SIMFIX- An Integrated Supervisor-Simulation Environment

Marcelo A. Trindade, Marco A. Meggiolaro, Moisés H. Szwarcman, Raul Q. Feitosa, Renato V. Lima
Catholic University of Rio de Janeiro
Department of Electrical Engineering
R. Marques de São Vicente 225 - 22 453-099 Rio de Janeiro - Brazil

3rd International Electronic Engineering Congress - INTERCON' 96 - August 11-17th - Trujilo, Peru

Abstract

This paper describes a tool (SIMFIX) addressed to control engineers and plant floor personal. It integrates a general purpose simulation package and a SCADA software in a single environment. With SIMFIX it is possible to build process models, simulate them and monitor the simulation using all the features available in a widely used SCADA software.

INTRODUCTION

Distributed Control Systems (DCS) [1] have been evolving rapidly since the mid-1980s thanks to its ability to improve accuracy, operability, and the ease of control strategies modifications in industrial plants. The DCS were initially built around proprietary hardware and software. With the increasing computational power of the personal computers, it became viable to use industry-standard computer hardware to develop cost effective DCSs. This encouraged the software industry to develop high quality inexpensive scan, control, alarm, and data acquisition (SCADA) packages.

One can find nowadays many SCADA packages from different software manufacturers, which incorporates monitoring, supervisory control, alarming control and reporting functions. Powerful tools to develop a friendly Man Machine Interface are also commonly available in the SCADA packages.

Simulation has played an important role in process engineering, control engineering and process operation. Taking advantage of highly precise mathematical models developed for physical, chemical, nuclear and many other kinds of processes, simulation allows to test and refine control strategies before having the process physically implemented. Sophisticated simulation tools became available in the last decade. Some of those tools are process specific, while others are general purpose.

The commercially available simulation packages allow the user to build a graphic interface to follow the evolution of the simulation along the time.

One of the most appealing features of SCADA packages is the ease the user has to design a Man Machine Interface for plant floor operator. Those interfaces are commonly designed to mimic the conventional panels, and can include some new features such as animation. The industrial plants that use Man Machine Interfaces are in general very different from those use by control engineers.

The tools available to build a graphic interface is quite different in those case, because they address different users.

The most SCADA software on the other side have a very limited simulation capability.

The SIMFIX [7] Supervisor Simulation package integrates in a single environment the functions of a SCADA software and the full potential of a simulation software. With SIMFIX it is possible to build process models exploring all the features of a general purpose simulation package, the MATLAB /SIMULINK [5,6] from Math Works Inc., and to build a Man Machine Interface familiar to the plant floor personal using a widely used SCADA software, the FIX DMACS [3,4] from Intellution.

For training operators one can build that way a flexible environment where a SCADA software monitors a virtual process that is being simulated by SIMULINK.

It is also possible to have a hybrid configuration, where part of a process is simulated and part is physically existent. It is very convenient when changes in process and/or in control strategies are to be tested.

SIMFIX automates the linking processes between both applications using an industry-standard data exchange protocol -DDE - supported by Windows, without requiring a deep knowledge about how that protocol works.

SOME FEATURES OF FIX DMACS AND MATLAB/SIMULINK

FIX DMACS for MS-Windows is an industry automation software that offers solutions for small installations or for large, networked configurations.

The core software performs the three most basic functions: data acquisition, data management and data presentation. These functions provide the basis for all industrial automation programs allowing specific applications to perform their assigned tasks.

FIX provides monitoring, supervisory control, alarming and control functions. It has the ability to create reports of critical systems and process information required by many plants that must report or store real-time data for later analysis. The archived data represents a powerful tool for process correction and optimization.

The open architecture of FIX system allows engineers to write software applications that solve specific problems and communicate with the SCADA applications developed based on FIX. Through the communication mechanisms supported by Windows (DDE, OLE, ODBC) it is also possible to exchange information with widely used software.

FIX DMACS incorporates two basic principles: true distributed processing and on-demand data transfer. It allows plants to distribute critical functions among all computers on the network.

FIX software can perform time-based or/and exception-based processing. This feature allows engineers to use the best processing strategy for each data point in a plant.

SIMULINK, as an extension to MATLAB, is a program for simulating dynamic systems. It has two phases of use: model definition and model analysis.

Models are created and edited in a new class of windows called *block diagram* windows defined by SIMULINK.

Built-in analysis tools are provided by SIMULINK for various simulation algorithms. There are tools for extracting linear models of systems, finding equilibrium points. The progress of a simulation can be viewed while simulation is running.

DYNAMIC DATA EXCHANGE (DDE)

Dynamic Data Exchange (DDE) is a mechanism for communication between applications supported by Windows.

Two applications running on Windows can communicate sending messages to each other. In the DDE concept there is always a server application able to send data and client applications, which gets the information sent by a server.

A client starts a DDE communication by sending a message to all other currently running applications to request data. In this message the general class of data required by the client is identified. Only the server application that owns this class of data answer the message by sending a message with the requested data.

The same program running on Windows may be the client in a communication with a program A and the server while communicating with a program B.

There are three basic types of DDE links:

*a) The Cold Link***:** a communication on a cold link begins when a client pass a message of initialization, identifying the server application and the document containing the desired data and the data itself. The identified server answers the client with a acknowledge message. Then, the client sends a message requiring a specific data item. The server sends a message containing the required data or informing that it is not able to provide the data. The DDE communication goes on this way till the client and the server send to each other a terminating message.

b) The Hot Link: in the hot link the initialization procedure is the same used as in the cold link. The operation procedure is quite different. In the hot link the server notifies the client every time an item's value changes. This is done through a warning message containing the updated value. The client may either acknowledge this message or not. When the client does not want anymore to be informed of changes on data item, it sends to the server a termination message. The advantage of this type of link is that the client does not send continually requesting messages to the servers.

*c) The Warm Link***:** it combines elements from the cold link and from the hot link, in such a way that the user can choose the kind of link he wants, according to each operation.

THE SIMULINK / FIX LINK

The SIMFIX program was developed to automate the linking process between FIX and SIMULINK. It was written in Visual Basic, due to the many facilities present in this language for DDE connections.

FIX works with DDE hot links while MATLAB works with DDE cold links. SIMFIX acts as an intermediate program, communicating with FIX through hot links and with SIMULINK through cold links.

The SIMFIX gets messages generated by SIMULINK/FIX and sends it to FIX/SIMULINK. in a cyclic fashion.

In order to establish the link between both programs using SIMFIX, some steps must be followed by the user:

Steps within FIX

The first step is to create the database in FIX in the usual way. The only restriction is that the variables in the FIX Database used by MATLAB/SIMULINK must be assigned to analog/digital input, analog/digital output or analog/digital register blocks, which are the only ones that support DDE connections.

The second step is to generate a text file with a WDB extension in the FIX Database Builder, which will contain the names of all the variables to be linked.

The user can draw the synoptic representing the process in FIX using all the resources available and any of the database blocks defined in step one.

Steps within SIMULINK

The only step to be carried out within SIMULINK is to construct the model to be simulated. The simulation will actually be started by MATLAB, which will exchange data with SIMULINK by means of a unique vector, whose elements are the state variables linked. The only concern in this step is assigning to the variables that will be linked as parameters the same name defined in FIX Database. The equivalence of the block names in FIX Database and the variable names in SIMULINK bond graph is used by SIMFIX to establish a consistent data exchange between both applications.

Steps within SIMFIX

The first step to be followed is to run SIMFIX. The SIMULINK and FIX software are started automatically by SIMFIX. The FIX View module is also started, and a opened on the screen. The SIMFIX's main screen (fig.1) also appears.

The second step is to open the WDB extension file created in the FIX Database by the '*Open...*' button in SIMFIX's main screen (fig. 1). The SIMFIX automatically creates as many boxes in its main screen as the number of variables read from the WDB text file. Those boxes show the same names of the blocks defined in the database, and their initial value may be entered manually if desired. The initial values of the first derivative of the state variables may also be set up. The SIMFIX creates as many boxes as necessary in its main screen to show the variable values.

The third step consists in linking the SIMFIX to FIX software. It is done by assigning the DDE addresses to the blocks defined in FIX database. The DDE addresses to be assigned are the ones from the boxes created in SIMFIX's main screen. This procedure is automatically done by pressing the *'Update PDB'* button. This procedure automatically opens the FIX Database Editor, and updates the DDE links of all blocks. The database file is also saved and the editor closes automatically. This procedure is required only in the first time that a specific group of blocks is used.

In SIMFIX's options screen it is possible to define the updating time period, which is the period between two consecutive updates between the SIMULINK and FIX variables. One may also choose a precision for the variables in the simulation, and which integrating method is to be used - Runge-Kutta, Gear, Adams, Euler or Linear Simulator. For more information on these methods, please refer to the SIMULINK Manual [6].

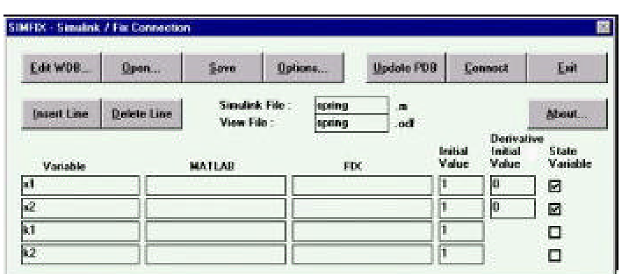

Fig.1 - SIMFIX's main screen

In the fourth step the SIMFIX software is linked to SIMULINK and the process is started. The name of the SIMULINK file, which contains the simulation bond graph, must be entered in SIMFIX's main screen. The connection starts by pressing the '*Connect*' button. The SIMFIX software will then manage to update the values of the state variables and parameters between SIMULINK and FIX, showing them in its main screen in real time. The mechanism behind this updating is the automatic creation by SIMFIX of a batch file executed by MATLAB. This file sets a simulation loop that:

- gets the values from FIX, monitored or altered by the user in the FIX *View* window,
- updates those values in the SIMULINK model,
- solves the bond graph for one more time step, and
- transfers the new parameter values from SIMULINK back to the FIX system.

The *'Disconnect'* button in SIMFIX is used to stop the process. By pressing it, a message is sent to MATLAB to stop the simulation loop. The *'Exit'* button automatically stops the process, closes all DDE links and also the SIMFIX, MATLAB/SIMULINK and FIX programs.

CONCLUSIONS

The SIMFIX is a powerful tool to build an environment combining the features of a simulation package and of SCADA software.

To use SIMFIX it is not required to know more than the basic concepts of both software. In the same way the user does need to know about the DDE protocols - the basic communication mechanism used by SIMFIX.

SIMFIX offers also the possibility to work with a partially implemented and partially simulated process.

REFERENCES

[1] M. Considine, Process/Indusrial Instruments & Controls Handbook, McGraw-Hill, 1993.

[2] Petzold, Programming Windows, Microsoft Press, 1990.

[3] Intellution Inc., FIXMMI System Setup-Introduction to FIX , 1995.

[4] Intellution Inc. FIXMMI - Advanced Tools Manual, 1995.

[5] Math Works Inc., High Performance Numeric Computation and Visualization Software- Realease Notes, Version 4.2, 1994.

[6] Math Works Inc., Simulink - Dynamic System Simulation Software - User's Guide, 1994.

[7] M. A. Meggiolaro, M. A. Trindade, R. V. Lima, SIMFIX - Conexão SIMULINK/FIX, Internal Report of the Dept. of Electrical Engineering, Catholic University of Rio de Janeiro, 1995.