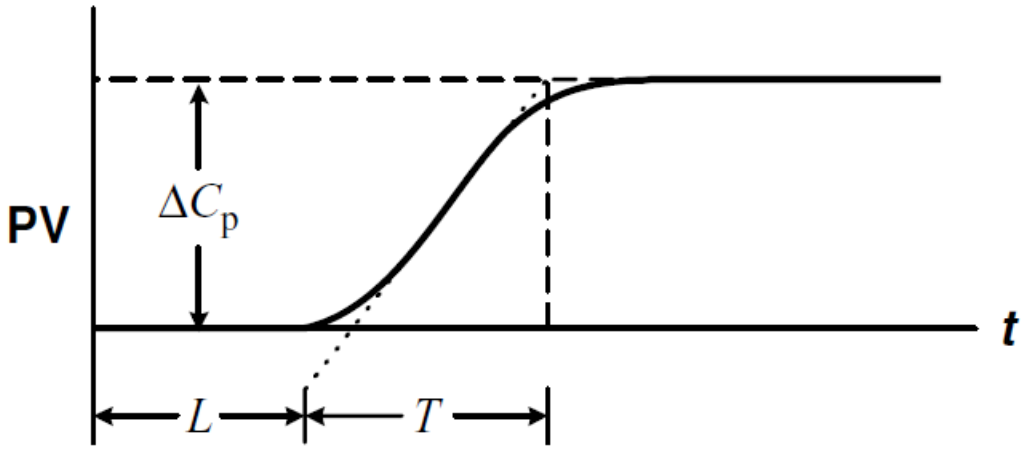


Figure 2.3 : Process Curve (Svrcek et al., 2007)

Where

*L* (min) lagtime

$T$ (min) time constant estimate

$P$ (%) initial step disturbance

$\Delta C_p(\%)$ change in PV in response to step disturbance, (change in PV)/ (PV span) *100

$N = \frac{\Delta C_p}{T}$ ($\%min^{-1}$) reaction rate

$R = \frac{L}{T} = \frac{NL}{\Delta C_p}$ (dimensionless) lag ratio

The Ziegler–Nichol's process reaction curve tuning method for a PI controller is as follows:

1. Determine a reasonable value for the step valve change P. This value is arbitrarily chosen, but typically 5 per cent is reasonable.
2. With the controller in manual mode, manually move the valve 'P' per cent.
3. Wait until the PV lines out to the new steady-state value.
4. Determine N and R from the process reaction curve.
5. Perform the following calculations:
   i. controller gain $Kc = \frac{0.9P}{NL}$
   ii. controller integral time $Ti = 3.33\ L$
6. Implement these recommendations for the controller settings in the controller.
7. Close the control loop by placing the controller in automatic mode.

### 2.4.2 Closed-loop tuning technique

The closed-loop technique of Ziegler and Nichols is another technique that is commonly used to determine the two important system constants, i.e., ultimate period and ultimate gain. Historically speaking, it was one of the first tuning techniques to be widely adopted.

In Ziegler–Nichols closed-loop tuning, as for the ATV technique, tuning values for proportional, integral and derivative controller parameters may be determined from the ultimate period and ultimate gain. However, Ziegler–Nichols closed-loop tuning is done by disturbing the closed-loop system and using the disturbance response to extract the values of these constants(Svrcek et al., 2007).

The Ziegler–Nichols closed-loop tuning method for a PI controller is as follows:

1. Attach a proportional-only controller with a low gain (no integral or derivative action).
2. Place the controller in automatic.
3. Increase proportional gain until a constant-amplitude limit cycle occurs.
4. Perform the following calculations:
    a. ultimate period $P_u$ = period taken from limit cycle
    b. ultimate gain $K_u$ = controller gain that produces the limit cycle
    c. controller gain $K_c = K_u/2.2$
    d. controller integral time $T_i = P_u/1.2$.

## 2.5 LabView Signal and Control System

PID Virtual Instruments (Vis) can be used with National Instruments hardware to develop LabVIEW control applications. Use I/O hardware, like a DAQ device, Field Point I/O modules, or a GPIB board, to connect PCs to the system you want to control. The I/O VIs provided in LabVIEW with the PID and Fuzzy Logic Toolkit can be used to develop a control application or modify the examples provided with the toolkit.

PID Vis in LabVIEW are capable of the following:

- Proportional (P); proportional-integral (PI); proportional-derivative
- (PD); and proportional-integral-derivative (PID) algorithms
- Gain-scheduled PID
- PID autotuning
- Error-squared PID
- Lead-Lag compensation
- Setpoint profile generation
- Multi-loop cascade control
- Feedforward control
- Override (minimum/maximum selector) control
- Ratio/bias control

### 2.5.1 LabVIEW PID VIs Implementation

#### 2.5.1.1 Error Calculation

The following formula represents the current error used in calculating proportional, integral, and derivative action.

$$e(k) = (SP - PV_f)$$

#### 2.5.1.2 Proportional Action

Proportional action is the controller gain times the error, as shown in the following formula.

$$u_p(k) = \left(K_c * e(k)\right)$$

#### 2.5.1.3 Trapezoidal Integration

Trapezoidal integration is used to avoid sharp changes in integral action when there is a sudden change in PV or SP. Use nonlinear adjustment of integral action to counteract overshoot. The larger the error, the smaller the integral action, as shown in the following formula.

$$u_I(k) = \frac{K_c}{T_i} \sum_{i=1}^{k} \left[\frac{e(i) + e(i-1)}{2}\right] \Delta t$$

#### 2.5.1.4 Partial Derivative Action

Because of abrupt changes in SP, only apply derivative action to the PV, not to the error e, to avoid derivative kick. The following formula represents the partial derivative action.

$$u_D(k) = -K_c \frac{T_d}{\Delta_t} \left(PV_f(k) - PV_f(k-1)\right)$$

#### 2.5.1.5 Controller Output

Controller output is the summation of the proportional, integral, and derivative action, as shown in the following formula.

$$u(k) = u_p(k) + u_I(k) + u_D(k)$$

#### 2.5.1.6 Output Limiting

The actual controller output is limited to the range specified for control output.

$$If\ u(k) \geq u_{\max}\ then\ u(k) = u_{\max}$$

And

$$If\ u(k) \leq u_{\min}\ then\ u(k) = u_{\min}$$

The following formula shows the practical model of the PID controller.

$$u(t) = K_c\left[(SP - PV) + \frac{1}{T_i}\int_0^t (SP - PV)dt - T_d\frac{dPV_f}{dt}\right]$$

## 2.6 Prior Work

### 2.6.1 Development of a Dynamic Simulator for Ghana Research Reactor 1

The main objective of this research was to develop a dynamic PC based simulator for GHARR-1, which can simulate the manual control of the reactor, the transient behavior due to thermal reactivity feedback as well as fission product poison build up. The simulator was developed using the graphically based programming tool, LabVIEW 2009 (Dzide, 2010).

The simulator was developed, viewing the entire system of the reactor as being made up of several smaller subsystems. These were modelled separately and brought together in the code to interact in the same way they do physically. Physical models of the MNSR core and its processes employed in the development of the simulator included the following: the neutronics point kinetic model with 6 groups of delayed neutrons and 9 groups of photo neutrons, heat transfer between the various zones of the reactor (both conduction and convection), xenon and samarium poisoning and thermal reactivity feedback.

From the results obtained the simulator was demonstrated to show the self-limiting nature of the MNSR in case of transients. It could simulate the temporal behavior of several reactor parameters such as output power, core coolant inlet/outlet temperature, bulk coolant temperature, fuel rod surface temperature and pool water temperature at various core reactivity levels. The simulator also allows the user to influence the reactivity levels in the reactor by simulating either the insertion or withdrawal of the control rod. The software incorporates the effects of thermal reactivity feedback as well as negative reactivity from the build-up of Xenon and Samarium within the reactor.

The results however showed deviations from those obtained experimentally. The power and temperature values were generally higher and approached peak values earlier than the experimental results. The deviations were low in the initially but increased significantly in the middle regions and decreased again when approaching the peak value

### 2.6.2 Development of an educational nuclear research reactor simulator

This paper introduced the development of a research reactor educational simulator based on LabVIEW that allows the raining of operators and studying different accident scenarios and the effects of operational parameters on the reactor behavior(Arafa et al., 2014).

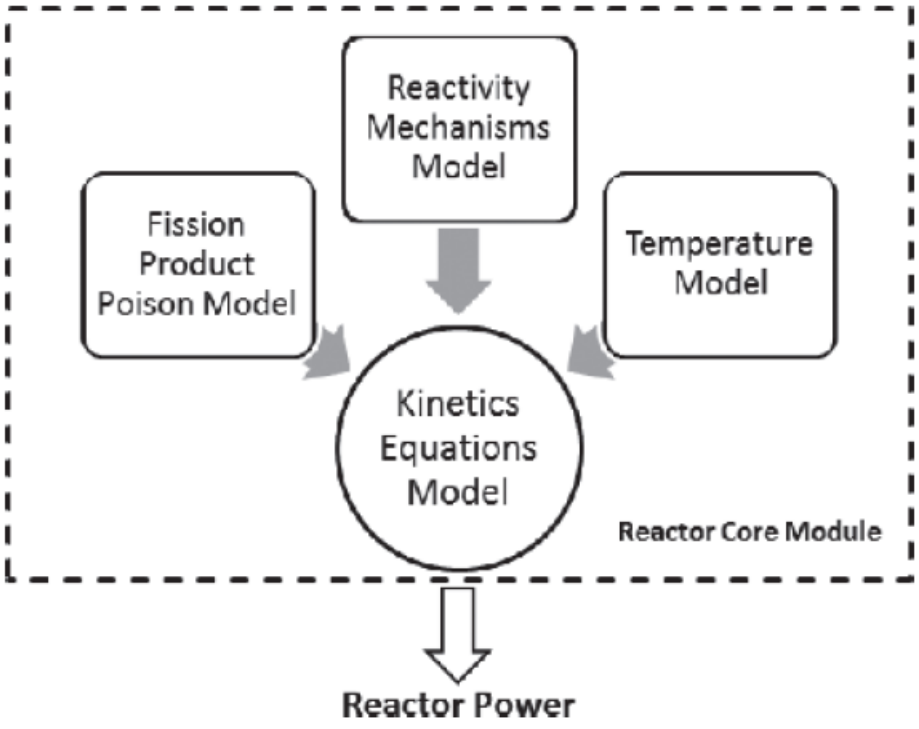


Figure 2.4: Design concepts of reactor core module (Arafa et al., 2014)

Figure 2.4 shows the block diagram of the module which consists of four main processes: Fission Product Poison, Reactivity Mechanisms, Temperature and the Kinetics Equations. These processes calculate properties of the reactor core, and store the calculation results of all processes to specially defined variables, according to their parameter type. These calculation results affect the calculation of the other simulator modules, which in turn have a feedback on the Reactor Core module. Each process is a separate LabVIEW VI file, contained in its own simulation sub-system, and used its own data block and functions.

The presented educational simulator with soft panels interface based on LabVIEW was built to imitate nuclear research reactors. The simulator uses models for the reactor parameters such as the neutronics and thermo hydraulic effects, and fission products poisoning. Although, the constructed simulator provides a simulation for a particular research reactor, it has the ability to simulate other reactor types by changing the inputs reactor parameters, and modifying the data display windows (Arafa et al., 2014).

# CHAPTER 3:METHODOLOGY

## 3.1 Introduction

In the development of the simulator, the entire system of the reactor was broken up into several smaller sub-systems. These sub-systems were built to interact with one another but also work separately.

## 3.2 Physical Models

Physical models of GHARR-1 and its processes used in designing of the simulator are as follows:

- neutronics point kinetic model having 6 groups of delayed neutrons and 9 groups of photo neutrons,
- the thermodynamics between the various zones of the reactor be it conduction or convection,
- xenon and samarium poisoning and
- thermal reactivity feedback.

### 3.2.1 Neutronics Model

This neutronics consisted of 2 ordinary differential equations, namely:

- the time rate of change of total reactor power

- the time rate of change of delayed neutron precursors from fission products as well as photo-neutron precursors.

Values for the delayed neutron fractions from both fission products (Lewis, 2008) and photo-neutrons (Hainoun & Khamis, 2000) were obtained from literature. It considered contributions of reactivity from the core excess, control rod, thermal feedback and poison feedback.

Assuming 6 groups of delayed neutrons and 9 groups of photo-neutrons, the neutron population can be obtained from the solution of the following sets of equations:

$$\frac{dn(t)}{dt} = \frac{\rho-\beta}{\Lambda}.n(t) + \sum_{i=1}^{6}\lambda_i C_i + \sum_{j=1}^{9}\lambda_j^p C_j^p \qquad (3\text{-}1)$$

$$\frac{dC_i}{dt} = \left(\frac{\gamma\beta_i}{\Lambda}\right)n(t) - \lambda_i C_i \qquad (3\text{-}2)$$

$$\frac{dC_j^p}{dt} = \left(\frac{\gamma_j^p\beta_j^p}{\Lambda}\right)n(t) - \lambda_j^p C_j^p \qquad (3\text{-}3)$$

Where:

$n(t)$ = neutron population density

$\rho(t)$ = net reactivity

$\beta$ = effective delayed neutron fraction

$\Lambda$ = mean neutron lifetime

$\lambda_i$ = decay constant of the $i^{th}$ precursor group

$C_i$ = concentration of the $i^{th}$ precursor group

$\gamma$ = efficiency of delayed and photo neutrons

During the execution of the algorithm, $P(t) \propto n(t)$, thus Equation (3-1) was redefined in terms of Power, $P$. TO represent all delayed and photo-neutrons, a single group was used, $C'$.

$$\frac{dn(t)}{dt} = \left(\frac{\rho-\beta}{\Lambda}\right) P(t) + \lambda C'(t) \quad (3\text{-}4)$$

$$\frac{dC_i}{dt} = \left(\frac{\gamma\beta_i}{\Lambda}\right) n(t) - \lambda_i C_i \quad (3\text{-}5)$$

### 3.2.2 Thermal Hydraulic Model

A system of equations was developed to using Newton's laws of convection is well as the Law of Conservation of Energy. It comprised heat transfer among the specific regions (see Figure 3.1and Figure 3.2) of the core. These ordinary differential equations are then represented in terms of physical properties of the reactor materials (water, fuel rod and beryllium shims), such as density, specific heat capacity, viscosity. These properties are however temperature dependent and hence are expressed as temperature dependent equations.

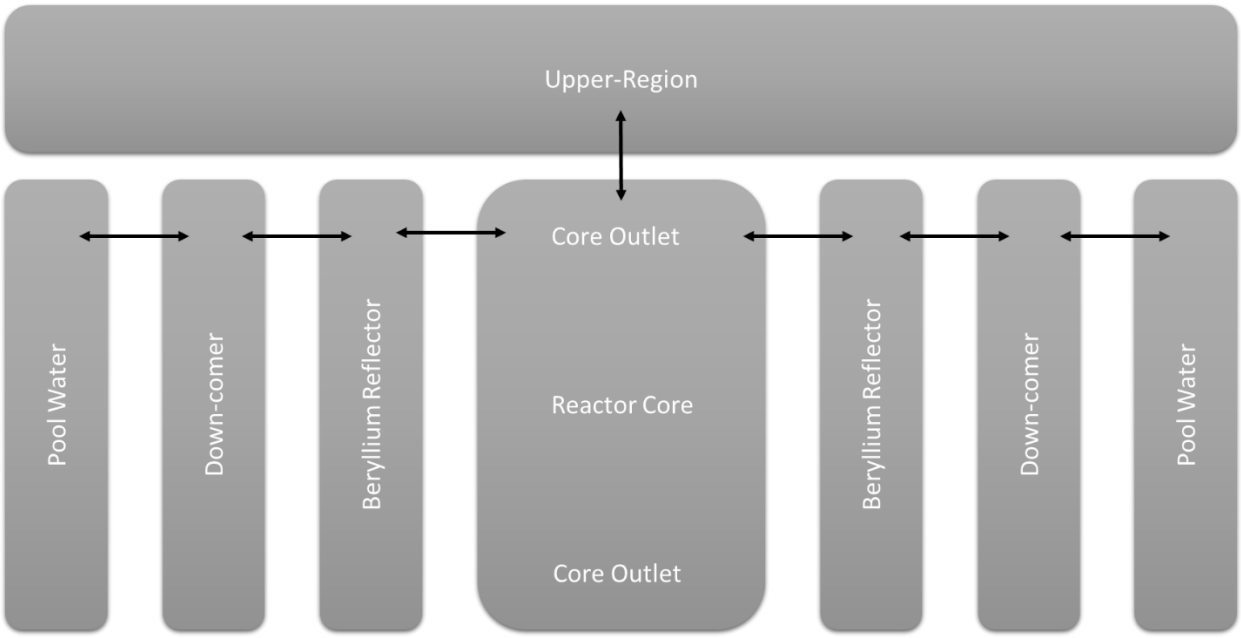


Figure 3.1: Heat Transfer Regions in GHARR-1

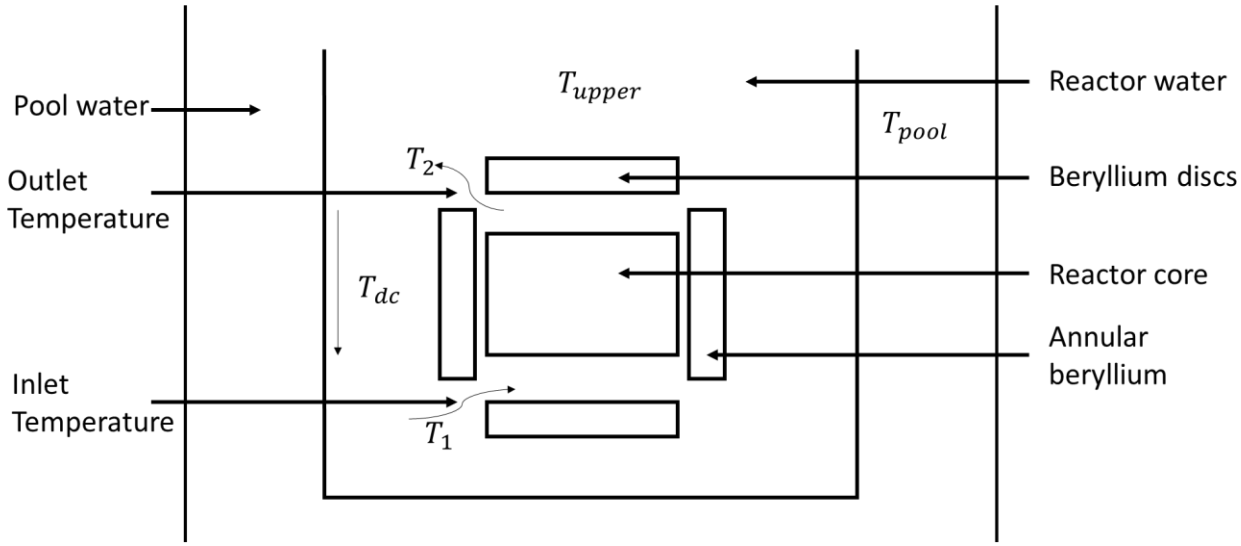


Figure 3.2: Longitudinal Section of Reactor Indicating Temperature Transfer

The following are the energy balance equations describing the heat transfer between the various regions of the reactor shown in the above diagram. Variables used here have been defined.

### 3.2.2.1 Heat Transfer (Fuel)

$$\left\{\begin{matrix}\text{Rate of Change}\\ \text{of}\\ \text{Heat in Fuel}\end{matrix}\right\} = \left\{\begin{matrix}\text{Rate of Change}\\ \text{of Heat}\\ \text{generated}\\ \text{by fission}\end{matrix}\right\} - \left\{\begin{matrix}\text{Rate of Change}\\ \text{of Heat}\\ \text{transferred to}\\ \text{moderator}\end{matrix}\right\} \tag{3-6}$$

$$(m_{fm}C_{pfm} + m_cC_{pc})\frac{dT_c}{dt} = P - A_sh \times \{T_c(t) = T_f(t)\} \tag{3-7}$$

$$\frac{dT_c}{dt} = \frac{P}{K} - L \times \{T_c(t) - T_f(t)\} \tag{3-8}$$

Where:

$$K = m_{fm}C_{pfm} + m_cC_{pc}, \qquad L = \frac{A_sh}{m_{fm}C_{pfm} + m_cC_{pc}}, \qquad T_f = \frac{T_1 + T_2}{2}$$

### 3.2.2.2 Heat transfer (Upper Region)

$$\left\{\begin{matrix}\text{Rate of Change}\\ \text{of Heat}\\ \text{in Upper Reactor}\\ \text{Water Region}\end{matrix}\right\} = \left\{\begin{matrix}\text{Rate of Heat}\\ \text{gain}\\ \text{from}\\ \text{Reactor Outlet}\end{matrix}\right\} - \left\{\begin{matrix}\text{Rate of Change}\\ \text{of Heat lost to}\\ \text{Pool Water}\end{matrix}\right\} \tag{3-9}$$

$$m_uC_{pfu}\frac{dT_u}{dt} = A_{up}U(t)(C_{pf2}\rho_{f2}T_2(t) - C_{pf3}\rho_{f3}T_3(t) - h_{cp}A_{cp}\{T_u(t) - T_p(t)\} \tag{3-10}$$

$$\frac{dT_3}{dt} = E_2 \times T_2(t) - E_u(t) \times T_u(t) - F \times \{T_u(t) - T_p(t)\} \tag{3-11}$$

Where:

$$E_2 = \frac{A_{up}C_{pf2}\rho_{f2}U(t)}{m_uC_{pfu}}, \qquad E_3 = \frac{A_{up}C_{pfu}\rho_{fu}U(t)}{m_uC_{pfu}}, \qquad F = \frac{h_{cp}A_{cp}}{m_uC_{pfu}}$$

### 3.2.2.3 Heat transfer (Beryllium)

$$\left\{\begin{matrix}\text{Rate of Change}\\ \text{of Heat}\\ \text{in Beryllium}\end{matrix}\right\} = \left\{\begin{matrix}\text{Rate of Heat}\\ \text{gain}\\ \text{from}\\ \text{Moderator}\end{matrix}\right\} - \left\{\begin{matrix}\text{Rate of Change}\\ \text{of Heat lost to}\\ \text{Down} - \text{comer}\\ \text{region}\end{matrix}\right\} \tag{3-12}$$

$$m_{Be}C_{pBe}\frac{dT_{Be}}{dt} = h_{Be}A_1\{T_f(t) - T_{Be}(t)\} - h_{Be}A_2\{T_{Be}(t) - T_{dc}(t)\} \tag{3-13}$$

$$\frac{dT_{Be}}{dt} = G \times \{T_f(t) - T_{Be}(t)\} - H \times \{T_{Be}(t) - T_{dc}(t)\} \quad (3\text{-}14)$$

Where:

$$G = \frac{h_{Be}A_1}{m_{Be}C_{pBe}}, \qquad H = \frac{h_{Be}A_2}{m_{Be}C_{pBe}}$$

### 3.2.2.4 Heat transfer (Down-Comer Region)

$$\begin{Bmatrix}\text{Rate of Change}\\ \text{of Heat in}\\ \text{Down} - \text{comer}\\ \text{Region}\end{Bmatrix} = \begin{Bmatrix}\text{Rate of Heat}\\ \text{gain}\\ \text{from}\\ \text{Beryllium}\end{Bmatrix} - \begin{Bmatrix}\text{Rate of Change}\\ \text{of Heat lost to}\\ \text{Pool Water}\end{Bmatrix} \quad (3\text{-}15)$$

$$m_{Be}C_{pBe}\frac{dT_{dc}}{dt} = h_{Bedc}A_{Bedc}\{T_{Be}(t) - T_{dc}(t)\} - h_{Rp}A_{Rp}\{T_{dc}(t) - T_p(t)\} \quad (3\text{-}16)$$

$$\frac{dT_{dc}}{dt} = S \times \{T_{Be}(t) - T_{dc}(t)\} - R \times \{T_{dc}(t) - T_p(t)\} \quad (3\text{-}17)$$

Where:

$$S = \frac{h_{Bedc}A_{Bedc}}{m_{dc}C_{pfdc}}, \qquad R = \frac{h_{Rp}A_{Rp}}{m_{dc}C_{pfdc}}$$

### 3.2.2.5 Heat Transfer (Pool Water)

$$\begin{Bmatrix}\text{Rate of Change}\\ \text{of Heat in}\\ \text{Pool Water}\end{Bmatrix} = \begin{Bmatrix}\text{Rate of Heat}\\ \text{gain}\\ \text{from}\\ \text{Down} - \text{Comer}\\ \text{Region}\end{Bmatrix} - \begin{Bmatrix}\text{Rate of Change}\\ \text{of Heat gained}\\ \text{from}\\ \text{Upper Region}\end{Bmatrix} \quad (3\text{-}18)$$

$$\frac{dT_p}{dt} = V \times \{T_{dc}(t) - T_p(t)\} - R \times \{T_u(t) - T_p(t)\} \quad (3\text{-}19)$$

Where:

$$V = \frac{h_{Rp}A_{Rp}}{m_pC_{pfp}}, \qquad R = \frac{h_{cp}A_{cp}}{m_pC_{pfp}}$$

### 3.2.3 Poison Model

Of importance in the reactor chain reaction are fission products with large absorption cross sections. They may be formed either directly from fission of from the $\beta^-$ decay chain of fission products. The following equations describe the rate of production of Iodine, Xenon, Promethium and Samarium from fission:

$$\frac{\partial \mathrm{I}}{\partial \mathrm{t}} = \gamma_{\mathrm{I}}\Sigma_{\mathrm{f}}\phi - \lambda_{\mathrm{I}}\mathrm{I} \tag{3-20}$$

$$\frac{\partial \mathrm{Xe}}{\partial \mathrm{t}} = \gamma_{\mathrm{Xe}}\Sigma_{\mathrm{f}}\phi - \lambda_{\mathrm{I}}\mathrm{I} - \lambda_{\mathrm{Xe}}\mathrm{Xe} - \sigma_{\mathrm{a}}^{\mathrm{Xe}}\phi\mathrm{Xe} \tag{3-21}$$

$$\frac{\partial \mathrm{Pm}}{\partial \mathrm{t}} = \gamma_{\mathrm{Pm}}\Sigma_{\mathrm{f}}\phi - \lambda_{\mathrm{Pm}}\mathrm{Pm} \tag{3-22}$$

$$\frac{\partial \mathrm{Sm}}{\partial \mathrm{t}} = \lambda_{\mathrm{Pm}}\mathrm{Pm} - \sigma_{\mathrm{a}}^{\mathrm{Xe}}\phi\mathrm{Pm} \tag{3-23}$$

Upon the solution of these equations, reactivity contribution by Xenon and Samarium can be represented in the following specific forms of Equation (2-25):

$$\rho(\mathrm{Xe}) = -\frac{\mathrm{Xe}.\sigma_{\mathrm{a}}^{\mathrm{Xe}}}{\Sigma_{\mathrm{a}}^{\mathrm{f}}} \tag{3-24}$$

$$\rho(\mathrm{Sm}) = -\frac{\mathrm{Sm}.\sigma_{\mathrm{a}}^{\mathrm{Sm}}}{\Sigma_{\mathrm{a}}^{\mathrm{f}}} \tag{3-25}$$

### 3.2.4 Control Rod Model

In this model, the relationship between reactivity introduced into the reactor by control rod movement and its position above the bottom of the core is given. This has been verified both experimentally and theoretically to approximately obey the following relationship (Lamarsh & Baratta, 2001):

$$\rho(x)_{control\ rod} = -\rho(H)\left[\frac{x}{H} - \frac{1}{2\pi}\sin\left(\frac{\mathrm{x}}{\mathrm{H}}\right)\right] \tag{3-26}$$

Where:

$x$ = position of control rod above the bottom of the core

$H$ = maximum possible height of the control rod above the core bottom

## 3.3 Design of LabVIEW based Algorithms

### 3.3.1 System Overview

The LabVIEW program was structured such that the whole system was represented as four separate subsystems interacting with each other. These subsystems are the Control Subsystem, the Neutronics Subsystem, the Thermal Hydraulics Subsystem and the Fission Product Poisoning Subsystem. Figure 3.3 shows the splitting of physical models and how they interact.

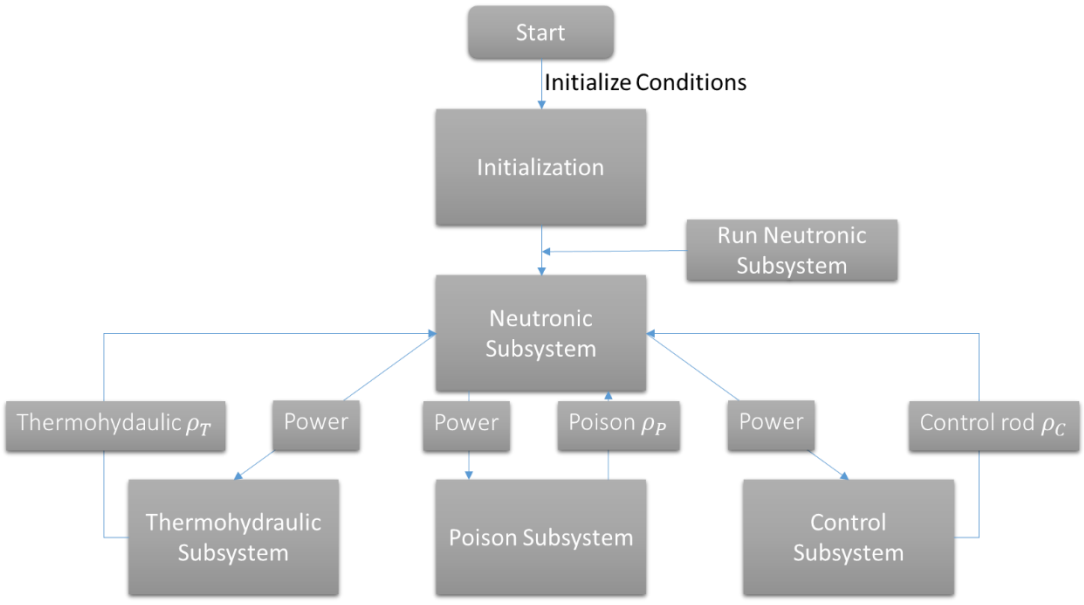


Figure 3.3: Main System Flow Chart

### 3.3.2 The Control Subsystem

This subsystem describes how the user inputs are processed in order to control the simulated nuclear process (see Figure 3.4). The reactivity values from the length of control rod at any point in time are the main output of this subsystem.

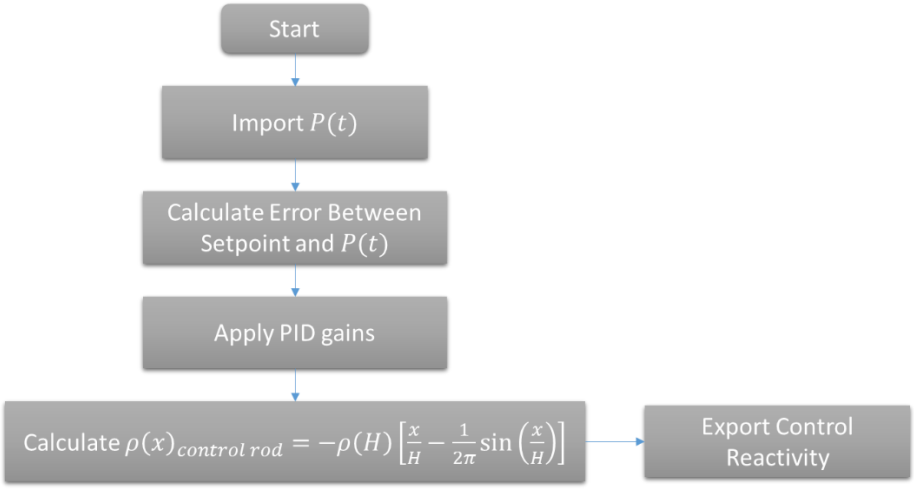


Figure 3.4: Control System Flow Chart

### 3.3.3 The Neutronics Subsystem

The algorithm used for the neutronics sub-system takes inputs from the three other subsystems to calculate the total reactivity at a given time (see Figure 3.5). This is used to solve the set of differential equations in the neutronics model. Its output is the total reactor power to the thermal hydraulics subsystem.

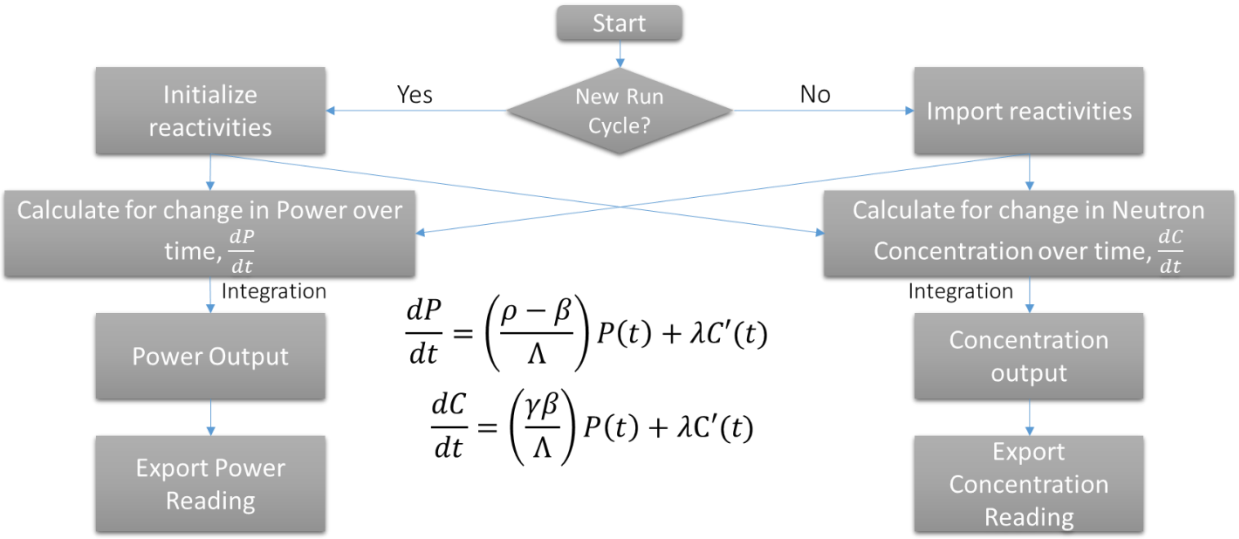


Figure 3.5: Neutronics Model Flow Chart

### 3.3.4 Thermal Hydraulics Subsystem

This takes in the reactor power as the main input and solves the heat balance ordinary differential equations numerically for the temperatures of the various zones and components of interest in the reactor. Figure 3.6 shows the thermal hydraulic flow chart employed.

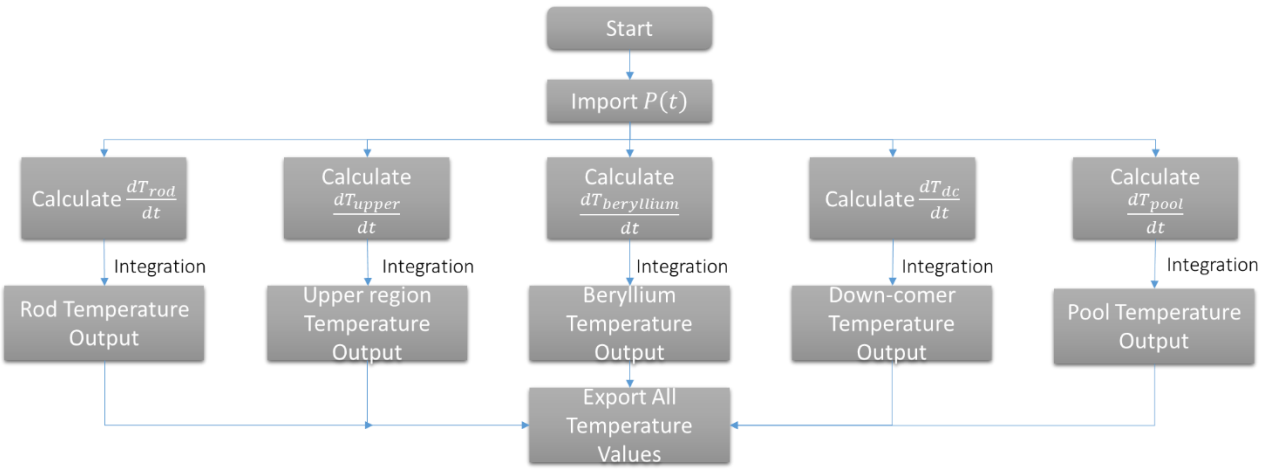


Figure 3.6: Thermal Hydraulic Model Flow Chart

### 3.3.5 Fission Product Poisoning Model

This algorithm calculates the concentrations of Xenon and Samarium within the reactor system from a set of differential equations. These were then used to calculate the reactivity removed by these poisons in reactor between start up and shutdown. The poison model used is showed in Figure 3.7.

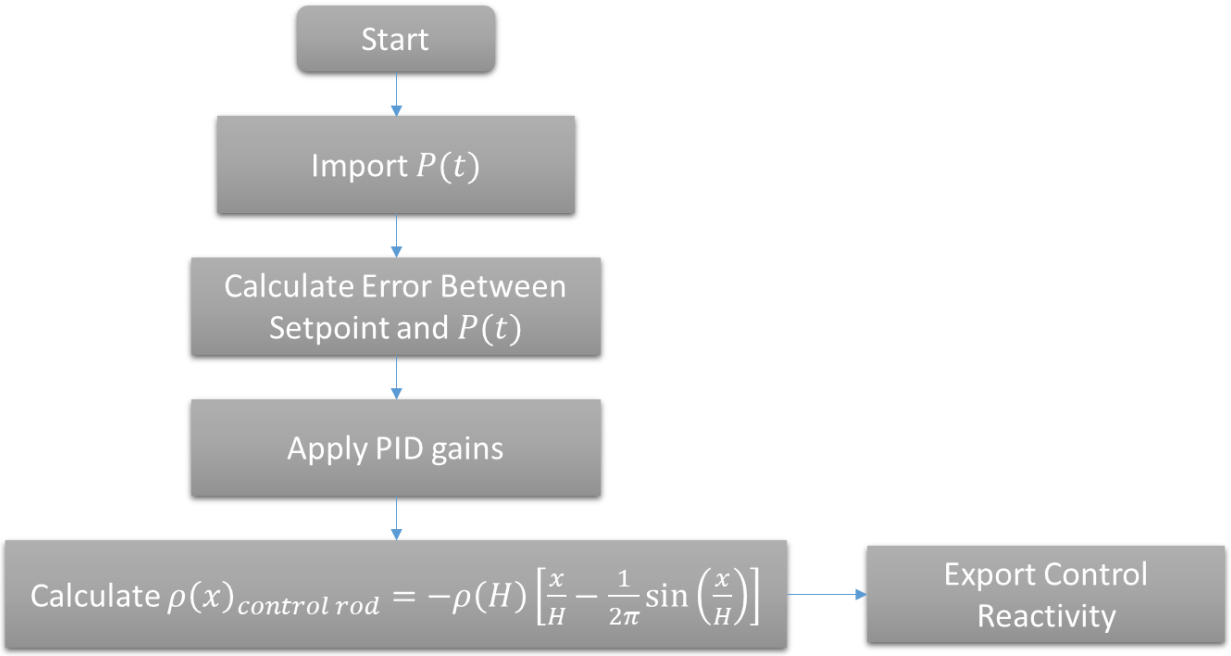


Figure 3.7 Poison Model Flow Chart

## 3.4 LabVIEW PID Design

A Proportional-Integral (PI) Controller type PID controller was used for developing the simulator. This was chosen to correspond with the fact that GHARR-1 uses a servomotor to move the control rod. Servomotors are designed to only need a proportional and integral control unit and thus the choice of PID controller.

### 3.4.1 Error Calculation

The error was calculated between a given power set point which is user defined and the power output of the neutronics subsystem at any given time $t$.

$$e(k) = \left(P_{setpoint} - P(t)\right) \tag{3-27}$$

The output of the controller provides a height, $x$, adjustment of the control rod which in turn creates a change in the reactivity input of the control rod. The proportional action of the controller gave:

$$x_p(k) = \left(K_c * e(k)\right) \tag{3-28}$$

The trapezoidal integration of the controller provided an equation:

$$x_I(k) = \frac{K_c}{T_i}\sum_{i=1}^{k}\left[\frac{e(i)+e(i-1)}{2}\right]\Delta t \tag{3-29}$$

A combination of Equation (1-28) and (1-29) will give the total output of the controller as:

$$\text{The height of Control rod, } x(k) = x_p(k) + x_I(k) \tag{3-30}$$

### 3.4.2 Output Limiting

The actual controller output was limited to the range specified for control output. This allowed for the use of different engineering variables as input and output. In the case of this research, that would be the input as error which was in Watts and the output as height which was in millimeters.

The algorithm used for this function was:

$$if\ x(k) \geq x_{\max}\ then\ x(k) = x_{\max}$$

and

$$f\ x(k) \leq x_{\min}\ then\ x(k) = x_{\min}$$

### 3.4.3 Controller Timing

To accurately design the control system, the timing of the PID controller, which instructs the amount of time between subsequent reads of input values, was linked to the timing of the entire simulation function. Table 3-1 shows the linked parameters between the simulation and the PID controller.

Table 3-1: Running Parameters for Simulation

| | |
|---|---|
| **Initial Step Size (seconds)** | 0.001 |
| **Minimum Step Size (seconds)** | 0.001 |
| **Maximum Step Size (seconds)** | 1 |

### 3.4.4 PID Tuning

The PID was initially tuned manually to provide LabVIEW's auto tuning algorithms a base with which to further tune the controller. Both the manual and auto tuning was done using the Zeigler Nichols Algorithm (LabView (National Instruments), 2009).

#### 3.4.4.1 Manual Tuning

The following controller tuning procedures are based on the work of Ziegler and Nichols, the developers of the quarter-decay ratio tuning techniques derived from a combination of theory and empirical observations (Corripio, 2000).

Although the closed-loop (ultimate gain) tuning procedure is very accurate, one must put the process in steady-state oscillation and observe the Process Variable (PV) on a strip chart. Complete the following steps to perform the closed-loop tuning procedure.

1. Set both the derivative time and the integral time on your PID controller to 0.
2. With the controller in automatic mode, carefully increase the proportional gain ($K_c$) in small increments. Make a small change in $SP$ to disturb the loop after each increment. As you increase $K_c$, the value of $PV$ should begin to oscillate. Keep making changes until the oscillation is sustained, neither growing nor decaying over time.
3. Record the controller proportional band ($PB_u$) as a percent, where $PB_u = 100/Kc$.
4. Record the period of oscillation ($T_u$) in minutes.
5. Multiply the measured values by the factors shown in Table 3-2 and enter the new tuning parameters into your controller. Table 3-2 provides the proper values for a quarter-decay ratio. If you want less overshoot, increase the gain Kc.

Table 3-2 : Zeigler Nichols Ratios

| Controller | PB (percentage) | Rest(minutes) | Rate(minutes) |
|---|---|---|---|
| ***P*** | $2.00PB_u$ | – | – |
| ***PI*** | $2.22PB_u$ | $0.83T_u$ | – |
| ***PID*** | $1.67PB_u$ | $0.50T_u$ | $0.125T_u$ |

For this research, the value of the controller constant, $K_c = 0.01$ at $T_u = 0.0096$. Considering only PI controllers our initial controller constants will be

$$K_c = 2.22 * \frac{100}{0.01} = 2.22, \qquad T_i = 0.83 * 0.0096 = 0.083$$

#### 3.4.4.2 Auto tuning

The PID Autotuning VI has several additional input and output values to specify the autotuning procedure. The two additional input values are "autotuning parameters" and "autotune?". The autotuning parameter's input is a cluster of parameters that the VI uses for the autotuning process. Because the autotuning wizard allows you to specify all of these parameters manually, the autotuning parameters input were left unwired. The 'autotune?' input takes a Boolean value supplied by a user control.

A Boolean control was wired on the front panel of the application to this input. When the user presses the Boolean control, the Autotuning Wizard opens automatically. The Boolean control mechanical action is set to Latch When Released so that the Autotuning Wizard does not open repeatedly when the user presses the control.

The Autotuning Wizard steps the user through the autotuning process. The PID Autotuning VI also has two additional output values "tuning completed?" and "PID gains out". The tuning completed? output is a Boolean value. It is usually FALSE and becomes TRUE only on the iteration during which the autotuning finishes. The autotuning procedure updates the PID parameters in PID gains out. Normally, PID gains out passes through PID gains and outputs PID gains out only when the autotuning procedure completes.

# CHAPTER 4:RESULTS AND DISCUSSION

## 4.1 Introduction

Results were taken from two main power simulation runs. These scenarios are classified by the power set-point when the control rod is fully inserted at the beginning of running the simulator.

## 4.2 Simulation Scenarios

The simulation scenarios were taken using the Runge-Kutta 45 variable method in LabVIEW to solve the sets of ordinary differential equations within the neutronics, thermal hydraulics and fission product poison subsystems in the simulation loop. The solver used the parameters in the Table 4-1.

Table 4-1: Initial Parameters used in running the simulation

| **ODE Solver Parameters** | |
|---|---|
| **Relative Tolerance** | 0.001 |
| **Absolute Tolerance** | $1 \times 10^{-7}$ |
| **Initial Step Size** | $0.001\ s$ |
| **Maximum Step Size** | $1\ s$ |
| **Minimum Step Size** | $0.001s$ |

### 4.2.1 Assumptions

This point kinetic neutronics model is mainly dependent on the assumptions that the neutrons produced from the fission chain reaction are mono-energetic. The delayed

neutrons created from the decay of fission products and the reflections of neutrons by the reflectors are accounted for by the assumption that they come from 6 and 9 groups of precursors respectively.

This thermal hydraulics model was based on a lumped parameter model approximation. This simplified the mathematics but however was expected to introduce some amount of error.

Steady-state analysis of the reactor thermal hydraulics has demonstrated that because of the low power density and the high thermal conductivity of the fuel element, the temperature difference between the center of the fuel meat and the fuel clad surface is $1^oC$ (Shi, 1990). It was therefore assumed that the temperature in the fuel and cladding regions were equal.

The average/bulk coolant temperature was also assumed to be the average of the core inlet and outlet temperatures. The temperatures were chosen because they represent the highest and lowest temperature of reactor water.

It has also been shown experimentally (Yongi, 1993) that the down-comer region temperature is approximately the same as the inlet temperature (i.e. $T_1 = T_4$) it was therefore assumed that they were equal in the implementation of this model.

The hydraulic diameter of the unit cross-sectional area of flow in the reactor core is important in calculating the heat transfer coefficient for heat transfer between the fuel element and the moderator. The lattice was not exactly triangular but it was found to be the most suitable scheme to use in this case since it was easier to calculate the cross-sectional area of its lattice while including minimal error. The unit cell of the lattice was therefore

assumed to be an equilateral triangle of sides equal to 10.95 mm. This length was chosen because it represented an average distance between adjacent fuel rods. The actual distances ranged between 10.8 and 11 mm.

## 4.3 Accuracy

The choice of a variable step size ordinary differential equation solver was made to avoid the drawback of having to adjust time steps in case of non-convergence, since the Runge-Kutta 27 method does automatic adjustment of its time steps to suit the specified tolerance. This is viewed as an improvement of the widely used 4th Order Runge-Kutta Method.

An absolute tolerance level of 0.0000001 and relative tolerance of 0.001 were found to be appropriate in choosing the required acceptable error for the simulation, by adjusting between the maximum and minimum step sizes of 1 and 0.001 seconds. Though it was a very wide range it was suitable enough to produce results without producing errors due to non-convergence.

### 4.3.1 Power set-point at 17 $kW$ from full rod insertion

This scenario shows the program running from start with a power set-point at 17 $kW$ from a fully inserted rod. Figure 4.1 through to Figure 4.4 show the various model output parameters as a function of time.

#### 4.3.1.1 Reactor Power versus Time

The GHARR-1 power against time is depicted in Figure 4.1. The figure shows power rising initially with a huge spike before levelling off at the preset value of 17000 Watts (17 $kW$). This spike is a result of the control rod withdrawing completely to allow power to rise above the preset value before normalizing. This causes a high positive reactivity feedback from the control rods allowing the power level to rise far above the given preset. However, after the spike the graph shows a correction of the power reading. After this correction, the power level stays constant till the end of the simulation showing a control of the power level by the simulator.

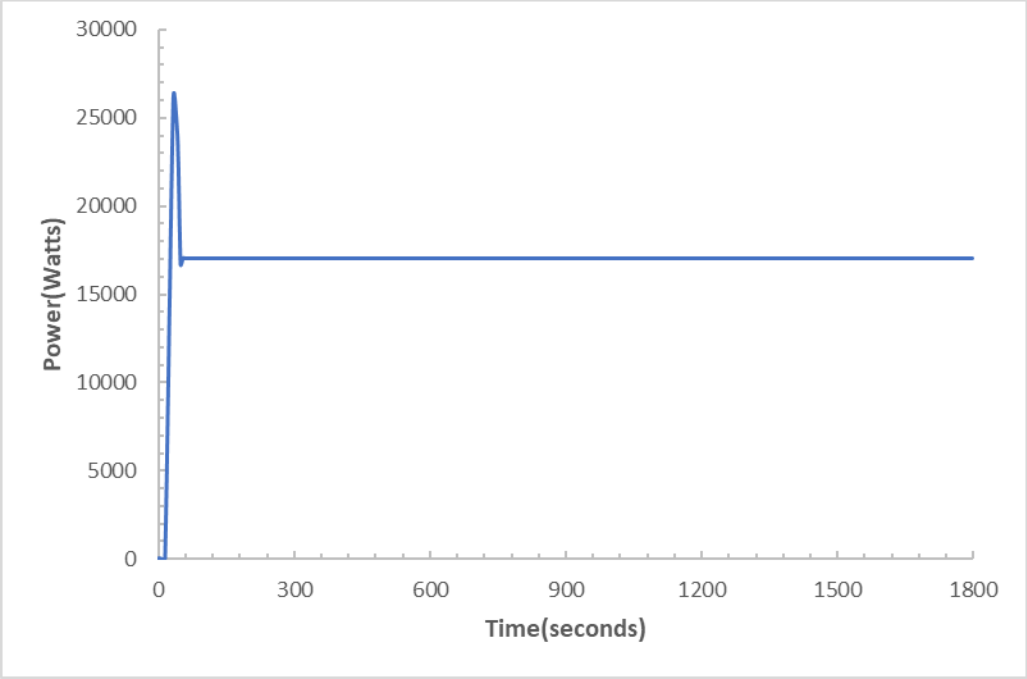


Figure 4.1:Power (W) against Time (s) with 17kW preset

#### 4.3.1.2 Inlet Temperature versus Time

The trend of the inlet temperature in Figure 4.2 shows the inlet temperature have a faster rate of increase till the approximate value of 1300 seconds when its rate of increase reduces. It keeps with this upward trend till the simulation is ended shows a constant gain of heat energy of fluid from the reactor core.

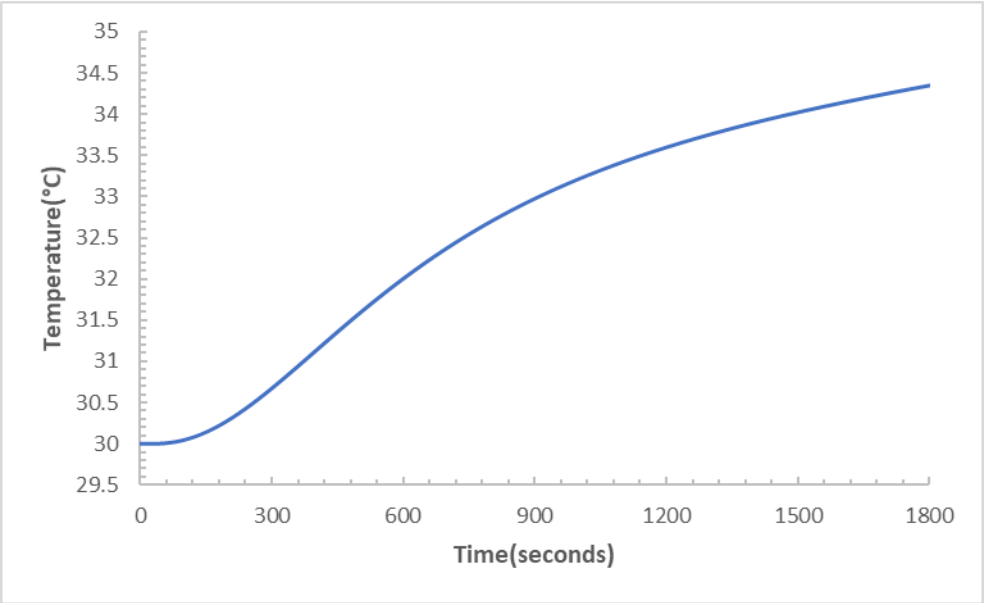


Figure 4.2: Inlet Temperature (°C) against Time (s) with 17kW preset

#### 4.3.1.3 Outlet Temperature versus Time

The trend of outlet temperature against time in Figure 4.3. This shows that the temperature of the outlet fluid has a direct relation to the power generation. This shows evidence of heat transfer from the fuel rods to the coolant.

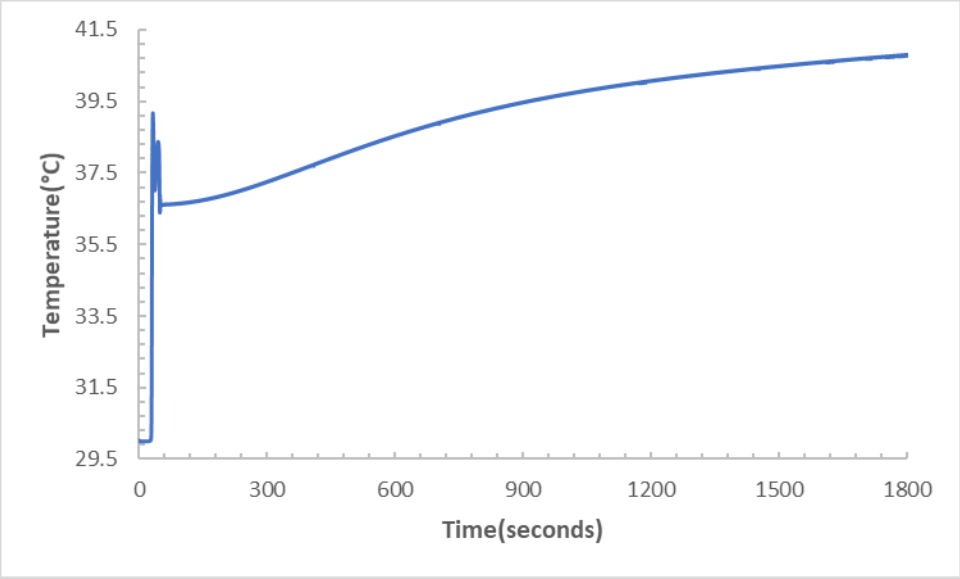


Figure 4.3: Outlet Temperature (°C) against Time (s) with 17 kW preset

#### 4.3.1.4 Clad Temperatures versus Time

The clad temperature is also in direct correlation to the power trend as shown in Figure 4.4. The clad temperature also shows a high spike in the initial trend but relaxes to a gradual increase as time goes on.

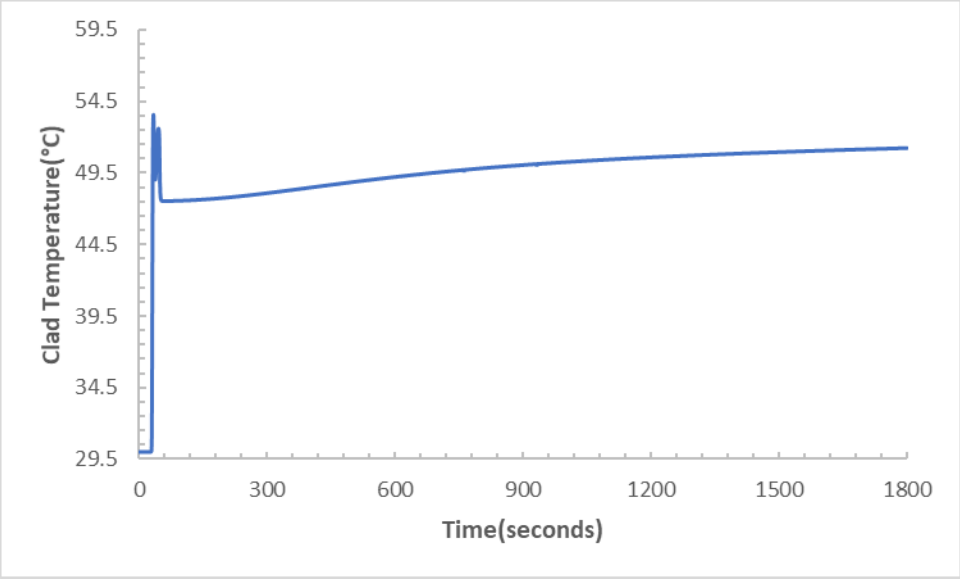


Figure 4.4: Clad Temperature (°C) against Time (s) with 17 kW preset

## 4.4 Discussion of Results

The simulator developed for an MNSR was designed taking GHARR-1 as a model. It took in inputs namely, initial core reactivity, control rod position and initial power preset to perform calculations foretelling reactor power, control rod position, temperatures across the reactor zones and fission product poison levels.

This software being designed for personal computers, had a goal of developing a system that not only mimicked the real reactor system behavior, but also was efficient in its use of computational time. For this reason, several simplifying assumptions were made in the models implemented by the code to cut back on computational cost.

The Runge-Kutta 27 method was chosen because it allowed the automatic adjustment of time steps, in order to avoid non-convergence situations that are more likely to occur for fixed step size methods.

## 4.5 Analysis of Results

It can be observed from the output of the simulation that the trends obeyed the expected patterns of a well-designed reactor. These include:

The relatively steep initial rise in reactor power to its peak value until negative reactivity from thermal feedback and poisons caused it to decrease gradually at a slower rate. This decline was however counteracted by the rise on the control rod height.

- The general rise and drop in temperature of the reactor outlet and fuel rod (that is, the temperature did not increase indefinitely). The power out also behaved in a similar manner.
- The very small and gradual increase in reactor inlet temperature and pool water with time.

The self-regulating effect of reactivity transients was evident as the slope of decrease of power always decreased with time.

## 4.6 Comparison with Operational Handbook of GHARR-1

To test the accuracy of the simulator to the actual Ghana Research Reactor 1, five separate inputs were compared to the simulator results. The parameters used for comparison are the control rod height, the inlet temperature and the difference between the inlet and outlet temperature at different times in the simulation. This was run for a $17kW$ power preset. Figure 4.5 and Figure 4.6 are representing the parametric comparison.

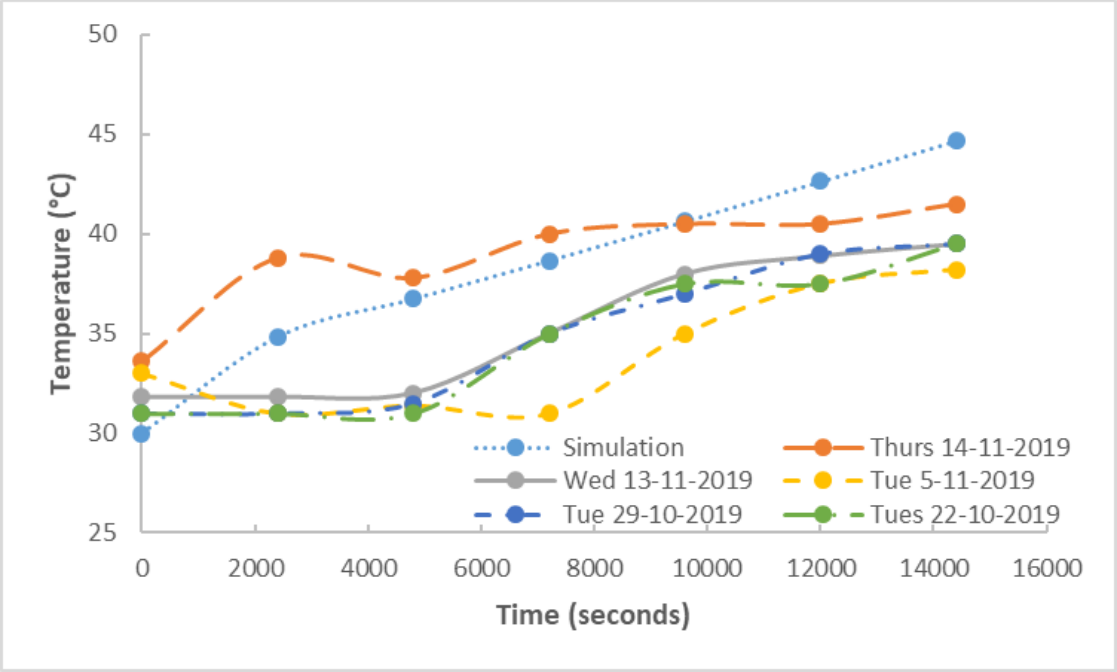


Figure 4.5: Comparison of inlet temperature between simulation and operational handbook

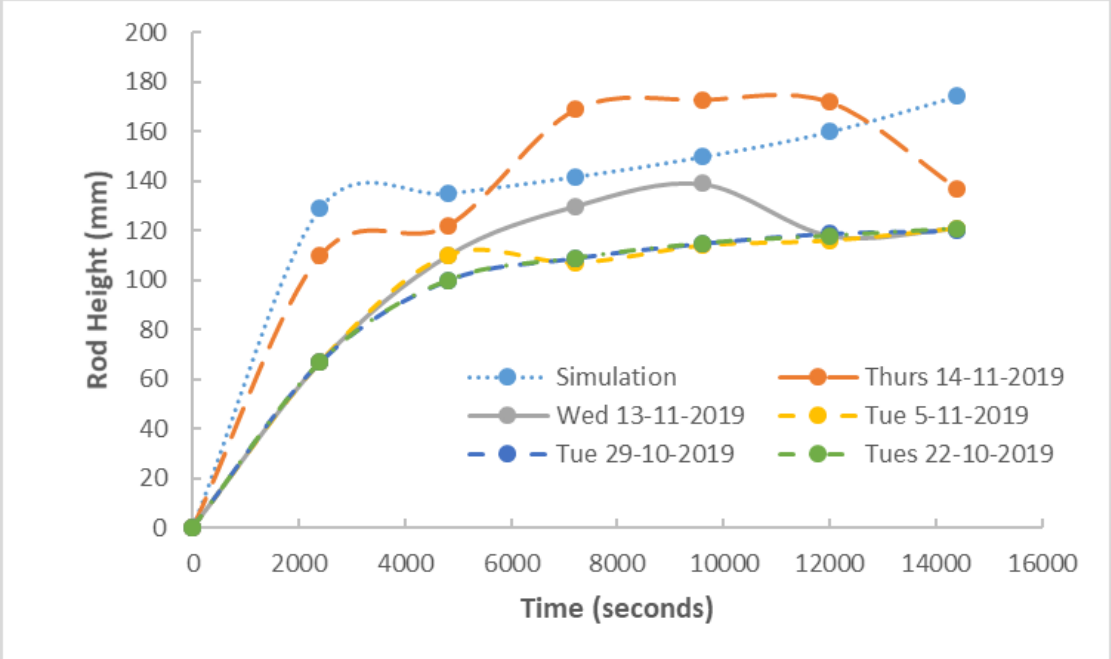


Figure 4.6: Comparison of control rod height between simulation and operational handbook

Figure 4.6 shows the comparison between control rod readings taking from various days as compared to that of the simulator. The graph shows a similar trend for the simulation control rod height and reading taken on Tuesday, the 5$^{th}$ of November 2019. There is however a difference in the height between said readings with that of the simulator being higher. This could be due to a higher production of fission poisons that need extra movement from the control rod to correct. For the temperature readings, it was noticed that the simulation followed qualitative trend as in the experimental readings.

# CHAPTER 5:CONCLUSION AND RECOMMENDATION

## 5.1 Conclusion

This research set out to design, produce a virtual dynamic simulator for neutronics thermal hydraulics, fission poison, and control models of an MNSR with GHARR-1 as a model.

The design of the virtual simulator was developed in LabVIEW with an emphasis on the Control and Simulation toolkit.

The previous work performed on GHARR-1 simulator development by (Dzide, 2010) was redeveloped in relation to the core conversion of GHARR-1. A PID Control System of GHARR-1 was designed using the PID toolkit in LabVIEW. For this PID system, an auto-tuning system was implemented using LabVIEW's Zeigler and Nichols heuristic algorithm representation.

Readings for parameters including; power (watts), control rod height (mm) temperature ($^0$C) and fission poison levels were taken and the state of the parameters during the execution of the simulator recorded.

From the analysis of the parameters in Chapter four, it is evident that the design and development of simulator is possible and the simulator can be used for training and education.

## 5.2 Recommendation

This dynamic virtual simulator for GHARR-1 can be used for training and education.

This program can be used in the instruction of NENG 654 Computational Methods in Power Systems in the School of Nuclear and Applied Science.

Different forms of control systems could be built for comparison with the PID control system. This could include Fuzzy Neural Logic, Linear Logic, On-Off Logic and Programmable Logic Controller.

The system could be made more robust by adding a speed control unit for the control rod. This would improve the response of the control rod to better resemble the GHARR-1 control rod.

In addition, there need to have accident scenarios carefully introduced into the already built simulator for effective training on the use of MNSR should also be considered.

## REFERENCES


Ahmed, Y., Ewa, I., Umar, I., Bezboruah, T., Johri, M., & Akaho, E. (2006). The low power miniature neutron source reactors: design, safety and applications. *Nigerian J. Phys.*, *14*.

Arafa, A., Saleh, H. I., & Ashoub, N. (2014). Development of an educational nuclear research reactor simulator. *Kerntechnik*, *79*(6), 518–527. https://doi.org/10.3139/124.110446

Asher, K. (2013). *Fourier series And Fourier Transform*. *4*(6), 73–76.

Astrom, K. J., & Murray, R. M. (2007). *Feedback Systems*. Princeton University Press. https://doi.org/10.1016/B978-012373947-6.00158-6

Bansal, H., Sharma, R., & Ponpathirkoottam, S. (2012). *PID Controller Tuning Techniques: A Review*. *2*, 168–176.

Corripio, A. B. (2000). *Tuning of Industrial Control Systems* (Second Edi). The Instrumentation, Systems, and Automation Society. All.

Dzide, S. K. (2010). *Development of Dynamic Simulator for GHANA RESEARCH REACTOR 1* (Vol. 1). University of Ghana.

Electrical4U. (2019). *Control System | Closed Loop Open Loop Control System | Electrical4U*. Electrical4U. https://www.electrical4u.com/control-system-closed-loop-open-loop-control-system/

Fara, P. (2015). Newton shows the light : a commentary on Newton his new theory about light and Subject Areas : *Philiosophical Transactions A*, *373*(2039).

Gosling, F. G. (1999). The Manhattan Project: Making the Atomic Bomb. *UNITED STATES DEPARTMENT OF ENERGY*, 66.

Hainoun, A., & Khamis, I. (2000). Determination of neutron generation time in miniature neutron source reactor by measurement of neutronics transfer function. *Nuclear Engineering and Design - NUCL ENG DES*, *195*, 299–305. https://doi.org/10.1016/S0029-5493(99)00160-0

Ho, A. K. (2014). Fundamental of PID Control. *PDHonline Course E331*, *331*, 1–20.

Jones, D. R., & Limited, S. (1992). Current applications of simulators in the process industries and future trends. *Operator Training Simulators, IEE Colloquium On*, 3/1-3/4.

Kalbasi, A., Krishnamurthy, D., Rolia, J., & Singhal, S. (2015). Simulation by example for complex systems. *Proceedings - Winter Simulation Conference*, *2015*, 974–985. https://doi.org/10.1109/WSC.2014.7019957

LabView (National Instruments). (2009). *PID and Fuzzy Logic Toolkit User Manual* (Issue June). National Instrument Corporation.

Lamarsh, J. R., & Baratta, A. J. (2001). Introduction to Nuclear Engineering. In *American Journal of Physics* (3rd Editio, Vol. 23, Issue 1). Prentice Hall. https://doi.org/10.1119/1.1933896

Lewis, E. E. (2008). *Fundamentals of Nuclear Reactor Physics* . Elsevier Inc . https://doi.org/10.1016/B978-0-12-370631-7.X0001-0

Muhammad, S. T., Ahmad, S. ul I., Chaudri, K. S., & Ahmad, A. (2008). Beryllium as reflector of MNSR. *Annals of Nuclear Energy*, *35*(9), 1708–1712. https://doi.org/10.1016/j.anucene.2008.02.005

Nesvizhevsky, V., & Villain, J. (2017). The discovery of the neutron and its consequences (1930–1940). *Comptes Rendus Physique*, *18*(9), 592–600. https://doi.org/https://doi.org/10.1016/j.crhy.2017.11.001

Odoi, H. C., Aboh, I. J. K., & Morman, J. A. (2016). Implementation of reactor core conversion for GHARR-1. *European Research Reactor Conference (RRFM) 2016: Conference Proceedings*, 1154. https://www.euronuclear.org/meetings/rrfm2016/transactions/rrfm2016-transactions.pdf

Shi, S. (1990). Low Power Research Reactor Thermal Hydraulics. *IAEA Workshop on Low Power Research Reactors*.

Stacey, W. M. (2007). Nuclear Reactor Physics: Second Edition. In *Nuclear Reactor Physics: Second Edition* (Second Edi). Wiley-VCH. https://doi.org/10.1002/9783527611041

Svrcek, W. Y., Mahoney, D. P., & Young, B. R. (2007). A Real-Time Approach to Process Control: Second Edition. In *A Real-Time Approach to Process Control: Second Edition*. https://doi.org/10.1002/9780470029558

US Department of Energy. (1993). DOE FUNDAMENTALS HANDBOOK NUCLEAR PHYSICS Volume 2 of 2. *Nuclear Physics*, *1*(January), 36.

Winsberg, E. B. (University of S. F. (2010). *Science in the Age of Computer Simulation*. The University of Chicago Press, Ltd., London.

Yongi, Z. (1993). *The Whole Simulated Heat Transfer Experiment and Calculations*. China Institute of Atomic Energy, Beijing, China.

Yongmao, Z., Fuxing, W., & deming, G. (1987). *The Miniature Neutron Source Reactor (MNSR) a Private Nuclear Tool BT - Neutron Radiography* (J. P. Barton, G. Farny, J.-L. Person, & H. Röttger (eds.); pp. 95–101). Springer Netherlands.

# Appendix A

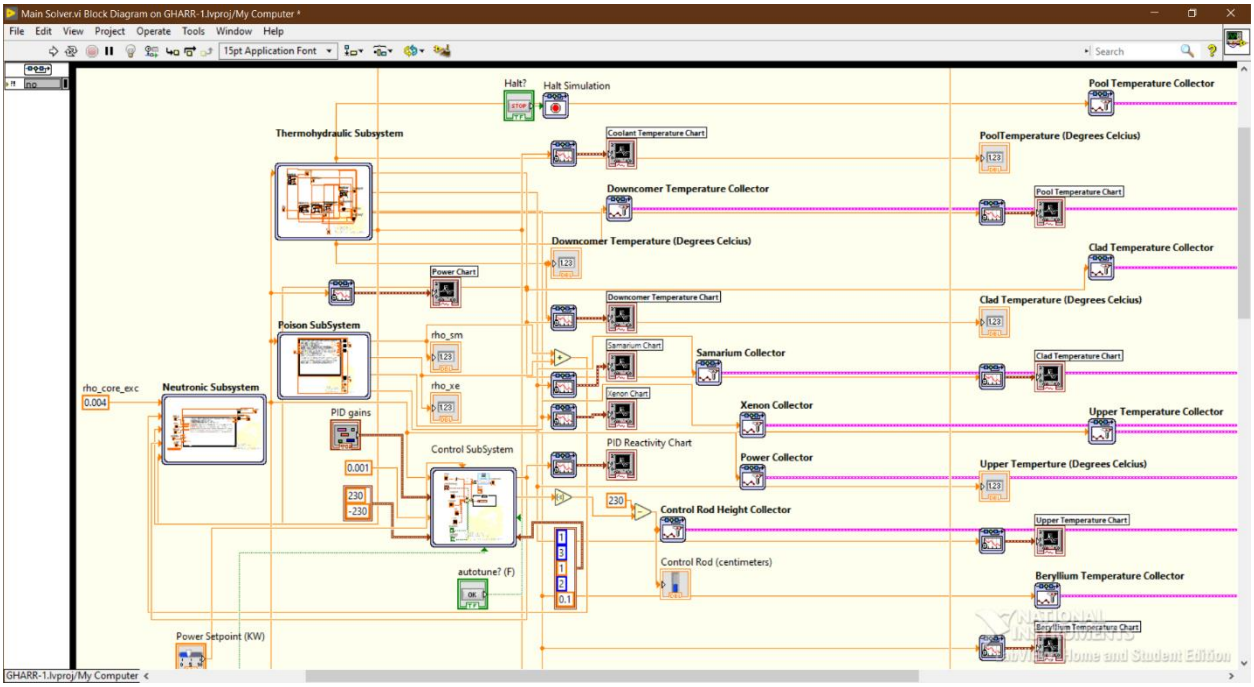


Figure 5.1 The back-end diagram for the main reactor simulator

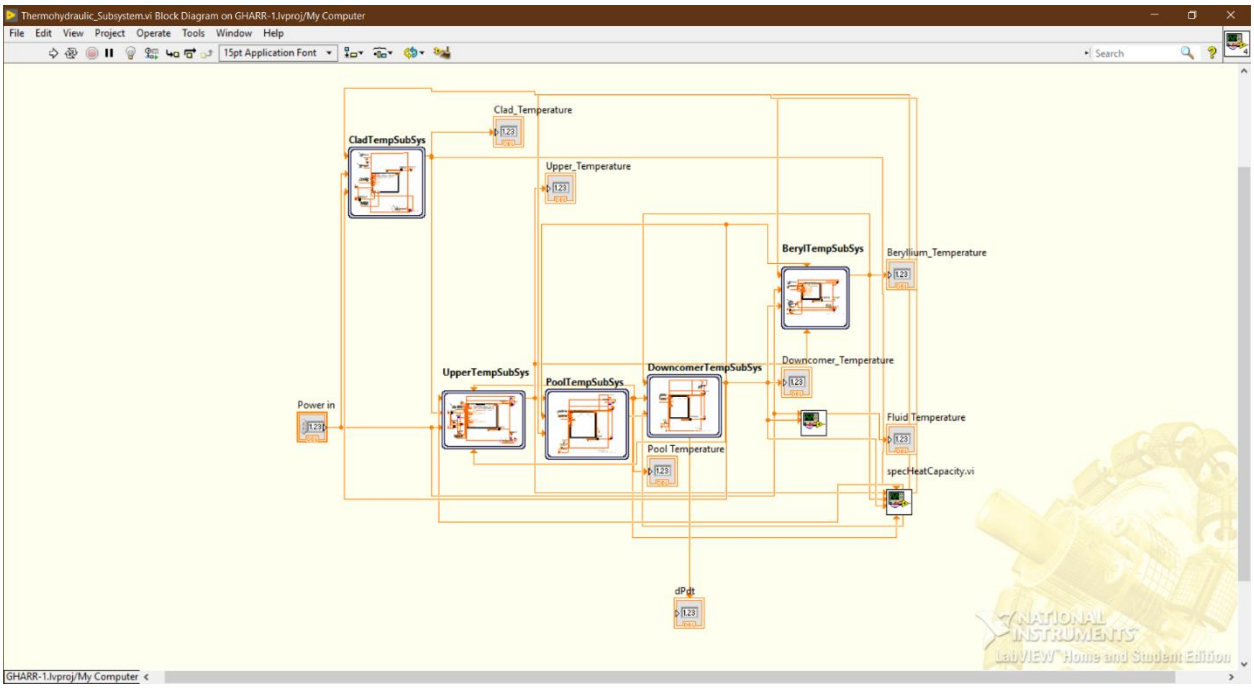


Figure 5.2 The back-end diagram for the thermohydraulic subsystem of the simulator

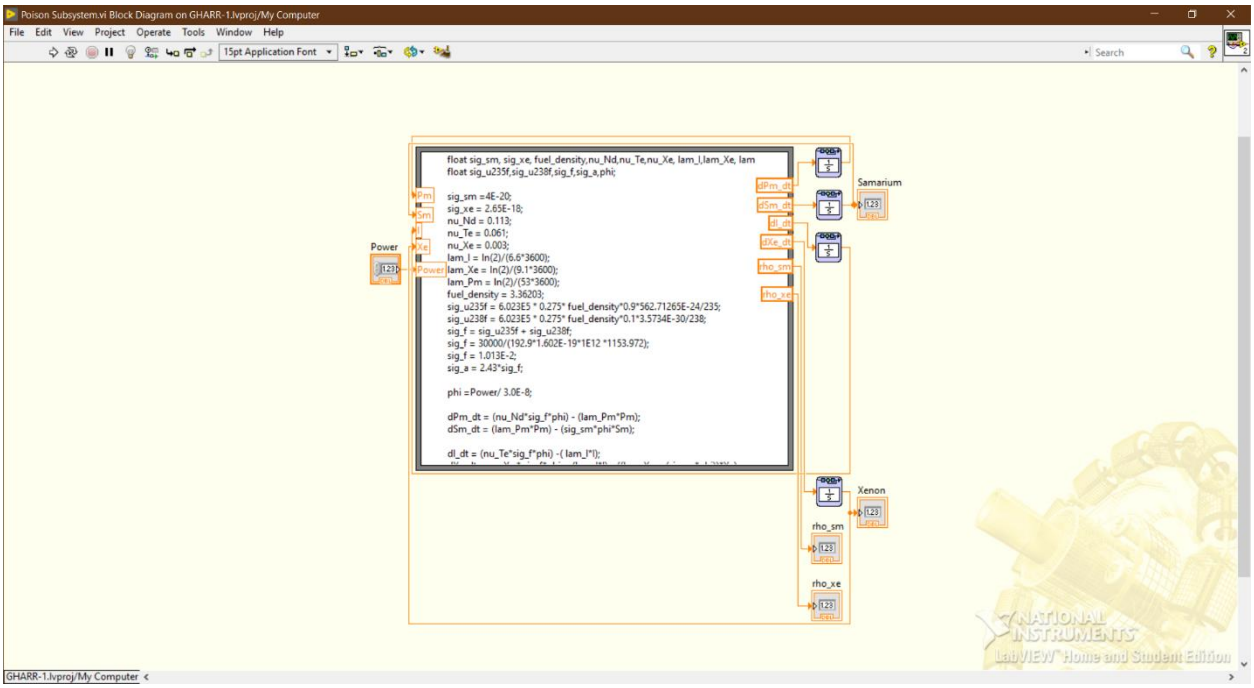


Figure 5.3 The back-end diagram for the poison subsystem of the simulator

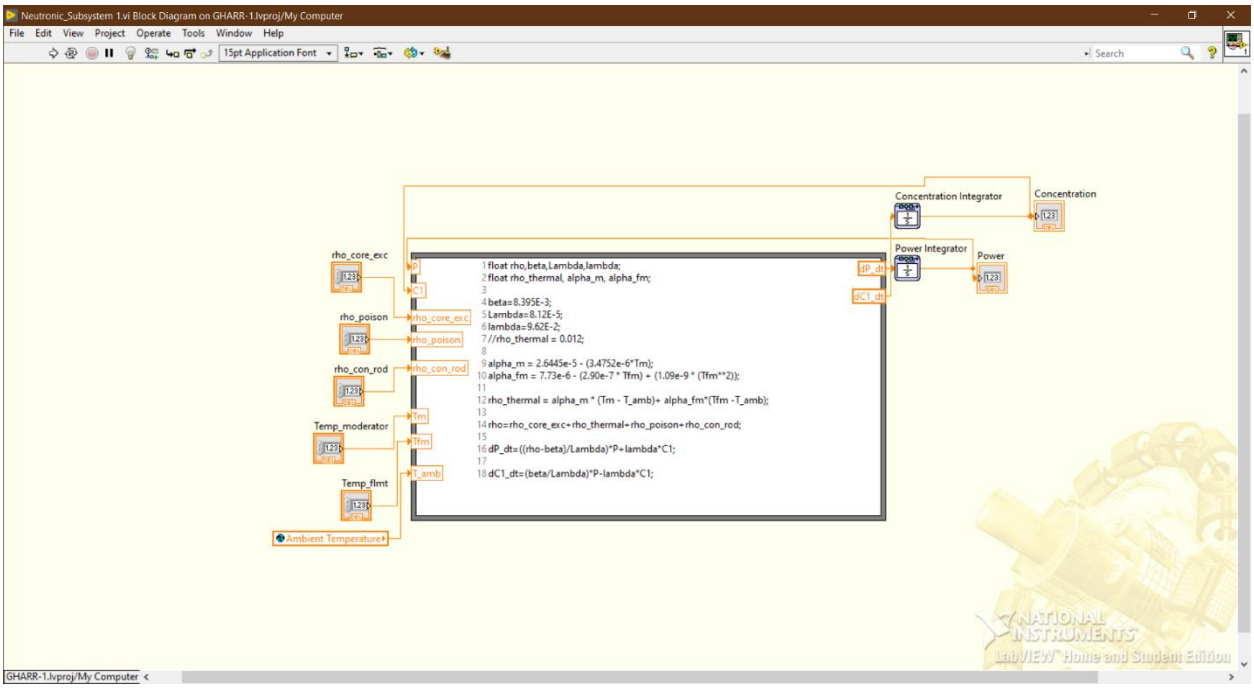


Figure 5.4 The back-end diagram for the neutronic subsystem of the simulator

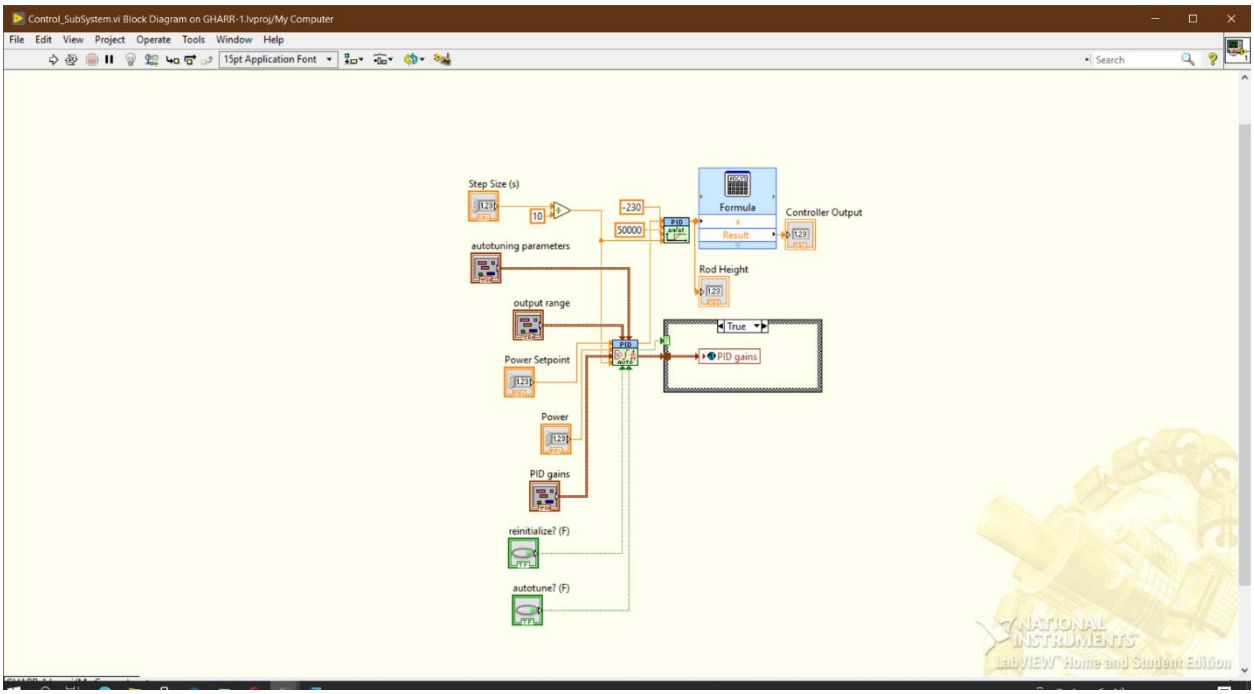


Figure 5.5 The back-end diagram for the control subsystem of the simulator

## Appendix B

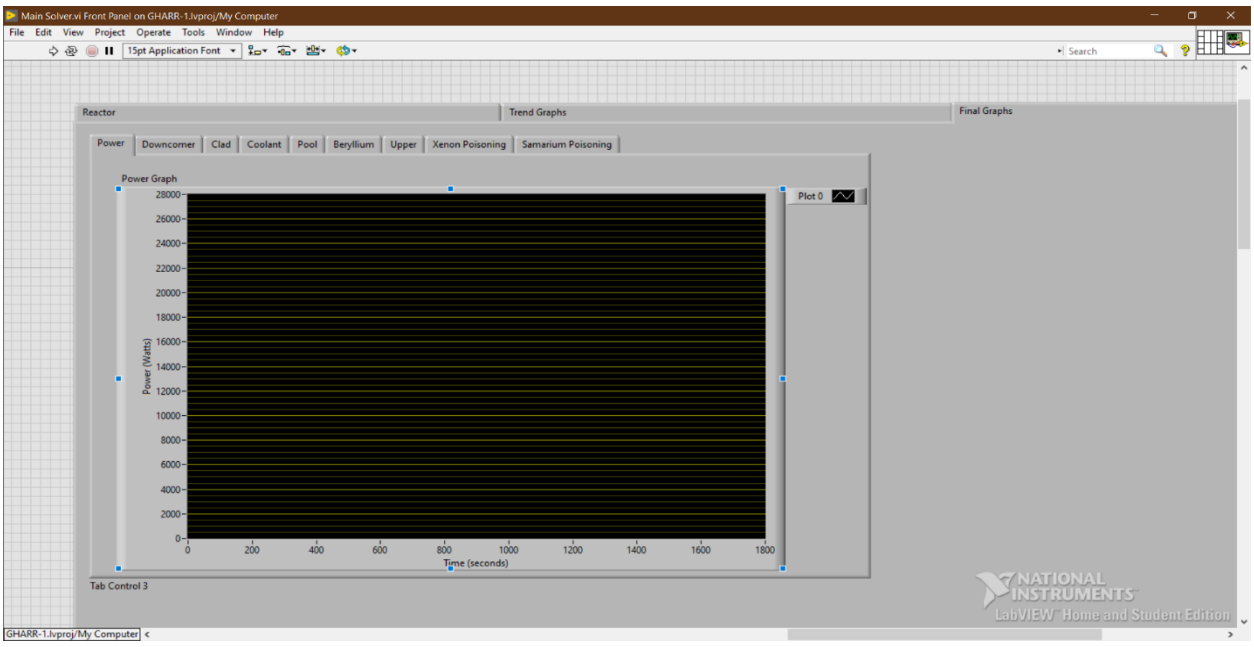


Figure 5.6 The front-end design for the final plots of the simulator

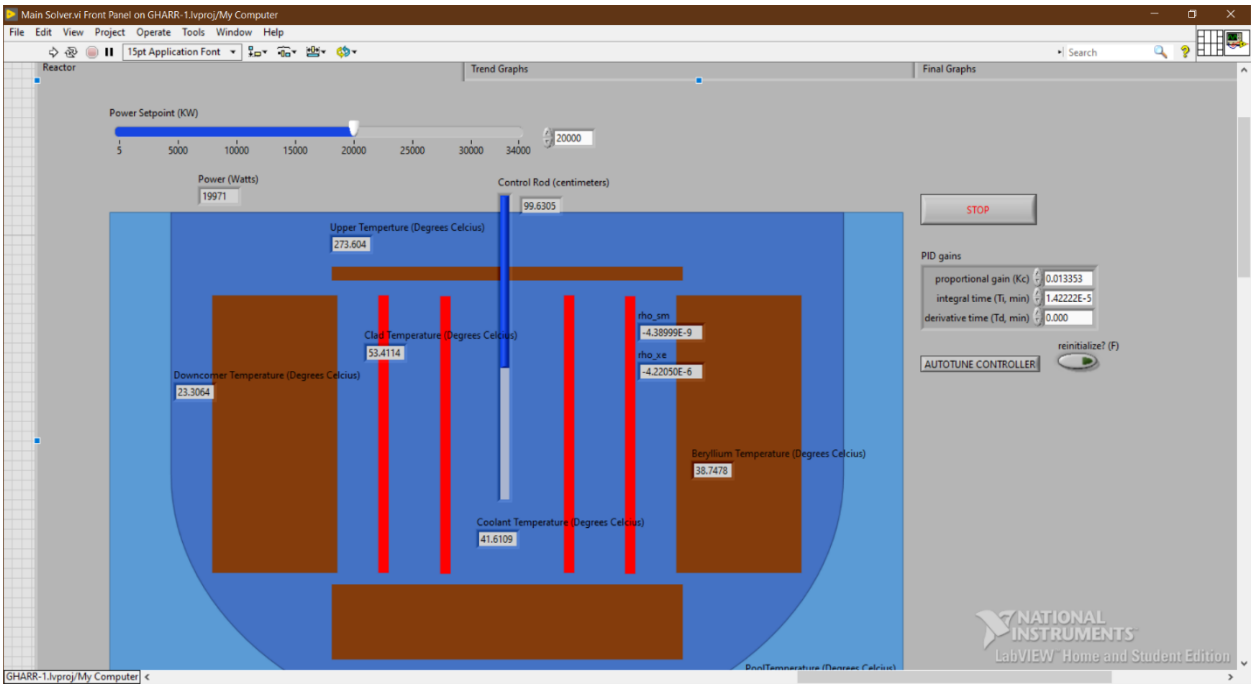


Figure 5.7 The front-end design for the reactor view of the simulator

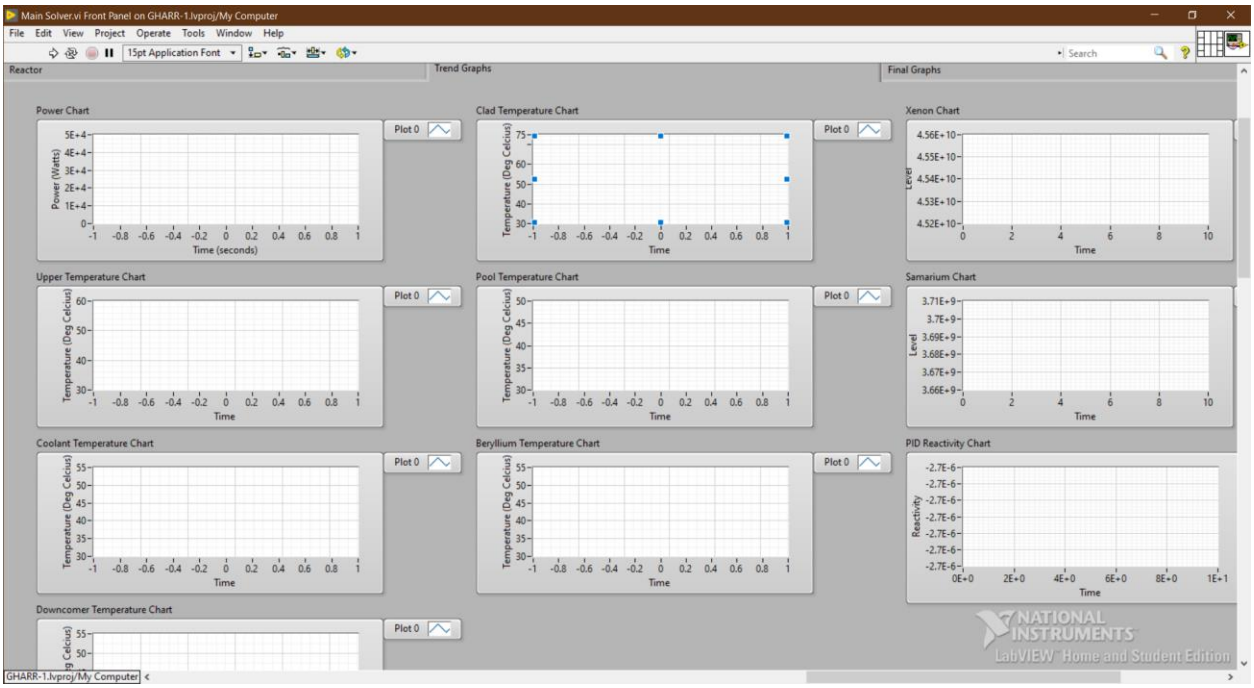


Figure 5.8 The front-end view for the trend plots of the simulator