VERIFYING CONTROL LOGIC SPECIFICATION USING MATHEMATICAL MODELING AND DYNAMIC SIMULATIONS

J. Perras, A. Xing and J. Harber
Atomic Energy of Canada Limited
2251 Speakman Drive, Mississauga, Ontario, Canada L5K 1B2

1. Abstract

For the Advanced CANDU Reactor®* (ACR-1000), process and reactor control design requirements are documented using the control functional specification (CFS). By incorporating use of IEC 61131-3 compliant languages the CFS is intended to be more precise than the traditional control program specification. This paper briefly describes the CFS and its verification through the use of mathematical modeling techniques and dynamic simulation.

2. Introduction

A CFS defines the control functionality of a process or reactor control system in a format that is independent of the implementation platform, making it applicable to software control systems, programmable logic controllers (PLCs), hardwired logic, etc. The control logic in a CFS is specified in terms of IEC 61131-3 supported languages, as either function block diagrams (FBD) or sequential function charts (SFC). The IEC 61131-3 standard also includes structured text, ladder diagram and instruction list languages, which are not normally part of the CFS. The use of IEC 61131-3 compliant languages reduces the ambiguity in the specification, as the syntax is standardized and may be carried over from device to device, system to system, or organization to organization.

In previous CANDU projects, the design for software control systems was specified using a program specification, wherein the requirements were stipulated in plain English (in conjunction with conditional constructs and mathematical equations where appropriate). Experience with the program specification has shown that the written requirements can be incomplete or misinterpreted due to the limitations of the language. The CFS is an effort to increase the precision in the control specification.

In addition, the move to IEC 61131-3 standardized methods makes the control specifications easily testable, as tools are readily available to implement languages such as function block diagrams or sequential function charts. As such, verification activities can be carried out immediately at the specification level, before the controls are implemented.

In order to verify the control specifications, mathematical models were developed for the process system to be used in conjunction with the control model. Process components were modelled using industry standard formulations such as ASME or IAPWS. Mathematical models were implemented in computer simulation software (in this case, MATLAB and Simulink software). This software is used to integrate the mathematical process model with the control design to simulate the dynamic response of the system.

---

* Advanced CANDU Reactor® (ACR-1000®) is a registered trademark of Atomic Energy of Canada Limited (AECL).

From these simulations, we may verify the control specifications, and obtain initial values for control parameters (e.g., gains and time constants within a PID controller). These values may be further tuned later in the design through the use of validated thermalhydraulic computer codes, such as CATHENA or MODTURC, before implementation or commissioning.

3. Control Functional Specification

As previously indicated, the CFS is based on the IEC 61131-3 standard languages for logic programming. The IEC standard was adopted to increase the precision in the control specification.

Central to the control functional specification is the function block diagram, a visual method of displaying the relationship from inputs (e.g., from switches or transmitters) to outputs (e.g., a controller or actuator). The FBD is similar to the method used to depict electronic circuits and combinatorial logic. The list of standard function blocks used for CFS is a subset of function blocks defined in the IEC 61131-3 standard. This list includes the following types of function blocks:

- Mathematical blocks (e.g., Absolute Value, Add, Subtract, Multiply, Divide);
- Logical blocks (e.g., AND, OR, XOR);
- Latches (e.g., Set-Reset, Reset-Set), and
- Timing blocks (e.g., On-delay timer, Clock).

Figure 1 illustrates an instance of an FBD, as utilized in the moderator temperature control specification.

The CFS also includes use of sequential function charts. The SFC is a graphical technique for describing the sequential behaviour of a control program. SFC depicts the control logic in terms of states and transitions between states. A state may contain any number of actions or functions; a transition between states is associated with a condition which, when true, causes the succeeding state to become active.

Figure 2 demonstrates a simple SFC.

Due to the nature of these graphical languages, the equivalence between FBD and SFC is imperfect. This motivates the decision to include both languages as part of the CFS: some constructs are very difficult to implement using the FBD, but may be trivial to describe using SFC, or vice versa. The combination of languages can be used coincidently to create concise and precise control specifications.

In addition, reviewability is increased through the adoption of these languages. The visual diagrams incorporate standard blocks and syntax, allowing for straightforward comprehension of the control specification.

4. Model

To prototype the CFS process, the ACR-1000 moderator system was used. The moderator system was selected as it is sufficiently simple (in terms of process dynamics) and the system necessitates real-time control. This work is a typical example of a control loop in the ACR-1000 design where we have applied modern engineering tools to the traditional CANDU design development of process control systems.

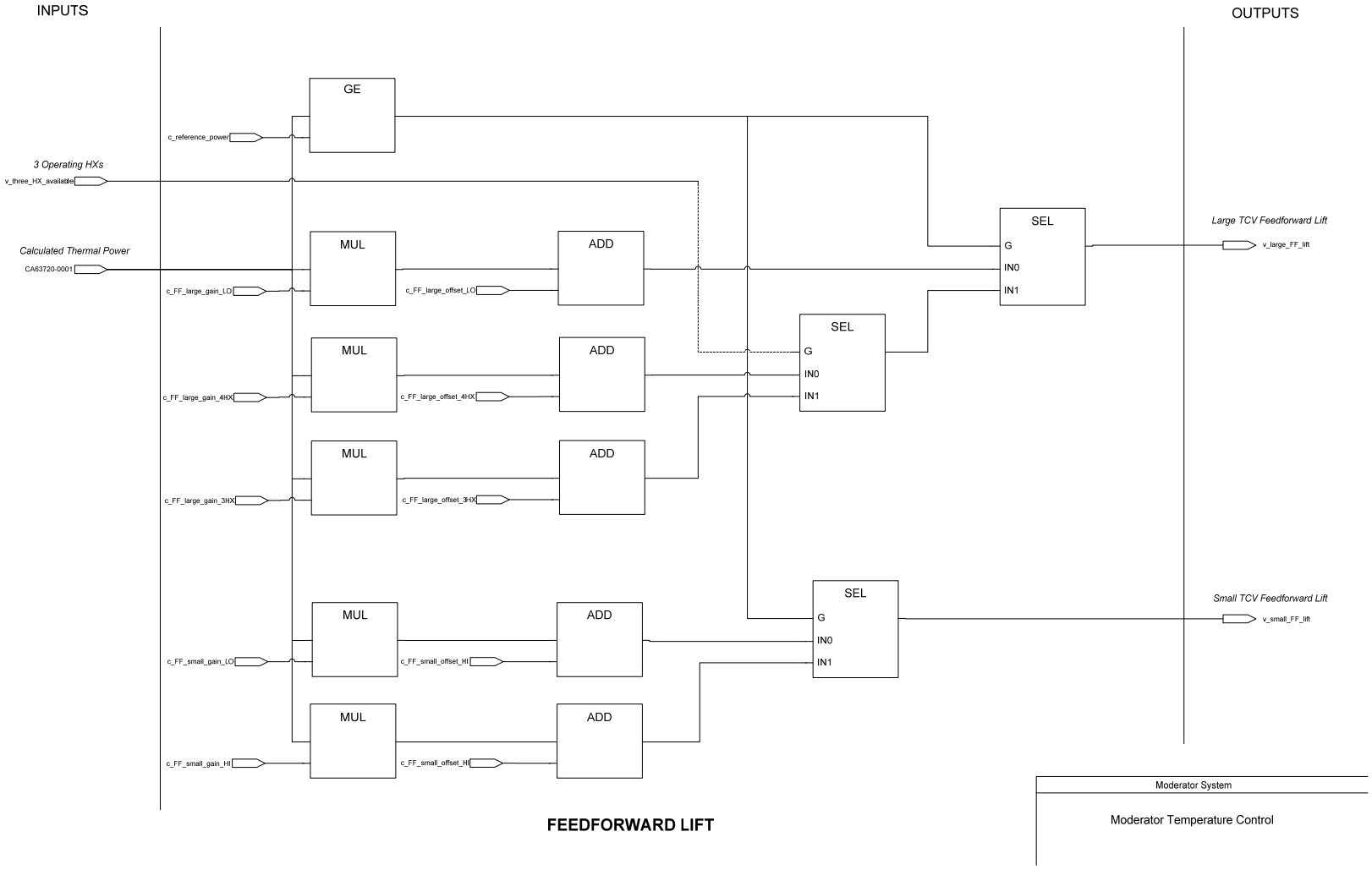

Figure 1 – Function Block Diagram Example

In brief, the D_2O moderator in the calandria regulates the speed of the neutrons produced by nuclear fission in the reactor core to promote additional fission reactions. One of the primary functions of the moderator control system is to maintain the temperature of the inventory in the calandria by removing the heat transferred to the D_2O moderator.

To accomplish the heat removal and moderator temperature control, the D_2O is circulated through the calandria vessel, moderator pumps and heat exchangers. Cooling water flow through the heat exchangers is regulated by temperature control valves (TCVs) which are modulated by the control system. The control system functionality was specified in the CFS format, using FBD to describe the control action.

A model was developed to simulate the dynamics of the system and its control under a variety of conditions and scenarios. The purpose of the model was to test the applicability, stability and robustness of the control algorithm and to estimate control parameters (such as PID values, lifts, etc). This was done to verify the control in an offline tool before spending any substantial time developing the software (or hardwired logic) or otherwise implementing the logic into a real control system. The model was not intended to substitute for a safety analysis code; the results obtained are preliminary and have been used strictly as an aid to the control design. Safety analysis/commissioning/etc may be done later as a part of design verification and validation activities.

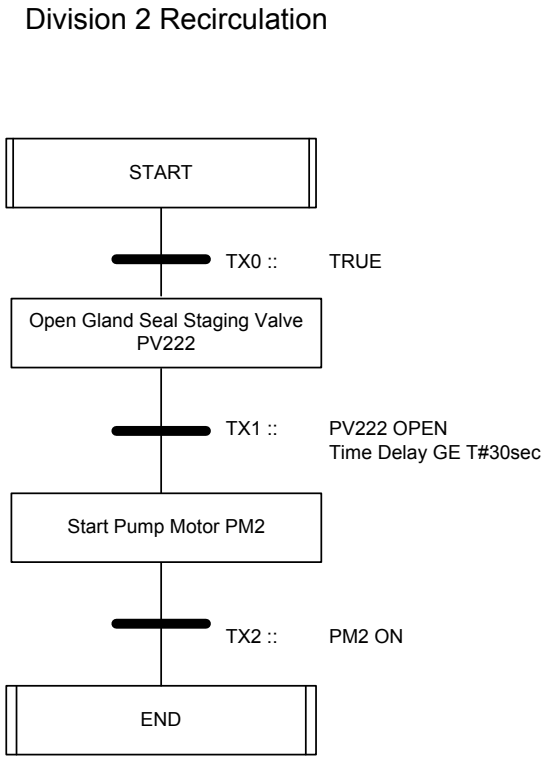

Figure 2 – Sequential Function Chart Example

The model for the moderator system consists of a representation of the control logic as described in the CFS and a model of the process system, which includes mathematical depictions of the calandria, heat exchanger, valves and piping.

Both the control and process system models were developed in the MATLAB/Simulink software environment. MATLAB is a high-level computing language used for data analysis and numeric computation. Simulink is a graphical interface for MATLAB that is used for simulation and model-based design of dynamic systems. Simulink is used to construct a model of a physical system by selecting and customizing function or blocks from the libraries contained within the software. The dynamic response of a system can be observed and analyzed by simulating the model created in Simulink.

Control Model

To implement the control algorithms – exactly as indicated by the CFS – Simulink library blocks were used to create replicas of the IEC 61131-3 standard function blocks. These standard function blocks were then organized as per the CFS to obtain a model of the control logic. The control model may be simulated on its own, as a discrete module, by supplying virtual inputs to the control and observing the outputs. However, the effectiveness of the control logic is best evaluated when it is integrated with the plant model to form a closed-loop system. Figure 3 exhibits part of the control model, as developed in Simulink.

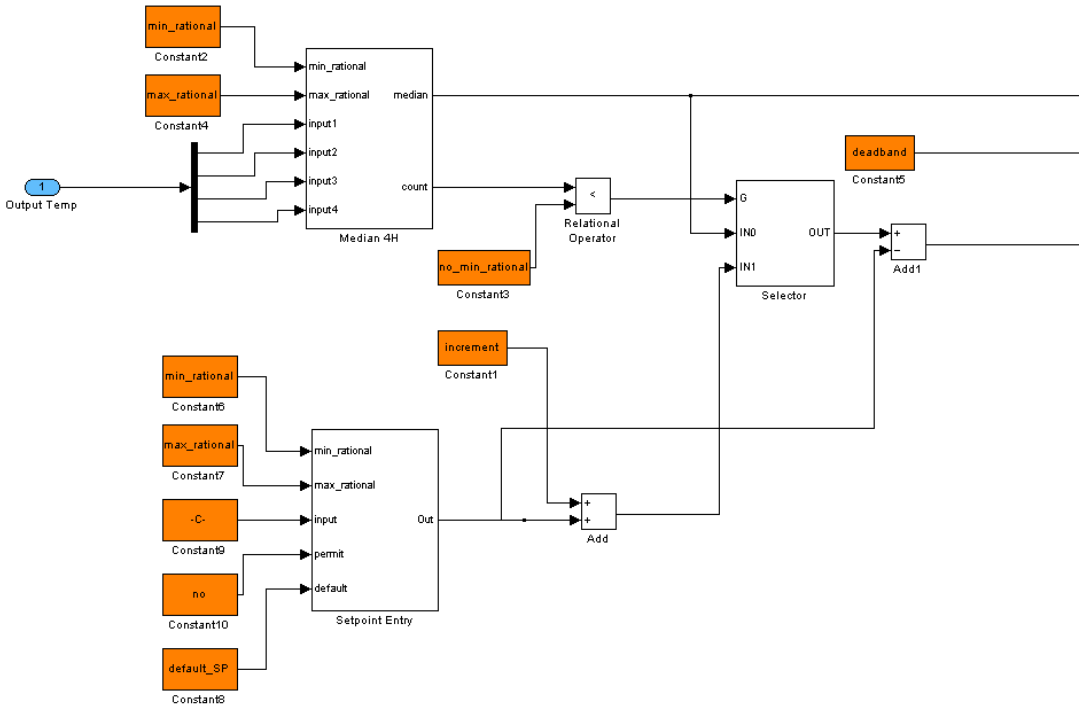

Figure 3 – Moderator Control Model Component

Plant Model

In order to predict the closed loop response of the system and evaluate its effectiveness, a representation of the process was required. To this end, process components were modelled mathematically from a variety of different sources, either using industry standard formulations, models obtained from validated codes or from mathematics/physics derivations. The moderator plant model consists of heat exchangers, the calandria vessel, control valves and piping.

The moderator system heat exchangers were represented by a standard model consisting of coupled partial differential equations (PDEs) (Equation 1).

$$m_h c_h \frac{\partial T_h}{\partial t} + W_h c_h L \frac{\partial T_h}{\partial x} = UA(T_h - T_c) \qquad (1)$$

$$m_c c_c \frac{\partial T_c}{\partial t} - W_c c_c L \frac{\partial T_c}{\partial x} = UA(T_c - T_h)$$

Per the model, the amount of heat transferred is a function of the involved temperatures (T_c, T_h), masses (m_c, m_h), flow rates (W_c, W_h), physical dimensions (length L, surface area A) and various coefficients (specific heats c_c, c_h and heat exchange coefficient U); where subscript h denotes hot (primary) side and subscript c denotes cold (secondary) side. In order to simulate these equations in software, the heat exchangers were nodalized per finite-difference methods (i.e., PDEs were discretized in space, resulting in a system of ordinary differential equations). Any number of nodes may be used (additional nodes result in increased accuracy at the expense of increased simulation time), but adequate results were obtained with as few as ten (Equation 2).

$$\frac{dT_{hn}}{dt} = \frac{W_h}{m_h}\frac{T_{hn-1} - T_{hn}}{\Delta x} + \frac{UA}{m_h c_h}(T_{cn} - T_{hn}) \qquad (2)$$

$$\frac{dT_{cn}}{dt} = \frac{W_c}{m_c}\frac{T_{cn+1} - T_{cn}}{\Delta x} + \frac{UA}{m_c c_c}(T_{hn} - T_{cn})$$

for n = 1,2,...,N nodes; where $\Delta x = N^{-1}$

The calandria assembly consists of a mass of D_2O which accepts a fraction of the heat load generated by the fission reaction in the fuel. To regulate the temperature at an acceptable level, inventory is circulated through heat exchangers where it is cooled before being reintroduced to the calandria. Similar to the heat exchanger, the calandria was nodalized and modelled in two dimensions. Parameters relevant to the calandria model are the heat loads, inlet/outlet flow rates, fluid densities and masses.

Valves, instrumentation and system piping were represented using simple first order approximations as provided in a number of sources.

Component models were connected together to obtain a representation of the process system, depicted in Figure 4.

The system model was then used in conjunction with the control model to create a closed loop system. The system is then simulated by varying the reactor power (and

subsequently, the moderator heat load) or various other system parameters. The Simulink model of the closed loop system is shown in Figure 5.

Constants associated with the plant model (e.g., heat transfer coefficients, instrument time constants and uncertainties, etc) were taken from either manufacturer's records or commissioning data from previous generation CANDU reactors. Should more realistic values become available, these constants may be adjusted to obtain a more true-to-life model.

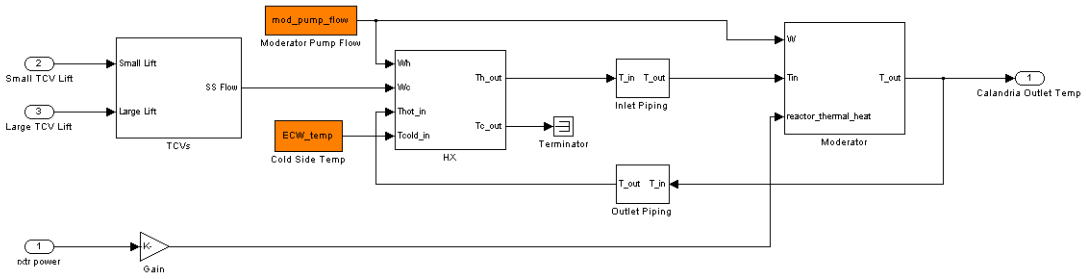

Figure 4 – Moderator Plant Model Overview

5. Simulation

To examine the response of the control under a variety of conditions, a number of test cases were defined. The test cases were intended to reflect the operating modes of a commissioned plant, including power manoeuvres, reactor trips, failure of components (e.g., valves, heat exchangers). These cases were simulated under a variety of control tuning profiles to contrast the performance of the system under various configurations.

Results obtained through model development and simulation of the moderator system and its control are iteratively used as inputs to the control design lifecycle. The result is a robust control that can respond quickly and appropriately to various transients and operating conditions.

While the simulation code has not been validated, steady state temperature profiles from the nodalized moderator model were similar to those obtained through use of the MODTURC code. The control algorithm developed through the CFS has been shown to respond well to the variety of test cases. A more realistic model can be constructed as more is learned about the system and components (e.g., through procurement data, test data), which will result in a tighter control.

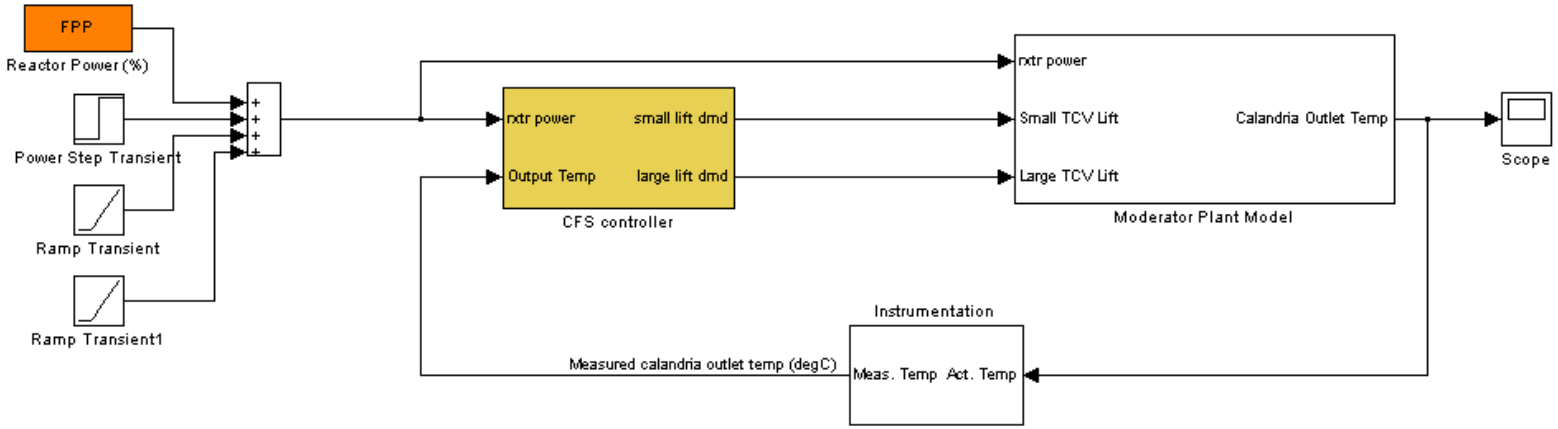

Figure 5 – Closed Loop Moderator System Model Overview

6. Conclusions

The control functional specification (CFS) is used to describe the control design for the ACR-1000. The CFS incorporates the use of IEC 61131-3 languages to unambiguously stipulate the control requirements. Requirements developed for the CFS are verified using mathematical modeling and dynamic simulations prior to implementation.

7. References

[1] International Electrotechnical Commission, Programmable Controllers Part 3: Programming Languages, IEC61131-3, 2003

[2] Harber, J., et al., "Documenting Control System Functionality for Digital Control Implementations," IAEA Technical Meeting, Chatou, France, September 2005